DOM-Resident Calibration Software

John Kelley
UW-Madison
July 29, 2004
Overview

- Stand-alone program runs on DOM mainboard
- Calibrates ATWDs, amplifiers, and PMT gain
- Stores results on DOM flash filesystem
- Authors: J. Kelley, J. Braun (derived from K. Hanson’s domcal.py)
ATWD Calibration

• Calibrate ATWD Y-axis to voltage
  – Pulser calibration
  – ATWD bin calibration
  – Amplifier calibration

• Calibrate ATWD X-axis to time

• Runs in ~1 minute
Step 1: Pulser Calibration

- Set pulser to known frequency and discriminator to known level
- Monitor trigger rate while adjusting pulser amplitude
- Translate amplitude DAC at 50% rate point to true amplitude (V), using known discriminator level
- Fit amplitude DAC to voltage relationship
Step 2: ATWD Bin Calibration

- Set front-end pedestal (bias) voltage to known value

- Record average ATWD pedestal — baseline shifts as bias is changed

- Fit relationship for each ATWD, channel, and bin (accounts for pedestal pattern!)

- DC bias is independent of channel amplification
Step 3: Amplifier Calibration

- Capture pulser waveforms of appropriate amplitude, find peak, convert to volts (from step 2)

- Convert pulser amplitude to volts (from step 1)

- Find mean and error of amplification for each channel
Step 4: Sampling Speed Calibration

- Capture clock waveforms using ATWD channel 3
- Count average number of positive zero-crossings
- Fit multiple of clock frequency versus sampling speed DAC
ATWDA Sampling Speed vs. DAC Setting

Data

0.0112\times x + 5.01
HV Gain Calibration

- Integrate SPE waveforms
  - Dynamically choose channel 0 or 1 to avoid saturation
  - Use previous calibration data to convert to pC

- Fit charge histogram

- Find peak and valley

- Fit log(HV) versus log(gain), using only good points

- Runs in 5-10 minutes
Charge Histogram Fitting

- Nonlinear Levenberg-Marquardt fit to:
  \[ A \cdot \exp(-Bx) + C \cdot \exp(-E(x-D)^2) \]

- Discard first few bins intelligently, pick starting parameter values

- Runs reasonably fast on DOM (< 5 sec)
P/V Calculation

- Find valley with Newton’s algorithm on derivative of fit
- Approximate peak with Gaussian max
- Heuristics
  - Did fit converge?
  - Is valley in sane location?
  - Is P/V realistic?
Gain vs. HV

- Calculate gain using SPE peak from fit

- Fit log(HV) versus log(gain) for voltages with good data

- Return fit and individual P/V points
Output Data

- Data stored on DOM flash filesystem (binary format)
  - Calibration results
  - DOM state (DACs, ADCs, temperature)

- Java application can control calibration, read binary result file, and create XML file on surface
Open Issues

- ATWDB may have small baseline shift — especially affects channel 2 gain calibration

- HV gain calibration uses hard-coded discriminator setting and doesn’t work well at room temperature

- Gain is very reproducible, but exact P/V ratio is variable

- New Rev5.0 ATWDs need different DAC settings — will support all revisions through #defines