

Glossary

- [B](#)
- [C](#)
- [D](#)
- [G](#)
- [H](#)
- [I](#)
- [M](#)
- [R](#)
- [S](#)
- [U](#)

B

Bayer Mosaic

The term Bayer Mosaic is often used in combination with single-chip CCD sensors to which a color filter is applied. These filters make individual pixels react differently to wavelengths of light. As there exists only one color component per pixel, the missing color components must be extracted.

The attached article will explain how a Bayer Mosaic Filter is designed and what methods are available for converting data to RGB.

[Back to top](#)

C

C-Mount 1" inch x 32 TPI UN 2A

Standard connection for lenses used with CCD cameras, with uniform connection thread (at the lens: connection thread 1"-32UN-2A) with uniform depth-of-field of 17.526 mm.

CameraLink

CameraLink (CL) is popular interface designed for computer vision applications. Specified for fast image transfers, the interface supports 8 to 16 bits bit depth per pixel and a maximum pixel clock of 85 MHz.

CameraLink offers three variants:

- BASE (max. 24 bits per clock)
- MEDIUM (max. 48 bits per clock)
- FULL (max. 64 bits per clock)

MATRIX VISION offers following CameraLink frame grabbers:

Features		mvGAM MA-CL	mvTITA N-CL	mvHYPE RION- CLb	mvHYPE RION- CLe	mvHYPE RION- CLm	mvHYPE RION- CLf
Supported configurations	1x BASE	yes	yes	yes	yes	yes	yes
	2x BASE				yes	yes	
	1x MEDIUM					yes	yes
	1x FULL						yes
Driver (mvIMPA CT Acquire)	Windows®	XP, Vista, 7	XP, Vista, 7	XP, Vista, 7	XP, Vista, 7	XP, Vista, 7	XP, Vista, 7
	Linux®			32 / 64 bits			
Bus	PCI	32 bits / 33 MHz Rev. 2.1	32 bits / 33 MHz Rev. 2.1				
	PCI			x1	x1	x4	x4

	Express®						
Continuous data rate	MB/s	95	100	200	200	640	640
			mvTITA N- CL/110: 110				
Max. pixel clock	MHz	66	66	85	85	85	85
				yes	yes	yes	yes



CCD vs. CMOS

There are two sensor technologies on the market: **CCD** and **CMOS**.

Normally, CCD sensors have a better image quality, lower noise and no fixed pattern noise. In contrast CMOS sensors are cheaper and also have additional features, which are not integrable to CCD technology. Usually CCD sensors stand out due to an higher dynamic; an enormous advantage mainly in applications with greater differences in brightness. Whether to use a gray scale sensor or a color sensor depends on the task. Some sensors are only available in one version. Color sensors have a color filter structure in front of the light sensitive sensor matrix, i.e. specific sensors receive only light of a specific color. This filter structures are permeable for IR light. To avoid falsification of the color during color acquisitions, a additional IR filter is needed. Due to the pixel-wise color change, however, this leads to a lesser spatial resolution. If a high color accuracy is needed like in color checkings of printouts or if a high spatial color resolution is needed, you have to use 3 chip cameras, which uses a separate chip for every color red, green and blue. A further aspect is the shutter. CCD and CMOS sensors are available with global shutter (full frame), simple CMOS sensors mostly have rolling shutter. With fast moving objects rolling shutter causes geometrical distortions due to the movement during the exposure.

Characteristics	CCD	CMOS
Signal at pixel output gate	Electron packagese	Voltage
Signal at chip output gate	Voltage (analog)	Bits (digital)
Signal at camera output gate	Bits (digital)	Bits (digital)
Filling faktor / aperture	High	Medium

System noise	Low	Medium
System complexity	High	Medium
Sensor complexity	Low	Low
Camera components	PCB + different chips + lens	High
Research and developing costs	Depends on application	Chip + lens
System costs	Depends on application	Depends on application
Performance	CCD	CMOS
Reactivity	Medium	A bit better
Dynamic	High	Medium
Uniformity of the pixels	High	Low .. medium
Uniform exposure time	Fast, combined	Slow
Speed	Medium .. high	Higher
Windowing	Limited	Extended
Antiblooming	High .. none	High
Power supply and pulsing	High voltage, different	Lower voltage, easy

In a nutshell

CCD sensors have a better image quality, a higher sensitivity and dynamic and a synchronous exposure control of all pixels, for the most part CMOS cameras are more compact, allow higher frame rates and can be used a little more variably.

CS-Mount 1" Zoll (inch) x 32 TPI UN 2A

Same as C-mount except for the depth-of-field. With CS-mount the depth-of-field is 12.5 mm. The CS-mount standard is suitable for short housings.

[Back to top](#)

D

Dual-GigE

Current interfaces like GigE, USB 2.0, etc. have a natural limit: the bandwidth. This limit prevents faster cameras with higher resolutions. Dual-GigE cameras provide a remedy and offer twice the bandwidth (≤ 240 Mbytes) with the same transparency in the software. i.e. the Dual-GigE camera appears as one device to the software. Neither a new standard nor new technology is necessary. GigE Vision 2.0 defines Dual-GigE and for this reason these cameras can use a fully developed and established standard. Easier programming and less power consumption compared to 10-GigabitEthernet as well as the possibility to use existing network infrastructure (10-GigabitEthernet networks are not yet widespread and are expensive) are several advantages compared to other interfaces like USB3, CoaXPress and Camera Link HS.

[Back to top](#)

G

GenICam

GenICam stands for **GEN**eric programming **I**nterface for **CAM**eras. It's a generic way to access and modify device parameters with a unified interface. A GenICam compliant device either directly provides a GenICam compliant description file (in internal memory) or the description file can be obtained from a local (harddisc etc.) or web location. A GenICam description file is something like a machine readable device manual. It provides a user readable name and value range for parameters that are offered by the device for reading and/or writing and instructions on what command must be sent to a device when the user modifies or reads a certain parameter from the device. These description files are written in XML.

For further informations on this topic please have a look at <http://www.genicam.org>.

GenTL

GenTL (*Transport Layer*) is the transport layer of the GenICam standards and is responsible for the transport of the camera in the user application.

GigE Vision

GigE Vision is a network protocol designed for the communication between an imaging device and an application. This protocol completely describes:

- device discovery
- data transmission
 - image data
 - additional data
- read/write of parameters.

GigE Vision uses UDP for data transmission to reduce overhead introduced by TCP.

Note: *UDP does not guarantee the order in which packets reach the client nor does it guarantee that packets arrive at the client at all. However GigE Vision defines mechanism that allows to recognize lost packets. This allows capture driver manufacturers to implement algorithms that can reconstruct images and other data by requesting the device to resend lost data packets until the complete buffer has been assembled. For further information please have a look at <http://www.machinevisiononline.org/public/articles/index.cfm?cat=167>*

The MATRIX VISION GigE Vision capture filter driver as well as the socket based acquisition driver and all MATRIX VISION GigE Vision compliant devices support resending thus lost data can be detected and in most cases reconstructed. This of course can not enhance the max. bandwidth of the transmission line thus if e.g. parts of the transmission line are overloaded for a longer period of time data will be lost anyway.

Both capture drivers will allow to fine tune the resend algorithm used internally and both drivers will also provide information about the amount of data lost and the amount of data that was re-requested. This information/configuration will be part of the drivers SDK. More information about it can be found in the corresponding interface description.

[Back to top](#)

H

High Dynamic Range

The HDR (High Dynamic Range) mode increases the usable contrast range. This is achieved by dividing the integration time in two or three phases. The exposure time proportion of the three phases can be set independently. Furthermore, it can be set, how many signal of each phase is charged.

Functionality

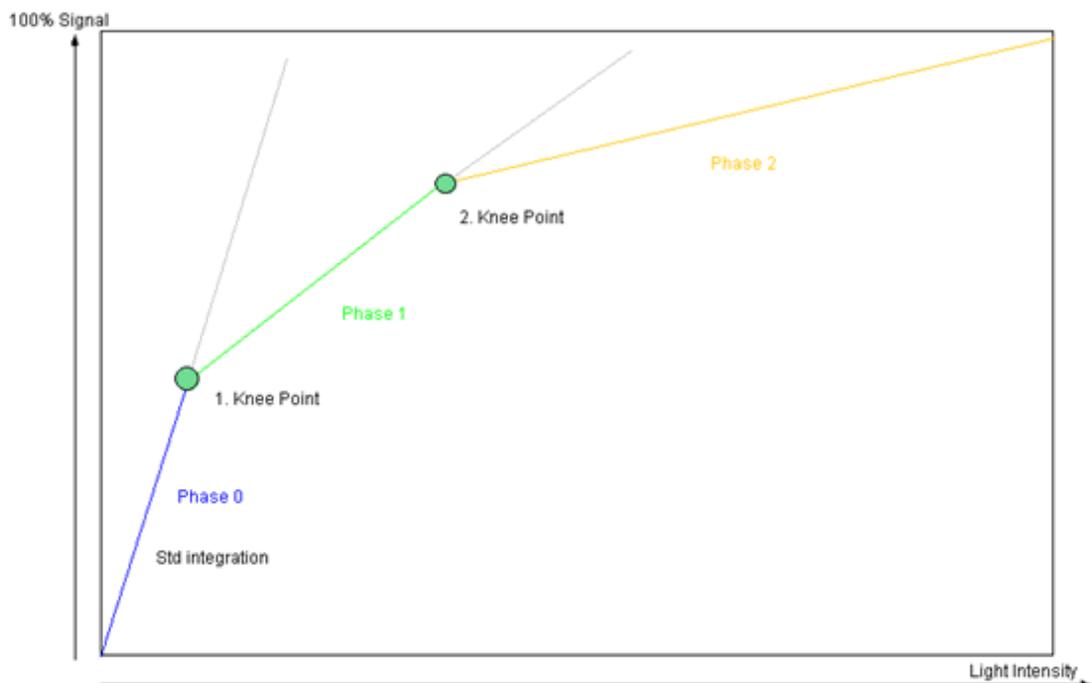


Figure 1: Diagram of the -x00w sensor's HDR mode

Description

- Phase 0
 - During T1 all pixels are integrated until they reach the defined signal level of Knee Point 1.
 - If one pixel reaches the level, the integration will be stopped.
 - During T1 no pixel can reached a level higher than P1.
- Phase 1
 - During T2 all pixels are integrated until they reach the defined signal level of Knee Point 2.
 - T2 is always smaller than T1 so that the percentage compared to the total exposure time is lower.
 - Thus, the signal increase during T2 is lower as during T1.

- The max. signal level of Knee Point 2 is higher than of Knee Point 1.
 - Phase 2
 - During T2 all pixels are integrated until the possible saturation.
 - T3 is always smaller than T2, so that the percentage compared to the total exposure time is again lower here.
 - Thus, the signal increase during T3 is lower as during T2.

For this reason, darker pixels can be integrated during the complete integration time and the sensor reaches its full sensitivity. Pixel's, which are limited at each Knee Points, lose a part of their integration time - even more, if they are brighter.

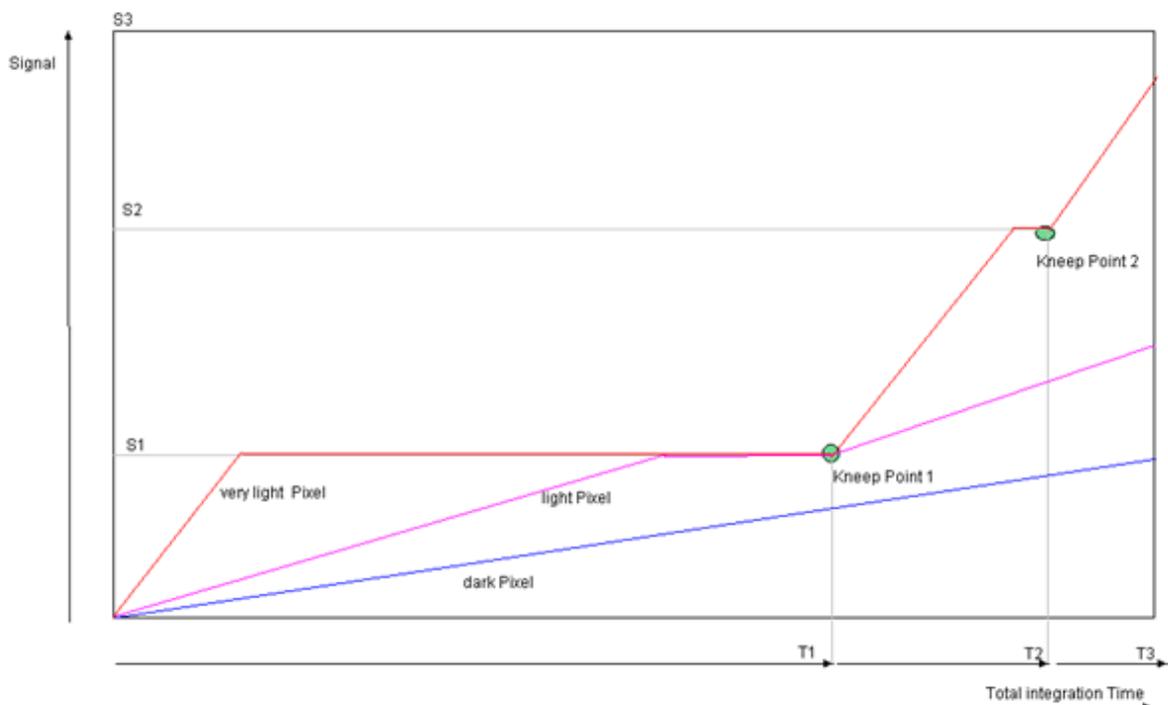


Figure 2: Integration time of different bright pixels

In the diagram you can see the signal line of three different bright pixels. The slope depends of the light intensity, thus it is per pixel the same here (granted that the light intensity is temporally constant).

Given that the very light pixel is limited soon at the signal levels S1 and S1, the whole integration time is lower compared to the dark pixel. In practice, the parts of the integration time are very different. T1, for example, is 95% of T_{total} , T2 only 4% and T3 only 1%. Thus, a high decrease of the very light pixels can be achieved. However, if you want to divide the integration thresholds into three parts that is $S2 = 2 \times S1$ and $S3 = 3 \times S1$, a hundredfold brightness of one pixel's step from S2 to S3, compared to the step from 0 and S1 is needed.

Using HDR with CMOS sensor -x00w

Figure 3 is showing the usage of the HDR mode. Here, an image sequence was created with the integration time between 10 μ s and 100ms. You can see three slopes of the HDR mode. The "waves" result from the rounding during the three exposure phases. They can only be partly adjusted during one line period of the sensor.

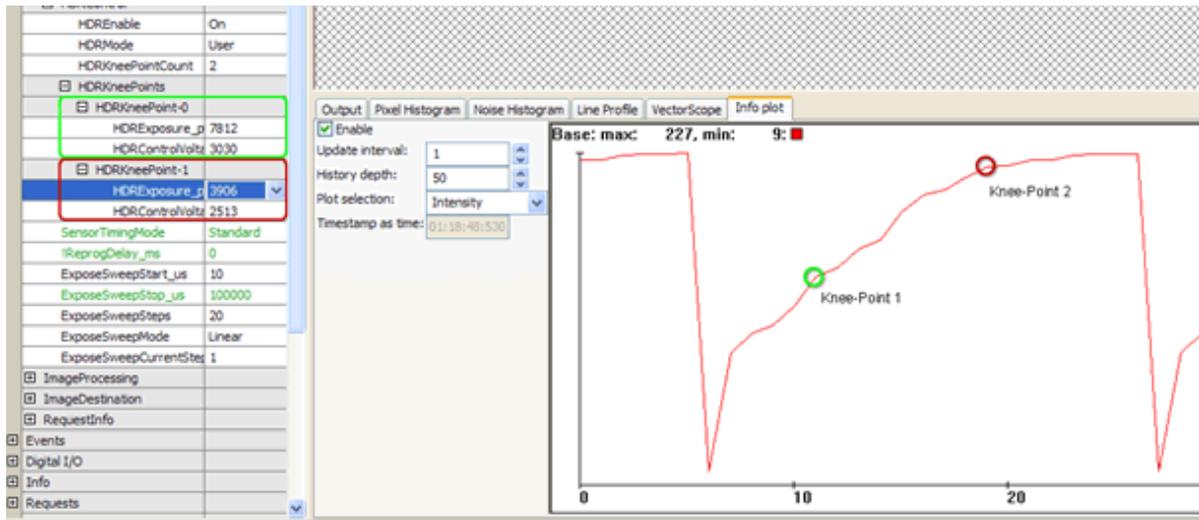


Figure 3: wxPropView HDR screenshot

Notes about the usage of the HDR mode with mvBlueFOX-200w

- In the HDR mode, the basic amplification is reduced by approx. 0.7, to utilize a huge, dynamic area of the sensor.
- If the manual gain is raised, this effect will be reversed.
- Exposure times, which are too low, make no sense. During the third phase, if the exposure time reaches a possible minimum (one line period), a sensible lower limit is reached.

Possible settings using mvBlueFOX-200w

HDREnable

- Off : Standard mode
- On : HDR mode on, reduced amplification
 - HDRMode:
 - Fixed setting with 2 Knee Points. modulation 0 .. 33% / 1 .. 66% / 2 .. 100%
 - Fixed0: Phase 1 exposure 12.5% , Phase 2 31.25% of total exposure

- Fixed1: Phase 1 exposure 6.25% , Phase 2 1.56% of total exposure
- Fixed2: Phase 1 exposure 3.12% , Phase 2 0.78% of total exposure
- Fixed3: Phase 1 exposure 1.56% , Phase 2 0.39% of total exposure
- Fixed4: Phase 1 exposure 0.78% , Phase 2 0.195% of total exposure
- Fixed5: Phase 1 exposure 0.39% , Phase 2 0.049% of total exposure
- User: Variable setting of the Knee Point (1..2), threshold and exposure time proportion
 - HDRKneePointCount: Number of Knee Points (1..2)
 - HDRKneePoints
 - HDRKneePoint-0
 - HDRExposure_ppm: Proportion of Phase 1 compared to total exposure in parts per million (ppm)
 - HDRControlVoltage_mV: Control voltage for exposure threshold of first Knee Point (3030mV is equivalent to approx. 33%)
 - HDRKneePoint-1
 - HDRExposure_ppm: Proportion of Phase 1 compared to total exposure in parts per million (ppm)
 - HDRControlVoltage_mV: Control voltage for exposure threshold of first Knee Point (2630mV is equivalent to approx. 66%)

HRTC

MATRIX VISION's **Hardware Real-Time Controller** (in short HRTC) is a component of the FPGA which is used for time-critical I/O and acquisition control. For this reason the HRTC supersedes the use of an external PLC for camera and process control in many cases. A HRTC program consists of a sequence of operating steps, which are processed by the controller. For the creation of these sequences, you can use, for example, the GUI tool **wxPropView**.

⊕ Device	
⊕ Image Setting	
⊕ Events	
⊕ Digital I/O	
⊕ DigitalInputs[2]	[Off , Off]
⊕ DigitalOutputs[4]	[Off , Off , Off , Off]
DigitalInputThreshold	2V
⊕ HardwareRealTimeController	
⊕ HRTCtrl_0	HRTC: Program running
ProgramSize	5
⊕ RTCtrlProgram	
⊕ Step0	{WaitClocks,99000}
OpCode	WaitClocks
Clocks_us	99000
⊕ Step1	{TriggerSet,[FrameID: 1]}
OpCode	TriggerSet
FrameID	1
⊕ Step2	{WaitClocks,1000}
OpCode	WaitClocks
Clocks_us	1000
⊕ Step3	{TriggerReset}
OpCode	TriggerReset
⊕ Step4	{Jump,[Address: 0]}
OpCode	Jump
Address	0
Mode	Run
ProgramState	Comple 5 words OK
Filename	default.rtp
int Load()	
int Save()	
⊕ Info	

Figure 1: Entering a Hardware Real-Time Controller program in wxPropView

The sample in Figure 1 shows how to achieve a defined image frequency of 10 images per second in five program steps.

For more information, please have a look at the corresponding product manual chapter "[HRTC - Hardware Real-Time Controller](#)" (here mvBlueFOX).

Note: To use the HRTC you have to set the trigger mode (*TriggerMode*) and the trigger source (*TriggerSource*) (C++ syntax):

```
CameraSettings->triggerMode = ctmOnRisingEdge
CameraSettings->triggerSource = ctsRTCtrl
```

With wxPropView:

Image Setting	
BasedOn	
Camera	
Aoi	X:0 Y:0 W:640 H:480
AutoExposeControl	Off
AutoGainControl	Off
Gain_dB	0.000 dB
OffsetAutoCalibration	On
OffsetAutoBlackLevel	0
TriggerMode	OnRisingEdge
TriggerSource	RTCtrl
PixelClock_kHz	24000

Figure 2: Set HRTC as TriggerSource

Areas of use and applications

Current trends in digital cameras are moving towards the usage of bus systems like IEEE1394, USB and Gigabit Ethernet, which are not capable of real-time. If applications with digital cameras do require complex trigger and flash control, cameras with I/O and trigger inputs like the mvBlueFOX may be used or additional separate I/O boards come into operation. When using I/O boards some uncertainty due to the latency of the bus systems still persists. Therefore it is more sensible to move the real-time relevant features into the camera and thus simplify the local system.

Possible applications (excerpt) are:

- Generation of trigger signals
- Synchronization of multiple cameras
- Fast generation of image sequences with different flash and exposure control
- Dark and light image acquisitions to generate a reference image
- Exposure control on images with different wave lengths (R/G/IR)

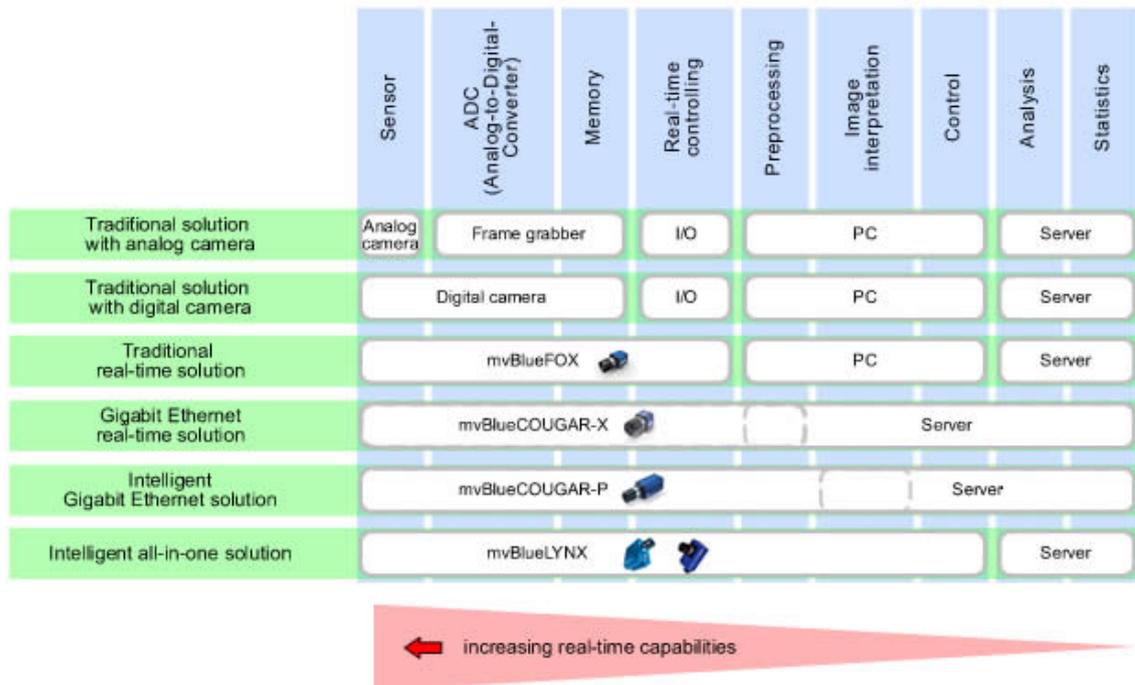


Figure 3: Processing chain from sensor to server: the more components in-between, the bigger the friction loss

Example: Delay the expose start of the following camera

If a defined delay should be necessary between cameras (here mvBlueFOX), the HRTC can do the synchronization work.

In this case, one camera must be the master. The external trigger signal that will start the acquisition must be connected to one of the cameras digital inputs. One of the digital outputs then will be connected to the digital input of the next camera. So camera one uses it's digital output to trigger camera two. How to connect the cameras to one another can also be seen in the following image:

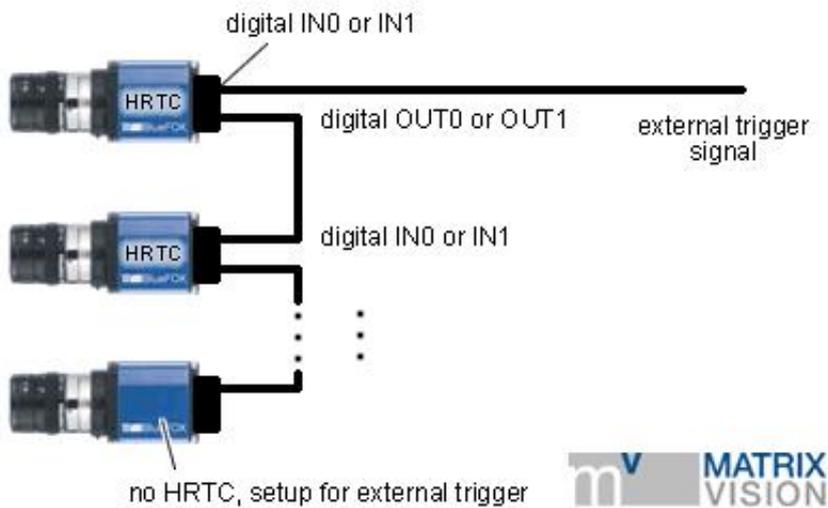


Figure 4: Connection diagram for a defined delay from the exposure start of one camera relative to another

Assuming that the external trigger is connected to digital input 0 of camera one and digital output 0 is connected to digital input 0 of camera two. Each additional camera will then be connected to it predecessor like camera 2 is connected to camera 1. The HRTC of camera one then has to be programmed somehow like this:

```

0. WaitDigin DigIn0->On
1. TriggerSet 0
2. WaitClocks
3. TriggerReset
4. WaitClocks
5. SetDigout DigOut0->On
6. WaitClocks 100µs
7. SetDigout DigOut0->Off
8. Jump 0

```

should not less than 100µs.

When the cameras are set up to start the exposure on the rising edge of the signal, of course is the desired minus . If more then two cameras shall be connected like this, every camera except the last one must run a program like the one discussed above. The delay times of course can vary.

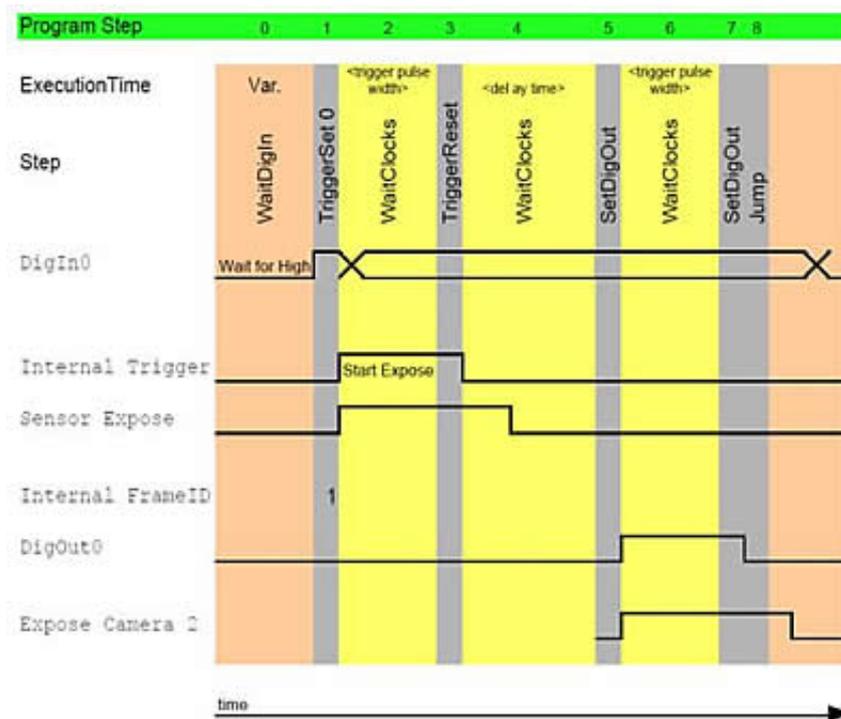


Figure 5: Delay the expose start of the following camera

Products

Following products have a HRTC:

- [mvBlueFOX](#)
- [mvBlueFOX-M](#)
- [mvBlueFOX-MLC](#)
- [mvBlueLYNX](#)
- [mvBlueLYNX-M](#)

[Back to top](#)

I

Image Processing Standards

The attached article describes consumer interface standards

- GenICam
- GigE Vision
- USB3 Vision

Industrial image processing

Industrial image processing or **digital image processing** means the digital processing of image data. In contrast to the manual processing of digital image editing the digital image processing generally applies following:

- extract specific criteria from the image data for further processings or analysis
- the image data aren't often needed and partly aren't even displayed
- the acquisition and the processing take place automatically by using predefined procedures and sequences in closed systems
- the complexity of the algorithms partly demands special hardware to process the image data

Areas of use and applications

Consecutively there are some example applications from different areas of use:

Automation, industry

- workpiece checking on dimensional accuracy and completeness
- surface checking on correct print and freedom from errormounting check of printed circuit boards
- control and tracing of the flow of material
- visualization of process flows in a control center

Microscopy (medicine, research)

- improvement and analysis of light microscopic shots
- acquisition and calculation of electron microscope images
- extraction of laser scan shots e.g. in the ophthalmology

Medicine

- eyes diagnosis (see above)
- categorisation of burns
- acquisition of tooth profiles

Safety engineering

- compression and recording of image sequences in banks and petrol stations
- analysis of objects in images (video sensor technology)
- infrared camera technology
- recording of cash machine transactions

Traffic engineering

- surveillance, reckoning and control of the flow of traffic
- billing of parking garage fees via licence plate OCR
- cameras in vehicles: distance measurements, electr. rear-view mirrors, road sign recordation
- surveillance/inspection of elements at risk (aircraft turbines)

Commerce

- empties check
- determination of the flow of consumers to control the deployment

[Back to top](#)

M

MARTIX VISION SDK's / interfaces

mvSDK

- oldest interface
- close to hardware
- creation time-consuming
- each hardware family with own driver DLL
- not object-oriented
- active support is discontinued

mvAcquireControl

- successor of mvSDK
- based on grabber.dll (mvSDK runs in the background)

- C and C++ interface for the user available
- active support is discontinued

mvIMPACT Acquire

- current interface since 2004
- incompatible to mvSDK or mvAcquireControl
- object-oriented
- C and .NET wrapper for the user available
- applications based on mvIMPACT Acquire compatible to all product families

MvIMPACT Acquire

mvIMPACT Acquire is the current application programming interface (API) for MATRIX VISION hardware. A number of current programming languages are supported like:

- C,
- C++,
- C# and
- VB.NET.

[Back to top](#)

R

Rolling Shutter

Some MATRIX VISION industrial cameras or smart cameras are using CMOS sensors with rolling shutter. This means that the exposure of the single image lines starts and ends at different times (see Figure 1).

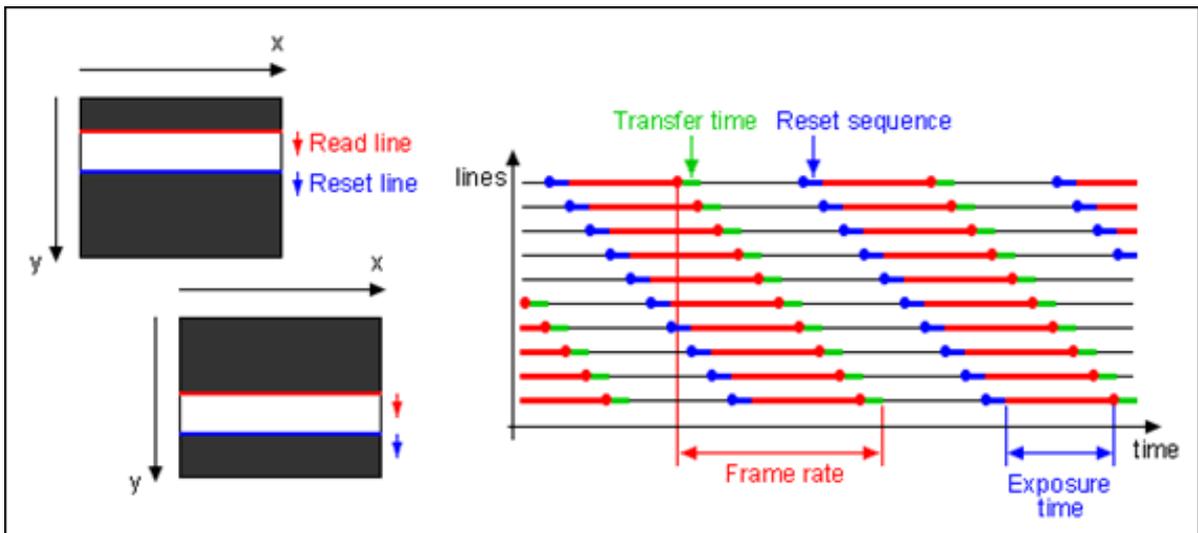


Figure 1: Exposure of every line starts and ends at different times

Figure 1 has two sections. The left side shows two states of a sensor with a line block, which shifts over the sensor. The white area represents the light sensitive area. This area shifts line-by-line top down the image. For example if you set the exposure time to 100 lines, this area will have a height of 100 lines. When the integration window shifts to the next line, the line first has to be reset ("Reset line"; blue in Figure 1). At the top of the area the line will be read out, after it was exposed for 100 lines ("Read line"; red in Figure 1).

Every line is opened according to the integration time, however, the exposure happens time-shifted. The right side of Figure 1 tries to clarify this with the help of a lines/time diagram. Line by line the start of the exposure ("Exposure time") shifts, i. e. rightwards in the diagram. Furthermore the right side of Figure 1 clarifies the phases of the image acquisition with a rolling shutter. Every line performs a reset sequence, a exposure process ("Exposure time") and a transfer process ("Transfer time"). The start of the transfer process of the first line and the end of the transfer process of the last line defines the frame rate. In short:

1

$$\text{Frame_rate} = \frac{1}{\text{transfer_time} * \text{image_height}}$$

If the exposure time longer than the transfer process, in the formula you have to replace the transfer time by the exposure time.

Exposure effects during horizontal movement

During an exposure if an object is moving horizontally you will receive a shifted image.

Figure 2 shows how the object moves from left to right. During the line-by-line delayed acquisition the rolling shutter of the sensor, which has only three lines for clarification, notices only parts of the object at other positions. The composed acquisition in the example displays the image shift. Additionally the movement of the object causes small blur effects.

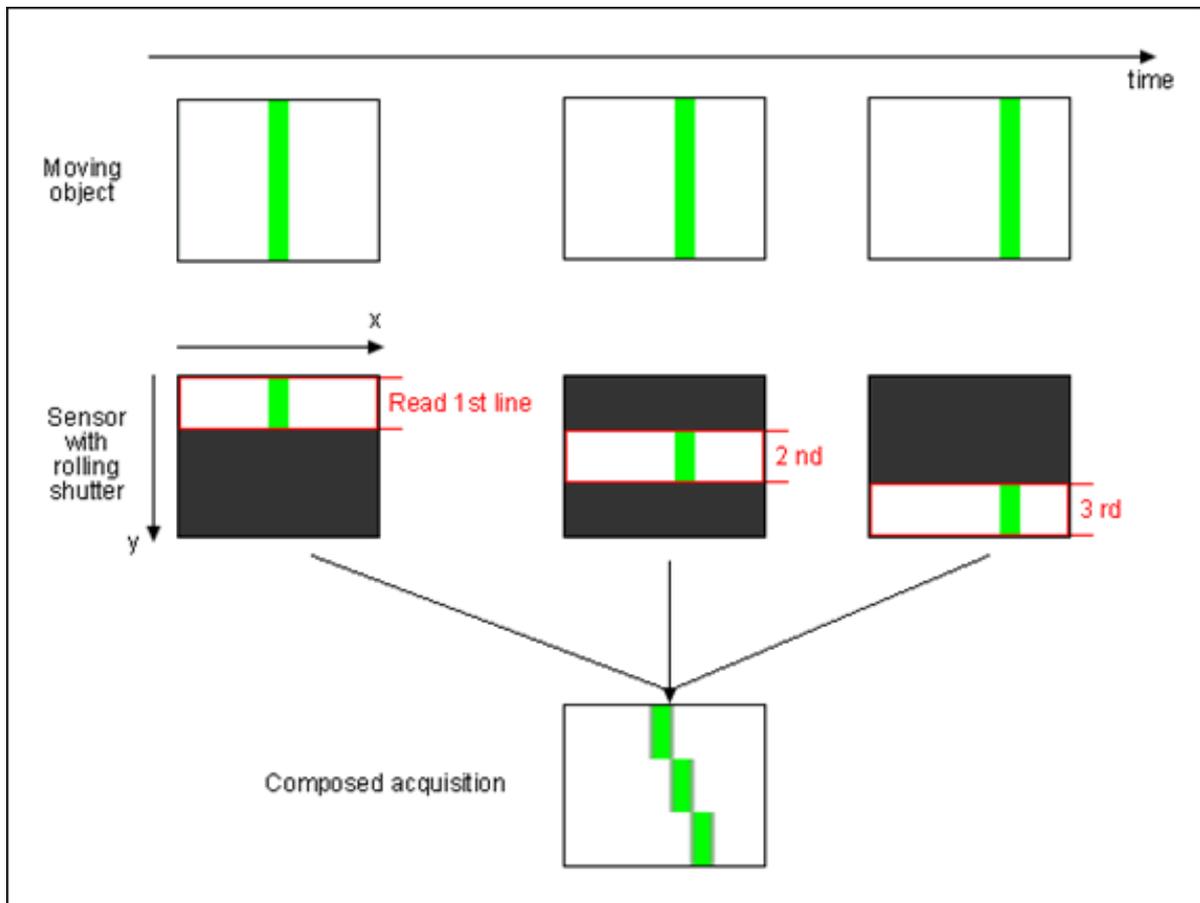


Figure 2: Shifted image during horizontal object movement

Looking at the line-by-line movement of the object, it is exciting, how short the covered distance of the object is. For example: If we have a rolling shutter sensor with the height of 480 pixels and a frame rate of 30Hz (30 images per second) an object with the speed of 10 meters per second (36 km/h) will cover a distance of 0.694 milimeters during every line change.

14400 lines are read out every second:

$$\text{read_lines} = 30\text{Hz} * 480 \text{ lines} = 14400 \frac{\text{lines}}{\text{s}}$$

The outcome of this is, every 0.0000694 seconds a line change happens:

$$\text{line_change} = \frac{1}{14400} = 0.0000694 \text{ s}$$

If the object has a speed of 10 m/s, it covers a distance of 0.694 mm during every line change:

$$s = v * t$$

$$s = 10 \frac{\text{m}}{\text{s}} * 0.0000694 \text{ s}$$

$$s = 0.000694 \text{ m} = 0.694 \text{ mm}$$

For this reason a rolling shutter sensor is suited for movement analyses with high frame rates.

Exposure effects during vertical movement

During an exposure if an object is moving vertically you will receive a compression or dilation. This depends on the direction of the movement. In Figure 3 an object moves vertically in opposition to the exposure direction of the sensor, which has only three lines for clarification. The example shows that the object is compressed. The image of the composed acquisition verifies this.

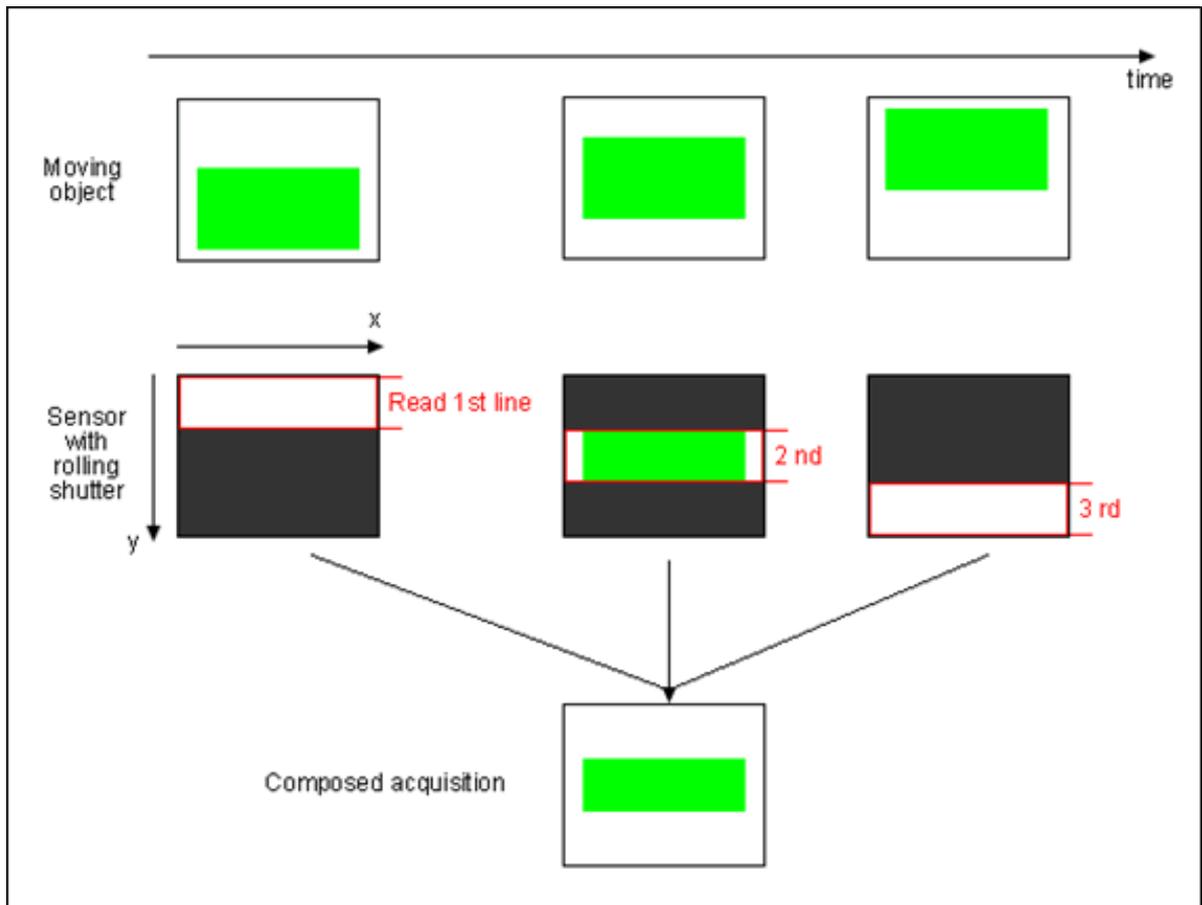


Figure 3: Compression during vertical object movement

The rolling shutter effect can be used in own interest. The line-by-line integration reduces the blur of a fast moving object. Consequently the integration time can be doubled.

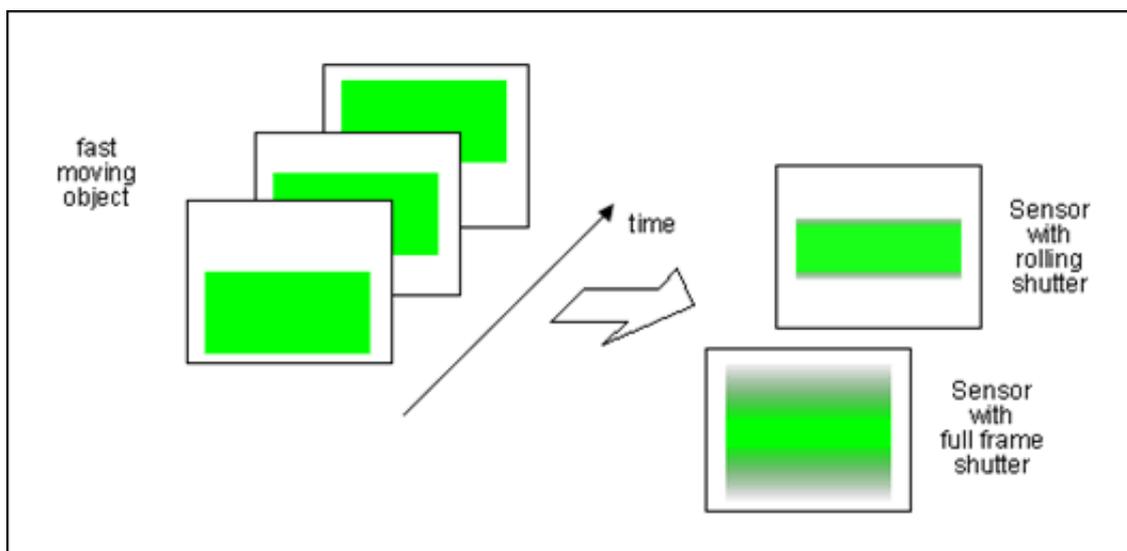


Figure 4: Reduced blur with rolling shutter

Flash control on CMOS sensors with rolling shutter

Flash control on CMOS sensors with rolling shutter is limited. The exposure time has to be long enough to flash all lines at the same time (see Figure 4 "All row integration"). The flash moment has to be in the area of "All row integration" and the flash time may not be longer than "All row integration"

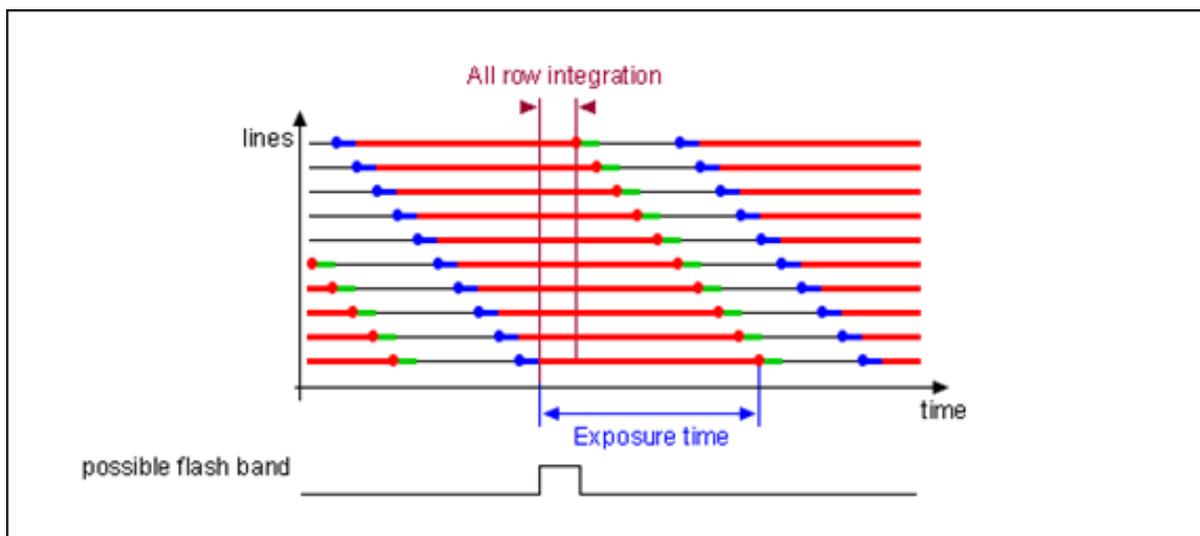


Figure 5: Possible flash window

In this modus operandi you have to mind, that there is no extraneous light during the exposure of the lines ("Exposure time").

[Back to top](#)

S

Sensor image errors

Due to random process deviations, not all pixels in an image sensor array will react in the same way even if the condition is the same. Although the sensor manufacturer are outside the control of this, it is said that these are errors. Well-known are

- **defective pixels**,
- **dark current** and
- variations of **sensitivity** of pixels.

With different correction methods, these image errors can be easily eliminated. The attached PDF describes the single errors and shows how to correct them using wxPropView.

Smear effect (CCD)

The smear is a known effect of CCD sensors and can be treated as parasitic sensitivity of the vertical shift register. Because we can shift much faster during idle and exposure, we can make the smear line partly less visible. MATRIX VISION offers a fast shifting mode called mvSmearReduction.

The attached application note describes the effect and the mvSmearReduction mode.

SNFC

Standard Feature Naming Convention of GenICam.

The latest GenICam properties list can be found here:

http://www.emva.org/genicam/genicam%E2%84%A2_document_download

The file is called "GenICam™ Standard Features Naming Convention (PDF)".

[Back to top](#)

U

USB vs. Firewire

In the meantime as a replacement for analog cameras with frame grabbers based on a PC many digital cameras with different interfaces are in place on the image processing market. While on the high end of the performance spectrum special frame grabber solutions are needed, in the low end cost effective USB 2.0 and FireWire solutions spread out.

Why do we have USB as interface for our industrial camera?

The USB technology offers in comparison to the FireWire some advantages:

- The USB interface is available on all current laptops, PCs and many embedded industrial systems.
- The USB interface guarantees a secured transfer.
- The bandwidth allocation works dynamically (in contrast to DCAM [1894-based Digital Camera Specification], which works isochronously).

Which advantages offers the industrial USB 2.0 camera mvBlueFOX from MATRIX VISION?

Beside the general advantages of the USB interface and the easy handling the mvBlueFOX features additional advantages:

- With HRTC (Hardware Real-Time Controller) it is possible to control the digital I/O flexibly.
- Flexible trigger possibilities are available.
- The cameras supports a dynamic camera control (changes of parameters "on-the-fly").
- Each image can be acquired with appropriate parameters.
- An industrial connector is available. The cameras are bus powered.
- The CCD sensors are adjusted. For this reason a position accuracy is reached of less than one percent.
- The cameras are available as OEM modules.
- User data can be saved on the camera.
- The drivers are powerful and provides many image processing functionality.
- A uniform driver architecture for all MATRIX VISION products (frame grabbers, cameras) and future products is available.
- The drivers are available for Linux (also for embedded systems like ARM, PPC etc.).

USB3 Vision

USB3 Vision is a standard for the image processing industry, which defines the communication between image acquisition device and application via USB 3.0. This protocol completely describes:

- device discovery
- data transmission
 - image data
 - additional data
- read/write of parameters.

[Back to top](#)