LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY
- LIGO -
CALIFORNIA INSTITUTE OF TECHNOLOGY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Document Type	**LIGO-T080262-00-D**	2008/10/02
Cleaning and Assembly Procedure for ISI Sensor Pods		
Hanson J., Kissel J.,O'Reilly B.		

Distribution of this draft:

LIGO Scientific Collaboration

California Institute of Technology
LIGO Project - MS 51-33
Pasadena CA 91125
Phone (626) 395-2129
Fax (626) 304-9834
E-mail: info@ligo.caltech.edu

Massachusetts Institute of Technology
LIGO Project - MS 20B-145
Cambridge, MA 01239
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

WWW: http://www.ligo.caltech.edu/

Processed with LaTeX on 2008/10/02

1 Scope

This document is intended to outline the procedure we will follow for cleaning and assembly of the seismometer pods at LIGO Livingston. The BSC prototype at LASTI is instrumented with six L4-C geophones, six GS-13 geophones and three STS-2 Streckeisens, the HAM-ISI prototypes at LLO and LHO are each instrumented with six GS-13 geophones.

The instruments are intended to be sealed in pods for placement into the Advanced LIGO vacuum system [1, 2, 3]. Assembly, cleaning and testing of the seismometers will be done at the Livingston facility.

Streckeisen quoted us an 18 to 24 month lead time for four STS-2 units, with the possibility that part of the order might end up consisting of the new STS-3 device. Because of this, and because tests at Stanford show that we get good performance from the Nanometrics Trillium 240 (T-240) we intend to instrument future BSCs with the T-240. The quoted lead time from Nanometrics for these devices is 60 to 90 days. Costs are comparable to the STS-2. The T-240 has the additional advantage that it does not need to be locked and unlocked.

In order to keep track of the pods and the devices they contain, we intend to stamp each pod with a unique serial number. Our proposed numbering scheme is: <device type>-<orientation>-#, where <device type> is one of L4C, GS13 or T-240 and <orientation> is either “V” for vertical or “H” for horizontal. The T-240 pods will not have an orientation field in their serial number.

Each Advanced LIGO interferometer will require 111 podded instruments plus some complement of spares. We intend to clean, assemble and test all of these devices at the Livingston facility.

2 Cleaning

The cleaning procedure is based on LIGO-E960022-06-E [5]. All of the pod hardware requires a 48 hour bake, to which we have to add an 8 hour ramp up time and a cooling off period. It is estimated that each load will take approximately four days. There is approximately two days of preparatory work for each load, but some of this can be concurrent with baking other loads. We intend to use a rack to stack the parts and hence reduce the total number of loads.

Cleaned parts will be wrapped as specified in reference [5] prior to bringing them across to the assembly area.

3 Assembly

Since the seismometers will not be cleaned initial assembly will take place outside of our clean lab. The device used to lock the GS-13 seismometers is described in detail in [4].

We have separate assembly documents for the L4-C [6] and the GS-13 [7]. Our document for the STS-2 assembly [8] will be replaced with one for the T-240.

4 Testing

Prior to assembling the pod each instrument is tested to ensure functionality. We place the instrument in proximity to a working STS-2 and compare the power spectra and the transfer function between the instruments. The instruments are placed on a granite slab which is in good contact with the ground, this ensures good quality data down to low frequencies. A thermal/acoustic/air-current isolation box made of insulating foam is placed around the instruments. For Advanced LIGO production we will use the same technique on a bigger scale, so that we can test more instruments at once.

Once the pods are assembled, they are again tested to verify that the cabling, feed-throughs, locking mechanism (for the GS-13s) etc. are all functioning as expected. These tests are done with the pod wrapped in clean aluminum foil, to avoid contamination.

4.1 Leak Testing

We also test the pods for leaks. This is accomplished by placing the pods in a vacuum oven and performing an RGA scan of the contents. The oven has a calibrated leak which enables us to evaluate what leak rate of Neon (if any) is coming from the pod. For the initial complement of 27 pods we found no leaks. Many pods were tested more than once due to other problems, none failed the leak check. We regard the risk from a leaky pod to be extremely low.

References

[1] Allied Spacesystems Inc. *L4-C Seismometer Pod Assembly* LIGO-D047820-A

[2] Allied Spacesystems Inc. *GS-13 Seismometer Pod Assembly* LIGO-D047810-A

[3] Allied Spacesystems Inc. *STS-2 Seismometer Pod Assembly* LIGO-D047790-A

[4] S. Wen *Operation Principles and Procedures for the GS-13 Locker* LIGO-T070046-00-E

[5] LIGO Systems Engineering *LIGO Vacuum Compatibility, Cleaning Methods and Qualification Procedures.* LIGO-E960022-06-E

[6] Joe Hanson, Jeff Kissel, Brian O'Reilly *Building L4-C Pods* LIGO-T080261-00-R

[7] Jeff Kissel, Joe Hanson, Brian O'Reilly *GS-13 Modification and Pod Assembly Procedure* LIGO-T080086-00-D

[8] Joe Hanson, Jeff Kissel, Brian O'Reilly *STS-2 Assembly* LIGO-T080260-00-R

A Matlab script used to leak check pods.

```
% Read txt data from RGA scan
% 2008-Feb-6, (BO'R following D. Coyne example)
% A   rga only, no leak
% B   rga and oven, main turbo on, no cal leak
% C   rga and oven, main turbo on, cal leak on

NumPods = 6;
[amuA,ampA]=ReadASCData('11apr08rga.asc',21);
[amuB,ampB]=ReadASCData('11apr08rga+oven.asc',21);
[amuC,ampC]=ReadASCData('11apr08rga+oven+leak.asc',21);

indx=find(ampA==0);
ampA(indx)=1e-18;
whos indx
indx=find(ampB==0);
ampB(indx)=1e-18;
whos indx
indx=find(ampC==0);
ampC(indx)=1e-18;
whos indx

% amuA = amuA(1:end-1);
% amuB = amuB(1:end-1);
% amuC = amuC(1:end-1);

figure(1);
semilogy(amuA,ampA,'r-',amuB,ampB,'b-',amuC,ampC,'k-');
title('Load leakcheck11apr08');
axis([0 100 1e-13 1e-8]);
xlabel('AMU');
ylabel('amp');
grid on;
orient landscape
legend('rga only, no leak', ...
'rga and oven, main turbo valved off, no cal leak', ...
'rga and oven, main turbo valved off, cal leak on');
print('-dpdf','leakcheck11apr08_AllScans.pdf')
%

% Calibrated Leak
% Accu-Flow Calibrated Leak
% Model CL-7-Std Mix-2CFF-300DOT-MFV, Serial No. 12603
% 1.67e-7 atm-cc/s = 1.25e-7 torr-liter/s
```

```
% 3%Kr/Xe/He/H2 1.8%N2 in Ar, i.e. 86.2% Ar (i.e. 1.00 - 4*0.03 - 0.018 = 0
% Ar leak rate:
LR=0.86*1.5e-7; %torr-liters/s

% Pumping speed at cracked hydrocarbon AMUs (~50) assumed to be about the
% same as for N2 Turbo V70
% Also had a Pfeiffer-Balzers TPU 060
%S=68; %liters/s
S=68; % V70 + TPU 060 liters/s

% AMU Shift
% Find the AMU associated with the Ar peak at 40
ind1=find(amuC > 39);
ind2=find(amuC > 41);
[a,b]=max(ampC(ind1(1):ind2(1)))
shift=amuC(ind1(b))-40;

% Calibration factor
ind1=find(amuC >= (40+shift));
cal=LR/(S*ampC(ind1(1))) %torr/Amp

LRcal = ['Calibtration = ' num2str(cal,2) ' torr/Amp'];

% Find the amplitide associated with the HE peak at 4
ind1=min(find(amuC > 3.5));
ind2=min(find(amuC > 4.5));
HeAmp=max(ampC(ind1:ind2));

% Hydrocarbon Outgassing Rate
ind41=find(amuC >= (41+shift));
ind43=find(amuC >= (43+shift));
ind53=find(amuC >= (53+shift));
ind55=find(amuC >= (55+shift));
ind57=find(amuC >= (57+shift));
amp41=ampC(ind41(1));
amp43=ampC(ind43(1));
amp53=ampC(ind53(1));
amp55=ampC(ind55(1));
amp57=ampC(ind57(1));
HC=(amp41+amp43+amp53+amp55+amp57)*cal*S
%noiselimitHC=5*1.2e-13*cal*S
amp61=ampC(find(amuC >= (60+shift)));
amp62=ampC(find(amuC >= (60+shift)));
amp63=ampC(find(amuC >= (60+shift)));
amp64=ampC(find(amuC >= (60+shift)));
amp65=ampC(find(amuC >= (60+shift)));
```

```
noiselimitHC=(amp61(1)+amp62(1)+amp63(1)+amp64(1)+amp65(1))*cal*S/5


HCOutgas=['Hydrocarbon outgassing rate = ' num2str(HC,2) ' torr-liter/s'];
HCNoise =['Noise limit = ' num2str(noiselimitHC,2) ' torr-liter/s'];

figure(2)
semilogy(amuC,ampC*cal,'k-');
title('RGA of Oven Load with calibrated leak');
axis([0 100 1e-13 1e-8]);
xlabel('AMU');
ylabel('torr');
grid on;
orient landscape
legend('oven bake load and calibrated leak');
aa=text(40,2e-10,HCOutgas);
set(aa,'FontSize',16,'Color','red','FontWeight','bold');
aa=text(40,9e-11,HCNoise);
set(aa,'FontSize',16,'Color','red','FontWeight','bold');
aa=text(40,5e-11,LRcal);
set(aa,'FontSize',16,'Color','red','FontWeight','bold')
print('-dpdf','leakcheck11apr08rga+oven_torr_leak.pdf')

figure(3)
semilogy(amuA,ampA,'k-');
title('RGA Only');
axis([0 100 8e-14 1e-8]);
xlabel('AMU');
ylabel('Amps');
grid on;
orient landscape
legend('rga chamber');
print('-dpdf','leakcheck11apr08rga.pdf')

figure(4)
semilogy(amuB,ampB,'k-');
title('RGA+Oven Only');
axis([0 100 8e-14 1e-8]);
xlabel('AMU');
ylabel('Amps');
grid on;
orient landscape
legend('rga+oven no leak');
print('-dpdf','leakcheck11apr08rga+oven.pdf')

% Leak Rates
```

```
LR_Kr=0.03*1.25e-7;
LR_He=0.03*1.25e-7;
LR_H2=0.03*1.25e-7;
LR_N2=0.018*1.25e-7;
LR_Ar=0.862*1.25e-7;
LR_air=1e-8;
LR_H2O=1.5e-7;

gas=zeros(1,100);
amus=1:100;

% Neon Leak Rates
%
% From AMU 20 peak. Calibration is from Ar cracking pattern to AMU 20 from
% calibrated leak. Also has a contribution of 2.3% from AMU 41 (C3H6, cyclo-

noleak_amp84=ampB(min(find(amuB >= (84+shift))));
leak_amp84=ampC(min(find(amuC >= (84+shift))));
noleak_amp41=ampB(min(find(amuB >= (41+shift))));
noleak_amp40=ampB(min(find(amuB >= (40+shift))));
leak_amp40=ampC(min(find(amuC >= (40+shift))));
noleak_amp20 = ampB(min(find(amuB >= (20+shift))));
leak_amp20 = ampC(min(find(amuC >= (20+shift))));
noleak_amp4 = ampB(min(find(amuB >= (4+shift))));
leak_amp4 = ampC(min(find(amuC >= (4+shift))));


peak20 = noleak_amp20 - noleak_amp40*0.05 - noleak_amp41*0.023;
cal20 = (LR_Ar*0.05)/(leak_amp20 - noleak_amp20);

neon_leak_rate_20 = cal20*peak20/NumPods;

% From calibration of Argon (AMU 40) peak

cal40 = LR_Ar/(leak_amp40 - noleak_amp40);
neon_leak_rate_40 = cal40*peak20/NumPods;


% From calibration of Helium (AMU 4) peak

cal4 = LR_He/(leak_amp4 - noleak_amp4);
neon_leak_rate_4 = cal4*peak20/NumPods;

% From calibration of Kr (AMU 84) peak
```

```
cal84 = LR_Kr/(leak_amp84 - noleak_amp84);

neon_leak_rate_84 = cal84*peak20/NumPods; % torr-liter/s

NeonLeakRate20 =['Ne Leak from Ar cracking to AMU 20 = '...
                 num2str(neon_leak_rate_20,2) ' torr-liter/s']
NeonLeakRate40 =['Ne Leak using Ar peak (AMU 40) = '...
                 num2str(neon_leak_rate_40,2) ' torr-liter/s']
NeonLeakRate4 =['Ne Leak  using He peak (AMU 4) = '...
                 num2str(neon_leak_rate_4,2) ' torr-liter/s']

NeonLeakRate84 =['Ne Leak using Kr peak (AMU 84) = '...
                 num2str(neon_leak_rate_84,2) ' torr-liter/s']

figure(5);
semilogy(amuB,ampB,'b-',amuC,ampC,'r-');
title('Load leakcheck11apr08');
axis([15 25 8e-14 1e-10]);
xlabel('AMU');
ylabel('amp');
grid on;
orient landscape
legend('rga and oven, no cal leak', ...
'rga and oven, cal leak on',2);
aa=text(19,4e-12,NeonLeakRate20);
set(aa,'FontSize',14,'Color','red','FontWeight','bold');
aa=text(19,1.5e-12,NeonLeakRate40);
set(aa,'FontSize',14,'Color','red','FontWeight','bold');
aa=text(19,5e-13,NeonLeakRate4);
set(aa,'FontSize',14,'Color','red','FontWeight','bold');
aa=text(19,2e-13,NeonLeakRate84);
set(aa,'FontSize',14,'Color','red','FontWeight','bold');

print('-dpdf','leakcheck11apr08_Neon.pdf')
```

B Sample RGA Scan from Leak Check

Figure 1(a) shows the standard RGA scan we perform to evaluate the cleanliness of a bake load. This is also done during the leak check to ensure that the assembly process has not caused contamination. Figure 1(b) plot is a zoom in around the AMU 20 peak which corresponds to Neon. Written on this figure are various Neon leak rates calculated from different peaks in the RGA spectrum.

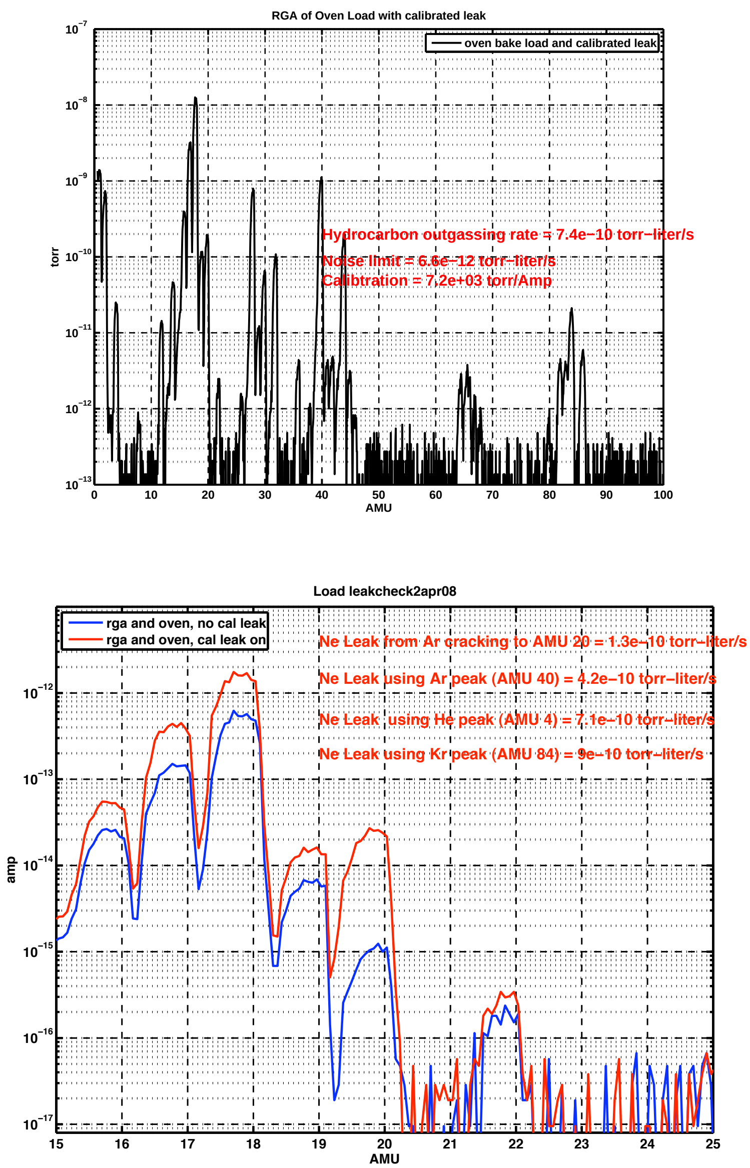

Figure 1: Zoomed in RGA scan of a pod showing the Neon peak at AMU=20. The Neon leak rate is calculated from different peaks in the spectrum.