

VMM3a/SRS readout

Martin Purschke, BNL

Matt Posik, TU

The TL;DR:

We agreed a that I would work in integrating the VMM3a readout into our standard DAQ system, RCDAQ

I have received the actual crate and FEC a while ago from Matt

I literally got the front-end and remaining stuff delivered this very morning, so there's not so much to report on the actual readout front

I'll be away first at Fermi and then on vacation, this will resume after.

Let me show a few slides why we think that this is useful...

Why RCDAQ for the SRS systems?

It is no exaggeration that a good fraction of RCDAQ's adaptation outside of our usual RHIC/EIC circles is driven by the ready support for the SRS-APV25 system that is still in widespread use

About every other month I get some email asking about something by groups that heard about it, either by finding the github or by word-of-mouth, usually about the SRS readout

Of course the BNL/TU/SBU/... SRS - APV25 endeavors

Also TU Munich, Arkington Texas, SBU medical dept, and on and on

From total scratch it takes you about 40 minutes until you see an APV25 waveform

Ready-made example setups that use the test pulse to verify

Of course you get the really cool features of RCDAQ as well – online monitoring support, Elog support, environment-capturing features, and on and on

RCDAQ – the “Really Cool Data Acquisition”

- My flagship product
- Has been around for about 13,14 years
- It pre-dates sPHENIX but is the DAQ system of the experiment
- ECCE/Detector 1 DAQ proposal heavily informed by RCDAC
- In use in most EIC-themed test beams since 2013, LDRDs, etc etc
- I count roughly 60 EIC-themed incarnations
- Also in use in many other places in the world as I said (also ATLAS for some test beams)

Actually if you have access to *any* kind of Linux box (your laptop, a VM, a Raspberry Pi, and, yes, a RCAS machine), you can run RCDAQ yourself!

RCDAQ has a “pretend readout device” so to play you don’t need any actual hardware.

<https://github.com/sPHENIX-Collaboration/rcdaq>

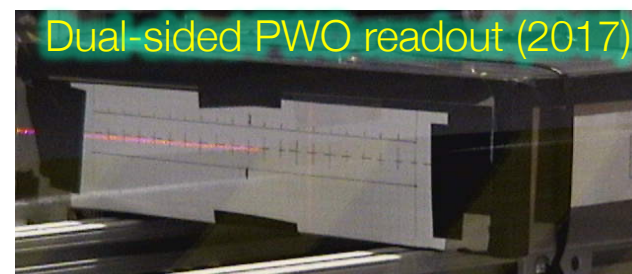
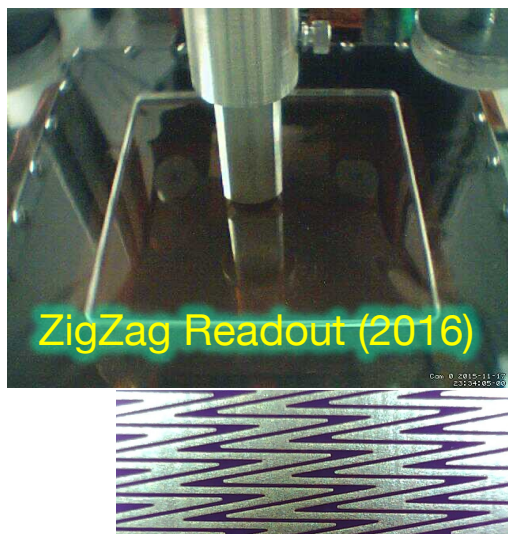
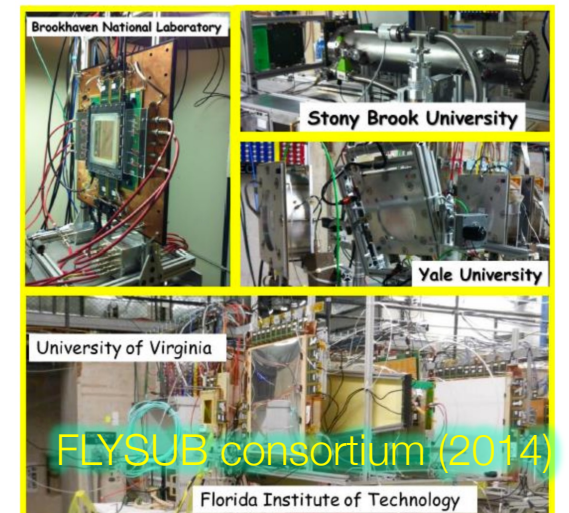
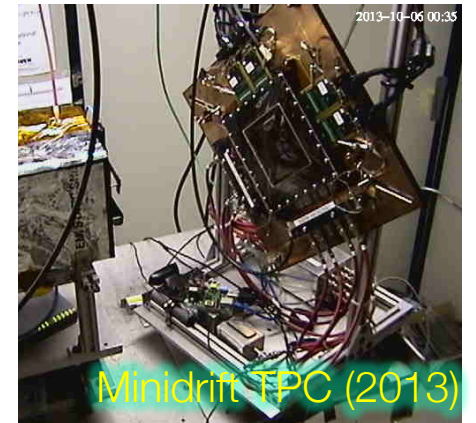
https://www.phenix.bnl.gov/~purschke/rcdaq/rcdaq_doc.pdf

Why is RCDAQ important for the EIC R&D?

Many lab measurements and test beam campaigns have been conducted with RCDAQ

In my recollection, 2013 was the first EIC-R&D appearance of RCDAQ at the Fermlabi test beam with the Minidrift TPC readout (SRS)

Especially that test beam (like next week) is sort of a well-established setup (all accelerator interfaces/positioning tables/Cherenkov ops are in place)



Again, why is the SRS support such a big thing?

Originally the SRS came “standard” with a stripped-down version of the ALICE DAQ system. Tricky setup, features you’d expect were hard to come by, analysis cumbersome

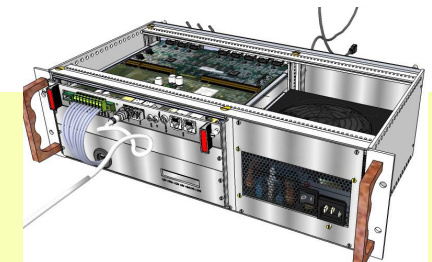
Here is how you set up a APV25-SRS at IP 10.0.0.2 for readout with RCDAQ:

```
rcdaq_client load librcdaqplugin_srs.so  
rcdaq_client create_device device_srs 1 1010 10.0.0.2 1
```

You load a plugin that teaches RCDAQ how a SRS system gets read out
Then create a readout device. Done.

Here is how we (2015 – FLYSUB @the FTBF) a DRS4 + a SRS system:

```
rcdaq_client load librcdaqplugin_srs.so  
rcdaq_client load librcdaqplugin_drs.so
```



```
rcdaq_client create_device device_drs -- 1 1020 0x10 -200 negative 400 1 0 1024  
rcdaq_client create_device device_srs 1 1010 10.0.0.2 1
```

You have the data, now what?

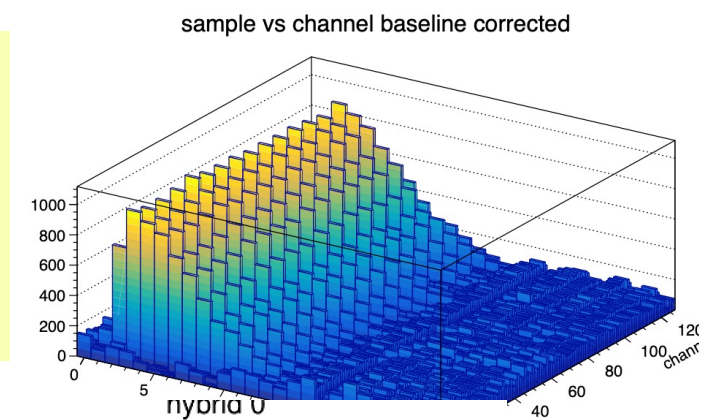
The analysis end comes with a well-defined set of data access APIs

You obtain a (C++) object that manages that device's data, and ask it questions

The APV25 is like a digital scope with 128 channels, from multiple APV cards

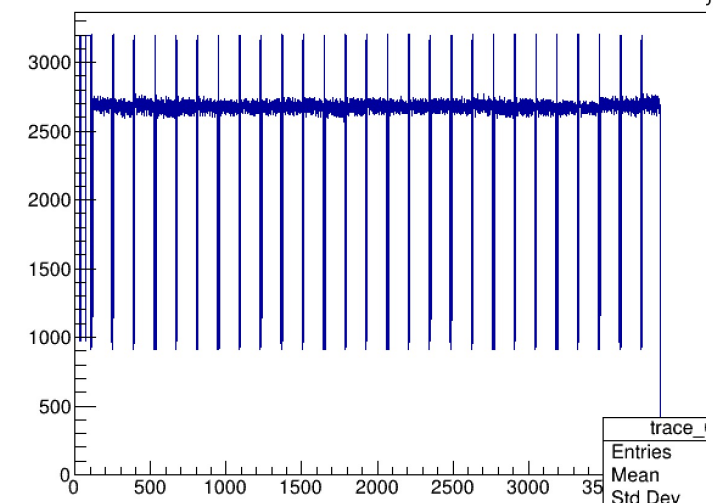
Here: fundamentally a 3-dimensional space – (channel, sample, card)

```
for (int i = 0; i < 28; i++)  
{  
    int this_sample = p->iValue (channel, i, card);  
}
```



You can also easily obtain the “FEC waveform” before decoding for timing in

As I said, from scratch it will take you less than one hour to get there.



Summary

I will try to get this same level of support (that is the plugin to read out the VMM3, and the corresponding data decoders)

This will be available to the community, of course.