Analysis of June 2, 2013 PSI data.
This notebook starts from reduced data (after noise analysis and baseline subtraction).
SNW-July 15, 2013

In[15]:=
```
SetDirectory["~seb/Desktop/PSI_data/"];
Namelist = FileNames[]
Namelist // Length;
nfiles = %
filename = Namelist[[3]]
```

Out[16]=
```
{.DS_Store, LeCroy-compress-2013-06-02-016.csv,
 LeCroy-compress-2013-06-02-017.csv,
 LeCroy-compress-2013-06-02-018.csv, LeCroy-compress-2013-06-02-019.csv,
 LeCroy-compress-2013-06-02-020.csv, LeCroy-SGM-2013-06-02-016.csv,
 LeCroy-SGM-2013-06-02-017.csv, LeCroy-SGM-2013-06-02-018.csv,
 LeCroy-SGM-2013-06-02-019.csv, LeCroy-SGM-2013-06-02-020.csv,
 LeCroy-SGM-2013-06-02-021.csv, LeCroy-SGM-2013-06-02-022.csv}
```

Out[18]=
```
13
```

Out[19]=
```
LeCroy-compress-2013-06-02-017.csv
```

In[20]:=
```
Timing[scopedata = Import[filename, "csv"];]
```

Out[20]=
```
{6.616606, Null}
```

In[21]:=
```
phase = Table[scopedata[[i, 1]], {i, 2500}];
ampl = Table[scopedata[[i, 2]], {i, 2500}];
v1 = Table[Take[scopedata[[i]], {3, 402}], {i, 2500}];
v2 = Table[Take[scopedata[[i]], {403, 802}], {i, 2500}];
v3 = Table[Take[scopedata[[i]], {803, 1202}], {i, 2500}];
```

Combine data from multiple runs.

In[26]:=
```
Do[
  Clear[scopedata];
  filename = Namelist[[nfil]];
  Print[filename];
  scopedata = Import[filename, "csv"];
phase = Join[phase, Table[scopedata[[i, 1]], {i, 2500}]];
  ampl = Join[ampl, Table[scopedata[[i, 2]], {i, 2500}]];
  v1 = Join[v1, Table[Take[scopedata[[i]], {3, 402}], {i, 2500}]];
  v2 = Join[v2, Table[Take[scopedata[[i]], {403, 802}], {i, 2500}]];
  v3 = Join[v3, Table[Take[scopedata[[i]], {803, 1202}], {i, 2500}]];
  , {nfil, 4, 6, 1}];
```

```
LeCroy-compress-2013-06-02-018.csv

LeCroy-compress-2013-06-02-019.csv

LeCroy-compress-2013-06-02-020.csv
```

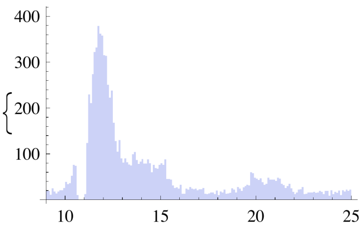
In[27]:=
```
outfile = StringReplace[filename, {"compress" → "ntuple"}]
```

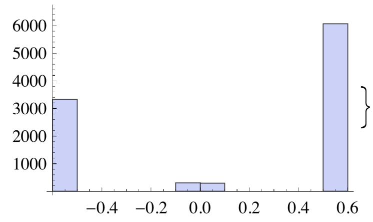
Out[27]=
```
LeCroy-ntuple-2013-06-02-020.csv
```

tof distributions

In[28]:=
```
{Histogram[phaseshift = Mod[phase, 19.75, 8.], {9, 25, .1}], Histogram[ampl]}
```

Out[28]=

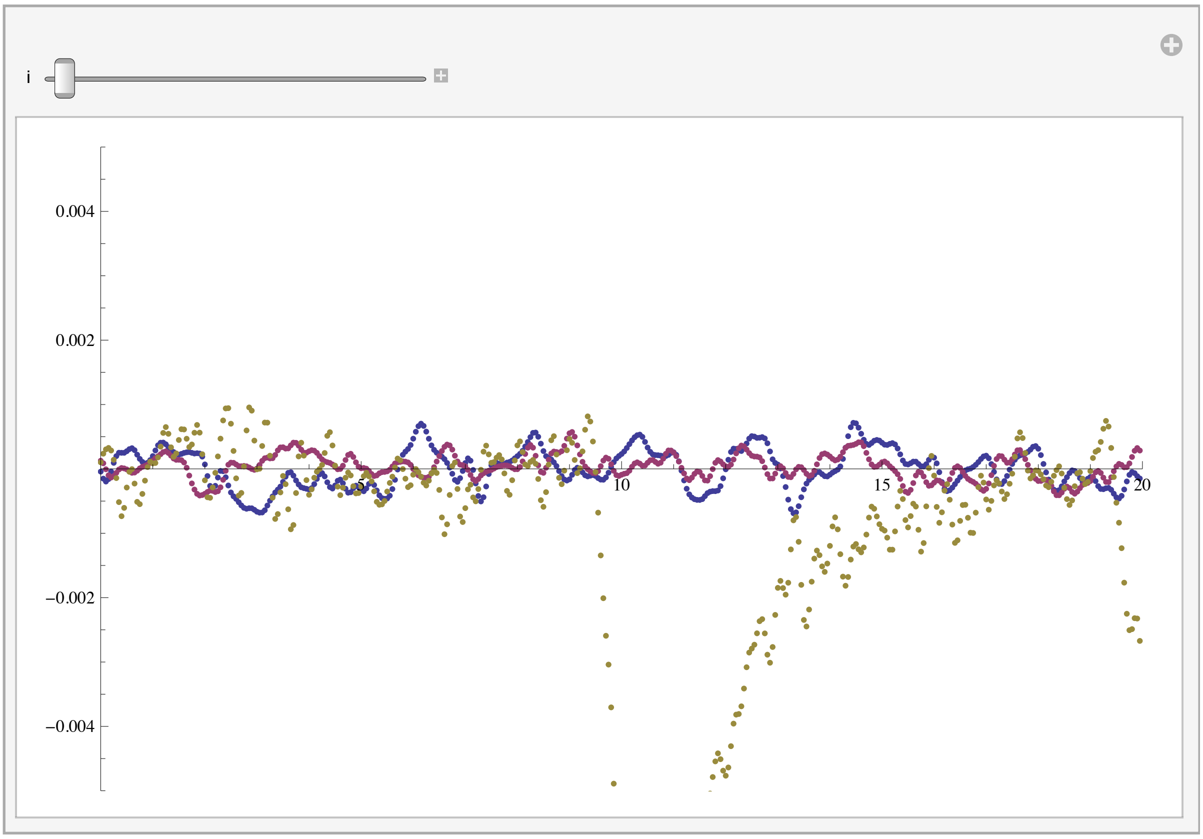




Inspect APD data.

In[29]:=
```
time = Range[0, 399] * .05;
Manipulate[ListPlot[{Transpose[{time, v1[[i]]}],
    Transpose[{time, v2[[i]]}], Transpose[{time, v3[[i]]}]},
  PlotRange → {{0, 20.}, {-.005, .005}}, ImageSize → Large], {i, 1, 2500, 1}]
```

Out[30]=

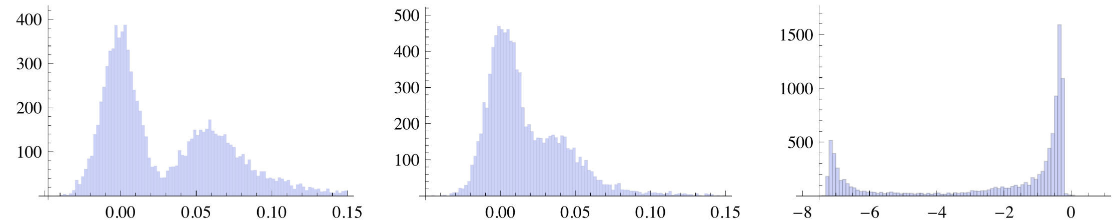


Integrate to find pulse area.

In[31]:=
```
Clear[ph1, ph2, ph3];
ph1 = Table[Sum[v1[[i, j]], {j, 100, 299, 1}], {i, 10000}];
ph2 = Table[Sum[v2[[i, j]], {j, 100, 299, 1}], {i, 10000}];
ph3 = Table[Sum[v3[[i, j]], {j, 100, 299, 1}], {i, 10000}];
```

In[35]:=
```
GraphicsRow[{Histogram[ph1, {-.05, .15, .002}], Histogram[ph2, {-.05, .15, .002}],
  Histogram[ph3, {-8, 1, .1}]}, ImageSize → Large]
```

Out[35]=



Now calculate expected pulse area from Q = qe*ne*APDGain*Ampgain, V/50 Ohms = i
Gains and detector Capacitance dependent deficit from Voltage Amplifier Model.
Looks like there is a factor of 4 less integrated charge than would predict for either 64mm^2
or 4mm^2 detectors.
The discrepancy is possibly even larger since the APD gain is likely a bit more than 600 at this bias.

In[36]:=
```
qe = -1.60217 * 10^-19;
ne = 6000; APDgain = 600;
defecit4mm = 0.5; defecit64 = 0.09;
NSolve[13 == 20 Log10[vgain], vgain] /. Rule → List;
Ampgain13 = %[[1, 1, 2]]
NSolve[20 == 20 Log10[vgain], vgain] /. Rule → List;
Ampgain20 = %[[1, 1, 2]]
```

Out[40]= 4.46684

Out[42]= 10.

In[43]:=
```
NSolve[vsum == 50 / (.05 * 10^-9) * qe * ne * APDgain * Ampgain13, vsum] /. Rule → List
pulsearea3 = %[[1, 1, 2]] * defecit4mm
```

Out[43]= {{{vsum, -2.57639}}}

Out[44]= -1.28819

In[45]:=
```
NSolve[vsum == 50 / (.05 * 10^-9) * qe * ne * APDgain * Ampgain20, vsum] /. Rule → List
pulsearea1 = %[[1, 1, 2]] * defecit64
```

Out[45]= {{{vsum, -5.76781}}}

Out[46]= -0.519103

Now create an n - Tuple to fish out the characteristics of the MIP signal and write it to file.
Also define some obvious cuts.

In[47]:=
```
ntuple = Transpose[{phaseshift, ampl, ph1, ph2, ph3}];
```

In[48]:=
```
Export[outfile, ntuple, "csv"]
```

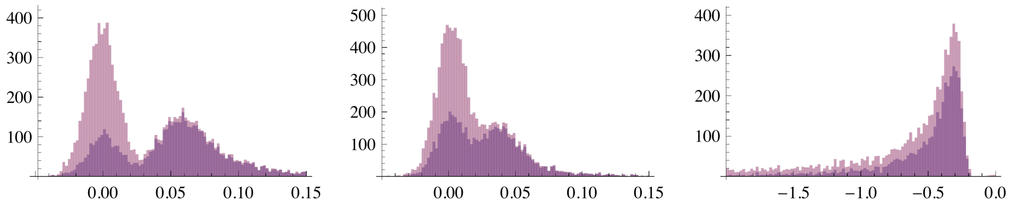
Out[48]=
```
LeCroy-ntuple-2013-06-02-020.csv
```

In[49]:=
```
phasepos = 1; amplpos = 2;
ph1pos = 3; ph2pos = 4; ph3pos = 5;
ph1and2ok[datum_List] :=
 (0.02 < Part[datum, ph2pos] < .1) && (.03 < Part[datum, ph1pos] < .15)
phasepi[datum_List] := (11 < Part[datum, phasepos] < 13)
ph3ok[datum_List] := (-1.2 < Part[datum, ph3pos] < -.2)
ph1ok[datum_List] := (0.03 < Part[datum, ph1pos] < .15)
ph2ok[datum_List] := (.02 < Part[datum, ph2pos] < 0.1)
amplok[datum_List] := (.4 < Part[datum, amplpos]) &&
  (.03 < Part[datum, ph1pos] < .15) && (0.02 < Part[datum, ph2pos] < .1)
```

Conditional Histograms

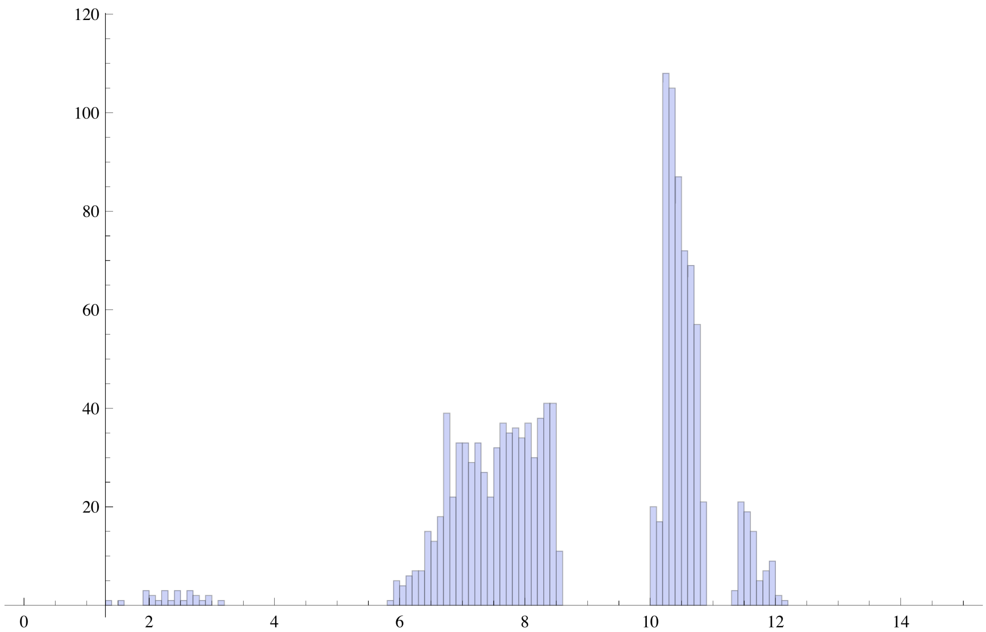
In[57]:=
```
GraphicsRow[
 {Histogram[{Select[ntuple, ph3ok][[All, ph1pos]], ntuple[[All, ph1pos]]},
   {-.05, .15, .002}], Histogram[{Select[ntuple, ph3ok][[All, ph2pos]],
    ntuple[[All, ph2pos]]}, {-.05, .15, .002}], Histogram[
   {Select[ntuple, ph1ok][[All, ph3pos]], ntuple[[All, ph3pos]]}, {-2, 0, .02}]}]
```

Out[57]=



In[58]:=
```
shiftedTOF = -(Select[ntuple, amplok][[All, phasepos]] - 22);
Histogram[shiftedTOF, {0, 15, .1}]
```

Out[59]=



The Tof Spectrum has roughly the expected time difference for

e, mu, and pions - see below.
The pion peak itself is split, probably because the clock period falls in the middle of the peak.
Clearly the clock phase analysis could use a little cleaning up.

In[60]:=
```
mu = ParticleData["Muon", "Mass"]; mpi = ParticleData["PiPlus", "Mass"];
me = ParticleData["Electron", "Mass"];
Solve[250 == mpi * beta / Sqrt[1 - beta * beta], beta] /. Rule → List;
vpi = %[[1, 1, 2]]
```

Out[62]= 0.87314524

In[63]:=
```
Solve[250 == me * beta / Sqrt[1 - beta * beta], beta] /. Rule → List;
ve = %[[1, 1, 2]]
```

Out[64]= 0.99999791

In[65]:=
```
Solve[250 == mu * beta / Sqrt[1 - beta * beta], beta] /. Rule → List;
vmu = %[[1, 1, 2]]
```

Out[66]= 0.921113762

In[67]:=
```
Solve[tofe == 2360 / 30 * (1 / ve - 1 / vpi), tofe]
```

Out[67]= {{tofe → -11.42891}}

In[68]:=
```
Solve[tofmu == 2360 / 30 * (1 / vmu - 1 / vpi), tofmu]
```

Out[68]= {{tofmu → -4.691885}}

In[69]:=
```
Dimensions[ph1]
Dimensions[v1]
```

Out[69]= {10 000}

Out[70]= {10 000, 400}

Now write out a set of event waveforms that are selected with the above cuts.

In[71]:=
```
Clear[v1write, v2write , v3write]
ngoodevents = 0; v1write = ConstantArray[0, {2704, 400}];
v2write = ConstantArray[0, {2365, 400}]; v3write = ConstantArray[0, {2365, 400}];
```

In[74]:=
```
Do[
  If[(.03 < ph1[[i]] < .15) &&
    (.02 < ph2[[i]] < .1) && (-1.2 < ph3[[i]] < -.2), , Goto[nogood]];
  ngoodevents ++;
  v1write[[ngoodevents]] = v1[[i]];
  v2write[[ngoodevents]] = v2[[i]]; v3write[[ngoodevents]] = v3[[i]];
  Label[nogood];
  , {i, 10 000}];
ngoodevents
```

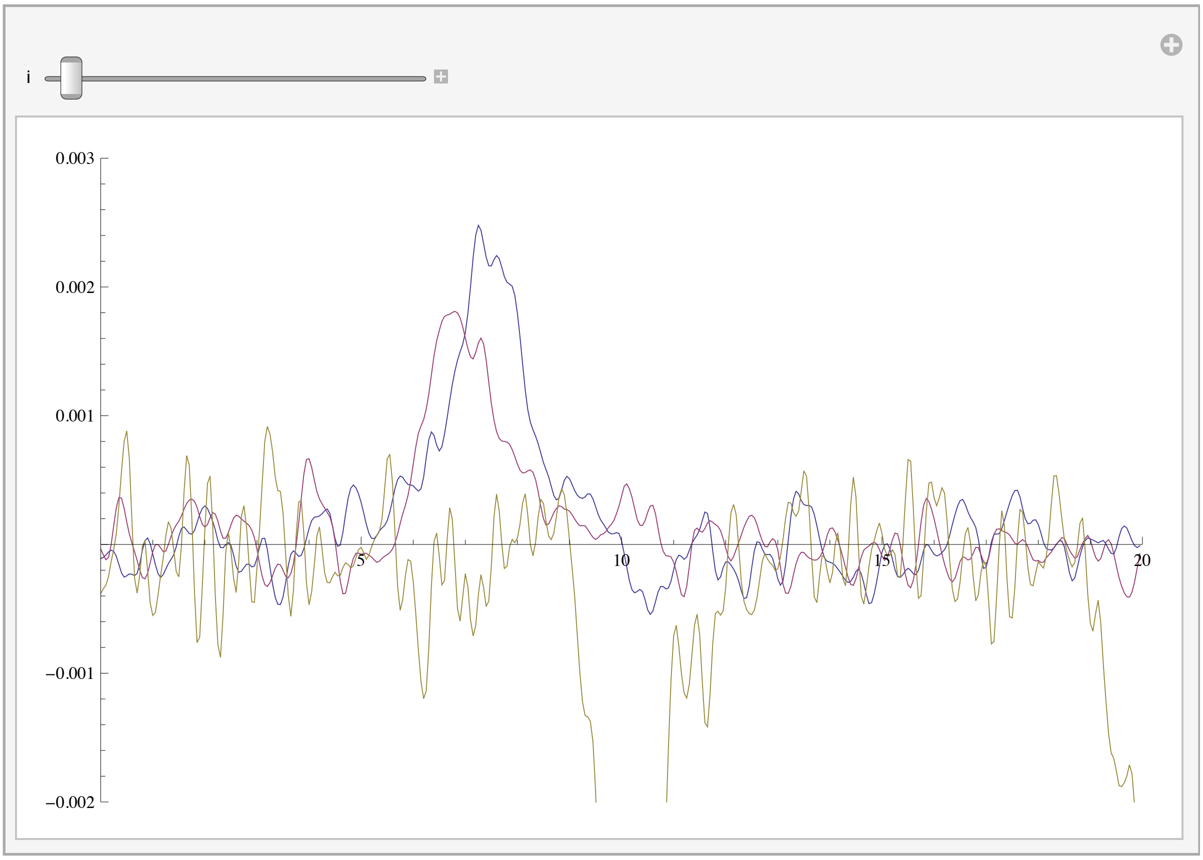
Out[75]= 2365

```
In[76]:= outfilewave = StringReplace[filename, {"compress" → "wavecut"}]
         Export[outfilewave, {v1write, v2write, v3write}, "csv"];

Out[76]= LeCroy-wavecut-2013-06-02-020.csv

In[78]:= Manipulate[ListPlot[{Transpose[{time, v1write[[i]]}],
            Transpose[{time, v2write[[i]]}], Transpose[{time, v3write[[i]]}]},
          PlotRange → {{0, 20.}, {-.002, .003}}, Joined → True, ImageSize → Large],
         {i, 1, 2365, 1}, SaveDefinitions → True]
```

Out[78]=

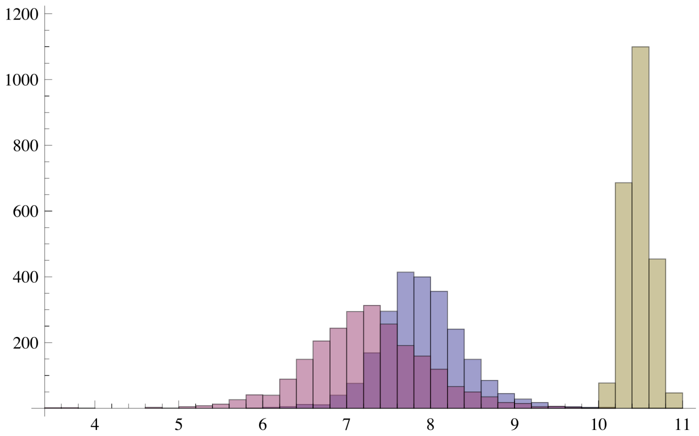


Since noise is so large in this run try timing just from the pulse centroid.

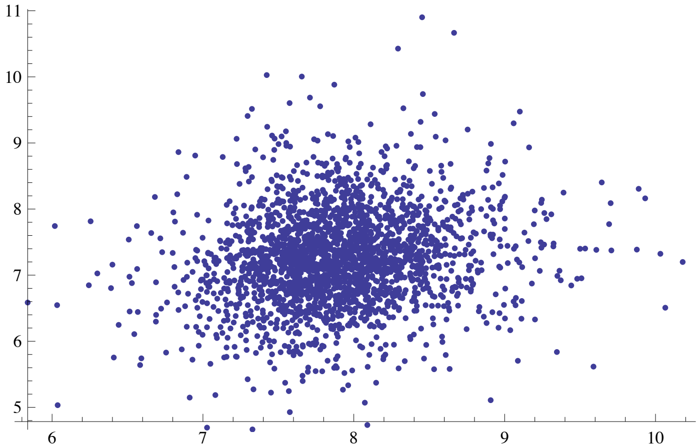
```
In[79]:= time3 = Take[time, {180, 239}]; time1 = Take[time, {100, 239}];

In[80]:= v1centroid = Table[(time1.Take[v1write[[i]], {100, 239}]) /
             Sum[v1write[[i, j]], {j, 100, 239}], {i, 2365}];
         v2centroid = Table[(time1.Take[v2write[[i]], {100, 239}]) /
             Sum[v2write[[i, j]], {j, 100, 239}], {i, 2365}];
         v3centroid = Table[(time3.Take[v3write[[i]], {180, 239}]) /
             Sum[v3write[[i, j]], {j, 180, 239}], {i, 2365}];
```

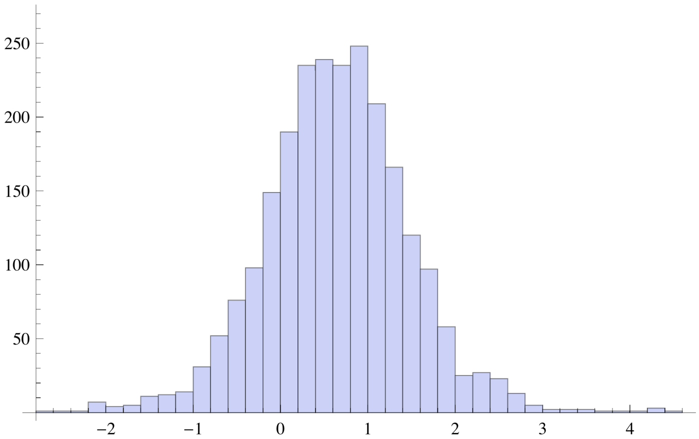
In[83]:=
```
Histogram[{v1centroid, v2centroid, v3centroid}]
```

Out[83]=


In[84]:=
```
ListPlot[Transpose[{v1centroid, v2centroid}]]
```

Out[84]=


In[85]:=
```
Histogram[vdif = (v1centroid - v2centroid)]
```

Out[85]=


As you can see directly from the scope traces, these data have a huge, noise dominated, time jitter.

In[86]:=

```
v3ave = Mean[v3centroid]
RootMeanSquare[(v3centroid - v3ave)]
```

Out[86]= 10.4748

Out[87]= 0.153843

In[88]:=

```
ave1 = Mean[v1centroid]
RootMeanSquare[(v1centroid - ave1)]
```

Out[88]= 7.90426

Out[89]= 0.507667

In[90]:=

```
avedif = Mean[vdif];
RootMeanSquare[(vdif - avedif)]
```

Out[91]= 0.830316

Now Plot the pulse peak amplitude for the selected set of MIP candidates.

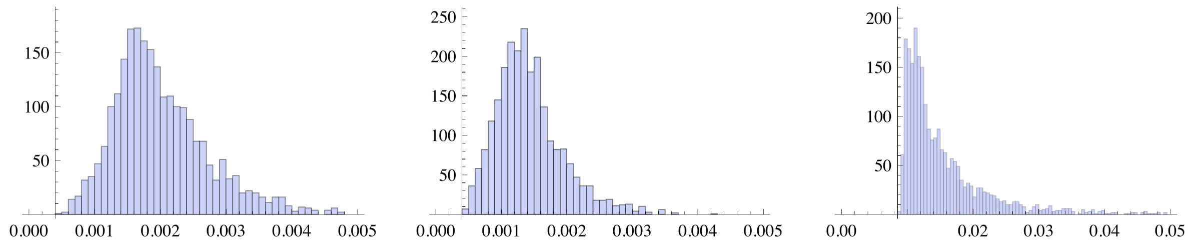
In[92]:=

```
v1peak = Table[Max[Take[v1write[[i]], {100, 239}]], {i, 2365}];
v2peak = Table[Max[Take[v2write[[i]], {100, 239}]], {i, 2365}];
v3peak = Table[Max[Take[-v3write[[i]], {180, 239}]], {i, 2365}];
```

In[102]:=

```
GraphicsRow[{Histogram[v1peak, {0., 0.005, .0001}],
  Histogram[v2peak, {0., 0.005, .0001}], Histogram[v3peak, {0., 0.05, .0005}]}]
```

Out[102]=



End of Notebook so far