Novitec Camera C# API

Generated by Doxygen 1.9.3

Chapter 1

Introduction to the C# interface reference

1.1 Introduction

This is the documentation for developers who want to work with the C# interface of Novitec Camera API.

1.2 Image acquisition

Every program written using this interface will start in one or the other form with creating an instance of the class novitec.CameraAPI.CLR.DeviceManager. Each application needs at least one instance of this class while devices shall be accessed. To find out how to gain access to a certain device look at the detailed description of this class.
Once a pointer to the desired device represented by an instance of the class novitec.CameraAPI.CLR.Handle has been obtained every other device related properties or functions can be accessed.
Some source code samples how to locate a certain novitec.CameraAPI.CLR.Handle also can be found in the detailed description of the class novitec.CameraAPI.CLR.DeviceManager.
To capture images, novitec.CameraAPI.CLR.Camera instance must be created. This class can be constructed by passing a pointer to the novitec.CameraAPI.CLR.Handle object obtained from the novitec.CameraAPI.CLR.↩ Devicemanager to the class constructor.
The novitec.CameraAPI.CLR.Camera class provides basically control function for image acquisition. Getting the first image might e.g. look like that:

```
using novitec.CameraAPI.CLR;
using System;
namespace Examples
{
    class Program
    {
        static void Main()
        {
            DeviceManager manager = new DeviceManager();
            Handle deviceHandle = new Handle();
            Camera camera = new Camera();
            Error err = new Error();
            uint numCameras;
            CheckError(manager.Update()); // Update device list. If this function is called, it will search
      devices on whole inferfaces.
            CheckError(manager.GetCameraHandle(4, deviceHandle)); // Get index 0 camera handle from device
      manager.
            CheckError(camera.Connect(deviceHandle)); // Connect Camera
            CheckError(camera.Start()); // Start image grab process.
            int grabbedImages = 0;
            for (int i = 0; i < 100; i++)
            {
                Image image = new Image();
                err = camera.GetImage(image, 1000);
                if (err.GetType() != ErrorType.NVT_OK)
```

```
                {
                    Console.WriteLine($"Error - {err.GetDescription()}");
                    continue;
                }
                byte[] data = image.GetData();
                Console.WriteLine($"image {grabbedImages++} | size = {image.GetDataSize()}, " +
                    $"data = {data[0].ToString("X")} {data[1].ToString("X")} {data[2].ToString("X")}
        {data[3].ToString("X")} ...");
            }
            CheckError(camera.Stop()); // Stop the image grab process.
            CheckError(camera.Disconnect()); // Disconnect camera.
        }
        static void CheckError(Error err)
        {
            if (err.GetType() != ErrorType.NVT_OK)
            {
                System.Diagnostics.StackFrame stackFrame = new System.Diagnostics.StackFrame(true);
                Console.WriteLine($"[Error] {err.GetDescription()}");
                Console.WriteLine("Press any key to exit...");
                Console.ReadKey();
                Environment.Exit(-1);
            }
        }
    }
}
```

This sample contains everything the user needs to do to capture one image including all initialization work and error handling for every source of error one can think of. Several example codes will provide an even better understanding of the interface.

1.3 Feature control

To control camera features, novitec.CameraAPI.CLR.GenICam.CameraInterface instance must be created. This class can be constructed from pointer of the novitec.CameraAPI.CLR.Camera object.
API follows GenICam standard, and functions are controlled by descriptor in XML format defined as GenAPI.
When creating a novitec.CameraAPI.CLR.GenICam.CameraInterface instance, the XML descriptor is loaded from the device and create a feature tree. User can search with the provided functions.
To control each features,
It can be created by specifying the name and type of the feature by novitec.CameraAPI.CLR.GenICam.CameraInterface.GetFeature() function.

```
using novitec.CameraAPI.CLR;
using novitec.CameraAPI.CLR.GenICam;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace featureControl
{
    class Program
    {
        static void Main(string[] args)
        {
            DeviceManager manager = new DeviceManager();
            Handle deviceHandle = new Handle();
            Camera camera = new Camera();
            Error err;
            uint numCameras;
            err = manager.Update(); // Update device list. If this function is called, it will search on
       whole interfaces.
            CheckError(err);
            err = manager.GetNumberOfCameras(out numCameras); // Get number of cameras.
            CheckError(err);
            for (uint i = 0; i < numCameras; i++)
            {
                CameraInfo info = new CameraInfo();
                manager.GetCameraInfo(i, info);
                Console.WriteLine($"{i} | {info.GetModelName()} - {info.GetSerialNumber()}");
            }
            Console.Write("Type camera index > ");
            int camIdx = -1;
            camIdx = Convert.ToInt32(Console.ReadLine());
            err = manager.GetCameraHandle((uint)camIdx, deviceHandle); // Get camera handle from device
       manager.
```

```
        CheckError(err);
        //err = manager.GetCameraHandleBySerial("ESB20049", deviceHandle); // Get camera handle by
    serial number.
        err = camera.Connect(deviceHandle); //connect camera
        CheckError(err);
        /*
         ******************
         * Feature Control *
         ******************
         */
        // 0.Get camera interface
        CameraInterface camIf = new CameraInterface(camera);
        // 1. Basic instruction
        {
            // - exposure time read
            ICategory iCate2 = camIf.GetFeature<ICategory>("DeviceControl");
            IInteger exposureTime = camIf.GetFeature<IInteger>("ExposureTime");
            int exposureTimeVal = (int)exposureTime.GetValue();
            Console.WriteLine($"current exposure time = {exposureTimeVal} us");
            // - exposure time write
            exposureTime.SetValue(5000);
            string unit = exposureTime.GetUnit();
            Console.WriteLine($"unit = {unit}");
        }
        // 2. Feature finding
        {
            // - get available features, refer to printFeatureTree() function
            Console.WriteLine("Camera feature list - ");
            printFeatureTree(camIf, IntPtr.Zero, 0);
        }
        // 3. Get feature information
        {
            // - description
            IInteger exposureTime = camIf.GetFeature<IInteger>("ExposureTime");
            Console.Write($"\n\n" +
                $"feature name = {exposureTime.GetFeatureName()}\n" +
                $"display name = {exposureTime.GetDisplayName()}\n" +
                $"description = {exposureTime.GetDescription()}\n");
            // - accessibility
            var accessMode = exposureTime.GetAccessMode(); //ReadWrite, ReadOnly or WriteOnly
            // - range
            if (exposureTime.HasMin() && exposureTime.HasMax())
            {
                double min = exposureTime.GetMin();
                double max = exposureTime.GetMax();
                Console.WriteLine($"value rate = {min} - {max}\n\n");
            }
        }
        // 4. Exception processing
        {
            // - type mismatching
            try
            {
                IFloat exposureTime = camIf.GetFeature<IFloat>("ExposureTime");
            }
            catch (novitec.CameraAPI.CLR.NovitecException ex)
            {
                Console.WriteLine(ex.Message);
            }
        }
        // 5. Feature Types
        {
            try
            {
                // - IInteger
                IInteger triggerDebounceActive = camIf.GetFeature<IInteger>("TriggerDebounceActive");
                triggerDebounceActive.SetValue(10);
                int triggerDebounceActiveValue = (int)triggerDebounceActive.GetValue();
                // - IBoolean
                IBoolean lineInverter = camIf.GetFeature<IBoolean>("LineInverter");
                lineInverter.SetValue(true);
                bool lineInverterValue = lineInverter.GetValue();
                // - IFloat
                IFloat gain = camIf.GetFeature<IFloat>("Gain");
                gain.SetValue(1.0);
                double gainValue = gain.GetValue();
                // - IString
                IString serialNumber = camIf.GetFeature<IString>("DeviceSerialNumber");
                string serialNumberValue = serialNumber.GetValue();
                // - IEnumeration
                IEnumeration filterSwitcher = camIf.GetFeature<IEnumeration>("FilterSwitcher");
                string filterSwitcherValue = filterSwitcher.GetSymbolicValue();
                int numEntries = filterSwitcher.GetNumberOfEntries();
                for (int i = 0; i < numEntries; i++)
                {
                    var entry = filterSwitcher.GetEntry(i);
                    string symbolic = entry.GetFeatureName();
```

```
                    int value = (int)entry.GetValue();
                    Console.WriteLine($"entry {value} - {symbolic}");
                }
                filterSwitcher.SetSymbolicValue("IRCutFilter");
                filterSwitcher.SetIntValue(0);
                // - ICommand
                ICommand triggerSoftware = camIf.GetFeature<ICommand>("TriggerSoftware");
                triggerSoftware.Execute();
            }
            catch (novitec.CameraAPI.CLR.NovitecException ex)
            {
                Console.WriteLine(ex.Message);
            }
        }
        err = camera.Disconnect();
        CheckError(err);
        Console.WriteLine("\nPress any key to exit...");
        Console.ReadKey();
    }
    static void CheckError(Error err)
    {
        if (err.GetType() != ErrorType.NVT_OK)
        {
            System.Diagnostics.StackFrame stackFrame = new System.Diagnostics.StackFrame(true);
            Console.WriteLine($"[Error] {err.GetDescription()}");
            Console.WriteLine("Press any key to exit...");
            Console.ReadKey();
            Environment.Exit(-1);
        }
    }
    static void printFeatureTree(CameraInterface camIf, IntPtr handle, int level)
    {
        IntPtr _hs = camIf.GetFirstChildFeature(handle);
        while (_hs != IntPtr.Zero)
        {
            for (int i = 0; i < level; i++)
                Console.Write("  ");
            if (level > 0)
                Console.Write("- ");
            Console.WriteLine($"{camIf.GetFeatureName(_hs)} [{camIf.GetFeatureType(_hs)}]");
            printFeatureTree(camIf, _hs, level + 1);
            _hs = camIf.GetNextSiblingFeature(_hs);
        }
    }
  }
}
```

1.3.1 Feature error handling

When handling features, different errors can occur: feature read/write fail, feature handles with incorrect type or no such feature exists. When an error occurs, API will throw an novitec.CameraAPI.CLR.NovitecException. You should handle exceptions by try, catch keyword.

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 4

File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 5

Class Documentation

5.1 novitec::CameraAPI::CLR::Camera Class Reference

The class which provides the camera control.

`#include <Camera.h>`

Public Member Functions

- **Camera** ()

  Initialize a new instance of the Camera class.
- ~**Camera** ()

  Destructor
- novitec::CameraAPI::Camera * GetCCamera ()

  Get the pointer to the instance of native(C++) Camera class.
- Error ^ Connect (Handle^ pHandle)

  Connect to a camera.
  By default, AccessMode is `AM_CONTROL`
- Error ^ Connect (Handle^ pHandle, AccessMode accessMode)

  Connect to a camera.
- Error ^ Disconnect ()

  Disconnect a camera.
- bool IsConnected ()

  Get the connection status to a camera.
- Error ^ Start ()

  Start grabbing images from a camera.
- Error ^ Stop ()

  Stop grabbing images from a camera.
- Error ^ GetImage ([System::Runtime::InteropServices::Out]Image^ pImage)

  Get an image from the frame buffer. Default timeout is 500ms.
- Error ^ GetImage ([System::Runtime::InteropServices::Out]Image^ pImage, unsigned int uiTimeout)

  Get an image from the frame buffer.
- Error ^ SetImageCallback (ImageCallbackHandler^ handler, IntPtr^ param)

  Set an image callback function.
  ***[WARNING]** It must be called before Camera::Start().*

- System::String ^ GetXML ()
  Get XML register descriptor. Default value of `forceUpdate` is `false`.
- System::String ^ GetXML (bool forceUpdate)
  Get XML register descriptor.
- Error ^ ReadRegister (unsigned int address, [System::Runtime::InteropServices::Out]unsigned int% value)
  Read register from device.
- Error ^ WriteRegister (unsigned int address, unsigned int value)
  Write register to device.
- Error ^ ReadMemory (unsigned int address, [System::Runtime::InteropServices::Out]array< Byte >^% value, unsigned int length)
  Read memory from device.
- Error ^ WriteMemory (unsigned int address, array< Byte >^ value, unsigned int length)
  Write memory to device.
- Error ^ SetDeviceEventCallback (DeviceEventCallbackHandler^ handler, IntPtr^ param)
  Set device event callback function.
- Error ^ SetBufferMode (BufferMode bufferMode, int numBuffers)
  Set frame buffer mode.
- Error ^ GetBufferMode ([System::Runtime::InteropServices::Out]BufferMode% bufferMode, [System::Runtime::InteropServices::Out]int% numBuffers)
  Get frame buffer mode.
- Error ^ UpdateFirmware (array< Byte >^ data, int length)
  Update firmware.

Protected Member Functions

- void **SetImageCallback_impl** (novitec::CameraAPI::Image *pImage, void *pVal)
- void **SetDeviceEventCallback_impl** (novitec::CameraAPI::SeverityLevel, novitec::CameraAPI::DeviceEvent, char *, void *)

Protected Attributes

- novitec::CameraAPI::Camera * **m_Camera**
- ImageCallbackHandler ^ **m_imageHandler**
- DeviceEventCallbackHandler ^ **m_deviceEventHandler**
- ImageCallbackHandler_impl ^ **handler_impl**
- IntPtr **pHandle_impl**
- novitec::CameraAPI::ImageCallbackFunc **pFunc_impl**

5.1.1 Detailed Description

The class which provides the camera control.

5.1.2 Member Function Documentation

5.1.2.1 Connect() [1/2]

```
Error ^ novitec::CameraAPI::CLR::Camera::Connect (
            Handle^ pHandle )
```

Connect to a camera.
By default, AccessMode is `AM_CONTROL`

Parameters

pHandle	The pointer to the camera handle.

[See Also] DeviceManager::GetCameraHandle()

Returns

novitec::CameraAPI::CLR::Error

5.1.2.2 Connect() [2/2]

```
Error ^ novitec::CameraAPI::CLR::Camera::Connect (
            Handle^ pHandle,
            AccessMode accessMode )
```

Connect to a camera.

Parameters

pHandle	The pointer to the camera handle.
accessMode	AcessMode

[See Also] DeviceManager::GetCameraHandle()

Returns

novitec::CameraAPI::CLR::Error

5.1.2.3 Disconnect()

```
Error ^ novitec::CameraAPI::CLR::Camera::Disconnect ( )
```

Disconnect a camera.

Returns

novitec::CameraAPI::CLR::Error

5.1.2.4 GetBufferMode()

```
Error ^ novitec::CameraAPI::CLR::Camera::GetBufferMode (
            [System::Runtime::InteropServices::Out] BufferMode% bufferMode,
            [System::Runtime::InteropServices::Out] int% numBuffers )
```

Get frame buffer mode.

Parameters

bufferMode	Buffer mode.
numBuffers	Number of frame buffers.

Returns

novitec::CameraAPI::CLR::Error

5.1.2.5 GetCCamera()

```
novitec::CameraAPI::Camera * novitec::CameraAPI::CLR::Camera::GetCCamera ( )
```

Get the pointer to the instance of native(C++) Camera class.

Returns

The pointer to the instance of native(C++) Camera class.

5.1.2.6 GetImage() [1/2]

```
Error ^ novitec::CameraAPI::CLR::Camera::GetImage (
            [System::Runtime::InteropServices::Out] Image^ pImage )
```

Get an image from the frame buffer. Default timeout is 500ms.

Parameters

pImage	[out] Image buffer to store the image.

Returns

novitec::CameraAPI::CLR::Error

5.1.2.7 GetImage() [2/2]

```
Error ^ novitec::CameraAPI::CLR::Camera::GetImage (
            [System::Runtime::InteropServices::Out] Image^ pImage,
            unsigned int uiTimeout )
```

Get an image from the frame buffer.

Parameters

pImage	[out] Image buffer to store the image.
uiTimeout	[in] Timeout (ms)

Returns

novitec::CameraAPI::CLR::Error

5.1.2.8 GetXML() [1/2]

```
System::String ^ novitec::CameraAPI::CLR::Camera::GetXML ( )
```

Get XML register descriptor. Default value of `forceUpdate` is `false`.

Returns

A string that contains XML register descriptor.

5.1.2.9 GetXML() [2/2]

```
System::String ^ novitec::CameraAPI::CLR::Camera::GetXML (
            bool forceUpdate )
```

Get XML register descriptor.

Parameters

forceUpdate	`true` to set force update; otherwise, `false`

Returns

A string that contains XML register descriptor.

5.1.2.10 IsConnected()

```
bool novitec::CameraAPI::CLR::Camera::IsConnected ( )
```

Get the connection status to a camera.

Returns

`true` if camera is connected; otherwise, `false`.

5.1.2.11 ReadMemory()

```
Error ^ novitec::CameraAPI::CLR::Camera::ReadMemory (
            unsigned int address,
            [System::Runtime::InteropServices::Out] array< Byte >^% value,
            unsigned int length )
```

Read memory from device.

Parameters

address	[in] register address.
value	[out] register value.
length	memory length.

Returns

novitec::CameraAPI::CLR::Error

5.1.2.12 ReadRegister()

```
Error ^ novitec::CameraAPI::CLR::Camera::ReadRegister (
            unsigned int address,
            [System::Runtime::InteropServices::Out] unsigned int% value )
```

Read register from device.

Parameters

address	[in] register address.
value	[out] register value.

Returns

novitec::CameraAPI::CLR::Error

5.1.2.13 SetBufferMode()

```
Error ^ novitec::CameraAPI::CLR::Camera::SetBufferMode (
            BufferMode bufferMode,
            int numBuffers )
```

Set frame buffer mode.

Parameters

bufferMode	Buffer mode.
numBuffers	Number of frame buffers.

Returns

novitec::CameraAPI::CLR::Error

5.1.2.14 SetDeviceEventCallback()

```
Error ^ novitec::CameraAPI::CLR::Camera::SetDeviceEventCallback (
            DeviceEventCallbackHandler^ handler,
            IntPtr^ param )
```

Set device event callback function.

Parameters

handler	Pointer to a handler which is called when device is disconnected, updated, etc.
param	Parameter

Returns

novitec::CameraAPI::CLR::Error

5.1.2.15 SetImageCallback()

```
Error ^ novitec::CameraAPI::CLR::Camera::SetImageCallback (
            ImageCallbackHandler^ handler,
            IntPtr^ param )
```

Set an image callback function.
[WARNING] It must be called before Camera::Start().

Parameters

handler	Pointer to a handler which is called by internal acquisition thread when new image is acquired.
param	Parameter

Returns

novitec::CameraAPI::CLR::Error

5.1.2.16 Start()

```
Error ^ novitec::CameraAPI::CLR::Camera::Start ( )
```

Start grabbing images from a camera.

Returns

novitec::CameraAPI::CLR::Error

5.1.2.17 Stop()

```
Error ^ novitec::CameraAPI::CLR::Camera::Stop ( )
```

Stop grabbing images from a camera.

Returns

novitec::CameraAPI::CLR::Error

5.1.2.18 UpdateFirmware()

```
Error ^ novitec::CameraAPI::CLR::Camera::UpdateFirmware (
            array< Byte >^ data,
            int length )
```

Update firmware.

Parameters

data	Binary firmware data.
length	Firmware data length.

Returns

novitec::CameraAPI::CLR::Error

5.1.2.19 WriteMemory()

```
Error ^ novitec::CameraAPI::CLR::Camera::WriteMemory (
            unsigned int address,
```

```
array< Byte >^ value,
unsigned int length )
```

Write memory to device.

Parameters

address	register address.
value	register value.
length	memory length.

Returns

novitec::CameraAPI::CLR::Error

5.1.2.20 WriteRegister()

```
Error ^ novitec::CameraAPI::CLR::Camera::WriteRegister (
            unsigned int address,
            unsigned int value )
```

Write register to device.

Parameters

address	register address.
value	register value.

Returns

novitec::CameraAPI::CLR::Error

The documentation for this class was generated from the following file:

- Camera.h

5.2 novitec::CameraAPI::CLR::CameraInfo Class Reference

GenICam Compliant Information

```
#include <Types.h>
```

Public Member Functions

- CameraInfo ()

  Initialize a new instance of the CameraInfo class.
- novitec::CameraAPI::CameraInfo * GetCCameraInfo ()

  Get pointer to the native(C++) CameraInfo class.
- String ^ GetID ()

Get unique ID of the device.

- String ^ GetManufacturer ()

  Get device vendor name.

- String ^ GetModelName ()

  Get device model name.

- String ^ GetTransportLayer ()

  Get transport layer technology that is suported.

- String ^ GetDisplayName ()

  Get display name for the device (including a unique ID).

- int GetAccessStatus ()

  Gets the access mode the GenTL Producer has on the opened device. (DEVICE_ACCESS_STATUS enumeration value).

- String ^ GetUserDefinedName ()

  Get user defined name. (GenTL v1.4)

- String ^ GetSerialNumber ()

  Get the device's serial number. (GenTL v1.4)

- String ^ GetDeviceVersion ()

  Get the device version. (GenTL v1.4)

- unsigned long long int GetTimestampFrequency ()

  Get tick-frequency of the timestamp clock. (GenTL v1.4)

Protected Attributes

- novitec::CameraAPI::CameraInfo * **m_CameraInfo**

5.2.1 Detailed Description

GenICam Compliant Information

5.2.2 Constructor & Destructor Documentation

5.2.2.1 CameraInfo()

```
CameraInfo::CameraInfo ( )
```

Initialize a new instance of the CameraInfo class.

summary> Destructor

5.2.3 Member Function Documentation

5.2.3.1 GetAccessStatus()

```
int CameraInfo::GetAccessStatus ( )
```

Gets the access mode the GenTL Producer has on the opened device. (DEVICE_ACCESS_STATUS enumeration value).

Returns

A `integer` that represents access mode the GenTL Producer has on the opened device.

5.2.3.2 GetCCameraInfo()

```
novitec::CameraAPI::CameraInfo * CameraInfo::GetCCameraInfo ( )
```

Get pointer to the native(C++) CameraInfo class.

Returns

Pointer to the native(C++) CameraInfo class.

5.2.3.3 GetDeviceVersion()

```
String CameraInfo::GetDeviceVersion ( )
```

Get the device version. (GenTL v1.4)

Returns

A `string` that represents the device version. (GenTL v1.4)

5.2.3.4 GetDisplayName()

```
String CameraInfo::GetDisplayName ( )
```

Get display name for the device (including a unique ID).

Returns

A `string` that represents display name for the device (including a unique ID).

5.2.3.5 GetID()

```
String CameraInfo::GetID ( )
```

Get unique ID of the device.

Returns

A `string` that repressents unique ID of the device.

5.2.3.6 GetManufacturer()

```
String CameraInfo::GetManufacturer ( )
```

Get device vendor name.

Returns

A `string` that represents device vendor name.

5.2.3.7 GetModelName()

```
String CameraInfo::GetModelName ( )
```

Get device model name.

Returns

A `string` that represents device model name.

5.2.3.8 GetSerialNumber()

```
String CameraInfo::GetSerialNumber ( )
```

Get the device's serial number. (GenTL v1.4)

Returns

A string that represents the device's serial number. (GenTL v1.4)

5.2.3.9 GetTimestampFrequency()

```
unsigned long long int CameraInfo::GetTimestampFrequency ( )
```

Get tick-frequency of the timestamp clock. (GenTL v1.4)

Returns

A `long long` that represents the tick-frequency of the timestamp clock. (GenTL v1.4)

5.2.3.10 GetTransportLayer()

```
String CameraInfo::GetTransportLayer ( )
```

Get transport layer technology that is suported.

Returns

A `string` that represents transport layer technology that is supported.

5.2.3.11 GetUserDefinedName()

```
String CameraInfo::GetUserDefinedName ( )
```

Get user defined name. (GenTL v1.4)

Returns

A `string` that represents user defined name. (GenTL v1.4)

The documentation for this class was generated from the following files:

- Types.h
- Types.cpp

5.3 novitec::CameraAPI::CLR::GenICam::CameraInterface Class Reference

The class that contains information about CameraInterface.

```
#include <CameraInterface.h>
```

Public Member Functions

- **CameraInterface** ()

  Initialize a new instance of the CameraInterface class.
- CameraInterface (novitec::CameraAPI::CLR::Camera^ device)

  Initialize a new instance of the CameraInterface class with pointer of camera.
- ~**CameraInterface** ()

  Destructor
- CameraInterface ^ **operator=** (const CameraInterface^&rhs)
- int Open (System::String^ XMLFilePath)

  Open camera interface from an XML file.
  ***[WARNING]** When using the function CameraInterface(novitec::CameraAPI::BaseDevice* device) to create CaermaInterface, the XML file is automatically loaded so calling Open() is not necessary.*
- int OpenFromMem (System::String^ xml)

  Open camera interface from an XML on the memory.
  ***[WARNING]** When using the function CameraInterface(novitec::CameraAPI::BaseDevice* device) to create CaermaInterface, the XML file is automatically loaded so calling Open() is not necessary.*
- int **Close** ()

  Close the camera interface.
- FeatureHandle GetFirstChildFeature ()

  Get first child feature handle.
  If input argument handle is 0(FEATURE_HANDLE_ROOT), it return first category.
- FeatureHandle GetFirstChildFeature (FeatureHandle handle)

  If input argument handle is 0(FEATURE_HANDLE_ROOT), it return first category.
- FeatureHandle GetNextSiblingFeature (FeatureHandle handle)

  Get next sibling feature handle.
- System::String ^ GetFeatureName (FeatureHandle handle)

  Get the name of the specified feature.
- System::String ^ GetFeatureType (FeatureHandle handle)

  Get the interface type of the specified feature.
- System::String ^ GetFeatureType (System::String^ featureName)

  Get the interface type of the specified feature by feature name.
- template<typename TFeature >
  TFeature GetFeature (FeatureHandle handle)

  Get a feature by handle.
- template<typename TFeature >
  TFeature GetFeature (System::String^ featureName)

  Get a feature by feature name.
  ***[Exception]** novitec::CameraAPI::CLR::NovitecException*
- unsigned int GetNumberOfInvalidator (System::String^ featureName)

  Get number of the invalidators for update linked features.
  ***[Exception]** novitec::CameraAPI::CLR::NovitecException*
- System::String ^ GetInvalidator (System::String^ featureName, int idx)

  Get invalidator for update linked features.
  ***[Exception]** novitec::CameraAPI::CLR::NovitecException*
- unsigned int GetNumberOfLocker (System::String^ featureName)

  Get number of the lockers for update linked features.
- System::String ^ GetLocker (System::String^ featureName, int idx)

  Get locker for update linked features.

Protected Attributes

- novitec::CameraAPI::GenICam::CameraInterface * **m_CameraInterface**

5.3.1 Detailed Description

The class that contains information about CameraInterface.

5.3.2 Constructor & Destructor Documentation

5.3.2.1 CameraInterface()

```
CameraInterface::CameraInterface (
            novitec::CameraAPI::CLR::Camera^ device )
```

Initialize a new instance of the CameraInterface class with pointer of camera.

Parameters

device	A pointer of Camera

5.3.3 Member Function Documentation

5.3.3.1 GetFeature() [1/2]

```
template<typename TFeature >
TFeature CameraInterface::GetFeature (
            FeatureHandle handle )
```

Get a feature by handle.

Template Parameters

TFeature	Specified type.

Parameters

handle	FeatureHandle

Returns

Feature of the specified type.

5.3.3.2 GetFeature() [2/2]

```
template<typename TFeature >
TFeature CameraInterface::GetFeature (
            System::String^ featureName )
```

Get a feature by feature name.
[Exception] novitec::CameraAPI::CLR::NovitecException

Template Parameters

TFeature	Specified type.

Parameters

featureName	Feature Name

Returns

Feature of the specified type.

5.3.3.3 GetFeatureName()

```
System::String CameraInterface::GetFeatureName (
            FeatureHandle handle )
```

Get the name of the specified feature.

Parameters

handle	FeatureHandle

Returns

The name of the specified feature.

5.3.3.4 GetFeatureType() [1/2]

```
System::String CameraInterface::GetFeatureType (
            FeatureHandle handle )
```

Get the interface type of the specified feature.

Parameters

handle	FeatureHandle

[Exception] novitec::CameraAPI::CLR::NovitecException

Returns

The interface type of the specified feature.

5.3.3.5 GetFeatureType() [2/2]

```
System::String CameraInterface::GetFeatureType (
            System::String^ featureName )
```

Get the interface type of the specified feature by feature name.

Parameters

featureName	FeatureHandle

[Exception] novitec::CameraAPI::CLR::NovitecException

Returns

The interface type of the specified feature.

5.3.3.6 GetFirstChildFeature() [1/2]

```
FeatureHandle CameraInterface::GetFirstChildFeature ( )
```

Get first child feature handle.
If input argument handle is 0(FEATURE_HANDLE_ROOT), it return first category.

Returns

A FeatureHandle if had child feature; a null pointer or reference otherwise.

5.3.3.7 GetFirstChildFeature() [2/2]

```
FeatureHandle CameraInterface::GetFirstChildFeature (
            FeatureHandle handle )
```

If input argument handle is 0(FEATURE_HANDLE_ROOT), it return first category.

Parameters

handle	Parent feature handle.

Returns

A FeatureHandle if had child feature; a null pointer or reference otherwise.

5.3.3.8 GetInvalidator()

```
System::String CameraInterface::GetInvalidator (
            System::String^ featureName,
            int idx )
```

Get invalidator for update linked features.
[Exception] novitec::CameraAPI::CLR::NovitecException

Parameters

featureName	Feature Name
idx	Index of invalidators.

Returns

Invalidator for update linked features

5.3.3.9 GetLocker()

```
System::String CameraInterface::GetLocker (
            System::String^ featureName,
            int idx )
```

Get locker for update linked features.

Parameters

featureName	Feature Name
idx	Index of lockers.

Returns

Locker for update linked features

5.3.3.10 GetNextSiblingFeature()

```
FeatureHandle CameraInterface::GetNextSiblingFeature (
            FeatureHandle handle )
```

Get next sibling feature handle.

Parameters

handle	FeatureHandle

Returns

FeatureHandle

5.3.3.11 GetNumberOfInvalidator()

```
unsigned int CameraInterface::GetNumberOfInvalidator (
            System::String^ featureName )
```

Get number of the invalidators for update linked features.
[Exception] novitec::CameraAPI::CLR::NovitecException

Parameters

featureName	Feature Name

Returns

Number of the invalidators for update linked features.

5.3.3.12 GetNumberOfLocker()

```
unsigned int CameraInterface::GetNumberOfLocker (
            System::String^ featureName )
```

Get number of the lockers for update linked features.

Parameters

featureName	Feature Name

Returns

Number of the invalidators for update linked features.

5.3.3.13 Open()

```
int CameraInterface::Open (
            System::String^ XMLFilePath )
```

Open camera interface from an XML file.
[WARNING] When using the function CameraInterface(novitec::CameraAPI::BaseDevice* device) to create CaermaInterface, the XML file is automatically loaded so calling Open() is not necessary.

Parameters

XMLFilePath	XML File Path

5.3.3.14 OpenFromMem()

```
int CameraInterface::OpenFromMem (
            System::String^ xml )
```

Open camera interface from an XML on the memory.
[WARNING] When using the function CameraInterface(novitec::CameraAPI::BaseDevice* device) to create CaermaInterface, the XML file is automatically loaded so calling Open() is not necessary.

Parameters

xml	XML String

The documentation for this class was generated from the following files:

- GenICam/CameraInterface.h
- GenICam/CameraInterface.cpp

5.4 novitec::CameraAPI::CLR::DeviceBase Class Reference

The class which provides the device control.

```
#include <DeviceBase.h>
```

Public Member Functions

- **DeviceBase** ()

  Initialize a new instance of the DeviceBase class.
- ~**DeviceBase** ()

  Destructor.
- virtual Error ^ Connect (Handle^ pHandle, AccessMode accessMode)=0

  Connect to a camera.
- virtual Error ^ **Disconnect** ()=0

  Disconnect a camera.
- virtual bool IsConnected ()=0

  Get the connection status to a camera.
- virtual Error ^ Start ()=0

  Start grabbing images from a camera.
- virtual Error ^ Stop ()=0

  Stop grabbing images from a camera.
- virtual Error ^ GetImage ([System::Runtime::InteropServices::Out]Image^ pImage, unsigned int ui↩ Timeout)=0

  Get an image from the frame buffer.
- virtual Error ^ SetImageCallback (ImageCallbackHandler^ func, void *param)=0

  Set an image callback function.
  ***[WARNING]** It must be called before Camera::Start().*
- InterfaceType GetInterfaceType ()

  Get interface type.

Protected Attributes

- InterfaceType **m_interfaceType**

5.4.1 Detailed Description

The class which provides the device control.

5.4.2 Member Function Documentation

5.4.2.1 Connect()

```
virtual Error ^ novitec::CameraAPI::CLR::DeviceBase::Connect (
            Handle^ pHandle,
            AccessMode accessMode )  [pure virtual]
```

Connect to a camera.

Parameters

pHandle	The pointer to the camera handle.
accessMode	AccessMode

Returns

novitec.CameraAPI.CLR.Error

5.4.2.2 GetImage()

```
virtual Error ^ novitec::CameraAPI::CLR::DeviceBase::GetImage (
            [System::Runtime::InteropServices::Out] Image^ pImage,
            unsigned int uiTimeout )  [pure virtual]
```

Get an image from the frame buffer.

Parameters

pImage	[out] Image buffer to store the image.
uiTimeout	[in] Timeout (ms)

Returns

novitec::CameraAPI::CLR::Error

5.4.2.3 GetInterfaceType()

```
InterfaceType DeviceBase::GetInterfaceType ( )
```

Get interface type.

Returns

5.4.2.4 IsConnected()

```
virtual bool novitec::CameraAPI::CLR::DeviceBase::IsConnected ( )  [pure virtual]
```

Get the connection status to a camera.

Returns

`true` if camera is connected; otherwise, `false`.

5.4.2.5 SetImageCallback()

```
virtual Error ^ novitec::CameraAPI::CLR::DeviceBase::SetImageCallback (
            ImageCallbackHandler^ func,
            void * param )  [pure virtual]
```

Set an image callback function.
[WARNING] It must be called before Camera::Start().

Returns

novitec::CameraAPI::CLR::Error

5.4.2.6 Start()

```
virtual Error ^ novitec::CameraAPI::CLR::DeviceBase::Start ( )  [pure virtual]
```

Start grabbing images from a camera.

Returns

novitec::CameraAPI::CLR::Error

5.4.2.7 Stop()

```
virtual Error ^ novitec::CameraAPI::CLR::DeviceBase::Stop ( )  [pure virtual]
```

Stop grabbing images from a camera.

Returns

novitec::CameraAPI::CLR::Error

The documentation for this class was generated from the following files:

- DeviceBase.h
- DeviceBase.cpp

5.5 novitec::CameraAPI::CLR::DeviceManager Class Reference

The class which provides the device management.

```
#include <DeviceManager.h>
```

Public Member Functions

- **DeviceManager** (void)

  Initialize a new instance of the DeviceManager class.
- ~**DeviceManager** (void)

  Destructor
- Error ^ Update ()

  Updates the internal list of available devices.
- Error ^ GetNumberOfCameras ([System::Runtime::InteropServices::Out]unsigned int% pNumCameras)

  Get the number of available cameras.
- Error ^ GetCameraHandle (unsigned int uiIndex, [System::Runtime::InteropServices::OutAttribute]Handle^ pHandle)

  Get the camera handle from a given index.
- Error ^ GetCameraHandleBySerial (System::String^ serialNumber, [System::Runtime::InteropServices::↩ OutAttribute]Handle^ pHandle)

  Get the camera handle from a given serial number.
- Error ^ GetCameraHandleByIPAddress (System::String^ ipAddress, [System::Runtime::InteropServices::↩ OutAttribute]Handle^ pHandle)

  Get the camera handle from a given IP address.
- Error ^ GetCameraInfo (unsigned int uiIndex, [System::Runtime::InteropServices::Out]CameraInfo^ p↩ CameraInfo)

  Get the camera information.
- Error ^ GetCameraInfo (unsigned int uiIndex, [System::Runtime::InteropServices::Out]GEVCameraInfo^ p↩ CameraInfo, [System::Runtime::InteropServices::Optional][System::Runtime::InteropServices::Out]NetworkAdapterInfo^ pAdapterInfo)

  Get the GigE Vision Camera information.
- Error ^ GetCameraInfo (unsigned int uiIndex, [System::Runtime::InteropServices::Out]U3VCameraInfo^ pU3VCameraInfo, [System::Runtime::InteropServices::Optional][System::Runtime::InteropServices::↩ Out]HostControllerInfo^ pHostControllerInfo)

  Get the USB3 Vision Camera information.
- Error ^ GetCameraInfo (unsigned int uiIndex, [System::Runtime::InteropServices::Out]NU3CameraInfo^ pU3VCameraInfo, [System::Runtime::InteropServices::Optional][System::Runtime::InteropServices::↩ Out]HostControllerInfo^ pHostControllerInfo)

  Get the Novitec USB 3.0 Camera information.
- Error ^ SendForceIP (unsigned int uiIndex, [System::Runtime::InteropServices::Optional]String^ ip, [System::Runtime::InteropServices::Optional]String^ subnetMask, [System::Runtime::InteropServices::↩ Optional]String^ gateway)

  Send Force IP Command. (IP restoration)

Protected Attributes

- novitec::CameraAPI::DeviceManager * **m_deviceManager**

5.5.1 Detailed Description

The class which provides the device management.

5.5.2 Member Function Documentation

5.5.2.1 GetCameraHandle()

```
Error novitec::CameraAPI::CLR::DeviceManager::GetCameraHandle (
            unsigned int uiIndex,
            [System::Runtime::InteropServices::OutAttribute] Handle^ pHandle )
```

Get the camera handle from a given index.

Parameters

uiIndex	[in] Index of camera
pHandle	[out] Handle of camera

Returns

novitec::CameraAPI::CLR::Error

5.5.2.2 GetCameraHandleByIPAddress()

```
Error novitec::CameraAPI::CLR::DeviceManager::GetCameraHandleByIPAddress (
            System::String^ ipAddress,
            [System::Runtime::InteropServices::OutAttribute] Handle^ pHandle )
```

Get the camera handle from a given IP address.

Parameters

ipAddress	[in] IP address
pHandle	[out] Handle of camera

Returns

novitec::CameraAPI::CLR::Error

5.5.2.3 GetCameraHandleBySerial()

```
Error novitec::CameraAPI::CLR::DeviceManager::GetCameraHandleBySerial (
            System::String^ serialNumber,
            [System::Runtime::InteropServices::OutAttribute] Handle^ pHandle )
```

Get the camera handle from a given serial number.

Parameters

serialNumber	[in] Serial Number
pHandle	[out] Handle of camera

Returns

novitec::CameraAPI::CLR::Error

5.5.2.4 GetCameraInfo() [1/4]

```
Error novitec::CameraAPI::CLR::DeviceManager::GetCameraInfo (
            unsigned int uiIndex,
            [System::Runtime::InteropServices::Out] CameraInfo^ pCameraInfo )
```

Get the camera information.

Parameters

uiIndex	[in] Index of camera
pCameraInfo	[out] CameraInfo of camera

Returns

5.5.2.5 GetCameraInfo() [2/4]

```
Error novitec::CameraAPI::CLR::DeviceManager::GetCameraInfo (
            unsigned int uiIndex,
            [System::Runtime::InteropServices::Out] GEVCameraInfo^ pCameraInfo,
            [System::Runtime::InteropServices::Out] NetworkAdapterInfo^ pAdapterInfo )
```

Get the GigE Vision Camera information.

Parameters

uiIndex	[in] Index of camera
pCameraInfo	[out] GEVCameraInfo of camera
pAdapterInfo	[out] NetworkAdapterInfo

Returns

5.5.2.6 GetCameraInfo() [3/4]

```
Error novitec::CameraAPI::CLR::DeviceManager::GetCameraInfo (
            unsigned int uiIndex,
```

```
            [System::Runtime::InteropServices::Out] NU3CameraInfo^ pU3VCameraInfo,
            [System::Runtime::InteropServices::Out] HostControllerInfo^ pHostControllerInfo )
```

Get the Novitec USB 3.0 Camera information.

Parameters

uiIndex	[in] The number of camera index.
pU3VCameraInfo	[in] A NU3CameraInfo of camera.
pHostControllerInfo	[out] HostControllerInfo

Returns

novitec::CameraAPI::CLR::Error

5.5.2.7 GetCameraInfo() [4/4]

```
Error novitec::CameraAPI::CLR::DeviceManager::GetCameraInfo (
            unsigned int uiIndex,
            [System::Runtime::InteropServices::Out] U3VCameraInfo^ pU3VCameraInfo,
            [System::Runtime::InteropServices::Out] HostControllerInfo^ pHostControllerInfo )
```

Get the USB3 Vision Camera information.

Parameters

uiIndex	[in] The number of camera index.
pU3VCameraInfo	[in] A U3VCameraInfo of camera.
pHostControllerInfo	[out] HostControllerInfo

Returns

novitec::CameraAPI::CLR::Error

5.5.2.8 GetNumberOfCameras()

```
Error novitec::CameraAPI::CLR::DeviceManager::GetNumberOfCameras (
            [System::Runtime::InteropServices::Out] unsigned int% pNumCameras )
```

Get the number of available cameras.

Parameters

pNumCameras	[out] Number of cameras.

Returns

novitec::CameraAPI::CLR::Error

5.5.2.9 SendForceIP()

```
Error novitec::CameraAPI::CLR::DeviceManager::SendForceIP (
            unsigned int uiIndex,
            [System::Runtime::InteropServices::Optional] String^ ip,
            [System::Runtime::InteropServices::Optional] String^ subnetMask,
            [System::Runtime::InteropServices::Optional] String^ gateway )
```

Send Force IP Command. (IP restoration)

If camera and PC can't be communicated because network band of camera is different and PC, you can use this function to recover the camera. If IP is not specified, it is set to a random IP in the same band. This function is only available for GigE Vision camera.

Parameters

uiIndex	The number of camera index.
ip	A `string` of camera IP address to change.
subnetMask	A `string` of subnet mask to change.
gateway	A `string` of gateway to change.

Returns

-rtsp_

5.5.2.10 Update()

```
Error novitec::CameraAPI::CLR::DeviceManager::Update ( )
```

Updates the internal list of available devices.

Returns

novitec::CameraAPI::CLR::Error

The documentation for this class was generated from the following files:

- DeviceManager.h
- DeviceManager.cpp

5.6 novitec::CameraAPI::CLR::Error Class Reference

The class for errors in API functions.

`#include <Error.h>`

Public Member Functions

- **Error** (void)

  Initialize a new instance of the Error class.
- Error (novitec::CameraAPI::Error &error)

  Initialize a new instance of the Error class from the native(C++) Error class.
- Error (const ErrorType errorType)

  Initialize a new instance of the Error class from the ErrorType.
- virtual ~**Error** (void)

  Destructor
- Error ^ **operator=** (Error^ error)
- Error ^ **operator=** (ErrorType^ errorType)
- ErrorType GetType (void)

  Get the error type.
- System::String ^ GetDescription (void)

  Get the error description.

Static Public Member Functions

- static bool **operator==** (const Error^ errorA, const Error^ errorB)
- static bool **operator==** (const Error^ error, const ErrorType^ errorType)
- static bool **operator!=** (const Error^ errorA, const Error^ errorB)
- static bool **operator!=** (const Error^ error, const ErrorType^ errorType)

5.6.1 Detailed Description

The class for errors in API functions.

5.6.2 Constructor & Destructor Documentation

5.6.2.1 Error() [1/2]

```
Error::Error (
            novitec::CameraAPI::Error & error )
```

Initialize a new instance of the Error class from the native(C++) Error class.

Parameters

error	The instance of native(C++) Error class which to create the new Error.

5.6.2.2 Error() [2/2]

```
Error::Error (
            const ErrorType errorType )
```

Initialize a new instance of the Error class from the ErrorType.

Parameters

errorType	novitec::CameraAPI::CLR::ErrorType

5.6.3 Member Function Documentation

5.6.3.1 GetDescription()

```
System::String Error::GetDescription (
            void  )
```

Get the error description.

Returns

A null-terminated string with explanatory information.

5.6.3.2 GetType()

```
ErrorType Error::GetType (
            void  )
```

Get the error type.

Returns

novitec::CameraAPI::CLR::ErrorType

The documentation for this class was generated from the following files:

- Error.h
- Error.cpp

5.7 novitec::CameraAPI::CLR::Gateway Struct Reference

Gateway Structure

`#include <Types.h>`

Public Attributes

- array< Byte > ^ **address** = gcnew array<Byte>(4)
  Address of Gateway

5.7.1 Detailed Description

Gateway Structure

The documentation for this struct was generated from the following file:

- Types.h

5.8 novitec::CameraAPI::CLR::GEVCameraInfo Class Reference

GigE Vision camera Information.

`#include <Types.h>`

Public Member Functions

- **GEVCameraInfo** ()
  Initialize a new instance of the GEVCameraInfo class.
- ~**GEVCameraInfo** ()
  Destructor
- novitec::CameraAPI::GEVCameraInfo * GetCGEVCameraInfo ()
  Get pointer to the native(C++) GEVCameraInfo class.

Public Attributes

- unsigned short **spec_version_major**

  Major version of the GigE Standard with which the device is compliant.
- unsigned short **spec_version_minor**

  Minor version of the GigE Standard with which the device is compliant.
- unsigned int **device_mode**

  Device Mode
- unsigned short **reserved_0**

  Reserved.
- array< Byte > ^ **device_MAC_address_h**

  The 2 high-order bytes of the MAC address. (16 bits / 2 bytes)
- array< Byte > ^ **device_MAC_address_l**

  The 4 low-order bytes of the MAC address. (32 bits / 2 bytes)
- unsigned int **IP_config_options**

  Network Interface Capability
- unsigned int **IP_config_current**

  Network Interface Configuration
- unsigned int **reserved_1**

  Reserved.
- unsigned int **reserved_2**

  Reserved.
- unsigned int **reserved_3**

  Reserved.
- array< Byte > ^ **current_IP**

  Current IP Address (4 bytes)
- unsigned int **reserved_4**

  Reserved.
- unsigned int **reserved_5**

  Reserved.
- unsigned int **reserved_6**

  Reserved.
- array< Byte > ^ **current_subnet_mask**

  Current Subnet Mask (4 bytes)
- unsigned int **reserved_7**

  Reserved.
- unsigned int **reserved_8**

  Reserved.
- unsigned int **reserved_9**

  Reserved.
- array< Byte > ^ **default_gateway**

  Current Gateway (4 bytes)
- String ^ **manufacturer_name**

  Device Vendor Name
- String ^ **model_name**

  Device Model Name
- String ^ **device_version**

  Device Version
- String ^ **manufacturer_specific_information**

  Manufacturer Specific Information
- String ^ **serial_number**

  Serial Number (16 bytes)
- String ^ **user_defined_name**

  User Defined Name (16 bytes)

Protected Attributes

- novitec::CameraAPI::GEVCameraInfo * **m_GEVCameraInfo**

5.8.1 Detailed Description

GigE Vision camera Information.

5.8.2 Member Function Documentation

5.8.2.1 GetCGEVCameraInfo()

```
novitec::CameraAPI::GEVCameraInfo * GEVCameraInfo::GetCGEVCameraInfo ( )
```

Get pointer to the native(C++) GEVCameraInfo class.

Returns

Pointer to the native(C++) GEVCameraInfo class.

The documentation for this class was generated from the following files:

- Types.h
- Types.cpp

5.9 novitec::CameraAPI::CLR::Handle Class Reference

The class which provides handle of device.

```
#include <Handle.h>
```

Public Member Functions

- **Handle** (void)
  Initialize a new instance of the Handle class.
- ~**Handle** (void)
  Destructor.
- novitec::CameraAPI::Handle * GetCHandle ()
  Get the pointer to the instance of native(C++) Handle class.
- System::String ^ GetDescriptor (void)
  Get descriptor of handle.
- InterfaceType GetInterfaceType (void)
  Get the interface type.
- IntPtr GetInterfaceHandle (void)
  Get the interface handle.

Protected Attributes

- novitec::CameraAPI::Handle * **handle**

5.9.1 Detailed Description

The class which provides handle of device.

5.9.2 Member Function Documentation

5.9.2.1 GetCHandle()

```
novitec::CameraAPI::Handle * Handle::GetCHandle ( )
```

Get the pointer to the instance of native(C++) Handle class.

Returns

The pointer to the instance of native(C++) Handle class.

5.9.2.2 GetDescriptor()

```
System::String Handle::GetDescriptor (
            void  )
```

Get descriptor of handle.

Returns

Descriptor of Handle

5.9.2.3 GetInterfaceHandle()

```
IntPtr Handle::GetInterfaceHandle (
            void  )
```

Get the interface handle.

Returns

Interface Handle

5.9.2.4 GetInterfaceType()

```
InterfaceType Handle::GetInterfaceType (
            void  )
```

Get the interface type.

Returns

Interface Type

The documentation for this class was generated from the following files:

- Handle.h
- Handle.cpp

5.10 novitec::CameraAPI::CLR::HostControllerInfo Class Reference

Host Controller Information

```
#include <Types.h>
```

Public Member Functions

- **HostControllerInfo** ()
  Initialize a new instance of the HostController class.
- ~**HostControllerInfo** ()
  Destructor
- novitec::CameraAPI::HostControllerInfo * GetCHostControllerInfo ()
  Get the pointer to the instance of native(C++) HostControllerInfo class.

Protected Attributes

- novitec::CameraAPI::HostControllerInfo * **m_HostControllerInfo**

5.10.1 Detailed Description

Host Controller Information

5.10.2 Member Function Documentation

5.10.2.1 GetCHostControllerInfo()

```
novitec::CameraAPI::HostControllerInfo * HostControllerInfo::GetCHostControllerInfo ( )
```

Get the pointer to the instance of native(C++) HostControllerInfo class.

Returns

The pointer to the instance of native(C++) HostController class.

The documentation for this class was generated from the following files:

- Types.h
- Types.cpp

5.11 novitec::CameraAPI::CLR::GenICam::IBoolean Class Reference

IBoolean class.

```
#include <ITypes.h>
```

Public Member Functions

- **IBoolean** ()
  Initialize a new instance of the IBoolean class.
- novitec::CameraAPI::GenICam::IBoolean (const novitec::CameraAPI::GenICam::IBoolean &CIBoolean)
  Initialize a new instance of the IBoolean class from the instance of native(C++) IBoolean class.
- ~**IBoolean** ()
  Destructor
- novitec::CameraAPI::GenICam::IBoolean * GetCIBoolean ()
  Get the pointer to the instance of native(C++) IBoolean class.
- bool GetValue ()
- bool GetCacheValue ()
  Get cached feature value. Returns the stored feature value without attemping to read register.
- void SetValue (bool value)
  Set value of the boolean type feature.
- System::String ^ GetType ()
  Get the feature type.
- System::String ^ GetTypeName ()
  Get the feature type name.
- System::String ^ GetCategory ()
  Get the category of the feature.
- System::String ^ GetFeatureName ()
  Get the name of feature.
- bool IsValid ()
  Get whether the feature is valid.
- bool IsImplemented ()
  Get whether the feature is implemented.
- bool IsLocked ()

Get whether the feature is locked.

- AccessMode GetAccessMode ()

  Get the access mode.

- Visibility GetVisibility ()

  Get the visibility.

- System::String ^ GetDisplayName ()

  Get the display name.

- System::String ^ GetToolTip ()

  Get the tooltip.

- System::String ^ GetDescription ()

  Get the description.

- System::String ^ GetElement (System::String^ elementName)

  Get the specified element.

Protected Attributes

- novitec::CameraAPI::GenICam::IBoolean * **m_IBoolean**

5.11.1 Detailed Description

IBoolean class.

5.11.2 Member Function Documentation

5.11.2.1 GetAccessMode()

```
AccessMode IBoolean::GetAccessMode ( )
```

Get the access mode.

Returns

AccessMode

5.11.2.2 GetCacheValue()

```
bool IBoolean::GetCacheValue ( )
```

Get cached feature value. Returns the stored feature value without attemping to read register.

Returns

Cached feature value.

5.11.2.3 GetCategory()

```
System::String IBoolean::GetCategory ( )
```

Get the category of the feature.

Returns

A string that represents the category of the feature.

5.11.2.4 GetCIBoolean()

```
novitec::CameraAPI::GenICam::IBoolean * IBoolean::GetCIBoolean ( )
```

Get the pointer to the instance of native(C++) IBoolean class.

Returns

The pointer to the instance of navtive(C++) IBoolean class.

5.11.2.5 GetDescription()

```
System::String IBoolean::GetDescription ( )
```

Get the description.

Returns

A `string` that represents the description.

5.11.2.6 GetDisplayName()

```
System::String IBoolean::GetDisplayName ( )
```

Get the display name.

Returns

A `string` that represents the display name.

5.11.2.7 GetElement()

```
System::String IBoolean::GetElement (
            System::String^ elementName )
```

Get the specified element.

Parameters

elementName	The pointer to a null-terminated string with value of specified element.

Returns

A `string` that represents the specified element.

5.11.2.8 GetFeatureName()

```
System::String IBoolean::GetFeatureName ( )
```

Get the name of feature.

Returns

A string that represents the name of feature.

5.11.2.9 GetToolTip()

```
System::String IBoolean::GetToolTip ( )
```

Get the tooltip.

Returns

A `string` that represents the tooltip.

5.11.2.10 GetType()

```
System::String IBoolean::GetType ( )
```

Get the feature type.

Returns

A string that represents the feature type.

5.11.2.11 GetTypeName()

```
System::String IBoolean::GetTypeName ( )
```

Get the feature type name.

Returns

A string that represents the feature type name.

5.11.2.12 GetValue()

```
bool IBoolean::GetValue ( )
```

Get value of the boolean type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register read fail.

Returns

The value of the boolean type feature.

5.11.2.13 GetVisibility()

```
Visibility IBoolean::GetVisibility ( )
```

Get the visibility.

Returns

Visibility

5.11.2.14 IsImplemented()

```
bool IBoolean::IsImplemented ( )
```

Get whether the feature is implemented.

Returns

`true` if the feature is implemented; otherwise, `false`.

5.11.2.15 IsLocked()

```
bool IBoolean::IsLocked ( )
```

Get whether the feature is locked.

Returns

`true` if the feature is locked; otherwise, `false`.

5.11.2.16 IsValid()

```
bool IBoolean::IsValid ( )
```

Get whether the feature is valid.

Returns

`true` if the feature is valid; otherwise, `false`.

5.11.2.17 novitec::CameraAPI::GenICam::IBoolean()

```
novitec::CameraAPI::CLR::GenICam::IBoolean::novitec::CameraAPI::GenICam::IBoolean (
            const novitec::CameraAPI::GenICam::IBoolean & CIBoolean )
```

Initialize a new instance of the IBoolean class from the instance of native(C++) IBoolean class.

Parameters

CIBoolean	

5.11.2.18 SetValue()

```
void IBoolean::SetValue (
            bool value )
```

Set value of the boolean type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register write fail.

Parameters

value	The value of the boolean type feature.

The documentation for this class was generated from the following files:

- GenICam/ITypes.h
- GenICam/ITypes.cpp

5.12 novitec::CameraAPI::CLR::GenICam::ICategory Class Reference

ICategory class.

`#include <ITypes.h>`

Public Member Functions

- **ICategory** ()

  Initialize a new instance of the ICategory class.
- novitec::CameraAPI::GenICam::ICategory (const novitec::CameraAPI::GenICam::ICategory &CICategory)

  Initialize a new instance of the ICategory class from the instance of native(C++) ICategory class.
- novitec::CameraAPI::GenICam::ICategory * GetCICategory ()

  Get the pointer to the instance of native(C++) ICategory class.
- System::String ^ GetType ()
- System::String ^ GetTypeName ()

  Get the feature type name.
- System::String ^ GetCategory ()

  Get the category of the feature.
- System::String ^ GetFeatureName ()

  Get the name of feature.
- bool IsValid ()

  Get whether the feature is valid.
- bool IsImplemented ()

  Get whether the feature is implemented.
- bool IsLocked ()

  Get whether the feature is locked.
- AccessMode GetAccessMode ()

  Get the access mode.
- Visibility GetVisibility ()

  Get the visibility.
- System::String ^ GetDisplayName ()

  Get the display name.
- System::String ^ GetToolTip ()

  Get the tooltip.
- System::String ^ GetDescription ()

  Get the description.
- System::String ^ GetElement (System::String^ elementName)

  Get the specified element.

Protected Attributes

- novitec::CameraAPI::GenICam::ICategory * **m_ICategory**

5.12.1 Detailed Description

ICategory class.

5.12.2 Member Function Documentation

5.12.2.1 GetAccessMode()

```
AccessMode ICategory::GetAccessMode ( )
```

Get the access mode.

Returns

AccessMode

5.12.2.2 GetCategory()

```
System::String ICategory::GetCategory ( )
```

Get the category of the feature.

Returns

A string that represents the category of the feature.

5.12.2.3 GetCICategory()

```
novitec::CameraAPI::GenICam::ICategory * ICategory::GetCICategory ( )
```

Get the pointer to the instance of native(C++) ICategory class.

Returns

The pointer to the instance of native(C++) ICategory class.

5.12.2.4 GetDescription()

```
System::String ICategory::GetDescription ( )
```

Get the description.

Returns

A `string` that represents the description.

5.12.2.5 GetDisplayName()

```
System::String ICategory::GetDisplayName ( )
```

Get the display name.

Returns

A `string` that represents the display name.

5.12.2.6 GetElement()

```
System::String ICategory::GetElement (
            System::String^ elementName )
```

Get the specified element.

Parameters

elementName	The pointer to a null-terminated string with value of specified element.

Returns

A `string` that represents the specified element.

5.12.2.7 GetFeatureName()

```
System::String ICategory::GetFeatureName ( )
```

Get the name of feature.

Returns

A string that represents the name of feature.

5.12.2.8 GetToolTip()

```
System::String ICategory::GetToolTip ( )
```

Get the tooltip.

Returns

A `string` that represents the tooltip.

5.12.2.9 GetType()

```
System::String ICategory::GetType ( )
```

Get the feature type.

Returns

A string that represents the feature type.

5.12.2.10 GetTypeName()

```
System::String ICategory::GetTypeName ( )
```

Get the feature type name.

Returns

A string that represents the feature type name.

5.12.2.11 GetVisibility()

```
Visibility ICategory::GetVisibility ( )
```

Get the visibility.

Returns

Visibility

5.12.2.12 IsImplemented()

```
bool ICategory::IsImplemented ( )
```

Get whether the feature is implemented.

Returns

`true` if the feature is implemented; otherwise, `false`.

5.12.2.13 IsLocked()

```
bool ICategory::IsLocked ( )
```

Get whether the feature is locked.

Returns

`true` if the feature is locked; otherwise, `false`.

5.12.2.14 IsValid()

```
bool ICategory::IsValid ( )
```

Get whether the feature is valid.

Returns

`true` if the feature is valid; otherwise, `false`.

5.12.2.15 novitec::CameraAPI::GenICam::ICategory()

```
novitec::CameraAPI::CLR::GenICam::ICategory::novitec::CameraAPI::GenICam::ICategory (
            const novitec::CameraAPI::GenICam::ICategory & CICategory )
```

Initialize a new instance of the ICategory class from the instance of native(C++) ICategory class.

Parameters

CICategory	The instance of native(C++) ICategory class which to create the new ICategory.

The documentation for this class was generated from the following files:

- GenICam/ITypes.h
- GenICam/ITypes.cpp

5.13 novitec::CameraAPI::CLR::GenICam::ICommand Class Reference

ICommand class.

```
#include <ITypes.h>
```

Public Member Functions

- **ICommand** ()

  Initialize a new instance of the ICommand class.
- novitec::CameraAPI::GenICam::ICommand (const novitec::CameraAPI::GenICam::ICommand &CICommand)

  Initialize a new instance of the ICategory class from the instance of natvie(C++) ICommand class.
- ~**ICommand** ()

  Destructor
- novitec::CameraAPI::GenICam::ICommand * GetCICommand ()

  Get the pointer to the instance of native(C++) ICommand class.
- void **Execute** ()

  Execute the command.
- bool IsDone ()

  Check if the command execution is done.
- System::String ^ GetType ()

  Get the feature type.
- System::String ^ GetTypeName ()

  Get the feature type name.
- System::String ^ GetCategory ()

  Get the category of the feature.
- System::String ^ GetFeatureName ()

  Get the name of feature.
- bool IsValid ()

  Get whether the feature is valid.
- bool IsImplemented ()

  Get whether the feature is implemented.
- bool IsLocked ()

  Get whether the feature is locked.
- AccessMode GetAccessMode ()

  Get the access mode.
- Visibility GetVisibility ()

  Get the visibility.
- System::String ^ GetDisplayName ()

  Get the display name.
- System::String ^ GetToolTip ()

  Get the tooltip.
- System::String ^ GetDescription ()

  Get the description.
- System::String ^ GetElement (System::String^ elementName)

  Get the specified element.

Protected Attributes

- novitec::CameraAPI::GenICam::ICommand * **m_ICommand**

5.13.1 Detailed Description

ICommand class.

5.13.2 Member Function Documentation

5.13.2.1 GetAccessMode()

```
AccessMode ICommand::GetAccessMode ( )
```

Get the access mode.

Returns

AccessMode

5.13.2.2 GetCategory()

```
System::String ICommand::GetCategory ( )
```

Get the category of the feature.

Returns

A string that represents the category of the feature.

5.13.2.3 GetCICommand()

```
novitec::CameraAPI::GenICam::ICommand * ICommand::GetCICommand ( )
```

Get the pointer to the instance of native(C++) ICommand class.

Returns

The pointer to the instance of navtive(C++) ICommand class.

5.13.2.4 GetDescription()

```
System::String ICommand::GetDescription ( )
```

Get the description.

Returns

A `string` that represents the description.

5.13.2.5 GetDisplayName()

```
System::String ICommand::GetDisplayName ( )
```

Get the display name.

Returns

A `string` that represents the display name.

5.13.2.6 GetElement()

```
System::String ICommand::GetElement (
            System::String^ elementName )
```

Get the specified element.

Parameters

elementName	The pointer to a null-terminated string with value of specified element.

Returns

A `string` that represents the specified element.

5.13.2.7 GetFeatureName()

```
System::String ICommand::GetFeatureName ( )
```

Get the name of feature.

Returns

A string that represents the name of feature.

5.13.2.8 GetToolTip()

```
System::String ICommand::GetToolTip ( )
```

Get the tooltip.

Returns

A `string` that represents the tooltip.

5.13.2.9 GetType()

```
System::String ICommand::GetType ( )
```

Get the feature type.

Returns

A string that represents the feature type.

5.13.2.10 GetTypeName()

```
System::String ICommand::GetTypeName ( )
```

Get the feature type name.

Returns

A string that represents the feature type name.

5.13.2.11 GetVisibility()

```
Visibility ICommand::GetVisibility ( )
```

Get the visibility.

Returns

Visibility

5.13.2.12 IsDone()

```
bool ICommand::IsDone ( )
```

Check if the command execution is done.

Returns

`true` if the command execution is done; otherwise, `false`.

5.13.2.13 IsImplemented()

```
bool ICommand::IsImplemented ( )
```

Get whether the feature is implemented.

Returns

`true` if the feature is implemented; otherwise, `false`.

5.13.2.14 IsLocked()

```
bool ICommand::IsLocked ( )
```

Get whether the feature is locked.

Returns

`true` if the feature is locked; otherwise, `false`.

5.13.2.15 IsValid()

```
bool ICommand::IsValid ( )
```

Get whether the feature is valid.

Returns

`true` if the feature is valid; otherwise, `false`.

5.13.2.16 novitec::CameraAPI::GenICam::ICommand()

```
novitec::CameraAPI::CLR::GenICam::ICommand::novitec::CameraAPI::GenICam::ICommand (
            const novitec::CameraAPI::GenICam::ICommand & CICommand )
```

Initialize a new instance of the ICategory class from the instance of natvie(C++) ICommand class.

Parameters

CICommand	The instance(C++) ICommand class which to create the new ICommand.

The documentation for this class was generated from the following files:

- GenICam/ITypes.h
- GenICam/ITypes.cpp

5.14 novitec::CameraAPI::CLR::GenICam::IEnumEntry Class Reference

IEnumEntry class.

`#include <ITypes.h>`

Public Member Functions

- **IEnumEntry** ()
  Initialize a new instance of the IEnumEntry class.
- IEnumEntry (const novitec::CameraAPI::GenICam::IEnumEntry &CIEnumEntry)
  Initialize a new instance of the IEnumEntry class from the instance of native(C++) IEnumEntry class.
- ~**IEnumEntry** ()
  Destructor
- novitec::CameraAPI::GenICam::IEnumEntry * GetCIEnumEntry ()
  Get the pointer to the instance of native(C++) IEnumEntry class.
- int64_t GetValue ()
  Get value of the enumeration entry type feature.
- System::String ^ GetType ()
  Get the feature type.
- System::String ^ GetTypeName ()
  Get the feature type name.
- System::String ^ GetCategory ()
  Get the category of the feature.
- System::String ^ GetFeatureName ()
  Get the name of feature.
- bool IsValid ()
  Get whether the feature is valid.
- bool IsImplemented ()
  Get whether the feature is implemented.
- bool IsLocked ()
  Get whether the feature is locked.
- AccessMode GetAccessMode ()
  Get the access mode.
- Visibility GetVisibility ()
  Get the visibility.
- System::String ^ GetDisplayName ()
  Get the display name.
- System::String ^ GetToolTip ()
  Get the tooltip.
- System::String ^ GetDescription ()
  Get the description.
- System::String ^ GetElement (System::String^ elementName)
  Get the specified element.

Protected Attributes

- novitec::CameraAPI::GenICam::IEnumEntry * **m_IEnumEntry**

5.14.1 Detailed Description

IEnumEntry class.

5.14.2 Constructor & Destructor Documentation

5.14.2.1 IEnumEntry()

```
IEnumEntry::IEnumEntry (
            const novitec::CameraAPI::GenICam::IEnumEntry & CIEnumEntry )
```

Initialize a new instance of the IEnumEntry class from the instance of native(C++) IEnumEntry class.

Parameters

CIEnumEntry	The instance of native(C++) IEnumEntry class which to create the new IEnumEntry.

5.14.3 Member Function Documentation

5.14.3.1 GetAccessMode()

```
AccessMode IEnumEntry::GetAccessMode ( )
```

Get the access mode.

Returns

AccessMode

5.14.3.2 GetCategory()

```
System::String IEnumEntry::GetCategory ( )
```

Get the category of the feature.

Returns

A string that represents the category of the feature.

5.14.3.3 GetCIEnumEntry()

```
novitec::CameraAPI::GenICam::IEnumEntry * IEnumEntry::GetCIEnumEntry ( )
```

Get the pointer to the instance of native(C++) IEnumEntry class.

Returns

The pointer to the instance of native(C++) IEnumEntry class.

5.14.3.4 GetDescription()

```
System::String IEnumEntry::GetDescription ( )
```

Get the description.

Returns

A `string` that represents the description.

5.14.3.5 GetDisplayName()

```
System::String IEnumEntry::GetDisplayName ( )
```

Get the display name.

Returns

A `string` that represents the display name.

5.14.3.6 GetElement()

```
System::String IEnumEntry::GetElement (
            System::String^ elementName )
```

Get the specified element.

Parameters

elementName	The pointer to a null-terminated string with value of specified element.

Returns

A `string` that represents the specified element.

5.14.3.7 GetFeatureName()

```
System::String IEnumEntry::GetFeatureName ( )
```

Get the name of feature.

Returns

A string that represents the name of feature.

5.14.3.8 GetToolTip()

```
System::String IEnumEntry::GetToolTip ( )
```

Get the tooltip.

Returns

A `string` that represents the tooltip.

5.14.3.9 GetType()

```
System::String IEnumEntry::GetType ( )
```

Get the feature type.

Returns

A string that represents the feature type.

5.14.3.10 GetTypeName()

```
System::String IEnumEntry::GetTypeName ( )
```

Get the feature type name.

Returns

A string that represents the feature type name.

5.14.3.11 GetValue()

```
int64_t IEnumEntry::GetValue ( )
```

Get value of the enumeration entry type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register read fail.

Returns

The value of the enumeration entry type feature.

5.14.3.12 GetVisibility()

```
Visibility IEnumEntry::GetVisibility ( )
```

Get the visibility.

Returns

Visibility

5.14.3.13 IsImplemented()

```
bool IEnumEntry::IsImplemented ( )
```

Get whether the feature is implemented.

Returns

`true` if the feature is implemented; otherwise, `false`.

5.14.3.14 IsLocked()

```
bool IEnumEntry::IsLocked ( )
```

Get whether the feature is locked.

Returns

`true` if the feature is locked; otherwise, `false`.

5.14.3.15 IsValid()

```
bool IEnumEntry::IsValid ( )
```

Get whether the feature is valid.

Returns

`true` if the feature is valid; otherwise, `false`.

The documentation for this class was generated from the following files:

- GenICam/ITypes.h
- GenICam/ITypes.cpp

5.15 novitec::CameraAPI::CLR::GenICam::IEnumeration Class Reference

IEnumeration class.

```
#include <ITypes.h>
```

Public Member Functions

- **IEnumeration** ()
  Initialize a new instance of the IEnumeration class.
- novitec::CameraAPI::GenICam::IEnumeration (const novitec::CameraAPI::GenICam::IEnumeration &CIEnumeration)
  Initialize a new instance of the IEnumeration class from the instance of native(C++) IEnumeration class.
- ~**IEnumeration** ()
  Destructor
- novitec::CameraAPI::GenICam::IEnumeration * GetCIEnumeration ()
  Get the pointer to the instance of native(C++) IEnumeration class.
- int GetIntValue ()
  Get integer value of the enumeration type feature.
- int GetCacheIntValue ()
  Get cached feature value. Returns the stored feature value without attemping to read register.
- void SetIntValue (int64_t value)
  Set integer value of the enumeration type feature.
- System::String ^ GetSymbolicValue ()
  Get symbolic value of the enumeration type feature.
- void SetSymbolicValue (System::String^ value)
  Set symbolic value of the enumeration type feature.
- int GetNumberOfEntries ()
  Get number of entries.
- IEnumEntry ^ GetEntry (int index)
  Get entry with given index.
- System::String ^ GetType ()

Get the feature type.

- System::String ^ GetTypeName ()

  Get the feature type name.
- System::String ^ GetCategory ()

  Get the category of the feature.
- System::String ^ GetFeatureName ()

  Get the name of feature.
- bool IsValid ()

  Get whether the feature is valid.
- bool IsImplemented ()

  Get whether the feature is implemented.
- bool IsLocked ()

  Get whether the feature is locked.
- AccessMode GetAccessMode ()

  Get the access mode.
- Visibility GetVisibility ()

  Get the visibility.
- System::String ^ GetDisplayName ()

  Get the display name.
- System::String ^ GetToolTip ()

  Get the tooltip.
- System::String ^ GetDescription ()

  Get the description.
- System::String ^ GetElement (System::String^ elementName)

  Get the specified element.

Protected Attributes

- novitec::CameraAPI::GenICam::IEnumeration * **m_IEnumeration**

5.15.1 Detailed Description

IEnumeration class.

5.15.2 Member Function Documentation

5.15.2.1 GetAccessMode()

```
AccessMode IEnumeration::GetAccessMode ( )
```

Get the access mode.

Returns

AccessMode

5.15.2.2 GetCacheIntValue()

```
int IEnumeration::GetCacheIntValue ( )
```

Get cached feature value. Returns the stored feature value without attemping to read register.

Returns

Cached feature value.

5.15.2.3 GetCategory()

```
System::String IEnumeration::GetCategory ( )
```

Get the category of the feature.

Returns

A string that represents the category of the feature.

5.15.2.4 GetCIEnumeration()

```
novitec::CameraAPI::GenICam::IEnumeration * IEnumeration::GetCIEnumeration ( )
```

Get the pointer to the instance of native(C++) IEnumeration class.

Returns

The pointer to the instance of native(C++) IEnumeration class.

5.15.2.5 GetDescription()

```
System::String IEnumeration::GetDescription ( )
```

Get the description.

Returns

A `string` that represents the description.

5.15.2.6 GetDisplayName()

```
System::String IEnumeration::GetDisplayName ( )
```

Get the display name.

Returns

A `string` that represents the display name.

5.15.2.7 GetElement()

```
System::String IEnumeration::GetElement (
            System::String^ elementName )
```

Get the specified element.

Parameters

elementName	The pointer to a null-terminated string with value of specified element.

Returns

A `string` that represents the specified element.

5.15.2.8 GetEntry()

```
IEnumEntry IEnumeration::GetEntry (
            int index )
```

Get entry with given index.

Parameters

index	Index of entries.

Returns

IEnumEntry

5.15.2.9 GetFeatureName()

```
System::String IEnumeration::GetFeatureName ( )
```

Get the name of feature.

Returns

A string that represents the name of feature.

5.15.2.10 GetIntValue()

```
int IEnumeration::GetIntValue ( )
```

Get integer value of the enumeration type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register read fail.

Returns

The integer value of the enumeration type feature.

5.15.2.11 GetNumberOfEntries()

```
int IEnumeration::GetNumberOfEntries ( )
```

Get number of entries.

Returns

Number of entries.

5.15.2.12 GetSymbolicValue()

```
System::String IEnumeration::GetSymbolicValue ( )
```

Get symbolic value of the enumeration type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register read fail.

Returns

The symbolic value of the enumeration type feature.

5.15.2.13 GetToolTip()

```
System::String IEnumeration::GetToolTip ( )
```

Get the tooltip.

Returns

A `string` that represents the tooltip.

5.15.2.14 GetType()

```
System::String IEnumeration::GetType ( )
```

Get the feature type.

Returns

A string that represents the feature type.

5.15.2.15 GetTypeName()

```
System::String IEnumeration::GetTypeName ( )
```

Get the feature type name.

Returns

A string that represents the feature type name.

5.15.2.16 GetVisibility()

```
Visibility IEnumeration::GetVisibility ( )
```

Get the visibility.

Returns

Visibility

5.15.2.17 IsImplemented()

```
bool IEnumeration::IsImplemented ( )
```

Get whether the feature is implemented.

Returns

`true` if the feature is implemented; otherwise, `false`.

5.15.2.18 IsLocked()

```
bool IEnumeration::IsLocked ( )
```

Get whether the feature is locked.

Returns

`true` if the feature is locked; otherwise, `false`.

5.15.2.19 IsValid()

```
bool IEnumeration::IsValid ( )
```

Get whether the feature is valid.

Returns

`true` if the feature is valid; otherwise, `false`.

5.15.2.20 novitec::CameraAPI::GenICam::IEnumeration()

```
novitec::CameraAPI::CLR::GenICam::IEnumeration::novitec::CameraAPI::GenICam::IEnumeration (
            const novitec::CameraAPI::GenICam::IEnumeration & CIEnumeration )
```

Initialize a new instance of the IEnumeration class from the instance of native(C++) IEnumeration class.

Parameters

CIEnumeration	

5.15.2.21 SetIntValue()

```
void IEnumeration::SetIntValue (
            int64_t value )
```

Set integer value of the enumeration type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register write fail.

Parameters

value	The integer value of the enumeration type feature.

5.15.2.22 SetSymbolicValue()

```
void IEnumeration::SetSymbolicValue (
            System::String^ value )
```

Set symbolic value of the enumeration type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register write fail.

Parameters

value	The symbolic value of the enumeration type feature.

The documentation for this class was generated from the following files:

- GenICam/ITypes.h
- GenICam/ITypes.cpp

5.16 novitec::CameraAPI::CLR::GenICam::IFloat Class Reference

IFloat class.

```
#include <ITypes.h>
```

Public Member Functions

- **IFloat** ()
  Initialize a new instance of the IFloat class.
- novitec::CameraAPI::GenICam::IFloat (const novitec::CameraAPI::GenICam::IFloat &CIFloat)
  Initialize a new instance of the IFloat class from the instance of native(C++) IFloat class.

- ~**IFloat** ()

  Destructor
- novitec::CameraAPI::GenICam::IFloat * GetCIFloat ()

  Get the pointer to the instance of native(C++) IFloat class.
- double GetValue ()

  Get value of the float type feature.
- double GetCacheValue ()

  Get cached feature value. Reutrns the stored feature value without attemping to read register.
- void SetValue (double value)

  Set value of the float type feature.
- bool HasMin ()

  Get whether the feature has minimum value.
- bool HasMax ()

  Get whether the feature has minimum value.
- bool HasInc ()

  Get whether the feature has minimum value.
- double GetMin ()

  Get the minimum value of float type feature.
- double GetMax ()

  Get the maximum value of float type feature.
- IncMode GetIncMode ()

  Get the increment mode.
- double GetInc ()

  Get the increment value.
- int GetNumberOfValidValues ()

  Get the list of valid values.
- double GetValidValue (int index)

  Get the valid value.
- FloatRepresentation GetRepresentation ()

  Get the representation.
- System::String ^ GetUnit ()

  Get the unit.
- System::String ^ GetType ()

  Get the feature type.
- System::String ^ GetTypeName ()

  Get the feature type name.
- System::String ^ GetCategory ()

  Get the category of the feature.
- System::String ^ GetFeatureName ()

  Get the name of feature.
- bool IsValid ()

  Get whether the feature is valid.
- bool IsImplemented ()

  Get whether the feature is implemented.
- bool IsLocked ()

  Get whether the feature is locked.
- AccessMode GetAccessMode ()

  Get the access mode.
- Visibility GetVisibility ()

  Get the visibility.
- System::String ^ GetDisplayName ()

Get the display name.

- System::String ^ GetToolTip ()

  Get the tooltip.

- System::String ^ GetDescription ()

  Get the description.

- System::String ^ GetElement (System::String^ elementName)

  Get the specified element.

Protected Attributes

- novitec::CameraAPI::GenICam::IFloat * **m_IFloat**

5.16.1 Detailed Description

IFloat class.

5.16.2 Member Function Documentation

5.16.2.1 GetAccessMode()

```
AccessMode IFloat::GetAccessMode ( )
```

Get the access mode.

Returns

AccessMode

5.16.2.2 GetCacheValue()

```
double IFloat::GetCacheValue ( )
```

Get cached feature value. Reutrns the stored feature value without attemping to read register.

Returns

Cached feature value.

5.16.2.3 GetCategory()

```
System::String IFloat::GetCategory ( )
```

Get the category of the feature.

Returns

A string that represents the category of the feature.

5.16.2.4 GetCIFloat()

```
novitec::CameraAPI::GenICam::IFloat * IFloat::GetCIFloat ( )
```

Get the pointer to the instance of native(C++) IFloat class.

Returns

The pointer to the instance of native(C++) IFloat class.

5.16.2.5 GetDescription()

```
System::String IFloat::GetDescription ( )
```

Get the description.

Returns

A `string` that represents the description.

5.16.2.6 GetDisplayName()

```
System::String IFloat::GetDisplayName ( )
```

Get the display name.

Returns

A `string` that represents the display name.

5.16.2.7 GetElement()

```
System::String IFloat::GetElement (
            System::String^ elementName )
```

Get the specified element.

Parameters

elementName	The pointer to a null-terminated string with value of specified element.

Returns

A `string` that represents the specified element.

5.16.2.8 GetFeatureName()

```
System::String IFloat::GetFeatureName ( )
```

Get the name of feature.

Returns

A string that represents the name of feature.

5.16.2.9 GetInc()

```
double IFloat::GetInc ( )
```

Get the increment value.

[Exception] novitec::CameraAPI::Exception
An exception will be raised if the minimum value is not specified.

Returns

The increment value.

5.16.2.10 GetIncMode()

```
IncMode IFloat::GetIncMode ( )
```

Get the increment mode.

Returns

IncMode

5.16.2.11 GetMax()

```
double IFloat::GetMax ( )
```

Get the maximum value of float type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be raised if the minimum value is not specified.

Returns

The maximum value of float type feature.

5.16.2.12 GetMin()

```
double IFloat::GetMin ( )
```

Get the minimum value of float type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be raised if the minimum value is not specified.

Returns

The minimum value of float type feature.

5.16.2.13 GetNumberOfValidValues()

```
int IFloat::GetNumberOfValidValues ( )
```

Get the list of valid values.

Returns

Values if the IncMode is IM_List.

5.16.2.14 GetRepresentation()

```
FloatRepresentation IFloat::GetRepresentation ( )
```

Get the representation.

Returns

FloatRepresentation

5.16.2.15 GetToolTip()

```
System::String IFloat::GetToolTip ( )
```

Get the tooltip.

Returns

A `string` that represents the tooltip.

5.16.2.16 GetType()

```
System::String IFloat::GetType ( )
```

Get the feature type.

Returns

A string that represents the feature type.

5.16.2.17 GetTypeName()

```
System::String IFloat::GetTypeName ( )
```

Get the feature type name.

Returns

A string that represents the feature type name.

5.16.2.18 GetUnit()

```
System::String IFloat::GetUnit ( )
```

Get the unit.

Returns

The unit.

5.16.2.19 GetValidValue()

```
double IFloat::GetValidValue (
            int index )
```

Get the valid value.

Parameters

index	Index of valid values list.

Returns

Valid value.

5.16.2.20 GetValue()

```
double IFloat::GetValue ( )
```

Get value of the float type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register read fail.

Returns

The value of the float type feature.

5.16.2.21 GetVisibility()

```
Visibility IFloat::GetVisibility ( )
```

Get the visibility.

Returns

Visibility

5.16.2.22 HasInc()

```
bool IFloat::HasInc ( )
```

Get whether the feature has minimum value.

Returns

`true` if the feature has increment value; otherwise, `false`.

5.16.2.23 HasMax()

```
bool IFloat::HasMax ( )
```

Get whether the feature has minimum value.

Returns

`true` if the feature has maximum value; otherwise, `false`.

5.16.2.24 HasMin()

```
bool IFloat::HasMin ( )
```

Get whether the feature has minimum value.

Returns

`true` if the feature has minimum value; otherwise, `false`.

5.16.2.25 IsImplemented()

```
bool IFloat::IsImplemented ( )
```

Get whether the feature is implemented.

Returns

`true` if the feature is implemented; otherwise, `false`.

5.16.2.26 IsLocked()

```
bool IFloat::IsLocked ( )
```

Get whether the feature is locked.

Returns

`true` if the feature is locked; otherwise, `false`.

5.16.2.27 IsValid()

```
bool IFloat::IsValid ( )
```

Get whether the feature is valid.

Returns

`true` if the feature is valid; otherwise, `false`.

5.16.2.28 novitec::CameraAPI::GenICam::IFloat()

```
novitec::CameraAPI::CLR::GenICam::IFloat::novitec::CameraAPI::GenICam::IFloat (
            const novitec::CameraAPI::GenICam::IFloat & CIFloat )
```

Initialize a new instance of the IFloat class from the instance of native(C++) IFloat class.

Parameters

CIFloat	The instance of native(C++) IFloat class which to create the new IFloat.

5.16.2.29 SetValue()

```
void IFloat::SetValue (
            double value )
```

Set value of the float type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register write fail.

Parameters

value	The value of the float type feature.

The documentation for this class was generated from the following files:

- GenICam/ITypes.h
- GenICam/ITypes.cpp

5.17 novitec::CameraAPI::CLR::GenICam::IInteger Class Reference

IInteger class.

```
#include <ITypes.h>
```

Public Member Functions

- **IInteger** ()
  Initialize a new instance of the IInteger class.
- novitec::CameraAPI::GenICam::IInteger (const novitec::CameraAPI::GenICam::IInteger &CIInteger)
  Initialize a new instance of the IInteger class from the instance of native(C++) IInteger class.
- ~**IInteger** ()
  Destructor
- novitec::CameraAPI::GenICam::IInteger * GetCIInteger ()
  Get the pointer to the instance of native(C++) IInteger class.
- int64_t GetValue ()
  Get value of the integer type feature.
- int64_t GetCacheValue ()
  Get cached feature value. Returns the stored feature value without attemping to read register.
- void SetValue (int64_t value)
  Set value of the integer type feature.

- bool HasMin ()

  Get whether the feature has minimum value.
- bool HasMax ()

  Get whether the feature has maximum value.
- bool HasInc ()

  Get whether the feature has increment value.
- int64_t GetMin ()

  Get the minimum value of integer type feature.
- int64_t GetMax ()

  Get the maximum value of integer type feature.
- IncMode GetIncMode ()

  Get the increment mode.
- int64_t GetInc ()

  Get the increment value.
- int GetNumberOfValidValues ()

  Get the list of valid values.
- int64_t GetValidValue (int index)

  Get the valid value.
- IntRepresentation GetRepresentation ()

  Get the representation.
- System::String ^ GetUnit ()

  Get the unit.
- System::String ^ GetType ()

  Get the feature type.
- System::String ^ GetTypeName ()

  Get the feature type name.
- System::String ^ GetCategory ()

  Get the category of the feature.
- System::String ^ GetFeatureName ()

  Get the name of feature.
- bool IsValid ()

  Get whether the feature is valid.
- bool IsImplemented ()

  Get whether the feature is implemented.
- bool IsLocked ()

  Get whether the feature is locked.
- AccessMode GetAccessMode ()

  Get the access mode.
- Visibility GetVisibility ()

  Get the visibility.
- System::String ^ GetDisplayName ()

  Get the display name.
- System::String ^ GetToolTip ()

  Get the tooltip.
- System::String ^ GetDescription ()

  Get the description.
- System::String ^ GetElement (System::String^ elementName)

  Get the specified element.

Protected Attributes

- novitec::CameraAPI::GenICam::IInteger * **m_IInteger**

5.17.1 Detailed Description

IInteger class.

5.17.2 Member Function Documentation

5.17.2.1 GetAccessMode()

```
AccessMode IInteger::GetAccessMode ( )
```

Get the access mode.

Returns

AccessMode

5.17.2.2 GetCacheValue()

```
int64_t IInteger::GetCacheValue ( )
```

Get cached feature value. Returns the stored feature value without attemping to read register.

Returns

Cached feature value.

5.17.2.3 GetCategory()

```
System::String IInteger::GetCategory ( )
```

Get the category of the feature.

Returns

A string that represents the category of the feature.

5.17.2.4 GetCIInteger()

```
novitec::CameraAPI::GenICam::IInteger * IInteger::GetCIInteger ( )
```

Get the pointer to the instance of native(C++) IInteger class.

Returns

The pointer to the instance of native(C++) IInteger class.

5.17.2.5 GetDescription()

```
System::String IInteger::GetDescription ( )
```

Get the description.

Returns

A `string` that represents the description.

5.17.2.6 GetDisplayName()

```
System::String IInteger::GetDisplayName ( )
```

Get the display name.

Returns

A `string` that represents the display name.

5.17.2.7 GetElement()

```
System::String IInteger::GetElement (
            System::String^ elementName )
```

Get the specified element.

Parameters

elementName	The pointer to a null-terminated string with value of specified element.

Returns

A `string` that represents the specified element.

5.17.2.8 GetFeatureName()

```
System::String IInteger::GetFeatureName ( )
```

Get the name of feature.

Returns

A string that represents the name of feature.

5.17.2.9 GetInc()

```
int64_t IInteger::GetInc ( )
```

Get the increment value.

[Exception] novitec::CameraAPI::Exception
An exception will be raised if the minimum value is not specified.

Returns

The increment value.

5.17.2.10 GetIncMode()

```
IncMode IInteger::GetIncMode ( )
```

Get the increment mode.

Returns

IncMode

5.17.2.11 GetMax()

```
int64_t IInteger::GetMax ( )
```

Get the maximum value of integer type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be raised if the minimum value is not specified.

Returns

The maximum value of integer type feature.

5.17.2.12 GetMin()

```
int64_t IInteger::GetMin ( )
```

Get the minimum value of integer type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be raised if the minimum value is not specified.

Returns

The minimum value of integer type feature.

5.17.2.13 GetNumberOfValidValues()

```
int IInteger::GetNumberOfValidValues ( )
```

Get the list of valid values.

Returns

Values if the IncMode is IM_List.

5.17.2.14 GetRepresentation()

```
IntRepresentation IInteger::GetRepresentation ( )
```

Get the representation.

Returns

IntRepresentation

5.17.2.15 GetToolTip()

```
System::String IInteger::GetToolTip ( )
```

Get the tooltip.

Returns

A `string` that represents the tooltip.

5.17.2.16 GetType()

```
System::String IInteger::GetType ( )
```

Get the feature type.

Returns

A string that represents the feature type.

5.17.2.17 GetTypeName()

```
System::String IInteger::GetTypeName ( )
```

Get the feature type name.

Returns

A string that represents the feature type name.

5.17.2.18 GetUnit()

```
System::String IInteger::GetUnit ( )
```

Get the unit.

Returns

The unit.

5.17.2.19 GetValidValue()

```
int64_t IInteger::GetValidValue (
            int index )
```

Get the valid value.

Parameters

index	Index of valid values list.

Returns

Valid value.

5.17.2.20 GetValue()

```
int64_t IInteger::GetValue ( )
```

Get value of the integer type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register read fail.

Returns

The value of the integer type feature.

5.17.2.21 GetVisibility()

```
Visibility IInteger::GetVisibility ( )
```

Get the visibility.

Returns

Visibility

5.17.2.22 HasInc()

```
bool IInteger::HasInc ( )
```

Get whether the feature has increment value.

Returns

`true` if the feature has increment value; otherwise, `false`.

5.17.2.23 HasMax()

```
bool IInteger::HasMax ( )
```

Get whether the feature has maximum value.

Returns

`true` if the feature has maximum value; otherwise, `false`.

5.17.2.24 HasMin()

```
bool IInteger::HasMin ( )
```

Get whether the feature has minimum value.

Returns

`true` if the feature has minimum value; otherwise, `false`.

5.17.2.25 IsImplemented()

```
bool IInteger::IsImplemented ( )
```

Get whether the feature is implemented.

Returns

`true` if the feature is implemented; otherwise, `false`.

5.17.2.26 IsLocked()

```
bool IInteger::IsLocked ( )
```

Get whether the feature is locked.

Returns

`true` if the feature is locked; otherwise, `false`.

5.17.2.27 IsValid()

```
bool IInteger::IsValid ( )
```

Get whether the feature is valid.

Returns

`true` if the feature is valid; otherwise, `false`.

5.17.2.28 novitec::CameraAPI::GenICam::IInteger()

```
novitec::CameraAPI::CLR::GenICam::IInteger::novitec::CameraAPI::GenICam::IInteger (
            const novitec::CameraAPI::GenICam::IInteger & CIInteger )
```

Initialize a new instance of the IInteger class from the instance of native(C++) IInteger class.

Parameters

CIInteger	The instance of native(C++) IInteger class which to create the new IInteger.

5.17.2.29 SetValue()

```
void IInteger::SetValue (
            int64_t value )
```

Set value of the integer type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register write fail.

Parameters

value	The value of the integer type feature.

The documentation for this class was generated from the following files:

- GenICam/ITypes.h
- GenICam/ITypes.cpp

5.18 novitec::CameraAPI::CLR::Image Class Reference

The class for image.

```
#include <Image.h>
```

Public Member Functions

- **Image** ()
  Initialize a new instance of the Image class.
- Image (const novitec::CameraAPI::Image &pImage)
  Initialize a new instance of the Image class from the instance of native(C++) Image class.
- ~**Image** ()
  Destructor.
- **!Image** ()
  Finalizer.
- int GetWidth ()
  Get the width of Image.
- int GetHeight ()
  Get the height of Image.
- int GetBPP ()
  Get the bpp of Image.

- unsigned long long GetTimeStamp ()

  Get the timestamp of Image.
- unsigned long long GetFrameNum ()

  Get the frame number of Image.
- array< Byte > ^ GetChunkData ()

  Get the chunk data of Image.
- array< Byte > ^ GetData ()

  Get the Image data.
- unsigned int GetDataSize ()

  Get size of the Image data.
- unsigned int GetChunkSize ()

  Get size of the chunk data.
- PayloadType GetPayloadType ()

  Get payload type.
- PixelFormat GetPixelFormat ()

  Get pixel format.
- novitec::CameraAPI::Image * GetCImage ()

  Get the pointer to the instance of native(C++) Image class.
- void Create (int iWidth, int iHeight, PixelFormat ePixelFormat)

  Create a new Image.
- void Create (int iWidth, int iHeight, PixelFormat ePixelFormat, unsigned int uiChunkSize)

  Create a new Image.
- void CreateJPEG (int iWidth, int iHeight, PixelFormat ePixelFormat, unsigned int uiDataSize)

  Create a new JPEG image.
- void CreateJPEG (int iWidth, int iHeight, PixelFormat ePixelFormat, unsigned int uiDataSize, unsigned int uiChunkSize)

  Create a new JPEG image.
- void Copy (array< Byte >^ pData, int iWidth, int iHeight, PixelFormat ePixelFormat)

  Copy the image from pixel data.
- void Copy (array< Byte >^ pData, unsigned int uiDataSize)

  Copy the image from pixel data.
- void **Release** ()

  Release all resources used by this Image.
- Error ^ Convert (Image^ dst, PixelFormat ePixelFormat)

  Convert the Image.
- Error ^ Convert (Image^ dst, PixelFormat ePixelFormat, DebayerMethod eDebayerMethod)

  Convert the Image.
- Error ^ Save (System::String^ fileName, FileFormat fileFormat)

  Save this image to the specified file in the specified pixel format.
- Error ^ Load (System::String^ fileName)

  Load image from file.

Protected Attributes

- novitec::CameraAPI::Image * **m_Image**

5.18.1 Detailed Description

The class for image.

5.18.2 Constructor & Destructor Documentation

5.18.2.1 Image()

```
Image::Image (
            const novitec::CameraAPI::Image & pImage )
```

Initialize a new instance of the Image class from the instance of native(C++) Image class.

Parameters

pImage	The instance of native(C++) Image class which to create the new Image.

5.18.3 Member Function Documentation

5.18.3.1 Convert() [1/2]

```
Error Image::Convert (
            Image^ dst,
            PixelFormat ePixelFormat )
```

Convert the Image.

Parameters

dst	[out] Destination Image
ePixelFormat	[in] Pixel Format

Returns

novitec::CameraAPI::CLR::Error

5.18.3.2 Convert() [2/2]

```
Error Image::Convert (
            Image^ dst,
            PixelFormat ePixelFormat,
            DebayerMethod eDebayerMethod )
```

Convert the Image.

Parameters

dst	[out] Destination Image
ePixelFormat	[in] Pixel Format
eDebayerMethod	[in] Debayer Method

Returns

novitec::CameraAPI::CLR::Error

5.18.3.3 Copy() [1/2]

```
void Image::Copy (
            array< Byte >^ pData,
            int iWidth,
            int iHeight,
            PixelFormat ePixelFormat )
```

Copy the image from pixel data.

Parameters

pData	Image Data
iWidth	Image Width
iHeight	Image Height
ePixelFormat	Pixel Format

5.18.3.4 Copy() [2/2]

```
void Image::Copy (
            array< Byte >^ pData,
            unsigned int uiDataSize )
```

Copy the image from pixel data.

Parameters

pData	Image Data
uiDataSize	Image Data Size

5.18.3.5 Create() [1/2]

```
void Image::Create (
            int iWidth,
            int iHeight,
            PixelFormat ePixelFormat )
```

Create a new Image.

Parameters

iWidth	Image Width
iHeight	Image Height
ePixelFormat	Pixel Format

5.18.3.6 Create() [2/2]

```
void Image::Create (
            int iWidth,
            int iHeight,
            PixelFormat ePixelFormat,
            unsigned int uiChunkSize )
```

Create a new Image.

Parameters

iWidth	Image Width
iHeight	Image Height
ePixelFormat	Pixel Format
uiChunkSize	Image Chunk Size

5.18.3.7 CreateJPEG() [1/2]

```
void Image::CreateJPEG (
            int iWidth,
            int iHeight,
            PixelFormat ePixelFormat,
            unsigned int uiDataSize )
```

Create a new JPEG image.

Parameters

iWidth	Image Width
iHeight	Image Height
ePixelFormat	Pixel Format
uiDataSize	Image Data Size

5.18.3.8 CreateJPEG() [2/2]

```
void Image::CreateJPEG (
            int iWidth,
            int iHeight,
            PixelFormat ePixelFormat,
            unsigned int uiDataSize,
            unsigned int uiChunkSize )
```

Create a new JPEG image.

Parameters

iWidth	Image Width
iHeight	Image Height
ePixelFormat	Pixel Format
uiDataSize	Image Data Size
uiChunkSize	Image Chunk Size

5.18.3.9 GetBPP()

```
int Image::GetBPP ( )
```

Get the bpp of Image.

Returns

The bpp of Image

5.18.3.10 GetChunkData()

```
array< Byte > Image::GetChunkData ( )
```

Get the chunk data of Image.

Returns

The chunk data of Image

5.18.3.11 GetChunkSize()

```
unsigned int Image::GetChunkSize ( )
```

Get size of the chunk data.

Returns

Size of the chunk data

5.18.3.12 GetCImage()

```
novitec::CameraAPI::Image * Image::GetCImage ( )
```

Get the pointer to the instance of native(C++) Image class.

Returns

The pointer to the instance of native(C++) Image class.

5.18.3.13 GetData()

```
array< Byte > Image::GetData ( )
```

Get the Image data.

Returns

Image Data

5.18.3.14 GetDataSize()

```
unsigned int Image::GetDataSize ( )
```

Get size of the Image data.

Returns

Size of Image data

5.18.3.15 GetFrameNum()

```
unsigned long long Image::GetFrameNum ( )
```

Get the frame number of Image.

Returns

The frame number of Image

5.18.3.16 GetHeight()

```
int Image::GetHeight ( )
```

Get the height of Image.

Returns

The height of Image

5.18.3.17 GetPayloadType()

```
PayloadType Image::GetPayloadType ( )
```

Get payload type.

Returns

Payload Type

5.18.3.18 GetPixelFormat()

```
PixelFormat Image::GetPixelFormat ( )
```

Get pixel format.

Returns

Pixel Format

5.18.3.19 GetTimeStamp()

```
unsigned long long Image::GetTimeStamp ( )
```

Get the timestamp of Image.

Returns

The bpp of Image.

5.18.3.20 GetWidth()

```
int Image::GetWidth ( )
```

Get the width of Image.

Returns

The Width of Image

5.18.3.21 Load()

```
Error Image::Load (
            System::String^ fileName )
```

Load image from file.

Parameters

fileName	A file name in which to load image.

Returns

novitec::CameraAPI::CLR::Error

5.18.3.22 Save()

```
Error Image::Save (
            System::String^ fileName,
            FileFormat fileFormat )
```

Save this image to the specified file in the specified pixel format.

Parameters

fileName	A file name in which to save this Image.
fileFormat	File format.

Returns

novitec::CameraAPI::CLR::Error

The documentation for this class was generated from the following files:

- Image.h
- Image.cpp

5.19 novitec::CameraAPI::CLR::GenICam::IString Class Reference

IString class.

```
#include <ITypes.h>
```

Public Member Functions

- **IString** ()
  Initialize a new instance of the IString class.
- novitec::CameraAPI::GenICam::IString (const novitec::CameraAPI::GenICam::IString &CIString)
  Initialize a new instance of the IString class from the instance of native(C++) IString class.
- ~**IString** ()
  Destructor
- novitec::CameraAPI::GenICam::IString * GetCIString ()
  Get the pointer to the instance of native(C++) IString class.
- System::String ^ GetValue ()
  Get value of the string type feature.
- System::String ^ GetCacheValue ()
  Get cached feature value. Returns the stored feature value without attemping to read register.
- void SetValue (System::String^ value)
  Set value of the string type feature.
- int GetMaxLength ()
  Get the maximum length of the string.
- System::String ^ GetType ()
  Get the feature type.
- System::String ^ GetTypeName ()
  Get the feature type name.
- System::String ^ GetCategory ()
  Get the category of the feature.
- System::String ^ GetFeatureName ()
  Get the name of feature.
- bool IsValid ()
  Get whether the feature is valid.

- bool IsImplemented ()

  Get whether the feature is implemented.
- bool IsLocked ()

  Get whether the feature is locked.
- AccessMode GetAccessMode ()

  Get the access mode.
- Visibility GetVisibility ()

  Get the visibility.
- System::String ^ GetDisplayName ()

  Get the display name.
- System::String ^ GetToolTip ()

  Get the tooltip.
- System::String ^ GetDescription ()

  Get the description.
- System::String ^ GetElement (System::String^ elementName)

  Get the specified element.

Protected Attributes

- novitec::CameraAPI::GenICam::IString * **m_IString**

5.19.1 Detailed Description

IString class.

5.19.2 Member Function Documentation

5.19.2.1 GetAccessMode()

```
AccessMode IString::GetAccessMode ( )
```

Get the access mode.

Returns

AccessMode

5.19.2.2 GetCacheValue()

```
System::String IString::GetCacheValue ( )
```

Get cached feature value. Returns the stored feature value without attemping to read register.

Returns

Cached feature value.

5.19.2.3 GetCategory()

```
System::String IString::GetCategory ( )
```

Get the category of the feature.

Returns

A string that represents the category of the feature.

5.19.2.4 GetCIString()

```
novitec::CameraAPI::GenICam::IString * IString::GetCIString ( )
```

Get the pointer to the instance of native(C++) IString class.

Returns

The pointer to the instance of native(C++) IString class.

5.19.2.5 GetDescription()

```
System::String IString::GetDescription ( )
```

Get the description.

Returns

A `string` that represents the description.

5.19.2.6 GetDisplayName()

```
System::String IString::GetDisplayName ( )
```

Get the display name.

Returns

A `string` that represents the display name.

5.19.2.7 GetElement()

```
System::String IString::GetElement (
            System::String^ elementName )
```

Get the specified element.

Parameters

elementName	The pointer to a null-terminated string with value of specified element.

Returns

A `string` that represents the specified element.

5.19.2.8 GetFeatureName()

```
System::String IString::GetFeatureName ( )
```

Get the name of feature.

Returns

A string that represents the name of feature.

5.19.2.9 GetMaxLength()

```
int IString::GetMaxLength ( )
```

Get the maximum length of the string.

Returns

The maximum length of the string.

5.19.2.10 GetToolTip()

```
System::String IString::GetToolTip ( )
```

Get the tooltip.

Returns

A `string` that represents the tooltip.

5.19.2.11 GetType()

```
System::String IString::GetType ( )
```

Get the feature type.

Returns

A string that represents the feature type.

5.19.2.12 GetTypeName()

```
System::String IString::GetTypeName ( )
```

Get the feature type name.

Returns

A string that represents the feature type name.

5.19.2.13 GetValue()

```
System::String IString::GetValue ( )
```

Get value of the string type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register read fail.

Returns

The value of the string type feature.

5.19.2.14 GetVisibility()

```
Visibility IString::GetVisibility ( )
```

Get the visibility.

Returns

Visibility

5.19.2.15 IsImplemented()

```
bool IString::IsImplemented ( )
```

Get whether the feature is implemented.

Returns

`true` if the feature is implemented; otherwise, `false`.

5.19.2.16 IsLocked()

```
bool IString::IsLocked ( )
```

Get whether the feature is locked.

Returns

`true` if the feature is locked; otherwise, `false`.

5.19.2.17 IsValid()

```
bool IString::IsValid ( )
```

Get whether the feature is valid.

Returns

`true` if the feature is valid; otherwise, `false`.

5.19.2.18 novitec::CameraAPI::GenICam::IString()

```
novitec::CameraAPI::CLR::GenICam::IString::novitec::CameraAPI::GenICam::IString (
            const novitec::CameraAPI::GenICam::IString & CIString )
```

Initialize a new instance of the IString class from the instance of native(C++) IString class.

Parameters

CIString	The instance of native(C++) IString class which to create the new IString.

5.19.2.19 SetValue()

```
void IString::SetValue (
            System::String^ value )
```

Set value of the string type feature.

[Exception] novitec::CameraAPI::Exception
An exception will be rised if the feature register write fail.

Parameters

value	The value of the string type feature.

The documentation for this class was generated from the following files:

- GenICam/ITypes.h
- GenICam/ITypes.cpp

5.20 novitec::CameraAPI::CLR::GenICam::ITypeBase Class Reference

The class on which the supported types are based.

```
#include <ITypes.h>
```

Public Member Functions

- **ITypeBase** ()
  Initialize a new instance of the ITypeBase class.
- ITypeBase (const novitec::CameraAPI::GenICam::ITypeBase CITypeBase)
  Initialize a new instance of the ITypeBase class from the instance of native(C++) ITypeBase class.
- ~**ITypeBase** ()
  Destructor
- novitec::CameraAPI::GenICam::ITypeBase * GetCITypeBase ()
  Get the pointer to the instance of native(C++) ITypeBase class.
- System::String ^ GetType ()
  Get the feature type.
- System::String ^ GetTypeName ()
  Get the feature type name.
- System::String ^ GetCategory ()
  Get the category of the feature.
- System::String ^ GetFeatureName ()
  Get the name of feature.
- bool IsValid ()
  Get whether the feature is valid.
- bool IsImplemented ()
  Get whether the feature is implemented.
- bool IsLocked ()

Get whether the feature is locked.

- AccessMode GetAccessMode ()

  Get the access mode.

- Visibility GetVisibility ()

  Get the visibility.

- System::String ^ GetDisplayName ()

  Get the display name.

- System::String ^ GetToolTip ()

  Get the tooltip.

- System::String ^ GetDescription ()

  Get the description.

- System::String ^ GetElement (System::String^ elementName)

  Get the specified element.

Protected Attributes

- novitec::CameraAPI::GenICam::ITypeBase * **m_ITypeBase**

5.20.1 Detailed Description

The class on which the supported types are based.

5.20.2 Constructor & Destructor Documentation

5.20.2.1 ITypeBase()

```
ITypeBase::ITypeBase (
            const novitec::CameraAPI::GenICam::ITypeBase CITypeBase )
```

Initialize a new instance of the ITypeBase class from the instance of native(C++) ITypeBase class.

Parameters

CITypeBase	The instance of native(C++) ITypeBase class which to create the new ITypeBase.

5.20.3 Member Function Documentation

5.20.3.1 GetAccessMode()

```
AccessMode ITypeBase::GetAccessMode ( )
```

Get the access mode.

Returns

AccessMode

5.20.3.2 GetCategory()

```
System::String ITypeBase::GetCategory ( )
```

Get the category of the feature.

Returns

A string that represents the category of the feature.

5.20.3.3 GetCITypeBase()

```
novitec::CameraAPI::GenICam::ITypeBase * ITypeBase::GetCITypeBase ( )
```

Get the pointer to the instance of native(C++) ITypeBase class.

Returns

The pointer to the instance of natvie(C++) ITypeBase class.

5.20.3.4 GetDescription()

```
System::String ITypeBase::GetDescription ( )
```

Get the description.

Returns

A `string` that represents the description.

5.20.3.5 GetDisplayName()

```
System::String ITypeBase::GetDisplayName ( )
```

Get the display name.

Returns

A `string` that represents the display name.

5.20.3.6 GetElement()

```
System::String ITypeBase::GetElement (
            System::String^ elementName )
```

Get the specified element.

Parameters

elementName	The pointer to a null-terminated string with value of specified element.

Returns

A `string` that represents the specified element.

5.20.3.7 GetFeatureName()

```
System::String ITypeBase::GetFeatureName ( )
```

Get the name of feature.

Returns

A string that represents the name of feature.

5.20.3.8 GetToolTip()

```
System::String ITypeBase::GetToolTip ( )
```

Get the tooltip.

Returns

A `string` that represents the tooltip.

5.20.3.9 GetType()

```
System::String ITypeBase::GetType ( )
```

Get the feature type.

Returns

A string that represents the feature type.

5.20.3.10 GetTypeName()

```
System::String ITypeBase::GetTypeName ( )
```

Get the feature type name.

Returns

A string that represents the feature type name.

5.20.3.11 GetVisibility()

```
Visibility ITypeBase::GetVisibility ( )
```

Get the visibility.

Returns

Visibility

5.20.3.12 IsImplemented()

```
bool ITypeBase::IsImplemented ( )
```

Get whether the feature is implemented.

Returns

`true` if the feature is implemented; otherwise, `false`.

5.20.3.13 IsLocked()

```
bool ITypeBase::IsLocked ( )
```

Get whether the feature is locked.

Returns

`true` if the feature is locked; otherwise, `false`.

5.20.3.14 IsValid()

```
bool ITypeBase::IsValid ( )
```

Get whether the feature is valid.

Returns

`true` if the feature is valid; otherwise, `false`.

The documentation for this class was generated from the following files:

- GenICam/ITypes.h
- GenICam/ITypes.cpp

5.21 novitec::CameraAPI::CLR::MACAddress Struct Reference

MAC Address Structure

```
#include <Types.h>
```

Public Attributes

- array< Byte > ^ **address** = gcnew array<Byte>(6)
  MAC Address

5.21.1 Detailed Description

MAC Address Structure

The documentation for this struct was generated from the following file:

- Types.h

5.22 novitec::CameraAPI::CLR::NetworkAdapterInfo Class Reference

Network Adapter Information

```
#include <Types.h>
```

Public Member Functions

- **NetworkAdapterInfo** ()

  Initialize a new instance of the NetworkAdapterInfo class.
- ~**NetworkAdapterInfo** ()

  Destructor
- novitec::CameraAPI::NetworkAdapterInfo * GetCNetworkAdapterInfo ()

  Get pointer to the native(C++) NetworkAdapterInfo class.
- String ^ GetAdapterName ()

  Get adapter name.
- String ^ GetFriendlyName ()

  Get friendly name.
- String ^ GetDescription ()

  Get description.
- MACAddress ^ GetMACAddress ()

  Get MAC address.
- int GetNumberOfSubnets ()

  Get number of subnets.
- int GetNumberOfGateways ()

  Get number of gateways.
- Subnet ^ GetSubnet (int idx)

  Get subnet of a specific index.
- Gateway ^ GetGateway (int idx)

  Get gateway of a specific index.

Protected Attributes

- novitec::CameraAPI::NetworkAdapterInfo * **m_NetworkAdapterInfo**

5.22.1 Detailed Description

Network Adapter Information

5.22.2 Member Function Documentation

5.22.2.1 GetAdapterName()

```
String NetworkAdapterInfo::GetAdapterName ( )
```

Get adapter name.

Returns

A `string` that represents the adapter name.

5.22.2.2 GetCNetworkAdapterInfo()

```
novitec::CameraAPI::NetworkAdapterInfo * NetworkAdapterInfo::GetCNetworkAdapterInfo ( )
```

Get pointer to the native(C++) NetworkAdapterInfo class.

Returns

Pointer to the native(C++) NetworkAdapterInfo class.

5.22.2.3 GetDescription()

```
String NetworkAdapterInfo::GetDescription ( )
```

Get description.

Returns

A `string` that represents the description.

5.22.2.4 GetFriendlyName()

```
String NetworkAdapterInfo::GetFriendlyName ( )
```

Get friendly name.

Returns

A `string` that represents the friendly name.

5.22.2.5 GetGateway()

```
Gateway NetworkAdapterInfo::GetGateway (
            int idx )
```

Get gateway of a specific index.

Parameters

idx	Index of gateway.

Returns

The `Gateway` that represents Gateway of a specific index.

5.22.2.6 GetMACAddress()

```
MACAddress NetworkAdapterInfo::GetMACAddress ( )
```

Get MAC address.

Returns

MACAddress Structure

5.22.2.7 GetNumberOfGateways()

```
int NetworkAdapterInfo::GetNumberOfGateways ( )
```

Get number of gateways.

Returns

A `integer` that represents the number of gateways.

5.22.2.8 GetNumberOfSubnets()

```
int NetworkAdapterInfo::GetNumberOfSubnets ( )
```

Get number of subnets.

Returns

A `integer` that represents the number of subnets.

5.22.2.9 GetSubnet()

```
Subnet NetworkAdapterInfo::GetSubnet (
            int idx )
```

Get subnet of a specific index.

Parameters

idx	Index of subnet.

Returns

The `Subnet` that represents subnet of a specific index.

The documentation for this class was generated from the following files:

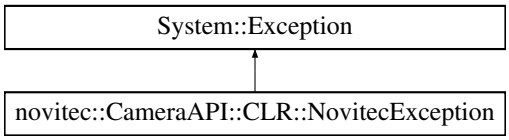
- Types.h
- Types.cpp

5.23 novitec::CameraAPI::CLR::NovitecException Class Reference

The class that throws exception for Novitec Camera API.

`#include <Exception.h>`

Inheritance diagram for novitec::CameraAPI::CLR::NovitecException:



Public Member Functions

- **NovitecException** ()
  Initialize a new instance of the NovitecException class.
- NovitecException (ErrorType code)
  Initialize a new instance of the NovitecException class from the specific ErrorType.
- NovitecException (ErrorType code, System::String^ message)
  Initialize a new instance of the NovitecException class from the specific ErrorType and message.
- ~**NovitecException** ()
  Destructor
- System::String ^ What ()
  Get the error message.
- ErrorType GetType () new
  Get the error type.

Public Attributes

- ErrorType **errorType**
  ErrorType
- System::String ^ **Message**
  Error Message

5.23.1 Detailed Description

The class that throws exception for Novitec Camera API.

5.23.2 Constructor & Destructor Documentation

5.23.2.1 NovitecException() [1/2]

```
NovitecException::NovitecException (
            ErrorType code )
```

Initialize a new instance of the NovitecException class from the specific ErrorType.

Parameters

code	ErrorType

5.23.2.2 NovitecException() [2/2]

```
NovitecException::NovitecException (
            ErrorType code,
            System::String^ message )
```

Initialize a new instance of the NovitecException class from the specific ErrorType and message.

Parameters

code	ErrorType
message	Error Message

5.23.3 Member Function Documentation

5.23.3.1 GetType()

```
ErrorType NovitecException::GetType ( )  [new]
```

Get the error type.

Returns

ErrorType

5.23.3.2 What()

```
System::String NovitecException::What ( )
```

Get the error message.

Returns

Error Message

The documentation for this class was generated from the following files:

- Exception.h
- Exception.cpp

5.24 novitec::CameraAPI::CLR::NU3CameraInfo Class Reference

Novitec USB 3.0 Camera Information.

```
#include <Types.h>
```

Public Member Functions

- **NU3CameraInfo** ()
  Initialize a new instance of the NU3CameraInfo class.
- ~**NU3CameraInfo** ()
  Destructor
- novitec::CameraAPI::NU3CameraInfo * GetCNU3CameraInfo ()
  Get pointer to the native(C++) NU3CameraInfo class.

Public Attributes

- String ^ **manufacturer**
  Device Vendor Name
- String ^ **model_name**
  Device Model Name
- String ^ **serial_number**
  Device Serial Number
- String ^ **device_version**
  Device Version

Protected Attributes

- novitec::CameraAPI::NU3CameraInfo * **m_NU3CameraInfo**

5.24.1 Detailed Description

Novitec USB 3.0 Camera Information.

5.24.2 Member Function Documentation

5.24.2.1 GetCNU3CameraInfo()

```
novitec::CameraAPI::NU3CameraInfo * NU3CameraInfo::GetCNU3CameraInfo ( )
```

Get pointer to the native(C++) NU3CameraInfo class.

Returns

Pointer to the native(C++) NU3CameraInfo class.

The documentation for this class was generated from the following files:

- Types.h
- Types.cpp

5.25 novitec::CameraAPI::CLR::PixelFormatUtil Class Reference

The class which provides the utils for PixelFormat.

```
#include <PixelFormatUtil.h>
```

Static Public Member Functions

- static int GetBitsPerPixel (PixelFormat pixelFormat)
  Get bits-per-pixel from PixelFormat.
- static int GetBytesPerPixel (PixelFormat pixelFormat)
  Gets bytes-per-pixel from PixelFormat.
- static System::String ^ GetPixelFormatName (PixelFormat pixelFormat)
  Get name of PixelFormat.
- static System::String ^ GetPixelFormatDescription (PixelFormat pixelFormat)
  Get description of PixelFormat.
- static bool Is3DFormat (PixelFormat pixelFormat)
  Returns a value that indicates whether the pixel format is type of 3D.

5.25.1 Detailed Description

The class which provides the utils for PixelFormat.

5.25.2 Member Function Documentation

5.25.2.1 GetBitsPerPixel()

```
int PixelFormatUtil::GetBitsPerPixel (
            PixelFormat pixelFormat )  [static]
```

Get bits-per-pixel from PixelFormat.

Parameters

pixelFormat	Pixel Format

Returns

The number of bits-per-pixel.

5.25.2.2 GetBytesPerPixel()

```
int PixelFormatUtil::GetBytesPerPixel (
            PixelFormat pixelFormat )  [static]
```

Gets bytes-per-pixel from PixelFormat.

Parameters

pixelFormat	Pixel Format

Returns

The number of bytes-per-pixel.

5.25.2.3 GetPixelFormatDescription()

```
System::String PixelFormatUtil::GetPixelFormatDescription (
            PixelFormat pixelFormat )  [static]
```

Get description of PixelFormat.

Parameters

pixelFormat	Pixel Format

Returns

A `string` that contains description of PixelFormat.

5.25.2.4 GetPixelFormatName()

```
System::String PixelFormatUtil::GetPixelFormatName (
            PixelFormat pixelFormat )  [static]
```

Get name of PixelFormat.

Parameters

pixelFormat	Pixel Format

Returns

A `string` that contains description of PixelFormat.

5.25.2.5 Is3DFormat()

```
bool PixelFormatUtil::Is3DFormat (
            PixelFormat pixelFormat )  [static]
```

Returns a value that indicates whether the pixel format is type of 3D.

Parameters

pixelFormat	Pixel Format

Returns

`true` if PixelFormat is type of 3D; otherwise, `false`.

The documentation for this class was generated from the following files:

- PixelFormatUtil.h
- PixelFormatUtil.cpp

5.26 novitec::CameraAPI::CLR::Subnet Struct Reference

Subnet Structure

`#include <Types.h>`

Public Attributes

- array< Byte > ^ **address** = gcnew array<Byte>(4)
- array< Byte > ^ **mask** = gcnew array<Byte>(4)

5.26.1 Detailed Description

Subnet Structure

The documentation for this struct was generated from the following file:

- Types.h

5.27 novitec::CameraAPI::CLR::U3VCameraInfo Class Reference

USB3.0 Vision Camera Information

`#include <Types.h>`

Public Member Functions

- **U3VCameraInfo** ()
  Initialize a new instance of the U3VCameraInfo class.
- ~**U3VCameraInfo** ()
  Destructor
- novitec::CameraAPI::U3VCameraInfo * GetCU3VCameraInfo ()
  Get pointer to the native(C++) U3VCameraInfo class.

Public Attributes

- unsigned short **spec_version_major**
  Major Version of USB3 Vision
- unsigned short **spec_version_minor**
  Minor Version of USB3 Vision
- String ^ **manufacturer_name**
  Device Vendor Name
- String ^ **model_name**
  Device Model Name
- String ^ **device_version**
  Device Version
- String ^ **manufacturer_specific_information**
  Manufacturer Specific Information
- String ^ **serial_number**
  Serial Number
- String ^ **user_defined_name**
  User Defined Name

Protected Attributes

- novitec::CameraAPI::U3VCameraInfo * **m_U3VCameraInfo**

5.27.1 Detailed Description

USB3.0 Vision Camera Information

5.27.2 Member Function Documentation

5.27.2.1 GetCU3VCameraInfo()

```
novitec::CameraAPI::U3VCameraInfo * U3VCameraInfo::GetCU3VCameraInfo ( )
```

Get pointer to the native(C++) U3VCameraInfo class.

Returns

Pointer to the native(C++) U3VCameraInfo class.

The documentation for this class was generated from the following files:

- Types.h
- Types.cpp

Chapter 6

File Documentation

6.1 Defs.h File Reference

In this page, enumerations are explained.

Enumerations

- enum class novitec::CameraAPI::CLR::ErrorType {
  NVT_ERROR_UNDEFINED = -1 , NVT_OK , NVT_ERROR_FAILED , NVT_ERROR_TIMEOUT ,
  NVT_ERROR_NO_IMAGE , NVT_ERROR_NOT_IMPLEMENTED , NVT_ERROR_NOT_OPENED ,
  NVT_ERROR_NOT_STARTED ,
  NVT_ERROR_ALREADY_OPENED , NVT_ERROR_ALREADY_STARTED , NVT_ERROR_FAILED_TO_OPEN
  , NVT_ERROR_FAILED_TO_START ,
  NVT_ERROR_FAILED_TO_GET_HANDLE , NVT_ERROR_INVALID_HANDLE , NVT_ERROR_FAILED_TO_ACCESS_REGIS
  , NVT_ERROR_INVALID_REGISTER_ADDRESS ,
  NVT_ERROR_FORMAT_NOT_SUPPORTED , NVT_ERROR_INVALID_FORMAT , NVT_ERROR_FAILED_TO_SAVE_IMAGE
  , NVT_ERROR_INVALID_CHANNEL_NUMBER ,
  NVT_ERROR_ABS_CONTROL_NOT_SUPPORTED , NVT_ERROR_IDLE_STATE , NVT_ERROR_INVALID_INTERFACE
  , NVT_ERROR_REG_INVALID_RESPONSE ,
  NVT_ERROR_REG_FAULT , NVT_ERROR_FAILED_TO_CREATE_SOCKET , NVT_ERROR_INVALID_ARGUMENT
  , NVT_ERROR_FAILED_TO_LOAD_IMAGE ,
  NVT_ERROR_FAILED_TO_LOAD_XML , NVT_ERROR_FAILED_TO_EXTRACT_XML_FILE , **NVT_↩**
  ERROR_INVALID_VALUE , NVT_ERROR_BUFFER_TOO_SMALL ,
  NVT_ERROR_INCOMPLETE_IMAGE , NVT_ERROR_INVALID_OPERATION , NVT_ERROR_INVALID_CAMERA_INTERFAC
  = 0x10000000 , NVT_ERROR_FEATURE_NOT_FOUND ,
  NVT_ERROR_FEATURE_TYPE_MISMATCH , NVT_ERROR_ELEMENT_NOT_FOUND , NVT_ERROR_XML_PARSE_ERRO
  , NVT_ERROR_GVCP_NOT_IMPLEMENTED = 0x20008001 ,
  NVT_ERROR_GVCP_INVALID_PARAMETER = 0x20008002 , NVT_ERROR_GVCP_STATUS_INVALID_ADDRESS
  = 0x20008003 , NVT_ERROR_GVCP_STATUS_WRITE_PROTECT = 0x20008004 , NVT_ERROR_GVCP_STATUS_BAD_ALI
  = 0x20008005 ,
  NVT_ERROR_GVCP_STATUS_ACCESS_DENIED = 0x20008006 , NVT_ERROR_GVCP_STATUS_BUSY
  = 0x20008007 , NVT_ERROR_GVCP_STATUS_PACKET_UNAVAILABLE = 0x2000800c , NVT_ERROR_GVCP_STATUS_DAT
  = 0x2000800d ,
  NVT_ERROR_GVCP_STATUS_INVALID_HEADER = 0x2000800e , NVT_ERROR_GVCP_STATUS_PACKET_NOT_YET_AVA
  = 0x20008010 , NVT_ERROR_GVCP_STATUS_PACKET_AND_PREV_REMOVED_FROM_MEMORY =
  0x20008011 , NVT_ERROR_GVCP_STATUS_PACKET_REMOVED_FROM_MEMORY = 0x20008012 ,
  NVT_ERROR_GVCP_STATUS_NO_REF_TIME = 0x20008013 , NVT_ERROR_GVCP_STATUS_PACKET_TEMPORARILY_U
  = 0x20008014 , NVT_ERROR_GVCP_STATUS_OVERFLOW = 0x20008015 , NVT_ERROR_GVCP_STATUS_ACTION_LATE
  = 0x20008016 ,
  NVT_ERROR_GVCP_STATUS_ERROR = 0x20008fff }

Error Type Enumeration

- enum class novitec::CameraAPI::CLR::AccessMode { AM_UNKNOWN = 0 , AM_NONE , AM_READ_ONLY , AM_CONTROL , AM_EXCLUSIVE }

Access Mode Enumeration

- enum class novitec::CameraAPI::CLR::DeviceEvent { NVT_DEVICE_DISCONNECTED = 0x40000001 , NVT_DEVICE_HEARTBEAT_TIMEOUT = 0x20000001 , NVT_DEVICE_FIRMWARE_IS_UPDATING = 0x10001000 , NVT_DEVICE_IO_OPERATION_ABORTED = 995 }

Device Event Enumeration

- enum class novitec::CameraAPI::CLR::SeverityLevel { SL_INFO = 0x10000000 , SL_WARNING = 0x20000000 , SL_ERROR = 0x40000000 , SL_FATAL = 0x80000000 }

Severity Level Enumeration

- enum class novitec::CameraAPI::CLR::InterfaceType { IF_UNDEFINED = 0 , U3V , GEV , ONVIF = 0xF000 , NOVITEC_PROTOCOL , NU3 , RTSP }

Interface Type Enumeration

- enum class novitec::CameraAPI::CLR::BufferMode { BM_DIRECT = 0 , BM_BUFFERED }

Buffer Mode Enumeration

- enum class novitec::CameraAPI::CLR::StreamProtocol { SP_GVSP = 0 , SP_TCP }

Stream Protocol Enumeration

- enum class novitec::CameraAPI::CLR::PayloadType { PT_UNKNOWN = 0 , PT_IMAGE = 1 , PT_RAW_DATA = 2 , PT_FILE = 3 , PT_JPEG = 5 , PT_JPEG2000 = 6 , PT_H264 = 7 , PT_CHUNK_ONLY = 8 }

Payload Type Enumeration

- enum class novitec::CameraAPI::CLR::PixelFormat { Mono1p = 0x01010037 , Mono2p = 0x01020038 , Mono4p = 0x01040039 , Mono8 = 0x01080001 , Mono8s = 0x01080002 , Mono10 = 0x01100003 , Mono10p = 0x010A0046 , Mono12 = 0x01100005 , Mono12p = 0x010C0047 , Mono14 = 0x01100025 , Mono14p = 0x010E0104 , Mono16 = 0x01100007 , BayerBG4p = 0x01040110 , BayerBG8 = 0x0108000B , BayerBG10 = 0x0110000F , BayerBG10p = 0x010A0052 , BayerBG12 = 0x01100013 , BayerBG12p = 0x010C0053 , BayerBG14 = 0x0110010C , BayerBG14p = 0x010E0108 , BayerBG16 = 0x01100031 , BayerGB4p = 0x0104010F , BayerGB8 = 0x0108000A , BayerGB10 = 0x0110000E , BayerGB10p = 0x010A0054 , BayerGB12 = 0x01100012 , BayerGB12p = 0x010C0055 , BayerGB14 = 0x0110010B , BayerGB14p = 0x010E0107 , BayerGB16 = 0x01100030 , BayerGR4p = 0x0104010D , BayerGR8 = 0x01080008 , BayerGR10 = 0x0110000C , BayerGR10p = 0x010A0056 , BayerGR12 = 0x01100010 , BayerGR12p = 0x010C0057 , BayerGR14 = 0x01100109 , BayerGR14p = 0x010E0105 , BayerGR16 = 0x0110002E , BayerRG4p = 0x0104010E , BayerRG8 = 0x01080009 , BayerRG10 = 0x0110000D , BayerRG10p = 0x010A0058 , BayerRG12 = 0x01100011 , BayerRG12p = 0x010C0059 , BayerRG14 = 0x0110010A , BayerRG14p = 0x010E0106 , BayerRG16 = 0x0110002F , RGBa8 = 0x02200016 , RGBa10 = 0x0240005F , RGBa10p = 0x02280060 , RGBa12 = 0x02400061 , RGBa12p = 0x02300062 , RGBa14 = 0x02400063 , RGBa16 = 0x02400064 , RGB8 = 0x02180014 , RGB8_Planar = 0x02180021 , RGB10 = 0x02300018 , RGB10_Planar = 0x02300022 , RGB10p = 0x021E005C , RGB10p32 = 0x0220001D , RGB12 = 0x0230001A , RGB12_Planar = 0x02300023 , RGB12p = 0x0224005D , RGB14 = 0x0230005E , RGB16 = 0x02300033 , RGB16_Planar = 0x02300024 , RGB565p = 0x02100035 , BGRa8 = 0x02200017 , BGRa10 = 0x0240004C , BGRa10p = 0x0228004D , BGRa12 = 0x0240004E , BGRa12p = 0x0230004F , BGRa14 = 0x02400050 , BGRa16 = 0x02400051 , BGR8 = 0x02180015 , BGR10 = 0x02300019 , BGR10p = 0x021E0048 , BGR12 = 0x0230001B , BGR12p = 0x02240049 ,

BGR14 = 0x0230004A , BGR16 = 0x0230004B , BGR565p = 0x02100036 , R8 = 0x010800C9 ,
R10 = 0x010A00CA , R12 = 0x010C00CB , R16 = 0x011000CC , G8 = 0x010800CD ,
G10 = 0x010A00CE , G12 = 0x010C00CF , G16 = 0x011000D0 , B8 = 0x010800D1 ,
B10 = 0x010A00D2 , B12 = 0x010C00D3 , B16 = 0x011000D4 , Coord3D_ABC8 = 0x021800B2 ,
Coord3D_ABC8_Planar = 0x021800B3 , Coord3D_ABC10p = 0x021E00DB , Coord3D_ABC10p_Planar =
0x021E00DC , Coord3D_ABC12p = 0x022400DE ,
Coord3D_ABC12p_Planar = 0x022400DF , Coord3D_ABC16 = 0x023000B9 , Coord3D_ABC16_Planar =
0x023000BA , Coord3D_ABC32f = 0x026000C0 ,
Coord3D_ABC32f_Planar = 0x026000C1 , Coord3D_AC8 = 0x021000B4 , Coord3D_AC8_Planar =
0x021000B5 , Coord3D_AC10p = 0x021400F0 ,
Coord3D_AC10p_Planar = 0x021400F1 , Coord3D_AC12p = 0x021800F2 , Coord3D_AC12p_Planar =
0x021800F3 , Coord3D_AC16 = 0x022000BB ,
Coord3D_AC16_Planar = 0x022000BC , Coord3D_AC32f = 0x024000C2 , Coord3D_AC32f_Planar =
0x024000C3 , Coord3D_A8 = 0x010800AF ,
Coord3D_A10p = 0x010A00D5 , Coord3D_A12p = 0x010C00D8 , Coord3D_A16 = 0x011000B6 ,
Coord3D_A32f = 0x012000BD ,
Coord3D_B8 = 0x010800B0 , Coord3D_B10p = 0x010A00D6 , Coord3D_B12p = 0x010C00D9 ,
Coord3D_B16 = 0x011000B7 ,
Coord3D_B32f = 0x012000BE , Coord3D_C8 = 0x010800B1 , Coord3D_C10p = 0x010A00D7 ,
Coord3D_C12p = 0x010C00DA ,
Coord3D_C16 = 0x011000B8 , Coord3D_C32f = 0x012000BF , Confidence1 = 0x010800C4 , Confidence1p
= 0x010100C5 ,
Confidence8 = 0x010800C6 , Confidence16 = 0x011000C7 , Confidence32f = 0x012000C8 , BiColorBGRG8
= 0x021000A6 ,
BiColorBGRG10 = 0x022000A9 , BiColorBGRG10p = 0x021400AA , BiColorBGRG12 = 0x022000AD ,
BiColorBGRG12p = 0x021800AE ,
BiColorRGBG8 = 0x021000A5 , BiColorRGBG10 = 0x022000A7 , BiColorRGBG10p = 0x021400A8 ,
BiColorRGBG12 = 0x022000AB ,
BiColorRGBG12p = 0x021800AC , SCF1WBWG8 = 0x01080067 , SCF1WBWG10 = 0x01100068 ,
SCF1WBWG10p = 0x010A0069 ,
SCF1WBWG12 = 0x0110006A , SCF1WBWG12p = 0x010C006B , SCF1WBWG14 = 0x0110006C ,
SCF1WBWG16 = 0x0110006D ,
SCF1WGWB8 = 0x0108006E , SCF1WGWB10 = 0x0110006F , SCF1WGWB10p = 0x010A0070 ,
SCF1WGWB12 = 0x01100071 ,
SCF1WGWB12p = 0x010C0072 , SCF1WGWB14 = 0x01100073 , SCF1WGWB16 = 0x01100074 ,
SCF1WGWR8 = 0x01080075 ,
SCF1WGWR10 = 0x01100076 , SCF1WGWR10p = 0x010A0077 , SCF1WGWR12 = 0x01100078 ,
SCF1WGWR12p = 0x010C0079 ,
SCF1WGWR14 = 0x0110007A , SCF1WGWR16 = 0x0110007B , SCF1WRWG8 = 0x0108007C ,
SCF1WRWG10 = 0x0110007D ,
SCF1WRWG10p = 0x010A007E , SCF1WRWG12 = 0x0110007F , SCF1WRWG12p = 0x010C0080 ,
SCF1WRWG14 = 0x01100081 ,
SCF1WRWG16 = 0x01100082 , YCbCr8 = 0x0218005B , YCbCr8_CbYCr = 0x0218003A , YCbCr10_CbYCr
= 0x02300083 ,
YCbCr10p_CbYCr = 0x021E0084 , YCbCr12_CbYCr = 0x02300085 , YCbCr12p_CbYCr = 0x02240086 ,
YCbCr411_8 = 0x020C005A ,
YCbCr411_8_CbYYCrYY = 0x020C003C , YCbCr422_8 = 0x0210003B , YCbCr422_8_CbYCrY =
0x02100043 , YCbCr422_10 = 0x02200065 ,
YCbCr422_10_CbYCrY = 0x02200099 , YCbCr422_10p = 0x02140087 , YCbCr422_10p_CbYCrY =
0x0214009A , YCbCr422_12 = 0x02200066 ,
YCbCr422_12_CbYCrY = 0x0220009B , YCbCr422_12p = 0x02180088 , YCbCr422_12p_CbYCrY =
0x0218009C , YCbCr601_8_CbYCr = 0x0218003D ,
YCbCr601_10_CbYCr = 0x02300089 , YCbCr601_10p_CbYCr = 0x021E008A , YCbCr601_12_CbYCr =
0x0230008B , YCbCr601_12p_CbYCr = 0x0224008C ,
YCbCr601_411_8_CbYYCrYY = 0x020C003F , YCbCr601_422_8 = 0x0210003E , YCbCr601_422_8_CbYCrY
= 0x02100044 , YCbCr601_422_10 = 0x0220008D ,
YCbCr601_422_10_CbYCrY = 0x0220009D , YCbCr601_422_10p = 0x0214008E , YCbCr601_422_10p_CbYCrY
= 0x0214009E , YCbCr601_422_12 = 0x0220008F ,

YCbCr601_422_12_CbYCrY = 0x0220009F , YCbCr601_422_12p = 0x02180090 , YCbCr601_422_12p_CbYCrY = 0x021800A0 , YCbCr709_8_CbYCr = 0x02180040 ,
YCbCr709_10_CbYCr = 0x02300091 , YCbCr709_10p_CbYCr = 0x021E0092 , YCbCr709_12_CbYCr = 0x02300093 , YCbCr709_12p_CbYCr = 0x02240094 ,
YCbCr709_411_8_CbYYCrYY = 0x020C0042 , YCbCr709_422_8 = 0x02100041 , YCbCr709_422_8_CbYCrY = 0x02100045 , YCbCr709_422_10 = 0x02200095 ,
YCbCr709_422_10_CbYCrY = 0x022000A1 , YCbCr709_422_10p = 0x02140096 , YCbCr709_422_10p_CbYCrY = 0x021400A2 , YCbCr709_422_12 = 0x02200097 ,
YCbCr709_422_12_CbYCrY = 0x022000A3 , YCbCr709_422_12p = 0x02180098 , YCbCr709_422_12p_CbYCrY = 0x021800A4 , YCbCr2020_8_CbYCr = 0x021800F4 ,
YCbCr2020_10_CbYCr = 0x023000F5 , YCbCr2020_10p_CbYCr = 0x021E00F6 , YCbCr2020_12_CbYCr = 0x023000F7 , YCbCr2020_12p_CbYCr = 0x022400F8 ,
YCbCr2020_411_8_CbYYCrYY = 0x020C00F9 , YCbCr2020_422_8 = 0x021000FA , YCbCr2020_422_8_CbYCrY = 0x021000FB , YCbCr2020_422_10 = 0x022000FC ,
YCbCr2020_422_10_CbYCrY = 0x022000FD , YCbCr2020_422_10p = 0x021400FE , YCbCr2020_422_10p_CbYCrY = 0x021400FF , YCbCr2020_422_12 = 0x02200100 ,
YCbCr2020_422_12_CbYCrY = 0x02200101 , YCbCr2020_422_12p = 0x02180102 , YCbCr2020_422_12p_CbYCrY = 0x02180103 , YUV8_UYV = 0x02180020 ,
YUV411_8_UYYVYY = 0x020C001E , YUV422_8 = 0x02100032 , YUV422_8_UYVY = 0x0210001F , Mono10Packed = 0x010C0004 ,
Mono12Packed = 0x010C0006 , BayerBG10Packed = 0x010C0029 , BayerBG12Packed = 0x010C002D , BayerGB10Packed = 0x010C0028 ,
BayerGB12Packed = 0x010C002C , BayerGR10Packed = 0x010C0026 , BayerGR12Packed = 0x010C002A , BayerRG10Packed = 0x010C0027 ,
BayerRG12Packed = 0x010C002B , RGB10V1Packed = 0x0220001C , RGB12V1Packed = 0x02240034 , YUV420_NV12_deprecated = 0x800C0001 ,
YUV420_NV12 = 0x820C0001 , InvalidPixelFormat = 0 }

Pixel Format Enumeration

- enum class novitec::CameraAPI::CLR::DebayerMethod { HQ_LINEAR = 1 , NEAREST_NEIGHBOR , EDGE_SENSE }

  Debayer Method Enumeration
- enum class novitec::CameraAPI::CLR::FileFormat { RAW = 1 , BMP , JPEG }

  File Format
- enum class novitec::CameraAPI::CLR::PeripheralDeviceType { UNDEFINED_DEVICE = 0 , DEV_UART }

  Peripheral Device Type Enumeration

6.1.1 Detailed Description

In this page, enumerations are explained.

6.1.2 Enumeration Type Documentation

6.1.2.1 AccessMode

```
enum class novitec::CameraAPI::CLR::AccessMode  [strong]
```

Access Mode Enumeration

Enumerator

AM_UNKNOWN	Unknown.
AM_NONE	No access.
AM_READ_ONLY	Read only.
AM_CONTROL	Read and write available.
AM_EXCLUSIVE	Exclusive.

6.1.2.2 BufferMode

`enum class novitec::CameraAPI::CLR::BufferMode [strong]`

Buffer Mode Enumeration

Enumerator

BM_DIRECT	Direct Mode
BM_BUFFERED	Buffered Mode

6.1.2.3 DebayerMethod

`enum class novitec::CameraAPI::CLR::DebayerMethod [strong]`

Debayer Method Enumeration

Enumerator

HQ_LINEAR	HQ Linear
NEAREST_NEIGHBOR	Nearest Neighbor
EDGE_SENSE	Edge Sense

6.1.2.4 DeviceEvent

`enum class novitec::CameraAPI::CLR::DeviceEvent [strong]`

Device Event Enumeration

Enumerator

NVT_DEVICE_DISCONNECTED	Camera is disconnected.
NVT_DEVICE_HEARTBEAT_TIMEOUT	Heartbeat timeout.
NVT_DEVICE_FIRMWARE_IS_UPDATING	Firmware update in progress.
NVT_DEVICE_IO_OPERATION_ABORTED	The I/O operation has been aborted because of either a thread exit or an application request.

6.1.2.5 ErrorType

```
enum class novitec::CameraAPI::CLR::ErrorType  [strong]
```

Error Type Enumeration

Enumerator

NVT_ERROR_UNDEFINED	Undefined Error
NVT_OK	Success
NVT_ERROR_FAILED	Failure
NVT_ERROR_TIMEOUT	Timeout. Failed to get an image within the specified duration.
NVT_ERROR_NO_IMAGE	No valid image was available.
NVT_ERROR_NOT_IMPLEMENTED	Not implemented.
NVT_ERROR_NOT_OPENED	Camera is not opened.
NVT_ERROR_NOT_STARTED	Camera is not started.
NVT_ERROR_ALREADY_OPENED	Camera is already opened.
NVT_ERROR_ALREADY_STARTED	Camera is already started.
NVT_ERROR_FAILED_TO_OPEN	Failed to open camera.
NVT_ERROR_FAILED_TO_START	Failed to start camera.
NVT_ERROR_FAILED_TO_GET_HANDLE	Failed to get the handle.
NVT_ERROR_INVALID_HANDLE	Invalid handle.
NVT_ERROR_FAILED_TO_ACCESS_REGISTER	Failed to access register. (Endpoint read/write failure)
NVT_ERROR_INVALID_REGISTER_ADDRESS	Invalid register address.
NVT_ERROR_FORMAT_NOT_SUPPORTED	Unsupported format.
NVT_ERROR_INVALID_FORMAT	Invalid format.
NVT_ERROR_FAILED_TO_SAVE_IMAGE	Failed to save image.
NVT_ERROR_INVALID_CHANNEL_NUMBER	Invalid channel number.
NVT_ERROR_ABS_CONTROL_NOT_SUPPORTED	ABS control not supported.
NVT_ERROR_IDLE_STATE	State idle.
NVT_ERROR_INVALID_INTERFACE	Invalid interface.
NVT_ERROR_REG_INVALID_RESPONSE	Invalid response received during register control. (Invalid checksum)
NVT_ERROR_REG_FAULT	Received fault response while register control. (Failed to control the sensor or an invalid input is used.)
NVT_ERROR_FAILED_TO_CREATE_SOCKET	Failed to create socket.
NVT_ERROR_INVALID_ARGUMENT	Invalid argument.
NVT_ERROR_FAILED_TO_LOAD_IMAGE	Failed to load image.
NVT_ERROR_FAILED_TO_LOAD_XML	Failed to load XML.
NVT_ERROR_FAILED_TO_EXTRACT_XML_FILE	Failed to extract camera description file.
NVT_ERROR_BUFFER_TOO_SMALL	Buffer size is too small.
NVT_ERROR_INCOMPLETE_IMAGE	Incomplete image.
NVT_ERROR_INVALID_OPERATION	Invalid operation.
NVT_ERROR_INVALID_CAMERA_INTERFACE	Invalid camera interface.
NVT_ERROR_FEATURE_NOT_FOUND	Failed to find the feature.
NVT_ERROR_FEATURE_TYPE_MISMATCH	Failed to match feature type.
NVT_ERROR_ELEMENT_NOT_FOUND	Failed to find the element.

Enumerator

NVT_ERROR_XML_PARSE_ERROR	Failed to parse register description xml file.
NVT_ERROR_GVCP_NOT_IMPLEMENTED	[GVCP] Command is not supported by the device.
NVT_ERROR_GVCP_INVALID_PARAMETER	[GVCP] Invalid parameter.
NVT_ERROR_GVCP_STATUS_INVALID_ADDRESS	[GVCP] Invalid address.
NVT_ERROR_GVCP_STATUS_WRITE_PROTECT	[GVCP] Write is protected.
NVT_ERROR_GVCP_STATUS_BAD_ALIGNMENT	[GVCP] A badly aligned address offset or data size was specified.
NVT_ERROR_GVCP_STATUS_ACCESS_DENIED	[GVCP] Access denied.
NVT_ERROR_GVCP_STATUS_BUSY	[GVCP] A required resource to service the request is not currently available.
NVT_ERROR_GVCP_STATUS_PACKET_↩ UNAVAILABLE	[GVCP] Packet is unavailable.
NVT_ERROR_GVCP_STATUS_DATA_OVERRN	[GVCP] Internal memory of GVSP transmitter overrun.
NVT_ERROR_GVCP_STATUS_INVALID_HEADER	[GVCP] The message header is not valid.
NVT_ERROR_GVCP_STATUS_PACKET_NOT_↩ YET_AVAILABLE	[GVCP] The request packet has not yet been acquired.
NVT_ERROR_GVCP_STATUS_PACKET_AND_↩ PREV_REMOVED_FROM_MEMORY	[GVCP] The requested packet and all previous ones are not available anymore and have been discarded from the GVSP transmitter memory.
NVT_ERROR_GVCP_STATUS_PACKET_↩ REMOVED_FROM_MEMORY	[GVCP] The requested packet and all previous ones are not available anymore and have been discarded from the GVSP transmitter memory.
NVT_ERROR_GVCP_STATUS_NO_REF_TIME	[GVCP] The device is not synchronized to a master clock to be used as time reference.
NVT_ERROR_GVCP_STATUS_PACKET_↩ TEMPORARILY_UNAVAILABLE	[GVCP] Packet is temporarily unavailable.
NVT_ERROR_GVCP_STATUS_OVERFLOW	[GVCP] A device queue or packet data has overflowed.
NVT_ERROR_GVCP_STATUS_ACTION_LATE	[GVCP] The requested scheduled action command was requested at a time that is already past.
NVT_ERROR_GVCP_STATUS_ERROR	[GVCP] Generic error.

6.1.2.6 FileFormat

```
enum class novitec::CameraAPI::CLR::FileFormat  [strong]
```

File Format

Enumerator

RAW	RAW
BMP	Bitmap
JPEG	JPEG

6.1.2.7 InterfaceType

```
enum class novitec::CameraAPI::CLR::InterfaceType  [strong]
```

Interface Type Enumeration

Enumerator

IF_UNDEFINED	Undefined Interface
U3V	USB 3.0 Vision
GEV	GigE Vision
ONVIF	ONVIF
NOVITEC_PROTOCOL	Novitec Protocol
NU3	Novitec USB 3.0
RTSP	RTSP

6.1.2.8 PayloadType

```
enum class novitec::CameraAPI::CLR::PayloadType  [strong]
```

Payload Type Enumeration

Enumerator

PT_UNKNOWN	Unknown
PT_IMAGE	Image
PT_RAW_DATA	Raw
PT_FILE	File
PT_JPEG	JPEG
PT_JPEG2000	JPEG2000
PT_H264	H.264
PT_CHUNK_ONLY	Only Chunk

6.1.2.9 PeripheralDeviceType

```
enum class novitec::CameraAPI::CLR::PeripheralDeviceType  [strong]
```

Peripheral Device Type Enumeration

Enumerator

UNDEFINED_DEVICE	Undefined Device
DEV_UART	UART

6.1.2.10 PixelFormat

```
enum class novitec::CameraAPI::CLR::PixelFormat  [strong]
```

Pixel Format Enumeration

Enumerator

Mono1p	Monochrome 1-bit packed
Mono2p	Monochrome 2-bit packed
Mono4p	Monochrome 4-bit packed
Mono8	Monochrome 8-bit
Mono8s	Monochrome 8-bit signed
Mono10	Monochrome 10-bit unpacked
Mono10p	Monochrome 10-bit packed
Mono12	Monochrome 12-bit unpacked
Mono12p	Monochrome 12-bit packed
Mono14	Monochrome 14-bit unpacked
Mono14p	Monochrome 14-bit packed
Mono16	Monochrome 16-bit
BayerBG4p	Bayer Blue-Green 4-bit packed
BayerBG8	Bayer Blue-Green 8-bit
BayerBG10	Bayer Blue-Green 10-bit unpacked
BayerBG10p	Bayer Blue-Green 10-bit packed
BayerBG12	Bayer Blue-Green 12-bit unpacked
BayerBG12p	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
BayerGB4p	Bayer Green-Blue 4-bit packed
BayerGB8	Bayer Green-Blue 8-bit
BayerGB10	Bayer Green-Blue 10-bit unpacked
BayerGB10p	Bayer Green-Blue 10-bit packed
BayerGB12	Bayer Green-Blue 12-bit unpacked
BayerGB12p	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
BayerGR4p	Bayer Green-Red 4-bit packed
BayerGR8	Bayer Green-Red 8-bit
BayerGR10	Bayer Green-Red 10-bit unpacked
BayerGR10p	Bayer Green-Red 10-bit packed
BayerGR12	Bayer Green-Red 12-bit unpacked
BayerGR12p	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

Enumerator

BayerRG4p	Bayer Red-Green 4-bit packed
BayerRG8	Bayer Red-Green 8-bit
BayerRG10	Bayer Red-Green 10-bit unpacked
BayerRG10p	Bayer Red-Green 10-bit packed
BayerRG12	Bayer Red-Green 12-bit unpacked
BayerRG12p	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
RGBa8	Red-Green-Blue-alpha 8-bit
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
RGB8	Red-Green-Blue 8-bit
RGB8_Planar	Red-Green-Blue 8-bit planar
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
RGB12	Red-Green-Blue 12-bit unpacked
RGB12_Planar	Red-Green-Blue 12-bit unpacked planar
RGB12p	Red-Green-Blue 12-bit packed
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
BGRa8	Blue-Green-Red-alpha 8-bit
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
BGR8	Blue-Green-Red 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
R8	Red 8-bit
R10	Red 10-bit
R12	Red 12-bit

Enumerator

R16	Red 16-bit
G8	Green 8-bit
G10	Green 10-bit
G12	Green 12-bit
G16	Green 16-bit
B8	Blue 8-bit
B10	Blue 10-bit
B12	Blue 12-bit
B16	Blue 16-bit
Coord3D_ABC8	3D coordinate A-B-C 8-bit
Coord3D_ABC8_Planar	3D coordinate A-B-C 8-bit planar
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_AC8	3D coordinate A-C 8-bit
Coord3D_AC8_Planar	3D coordinate A-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_A8	3D coordinate A 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_B8	3D coordinate B 8-bit
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_C8	3D coordinate C 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
Confidence1	Confidence 1-bit unpacked
Confidence1p	Confidence 1-bit packed
Confidence8	Confidence 8-bit
Confidence16	Confidence 16-bit

Enumerator

Confidence32f	Confidence 32-bit floating point
BiColorBGRG8	Bi-color Blue/Green - Red/Green 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
BiColorRGBG8	Bi-color Red/Green - Blue/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
SCF1WBWG8	Sparse Color Filter #1 White-Blue-White-Green 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
SCF1WGWB8	Sparse Color Filter #1 White-Green-White-Blue 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
SCF1WGWR8	Sparse Color Filter #1 White-Green-White-Red 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
SCF1WRWG8	Sparse Color Filter #1 White-Red-White-Green 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
YCbCr8	YCbCr 4:4:4 8-bit
YCbCr8_CbYCr	YCbCr 4:4:4 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
YCbCr411_8	YCbCr 4:1:1 8-bit
YCbCr411_8_CbYYCrYY	YCbCr 4:1:1 8-bit

Enumerator

YCbCr422_8	YCbCr 4:2:2 8-bit
YCbCr422_8_CbYCrY	YCbCr 4:2:2 8-bit
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
YCbCr601_8_CbYCr	YCbCr 4:4:4 8-bit BT.601
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_8	YCbCr 4:2:2 8-bit BT.601
YCbCr601_422_8_CbYCrY	YCbCr 4:2:2 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
YCbCr709_8_CbYCr	YCbCr 4:4:4 8-bit BT.709
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_8	YCbCr 4:2:2 8-bit BT.709
YCbCr709_422_8_CbYCrY	YCbCr 4:2:2 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
YCbCr2020_8_CbYCr	YCbCr 4:4:4 8-bit BT.2020
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

Enumerator

YCbCr2020_411_8_CbYYCrYY	YCbCr 4:1:1 8-bit 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_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
YUV8_UYV	YUV 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
Mono10Packed	GigE Vision specific format, Monochrome 10-bit packed
Mono12Packed	GigE Vision specific format, Monochrome 12-bit packed
BayerBG10Packed	GigE Vision specific format, Bayer Blue-Green 10-bit packed
BayerBG12Packed	GigE Vision specific format, Bayer Blue-Green 12-bit packed
BayerGB10Packed	GigE Vision specific format, Bayer Green-Blue 10-bit packed
BayerGB12Packed	GigE Vision specific format, Bayer Green-Blue 12-bit packed
BayerGR10Packed	GigE Vision specific format, Bayer Green-Red 10-bit packed
BayerGR12Packed	GigE Vision specific format, Bayer Green-Red 12-bit packed
BayerRG10Packed	GigE Vision specific format, Bayer Red-Green 10-bit packed
BayerRG12Packed	GigE Vision specific format, Bayer Red-Green 12-bit packed
RGB10V1Packed	GigE Vision specific format, Red-Green-Blue 10-bit packed - variant 1
RGB12V1Packed	GigE Vision specific format, Red-Green-Blue 12-bit packed - variant 1
YUV420_NV12_deprecated	deprecated pixel format. it's not have component info. please use 0x820C0001
YUV420_NV12	NOVITEC custom format, YUV420 NV12
InvalidPixelFormat	Invalid Pixel Format

6.1.2.11 SeverityLevel

```
enum class novitec::CameraAPI::CLR::SeverityLevel  [strong]
```

Severity Level Enumeration

Enumerator

SL_INFO	Info
SL_WARNING	Warning
SL_ERROR	Error
SL_FATAL	Fatal

6.1.2.12 StreamProtocol

```
enum class novitec::CameraAPI::CLR::StreamProtocol  [strong]
```

Stream Protocol Enumeration

Enumerator

SP_GVSP	GVSP
SP_TCP	TCP

6.2 GenICam/defs.h File Reference

In this page, enumerations for GenICam are explained.

Typedefs

- typedef System::IntPtr **novitec::CameraAPI::CLR::GenICam::FeatureHandle**

Enumerations

- enum class novitec::CameraAPI::CLR::GenICam::FeatureType { TUnknown = -1 , TInteger , TFloat , TString , TEnumeration , TCommand , TBoolean , TRegister , TCategory }

  Feature Type Enumeration
- enum class novitec::CameraAPI::CLR::GenICam::AccessMode { RW = 0 , RO , WO }

  Access Mode Enumeration
- enum class novitec::CameraAPI::CLR::GenICam::Visibility { BEGINNER = 0 , EXPERT , GURU , INVISIBLE }

  Visibility Enumeration
- enum class novitec::CameraAPI::CLR::GenICam::IntRepresentation { IR_Linear = 0 , IR_Logarithmic , IR_Boolean , IR_PureNumber , IR_HexNumber , IR_IPV4Address , IR_MACAddress }

  Integer Representation Enumeration
- enum class novitec::CameraAPI::CLR::GenICam::FloatRepresentation { FR_Linear = 0 , FR_Logarithmic , FR_PureNumber }

  Float Represesntation Enumeration
- enum class novitec::CameraAPI::CLR::GenICam::IncMode { IM_Linear , IM_Step , IM_List }

  Increment Mode Enumeration
- enum class novitec::CameraAPI::CLR::GenICam::Endianess { LittleEndian = 0 , BigEndian }

  Endianess Enumeration
- enum class novitec::CameraAPI::CLR::GenICam::Sign { Unsigned = 0 , Signed }

  Sign Enumeration

6.2.1 Detailed Description

In this page, enumerations for GenICam are explained.

6.2.2 Enumeration Type Documentation

6.2.2.1 AccessMode

`enum class novitec::CameraAPI::CLR::GenICam::AccessMode [strong]`

Access Mode Enumeration

Enumerator

RW	Read Write
RO	Read Only
WO	Write Only

6.2.2.2 Endianess

`enum class novitec::CameraAPI::CLR::GenICam::Endianess [strong]`

Endianess Enumeration

Enumerator

LittleEndian	Little Endian
BigEndian	Big Endian

6.2.2.3 FeatureType

`enum class novitec::CameraAPI::CLR::GenICam::FeatureType [strong]`

Feature Type Enumeration

Enumerator

TUnknown	Unknown
TInteger	Integer
TFloat	Float

Enumerator

TString	String
TEnumeration	Enumeration
TCommand	Command
TBoolean	Boolean
TRegister	Register
TCategory	Category

6.2.2.4 FloatRepresentation

`enum class novitec::CameraAPI::CLR::GenICam::FloatRepresentation [strong]`

Float Represesntation Enumeration

Enumerator

FR_Linear	Linear
FR_Logarithmic	Logarithmic
FR_PureNumber	Pure Number

6.2.2.5 IncMode

`enum class novitec::CameraAPI::CLR::GenICam::IncMode [strong]`

Increment Mode Enumeration

Enumerator

IM_Linear	Linear
IM_Step	Step
IM_List	List

6.2.2.6 IntRepresentation

`enum class novitec::CameraAPI::CLR::GenICam::IntRepresentation [strong]`

Integer Representation Enumeration

Enumerator

IR_Linear	Linear

Enumerator

IR_Logarithmic	Logarithmic
IR_Boolean	Boolean
IR_PureNumber	Pure Number
IR_HexNumber	Hex Number
IR_IPV4Address	IP Address
IR_MACAddress	MAC Address

6.2.2.7 Sign

`enum class novitec::CameraAPI::CLR::GenICam::Sign [strong]`

Sign Enumeration

Enumerator

Unsigned	Unsigned
Signed	Signed

6.2.2.8 Visibility

`enum class novitec::CameraAPI::CLR::GenICam::Visibility [strong]`

Visibility Enumeration

Enumerator

BEGINNER	Beginner
EXPERT	Expert
GURU	Guru
INVISIBLE	Invisible

Index