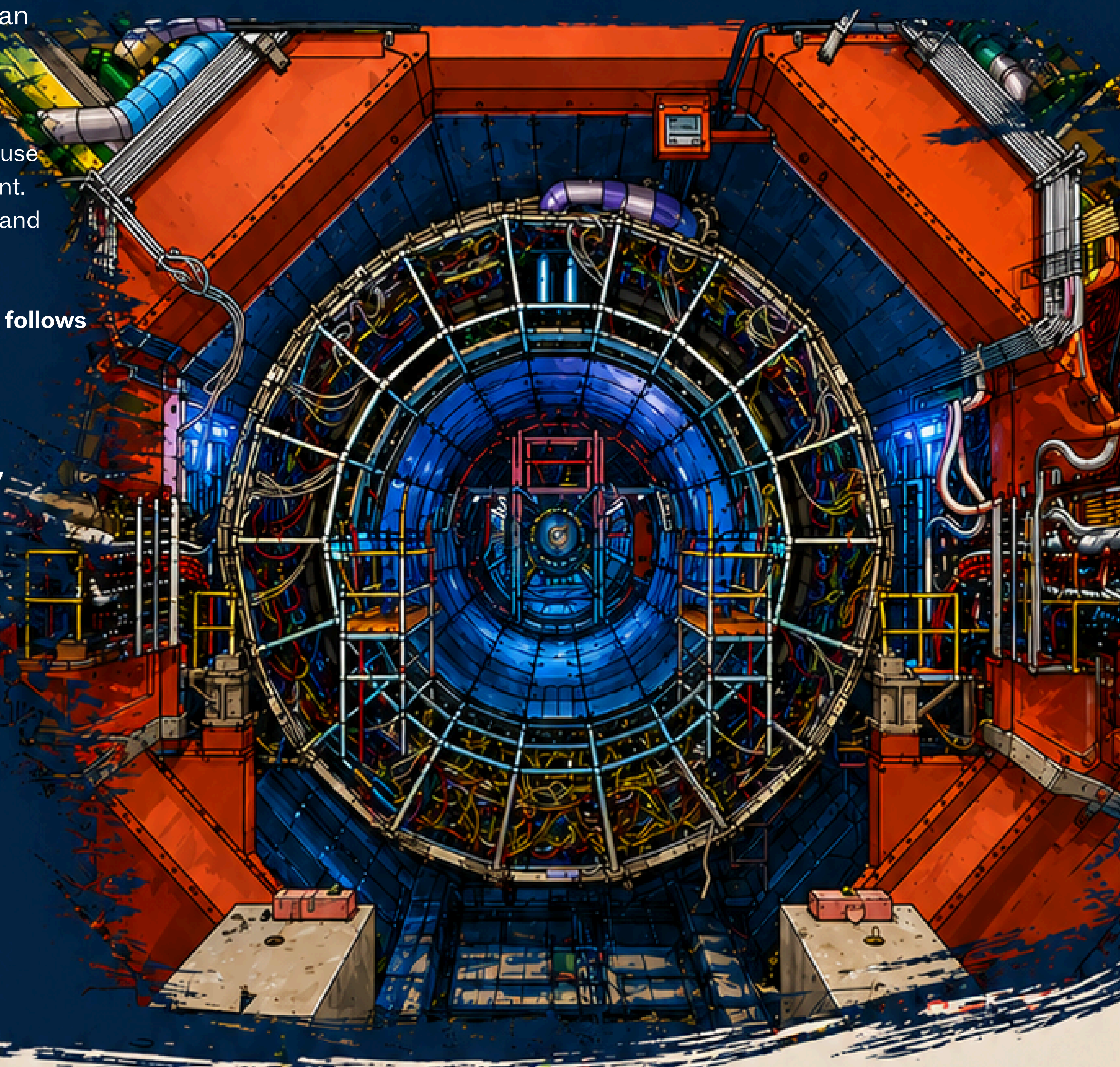
DEVELOPING SOFTWARE FOR HIGH-ENERGY READOUTS

Hasan Karabulut, Sindre Kvam & Arwa Shwan



Introduction

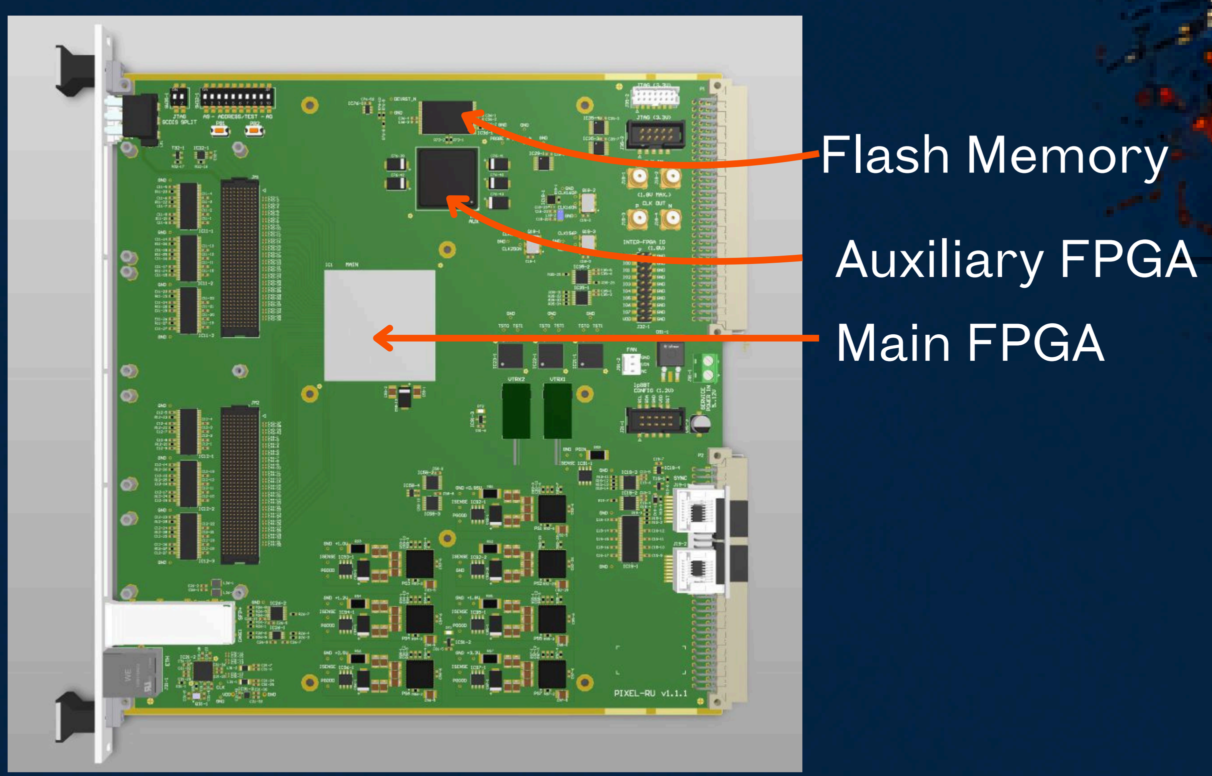
The Forward Calorimeter (FoCal) is a new detector to be installed in ALICE during long shutdown 3 for use in run 4.[1] Readout Units (RUs) are under development. They will operate in a harsh radiation environment, and ensuring correct behaviour is a challenge.

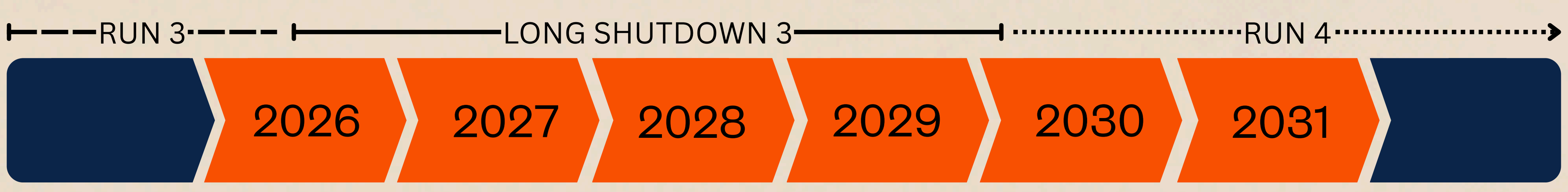
Objective:

Design a prototype of the test-suite in Python that follows an industry standard sowtware architecture.

This software suite must be able to:

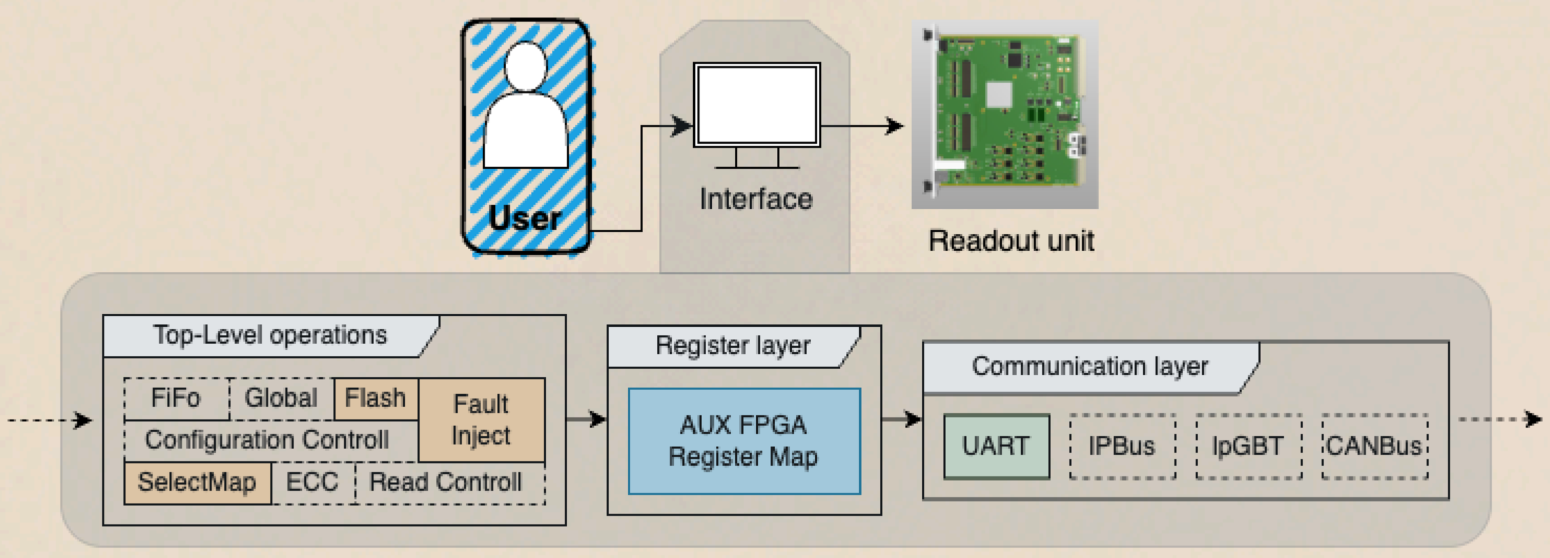
- **Support several hardware protocols**
- **Easily support extensions for new functionality**
- **Be usable for verification during development**
- **Act as a prototype for the production software installed in ALICE for Run 4**





Software Design

A layered design makes the software modular and easy to maintain. Each layer operates independently.



Abstraction hides complex functions and operations from the end-user

```
ser = serial.Serial(port, baud)
ser.write(bytes(0xA5, 0x20,
        0x22, 0, 1, data))
ser.close()
```

Without abstraction

```
UART.connect(port)
UART.config(**flash.pattern())
UART.write(data)
UART.disconnect()
```

With abstraction

The test-suite is tested in the lab on a RU from the ITS2 sub-detector. This is very similar to the final FoCal RU. The tests show that the software is working as intended.

Electronics in Radiation

The RUs uses commercial Field Programmable Gate Arrays (FPGA). Its functionality is stored in memory cells. Radiation can cause Single Event Upsets (SEUs). SEUs can cause functional failures in the FPGA design.[2] Radiation tolerance is therefore tested by:

Radiation beam test campaigns - i.e. bombarding the electronics with protons or neutrons and measure the behavior.

- Very realistic, but can be expensive and time consuming due to preparation and traveling. Challenging to isolate specific results and repeat them

Fault injection - i.e. artificially injecting bit-flips in the configuration memory through a software interface

- Time efficient, cost effective and allows for repeated tests in every design cycle. Only certain parts of the FPGA can be tested, and extensive tests take time with ~160 million bit-flips.

Conclusion

- The test-suite communicates successfully with an test-RU using UART protocol
- More hardware protocols (IPBus, IpGBT/I2C, CANBus) can easily be added
- The test-suite is easily adapted to the new FoCal RU when this is ready
- The test-suite supports *fault injection* giving a tabletop testbed for radiation environments

P299.7

UNIVERSITY OF BERGEN
Faculty of Science and Technology

[1] CERN (2026), *ALICE: FOCAL*, URL: https://alice-collaboration.web.cern.ch/menu_proj_items/FOCAL
[2] Alme, Johan (Aug. 2008). "Firmware Development and Integration for ALICE TPC and PHOS Front-end Electronics: A Trigger Based Readout and Control System Operating in a Radiation Environment" Bergen, Norway: University of Bergen.