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

```
SetDirectory["~seb/Desktop/PSI_data/"];
Namelist = FileNames[]
Namelist // Length;
nfiles = %
filename = Namelist[[2]]
Timing[scopedata = Import[filename, "csv"];]
```

```
{.DS_Store, LeCroy-compress-2013-06-02-045.csv,
 LeCroy-compress-2013-06-02-046.csv, LeCroy-compress-2013-06-02-047.csv,
 LeCroy-compress-2013-06-02-048.csv, LeCroy-compress-2013-06-02-049.csv}
```

```
6
```

```
LeCroy-compress-2013-06-02-045.csv
```

```
{7.061840, Null}
```

```
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.

```
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, 3, 6, 1}];
```

```
LeCroy-compress-2013-06-02-046.csv
```

```
LeCroy-compress-2013-06-02-047.csv
```

```
LeCroy-compress-2013-06-02-048.csv
```

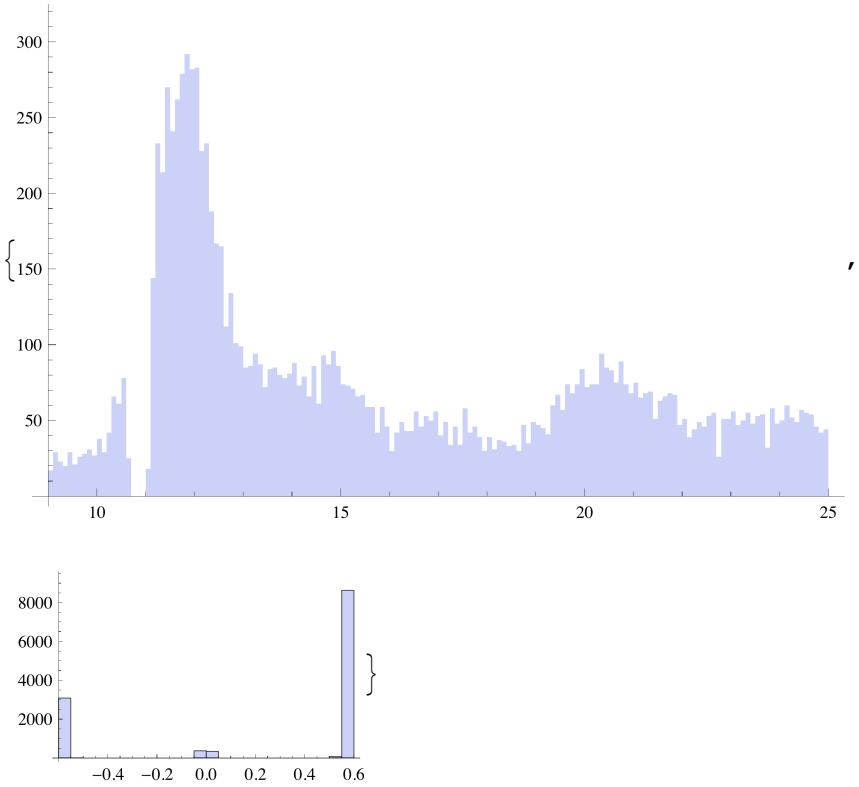
```
LeCroy-compress-2013-06-02-049.csv
```

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

```
LeCroy-ntuple-2013-06-02-049.csv
```

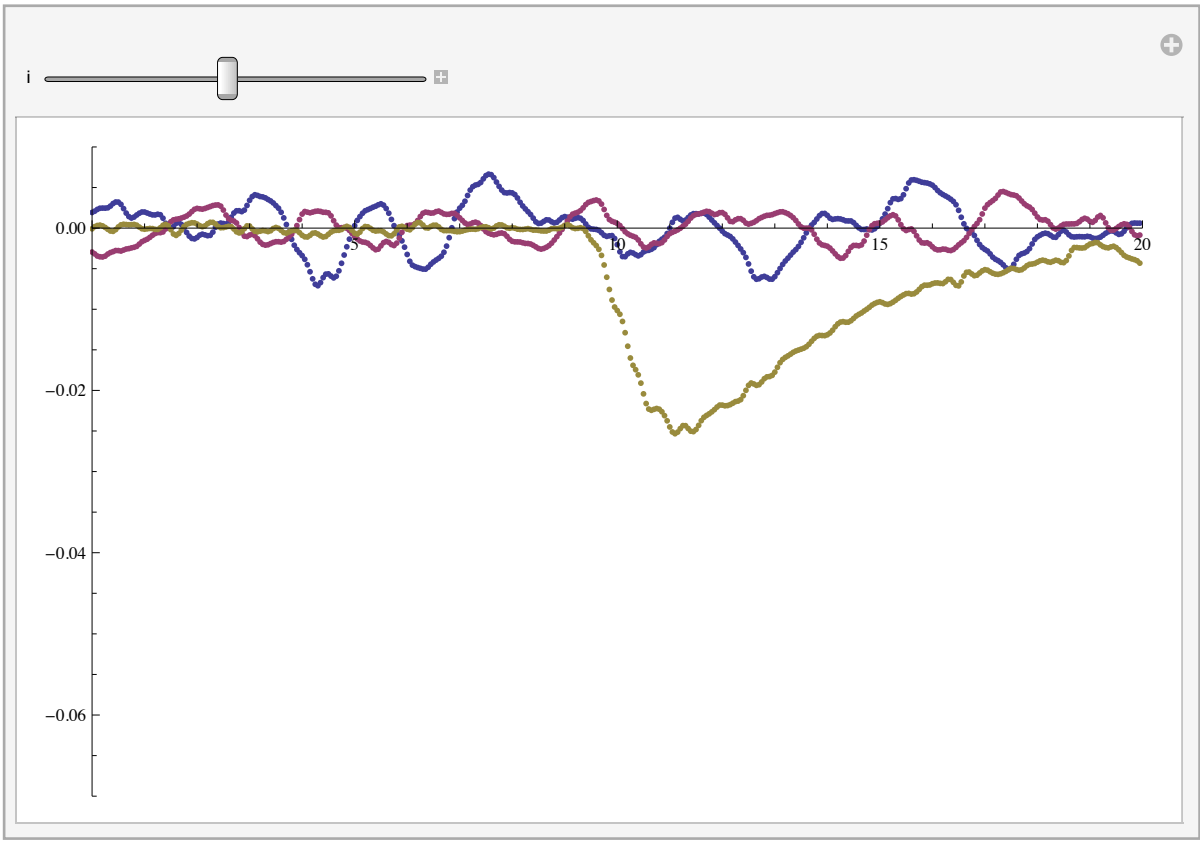
tof distributions

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



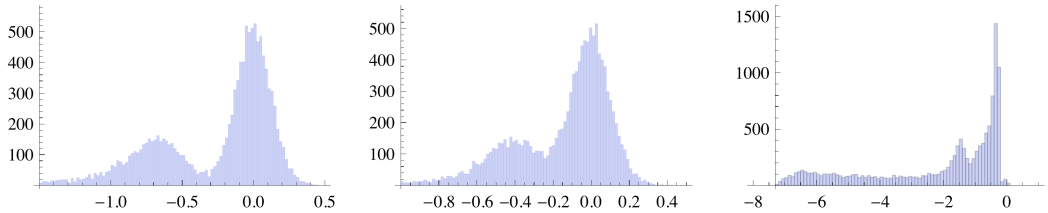
Inspect APD data.

```
time = Range[0, 399] * .05;
Manipulate[ListPlot[{Transpose[{time, v1[[i]]}], Transpose[{time, v2[[i]]}],
    Transpose[{time, v3[[i]]}]}, PlotRange → {{0, 20.}, {-.07, .01}},
  ImageSize → Large], {i, 1, 2500, 1}, SaveDefinitions → True]
```



Integrate to find pulse area.

```
ph1 = Table[Sum[v1[[i, j]], {j, 100, 299, 1}], {i, 12500}];
ph2 = Table[Sum[v2[[i, j]], {j, 100, 299, 1}], {i, 12500}];
ph3 = Table[Sum[v3[[i, j]], {j, 100, 299, 1}], {i, 12500}];

GraphicsRow[{Histogram[ph1, {-1.5, .5, .02}],
  Histogram[ph2, {-1, .5, .015}], Histogram[ph3, {-8, 1, .1}]}, ImageSize → Large]
```



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.

```
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]]
```

```
4.46684
```

```
10.
```

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

```
{{{vsum, -2.57639}}}
```

```
-1.28819
```

```
NSolve[vsum == 50 / (.05 * 10^-9) * qe * ne * APDgain * Ampgain13 * Ampgain20, vsum] /.
 Rule → List
pulsearea1 = %[[1, 1, 2]] * defecit64
```

```
{{{vsum, -25.7639}}}
```

```
-2.31875
```

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

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

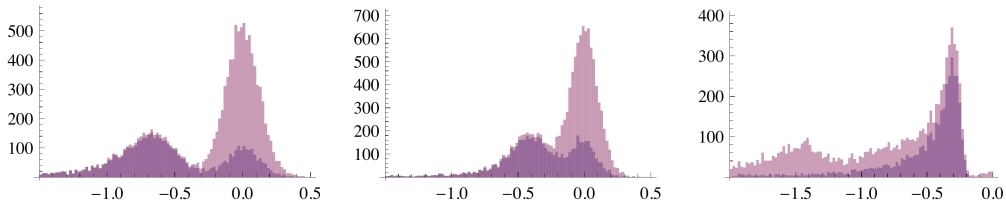
```
Export[outfile, ntuple, "csv"]
```

```
LeCroy-ntuple-2013-06-02-049.csv
```

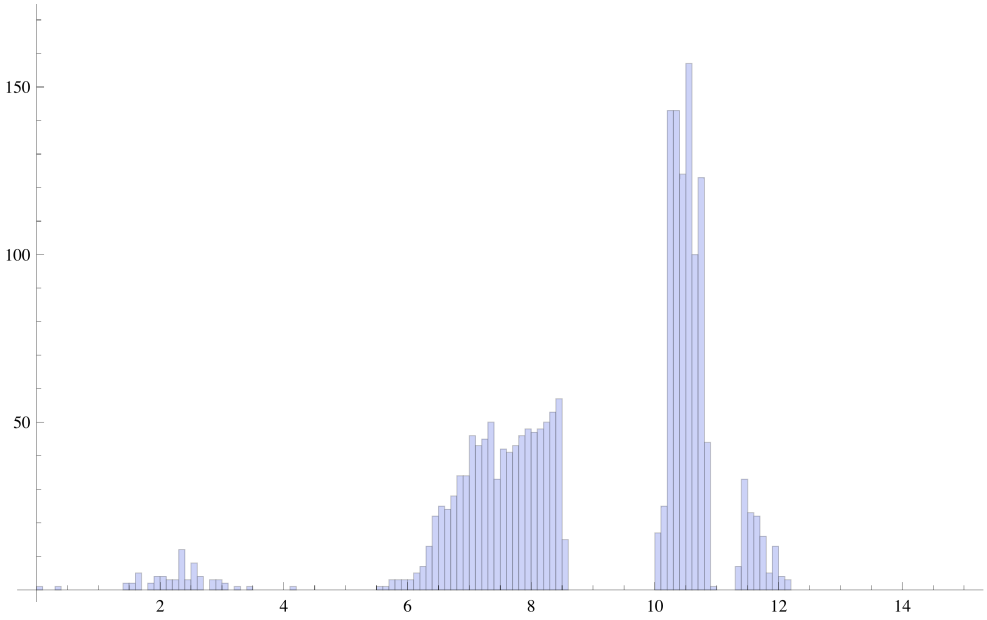
```
phasepos = 1; amplpos = 2;
ph1pos = 3; ph2pos = 4; ph3pos = 5;
ph1and2ok[datum_List] :=
 (-1 < Part[datum, ph2pos < -.2]) && (-1.3 < Part[datum, ph1pos] < -.3)
phasepi[datum_List] := (11 < Part[datum, phasepos] < 13)
ph3ok[datum_List] := (-1.2 < Part[datum, ph3pos] < -.2)
ph1ok[datum_List] := (-1.3 < Part[datum, ph1pos] < -.3)
ph2ok[datum_List] := (-1 < Part[datum, ph2pos] < -.2)
amplok[datum_List] := (.4 < Part[datum, amplpos]) &&
   (-1.3 < Part[datum, ph1pos] < -.3) && (-1.3 < Part[datum, ph1pos] < -.35)
```

Conditional Histograms

```
GraphicsRow[
 {Histogram[{Select[ntuple, ph3ok][[All, ph1pos]], ntuple[[All, ph1pos]]},
   {-1.5, .5, .02}], Histogram[{Select[ntuple, ph3ok][[All, ph2pos]],
    ntuple[[All, ph2pos]]}, {-1.5, .5, .02}], Histogram[
   {Select[ntuple, ph1ok][[All, ph3pos]], ntuple[[All, ph3pos]]}, {-2, 0, .02}]}]
```




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



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.

```
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]]
```

```
0.87314524
```

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

```
0.99999791
```

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

```
0.921113762
```

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

```
{{tofe → -11.42891}}
```

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

```
{{tofmu → -4.691885}}
```

```
Dimensions[ph1]
Dimensions[v1]
```

```
{12 500}
```

```
{12 500, 400}
```

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

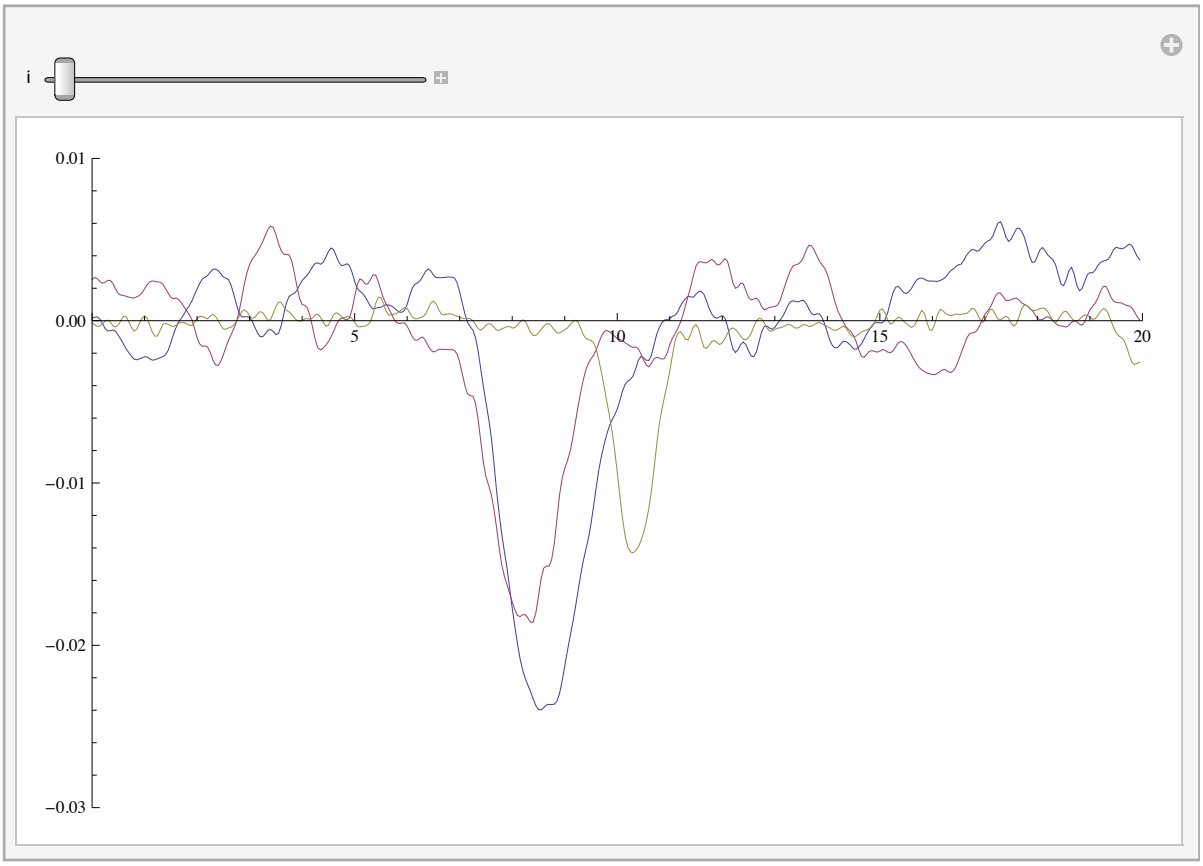
```
Clear[v1write, v2write , v3write]
ngoodevents = 0; v1write = ConstantArray[0, {2704, 400}];
v2write = ConstantArray[0, {2704, 400}]; v3write = ConstantArray[0, {2704, 400}];

Do[
  If[(-1.3 < ph1[[i]] < -.3) &&
     (-1 < ph2[[i]] < -.2) && (-1.2 < ph3[[i]] < -.2), , Goto[nogood]];
  ngoodevents++;
  v1write[[ngoodevents]] = v1[[i]];
  v2write[[ngoodevents]] = v2[[i]]; v3write[[ngoodevents]] = v3[[i]];
  Label[nogood];
  , {i, 12 500}];
ngoodevents
```

```
2704
```

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

```
LeCroy-wavecut-2013-06-02-049.csv
```

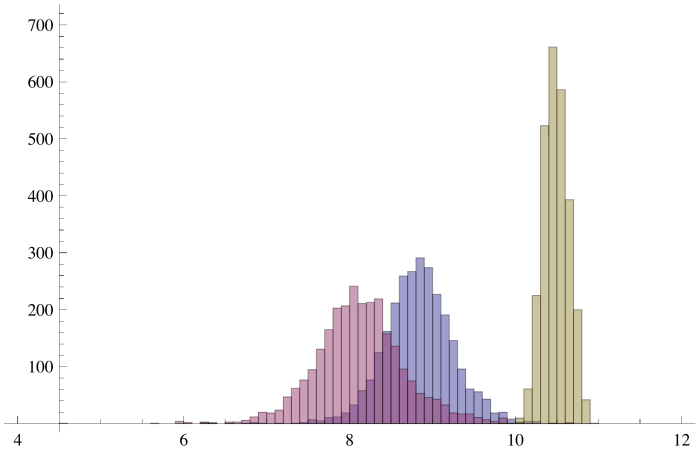
```
Manipulate[ListPlot[{Transpose[{time, v1write[[i]]}],
    Transpose[{time, v2write[[i]]}], Transpose[{time, v3write[[i]]}]},
  PlotRange → {{0, 20.}, {-.03, .01}}, Joined → True, ImageSize → Large],
 {i, 1, 2704, 1}, SaveDefinitions → True]
```



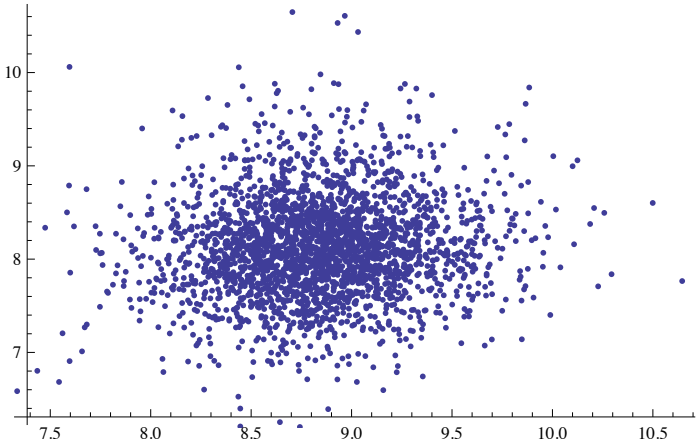
Since noise is so large in this run try timing just from the pulse centroid.

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

v1centroid = Table[(time1.Take[v1write[[i]], {100, 239}]) /
     Sum[v1write[[i, j]], {j, 100, 239}], {i, 2704}];
v2centroid = Table[(time1.Take[v2write[[i]], {100, 239}]) /
     Sum[v2write[[i, j]], {j, 100, 239}], {i, 2704}];
v3centroid = Table[(time3.Take[v3write[[i]], {180, 239}]) /
     Sum[v3write[[i, j]], {j, 180, 239}], {i, 2704}];
```

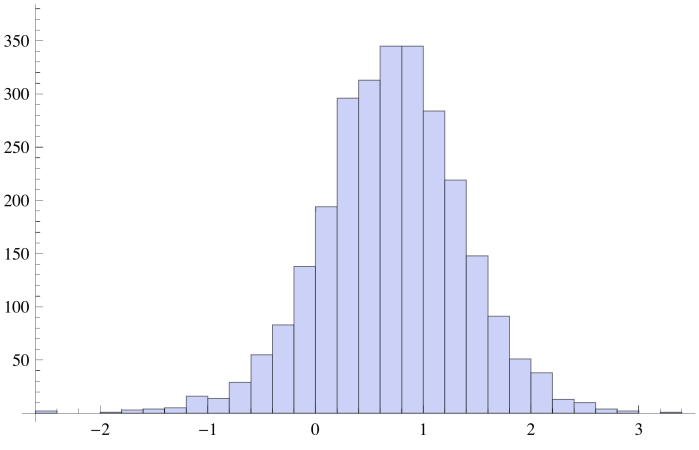
```
Histogram[{v1centroid, v2centroid, v3centroid}, {4, 12, .1}]
```




```
ListPlot[Transpose[{v1centroid, v2centroid}]]
```




```
Histogram[vdif = (v1centroid - v2centroid)]
```



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

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

```
10.4845
```

```
0.152869
```

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

```
8.83024
```

```
0.428974
```

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

```
0.657145
```

End of Notebook so far