LASER INTERFEROMETER GRAVITATIONAL WAVE OBSERVATORY

LIGO Laboratory / LIGO Scientific Collaboration

LIGO-T1200083-v1 *Advanced LIGO* April 13, 2012

aLIGO Kepco DC Power Supply Acceptance Testing Scripts

Carl Adams

Distribution of this document:
LIGO Scientific Collaboration

This is an internal working note
of the LIGO Laboratory.

California Institute of Technology
LIGO Project – MS 18-34
1200 E. California Blvd.
Pasadena, CA 91125
Phone (626) 395-2129
Fax (626) 304-9834
E-mail: info@ligo.caltech.edu

Massachusetts Institute of Technology
LIGO Project – NW22-295
185 Albany St
Cambridge, MA 02139
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

LIGO Hanford Observatory
P.O. Box 1970
Richland WA 99352
Phone 509-372-8106
Fax 509-372-8137

LIGO Livingston Observatory
P.O. Box 940
Livingston, LA 70754
Phone 225-686-3100
Fax 225-686-7189

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

aLIGO Kepco DC Power Supply Incoming Testing Scripts

1. Introduction

The testing scripts described below are for noise testing the *Kepco*™ **JQE-25** power supplies, as outlined in **LIGO-T1200070-v1 and LIGO-T1200084**-v1.

1.1. The noise measurement script "DCPS_fft" is a bash script, which utilizes python scripts written at the LIGO 40 meter lab, to configure and run the SR 785 and output text files of the measurement data.

1.1.1. This script is written to take a 3-part measurement of different overlapping frequency spans to allow for better resolution up to 3KHz, and still span up to 100KHz.

1.2. The matlab script "plotDCPSNoise_v#.m" creates a 'stitched' file from the 3 part data files to span the 0 – 100KHz measurement. Some of it's main features are:

1.2.1. Stitches 3 noise measurement files (allowing for better resolution up to 3K, and still span up to 100K)
1.2.2. Adds to header info about files used to create stitch file
1.2.3. Adds to header summary table of noise at 1, 10, 100, 1000, and 10000 Hz
1.2.4. Evaluates data vs. reference line (file for each measurement type)
1.2.5. Ignores 60 Hz harmonics and other known environmental noise (see array 'ignore')
1.2.6. Adds to header amplitude and frequency of each data point in excess of reference
1.2.7. Plots baseline, 0v, and 25v stitched data and reference on same axis
1.2.8. Outputs *.fig and *.jpg in addition to the stitched text files

2. Applicable Documents

2.1. LIGO-T1200070-v1, "aLIGO DC Power Supply Voltage Noise Requirement", by *Rich Abbott,* 8-Feb-2012

2.2. LIGO-G1200043-v3, "DCPS Acceptance Testing" presentation, by *David Kinzel*

2.3. LIGO-T1100478-v1, "Acceptance Testing Of aLIGO DC Power Supplies" by *Rich Abbott & David Kinzel* 9-Sep-2011.

2.4. LIGO-T1200084-v, "Kepco DC Power Supply Acceptance Testing"

3. Test equipment used.

3.1. *Stanford* **SR785**, 2 channel dynamic signal analyzer

3.2. GPIB to ethernet adapter, *Prologix,* or equivalent.

3.3. Computer running Linux or Mac OSX, with Matlab installed.

3.4. The "DCPS_fft" bash script will be used to configure and run the SR 785 to collect the noise measurement data.

3.4.1. This script was developed from SUS electronics testing scripts and works in

conjunction with “SUS_Measurement_Launcher”,"resetSR785.py", and “FFTSR785_v4.py” in the sus svn.

3.4.2. This script and other associated scripts are located in the sus svn at: https://redoubt.ligo-wa.caltech.edu/websvn/listing.php?repname=sus& just login using your ligo.org account.
Navigate to: trunk/electronicstesting/sus_gpib_scripts/sus_gpib_llo
for the bash and python scripts

3.5. The matlab script “plotDCPSNoise_v#.m” is used to ‘stitch’ the three part noise measurement into one data file and provide a comparison plot of all measurements.

3.5.1. This script is also located in the sus svn, browse to
trunk/electronicstesting/matlab_scripts/
for the matlab script.

4. Testing scripts Setup

4.1. To use a local machine to run the testing scripts do the following:

4.1.1. Decide where on the computer the SVN script folder will be, and create a folder “svn" (or other name of your choice) at that location.

4.1.2. To check out a local copy of the repo on your local machine, do the following:

4.1.2.1. Using the terminal, navigate to the location of the svn folder, using the "cd" command.

4.1.2.2. At a terminal prompt, copy/paste or type the following; " svn co https://redoubt.ligo-wa.caltech.edu/svn/sus/trunk/electronicstesting/ --username *your.name*", where *your.name* is your ligo.org name. This will copy or "checkout" the " electronicstesting/" folder and contents to the location that the command was entered from. This will include both the sus_gpib_scripts/sus_gpib_llo and matlab_scripts subfolders.

4.1.3. To update scripts, in the terminal, navigate to the appropriate folder under your local “svn” folder, and type at the $ prompt; “svn up”. This will update all scripts in all subfolders except any locally modified files.

4.1.4. The "SUS_Measurement_Launcher" is the only script that needs to be modified to run on a local computer, all other SR785 testing scripts are launched from this script. This is done by opening the script, with a text editor, such as “gedit”, "textedit.app" or "TextWrangler.app"

4.1.4.1. Replace "/Users/*username*/../svn/electronicstesting/" after the “root=" "” in line 6, with your path. e.g. #file path/location of the svn/ electronicstesting folder (this must be locally defined).

4.1.4.2. Also the ip address of the GPIB to LAN adaptor of the SR 785 needs to be modified.

4.1.4.2.1. To facilitate the use of multiple analyzers from a single script, only the root of the ip address, minus the last digit(s) (assuming all analyzers on the LAN only differ by the last digit(s)), is written in the script. Replace “IPstem="10.110.40.10"”, on line 13 with one appropriate for your LAN.

4.1.5. To make the test scripts easier to run, add a path to those scripts by modifying the "paths" file on your computer, by doing the following;

For mac os x in /etc/paths
just add new line with exact path at bottom of file:
/usr/bin
/bin
/usr/sbin
/sbin
/usr/local/bin
/Users/username/../SVN/electronicstesting/ sus_gpib_scripts/sus_gpib_llo

4.1.5.1. This was done by editing the "paths" file, located in the "etc" folder, on the root, and adding the last line above, for a general case; "Users/*username*/.../svn/electronicstesting/ sus_gpib_scripts/sus_gpib_llo", where "*username*" is the name of the user's home folder, "/.../" is the path from the home folder to the "svn" folder. You need to exit the terminal program and restart it for the changes to take affect.

4.1.6. Next, decide where the test data will be saved to, e.g."/Users/*username*/.../test data/DC_Power_Supplies/S12345678" with sub-folders for serial number.

4.1.7. The testing scripts work best with the Prologix LAN/GPIB adaptors used with the SR 785. Ensure that your device is properly setup as per your local requirements.

4.2. Run the Noise testing script.

4.2.1. Launch "Terminal.app", "X11" app, or other terminal program.

4.2.2. In the terminal screen, navigate to the data folder for the DCPS serial number, and launch "*SUS_Measurement_Launcher*". If the script folder was added to the path, you should just be able to type in the name of the file and it will launch. If not added to the path, you will have to type: "./Users/*username*/Documents/.../svn/electronicstesting/ sus_gpib_scripts/sus_gpib_llo/ *SUS_Measurement_Launcher"* to launch the script.

4.2.2.1. To run the script to test the DCPS for noise, select "11" and follow the directions in the script. It will open in a new X11 window. The first prompt will be for the last digit of the ip address for the SR785 to perform the test.

4.2.2.2. For test note, use "baseline, 0V, and 25V" respectively, and continue test.

4.2.3. The script will create three data files in the form "*Kepco_DCPS-HXXXXXX-Noise_Test-ZZ-part#-YYYYYYYY-TTTTTT.txt*",

where "*XXXXXX*" is the serial number.

where "*ZZ*" is the note, "baseline", "0V", or "25V".

where "*YYYYYYYY*" is the date code.

where "*TTTTTT*" is the time code.

where "*part#*" is the part string, 1, 2, or 3 for the 3-part measurement.

4.3. Matlab analysis script

4.3.1. This script, when launched, will prompt you to select a file in the folder of the DCPS you want to stitch and plot. Make sure the only *.txt files in the folder are the ones generated by the "DCPS_fft" bash script.

4.3.2. It will generate a new *.txt file with the stitched data for each noise measurement and then plot them all in a single axis for comparison.

4.3.3. The script also compares the test data to the reference for each stitched measurement in the folder. The comparison will ignore all 60 Hz harmonics and any know environmental noise as defined in the "ignore" array. The max noise from the 25V measurement is displayed on the plot.

4.3.3.1. Identify any environmental noise by performing a 'baseline' noise measurement with the power supply and load cables up as it will be during the

actual test. As any noise measured in the baseline, which exceeds the reference, is obviously not contributed by the power supply, we can add those frequencies to the "ignore" array to allow a clear picture of the noise of the power supply.

4.3.4. It will also output *.fig and *.jpg files in addition to the stitched text files.