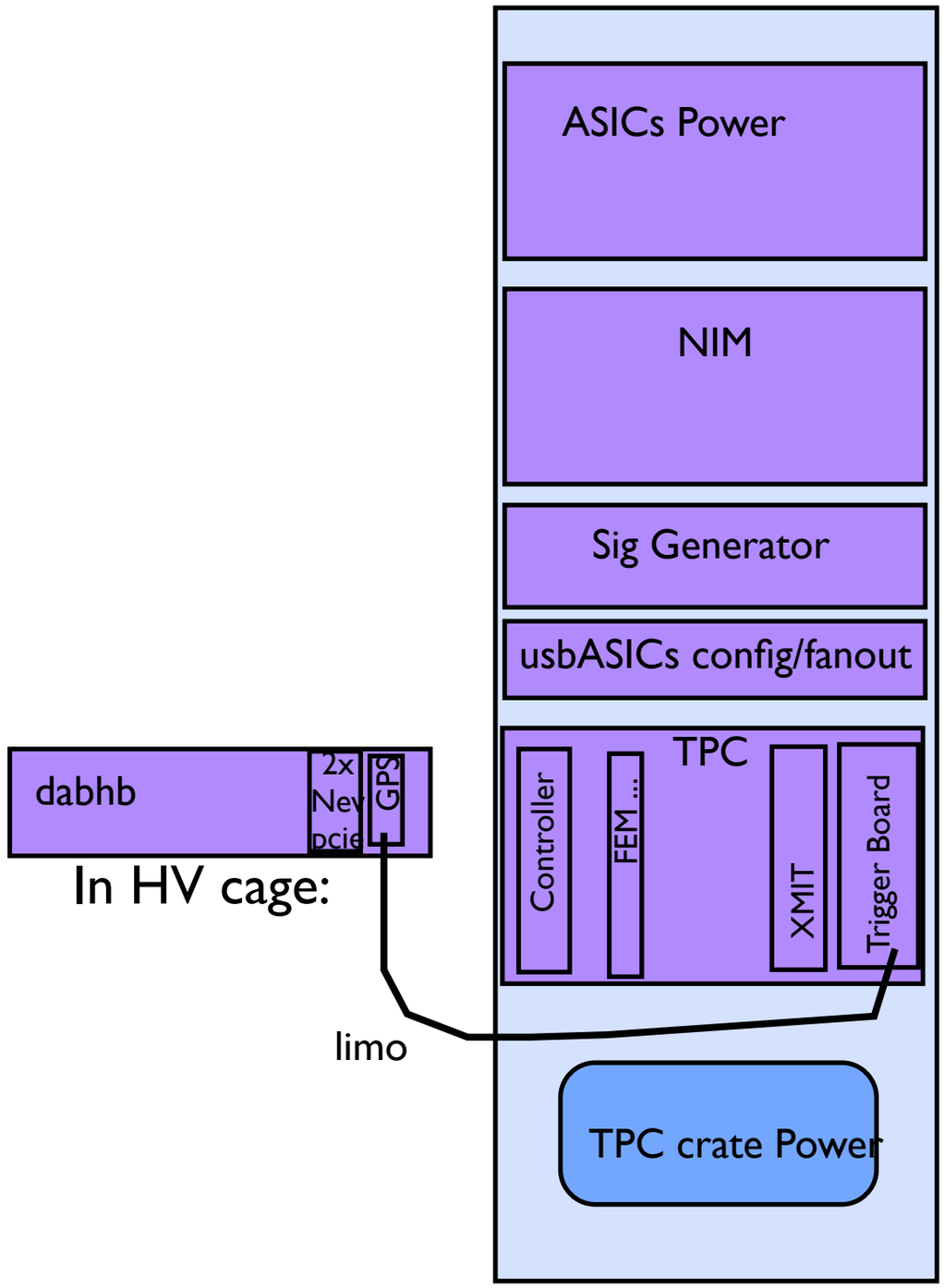
GPS Status

Eric, 14-Aug-2013

DAB MRT

Every second a pulse goes out over limo to Trigger Board, which forces a register to hold the current frame, sample, div. (1.6 msec, 0.5 musec, 16 nsec). On that same PPS an OS interrupt occurs on dabhb. The interrupt thread handles this interrupt and asks 2 questions:

(1) What is the frame,sample,div? - the answer requires issuing a query through the controller card, across the backplane and into the Trigger Board register.
(2) What is the latched GPS time at which the interrupt happened?



Interrupts begin on Connect

%MSG
ENTER STATE : ConfiguredMap::AwaitingConnection
%MSG-i WorkerThread: SebApplication 13-Aug-2013 16:05:30 CDT MF-online
SebApplication::connectionMonitorThreadFunction started running.
%MSG
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: 1376427932, micro seconds: 237, nano seconds: 0
GPSInterrupt: daq clock is 10803, 2612, 0 .... which is 17.2861 seconds.
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: 1376427933, micro seconds: 237, nano seconds: 0
GPSInterrupt: daq clock is 11428, 3237, 0 .... which is 18.2864 seconds.
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: 1376427934, micro seconds: 237, nano seconds: 0
GPSInterrupt: daq clock is 12053, 3862, 0 .... which is 19.2867 seconds.
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: **1376427935**, micro seconds: **137**, nano seconds: 0
GPSInterrupt: daq clock is 12678, 391, 0 .... which is **20.285** seconds.
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: **1376427936**, micro seconds: **137**, nano seconds: 0
GPSInterrupt: daq clock is 13303, 1016, 0 .... which is **21.2853** seconds.
%MSG-d SebApplication: SebApplication 13-Aug-2013 16:05:35 CDT MF-online
Emit event ConnectDone
%MSG
ENTER STATE : ConfiguredMap::Connected
PUSH TO STATE : ConnectedMap::AwaitingRun
%MSG-d SebApplication: SebApplication 13-Aug-2013 16:05:35 CDT MF-online
Called AwaitingConnectionEnd()
%MSG
%MSG-d SebApplication: SebApplication 13-Aug-2013 16:05:35 CDT MF-online
Called ConnectedStart()
%MSG
%MSG-d SebApplication: SebApplication 13-Aug-2013 16:05:35 CDT MF-online
Called AwaitingRunStart()
%MSG
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: 1376427937, micro seconds: 137, nano seconds: 0
GPSInterrupt: daq clock is 13928, 1641, 0 .... which is 22.2856 seconds.
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: 1376427938, micro seconds: 137, nano seconds: 0
GPSInterrupt: daq clock is 14553, 2266, 0 .... which is 23.2859 seconds.
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: 1376427939, micro seconds: 137, nano seconds: 0
GPSInterrupt: daq clock is 15178, 2891, 0 .... which is 24.2862 seconds.
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: 1376427940, micro seconds: 237, nano seconds: 0

It works!

BeginRunRequest received for partition number 0.
%MSG
ENTER STATE : ConnectedMap::Running
%MSG-d SebApplication: SebApplication 13-Aug-2013 16:05:50 CDT MF-online
Called AwaitingRunEnd()
%MSGPUSH TO STATE :
RunningMap::ProcessingFragments
%MSG-d SebApplication: SebApplication 13-Aug-2013 16:05:50 CDT MF-online
Called RunningStart()
%MSG
%MSG-d SebApplication: SebApplication 13-Aug-2013 16:05:50 CDT MF-online
Called ProcessingFragmentsStart()
%MSG
%MSG-i WorkerThread: SebApplication 13-Aug-2013 16:05:50 CDT MF-online
rawFragmentProducer::produceFragments started running.
%MSG
%MSG-i WorkerThread: SebApplication 13-Aug-2013 16:05:50 CDT MF-online
rawFragmentProducer::fillCircularBuffer started running.
%MSG
%MSG-i WorkerThread: SebApplication 13-Aug-2013 16:05:50 CDT MF-online
rawFragmentConsumer::consumeFragments started running.
%MSG
%MSG-d NUXMITStreamReader: SebApplication 13-Aug-2013 16:05:50 CDT MF-online
buffer allocation lower address =37900000
%MSG
%MSG-d NUXMITStreamReader: SebApplication 13-Aug-2013 16:05:50 CDT MF-online
buffer allocation higher address =00000000
%MSG
%MSG-d NUXMITStreamReader: SebApplication 13-Aug-2013 16:05:50 CDT MF-online
buffer allocation lower address =37940000
%MSG
%MSG-d NUXMITStreamReader: SebApplication 13-Aug-2013 16:05:50 CDT MF-online
buffer allocation higher address =00000000
%MSG
%MSG-d NUXMITStreamReader: SebApplication 13-Aug-2013 16:05:50 CDT MF-online
called dmaInitializeOnFirstLoop
%MSG
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: 1376427952, micro seconds: 237, nano seconds: 0
GPSInterrupt: daq clock is 23303, 2824, 0 .... which is 37.2862 seconds.
%MSG-d NUXMITStreamReader: SebApplication 13-Aug-2013 16:05:51 CDT MF-online
*** *** DMA timed out. DMAed 143772 bytes.*** ***
%MSG
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: 1376427953, micro seconds: 137, nano seconds: 0
GPSInterrupt: daq clock is 23928, 3449, 0 .... which is 38.2865 seconds.
%MSG-d NUXMITStreamReader: SebApplication 13-Aug-2013 16:05:52 CDT MF-online
We have called the maximum number of triggers: 100 (100 called).
%MSG
GPSInterrupt::gpsIntHandler (gpsTime), seconds since Jan 1, 1970: 1376427954, micro seconds: 137, nano seconds: 0
GPSInterrupt: daq clock is 24553, 4074, 0 .... which is 39.2868 seconds.

BeginRun

continues reporting every second

std::map<tbclkub_t,gps_time_t>

- in rawFragmentDMASource I create a map
- This is maintained until it is 1 day=3600*24 rows long. (Perhaps should shorten that.)
- Then I drop off oldest entries one at a time
- Only need the latest one to shove into each seb-10 crate_header

structs

```
// This struct will be a key in a map, so I must define "<".
typedef struct tbclkub
{
  uint32_t frame;
  uint16_t sample;
  uint16_t div;

    bool operator<(const tbclkub& mk) const
    {
        if (frame < mk.frame)
        {
           return true; // meaning: keep sorting.
         }
        return false;
    }

  tbclkub (uint32_t f=0, uint16_t s=0, uint16_t d=0): frame(f),sample(s),div(d) {  }

} tbclkub_t;
```

```
typedef struct gps_time
{
  // 2^32 = 4.E9 .Thus 32 bits allows for both (2013-1970)*3.14e7 seconds and
  // enough nanoseconds to span a second.
  uint32_t second; // seconds since the epoch.
  uint32_t micro;  // microseconds since the second.
  uint32_t nano;  // nanoseconds since the second.
  gps_time(){};
} gps_time_t;
```

rawFragmentDMASource.cpp

```
//Frame is most recent 2 MHz Frame number.
std::map<tbclkub_t,gps_time_t>::Iterator it = _gpsi._mapClockGPS.find(Frame);

if (it ==_gpsi._mapClockGPS.end()) --it;
tbclkub_t daqClock(it->first);
gps_time_t gpsTime(it->second);
    //Stuff this into seb-10 crate_Header
crate_Header.gps_time = gpsTime;
crate_Header.daqClock_time = daqClock;
```

At the root of the problem was an interesting bug

- I was destructing a copy of my GPSInterrupt class inadvertently.
- *Was making a copy* as I handed off handle to the class, rather than properly passing a reference to it.
- Destructor killed the interrupt thread handler. And so code froze on subsequent 1PPS interrupts coming from GPS card.
- Fix: don't call the destructor. Bad idea. Better: insert lines to make compiler disallow all (inadvertent) copying of the class. (Thx, Gennadiy.)

Outstanding issues

- There is ~1msec jitter on TriggerBoard time. Does it matter?
- There is infrequent 100 musec jitter on GPS time. Why exactly 100 musec? Does it matter?
- Why is the 100 nanosecond counter always 0?
- Convince ourselves that we've got corresponding interrupt/TB pulse. I could be too quick in requesting TB time. Insert a 1msec delay. Ensure that interrupt handler doesn't get behind.
- Get seb-10's CrateHeader gps word into EventHeader in assembler. Then check the EventHeader in assembled events.

LArTF

- We have a new production Symmetricom GPS card. Will put it in uboonedaq-prod-seb-10
- We need to get an antenna out at LArTF, as once was discussed. See no evidence of that happening yet.
-