BG Series
CMOS Camera

Instruction Manual

Model
Mono Camera   : BG040M / BG160M

Thank you for purchasing our product.

Before using this CMOS camera, please read through this instruction manual carefully in order to use this product correctly and safely.

After reading, keep this instruction manual handy so that you can refer to, whenever you need it.

Toshiba Teli Corporation

Information contained in this document is subject to change without prior notice.

Standard name might be trade mark of each company.

D4288264C

Contents

D4288264C

https://www.toshiba-teli.co.jp/en/

D4288264C

https://www.toshiba-teli.co.jp/en/

Safety Precautions

Before using this product, read these safety precautions carefully. Important information is shown in this Instruction Manual to protect users from bodily injuries and property damages, and to enable them to use the product safely and correctly.

Please be sure to thoroughly understand the meanings of the following signs and symbols before reading the main text that follow, and observe the instructions given herein.

[Definition of Safety Signs]

Safety Signs	Description
⚠ WARNING	Indicates a potentially hazardous situation that may result in death or serious injury (*1) in the event of improper handling.
⚠ CAUTION	Indicates a potentially hazardous situation that may result in light to moderate injuries (*2) or only in property damage (*3)in the event of improper handling.

Notes *1: "Serious injury" refers to cases of loss of eyesight, wounds, burns (high or low temperature), electric shock, broken bones, poisoning, etc., which leave after-effects or which require hospitalization or a long period of outpatient treatment of cure.

*2: "Light to moderate injuries" refers to injuries, burns, electric shock etc. that do not require hospitalization or long-term treatment.

*3: "Property damage" refers to cases of extensive damage involving damage to buildings, equipment, farm animals, pet animals and other belongings.

[Explanation of Safety Symbols]

Safety Symbols	Description
🚫 PROHIBITED	This sign indicates **PROHIBITION** (Do not). The content of prohibition is shown by a picture or words beside the symbol.
⬤ MANDATORY	This sign indicates **MANDATORY ACTION** (You are required to do). The content of action is shown by a picture or words beside the symbol.

D4288264C
https://www.toshiba-teli.co.jp/en/

General Handing

⚠WARNING

Unplug

- **Stop operation immediately when any abnormality or defect occurs.**
  If abnormal conditions are present, such as smoke, a burning smell, ingress of water or foreign matter, or if the equipment is dropped or malfunctions, fire or electric shock may result.
  Be always sure to disconnect the power cable from the wall socket at once and contact your dealer.

Do not get wet

- **Do not use the equipment in locations subject to water splashes.**
  Otherwise, fire or electric shock may result.

Never pull apart

- **Do not disassemble, repair, or modify the equipment.**
  Otherwise, fire or electric shock may result.
  For internal repair, inspection, or cleaning, contact your sales representative.

Avoid

- **Do not place anything on the equipment.**
  If metallic objects, liquid, or other foreign matter enters the equipment, fire or electric shock may result.

Avoid

- **Do not install the equipment in an unstable or inclined location or locations subject to vibration or impact.**
  Otherwise, the equipment may topple over and cause personal injury.

Do not touch

- **During an electrical storm, do not touch the power cable and the connection cable.**
  Otherwise, an electric shock may result.

Instruction

- **Use the specified voltage.**
  Use of an unspecified voltage may result in fire or electric shock.

Avoid

- **Do not be handled roughly, damaged, fabricated, bent forcefully, pulled, twisted, bundled, placed under heavy objects or heated the power cable and the connection cable.**
  Otherwise, fire or electric shock may result.

⚠ CAUTION

! Instruction	● **Observe the following when installing the equipment:** ·Do not cover the equipment with a cloth, etc. ·Do not place the equipment in a narrow location where heat is likely to accumulate. Otherwise, heat will accumulate inside the equipment, possibly resulting in a fire.
🚫 Avoid	● **Do not place the equipment in locations subject to high moisture, oil fumes, steam, or dust.** Otherwise, fire or electric shock may result.
🚫 Avoid	● **Do not install the equipment in locations exposed to direct sunlight or humidity.** Otherwise, the internal temperature of the equipment will rise, which may cause a fire.
! Instruction	● **Use only specified the power cable and the connection cables.** Otherwise, fire or electric shock may result.
🚫 Avoid	● **Do not give strong impact against the equipment.** It may cause the trouble.
! Instruction	● **When performing connection, turn off power.** When connecting the power cable and the connection cable, turn off the equipment power. Otherwise, fire or electric shock may result.
🚫 Avoid	● **Do not expose its camera head to any intensive light (such as direct sunlight).** Otherwise, its inner image pickup device might get damaged.
🚫 Avoid	● **Avoid short-circuiting signal output.** Otherwise, a malfunction may occur.
🚫 Avoid	● **Avoid giving a strong shock against the camera body.** It might cause a breakdown or damage. If your camera is used in a system where its camera connector is subjected to strong repetitive shocks, its camera connector is possible to break down. If you intend to use your camera in such a situation, if possible, bundle and fix a camera cable in the place near the camera, and do not transmit a shock to the camera connector.
! Instruction	● **Contact your sales representative to request periodic inspection and cleaning (every approx five years).** Accumulation of dust inside the equipment may result in fire or electric shock. For inspection and cleaning costs, contact your sales representative.

D4288264C

https://www.toshiba-teli.co.jp/en/

CASES FOR INDEMNITY (LIMITED WARRANTY)

We shall be exempted from taking responsibility and held harmless for damage or losses incurred by the user in the following cases.

- In the case damage or losses are caused by natural disasters, such as an earthquake and thunder, fire, or other acts of God, acts by a third party, deliberate or accidental misuse by the user, or use under extreme operating conditions.

- In the case of indirect, additional, consequential damages (loss of business interests, suspension of business activities) are incurred as result of malfunction or non-function of the equipment, we shall be exempted from responsibility for such damages.

- In the case damage or losses are caused by failure to observe the information contained in the instructions in this instruction manual and specifications.

- In the case damage or losses are caused by use contrary to the instructions in this instruction manual and specifications.

- In the case damage or losses are caused by malfunction or other problems resulting from unintended use of equipment or software etc. that are not specified.

- In the case damage or losses are caused by repair or modification conducted by the customer or any unauthorized third party (such as an unauthorized service representative).

- Expenses we bear on this product shall be limited to the individual price of the product.

- The item that is not described in specifications of this product is out of the guarantee.

- The case of damages or losses which are caused by incorrect connection of the cable is out of the guarantee.

RESTRICTION FOR USE

● Should the equipment be used in the following conditions or environments, give consideration to safety measures and inform us of such usage:

    1. Use of the equipment in the conditions or environment contrary to those specified, or use outdoors.

    2. Use of the equipment in applications expected to cause potential hazard to people or property, which require special safety measures to be adopted.

● This product can be used under diverse operating conditions. Determination of applicability of equipment or devices concerned shall be determined after analysis or testing as necessary by the designer of such equipment or devices, or personnel related to the specifications. Such designer or personnel shall assure the performance and safety of the equipment or devices.

● This product is not designed or manufactured to be used for control of equipment directly concerned with human life (*1) or equipment relating to maintenance of public services/functions involving factors of safety (*2). Therefore, the product shall not be used for such applications.

    (*1): Equipment directly concerned with human life refers to.

      - Medical equipment such as life-support systems, equipment for operating theaters.

      - Exhaust control equipment for exhaust gases such as toxic fumes or smoke.

      - Equipment mandatory to be installed by various laws and regulations such as the Fire Act or Building Standard Law

      - Equipment related to the above

    (*2): Equipment relating to maintenance of public services/functions involving factors of safety refers to.

      - Traffic control systems for air transportation, railways, roads, or marine transportation

      - Equipment for nuclear power generation

      - Equipment related to the above

Notes on using this product

● Handle carefully

   Do not drop the equipment or allow it to be subject to strong impact or vibration, as such action may cause malfunctions. Further, do not damage the connection cable, since this may cause wire breakage.

● Environmental operating conditions

   Do not use the product in locations where the ambient temperature or humidity exceeds the specifications.

   Otherwise, image quality may be degraded or internal components may be adversely affected. In particular, do not use the product in areas exposed to direct sunlight. Moreover, during shooting under high temperatures, vertical stripes or white spots (noise) may be produced, depending on the subject or camera conditions (such as increased gain). However, such phenomena are not malfunctions.

● Check a combination with the lens

   Depending on the lens and lighting you use, an image is reflected as a ghost in the imaging area. However, this is not because of a fault of the camera.

   In addition, depending on the lens you use, the performance of the camera may not be brought out fully due to deterioration in resolution and brightness in the peripheral area, aberration and others.

   Be sure to check a combination with the camera by using the lens and lightning you actually use.

   When installing a lens in the camera, make sure carefully that it is not tilted.

   In addition, use a mounting screw free from defects and dirt. Otherwise, the camera may be unable to be removed.

   Install a next lens; its dimension of protrusion from bottom of the screw is equal to or less than 10mm. If a lens does not stand to this condition, it might not be installed to this camera.
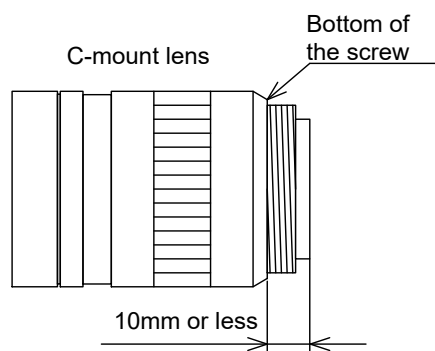
● Mounting to pedestal

   When mounting this product to a pedestal, make sure carefully that lens doesn't touch with the pedestal.

D4288264C
https://www.toshiba-teli.co.jp/en/

● Do not expose the camera's image-pickup-plane to sunlight or other intense light directly

   Its inner CMOS sensor might be damaged.

● Occurrence of moiré

   If you shoot thin stripe patterns, moiré patterns (interference fringes) may appear. This is not a malfunction.

● Occurrence of noise on the screen

   If an intense magnetic or electromagnetic field is generated near the camera or connection cable, noise may be generated on the screen. If this occurs, move the camera or the cable.

● Handling of the protective cap

   If the camera is not in use, attach the lens cap to the camera to protect the image pickup surface.

● If the equipment is not to be used for a long duration

   Turn off power to the camera for safety.

● Maintenance

   Turn off power to the equipment and wipe it with a dry cloth.

   If it becomes severely contaminated, gently wipe the affected areas with a soft cloth dampened with diluted neutral detergent. Never use alcohol, benzene, thinner, or other chemicals because such chemicals may damage or discolor the paint and indications.

   If the image pickup surface becomes dusty, contaminated, or scratched, consult your sales representative.

● Disposal

When disposing of the camera, it may be necessary to disassemble it into separate parts, in accordance with the laws and regulations of your country and/or municipality concerning environmental contamination.

This product is marked this symbol to subject to EU Waste Electrical & Electronic Equipment (WEEE) directive.
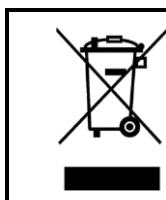
This equipment has been tested and found to comply with the limits for a class A digital device, pursuant to Part 15 of the FCC Rules.

These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment.

This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communication.

Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be require to correct the interference at his own expense.

[Phenomena specific to CMOS sensor]

- **Defective pixels**

  A CMOS image sensor is composed of photo sensor pixels in a square grid array. Due to the characteristics of CMOS image sensors, over- or under-driving of the pixels results in temporary white or black areas (as if these are noises) appearing on the screen. This phenomenon, which is not a defect is exacerbated under higher temperatures and long exposure time.

- **Image shading**

  The brightness of the upper part of the screen may be different from that of the lower part. Note that this is a characteristic of a CMOS image sensor and is not a fault.

	环保使用期限标识，是根据电子信息产品污染控制管理办法以及，电子信息产品污染控制标识要求(SJ/T11364-2014)、电子信息产品环保使用期限通则，制定的适用于中国境内销售的电子信息产品的标识。
  中华人民共和国 环保使用期限	电子信息产品只要按照安全及使用说明内容，正常使用情况下，从生产月期算起，在此期限内，产品中含有的有毒有害物质不致发生外泄或突变，不致对环境造成严重污染或对其人身、财产造成严重损害。 产品正常使用后，要废弃在环保使用年限内或者刚到年限的产品时，请根据国家标准采取适当的方法进行处置。 另外，此期限不同于质量/功能的保证期限。 The Mark and Information are applicable for People's Republic of China only.

＜产品中有毒有害物质或元素的名称及含量＞

部件名称	有毒有害物质或元素					
	铅（Pb）	汞（Hg）	镉（Cd）	六价铬 （Cr(VI)）	多溴联苯 （PBB）	多溴二苯醚 （PBDE）
相机本体	×	○	○	○	○	○
本表格依据SJ/T 11364的规定编制 ○：表示该有毒有害物质在该部件所有均质材料中的含量均在电子信息产品中有毒有害物质 　　的限量要求标准规定的限量要求(GB/T26572)以下 ×：表示该有毒有害物质至少在该部件的某一均质材料中的含量超出电子信息产品中有毒有 　　害物质的限量要求标准规定的限量要求(GB/T26572) 　This information is applicable for People's Republic of China only.						

リサイクルに関する情報（包装物）
有关再利用的信息(包装物)
Information on recycling of wrapping composition

箱 / 箱子 / Box

内部緩衝材料・袋
内部缓冲材料・袋
Internal buffer materials・Bag
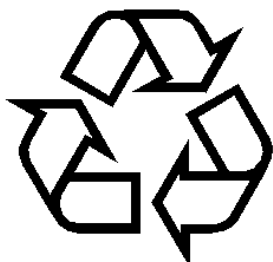
ペーパーボード
纸板
Paper board

PE-LD

D4288264C
https://www.toshiba-teli.co.jp/en/

Installation

Before using this product, you shall install application software to display image and control registers of camera, and IP configuration tool for network setting.

You can download the SDK for our GigE and USB3 camera products (TeliCamSDK) from the Service & Support section of our website.

User registration is necessary to use downloading service. Please make a user registration, or contact your dealer / distributor.

- TOSHIBA TELI CORPORATION Top Page
  https://www.toshiba-teli.co.jp/en/index.htm

- Service & Support
  https://www.toshiba-teli.co.jp/cgi/ss/en/service.cgi

Please refer to the TeliCamSDK startup guide, about Operation environment, Installation, and Setup.

D4288264C
https://www.toshiba-teli.co.jp/en/

Specifications

Overview

This BG series is an integrated-(one-body)-type camera that adopts a global shutter CMOS sensor.

These are BG040M (0.4M 1/2.9 type), BG160M (1.6M 1/2.9 type).

For video output and camera control, the Gigabit Ethernet® interface standard "IEEE802.3ab" is adopted for high transfer rate, and it is easy to integrate into industrial equipment.

Features

● High frame rate and high resolution

Supporting high frame rate, BG040M 291fps, BG160M 72fps.

● Global shutter

As it employs a global electronic shutter similar to a CCD image sensor, clear images of even fast-moving object are obtainable with less blur.

● Gigabit Ethernet interface (Power over Ethernet)

Video output and camera control are performed via the Gigabit Ethernet standard IEEE802.3ab interface. Data transfer is up to 1Gbps (Maximum) that enables to output uncompressed video data at high frame rate. By complying with IEEE802.3af Power over Ethernet (PoE), the power is supplied over single cable.

● GigEVision Ver 1.2 conformity

This product is based on GigEVision Camera Interface Standard for Machine Vision Ver 1.2.

● GenICam Ver 2.4, Ver 3.0 conformity

This product is based on GenICam (Generic Interface for Cameras) Ver 2.4 and Ver 3.0.

● IIDC2 Digital Camera Control Specification Ver.1.1.0

This product is based on IIDC2 Digital Camera Control Specification Ver.1.1.0

● Random Trigger Shutter

The Random Trigger Shutter function provides images in any timing by input of an external trigger signal. Trigger control from PC is available as well.

● Scalable

Selectable video output area. This mode achieves higher frame rate by reducing vertical output area. And reduces occupied data rate of Gigabit Ethernet by reducing horizontal output area.

● Binning mode

Pixel data is combined by vertical and horizontal.

● Decimation mode

Camera reads all effective areas at high speed by skipping lines.

● Compact and lightweight

This camera is compact and lightweight; it is easy to integrate into industrial equipment.

● ShortExposureMode

Firmware Ver.3.5.0 or later models have ShortExposureMode. By setting the ShortExposureMode to enable, high-speed exposure time setting from 1.08μs to 13.31μs is possible.

Configuration

The system configuration of this camera series is as follows;
This camera does not include any accessories. Please prepare other equipments separately.

● Camera: This product. (BG series)

● Camera mounting kit CPTBUBG (*1): To fix a camera to a tripod; attach this to the bottom of the camera.

● LAN Cable (*2): This cable is used to connect the camera to host equipment. Ethernet packets (stream and control) are transmitted via this cable. Please use a LAN cable that supports 1000BASE-T (Cat 5e or over). This product is able to connect a LAN cable that is equipped with screw lock mechanism. Please use it as needed.

● Network Interface Card (NIC) (*2): This is the interface card to connect to the camera. Usually this card is installed to expansion slot of host equipment such as PC etc. Please use 1000BASE-T NIC, supporting Jumbo Frame is recommended.

● PoE Switching HUB etc. (*2): Use these equipment when you supply power to the camera from PoE. Please use 1000BASE-T equipments, supporting Jumbo Frame is recommended.

*1: Optional part. Contact your dealer / distributor for details of option units.
*2: Commercial items.

Connection

● In the case of supply DC+12V to the camera.

BG Series
CMOS Camera

LAN Cable

← Control Packet   Stream Packet →

1. DC+12V IN
2. LINE0 IN
3. LINE1 IN / OUT
4. LINE2 OUT
5. I/O GND
6. GND

Network Interface
Card

Host (PC etc.)

Camera mounting kit
CPTBUBG
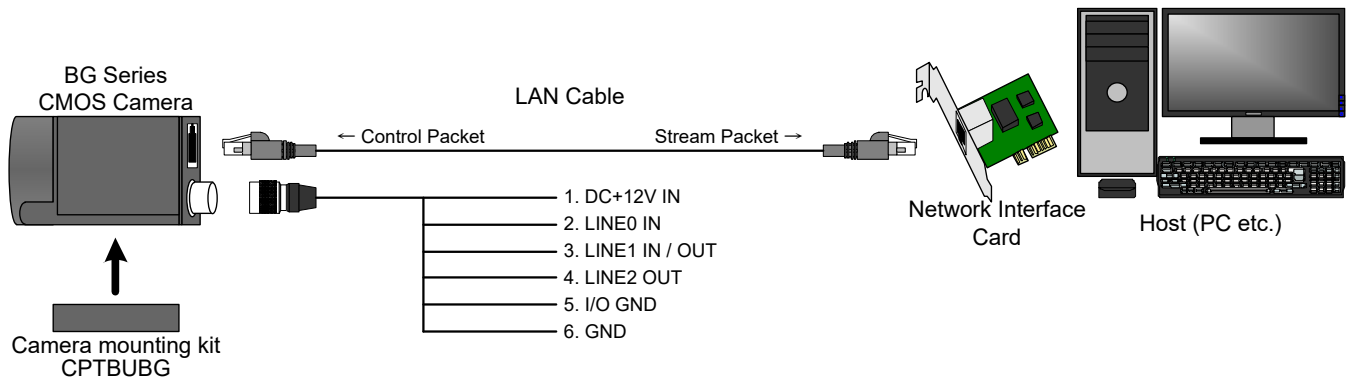
● In the case of supply PoE power to the camera.

BG Series
CMOS Camera

LAN Cable

← Control Packet   Stream Packet →
← PoE Power

Network Interface
Card

Host (PC etc.)

Camera mounting kit
CPTBUBG

PoE Switching HUB etc.
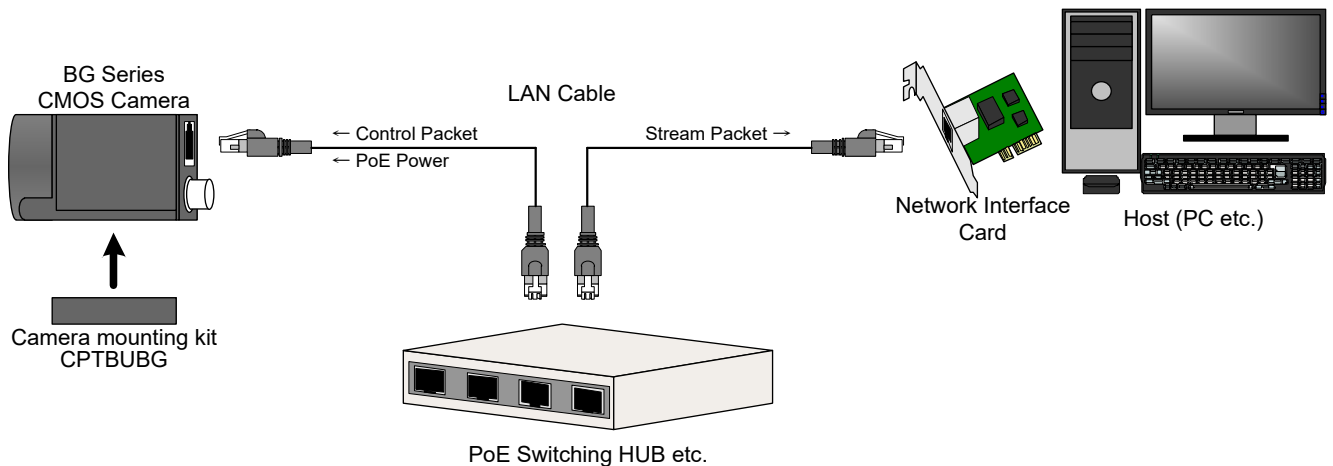
Notes on Connection:

- Please confirm the power supply of the camera off when plugging in or pulling out the I/O Connector. It causes the breakdown.

- If your camera is used in a system where its connectors are subjected to strong repetitive shocks, its connectors are possible to break down. If you use your camera in such a situation, use an LAN cable with a lock screw, and secure the camera cable as close as possible to the camera body for avoid physical shock to the camera connector.

- About Camera cable: In the case that electric-wire is long or thin, input voltage may not satisfy specifications of the power supply voltage of the camera by voltage drop. Please confirm it before use.

- Lost packets may occur by an electrical characteristic of the transmission line of using Ethernet devices (LAN cable, Network Interface Card, Switching HUB).

D4288264C
https://www.toshiba-teli.co.jp/en/

Connector Pin Assignment
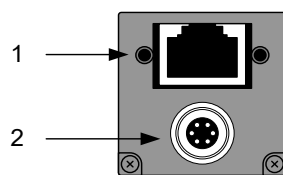
Rear View

1. Gigabit Ethernet Interface Connector

RJ-45 Jack

Pin No.	I/O	Signal	Function
1	I/O	BI_DA+ / VDC+	Bidirectional Data A (+) / Power (+)
2	I/O	BI_DA- / VDC+	Bidirectional Data A (-) / Power (+)
3	I/O	BI_DB+ / VDC-	Bidirectional Data B (+) / Power (-)
4	I/O	BI_DC+ / VDC+	Bidirectional Data C (+) / Power (+)
5	I/O	BI_DC- / VDC+	Bidirectional Data C (-) / Power (+)
6	I/O	BI_DB- / VDC-	Bidirectional Data B (-) / Power (-)
7	I/O	BI_DD+ / VDC-	Bidirectional Data D (+) / Power (-)
8	I/O	BI_DD- / VDC-	Bidirectional Data D (-) / Power (-)

2. I/O Connector

Connector (Camera side)　　HR10A-7R-6PB(73) (HIROSE ELECTRIC CO., LTD.) or equivalency product

Plug　　　(Cable side)　　HR10A-7P-6S(73)　(HIROSE ELECTRIC CO., LTD.) or equivalency product

* This camera cable is not an accessory of this product.

Connector view from mating face

Camera side　　　　　　　　　　　Cable side

↑TOP
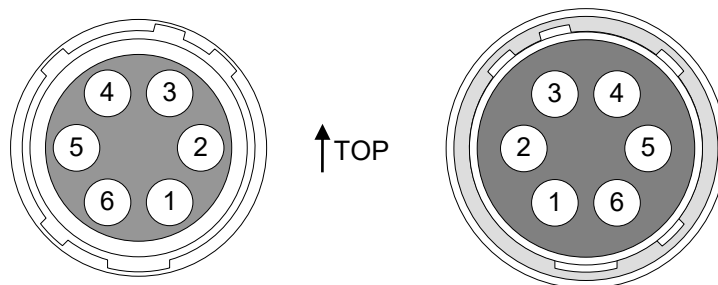
Pin No.	I/O	Signal	Function
1	I	+12V	Power
2	I	Line 0	External Trigger Input
3	I/O	Line 1	GPIO_0 Output / External Trigger Input
4	O	Line 2	GPIO_1 Output
5	-	I/O GND	I/O_Ground
6	-	GND	Ground

D4288264C

https://www.toshiba-teli.co.jp/en/

Notes on Power Supply:

This camera has two ways of power supply,

  - Supply from LAN cable (PoE)

  - Supply from camera cable (DC+12V ±10%)

If both of PoE and DC+12V are connected, power is supplied from PoE.

D4288264C

https://www.toshiba-teli.co.jp/en/

Outline Drawing
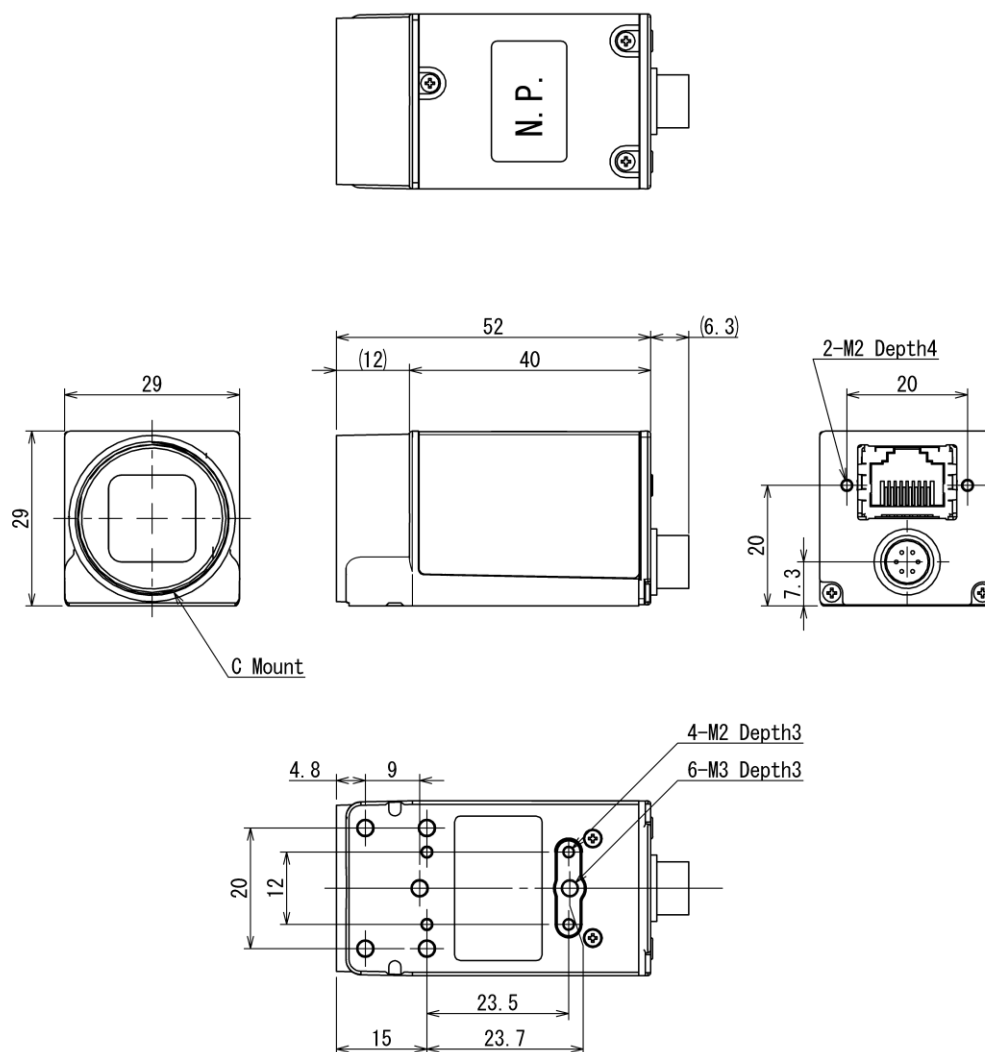

```
Specification
 Material
  Lens mount,Rear panel : Alminum die cast
  Cover                 : Anticorrosion aluminum alloy
 Processing
  Lens mount,Cover,Rear panel : Anion coating(black)
```

[Unit : mm]

D4288264C
https://www.toshiba-teli.co.jp/en/

General Specifications

Model Name		BG040M	BG160M
Imager		CMOS image sensor	
Maximum number of Video out pixels (H) x (V)		720×540	1440×1080
Optical Size		1/2.9 type	1/2.9 type
Scanning area (H) x (V) [mm]		5.02×3.82	5.00×3.75
Pixel size (H) x (V) [μm]		6.90×6.90	3.45×3.45
Scan method		Progressive	
Electronic shutter method		Global shutter	
Aspect ratio		4:3	
Sensitivity		1890lx, F5.6, 1/333s	1700lx, F5.6, 1/77s
Minimum illuminance		Ver.3.3.9 or earlier：F1.4, Gain +24dB, Video level 50%	
		Ver.3.5.0 or later　：F1.4, Gain +36dB, Video level 50%	
		Ver.3.3.9 or earlier：4lx Ver.3.5.0 or later　：1lx	Ver.3.3.9 or earlier：4lx Ver.3.5.0 or later　：1lx
Power supply		PoE (Power over Ethernet IEEE802.3af compliant) or DC+12V ±10% (ripple 100 mV(p-p) or less)	
Power consumption(*1)	PoE	3.3W(max)	
	DC12V	2.7W(max)	
Interface		Gigabit Ethernet IEEE802.3ab (1000BASE-T) conformity	
Transmission speed		1Gbps (Maximum)	
Protocol		GigEVision Camera Interface Standard for Machine Vision Ver 1.2	
Conformity cable		Twist pair (Category 5e or over)	
Cable length		Up to 100m (at the Unshielded Twist Pair (UTP) cable)	
Image output format		GVSP_PIX_MONO8: Mono 8bit	
		GVSP_PIX_MONO10: Mono 10bit	
		GVSP_PIX_MONO12: Mono 12bit	
Maximum Frame rate (at all pixels readout)		291fps @Mono 8bit 145fps @Mono 10bit/12bit	72fps @Mono 8bit 36fps @Mono 10bit/12bit
Dimensions		29 mm(W) x 29 mm (H) x 40 mm (D) (Not including protrusion)	
Mass		Approximately 59g	
Lens mount		C mount	
Flange back		17.526mm	
Camera body grounding insulation status		Non-conductive between circuit GND and camera body	

(*1) at all pixels readout

D4288264C

https://www.toshiba-teli.co.jp/en/
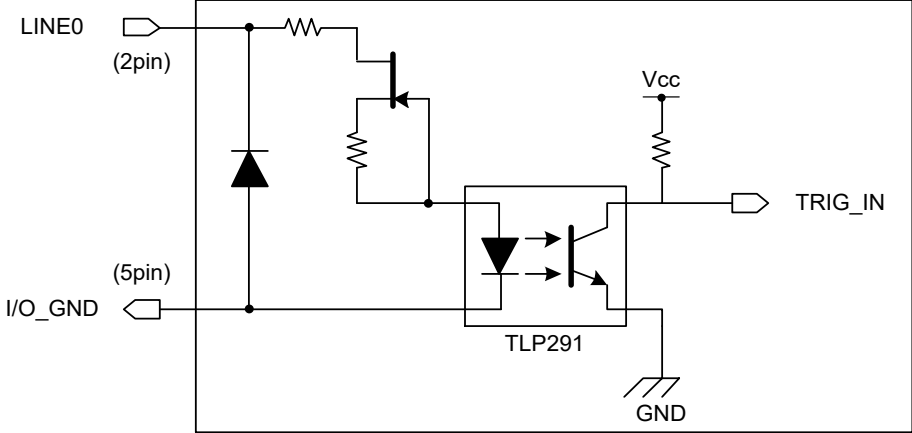
Notes on combination of C-mount lens:
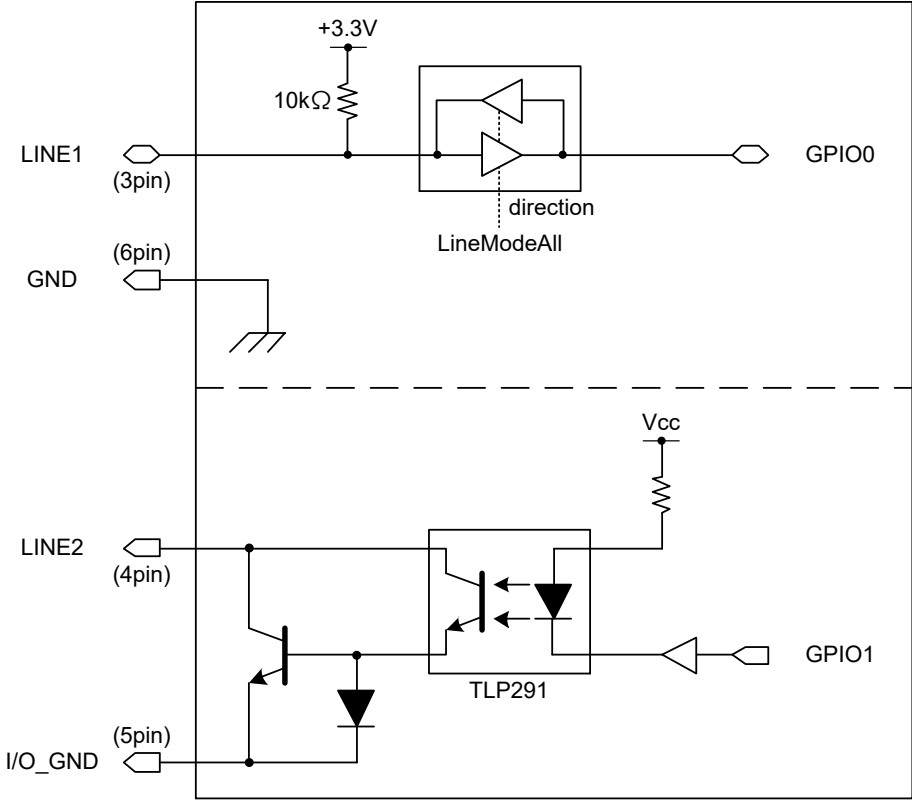
- Depending on the lens you use, the performance of the camera may not be brought out fully due to the deterioration in resolution and brightness in the peripheral area, occurrence of a ghost, aberration and others. When you check the combination between the lens and camera, be sure to use the lens you actually use.

- In addition, use a mounting screw free from defects and dirt. Otherwise, the camera may be unable to be removed.

- As for the C-mount lens used combining this camera, the projection distance from bottom of the screw should use 10mm or less.

Bottom of
the screw

C-mount lens

10mm or less
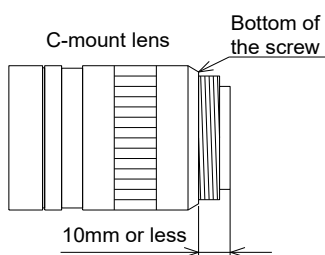
I/O Specification

Line name	Line0
Direction	IN
Input signal	External trigger
Input level	Low: 0 to 0.5V, High: 3.3 to 24.0V
Input current	5 to 15mA
Pulse width	200µs (minimum)
Input circuit	Photo coupler input
	  * Depending on cable length, cable kinds and input current of trigger input line, Random Trigger Shutter operation may not satisfy timing specification or camera may not receive External trigger signal.
Polarity	High active / Low active (factory setting: Low active)

D4288264C

Line name	Line1	Line2
Direction	IN/OUT	OUT
Input signal	External trigger	-
Output signal	TIMER0 ACTIVE, USER OUTPUT, EXPOSURE ACTIVE, FRAME ACTIVE, FRAME TRANSFER, FRAME TRIGGER WAIT	
Maximum voltage	3.3V	24.0V
Maximum current	+/-24mA (drive current)	50mA (input current)
Input / Output circuit	LVTTL	Open Collector
Input / Output circuit diagram		
Direction control	LineModeAll register (factory setting: IN)	-
Polarity	High active / Low active (factory setting: Low active)	
TIMER0 ACTIVE		
Delay	0 to 2000000µs (factory setting: 0s)	
Duration	0 to 2000000µs (factory setting: 0s)	
TimerTriggerSource	Line0Active, ExposureStart, FrameTrigger	

Notes of external trigger signal:

- Depending on cable length, kind of cable and input current of trigger input line, external trigger signal may not be accepted by camera.

- Line0 and Line1 have a different input level. Please use input level within the voltage described in this specification.

GPIO recommended circuit:

GPIO Input

Your System ←   → Camera Inside

LINE0 (2pin) — Active Low Pulse
3.3V～24.0V
Active High Pulse
TRIG_IN
Vcc
TLP291
I/O_GND (5pin)
Your Secondary GND
Camera Signal GND

LINE1 (3pin) — Active Low Pulse
+3.3V
10kΩ
Active High Pulse
direction=IN
LineModeAll
GPIO0
GND (6pin)
Your Signal GND
Camera Signal GND

GPIO Output

Your System ←   → Camera Inside

Active Low Pulse
LVTTL
LINE1 (3pin) — Active Low Pulse
+3.3V
10kΩ
direction=OUT
LineModeAll
GPIO0
GND (6pin)
Your Signal GND
Camera Signal GND

3.3V～24.0V
Active Low Pulse
LINE2 (4pin) — Active Low Pulse
Vcc
TLP291
GPIO1
I/O_GND (5pin)
Your Secondary GND

- Camera GND (Pin 6) and I/O_GND (Pin 5) are mutually isolated.

  It is also possible to connect them to common GND of your system.

- Camera GND (Pin 6) and I/O_GND (Pin 5) are isolated from camera frame.

  In using shield cable, the shield wire shall be connected to your frame ground or Camera GND (Pin 6).

- Please confirm the EMC adaptability in whole of your system.

D4288264C

https://www.toshiba-teli.co.jp/en/

● I/O Timing

- External trigger signal input

  External trigger signal input circuits of LINE0, LINE1 are different.

  Internal signal delay is dependent on LINE.

(a) Negative

(b) Positive
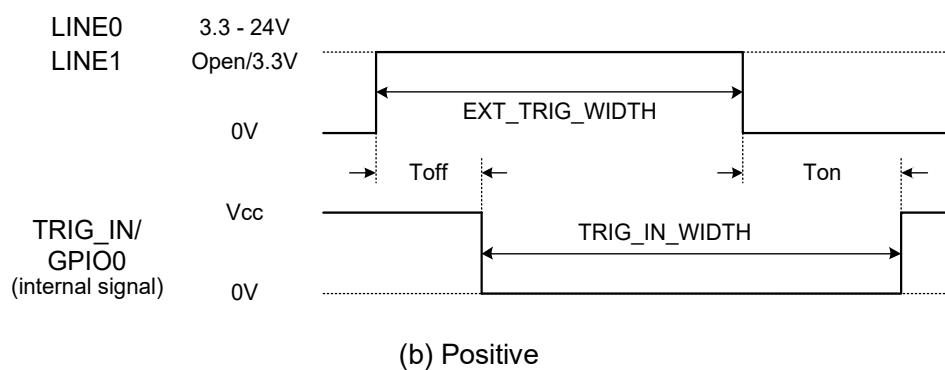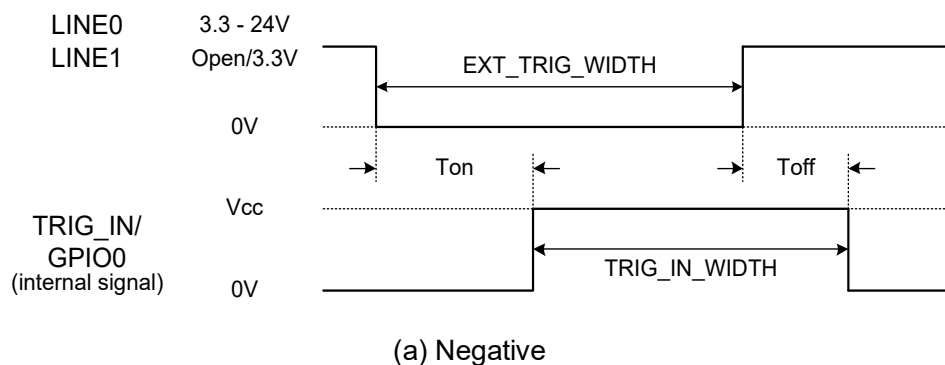
Trigger Signal Input Delay

EXT_TRIG_WIDTH   : The pulse width of the external trigger input.

Toff   : The delay time of falling edge.

Ton   : The delay time of rising edge.

TRIG_IN_WIDTH   : The pulse width of the trigger signal which is received inside of the camera.

    - Negative trigger : TRIG_IN_WIDTH = EXT_TRIG_WIDTH - (Ton - Toff)

    - Positive trigger : TRIG_IN_WIDTH = EXT_TRIG_WIDTH + (Ton - Toff)

	Signal Amplitude	T_{off} [µs]	T_{on} [µs]
LINE0	+3.3V	2.6	22
	+12V	2.0	25
	+24V	1.9	26
LINE1	Open/+3.3V	<0.02	<0.02

* Measured with 2.2k ohm pulled up register on LINE0.

* Toff and Ton are typical value. These values vary depending on operating environment.

- GPIO signal output

  GPIO signal output circuits of LINE1, LINE2 are different.

  Output signal delay is dependent on LINE.

LINE1 (LVTTL) LVTTL / 0V

LINE2 (OpenCollector) OpenCollector / 0V

Toff   Ton

(a) ActiveLow

LINE1 (LVTTL) LVTTL / 0V

LINE2 (OpenCollector) OpenCollector / 0V

Ton   Toff

(b) ActiveHigh
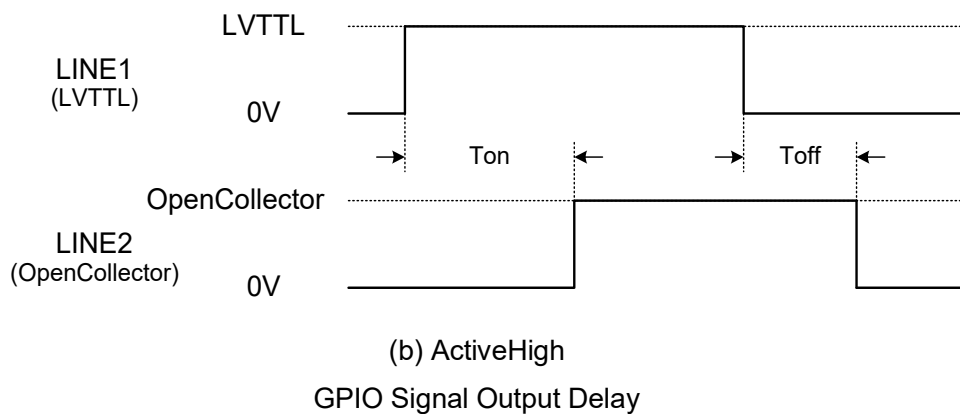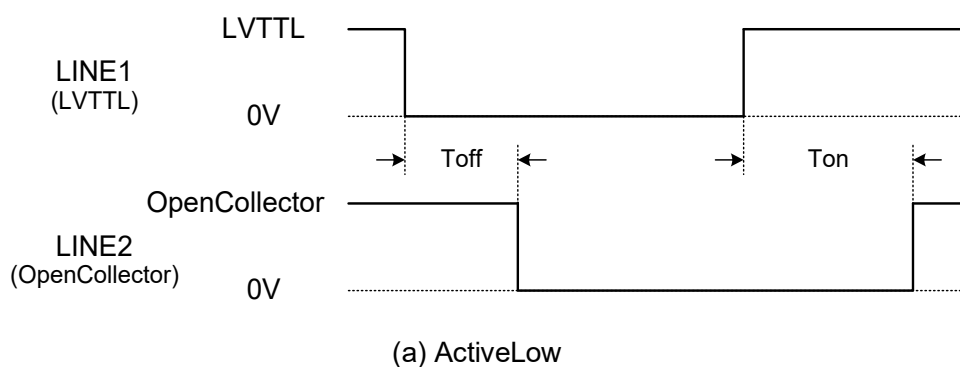
GPIO Signal Output Delay

Toff   : LINE1~LINE2 falling delay time

Ton   : LINE1~LINE2 rising delay time

Pull-up amplitude	Toff [μs]	Ton [μs]
+3.3V	3	22
+12V	4	26
+24V	5	26

* Measured with 2.2k ohm pulled up register on LINE2.

* Toff and Ton are typical value. These values are changed in operating environment.

D4288264C

https://www.toshiba-teli.co.jp/en/

Timing Specification

Image data outputs are transferred with the UDP protocol of Gigabit Ethernet. Timing numerical value below is described by absolute prerequisite that camera can use transmission band without restriction of other node. When there is other node using the same network, the value described below is not guaranteed.

● GbE bus transfer timing (at all pixels readout)

Image stream output
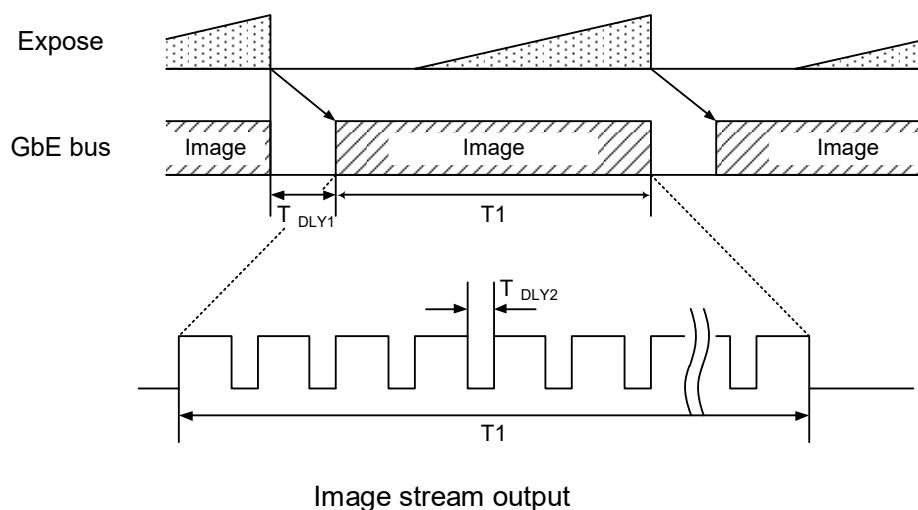
Model Name	T1 [ms]	Default Frame Rate [ms]
BG040M	Maximum frame rate (in [ms])	3.43
BG160M	of operating mode.	13.7

* T_{DLY1} : BlockStartDelay
* T_{DLY2} : SCPD (Stream Channel Packet Delay)

   BlockStartDelay and SCPD are able to be controlled by application software.

* Depending on Frame Rate and PacketSize setting, T_{DLY2} might be inserted automatically despite SCPD=0.
* Timing chart is at the time of 1000BASE-T connection.

● Random Trigger Shutter Operation

- Edge mode / Bulk mode (at all pixels readout)

Exposure time = ExposureTime register value

- Level mode (at all pixels readout)

Exposure time = Pulse width
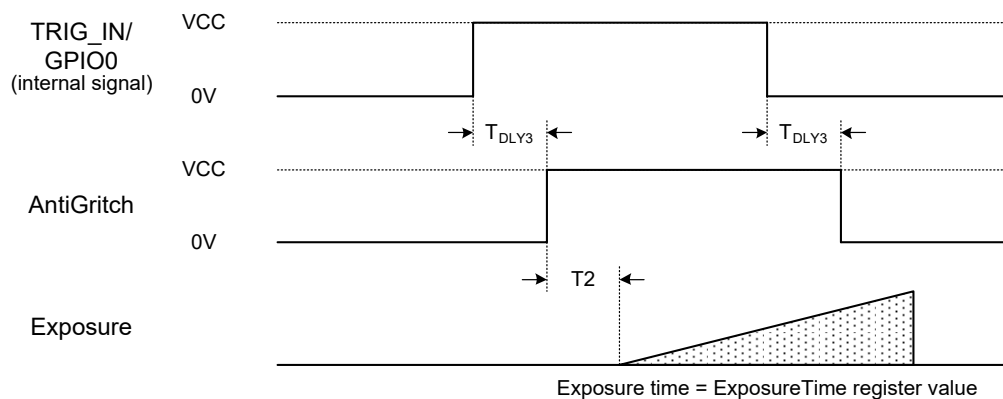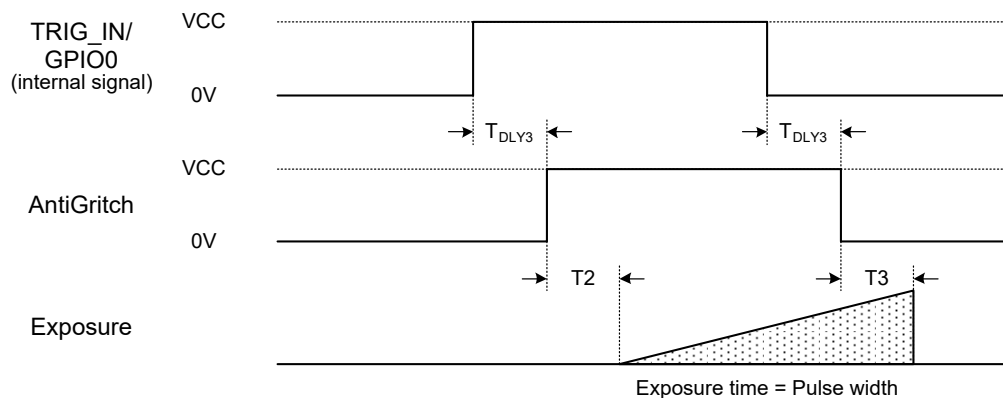
Random Trigger Shutter operation

Model Name	T2 [μs]	T3 [μs]
BG040M	12.1	26.3
BG160M	12.1	26.4

* T_{DLY3} : Delay time is the same as AntiGlitch setting value.

AntiChattering does not effect delay time.

* T2 and T3 are typical value.

Notes of random trigger shutter mode:

- In the period when FRAME_TRIGGER_WAIT (GPIO signal) is inactive, user must not input external trigger signal to this camera.

- When the interval of the input trigger signal is extremely short, or when the trigger signal is noisy, there is a possibility of causing the malfunction. In this case, please input a proper trigger signal.

Typical Spectral Response

* The lens characteristics and light source characteristics is not reflected in table.

< BG040M / BG160M >

https://www.toshiba-teli.co.jp/en/
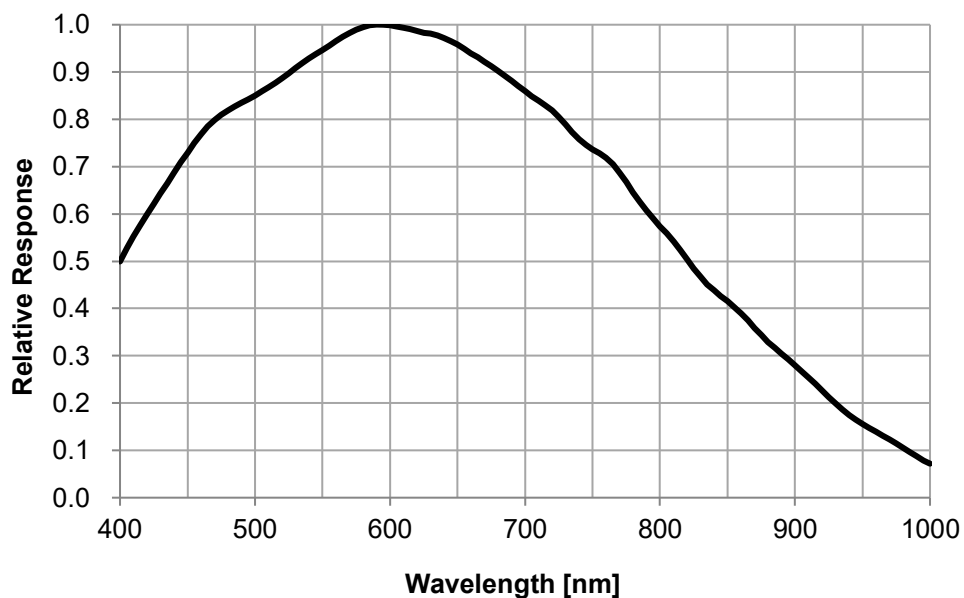
Operating Ambient Conditions

● Ambient conditions

   - Operating Assurance

       Temperature: 0°C ~ 40°C, Camera housing temperature: less than 60 °C

       Humidity: 10% ~ 90% (no condensation)

   - Storage Assurance

       Temperature: -20°C ~ 60°C

       Humidity: 90% or less (no condensation)

Notes on Heat Radiation:

About the upper limit of top surface temperature of camera housing and the allowed ambient temperature of each

model, please refer to the "Thermal design manual" on our website.

  Thermal Design Guide Line

    https://www.toshiba-teli.co.jp/en/products/industrial/info/t/t0001.htm

---

● EMC Conditions

   - EMI (Electro-Magnetic Interference):   EN61000-6-4

                                          FCC Part 15 Subpart B Class A

   - EMS (Electro-Magnetic Susceptibility):   EN61000-6-2

Notes on Conformity of the EMC:

The adaptability of the safety standard of this camera is assured in the condition of combination with the following parts:

<< PoE operation>>

   - PoE Switch   GS108PE   (NETGEAR Inc.)

   - LAN Cable   LD-TWST/BM30   (ELECOM CO., LTD)

<< DC operation>>

   - DC Cable   CPCBG-03

   - LAN Cable   LD-TWST/BM30   (ELECOM CO., LTD)

Please confirm the EMC adaptability when it combines with parts other than them.

Functions

This section introduces standard functions. BG040M and BG160M provide following functions.

Category	Function	
TransportLayerControl	TransportLayerControl	GigE Vision Bootstrap registers
DeviceControl	DeviceControl	Device information
ImageFormatControl	ImageFormatSelector	Image format selection
	Scalable	Scalable control
	Binning	Binning control
	Decimation	Decimation control
	Reverse	Image flip
	PixelFormat	Pixel format selection
	TestPattern	Test pattern control
AcquisitionControl	AcquisitionControl	Image stream start / stop
	ImageBuffer	Image buffer control
	TriggerControl	Trigger control
	ExposureTime	Exposure time control
DigitalIOControl	DigitalIOControl	GPIO signal control
VenderUniqueControl	AntiGlitch	AntiGlitch control
	AntiChattering	AntiChattering control
CounterAndTimerControl	TimerControl	Timer0Active signal control
AnalogControl	Gain	Gain control
	BlackLevel	Black level control
	Gamma	Gamma correction
	Sharpness	Sharpness control
ALCControl	ALCControl	ALC control
LUTControl	LUTControl	LUT control
SequentialShutterControl	SequentialShutterControl	Sequential shutter control
DPCControl	DPCControl	Defect pixel correction control
UserSetControl	UserSetControl	Load / Save user setting
EventControl	EventControl	Event packet control
ChunkDataControl	ChunkDataControl	Chunk data control

* BG040M and BG160M series have different register map from other BG series.

Please refer to [BG Series Register Map].

https://www.toshiba-teli.co.jp/en/

Features supported by each model are as follows.

Function	BG040M	BG160M
TransportLayerControl	○	○
DeviceControl	○	○
ImageFormatSelector	○	○
Scalable	○	○
Binning	○	○
Decimation	○	○
Reverse	○	○
PixelFormat	○	○
TestPattern	○	○
AcquisitionControl	○	○
ImageBuffer	○	○
TriggerControl	○	○
ExposureControl	○	○
DigitalIOControl	○	○
AntiGlitch	○	○
AntiChattering	○	○
TimerControl	○	○
Gain	○	○
BlackLevel	○	○
Gamma	○	○
Sharpness	○	○
ALCControl	○	○
LUTControl	○	○
SequentialShutterControl	○	○
DPCControl	○	○
UserSetControl	○	○
EventControl	○	○
Chunk	○	○

D4288264C
https://www.toshiba-teli.co.jp/en/

TransportLayerControl

This camera is based on GigE Vision.

Please refer to GigE Vision specification for details about Bootstrap Registers defined in GigE Vision.

AIA (Automated Imaging Association) GigE Vision Homepage.

http://www.visiononline.org/vision-standards-details.cfm?type=5

Followings are commonly used registers.

● GenICam Node

Register	Bootstrap Address	Length Byte / [bit]	GenICam Interface	Access	Description
GevVersionMajor	0x0000	[0:15]	IInteger	R	Major version of the specification.
GevVersionMinor	0x0000	[16:31]	IInteger	R	Minor version of the specification.
GevDeviceModeIsBigEndian	0x0004	[0]	IBoolean	R	Endianess of the device registers.
GevDeviceClass	0x0004	[1:3]	IEnumeration	R	Returns the class of the device.
GevDeviceModeCharacterSet	0x0004	[24:31]	IEnumeration	R	Character set used by all the strings of the bootstrap registers.
GevInterfaceSelector	-	4	IInteger	R	Selects which logical link to control.
GevMACAddress	0x0008	6	IInteger	R	MAC address of the logical link.
GevSupportedOptionSelector	-	-	IEnumeration	R/W	Selects the GEV option to interrogate for existing support.
GevSupportedOption	0x0010 0x092C 0x0930 0x0934 0x0D20	[29:31] [0] [0] [0:10] [25:31] [0:1] [30:31]	IBoolean	R	Returns if the selected GEV option is supported.
GevCurrentIPConfigurationLLA	0x0014	[29]	IBoolean	RW	Controls whether the Link Local Address IP configuration scheme is activated on the given logical link.
GevCurrentIPConfigurationDHCP	0x0014	[30]	IBoolean	RW	Controls whether the DHCP IP configuration scheme is activated on the given logical link.
GevCurrentIPConfigurationPersistentIP	0x0014	[31]	IBoolean	RW	Controls whether the Persistent IP configuration scheme is activated on the given logical link.
GevCurrentIPAddress	0x0024	4	IInteger	R	Reports the IP address for the given logical link.
GevCurrentSubnetMask	0x0034	4	IInteger	R	Reports the subnet mask of the given logical link.
GevCurrentDefaultGateway	0x0044	4	IInteger	R	Reports the default gateway IP address to be used on the given logical link.
GevFirstURL	0x0200	512	IString	R	Indicates the first URL to the GenICam XML device description file.

D4288264C
https://www.toshiba-teli.co.jp/en/

Register	Bootstrap Address	Length Byte / [bit]	GenICam Interface	Access	Description
GevSecondURL	0x0400	512	IString	R	Indicates the second URL to the GenICam XML device description file.
GevNumberOfInterfaces	0x0600	4	IString	R	Indicates the number of physical network interfaces supported by this device.
GevPersistentIPAddress	0x064C	4	IInteger	RW	Controls the Persistent IP address for this logical link.
GevPersistentSubnetMask	0x065C	4	IInteger	RW	Controls the Persistent subnet mask associated with the Persistent IP address on this logical link.
GevPersistentDefaultGateway	0x066C	4	IInteger	RW	Controls the persistent default gateway for this logical link.
GevLinkSpeed	0x0670	4	IInteger	R	Indicates the speed of transmission negotiated by the given logical link.
GevMessageChannelCount	0x0900	4	IInteger	R	Indicates the number of message channels supported by this device.
GevStreamChannelCount	0x0904	4	IInteger	R	Indicates the number of stream channels supported by this device.
GevHeartbeatTimeout	0x0938	4	IInteger	RW	Controls the current heartbeat timeout in milliseconds.
GevTimestampTickFrequency	0x093C	8	IInteger	R	Indicates the number of timestamp ticks in 1 second (frequency in Hz).
GevTimestampControlLatch	0x0944	[1]	ICommand	W	Latches the current timestamp counter into GevTimestampValue.
GevTimestampControlReset	0x0944	[0]	ICommand	W	Resets the timestamp counter to 0.
GevTimestampValue	0x0948	8	IInteger	R	Returns the latched 64-bit value of the timestamp counter.
GevDiscoveryAckDelay	0x0950	4	IInteger	R	Indicates the maximum randomized delay the device will wait to acknowledge a discovery command.
GevGVCPExtendedStatusCodesSelector	-	-	IEnumeration	R	Selects the GigE Vision version to control extended status codes for.
GevGVCPExtendedStatusCodes	0x0954	[29]	IBoolean	RW	Enables the generation of extended status codes.
GevGVCPHeartbeatDisable	0x0954	[31]	IBoolean	RW	Disables the GVCP heartbeat.
GevCCP	0x0A00	4	IEnumeration	RW	Controls the device access privilege of an application.
GevPrimaryApplicationSocket	0x0A04	[16:31]	IInteger	R	Returns the UDP source port of the primary application.
GevPrimaryApplicationIPAddress	0x0A14	4	IInteger	R	Returns the address of the primary application.
GevMCPHostPort	0x0B00	[16:31]	IInteger	RW	Controls the port to which the device must send messages.
GevMCDA	0x0B10	4	IInteger	RW	Controls the destination IP address for the message channel.
GevMCTT	0x0B14	4	IInteger	RW	Provides the transmission timeout value in milliseconds.
GevMCRC	0x0B18	4	IInteger	RW	Controls the number of retransmissions allowed when a message channel message times out.
GevMCSP	0x0B1C	4	IInteger	R	This feature indicates the source port for the message channel.
GevStreamChannelSelector	-	-	IInteger	R	Selects the stream channel to control.
GevSCPDirection	0x0D00	[0]	IEnumeration	R	Reports the direction of the stream channel.
GevSCPInterfaceIndex	0x0D00	[12:15]	IInteger	R	Index of the logical link to use.

Register	Bootstrap Address	Length Byte / [bit]	GenICam Interface	Access	Description
GevSCPHostPort	0x0D00	[16:31]	IInteger	RW	Controls the port to which a GVSP transmitter must send data stream or the port from which a GVSP receiver may receive data stream.
GevSCPSFireTestPacket	0x0D04	[0]	IBoolean	RW	Sends a test packet.
GevSCPSDoNotFragment	0x0D04	[1]	IBoolean	RW	The state of this feature is copied into the do not fragment bit of IP header of each stream packet.
GevSCPSBigEndian	0x0D04	[2]	IBoolean	RW	Endianess of multi-byte pixel data for this stream.
GevSCPSPacketSize	0x0D04	[16:31]	IInteger	RW	Specifies the stream packet size, in bytes, to send for a GVSP transmitter or specifies the maximum packet size supported by a GVSP receiver.
GevSCPD	0x0D08	4	IInteger	RW	Controls the delay (in timestamp counter unit) to insert between each packet for this stream channel.
GevSCDA	0x0D18	4	IInteger	RW	Controls the destination IP address to which a GVSP transmitter must send data stream or the destination IP address from which a GVSP receiver may receive data stream.
GevSCSP	0x0D1C	4	IInteger	R	Indicates the source port of the stream channel.
PayloadSize	-	-	IInteger	R	Provides the number of bytes transferred for each image or chunk on the stream channel.
BlockStartDelay	-	-	IInteger	RW	Specifies the delay between each frame for the given stream channel.
GevIPConfigurationStatus	-	-	IInteger	R	Reports the current IP configuration status.

● IIDC2 Register

Register	Address	Length Byte / [bit]	Access	Description
PayloadSize	0x20103C	4	R	Provides the number of bytes transferred for each image or chunk on the stream channel.
BlockStartDelay	0x20105C	4	R/W	Specifies the delay between each frame for the given stream channel.
GevIPConfigurationStatus	0x20107C	4	R	Reports the current IP configuration status.

● Note

CurrentSpeed register could be used for showing the warning message on your application when camera is connected to 100Base-T.

D4288264C
https://www.toshiba-teli.co.jp/en/

DeviceControl

Registers of this category provide various information of the camera.

● Registers

Register	Address	GenICam Interface	Length Byte / [bit]	Access	Description
DeviceVendorName	0x200070	String	16	R	Name of the manufacturer of the device.
DeviceModelName	0x200090	String	16	R	Model of the device.
DeviceManufacturerInfo	0x2000B0	String	16	R	Manufacturer information about the device.
DeviceVersion	0x2000D0	String	16	R	Version of the device.
DeviceFirmwareVersion	0x2000F0	String	16	R	Same as DeviceVersion in Bootstrap Registers
DeviceID	0x200110	String	16	R	Device's serial number.
DeviceUserID	-	String	16	RW	User-programmable device identifier.
DeviceTLType	-	IEnumeration	4	R	Transport Layer type of the device.
DeviceStreamChannelSelector	-	IInteger	4	R	Same as GevStreamChannelSelector in Transport Layer Control
DeviceStreamChannelType	-	IEnumeration	4	R	Same as GevSCPDirection in Transport Layer Control
DeviceStreamChannelEndianness	-	IBoolean	4	R	Same as GevSCPSBigEndian in Transport Layer Control
DeviceStreamChannelPacketSize	-	IInteger	4	RW	Same as GevSCPSPacketSize in Transport Layer Control
DeviceReset	0x20003C	Command	[0]	W	Resets the device.

● Note

- **DeviceReset**

  Camera initiates link negotiation and IP re-configuration by DeviceReset command.

  - The application is required to close and re-open the camera.

ImageFormatControl

Registers of this category are related to image format control.

Camera has three different banks of image format. You can select image format by ImageFormatSelector.
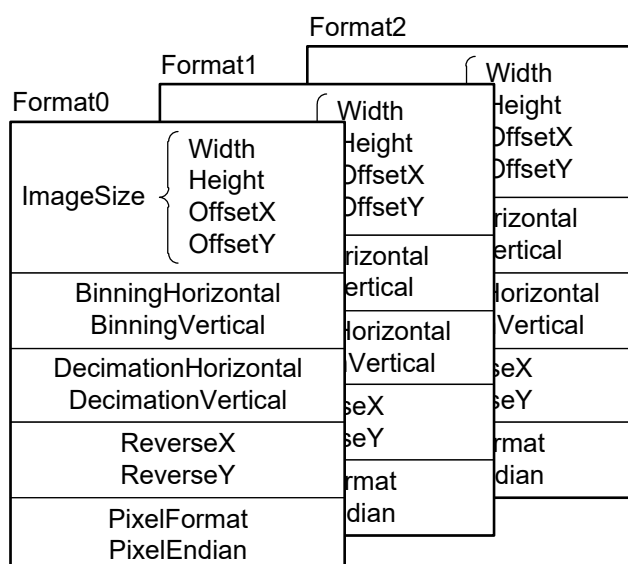
● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
ImageFormatSelector	IEnumeration	4	R/W	Selects an image format.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
ImageFormatSelector	Implemented	0x202020	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20202C	16	R	[0] : Format0 [1] : Format1 [2] : Format2
	Value	0x20203C	4	R/W	Selects an image format.

D4288264C

https://www.toshiba-teli.co.jp/en/

Table: List of registers to be applied by ImageFormatSelector

ImageFormat 0 - 2
Width
Height
OffsetX
OffsetY
BinningHorizontal
BinningVertical
DecimationHorizontal
DecimationVertical
ReverseX
ReverseY
PixelFormat
PixelEndian

● **Control with TeliCamSDK**

Camera feature API

Control ImageFormat using dedicated API.

API name	Description
GetCamImageFormatSelector	Get current ImageFormatSelector value
SetCamImageFormatSelector	Set new ImageFormatSelector value

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control ImageFormat using GenICam API.

◆ **ImageFormat**

Select an image format by 'ImageFormatSelector'.

Integer value and string value of Enumeration are as follows.

Integer	String
0 (*)	Format0
1	Format1
2	Format2

* initial factory setting

```
// GenICam node handle
CAM_NODE_HANDLE    hSelector = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "ImageFormatSelector", &hSelector);

// ImageFormat = Format2
Nd_SetEnumStrValue(s_hCam, hSelector, "Format2");
```

Please refer to [INode functions], [IEnumeration node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control ImageFormat by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ ImageFormat

Write to 'Value' field of 'ImageFormatSelector' register.

```
// ImageFormat = Format2
uint32_t       uiSelector;
uiSelector = 2;
Cam_WriteReg(s_hCam, 0x20203C, 1, &uiSelector);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

● Note

- Changing 'ImageFormatSelector' register value is invalid during image stream data output.

Scalable

Scalable function reads out the region of interest (ROI) of the sensor.

If height size is set small, it is possible to increase the frame rate.

Only single rectangle is selectable. Concave or convex shape is not selectable.

- Window size: $\{A + 4 \times m \text{ (H)}\} \times \{B + 2 \times n \text{ (V)}\}$

                    A, B = minimum unit size

                    m, n = integer

                    The window size is equal or less than maximum image size.

- Start address: $\{4 \times i \text{ (H)}\} \times \{2 \times j \text{ (V)}\}$

                    i, j = integer

                    The window size is equal or less than maximum image size.
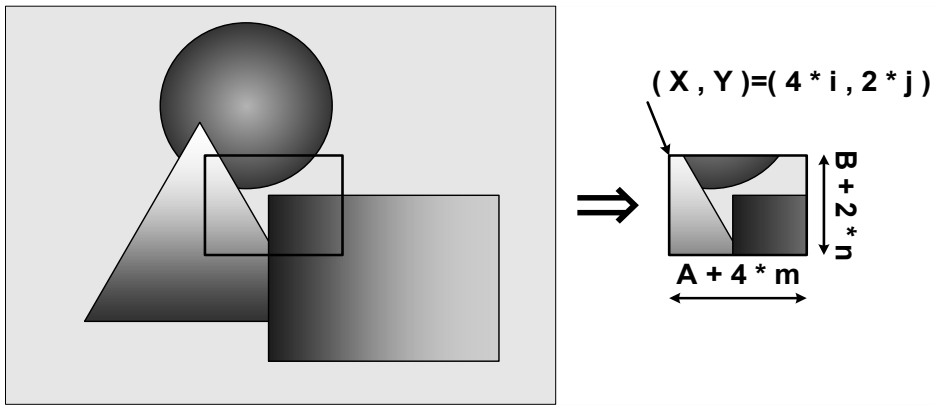
Scalable

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
Width	IInteger	4	R/W	Sets width (in pixels) of the image data.
Height	IInteger	4	R/W	Sets Height (in pixels) of the image data.
OffsetX	IInteger	4	R/W	Sets horizontal offset (in pixels) from the origin to the region of interest.
OffsetY	IInteger	4	R/W	Sets vertical offset (in pixels) from the origin to the region of interest.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
ImageSize	Implemented	0x202060	[31]	R	Returns the state whether the function is implemented.
	OffsetXMin	0x20206C	4	R	Returns the minimum starting position of the horizontal direction
	OffsetXInc	0x202070	4	R	Returns the unit starting position of the horizontal direction
	WidthMin	0x202074	4	R	Returns the minimum size of width
	WidthInc	0x202078	4	R	Returns the unit size of width
	SensorWidth	0x20207C	4	R	Returns effective Width of the sensor in pixels.
	OffsetYMin	0x202080	4	R	Returns the minimum starting position of the vertical direction
	OffsetYInc	0x202084	4	R	Returns the unit starting position of the vertical direction
	HeightMin	0x202088	4	R	Returns the minimum size of height
	HeightInc	0x20208C	4	R	Returns the unit size of height.
	SensorHeight	0x202090	4	R	Returns effective Height of the sensor in pixels
	OffsetX	0x202094	4	RW	Sets horizontal offset (in pixels) from the origin to the region of interest
	Width	0x202098	4	RW	Sets width (in pixels) of the image data.
	OffsetY	0x20209C	4	RW	Sets vertical offset (in pixels) from the origin to the region of interest
	Height	0x2020A0	4	RW	Sets Height (in pixels) of the image data.

D4288264C

https://www.toshiba-teli.co.jp/en/

● Control with TeliCamSDK

Camera feature API

Control Scalable using dedicated API.

API name	Description
GetCamSensorWidth	Get effective width of the sensor in pixels
GetCamSensorHeight	Get effective height of the sensor in pixels
GetCamRoi	Get ROI (Region of Interest) of the camera
SetCamRoi	Set ROI (Region of Interest) to the camera
GetCamWidthMinMax	Get minimum and maximum value
GetCamWidth	Get width of image
SetCamWidth	Set width of image
GetCamHeightMinMax	Get minimum and maximum value
GetCamHeight	Get height of image
SetCamHeight	Set height of image
GetCamOffsetXMinMax	Get minimum and maximum value
GetCamOffsetX	Get horizontal offset of image
SetCamOffsetX	Set horizontal offset of image
GetCamOffsetYMinMax	Get minimum and maximum value
GetCamOffsetY	Get vertical offset of image
SetCamOffsetY	Set vertical offset of image

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control Scalable using GenICam API.

◆ **Scalable**

```
// GenICam node handle
CAM_NODE_HANDLE    hWidth = NULL;
CAM_NODE_HANDLE    hHeight = NULL;
CAM_NODE_HANDLE    hOffsetX = NULL;
CAM_NODE_HANDLE    hOffsetY = NULL;

// ROI = {OffsetX, Width, OffsetY, Height};
uint64_t      ROI[] = {360,720, 270,540};

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "Width", &hWidth);
Nd_GetNode(s_hCam, "Height", &hHeight);
Nd_GetNode(s_hCam, "OffsetX", &hOffsetX);
Nd_GetNode(s_hCam, "OffsetY", &hOffsetY);

// Set ROI
Nd_SetIntValue(s_hCam, hWidth, ROI[1]);
Nd_SetIntValue(s_hCam, hOffsetX, ROI[0]);
Nd_SetIntValue(s_hCam, hHeight, ROI[3]);
Nd_SetIntValue(s_hCam, hOffsetY, ROI[2]);
```

D4288264C

https://www.toshiba-teli.co.jp/en/

If you're going to reduce width size, set Width, first. Then set OffsetX.

If you're going to increase width size, set OffsetX, first. Then set Width.

If you're going to reduce height size, set Height, first. Then set OffsetY.

If you're going to increase height size, set OffsetY, first. Then set Height.

Please refer to [INode functions] and [IInteger node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control Scalable by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **Scalable**

Write to 'Value' field of 'OffsetX', 'Width', 'OffsetY', 'Height' register.

```
// ROI = {OffsetX, Width, OffsetY, Height};
uint32_t      ROI[] = {360,720, 270,540};

// Set ROI (in one by one)
Cam_WriteReg(s_hCam, 0x202094, 1, &ROI[0]);
Cam_WriteReg(s_hCam, 0x202098, 1, &ROI[1]);
Cam_WriteReg(s_hCam, 0x20209C, 1, &ROI[2]);
Cam_WriteReg(s_hCam, 0x2020A0, 1, &ROI[3]);

// Set ROI (in block)
Cam_WriteReg(s_hCam, 0x202094, 4, &ROI[0]);
```

In IIDC2 registers access,

you can set OffsetX, Width, OffsetY, Height in any order. (in one by one access)

you can also set OffsetX, Width, OffsetY, Height with single access. (in block access)

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ **Minimum / Maximum Value**

Model	BG040M	BG160M
Width/OffsetX unit size	4	4
Height/OffsetY unit size	2	2
Minimum unit size	64 x 64	64 x 64
Maximum unit size (*)	720 x 540	1440 x 1080

* initial factory setting

Binning

In the binning mode, a pixel is added with the neighboring pixel(s).

This increases the sensitivity of the image. It's alike scalable, the frame rate can be faster and USB bandwidth occupation decrease.
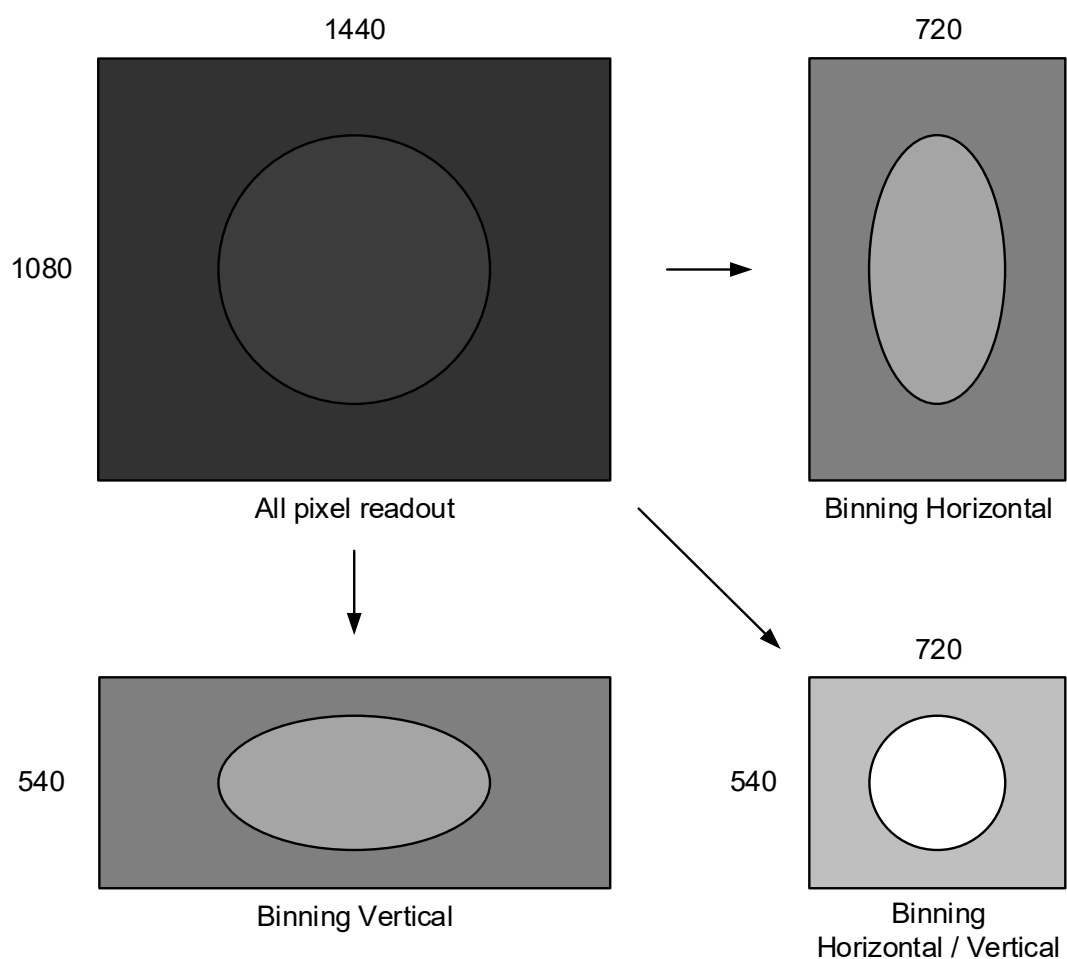
Binning operation (e.g. BG160M pixel, 2x2 binning)

D4288264C
https://www.toshiba-teli.co.jp/en/

●Framerate in each output format (fps)

- BG040M

Mono8		Horizontal	
		1	2
Vertical	1	291	437
	2	437	437

Mono10 / Mono12		Horizontal	
		1	2
Vertical	1	145	291
	2	291	320

- BG160M

Mono8		Horizontal		
		1	2	4
Vertical	1	72	145	226
	2	145	291	501
	4	291	501	501

Mono10 / Mono12		Horizontal		
		1	2	4
Vertical	1	36	72	145
	2	72	145	291
	4	145	291	317

D4288264C

https://www.toshiba-teli.co.jp/en/

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
BinningHorizontal	IInteger	4	R/W	Sets the Binning Horizontal.
BinningVertical	IInteger	4	R/W	Sets the Binning Vertical.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
BinningHorizontal	Implemented	0x202120	[31]	R	Returns the state whether the function is implemented.
	Min	0x202134	4	R	Returns the minimum value of BinningHorizontal Value register.
	Max	0x202138	4	R	Returns the maximum value of BinningHorizontal Value register.
	Value	0x20213C	4	R/W	Sets the Binning Horizontal.
BinningVertical	Implemented	0x202140	[31]	R	Returns the state whether the function is implemented.
	Min	0x202154	4	R	Returns the minimum value of BinningVertical Value register.
	Max	0x202158	4	R	Returns the maximum value of BinningVertical Value register.
	Value	0x20215C	4	R/W	Sets the Binning Vertical.

● Control with TeliCamSDK

Camera feature API

Control Binning using dedicated API.

API name	Description
GetCamBinningHorizontalMinMax	Get minimum and maximum value
GetCamBinningHorizontal	Get current BinningHorizontal value
SetCamBinningHorizontal	Set new BinningHorizontal value
GetCamBinningVerticalMinMax	Get minimum and maximum value
GetCamBinningVertical	Get current BinningVertical value
SetCamBinningVertical	Set new BinningVertical value

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

D4288264C

GenICam function API

Control Binning using GenICam API.

◆ **Binning**

```
// GenICam node handle
CAM_NODE_HANDLE    hBinning = NULL;

// Binning = 2x2
uint64_t      Binning = 2;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "BinningHorizontal", &hBinning);
// Nd_GetNode(s_hCam, "BinningVertical", &hBinning); // either will do

// Set Binning
Nd_SetIntValue(s_hCam, hBinning, Binning);
```

Please refer to [INode functions], [IInteger node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control Binning by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **Binning**

Write to 'Value' field of 'BinningHorizontal' register to control BinningHorizontal.

Or write to 'Value' field of 'BinningVertical' register to control BinningVertical.

```
// Binning = 2x2
uint32_t      Binning = 2;

// Set Binning
Cam_WriteReg(s_hCam, 0x20213C, 1, &Binning);
// Cam_WriteReg(s_hCam, 0x20215C, 1, &Binning); // either will do
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

● **Note**

- Binning is disabled when the camera is running in Decimation mode.

- Changing 'BinningHorizontal', 'BinningVertical' register value is invalid during image stream data output.

- Binning Horizontal / Vertical = 3 is not available.

Decimation

Decimation feature reads out all effective areas at high speed by skipping pixels and lines.

Decimation feature can make frame rate faster, and decrease interface bandwidth occupation.
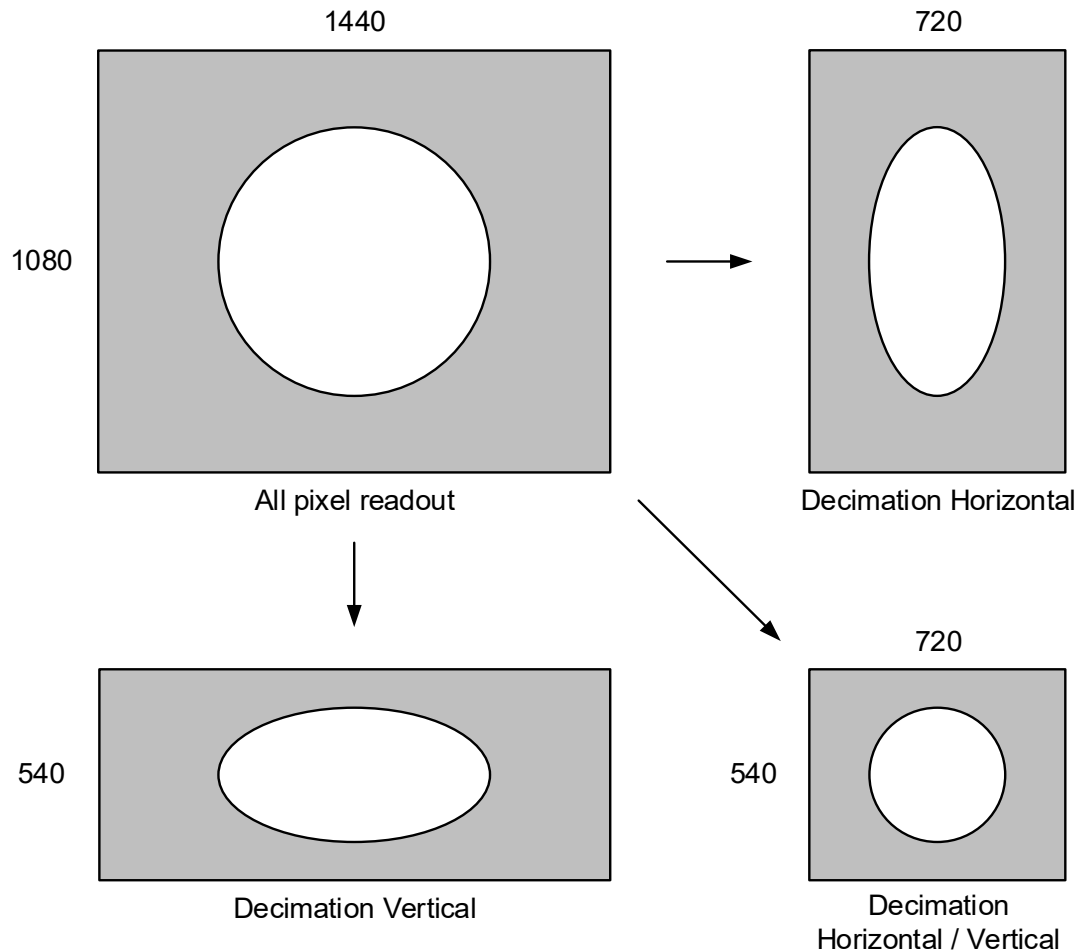
Decimation operation (e.g. BG160M, 2x2 decimation)

D4288264C
https://www.toshiba-teli.co.jp/en/

●FrameRate in each output format (fps)

- BG040M

Mono8		Horizontal	
		1	2
Vertical	1	291	437
	2	437	437

Mono10 / Mono12		Horizontal	
		1	2
Vertical	1	145	291
	2	291	320

- BG160M

Mono8		Horizontal		
		1	2	4
Vertical	1	72	145	226
	2	145	291	501
	4	226	501	501

Mono10 / Mono12		Horizontal		
		1	2	4
Vertical	1	36	72	145
	2	72	145	291
	4	145	291	317

D4288264C

https://www.toshiba-teli.co.jp/en/

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
DecimationHorizontal	IInteger	4	R/W	Sets the Decimation Horizontal.
DecimationVertical	IInteger	4	R/W	Sets the Decimation Vertical.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
DecimationHorizontal	Implemented	0x202160	[31]	R	Returns the state whether the function is implemented.
	Min	0x202174	4	R	Returns the minimum value of DecimationHorizontal Value register.
	Max	0x202178	4	R	Returns the maximum value of DecimationHorizontal Value register.
	Value	0x20217C	4	R/W	Sets the Decimation Horizontal.
DecimationVertical	Implemented	0x202180	[31]	R	Returns the state whether the function is implemented.
	Min	0x202184	4	R	Returns the minimum value of DecimationVertical Value register.
	Max	0x202188	4	R	Returns the maximum value of DecimationVertical Value register.
	Value	0x20218C	4	R/W	Sets the Decimation Vertical.

● Control with TeliCamSDK

Camera feature API

Control Decimation using dedicated API.

API name	Description
GetCamDecimationHorizontalMinMax	Get minimum and maximum value
GetCamDecimationHorizontal	Get current DecimationHorizontal value
SetCamDecimationHorizontal	Set new DecimationHorizontal value
GetCamDecimationVerticalMinMax	Get minimum and maximum value
GetCamDecimationVertical	Get current DecimationVertical value
SetCamDecimationVertical	Set new DecimationVertical value

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control Decimation using GenICam API.

◆ **Decimation**

```
// GenICam node handle
CAM_NODE_HANDLE    hDecimation = NULL;

// Decimation = 2x2
uint64_t      Decimation = 2;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "DecimationHorizontal", &hDecimation);
// Nd_GetNode(s_hCam, "DecimationVertical", &hDecimation); // either will do

// Set Decimation
Nd_SetIntValue(s_hCam, hDecimation, Decimation);
```

Please refer to [INode functions], [IInteger node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control Decimation by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **Decimation**

Write to 'Value' field of 'DecimationHorizontal' register to control DecimationHorizontal.

Or write to 'Value' field of 'DecimationVertical' register to control DecimationVertical.

```
// Decimation = 2x2
uint32_t      Decimation = 2;

// Set Decimation
Cam_WriteReg(s_hCam, 0x20213C, 1, &Decimation);
// Cam_WriteReg(s_hCam, 0x20215C, 1, &Decimation); // either will do
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

● **Note**

- Decimation is disabled when the camera is running in Binning mode.

- Changing 'DecimationHorizontal', 'DecimationVertical' register value is invalid during image stream data output.

- Decimation Horizontal / Vertical = 3 is not available.

Reverse

Reverse function flips image in horizontal and/or vertical direction.
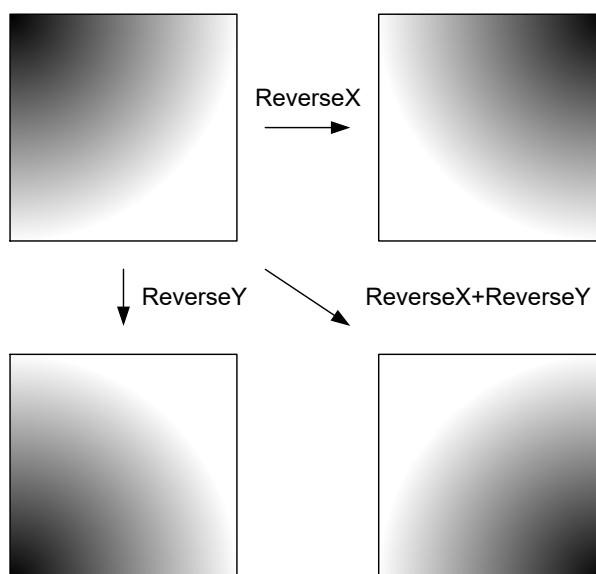
● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
ReverseX	IBoolean	4	R/W	Sets the Reverse Horizontal.
ReverseY	IBoolean	4	R/W	Sets the Reverse Vertical.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
ReverseX	Implemented	0x2021A0	[31]	R	Returns the state whether the function is implemented.
	Value	0x2021B0	4	R/W	Sets the Reverse Horizontal. [0] : Off [1] : On
ReverseY	Implemented	0x2021C0	[31]	R	Returns the state whether the function is implemented.
	Value	0x2021D0	4	R/W	Sets the Reverse Vertical. [0] : Off [1] : On

D4288264C

https://www.toshiba-teli.co.jp/en/

● Control with TeliCamSDK

Camera feature API

Control Reverse using dedicated API.

API name	Description
GetCamReverseX	Get current ReverseX value
SetCamReverseX	Set new ReverseX value
GetCamReverseY	Get current ReverseY value
SetCamReverseY	Set new ReverseY value

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control Reverse using GenICam API.

◆ **Reverse**

Sets the Reverse Horizontal by 'ReverseX'.

Sets the Reverse Vertical by 'ReverseY'.

```
// GenICam node handle
CAM_NODE_HANDLE   hReverseX = NULL;
CAM_NODE_HANDLE   hReverseY = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "ReverseX", &hReverseX);
Nd_GetNode(s_hCam, "ReverseY", &hReverseY);

// Set Reverse (flip horizontal and vertical direction)
Nd_SetBoolValue(s_hCam, hReverseX, true);
Nd_SetBoolValue(s_hCam, hReverseY, true);
```

Please refer to [INode functions], [IBoolean node functions] in [TeliCamAPI Library manual] for more detail.

D4288264C

https://www.toshiba-teli.co.jp/en/

Register access API

Control Reverse by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **Reverse**

Write to 'Value' field of 'ReverseX' register.

Write to 'Value' field of 'ReverseY' register.

```
// Set Reverse (flip horizontal and vertical direction)
uint32_t      dat = 1;
Cam_WriteReg(s_hCam, 0x2021B0, 1, &dat);
Cam_WriteReg(s_hCam, 0x2021D0, 1, &dat);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

● **Note**

- Changing 'ReverseX', 'ReverseY' register value is invalid during image stream data output.

D4288264C

https://www.toshiba-teli.co.jp/en/

PixelFormat

Select a pixel format of image stream data.

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
PixelCoding	IEnumeration	4	R/W	Selects a pixel coding.
PixelSize	IEnumeration	4	R/W	Selects a bit size of image pixel.
PixelFormat	IEnumeration	4	R/W	Selects a pixel format. PixelFormat is conformed to AIA Pixel Format Naming Convention.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
PixelCoding	Implemented	0x2020C0	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x2020CC ~0x2020D8	16	R	Returns the presence of the pixel coding. [0] : Mono
	Value	0x2020DC	4	R/W	Selects a pixel cording.
PixelSize	Implemented	0x2020E0	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x2020EC ~0x2020F8	16	R	[8] : Bpp8 [10] : Bpp10 [12] : Bpp12
	Value	0x2020FC	4	R/W	Selects a bit size of image pixel.
PixelFormat	Implemented	0x202400	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20240C ~0x202418	16	R	[1] : Mono8 [3] : Mono10 [5] : Mono12
	Value	0x20241C	4	R/W	Selects a pixel format

● **Supported PixelFormat**

PixelSize ╱ PixelCoding	Bpp8	Bpp10	Bpp12
Mono	Mono8 (*)	Mono10	Mono12
PixelFormat ID	0x01080001	0x01100003	0x01100005

* initial factory setting

● **Control with TeliCamSDK**

Camera feature API

Control PixelFormat using dedicated API.

API name	Description
GetCamPixelFormat	Get current PixelFormat value
SetCamPixelFormat	Set new PixelFormat value

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control PixelFormat using GenICam API.

◆ **PixelCoding/PixelSize**

To determine PixelFormat combination of 'PixelCoding' and 'PixelSize' register.

1. Select a pixel coding. ('PixelCoding').

   Integer value and string value of Enumeration are as follows.

Integer	String
0	Mono

2. Select a bit size of image pixel ('PixelSize').

Integer value and string value of Enumeration are as follows.

Integer	String
8	Bpp8
10	Bpp10
12	Bpp12

```
// GenICam node handle
CAM_NODE_HANDLE   hCoding = NULL;
CAM_NODE_HANDLE   hSize = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "PixelCoding", &hCoding);
Nd_GetNode(s_hCam, "PixelSize", &hSize);

// 1.Select a pixel coding.
Nd_SetEnumStrValue(s_hCam, hCoding, "Mono");
// 2.Select a pixel size.
Nd_SetEnumStrValue(s_hCam, hSize, "Bpp10");
```

◆   **PixelFormat**

1. Select a pixel format. ('PixelFormat').

Integer value and string value of Enumeration are as follows.

Integer	String
1	Mono8
3	Mono10
5	Mono12

```
// GenICam node handle
CAM_NODE_HANDLE   hFormat = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "PixelFormat", &hFormat);

// 1.Select a pixel format.
Nd_SetEnumStrValue(s_hCam, hFormat, "Mono10");
```

Please refer to [INode functions], [IEnumeration node functions] in [TeliCamAPI Library manual] for more detail.

D4288264C

https://www.toshiba-teli.co.jp/en/

Register access API

Control PixelFormat by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ PixelCoding/PixelSize

To determine PixelFormat combination of 'PixelCoding' and 'PixelSize' register.

1. Write to 'Value' field of 'PixelCoding' register.

2. Write to 'Value' field of 'PixelSize' register.

```
uint32_t coding = 0; // Mono
uint32_t size = 10; // Bpp10

// 1.Select a pixel coding.
Cam_WriteReg(s_hCam, 0x2020DC, 1, &coding);
// 2.Select a pixel size.
Cam_WriteReg(s_hCam, 0x2020FC, 1, &size);
```

◆ PixelFormat

1. Write to 'Value' field of 'PixelFormat' register.

```
uint32_t format = 3; // Mono10

// 1.Select a pixel format.
Cam_WriteReg(s_hCam, 0x20241C, 1, &format);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

● Note

- Changing 'PixelCoding', 'PixelSize' and 'PixelFormat' register value is invalid during image stream data output.

TestPattern

Camera supports test pattern data output. Following test patterns are available;
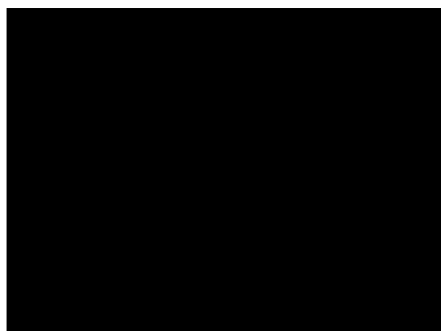
Black = All pixels 0 LSB @ 8-bit

White = All pixels 255 LSB @ 8-bit
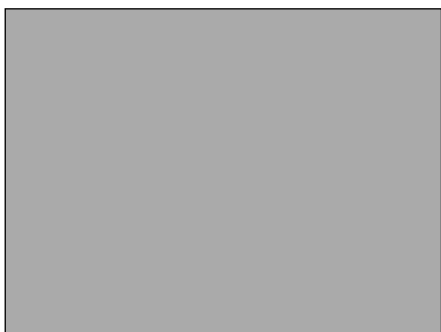
Grey A = All pixels 170 LSB @ 8-bit

Grey B = All pixels 85 LSB @ 8-bit
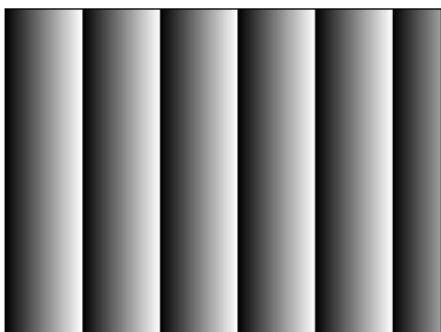
Horizontal ramp waveform

Vertical ramp waveform

Grey scale

Test pattern (e.g. BG160M @Mono8)

D4288264C

https://www.toshiba-teli.co.jp/en/

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
TestPattern	IEnumeration	4	R/W	Selects a Test pattern.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
TestPattern	Implemented	0x21F120	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x21F12C ~0x21F138	16	R	[0] : Off [1] : Black [2] : White [3] : GreyA [4] : GreyB [5] : GreyHorizontalRamp [6] : GreyScale [8] : GreyVerticalRamp
	Value	0x21F13C	4	R/W	Selects a test pattern.

● Control with TeliCamSDK

Camera feature API

Control TestPattern using dedicated API.

API name	Description
GetCamTestPattern	Get current TestPattern value
SetCamTestPattern	Set new TestPattern value

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

D4288264C

https://www.toshiba-teli.co.jp/en/

GenICam function API

Control TestPattern using GenICam API.

◆ **TestPattern**

Select a test pattern.

Integer value and string value of Enumeration are as follows.

Integer	String	Function
0 (*)	Off	Test pattern disable(Normal data output)
1	Black	All pixel = 0 LSB
2	White	All pixel = 255 @Mono8
3	GreyA	All pixel = 170 @Mono8
4	GreyB	All pixel = 85 @Mono8
5	GreyHorizontalRamp	Horizontal Ramp
6	GreyScale	Grey scale
8	GreyVerticalRamp	Vertical Ramp

* initial factory setting

```
// GenICam node handle
CAM_NODE_HANDLE   hNode = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "TestPattern", & hNode);

// 1.Select a test pattern.
Nd_SetEnumStrValue(s_hCam, hNode, "GreyHorizontalRamp");
```

Please refer to [INode functions], [IEnumeration node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control TestPattern by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **TestPattern**

Write to 'Value' field of 'TestPattern' register.

```
uint32_t dat = 5; // Horizontal Ramp

// 1.Select a test pattern.
Cam_WriteReg(s_hCam, 0x21F13C, 1, &dat);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

https://www.toshiba-teli.co.jp/en/

AcquisitionControl

AcquisitionControl features are related to image acquisition.

Camera starts image stream output by receiving AcquisitionStart command.

Camera stops image stream output by receiving AcquisitionStop command.

There are some registers that require camera to stop image stream output to change values.

Acquisition frame rate is variable. Maximum acquisition frame rate depends on camera operation mode.

(scalable, pixel format, etc.)

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
AcquisitionMode	IEnumeration	4	R/W	Selects an acquisition mode.
AcquisitionStart	ICommand	4	W	Executes the image stream output start.
AcquisitionStop	ICommand	4	W	Executes the image stream output stop.
AcquisitionAbort	ICommand	4	W	Executes the image stream output abort.
AcquisitionFrameCount	IInteger	4	R/W	Sets the number of frames to transfer in MultiFrame/ImageBuffer mode.
AcquisitionFrameRateEnable	IEnumeration	4	R/W	Selects an AcquisitionFrameRate setting priority.
AcquisitionFrameRate	IFloat	4	R/W	Sets frame rate of image stream.
AcquisitionFrameIntervalControl	IEnumeration	4	R/W	Selects an AcquisitionFrameInterval setting priority.
AcquisitionFrameInterval	IFloat	4	R/W	Sets frame interval of image stream.

D4288264C

https://www.toshiba-teli.co.jp/en/

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
AcquisitionMode	Implemented	0x203020	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20302C ~0x203038	16	R	[0]: Abort [1]: Stop [8]: Continuous [9]: MultiFrame [10]: Image Buffer Read
	Value	0x20303C	4	R/W	Selects an Acquisition mode.
AcquisitionFrameCount	Implemented	0x203040	[31]	R	Returns the state whether the function is implemented.
	Min	0x203054	4	R	Returns the minimum number of frames in MultiFrame/ImageBuffer mode.
	Max	0x203058	4	R	Returns the maximum number of frames in MultiFrame/ImageBuffer mode.
	Value	0x20305C	4	R/W	Sets the number of frames in MultiFrame/ImageBuffer mode.
AcquisitionFrameRate	Implemented	0x2030A0	[31]	R	Returns the state whether the function is implemented.
	Control	0x2030A8	4	R/W	[0] : NoSpecify [1] : Manual
	Mult	0x2030AC	4	R	Absolute Value [fps] = Raw * (Mult / Div)
	Div	0x2030B0	4	R	
	Min	0x2030B4	4	R	Returns the minimum value of FrameRate.
	Max	0x2030B8	4	R	Returns the maximum value of FrameRate.
	Value	0x2030BC	4	R/W	Sets frame rate of image stream in raw value.
AcquisitionFrameInterval	Implemented	0x2030C0	[31]	R	Returns the state whether the function is implemented.
	Control	0x2030C8	4	R/W	[0] : NoSpecify [1] : Manual
	Mult	0x2030CC	4	R	AbsoluteValue[sec] = Raw * (Mult / Div)
	Div	0x2030D0	4	R	
	Min	0x2030D4	4	R	Returns the minimum value of FrameInterval.
	Max	0x2030D8	4	R	Returns the maximum value of FrameInterval.
	Value	0x2030DC	4	R/W	Sets frame interval of image stream in raw value.

D4288264C

https://www.toshiba-teli.co.jp/en/

● Control with TeliCamSDK

Camera feature API

Control AcquisitionControl using dedicated API.

API name	Description
GetCamAcquistionFrameCountMinMax	Get minimum and maximum value
GetCamAcquistionFrameCount	Get current value
SetCamAcquistionFrameCount	Set new value
GetCamAcquistionFrameRateEnable	Get current value
SetCamAcquistionFrameRateEnable	Set new value
GetCamAcquistionFrameRateMinMax	Get minimum and maximum value
GetCamAcquistionFrameRate	Get current value
SetCamAcquistionFrameRate	Set new value

1. Set the number of frames to transfer in MultiFrame/ImageBuffer mode. (optional)

   Set 'AcquisitionFrameCount' with 'SetCamAcquistionFrameCount' function.

   *If you want to capture image stream continuously, it's not necessary to set an AcquisitionFrameCount.

2. Select an 'AcquisitionFrameRate' setting priority. (optional)

   Set 'AcquisitionFrameRateEnable' with 'SetCamAcquistionFrameRateEnable' function.

AcquisitionFrameRateEnablel	Function
False (*)	The frame rate is determined by giving priority to ExposureTime setting value.
True	The frame rate is determined by giving priority to AcquisitionFrameRate setting value.

* initial factory setting

3. Set frame rate of image stream. (optional)

   Set 'AcquisitionFrameRate' with 'SetCamAcquistionFrameRate' function.

   Or, set 'AcquisitionFrameInterval' with 'SetCamAcquistionFrameInterval' function.

   'AquisitionFrameInterval' is a reciprocal of 'AcquisitionFrameRate'.

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

4. Capture image stream.

   TeliCamAPI provides streaming functions including command handling and buffer handling to capture image stream simply.

   Please refer to [Camera streaming functions] in [TeliCamAPI Library manual]

   and [GrabStreamSimple] sample code in your TeliCamSDK install folder.

D4288264C

https://www.toshiba-teli.co.jp/en/

GenICam function API

Control AcquisitionControl using GenICam API.

1. Set the number of frames to transfer in MultiFrame/ImageBuffer mode. (optional)

   Get 'IInteger' interface handle of 'AcquisitionFrameCount' by 'Nd_GetNode'.

   Set AcquisitionFrameCount with IInteger node functions.

   (e.g. 'Nd_GetIntMin', 'Nd_GetIntMax', 'Nd_GetIntValue', 'Nd_SetIntValue')

   *If you want to capture image stream continuously, it's not necessary to set an AcquisitionFrameCount.

2. Select an 'AcquisitionFrameRate' setting priority. (optional)

   Get 'IEnumeration' interface handle of 'AcquisitionFrameRateEnable' by 'Nd_GetNode'.

   Set AcquisitionFrameRateEnable with IEnumeration node functions.

   (e.g. 'Nd_GetEnumIntValue', 'Nd_SetEnumIntValue', 'Nd_GetEnumStrValue', 'Nd_SetEnumStrValue')

   Integer value and string value of Enumeration are as follows.

Integer	String
0 (*)	False
1	True

   * initial factory setting

3. Set frame rate of image stream. (optional)

   Get 'IFloat' interface handle of 'AcquisitionFrameRate' or 'AcquisitionFrameInterval' by 'Nd_GetNode'.

   Set AcquisitionFrameRate or AcquisitionFrameInterval with IFloat node functions.

   'AquisitionFrameInterval' is a reciprocal of 'AcquisitionFrameRate'.

   (e.g. 'Nd_GetFloatMin', 'Nd_GetFloatMax', 'Nd_GetFloatValue', 'Nd_SetFloatValue')

4. Capture image stream.

   TeliCamAPI provides streaming functions including command handling and buffer handling to capture image stream simply.

   Please refer to [Camera streaming functions] in [TeliCamAPI Library manual] and [GrabStreamSimple] sample code in your TeliCamSDK install folder.

Register access API

Control AcquisitionControl by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

1. Set the number of frames to transfer in MultiFrame/ImageBuffer mode. (optional)

   Write to 'Value' field of 'AcquisitionFrameCount' register to control AcquisitionFrameCount.

https://www.toshiba-teli.co.jp/en/

2. Select an 'AcquisitionFrameRate' setting priority. (optional)

   Write to 'Control' field of 'AcquisitionFrameRate' register to control AcquisitionFrameRate setting priority.

3. Set frame rate of image stream. (optional)

   Write to 'Value' field of 'AcquisitionFrameRate' register to control AcquisitionFrameRate.

4. Capture image stream.

   TeliCamAPI provides streaming functions including command handling and buffer handling to capture image stream simply.

   Please refer to [Camera streaming functions] in [TeliCamAPI Library manual]

     and [GrabStreamSimple] sample code in your TeliCamSDK install folder.

   Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ **Minimum / Maximum Value**

AcquisitionFrameRate		BG040M	BG160M
Minimum	Raw Value	4095	
	Absolute Value	0.0625 [fps]	
Maximum	Raw Value	Depends on ImageFormatControl setting.	
	Absolute Value		
Initial Value	Raw Value	19088743	4772185
	Absolute Value	291.271 [fps]	72.818 [fps]
Formula		Absolute Value [fps] = Raw Value / 65536	

● **Note**

- Changing 'AcquisitionFrameRateEnable', 'AcquisitionFrameRate', 'AcquisitionFrameIntervalControl',

  'AcquisitionFrameInterval' register value is invalid during image stream data output.

- When exposure time setting is longer than frame rate setting, camera operation gives priority to exposure

  time setting.

Notes on Frame Drops of Image:

  Depends on your PC or interface card configurations, images may not be captured normally (e.g. frame drops may

occur). In this case, change to frame rate setting lower.

ImageBuffer

In ImageBuffer mode, Camera stores images temporarily in image buffer, and read them out in arbitrary timing.

This function is typically used in Random Trigger Shutter mode.

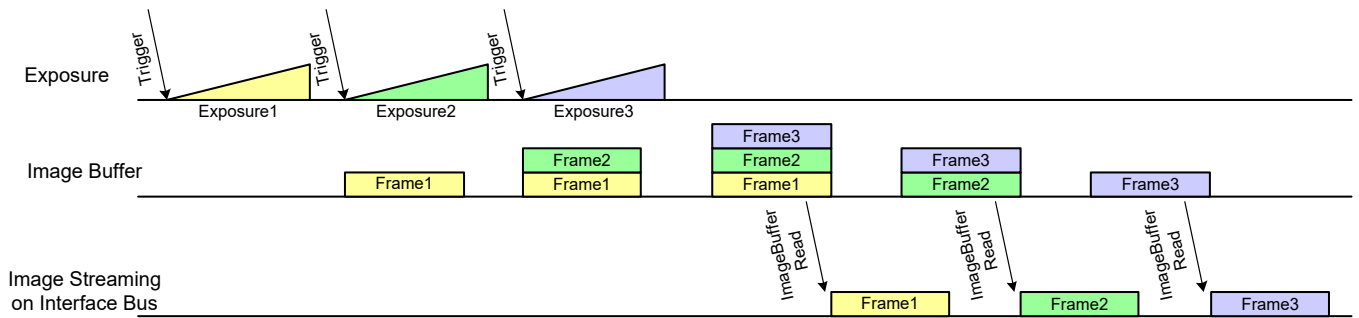Please refer to TriggerControl section as well.

Image Buffer

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
ImageBufferMode	IEnumeration	4	R/W	Sets an Image Buffer mode.
ImageBufferFrameCount	IInteger	4	R/W	Sets the number of frames to transfer in MultiFrame/ImageBuffer mode.
ImageBufferRead	ICommand	4	W	Read Image from Image Buffer

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
ImageBufferMode	Implemented	0x203060	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20306C ~0x203078	16	R	[0] : Off [1] : On
	Value	0x20307C	4	R/W	Sets an Image Buffer mode.
ImageBufferFrameCount	Implemented	0x203080	[31]	R	Returns the state whether the function is implemented.
	Min	0x203094	4	R	Returns the minimum number of frames in Image Buffer mode.
	Max	0x203098	4	R	Returns the maximum number of frames in Image Buffer mode.
	Value	0x20309C	4	R	Returns the number of frames in Image Buffer mode.

D4288264C

https://www.toshiba-teli.co.jp/en/

● Control with TeliCamSDK

Camera feature API

Control ImageBuffer using dedicated API.

API name	Description
GetCamImageBufferMode	Get current ImageBuffer mode
SetCamImageBufferMode	Set new ImageBuffer mode
GetCamImageBufferFrameCount	Get current ImageBufferFrameCount value
ExecuteCamImageBufferRead	Read Image from Image Buffer

◆ ImageBuffer

1. Select an Image Buffer mode.

   Set 'ImageBufferMode' with 'SetCamImageBufferMode' function.

2. Capture image stream.

   To start and stop capturing image is the same way as AcquisitionControl.

   Please refer to 'AcquisitionControl' section as well.

3. Read the number of frames in Image Buffer.

   Read the number of frames in Image Buffer with 'GetCamImageBufferFrameCount' function.

4. Read Image from Image Buffer.

   Start to read Image from Image Buffer with 'ExecuteCamImageBufferRead' function.

5. Receive Image from Image Buffer.

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control ImageBuffer using GenICam API.

◆ ImageBuffer

1. Select an Image Buffer mode by 'ImageBufferMode'.

   Integer value and string value of Enumeration are as follows.

Integer	String
0 (*)	Off
1	On

   * initial factory setting

2. Open and Start image stream.

   To start and stop capturing image is the same way as AcquisitionControl.

   Please refer to AcquisitionControl section as well.

3. Read the number of frames in Image Buffer by 'ImageBufferFrameCount'.

4. Read Image from Image Buffer by 'ImageBufferRead'.

5. Receive Image from Image Buffer.

6. Stop and Close image stream.

D4288264C
https://www.toshiba-teli.co.jp/en/

```
// GenICam node handle
CAM_NODE_HANDLE   hMode = NULL;
CAM_NODE_HANDLE   hCount = NULL;
CAM_NODE_HANDLE   hRead = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "ImageBufferMode", &hMode);
Nd_GetNode(s_hCam, "ImageBufferFrameCount", &hCount);
Nd_GetNode(s_hCam, "ImageBufferRead", &hRead);

// 1.Select an Image Buffer mode
Nd_SetEnumStrValue(s_hCam, hMode, "On");

// 2.Open and Start image stream.
//   2.1.Set Trigger mode
          SetCamTriggerMode(s_hCam, true);
          SetCamTriggerSource(s_hCam, CAM_TRIGGER_SOFTWARE);
//   2.2.Open Stream
          s_hStrmEvt = CreateEvent(NULL, FALSE, FALSE, NULL);
          Strm_OpenSimple(s_hCam, &s_hStrm, &s_uiImgBufSize, s_hStrmEvt);
          s_pucImgBuf = (uint8_t *)VirtualAlloc(NULL, s_uiImgBufSize,
MEM_RESERVE | MEM_COMMIT, PAGE_EXECUTE_READWRITE);
//   2.3.Stream Start
          Strm_Start(s_hStrm);
//   2.4.Execute Software Trigger
          ExecuteCamSoftwareTrigger(s_hCam);

// 3.Read the number of frames in Image Buffer by 'ImageBufferFrameCount'.
int64_t count = 0;
while(count==0)
{
          Nd_GetIntValue(s_hCam, hCount, &count);
}

// 4.Read Image from Image Buffer by 'ImageBufferRead'.
Nd_CmdExecute(s_hCam, hRead);

// 5.Receive Image from Image Buffer
          uint32_t uiSize   = s_uiImgBufSize;
          WaitForSingleObject(s_hStrmEvt, 1000);
          Strm_ReadCurrentImage(s_hStrm, s_pucImgBuf, &uiSize, NULL);

// 6.Stop and Close image stream.
//   6.1.Stream Stop
          Strm_Stop(s_hStrm);
//   6.2.Close Stream
          Strm_Close(s_hStrm);
          CloseHandle(s_hStrmEvt);
          VirtualFree(s_pucImgBuf, 0, MEM_RELEASE);
```

D4288264C

https://www.toshiba-teli.co.jp/en/

Control ImageBuffer by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **ImageBuffer**

1. Select an Image Buffer mode.

   Write to 'Value' field of 'ImageBufferMode' register to control ImageBufferMode.

2. Open and Start image stream.

   To start and stop capturing image is the same way as AcquisitionControl.

   Please refer to AcquisitionControl section as well.

3. Read the number of frames in Image Buffer.

   Read 'Value' field of 'ImageBufferFrameCount' register.

4. Read Image from Image Buffer.

   Write [10] to 'Value' field of 'AcquisitionCommand' register to execute ImageBufferRead.

   Please refer to AcquisitionCommand in AcquisitionControl section as well.

5. Receive Image from Image Buffer.

6. Stop and Close image stream.

https://www.toshiba-teli.co.jp/en/

```
// 1.Select an Image Buffer mode
uint32_t      dat = 1;
Cam_WriteReg(s_hCam, 0x20307C, 1, &dat);

// 2.Open and Start image stream.
//   2.1.Set Trigger mode
          SetCamTriggerMode(s_hCam, true);
          SetCamTriggerSource(s_hCam, CAM_TRIGGER_SOFTWARE);
//   2.2.Open Stream
          s_hStrmEvt = CreateEvent(NULL, FALSE, FALSE, NULL);
          Strm_OpenSimple(s_hCam, &s_hStrm, &s_uiImgBufSize, s_hStrmEvt);
          s_pucImgBuf = (uint8_t *)VirtualAlloc(NULL, s_uiImgBufSize,
MEM_RESERVE | MEM_COMMIT, PAGE_EXECUTE_READWRITE);
//   2.3.Stream Start
          Strm_Start(s_hStrm);
//   2.4.Execute Software Trigger
          ExecuteCamSoftwareTrigger(s_hCam);

// 3.Read the number of frames in Image Buffer.
uint32_t count = 0;
while(count==0)
{
          Cam_ReadReg(s_hCam, 0x20309C, 1, &count);
}

// 4.Read Image from Image Buffer. AcquisitionCommand = 10 : Image Buffer Read
dat = 10;
Cam_WriteReg(s_hCam, 0x20303C, 1, &dat);

// 5.Receive Image from Image Buffer
          uint32_t uiSize   = s_uiImgBufSize;
          WaitForSingleObject(s_hStrmEvt, 1000);
          Strm_ReadCurrentImage(s_hStrm, s_pucImgBuf, &uiSize, NULL);

// 6.Stop and Close image stream.
//   6.1.Stream Stop
          Strm_Stop(s_hStrm);
//   6.2.Close Stream
          Strm_Close(s_hStrm);
          CloseHandle(s_hStrmEvt);
          VirtualFree(s_pucImgBuf, 0, MEM_RELEASE);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

● **Note**

- The maximum number of frames storable depends on the image size. (maximum 64MByte.)

- The number of frames to be transferred by 'ImageBufferRead' command is determined by 'AcquisitionFrameCount'.

- Changing 'ImageBufferMode' register value is invalid during image stream data output.

D4288264C

https://www.toshiba-teli.co.jp/en/

TriggerControl

TriggerControl features are related to image acquisition using trigger.

This camera series provides two kinds of exposure synchronization.

1. Normal Shutter mode   : Free run operation (internal synchronization)
2. Random Trigger Shutter mode   : Synchronized with external trigger input

In Random Trigger Shutter mode, two kinds of trigger input are available.

1. Trigger signal via the I/O connector (HardwareTrigger)
2. Trigger command via software command (SoftwareTrigger)

The following table shows the combination of operation mode of this camera series.

Operation Mode

Trigger Mode	Synchronization	Exposure Control
Normal Shutter mode	Free run	'ExposureTime' register control
Random Trigger Shutter mode	HardwareTrigger	-Edge mode:TriggerSequence0 -Bulk mode:TriggerSequence6 'ExposureTime' register control
		-Level mode:TriggerSequence1 Trigger pulse width control
	SoftwareTrigger	-Edge mode:TriggerSequence0 -Bulk mode:TriggerSequence6 'ExposureTime' register control

* The camera operation not mentioned above is not supported.

D4288264C
https://www.toshiba-teli.co.jp/en/

- Edge mode (TriggerSequence0)

The exposure time is determined by Exposure Time setting.
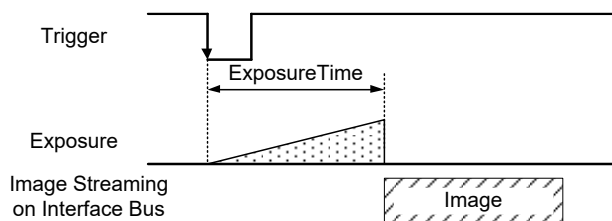
- Level mode (TriggerSequence1)

The exposure time is determined by the pulse width of the trigger signal.

This feature is not supported when the ShortExposureMode is enable (ON).
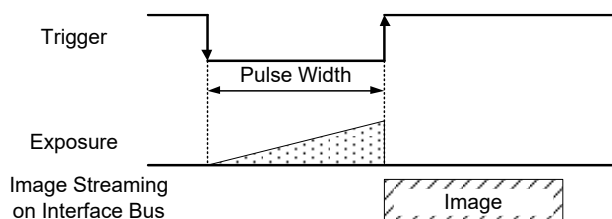
- Bulk mode (TriggerSequence6)

Camera exposes and transfers multiple frames by a single trigger.
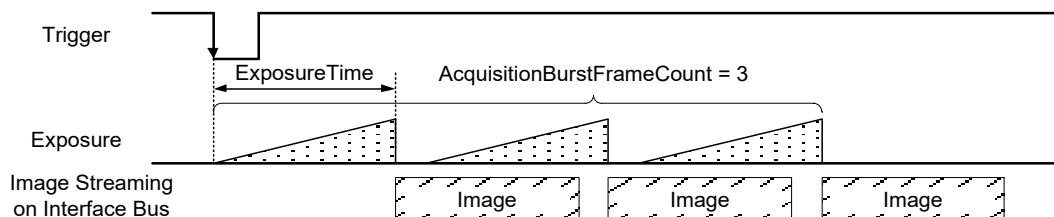
Operation point of HardwareTrigger is at the edge of trigger signal, and active edge polarity is able to change by register setting. And you can add delay time from trigger edge to exposure start by register setting.

Trigger Delay
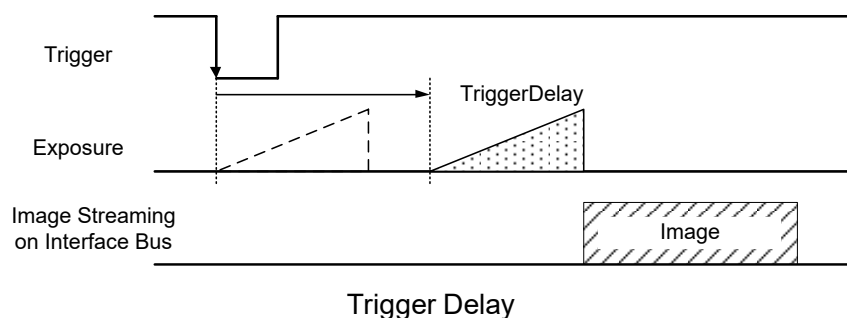
For details of Random Trigger Shutter operation, please refer to 'Timing' section in 'Specification'.

D4288264C
https://www.toshiba-teli.co.jp/en/

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
TriggerMode	IEnumeration	4	R/W	Selects a trigger mode.
TriggerSoftware	ICommand	4	W	Executes software trigger.
TriggerSource	IEnumeration	4	R/W	Selects a trigger source of random trigger shutter.
TriggerActivation	IEnumeration	4	R/W	Selects a trigger activation of hardware trigger.
TriggerDelay	IFloat	4	R/W	Sets a trigger delay.
TriggerSequence	IEnumeration	4	R/W	Selects a trigger sequence of random trigger shutter.
TriggerAdditionalParameter	IInteger	4	R/W	Sets the number of frames to exposure in Bulk mode.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
TriggerMode	Implemented	0x207020	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20702C ~0x207038	16	R	[0] : OFF (Normal shutter) [1] : ON (Random trigger shutter)
	Value	0x20703C	4	R/W	Selects a trigger mode.
TriggerSequence	Implemented	0x207040	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20704C ~0x207058	16	R	[0] : TriggerSequence0 (Edge mode) [1] : TriggerSequence1 (Level mode) [6] : TriggerSequence6 (Bulk mode)
	Value	0x20705C	4	R/W	Selects an exposure mode.
TriggerSource	Implemented	0x207060	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20706C ~0x207078	16	R	[0] : Line0 (Hardware Trigger) [1] : Line1 (Hardware Trigger) [64] : Software (SoftwareTrigger)
	Value	0x20707C	4	R/W	Selects a trigger source.
TriggerAdditionalParameter	Implemented	0x207080	[31]	R	Returns the state whether the function is implemented.
	Min	0x207094	4	R	Returns the minimum value of the number of images in Bulk mode.
	Max	0x207098	4	R	Returns the maximum value of the number of images in Bulk mode.
	Value	0x20709C	4	R/W	Sets the number of frames to exposure in Bulk mode.

D4288264C

https://www.toshiba-teli.co.jp/en/

Register	Field	Address	Length Byte / [bit]	Access	Description
TriggerDelay	Implemented	0x2070A0	[31]	R	Returns the state whether the function is implemented.
	Mult	0x2070AC	4	R	AbsoluteValue[sec] = Raw * (Mult / Div)
	Div	0x2070B0	4	R	
	Min	0x2070B4	4	R	Returns the minimum value of trigger delay.
	Max	0x2070B8	4	R	Returns the maximum value of trigger delay.
	Value	0x2070BC	4	R/W	Sets the trigger delay.(Raw value)
SoftwareTrigger	Implemented	0x207040	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20702C ~0x207038	16	R	[0] : Inactive [8] : Impulse
	Value	0x20705C	4	R/W	Executes software trigger.

● Control with TeliCamSDK

Camera feature API

Control Trigger features using dedicated API.

API name	Description
GetCamTriggerMode	Get current TriggerMode
SetCamTriggerMode	Set new TriggerMode
GetCamTriggerSequence	Get current TriggerSequence
SetCamTriggerSequence	Set new TriggerSequence
GetCamTriggerSource	Get current TriggerSource
SetCamTriggerSource	Set new TriggerSource
GetCamTriggerAdditionalParameterMinMax	Get minimum and maximum value
GetCamTriggerAdditionalParameter	Get current TriggerAdditionalParameter value
SetCamTriggerAdditionalParameter	Set new TriggerAdditionalParameter value
GetCamTriggerDelayMinMax	Get minimum and maximum value
GetCamTriggerDelay	Get current TriggerDelay value
SetCamTriggerDelay	Set new TriggerDelay value
ExecuteCamSoftwareTrigger	Execute Software Trigger

1. Select a trigger mode.

    Set 'TriggerMode' with 'SetCamTriggerMode' function.

2. Select a trigger sequence of random trigger shutter.

    Set 'TriggerSequence' with 'SetCamTriggerSequence' function.

3. Select a trigger source of random trigger shutter.

    Set 'TriggerSource' with 'SetCamTriggerSource' function.

4. Select a trigger activation of hardware trigger.

5. Set the number of frames to exposure in Bulk mode.

    Set 'TriggerAdditionalParameter' with 'SetCamTriggerAdditionalParameter' function.

6. Set a trigger delay.

    Set 'TriggerDelay' with 'SetCamTriggerDelay' function.

7. Capture image stream.

    To start and stop capturing image is the same way as AcquisitionControl.

    Please refer to AcquisitionControl section as well.

    Please refer to [Camera streaming functions] in [TeliCamAPI Library manual]

    and [GrabStreamSimple] sample code in your TeliCamSDK install folder.

8. Execute software trigger.

    Execute Software Trigger in SoftwareTrigger mode with 'ExecuteCamSoftwareTrigger' function.

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control Trigger features using GenICam API.

1. Select a trigger mode by 'TriggerMode'.

   Integer value and string value of Enumeration are as follows.

Integer	String
0 (*)	Off
1	On

   * initial factory setting

2. Select a trigger sequence of random trigger shutter by 'TriggerSequence'.

   Integer value and string value of Enumeration are as follows.

Integer	String
0 (*)	TriggerSequence0
1	TriggerSequence1
6	TriggerSequence6

   * initial factory setting

3. Select a trigger source of random trigger shutter by 'TriggerSource'.

   Integer value and string value of Enumeration are as follows.

Integer	String
0 (*)	Line0
1	Line1
64	Software

   * initial factory setting

4. Select a trigger activation of hardware trigger by 'TriggerActivation'.

5. Set the number of frames to exposure in Bulk mode by 'TriggerAdditionalParameter'.

6. Set a trigger delay by 'TriggerDelay'.

7. Capture image stream.

   To start and stop capturing image is the same way as AcquisitionControl.

   Please refer to AcquisitionControl section as well.

   Please refer to [Camera streaming functions] in [TeliCamAPI Library manual]

   and [GrabStreamSimple] sample code in your TeliCamSDK install folder.

8. Execute software trigger in SoftwareTrigger mode by 'TriggerSoftware'.

9. Receive Image

10. Stop and Close image stream.

```
// GenICam node handle
CAM_NODE_HANDLE   hMode = NULL;
CAM_NODE_HANDLE   hSequence = NULL;
CAM_NODE_HANDLE   hSource = NULL;
CAM_NODE_HANDLE   hAdditionalParameter = NULL;
CAM_NODE_HANDLE   hDelay = NULL;
CAM_NODE_HANDLE   hTriggerSoftware = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "TriggerMode", &hMode);
Nd_GetNode(s_hCam, "TriggerSequence", &hSequence);
Nd_GetNode(s_hCam, "TriggerSource", &hSource);
Nd_GetNode(s_hCam, "TriggerAdditionalParameter", &hAdditionalParameter);
Nd_GetNode(s_hCam, "TriggerDelay", &hDelay);
Nd_GetNode(s_hCam, "TriggerSoftware", &hTriggerSoftware);

// 1. Select a trigger mode by 'TriggerMode'.
Nd_SetEnumStrValue(s_hCam, hMode, "On");

// 2. Select a trigger sequence of random trigger shutter by 'TriggerSequence'.
Nd_SetEnumStrValue(s_hCam, hSequence, "TriggerSequence6");        // Bulk mode

// 3. Select a trigger source of random trigger shutter by 'TriggerSource'.
Nd_SetEnumStrValue(s_hCam, hSource, "Software");                 // Software

// 4.Select a trigger activation of hardware trigger by 'LineInverterAll'.
//    -- not applicable for Software Trigger

// 5.Set the number of frames to exposure in Bulk mode by 'TriggerAdditionalParameter'.
Nd_SetIntValue(s_hCam, hAdditionalParameter, 3);                // 3 frames

// 6.Set a trigger delay by 'TriggerDelay'.
Nd_SetFloatValue(s_hCam, hDelay, 1000.0);                       // 1ms

// 7.Open and Start image stream.
//   7.1.Open Stream
          s_hStrmEvt = CreateEvent(NULL, FALSE, FALSE, NULL);
          Strm_OpenSimple(s_hCam, &s_hStrm, &s_uiImgBufSize, s_hStrmEvt);
          s_pucImgBuf = (uint8_t *)VirtualAlloc(NULL, s_uiImgBufSize,
MEM_RESERVE | MEM_COMMIT, PAGE_EXECUTE_READWRITE);
//   7.2.Stream Start
          Strm_Start(s_hStrm);

// 8.Execute software trigger in SoftwareTrigger mode by 'TriggerSoftware'.
Nd_CmdExecute(s_hCam,hTriggerSoftware);

// 9.Receive Image
uint32_t uiSize   = s_uiImgBufSize;
CAM_IMAGE_INFO sImageInfo;
for(int i=0; i<3; i++)
{
          WaitForSingleObject(s_hStrmEvt, 1000);
          Strm_ReadCurrentImage(s_hStrm, s_pucImgBuf, &uiSize, &sImageInfo);
}
// 10.Stop and Close image stream.
//   10.1.Stream Stop
          Strm_Stop(s_hStrm);
//   10.2.Close Stream
          Strm_Close(s_hStrm);
          CloseHandle(s_hStrmEvt);
          VirtualFree(s_pucImgBuf, 0, MEM_RELEASE);
```

D4288264C

https://www.toshiba-teli.co.jp/en/

Please refer to [INode functions], [IEnumeration node functions][ICommand node functions][IInteger node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control Trigger features by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

1. Select a trigger mode.

   Write to 'Value' field of 'TriggerMode' register to control TriggerMode.

2. Select a trigger sequence of random trigger shutter.

   Write to 'Value' field of 'TriggerSequence' register to control TriggerSequence.

3. Select a trigger source of random trigger shutter.

   Write to 'Value' field of 'TriggerSource' register to control TriggerSource.

4. Select a trigger activation of hardware trigger.

5. Set the number of frames to exposure in Bulk mode.

   Write to 'Value' field of 'TriggerAdditionalParameter' register to control TriggerAdditionalParameter.

6. Set a trigger delay.

   Write to 'Value' field of 'TriggerDelay' register to control TriggerDelay.

7. Open and Start image stream.

   To start and stop capturing image is the same way as AcquisitionControl.

   Please refer to AcquisitionControl section as well.

   Please refer to [Camera streaming functions] in [TeliCamAPI Library manual]

   and [GrabStreamSimple] sample code in your TeliCamSDK install folder.

8. Execute software trigger.

   Write [8] to 'Value' field of 'SoftwareTrigger' register to control SoftwareTrigger.

9. Receive Image

10. Stop and Close image stream.

```
uint32_t       dat;

// 1. Select a trigger mode by 'TriggerMode'.
dat = 1;
Cam_WriteReg(s_hCam, 0x20703C, 1, &dat); // TriggerMode = On

// 2. Select a trigger sequence of random trigger shutter by 'TriggerSequence'.
dat = 6;
Cam_WriteReg(s_hCam, 0x20705C, 1, &dat); // TriggerSequence6 (Bulk mode)

// 3. Select a trigger source of random trigger shutter by 'TriggerSource'.
dat = 64;
Cam_WriteReg(s_hCam, 0x20707C, 1, &dat); // Software

// 4.Select a trigger activation of hardware trigger by 'LineInverterAll'.
//    -- not applicable for Software Trigger

// 5.Set the number of frames to exposure in Bulk mode by
'TriggerAdditionalParameter'.
dat = 3;
Cam_WriteReg(s_hCam, 0x20709C, 1, &dat); // 3 frames

// 6.Set a trigger delay by 'TriggerDelay'.
// TriggerDelay = 1000.0us (Raw value = 60000)
dat = 60000;
Cam_WriteReg(s_hCam, 0x2070BC, 1, &dat); // 1ms

// 7.Open and Start image stream.
//    7.1.Open Stream
          s_hStrmEvt = CreateEvent(NULL, FALSE, FALSE, NULL);
          Strm_OpenSimple(s_hCam, &s_hStrm, &s_uiImgBufSize, s_hStrmEvt);
          s_pucImgBuf = (uint8_t *)VirtualAlloc(NULL, s_uiImgBufSize,
MEM_RESERVE | MEM_COMMIT, PAGE_EXECUTE_READWRITE);
//    7.2.Stream Start
          Strm_Start(s_hStrm);

// 8.Execute software trigger in SoftwareTrigger mode by 'SoftwareTrigger'.
dat = 8;
Cam_WriteReg(s_hCam, 0x2070DC, 1, &dat); // SoftwareTrigger

// 9.Receive Image
uint32_t uiSize   = s_uiImgBufSize;
CAM_IMAGE_INFO sImageInfo;
for(int i=0; i<3; i++)
{
          WaitForSingleObject(s_hStrmEvt, 1000);
          Strm_ReadCurrentImage(s_hStrm, s_pucImgBuf, &uiSize, &sImageInfo);
}

// 10.Stop and Close image stream.
//    10.1.Stream Stop
          Strm_Stop(s_hStrm);
//    10.2.Close Stream
          Strm_Close(s_hStrm);
          CloseHandle(s_hStrmEvt);
          VirtualFree(s_pucImgBuf, 0, MEM_RELEASE);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ **Minimum / Maximum Value**

TriggerAdditional Parameter	Raw Value = Absolute (Float) Value
Minimum	0
Maximum	255
Initial Value	0

TriggerDelay		BG302LMG / BG505LMG
Minimum	Raw Value	0
	Absolute Value	0.00 [us]
Maximum	Raw Value	50000000
	Absolute Value	2000000.00 [us]
Initial Value	Raw Value	0
	Absolute Value	0.00 [us]
Formula		Absolute Value [us] = Raw Value / 25

● **Note**

- In SoftwareTrigger operation, the delay time from 'TriggerSoftware' to exposure is not guaranteed.
- BG series have two hardware trigger source.

  Please refer to 'I/O Specification' section as well.

TriggerSource	Description
Line0 (*)	I/O connector : 2 pin, Photo coupler 3.3 to 24.0V
Line1	I/O connector : 3 pin, High level 2.0 to 3.3V

* initial factory setting

- TriggerDelay is applied to both of HardwareTrigger and SoftwareTrigger.
- TriggerAdditionalParameter is only effective in Bulk mode.

ExposureTime

ExposureTime controls the duration where the image sensor is exposed to light.

This camera series provides three kinds of exposure time control mode.

Firmware Ver.3.5.0 or later models have ShortExposureMode which can set high-speed exposure time at the time of MANUAL setting. AutoExposure is not supported when the ShortExposureMode is enable (ON).

- NoSpecify : The exposure time is determined by 'AcquisitionFrameRate' register setting value
- Manual : The exposure time is determined by 'ExposureTime' register setting value.
- Auto : The exposure time is adjusted automatically.

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
ExposureTime	IFloat	4	R/W	Sets an exposure time.
ExposureAuto	IEnumeration	4	R/W	Adjusts a exposure time automatically.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
ExposureTime	Implemented	0x204020	[31]	R	Returns the state whether the function is implemented.
	Control	0x204028	4	R/W	[0]: NoSpecify [1]: Manual [2]: Auto
	Mult	0x20402C	4	R	Absolute Value [sec] = Raw * (Mult / Div)
	Div	0x204030	4	R	
	Min	0x204034	4	R	Returns the minimum value of ExposureTime Value register.
	Max	0x204038	4	R	Returns the maximum value of ExposureTime Value register.
	Value	0x20403C	4	R/W	Sets ExposureTime value. (Raw value)
ShortExposureMode	Implemented	0x204380	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20438C	4	R	[0] : Off [1] : On
	Value	0x20439C	4	R/W	Only firmware Ver.3.5.0 or later. Sets the activation of ShortExposureMode.

D4288264C

● Control with TeliCamSDK

Camera feature API

Control ExposureTime using dedicated API.

API name	Description
GetCamExposureTimeMinMax	Get minimum and maximum value
GetCamExposureTime	Get current ExposureTime value
SetCamExposureTime	Set new ExposureTime value
GetCamExposureTimeControl	Get current ExposureTime mode
SetCamExposureTimeControl	Set new ExposureTime mode
GetCamShortExposureMode	Get current ShortExposureMode
SetCamShortExposureMode	Set new ShortExposureMode

ExposureTimeControl parameter	Description
CAM_EXPOSURE_TIME_CONTROL_NO_SPECIFY	NoSpecify
CAM_EXPOSURE_TIME_CONTROL_MANUAL	Manual
CAM_EXPOSURE_TIME_CONTROL_AUTO	Auto

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control ExposureTime using GenICam API.

◆ **ExposureTime**

1. Sets an exposure time by 'ExposureTime'

```
// GenICam node handle
CAM_NODE_HANDLE = NULL;
CAM_NODE_HANDLE = NULL;
CAM_NODE_HANDLE = NULL;

// GenICam node handle
CAM_NODE_HANDLE    hNode = NULL;

// ExposureTime = 1000us
float64_t      dExposureTime = 1000.0;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "ExposureTime", &hNode);

// Set ExposureTime Value
Nd_SetFloatValue(s_hCam, hNode, dExposureTime);
```

◆ ExposureAuto

1. Selects an exposure time mode by 'ExposureAuto'

   Integer value and string value of Enumeration are as follows.

Integer	String	Description
0	DeviceSpecific	Operates NoSpecify mode.
1 (*)	Off	Operates Manual mode.
2	Continuous	Operates Auto mode.

   * initial factory setting

```
// GenICam node handle
CAM_NODE_HANDLE   hNode = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "ExposureAuto", &hNode);

// Set ExposureAuto = "Continuous "
Nd_SetEnumStrValue(s_hCam, hNode, "Continuous ");
```

Please refer to [INode functions], [IFloat node functions] and [IEnumeration node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control ExposureTime by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ ExposureTime

Write to 'Value' field of 'ExposureTime' register.

```
// ExposureTime = 1000us (Raw value = 60000)
uint32_t      uiExposureTimeRaw = 60000;

// Set ExposureTime Value
Cam_WriteReg(s_hCam, 0x20403C, 1, &uiExposureTimeRaw);
```

◆ ExposureAuto

Write to 'Control' field of 'ExposureTime' register to control ExposureAuto.

```
// ExposureAuto = "Continuous "
uint32_t      uiExposureAuto = 2;

// Set ExposureAuto = "Continuous "
Cam_WriteReg(s_hCam, 0x204028, 1, &uiExposureAuto);
```

◆ **ShortExposureMode**

Write to 'Value' field of 'ShortExposureMode' register to control ShortExposureMode.

```
// Set ShortExposureMode ON
uint32_t    dat = 1;
Cam_WriteReg(s_hCam, 0x20439C, 1, &dat);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ **Minimum / Maximum Value**

ShortExposureMode=OFF

ExposureTime		BG040M	BG160M
Minimum	Raw Value	Ver.3.3.9 or earlier：750 Ver.3.5.0 or later：500	
	Absolute Value	Ver.3.3.9 or earlier：30.00 [us] Ver.3.5.0 or later：20.00 [us]	
Maximum	Manual	Raw Value	400000000	
		Absolute Value	16000000.00 [us]	
	Auto	Raw Value	25000000	
		Absolute Value	1000000.00 [us]	
Initial Value	Raw Value	75000	325000
	Absolute Value	3000.00 [us]	13000.00 [us]
Formula		Absolute Value [us] = Raw Value / 25	

ShortExposureMode=ON

ExposureTime		BG040M	BG160M
Minimum	Raw Value	20	
	Absolute Value	1.08 [us]	
Maximum	Manual	Raw Value	247	
		Absolute Value	13.31 [us]	
	Auto	Raw Value	Not supported	
		Absolute Value		
Initial Value	Raw Value	247	
	Absolute Value	13.31 [us]	
Formula		Absolute Value [s] = Raw Value * 53 / 983850009	

D4288264C

https://www.toshiba-teli.co.jp/en/

> **Notes on ShortExposureMode setting:**
>
> - If you use ShortExposureMode, the image quality may deteriorate. Also, the actual exposure time may vary depending on the individual differences and the operating environment (such as the operating temperature). When using the ShortExposureMode, we ask you to have final image quality checked with your environment.
> - The brightness of the upper part of the screen may be different from that of the lower part. Note that this is a characteristic of a CMOS image sensor and is not a fault.
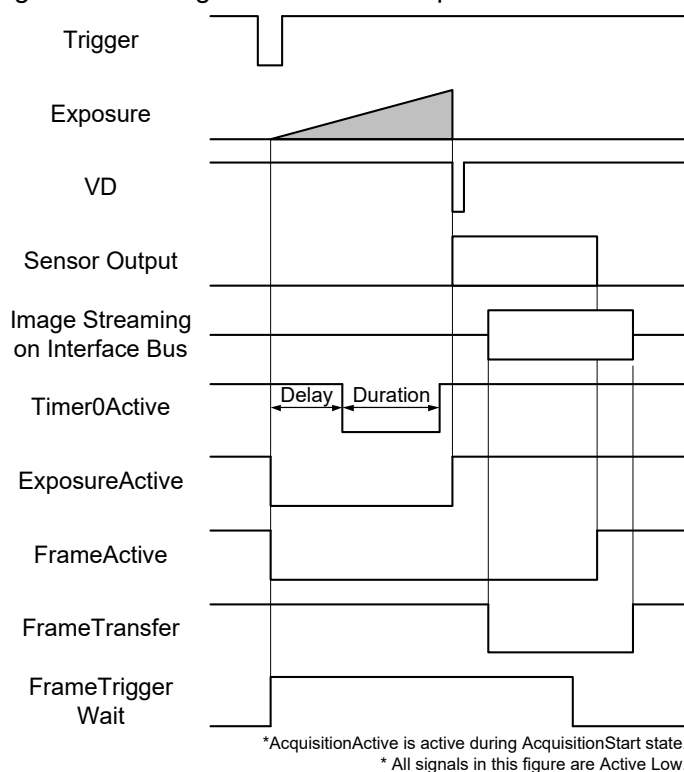
● Note

- In ExposureAuto mode, current exposure time will be reflected to ExposureTime value register.
- Changing 'ShortExposureMode' register value is invalid during image stream data output.

D4288264C
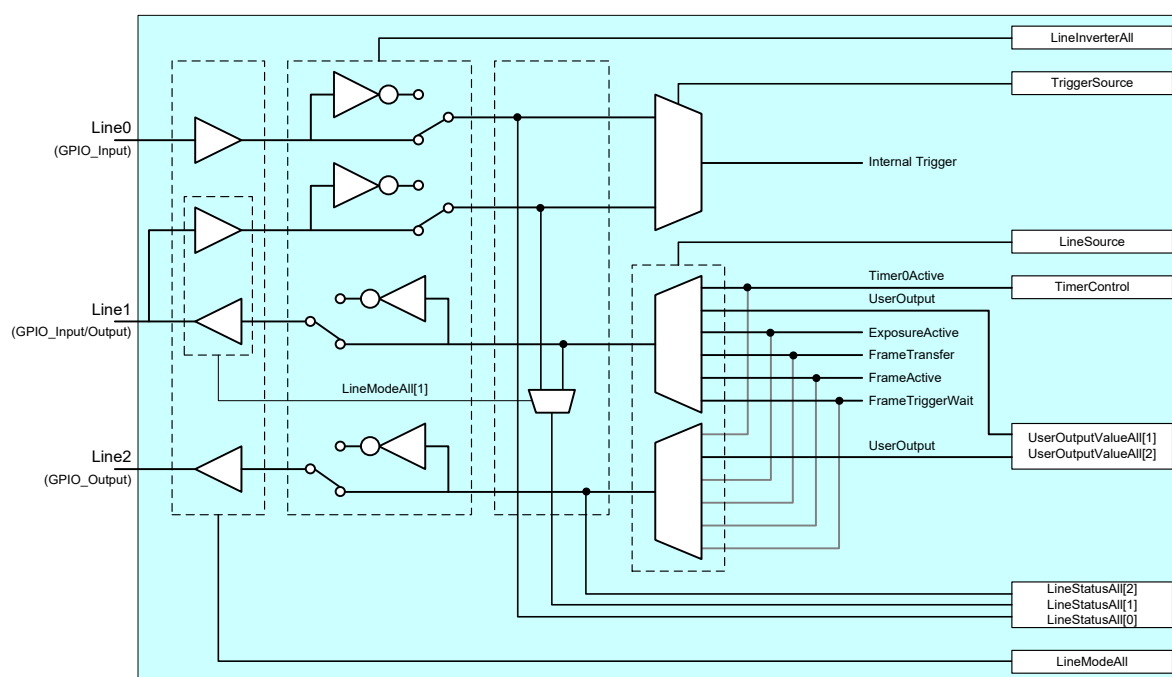
https://www.toshiba-teli.co.jp/en/

DigitalIOControl

This section describes DigitalIOControl features.

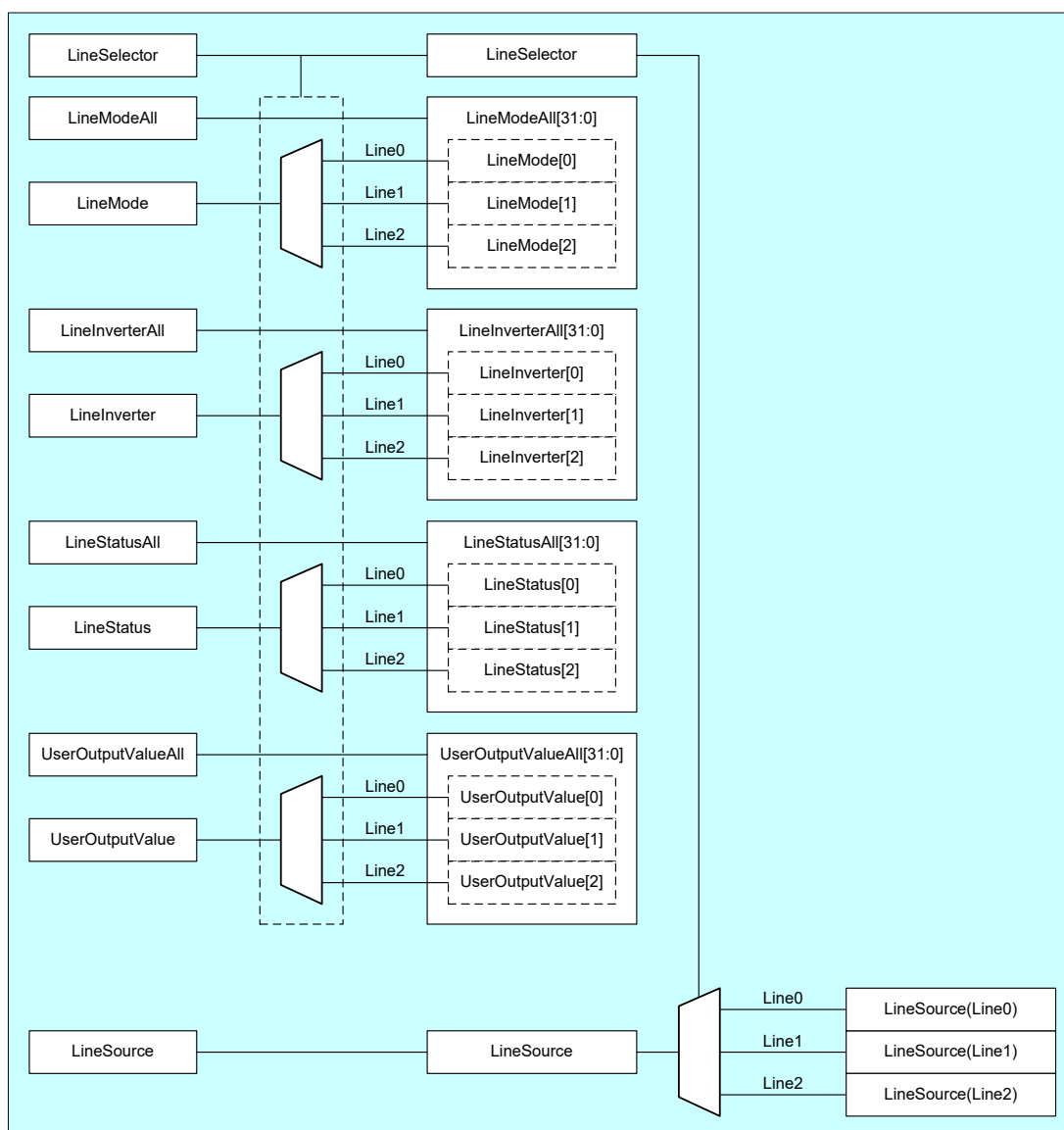This camera provides GPIO output selected by the register setting. And the polarity of the signal is able to switch by the register setting. The following chart shows the specifications of the selectable signals.

*AcquisitionActive is active during AcquisitionStart state.
* All signals in this figure are Active Low.

Selectable signals

GPIO internal circuit diagram

D4288264C
https://www.toshiba-teli.co.jp/en/

Details of each signal

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
LineMode	IEnumeration	4	R/W	Selects the Input / Output of each Line selected by LineSelector.
LineModeAll	IInteger	4	R/W	Selects the Input / Output of each Line.
LineInverter	IBoolean	4	R/W	Selects the polarity of each Line signal selected by LineSelector.
LineInverterAll	IInteger	4	R/W	Selects the polarity of each Line signal.
LineStatus	IBoolean	4	R	Returns the status of each Line signal selected by LineSelector.
LineStatusAll	IInteger	4	R	Returns the status of each Line signal.
UserOutputValue	IBoolean	4	R/W	Sets the user output value of each line selected by LineSelector.
UserOutputValueAll	IInteger	4	R/W	Sets the user output value of each Line.
LineSelector	IEnumeration	4	R/W	Selects the Line of I/O connector.
LineSource	IEnumeration	4	R/W	Selects the source of the output signal.

D4288264C

https://www.toshiba-teli.co.jp/en/

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
LineModeAll	Implemented	0x209020	[31]	R	Returns the state whether the function is implemented.
	BitWritable	0x20902C	4	R	[0]: Line0 [1]: Line1 [2]: Line2
	Value	0x209030	4	R/W	Selects the direction of each line signal.
LineInverterAll	Implemented	0x209040	[31]	R	Returns the state whether the function is implemented.
	Value	0x209050	4	R/W	Selects the inversion of I/O lines. Reflected both input and output buffers.
LineStatusAll	Implemented	0x209060	[31]	R	Returns the state whether the function is implemented.
	Value	0x209070	4	R	Returns the current status of all I/O lines.
UserOutputValueAll	Implemented	0x209080	[31]	R	Returns the state whether the function is implemented.
	Value	0x209090	4	R/W	Selects the output value.
LineSelector	Implemented	0x2090A0	[31]	R	Returns the state whether the function is implemented.
	Value	0x2090BC	4	R/W	Selects the I/O line to output LineSource.
LineSource	Implemented	0x2090C0	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x2090CC	4	R	[0] : Off [32] : UserOutput [64] : Timer0Active [99] : AcquisitionActive [106] : FrameTriggerWait [107] : FrameActive [115] : FrameTransferActive [123] : ExposureActive
	Value	0x2090DC	4	R/W	Selects the source signal.

D4288264C

https://www.toshiba-teli.co.jp/en/

● **Control with TeliCamSDK**

Camera feature API

Control DigitalIOControl using dedicated API.

| API name	Description
GetCamLineModeAll	Get current LineMode
GetCamLineInverterAll	Get current LineInverter
SetCamLineInverterAll	Set new LineInverter
GetCamLineStatusAll	Get current LineStatus
GetCamUserOutputValueAll	Get current UserOutput
SetCamUserOutputValueAll	Set new UserOutput
GetCamLineSource	Get current LineSource
SetCamLineSource	Set new LineSource

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

D4288264C

https://www.toshiba-teli.co.jp/en/

GenICam function API

Control DigitalIOControl using GenICam API.

◆ **LineModeAll**
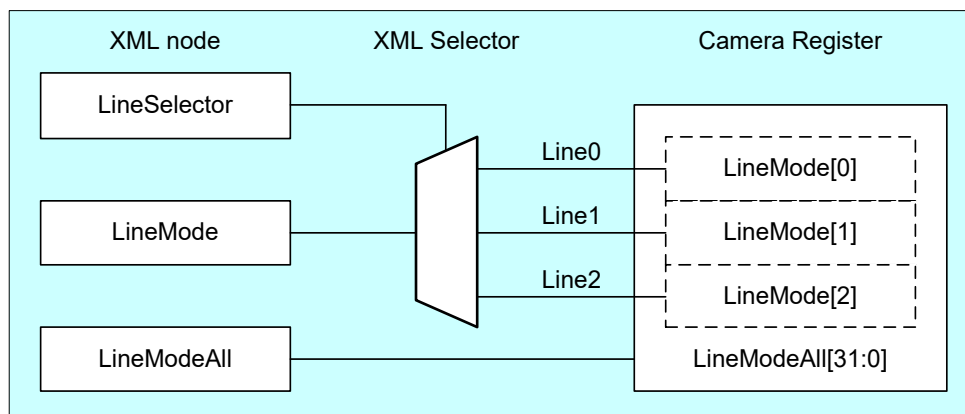
Selects the Input / Output of each Line by 'LineModeAll'.

Each bit of integer value corresponds to each Line (bit0=Line0(not changeable), bit1=Line1, bit2=Line2(not changeable)).

bit value	I/O Direction
0	Input
1	Output

LineModeAll Value	bit value [Line2] [Line1] [Line0]	I/O Direction [Line2] [Line1] [Line0]
0		
1	not used	
2		
3		
4 (*)	[1] [0] [0]	[out] [in] [in]
5	not used	
6	[1] [1] [0]	[out] [out] [in]
7	not used	

* initial factory setting

D4288264C

https://www.toshiba-teli.co.jp/en/
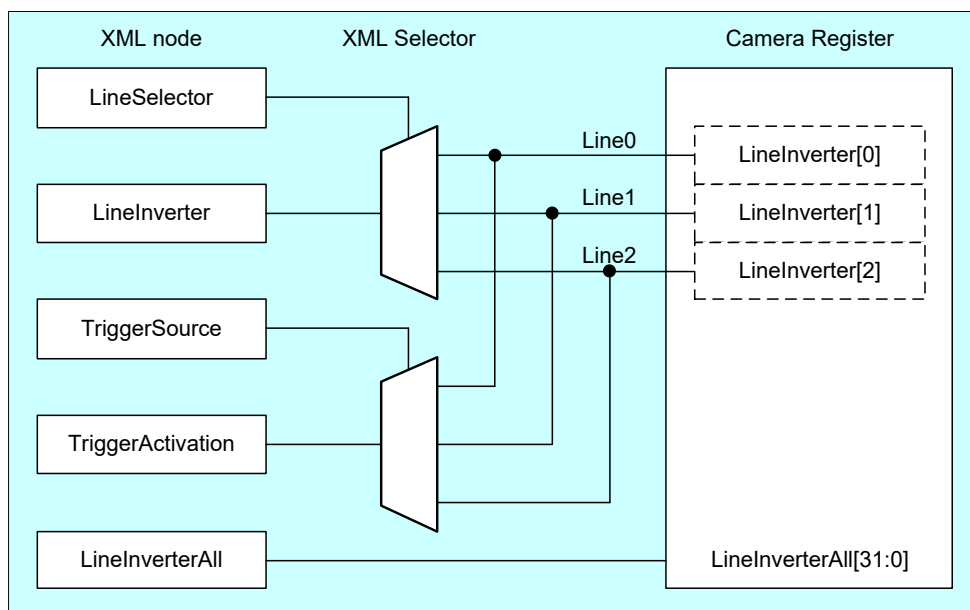
◆ LineInverterAll

Selects the polarity of each Line signal by 'LineInverterAll'.

Each bit of integer value corresponds to each Line (bit0=Line0, bit1=Line1, bit2=Line2).

bit value	Inverter
0	False (not inverted)
1	True (inverted)

LineInverterAll Value	bit value [Line2] [Line1] [Line0]	Inverter [Line2] [Line1] [Line0]
0 (*)	[0] [0] [0]	[off] [off] [off]
1	[0] [0] [1]	[off] [off] [on]
2	[0] [1] [0]	[off] [on] [off]
3	[0] [1] [1]	[off] [on] [on]
4	[1] [0] [0]	[on] [off] [off]
5	[1] [0] [1]	[out [off] [on]
6	[1] [1] [0]	[on] [on] [off]
7	[1] [1] [1]	[on] [on] [on]

* initial factory setting

D4288264C

https://www.toshiba-teli.co.jp/en/
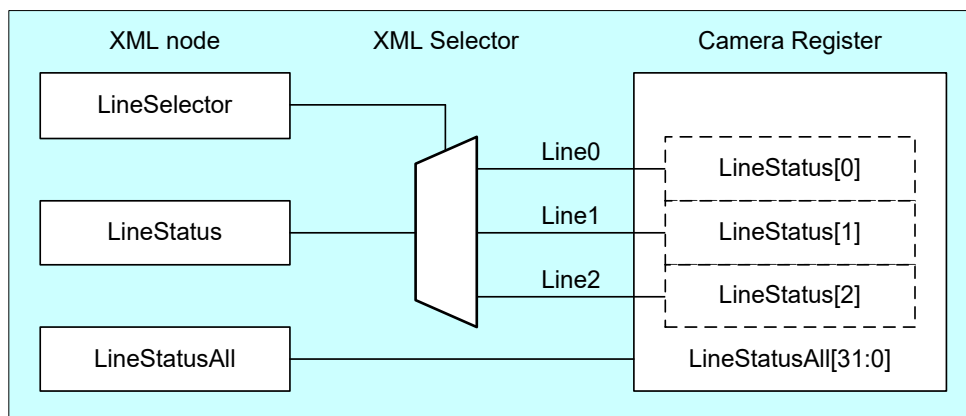
◆ LineStatusAll

Gets the status of each Line signal by 'LineStatusAll'

Each bit of integer value corresponds to each Line (bit0=Line0, bit1=Line1, bit2=Line2).

bit value	Line Status
0	False(Low)
1	True(High)

LineStatusAll Value	bit value [Line2] [Line1] [Line0]	Line Status [Line2] [Line1] [Line0]
0	[0] [0] [0]	[low] [low] [low]
1	[0] [0] [1]	[low] [low] [high]
2	[0] [1] [0]	[low] [high] [low]
3	[0] [1] [1]	[low] [high] [high]
4	[1] [0] [0]	[high] [low] [low]
5	[1] [0] [1]	[high] [low] [high]
6	[1] [1] [0]	[high] [high] [low]
7	[1] [1] [1]	[high] [high] [high]

D4288264C

https://www.toshiba-teli.co.jp/en/

◆ **UserOutputValueAll**
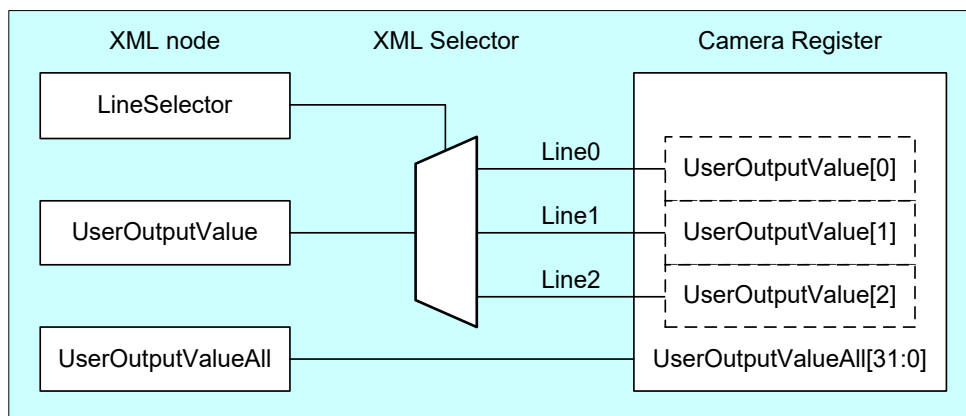
Sets the user output value by 'UserOutputValueAll'.

Each bit of integer value corresponds to each Line (bit0=Line0(not used), bit1=Line1, bit2=Line2).

bit value	Output
0	Low
1	High

UserOutputValueAll Value	bit value [Line2] [Line1]	Output [Line2] [Line1]
0 (*)	[0] [0]	[low] [low]
1		
2	[0] [1]	[low] [high]
3		
4	[1] [0]	[high] [low]
5		
6	[1] [1]	[high] [high]
7		

◆ **LineSelector**
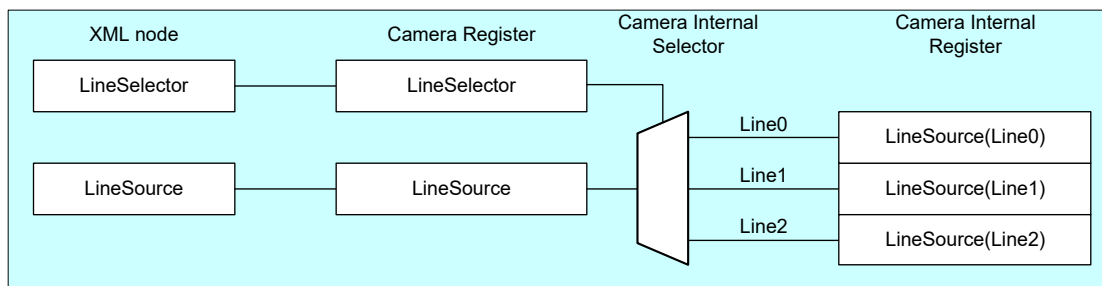
Selects the I/O line to output by 'LineSelector'.

Integer value and string value of Enumeration are as follows.

Integer	String
1	Line1
2	Line2

◆ LineSource

Selects the source of the output signal.

Integer value and string value of Enumeration are as follows.

Integer	String	Description
0	Off	Off
32	UserOutput	Outputs the value set in 'UserOutputValueAll'.
64	Timer0Active	This signal can be used as strobe control signal. The delay time and pulse width of this signal are configurable.
99	AcquisitionActive	Indicates AcquisitionStart state of camera.
106	FrameTriggerWait	Indicates that camera is ready to accept trigger signal. (both hardware and software)
107	FrameActive	Period from exposure start to sensor read-out completion.
115	FrameTransferActive	Period of transferring image streaming data on interface bus.
123	ExposureActive	Period from exposure start to exposure end.

```
// GenICam node handle
CAM_NODE_HANDLE   hMode = NULL;
CAM_NODE_HANDLE   hSelector = NULL;
CAM_NODE_HANDLE   hSource = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "LineModeAll", &hMode);
Nd_GetNode(s_hCam, "LineSelector", &hSelector);
Nd_GetNode(s_hCam, "LineSource", &hSource);

// Line2/Line1 = output, Line0 = input
int64_t     Mode = 6;
Nd_SetIntValue(s_hCam, hMode, Mode);

// Line1 = ExposureActive
Nd_SetEnumStrValue(s_hCam, hSelector, "Line1");
Nd_SetEnumStrValue(s_hCam, hSource, "ExposureActive");

// Line2 = FrameTransferActive
Nd_SetEnumStrValue(s_hCam, hSelector, "Line2");
Nd_SetEnumStrValue(s_hCam, hSource, "FrameTransferActive");
```

Please refer to [INode functions], [IInteger node functions] and [IEnumeration node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control DigitalIOControl by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **LineModeAll**

Write to 'Value' field of 'LineModeAll' register.

◆ **LineInverterAll**

Write to 'Value' field of 'LineInverterAll' register.

◆ **LineStatusAll**

Read 'Value' field of 'LineStatusAll' register.

◆ **UserOutputValueAll**

Write to 'Value' field of 'UserOutputAll' register.

◆ **LineSelector**

Write to 'Value' field of 'LineSelector' register.

◆ **LineSource**

Write to 'Value' field of 'LineSource' register.

```
// Line2/Line1 = output, Line0 = input
uint32_t     uiMode = 6;
Cam_WriteReg(s_hCam, 0x209030, 1, &uiMode);

// Set Value
uint32_t     uiSelector;
uint32_t     uiSource;
uiSelector = 1;         // Line1
uiSource = 123;         // ExposureActive
Cam_WriteReg(s_hCam, 0x2090BC, 1, &uiSelector);
Cam_WriteReg(s_hCam, 0x2090DC, 1, &uiSource);

uiSelector = 2;         // Line2
uiSource = 115;         // FrameTransferActive
Cam_WriteReg(s_hCam, 0x2090BC, 1, &uiSelector);
Cam_WriteReg(s_hCam, 0x2090DC, 1, &uiSource);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

● **Note**

- Line0 is dedicated input.

- Line1 is bi-directional. The default direction is input.

- Line2 is dedicated output.

- For [Timer0Active] signal, please refer to [TimerControl] Section for more detail.

D4288264C

https://www.toshiba-teli.co.jp/en/
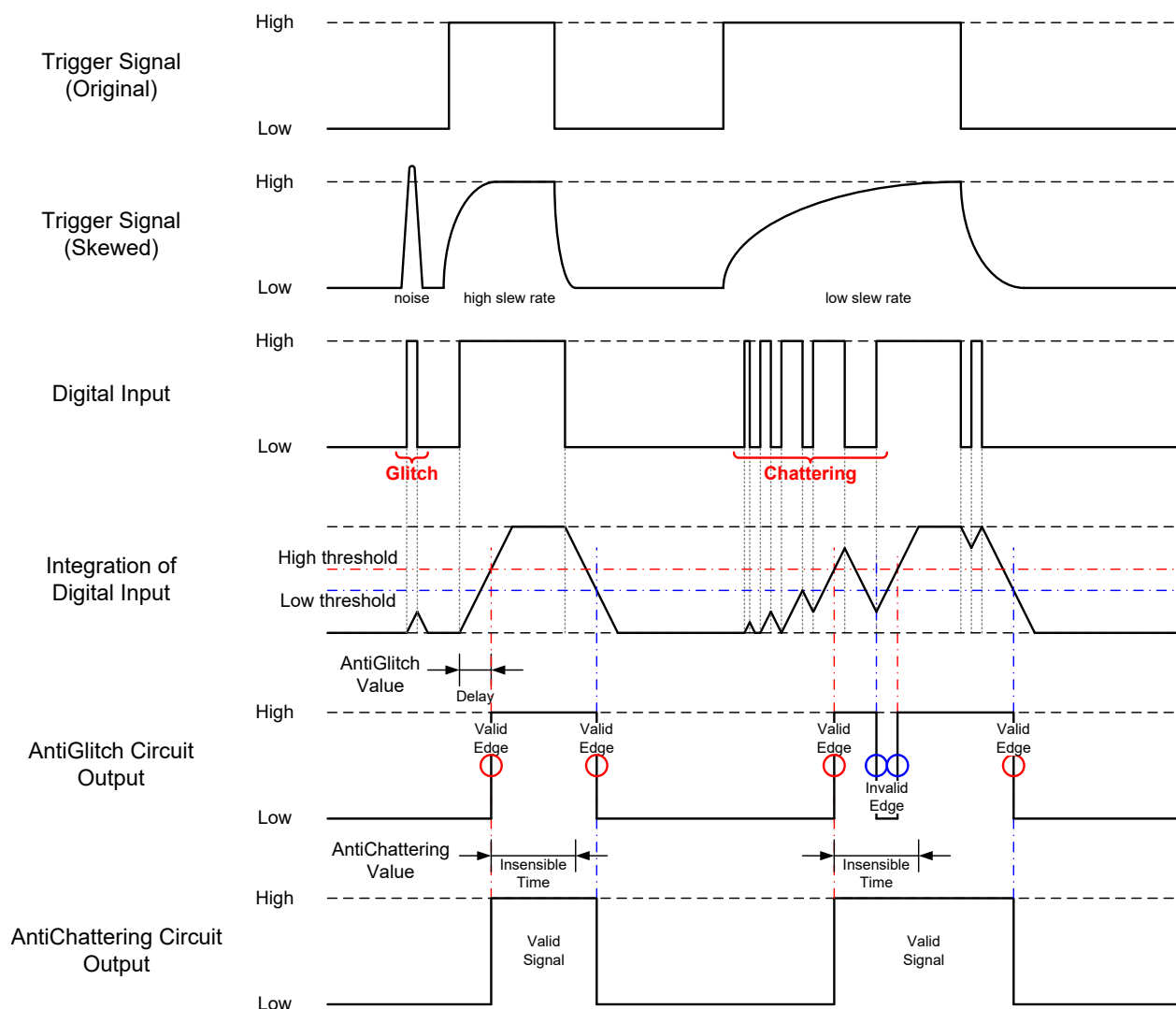
AntiGlitch / AntiChattering

AntiGlitch and AntiChattering functions filter noise and unstable state of the digital input (trigger signal).

AntiGlitch circuit performs the digital integration of the trigger signal.

It is effective to remove impulsive noise.

AntiChattering circuit sets the edge insensible time to avoid trigger malfunction.

It is effective to remove unstable logic state and switch-chattering.

AntiGlitch and AntiChattering

D4288264C

https://www.toshiba-teli.co.jp/en/

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
AntiGlitch	IFloat	4	R/W	Integration time of digital input signal. [sec]
AntiChattering	IFloat	4	R/W	Insensible time of digital input signal. [sec]

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
AntiGlitch	Implemented	0x21F3C0	[31]	R	Returns the state whether the function is implemented.
	Mult	0x21F3CC	4	R	Absolute Value [sec] = Raw * (Mult / Div)
	Div	0x21F3D0	4	R	
	Min	0x21F3D4	4	R	Returns the minimum value of AntiGlitch period.
	Max	0x21F3D8	4	R	Returns the maximum value of AntiGlitch period.
	Value	0x21F3DC	4	R/W	Sets the AntiGlitch period.(Raw value)
AntiChattering	Implemented	0x21F3E0	[31]	R	Returns the state whether the function is implemented.
	Mult	0x21F3EC	4	R	Absolute Value [sec] = Raw * (Mult / Div)
	Div	0x21F3F0	4	R	
	Min	0x21F3F4	4	R	Returns the minimum value of AntiGlitch period.
	Max	0x21F3F8	4	R	Returns the maximum value of AntiGlitch period.
	Value	0x21F3FC	4	R/W	Sets the AntiGlitch period.(Raw value)

● Control with TeliCamSDK

GenICam function API

Control AntiGlitch/AntiChattering using GenICam API.

◆ AntiGlitch/AntiChattering

1. Set Integration time of digital input signal in [sec] to 'AntiGlitch'.

2. Set Insensible time of digital input signal in [sec] to 'AntiChattering'.

```
// GenICam node handle
CAM_NODE_HANDLE    hGlitch = NULL;
CAM_NODE_HANDLE    hChattering = NULL;

// AntiGlitch = 1.0[us], AntiChattering = 10.0[us]
float64_t      dGlitch = 0.000001;
float64_t      dChattering = 0.000010;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "AntiGlitch", &hGlitch);
Nd_GetNode(s_hCam, "AntiChattering", &hChattering);

// Set Value
Nd_SetFloatValue(s_hCam, hGlitch, dGlitch);
Nd_SetFloatValue(s_hCam, hChattering, dChattering);
```

Please refer to [INode functions], [IFloat node functions] in [TeliCamAPI Library manual] for more detail.

D4288264C

https://www.toshiba-teli.co.jp/en/

Register access API

Control AntiGlitch/AntiChattering by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ AntiGlitch/AntiChattering

1. Write to 'Value' field of 'AntiGlitch' register.

2. Write to 'Value' field of 'AntiChattering' register.

```
// AntiGlitch = 1.0[us] (Raw value = 100)
// AntiChattering = 10.0[us] (Raw value = 1250)
uint32_t        uiAntiGlitchRaw = 100;
uint32_t        uiAntiChatteringRaw = 1250;

// Set Value
Cam_WriteReg(s_hCam, 0x21F3DC, 1, &uiAntiGlitchRaw);
Cam_WriteReg(s_hCam, 0x21F3FC, 1, &uiAntiChatteringRaw);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ Minimum / Maximum Value

		AntiGlitch	AntiChattering
Minimum	Raw Value	1	49
	Absolute Value	0.05 [us]	1.96 [us]
Maximum	Raw Value	40000	49999
	Absolute Value	2000.00 [us]	1999.96 [us]
Initial Value	Raw Value	1	49
	Absolute Value	0.05 [us]	1.96 [us]
Formula		Absolute Value [us] = Raw Value / 20	Absolute Value [us] = Raw Value / 25

D4288264C

TimerControl

This section describes TimerControl features.

This camera series is able to generate Timer0Active signal beginning from trigger or exposure start signal.

This signal can be used as strobe control signal.
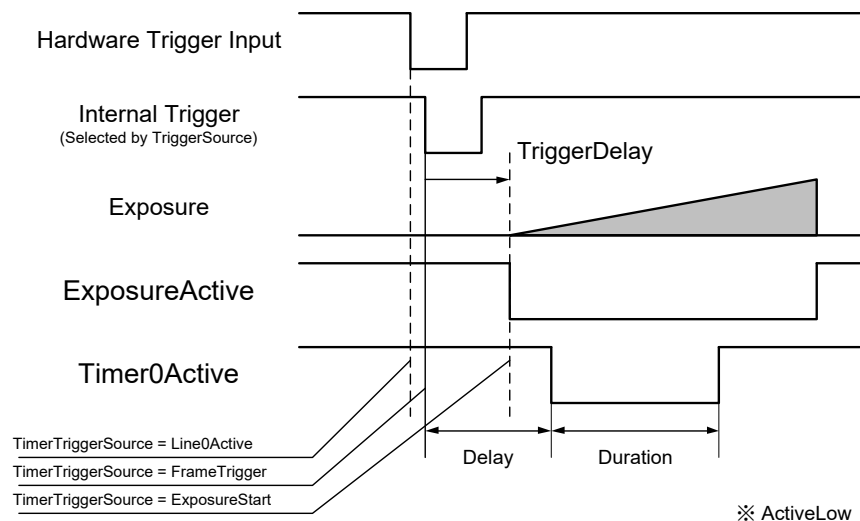
Timer0Active

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
TimerSelector	IEnumeration	4	R	Returns the selected timer name.
TimerDelay	IFloat	4	R/W	Sets the delay of Timer0Active signal.
TimerDuration	IFloat	4	R/W	Sets the duration of Timer0Active signal.
TimerTriggerSource	IEnumeration	4	R/W	Selects the source of Timer0Active pulse to start.

D4288264C

https://www.toshiba-teli.co.jp/en/

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
TimerSelector	Implemented	0x20A020	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20A02C	4	R	[0] : Timer0
	Value	0x20A03C	4	R/W	Returns the selected timer.
TimerDelay	Implemented	0x20A040	[31]	R	Returns the state whether the function is implemented.
	Mult	0x20A04C	4	R	AbsoluteValue [sec] = Raw * (Mult / Div)
	Div	0x20A050	4	R	
	Min	0x20A054	4	R	Returns the minimum value of TimerDelay Value register.
	Max	0x20A058	4	R	Returns the maximum value of TimerDelay Value register.
	Value	0x20A05C	4	R/W	Sets Timer0Active delay time.(Raw value)
TimerDuration	Implemented	0x20A060	[31]	R	Returns the state whether the function is implemented.
	Mult	0x20A06C	4	R	AbsoluteValue [sec] = Raw * (Mult / Div)
	Div	0x20A070	4	R	
	Min	0x20A074	4	R	Returns the minimum value of TimerDuration Value register.
	Max	0x20A078	4	R	Returns the maximum value of TimerDuration Value register.
	Value	0x20A07C	4	R/W	Sets Timer0Active duration time.(Raw value)
TimerTriggerSource	Implemented	0x20A080	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20A08C	4	R/W	[0] : Off [32] : Line0 [104] : FrameTrigger [124] : ExposureStart
	Value	0x20A09C	4	R/W	Selects the source of Timer0Active pulse to start.

● Control with TeliCamSDK

Camera feature API

Control TimerControl using dedicated API.

API name	Description
GetCamTimerDelayMinMax	Get minimum and maximum value
GetCamTimerDelay	Get current TimerDelay value
SetCamTimerDelay	Set new TimerDelay value
GetCamTimerDurationMinMax	Get minimum and maximum value
GetCamTimerDuration	Get current TimerDuration value
SetCamTimerDuration	Set new TimerDuration value
GetCamTimerTriggerSource	Get current TimerTriggerSource
SetCamTimerTriggerSource	Set new TimerTriggerSource

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

https://www.toshiba-teli.co.jp/en/

GenICam function API

Control TimerControl using GenICam API.

◆ **TimerDelay/TimerDuration/TimerTriggerSource**

1. Sets the delay of Timer0Active signal by 'TimerDelay'.

2. Sets the duration of Timer0Active signal by 'TimerDuration'.

3. Selects the source of Timer0Active pulse to start by 'TimerTriggerSource'.

   Integer value and string value of Enumeration are as follows.

Integer	String	Description
0	Off	DisablesTimer0Active signal.
32	Line0	Starts when Line0 is active.
104	FrameTrigger	Starts with the reception of the Frame Start Trigger.
124	ExposureStart	Starts with the reception of the Exposure Start

```
// GenICam node handle
CAM_NODE_HANDLE   hDelay = NULL;
CAM_NODE_HANDLE   hDuration = NULL;
CAM_NODE_HANDLE   hSource = NULL;

// TimerDelay = 1000.0[us], TimerDuration = 2000.0[us]
float64_t     dDelay = 1000.0;
float64_t     dDuration = 2000.0;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "TimerDelay", &hDelay);
Nd_GetNode(s_hCam, "TimerDuration", &hDuration);
Nd_GetNode(s_hCam, "TimerTriggerSource", &hSource);

// 1.Sets the delay of Timer0Active signal.
Nd_SetFloatValue(s_hCam, hDelay, dDelay);

// 2.Sets the dulation of Timer0Active signal.
Nd_SetFloatValue(s_hCam, hDuration, dDuration);

// 3.Selects the source of Timer0Active pulse to start.
Nd_SetEnumStrValue(s_hCam, hSource, "ExposureStart");
```

Please refer to [INode functions], [IFloat node functions] and [IEnumeration node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control TimerControl by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **TimerDelay/TimerDuration/TimerTriggerSource**

1. Write to 'Value' field of 'TimerDelay' register.

2. Write to 'Value' field of 'TimerDuration' register.

3. Write to 'Value' field of 'TimerTriggerSource' register.

```
// TimerDelay = 1000.0[us] (Raw value = 25000)
// TimerDuration = 2000.0[us] (Raw value = 50000)
// TimerTriggersource = "ExposureStart"
uint32_t      uiTimerDelayRaw = 25000;
uint32_t      uiTimerDurationRaw = 50000;
uint32_t      uiTimerTriggerSource = 124;

// Set Value
Cam_WriteReg(s_hCam, 0x20A05C, 1, &uiTimerDelayRaw);
Cam_WriteReg(s_hCam, 0x20A07C, 1, &uiTimerDurationRaw);
Cam_WriteReg(s_hCam, 0x21F27C, 1, &uiTimerTriggerSource);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ **Minimum / Maximum Value**

|		TimerDelay / TimerDuration
Minimum	Raw Value	0
	Absolute Value	0.00 [us]
Maximum	Raw Value	50000000
	Absolute Value	2000000.00 [us]
Initial Value	Raw Value	0
	Absolute Value	0.00 [us]
Formula		Absolute Value [us] = Raw Value / 25

● **Note**

'TimerTriggerSource' operation is as follows,

- Line0Active is available in Hardware Trigger mode, and to be responded to FrameTriggerError.

- FrameTrigger is available in both Hardware Trigger and Software Trigger mode, and not to be responded to FrameTriggerError.

- ExposureStart is available in both Hardware Trigger and Software Trigger mode, and not to be responded to FrameTriggerError. Timer0Active pulse delays TiggerDelay+TimerDelay[us].

Gain

This section describes Gain feature. This control adjusts an amplification factor applied to the output signal. Gain feature adjusts manual gain. GainAuto feature adjusts gain automatically.
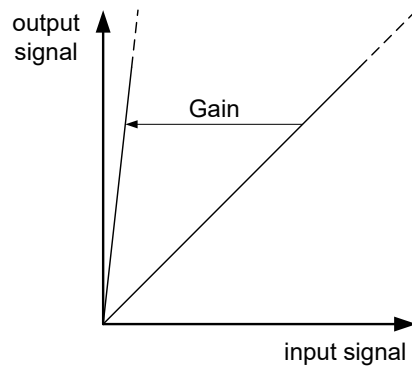
The formula of Gain value is as follows;

$$\text{output signal} = \text{input signal} \times 10^{\frac{\text{Gain}}{20}}$$

Gain is in [dB] unit.

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
Gain	IFloat	4	R/W	Sets a manual Gain.
GainAuto	IEnumeration	4	R/W	Adjusts Gain automatically.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
Gain	Implemented	0x204060	[31]	R	Returns the state whether the function is implemented.
	Control	0x204068	4	R/W	[1]: Manual [2]: Auto
	Mult	0x20406C	4	R	Absolute Value [dB] = Raw * (Mult / Div)
	Div	0x204070	4	R	
	Min	0x204074	4	R	Returns the minimum value of Gain Value register.
	Max	0x204078	4	R	Returns the maximum value of Gain Value register.
	Value	0x20407C	4	R/W	Gain value. (Raw value)

D4288264C

● Control with TeliCamSDK

Camera feature API

Control Gain using dedicated API.

API name	Description
GetCamGainMinMax	Get minimum and maximum value
GetCamGain	Get current Gain value
SetCamGain	Set new Gain value
GetCamGainAuto	Get current GainAuto mode
SetCamGainAuto	Set new GainAuto mode

GainAuto parameter	Description
CAM_GAIN_AUTO_OFF	Off
CAM_GAIN_AUTO_AUTO	Adjust continuously

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control Gain using GenICam API.

◆ **Gain**

Use 'IFloat' interface to control Gain.

```
// GenICam node handle
CAM_NODE_HANDLE    hNode = NULL;

// Gain = 6.0dB
float64_t       dGain = 6.0;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "Gain", &hNode);

// Set Gain Value
Nd_SetFloatValue(s_hCam, hNode, dGain);
```

◆ **GainAuto**

Use 'IEnumertion' interface to control GainAuto.

Integer value and string value of Enumeration are as follows.

Integer	String	Description
1	Off	Operates Manual mode.
2	Auto	Operates Auto mode.

D4288264C

```
// GenICam node handle
CAM_NODE_HANDLE    hNode = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "GainAuto", &hNode);

// Set GainAuto = "Auto"
Nd_SetEnumStrValue(s_hCam, hNode, "Auto");
```

Please refer to [INode functions], [IFloat node functions] and [IEnumeration node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control Gain by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ Gain

Write to 'Value' field of 'Gain' register.

```
// Gain = 6.0dB (Raw value = 60)
uint32_t      uiGainRaw = 60;

// Set Gain Value
Cam_WriteReg(s_hCam, 0x20407C, 1, &uiGainRaw);
```

◆ GainAuto

Write to 'Control' field of 'Gain' register.

```
// GainAuto = "Auto"
uint32_t      uiGainAuto = 2;

// Set GainAuto = "Auto"
Cam_WriteReg(s_hCam, 0x204068, 1, &uiGainAuto);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ Minimum / Maximum Value

Gain	Raw Value	Absolute (Float) Value
Minimum	0	0.00 [dB]
Maximum	Ver.3.3.9 or earlier : 240 Ver.3.5.0 or later : 360	Ver.3.3.9 or earlier : 24.00 [dB] Ver.3.5.0 or later : 36.00 [dB]
Initial Value	0	0.00 [dB]
Formula	Absolute Value [dB] = Raw Value / 10	

● Note

- In GainAuto mode, current Gain will be reflected to Gain value register.

- Range of gain setting at GainAuto mode is 0 to +24dB.

D4288264C
https://www.toshiba-teli.co.jp/en/

BlackLevel

This section describes BlackLevel feature. This control adjusts the black level applied to the output signal.

It is adjustable from -25% to +25% as white saturation level is 100%.

If BlackLevel is set lower than 0[%], the image level may not be saturated.
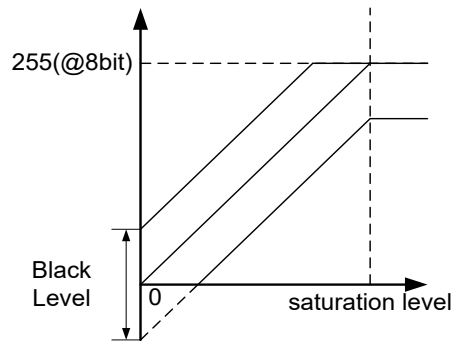
Black Level

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
BlackLevel	IFloat	4	R/W	Black Level value.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
BlackLevel	Implemented	0x204040	[31]	R	Returns the state whether the function is implemented.
	Mult	0x20404C	4	R	Absolute Value [%] = Raw * (Mult / Div)
	Div	0x204050	4	R	
	Min	0x204054	4	R	Returns the minimum value of Black Level Value register.
	Max	0x204058	4	R	Returns the maximum value of Black Level Value register.
	Value	0x20405C	4	R/W	Black Level value. (Raw value)

● Control with TeliCamSDK

Camera feature API

Control BlackLevel using dedicated API.

API name	Description
GetCamBlackLevelMinMax	Get minimum and maximum value
GetCamBlackLevel	Get current BlackLevel value
SetCamBlackLevel	Set new BlackLevel value

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

D4288264C

GenICam function API

Control BlackLevel using GenICam API.

◆ **BlackLevel**

Use 'IFloat' interface to control BlackLevel.

```
// GenICam node handle
CAM_NODE_HANDLE    hNode = NULL;

// BlackLevel = 25%
float64_t       dBlackLevel = 25.0;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "BlackLevel", &hNode);

// Set BlackLevel Value
Nd_SetFloatValue(s_hCam, hNode, dBlackLevel);
```

Please refer to [INode functions] and [IFloat node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control BlackLevel by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **BlackLevel**

Write to 'Value' field of 'BlackLevel' register.

```
// BlackLevel = 25% (Raw value = 256)
int32_t      iBlackLevelRaw = 256; // signed

// Set BlackLevel Value
Cam_WriteReg(s_hCam, 0x20405C, 1, &iBlackLevelRaw);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ **Minimum / Maximum Value**

BlackLevel	Raw Value	Absolute (Float) Value
Minimum	-256	-25.00[%]
Maximum	256	+25.00[%]
Initial Value	0	0.00[%]
Formula	Absolute Value [%] = Raw Value * 100 / 1024	

D4288264C

https://www.toshiba-teli.co.jp/en/

Gamma

This section describes Gamma feature. This control adjusts the gamma correction of pixel intensity.
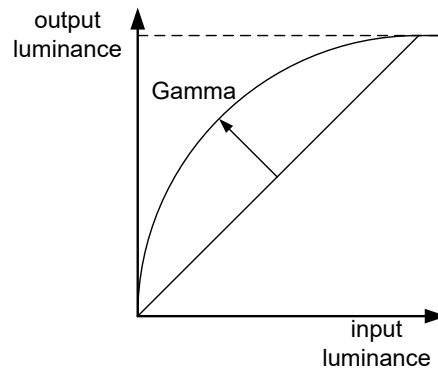
Gamma correction

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
Gamma	IFloat	4	R/W	Gamma value.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
Gamma	Implemented	0x204080	[31]	R	Returns the state whether the function is implemented.
	Mult	0x20408C	4	R	Absolute Value = Raw * (Mult / Div)
	Div	0x204090	4	R	
	Min	0x204094	4	R	Returns the minimum value of Gamma Value register.
	Max	0x204098	4	R	Returns the maximum value of Gamma Value register.
	Value	0x20409C	4	R/W	Gamma value. (Raw value)

● Control with TeliCamSDK

Camera feature API

Control Gamma using dedicated API.

API name	Description
GetCamGammaMinMax	Get minimum and maximum value
GetCamGamma	Get current Gamma value
SetCamGamma	Set new Gamma value

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

D4288264C

GenICam function API

Control Gamma using GenICam API.

◆ **Gamma**

Use 'IFloat' interface to control Gamma.

```
// GenICam node handle
CAM_NODE_HANDLE    hNode = NULL;

// Gamma = 0.45
float64_t        dGamma = 0.45;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "Gamma", &hNode);

// Set Gamma Value
Nd_SetFloatValue(s_hCam, hNode, dGamma);
```

Please refer to [INode functions] and [IFloat node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control Gamma by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **Gamma**

Write to 'Value' field of 'Gamma' register.

```
// Gamma = 0.45 (Raw value = 45)
uint32_t        uiGammaRaw = 45;

// Set Gamma Value
Cam_WriteReg(s_hCam, 0x20409C, 1, &uiGammaRaw);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ **Minimum / Maximum Value**

Gamma	Raw Value	Absolute (Float) Value
Minimum	45	0.45
Maximum	100	1.00
Initial Value	100	1.00
Formula	Absolute Value = Raw Value / 100	

D4288264C
https://www.toshiba-teli.co.jp/en/

Sharpness

This section describes Sharpness feature. This control enhances the edges of the image.
The edge enhancement will become stronger when the setting value is larger.
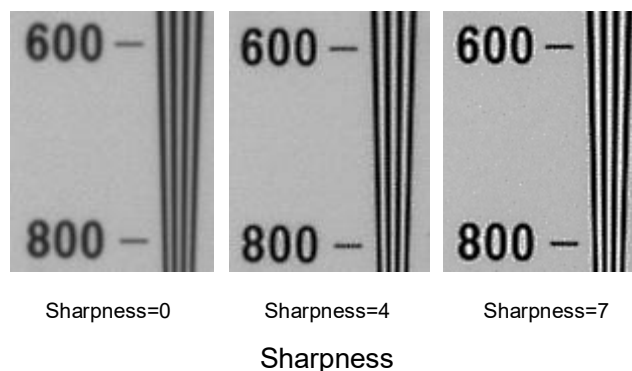
Sharpness=0   Sharpness=4   Sharpness=7

Sharpness

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
Sharpness	IInteger	4	R/W	Sets sharpness value.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
Sharpness	Implemented	0x2040A0	[31]	R	Returns the state whether the function is implemented.
	Min	0x2040B4	4	R	Returns the minimum value of sharpness Value register.
	Max	0x2040B8	4	R	Returns the maximum value of sharpness Value register.
	Value	0x2040BC	4	R/W	Sets sharpness value. (Raw value)

● Control with TeliCamSDK

Camera feature API

Control Sharpness using dedicated API.

API name	Description
GetCamSharpnessMinMax	Get minimum and maximum value
GetCamSharpness	Get current Sharpness value
SetCamSharpness	Set new Sharpness value

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control Sharpness using GenICam API.

◆ Sharpness

Use 'IInteger' interface to control Sharpness.

```
// GenICam node handle
CAM_NODE_HANDLE   hNode = NULL;

// Sharpness = 4
int64_t     nSharpness = 4;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "Sharpness", &hNode);

// Set Sharpness Value
Nd_SetFloatValue(s_hCam, hNode, nSharpness);
```

Please refer to [INode functions] and [IInteger node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control Sharpness by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ Sharpness

Write to 'Value' field of 'Sharpness' register.

```
// Sharpness = 4 (Raw value = 4)
uint32_t     uiSharpnessRaw = 4;

// Set Sharpness Value
Cam_WriteReg(s_hCam, 0x2040BC, 1, &uiSharpnessRaw);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ Minimum / Maximum Value

Sharpness	Raw Value	Absolute (Integer) Value
Minimum	0	0
Maximum	7	7
Initial Value	0	0
Formula	Absolute Value = Raw Value	

D4288264C

ALCControl

This section describes ALCControl feature.

This camera series provides some registers to adjust the various parameter of ALC operation.

- ALCPhotometricAreaSize defines photometric area size for measuring luminance.
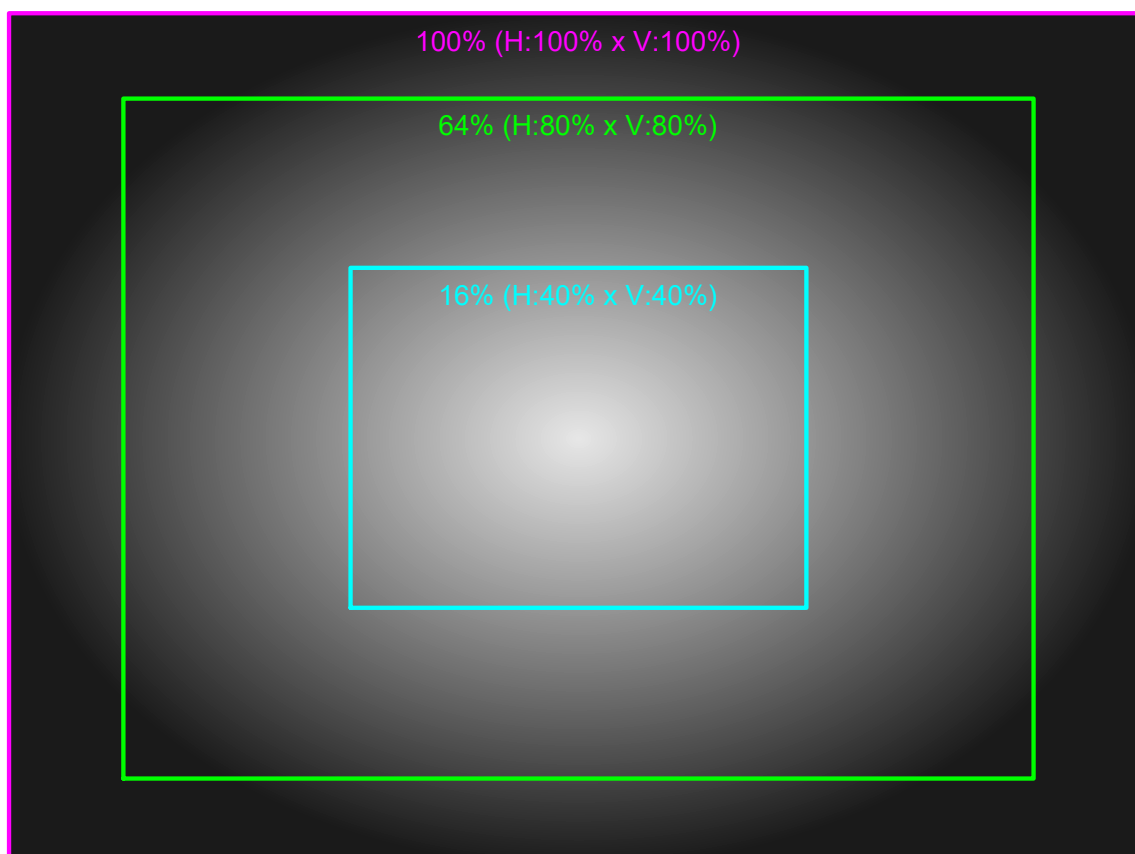
ALCPhotometricAreaSize (e.g. 100%, 80%, 40%)

- ALCExposureValue defines a correction value for a convergence value.

Final convergence value is determined by the following formula.

$$\text{Final convergence value} = 84(\text{Reference Luminance}) \times 2^{ALCExposureValue}$$

D4288264C

https://www.toshiba-teli.co.jp/en/

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
ALCPhotometricAreaSize	IFloat	4	R/W	Sets ALC photometric area size.
ALCExposureValue	IFloat	4	R/W	Sets ALC Exposure Value.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
ALCPhotometricAreaSize	Implemented	0x21F360	[31]	R	Returns the state whether the function is implemented.
	Mult	0x21F36C	4	R	AbsoluteValue= Raw * (Mult / Div)
	Div	0x21F370	4	R	
	Min	0x21F374	4	R	Returns the minimum value of ALC photometric area size.
	Max	0x21F378	4	R	Returns the maximum value of ALC photometric area size.
	Value	0x21F37C	4	R/W	Sets ALC photometric area size. (Raw value)
ALCExposureValue	Implemented	0x2040C0	[31]	R	Returns the state whether the function is implemented.
	Mult	0x2040CC	4	R	AbsoluteValue= Raw * (Mult / Div)
	Div	0x2040D0	4	R	
	Min	0x2040D4	4	R	Returns the minimum value of ALC Exposure Value.
	Max	0x2040D8	4	R	Returns the maximum value of ALC Exposure Value.
	Value	0x2040DC	4	R/W	Sets ALC Exposure Value. (Raw value)

D4288264C
https://www.toshiba-teli.co.jp/en/

● **Control with TeliCamSDK**

Control ALC parameters using GenICam API.

◆ **ALCPhotometricAreaSize/ALCExposureValue**

Use 'IFloat' interface to control ALC parameter.

```
// GenICam node handle
CAM_NODE_HANDLE   hSize = NULL;
CAM_NODE_HANDLE   hEV = NULL;

// ALCPhotometricAreaSize = 25%
float64_t      dSize = 25.0;
// ALCExposureValue = +1.0EV
float64_t      dEV = 1.0;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "ALCPhotometricAreaSize", &hSize);
Nd_GetNode(s_hCam, "ALCExposureValue", &hEV);

// Set ALCPhotometricAreaSize
Nd_SetFloatValue(s_hCam, hSize, dSize);
// Set ALCExposureValue
Nd_SetFloatValue(s_hCam, hEV, dEV);
```

Please refer to [INode functions] and [IFloat node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control ALC parameters by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **ALCPhotometricAreaSize/ALCExposureValue/ALCExposureTimeMin/ALCExposureTimeMax/ALCGainMin/ALCGainMax/ALCThrottle**

Write to 'Value' field of ALC parameter register.

```
// ALCPhotometricAreaSize = 25% (Raw value = 25)
uint32_t    uiSizeRaw = 25;
// ALCExposureValue = +1.0EV (Raw value = 10)
int32_t     iEVRaw = 10; // signed

// Set ALCPhotometricAreaSize
Cam_WriteReg(s_hCam, 0x21F37C, 1, &uiSizeRaw);
// Set ALCExposureValue
Cam_WriteReg(s_hCam, 0x2040DC, 1, &iEVRaw);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ **Minimum / Maximum Value**

ALCPhotometricAreaSize	Raw Value	Absolute (Float) Value
Minimum	1	1.00[%]
Maximum	100	100.00[%]
Initial Value	100	100.00[%]
Formula	Absolute Value [%] = Raw Value	

ALCExposureValue	Raw Value	Absolute (Float) Value
Minimum	-20	-2.0[EV]
Maximum	15	+1.5[EV]
Initial Value	0	0.0[EV]
Formula	Absolute Value [EV] = Raw Value / 10	

● **Note**

- The camera operation is not supported when the Random Trigger Shutter mode.

D4288264C

https://www.toshiba-teli.co.jp/en/

LUTControl

This function allows you to apply the arbitrary LUT(input: 12it, output: 12bit) to the output images.
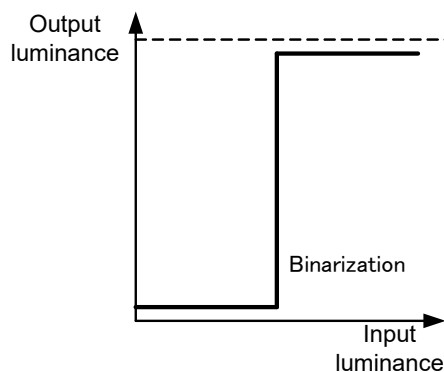
example of LUT setting

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
LUTEnable	IBoolean	4	R/W	Sets the activation of LUT function.
LUTIndex	IInteger	4	R/W	Sets the input level of LUT.
LUTValue	IInteger	4	R/W	Sets the output level of LUT.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
LUTEnable	Implemented	0x206020	[31]	R	Returns the state whether the function is implemented.
	Value	0x20603C	4	R/W	Sets the activation of LUT function. [0] : Off [1] : On
LUTValueAll	Implemented	0x2FFFDC	[31]	R	Returns the state whether the function is implemented.
	Value[0]	0x300000	4	R/W	Sets the output level of LUT.
	Value[1]	0x300004	4	R/W	Sets the output level of LUT.
	Value[2]	0x300008	4	R/W	Sets the output level of LUT.
	…	…	…	…	…
	Value[4095]	0x303FFC	4	R/W	Sets the output level of LUT.

● Control with TeliCamSDK

Camera feature API

Control LUT using dedicated API.

API name	Description
GetCamLUTEnable	Get activation status
SetCamLUTEnable	Set activation status
GetCamLUTValue	Get LUT value
SetCamLUTValue	Set LUT value

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control LUT using GenICam API.

◆ **LUT**

1. Set the input level of LUT to 'LUTIndex'.

2. Set the output level of LUT to 'LUTValue'

3. Set the activation of LUT function by 'LUTEnable'

```
// GenICam node handle
CAM_NODE_HANDLE   hIndex = NULL;
CAM_NODE_HANDLE   hValue = NULL;
CAM_NODE_HANDLE   hEnable = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "LUTIndex", &hIndex);
Nd_GetNode(s_hCam, "LUTValue", &hValue);
Nd_GetNode(s_hCam, "LUTEnable", &hEnable);

for(int64_t i=0; i<4096; i++)
{
    // 1.Set the input level of LUT to 'LUTIndex'.
    Nd_SetIntValue(s_hCam, hIndex, i);
    // 2.Set the output level of LUT to 'LUTValue'
    Nd_SetIntValue(s_hCam, hValue, 4095 - i); // invert
}

// 3.Set the activation of LUT function by 'LUTEnable'
Nd_SetBoolValue(s_hCam, hEnable, true);
```

Please refer to [INode functions], [IInteger node functions] and [IBoolean node functions] in [TeliCamAPI Library manual] for more detail.

D4288264C
https://www.toshiba-teli.co.jp/en/

Register access API

Control LUT by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ LUT

1. Set the output level of LUT.

   Write to 'Value[0]' to 'Value[4095]' of 'LUTValueAll' register to set the output level of LUT.

   The index of 'Value' register means the input level of LUT.

2. Set the activation of LUT function.

   Write to 'Value' field of 'LUTEnable' register to activate LUT function.

```
// 1.Set the output level of LUT.
uint64_t   addr = 0x300000;
uint32_t   dat;
for(int i=0; i<4096; i++, addr+= 0x4)
{
        dat = 4095 - i; // invert
        Cam_WriteReg(s_hCam, addr, 1, &dat);
}

// 2.Set the activation of LUT function.
dat = 1;
Cam_WriteReg(s_hCam, 0x20603C, 1, &dat);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ Minimum / Maximum Value

LUTIndex/LUTValue	Value
Minimum	0
Maximum	4095

D4288264C

SequentialShutterControl

Sequential Shutter function performs sequential capturing with applying the settings of UserSet that have been made entry in advance. This feature is not supported when the ShortExposureMode is enable (ON).
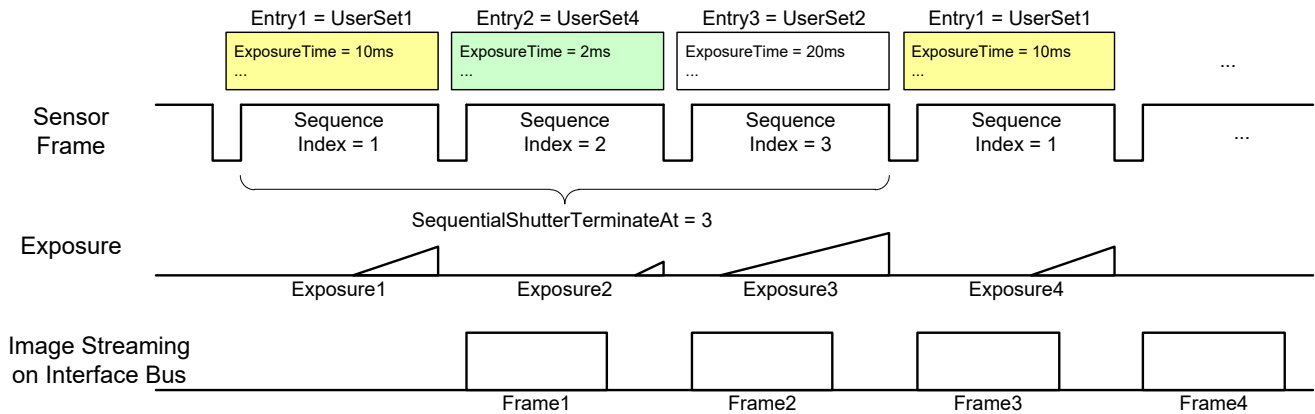
Sequential Shutter

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
SequentialShutterEnable	IEnumeration	4	R/W	Sets the activation of Sequential Shutter function.
SequentialShutterTerminateAt	IInteger	4	R/W	Sets the number of Index to repeat the sequence.
SequentialShutterIndex	IInteger	4	R/W	Sets the sequence number to register.
SequentialShutterEntry	IInteger	4	R/W	Sets the UserSet number to register to the sequence.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
SequentialShutterEnable	Implemented	0x21F300	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x21F30C	4	R	[0] : Off [1] : On
	Value	0x21F31C	4	R/W	Sets the activation of Sequential Shutter function.
SequentialShutterTerminateAt	Implemented	0x21F320	[31]	R	Returns the state whether the function is implemented.
	Value	0x21F33C	4	R/W	Sets the number of Tables to repeat the sequence.
SequentialShutterSequenceTable	Value[0]	0x500040	4	R/W	Sets the UserSet number to register to the first sequence.
	Value[1]	0x500044	4	R/W	Sets the UserSet number to register to the second sequence.
	…	…	…	…	…
	Value[15]	0x500078	4	R/W	Sets the UserSet number to register to the 16th sequence.

D4288264C

https://www.toshiba-teli.co.jp/en/

● Control with TeliCamSDK
Camera feature API

Control SequentialShutter using dedicated API.

API name	Description
GetCamSequentialShutterEnable	Get activation status
SetCamSequentialShutterEnable	Set activation status
GetCamSequentialShutterTerminateAtMinMax	Get minimum and maximum value
GetCamSequentialShutterTerminateAt	Get current SequentialShutterTerminateAt value
SetCamSequentialShutterTerminateAt	Set new SequentialShutterTerminateAt value
GetCamSequentialShutterIndexMinMax	Get minimum and maximum value
GetCamSequentialShutterEntryMinMax	Get minimum and maximum value
GetCamSequentialShutterEntry	Get current SequentialShutterEntry value
SetCamSequentialShutterEntry	Set new SequentialShutterEntry value

GenICam function API

Control SequentialShutter using GenICam API.

◆ SequentialShutter

1. Set the camera parameters and save them to UserSet. Repeat this as you required.
2. Set the sequence number to register to 'SequentialShutterIndex' and Sets the UserSet number to register to the sequence to 'SequentialShutterEntry'. Repeat this as you required.
3. Set the number of Index to repeat the sequence to 'SequentialShutterTerminateAt'.
4. Set the activation of SequentialShutter function by 'SequentialShutterEnable'.

   Integer value and string value of Enumeration are as follows.

Integer	String
0	Off
1	On

5. Capture image stream.

   To start and stop capturing image is the same way as AcquisitionControl.

   Please refer to [Camera streaming functions] in [TeliCamAPI Library manual]

   and [GrabStreamSimple] sample code in your TeliCamSDK install folder.

```
// 1.Set the camera parameters and save them to UserSet.
...

// GenICam node handle
CAM_NODE_HANDLE   hIndex = NULL;
CAM_NODE_HANDLE   hEntry = NULL;
CAM_NODE_HANDLE   hTerminateAt = NULL;
CAM_NODE_HANDLE   hEnable = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "SequentialShutterIndex", &hIndex);
Nd_GetNode(s_hCam, "SequentialShutterEntry", &hEntry);
Nd_GetNode(s_hCam, "SequentialShutterTerminateAt", &hTerminateAt);
Nd_GetNode(s_hCam, "SequentialShutterEnable", &hEnable);

int64_t     entry[] = {1,4,2};      // Sequence = {UserSet1,UserSet4,UserSet2}
int64_t     index;

for(index=0; index<3; index++)
{
 // 2.Set SequentialShutterIndex and SequentialShutterEntry.
 Nd_SetIntValue(s_hCam, hIndex, index+1); // 1 origin
 Nd_SetIntValue(s_hCam, hEntry, entry[index]);
}

// 3.Set SequentialShutterTerminateAt.
Nd_SetIntValue(s_hCam, hTerminateAt, index); // 3 UserSet will be repeated.

// 4.Set the activation of SequentialShutter function by 'SequentialShutterEnable'.
Nd_SetEnumStrValue(s_hCam, hEnable, "On");
```

Please refer to [INode functions], [IInteger node functions] and [IEnumeration node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control SequentialShutter by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

1. Set the camera parameters and save them to UserSet. Repeat this as you required.
2. Sets the UserSet number to register to the sequence.

   Write to 'Value[0]' to 'Value[15]' of 'SequentialShutterSequenceTable'.

   The index of 'Value' register means sequence index. (0 origin)
3. Set the number of Index to repeat the sequence.

   Write to 'Value' field of 'SequentialShutterTerminateAt'.
4. Set the activation of SequentialShutter function.

   Write to 'Value' field of 'SequentialShutterEnable' register to activate SequentialShutter function.

D4288264C
https://www.toshiba-teli.co.jp/en/

5. Capture image stream.

   To start and stop capturing image is the same way as AcquisitionControl.

   Please refer to [Camera streaming functions] in [TeliCamAPI Library manual]

   and [GrabStreamSimple] sample code in your TeliCamSDK install folder.

```
// 1.Set the camera parameters and save them to UserSet.
...

uint64_t   addr = 0x500040;
uint32_t   entry[] = {1,4,2};       // Sequence = {UserSet1,UserSet4,UserSet2}
uint32_t   index, dat;

for(index=0; index<3; index++,addr+=4)
{
  // 2.Set SequentialShutterIndex and SequentialShutterEntry.
  Cam_WriteReg(s_hCam, addr, 1, &entry[index]);
}

// 3.Set SequentialShutterTerminateAt.
Cam_WriteReg(s_hCam, 0x21F33C, 1, &index); // 3 UserSet will be repeated.

// 4.Set the activation of SequentialShutter function.
dat = 1;
Cam_WriteReg(s_hCam, 0x21F31C, 1, &dat);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ **Minimum / Maximum Value**

SequentialShutterIndex	setting value
Minimum (*)	1
Maximum	16

* initial factory setting

SequentialShutterEntry	setting value
Minimum (*)	1
Maximum	15

* initial factory setting

D4288264C
https://www.toshiba-teli.co.jp/en/

● Note

- Changing 'SequentialShutterEnable', 'SequentialShutterTerminateAt', 'SequentialShutterIndex', 'SequentialShutterEntry' register value is invalid during image stream data output.
- The following table is the list of registers applied to "SequentialShutter".

List of registers to be applied to SequentialShutter

Category	Register
ImageFormatControl	OffsetX
	OffsetY
ExposureControl	ExposureTime
DigitalIOControl	UserOutputValueAll
	LineSource
CounterAndTimerControl	TimerDuration
	TimerDelay

Category	Register
AnalogControl	Gain
	BlackLevel
	Gamma
	Sharpness
LUTControl	LUTEnable

DPCControl

This DPC (Defective Pixel Correction) function corrects defective pixels from the image sensor. Specifying X and Y coordinates of the defective pixels, the defective pixels are corrected by calculation from the neighboring pixels.

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
DPCEnable	IEnumeration	4	R/W	Sets the activation of DPC function.
DPCNumber	IInteger	4	R/W	Sets the number of pixels to correct.
DPCIndex	IInteger	4	R/W	Sets the index number for entry.
DPCEntryX	IInteger	4	R/W	Sets the X coordinate of defective pixel.
DPCEntryY	IInteger	4	R/W	Sets the Y coordinate of defective pixel.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
DPCEnable	Implemented	0x21F280	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x21F28C	4	R	[0] : Off [1] : On
	Value	0x21F29C	4	R/W	Sets the activation of DPC function.
DPCNumber	Implemented	0x21F2A0	[31]	R	Returns the state whether the function is implemented.
	Value	0x21F2BC	4	R/W	Sets the number of Pixels to correct.
DPCCoord	Value[0][0]	0x400000	4	R/W	Sets the first X coordinate of defective pixel.
	Value[0][1]	0x400004	4	R/W	Sets the first Y coordinate of defective pixel.
	Value[1][0]	0x400008	4	R/W	Sets the second X coordinate of defective pixel.
	Value[1][1]	0x40000C	4	R/W	Sets the second Y coordinate of defective pixel.
	…	…	…	…	…
	Value[255][0]	0x4007F8	4	R/W	Sets the 255th X coordinate of defective pixel.
	Value[255][1]	0x4007FC	4	R/W	Sets the 255th Y coordinate of defective pixel.

D4288264C

● Control with TeliCamSDK

GenICam function API

Control DPC using GenICam API.

◆ DPC

1. Set the coordinates of defective pixels by 'DPCIndex', 'DPCEntryX' and 'DPCEntryY'.

2. Sets the number of pixels to correct to 'DPCNumber'.

3. Sets the activation of DPC function by 'DPCEnable'.

Integer value and string value of Enumeration are as follows.

Integer	String
0	Off
1	On

```
// GenICam node handle
CAM_NODE_HANDLE   hIndex = NULL;
CAM_NODE_HANDLE   hEntryX = NULL;
CAM_NODE_HANDLE   hEntryY = NULL;
CAM_NODE_HANDLE   hNumber = NULL;
CAM_NODE_HANDLE   hEnable = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "DPCIndex", &hIndex);
Nd_GetNode(s_hCam, "DPCEntryX", &hEntryX);
Nd_GetNode(s_hCam, "DPCEntryY", &hEntryY);
Nd_GetNode(s_hCam, "DPCNumber", &hNumber);
Nd_GetNode(s_hCam, "DPCEnable", &hEnable);

int64_t entry[2][2] = {{100,200},{150,300}};// {x.y}
int64_t number;

for(number=0; number<2; number++)
{
 // 1.Set the coordinates of defective pixels by 'DPCIndex', 'DPCEntryX' and
'DPCEntryY'.
 Nd_SetIntValue(s_hCam, hIndex, number); // 0 origin
 Nd_SetIntValue(s_hCam, hEntryX, entry[number][0]);
 Nd_SetIntValue(s_hCam, hEntryY, entry[number][1]);
}

// 2.Sets the number of pixels to correct to 'DPCNumber'.
Nd_SetIntValue(s_hCam, hNumber, number); // 2 pixels are to be corrected.

// 3.Sets the activation of DPC function by 'DPCEnable'.
Nd_SetEnumStrValue(s_hCam, hEnable, "On");
```

Please refer to [INode functions], [IInteger node functions] and [IEnumeration node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control DPC by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ DPC

1. Set the coordinates of defective pixels.

   Write to 'Value[index][0]' and 'Value[index][1]' field of 'DPCCoord' register.

   The first index of 'Value' register indicates defective pixel number.

   The second index of [0] and [1] correspond to X and Y coordinates.

2. Sets the number of pixels to correct.

   Write to 'Value' field of 'DPCNumber' register.

3. Sets the activation of DPC function.

   Write [1] to 'Value' field of 'DPCEnable' register.

```
// 1.Set the coordinates of defective pixels.
uint32_t    entry[2][2] = {{100,200},{150,300}};          // {x.y}
uint32_t    number;
uint64_t    addr = 0x400000;
for(number=0; number<2; number++, addr+= 0x8)
{
          Cam_WriteReg(s_hCam, addr      , 1, &entry[number][0]);
          Cam_WriteReg(s_hCam, addr+0x4, 1, &entry[number][1]);
}

// 2.Sets the number of pixels to correct.
Cam_WriteReg(s_hCam, 0x21F2BC, 1, &number);

// 3.Set the activation of DPC function.
uint32_t dat = 1;
Cam_WriteReg(s_hCam, 0x21F29C, 1, &dat);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

◆ Minimum / Maximum Value

	DPCNumber	DPCIndex	DPCEntryX	DPCEntryY
Minimum	0	0	0	0
Maximum	256	255	WidthMax-1	HeightMax-1

UserSetControl

You are able to save a user setting to the non-volatile or volatile memory of the camera.

There are 15 user memory channels for user setting.

By using user memory, you are able to restore frequent used settings at the time of next start-up.

The following table is the list of registers applied to "UserSetLoad"/"UserSetSave","UserSetQuickSave".

List of registers to be applied to UserSet

Category	Register	Category	Register
ImageFormatControl	ImageFormatSelector	DigitalIOControl	AntiGlitch
	Width		AntiChattering
	Height	TimerControl	TimerTriggerSource
	OffsetX		TimerDuration
	OffsetY		TimerDelay
	Binning	AnalogControl	Gain
	Decimation		GainAuto
	Reverse		BlackLevel
	PixelFormat		Gamma
	TestPattern		Sharpness
AcquisitionControl	AcquisitionFrameRateEnable	ALCControl	ALCExposureValue
	AcquisitionFrameRate		ALCPhotometricAreaSize
	AcquisitionFrameIntervalControl	LUTControl	LUTEnable
	AcquisitionFrameInterval	SequentialShutterControl	SequentialShutterEnable(*)
TriggerControl	TriggerMode		SequentialShutterTerminateAt(*)
	TriggerSequence		SequentialShutterEntry(*)
	TriggerSource	DPCControl	DPCEnable(*)
	TriggerAdditionalParameter		DPCNumber(*)
	TriggerDelay		DPCEntryX(*)
ExposureControl	ExposureTime		DPCEntryY(*)
	ExposureAuto	EventControl	EventNotification
	ShortExposureMode		
DigitalIOControl	LineModeAll		
	LineInverterAll		
	UserOutputValueAll		
	LineSelector		
	LineSource		

(*) DPC and SequentialShutter entries are stored to a single channel. Entries are shared with all channels.

D4288264C

https://www.toshiba-teli.co.jp/en/

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
UserSetSelector	IEnumeration	4	R/W	Selects a channel of user setting.
UserSetLoad	ICommand	4	W	Loads a user setting.
UserSetSave	ICommand	4	W	Saves a user setting to non-volatile memory.
UserSetQuickSave	ICommand	4	W	Saves a user setting to volatile memory.
UserSetDefault	IEnumeration	4	R/W	Selects a channel of user setting when camera powers up.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
UserSetSelector	Implemented	0x208060	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20806C	4	R	[0] : Default [1] : UserSet1 … [15] : UserSet15
	Value	0x20807C	4	R/W	Selects the UserSet.
UserSetCommand	Implemented	0x208080	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x20808C	4	R	[0] : Done [8] : Load [9] : Save [120] : QuickSave
	Value	0x20809C	4	R/W	Selects the UserSet command.
UserSetDefault	Implemented	0x21F2E0	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x21F2EC	4	R	[0] : Default [1] : UserSet1 … [15] : UserSet15
	Value	0x21F2FC	4	R/W	Selects the number of UserSet when the camera starts up.

D4288264C
https://www.toshiba-teli.co.jp/en/

● Control with TeliCamSDK

Camera feature API

Control UserSetControl using dedicated API.

API name	Description
ExecuteCamUserSetLoad	Execute UserSetLoad
ExecuteCamUserSetSave	Execute UserSetSave
ExecuteCamUserSetSaveAndSetDefault	Execute UserSetDefault

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

GenICam function API

Control UserSetControl using GenICam API.

◆ **UserSetLoad/UserSetSave/UserSetQuickSave**

1. Select a channel of user setting by 'UserSetSelector'.

   Integer value and string value of Enumeration are as follows.

Integer	String	Description	Save	Load
0	Default	Initial factory setting.	-	○
1	UserSet1	Memory channel 1 for user setting. - DPC feature is enabled.	○	○
2~15	UserSet2~15	Memory channel 2 to 15 for user setting.	○	○

2. Execute UserSetLoad, UserSetSave or UserSetQuickSave

   When execute 'UserSetLoad', the camera loads user setting from the channel selected in 'UserSetSelector' register and applies them.

   When execute 'UserSetSave' or 'UserSetQuickSave', the camera saves user setting to the channel selected in 'UserSetSelector' register.

   After execution of 'UserSetSave', wait until 'UserSetSave' completion by Nd_GetCmdIsDone.

D4288264C
https://www.toshiba-teli.co.jp/en/

```
// GenICam node handle
CAM_NODE_HANDLE   hSelector = NULL;
CAM_NODE_HANDLE   hSave = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "UserSetSelector", &hSelector);
Nd_GetNode(s_hCam, "UserSetSave", &hSave);

// 1.Select a channel of user setting by 'UserSetSelector'.
uint64_t dat = 1; // UserSet1
Nd_SetEnumIntValue(s_hCam, hSelector, dat);

// 2. Execute UserSetSave
Nd_CmdExecute(s_hCam, hSave);

bool8_t bDone;
while(1) {
  Nd_GetCmdIsDone(s_hCam, hSave, &bDone);
  if(bDone == true)
    break;
  Sleep(0);
}
```

◆ **UserSetDefault**

Select a channel of user setting when camera powers up by 'UserSetDefault'.

Integer value and string value of Enumeration are as follows.

Integer	String	Description
0	Default	Initial factory setting.
1~15	UserSet1~15	Memory channel 1 to 15 for user setting.

```
// GenICam node handle
CAM_NODE_HANDLE   hSelector = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "UserSetDefault", &hSelector);

// 1.Select a channel of user setting when camera powers up by 'UserSetDefault'.
uint64_t dat = 1; // UserSet1
Nd_SetEnumIntValue(s_hCam, hSelector, dat);
```

Please refer to [INode functions], [IEnumeration node functions] and [ICommand node functions] in [TeliCamAPI Library manual] for more detail.

D4288264C

https://www.toshiba-teli.co.jp/en/

Register access API

Control UserSetControl by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **UserSetLoad/UserSetSave/UserSetQuickSave**

1. Select a channel of user setting by 'UserSetSelector'.

   Write to 'Value' field of 'UserSetSelector' register.

2. Execute UserSetLoad, UserSetSave or UserSetQuickSave

   Write [8] to 'Value' field of 'UserSetCommand' register to execute UserSetLoad.

   Write [9] to 'Value' field of 'UserSetCommand' register to execute UserSetSave.

   Write [120] to 'Value' field of 'UserSetCommand' register to execute UserSetQuickSave.

   After execution of 'UserSetSave', wait until 'UserSetCommand' is 'Done' (= 0).

```
uint32_t dat;

// 1.Select a channel of user setting by 'UserSetSelector'.
dat = 1; // UserSet1
Cam_WriteReg(s_hCam, 0x20807C, 1, &dat);

// 2. Execute UserSetSave

dat = 9; // UserSetSave

Cam_WriteReg(s_hCam, 0x20809C, 1, &dat);

while(1) {
 Cam_ReadReg(s_hCam, 0x20809C, 1, &dat);
 if(dat == 0)
   break;
 Sleep(0);
}
```

◆ **UserSetDefault**

Select a channel of user setting when camera powers up by 'UserSetDefault'.

```
// 1.Select a channel of user setting when camera powers up
uint32_t dat = 1; // UserSet1
Cam_WriteReg(s_hCam, 0x2080BC, 1, &dat);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

EventControl

Camera notifies FrameTrigger status and other information by USB3 Vision Event Packet.

- FrameTrigger   : Reception of Frame Start Trigger
- FrameTriggerError   : Rejection of Frame Start Trigger
- FrameTriggerWait   : Start of waiting for Frame Start Trigger
- FrameTransferStart   : Start of transferring streaming data
- FrameTransferEnd   : End of transferring streaming data
- ExposureStart   : Start of Exposure
- ExposureEnd   : End of Exposure
- Timer0Start   : Start of Timer0
- Timer0End   : End of Timer0

Events timing are as following chart.
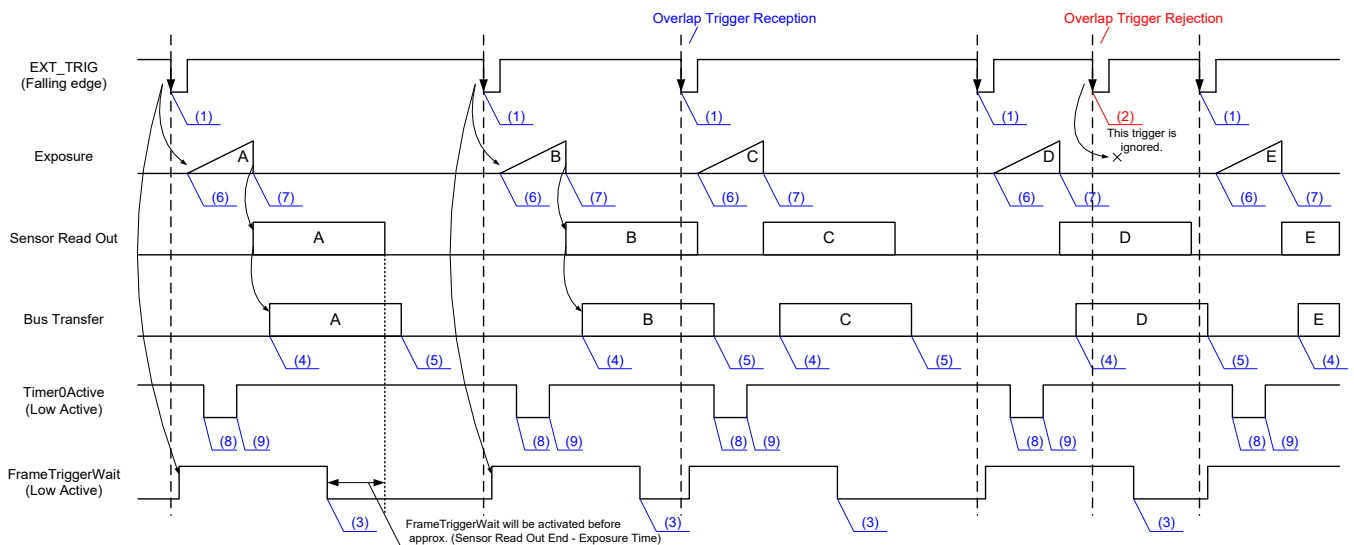
Event Name

(1) FrameTrigger   : Reception of Frame Start Trigger.
(2) FrameTriggerError   : Rejection of Frame Start Trigger.
(3) FrameTriggerWait   : Start of waiting for Frame Start Trigger.
(4) FrameTransferStart : Start of transferring streaming data.
(5) FrameTransferEnd   : End of transferring streaming data.
(6) ExposureStart   : Start of Exposure.
(7) ExposureEnd   : End of Exposure.
(8) Timer0Start   : Start of Timer0.
(9) Timer0End   : End of Timer0.

D4288264C

https://www.toshiba-teli.co.jp/en/

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
EventSelector	IEnumeration	4	R/W	Selects which Event to signal to the host application.
EventNotification	IEnumeration	4	R/W	Sets the activation of event notification

Event Name	Event ID	Event Data	Length Byte / [bit]	Description
EventFrameTrigger	0x9020	EventFrameTriggerTimestamp	8	Returns the timestamp at the time of Event.
EventFrameTriggeError	0x9021	EventFrameTriggerErrorTimestamp	8	Returns the timestamp at the time of Event.
EventFrameTriggeWait	0x9022	EventFrameTriggerWaitTimestamp	8	Returns the timestamp at the time of Event.
EventFrameTransferStart	0x9030	EventFrameTransferStartTimestamp	8	Returns the timestamp at the time of Event.
EventFrameTransferEnd	0x9031	EventFrameTransferEndTimestamp	8	Returns the timestamp at the time of Event.
EventExposureStart	0x9040	EventExposureStartTimestamp	8	Returns the timestamp at the time of Event.
EventExposureEnd	0x9041	EventExposureEndTimestamp	8	Returns the timestamp at the time of Event.
EventTimer0Start	0xA000	EventTimer0StartTimestamp	8	Returns the timestamp at the time of Event.
EventTimer0End	0xA100	EventTimer0EndTimestamp	8	Returns the timestamp at the time of Event.

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
EventNotificationOfFrame	Implemented	0x21F220	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x21F22C	4	R	[0] : FrameTrigger [1] : FrameTriggerError [2] : FrameTriggerWait [3] : FrameStart [4] : FrameEnd [16] : FrameTransferStart [17] : FrameTransferEnd
	Value	0x21F230	4	R/W	Sets the activation of event notification of frame.
EventNotificationOfExposure	Implemented	0x21F240	[31]	R	Returns the state whether the function is implemented.
	ListOfElements	0x21F24C	4	R	[0] : ExposureStart [1] : ExposureEnd
	Value	0x21F250	4	R/W	Selects the UserSet command.
EventNotificationOfTimerStart	Implemented	0x21F380	[31]	R	Sets the activation of event notification of exposure.
	ListOfElements	0x21F38C	4	R	[0] : Timer0Start
	Value	0x21F390	4	R/W	Sets the activation of event notification of timer start.
EventNotificationOfTimerEnd	Implemented	0x21F3A0	[31]	R	Sets the activation of event notification of exposure.
	ListOfElements	0x21F3AC	4	R	[0] : Timer0End
	Value	0x21F3B0	4	R/W	Sets the activation of event notification of timer end.

D4288264C

https://www.toshiba-teli.co.jp/en/

● **Control with TeliCamSDK**

Camera feature API

Control Event features using dedicated API.

API name		Description
High-level API functions	Evt_OpenSimple	Opens event interface
	Evt_Activate	Activates specified camera event.
	Evt_Deactivate	Deactivates specified camera event.
Low-level API functions	Evt_Open	Opens event interface
	Evt_CreateRequest	Creates EventRequest structure
	Evt_ReleaseRequest	Releases EventRequest structure
	Evt_EnqueueRequest	Puts specified EventRequest into the EventWaitQueue
	Evt_DequeueRequest	Retrieves an EventRequest structure from the EventCompleteQueue
	Evt_FlushWaitQueue	Stops all CameraEvent receiving operations, and moves all EventRequest structures in the EventWaitQueue to the EventCompleteQueue
Common functions	Evt_Close	Closes event interface

Please refer to [Camera event notification functions] in [TeliCamAPI Library manual] and [GrabEvent] sample code in your TeliCamSDK install folder.

GenICam function API

Please use dedicated API to control Event features.

Register access API

Please use dedicated API to control Event features.

D4288264C
https://www.toshiba-teli.co.jp/en/

Chunk

This section describes Chunk feature. Chunk data means tagged blocks of data.

The tags allow a chunk parser to dissect the data payload into its elements and to identify the content.

The length of a frame varies depending on the number of activated chunks.

Chunk

Chunk Data Structure

● GenICam Node

Name	Interface	Length Byte / [bit]	Access	Description
ChunkModeActive	IBoolean	4	R/W	Sets the activation of Chunk function.
ChunkSelector	IEnumeration	4	R/W	Select which Chunk to enable or control.
ChunkEnable	IBoolean	4	R/W	Enables the inclusion of the selected Chunk data in the payload of the image.
ChunkUserAreaLength	IInteger	4	R	Indicates the length of ChunkUserAreaTable.
ChunkUserAreaTable	IString	256	R/W	Set the free user space.(Maximum:256byte)
ChunkFrameID	IInteger	8	R	Returns the Block ID of the frame included in the Chunk data.
ChunkExposureTime	IFloat	4	R	Returns the ExposureTime of the frame included in the Chunk data.
ChunkGain	IFloat	4	R	Returns the Gain of the frame included in the Chunk data.
ChunkWhiteBalanceR	IFloat	4	R	Returns the WhiteBalanceR of the frame included in the Chunk data.
ChunkWhiteBalanceB	IFloat	4	R	Returns the WhiteBalanceB of the frame included in the Chunk data.
ChunkLineStatusAll	IInteger	4	R	Returns the LineStatusAll of the frame included in the Chunk data.
ChunkFrameBurstTriggerCount	IInteger	4	R	Returns the FrameBurstTriggerCount of the frame included in the Chunk data.
ChunkSequentialShutterNumber	IInteger	4	R	Returns the SequentialShutterNumber of the frame included in the Chunk data.
ChunkSequentialShutterElement	IInteger	4	R	Returns the SequentialShutterElement of the frame included in the Chunk data.

D4288264C

https://www.toshiba-teli.co.jp/en/

● IIDC2 Register

Register	Field	Address	Length Byte / [bit]	Access	Description
ChunkModeActive	Implemented	0x21D020	[31]	R	Returns the state whether the function is implemented.
	BitWritable	0x21D02C	4	R	[0]:Off [1]:On
	Value	0x21D030	4	R/W	Sets the activation of chunk feature.
ChunkEnableOfVendorSpecific	Implemented	0x21D040	[31]	R	Returns the state whether the function is implemented.
	BitWritable	0x21D04C	4	R	[0] : BlockID [8] : FrameBurstTriggerCount [9] : SequentialShutterNumber [10] : SequentialShutterElement [24] : UserArea
	Value	0x21D050	4	R/W	Sets the activation of chunk data.
ChunkEnableOfCat4	Implemented	0x21D0E0	[31]	R	Returns the state whether the function is implemented.
	BitWritable	0x21D0EC	4	R	[0] : ExposureTime [2] : Gain
	Value	0x21D0F0	4	R/W	Sets the activation of chunk data.
ChunkEnableOfCat5	Implemented	0x21D100	[31]	R	Returns the state whether the function is implemented.
	BitWritable	0x21D10C	4	R	[2] : WhiteBalaceR [3] : WhiteBalaceB
	Value	0x21D110	4	R/W	Sets the activation of chunk data.
ChunkEnableOfCat9	Implemented	0x21D180	[31]	R	Returns the state whether the function is implemented.
	BitWritable	0x21D18C	4	R	[2] : LineStatusAll
	Value	0x21D190	4	R/W	Sets the activation of chunk data.
ChunkUserArea	Implemented	0x21D7F0	[31]	R	Returns the state whether the function is implemented.
	NumberOfElements	0x21D7FC	[30:0]	R	Returns the length (in bytes) of ChunkUserArea Value.
	Value[0]	0x21D800	length	R/W	Sets the ASCII string of ChunkUserArea.
	…	…	…		
	Value[last]	0x21D800 +NumberOf Elements -4		R/W	Sets the ASCII string of ChunkUserArea.

D4288264C

https://www.toshiba-teli.co.jp/en/

● Control with TeliCamSDK

Camera feature API

Control ExposureTime using dedicated API.

API name	Description
GetCamChunkModeActive	Get current statusof chunk data output mode
SetCamChunkModeActive	Set chunk data output mode

Please refer to [Controlling camera feature functions] in [TeliCamAPI Library manual] for more detail.

● Control with TeliCamSDK

GenICam function API

Control Chunk using GenICam API.

◆ **Chunk**

1. Close image stream.

2. Sets the activation of Chunk feature by 'ChunkModeActive'.

3. Sets the activation of Chunk data by 'ChunkSelector' and 'ChunkEnable'.

   Most of the Chunk data has been fixed to output.

4. Sets the user string to 'ChunkUserAreaTable'. (optional)

5. Open image stream.

6. Capture image stream.

   TeliCamAPI provides streaming functions including command handling and buffer handling to capture image stream simply.

   Please refer to [Camera streaming functions] in [TeliCamAPI Library manual]

   and [GrabStreamSimple] sample code in your TeliCamSDK install folder.

7. Extract Chunk data using GenICam API.

   7-1. Allocate memory by 'Chunk_AttachedBuffer' .

   7-2. Read Chunk data.

D4288264C

https://www.toshiba-teli.co.jp/en/

```
// GenICam node handle
CAM_NODE_HANDLE    hMode = NULL;
CAM_NODE_HANDLE    hSelector = NULL;
CAM_NODE_HANDLE    hEnable = NULL;
CAM_NODE_HANDLE    hFrameID = NULL;
CAM_NODE_HANDLE    hExposureTime = NULL;
CAM_NODE_HANDLE    hUserAreaTable = NULL;

// Retrieve GenICam node.
Nd_GetNode(s_hCam, "ChunkModeActive", &hMode);
Nd_GetNode(s_hCam, "ChunkSelector", &hSelector);
Nd_GetNode(s_hCam, "ChunkEnable", &hEnable);
Nd_GetNode(s_hCam, "ChunkFrameID", & hFrameID);
Nd_GetNode(s_hCam, "ChunkExposureTime", &hExposureTime);
Nd_GetNode(s_hCam, "ChunkUserAreaTable", &hUserAreaTable);

// 2.Activate ChunkModeActive
Nd_SetBoolValue(s_hCam, hMode, true);

//3.Enable ChunkSelector
Nd_SetEnumStrValue(s_hCam, hSelector, "ExposureTime");
Nd_SetBoolValue(s_hCam, hEnable, true);
Nd_SetEnumStrValue(s_hCam, hSelector, " UserArea");
Nd_SetBoolValue(s_hCam, hEnable, true);


// 4.Set the user string
Nd_SetStrValue(s_hCam, hUserAreaTable, "Test");

// 5.6.Open and capture image
Strm_ReadCurrentImage(hStrm, pvPayloadBuf, &uiPyldSize, &sImageInfo);

// 7-1. Attach Buffer
Chunk_AttachBuffer(s_hStrm, pvPayloadBuf, PyldSize)

// 7-2.Get FrameID of Chunk data.
int64_t fid= 0;
Nd_GetIntValue(s_hCam, hFID, &fid);

// 7-2.Get Exposure Time of Chunk data.
float64_t   exptime = 0;
Nd_GetFloatValue(s_hCam, hExposureTime, &exptime);

// 7-2.Get User Area data of Chunk data.
char userarea[256];
Uint32_t uiSize = 256;
Nd_GetStrValue(s_hCam, hUserAreaTable, &userarea, &uiSize);
```

Please refer to [INode functions], [IInteger node functions], [IBoolean node functions] and [IEnumeration node functions] in [TeliCamAPI Library manual] for more detail.

Register access API

Control Chunk by accessing IIDC2 registers directly.

API name	Description
Cam_ReadReg	Read register value
Cam_WriteReg	Write register value

◆ **Chunk**

1. Close image stream.

2. Sets the activation of Chunk feature.

   Write to 'Value' field of 'ChunkModeActive' register.

3. Sets the activation of Chunk data.

   Write to 'Value' field of 'ChunkEnableOfVendorSpecific', 'ChunkEnableOfCat4',

   'ChunkEnableOfCat5' and 'ChunkEnableOfCat9' register.

4. Sets the user string. (optional)

   Write to 'Value[0]' to 'Value[last]' field of 'ChunkUserArea' register.

5. Open image stream.

6. Capture image stream.

   TeliCamAPI provides streaming functions including command handling and buffer handling to

   capture image stream simply.

   Please refer to [Camera streaming functions] in [TeliCamAPI Library manual]

   and [GrabStreamSimple] sample code in your TeliCamSDK install folder.

7. Extract Chunk data using GenICam API.

```
// 2.Activate ChunkModeActive
int32_t active = 1;
Cam_WriteReg(s_hCam, 0x21D030,1, &active);

// 3.Enable Chunk
int32_t cat4 = 5;
Cam_WriteReg(s_hCam, 0x21D0F0,1, &cat4);

// 5.6.Open and capture image
Strm_ReadCurrentImage(hStrm, pvPayloadBuf, &uiPyldSize, &sImageInfo);

// 7-1. Attach Buffer
Chunk_AttachBuffer(s_hStrm, pvPayloadBuf, PyldSize)

// 7-2.Get FrameID of Chunk data.
int64_t fid= 0;
Nd_GetIntValue(s_hCam, hFID, &fid);

// 7-2.Get Exposure Time of Chunk data.
float64_t  exptime = 0;
Nd_GetFloatValue(s_hCam, hExposureTime, &exptime);

// 7-2.Get User Area data of Chunk data.
char userarea[256];
uint32_t uiSize = 256;
Nd_GetStrValue(s_hCam, hUserAreaTable, &userarea, &uiSize);
```

Please refer to [Camera functions] in [TeliCamAPI Library manual] for more detail.

D4288264C

https://www.toshiba-teli.co.jp/en/

Appendix

UserSetSave and UserSetQuickSave difference

If you want to change multiple Camera Settings (e.g. ROI setting) at once, UserSet feature is useful.

Application needs to save the camera settings in UserSet memory in advance or in initialization stage.

By **UserSetSave**, UserSet is stored to non-volatile flash memory.

It takes about **700ms** to execute UserSetSave because Non-volatile flash memory needs to be erased before writing. If you want to switch over multiple different settings in run time, overhead time of **UserSetSave** is inevitable.
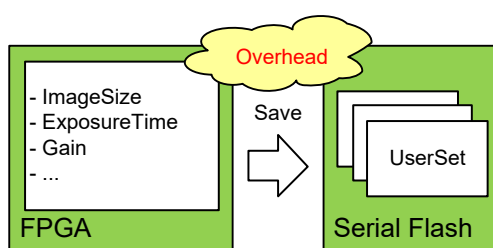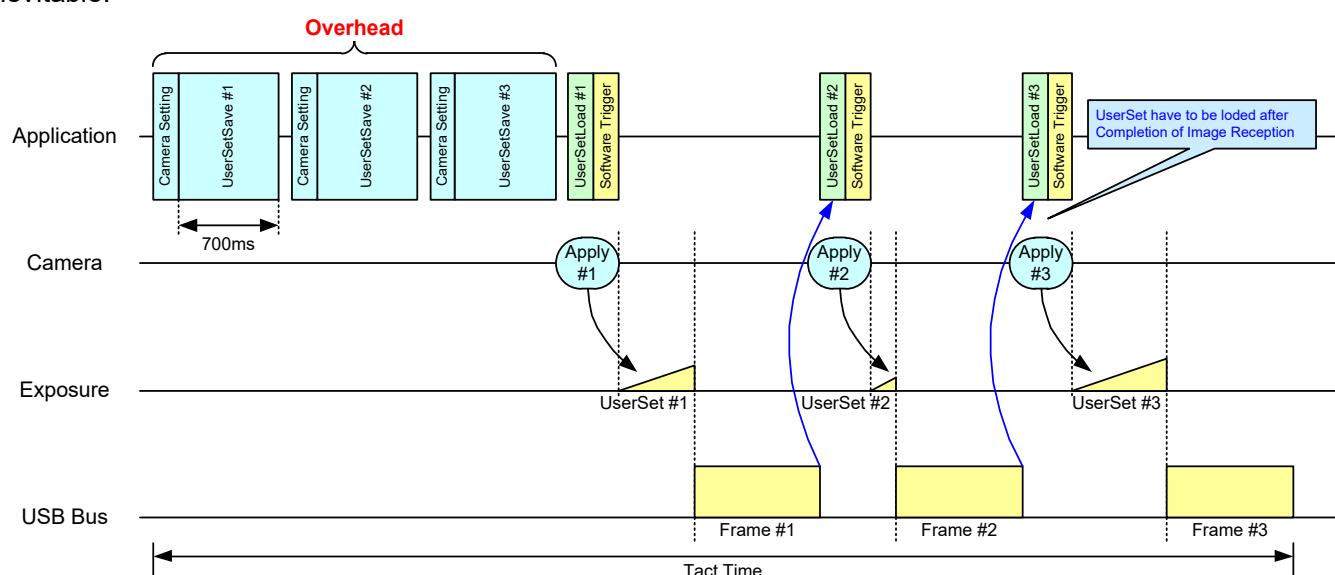
By **UserSetQuickSave**, UserSet is stored to internal RAM. It takes less than **100us** in execution.

It can reduce the overhead time of **UserSetSave** greatly.

You can also save UserSets to Serial Flash if necessary by **UserSetSave**.

D4288264C

https://www.toshiba-teli.co.jp/en/
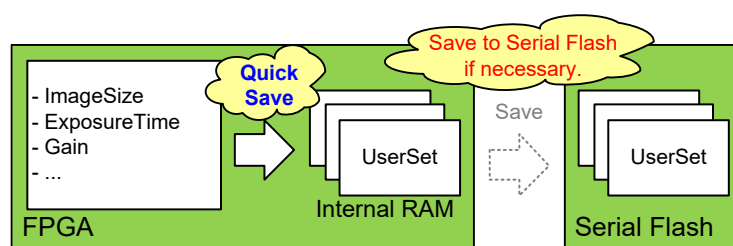
MultiFrame and Bulk function difference

This section describes the difference between MultiFrame function of AcquisitionMode and Bulk function of TriggerSequence.

- MultiFrame function sets limits to the number of frames to transfer with AcquisitionFrameCount register.
- Bulk function sets limits to the number of frames to exposure with TriggerAdditionalParameter register.

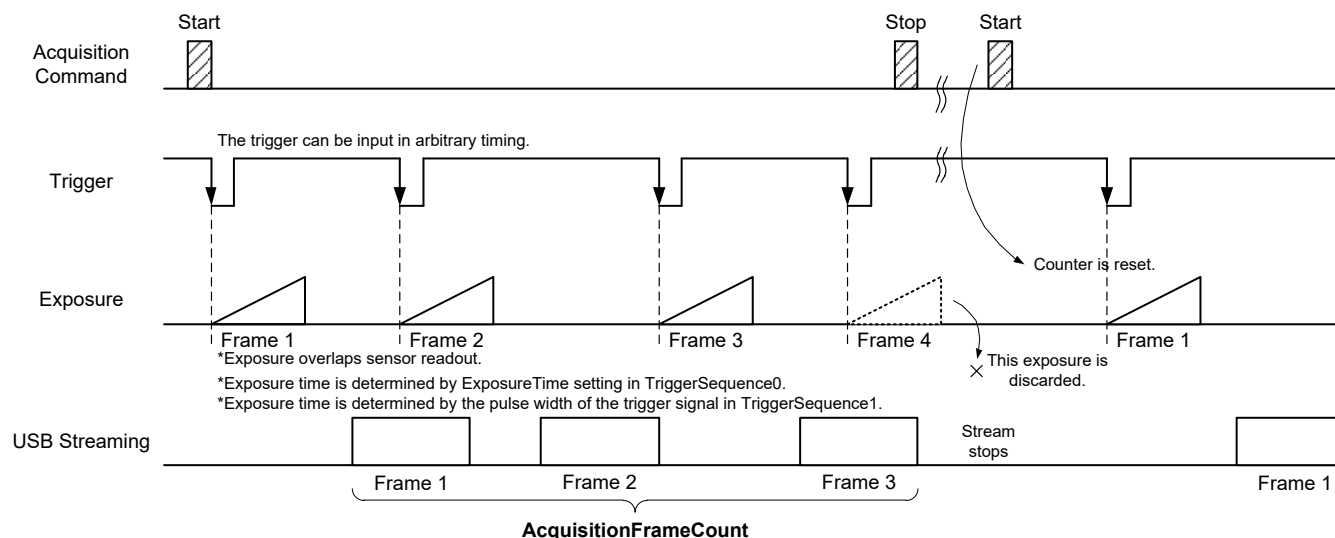**- MultiFrame function in Normal Shutter mode (TriggerMode = Off)**

Camera transfers "AcquisitionFrameCount" frame(s).

D4288264C
https://www.toshiba-teli.co.jp/en/
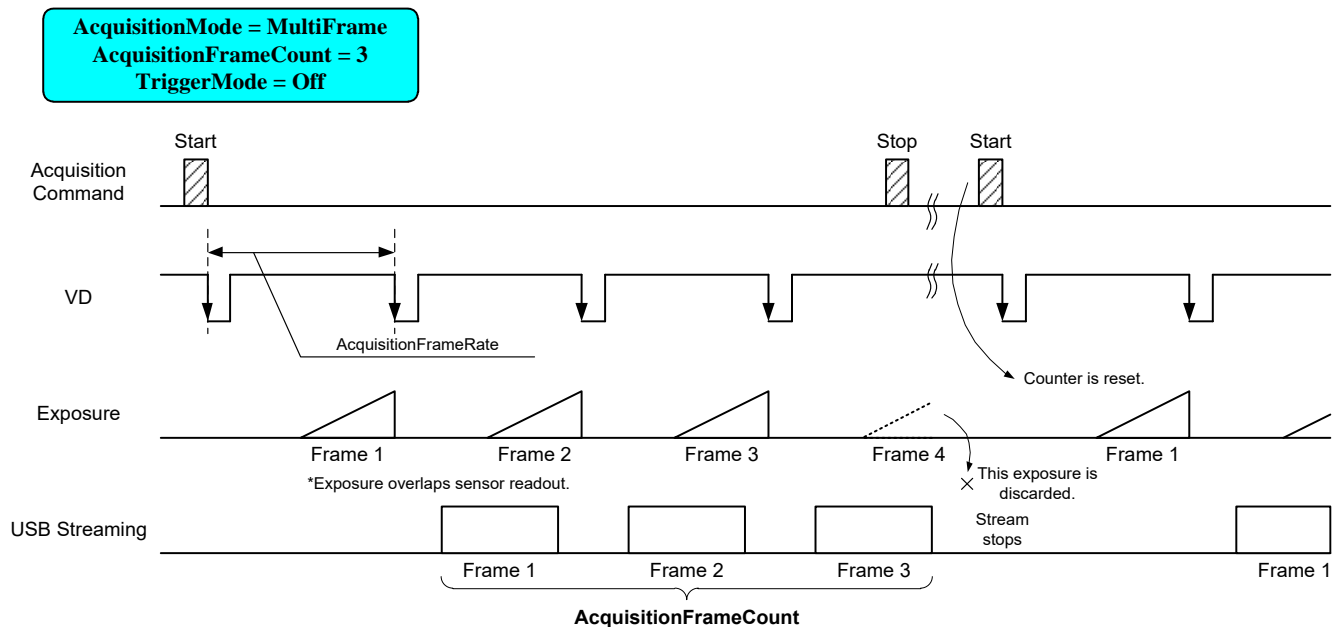
- MultiFrame function in Random Trigger Shutter mode (TriggerMode = On, TriggerSequence = 0 or 1)

Camera transfers "AcquisitionFrameCount" frame(s). It requires "AcquisitionFrameCount" time(s) of trigger.
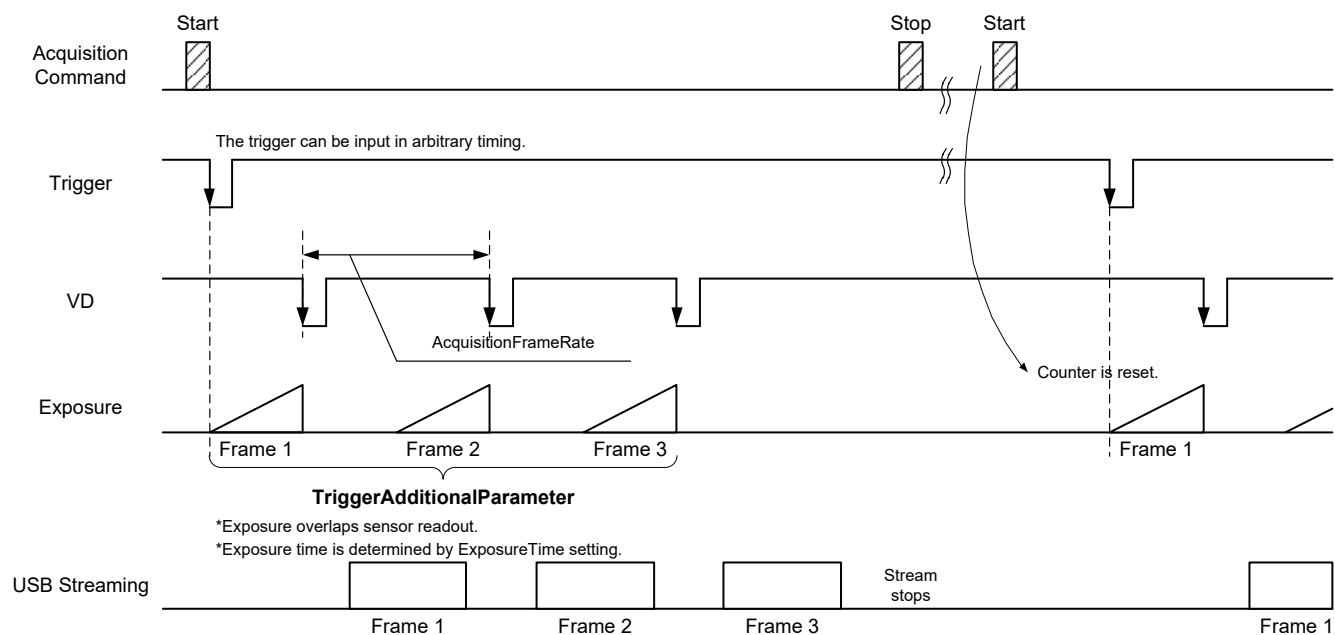

- Bulk function (TriggerMode = On, TriggerSequence = 6)

Camera transfers "TriggerAdditionalParameter" frame(s) by a single trigger.
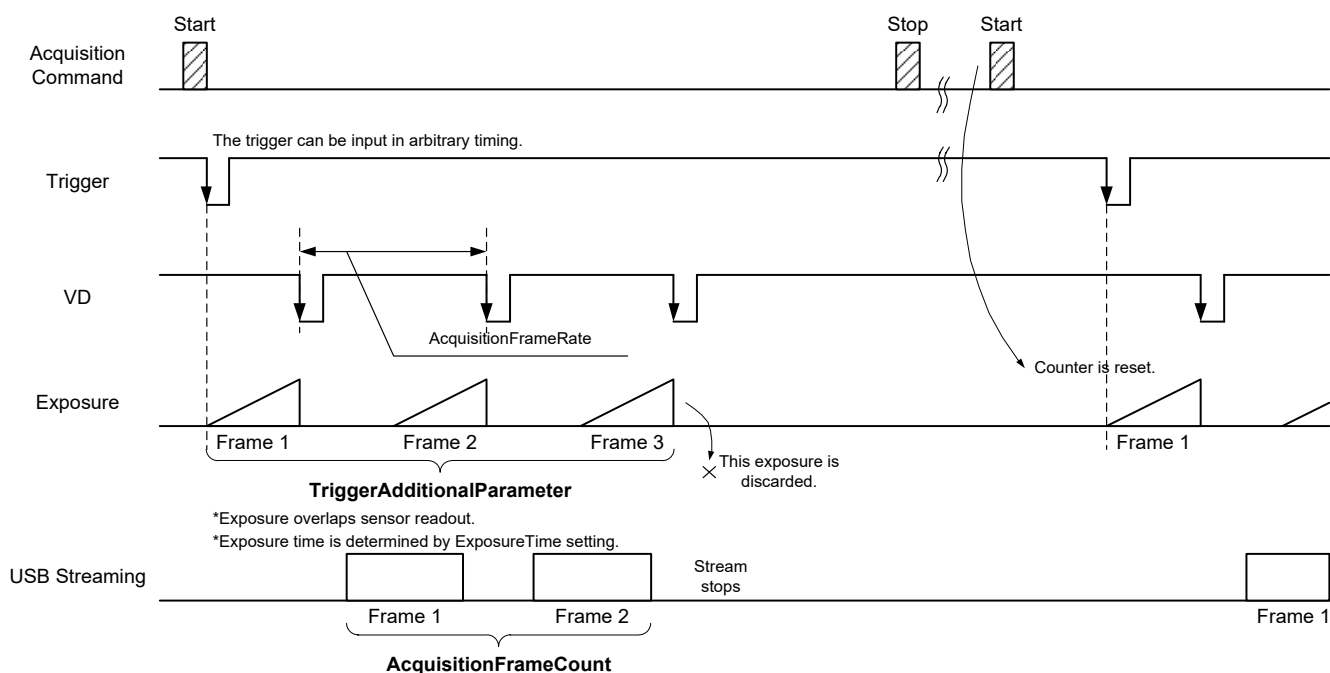
D4288264C

● Note

In Bulk function, you may want to set "Continuous" to "AcquisitionMode" register.

You can set "MultiFrame", but acquisition frame number is limited by "AcquisitionFrameCount.

AcquisitionMode = MultiFrame
AcquisitionFrameCount = 2
TriggerMode = On
TriggerSequence = 6
TriggerAdditionalParameter = 3

D4288264C

https://www.toshiba-teli.co.jp/en/

Warranty rules

● **Warranty term**

   Warranty term is 36 months after your purchase. We may assume the date of the purchase from our shipping date when the date is unidentified.

● **Limited Warranty**

   Free warranty is not applicable for the troubles, damages or losses caused by the cases of the followings, even if it is during the warranty term.

1. Natural exhaust, wear or degradation of a component parts
2. Handling against the instructions and conditions described in the instruction manual
3. Remodeling, adjustment and the part exchange. (including the opening of the enclosure box and the alteration)
4. Using the accessories not included with the product or our non-designated optional articles
5. Damages caused during the transportation or deficiency of the handling such as drop or fall of the products after the products having been transferred to customers, leaving the products to corrosive environment such as sunlight, fire, sand, soil, heat, moisture, or an inappropriate storing method
6. A fire, an earthquake, a flood, a lightning, or other natural disasters, pollution and a short circuit, abnormal voltage, excessive physical pressure, theft, other accident
7. When connected to a product which is not recommended
8. When connected to the power supply which is not suitable
9. Forgery product, products which does not have proper serial number, products of which serial number is forged, damaged or deleted
10. All defects that happened after the expiration for a warranty term

D4288264C

https://www.toshiba-teli.co.jp/en/

Repair

● **Repair methods**

    Exchange to a replacement or an equal function product.

● **Repair request methods**

    On the occasion of a repair request, please download the "Failure situation report sheet" from our website, fill in the necessary items and return it together with the defective product.

    Repair Request Methods

    https://www.toshiba-teli.co.jp/en/support/contact/failure_situation.htm

    Please read the following instructions carefully.

1. Please return our product alone, taking out of your equipment in case that our product is installed to an equipment
2. We are unable to return the information such as your own serial numbers, control number, the identification seal, if it is attached to the returned products. Please keep record before you return the product.
3. As the data saved in the camera will not be kept after the repair, please take out data before return.
4. We are unable to accept the cancellation after the repair request by the customer's reason.
5. About the repair product shipping expenses, please bear the charges when you return the product to us. We bear the charges to you from us only for a warranty period.
6. We are unable to accept your request of a delivery date and time of the product return, or the delivery method.
7. We are unable to accept a trouble factor investigation, the request of the repair report.
8. We accept a repair of out of warranty product, if it is reparable.
9. The proprietary rights of the repair request products after the exchange repair belong to us.
10. The immunity from responsibility of the product is applied in the repair completion products.

    * Please refer for the inquiry about the software to our website or sales personnel.

D4288264C

https://www.toshiba-teli.co.jp/en/