

The following pages contain some examples and notes on calibration and station evaluation that I have made concerning options such as CALEX available to amateur seismologists. There may be errors and or technical details missing. I have not seen much mention of these programs on the PSN but there may be some usefulness to others so I will pass on my thoughts.

First I consider calibration in general and refer you to the MSOP.

I will begin with the MSOP Chapter 4 and paragraph 4.5 specifically. This paragraph shows a ten step process to calibrate using a step response. The details are available at this link . <http://www.seismo.com/msop/msop79/inst/inst4.html#aa240>

First I will note that I find this equation for natural period confusing in the result.

$$T_s = T'_s / (1 + 0.53/\lambda^2)^{\frac{1}{2}}$$

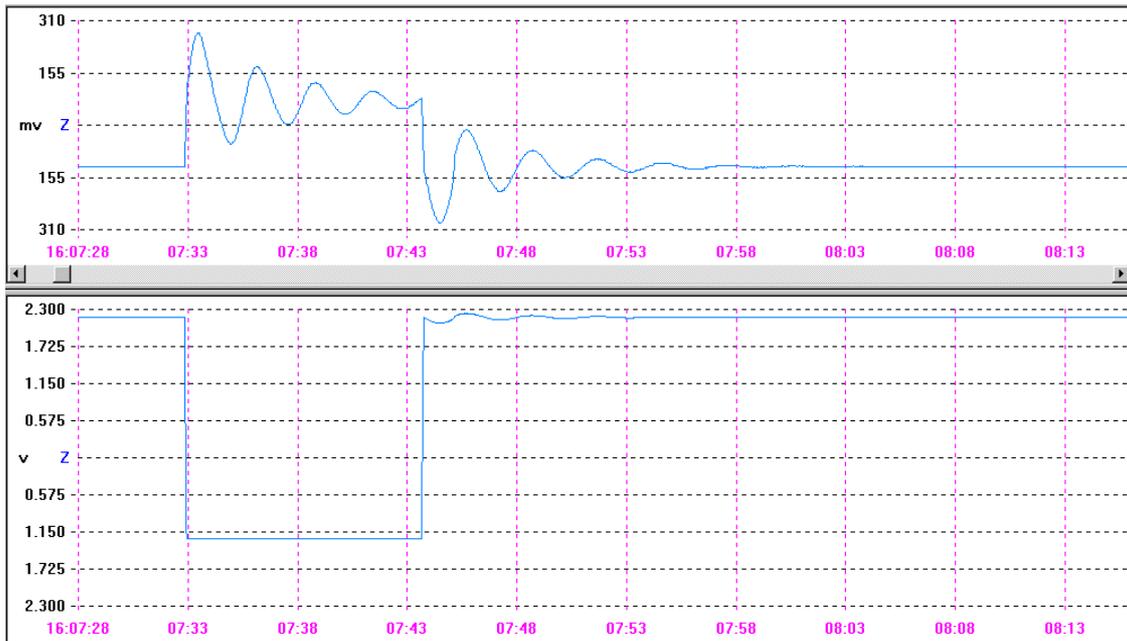
Please provide any insight you have on this subject as this equation seems to greatly underestimate the true natural period.

Otherwise, I found the process much easier to implement than attempting to make a weight lift test or by calibrating some bump input to the instrument. Using 2 AD channels to measure the impulse and response provides for time stamping of data automatically as you will see in following examples.

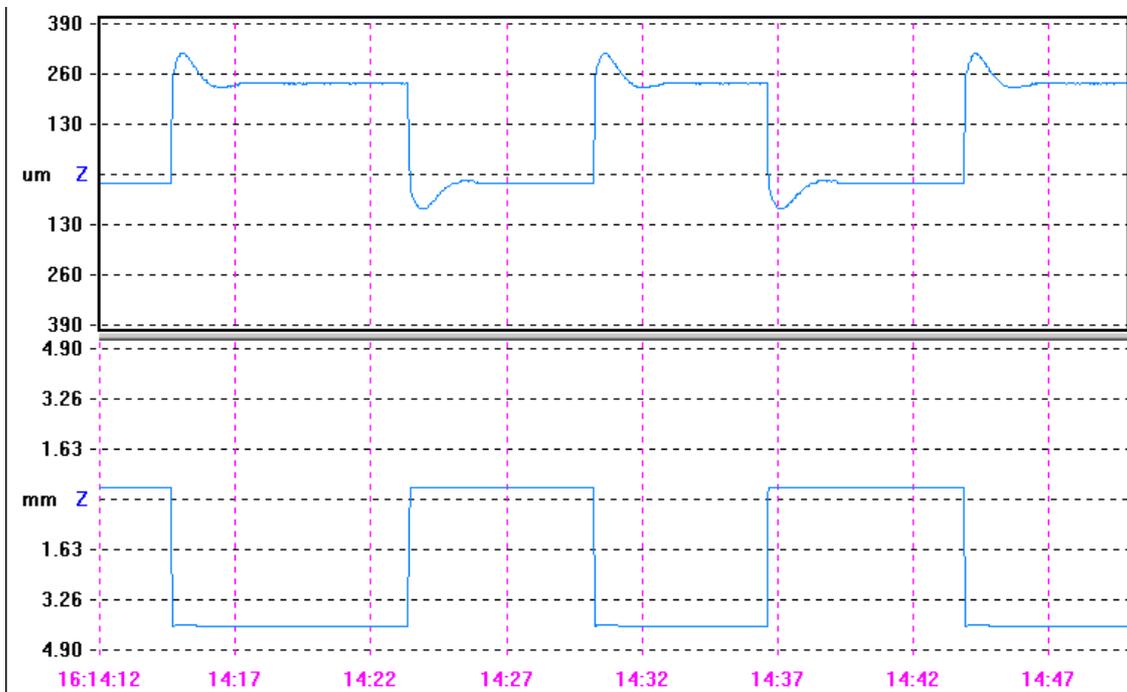
Below are some results shown from WinQuake. You will note the scale differences. I used a resistive voltage divider between the input AD channel and the instrument. This must be scaled back into the calculations that follow since the input is many times smaller than is actually recorded. For my test the voltage was divided by 100K and 500 ohms in series with the voltage across the 500 ohms going into the amp and the voltage drop across the 100K into the input data file. Thus the true gain is about 20 times higher than first appears from the displays and measurements. For the sensor without the amp I used a much larger portion of the total voltage across the sensor and also compensated for the low resistance of the coil in comparison to the divider resistance by attempting to match the divider and the coil equally and computing the resulting parallel resistance.

First is shown the under damped signal. Note the application of the current in the upper trace. The new zero level is approximately 125 on the scale and the oscillation is about this point. The first peak shows the overshoot due to the current accelerating the mass past the zero. On removal of the step current, the system oscillates about the unpowered zero point. Unlike the signal shown here you should wait longer between on and off to allow oscillation to stop so it does not affect the follow on signal. I would also point out that you will need to account for any non-zero DC in the AD signals of both traces. My off voltage is negative on the display meaning the step is larger from on to off. My output zero is also not exactly zero and again must be adjusted.

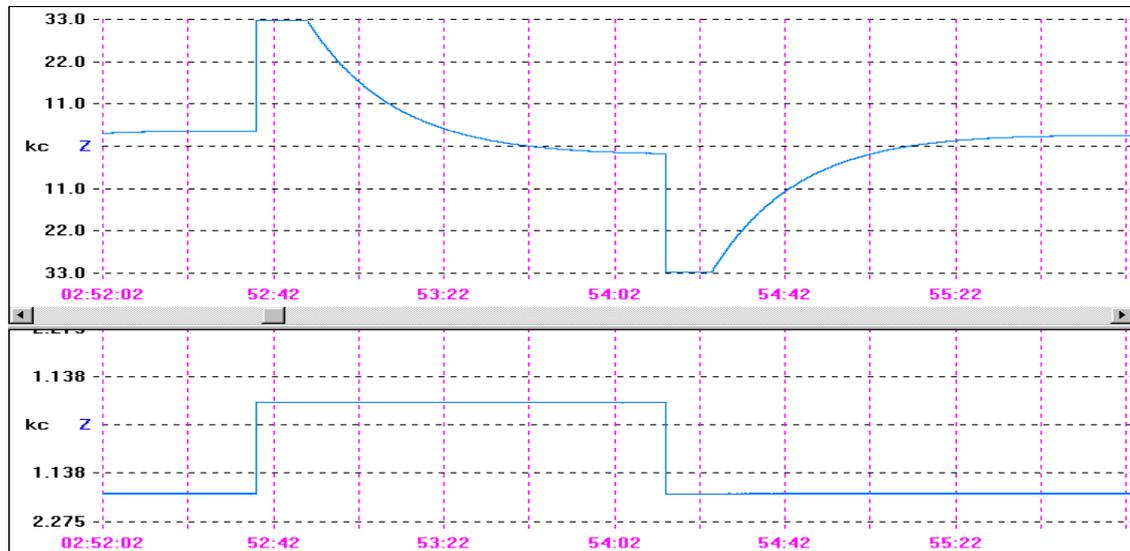
DC gain is change in V out divided by change of V in times the voltage divider ratio.



Here I show the trace of near .707 damping. Again the lower trace is the input step and the upper is the sensor output. Again note scale differences due to the voltage divider and I have not converted to voltage on this graphic. Also be aware that the voltage divider resistor is in parallel with the coil and thus has a damping effect. By comparing several different values of resistance you can determine G and the true critical damping resistance required.



The above was for a coil magnet sensor with no amp or filter attached. When we look at an amplifier filter we see some other issues with this method. Let's look at the trace of an amplifier built from the PSN schematic with no sensor attached.



In this graphic we see that the input step on the lower trace results in a decay to zero or below in the upper response. This is the effect of a high pass filter in the circuit. When testing a complete channel setup you will need to account for these effects. During damping tests in particular you will have a distorted picture of your damping if part of the decay is electronically induced by the filters in the system. Your true mechanical damping will likely be different than you see on the screen when amplifier effects are added to the decrements due to mechanical damping.

Now I will move on to a program named CALEX. You can see an overview at this link.

http://www.iris.edu/workshops/2009/metadata/lectures/11-11-09-wed_pm/Wielandt-calib-lecture/Weiland-calib1c.ppt#257,1,Title

and there is a demonstration exercise here.

http://ebooks.gfz-potsdam.de/pubman/item/escidoc:4061:3/component/escidoc:4062/EX_5.4_rev1.pdf

The software is available here. The information comes from the New MSOP.

5.8 Free Software

Source codes and DOS executables of the computer programs CALEX, DISPCAL, POLZERO, RECTAX, SINCAL, SINFIT, TILTCAL, TRIAX, TRICROSP, TWOCROSP,

and many others can be downloaded from the sites

<http://www.geophys.uni-stuttgart.de/~erhard/downloads/>

<http://www.software-for-seismometry.de/>

Some programs of tutorial character are also found there (FOURIERDEMO, FILTDEMO, SURFER).

Test data are supplied with most programs. Use the “software overview” on the website to select what you need, and read the “program descriptions” for details.

First I would say do not be overwhelmed by the description of the CALEX process. In reality all you need to do is record your random input and the resulting output signals. The program then computes your system response.

When you download the program you will get source code instructions and sample data. I compiled my version using GFortran which is available on the GNU public license. The standard data files used by CALEX are in Fortran format so there are 2 options. First, you can modify the source code to read a single column of data such as created by PSN ascii output or AmaSeis event data file created by selecting a time period from the helicorder. The second option is to create compatible data files by reformatting your data which is the method I chose. I created a C program to read a PSN format file and output five columns of data compatible with the eing, ausg formats. Documentation of the data format is included with the program. I can make my conversion program available compiled on Windows 98 but I do not know what other types of Windows it can be expected to run on. Next you will need a parameter file as below. Details are explained in the directions. My parameters were selected from earlier runs and so the program converged quickly in this sample.

Calibration of 2.8 sec vertical in a half-bridge

```
'din.txt'           ' input file
'dout.txt'          ' output file

1  alias           'Anti-alias period, 4-5 x sample int
5  m               'Num unknowns
0  m0
0  m1             'Num 1st high or low pass systems
1  m2             'num 2nd ord high low syst
2000 maxit
5e-4 qac
1e-3 finac
0  ns1
0  ns2

amp  0             10
del -0.004        0.001 'Channel del or other delay. For exact set
uncertainty to zero
sub  0.99         0.1   'Allowance for bridge
til  0.0          0.0

bp2                               'Senser
per  2.78         0.03
dmp  0.7          0.4

end
```

When the program runs it creates your system response output as follows.

Calibration of 2.8 sec vertical in a half-bridge

```
reading file dout      3930 / 72000 samples
dout.txt Number of Samples 3930 SPS: 30
3930 samples read
control parameters for the iteration:
  m  m0  m1  m2 maxit      qac      finac      alias
  5   0   0   1 2000  0.000500  0.001000  1.000000
anti-alias- filter of order 5, T= 1.00 s
writing file ausf
reading file din.      3930 / 3930 samples
din.txt Number of Samples 3930 SPS: 30
3930 samples read
writing file einf
reading start parameters:
  amp      0.000      10.000
  del      -0.004      0.001
  sub      0.990      0.100
  til      0.000      0.000
  bp2:
  per      2.780      0.030  bp2
  dmp      0.700      0.400  bp2
synthetics from start model:
writing file synt

  iter      RMS      amp      del      sub      per
dmp

    0  14.032650    0.000000  -0.004000    0.990000    2.780000
0.700000
      +  10.000000    0.001000    0.100000    0.030000
0.400000

    1  13.120801   -0.151972    0.000000   -0.017727    0.000000
0.000000
    2   8.280021   -0.162555    0.002583   -3.488080    0.027814
1.420653
   31  0.063004   -0.004858    0.002383  -10.461261   -0.058312
0.543338
   32  0.063004   -0.004857    0.002383  -10.461261   -0.058312
0.543339

final system parameters:

   32  0.063004   -0.048571   -0.003998   -0.056126    2.778251
0.917335
```

As you see I estimated my period at 2.78 seconds and damping at .7. The program determined the period is 2.778251 seconds and the damping is a bit high at .917335 after 32 iterations. Remember the bridge resistor increases the damping depending on value if it is not your exact damping resistance or it's in parallel with your damping resistor.

Now then what to do with all this information to see a picture of the system. For this I turned to the program SeismoWin.

http://mariottim.interfree.it/doc05_e.htm#seismowin

Note: This link is being replaced by

<http://www.sara.pg.it/software/seismowin/seismowin.msi>

Thank you to Mauro for noting the changed link:

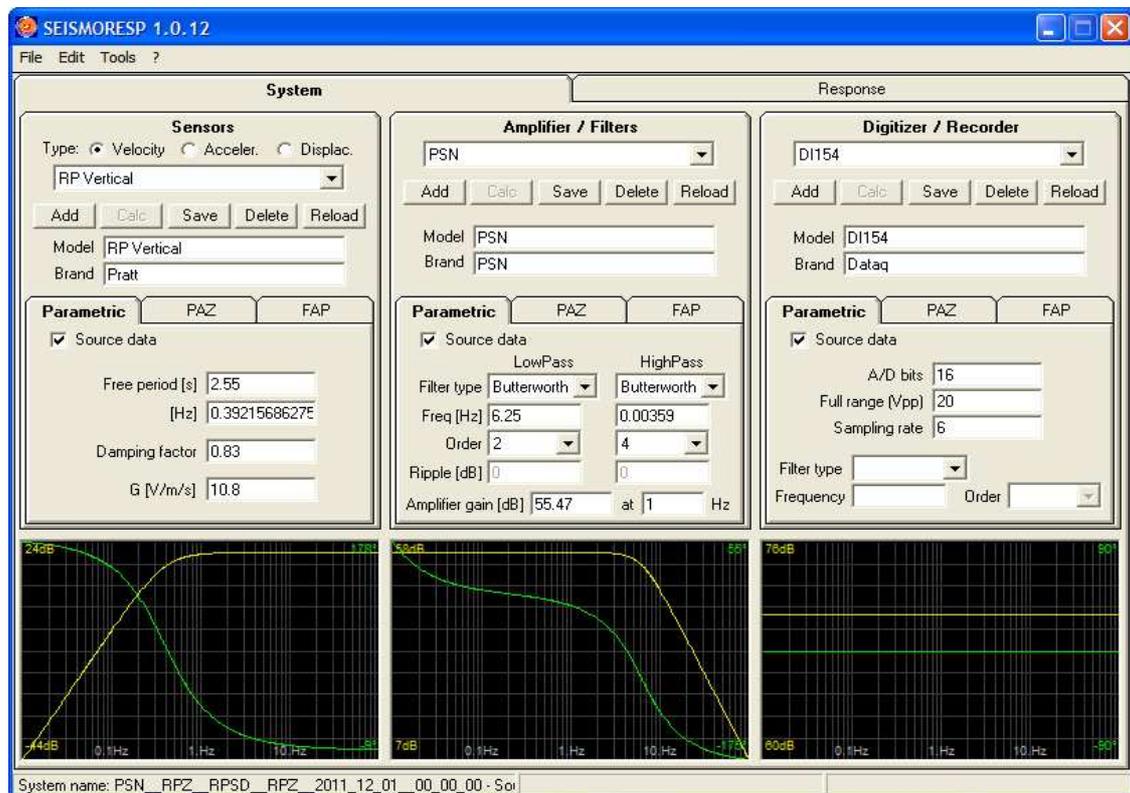
Mauro Mariotti

SARA electronic instruments s.r.l.

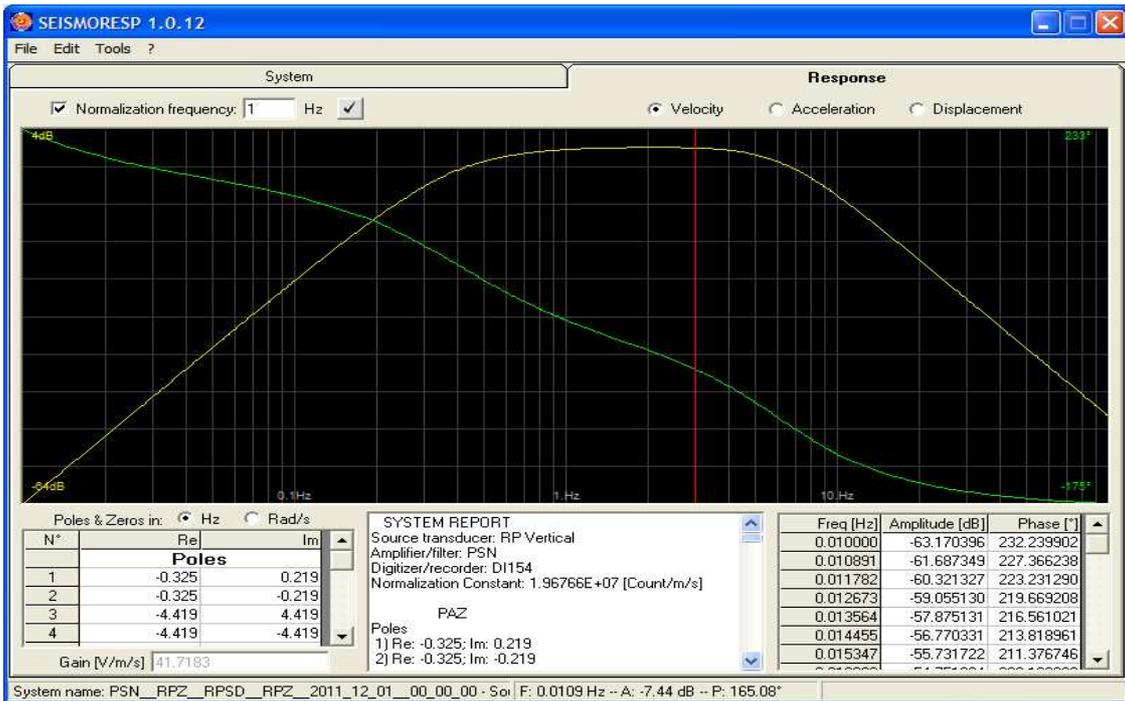
Via A.Mercuri 4 - 06129 - Perugia

Tel. +39 075 5051014 Fax +39 075 5006315

This software package has a nice system response module. Within the module you can insert your parameters and see the system response as well as create a response file for the system.

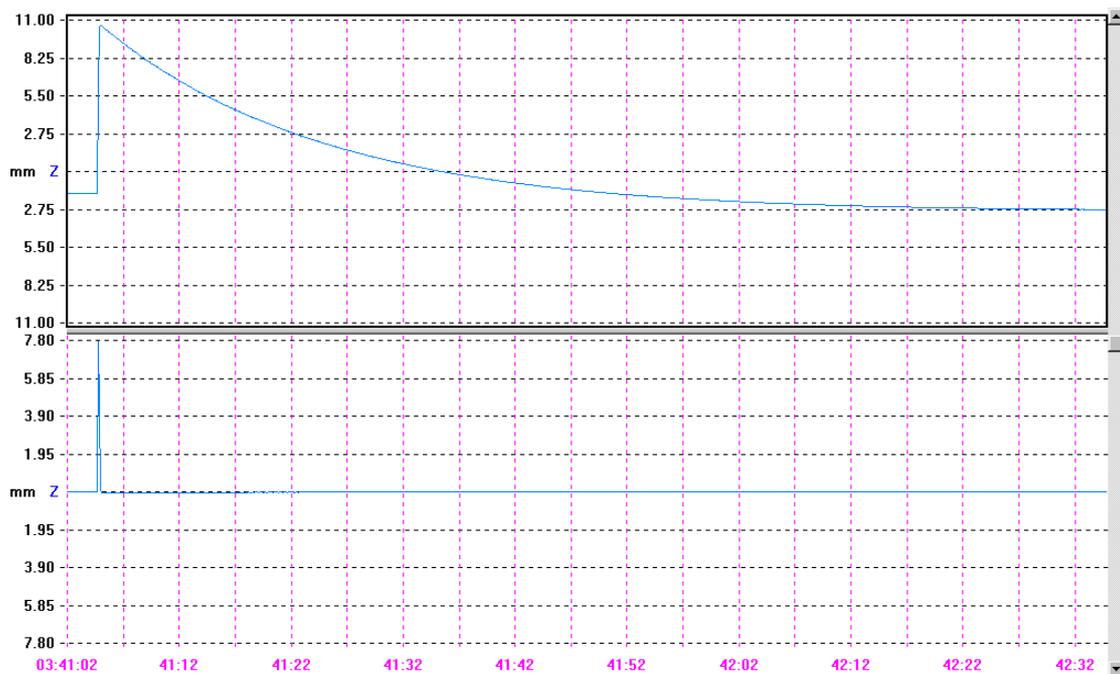


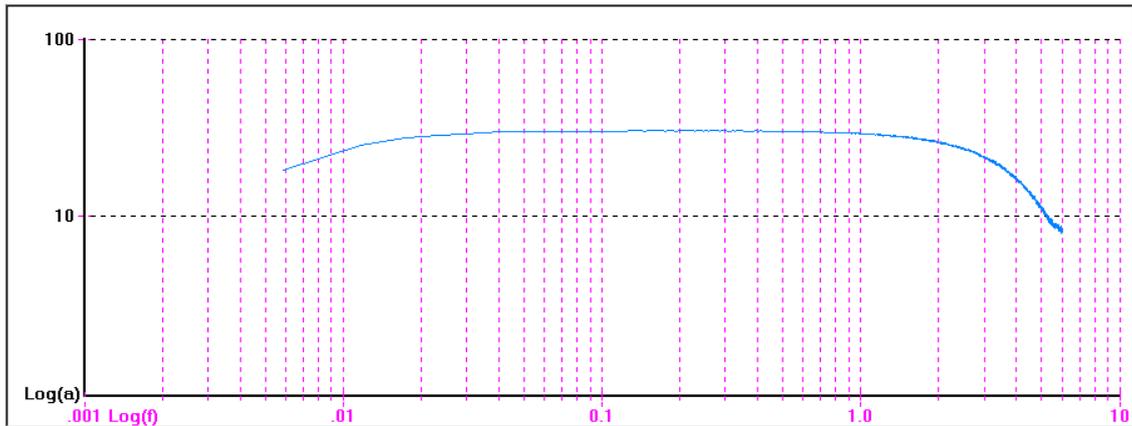
I used the parameters obtained by running CALEX on the seismometer with no amp and on the amp without the seismometer. The Digitizer documentation contains the information for the AD. Note here the amplifier gain is in dB. After calculating and saving the system, click on the Response tab and get the entire system description shown below. The output in the lower windows actually is in several forms such as poles and zeros, freq, amplitude, phase, etc.



Finally, I will demonstrate a method described by Bob McClure of the PSN to determine frequency response.

Bob's method uses an impulse or step and his WQFilter program to differentiate the output. First we see the step response with the decay caused by the filter. In the lower trace we see the differentiated signal of the step. Finally we do an FFT of the lower trace and see the response of the amplifier. Very neat and easy to do.



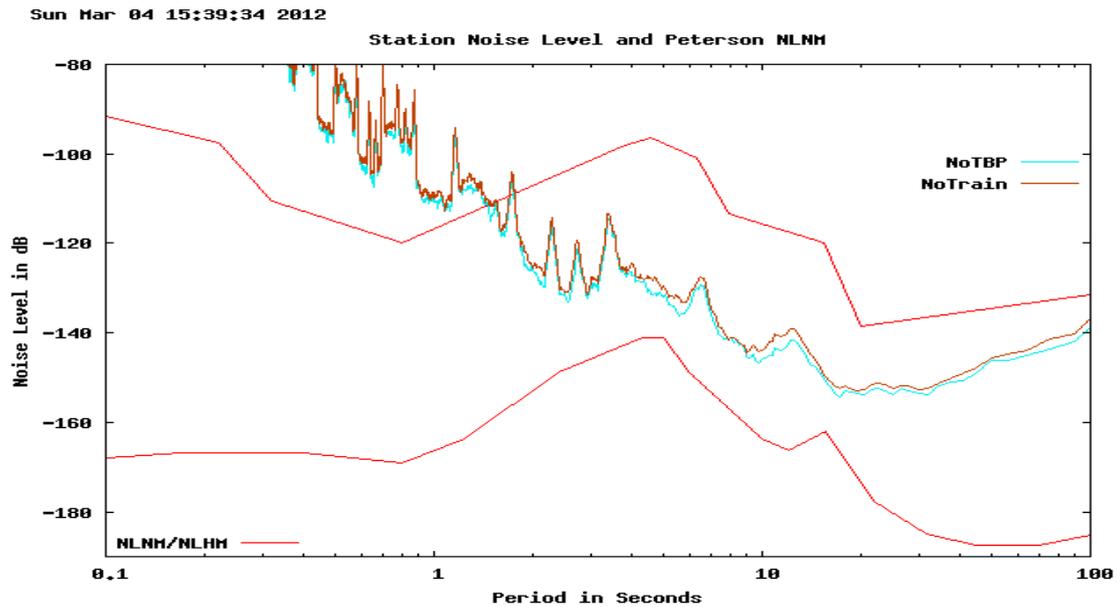


Now with our system described we can assess how our site is behaving.

John Lahr provided a program for comparing your performance to the low noise model of the earth. You can find it here.

<http://jclahr.com/science/software/psd/>

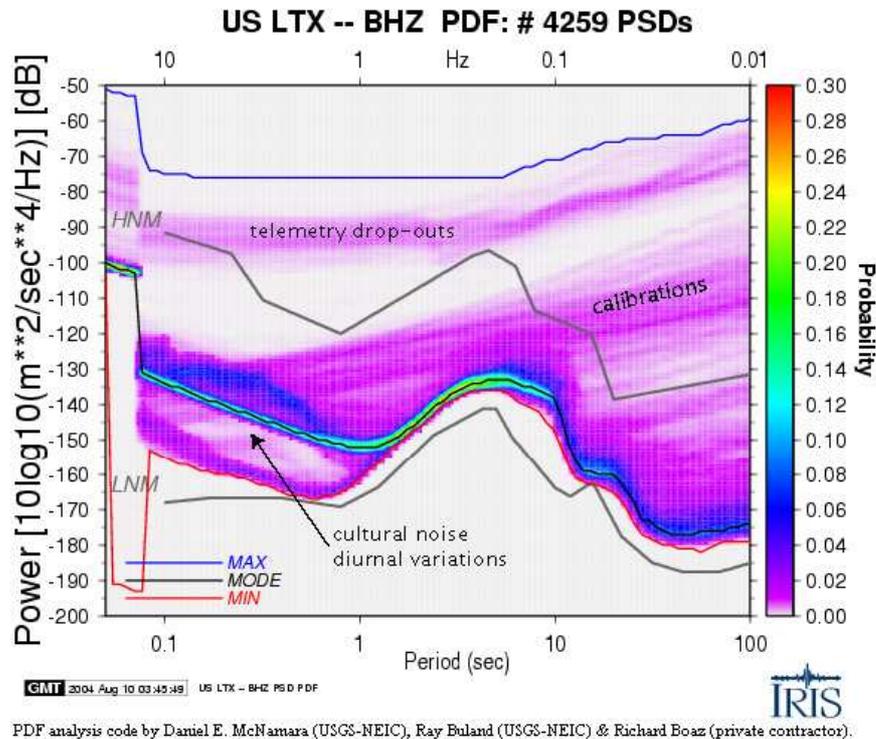
Here you are provided code and you can compile with Fortran. The data you need is automatically created by AmaSeis as EventData.csv in the Amaseis directory when you select a time period or you can easily create it from PSN ASCII by adding time stamps in a spreadsheet. The parameter file will require the poles and zeros that you created using SeismoWin in one of the prior steps or obtained in some other way. Again samples and instructions provided by John will assist you as you proceed. You results can be pasted into John's Excel sheet or you can recreate a similar sheet. If you are not familiar with Excel look closely at how John formatted the data to incorporate the NLNM on the same set of axis and scale. Here is an example created in GNUPlot.



If you want to take this one step farther there is a program titled PDFSA available here

<http://pubs.usgs.gov/of/2005/1438/pdf/OFR-1438.pdf>

This program will create noise plots of your system like the one below.



Software Availability

The new stand-alone software package is available on the world wide web (WWW) and on the USGS ANSS anonymous ftp server. To download a compressed tar ball of the software system, go to the following website and follow the downloading instructions.
<http://geohazards.cr.usgs.gov/staffweb/mcnamara/Software/PDFSA.html>

The download includes complete documentation on installing and running the noise processing system. It also includes several reference documents on interpreting the noise PDF plots (for example, McNamara and Buland, 2004).

I have successfully installed PDFSA on a PC running Linux. There is some difficulty in creating the required miniseed data structure from my PSN data that I have not solved and so I have not been able to evaluate my own data with this program at this point. I would like any suggestions as to how to easily convert PSN to miniseed.

In summary, all of the programs mentioned above, all of the programs needed to compile and run the programs and all of the documentation needed is available on line and at zero cost to the amateur seismograph operator. If you have any additional comments on these programs please address them on the PSN list.