Camera Applications with libcamera and PipeWire

Embedded Open Source Summit
Prague 2023

Kieran Bingham
Daniel Scally
dan.scally@ideasonboard.com
“libcamera exists, please use it.”
Hi, I’m Dan

• libcamera
• Applications
• Demo
• Summary
• Q+A
Hi, I’m Dan
libcamera
Applications
Demo
Summary
Q+A
camera@1 {
  ...
}

port@0 {
  ...
}
camera_csi_out: endpoint@0 {
    remote-endpoint = <&isp_csi_in>;
};

isp@0 {
  ...
}

port@0 {
  ...
}
isp_in: endpoint@0 {
    remote-endpoint = <&camera_csi_out>;
};

How it works with DT
How it's supposed to work on ACPI

Device (ISP)
{
    Name (_DSD, Package () {
        ToUUID("dbb8e3e6-5886-4ba6-8795-1319f52a966b"),
        Package () {
            "port@0", "PRT0",
        }
    })
}

Name (PRT0, Package() {
    ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
    Package () {
        "reg", 0, /* CSI-2 port number */
    },
    ToUUID("dbb8e3e6-5886-4ba6-8795-1319f52a966b"),
    Package () {
        Package () { "endpoint@0", "EP00" },
    }
})

Name (EP00, Package() {
    ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
    Package () {
        Package () { "reg", 0 },
        "remote-endpoint", Package () \_SB.PCI0.I2C2.CAM0, "port@0", "endpoint@0" },
    }
})

Device (CAM0)
{
    Name (_DSD, Package () {
        ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
        Package () {
            Package () { "compatible", Package () { "nokia.smia" } },
        },
        ToUUID("dbb8e3e6-5886-4ba6-8795-1319f52a966b"),
        Package () {
            Package () { "port@0", "PRT0" },
        }
    })
}

Name (PRT0, Package() {
    ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
    Package () {
        "reg", 0,
    },
    ToUUID("dbb8e3e6-5886-4ba6-8795-1319f52a966b"),
    Package () {
        Package () { "endpoint@0", "EP00" },
    }
})

Name (EP00, Package() {
    ToUUID("daffd814-6eba-4d8c-8a91-bc9bbf4aa301"),
    Package () {
        Package () { "reg", 0 },
        "remote-endpoint", Package () \_SB.PCI0.ISP, "port@0", "endpoint@0" },
    }
})
Device (CIO2)
{
  Name (_HID, "INT343E") // The CSI-2 Receiver
  ... and no ports / endpoints at all...
}

Device (CAM0)
{
  Name (_HID, "INT33BE" /* Camera Sensor OV5693 */)  
    Method (SSDB, 0, NotSerialized)
    {
      Name (PAR, Buffer (0x6C)
      {
/* Data representation as it is in ACPI SSDB buffer */
  struct cio2_sensor_ssdb {
      u8 version;
      u8 sku;
      u8 guid_csi2[16];
      u8 devfunction;
      u8 bus;
      u32 dphylinkenfuses;
      u32 clockdiv;
      u8 link;
      u8 lanes;
      u32 csiparams[10];
      u32 maxlanespeed;
      u8 sensorcalibfileidx;
      u8 sensorcalibfileidxInMBZ[3];
      u8 romtype;
      u8 vcmtype;
      u8 platforminfo;
      u8 platformsubinfo;
      u8 flash;
      u8 privacylevel;
      u8 degree;
      u8 mipilinkdefined;
      u32 mclkspeed;
      u8 controllogicid;
      u8 reserved1[3];
      u8 mclkport;
      u8 reserved2[13];
  } __packed;
/* Data representation as it is in ACPI SSDB buffer */
      return (PAR) /* \_SB\_PCI0.I2C2.CAM0.SSDB.PAR */
      }
      }
  }
struct property_entry cio2_properties[] = {
    PROPERTY_ENTRY_REF(
        "remote-endpoint",
        &sensor_endpoint),
};

struct software_node cio2_node = {
    .name = "int343e",
};

struct software_node cio2_port = {
    .name = "port@0",
    .parent = &cio2_node,
};

struct software_node cio2_endpoint = {
    .name = "endpoint@0",
    .parent = &cio2_port,
    .properties = cio2_properties,
};

How we fixed it

struct property_entry sensor_properties[] = {
    PROPERTY_ENTRY_REF(
        "remote-endpoint",
        &cio2_endpoint),
};

struct software_node sensor_node = {
    .name = "int33be",
};

struct software_node sensor_port = {
    .name = "port@0",
    .parent = &sensor_node,
};

struct software_node sensor_endpoint = {
    .name = "endpoint@0",
    .parent = &sensor_port,
    .properties = sensor_properties,
};

drivers/media/pci/intel/ipu3/cio2-bridge.c
The resulting topology
Hi, I’m Dan

libcamera

Applications

Demo

Summary

Q+A
Camera Pipeline Complexity
libcamera architecture
...and it works!
Hi, I’m Dan
libcamera
Applications
Demo
Summary
Q+A
The Linux multimedia camera stack

Linux Kernel / V4L2 Video Devices / V4L2 SubDevices / Media Controller

(Thanks to Robert Mader for this overview concept!)
So, What should I use?

User facing devices go well with PipeWire directly or through a supported framework.

Embedded devices should usually target supporting frameworks that help with the overall use case.

Frameworks or very custom / industrial devices or applications with specific camera requirements can use libcamera natively.

Product images are for illustration only
- Likely a good candidate for a direct libcamera implementation, *and* PipeWire support for desktop
- GStreamer pipeline support already possible through the gstlibcamerasrc
- Direct Python example now available
- We have a Google Summer of Code project actively looking at native libcamera integration now
Camera Streamer

- **/snapshot** (JPEG image)
  - Get a high-resolution snapshot image from the server.
  - Uses resolution specified by `-camera-snapshot.height`.
  - `/snapshot?max_delay=0` to get a snapshot captured exactly now.
  - `/snapshot?max_delay=300` (default) to get a cached snapshot captured up-to 300 ms in the past.

- **/stream** (MJPEG stream)
  - Get a live stream. Works everywhere, but consumes a ton of bandwidth.
  - Uses resolution specified by `-camera-stream.height`.

- **/webrtc** (HTTP page / iframe)
  - Get a live video using WebRTC (low-latency streaming with latency of around 100ms).
  - Uses resolution specified by `-camera-video.height`.

- **/video** (IP Camera)
  - Get a live (H264) video stream best suited to current browser in a maximum compatibility mode choosing automatically between one of the below formats.
  - Uses resolution specified by `-camera-video.height`.
  - `/video.mp4`
    - get a live video stream in MP4 format (Firefox, with latency of around 1s if FFmpeg enabled).

https://github.com/ayufan/camera-streamer
Camera Streamer

https://github.com/ayufan/camera-streamer
Video Conferencing with WebRTC is nearly* upstream

Demonstrated at the previous conference

- Lacks correct format negotiation
- Stride not correctly managed
  - Affects frame sizes that are not a multiple of 32, in width. 1280x720 is usable
- PipeWire issue with multiple cameras.
  - Fixed to single camera for the moment
- Segmentation faults with RPi chromium build with the V4L2 M2M decoders.
  - A pain - but not related to the camera work

Technical Showcase stand:
- Embedded Linux conference Europe (Dublin 2022)
Chromium Pipewire integration is /so/ close

- [Video Capture Linux: add backend for portal / pipewire cameras](https://chromium-review.googlesource.com/c/chromium/src/+/3308882)
- [Update config and include paths for PipeWire / WebRTC](https://chromium-review.googlesource.com/c/chromium/src/+/3308882)
Firefox support should land in 116
...it's working!
Desktop PipeWire support is growing fast
Gnome Camera App - Original Design

Phone

Because the entire 4:3 photo needs to fit onto the view and the screen is quite tall, we need to do something with the additional height. This is done by making the headerbar taller than usual and vertically centering the buttons.

https://www.omgubuntu.co.uk/2020/02/gnome-camera-app-mockup
https://gitlab.gnome.org/GNOME/Incubator/snapshot
Convergent Mobile Application Support
Hi, I’m Dan
●libcamera
●Applications
●Demo
●Summary
●Q+A
Hi, I’m Dan
• libcamera
• Applications
• Demo
• Summary
• Q+A
“A bunch of smart people are using libcamera, please follow them.”

- me
libcamera

Safe Rust bindings for libcamera

by Jurgis and 3 contributors. Co-owned by Dennis Shtatnov.

2 releases
0.2.1 Feb 18, 2023
0.2.0 Feb 18, 2023
0.1.0 Feb 5, 2023

#21 in Multimedia
33 downloads per month

MIT/Apache
265KB
3.5K SLoC

Dependencies
~0.7–2.7MB
~60K SLoC
- bitflags 2.0.0-rc.2
- drm-fourcc
- libc
- libcamera-sys
- num_enum
- smallvec
- thiserror

libcamera-rs

Experimental Rust bindings for libcamera.

https://lib.rs/crates/libcamera
By the way, we are hiring
jobs@ideasonboard.com