visiosens

User Manual

for

VFU-Camera-Series

Dear valid customer

Thank you for purchasing a VFU-Camera!

Please read the following chapters first to get a quick overview on what is new in this software release and on getting started with your new VFU Camera.

If you need further assistance please contact your local visiosens distributor for first level support in your language. This is:

Address:	Framos GmbH Zugspitzstr. 5 82049 Pullach Germany
Telephone:	+49.89.710667-58
Email:	support@framos.de

If you have any suggestions for improvement of our products, please send us a short feedback to feedback@visiosens.de. Thanks very much in advance.

Göttingen, August 1st, 2011

Lutz Brekerbohm, CEO visiosens GmbH

Table of Content

Table of Content 3
List of figures 4
1. Introduction 5
1.1 Safety Information 5
1.2 Operating Environment 5
1.3 EMC Directives 5
2. Installation 6
2.1 System Requirements 6
2.2 Installing VFU-Software on Windows© machines 7
2.2.1 SDK 7
2.2.2 Driver 9
2.3 Connecting a VFU Camera 10
3. Demo-Application 11
3.1 Starting the Demo-Application 11
3.2 Main Window 12
3.3 Image Settings Window 13
3.4 Brightness Control Window 14
3.4.1 Auto Brightness 15
3.4.2 Integration Time 16
3.4.3 Gain 16
3.5 White Balance Window 17
3.6 Camera Settings Window 17
3.6.1 Pixel Clock 18
3.6.2 Camera GPIO 18
3.6.3 Trigger IO 19
3.6.4 LED 20
3.6.5 Multiple Cameras Mode 21
3.7 User Flash 22
3.8 HDR 23
4. Software Development Kit 25
4.1 Philosophy 25
4.2 SDK Configuration 25
4.3 Logging 26

4.4 Image Processing 27
5. Camera Hardware 28
5.1 Family overview 28
5.2 Available sensors 28
5.3 I/O-functionality 29

List of figures

Figure 1: Setup Wizard 7
Figure 2: Installation Type 7
Figure 3: Successful Installation 8
Figure 4: Custom Installation 8
Figure 5: File Location 8
Figure 6: Unistallation 9
Figure 7: Driver Installation 9
Figure 8: Installation Type 9
Figure 9: Remove Driver 10
Figure 10: Camera Control 11
Figure 11: Main Window 12
Figure 12: Dropping Frames 12
Figure 13: Image Settings 13
Figure 14: Image ROI 13
Figure 15: Updating Image ROI 14
Figure 16: Image Flip 14
Figure 17: Brightness Control 14
Figure 18: Auto Brightness 15
Figure 19: Integration Time Setting 16
Figure 20: Gain Setting 16
Figure 21: White Balance 17
Figure 22: Camera Settings 17
Figure 23: Automatc Frequency Reduction 18
Figure 24: GPIO Settings 18
Figure 25: Trigger IO Control 19
Figure 26: Trigger Scheme 19
Figure 27: LED Control 20
figure 28: LED Target 20
Figure 29: LED Settings 21
Figure 30: Multiple Camera Mode 21
Figure 31: User Flash 22
Figure 32: User Strings 22
Figure 33: Camera Name 22
Figure 34: HDR Scheme 23
Figure 35: HDR Settings 24
Figure 36: I/O Port Schematic 29

1. Introduction

Visiosens GmbH has taken every possible care in preparing this manual. We, however, assume no liability for the content, completeness or quality of the information contained in this document. The content of this document is regularly updated and adapted to reflect the current status of the software. Furthermore, we do not guarantee that this product will function without errors, even if the stated specifications are adhered to.

Under no circumstances can we guarantee that a particular objective can be achieved with the purchase of this product.

Insofar as permitted under statutory regulations, we assume no liability for direct damage, indirect damage or damages suffered by third parties resulting from the purchase of this product. In no event shall any liability exceed the purchase price of the product.

Please note that the content of this manual is neither part of any previous or existing agreement, promise, representation or legal relationship, nor an alteration or amendment thereof. All obligations of visiosens GmbH result from the respective contract of sale, which also includes the complete and exclusively applicable warranty regulations. These contractual warranty regulations are neither extended nor limited by the information contained in this document. Should you require further information on this product, or encounter specific problems that are not discussed in sufficient detail in the manual, please contact your local visiosens distributor.

1.1 Safety Information

The product must be connected, taken into operation and maintained only by appropriately qualified personnel.

The error-free and safe operation of this product can only be ensured if it is properly transported, stored, set up and assembled, and operated and maintained with due care.

1.2 Operating Environment

Please comply with the requirements for the proper use of this product. Failure to do so will render the warranty void.

Do not subject this product to direct sunlight, moisture or shock. The environmental conditions specified in the relevant datasheet are required.

1.3 EMC Directives

Visiosens GmbH hereby confirms that this product has been developed, designed and manufactured in compliance with the EC Directive 89/336/EEC (Electromagnetic Compatibility). Compliance with the directives tested against following standards: EN 61000-6-2:2005, EN 61000-6-3:2001 + A11:2004

2. Installation

Please follow the next topics that describe the installation procedure.

2.1 System Requirements

- Minimum System Requirements
  - PC with Intel® or AMDprocessor (> 1.2 GHz)
    - restricted performance with Intel® Atom™ processors
  - 2 GB RAM
  - USB 2.0 Host Controller (EHCI)
- Supported Operating Systems
  - Microsoft Windows® XP with SP2 (32/64Bit)
  - Microsoft Windows® Vista (32/64Bit)
  - Microsoft Windows® 7 (32/64Bit)
- Further Software Requirements
  - VFU assembly (VFU.dll)
    - Microsoft .NET Framework 2.0
  - VFU Demo Application Example
    - Microsoft .NET Framework 3.5 with Service Pack 1

Optional requirements:

Requirements for GPU (OpenCL 1.0) accelerated functions

- *NVIDIA® Graphics cards*
  - GeForce® 8-series and later GPUs
    - Driver Release 265 (Version 266.58) and higher
  - Quadro® FX Series x700 and newer as well as the FX4600 and FX5600.
    - Driver Release 265 (Version 267.17) and higher
- *ATI Graphics cards*
  - ATI Radeon™ Series
    - AMD Catalyst Software Suite 11.2 and higher
  - ATI FirePro™ Series
    - ATI FirePro/GL/MV Software Suite 8.773.1.1 and higher
  - AMD FireStream™ Series
    - AMD Catalyst Software Suite 11.2 and higher
- *Intel Graphics cards currently do not support OpenCL.*

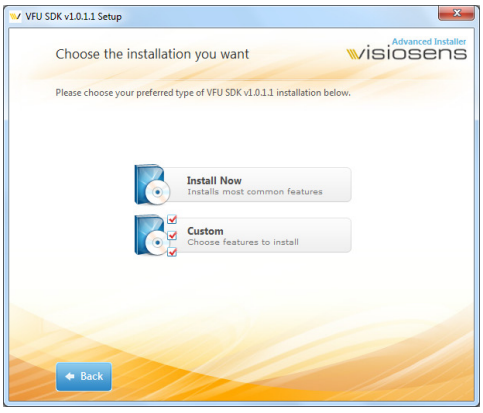
2.2 Installing VFU-Software on Windows© machines

To install the VFU-SDK please run the setup.exe file, provided in the archive file, and follow the procedure, described below, before plugging in the camera. Additional to the SDK the corresponding driver for 32bit or 64bit has to be installed. Please also make sure that there is no older driver version installed[1].

2.2.1 SDK

Click ->*Next*

Figure 1: Setup Wizard

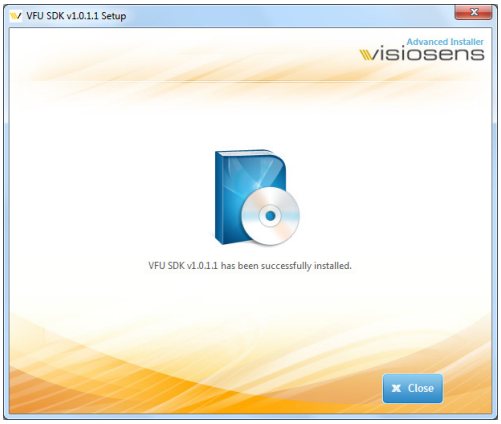


- Choose *Install Now* to use the default settings
- Choose *Custom* to define target folders and components to install

Figure 2: Installation Type

[1] If an older version is installed, please remove this manualy

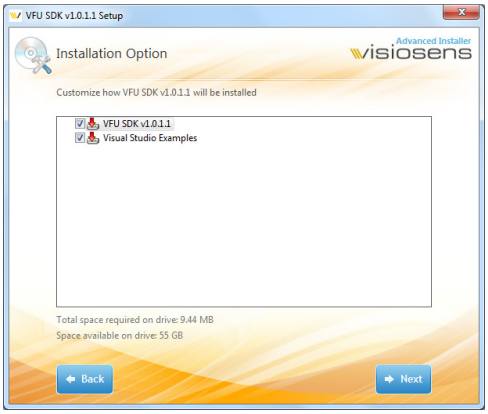
If chosen *Install Now*, the installation process will now be executed.



Click *Close* to exit the setup.

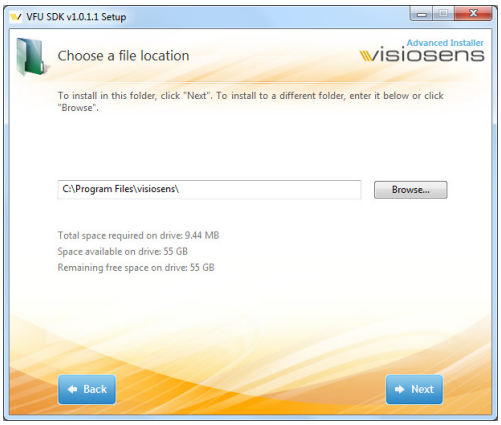
Figure 3: Successful Installation

If chosen *Custom*, please choose the components to be installed and



click ->*Next.*

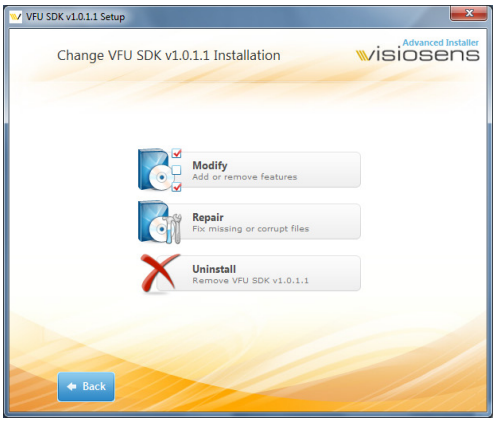
Figure 4: Custom Installation



Please choose an installation folder and click ->*Next*

Figure 5: File Location

To Uninstall the VFU-SDK, please run the setup and choose

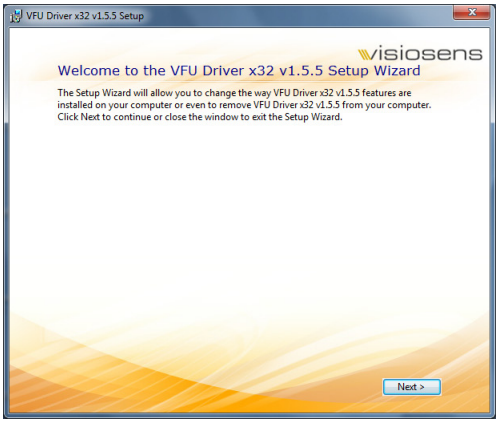


- *Modify* for Installation of single components
- *Repair* for recovery
- *Uninstall* to uninstall the VFU-SDK

Figure 6: Unistallation

2.2.2 Driver

Please start the VFUDriverx86_v1_5_5 installer to install the VFU-Driver package on a 32bit system or VFUDriverx64_v1_5_5 on a 64bit system.



Click ->*Next*

Figure 7: Driver Installation

- Choose *Typical* to use the default settings
- Choose *Custom* to define target folders and components to install

Figure 8: Installation Type

To Uninstall the VFU-Driver, please run the setup and choose *Remove*

Figure 9: Remove Driver

click ->*Next.*

2.3 Connecting a VFU Camera

Please install the software first as described in the “Installing VFU-Software on Windows machines” section. Connect the VFU Camera to the PC, using the USB 2.0 cable. It will be recognized automatically.

You can connect your VFU-Cameras to any USB 2.0 High-speed compatible USB port, either directly or via hubs and repeaters.

Please use only USB 2.0 High-speed certified cables or cables, provided by visiosens. Please also ensure that the USB port drives a current of minimum 200mA.

3. Demo-Application

The Demo-Application is designed to show the complete functionality of the VFU-Camera. This allows realizing a system specific setup within minutes. Moreover, the complete source code of the Demo-Application is provided within the SDK in order to minimize the customer's implementation workload to a minimum.

3.1 Starting the Demo-Application

When the Demo-Application is started, an overview list of connected cameras is given, as shown in figure Figure 10.

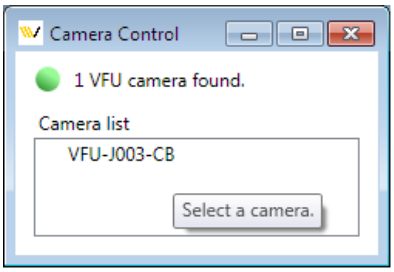


Figure 10: Camera Control

In this example one VFU-J003-CB is connected. If no VFU-Camera is connected, the control light will turn red. The overview list is updated every time a VFU-Camera is connected or disconnected.

Please note that cameras are differentiated by using a unique Camera Name. The usage is described in chapter 3.7.

To start the Main Window for the chosen camera, just click on the corresponding list item.

3.2 Main Window

The Main Window (s. Figure 11) is separated into four functional areas.

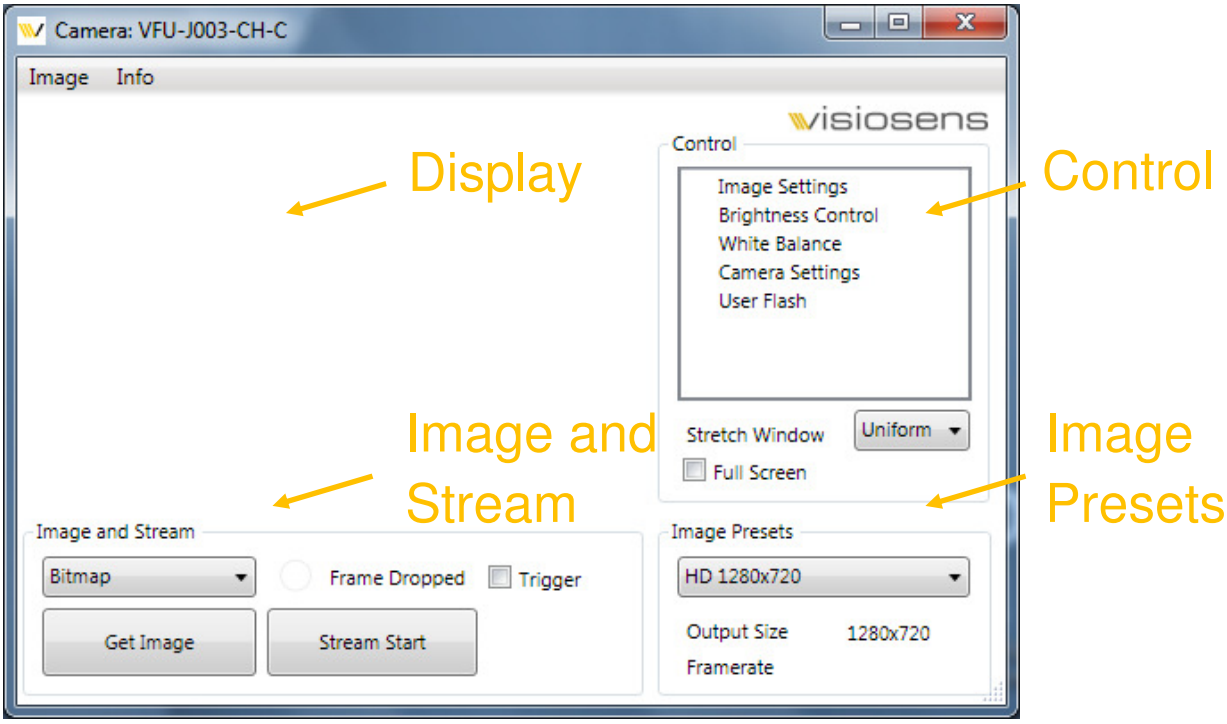


Figure 11: Main Window

1. Display
   In this area the Camera Images are displayed.

2. Image and Stream
   The Dropdown menu allows choosing the image type to be taken from the VFU-Camera.
   - Clicking the *Get Image* Button immediately requests an image from the VFU-Camera.
   - The *Stream Start* Button starts an image stream. This will run continuously until the *Stream Stop* is executed.
     - The Stream can either be free running (VFU-Camera is system master) or triggered (an external source is the system master).
     - If the *Trigger* checkbox is enabled, the stream is triggered. To test the functionality a software trigger can be initialized in this setting by clicking the test button.
     - The red indicator lamp will blink if the present bandwidth on USB2 is not enough to transport all images. In this case a frame will drop.

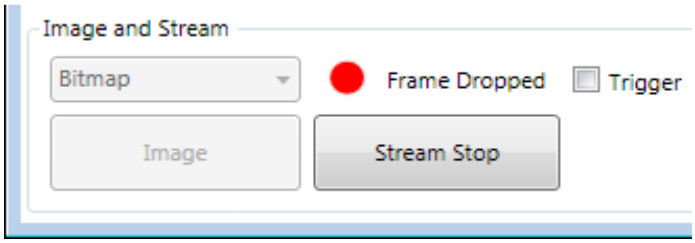


Figure 12: Dropping Frames

3. Image Presets
   The Image Presets allow choosing the image resolution as a *one click preset*. A detailed configuration can be made using the Image Settings in the Control functionalities.

4. Control
   The Camera Controls allow parameterizing the VFU-Camera parameters. To access the desired selection, please click on the corresponding list item. This will open a control window that contains the desired controls. These Windows are described in chapters 3.3 to 3.8.

3.3 Image Settings Window

The image settings (s. Figure 13) allow configuring the desired Region of Interest (ROI) that has to be read out from the camera and also the Image Orientation (Flip).

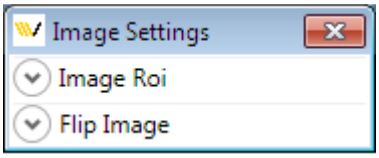


Figure 13: Image Settings

To access one of the topics, please click on the expand symbol.

In the figure Figure 14, a ROI of 2.560 (H) x 1.440 (V) pixel is set using a VFU-J003 with a native resolution of 10 Megapixel. In combination with the subsampling settings “binning” in horizontal (2x) and vertical (2x) direction, the resulting resolution of the output image is 1280x720 pixels.

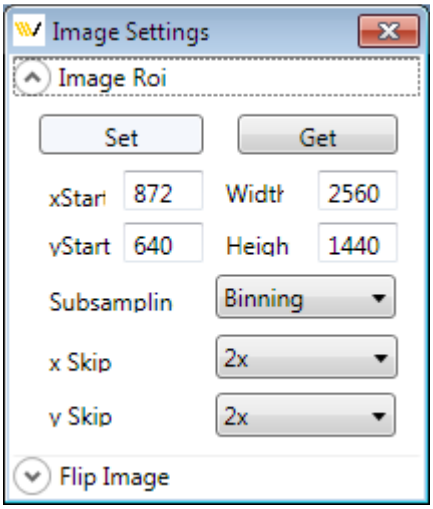


Figure 14: Image ROI

Depending on the camera hardware not all subsampling modes and -values are supported. The Drop-down menus contain just the supported modes.

If any changes have been made, the *Set*-Button[2] will switch to red (s. Figure 15). Click on *Set* in order to transmit the changes to the camera. If you want to readout the current settings use the *Get*-Button.

[2] Please notice, that setting a new ROI is only available when there is no streaming running.

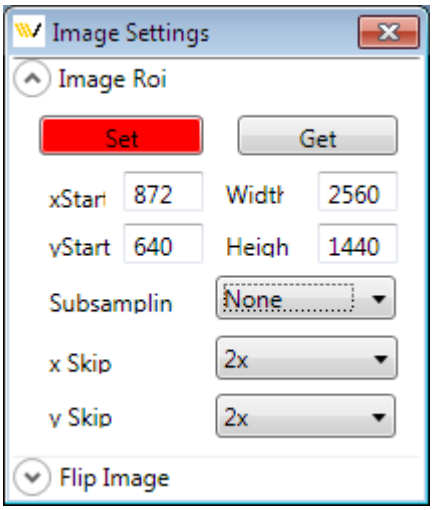


Figure 15: Updating Image ROI

In order to flip the output image in vertical and/or horizontal direction, check the corresponding item.

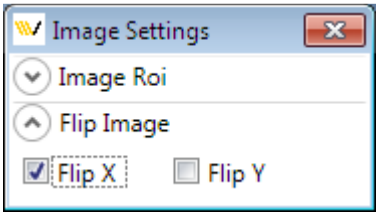


Figure 16: Image Flip

3.4 Brightness Control Window

The Brightness Control Window contains all necessary settings to control the image brightness. There are several ways to control the brightness of an output image.

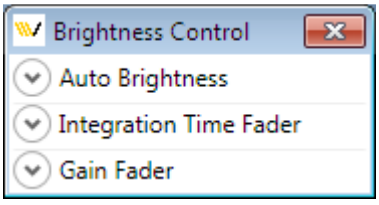


Figure 17: Brightness Control

3.4.1 Auto Brightness

As default the Auto Brightness functions are enabled. This ensures that images are correctly illuminated. Depending on the VFU-Camera type several different settings are available. This can be “Hardware” (not available on all types) and “Software”. The figure below shows a Software setting for both Auto Integration Time and Auto Gain control.

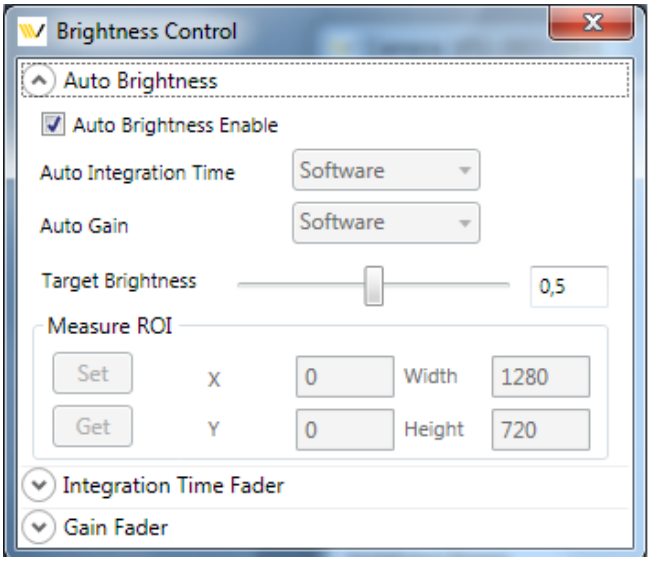


Figure 18: Auto Brightness

Several further options can be set in order to adapt to the target. In detail these are:

- Target Brightness
  The Target Brightness allows setting the mean value of the output image in terms of percent. This means if Target Brightness is set to 0.5 the mean value of the output image will be 128 (based on an 8-bit image range) or 512 (based on an 10-bit image range).

- Measure ROI
  Some application requires controlling of the image brightness using just a detail of the complete image. E.g. if on a high contrast image on detail has to have a defined brightness, brightness and other details are allowed to be dark or to be saturated.
  As default the Measure ROI is set to the complete image. If just a detail has to be used, set the corresponding ROI parameters and click *Set*. Use *Get* to get the current settings.

Please note that for *Get Image* the Auto Brightness will not be able to set the Target Brightness for the first image. The Auto Brightness will need several images to adapt to the scenario. During this the scenario must not change its brightness!

3.4.2 Integration Time

The Integration Time Fader allows setting the Integration Time of the image manually. This is useful if the Integration Time has to have a dedicated value in order to synchronize with target requirements or to avoid flicker on scenes that are illuminated with halogen lamps.

This function can also be used in combination with Auto Gain Control.

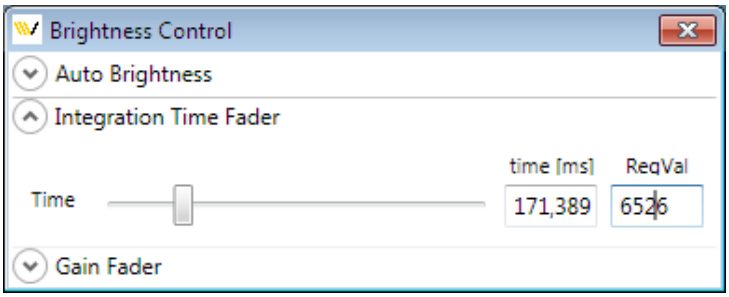


Figure 19: Integration Time Setting

The Integration Time is displayed in ms and also in Register Values of the VFU-Camera. Please use the *PageUp / PageDown* buttons for the fine tuning of the fader.

3.4.3 Gain

Depending on the type of VFU-Camera different Gain Controls are available. The figure below shows the Controls for a VFU-J003. There a Global Gain and Gains for the single color channels are present.

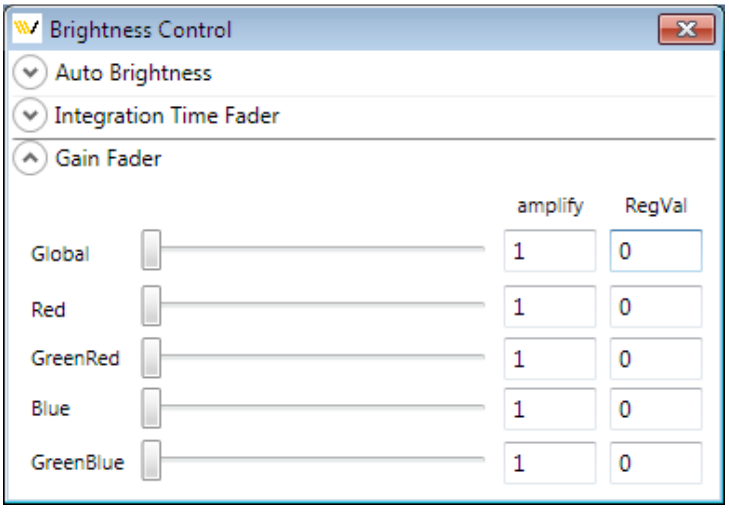


Figure 20: Gain Setting

The Gains are displayed in amplification factors and also in Register Values of the VFU-Camera. Please use the “PageUp” / “PageDown” buttons for the fine tuning of the fader.

3.5 White Balance Window

A Software White Balance can be applied to the image if the checkbox *White Balance Enabled* is checked (s. Figure 21). The default parameters for the color amplification factors are 1.

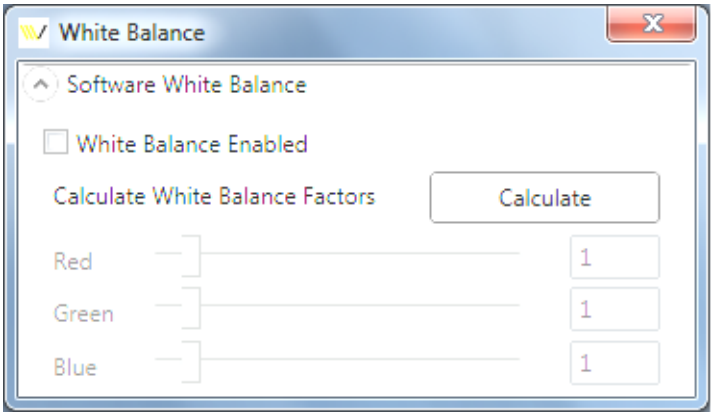


Figure 21: White Balance

To calculate the color amplification factors for the actual illumination scenario please click *Calculate*. The factors are updated immediately. Also a manual White Balance can be applied by changing the factors using the corresponding sliders.

The White Balance is computed either on the CPU or the GPU, depending on the system components (s. chapter 2.1) and the settings for the Image Processing (s. chapter 4.4).

3.6 Camera Settings Window

The Camera Settings allow setting all Hardware Parameters of the VFU-Camera.

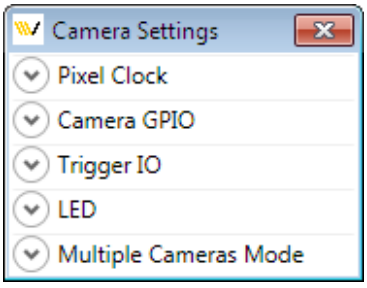


Figure 22: Camera Settings

3.6.1 Pixel Clock

The VFU-Cameras allow controlling the Pixel Clock Frequency of the image sensor. This enables to either increase the Frame Rate to a system specific maximum or to slow down the system in order not to stall the system resources and to decrease noise effects on the output images. Please note that the Integration Time has to be updated when the Pixel Clock Frequency was changed.

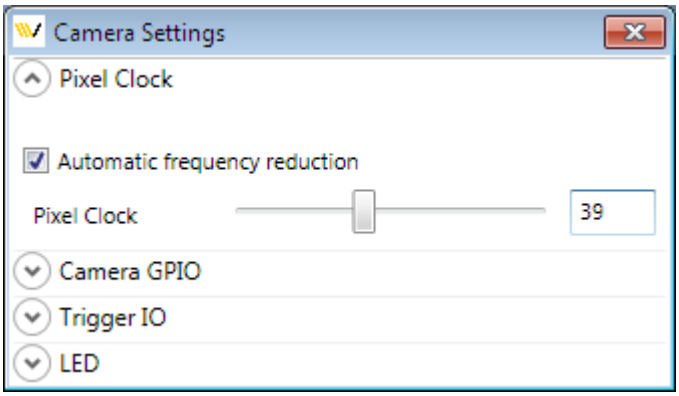


Figure 23: Automatc Frequency Reduction

When “Automatic Frequency Reduction” is enabled, the VFU-Camera decreases the Pixel Clock to a level that allows transmitting Image Data at the maximum available bandwidth and without losing frames.

Please note: If the Pixel Clock Frequency was changed the Integration Time has to be updated manually when Auto Brightness is not used (s. chapter 3.4.2).

3.6.2 Camera GPIO

The VFU-Camera series is equipped with four General Purpose Inputs/Outputs. Two of them are inputs, two are outputs.

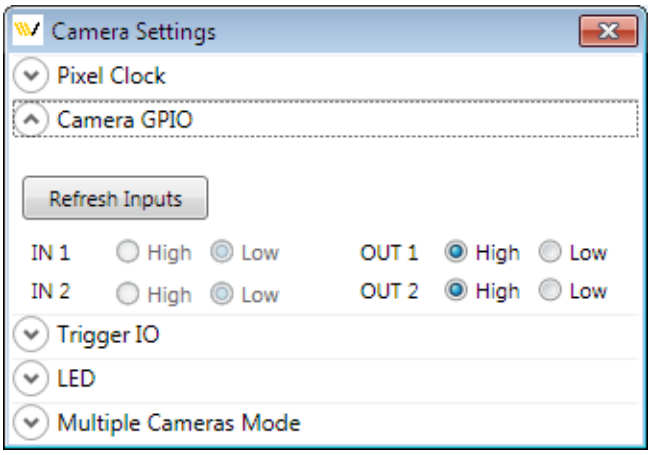


Figure 24: GPIO Settings

The *Refresh Inputs* button reads the current values of the inputs from the VFU-Camera.

The Outputs are set immediately after any change has been made on the check-items.

3.6.3 Trigger IO

Additionally to the GPIO the VFU-Camera series allows to synchronize the image exposure with other system components using the Trigger Input/Output functionality.

A trigger can be executed either on the rising or falling edge of the Trigger Input pin.

The Trigger Delay allows delaying the start of exposure in relation to the trigger event.

The Strobe Delay and Strobe Duration controls allow to define a Strobe Out signal on the corresponding pin with the delay in relation to the start of exposure and its duration.

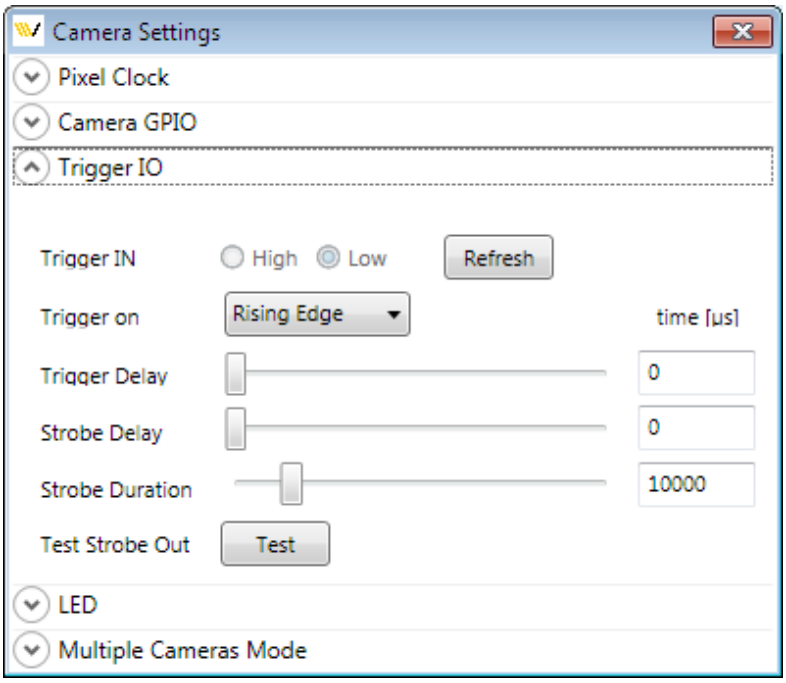


Figure 25: Trigger IO Control

The figure below shows a typical trigger scenario, starting on the rising edge of the Trigger In.

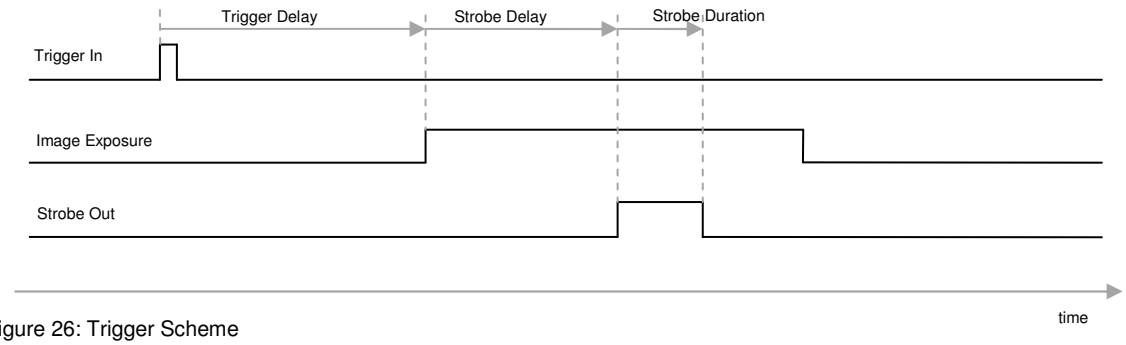


Figure 26: Trigger Scheme

The Test Button enables to check the Strobe Out functionality from the Demo Application without requiring a physical trigger source.

3.6.4 LED

The VFU-Camera series contains up to two 4-Chanel LED-Controllers. These allow controlling the four channels independently.

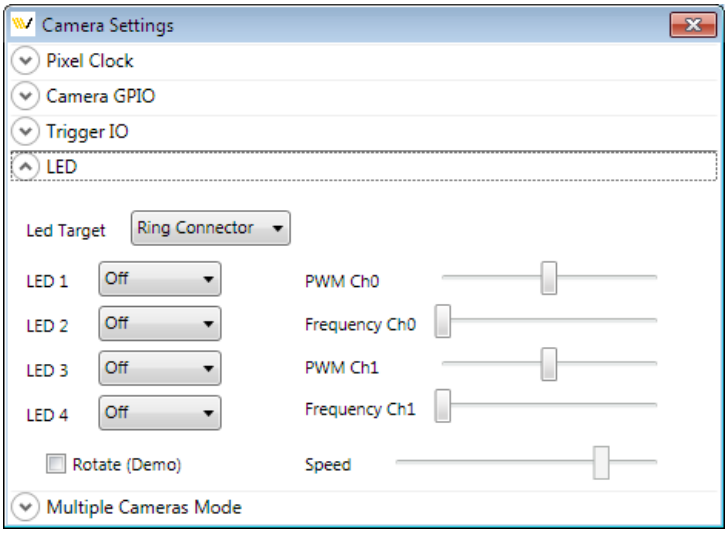


Figure 27: LED Control

Depending on the type of connected VFU-Camera, different LED Targets are available. These are the Ring-Connector for the Front-Site LED-Ring and the Backside Connector.

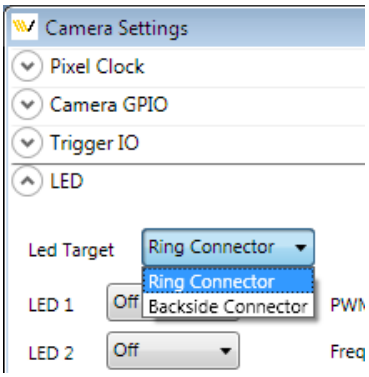


figure 28: LED Target

Beyond the possibility of turning the LED on or off, also an intensity regulation based on a Pulse Width Modulation (PWM) is available. There are two PWM available that can be controlled in base frequency (Frequency Chx) and in Pulse Width (PWM Chx)

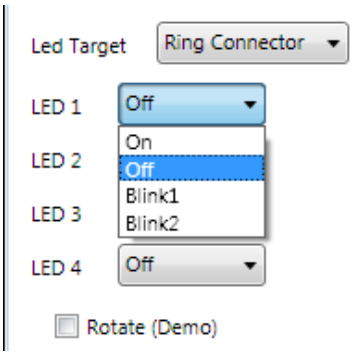


Figure 29: LED Settings

As a little demo also a rotating light for the Ring Light is available. To enable this demo check the Rotate option.

3.6.5 Multiple Cameras Mode

When using multiple VFU-Cameras, two approaches can be used:

- Synchronous: In this mode all Cameras are synchronized to each other. This is useful if all cameras shall have the same priority. Here one image is transferred from one camera, then one image from the second camera and so on.
- Asynchronous: In this mode images are transferred at the moment they are present in the camera.

  Advantages of the asynchronous mode:

  - Individual start of integration for each camera
  - Different trigger scenarios for each camera (e.g. one camera is triggered by an external event, another camera is in free running mode)

  Disadvantages of asynchronous mode:
  - When not enough bandwidth available, frames may drop.
  - Asynchronous Mode should only be used if each camera is connected to its own USB-Host-Controller.

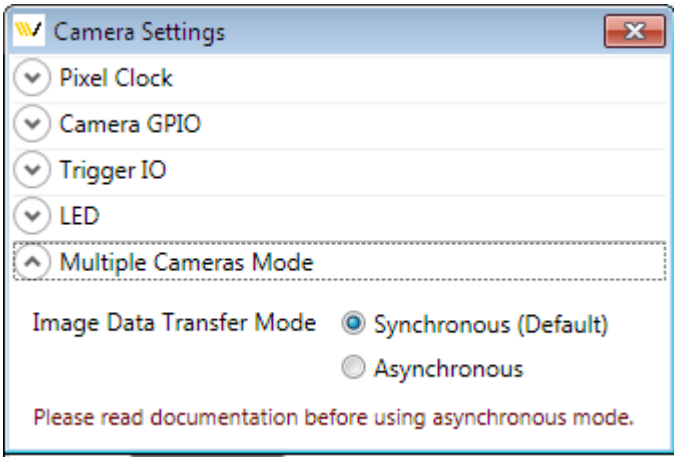


Figure 30: Multiple Camera Mode

3.7 User Flash

The VFU-Camera series supports different User Flash functionality for storing user data in a nonvolatile Flash-RAM.

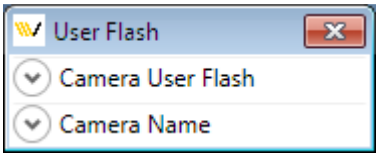


Figure 31: User Flash

The four User Strings can be accessed by the user by using “get” to read the corresponding string from cameras Flash-RAM and using “set” to write the string from the corresponding text field into the Flash-RAM.

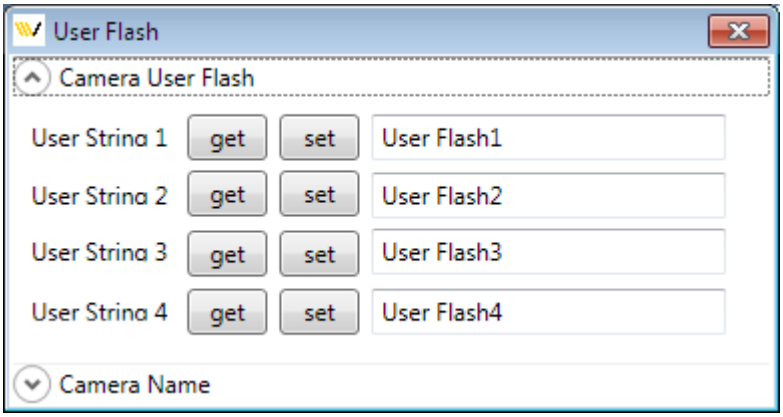


Figure 32: User Strings

Additional to the four User Strings also a Camera Name can be given, in order to identify the connected camera by its name.

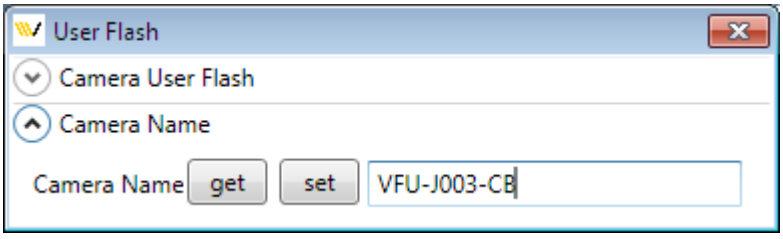


Figure 33: Camera Name

3.8 HDR

Depending on the type of the image sensor, several VFU-Cameras support a so-called High-Dynamic-Range (HDR) Modus. The HDR modus allows acquiring images with very high contrasts inside a scene.

To enable the HDR check Enable HDR. Auto Adjust controls the timing for the integration time of the single steps automatically. The number of Knees allows choosing the number of steps of integration. One Knee corresponds with two steps, two Knees with tree steps.

In Auto Adjust Mode the relative settings Knee1 / Knee 2 can be defined by setting V1/V2, T2/T3.

In Manual Mode all parameters have to be set individually and depending on the image scene. Also the Auto Brightness functionality is not available then. So it's highly recommended to use the Auto Adjust Mode.

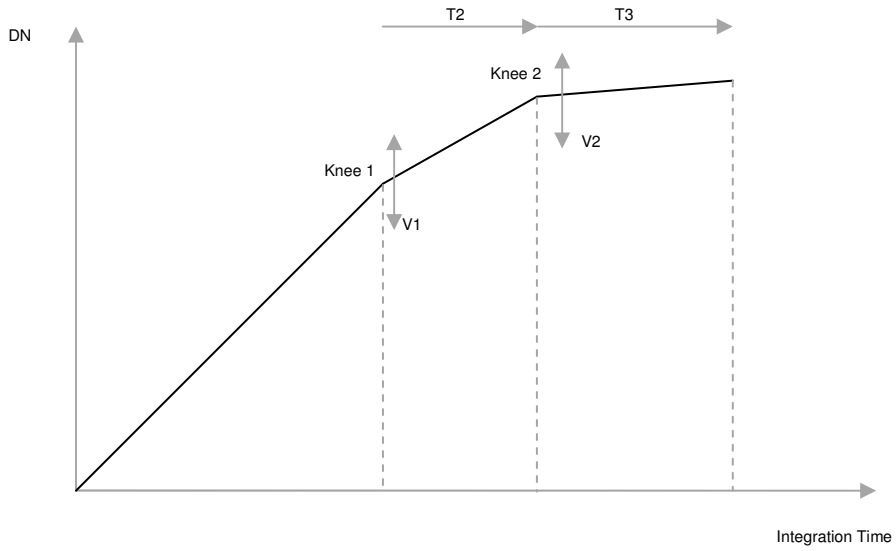


Figure 34: HDR Scheme

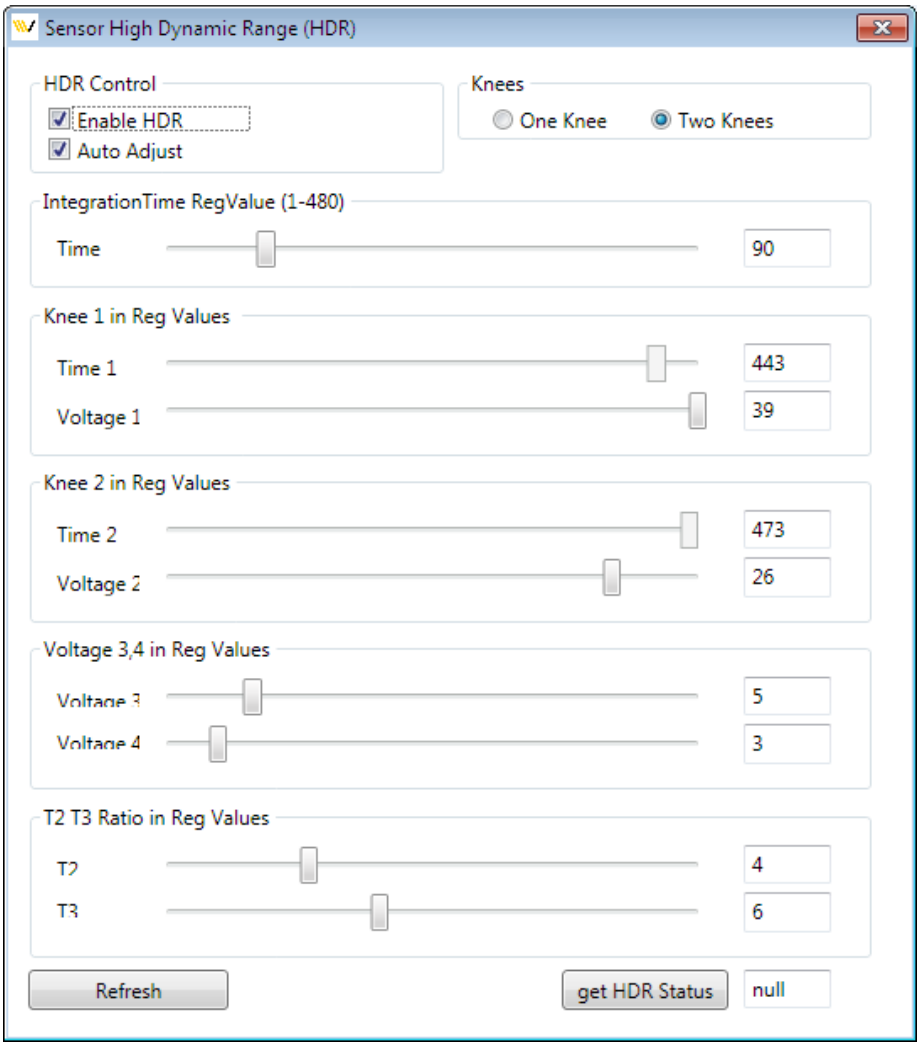


Figure 35: HDR Settings

4. Software Development Kit

4.1 Philosophy

Our philosophy is to enable you to realize your application with a minimum amount of development time. To be also independent of operation system structure, the SDK for VFU-Series is based on the DotNet-Framework for Windows© machines and the equivalent Mono-Framework for Linux. This allows changing your application to a different operation system, even when your application development is already done.

A major new feature is the so-called Self Describing Interface. This interface takes care of all camera specific parameters in a way that the Software Engineer don't have to take care about parameters such as e. g. available image sizes or gain settings. All the specific info can be requested by an equivalent info method. When using this interface, the Software Engineer can easily exchange the camera hardware, e.g. change to another image sensor without changing his implementation!

A complete implementation example is provided with the Demo Application. Please feel free to reuse this source code, but please notice that it is provided as is and visiosens cannot give any warranty of function for this.

A detailed description of all methods can be found in the XML-Documentation installed together with the Software Development Kit.

4.2 SDK Configuration

The behavior of VFU.dll can be configured using the DotNet Framework. Each application can contain its own schematic XML code. The corresponding configuration file must be located in the application folder of the corresponding application and must be named like the application itself and must have the ending *.config*.

Example:

- *Application: VFUWpfDemoApp.exe*
- *Corresponding configuration file: VFUWpfDemoApp.exe.config*

The configuration file can be created by using any text editor like e. g. Notepad. As default there is no configuration file, it has to be created during the implementation in the customers application.

The example below shows the minimum level implementation of the configuration file.

```
<?xml version="1.0"?>
<configuration>
</configuration>
```

The configuration of VFU.dll is possible within a section `<VFUDLL></VFUDLL>` within the section `<configuration></configuration>`. In this case, the implementation looks like:

```
<?xml version="1.0"?>
<configuration>
<!-- Register a section handler for the VFUDLL section -->
<configSections>
<section name="VFUDLL" type="System.Configuration.IgnoreSectionHandler"/>
</configSections>
<VFUDLL>
</VFUDLL>
</configuration>
```

Within the section <VFUDLL></VFUDLL> the section features given below can be integrated:

Configuration category	Section	Description
Logging	**<Logging> </Logging>**	**Logging-Function of VFU.DLL enable / disable and Logging-Parameters**
Image Processing	**<ImageProcessingManager> </ImageProcessingManager>**	**Configuration of internal Data- and Image-Processing**

Table 1: SDK configuration

Each category itself contains its own sections.

4.3 Logging

The Logging functionality is designed to simplify support and maintenance of the VFU-SDK in combination with the customer's application. During the operation the VFU.dll creates so-called *Logging files.* The names of these files depend on the name of the application that uses the dll. As an example the application *VFUWpfDemoApp.exe* will cause logging files that are named *Log_VFUWpfDemoApp.txt* and *Log_VFUWpfDemoApp.bin*.

The logging behavior of the VFU.dll can be configured by using

<Level value="…" />

If Logging shall be used, the element have to be configured like given below (this is the default configuration)

```
<Level value="all" />
```

If Logging has to be disabled, the element should be set to:

```
<Level value="off" />
```

Disabling the Logging functionality should be done only when the development process has been completed.

The element

<AppendToFilevalue="…" />

Defines, whether Logging files will be overwritten on every start of the application or whether files shall be append with new data (file size will increase!). The default configuration is

```
<AppendToFilevalue="true" />
```

for appending new data. For overwriting, the element should look like:

```
<AppendToFilevalue="false" />
```

The parameter

<MaximumFileSizeKBvalue="…" />

defines the maximum size for Logging files. If this size is exceded, Logging files will either be deleted or split into several files (backups) depending on the configuration of the element <MaxSizeRollBackups />. The maximum size is given in Kilobytes (KB). If the maximum has to be 100KB, the element should be set to:

```
<MaximumFileSizeKBvalue="100" />
```

As Default, 1024KB are used.

The parameter

<MaxSizeRollBackupsvalue="…" />

Defines the maximum number of Logging files. If one file with <MaximumFileSizeKB/> will exceded its defined size, the Logging will be continued with a new file. The files are numbered with increasing numbers within the file-ending.

The element <MaxSizeRollBackups/> defines how many additional files can be created. When this element is set to 0, the file will be deleted after exceding the maximum size and afterwards a new file will be created. If max. two backup files shall be used, the element has to be set to:

```
<MaxSizeRollBackupsvalue="2" />
```

This is the standard configuration.

4.4 Image Processing

The Image Processing inside the VFU.dll is responsible for Data- and Image-Processing-Operations. This operation can be executed either on the Host-CPU or on the Graphic-Card-Processor (GPU). The usage of the GPU will decrease the CPU-load. GPU processing will be automatically used if the graphic-card supports OpenCL™ 1.0.

The GPU usage can be defined using:

<OperationDeviceFilter>

If the GPU shall not be used, the following configuration has to be used:

```
<OperationDeviceFilter>
<IgnoreDevice type="opencl"/>
</OperationDeviceFilter>
```

5. Camera Hardware

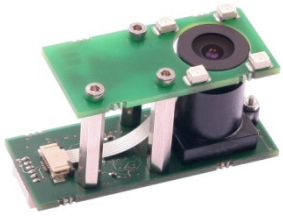
5.1 Family overview

The VFU-Camera-Series contains a wide range of different versions for each supported image sensor. These versions are:




VFU-xxxx-xB	VFU-xxxx-xB-M12	VFU-xxxx-xB-M12 + Lxxx	VFU-xxxx-xB-C /-CS
- Boardlevel version - prepared for mount - various I/O	- Boardlevel version - prepared for mount - various I/O - M12 mount (S-Mount)	- Boardlevel version - prepared for mount - various I/O - M12 mount (S-Mount) - LED-Ring (different wavelength available)	- Boardlevel version - prepared for mount - various I/O - C-Mount (with filter option) - CS-Mount (with filter option)



VFU-xxxx-xS-C / -CS	VFU-xxxx-xH-C /CS	VFU-xxxx-xH-M12	Custom Name Convention
- Boardlevel version - prepared for mount - various I/O - USB-Mini Connector - C-Mount (with filter option)	- Housed version - prepared for mount - various I/O - USB-Mini Connector - Mounting Options: - C-Mount - CS-Mount - M12-Mount (LED-Ring)	- Housed version - prepared for mount - various I/O - USB-Mini Connector - Mounting Options: - M12-Mount (LED-Ring)	Customized versions, please contact your local sales representative for further information

5.2 Available sensors

The sensors listed below are currently available in VFU-Camera-Series:

MT9V024	mono / RGB	Wide VGA, 1/3”, Global Shutter
MT9M033	mono / RGB	1.2MPixel, 1/3”, Rolling Shutter
MT9P031	mono / RGB	5MPixel, 1/2,5”, Rolling Shutter
MT9J003	mono / RGB	10MPixel, 1/2,3”, Rolling Shutter

table 2: Available sensors

5.3 I/O-functionality

I/O Port Connectivity:

Pin	I/O	connector
1	Trigger_IN	- Molex Picoblade 51021-0800 - Binder M12 Series
2	IN1	
3	IN2	
4	OUT1	
5	GND INx/OUTx	
6	OUT2	
7	Sensor Strobe f. Flashlight	
8	GND f Flashlight	
Inputs IN1 and IN2 are opto-coupled.		
Outputs OUT1 and OUT2 are opto-coupled (open collector). Please note that an extern Pullup-Resistor is necessary for operation.		

Table 3: I/O Port Connectivity

I/O Port Schematic:

Camera: User Circuit:

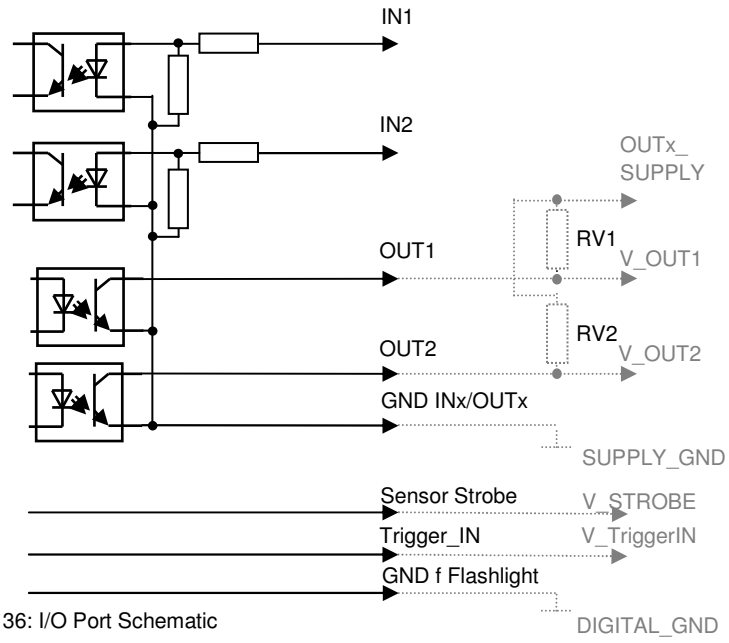


Figure 36: I/O Port Schematic

Electrical Characteristics I/O Port:

Symbol	Definiton	Condition	Min.	Typ.	Max.	Unit
V_IN1, V_IN2	Input Voltage Range on IN1, IN2		0	5	12	V
V_IH	Input HIGH Voltage on IN1, IN2		1,8	5	12	V
V_IL	Input LOW Voltage on IN1, IN2		0	0	0,5	V
I_VIN	Input Current on IN1, IN2	at V_INx = 1,8V at V_INx = 5V at V_INx = 12V		1,2 7 19		mA
OUTx_SUPPLY	Supply Voltage for Open Collector Outputs		0	5	24	V
RV1, RV2	Series Resistor		1,4	5,6		KΩ
V_OUT1, V_OUT2	Open Collector Output Voltage	at OUTx_SUPPLY=5V, RV=5,6KΩ at OUTx_SUPPLY=10V, RV=5,6KΩ at OUTx_SUPPLY=24V, RV=5,6KΩ	0,17 (L) 0,23 (L) 0,58 (L)		5 (H) 10 (H) 24 (H)	V
V_STROBE	Strobe Out Voltage		<0,75	2,9	3	V
I_STROBE	Strobe Out Current		0		10	mA
V_TriggerIN	Trigger In Voltage	 (H) (L) Hysteresis (H)<->(L)	-0,5 2,1 0	 60	3,8 3,8 0,8	V mV
I_TriggerIN	Trigger Input Current			1		µA

Table 4: Electrical Characteristics I/O Port

Caution: Stresses greater than those listed may cause permanent damage to the camera.

Caution: Do not connect GND Inx/OUTx with GND f Flashlight.