Technical Note

How to use Action Commands

How to use Action Commands in GigE Vision cameras:

An Action Command is a feature defined in the GigE Vision standard.

This document describes how to use Action Commands and provides some examples.

1. Basic Concept of the Action Command:

The Action Command is useful to control multiple devices in a common network at the same time. Devices, including cameras, can be divided into groups and a single command can be issued to all devices in a group simultaneously.

Action Commands work by first sending specific "key data" to the devices in a network. Once the key data is correlated to specific devices, a pre-defined action can be activated.

Two actions can be defined for each JAI GigE Vision Camera. Each action can be used as a trigger source for the following functions; Acquisition Start, Acquisition Stop, Frame Start, or Transfer Start. To use an Action Command, the trigger source of a function must be set to Action1 or Action2.

2. Settings for Devices receiving an Action Command:

The following parameters should be set for each camera or device that you want to control with an Action Command:

ActionDeviceKey:

ActionDeviceKey is a kind of password which enables Action Commands to be used securely. Devices react to Action Commands that include the correct ActionDeviceKey. The ActionDeviceKey cannot be read out from other devices because it is write-only. The user needs to memorize the key and the same key must be used within the software program. When an Action Command is received, the ActionDeviceKey is the first item checked.

ActionSelector:

JAI GigE Vision cameras support two Action Commands: Action1 and Action 2. If receiving an Action Command which is dedicated to Action1, a function whose trigger source is Action1 is activated. If receiving an Action command which is dedicated to Action2, a function whose trigger source is Action2 is activated.

ActionGroupKey

A key (value) used to activate an action. Once the ActionDeviceKey matches, then the ActionGroupKey is checked. This enables an Action Command to be applied to specific subsets of devices.

ActionGroupMask

A mask for grouping the Actions. Once the ActionGroupKey matches, the Action command is finally checked to see whether the Action Command's group mask matches the pre-defined ActionGroupMask in the device. If a Logical AND of the two masks results in True, the Action Command is determined to be applicable to the group and its related function is activated.

Fig. 1: Action Command Parameters

3. Examples:

This section describes some examples using two GigE Vision cameras and three GigE Vision strobe controllers. All devices support Action Commands. Each strobe controller has a different light source; IR, UV, and white visible light. Examples in this section trigger a camera and selected strobe at the same time using Action Commands.

Settings for each device are as follows:

@ Camera1:

// Trigger source for FrameStart: Action 1

TriggerSelector = FrameStart

TriggerMode = ON

TriggerSource = Action1

//Settings for Action1

ActionDeviceKey = 0x12345678

ActionSelector = 1

ActionGroupKey[1] = 0x00000001

ActionGroupMask[1] = 0x00000001

//Trigger source for AcquisitionEnd: Action 2

TriggerSelector = AcquisitionEnd

TriggerMode = ON

TriggerSource = Action2

//Settings for Action2

ActionDeviceKey = 0x12345678

ActionSelector = 2

ActionGroupKey[2] = 0x88888888

ActionGroupMask[2] = 0x00001000

@ Camera2:

// Trigger source for FrameStart: Action 2

TriggerSelector = FrameStart

TriggerMode = ON

TriggerSource = Action2

//Settings for Action2

ActionDeviceKey = 0x12345678

ActionSelector = 2

ActionGroupKey[2] = 0x00000001

ActionGroupMask[2] = 0x00000002

// Trigger source for AcquisitionEnd: Action 1

TriggerSelector = AcquisitionEnd

TriggerMode = ON

TriggerSource = Action1

//Settings for Action1

ActionDeviceKey = 0x12345678

ActionSelector = 1

ActionGroupKey[1] = 0x88888888

ActionGroupMask[1] = 0x00002000

@ IR Strobe:

//Trigger Strobe with Action1

TriggerSelector = Strobe

TriggerMode = ON

TriggerSource = Action1

//Action1 Command

ActionDeviceKey = 0x12345678

ActionSelector = 1

ActionGroupKey[1] = 0x00000001

ActionGroupMask[1] = 0x00010000

@ UV Strobe:

//Trigger Strobe with Action5

TriggerSelector = Strobe

TriggerMode = ON

TriggerSource = Action5

//Action5 Command

ActionDeviceKey = 0x12345678

ActionSelector = 5

ActionGroupKey[5] = 0x00000001

ActionGroupMask[5] = 0x00020000

@ White Visible Strobe:

//Trigger Strobe with Action1

TriggerSelector = Strobe

TriggerMode = ON

TriggerSource = Action1

//Action1 Command

ActionDeviceKey = 0x12345678

ActionSelector = 1

ActionGroupKey[1] = 0x00000001

ActionGroupMask[1] = 0x00040000

Note)

- ActionDeviceKey is common for all 5 devices enabling them to react to Action Commands.
- The number of the above Actions is just an example. It is up to each device how many Actions the device supports. JAI GigE Vision cameras support two Actions.

3.1 Trigger to Camera1 and white visible strobe:

Activate only Camera1 and the white visible strobe. No other devices are activated. Here is sample code created using the JAI SDK:

```
#define DEVICE_KEY 0x12345678

#define GROUP_KEY 0x00000001

//Bits calculation: 0x00000001|0x00040000 = 0x00040001

J_Factory_SendActionCommand(hFactory, DEVICE_KEY, GROUP_KEY, 0x00040001);
```

3.2 Trigger to Camera1, Camera2, and UV strobe:

```
#define DEVICE_KEY 0x12345678

#define GROUP_KEY 0x00000001

//Bits calculation: 0x00000001|0x00000002|0x00020000 = 0x00020003

J_Factory_SendActionCommand(hFactory, DEVICE_KEY, GROUP_KEY, 0x00020003);
```

3.3 Trigger to all cameras and strobes:

```
#define DEVICE_KEY 0x12345678

#define GROUP_KEY 0x00000001

J_Factory_SendActionCommand(hFactory, DEVICE_KEY, GROUP_KEY, 0xFFFFFFFF);
```

3.4 Trigger to all cameras:

```
#define DEVICE_KEY 0x12345678

#define GROUP_KEY 0x00000001

//Bits calculation: 0x00000001|0x00000002 = 0x00000003

J_Factory_SendActionCommand(hFactory, DEVICE_KEY, GROUP_KEY, 0x00000003);
```

3.5 Trigger to Camera2 only:

```
#define DEVICE_KEY 0x12345678

#define GROUP_KEY 0x00000001

J_Factory_SendActionCommand(hFactory, DEVICE_KEY, GROUP_KEY, 0x00000002);
```

3.6 Stop acquisition of Camera1:

```
#define DEVICE_KEY 0x12345678

#define GROUP_KEY_STOP 0x88888888
```

J_Factory_SendActionCommand(hFactory, DEVICE_KEY, GROUP_KEY_STOP, 0x00001000);

3.7 Stop Acquisition of Camera1 and Camera2:

#define DEVICE_KEY 0x12345678

#define GROUP_KEY_STOP 0x88888888

//Bits calculation: 0x00001000|0x00002000= 0x00003000

J_Factory_SendActionCommand(hFactory, DEVICE_KEY, GROUP_KEY_STOP, 0x00003000);

4. Scheduled Action Command:

IEEE1588-2008 standard defines a Precision Time Protocol (PTP) for synchronizing multiple devices in a network. Using PTP, multiple devices can be synchronized to the same Epoch Time (UNIX time). Moreover, the standard defines the concept of scheduled Action Commands. In this way, multiple cameras (synchronized by PTP) can execute a scheduled action at the same scheduled time (Epoch Time) with a clock accuracy in the nanosecond range.

This section describes how to set up a scheduled Action Command using the eBus Player for JAI application. For more details regarding the protocol of PTP and scheduled Action Commands, refer to the IEEE 1588-2008 standard document.

JAI cameras work as PTP slaves. At least one PTP server must exist in the network for PTP synchronization to work.
An example of a configuration with a PTP server, host PC, and two cameras is as follows:

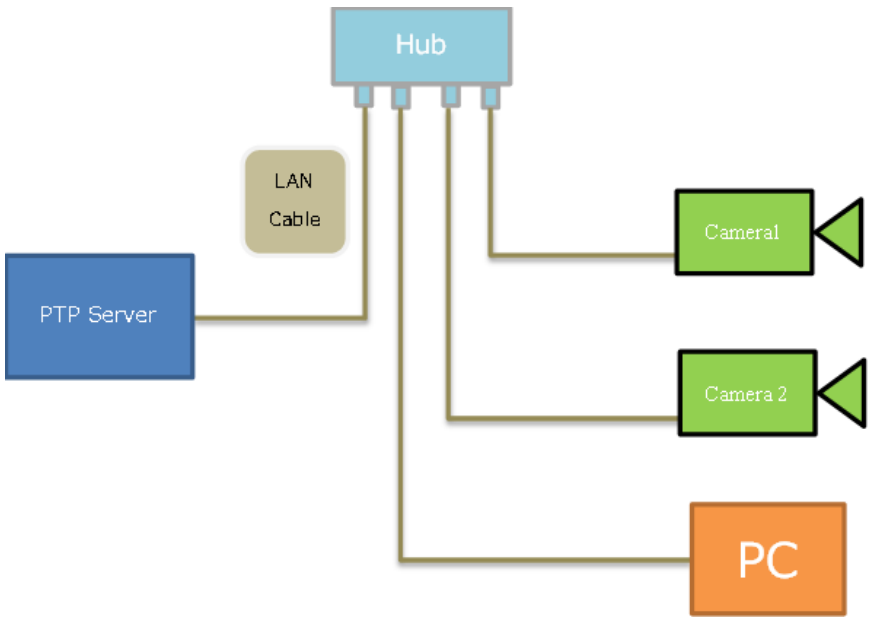

Fig. 2: Configuration Example

Using eBus Player for JAI, you can set up PTP and a scheduled Action Command as follows:

① In Device Control, set ActionDeviceKey, ActionGroupMask, and ActionGroupMask parameters which are described in previous sections.
② In Device Control, set GevIEEE1588 (in TransportLayerControl) to True.
③ Once GevIEEE1588Status is changed from Listening to Slave, PTP communication is established.
④ Select “GigE Vision Action Command” from the Tools pull-down menu and open “GigE Vision Action Command” dialog.

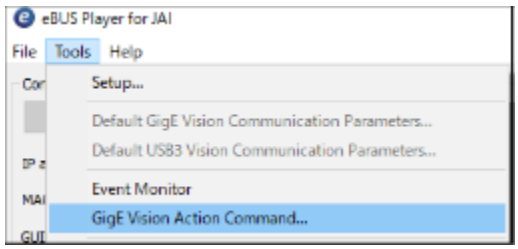

Fig. 3: Tools pull-down menu

⑤ In “Local Interfaces” group box, select the devices you want to receive the scheduled Action Command.

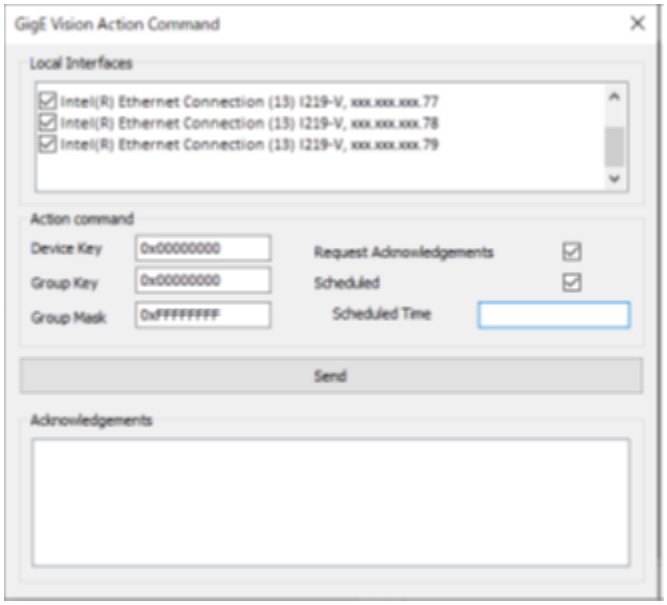

Fig. 4: Action Command Dialog

⑥ In “Action command” group box, set ActionDeviceKey, ActionGroupMask, and ActionGroupMask parameters based on the parameters set in step ①.

⑦ Tick the “Scheduled” check box and set the expected time of the action in the “Scheduled Time” field. The “Scheduled Time” should be entered as Epoch Time (UNIX timestamp). You can refer to the following URL to calculate Epoch Time. https://www.epochconverter.com/

⑧ Press the “Send” button. The keys and scheduled time set in steps ⑥ and ⑦ are then sent to the devices selected in step ⑤.

⑨ If the “Request Acknowledgements” check box is ticked, Acknowledgement is shown as follows:

OK: Action Command has been issued and selected camera(s) have received the command. The command will be executed at the specified time.

No Ref Time: In Device Control, GevIEEE1588 (in TransportLayerControl) is not “True”.

Overflow: Queue buffer of Action Command is full

Late: The command has been executed immediately because the “Scheduled Time” is in the past (earlier than the current time).

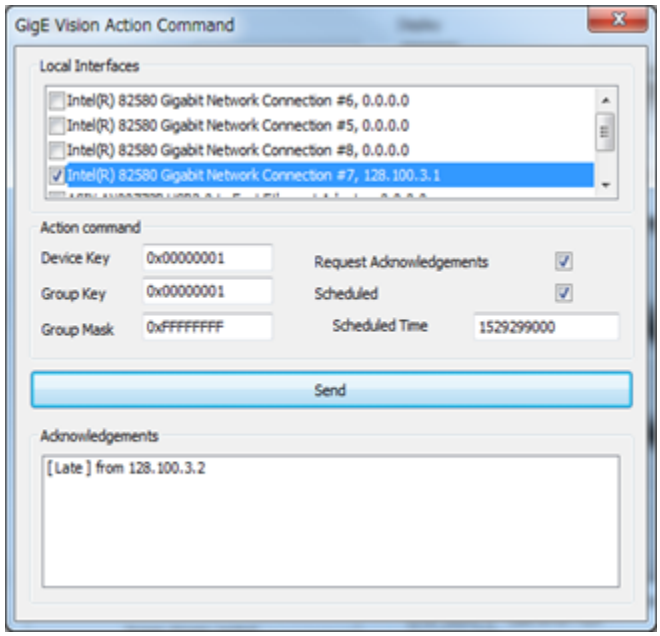

Fig. 5: Acknowledgements

End.

Technical Note

How to use Action Commands

Revision history

Revision	Date	Changes
1	2016/2/19	New release
2	2022/6/29	Added section 4: Scheduled Action Command