Technical Note	**LIGO-T1500351—v1**	2015/10/04

Quantization Noise in Advanced LIGO Digital Control Systems

Ayush Pandey, Christopher Wipf,
Rana Adhikari, Jameson Graef Rollins

Preface

.

The project was focused on analyzing quantization noise in the Advanced LIGO digital control systems. The theoretical background and a thorough literature review on the quantization noise subject (in general) has been presented in the first few sections of this report (sections 1 to 3). A brief description of the LIGO detector and it's working principle is given in section 4. Based on the theory developed in these sections, results on the quantization noise levels for Advanced LIGO digital filters, (in section 6) and for the digital to analog converters used in Advanced LIGO (in section 8) have been given. Finally, a denoising technique called noise shaping is described to reduce the noise levels due to DACs in a desired frequency band. Simulations and testing results to support the arguments are also given in section 9.

1 What is Quantization

Quantization is a process of mapping continuous set of values into a finite or countably infinite set of values by approximations such as truncation, rounding etc. The function which performs this approximation operation is called a quantizer. The loss in data in this process is known as quantization noise or quantization error which is equal to the difference between the original input sample and the output of the quantizer.

1.1 Examples

In signal processing, quantization can occur when two signals are added, for eg:

$$(1.25) + (2.34500000199999) = 3.5950000012 \tag{1}$$

when the correct answer is 3.59500000199999, giving an error of approximately 10^{-12}. The error in multiplication operation can be deduced similarly.

$$Error = 10^{-12} \tag{2}$$

quantization noise also occurs on rounding operations such as :

$$round(12.34544567) = 12.3454457 \tag{3}$$

and for truncation of numbers to accommodate numbers in limited precision:

$$truncate(13.456) = 13 \tag{4}$$

Some other examples with respect to quantization in image processing are given in [1].

2 Modeling Quantization Noise

Let x be the input signal to the quantizer and x' be the output given by the quantizer. Let quantization noise be given by ρ which is the difference between the output (or quantized input) and the input. Quantization operation is a nonlinear operation, shown by the fact that the input-output waveform of any quantizer looks like a staircase (or some other nonlinear function depending on the quantizer function). Modeling of such a nonlinear ρ can be done in various ways but before that some basic concepts and theorems need to be kept in mind which are mentioned in brief in the following section.

2.1 Quantizing Theorem I and II

Widrow and Kollár, in [2] have described the statistical theory of quantization. It has been explained that the quantization noise in a digital control system or for that matter any other digital signal processing application could be modeled by some mathematical algorithm. Two theorems on quantization noise from [2] are summarized below:

The Quantizing Theorem I (QT I) states that if the characteristic function (CF), $\phi(t)$ (the fourier transform of the Probability Density Function (PDF) of a random variable) of the the quantizer input x is band limited such that:

$$\phi_x(t) = 0; |t| > \pi/q \tag{5}$$

where q is equal to 1 Least Significant Bit (LSB), then the CF of the input to the quantizer maybe derived uniquely from the CF of the output x', the same statement follows for PDF of x and x'. In essence, the theorem states that the PDF's of input and output of the quantizer are uniquely related to each other.

The Quantizing Theorem II (QT II) on the other hand puts forward an important and a stronger result that the moment of the quantized variable x' (the output of quantizer) is equal to the moment of the sum of the input and a uniformly distributed noise which has a zero mean and mean square equal to $\frac{q^2}{12}$.

The details of these theorems along with their proofs have been covered in detail in [2], the book on Quantization Noise.

2.2 PQN Model: Additive Noise Approximation

To model the quantization noise, ρ in a form that is easy to analyze, we take help of the principle from statistics which states that the PDF of the sum, when two statistically independent signals are added together, is equal to the convolution of the PDF's of the individual signals. This implies that the CF of the sum would be the product of the two individual CF's (due to the duality property of fourier transform). Hence, the PDF of quantizer output $x', f_{x'}(x)$, which is a discrete signal, is equal to the samples of the smooth PDF of the the input signal added with a uniform noise n, $f_{x+n}(x)$. These two PDF's correspond to each other in such a way that the moments of the two are equal when quantizing theorems QT I and QT II are satisfied. (This is the analogy between sampling and quantization, that sampling of a signal is discretization on time scale (X axis) and quantization is discretization of the pdf (Y axis)). In this case, the quantizer can be replaced by Pseudo Quantization Noise model (PQN), which is an additive uniform noise model. This replacement by additive noise holds to a very good approximation for gaussian signals for $\sigma >= q$ where σ is the standard deviation of the PDF of the signal. There are other kinds of modeling techniques for floating point quantization and various other complex modeling techniques. The book [2] is being referred to the reader for all such analysis.

2.3 Assumptions

The PQN model is a linear noise model and hence is easy to work with for quantization noise. Some assumptions made, for it to be valid are:

1. Two different quantization noise sources in a control system should be uncorrelated to each other and also to the signal.

2. The signal is always scaled such that the quantizers are not underloaded or overloaded.

3. Highly nonlinear behaviours and limit cycles are absent.

3 Quantization Noise: Sources, Estimation and Analysis

3.1 Analog to Digital Controller

In a mixed signal control system, an ADC is used to convert the output analog signal into digital for processing in the digital controller. ADC is a hardware which samples the analog signal values at a frequency which should be more than twice the sampling frequency of the signal (Nyquist sampling theorem). After sampling, the signal is discretized in its amplitude, which is called the quantization of signal. It is here that the signal is actually converted into discrete values which is then called the converted digital form of the signal. There are various ways to improve upon the design of an ADC for a better quantization noise performance.

Although, ADC performance improvements and rigorous noise analysis was out of scope for this project but basic simulations and a literature review was done to cover completely the quantization noise sources in the control system. The following section describes a general quantizer model which performs rounding operation and could be used as an ADC quantizer:

Quantizer Model: An Example For an example on quantization, [3], let us take a ramp input signal. A ramp is an input signal which is a straight line when drawn on a 2D plane, i.e., $y = x$.
A kind of quantizer model is given by:

$$Q(x) = k.q.sgn(x). \left\lfloor \left| \frac{x}{q} \right| + \frac{1}{2} \right\rfloor \tag{6}$$

where,
k is any arbitrary scaling factor
q is equal to 1 Least Significant Bit (LSB) which shows the resolution of the ADC and,
x is the input to which $Q(x)$ is the output.

The output of the quantizer will be a staircase waveform around the ramp input. The error between input and output can be approximated as (the difference between input and output) shown in Figure 1. Root Mean Square (RMS) of the error can then be derived to finally be able to calculate the SNR (Signal to Noise Ratio) of the ADC which would give an idea about the noise level due to the ADC.
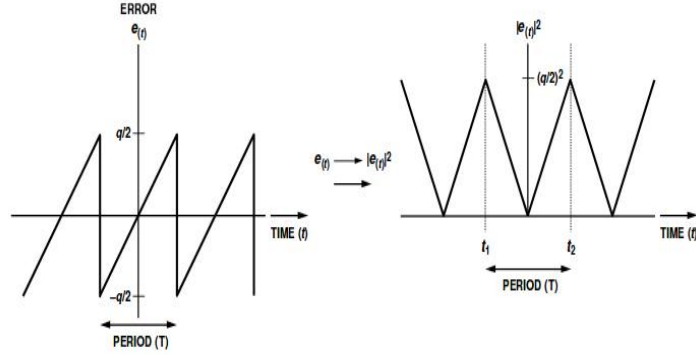
Figure 1: The approximated wave form for the quantization error taken from analog.com

To calculate the RMS we take a time interval $T = t_1 - t_2$ and write the equation of straight line in $y = mx + c$ form as:

$$e(t) - \frac{q}{2} = \left(\frac{\frac{q}{2} - \frac{-q}{2}}{t_2 - t_1} \right) . (t - t_2) \tag{7}$$

On simplyifying, we get,

$$e(t) = \left(\frac{q}{T} \right) . t + q. \left(\frac{1}{2} - \frac{t_2}{T} \right) \tag{8}$$

For the quantization error the RMS is given by:

$$e_{rms}^2 = \left| e^{\bar{2}}(t) \right| = \frac{1}{T} . \int_{t1}^{t2} |e(t)|^2 \, dt \tag{9}$$

Simplifying,

$$e_{rms}^2 = \frac{1}{T} . \int_{t1}^{t2} \left(\frac{q^2}{T^2} \right) . t^2 dt - \frac{1}{T} . \int_{t1}^{t2} q^2 . \left(\frac{(t_1 + t_2)^2}{4T^2} \right) dt \tag{10}$$

On solving we get (skipping the simple algebraic manipulations) that the RMS of the error is given by:

$$e_{rms} = \frac{q}{\sqrt{12}} \tag{11}$$

where q is the LSB of the quantizer representing the resolution of the taken ADC.

On simulation of the above in MATLAB and calculating the RMS, we observe that the RMS of the error is indeed equal to as given in equation 11. The script files for the same are available for reference on the Github repository at the URL mentioned in [4].

Now, to improve the noise performance and analysis for ADCs various techniques have been suggested. Mostly, the inner hardware of the ADC needs to be changed to improve noise performance. Also, dither signal is added to the input to randomize the input signal so that the noise and the input independence is followed to a better approximation. Some analysis on dithering signals[5] was also done in this study, but as mentioned previously, ADCs and ADC noise were not the primary focus of the project and hence were not studied in detail.

3.2 Digital Filters

Digital filters, also known as compensators in a control system, perform mathematical operations on the input to produce desired outputs. The design is represented in terms of transfer function, say H(z), where z is the z-transform variable. For more on z-transform and transfer fuctions, refer [6].

A transfer function is a unique representation for given position of poles and zeros on the z-plane. The implementation of a given transfer function can be done in infinitely different ways, depending on different state space descriptions or the so-called different SOS (second order sections) matrices. Various different filter structures are described in detail in [7]. The quantization noise, our primary concern in this project, also depends on the filter form realization. Intuitively, by different realization of filter (the filter structure), we basically mean different order in which the mathematical operations (additions, multiplications etc.) are performed by the filter.

Structure The following section details about some of the most commonly used filter structures along with their important features and trade-offs.

1. Direct Form I: The direct form I is an implementation of the difference equation describing the filter, as is. The roundoff noise for this realization has been found to be much more compared to other structures which has been proved in [7]. The advantage of having direct form I structure is in the applications where least hardware complexity is required. DF I leads to lowest chip area required for implementation on a Digital Signal Processor [8].

2. Direct Form II : The number of multiplications and delay registers are least for this form. [7] shows the quantization noise analysis for fixed point implementation. DF II form minimizes the coefficient sensitivities of the filter with respect to the quantization error which is proved in [8]. [10] showed for direct form filters that if poles and zeros are tightly clustered on the z-plane, then even small coefficient quantization errors may cause large shifts in the position of poles and zeros and hence changing the response of the filter and even tending to become unstable.

3. State space representation of digital filters is given by:

$$x_{k+1} = Ax_k + Bu_k \tag{12}$$
$$y_k = Ax_k + Du_k \tag{13}$$

On a given transformation T, to achieve low noise structure, we have

$$T_{k+1} = A'T_k + B'u_k \tag{14}$$
$$y_k = C'T_k + Du_k \tag{15}$$

where $A' = T^{-1}AT; B' = T^{-1}B; C'^t = C^tT$
To implement a filter as shown above, a time complexity of $\mathbb{O}((N+1)^2)$ is required

(direct form filters can be realised in linear time complexity) and hence these filters are not widely used. Moreover, as shown in [13], error feedback could be used to achieve super low noise forms in state variable representations. This is hard to implement practically as we don't have the error signal accessible to feed back. [14] shows how the error feedback helps in minimizing the error. Hence, state space representations of digital filters are used to achieve low noise forms but the time complexity trade off plays a major role in the choice of filter structure.

To estimate the quantization noise in digital filters, the output of the filter is subtracted from another output which is calculated using a higher level of precision than the original output. Other than the design of the filter, the precision at which it works plays a very important role. Two highly used precision types are described below:

Fixed Point Precision [7] describes in detail the noise in fixed-point precision calculations. Quantization noise in fixed point precision is both well researched and not relevant to this project because the calculations are being performed in floating point in the Advanced LIGO digital controllers, and hence the details are being skipped here and [7] is referred for all such analysis.

Floating Point Precision For floating point representations, a key point to keep in mind is that the quantization noise can no more be assumed to be independent of the input signal. In fact, the noise is directly dependent on what input is being given to the filter. Also, the noise cannot be assumed to be white in this representation and hence the noise analysis in floating point precision is a difficult task. An advantage due to the use of an exponent in the floating point representation is that the overflow condition is eliminated.

The book by Widrow and Kollár on quantization noise in Chapter 12, [2] looks at floating point quantization in depth and provides detailed floating-point noise analysis. Some direct formula to calculate quantization noise level in different filter structures have been given in [11]. The presentation [11] shows that noise is lower in a state space representation of a filter realized analogous to the analog biquad filter, compared to the Direct Form II (which is widely considered as the best filter structure). This "biquad filter" involves one more addition operation than the DF2 filter and hence is just a little more computationally complex than DF2. This filter structure having low noise has been referred to as a "biquad filter" in [11]. For floating point precision some very important results are available in the literature already existing. [12] mentions two theorems proving that the floating point quantization noise level is independent both of the ordering of sections in a cascade form and the ordering of poles and zeros between sections. The paper assumes that ordering of calculations and the poles and zeros within a section remain the same throughout.

3.3 Digital to Analog Converter

A DAC converts the digital signal from the controller to analog, to be fed to the actuators or other analog parts of the control systems. This conversion introduces quantization noise due

to the limited precision of the DAC hardware. Hence, if the precision of the DAC is p bits for an n-bit precise digital signal ($n > p$), the n-bit number is truncated to p-bit precision before giving it to the DAC for conversion to analog. This truncation operation (decrease in precision) leads to quantization noise called the DAC quantization noise. To estimate it, since both the truncated filter output and the original filter output are available in the digital controller code, a direct subtraction operation results in the noise. Commonly, n=64 while p=16 and hence the DAC is a major source of quantization noise in a control system.

4   LIGO

The Laser Interferometer Gravitational Wave Observatory (LIGO) [20] aims to detect gravitational waves (GW) [21] which would give us an opportunity to get access to completely new and exciting astronomical insights and scientific research. GWs are emitted by accelerating masses but due to their weak magnitudes only the most violent and massive events in the universe emit gravitational waves which could be detected by the LIGO detector. Some of these sources include binary stars or black holes orbiting around each other (Continuous GW), neutron star or black hole system merger (Inspiral GW), supernovae or gamma ray bursts (Burst GW) etc. Just like electromagnetic waves, gravitational waves have been predicted to have a frequency spectrum. The study of this frequency spectrum of gravitational waves along with many other new frontiers could lead to a completely different perception about science and astronomy. The existence of gravitational waves was first predicted by Einstein in his theory of general relativity in 1913. The detection (or the failure of it!) would help in establishing a stronger background in this field and would even prove (or disprove!, as the case maybe) Einstein's theory. Though the latter is almost impossible, as indirect detection of gravitational waves has alreaady been done but LIGO aims to directly detect the gravitational waves. GWs carry unique information about the most violent events in the universe, more importantly the events which can't be accessed by electromagnetic waves. Hence, gravitational waves will usher in a new area of astronomy as all of our astronomy in the past has been based on EM waves. The detection of GWs is bound to answer some of the most sought after questions in physics such as the formation of black holes, behaviour of matter under extreme temperature and pressure and many more.

4.1   Basic Working Principle

The LIGO setup consists of a laser interferometer [22] which is at the heart of detection of the gravitational waves. Gravitational waves have the property of stretching and squeezing the space-time through which they pass. This stretch and squeeze in the space-time changes the distance (length of the interferometer arm) between two points. This displacement produced is measured as strain, which transforms into the constructive or destructive interference between the original wave and the reflected laser light (which is stretched & squeezed in space-time by presence of a gravitational wave). To achieve reflection, a mirror is kept at the other end of the interferometer in both of its perpendicular arms [22]. There are many controllers which are used to control and suppress the mirror motion. [23] describes why movement of the mirrors and hence the motion control is necessary. [22] details more on the working principle of the LIGO interferometer.

4.2 LIGO and Quantization Noise

The problem of quantization noise is important to be solved for LIGO because it may be possible that at some places in the digital controllers the noise level might be so high that the signal wouldn't be detected leading to some very important signals being suppressed under the noise.

5 Advanced LIGO

Advanced Laser Interferometer Gravitational Wave Observatory (aLIGO) is an upgrade over the initial LIGO (iLIGO)[20] setup. Many different kinds of upgrades have been incorporated in the Adanced LIGO. The observations using the upgraded Advanced LIGO are currently underway at Hanford and Livingston. Some major upgrades in the Advanced LIGO have been focused on the seismic isolation of the mirrors using quadruple pendulum system compared to the single pendulum system in iLIGO. The active seismic isolation system using feedforward control also underwent many upgrades. The aLIGO uses increased laser power and signal recycling compared to the initial setup. All kinds of upgrades mean that there are many more degrees of freedom to control and hence the digital controller is much more complex in the aLIGO compared to the iLIGO.

5.1 Digital Filters

aLIGO digital filters are designed using the custom-made software called Foton.

About Foton The Foton is a software designed at LIGO which enables one to design a filter according to the specifications. It has a graphical user interface (GUI) which provides for various types of design methods such as the zero-pole-gain (ZPK) method or design using bode plots among others. Any user can easily design a filter using this GUI and then can save the design in a text file in a format which can be read by simple MATLAB / C codes, as per the needs. These text files can be read by the Foton software as well. Hence, whenever the need arises, a filter can be looked at using the Foton software by mentioning the file name in which the filter bank name exists. In this way, the thousands of digital filters have all been documented in these text files using the Foton software.

The foton filter design procedure and GUI is a lot like MATLAB's filter design and analysis tool (FDA Tool) in the signal processing toolbox.

The Foton software provides a complete analysis of a digital filter. For any given filter, its step response, impulse response and ramp response can be visualized. Also, various other parameters such as pole and zero location, transfer function etc. are available. For designing a filter, some common filter designs such as the butterworth, chebyshev and elliptical filter designs are available for use and modifications.

5.2 LIGO Filter Design

For the Advanced LIGO digital controllers the filter design was upgraded to a low noise form, in iLIGO, previously the digital filters were implemented in the direct form II (DF2) structures pertaining to its lower number of additions and multiplications. But, as described above, state-space representations lead to low noise forms, increasing the computational time in the process. [11] went on this line and suggested the use of a kind of biquad filter derived from state-space representation but only using one addition extra compared to DF2, in the process. The noise analysis shown in [11] proves that this indeed is a better choice for the digital filter structure as it provides for great increase in the SNR compared to the computation time penalty suffered. The aLIGO upgrade changed all digital filters from DF2 form to the low noise form as suggested by [11].

5.3 LIGO DAC

The aLIGO DAC is a 18-bit DAC made using a 16-bit and a 2-bit DAC together. These DACs work on integer 18-bit precision hence the floating values which are 64 bit precise are truncated to 18-bit precise integers. The quantization error during this operation is a big problem for various kinds of filter designs of aLIGO and hence analysis and mitigation techniques are important.

6 Part I: Filter Noise

6.1 Estimation

Section 3 gave an overview of quantization noise analysis in general and for digital filters with varying structures and precision levels. [24] implemented noise estimation for digital filters on these lines, by taking the difference between the single precision output and the double precision signal output. The double precision output noise is so less compared to the single precision noise that the difference output can be estimated to be equal to single precision quantization error. Since, all digital filters are already implemented in double precision in the LIGO setup, single precision noise is of no use. To obtain double precision noise, in [24], an extrapolation is done to estimate the approximate quantization noise occurring in digital filters implemented in double precision. The approximation method has been described by Denis in [25]. An empirically obtained factor equal to 10^{-7} is multiplied to the quantization noise values obtained for single precision. But, as is obvious that this is not the best way to achieve the estimation and isn't very foolproof.

So, this project improves upon the already existent technique by calculating the same output using the long double precision which is better than double for most [26] compilers, and then subtracting the two outputs, which would directly result in the quantization noise occurring in double precision filter implementation.

6.2 Analysis

For digital filter noise analysis, the following improvements were made to the previously existing analysis tool developed by Denis [24].

1. Accurate noise calculation (See section 6.1)

2. SNR warning message(s) on screen and SNR plots were incorporated in the new code.

3. Faster code: Takes far less time compared to previous implementation since the 32 second sampling time is skipped for the filters which are switched on but the input to them is all zero. In the long run of checking thousands of digital filters, saving 32 seconds on all the filters with zero data completes the analysis much faster.

One complete software tool was developed by incorporating all the changes and improvements described above which would automatically analyze quantization noise for all the digital filters. The software tool is a MATLAB function which can be called without giving any arguments. Once called, it starts checking all the filters one by one on its own until the end of the list. So, basically analyzing all the digital filters is just one click away. The list that this function looks into is a list of filter bank module names mentioned in the Foton's description of the filters.

NDS servers were used to login remotely to the sites to analyze the digital filters of the aLIGO sites at Livingston and Hanford. Proper log files were generated for errors along with the PSD plots which were saved to the disk for future reference. The complete saved collection of vector graphics of the plots is available for reference at [27].

6.3 Results

Testing at the 40m interferometer prototype   The tool developed was run to check all the digital filters implemented in the 40m interferometer at Caltech. There are far less number of digital filters implemented in the 40m compared to the aLIGO sites. All the filters that were switched on and working at the time of testing had a signal to noise ratio (SNR) of more than 10^3 atleast for the whole frequency range. A plot of one such filter is shown in the following figure.
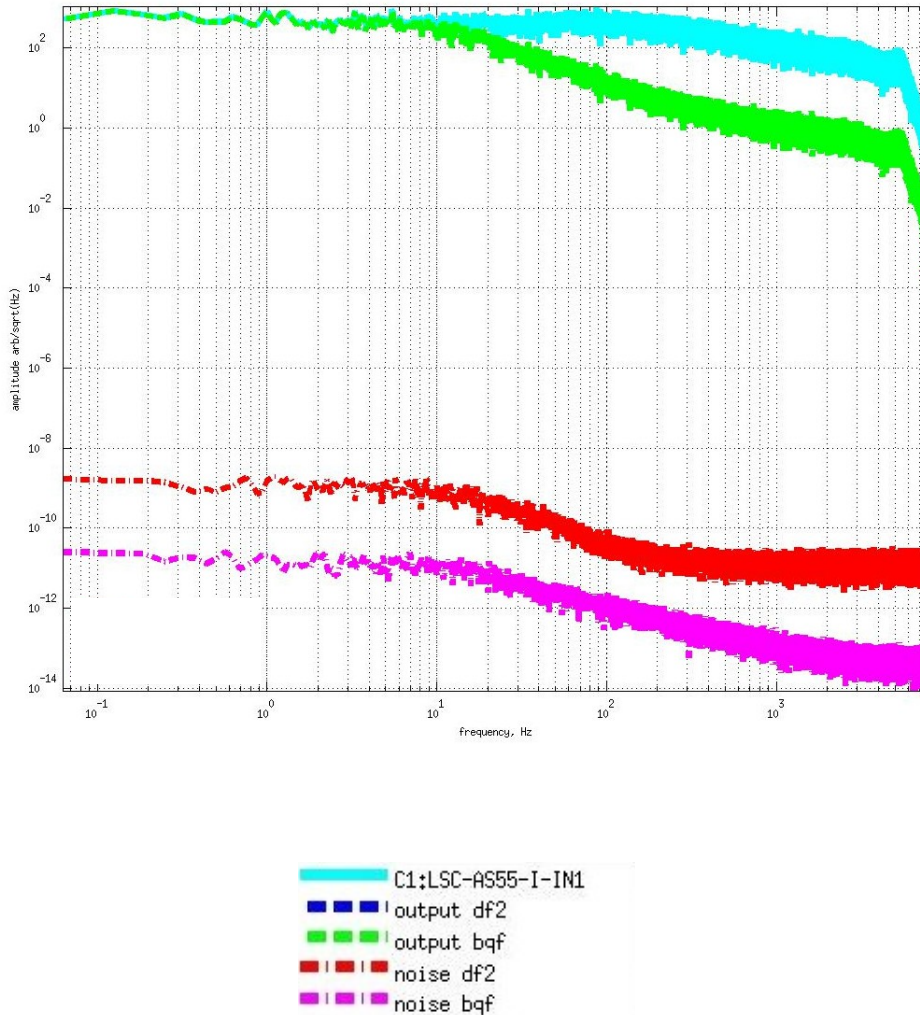
Figure 2: Analysis for Caltech 40m LSC ASC filter

Testing at aLIGO sites Some results are listed below:

1. For the low noise form "biquad filter" realization, the SNR for more than 90% of the filters analyzed was observed to be more than 10^{10} for all frequencies. Even for others, SNR was atleast 10^3. This type of response was observed for the complete frequency range except at very high frequencies (near the Nyquist frequency) where the output of the filter is usually rolled off to lower magnitudes. Some assorted results for some of the filters from both aLIGO sites are shown in the figure 3 and the ones following it.
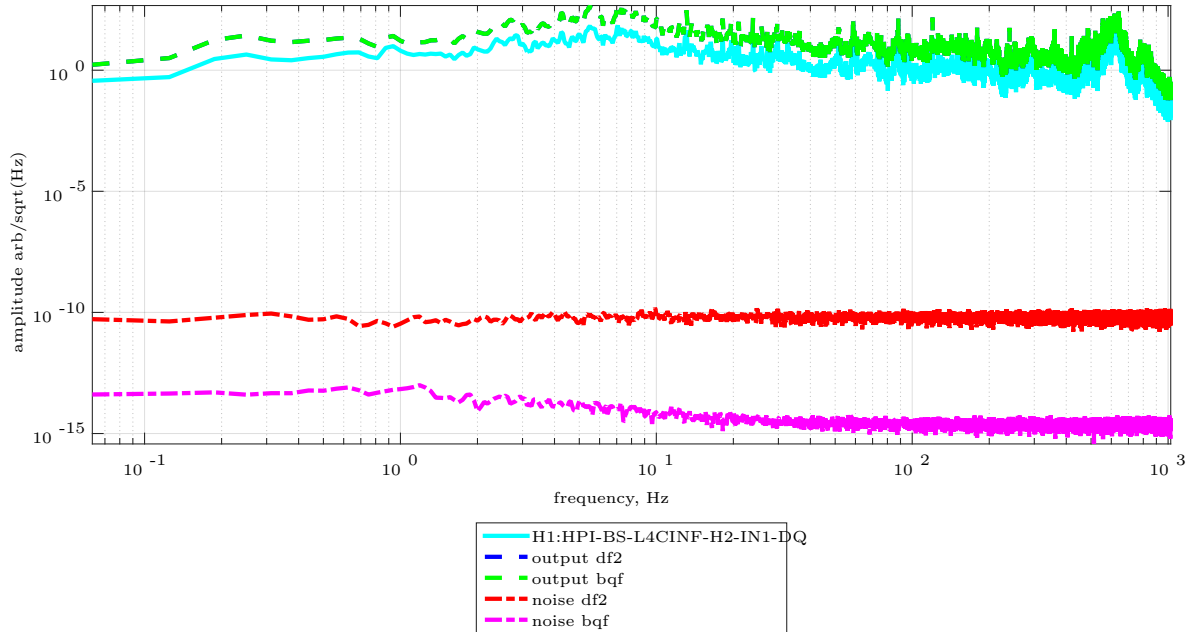
Figure 3: Analysis for Hanford HPI BS filter

The filter shown in figure 3 is representative of a collection of hundreds of filters which do not execute anything other than a gain multiplication operation. As expected, the quantization noise level is more than 10^{12} orders of magnitude below the output. The filter is a part of HPI subsystem of the aLIGO detector at Hanford.
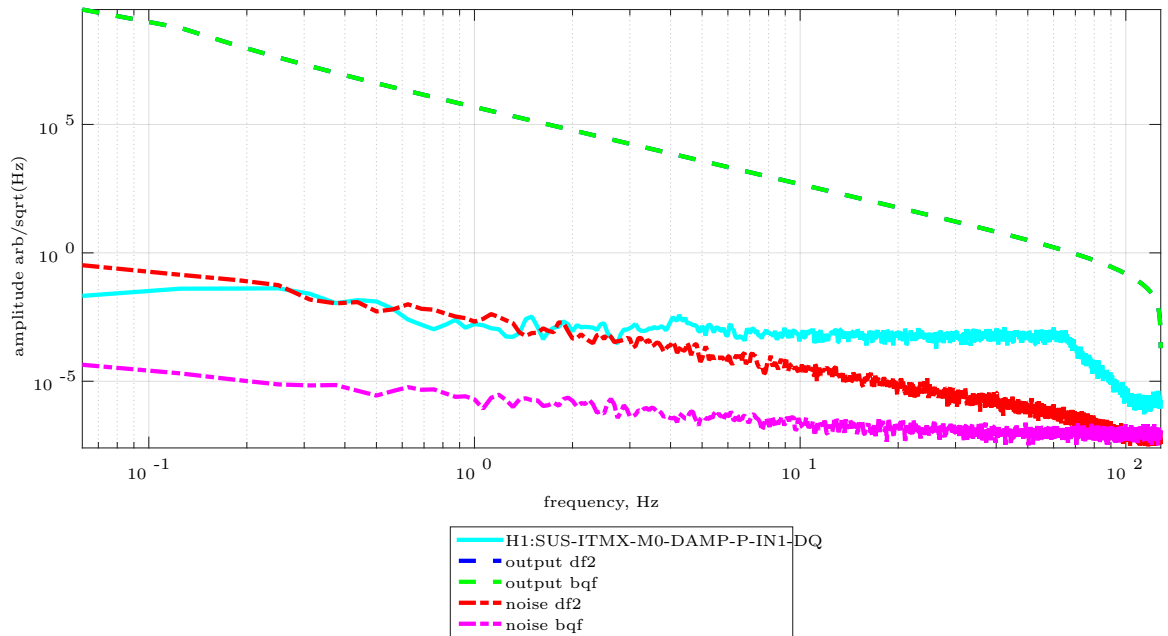
Figure 4: Analysis for Hanford SUS ITMX filter

Suspension subsytem's digital controller consists of some complex filters and one of those has been shown in figure 4. As mentioned above, the SNR falls down steeply as the output of the filter is rolled off at higher frequencies. But at lower frequencies, we observe that the SNR is maintained at more than atleast 10^6, which is good. The following figure shows the quantization noise level for the same filter at Livingston site.
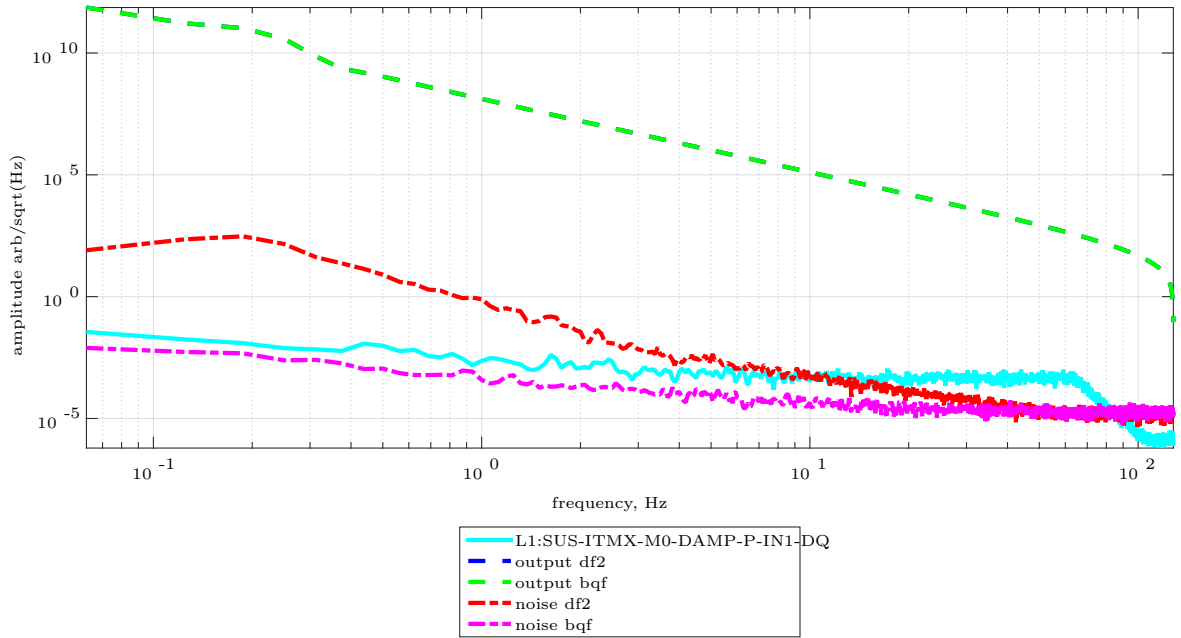
Figure 5: Analysis for Livingston SUS ITMX filter

2. For a few filters, the DF2 quantization noise level is even above the output for a range of frequencies. Though this is an alarming observation, but this does not affect the aLIGO digital control system at all because all digital filters have been implemented in the low noise form. This result further strengthens and proves the results shown by [11] and others. For these filters, the low noise form quantization noise level is considerably lower and the filter is safe. Some results are shown in the figure 6 and the ones following it.
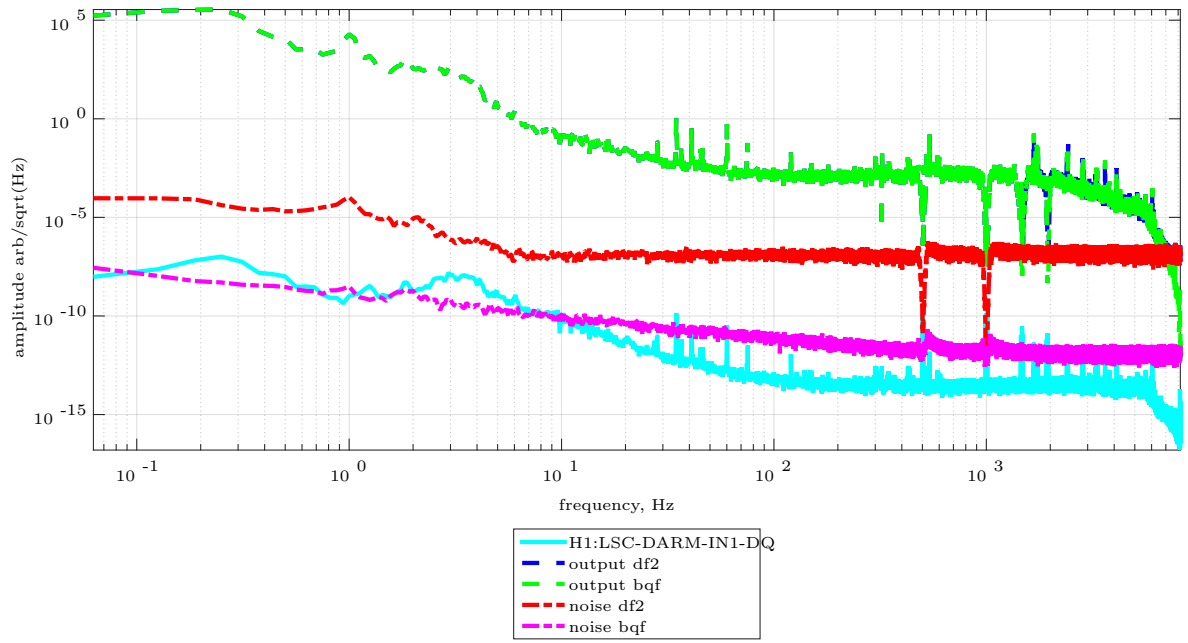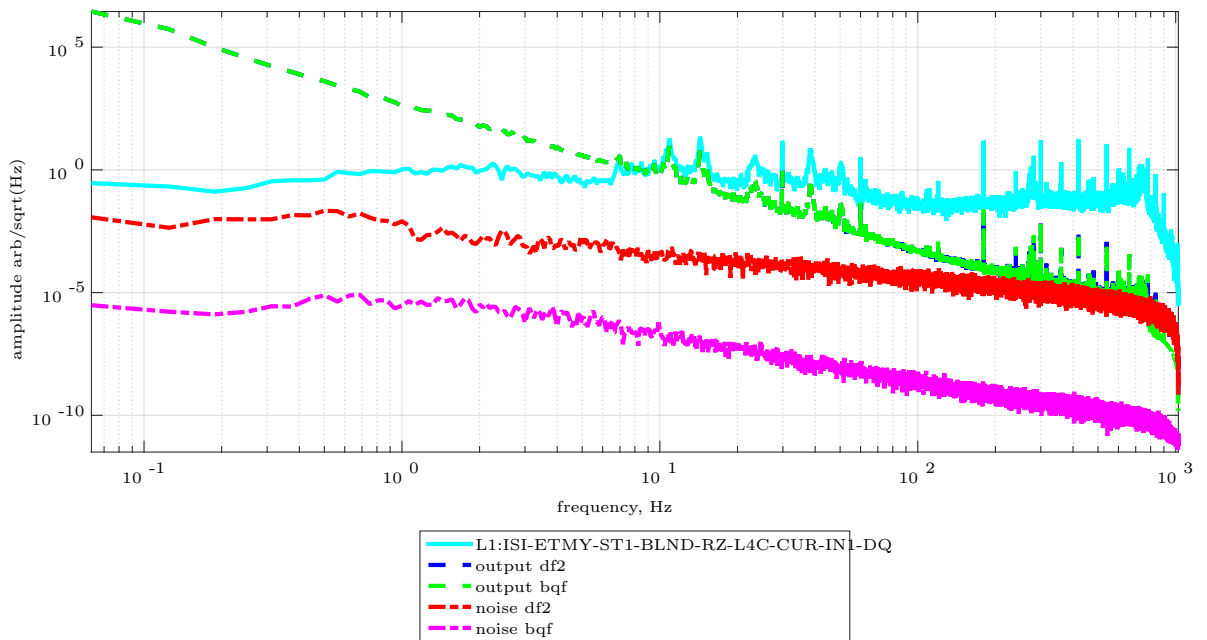
Figure 6: Analysis for Hanford LSC DARM filter

Figure 7: Analysis for Livingston ISI ETMY filter

3. For some filters, the DF2 performs equally well as the LNF. This result is shown in figure 8 and the one following it. As is clear from the figures, these filters are only performing a gain operation which results in the similar performance of the two kinds of filter structures.
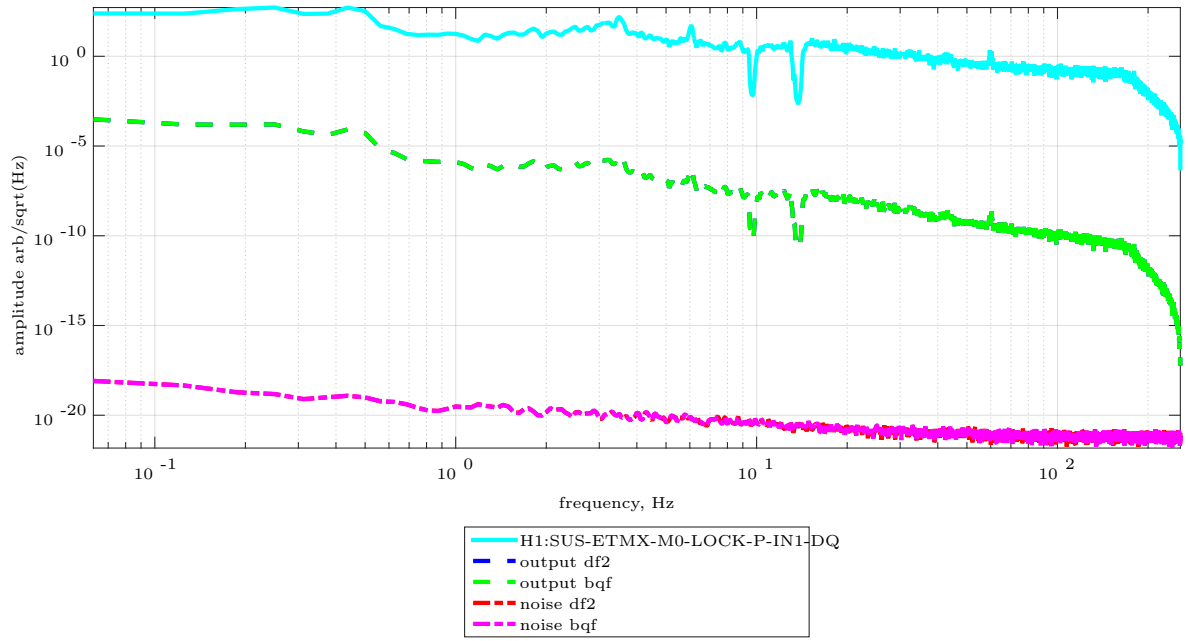
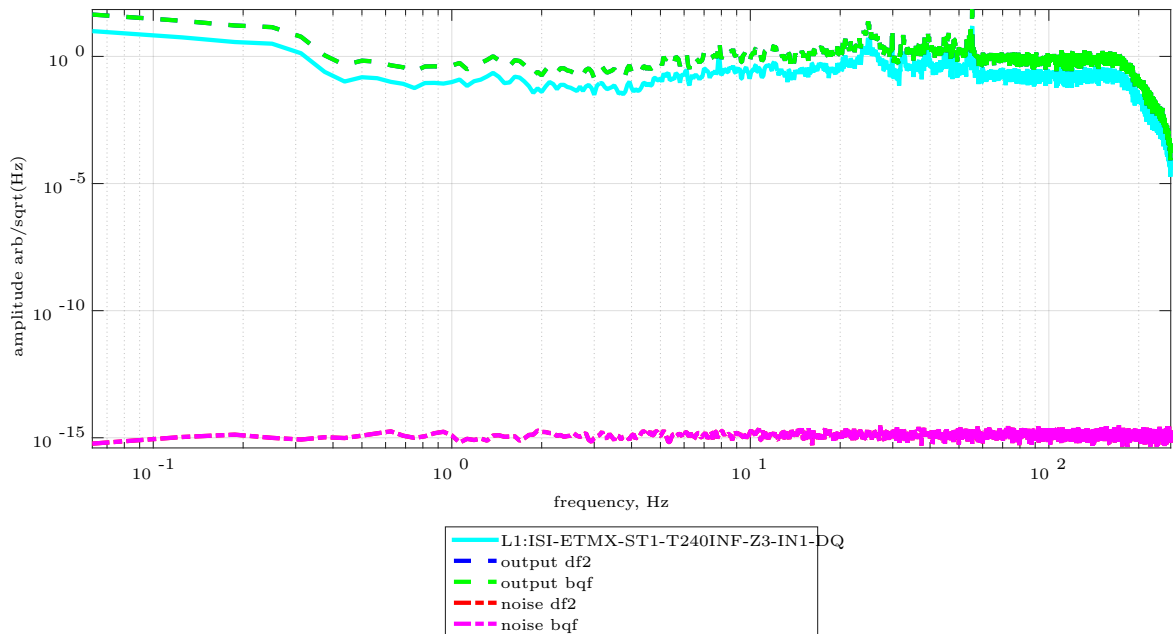Figure 8: Analysis for Hanford SUS ETMX filter

Figure 9: Analysis for Livingston ISI ETMX filter

7   Limitations

Some limitations of the analysis are:

1. Only data acquisition (DAQ) channels: As all work is being done remotely, hence only the channels for which the data is stored on a hard disk is available. So, only the channels ending in _DQ have been analyzed.

2. Only the filter channels which have been recorded to the disk were analyzed.

3. Various filters in the digital controller build up their outputs slowly via the history values, this project doesn't detect quantization noise problems for such cases as it has been assumed here that initially all data is zero.

4. Memory size being used by the software tool is higher than before due to the file handling mechanisms involved instead of mex for C-MATLAB interaction.

8   Part II : DAC Noise

8.1   Estimation and Analysis

DAC noise is easy to estimate in the code as access to both the input and the quantized output is available. A block diagram showing the measurement procedure is given in figure 10.
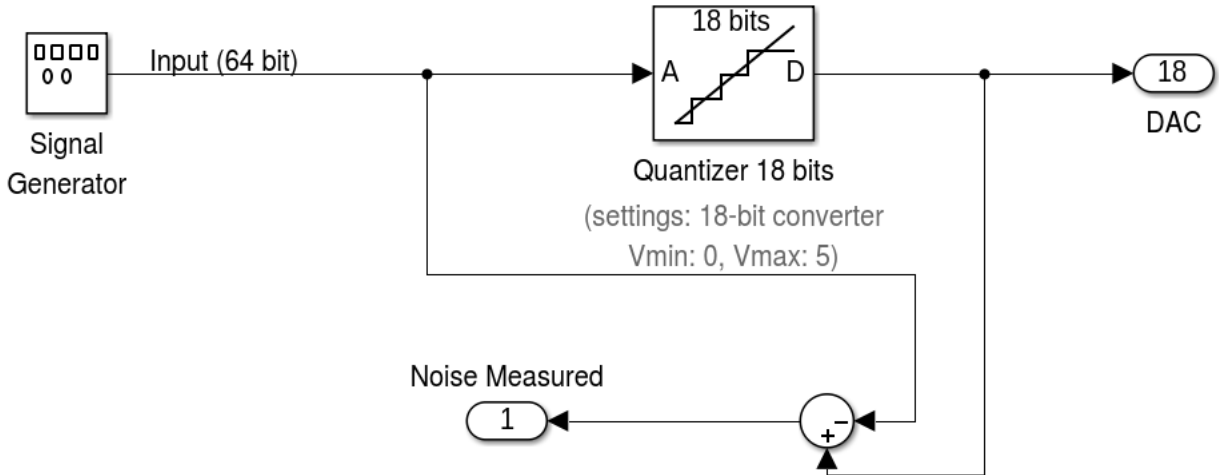

Figure 10: DAC noise measurement

The difference between the two gives us the noise. This noise can then be analyzed in the frequency domain, like before, by plotting the PSD for the input, output and the noise.
As described earlier, that the DAC noise is a major limiting factor in the digital controller as it truncates from a highly precise 64 bits to 18 bit integer precision. This fact was observed in the analysis, as the DAC noise floor was observed to be well above the digital filter noise floor.
The code developed for DAC noise analysis and visualization on MATLAB is available on [4] and can be used for DAC noise analysis for all kinds of signals.

8.2   Results

The already known or rather predicted DAC noise problem, was realized again through the DAC noise analysis done in this project. The DAC noise floor is well above the output level and is limiting the signals for a wide range of frequencies for some signals. A good DAC denoising technique is needed to prevent such high noise levels. One such technique called noise shaping has been described in the following sections.

9   Noise Shaping

Noise Shaping is a technique used to modify the frequency spectrum of the error signal in such a way that the noise power of the spectrum is more in the undesirable frequency band which leads to a higher SNR in the desirable frequency band. This technique is perfect for

our implementation as the frequency band of interest for gravitational wave detection is fixed and is < 100Hz. Hence, with the same DAC hardware, very low noise levels in this frequency band can be achieved using noise shaping. An algorithm was developed to implement noise shaping technique in the DAC code.

9.1   Algorithm

A general example   Consider the given system (see figure 11) where the input (denoted by x) which is double precision, is fed into the quantizer. The quantizer rounds it off and feeds to the DAC according to the DAC's resolution, hence incurring quantization error. To shape the quantization error, e (quantization error) is fed back as shown. The output of the quantizer is denoted by x' and the following equations explain how noise shaping is achieved.
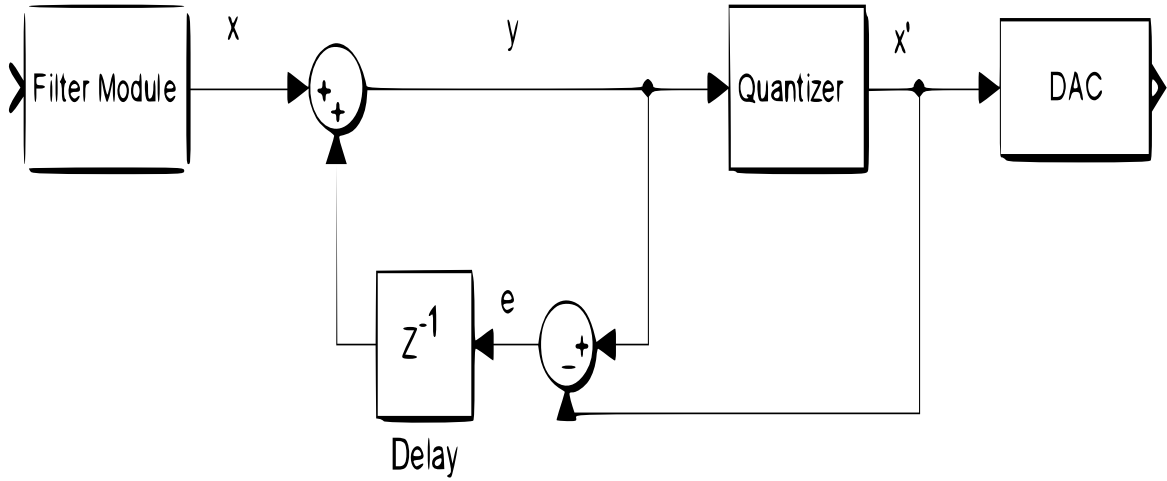

Figure 11: An explainatory block diagram example for noise shaping in DAC

From the block diagram (in time domain), we have,

$$y[n] = x[n] + e[n-1] \tag{16}$$
$$e[n] = y[n] - x'[n] \tag{17}$$

We have,

$$x'[n] = y[n] - e[n] \tag{18}$$

Now taking Z-Transform, we have

$$Y(z) = X(z) + \frac{E(z)}{z} \tag{19}$$
$$X'(z) = Y(z) - E(z) \tag{20}$$

Now from equations 19 and 20, we get the final result which shows how quantization error is added to the input to form the output x', and since we have fed the error back we will

see how the quantization noise is now shaped according to the transfer function we chose (in this case $1/z$):

$$X'(z) = X(z) + E(z)(1 - \frac{1}{z}) \tag{21}$$

The noise is shaped by a factor of $\frac{z-1}{z}$ which has a zero at $z = 1$ and a pole at $z = 0$. This means that at the frequency corresponding to $z = 0$, the gain will be high and at frequency corresponding to $z = 1$ the gain will be low since a zero is occurring there. In essence, we have a one-pole digital filter in front of us resulting due to the feedback of the error. In this way, noise shaping can be achieved to increase the SNR in the desired frequency band by choosing any arbitrary shaping filter (with some restrictions).

This was implemented in a code in MATLAB for simulation. The plot in figure 12 shows the result which verifies the theoretical fact that the noise should have a high pass shape.
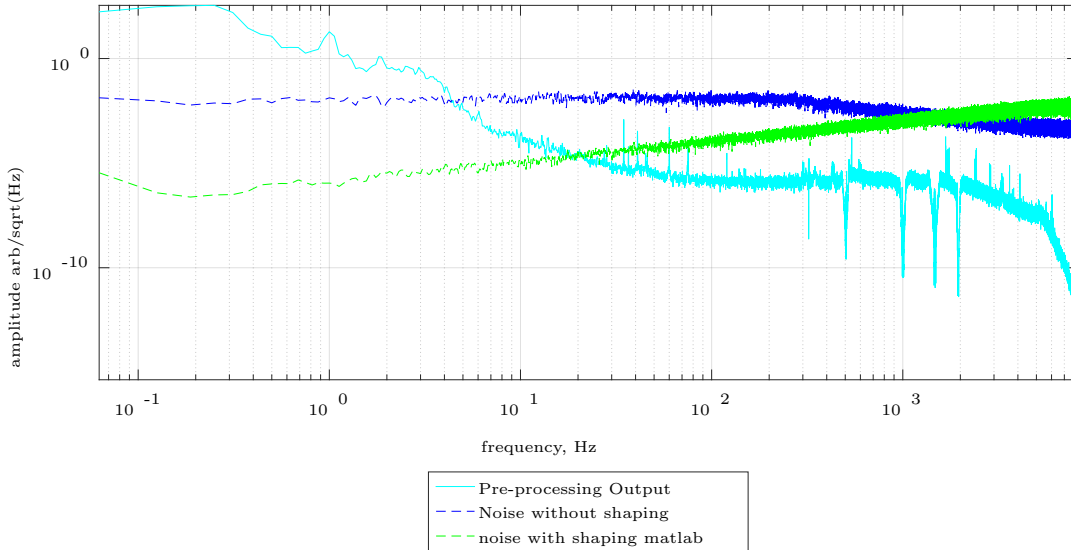

Figure 12: Error feedback leading to high pass shaped noise: MATLAB simulation

9.2 Generalized Noise Shaping Algorithm for aLIGO DAC

Using the above mentioned concept, a generalized noise shaping algorithm was developed and simulated on MATLAB. This noise shaping algorithm was designed such that it provided the elegance to the user by providing the option to design any arbitrary noise shaping filter to achieve any arbitrary shape of the noise [28]. So, for example, if a designer fancies the noise to be very low for a particular notch design in the filter design, then the noise shaping filter can be designed to be a notch filter of that frequency. This would lead to noise being shaped such that the noise would be very low at the given notch frequency. In this way, any arbitrary shape could be given to the noise, which is a very useful result and something

which could be very helpful in various ways.

The way this is achieved is shown below: After the delay element in the above description, if a H_{shaper} filter is put, then following similar analysis, an arbitrary noise shape could be achieved. The block diagram for the same is shown in the figure 13.
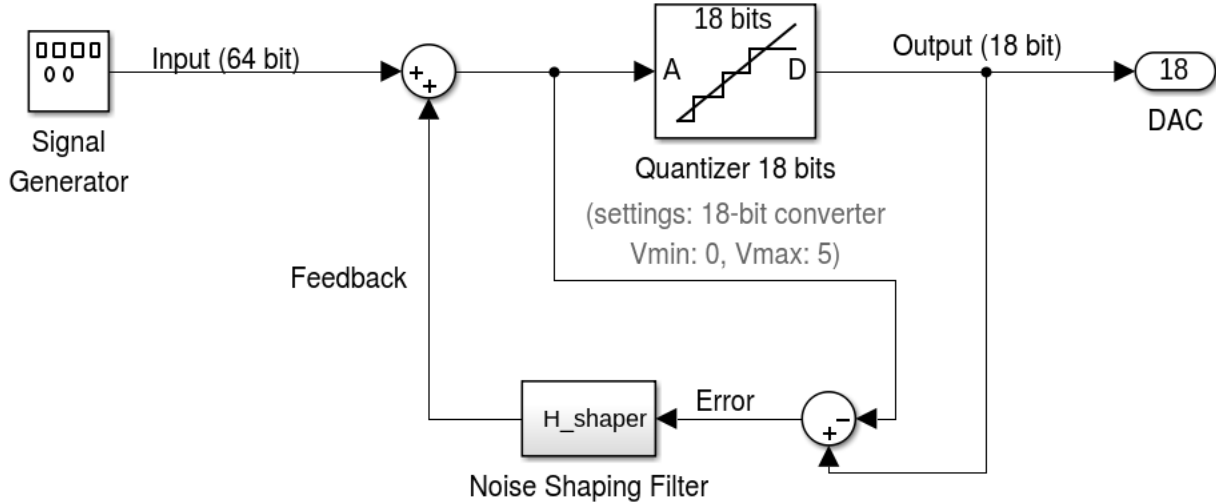

Figure 13: Generalized noise shaping block diagram

Following the analysis shown above, the following expression results:

$$X(z) = X(z) + E(z)(-1 + H_{shaper}(z)) \tag{22}$$

$H_{shaper}(z)$ allows for arbitrary filter design, and hence arbitrary noise shaping of the quantization noise.

9.3 Applications

General Applications   Noise shaping is a widely used technique in Digital Signal Processing (DSP) applications. One of the fields making use of noise shaping extensively is audio processing. The audio industry has progressed greatly, partly due to such DSP techniques. In digital audio, it is applied as a bit-reduction scheme. The quantization is spread according to the frequencies ear is more sensitive to. This leads in more pleasant sounding audio as noise is removed from it.

Apart from wide applications in the audio industry, the noise shaping technique is being used more and more in the modern ADCs. Along with this, the noise shaping technique is used in video and image processing as well. In these applications, the noise shaping is done in combination with dither. One such description for image processing is given in [29].

Possible applications at LIGO   As mentioned before, the noise shaping technique could be a very useful application for LIGO digital controllers as well because the GW detection is done at low frequencies, i.e. below 100 Hz. Hence, the DAC noise can be pushed out of

this band using the noise shaping technique. Also, the generalized noise shaping technique presented could find very wide and varied applications especially in freedom of filter design.

9.4   Simulation

The algorithm was implemented for simulation on MATLAB. The code developed was tested for various different filter designs for H_{shaper}. One such, notch filter design is shown in figure 14. The elegancy of the method is visible in the noise spectrum which is almost zero for the given notch frequency of the noise shaping filter. In this way, the noise can be shaped as desired. Also is interesting to observe in the given spectra is the noise level without any noise shaping. The comparison between the two spectra clearly accounts for the fact that the overall noise level is more when it is shaped.
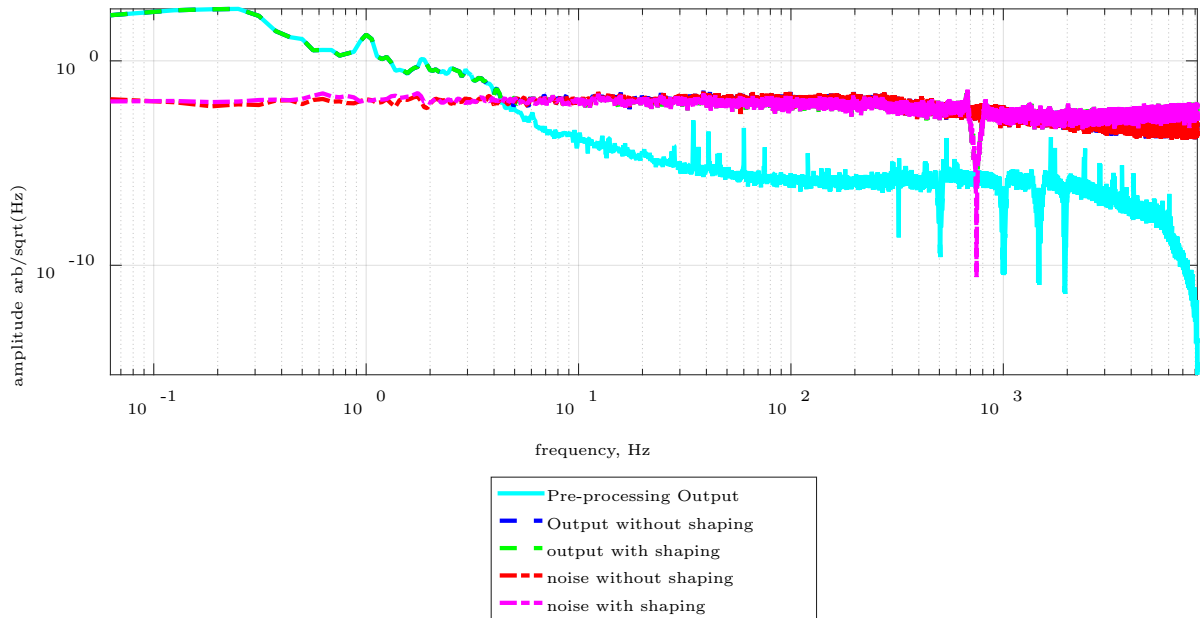

Figure 14: Notch shaped quantization noise : Simulation in MATLAB

A more applicable filter would be a high pass filter for implementation in LIGO. The figure 15 shows how the quantization noise level is very low for lower frequencies which is compensated by very high noise level in the higher frequency band.
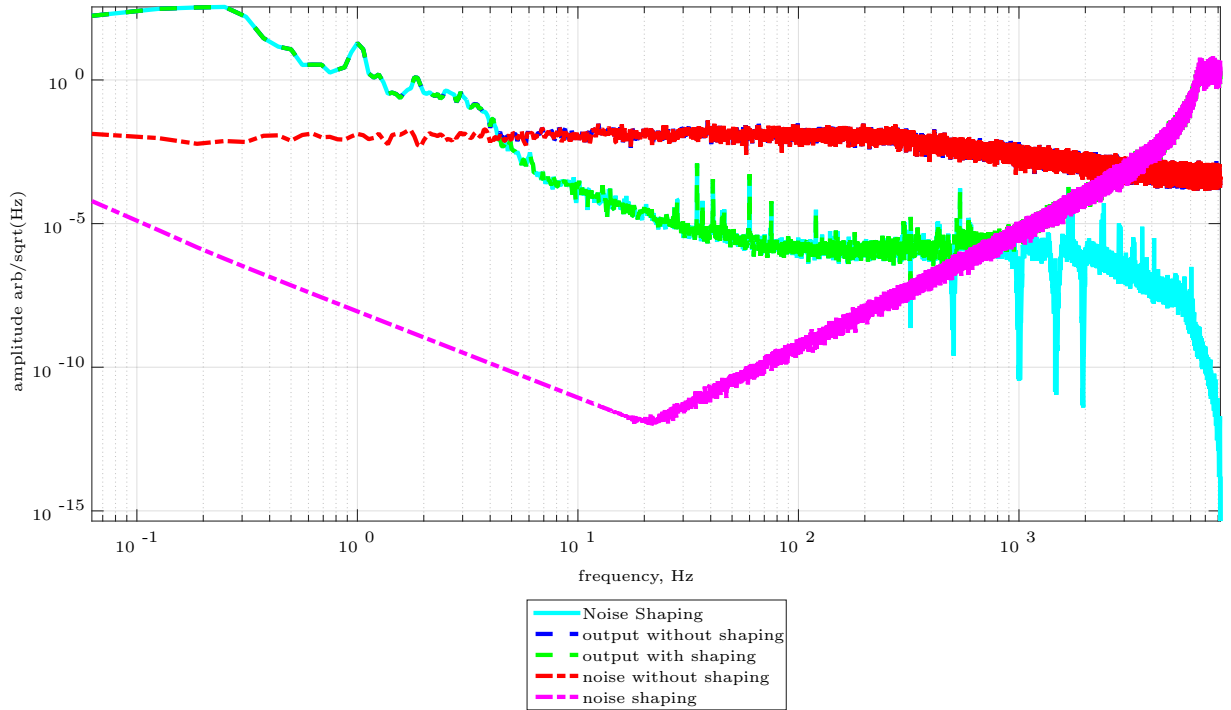
Figure 15: High pass shaped quantization noise which is good for use in GW detection : Simulation in MATLAB

10   Conclusions and Future Work

10.1   Part I: Digital Filters

A favorable conclusion that can be drawn out of the project is that majority of the filters (¿90%) had a signal to noise ratio of more than 10^{10} for all frequencies. Even for other filters among those that were checked (also see section 7) an SNR of more than 10^3 (atleast) was observed for all frequencies except near the nyquist frequency where the output is usually rolled off to zero.

The complete analysis is available at [27]. It has all the plots for the thousands of filters analyzed using the software tool. There still remains a lot to be done with respect to achieving a complete digital controller analysis as already mentioned in the limitations (in section 7). Hence, the future work would demand a complete in-depth analysis of all filters and all kinds of signals in the controller (even those which are not the final inputs and outputs) along with a solution to the filter history problem.

10.2   Part II: DAC

The DAC noise problem was a known fact, and the denoising algorithm in the name of noise shaping presented could be a really helpful technique not just for DAC denoising but also for various other kinds of noise shaping requirements that the filter designer might want. The future work requires a complete development and implementation of the noise shaping technique in the real time code to reap advantages of the results shown in the simulations.

References

[1] Tutorials Point, http://www.tutorialspoint.com/dip/Concept_of_Quantization.htm

[2] Widrow and Kollár, *Quantization Noise* http://www.mit.bme.hu/books/quantization

[3] Quantization of Signals, http://en.wikipedia.org/wiki/Quantization_(signal_processing)

[4] DAC Denoising http://github.com/rxa254/DACdenoising/

[5] Ayush Pandey, *Progress Report I : Quantization Noise in Digital Control Systems*

[6] Z-Transform, http://www.ele.uri.edu/Courses/ele541/tutorials/ztransforms/

[7] Oppenheim and Schafer, *Discrete-Time Signal Processing*

[8] S. Rahmanian, *An Optimal Structure for Implementation of Digital Filters*

[9] T. L. Chang, *A Unified Analysis of Roundoff Noise Reduction in Digital Filters*

[10] Kaiser, *An approach to implement Digital Filters*

[11] Matthew Evans, *Digital Filter Noise*

[12] Bing Zeng, *Analysis of Floating-Point Roundoff errors using dummy multiplier coefficient sensitivities*

[13] Chang, *Comparison of Roundoff Noise variances in several low round off noise digital filter structures*

[14] C T Mullis, *Roundoff Noise in digital filters: Frequency transformations and invariants*

[15] Bomar, *Roundoff Noise Analysis of State-Space Digital Filters implemented on Floating-Point Digital Signal Processors*

[16] Dehner, *Noise Optimized Filter Design* https://dx.doi.org/10.1016/s0165-1684(03)00075-6

[17] Michael Cerna and Audrey F. Harvey (2000) *The Fundamentals of FFT-Based Signal Analysis and Measurement*

[18] Jonah Kanner https://dcc.ligo.org/LIGO-G1500862

[19] Welch (1967), *The Use of Fast Fourier Transform for the Estimation of Power Spectra: A Method Based on Time Averaging Over Short, Modified Periodograms* , IEEE Transactions on Audio Electroacoustics, AU-15, 7073.

[20] B. P. Abott et. al., *LIGO: the laser Interferometer Gravitational Wave Observatory* http://dx.doi.org/10.1088/0034-4885/72/7/076901

[21] *On Gravitational Wave Detection:* https://nodus.ligo.caltech.edu:30889/wiki/doku.php?id=gw_detection_101_for_surf

[22] Frederick J Raab, *Overview of LIGO Instrumentation*

[23] L. Carbone et al., *Sensors and Actuators for the Advanced LIGO Mirror Suspensions* http://arxiv.org/pdf/1205.5643v1/

[24] Denis Martynov, *Checking Digital System* http://github.com/denismartynov/quantization http://nodus.ligo.caltech.edu:8080/40m/?mode=full&attach=1&reverse=0&npp=510&Author=den&subtext=digital

[25] Denis Martynov, http://nodus.ligo.caltech.edu:8080/40m/?mode=full&attach=1&reverse=0&npp=510&Author=den&subtext=digital

[26] Wikipedia : Long Double Precision, https://en.wikipedia.org/wiki/Long_double

[27] Ayush Pandey, *A complete collection of all plots for all digital filters at Hanford and Livingston* https://dcc.ligo.org/T1500351-v1

[28] Markus Nentwig, http://www.dsprelated.com/showarticle/184.php

[29] Cameron Nicklaus Christou,*Optimal Dither and Noise Shaping in Image Processing*