



Status of ISP Support in V4L2

Presented at Linux Media Summit 2026

2026-05-26, Nice, France

Bonjour

My name is Laurent Pinchart

- Founder and CEO of **Ideas on Board**
- Electrical engineer and software developer
- 20+ years of experience with **Linux** and cameras
- Linux kernel core contributor and maintainer
- Lead architect of the **libcamera** project



I am  and live in 

✉ laurent.pinchart@ideasonboard.com

📄 9423 1B98 0100 EC61 9AC1 0E10 F045 C2B9 6991 256E

Introduction



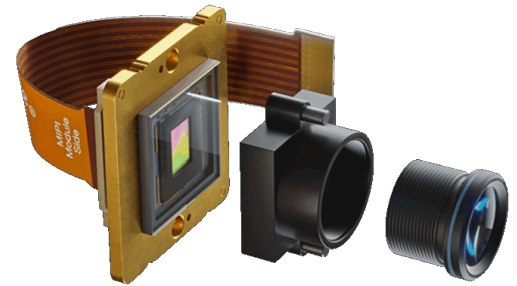
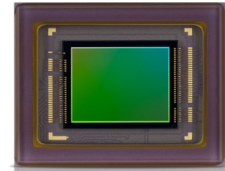
Image sensors

There's a steady flow of new drivers for image sensors. Merged since last media summit:

- OmniVision: OV6211, OG0VE1B, OV2735, OS05B10, OV2732
- Sony: IMX111
- Samsung: S5KJN1, S5K3M5
- Toshiba: T4KA3

All but one are mode-based.

More drivers under review on the list.



Hardware Support

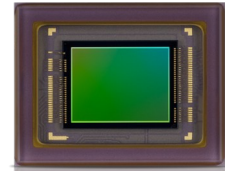


Image sensor vendors

Engaged with two sensor vendors to help them develop and upstream **dynamically configurable** drivers. A third important vendor is expected to follow.

How do we engage with more vendors, especially from Taiwan and China?

If sensor vendors can develop and upstream dynamically configurable drivers, why should we keep accepting mode-based drivers from other contributors?



Hardware Support



ISPs

Smaller flow: only one driver merged since last media summit (Arm Mali C55, Renesas RZ/V2H), but more on the list:

- camss (Qualcomm)
- neoisp (NXP i.MX95)
- rkisp2 (Rockchip RK3588)
- rpp-x1 (Dream Chip, Renesas R-Car V4H)

Interest from vendors is growing, but far from being a done deal. More outreach is required, and more outreach is ongoing.



Hardware Support



Software ISP

The libcamera software ISP is not just a way to support platforms without a software ISP. It provides a stop-gap solution on platforms where vendors don't want to provide or support an open solution. We have also found it can then provide an incentive for vendors to be more open.

Development of the software ISP is continuing, with a focus on more algorithms with GPU acceleration.



Hardware Support



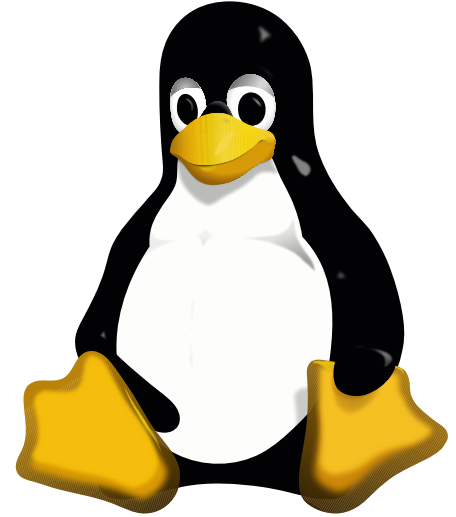
We need more, faster

Many new API extensions have been proposed:

- Internal pads
- Generic metadata
- Raw sensor model
- Multi-context
- Media jobs

Progress to merge those is very slow. More V4L2 shortcomings have been identified already.

How can we accelerate progress?



Kernel APIs

