pfRICH DAQ

Where are we?

- I had set up a new PC that we will use as DAQ platform
- Fast AMD CPU, 16 cores, ~6x as fast as what we are used to
- Lots of disk space (4x8TB ZFS RAID, 24TB usable
- Lots of PCIe slots
- All installed, quote from Alexander: “That was almost too easy”

Still work-in-progress – the busy latch

We need a meaningful (and fast) busy latch

Traditionally one uses a LeCroy 222 Gate generator set to infinity (latch)

Trigger signal comes in, sets the latch, edge of that latch trigger the DAQ

Now the latch is set, subsequent triggers are ignored (already set) until we reset it -> next trigger

That “reset the latch” is the real issue. In the old days, of a VME-based DAQ we had plenty of units that can just make a reset pulse

These days, it is fantastically hard for a PC to just make a hardware pulse

Our CAEN units can *in principle* do that, and we have an RCDAQ device for that – doesn’t work!

(CAEN calls their manuals that explain that “aspirational” – no joke.)

Next is a genuine on-board serial port (not a USB add-on), use the DTR line to make said pulse – this is the most viable solution right now.

I have been working on a FPGA-based busy latch, progress has been limited due to other commitments

Another alternative is a ESP32 microcontroller, easier than the FPGA

The problem with anything “external” is the latency to issue that reset, as you have to communicate with whatever unit.

Busy latch

The FPGA board is the Rolls-Royce version of such a latch, can in principle support a complete experiment's trigger needs with multiple sources etc

Indeed, this is a prototype of the sPHENIX trigger and timing board that did just that

This $6K unit is really total overkill for what amounts to a FlipFlop, but we have plenty of them

And it has the right (TTL-level) fast inputs and outputs, that's why I selected this

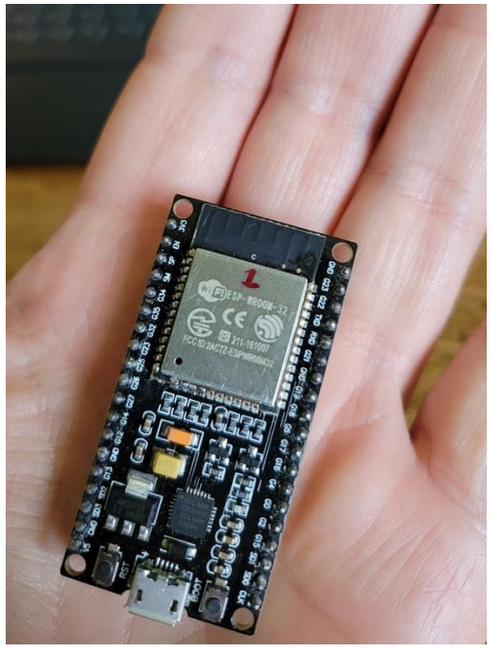
A spare unit in my office that we will pack

Alternative: A ESP32 "Wroom" microcontroller

I'm using those for basically everything that needs hardware interaction (my entire house automation (heat, AC, lights, garage door, ….works with those)

We'll see what's best and fastest.

At this point, we have the serial port as a viable fallback.



Final Concern: Where will the data go?

In order to meaningfully analyze the data, the data will need to be transferred to BNL soon-ish after they have been taken

We also cannot rely on the data being safe on the DAQ PC (it is in a rad-area after all)

We have extremely limited disk space at the SCDF (the new magnet test data will also have to go somewhere)

```
$ df -h  /eic/data/
Filesystem          Size  Used Avail Use% Mounted on
/gpfs/mnt/gpfs02  376T  370T  6.7T  99% /gpfs/mnt/gpfs02
```

We should triage the data on there if some space can be freed up.