TDIS DAQ
Data throughput discussion

P.M. King

Ohio University

7 August 2017

What if we record everything?

- We expect the trigger rate from electrons in the SBS to be 6 kHz.
- The RTPC will have about 36000 pads (total of both U and V pads)
- We want to determine the track time with a resolution of ~10 ns
- Last time, I made a loosely-supported guess that electrons reaching one pad would have a time spread of 50 ns (based on an average track hitting ~400 pads)
- Suppose we divide the 20 us into 500 40-ns time bins
  - Suppose we record a 12-bit ADC value for each channel for each time-bin:

    (6000 kHz)*(36000 pads)*(500 bins/pad)*(1.5 byte/bin) = 162 GB/s
    - The event size is 27e6 bytes

How many time bins are interesting?

- Assume we're using 500 40-ns time bins
  - Suppose we need 3 bins to identify the peak, and 10 to fully contain a pulse (guess based on Micromegas design)
- Using the rates and occupancies per layer (18000 pads/layer) from Rachel's "method 1" (slide 10 from talk on 12 July 2017)
  - 1H: (2.3 MHz)*(20 us)*(176 pads/track)/(18000 pads) = 0.45 hit/pad
  - 2H: (357 MHz)*(20 us)*(176 pads/track)/(18000 pads) = 69.8 hit/pad
- If we need 10 time bins per hit, we need an average of 4.5 bins out of 500 for 1H: (162 GB/s)*(4.5/500) = 1.46 GB/s
  - But for 2H, we would expect pulses to be overlapping on their tails, and just selecting interesting bins wouldn't reduce the 162 GB/s

What about time/amplitude only?

- Recalling the rates and occupancies per layer in “method 1”
  - 1H: (2.3 MHz)*(20 us)*(176 pads/track)/(18000 pads) = 0.45 hit/pad
  - 2H: (357 MHz)*(20 us)*(176 pads/track)/(18000 pads) = 69.8 hit/pad
- Suppose we have a 16-bit amplitude and 16-bit time recorded for each hit
  - 1H: (6000 kHz)*(36000 pads)*(0.45 hits/pad)*(4 bytes/hit) = 0.39 GB/s
  - 2H: (6000 kHz)*(36000 pads)*(69.8 hits/pad)*(4 bytes/hit) = 60.3 GB/s

How many tracks have “good” timing?

- Suppose we can identify tracks which are generated within a 50 ns window of the trigger; these tracks have one end at the longest time and the other at the shortest time (or curl back to the longest time)
  - Isolating all “off-time” tracks and discarding their hits would give data reduction of ~400 (for 50 ns window out of 20 us)
  - This would require at least doing basic track finding, if not the full track finding
    - Late recurved tracks may be tricky, as they probably look like on-time tracks but with an offset vertex.

What if we do full track reconstruction?

- Total number of tracks, using “method 1”
  - 1H: (2.3 MHz)*(20 us) = 46 tracks in RTPC
  - 2H: (357 MHz)*(20 us) = 7140 tracks in RTPC
- With the 6 kHz trigger rate, the 2H would be giving us 42.8 million tracks per second.
  - Twelve 16-bit parameters saved per track for the 2H target would give a data rate of 1 GB/s
  - I’m not sure if that would be sufficient to describe the tracks, and if we did this (and discarded all hits), we would have no ability to redo the tracking