Chris’ partial long-term list

DAQ:

* pulse-counting DAQ (I have it, I just haven't had time to play with it yet.)
* scalers (ditto)
* read proton current, chopper phase (is that in EPICS?)   into datastream
* XY-table - mostly done, check out W stage (and figure out where it got put away)
* run configuration log (part of DAQ GUI?)

Analysis:

* online -- determine interesting plots and data to show
    -- develop a system for comparing with golden run, and alarm if not the same
* online online -- check out WebOOT (maybe a strictly internal server, or use an external machine)
    -- maybe this could even replace the online system at the experiment
* library -- make TTreeRaw for new data-structure -- I'd like to brainstorm with everyone about improvements
    -- infrastructure for running scripts with different options (like runlists) from the command line (code exists)

Short-Term Goals

- ***Control Computer – UT order by Friday***
  - Parallel port
  - 2 ethernet ports (1 for bridge computer, another to communicate with ADCs)
  - 3 HDs to be safe (SSD for OS, RAIDed 1TB drives for /home)
  - Irakli/Kabir configure it – networking, establish communication with DAQ module
  - Write control GUI (can start with NPDGamma Perl TK) → take a run, specify #T0s end a run
    - Pull info from GUI into a file → runlist

- ***How to pull data onto Control computer***
  - HUB – private network (just one, instead of 4 like NDPGamma)
  - Ethernet? (isolated fiberoptic for the “dirty” module)

- ***What ‘s in the dirty module?***
  - SF-related stuff (current?)
  - Spin sequence
  - What else?

Short-Term Goals

- ***Headers for main data stream (proposed)*** **[to be implemented by D-tAcq]**

  5 Entries (preceding every 16ms pulse of data)

  (1) Integrity (0123ABCD)

  (2) Pulse # (relative to run start)

  (3) Time stamp

  (4) Microseconds

  (5) Integrity (0123ABCD)

Tbins – same as before?

Last week – seemed like 0.4 ms bins are fine to keep.
16 ms of data → adjust size of tbin slightly to grab more data ??? (can get 16.64 with .416 ms tbins)
Several ways to get there

Can we keep NPDGamma scheme? (path of least resistance)

Each of 4 main modules has 36x40 entries (36 wires, 40 tbins each), preceded by a header

16ms: 50kHz clock → samples taken every 20 us/800 samples per pulse/20samples/tbin

Once we settle on some version of the above:

Alter NPDGamma library to read new data structure (from wires – more branches will be added for new data, but this is the BIG place to start)

Add a branch that grabs info from the run file (so we know what kind of run it is)

By 11/20

1) Get and configure Picard
2) Define and implement data structure for “clean” modules
3) Minimal GUI to take data and write runlist file
4) Library to read data from “clean” modules
5) Headers written (Chris, is this doable quickly?)