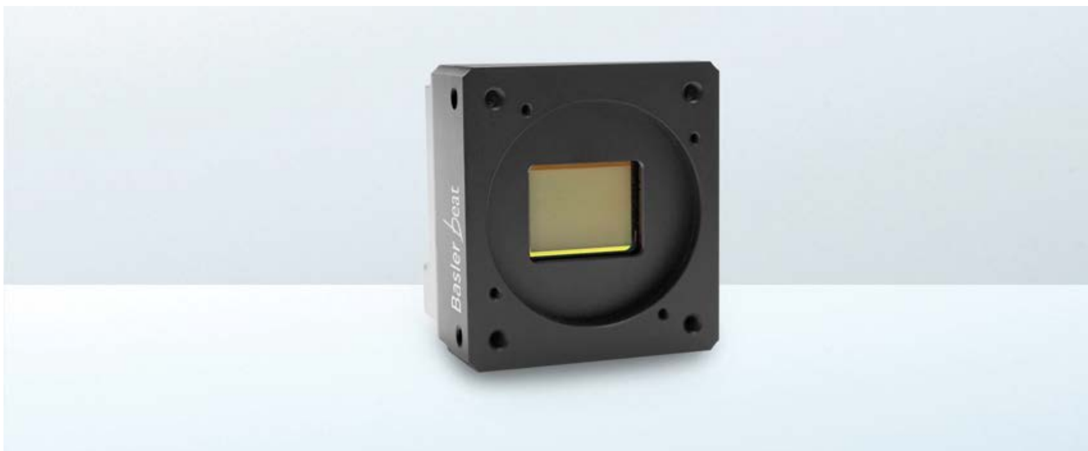
Basler beat



USER’S MANUAL FOR CAMERA LINK CAMERAS

Document Number: AW001308
Version: 04 Language: 000 (English)
Release Date: 15 June 2015

BASLER the power of sight

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.

Warranty Note

Do not open the housing of the camera. The warranty becomes void if the housing is opened.

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

Currently, two Basler beat Camera Link® camera models are available. The model names are indicated in the top row of the specification table (see below).

1.2 General Specifications

Specification	beA4000-62km	beA4000-62kc
Sensor Size (H x V pixels))	4096 x 3072	4088 x 3070
Sensor Type	CMOSIS CMV12000 v2 Progressive scan CMOS Global shutter	
Optical Size	APS-C	
Effective Sensor Diagonal	28.2 mm	
Pixel Size	5.5 µm x 5.5 µm	
Max Frame Rate	62 fps	
Mono/Color	Mono	Color
Sensor Bit Depth	12 bit; 10 bit @ 84.0 MHz & 1X10-1Y	
Data Output Type	Camera Link base, medium or full configuration	
Connectors	One 6-pin Hirose micro-miniature receptacle Two Camera Link 26-pin SDR connectors (Mini Camera Link connectors)	
Camera Link Pixel Clock Speed	32.5 MHz, 65 MHz, 84.0 MHz (selectable)	
Camera Link Tap Geometry	1X2-1Y, 1X3-1Y, 1X8-1Y, 1X10-1Y	
Pixel Formats	Mono 8 Mono 10* Mono 12*	Bayer GB 8 Bayer GB 10* Bayer GB 12*
Synchronization	Via external trigger signal, via software trigger signal or free run	
Exposure Control	Programmable via the camera API	
Camera Power Requirements	+12 VDC (-10 %) to +24 VDC (+5 %), < 1 ripple, supplied via the camera's 6-pin connector. Power supply must be able to supply at least 8 W.	
Max. Power Consumption	≈ 6 W @ 12 VDC	
I/O Lines	4 input lines (CC1 to CC4) and 1 output line (via Camera Link spare bit)	
Lens Adapter	Universal camera front, suitable for lens mount adapters with the following lens mounts: F-mount, M42 (×1.0 or ×0.75)-mount, M42 FBD 45.5 (×1.0)-mount, and M58 (×0.75). Lens adapters are not in the camera's scope of delivery and must be ordered separately as accessories. See Section 1.4.4 on page 11 for information about selecting a suitable lens adapter for your camera.	

Table 1: General Specifications - 12 MP Cameras

Specification	beA4000-62km	beA4000-62kc
Size (L x W x H)	40.21 mm x 56 mm x 62 mm (without lens adapter and connectors) 84.01 mm x 56 mm x 62 mm (with F-mount lens adapter and connectors) 53.51mm x 56 mm x 62 mm (with M42-mount lens adapter and connectors) 83.07 mm x 56 mm x 62 mm (with M42 FBD 45.5-mount lens adapter and connectors) 57.51 mm x 56 mm x 62 mm (with M58-mount lens adapter and connectors)	
Weight	≈ 210 g (typical) without lens adapter ≈ 300 g (typical) with F-mount lens adapter and connectors ≈ 230 g (typical) with M42-mount lens adapter and connectors ≈ 280 g (typical) with M42 FBD 45.5-mount lens adapter and connectors ≈ 240 g (typical) with M58-mount lens adapter and connectors	
Conformity	CE (in preparation), RoHS, FCC, UL, GenICam, Camera Link, IP30	

Table 1: General Specifications - 12 MP Cameras

*Available subject to camera parameter settings.

1.3 Spectral Response

The following graphs show the quantum efficiency curves for monochrome and color cameras.

ⓘ	The quantum efficiency curves exclude lens characteristics and light source characteristics.

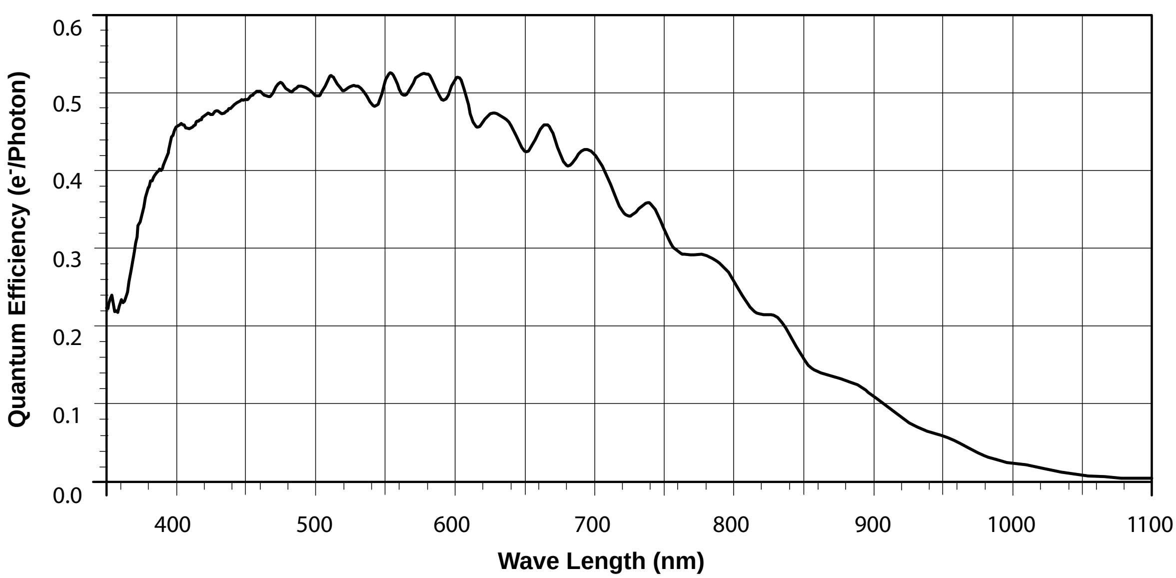


Fig. 1: Quantum Efficiency of the Monochrome Sensor in 12-Bit Depth Mode (Based on Sensor Vendor Information)

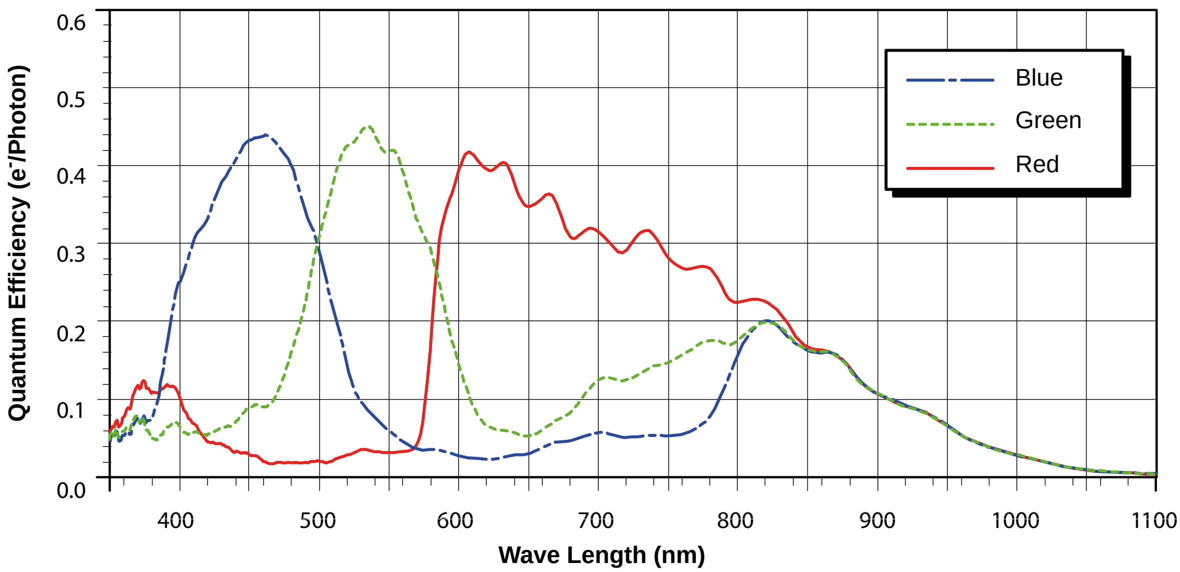


Fig. 2: Quantum Efficiency of the Color Sensor in 12-Bit Depth Mode (Based on Sensor Vendor Information)

1.4 Mechanical Specifications

1.4.1 Camera Dimensions and Mounting Points

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

The camera housings conform to the IP30 protection class provided the camera front is covered by the protective plastic seal that is shipped with the camera.

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

Camera housings are equipped with four mounting holes (4 x M4; 6.3 deep) on the front and two mounting holes (8 x M4; 6.3 deep) on each side as shown in the drawings. Four additional holes (4 x M2.5; 3.3 deep) are present on the camera front for mounting the lens adapter.

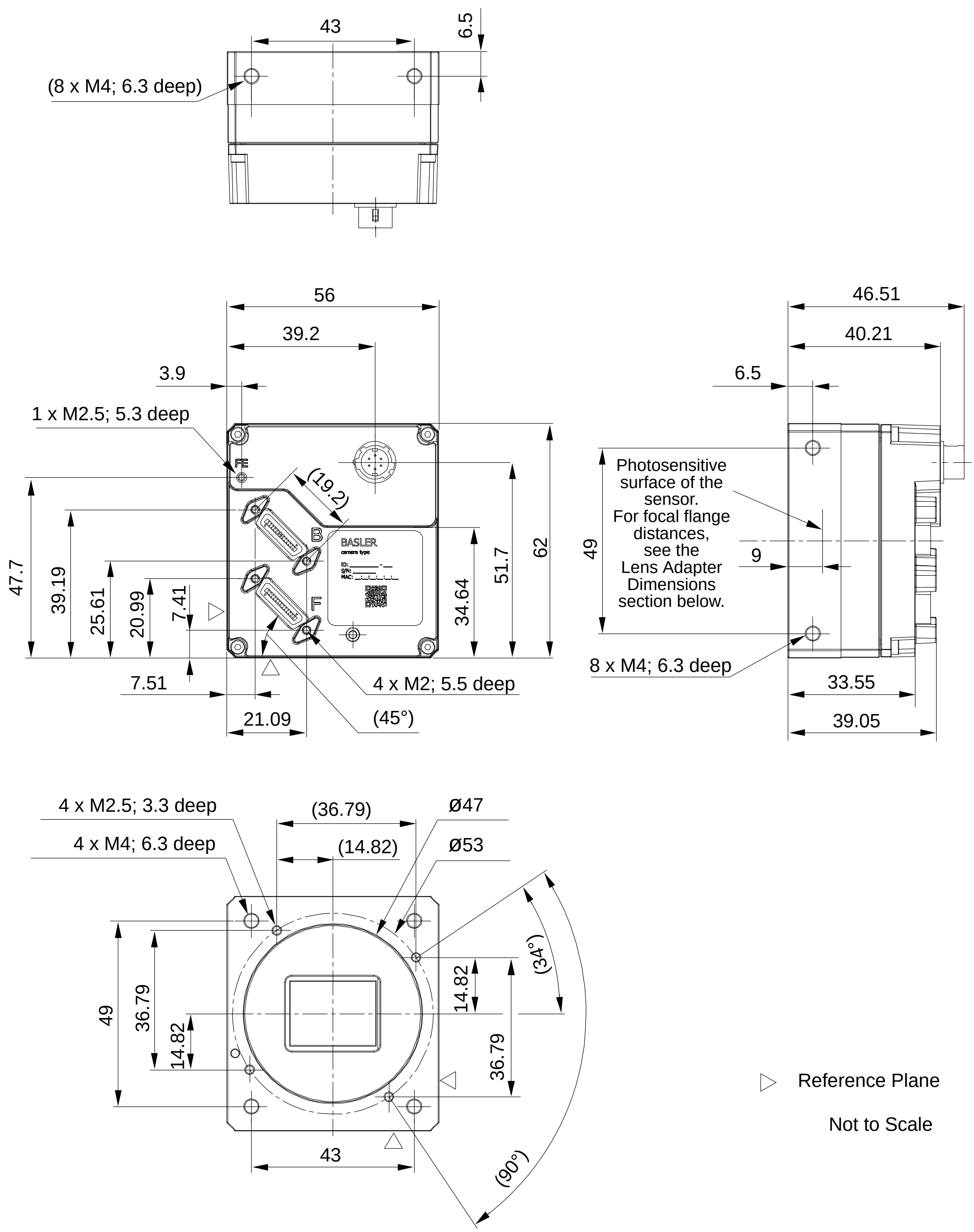


Fig. 3: Mechanical Dimensions (in mm)

1.4.2 Sensor Pixel Numbering Origin

The location of pixel 0, 0 on the camera's sensor is shown in Fig. 4. Pixel 0, 0 serves as the origin for pixel numbering.

A marker hole in the camera's front helps in identifying the location of pixel 0, 0.

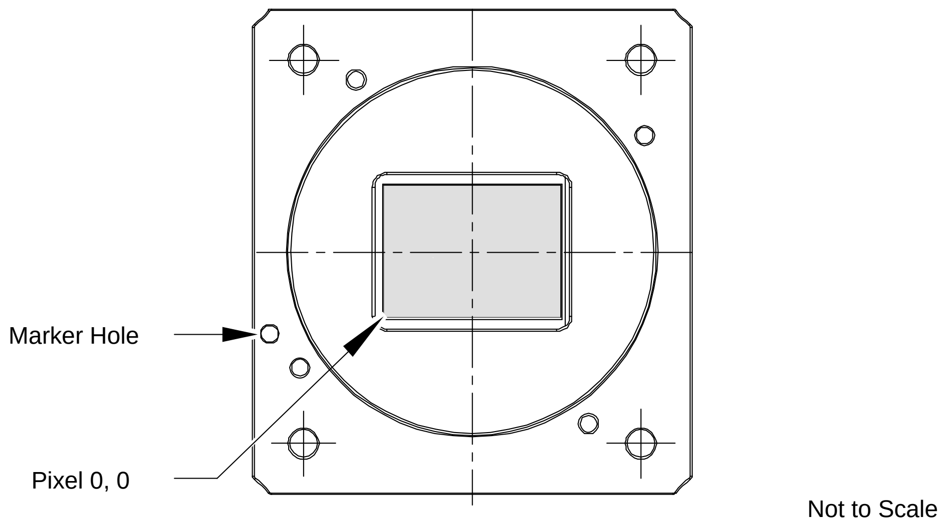


Fig. 4: Location of Pixel 0, 0 on the Sensor

1.4.3 Lens Adapter Dimensions

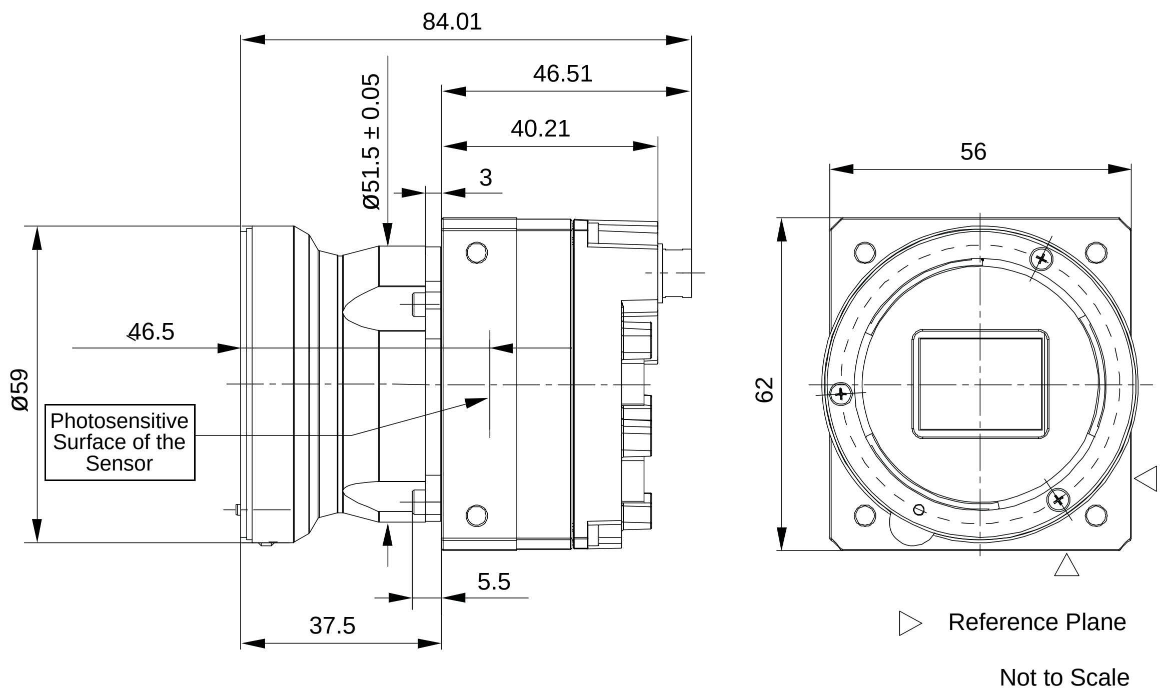


Fig. 5: F-Mount Adapter on a Basler beat Camera Link Camera; Dimensions in mm

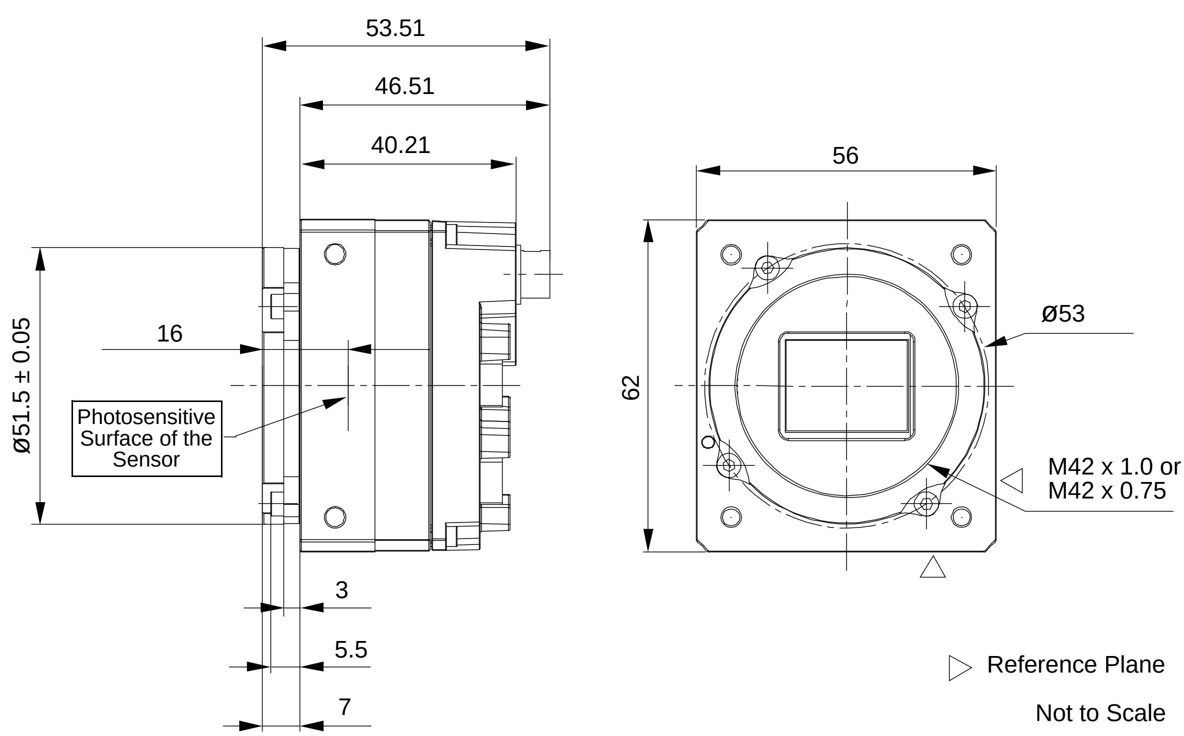


Fig. 6: M42 x 1.0 or M42 x 0.75 Mount Adapter on a Basler beat Camera Link Camera; Dimensions in mm

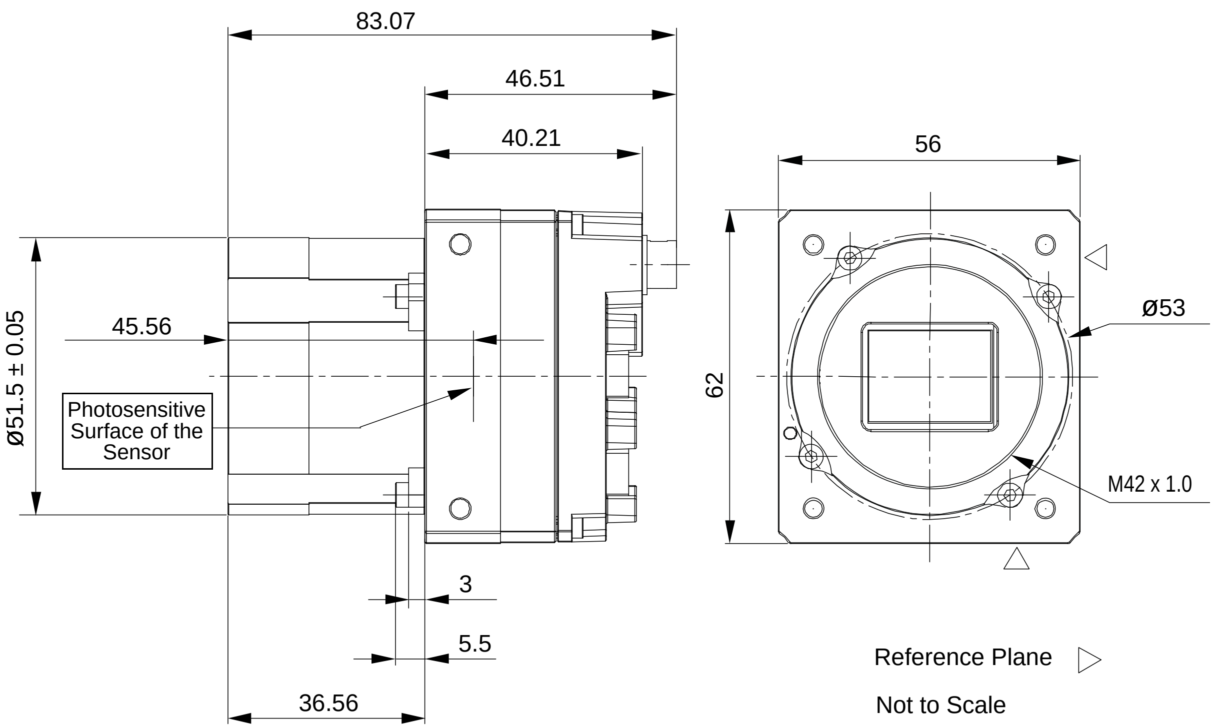


Fig. 7: M42 x 1.0 FBD 45.5 Mount Adapter on a Basler beat Camera Link Camera; Dimensions in mm

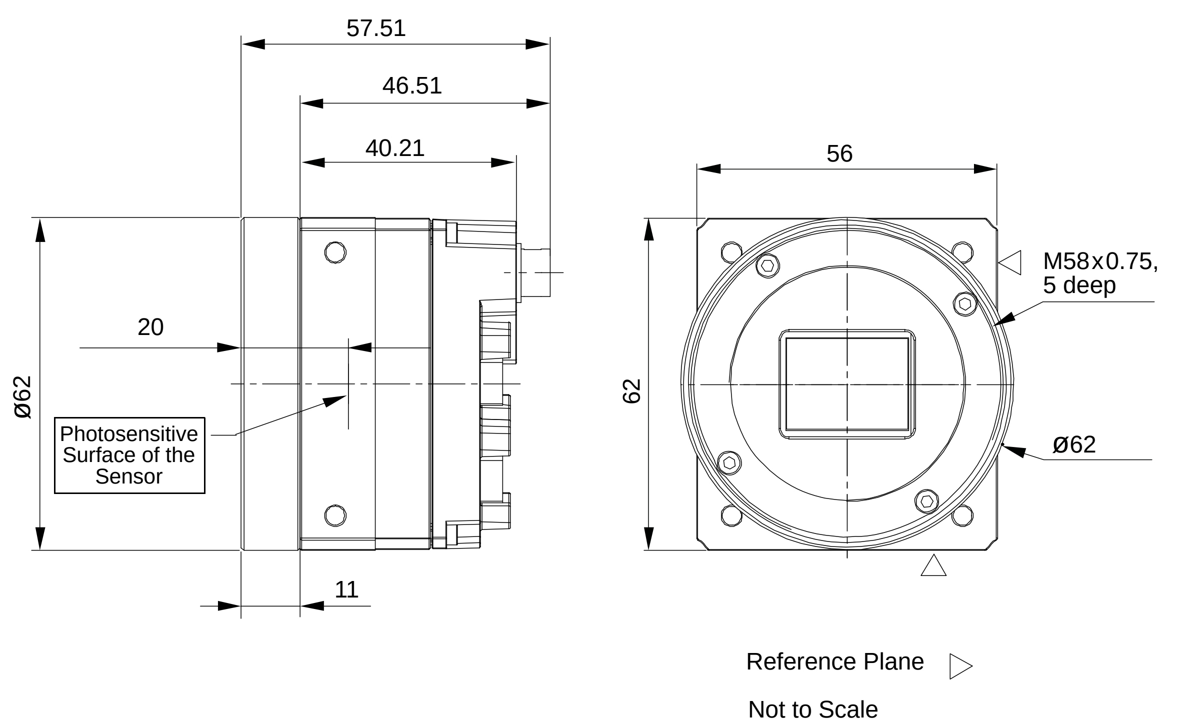


Fig. 8: M58 x 0.75 Mount Adapter on a Basler beat Camera Link Camera; Dimensions in mm

1.4.4 Attaching a Lens Adapter

Use the four M2.5 setscrews supplied with the lens adapter to lock the lens adapter to the camera. See Fig. 3 for information where to place the M2.5 setscrews.

NOTICE
Screwing with excessive torque can damage the camera, lens adapter or setscrews. When screwing in the supplied M2.5 setscrews, make sure to never exceed a torque of 0.4 Nm.

1.5 Software Licensing Information

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

LZ4 - Fast LZ compression algorithm

1.6 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.
- Try to use camera cables that are as short as possible and try to run the camera cables and power cables parallel to each other. Avoid coiling camera cables. If the cables are too long, use a meandering path rather then coiling the cables.
- 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 can 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.
- 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.

A functional earth connection on the back of the camera (labelled "FE", see Fig. 3 on page 7) allows to establish an electrically conducting connection to the camera housing if required by your application.

> ⓘ 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 camera's Downloads section of our website: www.baslerweb.com

1.7 Environmental Requirements

1.7.1 Temperature and Humidity

Allowed housing temperature range during operation:	0 °C ... +60 °C (+32 °F ... +140 °F)

Housing temperature according to UL 60950-1: max. 70 °C (+158 °F)
Ambient temperature according to UL 60950-1: max. 40 °C (+104 °F)
UL 60950-1 test conditions: no lens and no lens adapter attached to the camera and without efficient heat dissipation; ambient temperature kept at 40 °C (+104 °F).

Allowed humidity range during operation:	20 % ... 80 %, relative, non-condensing
Allowed storage temperature range:	-20 °C ... +80 °C (-4 °F ... +176 °F)
Allowed storage humidity range:	20 %... 80 %, relative, non-condensing

1.7.2 Heat Dissipation

You must provide sufficient heat dissipation to maintain the temperature of the camera housing at he maximum value or below as specified for the camera during operation (see above). Given the low power consumption of the camera its housing temperature during operation will generally stay within the specified limits. Overheating can only occur if heat dissipation is exceptionally restricted.

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 he maximum value as specified for the camera during operation (see above). Keep in mind that the camera will gradually become warmer after it was started. After some time, the housing temperature will have stabilized and will no longer increase.
- Provide sufficient heat dissipation by e.g. mounting the camera on a substantial, thermally conductive component that can act as a heat sink and by using a fan to provide an air flow over the camera.

For optimum image quality, we recommend not to operate the camera at elevated temperatures.

1.7.3 Imaging Sensor Over Temperature Condition

The camera has imaging sensor over temperature protection. If the temperature of the camera's imaging sensor rises above 75° C, an over temperature condition will be detected and the circuitry for the imaging sensor will switch off. In this situation, you will still be able to communicate with the camera but the camera will no longer acquire images.

Provide the necessary cooling when this situation arises. After the imaging sensor circuitry has sufficiently cooled bring the camera back to normal operation by either action:

- Carry out a camera restart by switching power off and on again or
- Carry out a camera reset (see Section 10.1 on page 161).

1.8 Precautions

⚠ DANGER

Electric Shock Hazard

Non-approved power supplies may cause electric shock. Serious injury or death may occur.

You must use a camera power supply which meets the Safety Extra Low Voltage (SELV) and Limited Power Source (LPS) requirements.

⚠ WARNING

Fire Hazard

Non-approved power supplies may cause fire and burns.

You must use a camera power supply which meets the Limited Power Source (LPS) requirements.

NOTICE

Avoid dust on the sensor.

- The camera is shipped with a protective plastic seal on the camera front. To avoid collecting dust on the camera's sensor, make sure that you always put the protective seal in place when there is no lens mounted on the camera.
- Every time you remove the protective seal, a lens or a lens adapter, make sure that the camera is pointing down.
- Never apply compressed air to the camera. This can easily contaminate optical components, particularly the sensor.

NOTICE

Applying incorrect power can severely damage the camera.

- You must supply camera power with the correct voltage: The camera‘s nominal operating voltage is +12 VDC (± 10 %) with an allowed range of +12 VDC (- 10 %) to +24 VDC (+ 5 %), effective on the camera‘s connector;< 1 % ripple.
- You must supply camera power with the correct polarity.
- You must avoid a voltage drop: If you supply camera power via a long cable a voltage drop can occur. We recommend that you provide camera power separately through the wires connecting to pins 1 and 2 of the receptacle. We also recommend that you provide ground separately to the wires connecting to pins 5 and 6.
- Make sure to use a power supply that can supply at least 8 W.

NOTICE

Making or breaking Camera Link connections incorrectly can severely damage the camera.

- Switch off the power to the 6-pin connector before you connect or disconnect the Camera Link cables.
- If you can’t switch off the power, be sure that:
  - The plug for the 6-pin connector is the last plug that you plug into the camera when making connections.
  - The plug for the 6-pin connector is the first plug that you unplug from the camera when breaking connections.

NOTICE

Incorrect plugs can damage the camera’s connectors.

The plug on the cable that you attach to the camera’s 6-pin connector must have 6 female pins.

NOTICE

Inappropriate code can cause unexpected camera behavior.

- 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.
- 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.
- The code snippets in this manual are written in C++. Other programming languages can also be used to write code for use with Basler pylon. When writing code, you should use a programming language that is both compatible with pylon and appropriate for your application. For more information about the programming languages that can be used with Basler pylon, see the documentation included with the pylon software.

Warranty Precautions

To ensure that your warranty remains in force:

Do not remove the camera’s serial number label

If the label is removed and the serial number can’t be read from the camera’s registers, the warranty is void.

Do not open the camera housing

Do not open the housing. Touching internal components may damage them.

Prevent ingress or insertion of foreign substances into the camera housing

Prevent liquid, flammable, or metallic substances from entering the camera housing. If operated with any foreign substances inside, the camera may fail or cause a fire.

Avoid electromagnetic fields

Do not operate the camera in the vicinity of strong electromagnetic fields. Avoid electrostatic charging.

Transport in original packaging

Transport the camera in its original packaging only. Do not discard the packaging.

Clean with care

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

- Before starting the cleaning procedure, disconnect the camera from camera power and I/O power.To clean the surface of the camera housing,
  - do not use solvents or thinners; they can damage the surface
  - use a soft, dry cloth that will not generate static during cleaning (cotton is a good choice). To remove severe stains, use a soft cloth dampened with a small quantity of window cleaner or neutral detergent, then wipe dry.
  - Make sure all window cleaner or detergent has evaporated after the cleaning procedure, before reconnecting the camera to power.

Read the manual

Read the manual carefully before using the camera.

2 Software and Hardware Installation

The information you will need to install and operate the camera is included in the *Installation and Setup Guide for Cameras Used With the Basler pylon Camera Software Suite* (AW000611).

You can download the *Installation and Setup Guide for Cameras Used With the Basler pylon Camera Software Suite* from the **Software Downloads** section of the Basler website: www.baslerweb.com

⚠ DANGER

Electric Shock Hazard

Non-approved power supplies may cause electric shock. Serious injury or death may occur.

You must use a camera power supply which meets the Safety Extra Low Voltage (SELV) and Limited Power Source (LPS) requirements.

⚠ WARNING

Fire Hazard

Non-approved power supplies may cause fire and burns.

You must use a camera power supply which meets the Limited Power Source (LPS) requirements.

2.1 Required Accessories

Note that the camera's scope of delivery does not include accessories needed for camera operation. Most of the accessories can be obtained from Basler.

For more information about suitable accessories and how to obtain them, see the Basler website www.baslerweb.com

The following accessories will generally be needed:

- Lens
- Lens adapter (see also Section 1.4.3 on page 9)
- Power supply (see also Section 5.4 on page 38)
- Power cable with suitable connectors (see also Section 5.3.1 on page 36)
- Camera Link cable with suitable connectors (see also Section 5.3.2 on page 37)
- IR cut filter (for color cameras only). To obtain best color reproduction from color cameras, we strongly recommend that you use an IR cut filter. The filter should transmit in a range from 400 nm to 650 nm, and it should cut off from 650 nm to at least 1100 nm. A suitable filter is the B+W 486.

3 Tools for Changing Camera Parameters

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

- The options available with the Basler pylon Camera Software Suite let you change parameters and control the camera by using a stand-alone GUI (known as the pylon Viewer) or by accessing the camera from within your software application using the pylon SDK.
- You can also control the camera and change parameters via direct access to the camera's register structure.

3.1 Basler pylon Camera Software Suite

The Basler pylon 4 Camera Software Suite is designed for use with all Basler cameras with the following interface types: IEEE 1394a, IEEE 1394b, GigE, USB 3.0. It can also be used with newer Basler Camera Link cameras. The pylon Camera Software Suite offers reliable, real-time image data transport into the memory of your PC at a very low CPU load.

You can download the Basler Camera Software Suite from the Basler website www.baslerweb.com

For information about the features included in the pylon Camera Software Suite and about installing pylon software, see the *Installation and Setup Guide for Cameras Used With the Basler pylon Camera Software Suite* (AW000611). You can download the document from the Basler website: www.baslerweb.com

The pylon Camera Software Suite is available for Windows and Linux operating systems and includes several tools, in particular the pylon Viewer and the pylon SDK, that you can use to change the parameters on your camera. The remaining sections in this chapter provide an introduction to these tools.

3.1.1 pylon Viewer

The pylon Viewer is included in Basler’s pylon camera software suite. The pylon Viewer is a standalone application that lets you change most of the camera’s parameter settings via a GUI-based interface. 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.

Note that with Camera Link cameras, such as the Basler beat, the viewer can’t display the images captured by the camera. Most frame grabbers include software that will display the captured images, so you can use the pylon Viewer to change the camera’s parameters and the frame grabber software to view the captured images.

3.1.2 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)
- pylon SDK for .NET (Windows).

Each SDK includes an 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 or .NET), a *Programmer’s Guide and Reference Documentation* is available. The documentation gives an introduction to the related pylon API and provides information about all methods and objects of the API.

The sample code included in this manual 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.

3.2 Basler Binary Protocol Library

Basler beat Camera Link 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 and control the camera 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 *Register Structure and Access Methods for Camera Link Cameras* (AW000997) that is specific for ace, aviator, and Basler beat Camera Link cameras. The document can be downloaded from the Basler website.

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

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.

Each camera provides feature such as a full frame shutter and electronic exposure time control. Exposure start, exposure time, and image data transmission can be controlled by parameter values transmitted to the camera via the Camera Link interface.

The camera employs a CMOS sensor chip (see Fig. 9 on page 28) designed for monochrome imaging. For color cameras, the sensor is covered by an additive color separation filter (Chapter 7 on page 107). Accumulated charges are read out simultaneously for each two neighboring rows of the sensor’s light-sensitive elements (pixels). After analog processing, 12-bit analog-to-digital conversion, and digital processing, the pixel data leave the sensor via sensor taps. Their number depends on the camera operation mode.

The pixel data are passed into an FPGA (Fig. 10 on page 29), are re-sorted and modified according to the set features, and then leave the FPGA. From the time the pixel data leave the FPGA until they enter the PC, the pixel data are clocked according to the set Camera Link clock speed.

The pixel data are transmitted to the Camera Link interface and then to the frame grabber in your PC. Several different "Camera Link tap geometries" are available to determine how the pixel data will be transmitted over the Camera Link interface. The user can select the desired Camera Link tap geometry.

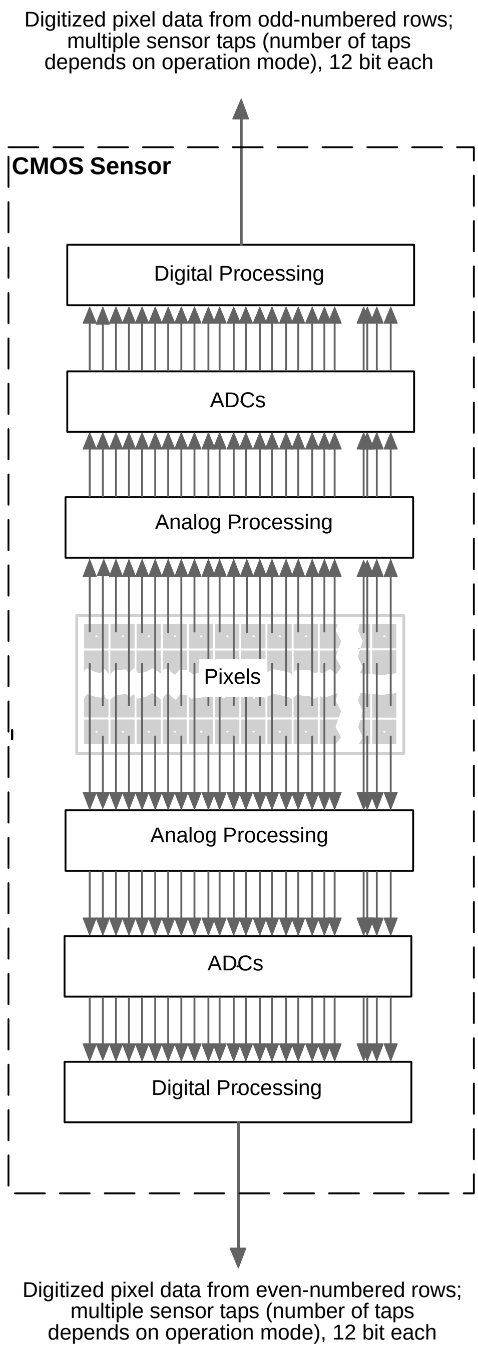


Fig. 9: CMOS Sensor Architecture

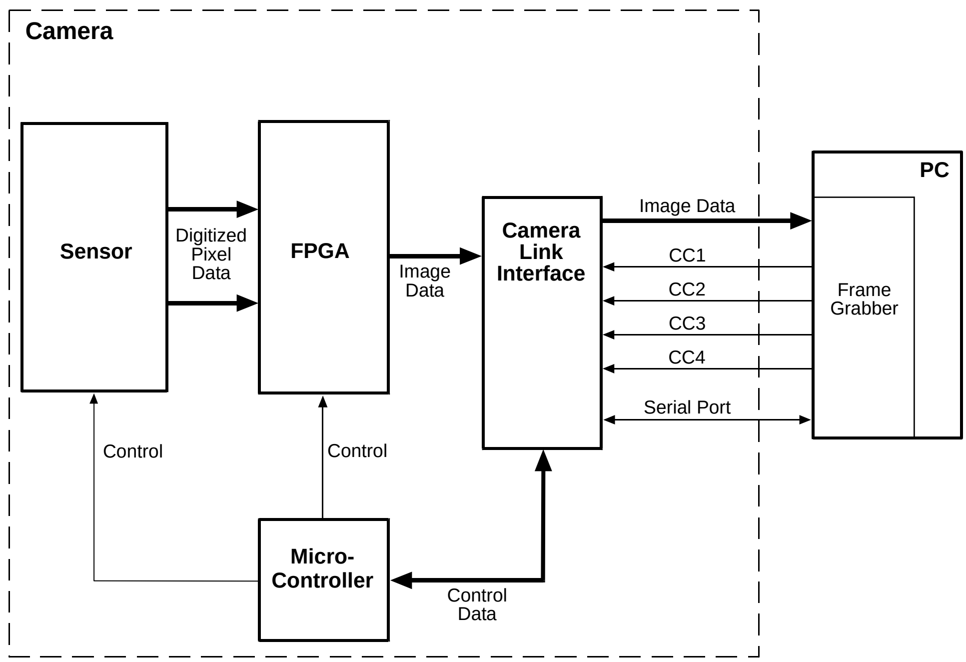


Fig. 10: Camera Block Diagram

5 Physical Interface

This chapter provides detailed information about the physical interface on the camera, such as pinouts and voltage requirements. The information will be especially useful during your initial design-in process.

5.1 General Description of the Connections

The camera is interfaced to external circuity via connectors located on the back of the housing:

- Two 26-pin, 0.03” pin spacing, Shrunk Delta Ribbon (SDR) female connectors used to transmit video data, control signals, and configuration commands. (This type of connector is also known as a Mini Camera Link connector).
- One 6-pin receptacle used to provide power to the camera.

There are also a connection point for functional earth (FE) and a LED indicator located on the back of the camera as shown in the drawing.

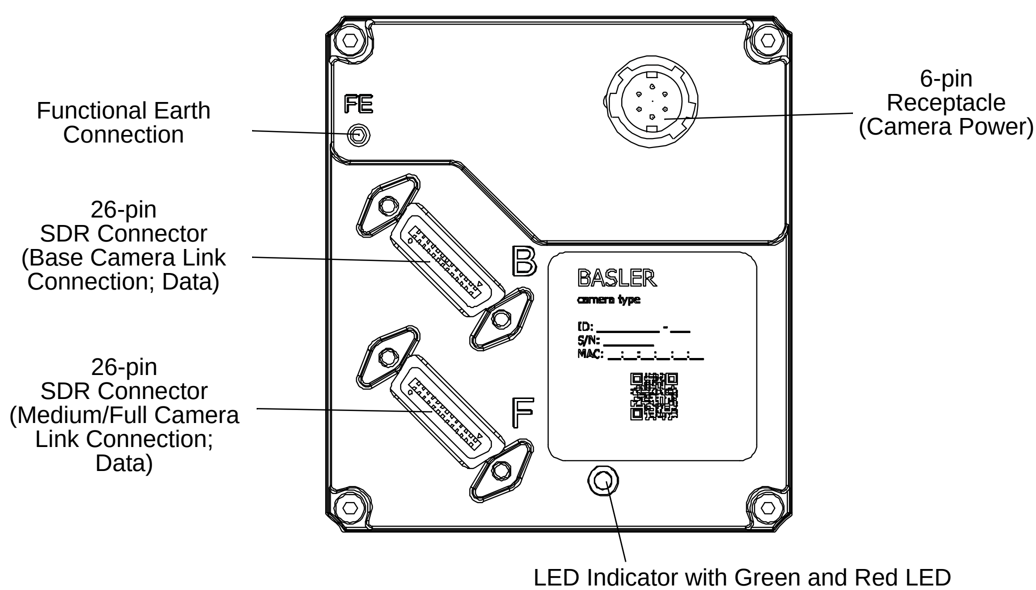


Fig. 11: Camera Connectors and LED Indicator

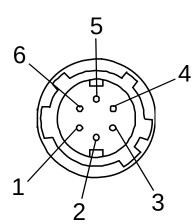
5.2 Camera Connectors, Pin Assignments, and Numbering

5.2.1 6-pin Receptacle

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

The 6-pin receptacle is a Hirose micro connector (part number HR10A-7R-6PB) or the equivalent.

The recommended mating plug on the cable is the female Hirose micro plug (part number HR10A-7P-6S) or the equivalent.



Pin	Designation
1	+12 VDC (- 10 %) to +24 VDC (+ 5 %), < 1 ripple, Camera Power *
2	+12 VDC (- 10 %) to +24 VDC (+ 5 %), < 1 ripple, Camera Power *
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.

See Section 5.4.1 on page 38 for details about supplying camera power.

5.2.2 26-pin SDR Connectors

Two 26-pin, 0.03” pin spacing, Shrunk Delta Ribbon (SDR) female connectors are as called for in the Camera Link specification.

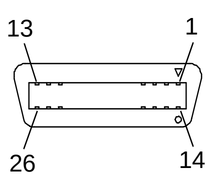
The recommended mating connector is also defined in the Camera Link Specification.

The connectors are used to transmit video data, control signals, and configuration commands. The connectors are called the base Camera Link SDR connector and the medium/full Camera Link SDR connector (see Fig. 11).

The pin assignments and pin numbering are shown in Table 3 on page 34 for the base Camera Link SDR connector and in Table 4 on page 35 for the medium/full Camera Link SDR connector.

ⓘ	The "SDR" (Shrunk Delta Ribbon) designation is the naming used by the 3M company. Other companies use different names for this type of connector. For example, Honda uses "HDR" as the connector name.

Pin Number	Signal Name	Direction	Level	Function
1, 26 *	-	In	-	Not used
13, 14*	-	-	-	Ground
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	Configurable
22	CC1+			
10	CC2+	Input	RS-644 LVDS	Configurable
23	CC2-			
11	CC3-	Input	RS-644 LVDS	Configurable
24	CC3+			
12	CC4+	Input	RS-644 LVDS	Configurable
25	CC4-			

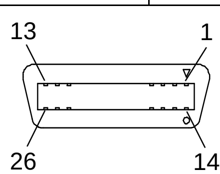


* Pins 1 and 26 are tied together in the camera.

** Pins 13 and 14 are tied together in the camera.

Table 3: Pin Assignments and Numbering for the Base Configuration 26-pin SDR Connector

Pin Number	Signal Name	Direction	Level	Function
1, 26 *	-	In	-	Not used
13, 14*	-	-	-	Ground
2	Y0-	Output	Camera Link LVDS	Data from transmitter circuit Y
15	Y0+			
3	Y1-	Output	Camera Link LVDS	Data from transmitter circuit Y
16	Y1+			
4	Y2-	Output	Camera Link LVDS	Data from transmitter circuit Y
17	Y2+			
6	Y3-	Output	Camera Link LVDS	Data from transmitter circuit Y
19	Y3+			
5	YClk-	Output	Camera Link LVDS	Pixel clock from transmitter circuit Y
18	YClk+			
7	T+			Connected to T- with 100R; not used
20	T-			Connected to T+ with 100R; not used
8	Z0-	Output	Camera Link LVDS	Data from transmitter circuit Z
21	Z0+			
9	Z1-	Output	Camera Link LVDS	Data from transmitter circuit Z
22	Z1+			
10	Z2-	Output	Camera Link LVDS	Data from transmitter circuit Z
23	Z2+			
12	Z3-	Output	Camera Link LVDS	Data from transmitter circuit Z
25	Z3+			
11	ZClk-	Output	Camera Link LVDS	Pixel clock from transmitter circuit Z
24	ZClk+			



* Pins 1 and 26 are tied together in the camera.
** Pins 13 and 14 are tied together in the camera.

Table 4: Pin Assignments and Numbering for the Medium/Full Configuration 26-pin SDR Connector

5.3 Cabling Requirements

5.3.1 Power Cable

A single power cable is used to supply auxiliary power to the camera. DC ground and the camera housing (along with the shield contacts of all connectors) are connected within the camera (see Fig. 12).

The end of the power cable that connects to the camera's 6-pin connector must be terminated with the female Hirose micro plug (part number HR10A-7P-6S) or the equivalent. The cable must be wired as shown in Fig. 12.

For proper EMI protection, the power cable terminated with the Hirose connector and attached to the camera must be a twin-cored, shielded cable. Also, the Hirose plug must be connected to the cable shield and the shield must be connected to earth ground at the power supply.

Close proximity to strong electromagnetic fields should be avoided.

NOTICE
An incorrect plug can damage the 6-pin connector. The plug on the cable that you attach to the camera's 6-pin connector must have 6 female pins.

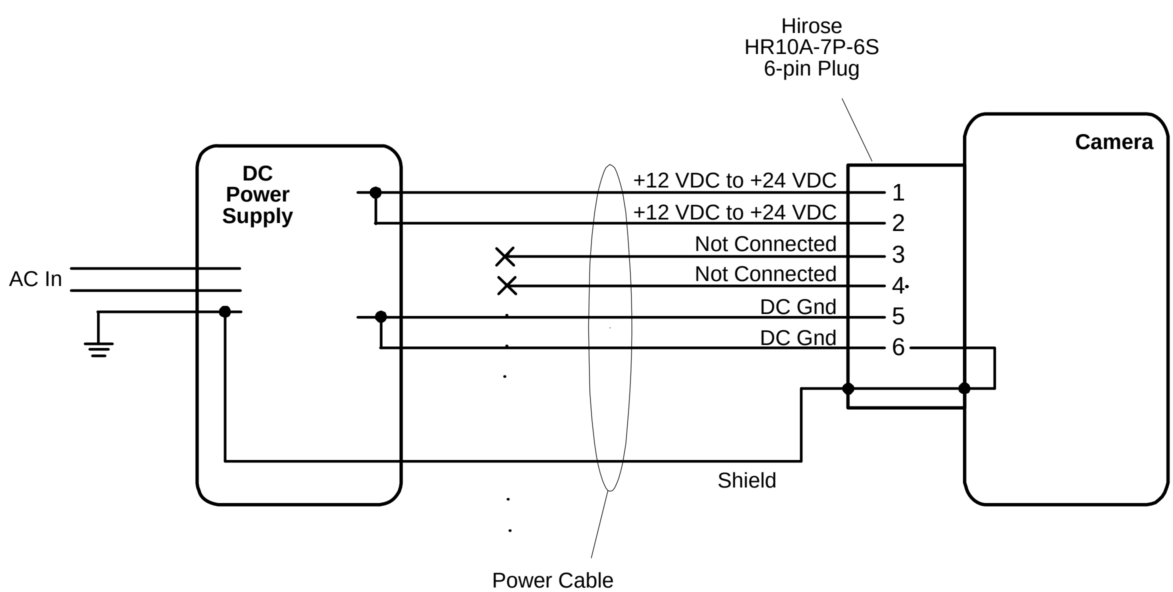


Fig. 12: Power Cable

5.3.2 Camera Link Cables

The Camera Link cables must meet the Mini Camera Link cable specifications specified in the Camera Link Standard.

Close proximity to strong electromagnetic fields should be avoided and the cables used should be as short as possible.

Cable Length Limitations

Generally, Camera Link cables of up to 10 m length can be used for Camera Link cameras. However, when operating cameras at pixel clock speeds of 65 MHz and above, we strongly recommend to use shorter cables in accord with Fig. 13 to ensure the integrity of data transmission. For example, when operating the camera at 84.0 MHz a Camera Link cable no longer than approximately 5 m should be used for reliable data transmission.

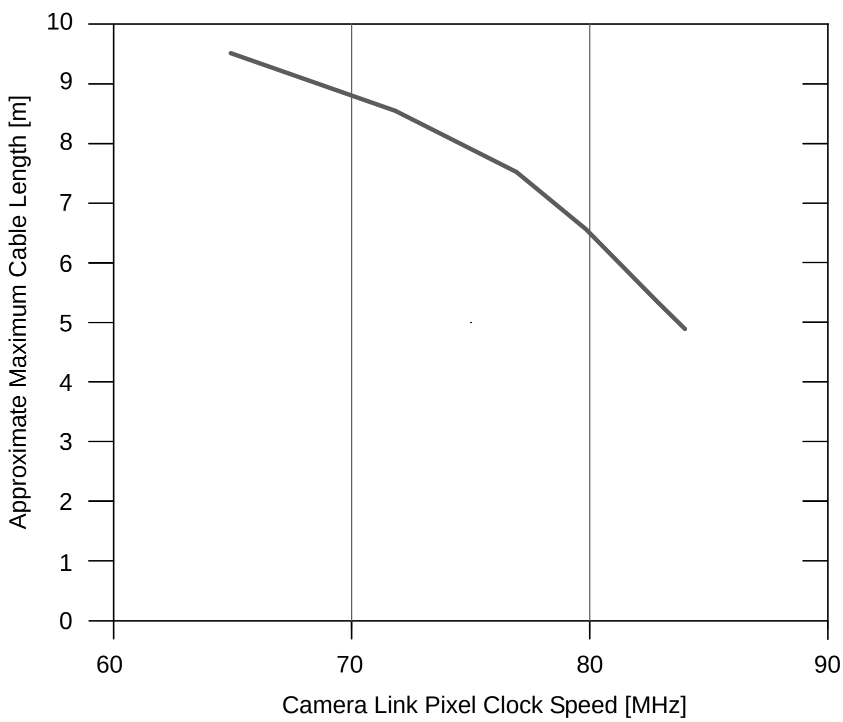


Fig. 13: Approximate Maximum Camera Link Cable Lengths for Reliable Data Transmission at Elevated Camera Link Pixel Clock Speeds.

Note that the maximum cable lengths indicated in Fig. 13 are only very approximate. For practical matters, the maximum cable length not only depends on Camera Link pixel clock speed but also on other factors e.g. on the quality of the frame grabber and on the harshness of the electromagnetic environment.

5.4 Camera Power

Power can be supplied to the camera via the 6-pin connector (auxiliary power).

5.4.1 Supplying Camera Power Via the 6-pin Connector

Power can be supplied to the camera via the 6-pin connector on the back of the camera. The nominal operating voltage is +12 VDC (± 10 %) with an allowed range of +12 VDC (-10 %) to +24 VDC (+5 %), effective on the camera's connector. Ripple must be less than one percent. Power consumption is as shown in the specification tables in Section 1 of this manual.

Close proximity to strong magnetic fields should be avoided.

For information about

- a description of the connector pinouts, see Section 5.2.1 on page 32
- a description of the power cable to be used with the 6-pin connector Section 5.3.1 on page 36

⚠ DANGER

Electric Shock Hazard

Non-approved power supplies may cause electric shock. Serious injury or death may occur.

You must use a camera power supply which meets the Safety Extra Low Voltage (SELV) and Limited Power Source (LPS) requirements.

⚠ WARNING

Fire Hazard

Non-approved power supplies may cause fire and burns.

You must use a camera power supply which meets the Limited Power Source (LPS) requirements.

NOTICE

- Applying incorrect power can damage the camera.

  The camera‘s nominal operating voltage is+12 VDC (± 10 %) with an allowed range of +12 VDC (- 10 %) to +24 VDC (+ 5 %), effective on the camera‘s connector;< 1 % ripple.
- Applying power with the wrong polarity can severely damage the camera.

  Make sure that the polarity of the power applied to the camera is correct. Applying power with the wrong polarity can severely damage the camera.
- Make sure to use a power supply that can supply at least 8 W.

NOTICE

An incorrect plug can damage the 6-pin connector.

The plug on the cable that you attach to the camera’s 6-pin connector must have 6 female pins.

NOTICE

Making or breaking Camera Link connections incorrectly can severely damage the camera.

1. Switch off the power to the 6-pin connector before you connect or disconnect the Camera Link cables.
2. If you can’t switch off the power, be sure that:
   a. The plug for the 6-pin connector is the last plug that you plug into the camera when making connections.
   b. The plug for the 6-pin connector is the first plug that you unplug from the camera when breaking connections.

5.4.2 LED Indicator

The LED indicator on the back of the camera signals whether power is present and also provides some basic error indications for the camera. For details, see Section 9.8.1 on page 145.

5.5 I/O in the Camera Link Interface

5.5.1 Inputs

The camera is equipped with four input lines built into the Camera Link interface. These lines are designated as CC1, CC2, CC3, and CC4 as specified in the Camera Link standard. Typically, input signals are applied to these lines by the frame grabber board attached to the camera. The frame grabber board can typically be configured to supply different types of signals to these inputs as required by the camera user.

On the camera side, you can select these inputs to act as the source signal for the acquisition start trigger frame start trigger signals.

For more information about using CC1, CC2, CC3, and CC4 on the specific type of frame grabber installed in your system, refer to the documentation for you frame grabber board.

You can also obtain some general information about how these lines are implemented in the Camera Link interface from the Basler document named *Basler ace and Basler beat Camera Link Information for Frame Grabber Designers* (AW000990). You can obtain the document from the Basler website:

www.baslerweb.com

5.5.1.1 Input Line Debouncers

Each individual input line is equipped with a debouncer. The debouncer aids in discriminating between valid and invalid input signals and only lets valid signals pass to the camera. The debouncer value specifies the minimum time that an input signal must remain high or remain low in order to be considered a valid input signal.

We recommend setting the debouncer value so that it is slightly greater than the longest expected duration of an invalid signal. Setting the debouncer to a value that is too short will result in accepting invalid signals. Setting the debouncer to a value that is too long will result in rejecting valid signals.

Note that the debouncer creates a delay between the arrival of a valid signal at the camera and its transfer to the camera's internal circuitry. The duration of the delay will be determined by the debouncer value.

Fig. 14 illustrates how the debouncer filters out invalid input signals, i.e. signals that are shorter than the debouncer value. The diagram also illustrates how the debouncer delays a valid signal.

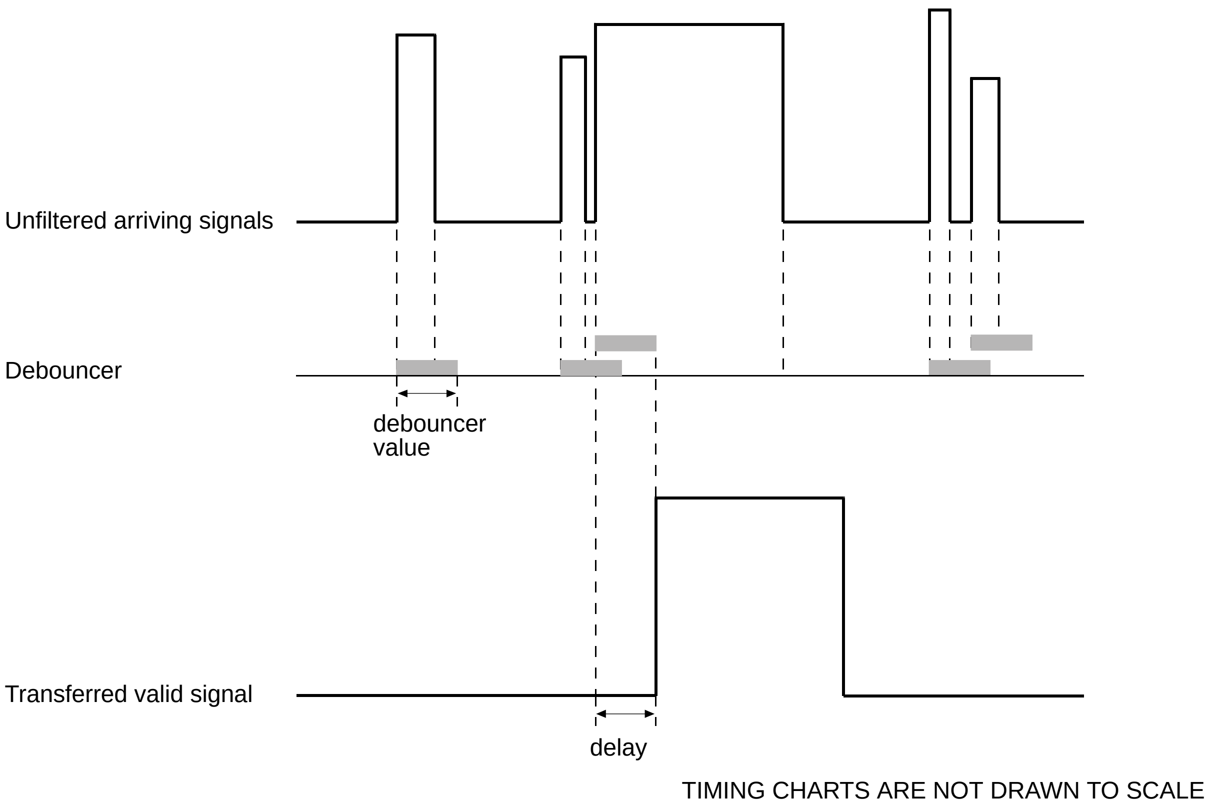


Fig. 14: Filtering of Input Signals by the Debouncer

Setting the Debouncer Using Basler pylon

The debouncer value is determined by the value of the Line Debouncer Time Abs parameter. The parameter is set in microseconds and can be set in a range from 0 to approximately 1 s.

To set the debouncer for the input lines:

1. Use the Line Selector to select CC1, CC2, CC3, or CC4.
2. Set the value of the Line Debouncer Time Abs parameter.

You can set the Line Selector and the value of the Line Debouncer Abs parameter 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:

```
// Select the CC1 line
Camera.LineSelector.SetValue( LineSelector_CC1 );

// Set the parameter value to 150 microseconds
Camera.LineDebouncerTimeAbs.SetValue( 150 );
```

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 3.1 on page 23.

Setting the Debouncer Using Direct Register Access

To set the debouncer for the input lines:

1. For the CC1 line, set the value (in milliseconds) of the Input Debouncer Time CC1 register as desired.
2. For the CC2 line, set the value of the Input Debouncer Time CC2 register.
3. For the CC3 line, set the value of the Input Debouncer Time CC3 register.
4. For the CC4 line, set the value of the Input Debouncer Time CC4 register.

For more information about direct register access, see Section 3.2 on page 25.

5.5.1.2 Input Line Inverters

You can set CC1, CC2, CC3, and the CC4 line to invert or not to invert the input signal.

Setting an Input Line for Invert Using Basler pylon

To set the invert function on an input line:

1. Use the Line Selector to select CC1, CC2, CC3, or CC4.
2. Set the value of the Line Inverter parameter to true to enable inversion on the selected line or 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:

```
// Select the CC1 line
Camera.LineSelector.SetValue( LineSelector_CC1 );
// Disable the inverter on the selected line
Camera.LineInverter.SetValue( false );
```

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 3.1 on page 23.

Setting an Input Line for Invert Using Direct Register Access

To set the invert function on an input line:

1. For the CC1 line, set the value of the Line Inverter CC1 register to 0 (false) or 1 (true) as desired.
2. For the CC2 line, set the value of the Line Inverter CC2 register.
3. For the CC3 line, set the value of the Line Inverter CC3 register.
4. For the CC4 line, set the value of the Line Inverter CC4 register.

For more information about direct register access, see Section 3.2 on page 25.

5.5.1.3 Selecting an Input Line as a Source Signal for a Camera Function

You can select an input line as the source signal for the following camera functions:

- acquisition start trigger
- frame start trigger

Note that to use an input line as the source signal for a camera function, you must apply a signal to the input line that is appropriately timed for the function.

Default Input Line Selection

Software trigger is selected as the source signal for the camera's acquisition start trigger function.

Line CC1 is selected as the source signal for the camera's frame start trigger function.

5.5.2 Outputs

As specified in the Camera Link standard, a "CL Spare" data bit is included in the Camera Link interface. On Basler beat Camera Link cameras, the CL Spare data bit can be used as a camera output line.

You can select any one of the camera's standard output signals to act as the source signal for the CL Spare output line. The camera has these standard output signals available:

- Acquisition Trigger Wait
- Frame Trigger Wait
- Exposure Active
- User Output
- Frame Cycle
- Off

When you select "user output" as the source signal for the CL Spare output line, you can use the camera's API to set the status of the line as you desire.

When you select "off" as the source signal, the output is disabled.

The CL Spare data bit is not available when the camera is used with either of these Camera Link tap geometries:

- 1X8-1Y Camera Link tap geometry used with the Mono 10 pixel format
- 1X10-1Y Camera Link tap geometry.

The CL Spare data bit is not directly accessible by the camera user. The data bit must be accessed via the frame grabber attached to the camera. Not all frame grabbers provide users with direct access to this bit. Consult your frame grabber supplier for more information.

For more information about using the CL Spare bit on your specific type of frame grabber installed in your system, refer to the documentation for you frame grabber board.

You can also obtain some general information about how this bit is implemented in the Camera Link interface from the Basler document named *Basler ace and Basler beat Camera Link Information for Frame Grabber Designers* (AW000990). You can obtain the document from the Basler website: www.baslerweb.com

5.5.2.1 Minimum Output Pulse Width

The minimum output pulse width feature allows you to set the output signal to a minimum width. In this way, you can ensure that even very narrow camera output signals will reliably be detected by other devices.

- 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.

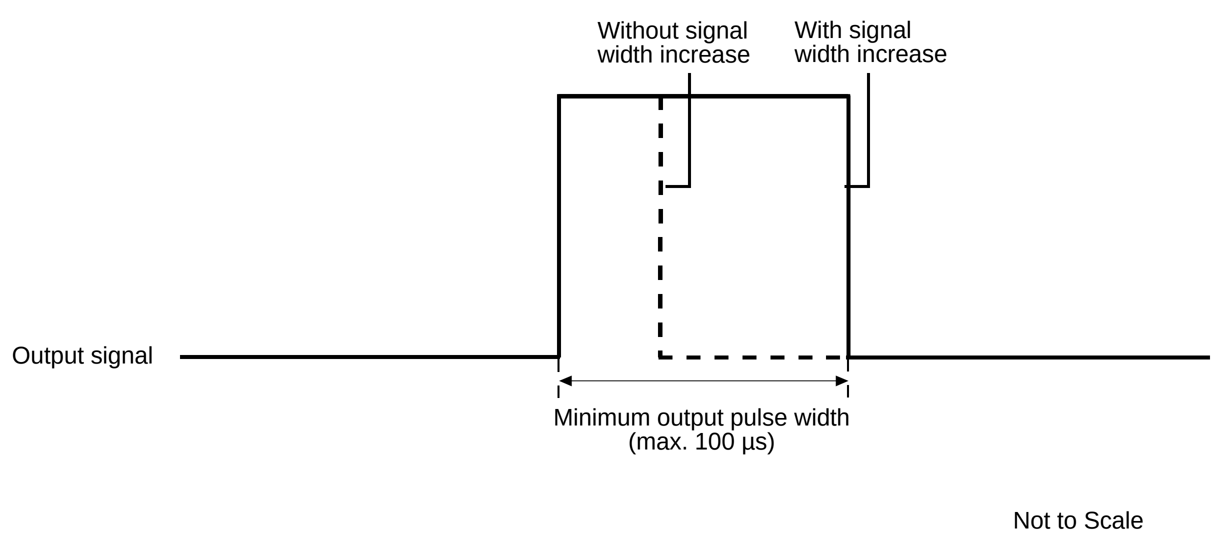


Fig. 15: Increasing the Signal Width of an Output Signal

Setting the Minimum Output Pulse Width

The minimum width of the output pulse can be set in a range from 0 to 100 µs.

ⓘ	Note that the CL Spare bit will not be available if the camera is set to operate with 1X8-1Y or the 1X10-1Y tap geometry, Therefore, in these cases, the minimum output pulse width can not be set.

To set the value for the minimum output pulse width using Basler pylon:

1. Use the Line Selector to select CL Spare.
2. Set the value (in microseconds) of the MinOutPulseWidthAbs parameter as desired to set the CL Spare output signals to a minimum width. The value can be set in a range from 0 to 100 µs.

The MinOutPulseWidthAbs parameter sets the CL Spare output signals to a minimum width. The parameter is set in microseconds and can be set in a range from 0 to 100 µs

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 output line
Camera.LineSelector.SetValue(LineSelector_CLSpare);

// 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 selecting the source signal for the output line on the camera, see Section 5.5.2.3 on page 47.

To set the value for the minimum output pulse width using direct register access:

1. Set the value (in microseconds) of the Min Out Pulse Width CL Spare register as desired.

For more information about direct register access, see Section 3.2 on page 25.

5.5.2.2 Output Line Inverters

You can set the CL Spare output line to invert or not to invert the input signal.

Setting the CL Spare Output Line for Invert Using Basler pylon

To set the invert function on the CL Spare output line:

1. Use the Line Selector to select CL Spare.
2. Set the value of the Line Inverter parameter to true to enable inversion on the selected line or 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:

```
// Select the CL Spare output line
Camera.LineSelector.SetValue( LineSelector_CLSpare );
// Disable the inverter on the selected line
Camera.LineInverter.SetValue( false );
```

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 3.1 on page 23.

Setting the CL Spare Output Line for Invert Using Direct Register Access

To set the invert function on the CL Spare output line:

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

For more information about direct register access, see Section 3.2 on page 25.

5.5.2.3 Selecting the Source Signal for the Output Line

To make the output line useful, you must select a source signal for the output line. For the available output signals, see Section 5.5.2 on page 44.

Selecting the Source Signal Using Basler pylon

To select a standard output signal as the source signal for the CL Spare output line or to select User Output or Off:

1. Use the Line Selector to select CL Spare as the output line.
2. Set the value of the Line Source Parameter to Acquisition Trigger Wait, Frame Trigger Wait, Exposure Active, User Output, Frame Cycle or Off. This will select the source signal for the line.

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:

```
// Select the acquisition trigger wait signal for the CL Spare output line
Camera.LineSelector.SetValue( LineSelector_CLSpare );
Camera.LineSource.SetValue( LineSource_AcquisitionTriggerWait );

// Select the frame trigger wait signal for the CL Spare output line
Camera.LineSelector.SetValue( LineSelector_CLSpare );
Camera.LineSource.SetValue( LineSource_FrameTriggerWait );

// Select the exposure active signal for the CL Spare output line
Camera.LineSelector.SetValue( LineSelector_CLSpare );
Camera.LineSource.SetValue( LineSource_ExposureActive );

// Select the user output signal for the CL Spare output line
Camera.LineSelector.SetValue( LineSelector_CLSpare );
Camera.LineSource.SetValue( LineSource_UserOutput );

// Select the frame cycle signal for the CL Spare output line
Camera.LineSelector.SetValue( LineSelector_CLSpare );
Camera.LineSource.SetValue( LineSource_FrameCycle);

// Disable the CL Spare output line
Camera.LineSelector.SetValue( LineSelector_CLSpare );
Camera.LineSource.SetValue( LineSource_Off );
```

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 3.1 on page 23.

For more information about working with outputs that have User Output ("user settable") as the signal source, see Section 5.5.2.4 on page 50.

Selecting the Source Signal Using Direct Register Access

To select a camera output signal as the source signal for the CL Spare output line or to designate the line as user settable via direct register access:

To select a standard output signal as the source signal for the CL Spare output line or to select User or Disabled:

1. Set the value of the Line Source CL Spare register to Acquisition Trigger Wait, Frame Trigger Wait, Exposure Active, User, Frame Cycle or Disable, as desired.
2. If you set User in the Line Source CL Spare register, set the value of the User Output CL Spare register to 1 (true) or 0 (false) as desired

For more information about direct register access, see Section 3.2 on page 25.

Default Output Line Source Signal Selections

By default, the camera’s Exposure Active signal is selected as the source signal for the CL Spare output line.

5.5.2.4 Setting the State of a User Settable Output Line

As mentioned in the previous section, you can select "user output" as the signal source for the CL Spare output line. For the output line that has "user output" as the signal source, you can use camera parameters to set the state of the line.

ⓘ	If you have the invert function enabled on the output line that is designated as a user output, the user setting sets the state of the line before the inverter.

Setting the State of the User Output Line Using pylon

To set the state of the user output line:

1. Use the User Output Selector to select CL Spare as the output line you want to set.
2. Set the value of the User Output Value parameter to true (high) or false (low). This will set the state of the selected 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 select "user settable" as the source signal for the CL spare output line and how to set the state of the output line:

```
// Select "user output" as the signal source for the CL Spare output line
Camera.LineSelector.SetValue( LineSelector_CLSpare );
Camera.LineSource.SetValue( LineSource_UserOutput );

//Set the state of the CL Spare output line and then read the state
Camera.UserOutputSelector.SetValue( UserOutputSelector_CLSpare );
Camera.UserOutputValue.SetValue( true );
bool currentUserOutputClSpareState = 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 of a User Output Line Using Direct Register Access

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

To set the state of the user settable output line:

1. Set the value of the User Output CL Spare register to 1 (true) or 0 (false) as desired.

For more information about direct register access, see Section 3.2 on page 25.

5.6 Checking the Status of the I/O Lines

5.6.1 Checking the Status of a Single Line

Checking the Status of an I/O Line Using Basler pylon

To check the current status of an I/O line:

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

You can set the Line Selector and read the Line Status 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 read the parameter value:

```
// Select the CC1 line and read the state
Camera.LineSelector.SetValue( LineSelector_CC1 );
bool CC1 = Camera.LineStatus.GetValue( );

// Select the CLSpare line and read the state
Camera.LineSelector.SetValue( LineSelector_ClSpare );
bool ClSpareState = 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 API and the pylon Viewer, see Section 3.1 on page 23.

Checking the Status of an I/O Line Using Direct Register Access

To check the current status of an I/O line:

1. For the CC1 line, read the value of the Line Status CC1 register. The value will indicate 1 (true) or 0 (false).
2. For the CC2 line, read the value of the Line Status CC2 register.
3. For the CC3 line, read the value of the Line Status CC3 register.
4. For the CC4 line, read the value of the Line Status CC4 register.
5. For the CL Spare line, read the value of the Line Status CL Spare register.

For more information about direct register access, see Section 3.2 on page 25.

5.6.2 Checking the Status of All Lines

Checking the Status Using Basler pylon

You can determine the current status of all input and output lines by reading 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 Fig. 16, certain bits in the value are associated with each I/O line, and each of these bits will indicate the status of the associated line. If a bit is 0, it indicates that the status of the associated line is currently low. If a bit is 1, it indicates that the status of the associated line is currently high.

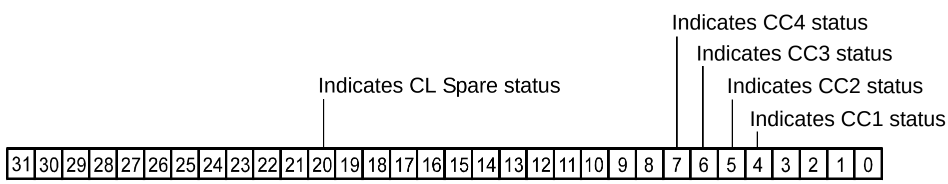


Fig. 16: Line Status All Parameter Bits

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

Checking the Status Using Direct Register Access

To check the current status of all I/O lines:

1. Read the value of the Line Status All register.

   The register holds a 32-bit value that indicates the status of each I/O line. The association of bits with I/O lines is similar to what was described above for access via Basler pylon.

For more information about direct register access, see Section 3.2 on page 25.

5.7 Checking the Line Logic of the I/O Lines

Checking the Line Logic Using Basler pylon

To check the line logic for an I/O line:

1. Use the Line Selector parameter to select an I/O line.
2. Read the value of the Line Logic parameter to determine the type of line logic used by the line. The parameter will indicate whether the logic is positive or negative and therefore whether the inverter is disabled (positive logic) or enabled (negative logic).

You can set the Line Selector and read the Line Logic 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 read the parameter value:

```
// Select the CC1 line and read the line logic type
Camera.LineSelector.SetValue( LineSelector_CC1 );
LineLogicEnums lineLogicCC1 = Camera.LineLogic.GetValue( );

// Select the CLSpare line and read line logic type
Camera.LineSelector.SetValue( LineSelector_ClSpare );
LineLogicEnums lineLogicClSpare = Camera.LineLogic.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 3.1 on page 23.

Checking the Line Logic Using Direct Register Access

To check the Line Logic of an I/O line via direct register access:

To check the line logic for an I/O line :

1. For the CC1 line, read the value of the Line Logic CC1 register. The value will indicate 1 (positive) or 0 (negative).
2. For the CC2 line, read the value of the Line Logic CC2 register.
3. For the CC3 line, read the value of the Line Logic CC3 register.
4. For the CC4 line, read the value of the Line Logic CC4 register.
5. For the CL Spare line, read the value of the Line Logic CL Spare register.

For more information about direct register access, see Section 3.2 on page 25.

6 Acquisition Control

This section provides detailed information about controlling frame acquisition. "Frame acquisition" refers to the process by which the camera obtains an image (a "frame").

You will find details about triggering frame acquisition, about setting the exposure time, about setting the camera's frame acquisition rate, and about how the camera's maximum allowed frame acquisition rate can vary depending on the current camera settings.

The following elements are involved in controlling the acquisition of frames:

- acquisition mode
- acquisition start trigger with an associated setting of the trigger mode
- frame start trigger with an associated setting of the trigger mode
- exposure time control

A code snippet illustrating acquisition control by setting parameter values via the API is given in Section 6.3.4 on page 68.

6.1 Acquisition Mode

The camera's Acquisition Mode parameter has two settings:

- Single frame: If the Acquisition Mode parameter is set for single frame, a single frame can be obtained with the next frame start trigger.
- Continuous: If the Acquisition Mode parameter is set for continuous, frame acquisitions can be triggered as desired. Each time a proper frame trigger is applied, the camera will acquire and transmit a frame.

6.2 Acquisition Start Triggering

The acquisition start trigger is essentially 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.

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 camera will respond to external acquisition start trigger signals. The initial acquisition status of the camera will be "waiting for acquisition start trigger" (see Fig. 28 on page 95). 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".

In the following, acquisition start triggering is presented in greater detail.

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 can't 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 for it to exit 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 options available for this 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 PC.
- One of the lines CC1, CC2, CC3 or CC4. For example, 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.

If the Trigger Source parameter for the acquisition start trigger is set to CC1, CC2, CC3, or CC4 you must also set the Trigger Activation parameter. The options available for this 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 acquisition start trigger signal to CC1, CC2, CC4, or CC4.

For more detailed information about the CC1, CC2, CC3, and CC4 inputs in the Camera Link interface, refer to the document called *Basler ace and beat Camera Link Information for Frame Grabber Designers* (AW000990). You can download 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 can't 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 theTrigger 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 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 API to set the Trigger Mode to on, the Trigger Source to CC1, 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 to line CC1
Camera.TriggerSource.SetValue (TriggerSource_CC1);
// 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 3.1 on page 21.

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, CC1, CC2, CC3 or CC4.
- For the selected trigger source (CC1, CC2, CC3, 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 direct register access, see Section 3.2 on page 23.

6.2.4 Using a Software Acquisition Start Trigger Signal

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 can't 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 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 3.1 on page 21.

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 direct register access, see Section 3.2 on page 23.

6.2.5 Using a Hardware Acquisition Start Trigger Signal

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 CC1, CC2, CC3 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 can’t react to frame start trigger signals when in this acquisition status. When the appropriate ExASTrig signal is applied to the selected source, 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 6.2.5.3 on page 63.

For more information about CC1, CC2, CC3, and CC4, see Section 6.1 on page 47.

6.2.5.2 Acquisition Start Trigger Delay

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

The acquisition start trigger delay may 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.

If you are parameterizing the camera with Basler pylon, the Trigger Delay Abs parameter will determine the length of the delay.

If you are parameterizing the camera via direct register access, the Trigger Delay Raw Acquisition Start register will determine the length of the delay.

> The acquisition start trigger delay will not operate if the Acquisition Start Trigger Mode parameter is set to off or if you are using a software acquisition start trigger.

6.2.5.3 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 software by using the pylon API. The following code snippet illustrates using the API to set the parameter values required to enable rising edge hardware acquisition start triggering with line CC1 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);
// Configure the CC1 line as an input
Camera.LineSelector.SetValue(LineSelector_CC1);
Camera.LineMode.SetValue(LineMode_Input);
// Set the source for the selected trigger to CC1
Camera.TriggerSource.SetValue (TriggerSource_CC1);
// 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 the CC1 line 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 3.1 on page 21.

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 CC1, CC2, CC3, 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 direct register access, see Section 3.2 on page 23.

6.3 Frame Start Triggering

The frame start trigger is used to start a frame acquisition. Keep in mind that the camera will only react to a frame start trigger when the frame start trigger is valid. If the frame start trigger is invalid, frame start triggers will be ignored.

6.3.1 Exposure Time Control

The exposure time for each frame acquisition is determined by the value of the camera's Exposure Time parameter. The parameter takes control when the Trigger Mode parameter for frame start triggering is set to Off or when it is set to On and used in conjunction with all kinds of triggers and trigger modes, except with the trigger width exposure mode (see Section 6.3.3 on page 67 through Section 6.3.6 on page 74).

The minimum and the maximum allowed exposure time for each frame acquisition are as shown in Table 5. The minimum allowed exposure times depend on Camera Link pixel clock speed and Camera Link tap geometry.

When the trigger width exposure mode is used the trigger signal itself will control the exposure time (see Section 6.3.6.2 on page 75) and will allow to exceed the maximum allowed exposure time given in Table 5.

ⓘ	Note that very long exposure times will usually not be useful as they may result in image quality degradation and overexposure.

Camera Link Pixel Clock Speed [MHz]	Camera Link Tap Geometry	Minimum Allowed Exposure Time [µs]	Maximum Allowed Exposure Time [µs]
32.5	1X2-1Y	65	10000000
	1X3-1Y	44	
	1X8-1Y	33	
	1X10-1Y	27	
65	1X2-1Y	33	
	1X3-1Y	44	
	1X8-1Y	33	
	1X10-1Y	27	
84.0	1X2-1Y	26	
	1X3-1Y	34	
	1X8-1Y	26	
	1X10-1Y	25	

Table 5: Minimum and Maximum Allowed Exposure Times for the Exposure Time Parameter

If you are using a GenICam compliant tool such as the Basler pylon Viewer and you attempt to set the exposure time to exactly the minimum allowed or to exactly the maximum allowed, you will see unusual error codes. This is an artifact of a rounding error in the GenICam interface architecture. As a workaround, you could set the exposure time slightly above the minimum or below the maximum. Values between the minimum and the maximum are not affected by the problem.

For more information about setting exposure time, see Section 6.3.4 on page 68.

6.3.2 Frame Start with Trigger Mode = Off

When the Trigger Mode parameter is set to Off, selection of a source signal for the frame start trigger is not required. With the trigger mode set to Off, the camera operates the frame start trigger automatically and generates the frame start triggers internally. The rate at which the frame start triggers are generated will be determined by the camera's Acquisition Frame Rate Abs parameter as far as the set acquisition frame rate is compatible with the current Exposure Time parameter setting:

- If the Acquisition Frame Rate Abs parameter is set to a value less than the maximum allowed frame acquisition rate, the camera will generate triggers at the rate specified by the parameter setting.
- If the Acquisition Frame Rate Abs parameter is set to a value greater than the maximum allowed frame acquisition rate, the camera will generate frame start triggers at the maximum allowed frame rate.
- If the Exposure Time parameter is set to a value that is too large for the Acquisition Frame Rate Abs parameter setting, the camera will generate frame start triggers at a decreased frame rate. This frame rate will be the maximum allowed as far as permitted by the current Exposure Time parameter setting.

For more information about the maximum allowed frame rate, see Section 6.7 on page 103.

Exposure Time Control with Trigger Mode = Off

When the trigger mode is set to Off, the exposure time for each frame acquisition is determined by the value of the camera's Exposure Time parameter. For ranges of allowed values, see Table 5 on page 65.

6.3.3 Frame Start with Trigger Mode = On

When Trigger Mode is set to On, you must select a source signal for the frame start trigger. Trigger Source specifies the source signal. The available selections for Trigger Source are:

- Software - When the frame start trigger source is set to software, the user triggers frame start by issuing a TriggerSoftware command to the camera from the host PC. Each time a TriggerSoftware command is received by the camera, the frame start trigger will become valid. It will become invalid during frame acquisition and will become valid again when the next TriggerSoftware command is received and when the camera is ready again for a new frame acquisition.
- 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 (a "hardware trigger") 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 (a "hardware trigger") into CC2 in the Camera Link interface.
- CC3 - When the source signal is set to CC3, you apply a frame start trigger signal to the camera by injecting an externally generated electrical signal (a "hardware trigger") into CC3 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 (a "hardware trigger") into CC4 in the Camera Link interface.

If Trigger Source is set to CC1, CC2, CC 3, or CC4, the user 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 source signal will start a frame acquisition.
- Falling Edge - specifies that a falling edge of the source signal will start a frame acquisition.

Typically, a frame grabber is used to supply an electrical frame start signal to CC1, CC2, CC3, or CC4.

By default, CC1 is selected as the source signal for the frame start trigger.

If the Frame Start Trigger Source parameter is set to CC1, CC2, CC 3, or CC4, the externally generated electrical signal applied to the selected input line must be held high for a minimum period for the camera to detect a transition from low to high and must be held low for a minimum period for the camera to detect a transition from high to low:

- The minimum period is 1 µs if the debouncer is not used.
- The minimum period is equal to or larger than the debouncer value if the debouncer is used.

For more information about using a software trigger to control frame acquisition start, see Section 6.3.5 on page 70.

For more information about using a hardware trigger to control frame acquisition start, see Section 6.3.6 on page 74.

Exposure Time Control with Trigger Mode = On

When the frame start trigger mode is set to On, the exposure time for each frame acquisition is determined by the value of the camera's exposure time parameter setting, except when the trigger width exposure mode is selected (for details, see Section 6.3.6.2 on page 75).

For ranges of allowed exposure time parameter values, see Table 5 on page 65.

For more information about setting exposure time, see Section 6.3.4 on page 68.

6.3.4 Setting Trigger Mode, Trigger Source, and Related Parameters

Setting the Parameters Using Basler pylon

You can set the Trigger Mode, the Trigger Source, 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 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 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 in (µs)
Camera.ExposureTimeAbs.SetValue( 300 );
// 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 3.1 on page 23.

Setting the Parameters Using Direct Register Access

To set the trigger mode and to select a trigger source:

1. Set the value of the Trigger Mode Frame Start register to On.
2. Set the value of the Trigger Source Frame Start register to Software, CC1, CC2, CC3, or CC4.
3. If the trigger source is set to CC1, CC2, CC3, or CC4, set the value of the Trigger Activation Frame Start register to Rising Edge or Falling Edge.

To set the frame start trigger mode, the exposure time, and the frame acquisition rate:

1. Set the value of the Trigger Mode Frame Start register to Off.
2. 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 beat cameras is 1 µs.

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

For more information about direct register access, see Section 3.2 on page 25.

6.3.5 Using a Software Frame Start Trigger Signal

6.3.5.1 Introduction

If the Frame Start Trigger Mode parameter 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. Fig. 17 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.

Timed Exposure Mode

When you are using a software trigger signal to start each frame acquisition, the camera's Exposure Mode 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 register if you are parameterizing the camera via direct register access.

The minimum and the maximum allowed exposure times for each acquired frame are as shown in Table 5 in Section 6.3.2 on page 66. Note that the minimum allowed exposure times depend on Camera Link pixel clock speed and Camera Link tap geometry. For more information, see Table 5 in Section 6.3.2 on page 66.

If you are using a GenICam compliant tool such as the Basler pylon Viewer and you attempt to set the exposure time to exactly the minimum allowed or to exactly the maximum allowed, you will see unusual error codes. This is an artifact of a rounding error in the GenICam interface architecture. As a work around, you could set the exposure time slightly above the minimum or below the maximum. Values between the minimum and the maximum are not affected by the problem.

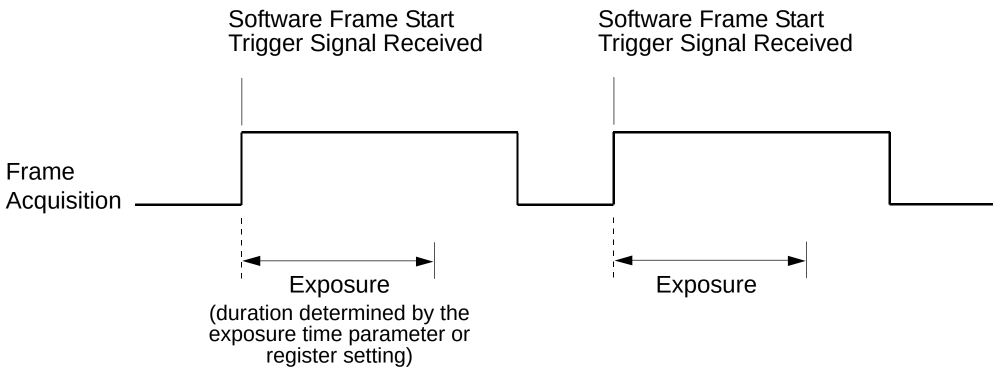


Fig. 17: Frame Acquisition with a Software Frame Start Trigger and Exposure Mode Set to Timed (Base Delays Omitted)

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. This situation is commonly referred to as "overtriggering" the camera. (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 6.3.5.2 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 6.7 on page 103.

Adjustment for Base Delays

To a minor extent, the exposure time will also be controlled by the exposure start delay and the exposure end delay ("base delays"; see Table 7 on page 83). To accurately obtain the wanted exposure time you must adjust the exposure time parameter setting by

- adding the exposure start delay to the wanted exposure time and
- subtracting the exposure end delay from the wanted exposure time.

Further adjustment is necessary if the debouncer is used. For more information about the necessary adjustments, see Section 6.4 on page 82.

6.3.5.2 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 API to set the parameter values and execute the commands related to software frame start triggering.

```
// 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( 300 );

     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 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 3.1 on page 23.

Setting the Parameters and Applying the Signal Using Direct Register Access

To set the parameters needed to perform software frame start triggering:

1. Set the value of the Acquisition Frame Period Enable register to 0 (false).
   (This will disable the camera's ability to internally control the frame period and allow you to control the frame rate with software trigger signals.)
2. Set the value of the Trigger Mode Frame Start register to On.
3. Set the value of the Trigger Source Frame Start register to Software.
4. Set the value of the Exposure Mode register to Timed.
5. 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 beat cameras is 1 µs.

   For example, if you set the Exposure Time Raw register to 1000, the exposure time would be 1000 µs (1000 × 1 µs = 1000 µs).
6. 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:

1. 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 direct register access, see Section 3.2 on page 25.

6.3.6 Using a Hardware Frame Start Trigger Signal

6.3.6.1 Introduction

If the Trigger Mode parameter for the frame start trigger is set to On and the Trigger Source parameter is set to CC1, CC2, CC3, 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, CC3, or CC4 as the trigger source, your frame grabber will apply the electrical signal to the selected input via the Camera Link cable. For more information about applying an ExFSTrig signal to CC1, CC2, CC3, or CC4, see the documentation for your frame grabber.

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 "overtriggering" the camera.

To avoid overtriggering, 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 6.7 on page 103.

For more information about setting the camera for hardware triggering and selecting the source to receive the ExFSTrig signal, see Section 6.3.6 on page 74.

For more information about CC1, CC2, CC3, and CC4, see Section 5.5.1 on page 40.

6.3.6.2 Exposure Modes

If you are triggering the start of frame acquisition with an externally generated frame start trigger (ExFSTrig) signal, the following exposure modes are available:

- Timed exposure mode
- Trigger Width exposure mode

Timed Exposure Mode

When the timed exposure 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.

The minimum and the maximum allowed exposure time for each acquired frame are as shown in Table 5 in Section 6.3.2 on page 66. Note that the minimum allowed exposure times depend on Camera Link pixel clock speed and Camera Link tap geometry.

ⓘ	If you are using a GenICam compliant tool such as the Basler pylon Viewer and you attempt to set the exposure time to exactly the minimum allowed or to exactly the maximum allowed, you will see unusual error codes. This is an artifact of a rounding error in the GenICam interface architecture. As a work around, you could set the exposure time slightly above the minimum or below the maximum. Values between the minimum and the maximum are not affected by the problem.

If the camera is set for rising edge triggering, the exposure time starts when the ExFSTrig signal rises. If the camera is set for falling edge triggering, the exposure time starts when the ExFSTrig signal falls. Fig. 18 illustrates timed exposure with the camera set for rising edge triggering.

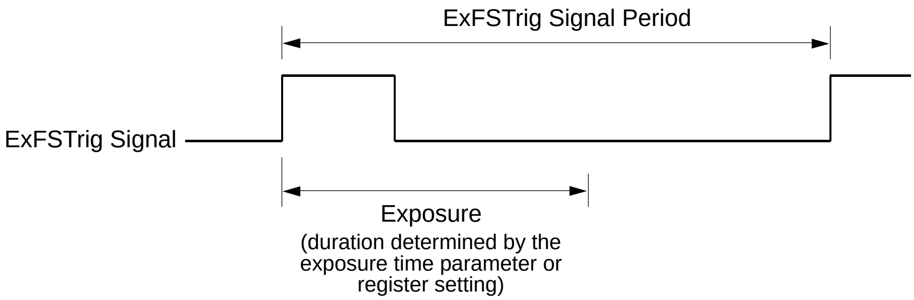


Fig. 18: Timed Exposure with Rising Edge Triggering

Note that if you attempt to trigger a new exposure start while the previous exposure is still in progress (overtriggering), the trigger signal will be ignored. This situation is illustrated in Fig. 19 for rising edge triggering.

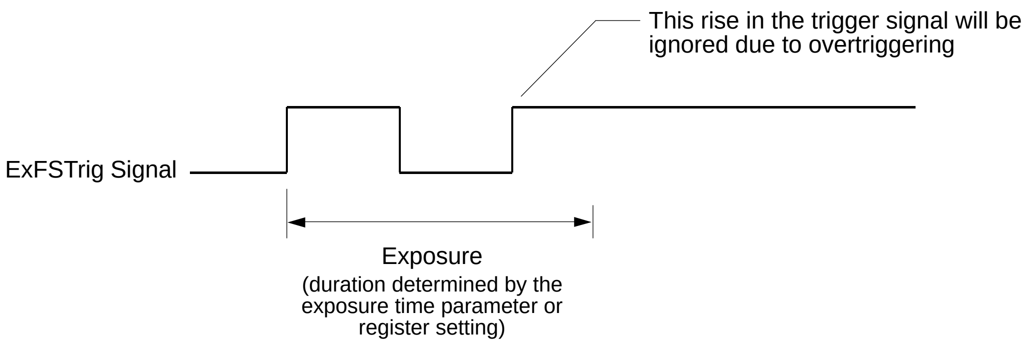


Fig. 19: Overtriggering with Timed Exposure

To a minor extent, the exposure time will also be controlled by the exposure start delay and the exposure end delay ("base delays"; see Table 7 on page 83). To adjust for the base delays, you must adjust the exposure time parameter setting by

- adding the exposure start delay to the wanted exposure time and
- subtracting the exposure end delay from the wanted exposure time.

Further adjustment is necessary if the debouncer is used. For more information about the necessary adjustments, see Section 6.4 on page 82.

Trigger Width Exposure Mode

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

When trigger width exposure mode is selected, the exposure time for each frame acquisition will mainly be controlled by the sum of two individual time periods (see Fig. 20 and Table 6):

- The first time period is the exposure time that is controlled by the ExFSTrig signal: If the camera is set for rising edge triggering, the first time period – and therewith the exposure time – begins when the ExFSTrig signal rises. The first time period ends when the ExFSTrig signal falls.

  If the camera is set for falling edge triggering, the first time period begins when the ExFSTrig signal falls. The first time period ends when the ExFSTrig signal rises.
- The second time period is the exposure time offset, C_4. It is automatically added to the first time period by the camera's sensor. The length of the exposure time offset depends on Camera Link clock speed and Camera Link tap geometry. The exposure time offset is usually similar to the camera's minimum exposure time.

  For the cameras' minimum allowed exposure times, see Table 5 on page 65.

  For exposure time offset values, C_4, see Table 6.

Camera Link Pixel Clock Speed [MHz]	Camera Link Tap Geometry	Exposure Time Offset, C_4 [µs]
32.5	1X2-1Y	64
	1X3-1Y	43
	1X8-1Y	32
	1X10-1Y	26
65	1X2-1Y	32
	1X3-1Y	43
	1X8-1Y	32
	1X10-1Y	26
84.0	1X2-1Y	25
	1X3-1Y	33
	1X8-1Y	25
	1X10-1Y	19

Table 6: Exposure Time Offset Values, C_4

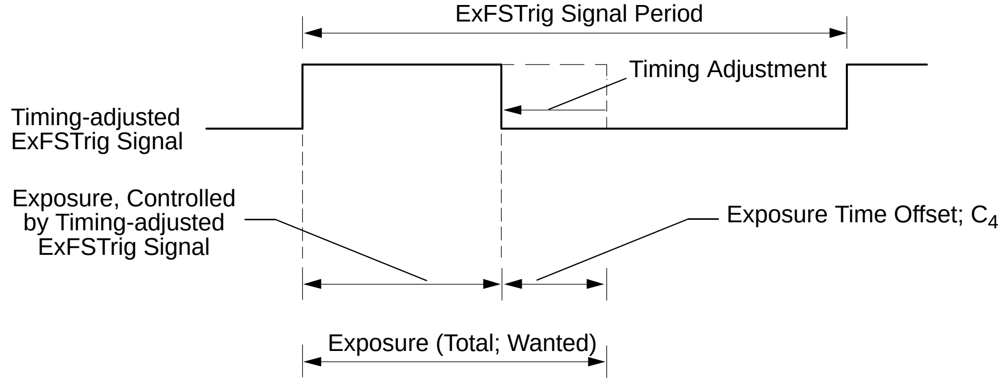


Fig. 20: Trigger Width Exposure with Adjusted Rising Edge Triggering; Adjustment for Base Delays Is Omitted

To obtain a wanted exposure time using the trigger width exposure mode you will have to adjust for the automatically exposure time offset, C_4:

- Subtract the applicable value for C_4 (see Table 6) from the wanted exposure time and use the resulting adjusted time
  - as the high time for the ExFSTrig signal if the signal is not inverted or
  - as the low time if the signal is inverted.

You must adjust the timing for the ExFSTrig signal **only** if you use the trigger width exposure mode. In all other exposure modes (where the Exposure Time parameter is set), the camera automatically adjusts for the exposure time offset, C_4.

Note that the occurrence of the exposure time offset does **not** affect the moment of exposure start.

To a minor extent, the exposure time will also be controlled by the exposure start delay and the exposure end delay ("base delays"; see Table 7 on page 83). To adjust for the base delays, you must adjust the ExSTrig signal timing by

- adding the exposure start delay to the wanted exposure time and
- subtracting the exposure end delay from the wanted exposure time.

For an accurate adjustment of the ExFSTrig signal timing you must also take account of the exposure start and exposure end delays ("base delays"; see Table 7 on page 83):

- add the exposure start delay to the wanted exposure time and
- subtract the exposure end delay from the wanted exposure time.

Example

Let's assume you are operating the camera at a Camera Link Pixel Clock speed of 32.5 MHz, you are using the 1X10-1Y Camera Link tap geometry, the camera is set for rising edge triggering, and you want to use am exposure time of 100 µs. Under these conditions 26 µs of exposure time (see Table 7) will be added automatically to the exposure time that is controlled by the ExFSTrig signal.

To achieve the wanted exposure time of 100 µs, you would therefore keep the ExFSTrig signal high for 74 µs (= 100 µs - 26 µs). Subsequently, the camera would add automatically 26 µs, giving a total of 100 µs exposure time which is the wanted exposure time.

Minimum and Maximum Exposure Times

The minimum allowed exposure times depend on Camera Link pixel clock speed and Camera Link tap geometry. The values are given in Table 5 on page 65. The maximum allowed exposure times are limited to 10000 µs. Keep in mind, however, that using a very long exposure time can lead to significant degradation of the image quality.

ⓘ	If you are using a GenICam compliant tool such as the Basler pylon Viewer and you attempt to set the exposure time to exactly the minimum allowed or to exactly the maximum allowed, you will see unusual error codes. This is an artifact of a rounding error in the GenICam interface architecture. As a work around, you could set the exposure time slightly above the minimum or below the maximum. Values between the minimum and the maximum are not affected by the problem.

When you use the trigger width exposure mode and want to monitor the Frame Trigger Wait signal to avoid overtriggering you must also set an Exposure Overlap Time Max value. This value will allow the Frame Trigger Wait signal to indicate when the camera becomes ready to receive an external frame start trigger.

ⓘ	The joint use of the Frame Trigger Wait signal and the Exposure Overlap Time Max setting is particularly useful when overlapping exposure with sensor readout. For more information, see Section 6.5 on page 87 and Section 6.6 on page 92.

For more information about the Frame Trigger Wait (ExSFTrig Wait) signal, see Section 6.6.2.2 on page 96

You can set an Exposure Overlap Time Max value in two ways: If you are parameterizing the camera with Basler pylon you must set the Exposure Overlap Time Max Abs parameter and if you are parameterizing the camera via direct register access you must set the value of the Exposure Overlap Time Max register.

The correct parameter values depend on the method that you choose to avoid overtriggering:

- If you want to monitor the Frame Trigger Wait signal to avoid overtriggering you must set Exposure Overlap Time Max 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 15 µs to 20 µs. In this case you would set the camera's Exposure Overlap Time Max to 15 µs.

> When using the trigger width exposure mode, consider the shortest **actual** exposure time when setting Exposure Overlap Time Max.

- If you want to avoid overtriggering by obeying certain timing limits during frame acquisition you must set the Exposure Overlap Time Max to an appropriate value.

  For more information about frame acquisition while obeying certain timing limits and setting the Exposure Overlap Time Max parameter value, see Section 6.5.1.2 on page 91.

6.3.6.3 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 software by using the pylon API.

The following code snippet illustrates using the 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 and CC1 as the trigger source.

```
// 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 for the timed exposure mode
Camera.ExposureMode.SetValue( ExposureMode_Timed );
// Set the exposure time
Camera.ExposureTimeAbs.SetValue( 300 );

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

The following code snippet illustrates using the 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 and with CC1 as the trigger source.

```
// 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 to CC1
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.

Setting the Parameters Using Direct Register Access and Applying the Signal

To set the parameters for hardware frame start triggering:

1. Set the value of the Acquisition Frame Period Enable register to 0 (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.)
2. Set the value of the Trigger Mode Frame Start register to On.
3. Set the value of the Trigger Source Frame Start register to receive the external trigger signal on, CC1, CC2, CC3, or CC4.
4. Set the value of the Trigger Activation Frame Start register to Rising Edge or Falling Edge as desired.
5. 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 Basler beat cameras is 1 µs.

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

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

For more information about direct register access, see Section 3.2 on page 25.

6.4 Acquisition Timing Charts

For simplicity, certain intervals, e.g. exposure start and exposure end delay, were not included in the preceding timing charts for frame start triggers. In addition, the various aspects of timing were so far discussed separate from each other. This section provides a more complete view.

Exposure Start and Exposure End Delays

When the frame start trigger mode is set to On and a frame start trigger is sent, there is a delay between the transition of the frame start signal and the actual start of exposure. For example, if you are using the timed exposure mode with rising edge triggering, there is a delay between the rise of the signal and the actual start of exposure.

There is also an exposure end delay, i.e., a delay between the point when exposure should end as explained in the diagrams on the previous page and when it actually does end.

The exposure start and exposure end delays listed below, relate to the interaction between the camera's microcontroller and the FPGA (see Chapter 4 on page 27). The delays apply, regardless of whether a software trigger or a hardware trigger is sent. However, the triggers will occur at different moments, due to the different signals and their different paths.

The exposure start and exposure end delays , also called "base delays", are given in Table 7: See also the timing charts of Fig. 21 and Fig. 22.

Camera Link Pixel Clock Speed [MHz]	Camera Link Tap Geometry	Exposure Start Delay [ns]	Exposure End Delay [ns]
32.5	1X2-1Y	850	650
	1X3-1Y	600	470
	1X8-1Y	490	380
	1X10-1Y	410	330
65	1X2-1Y	490	380
	1X3-1Y	600	470
	1X8-1Y	490	380
	1X10-1Y	410	330
84.0	1X2-1Y	410	330
	1X3-1Y	490	380
	1X8-1Y	410	330
	1X10-1Y	350	280

Table 7: Exposure Start and End Delays (Base Delays)

ⓘ If you are using a GenICam compliant tool such as the Basler pylon Viewer and you attempt to set the exposure time to exactly the minimum allowed or to exactly the maximum allowed, you will see unusual error codes. This is an artifact of a rounding error in the GenICam interface architecture. As a work around, you could set the exposure time slightly above the minimum or below the maximum. Values between the minimum and the maximum are not affected by the problem.

When using the debouncer, there is another component to the start and end delays. This second component is the debouncer setting for the input line. The debouncer setting for the input line must be added to the base start and end delays shown in Table 7 to determine the total start delay and end delay.
For example, assume that you are using a beA4000-62km/kc camera and that you have set the frame start trigger mode to On. Also assume that you use a Camera Link clock speed of 32.5 MHz, have set the camera to 1X2-1Y Camera Link tap geometry, have selected input line CC1 as the source signal for the frame start trigger and that the debouncer parameter for line CC1 is set to 5 µs. In this case:

Total Start Delay = Base Start Delay Value from Table 7 + Debouncer Setting

Total Start Delay = 0.85 µs + 5 µs

Total Start Delay = 5.85 µs

Total End Delay = Base End Delay Value from Table 7 + Debouncer Setting

Total End Delay = 0.65 µs + 5 µs

Total End Delay = 5.65 µs

Time Periods Involved in Frame Acquisition

The process of frame acquisition involves several time periods that are listed below. The time periods are also shown in their timing contexts in Fig. 21 (timed acquisition mode) and Fig. 22 (trigger width acquisition mode).

The charts assume that exposure is triggered by an externally generated frame start trigger (ExFSTrig) signal with rising edge activation, that the debouncer is not used, and that frame acquisitions do not overlap (see Section 6.5 on page 87).

The charts also include the frame valid signals (FVAL), preparing the transmission of frames to the frame grabber. For details about delay A and the additional signals and delays involved in frame transmission, see the *Basler ace and beat Camera Link Information for Frame Grabber Designers* (AW000990).

The **exposure start delay** is the amount of time between the point where the trigger signal transitions and the point where exposure actually begins.

Note that, if the debouncer feature is used, the debouncer setting for the input line must be added to the exposure start delay shown in Fig. 21 and Fig. 22 to determine the total start delay.

The **exposure time offset, C_4,** is automatically added to the exposure time as determined by the exposure time parameter or by the external trigger signal (in trigger width exposure mode). To obtain the wanted exposure time, you must adjust the timing for the ExFSTrig signal **only** if you use the trigger width exposure mode. When the exposure time parameter is used to determine exposure, the adjustment for C_4 is done automatically by the camera.

The **exposure end delay** is the amount of time between the point where the trigger signal transitions and the point where exposure actually ends.

The **exposure overhead, C_1,** occurs with every frame acquisition and is required to prepare the sensor for the next frame acquisition.

The **frame readout time** is the amount of time it takes to read out the data for an acquired frame from the imaging sensor. This time period will vary depending on several camera settings such as the height of the image AOI, the Camera Link tap geometry, the Camera Link pixel clock speed, and others. As explained below, you can read the value of the Readout Time parameter to determine the sensor readout time given the current camera settings.

The timing of the frame transmission out of the camera is described in detail in a separate document, in the *Basler ace and beat Camera Link Information for Frame Grabber Designers* (AW000990). Frame transmission is e.g. characterized by the frame transmission time and the transmission start delay.

- The **frame transmission time** is the amount of time it takes to transmit an acquired frame from the buffer in the camera to the host PC. The frame transmission time will always be equal to the frame readout time.
- The **transmission start delay** is the amount of time between the point where the camera begins reading out the acquired frame data from the sensor to the point where it begins transmitting the data for the acquired frame from the buffer to the host PC.

  For more information about the transmission start delay, see the *Basler ace and beat Camera Link Information for Frame Grabber Designers* (AW000990).

Timing Charts

The timing charts below assume that exposure is triggered by an externally generated frame start trigger (ExFSTrig) signal with rising edge activation. The charts illustrate operation in timed acquisition mode (Fig. 21) and trigger width acquisition mode (Fig. 22).

The frame valid signals (FVAL), included in the charts, prepare the transmission of frames from the camera to the frame grabber. The complete timings for frame transmission are described in the *Basler ace and beat Camera Link Information for Frame Grabber Designers* (AW000990). This document also provides detailed information about delay A and additional signals and delays.

The instants marked "end of exposure" in the timing charts in the *Basler ace and beat Camera Link Information for Frame Grabber Designers* (AW000990), correspond to the instants labelled "UEoE" ("user end of exposure") in Fig. 21 and Fig. 22 of this document.

"UEoE" indicates the instant when exposure for an individual frame ends **as controlled by the user:**

- In timed exposure mode, the instant occurs when the sensor actually ends exposure (as set by the exposure time parameter value; see Fig. 21).
- In trigger width exposure mode, the instant occurs when the ExFSTrig signal goes low (if the signal is not inverted; see Fig. 22). Accordingly, in this case the instant occurs when only part of the total exposure for an acquisition has elapsed.

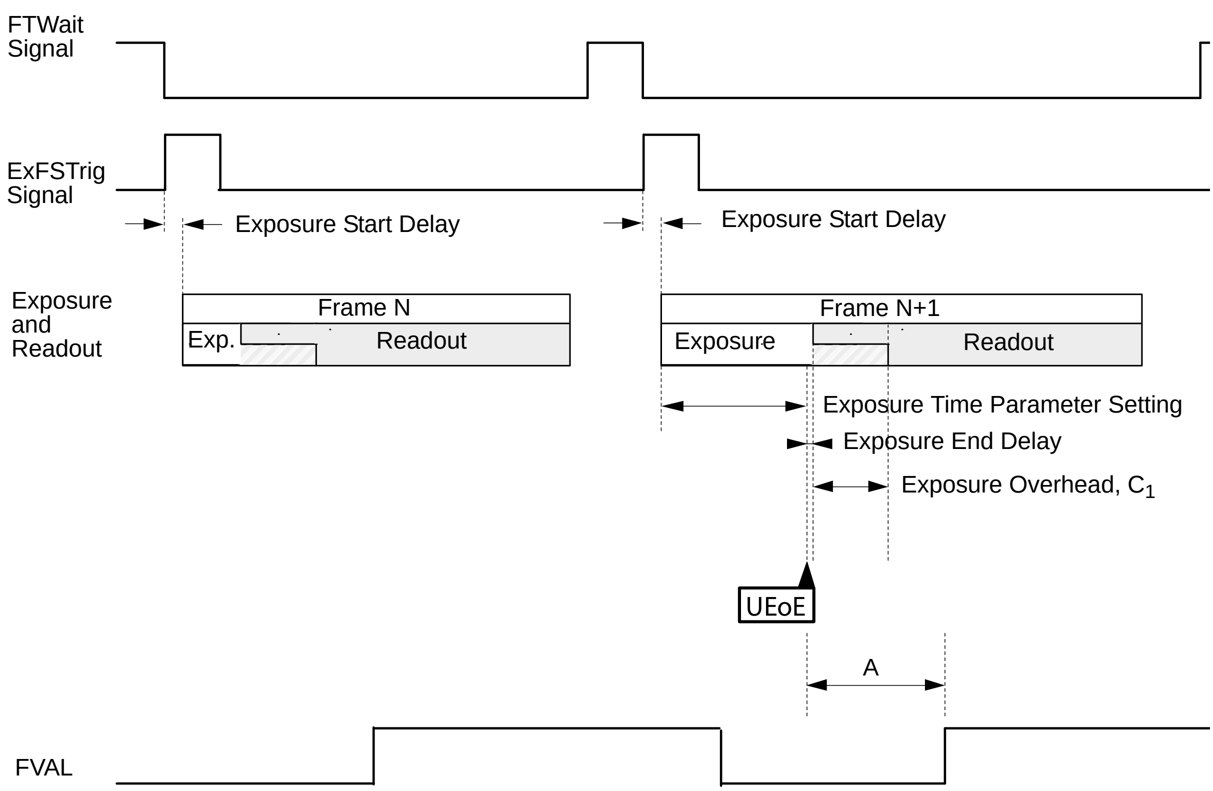


Fig. 21: Timing Chart for Frame Acquisition Using the Timed Exposure Mode

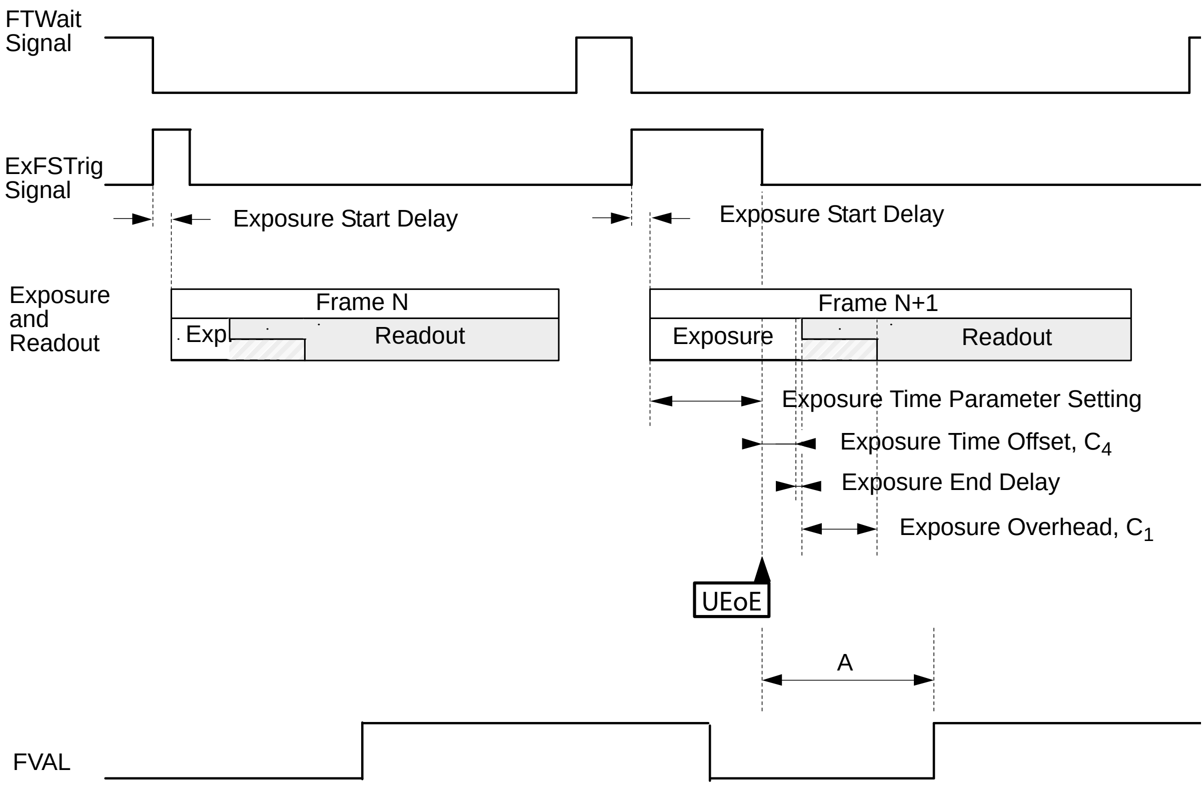


Fig. 22: Timing Chart for Frame Acquisition Using the Trigger Width Exposure Mode

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” acquisition and with “overlapped” acquisition.

Note: For simplicity, exposure start delays, exposure end delays and adjustments of the ExFSTrig signals when used in trigger width exposure mode, are omitted in the following figures. For adjustments in trigger width exposure mode, see Section 6.3.6 on page 74.

For the examples shown in this section the Trigger Activation frame start trigger signal is assumed to be set to Rising Edge. The timing adjustments described in Section 6.3.6 on page 74 are omitted here for simplicity.

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 Fig. 23 here an external frame start trigger us used with the camera set for the trigger width exposure mode.

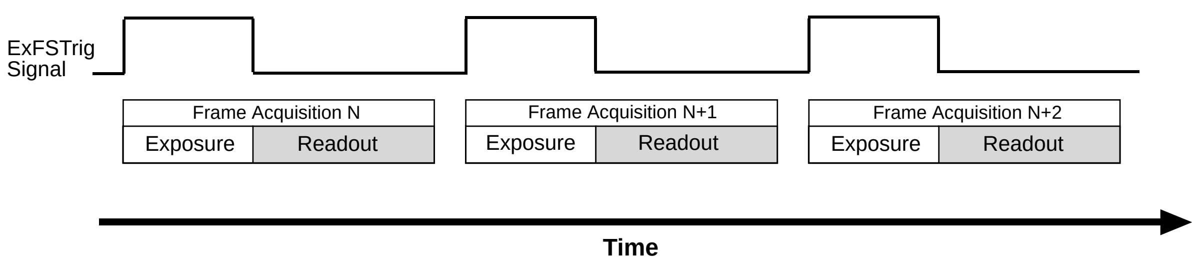


Fig. 23: Non-overlapped Acquisition: Readout and Subsequent Exposure Do Not Overlap

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 acquired frame. This situation is illustrated in Fig. 24 with the camera set for the trigger width exposure mode.

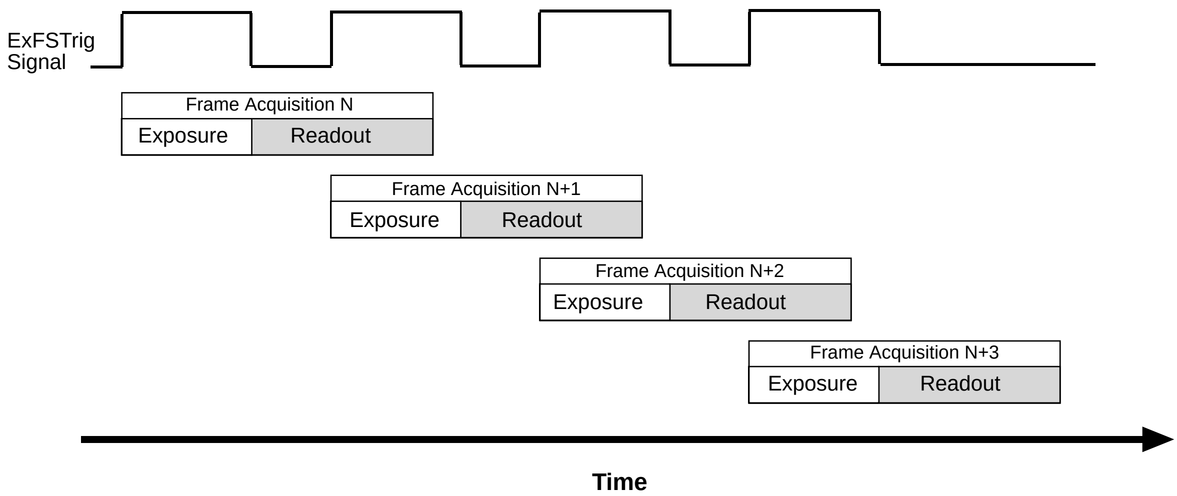


Fig. 24: Overlapped Exposure and Readout

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 overlap 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 the pylon API and the pylon Viewer, see Chapter 3 on page 23.

You can also determine the readout time via direct register access by reading out the value of the Readout Time Raw register.

6.5.1 Guidelines for Overlapped Operation

To ensure smooth frame acquisition and avoid overtriggering, you may only apply a frame acquisition-related trigger when the camera is waiting for it. If the trigger is nonetheless applied, the trigger will be ignored and considered an overtrigger.

The risk of overtriggering exists particularly for overlapped operation where the sequence of frame start triggers must be carefully coordinated both with the camera's exposure time and the sensor readout time.

The following examples use a non-inverted, rising edge external frame start trigger signal (ExFSTrig). For simplicity, exposure start delays, exposure end delays and adjustments of the ExFSTrig signals when used in trigger width exposure mode, are omitted in the following figures.

6.5.1.1 Illegal Frame Acquisition by Overtriggering

Certain attempts of triggering overlapped frame acquisition are illegal and do not result in frame acquisitions: When a frame start trigger signal attempts an illegal frame acquisition the trigger signal will be ignored and, accordingly, no frame acquisition will be performed. In addition, the trigger signal will be reported as an overtrigger (see also Section 6.3.5.1 on page 70 and Section 6.3.6.1 on page 74). Illegal triggering and impossible overlaps are shown in Fig. 25 on page 90 and Fig. 26 on page 90.

Illegal triggering when overlapping frame acquisitions:

- The frame start trigger goes high to start the exposure for frame acquisition N+1 before the exposure or the exposure overhead for frame acquisition N has ended (see Fig. 25 on page 90).

  This would result in the illegal overlap of exposures or of exposure and exposure overhead.

> The exposure overhead (see Fig. 25 on page 90) is part of every exposure process. For simplicity, it is omitted from the other figures illustrating exposure and readout (see, for example, Fig. 26 on page 90).
>
> The duration of the exposure overhead is expressed by constant C_1 (see also Section 6.5.1.2 on page 91 and Section 6.3.1 on page 64).

- The frame start trigger goes low to end the exposure for frame acquisition N+1 before readout for acquisition N has ended (premature exposure end; see Fig. 26 on page 90).

  This would result in the illegal overlap of two readouts (in trigger width exposure mode only).

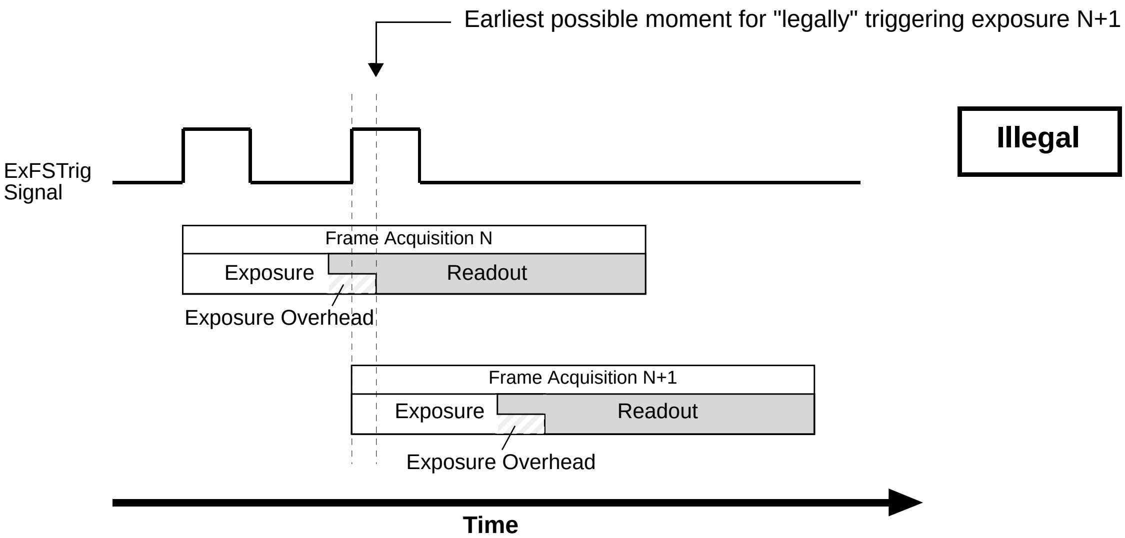


Fig. 25: Exposure N+1 Illegally Starts Before the Exposure Overhead for the Preceding Frame Acquisition N Has Ended; the Shown Overlap of Readouts is Also Illegal; Timed Exposure Mode Used as an Example

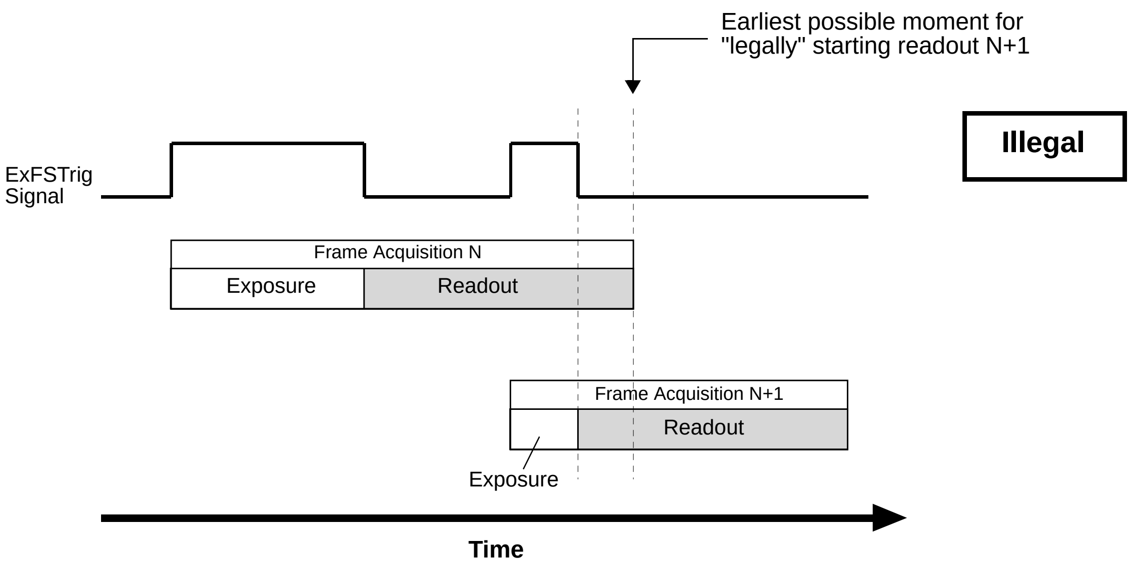


Fig. 26: Exposure N+1 Illegally Ends Before Readout of the Preceding Frame Acquisition N Has Ended; Applies to Trigger Width Exposure Mode Only

When the frame start trigger has illegally gone low to end the exposure for frame acquisition N+1 before readout for acquisition N has ended (in trigger width exposure mode; see Fig. 26), the camera will behave as shown in Fig. 27: The camera will extend the exposure and end it when the next valid trigger for ending exposure occurs.

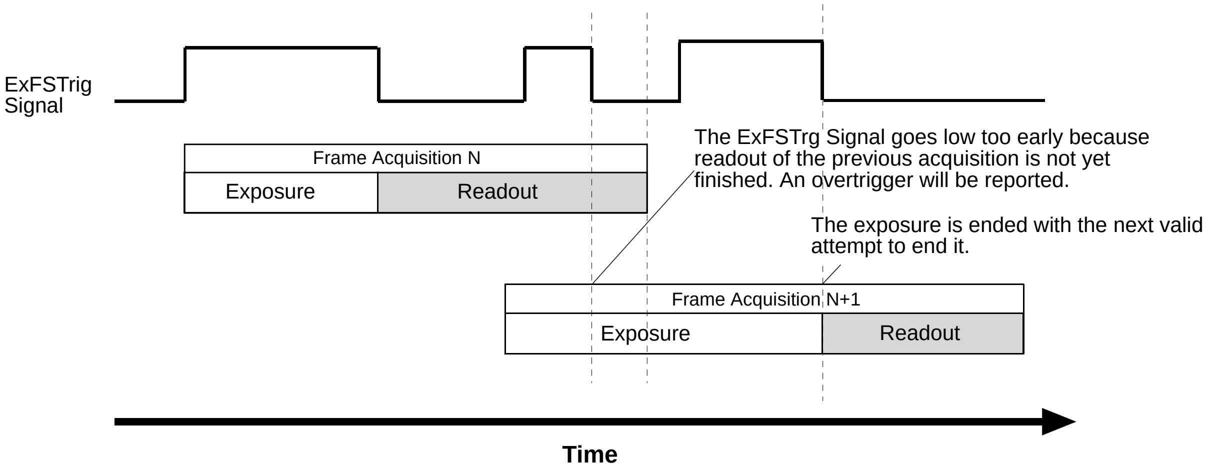


Fig. 27: Extension of Exposure N+1 After the Illegal Attempt of Ending It Too Early; Applies to Trigger Width Exposure Mode Only

6.5.1.2 Regular Frame Acquisition Avoiding Overtriggering

As mentioned above, you can avoid overtriggering by applying an acquisition-related trigger **only** when the camera is waiting for it.

You can achieve this goal by

- making use of acquisition monitoring tools, i.e. monitoring the camera's acquisition status and triggering only when the camera indicates that it is in the "waiting status" for the trigger or by
- strictly obeying timing limits for exposure and triggering.

Using Acquisition Monitoring Tools

To get informed whether the camera is waiting for a trigger you can use the acquisition monitoring tools described in Section 6.6 on page 92.

By applying an ExFSTrig signal as soon as the camera indicates that it is waiting for an ExFSTrig signal you can operate the camera in "overlapped mode" without overtriggering.

As an example, and in the context of overlapped exposure, the use of the frame trigger wait signal is described in Section 6.6.2.2 on page 96 for proper triggering with the frame start trigger. Both timed and trigger width exposure mode are considered.

6.6 Acquisition Monitoring Tools

The camera includes the acquisition status feature and generates these acquisition status and output signals that you can use to monitor the progress of frame acquisition by the camera:

- acquisition status:
  - frame trigger wait status
- output signals:
  - exposure active signal
  - acquisition trigger wait signal
  - frame trigger wait signal.
  - frame cycle signal

You can select one of the output signals as the source signal for the camera's CL Spare bit.

Note that the CL Spare bit will not be available if the camera is set to operate with 1X8-1Y or the 1X10-1Y tap geometry,

Therefore, in these cases, no output signal will be available.

6.6.1 Acquisition Status

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.

The camera's acquisition status indicator gives you the ability to check whether the camera is in a "waiting for frame start trigger" acquisition status. If you check the acquisition status before you apply 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 frame start trigger signals.

Determining the Acquisition Status Using Basler pylon

To determine the acquisition status for the frame start trigger:

1. Use the Acquisition Status Selector to select the Frame Trigger Wait status.
2. 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 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 3.1 on page 23.

Determining the Acquisition Status Using Direct Register Access

To determine the acquisition status for the frame start trigger:

1. 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 direct register access, see Section 3.2 on page 25.

6.6.2 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 overtrigger error.

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 an overtrigger error.

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 9.8 on page 145.

6.6.2.1 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 overtrigger error will be reported.

Fig. 28 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.

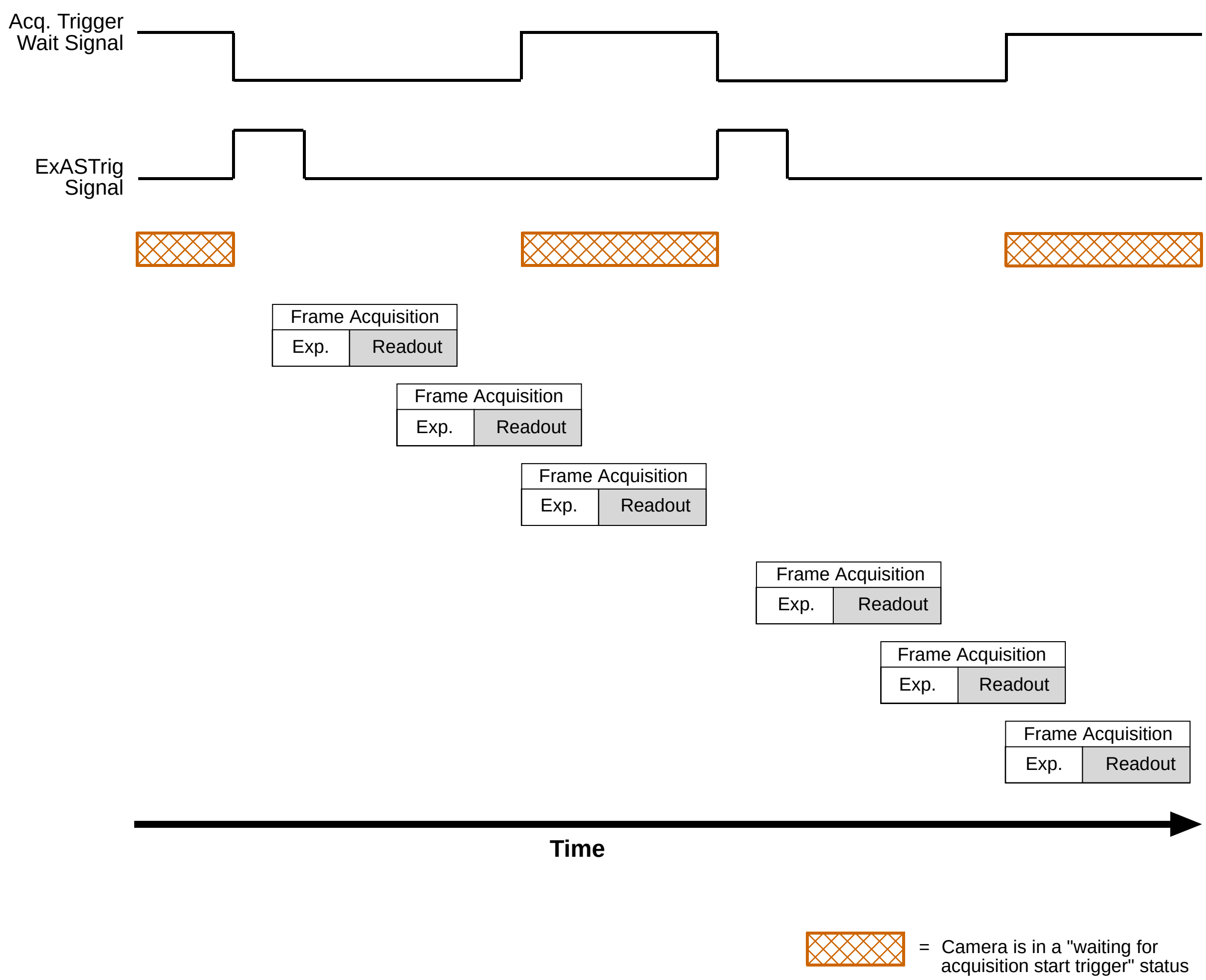


Fig. 28: Acquisition Trigger Wait Signal

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 9.8 on page 145.

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

To select the acquisition trigger wait signal as the source for the CL Spare output line:

1. Use the Line Selector to select the CL Spare line.
2. 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:

```
//Select the CL Spare line
Camera.LineSelector.SetValue( LineSelector_ClSpare );
//Set the source for the selected line
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 Chapter 3 on page 23.

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

To select the acquisition trigger wait signal as the source for the CL Spare output line:

1. Set the value of the Line Source CL Spare register to Acquisition Trigger Wait.

For more information about direct register access, see Chapter 3.2 on page 25.

6.6.2.2 Frame Trigger Wait Signal

As you are acquiring frame, 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.

You can select the frame trigger wait signal as the source signal for the CL spare bit on the camera.

If the camera is set to operate in the 1X8-1Y or the1X10-1Y tap geometry, the CL Spare bit will not be available.

For more information about selecting the source signal for the CL spare bit on the camera, see Section 5.5.2 on page 44.

Fig. 29 and Fig. 30 illustrate the Frame Trigger Wait signal with exposure and readout overlapped. The figures assume raising edge triggering.

Using the Frame Trigger Wait Signal with the Timed Exposure Mode

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 setting and on when readout of the current frame will end. This functionality is illustrated in Fig. 29.

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.

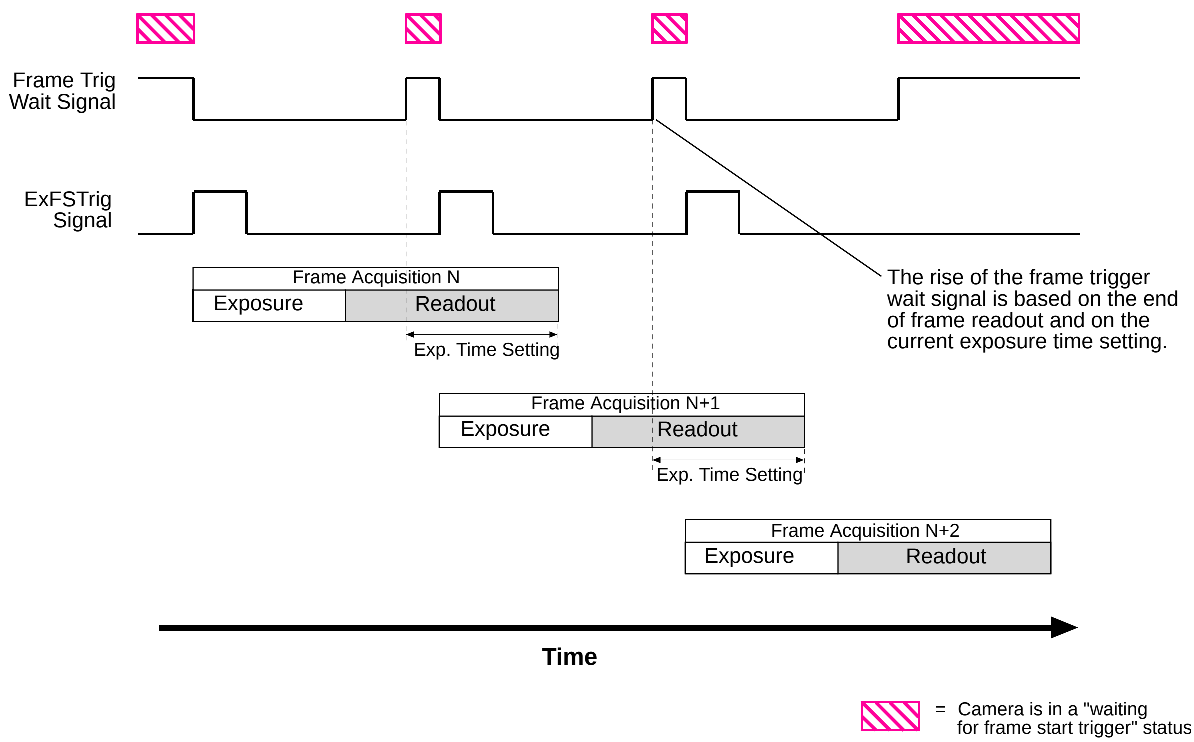


Fig. 29: Frame Trigger Wait Signal with the Timed Exposure Mode

Using the Frame Trigger Wait Signal with the Trigger Width 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 Overlap Time Max Abs parameter setting and on when readout of the current frame will end. This functionality is illustrated in Fig. 30.

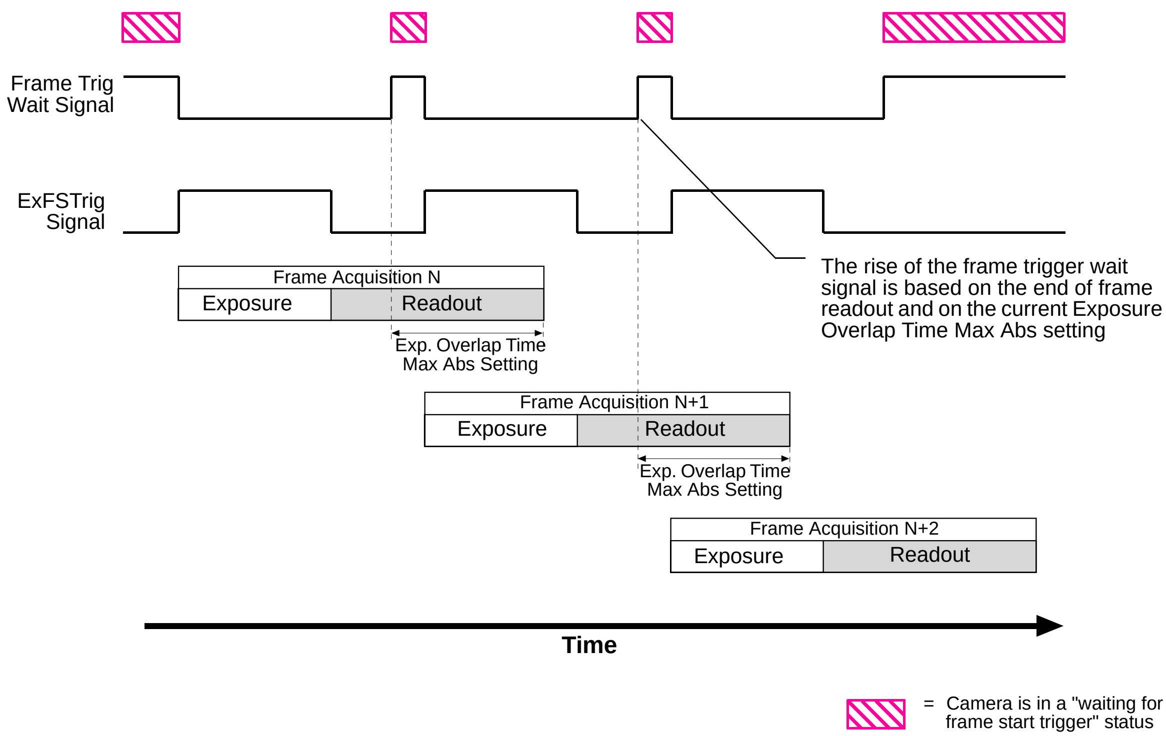


Fig. 30: Frame Trigger Wait Signal with the Trigger Width Exposure Mode

If you are operating the camera in the trigger width exposure mode, you must set an Exposure Overlap Time Max value. You can do this in either of two ways: If you are parameterizing the camera with Basler pylon you must set the Exposure Overlap Time Max Abs parameter and if you are parameterizing the camera via direct register access you must set the value of the Exposure Overlap Time Max register.

The correct parameter values depend on the method that you choose to avoid overtriggering:

You can avoid overtriggering the camera by always doing the following:

- Monitoring the camera's frame trigger wait signal and using the ExFSTrig signal to start exposure only when the frame trigger wait signal is high. You must set Exposure Overlap Time Max 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 15 µs to 20 µs. In this case you would set the camera's Exposure Overlap Time Max to 15 µs.
- Obeying certain timing limits during frame acquisition and setting the camera's Exposure Overlap Time Max to an appropriate value (see Section 6.5.1.2 on page 91

You can set an Exposure Overlap Time Max value in two ways: If you are parameterizing the camera with Basler pylon you must set the Exposure Overlap Time Max Abs parameter and if you are parameterizing the camera via direct register access you must set the value of the Exposure Overlap Time Max register.

For more information about avoiding overtriggering, see Section 6.5.1.2 on page 91.

Setting the Exposure Overlap Time Max Abs Parameter Using Basler pylon

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

```
// Set the Exposure Overlap Time Max to 15 µs
Camera.ExposureOverlapTimeMaxAbs.SetValue( 15);
```

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

Setting the Exposure Overlap Time Max Abs Parameter Using Direct Register Access

To set the Exposure Overlap Time Max Raw parameter:

1. Set the value of the Exposure Overlap Time Max Raw register.

For more information about direct register access, see Section 3.2 on page 25.

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

To select the frame trigger wait signal as the source for the CL Spare output line:

1. Select the frame trigger wait signal as the source signal for the camera's CL Spare output line: Use the Line Selector to select the CL Spare output line.
2. 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:

```
//Select the CL Spare line
Camera.LineSelector.SetValue( LineSelector_ClSpare );
//Set the source for the selected line
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 3.1.1 on page 24.

For more information about selecting the source signal for the CL spare bit on the camera, see Section 5.5.2 on page 44.

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

To select the frame trigger wait signal as the source for the CL Spare output line:

1. Set the value of the Line Source CL Spare register to Frame Trigger Wait.

For more information about direct register access, see Section 3.2 on page 25.

6.6.3 Exposure Active Signal

The camera's Exposure Active output signal will go high when the exposure time for each frame acquisition begins and goes low when the exposure time ends. The signal can for example be used as a flash trigger.

An example of the Exposure Active signal's behavior on a camera using a rising edge external frame start trigger signal (ExFSTrig) and the timed exposure mode is shown in Fig. 31.

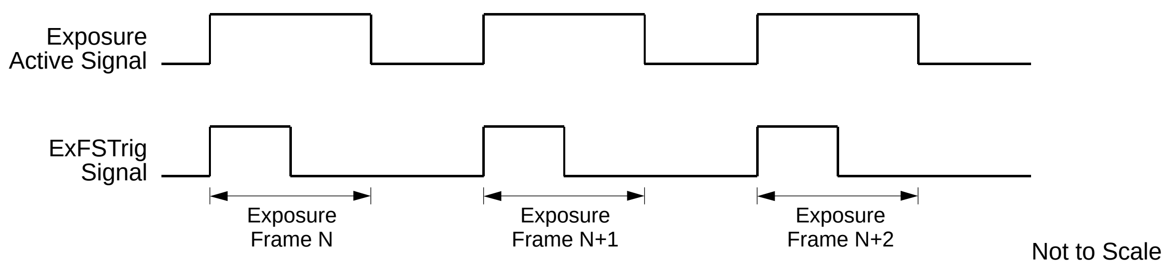


Fig. 31: Exposure Active Signal

Selecting the Exposure Active Signal as the Source Signal for the CLSpare Output Line Using Basler pylon

To select the exposure active signal as the source for the CL Spare output line:

1. Use the Line Selector to select the CLSpare line.
2. Set the value of the Line Source Parameter to the exposure active output 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:

```
//Select the CL Spare line
Camera.LineSelector.SetValue( LineSelector_ClSpare );
//Set the source signal for the selected line
Camera.LineSource.SetValue( LineSource_ExposureActive );
```

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

Selecting the Exposure Active Signal as the Source Signal for the CLSpare Output Line Using Direct Register Access

To select the exposure active signal as the source for the CL Spare output line:

1. Set the value of the Line Source CL Spare register to Exposure Active.

For more information about direct register access, see Section 3.2 on page 25.

6.6.4 Frame Cycle Signal

The camera can provide a "Frame Cycle" (FrmCyc) output signal. The signal goes high when the camera enters a waiting for frame trigger condition and goes low when the exposure time for the next triggered image ends as shown in Fig. 32. (In this example, the camera is operating in the timed exposure mode.)

The intention of this signal is to let you monitor these two important points in the acquisition process with a single output signal

.

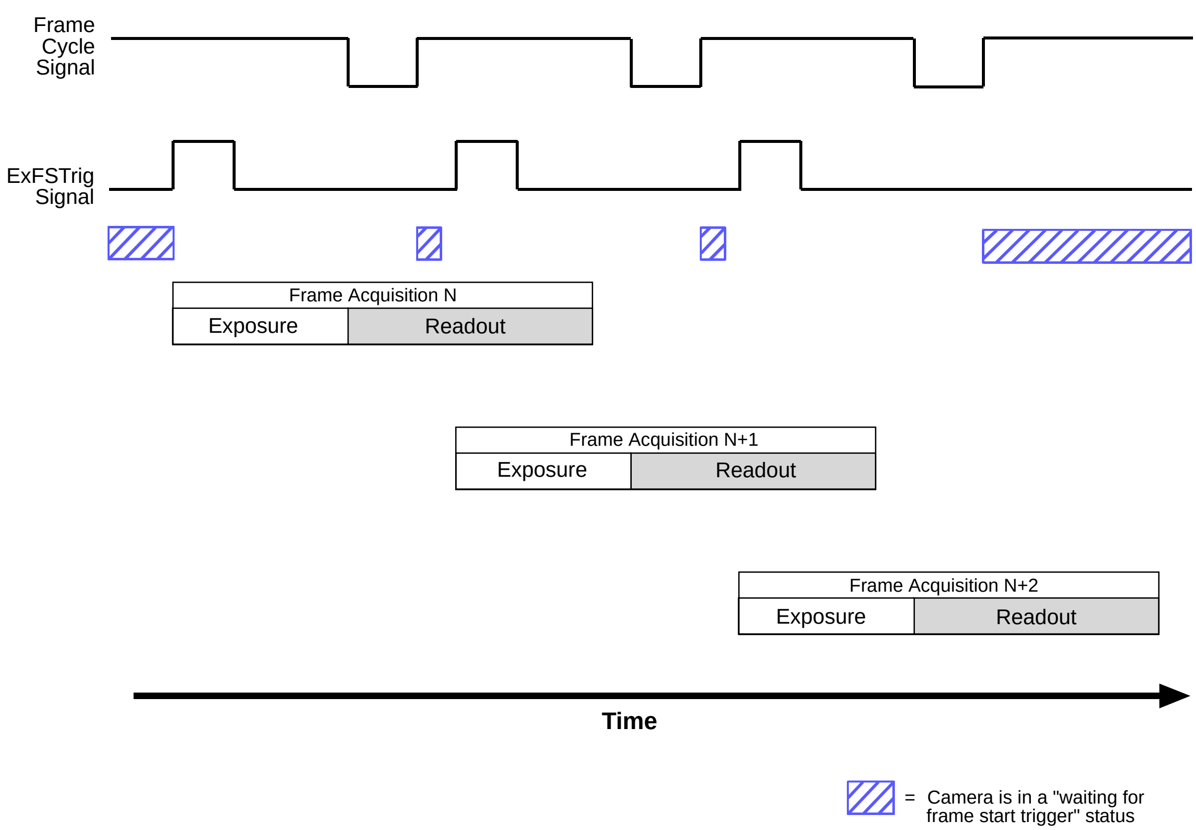


Fig. 32: Frame Cycle Signal

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

To select the frame cycle signal as the source for the CL Spare output line:

1. Use the Line Selector to select the CLSpare line.
2. Set the value of the Line Source parameter to the frame cycle output 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:

```
//Select the CL Spare line
Camera.LineSelector.SetValue( LineSelector_ClSpare );
//Set the source signal for the selected line
Camera.LineSource.SetValue( LineSource_FrameCycle );
```

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 Chapter 3 on page 23.

Selecting the Frame Cycle Signal as the Source Signal for the Output Line Using Direct Register Access

To select the frame cycle signal as the source for the CL Spare output line:

1. Set the value of the Line Source CL Spare register to Frame Cycle.

For more information about direct register access, see Section 3.2 on page 25.

6.7 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 Camera Link pixel clock speed and the Camera Link tap geometry settings.

  When the pixel clock speed is set to a high value, it will take less time to transfer captured images from the camera to the frame grabber in your host PC.

  When the camera is set for a tap geometry that uses more taps (e.g., the 1X3-1Y geometry uses three taps and the 1X8-1Y geometry uses eight taps), it can typically transfer data out of the camera faster.

  So if the camera is set for a high pixel clock speed and a high number of taps, it typically will have a much higher maximum allowed frame rate than when it is set for a low pixel clock speed and a low number of taps.
- The height of the current image AOI: A decreased height will increase the maximum allowed acquisition frame rate.
- Whether stacked zone imaging is enabled. Enabling stacked zone imaging may yield a significant increase in the maximum allowed frame rate, but this will only be true if the camera is not already reading out image data at a rate that is near to the current data carrying capacity of the Camera Link interface.
- The exposure time for acquired frames. If you use very long exposure times, you can acquire fewer frames per second.
- Whether the debouncer is enabled. Enabling the debouncer may result in a decreased maximum allowed frame rate.

> When the camera's acquisition mode is set to single frame, the maximum possible acquisition frame rate for a given AOI cannot be achieved. This results because the camera performs a complete internal setup cycle for each single frame and because it cannot be operated with overlapping sensor readout and exposure ("overlapped acquisition").
>
> To achieve the maximum possible acquisition frame rate, set the camera for the continuous acquisition mode and use overlapped acquisition.
>
> For more information about overlapped acquisition, see Section 6.5 on page 87.

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 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 Raw register (see below).

For more information about selectable Camera Link pixel clock speeds, see Section 9.1 on page 121.

For more information about Camera Link tap geometries, see Section 8.2 on page 119.

Checking the Maximum Allowed Frame Rate Using Basler pylon

You can use the Basler pylon API to read the current value of the Resulting Frame Rate Abs parameter from within your application software using the Basler pylon API. The following code snippet illustrates using the 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.

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

For more information about the pylon API and the pylon Viewer, see Section 3.1 on page 23.

Checking the Maximum Allowed Rate Using Direct Register Access

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:

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

To determine the resulting frame period:

1. 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 beat 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 10 ms (10000 ×1 µs = 10 ms).

For more information about direct register access, see Section 3.2 on page 25.

7 Color Creation and Enhancement

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

7.1 Color Creation

The sensor used in the cameras is equipped with an additive color separation filter known as a Bayer filter. The pixel formats available for output 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. The pattern of the Bayer filter used on the camera is as shown in Figure 33 (the figure shows the "RG" filter alignment as an example, with a "red" pixel as the first pixel in the first row and a "green" pixel as the second pixel in the first row). 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.)

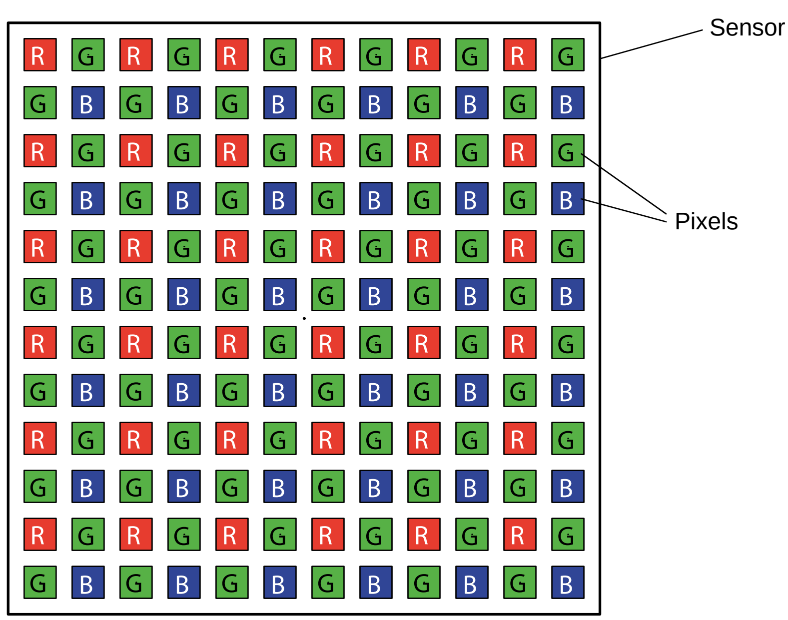


Fig. 33: Bayer Filter Pattern With RG Alignment (Example)

7.1.1 Bayer Color Filter Alignment

On all color camera models, the alignment of the filter to the pixels in the acquired images is Bayer GB. Bayer GB alignment means that pixel one and pixel two of the first line in each image transmitted will be green and blue respectively. And for the second line transmitted, pixel one and pixel two will be red 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.

The Pixel Color Filter parameter indicates the current alignment of the camera's Bayer filter to the pixels in the images captured by a color camera. You can tell how the current AOI is aligned to the Bayer filter by reading the value of the Pixel Color Filter parameter.

The image area of interest (AOI) parameters can only be changed in certain defined increments. The increments are designed so that the alignment of the Bayer filter to the pixels in the transmitted images will stay the same regardless of the position and size of the image 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 GB. The camera includes a mechanism that keeps the filter alignment constant when these features are used.

For more information about the camera's AOI feature, see Section 9.4 on page 126.

For more information about the reverse X and reverse Y features, see Section 9.6 on page 135.

7.1.2 Pixel Data Formats Available on Color Cameras

Bayer Formats

Cameras equipped with a Bayer pattern color filter can output pixel data in the Bayer GB 8, the Bayer GB 10, or the Bayer GB 12 pixel data format. When a color camera is set for one of these three pixel data output formats, the pixel data is not processed or interpolated in any way. So, for each pixel covered with a red lens, you get 8, 10, or 12 bits of red data. For each pixel covered with a green lens, you get 8, 10, or 12 bits of green data. And for each pixel covered with a blue lens, you get 8, 10, or 12 bits of blue data. (This type of pixel data is sometimes referred to as "raw" output.)

7.2 Color Enhancement Features

7.2.1 White Balance

White balance capability has been implemented on color models of the camera. White balancing can be used to adjust the color balance of the images transmitted from the camera.

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 value can range from 0.00 to 15.984. 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.

Setting the White Balance Using Basler pylon

To set the Balance Ratio parameter for a color:

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

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

```
// Set the red balance ratio
Camera.BalanceRatioSelector.SetValue(BalanceRatioSelector_Red);
Camera.BalanceRatioAbs.SetValue(1.20);

// Set the green balance ratio
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 3.1 on page 23.

Setting the White Balance Using Direct Register Access

To set the balance ratio parameters:

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

For more information about direct register access, see Section 3.2 on page 25.

7.2.2 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, 1023 for 10 bit output, and 4095 for 12 bit output.

The gamma correction value can be set in a range from 0 to 3.99998. For information on setting the gamma correction using hexadecimal instead of decimal figures, see "Setting Gamma Correction Using Hexadecimal Figures" on page 112.

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, 1023 at 10 bit output, and 4095 at 12 bit output) will not be corrected.

Enabling and Setting Gamma Correction Using Basler pylon

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 3.1 on page 23.

Enabling and Setting Gamma Correction Using Direct Register Access

To enable gamma correction and to set the gamma value:

1. Set the value of the Gamma Enable register to Enabled.
2. Set the value of the Gamma Selector register to sRGB or User.
3. If the Gamma Selector is set to User, set the value in the Gamma register to the desired gamma value.

For more information about direct register access, see Section 3.2 on page 25.

Setting Gamma Correction Using Hexadecimal Figures

If you want to change the gamma correction value using hexadecimal figures, a conversion factor is required. A gamma correction value of 1 (neutral gamma) corresponds to 65536 in the decimal system. If you want to set a gamma correction value other than 1, multiply the desired value by 65536 and then convert the result to the corresponding hexadecimal figure.

To set a gamma correction value of 0.42:

1. Perform the following calculation:
   0.42 x 65536 = 27525
2. Convert 27525 to the corresponding hexadecimal figure.
   This results in the number 0x6B85.

7.2.3 Matrix Color Transformation

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 any imperfections in the sensor's color generation process.

With the matrix color transformation, a first matrix transformation step ensures that the pixel values form 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.

The first camera parameter associated with matrix color transformation is the Color Transformation Selector parameter. This parameter is used to select the type of transformation that will be performed. For Basler beat Camera Link cameras, RGB to RGB is the only setting available. This setting simply means that the camera will not transform the red, green, and blue pixel values from the sensor into a different color space (such as YUV).

The second parameter associated with matrix color transformation is the Light Source Selector parameter. The following settings are available for this parameter:

- Off - No alterations will be made to the pixel values.
- 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.
- 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.
- 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.
- Custom - The user can set the values in the matrix as desired.

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.

Setting Matrix Color Transformation Using Basler pylon

You can set the Color Transformation Selector and Light Source Selector parameters from within your application software by using the Basler pylon API. The following code snippet illustrates using the API to set the parameter values:

```
// Select the color transformation type
Camera.ColorTransformationSelector.SetValue(ColorTransformationSelector_RGBtoRGB
);
// Set the light source selector so that no correction will be done
Camera.LightSourceSelector.SetValueLightSourceSelector_Off);
// Set the light source selector for daylight (at about 5000K)
Camera.LightSourceSelector.SetValueLightSourceSelector_Daylight);
// Set the light source selector for tungsten lighting
Camera.LightSourceSelector.SetValueLightSourceSelector_Tungsten);
// Set the light source selector for daylight (at about 6500K)
Camera.LightSourceSelector.SetValueLightSourceSelector_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 3.1 on page 23.

Setting Matrix Color Transformation Using Direct Register Access

To set matrix color transformation:

1. Set the value of the Light Source Selector register to Off, Daylight, Tungsten, Daylight 6500K, or Custom as desired.
2. Set the value of the Color Matrix Factor register as desired.

For more information about direct register access, see Section 3.2 on page 25.

7.2.3.1 "Custom" Light Source Setting

ⓘ	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.

Keep in mind that Basler beat Camera Link cameras use sensors with a Bayer pattern filter and that they do not interpolate the pixel values. This means that when the camera is a doing matrix correction for a particular pixel, only one actual color value will be available for that pixel. To fill in the other two color values in the 1 x 3 matrix, the camera uses values from neighboring pixels. After making the correction calculations, the camera only transmits the result for the actual pixel color and discards the other two calculated values. For example, if the camera is correcting the value for a red pixel, it will populate the 1 x 3 matrix with the actual value for the red pixel. It will then populate the green and blue positions in the matrix with values from a neighboring green and a neighboring blue pixel. After making the matrix correction calculations, the camera will discard the green and blue values from the result and only transmit the red value.

Each GainXY position can each 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 GainXX 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/pub/ntia-rpt/04-406/index.php

Setting Custom Matrix Values Using Basler pylon

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.SetValueColorTransformationValueSelector
_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.SetValueColorTransformationValueSelector
_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.

For more information about the pylon API and the pylon Viewer, see Section 3.1 on page 23.

Setting Custom Matrix Values Using Direct Register Access

To set custom matrix values:

To set Gain XY values in the matrix via direct register access:

1. Set the value of the Light Source Selector register to Custom.
2. Set the value of the following registers as desired:

   Color Matrix RGB 2 RGB 00

   Color Matrix RGB 2 RGB 01

   Color Matrix RGB 2 RGB 02

   Color Matrix RGB 2 RGB 10

   Color Matrix RGB 2 RGB 11

   Color Matrix RGB 2 RGB 12

   Color Matrix RGB 2 RGB 20

   Color Matrix RGB 2 RGB 21

   Color Matrix RGB 2 RGB 22

For more information about direct register access, see Section 3.2 on page 25.

8 Pixel Formats and Tap Geometries

This chapter provides information about the sensor bit depths and pixel formats available on the camera.

For a description of tap geometries and complete details regarding the way that pixel data is handled by the camera, refer to the Basler document named *Basler ace and beat Camera Link Information for Frame Grabber Designers* (AW000990), that is specific for Basler ace and Basler beat Camera Link cameras. You can obtain the document from the camera's Downloads section of the Basler website: www.baslerweb.com

8.1 Pixel Formats

Pixel Formats for Monochrome and Color Cameras

The choice of a pixel format determines the bit depth of the data transmitted from the camera for each pixel in the acquired frames. The availability of pixel formats is subject to the currently selected Camera Link tap geometry as shown in Table 8.

For information about Camera Link tap geometries, see Section 8.2 on page 119.

Camera Link Tap Geometry	Available Pixel Formats					
	Mono Cameras			Color Cameras		
	Mono 8	Mono 10	Mono 12	Bayer GB 8	Bayer GB 10	Bayer GB 12
1X2-1Y	●	●	●	●	●	●
1X3-1Y	●			●		
1X8-1Y	●	●		●	●	
1X10-1Y	●			●		

Table 8: Selected Camera Link Tap Geometry and Available Pixel Formats (● = format available)

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 API to set the parameter value:

```
// Set pixel format to Mono 8 for a monochrome camera
Camera.PixelFormat.SetValue( PixelFormat_Mono8 );
// Set pixel format to Mono 10 for a monochrome camera
Camera.PixelFormat.SetValue( PixelFormat_Mono10 );
// Set pixel format to Mono 12 for a monochrome camera
Camera.PixelFormat.SetValue( PixelFormat_Mono12 );
// Set pixel format to Bayer RG 8 for a color camera
Camera.PixelFormat.SetValue( PixelFormat_BayerGB8 );
// Set pixel format to Bayer RG 10 for a color camera
Camera.PixelFormat.SetValue( PixelFormat_BayerGB10 );
// Set pixel format to Bayer RG 12 for a color camera
Camera.PixelFormat.SetValue( PixelFormat_BayerGB12 );
```

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 3.1 on page 23.

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, Mono 10, Mono 12, Bayer GB 8, Bayer GB 10 or Bayer GB 12 as desired.

For more information about direct register access, see Section 3.2 on page 25.

8.2 Camera Link Tap Geometry

The Camera Link tap geometry determines how the data that is read out of the imaging sensor will be transmitted from the camera to the frame grabber in your host PC via the Camera Link interface. The selection of a camera link tap geometry also determines whether your camera will be operating in the base, medium, or full Camera Link configuration and which Camera Link connectors on the camera will be used to transmit pixel data. Table 9 indicates how the Camera Link interface will operate with each available tap geometry setting.

The e.g. X2 or X8 in the tap geometry names indicates the number of Camera Link taps that will be used for a given configuration (e.g., X2 means 2 taps). As a general rule of thumb, a camera will have a higher maximum allowed frame rate when it is operating with a tap geometry that uses more Camera Link taps.

Tap Geometry Setting	Camera Link Configuration	Camera Link Connectors Used to Transmit Data
1X2-1Y	Base	Base Only
1X3-1Y	Base	Base Only
1X8-1Y	Full	Base and Medium/Full
1X10-1Y	Full	Base and Medium/Full

Table 9: Camera Link Operation at Various Tap Geometry Settings

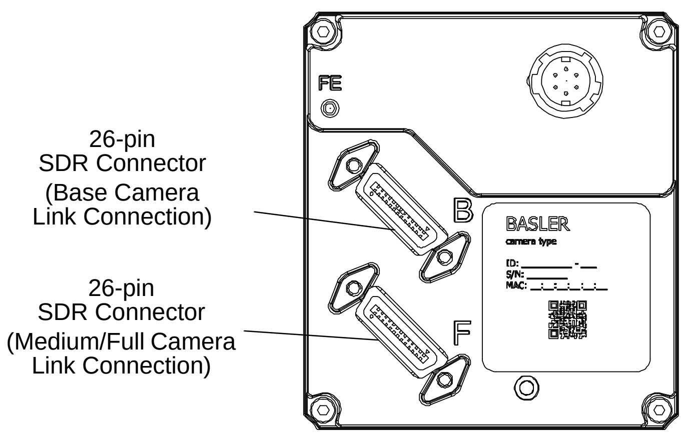


Fig. 34: Camera Link Connections

For a description of tap geometries refer to the Basler document named *Basler ace and beat Camera Link Information for Frame Grabber Designers* (AW000990), that is specific for Basler ace and beat Camera Link cameras. You can obtain the document from the camera’s Downloads section of the Basler website: www.baslerweb.com

8.2.1 Setting the Tap Geometry

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 API to set the tap geometry:

```
// Set the tap geometry to 1X2-1Y
Camera.ClTapGeometry.SetValue( ClTapGeometry_Geometry1X2_1Y );

// Set the tap geometry to 1X3-1Y
Camera.ClTapGeometry.SetValue( ClTapGeometry_Geometry1X3_1Y );

// Set the tap geometry to 1X8-1Y
Camera.ClTapGeometry.SetValue( ClTapGeometry_Geometry1X8_1Y );

// Set the tap geometry to 1X10-1Y
Camera.ClTapGeometry.SetValue( ClTapGeometry_Geometry1X10_1Y );
```

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 3.1 on page 23.

Setting the Tap Geometry Using Direct Register Access

To set the Camera Link tap geometry via direct register access:

- Set the value of the CL Tap Geometry register for 1X2-1Y, 1X3-1Y, 1X8-1Y or 1X10-1Y as desired.

For more information about direct register access, see Section 3.2 on page 25.

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 Camera Link Pixel Clock Speed

The camera provides 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 PC via the Camera Link interface. The following pixel clock speeds are available:

Pixel Clock Speed
32.5 MHz
65 MHz
84.0 MHz

Table 10: Available Camera Link Pixel Clock Speeds

The rate at which image data is transferred from the camera to the frame grabber depends on the set pixel clock speed. Setting the camera for a higher pixel clock speed will increase the transfer rate. 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 pixel clock 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 being transmitted 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 e.g. 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 line acquisition rate, see Section 6.7 on page 103.

For more information about configuration sets, see Section 9.12 on page 155.

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 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 65
Camera.ClPixelClock.SetValue( ClPixelClock_PixelClock65 );
```

ⓘ	You can use the pylon API to set the pixel clock speed to 32.5, 65, or 84.0 MHz. These are the only valid values for the pixel clock speed. If you attempt to use the 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 3.1 on page 25.

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 32.5, 65, or 84.0 MHz.

For more information about changing settings via direct register access, see Section 3.2 on page 25.

9.2 Gain

The camera's gain setting is adjustable. As shown in Figure 35, 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.

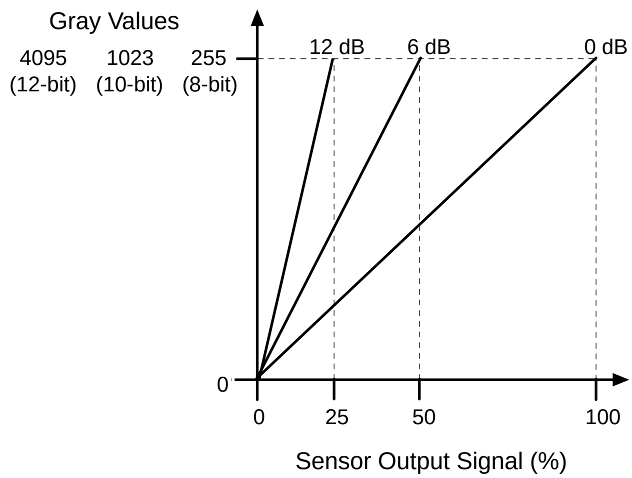


Fig. 35: Gain in dB

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.

The camera's gain is determined by the value of the Gain Raw parameter. Raw gain is adjusted on an integer scale. The minimum setting is 33 and the maximum setting is 511. If you know the current setting for the Gain Raw parameter, you can calculate the camera's gain in dB using the following formula:

$$\text{Gain in dB} = 20 \log_{10} (\text{Gain Raw Setting} / 32)$$

For example, if the current Gain Raw setting is 128, then:

$$\text{Gain in dB} = 20 \log_{10} (128 / 32)$$

$$\text{Gain in dB} = 12.0$$

Setting the Gain Using Basler pylon

To set the Gain Raw parameter value:

- Set the Gain Selector to All.
- 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 API to set the selector and the parameter value:

```
// Set the gain
Camera.GainSelector.SetValue( GainSelector_All );
Camera.GainRaw.SetValue( 128 );
```

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 3.1 on page 23.

Setting the Gain Using Direct Register Access

To set the gain via direct register access:

1. Set the value of the Gain All register.

For more information about direct register access, see Section 3.2 on page 25.

9.3 Black Level

Adjusting the camera's black level will result in an offset to the pixel values output by the camera. Increasing the black level setting will result in a positive offset in the digital values output for the pixels. Decreasing the black level setting will result in a negative offset in the digital values output for the pixels.

The black level can be set on an integer scale ranging from -2048 to 2047, regardless of the selected pixel format.

If the camera is set for a pixel format that yields an 8 bit pixel depth, an increase of 16 in the black level parameter setting will result in a positive offset of 1 in the digital values output for the pixels. And a decrease of 16 in the setting will result in a negative offset of 1 in the digital values output for the pixels.

If the camera is set for a pixel format that yields a 10 bit pixel depth, an increase of 4 in the black level parameter setting will result in a positive offset of 1 in the digital values output for the pixels. A decrease of 4 in the setting will result in a negative offset of 1 in the digital values output for the pixels.

If the camera is set for a pixel format that yields a 12 bit pixel depth, an increase of 1 in the black level parameter setting will result in a positive offset of 1 in the digital values output for the pixels. A decrease of 1 in the setting will result in a negative offset of 1 in the digital values output for the pixels.

Setting the Black Level Using Basler pylon

To set the black level:

1. Set the Black Level Selector to All.
2. 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 API to set the selector and the parameter value:

```
// Set the black level
Camera.BlackLevelSelector.SetValue ( BlackLevelSelector_All );
```

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 3.1 on page 23.

Setting the Black Level Using Direct Register Access

To set the black level via direct register access:

1. Set the value of the Black Level All register.

For more information about direct register access, see Section 3.2 on page 25.

9.4 Image Area of Interest (AOI)

ⓘ	When the image AOI feature is enabled, the stacked zone imaging feature will not be available. For more information about the stacked zone imaging feature, see Section 9.5 on page 129.

The image area of interest (Image AOI or AOI for short) feature lets you specify a portion of the sensor array and after each image is acquired, only the pixel information from the specified portion of the array will be read out of the sensor and transmitted from the camera.

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 36.

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 36.

The camera will only transmit pixel data from within the area defined by your settings. Information from the pixels outside of the area of interest is discarded.

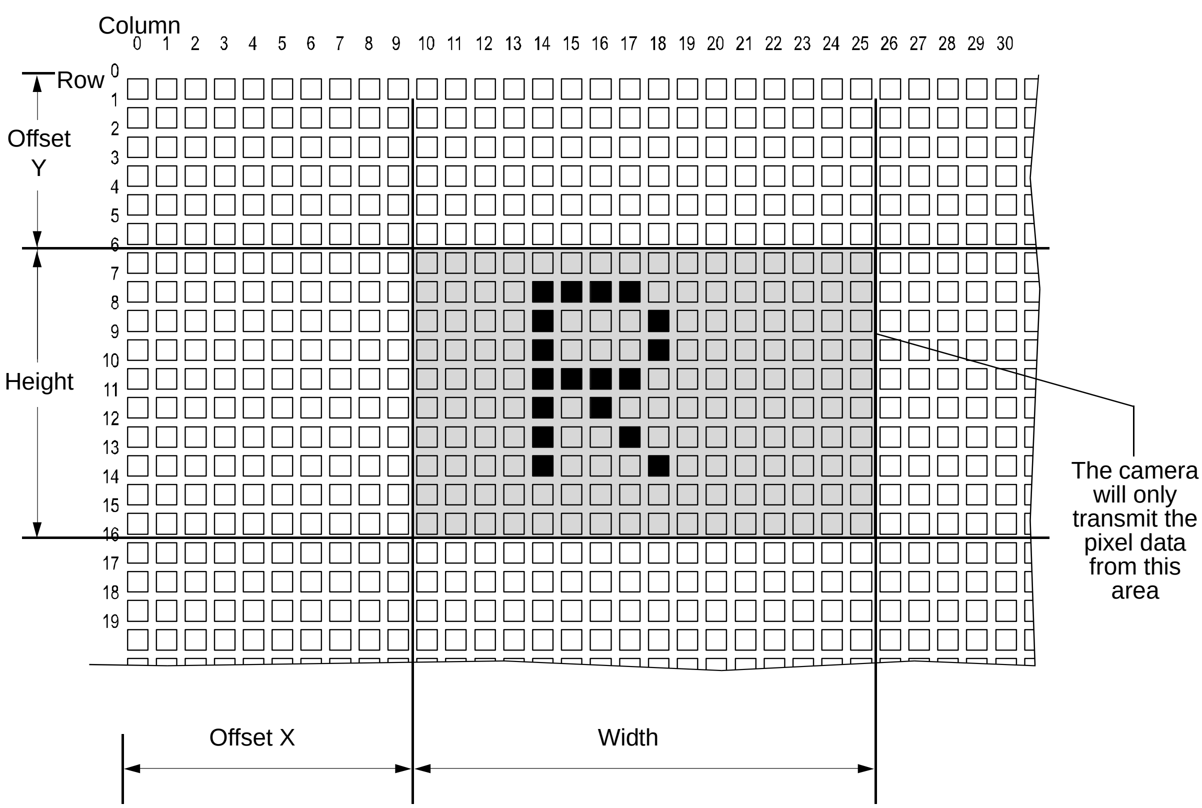


Fig. 36: Area of Interest

One of the main advantages of the AOI feature is that decreasing the size of the AOI can increase the camera's maximum allowed acquisition frame rate.

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).

For more information about how changing the AOI size affects the maximum allowed frame rate, see Section 6.7 on page 103.

9.4.1 Setting the Image AOI

By default, the AOI is set to use the full resolution of the camera's sensor. You can change the position and the size of the AOI by changing the value of the camera's Offset X, Offset Y, Width, and Height parameters. The allowed increments for the parameter vales are given in Table 11.

You can also enable automatic centering using the Center X and Center Y parameters.

When setting the camera's AOI, you must follow these guidelines:

General guidelines:

- The sum of the Offset X setting plus the Width setting must not exceed the width of the camera's sensor. For example, on the beA4000-62km, the sum of the Offset X setting plus the Width setting must not exceed 4096.
- The sum of the Offset Y setting plus the Height setting must not exceed the height of the camera's sensor. For example, on the beA4000-62km, the sum of the Offset Y setting plus the Height setting must not exceed 3072.

Operation-specific increments:

Parameter Name	Camera Link Tap Geometry Setting	Parameter Values for Mono Cameras	Parameter Values for Color Cameras
		Increments [Pixels]	Increments [Pixels]
Offset X	All	1	2
Offset Y	All	4	4
Width	1X2-1Y	8	8
	1X3-1Y	24	24
	1X8-1Y	8	8
	1X10-1Y	40	40
Height	All	4	4

Table 11: Increments [Pixels] for AOI Position and Size

For more information about Camera Link tap geometries, see Section 8.2 on page 119.

ⓘ	Your frame grabber may place additional restrictions on how the AOI size must be set. Check the documentation included with your frame grabber to determine its AOI requirements.

Setting the Image AOI Using Basler pylon

You can set the Offset X, Offset Y, Width, and Height parameter values from within your application software by using the Basler pylon API. The following code snippets illustrate using the API to get the maximum allowed settings and the increments for the Width and Height parameters. They also illustrate setting the Offset X, Offset Y, Width, and Height parameter values and enabling automatic AOI centering.

```
int64_t widthMax = Camera.Width.GetMax( );
int64_t widthInc = Camera.Width.GetInc();
Camera.Width.SetValue( 200 );
Camera.OffsetX.SetValue( 100 );

int64_t heightMax = Camera.Height.GetMax( );
int64_t heightInc = Camera.Height.GetInc();
Camera.Height.SetValue( 200 );
Camera.OffsetY.SetValue( 100 );

// Enable automatic X and Y centering
Camera.CenterX.SetValue( true );
Camera.CenterY.SetValue( true );
```

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 3.1 on page 23.

Setting the Image AOI Using Direct Register Access

To set the AOI Offset X, Offset Y, Width, and Height parameters via direct register access:

1. Set the value of the Offset X register.
2. Set the value of the Offset Y register.
3. Set the value of the Width register.
4. Set the value of the Height register.

To enable Center X and Center Y via direct register access:

1. Set the value of the Center X register.
2. Set the value of the Center Y register.

For more information about direct register access, see Section 3.2 on page 25.

9.5 Stacked Zone Imaging

ⓘ	When the stacked zone imaging feature is enabled, the image AOI feature will not be available. For more information about the image AOI feature, see Section 9.4 on page 126.

The stacked zone imaging feature lets you define up to eight areas on the sensor array, called "zones". All zones will be the same width and will be vertically aligned (see Figure 37). When an image is acquired, only the pixel information from the defined zones will be used for the image. The zones will be stacked together and will be transmitted from the camera as a single image.

You can initially set the positions and the sizes of the zones by setting the Offset X and Width parameters "globally" for all zones and then the Zone Offset Y and Zone Height parameters individually for each zone.

Since all of the zones will be the same width and vertically aligned, the Offset Y and Width parameters define the left and right borders for all of the zones (see Figure 37 on page 130). For the zones shown in In Figure 37, Offset X is set to 10 and Width is set to 16.

Details for configuring stacked zone imaging are given in Section 9.5.1.

As the image AOI feature will not be available when the stacked zone imaging is enabled, the following parameters will become read only:

- The Offset Y parameter becomes read only and will indicate the Y offset for the zone nearest to the top of the sensor.
- The Height parameter becomes read only and will indicate the total height of the image that will be transmitted from the camera (i.e., the sum of the heights of all zones where, however, possibly overlapping rows are only considered once).

Note that the Offset X and Width parameters are used for both, the image AOI feature and the stacked zone imaging feature.

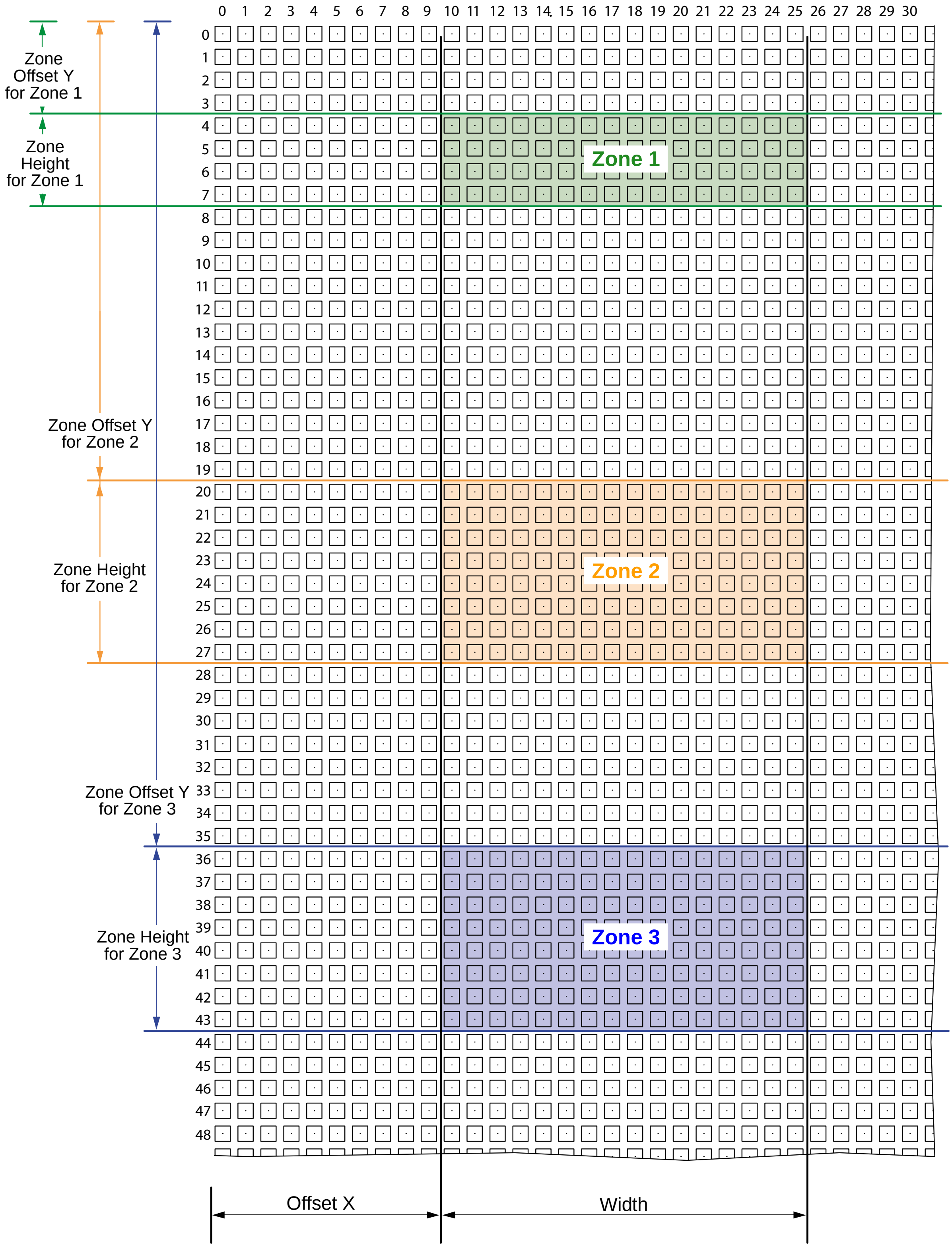


Fig. 37: Stacked Zone Imaging with Three Zones Defined as an Example

9.5.1 Setting Stacked Zone Imaging

As mentioned before, the Offset Y and Width parameters define the left and right borders for all of the zones. Therefore, all of the zones will be the same width and will be vertically aligned as shown in Figure 37 on page 130. In the figure, Offset X is set to 10 and the Width is set to 16.

Subsequently, each zone is individually enabled and defined. Up to 8 zones can be defined, with index numbers ranging from 1 through 8.

Once a zone has been enabled, you can use the Zone Offset Y parameter to set the offset (in pixels) between the top of the sensor and the top of the zone. And you can use the Zone Height parameter to set the height of the zone. The allowed increments for the parameters values are given in Table 11.

In Figure 37, for example, three zones have been enabled - zone 1, zone 2, and zone 3.

- The Offset X is set to 10 and the Width is set to 16. These settings apply to all zones.

For zone 1:

- The Stacked Zone Imaging Zone Offset Y parameter is set to 4.
- The Stacked Zone Imaging Zone Height parameter is set to 4.

For zone 2:

- The Stacked Zone Imaging Zone Offset Y parameter is set to 20.
- The Stacked Zone Imaging Zone Height parameter is set to 8.

For zone 3:

- The Stacked Zone Imaging Zone Offset Y parameter is set to 36.
- The Stacked Zone Imaging Zone Height parameter is set to 8.

With these settings, the camera would output an image that is 16 pixels wide and 20 rows high (the total height of the three zones).

When you are configuring stacked zones, you must follow these guidelines:

General guidelines and hints:

- When configuring stacked zone imaging you must first enable stacked zone imaging. The Stacked Zone Imaging Enable parameter is used to enable or disable stacked zone imaging. When the parameter is set to true, stacked zone imaging is enabled.
- Before configuring each zone individually you must set the Offset X and Width parameters. Their settings apply to all zones. Note that these parameters are also used for setting an AOI which, however, is not available when stacked zones imaging is used.
  - The sum of the Offset X setting plus the Width setting must not exceed the width of the camera's sensor. For example, on the beA4000-62km, the sum of the Offset X setting plus the Width setting must not exceed 4096.
- To configure each zone you must select the zone by its index number and then enable the zone by setting the Zone Enable parameter to true.
- You can then set the Zone Offset Y and Zone Height parameters for the selected zone.

- The sum of the Zone Offset Y setting plus the Zone Height setting must not exceed the height of the camera's sensor. For example, on the beA4000-62km, the sum of the Zone Offset Y setting plus the Zone Height setting must not exceed 3072.

- You are not required to enable the zones in numeric order. For example, you can enable zones 2, 4, and 6 and not enable zones 1, 3, and 5.
- You do not need to arrange the zones in numeric order from top to bottom on the sensor. For example, you could place zone 1 near the bottom of the sensor, zone 3 near the top, and zone 2 in the middle.

  But note that the camera always reads out and transmits the zones starting from the top of the sensor and going to the bottom, regardless of how the zone numbers are ordered. So the lines in the transmitted images will always be ordered from top to bottom in relation to the sensor.
- The zones can be set so that they overlap. When this happens, the camera will internally transform the overlapped zones into a single large zone that will be read out and transmitted as if it were simply a single large zone. (The lines included in the overlapping area will only be read out and transmitted once.)

Configuration-specific increments:

Parameter Name	Camera Link Tap Geometry Setting	Parameter Values for Mono Cameras	Parameter Values for Color Cameras
		Increments [Pixels]	Increments [Pixels]
Zone Offset X	All	2	2
Zone Offset Y	All	4	4
Zone Width	1X2-1Y	8	8
	1X3-1Y	24	24
	1X8-1Y	8	8
	1X10-1Y	40	40
Zone Height	All	4	4

Table 12: Increments [Pixels] for Zone Positions and Sizes

Setting Stacked Zone Imaging Using Basler pylon

You can set the parameter values associated with stacked zone imaging from within your application software by using the Basler pylon API. The following code snippets illustrate using the API to set up two zones.

```
// Enable stacked zone imaging
Camera.StackedZoneImagingEnable.SetValue( true );

// Set the width and offset X for the zones
Camera.Width.SetValue( 200 );
Camera.OffsetX.SetValue( 100 );

// Set zone 1

// Select the zone
Camera.StackedZoneImagingIndex.SetValue( 1 );
// Enable the selected zone
Camera.StackedZoneImagingZoneEnable.SetValue( true );
// Set the offset Y for the selected zone
Camera.StackedZoneImagingZoneOffsetY.SetValue( 100 );
// Set the height for the selected zone
Camera.StackedZoneImagingZoneHeight.SetValue( 100 );

// Set zone 2

// Select the zone
Camera.StackedZoneImagingIndex.SetValue( 2 );
// Enable the selected zone
Camera.StackedZoneImagingZoneEnable.SetValue( true );
// Set the offset Y for the selected zone
Camera.StackedZoneImagingZoneOffsetY.SetValue( 248 );
// Set the height for the selected zone
Camera.StackedZoneImagingZoneHeight.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 3.1 on page 23.

Setting Stacked Zone Imaging Using Direct Register Access

To enable stacked zone imaging via direct register access:

1. Set the value of the Stacked Zone Imaging Enable register to 1 (true).

To set the Offset X and Width parameters for the zones:

1. To set the Offset X and Width parameters for the zones:
2. Set the value of the Offset X register.
3. Set the value of the Width register.

To set up zone 1:

1. Enable zone 1 by setting the value of the Stacked Zone Imaging Zone 1 Enable register to 1 (true).
2. Set the Y offset for zone 1 by setting the value of the Stacked Zone Imaging Zone 1 Offset Y register.
3. Set the height for zone 1 by setting the value of the Stacked Zone Imaging Zone 1 Height register.

The other zones are set up in similar fashion.

For more information about direct register access, see Section 3.2 on page 25.

9.6 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.

9.6.1 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 38 shows a normal image on the left and an image captured with reverse X enabled on the right.

Normal Image **Reverse X Mirror Image**

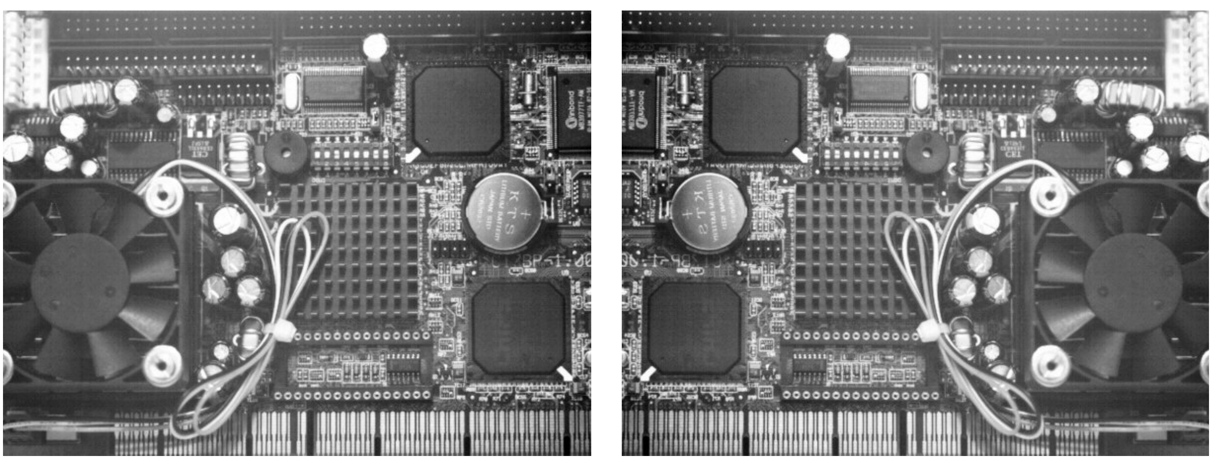


Fig. 38: Reverse X Mirror Imaging

Using AOIs with Reverse X

You can use the AOI feature when using the reverse X feature. Note, however, that the position of an AOI relative to the sensor remains the same regardless of whether or not the reverse X feature is enabled (see Figure 39).

As a consequence, an AOI will display different images depending on whether or not the reverse X feature is enabled.

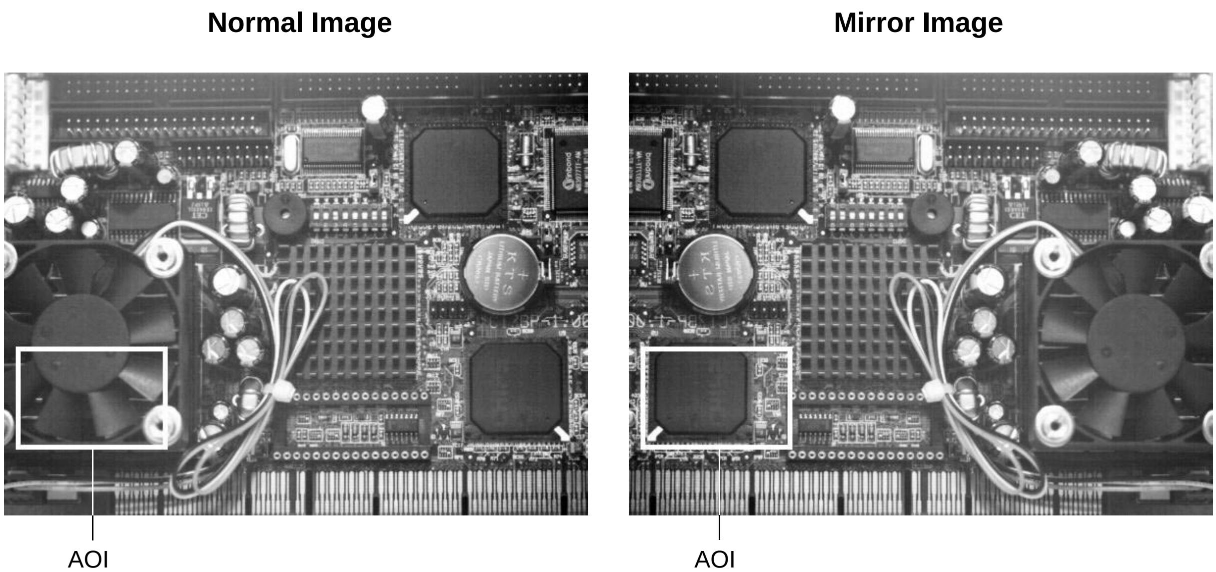


Fig. 39: Using an AOI with Reverse X Mirror Imaging

> For color cameras, provisions are made ensuring that the effective color filter alignment will be constant for normal and mirror images.

9.6.2 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 38 shows a normal image on the left and an image captured with reverse Y enabled on the right.

Normal Image **Reverse Y Mirror Image**

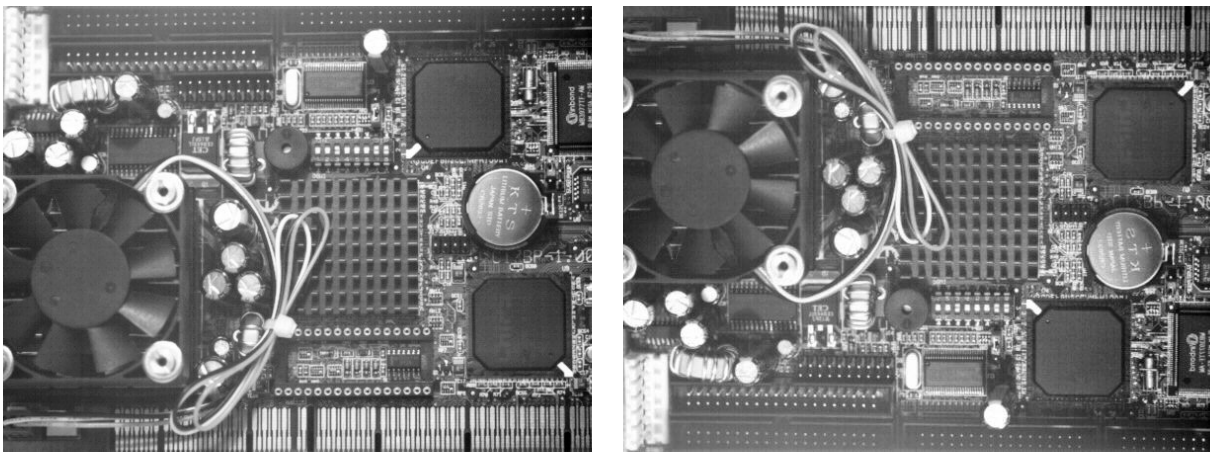


Fig. 40: Reverse Y Mirror Imaging

Using AOIs with Reverse Y

You can use the AOI feature when using the reverse X feature. Note, however, that the position of an AOI relative to the sensor remains the same regardless of whether or not the reverse Y feature is enabled (see Figure 41).

As a consequence, an image AOI will display different images depending on whether or not the reverse Y feature is enabled.

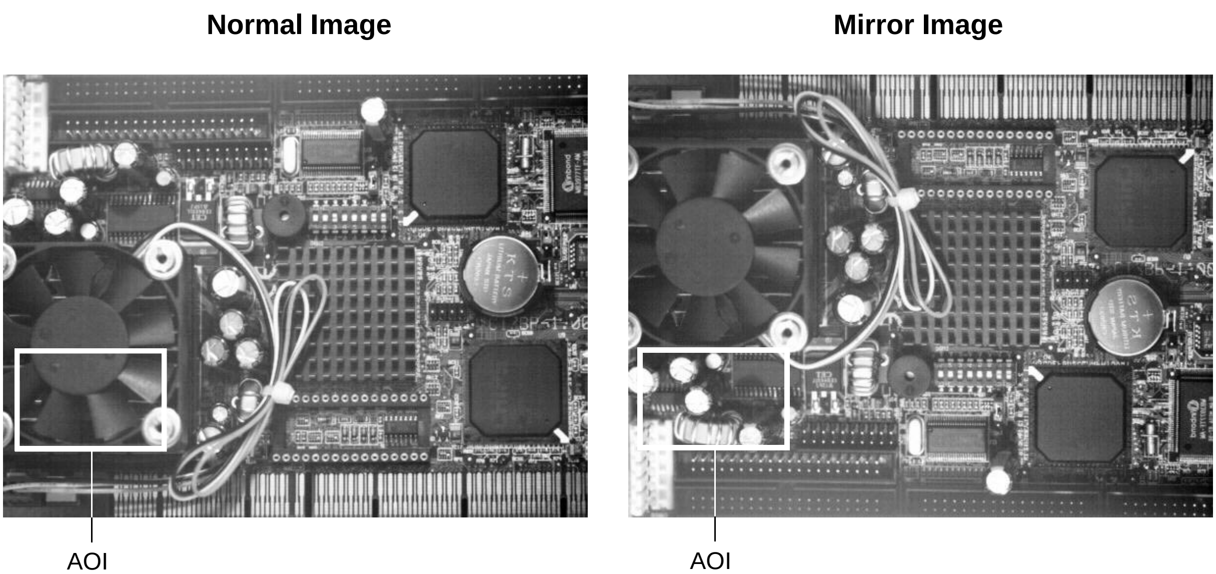


Fig. 41: Using an AOI with Reverse Y Mirror Imaging

ⓘ	For color cameras, provisions are made ensuring that the effective color filter alignment will be constant for normal and mirror images.

9.6.3 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 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 3.1 on page 23.

Enabling Reverse X and Y Using Direct Register Access

To enable reverse X via direct register access:

1. Set the value of the Reverse X register to 1 (enabled).

To enable reverse Y via direct register access:

1. Set the value of the Reverse Y register to 1 (enabled).

For more information about direct register access, see Section 3.2 on page 25.

9.7 Luminance Lookup Table (LUT)

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 data format, the camera transmits the actual 12 bit pixel values reported by the sensor.

The luminance lookup table (LUT) 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.

ⓘ	If the imaging sensor bit depth is set to 10 bits, the sensor will only capture pixel data at 10 bit depth. In this case, the pixel values output from the sensor wil be converted to 12 bit depth by padding the 10 bit values with two zeros as least significant bits. These converted 12 bit values will then be used as input to the lookup table. 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.

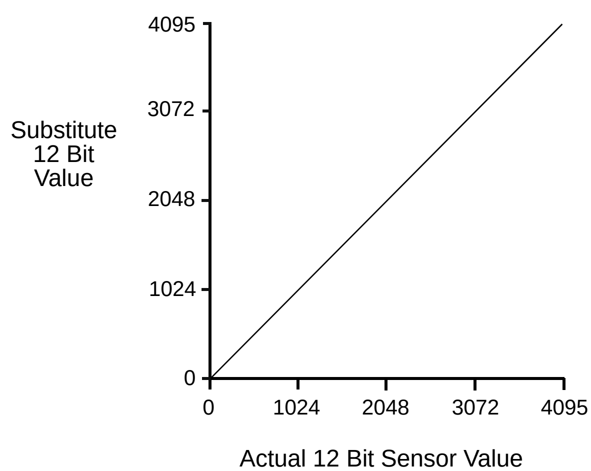


Fig. 42: Lookup Table with Values Mapped in a Linear Fashion

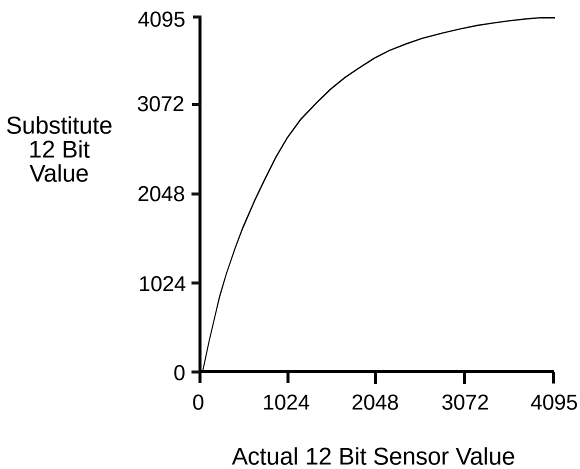


Fig. 43: Lookup Table with Values Mapped for Higher Camera Output at Low Sensor Readings

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 data 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 data 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 data 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.

9.7.1 Entering LUT Values and Enabling the LUT

Entering Values and Enabling the LUT Using Basler pylon

To enter values into the LUT and enable the LUT:

1. 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.)
2. Use the LUT Index parameter to select an index number.
3. Use the LUT Value parameter to enter the substitute value that will be stored at the index number that you selected in step 2.
4. Repeat steps 2 and 3 to enter other substitute values into the LUT as desired.
5. Use the LUT Enable parameter to enable the LUT.

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 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 );
```

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 3.1 on page 23.

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 direct register access, see Section 3.2 on page 25.

9.8 Error Detection

9.8.1 LED Indicator

The LED indicator on the back of the camera includes a small LED that can be lit red, green or orange. The LED indicates the camera’s current condition as shown in Table 13.

LED State	Status Indication
LED is Off	No power is supplied to the camera or the voltage is too low.
Continuous orange	The camera is booting up
Continuous green	The camera has booted up successfully and is OK.
Flashing green	The camera is set to expect an external trigger signal on one of the CC lines an input, but no trigger signal is present.
Flashing with red and green alternating	An error condition has been detected that may be correctable with user intervention. (See the next section for more information).
Quickly flashing red	Internal error. Contact Basler technical support.
Slowly flashing red	An over temperature condition has occurred.

Table 13: LED Indications

9.8.2 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	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.)
4	Over Temperature	The camera has stopped image acquisition due to overheating. Provide adequate cooling to the camera.
6	Insufficient Trigger Width	Trigger was too short to be detected by the camera.

Table 14: 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:

To check the error codes:

1. Read the value of the Last Error parameter. The Last Error parameter will indicate the last error code stored in the memory.
2. Execute the Clear Last Error Command to clear the last error code from the memory.
3. 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 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 3.1 on page 23.

Reading and Clearing the Error Codes Using Direct Register Access

To get the value of the last error code in the memory:

1. Read the value of the Last User Error register.

To clear the value of the last error code in the memory:

1. Set the value of the Clear Last User Error register to 1.

For more information about direct register access, see Section 3.2 on page 25

9.9 Test Images

The cameras include the ability to generate test images. Test images are used to check the camera's basic functionality and its ability to transmit images to the host PC. Test images can be used for service purposes and for failure diagnostics.

Test image generation is done internally by the camera's logic and does not use the optics or the imaging sensor. Six test images are available. They can be obtained when 8-bit, 10-bit, or 12-bit pixel formats are used.

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.

Digital features, however, can have effect on test images:

- Test images 1, 2, and 3: the camera's digital features, such as the luminance lookup table, will have no effect on the transmitted images.
- Test images 4 and 5: the camera's digital features, such as the luminance lookup table, 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.
- Test image 6: the camera's digital features, such as the luminance lookup table, will have no effect on the transmitted images, with one exception: When the Reverse X feature is enabled the test image will display anomalous colors.

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 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 3.1 on page 23.

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 direct register access, see Section 3.2 on page 25.

9.9.1 Test Image Descriptions

Test Image 1 - Fixed Diagonal Gray Gradient (8 bit)

Test image one will look similar to Figure 44, with fixed gray gradients ranging from black to white.

The test image demonstrates that the camera acquires an image and that it is received by the PC. The camera's digital features will have no effect.

The 8-bit test image is best suited for use when the camera is set for a monochrome 8 bit pixel format. However, the test image is also available for 10-bit and 12-bit pixel formats.

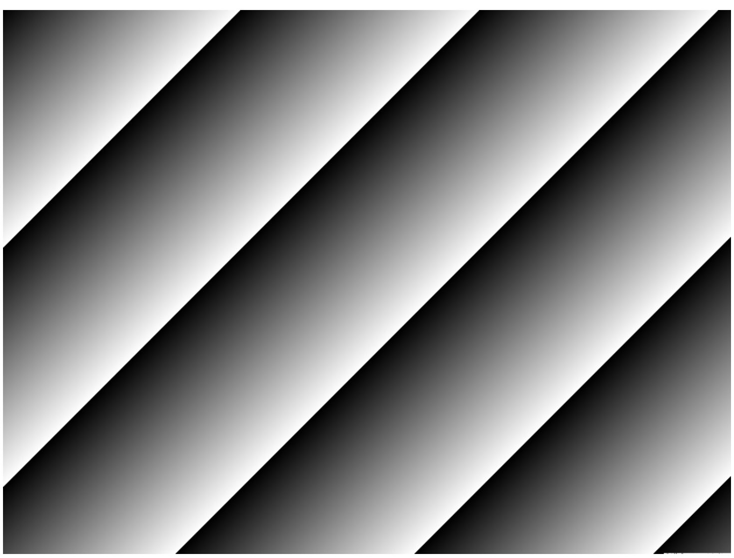


Fig. 44: Test Image One

Test Image 2 - Moving Diagonal Gray Gradient

The moving diagonal gray gradient test image is similar to test image 1, except that it is not stationary. The image moves by one pixel from right to left whenever a new image acquisition is triggered.

The test image demonstrates that the camera acquires a sequence of images and that they are received by the PC. The camera's digital features will have no effect.

Test Image 3 - Moving Diagonal Gray Gradient

The moving diagonal gray gradient test image is similar to test image 2, except that it is generated on the basis of 12-bit data. The image moves by one pixel from right to left whenever a new image acquisition is triggered.

The test image demonstrates that the camera acquires a sequence of images and that they are received by the PC. The camera's digital features will have no effect.

Test Image 4 - Moving Diagonal Gray Gradient Feature Test

Test image 4 is similar to test image 2, except that test image 4 **will** show the effects of a digital feature if it is enabled.

Test image 4 is therefore useful for checking the effects of digital features such as the luminance lookup table.

Test Image 5 - Moving Diagonal Gray Gradient Feature Test

Test image 5 is similar to test image 3, except that test image 5 **will** show the effects of a digital feature if it is enabled.

Test image 5 is therefore 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 a moving diagonal color gradient test image. It is available on color cameras only.

As shown in Figure 45, test image six consists of diagonal red, green, and blue gradients. The image moves by one pixel from right to left whenever a new image acquisition is triggered. The camera's digital features will have no effect, with one exception: When the Reverse X feature is enabled, gradients in green and magenta will be displayed.

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 selected pixel 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.

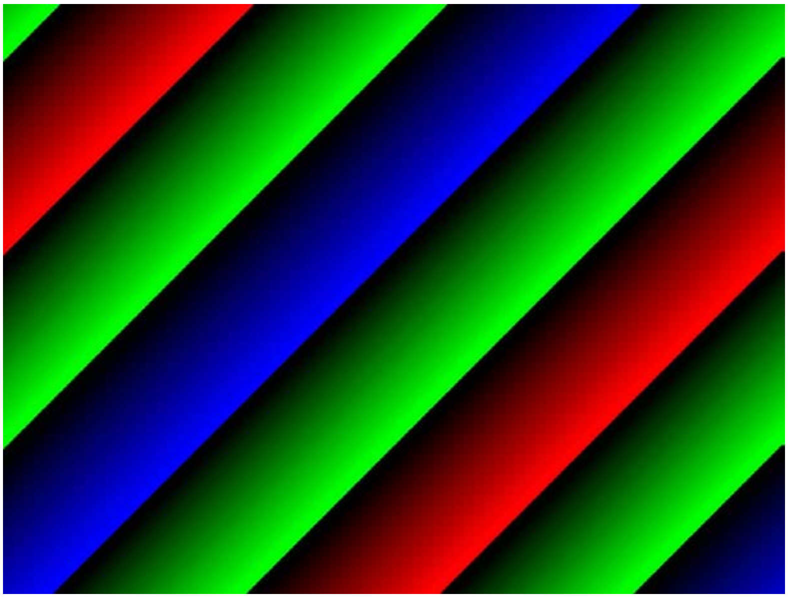


Fig. 45: Test Image Six (Reverse X Disabled)

9.10 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, beA4000-62km.
- 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. 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 Basler beat 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.
- Max Width (read only) - Indicates the camera's maximum area of interest (AOI) width setting.
- Max Height (read only) - Indicates the camera's maximum area of interest (AOI) height setting.

For more information about the pylon API and the pylon Viewer, see Section 3.1 on page 23.

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 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 3.2 on page 25.

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.
- Read the value in the Width Max register.
- Read the value in the Height Max register.

For more information about via direct register access, see Section 3.2 on page 25.

9.11 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 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 3.1 on page 23.

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 direct register access, see Section 3.2 on page 25.

9.12 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 user sets.

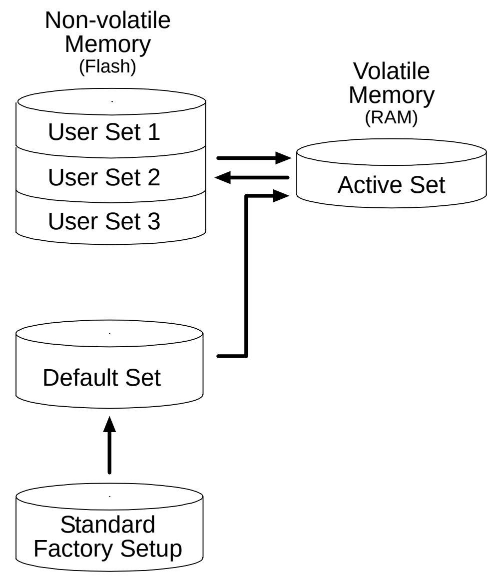


Fig. 46: Configuration Sets

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 direct register access, 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

The "default set" serves as a "placeholder" for a factory setup, lending particular accessibility to the factory setup that was selected as the default set:

The Standard Factory Setup - is the only available factory setup that can be selected as the "default set". Instructions for selecting the factory setup as the default set appear later in this section.

When a camera is manufactured, numerous tests are performed on the camera and a factory optimized setup is determined:

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.

The factory setup is saved in a permanent file in the camera's non-volatile memory. The file is not lost when the camera is reset or switched off and it cannot be changed.

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 this section.

User Sets

As mentioned above, 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 are three reserved areas in the camera's non-volatile memory available for saving configuration sets. A configuration set saved in a 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, a saved user set can be loaded into the active set. A 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 this section.

ⓘ	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.

Designating a Startup Set

You can designate the default set or one of the user sets 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 below.

9.12.1 Selecting a Factory Setup as the Default Set

The Standard Factory Setup is the only factory setup currently available. Accordingly, when the camera is delivered, the Standard Factory Setup will automatically be selected as the default set.

ⓘ	Loading the default set into the active set, with the Standard Factory Setup selected, 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.

Selecting a Factory Setup Using pylon

To select a factory setup to serve as the default set using Basler pylon (the Standard Factory Setup is the only choice available):

To select a factory setup to serve as the default set:

1. Set the Default Set Selector to the Standard 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 API to set the selector:

To select the Standard Factory Setup:

```
Camera.DefaultSetSelector.SetValue(DefaultSetSelector_Standard);
```

Selecting a Factory Setup Using Direct Register Access

To select a factory setup to serve as the default set:

1. Set the Default Set Selector register to the Standard Factory Setup.

9.12.2 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

To save the current active set to a user set:

1. Make changes to the camera's settings until the camera is operating in a manner that you would like to save.
2. Set the User Set Selector to User Set 1, User Set 2, or User Set 3 as desired.
3. 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 API to set the selector and execute the command. The camera settings are saved to User Set 1 as an example.

```
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 3.1 on page 23.

Saving User Sets Using Direct Register Access

To save the current active set to a user set:

1. Make changes to the camera's settings until the camera is operating in a manner that you would like to save.
2. Set the value of the User Set Selector register to User Set 1, 2, or 3 as desired.
3. Set the value of the User Set Save register to 1.

For more information about direct register access, see Section 3.2 on page 25.

9.12.3 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 previous parameters in the active set. Since the settings in the active set control the current operation of the camera, the new 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:

To save the current active set to a user set:

1. Set the User Set Selector to User Set 1, User Set 2, User Set 3, or Default as desired.
2. 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 API to set the selector and execute the command. User Set 2 and the default set are used as examples.

```
// 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 3.1 on page 23.

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:

To save the current active set to a user set:

1. Set the value of the User Set Selector register to User Set 1, 2, or 3, or to the Default set as desired.
2. Set the value of the User Set Load register to 1.

For more information about direct register access, see Section 3.2 on page 25.

9.12.4 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 API to set the selector. User Set 1 and the Default Set are used as examples.:

```
// 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 3.1 on page 23.

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:

To select a startup set::

1. 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 direct register access, see Section 3.2 on page 25.

10 Troubleshooting and Support

This chapter explains camera reset and outlines the resources available to you if you need help working with your camera.

10.1 Camera Reset

Some situations may require a camera reset. It is executed using the DeviceReset command parameter via the pylon API and is therefore also referred to as a "software reset".

During camera reset, camera power stays on. This is in contrast to camera restart where camera power is switched off and on again.

From the perspective of the Basler pylon software and the operating system, camera reset and camera restart both appear as a "surprise device removal" and must be handled accordingly.

> When camera reset is carried out, all settings stored in the camera's volatile memory are lost.
>
> If you want to preserve settings stored in the camera's volatile memory, safe them as a user set before carrying out camera reset. Note that some settings can not be saved in a user set, for example the settings for the luminance lookup table.
>
> For more information about saving user sets, see Section 9.12.2 on page 157.

Resetting the camera

After having issued the camera reset command perform the subsequently necessary steps, e.g. some cleanup on the PC, in accord with the DeviceRemovalHandling sample code that is included in the pylon SDK documentation.

After camera reset was carried out, allow some time to elapse until the camera is detected again.

For more information about the pylon API and SDK, see Section 3.1 on page 23.

10.2 Tech 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 Downloads and the Support sections of our 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.

10.3 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 our website: www.baslerweb.com

10.4 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 Frame grabber that you use with the camera: ______________________

4 Describe the problem in as much detail as possible: ______________________
(If you need more space, use an extra sheet of paper.)

5 If known, what's the cause of the problem? ______________________

6 When did the problem occur?

- ☐ After start.
- ☐ While running.
- ☐ After a certain action (e.g., a change of parameters): ______________________

7 How often did/does the problem occur?

- ☐ Once.
- ☐ Every time.
- ☐ Regularly when:

  ____

- ☐ Occasionally when:

  ____
  ____
  ____
  ____

8 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.

9 Did your application ever run without problems?

- ☐ Yes
- ☐ No

10 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.

If you cannot access the camera, please try to state the following parameter settings:

- ☐ Frame Size: ______________________________
- ☐ Pixel Format: ______________________________
- ☐ Exposure Time: ______________________________

  ____

11 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
AW00130801000	19 Sep 2014	First release of this document. Applies to the prototype camera only.
AW00130802000	24 Nov 2014	Minor corrections in the "Subpart ... of FCC Rules" Section on the back of the front page. Changed the camera name to Basler beat throughout the manual. Updated the camera model name to beA4000-62km. Added the beA4000-62kc camera. Modified Section 1.7 on page 14 to include contents for UL certification. Integrated precautions concerning SELV and LPS in Section 1.8 on page 16, Section 5 on page 31.
AW00130803000	16 Apr 2015	Applies to the prototype camera only. Added Chapter 6, Chapter 7, Chapter 8, and Chapter 9.
AW00130804000	15 June 2015	First release for series cameras.

Index