HANDBOOK

eGrabber

Gigelink Handbook

4400 eGrabber Gigelink
4400-EV eGrabber Gigelink 30-day evaluation license

This documentation is provided with **eGrabber 24.03.2** (doc build **2186**).
www.euresys.com

Contents

PART I

GETTING STARTED

1. Introduction

About this guide

This manual guides you through the standard preparation, installation and configuration so that you can connect to and set up a GigE Vision camera using the **Gigelink** library. You will mainly use **eGrabber Studio** for the initial connection and configuration.

This manual might refer to other documentation sources available in the eGrabber online documentation.

About eGrabber Gigelink

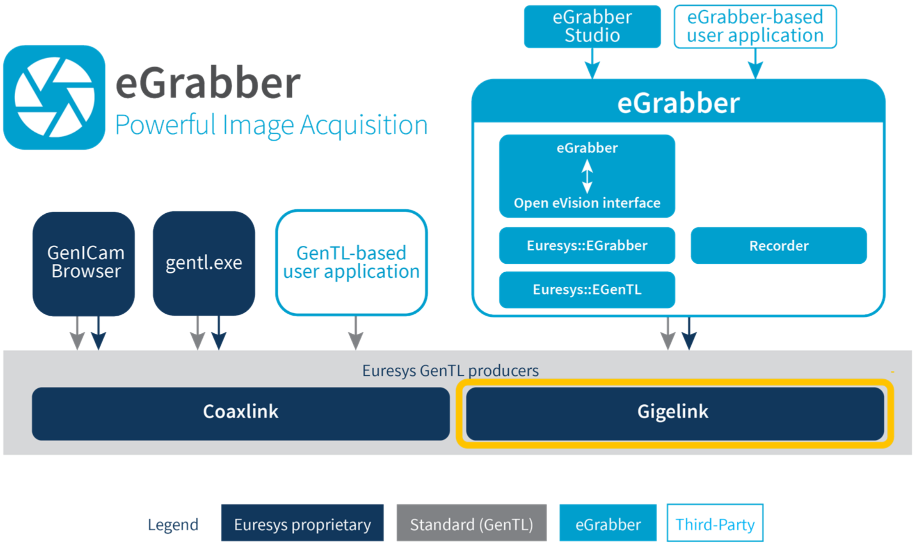
eGrabber Gigelink is a licensed library delivered with the **eGrabber** package. It provides a hardware-independent access to GigE Vision cameras.

With **eGrabber Gigelink**, programmers can use the **eGrabber** concepts, objects, and function calls to acquire images from GigE Vision cameras, in the same way as with CoaXPress cameras and the Coaxlink cards.

eGrabber Gigelink consists in the following software components:

- **eGrabber Gigelink** GenTL producer (gigelink.cti)
- GenApi implementation for Gigelink
- Optional filter driver to improve performances (by reducing the CPU load)

eGrabber Gigelink is compatible with GenICam and the **eGrabber Studio** application.



2. Installation

2.1. Overview

The installation consists of the following steps:

Preliminary steps

- Select the Network Interface Card

  Picture quality with GigE Vision highly depends on optional features of Ethernet adapters such as Jumbo Frame packets and high amount of on-board buffers. Contact the Euresys Sales teams should you need advice on the network interface cards providing the required features and resources.

- "Purchase a Gigelink License Key" on page 9

Installation steps

1. Install the **eGrabber** driver package

   The package includes all **eGrabber Gigelink** software components.

   Refer to "Driver Installation Procedure" on page 10

2. Activate the **Gigelink** license

   This guide describes how to activate an **eGrabber Gigelink** license using the **Neo License Manager** application.

   The **Neo License Manager** application is included in the **eGrabber** package and is available from the Windows Start menu.

   Refer to "Activate from the User Interface" on page 11

NOTE

Besides using the **Neo License Manager** user interface, you can also activate the **eGrabber Gigelink** license using the **Neo License Manager** command line (online or offline). Refer to the documentation on **Neo Licensing System** for more information about these activation methods.

2.2. Purchase a Gigelink License Key

Overview

To be able to use **eGrabber Gigelink**, you need to buy a **eGrabber Gigelink** license from **Euresys** or one of its distributors.

When you buy a license, you receive a **Ticket ID** corresponding to the license.

You will then use this **Ticket ID** to activate your license.

You can activate your license in one of the following ways:

- on the computer, as a software license
- on a **Neo USB Dongle** that you can buy from **Euresys**.

See also: For more information, refer to the user guide D214-Neo Licensing System available in https://documentation.euresys.com/Products/COAXLINK/COAXLINK/en-us/Content/00_Home/PDF_Guides.htm

Advantages and disadvantages of software license vs. dongle license

Software license - advantages

- You spare the use of an additional USB port on the computer, compared to the dongle license.

Software license - disadvantages

- If the computer fails, the **eGrabber Gigelink** license is lost with the computer.
- If you accidentally authenticate to a different computer, you are not allowed to re-authenticate.

Dongle license - advantages

- If you need to change the computer, you can easily transfer the **eGrabber Gigelink** license as this is stored on your USB dongle.

Dongle license - disadvantages

- The USB dongle must always be plugged into the computer.
- You have to buy the USB dongle in addition to the **eGrabber Gigelink** license key.

✓ TIP

We recommend to use a **Neo USB Dongle** for a more secure and an easily portable license storage solution.

2.3. Driver Installation Procedure

WARNING

Prior to executing this procedure, read the Important Notices section of the **eGrabber** release notes.

1. Open the support page of the Euresys website: https://www.euresys.com/support/ and click on the *Coaxlink series* icon to open the Coaxlink download area.

NOTE: The Euresys website download area may require user authentication. The user ID and password are not obtained, they are chosen by the user. Access is free and unrestricted.

2. Click on *eGrabber for Coaxlink and Grablink Duo* to display the file list corresponding to the latest available **eGrabber** release.

3. In the *Setup Files* section, select the file corresponding to your operating system and your processor architecture:
   egrabber-<OS>-<ARCH>(-<TYPE>)-<MA.MI.RE.BU>.<EXT> with
   - <OS> = {linux, macos, win7, win10} operating system
   - <ARCH> = {aarch64, x86_64} processor architecture
   - <TYPE> = {kext, dext} driver type (for macOS only)
   - <MA.MI.RE.BU> major version, minor version, revision and build numbers of the driver package
   - <EXT> = {exe, tar.gz, pkg} file type

See also: Supported Operating Systems topic in the Release Specification section of the **eGrabber** release notes.

NOTE

- For an installation on Windows 7 or 8.1, use the egrabber-win7-x86_64-<MA.MI.RE.BU>.exe setup file.
- For an installation on Windows 10 or 11, use the egrabber-win10-x86_64-<MA.MI.RE.BU>.exe setup file.

4. Launch the installer tool to install the driver files and software tools on your PC.

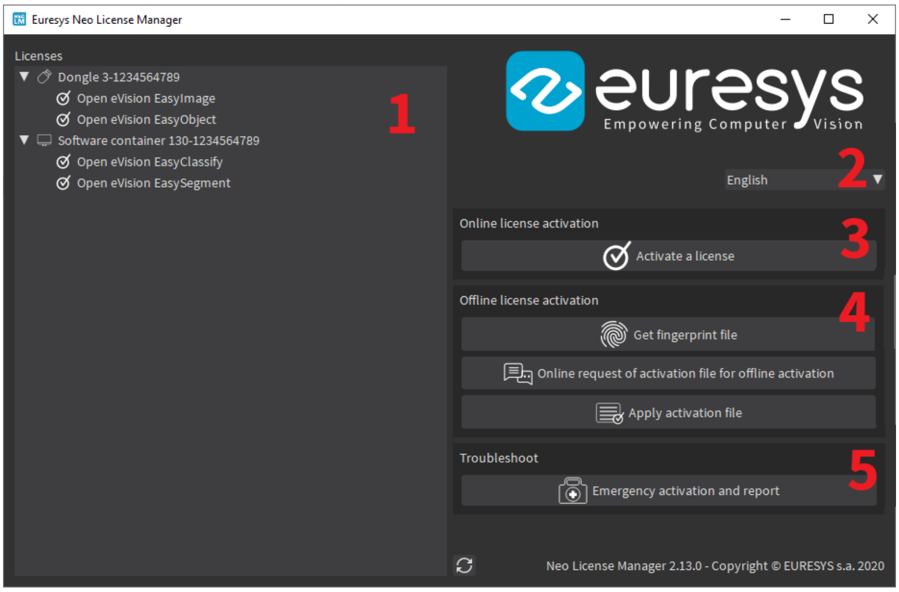
NOTE: If you have an existing **eGrabber driver** (or a **Coaxlink driver**) already installed, the installer tool prompts you to uninstall it before being able to continue. Otherwise, it prompts you for the selection of the destination folder.

2.4. Activate from the User Interface

The Neo License Manager user interface

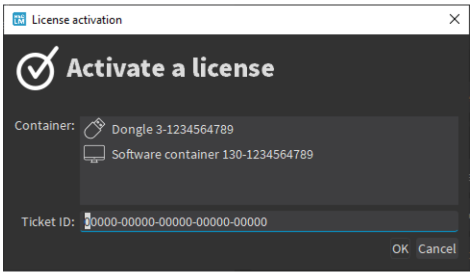
The **Neo License Manager** is organized in 5 zones:

1. A list of the license containers and the licenses they contain.
2. The interface language configuration.
3. The online **Neo License** activation.
4. The offline **Neo License** activation.
5. The troubleshooting.


Euresys Neo License Manager
Licenses
Dongle 3-1234564789
Open eVision EasyImage
Open eVision EasyObject
Software container 130-1234564789
Open eVision EasyClassify
Open eVision EasySegment
1
euresys
Empowering Computer Vision
English
2
Online license activation
Activate a license
3
Offline license activation
Get fingerprint file
Online request of activation file for offline activation
Apply activation file
4
Troubleshoot
Emergency activation and report
5
Neo License Manager 2.13.0 - Copyright © EURESYS s.a. 2020

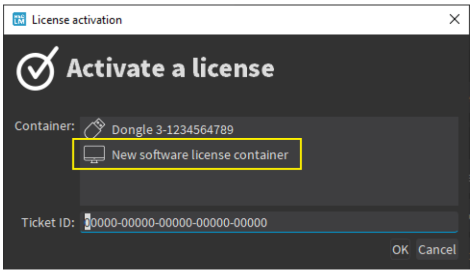
Activate an online license

1. Click on the Activate a license button to open the License activation dialog.



2. Enter the required information:
   - □ The container on which you want to activate the license.
   - □ The **Ticket ID** (5 groups of 5 alphanumeric characters separated by a dash).

If there is no software license container yet, the dialog proposes you to create a new one.



System reinstallation

If you have to reinstall the system and lost your licenses:

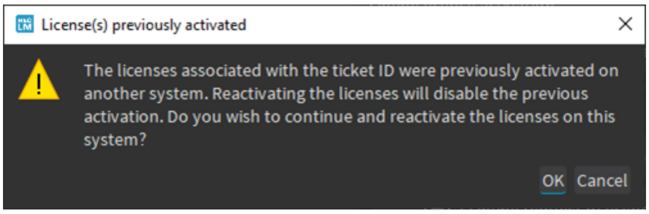
1. Activate the license with the old **Ticket ID** to try to recover all the licenses associated to the container where you first activated this **Ticket ID**.

   This only works if:

   - It is done in a new, empty container.
   - The system is recognizable (the hardware is more or less the same).

2. If this does not work or if you lost your licenses and the system has changed significantly (for instance after the failure and replacement of a hardware component), you need to request a reactivation:

   - Contact the support to obtain a reactivation authorization for the corresponding **Ticket ID**.
   - Once the reactivation is authorized, perform the license activation procedure as usual.
   - The **Neo License Manager** asks for the confirmation of the reactivation.



The licenses associated with the ticket ID were previously activated on another system. Reactivating the licenses will disable the previous activation. Do you wish to continue and reactivate the licenses on this system?

NOTE

The reactivation disables the previous activation on the other system.

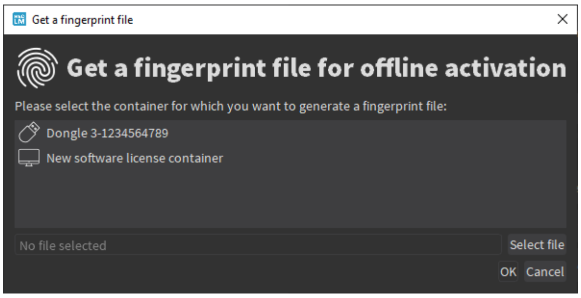
Activate an offline license

The offline license activation allows you to activate licenses on a *machine A* that is not connected to the Internet through a *machine B* that is connected to the Internet.

The offline license activation is a 3 steps procedure:

1. On *machine A*:

   a. Click on the Get fingerprint file button to open the Get a fingerprint file dialog.



Get a fingerprint file

Get a fingerprint file for offline activation

Please select the container for which you want to generate a fingerprint file:

Dongle 3-1234564789

New software license container

No file selected

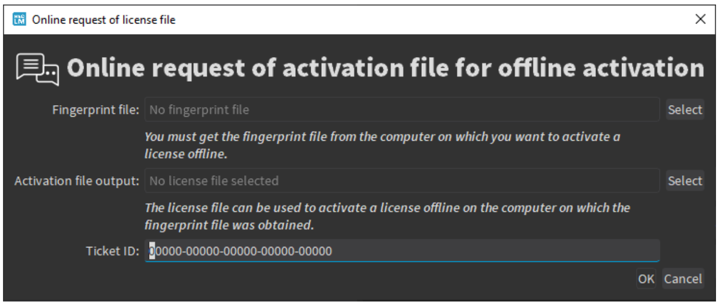
   b. Enter the required information::

   □ The license container on which you want to activate the license.

   □ The path to save the fingerprint file.

2. On *machine B*:

   a. Click on the Online request of license file for offline activation button to request the corresponding license file.



Online request of license file

Online request of activation file for offline activation

Fingerprint file: No fingerprint file

You must get the fingerprint file from the computer on which you want to activate a license offline.

Activation file output: No license file selected

The license file can be used to activate a license offline on the computer on which the fingerprint file was obtained.

Ticket ID: 00000-00000-00000-00000-00000

   b. Enter the required information:

   □ The Fingerprint file from *machine A*.

   □ The License file output path to save the license file.

   □ The **Ticket ID** (5 groups of 5 alphanumeric characters separated by a dash) for the license.

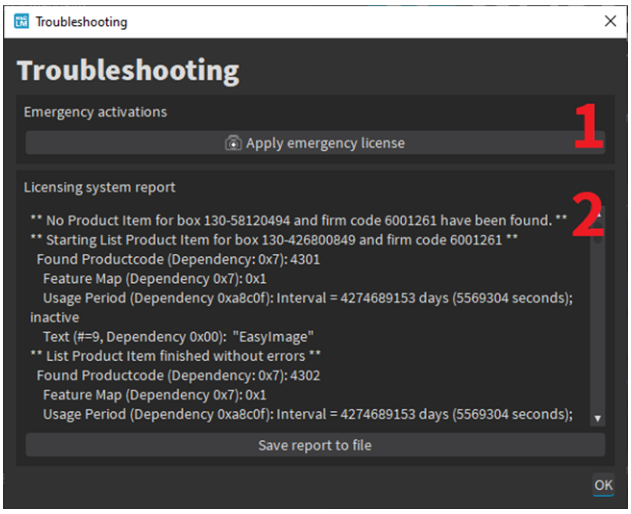
3. On *machine A*:

   a. Click on the Apply license file button.

   b. Select the license file generated in step 2.

Troubleshooting

1. Click on the Emergency activation and report button to open the Troubleshooting dialog.



```
 ** No Product Item for box 130-58120494 and firm code 6001261 have been found. **
 ** Starting List Product Item for box 130-426800849 and firm code 6001261 **
   Found Productcode (Dependency: 0x7): 4301
     Feature Map (Dependency 0x7): 0x1
     Usage Period (Dependency 0xa8c0f): Interval = 4274689153 days (5569304 seconds);
 inactive
     Text (#=9, Dependency 0x00):  "EasyImage"
 ** List Product Item finished without errors **
   Found Productcode (Dependency: 0x7): 4302
     Feature Map (Dependency 0x7): 0x1
     Usage Period (Dependency 0xa8c0f): Interval = 4274689153 days (5569304 seconds);
```

2. Click on the Apply emergency license button to temporarily activate licenses:
   - These licenses activate all **Euresys** products.
   - They are valid for 8 days.
   - You can repeat this a maximum of 3 times.
   - For security reasons, you cannot activate emergency licenses on virtual machines.
3. Click on the Save report to file button to generate a licensing system report and save it to a file.
   - The support team may require this report when you have a licensing issue.

3. Configuration

3.1. Configure the Network Interface Card

Overview

This section describes how to configure the network interface card (NIC) in MS Windows once the network card is installed.

The configuration includes the following steps:

1. "Access the network interface card settings" on page 15
2. "Check the filter driver" on page 16
3. "Set the IP address of the network interface card" on page 17
4. "Enable the Jumbo Packet settings" on page 20
5. "Set the Receive Buffer settings" on page 21
6. "Set Interrupt Moderation Settings" on page 22

Access the network interface card settings

To access the Network Interface Card settings:

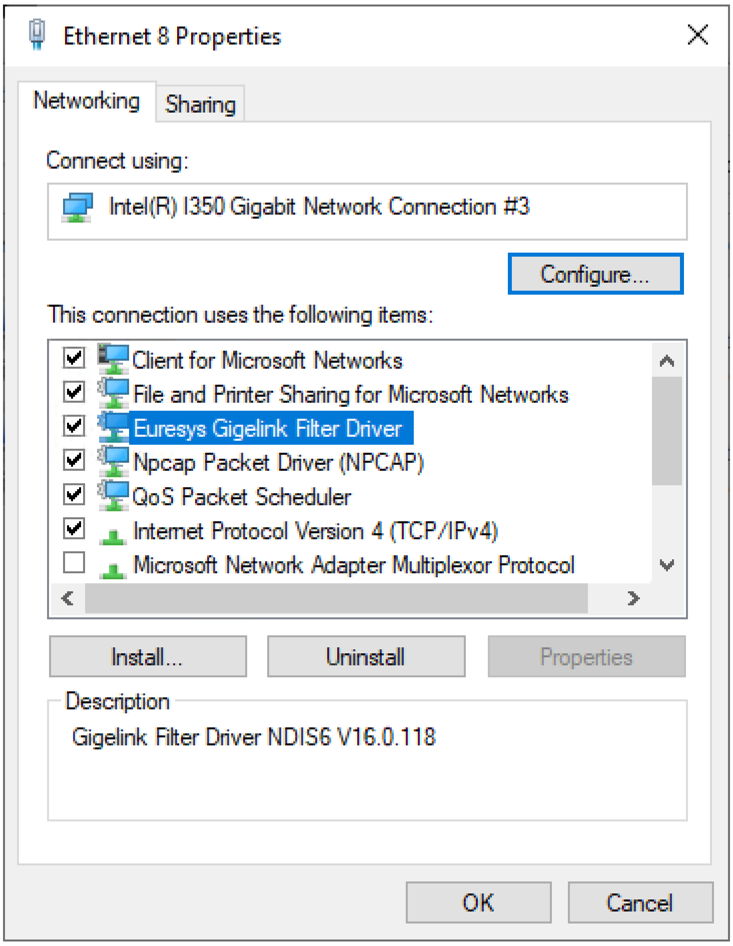
1. Open the Ethernet settings from the Windows Start menu
2. Select Change adapter options to access the list of network connections in Control Panel\Network and Internet\Network Connections.
3. Right-click the network interface card you want to configure and select Properties from the contextual menu.

The Properties dialog box of your Network Interface Card opens.

Check the filter driver

When the **eGrabber** driver is installed, the filter driver is enabled by default. In an operational environment, we recommend to keep the filter driver enabled. Check that the filter driver is enabled in your NIC properties, as shown on the screenshot below.

The filter driver operates directly with the NIC driver to intercept and handle the stream packets. It then sends these priority stream data directly to the application buffer, bypassing the system network IP stack.



Ethernet 8 Properties

Connect using: Intel(R) I350 Gigabit Network Connection #3

This connection uses the following items:

- ☑ Client for Microsoft Networks
- ☑ File and Printer Sharing for Microsoft Networks
- ☑ Euresys Gigelink Filter Driver
- ☑ Npcap Packet Driver (NPCAP)
- ☑ QoS Packet Scheduler
- ☑ Internet Protocol Version 4 (TCP/IPv4)
- □ Microsoft Network Adapter Multiplexor Protocol

Description: Gigelink Filter Driver NDIS6 V16.0.118

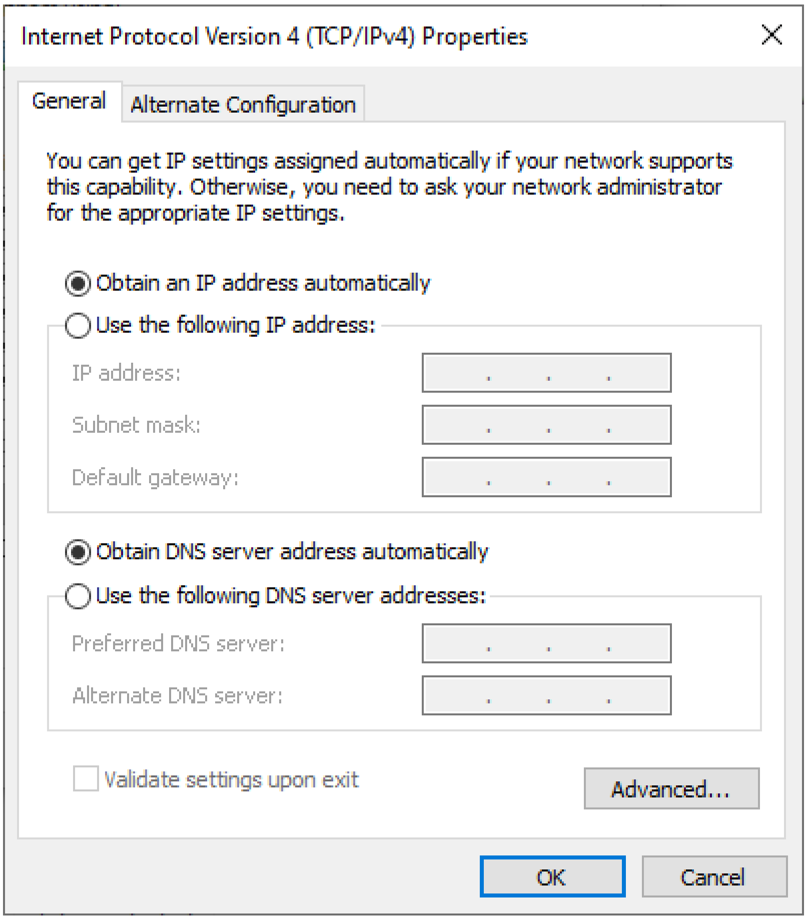
For test purposes, you can disable the filter driver in the NIC properties.

Set the IP address of the network interface card

In the Properties dialog box, select Internet Protocol Version 4 (TCP/IPv4) and click Properties.

With DHCP Server

- Select Obtain an IP address automatically and click OK to validate.



Internet Protocol Version 4 (TCP/IPv4) Properties

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

● Obtain an IP address automatically

○ Use the following IP address:

IP address: . . .

Subnet mask: . . .

Default gateway: . . .

● Obtain DNS server address automatically

○ Use the following DNS server addresses:

Preferred DNS server: . . .

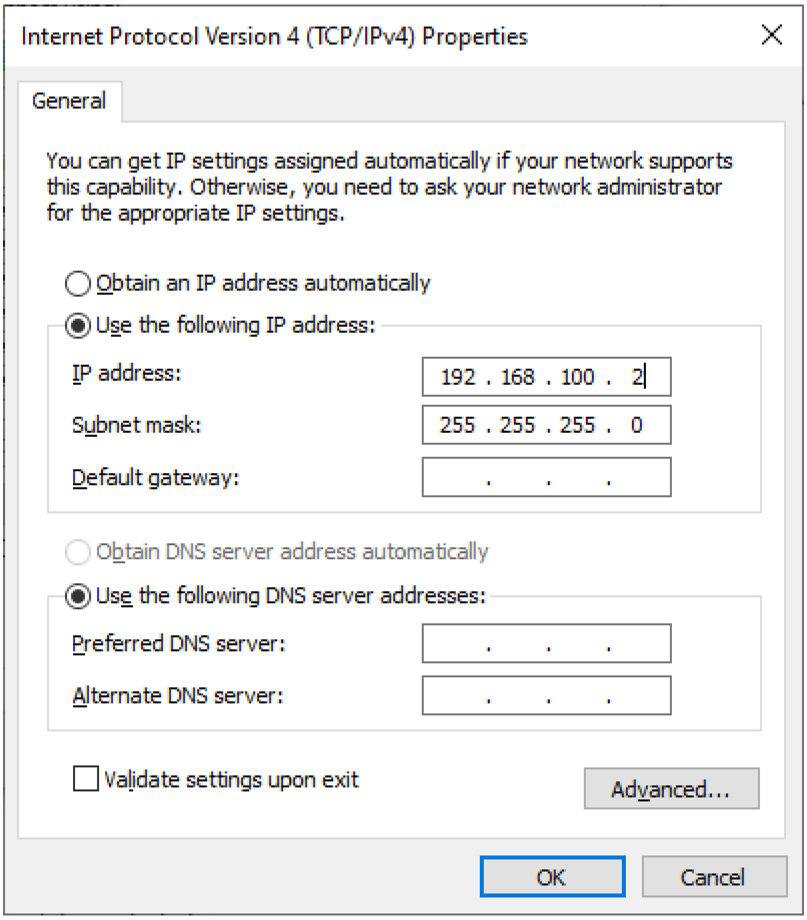
Alternate DNS server: . . .

□ Validate settings upon exit

Without DHCP server

1. Select Use the following IP address and define the IP address settings, for example as follows:
   - □ IP address : 192.168.XXX.2, with XXX in the range [0-255].
   - □ Subnet mask : 255.255.255.0.
   - □ Default gateway remains empty.

> WARNING
> If you use a multi-port NIC to connect to multiple cameras point-to-point, each point-to-point wire is considered as a separate LAN, you should define fixed NIC addresses and subnets for each port so that they do not collide, for example 192.168.100.2, 192.168.101.2, 192.168.102.2, etc.



Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 192 . 168 . 100 . 2

Subnet mask: 255 . 255 . 255 . 0

Default gateway: . . .

○ Obtain DNS server address automatically

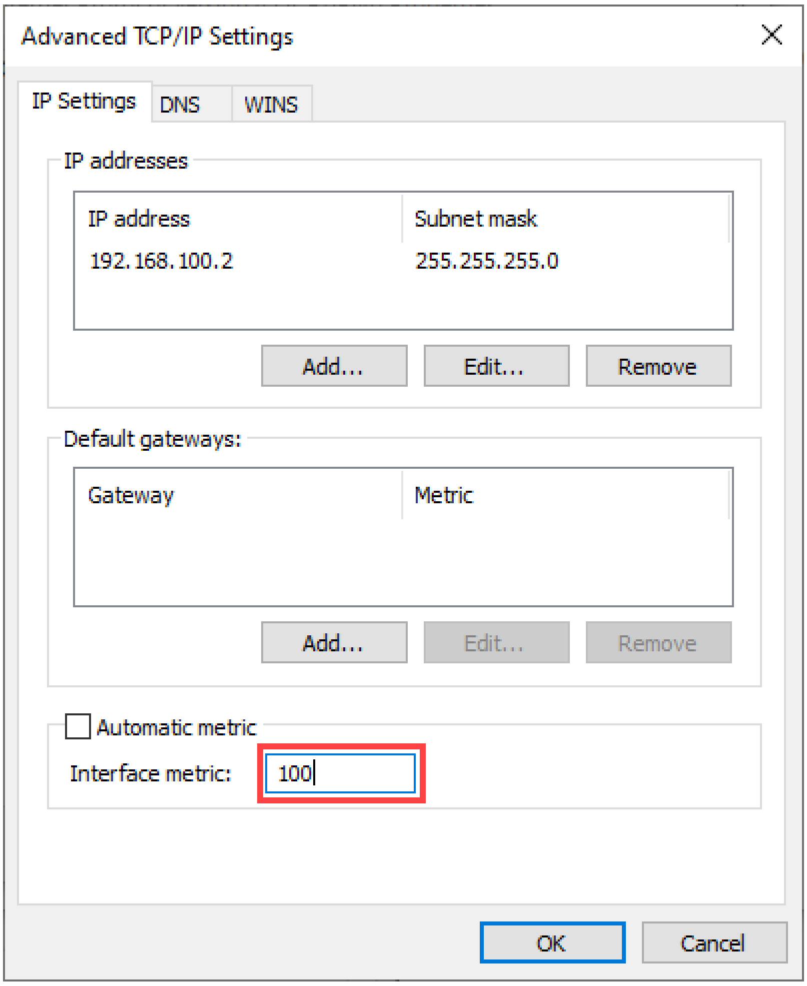
● Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

□ Validate settings upon exit

2. Click Advanced and set the metric to 100[1] in the IP Settings tab, if it is relevant to help keep unrelated traffic out of the GEV network:



Advanced TCP/IP Settings

IP Settings | DNS | WINS

IP addresses

IP address	Subnet mask
192.168.100.2	255.255.255.0

Default gateways:

Gateway	Metric

☐ Automatic metric

Interface metric: 100

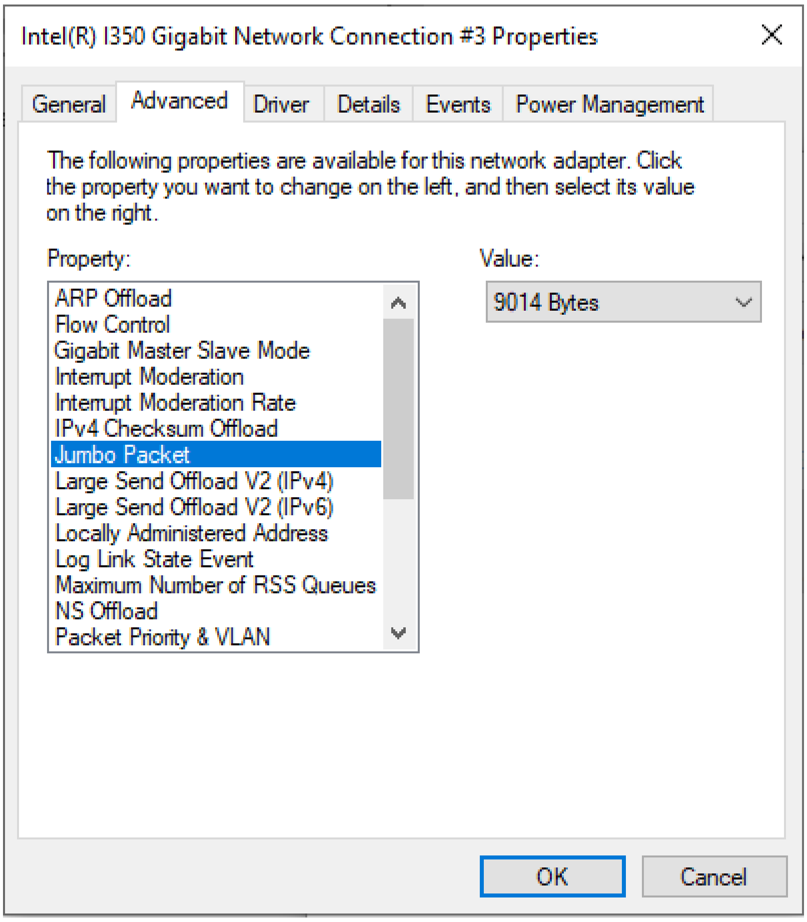
3. Click OK twice to validate and come back to the Properties dialog box.

[1] This value is provided for reference.

Enable the Jumbo Packet settings

Enable the Jumbo Packet settings allows you to transfer and receive larger streaming packets and reduce processing overhead.

1. In the Properties dialog box, click Configure.
2. In the dialog box, select the Advanced tab, select the Jumbo Packet in the Property list.
3. In the Value field, select the value 9014 Bytes as the maximum transmission unit (MTU) for the jumbo packets:



Intel(R) I350 Gigabit Network Connection #3 Properties

The following properties are available for this network adapter. Click the property you want to change on the left, and then select its value on the right.

Property:

- ARP Offload
- Flow Control
- Gigabit Master Slave Mode
- Interrupt Moderation
- Interrupt Moderation Rate
- IPv4 Checksum Offload
- Jumbo Packet
- Large Send Offload V2 (IPv4)
- Large Send Offload V2 (IPv6)
- Locally Administered Address
- Log Link State Event
- Maximum Number of RSS Queues
- NS Offload
- Packet Priority & VLAN

Value: 9014 Bytes

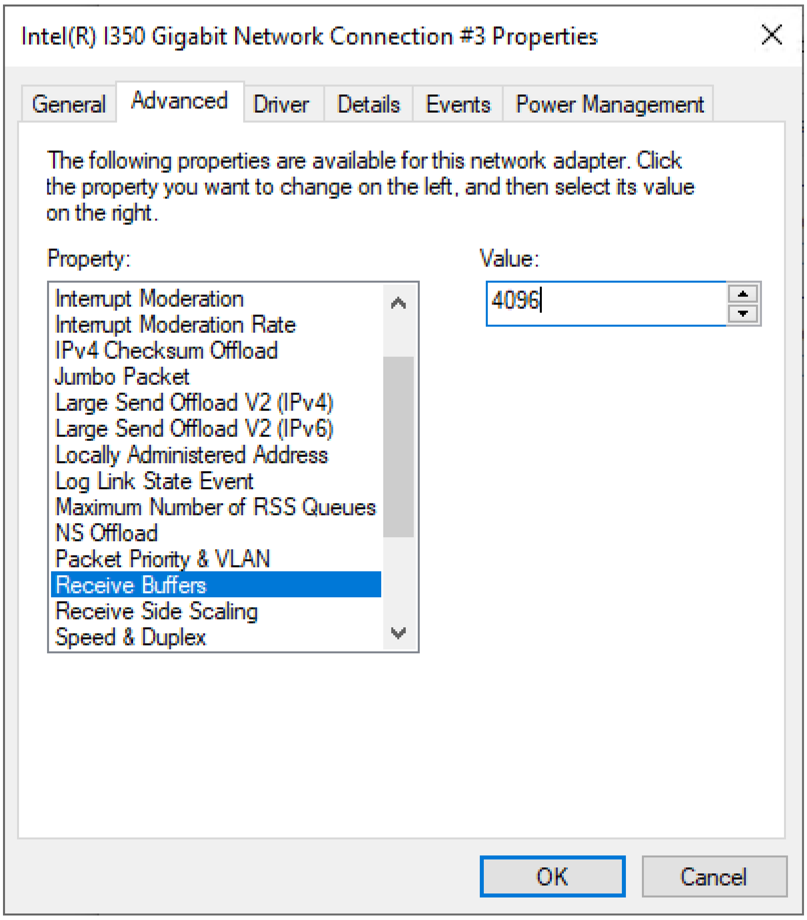
4. Set the next advanced parameter described below.

NOTE
Make sure that you enable the jumbo packets on the camera side (see "Set packet size to maximum" on page 28), and that other possible devices in the network between the camera and the host computer have jumbo frames enabled with the same MTU.

Set the Receive Buffer settings

The Receive Buffers setting allows you to define how much system memory is dedicated to store received packets. Increasing the buffers helps improve the reliability of the network traffic, but consumes system memory.

1. In the same Advanced tab where you have set the jumbo packets, select the Receive Buffers in the Property list.
2. In the Value field, select the value 4096[1] for receive buffers:



Intel(R) I350 Gigabit Network Connection #3 Properties

The following properties are available for this network adapter. Click the property you want to change on the left, and then select its value on the right.

Property:

- Interrupt Moderation
- Interrupt Moderation Rate
- IPv4 Checksum Offload
- Jumbo Packet
- Large Send Offload V2 (IPv4)
- Large Send Offload V2 (IPv6)
- Locally Administered Address
- Log Link State Event
- Maximum Number of RSS Queues
- NS Offload
- Packet Priority & VLAN
- Receive Buffers
- Receive Side Scaling
- Speed & Duplex

Value: 4096

3. Set the next advanced parameter described below.

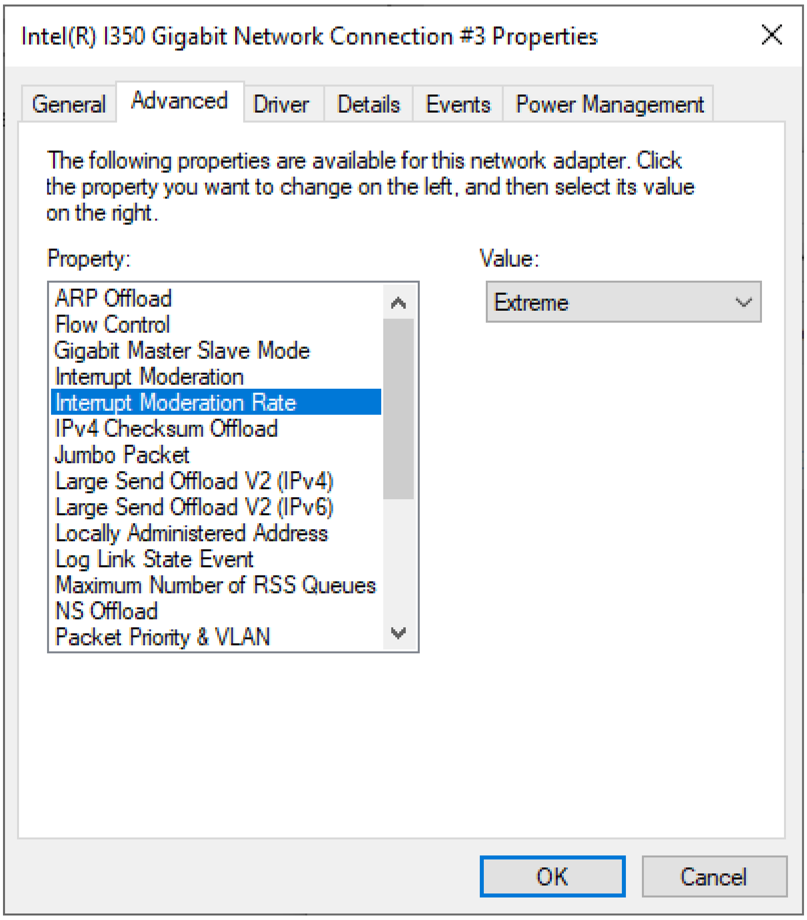
[1] This value is provided for reference.

Set Interrupt Moderation Settings

The Interrupt Moderation settings allow you to define how often the adapter interrupts the system to handle the incoming (and outgoing) packets.

With GigE Vision cameras, it is more appropriate to set the highest level for interrupt moderation rate. This means that the NIC does not interrupt the processes, which reduces the CPU utilization. Instead the NIC driver actively decides when to check and handle the received packets.

1. In the same Advanced tab where you have set the receive buffers, select Interrupt Moderation in the Property list and check that the setting is enabled.
2. Select Interrupt Moderation Rate in the Property list.
3. In the Value field, select Extreme.



Intel(R) I350 Gigabit Network Connection #3 Properties

The following properties are available for this network adapter. Click the property you want to change on the left, and then select its value on the right.

Property:

- ARP Offload
- Flow Control
- Gigabit Master Slave Mode
- Interrupt Moderation
- Interrupt Moderation Rate
- IPv4 Checksum Offload
- Jumbo Packet
- Large Send Offload V2 (IPv4)
- Large Send Offload V2 (IPv6)
- Locally Administered Address
- Log Link State Event
- Maximum Number of RSS Queues
- NS Offload
- Packet Priority & VLAN

Value: Extreme

4. Click OK to validate all settings you have defined and come back to the panel displaying the network connections.

3.2. Configure the Firewall

Most **GigE Vision** cameras support Firewall Traversal technique that allows seamless inter-operation through typical host firewalls such as **Windows Defender** firewall.

Older models, however, may lack this feature and require that the applications using **eGrabber Gigelink** (as well as **eGrabber Studio**) are explicitly added as "allowed apps" in the firewall.

In a standard installation, the applications to authorize are in c:\program files\euresys\egrabber\bin and c:\program files\euresys\egrabber\studio.

3.3. Configure the Camera Network Settings

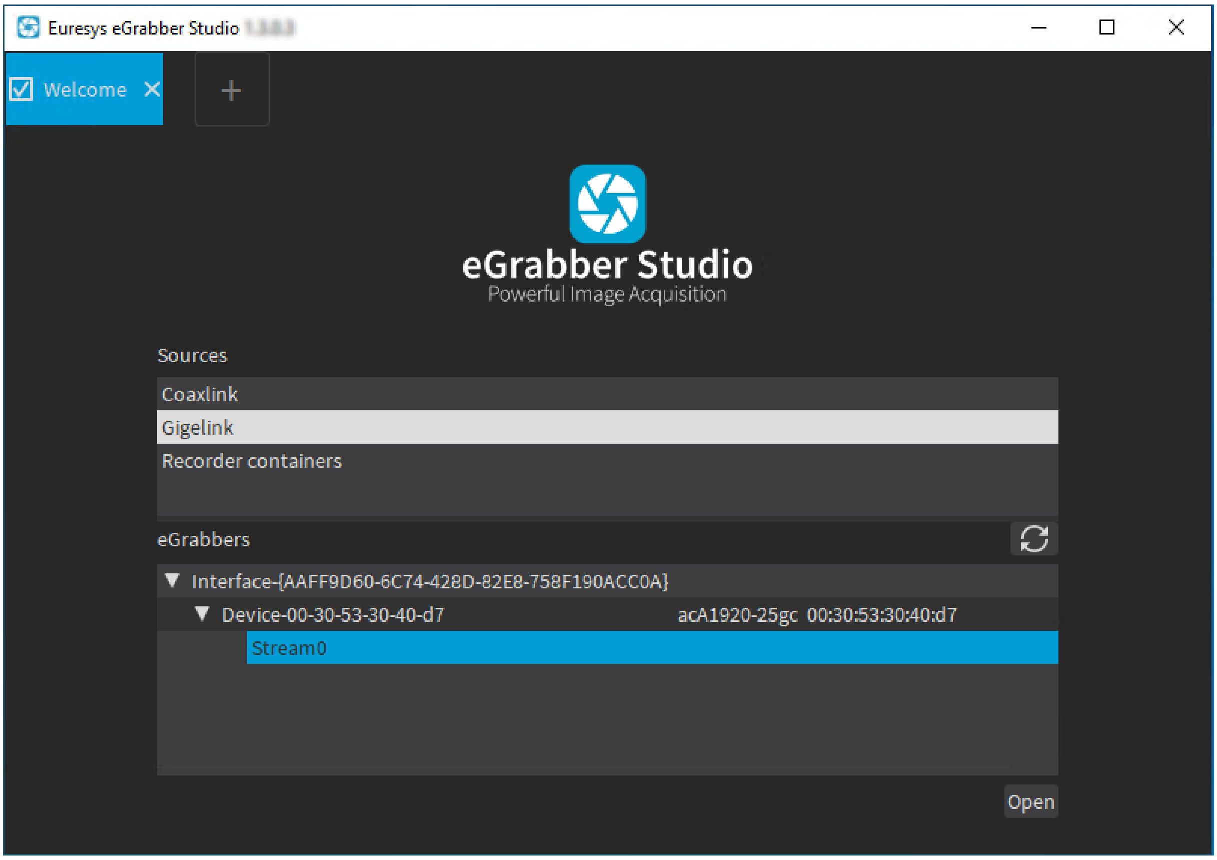
Overview

Follow the steps detailed below to configure the camera network settings. You can perform all steps in **eGrabber Studio**.

1. "Select the GigE Vision camera in eGrabber Studio" on page 24
2. "Configure the camera IP settings" on page 25 (if not defined yet)
3. "Set packet size to maximum" on page 28
4. "Set Packet Delay" on page 28 (if necessary)

Select the GigE Vision camera in eGrabber Studio

1. Open **Studio** from the Windows Start menu.
2. In the Sources area, select **Gigelink**:

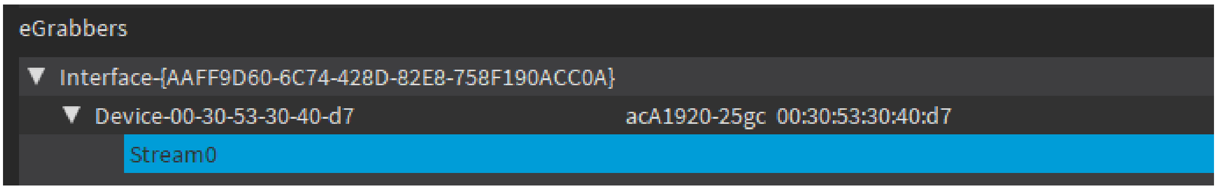


3. In the eGrabbers area, do one of the following actions. See "Information on Gigelink sources " on page 25 about displayed information.
   - □ If the desired interface, device and stream are displayed, double-click the **Gigelink** stream:

     The stream opens directly in a new tab. Go directly to the section "Set packet size to maximum" on page 28.
   - □ If only the desired interface is displayed, the camera is not connected. Thus double-click the **Gigelink** interface. The interface opens in a new tab. Go to the next section.

Information on Gigelink sources

In the Welcome window, the following information related to the interface (and device) are displayed for each **Gigelink** source:



eGrabbers

▼ Interface-{AAFF9D60-6C74-428D-82E8-758F190ACC0A}

▼ Device-00-30-53-30-40-d7 acA1920-25gc 00:30:53:30:40:d7

Stream0

Source type	Levels	Displayed information (from left to right)
Gigelink	1 - Interface	● Name of the **Gigelink** interface (based on the UID of the network interface connected to the **Gigelink** device)
	2 - Device	● Name of the **Gigelink** device ● Model of the remote device ● MAC address of the remote device
	3 - Stream	● Name of the **Gigelink** stream

Configure the camera IP settings

You can configure the camera IP settings using a dedicated script or **eGrabber Studio**.

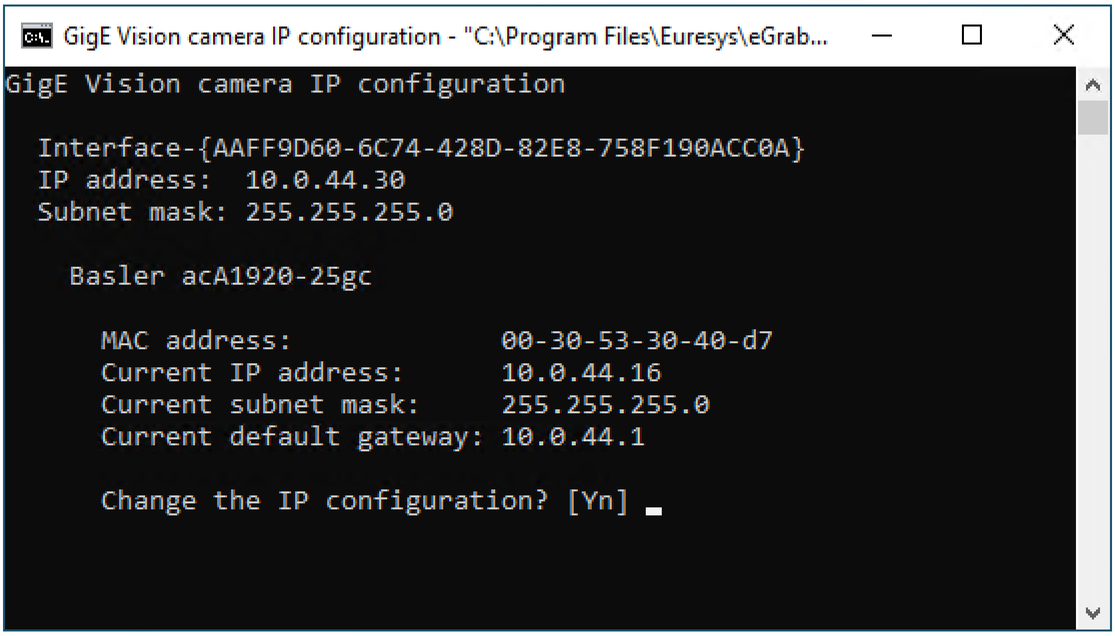
Configure the camera IP settings using a script

NOTE

You can use this script for any change in the IP configuration of the camera.

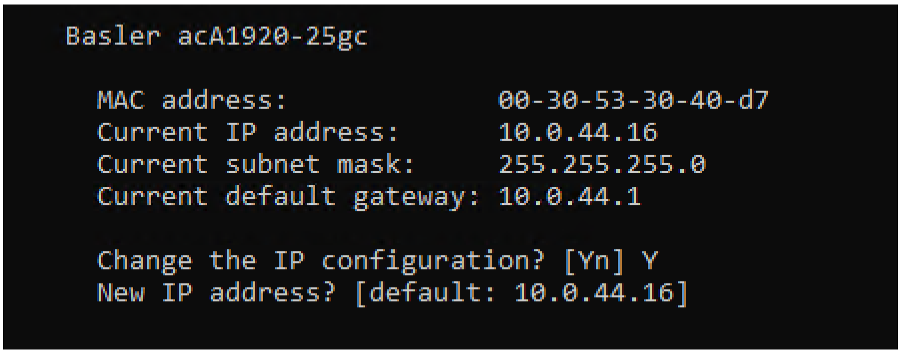
1. Locate and open the script **GigE Vision camera IP configuration** in one of the following ways:
   - □ Use the search field in the Windows Start menu.
   - □ Use a command line interface if you know the script location. The default location is : C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Euresys eGrabber

   The script opens and lists the network interfaces and the GigE Vision cameras attached to them:


GigE Vision camera IP configuration - "C:\Program Files\Euresys\eGrab...

```
GigE Vision camera IP configuration

  Interface-{AAFF9D60-6C74-428D-82E8-758F190ACC0A}
  IP address:  10.0.44.30
  Subnet mask: 255.255.255.0

    Basler acA1920-25gc

      MAC address:              00-30-53-30-40-d7
      Current IP address:       10.0.44.16
      Current subnet mask:      255.255.255.0
      Current default gateway: 10.0.44.1

      Change the IP configuration? [Yn] _
```

2. If any listed camera is misconfigured (that means that it isn't using the same LAN as the network interface card it is attached to), type Y on the line below its IP settings:



```
    Basler acA1920-25gc

      MAC address:              00-30-53-30-40-d7
      Current IP address:       10.0.44.16
      Current subnet mask:      255.255.255.0
      Current default gateway: 10.0.44.1

      Change the IP configuration? [Yn] Y
      New IP address? [default: 10.0.44.16]
```

3. Type a valid IP address in the same LAN and press ENTER, then fill in the next fields in the same way.

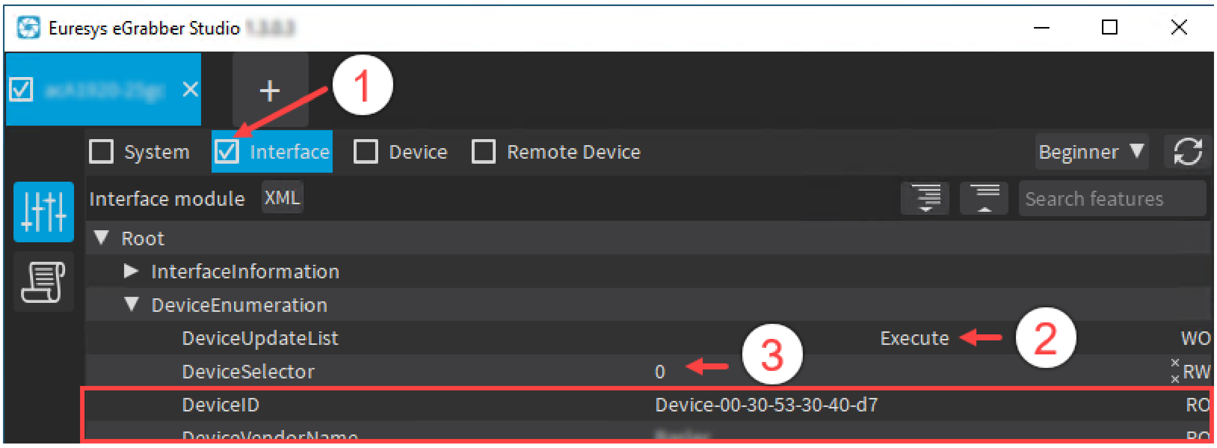
Once the camera IP settings are configured, open **eGrabber Studio** and connect to the GigE Vision camera to finalize the configuration of the network settings.

If **eGrabber Studio** was already running, make sure to refresh the list of discovered grabbers and to reconnect to the requested GigE Vision camera.

Configure the camera IP settings in eGrabber Studio

1. In the displayed Features pane, click the Interface tab to open it.
2. In the Interface module, in Root > DeviceEnumeration, execute the DeviceUpdateList command.
3. To browse through the cameras, type a number (0, 1, 2, and so on) in the DeviceSelector field and press ENTER.

   The parameter values of the selected camera are displayed below, starting with the DeviceID:



▼ Root
▶ InterfaceInformation
▼ DeviceEnumeration

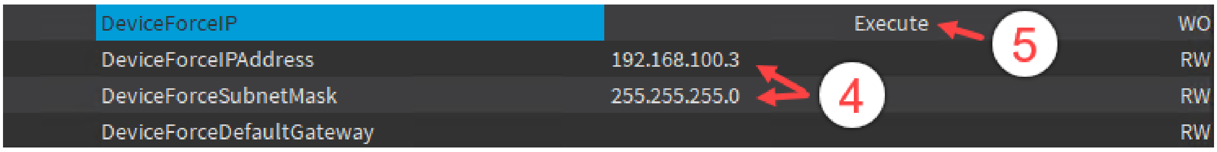
DeviceUpdateList	Execute	WO
DeviceSelector	0	RW
DeviceID	Device-00-30-53-30-40-d7	RO
DeviceVendorName		RO

4. Enter the desired values for the features DeviceForceIPAddress and DeviceForceSubnetMask, as well as for DeviceForceDefaultGateway, if your setup requires it.

   The subnet mask must be the same value as the NIC.

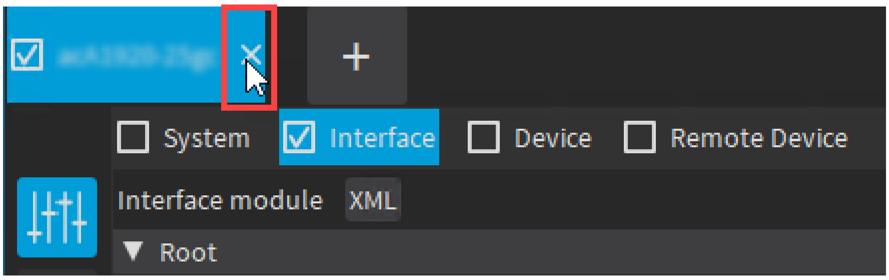
> NOTE
> Make sure the DeviceForceIPAddress is distinct from the NIC IP address configured in the section "Set the IP address of the network interface card" on page 17, but belongs to the same LAN (according to subnet mask). For example, if the NIC IP address is 192.168.100.2, the camera IP address can be 192.168.100.3.

5. When your have entered the desired values, click Execute on the right of DeviceForceIP.



DeviceForceIP	Execute	WO
DeviceForceIPAddress	192.168.100.3	RW
DeviceForceSubnetMask	255.255.255.0	RW
DeviceForceDefaultGateway		RW

6. Click the cross in the window tab to close the connection:



7. Connect to the stream of the configured GigE Vision camera as explained in section "Select the GigE Vision camera in eGrabber Studio" on page 24.

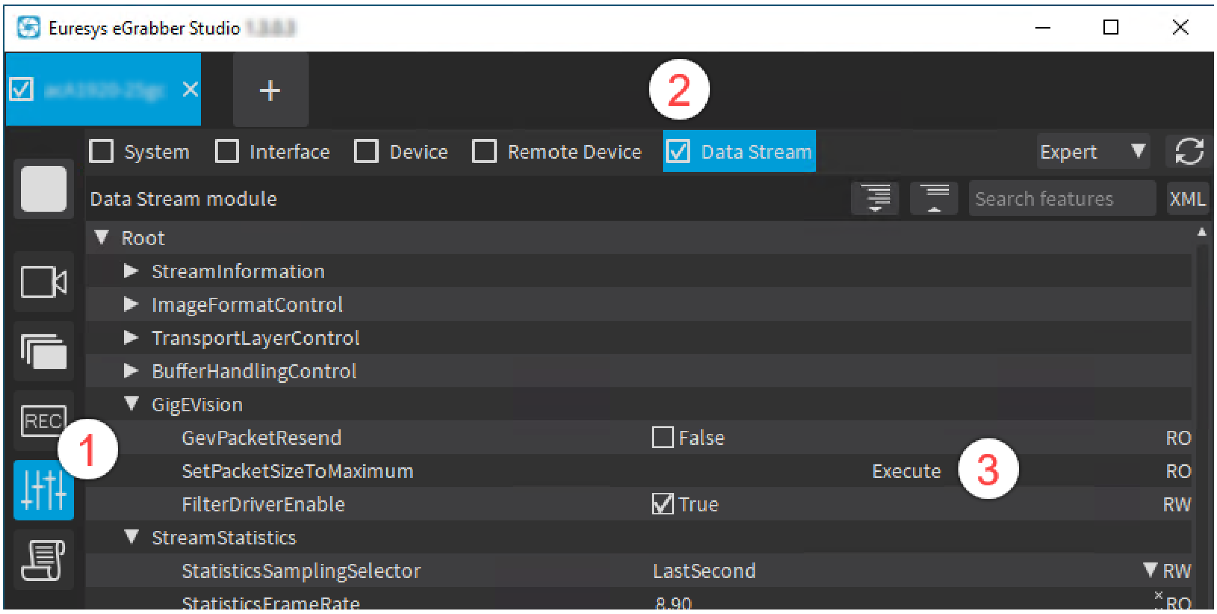
Set packet size to maximum

Gigelink can automatically configure camera packet size to use the maximum allowed in your setup. However, it only does so on-demand as the process implies trials-and-errors and takes 5-10 seconds.

> NOTE
> Make sure that you have enabled Jumbo Frames on your computer as described in the section "Enable the Jumbo Packet settings" on page 20.

When you are connected to your GigE Vision camera stream in **eGrabber Studio**, proceed as follows to set the packet size to maximum:

1. Click the Features icon on the left to open the Features pane.
2. Select the Data Stream module.
3. In the GigEVision section, click Execute on the SetPacketSizeToMaximum command.



Data Stream module

```
▼ Root
  ▶ StreamInformation
  ▶ ImageFormatControl
  ▶ TransportLayerControl
  ▶ BufferHandlingControl
  ▼ GigEVision
      GevPacketResend            □ False            RO
      SetPacketSizeToMaximum            Execute     RO
      FilterDriverEnable         ☑ True             RW
  ▼ StreamStatistics
      StatisticsSamplingSelector  LastSecond      ▼ RW
      StatisticsFrameRate         8.90              RO
```

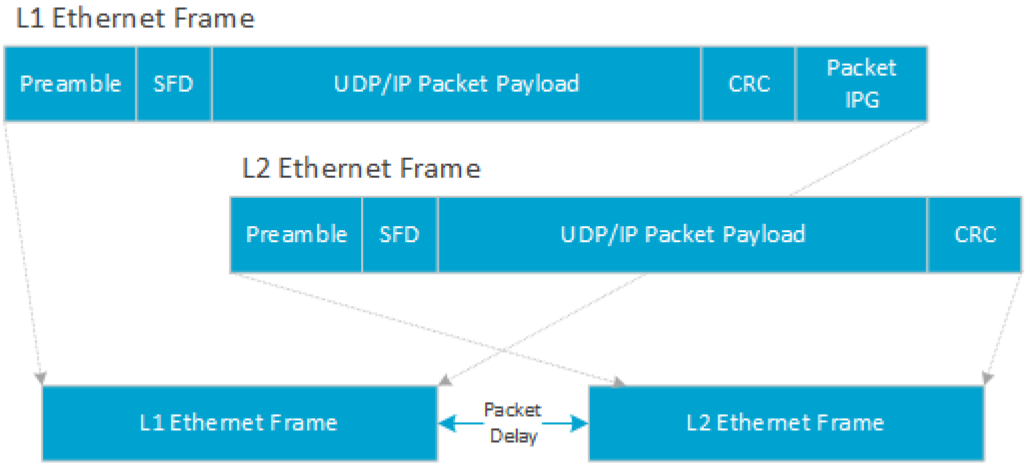
The value found should be reflected in GevSCPSPacketSize property of the remote device (Remote Device module > TransportLayer section).

> WARNING
> Packets used to determine the maximum size are particularly susceptible of being dropped by the firewall. When that occurs, SetPacketSizeToMaximum fails with OPAQUE_NETWORK error (code -10024 in Studio).

Set Packet Delay

This step is not necessary if **eGrabber Studio** correctly displays the images coming from the GigE Vision camera. However it helps fix issues where horizontal strips are visible through the image or the bottom of the image is missing.

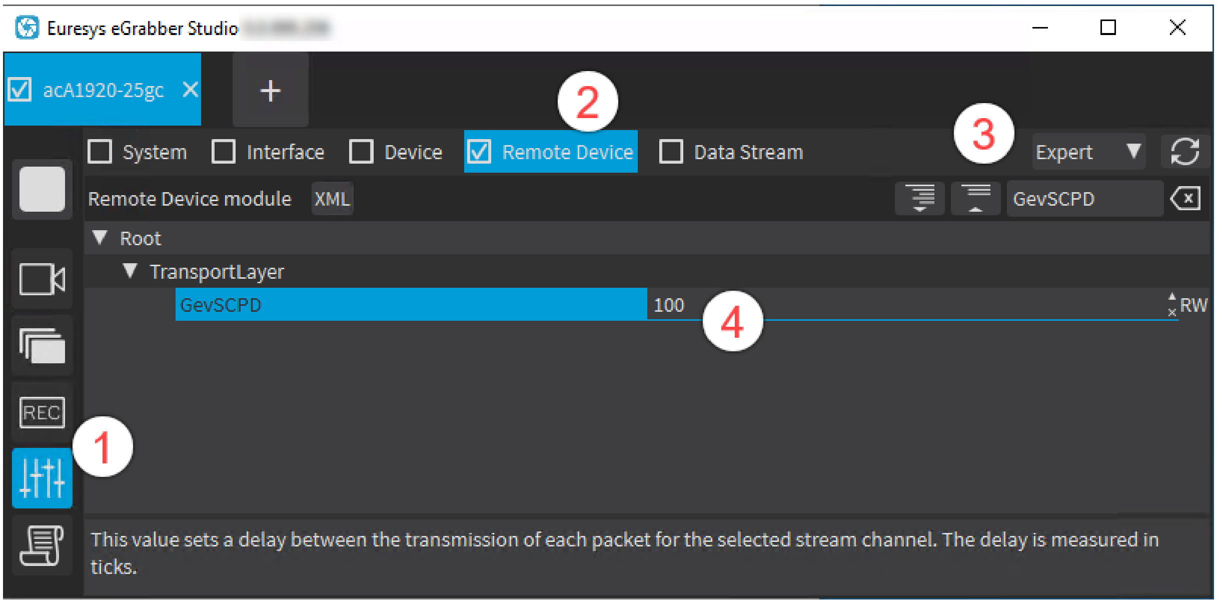
Packet Delay allows you to adjust spacing between packets, as shown at the bottom of the figure below. The default value is 0, but this may cause queue overflow and other problems due to PC performances. When this occurs, set the Packet Delay value to 100 or higher.



Ethernet frames and packet delay

To set packet delay, proceed as follows:

1. In **Studio**, click the Features icon on the left to open the Features pane.
2. Select the Remote Device module.
3. Search for the GevSCPD feature with at least the Expert user profile selected above the search field.
4. In the GevSCPD field, type 100 and press ENTER to validate.



This value sets a delay between the transmission of each packet for the selected stream channel. The delay is measured in ticks.

NOTE
As the Packet Delay value depends on the camera's time stamp counter, check the recommended values with the camera manufacturer.

3.4. Configure the Camera GenApi Features

Open a GigE Vision stream in eGrabber Studio

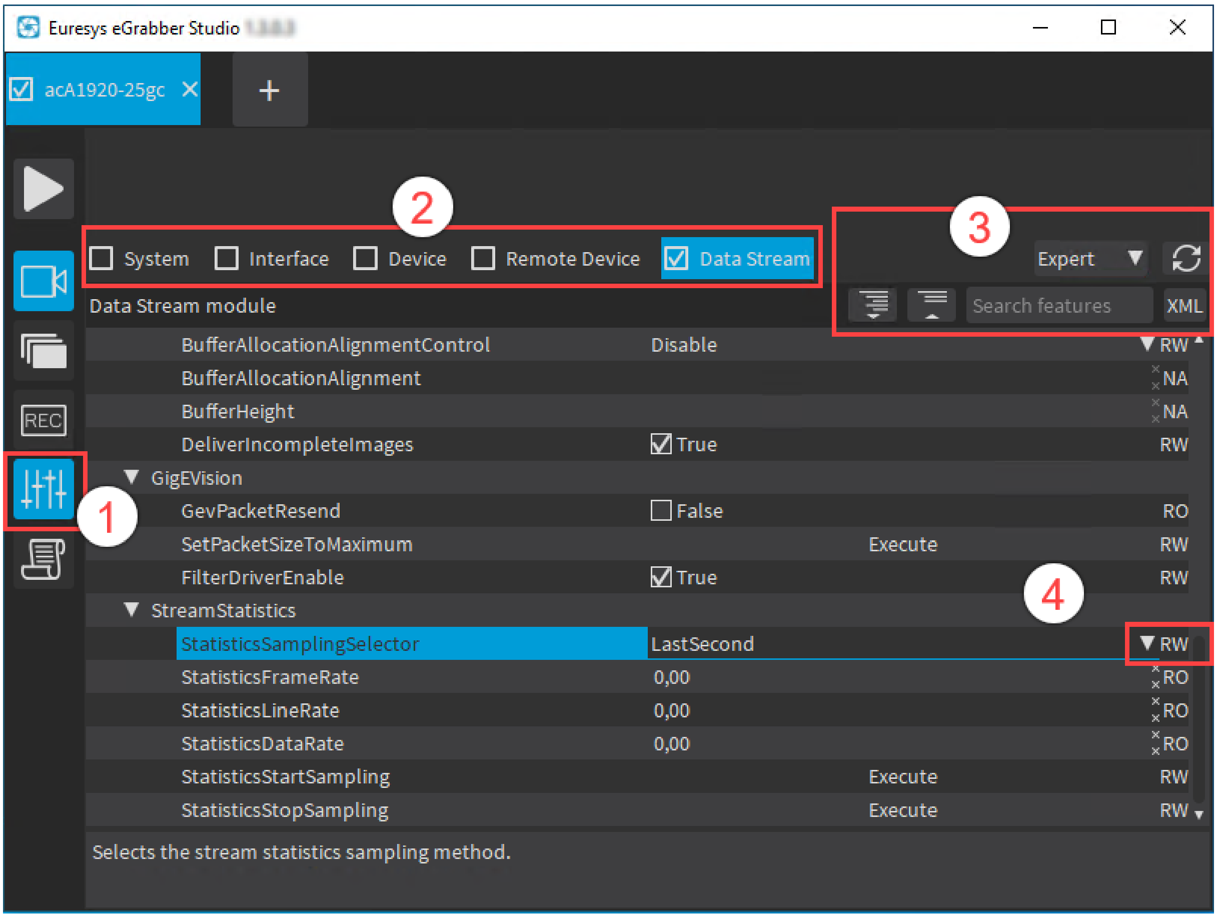
1. From the Windows Start menu, search for and open **Studio**.
2. In the Sources area, select **Gigelink**.
3. In the eGrabbers area, identify the desired **Gigelink** device and double-click its stream.

   The stream opens directly in the main window displayed as a tab.

Configure the GenApi features manually

The steps to configure a GenApi feature manually are highlighted on the screenshot below.

1. Once you have opened your GigE Vision stream in a window, click the Features button to open the Features pane.
2. In the Features pane, select the GenApi module you want to configure features in.
3. If requested, use the controls on the right to expand/collapse, filter and/or search for GenApi features.
4. When you have identified the feature you want to modify (it must be RW or WO to be editable), select or type the requested value and press ENTER on the keyboard.



Data Stream module

BufferAllocationAlignmentControl	Disable		RW
BufferAllocationAlignment			NA
BufferHeight			NA
DeliverIncompleteImages	☑ True		RW
▼ GigEVision			
GevPacketResend	☐ False		RO
SetPacketSizeToMaximum		Execute	RW
FilterDriverEnable	☑ True		RW
▼ StreamStatistics			
StatisticsSamplingSelector	LastSecond		RW
StatisticsFrameRate	0,00		RO
StatisticsLineRate	0,00		RO
StatisticsDataRate	0,00		RO
StatisticsStartSampling		Execute	RW
StatisticsStopSampling		Execute	RW

Selects the stream statistics sampling method.

Setting GenApi features

Configure the GenApi features with a script in eGrabber Studio

You can also define the requested values for several GenApi features in a script that you can apply in **eGrabber Studio**.

1. Prepare your script.

   You can refer to the following sources:

   □ sample configuration script for a GigE Vision camera connected with Gigelink

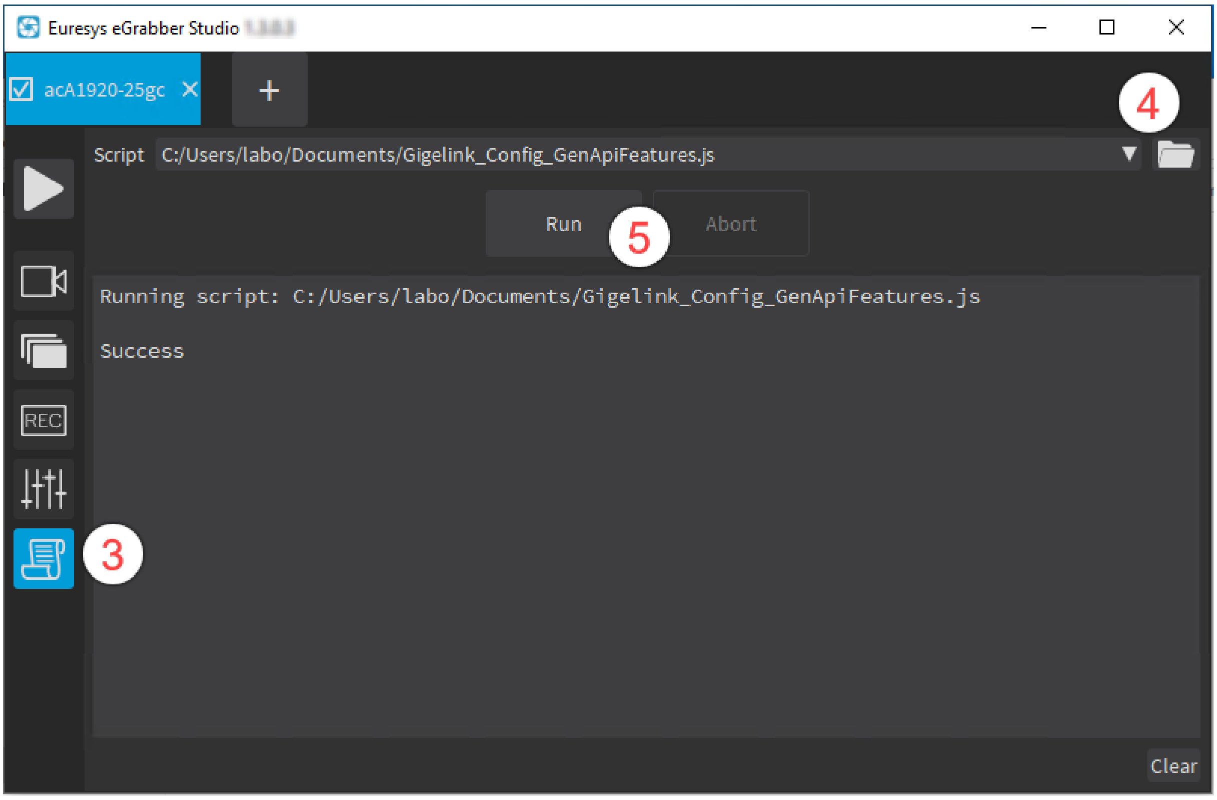
   □ detailed information on the syntax used in Euresys Script language.

2. Save your script on the computer where **eGrabber Studio** is installed.
3. In **eGrabber Studio**, once you have opened your GigE Vision stream in a window, click the Scripts button.

   See screenshot below.

4. Select your script using the Folder icon.
5. Click Run to execute the script.

The result of the script execution appears in the Script pane.


Euresys eGrabber Studio [redacted]
acA1920-25gc
+
4
Script C:/Users/labo/Documents/Gigelink_Config_GenApiFeatures.js
Run
5
Abort
REC
3
Clear

```
Running script: C:/Users/labo/Documents/Gigelink_Config_GenApiFeatures.js

Success
```

Sample configuration script for a GigE Vision camera

This section goes through a sample configuration for a GigE Vision camera, and provides more information on its instructions.

In the following sample script:

1. First, you configure the camera to work in the required mode to be able to use action commands.
2. Then, you configure the action command settings.
3. Finally, you configure the action 1 as a trigger for acquiring a frame.

```
// This script is used to configure the GenApi features so that a PC can send
// the appropriate Action Command to trigger a GigE Vision camera over the network.

var configure = require ('coaxlink://configurator.js');

var parameters = {
    OperatingMode: "SC",
    AcquisitionMode:"SingleFrame",
    ExposureTime:5500,
};
configure(grabbers[0], parameters);

grabbers[0].StreamPort.set("ControlRemoteDevice", "False");
grabbers[0].RemotePort.set("ActionDeviceKey", 0xcafebabe);
grabbers[0].DevicePort.set("ActionDeviceKey", 0xcafebabe);
grabbers[0].RemotePort.set("ActionSelector", 1);// must match 'Action1' below
grabbers[0].RemotePort.set("ActionGroupKey", 0x42);
grabbers[0].RemotePort.set("ActionGroupMask", 0xffffffff);
grabbers[0].DevicePort.set("ActionGroupKey", 0x42);
```

```
grabbers[0].DevicePort.set("ActionGroupMask", 0xffffffff);

// following parameters configure Action1 as frame trigger
grabbers[0].RemotePort.set("TriggerSelector", "FrameStart");
grabbers[0].RemotePort.set("TriggerMode", "On");
grabbers[0].RemotePort.set("TriggerSource", 'Action1');
```

Load configuration helper

The following line loads the configurator function into the variable configure.

```
var configure = require ('coaxlink://configurator.js');
```

Define generic configuration parameters

With the following instructions, you create the parameters variable and store parameter names and values in the variable.

```
var parameters = {
     OperatingMode: "SC",
     AcquisitionMode:"SingleFrame",
     ExposureTime:5500,
};
```

Apply generic configuration parameters

With the following instruction, you apply the configuration script (loaded in the configure variable) to the grabbers[0] to set the parameters stored in the parameters variable.

For more information on built-in objects grabbers, refer to the Euresys GenApi script section in the Programmer's Guide, sample script doc/grabbers.js.

```
configure(grabbers[0], parameters);
```

Configure the action command feature

1. In the first instruction below, you deactivate the automatic acquisition start and stop, as it can interfere while you are configuring the action command feature.
2. In the instructions that follow, you configure features used by the action command feature, i.e. DeviceKey, GroupKey and GroupMask, both on the device side and remote device side.

```
grabbers[0].StreamPort.set("ControlRemoteDevice", "False");
grabbers[0].RemotePort.set("ActionDeviceKey", 0xcafebabe);
grabbers[0].DevicePort.set("ActionDeviceKey", 0xcafebabe);
grabbers[0].RemotePort.set("ActionSelector", 1);// must match 'Action1' below
grabbers[0].RemotePort.set("ActionGroupKey", 0x42);
grabbers[0].RemotePort.set("ActionGroupMask", 0xffffffff);
grabbers[0].DevicePort.set("ActionGroupKey", 0x42);
grabbers[0].DevicePort.set("ActionGroupMask", 0xffffffff);
```

Define which trigger is activated when the action command is received

With the following instructions, you configure the FrameStart to be triggered via action command 1.

```
// following parameters configure Action1 as frame trigger
grabbers[0].RemotePort.set("TriggerSelector", "FrameStart");
grabbers[0].RemotePort.set("TriggerMode", "On");
grabbers[0].RemotePort.set("TriggerSource", 'Action1');
```

PART II
GENAPI FEATURES

1. Gigelink System Module Register Description

Categorized features list of Systemmodule version 24_03_2

1.1. Root Category

SystemInformation

Feature Info

Module	Category Path	Type	Access
System	Root	Category	RW

Category Members

See also: "SystemInformation Category" on page 39

InterfaceEnumeration

Feature Info

Module	Category Path	Type	Access
System	Root	Category	RW

Category Members

See also: "InterfaceEnumeration Category" on page 49

1.2. SystemInformation Category

TLVendorName

Feature Info

Module	Category Path	Type	Access
System	Root → SystemInformation	String	Imposed: RO

Short Description

Name of the GenTL Producer vendor.

TLModelName

Feature Info

Module	Category Path	Type	Access
System	Root → SystemInformation	String	Imposed: RO

Short Description

Name of the GenTL Producer.

TLID

Feature Info

Module	Category Path	Type	Access
System	Root → SystemInformation	String	Imposed: RO

Short Description

Unique identifier of the GenTL.

TLVersion

Feature Info

Module	Category Path	Type	Access
System	Root → SystemInformation	String	Imposed: RO

Short Description

Vendor specific version string.

TLPath

Feature Info

Module	Category Path	Type	Access
System	Root → SystemInformation	String	Imposed: RO

Short Description

Full path to the GenTL Producer driver including name and extension.

TLType

Feature Info

Module	Category Path	Type	Access
System	Root → SystemInformation	Enumeration	Imposed: RO

Short Description

Identifies the transport layer technology of the GenTL Producer implementation.

Enumeration Values

- GEV: This enumeration value indicates GigE Vision transport layer technology.

GenTLVersionMajor

Feature Info

Module	Category Path	Type	Access
System	Root → SystemInformation	IntReg	RO

Register Port: TLPort

Short Description

Major version number of the GenTL specification the GenTL Producer implementation complies with.

GenTLVersionMinor

Feature Info

Module	Category Path	Type	Access
System	Root → SystemInformation	IntReg	RO

Register Port: TLPort

Short Description

Minor version number of the GenTL specification the GenTL Producer implementation complies with.

LicenseStatus

Feature Info

Module	Category Path	Type	Access
System	Root → SystemInformation	Enumeration	Imposed: RO

Short Description

License Status.

Enumeration Values

- OK: License is OK.
- NotOK: License is not OK.
- Unknown: License status is unknown.

1.3. InterfaceEnumeration Category

InterfaceUpdateList

Feature Info

Module	Category Path	Type	Access
System	Root → InterfaceEnumeration	Command	Imposed: WO

Short Description

Updates the internal interface list.

InterfaceSelector

Feature Info

Module	Category Path	Type	Access
System	Root → InterfaceEnumeration	Integer	RW

Value Info

Minimum value: 0

Short Description

Selector for the different GenTL Producer interfaces.

Selected Features

- "InterfaceID" on page 52

InterfaceID

Feature Info

Module	Category Path	Type	Access
System	Root → InterfaceEnumeration	String	Imposed: RO

Short Description

GenTL Producer wide unique identifier of the selected interface.

2. Gigelink Interface Module Register Description

Categorized features list of Interfacemodule version 24_03_2

2.1. Root Category

InterfaceInformation

Feature Info

Module	Category Path	Type	Access
Interface	Root	Category	RW

Category Members

See also: "InterfaceInformation Category" on page 57

DeviceEnumeration

Feature Info

Module	Category Path	Type	Access
Interface	Root	Category	RW

Category Members

See also: "DeviceEnumeration Category" on page 63

2.2. InterfaceInformation Category

InterfaceID

Feature Info

Module	Category Path	Type	Access
Interface	Root → InterfaceInformation	String	Imposed: RO

Short Description

GenTL Producer wide unique identifier of the selected interface.

InterfaceType

Feature Info

Module	Category Path	Type	Access
Interface	Root → InterfaceInformation	Enumeration	Imposed: RO

Short Description

Identifies the transport layer technology of the interface.

Enumeration Values

- GEV: This enumeration value indicates GigE Vision transport layer technology.

IPAddress

Feature Info

Module	Category Path	Type	Access
Interface	Root → InterfaceInformation	String	Imposed: RO

Short Description

IP address of the network interface.

SubnetMask

Feature Info

Module	Category Path	Type	Access
Interface	Root → InterfaceInformation	String	Imposed: RO

Short Description

Subnet mask of the network interface.

AllowBroadcastAck

Feature Info

Module	Category Path	Type	Access
Interface	Root → InterfaceInformation	Boolean	RW

Short Description

Allow devices with mismatching network settings to be discovered through broadcast replies to DISCOVERY messages.

2.3. DeviceEnumeration Category

DeviceUpdateList

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	Command	Imposed: WO

Short Description

Updates the internal device list.

DeviceSelector

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	Integer	RW

Value Info

Minimum value: 0

Short Description

Selector for the different devices on this interface.

Selected Features

- "DeviceID" on page 66
- "DeviceVendorName" on page 67
- "DeviceModelName" on page 68
- "DeviceAccessStatus" on page 69
- "DeviceCurrentIPAddress" on page 70
- "DeviceCurrentSubnetMask" on page 71
- "DeviceCurrentDefaultGateway" on page 72
- "DeviceUserDefinedName" on page 73
- "DeviceSerialNumber" on page 74
- "DeviceVersion" on page 75
- "DeviceForceIP" on page 76

DeviceID

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	Imposed: RO

Short Description

Interface wide unique identifier of the selected device.

DeviceVendorName

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	Imposed: RO

Short Description

Name of the device vendor.

DeviceModelName

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	Imposed: RO

Short Description

Name of the device model.

DeviceAccessStatus

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	Enumeration	Imposed: RO

Short Description

Gives the device's access status at the moment of the last execution of DeviceUpdateList.

Enumeration Values

- Unknown: Unknown access.
- ReadWrite: Available to be opened with full access.
- ReadOnly: Available to be opened with read-only access.
- NoAccess: Not reachable.
- Busy: Already opened by another entity.
- OpenReadWrite: Opened with read-write access.
- OpenReadOnly: Opened with read-only access.

DeviceCurrentIPAddress

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	Imposed: RO

Short Description

Current IP address of the selected device.

DeviceCurrentSubnetMask

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	Imposed: RO

Short Description

Current subnet mask of the selected device.

DeviceCurrentDefaultGateway

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	Imposed: RO

Short Description

Current default gateway of the selected device.

DeviceUserDefinedName

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	Imposed: RO

Short Description

User-defined name of the selected device.

DeviceSerialNumber

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	Imposed: RO

Short Description

Serial Number of the selected device.

DeviceVersion

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	Imposed: RO

Short Description

Version of the selected device.

DeviceForceIP

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	Command	Imposed: WO

Short Description

Force use of DeviceForceIPAddress, DeviceForceSubnetMask, and DeviceForceDefaultGateway for the selected device.

DeviceForceIPAddress

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	RW

Short Description

IP address used in DeviceForceIP command.

DeviceForceSubnetMask

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	RW

Short Description

Subnet mask used in DeviceForceIP command.

DeviceForceDefaultGateway

Feature Info

Module	Category Path	Type	Access
Interface	Root → DeviceEnumeration	String	RW

Short Description

Default gateway used in DeviceForceIP command.

3. Gigelink Device Module Register Description

Categorized features list of Devicemodule version 24_03_2

3.1. Root Category

DeviceInformation

Feature Info

Module	Category Path	Type	Access
Device	Root	Category	RW

Category Members

See also: "DeviceInformation Category" on page 85

StreamEnumeration

Feature Info

Module	Category Path	Type	Access
Device	Root	Category	RW

Category Members

See also: "StreamEnumeration Category" on page 92

GigEVision

Feature Info

Module	Category Path	Type	Access
Device	Root	Category	RW

Category Members

See also: "GigEVision Category" on page 95

3.2. DeviceInformation Category

DeviceID

Feature Info

Module	Category Path	Type	Access
Device	Root → DeviceInformation	String	Imposed: RO

Short Description

Interface wide unique identifier of this device.

DeviceVendorName

Feature Info

Module	Category Path	Type	Access
Device	Root → DeviceInformation	String	Imposed: RO

Short Description

Name of the device vendor.

DeviceModelName

Feature Info

Module	Category Path	Type	Access
Device	Root → DeviceInformation	String	Imposed: RO

Short Description

Name of the device model.

DeviceAccessStatus

Feature Info

Module	Category Path	Type	Access
Device	Root → DeviceInformation	Enumeration	Imposed: RO

Short Description

Gives the device's access status at the moment of the last execution of DeviceUpdateList.

Enumeration Values

- Unknown: Unknown access.
- ReadWrite: Available to be opened with full access.
- ReadOnly: Available to be opened with read-only access.
- NoAccess: Not reachable.
- Busy: Already opened by another entity.
- OpenReadWrite: Opened with read-write access.
- OpenReadOnly: Opened with read-only access.

DeviceType

Feature Info

Module	Category Path	Type	Access
Device	Root → DeviceInformation	Enumeration	Imposed: RO

Short Description

Identifies the transport layer technology of the interface.

Enumeration Values

- GEV: This enumeration value indicates GigE Vision transport layer technology.

DeviceDescription

Feature Info

Module	Category Path	Type	Access
Device	Root → DeviceInformation	String	Imposed: RO

Short Description

Description of the device.

3.3. StreamEnumeration Category

StreamSelector

Feature Info

Module	Category Path	Type	Access
Device	Root → StreamEnumeration	Integer	RW

Value Info

Minimum value: 0

Short Description

Selector for the different stream channels.

Selected Features

- "StreamID" on page 94

StreamID

Feature Info

Module	Category Path	Type	Access
Device	Root → StreamEnumeration	String	Imposed: RO

Short Description

Device unique ID for the stream.

3.4. GigEVision Category

DeviceLinkHeartbeatTimeout

Feature Info

Module	Category Path	Type	Access
Device	Root → GigEVision	FloatReg	RW

Register Port: DevicePort

Value Info

Unit: microseconds

Short Description

Writing a new heartbeat timeout will affect both device GevHeartbeatTimeout and keep-alive delays on the host

ActionDeviceKey

Feature Info

Module	Category Path	Type	Access
Device	Root → GigEVision	IntReg	WO

Register Port: DevicePort

Short Description

Action Device Key.

ActionGroupKey

Feature Info

Module	Category Path	Type	Access
Device	Root → GigEVision	IntReg	RW

Register Port: DevicePort

Short Description

Action Group Key.

ActionGroupMask

Feature Info

Module	Category Path	Type	Access
Device	Root → GigEVision	IntReg	RW

Register Port: DevicePort

Short Description

Action Group Mask.

ExecuteAction

Feature Info

Module	Category Path	Type	Access
Device	Root → GigEVision	Command	Imposed: WO

Short Description

Execute Action.

4. Gigelink Data Stream Module Register Description

Categorized features list of Data Streammodule version 24_03_2

4.1. Root Category

StreamInformation

Feature Info

Module	Category Path	Type	Access
Data Stream	Root	Category	RW

Category Members

See also: "StreamInformation Category" on page 111

ImageFormatControl

Feature Info

Module	Category Path	Type	Access
Data Stream	Root	Category	RW

Category Members

See also: "ImageFormatControl Category" on page 116

TransportLayerControl

Feature Info

Module	Category Path	Type	Access
Data Stream	Root	Category	RW

Category Members

See also: "TransportLayerControl Category" on page 145

BufferHandlingControl

Feature Info

Module	Category Path	Type	Access
Data Stream	Root	Category	RW

Category Members

See also: "BufferHandlingControl Category" on page 148

StreamControl

Feature Info

Module	Category Path	Type	Access
Data Stream	Root	Category	RW

Category Members

See also: "StreamControl Category" on page 174

GigEVision

Feature Info

Module	Category Path	Type	Access
Data Stream	Root	Category	RW

Category Members

See also: "GigEVision Category" on page 170

EventControl

Feature Info

Module	Category Path	Type	Access
Data Stream	Root	Category	RW

Category Members

See also: "EventControl Category" on page 178

StreamStatistics

Feature Info

Module	Category Path	Type	Access
Data Stream	Root	Category	RW

Category Members

See also: "StreamStatistics Category" on page 186

4.2. StreamInformation Category

StreamID

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamInformation	String	Imposed: RO

Short Description

Device unique ID for the data stream.

StreamType

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamInformation	Enumeration	Imposed: RO

Short Description

Identifies the transport layer technology of the interface.

Enumeration Values

- GEV: This enumeration value indicates GigE Vision transport layer technology.

StreamDescription

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamInformation	String	Imposed: RO

Short Description

Description of the stream.

StreamConfigurationStatus

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamInformation	Enumeration	Imposed: RO

Short Description

Stream Configuration Status.

Enumeration Values

- OK:
- UnknownError:
- BayerDecoderAndUnpackingModeMsbNotAllowed:
- BayerDecoderAndUnpackingModeOffNotAllowed:
- BinningAndFormatNotAllowed:
- BinningAndUnpackingModeNotAllowed:
- BinningConfigAndFormatDepthNotAllowed:
- LleNotImplemented:
- LleAndFormatNotAllowed:
- LutAndBayerFormatNotAllowed:
- LutAndLleNotAllowed:
- LutAndMultiComponentFormatNotAllowed:
- LutAndPackedFormatNotAllowed:
- LutAndUnpackingModeOffNotAllowed:
- LutConfigAndFormatDepthDontMatch:
- WidthNotAligned:
- CameraWidthNotAlignedToBinningWindow:
- CameraHeightNotAlignedToBinningWindow:
- UnpackingModeOffAnd16bitFormatNotAllowed:
- UnpackingModeOffAndFormatNotAllowed:
- ReverseXAndFormatNotAllowed:
- ReverseXAndUnpackingModeNotAllowed:
- CameraWidthTooLargeForReverseX:

4.3. ImageFormatControl Category

PixelFormat

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → ImageFormatControl	Enumeration	Imposed: RO

Short Description

Pixel format of the image.

Enumeration Values

- BayerBG10pmsb: BayerBG10pmsb.
- BayerBG12pmsb: BayerBG12pmsb.
- BayerBG14pmsb: BayerBG14pmsb.
- BayerGB10pmsb: BayerGB10pmsb.
- BayerGB12pmsb: BayerGB12pmsb.
- BayerGB14pmsb: BayerGB14pmsb.
- BayerGR10pmsb: BayerGR10pmsb.
- BayerGR12pmsb: BayerGR12pmsb.
- BayerGR14pmsb: BayerGR14pmsb.
- BayerRG10pmsb: BayerRG10pmsb.
- BayerRG12pmsb: BayerRG12pmsb.
- BayerRG14pmsb: BayerRG14pmsb.
- Mono10pmsb: Mono10pmsb.
- Mono12pmsb: Mono12pmsb.
- Mono14pmsb: Mono14pmsb.
- RGB10pmsb: RGB10pmsb.
- RGB12pmsb: RGB12pmsb.
- RGB14pmsb: RGB14pmsb.
- RGBa10pmsb: RGBa10pmsb.
- RGBa12pmsb: RGBa12pmsb.
- RGBa14pmsb: RGBa14pmsb.
- YCbCr601_10pmsb: YCbCr601_10pmsb.
- YCbCr601_12pmsb: YCbCr601_12pmsb.
- YCbCr601_14pmsb: YCbCr601_14pmsb.

- YCbCr601_16: YCbCr601_16.
- YCbCr601_411_10pmsb: YCbCr601_411_10pmsb.
- YCbCr601_411_12pmsb: YCbCr601_411_12pmsb.
- YCbCr601_411_14pmsb: YCbCr601_411_14pmsb.
- YCbCr601_411_16: YCbCr601_411_16.
- YCbCr601_411_8: YCbCr601_411_8.
- YCbCr601_422_10pmsb: YCbCr601_422_10pmsb.
- YCbCr601_422_12pmsb: YCbCr601_422_12pmsb.
- YCbCr601_422_14pmsb: YCbCr601_422_14pmsb.
- YCbCr601_422_16: YCbCr601_422_16.
- YCbCr601_8: YCbCr601_8.
- YCbCr709_10pmsb: YCbCr709_10pmsb.
- YCbCr709_12pmsb: YCbCr709_12pmsb.
- YCbCr709_14pmsb: YCbCr709_14pmsb.
- YCbCr709_16: YCbCr709_16.
- YCbCr709_411_10pmsb: YCbCr709_411_10pmsb.
- YCbCr709_411_12pmsb: YCbCr709_411_12pmsb.
- YCbCr709_411_14pmsb: YCbCr709_411_14pmsb.
- YCbCr709_411_16: YCbCr709_411_16.
- YCbCr709_411_8: YCbCr709_411_8.
- YCbCr709_422_10pmsb: YCbCr709_422_10pmsb.
- YCbCr709_422_12pmsb: YCbCr709_422_12pmsb.
- YCbCr709_422_14pmsb: YCbCr709_422_14pmsb.
- YCbCr709_422_16: YCbCr709_422_16.
- YCbCr709_8: YCbCr709_8.
- YUV10pmsb: YUV10pmsb.
- YUV12pmsb: YUV12pmsb.
- YUV14pmsb: YUV14pmsb.
- YUV16: YUV16.
- YUV411_10pmsb: YUV411_10pmsb.
- YUV411_12pmsb: YUV411_12pmsb.
- YUV411_14pmsb: YUV411_14pmsb.
- YUV411_16: YUV411_16.
- YUV411_8: YUV411_8.
- YUV422_10pmsb: YUV422_10pmsb.
- YUV422_12pmsb: YUV422_12pmsb.

- YUV422_14pmsb: YUV422_14pmsb.
- YUV422_16: YUV422_16.
- YUV8: YUV8.
- B10: Blue 10-bit.
- B12: Blue 12-bit.
- B16: Blue 16-bit.
- B8: Blue 8-bit.
- BayerBG10: Bayer Blue-Green 10-bit unpacked.
- BayerBG10p: Bayer Blue-Green 10-bit packed.
- BayerBG10Packed: Bayer Blue-Green 10-bit packed.
- BayerBG12: Bayer Blue-Green 12-bit unpacked.
- BayerBG12p: Bayer Blue-Green 12-bit packed.
- BayerBG12Packed: Bayer Blue-Green 12-bit packed.
- BayerBG14: Bayer Blue-Green 14-bit.
- BayerBG14p: Bayer Blue-Green 14-bit packed.
- BayerBG16: Bayer Blue-Green 16-bit.
- BayerBG4p: Bayer Blue-Green 4-bit packed.
- BayerBG8: Bayer Blue-Green 8-bit.
- BayerGB10: Bayer Green-Blue 10-bit unpacked.
- BayerGB10p: Bayer Green-Blue 10-bit packed.
- BayerGB10Packed: Bayer Green-Blue 10-bit packed.
- BayerGB12: Bayer Green-Blue 12-bit unpacked.
- BayerGB12p: Bayer Green-Blue 12-bit packed.
- BayerGB12Packed: Bayer Green-Blue 12-bit packed.
- BayerGB14: Bayer Green-Blue 14-bit.
- BayerGB14p: Bayer Green-Blue 14-bit packed.
- BayerGB16: Bayer Green-Blue 16-bit.
- BayerGB4p: Bayer Green-Blue 4-bit packed.
- BayerGB8: Bayer Green-Blue 8-bit.
- BayerGR10: Bayer Green-Red 10-bit unpacked.
- BayerGR10p: Bayer Green-Red 10-bit packed.
- BayerGR10Packed: Bayer Green-Red 10-bit packed.
- BayerGR12: Bayer Green-Red 12-bit unpacked.
- BayerGR12p: Bayer Green-Red 12-bit packed.
- BayerGR12Packed: Bayer Green-Red 12-bit packed.
- BayerGR14: Bayer Green-Red 14-bit.

- BayerGR14p: Bayer Green-Red 14-bit packed.
- BayerGR16: Bayer Green-Red 16-bit.
- BayerGR4p: Bayer Green-Red 4-bit packed.
- BayerGR8: Bayer Green-Red 8-bit.
- BayerRG10: Bayer Red-Green 10-bit unpacked.
- BayerRG10p: Bayer Red-Green 10-bit packed.
- BayerRG10Packed: Bayer Red-Green 10-bit packed.
- BayerRG12: Bayer Red-Green 12-bit unpacked.
- BayerRG12p: Bayer Red-Green 12-bit packed.
- BayerRG12Packed: Bayer Red-Green 12-bit packed.
- BayerRG14: Bayer Red-Green 14-bit.
- BayerRG14p: Bayer Red-Green 14-bit packed.
- BayerRG16: Bayer Red-Green 16-bit.
- BayerRG4p: Bayer Red-Green 4-bit packed.
- BayerRG8: Bayer Red-Green 8-bit.
- BGR10: Blue-Green-Red 10-bit unpacked.
- BGR10p: Blue-Green-Red 10-bit packed.
- BGR12: Blue-Green-Red 12-bit unpacked.
- BGR12p: Blue-Green-Red 12-bit packed.
- BGR14: Blue-Green-Red 14-bit unpacked.
- BGR16: Blue-Green-Red 16-bit.
- BGR565p: Blue-Green-Red 5/6/5-bit packed.
- BGR8: Blue-Green-Red 8-bit.
- BGR8a32: BGR8a32.
- BGRa10: Blue-Green-Red-alpha 10-bit unpacked.
- BGRa10p: Blue-Green-Red-alpha 10-bit packed.
- BGRa12: Blue-Green-Red-alpha 12-bit unpacked.
- BGRa12p: Blue-Green-Red-alpha 12-bit packed.
- BGRa14: Blue-Green-Red-alpha 14-bit unpacked.
- BGRa16: Blue-Green-Red-alpha 16-bit.
- BGRa8: Blue-Green-Red-alpha 8-bit.
- BiColorBGRG10: Bi-color Blue/Green - Red/Green 10-bit unpacked.
- BiColorBGRG10p: Bi-color Blue/Green - Red/Green 10-bit packed.
- BiColorBGRG12: Bi-color Blue/Green - Red/Green 12-bit unpacked.
- BiColorBGRG12p: Bi-color Blue/Green - Red/Green 12-bit packed.
- BiColorBGRG8: Bi-color Blue/Green - Red/Green 8-bit.

- BiColorRGBG10: Bi-color Red/Green - Blue/Green 10-bit unpacked.
- BiColorRGBG10p: Bi-color Red/Green - Blue/Green 10-bit packed.
- BiColorRGBG12: Bi-color Red/Green - Blue/Green 12-bit unpacked.
- BiColorRGBG12p: Bi-color Red/Green - Blue/Green 12-bit packed.
- BiColorRGBG8: Bi-color Red/Green - Blue/Green 8-bit.
- Confidence1: Confidence 1-bit unpacked.
- Confidence16: Confidence 16-bit.
- Confidence1p: Confidence 1-bit packed.
- Confidence32f: Confidence 32-bit floating point.
- Confidence8: Confidence 8-bit.
- Coord3D_A10p: 3D coordinate A 10-bit packed.
- Coord3D_A12p: 3D coordinate A 12-bit packed.
- Coord3D_A16: 3D coordinate A 16-bit.
- Coord3D_A32f: 3D coordinate A 32-bit floating point.
- Coord3D_A8: 3D coordinate A 8-bit.
- Coord3D_ABC10p: 3D coordinate A-B-C 10-bit packed.
- Coord3D_ABC10p_Planar: 3D coordinate A-B-C 10-bit packed planar.
- Coord3D_ABC12p: 3D coordinate A-B-C 12-bit packed.
- Coord3D_ABC12p_Planar: 3D coordinate A-B-C 12-bit packed planar.
- Coord3D_ABC16: 3D coordinate A-B-C 16-bit.
- Coord3D_ABC16_Planar: 3D coordinate A-B-C 16-bit planar.
- Coord3D_ABC32f: 3D coordinate A-B-C 32-bit floating point.
- Coord3D_ABC32f_Planar: 3D coordinate A-B-C 32-bit floating point planar.
- Coord3D_ABC8: 3D coordinate A-B-C 8-bit.
- Coord3D_ABC8_Planar: 3D coordinate A-B-C 8-bit planar.
- Coord3D_AC10p: 3D coordinate A-C 10-bit packed.
- Coord3D_AC10p_Planar: 3D coordinate A-C 10-bit packed planar.
- Coord3D_AC12p: 3D coordinate A-C 12-bit packed.
- Coord3D_AC12p_Planar: 3D coordinate A-C 12-bit packed planar.
- Coord3D_AC16: 3D coordinate A-C 16-bit.
- Coord3D_AC16_Planar: 3D coordinate A-C 16-bit planar.
- Coord3D_AC32f: 3D coordinate A-C 32-bit floating point.
- Coord3D_AC32f_Planar: 3D coordinate A-C 32-bit floating point planar.
- Coord3D_AC8: 3D coordinate A-C 8-bit.
- Coord3D_AC8_Planar: 3D coordinate A-C 8-bit planar.
- Coord3D_B10p: 3D coordinate B 10-bit packed.

- Coord3D_B12p: 3D coordinate B 12-bit packed.
- Coord3D_B16: 3D coordinate B 16-bit.
- Coord3D_B32f: 3D coordinate B 32-bit floating point.
- Coord3D_B8: 3D coordinate B 8-bit.
- Coord3D_C10p: 3D coordinate C 10-bit packed.
- Coord3D_C12p: 3D coordinate C 12-bit packed.
- Coord3D_C16: 3D coordinate C 16-bit.
- Coord3D_C32f: 3D coordinate C 32-bit floating point.
- Coord3D_C8: 3D coordinate C 8-bit.
- CustomBayerBG14: CustomBayerBG14.
- CustomBayerGB14: CustomBayerGB14.
- CustomBayerGR14: CustomBayerGR14.
- CustomBayerRG14: CustomBayerRG14.
- CustomJFIF: CustomJFIF.
- G10: Green 10-bit.
- G12: Green 12-bit.
- G16: Green 16-bit.
- G8: Green 8-bit.
- Mono10: Monochrome 10-bit unpacked.
- Mono10p: Monochrome 10-bit packed.
- Mono10Packed: Monochrome 10-bit packed.
- Mono12: Monochrome 12-bit unpacked.
- Mono12p: Monochrome 12-bit packed.
- Mono12Packed: Monochrome 12-bit packed.
- Mono14: Monochrome 14-bit unpacked.
- Mono14p: Monochrome 14-bit packed.
- Mono16: Monochrome 16-bit.
- Mono1p: Monochrome 1-bit packed.
- Mono2p: Monochrome 2-bit packed.
- Mono32: Monochrome 32-bit.
- Mono4p: Monochrome 4-bit packed.
- Mono8: Monochrome 8-bit.
- Mono8s: Monochrome 8-bit signed.
- R10: Red 10-bit.
- R12: Red 12-bit.
- R16: Red 16-bit.

- R8: Red 8-bit.
- RGB10: Red-Green-Blue 10-bit unpacked.
- RGB10_Planar: Red-Green-Blue 10-bit unpacked planar.
- RGB10p: Red-Green-Blue 10-bit packed.
- RGB10p32: Red-Green-Blue 10-bit packed into 32-bit.
- RGB10V1Packed: Red-Green-Blue 10-bit packed - variant 1.
- RGB12: Red-Green-Blue 12-bit unpacked.
- RGB12_Planar: Red-Green-Blue 12-bit unpacked planar.
- RGB12p: Red-Green-Blue 12-bit packed.
- RGB12V1Packed: Red-Green-Blue 12-bit packed - variant 1.
- RGB14: Red-Green-Blue 14-bit unpacked.
- RGB16: Red-Green-Blue 16-bit.
- RGB16_Planar: Red-Green-Blue 16-bit planar.
- RGB565p: Red-Green-Blue 5/6/5-bit packed.
- RGB8: Red-Green-Blue 8-bit.
- RGB8_Planar: Red-Green-Blue 8-bit planar.
- RGB8a32: RGB8a32.
- RGBa10: Red-Green-Blue-alpha 10-bit unpacked.
- RGBa10p: Red-Green-Blue-alpha 10-bit packed.
- RGBa12: Red-Green-Blue-alpha 12-bit unpacked.
- RGBa12p: Red-Green-Blue-alpha 12-bit packed.
- RGBa14: Red-Green-Blue-alpha 14-bit unpacked.
- RGBa16: Red-Green-Blue-alpha 16-bit.
- RGBa8: Red-Green-Blue-alpha 8-bit.
- SCF1WBWG10: Sparse Color Filter #1 White-Blue-White-Green 10-bit unpacked.
- SCF1WBWG10p: Sparse Color Filter #1 White-Blue-White-Green 10-bit packed.
- SCF1WBWG12: Sparse Color Filter #1 White-Blue-White-Green 12-bit unpacked.
- SCF1WBWG12p: Sparse Color Filter #1 White-Blue-White-Green 12-bit packed.
- SCF1WBWG14: Sparse Color Filter #1 White-Blue-White-Green 14-bit unpacked.
- SCF1WBWG16: Sparse Color Filter #1 White-Blue-White-Green 16-bit unpacked.
- SCF1WBWG8: Sparse Color Filter #1 White-Blue-White-Green 8-bit.
- SCF1WGWB10: Sparse Color Filter #1 White-Green-White-Blue 10-bit unpacked.
- SCF1WGWB10p: Sparse Color Filter #1 White-Green-White-Blue 10-bit packed.
- SCF1WGWB12: Sparse Color Filter #1 White-Green-White-Blue 12-bit unpacked.
- SCF1WGWB12p: Sparse Color Filter #1 White-Green-White-Blue 12-bit packed.
- SCF1WGWB14: Sparse Color Filter #1 White-Green-White-Blue 14-bit unpacked.

- SCF1WGWB16: Sparse Color Filter #1 White-Green-White-Blue 16-bit.
- SCF1WGWB8: Sparse Color Filter #1 White-Green-White-Blue 8-bit.
- SCF1WGWR10: Sparse Color Filter #1 White-Green-White-Red 10-bit unpacked.
- SCF1WGWR10p: Sparse Color Filter #1 White-Green-White-Red 10-bit packed.
- SCF1WGWR12: Sparse Color Filter #1 White-Green-White-Red 12-bit unpacked.
- SCF1WGWR12p: Sparse Color Filter #1 White-Green-White-Red 12-bit packed.
- SCF1WGWR14: Sparse Color Filter #1 White-Green-White-Red 14-bit unpacked.
- SCF1WGWR16: Sparse Color Filter #1 White-Green-White-Red 16-bit.
- SCF1WGWR8: Sparse Color Filter #1 White-Green-White-Red 8-bit.
- SCF1WRWG10: Sparse Color Filter #1 White-Red-White-Green 10-bit unpacked.
- SCF1WRWG10p: Sparse Color Filter #1 White-Red-White-Green 10-bit packed.
- SCF1WRWG12: Sparse Color Filter #1 White-Red-White-Green 12-bit unpacked.
- SCF1WRWG12p: Sparse Color Filter #1 White-Red-White-Green 12-bit packed.
- SCF1WRWG14: Sparse Color Filter #1 White-Red-White-Green 14-bit unpacked.
- SCF1WRWG16: Sparse Color Filter #1 White-Red-White-Green 16-bit.
- SCF1WRWG8: Sparse Color Filter #1 White-Red-White-Green 8-bit.
- YCbCr10_CbYCr: YCbCr 4:4:4 10-bit unpacked.
- YCbCr10p_CbYCr: YCbCr 4:4:4 10-bit packed.
- YCbCr12_CbYCr: YCbCr 4:4:4 12-bit unpacked.
- YCbCr12p_CbYCr: YCbCr 4:4:4 12-bit packed.
- YCbCr2020_10_CbYCr: YCbCr 4:4:4 10-bit unpacked BT.2020.
- YCbCr2020_10p_CbYCr: YCbCr 4:4:4 10-bit packed BT.2020.
- YCbCr2020_12_CbYCr: YCbCr 4:4:4 12-bit unpacked BT.2020.
- YCbCr2020_12p_CbYCr: YCbCr 4:4:4 12-bit packed BT.2020.
- YCbCr2020_411_8_CbYYCrYY: YCbCr 4:1:1 8-bit BT.2020.
- YCbCr2020_422_10: YCbCr 4:2:2 10-bit unpacked BT.2020.
- YCbCr2020_422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked BT.2020.
- YCbCr2020_422_10p: YCbCr 4:2:2 10-bit packed BT.2020.
- YCbCr2020_422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed BT.2020.
- YCbCr2020_422_12: YCbCr 4:2:2 12-bit unpacked BT.2020.
- YCbCr2020_422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked BT.2020.
- YCbCr2020_422_12p: YCbCr 4:2:2 12-bit packed BT.2020.
- YCbCr2020_422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed BT.2020.
- YCbCr2020_422_8: YCbCr 4:2:2 8-bit BT.2020.
- YCbCr2020_422_8_CbYCrY: YCbCr 4:2:2 8-bit BT.2020.
- YCbCr2020_8_CbYCr: YCbCr 4:4:4 8-bit BT.2020.

- YCbCr411_8: YCbCr 4:1:1 8-bit.
- YCbCr411_8_CbYYCrYY: YCbCr 4:1:1 8-bit.
- YCbCr420_8_YY_CbCr_Semiplanar: YCbCr 4:2:0 8-bit YY/CbCr Semiplanar.
- YCbCr420_8_YY_CrCb_Semiplanar: YCbCr 4:2:0 8-bit YY/CrCb Semiplanar.
- YCbCr422_10: YCbCr 4:2:2 10-bit unpacked.
- YCbCr422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked.
- YCbCr422_10p: YCbCr 4:2:2 10-bit packed.
- YCbCr422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed.
- YCbCr422_12: YCbCr 4:2:2 12-bit unpacked.
- YCbCr422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked.
- YCbCr422_12p: YCbCr 4:2:2 12-bit packed.
- YCbCr422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed.
- YCbCr422_8: YCbCr 4:2:2 8-bit.
- YCbCr422_8_CbYCrY: YCbCr 4:2:2 8-bit.
- YCbCr422_8_YY_CbCr_Semiplanar: YCbCr 4:2:2 8-bit YY/CbCr Semiplanar.
- YCbCr422_8_YY_CrCb_Semiplanar: YCbCr 4:2:2 8-bit YY/CrCb Semiplanar.
- YCbCr601_10_CbYCr: YCbCr 4:4:4 10-bit unpacked BT.601.
- YCbCr601_10p_CbYCr: YCbCr 4:4:4 10-bit packed BT.601.
- YCbCr601_12_CbYCr: YCbCr 4:4:4 12-bit unpacked BT.601.
- YCbCr601_12p_CbYCr: YCbCr 4:4:4 12-bit packed BT.601.
- YCbCr601_411_8_CbYYCrYY: YCbCr 4:1:1 8-bit BT.601.
- YCbCr601_422_10: YCbCr 4:2:2 10-bit unpacked BT.601.
- YCbCr601_422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked BT.601.
- YCbCr601_422_10p: YCbCr 4:2:2 10-bit packed BT.601.
- YCbCr601_422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed BT.601.
- YCbCr601_422_12: YCbCr 4:2:2 12-bit unpacked BT.601.
- YCbCr601_422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked BT.601.
- YCbCr601_422_12p: YCbCr 4:2:2 12-bit packed BT.601.
- YCbCr601_422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed BT.601.
- YCbCr601_422_8: YCbCr 4:2:2 8-bit BT.601.
- YCbCr601_422_8_CbYCrY: YCbCr 4:2:2 8-bit BT.601.
- YCbCr601_8_CbYCr: YCbCr 4:4:4 8-bit BT.601.
- YCbCr709_10_CbYCr: YCbCr 4:4:4 10-bit unpacked BT.709.
- YCbCr709_10p_CbYCr: YCbCr 4:4:4 10-bit packed BT.709.
- YCbCr709_12_CbYCr: YCbCr 4:4:4 12-bit unpacked BT.709.
- YCbCr709_12p_CbYCr: YCbCr 4:4:4 12-bit packed BT.709.

- YCbCr709_411_8_CbYYCrYY: YCbCr 4:1:1 8-bit BT.709.
- YCbCr709_422_10: YCbCr 4:2:2 10-bit unpacked BT.709.
- YCbCr709_422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked BT.709.
- YCbCr709_422_10p: YCbCr 4:2:2 10-bit packed BT.709.
- YCbCr709_422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed BT.709.
- YCbCr709_422_12: YCbCr 4:2:2 12-bit unpacked BT.709.
- YCbCr709_422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked BT.709.
- YCbCr709_422_12p: YCbCr 4:2:2 12-bit packed BT.709.
- YCbCr709_422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed BT.709.
- YCbCr709_422_8: YCbCr 4:2:2 8-bit BT.709.
- YCbCr709_422_8_CbYCrY: YCbCr 4:2:2 8-bit BT.709.
- YCbCr709_8_CbYCr: YCbCr 4:4:4 8-bit BT.709.
- YCbCr8: YCbCr 4:4:4 8-bit.
- YCbCr8_CbYCr: YCbCr 4:4:4 8-bit.
- YUV411_8_UYYVYY: YUV 4:1:1 8-bit.
- YUV422_8: YUV 4:2:2 8-bit.
- YUV422_8_UYVY: YUV 4:2:2 8-bit.
- YUV422Packed: YUV422Packed.
- YUV8_UYV: YUV 4:4:4 8-bit.

PixelFormatNamespace

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → ImageFormatControl	Enumeration	Imposed: RO

Short Description

Namespace of the pixel format.

Enumeration Values

- Unknown: Unknown.
- GEV: GEV.
- IIDC: IIDC.
- PFNC_16BIT: PFNC 16-bit.
- PFNC_32BIT: PFNC 32-bit.

PixelSize

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → ImageFormatControl	IntReg	RO

Register Port: StreamPort

Short Description

Pixel size in bits.

PixelComponentCount

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → ImageFormatControl	IntReg	RO

Register Port: StreamPort

Short Description

Number of components per pixel.

Width

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → ImageFormatControl	Integer	Imposed: RO

Short Description

Width of the image.

Height

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → ImageFormatControl	Integer	Imposed: RO

Short Description

Height of the image.

ImageFormatSource

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → ImageFormatControl	Enumeration	RW

Short Description

Source of remote device image format.

Enumeration Values

- RemoteDevice: Remote device (PixelFormat, Width, Height).
- DataStream: Data stream (RemotePixelFormat, RemoteWidth, RemoteHeight).

RemotePixelFormat

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → ImageFormatControl	Enumeration	RW

Short Description

Pixel format of the remote device image.

Enumeration Values

- BayerBG10pmsb: BayerBG10pmsb.
- BayerBG12pmsb: BayerBG12pmsb.
- BayerBG14pmsb: BayerBG14pmsb.
- BayerGB10pmsb: BayerGB10pmsb.
- BayerGB12pmsb: BayerGB12pmsb.
- BayerGB14pmsb: BayerGB14pmsb.
- BayerGR10pmsb: BayerGR10pmsb.
- BayerGR12pmsb: BayerGR12pmsb.
- BayerGR14pmsb: BayerGR14pmsb.
- BayerRG10pmsb: BayerRG10pmsb.
- BayerRG12pmsb: BayerRG12pmsb.
- BayerRG14pmsb: BayerRG14pmsb.
- Mono10pmsb: Mono10pmsb.
- Mono12pmsb: Mono12pmsb.
- Mono14pmsb: Mono14pmsb.
- RGB10pmsb: RGB10pmsb.
- RGB12pmsb: RGB12pmsb.
- RGB14pmsb: RGB14pmsb.
- RGBa10pmsb: RGBa10pmsb.
- RGBa12pmsb: RGBa12pmsb.
- RGBa14pmsb: RGBa14pmsb.
- YCbCr601_10pmsb: YCbCr601_10pmsb.
- YCbCr601_12pmsb: YCbCr601_12pmsb.
- YCbCr601_14pmsb: YCbCr601_14pmsb.

- YCbCr601_16: YCbCr601_16.
- YCbCr601_411_10pmsb: YCbCr601_411_10pmsb.
- YCbCr601_411_12pmsb: YCbCr601_411_12pmsb.
- YCbCr601_411_14pmsb: YCbCr601_411_14pmsb.
- YCbCr601_411_16: YCbCr601_411_16.
- YCbCr601_411_8: YCbCr601_411_8.
- YCbCr601_422_10pmsb: YCbCr601_422_10pmsb.
- YCbCr601_422_12pmsb: YCbCr601_422_12pmsb.
- YCbCr601_422_14pmsb: YCbCr601_422_14pmsb.
- YCbCr601_422_16: YCbCr601_422_16.
- YCbCr601_8: YCbCr601_8.
- YCbCr709_10pmsb: YCbCr709_10pmsb.
- YCbCr709_12pmsb: YCbCr709_12pmsb.
- YCbCr709_14pmsb: YCbCr709_14pmsb.
- YCbCr709_16: YCbCr709_16.
- YCbCr709_411_10pmsb: YCbCr709_411_10pmsb.
- YCbCr709_411_12pmsb: YCbCr709_411_12pmsb.
- YCbCr709_411_14pmsb: YCbCr709_411_14pmsb.
- YCbCr709_411_16: YCbCr709_411_16.
- YCbCr709_411_8: YCbCr709_411_8.
- YCbCr709_422_10pmsb: YCbCr709_422_10pmsb.
- YCbCr709_422_12pmsb: YCbCr709_422_12pmsb.
- YCbCr709_422_14pmsb: YCbCr709_422_14pmsb.
- YCbCr709_422_16: YCbCr709_422_16.
- YCbCr709_8: YCbCr709_8.
- YUV10pmsb: YUV10pmsb.
- YUV12pmsb: YUV12pmsb.
- YUV14pmsb: YUV14pmsb.
- YUV16: YUV16.
- YUV411_10pmsb: YUV411_10pmsb.
- YUV411_12pmsb: YUV411_12pmsb.
- YUV411_14pmsb: YUV411_14pmsb.
- YUV411_16: YUV411_16.
- YUV411_8: YUV411_8.
- YUV422_10pmsb: YUV422_10pmsb.
- YUV422_12pmsb: YUV422_12pmsb.

- YUV422_14pmsb: YUV422_14pmsb.
- YUV422_16: YUV422_16.
- YUV8: YUV8.
- B10: Blue 10-bit.
- B12: Blue 12-bit.
- B16: Blue 16-bit.
- B8: Blue 8-bit.
- BayerBG10: Bayer Blue-Green 10-bit unpacked.
- BayerBG10p: Bayer Blue-Green 10-bit packed.
- BayerBG10Packed: Bayer Blue-Green 10-bit packed.
- BayerBG12: Bayer Blue-Green 12-bit unpacked.
- BayerBG12p: Bayer Blue-Green 12-bit packed.
- BayerBG12Packed: Bayer Blue-Green 12-bit packed.
- BayerBG14: Bayer Blue-Green 14-bit.
- BayerBG14p: Bayer Blue-Green 14-bit packed.
- BayerBG16: Bayer Blue-Green 16-bit.
- BayerBG4p: Bayer Blue-Green 4-bit packed.
- BayerBG8: Bayer Blue-Green 8-bit.
- BayerGB10: Bayer Green-Blue 10-bit unpacked.
- BayerGB10p: Bayer Green-Blue 10-bit packed.
- BayerGB10Packed: Bayer Green-Blue 10-bit packed.
- BayerGB12: Bayer Green-Blue 12-bit unpacked.
- BayerGB12p: Bayer Green-Blue 12-bit packed.
- BayerGB12Packed: Bayer Green-Blue 12-bit packed.
- BayerGB14: Bayer Green-Blue 14-bit.
- BayerGB14p: Bayer Green-Blue 14-bit packed.
- BayerGB16: Bayer Green-Blue 16-bit.
- BayerGB4p: Bayer Green-Blue 4-bit packed.
- BayerGB8: Bayer Green-Blue 8-bit.
- BayerGR10: Bayer Green-Red 10-bit unpacked.
- BayerGR10p: Bayer Green-Red 10-bit packed.
- BayerGR10Packed: Bayer Green-Red 10-bit packed.
- BayerGR12: Bayer Green-Red 12-bit unpacked.
- BayerGR12p: Bayer Green-Red 12-bit packed.
- BayerGR12Packed: Bayer Green-Red 12-bit packed.
- BayerGR14: Bayer Green-Red 14-bit.

- BayerGR14p: Bayer Green-Red 14-bit packed.
- BayerGR16: Bayer Green-Red 16-bit.
- BayerGR4p: Bayer Green-Red 4-bit packed.
- BayerGR8: Bayer Green-Red 8-bit.
- BayerRG10: Bayer Red-Green 10-bit unpacked.
- BayerRG10p: Bayer Red-Green 10-bit packed.
- BayerRG10Packed: Bayer Red-Green 10-bit packed.
- BayerRG12: Bayer Red-Green 12-bit unpacked.
- BayerRG12p: Bayer Red-Green 12-bit packed.
- BayerRG12Packed: Bayer Red-Green 12-bit packed.
- BayerRG14: Bayer Red-Green 14-bit.
- BayerRG14p: Bayer Red-Green 14-bit packed.
- BayerRG16: Bayer Red-Green 16-bit.
- BayerRG4p: Bayer Red-Green 4-bit packed.
- BayerRG8: Bayer Red-Green 8-bit.
- BGR10: Blue-Green-Red 10-bit unpacked.
- BGR10p: Blue-Green-Red 10-bit packed.
- BGR12: Blue-Green-Red 12-bit unpacked.
- BGR12p: Blue-Green-Red 12-bit packed.
- BGR14: Blue-Green-Red 14-bit unpacked.
- BGR16: Blue-Green-Red 16-bit.
- BGR565p: Blue-Green-Red 5/6/5-bit packed.
- BGR8: Blue-Green-Red 8-bit.
- BGR8a32: BGR8a32.
- BGRa10: Blue-Green-Red-alpha 10-bit unpacked.
- BGRa10p: Blue-Green-Red-alpha 10-bit packed.
- BGRa12: Blue-Green-Red-alpha 12-bit unpacked.
- BGRa12p: Blue-Green-Red-alpha 12-bit packed.
- BGRa14: Blue-Green-Red-alpha 14-bit unpacked.
- BGRa16: Blue-Green-Red-alpha 16-bit.
- BGRa8: Blue-Green-Red-alpha 8-bit.
- BiColorBGRG10: Bi-color Blue/Green - Red/Green 10-bit unpacked.
- BiColorBGRG10p: Bi-color Blue/Green - Red/Green 10-bit packed.
- BiColorBGRG12: Bi-color Blue/Green - Red/Green 12-bit unpacked.
- BiColorBGRG12p: Bi-color Blue/Green - Red/Green 12-bit packed.
- BiColorBGRG8: Bi-color Blue/Green - Red/Green 8-bit.

- BiColorRGBG10: Bi-color Red/Green - Blue/Green 10-bit unpacked.
- BiColorRGBG10p: Bi-color Red/Green - Blue/Green 10-bit packed.
- BiColorRGBG12: Bi-color Red/Green - Blue/Green 12-bit unpacked.
- BiColorRGBG12p: Bi-color Red/Green - Blue/Green 12-bit packed.
- BiColorRGBG8: Bi-color Red/Green - Blue/Green 8-bit.
- Confidence1: Confidence 1-bit unpacked.
- Confidence16: Confidence 16-bit.
- Confidence1p: Confidence 1-bit packed.
- Confidence32f: Confidence 32-bit floating point.
- Confidence8: Confidence 8-bit.
- Coord3D_A10p: 3D coordinate A 10-bit packed.
- Coord3D_A12p: 3D coordinate A 12-bit packed.
- Coord3D_A16: 3D coordinate A 16-bit.
- Coord3D_A32f: 3D coordinate A 32-bit floating point.
- Coord3D_A8: 3D coordinate A 8-bit.
- Coord3D_ABC10p: 3D coordinate A-B-C 10-bit packed.
- Coord3D_ABC10p_Planar: 3D coordinate A-B-C 10-bit packed planar.
- Coord3D_ABC12p: 3D coordinate A-B-C 12-bit packed.
- Coord3D_ABC12p_Planar: 3D coordinate A-B-C 12-bit packed planar.
- Coord3D_ABC16: 3D coordinate A-B-C 16-bit.
- Coord3D_ABC16_Planar: 3D coordinate A-B-C 16-bit planar.
- Coord3D_ABC32f: 3D coordinate A-B-C 32-bit floating point.
- Coord3D_ABC32f_Planar: 3D coordinate A-B-C 32-bit floating point planar.
- Coord3D_ABC8: 3D coordinate A-B-C 8-bit.
- Coord3D_ABC8_Planar: 3D coordinate A-B-C 8-bit planar.
- Coord3D_AC10p: 3D coordinate A-C 10-bit packed.
- Coord3D_AC10p_Planar: 3D coordinate A-C 10-bit packed planar.
- Coord3D_AC12p: 3D coordinate A-C 12-bit packed.
- Coord3D_AC12p_Planar: 3D coordinate A-C 12-bit packed planar.
- Coord3D_AC16: 3D coordinate A-C 16-bit.
- Coord3D_AC16_Planar: 3D coordinate A-C 16-bit planar.
- Coord3D_AC32f: 3D coordinate A-C 32-bit floating point.
- Coord3D_AC32f_Planar: 3D coordinate A-C 32-bit floating point planar.
- Coord3D_AC8: 3D coordinate A-C 8-bit.
- Coord3D_AC8_Planar: 3D coordinate A-C 8-bit planar.
- Coord3D_B10p: 3D coordinate B 10-bit packed.

- Coord3D_B12p: 3D coordinate B 12-bit packed.
- Coord3D_B16: 3D coordinate B 16-bit.
- Coord3D_B32f: 3D coordinate B 32-bit floating point.
- Coord3D_B8: 3D coordinate B 8-bit.
- Coord3D_C10p: 3D coordinate C 10-bit packed.
- Coord3D_C12p: 3D coordinate C 12-bit packed.
- Coord3D_C16: 3D coordinate C 16-bit.
- Coord3D_C32f: 3D coordinate C 32-bit floating point.
- Coord3D_C8: 3D coordinate C 8-bit.
- CustomBayerBG14: CustomBayerBG14.
- CustomBayerGB14: CustomBayerGB14.
- CustomBayerGR14: CustomBayerGR14.
- CustomBayerRG14: CustomBayerRG14.
- CustomJFIF: CustomJFIF.
- G10: Green 10-bit.
- G12: Green 12-bit.
- G16: Green 16-bit.
- G8: Green 8-bit.
- Mono10: Monochrome 10-bit unpacked.
- Mono10p: Monochrome 10-bit packed.
- Mono10Packed: Monochrome 10-bit packed.
- Mono12: Monochrome 12-bit unpacked.
- Mono12p: Monochrome 12-bit packed.
- Mono12Packed: Monochrome 12-bit packed.
- Mono14: Monochrome 14-bit unpacked.
- Mono14p: Monochrome 14-bit packed.
- Mono16: Monochrome 16-bit.
- Mono1p: Monochrome 1-bit packed.
- Mono2p: Monochrome 2-bit packed.
- Mono32: Monochrome 32-bit.
- Mono4p: Monochrome 4-bit packed.
- Mono8: Monochrome 8-bit.
- Mono8s: Monochrome 8-bit signed.
- R10: Red 10-bit.
- R12: Red 12-bit.
- R16: Red 16-bit.

- R8: Red 8-bit.
- RGB10: Red-Green-Blue 10-bit unpacked.
- RGB10_Planar: Red-Green-Blue 10-bit unpacked planar.
- RGB10p: Red-Green-Blue 10-bit packed.
- RGB10p32: Red-Green-Blue 10-bit packed into 32-bit.
- RGB10V1Packed: Red-Green-Blue 10-bit packed - variant 1.
- RGB12: Red-Green-Blue 12-bit unpacked.
- RGB12_Planar: Red-Green-Blue 12-bit unpacked planar.
- RGB12p: Red-Green-Blue 12-bit packed.
- RGB12V1Packed: Red-Green-Blue 12-bit packed - variant 1.
- RGB14: Red-Green-Blue 14-bit unpacked.
- RGB16: Red-Green-Blue 16-bit.
- RGB16_Planar: Red-Green-Blue 16-bit planar.
- RGB565p: Red-Green-Blue 5/6/5-bit packed.
- RGB8: Red-Green-Blue 8-bit.
- RGB8_Planar: Red-Green-Blue 8-bit planar.
- RGB8a32: RGB8a32.
- RGBa10: Red-Green-Blue-alpha 10-bit unpacked.
- RGBa10p: Red-Green-Blue-alpha 10-bit packed.
- RGBa12: Red-Green-Blue-alpha 12-bit unpacked.
- RGBa12p: Red-Green-Blue-alpha 12-bit packed.
- RGBa14: Red-Green-Blue-alpha 14-bit unpacked.
- RGBa16: Red-Green-Blue-alpha 16-bit.
- RGBa8: Red-Green-Blue-alpha 8-bit.
- SCF1WBWG10: Sparse Color Filter #1 White-Blue-White-Green 10-bit unpacked.
- SCF1WBWG10p: Sparse Color Filter #1 White-Blue-White-Green 10-bit packed.
- SCF1WBWG12: Sparse Color Filter #1 White-Blue-White-Green 12-bit unpacked.
- SCF1WBWG12p: Sparse Color Filter #1 White-Blue-White-Green 12-bit packed.
- SCF1WBWG14: Sparse Color Filter #1 White-Blue-White-Green 14-bit unpacked.
- SCF1WBWG16: Sparse Color Filter #1 White-Blue-White-Green 16-bit unpacked.
- SCF1WBWG8: Sparse Color Filter #1 White-Blue-White-Green 8-bit.
- SCF1WGWB10: Sparse Color Filter #1 White-Green-White-Blue 10-bit unpacked.
- SCF1WGWB10p: Sparse Color Filter #1 White-Green-White-Blue 10-bit packed.
- SCF1WGWB12: Sparse Color Filter #1 White-Green-White-Blue 12-bit unpacked.
- SCF1WGWB12p: Sparse Color Filter #1 White-Green-White-Blue 12-bit packed.
- SCF1WGWB14: Sparse Color Filter #1 White-Green-White-Blue 14-bit unpacked.

- SCF1WGWB16: Sparse Color Filter #1 White-Green-White-Blue 16-bit.
- SCF1WGWB8: Sparse Color Filter #1 White-Green-White-Blue 8-bit.
- SCF1WGWR10: Sparse Color Filter #1 White-Green-White-Red 10-bit unpacked.
- SCF1WGWR10p: Sparse Color Filter #1 White-Green-White-Red 10-bit packed.
- SCF1WGWR12: Sparse Color Filter #1 White-Green-White-Red 12-bit unpacked.
- SCF1WGWR12p: Sparse Color Filter #1 White-Green-White-Red 12-bit packed.
- SCF1WGWR14: Sparse Color Filter #1 White-Green-White-Red 14-bit unpacked.
- SCF1WGWR16: Sparse Color Filter #1 White-Green-White-Red 16-bit.
- SCF1WGWR8: Sparse Color Filter #1 White-Green-White-Red 8-bit.
- SCF1WRWG10: Sparse Color Filter #1 White-Red-White-Green 10-bit unpacked.
- SCF1WRWG10p: Sparse Color Filter #1 White-Red-White-Green 10-bit packed.
- SCF1WRWG12: Sparse Color Filter #1 White-Red-White-Green 12-bit unpacked.
- SCF1WRWG12p: Sparse Color Filter #1 White-Red-White-Green 12-bit packed.
- SCF1WRWG14: Sparse Color Filter #1 White-Red-White-Green 14-bit unpacked.
- SCF1WRWG16: Sparse Color Filter #1 White-Red-White-Green 16-bit.
- SCF1WRWG8: Sparse Color Filter #1 White-Red-White-Green 8-bit.
- YCbCr10_CbYCr: YCbCr 4:4:4 10-bit unpacked.
- YCbCr10p_CbYCr: YCbCr 4:4:4 10-bit packed.
- YCbCr12_CbYCr: YCbCr 4:4:4 12-bit unpacked.
- YCbCr12p_CbYCr: YCbCr 4:4:4 12-bit packed.
- YCbCr2020_10_CbYCr: YCbCr 4:4:4 10-bit unpacked BT.2020.
- YCbCr2020_10p_CbYCr: YCbCr 4:4:4 10-bit packed BT.2020.
- YCbCr2020_12_CbYCr: YCbCr 4:4:4 12-bit unpacked BT.2020.
- YCbCr2020_12p_CbYCr: YCbCr 4:4:4 12-bit packed BT.2020.
- YCbCr2020_411_8_CbYYCrYY: YCbCr 4:1:1 8-bit BT.2020.
- YCbCr2020_422_10: YCbCr 4:2:2 10-bit unpacked BT.2020.
- YCbCr2020_422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked BT.2020.
- YCbCr2020_422_10p: YCbCr 4:2:2 10-bit packed BT.2020.
- YCbCr2020_422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed BT.2020.
- YCbCr2020_422_12: YCbCr 4:2:2 12-bit unpacked BT.2020.
- YCbCr2020_422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked BT.2020.
- YCbCr2020_422_12p: YCbCr 4:2:2 12-bit packed BT.2020.
- YCbCr2020_422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed BT.2020.
- YCbCr2020_422_8: YCbCr 4:2:2 8-bit BT.2020.
- YCbCr2020_422_8_CbYCrY: YCbCr 4:2:2 8-bit BT.2020.
- YCbCr2020_8_CbYCr: YCbCr 4:4:4 8-bit BT.2020.

- YCbCr411_8: YCbCr 4:1:1 8-bit.
- YCbCr411_8_CbYYCrYY: YCbCr 4:1:1 8-bit.
- YCbCr420_8_YY_CbCr_Semiplanar: YCbCr 4:2:0 8-bit YY/CbCr Semiplanar.
- YCbCr420_8_YY_CrCb_Semiplanar: YCbCr 4:2:0 8-bit YY/CrCb Semiplanar.
- YCbCr422_10: YCbCr 4:2:2 10-bit unpacked.
- YCbCr422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked.
- YCbCr422_10p: YCbCr 4:2:2 10-bit packed.
- YCbCr422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed.
- YCbCr422_12: YCbCr 4:2:2 12-bit unpacked.
- YCbCr422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked.
- YCbCr422_12p: YCbCr 4:2:2 12-bit packed.
- YCbCr422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed.
- YCbCr422_8: YCbCr 4:2:2 8-bit.
- YCbCr422_8_CbYCrY: YCbCr 4:2:2 8-bit.
- YCbCr422_8_YY_CbCr_Semiplanar: YCbCr 4:2:2 8-bit YY/CbCr Semiplanar.
- YCbCr422_8_YY_CrCb_Semiplanar: YCbCr 4:2:2 8-bit YY/CrCb Semiplanar.
- YCbCr601_10_CbYCr: YCbCr 4:4:4 10-bit unpacked BT.601.
- YCbCr601_10p_CbYCr: YCbCr 4:4:4 10-bit packed BT.601.
- YCbCr601_12_CbYCr: YCbCr 4:4:4 12-bit unpacked BT.601.
- YCbCr601_12p_CbYCr: YCbCr 4:4:4 12-bit packed BT.601.
- YCbCr601_411_8_CbYYCrYY: YCbCr 4:1:1 8-bit BT.601.
- YCbCr601_422_10: YCbCr 4:2:2 10-bit unpacked BT.601.
- YCbCr601_422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked BT.601.
- YCbCr601_422_10p: YCbCr 4:2:2 10-bit packed BT.601.
- YCbCr601_422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed BT.601.
- YCbCr601_422_12: YCbCr 4:2:2 12-bit unpacked BT.601.
- YCbCr601_422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked BT.601.
- YCbCr601_422_12p: YCbCr 4:2:2 12-bit packed BT.601.
- YCbCr601_422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed BT.601.
- YCbCr601_422_8: YCbCr 4:2:2 8-bit BT.601.
- YCbCr601_422_8_CbYCrY: YCbCr 4:2:2 8-bit BT.601.
- YCbCr601_8_CbYCr: YCbCr 4:4:4 8-bit BT.601.
- YCbCr709_10_CbYCr: YCbCr 4:4:4 10-bit unpacked BT.709.
- YCbCr709_10p_CbYCr: YCbCr 4:4:4 10-bit packed BT.709.
- YCbCr709_12_CbYCr: YCbCr 4:4:4 12-bit unpacked BT.709.
- YCbCr709_12p_CbYCr: YCbCr 4:4:4 12-bit packed BT.709.

- YCbCr709_411_8_CbYYCrYY: YCbCr 4:1:1 8-bit BT.709.
- YCbCr709_422_10: YCbCr 4:2:2 10-bit unpacked BT.709.
- YCbCr709_422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked BT.709.
- YCbCr709_422_10p: YCbCr 4:2:2 10-bit packed BT.709.
- YCbCr709_422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed BT.709.
- YCbCr709_422_12: YCbCr 4:2:2 12-bit unpacked BT.709.
- YCbCr709_422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked BT.709.
- YCbCr709_422_12p: YCbCr 4:2:2 12-bit packed BT.709.
- YCbCr709_422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed BT.709.
- YCbCr709_422_8: YCbCr 4:2:2 8-bit BT.709.
- YCbCr709_422_8_CbYCrY: YCbCr 4:2:2 8-bit BT.709.
- YCbCr709_8_CbYCr: YCbCr 4:4:4 8-bit BT.709.
- YCbCr8: YCbCr 4:4:4 8-bit.
- YCbCr8_CbYCr: YCbCr 4:4:4 8-bit.
- YUV411_8_UYYVYY: YUV 4:1:1 8-bit.
- YUV422_8: YUV 4:2:2 8-bit.
- YUV422_8_UYVY: YUV 4:2:2 8-bit.
- YUV422Packed: YUV422Packed.
- YUV8_UYV: YUV 4:4:4 8-bit.

RemoteWidth

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → ImageFormatControl	Integer	RW

Value Info

Minimum value: 1

Short Description

Width of the remote device image.

RemoteHeight

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → ImageFormatControl	Integer	RW

Value Info

Minimum value: 1

Short Description

Height of the remote device image.

4.4. TransportLayerControl Category

PayloadSize

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → TransportLayerControl	Integer	Imposed: RO

Short Description

Expected size of buffers for the data stream.

ControlRemoteDevice

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → TransportLayerControl	Boolean	RW

Short Description

Defines whether or not to start and stop the remote device by automatically executing AcquisitionStart and AcquisitionStop commands.

4.5. BufferHandlingControl Category

StreamAnnouncedBufferCount

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Integer	Imposed: RO

Short Description

Number of announced buffers on the stream.

StreamBufferHandlingMode

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Enumeration	RW

Short Description

Available buffer handling modes of this Stream.

Enumeration Values

- Default: Default Buffer Handling Mode.

StreamAnnounceBufferMinimum

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Integer	Imposed: RO

Short Description

Minimal number of buffers to announce to enable selected buffer handling mode.

StreamAcquisitionModeSelector

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Enumeration	RW

Short Description

Available buffer handling modes of this Stream. Deprecated.

Enumeration Values

- Default: Default Buffer Handling Mode.

BufferAllocationAlignmentControl

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Enumeration	RW

Short Description

Buffer Allocation Alignment Control.

Enumeration Values

- Disable: Disable aligned buffer allocation.
- Enable: Enable aligned buffer allocation.

BufferAllocationAlignment

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Integer	RW

Value Info

Minimum value: 1

Short Description

Alignment of buffers allocated by DSAllocAndAnnounceBuffer, should be a power of 2.

BufferInfoSource

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Enumeration	RW

Short Description

Source of buffer info.

Enumeration Values

- ImageHeader: Remote device image header (PixelFormat, Width, Height).
- DataStream: Data stream (BufferInfoPixelFormat, BufferInfoWidth, BufferInfoHeight).

BufferInfoWidth

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Integer	RW

Value Info

Minimum value: 1

Short Description

Imposed width.

BufferInfoHeight

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Integer	RW

Value Info

Minimum value: 0

Short Description

Imposed height.

BufferInfoPixelFormat

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Enumeration	RW

Short Description

Imposed pixel format.

Enumeration Values

- BayerBG10pmsb: BayerBG10pmsb.
- BayerBG12pmsb: BayerBG12pmsb.
- BayerBG14pmsb: BayerBG14pmsb.
- BayerGB10pmsb: BayerGB10pmsb.
- BayerGB12pmsb: BayerGB12pmsb.
- BayerGB14pmsb: BayerGB14pmsb.
- BayerGR10pmsb: BayerGR10pmsb.
- BayerGR12pmsb: BayerGR12pmsb.
- BayerGR14pmsb: BayerGR14pmsb.
- BayerRG10pmsb: BayerRG10pmsb.
- BayerRG12pmsb: BayerRG12pmsb.
- BayerRG14pmsb: BayerRG14pmsb.
- Mono10pmsb: Mono10pmsb.
- Mono12pmsb: Mono12pmsb.
- Mono14pmsb: Mono14pmsb.
- RGB10pmsb: RGB10pmsb.
- RGB12pmsb: RGB12pmsb.
- RGB14pmsb: RGB14pmsb.
- RGBa10pmsb: RGBa10pmsb.
- RGBa12pmsb: RGBa12pmsb.
- RGBa14pmsb: RGBa14pmsb.
- YCbCr601_10pmsb: YCbCr601_10pmsb.
- YCbCr601_12pmsb: YCbCr601_12pmsb.
- YCbCr601_14pmsb: YCbCr601_14pmsb.

- YCbCr601_16: YCbCr601_16.
- YCbCr601_411_10pmsb: YCbCr601_411_10pmsb.
- YCbCr601_411_12pmsb: YCbCr601_411_12pmsb.
- YCbCr601_411_14pmsb: YCbCr601_411_14pmsb.
- YCbCr601_411_16: YCbCr601_411_16.
- YCbCr601_411_8: YCbCr601_411_8.
- YCbCr601_422_10pmsb: YCbCr601_422_10pmsb.
- YCbCr601_422_12pmsb: YCbCr601_422_12pmsb.
- YCbCr601_422_14pmsb: YCbCr601_422_14pmsb.
- YCbCr601_422_16: YCbCr601_422_16.
- YCbCr601_8: YCbCr601_8.
- YCbCr709_10pmsb: YCbCr709_10pmsb.
- YCbCr709_12pmsb: YCbCr709_12pmsb.
- YCbCr709_14pmsb: YCbCr709_14pmsb.
- YCbCr709_16: YCbCr709_16.
- YCbCr709_411_10pmsb: YCbCr709_411_10pmsb.
- YCbCr709_411_12pmsb: YCbCr709_411_12pmsb.
- YCbCr709_411_14pmsb: YCbCr709_411_14pmsb.
- YCbCr709_411_16: YCbCr709_411_16.
- YCbCr709_411_8: YCbCr709_411_8.
- YCbCr709_422_10pmsb: YCbCr709_422_10pmsb.
- YCbCr709_422_12pmsb: YCbCr709_422_12pmsb.
- YCbCr709_422_14pmsb: YCbCr709_422_14pmsb.
- YCbCr709_422_16: YCbCr709_422_16.
- YCbCr709_8: YCbCr709_8.
- YUV10pmsb: YUV10pmsb.
- YUV12pmsb: YUV12pmsb.
- YUV14pmsb: YUV14pmsb.
- YUV16: YUV16.
- YUV411_10pmsb: YUV411_10pmsb.
- YUV411_12pmsb: YUV411_12pmsb.
- YUV411_14pmsb: YUV411_14pmsb.
- YUV411_16: YUV411_16.
- YUV411_8: YUV411_8.
- YUV422_10pmsb: YUV422_10pmsb.
- YUV422_12pmsb: YUV422_12pmsb.

- YUV422_14pmsb: YUV422_14pmsb.
- YUV422_16: YUV422_16.
- YUV8: YUV8.
- B10: Blue 10-bit.
- B12: Blue 12-bit.
- B16: Blue 16-bit.
- B8: Blue 8-bit.
- BayerBG10: Bayer Blue-Green 10-bit unpacked.
- BayerBG10p: Bayer Blue-Green 10-bit packed.
- BayerBG10Packed: Bayer Blue-Green 10-bit packed.
- BayerBG12: Bayer Blue-Green 12-bit unpacked.
- BayerBG12p: Bayer Blue-Green 12-bit packed.
- BayerBG12Packed: Bayer Blue-Green 12-bit packed.
- BayerBG14: Bayer Blue-Green 14-bit.
- BayerBG14p: Bayer Blue-Green 14-bit packed.
- BayerBG16: Bayer Blue-Green 16-bit.
- BayerBG4p: Bayer Blue-Green 4-bit packed.
- BayerBG8: Bayer Blue-Green 8-bit.
- BayerGB10: Bayer Green-Blue 10-bit unpacked.
- BayerGB10p: Bayer Green-Blue 10-bit packed.
- BayerGB10Packed: Bayer Green-Blue 10-bit packed.
- BayerGB12: Bayer Green-Blue 12-bit unpacked.
- BayerGB12p: Bayer Green-Blue 12-bit packed.
- BayerGB12Packed: Bayer Green-Blue 12-bit packed.
- BayerGB14: Bayer Green-Blue 14-bit.
- BayerGB14p: Bayer Green-Blue 14-bit packed.
- BayerGB16: Bayer Green-Blue 16-bit.
- BayerGB4p: Bayer Green-Blue 4-bit packed.
- BayerGB8: Bayer Green-Blue 8-bit.
- BayerGR10: Bayer Green-Red 10-bit unpacked.
- BayerGR10p: Bayer Green-Red 10-bit packed.
- BayerGR10Packed: Bayer Green-Red 10-bit packed.
- BayerGR12: Bayer Green-Red 12-bit unpacked.
- BayerGR12p: Bayer Green-Red 12-bit packed.
- BayerGR12Packed: Bayer Green-Red 12-bit packed.
- BayerGR14: Bayer Green-Red 14-bit.

- BayerGR14p: Bayer Green-Red 14-bit packed.
- BayerGR16: Bayer Green-Red 16-bit.
- BayerGR4p: Bayer Green-Red 4-bit packed.
- BayerGR8: Bayer Green-Red 8-bit.
- BayerRG10: Bayer Red-Green 10-bit unpacked.
- BayerRG10p: Bayer Red-Green 10-bit packed.
- BayerRG10Packed: Bayer Red-Green 10-bit packed.
- BayerRG12: Bayer Red-Green 12-bit unpacked.
- BayerRG12p: Bayer Red-Green 12-bit packed.
- BayerRG12Packed: Bayer Red-Green 12-bit packed.
- BayerRG14: Bayer Red-Green 14-bit.
- BayerRG14p: Bayer Red-Green 14-bit packed.
- BayerRG16: Bayer Red-Green 16-bit.
- BayerRG4p: Bayer Red-Green 4-bit packed.
- BayerRG8: Bayer Red-Green 8-bit.
- BGR10: Blue-Green-Red 10-bit unpacked.
- BGR10p: Blue-Green-Red 10-bit packed.
- BGR12: Blue-Green-Red 12-bit unpacked.
- BGR12p: Blue-Green-Red 12-bit packed.
- BGR14: Blue-Green-Red 14-bit unpacked.
- BGR16: Blue-Green-Red 16-bit.
- BGR565p: Blue-Green-Red 5/6/5-bit packed.
- BGR8: Blue-Green-Red 8-bit.
- BGR8a32: BGR8a32.
- BGRa10: Blue-Green-Red-alpha 10-bit unpacked.
- BGRa10p: Blue-Green-Red-alpha 10-bit packed.
- BGRa12: Blue-Green-Red-alpha 12-bit unpacked.
- BGRa12p: Blue-Green-Red-alpha 12-bit packed.
- BGRa14: Blue-Green-Red-alpha 14-bit unpacked.
- BGRa16: Blue-Green-Red-alpha 16-bit.
- BGRa8: Blue-Green-Red-alpha 8-bit.
- BiColorBGRG10: Bi-color Blue/Green - Red/Green 10-bit unpacked.
- BiColorBGRG10p: Bi-color Blue/Green - Red/Green 10-bit packed.
- BiColorBGRG12: Bi-color Blue/Green - Red/Green 12-bit unpacked.
- BiColorBGRG12p: Bi-color Blue/Green - Red/Green 12-bit packed.
- BiColorBGRG8: Bi-color Blue/Green - Red/Green 8-bit.

- BiColorRGBG10: Bi-color Red/Green - Blue/Green 10-bit unpacked.
- BiColorRGBG10p: Bi-color Red/Green - Blue/Green 10-bit packed.
- BiColorRGBG12: Bi-color Red/Green - Blue/Green 12-bit unpacked.
- BiColorRGBG12p: Bi-color Red/Green - Blue/Green 12-bit packed.
- BiColorRGBG8: Bi-color Red/Green - Blue/Green 8-bit.
- Confidence1: Confidence 1-bit unpacked.
- Confidence16: Confidence 16-bit.
- Confidence1p: Confidence 1-bit packed.
- Confidence32f: Confidence 32-bit floating point.
- Confidence8: Confidence 8-bit.
- Coord3D_A10p: 3D coordinate A 10-bit packed.
- Coord3D_A12p: 3D coordinate A 12-bit packed.
- Coord3D_A16: 3D coordinate A 16-bit.
- Coord3D_A32f: 3D coordinate A 32-bit floating point.
- Coord3D_A8: 3D coordinate A 8-bit.
- Coord3D_ABC10p: 3D coordinate A-B-C 10-bit packed.
- Coord3D_ABC10p_Planar: 3D coordinate A-B-C 10-bit packed planar.
- Coord3D_ABC12p: 3D coordinate A-B-C 12-bit packed.
- Coord3D_ABC12p_Planar: 3D coordinate A-B-C 12-bit packed planar.
- Coord3D_ABC16: 3D coordinate A-B-C 16-bit.
- Coord3D_ABC16_Planar: 3D coordinate A-B-C 16-bit planar.
- Coord3D_ABC32f: 3D coordinate A-B-C 32-bit floating point.
- Coord3D_ABC32f_Planar: 3D coordinate A-B-C 32-bit floating point planar.
- Coord3D_ABC8: 3D coordinate A-B-C 8-bit.
- Coord3D_ABC8_Planar: 3D coordinate A-B-C 8-bit planar.
- Coord3D_AC10p: 3D coordinate A-C 10-bit packed.
- Coord3D_AC10p_Planar: 3D coordinate A-C 10-bit packed planar.
- Coord3D_AC12p: 3D coordinate A-C 12-bit packed.
- Coord3D_AC12p_Planar: 3D coordinate A-C 12-bit packed planar.
- Coord3D_AC16: 3D coordinate A-C 16-bit.
- Coord3D_AC16_Planar: 3D coordinate A-C 16-bit planar.
- Coord3D_AC32f: 3D coordinate A-C 32-bit floating point.
- Coord3D_AC32f_Planar: 3D coordinate A-C 32-bit floating point planar.
- Coord3D_AC8: 3D coordinate A-C 8-bit.
- Coord3D_AC8_Planar: 3D coordinate A-C 8-bit planar.
- Coord3D_B10p: 3D coordinate B 10-bit packed.

- Coord3D_B12p: 3D coordinate B 12-bit packed.
- Coord3D_B16: 3D coordinate B 16-bit.
- Coord3D_B32f: 3D coordinate B 32-bit floating point.
- Coord3D_B8: 3D coordinate B 8-bit.
- Coord3D_C10p: 3D coordinate C 10-bit packed.
- Coord3D_C12p: 3D coordinate C 12-bit packed.
- Coord3D_C16: 3D coordinate C 16-bit.
- Coord3D_C32f: 3D coordinate C 32-bit floating point.
- Coord3D_C8: 3D coordinate C 8-bit.
- CustomBayerBG14: CustomBayerBG14.
- CustomBayerGB14: CustomBayerGB14.
- CustomBayerGR14: CustomBayerGR14.
- CustomBayerRG14: CustomBayerRG14.
- CustomJFIF: CustomJFIF.
- G10: Green 10-bit.
- G12: Green 12-bit.
- G16: Green 16-bit.
- G8: Green 8-bit.
- Mono10: Monochrome 10-bit unpacked.
- Mono10p: Monochrome 10-bit packed.
- Mono10Packed: Monochrome 10-bit packed.
- Mono12: Monochrome 12-bit unpacked.
- Mono12p: Monochrome 12-bit packed.
- Mono12Packed: Monochrome 12-bit packed.
- Mono14: Monochrome 14-bit unpacked.
- Mono14p: Monochrome 14-bit packed.
- Mono16: Monochrome 16-bit.
- Mono1p: Monochrome 1-bit packed.
- Mono2p: Monochrome 2-bit packed.
- Mono32: Monochrome 32-bit.
- Mono4p: Monochrome 4-bit packed.
- Mono8: Monochrome 8-bit.
- Mono8s: Monochrome 8-bit signed.
- R10: Red 10-bit.
- R12: Red 12-bit.
- R16: Red 16-bit.

- R8: Red 8-bit.
- RGB10: Red-Green-Blue 10-bit unpacked.
- RGB10_Planar: Red-Green-Blue 10-bit unpacked planar.
- RGB10p: Red-Green-Blue 10-bit packed.
- RGB10p32: Red-Green-Blue 10-bit packed into 32-bit.
- RGB10V1Packed: Red-Green-Blue 10-bit packed - variant 1.
- RGB12: Red-Green-Blue 12-bit unpacked.
- RGB12_Planar: Red-Green-Blue 12-bit unpacked planar.
- RGB12p: Red-Green-Blue 12-bit packed.
- RGB12V1Packed: Red-Green-Blue 12-bit packed - variant 1.
- RGB14: Red-Green-Blue 14-bit unpacked.
- RGB16: Red-Green-Blue 16-bit.
- RGB16_Planar: Red-Green-Blue 16-bit planar.
- RGB565p: Red-Green-Blue 5/6/5-bit packed.
- RGB8: Red-Green-Blue 8-bit.
- RGB8_Planar: Red-Green-Blue 8-bit planar.
- RGB8a32: RGB8a32.
- RGBa10: Red-Green-Blue-alpha 10-bit unpacked.
- RGBa10p: Red-Green-Blue-alpha 10-bit packed.
- RGBa12: Red-Green-Blue-alpha 12-bit unpacked.
- RGBa12p: Red-Green-Blue-alpha 12-bit packed.
- RGBa14: Red-Green-Blue-alpha 14-bit unpacked.
- RGBa16: Red-Green-Blue-alpha 16-bit.
- RGBa8: Red-Green-Blue-alpha 8-bit.
- SCF1WBWG10: Sparse Color Filter #1 White-Blue-White-Green 10-bit unpacked.
- SCF1WBWG10p: Sparse Color Filter #1 White-Blue-White-Green 10-bit packed.
- SCF1WBWG12: Sparse Color Filter #1 White-Blue-White-Green 12-bit unpacked.
- SCF1WBWG12p: Sparse Color Filter #1 White-Blue-White-Green 12-bit packed.
- SCF1WBWG14: Sparse Color Filter #1 White-Blue-White-Green 14-bit unpacked.
- SCF1WBWG16: Sparse Color Filter #1 White-Blue-White-Green 16-bit unpacked.
- SCF1WBWG8: Sparse Color Filter #1 White-Blue-White-Green 8-bit.
- SCF1WGWB10: Sparse Color Filter #1 White-Green-White-Blue 10-bit unpacked.
- SCF1WGWB10p: Sparse Color Filter #1 White-Green-White-Blue 10-bit packed.
- SCF1WGWB12: Sparse Color Filter #1 White-Green-White-Blue 12-bit unpacked.
- SCF1WGWB12p: Sparse Color Filter #1 White-Green-White-Blue 12-bit packed.
- SCF1WGWB14: Sparse Color Filter #1 White-Green-White-Blue 14-bit unpacked.

- SCF1WGWB16: Sparse Color Filter #1 White-Green-White-Blue 16-bit.
- SCF1WGWB8: Sparse Color Filter #1 White-Green-White-Blue 8-bit.
- SCF1WGWR10: Sparse Color Filter #1 White-Green-White-Red 10-bit unpacked.
- SCF1WGWR10p: Sparse Color Filter #1 White-Green-White-Red 10-bit packed.
- SCF1WGWR12: Sparse Color Filter #1 White-Green-White-Red 12-bit unpacked.
- SCF1WGWR12p: Sparse Color Filter #1 White-Green-White-Red 12-bit packed.
- SCF1WGWR14: Sparse Color Filter #1 White-Green-White-Red 14-bit unpacked.
- SCF1WGWR16: Sparse Color Filter #1 White-Green-White-Red 16-bit.
- SCF1WGWR8: Sparse Color Filter #1 White-Green-White-Red 8-bit.
- SCF1WRWG10: Sparse Color Filter #1 White-Red-White-Green 10-bit unpacked.
- SCF1WRWG10p: Sparse Color Filter #1 White-Red-White-Green 10-bit packed.
- SCF1WRWG12: Sparse Color Filter #1 White-Red-White-Green 12-bit unpacked.
- SCF1WRWG12p: Sparse Color Filter #1 White-Red-White-Green 12-bit packed.
- SCF1WRWG14: Sparse Color Filter #1 White-Red-White-Green 14-bit unpacked.
- SCF1WRWG16: Sparse Color Filter #1 White-Red-White-Green 16-bit.
- SCF1WRWG8: Sparse Color Filter #1 White-Red-White-Green 8-bit.
- YCbCr10_CbYCr: YCbCr 4:4:4 10-bit unpacked.
- YCbCr10p_CbYCr: YCbCr 4:4:4 10-bit packed.
- YCbCr12_CbYCr: YCbCr 4:4:4 12-bit unpacked.
- YCbCr12p_CbYCr: YCbCr 4:4:4 12-bit packed.
- YCbCr2020_10_CbYCr: YCbCr 4:4:4 10-bit unpacked BT.2020.
- YCbCr2020_10p_CbYCr: YCbCr 4:4:4 10-bit packed BT.2020.
- YCbCr2020_12_CbYCr: YCbCr 4:4:4 12-bit unpacked BT.2020.
- YCbCr2020_12p_CbYCr: YCbCr 4:4:4 12-bit packed BT.2020.
- YCbCr2020_411_8_CbYYCrYY: YCbCr 4:1:1 8-bit BT.2020.
- YCbCr2020_422_10: YCbCr 4:2:2 10-bit unpacked BT.2020.
- YCbCr2020_422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked BT.2020.
- YCbCr2020_422_10p: YCbCr 4:2:2 10-bit packed BT.2020.
- YCbCr2020_422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed BT.2020.
- YCbCr2020_422_12: YCbCr 4:2:2 12-bit unpacked BT.2020.
- YCbCr2020_422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked BT.2020.
- YCbCr2020_422_12p: YCbCr 4:2:2 12-bit packed BT.2020.
- YCbCr2020_422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed BT.2020.
- YCbCr2020_422_8: YCbCr 4:2:2 8-bit BT.2020.
- YCbCr2020_422_8_CbYCrY: YCbCr 4:2:2 8-bit BT.2020.
- YCbCr2020_8_CbYCr: YCbCr 4:4:4 8-bit BT.2020.

- YCbCr411_8: YCbCr 4:1:1 8-bit.
- YCbCr411_8_CbYYCrYY: YCbCr 4:1:1 8-bit.
- YCbCr420_8_YY_CbCr_Semiplanar: YCbCr 4:2:0 8-bit YY/CbCr Semiplanar.
- YCbCr420_8_YY_CrCb_Semiplanar: YCbCr 4:2:0 8-bit YY/CrCb Semiplanar.
- YCbCr422_10: YCbCr 4:2:2 10-bit unpacked.
- YCbCr422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked.
- YCbCr422_10p: YCbCr 4:2:2 10-bit packed.
- YCbCr422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed.
- YCbCr422_12: YCbCr 4:2:2 12-bit unpacked.
- YCbCr422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked.
- YCbCr422_12p: YCbCr 4:2:2 12-bit packed.
- YCbCr422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed.
- YCbCr422_8: YCbCr 4:2:2 8-bit.
- YCbCr422_8_CbYCrY: YCbCr 4:2:2 8-bit.
- YCbCr422_8_YY_CbCr_Semiplanar: YCbCr 4:2:2 8-bit YY/CbCr Semiplanar.
- YCbCr422_8_YY_CrCb_Semiplanar: YCbCr 4:2:2 8-bit YY/CrCb Semiplanar.
- YCbCr601_10_CbYCr: YCbCr 4:4:4 10-bit unpacked BT.601.
- YCbCr601_10p_CbYCr: YCbCr 4:4:4 10-bit packed BT.601.
- YCbCr601_12_CbYCr: YCbCr 4:4:4 12-bit unpacked BT.601.
- YCbCr601_12p_CbYCr: YCbCr 4:4:4 12-bit packed BT.601.
- YCbCr601_411_8_CbYYCrYY: YCbCr 4:1:1 8-bit BT.601.
- YCbCr601_422_10: YCbCr 4:2:2 10-bit unpacked BT.601.
- YCbCr601_422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked BT.601.
- YCbCr601_422_10p: YCbCr 4:2:2 10-bit packed BT.601.
- YCbCr601_422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed BT.601.
- YCbCr601_422_12: YCbCr 4:2:2 12-bit unpacked BT.601.
- YCbCr601_422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked BT.601.
- YCbCr601_422_12p: YCbCr 4:2:2 12-bit packed BT.601.
- YCbCr601_422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed BT.601.
- YCbCr601_422_8: YCbCr 4:2:2 8-bit BT.601.
- YCbCr601_422_8_CbYCrY: YCbCr 4:2:2 8-bit BT.601.
- YCbCr601_8_CbYCr: YCbCr 4:4:4 8-bit BT.601.
- YCbCr709_10_CbYCr: YCbCr 4:4:4 10-bit unpacked BT.709.
- YCbCr709_10p_CbYCr: YCbCr 4:4:4 10-bit packed BT.709.
- YCbCr709_12_CbYCr: YCbCr 4:4:4 12-bit unpacked BT.709.
- YCbCr709_12p_CbYCr: YCbCr 4:4:4 12-bit packed BT.709.

- YCbCr709_411_8_CbYYCrYY: YCbCr 4:1:1 8-bit BT.709.
- YCbCr709_422_10: YCbCr 4:2:2 10-bit unpacked BT.709.
- YCbCr709_422_10_CbYCrY: YCbCr 4:2:2 10-bit unpacked BT.709.
- YCbCr709_422_10p: YCbCr 4:2:2 10-bit packed BT.709.
- YCbCr709_422_10p_CbYCrY: YCbCr 4:2:2 10-bit packed BT.709.
- YCbCr709_422_12: YCbCr 4:2:2 12-bit unpacked BT.709.
- YCbCr709_422_12_CbYCrY: YCbCr 4:2:2 12-bit unpacked BT.709.
- YCbCr709_422_12p: YCbCr 4:2:2 12-bit packed BT.709.
- YCbCr709_422_12p_CbYCrY: YCbCr 4:2:2 12-bit packed BT.709.
- YCbCr709_422_8: YCbCr 4:2:2 8-bit BT.709.
- YCbCr709_422_8_CbYCrY: YCbCr 4:2:2 8-bit BT.709.
- YCbCr709_8_CbYCr: YCbCr 4:4:4 8-bit BT.709.
- YCbCr8: YCbCr 4:4:4 8-bit.
- YCbCr8_CbYCr: YCbCr 4:4:4 8-bit.
- YUV411_8_UYYVYY: YUV 4:1:1 8-bit.
- YUV422_8: YUV 4:2:2 8-bit.
- YUV422_8_UYVY: YUV 4:2:2 8-bit.
- YUV422Packed: YUV422Packed.
- YUV8_UYV: YUV 4:4:4 8-bit.

BufferHeight

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Integer	RW

Value Info

Unit: lines

Short Description

Height of the image in line-scan mode. This feature is only used in line-scan acquisition scenarios to compute PayloadSize

DeliverIncompleteImages

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → BufferHandlingControl	Boolean	RW

Short Description

Deliver Incomplete Images.

4.6. GigEVision Category

GevPacketResend

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → GigEVision	Boolean	Imposed: RO

Short Description

Gev Packet Resend.

SetPacketSizeToMaximum

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → GigEVision	Command	Imposed: WO

Short Description

Send test packets to figure out maximum packet size for this stream and save the highest working size into GevSCPSPacketSize.

WARNING
This feature can only be activated or de-actived when the data-stream is inactive.

FilterDriverEnable

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → GigEVision	Boolean	RW

Short Description

Toggle reception of data through filter driver.

4.7. StreamControl Category

StreamReset

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamControl	Command	Imposed: WO

Short Description

Stream Reset.

WARNING
This feature can only be activated or de-actived when the data-stream is inactive.

ActivateCic

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamControl	Command	Imposed: WO

Short Description

Activate the Camera and Illumination Controller if the data stream was started with ACQ_START_FLAGS_CUSTOM_DO_NOT_ACTIVATE_SEQUENCER.

DeactivateCic

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamControl	Command	Imposed: WO

Short Description

Deactivate the Camera and Illumination Controller.

4.8. EventControl Category

EventSelector

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → EventControl	Enumeration	RW

Short Description

Select an event.

Selected Features

- "EventNotification" on page 180
- "EventNotificationContext1" on page 181
- "EventNotificationContext2" on page 182
- "EventNotificationContext3" on page 183
- "EventCount" on page 184
- "EventCountReset" on page 185

Enumeration Values

- LostFrames: Frames never received. Cause unknown.
- CancelledFrames: Frames never sent by the camera.
- IncompleteFrame: Frames delivered while there are still missing packets.
- Failure: Fatal error that terminated acquisition.

EventNotification

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → EventControl	Boolean	RW

Short Description

Activate or deactivate the notification to the host application of the occurrence of the selected event.

EventNotificationContext1

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → EventControl	Enumeration	RW

Short Description

Select context information reported in EVENT_DATA_CUSTOM_CONTEXT_1.

Enumeration Values

- Nothing: No information.
- EventSpecific: Event-specific context information.
- BlockIDlo: GVSP Block identifier (lower half).
- BlockIDhi: GVSP Block identifier (higher half).

EventNotificationContext2

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → EventControl	Enumeration	RW

Short Description

Select context information reported in EVENT_DATA_CUSTOM_CONTEXT_2.

Enumeration Values

- Nothing: No information.
- EventSpecific: Event-specific context information.
- BlockIDlo: GVSP Block identifier (lower half).
- BlockIDhi: GVSP Block identifier (higher half).

EventNotificationContext3

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → EventControl	Enumeration	RW

Short Description

Select context information reported in EVENT_DATA_CUSTOM_CONTEXT_3.

Enumeration Values

- Nothing: No information.
- EventSpecific: Event-specific context information.
- BlockIDlo: GVSP Block identifier (lower half).
- BlockIDhi: GVSP Block identifier (higher half).

EventCount

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → EventControl	IntReg	RO

Register Port: StreamPort

Short Description

Number of occurrences of the selected event (32-bit counter).

EventCountReset

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → EventControl	Command	Imposed: WO

Short Description

Reset the selected EventCount.

4.9. StreamStatistics Category

StatisticsSamplingSelector

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamStatistics	Enumeration	RW

Description

Selects the stream statistics sampling method.

Default value: LastSecond.

Selected Features

- "StatisticsFrameRate" on page 188
- "StatisticsLineRate" on page 189
- "StatisticsDataRate" on page 190

Enumeration Values

- LastSecond: During the last second.
- LastTenSeconds: During the last 10 seconds.
- Last2Buffers: For the last 2 buffers.
- Last10Buffers: For the last 10 buffers.
- Last100Buffers: For the last 100 buffers.
- Last1000Buffers: For the last 1000 buffers.
- LastAcquisition: During the last acquisition activity period. Namely since the last DSStartAcquisition() function call until now, if the acquisition is still active otherwise until the last DSStopAcquisition() function call.
- Custom: Custom sampling using StatisticsStartSampling and StatisticsStopSampling commands.

StatisticsFrameRate

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamStatistics	FloatReg	RO

Register Port: StreamPort

Value Info

Unit: Fps (Frames per second)

Description

Average frame delivery rate using the selected sampling method.

NOTE
This feature is only available for area-scan firmware variants.

NOTE
The statistics measures the frame rate at the level of the PCI Express interface, NOT at the level of the CoaXPress interface!

StatisticsLineRate

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamStatistics	FloatReg	RO

Register Port: StreamPort

Value Info

Unit: Lps (Lines per second)

Description

Average line delivery rate using the selected sampling method.

NOTE
This feature is only available for line-scan firmware variants.

NOTE
The statistics measures the line rate at the level of the PCI Express interface, NOT at the level of the CoaXPress interface!

StatisticsDataRate

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamStatistics	FloatReg	RO

Register Port: StreamPort

Value Info

Unit: MBps (Megabytes per second)

Short Description

Get the average PCI data delivery rate using the selected sampling method.

StatisticsStartSampling

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamStatistics	Command	Imposed: WO

Short Description

Start sampling the stream data. Applies only when StatisticsSamplingSelector = Custom.

StatisticsStopSampling

Feature Info

Module	Category Path	Type	Access
Data Stream	Root → StreamStatistics	Command	Imposed: WO

Short Description

Stop sampling the stream data. Applies only when StatisticsSamplingSelector = Custom.