Pymosa: A smaller and faster Mimosa26 telescope redout

Y. Dieter, J. Eschweiler, T. Hemperek, T. Hirono, J. Janssen, Y. Liu, D.-L. Pohl

Plans and Status

Legend:

- Bonn milestone — Readyness in %
- DESY (?) milestone — Readyness in %

1. **Legacy mode**: Like current readout, 2 frames hardware trigger veto, send two frames per TLU trigger → Same speed, good for cross checking setup

- EUDAQ 1.x / 2.0 producer — 40 %
- EUDAQ 1.x / 2.0 converter — 0 %
- Documentation — 40 %
- (Adjustable) TLU trigger veto — 90 %
- Mimosa26 tuning — 0 %
- EUDAQ 1.x / 2.0 Integration (online monitor etc.) — 0 %

2. **Fast mode**: ~ 1 frame hardware trigger veto, send one frame per TLU trigger → Double the speed

- Create start of frame time stamps — 90 %
- TLU to frame time stamps correlation — 60 %
- Enhance of Mimosa data interpreter — 70 %
- System tests with particle beam — 40 %

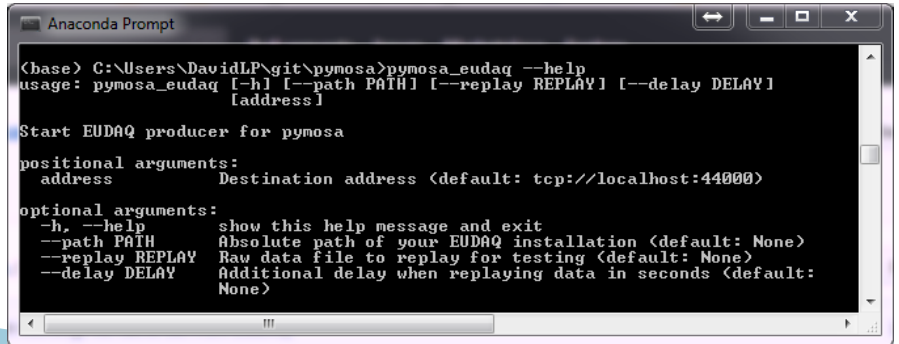
3. **Continuous mode**: No trigger veto, sending hits multiple times per trigger or use requested trigger range feature of EUDAQ2, event building in software → one order of magnitude more speed

- Realtime data duplication — 0 %
- System tests with particle beam — 40 %
- EUDAQ 2.0 Integration — 0 %

EUDAQ 1.x / 2.0 producer

- Fixing bug / add feature in Python interface for EUDAQ 1.x producer:
  https://github.com/eudaq/eudaq/pull/470
  https://github.com/eudaq/eudaq/pull/472
- EUDAQ 2.0 producer has much better Python integration (Yi Liu)
- Add python producer to pymosa: https://github.com/SiLab-Bonn/pymosa/blob/eudaq/pymosa/eudaq.py
- Replay data feature for testing and debugging (tbd)
- Real-time data interpretation for event sending (ongoing)
- Command line interface to ease usage: „pymosa_eudaq”

Example: command line interface



EUDAQ 1.x / 2.0 converter

- Since we have to do on the fly data interpretation we would like to send hits and not raw data
- Shall we encode hits as a „raw data event“ for EUDAQ 1.x?
- Python EUDAQ 1.x interface only supports raw data event sending, is there a hit data sending in EUDAQ 2.0?

Documentation

- To be hosted on the github project wiki: https://github.com/SiLab-Bonn/pymosa/wiki
- To help users to setup pymosa
- Starting to write installation / usage instructions here: https://github.com/SiLab-Bonn/pymosa/wiki/Eudaq-integration

Snippet from wiki page

Usage of pymosa

A simple command line interface is provided to start the pymosa producer:

```
pymosa_eudaq --help
```

Please read the help output for program parameters.

If you did not add the EUDAQ directory to the `PYTHONPATH` explicitly after installation (see above) you can give the path when running `pymosa_eudaq`, e.g.:

```
pymosa_eudaq --path /home/user/git/eudaq
```

TLU trigger veto

- Needed to fake triggered readout as it is done now
- It is faked triggered readout since we still read all Mimosa26 data
- We just send events with trigger; done in software and real-time
- Trigger veto in hardware implemented
- First test: saw expected reduced trigger rate due to veto

- Mechanism: Use trigger acknowledge signal from TLU FSM in FPGA
  - After accepting one trigger have to set acknowledge (indicate ready for next trigger)
  - Usually (continuous M26 readout): Set acknowledge immediately after accepting trigger
  - For **legacy/fast mode:** wait programmable time until trigger is acknowledged (2 x 115.2 us)
  - Result: Faked triggered readout with **1 trigger / 2 frames** (programmable)

Mimosa26 tuning

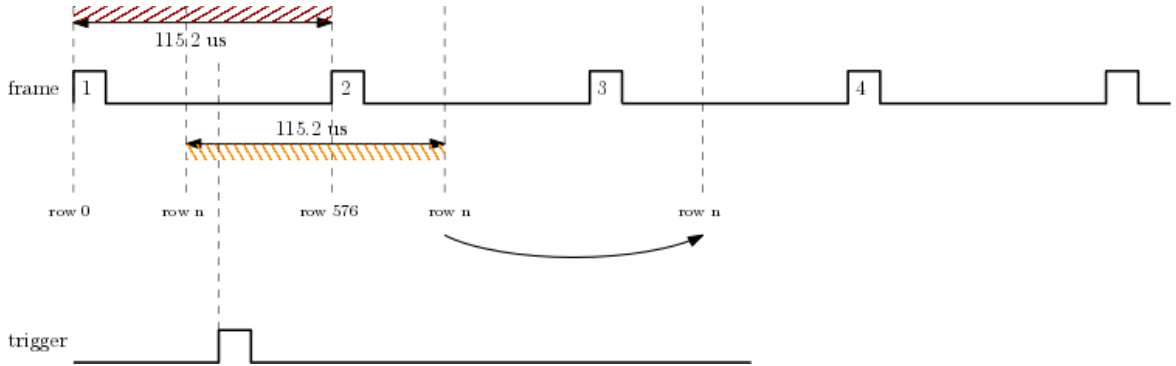
- Needed to be able to use new system with any telescope, when we do not want to translate the existing config files for all telescope
- Needed to mask noisy pixels that can deplete data bandwidth of continuous readout → main tuning goal is to reduce data rate
- Advantage: recover tuning degradation over time, „best“ possible tuning for the test beam conditions
- We are not really able to give the threshold setting in multiples of the noise value (as it is done now?)
- Tuning algorithm to be discussed and tested

Start of frame time stamps

- We create a time stamp in the raw data stream with 40 MHz when the new Mimosa26 frame readout starts
- Using the Marker (MKD) signal of Mimosa26
  - Four clock cycles high if new frame readout starts

Time stamps correlation

- We create a time stamp in the raw data stream with 40 MHz when the new Mimosa26 frame readout starts
- We create a time stamp in the raw data stream with 40 MHz when we have a TLU trigger word
- This allows us to assign with 115 us window a (or multiple) trigger to a Mimosa26 Row
- Proof of principle with test beam data at ELSA already done



Enhance of Mimosa data interpreter

- Make code readable
- Add unit tests for code quality
- Add code documentation
- Ongoing (Jens & Yannick)
- Test speed for real-time data interpretation

Real-time data duplication

- Since EUDAQ test beam data analysis is event based we can duplicate hits for overlapping events
- Needs a good time reference and offline data correlation in software to this reference
- Software work needed here