Basler aviator

USER'S MANUAL FOR CAMERA LINK CAMERAS

Document Number: AW000830
Version: 09   Language: 000 (English)
Release Date: 05 March 2018

For customers in the USA

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

You are cautioned that any changes or modifications not expressly approved in this manual could void your authority to operate this equipment.

The shielded interface cable recommended in this manual must be used with this equipment in order to comply with the limits for a computing device pursuant to Subpart B of Part 15 of FCC Rules.

For customers in Canada

This apparatus complies with the Class A limits for radio noise emissions set out in Radio Interference Regulations.

Pour utilisateurs au Canada

Cet appareil est conforme aux normes Classe A pour bruits radioélectriques, spécifiées dans le Règlement sur le brouillage radioélectrique.

Life Support Applications

These products are not designed for use in life support appliances, devices, or systems where malfunction of these products can reasonably be expected to result in personal injury. Basler customers using or selling these products for use in such applications do so at their own risk and agree to fully indemnify Basler for any damages resulting from such improper use or sale.

Contacting Basler Support Worldwide

Europe, Middle East, Africa

Basler AG
An der Strusbek 60–62
22926 Ahrensburg
Germany

Tel. +49 4102 463 515
Fax +49 4102 463 599

support.europe@baslerweb.com

The Americas

Basler, Inc.
855 Springdale Drive, Suite 203
Exton, PA 19341
USA

Tel. +1 610 280 0171
Fax +1 610 280 7608

support.usa@baslerweb.com

Asia-Pacific

Basler Asia Pte. Ltd.
35 Marsiling Industrial Estate Road 3
#05–06
Singapore 739257

Tel. +65 6367 1355
Fax +65 6367 1255

support.asia@baslerweb.com

www.baslerweb.com

Table of Contents

1 Specifications, Requirements, and Precautions

This chapter lists the camera models covered by the manual. It provides the general specifications for those models and the basic requirements for using them.

This chapter also includes specific precautions that you should keep in mind when using the cameras. We strongly recommend that you read and follow the precautions.

1.1 Models

The current Basler aviator Camera Link® camera models are listed in the top row of the specification table on the next page of this manual. The camera models are differentiated by whether the camera's sensor is mono or color.

Unless otherwise noted, the material in this manual applies to all of the camera models listed in the specification table. Material that only applies to a particular camera model or to a subset of models, such as to color cameras only, will be so designated.

1.2 General Specifications

Specification	avA1000-120km	avA1000-120kc	avA1600-65km	avA1600-65kc
Sensor Resolution (H x V pixels) nominal: maximum:	 1024 x 1024 pixels 1040 x 1040 pixels	 1024 x 1024 pixels 1036 x 1036 pixels	 1600 x 1200 pixels 1640 x 1240 pixels	 1600 x 1200 pixels 1636 x 1236 pixels
Sensor Type	ON Semiconductor® KAI-01050 Progressive scan CCD		ON Semiconductor® KAI-02050 Progressive scan CCD	
Optical Size	1/2"		2/3"	
Effective Sensor Diagonal	5,5 mm	5,5 mm	5,5 mm	5,5 mm
Pixel Size	5.5 µm x 5.5 µm			
Max Frame Rate (at nominal resolution)	120 fps		67 fps	
Mono/Color	Mono	Color	Mono	Color
ADC Bit Depth	12 bit			
Data Output Type	Camera Link Base Configuration			
Camera Link Clock Speed	20 MHz, 32.5 MHz, 40 MHz, 48 MHz, or 65 MHz (selectable)			
Camera Link Tap Geometry	1X-1Y, 1X2-1Y or 1X-2YE			
Pixel Format	Mono Models: Mono 8 Mono 10 Mono 12		Color Models: Bayer GR 8 Bayer GR 10 Bayer GR 12	
Synchronization	Via hardware trigger, via software trigger, or free run			
Exposure Time Control	Trigger width or timed			
Camera Power Requirements	+12 VDC (± 10%), < 1% ripple			
Power Consumption	5.0 W @ 12 VDC		5.5 W @ 12 VDC	
I/O Lines	12-pin receptacle: 2 input lines (Line 1, Line 2), 1 output line (Output Line 1) 26-pin Camera Link connector: 3 input lines -> CC1, CC2, CC4			
Lens Adapter	C-mount			
Size (L x W x H)	41.2 mm x 62.0 mm x 62.0 mm (without lens adapter or connectors) 56.8 mm x 62.0 mm x 62.0 mm (with lens adapter and connectors)			
Weight	≈ 300 g (typical)			

Table 1: General Specifications

Specification	avA1000-120km	avA1000-120kc	avA1600-65km	avA1600-65kc
Conformity	CE (includes ROHS), FCC, GenICam, IP30, REACH The EU Declaration of Conformity is available on the Basler website: www.baslerweb.com			
Software	Basler pylon Camera Software Suite (version 4.0 or higher) Available for Windows, Linux x86, Linux ARM, and OS X			

Table 1: General Specifications

Specification	avA1900-60km	avA1900-60kc	avA2300-30km	avA2300-30kc
Sensor Resolution (H x V pixels) nominal: maximum:	 1920 x 1080 pixels 1960 x 1120 pixels	 1920 x 1080 pixels 1956 x 1116 pixels	 2330 x 1750 pixels 2360 x 1776 pixels	 2332 x 1752 pixels 2356 x 1772 pixels
Sensor Type	ON Semiconductor® KAI-02150 Progressive scan CCD		ON Semiconductor® KAI-04050 Progressive scan CCD	
Optical Size	2/3"		1"	
Effective Sensor Diagonal	5,5 mm	5,5 mm	5,5 mm	5,5 mm
Pixel Size	5.5 µm x 5.5 µm			
Max Frame Rate (at nominal resolution)	62 fps		31 fps	
Mono/Color	Mono	Color	Mono	Color
ADC Bit Depth	12 bit			
Data Output Type	Camera Link Base Configuration			
Camera Link Clock Speed	20 MHz, 32.5 MHz, 40 MHz, 48 MHz, or 65 MHz (selectable)			
Camera Link Tap Geometry	1X-1Y, 1X2-1Y or 1X-2YE			
Pixel Format	Mono Models: Mono 8 Mono 10 Mono 12		Color Models: Bayer GR 8 Bayer GR 10 Bayer GR 12	
Synchronization	Via hardware trigger, via software trigger, or free run			
Exposure Time Control	Trigger width or timed			
Camera Power Requirements	+12 VDC (± 10%), < 1% ripple			
Power Consumption	5.5 W @ 12 VDC		6.0 W @ 12 VDC	
I/O Lines	12-pin receptacle: 2 input lines (Line 1, Line 2), 1 output line (Output Line 1) 26-pin Camera Link connector: 3 input lines -> CC1, CC2, CC4			
Lens Adapter	C-mount			
Size (L x W x H)	41.2 mm x 62.0 mm x 62.0 mm (without lens adapter or connectors) 56.8 mm x 62.0 mm x 62.0 mm (with lens adapter and connectors)			
Weight	≈ 300 g (typical)			
Conformity	CE (includes ROHS), FCC, GenICam, IP30, REACH The EU Declaration of Conformity is available on the Basler website: www.baslerweb.com			

Table 2: General Specifications

Specification	avA1900-60km	avA1900-60kc	avA2300-30km	avA2300-30kc
Software	Basler pylon Camera Software Suite (version 4.0 or higher) Available for Windows, Linux x86, Linux ARM, and OS X			

Table 2: General Specifications

1.3 Accessories

Fig. 1: Basler Accessories

Basler’s cooperation with carefully selected suppliers means you get accessories you can trust which makes building a high-performance image processing system hassle-free.

Key Reasons for Choosing Lenses, Cables, and Other Accessories from Basler

- Perfect match for Basler cameras
- One-stop-shopping for your image processing system
- Stable performance through highest quality standards
- Easy integration into existing systems
- Expert advice during selection process

See the Basler website for information about Basler’s extensive accessories portfolio (e.g. cables, lenses, host adapter cards, switches): www.baslerweb.com

1.4 Spectral Response

1.4.1 Monochrome Cameras

The following graph shows the spectral response for all monochrome cameras.

ℹ	The spectral response curve excludes lens characteristics and light source characteristics.

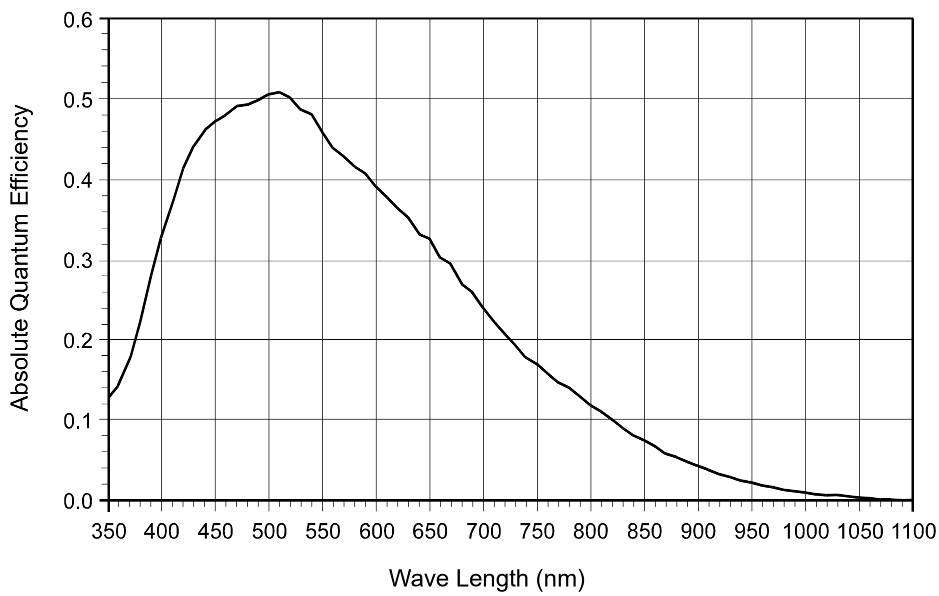


Fig. 2: Mono Camera Spectral Response

1.4.2 Color Cameras

The following graph shows the spectral response for all color cameras.

ℹ	The spectral response curves excludes lens characteristics, light source characteristics, and IR cut filter characteristics. To obtain best performance from color models of the camera, use of a dielectric IR cut filter is recommended. The filter should transmit in a range from 450 nm to 620 nm, and it should cut off from 700 to 1100 nm. A suitable IR cut filter is included in the standard C-mount lens adapter on color models of the camera.

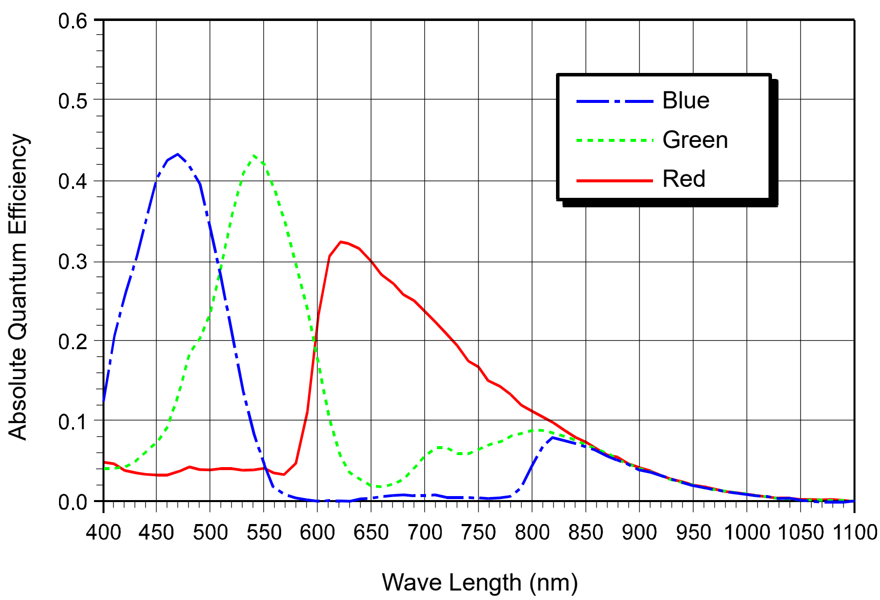


Fig. 3: Color Camera Spectral Response

1.5 Mechanical Specifications

The camera housing conforms to protection class IP30 assuming that the lens mount is covered by a lens or by the cap that is shipped with the camera.

1.5.1 Camera Dimensions and Mounting Points

The cameras are manufactured with high precision. Planar, parallel, and angular sides ensure precise mounting with high repeatability.

Camera housings are equipped with four mounting holes on the front and two mounting holes on each side as shown in the drawings.

For mounting on a tripod, a suitable tripod adapter is available from Basler.

The camera's dimensions in millimeters are as shown in the drawings below.

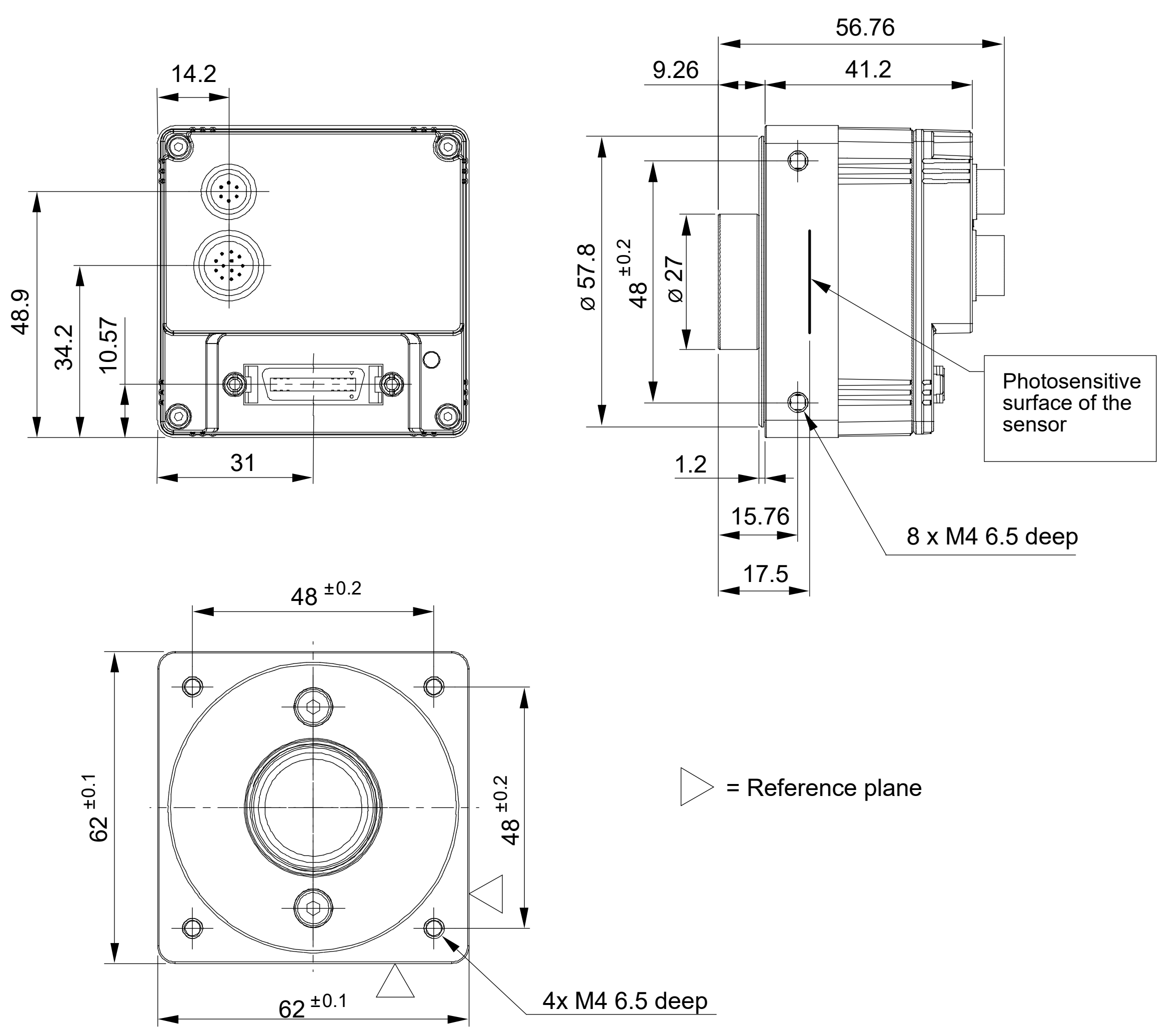


Fig. 4: Mechanical Dimensions (in mm)

1.5.2 Sensor Positioning Accuracy

The sensor positioning accuracy is as shown in the drawings below.

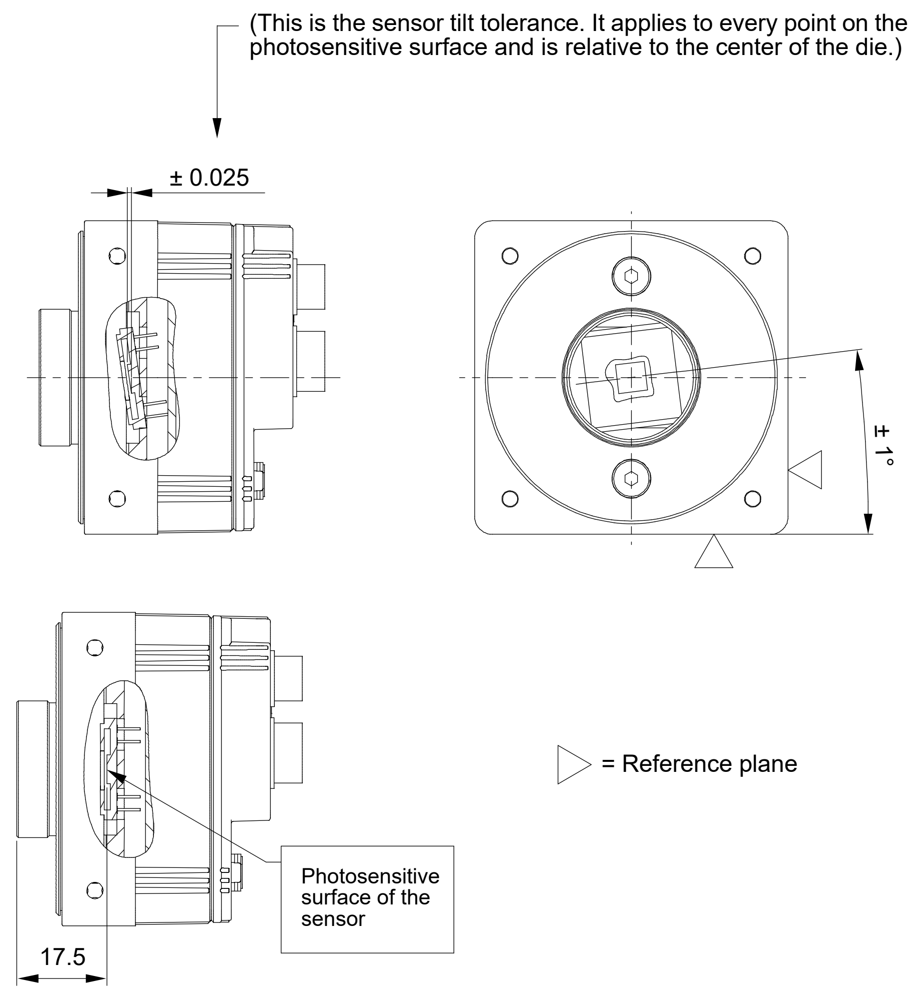


Fig. 5: Sensor Positioning Accuracy (in mm unless otherwise noted)

1.5.3 Maximum Lens Thread Length on Color Cameras

The C-mount lens adapter on color models of the camera is normally equipped with an internal IR cut filter. As shown below, the length of the threads on any lens you use with a color camera must be less than 7.5 mm. If a lens with a longer thread length is used, the IR cut filter will be damaged or destroyed and the camera will no longer operate.

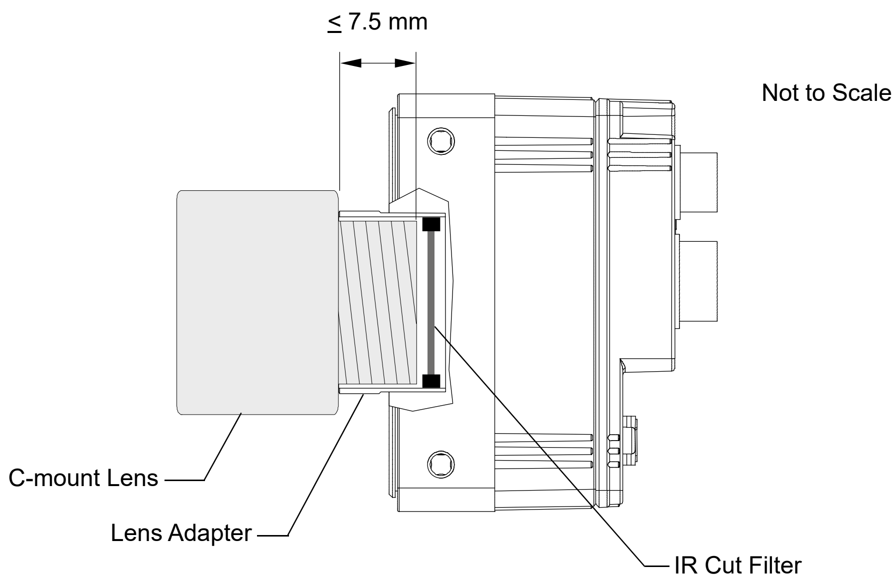


Fig. 6: Maximum Lens Thread Length on Color Cameras

ℹ	C-mount color cameras that do not include an internal IR cut filter, are available on request. Monochrome cameras are not normally equipped with an internal IR cut filter, however, they can be equipped with an internal filter on request.

1.5.4 Mechanical Stress Test Results

Aviator cameras were submitted to an independent mechanical testing laboratory and subjected to the stress tests listed below. The mechanical stress tests were performed on selected camera models. After mechanical testing, the cameras exhibited no detectable physical damage and produced normal images during standard operational testing.

Test	**Standard**	**Conditions**
Vibration (sinusoidal, each axis)	DIN EN 60068-2-6	10-58 Hz / 1.5 mm_58-500 Hz / 20 g_1 Octave/Minute 10 repetitions
Shock (each axis)	DIN EN 60068-2-27	20 g / 11 ms / 10 shocks positive 20 g / 11 ms / 10 shocks negative
Bump (each axis)	DIN EN 60068-2-29	20 g / 11 ms / 100 shocks positive 20 g / 11 ms / 100 shocks negative
Vibration (broad-band random, digital control, each axis)	DIN EN 60068-2-64	15-500 Hz / 0.05 PSD (ESS standard profile) / 00:30 h

Table 1: Mechanical Stress Tests

The mechanical stress tests were performed with a dummy lens connected to a C-mount. The dummy lens was 35 mm long and had a mass of 66 g. Using a heavier or longer lens requires an additional support for the lens.

1.6 Software Licensing Information

1.6.1 LZ4 Licensing

The software in the camera includes the LZ4 implementation. The copyright information for this implementation is as follows:

1.7 Avoiding EMI and ESD Problems

The cameras are frequently installed in industrial environments. These environments often include devices that generate electromagnetic interference (EMI) and they are prone to electrostatic discharge (ESD). Excessive EMI and ESD can cause problems with your camera such as false triggering or can cause the camera to suddenly stop capturing images. EMI and ESD can also have a negative impact on the quality of the image data transmitted by the camera.

To avoid problems with EMI and ESD, you should follow these general guidelines:

- Always use high quality shielded cables. The use of high quality cables is one of the best defenses against EMI and ESD.
- Avoid coiling camera cables. If the cables are too long, use a meandering path.
- Avoid placing camera cables parallel to wires carrying high-current, switching voltages such as wires supplying stepper motors or electrical devices that employ switching technology. **Placing camera cables near to these types of devices may cause problems with the camera.**
- Attempt to connect all grounds to a single point, e.g., use a single power outlet for the entire system and connect all grounds to the single outlet. This will help to avoid large ground loops. (Large ground loops can be a primary cause of EMI problems.)
- Use a line filter on the main power supply.
- Install the camera and camera cables as far as possible from devices generating sparks. If necessary, use additional shielding.
- Decrease the risk of electrostatic discharge by taking the following measures:
  - Use conductive materials at the point of installation (e.g., floor, workplace).
  - Use suitable clothing (cotton) and shoes.
  - Control the humidity in your environment. Low humidity can cause ESD problems.

i	The Basler application note called *Avoiding EMI and ESD in Basler Camera Installations* provides much more detail about avoiding EMI and ESD. This application note can be obtained from the Basler website: www.baslerweb.com

1.8 Environmental Requirements

1.8.1 Temperature and Humidity

Housing temperature during operation:	0 °C ... +50 °C (+32 °F ... +122 °F)
Humidity during operation:	20 % ... 80 %, relative, non-condensing
Storage temperature:	-20 °C ... +80 °C (-4 °F ... +176 °F)
Storage humidity:	20 % ... 80 %, relative, non-condensing

1.8.2 Heat Dissipation

You must provide sufficient heat dissipation to maintain the temperature of the camera housing at 50 °C or less. Since each installation is unique, Basler does not supply a strictly required technique for proper heat dissipation. Instead, we provide the following general guidelines:

- In all cases, you should monitor the temperature of the camera housing and make sure that the temperature does not exceed 50 °C. Keep in mind that the camera will gradually become warmer during the first 1.5 hours of operation. After 1.5 hours, the housing temperature should stabilize and no longer increase.
- If your camera is mounted on a substantial metal component in your system, this may provide sufficient heat dissipation.
- The use of a fan to provide air flow over the camera is an extremely efficient method of heat dissipation. The use of a fan provides the best heat dissipation.

1.9 Precautions

DANGER

Electric Shock Hazard

Risk of Burn or Death.

The power supplies used for supplying

- power to the I/O lines and
- camera power

must meet the Safety Extra Low Voltage (SELV) and Limited Power Source (LPS) requirements.

WARNING

Fire Hazard

Risk of Burn

The power supplies used for supplying

- power to the I/O lines and
- camera power

must meet the Limited Power Source (LPS) requirements.

NOTICE

If the camera power voltage is greater than +13.2 VDC, damage to the camera can result. If the voltage is less than +10.8 VDC, the camera may operate erratically.

Applying power with the wrong polarity can result in severe damage to the camera.

1. Always make sure that the voltage of the camera power is within the specified range.
2. Always make sure that the polarity of the applied voltage is correct.

NOTICE

Incorrect plugs can damage the camera’s connectors.

1. The plug on the cable that you attach to the camera’s 12-pin connector must have 12 female pins. Using a plug designed for a smaller or a larger number of pins can damage the connector.
2. The plug on the cable that you attach to the camera’s 6-pin connector must have 6 female pins. Using a plug designed for a smaller or a larger number of pins can damage the connector.

NOTICE

Making or breaking connections incorrectly can damage the camera.

1. Be sure that all power to your camera and to your host computer is switched off before you make or break connections to the camera. Making or breaking connections when power is on can result in damage to the camera or to the frame grabber.
2. If you can't switch off the power, be sure that:
   a. The camera power plug is the last plug that you plug into the camera when making connections.
   b. The camera power plug is the first plug that you unplug from the camera when breaking connections.

NOTICE

Avoid dust on the sensor.

The camera is shipped with a protective plastic seal on the lens mount. To avoid collecting dust on the camera's IR cut filter (color cameras) or sensor (mono cameras), make sure that you always put the protective seal in place when there is no lens mounted on the camera.

To avoid collecting dust on the camera's IR cut filter (color cameras) or sensor (mono cameras), make sure to observe the following:

- Always put the plastic cap in place when there is no lens mounted on the camera.
- Make sure that the camera is pointing down every time you remove or replace the plastic cap or a lens.

Never apply compressed air to the camera. This can easily contaminate optical components, particularly the sensor.

NOTICE

On color cameras, the lens thread length is limited.

Color models of the camera with a C-mount lens adapter are equipped with an IR cut filter mounted inside of the adapter. The location of this filter limits the length of the threads on any lens you use with the camera. If a lens with a very long thread length is used, the IR cut filter will be damaged or destroyed and the camera will no longer operate. Do not use a lens with a thread length greater than 7.5 mm.

For more specific information about the lens thread length, see Section 1.5.3 on page 11.

NOTICE

Inappropriate code may cause unexpected camera behavior.

1. The code snippets provided in this manual are included as sample code only. Inappropriate code may cause your camera to function differently than expected and may compromise your application.
2. To ensure that the snippets will work properly in your application, you must adjust them to meet your specific needs and must test them thoroughly prior to use.
3. The code snippets in this manual are written in C++. Other programming languages can also be used to write code for use with the Basler pylon Camera Software Suite. When writing code, you should use a programming language that is both compatible with the Basler pylon Camera Software Suite and appropriate for your application. For more information about the programming languages that can be used with the Basler pylon Camera Software Suite, see the documentation included with the pylon package.

NOTICE

Cleaning of the sensor

Avoid cleaning the surface of the camera's sensor if possible. If you must clean it:

- Before starting, disconnect the camera from camera power and I/O power.
- Use a soft, lint-free cloth dampened with a small amount of window cleaner.
- Because electrostatic discharge can damage the sensor, you must use a cloth that won't generate static during cleaning (cotton is a good choice).
- Make sure the window cleaner has evaporated after cleaning, before reconnecting the camera to power.

Cleaning of the housing:

- Before starting, disconnect the camera from camera power and I/O power.
- Use a soft dry cloth. To remove severe stains, use s soft cloth dampened with a small quantity of detergent, then wipe dry.
- Because electrostatic discharge can damage the camera, you must use a cloth that won't generate static during cleaning (cotton is a good choice).

Do not use solvents or thinners. They can damage the surface finish.

Observe the Following Items:

- Do not remove the camera’s product label that contains the serial number.
- Do not open the housing and do not touch the internal components, you may damage them.
- Prevent ingress or insertion of foreign substances into the camera housing. If operated with any foreign substances inside, the camera may fail or cause a fire.
- Do not operate the camera in the vicinity of strong electromagnetic fields. Avoid electrostatic charging.
- Transport the camera in its original packaging only. Do not discard the packaging.

2 Physical Interface

This chapter provides detailed information, such as pin assignments and voltage requirements, for the camera's physical interface. This information is especially useful during your initial design-in.

ℹ	Note that separate power must be supplied: ■ power to operate the camera ("camera power", see Section 2.2.3 on page 24) ■ power to operate inputs and output of the 12-pin receptacle (see Section 2.3.3.1 on page 27 and Section 2.3.4.1 on page 29).

Camera parameters can be changed by these alternative mechanisms (see Section 5 on page 61):

- The Basler pylon Camera Software Suite provides options for changing parameters and controlling the camera by means of the
  - pylon Viewer (a stand-alone GUI) or by employing the
  - API to access the camera from within your software application (see below).
- Direct access to the camera's register structure provides another means for controlling the camera and changing parameters (see Section 5.2 on page 62).

2.1 General Description of the Camera Connections

The camera is interfaced to external circuity via connectors located on the back of the housing (Figure 7 on page 21):

- A 6-pin receptacle used to provide power to the camera. The receptacle is a Hirose micro receptacle (part number HR10A-7R-6PB) or the equivalent.
- A 12-pin receptacle used to provide power and access to the camera's I/O lines. The receptacle is a Hirose micro receptacle (part number HR10A-10R-12P) or the equivalent.
- A 26-pin, 0.050" Mini D Ribbon (MDR) female connector used to transmit video data, control signals, and configuration commands.

A power, status and error LED indicator is located on the back of the camera (Figure 7 below).

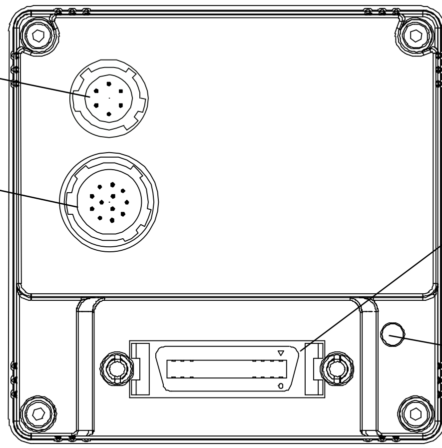


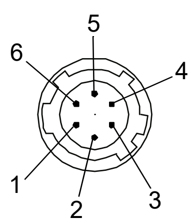
Fig. 7: Camera Connectors and LED Indicator

2.2 Camera Power Interface and Requirements

2.2.1 6-Pin Receptacle

The 6-pin receptacle is used to supply camera power. The pin assignments and pin numbering for the receptacle are as shown in Table 2.

For information about the connecting cable and its installation, see Section 2.2.2 on page 22. For more information about voltage requirements, see Section 2.2.3 on page 24.



Pin	Designation
1	+12 VDC Camera Power (+12 VDC ± 10%) *
2	+12 VDC Camera Power (+12 VDC ± 10%) *
3	Not Connected
4	Not Connected
5	DC Ground **
6	DC Ground **

Table 2: Pin Assignments and Numbering for the 6-pin Receptacle

* Pins 1 and 2 are tied together inside of the camera.

** Pins 5 and 6 are tied together inside of the camera.

2.2.2 Camera Power Cable and Grounding

Camera power for camera operation is supplied to the camera via the camera power cable and the camera's 6-pin connector.

You can select from suitable cables, offered by Basler. See the Basler website for information about their characteristics. For voltage requirements of camera power, see Section 2.2.3 on page 24.

For proper EMI protection, the power cable must be firmly connected to the camera and to the power supply to ensure good contact of the cable shield. Follow the grounding recommendations given in Section 2.2.2.1 on page 23.

Basler strongly recommends to keep the camera power cable as short as possible and avoid close proximity to strong magnetic fields.The maximum camera power cable length must not exceed 10 m. In practice, however, the maximum allowed cable length for your application can be distinctly shorter due to your specific operating conditions. For further information and for advice about selecting the optimum cable, contact your sales representative.

2.2.2.1 Grounding Recommendations

For proper EMI protection, the cable shields and their electrical contacts must be installed as shown in Figure 8. The cable shields must be connected to the camera housing via the camera's connectors. The cable shield for the camera power cable must also be connected to earth ground at the power supply.

Close proximity to strong magnetic fields should be avoided.

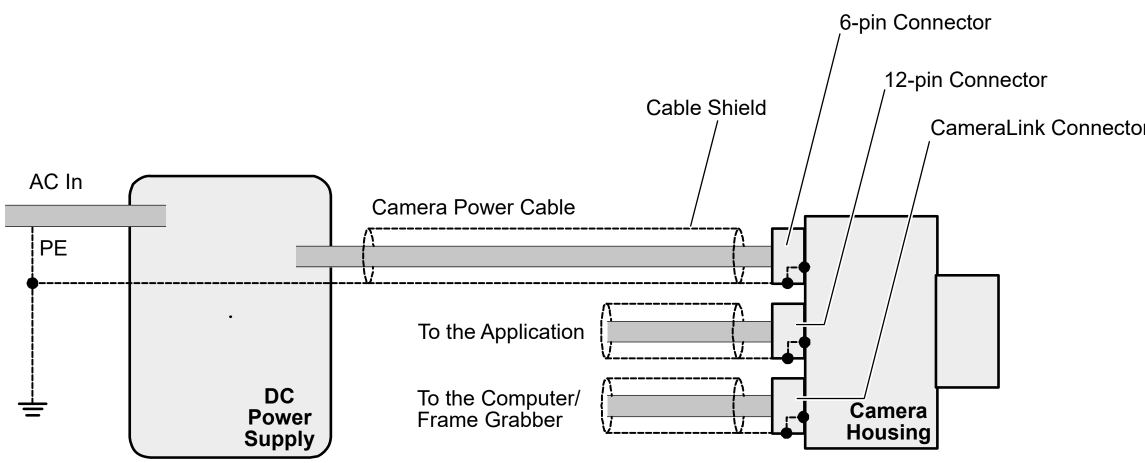


Fig. 8: Grounding Scheme for Camera, Cables, and Power Supply

NOTICE
To avoid a voltage drop when there are long wires between the power supply and the camera, Basler recommends that you provide +12 VDC through two separate wires between the power supply and pins 1 and 2 in the camera's 6-pin connector. Basler also recommends that you provide ground through two separate wires between the power supply and pins 5 and 6.

2.2.3 Camera Power

Camera power must be supplied to the 6-pin connector on the camera via a cable from your power supply. Nominal operating voltage is +12 VDC (± 10%) at the 6-pin connector with less than one percent ripple. Power consumption is as shown in the specification tables in Chapter 1 of this manual.

NOTICE

Applying incorrect power can damage the camera.

The camera's nominal operating voltage is +12 VDC (± 10 %). If the camera power voltage is greater than +13.2 VDC, damage to the camera can result. If the voltage is less than +10.8 VDC, the camera may operate erratically.

Applying power with the wrong polarity can result in severe damage to the camera.

1. Always make sure that the voltage of the camera power is within the specified range.
2. Always make sure that the polarity of the applied voltage is correct.

NOTICE

Making or breaking connections incorrectly can damage the camera.

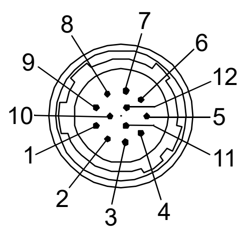
1. Be sure that all power to your camera and to your host computer is switched off before you make or break connections to the camera. Making or breaking connections when power is on can result in damage to the camera or to the frame grabber.
2. If you can't switch off the power, be sure that:
   a. The camera power plug is the last plug that you plug into the camera when making connections.
   b. The camera power plug is the first plug that you unplug from the camera when breaking connections.

2.3 I/O Interface and Requirements

2.3.1 12-Pin Receptacle

The 12-pin receptacle is used to access the camera's two physical input lines (In 1 and In 2) and one physical output line (Out 1) and provide them with power. The pin assignments and pin numbering for the receptacle are as shown in Table 3.

For information about voltage requirements, see Section 2.3.3.1 on page 27 and Section 2.3.4.1 on page 29.



Pin	Designation	Pin	Designation
1	Not connected	7	Not connected
2	Not connected	8	Not connected
3	In 1	9	Not connected
4	In 2	10	I/O VCC
5	I/O Ground	11	Not connected
6	Out 1	12	Not connected

Table 3: Pin Assignments and Numbering for the 12-pin Receptacle

2.3.2 I/O Cable

An I/O cable connects to the camera's 12-pin I/O connector and carries

- the I/O signals and
- power for the I/Os (not to be confused with camera power; see Section 2.2.3 on page 24).

Basler offers two types of the I/O cable:

- Power-I/O cable
- Power- I/O PLC+ cable.

The Power-I/O PLC+ cable is a variant of the Power-I/O cable. It is designed for use with devices where a logical zero is indicated by floating voltage or by undefined voltage above 1.4 VDC. This applies, for example, to PLC devices.

The Power-I/O PLC+ cable adjusts the logical zero voltages to steadier and correct voltages. The PLC I/O cable also improves the protection against negative voltage, reverse polarity, and EMI/ ESD.

See the Basler website for information about cable characteristics and how to obtain cables. In the downloads sections, you can find the related Technical Specifications.

Basler strongly recommends to keep the I/O cable as short as possible and avoid close proximity to strong magnetic fields.The maximum camera I/O cable length must not exceed 10 m. In practice, however, the maximum allowed cable length for your application can be distinctly shorter due to your specific operating conditions. For further information and for advice about selecting the optimum cable, contact your sales representative.

For information about voltage requirements for the

- cable input power to the Power-I/O cable, see Section 2.3.3.1 on page 27
- power applied to the camera's I/O pins, Section 2.3.3.1 on page 27
- significance of voltages of the I/O signals, see Section 2.3.3.1 on page 27 and Section 2.3.4.1 on page 29.

2.3.3 Input Lines

The camera is equipped with two physical input lines designated as In 1 (input line 1) and In 2 (input line 2). The input lines are accessed via the 12-pin receptacle on the back of the camera.

2.3.3.1 Voltage and Current Requirements

The following voltages apply **at the camera's I/O input pins** (pins 3 and 4 of the 12-pin receptacle, see Section 2.3.1 on page 25). The same voltages apply to cable input power for the Power-I/O cable.

Voltage	Significance
+0 to +24 VDC	Operating voltage.
+0 to +1.4 VDC	The voltage indicates a logical 0.
> +1.4 to +2.2 VDC	Region where the transition threshold occurs; the logical state is not defined in this region.
> +2.2 VDC	The voltage indicates a logical 1.

Table 4: Voltage Requirements for the I/O Input When Using the Standard I/O Cable

The current draw for each input line is between 5 and 15 mA.

For information about the choice and characteristics of connecting I/O cables, see Section 2.3.2 on page 26

2.3.3.2 Circuit Diagram

The camera is equipped with two physical input lines designated as In 1 (input line 1) and In 2 (input line 2). The input lines are accessed via the 12-pin receptacle on the back of the camera.

Each input line is opto-isolated (shown for line 1 as an example). See the previous section for input voltages and their significances. Figure 9 shows an example of a typical circuit you can use to input a signal into the camera.

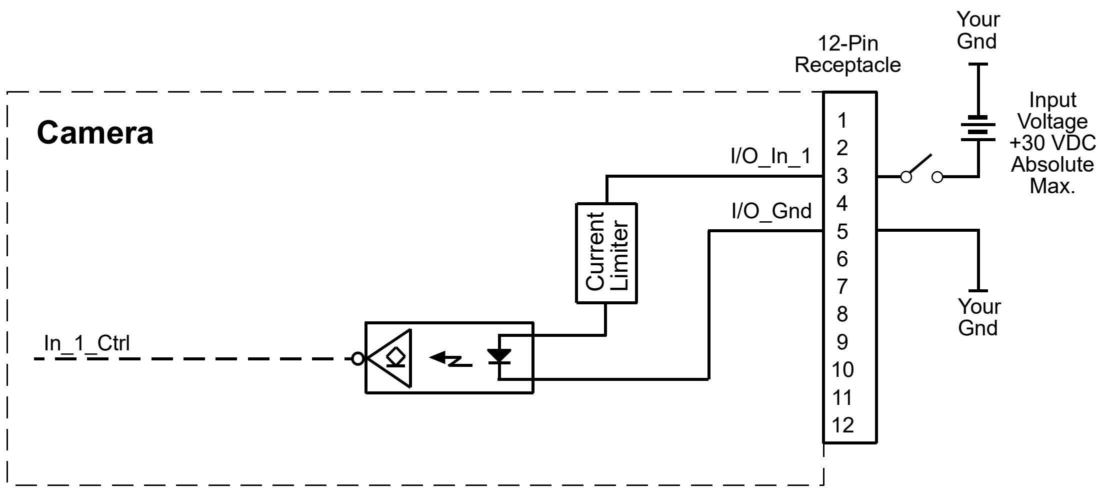


Fig. 9: Typical Input Line Circuit Diagram (Simplified)

For more information about

- input line pin assignments and pin numbering, see Section 2.3.1 on page 25.
- the delays between transition of a frame trigger and the begin of exposure ("exposure start delay"), see Table 8 on page 106.
- selecting the input lines, see Section 2.3.3.3 on page 29.

2.3.3.3 Selecting an Input Line as the Source Signal for a Camera Function

You can select input line 1 or input line 2 to act as the source signal for the following camera functions:

- the acquisition start trigger
- the frame start trigger

Note that when an input line has been selected as the source signal for a camera function, you must apply an electrical signal to the input line that is appropriately timed for the function.

For more information about selecting input line 1 or input line 2 as the source signal

- for a hardware **acquisition start** trigger function, see Section 6.2 on page 69.
- for a hardware **frame start** trigger function, see Section 6.3 on page 77.

For information about the time interval between the moment when the frame start trigger signal transitions and the moment when exposure actually begins (exposure start delay), see Section 6.7 on page 106

- The **exposure start delay** is the time interval between the moment when the frame start trigger signal transitions and the moment when exposure actually begins (see).

2.3.4 Output Line

The camera is equipped with one physical output line designated as Out 1 (output line 1). The output line is accessed via the 12-pin receptacle on the back of the camera (see Section 2.3.1 on page 25).

2.3.4.1 Voltage and Current Requirements

The following voltage requirements apply to the I/O output VCC (pin 10 of the 12-pin receptacle, see Section 2.3.1 on page 25):

Voltage	Significance
< +3.3 VDC	The I/O output may operate erratically.
+3.3 to +24 VDC	Recommended operating voltage.

Table 5: Voltage Requirements for the I/O Output VCC

The maximum current allowed through an output circuit is 50 mA.

2.3.4.2 Circuit Diagram

The camera is equipped with one physical output line designated as Out 1 (output line 1). The output line is accessed via the 12-pin receptacle on the back of the camera.

As shown in the I/O circuit diagram, the output line is opto-isolated. See the previous section for the recommended operating voltage.

A conducting transistor means a logical one and a non-conducting transistor means a logical zero.

Figure 10 shows a typical circuit you can use to monitor the output line with a voltage signal.

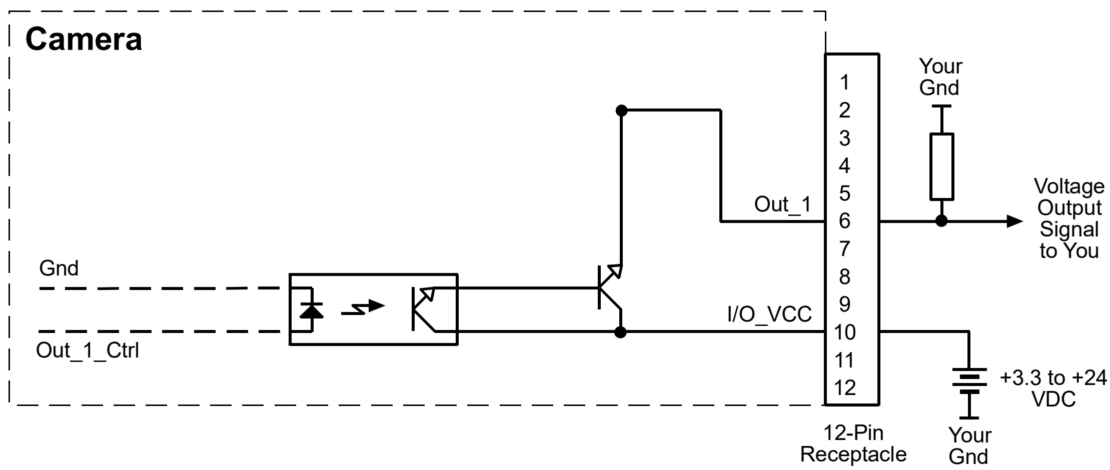


Fig. 10: Typical Voltage Output Circuit (Simplified)

Figure 11 shows a typical circuit you can use to monitor the output line with an LED or an opto-coupler. In this example, the voltage for the external circuit is +24 VDC. Current in the circuit is limited by an external resistor.

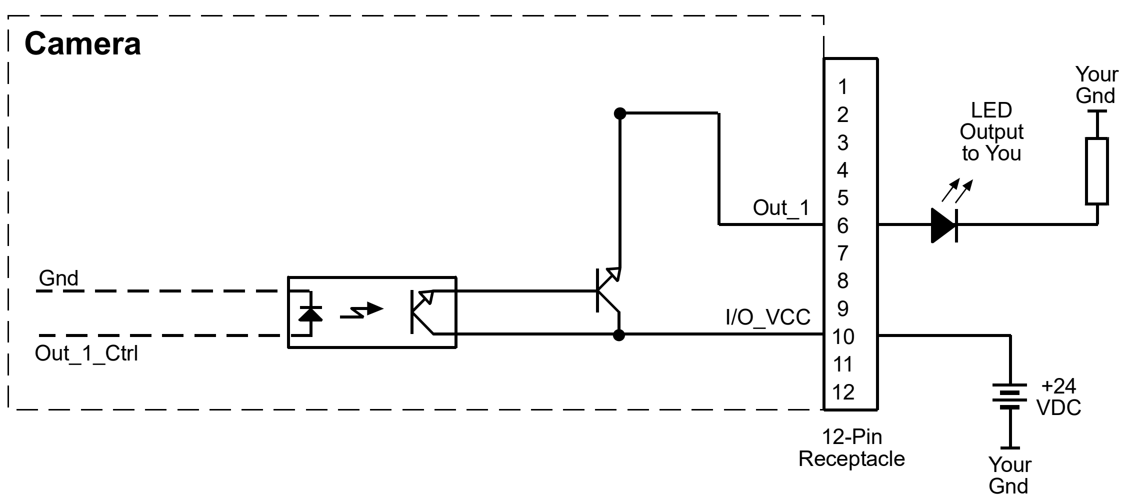


Fig. 11: Typical LED Output Signal (Simplified)

For more information about

- output line pin assignments and pin numbering, see Section 2.3.1 on page 25.
- selecting a source signal for the output line, see Section 2.3.4.4 on page 32.

2.3.4.3 Response Times

Response times for the output line on the camera are as shown in Figure 12.

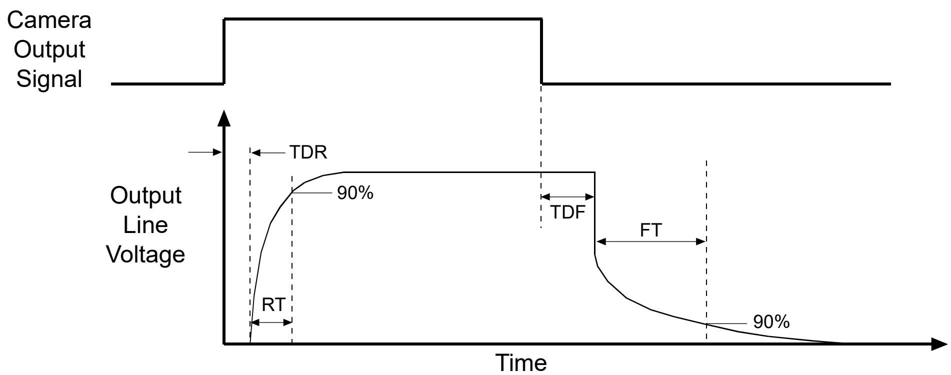


Fig. 12: Output Line Response Times

Time Delay Rise (TDR) = 1.5 µs (typical)

Rise Time (RT) = 1.0 - 2.0 µs (typical)

Time Delay Fall (TDF) = 40 µs (typical)

Fall Time (FT) = 5 - 10 µs (typical)

i	The response times for the output line on your camera typically fall into the ranges specified above. The exact response time for your specific application depends on the external resistor and the applied voltage you use.

2.3.4.4 Selecting a Source Signal for the Output Line

To make the physical output line useful, you must select a source signal for the line. The camera has several standard output signals available and any one of them can be selected to act as the source signal for the output line.

- Acquisition Trigger Wait
- Frame Trigger Wait
- Exposure Active
- Timer Active
- User Output
- Sync User Output

For more information about selecting a source signal for the output line, see Section 3.2 on page 43.

2.4 Camera Link Interface

The Camera Link interface on the aviator is accessed via the 26-pin MDR connector. The Camera Link interface is designed to be completely compatible with the base Camera Link standard.

The camera has a default Camera Link pixel clock speed of 65 MHz. The pixel clock speed can also be changed to 20 MHz, 32.5 MHz, 40 MHz, or 48 MHz.

The camera can output pixel data in 8 bit, 10 bit, or 12 bit pixel formats and in the 1X-1Y, 1X2-1Y or 1X-2YE Camera Link tap geometries.

For more information about changing the Camera Link pixel clock speed, see Section 9.2 on page 149.

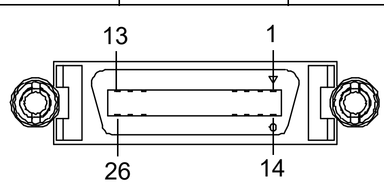
If you plan to design your own frame grabber, or if you require specific details regarding the way that the Camera LInk interface is implemented on the camera, refer to the document called *Aviator Information for Frame Grabber Designers* (AW000831). You can download the document from the Basler website: www.baslerweb.com.

2.4.1 26-Pin MDR Connector

The 26-pin, 0.050” Mini D Ribbon (MDR) female connector is used to transmit video data, control signals, and configuration commands. The pin assignments and pin numbering for the MDR connector are as shown in Table 6.

Suitable Camera Link cables are available from Basler. For more information, see the Basler website.

Pin Number	Signal Name	Direction	Level	Function
1, 13, 14, 26[1]	Gnd	Input	Ground	Ground for the inner shield of the cable
2	X0-	Output	Camera Link LVDS	Data from transmitter circuit X
15	X0+			
3	X1-	Output	Camera Link LVDS	Data from transmitter circuit X
16	X1+			
4	X2-	Output	Camera Link LVDS	Data from transmitter circuit X
17	X2+			
6	X3-	Output	Camera Link LVDS	Data from transmitter circuit X
19	X3+			
5	XClk-	Output	Camera Link LVDS	Pixel clock from transmitter circuit X
18	XClk+			
7	SerTC+	Input	RS-644 LVDS	Serial communication data receive (SerTC = "serial to camera")
20	SerTC-			
8	SerTFG-	Output	RS-644 LVDS	Serial communication data transmit (SerTFG = "serial to frame grabber")
21	SerTFG+			
9	CC1-	Input	RS-644 LVDS	ExTrig (external trigger)
22	CC1+			
10	CC2+	Input	RS-644 LVDS	Not assigned
23	CC2-			
11	None	None	None	None
24	None			
12	CC4+	Input	RS-644 LVDS	Not assigned
25	CC4-			



[1] Pins 1, 13, 14, and 26 are all tied to ground inside of the camera.

Table 6: Pin Assignments and Numbering for the 26-pin MDR Connector

2.4.2 Acquisition Start and Frame Start Trigger Signals

As specified in the Camera Link standard, several camera control channels are built into the Camera Link interface.These control channels are designated as CC1, CC2, and CC4.

On aviator cameras, the CC1 (camera control 1) channel is assigned by default to be the source signal for a hardware frame start trigger signal. Camera Link compatible frame grabbers are typically designed to supply a frame start trigger signal to the camera via CC1 in the Camera Link interface.

The camera can also be set so that one of these control channels is assigned to be the source signal for a hardware acquisition start trigger signal.

For more information about setting and using acquisition start and frame start trigger signals, see Section 6.2 on page 69 and Section 6.3 on page 77.

2.4.3 Camera Link Serial Port Baud Rate

As specified in the Camera Link standard, an RS-644 serial connection must be included on all Camera Link compliant frame grabbers. Your computer and your camera are able to communicate via this serial port built into the frame grabber. If you are using the Basler pylon Camera Software Suite to operate the camera, the serial port is used to communicate pylon commands and replies to and from the camera. If you are operating the camera via direct register access, the serial port is used to communicate read and write requests to and from the camera.

The serial port can operate at the following baud rates: 9600, 19200, 38400, 57600, 115200, 230400, 460800, and 921600.

You can change the baud rate for the serial port by setting the Camera Link Serial Port Baud Rate parameter.

i	The camera will return to the 9600 baud setting when it is reset or powered off.

Setting the Baud Rate Using Basler pylon

You can use the pylon API to set the Camera Link Serial Port Baud Rate parameter value from within your application software. The following code snippet illustrates using the pylon API to set the parameter value:

```
// Set the serial port baud rate to 115200
Camera.ClSerialPortBaudRate.SetValue( ClSerialPortBaudRate_Baud115200 );
```

You can also use the Basler pylon Viewer application to easily set the parameter.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Baud Rate Using Direct Register Access

To set the Camera Link serial port baud rate via direct register access:

- Set the value of the CL Serial Port Baud Rate register to the desired baud rate.

For more information about changing settings via direct register access, see Section 5 on page 61.

2.4.4 Inter-Line Delay

The inter-line delay parameter determines the delay between the end of transmission of a line and the start of transmission of the next line (within a frame).

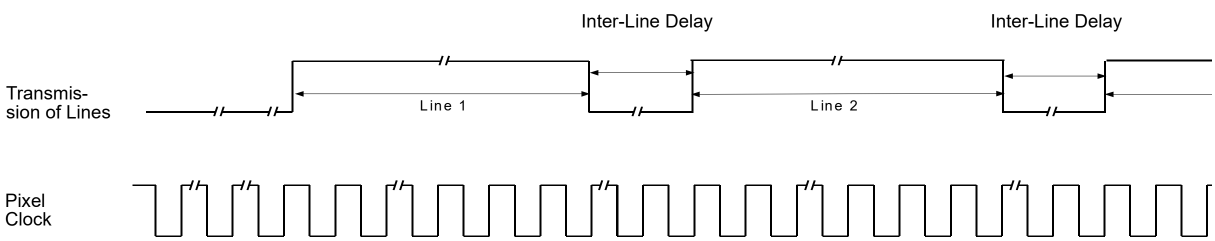


Fig. 13: Inter-Line Delay Parameter (Valid for Lines Within a Frame)

The Line Valid Low (Raw) parameter sets an integer value to express the delay as a number of pixel clock cycles. The line valid signal (LVAL) remains low during the set number of pixel clock cycles. The time interval of the delay is the product of the set value of the inter-line delay parameter and the period of the currently selected pixel clock.

Range of values of the Line Valid Low (Raw): 4 - 513
The default value of the parameter is the minimum allowed number of inter-line clock cycles.

The Line Valid Low (Abs) parameter sets a floating point value to express the delay in microseconds. The set value will be automatically adjusted to the nearest value that can be expressed as a multiple of the period of the currently selected pixel clock.

Note

Increasing the inter-line delay parameters results in a lower acquisition frame rate.

Setting the Inter-Line Delay Using Basler pylon

You can use the pylon API to set the inter-line delay from within your application software. The following code snippet illustrates using the pylon API to set the inter-line delay:

```
// Set the inter-line delay
Camera.ClInterLineDelayRaw.SetValue( 4 );
```

Setting the Inter-Line Delay Using Direct Register Access

To set the inter-line delay parameter via direct register access:

- Set the value of the CL Interline Delay register.

For more information about changing settings via direct register access, see Section 5 on page 61.

3 I/O Control

This section describes how to configure the camera's five input lines and one output line. It also provides information about monitoring the state of the input and output lines.

For more detailed information about the physical and electrical characteristics of the input and output lines, see Section 2.3.3 on page 27 and Section 2.3.4 on page 29.

3.1 Configuring Input Lines

3.1.1 Assigning an Input Line to Receive a Hardware Trigger Signal

You can assign the camera's input lines to receive external hardware trigger (ExTrig) signals. The incoming ExTrig signals can then be used to control image acquisition.

Section 6.2.5.2 on page 75 and Section 6.3.3.4 on page 90 explain how to configure the camera to react to a hardware trigger signal and how to assign an input line to receive the hardware trigger signal.

3.1.2 Input Line Debouncer

The Line Debouncer feature allows you to filter invalid hardware input signals. Only valid signals are allowed to pass through to the camera and become effective.

Prerequisites

The camera must be configured for hardware triggering.

How It Works

The line debouncer filters out unwanted short signals (contact bounce) from the rising and falling edges of incoming hardware trigger signals. The line debouncer employs a clock and evaluates all changes and durations of logical states of hardware signals to distinguish between valid and invalid signals.

The maximum duration of the evaluation period (the "line debouncer time") is defined by the LineDebouncerTimeAbs parameter.

The clock starts counting whenever a hardware signal changes its logical state (high to low or vice versa). If the duration of the new logical state is shorter than the line debouncer time specified, the new logical state is considered invalid and has no effect. If the duration of the new logical state is as long as the line debouncer time or longer, the new logical state is considered valid and is allowed to become effective in the camera.

Default value: 10 µs

i	Specifying a line debouncer time introduces a delay between a valid trigger signal arriving at the camera and the moment the related change of logical state is passed on to the camera. The duration of the delay is at least equal to the value of the LineDebouncerTimeAbs parameter. This is because the camera waits for the time specified as the line debouncer time to determine whether the signal is valid.

Figure 14 illustrates how the line debouncer filters out invalid signals from the rising and falling edge of a hardware trigger signal. Line debouncer times that actually allow a change of logical state in the camera are labeled "OK". Also illustrated are the delays of logical states inside the camera relative to the hardware trigger signal.

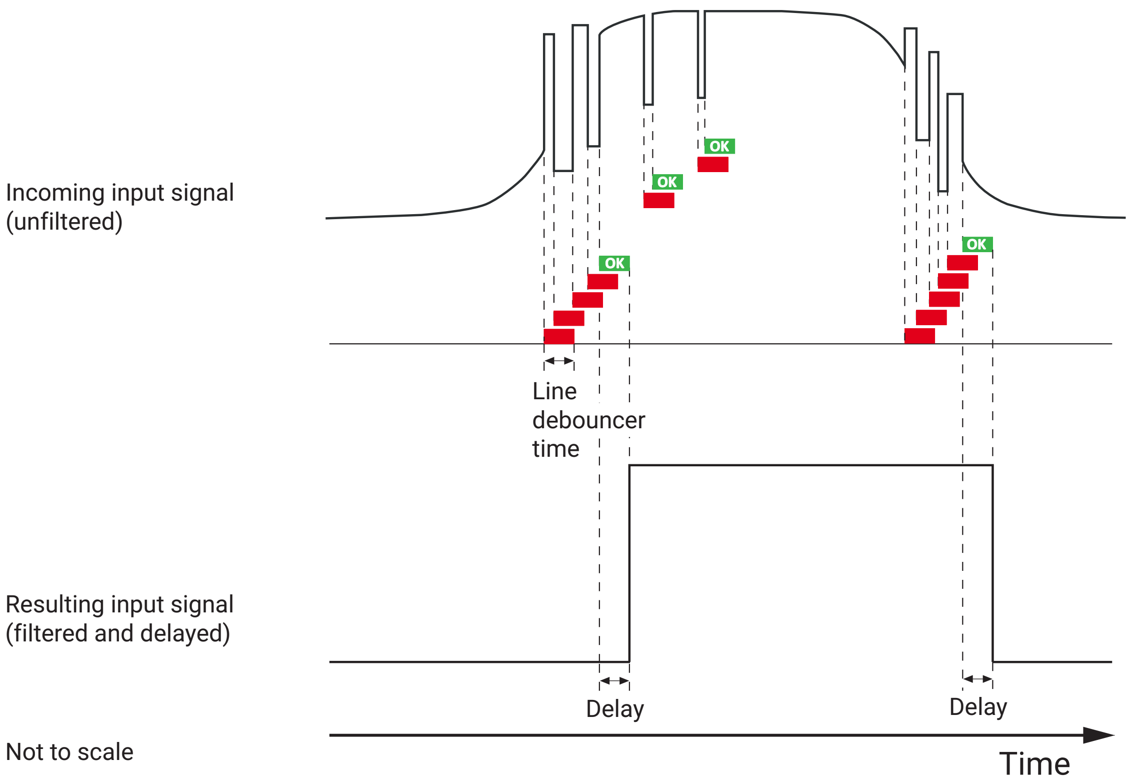


Fig. 14: Filtering of Invalid Hardware Signals by the Debouncer

Setting the Input Debouncer Using Basler pylon

The debouncer value is determined by the value of the LineDebouncerTimeAbs parameter value. The parameter is set in microseconds and can be set in a range from 0 to approximately 20 ms.

To set a debouncer:

- Use the Line Selector to select Line 1, Line 2, CC1, CC2, or CC4
- Set the value of the LineDebouncerTime Abs parameter.

You can set the Line Selector and the value of the LineDebouncerTimeAbs parameter from within your application software by using the pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
// Select the input line
Camera.LineSelector.SetValue( LineSelector_Line1 );

// Set the parameter value to 100 microseconds
Camera.LineDebouncerTimeAbs.SetValue( 100 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon Viewer, see Section 5.2 on page 62.

Setting the Input Debouncer Using Direct Register Access

To set the value of the input line debouncers via direct register access:

- For Line 1, set the value of the Input Debouncer Time Line1 register as desired (the value represents milliseconds).
- For Line 2, set the value of the Input Debouncer Time Line2 register.
- For the CC1 line, set the value of the Input Debouncer Time CC1 register.
- For the CC2 line, set the value of the Input Debouncer Time CC2 register.
- For the CC4 line, set the value of the Input Debouncer Time CC4 register.

For more information about direct register access, see Section 5 on page 61.

3.1.3 Using an Unassigned Input Line to Receive a User Input Signal

You can use an unassigned input line to receive your own, user-generated input signal. The electrical characteristics of your input signal must meet the requirements shown in the Physical Interface section of this manual.

You can use the Line Status or Line Status All parameters to monitor the state of the input line that is receiving the user-defined signal.

i	A line assigned to receive an ExTrig input signal can't be used to receive a user-designed input signal.

For more information about using the Line Status and Line Status All parameters, see Section 3.3.1 on page 52 and Section 3.3.2 on page 53.

3.1.4 Setting an Input Line for Invert

Setting an Input Line for Invert Using Basler pylon

You can set each individual input line to invert or not to invert the incoming electrical signal. To set the invert function on an input line:

- Use the Line Selector to select CC1, CC2, CC4, Line 1, or Line 2.
- Set the value of the Line Inverter parameter to true to enable inversion on the selected line and to false to disable inversion.

You can set the Line Selector and the Line Inverter parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
// Enable the inverter on line 1
Camera.LineSelector.SetValue( LineSelector_Line1 );
Camera.LineInverter.SetValue( true );
```

For detailed information about using the pylon API, refer to the Basler pylon Programmer's Guide and API Reference.

You can also use the Basler pylon Viewer application to easily set the parameters.

Setting an Input Line for Invert Using Direct Register Access

To set the invert function on an input line via direct register access:

- For Line 1, set the value of the Line Inverter Line1 register to 0 (false) or 1 (true) as desired.
- For Line 2, set the value of the Line Inverter Line2 register.
- For the CC1 line, set the value of the Line Inverter CC1 register.
- For the CC2 line, set the value of the Line Inverter CC2 register.
- For the CC4 line, set the value of the Line Inverter CC4 register.

For more information about direct register access, see Section 5 on page 61.

3.2 Configuring the Output Line

3.2.1 Assigning a Camera Output Signal to a Physical Output Line

The camera is equipped with one physical output line designated as output line 1. You can select any one of the camera's standard output signals to act as the source signal for output line 1.The camera has a variety of standard output signals available including:

- Acquisition Trigger Wait
- Frame Trigger Wait
- Exposure Active
- Timer Active
- User Output
- Sync User Output

You can also designate the output line as "user settable". If the output line is designated as user settable, you can use the camera's API to set the state of the line as desired.

Assigning a Camera Output Signal to a Physical Output Line Using Basler pylon

To assign an output signal to the output line or to designate the line as user settable:

- Use the Line Selector to select output line 1.
- Set the value of the Line Source Parameter to one of the available output signals or to user settable. This will set the source signal for the selected line

By default, the Exposure Active signal is selected as the source signal for Output Line 1.

You can set the Line Selector and the Line Source parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
Camera.LineSelector.SetValue( LineSelector_Out1 );
Camera.LineSource.SetValue( LineSource_ExposureActive );
```

For detailed information about using the pylon API, refer to the Basler pylon Programmer's Guide and API Reference.

You can also use the Basler pylon Viewer application to easily set the parameters.

Assigning a Camera Output Signal to a Physical Output Line Using Direct Register Access

To assign an output signal to an output line via direct register access:

Set the value of the Line Source Line 1 register to Acquisition Trigger Wait, Frame Trigger Wait, Exposure Active, Timer Active, or User as desired.

For more information about

- the pylon Viewer, see Section 5 on page 61.
- setting the state of user settable output signals, see Section 3.2.2 on page 44.
- working with the timer output signals, see Section 3.2.4 on page 46.
- the exposure active signal, see Section 6.6.1 on page 97.

3.2.2 Setting the State of a User Settable Output Line

As mentioned in the previous section, you can designate the camera's output line as "user settable". If you have designated the output line as user settable, you can use camera parameters to set the state of the line.

Setting the State of a User Settable Output Line Using Basler pylon

To set the state of a user settable output line:

- Use the User Output Selector to select output line 1.
- Set the value of the User Output Value parameter to true (1) or false (0). This will set the state of the output line.

You can set the Output Selector and the User Output Value parameter from within your application software by using the pylon API. The following code snippet illustrates using the API to designate the output line as user settable and setting the state of the output line:

```
// Set output line 1 to user settable
Camera.LineSelector.SetValue( LineSelector_Out1 );
Camera.LineSource.SetValue( LineSource_UserOutput );
// Set the state of output line 1
Camera.UserOutputSelector.SetValue( UserOutputSelector_UserOutput1 );
Camera.UserOutputValue.SetValue( true );
bool currentUserOutput1State = Camera.UserOutputValue.GetValue( );
```

For detailed information about using the pylon API, refer to the Basler pylon Programmer's Guide and API Reference.

You can also use the Basler pylon Viewer application to easily set the parameters.

Setting the State Using Direct Register Access

To set the state of a user settable output line via direct register access:

For output line 1, set the value of the Line 5 register to 1 (true) or 0 (false) as desired.

3.2.3 Setting the Output Line for Invert

Setting an Output Line for Invert Using pylon

You can set the output line to invert or not to invert the outgoing signal. To set the invert function on an output line:

- Use the Line Selector to select output line 1.
- Set the value of the Line Inverter parameter to true to enable inversion on output line 1 and to false to disable inversion.

You can set the Line Selector and the Line Inverter parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
// Enable the inverter on output line 1
Camera.LineSelector.SetValue( LineSelector_Out1 );
Camera.LineInverter.SetValue( true );
```

For detailed information about using the pylon API, refer to the Basler pylon Programmer's Guide and API Reference.

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon Viewer, see Section 5 on page 61.

Setting an Output Line for Invert Using Direct Register Access

To set the invert function on an output line via direct register access:

For output line 1, set the value of the Line Inverter Line 5 register to 0 (false) or 1 (true) as desired.

3.2.4 Working with Timers

The camera has two timer output signals available: Timer 1 and Timer 2. As shown in Figure 15, each timer works as follows:

- A trigger source event occurs that starts the timer.
- A delay period begins to expire.
- When the delay expires, the timer signal goes high and a duration period begins to expire.
- When the duration period expires, the timer signal goes low.

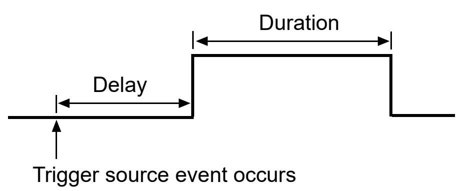


Fig. 15: Timer Signal

Currently, the only trigger source event available to start the timer is "exposure active". In other words, you can use exposure start to trigger the start of a timer.

Timer 1 can only be assigned to output line 1.

If you require the timer signal to be high when the timer is triggered and to go low when the delay expires, simply set the output line to invert.

3.2.4.1 Setting the Timer 1 Parameters

Setting the Timer 1 Parameters Using Basler pylon

When Basler pylon is used, setting Timer 1 is a three step process:

- Use the Timer Selector parameter to select Timer 1.
- Set the value of the Timer Delay Abs parameter.
- Set the value of the Timer Duration Abs parameter.

The Timer Delay Abs parameter sets the timer delay in microseconds, and the Timer Duration Abs parameter sets the timer duration in microseconds. The delay and the duration should be set in increments of 1 microsecond.

You can use pylon API to set the Timer Selector, the Timer Delay Abs, and the Timer Duration Abs parameters from within your application software. The following code snippet illustrates using the pylon API to set the parameters:

```
Camera.TimerSelector.SetValue( TimerSelector_Timer1 );
Camera.TimerDelayAbs.SetValue( 100 );
Camera.TimerDurationAbs.SetValue( 200 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Timer 1 Parameters Using Direct Register Access

To set the timer 1 parameters via direct register access:

- Set the value of the Timer Delay Raw Timer 1 register.
- Set the value of the Timer Duration Raw Timer 1 register.

A value in a raw register is simply an integer value with no units. To determine what the actual delay or duration time will be, you must multiply the raw value by the camera's time base. The time base on aviator cameras is 1 µs.

For example, if you set the Timer Delay Raw Timer 1 register to 100, the delay time for timer 1 would be 100 µs (100 x 1 µs = 100 µs).

For more information about changing settings via direct register access, see Section 5 on page 61.

3.2.4.2 Setting the Trigger Source for a Timer

To set the trigger source for a timer:

- Use the Timer Selector to select timer 1 or timer 2.
- Set the value of the Timer Trigger Source parameter to exposure active. This will set the selected timer to use the start of exposure to begin the timer.

You can set the Trigger Selector and the Timer Trigger Source parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
Camera.TimerSelector.SetValue( TimerSelector_Timer1 );
Camera.TimerTriggerSource.SetValue( TimerTriggerSource_ExposureStart );
```

For detailed information about using the pylon API, refer to the Basler pylon Programmer's Guide and API Reference.

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon Viewer, see Section 5 on page 61.

3.2.4.3 Setting a Timer Delay Time

There are two ways to set the delay time for a timer: by setting "raw" values or by setting an "absolute value". You can use whichever method you prefer to set the delay time.

Setting the Delay with Raw Values

When the delay time for a timer is set using "raw" values, the delay time will be determined by a combination of two elements. The first element is the value of the Timer Delay Raw parameter, and the second element is the Timer Delay Time Base. The delay time is the product of these two elements:

Delay Time = (Timer Delay Raw Parameter Value) x (Timer Delay Time Base)

By default, the Timer Delay Time Base is fixed at 1 µs. Typically, the delay time is adjusted by setting the Timer Delay Raw parameter value.

The Timer Delay Raw parameter value can range from 0 to 4095. So if the value is set to 100, for example, the timer delay will be 100 x 1 µs or 100 µs.

To set the delay for a timer:

- Use the Timer Selector to select a timer.
- Set the value of the Timer Delay Raw parameter.

You can set the Timer Selector and the Timer Delay Raw parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
Camera.TimerSelector.SetValue( TimerSelector_Timer1 );
Camera.TimerDelayRaw.SetValue( 100 );
```

For detailed information about using the pylon API, refer to the Basler pylon Programmer’s Guide and API Reference.

You can also use the Basler pylon Viewer application to easily set the parameters.

Changing the Delay Time Base

By default, the Timer Delay Time Base is fixed at 1 µs (minimum value), and the timer delay is normally adjusted by setting the value of the Timer Delay Raw parameter. However, if you require a delay time that is longer than what you can achieve by changing the value of the Timer Delay Raw parameter alone, the Timer Delay Time Base Abs parameter can be used to change the delay time base.

The Timer Delay Time Base Abs parameter value sets the delay time base in µs. The default is 1 µs and it can be changed in 1 µs increments.

Note that there is only one timer delay time base and it is used by all four of the available timers.

You can set the Timer Delay Time Base Abs parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to set the parameter value:

```
Camera.TimerDelayTimebaseAbs.SetValue( 5 );
```

Setting the Delay with an Absolute Value

You can also set the Timer delay by using an "absolute" value. This is accomplished by setting the Timer Delay Abs parameter. The units for setting this parameter are µs and the value can be set in increments of 1 µs.

To set the delay for a timer using an absolute value:

- Use the Timer Selector to select a timer.
- Set the value of the Timer Delay Abs parameter.

You can set the Timer Selector and the Timer Delay Abs parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
Camera.TimerSelector.SetValue( TimerSelector_Timer1 );
Camera.TimerDelayAbs.SetValue( 100 );
```

When you use the Timer Delay Abs parameter to set the delay time, the camera accomplishes the setting change by automatically changing the Timer Delay Raw parameter to achieve the value specified by the Timer Delay Abs setting. This leads to a limitation that you must keep in mind if you use Timer Delay Abs parameter to set the delay time. That is, you must set the Timer Delay Abs parameter to a value that is equivalent to a setting you could achieve by using the Timer Delay Raw and the current Timer Delay Base parameters. For example, if the time base was currently set to 50 µs, you could use the Timer Delay Abs parameter to set the delay to 50 µs, 100 µs, 150 µs, etc.

Note that if you set the Timer Delay Abs parameter to a value that you could not achieve by using the Timer Delay Raw and current Timer Delay Time Base parameters, the camera will automatically change the setting for the Timer Delay Abs parameter to the nearest achieveable value.

You should also be aware that if you change the delay time using the raw settings, the Timer Delay Abs parameter will automatically be updated to reflect the new delay time.

3.2.4.4 Setting a Timer Duration Time

There are two ways to set the duration time for a timer: by setting "raw" values or by setting an "absolute value". You can use whichever method you prefer to set the duration time.

Setting the Duration with Raw Values

When the duration time for a timer is set using "raw" values, the duration time will be determined by a combination of two elements. The first element is the value of the Timer Duration Raw parameter, and the second element is the Timer Duration Time Base. The duration time is the product of these two elements:

Duration Time = (Timer Duration Raw Parameter Value) x (Timer Duration Time Base)

By default, the Timer Duration Time Base is fixed at 1 µs. Typically, the duration time is adjusted by setting only the Timer Duration Raw parameter value.

The Timer Duration Raw parameter value can range from 1 to 4095. So if the value is set to 100, for example, the timer duration will be 100 x 1 µs or 100 µs.

To set the duration for a timer:

- Use the Timer Selector to select a timer.
- Set the value of the Timer Duration Raw parameter.

You can set the Timer Selector and the Timer Duration Raw parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
Camera.TimerSelector.SetValue( TimerSelector_Timer1 );
Camera.TimerDurationRaw.SetValue( 100 );
```

For detailed information about using the pylon API, refer to the Basler pylon Programmer's Guide and API Reference.

You can also use the Basler pylon Viewer application to easily set the parameters.

Changing the Duration Time Base

By default, the Timer Duration Time Base is fixed at 1 µs, and the timer duration is normally adjusted by setting the value of the Timer Duration Raw parameter. However, if you require a duration time that is longer than what you can achieve by changing the value of the Timer Duration Raw parameter alone, the Timer Duration Time Base Abs parameter can be used to change the duration time base.

The Timer Duration Time Base Abs parameter value sets the duration time base in µs. The default is 1 µs and it can be changed in 1 µs increments.

Note that there is only one timer duration time base and it is used by all four of the available timers.

You can set the Timer Duration Time Base Abs parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to set the parameter value:

```
Camera.TimerDurationTimebaseAbs.SetValue( 5 );
```

Setting the Duration with an Absolute Value

You can also set the Timer duration by using an "absolute" value. This is accomplished by setting the Timer Duration Abs parameter. The units for setting this parameter are µs and the value can be set in increments of 1 µs.

To set the duration for a timer using an absolute value:

- Use the Timer Selector to select a timer.
- Set the value of the Timer Duration Abs parameter.

You can set the Timer Selector and the Timer Duration Abs parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
Camera.TimerSelector.SetValue( TimerSelector_Timer1 );
Camera.TimerDurationAbs.SetValue( 100 );
```

When you use the Timer Duration Abs parameter to set the duration time, the camera accomplishes the setting change by automatically changing the Timer Duration Raw parameter to achieve the value specified by the Timer Duration Abs setting. This leads to a limitation that you must keep in mind if you use Timer Duration Abs parameter to set the duration time. That is, you must set the Timer Duration Abs parameter to a value that is equivalent to a setting you could achieve by using the Timer Duration Raw and the current Timer Duration Base parameters. For example, if the time base was currently set to 50 µs, you could use the Timer Duration Abs parameter to set the duration to 50 µs, 100 µs, 150 µs, etc.

If you read the current value of the Timer Duration Abs parameter, the value will indicate the product of the Timer Duration Raw parameter and the Timer Duration Time Base. In other words, the Timer Duration Abs parameter will indicate the current duration time setting.

You should also be aware that if you change the duration time using the raw settings, the Timer Duration Abs parameter will automatically be updated to reflect the new duration time.

3.3 Checking the State of the I/O Lines

3.3.1 Checking the State of a Single Line

Checking the State Using Basler pylon

You can determine the current state of any one of the camera's I/O lines using Basler pylon:

- Use the Line Selector parameter to select a line.
- Read the value of the Line Status parameter to determine the current state of the selected line. A value of true means the line's state is currently high and a value of false means the line's state is currently low.

You can set the Line Selector and read the Line Status parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to set the selector and read the parameter value:

```
// Select output line 2 and read the state
Camera.LineSelector.SetValue( LineSelector_Out2 );
bool outputLine2State = Camera.LineStatus.GetValue( );
```

For detailed information about using the pylon API, refer to the Basler pylon Programmer's Guide and API Reference.

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon Viewer, see Section 5 on page 61.

Checking the State Using Direct Register Access

To check the current state of an I/O line via direct register access:

- For Line 1, read the value of the Line Status Line 1 register. The value will indicate 1 (true) or 0 (false).
- For Line 2, read the value of the Line Status Line 2 register.
- For the CC1 line, read the value of the Line Status CC1 register.
- For the CC2 line, read the value of the Line Status CC2 register.
- For the CC4 line, read the value of the Line Status CC4 register.

For more information about direct register access, see Section 5 on page 61.

3.3.2 Checking the State of All Lines

You can determine the current state of all input and output lines with a single operation. To check the state of all lines:

Read the value of the Line Status All parameter.

You can read the Line Status All parameter value from within your application software by using the pylon API. The following code snippet illustrates using the API to read the parameter value:

```
int64_t lineState = Camera.LineStatusAll.GetValue( );
```

The Line Status All parameter is a 32 bit value. As shown in Figure 16, certain bits in the value are associated with each line, and the bits will indicate the state of the lines. If a bit is 0, it indicates that the state of the associated line is currently low. If a bit is 1, it indicates that the state of the associated line is currently high.

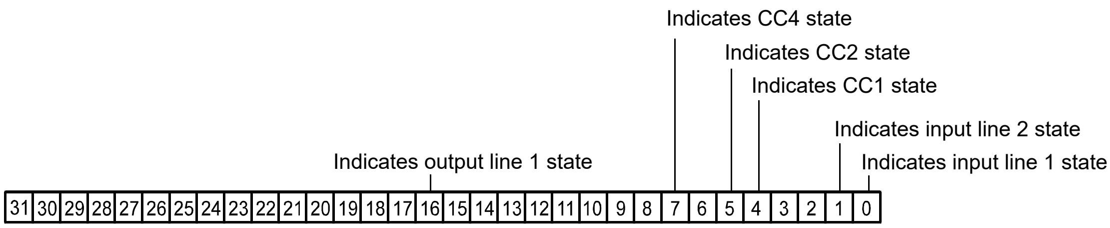


Fig. 16: Line Status All Parameter Bits

Checking the State Using Direct Register Access

To check the current state of all I/O lines via direct register access, read the value of the Line Status All register. The register holds a 32 bit value that indicates the state of of each I/O line. The mapping of the bits in the value to I/O lines is similar to the mapping described above for access via Basler pylon.

For more information about direct register access, see Section 5 on page 61.

4 Camera Functional Description

This chapter provides an overview of the camera's functionality from a system perspective. The overview will aid your understanding when you read the more detailed information included in the later chapters of the user's manual.

4.1 Overview

Each camera provides features such as a full frame shutter and electronic exposure time control. Exposure start, exposure time, and charge readout can be controlled by parameters transmitted to the camera via the Camera Link interface.

Exposure start can also be controlled via an externally generated electrical trigger (ExFrameStTrig) signal. The ExFrameStTrig signal facilitates periodic or non-periodic acquisition start. Modes are available that allow the length of exposure time to be directly controlled by the ExFrameStTrig signal or to be set for a pre-programmed period of time.

Accumulated charges are read out of the sensor when exposure ends. The sensor can be read out

- in a **four tap** fashion or
- in a **one tap** fashion.

4.1.1 Four Tap Sensor Digitization Mode

With four tap sensor digitization, the **sensor is divided into quadrants**, and a **separate electronic** circuit is used to read out the pixels in each quadrant (see Figure 17 on page 56). **Each of the electronic circuits** used to read out a quadrant of the sensor is referred to as **a tap**. The advantage of the four tap digitization scheme is that it makes readout very fast because the four circuits are used simultaneously to read out the sensor.

After an image has been captured (i.e., exposure has ended), the pixels in the sensor become ready to be read out. At readout, the accumulated charges are transported from the pixels to the sensor's vertical shift registers. The charges from the top line of pixels in the array are then moved to the upper horizontal shift register and the charges from the bottom line of pixels are moved to the lower horizontal shift register as shown in Figure 17. Once this has been accomplished, the following operations are performed simultaneously:

- Charges from the left half of the top line are moved out of the upper horizontal shift register. The left half of the upper horizontal shift register shifts out charges from left to right, that is, pixel 1, pixel 2, pixel 3, and so on.
- Charges from the right half of the top line are moved out of the upper horizontal shift register. The right half of the upper horizontal shift register shifts out charges from right to left, that is, pixel n, pixel n-1, pixel n-2, and so on (where n is the last pixel in a line).
- Charges from the left half of the bottom line are moved out of the lower horizontal shift register. The left half of the lower horizontal shift register shifts out charges from left to right, that is, pixel 1, pixel 2, pixel 3, and so on.
- Charges from the right half of the bottom line are moved out of the lower horizontal shift register. The right half of the lower horizontal shift register shifts out charges from right to left, that is, pixel n, pixel n-1, pixel n-2, and so on (where n is the last pixel in a line).

As the charges move out of the horizontal shift registers, they are converted to voltages proportional to the size of each charge. Each voltage is then amplified by a Variable Gain Control (VGC) and digitized by an Analog-to-Digital converter (ADC). For optimal digitization, gain and black level can be adjusted by setting camera parameters.

After each voltage has been amplified and digitized, it passes through an FPGA and into an image buffer. All shifting of charges from the vertical to the horizontal registers and out of the horizontal registers is clocked according to the camera's internal data rate. Shifting continues until all image data has been read out of the sensor.

As the pixel data passes through the FPGA and into the buffer, it is reordered so that the pixel data for each line is in ascending order from pixel 1 through pixel n.

There are **three "Camera Link tap geometries**" that can be used to determine how the pixel data in the image buffer will be transmitted over the Camera Link interface to the frame grabber in your computer:

- **1X-1Y** and **1X2-1Y**: When 1X-1Y or 1X2-1Y is selected, the pixel data in the image buffer will be transmitted line-by-line in ascending order from line one through the last line in the image. If the camera is set for 1X-1Y or 1X2-1Y, the entire image must be read out of the imaging sensor and stored in the image buffer before the camera can begin to transmit pixel data via the Camera Link interface.
- **1X-2YE**: When 1X-2YE is selected, first, the pixels in the first line in the image and pixels in the last line (line n) in the image will be transmitted. Next, the pixels in the second line of the image

and the pixels in the next to last line (line n-1) in the image will be transmitted. Next, the pixels in the third line of the image and in line n-2 will be transmitted. And so on.
If the camera is set for 1X-2YE, the camera can begin transmitting pixel data via the Camera Link interface before the entire image has been read out of the imaging sensor.

For more detailed information about tap geometries, see Section 8.2 on page 143.

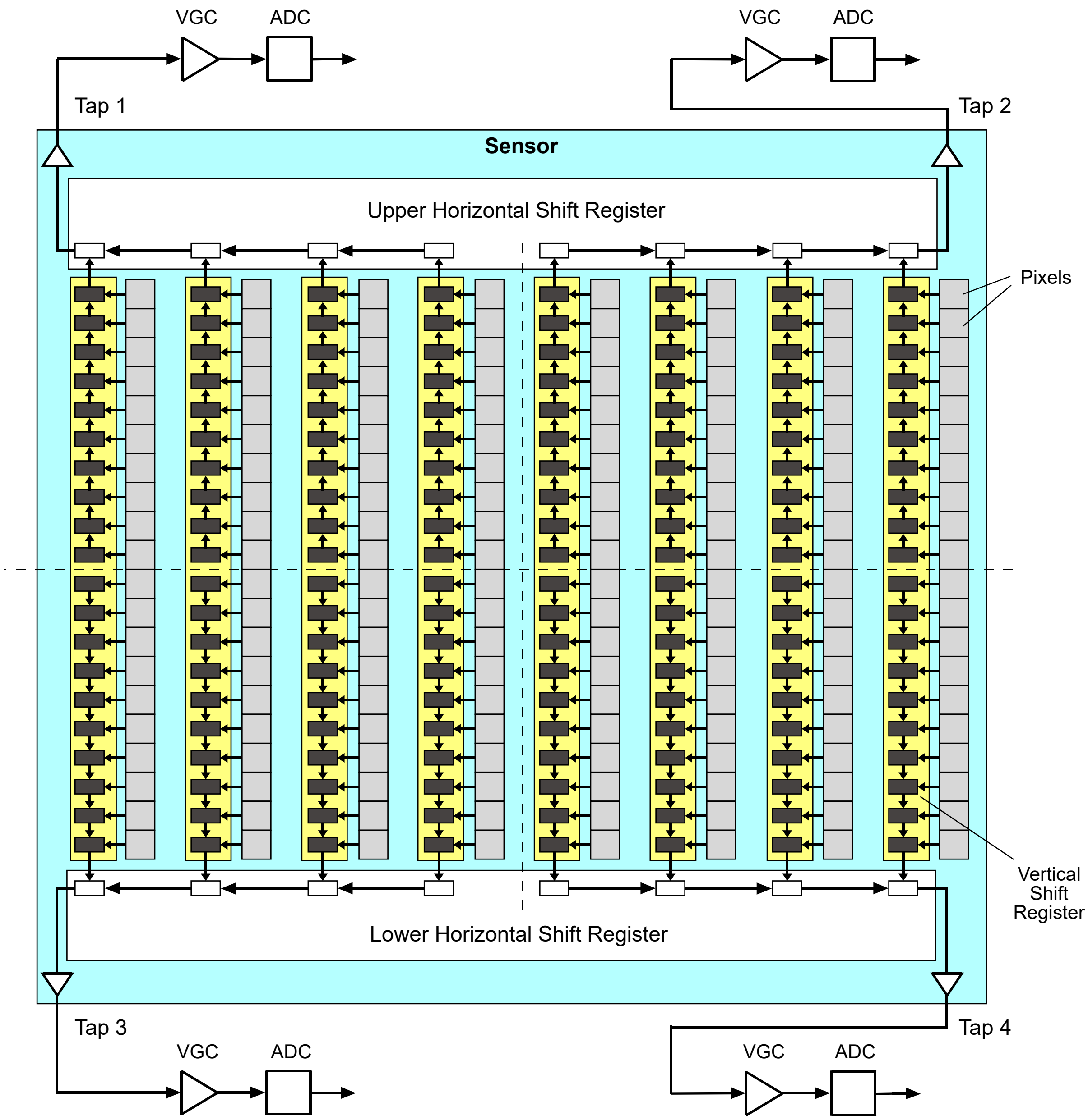


Fig. 17: Four Tap Sensor Readout Mode

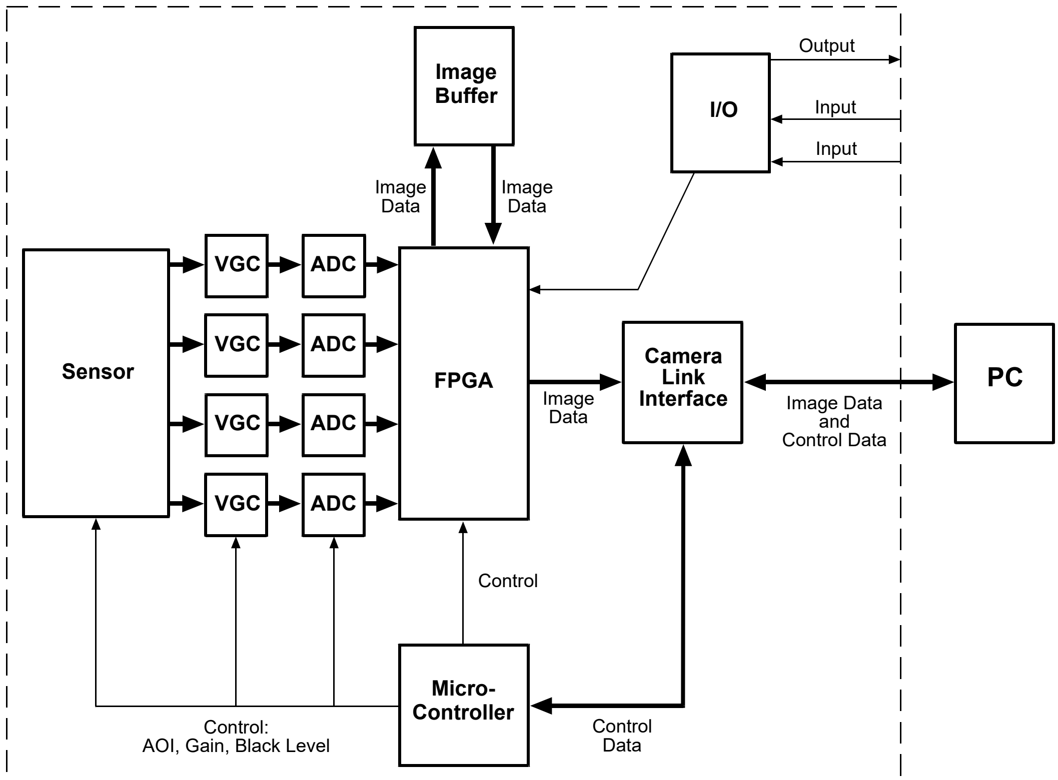


Fig. 18: Camera Block Diagram - Four Tap Mode

4.1.2 One Sensor Tap Digitization Mode

With one tap sensor digitization, only **one electronic circuit** is used to read out the pixels in the sensor (see Figure 19 on page 59). The advantage of the one tap digitization scheme is that it eliminates the need to balance four different readout circuits as is required with four tap readout. The drawback with one tap readout is that since it takes much longer to read out the sensor when using one tap, the camera's maximum achieveable frame rate is limited.

After an image has been captured (i.e., exposure has ended), the pixels in the sensor become ready to be read out. At the start of readout, the charges accumulated during exposure are transported from the pixels to the sensor's vertical shift registers. The charges from the top line of pixels in the array are then moved to the upper horizontal shift register as shown in Figure 19.

Next, the charges are moved out of the upper horizontal shift register. As the charges move out of the horizontal shift register, they are converted to voltages proportional to the size of each charge. Each voltage is then amplified by a Variable Gain Control (VGC) and digitized by an analog-to-digital converter (ADC). For optimal digitization, gain and black level can be adjusted by setting camera parameters.

After each voltage has been amplified and digitized, it passes through an FPGA and into an image buffer. All shifting of charges from the vertical to the horizontal register and out of the horizontal register is clocked according to the camera's internal data rate. Shifting continues in a line-by-line fashion until all image data has been read out of the sensor.

Once all of the pixel data has been read out of the imaging sensor and into the image buffer, the data can begin being transmitted via the Camera Link interface to the frame grabber in your computer.

When the camera is set for one tap digitization,

- the **1X2-1Y or** the **1X-1Y** "Camera Link tap geometry" **can be used** to transmit the pixel data over the Camera Link interface to the frame grabber in your computer.
- the **1X-2YE** "Camera Link tap geometry" **cannot be used**.

For more detailed information about tap geometries, see Section 8.2 on page 143.

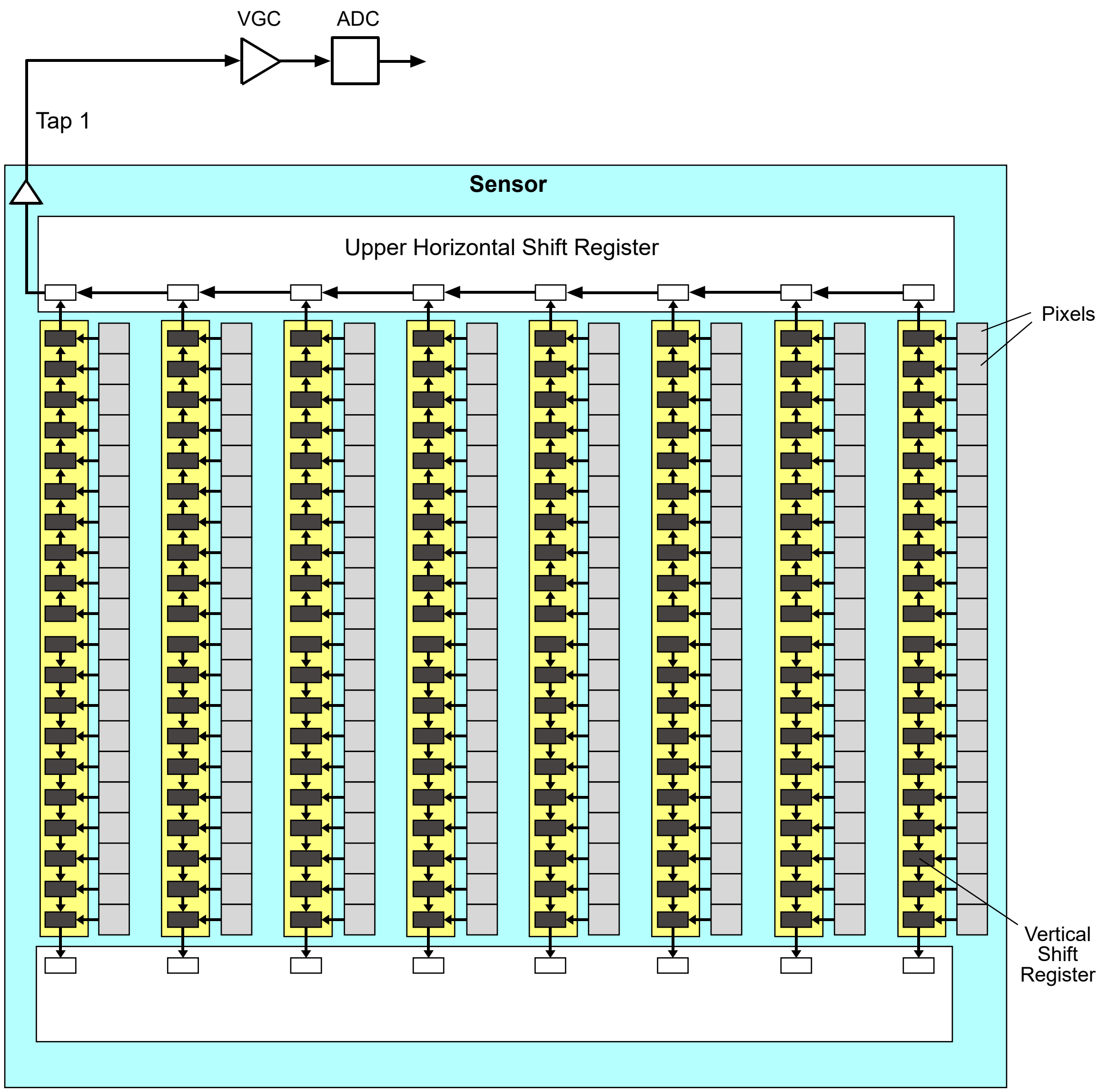


Fig. 19: One Tap Sensor Readout Mode

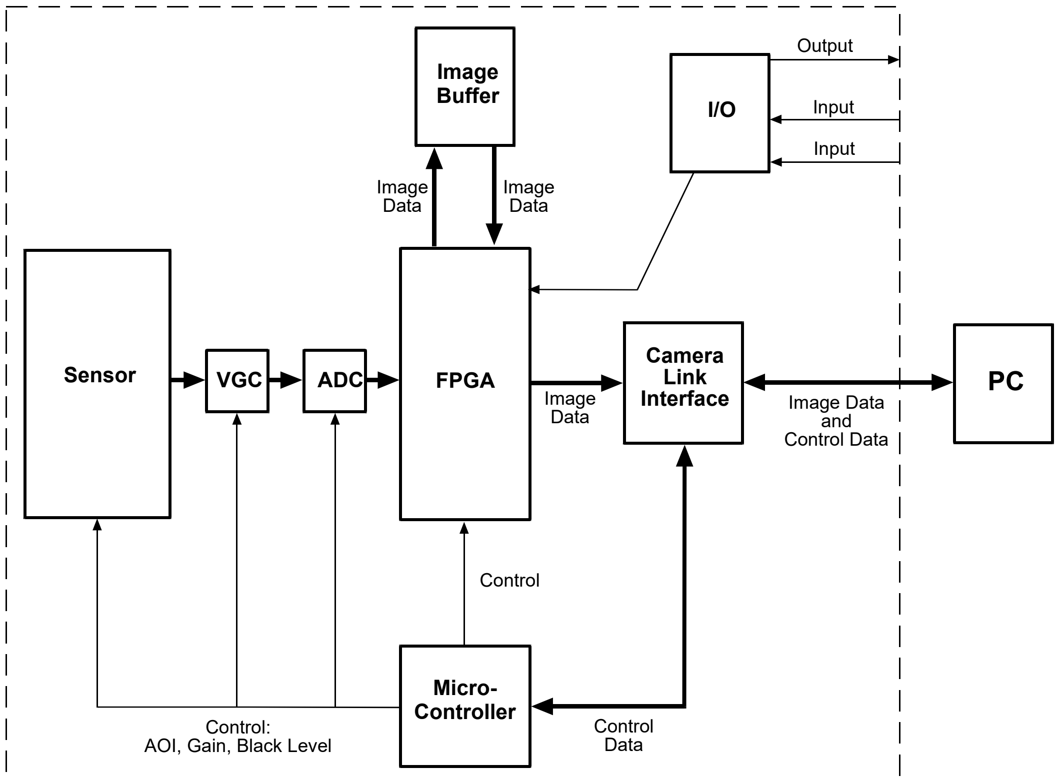


Fig. 20: Camera Block Diagram - One Tap Mode

5 Tools for Changing Camera Parameters

This chapter gives an overview of the options available for changing the camera's parameters:

- The Basler pylon Camera Software Suite provides options for changing parameters and controlling the camera by means of the
  - pylon Viewer (a stand-alone GUI) or by employing the
  - API to access the camera from within your software application (see below).
- Direct access to the camera's register structure provides another means for controlling the camera and changing parameters (see Section 5.2 on page 62).

5.1 Basler pylon Camera Software Suite

The Basler pylon Camera Software Suite is designed to operate all Basler cameras that have an IEEE 1394a/b interface, a GigE interface, or a USB 3.0 interface. The Basler pylon Camera Software Suite also operates newer Basler cameras with a Camera Link interface. The pylon Camera Software Suite offers reliable, real-time image data transport into the memory of your computer at a very low CPU load.

The options available with the Basler pylon Camera Software Suite let you

- change parameters and control the camera by using a standalone GUI known as the Basler pylon Viewer.
- change parameters and control the camera from within your software application using the Basler pylon SDKs.

You can download the Basler Camera Software Suite from the Basler website:

www.baslerweb.com

To help you install the necessary components for aviator Camera Link cameras, refer to the *Installation and Setup Guide for Cameras Used With the Basler pylon Camera Software Suite* (AW000611). You can download the guide from the Basler website: www.baslerweb.com

The pylon Camera Software Suite includes several tools that you can use to change the parameters on your camera, including the pylon Viewer and the pylon API for different programming languages. The remaining sections in this chapter provide an introduction to these tools.

5.1.1 pylon Viewer

The pylon Viewer is included in the Basler pylon Camera Software Suite. The pylon Viewer is a standalone application that lets you view and change most of the camera's parameter settings via a GUI. Using the pylon Viewer software is a very convenient way to get your camera up and running quickly when you are doing your initial camera evaluation or doing a camera design-in for a new project.

For more information about using the viewer, refer to the *Installation and Setup Guide for Cameras Used With the Basler pylon Camera Software Suite* (AW000611). You can download the guide from the Basler website:
www.baslerweb.com.

5.1.2 pylon SDKs

Three pylon SDKs are part of the Basler pylon Camera Software Suite:

- pylon SDK for C++ (Windows and Linux)
- pylon SDK for C (Windows and Linux)
- pylon SDK for .NET / C# (Windows)

Each SDK includes an application programming interface (API), a set of sample programs, and documentation:

- You can access all of the camera's parameters and control the camera's full functionality from within your application software by using the matching pylon API (C++, C, or .NET).
- The sample programs illustrate how to use the pylon API to parameterize and operate the camera.
- For each environment (C++, C, and .NET), a *Programmer's Guide and Reference Documentation* is available. The documentation gives an introduction to the pylon API and provides information about all methods and objects of the API.

The pylon SDKs are available from the Basler website: www.baslerweb.com

5.2 The Basler Binary Protocol Library

Basler aviator cameras have blocks of mapped memory space known as registers. By reading values from the registers, you can determine basic information about the camera and information about the camera's current settings. By writing values to the registers, you can control how the camera's features will operate.

If you use the Basler pylon software described in the previous section, the camera's register structure is hidden. With pylon, a series of function calls allows you to change camera parameter settings without the need to know anything about the register that underlies each parameter.

If you desire, you can also change the camera parameter settings by directly accessing the camera's register structure. The Basler Binary Protocol Library (BBPL) provides functions that allow you to read data from or write data to the camera's registers. The BBPL is an extension of the clALLSerial/clSerial API defined in Appendix B of the Camera Link Standard version 1.1 or higher. The BBPL adds convenience functions to this API that allow you to read from and write to the registers in Basler Camera Link cameras. The read and write requests are transmitted to the camera via a serial link between the camera and the frame grabber; the serial link is part of the standard Camera Link interface.

Sample code showing how to use the BBPL along with supporting documentation can be downloaded from the Basler website: www.baslerweb.com.

When using the BBPL to change parameter values, you will need to know the details of the camera's register structure. For details of the register structure, refer to the document called *aviator Register Structure and Access Methods* (AW000833). The document can be downloaded from the Basler website: www.baslerweb.com.

Note that if you are using an earlier Baser Camera Link camera that was originally designed to work with the Basler Binary Protocol II (e.g., the A400k, L400k, L800k, and the sprint), you can now use either the BBPL or the Binary Protocol II to access the camera's registers.

6 Image Acquisition Control

This chapter provides detailed information about controlling image acquisition. You will find information about triggering image acquisition, about setting the exposure time for acquired images, about controlling the camera's image acquisition rate, and about how the camera's maximum allowed image acquisition rate can vary depending on the current camera settings.

> **Note**
>
> The sample code included in this chapter represents "low level" code that is actually used by the camera.
> Many tasks, however, can be programmed more conveniently with fewer lines of code when employing the Instant Camera classes, provided by the Basler pylon C++ API. For information about the Instant Camera classes, see the *C++ Programmer's Guide and Reference Documentation* delivered with the Basler pylon Camera Software Suite.

6.1 Overview

This section presents an overview of the elements involved with controlling the acquisition of images. Reading this section will give you an idea about how these elements fit together and will make it easier to understand the detailed information in the sections that follow.

Three major elements are involved in controlling the acquisition of images:

- The acquisition start trigger
- The frame start trigger
- Exposure time control

When reading the explanations in the overview and in this entire chapter, keep in mind that the term "frame" is typically used to mean a single acquired image.

When reading the material in this chapter, it is helpful to refer to Figure 21 on page 66 and to the use case diagrams in Section 6.9 on page 115. These diagrams present the material related to the acquisition start trigger and the frame start trigger in a graphical format.

Acquisition Start Trigger

The acquisition start trigger is essentially an enabler for the frame start trigger.

The acquisition start trigger has two modes of operation: off and on.

If the Trigger Mode parameter for the acquisition start trigger is set to off, the camera will generate all required acquisition start trigger signals internally, and you do not need to apply acquisition start trigger signals to the camera.

If the Trigger Mode parameter for the acquisition start trigger is set to on, the initial acquisition status of the camera will be "waiting for acquisition start trigger" (see Figure 21 on page 66). When the camera is in this acquisition status, it cannot react to frame start trigger signals. When an acquisition start trigger signal is applied to the camera, the camera will exit the "waiting for acquisition start trigger" acquisition status and enter a "waiting for frame start trigger" acquisition status. The camera can then react to frame start trigger signals. The camera will continue to react to frame start trigger signals until the number of frame start trigger signals it has received is equal to an integer parameter setting called the Acquisition Frame Count. At that point, the camera will return to the "waiting for acquisition start trigger" acquisition status and will remain in that status until a new acquisition start trigger signal is applied.

As an example, assume that the Acquisition Frame Count parameter is set to three and that the camera is in a "waiting for acquisition start trigger" acquisition status. When an acquisition start trigger signal is applied to the camera, it will exit the "waiting for acquisition start trigger" acquisition status and enter the "waiting for frame start trigger" acquisition status. Once the camera has received three frame start trigger signals, it will return to the "waiting for acquisition start trigger" acquisition status. At that point, you must apply a new acquisition start trigger signal to the camera to make it exit "waiting for acquisition start trigger".

Frame Start Trigger

Assuming that an acquisition start trigger signal has just been applied to the camera, the camera will exit from the "waiting for acquisition start trigger" acquisition status and enter a "waiting for frame start trigger" acquisition status. Applying a frame start trigger signal to the camera at this point will exit the camera from the "waiting for frame start trigger" acquisition status and will begin the process of exposing and reading out a frame (see Figure 21 on page 66). As soon as the camera is ready to accept another frame start trigger signal, it will return to the "waiting for frame start trigger" acquisition status. A new frame start trigger signal can then be applied to the camera to begin another frame exposure.

The frame start trigger has two modes: off and on.

If the Trigger Mode parameter for the frame start trigger is set to off, the camera will generate all required frame start trigger signals internally, and you do not need to apply frame start trigger signals to the camera. The rate at which the camera will generate the signals and acquire frames will be determined by the way that you set several frame rate related parameters.

If the Trigger Mode parameter for the frame start trigger is set to on, you must trigger frame start by applying frame start trigger signals to the camera. Each time a trigger signal is applied, the camera will begin a frame exposure. When frame start is being triggered in this manner, it is important that you do not attempt to trigger frames at a rate that is greater than the maximum allowed. (There is a detailed explanation about the maximum allowed frame rate at the end of this chapter.) Frame start trigger signals that are applied to the camera when it is not in a "waiting for frame start trigger" acquisition status will be ignored.

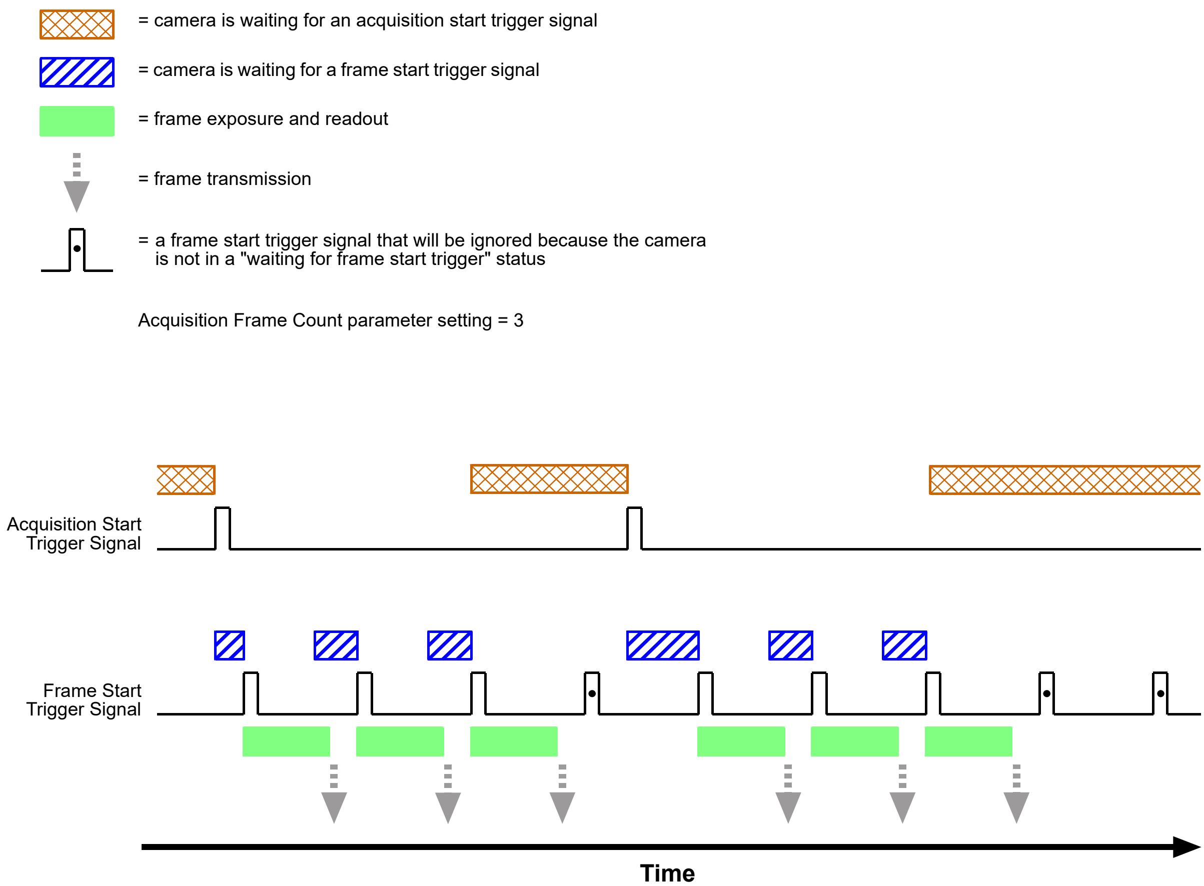


Fig. 21: Acquisition Start and Frame Start Triggering

Applying Trigger Signals

The paragraphs above mention "applying a trigger signal". There are two ways to apply an acquisition start or a frame start trigger signal to the camera: via software or via hardware.

To apply trigger signals

- **via software**, you must first select the acquisition start or the frame start trigger and then indicate that **software will be used as the source** for the selected trigger signal.
  At that point, the Basler pylon API or direct camera register access can be used to apply a software trigger signal to the camera for the selected trigger.

> **Note**
>
> When applying trigger signals via software there is a delay that depends on the Camera Link interface. Note that, as a consequence, software trigger signals cannot be used to realize real-time capable triggers.

- **via hardware**, you must first select the acquisition start or the frame start trigger.
  You then select one of the **camera's physical inputs to be used as the source** for the selected trigger. At that point, each time the proper **electrical signal** is applied to the selected input, an occurrence of the selected trigger signal will be recognized by the camera.

The Trigger Selector (This Only Applies When You Are Using the Basler pylon API)

If you are using the Basler pylon API to parameterize the camera, the concept of the "trigger selector" is very important to understand when working with the acquisition start and frame start triggers. Many of the parameter settings and the commands that apply to the triggers have names that are not specific to a particular type of trigger, for example, the acquisition start trigger has a mode setting and the frame start trigger has a mode setting. But in Basler pylon there is a single parameter, the Trigger Mode parameter, that is used to set the mode for both of these triggers. Also, the Trigger Software command mentioned earlier can be executed for either the acquisition start trigger or the frame start trigger. So if you want to set the Trigger Mode or execute a Trigger Software command for the acquisition start trigger rather than the frame start trigger, how do you do it? The answer is, by using the Trigger Selector parameter. Whenever you want to work with a specific type of trigger, your first step is to set the Trigger Selector parameter to the trigger you want to work with (either the acquisition start trigger or the frame start trigger). At that point, the changes you make to the Trigger Mode, Trigger Source, etc., will be applied to the selected trigger only.

Exposure Time Control

When a frame start trigger signal is applied to the camera, the camera will begin to acquire a frame. A critical aspect of frame acquisition is how long the pixels in the camera's sensor will be exposed to light during the frame acquisition.

If the camera is set for

- **software frame start triggering**, the camera’s Exposure Time parameter will determine the exposure time for each frame.
- **hardware frame start triggering**, there are two modes of operation:
  - With the **"timed" mode**, the Exposure Time parameter will determine the exposure time for each frame.
  - With the **"trigger width" mode**, the way that you manipulate the rise and fall of the hardware signal will determine the exposure time.

  The "trigger width" mode is especially useful,

  - if you want to change the exposure time from frame to frame, and
  - if you require exposure times that are longer than the maximum possible exposure time you can set via the exposure time parameter.

  You can use the sequencer feature as an alternative to the "trigger width" mode if you require exposure times that are periodically changing from frame to frame. For information on the sequencer feature, see Section 9.11 on page 197.

6.2 The Acquisition Start Trigger

(When reading this section, it is helpful to refer to Figure 21 on page 66.)

The acquisition start trigger is used in conjunction with the frame start trigger to control the acquisition of frames. In essence, the acquisition start trigger is used as an enabler for the frame start trigger. Acquisition start trigger signals can be generated within the camera or may be applied externally as software or hardware acquisition start trigger signals.

When the acquisition start trigger is enabled, the camera's initial acquisition status is "waiting for acquisition start trigger". When the camera is in this acquisition status, it will ignore any frame start trigger signals it receives. If an acquisition start trigger signal is applied to the camera, it will exit the "waiting for acquisition start trigger" acquisition status and enter the "waiting for frame start trigger" acquisition status. In this acquisition status, the camera can react to frame start trigger signals and will begin to expose a frame each time a proper frame start trigger signal is applied.

A primary feature of the acquisition start trigger is that after an acquisition start trigger signal has been applied to the camera and the camera has entered the "waiting for frame start trigger" acquisition status, the camera will return to the "waiting for acquisition start trigger" acquisition status once a specified number of frame start triggers has been received. Before more frames can be acquired, a new acquisition start trigger signal must be applied to the camera to exit it from "waiting for acquisition start trigger" status. Note that this feature only applies when the Trigger Mode parameter for the acquisition start trigger is set to on. This feature is explained in greater detail in the following sections.

6.2.1 Acquisition Start Trigger Mode

The main parameter associated with the acquisition start trigger is the Trigger Mode parameter. The Trigger Mode parameter for the acquisition start trigger has two available settings: off and on.

6.2.1.1 Acquisition Start Trigger Mode = Off

When the Trigger Mode parameter for the acquisition start trigger is set to off, the camera will generate all required acquisition start trigger signals internally, and you do not need to apply acquisition start trigger signals to the camera.

6.2.1.2 Acquisition Start Trigger Mode = On

When the Trigger Mode parameter for the acquisition start trigger is set to on, the camera will initially be in a "waiting for acquisition start trigger" acquisition status and cannot react to frame start trigger signals. You must apply an acquisition start trigger signal to the camera to exit the camera from the "waiting for acquisition start trigger" acquisition status and enter the "waiting for frame start trigger" acquisition status. The camera can then react to frame start trigger signals and will continue to do so until the number of frame start trigger signals it has received is equal to the current Acquisition

Frame Count parameter setting. The camera will then return to the "waiting for acquisition start trigger" acquisition status. In order to acquire more frames, you must apply a new acquisition start trigger signal to the camera to exit it from the "waiting for acquisition start trigger" acquisition status.

When the **Trigger Mode** parameter for the acquisition start trigger is set to on, you must select a source signal to act as the acquisition start trigger. The **Trigger Source** parameter specifies the source signal. The available selections for the Trigger Source parameter are:

- **Software** - When the source signal is set to software, you apply an acquisition start trigger signal to the camera by executing a **Trigger Software command** for the acquisition start trigger on the host computer.
- **Line 1** - When the source signal is set to line 1, you apply an acquisition start trigger signal to the camera by injecting an externally generated electrical signal (commonly referred to as a hardware trigger signal) into **physical input line 1 on the camera**.
- **Line 2** - When the source signal is set to line 2, you apply an acquisition start trigger signal to the camera by injecting an externally generated electrical signal into **physical input line 2** on the camera.
- **CC1** - When the source signal is set to CC1, you apply an acquisition start trigger signal to the camera by injecting an externally generated electrical signal into CC1 in the Camera Link interface.
- **CC2** - When the source signal is set to CC2, you apply an acquisition start trigger signal to the camera by injecting an externally generated electrical signal into CC2 in the Camera Link interface.
- **CC4** - When the source signal is set to CC4, you apply an acquisition start trigger signal to the camera by injecting an externally generated electrical signal into CC4 in the Camera Link interface.

If the Trigger Source parameter for the acquisition start trigger is set to line 1, line 2, CC1, CC2, or CC4, you must also set the **Trigger Activation** parameter. The available settings for the Trigger Activation parameter are:

- Rising Edge - specifies that a rising edge of the hardware trigger signal will act as the acquisition start trigger.
- Falling Edge - specifies that a falling edge of the hardware trigger signal will act as the acquisition start trigger.

> Typically, a frame grabber is used to supply the electrical frame start trigger signal to CC1, CC2, or CC4.

For more detailed information about

- input line 1 and input line 2 on the camera, see Section 2.3.3 on page 27.
- the CC1, CC2, and CC4 inputs in the Camera Link interface, refer to the document called *Aviator Information for Frame Grabber Designers* (AW000831). You can obtain the document from the Basler website: www.baslerweb.com

6.2.2 Acquisition Frame Count

When the Trigger Mode parameter for the acquisition start trigger is set to on, you must set the value of the camera's Acquisition Frame Count parameter. The value of the Acquisition Frame Count can range from 1 to 255.

With acquisition start triggering on, the camera will initially be in a "waiting for acquisition start trigger" acquisition status. When in this acquisition status, the camera cannot react to frame start trigger signals. If an acquisition start trigger signal is applied to the camera, the camera will exit the "waiting for acquisition start trigger" acquisition status and will enter the "waiting for frame start trigger" acquisition status. It can then react to frame start trigger signals. When the camera has received a number of frame start trigger signals equal to the current Acquisition Frame Count parameter setting, it will return to the "waiting for acquisition start trigger" acquisition status. At that point, you must apply a new acquisition start trigger signal to exit the camera from the "waiting for acquisition start trigger" acquisition status.

6.2.3 Setting The Acquisition Start Trigger Mode and Related Parameters

Setting the Parameters Using Basler pylon

You can set the Trigger Mode and Trigger Source parameter values for the acquisition start trigger and the Acquisition Frame Count parameter value from within your application software by using the pylon API.

The following code snippet illustrates using the pylon API to set the acquisition start Trigger Mode to on, the Trigger Source to software, and the Acquisition Frame Count to 5:

```
// Select the acquisition start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_AcquisitionStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_On );
// Set the source for the selected trigger
Camera.TriggerSource.SetValue ( TriggerSource_Software );
// Set the acquisition frame count
Camera.AcquisitionFrameCount.SetValue( 5 );
```

The following code snippet illustrates using the pylon API to set the Trigger Mode to on, the Trigger Source to line 1, the Trigger Activation to rising edge, and the Acquisition Frame Count to 5:

```
// Select the acquisition start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_AcquisitionStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_On );
// Set the source for the selected trigger
```

```
Camera.TriggerSource.SetValue ( TriggerSource_Line1 );
// Set the activation mode for the selected trigger to rising edge
Camera.TriggerActivation.SetValue( TriggerActivation_RisingEdge );
// Set the acquisition frame count
Camera.AcquisitionFrameCount.SetValue( 5 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Parameters Using Direct Register Access

To set the parameters related to the acquisition start trigger via direct register access:

- Set the value of the Trigger Mode Acquisition Start register to on.
- Set the value of the Trigger Source Acquisition Start register to software, line 1, line 2, CC1, CC2, or CC4.
- If the trigger source is set to line 1, line 2, CC1, CC2, or CC4, set the value of the Trigger Activation Acquisition Start register to rising edge or falling edge.
- Set the value of the Acquisition Frame Count register as desired.

For more information about changing settings via direct register access, see Section 5 on page 61.

6.2.4 Using a Software Acquisition Start Trigger

6.2.4.1 Introduction

If the camera’s Trigger Mode parameter for the acquisition start trigger is set to on and the Trigger Source parameter is set to software, you must apply a software acquisition start trigger signal to the camera before you can begin frame acquisition.

The camera will initially be in a "waiting for acquisition start trigger" acquisition status. It cannot react to frame trigger signals when in this acquisition status. When a software acquisition start trigger signal is received by the camera, it will exit the "waiting for acquisition start trigger" acquisition status and will enter the "waiting for frame start trigger" acquisition status. It can then react to frame start trigger signals. When the number of frame start trigger signals received by the camera is equal to the current Acquisition Frame Count parameter setting, the camera will return to the "waiting for acquisition start trigger" acquisition status. When a new software acquisition start trigger signal is applied to the camera, it will again exit from the "waiting for acquisition start trigger" acquisition status and enter the "waiting for frame start trigger" acquisition status.

Section 6.2.4.2 includes more detailed information about applying a software acquisition start trigger to the camera using Basler pylon or via direct register access.

6.2.4.2 Setting the Parameters Related to Software Acquisition Start Triggering and Applying a Software Trigger Signal

Setting the Parameters and Applying the Signal Using Basler pylon

You can set all of the parameters needed to perform software acquisition start triggering from within your application software by using the pylon API. The following code snippet illustrates using the pylon API to set the parameter values and execute the commands related to software acquisition start triggering:

```
// Select the acquisition start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_AcquisitionStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_On );
// Set the source for the selected trigger
Camera.TriggerSource.SetValue ( TriggerSource_Software );
// Set the acquisition frame count
Camera.AcquisitionFrameCount.SetValue( 5 );
// Execute a trigger software command to apply a software acquisition start trigger
// signal to the camera
Camera.TriggerSoftware.Execute( );

// Note: as long as the Trigger Selector is set to Acquisition Start, executing
// a Trigger Software command will apply an acquisition start software trigger
```

```
// signal to the camera
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Parameters and Applying the Signal Using Direct Register Access

To set the parameters needed to perform software acquisition start triggering via direct register access:

- Set the value of the Trigger Mode Acquisition Start register to On.
- Set the value of the Trigger Source Acquisition Start register to Software.
- Set the value of the Acquisition Frame Count register as desired.
- Set the value of the Trigger Software Acquisition Start register to 1.
  Setting the value of this register to 1 applies a software acquisition start trigger signal to the camera. The register resets to 0 when execution is complete.

For more information about changing settings via direct register access, see Section 5 on page 61.

6.2.5 Using a Hardware Acquisition Start Trigger

6.2.5.1 Introduction

If the Trigger Mode parameter for the acquisition start trigger is set to on and the Trigger Source parameter is set to line 1, line 2, CC1, CC2, or CC4, an externally generated electrical signal injected into the selected source will act as the acquisition start trigger signal for the camera. This type of trigger signal is generally referred to as a hardware trigger signal or as an external acquisition start trigger signal (ExASTrig).

A rising edge or a falling edge of the ExASTrig signal can be used to trigger acquisition start. The Trigger Activation parameter is used to select rising edge or falling edge triggering.

When the Trigger Mode parameter is set to on, the camera will initially be in a "waiting for acquisition start trigger" acquisition status. It cannot react to frame start trigger signals when in this acquisition status. When the appropriate ExASTrig signal is applied to the selected source (e.g, a rising edge of the signal for rising edge triggering), the camera will exit the "waiting for acquisition start trigger" acquisition status and will enter the "waiting for frame start trigger" acquisition status. It can then react to frame start trigger signals. When the number of frame start trigger signals received by the camera is equal to the current Acquisition Frame Count parameter setting, the camera will return to the "waiting for acquisition start trigger" acquisition status. When a new ExASTrig signal is applied to the selected source, the camera will again exit from the "waiting for acquisition start trigger" acquisition status and enter the "waiting for frame start trigger" acquisition status.

For more information about setting the camera for hardware acquisition start triggering and selecting the source to receive the ExASTrig signal, see Section 6.2.5.2 on page 75.

For more information about the electrical requirements for line 1 and line 2, see Section 2.3.3 on page 27.

For more information about CC1, CC2, and CC4, see Section 2.4.1 on page 33.

6.2.5.2 Setting the Parameters Related to Hardware Acquisition Start Triggering and Applying a Hardware Trigger Signal

Setting the Parameters Using Basler pylon and Applying a Signal

You can set all of the parameters needed to perform hardware acquisition start triggering from within your application by using the pylon API. The following code snippet illustrates using the pylon API to set the parameter values required to enable rising edge hardware acquisition start triggering with line 1 as the trigger source:

```
// Select the acquisition start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_AcquisitionStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_On );
// Set the source for the selected trigger
Camera.TriggerSource.SetValue ( TriggerSource_Line1 );
```

```
// Set the activation mode for the selected trigger to rising edge
Camera.TriggerActivation.SetValue( TriggerActivation_RisingEdge );
// Set the acquisition frame count
Camera.AcquisitionFrameCount.SetValue( 5 );

// Apply a rising edge of the externally generated electrical signal
// (ExASTrig signal) to line 1 on the camera
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Parameters Using Direct Register Access and Applying a Signal

To set the parameters needed to perform hardware acquisition start triggering via direct register access:

- Set the value of the Trigger Mode Acquisition Start register to on.
- Set the value of the Trigger Source Acquisition Start register to line 1, line 2, CC1, CC2, or CC4.
- Set the value of the Trigger Activation Acquisition Start register to rising edge or falling edge.
- Set the value of the Acquisition Frame Count register as desired.

Apply the appropriate externally generated electrical signal (ExASTrig signal) to the selected trigger source.

For more information about changing settings via direct register access, see Section 5 on page 61.

6.3 The Frame Start Trigger

The frame start trigger is used to begin frame acquisition.

Assuming that the camera is in a "waiting for frame start trigger" acquisition status, it will begin a frame acquisition each time it receives a frame start trigger signal.

(For a quick overview of acquisition start triggering and frame start triggering, see Section 6.1 on page 64.)

When reading this section, it is helpful to refer to Figure 21 on page 66 and the use case diagrams that appear in Section 6.9 on page 115.

6.3.1 Frame Start Trigger Mode

The main parameter associated with the frame start trigger is the Trigger Mode parameter. The Trigger Mode parameter for the frame start trigger has two available settings: off and on.

6.3.1.1 Frame Start Trigger Mode = Off

When the Trigger Mode parameter for the frame start trigger is set to off, the camera will generate all required frame start trigger signals internally, and you do not need to apply frame start trigger signals to the camera.

If you are using **Basler pylon** to parameterize the camera, the rate at which the frame start trigger signals will be generated is determined by the camera's Acquisition Frame Rate Abs parameter:

- If the parameter is not enabled, the camera will generate frame start trigger signals at the maximum rate allowed given the current camera settings.
- If the parameter is enabled and is set to a value less than the maximum allowed frame rate, the camera will generate frame start trigger signals at the rate specified by the parameter setting.
- If the parameter is enabled and is set to a value greater than the maximum allowed frame rate, the camera will generate frame start trigger signals at the maximum allowed frame rate.

If you are using **direct register access** to parameterize the camera, the rate at which the frame start trigger signals will be generated is determined by the camera's Acquisition Frame Period Raw parameter (Frame Rate = 1/ Frame Period):

- If the parameter is not enabled, the camera will generate frame start trigger signals at the maximum rate allowed given the current camera settings.
- If the parameter is enabled and is set to a value that would result in a frame rate less than the maximum allowed, the camera will generate frame start trigger signals at the rate that results from the parameter setting.
- If the parameter is enabled and is set to a value that would result in a frame rate greater than the maximum allowed given the current camera settings, the camera will generate frame start trigger signals at the maximum allowed frame rate.

> The camera will only react to frame start triggers when it is in a "waiting for frame start tigger" acquisition status. For more information about the acquisition status, see Section 6.1 on page 64 and Section 6.2 on page 69.

Exposure Time Control with the Frame Start Trigger Off

When the Trigger Mode parameter for the frame start trigger is set to off, the exposure time for each frame acquisition is determined by:

- the value of the camera's Exposure Time Abs parameter, if you are parameterizing the camera using Basler pylon.
- the value in the camera's Exposure Time Raw register, if you are parameterizing the camera using direct register access.

For more information about setting the Exposure Time Abs parameter or the Exposure Time Raw register, see Section 6.4 on page 92.

6.3.1.2 Frame Start Trigger Mode = On

When the Trigger Mode parameter for the frame start trigger is set to on, you must apply a frame start trigger signal to the camera each time that you want to begin a frame acquisition. The Trigger Source parameter specifies the source signal that will act as the frame start trigger. The available selections for the Trigger Source parameter are:

- Software - When the source signal is set to software, you apply a frame start trigger signal to the camera by executing a Trigger Software command for the frame start trigger on the host .
- Line 1 - When the source signal is set to line 1, you apply a frame start trigger signal to the camera by injecting an externally generated electrical signal (commonly referred to as a hardware trigger signal) into physical input line 1 on the camera.
- Line 2 - When the source signal is set to line 2, you apply a frame start trigger signal to the camera by injecting an externally generated electrical signal into physical input line 2 on the camera.
- CC1 - When the source signal is set to CC1, you apply a frame start trigger signal to the camera by injecting an externally generated electrical signal into CC1 in the Camera Link interface.
- CC2 - When the source signal is set to CC2, you apply a frame start trigger signal to the camera by injecting an externally generated electrical signal into CC2 in the Camera Link interface.
- CC4 - When the source signal is set to CC4, you apply a frame start trigger signal to the camera by injecting an externally generated electrical signal into CC4 in the Camera Link interface.

If the Trigger Source parameter is set to line 1, line 2, CC1, CC2, or CC4, you must also set the Trigger Activation parameter. The available settings for the Trigger Activation parameter are:

- Rising Edge - specifies that a rising edge of the hardware trigger signal will act as the frame start trigger.
- Falling Edge - specifies that a falling edge of the hardware trigger signal will act as the frame start trigger.

> Typically, a frame grabber is used to supply an electrical frame start signal to CC1, CC2, or CC4.
>
> By default, CC1 is selected as the source signal for the frame start trigger.

For more information about

- using a software trigger to control frame acquisition start, see Section 6.3.2 on page 82.
- using a hardware trigger to control frame acquisition start, see Section 6.3.3 on page 86.

Exposure Time Control with the Frame Start Trigger On

When the Trigger Mode parameter for the frame start trigger is set to on and the Trigger Source parameter is set to software, the exposure time for each frame acquisition is determined by:

- the value of the camera's Exposure Time Abs parameter, if you are parameterizing the camera using Basler pylon.
- the value in the camera's Exposure Time Raw register, if you are parameterizing the camera using direct register access.

When the Trigger Mode parameter is set to on and the Trigger Source parameter is set to line 1, line 2, CC1, CC2, or CC4, the exposure time for each frame acquisition can be controlled using one of the values mentioned in the two bullet points above or it can be controlled by manipulating the hardware trigger signal.

For more information about controlling exposure time

- when using a software trigger, see Section 6.3.2 on page 82.
- when using a hardware trigger, see Section 6.3.3 on page 86.

6.3.1.3 Setting the Frame Start Trigger Mode and Related Parameters

Setting the Parameters Using Basler pylon

You can set the Trigger Mode and related parameter values from within your application software by using the pylon API. If your settings make it necessary, you can also set the Trigger Source parameter.

The following code snippet illustrates using the pylon API to set the Trigger Mode parameter for the frame start trigger to on and the Trigger Source parameter to CC1:

```
// Select the frame start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_FrameStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_On );
// Set the source for the selected trigger
Camera.TriggerSource.SetValue ( TriggerSource_CC1 );
```

The following code snippet illustrates using the pylon API to set the Trigger Mode parameter for the frame start trigger to off and the Acquisition Frame Rate Abs parameter to 60:

```
// Select the frame start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_FrameStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_Off );
// Set the exposure time
Camera.ExposureTimeAbs.SetValue( 3000 );
// Enable the acquisition frame rate parameter and set the frame rate. (Enabling
// the acquisition frame rate parameter allows the camera to control the frame
// rate internally.
Camera.AcquisitionFrameRateEnable.SetValue( true );
Camera.AcquisitionFrameRateAbs.SetValue( 60.0 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Parameters Using Direct Register Access

To set the trigger mode for the frame start trigger to on and to select a trigger source via direct register access:

- Set the value of the Trigger Mode Frame Start register to on.
- Set the value of the Trigger Source Frame Start register to software, line 1, line 2, CC1, CC2, or CC4.
- If the trigger source is set to line 1, line 2, CC1, CC2, or CC4, set the value of the Trigger Activation Frame Start register to rising edge or falling edge.

To set the trigger mode for the frame start trigger to off, set the exposure time, and set the frame acquisition rate via direct register access:

- Set the value of the Trigger Mode Frame Start register to off.
- Set the value of the Exposure Time Raw register as desired.

  A value in a raw register is simply an integer value with no units. To determine what the actual setting will be, you must multiply the value in the raw register by the camera’s time base. The time base on aviator cameras is 1 µs.

  For example, if you set the Exposure Time Raw register to 1000, the exposure time would be 1000 µs (1000 x 1 µs = 1000 µs).
- Set the value of the Acquisition Frame Period Enable register to true.
  (This will enable the camera’s ability to internally control the frame period.)
- Set the value of the Acquisition Frame Period Raw register as desired.
  (Frame Rate = 1 /Frame Period.)

For more information about changing settings via direct register access, see Section 5 on page 61.

6.3.2 Using a Software Frame Start Trigger

6.3.2.1 Introduction

If the Trigger Mode parameter for the frame start trigger is set to on and the Trigger Source parameter is set to software, you must apply a software frame start trigger signal to the camera to begin each frame acquisition. Assuming that the camera is in a "waiting for frame start trigger" acquisition status, frame exposure will start when the software frame start trigger signal is received by the camera. Figure 22 illustrates frame acquisition with a software frame start trigger signal.

When the camera receives a software trigger signal and begins exposure, it will exit the "waiting for frame start trigger" acquisition status because at that point, it cannot react to a new frame start trigger signal. As soon as the camera is capable of reacting to a new frame start trigger signal, it will automatically return to the "waiting for frame start trigger" acquisition status.

When you are using a software trigger signal to start each frame acquisition, the camera's Exposure Mode parameter must be set to timed. The exposure time for each acquired frame will be determined by the value of the camera's Exposure Time Abs parameter if you are parameterizing the camera with Basler pylon or by the Exposure Time Raw parameter if you are parameterizing the camera via direct register access.

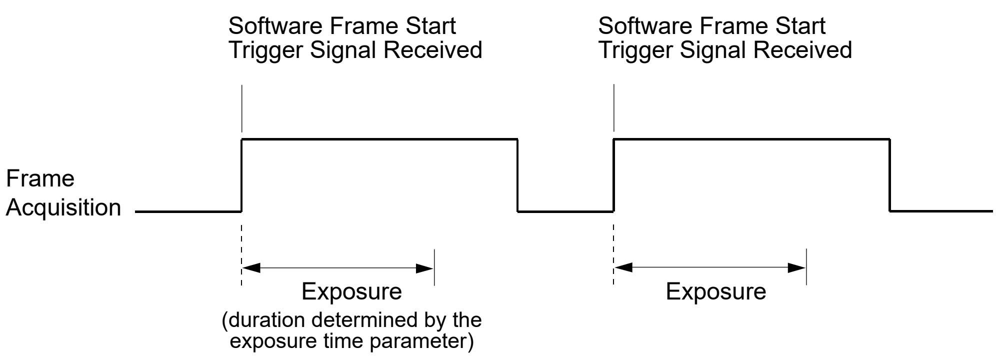


Fig. 22: Frame Acquisition with a Software Frame Start Trigger

When you are using a software trigger signal to start each frame acquisition, the frame rate will be determined by how often you apply a software trigger signal to the camera, and you should not attempt to trigger frame acquisition at a rate that exceeds the maximum allowed with the current camera settings. (There is a detailed explanation about the maximum allowed frame rate at the end of this chapter.) Software frame start trigger signals that are applied to the camera when it is not ready to receive them will be ignored.

Section 6.3.2.3 includes more detailed information about applying a software frame start trigger signal to the camera using Basler pylon or via direct register access.

For more information about determining the maximum allowed frame rate with the current camera settings, see Section 6.8 on page 112.

6.3.2.2 Acquisition Status

If the camera is currently in the process of acquiring a frame, it may not be in a "waiting for frame start trigger" acquisition status (i.e., it may not be ready to receive a new software frame start trigger signal and begin acquiring a new frame).

You can use the acquisition status feature to determine whether the camera is ready to perform a new frame acquisition. If you check the acquisition status before you apply each software frame start trigger signal, you can avoid triggering the camera at a rate that exceeds the maximum allowed with the current camera settings.

To determine the acquisition status of the camera via the Basler pylon API:

- Use the Acquisition Status Selector to select the Frame Trigger Wait status.
- Read the value of the Acquisition Status parameter.

  If the value is set to "false", the camera is not ready to receive a frame start trigger signal.

  If the value is set to "true", the camera is ready to receive a frame start trigger signal.

To determine the acquisition status of the camera via direct register access:

- Read the value of the Status Frame Trigger Wait register.

  If the value is set to 0 the camera is not ready to receive a frame start trigger signal.

  If the value is set to 1, the camera is ready to receive a frame start trigger signal.

Section 6.3.2.3 includes more detailed information about checking the acquisition status using Basler pylon or via direct register access.

6.3.2.3 Setting the Parameters Related to Software Frame Start Triggering and Applying a Software Trigger Signal

Setting the Parameters and Applying the Signal Using Basler pylon

You can set all of the parameters needed to perform software frame start triggering from within your application software by using the Basler pylon API. The following code snippet illustrates using the pylon API to set the parameter values and execute the commands related to software frame start triggering. In this example, the acquisition start trigger mode will be set to off:

```
// Select the acquisition start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_AcquisitionStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_Off );
// Disable the acquisition frame rate parameter (this will disable the camera’s
// internal frame rate control and allow you to control the frame rate with
// software frame start trigger signals)
Camera.AcquisitionFrameRateEnable.SetValue( false );
// Select the frame start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_FrameStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_On );
// Set the source for the selected trigger
Camera.TriggerSource.SetValue ( TriggerSource_Software );
// Set for the timed exposure mode
Camera.ExposureMode.SetValue( ExposureMode_Timed );
// Set the exposure time
Camera.ExposureTimeAbs.SetValue( 3000 );

     while ( ! finished )
     {
         // Execute a trigger software command to apply a frame start
         // trigger signal to the camera
         Camera.TriggerSoftware.Execute( );
         // Retrieve acquired frame here
     }

// Note: as long as the Trigger Selector is set to FrameStart, executing
// a Trigger Software command will apply a software frame start trigger
// signal to the camera
```

The following code snippet illustrates using the pylon API to check the acquisition status:

```
// Set the acquisition status selector
Camera.AcquisitionStatusSelector.SetValue
( AcquisitionStatusSelector_FrameTriggerWait );

// Read the acquisition status
```

```
bool IsWaitingForFrameTrigger = Camera.AcquisitionStatus.GetValue();
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Parameters and Applying the Signal Using Direct Register Access

To set the parameters needed to perform software frame start triggering via direct register access (with the acquisition start trigger mode set to off):

- Set the value of the Trigger Mode Acquisition Start register to off.
- Set the value of the Acquisition Frame Period Enable register to false.
  (This will disable the camera's ability to internally control the frame period and allow you to control the frame rate with a software trigger signals.)
- Set the value of the Trigger Mode Frame Start register to on.
- Set the value of the Trigger Source Frame Start register to software.
- Set the value of the Exposure Mode register to timed.
- Set the value of the Exposure Time Raw parameter as desired.

  A value in a raw register is simply an integer value with no units. To determine what the actual setting will be, you must multiply the value in the raw register by the camera's time base. The time base on aviator cameras is 1 µs.

  For example, if you set the Exposure Time Raw register to 1000, the exposure time would be 1000 µs (1000 x 1 µs = 1000 µs).
- Set the value of the Trigger Software Frame Start register to 1.
  Setting the value of this register to 1 applies a software frame start trigger to the camera. The register resets to 0 when execution is complete.

To determine the acquisition status of the camera via direct register access:

- Read the value of the Status Frame Trigger Wait register.
  A value of 0 indicates that the camera is not ready to receive a frame start trigger.
  A value of 1 indicates that the camera is ready to receive a frame start trigger.

For more information about changing settings via direct register access, see Section 5 on page 61.

6.3.3 Using a Hardware Frame Start Trigger

6.3.3.1 Introduction

If the Trigger Mode parameter for the frame start trigger is set to on and the Trigger Source parameter is set to line 1, line 2, CC1, CC2, or CC4, an externally generated electrical signal injected into the selected source will act as the frame start trigger signal for the camera. This type of trigger signal is generally referred to as a hardware trigger signal or as an external frame start trigger signal (ExFSTrig signal).

A rising edge or a falling edge of the ExFSTrig signal can be used to trigger frame acquisition. The Trigger Activation parameter is used to select rising edge or falling edge triggering.

Assuming that the camera is in a "waiting for frame start trigger" acquisition status, frame acquisition will start when the hardware frame start trigger signal is received by the camera.

When the camera receives a hardware trigger signal and begins exposure, it will exit the "waiting for frame start trigger" acquisition status because at that point, it cannot react to a new frame start trigger signal. As soon as the camera is capable of reacting to a new frame start trigger signal, it will automatically return to the "waiting for frame start trigger" acquisition status.

When the camera is operating under control of an ExFSTrig signal, the period of the ExFSTrig signal will determine the rate at which the camera will acquire frames:

$$\frac{1}{\text{ExFSTrig period in seconds}} = \text{Frame Rate}$$

For example, if you are operating a camera with an ExFSTrig signal period of 20 ms (0.020 s):

$$\frac{1}{0.020} = 50 \text{ fps}$$

So in this case, the frame rate is 50 fps.

If you have selected CC1, CC2, or CC4 as the trigger source, your frame grabber will typically apply the electrical signal to the selected input via the Camera Link cable. For more information about applying an ExFSTrig signal to CC1, CC2, or CC4, see the documentation for your frame grabber.

If you have selected line 1 or line 2 as the trigger source, some other kind of external electrical device will be used to apply the electrical signal to the selected input.

If you are triggering frame acquisition with an ExFSTrig signal and you attempt to acquire frames at too high a rate, some of the frame trigger signals that you apply will be received by the camera when it is not in a "waiting for frame start trigger" acquisition status. The camera will ignore any frame start trigger signals that it receives when it is not "waiting for frame start trigger". (This situation is commonly referred to as "over triggering" the camera.

To avoid over triggering, you should not attempt to acquire frames at a rate that exceeds the maximum allowed with the current camera settings.

For more information about

- determining the maximum allowed frame rate with the current camera settings, see Section 6.8 on page 112.
- setting the camera for hardware triggering and selecting the source to receive the ExFSTrig signal, see Section 6.3.3.4 on page 90.
- the electrical requirements for line 1 or line 2, see Section 2.3.3 on page 27.

6.3.3.2 Exposure Modes

If you are triggering the start of frame acquisition with an externally generated frame start trigger (ExFSTrig) signal, two exposure modes are available: timed and trigger width.

Timed Exposure Mode

When timed mode is selected, the exposure time for each frame acquisition is determined by:

- the value of the camera's Exposure Time Abs parameter, if you are parameterizing the camera with Basler pylon.
- the value of the Exposure Time Raw register if you are parameterizing the camera via direct register access.

If the camera is set

- for rising edge triggering, the exposure time starts when the ExFSTrig signal rises.
- for falling edge triggering, the exposure time starts when the ExFSTrig signal falls.

Figure 23 illustrates timed exposure with the camera set for rising edge triggering.

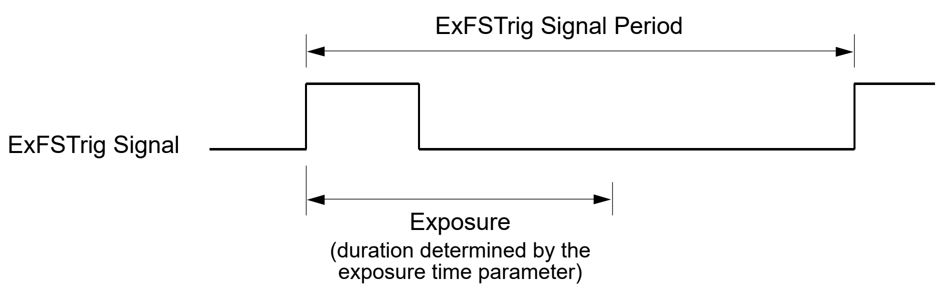


Fig. 23: Timed Exposure with Rising Edge Triggering

For more information about setting the exposure time, see Section 6.3.3.4 on page 90.

Trigger Width Exposure Mode

When trigger width exposure mode is selected, the length of the exposure for each frame acquisition will be directly controlled by the ExFSTrig signal. If the camera is set for rising edge triggering, the exposure time begins when the ExFSTrig signal rises and continues until the ExFSTrig signal falls. If the camera is set for falling edge triggering, the exposure time begins when the ExFSTrig signal falls and continues until the ExFSTrig signal rises. Figure 24 illustrates trigger width exposure with the camera set for rising edge triggering.

Trigger width exposure is especially useful if you intend to vary the length of the exposure time for each acquired frame.

If you want to set exposure times that are longer than the maximum possible exposure time settings you can use the trigger width mode.

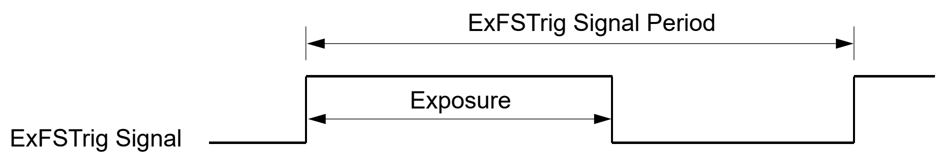


Fig. 24: Trigger Width Exposure with Rising Edge Triggering

When you operate the camera in trigger width exposure mode, you must also set the camera's Exposure Time Abs or Exposure Time Raw parameter. This parameter setting will be used by the camera to operate the Frame Trigger Wait signal.

You should set the Exposure Time Abs or Exposure Time Raw parameter value to represent the shortest exposure time you intend to use. For example, assume that you will be using trigger width exposure mode and that you intend to use the ExFSTrig signal to vary the exposure time in a range from 3000 µs to 5500 µs. In this case you would set e.g. the camera's Exposure Time Abs parameter to 3000 µs.

6.3.3.3 Frame Start Trigger Delay

The frame start trigger delay feature lets you specify a delay (in microseconds) that will be applied between the receipt of a hardware frame start trigger and when the trigger will become effective.

The frame start trigger delay can be specified in the range from 0 to 1000000 µs (equivalent to 1 s). When the delay is set to 0 µs, no delay will be applied.

To set the frame start trigger delay:

- Set the camera's Trigger Selector parameter to frame start.
- Set the value of the Trigger Delay Abs parameter.

i	The frame start trigger delay will not operate, if the Frame Start Trigger Mode parameter is set to off or if you are using a software frame start trigger.

6.3.3.4 Setting the Parameters Related to Hardware Frame Start Triggering and Applying a Hardware Trigger Signal

Setting the Parameters Using Basler pylon and Applying the Signal

You can set all of the parameters needed to perform hardware frame start triggering from within your application by using the pylon API. The following code snippet illustrates using the pylon API to set the parameter values and execute the commands related to hardware frame start triggering with the camera set for the timed exposure mode with rising edge triggering. In this example, the trigger mode for the acquisition start trigger will be set to off:

```
// Select the acquisition start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_AcquisitionStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_Off );
// Disable the acquisition frame rate parameter (this will disable the camera's
// internal frame rate control and allow you to control the frame rate with
// external frame start trigger signals)
Camera.AcquisitionFrameRateEnable.SetValue( false );
// Select the frame start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_FrameStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_On );
// Set the source for the selected trigger
Camera.TriggerSource.SetValue ( TriggerSource_CC1 );
// Set the trigger activation mode to rising edge
Camera.TriggerActivation.SetValue( TriggerActivation_RisingEdge );
// Set the trigger delay for one millisecond (1000us == 1ms == 0.001s)
double TriggerDelay_us = 1000.0;
Camera.TriggerDelayAbs.SetValue( TriggerDelay_us );
// Set for the timed exposure mode
Camera.ExposureMode.SetValue( ExposureMode_Timed );
// Set the exposure time
Camera.ExposureTimeAbs.SetValue( 3000 );

// Frame acquisition will start each time the externally generated
// frame start trigger signal (ExFSTrig signal)goes high
```

The following code snippet illustrates using the pylon API to set the parameter values and execute the commands related to hardware frame start triggering with the camera set for the trigger width exposure mode with rising edge triggering. In this example, the trigger mode for the acquisition start trigger will be set to off:

```
// Select the acquisition start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_AcquisitionStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_Off );
```

```
// Disable the acquisition frame rate parameter (this will disable the camera's
// internal frame rate control and allow you to control the frame rate with
// external frame start signals)
Camera.AcquisitionFrameRateEnable.SetValue( false );
// Select the frame start trigger
Camera.TriggerSelector.SetValue( TriggerSelector_FrameStart );
// Set the mode for the selected trigger
Camera.TriggerMode.SetValue( TriggerMode_On );
// Set the source for the selected trigger
Camera.TriggerSource.SetValue ( TriggerSource_CC1 );
// Set the trigger activation mode to rising edge
Camera.TriggerActivation.SetValue( TriggerActivation_RisingEdge );
// Set for the trigger width exposure mode
Camera.ExposureMode.SetValue( ExposureMode_TriggerWidth );

// Frame acquisition will start each time the externally generated
// frame start trigger signal (ExFSTrig signal)goes high
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Parameters Using Direct Register Access and Applying the Signal

To set the parameters needed to perform hardware frame start triggering via direct register access (with the trigger mode for the acquisition start trigger set to off):

- Set the value of the Trigger Mode Acquisition Start register to off.
- Set the value of the Acquisition Frame Period Enable register to false.
  (This will disable the camera's ability to control the frame period internally and allow you to control the frame rate with an external signal.)
- Set the value of the Trigger Mode Frame Start register to On.
- Set the value of the Trigger Source Frame Start register to receive the external trigger signal on CC1, CC2, or CC4, or on input line 1 or input line 2.
- Set the value of the Trigger Activation Frame Start register to rising edge or falling edge as desired.
- Set the value of the Exposure Mode register to timed or to trigger width.

  If the mode is set to timed, set the value of the Exposure Time Raw register as desired.

  A value in a raw register is simply an integer value with no units. To determine what the actual setting will be, you must multiply the value in the raw register by the camera's time base. The time base on aviator cameras is 1 µs.

  For example, if you set the Exposure Time Raw register to 1000, the exposure time would be 1000 µs (1000 x 1 µs = 1000 µs).

Apply the appropriate externally generated electrical signal (ExFSTrig signal) to the selected trigger source.

For more information about changing settings via direct register access, see Section 5 on page 61.

6.4 Setting the Exposure Time

This section (Section 6.4) describes how the exposure time can be adjusted "manually", i.e., by setting the value of the exposure time parameter.

The camera also has an Exposure Auto function that can automatically adjust the exposure time. **Manual adjustment of the exposure time parameter will only work correctly if the Exposure Auto function is disabled.**

For more information about

- auto functions in general, see Section 9.8 on page 173.
- the Exposure Auto function in particular, see Section 9.8.6 on page 185.

Note that on cameras delivered from the factory, the exposure auto function will become enabled whenever you power on or reset the camera.

This behavior happens due to the way that the camera's configuration sets are set in the factory. More specifically, it happens because the auto functions factory setup is selected as the default configuration set and the default configuration set is designated as the "startup" set. To change this behavior, you must change the configuration set settings. For more information about configuration sets, see Section 9.19 on page 256.

If you are operating the camera in any one of the following ways, you must use the camera's Exposure Time parameter to set the exposure time:

- the frame start trigger mode is set to off
- the frame start trigger mode is set to on and the trigger source is set to software
- the frame start trigger mode is set to on, the trigger source is set to line 1, line 2, CC1, CC2, or CC4, and the exposure mode is set to timed

The exposure time must not be set below a minimum specified value. The minimum exposure time for each camera model is shown in Table 7.

The maximum possible exposure time that can be set is also shown in Table 7.

Camera Model	Minimum Allowed Exposure Time	Maximum Possible Exposure Time
avA1000-120km/kc	12 µs	2500000 µs
avA1600-65km/kc	12 µs	2500000 µs
avA1900-60km/kc	12 µs	2500000 µs
avA2300-30km/kc	18 µs	2500000 µs

Table 7: Minimum Allowed Exposure Time and Maximum Possible Exposure Time

Setting the Exposure Time Using Basler pylon

If you are parameterizing the camera with Basler pylon, the exposure time is determined by the setting of the Exposure Time Abs parameter. The Exposure Time Abs parameter sets the exposure time in µs. The exposure time should be set in increments of 1 µs.

You can use the pylon API to set the Exposure Time Abs parameter value from within your application software. The following code snippet illustrates using the pylon API to set the parameter value:

```
// Set the exposure time to 1000 µs
Camera.ExposureTimeAbs.SetValue( 1000 );
```

You can also use the Basler pylon Viewer application to easily set the parameter.

If you want to set exposure times that are longer than the maximum possible exposure time settings indicated in Table 7 you can use the trigger width mode.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Exposure Time Using Direct Register Access

If you are parameterizing the camera via direct register access, the Exposure Time Raw register sets the exposure time.

To set the exposure time via direct register access:

- Set the value of the Exposure Time Raw register.

A value in a raw register is simply an integer value with no units. To determine what the actual exposure time will be, you must multiply the value in the raw register by the camera's time base. The time base on aviator cameras is 1 µs.

For example, if you set the Exposure Time Raw register to 1000, the exposure time would be 1000 µs (1000 x 1 µs = 1000 µs).

For more information about changing settings via direct register access, see Section 5 on page 61.

6.5 Overlapping Exposure with Sensor Readout

The frame acquisition process on the camera includes two distinct parts. The first part is the exposure of the pixels in the imaging sensor. Once exposure is complete, the second part of the process – readout of the pixel values from the sensor – takes place. In regard to this frame acquisition process, there are two common ways for the camera to operate: with “non-overlapped” exposure and with “overlapped” exposure.

In the **non-overlapped mode** of operation, each time a frame is acquired the camera completes the entire exposure/readout process before acquisition of the next frame is started. The exposure for a new frame does not overlap the sensor readout for the previous frame. This situation is illustrated in Figure 25 with the camera set for the trigger width exposure mode.

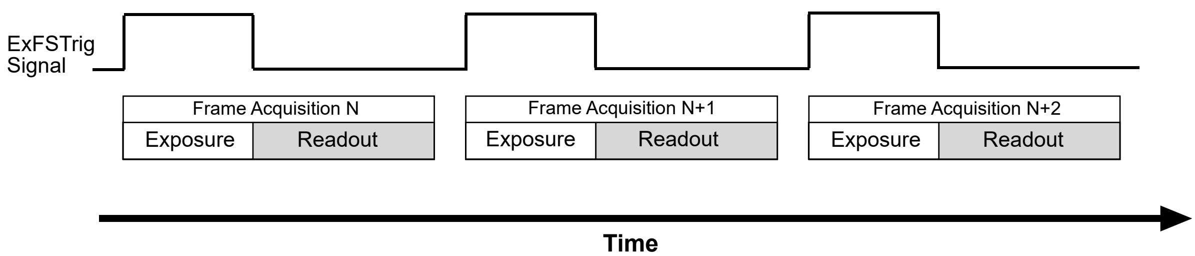


Fig. 25: Non-overlapped Exposure and Readout

In the **overlapped mode** of operation, the exposure of a new frame begins while the camera is still reading out the sensor data for the previously acquire frame. This situation is illustrated in Figure 26 with the camera set for the trigger width exposure mode.

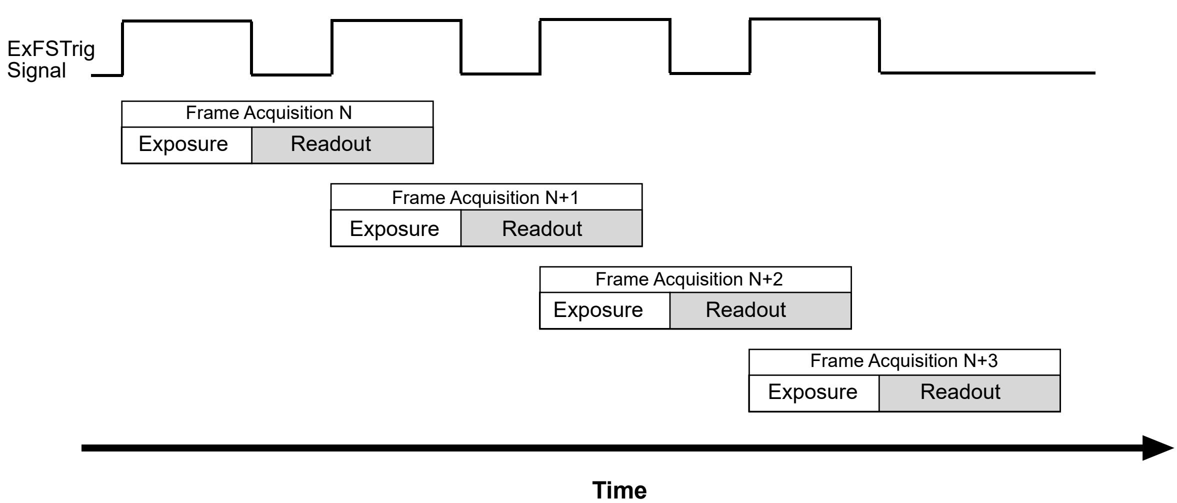


Fig. 26: Overlapped Exposure and Readout

Determining whether your camera is operating with overlapped or non-overlapped exposure and readout is not a matter of issuing a command or switching a setting on or off. Rather the way that you operate the camera will determine whether the exposures and readouts are overlapped or not. If we define the “frame period” as the time from the start of exposure for one frame acquisition to the start of exposure for the next frame acquisition, then:

- Exposure will not overlap when: Frame Period > Exposure Time + Readout Time
- Exposure will overlap when: Frame Period ≤ Exposure Time + Readout Time

You can determine the readout time by reading the value of the Readout Time Abs parameter. The parameter indicates what the readout time will be in microseconds given the camera’s current settings. You can read the Readout Time Abs parameter value from within your application software by using the Basler pylon API. The following code snippet illustrates using the API to get the parameter value:

```
double ReadoutTime = Camera.ReadoutTimeAbs.GetValue( );
```

You can also use the Basler pylon Viewer application to easily get the parameter value.

For more information about determining the frame readout time, see Section 6.7 on page 106.

Guideline for Overlapped Operation with Trigger Width Exposure

If the camera is set for the trigger width exposure mode and you are operating the camera in a way that readout and exposure will be overlapped, keep in mind:

You must not end the exposure time of the current frame acquisition until readout of the previously acquired frame is complete.

If this guideline is violated, the camera will drop the frame for which the exposure was just ended and will declare an over trigger error. This situation is illustrated in Figure 27 with the camera set for the trigger width exposure mode with rising edge triggering.

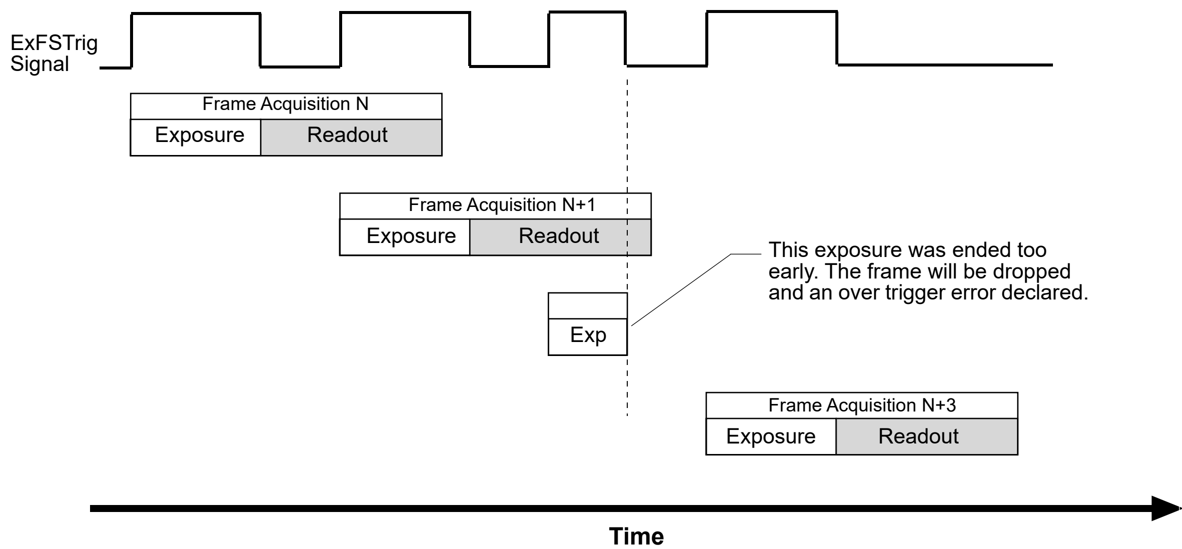


Fig. 27: Overtriggering Caused by an Early End of Exposure

You can avoid violating this guideline by using the camera's Frame Trigger Wait signal to determine when exposure can safely begin and by properly setting the camera's Exposure Time Abs or Exposure Time Raw parameter.

For more information about

- the Frame Trigger Wait signal and the Exposure Time Abs or Exposure Time Raw parameter, see Section 6.6.3.2 on page 102.
- trigger width exposure, see Section 6.3.3.2 on page 87.
- over trigger error, see Section 9.10 on page 194.

6.6 Acquisition Monitoring Tools

6.6.1 The Exposure Active Signal

The camera can provide an "exposure active" (ExpAc) output signal. The signal goes high when the exposure time for each frame acquisition begins and goes low when the exposure time ends as shown in Figure 28. This signal can be used as a flash trigger and is also useful when you are operating a system where either the camera or the object being imaged is movable. For example, assume that the camera is mounted on an arm mechanism and that the mechanism can move the camera to view different portions of a product assembly. Typically, you do not want the camera to move during exposure. In this case, you can monitor the ExpAc signal to know when exposure is taking place and thus know when to avoid moving the camera.

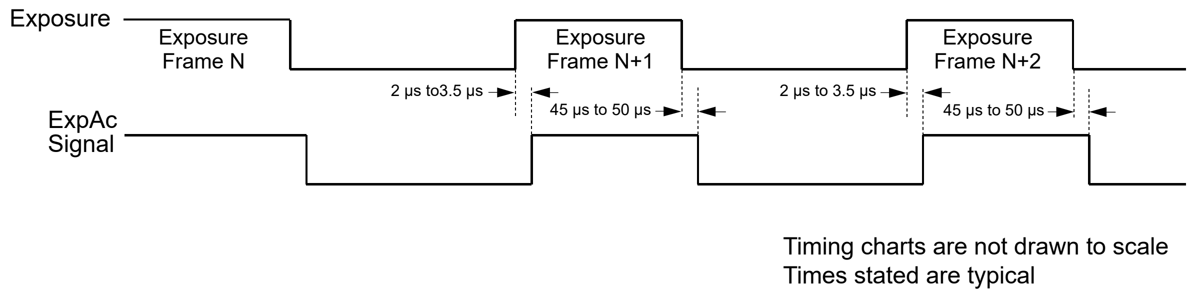


Fig. 28: Exposure Active Signal

When you use the exposure active signal, be aware that there is a delay in the rise and the fall of the signal in relation to the start and the end of exposure. See Figure 28 for details.

The exposure active output signal can be assigned to camera output line 1.

For more information about

- changing which camera output signal is assigned to output line 1, see Section 3.2 on page 43.
- the electrical characteristics of the camera's output line, see Section 2.3.4 on page 29.

6.6.2 Acquisition Status Indicator

If a camera receives a software acquisition start trigger signal when it is not in a "waiting for acquisition start trigger" acquisition status, it will simply ignore the trigger signal and will generate an acquisition start overtrigger event.

If a camera receives a software frame start trigger signal when it is not in a "waiting for frame start trigger" acquisition status, it will simply ignore the trigger signal and will generate a frame start overtrigger event.

The camera's acquisition status indicator gives you the ability to check whether the camera is in a "waiting for acquisition start trigger" acquisition status or in a "waiting for frame start trigger" acquisition status. If you check the acquisition status before you apply each software acquisition start trigger signal or each software frame start trigger signal, you can avoid applying trigger signals to the camera that will be ignored.

The acquisition status indicator is designed for use when you are using host control of image acquisition, i.e., when you are using software acquisition start and frame start trigger signals.

For more information about the overtrigger error, see Section 9.10.2 on page 195.

Checking the Acquisition Status Using Basler pylon

To determine the acquisition status of the camera via the Basler pylon API:

- Use the Acquisition Status Selector to select the Acquisition Trigger Wait status or the Frame Trigger Wait status.
- Read the value of the Acquisition Status parameter.

  If the value is set to "false", the camera is not waiting for the trigger signal.

  If the value is set to "true", the camera is waiting for the trigger signal.

You can check the acquisition status from within your application software by using the Basler pylon API. The following code snippet illustrates using the API to check the acquisition status:

```
// Check the acquisition start trigger acquisition status
// Set the acquisition status selector
Camera.AcquisitionStatusSelector.SetValue
( AcquisitionStatusSelector_AcquisitionTriggerWait );
// Read the acquisition status
bool IsWaitingForAcquisitionTrigger = Camera.AcquisitionStatus.GetValue();

// Check the frame start trigger acquisition status
// Set the acquisition status selector
Camera.AcquisitionStatusSelector.SetValue
( AcquisitionStatusSelector_FrameTriggerWait );
// Read the acquisition status
bool IsWaitingForFrameTrigger = Camera.AcquisitionStatus.GetValue();
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Checking the Acquisition Status Using Direct Register Access

To determine the acquisition start trigger status via the direct register access:

- Read the value of the Status Acquisition Trigger Wait register.
  If the value is set to 0, the camera is not waiting for the trigger signal.
  If the value is set to 1, the camera is waiting for the trigger signal.

To determine the frame start trigger status via the direct register access:

- Read the value of the Status Frame Trigger Wait register.
  If the value is set to 0, the camera is not waiting for the trigger signal.
  If the value is set to 1, the camera is waiting for the trigger signal.

For more information about changing settings via direct register access, see Section 5 on page 61.

6.6.3 Trigger Wait Signals

If a camera receives a hardware acquisition start trigger signal when it is not in a "waiting for acquisition start trigger" acquisition status, it will simply ignore the trigger signal and will generate an acquisition start overtrigger event.

If a camera receives a hardware frame start trigger signal when it is not in a "waiting for frame start trigger" acquisition status, it will simply ignore the trigger signal and will generate a frame start overtrigger event.

The camera's acquisition trigger wait signal gives you the ability to check whether the camera is in a "waiting for acquisition start trigger" acquisition status. If you check the acquisition trigger wait signal before you apply each hardware acquisition start trigger signal, you can avoid applying acquisition start trigger signals to the camera that will be ignored.

The camera's frame trigger wait signal gives you the ability to check whether the camera is in a "waiting for frame start trigger" acquisition status. If you check the frame trigger wait signal before you apply each hardware frame start trigger signal, you can avoid applying frame start trigger signals to the camera that will be ignored.

These signals are designed to be used when you are triggering acquisition start or frame start via a hardware trigger signal.

For more information about the overtrigger error, see Section 9.10.2 on page 195.

6.6.3.1 Acquisition Trigger Wait Signal

As you are acquiring frames, the camera automatically monitors the acquisition start trigger status and supplies a signal that indicates the current status. The Acquisition Trigger Wait signal will go high whenever the camera enters a "waiting for acquisition start trigger" status. The signal will go low when an external acquisition start trigger (ExASTrig) signal is applied to the camera and the camera exits the "waiting for acquisition start trigger status". The signal will go high again when the camera again enters a "waiting for acquisition trigger" status and it is safe to apply the next acquisition start trigger signal.

If you base your use of the ExASTrig signal on the state of the acquisition trigger wait signal, you can avoid "acquisition start overtriggering", i.e., applying an acquisition start trigger signal to the camera when it is not in a "waiting for acquisition start trigger" acquisition status. If you do apply an acquisition start trigger signal to the camera when it is not ready to receive the signal, it will be ignored and an acquisition start overtrigger event will be reported.

Figure 29 illustrates the Acquisition Trigger Wait signal with the Acquisition Frame Count parameter set to 3 and with exposure and readout overlapped. The figure assumes that the trigger mode for the frame start trigger is set to off, so the camera is internally generating frame start trigger signals.

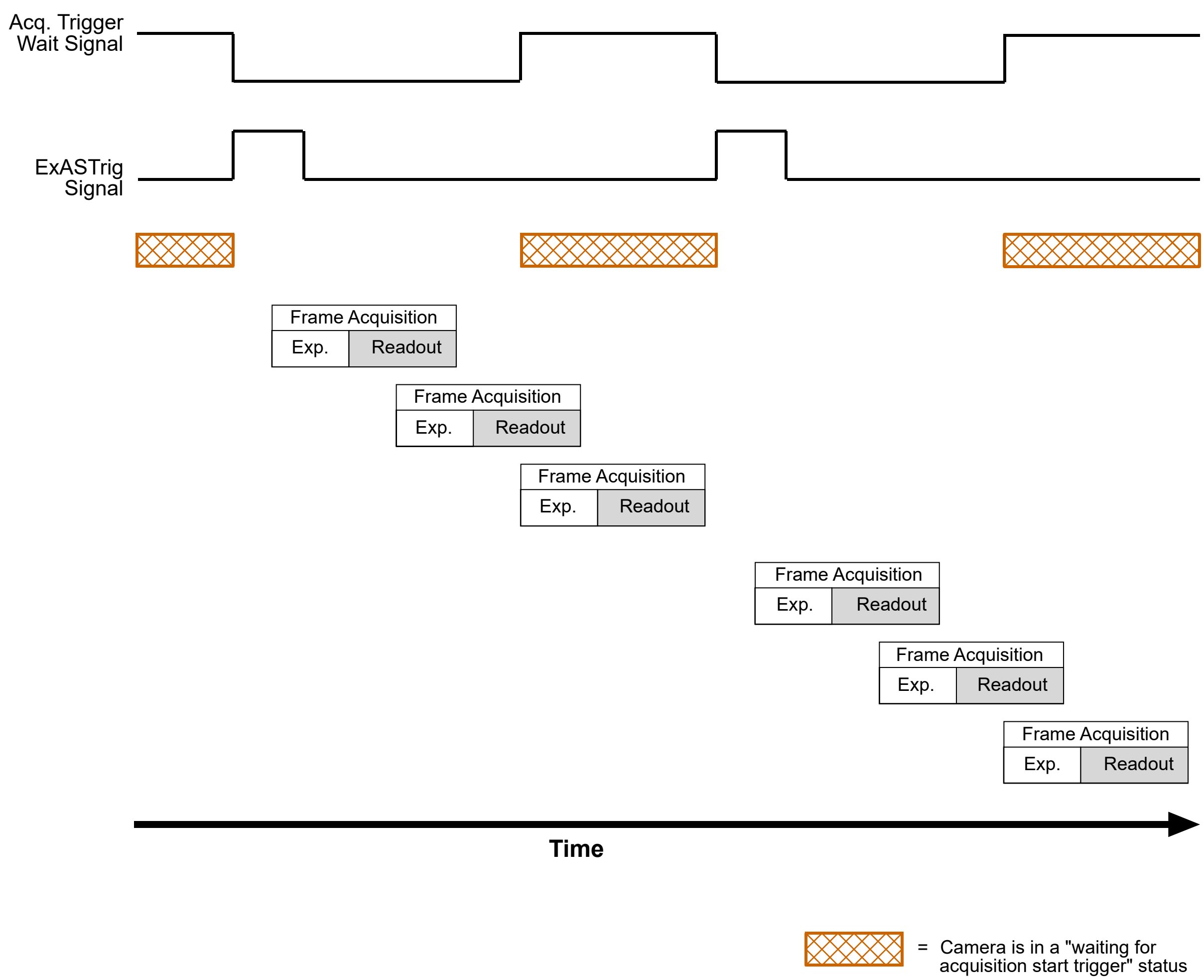


Fig. 29: Acquisition Trigger Wait Signal

i	The acquisition trigger wait signal will only be available when hardware acquisition start triggering is enabled.

For more information about the overtrigger error, see Section 9.10.2 on page 195.

Selecting the Acquisition Trigger Wait Signal as the Source Signal for the Output Line Using Basler pylon

The acquisition trigger wait signal can be selected to act as the source signal for camera output line 1. Selecting a source signal for an output line is a two step process:

- Use the Line Selector to select output line 1.
- Set the value of the Line Source Parameter to the acquisition trigger wait signal.

You can set the Line Selector and the Line Source parameter value from within your application software by using the Basler pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
Camera.LineSelector.SetValue( LineSelector_Out1 );
Camera.LineSource.SetValue( LineSource_AcquisitionTriggerWait );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Selecting the Acquisition Trigger Wait Signal as the Source Signal for an Output Line Using Direct Register Access

To select the acquisition trigger wait signal as the source signal for camera output line 1 via direct register access:

- Set the value of the Line Source Line 5 register to Acquisition Trigger Wait.

For more information about changing settings via direct register access, see Section 5 on page 61.

6.6.3.2 Frame Trigger Wait Signal

Overview

As you are acquiring frames, the camera automatically monitors the frame start trigger status and supplies a signal that indicates the current status. The Frame Trigger Wait signal will go high whenever the camera enters a "waiting for frame start trigger" status. The signal will go low when an external frame start trigger (ExFSTrig) signal is applied to the camera and the camera exits the "waiting for frame start trigger status". The signal will go high again when the camera again enters a "waiting for frame trigger" status and it is safe to apply the next frame start trigger signal.

If you base your use of the ExFSTrig signal on the state of the frame trigger wait signal, you can avoid "frame start overtriggering", i.e., applying a frame start trigger signal to the camera when it is not in a "waiting for frame start trigger" acquisition status. If you do apply a frame start trigger signal to the camera when it is not ready to receive the signal, it will be ignored and a frame start overtrigger event will be reported.

Figure 30 illustrates the Frame Trigger Wait signal. The camera is set for the trigger width exposure mode with rising edge triggering and with exposure and readout overlapped.

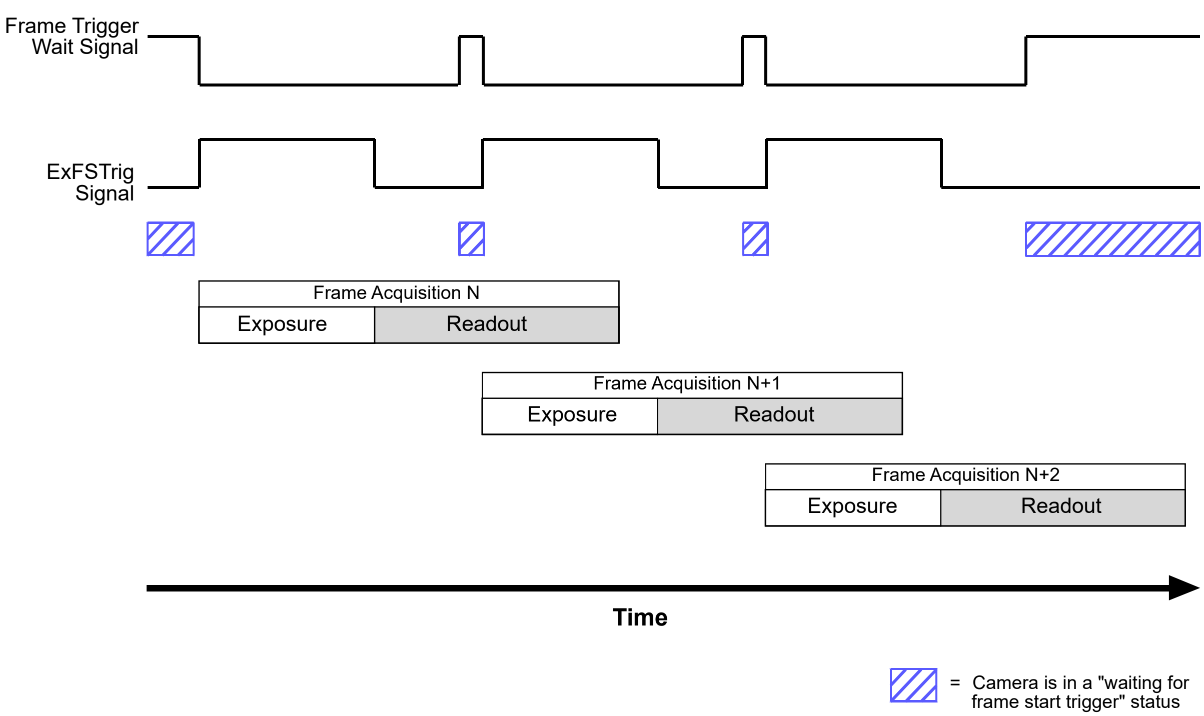


Fig. 30: Frame Trigger Wait Signal

i	The frame trigger wait signal will only be available when hardware frame start triggering is enabled.

For more information about the overtrigger error, see Section 9.10.2 on page 195.

Frame Trigger Wait Signal Details

When the camera is set for the timed exposure mode, the rise of the Frame Trigger Wait signal is based on the current Exposure Time Abs parameter setting and on when readout of the current frame will end. This functionality is illustrated in Figure 31.

If you are operating the camera in the timed exposure mode, you can avoid overtriggering by always making sure that the Frame Trigger Wait signal is high before you trigger the start of frame capture.

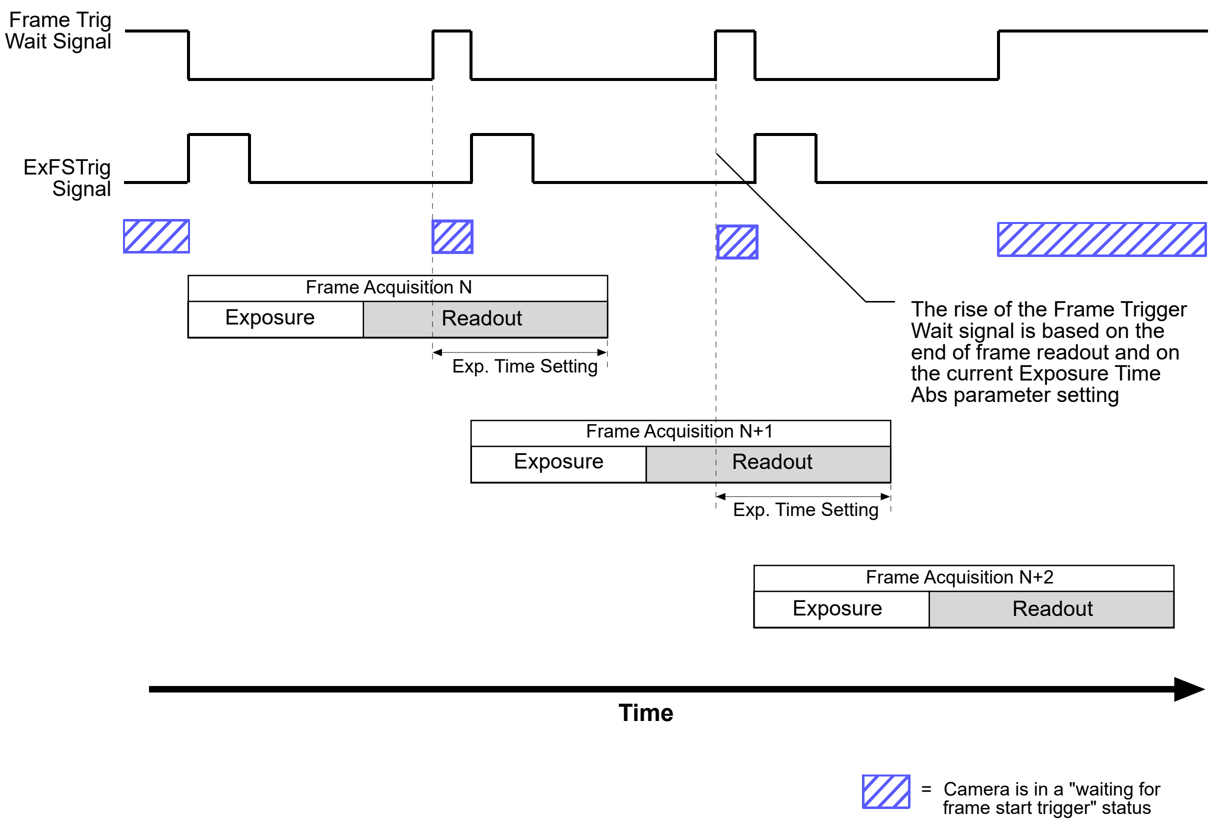


Fig. 31: Frame Trigger Wait Signal with the Timed Exposure Mode

When the camera is set for the trigger width exposure mode, the rise of the Frame Trigger Wait signal is based on the Exposure Time Abs or Exposure Time Raw parameter setting and on when readout of the current frame will end. This functionality is illustrated in Figure 32.

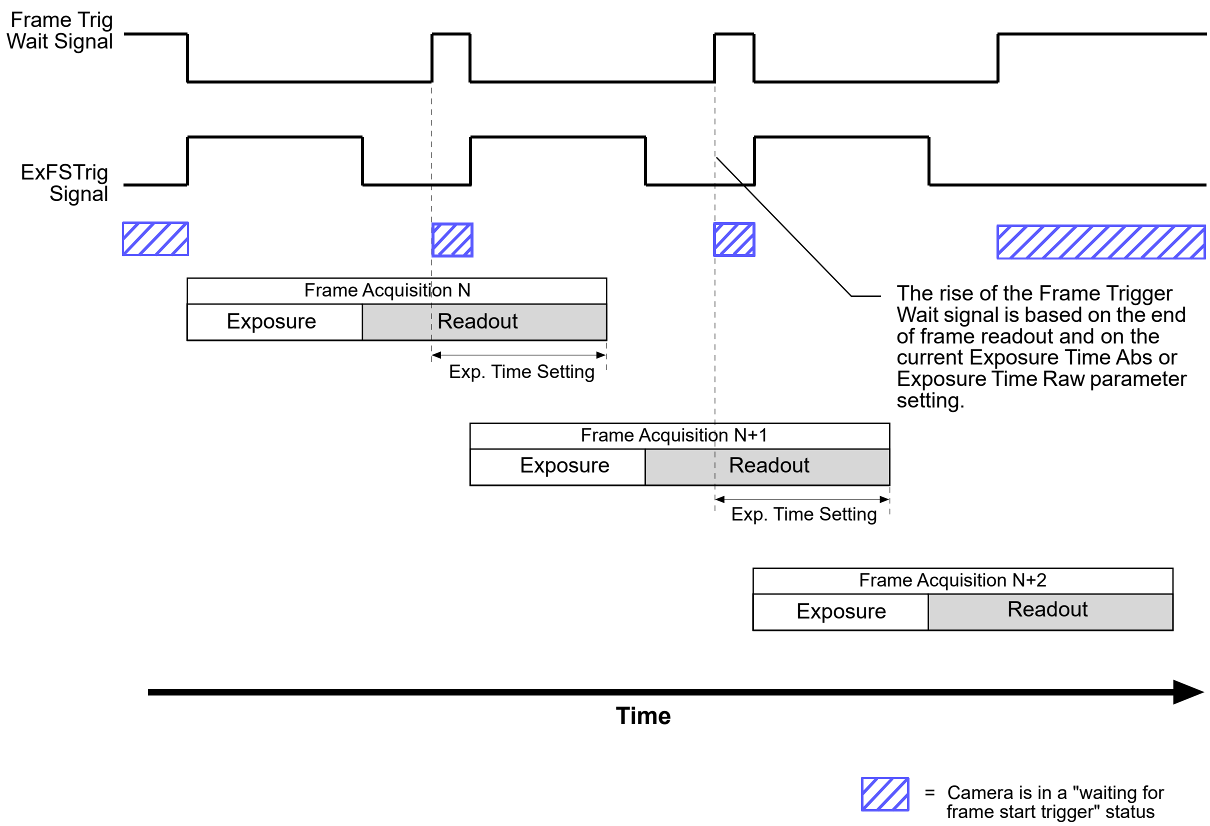


Fig. 32: Frame Trigger Wait Signal with the Trigger Width Exposure Mode

If you are operating the camera in the trigger width exposure mode, you can avoid overtriggering the camera by always doing the following:

- Setting the camera's Exposure Time Abs or Exposure Time Raw parameter so that it represents the smallest exposure time you intend to use.
- Making sure that your exposure time is always equal to or greater than the setting for the Exposure Time Abs or Exposure Time Raw parameter.
- Monitoring the camera's Frame Trigger Wait signal and only using the ExFSTrig signal to start exposure when the Frame Trigger Wait signal is high.

You should set the Exposure Time Abs or Exposure Time Raw parameter value to represent the shortest exposure time you intend to use. For example, assume that you will be using trigger width exposure mode and that you intend to use the ExFSTrig signal to vary the exposure time in a range from 3000 µs to 5500 µs. In this case you would e.g. set the camera's Exposure Time Abs parameter to 3000 µs.

For information about setting the Exposure Time parameters, see Section 6.4 on page 92.

Selecting the Frame Trigger Wait Signal as the Source Signal for the Output Line Using Basler pylon

The frame trigger wait signal can be selected to act as the source signal for camera output line 1. Selecting a source signal for the output line is a two step process:

- Use the Line Selector to select output line 1.
- Set the value of the Line Source Parameter to the frame trigger wait signal.

You can set the Line Selector and the Line Source parameter value from within your application software by using the Basler pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
Camera.LineSelector.SetValue( LineSelector_Out1 );
Camera.LineSource.SetValue( LineSource_FrameTriggerWait );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Selecting the Frame Trigger Wait Signal as the Source Signal for an Output Line Using Direct Register Access

To select the frame trigger wait signal as the source signal for camera output line 1 via direct register access:

Set the value of the Line Source Line 5 register to Frame Trigger Wait.

For more information about changing settings via direct register access, see Section 5 on page 61.

6.7 Acquisition Timing Charts

Timing charts for frame acquisition and transmission appear at the end of this section. The charts assume that exposure is triggered by an externally generated frame start (an ExFSTrig) signal with rising edge activation and that the camera is set for the timed exposure mode.

As the timing charts show, there is a slight delay between the rise of the ExFSTrig signal and the start of exposure. After the exposure time for an image acquisition is complete, the camera begins reading out the acquired image data from the imaging sensor into a buffer in the camera. This buffering technique avoids the need to exactly synchronize the clock used for sensor readout with the data transmission over the Camera Link connection and is an important element in achieving the highest possible frame rate with the best image quality.

The charts also show that there is a delay between the point where the camera begins reading image data out of the imaging sensor and when it starts transmitting the data from the image buffer to the host computer. The duration of this delay varies significantly depending on whether the camera is set for the 1X-1Y, the 1X2-1Y or the 1X-2YE Camera Link tap geometry.
When set for

- the 1X2-1Y or the 1X-1Y geometry, the camera must read out and buffer the entire image before it begins transmitting image data.
- the 1X-2YE geometry, the camera will begin transmitting image data before it has read out and buffered the entire image.

This difference in the starting point of image data transmission is the main difference between the two timing charts.

In the timing charts:

- The **exposure start delay** is the amount of time between the point where the frame start trigger signal transitions and the point where exposure actually begins.
- The **frame readout time** is the amount of time it takes to read out the data for an acquired image from the sensor into the image buffer.
- The **transmission start delay** is the amount of time between the point where the camera begins reading image data out of the sensor to the point where the camera begins transmitting the data from the image buffer to the host computer.
- The **frame transmission time** is the amount of time it takes to transmit the acquired image from the buffer in the camera to the host computer via the Camera Link interface.

The table below shows the exposure start delay for each camera model:

Camera Model	Exposure Start Delay
avA1000-120km/kc	42 µs
avA1600-65km/kc	57 µs
avA1900-60km/kc	64 µs
avA2300-30km/kc	79 µs

Table 8: Exposure Start Delays

As shown in the timing charts, the exposure of a new image can start while the camera is reading a previously acquired image out of the sensor. However, **the exposure time for the new image must not end while the readout of the previous image is still in progress**. The best way to avoid this problem is to determine the maximum allowed frame rate with the camera's current settings and to simply make sure that you do not attempt to capture images at a rate that is higher than the allowed maximum.

For more information about

- determining the maximum allowed frame rate with the current camera settings, see Section 6.8 on page 112.
- frame buffering, see Section 4 on page 54.
- the 1X-1Y, 1X2-1Y, and 1X-2YE tap geometries, see Section 8.2 on page 143.

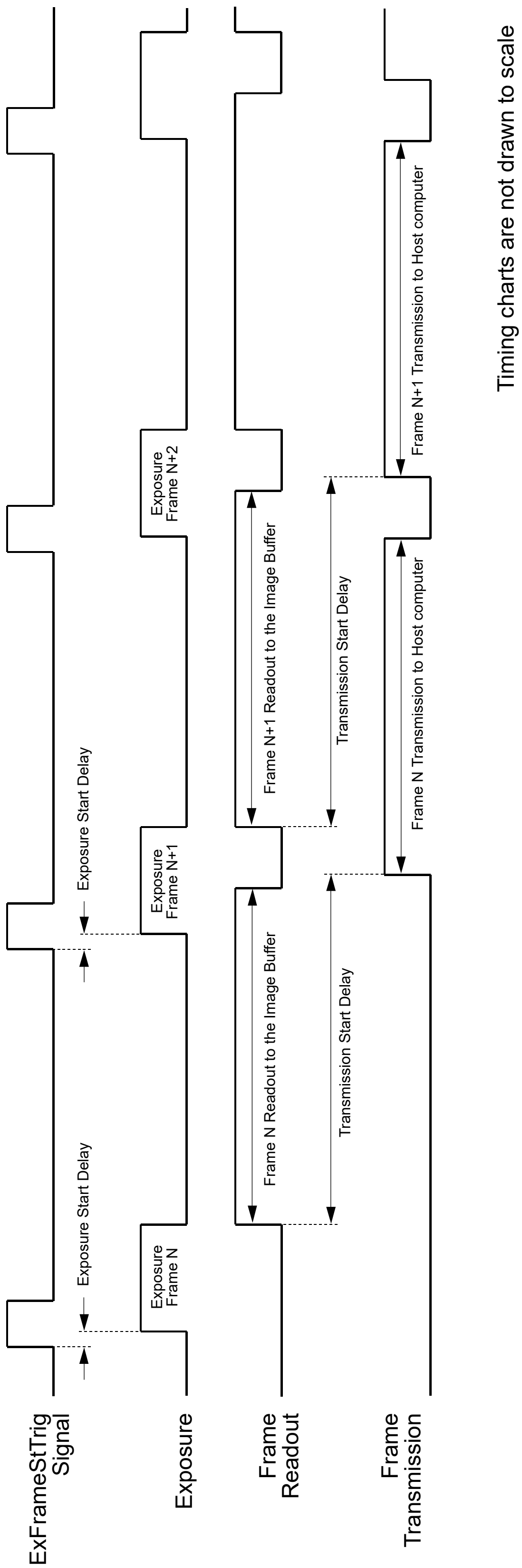


Fig. 33: Acquisition Timing Chart with 1X-!Y or 1X2-1Y Camera Link Tap Geometry

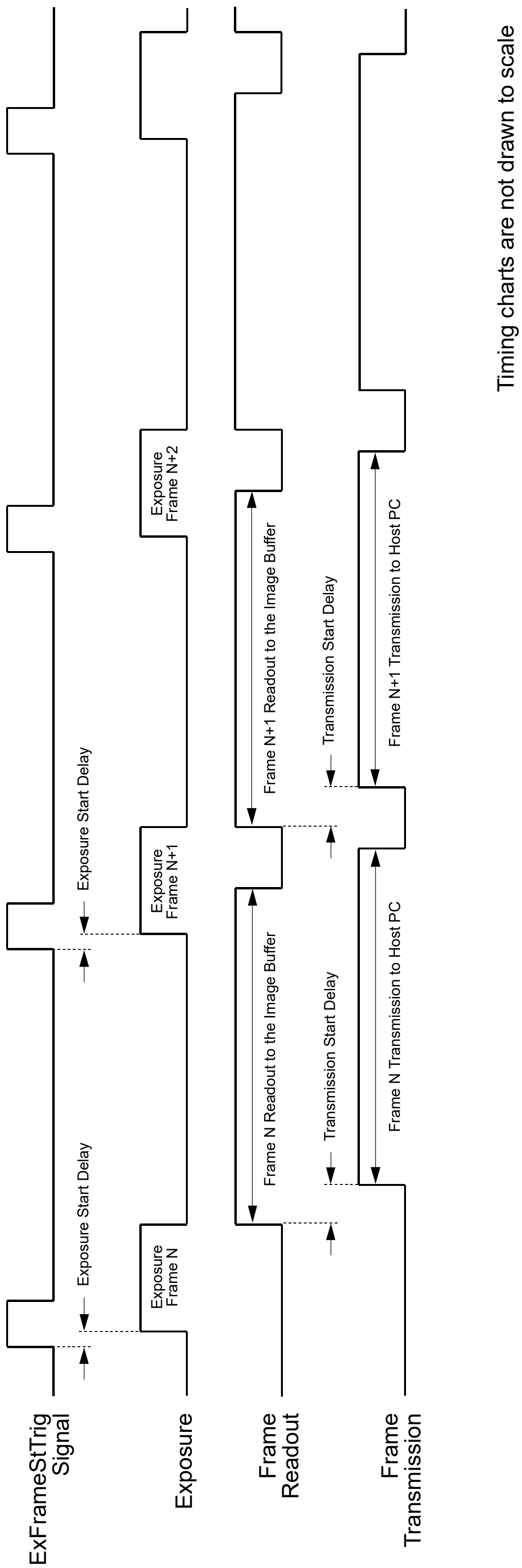


Fig. 34: Acquisition Timing Chart with 1X-2YE Camera Link Tap Geometry

Frame Readout Time

You can calculate a close approximation of the frame readout time by using this formula:

$$\text{Frame Readout Time} = \frac{1}{\text{Maximum Allowed Frame Rate}}$$

You can determine the readout time by reading the value of the Readout Time Abs parameter. The parameter indicates what the readout time will be in microseconds given the camera's current settings. You can read the Readout Time Abs parameter value from within your application software by using the Basler pylon API. The following code snippet illustrates using the API to get the parameter value:

```
double ReadoutTime = Camera.ReadoutTimeAbs.GetValue( );
```

You can also use the Basler pylon Viewer application to easily get the parameter value.

For more information about determining the maximum allowed frame rate given the current camera settings, see Section 6.8 on page 112.

Transmission Start Delay

As shown in the table below, the transmission start delay depends on the camera model and on the Camera Link Tap Geometry parameter setting.

Camera Link Tap Geometry Setting	**avA1000-120km/kc** (at 1024 x 1024)	**avA1600-65km/kc** (at 1600 x 1200)	**avA1900-60km/kc** (at 1920 x 1080)	**avA2300-30km/kc** (at 2330 x 1750)
1X-1Y	8.28 ms	14.47 ms	15.30 ms	29.41 ms
1X2-1Y	8.28 ms	14.47 ms	15.30 ms	29.41 ms
1X-2YE	215 µs	581 µs	674 µs	584 µs

Table 9: Transmission Start Delays

For more information about the 1X-1Y, 1X2-1Y and 1X-2YE tap geometries, see Section 8.2 on page 143.

Frame Transmission Time

You can calculate the time that it will take to transmit a frame from the camera to the host computer by using this formula:

Frame Transmission Time = (AOI Width + 16) x AOI Height x C / T
[T = Number of taps; for 1X-1Y -> T =1; for 1X2-1Y and 1x-2YE -> T = 2]

Where: C = 50.000 ns, if the camera is set for a 20 MHz Camera Link clock speed
30.768 ns, if the camera is set for a 32.5 MHz Camera Link clock speed
25.000 ns, if the camera is set for a 40 MHz Camera Link clock speed
20.086 ns, if the camera is set for a 48 MHz Camera Link clock speed
15.384 ns, if the camera is set for a 65 MHz clock speed

For more information about the AOI Height and AOI Width, see Section 9.7 on page 167.

6.8 Maximum Allowed Frame Acquisition Rate

The maximum allowed frame acquisition rate for your camera is not static. It can vary depending on how certain camera features are set. In general, the following factors can affect the maximum allowed frame rate:

- The amount of time it takes to read the data for an acquired image (known as a frame) out of the imaging sensor and into the camera's frame buffer. This time varies depending on the height of the frame. The frame height is determined by the camera's AOI settings. If you use shorter AOIs (i.e., AOIs that include fewer lines), it takes less time to read an acquired frame out of the sensor and you can acquire more frames per second.
- The Camera Link pixel clock setting. If the pixel clock is set to a low value, it will take longer to transfer captured images from the camera to the frame grabber in your host computer. With lower pixel clock speeds, you can acquire fewer frames per second.
- The setting for the prelines feature. If you use a higher prelines setting, you can acquire fewer frames per second.
- The exposure time for acquired frames. If you use very long exposure times, you can acquire fewer frames per second.
- The setting for the sensor digitization taps feature. If this feature is set to four taps, you will be able to acquire frames at a much higher rate than if it is set to one tap.
- The binning feature. If vertical binning is enabled, the maximum allowed frame rate will be increased.

There are several ways that you can determine the maximum allowed acquisition frame rate with your current camera settings:

- You can go to the Support section of the Basler website and use the online frame rate calculator:
  www.baslerweb.com
- You can use Basler pylon to read the value of the camera's Resulting Frame Rate Abs parameter (see below).
- You can use direct register access to read the value of the Resulting Frame Period register (see below).

For more information about AOI settings, see Section 9.7 on page 167.

For more information about selectable pixel clock speeds, see Section 9.2 on page 149.

Using Basler pylon to Check the Maximum Allowed Frame Rate

You can use the following Basler pylon tools to read the current value of the Resulting Frame Rate Abs parameter from within your application software:

- the Basler pylon API
- the Basler pylon Viewer

The following code snippet illustrates using the pylon API to get the parameter value:

```
// Get the resulting frame rate
double resultingFps = Camera.ResultingFrameRateAbs.GetValue();
```

The Resulting Frame Rate Abs parameter takes all camera settings that can influence the frame rate into account and indicates the maximum allowed frame rate given the current settings.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Using Direct Register Access to Check the Maximum Allowed Rate

When using direct register access, you work with the "resulting frame period" rather than the frame acquisition rate. Once the resulting frame period is known, the maximum allowed frame acquisition rate can be determined by:

$$\text{Max Frame Acquisition Rate} = \frac{1}{\text{Resulting Frame Period in Seconds}}$$

To determine the resulting frame period:

- Check the value of the Resulting Frame Period Raw register.

The Resulting Frame Period Raw register takes all of camera settings that can influence the frame period into account and indicates the minimum allowed frame period given the current settings.

A value in a raw register is simply an integer value with no units. To determine what the actual frame period will be, you must multiply the value in the raw register by the camera's time base. The time base on aviator cameras is 1 µs.

For example, if you read the Resulting Frame Period Raw register and find that its value is 10000, the resulting frame period would be 10000 µs (10000 x 1 µs = 10000 µs).

6.8.1 Increasing the Maximum Allowed Frame Rate

You may find that you would like to acquire frames at a rate higher than the maximum allowed with the camera's current settings. In this case, you must adjust one or more of the factors that can influence the maximum allowed rate and then check to see if the maximum allowed rate has increased:

- If you have the sensor digitization feature set to one, consider changing the value to four. This usually will greatly increase the maximum allowed frame rate.
- Decreasing the height of the AOI can have a significant impact on the maximum allowed frame rate. If possible in your application, decrease the height of the AOI.
- If you have the Camera Link pixel clock speed on your camera set to a low value, consider setting it to a higher value. Be aware, however, that some frame grabbers cannot handle

higher pixel clock speeds. Before you increase the setting for the camera's pixel clock, make sure that your frame grabber is compatible with the higher setting.

- The setting for the prelines feature can have an impact on the maximum allowed frame rate. Lowering the setting for the prelines feature can increase the maximum allowed frame rate. However, lowering the setting can also have a negative impact on image quality, especially if your AOI height is small. So changing the prelines setting may involve a trade-off between increasing the maximum allowed frame rate and lowering image quality.
- If you are using normal exposure times and you are using the camera at it's maximum resolution, your exposure time will not normally restrict the frame rate. However, if you are using long exposure times or a small area of interest, it is possible that your exposure time is limiting the maximum allowed frame rate. If you are using a long exposure time or a small AOI, try using a shorter exposure time and see if the maximum allowed frame rate increases. (You may need to compensate for a shorter exposure time by using a brighter light source or increasing the opening of your lens aperture.)
- Using vertical binning will increase the maximum allowed frame rate. However, using binning also decreases the sensor's effective resolution.

> An important thing to keep in mind is a common mistake new camera users frequently make when they are working with exposure time. They will often use a very long exposure time without realizing that this can severely limit the camera's maximum allowed frame rate. As an example, assume that your camera is set to use a 1/2 second exposure time. In this case, because each frame acquisition will take at least 1/2 second to be completed, the camera will only be able to acquire a maximum of two frames per second. Even if the camera's nominal maximum frame rate is, for example, 100 frames per second, it will only be able to acquire two frames per second because the exposure time is set much higher than normal.

For more information about

- the sensor digitization taps feature, see Section 9.1 on page 148 and Section 4.1 on page 54.
- the AOI settings, see Section 9.7 on page 167.
- the prelines feature, see Section 9.7.2 on page 171.
- selectable pixel clock speeds, see Section 9.2 on page 149.
- the exposure time, see Section 6.4 on page 92.
- binning, see Section 9.12 on page 233.

6.9 Use Case Descriptions and Diagrams

The following pages contain a series of use case descriptions and diagrams. The descriptions and diagrams are designed to illustrate how acquisition start triggering and frame start triggering work in some common situations and with some common combinations of parameter settings.

These use cases do not represent every possible combination of the parameters associated with acquisition start and frame start triggering. They are simply intended to aid you in developing an initial understanding of how these two triggers interact.

In each use case diagram, the black box in the upper left corner indicates how the parameters are set.

i	The use case diagrams are representational. They are not drawn to scale and are not designed to accurately describe precise camera timings.

Use Case 1 - Acquisition and Frame Start Triggers Both Off (Free Run)

Use case one is illustrated on page 116.

In this use case, the Trigger Mode parameter for the acquisition start trigger and the Trigger Mode parameter for the frame start trigger are both set to off. The camera will generate all required acquisition start and frame start trigger signals internally. When the camera is set this way, it will constantly acquire images without the need for any type of triggering by the user. This use case is commonly referred to as "free run".

The rate at which the camera will acquire images will normally be determined by the camera's Acquisition Frame Rate Abs parameter when the camera is parameterized via Basler pylon or by the Acquisition Frame Period Raw register when the camera is parameterized via direct register access. If the Acquisition Frame Rate Abs parameter or the Acquisition Frame Period Raw register (respectively) is disabled, the camera will acquire frames at the maximum allowed frame rate.

In the real world, cameras are used in free run for many applications. One example is for aerial photography. A camera set for free run is used to capture a continuous series of images as an aircraft overflies an area. The images can then be used for a variety of purposes including vegetation coverage estimates, archaeological site identification, etc.

For more information about the Acquisition Frame Rate Abs parameter and the Acquisition Frame Period Raw register, see Section 6.3.1.1 on page 77 and Section 6.3.1.3 on page 80.

Use Case: "Free Run" (Acquisition Start Trigger Off and Frame Start Trigger Off)

The acquisition start trigger is off. The camera will generate acquisition start trigger signals internally with no action by the user.

The frame start trigger is off. The camera will generate frame start trigger signals internally with no action by the user.

Settings: Trigger Mode for the acquisition start trigger = Off
Trigger Mode for the frame start trigger = Off

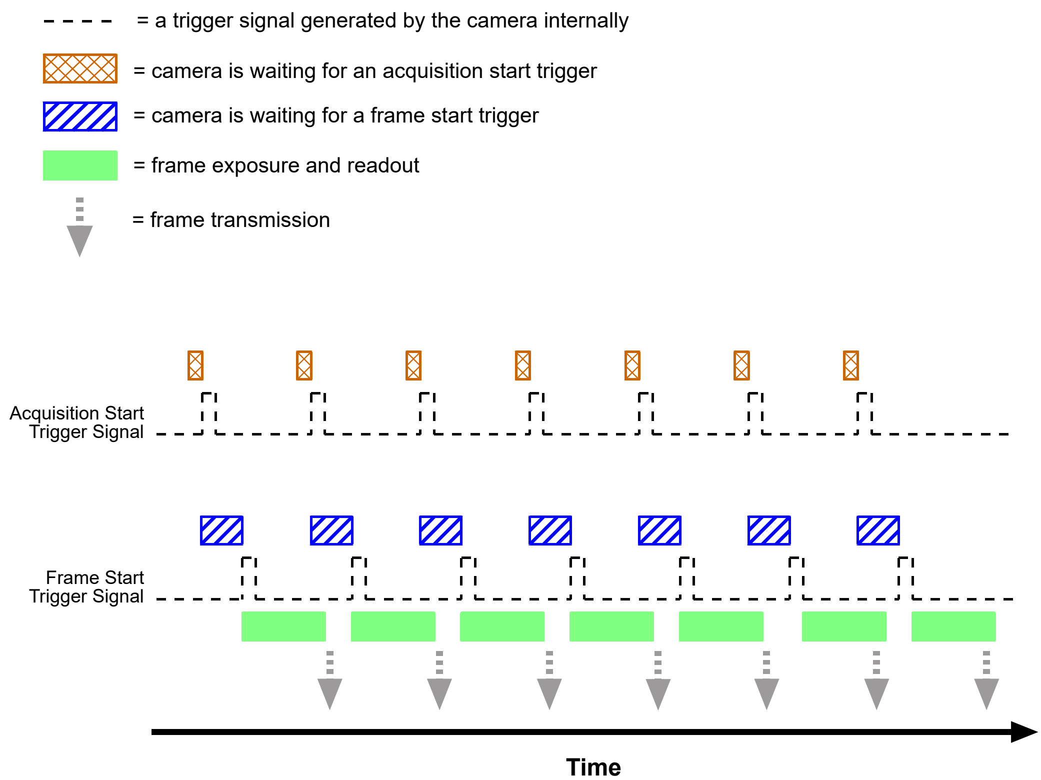


Fig. 35: Use Case 1 - Acquisition Start Trigger Off and Frame Start Trigger Off

Use Case 2 - Acquision Start Trigger Off - Frame Start Trigger On

Use case two is illustrated on page 118.

In this use case, the Trigger Mode parameter for the acquisition start trigger is set to off and the Trigger Mode parameter for the frame start trigger is set to on.

Because the acquisition start trigger is set to off, the user does not need to apply acquisition start trigger signals to the camera. The camera will generate all required acquisition start trigger signals internally.

Because the frame start trigger is set to on, the user must apply a frame start trigger signal to the camera in order to begin each frame exposure. In this case, we have set the frame start trigger signal source to line 1 and the activation to rising edge, so the rising edge of an externally generated electrical signal applied to line 1 will serve as the frame start trigger signal.

This type of camera setup is used frequently in the real world. One example might be a wood products inspection system used to inspect the surface of pieces of plywood on a conveyor belt as they pass by a camera. In this situation, a sensing device is usually used to determine when a piece of plywood on the conveyor is properly positioned in front of the camera. When the plywood is in the correct position, the sensing device transmits an electrical signal to line 1 on the camera. When the electrical signal is received on line 1, it serves as a frame start trigger signal and initiates a frame acquisition. The frame acquired by the camera is transmitted to an image processing system that will inspect the image and determine if there are any defects in the plywood’s surface.

Use Case:	Acquisition Start Trigger Off and Frame Start Trigger On The acquisition start trigger is off. The camera will generate acquisition start trigger signals internally with no action by the user. The frame start trigger is on, and the frame start trigger source is set to line 1. The user must apply a frame start trigger signal to line 1 to start each frame exposure.
Settings:	Trigger Mode for the acquisition start trigger = Off Trigger Mode for the frame start trigger = On Trigger Source for the frame start trigger = Line 1 Trigger Activation for the frame start trigger = Rising Edge

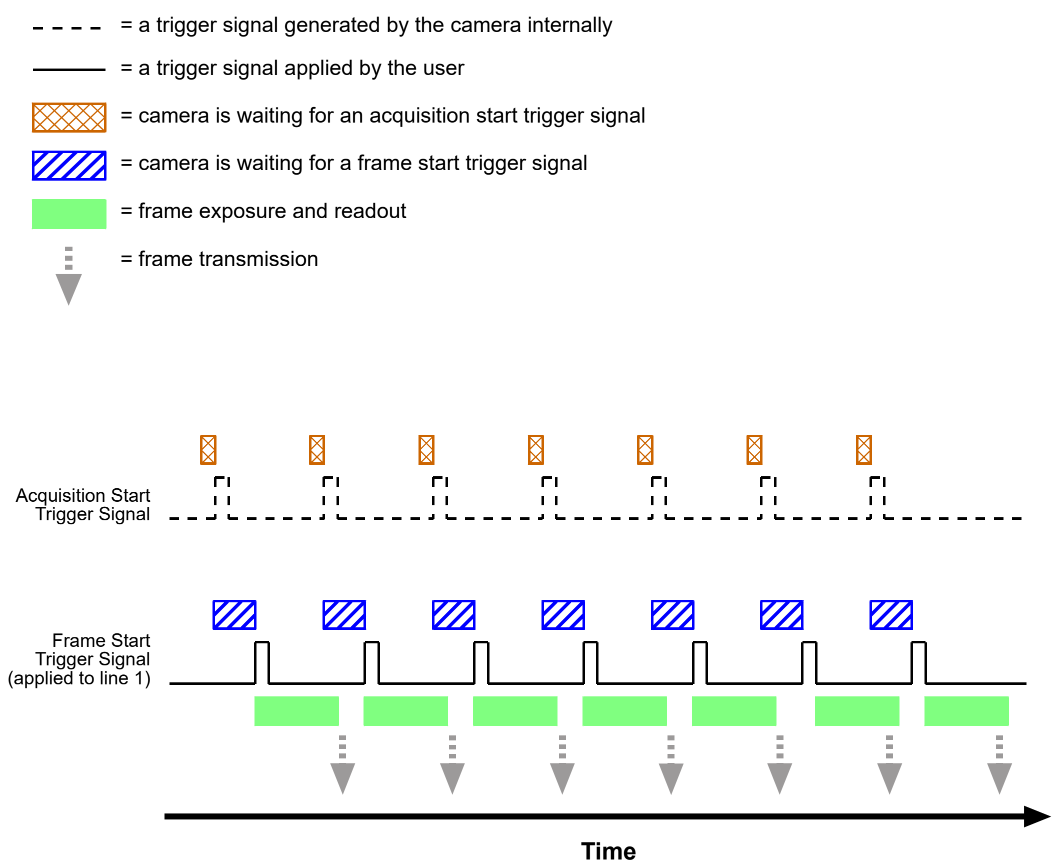


Fig. 36: Use Case 2 - Acquisition Start Trigger Off and Frame Start Trigger On

Use Case 3 - Acquision Start Trigger On - Frame Start Trigger Off

Use case three is illustrated on page 120.

In this use case, the Trigger Mode parameter for the acquisition start trigger is set to on and the Trigger Mode parameter for the frame start trigger is set to off.

Because the acquisition start trigger mode is set to on, the user must apply an acquisition start trigger signal to the camera. In this case, we have set the acquisition start trigger signal source to line 1 and the activation to rising edge, so an externally generated electrical signal applied to line 1 will serve as the acquisition start trigger signal. The Acquisition Frame Count parameter has been set to 3.

When a rising edge of the electrical signal is applied to line 1, the camera will exit the "waiting for acquisition start trigger" acquisition status and enter a "waiting for frame start trigger" acquisition status. Once the camera has acquired 3 frames, it will re-enter the "waiting for acquisition start trigger" acquisition status. Before any more frames can be acquired, a new rising edge must be applied to line 1 to make the camera exit the "waiting for acquisition start trigger" acquisition status.

Because the frame start trigger is set to off, the user does not need to apply frame start trigger signals to the camera. The camera will generate all required frame start trigger signals internally. The rate at which the frame start trigger signals will be generated is normally determined by the camera's Acquisition Frame Rate Abs parameter when the camera is parameterized via Basler pylon or by the Acquisition Frame Period Raw register the camera is parameterized via direct register access. If the Acquisition Frame Rate Abs parameter or the Acquisition Frame Period Raw register (respectively) is disabled, the camera will acquire frames at the maximum allowed frame rate.

This type of camera setup is used frequently in "intelligent traffic systems." With these systems, a typical goal is to acquire several images of a car as it passes through a toll booth. A sensing device is usually placed at the start of the toll booth area. When a car enters the area, the sensing device applies an electrical signal to line 1 on the camera. When the electrical signal is received on line 1, it serves as an acquisition start trigger signal and the camera exits from the "waiting for acquisition start trigger" acquisition status and enters a "waiting for frame trigger" acquisition status. In our example, the next 3 frame start trigger signals internally generated by the camera would result in frame acquisitions. At that point, the number of frames acquired would be equal to the setting for the Acquisition Frame Count parameter. The camera would return to the "waiting for acquisition start trigger" acquisition status and would no longer react to frame start trigger signals. It would remain in this condition until the next car enters the booth area and activates the sensing device.

This sort of setup is very useful for traffic system applications because multiple frames can be acquired with only a single acquisition start trigger signal pulse and because frames will not be acquired when there are no cars passing through the booth (this avoids the need to store images of an empty toll both area.)

For more information about the Acquisition Frame Rate Abs parameter and the Acquisition Frame Period Raw register, see Section 6.3.1.1 on page 77 and Section 6.3.1.3 on page 80.

Use Case: Acquisition Start Trigger On and Frame Start Trigger Off

The acquisition start trigger is on, and the acquisition start trigger source is set to line 1. The user must apply an acquisition start trigger signal to line 1 to make the camera exit the "waiting for acquisition start trigger" acquisition status. Because the acquisition frame count is set to 3, the camera will re-enter the "waiting for acquisition start trigger" acquisition status after 3 frames have been acquired.

The frame start trigger is off. The camera will generate frame start trigger signals internally with no action by the user.

Settings:
Trigger Mode for the acquisition start trigger = On
Trigger Source for the acquisition start trigger = Line 1
Trigger Activation for the acquisition start trigger = Rising Edge
Acquisition Frame Count = 3
Trigger Mode for the frame start trigger = Off

- - - - = a trigger signal generated by the camera internally

——— = a trigger signal applied by the user

= camera is waiting for an acquisition start trigger signal

= camera is waiting for a frame start trigger signal

= frame exposure and readout

= frame transmission

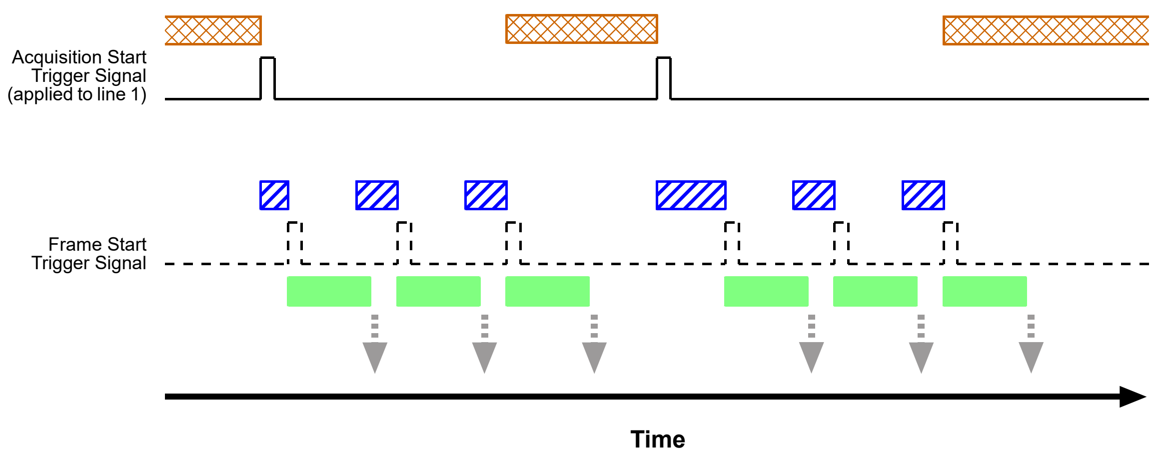


Fig. 37: Use Case 3 - Acquisition Start Trigger On and Frame Start Trigger Off

Use Case 4 - Acquisition Start and Frame Start Triggers On

Use case four is illustrated on page 122.

In this use case, the Trigger Mode parameter for the acquisition start trigger is set to on and the Trigger Mode parameter for the frame start trigger is set to on.

Because the acquisition start trigger mode is set to on, the user must apply an acquisition start trigger signal to the camera. In this case, we have set the acquisition start trigger signal source to software, so the execution of an acquisition trigger software command will serve as the acquisition start trigger signal. The Acquisition Frame Count parameter is set to 3.

When an acquisition trigger software command is executed, the camera will exit the "waiting for acquisition start trigger" acquisition status and enter a "waiting for frame start trigger" acquisition status. Once the camera has acquired 3 frames, it will re-enter the "waiting for acquisition start trigger" acquisition status. Before any more frames can be acquired, a new acquisition trigger software command must be executed to make the camera exit the "waiting for acquisition start trigger" acquisition status.

Because the frame start trigger is set to on, the user must apply a frame start trigger signal to the camera in order to begin each frame acquisition. In this case, we have set the frame start trigger signal source to line 1 and the activation to rising edge, so the rising edge of an externally generated electrical signal applied to line 1 will serve as the frame start trigger signal. Keep in mind that the camera will only react to a frame start trigger signal when it is in a "waiting for frame start trigger" acquisition status.

A possible use for this type of setup is a conveyor system that moves objects past an inspection camera. Assume that the system operators want to acquire images of 3 specific areas on each object, that the conveyor speed varies, and that they do not want to acquire images when there is no object in front of the camera. A sensing device on the conveyor could be used in conjunction with a computer to determine when an object is starting to pass the camera. When an object is starting to pass, the computer will execute an acquisition start trigger software command, causing the camera to exit the "waiting for acquisition start trigger" acquisition status and enter a "waiting for frame start trigger" acquisition status.

An electrical device attached to the conveyor could be used to generate frame start trigger signals and to apply them to line 1 on the camera. Assuming that this electrical device was based on a position encoder, it could account for the speed changes in the conveyor and ensure that frame trigger signals are generated and applied when specific areas of the object are in front of the camera. Once 3 frame start trigger signals have been received by the camera, the number of frames acquired would be equal to the setting for the Acquisition Frame Count parameter, and the camera would return to the "waiting for acquisition start trigger" acquisition status. Any frame start trigger signals generated at that point would be ignored.

This sort of setup is useful because it will only acquire frames when there is an object in front of the camera and it will ensure that the desired areas on the object are imaged. (Transmitting images of the "space" between the objects would be a waste of bandwidth and processing them would be a waste of processor resources.)

Use Case: Acquisition Start Trigger On and Frame Start Trigger On

The acquisition start trigger is on, and the acquisition start trigger source is set to software. The user must execute an acquisition start trigger software command to make the camera exit the "waiting for acquisition start trigger" acquisition status. Because the acquisition frame count is set to 3, the camera will re-enter the "waiting for acquisition start trigger" acquisition status after 3 frame trigger signals have been applied.

The frame start trigger is on, and the frame start trigger source is set to line 1. The user must apply a frame start trigger signal to line 1 to start each frame exposure.

Settings:
Trigger Mode for the acquisition start trigger = On
Trigger Source for the acquisition start trigger = Software
Acquisition Frame Count = 3
Trigger Mode for the frame start trigger = On
Trigger Source for the frame start trigger = Line 1
Trigger Activation for the frame start trigger = Rising Edge

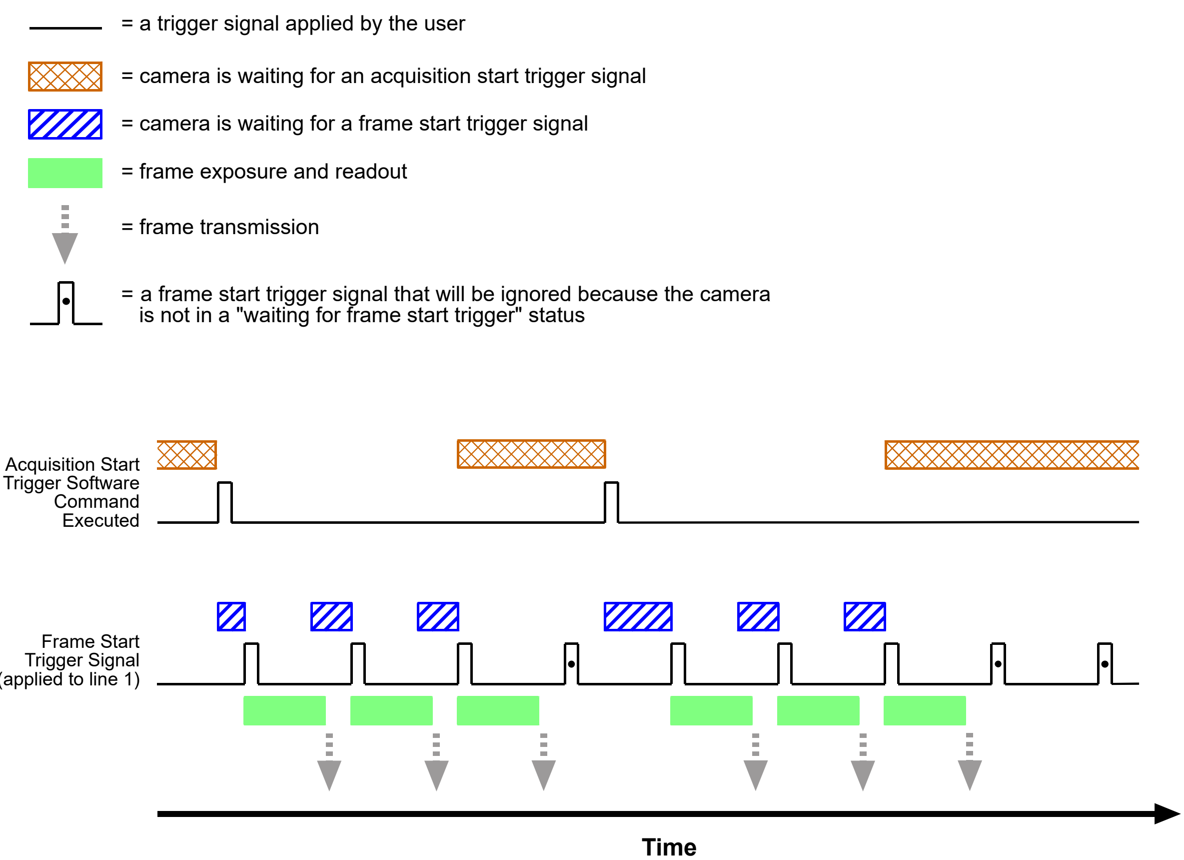


Fig. 38: Use Case 4 - Acquisition Start Trigger On and Frame Start Trigger On

7 Color Creation and Enhancement

This chapter provides information about how color images are created on different camera models and about the features available for adjusting the appearance of the colors.

7.1 Color Creation

The sensor used in color models of the camera is equipped with an additive color separation filter known as a Bayer filter. The pixel data output formats available on color cameras are related to the Bayer pattern, so you need a basic knowledge of the Bayer filter to understand the pixel formats. With the Bayer filter, each individual pixel is covered by a part of the filter that allows light of only one color to strike the pixel. As the figure illustrates, within each square of four pixels, one pixel sees only red light, one sees only blue light, and two pixels see only green light. (This combination mimics the human eye's sensitivity to color.) The alignment of the Bayer filter with respect to the sensor is shown in Figure 39 as an example only; the figure shows "GR" alignment.

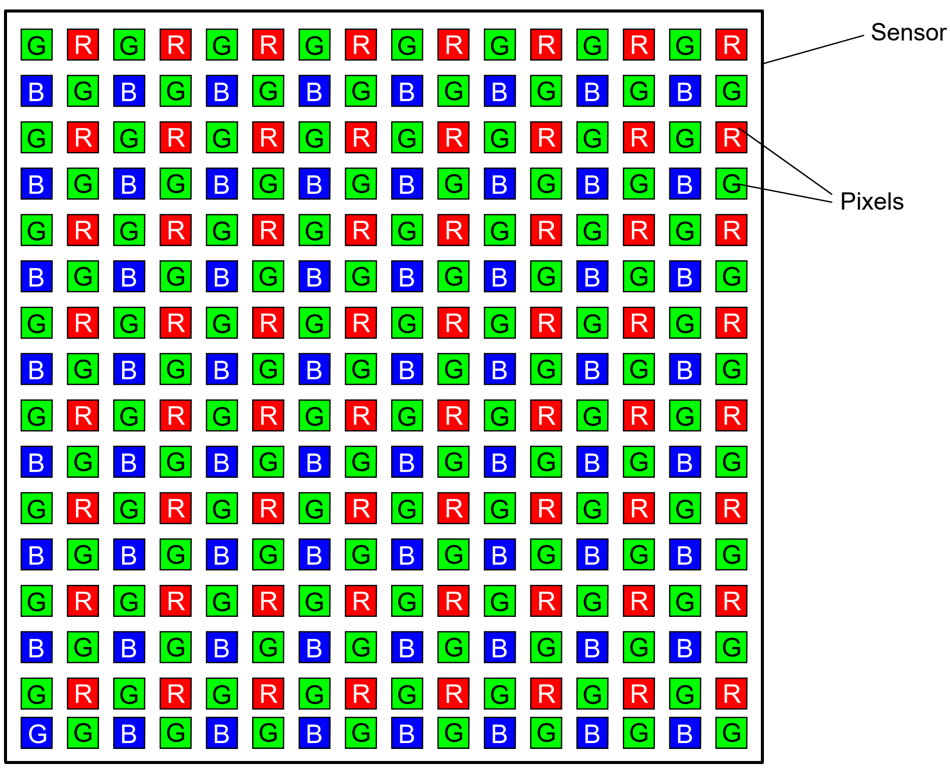


Fig. 39: Bayer Filter Pattern

7.1.1 Color Filter Alignment on the aviator

The alignment of the Bayer filter to the pixels in the images acquired by color versions of the camera is Bayer GR.

Bayer GR alignment means that the first and second pixel of the first line in each image transmitted will be green and red respectively. And for the second line transmitted, the first pixel and the second pixel will be blue and green respectively. Since the pattern of the Bayer filter is fixed, you can use this information to determine the color of all of the other pixels in the image.

Since the area of interest (AOI) width, and height can only be changed in increments of 4 on color cameras, the alignment of the Bayer filter to the pixels in the transmitted images will stay the same regardless of the size of the AOI.

When either the reverse X feature or the reverse Y feature or both are used, the alignment of the color filter to the image remains Bayer GR. The camera includes a mechanism that keeps the filter alignment constant when these features are used.

For more information about

- the AOI feature, see Section 9.7 on page 167.
- the reverse X and reverse Y features, see Section 9.13 on page 237.

7.2 Integrated IR Cut Filter

Color models of the camera that have a C-mount lens adapter are equipped with an IR cut filter as standard equipment. The filter is mounted inside of the lens adapter. Cameras without an IR cut filter are available on request.

Monochrome cameras do not include an IR cut filter in the lens adapter. Monochrome cameras with a C-mount lens adapter can be equipped with a filter on request.

NOTICE
On color cameras, the lens thread length is limited. Color models of the camera with a C-mount lens adapter are equipped with an IR cut filter mounted inside of the adapter. The location of this filter limits the length of the threads on any lens you use with the camera. If a lens with a very long thread length is used, the IR cut filter will be damaged or destroyed and the camera will no longer operate. Do not use a lens with a thread length greater than 7.5 mm.

For more information about the location of the IR cut filter, see Section 1.4.2 on page 8.

7.3 Color Enhancement Features

7.3.1 White Balance

The white balance feature can be used to adjust the color balance of the images transmitted from the camera.

This section (Section 7.3.1) describes how the camera's white balance can be adjusted "manually", i.e., by setting the value of the individual white balance related parameters.

Setting the White Balance

> The camera also has a White Balance Auto function that can automatically adjust the white balance. **Manual adjustment of the white balance parameters will only work correctly, if the White Balance Auto function is disabled.**
>
> For more information about
>
> - auto functions in general, see Section 9.8 on page 173.
> - the White Balance Auto function in particular, see Section 9.8.9 on page 190.
>
> When you are using matrix color transformation and you set the Light Source Selector parameter to match your light source characteristics, the camera will automatically make adjustments to the white balance settings so that they are best suited for the light source you selected.
>
> For more information about matrix color transformation, see Section 7.3.3 on page 130 and Section 7.3.3 on page 130.

With the white balancing scheme used on these cameras, the red intensity, green intensity, and blue intensity can each be adjusted. For each color, a Balance Ratio parameter is used to set the intensity of the color. If the Balance Ratio parameter for a color is set to a value of 1, the intensity of the color will be unaffected by the white balance mechanism. If the ratio is set to a value lower than 1, the intensity of the color will be reduced. If the ratio is set to a value greater than 1, the intensity of the color will be increased. The increase or decrease in intensity is proportional. For example, if the balance ratio for a color is set to 1.2, the intensity of that color will be increased by 20%.

The Balance Ratio Abs parameter value can range from 0.00 to 15.9844. But you should be aware that if you set the balance ratio for a color to a value lower than 1, this will not only decrease the intensity of that color relative to the other two colors, but will also decrease the maximum intensity that the color can achieve. For this reason, we don't normally recommend setting a balance ratio less than 1 unless you want to correct for the strong predominance of one color.

To set the Balance Ratio Abs parameter for a color:

- Set the Balance Ratio Selector to red, green, or blue.
- Set the Balance Ratio Abs parameter to the desired value for the selected color.

You can set the Balance Ratio Selector and the Balance Ratio Abs parameter value from within your application software by using the Basler pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
Camera.BalanceRatioSelector.SetValue( BalanceRatioSelector_Green );
Camera.BalanceRatioAbs.SetValue( 1.20 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

White Balance Reset

The camera includes a White Balance Reset command that can be used to reset the white balance adjustments. This feature is especially useful, if you have badly misadjusted the white balance and you want to quickly return to reasonable settings. When the reset command is used, it will return the camera to the settings defined by your current Light Source Selector parameter setting.

You can execute the White Balance Reset command from within your application software by using the pylon API. The following code snippet illustrates using the API to execute the command:

```
// Reset the white balance adjustments
Camera.BalanceWhiteReset.Execute( );
```

You can also use the Basler pylon Viewer application to easily execute the command.

Setting the White Balance Using Direct Register Access

To set the balance ratio parameters via direct register access:

- Set the value of White Balance Red register to adjust the red ratio.
- Set the value of the White Balance Green register to adjust the green ratio.
- Set the value of the White Balance Blue register to adjust the blue ratio.

For more information about changing settings via direct register access, see Section 5 on page 61.

7.3.2 Gamma Correction

The gamma correction feature lets you modify the brightness of the pixel values output by the camera's sensor to account for a non-linearity in the human perception of brightness.

There are two modes of gamma correction available on the camera: sRGB and User.

sRGB Gamma

When the camera is set for sRGB gamma correction, it automatically sets the gamma correction to adjust the pixel values so that they are suitable for display on an sRGB monitor. If you will be displaying the images on an sRGB monitor, using this type of gamma correction is appropriate.

User Gamma

With User type gamma correction, you can set the gamma correction value as desired.

To accomplish the correction, a gamma correction value (γ) is applied to the brightness value (Y) of each pixel according to the following formula:

$$Y_{corrected} = \left(\frac{Y_{uncorrected}}{Y_{max}}\right)^{\gamma} \times Y_{max}$$

The formula uses uncorrected and corrected pixel brightnesses that are normalized by the maximum pixel brightness. The maximum pixel brightness equals 255 for 8 bit output and 4095 for 12 bit output.

The gamma correction value can be set in a range from 0 to 3.99998.

When the gamma correction value is set to 1, the output pixel brightness will not be corrected.

A gamma correction value between 0 and 1 will result in increased overall brightness, and a gamma correction value greater than 1 will result in decreased overall brightness.

In all cases, black (output pixel brightness equals 0) and white (output pixel brightness equals 255 at 8 bit output and 4095 at 12 bit output) will not be corrected.

Enabling and Setting Gamma Correction

You can enable or disable the gamma correction feature by setting the value of the Gamma Enable parameter.

You can use the Gamma Selector to select either sRGB or user gamma correction.

If you select user gamma correction, you can use the Gamma parameter to set the gamma correction value.

You can set the Gamma Enable parameter, use the Gamma Selector, and set Gamma parameter values from within your application software by using the Basler pylon API. The following code snippet illustrates using the API to set the parameter values for sRGB type correction:

```
// Enable the Gamma feature
Camera.GammaEnable.SetValue( true );
// Set the gamma type to sRGB
Camera.GammaSelector.SetValue ( GammaSelector_sRGB );
```

The following code snippet illustrates using the API to set the parameter values for user type correction:

```
// Enable the Gamma feature
Camera.GammaEnable.SetValue( true );
// Set the gamma type to User
Camera.GammaSelector.SetValue ( GammaSelector_User );
// Set the Gamma value to 1.2
Camera.Gamma.SetValue( 1.2 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Enabling Gamma Correction and Setting Gamma Using Direct Register Access

To enable gamma correction and to set the gamma value via direct register access:

- Set the value of the Gamma Enable register to Enabled.
- Set the value in the Gamma register to the desired gamma value.

For more information about changing settings via direct register access, see Section 5 on page 61.

7.3.3 Matrix Color Transformation on Color Models

Introduction

The main objective of matrix color transformation is to make corrections to the color information that will account for the type of lighting used during image acquisition and to compensate for imperfections in the sensor's color generation process.

With the matrix color transformation, a first matrix transformation step ensures that the pixel values from the sensor are available in RGB color space, i.e. as R, G, or B component for each pixel. A second transformation step takes account of the specific pre-selected light source. The vector consisting of the R, G, or B component for each pixel in the image is multiplied by a matrix containing a set of correction values.

Matrix Color Transformation Parameters

The initial parameter that you must consider when working with the matrix color transformation feature is the Processed Raw Enable parameter. If the camera is set to output pixel data in the Bayer xx format, then the Processed Raw Enable parameter must be set to "enabled" to allow color enhancements to be performed. Setting this parameter to enabled will allow the camera to perform color enhancements on the raw RGB data from the sensor and still be able to output the pixel data in one of the Bayer formats. If the camera is set for a Bayer xx pixel data output format and the Processed Raw Enable parameter isSection 7.3.3 on page 130 not set to enabled, the matrix color transformation feature and the color adjustment feature will have no effect on camera operation.

The first parameter associated with the matrix color transformation feature is the **Color Transformation Selector** parameter. This parameter is used to select the type of transformation that will be performed before color correction for a specific light source is performed (addressed by the second parameter). For cameras equipped with a Bayer pattern filter on the imaging sensor, RGB to RGB is the only setting available. This setting means that the matrix color transformation process will not transform the red, green, and blue pixel values from the sensor into a different color space.

The second parameter associated with matrix color transformation is the **Light Source Selector** parameter that defines different light source presets. The following settings are available for this parameter:

- Off - No alterations will be made to the pixel values, i.e. no color transformation and gamma is applied.
- Tungsten - This setting will automatically populate the matrix with a pre-selected set of values that will make appropriate corrections for images captured with tungsten lighting that has a color temperature of about 2500K to 3000K. When you select this setting, the camera will also adjust the white balance settings and the color adjustment settings so that they are appropriate for a tungsten light source.
- Daylight - This setting will automatically populate the matrix with a pre-selected set of values that will make appropriate corrections for images captured with daylight lighting that has a color temperature of about 5000K. When you select this setting, the camera will also adjust the

white balance settings and the color adjustment settings so that they are appropriate for a daylight light source with a color temperature of about 5000K.

- Daylight 6500K - This setting will automatically populate the matrix with a pre-selected set of values that will make appropriate corrections for images captured with daylight lighting that has a color temperature of about 6500K. When you select this setting, the camera will also adjust the white balance settings and the color adjustment settings so that they are appropriate for a daylight light source with a color temperature of about 6500K.
- Custom - The user can set the values in the matrix as desired. When you select this setting, the camera will also adjust the white balance settings and the color adjustment settings so that they have neutral values (line sum of coefficients = 1).

In almost all cases, selecting one of the settings that populate the matrix with pre-selected values will give you excellent results with regard to correcting the colors for the light source you are using.

The custom setting should only be used by someone who is thoroughly familiar with matrix color transformations. Instructions for using the custom setting appear in the next section.

The third parameter associated with matrix color transformation is the **Color Transformation Matrix Factor** parameter. This parameter determines how strong an effect the matrix correction function will have on the colors output by the camera. The parameter setting is a floating point value that can range from 0 to 1. When the parameter value is set to 0, matrix correction will have no effect. When the value is set to 1, matrix correction will have its maximum effect.

As an alternative, the Color Transformation Matrix Factor parameter value can be entered as an integer value on a scale ranging from 0 to 65536. This integer range maps linearly to the floating point range with 0 being equivalent to 0 and 65536 being equivalent to 1. The integer values can be entered using the Color transformation Matrix Factor Raw parameter.

> When the Light Source Selector parameter is set to off or custom, the Color Transformation Matrix Factor parameter will not be available.

Setting Matrix Color Transformation

You can set the Processed Raw Enable, Color Transformation Selector and Light Source Selector parameter values from within your application software by using the Basler pylon API.
In this example, we assume that you want to set your camera for Bayer BG 8 output, and therefore you must set the Processed Raw Enable parameter value to enabled.

The following code snippet illustrates using the API to set the parameter values:

```
// Set the camera for Bayer GR8 pixel data output format
Camera.PixelFormat.SetValue( PixelFormat_BayerGR8 );
// Because the camera is set for a Bayer output format, the Processed
Raw
// Enabled parameter must be set to enabled
Camera.ProcessedRawEnable.SetValue( true );
```

```
// Select the matrix color transformation type
Camera.ColorTransformationSelector.SetValue
    ( ColorTransformationSelector_RGBtoRGB );

// Set the light source selector so that no correction will be done
Camera.LightSourceSelector.SetValue
    ( LightSourceSelector_Off );
// Set the light source selector for tungsten lighting
Camera.LightSourceSelector.SetValue
    ( LightSourceSelector_Tungsten );
// Set the light source selector for daylight (at about 5000K)
Camera.LightSourceSelector.SetValue
    ( LightSourceSelector_Daylight );
// Set the light source selector for daylight (at about 6500K)
Camera.LightSourceSelector.SetValue
    ( LightSourceSelector_Daylight6500K );

// Set the matrix correction factor
Camera.ColorTransformationMatrixFactor.SetValue( 0.50 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

7.3.3.1 The Custom Light Source Setting

i	The "Custom" setting for the Light Source Selector parameter is intended for use by someone who is thoroughly familiar with matrix color transformations. **It is nearly impossible to enter correct values in the conversion matrix by trial and error.**

The RGB to RGB color matrix conversion for each pixel is performed by multiplying a 1 x 3 matrix containing R, G, and B color values with a 3 x 3 matrix containing correction values. Each column in the 3 x 3 matrix can be populated with values of your choice. In other words:

$$\begin{bmatrix} Gain00 & Gain01 & Gain02 \\ Gain10 & Gain11 & Gain12 \\ Gain20 & Gain21 & Gain22 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} R' \\ G' \\ B' \end{bmatrix}$$

Where Gain00, Gain01, etc. are settable values.

Each GainXY position can be populated with a floating point value ranging from -8.0 to +7.96875 by using the Color Transformation Value Selector to select one of the GainXY positions in the matrix and using the Color transformation Value parameter to enter a value for that position.

As an alternative the Gain XY values can each be entered as an integer value on a scale ranging from -256 to +255. This integer range maps linearly to the floating point range with -256 being equivalent to -8.0, 32 being equivalent to 1.0, and +255 being equivalent to +7.96875. The integer values can be entered using the Color transformation Value Raw parameter.

A reference article that explains the basics of color matrix transformation for video data can be found at:
http://www.its.bldrdoc.gov/publications/2437.aspx

Setting Custom Matrix Values

You can set the Color Transformation Value Selector, Color Transformation Value, and Color Transformation Value Raw parameters from within your application software by using the Basler pylon API. The following code snippet illustrates using the API to set the values in the matrix. Note that the values in this example are just randomly selected numbers and do not represent values that you should actually use.

```
// Set the light source selector for custom
Camera.LightSourceSelector.SetValue ( LightSourceSelector_Custom );

// Select a position in the matrix
Camera.ColorTransformationValueSelector.SetValue
     ( ColorTransformationValueSelector_Gain01 );
// Set the value for the selected position as a floating point value
Camera.ColorTransformationValue.SetValue( 2.11 );

// Select a position in the matrix
Camera.ColorTransformationValueSelector.SetValue
     ( ColorTransformationValueSelector_Gain12 );
// Set the value for the selected position as an integer value
Camera.ColorTransformationValueRaw.SetValue( 135 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

7.3.4 Color Adjustment

On all color cameras equipped with a Bayer pattern filter the pixel values output from the sensor reside in the RGB color space.

The camera’s color adjustment feature lets you adjust hue and saturation for the primary and secondary colors in the RGB color space. Each adjustment affects those colors in the image where the adjusted primary or secondary color predominates. For example, the adjustment of red affects the colors in the image with a predominant red component.

For the color adjustments to work properly, the white balance must be correct.

See Section 7.3.1 on page 126 for more information about the white balance and see Section 7.3.5 on page 139 for an overall procedure for setting the color enhancement features.

Although color adjustment can be used without also using color matrix transformation, we nonetheless strongly recommend to also use color matrix transformation to make full use of the camera’s color enhancement capabilities.

See Section 7.3.3 on page 130 for more information about color matrix transformation.

The RGB Color Space

The RGB color space includes light with the primary colors red, green, and blue and all of their combinations. When red, green, and blue light are combined and when the intensities of R, G, and B are allowed to vary independently between 0% and 100%, all colors within the RGB color space can be formed. Combining colored light is referred to as additive mixing.

When two primary colors are mixed at equal intensities, the secondary colors will result. The mixing of red and green light produces yellow light (Y), the mixing of green and blue light produces cyan light (C), and the mixing of blue and red light produces magenta light (M).

When the three primary colors are mixed at maximum intensities, white will result. In the absence of light, black will result.

The color space can be represented as a color cube (see Figure 40 on page 135) where the primary colors R, G, B, the secondary colors C, M, Y, and black and white define the corners. All shades of gray are represented by the line connecting the black and the white corner.

For ease of imagination, the color cube can be projected onto a plane (as shown in Figure 40) such that a color hexagon is formed. The primary and secondary colors define the corners of the color hexagon in an alternating fashion. The edges of the color hexagon represent the colors resulting from mixing the primary and secondary colors. The center of the color hexagon represents all shades of gray including black and white.

The representation of any arbitrary color of the RGB color space will lie within the color hexagon. The color will be characterized by its hue and saturation:

- Hue specifies the kind of coloration, for example, whether the color is red, yellow, orange etc.
- Saturation expresses the colorfulness of a color. At maximum saturation, no shade of gray is present. At minimum saturation, no "color" but only some shade of gray (including black and white) is present.

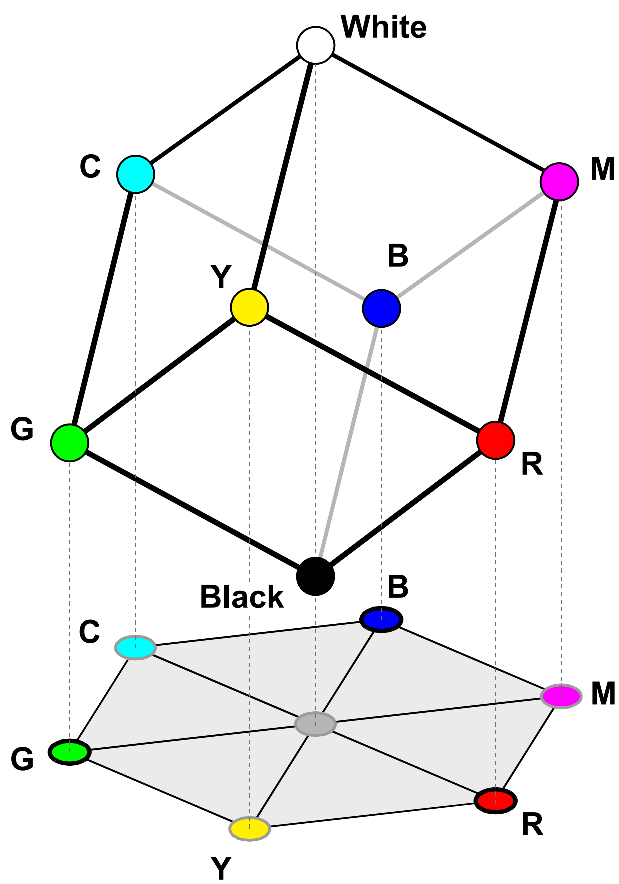


Fig. 40: RGB Color Cube With YCM Secondary Colors, Black, and White, Projected On a Plane

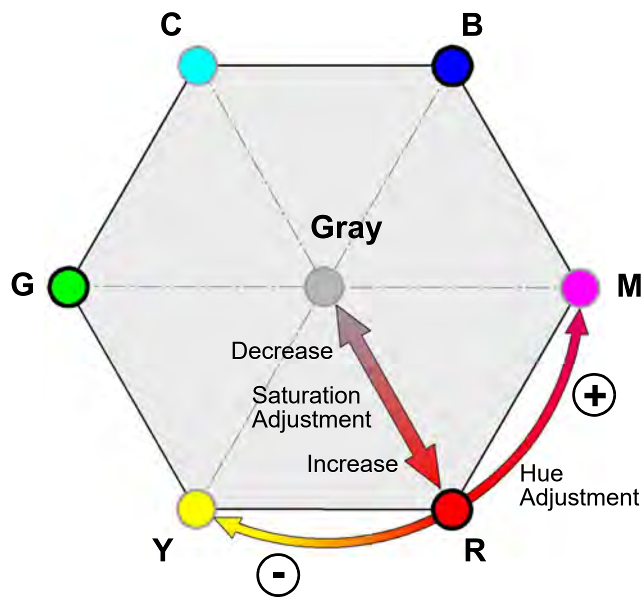


Fig. 41: Hue and Saturation Adjustment In the Color Hexagon. Adjustments Are Indicated for Red as an Example

Hue and Saturation Adjustment

The color adjustment feature lets you adjust hue and saturation for the primary and the secondary colors. Each adjustment affects those areas in the image where the adjusted color predominates. For example, the adjustment of red affects the colors in the image with a predominantly red component.

Keep in mind that when you adjust a color, the colors on each side of it in the color hexagon will also be affected to some degree. For example, when you adjust red, yellow and magenta will also be affected.

- In the color hexagon, the adjustment of hue can be considered as a rotation between hues. Primary colors can be rotated towards, and as far as, their neighboring secondary colors. And secondary colors can be rotated towards, and as far as, their neighboring primary colors.

  For example, when red is rotated in negative direction towards yellow, then, for example, purple in the image can be changed to red and red in the image can be changed to orange.
  Red can be rotated as far as yellow, where red will be completely transformed into yellow.

  When red is rotated in a positive direction towards magenta, then, for example, orange in the image can be changed to red and red in the image can be changed to purple.
  Red can be rotated as far as magenta, where red will be completely transformed into magenta.
- Adjusting saturation changes the colorfulness (intensity) of a color. The color adjustment feature lets you adjust saturation for the primary and secondary colors.

  For example, if saturation for red is increased, the colorfulness of red colors in the image will increase. If red is set to minimum saturation, red will be replaced by gray for "red" colors in the image.

Color Adjustment Parameters

The initial parameter that you must consider when working with the color adjustment feature is the Processed Raw Enable parameter. If you are working with a camera that is set to output pixel data in a Bayer xx format, then the Processed Raw Enabled parameter must be set to "enabled", if you want to use color enhancement. The camera will then be able to perform color enhancements on the raw RGB data from the sensor and still be able to output the pixel data in one of the Bayer formats. If the camera is set for a Bayer xx pixel data output format and the Processed Raw Enable parameter is not set to enabled, the matrix color transformation feature and the color adjustment feature will have no effect on the camera operation.

You can enable or disable the color adjustment feature by setting the value of the Color Adjustment Enable parameter to true or false.

You can use the Color Adjustment Selector parameter to select a color to adjust. The colors you can select are: red, yellow, green, cyan, blue, and magenta.

You can use the Color Adjustment Hue parameter to set the hue for the selected color as a floating point value in a range from -4.0 to +3.96875.

As an alternative, you can use the Color Adjustment Hue Raw parameter to set the hue as an integer value on a scale ranging from -128 to +127. This integer range maps linearly to the floating point range with -256 being equivalent to -4.0, 32 being equivalent to 1.0, and +255 being equivalent to +3.96875.

You can use the Color Adjustment Saturation parameter to set the saturation for the selected color as a floating point value in a range from 0.0 to +1.99219.

As an alternative, you can use the Color Adjustment Saturation Raw parameter to set the saturation as an integer value on a scale ranging from 0 to 255. This integer range maps linearly to the floating point range with 0 being equivalent to 0.0, 128 being equivalent to 1.0, and +255 being equivalent to +1.99219.

Enabling and Setting Color Adjustment

You can set the Processed Raw Enable, Color Adjustment Enable, Color Adjustment Selector, Color Adjustment Hue, Color Adjustment Hue Raw, Color Adjustment Saturation, and Color Adjustment Saturation Raw parameter values from within your application software by using the Basler pylon API. In this example, we assume that you want to set your camera for Bayer BG8 output, and therefore you must set the Processed Raw Enable parameter value to enabled.

The following code snippet illustrates using the API to set the parameter values:

```
// Set the camera for Bayer BG8 pixel data output format
Camera.PixelFormat.SetValue( PixelFormat_BayerBG8 );
// Because the camera is set for a Bayer output format, the Processed
// Raw Enabled parameter must be set to enabled
Camera.ProcessedRawEnable.SetValue( true );
```

```
// Enable the Color Adjustment feature
Camera.ColorAdjustmentEnable.SetValue( true );

// Select red as the color to adjust
Camera.ColorAdjustmentSelector.SetValue( ColorAdjustmentSelector_Red
);

// Set the red hue as a floating point value
Camera.ColorAdjustmentHue.SetValue( -1.125 );
// Set the red saturation as a floating point value
Camera.ColorAdjustmentSaturation.SetValue( 1.375 );

// Select cyan as the color to adjust
Camera.ColorAdjustmentSelector.SetValue( ColorAdjustmentSelector_-
Cyan );

// Set the cyan hue as an integer value
Camera.ColorAdjustmentHueRaw.SetValue( -36 );
// Set the cyan saturation as an integer value
Camera.ColorAdjustmentSaturationRaw.SetValue( 176 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Color Adjustment Reset

The camera includes a Color Adjustment Reset command that can be used to reset the color adjustments. This feature is especially useful, if you have badly misadjusted the colors and you want to quickly return to reasonable settings. When the reset command is used, it will return the camera to the settings defined by your current Light Source Selector parameter setting.

You can execute the Color Adjustment Reset command from within your application software by using the pylon API. The following code snippet illustrates using the API to execute the command:

```
// Reset the color adjustments
Camera.ColorAdjustmentReset.Execute( );
```

You can also use the Basler pylon Viewer application to easily execute the command.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

7.3.5 A Procedure for Setting the Color Enhancements

When setting the color enhancements on the camera, we recommend using the procedure outlined below. Since it makes changing camera parameters quick and easy, we also recommend using the Basler pylon Viewer software when you are making adjustments.

1. Arrange your camera so that it is viewing a scene similar to what it will view during actual operation. Make sure that the lighting for the scene is as close as possible to the actual lighting you will be using during normal operation. (Using lighting that represents your normal operating conditions is extremely important.)
2. We recommend including a standard color chart within your camera's field of view when you are adjusting the color enhancements. This will make it much easier to know when the colors are properly adjusted. One widely used chart is the ColorChecker® chart (also known as the Macbeth chart).
3. To start, leave the Light Source Selector parameter at the default setting.
4. Begin capturing images and check the basic image appearance. Set the exposure time and gain so that you are acquiring good quality images. It is important to make sure that the images are not over exposed. Over exposure can have a significant negative effect on the fidelity of the color in the acquired images.
5. Adjust the white balance. An easy way to set the white balance is to use the "once" function on the camera's balance white auto feature.
6. Set the gamma value. You should set the value to match the gamma on the monitor you are using to view acquired images. When gamma is set correctly, there should be a smooth transition from the lightest to the darkest gray scale targets on your color chart.

   (The sRGB gamma preset will give you good results on most CRT or LCD monitors.)
7. Examine the colors and see, if they are satisfactory at this point. If not, chose a different setting for the Light Source Selector parameter. Try each mode and determine which one gives you the best color results.
8. The color fidelity should now be quite good. If you want to make additional changes, adjust the hue and saturation by using the color adjustment feature. Keep in mind that when you adjust a color, the colors on each side of it in the color hexagon will also be affected to some degree. For example, when you adjust red, yellow and magenta will also be affected.

   When you are making hue and saturation adjustments, it is a good idea to start by concentrating on one line in the color chart. Once you have the colors in a line properly adjusted, you can move on to each of the other lines in turn.

i	When you first start working with the color enhancement tools, it is easy to badly misadjust the color adjustment settings and not be able to bring them back into proper adjustment. You can easily recover from this situation by using the camera's color adjustment reset command (see page 138). Another way to recover is to make the cameras "color factory setup" the default configuration set and then to load the default configuration set into the camera's active set. See the next section for more information about the camera's color factory setup.

7.3.6 The "Color" Factory Setup

When a camera leaves the factory, it contains several "factory setups" stored in its permanent memory. A factory setup is simply a collection of settings for the parameters needed to operate the camera. Each one of the factory setups is optimized to make the camera perform well in a particular situation. One of the setups is known as the "color factory setup", and the parameter settings contained in the color factory setup are optimized to produce good color images under the most common lighting conditions.

To make the parameters contained in the color factory setup become the ones that are actively controlling camera operation, you must select the color factory setup as the default camera configuration set and then you must load the default configuration set into the camera's active configuration set. When you do this, it will:

- Set the Gamma Selector parameter to sRGB
- Set the Processed Raw Enable parameter to enabled.
- Set the Light Source Selector parameter to Daylight 6500.
- Set the white balance parameters to values that are suitable for daylight lighting.

If you have badly misadjusted the settings for the color enhancement features on the camera, it may be difficult to bring the settings back into proper adjustment. Selecting the color factory setup as the default set and then loading the default set into the active set is a good way to recover from gross misadjustment of the color features.

For more information about the factory setups and about selecting and loading configuration sets, see Section 9.19 on page 256.

8 Pixel Formats, and Tap Geometries

This chapter provides information about the pixel formats and the Camera Link tap geometries available on the camera. By selecting a pixel format, you determine the bit depth of the image data transmitted by the camera. By selecting a tap geometry, you determine how the pixel data will be transmitted over the Camera Link interface.

i	If you plan to design your own frame grabber, or if you would like complete details regarding the way that pixel data is handled by the camera, refer to the document called *Aviator Information for Frame Grabber Designers* (AW000831). You can download the document from the Basler website: www.baslerweb.com

8.1 Pixel Formats

The camera uses 12 bit ADCs to digitize the pixel values captured by the camera's imaging sensor. The choice of a pixel format determines whether all 12 bits of data will be transmitted from the camera for each pixel in the acquired frames or if only 8 or 10 bits will be transmitted for each pixel. (This is often referred to as setting the bit depth of the pixels.)

On **monochrome** versions of the camera, the pixel format can be set to

- Mono 8,
- Mono 10, or
- Mono 12.

With the Mono 8 setting, the 8 most significant bits from the camera's ADCs will be transmitted for each pixel and the 4 least significant bits will be dropped. With the Mono 10 setting, the 10 most significant bits from the camera's ADCs will be transmitted for each pixel and the 2 least significant bits will be dropped. With the Mono 12 setting, all 12 bits from the camera's ADCs will be transmitted for each pixel.

On **color** versions of the camera, the pixel format can be set to

- Bayer GR 12,
- Bayer GR 10, or
- Bayer GR 8.

With the Bayer GR 8 setting, the 8 most significant bits from the camera's ADCs will be transmitted for each pixel and the 4 least significant bits will be dropped. With the Bayer GR 10 setting, the 10 most significant bits from the camera's ADCs will be transmitted for each pixel and the 2 least significant bits will be dropped. With the Bayer GR 12 setting, all 12 bits from the camera's ADCs will be transmitted for each pixel.

The Bayer GR refers to the color filter used in color versions of the camera.

For more information about the Bayer filter on color cameras, see Section 7.1 on page 123.

Setting the Pixel Format Using Basler pylon

You can use the pylon API to set the Pixel Format parameter value from within your application. The following code snippet illustrates using the pylon API to set the parameter value:

```
// Set pixel format to Mono 8
Camera.PixelFormat.SetValue( PixelFormat_Mono8 );

// Set pixel format to Mono 10
Camera.PixelFormat.SetValue( PixelFormat_Mono10 );

// Set pixel format to Mono 12
Camera.PixelFormat.SetValue( PixelFormat_Mono12 );

// Set pixel format to Bayer GR 8
Camera.PixelFormat.SetValue( PixelFormat_BayerGR8 );

// Set pixel format to Bayer GR 10
Camera.PixelFormat.SetValue( PixelFormat_BayerGR10 );

// Set pixel format to Bayer GR 12
Camera.PixelFormat.SetValue( PixelFormat_BayerGR12 );
```

You can also use the Basler pylon Viewer application to easily set the parameter.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Pixel Format Using Direct Register Access

To set the sensor pixel format via direct register access:

Set the value of the Pixel Format register to Mono 8 or Mono 12 on monochrome cameras or to Basler GR 8 or Basler GR 12 on color cameras.

For more information about setting parameters via direct register access, see Section 5 on page 61.

8.2 Camera Link Tap Geometry

After each image exposure is complete, the acquired image data is read out of the camera's imaging sensor and into an image buffer in the camera.
The **Camera Link Tap Geometry** parameter determines **how** the **data** in the image buffer will be **transmitted from the camera to the frame grabber** in your host computer via the Camera Link interface.

There are three tap geometry settings available:

- 1X-1Y,
- 1X2-1Y, and
- 1X-2YE.

This section describes the basics of Camera Link tap geometry. For more complete tap geometry details, refer to the document called *aviator Information for Frame Grabber Designers* (AW000831).

Basics of the 1X-1Y Tap Geometry

When a camera is set for the 1X-1Y tap geometry, the data for one pixel are transmitted on each cycle of the Camera Link pixel clock (this is usually referred to as a "one tap" configuration"). With the 1X-1Y geometry, the pixel data are transmitted from the image memory to the frame grabber in the following order:

- When image data transmission begins, on the first cycle of the camera's pixel clock the data for pixel 1 in line 1 of the image are transmitted.
- On the next cycle of the pixel clock, the data for pixel 2 in line 1 are transmitted.
- On the next cycle of the pixel clock, the data for pixel 3 in line 1 are transmitted.
- This process continues, i.e., transmitting the data for one pixel on each clock cycle, until all of the pixel data for line 1 have been transmitted.
- Once the pixel data for line 1 have all been transmitted, the camera begins transmitting pixel data for line 2. The data is transmitted in a similar fashion to line 1, i.e., one pixel at a time in ascending order.
- Once the pixel data for line 2 has been transmitted, the camera continues transmitting the pixel data for the remaining lines in ascending order.

This transmission scheme is shown graphically in Figure 42.

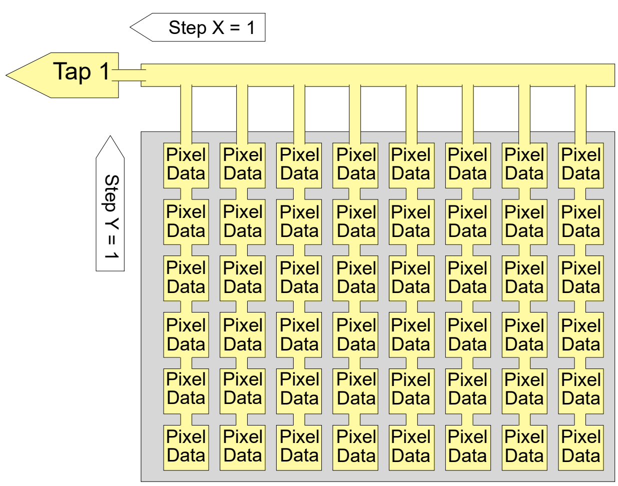


Fig. 42: 1X-1Y Tap Geometry

The **advantage** of selecting the 1X-1Y tap geometry is that almost all commonly available Camera Link frame grabbers can accept image data transmitted this format using a standard frame grabber configuration file.

The **disadvantage** of selecting the 1X-1Y tap geometry is that when an acquired image is being read out of the imaging sensor and into the camera's image buffer, the entire image must be read into the buffer before pixel data transmission can begin. This leads to significant latency between the point where an image is acquired and the point where the image data begins to be transmitted to the frame grabber in the computer.

Basics of the 1X2-1Y Tap Geometry

When a camera is set for the 1X2-1Y tap geometry, the data for two pixels are transmitted on each cycle of the Camera Link pixel clock (this is usually referred to as a "two tap" configuration"). With the 1X2-1Y geometry, the pixel data are transmitted from the image memory to the frame grabber in the following order:

- When image data transmission begins, on the first cycle of the camera's pixel clock the data for pixel 1 and pixel 2 in line 1 of the image are transmitted.
- On the next cycle of the pixel clock, the data for pixel 3 and pixel 4 in line 1 are transmitted.
- On the next cycle of the pixel clock, the data for pixel 5 and pixel 6 in line 1 are transmitted.
- This process continues, i.e., transmitting the data for two pixels on each clock cycle, until all of the pixel data for line one have been transmitted.
- Once the pixel data for line 1 have all been transmitted, the camera begins transmitting pixel data for line 2. The data is transmitted in a similar fashion to line 1, i.e., two pixels at a time in ascending order.
- Once the pixel data for line 2 has been transmitted, the camera continues transmitting the pixel data for the remaining lines in ascending order.

This transmission scheme is shown graphically in Figure 43.

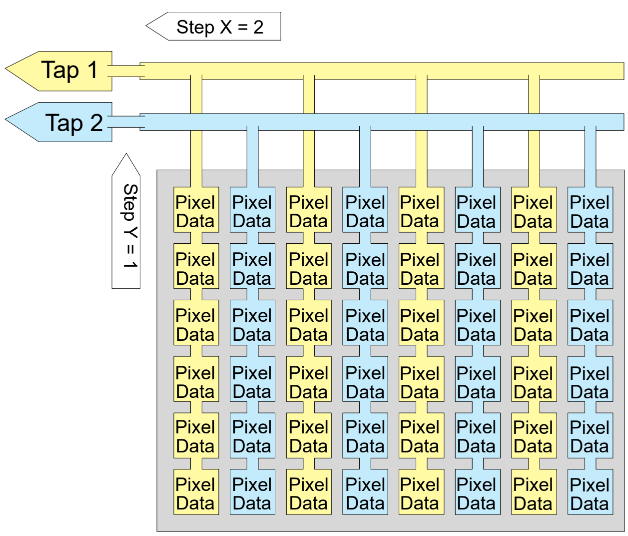


Fig. 43: 1X2-1Y Tap Geometry

The **advantage** of selecting the 1X2-1Y tap geometry is that almost all commonly available Camera Link frame grabbers can accept image data transmitted this format using a standard frame grabber configuration file.

The **disadvantage** of selecting the 1X2-1Y tap geometry is that when an acquired image is being read out of the imaging sensor and into the camera's image buffer, the entire image must be read into the buffer before pixel data transmission can begin. This leads to significant latency between the point where an image is acquired and the point where the image data begins to be transmitted to the frame grabber in the computer.

Basics of the 1X-2YE Tap Geometry

When a camera is set for the 1X-2YE tap geometry, the data for two pixels are transmitted on each cycle of the Camera Link pixel clock (this is usually referred to as a "two tap" configuration". With the 1X-2YE geometry, the pixel data are transmitted from the image memory to the frame grabber in the following order:

- When image transmission begins, on the first cycle of the camera's pixel clock the data for pixel 1 in the first line of the image and for pixel 1 in the last line of the image (line H) are transmitted (H = height = the last line in the image)
- On the next cycle of the pixel clock, the data for pixel 2 in the first line and for pixel 2 in line H are transmitted.
- On the next cycle of the pixel clock, the data for pixel 3 in line 1 and for pixel 3 in line H are transmitted.
- This process continues, i.e., transmitting the data for two pixels on each clock cycle, until all of the pixel data for line 1 and line H have been transmitted.
- Once the pixel data for line 1 and line H have all been transmitted, the camera begins transmitting pixel data for line 2 and for line H-1 (the next to the last line). The data is transmitted in a similar fashion to line 1 and line H, i.e., two pixels at a time in ascending order.
- The camera continues transmitting the pixel data for the remaining lines in similar fashion.

This transmission scheme is shown graphically in Figure 44.

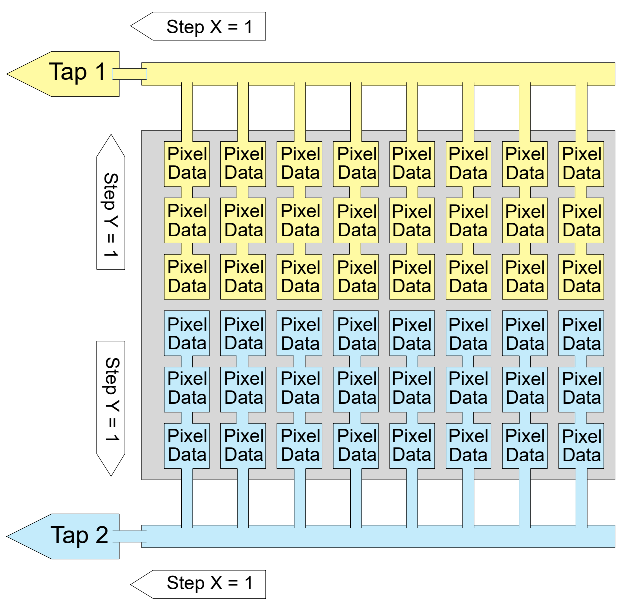


Fig. 44: 1X-2YE Tap Geometry

The **advantage** of selecting the 1X-2YE tap geometry is that when an acquired image is being read out of the imaging sensor and into the camera's image buffer, pixel data transmission can begin before the entire image is read out and buffered. This means that the latency between the point where an image is acquired and the point where the image data begins to be transmitted to the frame grabber in the computer is very low.

The **disadvantage** of selecting the 1X-2YE tap geometry is that Camera Link frame grabbers must typically use a specifically prepared frame grabber configuration file to be able to accept image data transmitted in this fashion. Camera users should contact the manufacturer of their frame grabber to obtain the necessary configuration file.

ℹ	The 1X-2YE tap geometry **can only be used** when the camera is **set for four tap sensor digitization**. For more information about setting the tap digitization, see Section 9.1 on page 148.

Setting the Tap Geometry Using Basler pylon

You can use the pylon API to set the Camera Link tap geometry from within your application software. The following code snippet illustrates using the pylon API to set the tap geometry:

```
// Set the tap geometry to 1X-1Y
Camera.ClTapGeometry.SetValue( ClTapGeometry_Geometry1X_1Y );

// Set the tap geometry to 1X2-1Y
Camera.ClTapGeometry.SetValue( ClTapGeometry_Geometry1X2_1Y );

// Set the tap geometry to 1X-2YE
Camera.ClTapGeometry.SetValue( ClTapGeometry_Geometry1X-2YE );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Tap Geometry Using Basler pylon

To set the Camera Link tap geometry via direct register access:

Set the value of the CL Tap Geometry register for 1X-1Y, 1X2-1Y or 1X-2YE.

For more information about setting parameters via direct register access, see Section 5 on page 61.

9 Features

This chapter provides detailed information about the standard features available on each camera. It also includes an explanation of their operation and the parameters associated with each feature.

9.1 Sensor Digitization Taps

The camera can be set to four tap or to one tap sensor digitization.

- With **four tap digitization**, **four separate electronic circuits** (taps) are used to read out pixel values from the sensor after a frame has been exposed. The main advantage of four tap mode is that it reads out the sensor very rapidly and can result in higher maximum allowed frame acquisition rates.
- With **one tap digitization**, a **single electronic circuit** (tap) is used to read out pixel values from the sensor after a frame has been exposed. The main advantage of one tap mode is that it can produce a more uniform image. A disadvantage is that in single tap mode, the maximum allowed frame acquisition rate will be significantly reduced.

For more information about sensor digitization taps, see Section 4 on page 54.

Setting the Sensor Digitization Taps Using Basler pylon

You can use the pylon API to set the sensor digitization taps from within your application software. The following code snippet illustrates using the pylon API to set the exposure mode:

```
// Set the digitization taps to 4
Camera.SensorDigitizationTaps.SetValue( SensorDigitizationTaps_Four);

// Set the digitization taps to 1
Camera.SensorDigitizationTaps.SetValue( SensorDigitizationTaps_One );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Sensor Digitization Taps Using Direct Register Access

To set the sensor digitization taps via direct register access:

- Set the value of the Sensor Digitization Taps register for one tap or four tap digitization.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.2 Camera Link Pixel Clock Speed

The camera features selectable Camera Link pixel clock speeds. The pixel clock speed determines the rate at which pixel data will be transmitted from the camera to the frame grabber in your computer via the Camera Link interface. The available pixel clock speeds are: 20 MHz, 32.5 MHz, 40 MHz, 48 MHz, and 65 MHz. The default clock speed is 65 MHz.

Setting the camera for a higher pixel clock speed will increase the rate at which image data is transferred from the camera to the frame grabber. Some frame grabbers, however, cannot operate at the higher clock speeds. So it is important that you determine the maximum clock speed that your frame grabber can handle and that you set the camera's speed no higher than the frame grabber's maximum.

Keep in mind that if you set the camera for one of the lower pixel clock speeds, it may limit the camera's maximum allowed frame acquisition rate.

If you change the clock speed while the camera is in the process of acquiring images:

- Triggering and image acquisition will stop.
- Any acquired image that is already in the camera's image buffer will be delivered.
- The camera's clock speed will be changed internally.
- Once the change is complete, triggering and image acquisition will resume.

The Camera Link clock speed setting will be stored in the camera's configuration sets. This means, for example, that if you have a different clock speed setting stored in user set 1 and user set 2 and you change the active set from user set 1 to user set 2, the clock speed will change.

For more information about

- the maximum allowed frame aquisition rate, see Section 6.8 on page 112.
- configuration sets, see Section 9.19 on page 256.

Setting the Camera Link Pixel Clock Using Basler pylon

You can use the pylon API to set the Camera Link pixel clock speed from within your application software. The following code snippet illustrates using the pylon API to set the clock speed:

```
// Set the Camera Link pixel clock speed 32.5
Camera.ClPixelClock.SetValue( ClPixelClock_PixelClock32_5 );

// Set the Camera Link pixel clock speed to 48
Camera.ClPixelClock.SetValue( ClPixelClock_PixelClock48 );
```

> You can use the pylon API to set the pixel clock speed to 20, 32.5, 40, 48, or 65 MHz. These are the only valid values for the pixel clock speed. If you attempt to use the pylon API to set the clock speed to a value other then these, the camera will automatically round the setting down to the nearest valid speed.

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Camera Link Pixel Clock Using Direct Register Access

To set the Camera Link pixel clock speed via direct register access:

Set the value of the CL Pixel Clock register for 20 MHz, 32.5 MHz, 40 MHz, 48 MHz, or 65 MHz.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.3 Gain

ℹ	This section (Section 9.3) describes the basic theory of gain and how gain can be adjusted "manually", i.e., by setting the value of the individual gain related parameters. The camera also has a Gain Auto function that can automatically adjust the gain. **Manual adjustment of the gain parameters will only work correctly if the Gain Auto function is disabled.** For more information about ■ auto functions in general, see Section 9.8 on page 173. ■ the Gain Auto function in particular, see Section 9.8.5 on page 182. Note that on cameras delivered from the factory, the exposure auto function will become enabled whenever you power on or reset the camera. This behavior happens due to the way that the camera's configuration sets are set in the factory. More specifically, it happens because the auto functions factory setup is selected as the default configuration set and the default configuration set is designated as the "startup" set. To change this behavior, you must change the configuration set settings. For more information about configuration sets, see Section 9.19 on page 256.

The camera's gain setting is adjustable. As shown in Figure 45, increasing the gain increases the slope of the response curve for the camera. This results in a higher gray value output from the camera for a given amount of output from the imaging sensor. Decreasing the gain decreases the slope of the response curve and results in a lower gray value for a given amount of sensor output.

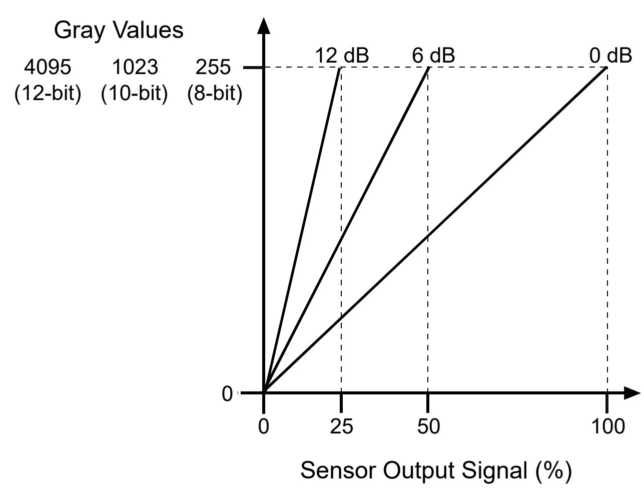


Fig. 45: Gain in dB

Increasing the gain is useful when at your brightest exposure, a gray value lower than 255 (in modes that output 8 bits per pixel), 1023 (in modes that output 10 bits per pixel), or 4095 (in modes that output 12 bits per pixels) is reached. For example, if you found that at your brightest exposure the gray values output by the camera were no higher than 127 (in an 8 bit mode), you could increase the gain to 6 dB (an amplification factor of 2) and thus reach gray values of 254.

9.3.1 Gain with Four Tap Sensor Digitization

When the camera is set for four tap sensor digitization, the imaging sensor is divided into four quadrants for readout purposes. Each quadrant is read out by a separate tap (electronic circuit). As a result of this design, there are five gain parameters available: Gain All, Gain Tap 1, and Gain Tap 2, Gain Tap 3, and Gain Tap 4.

Gain All is a global adjustment, i.e., its setting affects all four quadrants of the sensor.

Gain Tap 1 sets an additional amount of gain for the top left quadrant of the sensor. The total gain for the top left quadrant will be the sum of the Gain All value plus the Gain Tap 1 value.

Gain Tap 2 sets an additional amount of gain for the top right quadrant of the sensor. The total gain for the top right quadrant will be the sum of the Gain All value plus the Gain Tap 2 value.

Gain Tap 3 sets an additional amount of gain for the bottom left quadrant of the sensor. The total gain for the bottom left quadrant will be the sum of the Gain All value plus the Gain Tap 3 value.

Gain Tap 4 sets an additional amount of gain for the bottom right quadrant of the sensor. The total gain for the bottom right quadrant will be the sum of the Gain All value plus the Gain Tap 4 value.

The settings for the gain parameters must adhere to the following limits:

- The Gain All parameter value can be set in a range from 0 to 600.
- The Gain Tap 1, Gain Tap 2, Gain Tap 3, or Gain Tap 4 parameters values can each be set in a range from 0 to 600.
- The sum of the Gain All setting plus
  - the Gain Tap 1 setting must be between 0 and 600 (inclusive).
  - the Gain Tap 2 setting must be between 0 and 600 (inclusive).
  - the Gain Tap 3 setting must be between 0 and 600 (inclusive).
  - the Gain Tap 4 setting must be between 0 and 600 (inclusive).

i	For normal operation, we strongly recommend that you set the value of all four tap gains to the minimum and that you simply use Gain All to set the gain. Typically, the tap gains are only used if you want to adjust the gain balance between the quadrants of the sensor.

If you know the current settings for Gain All, Gain Tap 1, Gain Tap 2, Gain Tap 3, and Gain Tap 4, you can use the formula below to calculate the dB of gain that will result on each tap:

Gain on Tap N = (0.0359 x Gain All Setting) + (0.0359 x Gain Tap N Setting)

Where N is 1, 2, 3, or 4

For example, assume that you have set the Gain All to 450 and the tap 1 gain to 0. The gain on tap 1 would be:

Gain on Tap 1 = (0.0359 x 450) + (0.0359 x 0) = 16.2 dB

Setting the Gain Using Basler pylon

When the camera is set to four tap digitization mode, setting the gain with Basler pylon is a several step process:

To set the Gain All parameter value:

- Set the Gain Selector to All.
- Set the Gain Raw parameter to your desired value.

To set the Gain Tap 1, Gain Tap 2, Gain Tap 3, or Gain Tap 4 parameter values:

- Set the Gain Selector to Tap 1, Tap 2, Tap 3, or Tap 4.
- Set the Gain Raw parameter to your desired value.

You can use the pylon API to set the Gain Selector and the Gain Raw parameter values from within your application software. The following code snippet illustrates using the pylon API to set the selector and the parameter value:

```
// Set Gain All
Camera.GainSelector.SetValue( GainSelector_All );
Camera.GainRaw.SetValue( 450 );

// Set Gain Tap 1
Camera.GainSelector.SetValue( GainSelector_Tap1 );
Camera.GainRaw.SetValue( 0 );

// Set Gain Tap 2
Camera.GainSelector.SetValue( GainSelector_Tap2 );
Camera.GainRaw.SetValue( 0 );

// Set Gain Tap 3
Camera.GainSelector.SetValue( GainSelector_Tap3 );
Camera.GainRaw.SetValue( 0 );

// Set Gain Tap 4
Camera.GainSelector.SetValue( GainSelector_Tap4 );
Camera.GainRaw.SetValue( 0 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Gain Using Direct Register Access

Setting the gain via direct register access is a several step process:

- Set the value of the Gain All register.
- Set the value of the Gain Tap 1 register.
- Set the value of the Gain Tap 2 register.
- Set the value of the Gain Tap 3 register.
- Set the value of the Gain Tap 4 register.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.3.2 Gain with One Tap Sensor Digitization

When the camera is set for one tap sensor digitization, a single tap (electronic circuit) is used to read out the sensor. As a result of this design, there is one gain parameter available when the camera is set for one tap digitization: Gain All.

The minimum, maximum, and adjustment methods for this parameter are similar to those described in the previous section.

9.4 Black Level

9.4.1 Black Level with Four Tap Sensor Digitization

Adjusting the camera's black level will result in an offset to the pixel values output from the camera.

As mentioned in the "Functional Description" section of this manual, when the camera is set for four tap sensor digitization, the imaging sensor is divided into four quadrants for readout purposes. Each quadrant is read out by a separate tap (electronic circuit). As a result of this design, there are five black level parameters available: Black Level All, Black Level Tap 1, and Black Level Tap 2, Black Level Tap 3, and Black Level Tap 4.

Black Level All is a global adjustment, i.e., its setting affects all four quadrants of the sensor.

Black Level Tap 1 sets an additional amount of black level for the top left quadrant of the sensor. The total black level for the top left quadrant will be the sum of the Black Level All value plus the Black Level Tap 1 value.

Black Level Tap 2 sets an additional amount of black level for the top right quadrant of the sensor. The total black level for the top right quadrant will be the sum of the Black Level All value plus the Black Level Tap 2 value.

Black Level Tap 3 sets an additional amount of black level for the bottom left quadrant of the sensor. The total black level for the bottom left quadrant will be the sum of the Black Level All value plus the Black Level Tap 3 value.

Black Level Tap 4 sets an additional amount of black level for the bottom right quadrant of the sensor. The total black level for the bottom right quadrant will be the sum of the Black Level All value plus the Black Level Tap 4 value.

If the camera is set for

- **8 bit pixel depth**, an increase of 64 in a black level setting will result in a positive offset of 1 in the pixel values output from the camera. And a decrease of 64 in a black level setting result in a negative offset of 1 in the pixel values output from the camera.
- **10 bit pixel depth**, an increase of 16 in a black level setting will result in a positive offset of 1 in the pixel values output from the camera. And a decrease of 16 in a black level setting result in a negative offset of 1 in the pixel values output from the camera.
- **12 bit pixel depth**, an increase of 4 in a black level setting will result in a positive offset of 1 in the pixel values output from the camera. A decrease of 4 in a black level setting will result in a negative offset of 1 in the pixel values output from the camera.

When adjusting the black levels, the following guidelines must be met:

- The sum of the Black Level All plus the Black Level Tap 1 parameter settings must be less than or equal to 950.
- The sum of the Black Level All plus the Black Level Tap 2 parameter settings must be less than or equal to 950.
- The sum of the Black Level All plus the Black Level Tap 3 parameter settings must be less than or equal to 950.
- The sum of the Black Level All plus the Black Level Tap 4 parameter settings must be less than or equal to 950.

i	For normal operation, we recommend that you set the value of the tap black levels to zero and that you simply use Black Level All to set the black level. Typically, the tap black level settings are only used if you want to adjust the black level balance between the quadrants of the sensor.

Setting the Black Level Using Basler pylon

When the camera is set for four tap digitization mode, setting the black level with Basler pylon is a several step process:

To set the Black Level All parameter value:

- Set the Black Level Selector to All.
- Set the Black Level Raw parameter to your desired value.

To set the Black Level Tap 1, Black Level Tap 2, Black Level Tap 3, or Black Level Tap 4 parameter value:

- Set the Black Level Selector to Tap 1, Tap 2, Tap 3, or Tap 4.
- Set the Black Level Raw parameter to your desired value.

You can use the pylon API to set the Black Level Selector and the Black Level Raw parameter values from within your application software. The following code snippet illustrates using the pylon API to set the selector and the parameter value:

```
// Set Black Level All
Camera.BlackLevelSelector.SetValue ( BlackLevelSelector_All );
Camera.BlackLevelRaw.SetValue( 64 );

// Set Black Level Raw Tap 1
Camera.BlackLevelSelector.SetValue ( BlackLevelSelector_Tap1 );
Camera.BlackLevelRaw.SetValue( 0 );

// Set Black Level Raw Tap 2
```

```
Camera.BlackLevelSelector.SetValue ( BlackLevelSelector_Tap2 );
Camera.BlackLevelRaw.SetValue( 0 );

// Set Black Level Raw Tap 3
Camera.BlackLevelSelector.SetValue ( BlackLevelSelector_Tap3 );
Camera.BlackLevelRaw.SetValue( 0 );

// Set Black Level Raw Tap 4
Camera.BlackLevelSelector.SetValue ( BlackLevelSelector_Tap4 );
Camera.BlackLevelRaw.SetValue( 0 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Black Level Using Direct Register Access

Setting the black level via direct register access is a several step process:

- Set the value of the Black Level All register.
- Set the value of the Black Level Tap 1 register.
- Set the value of the Black Level Tap 2 register.
- Set the value of the Black Level Tap 3 register.
- Set the value of the Black Level Tap 4 register.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.4.2 Black Level with One Tap Sensor Digitization

When the camera is set for one tap sensor digitization, a single tap (electronic circuit) is used to read out the sensor. As a result of this design, there is one black level parameter available when the camera is set for one tap digitization: Black Level All.

The minimums, maximums, and adjustment methods for this parameter are similar to those described in the previous section.

9.5 Remove Parameter Limits

For each camera parameter, the allowed range of parameter values is normally limited. The factory limits are designed to ensure optimum camera operation and, in particular, good image quality. For special camera uses, however, it may be helpful to set parameter values outside of the factory limits.

The remove parameter limits feature lets you remove the factory parameter limits for certain parameters. When the factory parameter limits are disabled, the parameter values can be set within extended limits. Typically, the range of the extended limits is dictated by the physical restrictions of the camera's electronic devices, such as the absolute limits of the camera's variable gain control.

i	Currently, the parameter limits can only be removed on the gain and the prelines features.

Removing the Parameter Limits Using Basler pylon

Removing the limits for a parameter using Basler pylon is a two step process:

- Use the Parameter Selector to select the parameter whose limits you wish to disable.
- Set the value of the Remove Limits parameter.

You can use the pylon API to set the Parameter Selector and the value of the Remove Limits parameter from within your application software. The following code snippet illustrates using the pylon API to set the selector and the parameter value:

```
// Select the Gain parameter
Camera.ParameterSelector.SetValue( ParameterSelector_Gain );

// Remove the factory limits for the selected parameter (Gain)
Camera.RemoveLimits.SetValue( true );
```

You can also use the Basler pylon Viewer application to easily set the parameters. Note that the remove parameter limits feature will only be available at the "guru" viewing level.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Removing Parameter Limits Using Direct Register Access

To remove the limits for the Gain parameter using direct register access:

- Set the value of the Remove Parameter Limits Gain register to enabled.

To remove the limits for the Prelines parameter using direct register access:

- Set the value of the Remove Parameter Limits Prelines register.

For more information about changing settings via direct register access, see Section 5 on page 61.

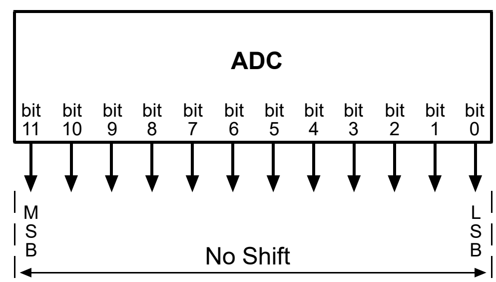
9.6 Digital Shift

The digital shift feature lets you change the group of bits that is output from each ADC in the camera. Using the digital shift feature will effectively multiply the output of the camera by 2 times, 4 times, 8 times, or 16 times. The next two sections describe how the digital shift feature works when the camera is set for a 12 bit pixel format and when it is set for a 8 bit pixel format. There is also a section describing precautions that you must observe when using the digital shift feature and a section that describes enabling and setting the digital shift feature.

9.6.1 Digital Shift with 12 Bit Pixel Formats

No Shift

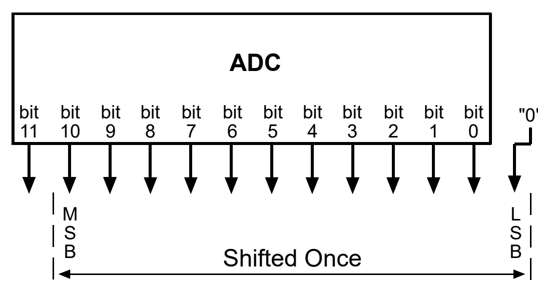
As mentioned in the Functional Description section of this manual, the camera uses 12 bit ADCs to digitize the output from the imaging sensor. When the camera is set for a pixel format that outputs pixel data at 12 bit effective depth, by default, the camera transmits the 12 bits that are output from each ADC.



Shift by 1

When the camera is set to shift by 1, the output from the camera will include bit 10 through bit 0 from each ADC along with a zero as an LSB.

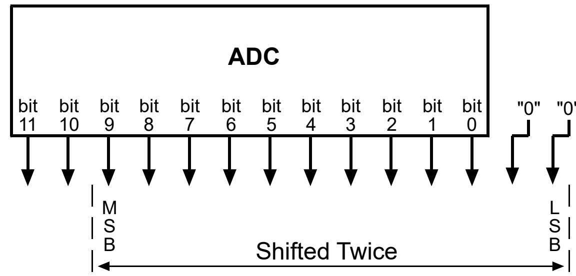
The result of shifting once is that the output of the camera is effectively multiplied by 2. For example, assume that the camera is set for no shift, that it is viewing a uniform white target, and that under these conditions the reading for the brightest pixel is 100. If you changed the digital shift setting to shift by 1, the reading would increase to 200.



When the camera is set to shift by 1, the least significant bit output from the camera for each pixel value will be 0. This means that no odd gray values can be output and that the gray value scale will only include values of 2, 4, 6, 8, 10, and so on. This absence of some gray values is commonly referred to as "missing codes".

If the pixel values being output by the camera’s sensor are high enough to set bit 11 to 1, we recommend not using shift by 1. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 1 setting when your pixel readings with a 12 bit pixel format selected and with digital shift disabled are all less than 2048.

Shift by 2



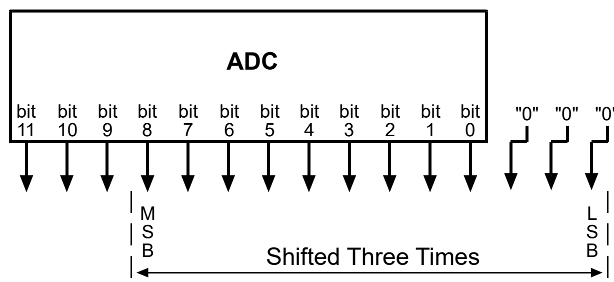
When the camera is set to shift by 2, the output from the camera will include bit 9 through bit 0 from each ADC along with 2 zeros as LSBs.

The result of shifting twice is that the output of the camera is effectively multiplied by 4.

When the camera is set to shift by 2, the 2 least significant bits output from the camera for each pixel value will be 0. This means that the gray value scale will only include every 4th gray value, for example, 4, 8, 12, 16, 20, and so on.

If the pixel values being output by the camera's sensor are high enough to set bit 10 or bit 11 to 1, we recommend not using shift by 2. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 2 setting when your pixel readings with a 12 bit pixel format selected and with digital shift disabled are all less than 1024.

Shift By 3



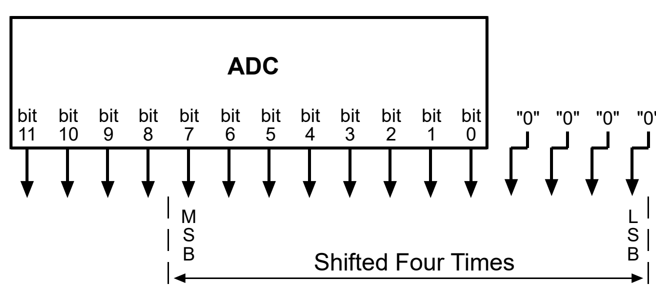
When the camera is set to shift by 3, the output from the camera will include bit 8 through bit 0 from each ADC along with 3 zeros as LSBs.

The result of shifting 3 times is that the output of the camera is effectively multiplied by 8.

When the camera is set to shift by 3, the 3 least significant bits output from the camera for each pixel value will be 0. This means that the gray value scale will only include every 8th gray value, for example, 8, 16, 24, 32, and so on.

If the pixel values being output by the camera's sensor are high enough to set bit 9, bit 10, or bit 11 to 1, we recommend not using shift by 3. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 3 setting when your pixel readings with a 12 bit pixel format selected and with digital shift disabled are all less than 512.

Shift By 4



When the camera is set to shift by 4, the output from the camera will include bit 7 through bit 0 from each ADC along with 4 zeros as LSBs.

The result of shifting 4 times is that the output of the camera is effectively multiplied by 16.

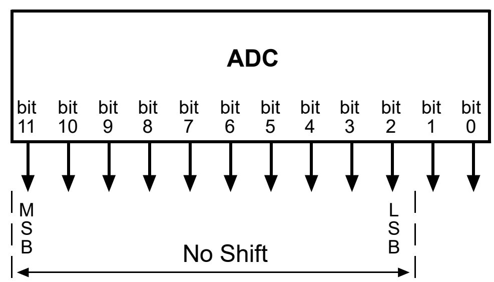
When the camera is set to shift by 4, the 4 least significant bits output from the camera for each pixel value will be 0. This means that the gray value scale will only include every 16th gray value, for example, 16, 32, 48, 64, and so on.

If the pixel values being output by the camera's sensor are high enough to set bit 8, bit 9, bit 10, or bit 11 to 1, we recommend not using shift by 4. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 4 setting when your pixel readings with a 12 bit pixel format selected and with digital shift disabled are all less than 256.

9.6.2 Digital Shift With 10 Bit Pixel Formats

No Shift

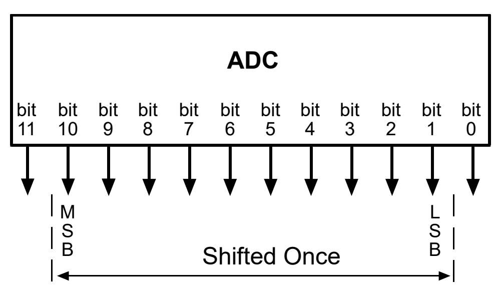
As mentioned in the Functional Description section of this manual, the camera uses 12 bit ADCs to digitize the output from the imaging sensor. When the camera is set for a pixel format that outputs pixel data at 10 bit effective depth, by default, the camera drops the two least significant bits from each ADC and transmits the 10 most significant bits (bits 11 through 2).



Shift by 1

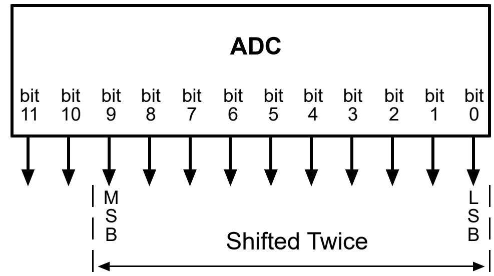
When the camera is set to shift by 1, the output from the camera will include bit 10 through bit 1 from each ADC.

The result of shifting once is that the output of the camera is effectively multiplied by 2. For example, assume that the camera is set for no shift, that it is viewing a uniform white target, and that under these conditions the reading for the brightest pixel is 100. If you changed the digital shift setting to shift by 1, the reading would increase to 200.



If the pixel values being output by the camera's sensor are high enough to set bit 11 to 1, we recommend not using shift by 1. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 1 setting when your pixel readings with a 10 bit pixel format selected and with digital shift disabled are all less than 512.

Shift by 2

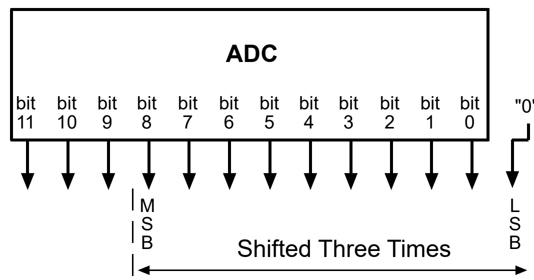


When the camera is set to shift by 2, the output from the camera will include bit 9 through bit 0 from each ADC.

The result of shifting twice is that the output of the camera is effectively multiplied by 4.

If the pixel values being output by the camera's sensor are high enough to set bit 10 or bit 11 to 1, we recommend not using shift by 2. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 2 setting when your pixel readings with a 10 bit pixel format selected and with digital shift disabled are all less than 256.

Shift By 3



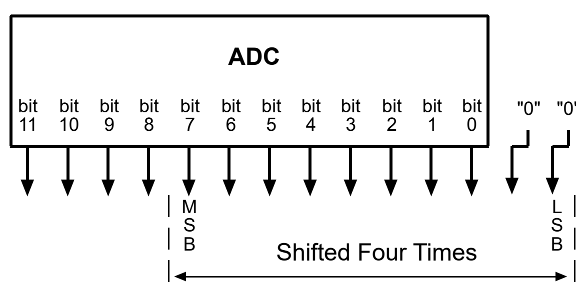
When the camera is set to shift by 3, the output from the camera will include bit 8 through bit 0 from each ADC along with a zero as an LSB.

The result of shifting 3 times is that the output of the camera is effectively multiplied by 8.

When the camera is set to shift by 3, the least significant bit output from the camera for each pixel value will be 0. This means that no odd gray values can be output and that the gray value scale will only include values of 2, 4, 6, 8, 10, and so on. This absence of some gray values is commonly referred to as "missing codes".

If the pixel values being output by the camera's sensor are high enough to set bit 9, bit 10, or bit 11 to 1, we recommend not using shift by 3. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 3 setting when your pixel readings with a 10 bit pixel format selected and with digital shift disabled are all less than 128.

Shift By 4



When the camera is set to shift by 4, the output from the camera will include bit 7 through bit 0 from each ADC along with 2 zeros as LSBs.

The result of shifting 4 times is that the output of the camera is effectively multiplied by 16.

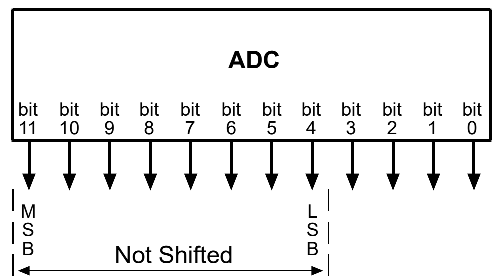
When the camera is set to shift by 4, the 2 least significant bits output from the camera for each pixel value will be 0. This means that the gray value scale will only include every 4th gray value, for example, 4, 8, 16, 20, and so on.

If the pixel values being output by the camera's sensor are high enough to set bit 8, bit 9, bit 10, or bit 11 to 1, we recommend not using shift by 4. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 4 setting when your pixel readings with a 10 bit pixel format selected and with digital shift disabled are all less than 64.

9.6.3 Digital Shift With 8 Bit Pixel Formats

No Shift

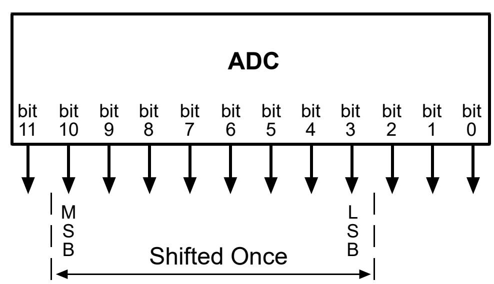
As mentioned in the Functional Description section of this manual, the camera uses 12 bit ADCs to digitize the output from the imaging sensor. When the camera is set for a pixel format that outputs pixel data at 8 bit effective depth, by default, the camera drops the 4 least significant bits from each ADC and transmits the 8 most significant bits (bit 11 through 4).



Shift by 1

When the camera is set to shift by 1, the output from the camera will include bit 10 through bit 3 from each ADC.

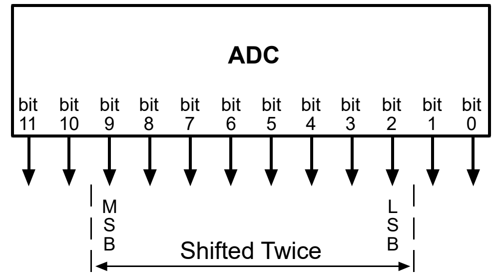
The result of shifting once is that the output of the camera is effectively multiplied by 2. For example, assume that the camera is set for no shift, that it is viewing a uniform white target, and that under these conditions the reading for the brightest pixel is 10. If you changed the digital shift setting to shift by 1, the reading would increase to 20.



If the pixel values being output by the camera's sensor are high enough to set bit 11 to 1, we recommend not using shift by 1. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 1 setting when your pixel readings with an 8 bit pixel format selected and with digital shift disabled are all less than 128.

Shift by 2

When the camera is set to shift by 2, the output from the camera will include bit 9 through bit 2 from each ADC.

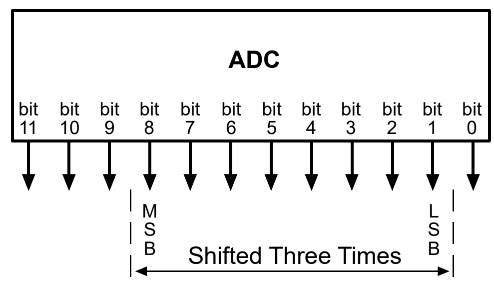


The result of shifting twice is that the output of the camera is effectively multiplied by 4.

If the pixel values being output by the camera's sensor are high enough to set bit 10 or bit 11 to 1, we recommend not using shift by 2. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 2 setting when your pixel readings with an 8 bit pixel format selected and with digital shift disabled are all less than 64.

Shift by 3

When the camera is set to shift by 3, the output from the camera will include bit 8 through bit 1 from each ADC.

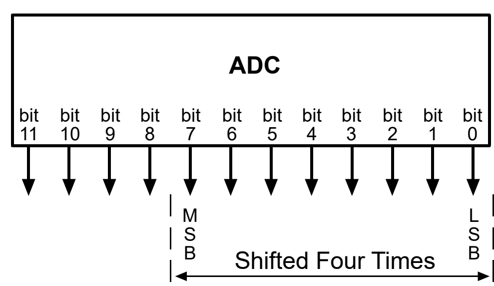


The result of shifting three times is that the output of the camera is effectively multiplied by 8.

If the pixel values being output by the camera's sensor are high enough to set bit 9, bit 10, or bit 11 to 1, we recommend not using shift by 3. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 3 setting when your pixel readings with an 8 bit pixel format selected and with digital shift disabled are all less than 32.

Shift by 4

When the camera is set to shift by 4, the output from the camera will include bit 7 through bit 0 from each ADC.



The result of shifting four times is that the output of the camera is effectively multiplied by 16.

If the pixel values being output by the camera's sensor are high enough to set bit 8, bit 9, bit 10, or bit 11 to 1, we recommend not using shift by 4. If you do nonetheless, all bits output from the camera will automatically be set to 1. Therefore, you should only use the shift by 4 setting when your pixel readings with an 8 bit pixel format selected and with digital shift disabled are all less than 16.

9.6.4 Precautions When Using Digital Shift

There are several checks and precautions that you must follow before using the digital shift feature. The checks and precautions differ depending on whether the camera will be set for a 12 bit pixel format, a 10 bit pixel format, or an 8 bit pixel format in your application.

If you will be using a 12 bit pixel format, make this check:

Set the camera for a 12 bit pixel format and **no digital shift**.

Check the output of the camera under your normal lighting conditions and note the readings for the brightest pixels.

- If any of the readings are above 2048, do not use digital shift.
- If all of the readings are below 2048, you can safely use the shift by 1 setting.
- If all of the readings are below 1024, you can safely use the shift by 1 or 2 settings.
- If all of the readings are below 512, you can safely use the shift by 1, 2, or 3 settings.
- If all of the readings are below 256, you can safely use the shift by 1, 2, 3, or 4 settings.

If you will be using a 10 bit pixel format, make this check:

Set the camera for a 10 bit pixel format and **no digital shift**.

Check the output of the camera under your normal lighting conditions and note the readings for the brightest pixels.

- If any of the readings are above 512, do not use digital shift.
- If all of the readings are below 512, you can safely use the shift by 1 setting.
- If all of the readings are below 256, you can safely use the shift by 1 or 2 settings.
- If all of the readings are below 128, you can safely use the shift by 1, 2, or 3 settings.
- If all of the readings are below 64, you can safely use the shift by 1, 2, 3, or 4 settings.

If you will be using an 8 bit format, make this check:

Set the camera for a 8 bit pixel format and **no digital shift**.

Check the output of the camera under your normal lighting conditions and note the readings for the brightest pixels.

- If any of the readings are above 128, do not use digital shift.
- If all of the readings are below 128, you can safely use the shift by 1 setting.
- If all of the readings are below 64, you can safely use the shift by 1 or 2 settings.
- If all of the readings are below 32, you can safely use the shift by 1, 2, or 3 settings.
- If all of the readings are below 16, you can safely use the shift by 1, 2, 3, or 4 settings.

9.6.5 Enabling and Setting Digital Shift

Enabling and Setting Digital Shift Using Basler pylon

You can enable or disable the digital shift feature by setting the value of the Digital Shift parameter. When the parameter is set to zero, digital shift will be disabled. When the parameter is set to 1, 2, 3, or 4, digital shift will be set to shift by 1, shift by 2, shift by 3, or shift by 4 respectively.

You can use the pylon API to set the Digital Shift parameter values from within your application software. The following code snippet illustrates using the pylon API to set the parameter values:

```
// Disable digital shift
Camera.DigitalShift.SetValue( 0 );

// Enable digital shift by 2
Camera.DigitalShift.SetValue( 2 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Enabling and Setting Digital Shift Using Direct Register Access

To enable or disable the digital shift feature via direct register access:

- Set the value of the Digital Shift register to 0, 1, 2, 3, or 4.

When the register is set to 0, digital shift will be disabled. When the register is set to 1, 2, 3, or 4, digital shift will be set to shift by 1, shift by 2, shift by 3, or shift by 4 respectively.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.7 Image Area of Interest (AOI)

The image area of interest (AOI) feature lets you specify a portion of the imaging sensor array and after each frame is acquired, only the pixel information from the specified portion of the array will be transmitted to the host computer. Depending on the tap geometry, the position of the AOI is defined as follows:

- If the camera is set to one of the following Camera Link tap geometries, **i.e. 1X-1Y or 1X2-1Y**: The area of interest is referenced to the top left corner of the sensor array. The top left corner is designated as column 0 and row 0 as shown in Figure 46.
  The location and size of the area of interest is defined by declaring an offset X, a width, an offset Y, and a height. For example, suppose that you specify the offset X as 10, the width as 16, the offset Y as 6, and the height as 10. The area of the array that is bounded by these settings is shown in Figure 46.
  If the 1X-1Y or 1X2-1Y tap geometry is selected, the AOI feature also includes Center X and a Center Y capabilities. When Center X is enabled, the camera will automatically center the AOI along the sensor's X axis (and will disable the Offset X setting). When Center Y is enabled, the camera will automatically center the AOI along the sensor's Y axis (and will disable the Offset Y setting).

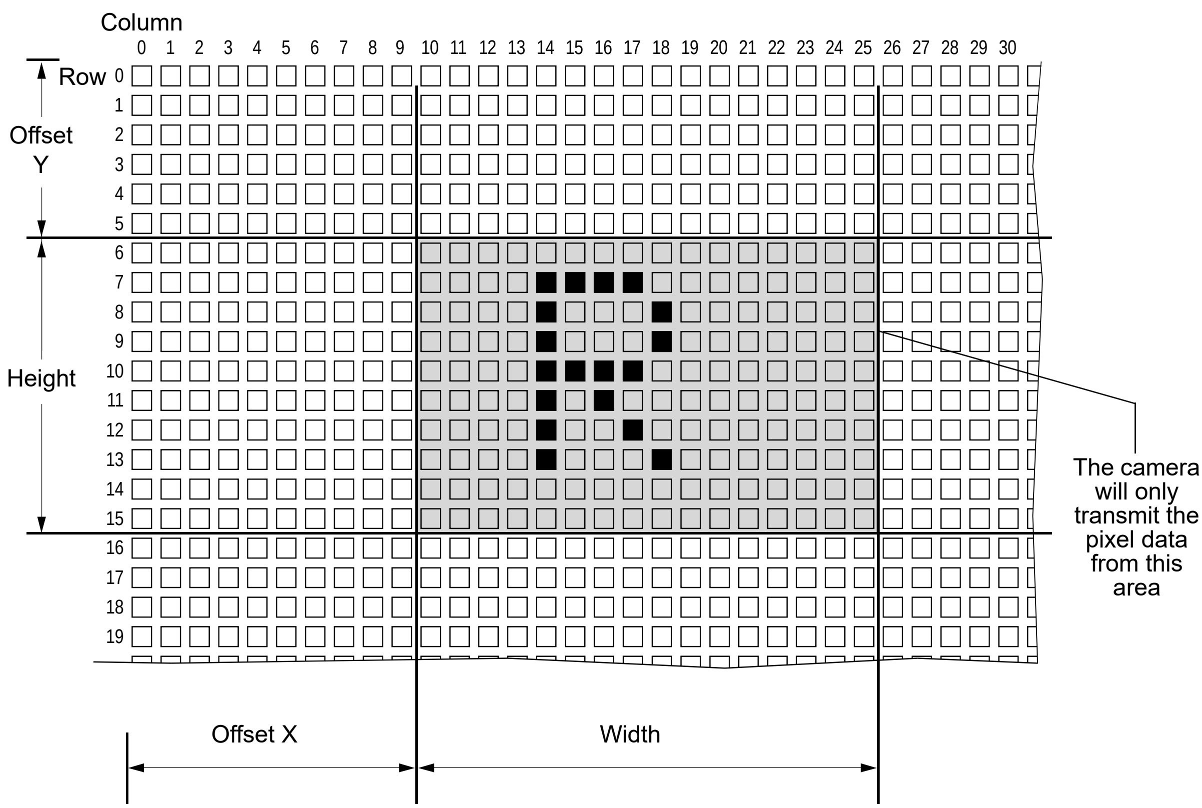


Fig. 46: Area of Interest for Tap Geometries 1X-1Y And 1X2-1Y

- If the camera is set to the Camera Link **tap geometry 1X-2YE**, the Center Y capability is disabled (see Figure 47):
  The size of the area of interest is defined by declaring a width in columns of pixels and a height

in lines of pixels. Once you set the width and height of the AOI, the camera will automatically position the AOI at the center of the sensor.

For example, suppose that you specify the width as 20 columns and the height as 16 lines. The size and position of the AOI will be as shown in Figure 47.

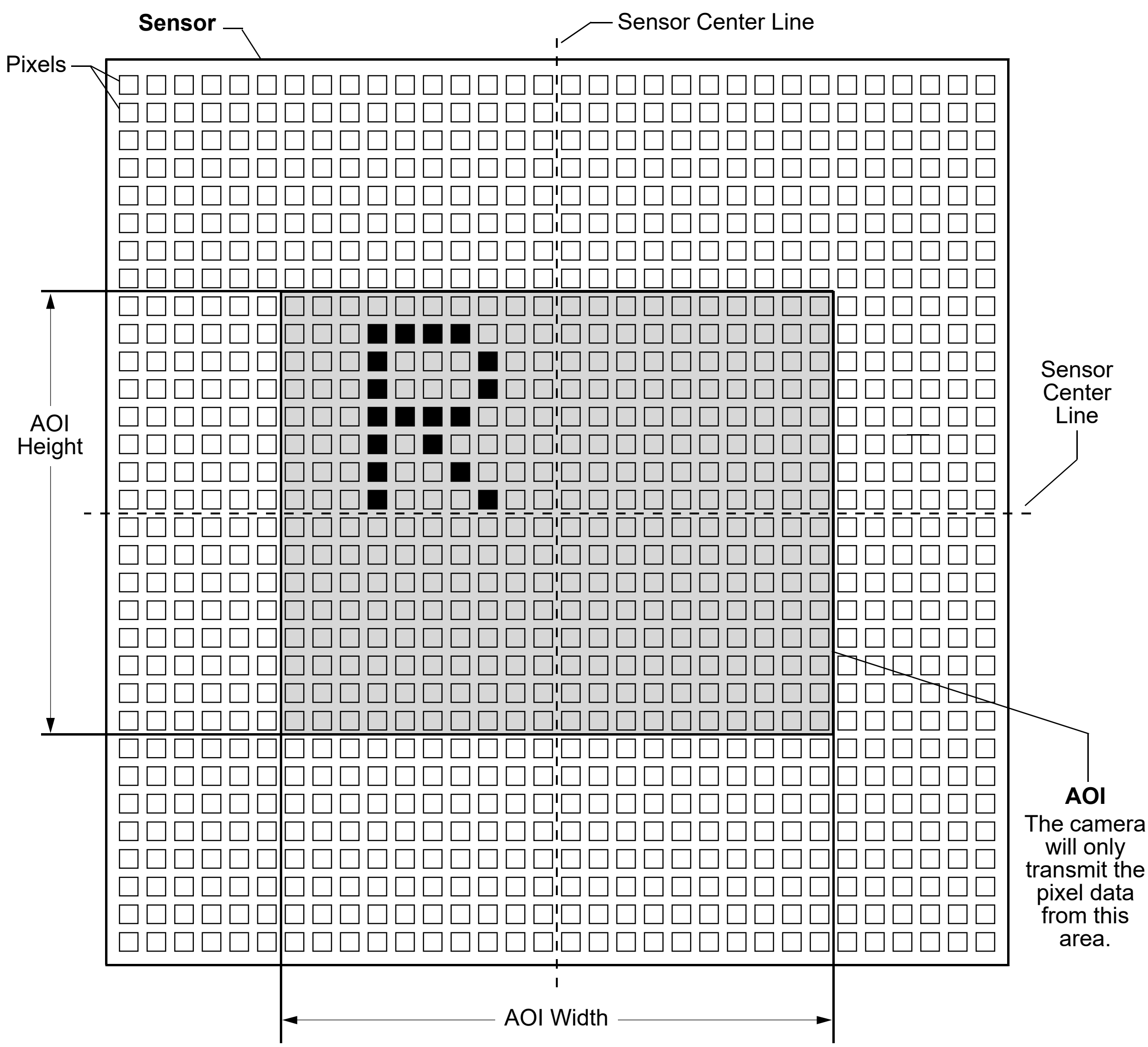


Fig. 47: Area of Interest for Tap Geometry 1X-2YE

The camera will only transfer pixel data from within the area defined by your settings. Information from the pixels outside of the area of interest is discarded.

One of the main advantages of the AOI feature is that decreasing the height of the AOI can increase the camera's maximum allowed frame acquisition rate. Typically, as the height of the AOI is decreased, the camera's maximum allowed frame rate will increase.

For more information about how changing the AOI height affects the maximum allowed frame rate, see Section 6.8 on page 112.

9.7.1 Setting the Image AOI

By default, the image AOI is set to use the nominal resolution for your camera model (see Section 1.2 on page 2 to determine the nominal resolution of your camera model). You can change the size of the AOI by changing the Width, and Height parameters.

When changing the width and height parameters, the following guidelines must be met:

- On monochrome versions of the camera, the width and height of the AOI can be set in increments of 2.
- On color versions of the camera, the width and height of the AOI can be set in increments of 4.
- As shown in Table 10, the minimum allowed setting for the AOI Height depends on the camera model and whether binning is enabled.

Camera Model	No Vertical Binning	Vertical Binning by 2 Enabled	Vertical Binning by 3 Enabled	Vertical Binning by 4 Enabled
avA1000-120km	128	64	42	32
avA1000-120kc	128	NA	NA	NA
avA1600-65km	128	128	128	128
avA1600-65kc	128	NA	NA	NA
avA1900-60km	128	128	128	128
avA1900-60kc	128	NA	NA	NA
avA2300-30km	444	444	444	444
avA2300-30kc	444	NA	NA	NA
NA = binning is not available on color cameras				

Table 10: Minimum AOI Height Settings

i	Normally the Width, and Height parameter settings refer to the physical columns and lines in the sensor. But if binning is enabled, these parameters are set in terms of "virtual" columns and lines. For more information about binning, see Section 9.12 on page 233.

Setting the Image AOI Using Basler pylon

You can use the pylon API to set the Width and Height parameter values from within your application software. The following code snippets illustrate using the pylon API to get the maximum allowed settings and the increments for the Width and Height parameters. They also illustrate setting the Width and Height parameter values.

```
// Get max width allowed, get width increment, set the width.
int64_t widthMax = Camera.Width.GetMax( );
int64_t widthInc = Camera.Width.GetInc();
Camera.Width.SetValue( 200 );

// Get max height allowed, get height increment, set the height.
int64_t heightMax = Camera.Height.GetMax( );
int64_t heightInc = Camera.Height.GetInc();
Camera.Height.SetValue( 200 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Image AOI Using Direct Register Access

To set the AOI width and height via direct register access:

- Set the value of the Width register.
- Set the value of the Height register.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.7.2 Prelines

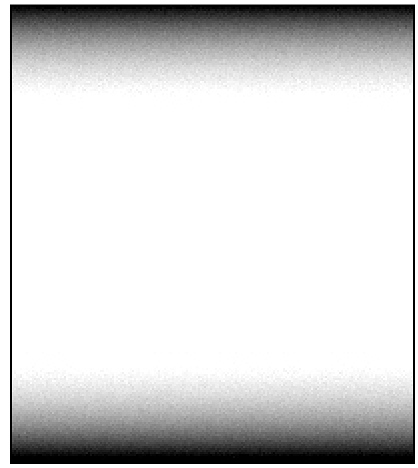


Fig. 48: Dark Areas in Small AOI

As you work with the camera's AOI feature, you may notice that in some situations dark areas appear near the top and the bottom of acquired images as shown in Figure 48. (The image in the figure was acquired using a light gray test pattern.)

These dark areas typically will not be present when large AOIs are used, but will become more noticeable when the AOI height is smaller. The effect will be most noticeable when the AOI height is very small. The effect will be especially apparent if the area of the sensor outside of the AOI is very brightly illuminated.

The prelines feature is designed to minimize this effect. The minimum and maximum settings for the Prelines parameter depend on your camera model as shown in Table 11. Higher prelines settings result in better elimination of any dark areas at the top and bottom of the acquired images. At the highest setting, the prelines feature will minimize or eliminate these dark areas even when the AOI is very small.

Camera Model	Min Setting	Max Setting
avA1000-120km/kc	1	192
avA1600-65km/kc	1	192
avA1900-60km/kc	1	192
avA2300-30km/kc	1	192

Table 11: Minimum and Maximum Preline Settings

There is a trade-off when using the prelines feature. As mentioned earlier in this section, the camera's maximum allowed frame rate will typically increase as the height of the AOI is made smaller. Using the prelines feature impacts the relationship between the AOI height and the maximum frame rate. When you use a large prelines setting, you will see less of an increase in the maximum allowed frame rate as you make the AOI height smaller.

Setting the Prelines Using Basler pylon

You can use the pylon API to set the Prelines parameter value from within your application software. The following code snippet illustrates using the pylon API to set the parameter value:

```
Camera.Prelines.SetValue( 48 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Prelines Using Direct Register Access

To set the prelines via direct register access:

- Set the value of the Prelines register.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.8 Auto Functions

ℹ	Note that on cameras delivered from the factory, the exposure auto function will become enabled whenever you power on or reset the camera. This behavior happens due to the way that the camera's configuration sets are set in the factory. More specifically, it happens because the auto functions factory setup is selected as the default configuration set and the default configuration set is designated as the "startup" set. To change this behavior, you must change the configuration set settings. For more information about configuration sets, see Section 9.19 on page 256.

ℹ	The auto functions feature will not work, if the sequencer feature is enabled. For more information about the sequencer feature, see Section 9.11 on page 197.

9.8.1 Common Characteristics

Auto functions control image properties and are the "automatic" counterparts of certain features, such as the gain feature or the white balance feature, which normally require "manually" setting the related parameter values. Auto functions are particularly useful when an image property must be adjusted quickly to achieve a specific target value and when a specific target value must be kept constant in a series of images.

An Auto Function Area of Interest (Auto Function AOI) lets you designate a specific part of the image as the base for adjusting an image property. Each auto function uses the pixel data from an Auto Function AOI for automatically adjusting a parameter value and, accordingly, for controlling the related image property. Some auto functions always share an Auto Function AOI.

An auto function automatically adjusts a parameter value until the related image property reaches a target value, and the parameter value cannot be manually set.

For some auto functions, the target value is fixed. For other auto functions, the target value can be set, as can the limits between which the related parameter value will be automatically adjusted. For example, the gain auto function lets you set an average gray value for the image as a target value and also set a lower and an upper limit for the gain parameter value.

Generally, the different auto functions can operate at the same time. For more information, see the following sections describing the individual auto functions.

A target value for an image property can only be reached if it is in accord with all pertinent camera settings and with the general circumstances used for capturing images. Otherwise, the target value will only be approached.

For example, with a short exposure time, insufficient illumination, and a low setting for the upper limit of the gain parameter value, the Gain Auto function may not be able to achieve the current target average gray value setting for the image.

You can use an auto function when binning is enabled (monochrome cameras only). An auto function uses the binned pixel data and controls the image property of the binned image.

9.8.2 Auto Function Operating Modes

The following auto function modes of operation are available:

- The auto functions provide the "**once**" mode of operation. When the "once" mode of operation is selected, the parameter values are automatically adjusted until the related image property reaches the target value. After the automatic parameter value adjustment is complete, the auto function will automatically be set to "off" and the new parameter value will be applied to the following images.

  The parameter value can be changed by using the "once" mode of operation again, by using the "continuous" mode of operation, or by manual adjustment.

  If an auto function is set to the "once" operation mode and if the circumstances will not allow reaching a target value for an image property, the auto function will try to reach the target value for a maximum of 30 images and will then be set to "off".

- The auto functions also provide a "**continuous**" mode of operation where the parameter value is adjusted repeatedly while images are acquired.

  Depending on the current frame rate, the automatic adjustments will usually be carried out for every or every other image.

  The repeated automatic adjustment will proceed until the "once" mode of operation is used or until the auto function is set to "off", in which case the parameter value resulting from the latest automatic adjustment will operate, unless the parameter is manually adjusted.
- When an auto function is set to "off", the parameter value resulting from the latest automatic adjustment will operate, unless the parameter is manually adjusted.

	You can enable auto functions and change their settings while the camera is capturing images ("on the fly").

	If you have set an auto function to "once" or "continuous" operation mode while the camera was continuously capturing images, the auto function will become effective with a short delay and the first few images may not be affected by the auto function.

9.8.3 Auto Function AOIs

Each auto function uses the pixel data from an Auto Function AOI for automatically adjusting a parameter value, and accordingly, for controlling the related image property. Within these limitations, auto functions can be assigned to Auto Function AOIs as desired.

Each Auto Function AOI has its own specific set of parameter settings, and the parameter settings for the Auto Function AOIs are not tied to the settings for the AOI that is used to define the size of captured images (Image AOI). For each Auto Function AOI, you can specify a portion of the sensor array and only the pixel data from the specified portion will be used for auto function control. Note that an Auto Function AOI can be positioned anywhere on the sensor array.

An Auto Function AOI is referenced to the top left corner of the sensor array. The top left corner of the sensor array is designated as column 0 and row 0 as shown in Figure 49.

The location and size of an Auto Function AOI is defined by declaring an X offset (coordinate), a width, a Y offset (coordinate), and a height. For example, suppose that you specify the X offset as 14, the width as 5, the Y offset as 7, and the height as 6. The area of the array that is bounded by these settings is shown in Figure 49.

Only the pixel data from the area of overlap between the Auto Function AOI defined by your settings and the Image AOI will be used by the related auto function.

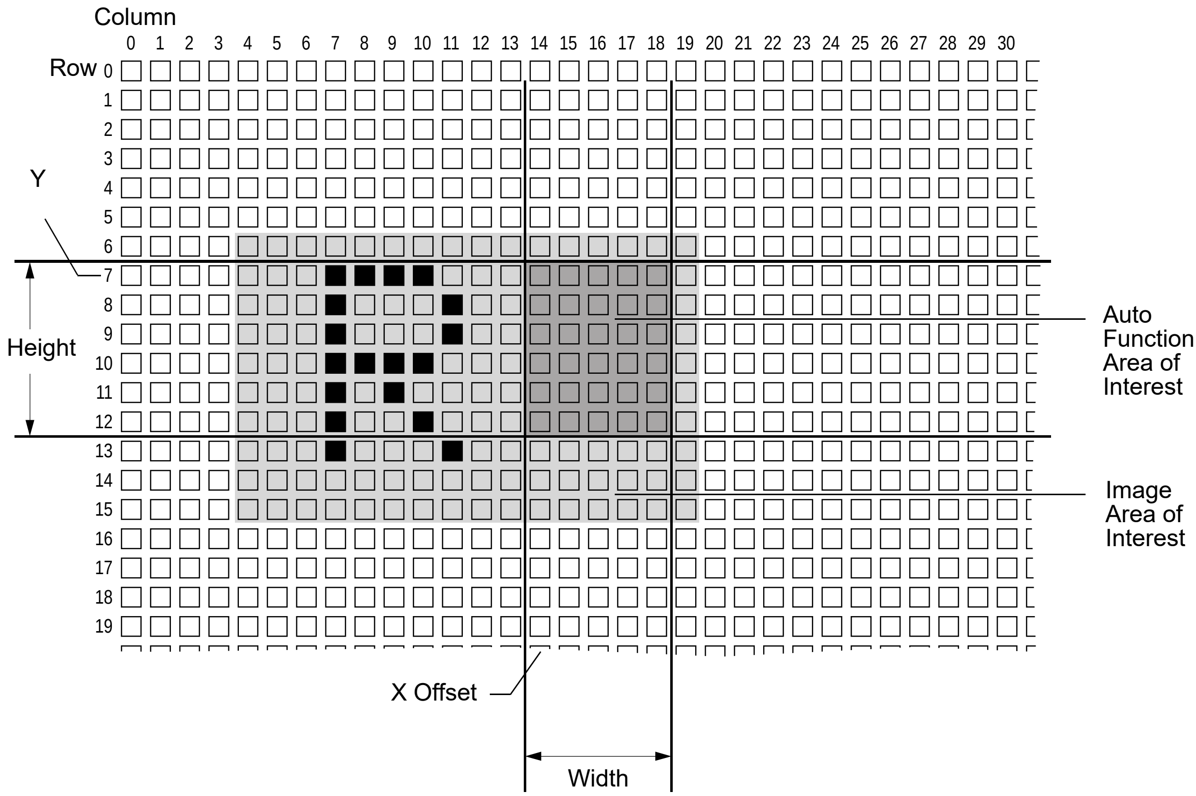


Fig. 49: Auto Function Area of Interest and Image Area of Interest

9.8.3.1 Positioning of an Auto Function AOI Relative to the Image AOI

The size and position of an Auto Function AOI can be, but need not be, identical to the size and position of the Image AOI. Note that the overlap between an Auto Function AOI and the Image AOI determines whether and to what extent the auto function will control the related image property. Only the pixel data from the areas of overlap of an Auto Function AOI and the Image AOI will be used by the auto function to control the image property of the entire image.

Different degrees of overlap are illustrated in Figure 50. The hatched areas in the figure indicate areas of overlap.

- If the Auto Function AOI is completely included in the Image AOI (see (a) in Figure 50), all pixel data from the Auto Function AOI will be used to control the image property.
- If the Image AOI is completely included in the Auto Function AOI (see (b) in Figure 50), only the pixel data from the Image AOI will be used to control the image property.
- If the Image AOI only partially overlaps the Auto Function AOI (see (c) in Figure 50), only the pixel data from the area of partial overlap will be used to control the image property.
- If the Auto Function AOI does not overlap the Image AOI (see (d) in Figure 50), the Auto Function will **not** control the image property. For details, see the sections below, describing the individual auto functions.

i	We strongly recommend completely including the Auto Function AOI within the Image AOI, or, depending on your needs, setting identical positions and sizes for the Auto Function AOIs and the Image AOI.

i	You can use auto functions when also using the reverse X and reverse Y mirroring features. For information about the behavior of Auto Function AOIs when also using the reverse X or reverse Y mirroring feature, see the "Mirror Image" section.

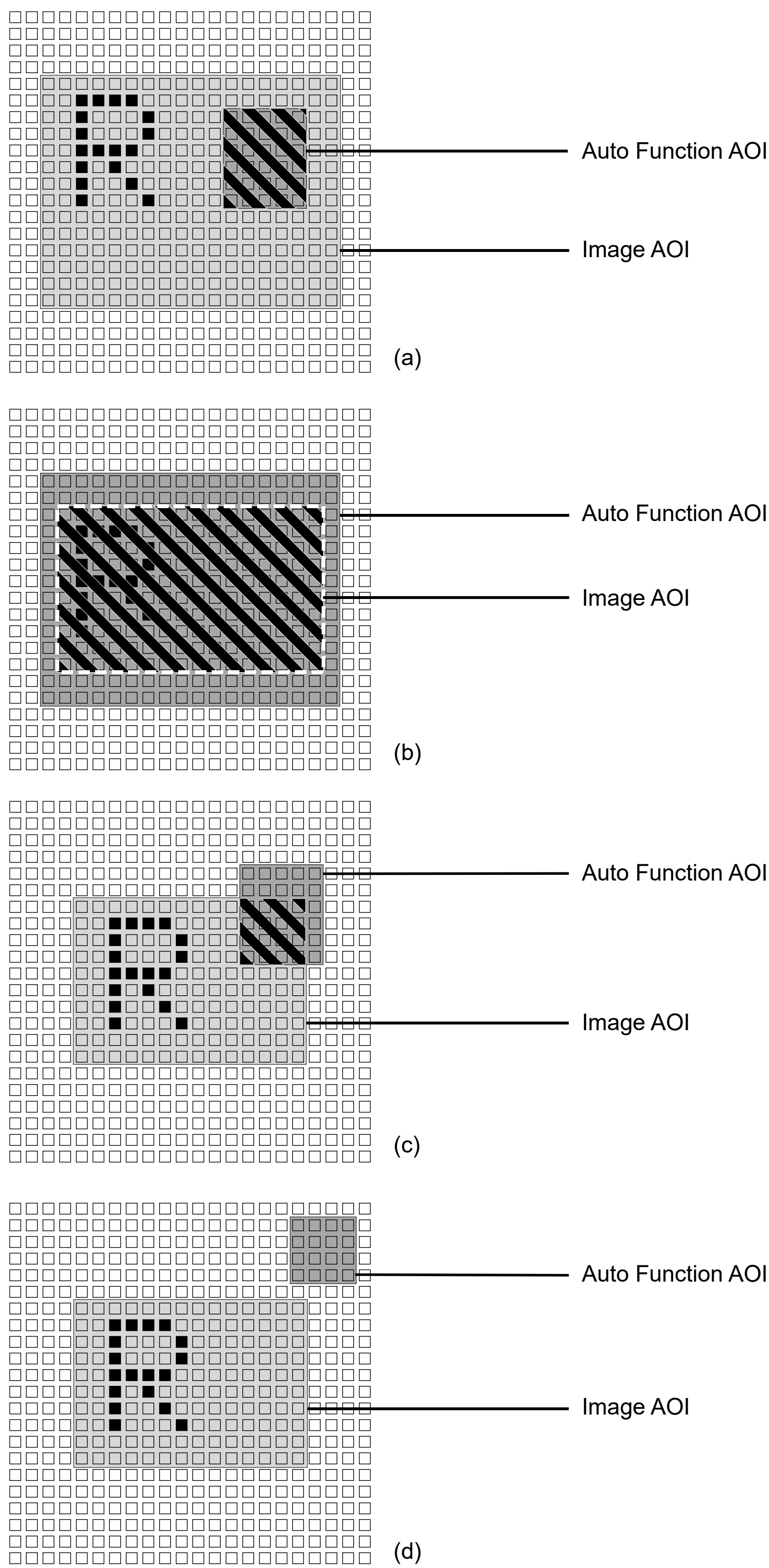


Fig. 50: Various Degrees of Overlap Between the Auto Function AOI and the Image AOI

9.8.3.2 Setting an Auto Function AOI

Setting an Auto Function AOI is a two-step process: You must first select the Auto Function AOI related to the auto function that you want to use and then set the position and the size of the Auto Function AOI.

By default, an Auto Function AOI is set to the full resolution of the camera's sensor. You can change the position and the size of an Auto Function AOI by changing the value of the Auto Function AOI's X Offset, Y Offset, Width, and Height parameters.

- The value of the X Offset parameter determines the starting column for the Auto Function AOI.
- The value of the Y Offset parameter determines the starting line for the Auto Function AOI.
- The value of the Width parameter determines the width of the Auto Function AOI.
- The value of the Height parameter determines the height of the Auto Function AOI.

When you are setting an Auto Function AOI, you must follow these guidelines:

- The sum of the X Offset setting plus the Width setting must not exceed the width of the camera's sensor. For example, on the avA1000-120km, the sum of the X Offset setting plus the Width setting must not exceed 1040.
- The sum of the Y Offset setting plus the Height setting must not exceed the height of the camera's sensor. For example, on the avA1000-120km, the sum of the X Offset setting plus the Width setting must not exceed 1040.
- On monochrome versions of the camera, the width and height of the AOI can be set in increments of 2.
- On color versions of the camera, the width and height of the AOI can be set in increments of 4.

On color cameras, we strongly recommend setting the X Offset, Y Offset, Width, and Height parameters for an Auto Function AOI in increments of 2 to make the Auto Function AOI match the Bayer filter pattern of the sensor. For example, you should set the X Offset parameter to 0, 2, 4, 6, 8, etc.

Normally, the X Offset, Y Offset, Width, and Height parameter settings for an Auto Function AOI refer to the physical columns and lines in the sensor. But if binning is enabled (monochrome cameras only), these parameters are set in terms of "virtual" columns and lines, i.e., the settings for an Auto Function AOI will refer to the binned lines and columns in the sensor and not to the physical lines in the sensor as they normally would.

For more information about the concept of a "virtual" sensor, see Section 9.12 on page 233.

Setting an Auto Function AOI Using Basler pylon

You can select an Auto Function AOI and set the X Offset, Y Offset, Width, and Height parameter values for the Auto Function AOI from within your application software by using the pylon API. The following code snippet illustrates using the pylon API to select Auto Function AOI one and to get the maximum allowed settings for the Width and Height parameters. The snippet also illustrates setting the X Offset, Y Offset, Width, and Height parameter values.

```
// Select auto function AOI 1
// Set position and size of the selected auto function AOI
Camera.AutoFunctionAOISelector.SetValue( AutoFunctionAOISelector_AOI1
);
Camera.AutoFunctionAOIOffsetX.SetValue( 0 );
Camera.AutoFunctionAOIOffsetY.SetValue( 0 );
Camera.AutoFunctionAOIWidth.SetValue(
Camera.AutoFunctionAOIWidth.GetMax() );
Camera.AutoFunctionAOIHeight.SetValue(
Camera.AutoFunctionAOIHeight.GetMax() );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting an Auto Function AOI Using Direct Register Access

To set the X Offset, Y Offset, Width, and Height for Auto Function 1 via direct register access:

- Set the value of the Auto AOI 1 Left register to set the X offset.
- Set the value of the Auto AOI 1 Top register to set the Y offset.
- Set the value of the Auto AOI 1 Width register to set the width.
- Set the value of the Auto AOI 1 Height register to set the height.

To set the X Offset, Y Offset, Width, and Height for Auto Function 2 via direct register access:

- Set the value of the Auto AOI 2 Left register to set the X offset.
- Set the value of the Auto AOI 2 Top register to set the Y offset.
- Set the value of the Auto AOI 2 Width register to set the width.
- Set the value of the Auto AOI 2 Height register to set the height.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.8.4 Using an Auto Function

To use an auto function, carry out the following steps:

1. Select an Auto Function AOI.
2. Assign the auto function you want to use to the selected Auto Function AOI.
3. Unassign the auto function you want to use from the other Auto Function AOI.
4. Set the position and size of the Auto Function AOI.
5. If necessary, set the lower and upper limits for the auto functions’s parameter value.
6. If necessary, set the target value.
7. Set the GrayValueAdjustmentDampingAbs parameter.
8. If necessary, set the auto function profile to define priorities between auto functions.
9. Enable the auto function by setting it to "once" or "continuous".

For more information about the individual settings, see the next sections that describe the individual auto functions.

9.8.5 Gain Auto

ℹ	Note that on cameras delivered from the factory, the exposure auto function will become enabled whenever you power on or reset the camera. This behavior happens due to the way that the camera's configuration sets are set in the factory. More specifically, it happens because the auto functions factory setup is selected as the default configuration set and the default configuration set is designated as the "startup" set. To change this behavior, you must change the configuration set settings. For more information about configuration sets, see Section 9.19 on page 256.

Gain Auto is the "automatic" counterpart to manually setting the Gain Raw All. When the gain auto function is operational, the camera will automatically adjust Gain Raw All within set limits until a target average gray value for the pixel data from the related Auto Function AOI is reached. (Automatic adjustments for Gain Tap 1, Gain Tap 2, Gain Tap 3, and Gain Tap 4 are not available.)

The gain auto function can be operated in the "once" and the "continuous" modes of operation.

If the related Auto Function AOI does not overlap the Image AOI (see the "Auto Function AOI" section) the pixel data from the Auto Function AOI will not be used to control the gain. Instead, the current manual setting for Gain Raw All will control the gain.

The gain auto function and the exposure auto function can be used at the same time. In this case, the auto function profile feature also takes effect. By default, the auto function profile feature minimizes gain.

For more information about

- the "manual" gain settings, see Section 9.3 on page 151.
- the auto function profile Section 9.8.8 on page 189.

The limits within which the camera will adjust the Gain Raw All are defined by the Auto Gain Raw Upper Limit and the Auto Gain Raw Lower Limit parameters. The minimum and maximum allowed settings for the Auto Gain Raw Upper Limit and Auto Gain Raw Lower Limit parameters depend on the current pixel format, on the current settings for binning, and on whether or not the parameter limits for manually setting the gain feature are disabled.

The Auto Target Value parameter defines the target average gray value that the gain auto function will attempt to achieve when it is automatically adjusting the Gain Raw All. The target average gray value can range from 50 (black) to 205 (white) when the camera is set for 8 bit output. When the camera is set for 10 bit output, the target gray value can range from 200 to 820. And when the camera is set for 12 bit output, the target gray value can range from 800 to 3280.

Setting the Gain Auto Function Using Basler pylon

Setting the gain auto functionality using Basler pylon is a several step process:

- Select the Auto Function AOI that was related to Gain Auto.
- Set the value of the Offset X, Offset Y, Width, and Height parameters for the AOI.
- Set the Gain Selector to All.
- Set the value of the Auto Gain Raw Lower Limit and Auto Gain Raw Upper Limit parameters.
- Set the value of the Auto Target Value parameter.
- Set the value of the Gain Auto parameter for the "once" or the "continuous" mode of operation.

You can set the gain auto functionality from within your application software by using the pylon API. The following code snippets illustrate using the pylon API to set the exposure auto functionality:

```
// Select the auto function AOI that was related to Gain Auto
// It is assumed here that auto function AOI 1 was related to Gain Auto
Camera.AutoFunctionAOISelector.SetValue( AutoFunctionAOISelector_AOI1
);

// Set the position and size of selected auto function AOI. In this
example, we set
// auto function AOI to cover the entire sensor.
Camera.AutoFunctionAOIOffsetX.SetValue( 0 );
Camera.AutoFunctionAOIOffsetY.SetValue( 0 );
Camera.AutoFunctionAOIWidth.SetValue(
Camera.AutoFunctionAOIWidth.GetMax() );
Camera.AutoFunctionAOIHeight.SetValue(
Camera.AutoFunctionAOIHeight.GetMax() );

// Select gain all and set the upper and lower gain limits for the gain
// auto function.
Camera.GainSelector.SetValue( GainSelector_All );
Camera.AutoGainRawLowerLimit.SetValue( Camera.GainRaw.GetMin() );
Camera.AutoGainRawUpperLimit.SetValue( Camera.GainRaw.GetMax() );

// Set target gray value for the gain auto function.
Camera.AutoTargetValue.SetValue( 128 );

// Set the mode of operation for gain auto function.
Camera.GainAuto.SetValue( GainAuto_Once );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Gain Auto Function Using Direct Register Access

Setting the gain auto functionality via direct register access is a several step process:

- Set the position and size of Auto Function AOI 1 by setting the value of the Auto AOI 1 Left register, the value of the Auto AOI 1 Top register, the value of the Auto AOI 1 Width register, and the value of the Auto AOI 1 Height register.

  (**Note:** This step assumes that the Gain Auto function is assigned to Auto Function AOI 1. If the Gain Auto function is assigned to Auto Function AOI 2, you would set the Auto Function AOI 2 registers.)
- Set the value of the Auto Gain Lower Limit register and the Auto Gain Upper Limit register.
- Set the value of the Auto Target Value register.
- Enable the gain auto function by selecting the value of the Gain Auto register to the Once or the Continuous mode of operation as desired.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.8.6 Exposure Auto

(i)	Note that on cameras delivered from the factory, the exposure auto function will become enabled whenever you power on or reset the camera. This behavior happens due to the way that the camera's configuration sets are set in the factory. More specifically, it happens because the auto functions factory setup is selected as the default configuration set and the default configuration set is designated as the "startup" set. To change this behavior, you must change the configuration set settings. For more information about configuration sets, see Section 9.19 on page 256. The exposure auto function will not work if the camera's exposure mode is set to trigger width. For more information about the trigger width exposure mode, see Section 6.3.3.2 on page 87.

Exposure Auto is the "automatic" counterpart to manually setting the Exposure Time Abs parameter. The exposure auto function automatically adjusts the Exposure Time Abs parameter. within set limits until a target average gray value for the pixel data from the related Auto Function AOI is reached.

The exposure auto function can be operated in the "once" and continuous" modes of operation. On cameras delivered from the factory, by default the exposure auto function is set to "continuous" mode of operation.

If the related Auto Function AOI does not overlap the Image AOI (see the "Auto Function AOI" section), the pixel data from the Auto Function AOI will not be used to control the exposure time. Instead, the current manual setting for the Exposure Time Abs parameter will control the exposure time.

The exposure auto function and the gain auto function can be used at the same time. In this case, the auto function profile feature also takes effect. By default, the auto function profile feature minimizes gain.

When trigger width exposure mode is selected, the exposure auto function is not available.

For more information about

- the "manual" exposure time setting, see Section 6.4 on page 92.
- the trigger width exposure mode, see Section 6.3.3.2 on page 87.
- the auto function profile Section 9.8.8 on page 189.

The limits within which the camera will adjust the Auto Exposure Time Abs parameter are defined by the Auto Exposure Time Abs Upper Limit and the Auto Exposure Time Abs Lower Limit parameters. The current minimum and the maximum allowed settings for the Auto Exposure Time Abs Upper Limit parameter and the Auto Exposure Time Abs Lower Limit parameters depend on the minimum allowed and maximum possible exposure time for your camera model.

The Auto Target Value parameter defines the target average gray value that the exposure auto function will attempt to achieve when it is automatically adjusting the Exposure Time Abs value. The target average gray value can range from 50 (black) to 205 (white) when the camera is set for 8 bit output. When the camera is set for 12 bit output, the target gray value can range from 800 to 3280.

ℹ	If the Exposure Time Abs Upper Limit Parameter is set to a sufficiently high value, the camera's maximum allowed frame rate may be decreased.

Setting the Exposure Auto Function Using Basler pylon

Setting the exposure auto functionality using Basler pylon is a several step process:

- Select the Auto Function AOI that was related to Exposure Auto.
- Set the value of the Offset X, Offset Y, Width, and Height parameters for the AOI.
- Set the value of the Auto Exposure Time Abs Lower Limit and Auto Exposure Time Abs Upper Limit parameters.
- Set the value of the Auto Target Value parameter.
- Set the value of the Exposure Auto parameter for the "once" or the "continuous" mode of operation.

You can set the exposure auto functionality from within your application software by using the pylon API. The following code snippets illustrate using the pylon API to set the exposure auto functionality:

```
// Select auto function AOI that was related to Exposure Auto
// It is assumed here that auto function AOI 1 was related to Exposure Auto
Camera.AutoFunctionAOISelector.SetValue( AutoFunctionAOISelector_AOI1 );

// Set the position and size of selected auto function AOI. In this
// example, we set auto function AOI to cover the entire sensor.
Camera.AutoFunctionAOIOffsetX.SetValue( 0 );
Camera.AutoFunctionAOIOffsetY.SetValue( 0 );
Camera.AutoFunctionAOIWidth.SetValue( Camera.AutoFunctionAOIWidth.GetMax() );
Camera.AutoFunctionAOIHeight.SetValue( Camera.AutoFunctionAOIHeight.GetMax() );

// Set the exposure time limits for the exposure auto function
Camera.AutoExposureTimeAbsLowerLimit.SetValue( 1000 );
Camera.AutoExposureTimeAbsUpperLimit.SetValue( 1.0E6 );

// Set target gray value for the exposure auto function
// If gain auto is enabled, this target is also used for gain auto
// control)
Camera.AutoTargetValue.SetValue( 128 );
```

```
// Set the mode of operation for the exposure auto function
Camera.ExposureAuto.SetValue( ExposureAuto_Continuous );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Exposure Auto Function Using Direct Register Access

Setting the exposure auto functionality via direct register access is a several step process:

- Set the position and size of Auto Function AOI 1 by setting the value of the Auto AOI 1 Left register, the value of the Auto AOI 1 Top register, the value of the Auto AOI 1 Width register, and the value of the Auto AOI 1 Height register.

  (**Note:** This step assumes that the Exposure Auto function has been assigned to Auto Function AOI 1. If the Exposure Auto function is assigned to Auto Function AOI 2, you would set the Auto Function AOI 2 registers.)
- Set the value of the Auto Exposure Lower Limit register and the Auto Exposure Upper Limit register.
- Set the value of the Auto Target Value register.
- Enable the gain auto function by selecting the value of the Exposure Auto register to the Once or the Continuous mode of operation as desired.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.8.7 Gray Value Adjustment Damping

The gray value adjustment damping controls the rate by which pixel gray values are changed when Exposure Auto and/or Gain Auto are enabled.

If an adjustment damping factor is used, the gray value target value is not immediately reached, but after a certain "delay". This can be useful, for example, when objects move into the camera's view area and where the light conditions are gradually changing due to the moving objects.

By default, the gray value adjustment damping is set to 0.6836. This is a setting where the damping control is as stable and quick as possible.

Setting the Adjustment Damping

The gray value adjustment damping is determined by the value of the Gray Value Adjustment Damping Abs parameter. The parameter can be set in a range from 0.0 to 0.78125.

The higher the value, the lower the adjustment damping is, i.e.

- the sooner the target value will be reached,
- the adaptation is realized over a smaller number of frames.

  Examples:
  0.6836 = Default value the camera starts with. There is a relatively immediate continuous adaptation to the target gray value.
  If you set the value to 0.5, there would be more interim steps; the target value would be reached after a "higher" number of frames.

You can set the gray value adjustment damping from within your application software by using the pylon API. The following code snippets illustrate using the API to set the gray value adjustment damping:

```
Camera.GrayValueAdjustmentDampingRaw.SetValue(600);
Camera.GrayValueAdjustmentDampingAbs.SetValue(0.5859);
```

You can also use the Basler pylon Viewer application to easily set the parameters.

Setting the Gray Value Adjustment Damping Using Direct Register Access

To set the gray value adjustment damping via direct register access:

- Set the value of the Gray Value Adjustment Damping register.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.8.8 Auto Function Profile

If you want to use the gain auto function and the exposure auto function at the same time, the auto function profile feature also takes effect. The auto function profile specifies whether the gain or the exposure time will be kept as low as possible when the camera is making automatic adjustments to achieve a target average gray value for the pixel data from the Auto Function AOI that was related to the gain auto and the exposure auto function. By default, the auto function profile feature minimizes gain.

If you want to use the gain auto and the exposure auto functions at the same time, you should set both functions for the continuous mode of operation.

Setting the Auto Function Profile Using Basler pylon

Setting the camera with Basler pylon to use the gain auto function and the exposure auto function at the same time is a several step process:

- Set the value of the Auto Function Profile parameter to specify whether gain or exposure time will be minimized during automatic adjustments.
- Set the value of the Gain Auto parameter to the "continuous" mode of operation.
- Set the value of the Exposure Auto parameter to the "continuous" mode of operation.

You can set the auto function profile from within your application software by using the pylon API. The following code snippet illustrates using the pylon API to set the auto function profile. As an example, Gain Auto is set to be minimized during adjustments:

```
// Use GainAuto and ExposureAuto simultaneously
Camera.AutoFunctionProfile.SetValue( AutoFunctionProfile_GainMinimum
);
Camera.GainAuto.SetValue( GainAuto_Continuous );
Camera.ExposureAuto.SetValue( ExposureAuto_Continuous );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Auto Function Profile Using Direct Register Access

Setting the camera to use the gain auto function and the exposure auto function at the same time via direct register access is a several step process:

- Set the value of the Auto Function Profile register to specify whether gain or exposure time will be minimized during automatic adjustments.
- Set the value of the Gain Auto register for the Continuous" mode of operation.
- Set the value of the Exposure Auto register for the Continuous mode of operation.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.8.9 Balance White Auto

ℹ	Note that on cameras delivered from the factory, the exposure auto function will become enabled whenever you power on or reset the camera. This behavior happens due to the way that the camera's configuration sets are set in the factory. More specifically, it happens because the auto functions factory setup is selected as the default configuration set and the default configuration set is designated as the "startup" set. To change this behavior, you must change the configuration set settings. For more information about configuration sets, see Section 9.19 on page 256.

Balance White Auto is the "automatic" counterpart to manually setting the white balance. The balance white auto function is only available on color models.

Automatic white balancing is a two-step process. First, the Balance Ratio Abs parameter values for red, green, and blue are each set to 1.5. Then, assuming a "gray world" model, the Balance Ratio Abs parameter values are automatically adjusted such that the average values for the "red" and "blue" pixels match the average value for the "green" pixels.

The balance white auto function uses the Auto Function AOI that was related to the Balance White Auto function. The balance white auto function can be operated in the "once" mode of operation and in the "continuous" mode of operation. For information about the "once" mode of operation and the "continuous" mode of operation, see Section 9.8.2 on page 174.

If the related Auto Function AOI does not overlap the image AOI (see the "Auto Function AOI" section) the pixel data from the Auto Function AOI will not be used to control the white balance of the image. However, as soon as the Balance White Auto function is set to "once" operation mode, the Balance Ratio parameter values for red, green, and blue are each set to 1.5. These settings will then control the white balance of the image.

For more information about the "manual" white balance settings, see Section 7.3.1 on page 126.

Setting the Balance White Auto Function Using Basler pylon

Setting the balance white auto functionality using Basler pylon is a several step process:

- Select the Auto Function AOI to which the Balance White Auto is assigned.
- Set the value of the Offset X, Offset Y, Width, and Height parameters for the AOI.
- Set the value of the Exposure Auto parameter for the "once" or the "continuous" mode of operation.

You can set the white balance auto functionality from within your application software by using the pylon API. The following code snippets illustrate using the pylon API to set the balance auto functionality:

```
// Select Auto Function AOI to which the Balance White Auto function
// is assigned. For this example, assume that the Balance White Auto
// function is assigned to Auto AOI 2
Camera.AutoFunctionAOISelector.SetValue( AutoFunctionAOISelector_AOI2
```

```
);

// Set the position and size of selected auto function AOI. In this
// example, we set auto function AOI to cover the entire sensor.
Camera.AutoFunctionAOIOffsetX.SetValue( 0 );
Camera.AutoFunctionAOIOffsetY.SetValue( 0 );
Camera.AutoFunctionAOIWidth.SetValue(
Camera.AutoFunctionAOIWidth.GetMax() );
Camera.AutoFunctionAOIHeight.SetValue(
Camera.AutoFunctionAOIHeight.GetMax() );

// Set mode of operation for balance white auto function
Camera.BalanceWhiteAuto.SetValue( BalanceWhiteAuto_Once );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting the Balance White Auto Function Using Direct Register Access

Setting the balance white auto functionality via direct register access is a several step process:

- Set the position and size of Auto Function AOI 2 by setting the value of the Auto AOI 2 Left register, the value of the Auto AOI 2 Top register, the value of the Auto AOI 2 Width register, and the value of the Auto AOI 2 Height register.

  (**Note:** This step assumes that the Balance White Auto function has been assigned to Auto Function AOI 2. If the Exposure Auto function is assigned to Auto Function AOI 1, you would set the Auto Function AOI 1 registers.)
- Enable the balance white auto function in the once mode of operation by selecting the value of the Balance White Auto register to Once.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.9 Minimum Output Pulse Width

An output signal sent by the camera may be too narrow for some receivers to be detected. To ensure reliable detection, the Minimum Output Pulse Width feature allows you to increase the signal width to a set minimum width:

- If the signal width of the original output signal is narrower than the set minimum the Minimum Output Pulse Width feature will increase the signal width to the set minimum before the signal is sent out of the camera (see the figure below).
- If the signal width of the original output signal is equal to or wider than the set minimum the Minimum Output Pulse Width feature will have no effect. The signal will be sent out of the camera with unmodified signal width.

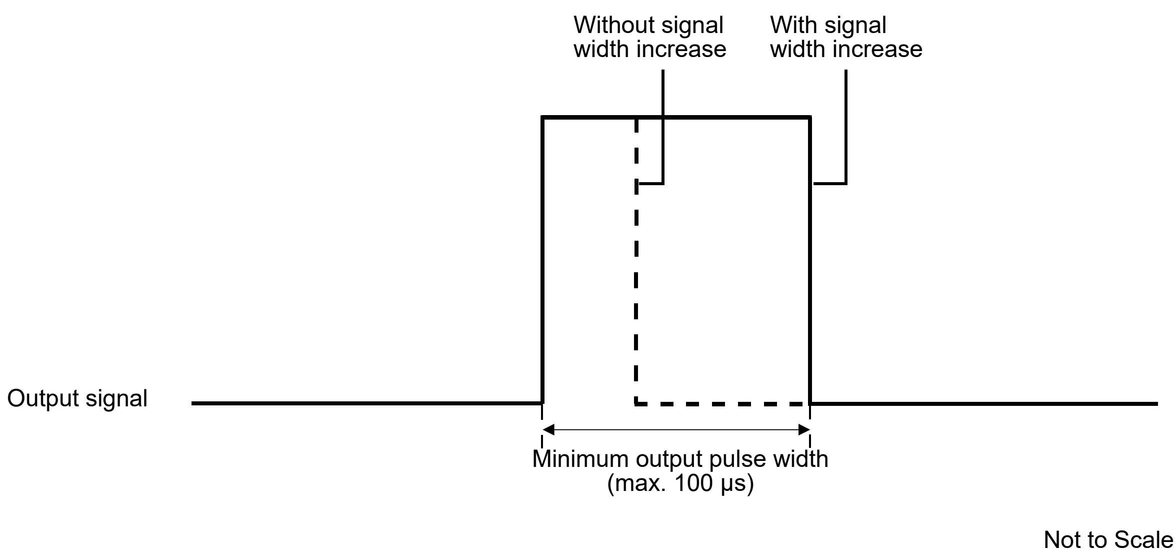


Fig. 51: Increasing the Signal Width of an Output Signal

Setting the Minimum Output Pulse Width Using Basler pylon

The minimum output pulse width is determined by the value of the MinOutPulseWidthAbs parameter. The parameter is set in microseconds and can be set in a range from 0 to 100 µs.

To set the minimum output pulse width parameter value:

- Use the Line Selector to select the camera output line 1.
- Set the value of the MinOutPulseWidthAbs parameter.

You can set the Line Selector and the value of the MinOutPulseWidthAbs parameter from within your application software by using the pylon API. The following code snippet illustrates using the API to set the selector and the parameter value:

```
// Select the input line
```

```
Camera.LineSelector.SetValue(LineSelector_Out1);

// Set the parameter value to 10.0 microseconds
Camera.MinOutPulseWidthAbs.SetValue(10.0);
```

For detailed information about using the pylon API, refer to the Basler pylon Programmer's Guide and API Reference.

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon Viewer, see Section 5.2 on page 62.

Setting the Minimum Output Pulse Width Using Direct Register Access

To set the Minimum Output Pulse Width via direct register access:

- Set the value of the Min Out Pulse Width Line 1 register.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.10 Error Detection

9.10.1 LED Indicator

The power, status, and error LED indicator on the back of the camera includes both a small red LED and a small green LED.

The camera can detect several user correctable errors. If one of these errors is present, the camera sets an error code and flashes both the red and green LEDs in the LED indicator as shown in Table 12. For information about the error codes, see Section 9.10.2 on page 195.

LED State	Status Indication
Red and Green Both Off	No camera power supplied to the camera.
Continuous Green / Red Off	The camera has booted up successfully and is OK.
Flashing Green / Red Off	The camera is set to expect an external trigger signal on an input, but no trigger signal is present.
Green and Red Both Flashing Alternately	An error condition has been detected that may be correctable with user intervention. (See the next section for more information).
Flashing Red / Green Off	Internal error. Contact Basler technical support.

Table 12: LED Indications

i	During the camera bootup process, both, the red and the green LEDs, will be lit simultaneously, resulting in an orange color.

9.10.2 Error Codes

The camera can detect several user correctable errors. If one of these errors is present, the camera will set an error code and will flash both the red and green LEDs in the LED indicator.

The following table indicates the available error codes:

Code	Condition	Meaning
0	No Error	The camera has not detected any errors since the last time that the error memory was cleared.
1	Overtrigger	An overtrigger has occurred. The user has applied an acquisition start trigger to the camera when the camera was not in a waiting for acquisition start condition. Or, the user has applied a frame start trigger to the camera when the camera was not in a waiting for frame start condition.
2	User set load	An error occurred when attempting to load a user set. Typically, this means that the user set contains an invalid value. Try loading a different user set .
3	Invalid Parameter	A parameter is set out of range or in an otherwise invalid manner. (Typically, this error only occurs when the user is setting parameters via direct register access.)

Table 13: Error Codes

When the camera detects a user correctable error, it sets the appropriate error code in an error memory. If two or three different detectable errors have occurred, the camera will store the code for each type of error that it has detected (it will store one occurrence of the each code no matter how many times it has detected the corresponding error).

You can use the following procedure to check the error codes:

- Read the value of the Last Error parameter. The Last Error parameter will indicate the last error code stored in the memory.
- Execute the Clear Last Error Command to clear the last error code from the memory.
- Continue reading and clearing the last error until the parameter indicates a No Error code.

Reading and Clearing the Error Codes Using Basler pylon

You can use the pylon API to read the value of the Last Error parameter and to execute a Clear Last Error command from within your application software. The following code snippets illustrate using the pylon API to read the parameter value and execute the command:

```
// Read the value of the last error code in the memory
LastErrorEnums lasterror = Camera.LastError.GetValue();

// Clear the value of the last error code in the memory
Camera.ClearLastError.Execute( );
```

You can also use the Basler pylon Viewer application to easily set the parameter and execute the command.

For more information about the pylon API and the pylon Viewer, see Section 5.1 on page 61.

Reading and Clearing the Error Codes Using Direct Register Access

To get the value of the last error code in the memory via direct register access:

- Read the value of the Last User Error register.

To clear the value of the last error code in the memory via direct register access:

- Set the value of the Clear Last User Error register to 1.

For more information about direct register access, see Section 5.2 on page 62.

9.11 Sequencer

i	The sequencer feature will not work, if the auto functions feature is enabled. For more information about the auto functions feature, see Section 9.8 on page 173.

The sequencer feature allows to apply specific sets of configuration parameter settings, called sequence sets, to a sequence of image acquisitions. As the images are acquired, one sequence set after the other is applied. This makes it possible to respond to different imaging requirements and conditions, that may, for example, result from changing illumination, while a sequence of images is acquired.

Three sequence advance modes (auto, controlled and free selection sequence advance modes) provide different schemes for advancing from one sequence set to the next (see below for details).

The Sequencer and the Active Configuration Set

During operation, the camera is controlled by a set of configuration parameters that reside in the camera's volatile memory. This set of parameters is known as the active configuration set or "active set" for short. When you use the pylon API or the pylon Viewer to make a change to a camera parameter such as the Gain, you are making a change to the active set. And since the active set controls camera operation, you will see a change in camera operation when you change a parameter in the active set. For more information about the active set, see the "Configuration Sets" section.

The parameters in the active set can be divided into two types (as shown in Figure 52 on page 198):

- "non-sequence" parameters

  The values of the non-sequence parameters cannot be changed using the sequencer feature.
- "sequence" parameters

  The values of the sequence parameters can be set very quickly by using sequence sets: Because the sequence sets reside in the camera, you can replace the values in the active set with values from one of the sequence sets almost instantaneously as images are acquired. Using the sequencer feature has no effect on the camera's frame rate.

The sequence set currently defining the parameter values of the active set is also called the "current set".

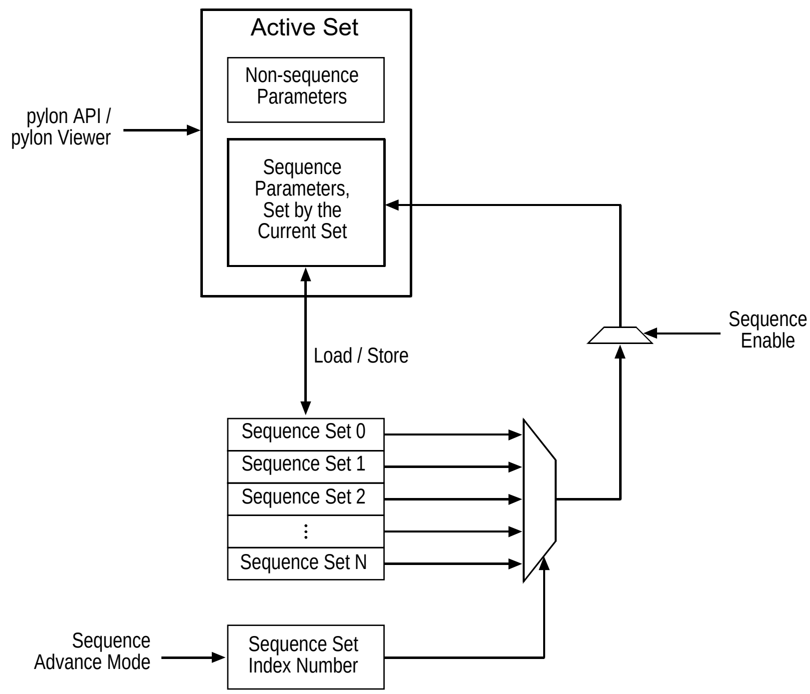


Fig. 52: Sequence Feature Block Diagram

The following parameters are examples of parameters that are included in each sequence set:

Exposure Time	Reverse X
Enable Acquisition Frame Rate	Reverse Y
Acquisition Frame Rate	Digital Shift
Width	LUT Enable
Height	Color Transformation Value
X Offset	Color Transformation Matrix Factor
Y Offset	Color Adjustment Enable
Center X	Color Adjustment Hue
Center Y	Color Adjustment Saturation
Binning Horizontal	Light Source Selector
Binning Vertical	Timer Delay*
Pixel Format	Timer Duration*
Test Image	Timer Delay Timebase*
Gain	Timer Duration Timebase*
Processed Raw Enable	Sequence Set Executions**
Black Level	

* This parameter is individually available for timer 1, timer 2.
**This parameter is only available in auto sequence advance mode.

Sequence Set Configuration

Before the sequencer feature can be used you must populate the sequence sets with the parameter values of the sequence parameters and store the sequence sets in the camera's memory. Each sequence set is identified by a sequence set index number starting from zero. After storing, the sequence sets are available for use by the sequencer feature.

Some sequence advance modes require the storing of additional settings, for example, the total number of sequence sets you want to use, the number of consecutive uses of a sequence set or the source to control sequence set advance. For details about populating sequence sets and making related settings see the sections below explaining the sequence advance modes.

i	When the sequencer feature is enabled, the sequence parameter values of the current sequence set cannot be read or changed using the pylon API or the pylon Viewer. Only those sequence parameter values will be displayed that were active before the sequencer was enabled. You will not be able to "see" the parameter values set by the current set. Make sure the sequencer feature is disabled when configuring sequence sets.

i	Because the sequence sets only reside in volatile memory they are lost if the camera is reset or switched off. If you are using the sequencer feature, you must populate the sequence sets after each camera reset or startup. Note also that sequence sets can not be saved in user sets.

Sequence Advance

As explained above, a sequence set can only control the operation of the camera after its parameter values were loaded into the active set. The loading into the active set and therefore the selection of a sequence set as the current set for a specific image acquisition are performed according to the selected sequence advance mode. The selection of a sequence set as the current set is always linked to the frame start trigger signals unless software commands are used (see below). Accordingly, a sequence advance mode provides a scheme for advancing from one sequence set to the next as frames are triggered.

The following three sequence advance modes are available:

- **Auto sequence advance mode**: Sequence set advance is automatically controlled by the camera. The camera will cycle through the available sequence sets in ascending sequence set index number as frames are triggered. Individual sequence sets can be used consecutively. After one sequence set cycle is complete another one will start automatically.
  For more information, see Section 9.11.1 on page 202.
- **Controlled sequence advance mode**: Sequence set advance is controlled by a source that can be selected. The available sources are automatic control by the camera (the "always active" setting), an input line or the "disabled" setting allowing sequence set advance only by software commands.The camera will cycle through the available sequence sets in ascending

sequence set index number as frames are triggered. After one sequence set cycle is complete another one will start automatically.
For more information, see Section 9.11.2 on page 207.

- **Free selection sequence advance mode**: Sequence set advance by selecting sequence sets at will from the available sequence sets. The selection is controlled by the states of the input lines.
For more information, see Section 9.11.3 on page 226.

The regular cycling through the sequence sets according to the Auto or Controlled advance modes can be modified at any time during the cycling:

- a **restart** starts a new sequence set cycle before the previous cycle is completed.
  The restart can be controlled
  - by the states of an input line (controlled sequence advance only);
    or
  - by a software command
- a non-cyclical **advance** allows to skip a sequence set and will advance to the sequence set after the next. The non-cyclical advance can be controlled
  - by the states of an input line
    or
  - by a software command.

Advance or restart controlled by an input line are also called "synchronous advance" and "synchronous restart" because the checking of the states of an input line is always linked to a frame trigger signal.

Advance or restart controlled by a software command are also called "asynchronous advance" and "asynchronous restart" because they are not linked to a frame start trigger signal.

Synchronous advance and restart are part of the standard operation of the sequencer feature and should generally be used. Asynchronous advance and restart are not suitable for standard operation because of the associated delays:

The delay between sending a software command and it becoming effective will depend on the specific installation and the current load on the network. Accordingly, the number of image acquisitions that may occur between sending the software command and it becoming effective can not be predicted. Asynchronous advance and restart are therefore not suitable for real-time applications, they may, however, be useful for testing purposes.

We strongly recommend to **only** use synchronous advance and synchronous restart for real-time applications.

Using the Load Command

There is also the Sequence Set Load command that may be useful when working with the sequence sets for testing purposes. If you use the Sequence Set Selector parameter to select a sequence set and then you execute the Sequence Set Load command, the sequence parameter values in the active set will be replaced by the values stored in the selected sequence set.

This ability can be useful in two situations:

- If you simply want to see how the parameters currently stored in one of the sequence sets will affect camera operation, you can load the parameters from that sequence set into the active parameter set and see what happens.
- If you want to prepare a new sequence set and you know that an existing set is already close to what you will need, you can load the existing sequence set into the active set, make some small changes to the active set, and then save the active set as a new sequence set.

Make sure the sequencer feature is disabled before issuing the Sequence Set Load command.

> Replacing the sequence parameter values in the active set via the Sequence Set Load command is associated with a delay between sending the software command and it becoming effective. The delay will depend on the specific installation and the current load on the network. Accordingly, the number of image acquisitions that may occur between sending the command and it becoming effective can not be predicted. The Sequence Set Load command is therefore not suitable for real-time applications, it may, however, be useful for testing purposes.

The following code snippet illustrates using the API to load the sequence parameter values from sequence set 0 into the active set:

```
// Select sequence set with index number 0
Camera.SequenceSetIndex.SetValue( 0 );
// Load the sequence parameter values from the sequence set into the
active set
Camera.SequenceSetLoad.Execute( );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

Use Case Diagrams Illustrating Sequencer Operation

The sections below explain the sequence advance modes in detail. Use case descriptions and diagrams are designed to illustrate how the sequence advance modes work in some common situations and with some common combinations of parameter settings.

These use cases do not represent every possible combination of the parameters associated with sequence advance mode operation. They are simply intended to aid you in developing an initial understanding of how the sequence advance modes work.

In each use case diagram, the black box in the upper left corner indicates how the parameters are set.

i	The use case diagrams are representational. They are not drawn to scale and are not designed to accurately describe precise camera timings.

9.11.1 Auto Sequence Advance Mode

When the auto sequence advance mode is selected the advance from one sequence set to the next occurs automatically as frame triggers are received. The advance proceeds in ascending sequence set index numbers and subject to the Sequence Set Executions parameter value. It specifies how many times each sequence set is consecutively used. After the sequence set with the highest index number was used as many times as specified by the Sequence Set Executions parameter value, the sequence set cycle starts again with sequence set 0.

The Sequence Set Total Number parameter specifies the total number of different sequence sets that are available and included within a sequence set cycle. The maximum number is 64.

9.11.1.1 Operation

Operating the Sequencer

The following use case (see also Figure 53) illustrates the operation of the sequencer in auto sequence advance mode. As images are captured continuously, the camera advances automatically with no action by the user from one sequence set to the next in ascending sequence set index numbers. The advance is also subject to the Sequence Set Executions parameter settings. After one sequence set cycle is complete, another one starts.

In this use case, the Sequence Set Total Number parameter was set to six. Accordingly, the available sequence set index numbers range from 0 through 5. The Sequence Set Executions parameter was set to 1 for sequence sets 0, 2, 3, and 4, to 2 for sequence set 5, and to 3 for sequence set 1. The frame start trigger is set for rising edge triggering.

Assuming that the camera is in the process of continuously capturing images, the sequencer feature operates as follows:

- When the sequencer feature becomes enabled, the sequence set cycle starts: The parameter values of the sequence set with sequence set index number 0 are loaded into the active set modifying the active set.

  When a frame start trigger is received, sequence set 0 is used for the image acquisition.
- When the next frame start trigger is received, the camera checks the current Sequence Set Executions parameter value. Because the Sequence Set Executions parameter is set to 1 for sequence set 0, this sequence set is only used once and therefore the camera advances to the next sequence set: The parameter values of sequence set 1 are loaded into the active set and are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the current Sequence Set Executions parameter value. Because the Sequence Set Executions parameter is set to 3 for sequence set 1, this sequence set is used a second time: The parameter values of sequence set 1 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the current Sequence Set Executions parameter value. Because the Sequence Set Executions parameter is set to 3 for sequence set 1, this sequence set is used a third time: The parameter values of sequence set 1 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the current Sequence Set Executions parameter value. Because the Sequence Set Executions parameter is set to 3 for sequence set 1, this sequence set can not, after three uses, be used again in the current sequence set cycle. Therefore, the camera advances to the next sequence set: The parameter values of sequence set 2 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the current Sequence Set Executions parameter value. Because the Sequence Set Executions parameter is set to 1 for sequence set 2, this sequence set is only used once and therefore the camera advances to the next sequence set: The parameter values of sequence set 3 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the current Sequence Set Executions parameter value. Because the Sequence Set Executions parameter is set to 1 for sequence set 3, this sequence set is only used once and therefore the camera advances to the next sequence set: The parameter values of sequence set 4 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the current Sequence Set Executions parameter value. Because the Sequence Set Executions parameter is set to 1 for sequence set 4, this sequence set is only used once and therefore the camera advances to the next sequence set: The parameter values of sequence set 5 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the current Sequence Set Executions parameter value. Because the Sequence Set Executions parameter is set to 2 for sequence set 5, this sequence set is used a second time: The parameter values of sequence set 5 are used for the image acquisition.

  The camera has cycled once through the complete sequence set cycle.
- When the next frame start trigger is received, the camera checks the current Sequence Set Executions parameter value. Because the Sequence Set Executions parameter is set to 2 for sequence set 5, this sequence set can not, after two uses, be used again in the current

sequence set cycle. Therefore the camera advances to the next sequence set: The parameter values of sequence set 0 are used for the image acquisition.

Another sequence set cycle has started.

- The sequencer feature is disabled while frame exposure and readout are in progress. The complete frame is transmitted and the cycling through sequence sets is terminated. The sequencer parameter values in the active set return to the values that existed before the sequencer feature was enabled.

Use Case 1: Operation in auto sequence advance mode:
Automatic cycling through the sequence set cycles with no action by the user. Enabling and disabling of the sequencer feature.

Settings: Sequence Set Total Number = 6
Sequence Set Executions = 1 for sequence sets 0, 2, 3, and 4
Sequence Set Executions = 2 for sequence set 5
Sequence Set Executions = 3 for sequence set 1

Frame start trigger --> Set to rising edge triggering

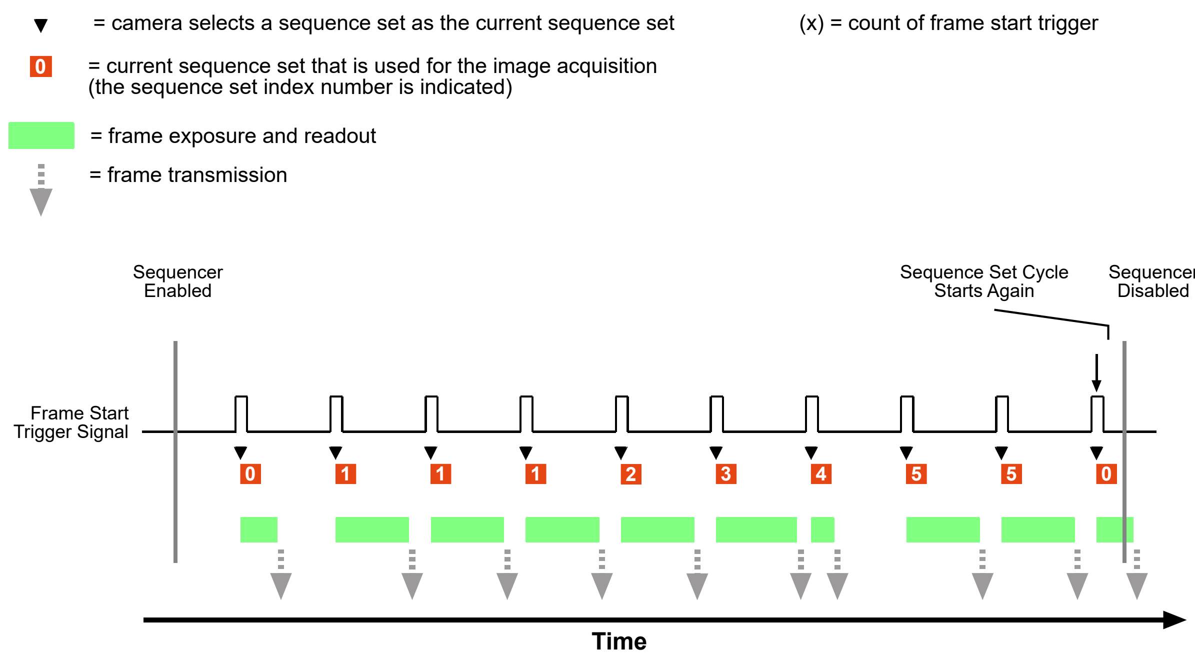


Fig. 53: Sequencer in Auto Sequence Advance Mode

Operating the Sequencer Using Basler pylon

You can use the pylon API to set the parameters for operating the sequencer in Auto sequence advance mode from within your application software.

The following code snippet illustrates enabling the sequencer and disabling the sequencer. The example assumes that sequence sets were previously configured and are currently available in the camera's memory.

```
// Enable the sequencer feature
Camera.SequenceEnable.SetValue( true );

// Disable the sequencer feature
Camera.SequenceEnable.SetValue( false );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

9.11.1.2 Configuration

Configuring Sequence Sets and Advance Control

Use the following procedure for populating sequence sets and making the related settings:

1. Make sure that the sequencer feature is disabled.
2. Set the Sequence Advance Mode parameter to Auto.
3. Set the Sequence Set Total Number parameter. The maximum number is 64.
4. Select a sequence set index number by setting the Sequence Set Index parameter. The available numbers range from 0 to 63.

   When configuring sequence sets make sure to always use a continuous series of index numbers starting with index number 0 and ending with the Sequence Set Total Number parameter value minus one. For example, specifying a series of sequence sets only with index numbers 5, 6, and 8 is not allowed. If you did nonetheless, the not explicitly configured sequence sets would, within the scope of the sequence set total number, be populated by default parameter values.
5. Set up your first acquisition scenario (i.e., lighting, object positioning, etc.)
6. Adjust the camera parameters to get the best image quality with this scenario (you are adjusting all parameters in the active set).
7. Set the Sequence Set Executions parameter. The available numbers range from 1 to 256.
8. Execute the Sequence Set Store command to copy the sequence parameter values currently in the active set into the selected sequence set. Any already existing parameter values in the sequence set will be overwritten.
9. Repeat the above steps starting from step 4 for the other sequence sets.

Configuring Sequence Sets and Advance Control Using Basler pylon

You can use the pylon API to set the parameters for configuring sequence sets from within your application software.

The following code snippet gives example settings. It illustrates using the API to set the auto sequence advance mode, set the total number of sequence sets to 2, set the numbers of consecutive sequence set executions and populate sequence sets 0 and 1 by storing the sequence parameter values from the active set in the sequence sets:

```
// Disable the sequencer feature
Camera.SequenceEnable.SetValue( false );

// Set the Auto sequence advance mode
Camera.SequenceAdvanceMode.SetValue( SequenceAdvanceMode_Auto );

// Set the total number of sequence sets
Camera.SequenceSetTotalNumber.SetValue( 2 );

// Select sequence set with index number 0
Camera.SequenceSetIndex.SetValue( 0 );

// Set up the first acquisition scenario (lighting, object position,
etc.) and
// adjust the camera parameters for the best image quality.

// Set the number of sequence set uses
Camera.SequenceSetExecutions.SetValue( 1 );

// Store the sequence parameter values from the active set in the
selected sequence
// set
Camera.SequenceSetStore.Execute( );

// Select sequence set with index number 1
Camera.SequenceSetIndex.SetValue( 1 );

// Set up the second acquisition scenario (lighting, object position,
etc.) and
// adjust the camera parameters for the best image quality.

// Set the number of sequence set uses
Camera.SequenceSetExecutions.SetValue( 4 );

// Store the sequence parameter values from the active set in the
selected sequence
// set
Camera.SequenceSetStore.Execute( );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

9.11.2 Controlled Sequence Advance Mode

When the controlled sequence advance mode is selected the advance from one sequence set to the next proceeds in ascending sequence set index numbers according to the selected sequence control source:

- Always Active:
  The advance from one sequence set to the next proceeds automatically as frame triggers are received.
- Line 1, Line 2, CC1, CC2 or CC4:
  The advance from one sequence set to the next proceeds according to the states of the selected input line.
- Disabled:
  The advance from one sequence set to the next is only controlled by AsyncAdvance software commands.

The Sequence Set Total Number parameter specifies the total number of different sequence sets that are available and included within a sequence set cycle. The maximum number is 64.

9.11.2.1 Operation with the "Always Active" Sequence Control Source

Operating the Sequencer

When the Always Active sequence control source is selected the advance from one sequence set to the next proceeds automatically in ascending sequence set index numbers as frame start triggers are received.

The following use case (see also Figure 54) illustrates the operation of the sequencer in controlled sequence advance mode with Always Active selected as the sequence control source. As images are captured continuously, the camera advances automatically with no action by the user from one sequence set to the next in ascending sequence set index numbers. After one sequence set cycle is complete, another one starts.

i	This way of operating the sequencer feature is similar to operating it in auto sequence advance mode when each sequence set is used only once per sequence set cycle. Here, however, the first sequence set used for image acquisition after the sequencer feature was enabled is sequence set 1 as opposed to sequence set 0 in auto sequence advance mode.

In this use case, the Sequence Set Total Number parameter is set to six. Accordingly, the available sequence set index numbers range from 0 through 5. The frame start trigger is set for rising edge triggering.

Assuming that the camera is in the process of continuously capturing images, the sequencer feature operates as follows:

- When the sequencer feature becomes enabled, the sequence set cycle starts: The parameter values of the sequence set with sequence set index number 0 are loaded into the active set modifying the active set.

  When a frame start trigger is received, the camera automatically advances to the next sequence set: The parameter values of sequence set 1 are used for the image acquisition.
- When the next frame start trigger is received, the camera advances to the next sequence set: The parameter values of sequence set 2 are used for the image acquisition.
- When the next frame start trigger is received, the camera advances to the next sequence set: The parameter values of sequence set 3 are used for the image acquisition.
- and so on. Note that the camera has cycled once through the complete sequence set cycle when sequence set 5 was used. With the next frame start trigger, a new sequence set cycle starts where sequence set 0 is used.
- After the sequencer feature is disabled, the cycling through sequence sets is terminated. The sequencer parameter values in the active set return to the values that existed before the sequencer feature was enabled.

Use Case:	Operation in controlled sequence advance mode with Always Active as the sequence control source: Automatic cycling through the sequence set cycles with no action by the user. Enabling and disabling of the sequencer feature.
Setting:	Sequence Set Total Number = 6

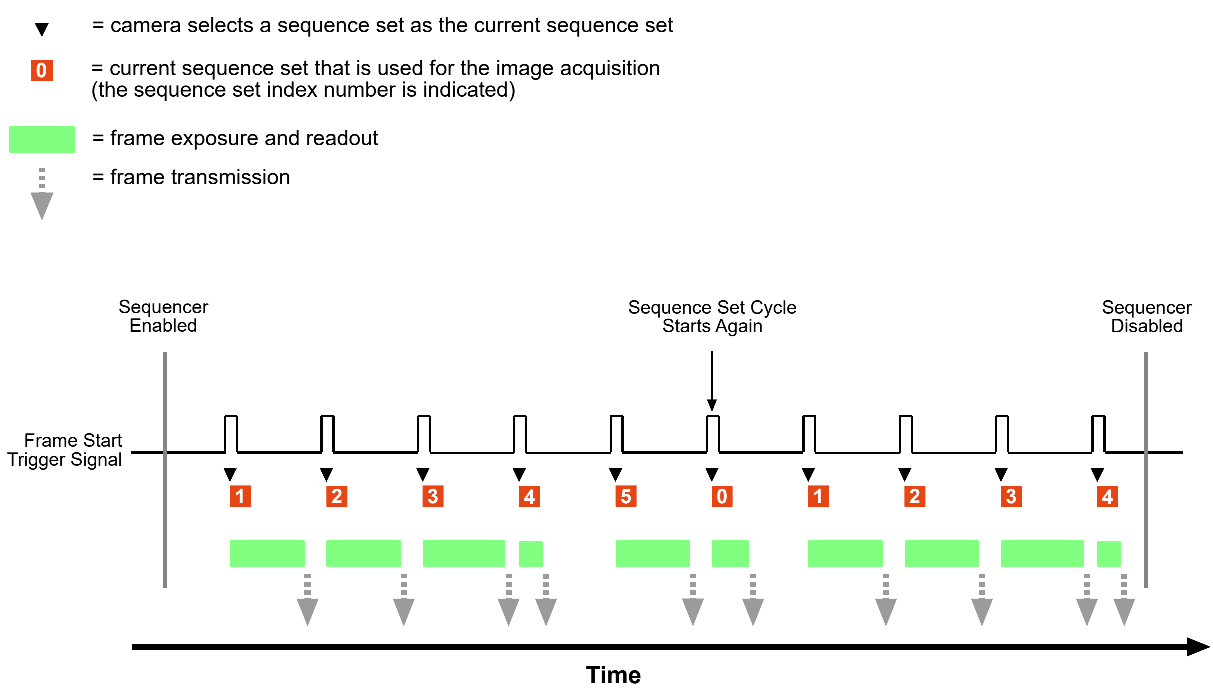


Fig. 54: Sequencer in Controlled Sequence Advance Mode with Always Active as the Sequence Control Source

Synchronous Restart

You can restart the sequence cycle with input line 1 or input line 2 as the source for controlling sequence cycle restart.

In the following use case (see also Figure 55), the same settings were made as in the previous use case: The Sequence Set Total Number parameter was set to six. Accordingly, the available sequence set index numbers range from 0 through 5. The frame start trigger is set for rising edge triggering. In addition, Line 1 was selected as the source for controlling restart. Line 1 is not set for invert.

Assuming that the camera is in the process of continuously capturing images, the sequencer feature operates as follows:

- When the sequencer feature becomes enabled, the sequence set cycle starts: The parameter values of the sequence set with sequence set index number 0 are loaded into the active set modifying the active set.

  When a frame start trigger is received, the camera automatically advances to the next sequence set: The parameter values of sequence set 1 are loaded into the active set and are used for the image acquisition.
- When the next frame start trigger is received, the camera advances to the next sequence set: The parameter values of sequence set 2 are used for the image acquisition.
- When the next frame start trigger is received, the camera advances to the next sequence set: The parameter values of sequence set 3 are used for the image acquisition.
- When the next frame start trigger is received, input line 1 is found to be high. Accordingly, another sequence set cycle is started and the parameter values of sequence set 0 are used for the image acquisition.

  Note that the synchronous restart has priority here over the automatic sequence set advance that results from the Always Active sequence control source. Without the priority rule, sequence set 1 would be used.

  Note that the state of input line 1 goes high well ahead of the frame start trigger.

> To ensure reliable synchronous sequence set restart, allow the elapse of at least one microsecond between setting the state of the input line and the rise of the frame start trigger signal.
>
> Also, maintain the state of the input line at least for one microsecond after the frame start trigger signal has risen.

Note also that the camera briefly exits the "waiting for frame start trigger" status while the input line changes its state. This happens when input line 1 changes its state before the fourth frame start trigger is received (see also Figure 55).

ℹ	Make sure not to send a frame start trigger while the input line changes its state. During this period, the camera will not wait for a frame start trigger and any frame start trigger will be ignored. Make sure to only send a frame start trigger when the camera is in "waiting for frame start trigger" status. For information about possibilities of getting informed about the "waiting for frame start trigger" status, see the Acquisition Monitoring Tools section.

- When the next frame start trigger is received, the camera advances to the next sequence set: The parameter values of sequence set 1 are used for the image acquisition.
- When the next frame start trigger is received, input line 1 is found to be high. Accordingly, another sequence set cycle is started and the parameter values of sequence set 0 are used for the image acquisition. As explained above, synchronous restart has priority here over the automatic sequence set advance.
- When the next frame start triggers are received, the camera advances to the next sequence sets and uses them for image acquisition in accord with the Always Active sequence control source and as described in the previous use case.

Use Case:	Operation in controlled sequence advance mode with Always Active as the sequence control source: Automatic cycling through the sequence set cycles with two synchronous restarts controlled by input line 1.
Setting:	Sequence Set Total Number = 6 Line 1 (not set for invert) is selected as the source for controlling restart

= camera is waiting for a frame start trigger

= camera selects a sequence set as the current sequence set

= current sequence set that is used for the image acquisition (the sequence set index number is indicated)

= frame exposure and readout

= frame transmission

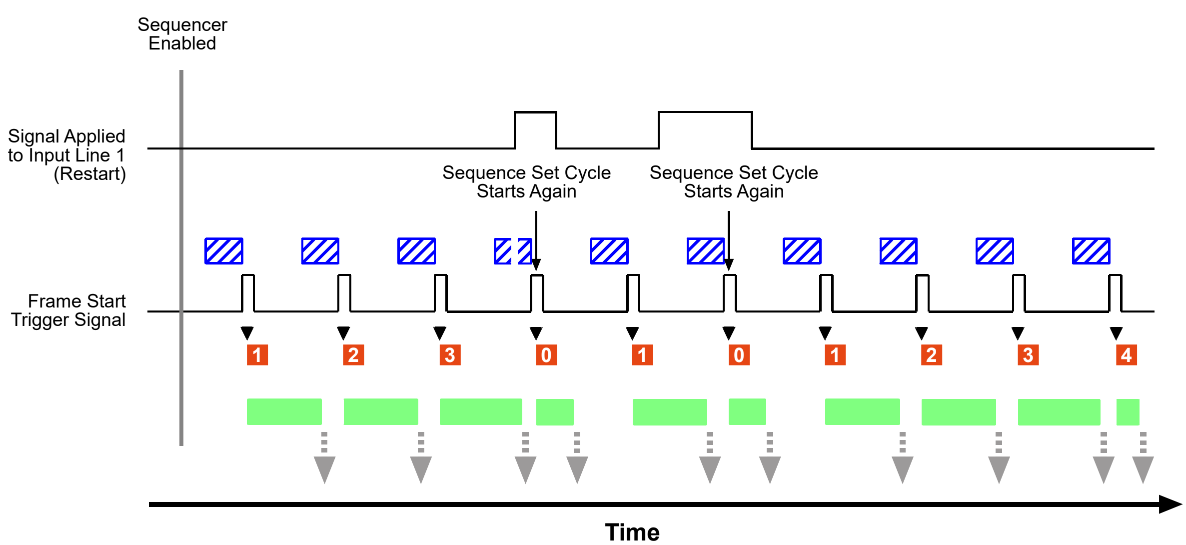


Fig. 55: Sequencer in Controlled Sequence Advance Mode with Always Active as the Sequence Control Source and Synchronous Restart Controlled by Line 1

9.11.2.2 Operation with an Input Line as Sequence Control Source

Operating the Sequencer

When the input line sequence control source is selected the advance from one sequence set to the next is controlled according to the states of the selected input lines. The advance proceeds in ascending sequence set index numbers as frame start triggers are received.

i	This section assumes that Line 1 is selected as the sequence control source. All explanations, however, apply equally well to other input lines (e.g. Line 2 or a CC line) as the sequence control source.

The following use case (see also Figure 56) illustrates the operation of the sequencer in controlled sequence advance mode with Line 1 selected as the sequence control source. The camera advances from one sequence set to the next in ascending sequence set index numbers. After one sequence set cycle is complete, another one starts. The sequence set advance is controlled by the states of Line 1. Line 1 is not set for invert.

In this use case, the Sequence Set Total Number parameter was set to six. Accordingly, the available sequence set index numbers range from 0 through 5. The frame start trigger is set for rising edge triggering.

Assuming that the camera is in the process of continuously capturing images, the sequencer feature operates as follows:

- When the sequencer feature becomes enabled, the sequence set cycle starts: The parameter values of the sequence set with sequence set index number 0 are loaded into the active set modifying the active set.

  When a frame start trigger is received, the camera checks the state of input line 1. Input line 1 is found to be low (the line status equals zero) and therefore no new sequence parameter values are loaded into the active set. The parameter values of sequence set 0 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the state of input line 1. Input line 1 is found to be high (the line status equals one) and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 1 are used for the image acquisition.

  Note that the state of input line 1 goes high well ahead of the frame start trigger.

i	To ensure reliable selection of a sequence set, allow the elapse of at least one microsecond between setting the states of the input lines and the rise of the frame start trigger signal. Also, maintain the states of the input lines at least for one microsecond after the frame start trigger signal has risen.

Note also that the camera briefly exits the "waiting for frame start trigger" status while an input line changes its state. This happened when input line 1 changed its state before the second frame start trigger was received (see also Figure 56).

> Make sure not to send a frame start trigger while the input line changes its state. During this period, the camera will not wait for a frame start trigger and any frame start trigger will be ignored.
>
> Make sure to only send a frame start trigger when the camera is in "waiting for frame start trigger" status.
>
> For information about possibilities of getting informed about the "waiting for frame trigger" status, see the Acquisiton Monitoring Tools section.

- When the next frame start trigger is received, the camera checks the state of input line 1. Input line 1 is found to be low and therefore no new sequence parameter values are loaded into the active set. The parameter values of sequence set 1 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the state of input line 1. Input line 1 is found to be low and therefore no new sequence parameter values are loaded into the active set. The parameter values of sequence set 1 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the state of input line 1. Input line 1 is found to be high and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 2 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the state of input line 1. Input line 1 is found to be high and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 3 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the state of input line 1. Input line 1 is found to be high and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 4 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the state of input line 1. Input line 1 is found to be high and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 5 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the state of input line 1. Input line 1 is found to be low and therefore no new sequence parameter values are loaded into the active set. The parameter values of sequence set 5 are used for the image acquisition.

  The camera has cycled once through the complete sequence set cycle.
- When the next frame start trigger is received, the camera checks the state of input line 1. Input line 1 is found to be high and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 0 are used for the image acquisition.

Another sequence set cycle has started.

- After frame exposure and readout are completed, the sequencer feature is disabled. The cycling through sequence sets is terminated. The sequencer parameter values in the active set return to the values that existed before the sequencer feature was enabled.

Use Case:	Operation in controlled sequence advance mode with Line 1 as the sequence control source: Cycling through the sequence set cycles according to the states of input line 1 (not set for invert). Enabling and disabling of the sequencer feature.
Setting:	Sequence Set Total Number = 6

= camera is waiting for a frame start trigger

= camera selects a sequence set as the current sequence set

0 = current sequence set that is used for the image acquisition (the sequence set index number is indicated)

= frame exposure and readout

= frame transmission

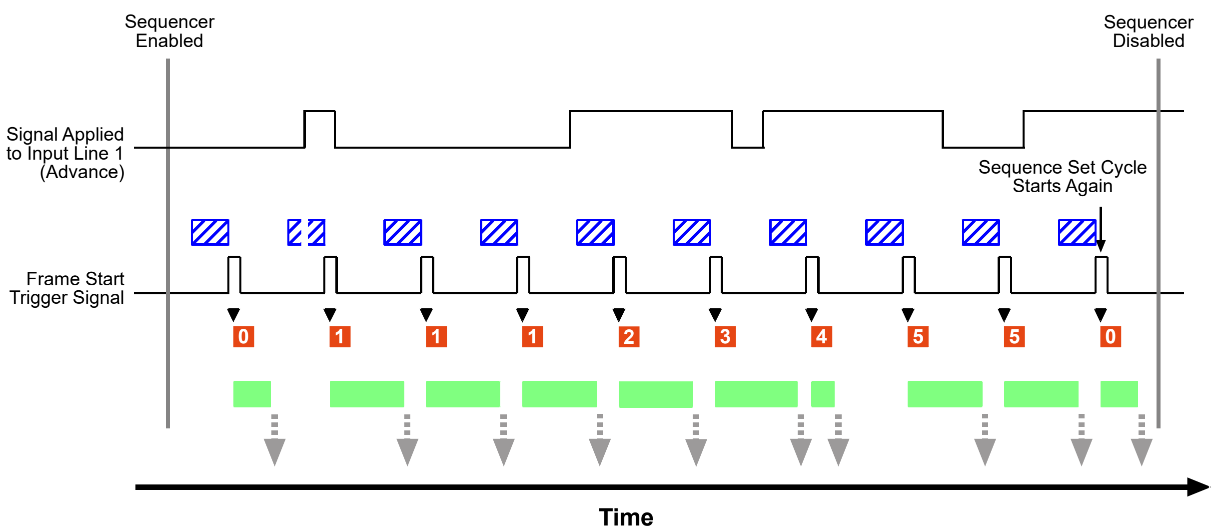


Fig. 56: Sequencer in Controlled Sequence Advance Mode with Line 1 as the Sequence Control Source

Synchronous Restart

You can restart the sequence cycle by selecting the input line that is not used for sequence advance control as the source for controling sequence cycle restart.

In the following use case (see also Figure 57), the same settings were made as in the previous use case: The Sequence Set Total Number parameter was set to six. Accordingly, the sequence set index numbers range from 0 through 5. The frame start trigger is set for rising edge triggering.

Line 1 is selected as the sequence control source for controlling sequence set advance. In addition, Line 2 is selected as the source for controlling sequence cycle restart. Both input lines are not set for invert.

Assuming that the camera is in the process of continuously capturing images, the sequencer feature operates as follows:

- When the sequencer feature becomes enabled, the sequence set cycle starts: The parameter values of the sequence set with sequence set index number 0 are loaded into the active set modifying the active set.

  When a frame start trigger is received, the camera checks the states of input lines 2 and 1. Input line 2 is found to be low and therefore the sequence cycle is not restarted. Input line 1 is found to be low and therefore no new sequence parameter values are loaded into the active set. The parameter values of sequence set 0 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the states of input lines 2 and 1. Input line 2 is found to be low and therefore the sequence cycle is not restarted. Input line 1 is found to be high and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 1 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the states of input lines 2 and 1. Input line 2 is found to be low and therefore the sequence cycle is not restarted. Input line 1 is found to be high and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 2 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the states of input lines 2 and 1. Input line 2 is found to be high and therefore the sequence cycle is restarted. Input line 1 is found to be low but this has no significance: Synchronous restart has priority over the secquence set advance control. The parameter values of sequence set 0 are used for the image acquisition.

  Another sequence set cycle has started.

  Note that the state of input line 2 goes high well ahead of the frame start trigger.

i	To ensure reliable selection of a sequence set, allow the elapse of at least one microsecond between setting the states of the input lines and the rise of the frame start trigger signal. Also, maintain the states of the input lines at least for one microsecond after the frame start trigger signal has risen.

Note also that the camera briefly exits the "waiting for frame start trigger" status while an input line changes its state. This happened, for example, when input line 2 changed its state before the fourth frame start trigger was received (see also Figure 57).

	Make sure not to send a frame start trigger while an input line changes its state. During this period, the camera will not wait for a frame start trigger and any frame start trigger will be ignored. Make sure to only send a frame start trigger when the camera is in "waiting for frame start trigger" status. For information about possibilities of getting informed about the "waiting for frame trigger" status, see the Acquisiton Monitoring Tools section.

- When the next frame start trigger is received, the camera checks the states of input lines 2 and 1. Input line 2 is found to be low and therefore the sequence cycle is not restarted. Input line 1 is found to be high and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 1 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the states of input lines 2 and 1. Input line 2 is found to be low and therefore the sequence cycle is not restarted. Input line 1 is found to be high and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 2 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the states of input lines 2 and 1. Input line 2 is found to be low and therefore the sequence cycle is not restarted. Input line 1 is found to be high and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 3 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the states of input lines 2 and 1. Input line 2 is found to be high and therefore the sequence cycle is restarted. Input line 1 is found to be high but this has no significance: Synchronous restart has priority over the secquence set advance control. The parameter values of sequence set 0 are used for the image acquisition.

  Another sequence set cycle has started.
- When the next frame start trigger is received, the camera checks the states of input lines 2 and 1. Input line 2 is found to be low and therefore the sequence cycle is not restarted. Input line 1 is found to be high and therefore the parameter values of the next sequence set are loaded into the active set. The parameter values of sequence set 1 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the states of input lines 2 and 1. Input line 2 is found to be low and therefore the sequence cycle is not restarted. Input line 1 is found to be low and therefore no new sequence parameter values are loaded into the active set. The parameter values of sequence set 1 are used for the image acquisition.

Use Case:	Operation in controlled sequence advance mode with Line 1 asthe sequence control source: Cycling through the sequence set cycles according to the states of input line 1 (not set for invert) with two synchronous restarts controlled by input line 2.
Setting:	Sequence Set Total Number = 6 Line 2 (not set for invert) is selected as the source for controlling restart

= camera is waiting for a frame start trigger

= camera selects a sequence set as the current sequence set

= current sequence set that is used for the image acquisition (the sequence set index number is indicated)

= frame exposure and readout

= frame transmission

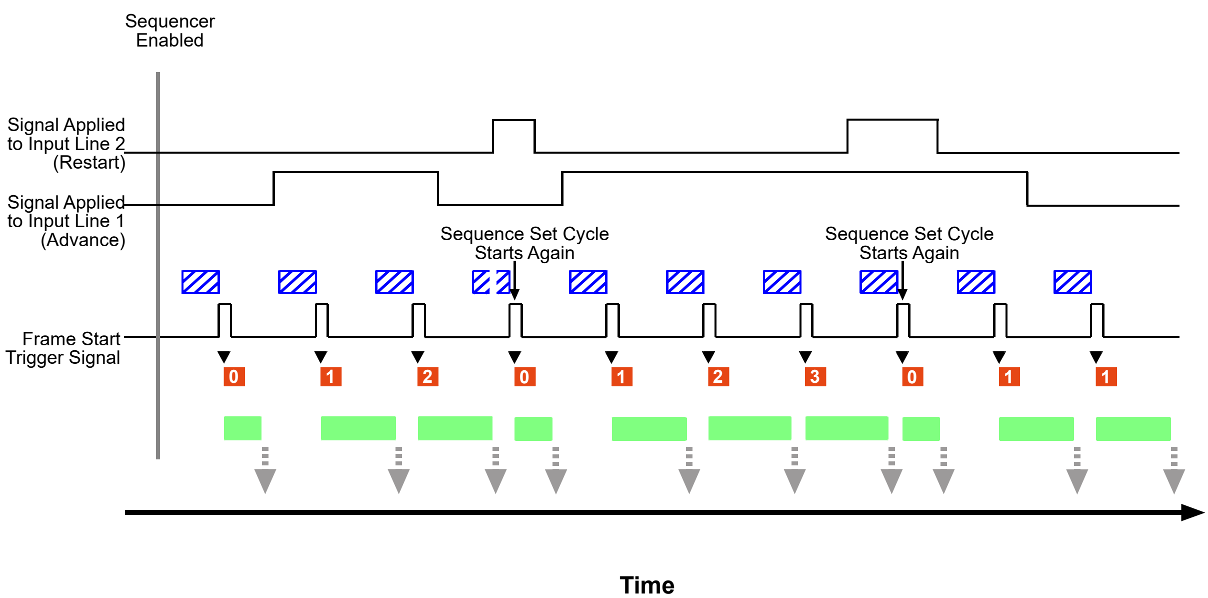


Fig. 57: Sequencer in Controlled Sequence Advance Mode with Line 1 as the Sequence Control Source and Synchronous Restart Controlled by Line 2

9.11.2.3 Operation with the "Disabled" Sequence Control Source

Operating the Sequencer

When the Disabled sequence control source is selected the advance from one sequence set to the next proceeds in ascending sequence set index numbers and is only possible by asynchronous advance.

Similarly, sequence set restart is only possible by asynchronous restart.

> The delay between sending an AsyncAdvance or an AsyncRestart software command and it becoming effective will depend on the specific installation and the current load on the network. Accordingly, the number of image acquisitions that may occur between sending the software command and it becoming effective can not be predicted. Using the sequencer feature with Disabled sequence control source is therefore not suitable for real-time applications, it may, however, be useful for testing purposes.
>
> We strongly recommend **not** to use the sequencer feature with Disabled sequence control source for real-time applications.

The following use case (see also Figure 58) illustrates the operation of the sequencer in controlled sequence advance mode with Disabled selected as the sequence control source. Sequence set advance proceeds in ascending sequence set index numbers subject to asynchronous advance commands. After one sequence set cycle is complete, another one starts. Sequence set cycle restarts are subject to asynchronous restart commands.

In this use case, the Sequence Set Total Number parameter was set to six. Accordingly, the available sequence set index numbers range from 0 through 5. The frame start trigger is set for rising edge triggering.

Assuming that the camera is in the process of continuously capturing images, the sequencer feature operates as follows:

- When the sequencer feature becomes enabled, the sequence set cycle starts: The parameter values of the sequence set with sequence set index number 0 are loaded into the active set modifying the active set.

  When a frame start trigger is received, the camera checks the active set and uses it for the image acquisition. The parameter values of sequence set 0 are used.
- An AsyncAdvance command is sent. After some delay, the parameter values of the next sequence set will be loaded into the active set. It is assumed here that the delay between sending the AsyncRestart command and it becoming effective will allow the acquisition of two more images.
- When the next frame start trigger is received, the camera checks the active set and uses it for the image acquisition. The parameter values of sequence set 0 are used.

The AsyncAdvance command has not yet become effective because of the assumed associated delay.

- When the next frame start trigger is received, the camera checks the active set and uses it for the image acquisition. The parameter values of sequence set 0 are used.

  The AsyncAdvance command has not yet become effective because of the assumed associated delay.
- When the AsyncAdvance command becomes effective, the camera happens to be in "waiting for frame start trigger" status. The parameter values of the next sequence set, i.e. of sequence set 1, are loaded into the active set. Note that the camera briefly exits the "waiting for frame start trigger" status while the parameter values of sequence set 1 are loaded into the active set (see also Figure 58).

Make sure not to send a frame start trigger while the parameter values of a sequence set are loaded into the active set. During this period, the camera will not wait for a frame start trigger and any frame start trigger will be ignored.

Make sure to only send a frame start trigger when the camera is in "waiting for frame start trigger" status.

For information about possibilities of getting informed about the "waiting for frame start trigger" status, see the Acquisition Monitoring Tools section.

- When the next frame start trigger is received, the camera checks the active set and uses it for the image acquisition. The parameter values of sequence set 1 are used.
- When the next frame start trigger is received, the camera checks the active set and uses it for the image acquisition. The parameter values of sequence set 1 are used.
- When the next frame start trigger is received, the camera checks the active set and uses it for the image acquisition. The parameter values of sequence set 1 are used.
- An AsyncRestart command is sent. After some delay, the parameter values of sequence set 0 will be loaded into the active set. It is assumed here that the delay between sending the AsyncRestart command and it becoming effective will allow the acquisition of two more images.
- When the next frame start trigger is received, the camera checks the active set and uses it for the image acquisition. The parameter values of sequence set 1 are used.

  The AsyncRestart command has not yet become effective because of the assumed associated delay.
- When the next frame start trigger is received, the camera checks the active set and uses it for the image acquisition. The parameter values of sequence set 1 are used.

  The AsyncRestart command has not yet become effective because of the assumed associated delay.
- When the AsyncRestart command becomes effective, the camera happens to be in "waiting for frame start trigger" status. The parameter values of sequence set 0 are loaded into the active set. Note that the camera briefly exits the "waiting for frame start trigger" status while the parameter values of sequence set 1 are loaded into the active set (see also Figure 58).

	Make sure not to send a frame start trigger while the parameter values of a sequence set are loaded into the active set. During this period, the camera will not wait for a frame start trigger and any frame start trigger will be ignored. Make sure to only send a frame start trigger when the camera is in "waiting for frame start trigger" status. For information about possibilities of getting informed about the "waiting for frame start trigger" status, see the Acquisition Monitoring Tools section.

- When the next frame start trigger is received, the camera checks the active set and uses it for the image acquisition. The parameter values of sequence set 0 are used.

  Another sequence set cycle has started
- When the next frame start trigger is received, the camera checks the active set and uses it for the image acquisition. The parameter values of sequence set 0 are used.
- While frame exposure and readout are in progress, the sequencer feature is disabled. The complete frame is transmitted and the cycling through sequence sets is terminated. The sequencer parameter values in the active set return to the values that existed before the sequencer feature was enabled.

Use Case: Operation in controlled sequence advance mode with Disabled sequence control source:

Cycling through the sequence set cycles only due to one asynchronous advance and one asynchronous restart. Enabling and disabling of the sequencer feature.

Setting: Sequence Set Total Number = 6

= asynchronous advance (AsyncAdvance command)

= delay between sending the advance command and it becoming effective

= asynchronous restart (AsyncRestart command)

= delay between sending the restart command and it becoming effective

= camera is waiting for a frame start trigger

= camera selects a sequence set as the current sequence set

0 = current sequence set that is used for the image acquisition
(the sequence set index number is indicated)

= frame exposure and readout

= frame transmission

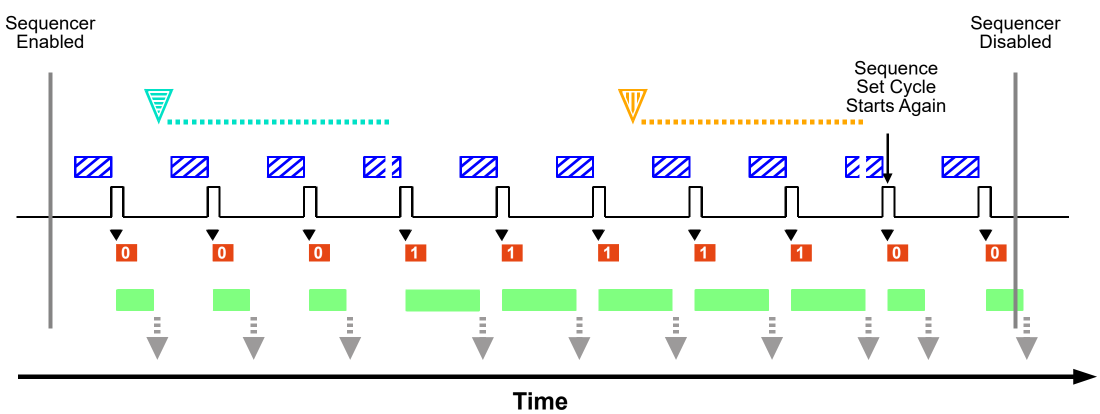


Fig. 58: Sequencer in Controlled Sequence Advance Mode with Disabled as the Sequence Control Source and Asynchronous Advance and Restart

Operating the Sequencer Using Basler pylon

You can use the pylon API to set the parameters for operating the sequencer in Controlled sequence advance mode from within your application software.

The following code snippet illustrates enabling and disabling the sequencer. The example assumes that sequence sets were previously configured and are currently available in the camera's memory.

```
// Enable the sequencer feature
Camera.SequenceEnable.SetValue( true );

// Disable the sequencer feature
Camera.SequenceEnable.SetValue( false );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

9.11.2.4 Configuration

Configuring Sequence Sets and Advance Control

Use the following procedure for populating sequence sets and setting the sources for sequence set advance and sequence cycle restart:

- Make sure that the sequencer feature is disabled.
- Set the Sequence Advance mode to Controlled.
- Set the Sequence Set Total Number parameter. The maximum number is 64.
- Set the Sequence Control Selector parameter to Advance to configure sequence set advance.
- Set the Sequence Control Source parameter to specify the source that will control synchronous sequence set advance. Note that the same source will apply to all sequence sets available at the same time in the camera.

  The following sources are available:
  - Always Active
  - Input lines: e.g. Line 1, Line 2, CC lines
  - Disabled
  - Set the Sequence Control Selector parameter to Restart to configure sequence set cycle restart.
  - Set the Sequence Control Source parameter to specify the source for restart.

ℹ	Never choose the same source for sequence set advance and sequence set cycle restart, with one exception: If you want to only use asynchronous advance and restart, choose Disabled as the source for advance and restart.

The following sources are available:

- Input lines: e.g. Line 1, Line 2, CC lines
- Disabled

- Select a sequence set index number by setting the Sequence Set Index parameter. The available numbers range from 0 to 63.

  When selecting index numbers for configuring, make sure to always start a sequence with 0 and to only set a continuous series of index numbers. For example, specifying a sequence of sets only with index numbers 5, 6, and 8 is therefore not allowed. If you did nonetheless, the not explicitly configured sequence sets would - within the scope of the sequence set total number - be populated by default parameter values.
- Set up your first acquisition scenario (i.e., lighting, object positioning, etc.)
- Adjust the camera parameters to get the best image quality with this scenario (you are adjusting the parameters in the active set).
- Execute the Sequence Set Store command to copy the sequence parameter values currently in the active set into the selected sequence set. (Any existing parameter values in the sequence set will be overwritten.)
- Repeat the above steps for the other sequence sets.

For information about setting the input line for invert, see Section 3.1.4 on page 41.

Configuring Sequence Sets and Advance Control Using Basler pylon

You can use the pylon API to set the parameters for configuring sequence sets from within your application software.

The following code snippet gives example settings. It illustrates using the API to set the controlled sequence advance mode. In the example, Line 1 is set as the sequence control source for synchronous sequence set advance, Disabled is set as the sequence control source to allow asynchronous sequence cycle reset, the total number of sequence sets is set to 2, sequence sets 0 and 1 are populated by storing the sequence parameter values from the active set in the sequence sets, and to enable the sequencer feature:

```
// Disable the sequencer feature
Camera.SequenceEnable.SetValue( false );

// Set the Controlled sequence advance mode and set line 1 as the
```

```
// sequence control source for synchronous sequence set advance
Camera.SequenceAdvanceMode.SetValue( SequenceAdvanceMode_Controlled
);
Camera.SequenceControlSelector.SetValue(
SequenceControlSelector_Advance );
Camera.SequenceControlSource.SetValue( SequenceControlSource_Line1 );

// Set Disabled as the source because synchronous sequence set cycle
// restart will not be used
Camera.SequenceControlSelector.SetValue(
SequenceControlSelector_Restart );
Camera.SequenceControlSource.SetValue( SequenceControlSource_Disabled
);

// Set the total number of sequence sets
Camera.SequenceSetTotalNumber.SetValue( 2 );

// Select sequence set with index number 0
Camera.SequenceSetIndex.SetValue( 0 );

// Set up the first acquisition scenario (lighting, object position,
// etc.) and adjust the camera parameters for the best image quality.
// Store the sequence parameter values from the active set in the
// selected sequence set
Camera.SequenceSetStore.Execute( );

// Select sequence set with index number 1
Camera.SequenceSetIndex.SetValue( 1 );

// Set up the second acquisition scenario (lighting, object position,
// etc.) and adjust the camera parameters for the best image quality.
// Store the sequence parameter values from the active set in the
// selected sequence set
Camera.SequenceSetStore.Execute( );

// Enable the sequencer feature
Camera.SequenceEnable.SetValue( true );
```

The following code snippet illustrates using the API to load the sequence parameter values from sequence set 0 into the active set:

```
// Select sequence set with index number 0
Camera.SequenceSetIndex.SetValue( 0 );
// Load the sequence parameter values from the sequence set into the
// active set
Camera.SequenceSetLoad.Execute( );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

9.11.3 Free Selection Sequence Advance Mode

When the free selection sequence advance mode is selected the advance from one sequence set to the next as frame start triggers are received does not adhere to a specific preset sequence: The sequence sets can be selected at will using the states of input lines: The states of the input lines set the sequence set addresses. These correspond to the sequence set index numbers and accordingly, the related sequence set is selected. For details about selecting sequence sets via the sequence set address, see the "Selecting Sequence Sets" section.

The states of the input lines are checked if more than two sequence sets are available. The states of one input line is checked when only two sequence sets are available.

The Sequence Set Total Number parameter specifies the total number of different sequence sets that are available. The maximum number is 4.

9.11.3.1 Operation

Operating the Sequencer

The following use case (see also Figure 59) illustrates the operation of the sequencer in free selection sequence advance mode.

In this use case, the Sequence Set Total Number parameter was set to four. Accordingly, the sequence set index numbers range from 0 through 3. Input line 1 sets bit 0 of the sequence set address. Input line 2 sets bit 1 of the sequence set address. Both input lines are not set for invert. The frame start trigger is set for rising edge triggering.

Assuming that the camera is in the process of continuously capturing images, the sequencer feature operates as follows:

- When the sequencer feature becomes enabled and a frame start trigger is received, the camera checks the states of input lines 1 and 2. Input line 1 is found to be high and input line 2 is found to be low. This corresponds to the address of sequence set 1. Accordingly, sequence set 1 is selected. Its parameter values are loaded into the active set and are used for the image acquisition.

  Note that the state of input line 1 goes high well ahead of the frame start trigger.

i	To ensure reliable selection of a sequence set, allow the elapse of at least one microsecond between setting the states of the input lines and the rise of the frame start trigger signal. Also, maintain the states of the input lines at least for one microsecond after the frame start trigger signal has risen.

  Note also that the camera briefly exits the "waiting for frame start trigger" status while an input line changes its state. This happens, for example, when input line 1 changes its state before the first frame start trigger is received (see also Figure 59).

ℹ	Make sure not to send a frame start trigger while an input line changes its state. During this period, the camera will not wait for a frame start trigger and any frame start trigger will be ignored. Make sure to only send a frame start trigger when the camera is in "waiting for frame start trigger" status. For information about possibilities of getting informed about the "waiting for frame trigger" status, see the Acquisition Monitoring Tools section.

- When the next frame start trigger is received, the camera checks the states of input lines 1 and 2. Because the states have not changed the parameter values of sequence set 1 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the states of input lines 1 and 2. The states of both input lines are found to be low. This corresponds to the address of sequence set 0. Accordingly, sequence set 0 is selected. The parameter values of sequence set 0 are used for the image acquisition.
- When the next frame start trigger is received, the camera checks the states of input lines 1 and 2. Input line 1 is found to be low and input line 2 is found to be high. This corresponds to the address of sequence set 2. Accordingly, sequence set 2 is selected. The parameter values of sequence set 0 are used for the image acquisition.
- The sequence sets for the next five frame start triggers are selected and used according to the scheme that applied to the preceding frame start triggers.
- While frame exposure and readout for the fifth frame start trigger are in progress, the sequencer feature is disabled. The complete frame is transmitted. The sequencer parameter values in the active set return to the values that existed before the sequencer feature was enabled.

Use Case: Operation in free selection sequence advance mode.
Sequence sets are selected at will. The selection is controlled by the states of the input lines.

Settings: Sequence Set Total Number = 4
Input line 1 (not set for invert) sets bit 0 of the sequence set address.
Input line 2 (not set for invert) sets bit 1of the sequence set address.

= camera is waiting for a frame start trigger

▼ = camera selects a sequence set as the current sequence set

0 = current sequence set that is used for the image acquisition (the sequence set index number is indicated)

= frame exposure and readout

= frame transmission

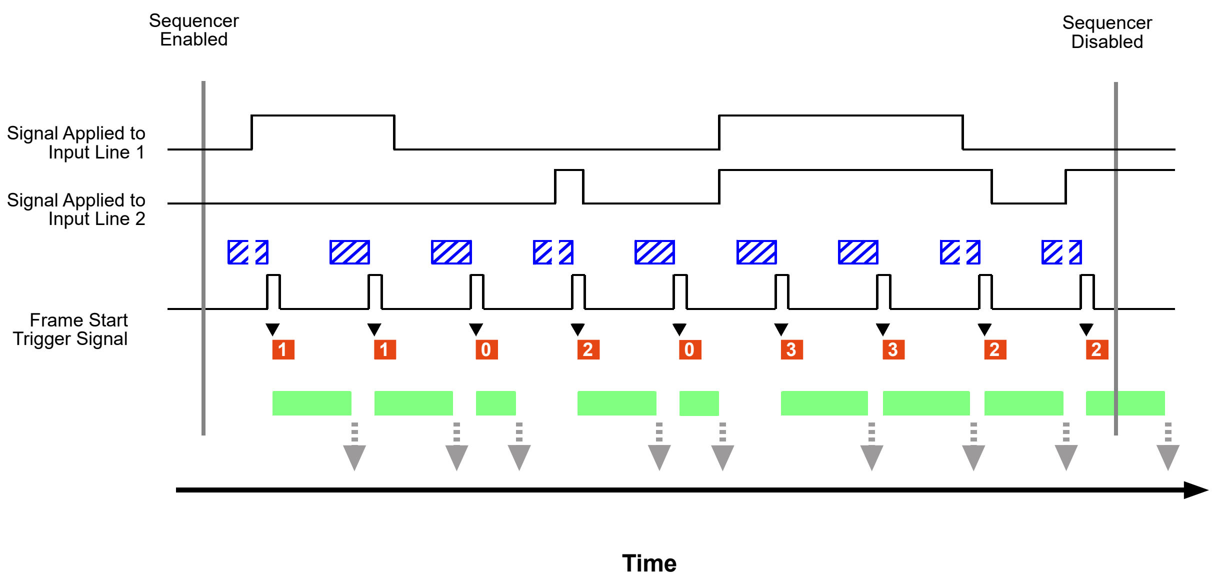


Fig. 59: Sequencer in Free Selection Mode

Operating the Sequencer Using Basler pylon

You can use the pylon API to set the parameters for operating the sequencer in Free Selection sequence advance mode from within your application software.

The following code snippet illustrates enabling and disabling the sequencer. The example assumes that sequence sets were previously configured and are currently available in the camera's memory.

```
// Enable the sequencer feature
Camera.SequenceEnable.SetValue( true );
```

```
// Disable the sequencer feature
Camera.SequenceEnable.SetValue( false );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

Selecting Sequence Sets

Each sequence set is identified by a sequence set index number, starting from zero. The states of the input lines select between the sequence sets by setting the (big endian) sequence set addresses. The addresses are simply the binary expressions of the sequence set index numbers. A maximum of four sequence sets can be used:

Sequence Set Address		**Related Sequence Set**
Bit 1	**Bit 0**	
0	0	Sequence Set 0
0	1	Sequence Set 1
1	0	Sequence Set 2
1	1	Sequence Set 3

Table 14: Sequence Set Addresses and Related Sequence Sets

The Sequence Set Total Number parameter specifies the total number of sequence sets that will be available. The parameter also specifies the length of the settable sequence set address.

- If the Sequence Set Total Number parameter is set to two:
  Bit 0 of the binary sequence set index number (see Table 14) can be set. When the bit is set to 0, sequence set 0 will be selected and when the bit is set to 1, sequence set 1 will be selected (see Table 14).

  You can use the states of either input line 1 or input line 2 to set bit 0.
  - If the input line is not set for invert the high state of the input line will set bit 0 to 1 and the low state will set bit 0 to 0.
  - If the input line is set for invert the low state of the input line will set bit 0 to 1 and the high state will set bit 0 to 0.
- If the Sequence Set Total Number parameter is set to higher than two:
  Bits 1 and 0 of the binary sequence set index number (see Table 14) can be set. When e.g. bit 0 is set to 0 and bit 1 is set to 1, sequence set 2 will be selected (see Table 14)

  You can use the states of input line 1 to set bit 0 and the states of input line 2 to set bit 1 or vice versa.
  - If the input lines are not set for invert the high states of the input lines will set the bits to 1 and the low states will set the bits to 0.
  - If the input lines are set for invert the low states of the input lines will set the bits to 1 and the high states will set the bits to 0.

For information about setting an input line for invert, see Section 3.1.4 on page 41.

9.11.3.2 Configuration

Configuring Sequence Sets and Advance Control

Use the following procedure for populating sequence sets and setting the source for sequence set advance:

1. Make sure that the sequencer feature is disabled.
2. Set the Sequence Advance Mode parameter to Free Selection.
3. Set the Sequence Set Total Number parameter. The maximum number is 4.
4. Select the sequence set address bits and set the input lines that will act as the control sources:
   - If the Set Total Number parameter was set to two:
     a. Bit 0 will be selected by default as the sequence set address bit. Set input line 1 or input line 2 as the control source for setting bit 0.
   - If the Set Total Number parameter was set to higher than two:
     a. Select bit 0 of the sequence set address bit.
     b. Set input line 1 or input line 2 as the control source for setting bit 0.
     c. Select bit 1 of the sequence set address bit.
     d. Set the input line as the control source for setting bit 1: Chose the input line not used for setting bit 0.
5. Use the Sequence Set Index parameter to select a sequence set index number for the sequence set currently being populated. The available numbers are 0 through 3.
6. Set up your first acquisition scenario (i.e., lighting, object positioning, etc.)
7. Adjust the camera parameters to get the best image quality with this scenario (you are adjusting the parameters in the active set).
8. Execute the Sequence Set Store command to copy the sequence parameter values currently in the active set into the selected sequence set. (Any existing parameter values in the sequence set will be overwritten.)
9. Repeat the above steps for the other sequence sets, starting from step 5.

Configuring Sequence Sets and Advance Control Using Basler pylon

You can use the pylon API to set the parameters for configuring sequence sets from within your application software and make settings for their selection when images are acquired.

The following code snippet gives example settings. It illustrates using the API to set the free selection sequence advance mode with line 1 as the control source for bit 0 and line 2 as the control source for bit 1 of the sequence set address, set the total number of sequence sets to 3, and populate sequence sets 0 through 2 by storing the sequence parameter values from the active set in the sequence sets:

```
// Disable the sequencer feature
Camera.SequenceEnable.SetValue( false );
```

```
// Set the Free Selection sequence advance mode
Camera.SequenceAdvanceMode.SetValue(
SequenceAdvanceMode_FreeSelection );

// Set the total number of sequence sets
Camera.SequenceSetTotalNumber.SetValue( 3 );

// Set line 1 as the control source for setting sequence set address
// bit 0
Camera.SequenceAddressBitSelector.SetValue(
SequenceAddressBitSelector_Bit0 );
Camera.SequenceAddressBitSource.SetValue(
SequenceAddressBitSource_Line1 );

// Set line 2 as the control source for setting sequence set address
// bit 1
Camera.SequenceAddressBitSelector.SetValue(
SequenceAddressBitSelector_Bit1 );
Camera.SequenceAddressBitSource.SetValue(
SequenceAddressBitSource_Line2 );

// Select sequence set with index number 0
Camera.SequenceSetIndex.SetValue( 0 );

// Set up the first acquisition scenario (lighting, object position,
// etc.) and adjust the camera parameters for the best image quality.
// Store the sequence parameter values from the active set in the
// selected sequence set
Camera.SequenceSetStore.Execute( );

// Select sequence set with index number 1
Camera.SequenceSetIndex.SetValue( 1 );

// Set up the second acquisition scenario (lighting, object position,
// etc) and adjust the camera parameters for the best image quality.

// Store the sequence parameter values from the active set in the
// selected sequence set
Camera.SequenceSetStore.Execute( );
```

```
// Select sequence set with index number 2
Camera.SequenceSetIndex.SetValue( 2 );

// Set up the third acquisition scenario (lighting, object position,
// etc.) and adjust the camera parameters for the best image quality.
// Store the sequence parameter values from the active set in the
// selected sequence set
Camera.SequenceSetStore.Execute( );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

9.12 Binning

The binning feature is only available on the monochrome cameras.

Binning increases the camera's response to light by summing the charges from adjacent pixels into one pixel. Two types of binning are available: vertical binning and horizontal binning.

With vertical binning, adjacent pixels from 2 lines, 3 lines, or a maximum of 4 lines in the imaging sensor array are summed and are reported out of the camera as a single pixel. Figure 60 illustrates vertical binning.

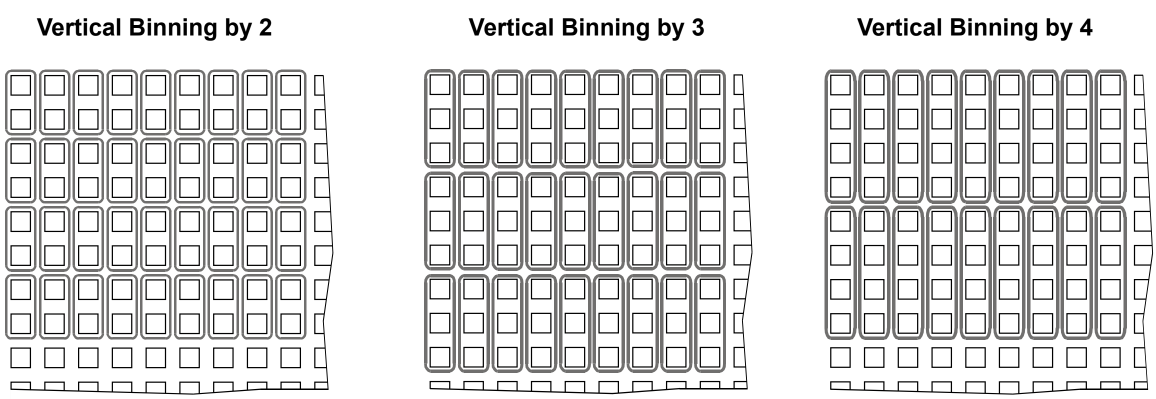


Fig. 60: Vertical Binning

With horizontal binning, adjacent pixels from 2 columns, 3 columns, or a maximum of 4 columns are summed and are reported out of the camera as a single pixel. Figure 61 illustrates horizontal binning.

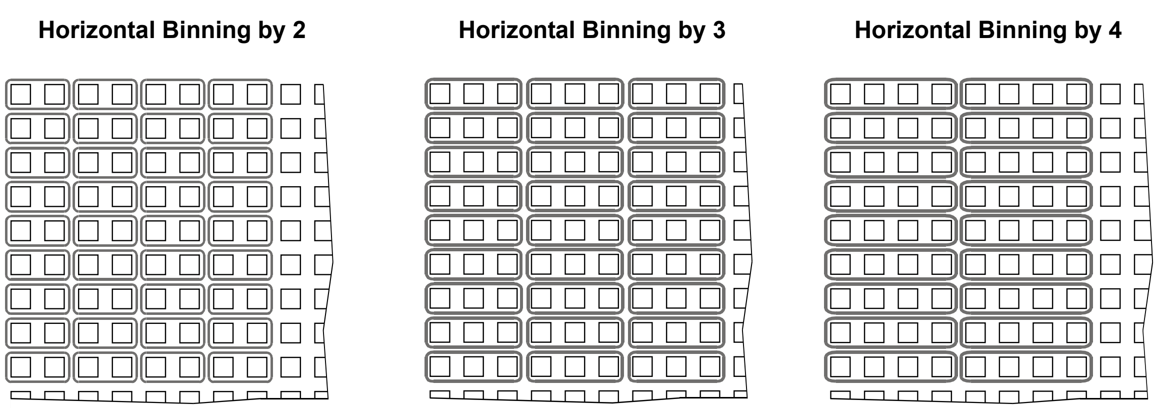


Fig. 61: Horizontal Binning

You can combine vertical and horizontal binning. This, however, may cause objects to appear distorted in the image. For more information on possible image distortion due to combined vertical and horizontal binning, see the next section.

Setting Binning Using Basler pylon

You can enable vertical binning by setting the Binning Vertical parameter. Setting the parameter's value to 2, 3, or 4 enables vertical binning by 2, vertical binning by 3, or vertical binning by 4 respectively. Setting the parameter's value to 1 disables vertical binning.

You can enable horizontal binning by setting the Binning Horizontal parameter. Setting the parameter's value to 2, 3, or 4 enables horizontal binning by 2, horizontal binning by 3, or horizontal binning by 4 respectively. Setting the parameter's value to 1 disables horizontal binning.

You can use the pylon API to set the Binning Vertical or the Binning Horizontal parameter value from within your application software. The following code snippet illustrates using the pylon API to set the parameter values:

```
// Enable vertical binning by 2
Camera.BinningVertical.SetValue( 2 );

// Enable horizontal binning by 4
Camera.BinningHorizontal.SetValue( 4 );

// Disable vertical and horizontal binning
Camera.BinningVertical.SetValue( 1 );
Camera.BinningHorizontal.SetValue( 1 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting Binning Using Direct Register Access

To enable vertical binning via direct register access:

- Set the value of the Binning Vertical register.

Setting the register's value to 2, 3, or 4 enables vertical binning by 2, vertical binning by 3, or vertical binning by 4 respectively. Setting the register's value to 1 disables vertical binning.

To enable horizontal binning via direct register access:

- Set the value of the Binning Horizontal register.

Setting the register's value to 2, 3, or 4 enables horizontal binning by 2, horizontal binning by 3, or horizontal binning by 4 respectively. Setting the register's value to 1 disables horizontal binning.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.12.1 Considerations When Using Binning

Increased Response to Light

Using binning can greatly increase the camera's response to light. When binning is enabled, acquired images may look overexposed. If this is the case, you can reduce the lens aperture, reduce the intensity of your illumination, reduce the camera's exposure time setting, or reduce the camera's gain setting.

Reduced Resolution

Using binning effectively reduces the resolution of the camera's imaging sensor. For example, the sensor in the avA1000-120km camera normally has a maximum resolution of 1024 (H) x 1024 (V) pixels. If you set this camera to use horizontal binning by 3 and vertical binning by 3, the effective maximum resolution of the sensor is reduced to 341 (H) by 341 (V). (Note that the 1024 pixel dimensions of the sensor are not evenly divisible by 3, so we rounded down to the nearest whole number.)

Possible Image Distortion

Objects will only appear undistorted in the image if the numbers of binned lines and columns are equal. With all other combinations, the imaged objects will appear distorted. If, for example, vertical binning by 2 is combined with horizontal binning by 4 the widths of the imaged objects will appear shrunken by a factor of 2 compared to the heights.

If you want to preserve the aspect ratios of imaged objects when using binning, you must use vertical and horizontal binning where equal numbers of lines and columns are binned, e.g. vertical binning by 3 combined with horizontal binning by 3.

Binning's Effect on AOI Settings

When you have the camera set to use binning, keep in mind that the settings for your area of interest (AOI) will refer to the binned lines and columns in the sensor and not to the physical lines in the sensor as they normally would. Another way to think of this is by using the concept of a "virtual sensor." For example, assume that you are using an avA1000-120km camera set for 3 by 3 binning as described above. In this case, you would act as if you were actually working with a 341 column by 341 line sensor when setting your AOI parameters. The maximum AOI width would be 341 and the maximum AOI height would be 341. When you set the Width for the AOI, you will be setting this value in terms of virtual sensor columns. And when you set the Height for the AOI, you will be setting this value in terms of virtual sensor lines.

For more information about the area of interest (AOI) feature, see Section 9.7 on page 167

Binning's Effect on the Maximum Allowed Frame Rate

Using vertical binning will increase the camera's maximum allowed frame rate.

For more information about determining the camera maximum allowed frame rate, see Section 6.8 on page 112.

9.13 Mirror Imaging

The camera's reverse X and reverse Y functions let you flip the captured images horizontally and/or vertically before they are transmitted from the camera.

Note that the reverse X and reverse Y functions may both be enabled at the same time if so desired.

9.13.1 Reverse X

The reverse X feature is a horizontal mirror image feature. When the reverse X feature is enabled, the pixel values for each line in a captured image will be swapped end-for-end about the line's center. This means that for each line, the value of the first pixel in the line will be swapped with the value of the last pixel, the value of the second pixel in the line will be swapped with the value of the next-to-last pixel, and so on.

Figure 62 shows a normal image on the left and an image captured with reverse X enabled on the right.

Normal Image **Reverse X Mirror Image**

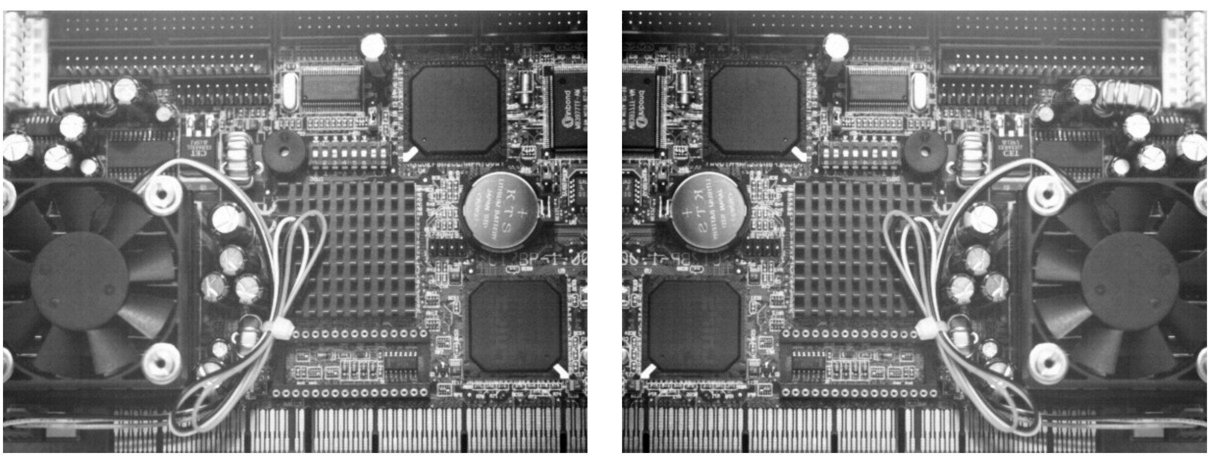


Fig. 62: Reverse X Mirror Imaging

> On color models of the camera, when either the reverse X feature or the reverse Y feature or both are used, the alignment of the color filter to the image remains Bayer GR. The camera includes a mechanism that keeps the filter alignment constant when these features are used. For more information about the color filter, see Section 7.1 on page 123.

The Effect of Reverse X on the Auto Function AOIs

If you are using the camera's auto functions, you should be aware of the effect that using the reverse X feature will have on the auto function AOIs. When reverse X is used, the position of the auto function AOIs relative to the sensor remains the same. As a consequence, each auto function AOI will include a different portion of the captured image depending on whether or not the reverse X feature is enabled. Figure 63 shows the effect of that reverse X mirroring will have on the auto function AOIs.

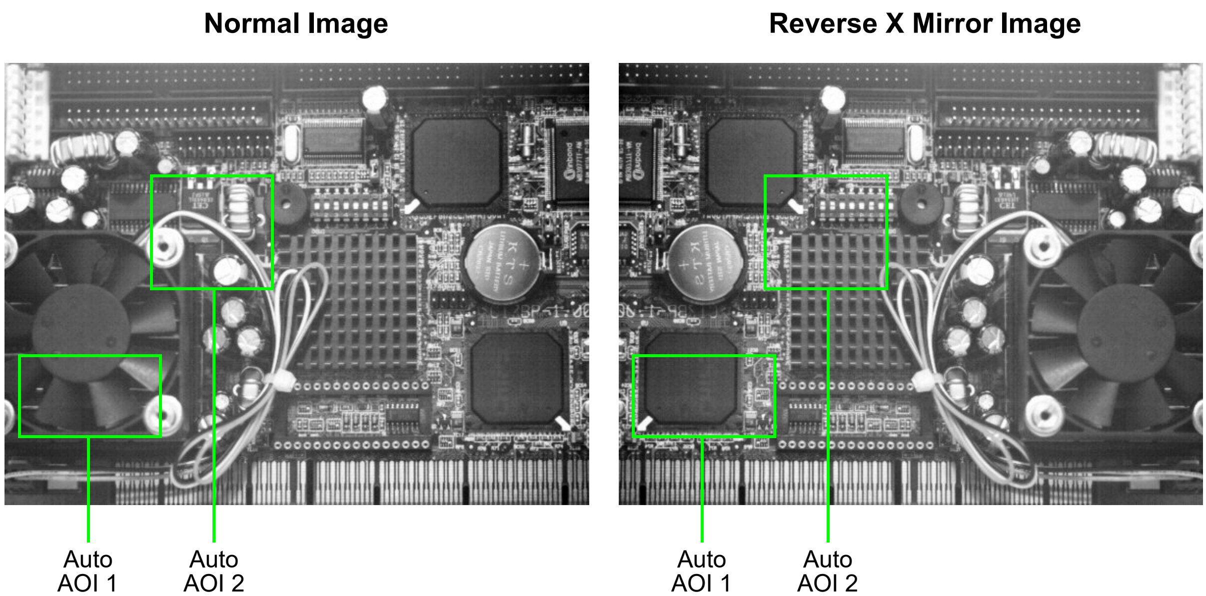


Fig. 63: Using Reverse X Mirror Imaging with Auto Functions Enabled

For more information about auto functions and auto function AOIs, see Section 9.8 on page 173.

9.13.2 Reverse Y

The reverse Y feature is a vertical mirror image feature. When the reverse Y feature is enabled, the lines in a captured image will be swapped top-to-bottom. This means that the top line in the image will be swapped with the bottom line, the next-to-top line will be swapped with the next-to-bottom line, and so on.

Figure 64 shows a normal image on the left and an image captured with reverse Y enabled on the right.

Normal Image **Reverse Y Mirror Image**

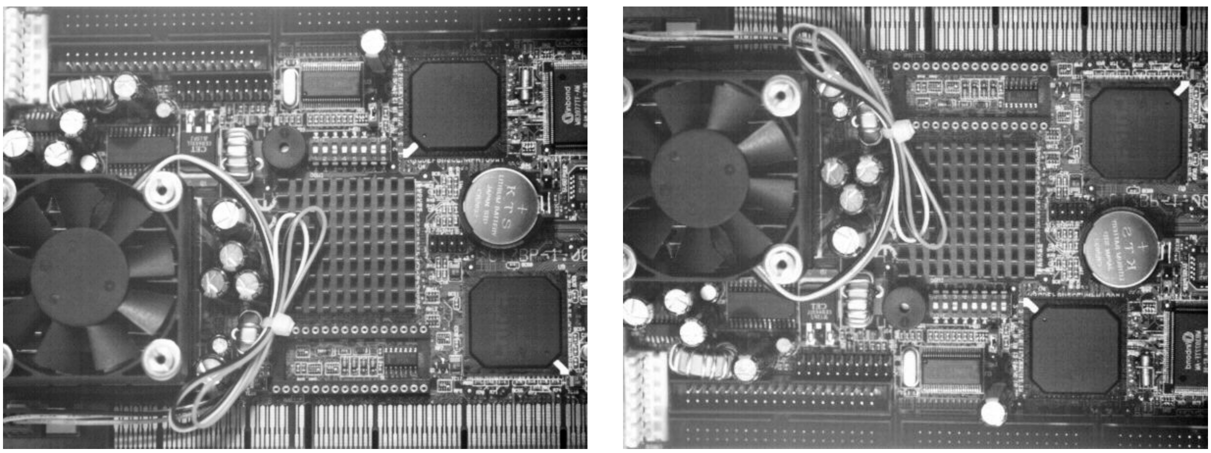


Fig. 64: Reverse Y Mirror Imaging

i	On color models of the camera, when either the reverse X feature or the reverse Y feature or both are used, the alignment of the color filter to the image remains Bayer GR. The camera includes a mechanism that keeps the filter alignment constant when these features are used. For more information about the color filter, see Section 7.1 on page 123.

The Effect of Reverse Y on the Auto Function AOIs

If you are using the camera's auto functions, you should be aware of the effect that using the reverse Y feature will have on the auto function AOIs. When reverse Y is used, the position of the auto function AOIs relative to the sensor remains the same. As a consequence, each auto function AOI will include a different portion of the captured image depending on whether or not the reverse Y feature is enabled. Figure 65 shows the effect of that reverse Y mirroring will have on the auto function AOIs.

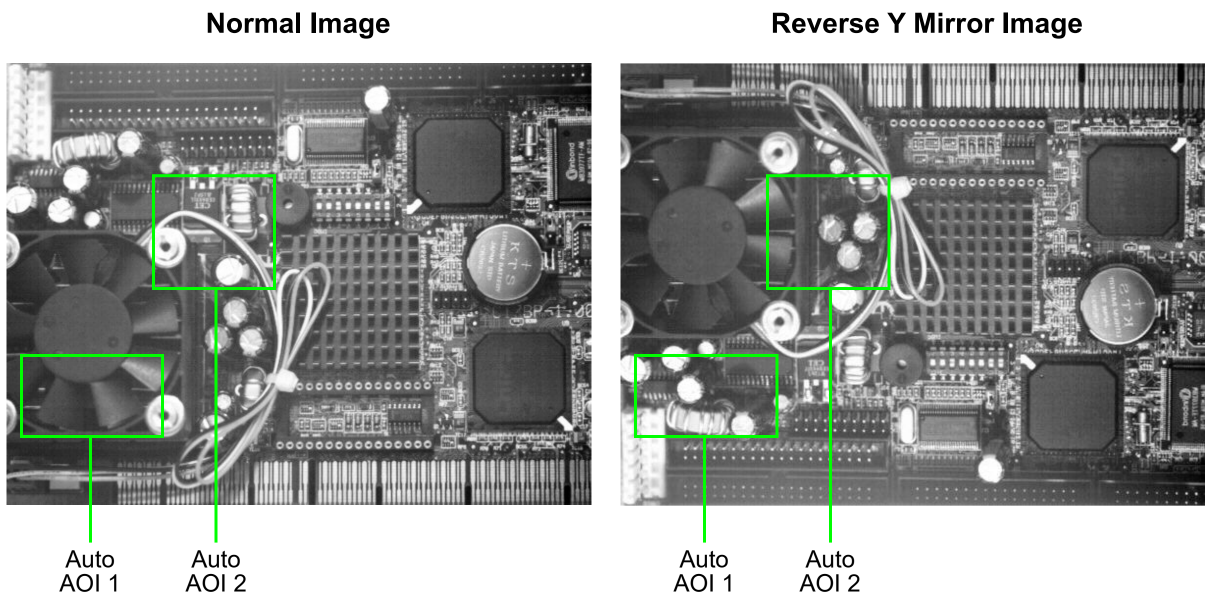


Fig. 65: Using Reverse Y Mirror Imaging with Auto Functions Enabled

For more information about auto functions and auto function AOIs, see Section 9.8 on page 173.

9.13.3 Enabling Reverse X and Reverse Y

Enabling Reverse X and Y Using Basler pylon

You can enable the reverse X and reverse Y features by setting the Reverse X and the Reverse Y parameter values. You can use the pylon API to set the parameter values from within your application software. The following code snippet illustrates using the pylon API to set the parameter values:

```
// Enable reverse X
Camera.ReverseX.SetValue(true);

// Enable reverse Y
Camera.ReverseY.SetValue(true);
```

You can also use the Basler pylon Viewer application to easily set the parameter.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Enabling Reverse X and Y Using Direct Register Access

To enable reverse X via direct register access:

- Set the value of the Reverse X register to 1 (enabled).

To enable reverse Y via direct register access:

- Set the value of the Reverse Y register to 1 (enabled).

For more information about changing settings via direct register access, see Section 5 on page 61.

9.14 Luminance Lookup Table

The type of electronics used on the camera allows the camera's sensor to acquire pixel values at a 12 bit depth. Normally, when a camera is set for a 12 bit pixel format, the camera transmits the actual 12 bit pixel values reported by the sensor.

The luminance lookup table feature lets you create a custom 12 bit to 12 bit lookup table that maps the actual 12 bit values output from the sensor to substitute 12 bit values of your choice. When the lookup table is enabled, the camera will replace the actual pixel values output from the sensor with the substitute values from the table.

The lookup table has 4096 indexed locations with a 12 bit value stored at each index. The values stored in the table are used like this:

- When the sensor reports that a pixel has an actual 12 bit value of 0, the substitute 12 bit value stored at index 0 will replace the actual pixel value.
- The numbers stored at indices 1 through 7 are not used.
- When the sensor reports that a pixel has an actual 12 bit value of 8, the substitute 12 bit value stored at index 8 will replace the actual pixel value.
- The numbers stored at indices 9 through 15 are not used.
- When the sensor reports that a pixel has an actual 12 bit value of 16, the substitute 12 bit value stored at index 16 will replace the actual pixel value.
- The numbers stored at indices 17 through 23 are not used.
- When the sensor reports that a pixel has an actual 12 bit value of 24, the substitute 12 bit value stored at index 24 will replace the actual pixel value.
- And so on.

As you can see, the table does not include a defined 12 bit substitute value for every actual pixel value that the sensor can report. If the sensor reports an actual pixel value that is between two values that have a defined substitute, the camera performs a straight line interpolation to determine the substitute value that it should use. For example, assume that the sensor reports an actual pixel value of 12. In this case, the camera would perform a straight line interpolation between the substitute values at index 8 and index 16 in the table. The result of the interpolation would be used by the camera as the substitute.

Another thing to keep in mind about the table is that index 4088 is the last index that will have a defined substitute value associated with it (the values at indices 4089 through 4095 are not used.) If the sensor reports an actual value greater than 4088, the camera will not be able to perform an interpolation. In cases where the sensor reports an actual value greater than 4088, the camera simply uses the 12 bit substitute value from index 4088 in the table.

i	There is only one lookup table. When the lookup table is enabled on color cameras, the single table is used for red, green, and blue pixel values. The values for the luminance lookup table are not saved in the user sets and are lost when the camera is reset or switched off. If you are using the lookup table feature, you must reenter the lookup table values after each camera startup or reset.

The advantage of the luminance lookup table feature is that it lets a user customize the response curve of the camera. The graphs below represent the contents of two typical lookup tables. The first graph is for a lookup table where the substitute values are designed so that the output of the camera increases linearly as the actual sensor output increases. The second graph is for a lookup table where the substitute values are designed so that the camera output increases quickly as the actual sensor output moves from 0 through 2048 and increases gradually as the actual sensor output moves from 2049 through 4096.

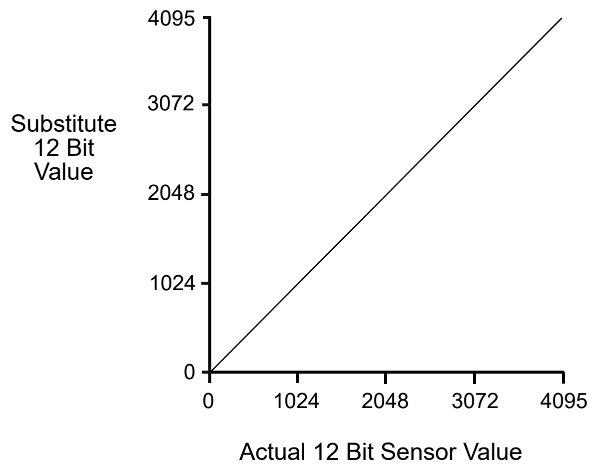


Fig. 66: Lookup Table with Values Mapped in a Linear Fashion

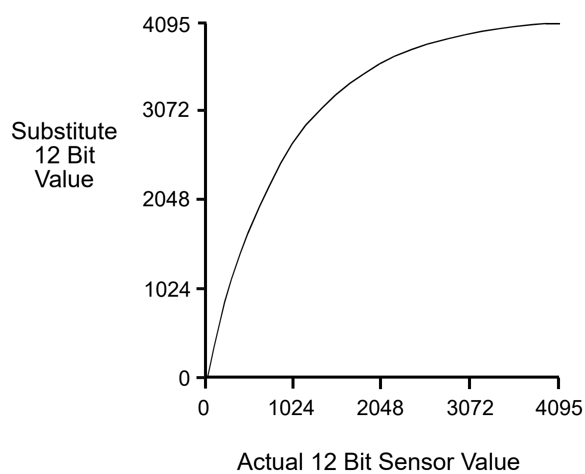


Fig. 67: Lookup Table with Values Mapped for Higher Camera Output at Low Sensor Readings

Using the Luminance Lookup Table to Get 10 Bit or 8 Bit Output

As mentioned above, when the camera is set for a 12 bit pixel format, the lookup table can be used to perform a 12 bit to 12 bit substitution. The lookup table can also be used in 12 bit to 10 bit or 12 bit to 8 bit fashion.

To use the table in 12 bit to 10 bit fashion, you enter 12 bit substitution values into the table and enable the table as you normally would. But instead of setting the camera for a 12 bit pixel format, you set the camera for a 10 bit format (such as Mono 10). In this situation, the camera will first use the values in the table to do a 12 bit to 12 bit substitution. It will then truncate the least significant 2 bits of the substitute value and will transmit the remaining 10 most significant bits.

To use the table in 12 bit to 8 bit fashion, you enter 12 bit substitution values into the table and enable the table as you normally would. But instead of setting the camera for a 12 bit pixel format, you set the camera for an 8 bit format (such as Mono 8). In this situation, the camera will first use the values in the table to do a 12 bit to 12 bit substitution. It will then truncate the least significant 4 bits of the substitute value and will transmit the remaining 8 most significant bits.

9.14.1 Entering LUT Values and Enabling the LUT

Entering Values and Enabling the LUT Using Basler pylon

You can enter values into the luminance lookup table (LUT) and enable the use of the lookup table by doing the following:

- Use the LUT Selector to select a lookup table. (Currently there is only one lookup table available, i.e., the "luminance" lookup table described above.)
- Use the LUT Index parameter to select an index number.
- Use the LUT Value parameter to enter the substitute value that will be stored at the index number that you selected in step 2.
- Repeat steps 2 and 3 to enter other substitute values into the table as desired.
- Use the LUT Enable parameter to enable the table.

You can use the pylon API to set the LUT Selector, the LUT Index parameter, and the LUT Value parameter from within your application software. The following code snippet illustrates using the pylon API to set the selector and the parameter values:

```
// Select the lookup table
Camera.LUTSelector.SetValue( LUTSelector_Luminance );

// Write a lookup table to the device.
// The following lookup table causes an inversion of the sensor values
// ( bright -> dark, dark -> bright )
for ( int i = 0; i < 4096; i += 8 )
{
   Camera.LUTIndex.SetValue( i );
```

```
    Camera.LUTValue.SetValue( 4095 - i );
}
// Enable the lookup table
Camera.LUTEnable.SetValue( true );
```

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Entering Values and Enabling the LUT Using Direct Register Access

When setting up the luminance lookup table via direct register access, two registers are involved: the LUT register and the LUT Enable register.

The LUT register is simply an array register that holds the 4096 12 bit values described earlier in this section. As a first step to using the lookup table feature. You must populate this register with 12 bit values.

Once the LUT register has been populated, you can enable the use of the lookup table by setting the value of the LUT Enable register to 1 (enabled).

For more information about changing settings via direct register access, see Section 5 on page 61.

9.15 User Defined Values

The camera can store five "user defined values". These five values are 32 bit signed integer values that you can set and read as desired. They simply serve as convenient storage locations for the camera user and have no impact on the operation of the camera.

The five values are designated as Value 1, Value 2, Value 3, Value 4, and Value 5.

Setting User Defined Values Using Basler pylon

Setting a user defined value using Basler pylon is a two step process:

- Set the User Defined Value Selector to Value 1, Value 2, Value 3, Value 4, or Value 5.
- Set the User Defined Value parameter to the desired value for the selected value.

You can use the pylon API to set the User Defined Value Selector and the User Defined Value parameter value from within your application software. The following code snippet illustrates using the pylon API to set the selector and the parameter value:

```
// Set user defined value 1
Camera.UserDefinedValueSelector.SetValue(
UserDefinedValueSelector_Value1 );
Camera.UserDefinedValue.SetValue( 1000 );

// Set user defined value 2
Camera.UserDefinedValueSelector.SetValue(
UserDefinedValueSelector_Value2 );
Camera.UserDefinedValue.SetValue( 2000 );

// Get the value of user defined value 1
Camera.UserDefinedValueSelector.SetValue(
UserDefinedValueSelector_Value1 );
int64_t UserValue1 = Camera.UserDefinedValue.GetValue();
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Setting User Defined Values Using Direct Register Access

To set the user defined values via direct register access:

- Set the value of the User Defined Value 1 register, the User Defined Value 2 register, the User Defined Value 3 register, the User Defined Value 4 register, or the User Defined Value 5 register as desired.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.16 Test Images

All cameras include the ability to generate test images. Test images are used to check the camera's basic functionality and its ability to transmit an image to the host computer. Test images can be used for service purposes and for failure diagnostics. For test images, the image is generated internally by the camera's logic and does not use the optics or the imaging sensor. Five test images are available (for color cameras six test images).

The Effect of Camera Settings on Test Images

When any of the test image is active, the camera's analog features such as gain, black level, and exposure time have no effect on the images transmitted by the camera. For test images 1, 2, and 3, and 6 the camera's digital features, such as the luminance lookup table, will also have no effect on the transmitted images. But for test images 4 and 5, the cameras digital features will affect the images transmitted by the camera. This makes test images 4 and 5 as good way to check the effect of using a digital feature such as the luminance lookup table.

Enabling a Test Image Using Basler pylon

With Basler pylon, the Test Image Selector is used to set the camera to output a test image. You can set the value of the Test Image Selector to enable one of the test images or to "test image off".

You can use the pylon API to set the Test Image Selector from within your application software. The following code snippets illustrate using the pylon API to set the selector:

```
// Set for no test image
Camera.TestImageSelector.SetValue( TestImageSelector_Off );

// Set for test image 1
Camera.TestImageSelector.SetValue( TestImageSelector_Testimage1 );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Enabling a Test Image Using Direct Register Access

To enable a test image via direct register access:

Set the value of the Test image Selector Register to Test Image 1, 2, 3, 4, 5 or 6 as desired.

To disable test images:

Set the value of the Test image Selector Register to Off.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.16.1 Test Image Descriptions

Test Image 1 - Fixed Diagonal Gray Gradient (8 bit)

This 8 bit fixed diagonal gray gradient test image is best suited for use when the camera is set for monochrome 8 bit output. The test image consists of fixed diagonal gray gradients ranging from 0 to 255.

If the camera is set for 8 bit output and is operating at full resolution, test image one will look similar to Figure 68.

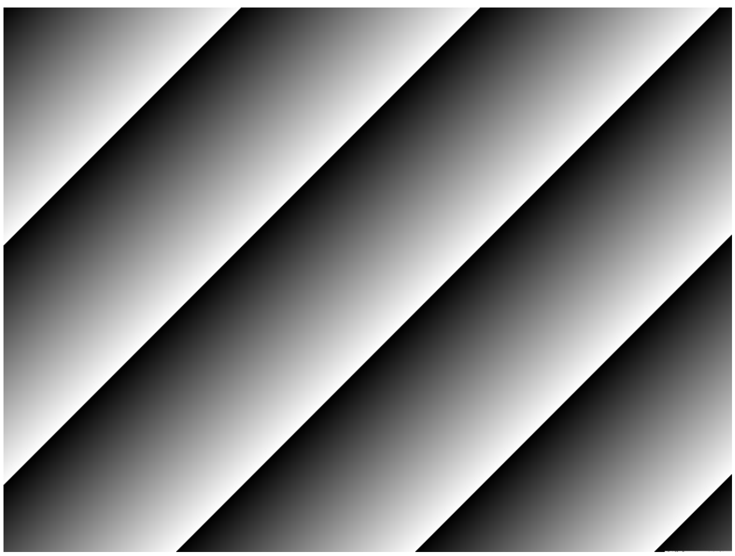


Fig. 68: Test Image 1

Test Image 2 - Moving Diagonal Gray Gradient (8 bit)

The 8 bit moving diagonal gray gradient test image is similar to test image 1, but it is not stationary. The image moves by one pixel from right to left whenever a new image acquisition is initiated. The test pattern uses a counter that increments by one for each new image acquisition.

Test Image 3 - Moving Diagonal Gray Gradient (10 bit or 12 bit)

Test image 3 is a moving diagonal gray gradient test image similar to test image 2, but it is a 10 bit pattern if the camera is set to output pixel data at 10 bit depth or a 12 bit pattern if the camera is set to output pixel data at 12 bit depth. The image moves by one pixel from right to left whenever a new image acquisition is initiated. The test pattern uses a counter that increments by one for each new image acquisition.

Test Image 4 - Moving Diagonal Gray Gradient Feature Test (8 bit)

The basic appearance of test image 4 is similar to test image 2 (the 8 bit moving diagonal gray gradient image). The difference between test image 4 and test image 2 is this: if a camera feature that involves digital processing is enabled, test image 4 **will** show the effects of the feature while test image 2 **will not**. This makes test image 4 useful for checking the effects of digital features such as the luminance lookup table.

Test Image 5 - Moving Diagonal Gray Gradient Feature Test (10 bit or 12 bit)

The basic appearance of test image 5 is similar to test image 3 (the 10 or 12 bit moving diagonal gray gradient image). The difference between test image 5 and test image 3 is this: if a camera feature that involves digital processing is enabled, test image 5 **will** show the effects of the feature while test image 3 **will not**. This makes test image 5 useful for checking the effects of digital features such as the luminance lookup table

Test Image 6 - Moving Diagonal Color Gradient

Test image 6 is an 8 bit fixed diagonal color gradient test image. Test image 6 is available on color cameras only. When a color camera is set for test image 6, it delivers pixel data in the Bayer GR 8 format.

This test image can be used to test a color camera's basic ability to transmit a color image.

It can also be used to test whether your frame grabber is correctly set to interpolate images transmitted in the Bayer GR 8 format. If the colors in the images from your frame grabber do not exactly match the colors in test image 6 as shown below, then your frame grabber is incorrectly set.

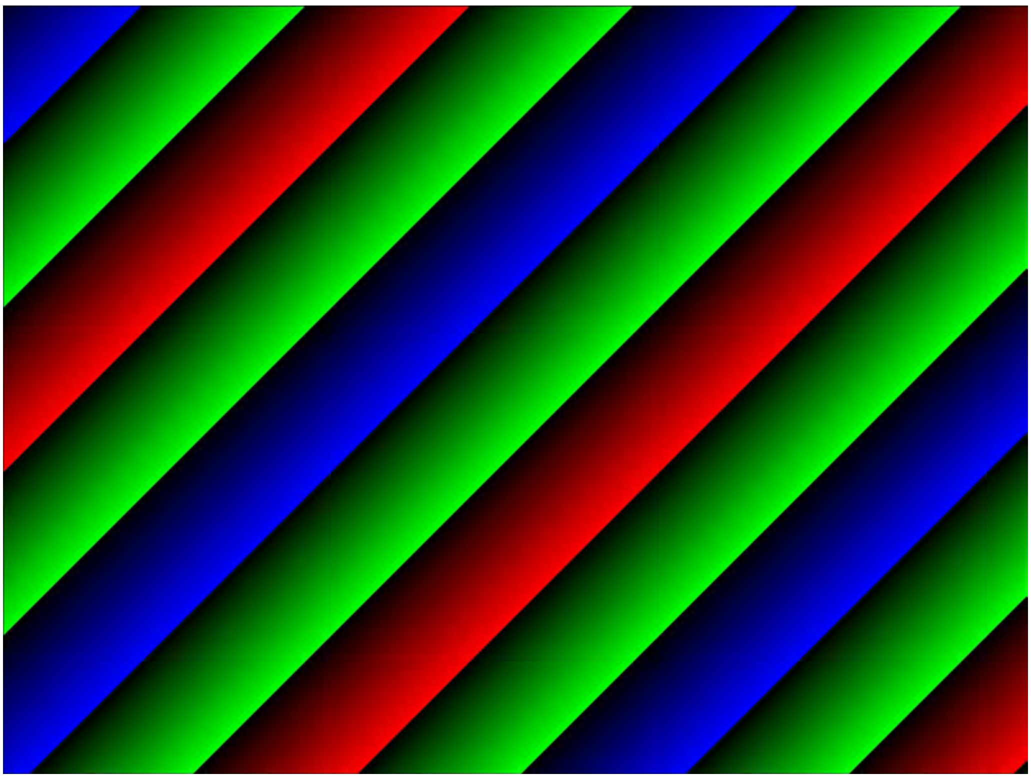


Fig. 69: Test Image 6

9.17 Device Information Parameters

Each camera includes a set of "device information" parameters. These parameters provide some basic information about the camera. The device information parameters include:

- Device Vendor Name (read only) - contains the name of the camera’s vendor. This string will always indicate Basler as the vendor.
- Device Model Name (read only) - contains the model name of the camera, for example, avA1000-120km.
- Device Manufacturer Info (read only) - can contain some information about the camera manufacturer. This string usually indicates "none".
- Device Version (read only) - contains the device version number for the camera. This is usually the material number of the device.
- Device Firmware Version (read only) - contains the version of the firmware the camera.
- Device ID (read only) - typically contains the serial number of the camera.
- Device User ID (read / write) - is used to assign a user defined name to a device. This name will be displayed in the Basler pylon Viewer and the Basler pylon CL Configurator. The name will also be visible in the "friendly name" field of the device information objects returned by pylon’s device enumeration procedure.
- Device Scan Type (read only) - contains the scan type of the camera, for example, line scan or area scan. The aviator will always indicate area scan.
- Sensor Width (read only) - contains the physical width of the sensor in pixels.
- Sensor Height (read only) - contains the physical height of the sensor.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Working with Device Information Parameters Using Basler pylon

You can use the pylon API to read the values for all of the device information parameters or set the value of the Device User ID parameter from within your application software. The following code snippets illustrate using the pylon API to read the parameters or write the Device User ID:

```
// Read the Vendor Name parameter
Pylon::String_t vendorName = Camera.DeviceVendorName.GetValue();

// Read the Model Name parameter
Pylon::String_t modelName = Camera.DeviceModelName.GetValue();

// Read the Manufacturer Info parameter
Pylon::String_t manufacturerInfo =
Camera.DeviceManufacturerInfo.GetValue();

// Read the Device Version parameter
Pylon::String_t deviceVersion = Camera.DeviceVersion.GetValue();
```

```
// Read the Firmware Version parameter
Pylon::String_t firmwareVersion =
Camera.DeviceFirmwareVersion.GetValue();

// Read the Device ID parameter
Pylon::String_t deviceID = Camera.DeviceFirmwareVersion.GetValue();

// Write and read the Device User ID
Camera.DeviceUserID = "custom name";
Pylon::String_t deviceUserID = Camera.DeviceUserID.GetValue();

// Read the Sensor Width parameter
int64_t sensorWidth = Camera.SensorWidth.GetValue();

// Read the Sensor Height parameter
int64_t sensorHeight = Camera.SensorHeight.GetValue();

// Read the Max Width parameter
int64_t maxWidth = Camera.WidthMax.GetValue();

// Read the Max Height parameter
int64_t maxHeight = Camera.HeightMax.GetValue();
```

You can also use the Basler pylon Viewer application to easily read the parameters and to read or write the Device User ID.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Working with Device Information Parameters Using Direct Register Access

When working with the camera via direct register access, you can do the following:

- Read the value in the Device Vendor Name register.
- Read the value in the Device Model Name register.
- Read the value in the Device Manufacturer Info register.
- Read the value in the Device Version register.
- Read the value of the Device Firmware Version register.
- Read the value in the Device ID register.
- Read the value in or set the value of the Device User ID register.
- Read the value in the Device Scan Type register.
- Read the value in the Sensor Width register.
- Read the value in the Sensor Height register.

For more information about reading or changing settings via direct register access, see Section 5 on page 61.

9.18 Imaging Sensor Temperature Monitoring and Over Temperature Detection

9.18.1 Reading the Imaging Sensor Temperature

The camera is equipped with a temperature sensor mounted on the imaging sensor board. The temperature sensor lets you read the current temperature of the camera's imaging sensor board in degrees C.

Reading the Imaging Sensor Board Temperature Using Basler pylon

You can use the pylon API to read the imaging sensor board temperature in degrees C from within your application software. Reading the temperature using Basler pylon is a two step process:

- Select the imaging sensor board temperature sensor.
- Read the temperature.

The following code snippet illustrates using the pylon API to read the temperature in degrees C:

```
// Select the imaging sensor board temperature sensor
Camera.TemperatureSelector.SetValue ( TemperatureSelector_Sensorboard
);

// Read the imaging sensor board temperature
double imgSensorTemp = Camera.TemperatureAbs.GetValue();
```

You can also use the Basler pylon Viewer application to easily read the temperature.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Reading the Imaging Sensor Board Temperature Using Direct Register Access

To read the temperature of the imaging sensor board in degrees C via direct register access:

- Read the value of the Device Temperature Sensor Board register.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.18.2 Imaging Sensor Temperature Conditions

The temperature sensor is used to monitor the temperature of the camera's imaging sensor board.

The camera also has imaging sensor over temperature protection.

If the **temperature of the imaging sensor board** rises above 75° C, an **over temperature condition** will be detected and the circuitry on the imaging sensor board will switch off. In this situation, you will still be able to communicate with the camera, however, the camera will not be able to acquire or transmit images. The imaging sensor board circuitry will remain off until its temperature falls below 75° C. Once the temperature is below 75°, the error condition will clear. After the error condition clears, the camera must be restarted before it will begin operating normally.
For information on how to check for an over temperature condition, see below.

Checking for an Imaging Sensor Board Overtemp Condition Using Basler pylon

You can use the pylon API from within your application software to check whether the imaging sensor board is currently in an overtemp condition. Checking for the temperature condition using Basler pylon is a two step process:

- Select the imaging sensor board temperature sensor.
- Check for an overtemp condition.

The following code snippet illustrates using the pylon API to read the temperature in degrees C:

```
// Select the imaging sensor board temperature sensor
Camera.TemperatureSelector.SetValue ( TemperatureSelector_Sensorboard
);
// Check for an imaging sensor board overtemp condition
bool imgSensorOvertemp = Camera.OverTemperature.GetValue();
```

You can also use the Basler pylon Viewer application to easily check for an overtemperature condition.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Checking for an Imaging Sensor Board Overtemp Condition Using Direct Register Access

To check via direct register access whether the camera is currently in an imaging sensor board over temperature condition or not via direct register access:

- Read the value of the Over Temperature Sensor Board register.

For more information about checking settings via direct register access, see Section 5 on page 61.

9.19 Configuration Sets

A configuration set is a group of values that contains all of the parameter settings needed to control the camera. There are three basic types of configuration sets: the active set, the default set, and the user set.

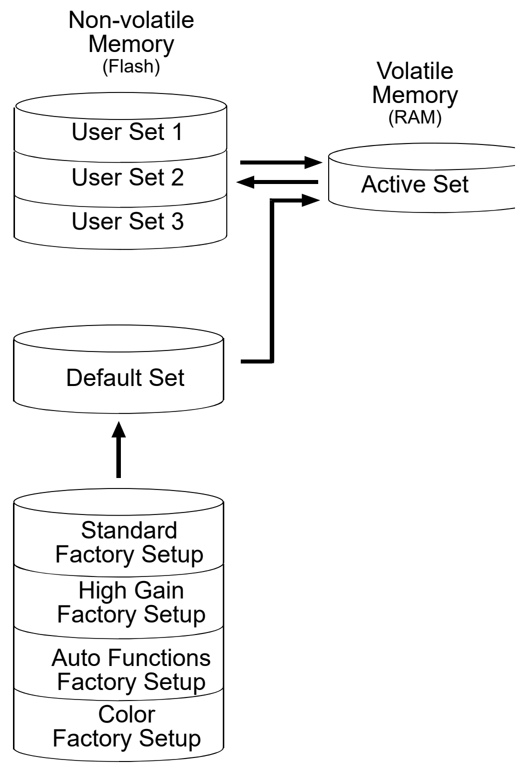


Fig. 70: Configuration Sets

The Active Set

The active set contains the camera's current parameter settings and thus determines the camera's performance, that is, what your image currently looks like. When you change parameter settings using the pylon API or the pylon Viewer, you are making changes to the active set. The active set is located in the camera's volatile memory and the settings are lost if the camera is reset or if power is switched off.

The Default Set

When a camera is manufactured, numerous tests are performed on the camera and four factory optimized setups are determined. The four factory optimized setups are:

- The Standard Factory Setup - is optimized for average conditions and will provide good camera performance in many common applications. In the standard factory setup, the gain is set to a low value, and all auto functions are set to off.
- The High Gain Factory Setup - is similar to the standard factory setup, but the gain is set to + 6 dB.
- The Auto Functions Factory Setup - is similar to the standard factory setup, but the Exposure Auto auto function is enabled and set to the continuous mode of operation.
- The Color Factory Setup - is optimized to yield the best color fidelity with daylight lighting.

The factory setups are saved in permanent files in the camera's non-volatile memory. They are not lost when the camera is reset or switched off and they cannot be changed.

You can select one of the factory setups to be the camera's "default set". Instructions for selecting which factory setup will be used as the default set appear later in the Configuration Sets section. Note that your selection of which factory setup will serve as the default set will not be lost when the camera is reset or switched off.

When the camera is running, the default set can be loaded into the active set. The default set can also be designated as the "startup" set, i.e., the set that will be loaded into the active set whenever the camera is powered on or reset. Instructions for loading the default set into the active set and for designating which set will be the startup set appear later in the Configuration Sets section.

User Set

The active configuration set is stored in the camera's volatile memory and the settings are lost if the camera is reset or if power is switched off. The camera can save most of the settings from the current active set to a reserved area in the camera's non-volatile memory. A configuration set that has been saved in the non-volatile memory is not lost when the camera is reset or switched off. There is one reserved area in the camera's non-volatile memory available for saving a configuration set. A configuration set saved in the reserved area is commonly referred to as a "user set".

The three available user sets are called User Set 1, User Set 2, and User Set 3.

When the camera is running, the saved user set can be loaded into the active set. The saved user set can also be designated as the "startup" set, i.e., the set that will be loaded into the active set whenever the camera is powered on or reset. Instructions for loading a saved user set into the active set and for designating which set will be the startup set appear later in the Configuration sets section.

i	The values for the luminance lookup table are not saved in the user set and are lost when the camera is reset or switched off. If you are using the lookup table feature, you must reenter the lookup table values after each camera startup or reset.

Designating a Startup Set

You can designate the default set or the user set as the "startup" set. The designated startup set will automatically be loaded into the active set whenever the camera starts up at power on or after a reset. Instructions for designating the startup set appear later in the Configuration Sets section.

9.19.1 Selecting a Factory Setup as the Default Set

When the camera is delivered, the Auto Functions Factory Setup will be selected as the default set. You can, however, select any one of the four factory setups to serve as the default set.

Selecting a Factory Setup Using pylon

To select which factory setup will serve as the default set using Basler pylon:

Set the Default Set Selector to the Standard Factory Setup, High Gain Factory Setup, Auto Functions Factory Setup or Color Factory Setup.

You can set the Default Set Selector from within your application software by using the pylon API. The following code snippet illustrates using the pylon API to set the selector:

```
// If you want to select the Standard Factory Setup:
   Camera.DefaultSetSelector.SetValue(DefaultSetSelector_Standard);
// If you want to select the High Gain Factory Setup:
   Camera.DefaultSetSelector.SetValue(DefaultSetSelector_HighGain);
// If you want to select the Auto Functions Factory Setup:
   Camera.DefaultSetSelector.SetValue(DefaultSetSelector_AutoFunctions);
```

Selecting a Factory Setup Using Direct Register Access

To select which factory setup will serve as the default set via direct register access:

- Set the Default Set Selector register to the Standard Factory Setup, High Gain Factory Setup or Auto Functions Factory Setup.

i	Selecting which factory setup will serve as the default set is only allowed when the camera is idle, i.e. when it is not acquiring images continuously or does not have a single image acquisition pending. Selecting the standard factory setup as the default set and then loading the default set into the active set is a good course of action if you have grossly misadjusted the settings in the camera and you are not sure how to recover. The standard factory setup is optimized for use in typical situations and will provide good camera performance in most cases.

9.19.2 Saving User Sets

You can save the current parameter set being used by the camera (i.e., the "active" set in the camera's volatile memory) to user set 1, user set 2, or user set 3. The user sets are stored in the camera's non-volatile memory and will be retained when the camera power is switched off or the camera is reset. When you save the active set to a user set, any parameter data already in that user set will be overwritten.

Saving User Sets Using Basler pylon

Using Basler pylon to save the current active set to a user set in the camera's non-volatile memory is a several step process:

- Make changes to the camera's settings until the camera is operating in a manner that you would like to save.
- Set the User Set Selector to User Set 1, User Set 2, or User Set 3 as desired.
- Execute a User Set Save command to save the active set to the selected user set.

Saving an active set to a user set in the camera's non-volatile memory will overwrite any parameters that were previously saved in that user set.

You can use the pylon API to set the User Set Selector and to execute the User Set Save command from within your application software. The following code snippet illustrates using the pylon API to set the selector and execute the command:

```
Camera.UserSetSelector.SetValue( UserSetSelector_UserSet1 );
Camera.UserSetSave.Execute( );
```

You can also use the Basler pylon Viewer application to easily set the parameters.

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Saving User Sets Using Direct Register Access

To save the current active set to a user set in the camera's non-volatile memory via direct register access:

- Make changes to the camera's settings until the camera is operating in a manner that you would like to save.
- Set the value of the User Set Selector register to User Set 1, 2, or 3 as desired.
- Set the value of the User Set Save register to 1.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.19.3 Loading a Saved User Set or the Default Set into the Active Set

If you have saved a configuration set into one of the user sets in the camera's non-volatile memory, you can load the saved user set into the camera's active set. When you do this, the parameters stored in the user set overwrite the parameters in the active set. Since the settings in the active set control the current operation of the camera, the settings from the loaded user set will now be controlling the camera.

You can also load the default set into the camera's active set.

Loading a user set or the default set into the active set is only allowed when the camera is idle, i.e. when it is not acquiring an image.

Assuming that you have selected the standard factory setup as the default set, loading the default set into the active set is a good course of action if you have grossly misadjusted the settings in the camera and you are not sure how to recover. The standard factory setup is optimized for use in typical situations and will provide good camera performance in most cases.

Loading a Set Using Basler pylon

Loading a saved user set or the default set from the camera's non-volatile memory into the active set using Basler pylon is a two step process:

- Set the User Set Selector to User Set 1, User Set 2, User Set 3, or Default as desired.
- Execute a User Set Load command to load the selected set into the active set.

You can use the pylon API to set the User Set Selector and to execute the User Set Load command from within your application software. The following code snippet illustrates using the pylon API to set the selector and execute the command:

```
// Load user set 2 into the active set
Camera.UserSetSelector.SetValue( UserSetSelector_UserSet2 );
Camera.UserSetLoad.Execute( );

// Load the default set into the active set
Camera.UserSetSelector.SetValue( UserSetSelector_Default );
Camera.UserSetLoad.Execute( );
```

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Loading a Set Using Direct Register Access

To load a saved user set or the default set from the camera's non-volatile memory into the active set via direct register access:

- Set the value of the User Set Selector register to User Set 1, 2, or 3, or to the Default set as desired.
- Set the value of the User Set Load register to 1.

For more information about changing settings via direct register access, see Section 5 on page 61.

9.19.4 Selecting a "Startup" Set

You can select the default set or one of the user sets stored in the camera's non-volatile memory to be the "startup" set. The configuration set that you select as the startup set will be loaded into the active set whenever the camera starts up at power on or after a reset.

Selecting the Startup Set Using Basler pylon

With Basler pylon, the User Set Default Selector parameter is used to select User Set 1, User Set 2, User Set 3, or the Default Set as the startup set.

You can use the pylon API to set the User Set Default Selector parameter from within your application software. The following code snippet illustrates using the pylon API to set the selector:

```
// Designate user set 1 as the startup set
Camera.UserSetDefaultSelector.SetValue(
UserSetDefaultSelector_UserSet1 );

// Designate the default set as the startup set
Camera.UserSetDefaultSelector.SetValue(
UserSetDefaultSelector_Default );
```

For more information about the pylon API and the pylon Viewer, see Section 5 on page 61.

Selecting the Startup Set Using Direct Register Access

When using direct register access, the User Set Default Selector register is used to select the startup set:

- Set the value of the User Set Default Selector register for User Set 1, User Set 2, User Set 3, or Default as desired.

For more information about changing settings via direct register access, see Section 5 on page 61.

10 Technical Support

This chapter outlines the resources available to you if you need help working with your camera.

10.1 Technical Support Resources

If you need advice about your camera or if you need assistance troubleshooting a problem with your camera, you can contact the Basler technical support team for your area. Basler technical support contact information is located in the front pages of this manual.

You will also find helpful information such as frequently asked questions, downloads, and application notes in the Support section of the Basler website: www.baslerweb.com

If you do decide to contact Basler technical support, please take a look at the form that appears on the last two pages of this section before you call. Filling out this form will help make sure that you have all of the information the Basler technical support team needs to help you with your problem.

10.2 Obtaining an RMA Number

Whenever you want to return material to Basler, you must request a Return Material Authorization (RMA) number before sending it back. The RMA number **must** be stated in your delivery documents when you ship your material to us! Please be aware that if you return material without an RMA number, we reserve the right to reject the material.

You can find detailed information about how to obtain an RMA number in the Support section of the Basler website: www.baslerweb.com

10.3 Before Contacting Basler Technical Support

To help you as quickly and efficiently as possible when you have a problem with a Basler camera, it is important that you collect several pieces of information before you contact Basler technical support.

Copy the form that appears on the next two pages, fill it out, and fax the pages to your local dealer or to your nearest Basler support center. Or, you can send an e-mail listing the requested pieces of information and with the requested files attached. Basler technical support contact information is shown in the title section of this manual.

1. The camera's product ID: ____________________
2. The camera's serial number: ____________________
3. Your operating system: ____________________
4. Frame grabber that you use with the camera: ____________________
5. Describe the problem in as much detail as possible: (If you need more space, use an extra sheet of paper.)
6. If known, what's the cause of the problem?
7. When did/does the problem occur?
   - ☐ At startup.
   - ☐ While running.
   - ☐ After a certain action (e.g., a change of parameters):
8. How often did/does the problem occur?
   - ☐ Once.
   - ☐ Every time.
   - ☐ Regularly when:
   - ☐ Occasionally when:

9. How severe is the problem?

- ☐ Camera can still be used.
- ☐ Camera can be used after I take this action:

  ____

  ____

- ☐ Camera can no longer be used.

10. Did your application ever run without problems?

- ☐ Yes
- ☐ No

11. Parameter set

It is very important for Basler technical Support to get a copy of the exact camera parameters that you were using when the problem occurred.

To make note of the parameters, use the Basler pylon Viewer.

Select *Camera* menu > *Save Features...* command.
All feature settings are then saved as a text file in a pylon Feature Stream file (***.pfs**).

If you cannot access the camera, please try to state the following parameter settings:

- ☐ Pixel Format: ______________________________
- ☐ Image Size (AOI): ______________________________
- ☐ Exposure Time Control Mode ______________________________
- ☐ Exposure Time: ______________________________
- ☐ Gain: ______________________________
- ☐ Black Level: ______________________________
- ☐ Frame Rate: ______________________________

12. Live image/test image

If you are having an image problem, try to generate and save live images that show the problem. Also generate and save test images. Please save the images in BMP format, zip them, and send them to Basler technical support.

Revision History

Doc. ID Number	Date	Changes
AW00083001000	14 May 2009	Initial release of this document. Applies to prototype cameras.
AW00083002000	13 Jul 2009	Initial release of this document for series production cameras.
AW00083003000	7 Sep 2009	Added appropriate information for the avA1600-65km/kc and avA1900-60km/kc prototype cameras.
AW00083004000	15 Dec 2009	Added information throughout the document regarding the selectable pixel clock feature, the 2 tap 10 bit output mode, and the 1X2-1Y and 1X-2YE Camera Link tap geometries. Added information to Section 7.11.1 on page 145 regarding the minimum allowed setting for the AOI height. Added Section 7.18 on page 168 describing the imaging sensor temperature sensor and the imaging sensor over temperature condition. Added information to Section 7.19 on page 170 regarding 10 bit test images and the color test image.
AW00083005000	21 Apr 2010	Added appropriate information throughout the document regarding the new avA2300-30km/kc. Added appropriate information throughout the document regarding the new acquisition trigger functionality. Corrected the information about the BBPL in Section 4.2 on page 46. Corrected the gain values stated in Section 7.3.1 on page 107.
AW00083006000	28 Sep 2010	Corrected various typos in Section 5 on page 49.
AW00083007000	17 Dec 2010	Revised the document title.

Doc. ID Number	Date	Changes
AW00083**08**000	Oct 2011	Indicated Basler AG as bearer of the copyright on the back of the front page. Indicated pixel data formats in Section 1.2 on page 2. Added the 1X-1Y tap geometry ■ in Section 1.2 on page 2 ■ on page 110 and ■ in Section 8.2 on page 143. Added code snippets for Mono 10 and Bayer GR 10 in Section 8.2 on page 143. Noted that the Bayer filter alignment shown in Section 8.3 on page 133 serves as an example only. Changed the max. prelines setting for the avA1000-120km/kc in Section 9.7.2 on page 171.
	22 Oct 2013	New cover photo. Inserted new Basler logo on the front page. Renaming throughout the manual: ■ new name: Basler pylon Camera Software Suite ■ changed pylon Viewer Tool to pylon Viewer Replaced Kodak by Truesense Imaging in Section 1.2 on page 2. Updated the sensor resolution of the avA1600-65km and avA1600-65kc in Section 1.2 on page 2. Updated the mail addresses in the Contact section. Entered the new IR cut filter characteristics data in Section 1.4.2 on page 8. Added LZ4 licensing information in Section 1.6.1 on page 13. Added the 20 MHz pixel clock information in ■ Section 1.2 on page 2 ■ Section 2.11 on page 42 ■ Section 9.2 on page 149 Removed sentences "A power supply and cable assembly that meets these requirements is available from Basler. Contact your Basler sales representative for more information." in Section 2.6.1 on page 33. Added Section 2.4.4 on page 36 ("Inter-Line Delay"). Simplified Figure 11 on page 37, Figure 12 on page 38 and Figure 13 on page 39. Removed Section '2.7 I/O Line Schematic'. Moved the "Setting the Timer 1 Parameters" section to Section 3.2.4.1 on page 46. Added new section ("I/O Control" Section 3 on page 37). Adapted Camera Link tap geometry sections: ■ Section 4 on page 54 ■ Section 8.2 on page 143

Doc. ID Number	Date	Changes
AW00083**08**000	22 Oct 2013	Removed abs. max. voltages (30.0 VDC) from sub-sections of Section 2. Added paragraph on exposure time settings in "Trigger Width Exposure Mode" section on page 88. Added Section 6.3.3.3 on page 89 (Frame Start Trigger Delay). Added "Acquisition Monitoring Tools", section 6.6 on page 97 (including "Acquistion Status Indicator" and "Trigger Wait Signals" sections). Moved "The Bayer Color Filter" Section from "Pixel Formats, Tap Geometries, and Color Filters" to the new "Color Creation" sub-section in "Color Creation and Enhancement". Updated the balance ratio value range in Section 7.3.1 on page 126. Added Section 6.5 on page 94 ("Overlapping Exposure with Sensor Readout"). Added the "White Balance Reset" sub-section on page 127 (White Balance section). Modified the Section 7.3.2 on page 128 (added sRGB and User gamma in the "Gamma Correction" section). Added new "Color Adjustment" Section 7.3.4 on page 134. Modified note on availability of the auto functions when the cameras are powered on or reset: ▪ Section 6.4 on page 92 ▪ Section 9.3 on page 151 and ▪ in all sub-sections of the "Auto Functions" section from page 173 on. Integrated that the balance white auto function can now also be operated in the "continuous" mode of operation (Section 9.8.9 on page 190). Renamed "Disable Parameter Limits" feature: Correct name "Remove Parameter Limits". Modified note on remove parameter limits on Section 9.5 on page 158: The feature can only be removed on the gain and the prelines features. Exchanged Exposure Time register example by Prelines register example. Added, for cameras delivered from the factory, that the exposure auto function is set to "continuous" mode of operation in Section 9.8.6 on page 185. Added the debouncer feature, see Section 3.1.2 on page 38. Removed section "Integrated IR Cut Filter (on Color Models)" (it is now in the "Color Creation and Enhancement" section). Added the gray value adjustment damping feature in Section 9.8.7 on page 188. Added sentence concerning gray value adjustment damping in Section "Using an Auto Function" on page 181. Added the minimum output pulse width feature, see Section 9.9 on page 192. Added Frame Start Trigger Delay describing the error detection. Adapted Section 9.7 on page 167: free positioning of the AOI for 1X-1Y and 1X2-1Y tap geometries.

Doc. ID Number	Date	Changes
AW00083**08**000	22 Oct 2013	Added note that the auto functions feature will not work, if the sequencer feature is enabled (Section 9.8 on page 173). Added the sequencer feature in Section 9.11 on page 197. Added the "Error Detection" section in 9.10 on page 194. Deleted sentence "Some auto functions always share an Auto Function AOI and some auto functions can use their own individual Auto Function AOIs." Section 9.8.3 on page 175. Added the Save Feature... command on page 265 in Section 10.
AW00083**09**000	05 Mar 2018	Removed cross-references to the *Installation and Setup Guide for Camera Link Cameras* (AW000996) throughout the manual. Updated the layout of the information and notice boxes throughout the manual. Replaced "pixel data format" by "pixel format" throughout the manual. Changed "Subpart J" to "Subpart B" on the reverse of the front page. Renamed "Warranty Precautions" to "**Cleaning of the sensor**" section and revised its content on page 18. Changed "Truesense Imaging" to "ON Semiconductor" in Section 1.2 on page 2. Removed "Max" from "Max Power Consumption" in the specifications table in Section 1.2 on page 2. Indicated RoHS conformity as included in CE conformity and added REACH conformity in Section 1.2 on page 2. Updated information about the Basler pylon Camera Software Suite in Section 1.2 on page 2. Added Section 1.3 on page 6 "Accessories". Removed the cable requirement of "correct length" in Section 1.7 on page 14. Added Section 2.2.2.1 on page 23 "Grounding Recommendations". Added precautions not to apply compressed air and about cleaning the sensor in Section 1.9 on page 16. Added precautions relating to SELV and LPS requirements in Section 1.9 on page 16. Renamed and updated the "Observe the following items:" section on page 19. Re-arranged section in Chapter 2 on page 20 with some rephrasing and transferred the content of the "LED Indicator" section to Section 9.10.1 on page 194. Added an introductory note about power supply in Chapter 2 "Physical Interface". Removed the "Camera Connector Types" section and transferred information about receptacle part numbers to Section 2.1 on page 20.

Doc. ID Number	Date	Changes
		cont'd Removed information about plugs and wiring from section "Physical Interface" on page 20. Updated Section 3.1.2 on page 38 ("Input Line Debouncer"). Updated Chapter 5 on page 61. Updated Section 5.1.2 on page 62. Added a note about the Instant Camera classes in Chapter 6 on page 64. Removed the ExposureOverlapTimeMaxAbs parameter and information about setting the parameter in Chapter 6 on page 64 and suggested use of the ExposureTimeAbs and ExposureTimeRaw parameters instead. Added a hint in Section 6.6.3.2 on page 102 for using direct register access for selecting frame trigger wait as a source signal for an output line Replaced "micro lens" by "filter" in Section 7.1 on page 123. Updated Section 7.1 on page 123 (added explanation to LightSourceSelector Off and how the coefficients are set). Removed reference to ADCs in Section 9.16 on page 247.

Index