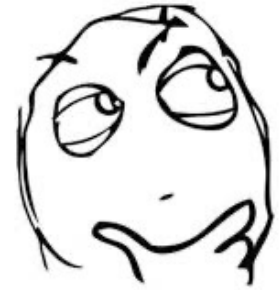
Caribou: A Versatile Data Acquisition System for Silicon Pixel Detectors

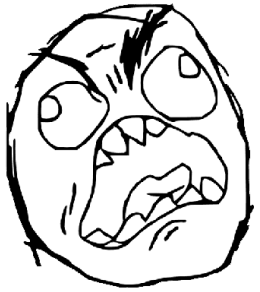
M. Benoit (ORNL), E. Buschmann (BNL), Hucheng Chen (BNL), D. Dannheim (CERN), R. Palomo (Sevilla), **Y. Otarid (CERN),** M. Pijacki (Carleton), S. Spannagel (DESY), T. Vanat (DESY).

3rd CERN System-on-Chip Workshop **04/10/2023**

A particular solution to a particular need

Most silicon pixel detectors share the same power, control and readout concepts
(different voltage levels, number of channels, protocols …)





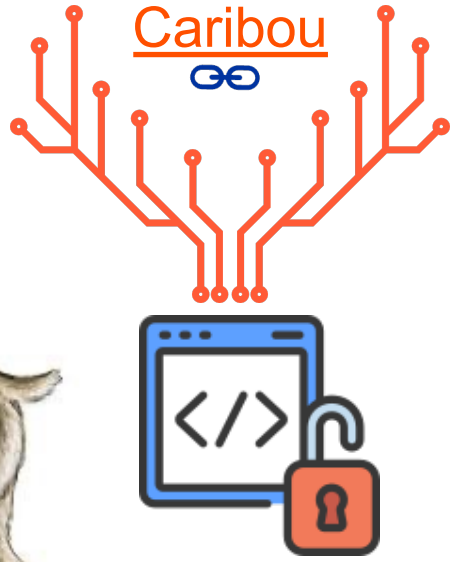
Every new detector drives the development of a new DAQ system or modification of an exisiting one
(time consuming, not very innovative)

Why not a common versatile DAQ system ?
(Common hardware, firmware and software cores, keeping the focus on detector integration)

Collaborating towards the open

Open source hardware, firmware and software for laboratory and beam tests

Caribou



DOI 10.1088/1748-0221/12/01/P01008
DOI https://doi.org/10.22323/1.370.0100
DOI 10.1088/1748-0221/18/02/C02005

Maintained by a collective effort of hardware, firmware and software developers

DESY.

CERN

OAK RIDGE National Laboratory

Carleton University

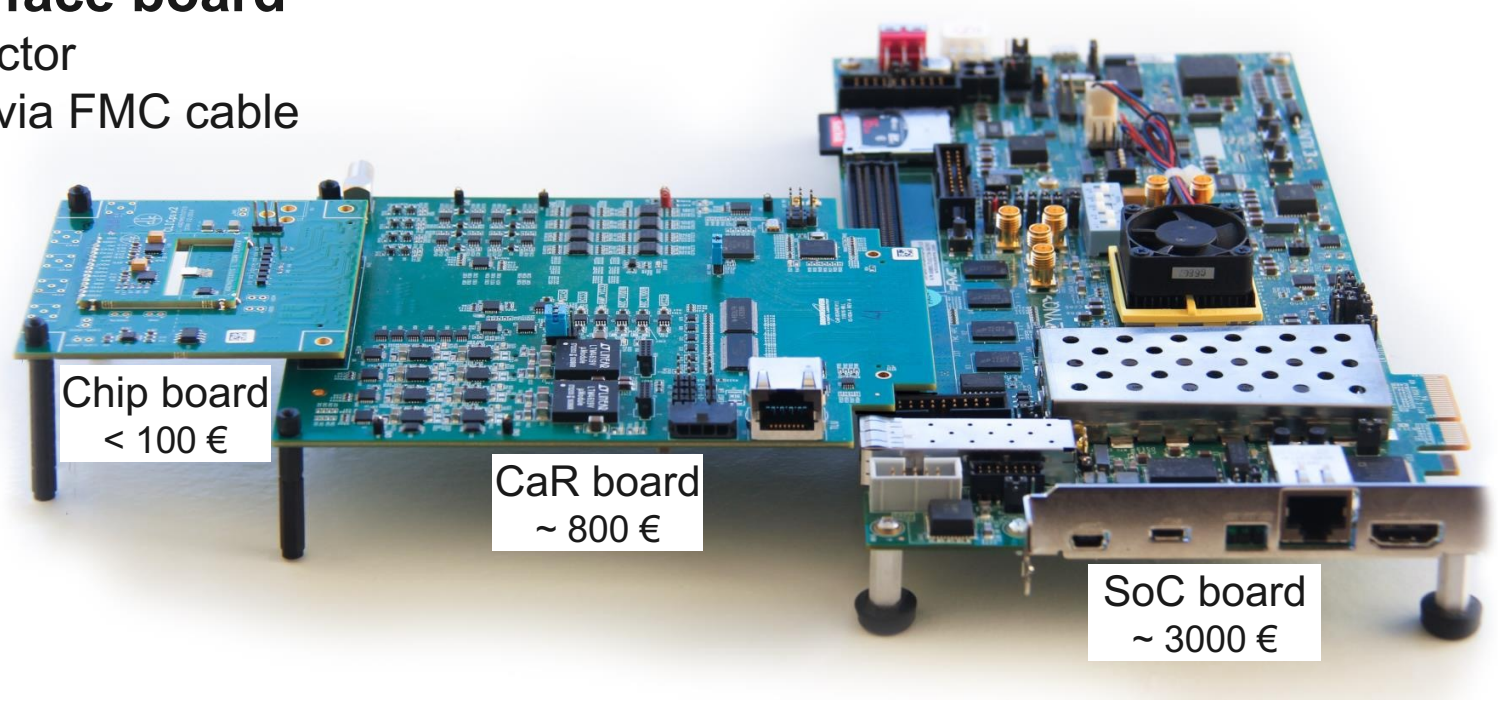
Brookhaven National Laboratory

Caribou hardware architecture

- **System-on-Chip (SoC) board**
  - ie: Xilinx ZC706 evaluation board
  - Embedded CPU runs DAQ and control software
  - FPGA runs custom firmware for detector control and readout

- **Control and Readout (CaR) interface board**
  - Physical interface from SoC to detector
  - CaR – SoC connection extendable via FMC cable

- **Detector (chip) carrier board**
  - Custom low-cost PCB



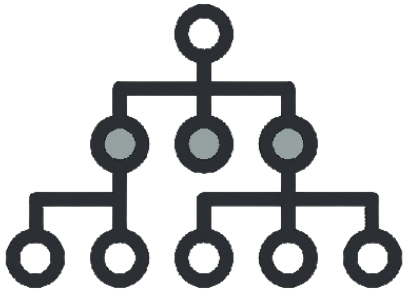
System-on-Chip board

Xilinx ZC706 evaluation board

Zynq-7000 XC7Z045-2FFG900C SoC

Processing System (PS)

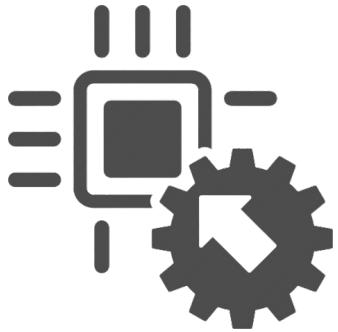
2 x ARM Cortex-A9 MPCore CPUs
Yocto-based Linux
Network/ssh control interface
Caribou DAQ software (Peary)



https://www.xilinx.com/products/boards-and-kits/ek-z7-zc706-g.html

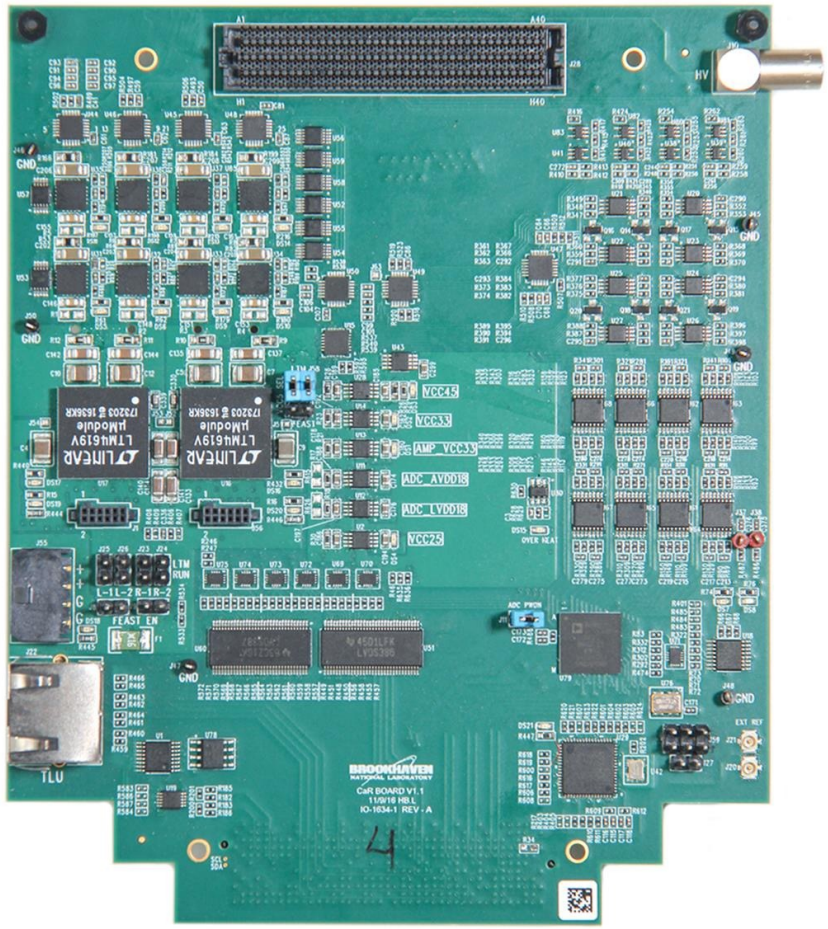
Programmable Logic (PL)

Kintex-7 FPGA
AXI control interface
Caribou firmware



Control and Readout (CaR) board

Feature	Description
Adjustable Power Supplies	8 units, 0.8 – 3.6 V, 3 A
Adjustable Voltage References	32 units, 0 – 4 V
Adjustable Current References	8 units, 0 – 1 mA
Voltage Inputs to Slow ADC	8 channels, 50 kSPS, 12-bit, 0 – 4 V
Analog Inputs to Fast ADC	16 channels, 65 MSPS, 14-bit, 0 – 1 V
Programmable Injection Pulsers	4 units
Full-Duplex High-Speed GTx Links	8 links, <12 Gbps
LVDS Links	17 bidirectional links
Input/Output Links	10 output links, 14 input links, 0.8 – 3.6 V
Programmable Clock Generator	Included
External TLU Clock Reference	Included
External High-Voltage (HV) Input	Included
FEAST Module Compatibility	Supported
FMC Interface to FPGA	Included
SEARAY Interface to Detector Chip	320-pin connector



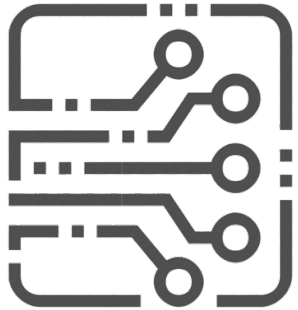
https://gitlab.cern.ch/Caribou/hardware/carboard

Resources for various target applications

20 CaR boards v1.4 produced and distributed within RD50 common project

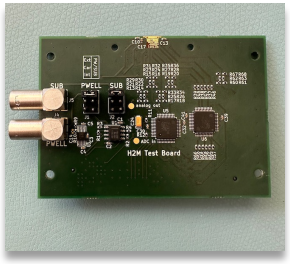
Detector carrier board (chip board)

- **Detector-specific**
  - Physical hardware hosting the detector
  - Only provide passives and detector-specific components

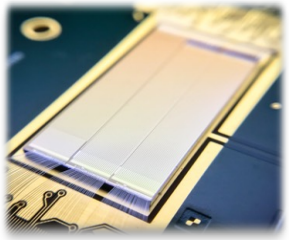


- **Multiple detectors already supported:**

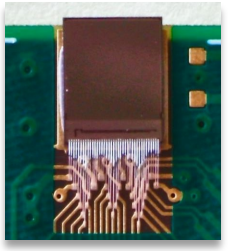
H2M



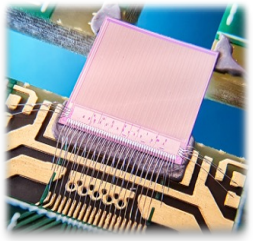
ATLASpix



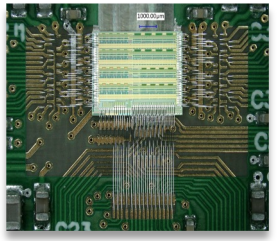
CLICpix2



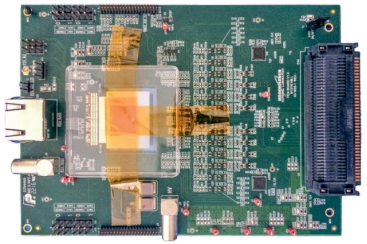
CLICTD



FASTPIX

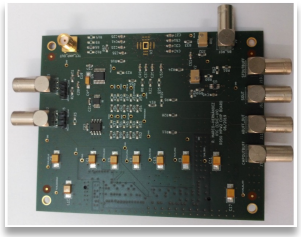


H35Demo/FEI4

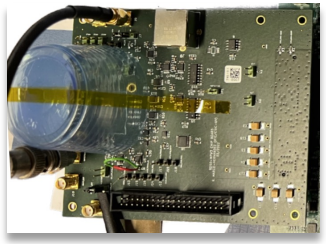


RD50-MPW1

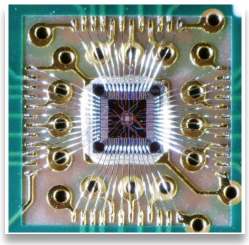
RD50-MPW2



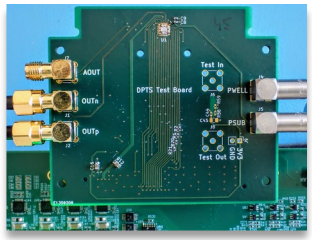
RD50-MPW3



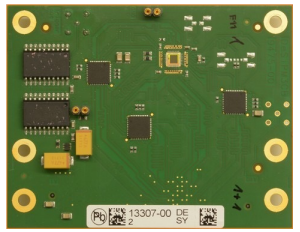
APTS



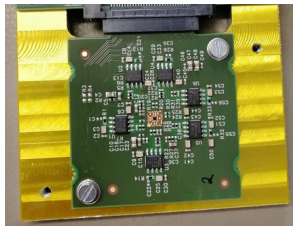
DPTS



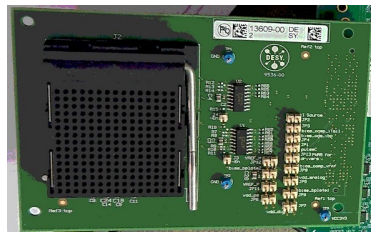
dSiPM



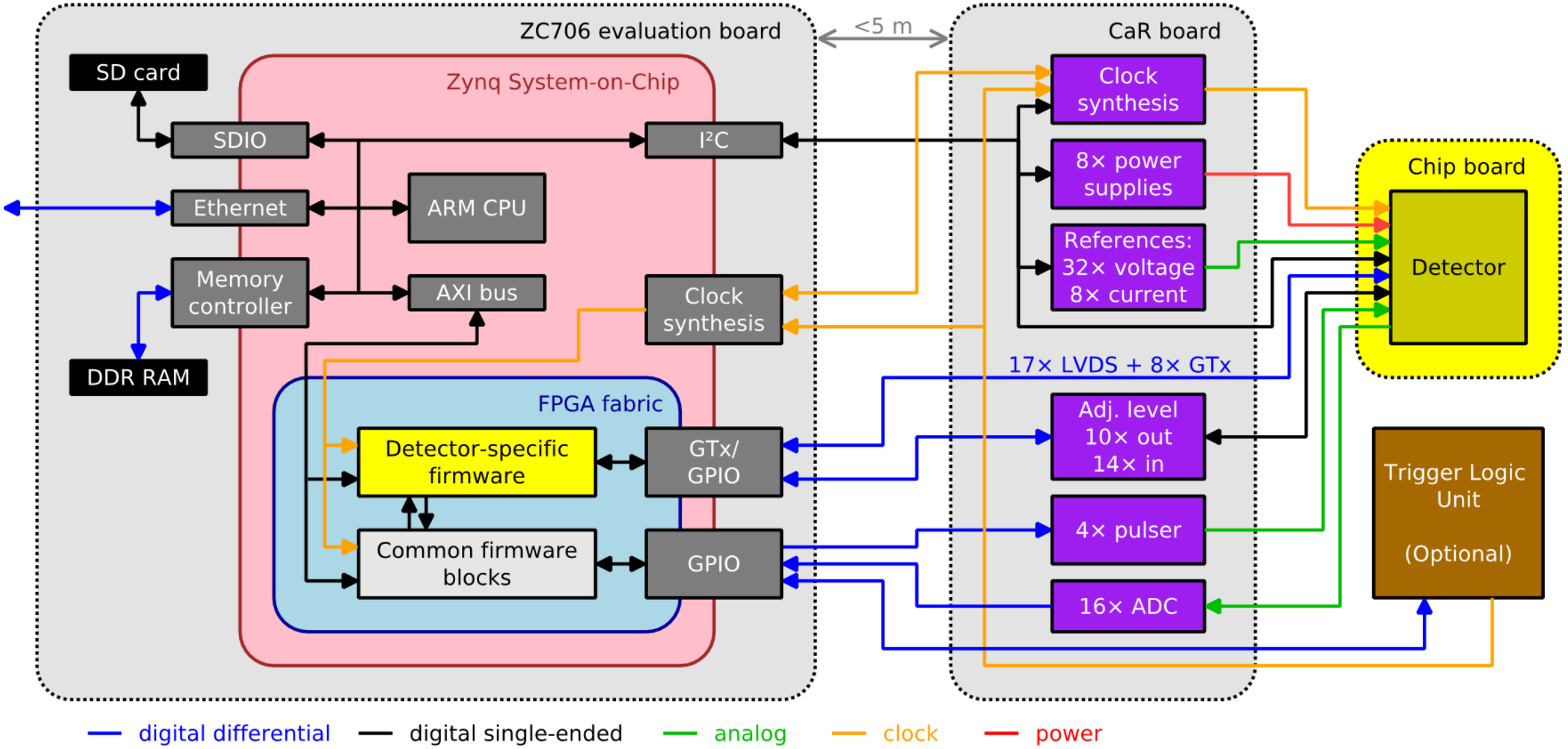
MLR1



CoRDIA

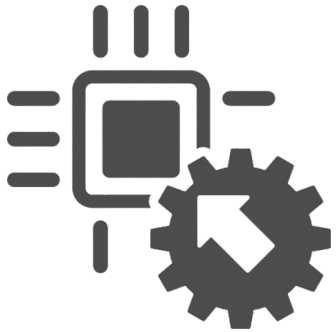


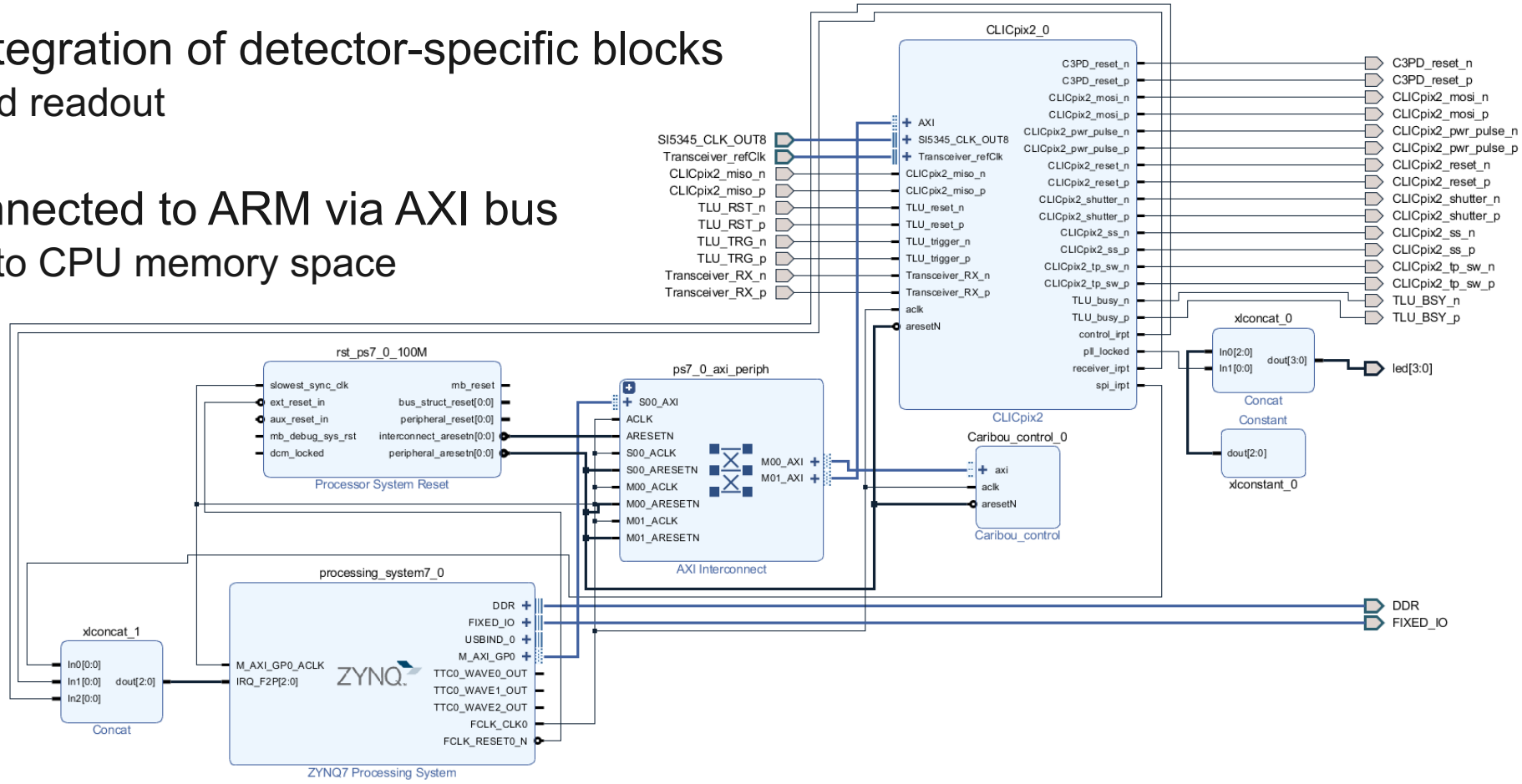
Caribou system architecture



Caribou FPGA firmware

- Combination of custom and Xilinx IP cores
  - Common logic
- Development and integration of detector-specific blocks
  - Detector control and readout
- Firmware blocks connected to ARM via AXI bus
  - Registers mapped to CPU memory space



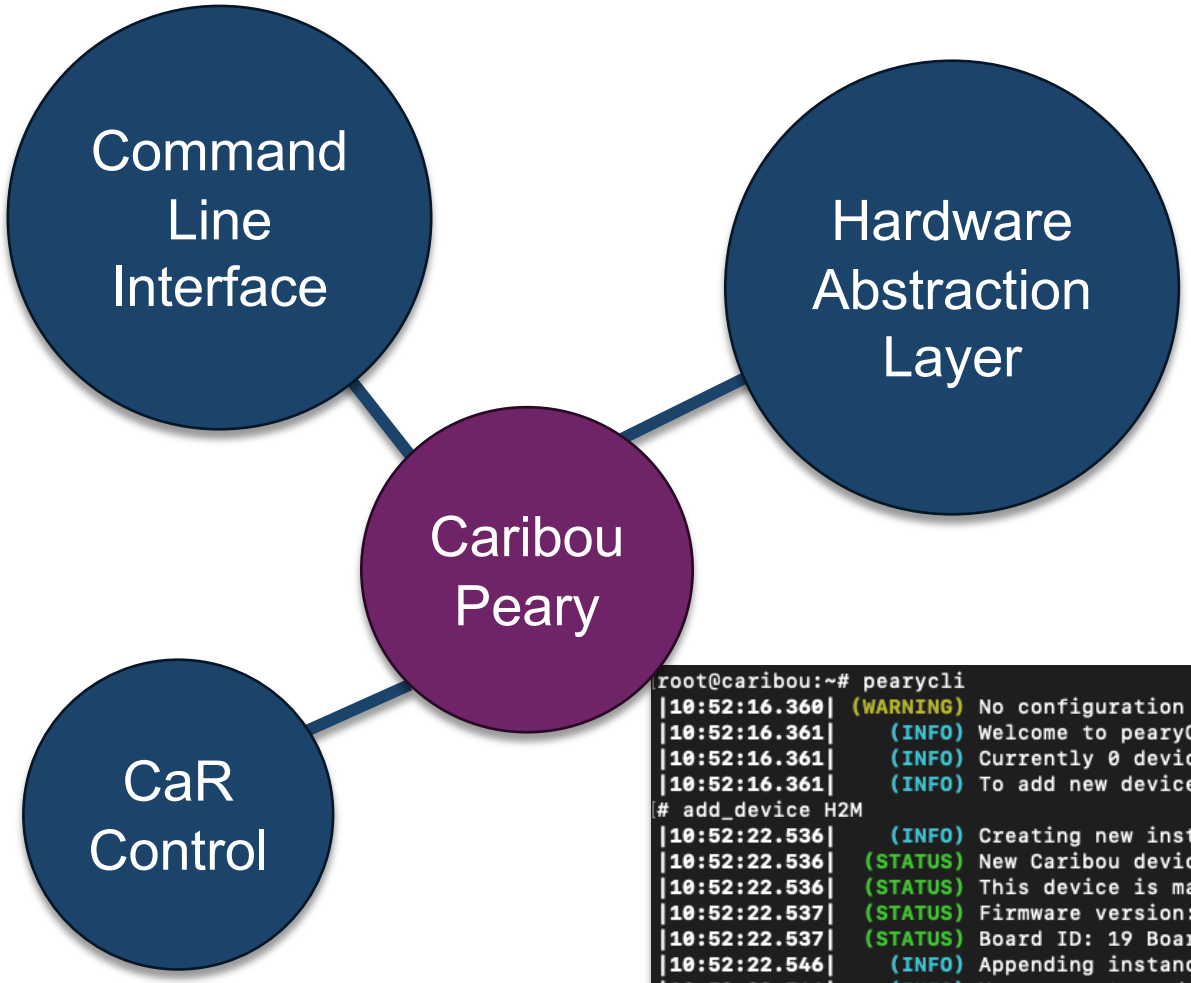


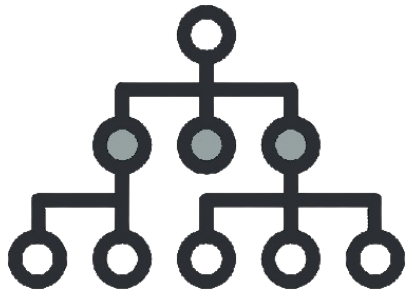
Caribou software – OS – Linux

- Yocto-based Linux distribution
- OpenEmbedded build system
- Yocto reference embedded distribution: Poky
  - Standard Linux packages (ssh, python, git, NTP, …)
  - Community-developed layer for Xilinx ZC706 (meta-xilinx)
  - Custom layers with user-specific software and recipes (meta-caribou)
- OS boot from SD card
  - Boot partition with bitstream and boot configuration (ie: MAC address)
  - Linux partition with root filesystem
- Gitlab CI-based nightly build
  - Preserved build cache (~50 GB) and single recipie rebuild
  - DAQ software included

Caribou software – DAQ – Peary

Access to device functions and routines





Interface between SW and HW + Periphals handling

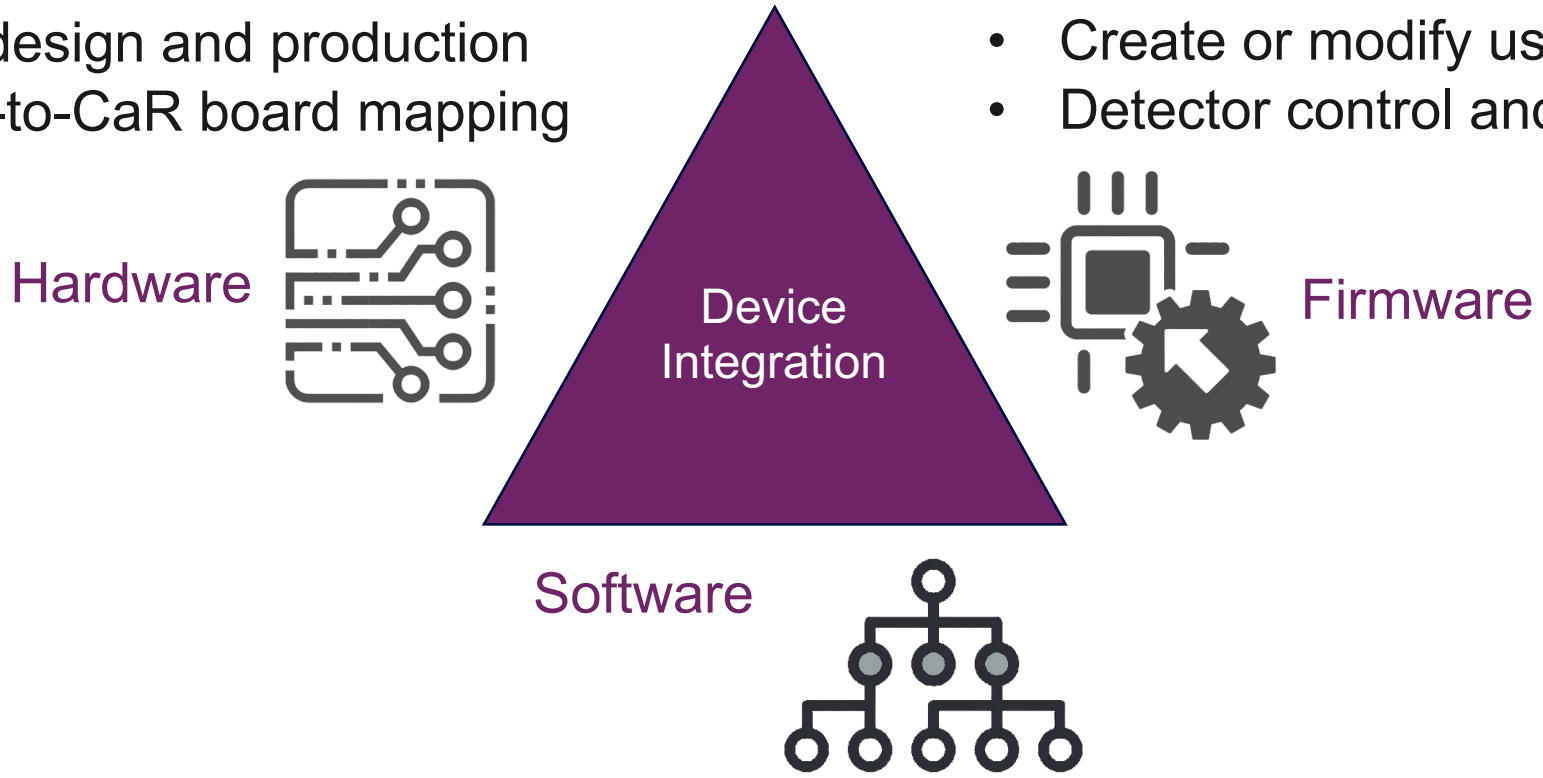
Voltage supply and ADC control

```
root@caribou:~# pearycli
|10:52:16.360| (WARNING) No configuration file provided, all devices will use defaults!
|10:52:16.361|    (INFO) Welcome to pearyCLI.
|10:52:16.361|    (INFO) Currently 0 devices configured.
|10:52:16.361|    (INFO) To add new devices use the "add_device" command.
(# add_device H2M
|10:52:22.536|    (INFO) Creating new instance of device "H2M".
|10:52:22.536|  (STATUS) New Caribou device instance, version peary v0.11.0+447^g638de98f
|10:52:22.536|  (STATUS) This device is managed through the device manager.
|10:52:22.537|  (STATUS) Firmware version: 0x00000000 (0/0/2000 0:0:0)
|10:52:22.537|  (STATUS) Board ID: 19 Board Revision: 0x01
|10:52:22.546|    (INFO) Appending instance to device list, device ID 0
|10:52:22.546|    (INFO) Manager returned device ID 0.
(# powerOn 0
|10:52:28.500|    (INFO) [H2M] Powering up
```

Device integration workflow

- Chip board design and production
- Ensure chip-to-CaR board mapping

- Create or modify user-specific blocks
- Detector control and readout



- Define CaR board peripherals and firmware registers mapping
- Create detector-specific class with custom functions

Application examples

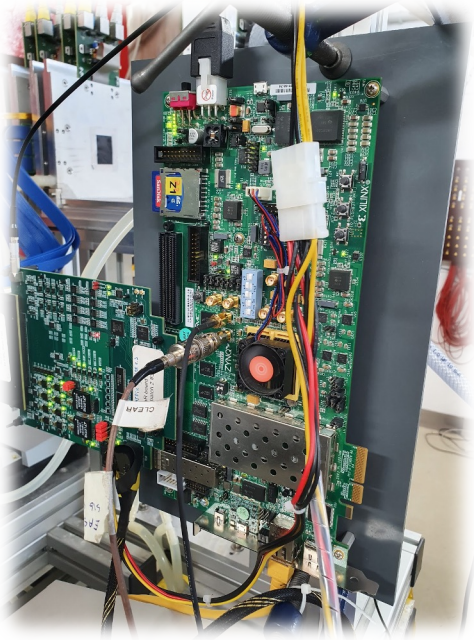
- Support for various readout schemes
  - Digital interface via GTx or LVDS
  - Analogue waveforms (ADC or oscilloscope)

- Integration in beam telescope setups
  - FEI4, Timepix3, Mimosa, ALPIDE

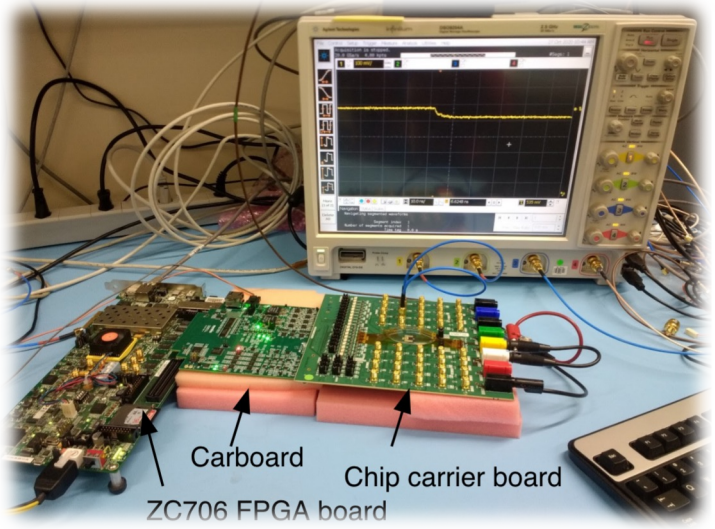
Telescope integration

CLICdp Timepix3 @ CERN

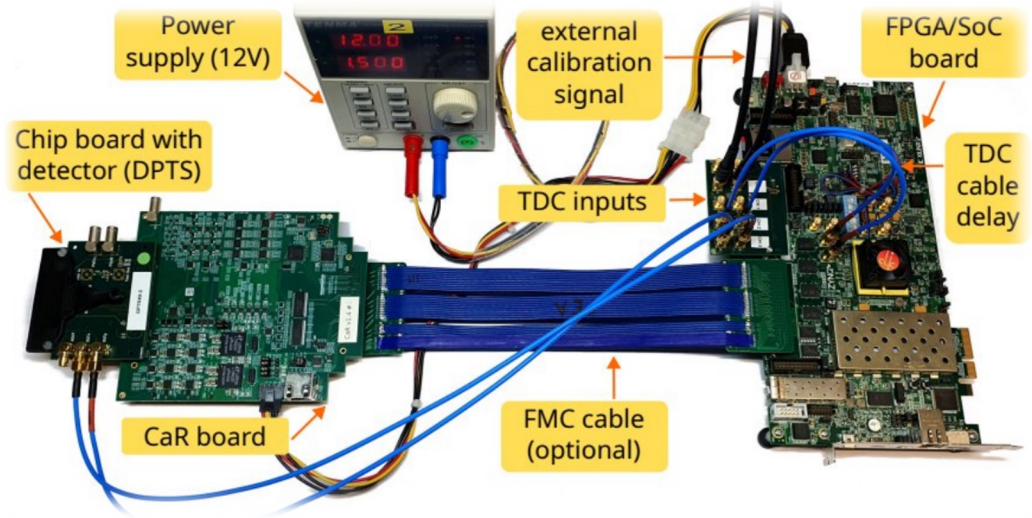
MIMOSA @ DESY



FASTpix with oscilloscope readout



DPTS with TDC in FPGA readout

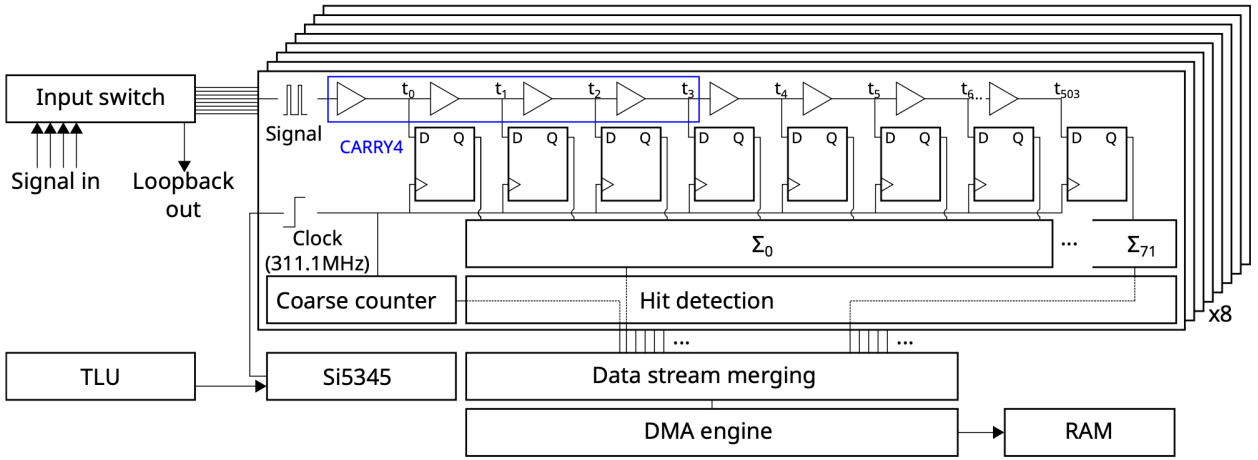


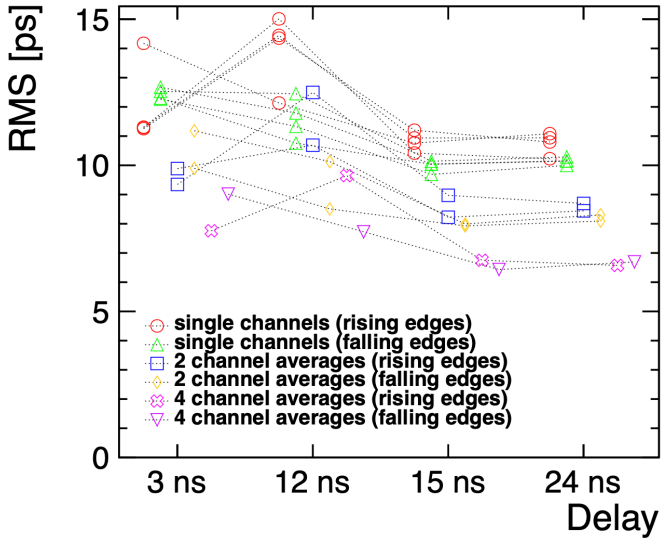
ALPIDE @ MAMI

Some recent extensions

Focus on picosecond timing applications

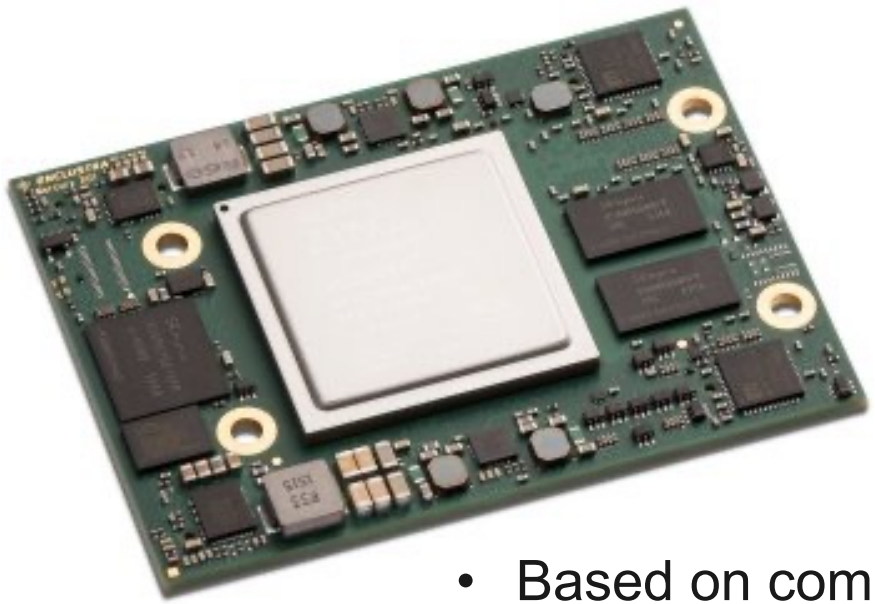
- Motivation: support of waveform sampling and timing of digital pulses (FASTPIX, APTS, DPTS)
- Oscilloscope readout support in Peary software
  - For both lab measurement and test beams
- 8-channel TDC with picosecond resolution and dead time free sampling/readout

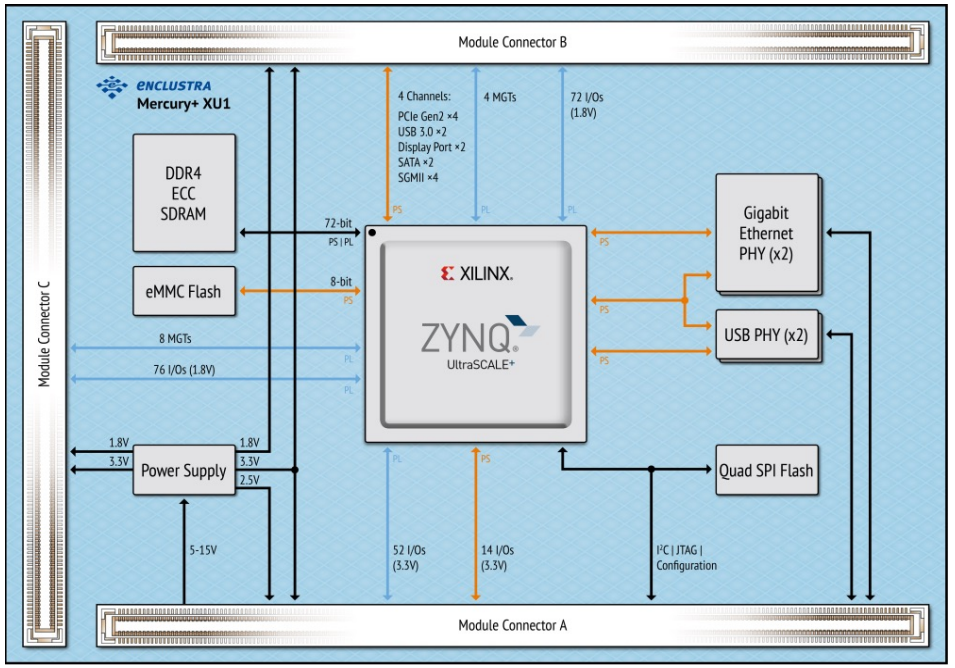




- Publication
- TWEPP 2022

Future plans – Caribou 2.0

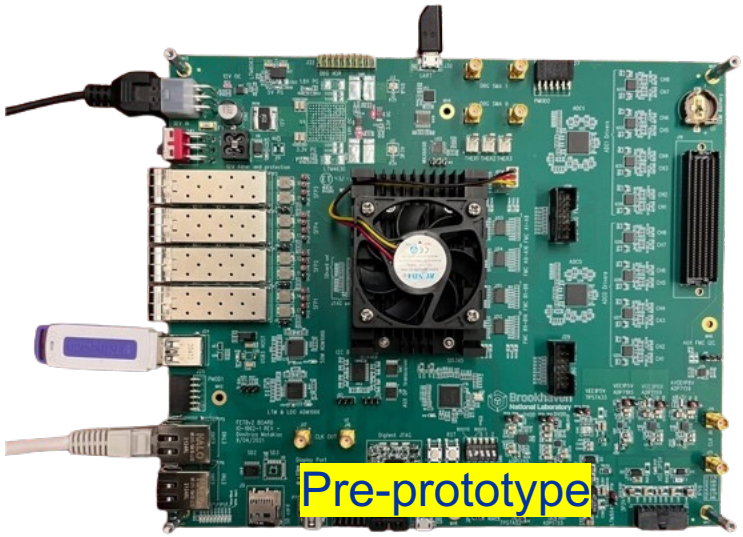




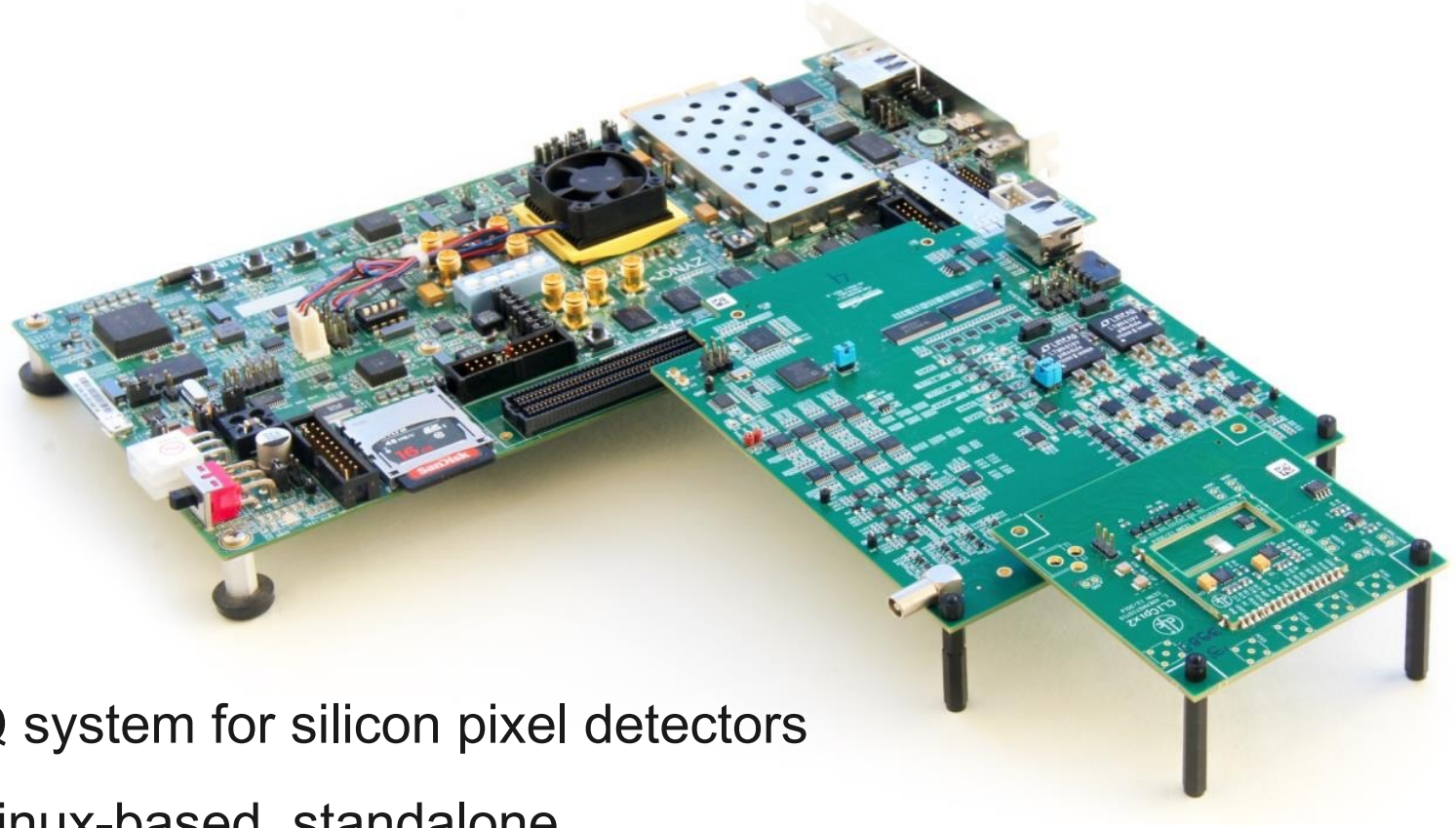
- Based on commercial **System-on-Module** (SoM)
  - Optimize system cost, increase flexibility and performance (GbEth, SFP+, configurable bias polarity, mutli-device support …)
- **Mercury+ XU1** System-on-Chip
  - ZYNQ US+ SoC FPGA
  - More resources and ARM processing power
- **Integrated to CaR board**

Caribou 2.0 – Prototyping phase

- CaR board 2.0 schematics/layout still under development
- Enclustra Mercury+ XU1 SoC module and Mercury+ ST1 base board for protoyping
- Core development plan:
  - Finalize CaR board design
  - Firmware structure revision:
    - single repository, configurable builds, pool of IPs, streamlined documentation website
  - Migrate Yocto build and Peary software to the Zynq UltraScale+ SoC

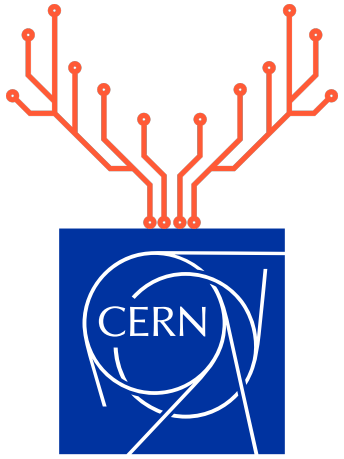


Summary



- Caribou is:
  - A versatile DAQ system for silicon pixel detectors
  - Open source, Linux-based, standalone
  - Proving excellent operation on many detector prototypes
  - Entering its upgrade phase with many improvements to come

Thank you



Contact

CERN
Younes Otarid
EP R&D
younes.otarid@cern.ch

home.cern