```
SetDirectory[""/Users/seb/Desktop/PSI_June
   _13/June2/in_front/HV1814/cascade/nolucite/""];
Namelist = FileNames[];
Namelist // Length;
nfiles = %
```

In[1]:=
```
filename =
 "/Users/seb/Desktop/PSI_June_13/June2/in_front/HV1814/cascade/nolucite/LeCroy-
   SGM-2013-06-02-045.csv"
```

Out[1]=
```
/Users/seb/Desktop/PSI_June_13/June2/in_front/HV1814/cascade/nolucite/LeCroy-
  SGM-2013-06-02-045.csv
```

In[2]:=
```
Print [filename]
```

```
/Users/seb/Desktop/PSI_June_13/June2/in_front/HV1814/cascade/nolucite/LeCroy-SGM-2013
  -06-02-045.csv
```

In[3]:=
```
Timing[scopedata = Import[filename, "csv"];]
```

Out[3]=
```
{74.895624, Null}
```

In[4]:=
```
Dimensions[scopedata]
```

Out[4]=
```
{10000, 4005}
```

# Name the output file

In[5]:=
```
outfile = StringReplace[filename, {"SGM" → "compress"}]
```

Out[5]=
```
/Users/seb/Desktop/PSI_June_13/June2/in_front/HV1814/cascade/nolucite/LeCroy-
  compress-2013-06-02-045.csv
```

# Pick 1 event for displaying.

In[6]:=
```
eventno = 31;
index = (eventno - 1) * 4;
firstrec = scopedata[[31]]; secndrec = scopedata[[32]];
thrdrec = scopedata[[33]]; frthrec = scopedata[[34]];
```

In[9]:=
```
filestruct = Dimensions[scopedata];
records = Dimensions[scopedata][[1]];
nevents = Dimensions[scopedata][[1]] / 4;
reclen = Dimensions[scopedata][[2]];
Print[{"There are   ", nevents,
  " events and the data dimensionality is  ", filestruct}]
```

```
{There are   , 2500,  events and the data dimensionality is  , {10000, 4005}}
```

# Channel 1 and Channel 2 are 64 mm^2 APDs. Channel 3 is 4 mm^2 APD (negative).

In[14]:=
```
Print[{"channels are read in the following order:",
  scopedata[[1, 1]], scopedata[[2, 1]], scopedata[[3, 1]],
  scopedata[[4, 1]], scopedata[[5, 1]], scopedata[[6, 1]], ",etc"}]
```

```
{channels are read in the following order:, 1., 1., 1., 1., 1., 1., ,etc}
```

# Set up to plot scope traces for a particular event (eventno).

In[15]:=
```
chan1 = firstrec[[1]]; chan2 = secndrec[[1]];
chan3 = thrdrec[[1]]; chan4 = frthrec[[1]];
dt = firstrec[[3]] * 10^9
```

Out[16]=
```
0.05
```

In[17]:=
```
tstart = 0;
v3rec = scopedata[[21]];
v1 = Drop[firstrec, 3]; v2 = Drop[secndrec, 3];
v3 = Drop[thrdrec, 3]; v4 = Drop[frthrec, 3];
npts = Dimensions[v1][[1]]
Print[Style[{" 4 Channels read out in ", dt, " nanosecond bins from ",
   tstart, " to ", tstart + npts * dt " nanoseconds "}, 18, Red]]
```

Out[20]=
```
4002
```

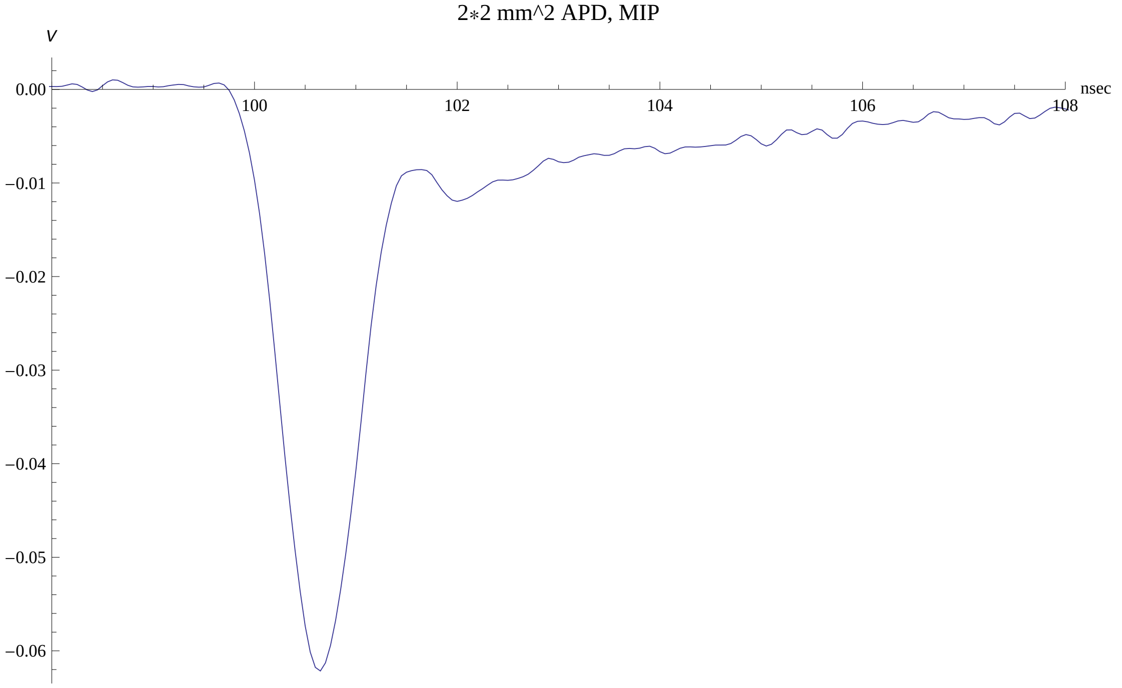
```
{ 4 Channels read out in , 0.05,
  nanosecond bins from , 0,  to , 200.1  nanoseconds }
```

In[22]:=
```
time = Table[(i - 1) * dt, {i, npts}];
```

Look at typical signal on small APD. Not bad!

In[23]:=
```
v3rec = scopedata[[22]]; v5 = Drop[v3rec, 3];
ListPlot[Transpose[{time, v5}],
 Joined → True, ImageSize → Large, AxesLabel → {nsec, V},
 PlotRange → {{98, 108}, Full}, PlotLabel → "2*2 mm^2 APD, MIP"]
```

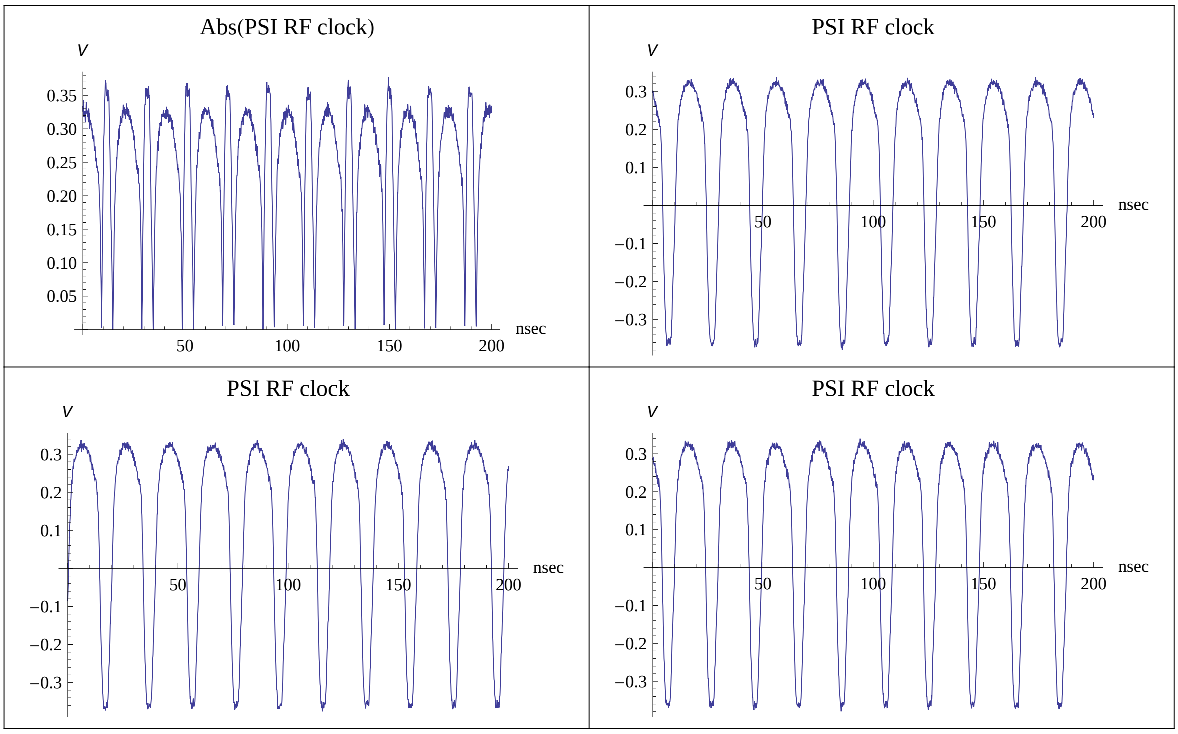
Out[24]=




Now examine Accelerator Clock data on Channel 4. Sin fit looks as good as any tool for phase info.

```
GraphicsGrid[{ListPlot[Transpose[{time, Abs[v1 + .4]}], Joined → True,
   AxesLabel → {nsec, V}, PlotRange → Full, PlotLabel → "Abs(PSI RF clock)"],
  ListPlot[Transpose[{time, v2 + .4}], Joined → True, AxesLabel → {nsec, V},
   PlotRange -> Full, PlotLabel → "PSI RF clock"]
 }, Frame → All, ImageSize → Full]
```

Out[25]=



Find Period of Accelerator Clock

In[26]:=
```
f = Interpolation[Transpose[{time, v1 + .4}], InterpolationOrder → 1];
lech2 = Table[
   x /. FindRoot[f[x], {x, n}, WorkingPrecision → 5]
   , {n, 30, 90, 20}];
Mean[Differences[lech2]]
```

Out[28]= 19.75

Fit to sin wave of form
V(t)=A0*Sin[2*Pi*(t-b)/tau]
to find the phase, b

In[29]:=
```
model = a Sin[.318 (t - b)];
Clear[trace]; Clear[tr1]; trace = ConstantArray[0, 4005];
trace = scopedata[[31]];
tr1 = Drop[trace, 3];
```

In[33]:=
```
fit = FindFit[Transpose[{time, tr1 + .4}], model, {a, b}, t]
fitpar = % /. Rule → List
phase = fitpar[[2, 2]]
```

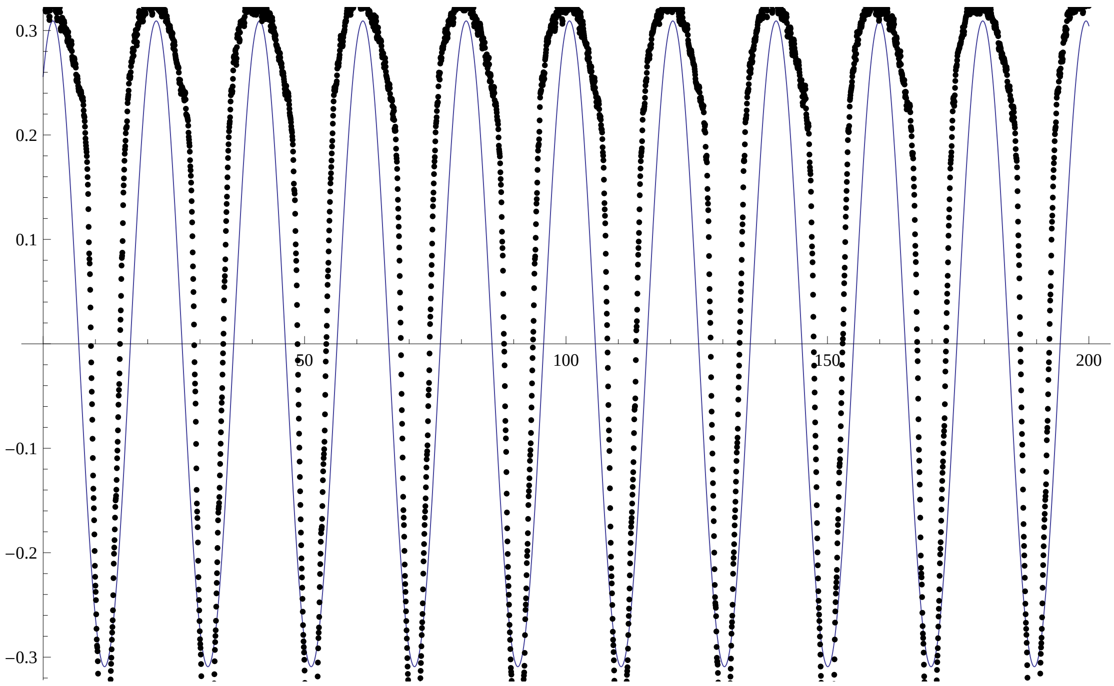
Out[33]= {a → 0.309165, b → -3.06768}

Out[34]= {{a, 0.309165}, {b, -3.06768}}

Out[35]= -3.06768

In[36]:=
```
modelf = Function[{t}, Evaluate[model /. fit]];
Plot[modelf[t], {t, 0, 200}, ImageSize → Large,
 Epilog → Map[Point, Transpose[{time, tr1 + .4}]]]
```

Out[37]=

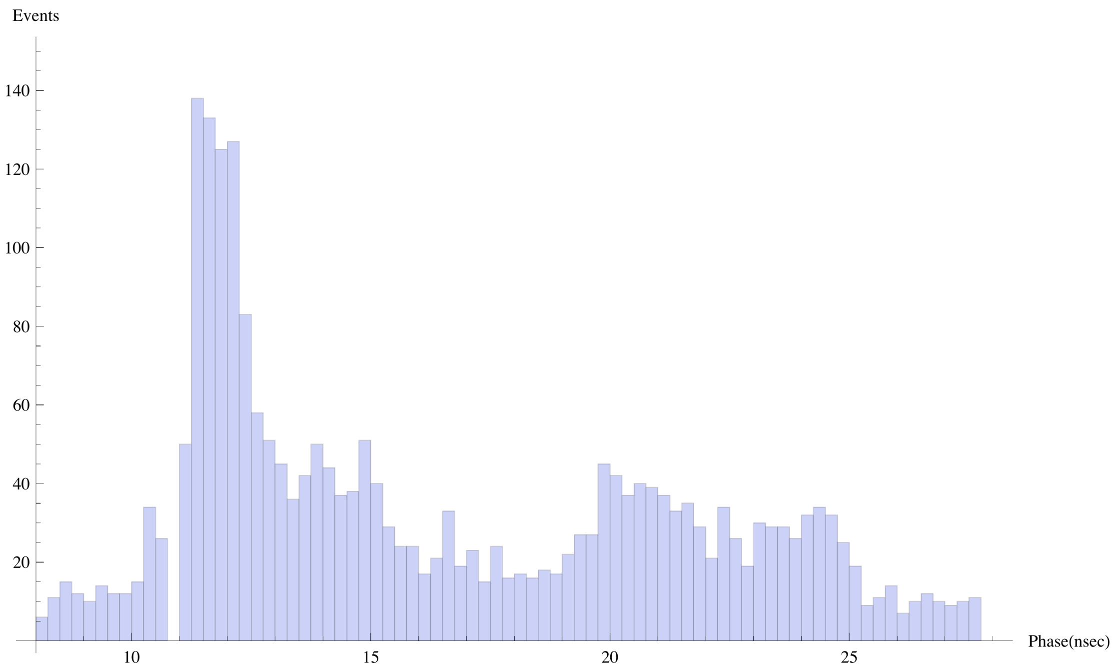


## Decoding for Segmented mode data

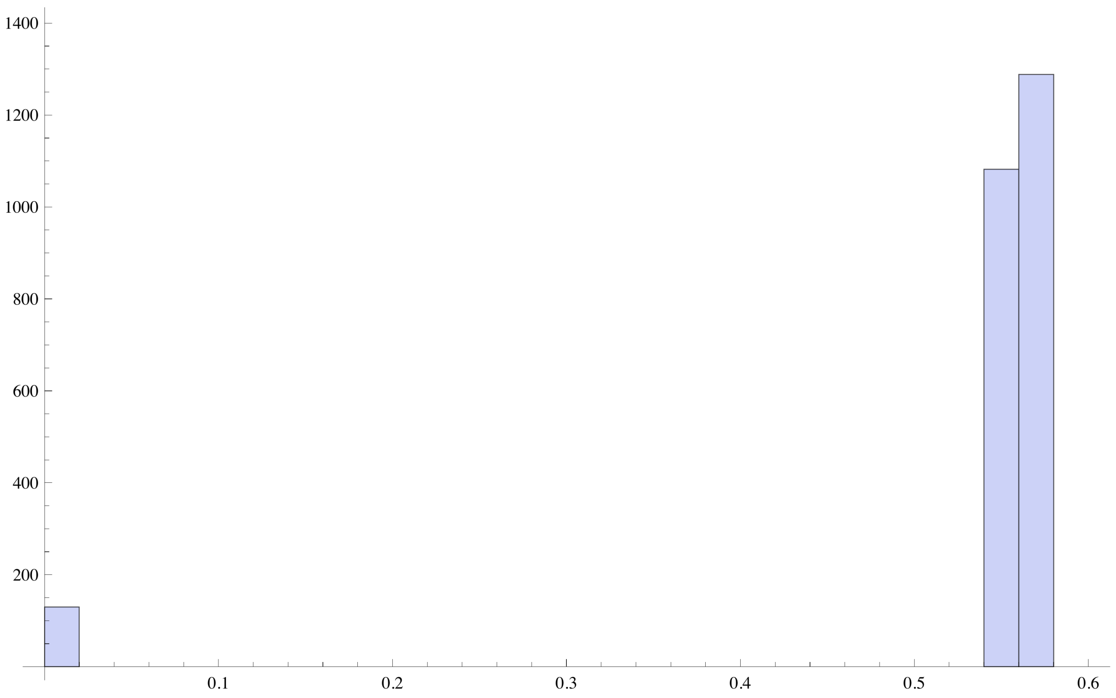
In[38]:=
```
nsegment = 10; nloop = 25; nchan = 4; ichan = 4;
Clear[v]; v = ConstantArray[0, {2500, 4002}];
(*Do[cannel index-   but for moment only channel 4*)
Do[(*loop index*)
  Do[(*segemt index*)
    nev = iseg + 10 * (iloop - 1);
    trace = scopedata[[10 * (ichan - 1) + iseg + 40 * (iloop - 1)]];
    v[[nev]] = Drop[trace, 3];
    , {iseg, 1, 10, 1}];
  , {iloop, 1, 250, 1}];

model = a * a Sin[.318 (t - b)];
fitphase = Table[FindFit[Transpose[{time, (v[[iev]] + .4)}], model, {a, b}, t] /.
    Rule → List, {iev, 1, 2500}];
```

In[77]:=
```
phase = Table[fitphase[[i, 2, 2]], {i, 2500}];
phaseshift = Mod[phase, 19.75, 8.];
ampl = Table[fitphase[[i, 1, 2]], {i, 2500}];
Histogram[phaseshift, {8, 28, .25},
 AxesLabel → {"Phase(nsec)", "Events"}, ImageSize → Large]
Histogram[Abs[ampl], {0, .6, .02}]
```

Out[80]=




Out[81]=




Estimate Distance scale from spread in Time of Flight

In[82]:=

mass of muon »

```
ParticleData["Muon", "Mass"]
```

```
105.658369`9.
```

```
In[83]:= mu = %
Out[83]= 105.658369

In[84]:= mass of pion »
         ParticleData["PiPlus", "Mass"]
Out[84]= 139.57018

In[85]:= mpi = %
Out[85]= 139.57018

In[89]:= mass of electron »
         Result
         510.9989 keV/c^2
Out[89]= 510.999 keV/c^2

In[90]:= me = %[[1]] / 1000
Out[90]= 0.510999

In[96]:= Solve[250 == mpi * beta / Sqrt[1 - beta * beta], beta] /. Rule → List
Out[96]= {{{beta, 0.87314524}}}

In[97]:= vpi = %[[1, 1, 2]]
Out[97]= 0.87314524

In[98]:= Solve[250 == me * beta / Sqrt[1 - beta * beta], beta] /. Rule → List
         ve = %[[1, 1, 2]]
Out[98]= {{{beta, 0.999998}}}
Out[99]= 0.999998

In[101]:= Solve[250 == mu * beta / Sqrt[1 - beta * beta], beta] /. Rule → List
          vmu = %[[1, 1, 2]]
Out[101]= {{{beta, 0.921113762}}}
Out[102]= 0.921113762

In[106]:= Solve[8 == L / 30 * (1 / ve - 1 / vpi), L]
Out[106]= {{L → -1651.95}}
```

So to get a time difference between pions and electrons of order 8 nsec it takes a distance from the production target of about 16.5 meters.
Is this about right?

```
In[254]:= timefun[tbin_] := (tbin - 1) * dt
          bin[t_] := (t) / dt + 1
```

In[259]:=
```
timefun[1940]
```

Out[259]= 96.95

# Now examine APD channels
# small APD (Scope input#3)

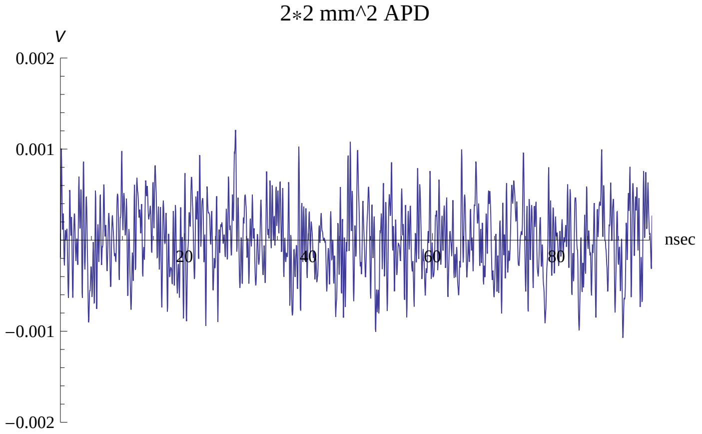
In[107]:=
```
nsegment = 10; nloop = 25; nchan = 4; ichan = 3;
Clear[v]; v = ConstantArray[0, {2500, 4002}];
(*Do[cannel index-   but for moment only channel 4*)
Do[(*loop index*)
  Do[(*segemt index*)
    nev = iseg + 10 * (iloop - 1);
    trace = scopedata[[10 * (ichan - 1) + iseg + 40 * (iloop - 1)]];
    v[[nev]] = Drop[trace, 3];
    , {iseg, 1, 10, 1}];
  , {iloop, 1, 1, 1}];

offset3 = Table[Mean[Take[v[[i]], 1980]], {i, 1, 10, 1}];
v3 = Table[(v[[i]] - offset3[[i]]), {i, 1, 10, 1}];
noise3 = Table[RootMeanSquare[Take[v[[i]], 1980]], {i, 1, 10, 1}];
Print[" Average offset=", 1000 * Mean[offset3],
 " mV and average noise= ", 1000 * Mean[noise3], " mV"]
```

Out[130]= {0.000388008, 0.000397334, 0.00048456, 0.000394543, 0.000461283,
0.000397423, 0.000414631, 0.000521328, 0.000371244, 0.000486995}

Out[155]= {0.000557142, 0.000580439, 0.000785259, 0.000562952, 0.000609128,
0.000696421, 0.00056338, 0.000659155, 0.000534598, 0.000618936}

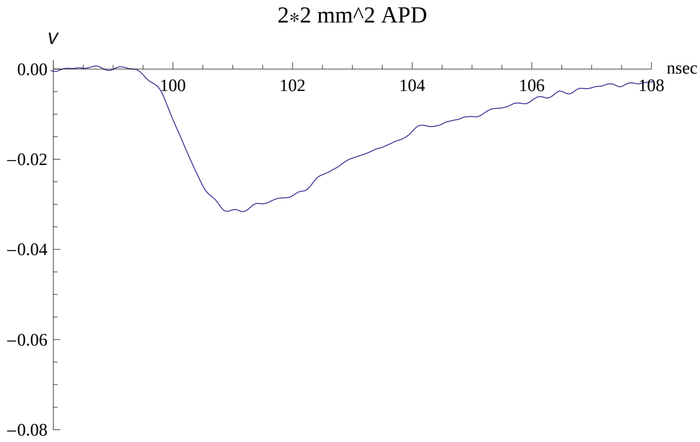
In[157]:=
```
ListPlot[Transpose[{time, v3[[10]]}], Joined → True, AxesLabel → {nsec, V},
 PlotRange → {{0, 95}, {-.002, .002}}, PlotLabel → "2*2 mm^2 APD"]
```

Out[157]=



In[159]:=
```
ListPlot[Transpose[{time, v3[[10]]}], Joined → True, AxesLabel → {nsec, V},
 PlotRange → {{98, 108}, {-.08, .002}}, PlotLabel → "2*2 mm^2 APD"]
```

Out[159]=




## Now calculate expected peak pulse height assuming roughly triangular waveform 5 nsec wide at the base. i_peak*1/2*5nsec=Q=qe*ne*APDGain*Ampgain, V_peak=i_peak*50 Ohms

In[285]:=
```
qe = -1.60217 * 10^-19
```

Out[285]= $-1.60217\times10^{-19}$

In[286]:=
```
ne = 6000; APDgain = 600;
NSolve[13 == 20 Log10[vgain], vgain] /. Rule → List;
Ampgain = %[[1, 1, 2]]
```

Out[288]= 4.46684

In[289]:=
```
NSolve[vpeak == 50 * 1 / (.5 * 5 * 10^-9) * qe * ne * APDgain * Ampgain, vpeak]
```

Out[289]= {{vpeak → -0.0515277}}

## Large APD (Scope input channel#1)

In[160]:=
```
nsegment = 10; nloop = 25; nchan = 4; ichan = 1;
Clear[v]; v = ConstantArray[0, {2500, 4002}];
(*Do[cannel index-   but for moment only channel 4*)
Do[(*loop index*)
  Do[(*segemt index*)
    nev = iseg + 10 * (iloop - 1);
    trace = scopedata[[10 * (ichan - 1) + iseg + 40 * (iloop - 1)]];
    v[[nev]] = Drop[trace, 3];
    , {iseg, 1, 10, 1}];
  , {iloop, 1, 1, 1}];

offset1 = Table[Mean[Take[v[[i]], 1980]], {i, 1, 10, 1}]
v1 = Table[(v[[i]] - offset1[[i]]), {i, 1, 10, 1}];
noise1 = Table[RootMeanSquare[Take[v[[i]], 1980]], {i, 1, 10, 1}]
```