Analysis of amplifier bench test data. Because we were having HV issues with the APD
telescope at the time, the Amp input was from a pin diode which is packaged with the
980 nm Vcsel we use for APD setup. Note that Vcsel was driven by a wide pulse (all that was
available in CERN electronics pool).
SNW-Feb. 16, 2014

First calculate expected signal amplitude for min. ionizing particle in the 8 x8 mm^2 APDs (area enters because this has large capacitance leading to a loss in peak amplitude into a 50 Ω voltage amplifier - by a bit over a factor of 10).

In[514]:=
```
ehpairs = 90; thick = 40; G_APD = 600.; t_pulse = 8 * 10^-9; Qe = 1.6 * 10^-19; R = 50.;
Q = ehpairs * thick * G_APD * Qe;
i_peak = Q / (t_pulse / 2);
v_peak = i_peak * R
```

Out[517]= 0.00432

Conclusion : We should expect an input signal with peak amplitude of about 4 mV into the preamp and should take into account a factor of about 10 reduction for the capacitance. This means that, with a 1 nsec risetime signal and a preamp gain of 100, the preamp noise at the output, in order to have less than 20 picosecond contribution to the time jitter, should be less than 1 mV- ie :

In[518]:=
```
G_AMP = 100.; SNR = 50.;
```

In[519]:=
```
Noise_Target = v_peak * G_AMP / 10. / SNR
```

Out[519]= 0.000864

In[520]:=
```
(*Quantity["ElectronCharge"]*)
```

Below we find that we are close to that, at least in the Pin diode model for our setup. This should now be confirmed with the actual telescope.

In[521]:=
```
SetDirectory["~bastian/Desktop/jan29scope/"];
Namelist = FileNames[]
Namelist // Length;
nfiles = %;
```

Out[522]=
```
{amplifierissues.docx, drs4data.dat, fritz_amp10.tif, fritz_amp11.tif,
 fritz_amp1.tif, fritz_amp2.tif, fritz_amp3.tif, fritz_amp4.tif,
 fritz_amp5.tif, fritz_amp6.tif, fritz_amp7.tif, fritz_amp8.tif,
 fritz_amp9.tif, fritz_amp.tif, LeCroy-RTW-2014-01-30-001.csv,
 LeCroy-RTW-2014-01-30-002.csv, LeCroy-RTW-2014-01-30-003.csv,
 LeCroy-RTW-2014-01-30-004.csv, LeCroy-RTW-2014-01-31-001.csv}
```

4 channels of LRS 1 GHz scope with 5 GHz sampling

In[525]:=
```
filename = Namelist[[15]]
scopedata = Import[filename, "csv"];
```

Out[525]=
```
LeCroy-RTW-2014-01-30-001.csv
```

In[527]:=
```
Dimensions[scopedata]
t0 = scopedata[[1, 2]]; dt = scopedata[[1, 3]] * 10^9;
time = Range[0, 501] * dt;
```

Out[527]=
```
{800, 505}
```

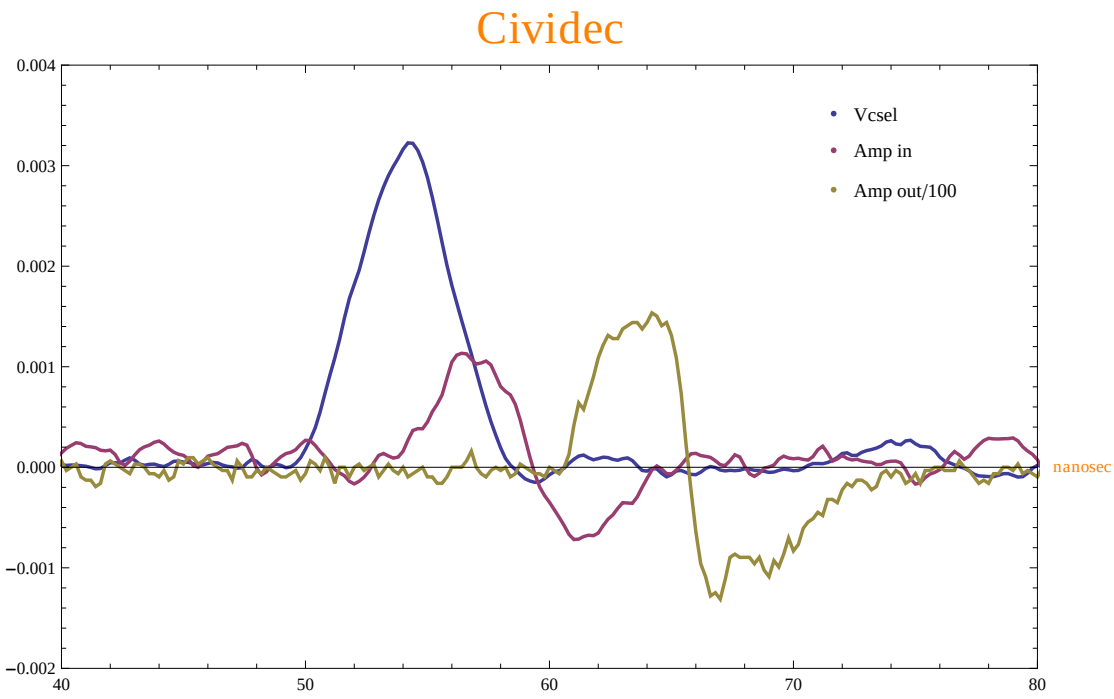
In[530]:=
```
v1 = Table[Take[scopedata[[4 * (i - 1) + 1]], {4, 505}], {i, 200}];
v2 = Table[Take[scopedata[[4 * (i - 1) + 2]], {4, 505}], {i, 200}];
v3 = Table[Take[scopedata[[4 * (i - 1) + 3]], {4, 505}], {i, 200}];
v4 = Table[Take[scopedata[[4 * (i - 1) + 4]], {4, 505}], {i, 200}];
```

‘iorder ”specifies level of filtering. ie iorder=5 reduces 5 GHz to 1 GHz average of 5 pts. Expect noise reduction from sampling to be 1/sqrt(5).

In[534]:=
```
iorder = 1;
i = 1;
```

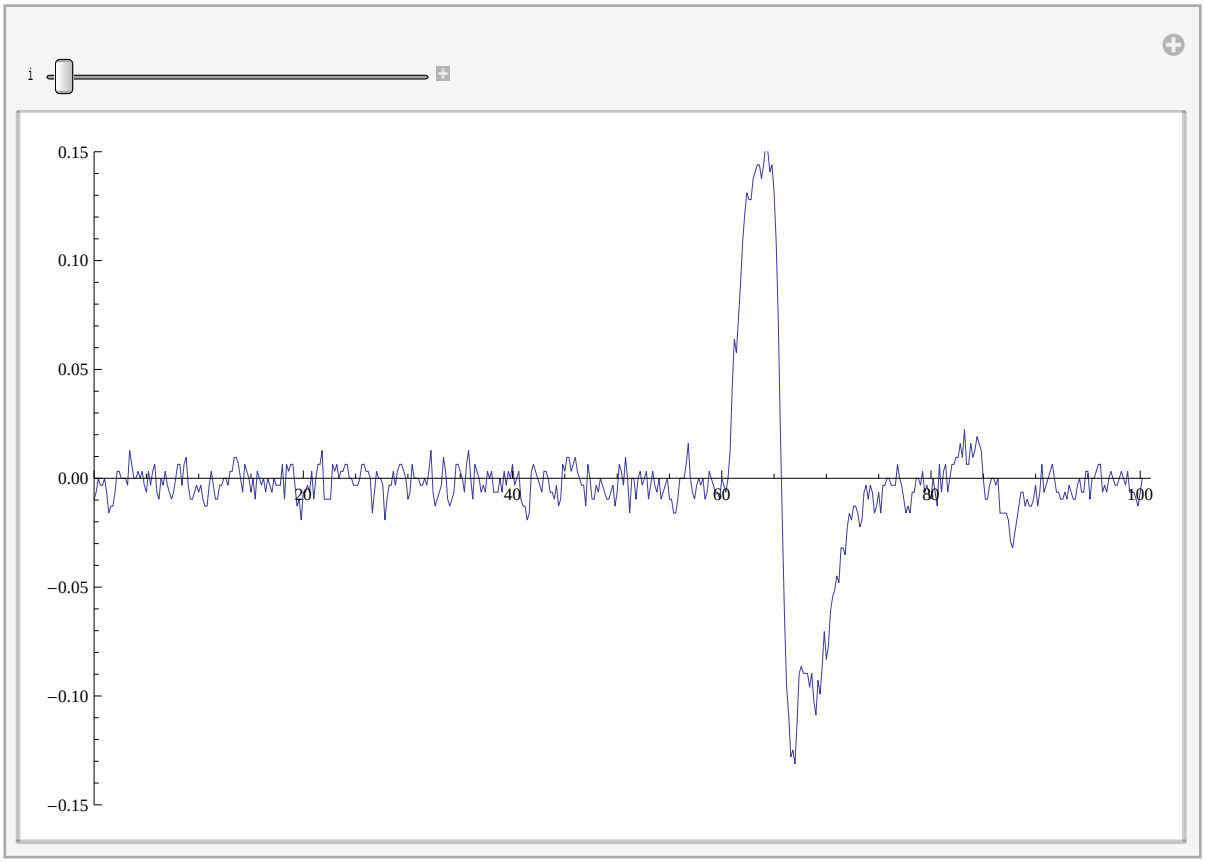
In[536]:=
```
ListPlot[{Transpose[
   {MovingAverage[time, iorder], MovingAverage[v2[[i]] / 500., iorder]}],
  Transpose[{MovingAverage[time, iorder],
    MovingAverage[(+.0025 + v3[[i]]), iorder]}], Transpose[
   {MovingAverage[time, iorder], MovingAverage[v4[[i]] / 100., iorder]}]},
 PlotStyle → Directive[AbsoluteThickness[1.8]],
 LabelStyle → Directive[Orange, Bold],
 PlotLabel → Style[Cividec, Large],
 AxesLabel → {nanosec,}, PlotStyle → {Red, Blue, Green},
 PlotLegends → Placed[{"Vcsel", "Amp in", "Amp out/100"}, {0.85, 0.85}],
 PlotRange → {{40, 80.}, {-.002, .004}}, Joined → True,
 ImageSize → Large, Frame → True, FrameStyle → Black]
```

Out[536]=



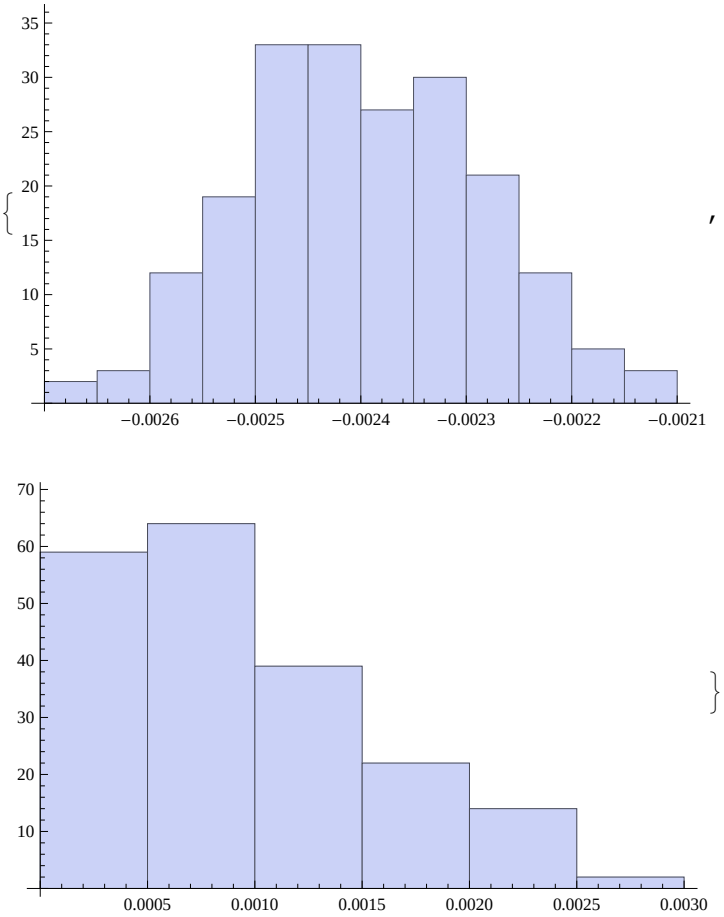
In[537]:=
```
v4save = v4[[1]];
iorder = 1;
Manipulate[ListPlot[
  Transpose[{MovingAverage[time, iorder], MovingAverage[v4save, iorder]}],
  PlotRange → {{0, 101.}, {-.15, .15}}, Joined → True, ImageSize → Large],
 {i, 1, 200, 1}, SaveDefinitions → True]
```

Out[539]=



In[540]:=
```
inbase = Table[Mean[Take[v3[[i]], 300]], {i, 200}];
innoise = Table[RootMeanSquare[Take[v3[[i]], 300] - outbase[[i]]], {i, 200}];
{Histogram[inbase, ImageSize → Medium], Histogram[innoise, ImageSize → Medium]}
Print[Style["    volts=  Average Input Noise" Mean[innoise] , Large, Red]]
```

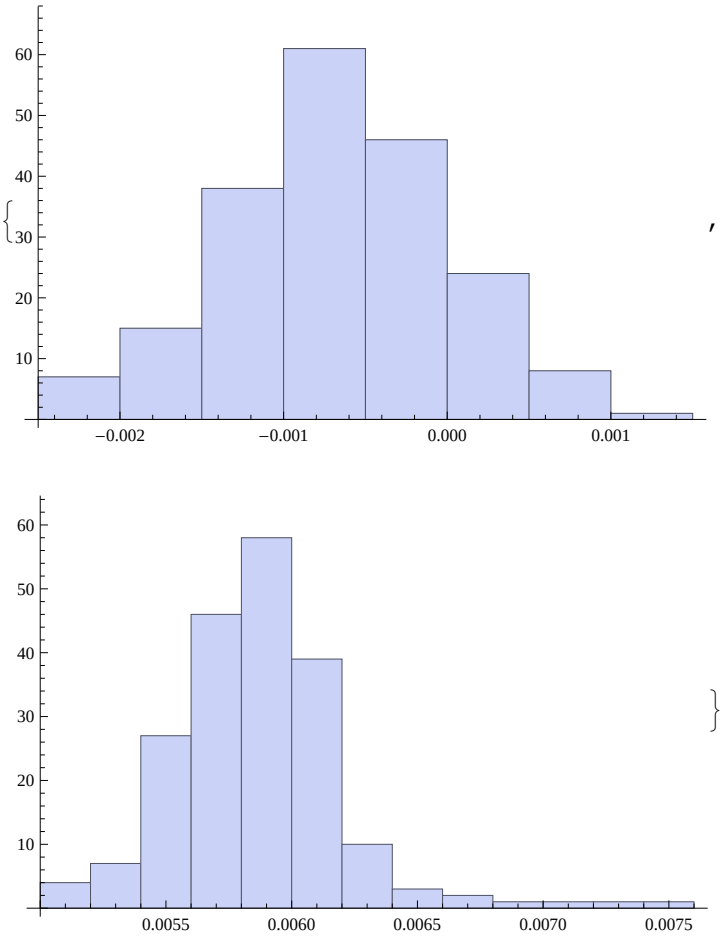
Out[542]=



0.000953353   volts= Average Input Noise

In[544]:=
```
outbase = Table[Mean[Take[v4[[i]], 300]], {i, 200}];
outnoise = Table[RootMeanSquare[Take[v4[[i]], 300] - outbase[[i]]], {i, 200}];
{Histogram[outbase, ImageSize → Medium], Histogram[outnoise, ImageSize → Medium]}
Print[Style["    volts=  Average Output Noise" Mean[outnoise] , Large, Red]]
```

Out[546]=



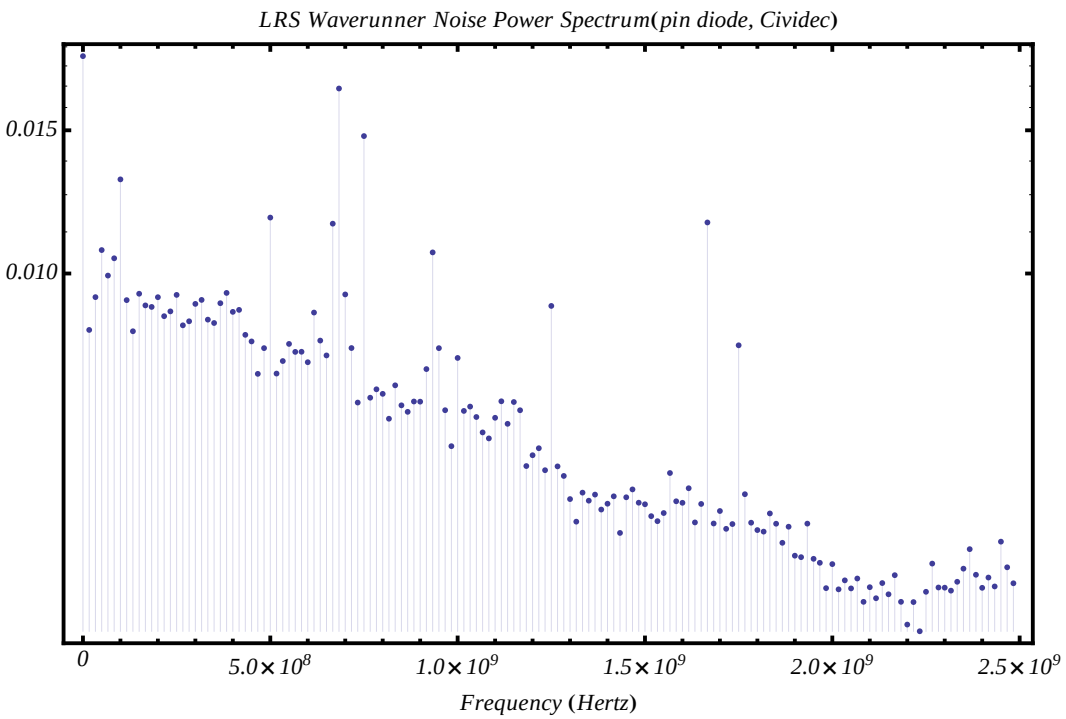
0.00586694 volts= Average Output Noise

Do a Fourier analysis of scope noise.

In[548]:=
```
samplFreq = 1 / dt * 10^9; fspi = (samplFreq / 2) / 150;
fftbins = fspi * Range[0, (150 - 1)];
```

In[550]:=
```
fftave = Sum[Take[Abs[Fourier[Take[v4[[i]], 300]]], 150], {i, 200}] / 150.;
ListLogPlot[Transpose[{fftbins, fftave}], Filling → Axis, Frame → True,
 FrameStyle → Thick, FrameLabel → {{, }, {"Frequency (Hertz)",
    "LRS Waverunner Noise Power Spectrum(pin diode, Cividec)"}},
 LabelStyle → {Medium, Italic, Bold}, ImageSize → Large]
```

Out[550]=



Now repeat for data with preamp input removed.

In[551]:=
```
filename = Namelist[[16]]
Clear[scopedata];
scopedata = Import[filename, "csv"];
```

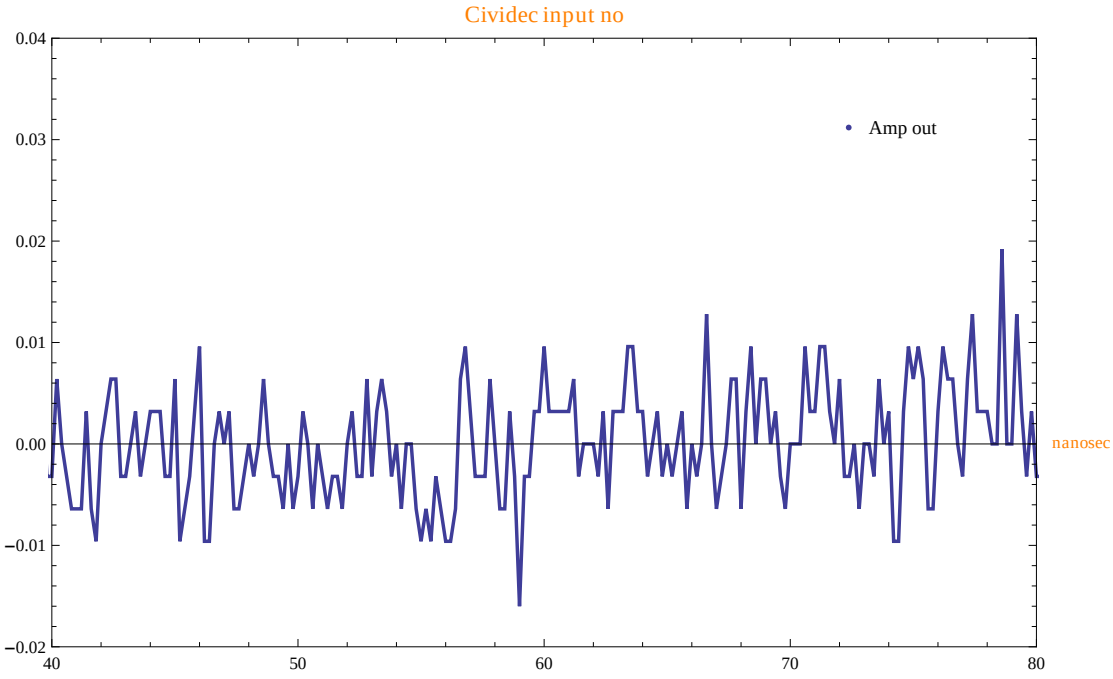
Out[551]= LeCroy-RTW-2014-01-30-002.csv

In[554]:=
```
Dimensions[scopedata]
t0 = scopedata[[1, 2]]; dt = scopedata[[1, 3]] * 10^9;
time = Range[0, 501] * dt;
v4 = Table[Take[scopedata[[4 * (i - 1) + 4]], {4, 505}], {i, 200}];
```

Out[554]= {800, 505}

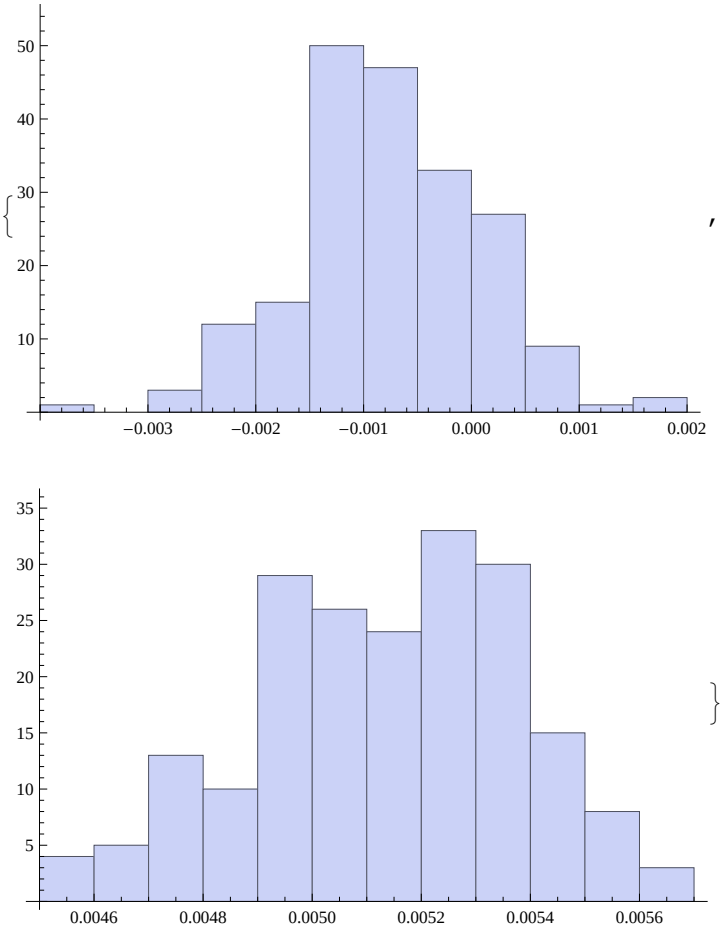
In[558]:=
```
i = 1;
ListPlot[Transpose[{time, v4[[i]]}],
 PlotStyle → Directive[AbsoluteThickness[1.8]],
 LabelStyle → Directive[Orange, Bold], PlotLabel → Cividec no input,
 AxesLabel → {nanosec,}, PlotStyle → {Green},
 PlotLegends → Placed[{"Amp out"}, {0.85, 0.85}],
 PlotRange → {{40, 80.}, {-.02, .04}}, Joined → True,
 ImageSize → Large, Frame → True, FrameStyle → Black]
```

Out[559]=



In[560]:=
```
outbase = Table[Mean[Take[v4[[i]], 300]], {i, 200}];
outnoise = Table[RootMeanSquare[Take[v4[[i]], 300] - outbase[[i]]], {i, 200}];
{Histogram[outbase, ImageSize → Medium], Histogram[outnoise, ImageSize → Medium]}
Print[Style["    volts=  Average Output Noise" Mean[outnoise] , Large, Red]]
```

Out[562]=



0.00513468   volts= Average Output Noise