
HE IG VD

Intelligence Artificielle pour les systèmes autonomes (IAA)

Systèmes autonomes embarqués

Prof. Yann Thoma - Prof. Marina Zapater

Février 2024

Basé sur le cours du Prof. A. Geiger

Hes·so
Haute Ecole Spécialisée de Suisse occidentale

Reconfigurable & embedded Digital Systems

Outline

Today's lesson

→ **Sensors**

→ The embedded systems equipped in cars

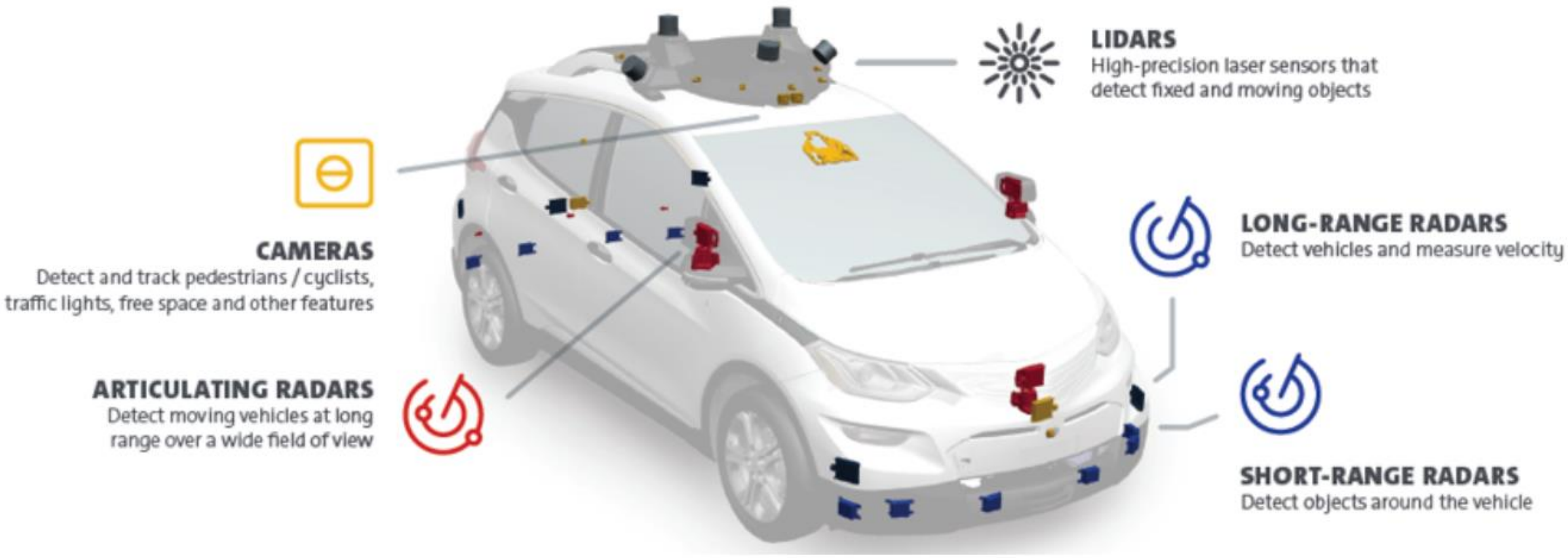
→ The crazyflie 2.1 hardware and software

Sensors equipped in an autonomous car

To enable 3D vision

→ Providing knowledge of the environment's geometry

→ Combining multiple different 3D sensor types



Enabling 3D Vision

Active and passive sensors

→ **Active**: emit sound/radio/light waves

- `Ultrasonic : short range (5 m)`
  `⇒ Parking, blind spot detection`
- `Radar : long range (300 m), low resolution`
  `⇒ Adaptive cruise control (ACC)`
- `Lidar : long range (100 m), mid resolution`
  `⇒ Self-driving vehicle prototypes (expensive)`

→ **Passive** : do not emit any waves

- `Stereo cameras: mid range (50 m)`
  `⇒ Cheap & high resolution, but require processing to obtain depth; accuracy depends on distance/texture`

Stereo cameras

Disparity Estimation to create 2.5D maps (not 3D yet)

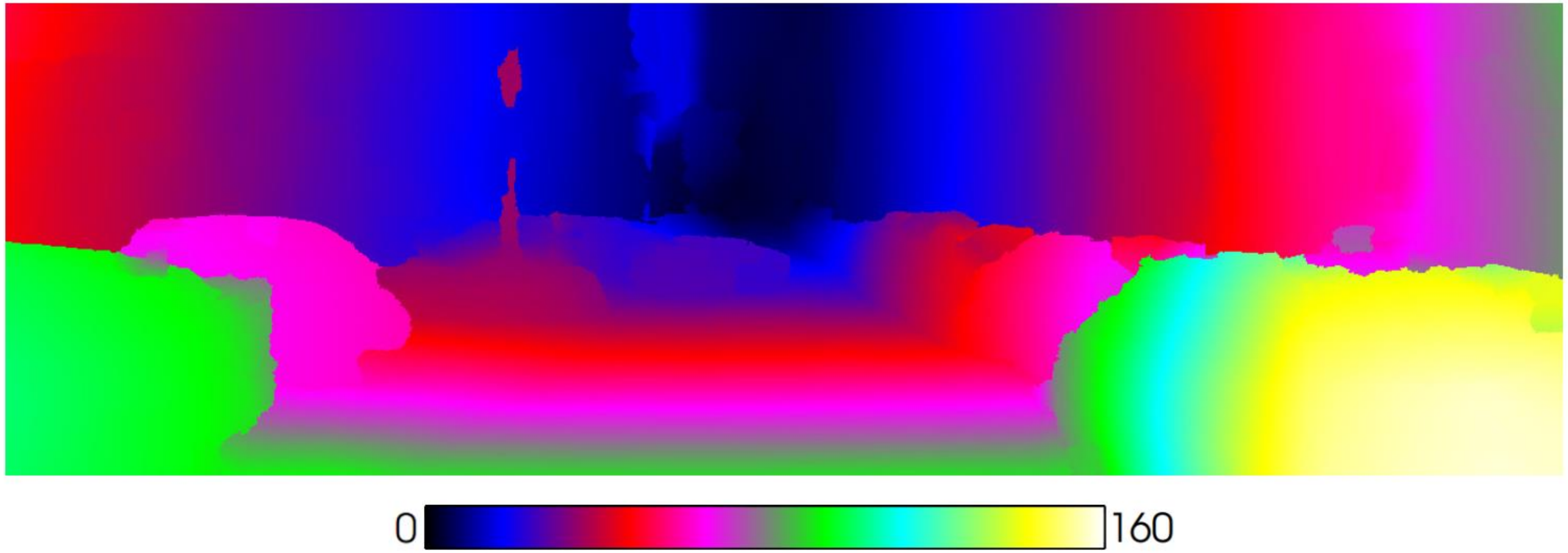
→ Input: Images of the 2 (laterally) displaced cameras (captured at the same time)

→ Output: Horizontal displacement (=disparity) per pixel (disparity map)

→ Disparity (per pixel) is anti-proportional to scene depth : we can create a 2.5D map

Stereo cameras

Disparity Estimation to create 2.5D maps (not 3D yet)



→ Input: Images of the 2 (laterally) displaced cameras (captured at the same time)

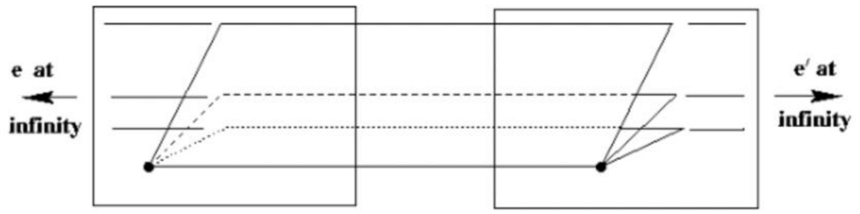
→ Output: Horizontal displacement (=disparity) per pixel (disparity map)

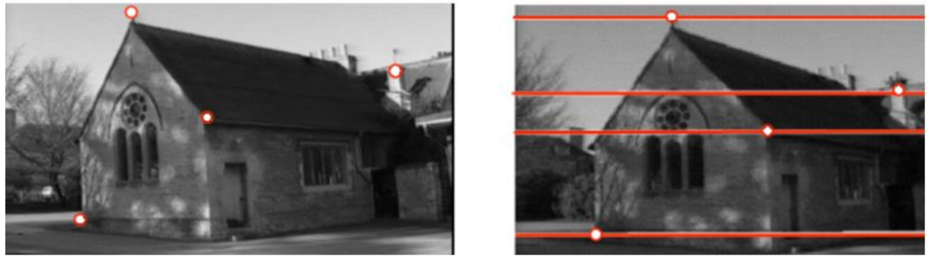
→ Disparity (per pixel) is anti-proportional to scene depth : we can create a 2.5D map

Constructing a dense 2.5D disparity map from 2 images

Pipeline

1. Calibrating cameras
2. Rectifying images
   - Compensating the fact that cameras are not perfecly aligned
3. Compute disparity map
4. Remove outliers
5. Obtain depth from disparity (triangulation)
6. Constructing 3D model (volumetric fusion)

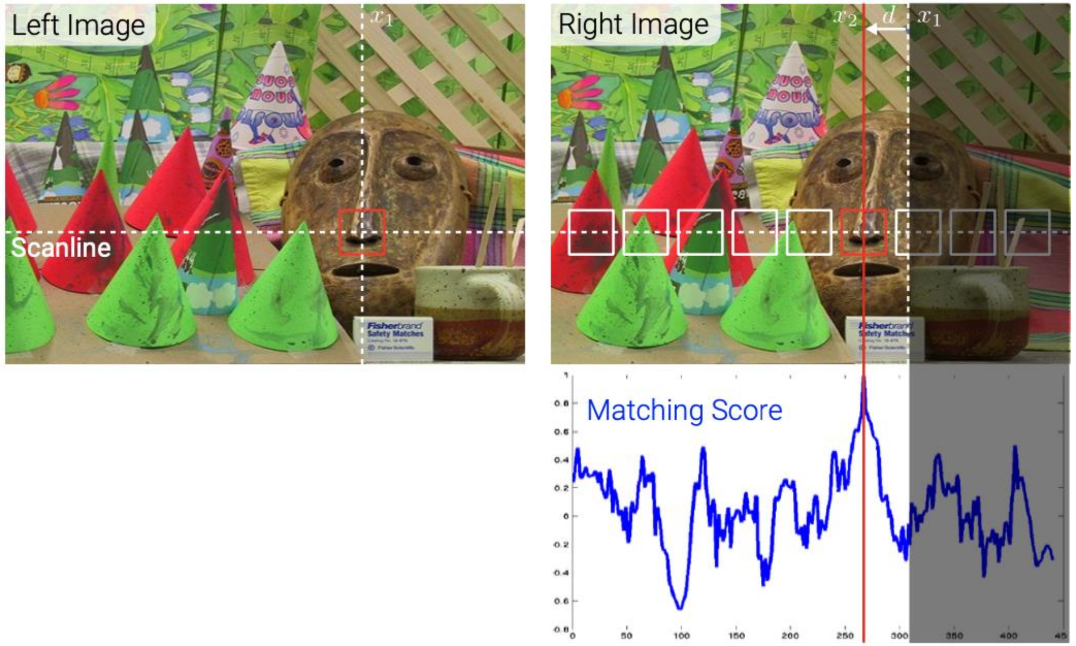


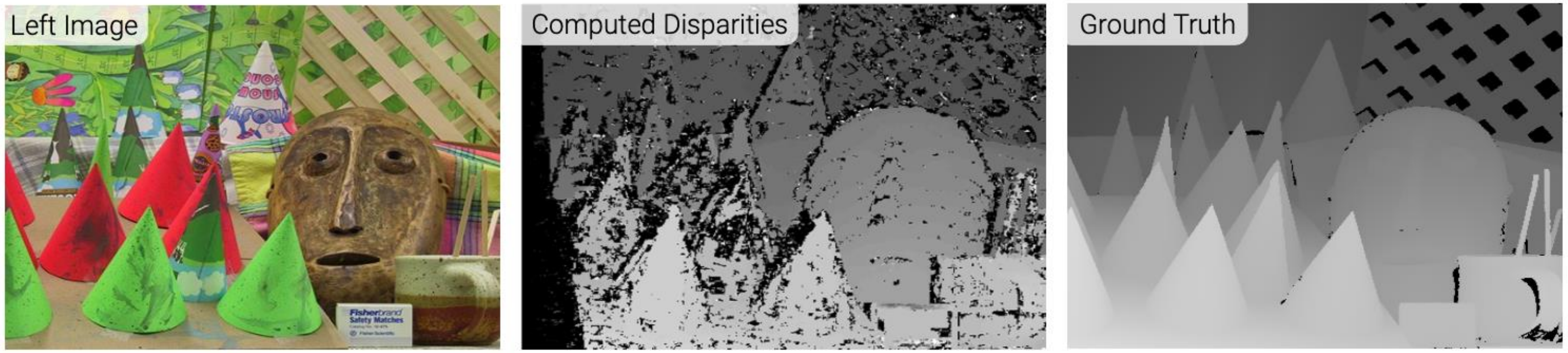


Constructing a dense 2.5D disparity map from 2 images

Pipeline

1. Calibrating cameras
2. Rectifying images
3. Compute disparity map
   - Stereo matching
4. Remove outliers
5. Obtain depth from disparity (triangulation)
6. Constructing 3D model (volumetric fusion)





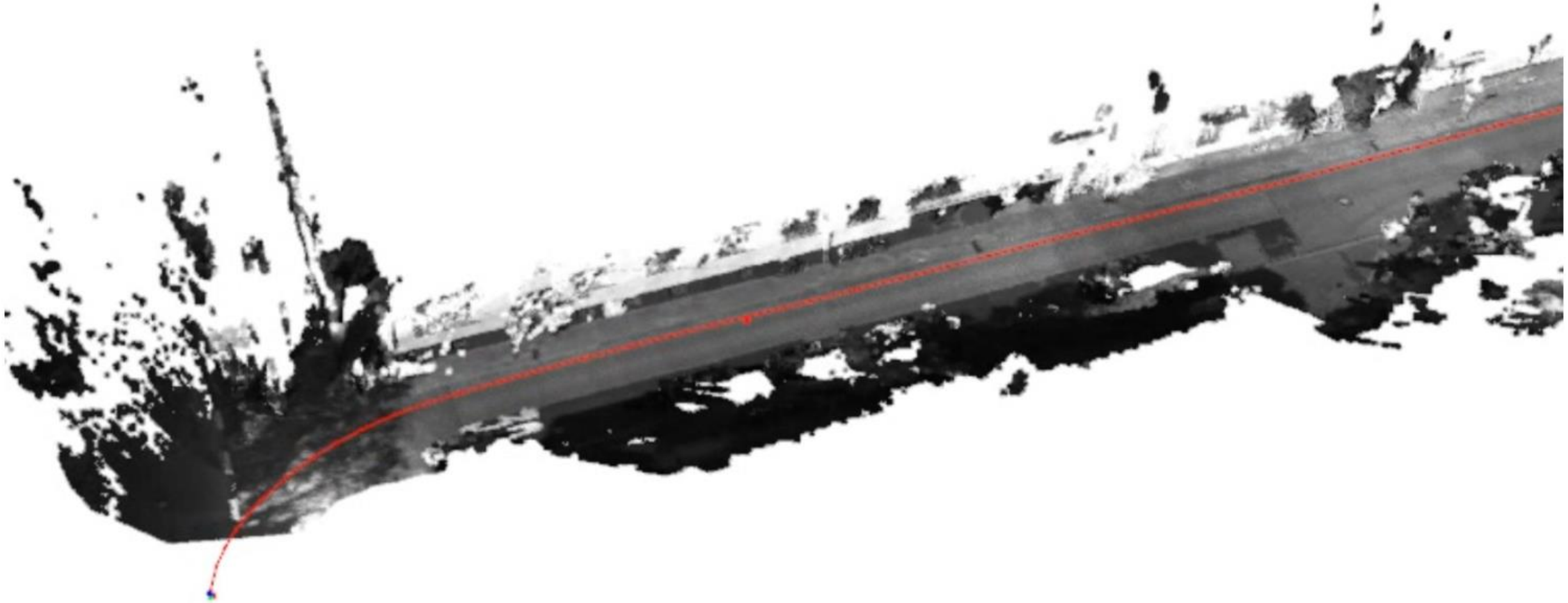
Constructing a dense 2.5D disparity map from 2 images

Pipeline

1. Calibrating cameras
2. Rectifying images
3. Compute disparity map
4. Remove outliers
5. Obtain depth from disparity (triangulation)
6. Constructing 3D model (volumetric fusion)

3D reconstruction using stereo cameras

Multiple 2.5D maps fused into a 3D reconstruction



Outline

Today’s lesson

→ Sensors

→ **The embedded systems equipped in cars**

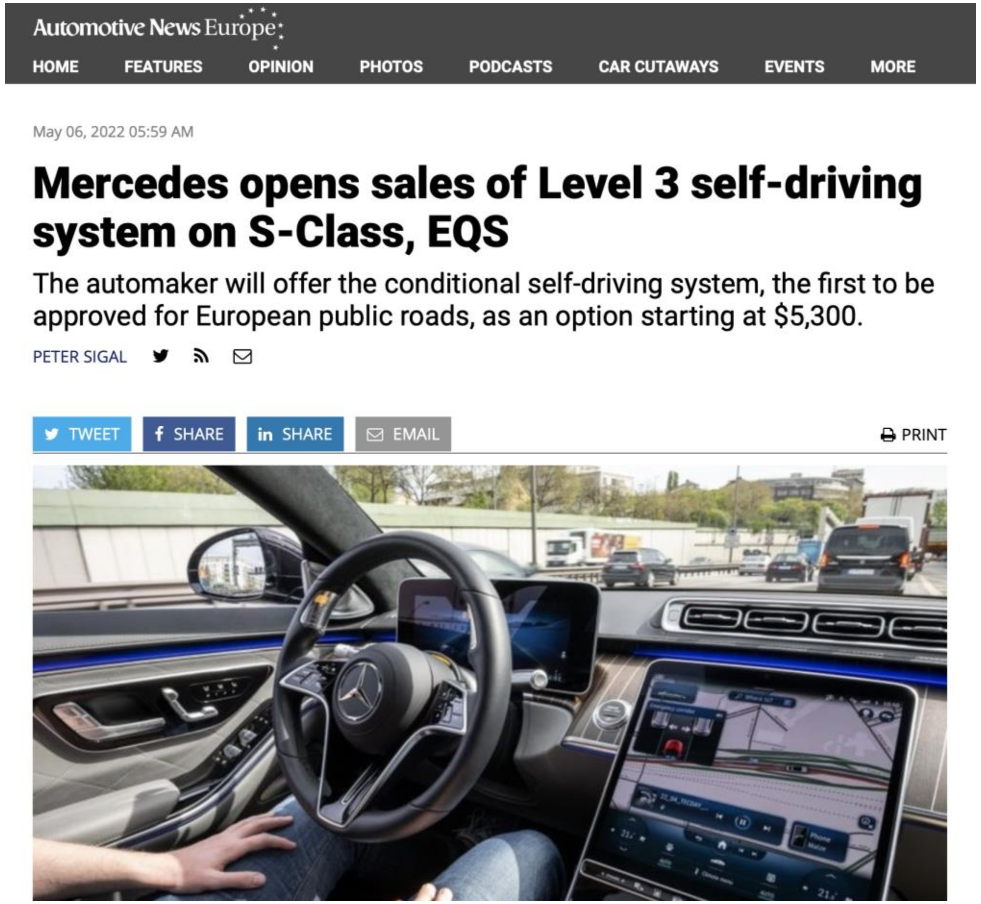
→ The crazyflie 2.1 hardware and software

Companies providing Level2+ (and Level 3)

And companies providing HW/SW to those companies

→ Tesla:
- “The first ones”
- Providing Level 2 (certified)
- And claiming higher levels (Level 5) being tested in shadow mode.

→ Mercedes (S-class and EQS)
- First ones providing (2022) Level 3 certified car in Europe
- Openly announced partnership with NVIDIA



Automotive News Europe

HOME FEATURES OPINION PHOTOS PODCASTS CAR CUTAWAYS EVENTS MORE

May 06, 2022 05:59 AM

Mercedes opens sales of Level 3 self-driving system on S-Class, EQS

The automaker will offer the conditional self-driving system, the first to be approved for European public roads, as an option starting at $5,300.

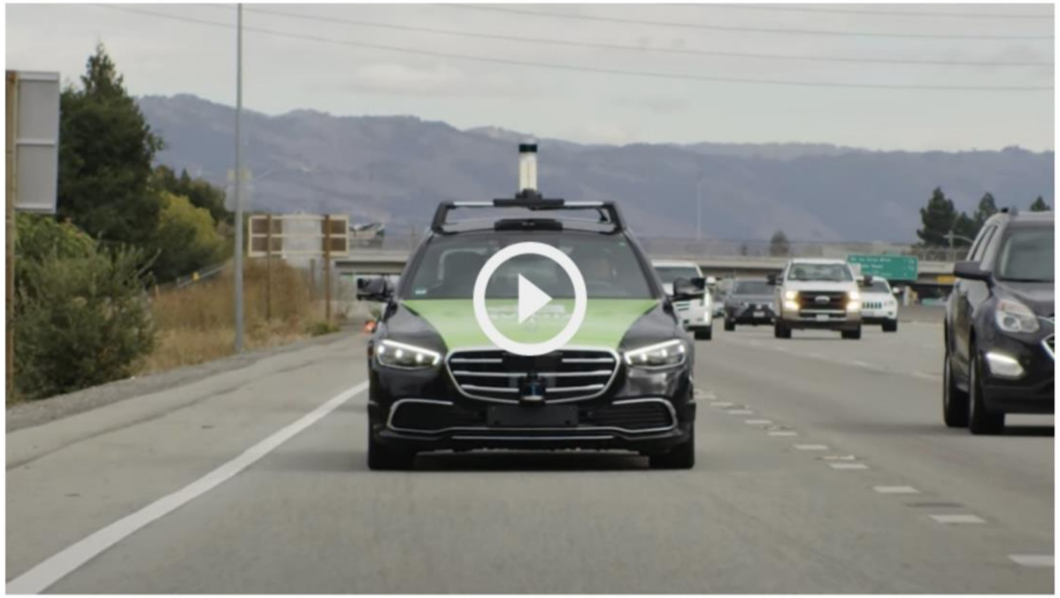
PETER SIGAL

TWEET SHARE SHARE EMAIL PRINT

https://www.forbes.com/sites/bradtempleton/2019/04/29/teslas-shadow-testing-offers-a-useful-advantage-on-the-biggest-problem-in-robocars/?sh=798773cf3c06
https://europe.autonews.com/automakers/mercedes-opens-sales-level-3-self-driving-system-s-class-eqs

Driving with NVIDIA’s self-driving car

Mercedes S Class models (Level2+) powered by NVIDIA



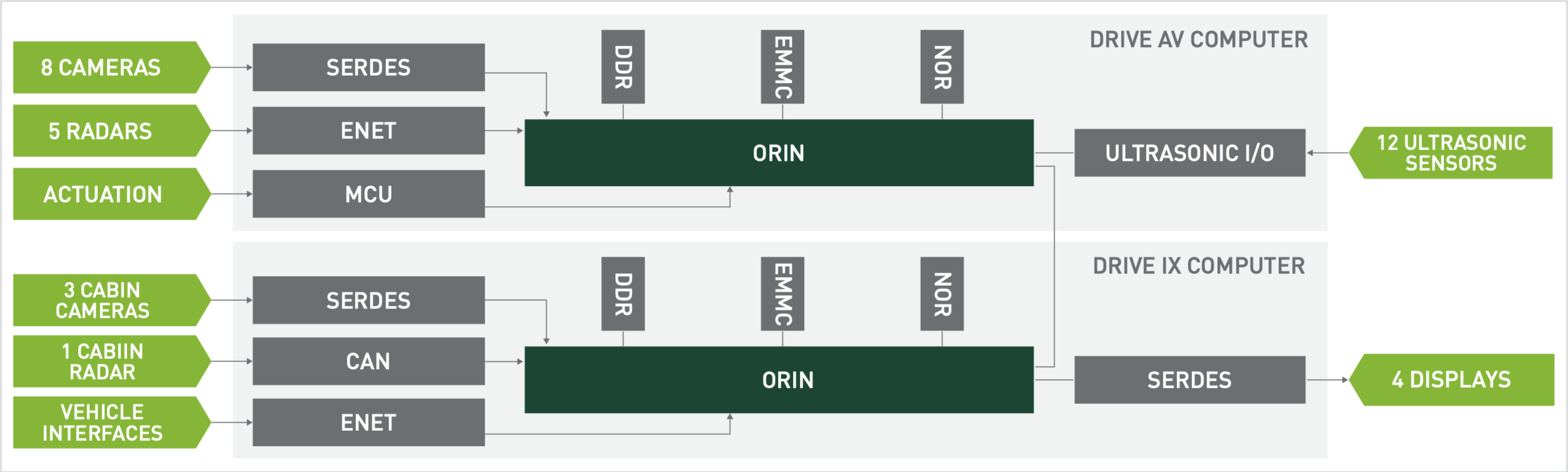
Ride with NVIDIA’s Self-Driving Car

https://blogs.nvidia.com/blog/2020/09/02/mercedes-benz-s-class-ai-cockpit-powered-nvidia/
https://www.nvidia.com/en-us/self-driving-cars/drive-chauffeur/

”Embedded processing” in autonomous cars

NVIDIA Drive : for Level 2+ autonomous driving

→ Working on NVIDIA Drive Orin SoC

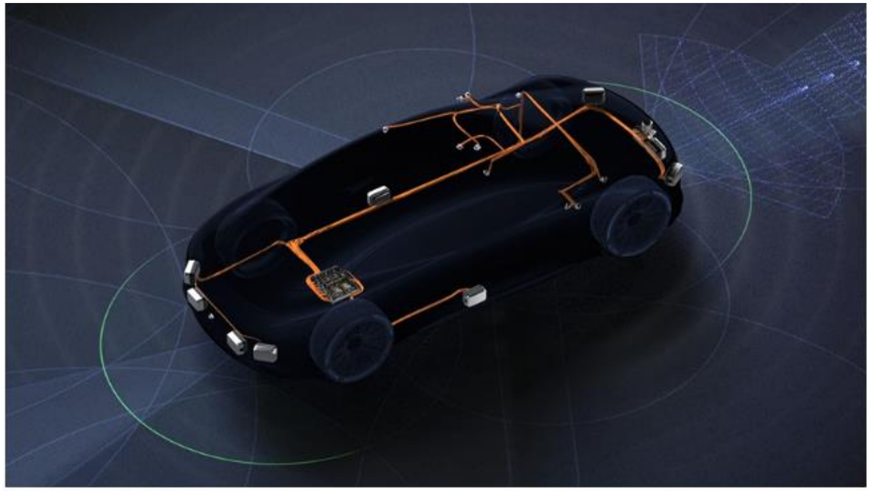


https://www.nvidia.com/content/dam/en-zz/Solutions/self-driving-cars/drive-platform/auto-print-drive-product-brief-final.pdf

Hardware capabilities

Drive AGX Orin DevKit

Component	Description	Details
One Orin SoC	12 Cortex-A78A CPU	
	Deep Learning Accelerators (DLA)	87 TOPS (INT8)
	NVIDIA Ampere architecture-class integrated GPU	167 TOPS (INT8) | 5.2 TOPS (FP32)
	Programmable Vision Accelerators (PVA)	2048 GMACS (INT8) | 512 GMACS (INT16)
	Image Signal Processor (ISP)	1.85Gigapixels/s
	Video encoder	Up to 1.0GPix/s (H.265)
	Video decoder	Up to 1.9GPix/s
	Memory bandwidth (256-Bit LPDDR5)	~200GB/s
DRIVE AGX System I/O	Camera	90Gb/s over 16x GMSL(R) ports
	LIDAR/Radar	~30Gb/s over Ethernet
	Vehicle IO	6 CAN interfaces
Included Accessories	Network Interface Adapter	Converts 1GbE Rugged Auto HMT-D IO to Standard Ethernet RJ45 IO
	Other cables	Connects the DRIVE AGX system to a host development computer



https://developer.nvidia.com/drive/hyperion

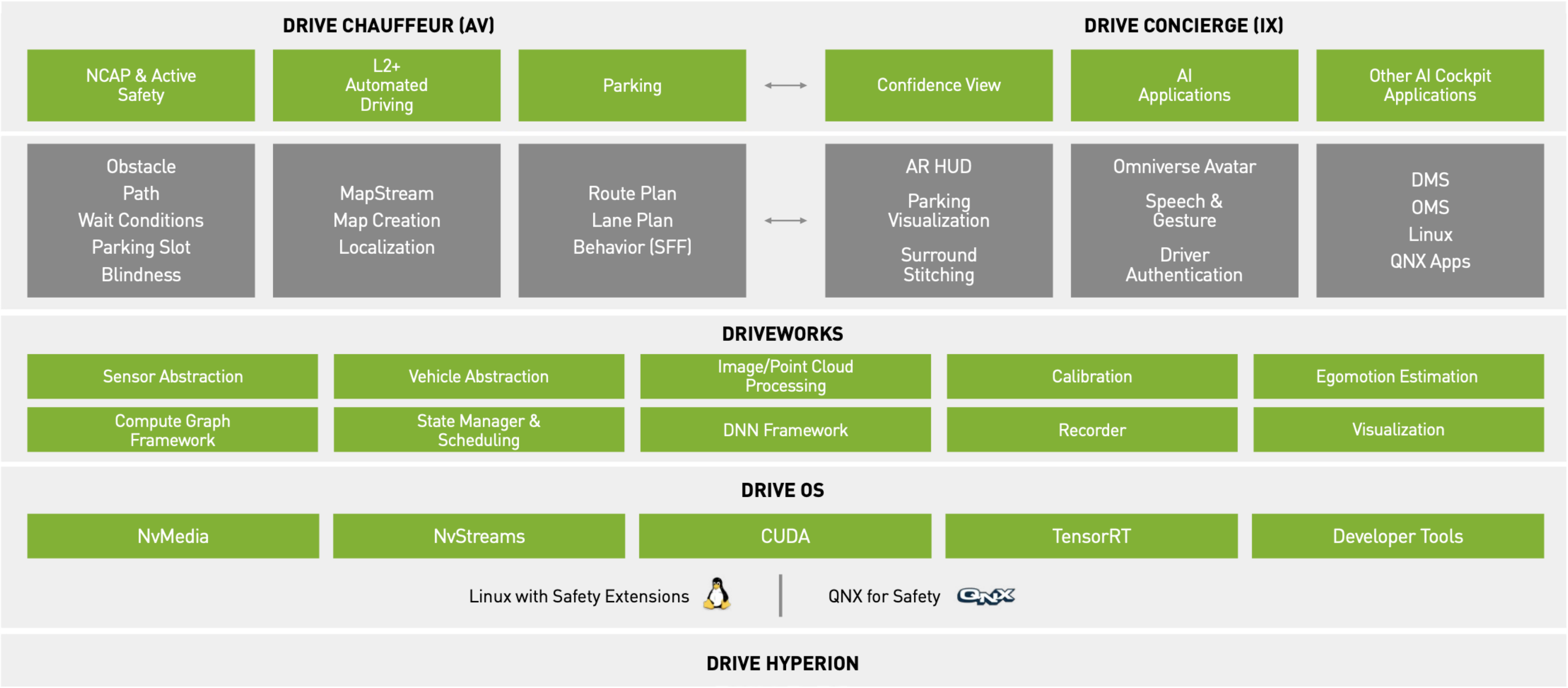
Supported software features

All supported features in a Mercedes Level2+ car today

Active Safety	Highway Driving	Urban Driving and Parking	Cockpit
> Automatic Emergency Braking > Automatic Emergency Steering > Lane Departure Warning > Lane Keeping Assist > Blind Spot Monitoring > Traffic Sign Assist > Stop Sign and Traffic Light Assist	> Adaptive Cruise Control > Lane Centering > Driver-Initiated Lane Change > Automatic Lane Change > Lane Fork to Follow Route (Highway Interchange) > Lane Merge > Speed Adaptations for Curves and Speed Limit Changes	> Traffic Light Stop at Intersection > Protected Intersection Turn > Unprotected Intersection Turn > Roundabout > Yield to Pedestrian Crossing > Parking Assist > Remote Parking	> Confidence View > Augmented Reality > AR HUD > Parking Visualization > Fused Awareness > Conversational AI > Driver/Occupant Monitoring > Activity Monitoring

NVIDIA Drive SDK

For NVIDIA Drive Orin SoC



Sommaire

Cours d’aujourd’hui

→ Sensors

→ The embedded systems equipped in cars

→ **The crazyflie 2.1 hardware and software**

Crazyflie 2.1 – The drone itself

Main HW characteristics

→ Crazyflie 2.1 nanodrone
- Main processor: ARM Cortex-M4 @168MHz, 192kb SRAM, 1Mb
- Radio: ARM Cortex-M0, 32Mhz, 16kb SRAM, 128kb flash
  - 2.4GHz ISM band radio
- Micro-USB connector

→ IMU specification
- 3 axis accelerometer / gyroscope (BMI088)
- High precision pressure sensor (BMP388)

→ Flying time of 7 minutes (charging time 40min)
- Maximum payload of 15g (the drone’s weight is 27g)
- Labs: careful! Keep your done charged!

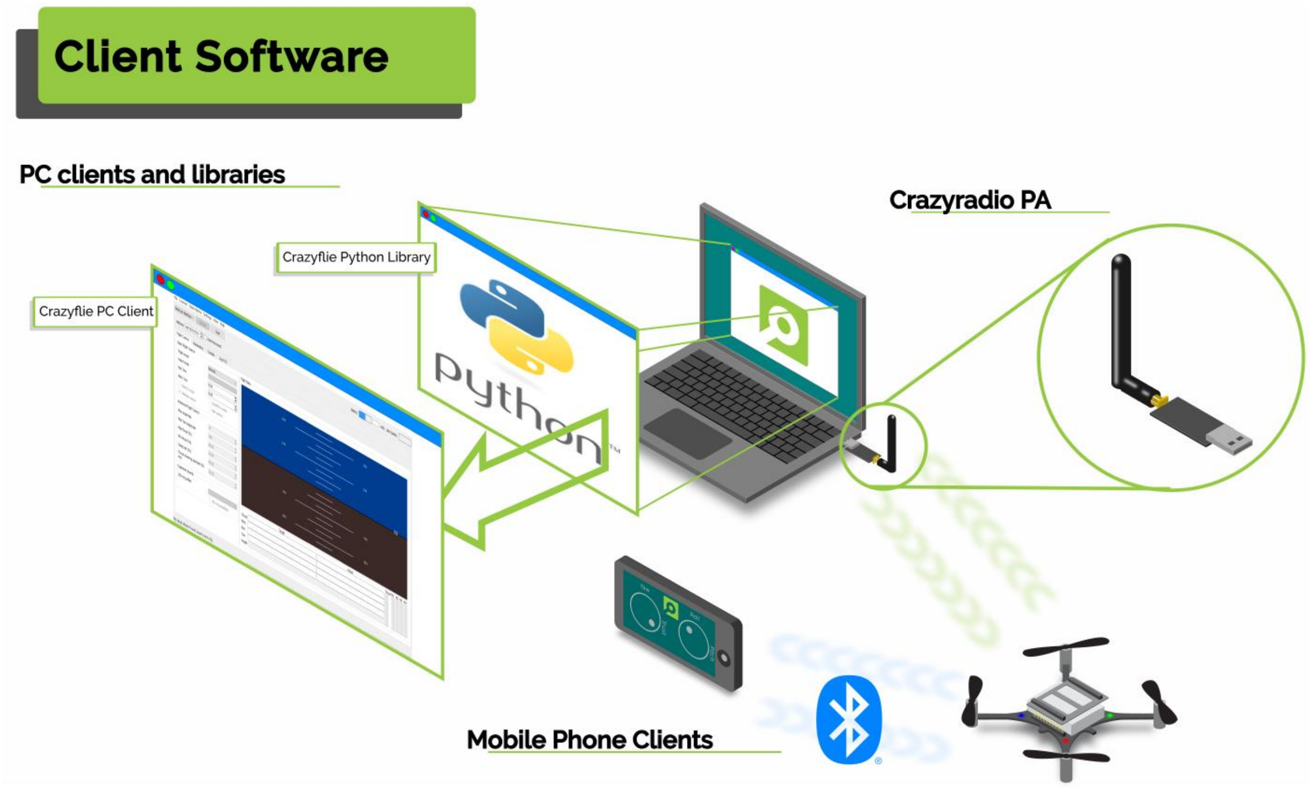
→ Open-source !

Programming and controlling the Crazyflie 2.1

Additional HW (Crazyradio PA) and SW needed to program the drone

→ To program and control the crazyflie you need:
- Crazyradio PA (USB dongle) connected to your PC
- Initially you can control the crazyflie with a phone (or remote control)

→ The drone uses a ROS-based operating system



Adding new shields (decks) to enable autonomous flying

The AI-deck and the Flow deck

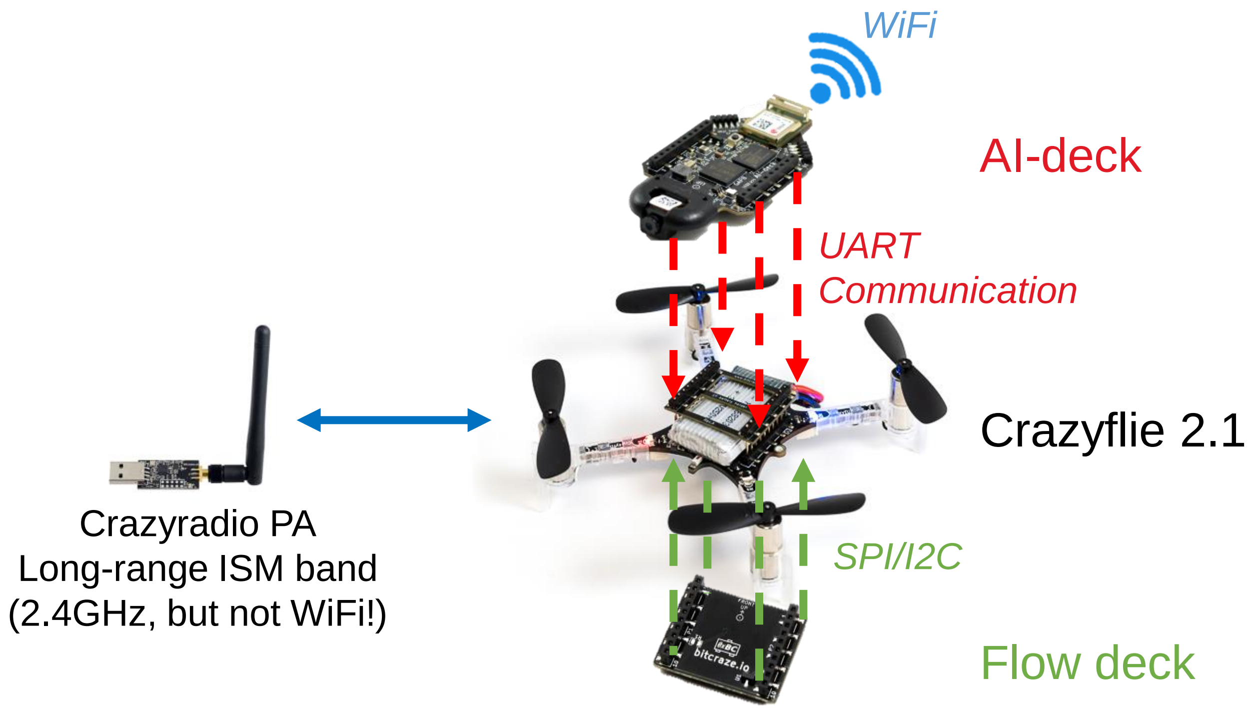
→ AI-deck:

- Camera, WiFi and AI accelerator
- Attached on top of the Crazyflie 2.1

→ Crazyflie 2.1

→ Flow deck v2

- Height control for more stable flying
- Attached to the bottom of the drone, sensor looking down



AI-deck specifications

Adding a camera, WiFi and acceleration for AI tasks

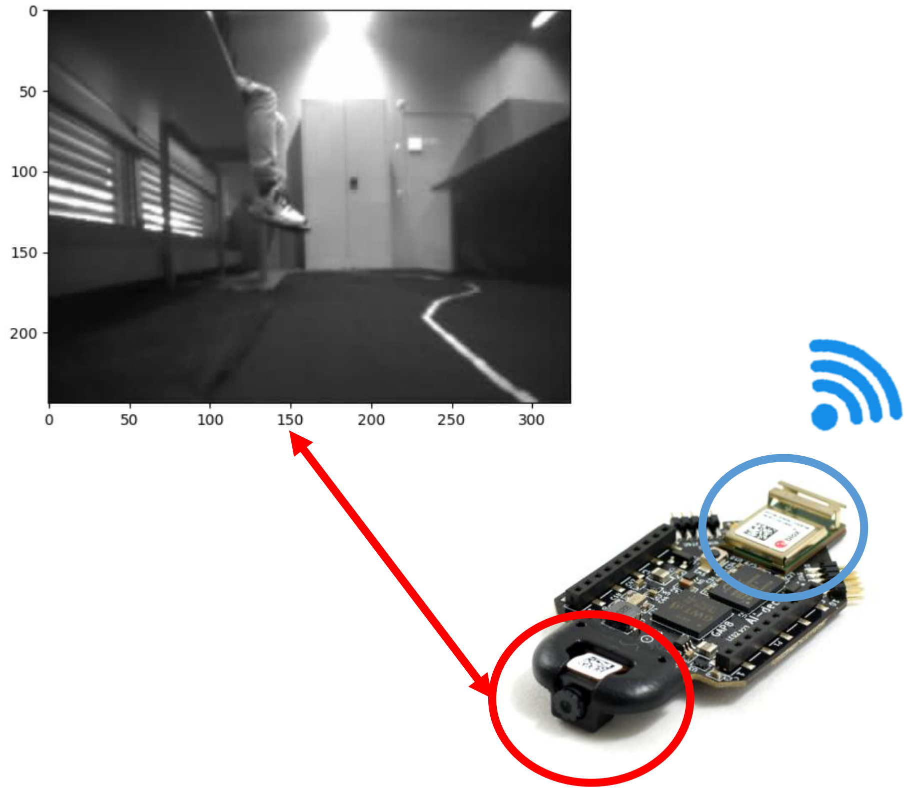
→ The GAP8 chip: a RISC-V based accelerator
- GAP8 – Ultra low power 8+1 core RISC-V
- 250MHz internal clock (max)
- ~8Gflops at tens of mW
- A Convolutional Neural Network accelerator (HWCE)

→ Himax HM01B0 – Ultra low power 320×320 monochrome camera.

→ WiFi Connection (for streaming camera images)
- 512 Mbit HyperFlash and 64 Mbit HyperRAM
  ESP32 for WiFi

→ UART connection to the Crazyflie

→ Based on the PULP cores from ETHZ



https://www.bitcraze.io/documentation/hardware/ai_deck_1_1/ai_deck_1_1-datasheet.pdf
https://greenwaves-technologies.com/wp-content/uploads/2021/04/Product-Brief-GAP8-V1_9.pdf

The “brain” inside the AI-deck

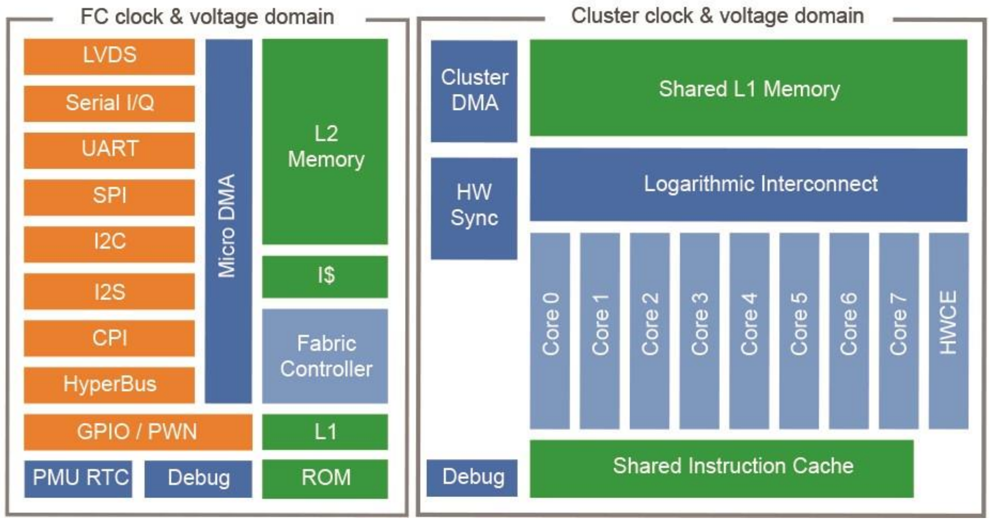
GAP8 hardware and software architecture

→ Software

- PULP OS
- Or FreeRTOS (used in this course, better support)

→ Hardware

- GAP8 is based on the PULP RI5CY core
  - 4-stage in-order 32-bit RISC-V cores
  - 8 cores for acceleration + 1 core for control
  - No FP unit !



- HWCE (hardware convolution engine) : for running CNNs
  - Input and output pixels and weights must be 16-bit, 8-bit or 4-bit fixed-point numbers

https://gwt-website-files.s3.amazonaws.com/gap8_datasheet.pdf
https://pulp-platform.org/docs/ri5cy_user_manual.pdf

Improving Crazyflie’s positioning

Optical navigation

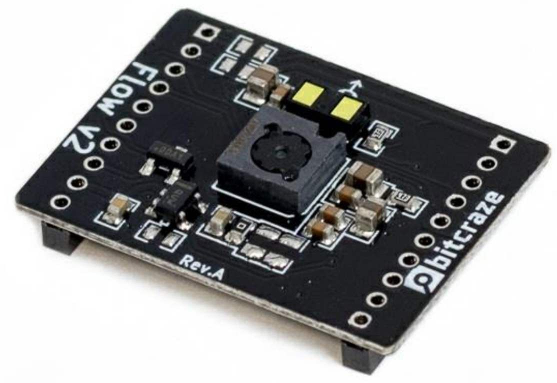
→ VL53L1x ToF sensor to measure distance to the ground:

- up to 4 meters within a few millimeters

→ PMW3901 optical flow sensor measures movements on the ground

→ Goal: create a flying 3D robot that can be pre-programmed to fly distances in any direction

- Very stable flying platform



TODOs for today

Exercises

1. Analysis of the NVIDIA Drive solution in detail
    - https://www.nvidia.com/content/dam/en-zz/Solutions/self-driving-cars/drive-platform/auto-print-drive-product-brief-final.pdf
    - Hardware, software and simulation solutions

2. Analysis of the Crazyflie
    - All components and shield
    - RI5CY cores and GAP8

Lab material

REDS
Institut
Reconfigurable
and Embedded
Digital Systems

Hes·so
Haute Ecole Spécialisée
de Suisse occidentale