

Writing Applications with xiAPI

Default parameters

After camera is opened by xiOpenDevice the default camera parameters are set by API. The default parameters might be different in different API versions. In order to ensure that your application will have camera in expected state with any API version - please set all parameters expected by your application to required value.

Device IO

XI_PRM_TRG_SOURCE or "trigger_source"

Description: Defines source of trigger.

Note: To set input as external trigger, XI_PRM_GPI_MODE of selected input should be set to XI_GPI_TRIGGER. See example at XI_PRM_GPI_MODE.

Type: Enumerator.

Default value: XI_TRG_OFF

Usage:

```
int trigger_source = 0;
xiGetParamInt(handle, XI_PRM_TRG_SOURCE, &trigger_source);
xiSetParamInt(handle, XI_PRM_TRG_SOURCE, XI_TRG_OFF);
```

Value	Description
XI_TRG_OFF	Capture of next image is automatically started after previous.
XI_TRG_EDGE_RISING	Capture is started on rising edge of selected input.
XI_TRG_EDGE_FALLING	Capture is started on falling edge of selected input
XI_TRG_SOFTWARE	Capture is started with software trigger.
XI_TRG_LEVEL_HIGH	Specifies that the trigger is considered valid as long as the level of the source signal is high.
XI_TRG_LEVEL_LOW	Specifies that the trigger is considered valid as long as the level of the source signal is low.

[Example: For more examples see xiAPI Camera Trigger and Synchronization Signals.](#)

```
// enable trigger by software
xiSetParamInt(handle, XI_PRM_TRG_SOURCE, XI_TRG_SOFTWARE);
// start acquisition
xiStartAcquisition(handle);
int frames=5;
while (frames-->0)
{
    // trigger next image
    int trigger_retry=1000;
    while(--trigger_retry)
```

```

{
if (XI_OK == xiSetParamInt(handle, XI_PRM_TRG_SOFTWARE, 1))
    break;
// returns error in case if it is too early
// to start next exposure
Sleep(1);
}
// get the image
xiGetImage(handle, 1000, &image);
}

```

XI_PRM_TRG_SOFTWARE or "trigger_software"

Description: Generates an internal trigger. [XI_PRM_TRG_SOURCE](#) has to be set to [XI_TRG_SOFTWARE](#).

Note: Some models ([xiMU](#) - MU9 and [xiQ](#)) return error code if sensor is not ready to start exposure of next image. Other cameras return `XI_OK` even if sensor is not ready to start exposure.

Type: Integer.

Default value: 0

Usage:

```

int value = 0;
xiSetParamInt(handle, XI_PRM_TRG_SOFTWARE, value);

```

XI_PRM_TRG_SELECTOR or "trigger_selector"

Description: This parameter selects the type of trigger. For more information about enumerator [XI_TRG_SEL_EXPOSURE_ACTIVE](#) please refer to our [Exposure Defined by Trigger Pulse Length](#) support page.

For more information about enumerators: [XI_TRG_SEL_FRAME_BURST_START](#),

[XI_TRG_SEL_FRAME_BURST_ACTIVE](#) please refer to our [Frame Burst Modes](#) support page.

For more information about enumerator [XI_TRG_SEL_EXPOSURE_START](#) please refer to our [Multiple exposures in one frame](#) support page.

Type: Enumerator.

Default value: `XI_TRG_SEL_FRAME_START`

Usage:

```

int trigger_selector = 0;
xiGetParamInt(handle, XI_PRM_TRG_SELECTOR, &trigger_selector);
xiSetParamInt(handle, XI_PRM_TRG_SELECTOR, XI_TRG_SEL_FRAME_START);

```

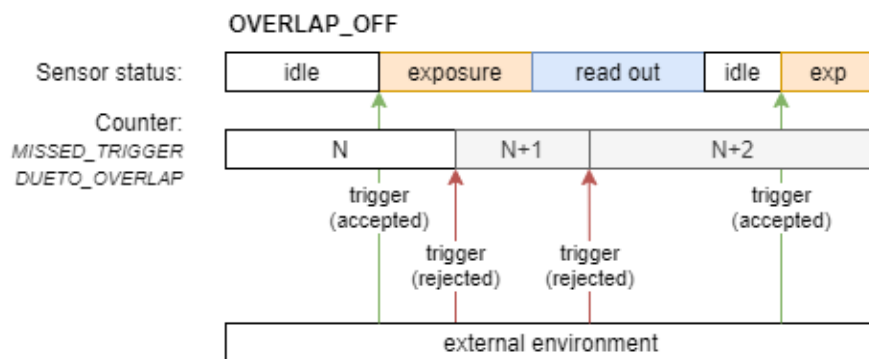
Value	Description
<code>XI_TRG_SEL_FRAME_START</code>	Trigger starts the capture of one frame
<code>XI_TRG_SEL_EXPOSURE_ACTIVE</code>	Trigger controls the start and length of the exposure.
<code>XI_TRG_SEL_FRAME_BURST_START</code>	Trigger starts the capture of the bursts of frames in an acquisition.

XI_TRG_SEL_FRAME_BURST_ACTIVE	Trigger controls the duration of the capture of the bursts of frames in an acquisition.
XI_TRG_SEL_MULTIPLE_EXPOSURES	Trigger which when first trigger starts exposure and consequent pulses are gating exposure(active HI)
XI_TRG_SEL_EXPOSURE_START	Trigger controls the start of the exposure of one Frame.
XI_TRG_SEL_MULTI_SLOPE_PHASE_CHANGE	Trigger controls the multi slope phase in one Frame (phase0 -> phase1) or (phase1 -> phase2).
XI_TRG_SEL_ACQUISITION_START	Trigger starts acquisition of first frame.

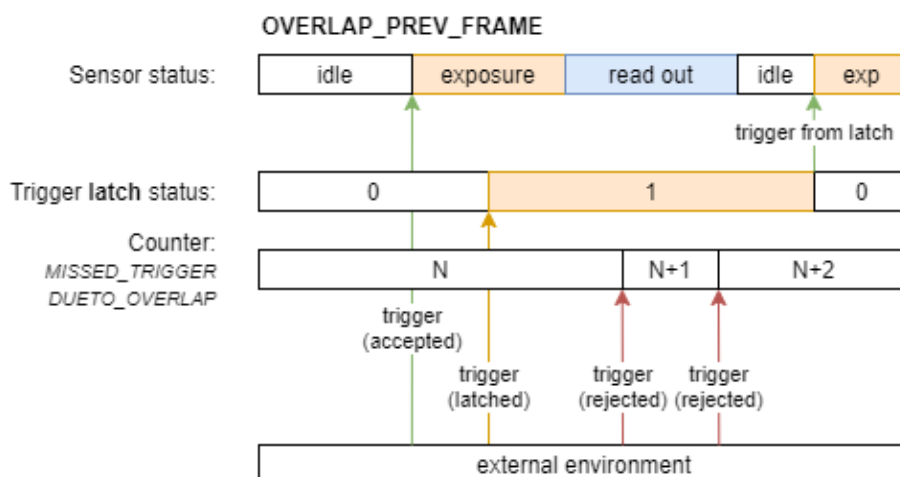
XI_PRM_TRG_OVERLAP or "trigger_overlap"

Description: Specifies the type of trigger overlap permitted with the previous frame. This defines when a valid trigger will be accepted (or latched) for a new frame.

In XI_TRG_OVERLAP_OFF - no trigger overlap is permitted. If camera is in read-out phase, all triggers are rejected.



In XI_TRG_OVERLAP_PREV_FRAME - trigger is accepted by camera any time. If sensor is not ready for the next exposure - the trigger is latched and sensor starts exposure as soon as exposure can be started with defined exposure time.



Type: Enumerator.

Default value: XI_TRG_OVERLAP_PREV_FRAME

Is invalidated by: XI_PRM_TRG_SELECTOR, XI_PRM_EXPOSURE_BURST_COUNT

Usage:

```
int trigger_overlap = 0;
xiGetParamInt(handle, XI_PRM_TRG_OVERLAP, &trigger_overlap);
xiSetParamInt(handle, XI_PRM_TRG_OVERLAP, XI_TRG_OVERLAP_OFF);
```

Value	Description
XI_TRG_OVERLAP_OFF	No trigger overlap is permitted. If camera is in read-out phase, all triggers are rejected.
XI_TRG_OVERLAP_READ_OUT	Trigger is accepted only when sensor is ready to start next exposure with defined exposure time. Trigger is rejected when sensor is not ready for new exposure with defined exposure time. (see Note1)
XI_TRG_OVERLAP_PREV_FRAME	Trigger is accepted by camera any time. If sensor is not ready for the next exposure - the trigger is latched and sensor starts exposure as soon as exposure can be started with defined exposure time.

Note1: This mode is planned and not yet supported by cameras.

XI_PRM_ACQ_FRAME_BURST_COUNT or "acq_frame_burst_count"

Description: Sets the number of frames to be acquired after trigger pulse has been sent to the camera. This setting is valid only if the trigger selector is set to FrameBurstStart. For more info please refer to our [Frame Burst Modes](#) support page. If burst count is set to zero (0) then number of acquired frames will not be limited (=endless).

Type: Integer.

Default value: 1

Usage:

```
int value = 0;
xiGetParamInt(handle, XI_PRM_ACQ_FRAME_BURST_COUNT, &value);
xiSetParamInt(handle, XI_PRM_ACQ_FRAME_BURST_COUNT, value);
```

XI_PRM_TIMESTAMP or "timestamp"

Description: Reads the current timestamp value from camera in nanoseconds (only valid for xiB, xiC, xiX camera families).

Type: Unsigned integer 64 bit.

Default value: 0

Usage:

```
uint64_t value = 0;
DWORD size = sizeof(value);
XI_PRM_TYPE type = xiTypeInteger64;
xiGetParam(handle, XI_PRM_TIMESTAMP, &value, &size, &type);
```

GPIO Setup

XI_PRM_GPI_SELECTOR or "gpi_selector"

Description: Selects GPI.

Type: Enumerator.

Default value: 1

Usage:

```
int gpi_selector = 0;
```

```
xiGetParamInt(handle, XI_PRM_GPI_SELECTOR, &gpi_selector);
```

```
xiSetParamInt(handle, XI_PRM_GPI_SELECTOR, XI_GPI_PORT1);
```

Value	Description
XI_GPI_PORT1	GPI port 1
XI_GPI_PORT2	GPI port 2
XI_GPI_PORT3	GPI port 3
XI_GPI_PORT4	GPI port 4
XI_GPI_PORT5	GPI port 5
XI_GPI_PORT6	GPI port 6
XI_GPI_PORT7	GPI port 7
XI_GPI_PORT8	GPI port 8
XI_GPI_PORT9	GPI port 9
XI_GPI_PORT10	GPI port 10
XI_GPI_PORT11	GPI port 11
XI_GPI_PORT12	GPI port 12

[Example: See XI_PRM_GPI_LEVEL](#)

On each camera family or model the relation of gpi_index and physical pin differs. Following table list the camera families and gpi_index relations.

Models: MQ-S7 cameras (MQ.*-S7)

gpi_index	physical pin on the camera	cable color
1	3 (IN1 - optical)	Blue

Models: MQ cameras (MQ.*)

gpi_index	physical pin on the camera	cable color
1	1 (IN1 - optical)	Red

Models: MJ cameras (MJ.*)

gpi_index	physical pin on the camera
1	3 (IN1 - optical)

Models: MC -UC cameras (MC.*-UC)

gpi_index	physical pin on the camera
1	11 (INOUT1)

2	3 (INOUT2)
---	------------

Models: MC -FL/FV cameras (MC.*-F.*)

gpi_index	physical pin on the camera
1	15 (IN1 - optical)

Models: MC -UB,-TC cameras (MC.*)

gpi_index	physical pin on the camera
1	5 (IN1 - optical)
2	8 (INOUT1)
3	2 (INOUT2)

Models: MU181CR-ON camera (MU181CR-.*)

gpi_index	physical pin on the camera
1	1 (GX1)
2	3 (GX2)
3	5 (GX3)

Models: MU TOF cameras (MU.*TG.*-UC)

gpi_index	physical pin on the camera
1	11 (INOUT1)
2	3 (INOUT2)

Models: MU cameras (MU.*-.*)

gpi_index	physical pin on the camera
1	1 (GX1)
2	3 (GX2)
3	5 (GX3)
4	7 (GX4)

Models: MX X2G2 cameras (MX.*X2G2.*)

gpi_index	physical pin on the camera
1	22 (IN1 - optical)
2	20 (INOUT1)
3	21 (INOUT2)

Models: MX X4G2 cameras (MX.*X4G2.*)

gpi_index	physical pin on the camera
1	2 (IN1 - optical)
2	4 (IN2 - optical)
3	6 (INOUT1)
4	7 (INOUT2)
5	45 (INOUT3)
6	46 (INOUT4)

Models: MX X4G3 cameras (MX.*X4G3.*)

gpi_index	physical pin on the camera
1	2 (IN1 - optical)
2	1 (IN2 - optical)
3	7 (INOUT1)
4	9 (INOUT2)
5	8 (INOUT3)
6	12 (INOUT4)

Models: MX X8G3 cameras (MX.*X8G3-FF.*)

gpi_index	physical pin on the camera
1	IO-A 1 (IN1 - optical)
2	IO-A 3 (IN2 - optical)
3	IO-B 1 (INOUT1)
4	IO-B 3 (INOUT2)
5	IO-B 4 (INOUT3)
6	IO-B 6 (INOUT4)

Models: CB X8G3 cameras (CB.*X8G3.*)

gpi_index	physical pin on the camera
1	2 (IN1 - optical)
2	1 (IN2 - optical)
3	7 (INOUT1)
4	9 (INOUT2)
5	8 (INOUT3)
6	12 (INOUT4)

Models: CB X4G2 cameras (CB.*)

gpi_index	physical pin on the camera
1	3 (IN1 - optical)
2	4 (IN2 - optical)
3	6 (INOUT1)
4	7 (INOUT2)
5	11 (INOUT3)
6	12 (INOUT4)

Models: MX377 MTP cameras (MX377.*MTP.*)

gpi_index	physical pin on the camera
2	8 (IN2 - optical)
3	13 (IN3 - optical)
4	2 (INOUT1)
5	3 (INOUT2)
6	4 (INOUT3)
7	5 (INOUT4)
8	12 (INOUT5)

9	14 (INOUT6)
10	9 (INOUT7)
11	7 (INOUT8)

XI_PRM_GPI_MODE or "gpi_mode"

Description: Defines GPI functionality.

Note1: To use GPI as trigger source, the [XI_PRM_TRG_SOURCE](#) should be also set to [XI_TRG_EDGE_RISING](#) or [XI_TRG_EDGE_FALLING](#)

Note2: If bidirectional (input/output) pin is used, set [XI_PRM_GPO_MODE](#) to [XI_GPO_HIGH_IMPEDANCE](#). This will disable output driver on the pin from camera.

Type: Enumerator.

Default value: XI_GPI_OFF

Is invalidated by: [XI_PRM_GPI_SELECTOR](#)

Usage:

```
int gpi_mode = 0;
xiGetParamInt(handle, XI_PRM_GPI_MODE, &gpi_mode);
xiSetParamInt(handle, XI_PRM_GPI_MODE, XI_GPI_OFF);
```

Value	Description
XI_GPI_OFF	Input is not used for triggering, but can be used to get parameter GPI_LEVEL. This can be used to switch I/O line on some cameras to input mode.
XI_GPI_TRIGGER	Input can be used for triggering.
XI_GPI_EXT_EVENT	External signal input (not implemented)

Example:

```
// select digital input (for xiQ=1, for xiC=1 or 2)
int input_id = 1;
xiSetParamInt(handle, XI_PRM_GPI_SELECTOR, input_id);
// set input as frame trigger
xiSetParamInt(handle, XI_PRM_GPI_MODE, XI_GPI_TRIGGER);
// enable triggering of image from digital input
xiSetParamInt(handle, XI_PRM_TRG_SOURCE, XI_TRG_EDGE_RISING);
```

XI_PRM_GPI_LEVEL or "gpi_level"

Description: Level of digital input selected by [XI_PRM_GPI_SELECTOR](#).

Note: When used on pin that could be input or output (E.g. pin 8 on MC023 camera), then associated GPO needs to be in mode [XI_GPO_HIGH_IMPEDANCE](#) . Otherwise pin can be pulled down ([GPO_OFF](#)) or up ([GPO_ON](#)). Such pins are [HIGH_IMPEDANCE](#) as default so application does not to setup it when used only as input.

Type: Integer.

Default value: 0

Usage:


```
int value = 0;
xiGetParamInt(handle, XI_PRM_GPI_LEVEL, &value);
```

Example:

```
// select digital input (different mapping to physical pin on each model, see table below)
int gpi_index = 1;
xiSetParamInt(handle, XI_PRM_GPI_SELECTOR, gpi_index);
// get input level
int gpi_level = 0;
xiGetParamInt(handle, XI_PRM_GPI_LEVEL, &gpi_level);
printf("Level on digital input %d is %d\n", gpi_index, gpi_level);
```

XI_PRM_GPI_LEVEL_AT_IMAGE_EXP_START or "gpi_level_at_image_exp_start"

Description: Level of digital input selected by [XI_PRM_GPI_SELECTOR](#) sampled at exposure start of the last image received by [GetImage](#).

Type: Integer.

Default value: 0

Usage:

```
int value = 0;
xiGetParamInt(handle, XI_PRM_GPI_LEVEL_AT_IMAGE_EXP_START, &value);
```

Example:

```
if (XI_OK == xiGetImage(xiH, 5000, &image))
{
    int level_input_1_at_exp_start = 0;
    xiSetParamInt(xiH, XI_PRM_GPI_SELECTOR, 1);
    xiGetParamInt(xiH, XI_PRM_GPI_LEVEL_AT_IMAGE_EXP_START, &level_input_1_at_exp_start);
}
```

XI_PRM_GPI_LEVEL_AT_IMAGE_EXP_END or "gpi_level_at_image_exp_end"

Description: Level of digital input selected by [XI_PRM_GPI_SELECTOR](#) sampled at exposure end of the last image received by [GetImage](#).

Type: Integer.

Default value: 0

Usage:

```
int value = 0;
xiGetParamInt(handle, XI_PRM_GPI_LEVEL_AT_IMAGE_EXP_END, &value);
```

Example:

```
if (XI_OK == xiGetImage(xiH, 5000, &image))  
{  
    int level_input_2_at_exp_end = 0;  
    xiSetParamInt(xiH, XI_PRM_GPI_SELECTOR, 2);  
    xiGetParamInt(xiH, XI_PRM_GPI_LEVEL_AT_IMAGE_EXP_END, &level_input_2_at_exp_end);  
}
```

XI_PRM_GPO_SELECTOR or "gpo_selector"

Description: Selects GPO.

Type: Enumerator.

Default value: 1

Usage:

```
int gpo_selector = 0;  
xiGetParamInt(handle, XI_PRM_GPO_SELECTOR, &gpo_selector);  
xiSetParamInt(handle, XI_PRM_GPO_SELECTOR, XI_GPO_PORT1);
```

Value	Description
XI_GPO_PORT1	GPO port 1
XI_GPO_PORT2	GPO port 2
XI_GPO_PORT3	GPO port 3
XI_GPO_PORT4	GPO port 4
XI_GPO_PORT5	GPO port 5
XI_GPO_PORT6	GPO port 6
XI_GPO_PORT7	GPO port 7
XI_GPO_PORT8	GPO port 8
XI_GPO_PORT9	GPO port 9
XI_GPO_PORT10	GPO port 10
XI_GPO_PORT11	GPO port 11
XI_GPO_PORT12	GPO port 12

Example: See XI_PRM_GPO_MODE

On each camera family or model the relation of gpo_index and physical pin differs. Following table list the camera families and gpi_index relations.

Models: MQ-S7 cameras (MQ.*-S7)

gpo_index	physical pin on the camera	cable color
1	4 (OUT1 - optical)	Black

Models: MQ cameras (MQ.*)

gpo_index	physical pin on the camera	cable color
1	3 (OUT1 - optical)	White

Models: MJ cameras (MJ.*)

gpo_index	physical pin on the camera
1	1 (OUT1 - optical)

Models: MC -UC cameras (MC.*-UC)

gpo_index	physical pin on the camera
2	12 (OUT1)
3	11 (INOUT1)
1	4 (OUT2)
4	3 (INOUT2)

Models: MC -FL/FV cameras (MC.*-F.*)

gpo_index	physical pin on the camera
1	13 (OUT1 - optical)

Models: MC -UB,-TC cameras (MC.*)

gpo_index	physical pin on the camera
1	3 (OUT1 - optical)
2	8 (INOUT1)
3	2 (INOUT2)

Models: MU181CR-ON camera (MU181CR-.*)

gpo_index	physical pin on the camera
1	1 (GX1)
2	3 (GX2)
3	5 (GX3)

Models: MU TOF cameras (MU.*TG.*-UC)

gpo_index	physical pin on the camera
3	12 (OUT1)
1	11 (INOUT1)
4	4 (OUT2)

Models: MU cameras (MU.*-.*)

gpo_index	physical pin on the camera
1	1 (GX1)
2	3 (GX2)
3	5 (GX3)
4	7 (GX4)

Models: MX X2G2 cameras (MX.*X2G2.*)

gpo_index	physical pin on the camera
1	24 (OUT1 - optical)
2	20 (INOUT1)
3	21 (INOUT2)

Models: MX X4G2 cameras (MX.*X4G2.*)

gpo_index	physical pin on the camera
1	50 (OUT1 - optical)
2	48 (OUT2 - optical)
3	6 (INOUT1)
4	7 (INOUT2)
5	45 (INOUT3)
6	46 (INOUT4)

Models: MX X4G3 cameras (MX.*X4G3.*)

gpo_index	physical pin on the camera
1	4 (OUT1 - optical)
2	3 (OUT2 - optical)
3	7 (INOUT1)
4	9 (INOUT2)
5	8 (INOUT3)
6	12 (INOUT4)

Models: MX X8G3 cameras (MX.*X8G3-FF.*)

gpo_index	physical pin on the camera
1	IO-A 4 (OUT1 - optical)
2	IO-A 6 (OUT2 - optical)
3	IO-B 1 (INOUT1)
4	IO-B 3 (INOUT2)
5	IO-B 4 (INOUT3)
6	IO-B 6 (INOUT4)

Models: CB X8G3 cameras (CB.*X8G3.*)

gpo_index	physical pin on the camera
1	4 (OUT1 - optical)
2	3 (OUT2 - optical)
3	7 (INOUT1)
4	9 (INOUT2)
5	8 (INOUT3)
6	12 (INOUT4)

Models: CB X4G2 cameras (CB.*)

gpo_index	physical pin on the camera
1	8 (OUT1 - optical)
2	9 (OUT2 - optical)
3	6 (INOUT1)
4	7 (INOUT2)
5	11 (INOUT3)
6	12 (INOUT4)

Models: MX377 MTP cameras (MX377.*MTP.*)

gpo_index	physical pin on the camera
2	19 (OUT2 - optical)
3	17 (OUT3 - optical)
4	2 (INOUT1)
5	3 (INOUT2)
6	4 (INOUT3)
7	5 (INOUT4)
8	12 (INOUT5)
9	14 (INOUT6)
10	9 (INOUT7)
11	7 (INOUT8)

XI_PRM_GPO_MODE or "gpo_mode"

Description: Defines GPO functionality.

Note1: On some camera models (MR, MH): Modes FRAME_ACTIVE or EXPOSURE_ACTIVE are supported only if XI_PRM_TRG_SOURCE is set to XI_TRG_SOFTWARE or XI_TRG_EDGE_RISING or XI_TRG_EDGE_FALLING. See section XI_PRM_TRG_SOURCE On models xiMU (MU9) xiQ.

Note2: Some camera families (e.g. MR) does not support the software control of outputs. Only one of mode: FRAME_ACTIVE and EXPOSURE_ACTIVE can be set.

Note3: Duration of pulse depends on camera model and polarity of signal.

Note4: Each bidirectional line has only one control for inverter (as in GenICam-SFNC). If output mode with _NEG extension is set then also input signal becomes inverted.

Type: Enumerator.

Default value: XI_GPO_OFF

Is invalidated by: [XI_PRM_GPO_SELECTOR](#)

Usage:

```
int gpo_mode = 0;
xiGetParamInt(handle, XI_PRM_GPO_MODE, &gpo_mode);
xiSetParamInt(handle, XI_PRM_GPO_MODE, XI_GPO_OFF);
```

Value	Description
XI_GPO_OFF	Output is off (zero voltage or switched_off)
XI_GPO_ON	Output is on (voltage or switched_on)
XI_GPO_FRAME_ACTIVE	Output is on while frame exposure,read,transfer.
XI_GPO_FRAME_ACTIVE_NEG	Output is off while frame exposure,read,transfer.
XI_GPO_EXPOSURE_ACTIVE	Output is on while frame exposure
XI_GPO_EXPOSURE_ACTIVE_NEG	Output is off while frame exposure
XI_GPO_FRAME_TRIGGER_WAIT	Output is on while camera is ready for trigger
XI_GPO_FRAME_TRIGGER_WAIT_NEG	Output is off while camera is ready for trigger.
XI_GPO_EXPOSURE_PULSE	Output is on short pulse at the beginning of frame exposure.
XI_GPO_EXPOSURE_PULSE_NEG	Output is off short pulse at the beginning of frame exposure.

XI_GPO_BUSY	Output is on when camera has received trigger until end of transfer
XI_GPO_BUSY_NEG	Output is off when camera has received trigger until end of transfer
XI_GPO_HIGH_IMPEDANCE	Associated pin is in high impedance (tri-stated) and can be driven externally. E.g. for triggering or reading status by GPI_LEVEL.
XI_GPO_FRAME_BUFFER_OVERFLOW	Frame buffer overflow status.
XI_GPO_EXPOSURE_ACTIVE_FIRST_ROW	Output is on while the first row exposure.
XI_GPO_EXPOSURE_ACTIVE_FIRST_ROW_NEG	Output is off while the first row exposure.
XI_GPO_EXPOSURE_ACTIVE_ALL_ROWS	Output is on while all rows exposure together.
XI_GPO_EXPOSURE_ACTIVE_ALL_ROWS_NEG	Output is off while all rows exposure together.
XI_GPO_TXD	Output is connected to TXD of UART module

Example:

```
xiSetParamInt(handle, XI_PRM_GPO_SELECTOR, 1);
// make output on (one)
xiSetParamInt(handle, XI_PRM_GPO_MODE, XI_GPO_ON);
Sleep(1000); // wait to see the output is on (e.g. LED)
// make output off
xiSetParamInt(handle, XI_PRM_GPO_MODE, XI_GPO_OFF);
```

XI_PRM_LED_SELECTOR or "led_selector"

Description: Selects LED.

Type: Enumerator.

Default value: 1

Usage:

```
int led_selector = 0;
xiGetParamInt(handle, XI_PRM_LED_SELECTOR, &led_selector);
xiSetParamInt(handle, XI_PRM_LED_SELECTOR, XI_LED_SEL1);
```

Value	Description
XI_LED_SEL1	LED 1
XI_LED_SEL2	LED 2
XI_LED_SEL3	LED 3
XI_LED_SEL4	LED 4
XI_LED_SEL5	LED 5

XI_PRM_LED_MODE or "led_mode"

Description: Defines LED functionality.

Type: Enumerator.

Default value: XI_LED_HEARTBEAT

Is invalidated by: [XI_PRM_LED_SELECTOR](#)

Usage:

```
int led_mode = 0;
xiGetParamInt(handle, XI_PRM_LED_MODE, &led_mode);
xiSetParamInt(handle, XI_PRM_LED_MODE, XI_LED_HEARTBEAT);
```

Value	Description
XI_LED_HEARTBEAT	Set led to blink (1 Hz) if link is OK.
XI_LED_TRIGGER_ACTIVE	Set led to blink if trigger detected.
XI_LED_EXT_EVENT_ACTIVE	Set led to blink if external signal detected.
XI_LED_LINK	Set led to blink if link is OK.
XI_LED_ACQUISITION	Set led to blink if data streaming
XI_LED_EXPOSURE_ACTIVE	Set led to blink if sensor integration time.
XI_LED_FRAME_ACTIVE	Set led to blink if device busy/not busy.
XI_LED_OFF	Set led to off.
XI_LED_ON	Set led to on.
XI_LED_BLINK	Blinking (1Hz).

XI_PRM_DEBOUNCE_EN or "dbnc_en"

Description: Enable/Disable debounce to selected GPI (XI_PRM_GPI_SELECTOR parameter). (see Note 1)

Note1: Parameter is available only for xiQ camera models.

Type: Integer.

Default value: XI_OFF

Usage:

```
int value = 0;
xiGetParamInt(handle, XI_PRM_DEBOUNCE_EN, &value);
xiSetParamInt(handle, XI_PRM_DEBOUNCE_EN, XI_ON);
```