MU2E/PHYSICS/xxxxx
version 1.0 February 13, 2024

Commissioning of the Mu2e Data AcQuisition system and the Vertical Slice Test of the straw tracker

11. An artdaq Demo tests

Pavel Murat (FNAL)
Sara Gamba (University of Pisa)
version 1.0 February 13, 2024

Abstract

This note presents the initial results of an analysis on the artdaq rate.

Contents

1 Notes for the authors

1.1 Revision history

- v1.01: initial version

2 Running an artdaq demo

We were running an artdaq demo simulation to test if the artdaq supports high event rates: we should be able to get data at a rate of more than 200MB/s. This demo is called ToySim and it is taken from $ots\_mu2e\_tracker$, adding a $Generators$ directory. This directory is very similar to $artdaq\_demoartdaq-demoGenerators$ directory. We were sending some random events to two boardreaders. These events were taken by two eventbuilders that have a common data logger. A dispatcher, which aggregates DQM metrics and presents them to a visualizer application, was used. We were trying to run the demo changing some parameters:

- time between events: the function called $FillBuffer(buffer, bytes\_read)$ is called every time after $throttle\_usecs$ μs;
- size of the events: $nADCchannels$ is the variable we are changing and it is used to define how many bytes are readout ($bytes\_read = sizeof(demo :: ToyFragment :: Header) + nADCchannels \times sizeof(data\_t)$, so the dimension will be this number × 2Bytes more or less);
- number of eventbuilders (1 or 2);
- number of boardreaders (1 or 2);
- the presence of a dispatcher;
- the transferPluginType: $Shmem$ and $TCPSocket$.

If the artdaq framework is able to process this events, we should see a $GetNext$ Frequency equal to the inverse of the $throttle\_usecs$ μs value. We have tried to change the variables $nADCchannels$ and $throttle\_usecs$.

3 Results

During the running some errors appeared, as the following:

- Bad Omen: Data Buffer has exceeded its size limits.
  (seq_id=125, $frag\_id$=0, frags=1001/1000, szB=200248048/1048576000), timestamps=124-1124 with this type of error the rate is not stable;
- Back-pressure condition: All Shared Memory buffers have been full for 12.025 s!

We report some tables that show our results.

bytes	throttle_usecs	result
200k	0	BAD OMEN & back pressure
200k	10 (100kHz)	BAD OMEN & back pressure
200k	100 (10kHz)	BAD OMEN & back pressure
200k	1000 (1kHz)	BAD OMEN & back pressure
200k	10000 (100Hz)	BAD OMEN & back pressure
200k	100000 (10Hz)	BAD OMEN & back pressure

Table 1: We tried to change rates and to use a fixed number of bytes (200kB). Results: we cannot operate with this event size.

bytes	throttle_usecs	result
100k	1000 (1kHz)	BAD OMEN
100k	2000 (500Hz)	BAD OMEN
100k	4000 (250Hz)	BAD OMEN
100k	5000 (200Hz)	BAD OMEN
100k	6000 (166Hz)	OKAY: rate 163Hz
100k	10000 (100Hz)	OKAY: rate 98.5Hz

Table 2: We tried to change rates and to use a fixed number of bytes (100kB). Results: at 20MB/s more or less it gets errors.

bytes	throttle_usecs	result
40k	2000 (500Hz)	BAD OMEN
40k	3000 (333Hz)	OKAY: rate 320Hz
40k	5000 (200Hz)	OKAY: rate 195Hz
40k	10000 (100Hz)	OKAY: rate 98.7Hz

Table 3: We tried to change rates and to use a fixed number of bytes (40k). Results: at 20MB/s more or less it gets errors.

bytes	throttle_usecs	result
70k	5000 (200Hz)	OKAY: rate 195Hz
70k	4000 (250Hz)	OKAY: rate 242Hz and BAD OMEN after 1 m
70k	3000 (333Hz)	BAD OMEN

Table 4: We tried to change rates and to use a fixed number of bytes (70k). Results: at 20MB/s more or less it gets errors.

We have tried to change also the other variables listed before, but nothing else changed. As we can see it is not possible to run artdaq at rates higher than 20MB/s and this needs to be fixed.