Industrial Cameras:
USB - 22, 27, 37, 42 and 72 Series
- Trigger and I/O

THE IMAGINGSOURCE
TECHNOLOGY BASED ON STANDARDS

Content

THE IMAGINGSOURCE
TECHNOLOGY BASED ON STANDARDS

Introduction

In addition to a number of other applications, The Imaging Source USB CMOS cameras are also used for industrial purposes. Most of these applications require a trigger input. These features are provided by all USB CMOS cameras whose product code contains a "BUC" (e.g. DMK 22BUC03, DMK 42BUC03, DFK 72BUC02, etc.).

The trigger input and the digital output of all housed cameras is available via an 4-pin Hirose connector while the board cameras provide a 4-pin Molex PicoBlaze connector for this purpose.

This manual describes the use of all The Imaging Source USB CMOS cameras of the Dxx 22xUC03, Dxx 42xUC03 and Dxx 72xUC02 series that provide an external trigger.

The following steps describe the use of the most commonly used digital input - the trigger input:

- Setup the camera as described in the Getting Started document that accompanies the camera.

- Install the software IC Capture as described in the Getting Started document.

- Make sure that the camera works correctly with IC Capture.

- Connect a trigger source that creates a positive pulse to pin 3 and 4 (pls. see page 7 for details) of the cameras Hirose connector as shown at the right hand side. The height of the pulse may lie between 3.3 and 12 V.
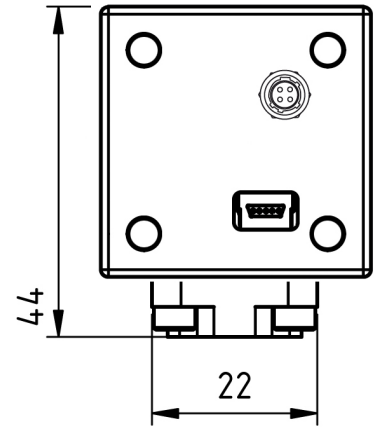
- Enable the trigger mode clicking: Device > Properties... > Special> Enable (please see the images below).
  If you see the dialog Activation External Trigger, please click "Disable all property automations...". Then, please set all camera parameters according to the requirements of your application. Please find these parameters in the Device Properties dialog (Device > Properties...):
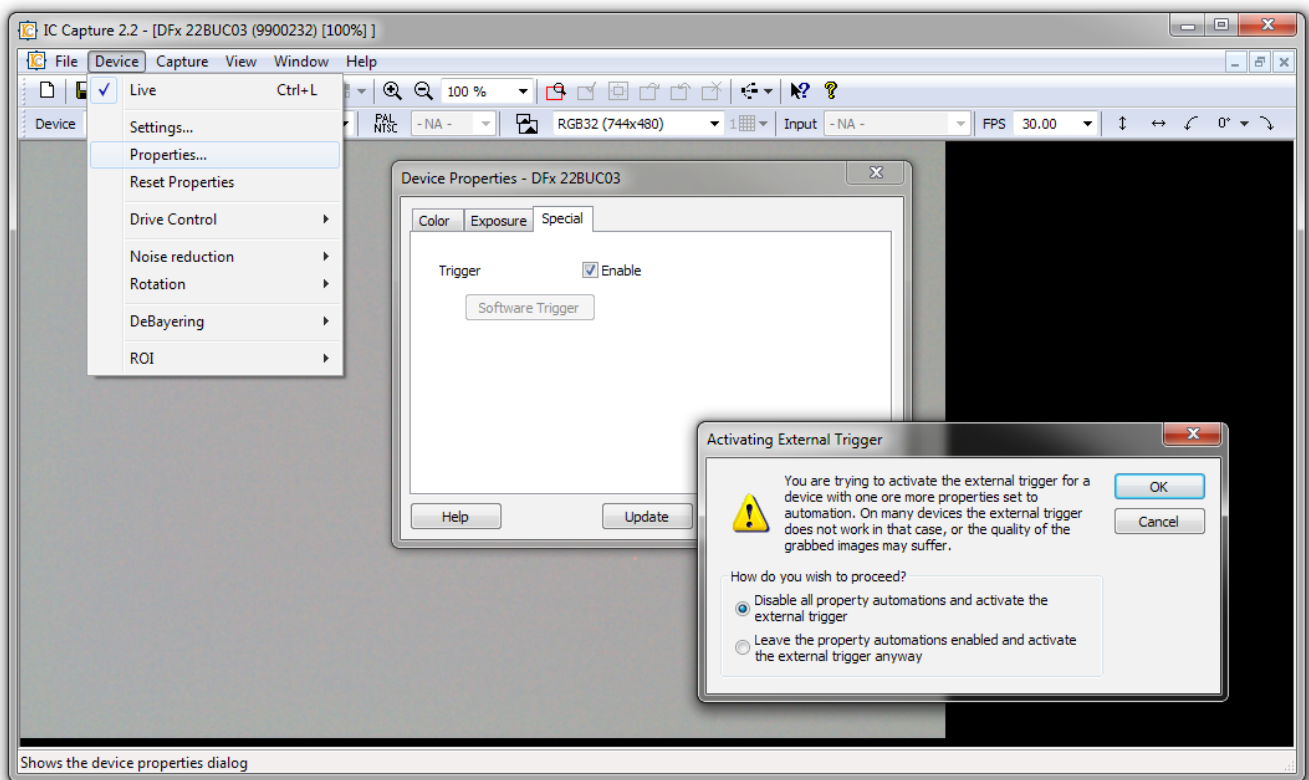
Trigger input - hardware and timing

The trigger input of The Imaging Source USB CMOS cameras is opto-coupled. It permits positive trigger pulses with any amplitude between 3.3 and 12 V (24 V using a series resistance of 1 kOhm).

R = 820 Ohm   Trigger_in   3,3 V up to 12 V

1 kOhm   R = 820 Ohm   Trigger_in   12 V up to 24 V
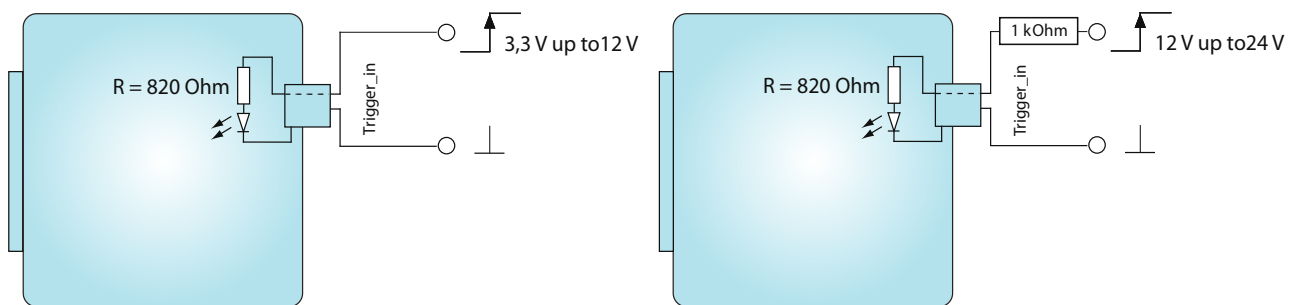
The Imaging Source USB CMOS cameras offers two different modes of operation:

Free running: The cameras generate a sequence of single images. The camera's clock generator determines the actual moment of exposure and thus the timing of the sequence. Please note that this timing cannot be controlled externally. Therefore, this mode of operation is called "free running".

Trigger: The cameras offer a trigger input to determine the moment of exposure. The length of the exposure can be set via software. The duration of the image readout is the reciprocal of the current frame rate. Once the image readout has finished, the camera is able to accept a new trigger pulse at any time.
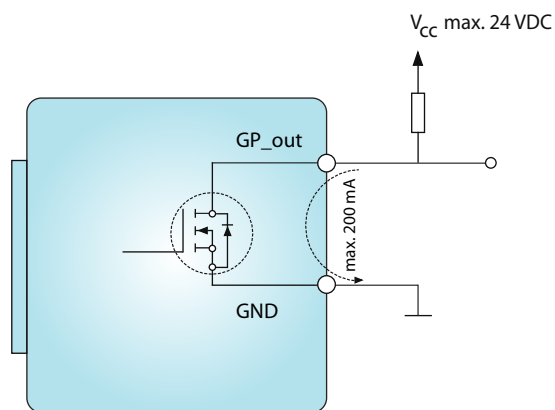
Trigger_in   > 10 μs

Exposure

Image readout   * see remarks below
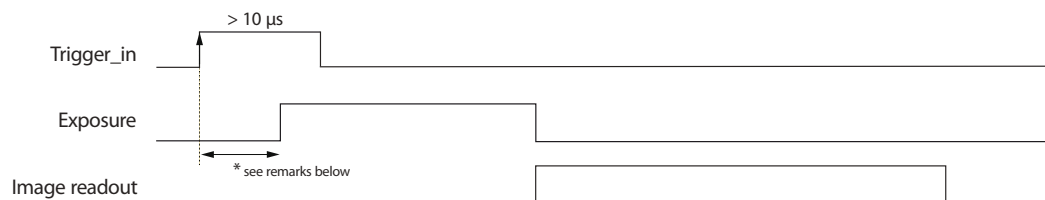
* Dxx 22BUC03:   9.4 μs (at max. frame rate)
  Dxx 42BUC03:   137 μs (at max. frame rate)
  Dxx 72BUC02:   rolling shutter

Connecting the general purpose output

(open drain)

The general purpose output is based on an open drain MOSFET. Please use a pull-up resistor to connect this output to Vcc (max. 24 VDC). Please note that the max drain current is 200 mA.
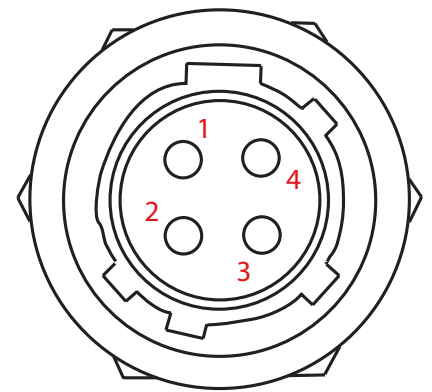
Trigger and digital out connectors

As already mentioned in the introduction, there are two ways to access the cameras trigger input and digital output: the housed cameras provide a 4-pin Hirose connector while the board cameras provide a 4-pin Molex PicoBlaze connector for this purpose.

Hirose connector

Please find the pinout of the Hirose connector in the table below and the position of these pins in the drawing on the right hand side. The part number of this connector is HR10A-7R-4P. To realize a trigger cable you need a Hirose connector HR10A-7P-4S.
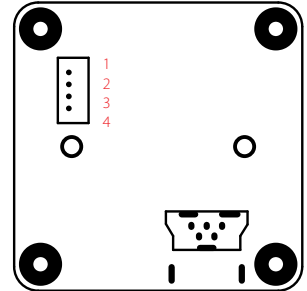
rear view of camera

Pin	Signal	I/O	Remarks	Characteristics			
				Min	Typ	Max	Unit
Pin 1	GND	G	External ground	-	-	-	-
Pin 2	GP_out	O	General purpose output (open drain)	-	-	24.0[1]	V
Pin 3	Trigger_in (-)	I	Start of exposure(optocoupler **ground)**	-	-	-	-
Pin 4	Trigger_in (+)	I	Start of exposure (optocoupler **signal**)	3.3	-	12.0	V

Please note:
   All specifications are subject to change without notice
   [1] max. 0.2 A (ID) for open drain MOSFET.

I/O pin legend:
   G External Ground
   I   Input
   O Output

Molex PicoBlade Connector 22/42/72 series

All board cameras of this series provide a 4-pin Molex PicoBlade connector for the trigger input and digital output. These cameras are shipped with a short open ended cable to simplify the access of the PicoBlade pins (part number the Pico Blade connector: Molex - 51021-0400). Please find the pinout of the PicoBlade connector in the table below and the position of these pins in the drawing on the right hand side.
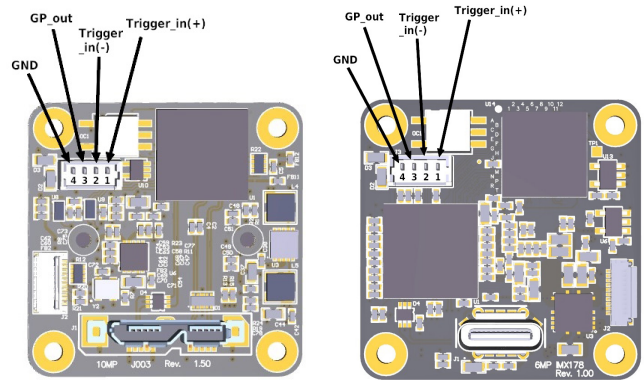

Pin	Color	Signal	I/O	Remarks	Characteristics			
					Min	Typ	Max	Unit
Pin 1	White	Trigger_in (-)	I	Start of exposure (optocoupler **ground**)	-	-	-	-
Pin 2	Yellow	Trigger_in (+)	I	Start of exposure (optocoupler **signal**)	3.3	-	12.0	V
Pin 3	Black	GND	G	External ground	-	-	-	-
Pin 4	Blue	GP_out	O	General purpose output (open drain)	-	-	24.0[1]	V

Please note:
All specifications are subject to change without notice
[1] max. 0.2 A (ID) for open drain MOSFET.

I/O pin legend:
G External Ground
I   Input
O Output

Molex PicoBlade Connector 27/37 series

All board cameras of this series provide a 4-pin Molex PicoBlade connector for the trigger input and digital output. These cameras are shipped with a short open ended cable to simplify the access of the PicoBlade pins (part number the Pico Blade connector: Molex - 51021-0400). Please find the pinout of the PicoBlade connector in the table below and the position of these pins in the drawing on the right hand side.

27 series

37 series

Pin	Color	Signal	I/O	Remarks	Characteristics			
					Min	Typ	Max	Unit
Pin 1	Orange	Trigger_in (+)	I	Start of exposure (optocoupler **signal**)	3.3	-	24.0	V
Pin 2	Violet	Trigger_in (-)	I	Start of exposure (optocoupler **ground**)	-	-	-	-
Pin 3	Green	GP_out	O	General purpose output (open drain)	-	-	24.0[1]	V
Pin 4	Black	GND	G	External Ground	-	-	-	-

Please note:
All specifications are subject to change without notice
[1] max. 0.2 A (ID) for open drain MOSFET.

I/O pin legend:
G External Ground
I Input
O Output

THE IMAGINGSOURCE
TECHNOLOGY BASED ON STANDARDS

Programming examples with IC Imaging Control®

All The Imaging Source cameras are shipped with the SDK IC Imaging Control®. IC Imaging Control® removes a lot of programming effort, since it offers many ready-to-use basic procedures.

Below are brief examples in Visual Basic to give you an idea of how to use IC Imaging Control® to control the trigger and the digital I/Os. You can learn more about IC Imaging Control® and download sample source code at www.imagingcontrol.com. Additionally, our support department (support@imagingcontrol.com) has some more detailed programming examples available for you.

Using the trigger

The program begins by assigning the video device (in this case the FireWire camera DFK 42BUC03), defines a `Video-Format` and sets the camera's operation mode to `DeviceTrigger`.

"Also the IC Imaging Control property "LiveCaptureContinuous" is set to true in order to save an image provided by the camera into the IC Imaging Control ring buffer and call the "ImageAvailable" event automatically, thus the application can process the new image.

In this sample the image is saved as JPEG image into the file "image.jpg"."

```
private void Form1_Load(object sender, EventArgs e)
{
    icImagingControl1.Device = "DFK 42BUC03";
    icImagingControl1.VideoFormat = "RGB32 (1280x960)";
    icImagingControl1.DeviceTrigger = true;
    icImagingControl1.LiveCaptureContinuous = true;
    icImagingControl1.LiveStart();
}

// This function is called, when a new image is provided after
// a trigger pulse occurred.
private void icImagingControl1_ImageAvailable(object sender,
    ICImagingControl.ImageAvailableEventArgs e)
{
    // Do something with the image, e.g. save it to a file.
    e.ImageBuffer.SaveAsJpeg("image.jpg", 90);
}
```

The first three program lines are similar to those of the preceding example (Activating the strobe output).

The main difference is to be found at the programs end:
The command `VCDProp.OnePush VCDElement_GPIORead` reads the digital inputs state, while `Debug.Print VCDProp.RangeValue(VCDElement_GPIOIn)` indicates this state in terms of a debug output.

```
// Declare the properties container.
private VCDSimpleProperty VCDProp;

private void Form1_Load(object sender, EventArgs e)
{
    icImagingControl1.Device = "DFK 42BUC03";
            // Query the camera properties.
    VCDProp = VCDSimpleModule.GetSimplePropertyContainer(icImagingControl1.VCDPropertyItems);

            // Read the GP Input
    VCDProp.OnePush(VCDIDs.VCDElement_GPIORead);

            // Show the read value of the input.
    Console.WriteLine("Input State : " +
        VCDProp.RangeValue[VCDIDs.VCDElement_GPIOIn]);
}
```

The first three program lines are similar to those of the preceding example (Reading the digital input).
The main difference is to be found at the end of the programs: The command VCDProp.`RangeValue` sets the variable
`VCDElement_GPIOOut` to 0, whereupon `VCDProp.OnePush VCDElement_GPIOWrite` copies the content of this variable (0 in our case) to the digital output.

```
// Declare the properties container.
private VCDSimpleProperty VCDProp;

private void Form1_Load(object sender, EventArgs e)

{
   icImagingControl1.Device ="DFK 42BUC03";
        // Query the camera properties.
        VCDProp = VCDSimpleModule.GetSimplePropertyContainer(icImagingControl1.VCDPropertyItems);

        // Set the new value for the output pin.
   VCDProp.RangeValue[VCDIDs.VCDElement_GPIOOut] = 0;

        // Write the output.
   VCDProp.OnePush(VCDIDs.VCDElement_GPIOWrite);
}
```