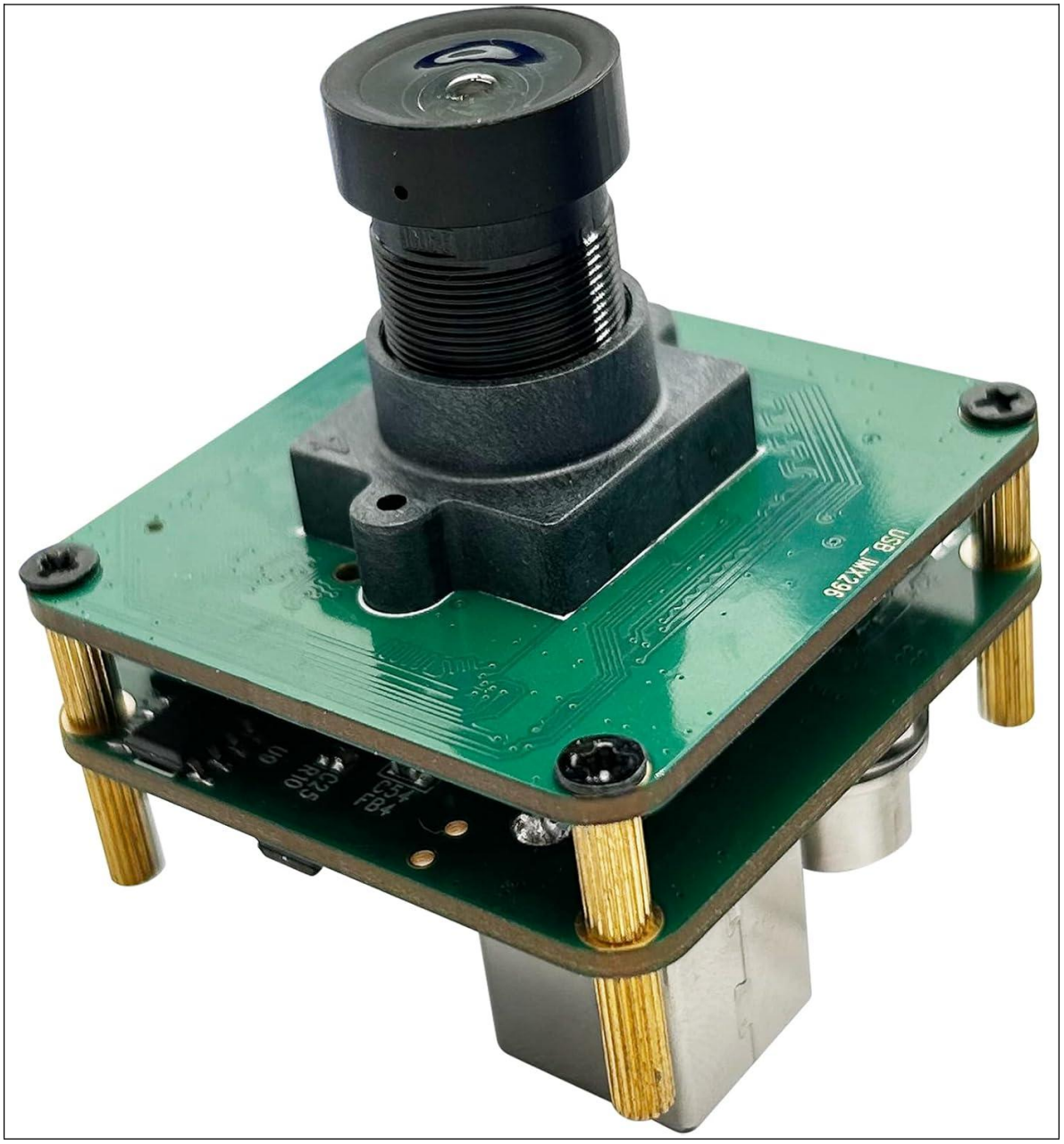
U3V-CAM-IMX296 User Manual



Date	Version	Description
2025/11/24	V1.0	

www.inno-maker.com	www.github.com/inno-maker
sales@inno-maker.com	support@inno-maker.com

1. Product Overview

The U3V-CAM-IMX296 is a high-performance USB3 Vision industrial camera featuring the Sony IMX296LLR (monochrome) global shutter CMOS sensor. With a resolution of 1.58 MP (1456 × 1088) and a full-resolution frame rate of **60 fps** in MONO8 mode, it delivers reliable, distortion-free imaging for demanding machine-vision applications such as motion analysis, automation, robotics, and scientific imaging.

The camera implements the **USB3 Vision protocol completely and without compromise**, offering every standard feature (arbitrary ROI, analog gain, long exposure, opto-isolated I/O, high-precision timestamp, etc.) through standard GenICam nodes — truly plug-and-play across all major software platforms.

Equipped with a true global shutter sensor and full GenICam trigger control, it supports opto-isolated Hardware Trigger (5–24 V) and Software Trigger with Triggersource = Software. The opto-isolated Strobe output,plus 0.1 µs hardware timestamping ensure precise exposure and lighting synchronization, making the camera ideal for motion-triggered inspection, robotic pick-and-place, area-scan emulation, and any application requiring deterministic timing.

1.1 Key Features

- Sony IMX296 1/2.9" global shutter CMOS sensor (diagonal 6.3 mm)
- Resolution: 1456 × 1088 pixels (1.58 MP), pixel size 3.45 µm × 3.45 µm
- Full-resolution 1468x1088 MONO8 @ 60 fps (real sustained USB 3.0 bandwidth)
- Arbitrary ROI (single regions) – frame rate increases inversely with active rows, exceeding 1900 fps in typical reduced-height ROIs
- True analog gain value 0-48d (gainraw 0 – 480), 0.1 dB step(gainraw 1)
- Manual shutter with true long-exposure capability (≥ 16 seconds)
- Hardware & software trigger modes (opto-isolated input)
- Opto-isolated strobe output
- High-precision hardware timestamp (0.1 µs = 100 ns resolution)
- On-board temperature sensor with real-time readout
- “Find Me” LED for instant multi-camera identification
- Device reset via GenICam command
- Globally unique 64-bit serial number (factory-burned)
- 100% compliant with USB3 Vision v1.0 and GenICam 3.x
- Plug-and-play with OpenSource Aravis, eBus Universal Pro(Free License Version),

1.2 Specifications

Sensor & Imaging	
Sensor	Sony IMX296LLR (Monochrome, Global Shutter, Pregius Gen2 process)
Resolution	1456 (H) × 1088 (V) (1.58 MP)
Pixel Size	3.45 µm × 3.45 µm
Optical Format	1/2.9" (diagonal 6.3 mm)
Max Frame Rate (Full ROI)	60 fps 1456 (H) × 1088 (V) (1.58 MP) @ Mono8
Frame Rate (Reduced ROI)	Inversely proportional to active rows; typically, 1900+ fps
Pixel Formats	Mono8 (native), Mono10 (firmware upgradeable)
Exposure Time	29 µs – ≥15 seconds (true long exposure) >=16504 us Enter Long Exposure Mode
Analog Gain	0 – 48 dB (0.1 dB steps, true analog)
Dynamic Range / SNR	~71 dB / ~40 dB
Shutter Type	Global Shutter (zero rolling distortion)
USB3 Vision / GenICam Standard Features	
Protocol Compliance	100 % USB3 Vision v1.0 + GenICam 3.x (official A3 Compliance Test Tool – all green)
Streaming Protocol	U3VSP (leader/payload/trailer, packet resend, auto packet-size negotiation)
Arbitrary ROI	Width, Height, OffsetX/Y, ReverseX/Y, ROIs supported
Acquisition Control	TriggerMode(On/Off) TriggerSoftware(Software Trigger Signal) Trigger Source: (Software:Software Trigger Line1:Hardware Trigger)
Analog & Digital Control	GainSelector(All) GainRaw(0-480)
Device Control	DeviceTemperature (±0.1 °C), DeviceReset, Unique 64-bit Serial Number, Find-Me LED
Electrical & I/O	
Power Supply	USB 3.0 bus-powered, 5 V / ⩽ 3.2 W
Trigger Input (Line 1)	Opto-isolated, 5–24 V, rising/falling edge
Strobe Output	Opto-isolated open-collector, max 100 mA, programmable polarity/delay/duration (0.1 µs step)
Hardware Timestamp	0.1 µs (100 ns) resolution, FPGA-based 10 MHz counter
Mechanical & Environmental	
Dimensions (W × H × D)	

Weight	= 45 g
Lens Mount	M12 Lens **YT10068-2mp** C-Mount (CS-Mount adapter Only)
Housing	N/A
Operating Temperature	–10 °C to +65 °C (sensor temperature monitored & protected)
Storage Temperature	–30 °C to +80 °C
Humidity	20–80 % RH (non-condensing)
Protection Class	
Certifications	
Software & Compatibility	
SDK	Pleora eBus Universal (demo version + eBus Player included) • Windows • Linux Aravis 0.8.35 • Linux
GenICam Interface	100 % standard XML, zero proprietary nodes
Supported OS	Windows 10/11, Linux (Ubuntu 20.04/22.04+), ARM64 (NVIDIA Jetson validated),Raspberry PI,etc…

2 eBus Player Node Map & GenICam Parameters

Connect the camera in eBus Player → Select the device → Click "Guru" mode → Expand "GenICam" tree.

All parameters below are 100 % standard GenICam SFNC nodes – the camera embeds a complete XML description file that eBus Player loads automatically from the device (no local XML file needed).

No.	Key Parameters	eBus Player Path (Guru Mode)	Exact GenICam Node Name(s)	Access	Range / Values (IMX296)	Notes / Customer Benefit
1	Arbitrary ROI – FPS dramatically increases with fewer rows	ImageFormatControl → Region Selector	Width Height OffsetX OffsetY ReverseX ReverseY	RW	Width: 80–1456 (step 4) Height: 4–1088 (step 4)	—
2	True Analog Gain 0–48 （Gainraw 0-480）	AnalogControl → Gain Selector	GainRaw	RW	0 – 48 dB step 0.1 dB （Gainraw 0-480）	Real analog gain, extremely low noise
3	Manual shutter + true long exposure	AcquisitionControl →	ExposureMode = Timed ExposureTime	RW	29 µs – ≥ 15.534s Long Exposure>= 16504 us	Scientific & low-light imaging
4	Real-time temperature measurement	DeviceControl → Device Temperature Selector	DeviceTemperature	RO	±0.1 °C accuracy	
5	“Find Me” LED	DeviceControl →	Find Me	RW	Blink	Instantly locate camera in multi-cam rack
6	Remote Device Reset	DeviceControl →	DeviceReset	Exec	Execute	One-click reboot without unplugging
7	Free Running	Default	Default	Exec	Power On	Plug and Play
8	Hardware Trigger (opto-isolated)	AcquisitionControl → Trigger Mode	TriggerMode = On TriggerSource =	RW	Line1: 5–24 V opto-isolated,	Highest triggered frame rate

		Trigger Source Trigger Activation	Line1 TriggerActivation =(Falling Edge, Rising Edge)		FallingEdge RisingEdge	
9	Software Trigger	AcquisitionControl → Trigger Mode Trigger Source Trigger Software	TriggerMode = On TriggerSource = Software Trigger Software= Software	RW	—	Highest triggered frame rate
10	Opto-isolated Strobe output	AcquisitionControl → TriggerMode	Off(Sensor Output Strobe Signal) On(FPGA)	RW	N/A	Perfect flash synchronization
11	Always On 0.1 µs hardware timestamp	DeviceControl → Device Timestamp Latch	Device Timestamp Latch	RW/RO	0.1 µs resolution (10 MHz counter)	Identify Data timestamp
12	Globally unique serial number	DeviceControl → Device Serial Number	DeviceSerialNumber	RO	Factory-burned ID	Asset management & anti-counterfeiting
13	Native MONO8 output	ImageFormatControl → Pixel Format	PixelFormat	RW	Mono8 (default),	Only ~760 Mbps at full resolution 1456x1088
14	100 % USB3 Vision	Root → All categories	—	—	—	Works with any GenICam software without patches

3 Trigger & Strobe Timing

3.1 Trigger signal on imx296

Parameter List of Global Shutter (Fast Trigger Mode)

Item	Symbol	Min.	Typ.	Max.	Unit
Integration start delay	t_{TGST}	—	—	TBD	µs
Integration end delay	t_{TGED}	—	—	TBD + t_{OFFSET}	µs
Pulse width	t_{TGSE}	TBD	—	—	µs
Next trigger rise / fall prohibited period (All-pixel)	t_{TGPD}	1132	—	—	H
Next trigger rise / fall prohibited period (2 × 2 Vertical FD binning mode)		588			
Next trigger rise / fall prohibited period (ROI)		V_{TR} *1	—	—	
Data output delay (All-pixel / ROI)	t_{TGDLY}	—	15	—	H

*1 V_{TR} = ROIWV1 + 30

Exposure time [s] = (XTRIG low level pulse width [µs]) + 13.73 (TBD)[µs]*1
*1: Exposure time error (t_{OFFSET})

tH =14.815us

3.2 ROI WV1: ROI output height.

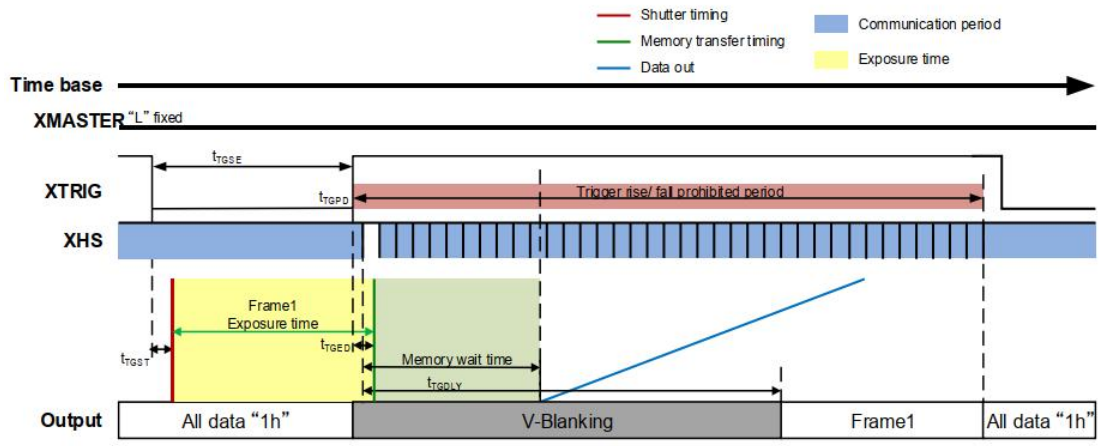
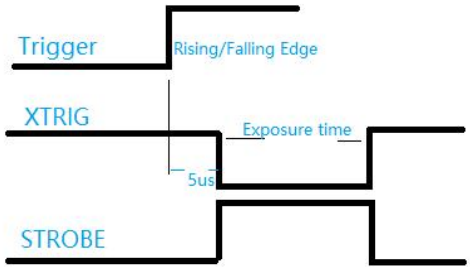


Image Drawing of Global Shutter (Fast Trigger Mode) (4-wire)

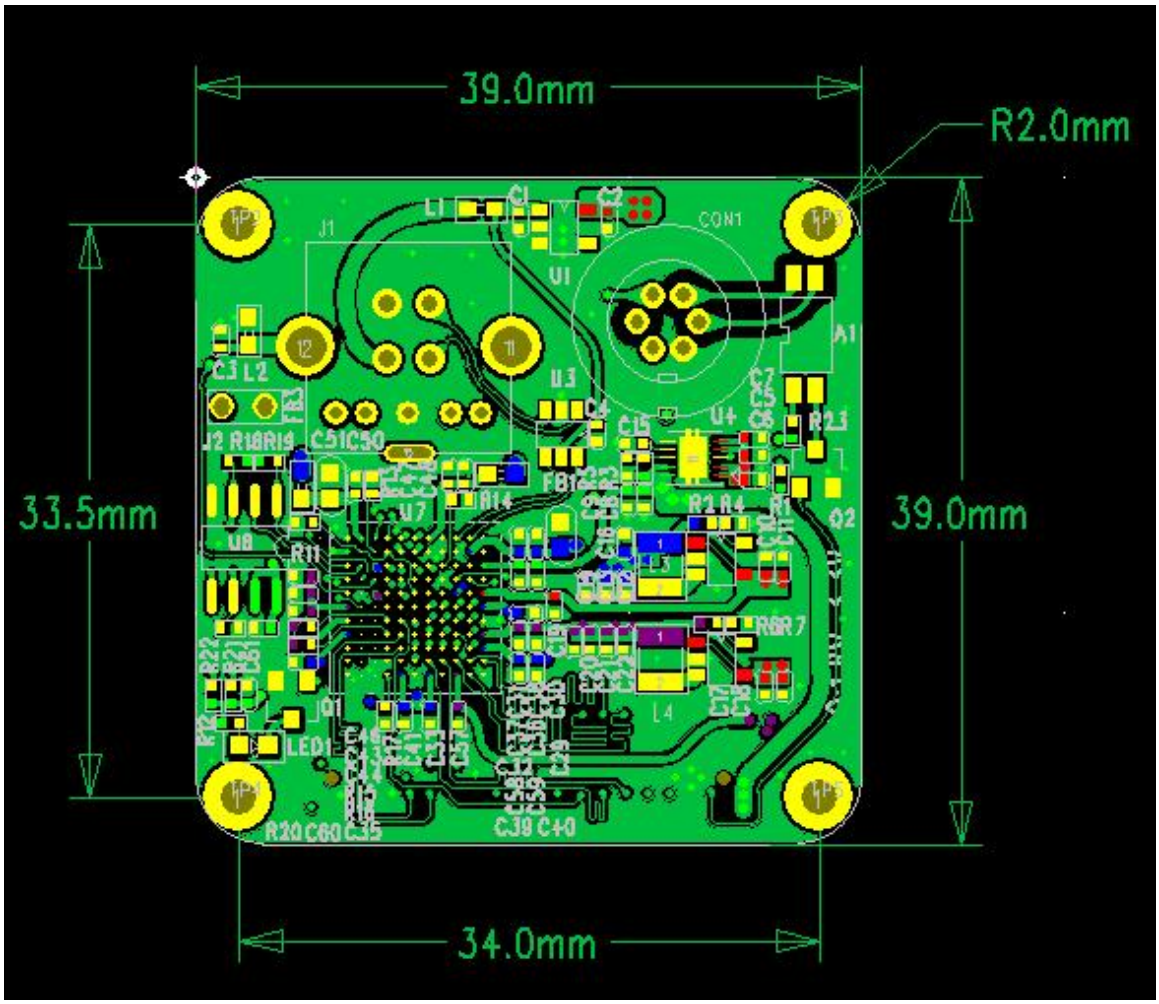
3.3 Trigger and STROBE signals on Device connector:



4 Hardware

4.1 Size information

4.1.1 2D



4.1.2 Mechanical Drawings(TBD)

TBD

4.2 LENS

4.2.1 M12 Lens (Default)

M12 Lens Holder

M12 Lens Part Number

YT10068-2mp 6MM， No IR Filter

M12 Lens Datasheet

NO.	项目(Items)		具体规格（Specification）					
1	F No.		1.0±10%					
2	焦距(Focal-Length)		6.0 ±5%					
3	光学后焦 (Optical Back Focal Length)		4.79±0.2 (in air)					
4	机械后焦 (Mechanical Back Focal Length)		4.11±0.2 (in air)					
5	镜头总长(TTL)		22.20±0.2 (in air)					
6	像面大小（Image circle)		Φ6.8 (MAX)					
7	镜片构成（Lens structure)		3G3P（第一枚镜片为塑胶镜片时，仅可使用酒精擦拭3次） (The first optical lens is plastic lens, could only be wiped with alcohol 3 times)					
8	接口（Mount）		M12*P0.5					
9	镜头与底座螺纹配合扭力 (Clamp Force)		60-600gf-cm					
10	视场角(FOV)	sensor型号	H(水平)		V(垂直)		D(对角)	
		1/2.7"(16:9)	57.2°		31.4°		66.5°	
		1/2.8"(16:9)	55.2°		30.3°		64.1°	
		1/2.9"(16:9)	53.2°		29.2°		61.6°	
11	光学畸变（Optical Distortion)		1/2.7"	-16.4%	1/2.8"	-15.3%	1/2.9"	-14.2%
12	相对亮度（Relative Illumination)		1/2.7"	39%	1/2.8"	41%	1/2.9"	43%
13	最大主光线夹角（CRA)		1/2.7"	14.3°	1/2.8"	13.9°	1/2.9"	13.4°
14	近摄距（M.O.D)		0.2m					
15	解像标准（Resolution)		分辨率（Resolution): 1920*1080 （2MP）					
16	重量 (Weight)		/					
17	操作方法 (Operation)		聚焦 (Focus)		手动 (Manual)			
			光圈 (Iris)		固定 (Fixed)			
18	环保&安全（HSF&Safety)		RoHS					

4.2.2 CS Lens (Options)

CS Lens Holder

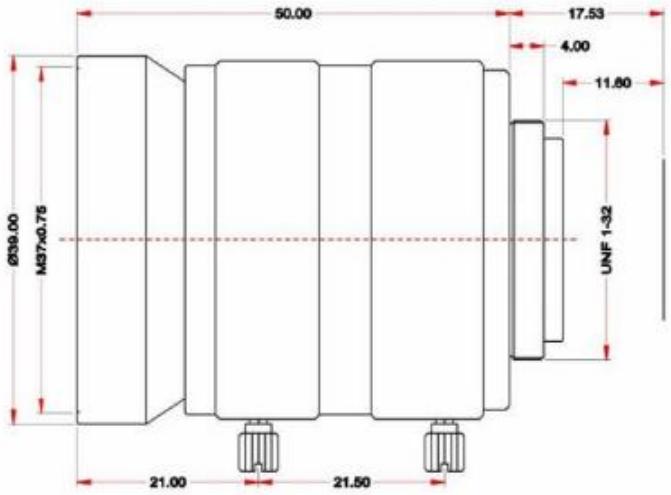
CS Lens Part Number

FA1614/10MP/C

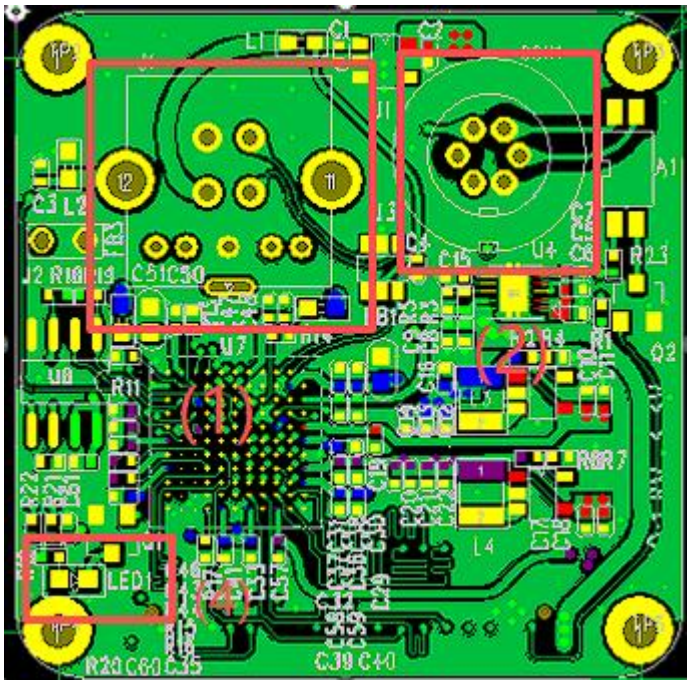
CS LENS Datasheet

- ⊙ 16mm
- ⊙ 1" (Φ16、12.8x9.6)
- ⊙ 10Mega Pixel
- ⊙ C
- ⊙ 手动/锁紧(W/Lock)
- ⊙ IR Correction

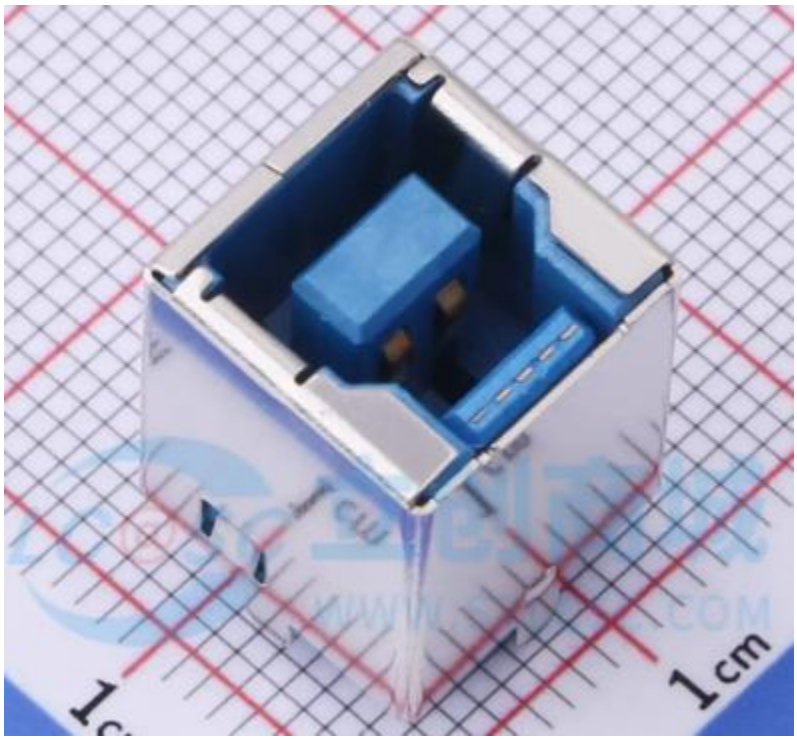
	类 别 (Category)		规 格 参 数(Parameters)
1.	型号 (Model No)		FA1614/10mp/C 蓝
2.	焦距 (Focal Length)		16mm
3.	像面规格 (Format)		1" (Φ16、12.8x9.6)
4.	光圈范围 (D/f′)		F1.4-16
5.	接口 (Mount)		C-Mounts
6.	视场角 (H/V)	1"	44.6°x33.6°
		2/3"	30.0°x23.2°
		1/1.8"	24.7°x18.6°
		1/2"	21.8°x16.4°
7.	后截距 (Back Focal Length)		17.53mm
8.	光学总长 (Optical length)		67.53mm
9.	光圈类型 (Iris type)		手动/锁紧 (W/Lock) M-iris
10.	分辨率 (MTF)		10Mega Pixel
11.	近摄距 (M.O.D) (m)		0.20m
12.	外形尺寸 (Dimension)		Φ39x50.0
13.	重量 (Weight) (g)		133.7g
14.	畸变 (Distortion)		1" (-0.7%) 1/2" (-0.5%) 1/3"(-0.15%)
15.	产品构成 (Structure)		铝合金 (阳极黑) +9G
16.	备注 (Remarks)		UV M37.0x0.75P



4.3 I/O Connector Pinout

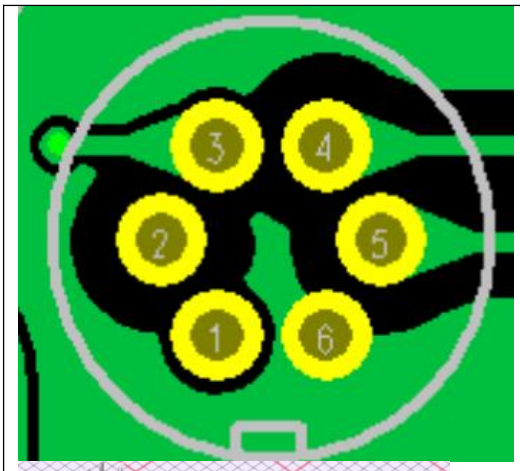


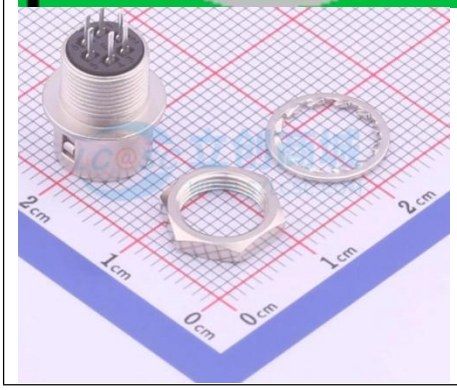
4.3.1 (1) USB3.0 Connector



Part Number: HC-ST-030-10B-L-Z-30-R

4.3.2 (2) Trigger/Strobe/IO Connector





Onboard Connector Part Number:
HR10A-7R-6P

- **HR10A-7R-6PB(73)**
- **HR10A-7R-6P(73)**

PIN1	GPIO_B33,Reserve
PIN2	Trig +
PIN3	GPIO_A33, Reserve
PIN4	STROBE+
PIN5	STROBE- GND/ Trig- GND/
PIN6	GND

4.3.3 (3) Trigger/Strobe/IO Cable

Cable Part Number: 6-pin Hirose Connector Power & Trigger Cable，1M

- Connector: **HR10A-7P-6S**

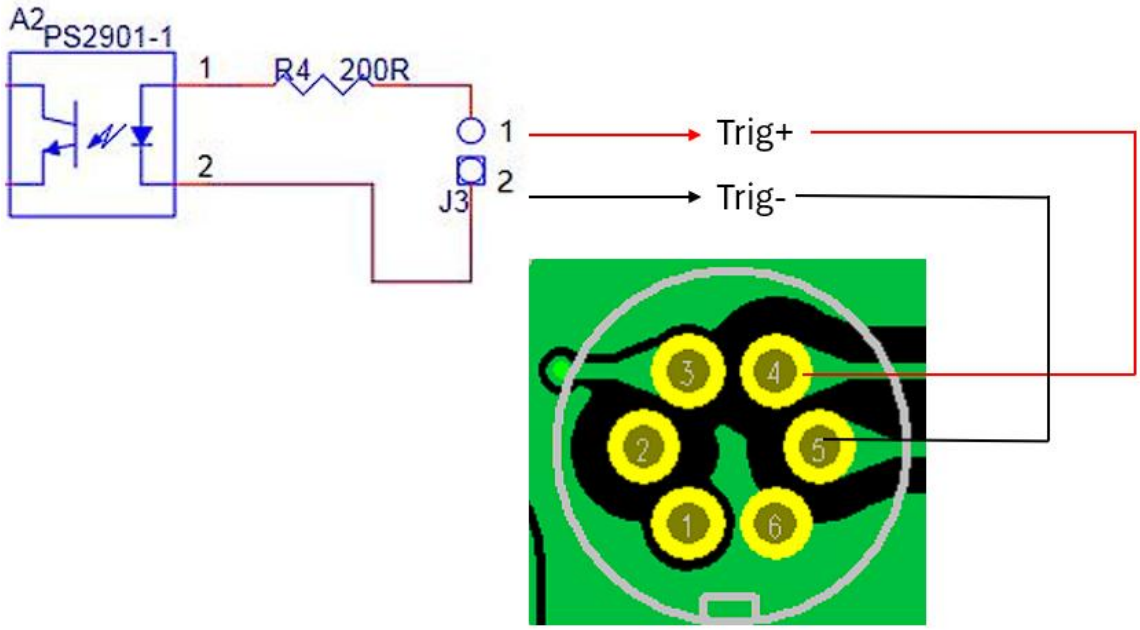
Brown	**Triger +**	**PIN2**
Blue	**STROBE+**	**PIN4**
Yellow	**STROBE-/Triger -/GND_ISO**	**PIN5**

4.3.4 (4)Led

Green	Power On
Flashing	Find Me

4.4 Trigger IN Circuit Diagrams

4.3.1 Trigger Circuit On Board （Internal Usage）



For example, VCC = 12V, Vf = 1.25V

The calculations done here are based on 12VDC. Please do follow these calculations for other voltages like 24VDC.

Let's take the current through IR LED I_f = 20mA.
Voltage drop across the IR LED = 1.25V
The value of Resistor $R_1 = (V_{cc}-V_f)/I_f = (12 – 1.25)/0.02 = 537.5\ \Omega$
Wattage of resistor $R_1 > I_f^2 * R_1 = 0.02^2*537.5 = 0.215W$
Wattage of the resistor R_1 selected should be greater than 0.215W.

And there is a resistor on board(R4 = 200Ω), So the R_add = R1 - R4 = 537.5 - 200 = 337.5Ω

4.3.2 Trigger Circuit Case with Raspberry PI5

We use Raspberry PI5 GPIO23/GND and scripts to generate trigger signal.

Script code is as below, save it to .sh file.

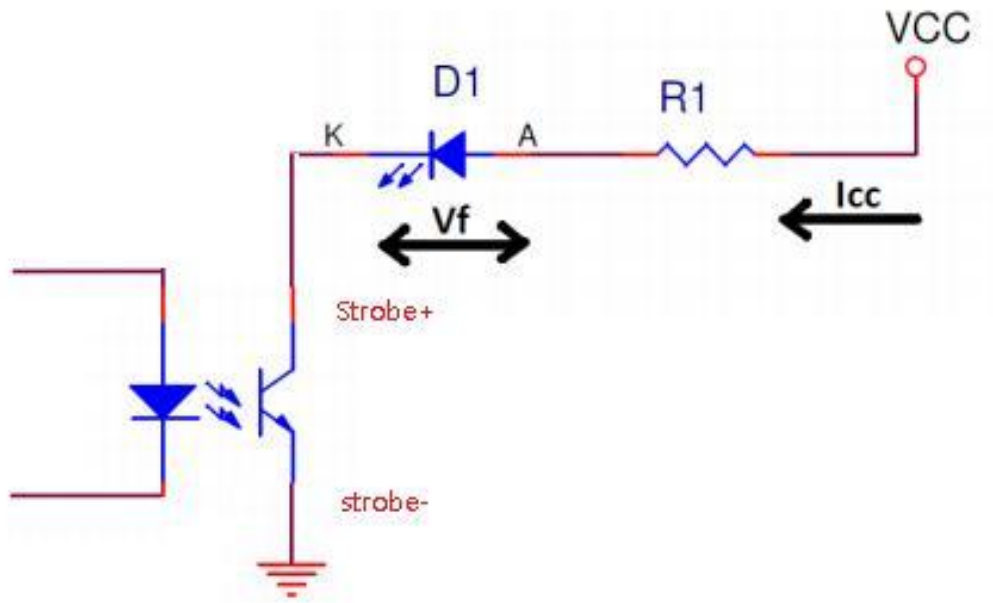
```
while true;do
        gpioset gpiochip0 23=1
        sleep 1.9999
        gpioset gpiochip0 23=0
        sleep 0.0033
done
```

Open another terminal window to run the script

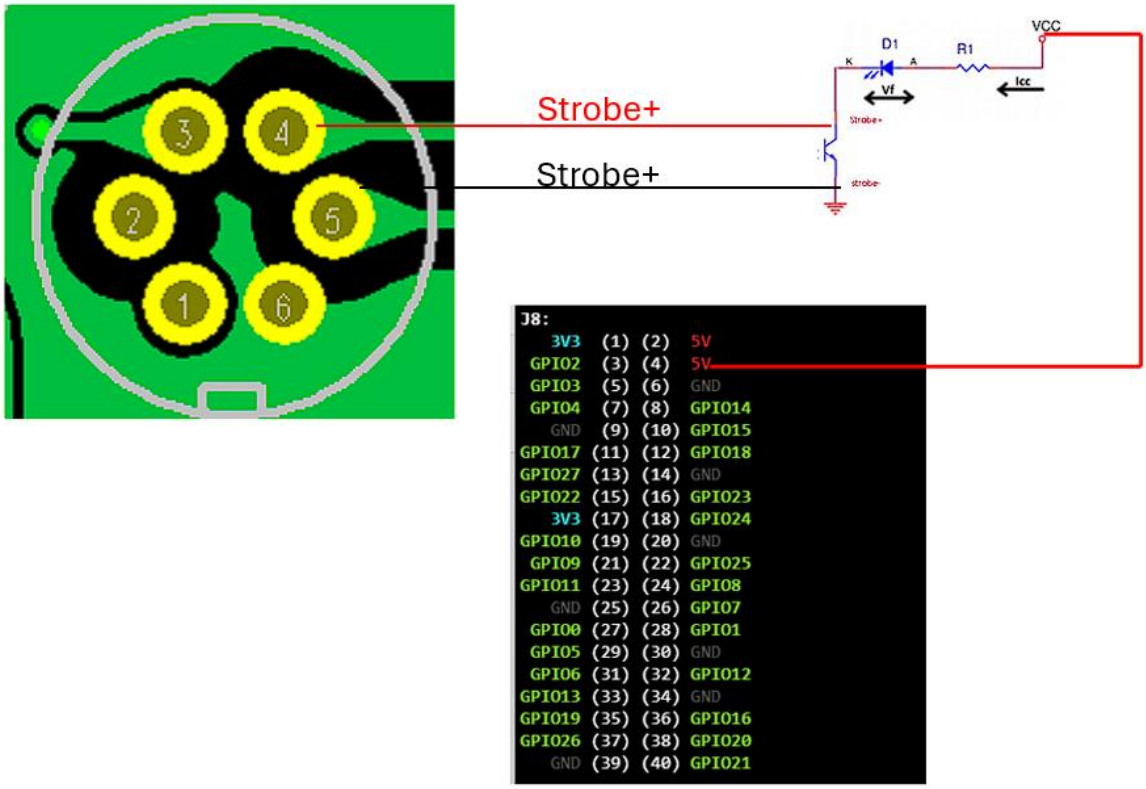
```
pi@raspberrypi:~/cam-imx296raw-trigger $ ./imx296.sh
```

4.5 Strobe Out Circuit Diagrams

4.4.1 Reference Outlet Circuit



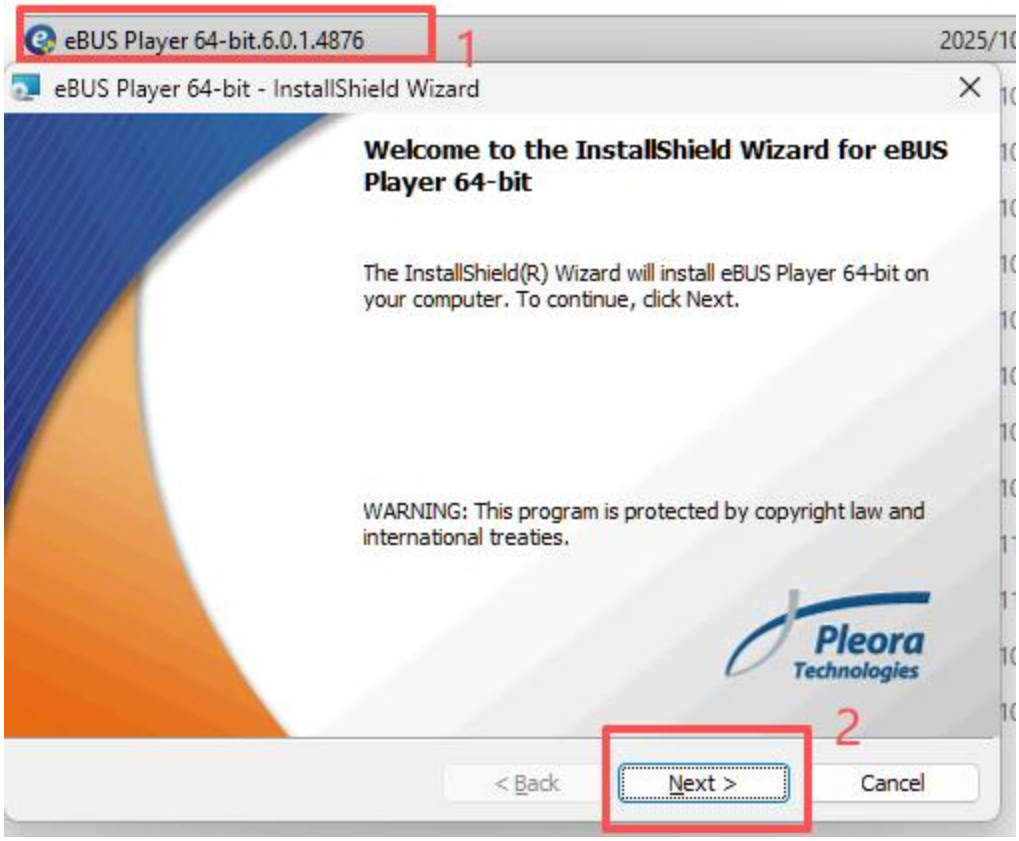
4.4.2 Connection Case with Raspberry PI5



5 Pleora eBus Player

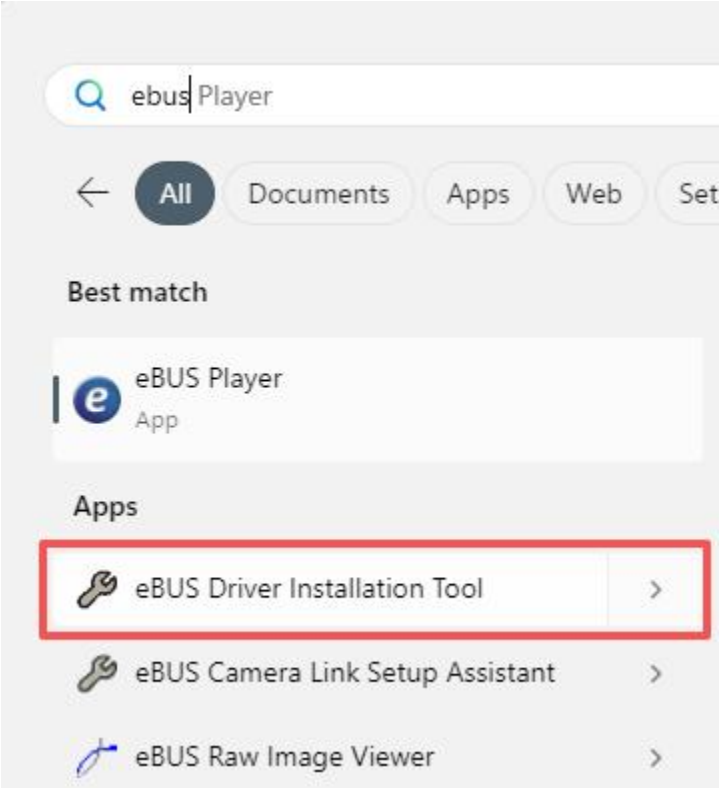
5.1 Installation Guide On Windows

5.1.1 Install ebus_player

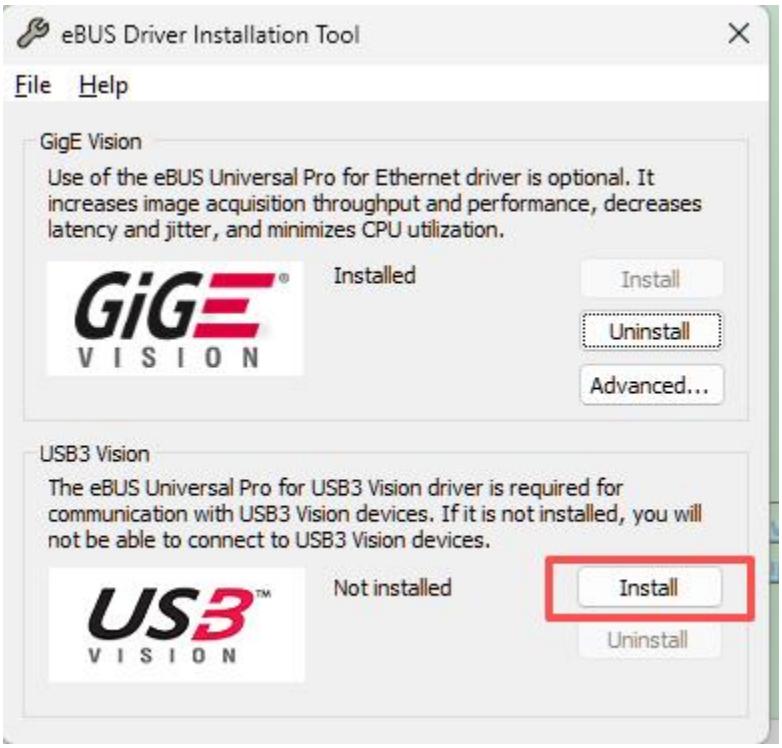


5.1.2 Install Drivers

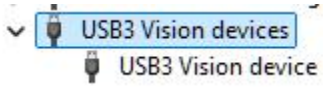
Find eBus Driver Installation Tool from your APP.



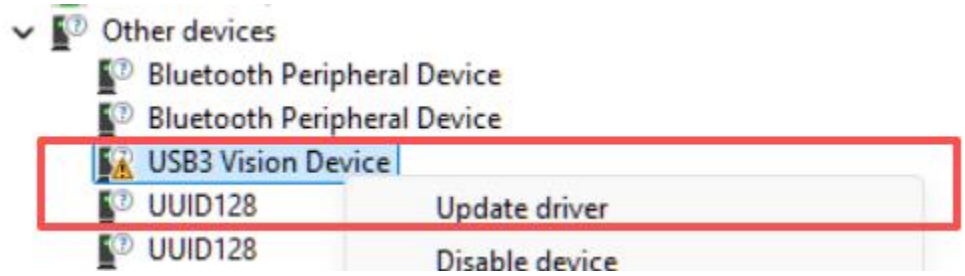
Click Install After below windows shows.

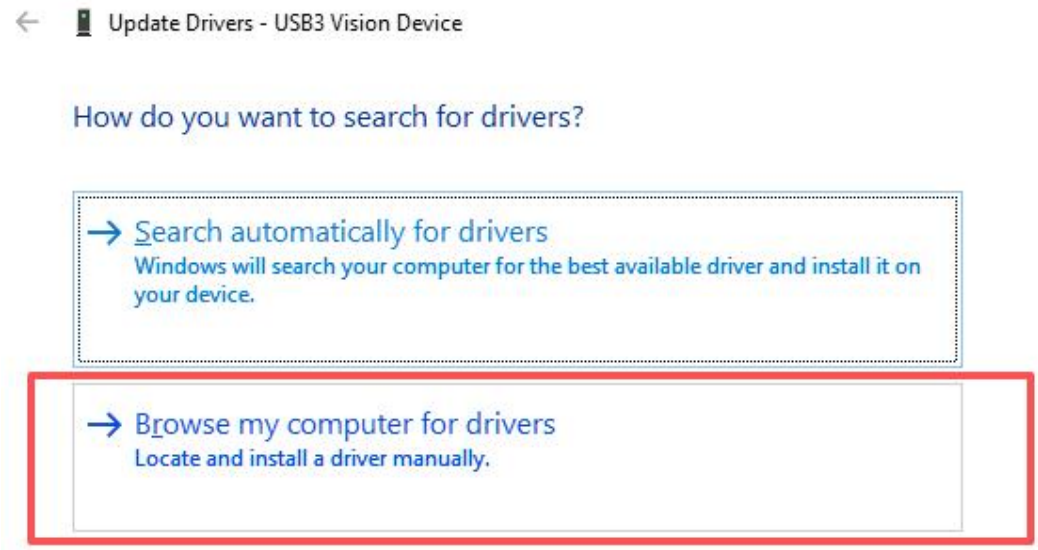


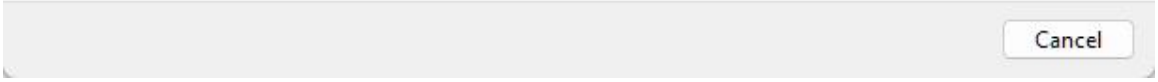
After Driver install successfully, you can find USB3 Vision Devices From your device Manager.

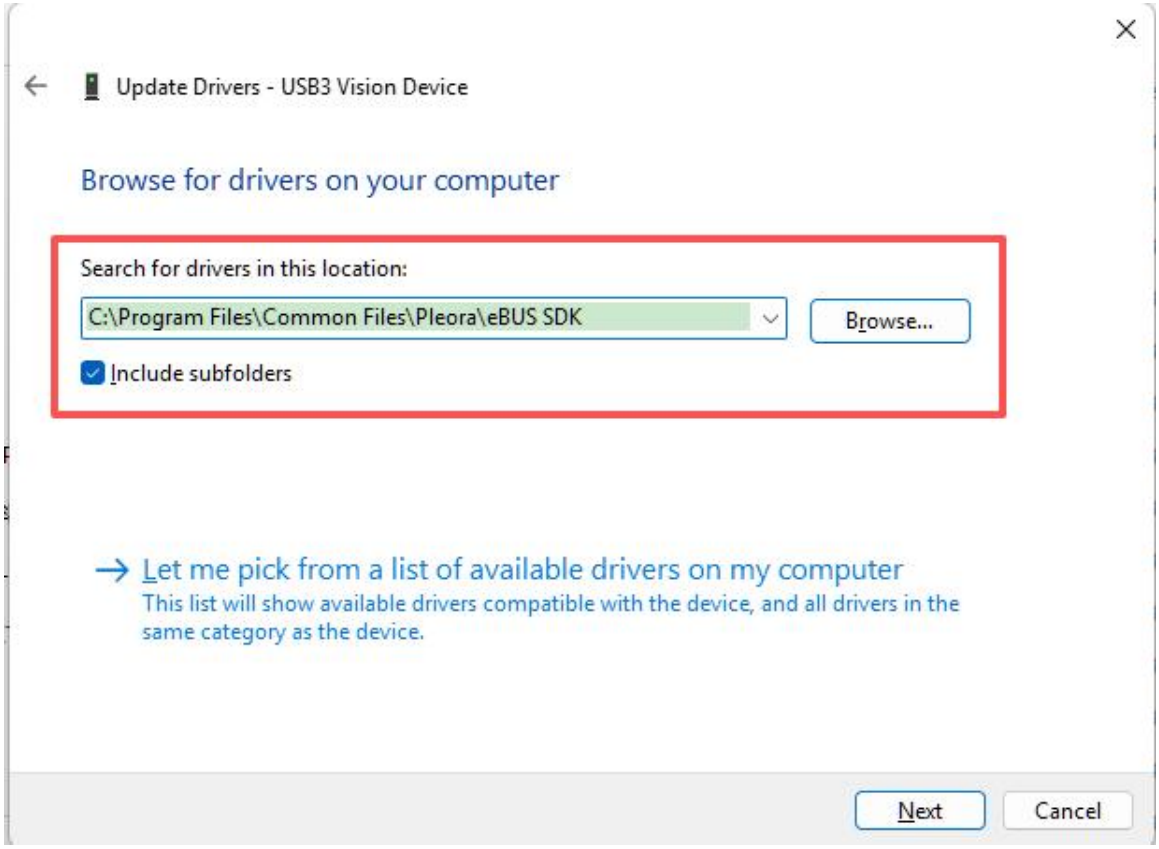


Note: If you still can not find USB3 Vision Device from your Device Manager. Then follow below step to install driver manually.





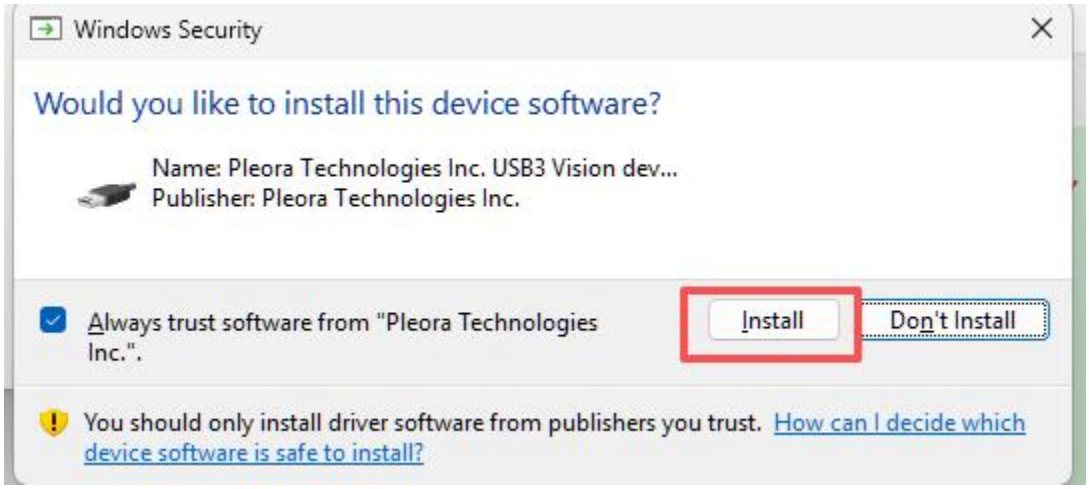




The path depends on the folder you install your software, like below normally:

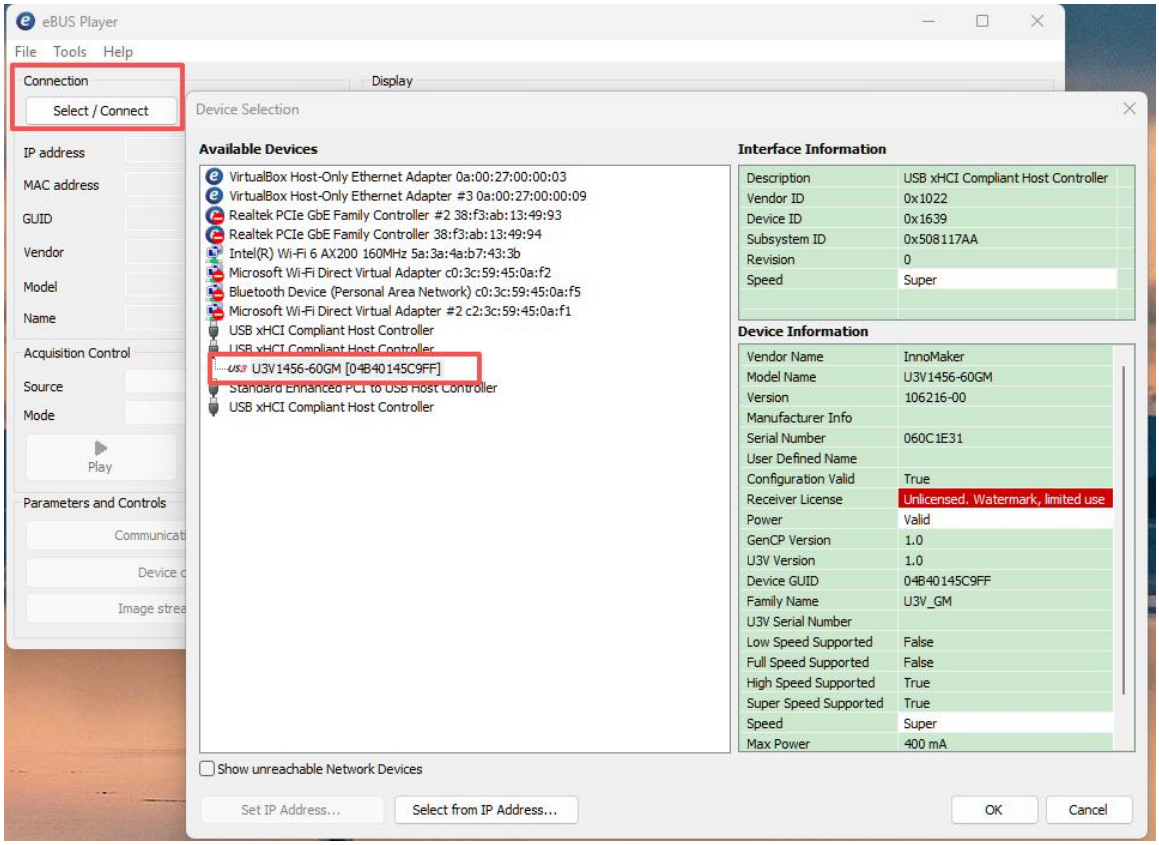
C:\Program Files\Common Files\Pleora\eBUS SDK

XX:\XX\Common Files\Pleora\eBUS SDK



5.1.2 Launch Ebus_player

Double click eBus Player on your desktop.



5.2 Installation Guide On Linux

We test it with raspberry pi5 os Debian bookworm.

5.2.1 Install ebus_player

Download source from our GitHub, follow below command

```
cd U3V-CAM-IMX296
sudo chmod -R a+rwx *
cd ebus_sdk_raspberry_pi/
sudo dpkg -i eBUS_SDK_JAI_Raspberry_Pi4_Pi5_linux-aarch64-arm-6.5.3-7155.deb
```

5.2.2 Install below qt libraries

Note: After ebus sdk install, install below qt libraries for launching ebus_sdk.

```
sudo apt-get update
sudo apt-get install libqt5opengl5
```

Remark: If you runs on the Trixie OS:

```
echo "deb http://deb.debian.org/debian bookworm main" | sudo tee /etc/apt/sources.list.d/bookworm.list

sudo apt update

sudo apt install -t bookworm libavcodec59 libavformat59 libavutil57 libswscale6 libswresample4
```

After it installs successfully, clean up the repo so it doesn't cause issues later:

```
sudo rm /etc/apt/sources.list.d/bookworm.list

sudo apt update
```

5.2.3 Launch Ebus_player

```
cd /opt/jai/ebus_sdk/linux-aarch64-arm/bin
sudo ./eBUSPlayerJAI
```

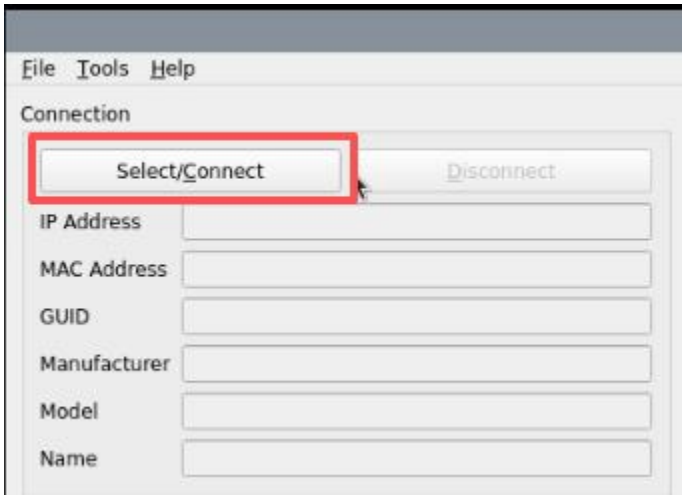
5.3 Open U3V-IMX296-GS Via eBus_Player

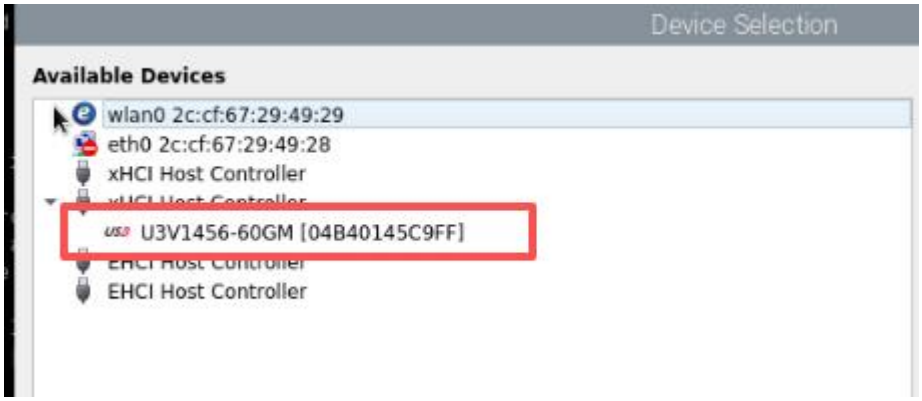
Same Operation on Linux And windows.Follow below steps to open our device U3V-IMX296-GS.

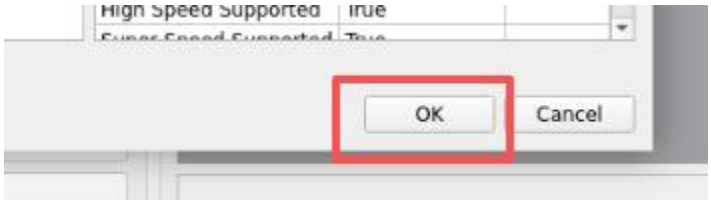
5.3.1 Select Device

Plug USB Cable to Computer, Launch eBus_Player

Click Select/Connect, Choose U3V1456-60GM[04B40145C9FF], Click OK







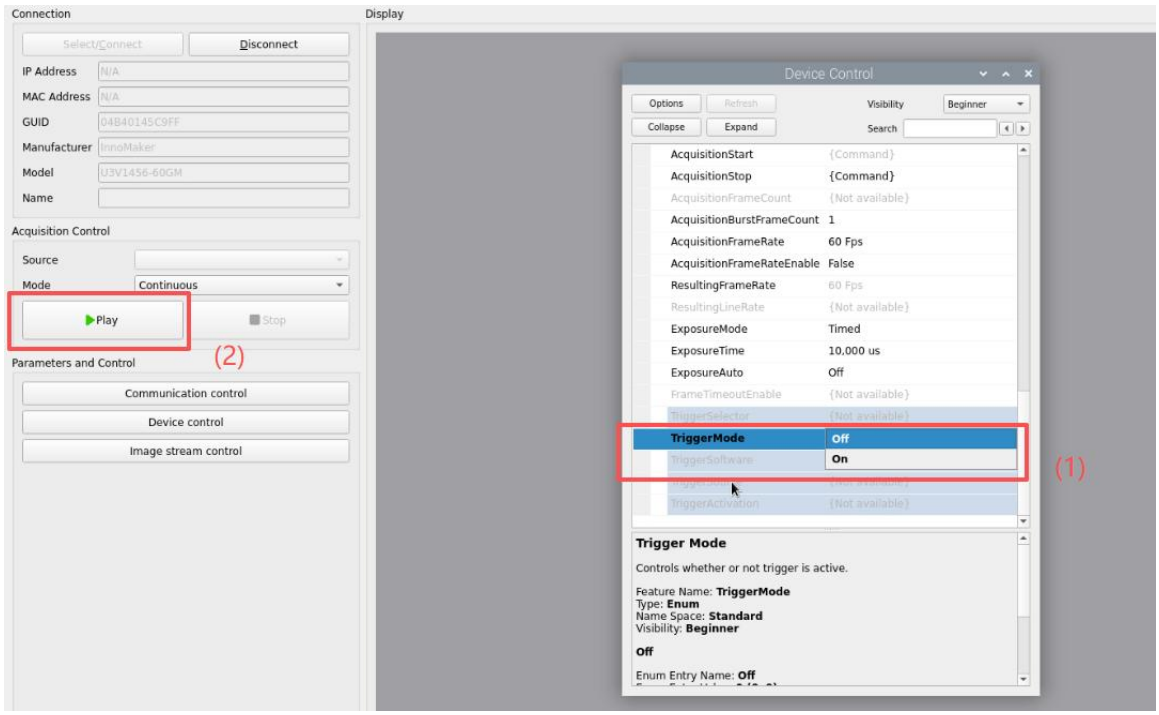
5.3.2 Mode

Click Device control, Choose Guru from Visibility.

DEVICE Control→Guru

6 Key Function Operation Guide

6.1 Free Running Mode



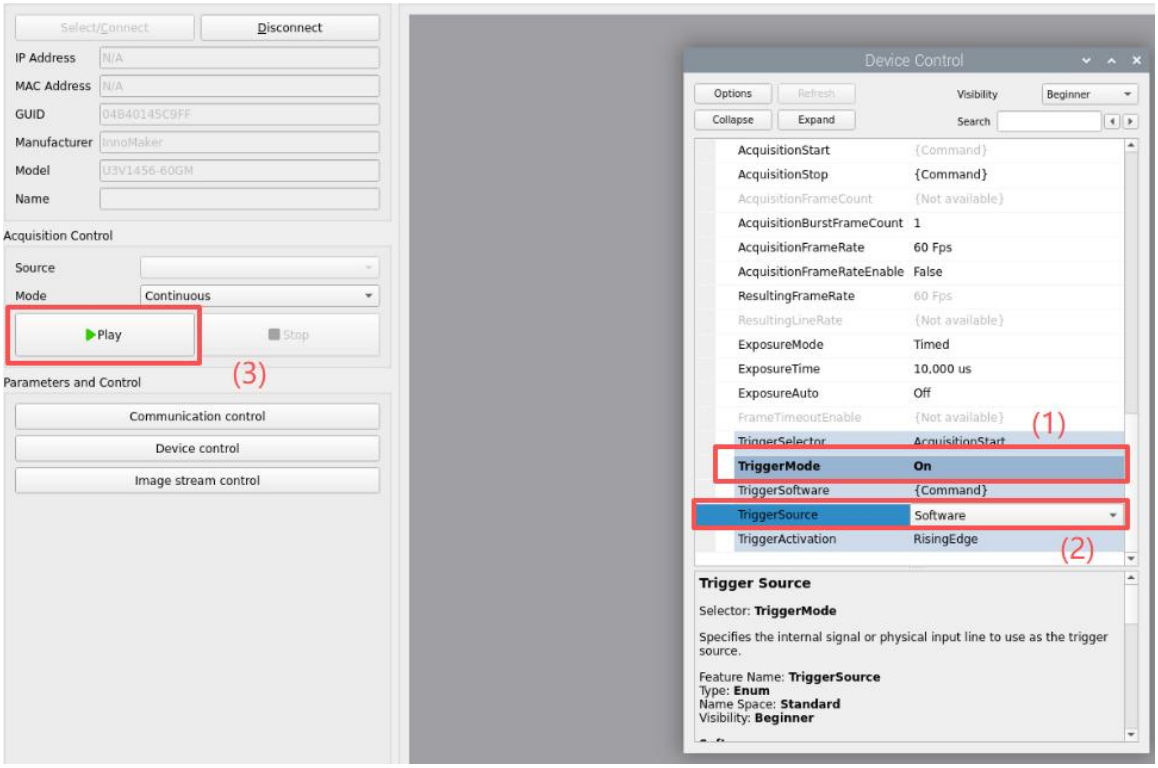
(1)TriggerMode choose off

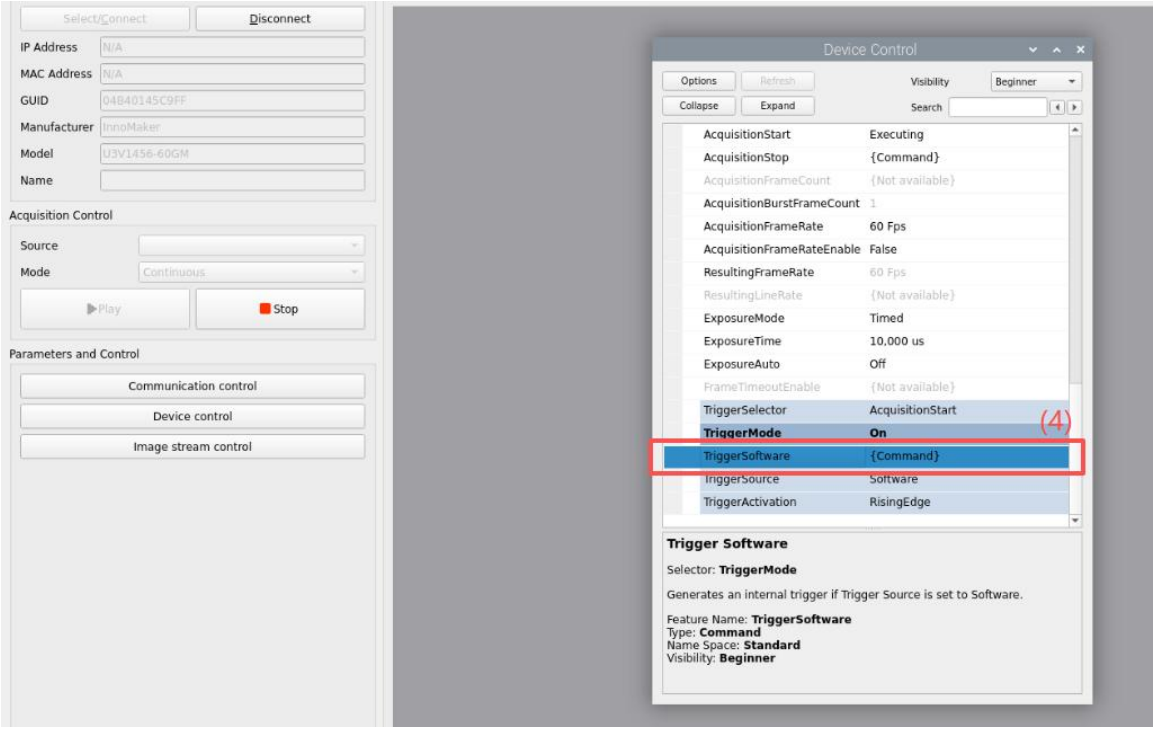
(2) Click Play

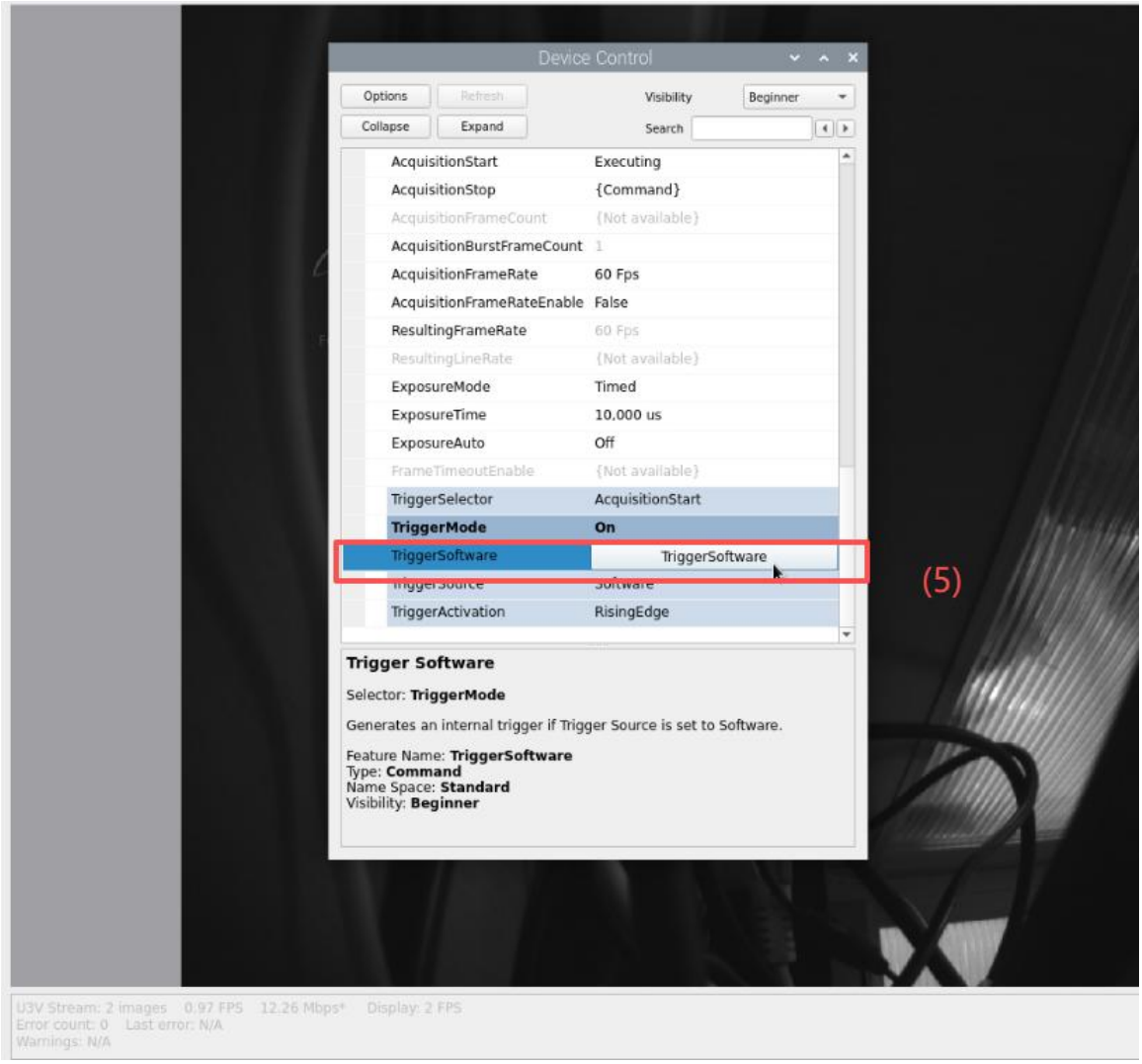
6.2 Software Trigger

6.2.1 Follow Chapter 4.4&4.5 for Circuit routing.

6.2.2 Ebus_Player Operation




Select/Connect
Disconnect
IP Address
MAC Address
GUID
Manufacturer
Model
Name
Acquisition Control
Source
Mode
Play
Stop
Parameters and Control
Communication control
Device control
Image stream control
Device Control
Options
Refresh
Visibility
Beginner
Collapse
Expand
Search
AcquisitionStart
Executing
AcquisitionStop
{Command}
AcquisitionFrameCount
{Not available}
AcquisitionBurstFrameCount
AcquisitionFrameRate
60 Fps
AcquisitionFrameRateEnable
False
ResultingFrameRate
ResultingLineRate
ExposureMode
Timed
ExposureTime
10,000 us
ExposureAuto
Off
FrameTimeoutEnable
TriggerSelector
AcquisitionStart
TriggerMode
On
TriggerSoftware
{Command}
TriggerSource
Software
TriggerActivation
RisingEdge
(4)
Trigger Software
Selector: TriggerMode
Generates an internal trigger if Trigger Source is set to Software.
Feature Name: TriggerSoftware
Type: Command
Name Space: Standard
Visibility: Beginner



(1)TriggerMode : On

(2)TriggerSource : Software

(3)Click Play

(4)Double Click { Command}

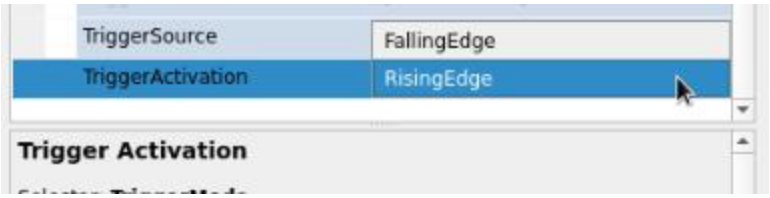
6.2.3 Start software trigger.

Click TriggerSoftware Button.

6.2.4 FallingEdge/Rising Edge

You can also select FallingEdge/Rising Edge from Trigger Activation.

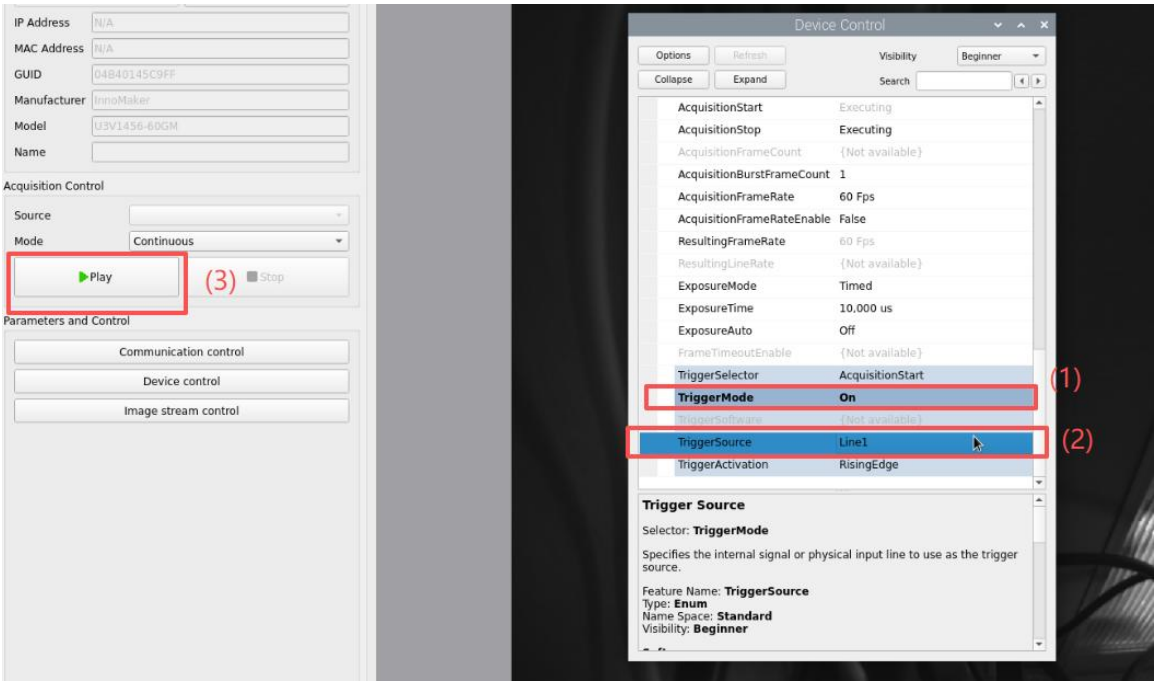
Follow Chapter 3 for details



6.3 Hardware Trigger

6.3.1 Follow Chapter 4.4&4.5 for Circuit routing.

6.3.2 Ebus_Player Operation



(1)Trigger Mode:On

(2)TriggerSource Line1

(3)Play

6.3.3 Run Hardware Trigger Source

We Use raspberry pi5 GPIO23,GPIO25 to generate source signal with a simple scripts.

```
pi@raspberrypi:~ $ cd U3V-CAM-IMX296/
pi@raspberrypi:~ $ su ./imx296.sh
```

```
while true;do
        gpioset gpiochip0 23=1
        sleep 1.9999
        gpioset gpiochip0 23=0
        sleep 0.0033
done
```

6.3.4 FallingEdge/Rising Edge

You can also select FallingEdge/Rising Edge from Trigger Activation.

Follow Chapter 3 for details

7 Aravis

7.1 About Aravis

Features of Aravis Open-Source Software

Aravis is an Open-source software for industrial vision

Aravis is widely used in robotics, industrial automation, scientific imaging, and research because of its reliability, speed, and excellent GenICam compliance. If you need installation instructions or usage examples, feel free to ask!

Aravis is a lightweight, open-source vision library built on Glib/GObject, designed for high-performance video streaming from GenICam-compliant industrial cameras. It runs on Linux, macOS, and Windows. Below is a summary of its key features:

Category	Key Features
Core Functionality	- Implements GenICam standard for camera discovery, control, and image acquisition.

	- Supports GigE Vision (Gigabit Ethernet) and USB3 Vision cameras. - Works with many major brands (Basler, Allied Vision, Point Grey/FLIR, JAI, The Imaging Source, Dalsa, Smartek, etc.).
Protocol Support	- Full GigE Vision 1.x and 2.x support (including GVSP packet resend and leader/trailer). - USB3 Vision support. - Multiple cameras can stream simultaneously on the same host.
Tools & Viewers	- arv-viewer: A powerful GTK-based viewer with live display, histogram, ROI control, register inspection, and GenICam feature tree. - arv-fake-camera: GigE Vision camera emulator for testing without real hardware. - Command-line tools: arv-tool, arv-register-viewer, etc.
Integration	- GStreamer plugin (aravissrc) for easy integration into multimedia pipelines (OpenCV, ROS, etc.). - ROS 1 and ROS 2 drivers available (camera_aravis package). - Bindings for Python (via GObject introspection) and other languages.
Performance	- Zero-copy buffer handling when possible. - Highly optimized for low latency and high frame rates. - Supports jumbo frames, packet size tuning, and real-time kernel optimizations.
Build & Dependencies	- Minimal dependencies: glib2, libxml2, zlib. - Optional: libusb-1.0 (for USB3 Vision), GTK4 (for viewer), GStreamer, libnotify, etc. - Can be compiled with Meson or CMake.
License &	- Licensed under LGPL v2.1+, allowing use in both open-source and

Community	proprietary projects. – Active community; contributions (code, bug reports, camera donations) are welcome.

Official repository: https://github.com/AravisProject/aravis

7.2 Aravis Install Guide

You can refer to the link below for latest version, we use aravis-0.8.35 On Raspberry PI5 for our demo.

https://aravisproject.github.io/aravis/aravis-stable/building.html#installing-aravis

6.2.1 Download source code

Download From our GitHub, extract aravis-0.8.35.tar.xz

```
cd U3V-IMX296-GS
sudo tar -xf aravis-0.8.35.tar.xz
cd aravis-0.8.35
sudo chmod -R a+rwx *
```

Note: Install below dependence before build for raspberry pi trixie os

```
sudo apt install libxml2-dev

sudo apt install libglib2.0-dev

sudo apt install gettext

sudo apt install gobject-introspection libgirepository1.0-dev \

libgtk-3-dev \

libgstreamer1.0-dev libgstreamer-plugins-base1.0-dev
```

```
sudo meson setup build
cd build
sudo ninja
sudo ninja install
```

```
aravis 0.8.35

  Directories
    prefix          : /usr/local
    bindir          : bin
    libdir          : lib/aarch64-linux-gnu
    datadir         : share

  Options
    Viewer          : true
    GStreamer plugin: true
    USB support     : true

Found ninja-1.11.1 at /usr/bin/ninja
```

```
pi@raspberrypi:~/U3V-CAM-IMX296/aravis-0.8.35/build $ ninja
[185/185] Linking target viewer/arv-viewer-0.8
```

Note: If you see errors when meson setup build.

(1): The build can be configured at any time using meson configure in the build directory. meson configure invoked without any other argument will show the configuration options.
On some platforms (like Ubuntu), you may have to configure the dynamic linker (ld) to let it know where the aravis libraries are installed, and run ldconfig as root in order to update ld cache.

```
sudo ldconfig
```

Install dependencies on Ubuntu 20.04
Prior to running meson and ninja, dependencies can be installed using the following(tested on Ubuntu 20.04):

```
sudo apt install libxml2-dev libglib2.0-dev cmake libusb-1.0-0-dev gobject-introspection \
        libgtk-3-dev gtk-doc-tools  xsltproc libgstreamer1.0-dev \
        libgstreamer-plugins-base1.0-dev libgstreamer-plugins-good1.0-dev \
        libgirepository1.0-dev gettext
```

(2): Reconfigure build
You can check missing libraries install missing libraries and re-do meson by below command
Check missing libraries

```
sudo meson setup --reconfigure build
```

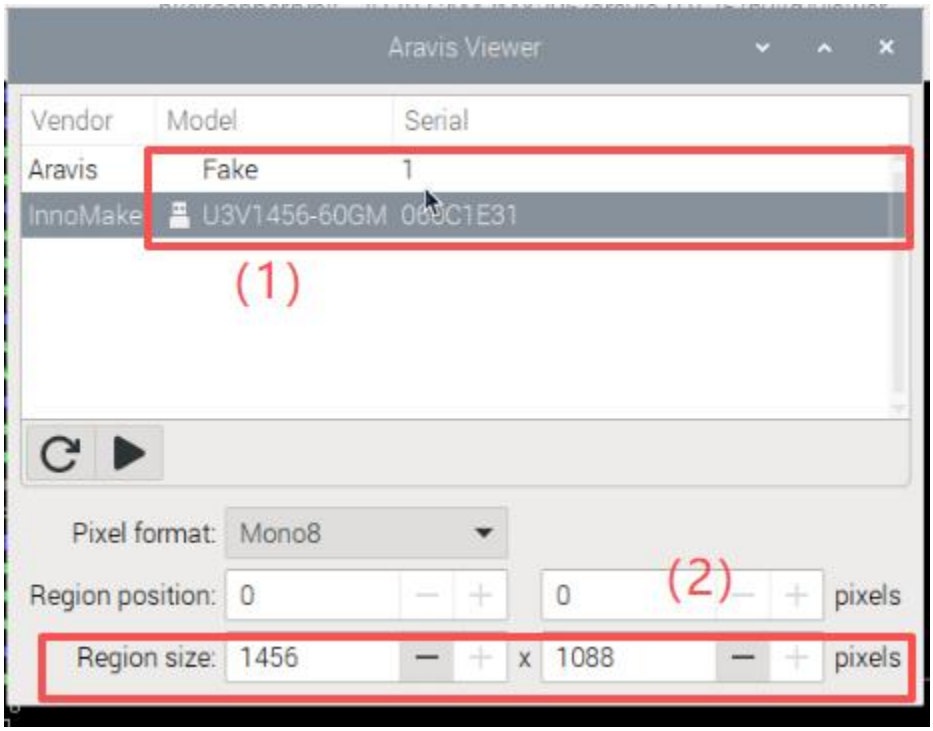
```
Install libraries,the wipe it.
sudo meson setup –wipe build
```

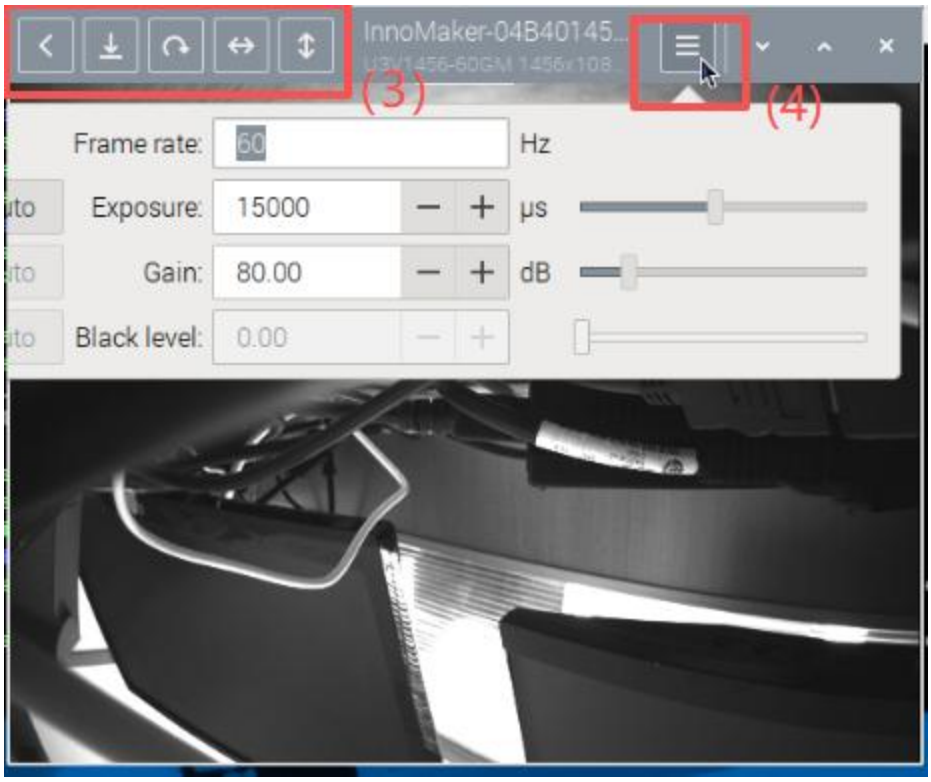
7.3 Launched Aravis

After Aravis installation.

```
cd ~/aravis-0.8.35/build/viewer/
sudo ./arv-viewer-0.8
```

7.4 Aravis Key Parameter





(1)Device Information

(2) Region Size Resolution

(3) 

Save snap shot

Rotate Image to the Right

Flip Image Hozizontally

Flip Image Vertically

(4)Frame Rate,Exposure,Gain

8 Packing List

- **1x U3V-CAM-IM296 With M12 Lens Installed**
- **1x CS Lens Mount**
- **1x USB3.0 Type B Cable 1metre**
- **1x 6-pin Hirose Connector Power & Trigger Cable**

9 Download

https://github.com/INNO-MAKER/u3v-cam-imx296

9.1 Software

- eBusPlayer Windows SDK
- eBusPlayer Linux SDK
- Open source Aravis Linux SDK
- User Manual
- Pre-install Raspberry PI OS System IMG(Install eBusplayer and Aravis)