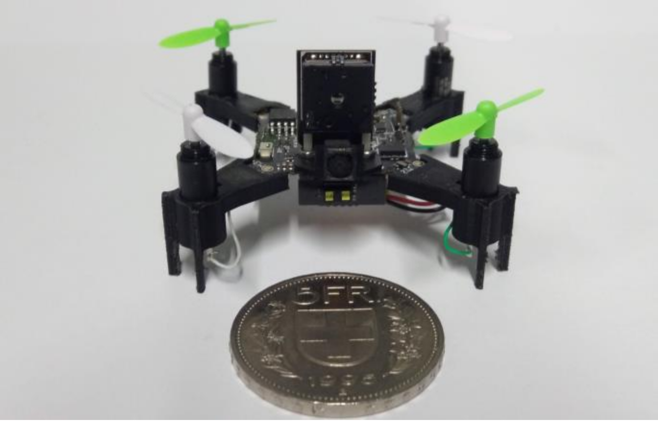
DESIGN, AUTOMATION & TEST IN EUROPE
DATE 21
01 – 05 February 2021 · virtual conference
The European Event for Electronic System Design & Test

ETH zürich
FNSNF
SCHWEIZERISCHER NATIONALFONDS ZUR FÖRDERUNG DER WISSENSCHAFTLICHEN FORSCHUNG
SPARK: 190880
BRIDGE Discovery: 187087

Fünfliber-Drone

A Modular Open-Platform 18-grams Autonomous Nano-Drone



Hanna Müller*, Daniele Palossi*†, Stefan Mach*, Francesco Conti‡, and Luca Benini*‡
*Integrated Systems Laboratory - ETH Zurich, Switzerland
†Dalle Molle Institute for Artificial Intelligence - University of Lugano and SUPSI, Switzerland
‡Department of Electrical, Electronic and Information Engineering - University of Bologna, Italy Email:
hanmuell, dpalossi, smach, fconti, lbenini@ethz.ch

Why autonomous (nano-)UAVs?

Rescue Missions

Entertainment

Monitoring

Inspection

Why nano-sized?

- More agile
- Less intrusive
- Enhanced safety
- Reduced cost

Autonomous nano-UAV - Challenges

Challenging trade-off between power consumption[1] and onboard processing[2]

- Autonomous navigation bound: >10frames/s
- Compuation and sensing < 10% of power

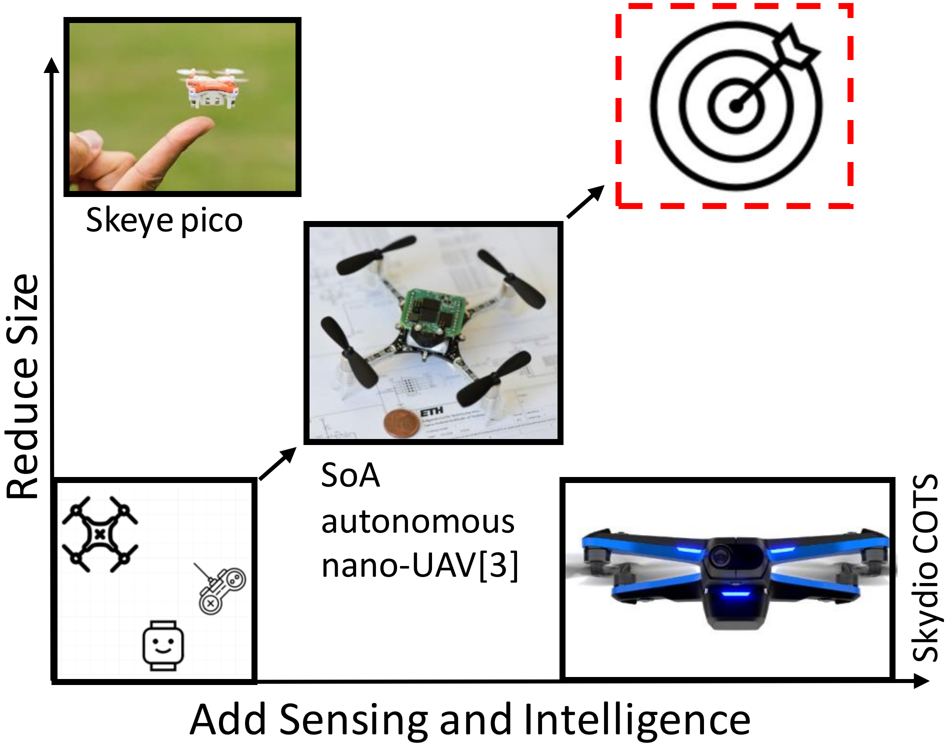
Vehicle class	Ø : Weight [cm:Kg]	Power [W]	Onboard Device
standard-size	≥ 50 : ≥ 1	≥ 100	Desktop
micro-size	~25 : ~0.5	~50	Embedded
nano-size	~10 : ~0.01	**~5**	MCU
pico-size	≤ 2 : ≤ 0.001	≤ 0.1	ULP

An autonomous nano-UAV has to process >10frames/s in <500mW

Nano-UAVs require *“Extreme near-sensor processing”*

[1] R. J. Wood et al., “Progress on ‘Pico' Air Vehicles”, Cham: Springer International Publishing, 2017.
[2] A. Briod et al., “Optic-flow Based Control of a 46g Quadrotor”, IROS 2013.

Smaller and Smarter



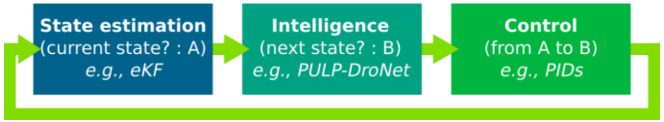
Our contributions:

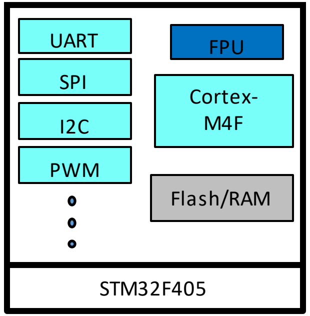
- 2x smaller than SoA
- Sensor-rich
- Single SoC based
  - Components use less space
  - Simpler design
  - Less communication
  - More energy-efficient

[3] D.Palossi et al., "A 64mW DNN-based Visual Navigation Engine for Autonomous Nano-Drones", IEEE Internet of Things Journal 2019

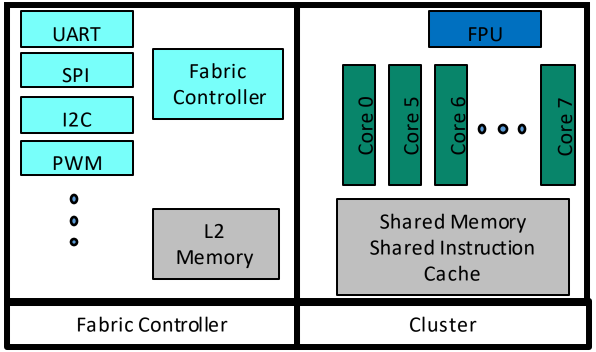
Implementing Intelligence

Autonomous robot lifecycle:





Crazyflie
- Open soft/hardware



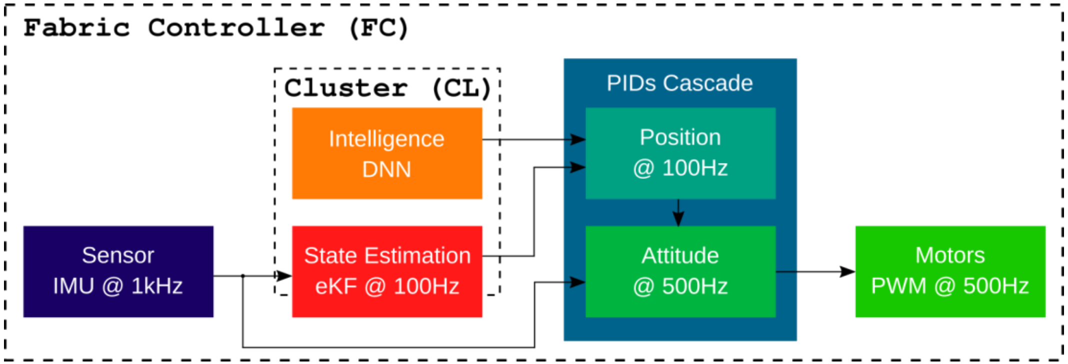
PULP
Parallel Ultra Low Power

Mr. Wolf

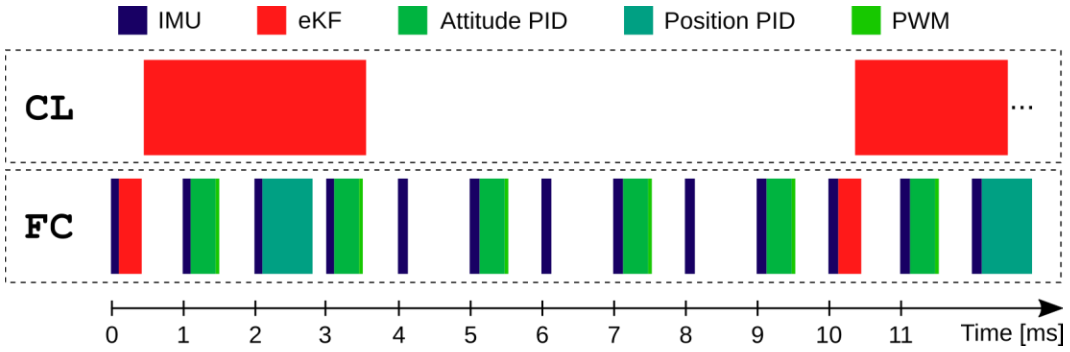
Cores	1 ARM Cortex-M4F	1x RV42IMC + 8x RV32IMFCXpulp
Frequ.	168MHz	up to FC@450MHz/Cluster@350MHz
RAM	192kB	L1: 8kB (FC) 64kB (Cluster), L2: 512kB
FPU	Yes	FC: No, Cluster: Yes
OS	FreeRTOS	pulpOS (no preemption)

Software Architecture

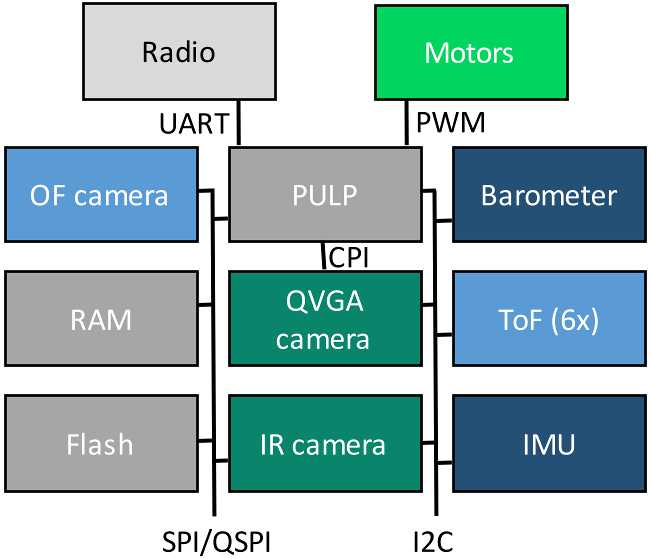
- **FPU only on cluster**
  - EKF on cluster
  - Soft-float PIDs

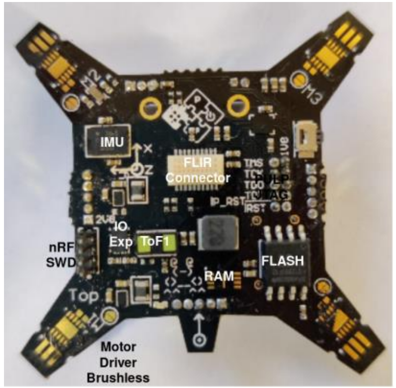


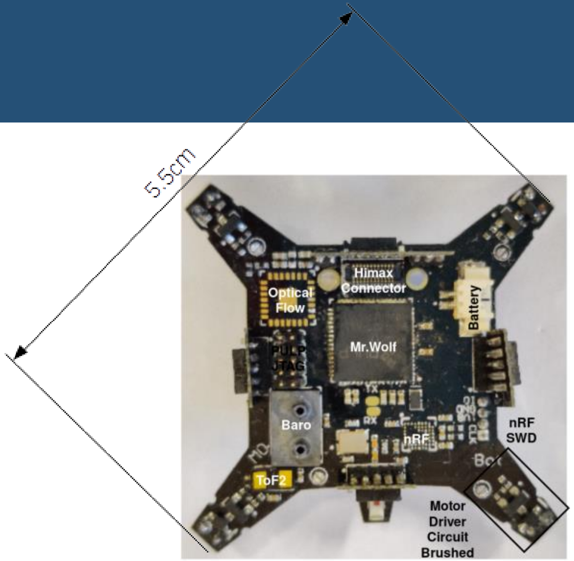
- **Scheduling**
  - PIDs on FC
  - Example @100MHz

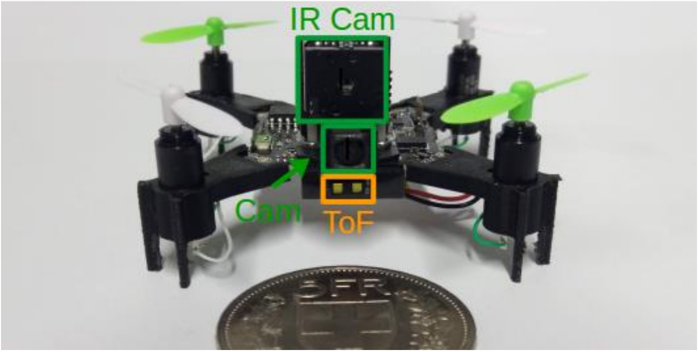


Hardware Architecture



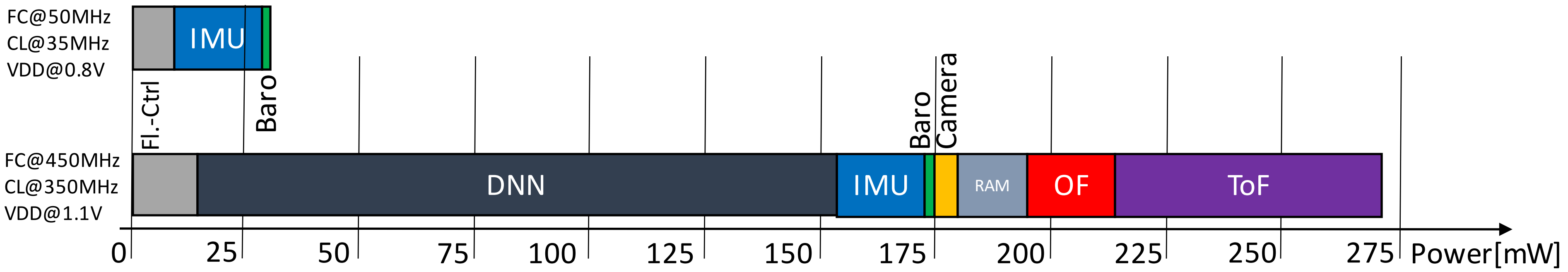






Results

- **Only flight controller: 9mW (with sensors 30mW)**
- **Full computational capability: 153mW (with sensors 271mW)**
  - Flight controller uses 11% of CL, 9% of FC
  - Navigational engine can run at up to 18Hz
  - Computation uses 6% of overall power



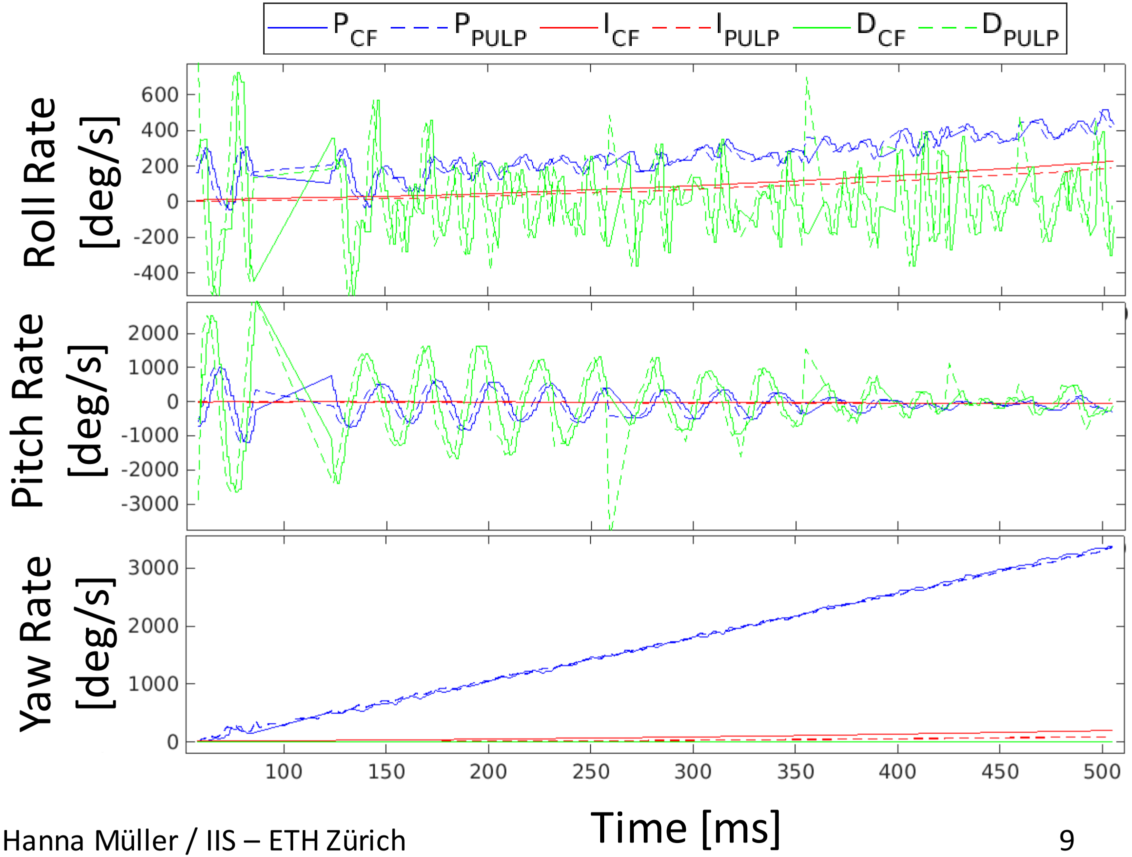
Results - Accuracy

Setup

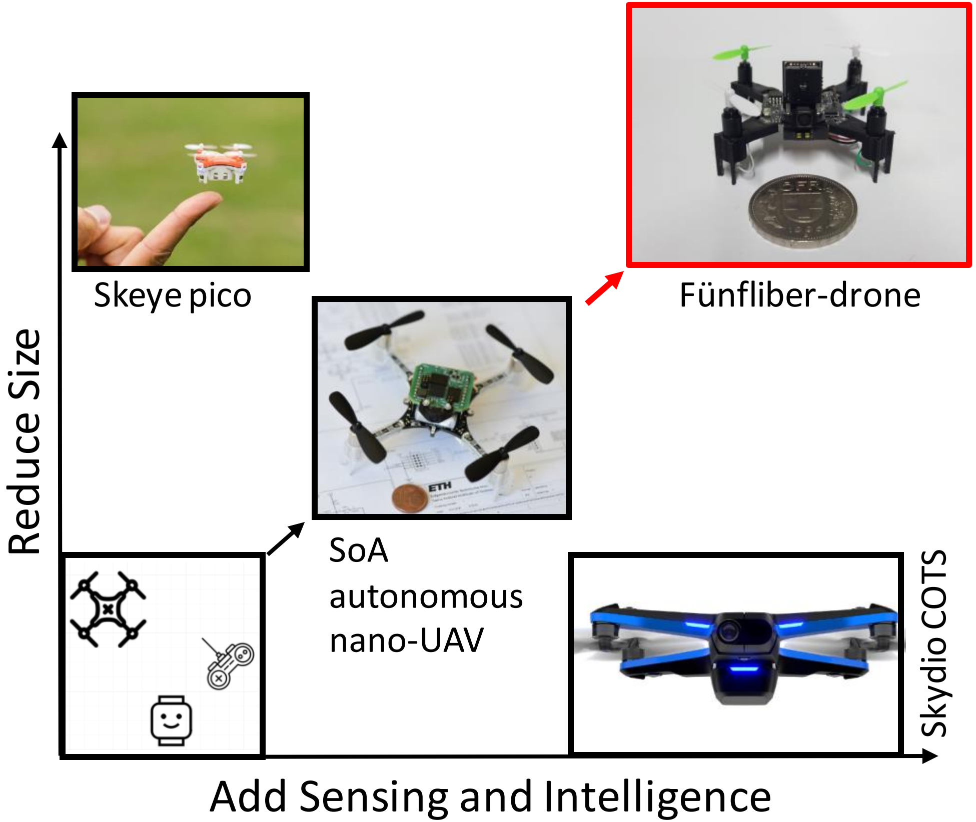
- Log ground truth from Crazyflie
- Feed sensor (IMU) inputs to PULP
- Log output from PULP

Example: PID output

- GT tracked nicely
- Spikes at missed datapoints



Conclusion



- 18grams, 7.2cm diameter
- Sensor-rich
- Single SoC based;
  - 9mW for minimal flight controller
  - 271mW for SoA navigational DNN incl. flight controller
- Open-source

https://github.com/pulp-platform/fuenfliber