

FIT CONTROL SYSTEM

STATUS REPORT

Paweł Russek, Sebastian Koryciak

AGH University of Krakow, Faculty of Computer Science, Electronics, and Telecommunications

STUDENTS FROM AGH

Students at CERN:

- Natalia Ciężka – WinCC O.A.
- Wiktor Pierożak – ALFRED
- Franciszek Urbański – ALFRED

DCS
10.07-07.08

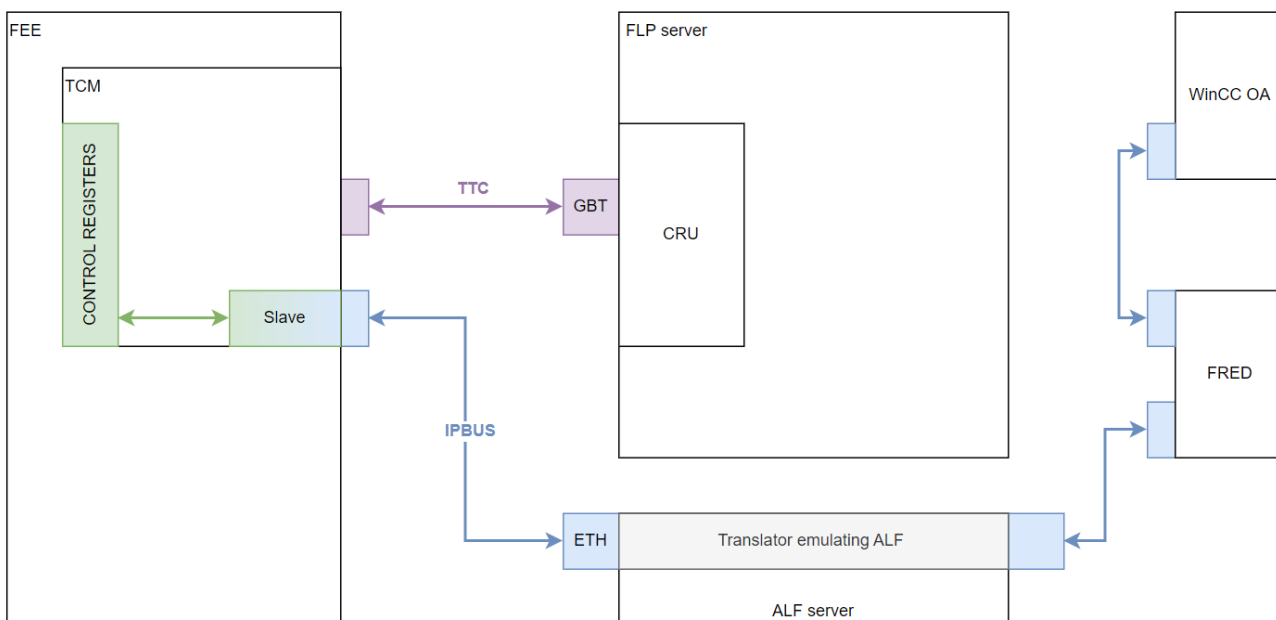
- Oleksandr Savchenko – QC shifts

FPGA
12.07-25.07

Students in ALICE-FIT laboratory at AGH:

- Krzysztof Płonka – PCB design for test setup
- Łukasz Przystupa – software for test setup

TEST
01.07-26.07



Software implemented or tested:

1. IPbus boost::asio
 - Based on the Control Server implementation
 - Uses boost::asio library (available on the FLP)
 - Aims to copy all Control Server implementation functionalities
2. IPbus-SWT
 - SWT frame
 - FRED compatible
3. FRED + ALF Simulator @ AGH's FLP

```
[flp@dhcp-pc208 test]$ ./bin/FREDServer
[INFO] [2024:07:16 15:44:56.315958] FRED launched!
[INFO] [2024:07:16 15:44:56.325016] Parsing started.
[INFO] [2024:07:16 15:44:56.326152] Parsing Completed. Starting FRED.
[INFO] [2024:07:16 15:44:56.329761] FRED running.
[INFO] [2024:07:16 15:44:56.330728] Server FRED_0 started!

[flp@dhcp-pc208 build]$ ./bin/AlfSimulator --alf_id 0
[INFO] [2024-07-16 15:43:52.908154] ALF Simulator with DNS name "ALF_0" is running!
[INFO] [2024-07-16 15:43:52.914460] press CTRL-C to exit...

[1] 0:./bin/FREDServer* "dhcp-pc208.elektro.ag" 15:45 16-Jul-24
```

Repositories

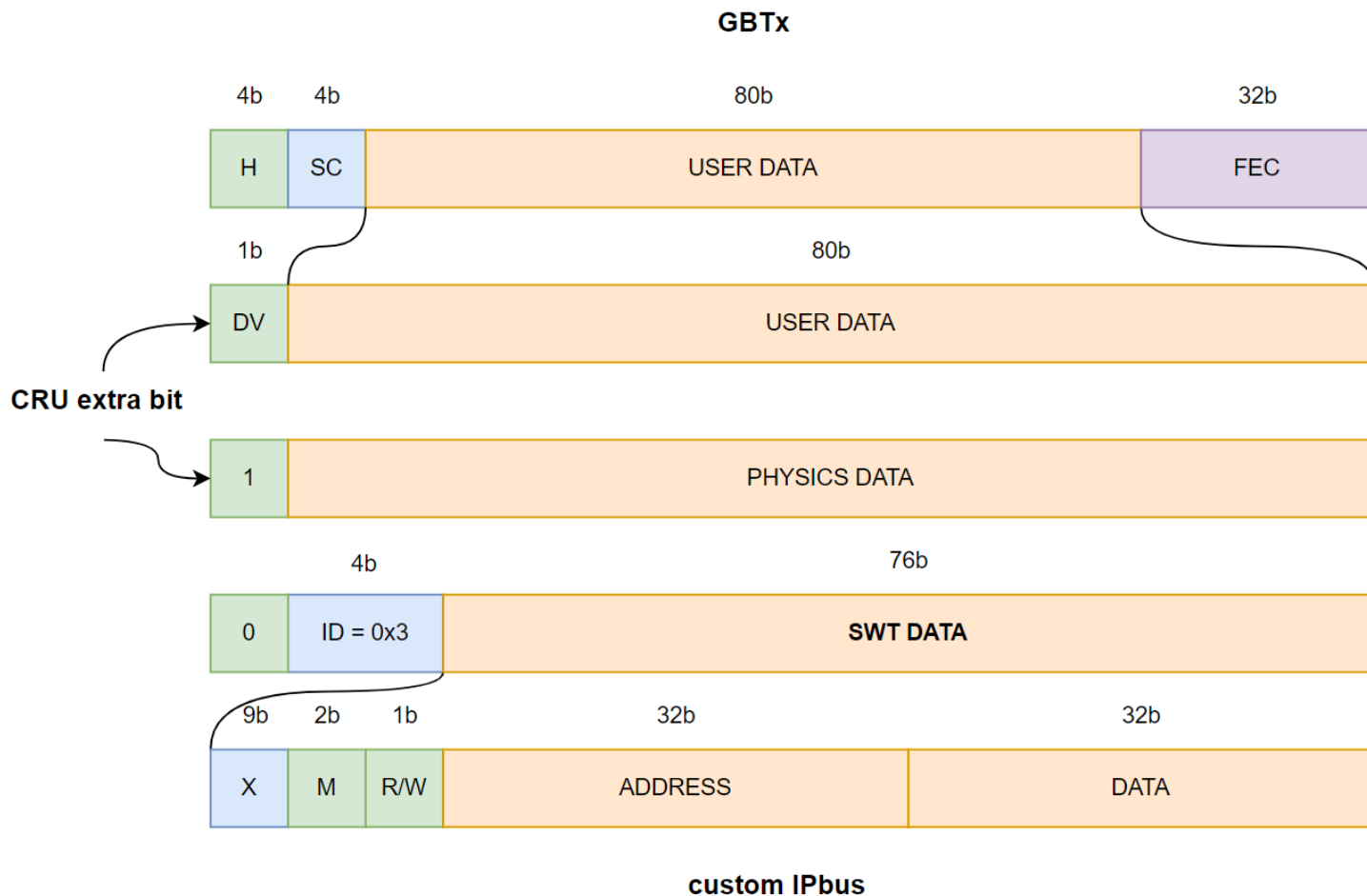
IPbus: <https://github.com/VictorPierozak/IPbus>

IPbus-SWT: <https://github.com/VictorPierozak/lpbusSWT/tree/master>

ALF: <https://github.com/VictorPierozak/ALFIPbus>

FEE CONTROL SYSTEM – IPBUS AS SWT IN GBTX

- Custom solution.
- Reusable in both software and hardware implementations.



IPbus operations in SWT

MM R/W ADDRESS DATA -> RESPONSE

2b 1b 32b 32b -> 32b

READ non-inc (FIFO)

MM R/W ADDRESS DATA -> RESPONSE

00 0 ADDRESS DONTCARE -> X
00 0 ADDRESS DONTCARE -> X

READ inc

MM R/W ADDRESS DATA -> RESPONSE

00 0 ADDRESS DONTCARE -> X
00 0 ADDRESS+1 DONTCARE -> X

WRITE non-inc (FIFO)

MM R/W ADDRESS DATA -> RESPONSE

00 1 ADDRESS DATA -> OK
00 1 ADDRESS DATA -> OK

WRITE inc

MM R/W ADDRESS DATA -> RESPONSE

00 1 ADDRESS DATA -> OK
00 1 ADDRESS+1 DATA -> OK

RMW bits $X \leftarrow (X \& A) \mid B$

MM R/W ADDRESS DATA -> RESPONSE

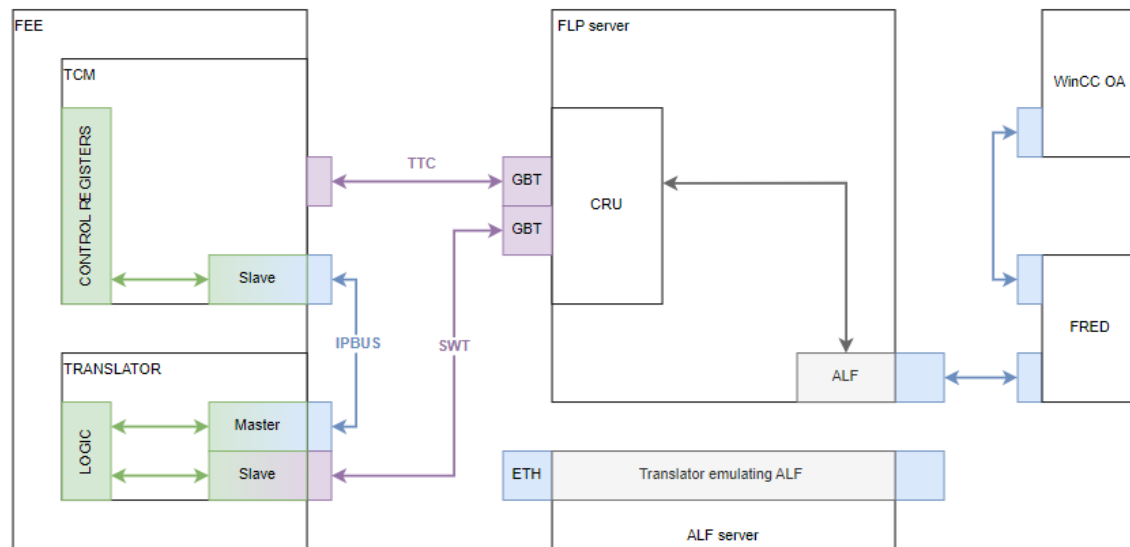
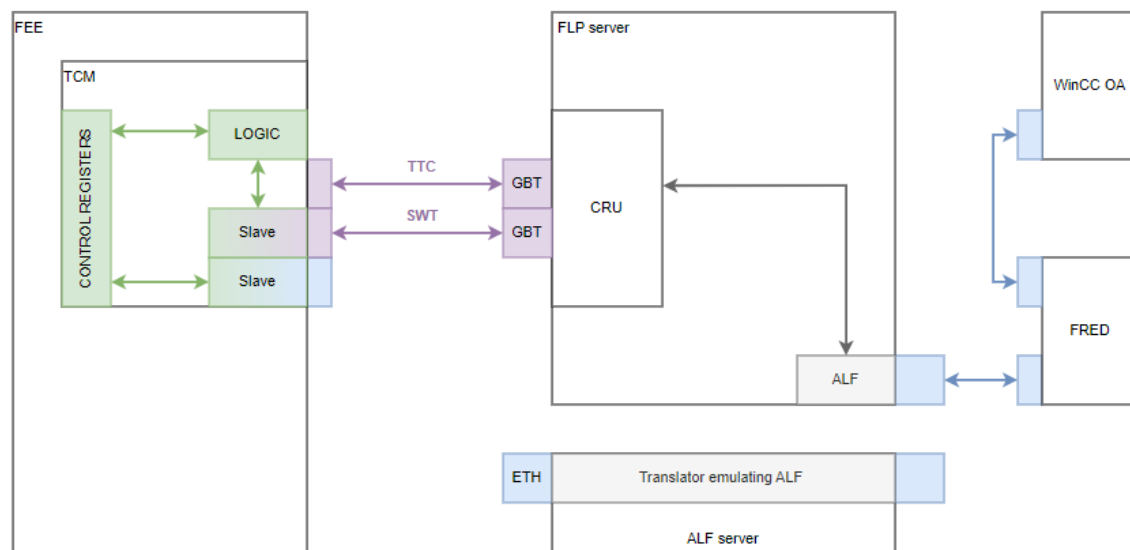
01 0 ADDRESS A -> preX
01 1 ADDRESS B -> OK

RMW Sum $X \leftarrow (X + A)$

MM R/W ADDRESS DATA -> RESPONSE

10 0 ADDRESS A -> preX

FEE CONTROL SYSTEM – HARDWARE SOLUTION



Compatibility with WinCC and FRED software implementation.

ALF on FLP will see only SWT frames.

Possible solutions:

1. GBT-SWT on TCM:

- TCM version 1.1 (for each detector) – small modifications
- Additional SFP+ gate on TCM
- Additional Clock generator for MGT in FPGA, or second output from the existing one (Si5338A)
- Custom logic in FPGA gateway on TCM
- Backward compatibility

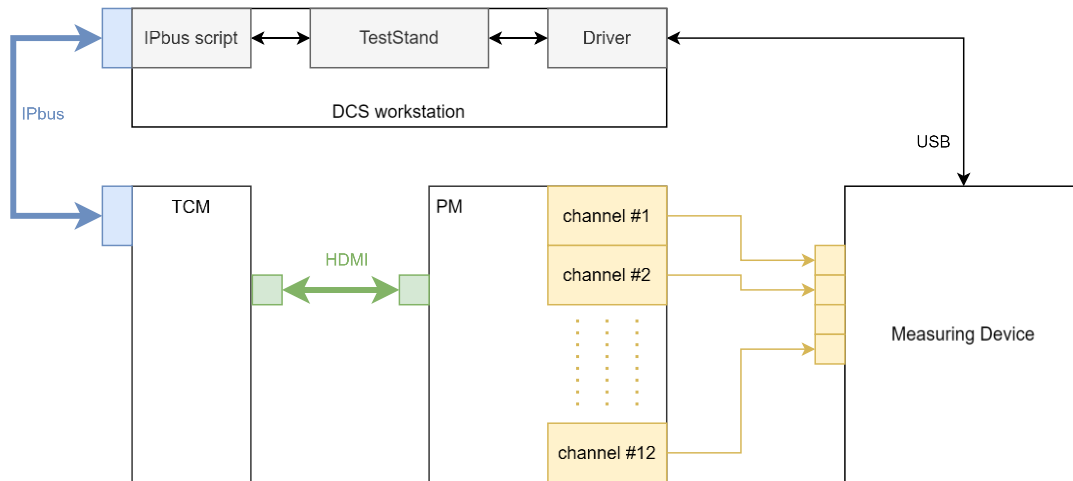
2. Custom translator:

- Extra PCB in Wiener crate
- TCM form factor
- One translator for all 3 detectors (possible more for future)
- Custom logic in separated FPGA
- Backward compatibility

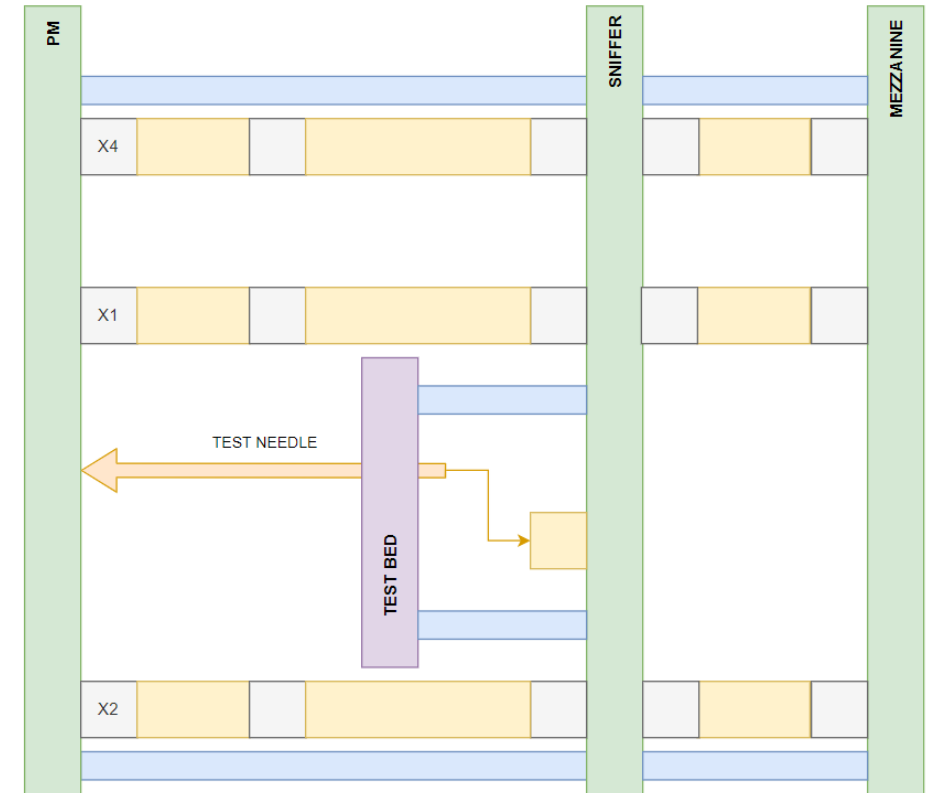
QUESTION: Will there be TCM v1.1? (PW)

SNIFFER concept:

- Two channel board can be used alone, or in parallel with any amount of channels
- Custom 3D printed test bed can be used as a test jig with pogo pins for test points on PM and CFD side
- use case #1: test new mezzanine concepts with current PM
- use case #2: test PM and CFD boards after production
- use case #3: test modified PM



1. Diagram of test setup.

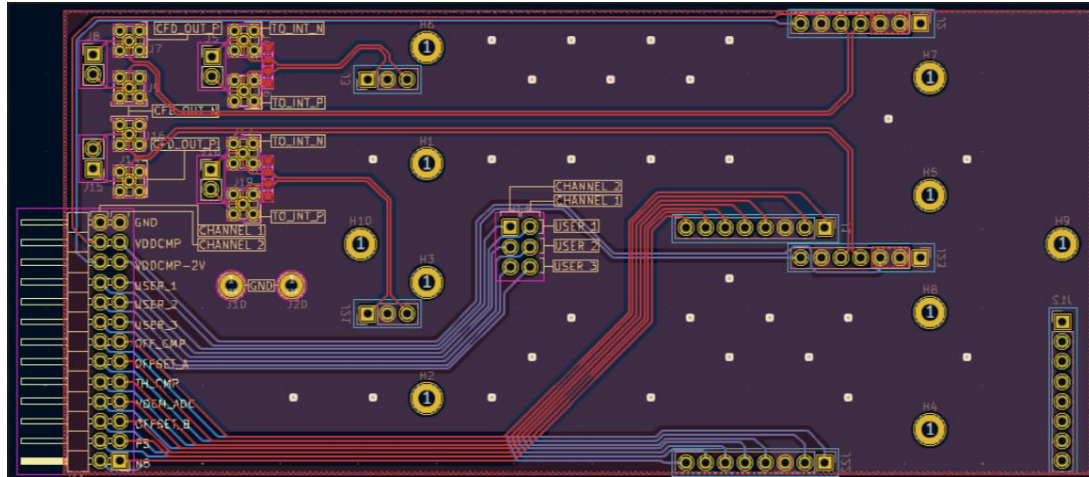


2. Diagram of SNIFFER and TEST BED placement between PM and mezzanine board.

A SET OF TEST AUTOMATIONS FOR PM - CFD

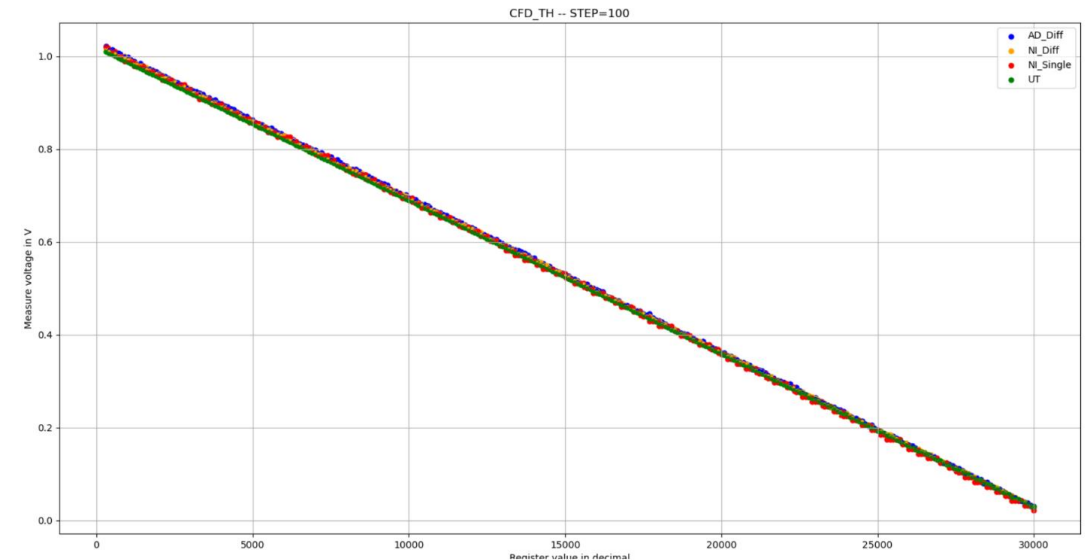
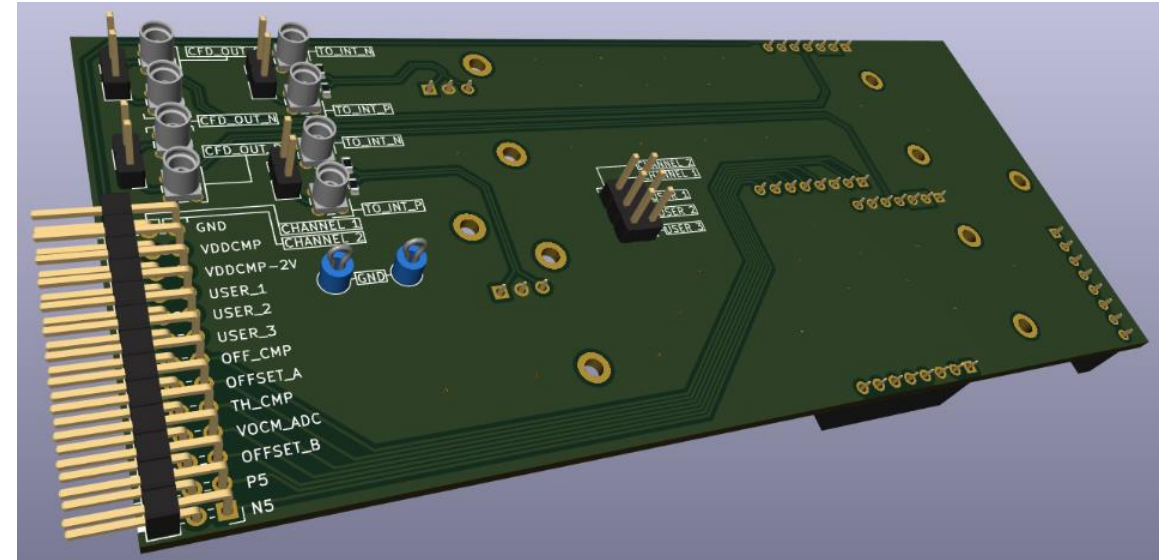
SNIFFER v1.0:

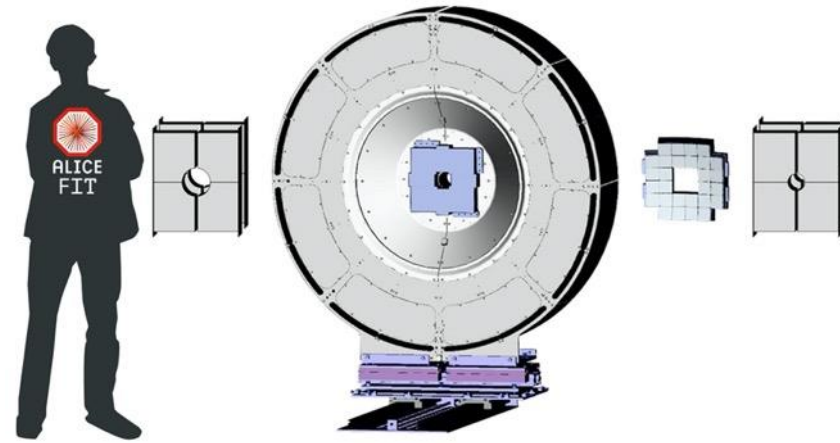
- Only analog outputs.
- Dedicated software in python for test automations.
- IPbus in python for compatibility.



SNIFFER v2.0 (in development):

- ADC build-in
- RS232 communication
- Compatibility with WinCC panels





THANK YOU FOR YOUR ATTENTION !