

Technical Note

How to Set/Get Camera Parameters in eBUS SDK for JAI

How to Set/Get Camera Parameters in eBUS SDK for JAI

Abstract

The eBUS SDK for JAI supports USB3 Vision and GigE Vision cameras from JAI. Camera parameters may be different for each camera model. This document describes the methods to set/get parameters when using the eBUS SDK for JAI.

1. Method to Set/Get values to/from parameters

The functions in the eBUS SDK for JAI are common for both USB3 Vision and GigE Vision. The following functions can be used to set/get values to/from known parameters.

```
PV_GENICAM_API PvResult GetIntegerValue( const PvString &aName, int64_t &aValue );
PV_GENICAM_API PvResult SetIntegerValue( const PvString &aName, int64_t aValue );
PV_GENICAM_API PvResult GetFloatValue( const PvString &aName, double &aValue );
PV_GENICAM_API PvResult SetFloatValue( const PvString &aName, double aValue );
PV_GENICAM_API PvResult GetEnumValue( const PvString &aName, PvString &aValue );
PV_GENICAM_API PvResult GetEnumValue( const PvString &aName, int64_t &aValue );
PV_GENICAM_API PvResult SetEnumValue( const PvString &aName, const PvString &aValue );
PV_GENICAM_API PvResult SetEnumValue( const PvString &aName, int64_t aValue );
PV_GENICAM_API PvResult GetBooleanValue( const PvString &aName, bool &aValue );
PV_GENICAM_API PvResult SetBooleanValue( const PvString &aName, bool aValue );
PV_GENICAM_API PvResult GetStringValue( const PvString &aName, PvString &aValue );
PV_GENICAM_API PvResult SetStringValue( const PvString &aName, const PvString &aValue );
PV_GENICAM_API PvResult ExecuteCommand( const PvString &aName );
```

Detailed descriptions of each function can be found in the help file of the eBUS SDK for JAI.

To achieve setting/getting values, “aName” (type string) is needed to specify the parameter. Since the actual name of “aName” would vary for each camera model, the SDK manual does not specify the names. Instead, “aName” is defined in the XML file in each camera. The next section shows how you can determine the right value for “aName.”



Technical Note

How to Set/Get Camera Parameters in eBUS SDK for JAI

2. Functions for each type of parameter

2.1. “aName” in eBUS Player

The easiest way to determine the correct “aName” parameter is by using the eBUS Player. After connecting a camera, open “Device Control”. In the Device Control window, click on a parameter and the detail information for that parameter will appear at the bottom of Device Control window.

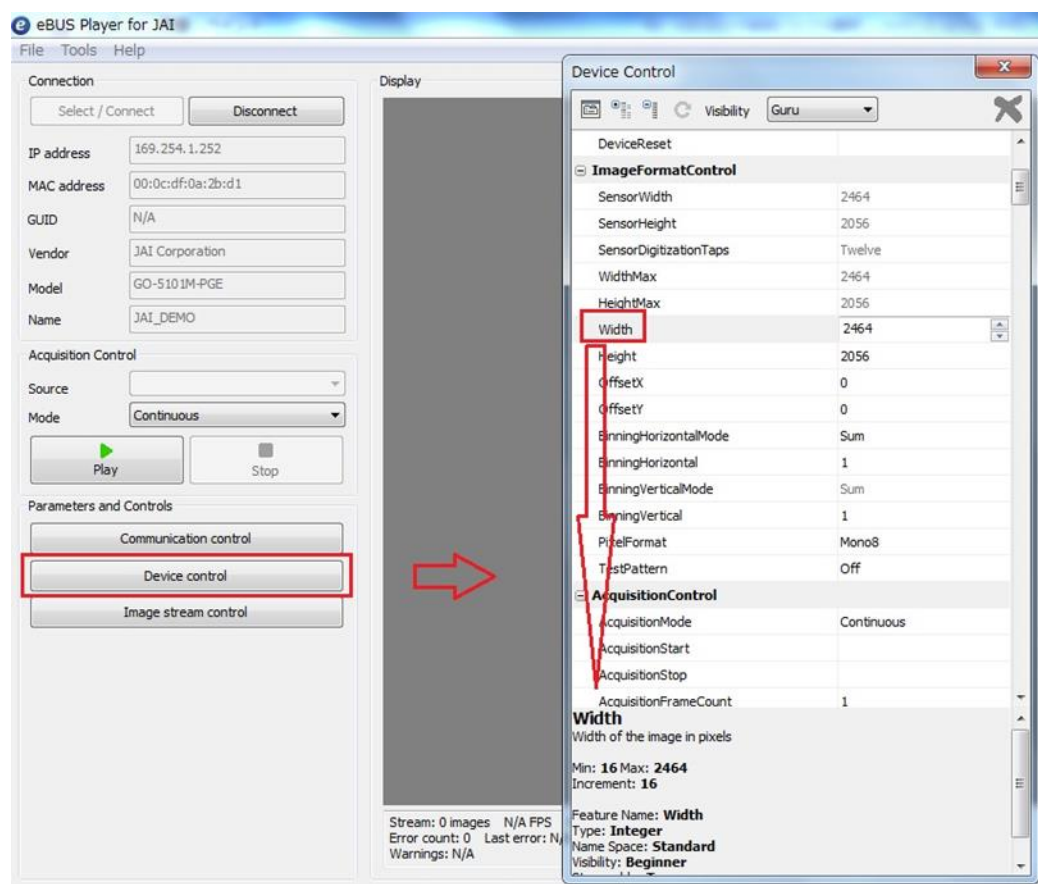


Figure 1

The string provided for “Feature Name” is the “aName” parameter for this feature. Also note the “Type” field. This provides the parameter type and therefore the type of function that must be used to get/set parameters.



Technical Note

How to Set/Get Camera Parameters in eBUS SDK for JAI

2.2. Type: Integer

Figure 1 shows that the parameter “Width” has a type of Integer. For Integer type parameters, the following functions are used:

```
PV_GENICAM_API PvResult GetIntegerValue( const PvString &aName, int64_t &aValue );
```

```
PV_GENICAM_API PvResult SetIntegerValue( const PvString &aName, int64_t aValue );
```

2.3. Type: Float

Figure 2 below shows that the parameter type for AcquisitionFrameRate is Float. For Float type parameters, the following functions are used:

```
PV_GENICAM_API PvResult GetFloatValue( const PvString &aName, double &aValue );
```

```
PV_GENICAM_API PvResult SetFloatValue( const PvString &aName, double aValue );
```

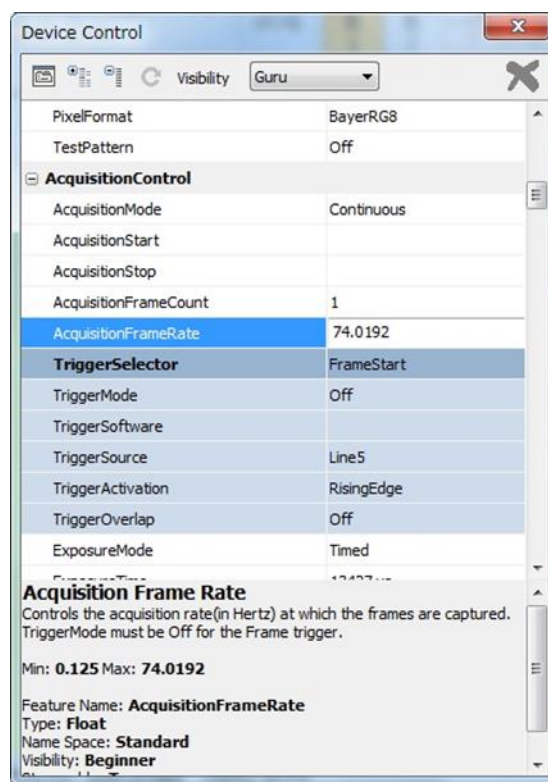


Figure 2



Technical Note

How to Set/Get Camera Parameters in eBUS SDK for JAI

2.4. Type: Enumeration

Figure 3 shows that the parameter type for PixelFormat is Enumeration. For Enumeration type parameters, the following functions are used:

```
PV_GENICAM_API PvResult GetEnumValue( const PvString &aName, PvString &aValue );
PV_GENICAM_API PvResult GetEnumValue( const PvString &aName, int64_t &aValue );
PV_GENICAM_API PvResult SetEnumValue( const PvString &aName, const PvString &aValue );
PV_GENICAM_API PvResult SetEnumValue( const PvString &aName, int64_t aValue );
```

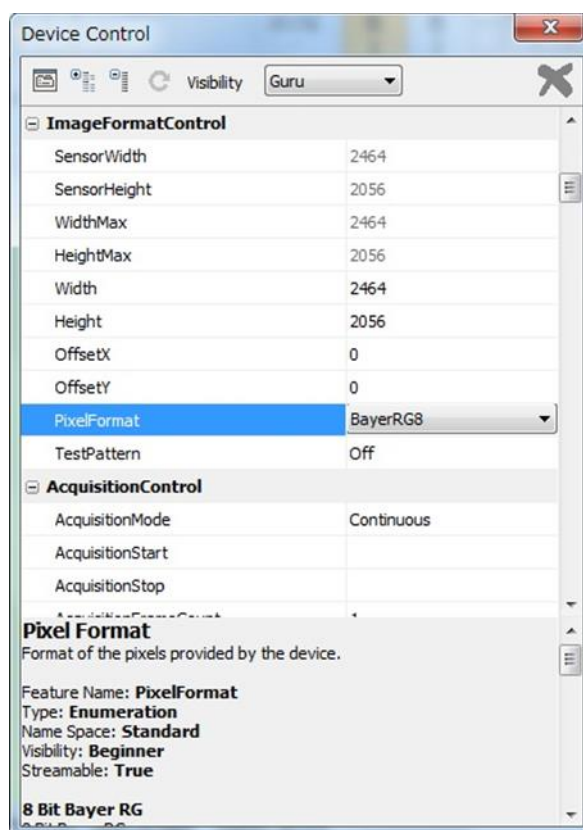


Figure 3



Technical Note

How to Set/Get Camera Parameters in eBUS SDK for JAI

The “aName” value for Enumeration type parameters can be found by selecting the desired enumeration parameter and checking in the “Enum Entry Name” field as shown in Figure 4 below.

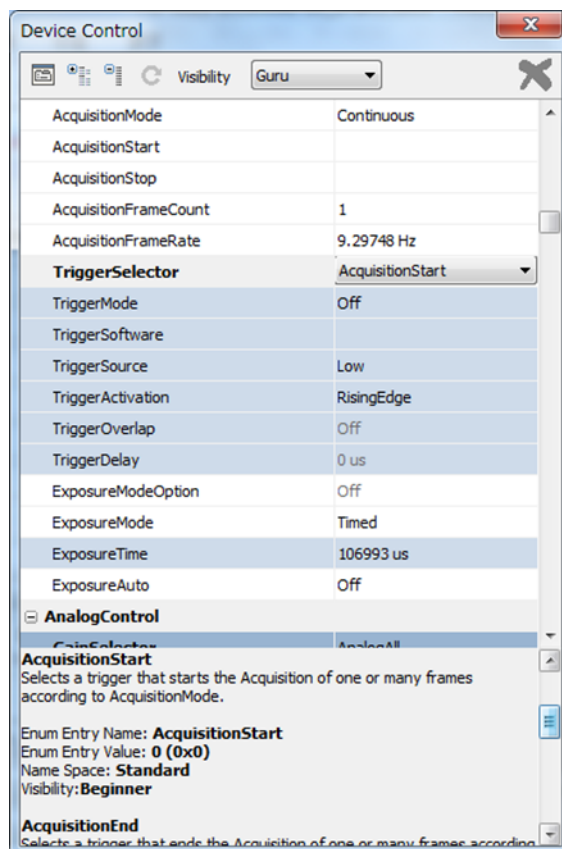


Figure 4



Technical Note

How to Set/Get Camera Parameters in eBUS SDK for JAI

2.5. Type: Boolean

Figure 5 shows that the parameter type for UserOutputValue is Boolean. For Boolean type parameters, the following functions are used:

```
PV_GENICAM_API PvResult GetBooleanValue( const PvString &aName, bool &aValue );
```

```
PV_GENICAM_API PvResult SetBooleanValue( const PvString &aName, bool aValue );
```

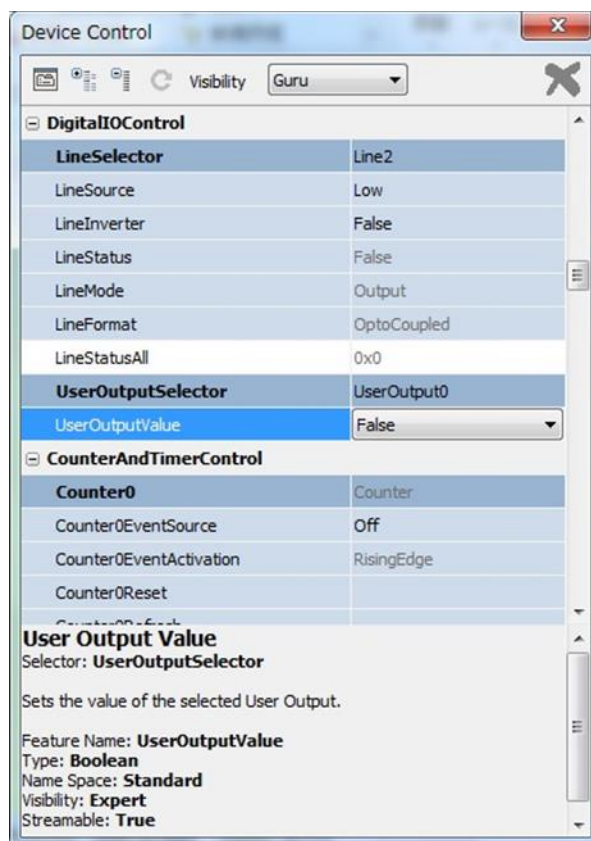


Figure 5

Technical Note

How to Set/Get Camera Parameters in eBUS SDK for JAI

2.6. Type: String

Figure 6 shows that the parameter type for DeviceUserID is String. For String type parameters, the following functions are used:

```
PV_GENICAM_API PvResult GetStringValue( const PvString &aName, PvString &aValue );
```

```
PV_GENICAM_API PvResult SetStringValue( const PvString &aName, const PvString &aValue );
```

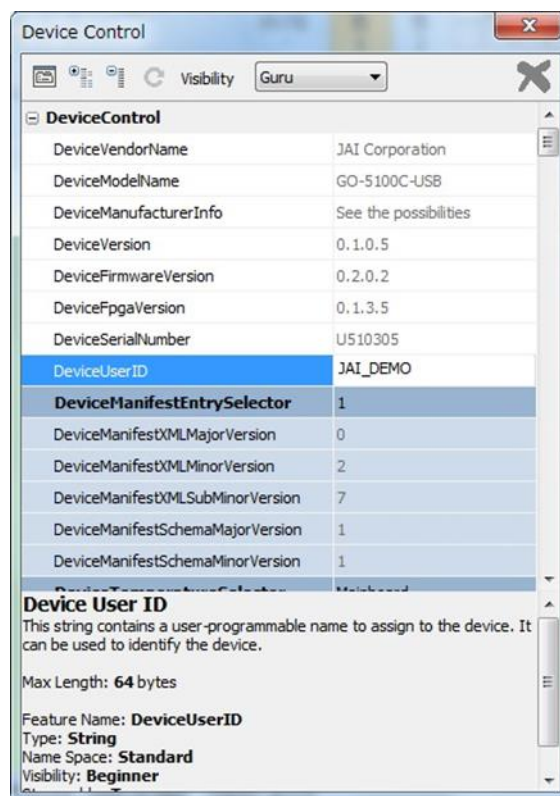


Figure 6

Technical Note

How to Set/Get Camera Parameters in eBUS SDK for JAI

2.7. Type: Command

Figure 7 shows that the parameter type for TriggerSelector is Command. For Command type parameters, the following function is used:

```
PV_GENICAM_API PvResult ExecuteCommand( const PvString &aName );
```

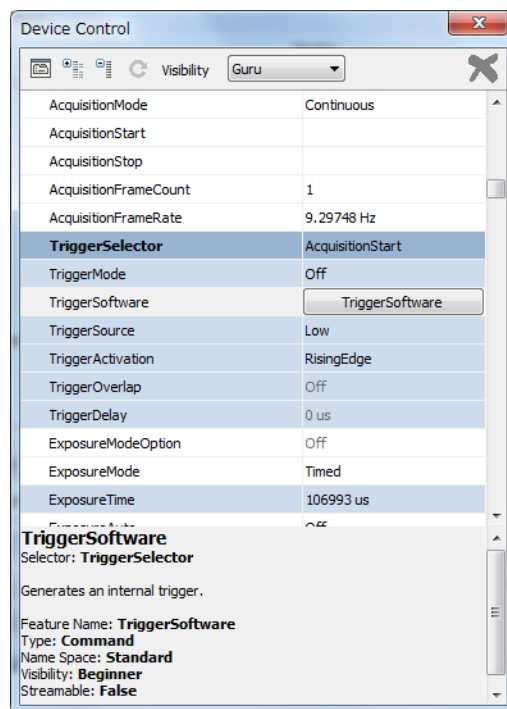


Figure 7

3. Selector parameters

Some parameters have names with “Selector” such as “TriggerSelector” or “UserSetSelector”.

These parameters have associated sub-parameters. In the eBUS Player, Selector parameters are shown with a background color and associated sub-parameters are shown with a background color.

Each main Selector parameter has a list of values which can be selected by the user. Each parameter option for the main Selector parameter is associated with its own set of sub-parameter values. In other words, a set of associated sub-parameter values is maintained for each parameter value option of the main Selector.

For example, “TriggerSelector” has a list of selectable parameter values including “AcquisitionStart”, “FrameStart”, etc. “TriggerSource”, “TriggerMode”, etc. exist as a set of *independent* associated sub-parameters for each of the TriggerSelector options. If you change the “TriggerSource” parameter value associated with “AcquisitionStart”, the “TriggerSource” parameter associated with “FrameStart” is not changed because it has its own independent “TriggerSource” parameter.



Technical Note

How to Set/Get Camera Parameters in eBUS SDK for JAI

For this reason, when setting/getting associated sub-parameters, the main selector parameter must be selected first to make the appropriate sub-parameters available.

In the Device Control window of the eBUS Player, once you select one of the main Selector parameters, the values of the associated sub-parameters are changed automatically to reflect the set of associated sub-parameter values belonging to the selected main parameter.

4. Visibility of parameters in the Device Control window

A visibility level is defined for each camera parameter. Depending on the Visibility setting of the Device Control window, the parameters that are shown will be different. The default Visibility value for the eBUS Player is “Beginner”.

“Expert” can show more parameters. “Guru” shows all parameters.

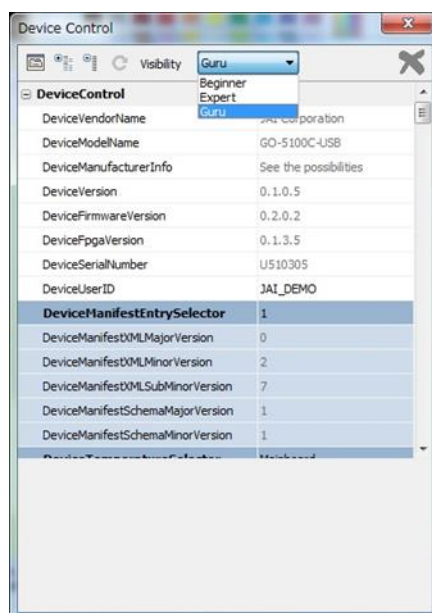


Figure 8



Technical Note

How to Set/Get Camera Parameters in eBUS SDK for JAI

Your preferred Visibility level can be remembered in the eBUS Player by using the command: Tools->Save Preferences.

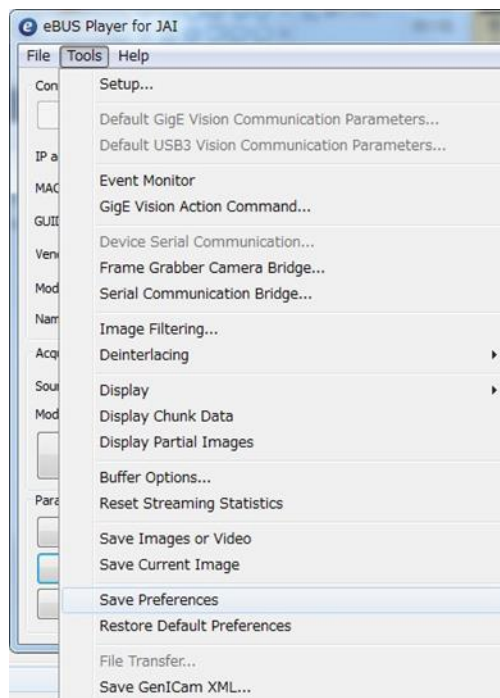


Figure 9

Revision History

[illegible]