

# Trigger data in the video stream

## TABLE OF CONTENTS

- **DOCUMENT HISTORY**
- **1. OVERVIEW**
- **2. THEORY**
  - **2.1 Trigger data block**
  - **2.2 MJPEG**
  - **2.3 MPEG-4**
- **3. CONFIGURATION**
  - **3.1 Parameters**

### DOCUMENT HISTORY

2004-Apr-27 1.00	Initial version.
2004-Jun-16 1.10	Trigger data parameters have been moved from parameter group ImageSource.I# to Image.I#.

### 1. OVERVIEW

Trigger data describes different conditions in the Axis video product, for example digital input states, motion detection, video loss, etc. This document describes how to include trigger data in two different streaming formats, MJPEG and MPEG-4.

### 2. THEORY

#### 2.1 Trigger data block

The trigger data block contains the states of all triggers of interest, as opposed to just including a changed state in one block when it occurs. This allows the receiving application to lose blocks without necessarily losing a changed state.

The trigger data block format is the same for both MJPEG and MPEG-4. It's basically a block of text containing trigger states in this format:

```
<trigger>:<state>;<trigger>:<state>;...
```

Where <trigger> is a tag for the trigger and <state> is a text describing the state. The following table lists defined trigger tags and their possible states.

Trigger tag	Description	States
IO0 ... IO $n$	Status for digital input 0 to $n$ , where $n$ is the total number of inputs minus one. Note that state "1" means that the input has triggered, which isn't necessarily the same as that the input is high. Each input can be configured when to trigger.	0 1
V0 ... V $n$	Video loss status for video source 0 to $n$ , where $n$ is the total number of video sources minus one. State "1" means that there is no video for that video source.	0 1
M0 ... M $n$	Motion detection status for window 0 to $n$ , where $n$ is the total number of motion detection windows minus one. State "1" means that motion has been detected for this window, i.e. the motion level is above the configured threshold.	0 1
ML0 ... ML $n$	Motion detection level for window 0 to $n$ , where $n$ is the total number of motion detection windows minus one.	000 ... 100

New trigger tags may be added in the future. Any trigger tags not understood by the interpreter should simply be ignored.

**Example:**

```
IO0:0;IO1:1;IO2:1;IO3:0;V0:0;M0:1;ML0:035;
```

This should be interpreted as that input 0 and 3 are not triggered, and that input 1 and 2 are triggered. Video source 0 has video. Motion is detected for window 0. Motion level for window 0 is 35.

## 2.2 MJPEG

For MJPEG the trigger data block is included as a comment header for every image. Each image may contain several comments in this format:

Field	Size (bytes)	Value	Description
marker	2	0xFF 0xFE	JPEG comment marker.
length	2	4...260	Length of the comment.
axis_id	1	0x0A	This marks the comment as defined by Axis.
trigger_id	1	0x03	This marks the comment as trigger data.
data	length-4		The payload = the trigger data block.

## 2.3 MPEG-4

For MPEG-4 the trigger data block is included as "user data" in the GOV header. The GOV header may contain several data blocks following each other in this format:

Field	Size (bytes)	Value	Description
marker	4	0x00 0x00 0x01 0xB2	User data start code.
axis_id	1	0x0A	This marks the user data as defined by Axis.
trigger_id	1	0x03	This marks the user data as trigger data.
data	Length of trigger data	1 ... 256	The payload = the trigger data block.

The GOV header is inserted into the MPEG-4 stream at regular intervals, typically every 8th frame (this is configurable). A new GOV header can also be forced to be inserted as soon as possible when a trigger changes state. This is done for I/O triggers to reduce the latency.

## 3. CONFIGURATION

### 3.1 Parameters

The following parameters controls inclusion of trigger data in the video stream:

Parameter name	Default values	Valid values	Security level (get/set)	Description
Image.TriggerDataEnabled	no	no, yes	4/4	This parameter enables or disables the <i>triggerd</i> daemon, which is responsible for including trigger data in the video stream. You must set it to "yes" in order to make the other parameters work.
Image.I#.TriggerData.IOEnabled	yes	no, yes	4/4	This parameter enables inclusion of digital input states for image configuration #. Setting it to "yes" will include the states of all digital inputs in the trigger data for this image configuration. If you just want to include some digital inputs you should set it to "no" and use the UserTriggers parameter instead.
Image.I#.TriggerData.MotionDetectionEnabled	yes	no, yes	4/4	This parameter enables inclusion of motion detection states for image configuration #. Setting it to "yes" will include the motion detection states of the windows that belong to this image source in the trigger data. If you just want to include some window states or window states belonging to other image sources you should set it to "no" and use the UserTriggers parameter instead.
Image.I#.TriggerData.MotionLevelEnabled	no	no, yes	4/4	This parameter enables inclusion of motion detection levels for image configuration #. Setting it to "yes" will include the motion detection levels of the windows that belong to this image source in the trigger data. If you just want to include some window levels or window levels belonging to other image sources you should set it to "no" and use the UserTriggers parameter instead.
Image.I#.TriggerData.VideoLossEnabled	yes	no, yes	4/4	This parameter enables inclusion of video loss status for image configuration #. Setting it to "yes" will include the video loss status of the image source in the trigger data. If you want to include video loss states for other image sources you should use the UserTriggers parameter.
Image.I#.TriggerData.UserTriggers		A string	4/4	This string enables inclusion of user defined triggers for image configuration #. The string should be in the format "<trigger1>;<trigger2>;<trigger3>;...". See example 2 below.

In the following examples we have a video server with four video inputs and four digital inputs. We have configured one motion detection window per video input.

#### Example 1: Default configuration

Set Image.TriggerDataEnabled="yes" to enable trigger data. Apart from that we use the default configuration:

```
Image.I0.TriggerData.IOEnabled="yes"
Image.I0.TriggerData.MotionDetectionEnabled="yes"
Image.I0.TriggerData.MotionLevelEnabled="no"
Image.I0.TriggerData.VideoLossEnabled="yes"
Image.I0.TriggerData.UserTriggers=""
Image.I1.TriggerData.IOEnabled="yes"
Image.I1.TriggerData.MotionDetectionEnabled="yes"
Image.I1.TriggerData.MotionLevelEnabled="no"
Image.I1.TriggerData.VideoLossEnabled="yes"
```

```
Image.I1.TriggerData.UserTriggers=""
Image.I2.TriggerData.IOEnabled="yes"
Image.I2.TriggerData.MotionDetectionEnabled="yes"
Image.I2.TriggerData.MotionLevelEnabled="no"
Image.I2.TriggerData.VideoLossEnabled="yes"
Image.I2.TriggerData.UserTriggers=""
Image.I3.TriggerData.IOEnabled="yes"
Image.I3.TriggerData.MotionDetectionEnabled="yes"
Image.I3.TriggerData.MotionLevelEnabled="no"
Image.I3.TriggerData.VideoLossEnabled="yes"
Image.I3.TriggerData.UserTriggers=""
```

For image configuration 0 this would result in a trigger data block similar to this:

```
IO0:0;IO1:0;IO2:0;IO3:0;M0:0;V0:0;
```

And for image configuration 1:

```
IO0:0;IO1:0;IO2:0;IO3:0;M1:0;V1:0;
```

And for image configuration 2:

```
IO0:0;IO1:0;IO2:0;IO3:0;M2:0;V2:0;
```

And for image configuration 3:

```
IO0:0;IO1:0;IO2:0;IO3:0;M3:0;V3:0;
```

### **Example 2: All in one**

Include all trigger states in the trigger data block for image configuration 0:

```
Image.I0.TriggerData.IOEnabled="yes"
Image.I0.TriggerData.MotionDetectionEnabled="no"
Image.I0.TriggerData.MotionLevelEnabled="no"
Image.I0.TriggerData.VideoLossEnabled="no"
Image.I0.TriggerData.UserTriggers="M0;M1;M2;M3;ML0;ML1;ML2;ML3;V0;V1;V2;V3"
```

This would result in a trigger data block similar to this:

```
IO0:0;IO1:0;IO2:0;IO3:0;M0:0;M1:0;M2:0;M3:0;ML0:000;ML1:000;ML2:000;M3:000;V0:0;V1:0;V2:0;V3:0;
```