Harvard Cosmic-Ray Telescope with MMFE8: Plans and Progress

M. Bledsoe, N. Felt, M. Franklin, B. Garber, P. Giromini, J. Grotto, J. Huth, **C. Rogan**, A. Tuna, A. Wang

HARVARD
UNIVERSITY

ATLAS NSW Electronics - March 21, 2016

Outline

- **Goal:** Cosmic-ray telescope with MMFE-8 front-end boards for testing MicroMegas readout and trigger
  - DAQ requirements and strategy
  - Timing and amplitude calibration
- Observed issues from calibration stress-tests
  - Diagnoses and treatment
- Outlook

Harvard Cosmic Ray Telescope

Harvard Cosmic Ray Telescope (HCRT)

ATL-COM-UPGRADE-2014-038

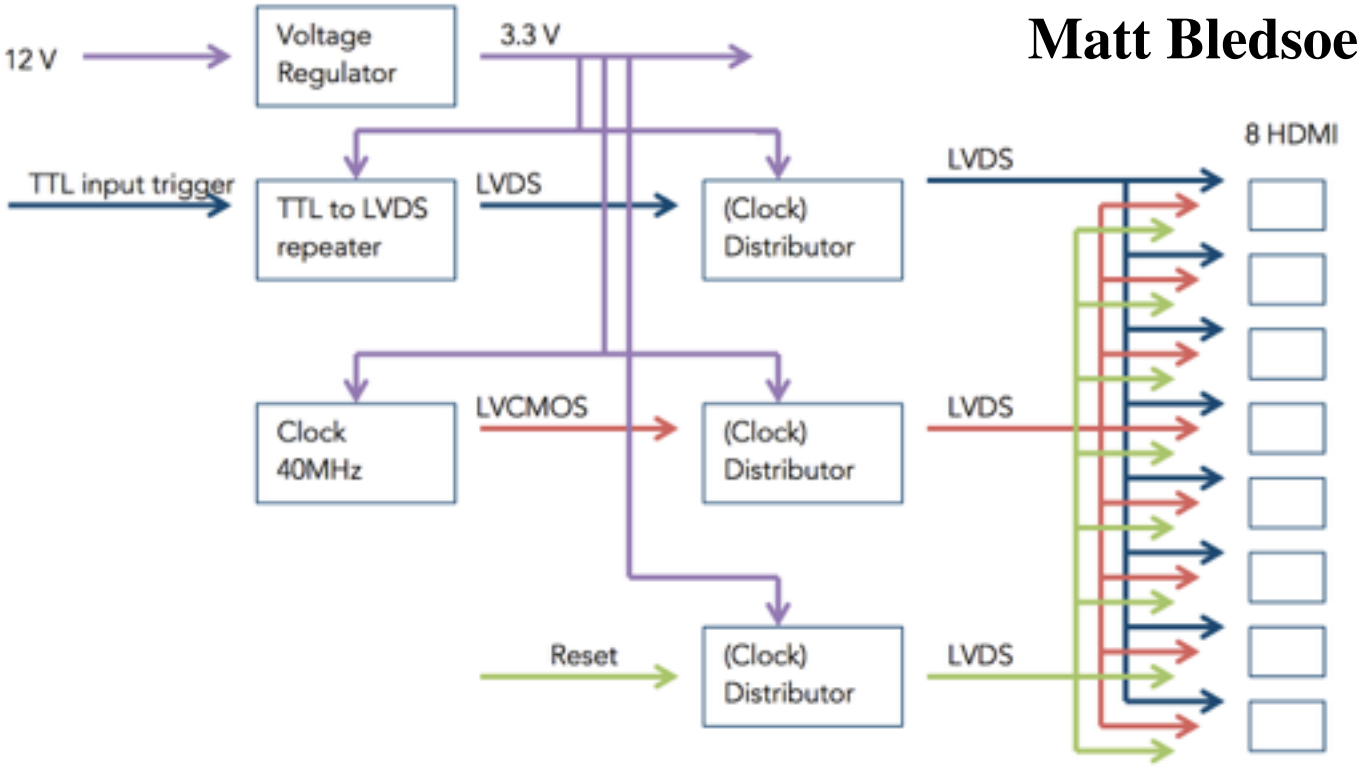
- Have previously collected cosmic ray data using Mini-1 board with VMM1
- Working to reproduce set-up with octuplet equipped with MMFE-8 and VMM2

DAQ Requirements

- On-going work on multiple needs of working DAQ system:
  - Testing/debugging MMFE-8 firmware provided by Arizona
  - Multiple MMFE-8 simultaneous readout (see **Alex Tuna**'s talk)
  - Scintillator/PMT trigger interface to FE boards and DAQ
  - VMM synchronization over multiple boards

Trigger distribution

Harvard undergrads
Matt Bledsoe and **Julia Grotto**



- Trigger distribution system under development
- To be used for VMM synchronization (see next slide), synchronized test-pulsing (ART testing), and triggered readout

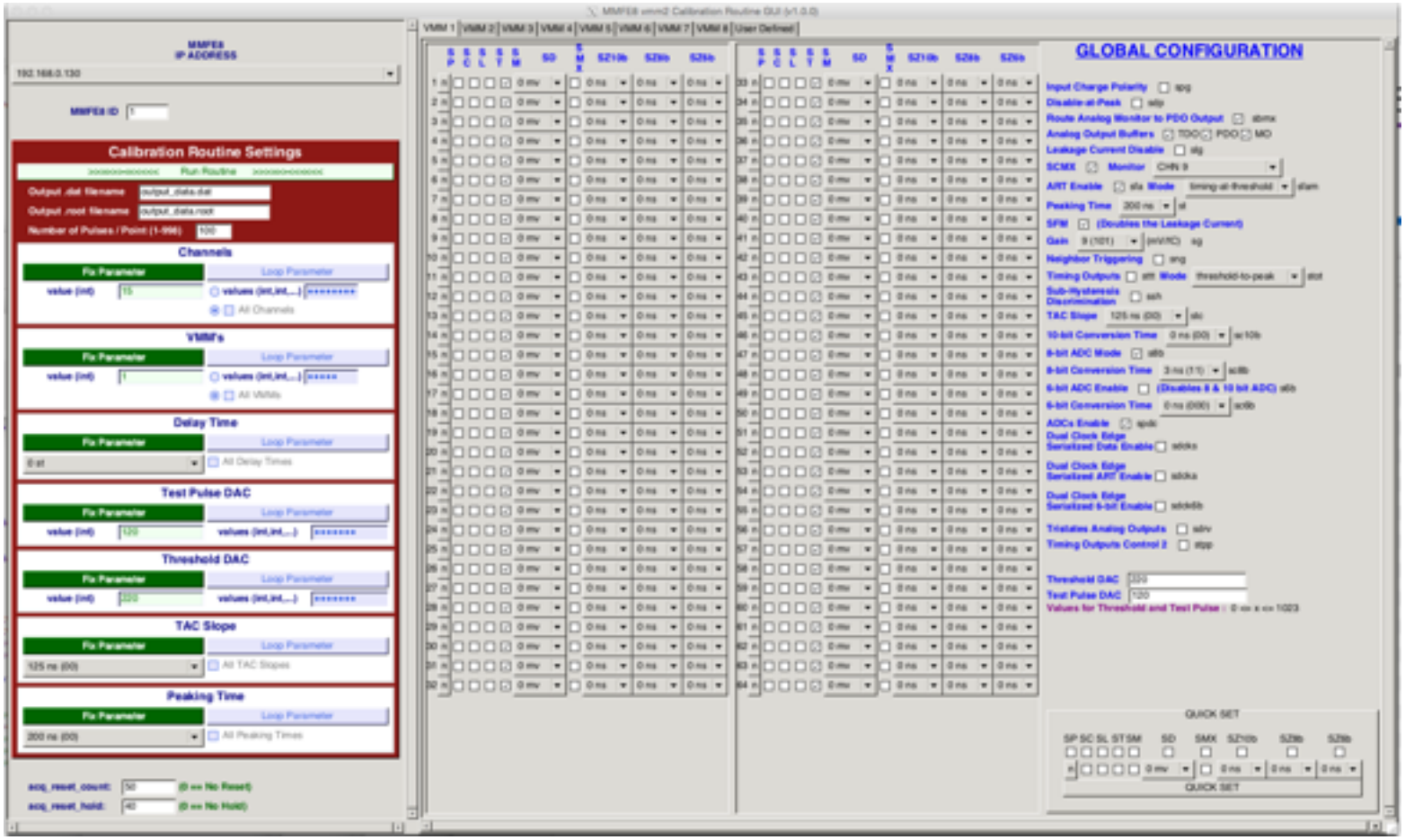
VMM synchronization

- Plan for data-taking with 8 MMFE-8 boards x 8 VMM/board includes synchronization using an external trigger and common soft resets. See **Ann Wang**'s talk discussing these plans: (2/12/2016) https://indico.cern.ch/event/496030/
- Firmware code for necessary components of scheme completed
  - stopping/synchronizing CKBC
  - registers to signal arrival of external trigger to DAQ gui interface
  - soft resets (currently required to preserve token logic) and FIFO resets
- Waiting for trigger-setup to fully test code and system

Timing and amplitude calibration

- Developing calibration system for amplitude (PDO) and timing (TDO) measurements
- Calibration of multiple MMFE8 boards with multiple VMMs requires ability to efficiently scan over all boards/ VMMs/channels with different configuration variations
  - test pulse DAC variations for PDO calibration
  - test pulse timing variations (w.r.t. CKBC) for TDO calibration
  - studies of effects of other variations (DAC threshold, peaking time, TAC slope, etc.) on amplitude and timing measurements

Calibration routine gui

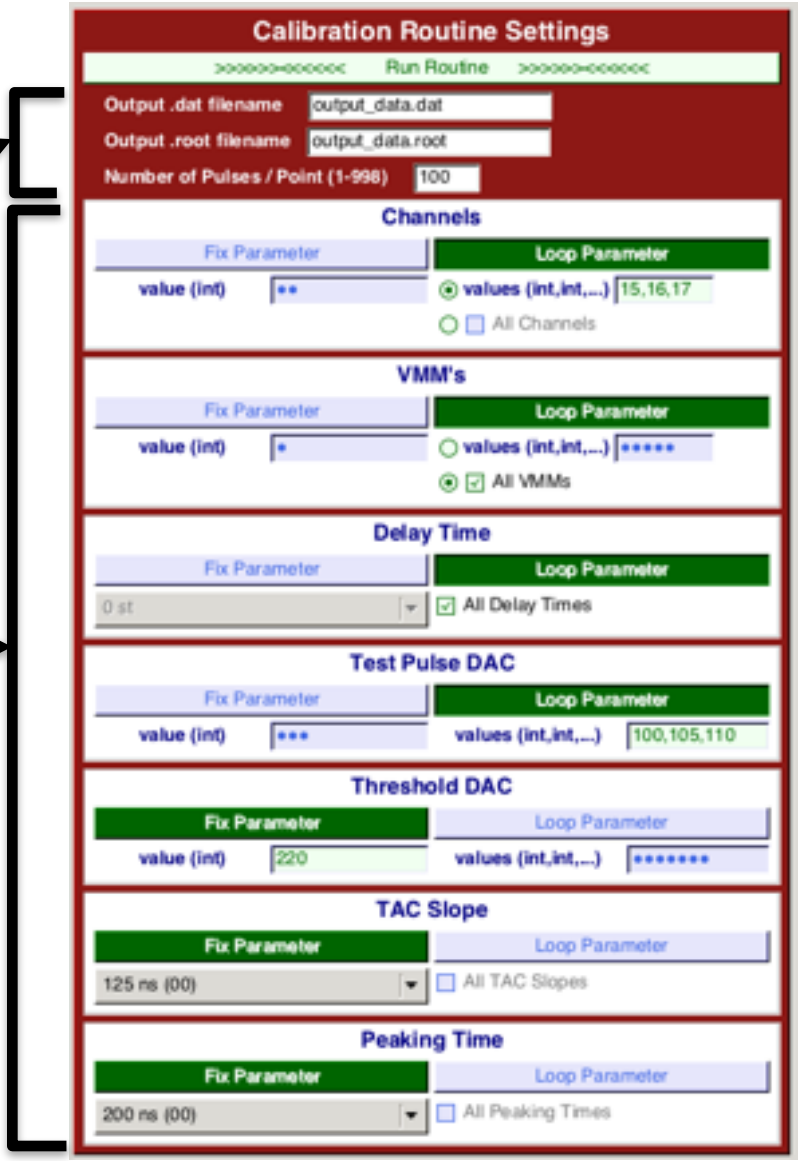


- Python-based gui adapted from DAQ gui from Arizona with built-in, automated, and configurable calibration routine implementation

Calibration routine gui

Automated calibration routine parameters configurable using interactive gui panel

- Configurable data output in custom .dat (txt) and .root format along with requested number of test pulses per configuration
- Configurable scan parameters. Each parameter can be **fixed** or **looped** over multiple parameters. Nested for-loops scan over all combinations of looped parameters:
  - channels
  - VMMs
  - delay time (TDO calibration)
  - test pulse DAC
  - threshold DAC
  - TAC slope
  - peaking time

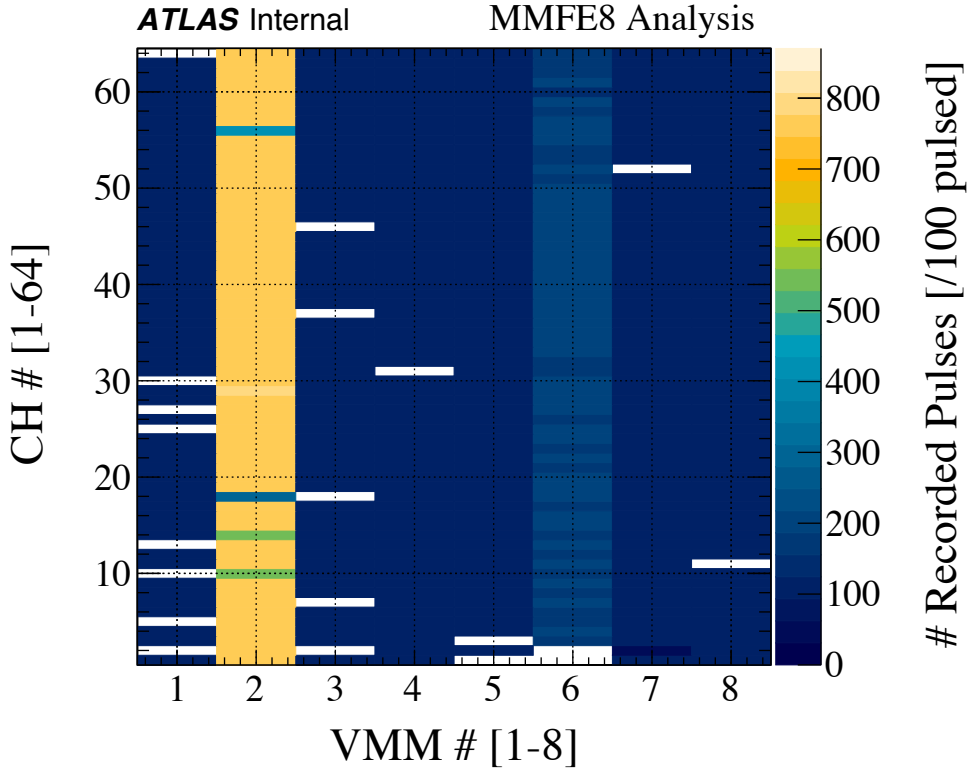


Calibration routine test

First tests of calibration routine on single MMFE8 immediately revealed previously unknown issues…

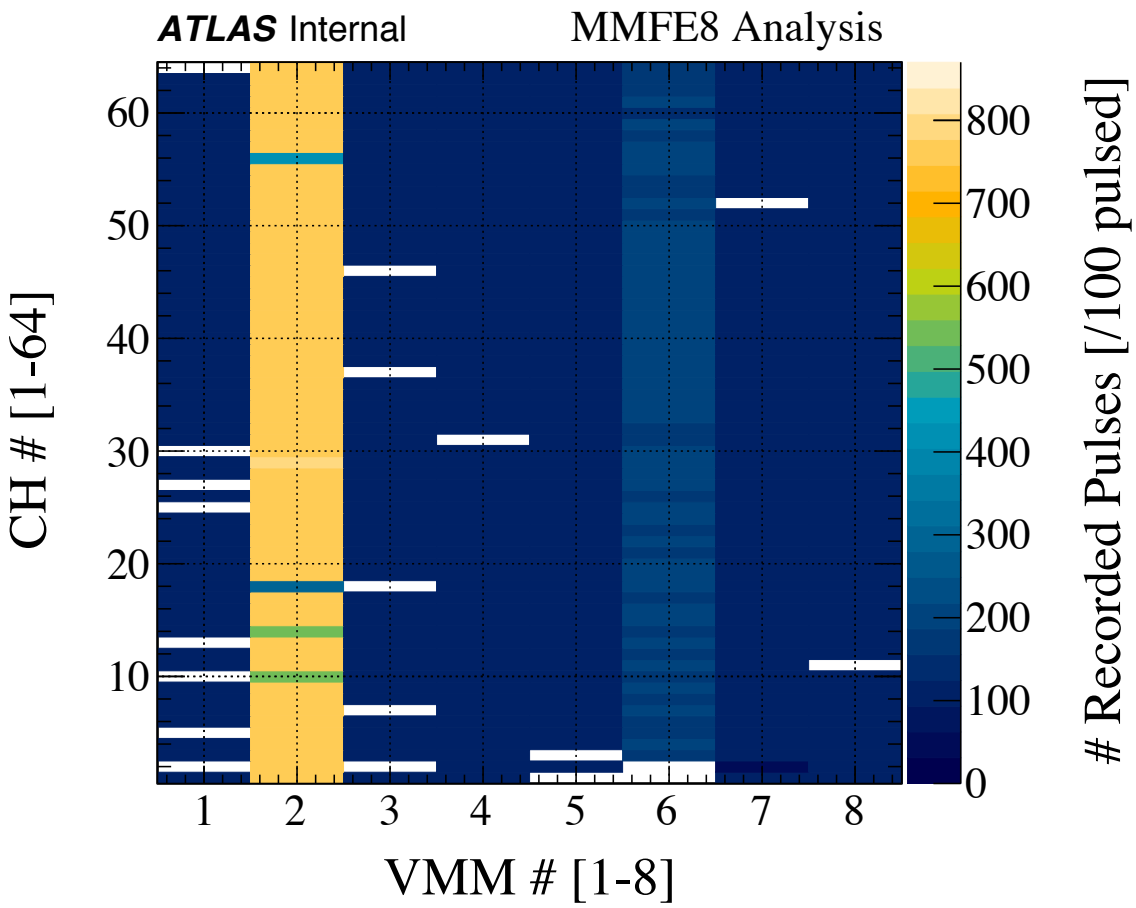
Through-out this talk, default calibration routine configuration used unless otherwise noted:

- All channels/all VMMs scanned
- 100 test pulses per channel/configuration
- test pulse DAC = 120
- threshold DAC = 220
- TAC slope = 125 ns
- peaking time = 200 ns
- gain = 9 mV/fC
- DAQ termination for each channel/configuration once 200 data instances with pulsed channel = recorded channel are registered



should never encounter forced termination…

Observed issues



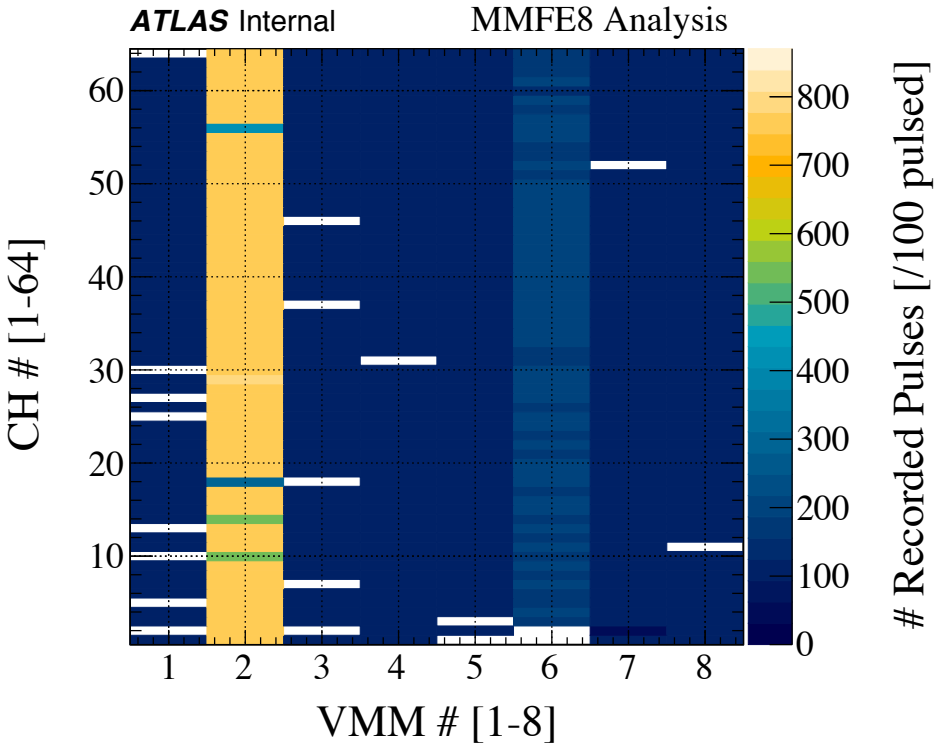
Pulsed channels with more than 100 recorded events indicate presence of additional erroneous events

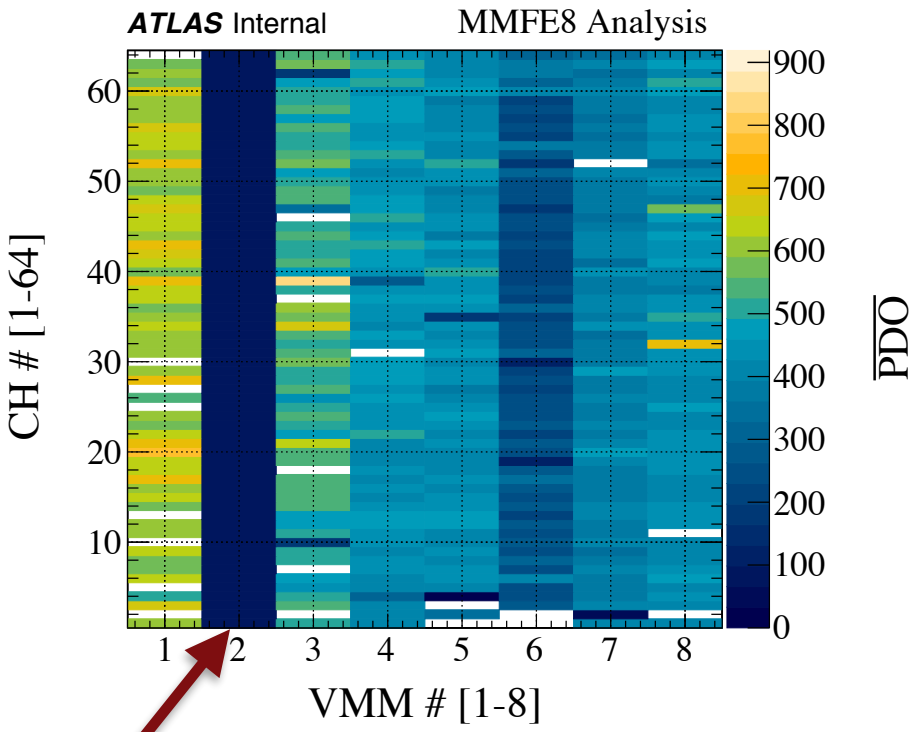
Channels with more than 200 recorded events indicate mis-assignment of channel addresses in recorded data

Issue #1

- **Issue:** Re-discovered "Class II"-like events as described by Wenxiang Ding (http://indico.cern.ch/event/465405/)
- **Symptoms:** simultaneously pulsing a "bad" channel (always ch1-4) and "healthy" channel results in channel address mis-assignment with colorful modular arithmetic:
  - Ex. VMM #5, channel 4:
  - Pulsing ch4 + ch5 ⇒ data in ch1 (even if masked!)
  - Pulsing ch4 + ch6 ⇒ data in ch2
  - Pulsing ch4 + ch7 ⇒ data in ch3
  - Pulsing ch4 + ch8 ⇒ data in ch4
  - Pulsing ch4 + ch9 ⇒ data in ch1
  - Pulsing ch4 + ch10 ⇒ data in ch2
  - …
  - modular arithmetic with modulus set by "bad" channel number…
- **Treatment:** masking observed "bad" channels removes problem

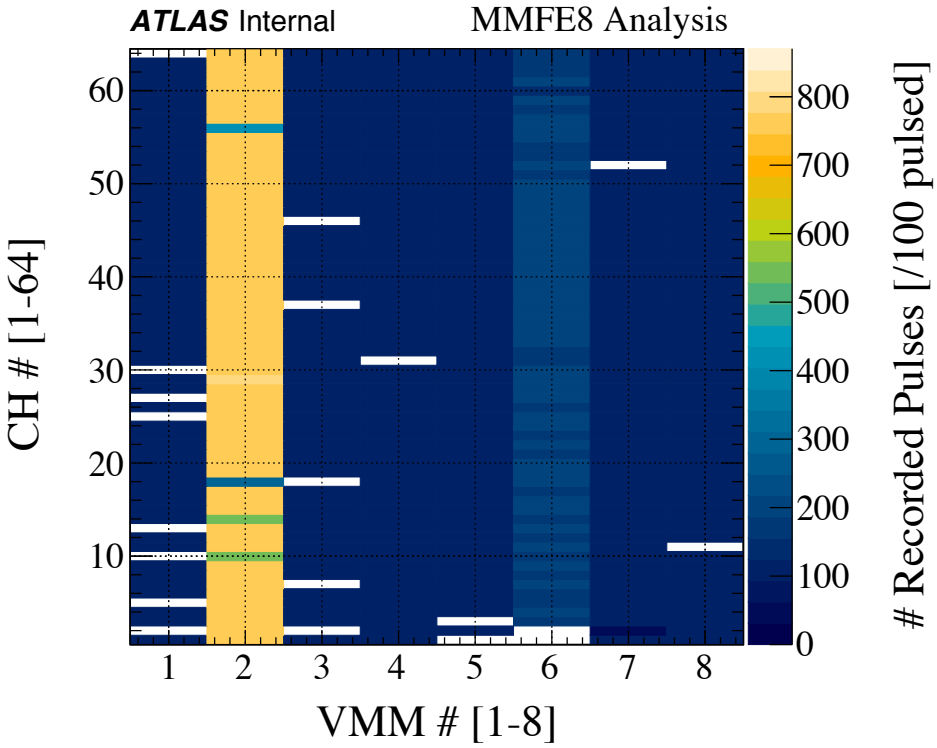
Observed issues

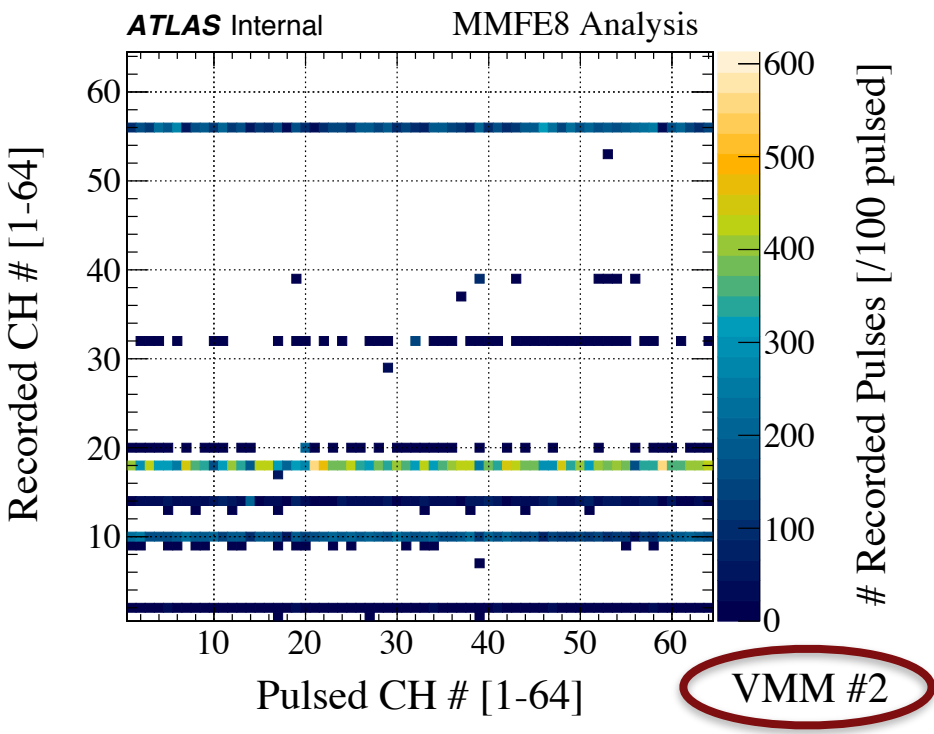




- **Issue #1 treatment does not fix other observed issues**
- **VMM #2 exhibits large number of additional, erroneous events**
- **PDO and TDO values of zero for all these VMM #2 events**

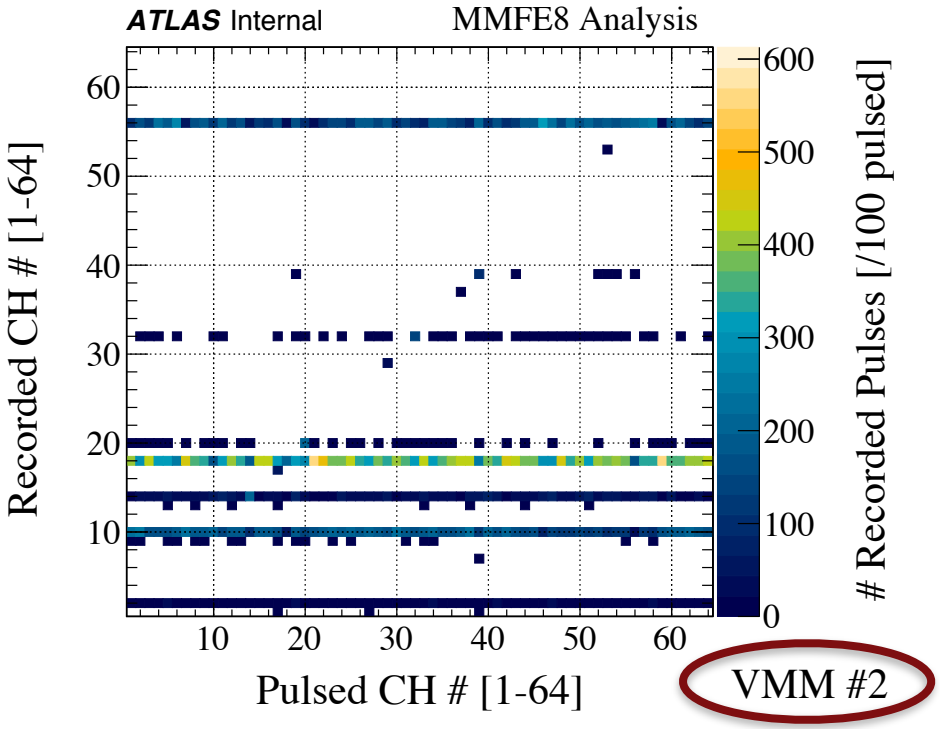
Observed issues

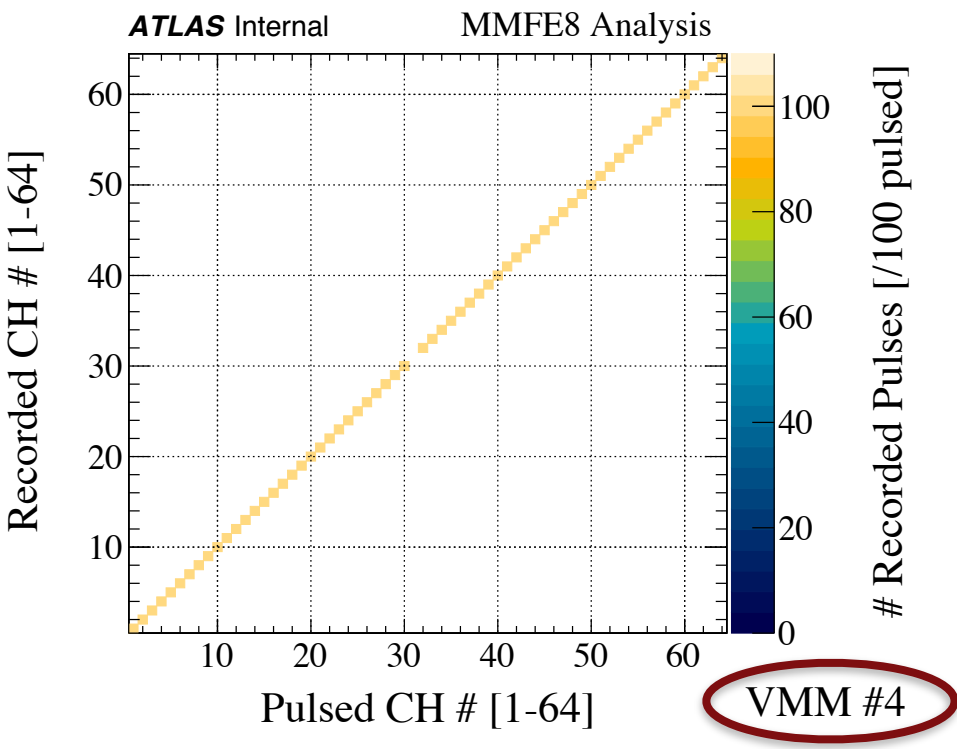




- **Events are assigned to a collection of re-occuring channels (rarely the correct one)**

Observed issues



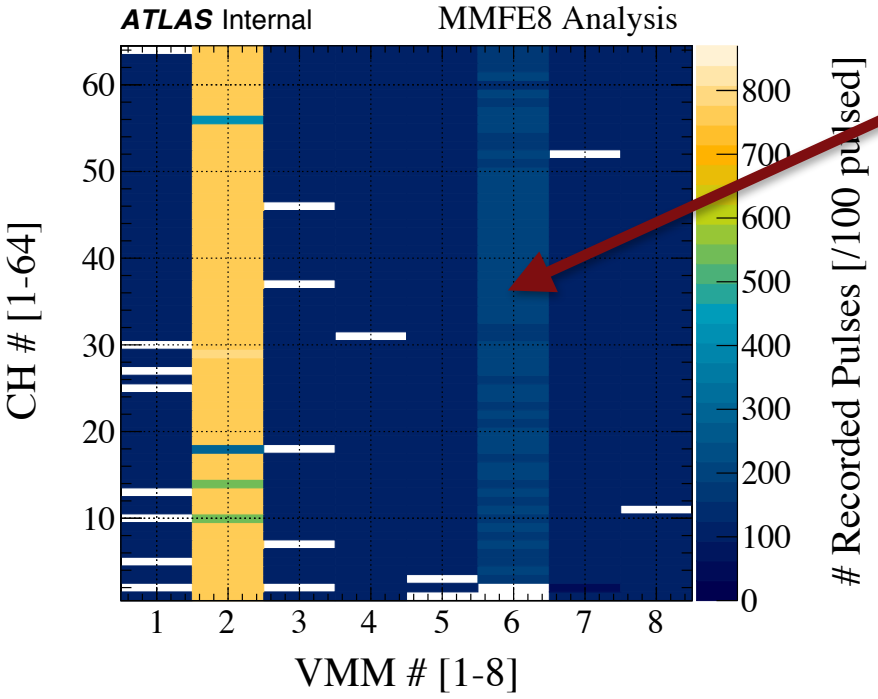


- **"Healthy" output should look like the above-right figure**

Issue #2

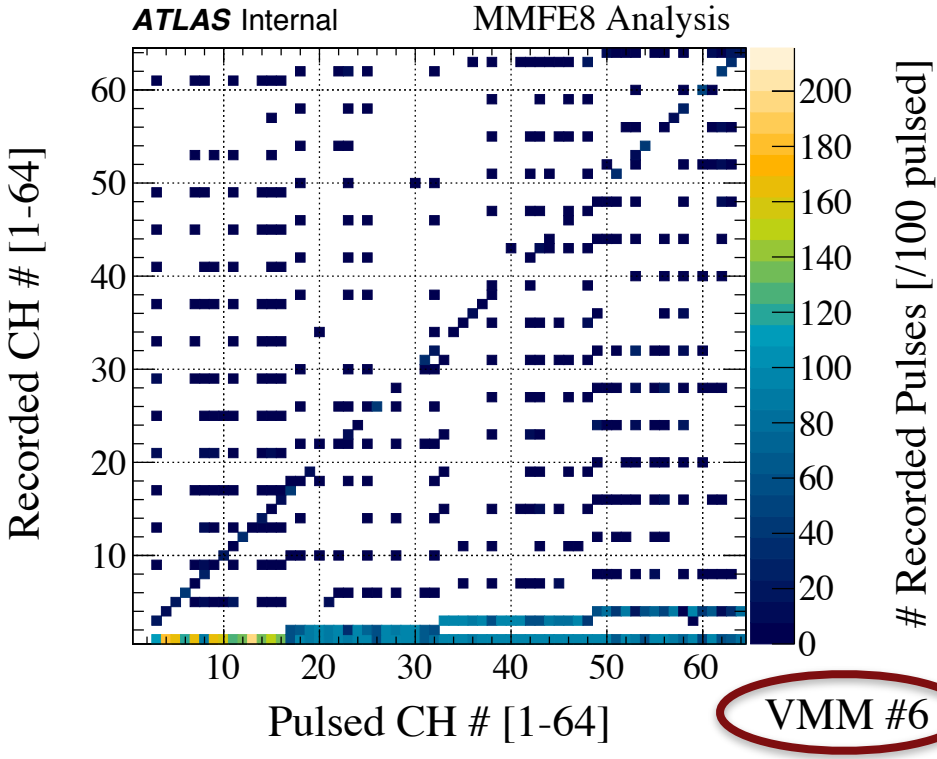
- **Issue:** On MMFE8 tested, VMM #2 records too many events, with zero PDO and TDO, with incorrectly-assigned channel addresses
- **Symptoms:** Dedicated tests reveal that global reset plus VMM loading (even with intermediate system reset) fills FPGA FIFO with erroneous events
- **Treatment:** Performing additional system reset after VMM load clears FIFO of these events ⇒ observed issue disappears

Observed issues



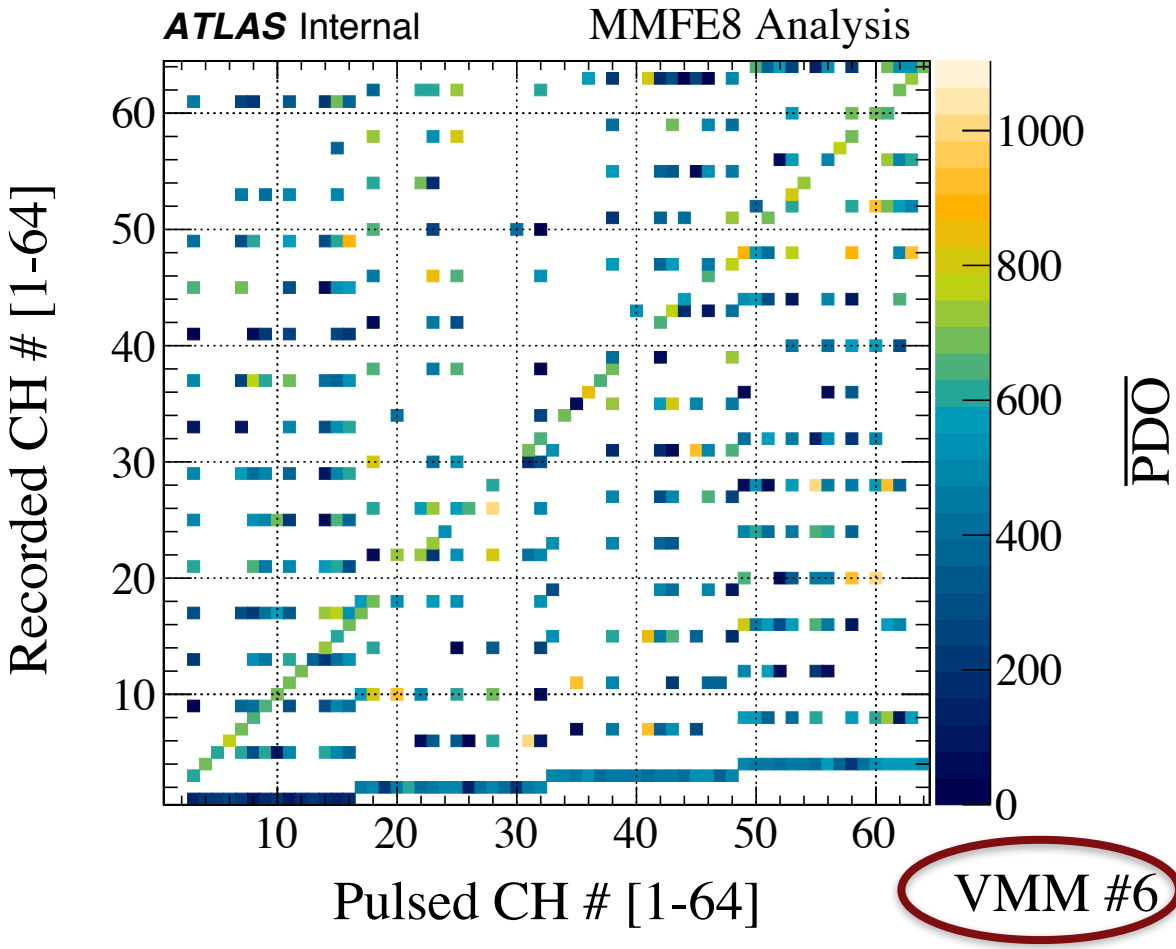
Excess events also appear on VMM #6

Recorded channel addresses reveals rich modular-arithmetic structure…

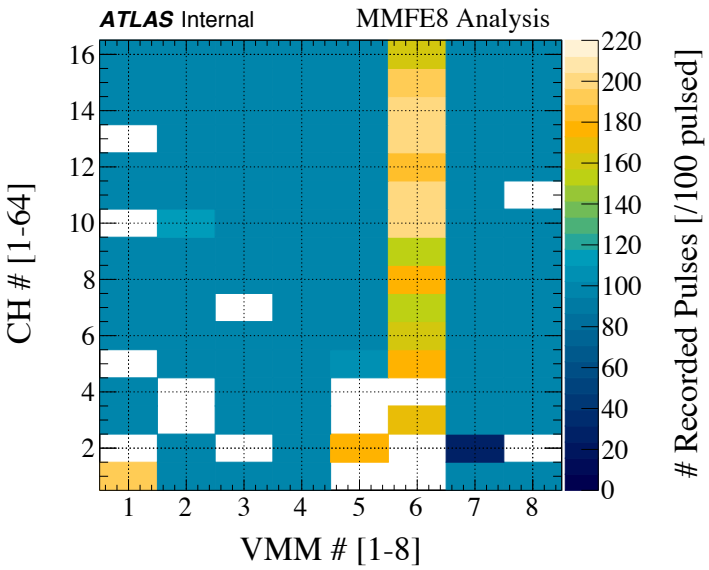


Observed issues

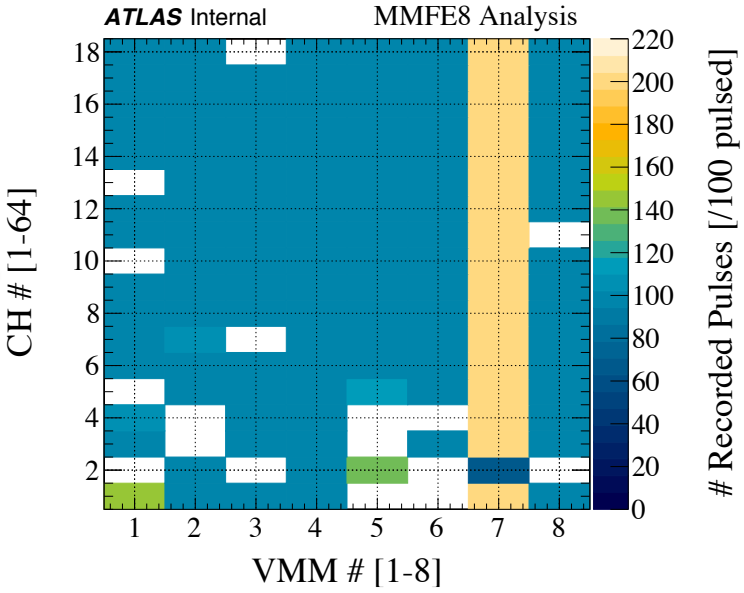
- Majority of events are assigned to channels 1-4, depending on ch / 16
- Other events assigned throughout rest of VMM channels, in steps of 4 channels offset by ch / 16
- PDO for correctly assigned addresses have ~healthy values, mis-assigned events appear pathological



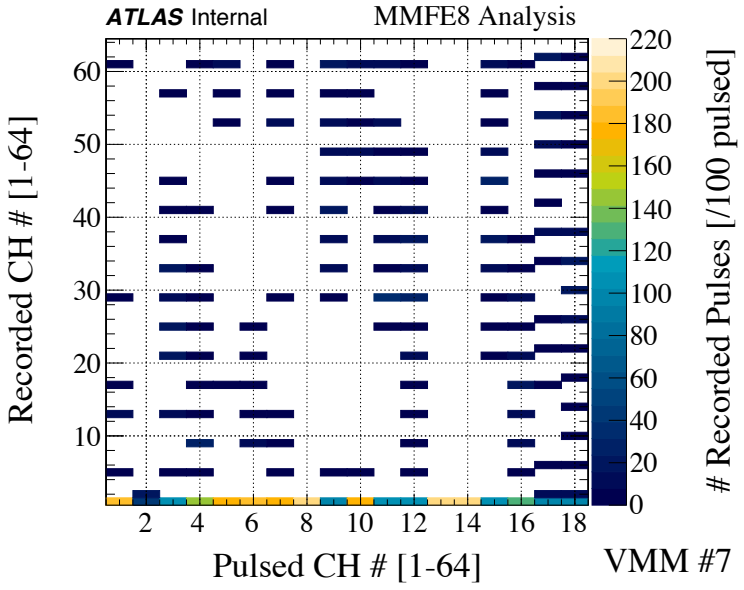
Observed issues



Firmware changes

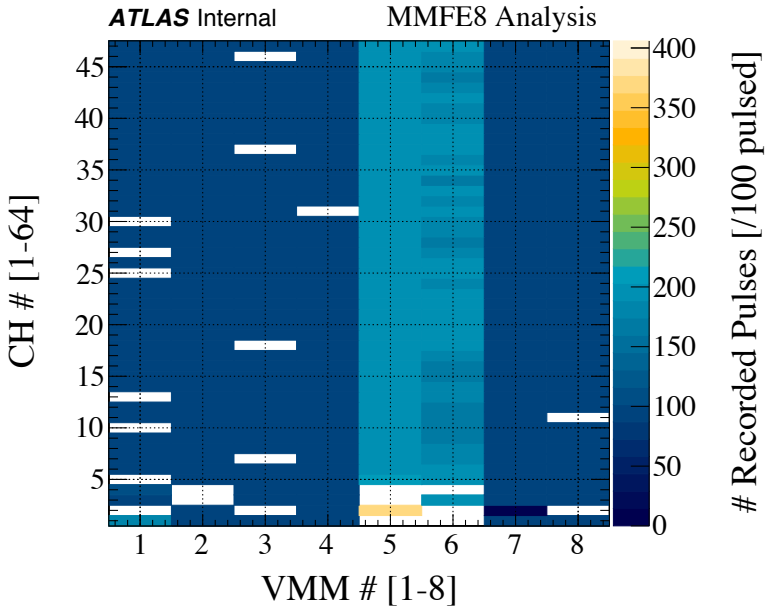


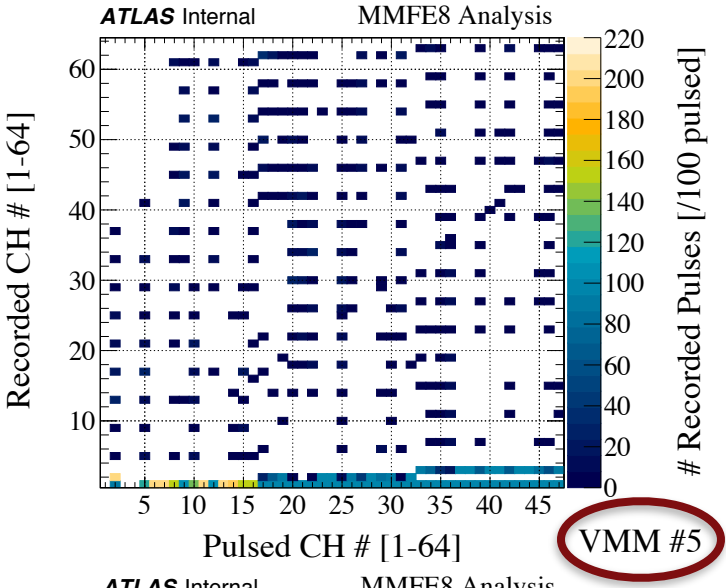
Seemingly inconsequential changes to firmware (removing redundant sensitivities, re-ordering order-insensitive code lines,…) can cause issue to move to different VMMs



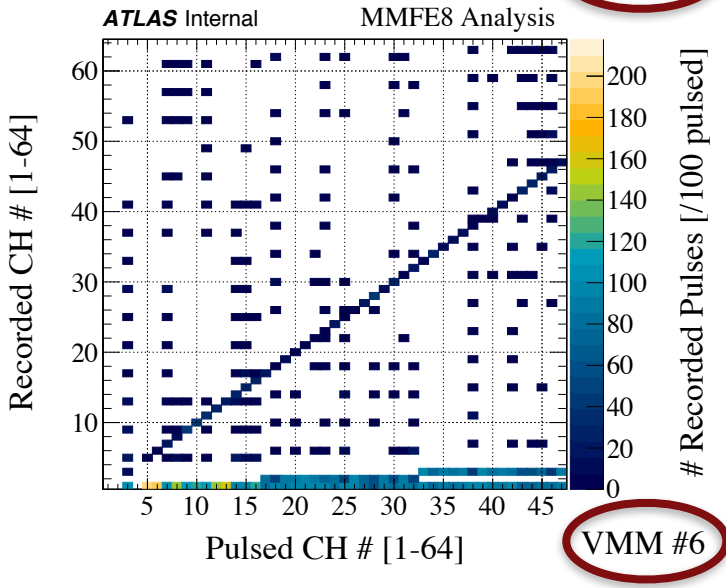
Exhibits same channel address pattern

Observed issues





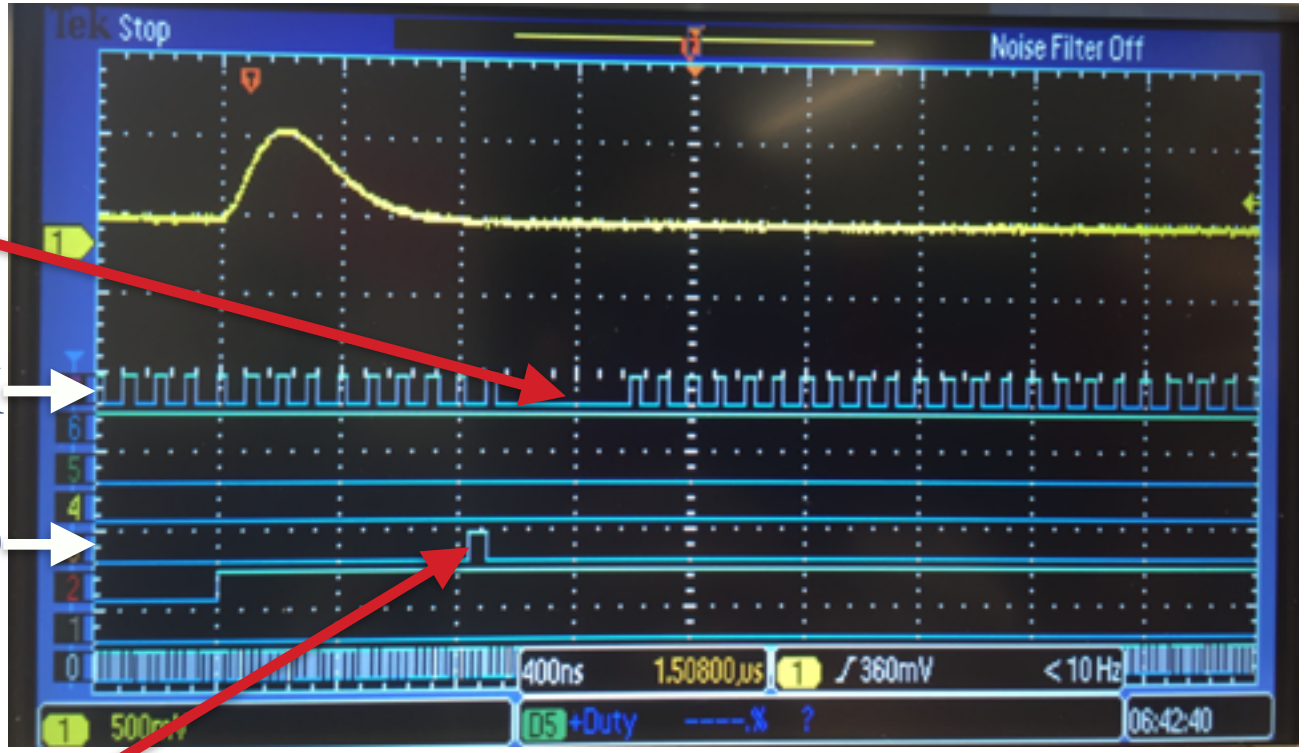
Changing the frequency of the test pulse clock can also cause changes to the VMM(s) exhibiting problems, still with the same pattern… (here, 1 kHz)



Also changed CKTK duty cycle to 35% (suggestion of Gianluigi) ⇒ important change that helped us understand phenomenology

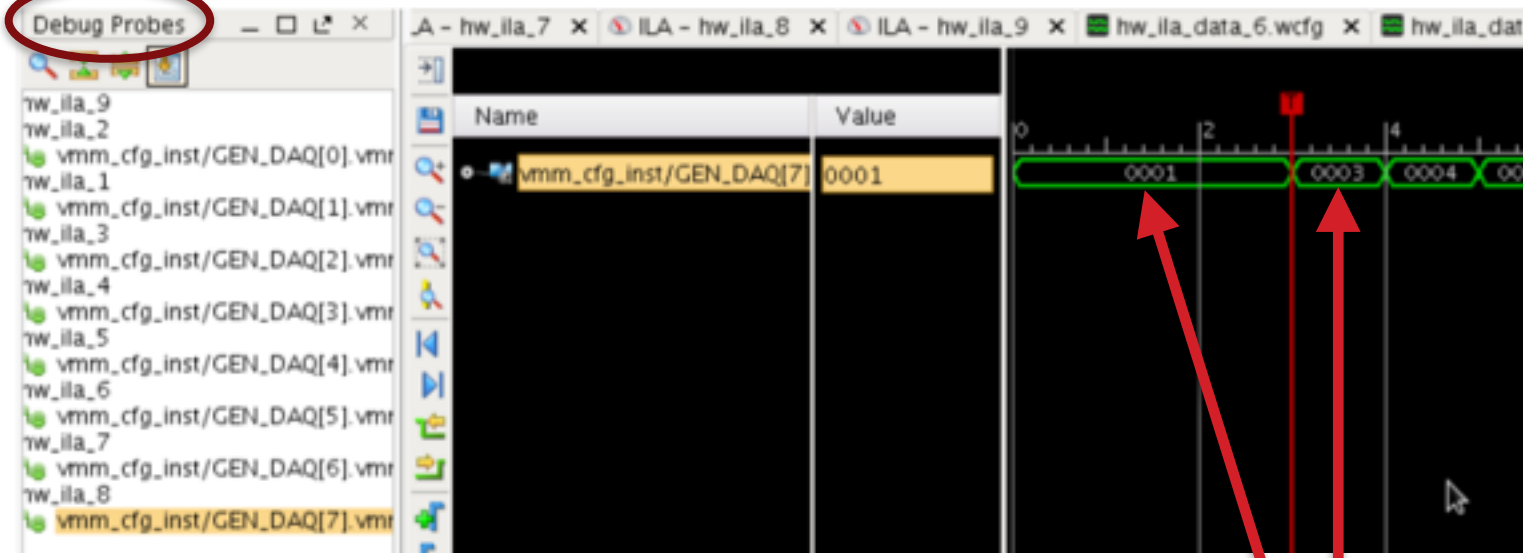
Observed issues

Problematic VMMs often have CKTK up even during data acquisition (not here)



Truncated data0 on scope corresponds to pathological recorded data

Observed issues



- "***dt_state***" signal in firmware corresponds to 9-state state-machine which manages data acquisition from VMMs into FPGA FIFO
- Observed to "jump" states in problematic VMMs (state 1 → state 3)
- four flip-flops control ***dt_state***, so '0001' → '0011' indicative of first one "sticking"

Issue #3

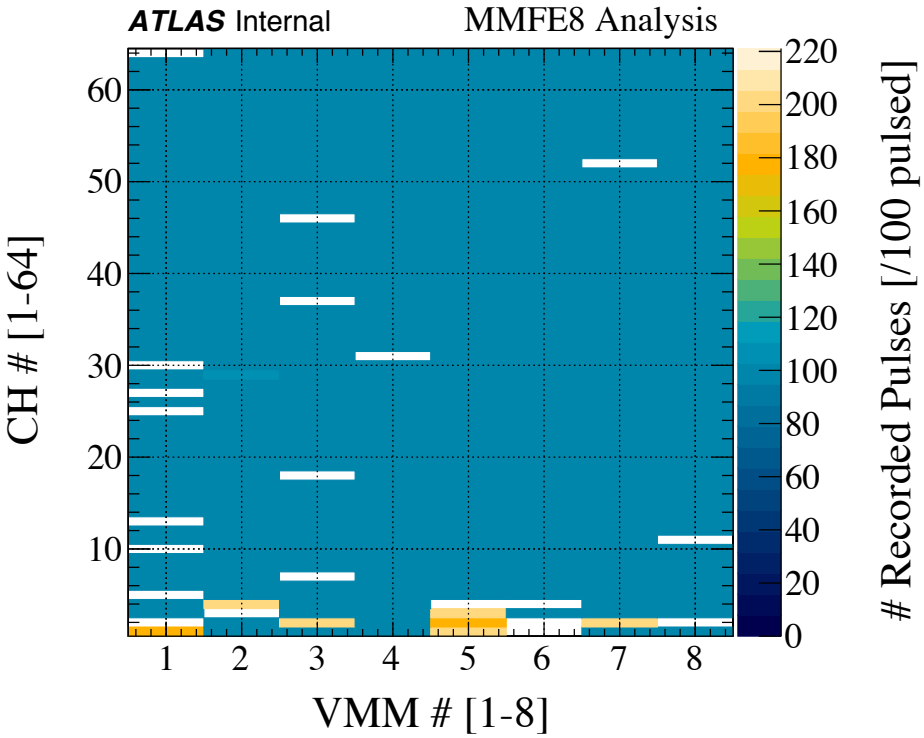
- **Issue/Symptoms:** Varying VMMs exhibiting incorrect channel assignment and pathological data for all channels, corresponding to truncated data0, incorrect state-machine progression, and often problems with CKTK not disabling during data acquisition by the on-board FPGA
- **Diagnosis/Treatment:** Data arriving from VMMs sometimes does not satisfy timing requirements of state-machine (different VMMs w/ different time skew), causing observed problems with problematic sensitivity to unstable timing
  - Fixed by feeding VMM data0 signals through two flip-flops clocked on 200 MHz clock
  - First flip-flop synchronizes data0 signal, second guards against meta-stable states (< 5 ns to steady-state)

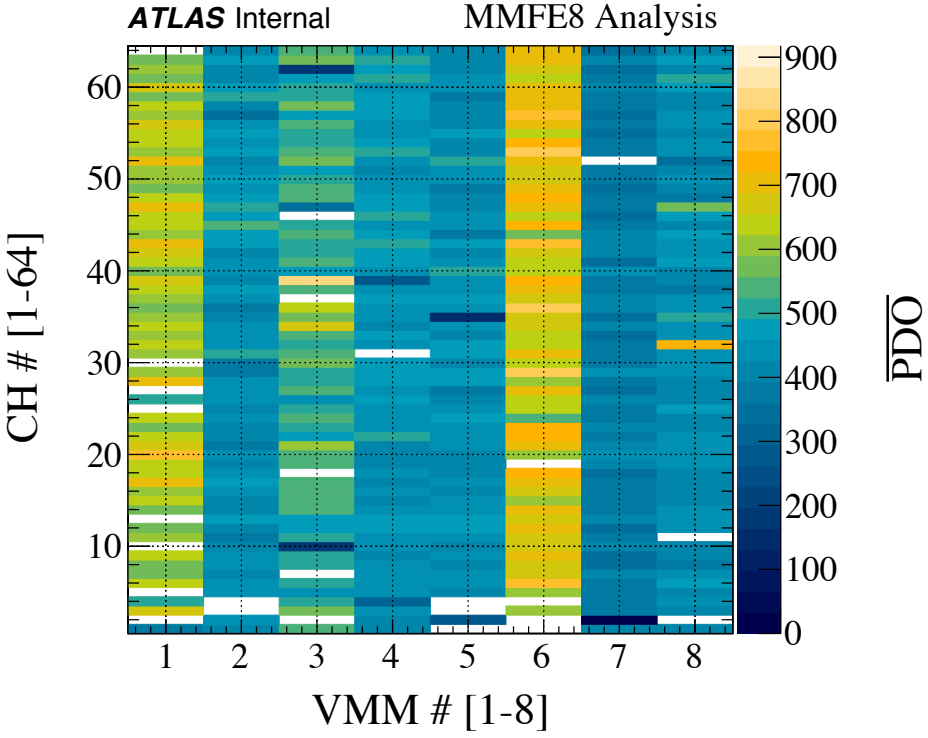
new to toplevel.vhd:

```
sync_data0: process (clk_200)
begin
  if rising_edge(clk_200) then
    for I in 0 to 7 loop
      vmm_data0_sync_vec(I) <= vmm_data0_async_vec(I);
      vmm_data0_vec(I) <= vmm_data0_sync_vec(I);
    end loop;
  end if;
end process sync_data0;
```

  - Should do this for all signals arriving from VMMs (currently done for data0 and data1)

Resolved issues





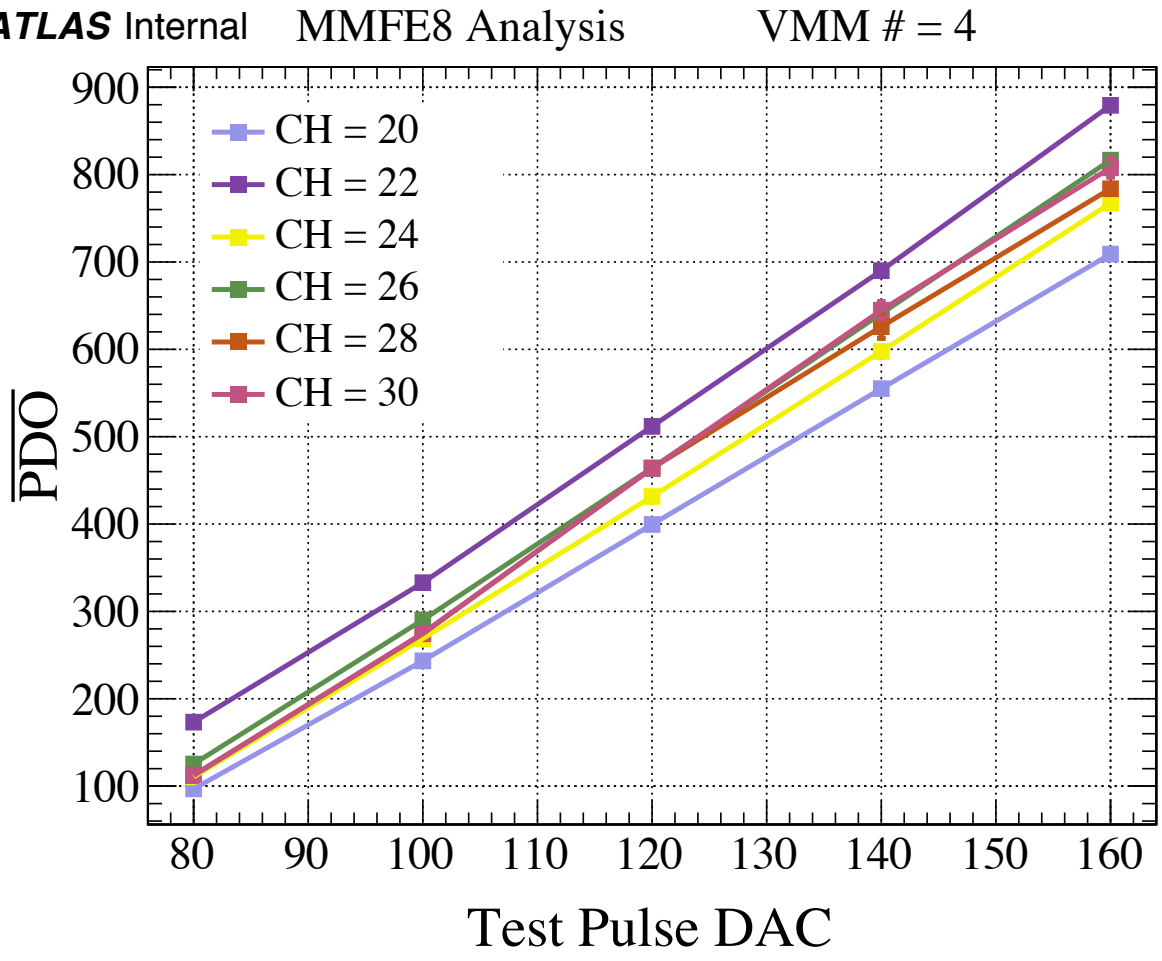
Fixes for issues #1-#3 removes all extra events (apart from some ‘Class II’ channels 1-4)

Average PDO’s appear ~sane…

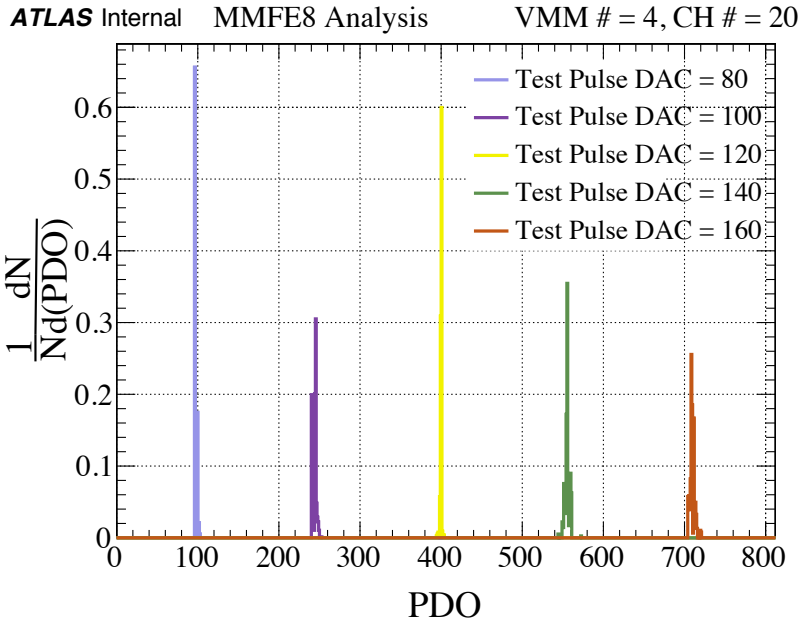
…can now proceed with calibration!

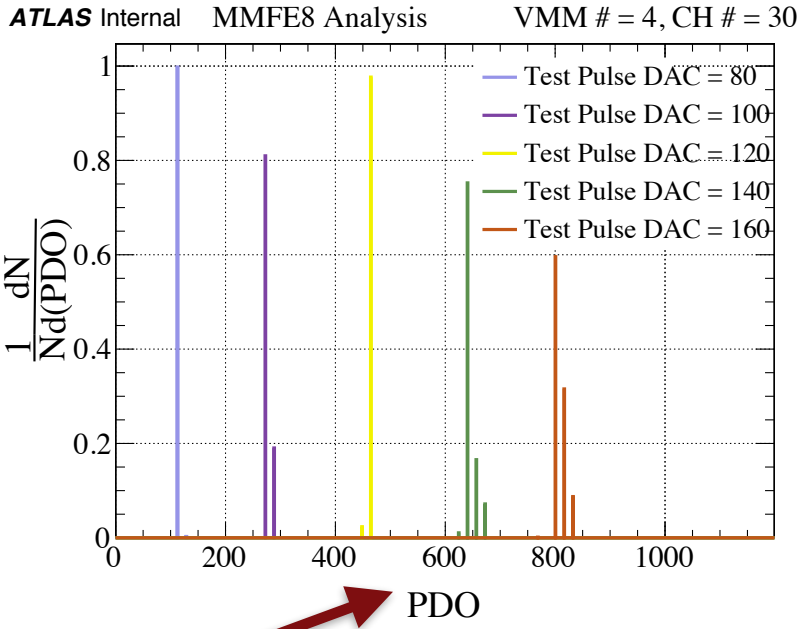
PDO calibration

Average PDO scales as expected with input test pulse DAC



PDO calibration



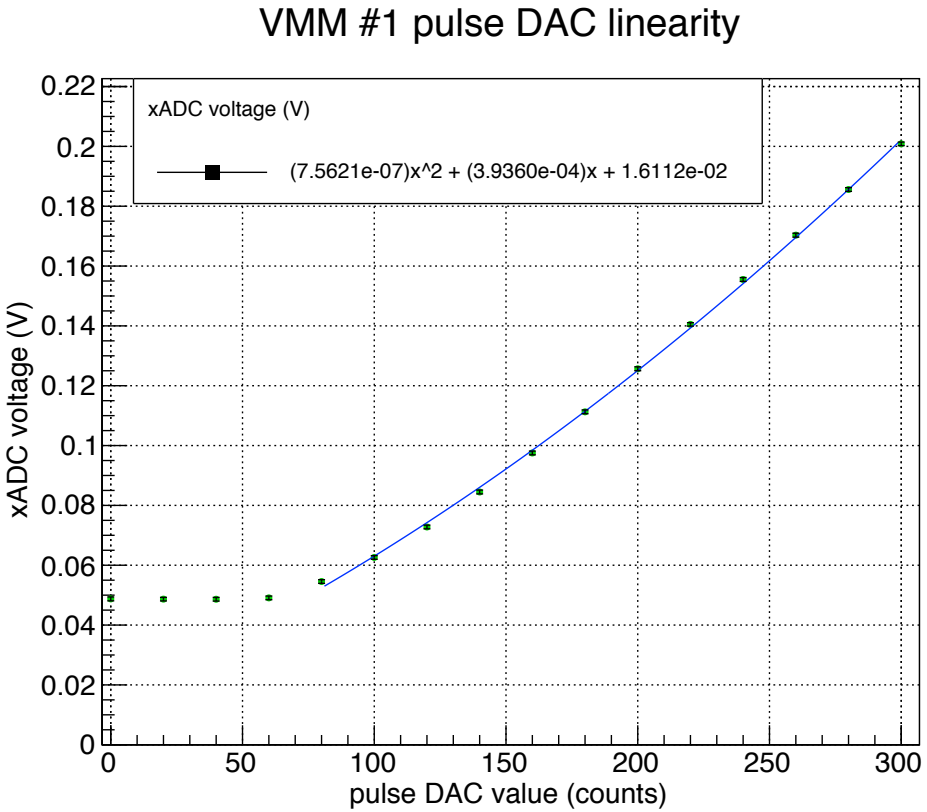


For some channels we observe loss of least significant ADC bits (all peaks 16 counts apart…)

xADC PDO calibration

- Even with same input test pulse DAC, DAC conversion is different for each VMM
- We are using the FPGA on-board ADC built by Arizona to more accurately calibrate channel PDOs in automated procedure
- Tested and appears to be working correctly

Harvard undergrad **Ben Garber**

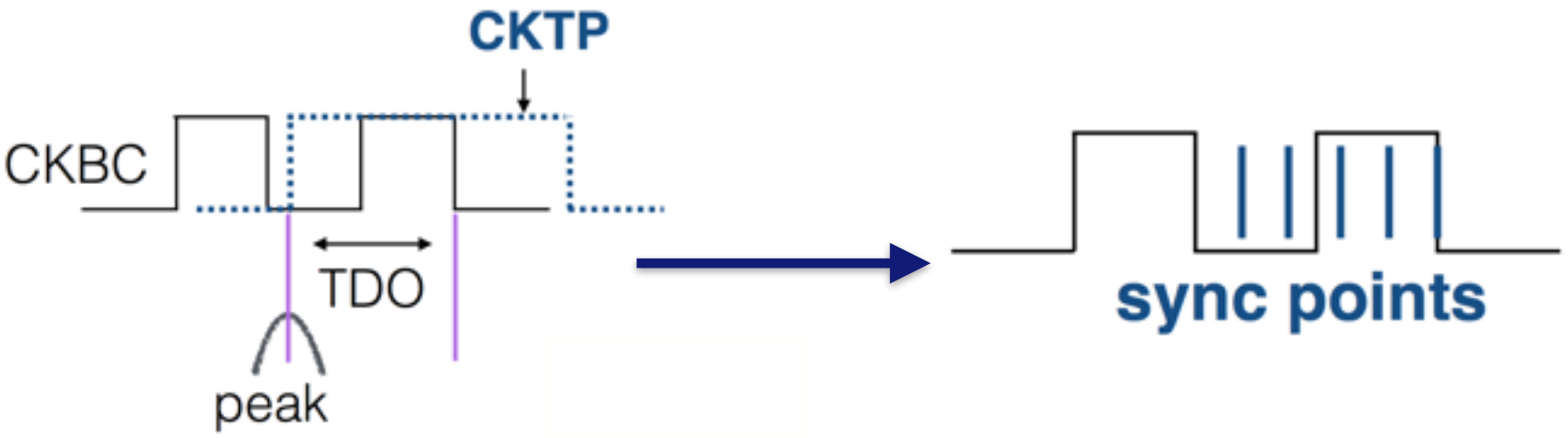


TDO calibration

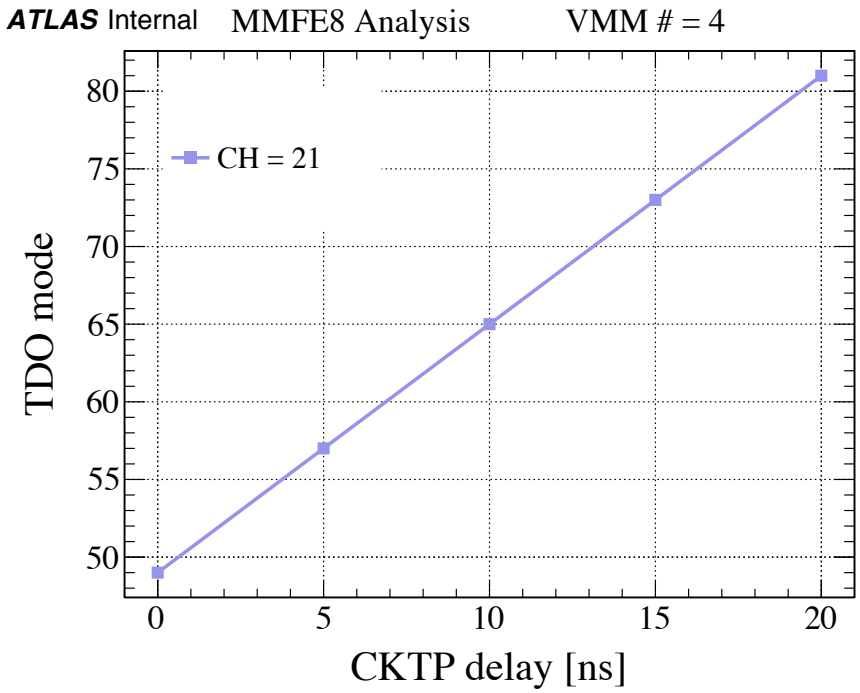
Timing (TDO) measurement calibration through varying timing delay by:

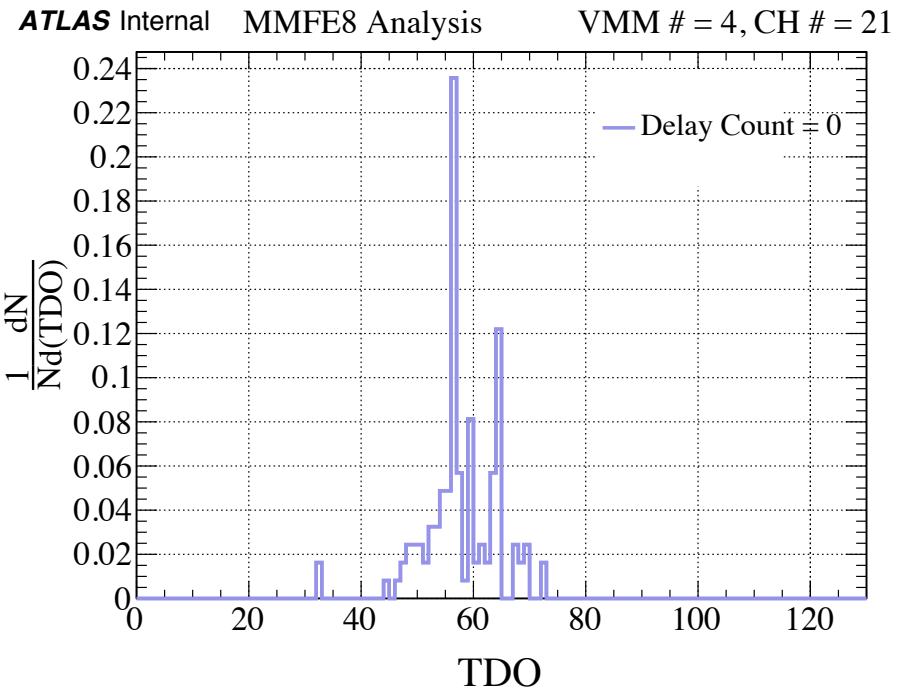
- Synchronizing CKTP to CKBC
- Delaying CKTP relative to CKBC by N "delay counts" on 200 MHz mother clock (5 ns steps)

See **Ann Wang**'s talk for more details: (https://indico.cern.ch/event/496030/)



Timing calibration





- Can see effect of 5 ns delay counts on measured TDO
- First look reveals relatively wide TDO distributions
- Distribution becomes narrower for increasing PDO - this dependence of TDO resolution on PDO previously not observed with VMM1 ⇒ under investigation

Outlook

- Goal of working DAQ for Harvard cosmic-ray telescope with MicroMegas octuplet + MMFE-8 +VMM2 progressing rapidly on several fronts
- Stress test of MMFE-8 readout with automated calibration routine revealed several issues ⇒ have developed fixes for all observed problems
  - Masking problematic channels
  - Additional system resets to clear FPGA FIFO
  - Timing fixes to firmware for asynchronous data signals from VMMs
- Now proceeding with timing and amplitude calibration for all VMMs/channels on available MMFE-8's (currently 2)

Additional Slides

Previous Notes and Talks

- "Test of a resistive micromega v3.0 prototype with VMM1 readout using ≥ 0.8 GeV/c^2 cosmic muons",
  ATL-COM-UPGRADE-2014-038
- "Test of the VMM1 Address in Real Time (ART) output using ≥ 0.8 GeV/c^2 cosmic muons",
  ATL-COM-MUON-2014-069
- "Re-analysis of the 2012 test-beam data",
  ATL-COM-MUON-2014-051
- "Bench test of VMM2 mini-1 boards",
  ATL-COM-MUON-2015-078
- Measurement of the Noise in an MMFE-8 Front-End Board,
  https://indico.cern.ch/event/465405/
- DAQ System for Testing the Micromega Trigger with Cosmic Rays at Harvard,
  https://indico.cern.ch/event/496030/