Performance and Architecture Group

DUNE DAQ consortium

Group Kickoff Slides v0.0

26 September 2017

Giles Barr

Same slides shown at 3pm meeting Monday 25th

presented here briefly (just mentioning the first-pass reference design (slide 8), saying it is to define some terms to get going, and then showing the summary slide (silde 10)).

Introduction

- The five working groups of the DAQ consortium
  - P&A: Performance and Architecture – Giles B. + other.
  - Data selection, timing and calibration – Josh K.
  - Hardware – Matt G.
  - Computing – Kurt B.
  - Integration

Also, we have:

- DAQ and simulation – Georgia K. and Justo M.-A. S.

Conclusion #1 of this talk will be that the DAQ and simulations group will continue as-is. Keep doing what you do. You may get requests from other groups to extend plots or other studies. Make it bi-directional, i.e. also join some other groups and give your ideas to get them pointing in the right direction.

What role does performance and architecture fill?

Short term: Providing options and studies to inform the money-matrix and funding processes which end in the TP and TDR?

Medium term: Make sure that the bits we are planning for the TDR fit together well (horse, not camel).

Long term: Keep the design of the DAQ internally consistent through to implementation.

The next two questions are best answered together

Q1: OK, what do we do to get started? How can I help?

Q2: How does P&A connect with the other DAQ groups.

How I thought about these two questions:

- We have to find the best technical solution – and provide input into the search for funding that is happening collaboration-wide.
- Call [here] for new ideas to be presented (and written up).
- In parallel, start some detailed studies: Identify design paths (based on differences in designs we have so far RCE, FELIX, Dual-Phase, etc.)
  - "Max processing in FPGA" design path
  - "Max processing in computing" design path
  - "Cheapest FPGA, rest computing" design path
  - Compare computers at 4850l vs Computers on surface + big network
  - Timing distribution
  - Photon detector readout ← Urgent
- I have suggested what I think are the most urgent design studies on the next page, and also guessed which working group they may be in. Please comment, and think up other urgent and useful studies. Please volunteer to do one/some of them. (The principle is for the Performance and Architecture group to try to put into one of the other groups whenever possible).
- To aid discussion, i.e. so we are using the same words, I have tried to define a first-pass-reference design (FPR-design) starting from slide 5.

Detailed studies, here is a first go at a list....

"Max processing in FPGA" design path

- Tabulate amount of FPGA real-estate needed. E.g. trigger selection, compression; give ideas for cost. Physics studies on how many bits to use in filtering calculations.

"Max processing in computing" design path

- Invent a generic filtering algorithm in case of hi-noise scenario, test algorithms to get an idea how many computers we need.

"Cheapest FPGA, rest computing" design path

- List the minimal-FPGA schemes. Where does the opto-isolation go? Work on costs, power, how to do housing/cooling.

Compare computers at 4850l vs Computers on surface + big network

- Think out options for surface buildings (as a function of size of computers). Capacity of links up shaft.

Timing distribution

- Find out requirements from electronics and PDs

Photon detector readout

- Urgently need thinking more about Photon detectors: (Start from A. Himmel's talk to DAQsim group for single-phase info 10Jul2017). Physics questions: e.g. Do we need a fast-level trigger from PDs to buffer data? What photon detector data is needed in a SNB trigger that appears 3 secs after the start of the burst?

First attempt at 'first-pass-reference design' to aid discussion (1)

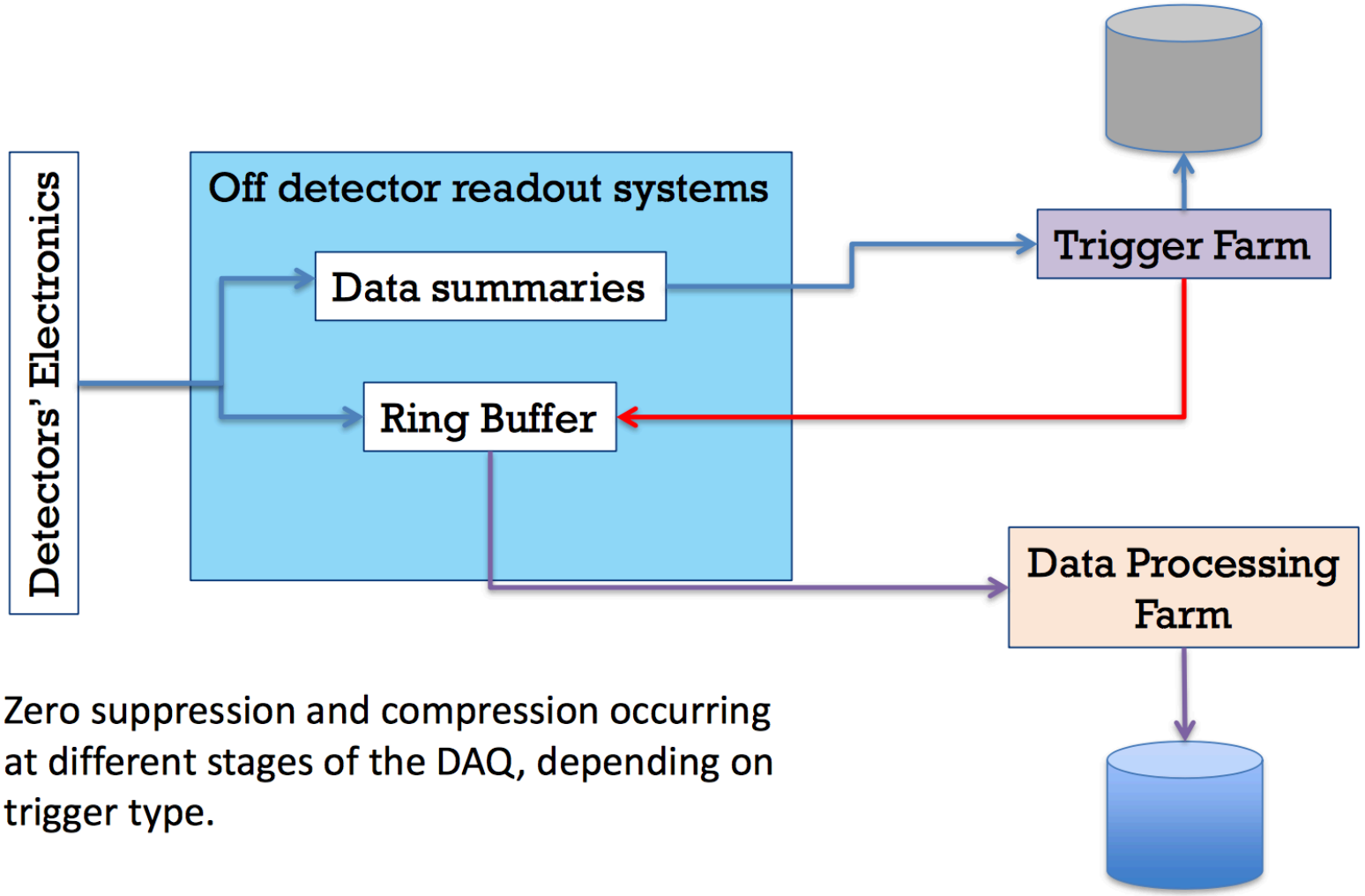
First attempt at defining a simplified 'reference', a set of coordinate axes to aid discussion (I am trying not to say 'baseline').

- Start with Giovanna's concept diagram from the Nov 2016 ProtoDUNE-SP DAQ review [next slide] (page 8 of https://indico.fnal.gov/getFile.py/access?contribId=1&resId=1&materialId=slides&confId=12392
- Then, by looking at Giovanna's diagram, we can define some concepts and expand them a bit to make a framework for discussion.

Note for people who are new and looking for documentation, the two DAQ design iterations we have written up for review are

1. from the Nov 2016 ProtoDUNE review, see https://indico.fnal.gov/conferenceDisplay.py?confId=12392)
2. the CD1 refresh technical DAQ review in May 2015 at https://web.fnal.gov/project/LBNF/ReviewsAndAssessments/Dune%20FD%20Design%20Review/SitePages/Review%20Agenda.aspx*

Sketch of the DUNE DAQ



Zero suppression and compression occurring at different stages of the DAQ, depending on trigger type.

First attempt at 'first-pass-reference design' to aid discussion (2)

By looking at Giovanna's diagram, we can define some concepts and expand them a bit to make a framework for discussion.

- Buffer: The noise in the single-phase TPC is too high to do lossy zero suppression, so the design must contain a memory buffer to store all the data for as long as it takes to make a trigger decision. Likely also the preferred solution for the dual-phase although because there is less noise, this is not so mandatory. Photon detectors - buffering architecture to be decided.
- Trigger feature finding: Initial hit finding must be done locally; it is impractical to examine all the data in one place. Design hit finding algorithms to cope with the specified S/N from the electronics groups; but also design in hooks when possible for future extensions to cope with unexpectedly higher noise, or pickup features. Need to evaluate and cost these extensions now (treat as risks for now).
- Distinguish 'Interaction triggers' from 'SN-burst triggers'. Interaction triggers work on quanta of data that are O(1-drift-time) or less and includes individual SN-interaction neutrinos, relic-SNs, cosmic rays, calibration triggers, beam triggers, atmospheric neutrino and proton decay candidates. 'SN-burst triggers' supplement this data collection by identifying candidate time-windows when an SNB is present and can cause complete readouts for a period of time (limited by the amount of money we have for buffer memory) and can cause periods when thresholds for keeping low energy events is reduced (to collect a good bite of the noise).

First attempt at 'first-pass-reference design' to aid discussion (3)

- Multiple levels for 'interaction triggers': (ordered by latency: low to high)
  - L1 - Hardware real-time fast trigger. This may be needed (essential?) for retaining detail of photon detectors, latency timescale based on buffering capability of this photon detector detail.
  - L2 - Software real-time trigger. Decision in trigger farm in O(t=1-drift-window) based on lists of locally found TPC hits and/or L1 triggers. This is the main trigger, and is what we have been considering in the software-simulation group so far. Raw TPC data retained in buffers until L2 decision made. Beam-spill trigger is inserted at L2.
  - L3 - Software trigger based on full data. Look at full data and optionally discard before first write to disk (in this sense it is still real-time). Can cookie-cut and compress also.
  - L4 - 'Offline' software processing prior to long term archival. It is after disk writing, so can be a batch farm.

  The important requirements for each level are max-latency, max-output-trigger-rate, input-data-spec.

L1 may go away in the future. If the requirements for Photon Detector readout and hardware designs allows buffer to be as long as for the TPC, then L1 functions will be done at L2

First attempt at 'first-pass-reference design' to aid discussion (4)

- Supernova Bursts: SNB is the most tricky physics to collect, because events are low energy and have very tiny photons surrounding that carry information. There are many components to this:
  a) In the L2 trigger farm, some software to spot when the number of low-energy events increases and to define supernova burst time windows (SNBTW) around it
  b) Control of buffers to save a time window of complete raw data based on SNBTW.
  c) Using a wider SNBTW, lower the L3 threshold for retaining a larger amount of low energy data than usual.

  (a),(b) and (c) address SNBs where our trigger has the capability of recognizing the SNB in real time based on our data alone. We must also study the efficacy of retaining the events at some stage (e.g. at L3 and/or retaining the trigger hits) for several hours in case we receive a SNEWS warning.

- Redundancy: Keep the four caverns independent so the DAQ cannot crash all at once and miss an SNB entirely. So we use Special network protocols to communicate between caverns, e.g. subscription sockets. [Pass info between caverns about: Increases in SNB candidate events, beam-spill trigger info, other triggers where we want four caverns simultaneously...]. Run control should NOT have one 'stop run' button, but four separate buttons.

Summary

- Time for new ideas to come forward – crucial, because if there is a very good idea we let pass us by, our job in 2022 will be harder.
- Also time to start doing studies for the options we have. I have made lists (summary on slide 4, detail on slide 12), use these to inspire study ideas of your own and let us know!
- Money matrix is opportunity to ask for what we need. But also danger that if we don't ask (or don't know we need yet) it will be difficult to find later. (e.g. a surface building)
- Important to sort out requirements ahead of engineering decisions
  - Noise
  - SNB
  - Physics triggers
  - Level of ROI/cookie cutting
  - What if we don't look at all of the detector?
  - Interfaces

Backup

Initial study list - draft

Max FPGA path

- Work from cost: List FPGAs, digikey price, features.
- Generic FIR-tap/DL alg: size of FPGA footprint
- Sweet-spot in width of multiply ops from HW perspective
- (side study) Do these studies also work with 'Net-FPGA' PCIe card in max-computing path?
- Physics study from multiply lane reductions
- Compression - summarise current state. Fixed max output block size, effect of noise, are x2 or x4 reasonable goals
- Compression. Noisy channel mitigation (zeroing low bits of noisy samples, downsampling to 1MHz noisy channels).
- Generic physics study on optimum ADC sampling rate. Pitch study as tradeoff between high rate and more detector volume.
- Generic physics study on optimum ADC sampling in PD readout. Pitch study optimising vs detector volume.
- Capture requirement: buffering for L1 trigger, buffering for SNB. List options for satisfying it.
- Study layout options for max-FPGA like scheme. Where are FPGAs? Where is cooling? Where is opto-isolation?

Cheapest FPGA, rest computing path

- List minimalistic FPGA schemes
- Work from cost: Study FPGA part-numbers for minimal schemes. Tradeoff between more or less parallel FPGAs based on cost
- Where is opto-isolation in cheap-FPGA schemes
- Summarise optimisation choices of link from FPGA to computer
- Multiplexing of links from FPGA board to computers: Best done on FPGA board or external Ethernet switches?

Computers 4850l vs surface + big network

- Options for surface buildings (for 100, 200, 500 and 1000 nodes at 500W each)
- Capacity of underground area 50 racks/4. + other underground options
- Methods of using 96/4 optical pairs to surface. DWDM or switching or ... Future evolution of technology.
- Strategy for upgrading from 96 pairs (i.e. figure out how to decide).

Max-computing path

- Invent generic FIR-tap/DL alg (for hi-noise)
- Contruct scheme for lo-noise
- Test simple and complex algs on GPUs
- Test simple and complex algs on CPUs
- Does compressed input affect CPU/GPU time
- Identify architectural simplifications (e.g. L1 trig farm splitting)
- Devise packet handling
- Devise comm-error recovery, do we need to end run every time it happens?
- Devise SNB buffer scheme (should be easy)

Photon detector requirement capture

- Review A. Himmel's talk in DAQSim group
- Ask questions about it
- Understand the dual-phase PhotDet design
- Come up with straw-man scheme for Photdet DAQ
- Come up with requirements for DAQ from PhotDet
- What photon detector data is needed for an SNB that triggers 3 secs after start of burst?
- Physics question: Do we need a fast level of triggering from PDs to buffer data? Specify latency range needed.
- Revise our straw-man scheme based on input
- Devise how PD scheme fits with rest of DAQ

Timing distribution

- Find out requirements from electronics, PDs
- Capture requirement: Must phase of 2MHz digitisation clock be same across APAs in cavern? [Much simpler for us if no!]
- Capture requirement: What is timing accuracy needed for TPC data (easy question, the worst option for synchronising is 100ns)
- Capture requirement: What is timing accurcy needed for photon detector data (harder question)
- Compare free-running local clock & local measurements of time markers v.s. sending global clock around with superimposed markers.
- Modularity: Choose between (a)->(b) or (b)->(a) Where (a) is distribute to each flange and (b) is split to TPC and PhotDet. Same for SP and DP?

All very preliminary lists, still under construction.