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

06/08/2026

Data Acquisition and Signal Processing for the GQuEST Experiment

Ciara Treanor

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

Massachusetts Institute of Technology
LIGO Project, Room NW17-161
Cambridge, MA 02139
Phone (617) 253-4824
Fax (617) 253-7014
E-mail: info@ligo.mit.edu

LIGO Hanford Observatory
Route 10, Mile Marker 2
Richland, WA 99352
Phone (509) 372-8106
Fax (509) 372-8137
E-mail: info@ligo.caltech.edu

LIGO Livingston Observatory
19100 LIGO Lane
Livingston, LA 70754
Phone (225) 686-3100
Fax (225) 686-7189
E-mail: info@ligo.caltech.edu

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

Contents

1 Introduction **2**

1.1 GQuEST 2

2 Objectives **3**

3 Approach **3**

3.1 Photodiode Characterization 3

3.2 Data Acquisition System 7

4 Future Plan of Action **10**

1 Introduction

1.1 GQuEST

Theoretically, the quantum nature of gravity is expected to stochastically produce "geontropic" space-time fluctuations, or geometric fluctuations in space-time due to entropy. The gravity from the quantum entanglement of space-time (GQuEST) experiment aims to detect these space-time fluctuations as evidence of quantum gravity. GQuEST will use a Michelson interferometer with 7 m long arms and a 1550 nm frequency-stabilized laser with a superconducting nanowire single-photon detector (SNSPD).

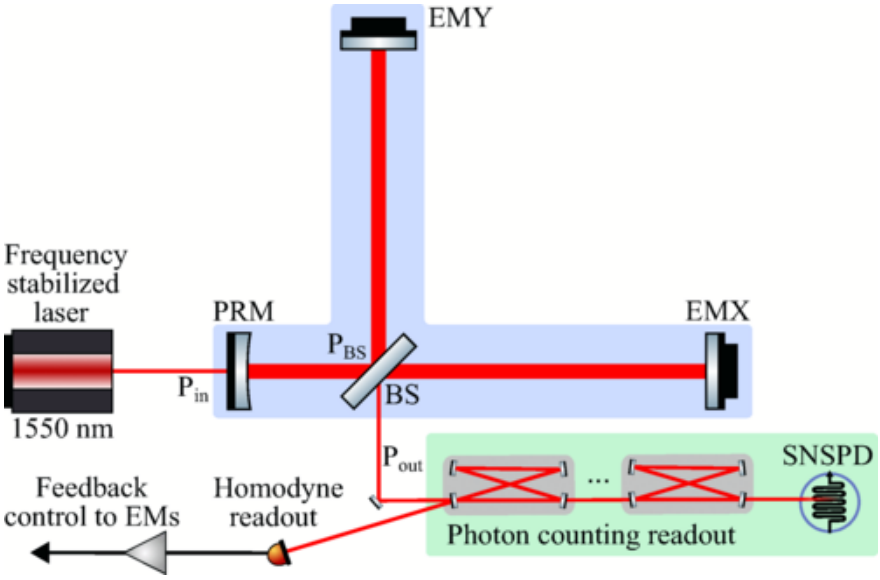


Figure 1: Simplified GQuEST experimental design [1].

The SNSPD will be used in addition to homodyne measurements to get a more precise readout not subject to the limit from photon shot noise. In homodyne, the system measures the signal times the local oscillator, while photon-counting filters out the local oscillator to recover phase, an important metric for LIGO when measuring gravitational waves. GQuEST will be measuring what is believed to be the stocahstic and broadband signal of spacetime fluctuations, meaning phase measurements are no longer necessary. Instead GQuEST will measure a power that is the amplitude of the fluctuations occurring at a specified frequency (roughly 17MHz) where the experimental and background noise is at a minimum.

The GQuEST experiment is motivated by recent geontropic quantum gravity models which suggest that space-time is emergent from quantum entanglement. In these models, the volume of space-time is not a smooth continuum but contains a fundamental graininess linked to the information density on its boundary. These fluctuations are predicted to accumulate over macroscopic distances, manifesting as a stochastic "metric noise".

By measuring the cross-correlation between two independent interferometers, we aim to iso-

late this common space-time "shimmer" from the uncorrelated quantum shot noise inherent in individual detectors. The introduction of SNSPDs is a critical advancement; by counting individual photons, the experiment can extract Fisher information regarding the path-length fluctuations much more efficiently than traditional homodyne systems, effectively bypassing the Standard Quantum Limit [1].

2 Objectives

The initial goals in the first three weeks of this project were:

- To test a modified photodiode's frequency response.
- To create data acquisition code based on the code from the Holometer experiment.

3 Approach

3.1 Photodiode Characterization

The first phase for this project , over Weeks 1 and 2, was to build a circuit to use to test the frequency response of modified photodiode and compare it to an unmodified model. We need to characterize these photodiodes for future use in quantifying laser and mechanical noise in the interferometer and filter cavities for fine tuning the GQuEST experiment. To do this, a photodiode was modified to be able to handle higher power and thus its response needed to be characterized in relation to an unmodified version. Lee McCuller modified a Thorlabs PDA05CF with an inductor/resistor pair to lower the gain (approximately 0.1 the initial value) at DC, while retaining high gain at RF.

I used a bias tee circuit to power an LED. I built the circuit on a breadboard, powered with 6-8 V DC, and initially tested the circuit outputs on various inputs using a Siglent oscilloscope and a Moku Go. The photodiode showed with 400 mVpp - 1 Vpp. I attempted to use a 1550 nm (Thorlabs LED1550L), 780 nm (Thorlabs LED780L), and 570 nm LED on the photodiode to get the frequency response function, but switched to a 685 nm laser diode (Thorlabs HL6750MG) for a stronger output signal.

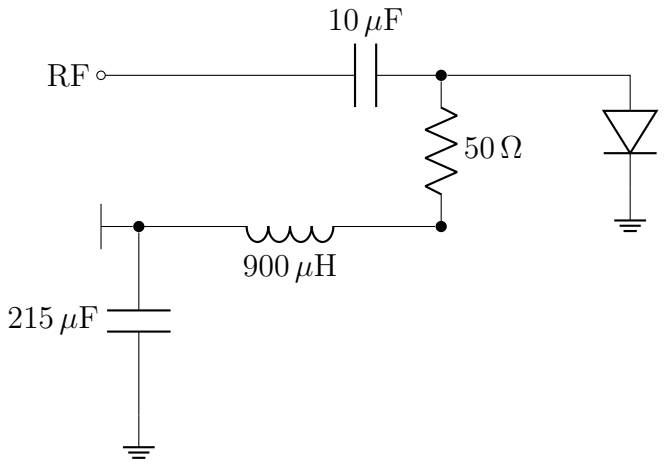


Figure 2: Bias tee circuit used to modulate 685 laser diode

The photodiode used is an Indium Gallium Arsenide (InGaAs) fixed gain amplified detector. The frequency response was tested using the same circuit powered with 8 V DC (with 100 mA current limit) and swept from 1 kHz to 10 MHz with 1 Vpp AC.

The frequency response function of the modified photodiode and an unmodified photodiode confirmed the expected results. At DC, the modified photodiode is theoretically expected to have around 0.1 the amplitude of the unmodified one, while they perform similarly at high frequencies.

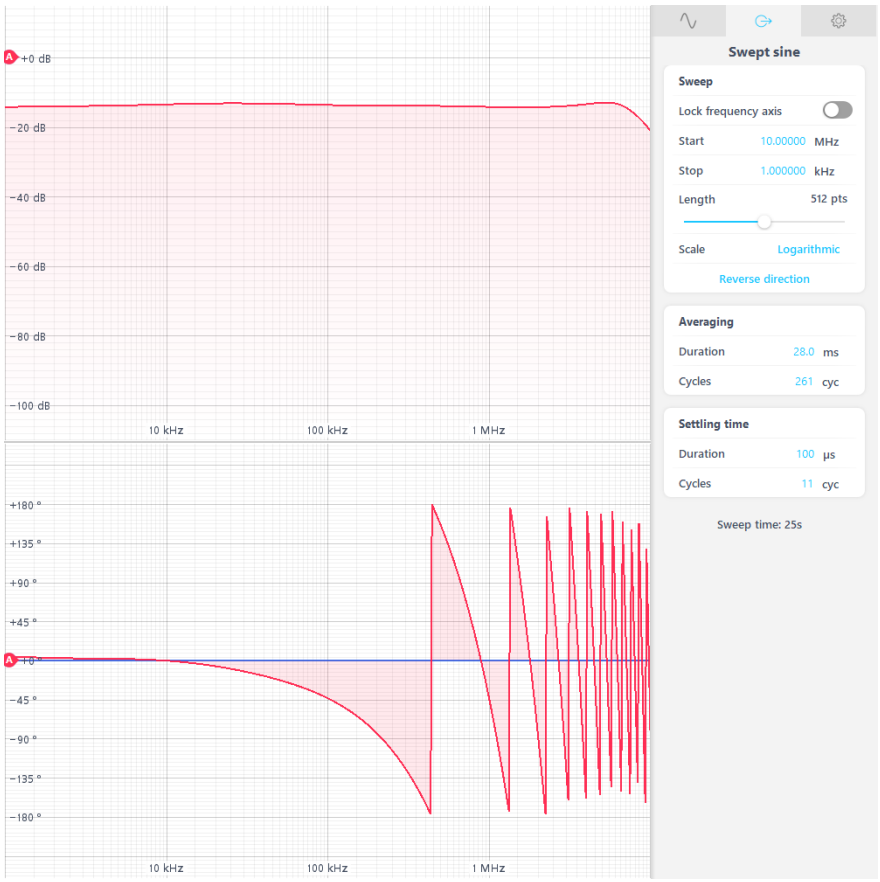


Figure 3: Frequency response of unmodifed photodiode

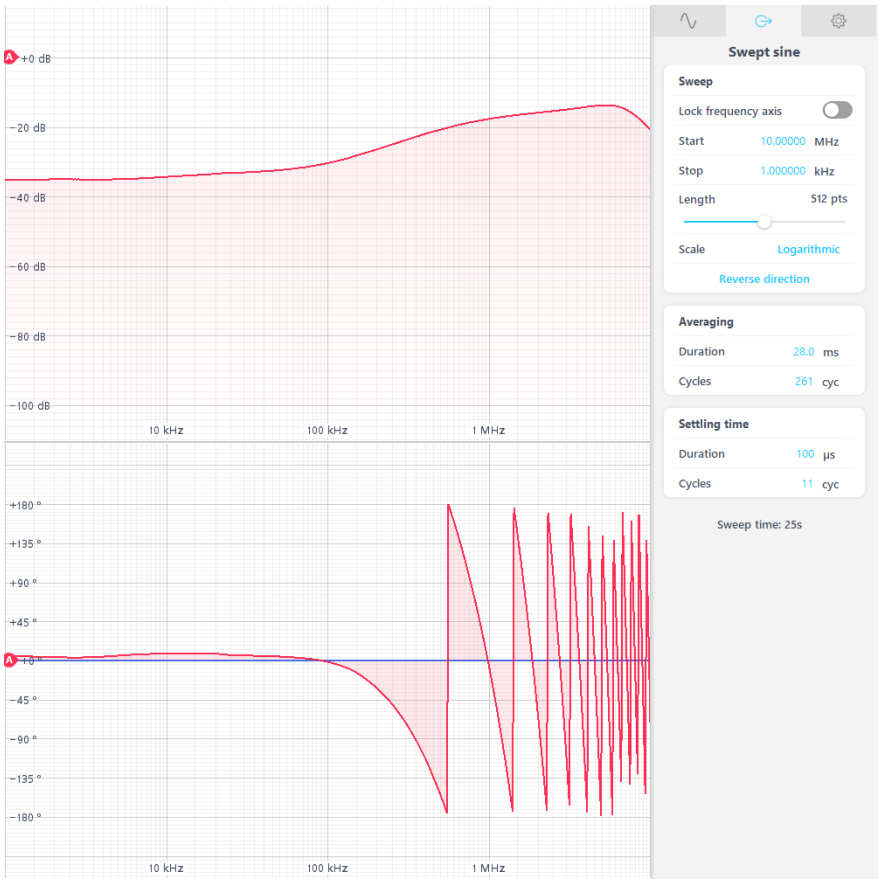


Figure 4: Frequency response of modified photodiode

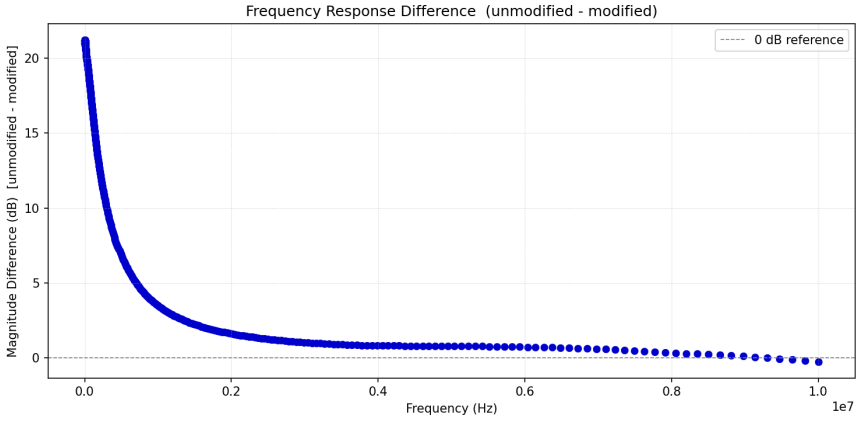


Figure 5: Difference between frequency response of unmodifed and modified photodiode

Figure 5 clearly shows that the modified photodiode has 20 dB lower amplitude (10%) than the unmodified at DC, but the two remain within 1 dB at high frequency.

3.2 Data Acquisition System

GQuEST's data acquisition code is modelled after that of the Holometer experiment [2]. I spent most of Week 3 attempting to modify the existing C++ and Python code from the Holometer experiment to interface with GQuEST's 4-channel, 16Bit, 125 MS/s per channel digitizer from Advantech.

The photodetector measures the incident power of the light as a voltage, and the program discretizes the data by creating frequency bins ($\sim$0.655ms integration time for the Holometer gave $N \approx 66,000$) [2]. Each of these bins constitutes one frame. Assuming the signal is stationary, we process the data using these frames and then average to get the final results.

Given that GQuEST is looking for signals on the order of MHz, it needs to have a MHz sampling rate, which results in an enormous rate of data acquisition which is unfeasible to store. As a result, we will store only time-averaged spectra, which are calculated in real time.

After obtaining the N frames, we apply a Fourier transform. First, we apply the Hanning window function to attenuate the ends of the frame [3]:

$$\omega_n = \frac{1}{2}[1 - \cos(\frac{2\pi n}{N})] \tag{1}$$

Next, we perform a Fast Fourier Transform on the data and average over all frames. We can then store the resulting averaged vectors in a covariance matrix, where the diagonal elements represent the time-averaged power spectrum of each channel and the off-diagonal elements represent the time-averaged cross-correlations between channels [2]. The matrix is symmetric so we can set the lower triangular elements to 0.

Each matrix contains 4 real diagonal elements, each 64 bits or 8 bytes. The 6 complex off-diagonal elements are represented by 2 floating point numbers, so require 16 bytes. This means each matrix requires 128 bytes of storage.

Based on the sampling rate and integration time, the number of samples is:

$$N = 100 \text{ MHz} \times 0.655 \text{ ms} \approx 66,000 \tag{2}$$

And therefore the number of frequency bins is:

$$\#\text{ frequency bins} = \frac{N}{2} \approx 33,000 \tag{3}$$

Therefore the amount of data for each frame is:

$$128 \text{ bytes} \times 33,000 \text{ per frame} = 4,224,000 \text{ bytes/frame} \approx 4 \text{ MB/frame} \quad (4)$$

This is a significant amount of data, so we must determine how often to save a frame in order to preserve storage space. Waiting at least one second between saves to maintain 4 MB/s or lower data rate is a reasonable limitation. The raw data rate of 6.4×10^9 bits/s is roughly equal to 800 MB/s, so this offers roughly 200-fold compression.

This process will reduce the total data footprint substantially, allowing for long-term integration without exceeding storage limits. In order to process the data as fast as it is collected, the program must perform FFTs as fast as each frame takes, which is equal to the integration time of ∼0.655ms. Therefore, the program must be optimized, and the limits of the hardware and the program speed (specifically, FFT calculation) must be taken into account when determining the integration time.

I spent most of Week 3 reading the Advantech ADC data acquisition code and the Holometer data acquisition and signal processing code to understand the data flow in each system and how to interface the two.

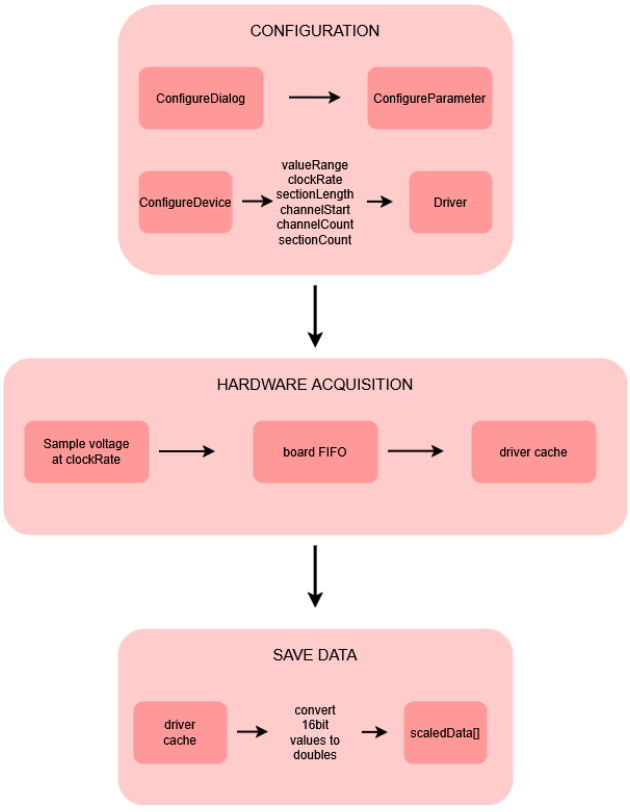


Figure 6: Adventech ADC data acquisition

Essentially, the Advantech code provides a prebuilt system for analog input data acquisition.

This allows the user to edit the number of channels and the sampling rate used, which is then sent to the hardware. The voltages are sampled at the chosen clock rate and stored as 16-bit integers on the board, and then send to the computer where it is then converted into doubles and stored in an array. For our purposes, we will actually leave the voltages as 16-bit integers, as the signal processing code takes 16-bit integers as input.

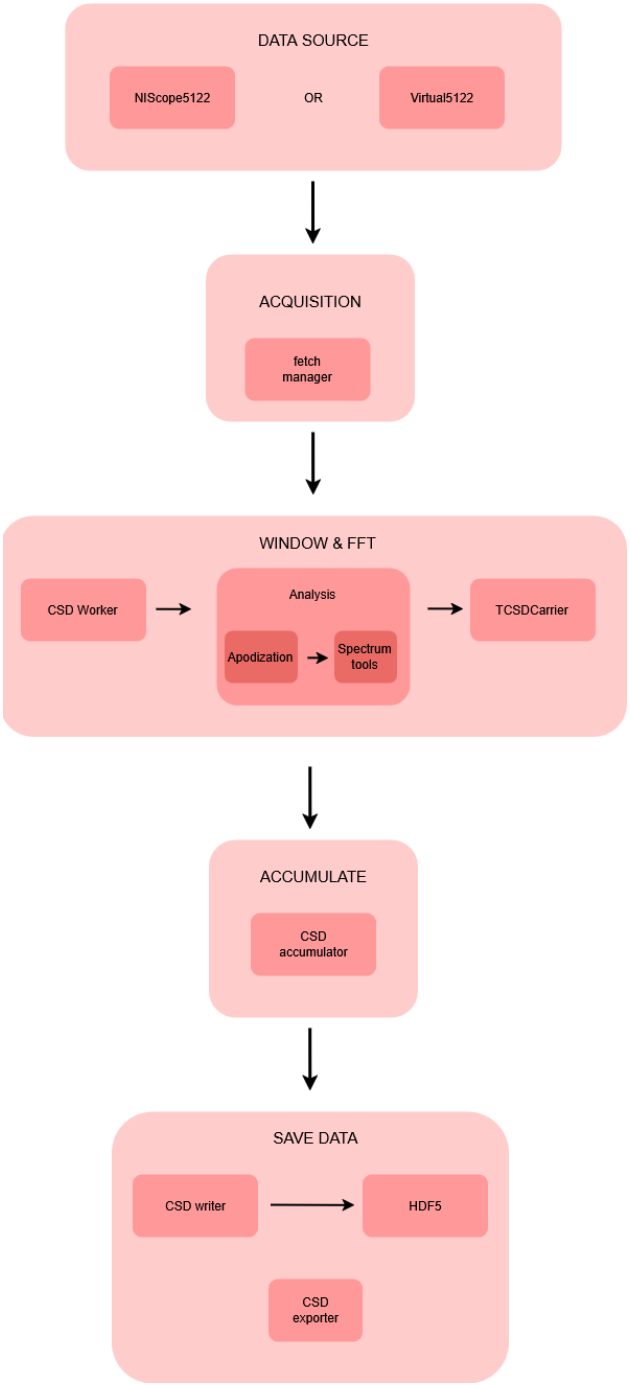


Figure 7: Fermilab Holometer data acquisition and signal processing

The signal processing code takes voltage values (either from a real ADC or simulated source), where it is windowed (apodization) and FFTs are performed (spectrum tools). Then, it is accumulated and saved.

I then attempted to create simulated data in the same format as the digitizer output to test the signal processing, but this has not been successful yet; I plan to finish this in Week 4.

4 Future Plan of Action

In the next few weeks, I plan to finish altering the data acquisition code and use it to help test the photodiode with a laser setup.

Week 4: Edit the data acquisition code to take data from the real digitizer.

Week 5: Set up the photodiode for use with a laser. Test the code with this setup.

Weeks 6 & 7: Perform more tests with the data acquisition code and the photodiode to determine limitations, and make any necessary changes.

I will use the libraries and objects in the sample code from Advantech to intake data from the ADC, then transfer the data into the Holometer's jstream signal processing code directly as 16-bit integers. Then, I can test this system with the photodiode to ensure it is reading and processing information correctly. By modulating a laser diode or LED, I can create a predictable peak in the spectrum which I can verify in the spectra computed by the code.

References

[1]S. M. Vermeulen, T. Cullen, D. Grass, I. A. O. MacMillan, A. J. Ramirez, J. Wack, B. Korzh, V. S. H. Lee, K. M. Zurek, C. Stoughton, and L. McCuller, "Photon-counting interferometry to detect geontropic space-time fluctuations with gquest", Physical Review X **15**, 011034 (2025).

[2]J. Richardson, *The holometer data acquisition system*, Technical Document, Details the data processing computation performed by the Julie Stream data acquisition code (Fermi National Accelerator Laboratory, June 2013).

[3]M. Cerna and A. F. Harvey, *The fundamentals of fft-based signal analysis and measurement*, Application Note 041 (National Instruments, July 2000).