- APROVIS3D -

Analog PROcessing of bioinspired VIsion Sensors for 3D reconstruction

Document Reference:
Title: Deliverable D.4.1 Report on digital/analog components + D4.2 Report on the software/firmware architecture
Contractor: **Partners**
Prepared by: **ETHZ, UCA, NTUA, IMSE**
Document Type: **Deliverable**

Version: **1**		Pages: **1**
Classification: **External document**		

Document Track

Version	Date	Remarks and Authors
1.0	01/05/2022	First Draft (Michele Magno - ETH)
1.2	21/02/2023	Final Version all partners

Authors

	Role / Function	Name	Organisation
Prepared by	Partner	George Karras	NTUA
Checked by	Partner	Teresa Serrano	IMSE
Released by	WP4 Leader	Michele Magno	ETHZ
Approved by	Coordinator	Jean Martinet	UCA

1 Introduction

Neuromorphic computing has seen tremendous research interest in recent years and promises to be a key technology for the next generation of processing systems. Neuromorphic algorithms, in particular Spiking Neural Networks (SNN) have been successfully applied to many problems especially in combination with event-based camera. Then the majority of work include machine learning tasks such as low latency computer vision for autonomous vehicles and drones. Real-world applications tend to belong to the former category, and the successful use of SNN on the edge is dependent on two key factors: Flexible and energy-efficient accelerators to which the SNN can be mapped, and their integration in a sensor node which collects the data to be processed.

SNN accelerators have been developed by industrial actors as well as the research community. IBM released TrueNorth a fully digital neurosynaptic computer with 1 million neurons connected by 256 million synapses. ARM released Spinnaker with thousands parallel core to run different range of SNN. Intel has released Loihi and its successor Loihi 2 with 131K and 1M neurons, respectively. The Loihi family of chips also integrate x86 cores for management and general-purpose computing, making them complete systems-on-chip. These industrial designs are distinguished by their large size (TrueNorth: 430 square millimeter), Loihi:
60 square millimeter, Loihi 2: 31 square millimeter), advanced software support and the capabilities to map advanced SNN to a single chip as well as to connect multiple chips to map even larger applications.

This report focuses on the system integration aspect of the APROVIS3D project. In particular, the report focuses on the video sensors and their interface in the proposed intelligent system. The novel DVS sensor is discussed and presented as well as the sensors interface. Moreover, the task aims at representing and bringing close to the sensors' artificial intelligence, and in particular the selection of the neuromorphic platform that can run Spiking Neural Networks on a real system platform. After a literature investigation, we have decided to carry out an accurate evaluation of three different neuromorphic platforms, the Spinnaker from ARM, Loihi from Intel, and Kraken, a novel system on chip with a neuromorphic accelerator designed by ETH Zurich and embedded in a prototype embedded system.

The technical details of the hardware components used in this project are given here. The interfaces between these components are also described with a special focus on the required communication protocols.

The proposed system illustrated in this project relies on two modules:

1. Foveated Dynamic Vision Sensor (fDVS)
   a. The fDVS is developed at the Institute of Microelectronics Sevilla (IMSE)
   b. This sensor will be responsible for recording events of the coastline from the UAV
   c. A special communication protocol interfaces the sensor with the neuromorphic processor

2. Neuromorphic platform processing where 3 processor has been evaluated.
   a. Neuromorphic Hardware: SpiNNaker
      i. The SpiNNaker is developed by the Advanced Processor Technologies Group (APT)
      ii. This neuromorphic processor will be responsible for processing the incoming events and outputting some scene information (e.g. optical flow, depth estimation, coastline detection…)

    iii. It receives live input from the fDVS and can send its output either to an on-board host computer for further processing or directly to motor-controlling ESCs

b. Neuromorphic Hardware: Loihi

    i. The Loihi chip is developed from Intel Labs
    ii. This will be kept for comparison and evaluation of the algorithms implemented in 2b.
    iii. The plan is **not** to use Intel's Loihi in the closed-loop system but to evaluate is performance

c. Full embedded SoC with Neuromorphic accelerator.

    i. The SoC is developed as a research professor at ETH Zurich.
    ii. A full work prototype of an evaluation board that can host a DVS camera is working and it will be evaluated.

Finally, the report presents also the whole architecture of the drone that will be used as testbed of the demonstrators.

The main goal of this report is to present the selected hardware architecture of the UAV prototype that includes the DVS camera, the processing unit (and the selection of different processing units)

2 Foveated Dynamic Vision Sensors (fDVS):

EF-DVS developed by university of IMSE :

The first prototype of EF-DVS developed by IMSE in the APROVIS3D project has a 128x128 resolution. Events are coded with an additional bit sign which indicated if the pixel illumination has gone under a positive or negative change. Consequently, the sensor needs 15 bits to code its output event addressed in parallel and the same 20-pin parallel AER connector of previous prototypes has been used.

The protocol sends in a parallel way all the bit component of x and y addresses for each event and an **ack** and **req** signal with a 4-phase handshaking protocol. It is a fully parallel event read out with the same hand shaking protocol and logic levels than was used in previous DVS iterations.

For communication of the sensor output events, the sensor pcb contains a 20-pin parallel connector where addresses of up to 16 bits can be paralelly read-out using the 4-phase handshaking protocol which uses the Ack and Rqst signals. Figure XX shows the bottom and top down view of the parallel AER connector mounted in the sensor pcb.

Additionally, the EF-DVS prototype has been mounted in a PCB that contains an STM32 bit microcontroller able to timestamp the events and send them to a host PC through a USB interface.

Interface DVS-SpiNNaker:

For the DVS, a student at IMSE (Amirreza Yousezafdeh) developed a Spartan6 FPGA code to interface the DVS with the SATA connector of the SpiNN5 board (see Fig.1), and a code to program the Spartan6 of the SpiNN5 board to interface the events received by the SATA connector and convert them to the Spinnaker event format. These codes are available:

https://ieeexplore.ieee.org/document/8010303/media#media

The DVS also connected with a Spinn4 board using an Raggestone2 board (see Fig.2). The code for the protocol conversion was developed by the Manchester group:

http://spinnakermanchester.github.io/docs/spinn-app-8.pdf

https://github.com/SpiNNakerManchester/spio

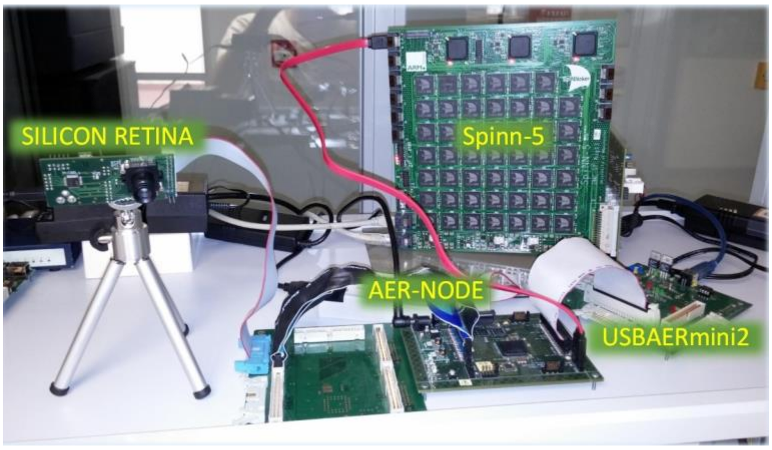


Fig.1:Interfacing Silicon retina to

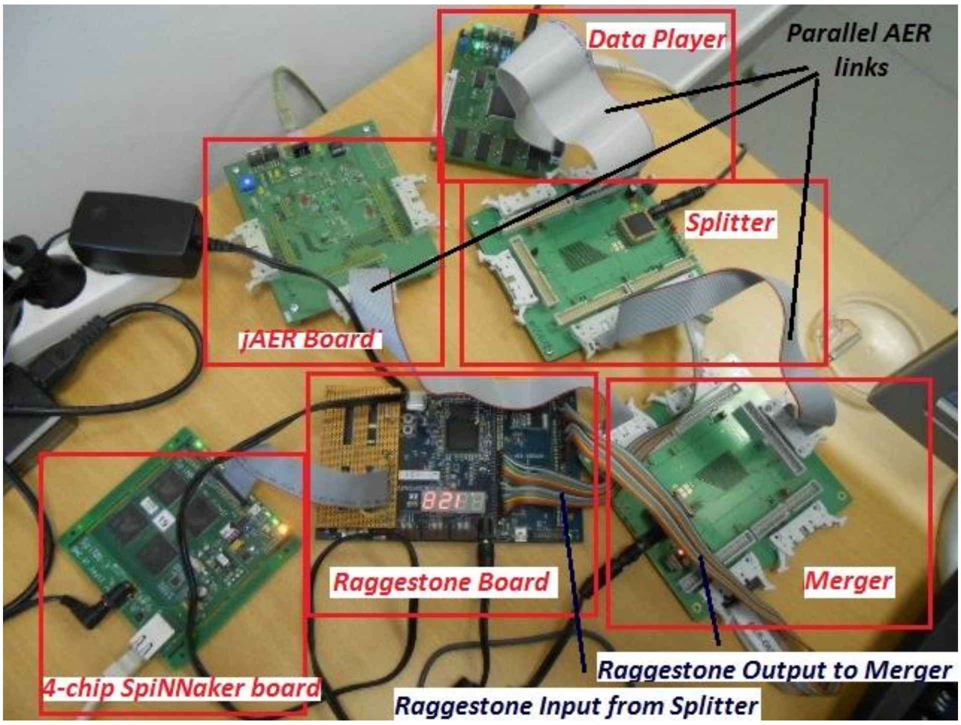


Fig.2:DVS-SpiNNaker4 connection using a commercial Raggedstone interfacing FPGA. This realises the conversion from the DVS protocol to the required

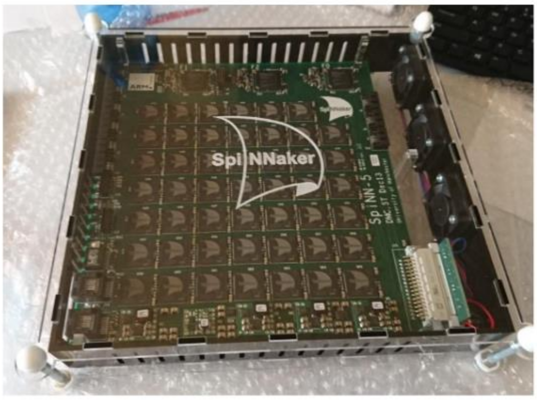
3 Neuromorphic processor Spinnaker (UCA)

SpiNNaker is an experimental hardware designed by a research group at the Department of Computer Science, University of Manchester, taking part in the Human Brain Project. Its vocation is to provide an effective hardware support for Spiking Neural Networks. There are other available solutions like TrueNorth, NeuroGrid, BrainScales, Loihi and others. SpiNNaker is a massively parallel, manycore supercomputer architecture that can simulates up to hundreds of thousands neurons with 518,400 processors. These processors possess 18 ARMx86 cores and are put together to simulate SNN by software and specific routing, on and between, chips.

SpiNNaker

→ 18 ARM968 processing cores per chip

SpiNN 3:
4 SpiNNaker chips

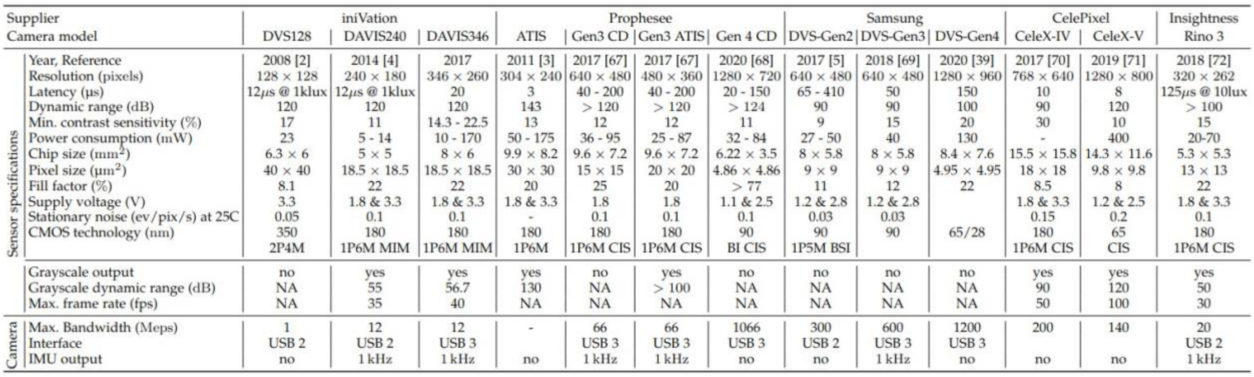


SpiNN 5:
48 SpiNNaker chips

We have used both single boards with 4 chips and 48 chips. We also used the online access to the SpiNNaker via the Human Brain Project portal project).

Simulation on SpiNN-3 and SpiNN-5 is done through a PyNN interface. PyNN is a simulation language for neural networks, described in the work of Davison et al. [2009]. Using python to code, we can then simulate neural networks on some of the accessible simulators, NEURON, NEST, and Brian, and on the SpiNNaker and BrainScaleS neuromorphic hardware systems.

Comparison of commercial or prototype event cameras. Values are approximate since there is no standard measurement testbed.



	Supplier	iniVation			Prophesee				Samsung			CelePixel		Insightness
	Camera model	DVS128	DAVIS240	DAVIS346	ATIS	Gen3 CD	Gen3 ATIS	Gen 4 CD	DVS-Gen2	DVS-Gen3	DVS-Gen4	CeleX-IV	CeleX-V	Rino 3
Sensor specifications	Year, Reference	2008 [2]	2014 [4]	2017	2011 [3]	2017 [67]	2017 [67]	2020 [68]	2017 [5]	2018 [69]	2020 [39]	2017 [70]	2019 [71]	2018 [72]
	Resolution (pixels)	128 × 128	240 × 180	346 × 260	304 × 240	640 × 480	480 × 360	1280 × 720	640 × 480	640 × 480	1280 × 960	768 × 640	1280 × 800	320 × 262
	Latency (μs)	12μs @ 1klux	12μs @ 1klux	20	3	40 - 200	40 - 200	20 - 150	65 - 410	50	150	10	8	125μs @ 10lux
	Dynamic range (dB)	120	120	120	143	> 120	> 120	> 124	90	90	100	90	120	> 100
	Min. contrast sensitivity (%)	17	11	14.3 - 22.5	13	12	12	11	9	15	20	30	10	15
	Power consumption (mW)	23	5 - 14	10 - 170	50 - 175	36 - 95	25 - 87	32 - 84	27 - 50	40	130	-	400	20-70
	Chip size (mm^2)	6.3 × 6	5 × 5	8 × 6	9.9 × 8.2	9.6 × 7.2	9.6 × 7.2	6.22 × 3.5	8 × 5.8	8 × 5.8	8.4 × 7.6	15.5 × 15.8	14.3 × 11.6	5.3 × 5.3
	Pixel size (μm^2)	40 × 40	18.5 × 18.5	18.5 × 18.5	30 × 30	15 × 15	20 × 20	4.86 × 4.86	9 × 9	9 × 9	4.95 × 4.95	18 × 18	9.8 × 9.8	13 × 13
	Fill factor (%)	8.1	22	22	20	25	20	> 77	11	12	22	8.5	8	22
	Supply voltage (V)	3.3	1.8 & 3.3	1.8 & 3.3	1.8 & 3.3	1.8	1.8	1.1 & 2.5	1.2 & 2.8	1.2 & 2.8		1.8 & 3.3	1.2 & 2.5	1.8 & 3.3
	Stationary noise (ev/pix/s) at 25C	0.05	0.1	0.1	-	0.1	0.1	0.1	0.03	0.03		0.15	0.2	0.1
	CMOS technology (nm)	350 2P4M	180 1P6M MIM	180 1P6M MIM	180 1P6M	180 1P6M CIS	180 1P6M CIS	90 BI CIS	90 1P5M BSI	90	65/28	180 1P6M CIS	65 CIS	180 1P6M CIS
	Grayscale output	no	yes	yes	yes	no	yes	no	no	no	no	yes	yes	yes
	Grayscale dynamic range (dB)	NA	55	56.7	130	NA	> 100	NA	NA	NA	NA	90	120	50
	Max. frame rate (fps)	NA	35	40	NA	NA	NA	NA	NA	NA	NA	50	100	30
Camera	Max. Bandwidth (Meps)	1	12	12	-	66	66	1066	300	600	1200	200	140	20
	Interface	USB 2	USB 2	USB 3		USB 3	USB 3	USB 3	USB 2	USB 3	USB 3			USB 2
	IMU output	no	1 kHz	1 kHz	no	1 kHz	1 kHz	no	no	1 kHz	no	no	no	1 kHz

4 Other Neuromorphic processors: Intel Loihi and ETH-Kraken SoC

4.1 Intel Loihi

During the project we have instegiated on the use of the neuromorphic research chip Loihi was used to run the SNNs. The proposed designed algorithms were evaluated both on the Nahuku32 platform, running on Intel's cloud server, as well as on the Kapoho Bay platform. While both platforms utilise the same Loihi chip for computation, the Nahuku32 board consists of 32 Loihi chips and allows run-time performance characterization by using the Energy Probe and Execution Time Probe object directly from Intel's API. The Kapoho Bay, shown in figure below, comes in a USB form factor with 2 Loihi chips containing a total of 256 neuro-cores where 262144 neurons and up to 260 million synapses can be implemented. It can be easily interfaced for live

communication with the host system (an UP board featuring an IntelRATOM™x5-Z8350 Processor with 64 bits up to 1.92GHz and 4GB of RAM. The UP
board is running Ubuntu 18.04. Python 3.7 and version 1.0.0 of Intel's NxSDK used to program the SNNs) by programming the three embedded x86 processors which are used for monitoring and I/O spike management embedded on the Loihi chip.

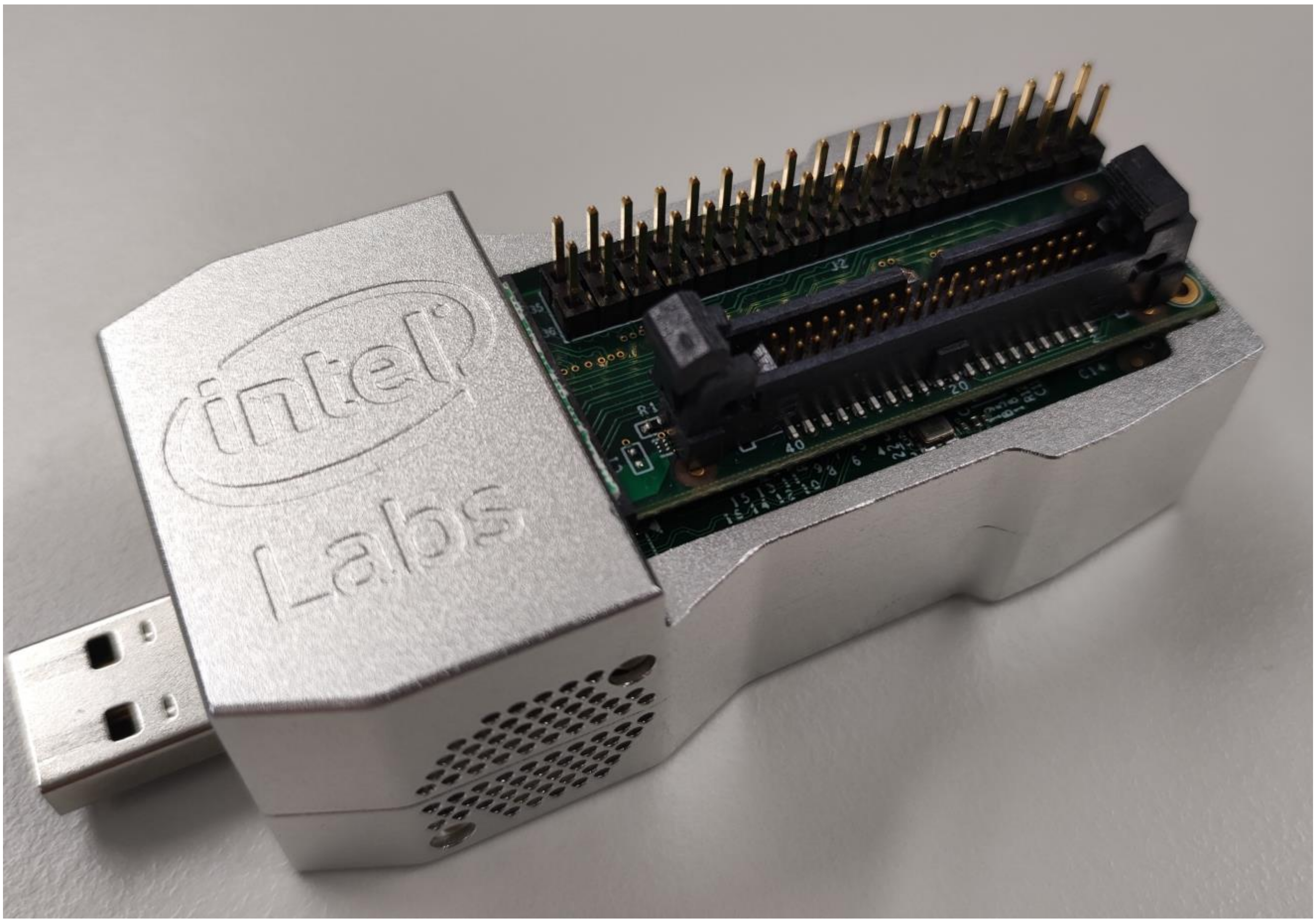


Fig. 2: Intel's Kapoho Bay

4.2 ETH Zurich Novel SOC Kraken and Evaluation board designed during the project.

The Kraken System on Chip (SoC) has an heterogeneous architecture composed of three main subsystems. A block diagram representation of the Kraken chip is reported in figure 3. The first subsystem, the is fabric controller (FC), built around a 32bits RISC-V core which acts as a main programmable control unit for the whole SoC. The FC hosts the main interconnection busses towards the main L2 memory and the APB bus, which controls all the SoC peripherals. As part of the FC domain, we also find a compliant RISC-V debug unit accessible via JTAG, the event unit, which collects interrupt events generated by the peripherals and redirects them towards the core interrupt controller, and four real-time counters (timers) that can be used to generate \PWM signals or internal time references for low latency control of actuator such as the drone motors. Moreover, the FC hosts the power management unit, accessible through a memory-mapped register interface by the FC core.

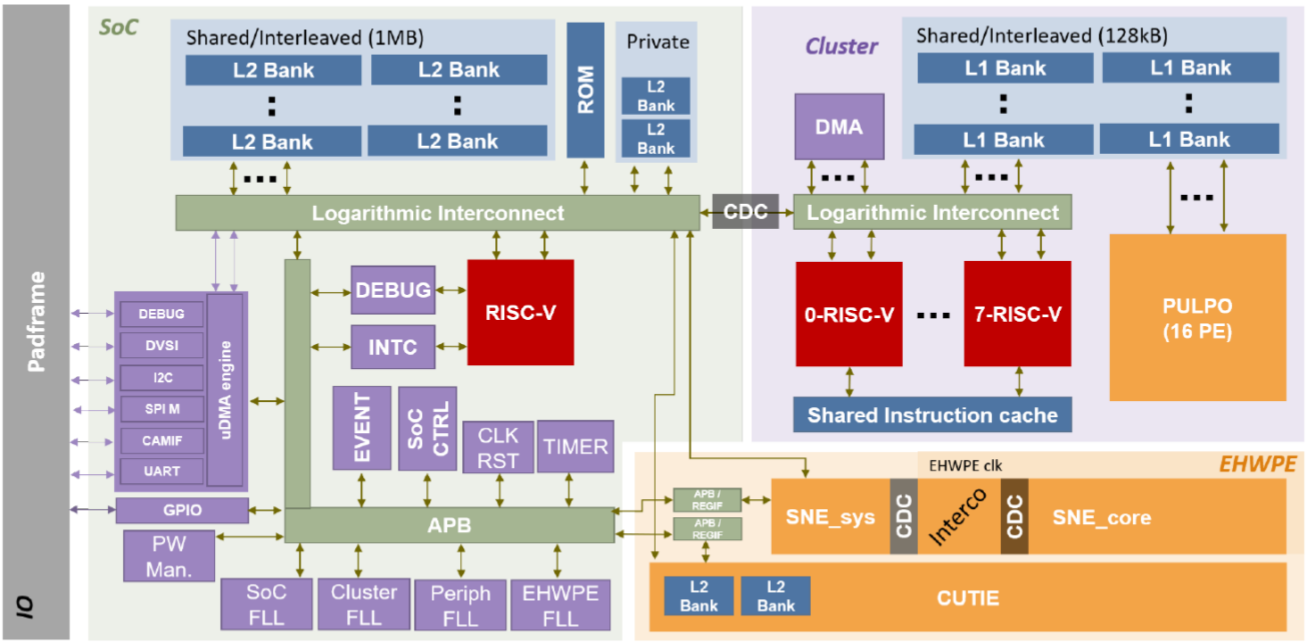


Fig. 3: Kraken SoC Block Diagram

The second subsystem is a RISC-V based general-purpose accelerator called `cluster''. The cluster domain hosts eight RISC-V cores enhanced with dedicated extensions like hardware loops, multiply and accumulate, and vectorial instructions for low-precision machine learning workloads. Fast event management, parallel thread dispatching, and synchronization are supported by a dedicated hardware block (HW Sync), enabling very fine-grained parallelism and high energy efficiency in parallel workloads. The cluster can be clock-gated with a single core granularity, reducing the dynamic power consumption while waiting for cores synchronization.

The third domain, the EHWPE, hosts two accelerators. One is the Spikining Neural Engine (SNE) neuromorphic accelerator, and the other one is CUTIE, a ternary weight neural network accelerator. The EHWPE domain operates on two independent clock domains.

.

Kraken evaluation board

The Kraken evaluation board integrates the Kraken chip with the external components needed to implement complete applications. A USB-C connector is used for JTAG and UART data transfer and supplies a 5V rail from which all other power supplies are derived. Kraken's three power domains are supplied by individually runtime-configurable buck converters, allowing for application-controlled. Connectivity to the DVS camera is provided through an 80-pin low-profile board-to-board connector which also supplies the camera board with the required 1.2volt and 5volt rails. Off-chip memory is present in the form of a combined HyperFlash/HyperRAM chip as well as a quad-SPI flash memory chip, which can also be used to store application code for standalone booting.

Arduino headers and a ribbon connector provide additional connectivity, with level shifters between every off-chip connector and Kraken's I/O pins. A figure of the board is illustrate below.

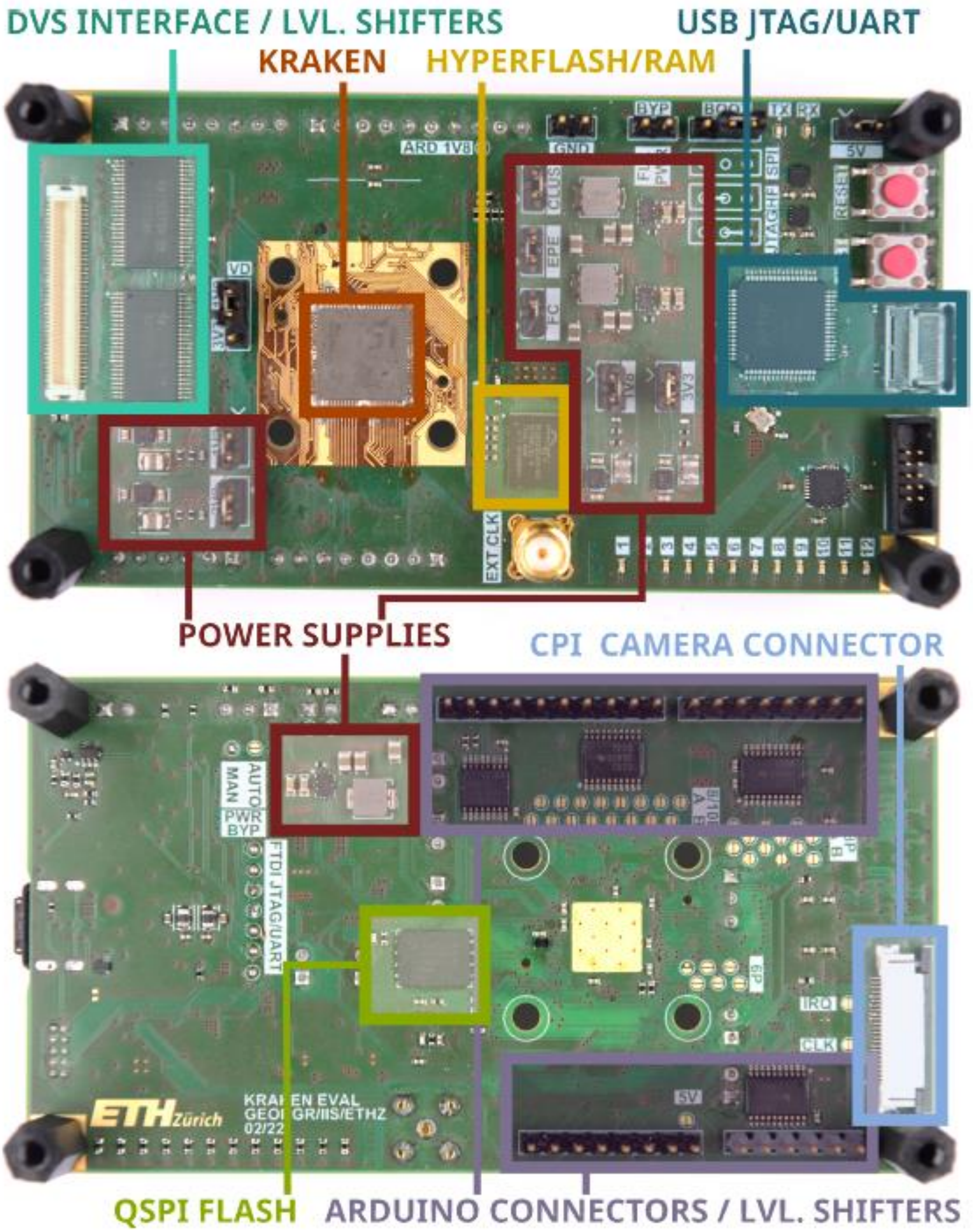


Figure 4. Kraken evaluation board including a CPI camera connector and DVS interfaces for direct connection to cameras.

5 Whole hardware architecture

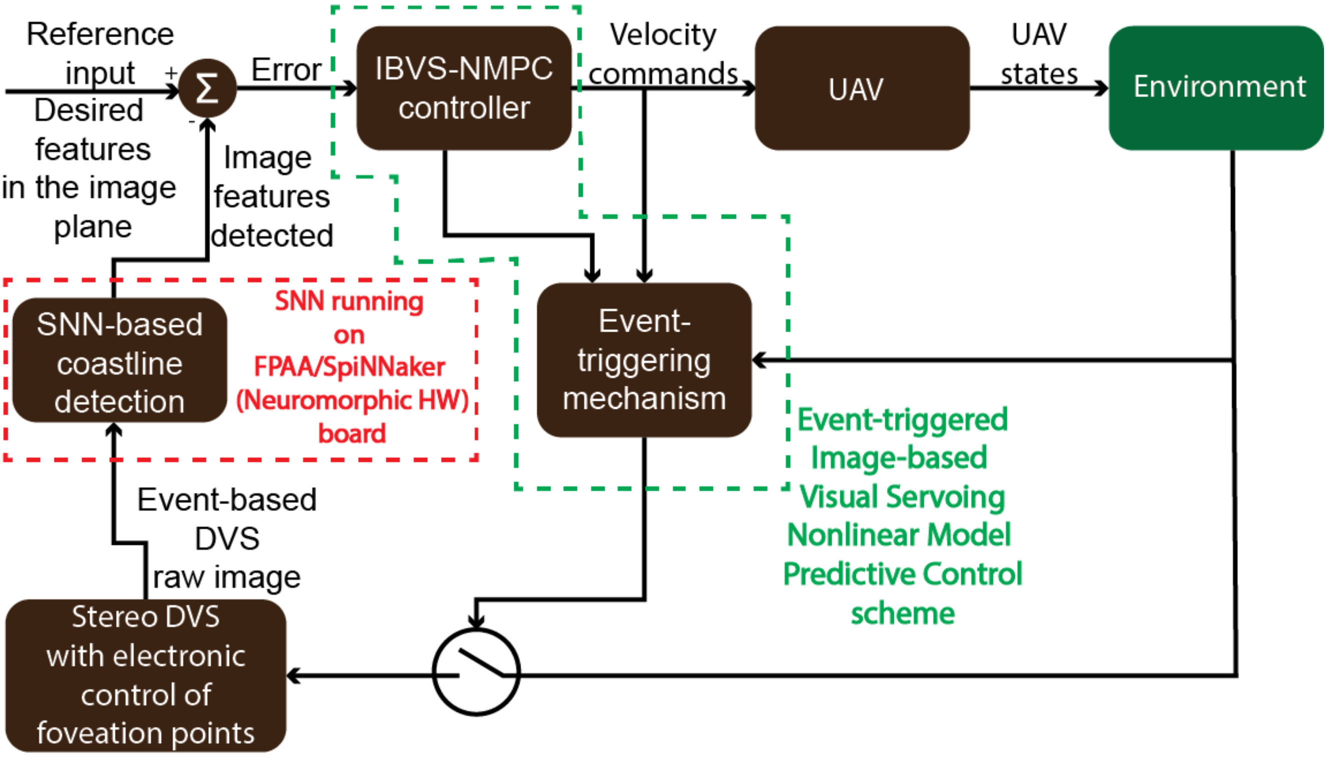

Reference input
+
Σ
-
Desired features in the image plane
Error
IBVS-NMPC controller
Velocity commands
UAV
UAV states
Environment
Image features detected
SNN-based coastline detection
SNN running on FPAA/SpiNNaker (Neuromorphic HW) board
Event-triggering mechanism
Event-triggered Image-based Visual Servoing Nonlinear Model Predictive Control scheme
Event-based DVS raw image
Stereo DVS with electronic control of foveation points

Figure overview.

1.1 Digital Board from Flight navigation Jetson Nano Nvida

The perception system, depicted in Fig. consists of a DVS and a hybrid computing system, a combination of SpiNN-3 neuromorphic computing platform and the onboard computer. The vision sensor communicates with the computing system with an optimized VLSI design running on an FPGA board to avoid latency. SpiNN-3 board is used to implement the core of the contour detection algorithm. The onboard computer is used to run the tracking algorithm on the outcome of the SpiNN-3 board and simultaneously the event-based control strategy using the Robot Operation System (ROS). The MAVROS communication protocol is used to send the velocity commands produced by the control scheme to the octocopter's microcontroller. The low-level control of the octocopter, is used to handle the Pixhawk's velocity commands.

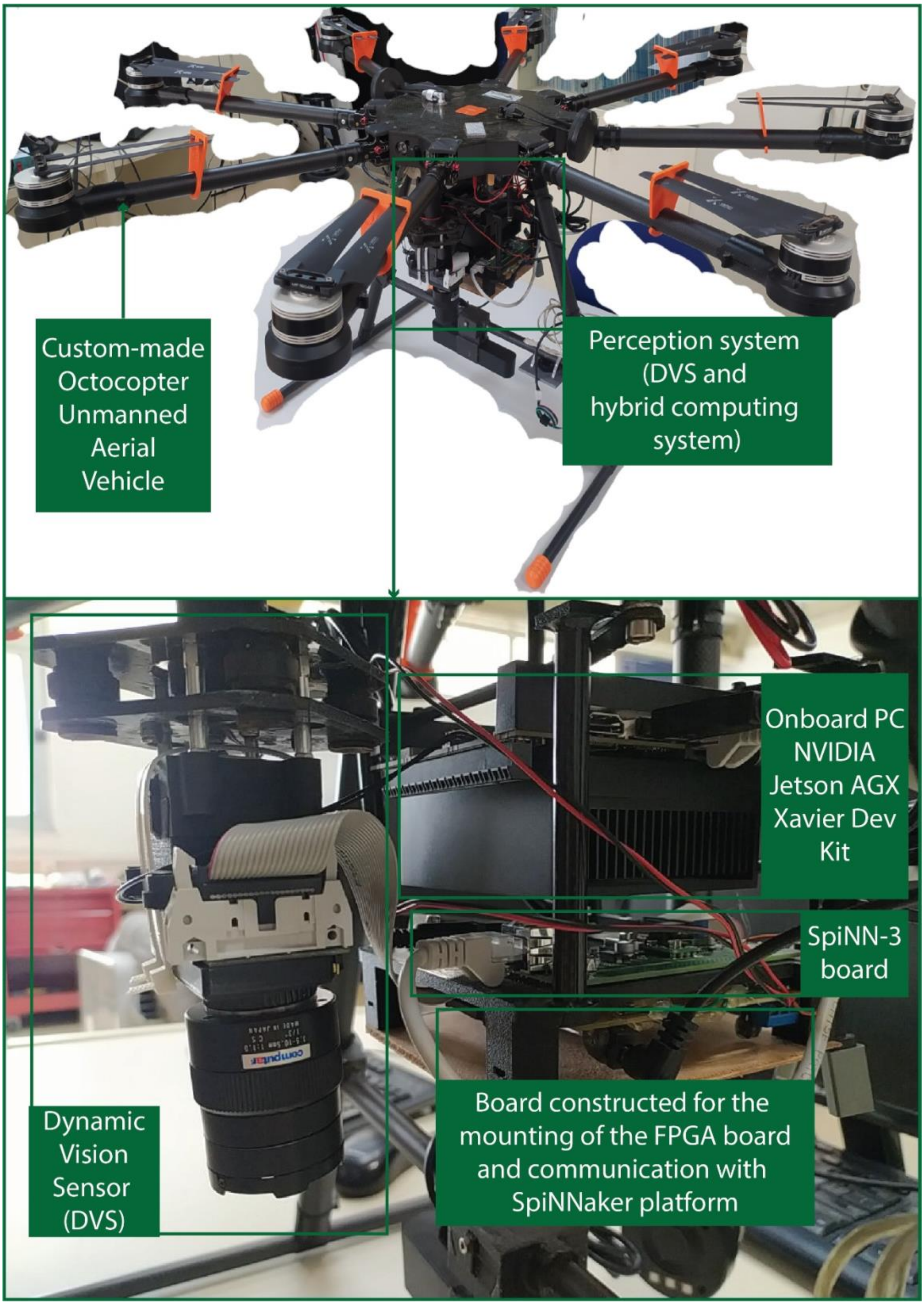
1.2 Hardware for the Drone (Propeller etc.)

The unmnanned aerial vehicle is a crucial part of the APROVIS3D project and, thus, a vehicle which satisfy the requirements of the project is necessary for the success of the operation. The NTUA octocopter is a complicated robotic system, composed of multiple parts, which turn it into a powerful and fully autonomous Unmanned Aerial Vehicle. More precisely, the NTUA octocopter is loaded with the Ardupilot firmware responsible for controlling the aircraft through all regimes of flight. Ardupilot runs on the Pixhawk Cube Orange \cite{ardupilot}, the heart of the system where all the necessary hardware, i.e. ESCs and sensors, is integrated. The autopilot provides a set of modes which vary from semi-manual control to entirely autonomous, and, hence, the level of the authority given to the human pilot is adjusted correspondingly.

Additionally, the NTUA octocopter is equipped with navigation sensors which provide information about the vehicle position, velocity and angular orientation. Specifically, the following sensors are available:

- A rangefinder which is the primary altitude source,
- A compass or magnetometer providing heading/yaw measurements,
- A GPS which contributes to the estimation of the velocity and the position of the multirotor and
- An IMU which measures the linear accelerations and the body angular rates.

The above sensors are fused using an Extended Kalman Filter implemented by the Autopilot side and consequently, a proper state estimation is provided during the flight. The safe navigation of the NTUA octocopter requires a sensor capable of executing accurate and robust coastline detection and, thus, the DVS is used combined with a ZED 2 stereo camera to cross reference the event- and frame-based information respectively. Additionally, the execution of computationally expensive algorithms such as image processing or event-triggering image-based visual servoing model predictive control is a necessary prerequisite and, consequently, the incorporation of a powerful onboard computer is inevitable. Among the various embedded computers, Jetson AGX Xavier can be distinguished owing to its high performance. Beyond this, the Jetson AGX Xavier is suitable for UAV applications where size, weight, and power consumption play a crucial role. The aforementioned system is appropriately set up in order to interface with the flight controller using the MAVLink protocol. The real-time control of the vehicle is achieved using the Robot Operating System (ROS) and particularly, through the MAVROS node, which provides communication between ROS and Ardupilot vehicles.


Custom-made Octocopter Unmanned Aerial Vehicle
Perception system (DVS and hybrid computing system)
Onboard PC NVIDIA Jetson AGX Xavier Dev Kit
SpiNN-3 board
Dynamic Vision Sensor (DVS)
Board constructed for the mounting of the FPGA board and communication with SpiNNaker platform