DECTRIS

User Manual

DECTRIS POLLUX®

Document Version 1.0.3

DECTRIS Ltd.

5405 Baden-Daettwil, Switzerland

www.dectris.com

TABLE OF CONTENTS

1. GENERAL INFORMATION

1.1. Contact and Support

Address	**DECTRIS Ltd** Taefernweg 1 5405 Baden-Daettwil Switzerland	**DECTRIS USA Inc.** 1500 Walnut Street, Suite 1630 Philadelphia, PA 19102 USA	**DECTRIS Japan K.K.** Nakagawa building 801 3-37 Higashi-Nobusue Himeji-shi Hyogo 670-0965 Japan
Phone	+41 56 500 21 02	+1 267 924 5179	+81 79 280 9585
Website	www.dectris.com		
Email	support@dectris.com		

1.2. Explanation of Symbols

DANGER
Danger blocks are used to indicate immediate danger or risk to personnel or equipment.

WARNING
Warning blocks are used to indicate danger or risk to personnel or equipment.

CAUTION
Caution blocks are used to indicate danger or risk to equipment.

NOTICE
Information blocks are used to highlight important information.

1.3. Warranty Information

Should your detector require warranty service, contact DECTRIS® for further information. Before shipping the system back, please contact DECTRIS® to receive the necessary transport and shipping information. Make sure that the original packaging is used when returning the system.

CAUTION
Do not ship the system back before you receive the necessary transport and shipping information.

CAUTION
Opening the detector or the power supply housing without explicit instructions from DECTRIS® will void the warranty.

1.4. Disclaimer

DECTRIS® has carefully compiled the contents of this manual according to the current state of knowledge. Damage and warranty claims arising from missing or incorrect data are excluded.

DECTRIS® bears no responsibility or liability for damage of any kind, also for indirect or consequential damage resulting from the use of this system.

DECTRIS® is the sole owner of all user rights related to the contents of the manual (in particular information, images or materials), unless otherwise indicated. Without the written permission of DECTRIS® it is prohibited to integrate the protected contents in this publication into other programs or other websites or to use them by any other means.

DECTRIS® reserves the right, at its own discretion and without liability or prior notice, to modify and/or discontinue this publication in whole or in part at any time, and is not obliged to update the contents of the manual.

In the case that a translation of this manual is provided for convenience, the English version shall remain as the single source of truth. Any conflict between a translated version and the English original will be resolved in favor of the English text.

2. SAFETY INSTRUCTIONS

CAUTION
Please read these safety instructions before operating the detector system.

- Before turning the power supply on, check the supply voltage against the label on the power supply. Using an improper main voltage will destroy the power supply and damage the detector.
- Power down the detector system before connecting or disconnecting any cable.
- Make sure the cables are connected and properly secured.
- Avoid pressure or tension on the cables.
- Operating the detector outside of the specified conditions could damage the system.
- The detector is not specified to withstand direct beam at a synchrotron. Such exposure will damage the exposed pixels.
- Place the protective cover on the detector when it is not in use to prevent the detector from accidental damage.
- Do not install additional software or change the operating system of the Detector Control System.
- Do not touch the front window of the detector or the sensor surface.

3. SYSTEM DESCRIPTION

3.1. Components

The DECTRIS POLLUX® detector system consists of the following components:

- POLLUX detector head
- POLLUX Detector Control System (DCS)
- Power supply unit (PSU)
- One USB-C cable per POLLUX detector head
- Ethernet cable
- Power cable

3.2. Basic Functionality

DECTRIS® X-ray detectors provide direct detection of X-rays with optimized solid-state sensors and CMOS readout ASICs in hybrid pixel technology. Well-proven standard technologies are employed independently for both the sensor and the CMOS readout ASIC. The POLLUX hybrid pixel detector is composed of a sensor, which is a two-dimensional array of photodiodes or photoconductors processed in a high-resistivity semiconductor, connected to an array of readout channels designed in advanced CMOS technology.

The POLLUX detectors operate in single-photon counting mode and provide outstanding data quality. They feature high dynamic range, zero dark signal and zero readout noise and hence achieve optimal signal-to-noise ratio at short readout time and high frame rates.

3.2.1. Key advantages

- Direct detection of X-rays
- Single-photon counting
- Excellent signal-to-noise ratio and high dynamic range (zero dark signal, zero readout noise)
- Two energy discriminating thresholds
- Continuous readout with zero dead time and high frame rates
- Shutterless operation
- Full factory calibration in the specified energy range

3.3. Continuous Readout

POLLUX detectors incorporate a continuous readout feature, which facilitates high frame rates and minimizes dead time between frames. Each pixel of the ASIC (Figure 1) is equipped with two independent digital counters, allowing for instantaneous switching during photon count acquisition. Upon completion of a frame acquisition, the pixels seamlessly switch from using one digital counter to the other. This design ensures that data acquisition continues in one counter while the other is being read out.

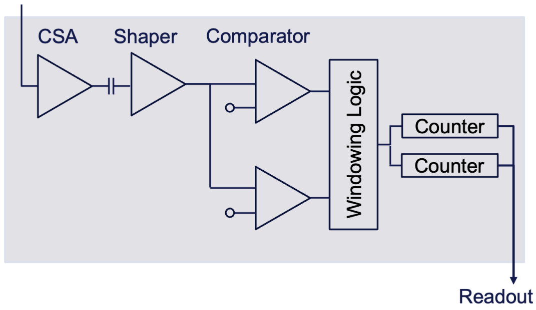


Figure 1. Schematic representation of the main building blocks of a pixel of the POLLUX ASIC. CSA: Charge-sensitive amplifier.

3.4. Auto-Summation

Auto-summation is a further benefit of continuous readout in POLLUX detectors. While a single frame is limited to 12 bit from the digital counter, auto-summation extends the data depth up to 32 bit, or more than 4.2 billion counts per pixel, depending on the number of summed frames in an image.

To record images with a certain exposure time (`count_time` parameter in the SIMPLON API), the detector firmware will operate the ASIC at the highest possible frame rate such that summing of N frames acquired with a "frame count time" (`frame_count_time` parameter in the SIMPLON API) will result in an image with the requested exposure time (`N * frame_count_time= count_time`). The count rate correction is performed on the individual frames before summation, resulting in a more accurate count rate correction for intensities that are not constant during the exposure time of an image.

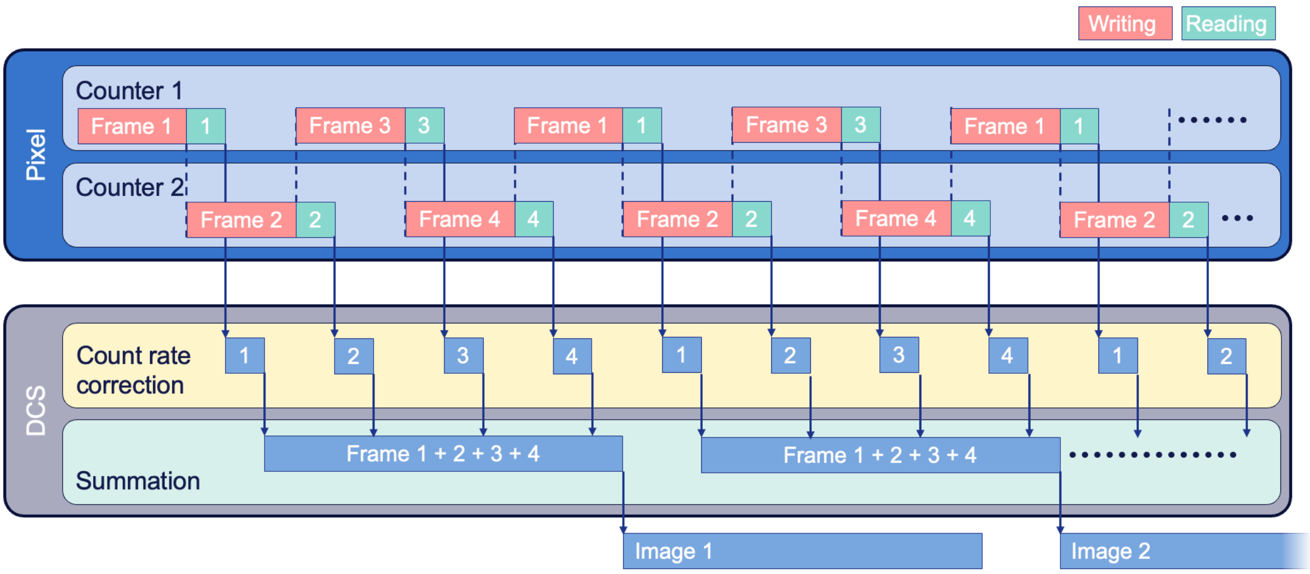


Figure 2. Example and schematic illustration of continuous readout and auto-summation. In this example, each image acquired with the detector is composed of 4 frames recorded in continuous readout. After the acquisition of frame 1 in counter 1, the writing of photon counts continues in counter 2 for frame 2, while the detector control system (DCS) reads out frame 1 from counter 1 and performs the count rate correction of this frame. After acquisition of frame 2 in counter 2, the pixel again switches to counter 1 for recording frame 3, while the DCS reads out frame 2 and performs the count rate correction. After acquisition of frame 4 of image 1, the pixel continues to acquire frame 1 of image 2 while the DCS outputs image 1 that is composed of 4 individually count rate corrected frames.

4. TECHNICAL SPECIFICATIONS

4.1. Specifications

4.1.1. Available Detector Models

The POLLUX detectors come in a variety of configurations. Customers can choose between two active area sizes: **POLLUX** (14.4 mm × 19.2 mm) and **POLLUX PANORAMA** (14.4 mm × 57.9 mm).

Furthermore, two pixel geometries are offered: square 75 µm × 75 µm pixels, and rectangular 37.5 µm × 150 µm pixels. The rectangular pixel option, **POLLUX PANORAMA HR**, provides higher 1D spatial resolution along the long axis of the active area due to its smaller pixel pitch in that direction, compared to the standard square pixel geometry.

4.1.2. Detector Specifications

Table 1. List of detector specifications

	POLLUX	POLLUX PANORAMA	POLLUX PANORAMA HR
Pixel size	75 µm x 75 µm	75 µm x 75 µm	37.5 µm x 150 µm
Pixel Array (W x H)	256 px x 192 px = 49'152 px	772 px x 192 px = 148'224 px	1544 px x 96 px = 148'224 px
Active Area (W x H)	19.2 mm x 14.4 mm = 276.5 mm^2	57.9 mm x 14.4 mm = 833.7 mm^2	
Detector head dimensions (W x H x D)	29.2 mm x 62 mm x 36.5 mm	67.9 mm x 62 mm x 36.5 mm	
Detector head weight	130 g	275 g	
Sensor material	Silicon		
Sensor thickness	320 µm		
Photon energy	4.5 - 9.4 keV		
Calibrated threshold range	2.7 - 12.0 keV		
Image bit depth	32 bit		
Readout bit depth	12 bit		
Maximum count rate[1]	1 x 10^6 photons/s/pixel		
Number of thresholds	2		
Energy resolution (FWHM @8 keV)	< 600 eV		
Readout time	continuous readout with zero deadtime		

	POLLUX	POLLUX PANORAMA	POLLUX PANORAMA HR
Maximum frame rate		2D: 360 Hz[2] 1D: 360 Hz 0D: 360 Hz	
Point-spread function		1 pixel (FWHM)	
Detector Control System dimensions (W x H x D)		141.0 mm x 44.5 mm x 172.4 mm	
Detector Control System weight		0.7 kg	
Overvoltage category		II	
Means of protection		Class I	
Pollution degree		II	
Maximum operating altitude		2000 m a.s.l.	

[1]Achieved in single threshold mode.

[2]360 Hz with 1 or 2 POLLUX heads or 1 POLLUX PANORAMA. Reduced to 90 Hz with 2 PANORAMA heads.

4.2. Ratings

4.2.1. Detector Head

Table 2. Detector head power ratings

	POLLUX	POLLUX PANORAMA
Input voltage	+9 V DC	
Input power	2.5 W	5 W

4.2.2. Detector Control System (DCS)

Table 3. DCS ratings

DCS power input	+12 V DC, 20 W
External trigger input	High level: 2.1 - 5.0 V Low level: 0.0 - 0.8 V **CAUTION** Do not exceed 5 V input. Higher voltages will damage the Detector Control System (DCS).

External trigger input impedance	Input impedance: 47 kΩ Connector: LEMO® Type 00
Trigger output	High level: 2.16 - 3.3 V Low level: 0.0 - 0.64 V Max. current: 16 mA Impedance: 40 Ω Connector: LEMO® Type 00

4.2.3. Power Supply Unit

NOTICE
Please consult the user documentation of the Mean Well GSM40A12-P1J power supply unit for details.

CAUTION
Only use the power supply delivered with the POLLUX system.

Table 4. Power Supply Unit Ratings

Power supply unit power input	100 - 240 VAC (±10%), 50/60 Hz, 1A max.
Power supply unit power output	12 V DC, 3.34 A max., 40 W max.
AC connector	IEC-320-C14 input inlet
Case dimensions (W x H x D)	123 mm x 50 mm x 31.5 mm
Weight	290 g

4.3. Ambient Conditions

The POLLUX detector is designed for indoor use. The ambient conditions shown in Table 5 must be satisfied.

Table 5. Ambient conditions for detector operation and storage

Ambient condition	Value
Temperature range during operation	+18 °C to +35 °C
Extended temperature range during operation **NOTICE** When operating in the extended operating temperature range, the data quality may be impacted.	+18 °C to +40 °C
Humidity during operation	< 80 % at 20 °C, non-condensing

Ambient condition	Value
Storage temperature	+15 °C to +40 °C
Storage humidity	< 40 % at 20 °C, non-condensing

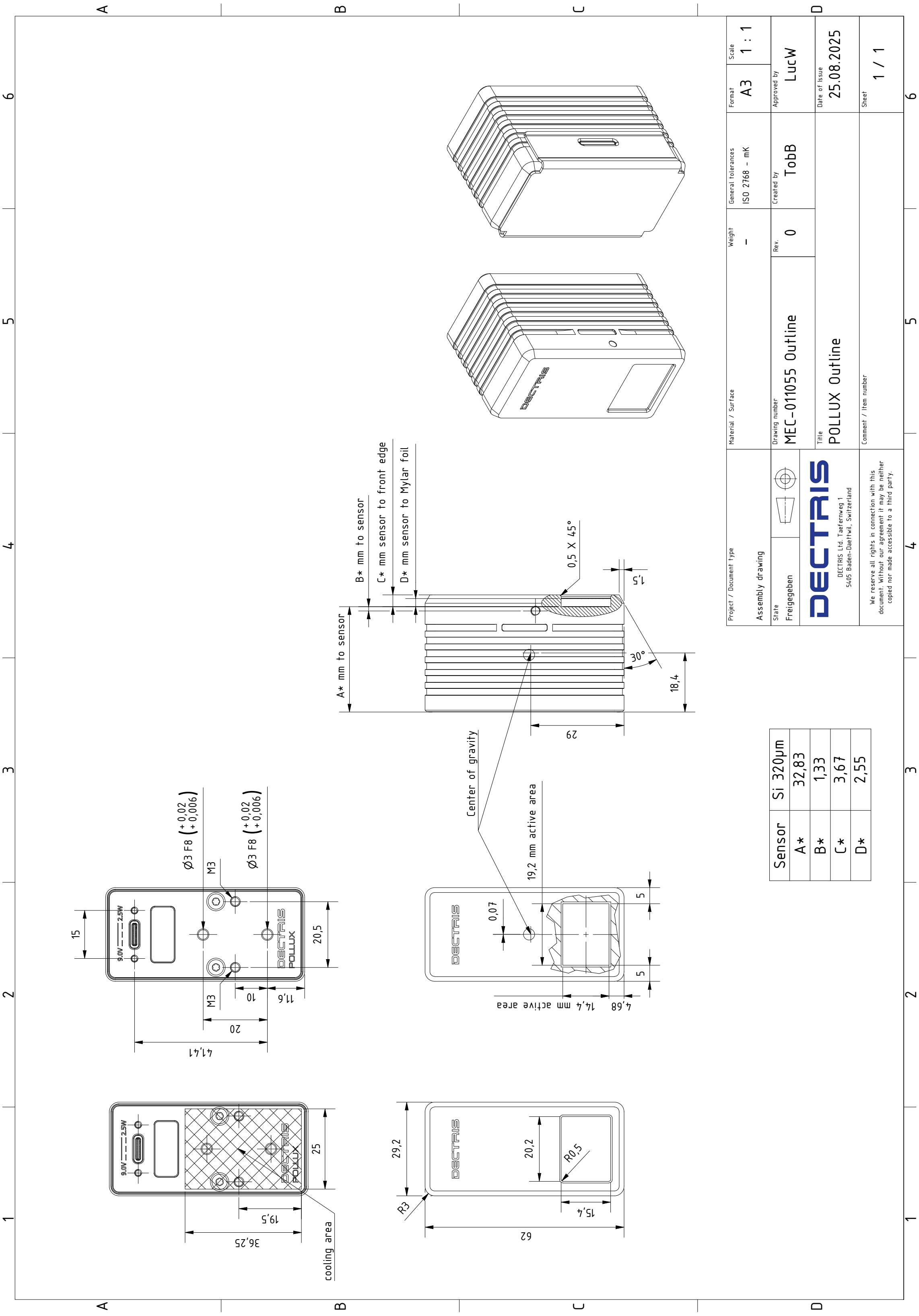
CAUTION
Please ensure the following points when storing the detector:

- Use a drying agent to make sure the humidity inside the storage / transport box does not exceed the specified humidity range.
- Make sure the temperature inside the storage / transport box does not exceed the specified range.

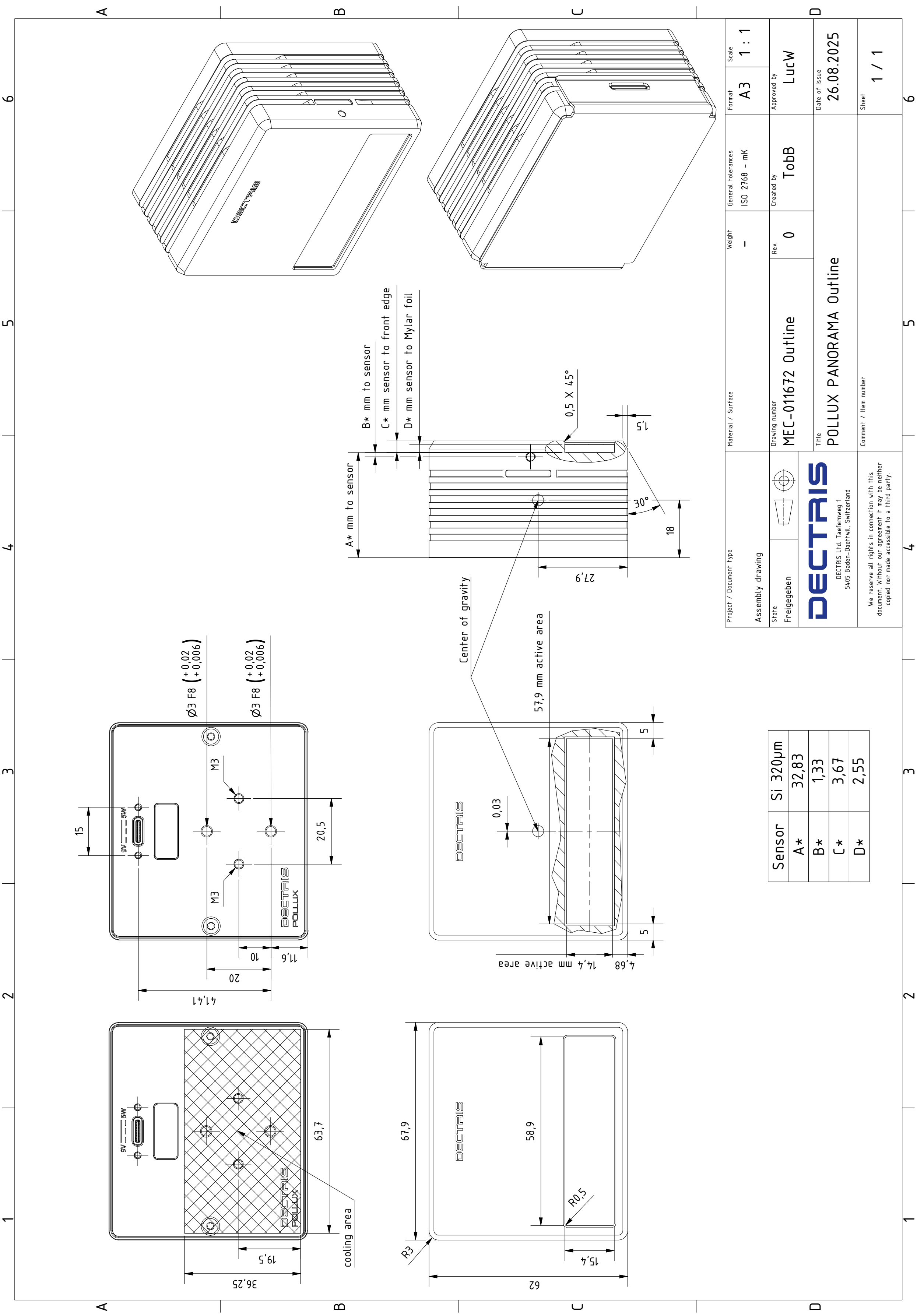
5. DETECTOR COMPONENTS AND CONNECTORS

5.1. Technical Drawings

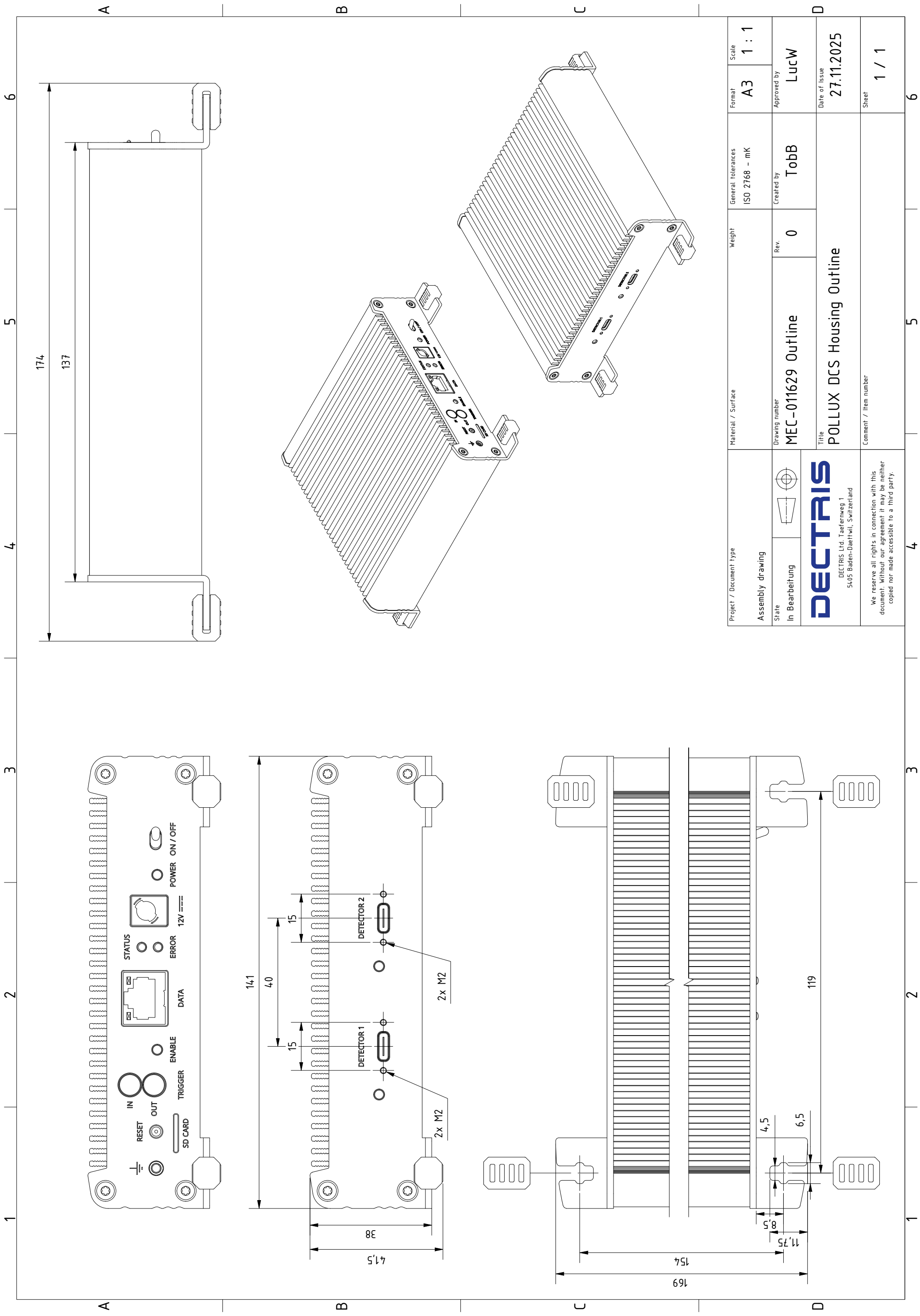
5.1.1. POLLUX Detector Outline



5.1.2. POLLUX Panorama Detector Outline

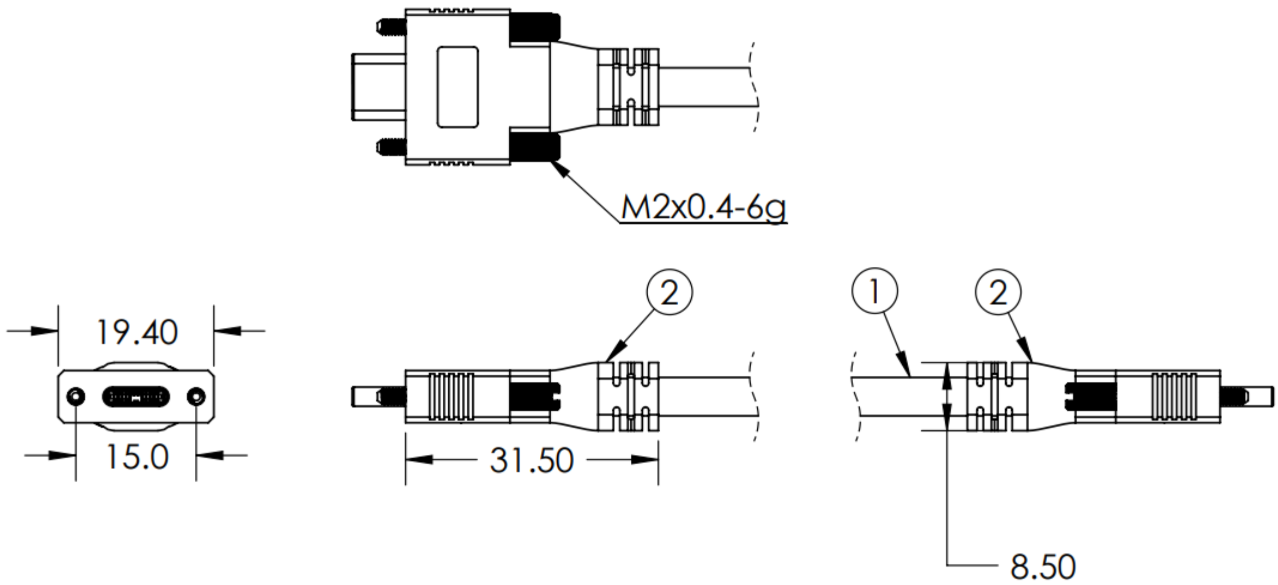


5.1.3. Detector Control System (DCS) Outline

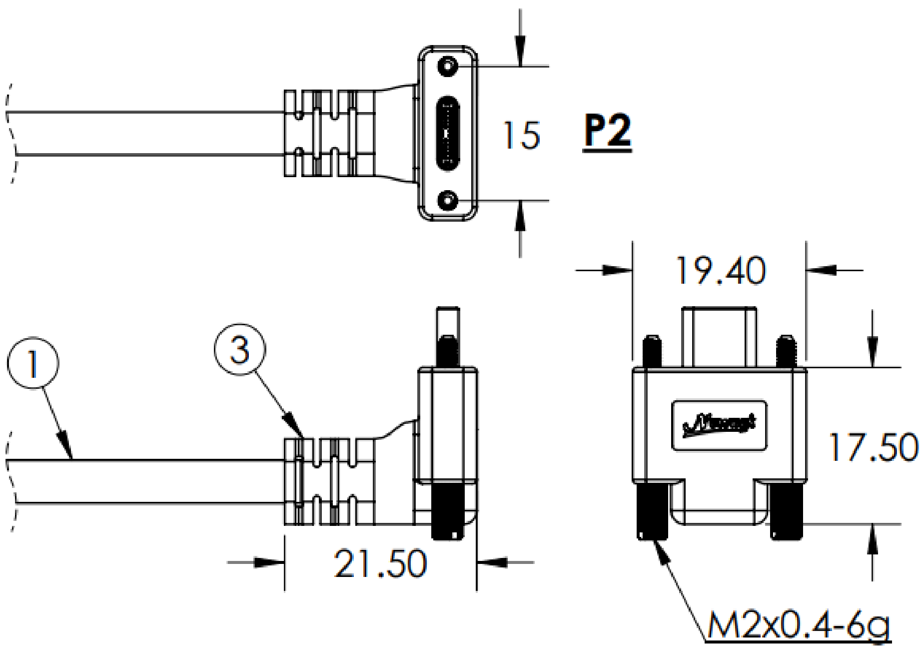


NOTICE
3D step files of the POLLUX detector are available on request. Please contact DECTRIS® technical support for more information.

5.1.4. USB-C Cable Connectors



Drawing of the straight USB-C connector. Units in mm. (1) Cable. Maximum length of 5m. (2) Strain relief.



Drawing of the angled USB-C connector. Units in mm. (1) Cable. Maximum length of 5m.. (3) Strain relief.

5.2. Front Side of the Detector

DANGER
Danger of electric shock. Do not touch the Mylar® foil. The sensors behind the Mylar® foil are operated at high voltages. Touching the Mylar® foil can cause an electrical shock.

WARNING
Do not touch the Mylar® foil to avoid damaging the sensors.

CAUTION
The cover does not protect the detector from X-rays and cannot be used as a beamstop.

The POLLUX detector head comes with a protective cover (0.4 mm, aluminum) for the front window, which should only be removed during operation. The sensors are behind a 12 μm thick Mylar® (PET) foil coated with aluminium to protect them from humidity, dust and from being touched.

To remove the protective cover, hold it from both sides and pull it away from the detector face. To mount it back on, align the protective cover with the front of the detector and push until it snaps into place. Ensure you do not touch the Mylar® window during this process. Avoid excessive force to prevent the cover from bending.

Figure 3. The POLLUX and POLLUX Panorama detector heads with the protective cover removed (front view)

5.3. Back Side of the Detector

Figure 4. The POLLUX Panorama detector (back view). The back of the detector head features the USB-C data port for the connection to the detector control system (DCS).

5.4. Detector Control System (DCS)

> **CAUTION**
> Do not access or modify the operating system of the detector control system (DCS).

The POLLUX detector control system (DCS) controls one or two POLLUX detector heads. The DCS has to be connected point-to-point to the detector head with a USB-C connector. The DCS can be integrated into the site network infrastructure using one of the interfaces described in Table 6. The DCS is optimized for performance and stability of operation. In order to achieve these goals we deliver the DCS with fixed firmware (BIOS etc.) and software (OS) version. The DCS must not be operated in an environment where unauthorized access is possible.

Using the web front-end, it is possible to restart the POLLUX control service, make an update, and shut down or reboot the detector control system. Any further control of the detector is carried out via the SIMPLON API (see separate documentation).

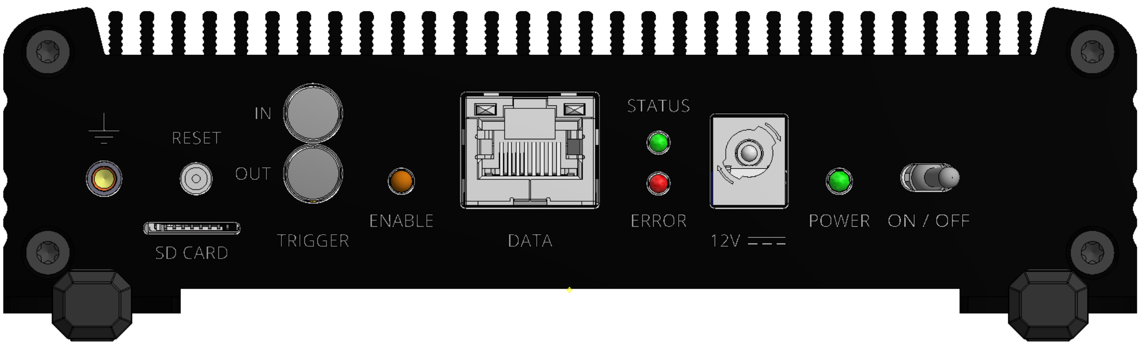


Figure 5. Front side of the detector control system.

The detector control system can be turned on and off by flipping the power switch. Turning off the detector control system will cut all power to the detector heads.

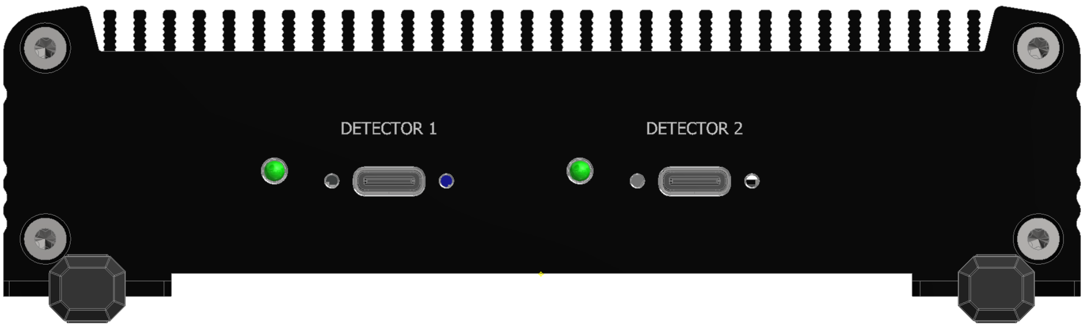


Figure 6. Back side of the detector control system.

5.4.1. Detector Control System Connectors

Table 6. Detector control system connectors

Connector	Description
+12V DC	DC power connector for the power supply. **CAUTION** • Only use the power supply unit provided by DECTRIS® to avoid damage to the detector. • When integrating the DCS into a custom setup, ensure that the power plug remains accessible to the operator at all times.
ETH	Interface Name: user1p1 (1 GBase-T) User configurable GbE Network Interface Default: DHCP with Link-Local fallback IP (`169.254.254.1`, see Section 6.2)
Trigger In	External trigger input Use a LEMO® Type 00 (NIM/CAMAC) cable.
Trigger Out	Enable out, high when counting is enabled. Use a LEMO® Type 00 (NIM/CAMAC) cable.
⏚	Functional earth connector **NOTICE** Although the detector may already be grounded via the mounting bolts, it must also be grounded through the functional ground connector at the back to establish a defined and reliable ground reference. Make sure not to create unintended ground loops.

Connector	Description
Detector 1 | 2	USB-C data port for the connection to the detector head. **CAUTION** Only use the USB-C data cables provided by DECTRIS® to avoid damaging the detector.

5.4.2. Status LEDs

Table 7. Meaning of the status LEDs on the detector control system

LED	Behavior	Description
Power	Static green	The DCS is powered.
	Off	No power to the DCS. Either the switch is off or the power is not connected.
	Blinking green	The internal fuse is triggering. Make sure the correct power supply is connected or contact DECTRIS® support.
Status	Off	The DCS has not finished booting.
	Static green	The DCS is ready to receive commands.
Error	Static red	The DCS is in an error state.
Enable	Orange	Acquisition ongoing.
Detector 1 | 2	Off	No connection detected.
	Blinking green (0.5 Hz)	Connection to detector head established, detector is still un-initialized.
	Static green	Detector is initialized and ready for use.
	Blinking red (4 Hz)	Error detected on detector head.
	Blinking orange (1 Hz)	Connection to detector head detected but communication not possible.
	Static red	Unknown device connected.

6. GETTING STARTED

6.1. Setting up the POLLUX detector system

CAUTION
Before operating the detector, make sure you have read the User Manual.

CAUTION
Operate only detector head(s) with the corresponding DCS delivered by DECTRIS® as a part of the detector system. The Factory Acceptance Test sheet delivered with the system contains the serial numbers of the detector heads and DCS forming the system.

WARNING
Only connect POLLUX detector heads to the detector control system (DCS) and never plug the POLLUX detector heads to any other USB device.

WARNING
Always ensure that the Detector Control System (DCS), detector head(s), and power supply have adequate ventilation.

1. Securely mount the detector heads to your instrument. The mounting points on the back of the head consist of 2 x ⌀ 3 mm holes for positioning accuracy and 2 x M3 threaded holes to secure the head.

   CAUTION

   Hot Surface

   The rear cooling surface becomes hot during operation. To prevent burns, prior to applying power, the detector head must be installed on a metal mount that provides adequate heat transfer and fully covers the designated cooling surface (refer to Section 5.1). Allow the device to cool for 5 minutes before dismounting.

2. Install the Detector Control System in a suitable location. The Detector Control System (DCS) and the power supply can be mounted in any orientation. The rubber pads on the DCS's feet can be removed, and the feet can be used for mounting the DCS.
3. Connect the DCS to ground potential, using the functional ground connector.
4. Connect the DCS to the power supply.
5. Connect the detector head to the DCS using the provided USB-C cable. Tighten the locking screws only finger-tight. Stop as soon as you feel resistance, and never exceed 0.1 Nm. A gap of approx. 1 mm between the plug shoulder and the detector head housing is normal. Do not force the screw further to close this gap, doing so will damage the connector. The plug on the DCS side will sit flush with the housing.
6. Connect the local network cable to the DCS.
7. Follow instructions in Section 6.2 to access the detector control system.

6.2. Connecting to the Detector Control System (DCS)

NOTICE
The DCS is shipped with **DHCP enabled** by default. In the event that a DHCP server is unavailable to assign an IP address, the system uses Zeroconf to automatically assign a link-local fallback IP (typically `169.254.254.1`).

A computer with a LAN connection is required to control the POLLUX detector system. After setting up the computer and connecting it to the DCS network port using an Ethernet cable, the computer will be able to communicate with the DCS. To access the web interface, use any internet browser and type the IP address into the address bar.

6.2.1. First connection to the Detector Control System

To connect to the DCS for the first time, several methods are available. In this section, we present two methods that allow for an easy start.

Hostname Resolution
If the DCS is connected to a network with DHCP, you can reach the device via its auto-generated hostname.

NOTICE
The DCS derives its unique hostname from the last 6 digits of its programmed MAC address. Example: a MAC address of `18:9B:A5:01:00:0B` translates to the hostname `pollux-01000B`.

The MAC address can be found on the underside of the DCS.

1. Ensure your PC is in the same subnet as the DCS.
2. You can connect to the system via a web browser using the local domain (e.g. `pollux-01000B.local` - see info box above).

Link-Local Fallback IP (Zeroconf)
If a DHCP server is unavailable, or if you are establishing a direct point-to-point connection between the host PC and the DCS, the system will use its fallback IP address.

NOTICE
If a DHCP server is unavailable, the DCS will attempt to assign itself the address `169.254.254.1`. The DCS will probe the local subnet and check for conflicts. If `169.254.254.1` is in use, it will increment the last digit until an available IP is found.

1. Make a point-to-point connection between the DCS and your PC.
2. Ensure the host PC's network interface is set to DHCP. Without a DHCP server, the PC will automatically assign itself an IP in the `169.254.0.0/16` range. Alternatively, assign a static IP in the `169.254.0.0/16` range to the host PC.
3. Once the physical link is detected and the link-local IP on the PC has been auto-assigned, navigate to `169.254.254.1` or use the hostname to interface with the DCS.

NOTICE
Note that even when a static IP is manually configured, the `169.254.254.1` fallback IP remains active. This allows maintenance access via a direct connection, even if the static IP configuration is unknown.

6.2.2. Network Settings Recommendations

For ongoing detector control, relying on standard DHCP or link-local addresses is not advised.

- It is recommended to assign a static IP address, create an IP reservation on your DHCP server, or add a DNS entry for the system. This ensures that the IP address does not change randomly during operation.
- You can change the IP settings to a static IP or re-enable DHCP by navigating to the Network tab in the DCS interface (see Section 7.2).

6.3. Vacuum Operation

POLLUX detectors can be provided for vacuum operation. The typical reachable vacuum is 10^{-3} mbar. The outgassing rate and content are not specified. The vacuum compatibility option guarantees that the detector will operate in the mentioned vacuum range.

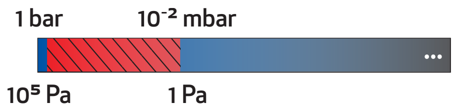
To verify that your detector is vacuum compatible, check that the vacuum compatibility option is included in the order confirmation received from DECTRIS®.

> **WARNING**
> A vacuum compatible POLLUX detector requires external cooling for in-vacuum operation. Heat transfer via the bottom of the detector is imperative for stable operation. Refer to Table 2 for the detector head power ratings. See Section 5.1.1 for cooling area dimensions (bottom area of the detector head).

6.3.1. Ambient Conditions for In-Vacuum Use

For in-vacuum operation of the detector, the following conditions must be fulfilled:

Table 8. In-Vacuum Operating Conditions

In-Vacuum Condition	Definition
Pressure during operation	Atmospheric pressure or less than 0.01 mbar 
Detector head temperature during operation	40°C to 60°C
Chamber temperature during ”bake-out” (detector unpowered)	max. +60°C **WARNING** Detector must be unpowered during bakeout.

6.3.2. Operation Procedure in Vacuum

Make sure the vacuum conditions in Table 8 are met and follow the procedures below for venting and pumping down the vacuum chamber.

WARNING
Always ensure the detector has NO POWER and is WARMED UP to room temperature before pumping down or venting. Otherwise, it could be damaged through electrical discharge or condensation.

Pumping Down the Vacuum Chamber

- Mount the detector head properly inside the vacuum chamber.
- Connect the detector USB-C cable inside and outside the vacuum chamber.
- Connect the USB-C cable to the detector control system (DCS) outside the vacuum chamber.
- Make sure the connection to the detector head is working by initializing the detector.
- Make sure the detector head is POWERED OFF by turning off the detector control system (DCS) and disconnecting it from the power supply.
- Close the vacuum chamber and start to pump down.
- After the pressure inside the vacuum chamber is sufficiently low to meet the operating conditions, connect the detector control system to the power supply and power up the detector system.

Venting the Vacuum Chamber

- Turn OFF the power to the detector by disconnecting the power supply to the detector control system (DCS).
- Make sure the detector head is at room temperature or higher.
- Vent the chamber.

6.4. Cleaning and Maintenance

CAUTION
The Mylar® foil must not be touched or cleaned. If it is damaged, please contact DECTRIS® technical support.

The detector housing can be cleaned with a soft tissue.

The POLLUX detector does not require any maintenance.

7. WEB INTERFACE

7.1. Overview

The POLLUX web interface (Figure 7) provides simple access to basic functions and settings of the detector system for installation, debugging, and system updates. For productive operation of the detector, please refer to the SIMPLON API documentation.

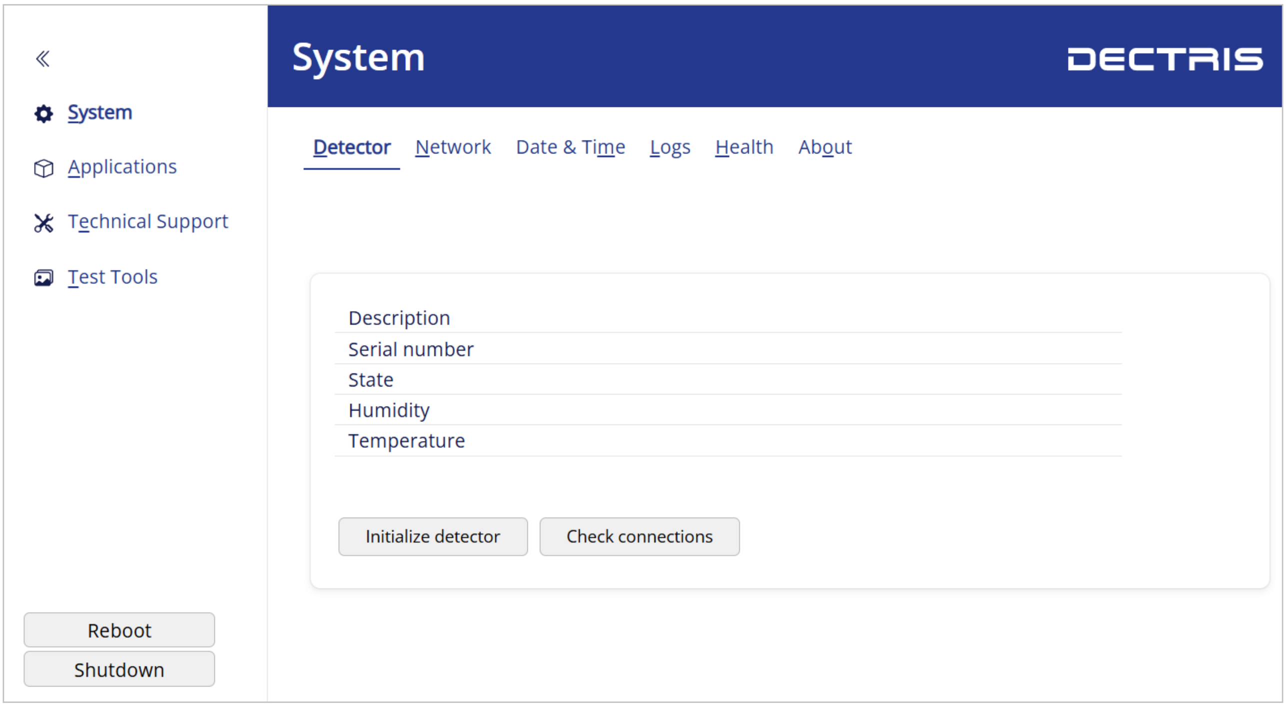


Figure 7. POLLUX home page.

Table 9 summarizes the functions available through the left panel of the web system interface.

Table 9. Menu items of the POLLUX web interface.

Menu Item	Content
System	View the system information and access the Detector Control System settings (see Section 7.2)
Applications	Manage the detector and calibration applications. This tab provides functions to start and stop the detector software, change software versions and upgrade the detector software to a new version.
Technical Support	Create a bug report that can be downloaded and sent to support@dectris.com. Note: The bug report is not sent automatically.

Menu Item	Content
Test Tools	Acquire images with these tools for **basic** detector configuration, data acquisition and image previewing via the web interface. Available from software version 2026.1. See Section 8.2.1 for more details. **NOTICE** These are out-of-the-box tools to help acquire your first images, with a limited scope. They are not a full performance solution for data acquisition, image processing and analysis. See Section 8.3 for more in-depth functionality. **CAUTION** Actions on this interface may affect active acquisitions or impact other users concurrently operating the detector.

7.2. System Settings and Administration

To access the POLLUX system settings, click on the corresponding menu item on the homepage. The System tab shows the detector information and allows to configure the network settings of the Detector Control System.

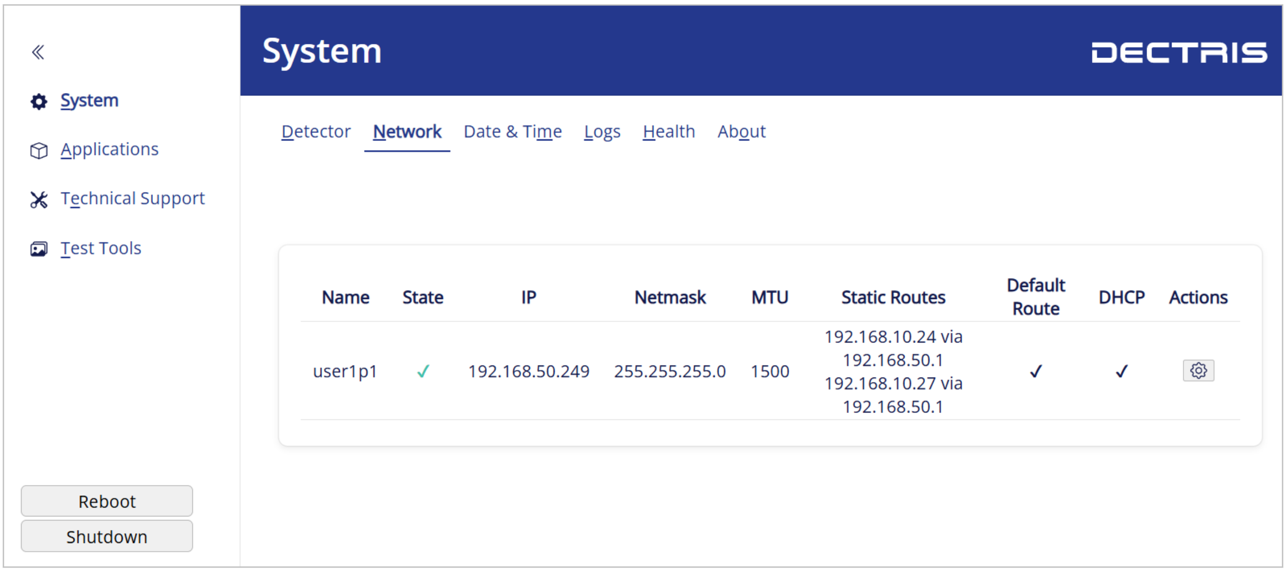


Figure 8. Screenshot of the System settings showing the network configuration page.

Table 10. Description of the System menu Items

Tab	Description
Detector	See the detector status and initialize the detector. The detector status can only be read out when the detector is initialized and the detector application is running.
Network	Configure the user accessible network interfaces.

Tab	Description
Date & Time	Configure date and time on the Detector Control System. Provides the possibility to add an NTP server.
Logs	Select and view detector logs, including error, warning and debugging information.
Health	View system health information, such as temperature and humidity history, as well as board errors.
About	Display system details.

8. GENERAL USAGE OF THE DETECTOR SYSTEM

8.1. Detector Control and Output

The POLLUX detector system is controlled through the SIMPLON API, an interface to the detector that is based on the http protocol. The API documentation supplied with the system describes this interface in detail and allows for easy integration of the detector control into the user instrumentation independent of the operating system or programming language being used. Please refer to the SIMPLON API Reference document for details.

The data recorded by the detector can be accessed in different ways. Images can be stored by the FileWriter on the Detector Control System as HDF5 files, which include metadata in a NeXus-compatible format. Buffered files have to be regularly fetched and subsequently deleted on the Detector Control System as the buffer space is limited[1]. Support for reading out multiple thresholds has been introduced in FileWriter2. See the SIMPLON API Reference for further details.

Data can also be fetched through the Stream interface, which has a low latency and offers utmost flexibility. The metadata is transferred as part of the header. Streamed data is not buffered and will be lost if not fetched or incompletely fetched. With the introduction of Stream2 it is possible to retrieve the images from multiple thresholds simultaneously at full frame rates.

Lastly, the Monitor interface provides images as raw tiff files with minimal metadata. This is a low performance and low bandwidth interface, which allows to retrieve the latest recorded image for monitoring purposes during acquisitions. The Monitor offers a relatively small buffer. In case of buffer overflow, the oldest images will be overwritten. The Monitor also allows to retrieve the images from multiple thresholds.

8.2. Recording an Image or an Image Series

CAUTION
Data might have to be fetched concurrently to a running image series. The lifespan of the data on the Detector Control System is dependent on the configuration of your system as well as the interface used for collecting data. Data not fetched within this lifespan is permanently lost.

To record images or image series, the following steps need to be performed through the SIMPLON API (see the API Reference for details).

1. Make sure the detector has been set up according to the steps described in Section 6.
2. Initialize the detector if this has not yet been done. The detector does not need to be reinitialized unless the detector entered an error status.
3. Set the detector parameters for data acquisition and specify and configure the desired output interface (FileWriter and/or Stream interface). A list of essential configuration parameters can be found in Section 8.3.1.
4. Arm the detector.
5. Record the image or image series.
   • Send trigger(s) to record the image or image series as previously configured.
   • Fetch data through the data interface(s).

[1] Buffer space varies dependent on the configuration of your system, buffer overflow will cause loss of data. See API Reference for further details.

6. Disarm the detector (to ensure files are finalized and closed).
7. Repeat from step 3 for further data acquisition with different settings, or from step 4 for identical settings.

8.2.1. Basic Image Acquisition via the Web Interface

The POLLUX web interface has built-in tools to acquire and display your first images in a simple way (available from software version 2026.1).

After accessing the web interface (Section 7.1), click the **Test Tools** on the left panel.

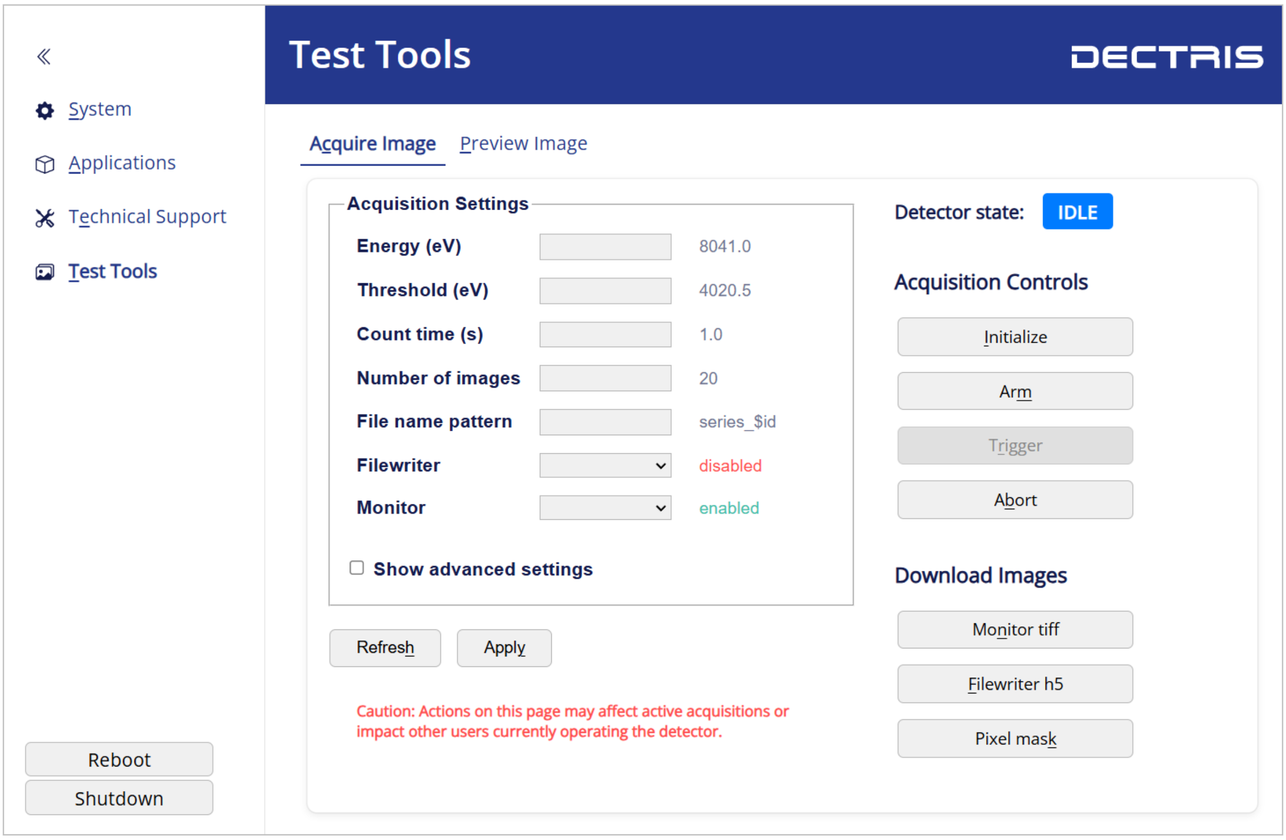


Figure 9. Basic detector configuration via the Web Interface

The **Acquire Image** tab allows you to set the most essential configuration parameters for basic image acquisition (see Section 8.3.1 for details). Make sure to enable the FileWriter and/or the Monitor interfaces to record and download the images.

CAUTION
The *Advanced Settings* menu should only be used in very specific cases and applications. Changing the advanced parameters will impact the data quality and should only be done with full understanding of the outcome. Make sure to read the SIMPLON API Reference manual for details.

When changing parameters, first click the Apply button and then Arm to save the settings on the detector.

The Trigger button will start the image series via the internal software trigger (see INTS mode in Section 9.2). The detector will disarm automatically once the series is finished.

The Abort button can be used to immediately stop an ongoing acquisition, but note that the last running image will be lost when aborted.

The images can be downloaded in TIFF format via the Monitor interface, or HDF5 through the FileWriter.

The Pixel Mask can also be downloaded as a TIFF file. See Section 10 for more details.

NOTICE
These are out-of-the-box tools to help acquire your first images, with a limited scope. They are not a full performance solution for data acquisition, image processing and analysis. See Section 8.3 for more in-depth functionality.

CAUTION
Actions on this interface may affect active acquisitions or impact other users concurrently operating the detector.

8.2.2. Displaying an Image via the Web Interface

TIFF images can be viewed via the Web Interface under the **Preview Image** tab of the Test Tools (available from software version 2026.1).

This feature allows to load images from the Monitor Interface or upload TIFF images from your local storage. Monitor images can also be played automatically.

The tool provides basic display settings and allows to save the current image as PNG or TIFF.

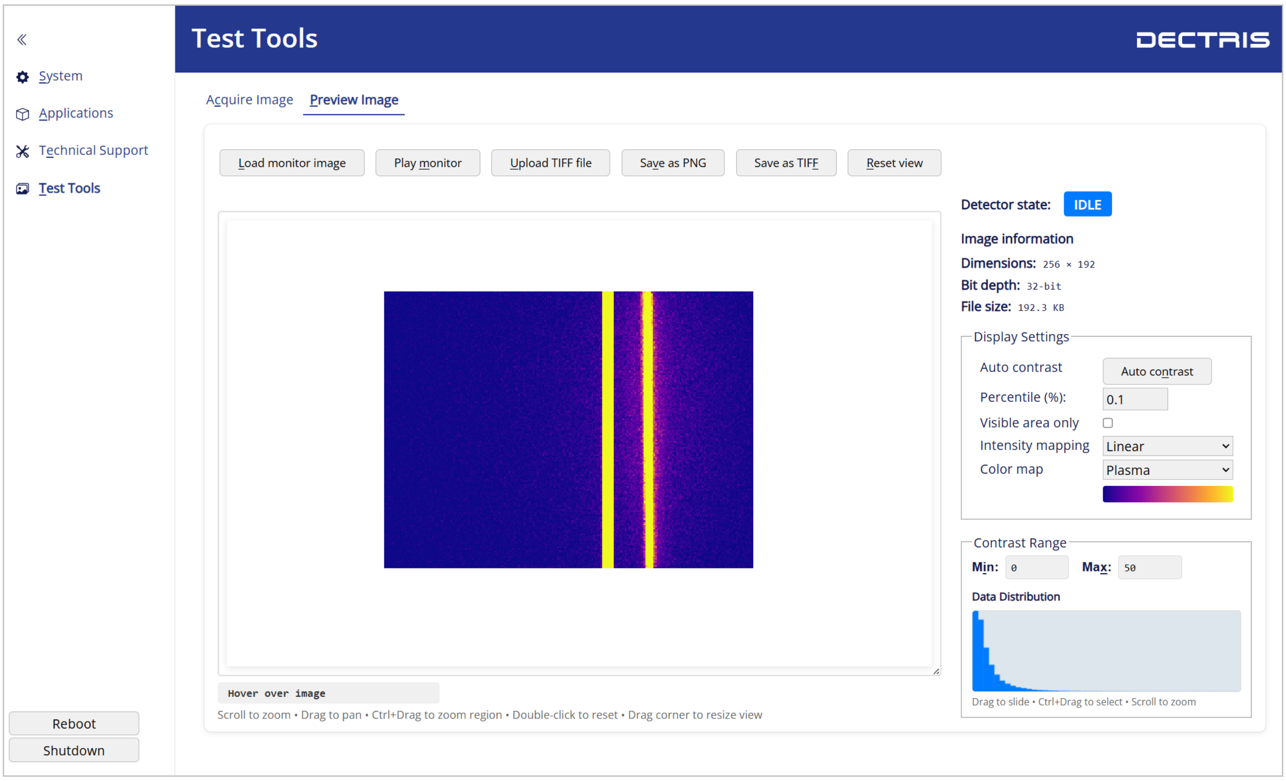


Figure 10. Viewing an example TIFF image via the Web Interface

8.3. Control of the Detector from a Specific Environment

Integrating the detector into a specific environment requires understanding of the necessary detector functions. The API reference will list all possible commands and features, but it does not give an explanation of the required functionality. Section 8.3.1 and Section 8.3.2 cover a selection of essential and situational parameters respectively.

8.3.1. Main Configuration Parameters

The parameters described in this section allow control of the detector and data acquisition. Data will be acquired, however further configuration of the interface for data retrieval might be necessary depending on your set up. For starting a data acquisition, only the following parameters need to be adjusted:

- detector | config | count_time
- detector | config | frame_time
- detector | config | photon_energy

The parameter count_time refers to the actual time the detector counts photons, while frame_time is the interval between acquisitions of subsequent frames (i.e. period) and defines the frame rate.

The detector configuration parameter photon_energy has to be set to the X-ray energy used for the experiment. There is a convenience function for setting the photon_energy, called element.

- detector | config | element

The parameter *element* accepts the chemical symbols for elements, e.g., ”Cu”, ”Mo”, as argument and sets photon_energy to the $K_{\alpha 1}$ emission line for that element.

- detector | config | ntrigger
- detector | config | nimages

The ntrigger parameter configures the number of expected triggers for one acquisition series. Setting values greater than 1 for ntrigger allows several trigger commands or external trigger pulses per arm/disarm sequence. This mode allows recording several series of nimages with the same parameters.

The parameter nimages configures the number of images that will be acquired for one trigger. In internally and externally triggered series (”ints” and ”exts”, respectively), the detector considers a trigger as the start of a series of n images. For example a single image is considered as a series of images containing 1 image. Once the detector has been armed, a series can be started by issuing a trigger command or triggering the detector using an electronic pulse on the external trigger input. In ”inte” and ”exte” trigger modes the parameter nimages should be set to 1, as every trigger will always trigger a single image. To switch between the trigger modes (see Section 9) one can use the configuration parameter trigger_mode.

- detector | config | trigger_mode

NOTICE
Please note that the acquired data can be retrieved via different interfaces. For details, please see the API Reference.

With the filewriter mode set to ”enabled”, the acquired data is written to HDF5 files. The FileWriter has the following important configuration parameters:

- filewriter | config | mode
- filewriter | config | name_pattern
- filewriter | config | nimages_per_file

- filewriter | config | compression_enabled

The filewriter name_pattern sets the name for the HDF5 files. When using the pattern ”$id”, it will be replaced with a sequence identification number and can be used to differentiate between subsequent series. The sequence identification number is reset after initializing the detector. The parameter nimages_per_file sets the number of images stored per data file. A value of 1000 (default) means that for every 1000th image, a data file is created. If for example, 1800 images are expected to be recorded, the arm, trigger, disarm sequence means that a master file is created in the data directory after arming the detector. The trigger starts the image series and after 1000 recorded images one data container is made available on the buffer of the Detector Control System. No further files will be made available until the series is finished either by completing the nth image (nimages) of the nth trigger (ntrigger) or by ending the series using the detector command disarm. As soon as either criteria is met, the second data container, with 800 images in this example, is closed and made available for fetching.

8.3.2. Additional Configuration Parameters

> **NOTICE**
> The following parameters are for special conditions and should be set with care and with understanding of the consequences. Changing these parameters to non-default values can have a substantial negative impact on data quality!

- detector | config | threshold/n/energy
- detector | config | threshold/n/mode
- detector | config | threshold/difference/mode

The POLLUX detectors provide two independent thresholds and it is possible to read out either threshold 1 or the difference between thresholds 1 and 2. The thresholds can be enabled or disabled using the threshold/n/mode configuration parameter, where *n* is the threshold number. The difference mode is enabled with the threshold/difference/mode parameter. This requires thresholds 1 and 2 to be disabled beforehand.

When setting the photon_energy, the lower threshold (n=1) is set automatically to a default value of 50 % of the photon_energy. The value of threshold/1/energy should only be changed in cases where the suppression of fluorescence is a necessity in the experiment. The value of threshold/1/energy should be kept within 50% to 95% of the incoming photon_energy. The API incorporates no sanity check on the threshold/n/energy.

Corrections are enabled by default, but can be turned off using the following detector configuration parameters. In the vast majority of experiments data quality benefits from the data corrections. Therefore, disabling either correction will likely result in inferior data quality.

- detector | config | countrate_correction_applied
- detector | config | flatfield_correction_applied
- detector | config | pixel_mask_applied

Further parameters, represented by the following selection, allow to enrich the meta-data of the image (series) with experimental data.

- beam_center_x
- beam_center_y
- detector_distance
- detector_orientation
- detector_translation
- wavelength (see Section 8.3.3 for dependency with photon_energy)

Further parameters and their function are described in the API Reference document.

8.3.3. Interdependency of Configuration Parameters

Interdependency of Calibration Parameters
The following calibration parameters have an implied or direct dependency. Changing either of the parameters might influence other parameters in the list.

- detector | config | photon_energy

Changing photon_energy sets element to an empty string and sets wavelength to its corresponding value. The threshold_energy is set to half of photon_energy, which is the optimal threshold energy in most cases. The threshold/2/energy is set to 1.15 x photon_energy, which can be used to suppress higher energy photons and cosmic radiation. In order to use values different from the defaults, the threshold energies should be explicitly set after setting photon_energy. The flatfield is recalculated whenever a calibration relevant parameter is changed.

- detector | config | element

Setting the element is equivalent to setting photon_energy to the energy of the $K_{\alpha 1}$ emission line of the element. Hence, photon_energy, wavelength and all parameters that depend on photon_energy are changed accordingly.

- detector | config | wavelength

Setting the wavelength is equivalent to setting photon_energy to the equivalent energy of the wavelength. Hence, photon_energy, element and all parameters that depend on photon_energy are changed accordingly.

- detector | config | threshold/n/energy

Changing the threshold energy causes the flatfield for this threshold to be recalculated.

- detector | config | flatfield

The flatfield applied for a given photon_energy and threshold is a result of the detector calibration. During the factory calibration a multitude of flatfields at different settings have been recorded to ensure optimal data quality of the flatfield for all common settings.

Interdependency of Timing Parameters
The following parameters are essential for exposure timing. Changing these values might influence other values, as described below.

- detector | config | frame_time

If frame_time conflicts with the current count_time, then count_time is set to the difference of frame_time and detector_readout_time.

- detector | config | count_time

If *count_time* conflicts with frame_time, then frame_time is set to the sum of count_time and detector_readout_time. To acquire images with a certain frame rate and best possible duty cycle, a simple procedure is to first set count_time to the inverse of the frame rate and subsequently frame_time to the inverse of the frame rate.

8.3.4. Examples

cURL
cURL, an abbreviation for Client for URLs, is a robust command-line tool designed for making HTTP requests. It is compatible with most operating systems and serves as an effective means of interacting with the SIM-

PLON API. Below, several example cURL prompts are provided to illustrate how to read and write detector settings, as well as interface files stored on the system. The examples assume the IP address of the system to be 169.254.254.1.

Commands can be transmitted using a PUT request.

```
# Reboot the DCU
curl -X PUT http://169.254.254.1/system/api/1.8.0/command/reboot

# Initialize the detector
curl -X PUT http://169.254.254.1/detector/api/1.8.0/command/initialize

# Arm the detector for acquisition
curl -X PUT http://169.254.254.1/detector/api/1.8.0/command/arm

# Send a software trigger signal
curl -X PUT http://169.254.254.1/detector/api/1.8.0/command/trigger
```

Configuration parameters are modifiable through PUT requests with an additional payload and retrievable via GET requests in the SIMPLON API. As the API primarily uses JSON as the default encoding, it is essential that the payload of a PUT request be in the form of a JSON-encoded string.

```
# Set parameter of configuration key e.g. setting the count_time to 10 seconds
curl -X PUT -d "{\"value\":10}" -H "Content-Type: application/json"
http://169.254.254.1/detector/api/1.8.0/config/count_time

# Read configuration key of e.g. count_time
curl -X GET http://169.254.254.1/detector/api/1.8.0/config/count_time
```

The FileWriter interface creates HDF5 files compatible with applications like ALBULA. It must be enabled explicitly and a custom filename can be specified. Once an acquisition is completed with the FileWriter interface enabled, the buffered HDF5 files become accessible on the system. Users can then conveniently download or delete these files as needed.

```
# Enable the filewriter interface
curl -X PUT -d "{\"value\":\"enabled\"}" -H "Content-Type: application/json"
http://169.254.254.1/filewriter/api/1.8.0/config/mode

# Change the filename to 'example_file'
curl -X PUT -d "{\"value\":\"example_file\"}" -H "Content-Type: application/json"
http://169.254.254.1/filewriter/api/1.8.0/config/name_pattern

# Request a list of all filewriter files on the system
curl -X GET http://169.254.254.1/filewriter/api/1.8.0/status/files

# Download file 'example_file_master.h5'
curl -X GET http://169.254.254.1/data/example_file_master.h5
--output example_file_master.h5

# Delete filewriter file 'example_file_master.h5'
curl -X DELETE http://169.254.254.1/data/example_file_master.h5
```

Python3

Python offers a convenient and versatile approach for operating the detector through the SIMPLON API. Below is an example Python script that uses the DEigerClient[2] class to operate the detector through the SIMPLON API. The example assumes the IP address of the system to be 169.254.254.1.

```python
from DEigerClient import DEigerClient

# Create an instance of the client
client = DEigerClient('169.254.254.1')

# Initialize the detector (only needed once)
client.sendDetectorCommand('initialize')

# Enables the filewriter interface and change the name_pattern to 'example_file_$id'
client.setFileWriterConfig('mode', 'enabled')
client.setFileWriterConfig('name_pattern', 'example_file_$id')

# Set detector configuration parameters (10 images with 0.1 s exposure time each)
client.setDetectorConfig('nimages', 10)
client.setDetectorConfig('count_time', 0.1)
client.setDetectorConfig('frame_time', 0.1)

# Arm the detector, trigger the acquisition and disarm the detector
client.sendDetectorCommand('arm')
client.sendDetectorCommand('trigger')
client.sendDetectorCommand('disarm')

# Download the files from the system to the current folder
client.fileWriterSave('example_file_master.h5', '.')
client.fileWriterSave('example_file_data_000001.h5', '.')

# Delete filewriter file 'example_file_master.h5'
client.fileWriterFiles('example_file_master.h5', 'DELETE')
client.fileWriterFiles('example_file_data_000001.h5', 'DELETE')

# Delete all filewriter files
client.sendFileWriterCommand('clear')
```

[2] You can request the DEigerClient python script from DECTRIS Support (support@dectris.com).

9. TRIGGER MODES

9.1. Introduction

NOTICE
The examples in this section should work on every POLLUX system. If the settings or the external trigger pulses applied are out of specification, acquisitions will not be performed and the measurement obtained with the detector might be incomplete. Make sure to consult the Ratings subsection in the Technical Specifications for the correct trigger input range. All values used in the examples are for demonstration purposes only and should be adapted to meet the requirements of your application.

In order to record an image or a series of images, the POLLUX detector has to be initialized, configured, armed, and triggered. The steps necessary to record an image series are comprehensively described in Section 8.2. The detector can be triggered through software (internal trigger) or by an externally applied trigger signal (external trigger). The different trigger modes available are described in the following sections.

9.2. INTS - Internal (Software) Triggering

An exposure (series) can be triggered by using a software trigger. This is the default mode of operation.

Example detector configuration for internally triggered exposure series:

detector | config | trigger_mode	{"value": "ints"}
detector | config | nimages	{"value": 10}
detector | config | ntrigger	{"value": 3}
detector | config | frame_time	{"value": 1}
detector | config | count_time	{"value": 0.7}

The detector starts the first exposure after the trigger command has been received and processed[3]. All subsequent frames are triggered according to the configuration of the frame_time and count_time parameters. The detector records nimages frames per trigger and stays armed until ntrigger are received. Figure 11 depicts an internally triggered series defined by frame_time, count_time and nimages.

[3]As the trigger command is sent over a TCP/IP connection the exact latency of the start of the exposure is hard to predict.

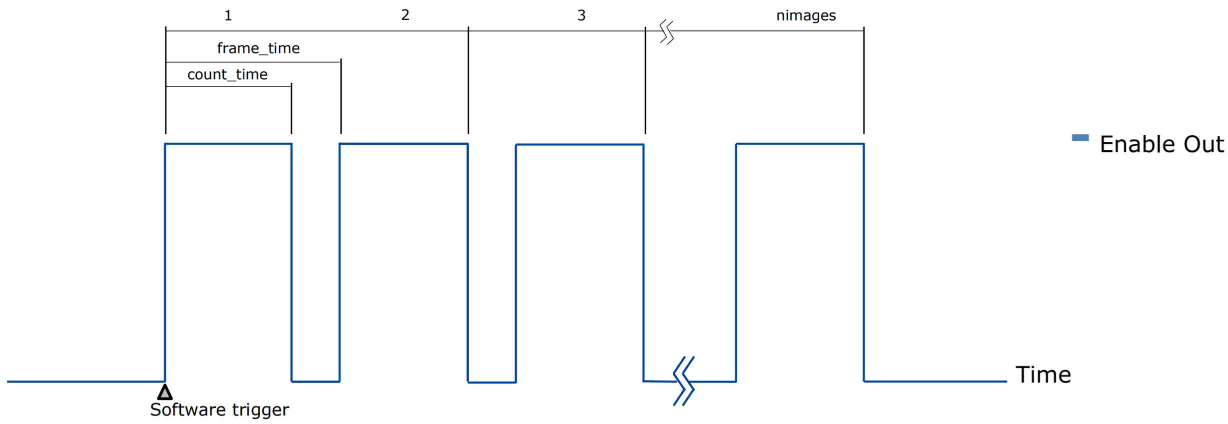


Figure 11. INTS - Series of exposures, defined by frame_time, count_time and nimages, triggered by a software trigger.

9.3. INTE – Internal (Software) Enable

In the trigger_mode 'inte' a single image or series of images with varying exposure times can be started by issuing a number ntrigger of trigger commands. Unlike the trigger_mode 'ints', the 'inte' trigger commands take an additional argument containing the count_time for the subsequent frame. In all enable modes the detector configuration parameter nimages is implied to be 1. The number of frames in a series therefore is solely based on the value of the parameter ntrigger.

NOTICE
The configured count_time and frame_time should be close to the count time and frame time of the shortest expected exposure in the configured series. The set count_time will be used to calculate internal auto-summation configuration values (see Section 3.4). In most situations a reasonable estimate of these values is sufficient.

Example detector configuration for an internally enabled exposure series:

detector | config | trigger_mode	{"value": "inte"}
detector | config | nimages	{"value": 1}
detector | config | ntrigger	{"value": 3}
detector | config | frame_time	{"value": 1.0}
detector | config | count_time	{"value": 0.7}

The detector starts the first exposure after a trigger command has been received and processed[4]. All subsequent frames are triggered by individual trigger commands with an additional argument containing the count_time of the triggered frame. The detector stays armed until ntrigger are issued or the detector is disarmed. Figure 12 depicts an internally enabled exposure series defined by count_time (payload of the trigger command) and ntrigger. Table 11 summarizes the commands issued to record the same series.

[4]As the trigger command is sent over a TCP/IP connection the exact latency of the start of the exposure is hard to predict.

Table 11. Command sequence for an internally enabled (inte) series.

Method	Parameter	Payload
PUT	detector | command | arm	
PUT	detector | command | trigger	{"value": 0.7}
PUT	detector | command | trigger	{"value": 2.1}
...		
PUT	detector | command | trigger	{"value": 0.7}

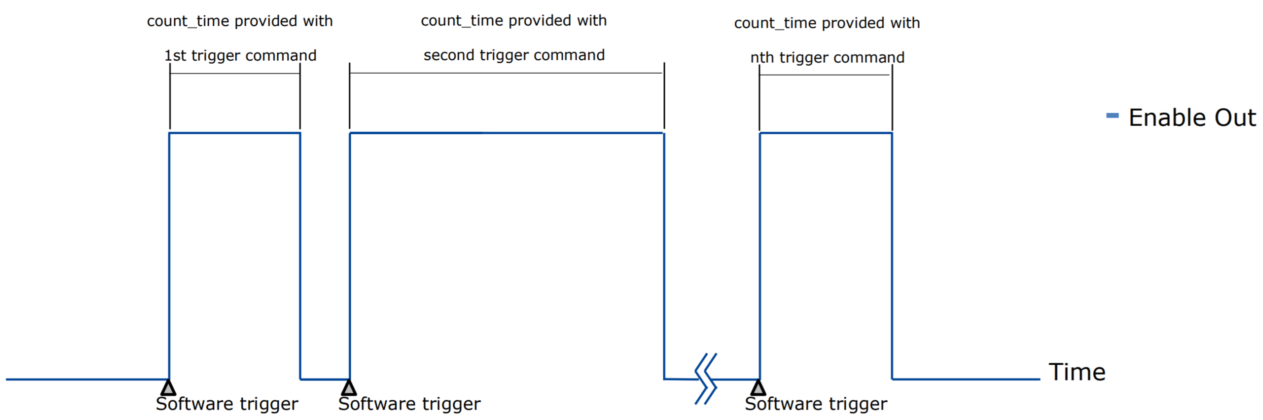


Figure 12. INTE - Series of exposures, defined by count_time (payload of the trigger command) and ntrigger, triggered by a software trigger.

9.4. EXTS - Externally Triggered Exposure Series

CAUTION
Consult the Technical Specifications for details about the required electrical characteristics of the trigger signal.

The POLLUX detector systems also support external triggering. In the trigger_mode 'exts', nimages are recorded per trigger until ntrigger are received. Both count_time as well as frame_time are defined by the configuration. Example settings for an externally triggered exposure series are shown here:

detector | config | trigger_mode	{"value": "exts"}
detector | config | nimages	{"value": 10}
detector | config | ntrigger	{"value": 1}
detector | config | frame_time	{"value": 1.0}
detector | config | count_time	{"value": 0.7}

After the detector has been initialized, configured, and armed the acquisition can be triggered by a single external trigger pulse. The detector starts exposing after the (electrical) trigger signal has been issued.

All subsequent frames are internally triggered according to the information previously configured by the frame_time and count_time parameters. The detector records nimages frames and stays armed until ntrigger are received. Figure 13 depicts an externally triggered series defined by frame_time, count_time and nimages.

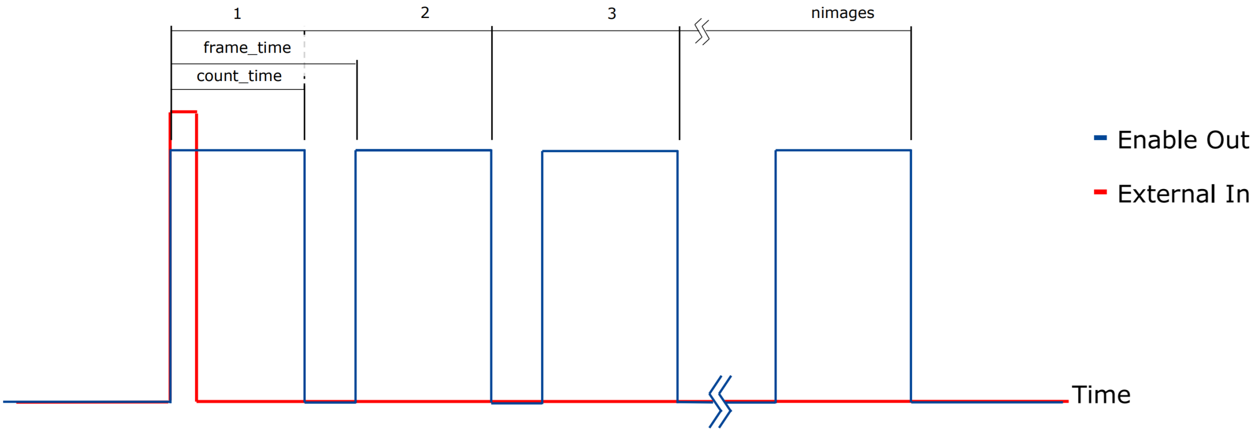


Figure 13. EXTS - Exposure series defined by frame_time, count_time and nimages, triggered by a single external trigger pulse. Note that the periods are not drawn true to scale.

9.5. EXTE - Externally Enabled Exposure Series

CAUTION
Consult the Technical Specifications for details about the required electrical characteristics of the trigger signal.

The POLLUX detector systems also support external enabling. In the external enable mode 'exte' a series of ntrigger frames can be recorded. The count time as well as the period of individual frames of a series are defined by the duration of the high state of the external enable signal. In all enable modes the detector configuration parameter nimages is implied to be 1. The number of frames in a series therefore is solely based on the value of the parameter ntrigger.

NOTICE
The configured count_time and frame_time should be close to the count time and frame time of the shortest expected exposure in the configured series. The set count_time will be used to calculate internal auto-summation configuration values (see Section 3.4). In most situations, a reasonable estimate of these values is sufficient.

Example detector configuration for externally enabled exposure series:

detector | config | trigger_mode	{"value": "exte"}
detector | config | nimages	{"value": 1}
detector | config | ntrigger	{"value": 10}
detector | config | frame_time	{"value": 1.0}
detector | config | count_time	{"value": 0.7}

After arming the detector, the acquisition can be enabled by an external signal. The value ntrigger defines how often this can be repeated. The detector starts exposing the first image after the rising edge and stops after the falling edge of the external trigger signal. In the same manner, all subsequent frames are externally enabled. The count time and period are therefore solely determined by the external enable signal and the limitations of your detector system. The detector records as many frames as valid (according to the specifications) enable pulses are received until the value set for ntrigger is reached. Figure 14 illustrates a externally enabled series.

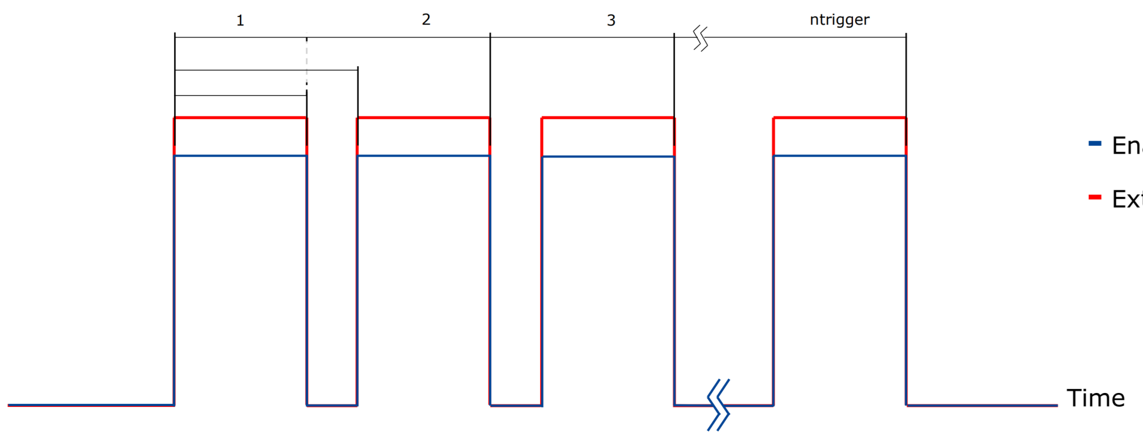


Figure 14. EXTE - Exposures defined by external enable

9.6. EIES - Externally Interrupted Exposure Series

In the externally interrupted exposure series ("eies" mode), frames will be started and stopped via the external trigger. The first trigger will start the first frame. Subsequent triggers will stop the running exposure and immediately start the next exposure. The length of the trigger pulse itself is inconsequential. The total number of frames to be acquired is defined by ntrigger. However, in this mode, it is necessary to send an additional trigger (ntrigger + 1) to stop the last frame and end the series. Setting the count_time and frame_time as close to the expected real value is recommended to ensure the detector count rate correction works optimally. This mode is useful when the exact length of an exposure is unknown and no gating signal is available.

It is also possible to skip triggers if not all triggers should start a new frame. This can be done by configuring the parameter ntriggers_skipped. If set to 0, every trigger starts a new frame. If set to 1, every second trigger starts a new frame (skipping 1). If set to 2, every third trigger starts a new frame, and so on. The final number of images acquired will still be equal to the value of ntrigger.

NOTICE
To use this mode, the parameter nimages has to be set to 1. The number of frames to be acquired will be defined by ntrigger. This configuration must be set before enabling the "eies" mode, otherwise an error will be returned and the mode cannot be enabled.

Example configuration for an externally interrupted exposure series:

detector | config | nimages	{"value": 1}
detector | config | ntrigger	{"value": 4}
detector | config | trigger_mode	{"value": "eies"}

detector | config | frame_time	{"value": 1.0}
detector | config | count_time	{"value": 0.99}
detector | config | ntriggers_skipped	{"value": 0}

The configuration above will generate 4 images, with a total of 5 external triggers needed (ntrigger + 1). The additional trigger is required to end the last frame of the series.

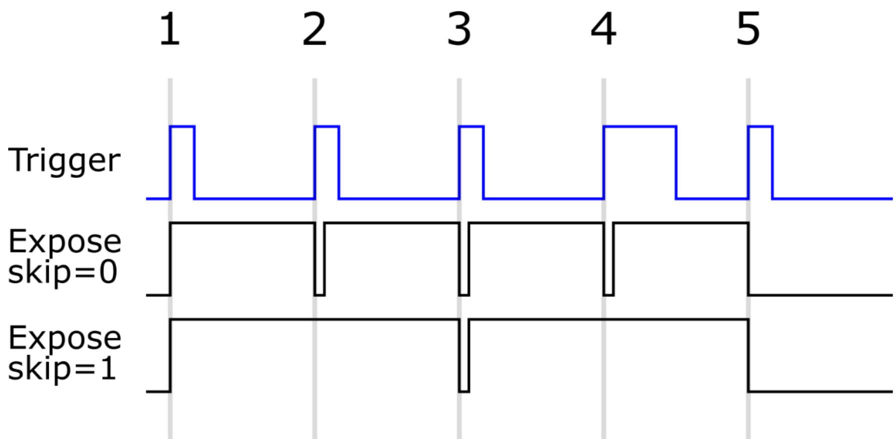


Figure 15. EIES - Schematic illustrating the "eies" mode and the usage of the ntriggers_skipped parameter.

10. PIXEL MASK

10.1. Applying the pixel mask

The detector configuration parameter pixel_mask_applied enables (True) or disables (False) applying the pixel mask on the acquired data. Every threshold has its own pixel mask. These masks may differ from one another. If true (default), pixels which have any bit set in the pixel_mask are flagged with ($2^{\text{image bit depth}}-1$). Please consult the API Reference for details on the detector configuration parameter pixel_mask.

10.2. Updating the pixel mask

10.2.1. Overview

Updating the pixel mask of a POLLUX detector involves four basic steps:

1. Retrieving the current pixel mask from the detector system via the SIMPLON API.
2. Manipulating the pixel mask to add or update pixels.
3. Uploading the updated pixel mask to the detector system via the SIMPLON API.
4. Persistently storing the updated pixel mask on the detector system by sending the detector command "arm".

10.2.2. Retrieving the current mask from the detector system

The pixel mask can be retrieved from the detector system by a GET request on the detector configuration parameter threshold/n/pixel_mask where *n* is the threshold number. The data of the pixel mask is retrieved either as tiff or in JSON serialization by choosing application/tiff or application/json in the get request accordingly.

10.2.3. Manipulating the pixel mask

NOTICE
For details about the pixel values in the pixel mask and their meaning, consult the API Reference.

TIFF

If the pixel mask is retrieved and stored as tiff, the uint32 data in the tiff file can be manipulated with ALBULA API.

JSON

If the pixel mask is retrieved as JSON, the HTTP reply has to be parsed correctly into an array. Please see the example below and the API Reference for details. The values in this array can then be manipulated to reflect the required updates of the pixel mask. After updating the array, it has to be serialized again in JSON according to the specifications in the API Reference.

10.2.4. Uploading and storing the pixel mask

The pixel mask is uploaded by sending a PUT request on the detector configuration parameter pixel_mask with the new mask as data. After sending the detector command “arm”, the updated pixel mask is permanently stored on the detector system.

10.2.5. Python example

The following Python code using common libraries provides a simple example for updating the pixel mask:

```python
import json
import numpy
import requests
from base64 import b64encode, b64decode

# Simple class to get/set a pixel mask or flatfield
class DMaskClient:
    def __init__(self, ip , port=80):
        self._ip = ip
        self._port = port

    def mask(self, mask):
        url = f'http://{self._ip}:{self._port}/detector/api/1.8.0/config/{mask}'
        darray = requests.get(url).json()['value']
        return numpy.frombuffer(b64decode(darray['data']),
            dtype=numpy.dtype(str(darray['type']))).reshape(darray['shape'])

    def setMask(self, ndarray, mask):
        url = f'http://{self._ip}:{self._port}/detector/api/1.8.0/config/{mask}'
        data_json = json.dumps({'value': {
                                '__darray__': (1,0,0),
                                'type': ndarray.dtype.str,
                                'shape': ndarray.shape,
                                'filters': ['base64'],
                                'data': b64encode(ndarray.data).decode('ascii') }})
        requests.put(url, data_json)

if __name__ == '__main__':
    # Create instance of the mask client
    maskClient = DMaskClient('169.254.254.1')

    # Get the pixel mask
    pixelMask = maskClient.mask('pixel_mask')

    # Copy the mask to writeable buffer, necessary for numpy>=1.16.0
    pixelMask = numpy.copy(pixelMask)

    # Set a new dead pixel [y,x]
    pixelMask[123, 234] = 2

    # Set a new noisy pixel [y,x]
    pixelMask[234, 123] = 8

    # Upload the updated pixel mask
    maskClient.setMask(pixelMask, 'pixel_mask')

    # Ensure that you 'arm' the detector to save the new pixel masks persistently
```

11. TROUBLESHOOTING

Table 12 provides an overview of possible problems with the detector and instructions in order to solve the problem. If the problem you are experiencing is not listed below or if the instructions do not help, please contact support@dectris.com. A bug report can be created via the web interface (see Section 7.1).

The LEDs on the Detector Control System (DCS) can provide valuable information for troubleshooting. Check Section 5.4.2 for further information.

Table 12. Troubleshooting common issues

Problem	Possible Cause	Solution
DCS not responsive when attempting to power it on and the status LED remains unlit.	Device not connected to power.	Make sure the power supply unit is properly connected and the power is switched on.
	Wrong power supply.	Only use the power supply delivered with the instrument. Check Section 4.2.3 for power supply specifications.
DCS not responsive when attempting to power it on and the status LED is blinking.	Fuse has been triggered.	1. Check the power supply and make sure it meets the specifications described in Section 4.2.3. 2. Turn off the DCS. 3. Unplug the power supply from the mains connection and wait 20 seconds. 4. Plug the power supply back in and turn the DCS back on. 5. If the problem persists, contact DECTRIS® support.
No connection to the detector head.	The detector head is not connected to the DCS or the connection is faulty.	• Make sure that the USB-C data cable is properly connected between the detector head and the DCS. • Reseat the cable on both ends if necessary. • Make sure that only the cable provided by DECTRIS® is used. Plugging a different cable into the detector head or the DCS may cause damage.

TRADEMARKS AND PATENTS

Registered Trademarks ®

- "DECTRIS”: AU, AUS, CH, CN, DE, FR, IT, JP, KR, UK, USA
- ”DETECTING THE FUTURE”: AUS, CH, CN, EU, JP, KR, UK, USA
- ”DECTRIS INSTANT RETRIGGER”: AUS, CH, CN, EU, JP, KR, UK, USA
- ”DECTRIS EIGER”: AUS, CH, CN, EU, JP, KR, UK, USA
- ”DECTRIS PILATUS”: AUS, CH, CN, EU, JP, KR, USA
- "DECTRIS MYTHEN": AUS, CH, CN, EU, JP, KR, UK, USA
- ”DECTRIS SELUN”: AUS, CH, CN, EU, JP, KR, UK, USA
- ”Lines-ROI”: CH, EU, JP
- "POLLUX": AUS, CH, CN, EU, JP, KR, UK
- "JUNGFRAU": CH, CN, EU, JP, KR, UK, USA