Weekly Meeting

June 28 2017

Calibration Meeting Last Week

- Introduced the IO Library to the calibration people
- Seemed to go well – people were pleased that they could use the IO Library in their DAQ
- Two major questions arose:
  - Can the IO Library write to file?
  - How flexible is the data format?

Calibration Meeting Last Week

Can the IO Library write to file?

- No.
- People like the idea of the IO Library being able to write to file
  - Makes DAQ much simpler from user's standpoint
- People don't like the idea of the IO Library having MIDAS dependence

- Overlaps with questions I proposed: should the DMC have MIDAS dependence in order to write to file?

Calibration Meeting Last Week

How flexible is the data format?

- “The data format is not written in stone, but it’s written in hardening cement”
- For instance, if they want to be able to add information about muon veto – where would it go?
- Same implications for DMC – there is lots of information that needs to be included somewhere

Calibration Meeting Last Week

How flexible is the data format?

- One proposal: add extra data blocks

Author and Contact: Belina von Krosigk, UBC

DATA FORMAT VERSION 1: Created: 05. Feb. '16, Last updated: 30. Jun. '16

		bits	31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
			0x9 | format version=1 | total n triggers read
x N triggers			0x5 | event size in bytes
			trigger ID
			trigger type
			global timestamp low
			global timestamp high
			0x7 | n primitives in event
			length of entry (=0x6 block) in bytes
	x N prims		0x6 | trig status | pileup | detector id | index
			UT at which rt was issued
			time fraction rt was run (100nsec/count)
			time of trigger in sec | time rt was run in sec
			mask pairs | time fraction of trigger (100nsec/count)
			trigger word | peak amplitude
			0x3 | n detectors in event
	x N dets		0x2 | detector type | detector id | index
			DCRC1 serial number | DCRC1 version | DCRC0 serial number | DCRC0 version
			0x4 | readout status | series time in sec
			series time fraction (100nsec/count)
			0x0 | n channels to follow
		x N channels	0x1 | pre-trigger offset (22 bits) | ch num | ch type
			n pre-pulse samples
			n on-pulse samples
			n post-pulse samples
			sampling rate high in kHz | sampling rate low in kHz
			samp1 | samp0
			samp3 | samp2
			⋮
			sampN | sampN-1
			0x8 | total n preceding triggers
			0x11 | DMC Header
			DMC Data
			trigger type
			DMC Data
			DMC Data

DAQ Meeting this Week

- Belina, Amy and I will start a discussion about IO Library
  - Should the Library write to file?
  - How flexible is the data format? How should that extra data be added?
- Possible follow-up meeting on Friday

DMC Data

- I compared the data from the Soudan binary writer to the SNOLAB data format

SNOLAB Format Data	DMC Equivalent?		DMC Soudan Binary Data			SNOLAB DAQ Equivalent?	
format version	N/A		01020304	header word 1, endianess?		No	
total trigger read (num events)	N/A		data minor	header word 2		No	
event size bytes	Maybe		data major			No	
trigger ID	Yes		daq minor			No	
trigger type	Maybe		daq major			No	
global timestamp	Yes		detector config record ID	00010000		No	
num primitives	No		detector config record Length	(8*52 + 4*40)*numDetectors		No	
prim size bytes	No		phonon channel header	00010001		N/A	
trig status	No		phonon channel length	44		No	
pile up	No		detector code	calculation		Yes	detector ID
detector ID	Yes		tower array	1		No	
index	No		phonon driver gain	loop gain feedback*100		No	
unixtime	Maybe		QET bias	calculation		No	
rt time frac	Maybe		SQUID bias	0	FILE CONFIGURATION HEADER	No	
trigger time	Maybe		SQUID lock point	0		No	
rt time	Maybe		RTF phonon trigger offset	0		Maybe	pre-trigger offset?
mask pairs	No		phonon var gain	loopGainSquid'		No	
trigger time frac	Maybe		delta T phonon	batsTime_dt_p*1e9		No	
trigger word	No		t0 phonon	batsTime_timeOffset_p*1e9		Maybe	series time/trigger time?
peak amplitude	No		trace length phonon	batsTime_totalIter_p*1e9		No	
num detectors	N/A		charge channel header	00010002		N/A	
detector type	No		charge channel length	32		No	
detector ID	Yes		detector code	calculation		Yes	detector ID
index	No		charge driver gain	q_drivergain*100		No	
DCRC1 serial number	No		charge bias			No	
DCRC1 version	No		delta T charge	batsTime_dt_q*1e9		No	
DCRC0 serial number	No		t0 charge	batsTime_timeOffset_q*1e9		Maybe	series time/trigger time?
DCRC0 version	No		trace length charge	batsTime_totalIter_1*1e9		No	
readout status	No		event type	9		Maybe	trigger type?
series time in sec	Maybe		event class cat	0	EVENT CONGIFURATION HEADER	No	
series time frac	Maybe		event ID			Maybe	trigger ID
num channels	N/A		event length	in bytes?		Maybe	event size bytes?
pre-trigger offset	Maybe		logical Record type	00000002		No	
channel num	Yes		logical record length	4*6		No	
channel type	Yes		logical record series num1	SN1		No	
pre pulse length	Maybe		logical record series num2	SN2	ADMINISTRATIVE RECORD	No	
on pulse length	Maybe		logical record event number	calculation, event within file?		Yes	Trigger ID
post pulse length	Maybe		logical record event time	3329627852		Maybe	series time/trigger time?
sampling rate high	No		logical record time since last event	0		No	
sampling rate low	No		logical record live since last event	0		No	
waveform	Yes		logical record type	0000011		No	
num preceding triggers	No		logical record trace length	calculation		No	
			logical record book keep head	00000011		N/A	
			logical record book keep len	12		No	
			logical record dig base add	0		No	
			logical record dig channel	0		No	
			logical record detector code	calculation		Yes	detector ID
			logical record time base head	00000012	FOR EACH CHANNEL	N/A	
			logical record time base len	12		No	
			logical record t0	batsTime_timeOffset_q/p*1e9		Maybe	series time/trigger time?
			logical record delta T	batsTime_dt_q/p*1e9		No	
			logical record num Points	batsTime_toalIter_q/p		Maybe	num samples?
			logical record trace head	00000013		N/A	
			logical record num samples	batsTime_totalIter_q/p		Maybe	num samples?
			data array	waveform data		Yes	waveform data

DMC Data

- Some of this data is written to file at another stage
  - E.g. QET Bias, SQUID Bias, are parameters in ODB, that are written out when ODB gets written out
- Some data I’m not sure what it means
  - E.g. Data Major, Data Minor?
- There is SNOLAB data that has no DMC equivalent, but I’m guessing some of this will be placeholders
  - E.g. DCRC serial num and version
- Where is the DMC truth information? It doesn’t seem to be written out in this binary format.

C++/python extensions

- Been working on getting a python script to import a c++ shared library.
- Using SWIG to generate an extension/wrapper that python can read
- I’ve been able to get a simple example to work using function in a c code, but having problems with the IO Library (c++)
  - I can’t even get the same simple example to work using c++ code!
  - I can generate the shared library, but it doesn’t import properly

C++/python extensions

```
cdms_iolibrary.cxx  cdms_iolibrary.h  example.i  libcdms_iolibrary.so  main.py  test
```

mwilson@cdms:~/data_iolibrary/python_wrap_test

2. mwilson@cdms:~/data_iolibrary/py

```
 GNU nano 2.3.1                                    File: example.i

%module mylib

%{
#include "cdms_iolibrary.h"
%}

%include "cdms_iolibrary.h"
```

C++/python extensions

```
cdms_iolibrary.cxx  cdms_iolibrary.h  example.i  libcdms_iolibrary.so  main.py  test
```

```
[mwilson@cdms python_wrap_test]$ swig -python -c++ example.i
[mwilson@cdms python_wrap_test]$ ls
cdms_iolibrary.cxx  cdms_iolibrary.h  example.i  example_wrap.cxx  libcdms_iolibrary.so  main.py  mylib.py  test
```

Should generate example.py

C++/python extensions

```
cdms_iolibrary.cxx  cdms_iolibrary.h  example.i  libcdms_iolibrary.so  main.py  test
```

```
[mwilson@cdms python_wrap_test]$ swig -python -c++ example.i
[mwilson@cdms python_wrap_test]$ ls
cdms_iolibrary.cxx  cdms_iolibrary.h  example.i  example_wrap.cxx  libcdms_iolibrary.so  main.py  mylib.py  test
```

```
mwilson@cdms python_wrap_test]$ g++ -std=c++11 -fPIC -c cdms_iolibrary.cxx example_wrap.cxx -I/usr/include/python2.7
mwilson@cdms python_wrap_test]$ ls
dms_iolibrary.cxx  cdms_iolibrary.h  cdms_iolibrary.o  example.i  example_wrap.cxx  example_wrap.o  libcdms_iolibrary.so  main.py  mylib.py  test
```

```
mwilson@cdms python_wrap_test]$ g++ -std=c++11 -fPIC -shared cdms_iolibrary.o example_wrap.o -o _example.so
mwilson@cdms python_wrap_test]$ ls
dms_iolibrary.cxx  cdms_iolibrary.o  _example.so       example_wrap.o        main.py   test
dms_iolibrary.h    example.i         example_wrap.cxx  libcdms_iolibrary.so  mylib.py
```

C++/python extensions

```
[mwilson@cdms python_wrap_test]$ python
Python 2.7.5 (default, Nov  6 2016, 00:28:07)
[GCC 4.8.5 20150623 (Red Hat 4.8.5-11)] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import example
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: No module named example
>>>
```

Go into mylib.py, replace all _mylib with _example, and change name from mylib.py to example.py

```
>>> import example
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "example.py", line 26, in <module>
    _example = swig_import_helper()
  File "example.py", line 22, in swig_import_helper
    _mod = imp.load_module('_example', fp, pathname, description)
ImportError: dynamic module does not define init function (init_example)
```

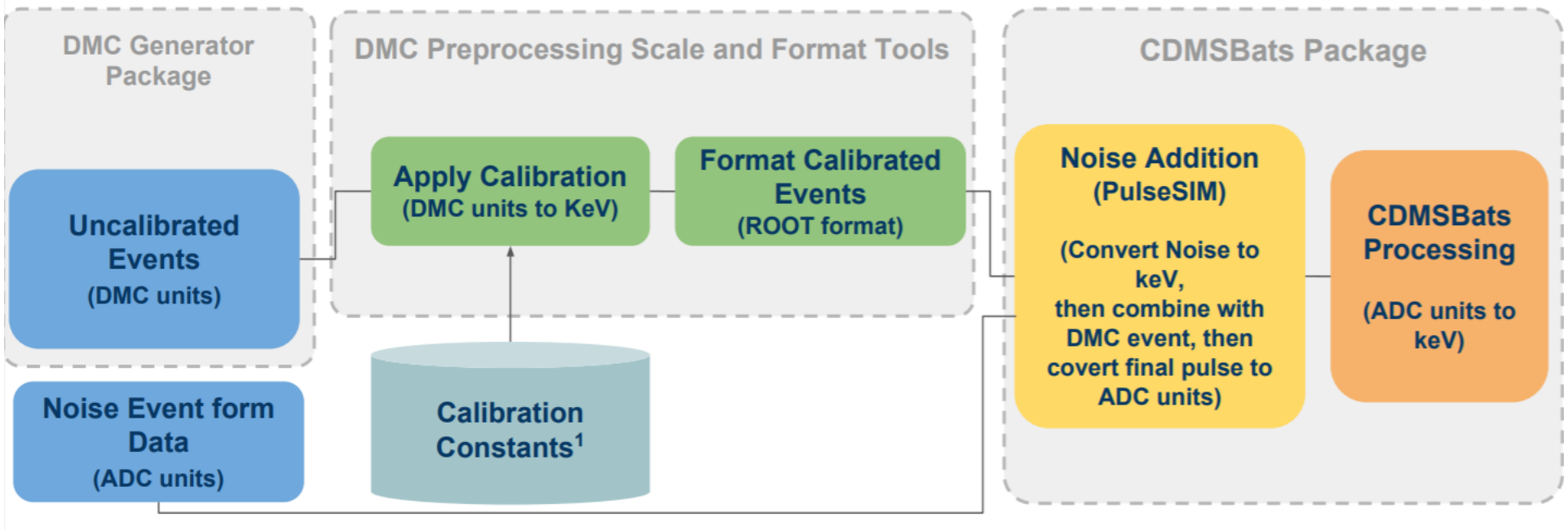
C++/python extensions

With basic c++ example (taken from the simple example that works)

```
>>> import example
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "example.py", line 26, in <module>
    _example = swig_import_helper()
  File "example.py", line 22, in swig_import_helper
    _mod = imp.load_module('_example', fp, pathname, description)
ImportError: ./_example.so: undefined symbol: __gxx_personality_v0
>>>
```

Other News

- Leave on Sunday, back on the 18th
  - Will have access to emails, may be slow to respond
- UBC-TRIUMF-Collaboration Meeting trip booked
- There is a new grad student (I think at A&M) who is specifically working on noise integration into DMC


DMC Generator Package
DMC Preprocessing Scale and Format Tools
CDMSBats Package
Uncalibrated Events
(DMC units)
Apply Calibration
(DMC units to KeV)
Format Calibrated Events
(ROOT format)
Noise Addition
(PulseSIM)
(Convert Noise to keV, then combine with DMC event, then covert final pulse to ADC units)
CDMSBats Processing
(ADC units to keV)
Noise Event form Data
(ADC units)
Calibration Constants[1]