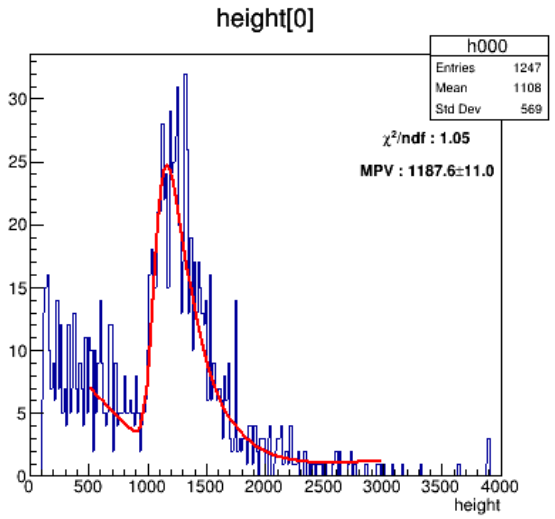
190408

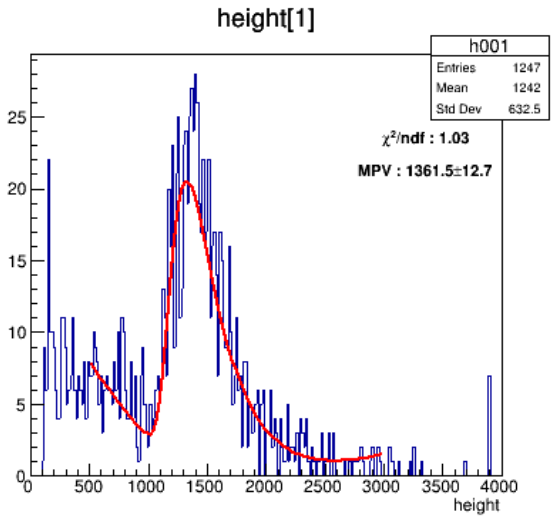
ByungChan lee

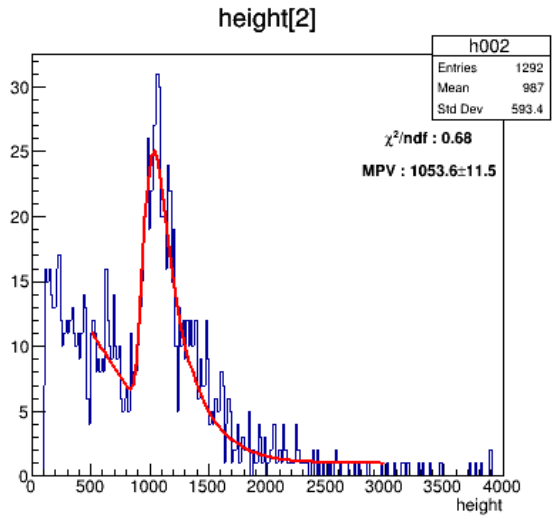
Gain Calibration

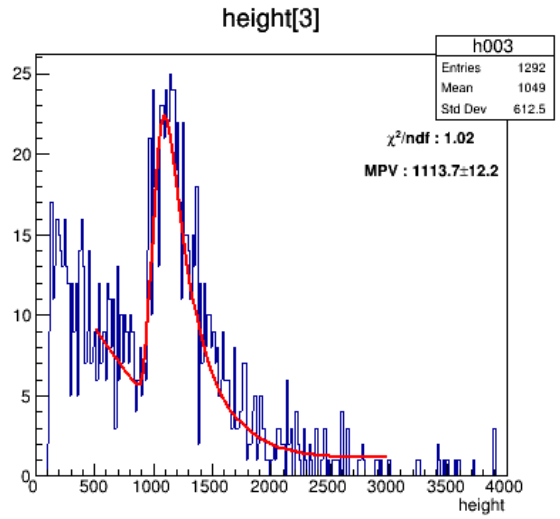
- Put a bar between two bars
- Incident angle will be limited

- Better result is expected for 6 multiplicity



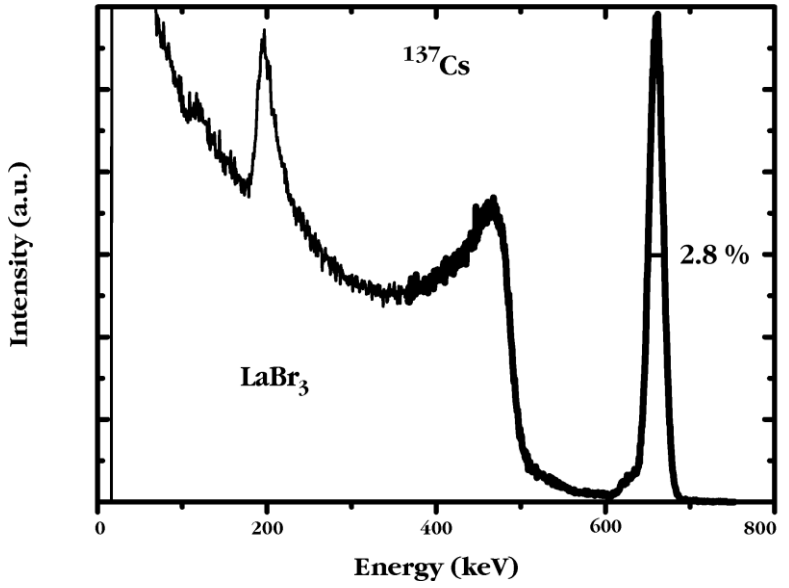




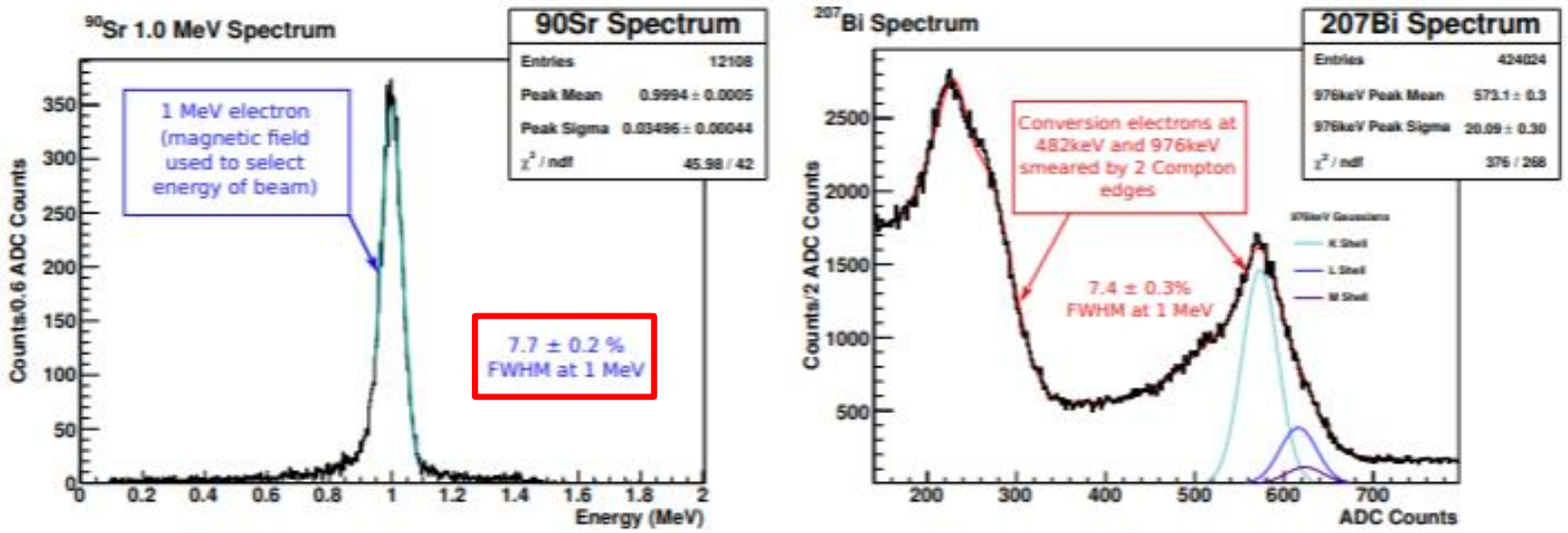


Comparing Energy Resolution (height / Integration)

- Energy Resolution can be measured by sending monoenergetic beam of radiation
- Energy Resolution = ΔE / E
  - ΔE : The Full Width at Half Maximum (FWHM)
  - FWMH = $2\sqrt{2\ ln2}\sigma\ \approx 2.35\sigma$
  - $R \sim 1/\sqrt{E}$
- In general, pulse height distribution is used
- $(\Delta E)^2 = (\Delta E_{scin})^2 + (\Delta E_{pmt})^2 + (\Delta E_{elect})^2 + \ \ldots$

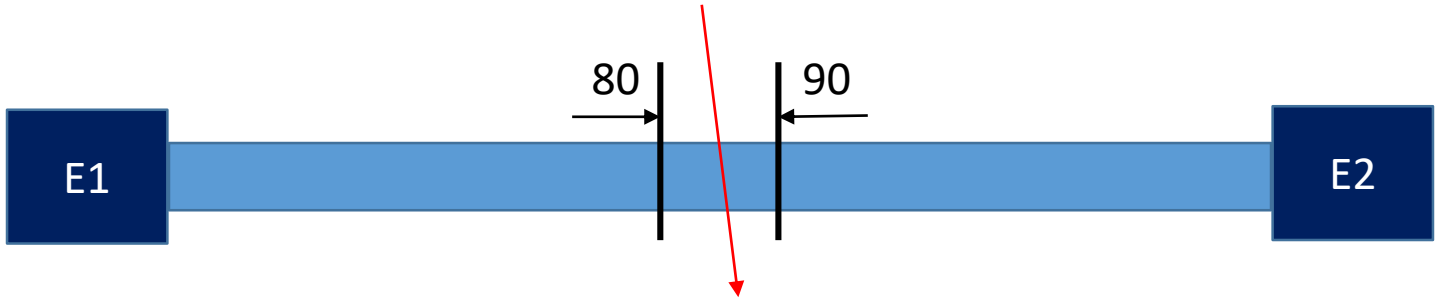


M. Moszyński , 'Energy resolution of scintillation detectors', 2005



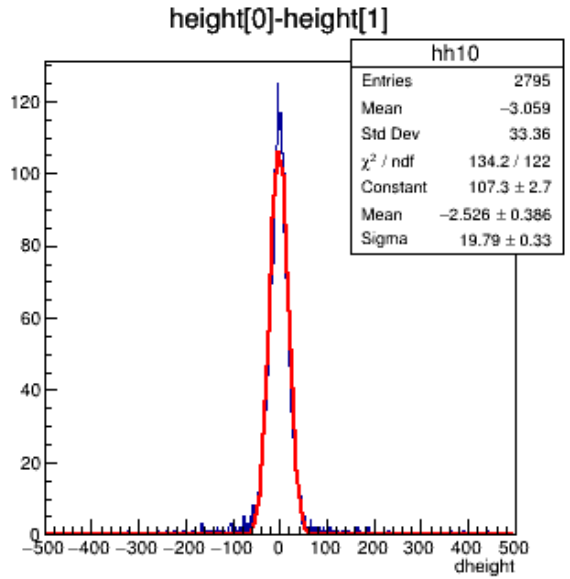
(a) ^{90}Sr method (b) ^{207}Bi method

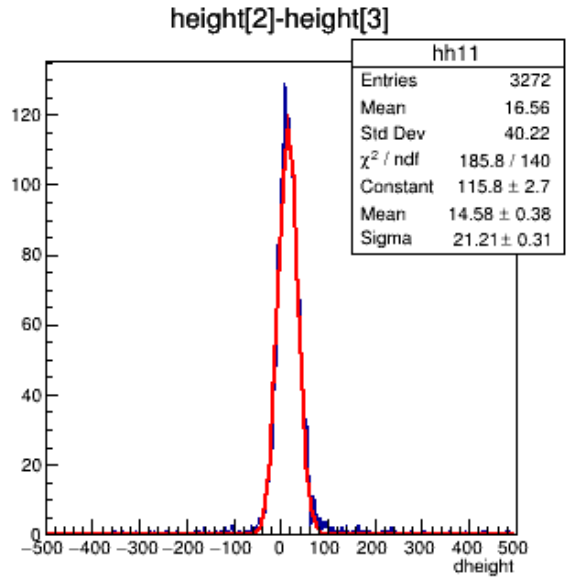
Figure 4: Two different fitting procedures for energy resolution acquisition using (a) ^{90}Sr and (b) ^{207}Bi for a 255mm ⌀ ×180mm hexagonal EJ-200 block coupled to an 8" Hamamatsu R5912-MOD PMT via Glycerol coupling fluid, with 75μm PTFE wrapping on the sides and 12μm Mylar wrapping on the entrance face. The two methods are consistent with each other, with a $\Delta E/E$ of 7.7±0.2% FWHM at 1MeV measured with ^{90}Sr and of 7.4±0.3% FWHM at 1MeV measured with ^{207}Bi. Systematic corrections have been taken into account for both of the results, introduced by the respective data acquisition systems.

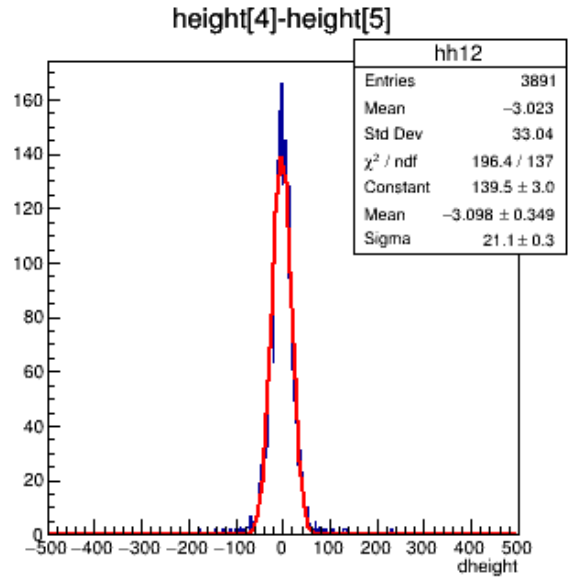


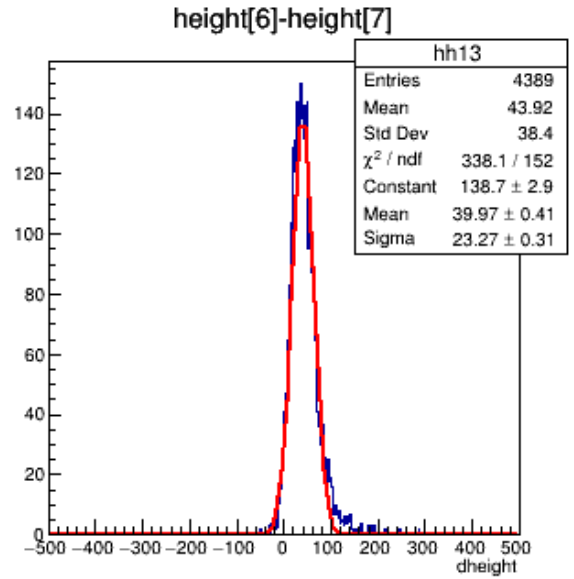
- Center Cut : 80cm < Hit position < 90cm
- Assume that two PMTs measure the same energy
- The gains of 2 PMTs must be calibrated first

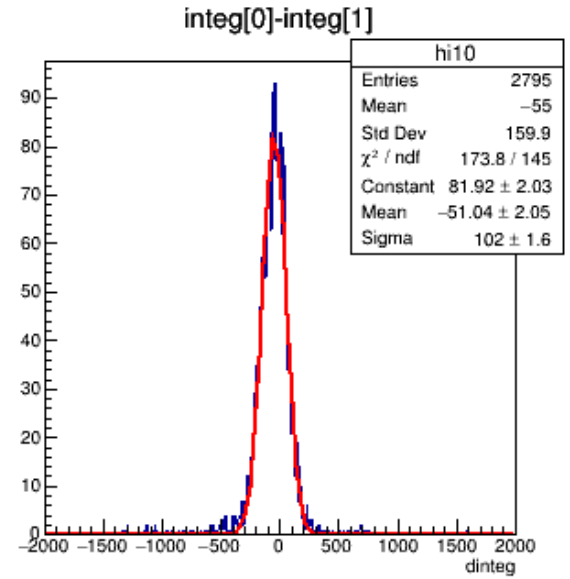
- (height or Integral) ΔE = E1 – E2 $\Rightarrow$ ${\sigma_{\Delta E}}^2 = {\sigma_{E1}}^2 + {\sigma_{E2}}^2$ ($\approx 2{\sigma_{E1}}^2$)
- R = (2.35$\sqrt{{\sigma_{\Delta E}}^2/2}$) / MPV

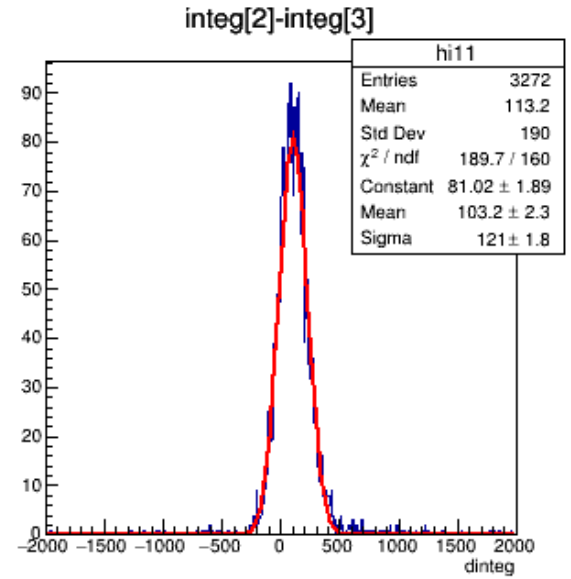

height[0]-height[1]
hh10
Entries 2795
Mean −3.059
Std Dev 33.36
χ² / ndf 134.2 / 122
Constant 107.3 ± 2.7
Mean −2.526 ± 0.386
Sigma 19.79 ± 0.33
dheight

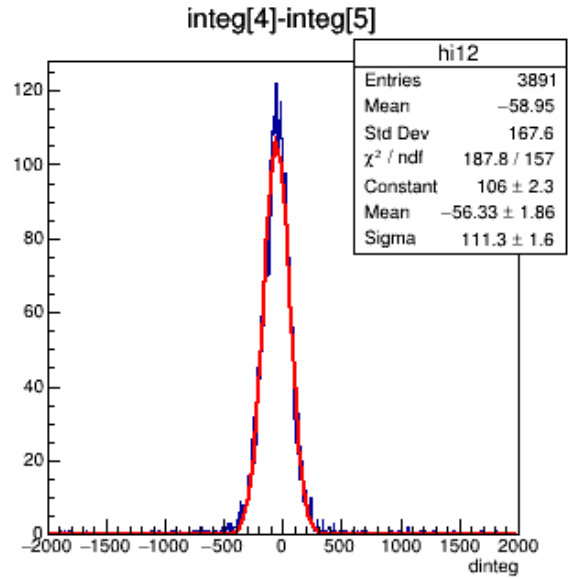

height[2]-height[3]
hh11
Entries 3272
Mean 16.56
Std Dev 40.22
χ² / ndf 185.8 / 140
Constant 115.8 ± 2.7
Mean 14.58 ± 0.38
Sigma 21.21 ± 0.31
dheight

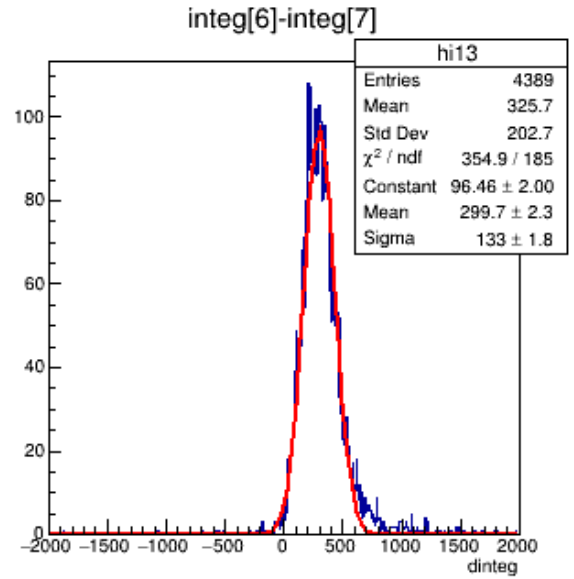

height[4]-height[5]
hh12
Entries 3891
Mean −3.023
Std Dev 33.04
χ² / ndf 196.4 / 137
Constant 139.5 ± 3.0
Mean −3.098 ± 0.349
Sigma 21.1 ± 0.3
dheight

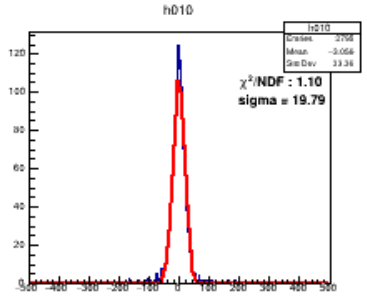

height[6]-height[7]
hh13
Entries 4389
Mean 43.92
Std Dev 38.4
χ² / ndf 338.1 / 152
Constant 138.7 ± 2.9
Mean 39.97 ± 0.41
Sigma 23.27 ± 0.31
dheight

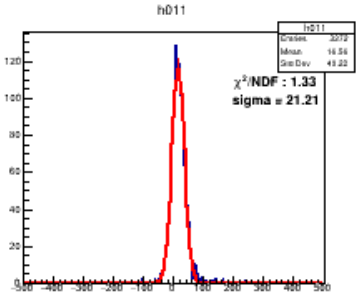

integ[0]-integ[1]
hi10
Entries 2795
Mean −55
Std Dev 159.9
χ² / ndf 173.8 / 145
Constant 81.92 ± 2.03
Mean −51.04 ± 2.05
Sigma 102 ± 1.6
dinteg

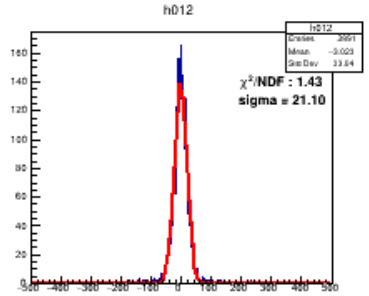

integ[2]-integ[3]
hi11
Entries 3272
Mean 113.2
Std Dev 190
χ² / ndf 189.7 / 160
Constant 81.02 ± 1.89
Mean 103.2 ± 2.3
Sigma 121 ± 1.8
dinteg

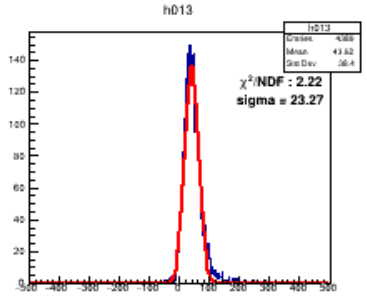

integ[4]-integ[5]
hi12
Entries 3891
Mean −58.95
Std Dev 167.6
χ² / ndf 187.8 / 157
Constant 106 ± 2.3
Mean −56.33 ± 1.86
Sigma 111.3 ± 1.6
dinteg

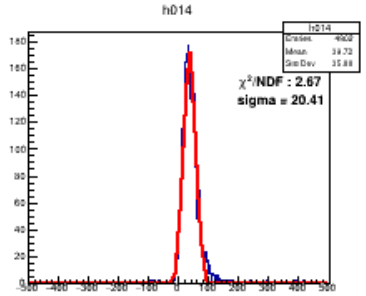

integ[6]-integ[7]
hi13
Entries 4389
Mean 325.7
Std Dev 202.7
χ² / ndf 354.9 / 185
Constant 96.46 ± 2.00
Mean 299.7 ± 2.3
Sigma 133 ± 1.8
dinteg

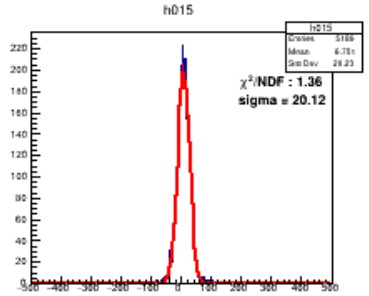

h010
χ²/NDF : 1.10
sigma = 19.79

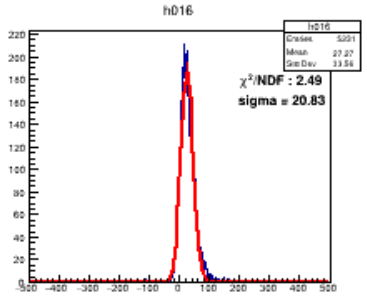

h011
χ²/NDF : 1.33
sigma = 21.21

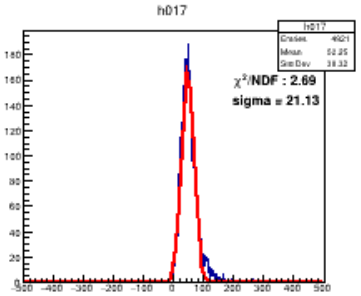

h012
χ²/NDF : 1.43
sigma = 21.10

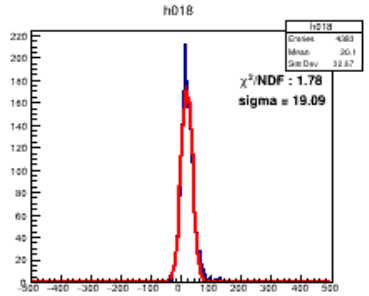

h013
χ²/NDF : 2.22
sigma = 23.27

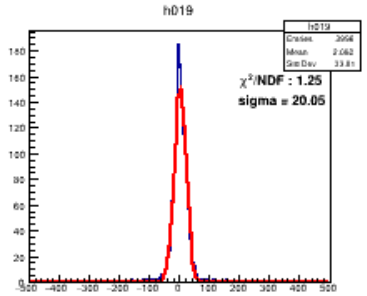

h014
χ²/NDF : 2.67
sigma = 20.41

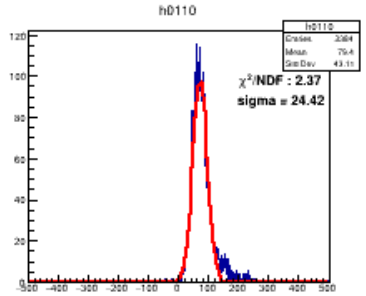

h015
χ²/NDF : 1.36
sigma = 20.12

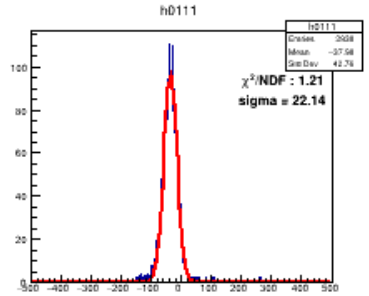

h016
χ²/NDF : 2.49
sigma = 20.83

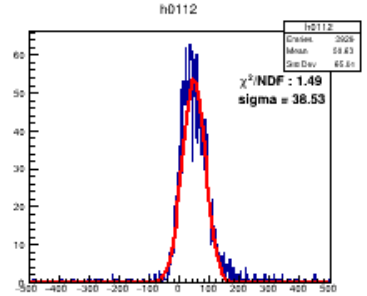

h017
χ²/NDF : 2.69
sigma = 21.13

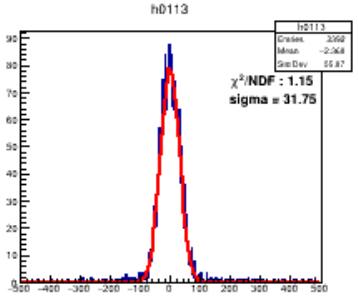

h018
χ²/NDF : 1.78
sigma = 19.09

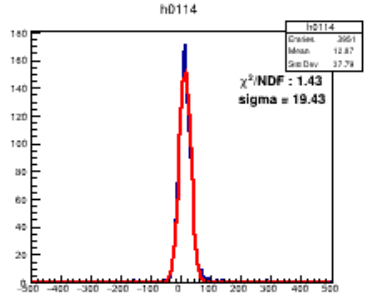

h019
χ²/NDF : 1.25
sigma = 20.05

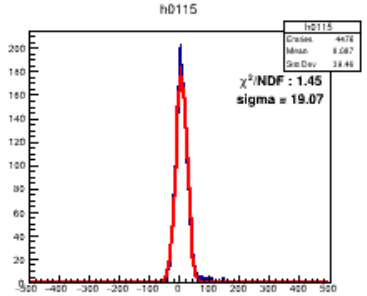

h0110
χ²/NDF : 2.37
sigma = 24.42

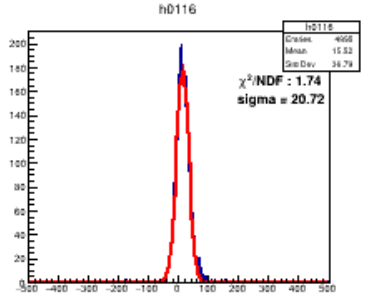

h0111
χ²/NDF : 1.21
sigma = 22.14

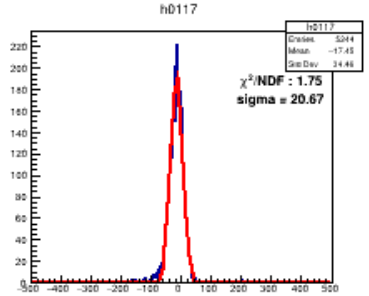

h0112
χ²/NDF : 1.49
sigma = 38.53

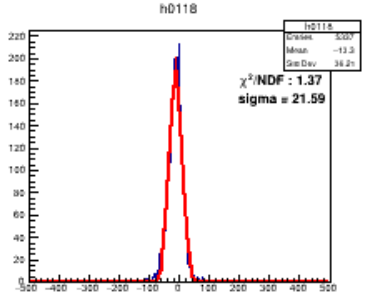

h0113
χ²/NDF : 1.15
sigma = 31.75

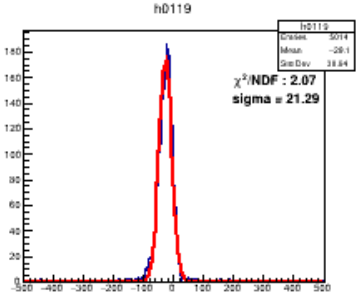

h0114
χ²/NDF : 1.43
sigma = 19.43

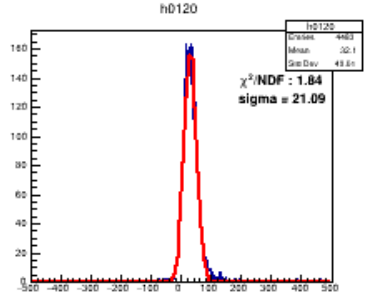

h0115
χ²/NDF : 1.45
sigma = 19.07

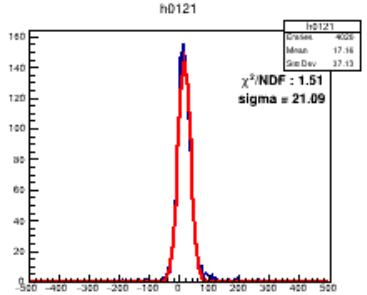

h0116
χ²/NDF : 1.74
sigma = 20.72

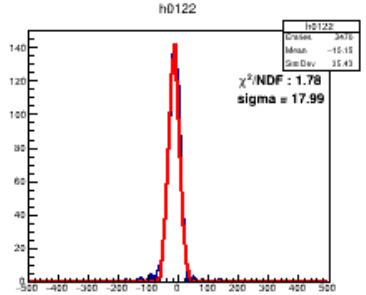

h0117
χ²/NDF : 1.75
sigma = 20.67

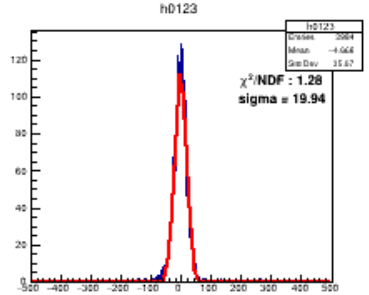

h0118
χ²/NDF : 1.37
sigma = 21.59

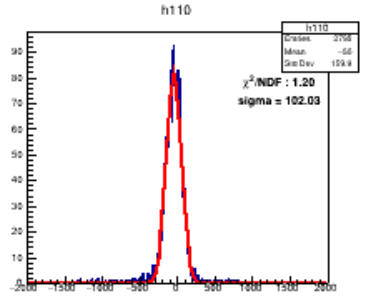

h0119
χ²/NDF : 2.07
sigma = 21.29

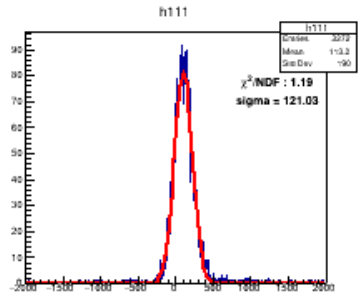

h0120
χ²/NDF : 1.84
sigma = 21.09

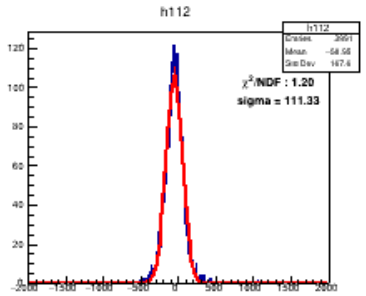

h0121
χ²/NDF : 1.51
sigma = 21.09

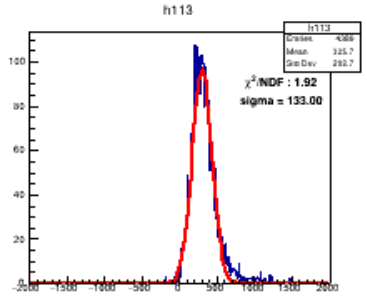

h0122
χ²/NDF : 1.78
sigma = 17.99

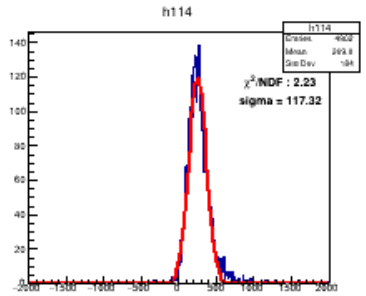

h0123
χ²/NDF : 1.28
sigma = 19.94

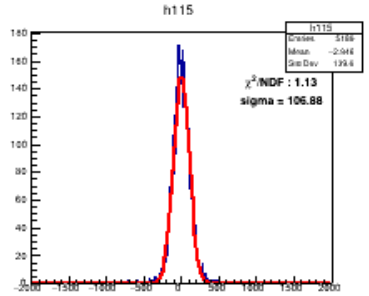

h110
χ²/NDF : 1.20
sigma = 102.03

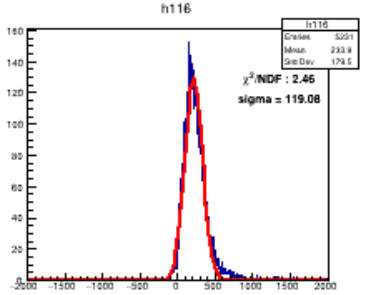

h111
χ²/NDF : 1.19
sigma = 121.03

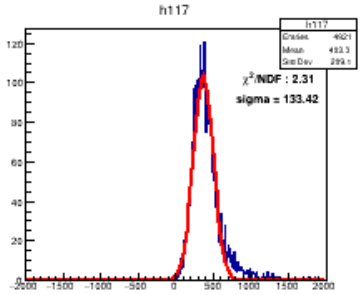

h112
χ²/NDF : 1.20
sigma = 111.33

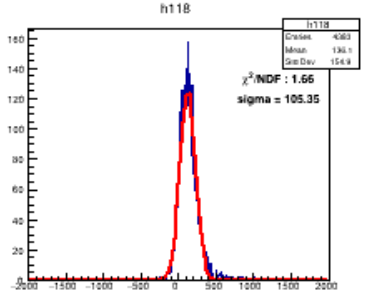

h113
χ²/NDF : 1.92
sigma = 133.00

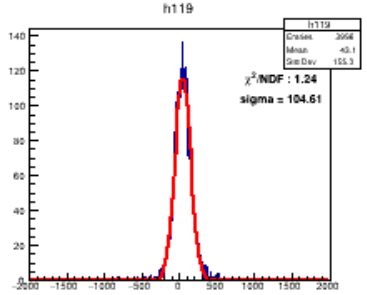

h114
χ²/NDF : 2.23
sigma = 117.32

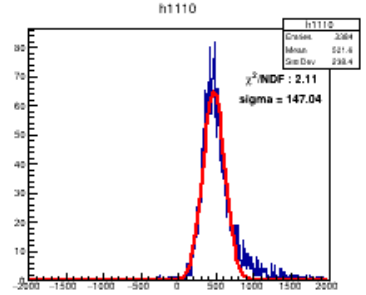

h115
χ²/NDF : 1.13
sigma = 106.88

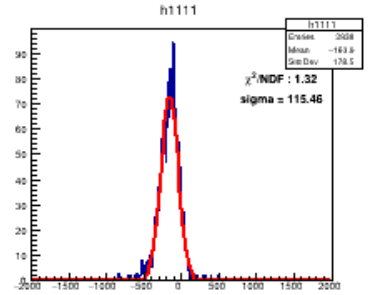

h116
χ²/NDF : 2.46
sigma = 119.08

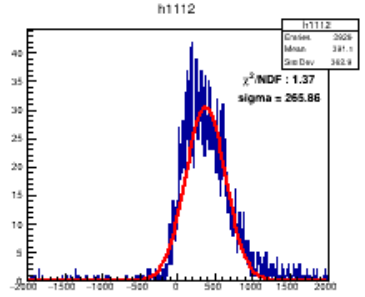

h117
χ²/NDF : 2.31
sigma = 133.42

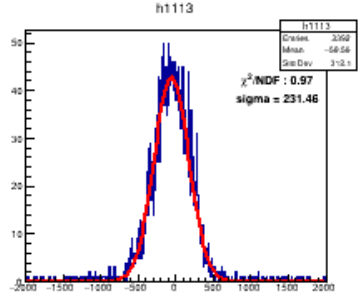

h118
χ²/NDF : 1.66
sigma = 105.35

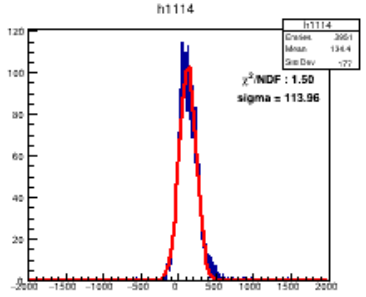

h119
χ²/NDF : 1.24
sigma = 104.61

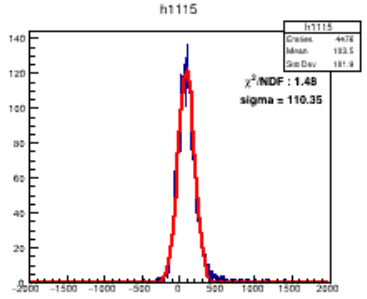

h1110
χ²/NDF : 2.11
sigma = 147.04

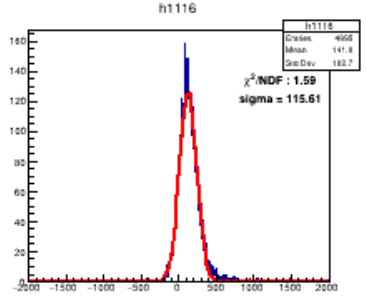

h1111
χ²/NDF : 1.32
sigma = 115.46

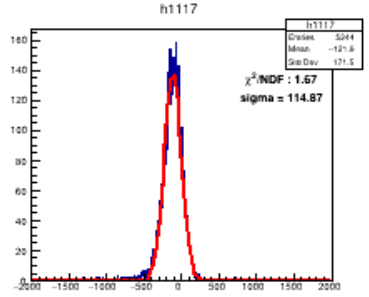

h1112
χ²/NDF : 1.37
sigma = 265.86

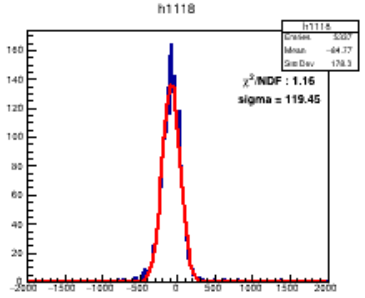

h1113
χ²/NDF : 0.97
sigma = 231.46

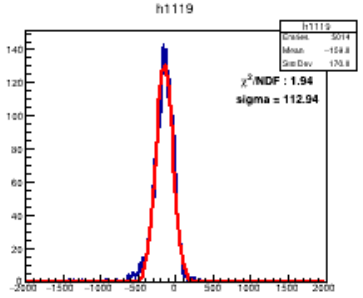

h1114
χ²/NDF : 1.50
sigma = 113.96

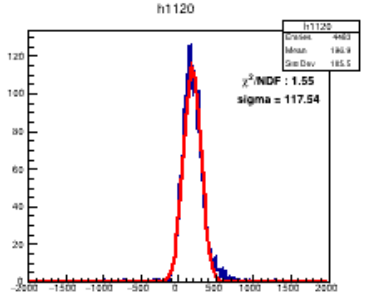

h1115
χ²/NDF : 1.48
sigma = 110.35

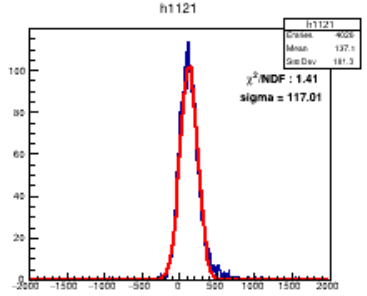

h1116
χ²/NDF : 1.59
sigma = 115.61

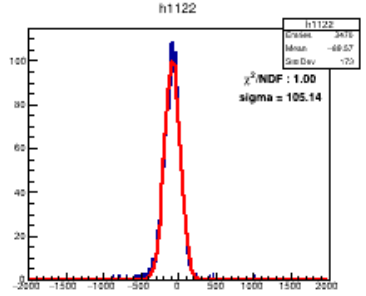

h1117
χ²/NDF : 1.67
sigma = 114.87

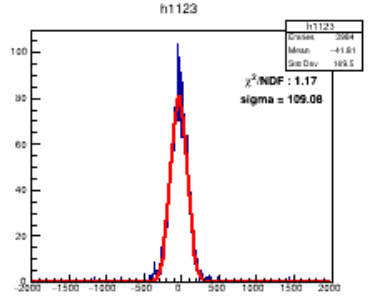

h1118
χ²/NDF : 1.16
sigma = 119.45


h1119
χ²/NDF : 1.94
sigma = 112.94


h1120
χ²/NDF : 1.55
sigma = 117.54


h1121
χ²/NDF : 1.41
sigma = 117.01


h1122
χ²/NDF : 1.00
sigma = 105.14


h1123
χ²/NDF : 1.17
sigma = 109.08

Comparison

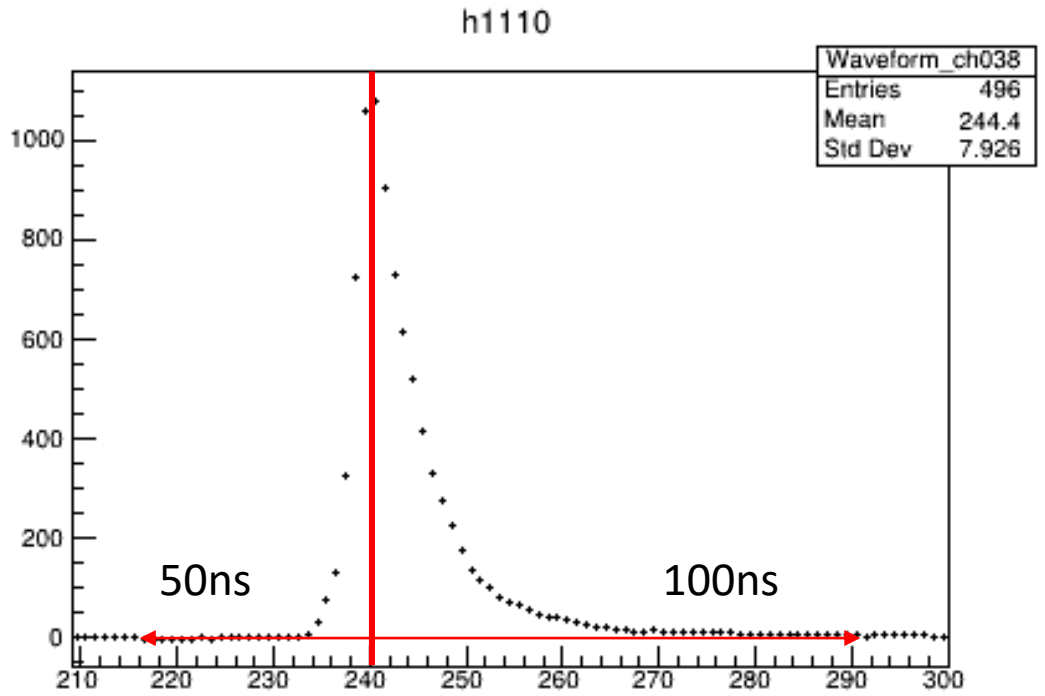
- FADCT_000747 48channels # 208,344

	Height	Integration
MPV	234.98	1818.07
Sigma (E1-E2)	21.95	127.08
Energy Resolution	15.6%	11.6%

- 20190329YTAB004 4channels # 30,000

	Height	Integration
MPV	697.5	5265.1
Sigma (E1-E2)	64.5	362.3
Energy Resolution	15.4%	11.5%

Integral on EvRec




```
Double_t CalcInteg(FChannelS* ch){

  ////////////////////////////////////////////////////
  // Calculate the pedestal
  Double_t ped = CalcPed(ch);
  ////////////////////////////////////////////////////

  ////////////////////////////////////////////////////
  // Check maximum point
  Double_t height = 0;
  Int_t iMax = -1;
  for ( Int_t i = 0 ; i < ch->GetNdp() ; i++) {
    if ( TMath::Abs(ch->GetWaveform()[i] - ped) > height ) {
      height = TMath::Abs(ch->GetWaveform()[i] - ped);
      iMax = i;}}
  ////////////////////////////////////////////////////

  ////////////////////////////////////////////////////
  // Integrate around the peak
  const Int_t befIMax = 25;
  const Int_t aftIMax = 50;
  Double_t integ = 0;
  for ( Int_t i = iMax - befIMax ; i < iMax + aftIMax ; i++)
    integ += ch->GetWaveform()[i] - ped;
  ////////////////////////////////////////////////////

  return integ;}
```