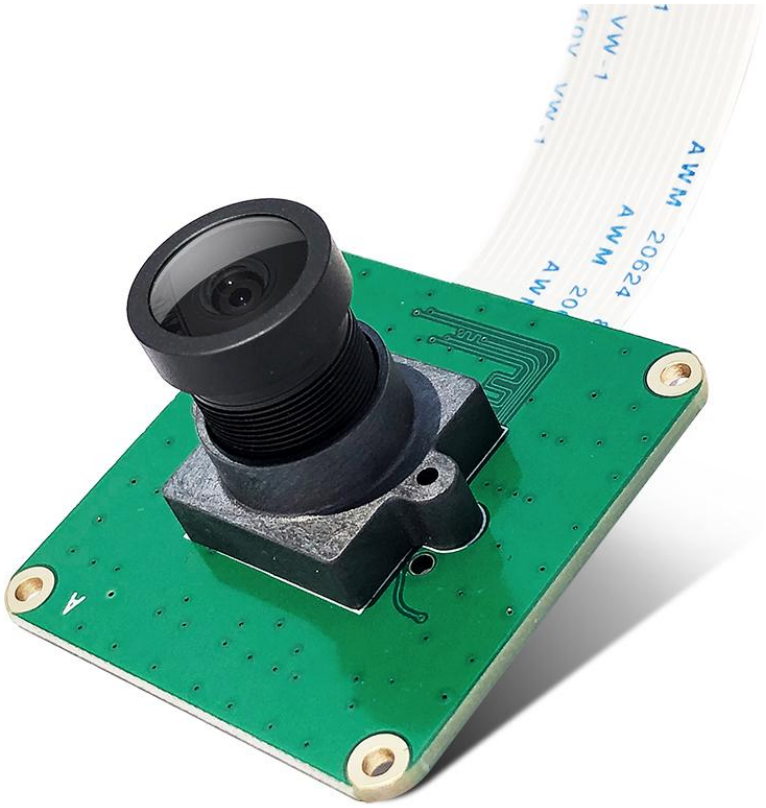
CAM-MIPI290-327-462RAW UserManual



1. General

CAM-MIPI327RAW, CAM-MIPI462RAW, and CAM-MIPI290RAW are modules designed based on Sony's IMX327LQR-C, IMX462LQR-C, and IMX290LLR-C sensors, respectively. These sensor characteristics are quite similar and the software is also relatively compatible. In Linux, they use mutually compatible drivers. Therefore, this manual groups these three camera boards together.

All three sensors belong to Sony's STARVIS series. STARVIS is a trademark of Sony Corporation. STARVIS is back-illuminated pixel technology used in CMOS image sensors for surveillance camera applications. It features a sensitivity of 2000 mV or more per 1 μm^2 (for color products, when imaging with a 706 cd/m^2 light source, F5.6 in 1-second accumulation equivalent), and achieves high picture quality in both visible-light and near-infrared light regions

On-board sensor features with diagonal 6.46 mm (Type 1/2.8) and up to 1920 (H) × 1080 (V) Resolution. With triple power supply, and has low power consumption. High sensitivity, low dark current and no smear are achieved through the adoption of R, G and B primary color mosaic filters. This chip features an electronic shutter with variable charge-integration time. Suitable for surveillance cameras, FA (Factory Automation) cameras, and industrial cameras.

This series of cameras board is paired with a wide-angle camera and features a CSI interface compatible with Raspberry Pi camera interface, making it adaptable to the entire series of Raspberry Pi boards(PI5/PI4/PI3/CM4/CM3/Zero/Zero W). The camera board can directly use the Raspberry Pi's built-in driver, without the need for additional driver installation. It can also be tested directly using the libcamera and RpiCam tools available in the system.

2. Features

(1) CAM-IMX462RAW is an Industrial Camera Module for whole series Raspberry Pi.Support libcamera/rpicam tools.

(2) On-board STARVIS IMX462LQR Color CMOS Sensor Active Pixel type Solid-state Image Sensor with Square Pixel Array 2 MP Effective Pixels. Diagonal 6.46 mm (Type 1/2.8). Pixel count: 1920x1080.

(3) High dynamic range (HDR): Supports a wide dynamic range, allowing for clear detail representation in high-contrast scenes. STARV technology: Delivers clear and detailed images even in low-light environments

(2) The built-in IMX462 driver on the Raspberry Pi Os supports RAW10 and RAW12 output formats, with resolutions of 1920x1080, 1280x720. The maximum frame rate can reach up to 60fps.

(3) Comes a wide angle Lens. Fov(D)=148 degrees, Fov(H)=118 degrees. Focal distance is adjustable.

3. Hardware Description

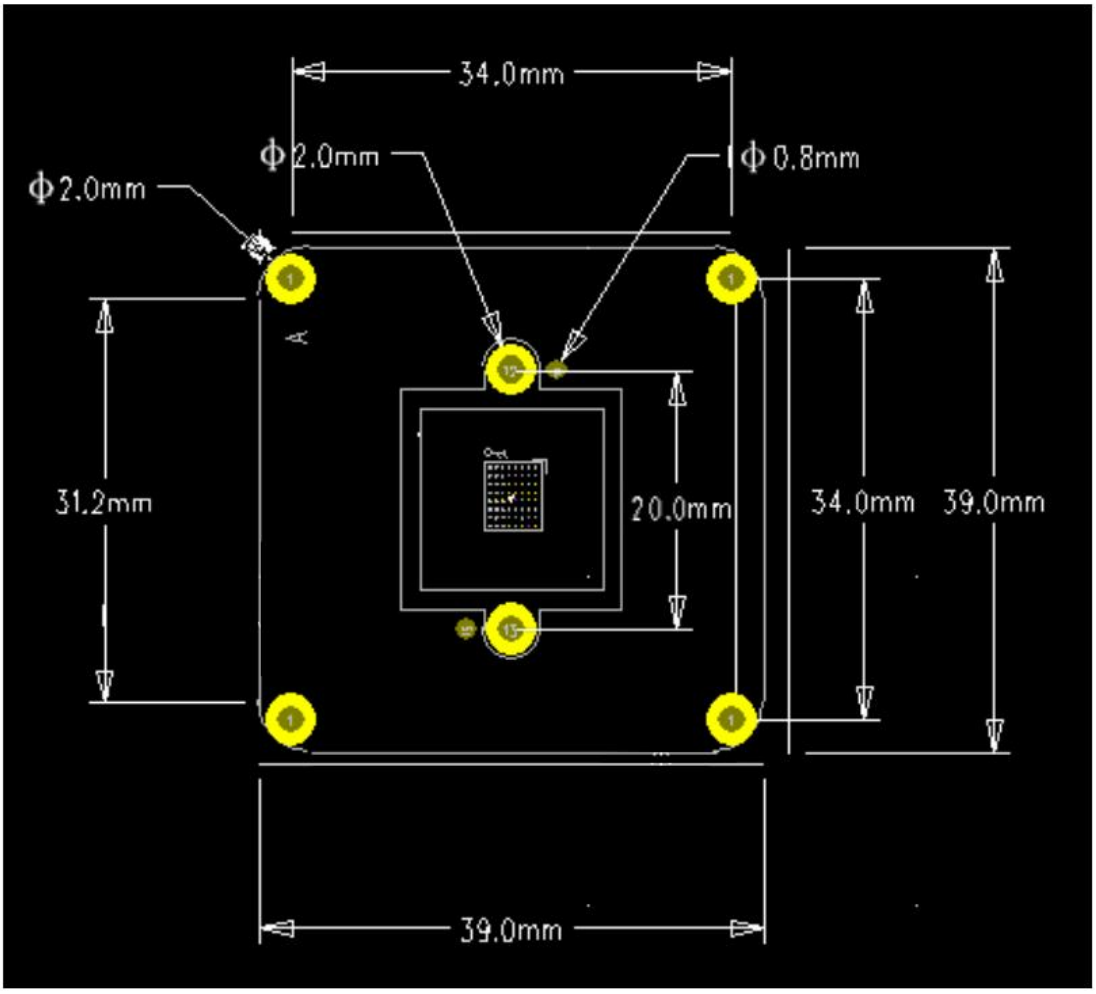
2.1 Overview

IMX290LLR-C/IMX462LQR-C /IMX327LQR-C Sensor		
Optical size	Type 1/2.8" , Diagonal 6.46 mm	
Effective pixels	2.13 M pixels	
Number of recommended recording pixels	2.07M pixels	
Readout rate	RAW10: 1280x720 @60fps 1920x1080 @60fps RAW12: 1280x720 @60fps 1920x1080 @60fps	
Sensor Resolution	1920 (H) × 1080 (V)	
Sensor size		
Pixel size /Unit cell size	2.9 µm (H) × 2.9 µm (V)	
Wide dynamic range	IMX290LLR IMX327LQR	Multiple exposure WDR Digital overlap WDR
	IMX462LQR	Multiple exposure HDR Digital overlap HDR
CDS / PGA function	IMX290LLR	0 dB to 30 dB: Analog Gain 30 dB (step pitch 0.3 dB) 30.3 dB to 72 dB: Analog Gain 30 dB + Digital Gain 0.3 to 42 dB (step pitch 0.3 dB)
	IMX462LQR	0 dB to 29.4 dB: Analog Gain 29.4 dB (step pitch 0.3 dB) 29.7 dB to 71.4 dB: Analog Gain 29.4 dB + Digital Gain 0.3 to 42 dB (step pitch 0.3 dB)
	IMX327LQR	0 dB to 27 dB: Analog Gain 27 dB (step pitch 0.3 dB) 27.3 dB to 69 dB: Analog Gain 27 dB + Digital Gain 0.3 to 42 dB (step pitch 0.3 dB)
Readout mode	All-pixel scan mode 720p-HD readout mode Window cropping mode	

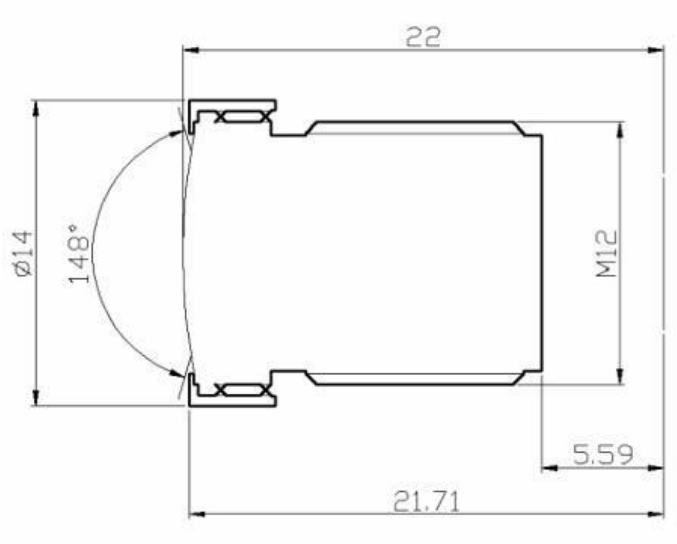
	Vertical / Horizontal direction-normal / inverted readout mode
Output formats	RAW12/RAW10
Optical black	Horizontal (H) direction: Front 0 pixels, rear 0 pixels Vertical (V) direction: Front 10 pixels, rear 0 pixels
Dummy	Horizontal (H) direction: Front 0 pixels, rear 3 pixels Vertical (V) direction: Front 0 pixels, rear 0 pixels
Night vision	STARVIS
PCBA	
Size	39mm x 39mm
Weight	4g
Mounting Hole	Φ2mm x 4
Lens	
Field of view	Fov(D) = 148 degrees , Fov(H) = 118 degrees
Focal Length	2.8 mm
TV DISTORTION	<-17%
F(N) /Aperture	2.2
Focal Distance	Adjustable
Software	
Linux integration	V4L2 Libcamera RpiCam

2.2 Size

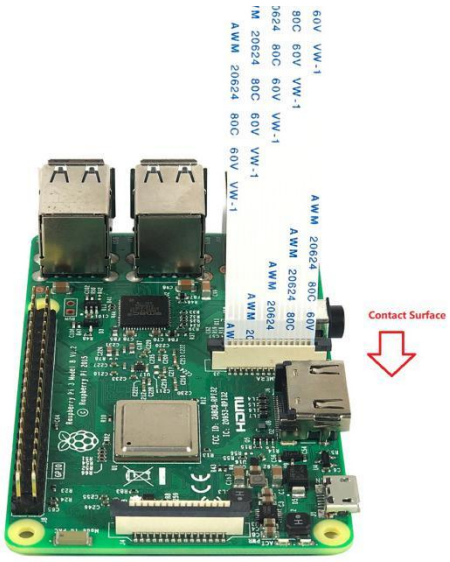
2.2.1 PCB Size

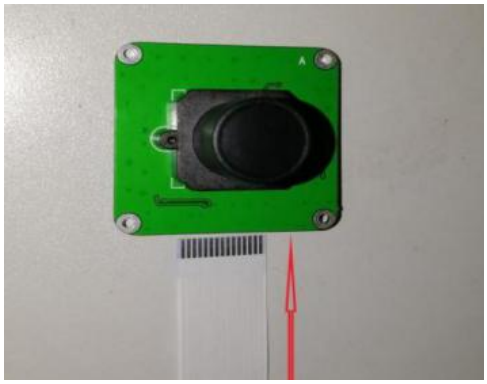


2.2.2 Len Size



2.3 Connection Of The Hardware

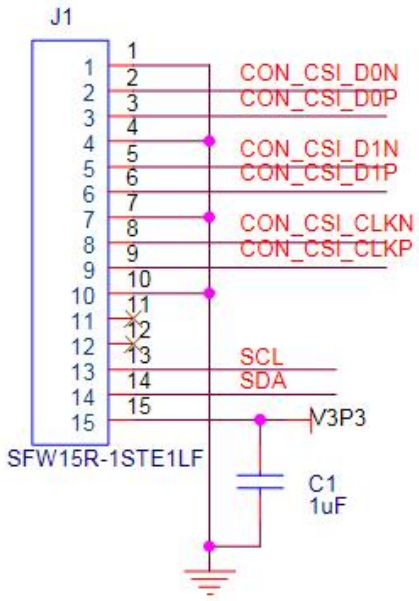




2.4 Pin-Out

3.4.1 Signal/Power Connector J1

The J1 pin map is same Raspberry Pi camera.



PIN	Symbol	Description
1	GND	Ground Pin
2	CON_CSI_D0N	Pixel Data Lane0 Negative
3	CON_CSI_D0P	Pixel Data Lane0 Positive
4	GND	Ground Pin
5	CON_CSI_D1N	Pixel Data Lane1 Negative
6	CON_CSI_D1P	Pixel Data Lane1Positive
7	GND	Ground Pin
8	CON_CSI_CLKN	Pixel Clock Output Form Sensor Negative
9	CON_CSI_CLKP	Pixel Clock Output Form Sensor Positive
10	GND	Ground Pin
11	None	None
12	None	None
13	SCL	CLK input, SIO_C of SCCB
14	SDA	DATA input,SIO_D of SCCB
15	3.3V Power	Power Supply

4. Using Raspbian Build-In Driver

4.1 Load Raspberry Pi image

Prepare a capacity of more than 8GB TF card(16Gb Class10 is better) and a card reader. Load the image file on to the SD card, using the instructions provided on the Raspberry Pi website for Linux, Mac or PC:
https://www.raspberrypi.com/documentation/computers/getting-started.html#installing-the-operating-system

Raspbian Image download:
https://www.raspberrypi.org/downloads/

4.2 Driver Sources Codes

The existing IMX290 driver and overlay also work fine with IMX327 and IMX426.

The open source driver on Raspbian:
https://github.com/raspberrypi/linux/blob/rpi-5.10.y/drivers/media/i2c/imx290.c

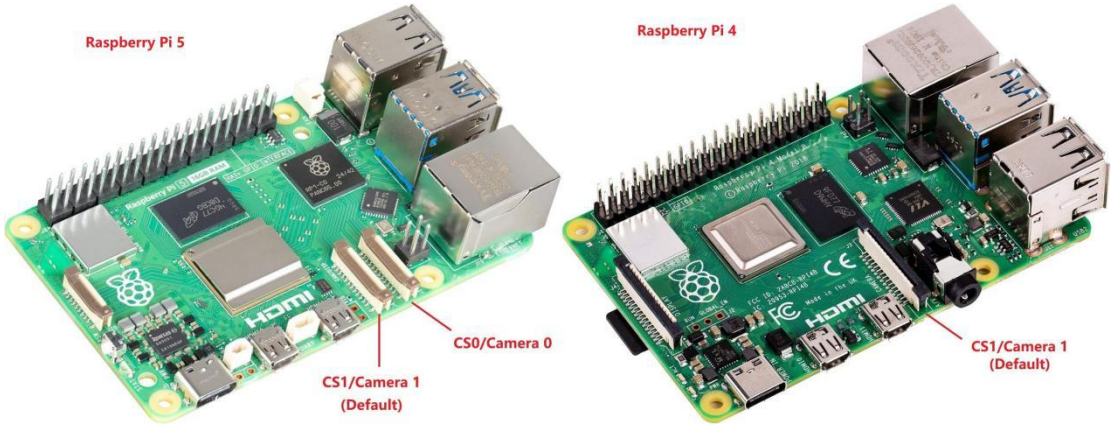
Reference codes：
https://github.com/torvalds/linux/blob/master/drivers/media/i2c/imx290.c

4.3 Dtoverlay on Raspberry Pi board

(1) Open the config.txt on Raspbian:

New version:
sudo nano /boot/firmware/config.txt
Legacy version:
sudo nano /boot/config.txt

(2)
The Pi 5 board exposes both the CS0 and CS1 interfaces, while the Pi 4 board only exposes the CS1 interface. However, the CM4 core board exposes both the CS0 and CS1 interfaces. .



Add the following text below the [all] line in the config.txt file.,The default interface being used is cam1.
For CSI 1:
dtoverlay=imx290,clock-frequency=74250000,cam1
For CSI 0:
dtoverlay=imx290,clock-frequency=74250000,cam0

```
# Automatically load initramfs files, if found
auto_initramfs=1

# Enable DRM VC4 V3D driver
dtoverlay=vc4-kms-v3d
max_framebuffers=2

# Don't have the firmware create an initial video= setting in cmdline.txt.
# Use the kernel's default instead.
disable_fw_kms_setup=1

# Run in 64-bit mode
arm_64bit=1

# Disable compensation for displays with overscan
disable_overscan=1

# Run as fast as firmware / board allows
arm_boost=1

[cm4]
# Enable host mode on the 2711 built-in XHCI USB controller.
# This line should be removed if the legacy DWC2 controller is required
# (e.g. for USB device mode) or if USB support is not required.
otg_mode=1

[cm5]
dtoverlay=dwc2,dr_mode=host

[all]
dtoverlay=imx290,clock-frequency=74250000,cam1
dtoverlay=imx290,clock-frequency=74250000,cam0
```

(3) And then press ctrl+ x to exit nad press 'y' to save.

(4) Rebooted your Pi

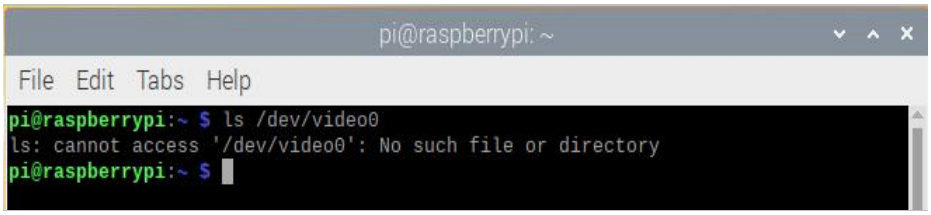
```
sudo reboot
```

(5) Use below command to check the camera is ready.

```
ls /dev/video0
```

Successful:

Unsuccessful:



(6) Some older versions of the tuning files may have issues. You can download and use the latest version from our GitHub. This issue does not occur in newer Raspberry Pi systems. After testing

the actual performance, you can decide whether to download the tuning file.

Download tuning file and point to use this tuning file
git clone https://github.com/INNO-MAKER/CAM-MIPI327RAW-and-CAM-MIPI462RAW.git

Or download from below link :
https://github.com/INNO-MAKER/CAM-MIPI327RAW-and-CAM-MIPI462RAW/blob/main/innomakerpi5_imx290.json

Use the following command to perform a simple test
libcamera-still -t 0 --tuning-file /home/pi/CAM-MIPI327RAW-and-CAM-MIPI462RAW/innomakerpi5_imx290.json

4.4 Rpicam-apps / Libcamera

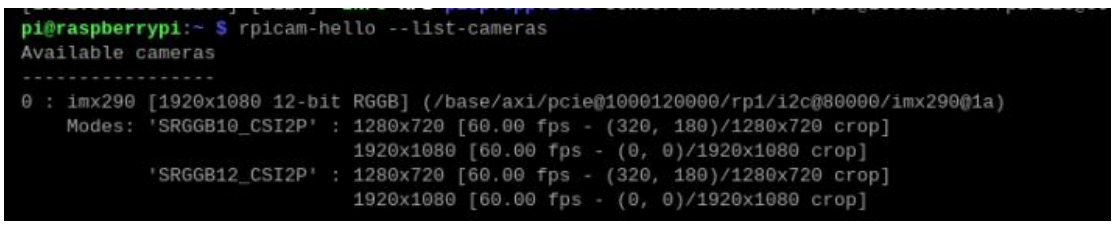
libcamera is an open source Linux community project. More information is available at the libcamera website:
https://libcamera.org/

The libcamera source code can be found and checked out from the official libcamera repository.
https://git.linuxtv.org/libcamera.git/

Raspberry Pi OS Bookworm renamed the camera capture applications from libcamera-* to rpicam-*. Symbolic links allow users to use the old names for now. Adopt the new application names as soon as possible. Raspberry Pi OS versions prior to Bookworm still use the libcamera-* name.
https://www.raspberrypi.com/documentation/computers/camera_software.html#rpicam-apps
https://github.com/raspberrypi/rpicam-apps

4.4.1 Camera Information.

Enter the following command to query camera information, and you will be able to see the data formats and frame rates supported by OV9281 on the Raspberry Pi. R8 and R10 represent the RAW8 and RAW10 original data formats, respectively.
rpicam-hello --list-cameras



4.4.2 Camera Preview

The following command can be used to directly open the camera preview. Two cameras can be opened simultaneously.

```
rpicam-hello -t 0 --camera 0
rpicam-hello -t 0 --camera 1
```

4.4.3 Camera Modes

You can use the result of --list-cameras, and then specify the camera operating mode with the --mode option in a colon-separated format:
<width>:<height>:<bit-depth>:<packing>

If the values you provide do not match exactly, the system will select the closest available option for the sensor. You can use either packed (P) or unpacked (U) packing formats. This affects the format of stored video and still images, but it does not affect the format of frames passed to the preview window. The default bit-depth is 12, and the default packing is P (packed).

Set the working mode of camera 1 to 1920x1080 RAW12, and 60 frame rate using the following command

```
rpicam-still --viewfinder-mode 1920:1080:12 --framerate 60 -t 0 --camera 1
```

Set the working mode of camera 0 to 1280x800, RAW10, and 114 frame rate using the following command

```
rpicam-still --viewfinder-mode 1280:720:10 --framerate 60 -t 0 --camera 0
```

5. User Manual Version Descriptions

Version	Description	Date	E-mail
V1.0		2021.11.25	support@inno-maker.com sales@inno-maker.com
V1.1	Add camera runing on the CM4 Add Rpican	2024.08.09	support@inno-maker.com sales@inno-maker.com
V1.2	Rewrite the description of Rpicam-apps / Libcamera.	2025.09.20	support@inno-maker.com sales@inno-maker.com

If you have any suggestions, ideas, codes and tools please feel free to email to me. I will update the user manual and record your name and E-mail in list. Look forward to your letter and kindly share.