Camera modules in firmware and UAPI

Sakari Ailus <sakari.ailus@linux.intel.com>
2025-05-13

What are camera modules?

- Generally an entity that contains one or more of the following: the camera sensor, VCM, lens, spring, UV filter and other components
- The terminology varies, in some contexts additional assumptions are made
- Some also use the term “sensor module”, but it’s less established
- Little or no functionality from software point of view
- No system firmware nodes for camera modules

Why should the camera modules be present in system firmware?

- Field of view
- Effect of applying comment to VCM (spring constant and force / current)
- Lens and its mounting position
- LSC table
- Camera module does matter for the entire camera stack!

Camera module database

- Store information on modules that isn’t best kept in DT
- Field of view
- Lens, spring constant, VCM, LSC tables
- Dimensions
- Electrical connections
- On unnamed modules especially
  - Photos
  - Description of the module
  - Any strings printed on the module

How to identify camera modules?

- Generally camera modules have been manufactured by a vendor and they have a model name, perhaps even a revision
  - But this information is not always available!
- Two main cases
  - Module model known
  - Module model not known

Devicetree

- String-based identification for named modules
  - Akin to the `compatible` string
  - “`gpio`” if the vendor is unknown
  - E.g. “arducam,B0449” (for Raspberry Pi, imx519 sensor)
- Unnamed modules get a numerical id
  - E.g. 1 (Nokia N900 front camera, vs6555 sensor)

UAPI

- UAPI to focus on device identification
- Module identification string or numerical ID, via controls in the sensor sub-device
- Revisions, too
- The rest of the necessary information from the database (LSC table etc.) probably best kept in userspace