ENPH 454 Advanced Engineering Physics Design Project-Final Report

Open Air Active Noise Control

Department of Physics, Engineering Physics & Astronomy

Queen’s University, Kingston, ON, Canada

Date of submission: Dec 12, 2022

Professor Jordan Morelli

Connor Legg

David Puddy

Declan Bracken

Edwin Feng

Matthew Boxer

Michael Butler

Abstract

Presented in this report is the design and testing of an open-air active noise control system. The initial project scope aimed to use an FxLMS algorithm to reduce single tone, multitone, white noise, and a sine swept signal by at least 1 decibel. Using a closed loop feedback system between an error microphone and anti-noise speaker, the project aimed to reach steady state noise control in under 1 second. This report successfully presents single tone noise control for a 528 Hz signal by up to -28 dB (voltage scale) and multitone control with 528 and 392 Hz signals by up to -10.9 dB and -1.9 dB respectively. The final product reaches steady state within 10 seconds but could be improved with minor changes.

Contents

Table of Figures

Table Of Tables

Introduction

Motivation

The adverse health effects of noise pollution range from disturbance of rest and sleep to noise-induced hearing impairment, which is globally "the most prevalent irreversible occupational hazard" and affects millions of people daily [1]. With an increase in urban living density, the number of residential homes exposed to noise pollution has significantly increased, from sources such as highway or airport traffic noise [2].

This raises the engineering question of "how can we effectively mitigate noise pollution in residential areas?" One avenue is investigating the architectural properties of these residential buildings. Many residential buildings are constructed with materials such as wood, brick, and drywall which have strong sound dampening properties [3]. Comparatively, glass windows are poor insulators of low frequency ambient noise, creating a pathway for noise pollution [4].

This project will focus on solving this engineering challenge, specifically investigating how noise can be reduced through windows as a pathway.

Key stakeholders have been identified as (1) Businesses that require quiet environments, (2) Users of noisy public spaces, (3) Individuals with sensory issues, and (4) Educational institutions.

Background Research

Conventional methods for noise controlling involve passive systems, including noise absorption or attenuation, like fiberglass ceiling panels or double-pane windows, others include noise baffles like sound-deflecting highway barriers [5]. However, while these are comparatively well explored, the team investigated an alternative to these solutions – active noise cancellation (ANC).

The premise of ANC was first investigated in 1933 by Paul Lueg, who patented the theory of using phase-advancing waves to cancel sinusoidal tones in ducts and inverting polarity to cancel sounds around a loudspeaker [6]. However, this remained theoretical at the time, given the lack of suitably sensitive equipment. From there advancements on the technology were completed by the Air Force Research Laboratory which focused on hearing conservation and protection for pilots, and later Bose released working prototypes for acoustic noise cancelling headphones [7].

Recent developments in noise control have investigated the use of piezoelectric devices operating as sensors and as actuators in active noise control systems. An example of such design includes the use of an array of piezoelectric actuators positioned on the inner surface of a jet engine inlet cylinder to provide an interfering noise field [8].

Scope

The time allocated for the project was 13 weeks, with a budget of $1000. Given these constraints it is necessary to appropriately scope out the project's mission and specific goals.

The project mission is to detect, analyze, and invert high nuisance outdoor noise to destructively reduce unwanted noise. Initially this was to be achieved by analyzing sounds incident on a closed window and generating a destructively interfering signal using a transducer. This was later realigned to focus on open-air noise cancellation, achieved through a microphone and speaker setup.

Below is a list of quantifiable and measurable objectives for the product to achieve. The success of which will be discussed in the Final Product Results section.

A minimum viable product will aim to:

1. Reduce the intensity of a single-tone signal in the frequency range of a human voice (50 – 3000 Hz) by a minimum 1dB, a rough estimate of the sound reduction provided by switching a single pane window to a double pane window [9]

A final product will aim to do the above, as well as:

1. Reduce the intensity of multi-tone signal, with frequencies taken in the frequency range of a human voice (50 – 3000 Hz), by a minimum 1dB
2. Reduce a sine swept signal ranging from 100Hz to 2000Hz by a minimum of 1dB
3. White noise cancellation by 1dB
4. Reach "steady state" within 1 second

Design Decisions

Types of Active Noise Control (ANC)

There are two main ANC methodologies, feedback (FB) and feedforward (FF) control. For feedback ANC, the microphone is placed behind the source and anti-noise speaker, picking up a superposition of both signals. This net signal is then processed by an algorithm that inverts the source noise's phase which is sent to the anti-noise speaker, ideally creating a silence zone at the microphone's location. Any failure of the system will result in noise recorded at the microphone which is why this noise is hence referred to as the error signal. This means that feedback systems have high response latency but can fine tune their operation for maximum cancellation of the source noise as the signal approaches the steady state.

For feedforward ANC, the microphone is placed between the noise source and anti-noise speaker, receiving only the source signal. The source signal is then inverted by the algorithm and sent to the anti-noise speaker to destructively interfere in the same way as feedback ANC. Feedforward ANC has the capability to adapt to changes in the source noise, before the system sends the anti-noise to the area of interest, allowing for time variant control. This does mean that feedforward systems cannot measure the noise at the area of interest, meaning miscalibration or other errors in processing cannot be detected. As such, feedforward ANC works best in controlled environments with fewer well defined noise sources. If the sources of noise are arranged too diversely the performance of a feedforward ANC deteriorates due to the various geometries and direction of the acoustic sources creating unknown dephasing between the reference microphone and area of interest [9].

The hybrid feedback-feedforward (HFF) adaptive system uses two microphones to measure both the source signal and error signal [10]. HFF is highly accurate, yet the system is more costly and complicated than the other two ANC methods. All ANC design options are summarized in Figure 1.

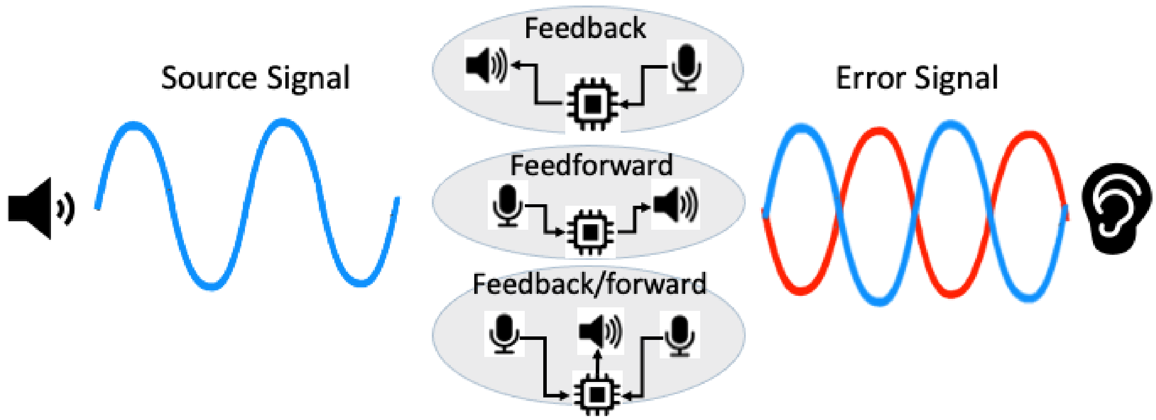

Figure 1: three types of ANC NN: feedback (FB), feedforward (FF), and hybrid Feedback-feedforward (HFF). Circuitry of all of which aim to produce deconstructive interfering waves to cancel unwanted acoustic signals at audience.

In-Pane and Open-Air ANC

Windows are the most significant source of unwanted noise for domestic residents [10]. As such, the initially proposed device would cancel vibrations using a window as the medium for sound propagation. Testing was performed with some small piezoelectric transducers using a speaker to vibrate the window, but measurements taken with an Arduino compared to an oscilloscope did not match because the transducers were not sensitive enough to make accurate detections.

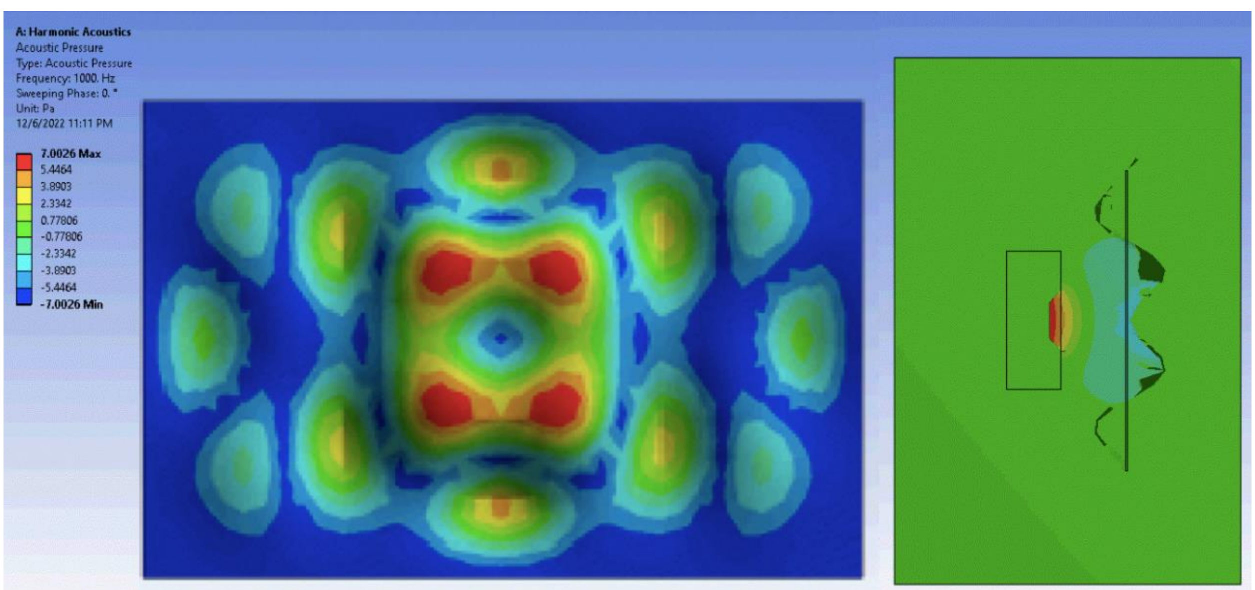

Figure 2: Ansys simulation of vibroacoustic waves at 1000 Hz coupling from a simple loudspeaker through air into a planar window (1/8-inch thickness) with fixed edges. An excitation frequency of 0.5 N/m^2 is applied for the loudspeaker.

To determine the feasibility of measuring the vibrations of a planar window, an Ansys simulation modelling the vibroacoustic coupling from soundwaves produced by a conical speaker unit into a window through the air was developed. This simulation was designed to determine the degree of deformation and mode shape of the window assuming clamped edges. The simulation results using excitation frequencies from 100 to 1000 Hz developed a maximum window displacement on the nanometer scale ($7\ nm$). This simulation proved that the piezoelectric transducers available in the lab would be insufficient for detecting such small vibrations. As such, the scope was adjusted to focus on open-air noise cancellation using microphones and speakers.

The FxLMS Algorithm

Modern ANC systems use digital signal processing algorithms to measure a disturbance signal and adaptively generate a destructively interfering output signal [11]. One of the most common algorithms is the Filtered-x Least Mean Squares (FxLMS), which uses a simple neural network for feedback ANC. The algorithm assumes one microphone measures the total noise, which is the error signal, while two speakers produce the disturbance and anti-noise signals. While the disturbance is on, a vector of length L holds the previous L error samples, which are multiplied by a set of weight values to generate the next anti-noise sample. The weights are adjusted with gradient descent to minimize the error signal, and the cycle repeats [12]. Due to hardware delay, the team decided to implement the Block-FxLMS algorithm instead, which processes and outputs data in N-sample frames.

An important consideration for the FxLMS algorithm is the secondary path, which is the physical path between the anti-noise speaker and the microphone. When anti-noise is played through the speaker, the microphone picks up a signal that is slightly quieter and out of phase with the initial signal. To obtain destructive interference at the microphone, this amplitude and phase difference must be accounted for in the anti-noise output [12].

For this project, the minimum viable product is an ANC system capable of cancelling a single known frequency. In this case, the feedforward neural network is modified slightly. For each frame of error data, a reference sine and cosine function are generated and filtered through the secondary path before updating the weights with equation (1). Then, these new weights are used to generate the next frame of anti-noise data using according to equation (2). This type of control is called sine-in-sine-out (SISO) ANC since the inputs and outputs are always pure sinusoids, as seen in Figure 3.

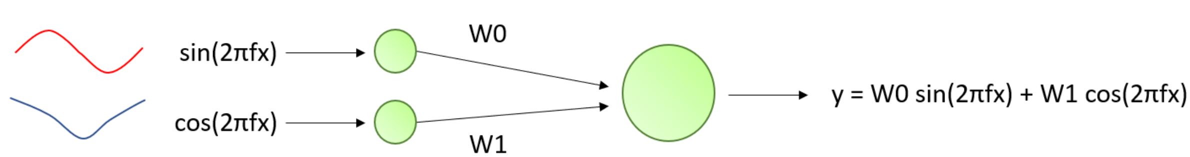

Figure 3: SISO ANC visualized as a single layer neural network.

In the following equations, W is the weight vector, e is the error data, x is the reference vector, x' is the filtered reference vector, y is the anti-noise data, and α is an adjustable learning rate. The weight vector has two components, while the reference vector and filtered reference vector have two rows which are the sine and cosine function values for the given data frame. The error and anti-noise are both one frame of signal data.

$$\vec{W}(n+N) = \vec{W}(N) - \alpha \sum_{i=1}^{N} e(i)\vec{x}'(i) \tag{1}$$

$$\vec{y}(n+N) = \vec{W}(n+N) \cdot \vec{x}(n+N) \tag{2}$$

Multiplying the error data by the filtered reference sine and cosine and summing up the resulting vectors gives a value that represents how in phase or out of phase the signals are. For example, a large positive value occurs if the error is in phase with the filtered sine function, meaning the weight coefficient of the sine function should decrease. A large negative value occurs if the error is out of phase with the filtered sine function, meaning the weight coefficient should increase. This method quickly finds the desired phase of the anti-noise signal. A diagram of the Block-FxLMS algorithm is shown below in Figure 4, with the weight update method shown in Figure 5.

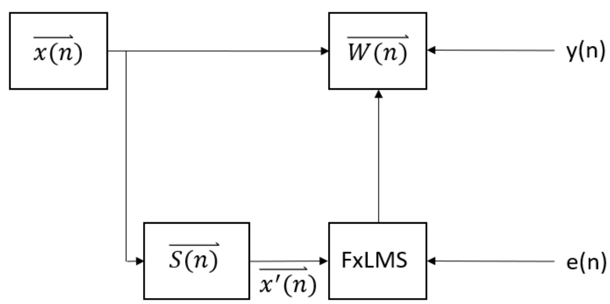

Figure 4: Diagram of the FxLMS algorithm with arrows representing how information flows throughout the system.

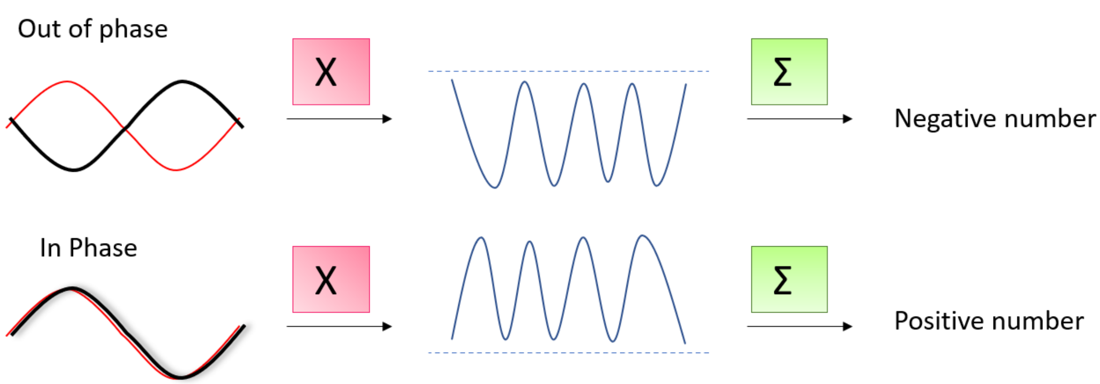

Figure 5: Diagram depicting how weights are updated during each frame. The red curve is the error signal, while the black curve is either the filtered sine or cosine reference functions.

To filter the reference sine and cosine functions, the transfer function can be expressed in real and imaginary parts and then multiplied by the reference signal as in equation (3). Multiplying by an imaginary number is equivalent to a phase shift to the left by 90 degrees, so sine becomes cosine and cosine becomes negative sine according to equation (4) [13].

$$\vec{x'}(n) = \left[Re(H) + i\big(Im(H)\big)\right]\vec{x}(n) \tag{3}$$

$$isin(x) = \cos(x)\,, icos(x) = -\sin(x) \tag{4}$$

Other ANC algorithms exist but are generally more complicated to implement, such as the linear system of phasors algorithm (LSP) [12] or the recursive least mean squares (RLMS) [14].

Methodology

A Gantt chart showing the stages of project development is shown below in Figure 6.

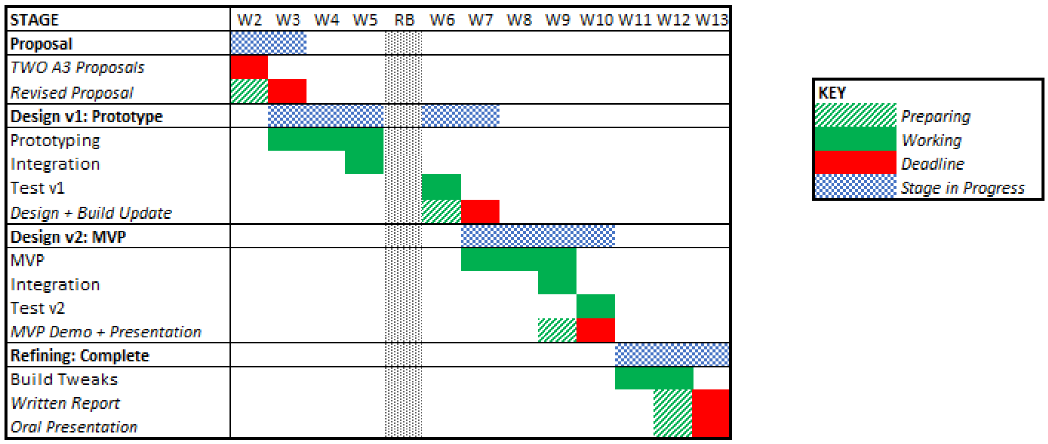

Figure 6: Gantt chart for the project showing each stage and its corresponding duration.

Physical Implementation

A flow chart of the full operational system including all hardware and software components can be seen in Figure 7. The microphone and both speakers were housed in a sound chamber with foam insulation to absorb reflections and minimize noise pollution during testing. Pictures of the chamber are shown below in Figure 8 and the inside in Figure 9.

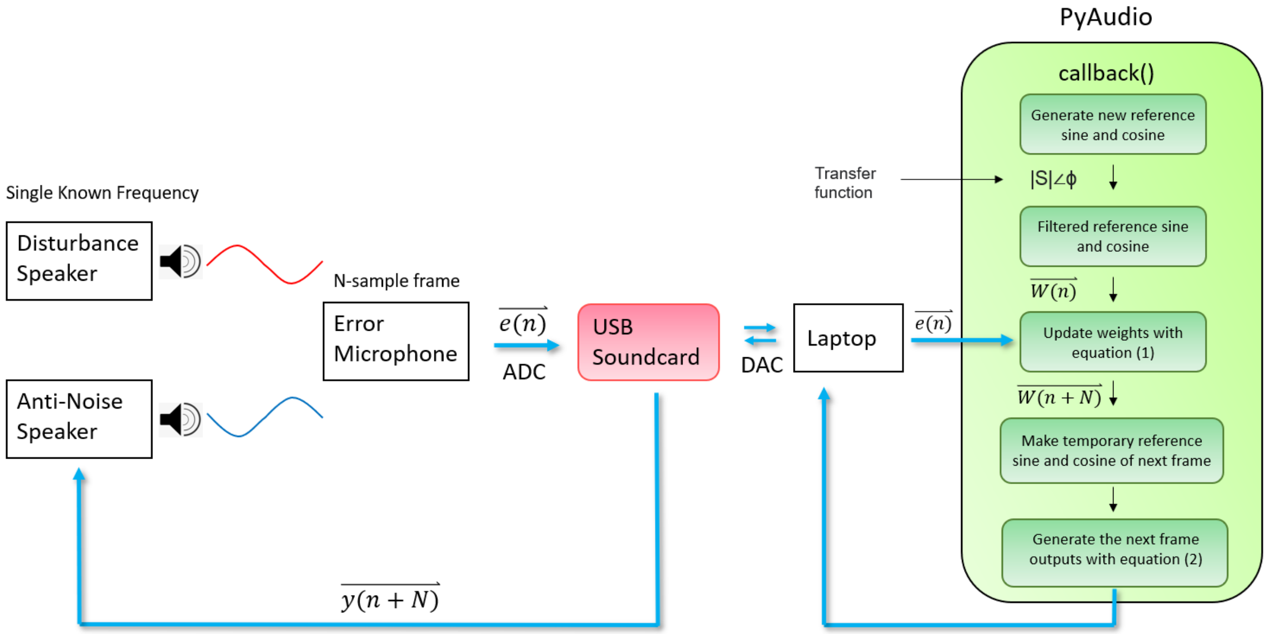

Figure 7: Diagram of the physical setup for a single tone ANC system using the Block-FxLMS algorithm. The setup uses two speakers and one microphone, with a USB soundcard acting as an ADC and DAC. The laptop runs Python code to implement the signal processing.

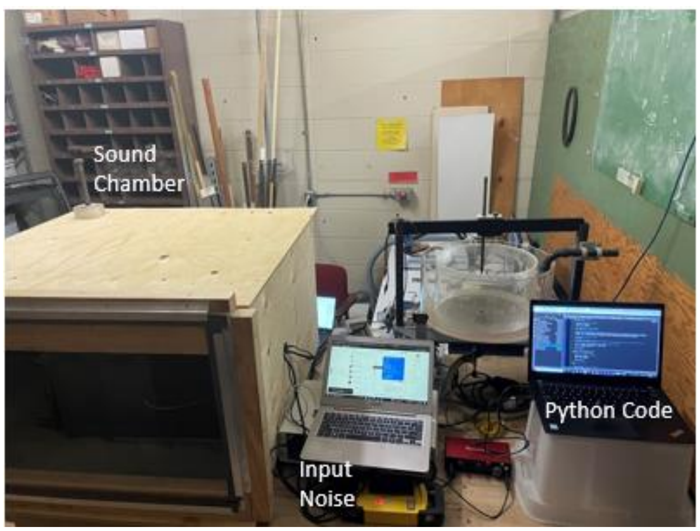

Figure 8: Physical setup of experiment. This includes the sound chamber to mitigate outside noise, an input for the disturbance speaker as well as the computer to run Block-FxLMS algorithm.

Figure 9: Inside the sound chamber and all components acting to achieve active noise cancelling.

Testing and Iteration

Speaker Troubleshooting

The experimentation of open-air noise control began with testing ANC algorithms using 2 Bose color sound link portable speakers, to no success. By performing a controlled test unrelated to the cancelling algorithm, the team was able to determine that the speakers had a non-linear frequency drift. The test setup consisted of playing a noisy single tone signal (528 Hz) through both speakers, and systematically altering the phase of one signal at even time intervals. If the signals are outputting the same frequency, the phase change (performed computationally by skipping frames) should iterate through regions of constructive and destructive interference. Initial testing with the portable speakers is seen in Figure 10.

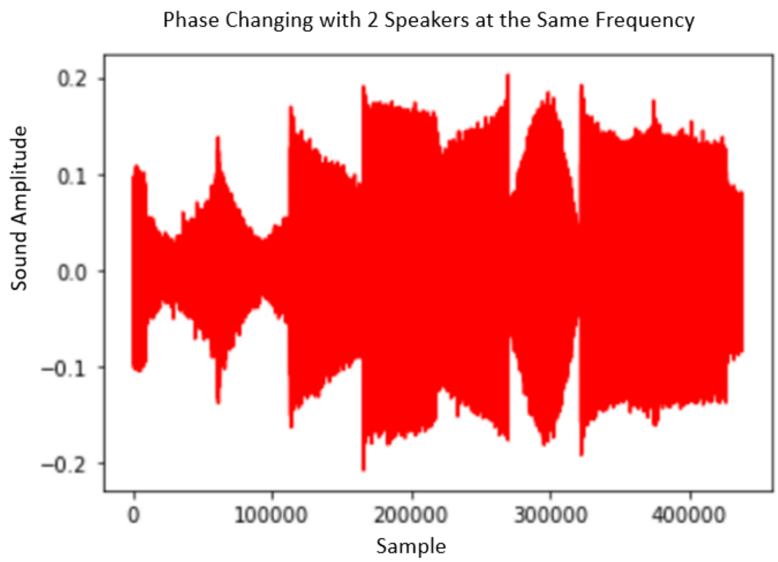

Figure 10: Error microphone signal over time plotted in python. Data captured using a standard frame rate of 44100 fps, signal phase is shifted by 1 frame every second. Phase changes are visible but difficult to discern due to frequency drift in the error microphone.

After analyzing the data in Figure 10, the team determined that the speakers must be at fault. New Bose loudspeakers were acquired, and when performing the same test again found results which matched theory well.

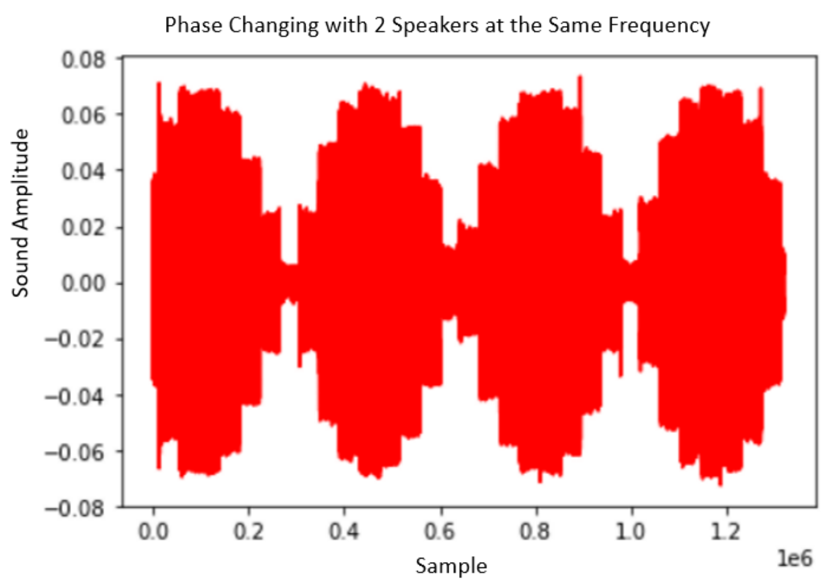

Figure 11: Error microphone signal over time measured in frames. Taken again at a standard frame rate of 44100, this time using Bose Loudspeakers. Areas of destructive interference are observed during frame counts of 300000, 650000, and 1000000.

Distinct areas of destructive and constructive interference are observed in the new data, validating the hypothesis that the ANC algorithm was initially unsuccessful due to hardware inaccuracies.

Diagnosing Secondary Path Error

Initial success came with cancelling a single tone 528 Hz signal after changing speakers, however results were inconsistent. Individual trials experienced variable success, some interfering destructively, but only to a mild extent, while others interfered constructively. This was due to transfer function variations between trials.

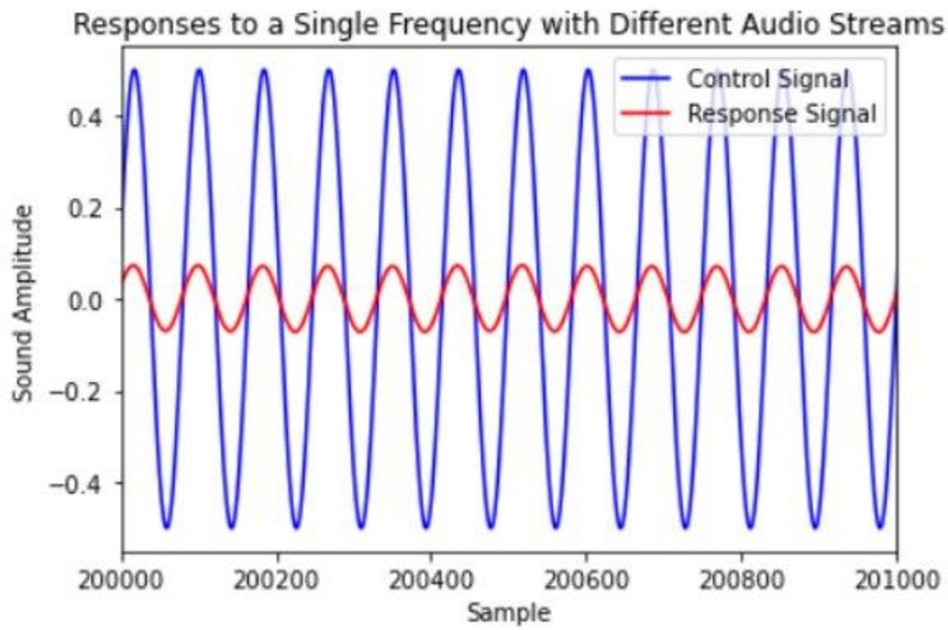

Figure 12: Calculated transfer function for the first trial run plotted in python. The blue curve is the digital signal the speaker is commanded to output, while the red curve is the signal the error microphone reads. Note, for this trial these signals are roughly in phase.

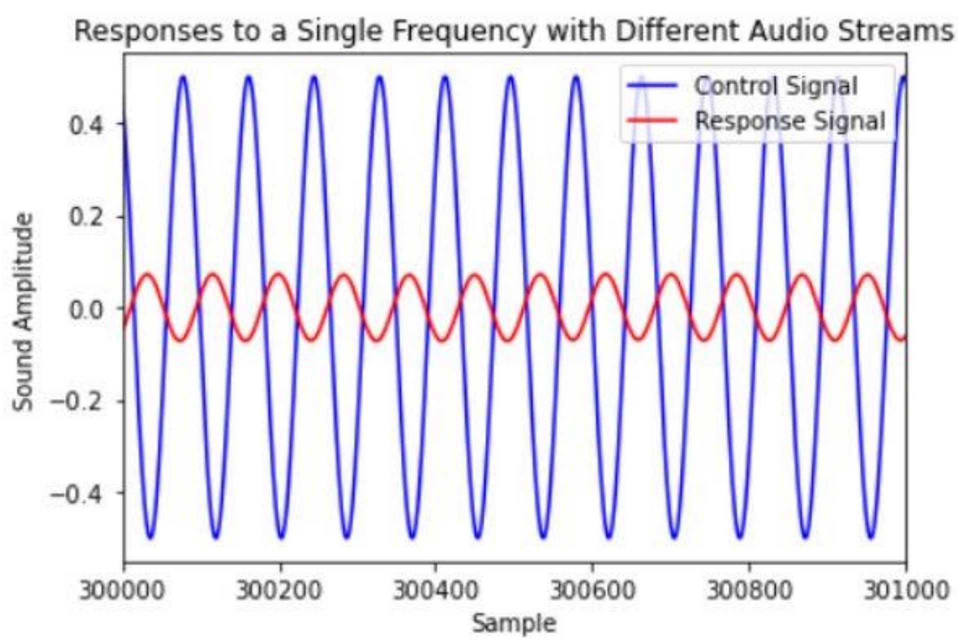

Figure 13: Calculated transfer function for the second trial run. Note, these curves are now out of phase.

Theoretically, the transfer function for a given speaker position should not change. This is because the physical distance through which sound propagates, and their nature as spherical harmonic pressure waves, is consistent. As observed in Figure 12 and Figure 13, this was not the case. It was hypothesized that when initiating an audio stream, there is some variable start-up delay between when the speaker is commanded to play and when the anti-noise begins playing. As a result, the transfer function's phase varies by as much as 360 degrees between trial runs.

20 sample transfer functions were measured, with the expectation that the phase data would fit a normal distribution around some central region. The data, however, does not support this claim as the phase change varies by as much as 360 degrees and possess 2 local maximums around 150 and 300 degrees as can be seen in Figure 14.

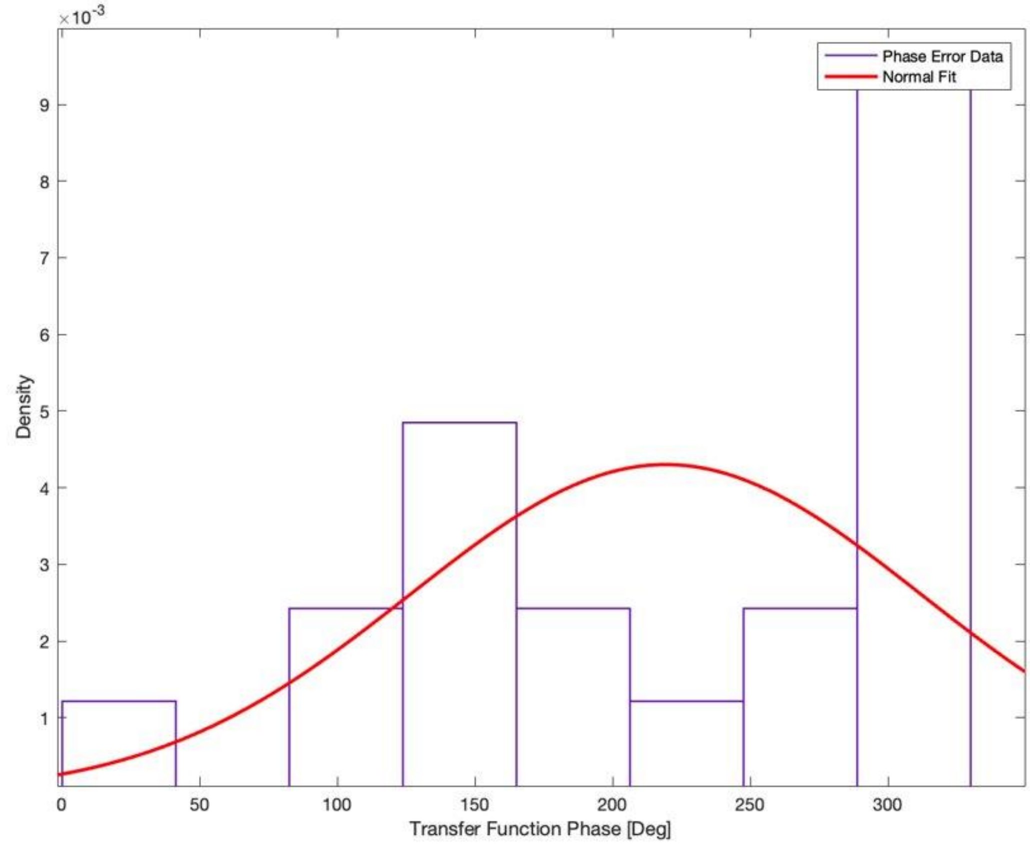

Figure 14: Normal distribution fit to the phase change data of the secondary path transfer function for 20 trials. X axis is phase change between the output sound and the error microphone signal.

Fixing this issue required amalgamating existing sets of code so the transfer function and ANC algorithm could play on the same audio stream. This proved successful and results became consistent. Figure 15 shows a successful ANC trial with noise diminishing over 15 seconds.

Final Product Results

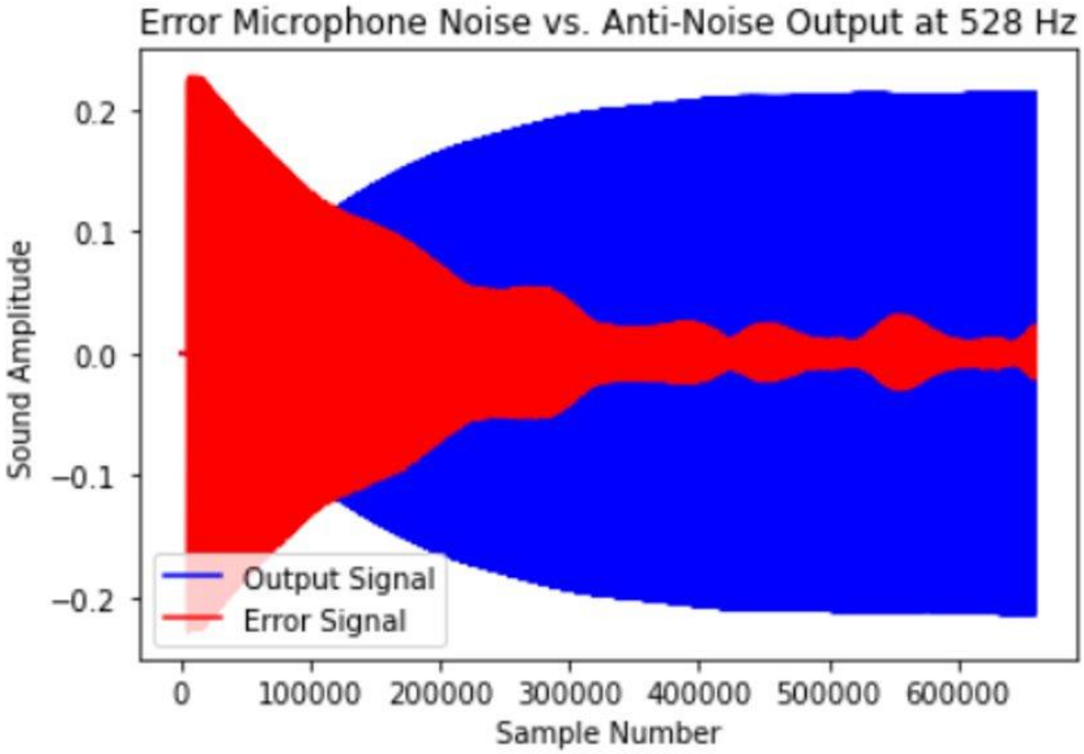

Figure 15: ANC example 1 plotted in python. Red and blue signals denote the error microphone and output sound signal respectively. Time is on the x axis in units of frames, taken at a standard frame rate of 44100.

Steady state is never completely reached due to variations in error signal amplitude. Data between the beginning and end of the error signal is analyzed in Figure 16, showing a 28.4 decibel decrease between the 2 regions of interest, on a voltage scale. This translates to a 96% reduction in noise amplitude according to equation (5). Of note is that this noise reduction is measured in the frequency domain for a distinct value, not the overall noise level which is higher due to ambient lab sound and variable frequency responses in the speaker.

$$dB_{gain} = 20\log_{10}\frac{V_{final}}{V_{initial}} \tag{5}$$

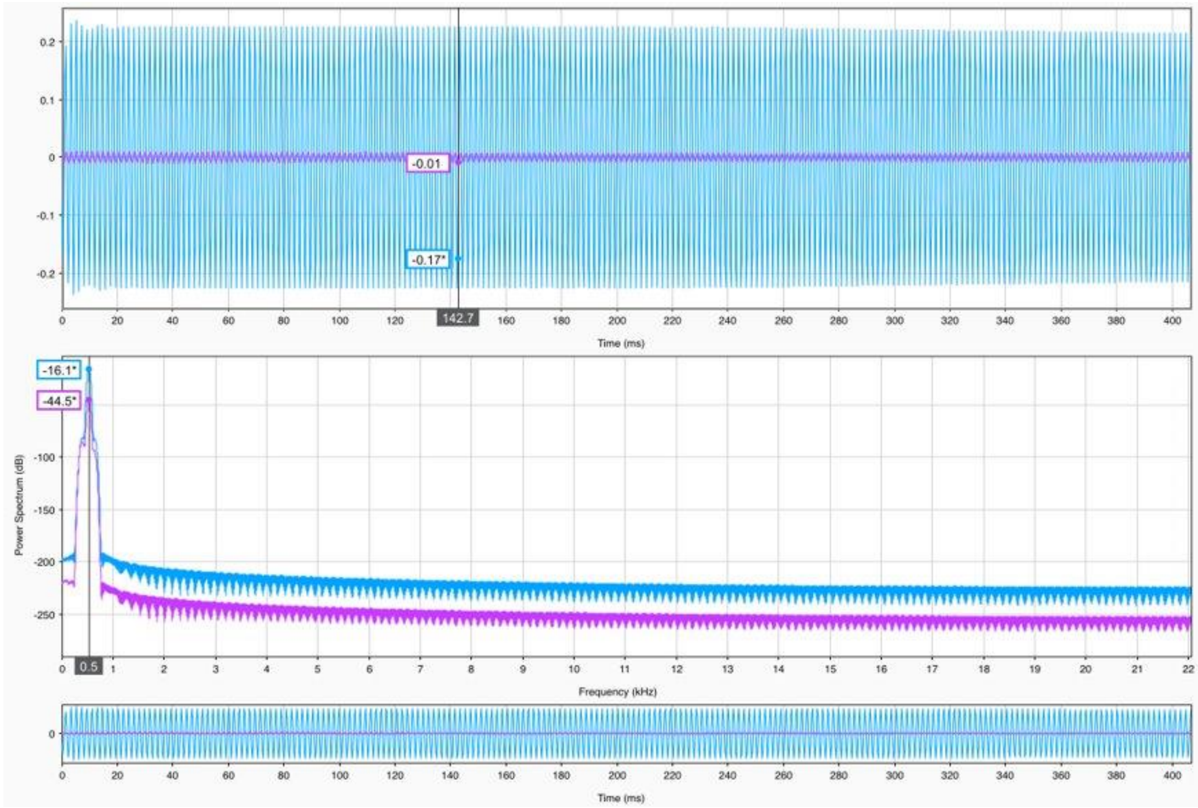

Figure 16: Initial vs Steady state amplitude comparison for ANC example 1 (top graph) visualized in MATLAB's signal analyzer. Fourier transformed signal with an applied bandpass filter between 450 and 600 Hz (bottom graph). A cursor identifies the 528 signal and the decibel drop between initial and final output.

Results for a secondary trial run with a doubled frame size (2048) are seen in Figure 17 and Figure 18. Similar results are produced with a notably smoother curve, but longer time requirements to reach a steady state. The noise reduction is measured at -25.4 dB, equivalent to a 95% amplitude reduction at steady state.

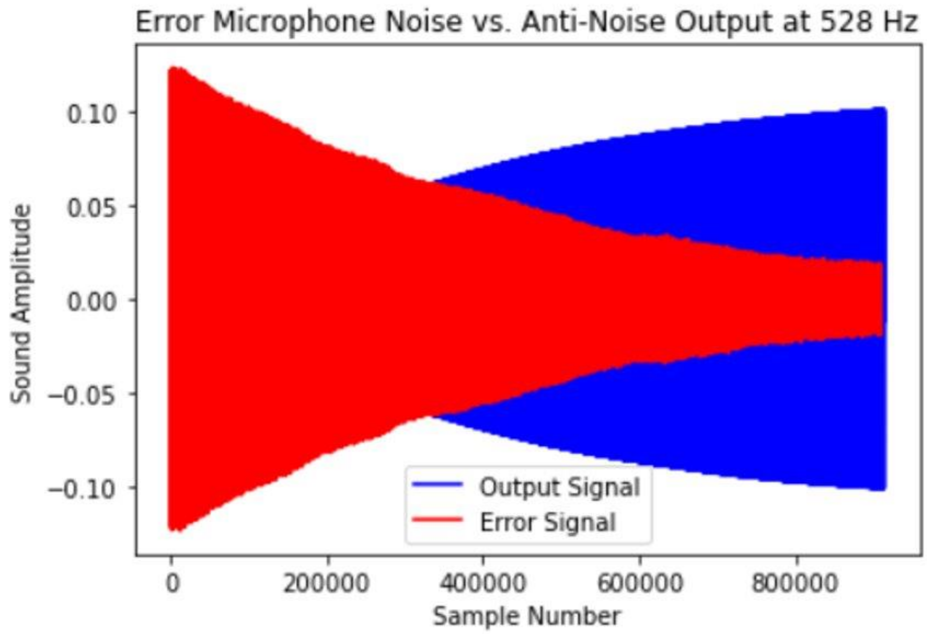

Figure 17: ANC example 2 at 528 Hz with a doubled frame size, creating smoother responses. Red and blue curves represent the error microphone and output signal respectively.

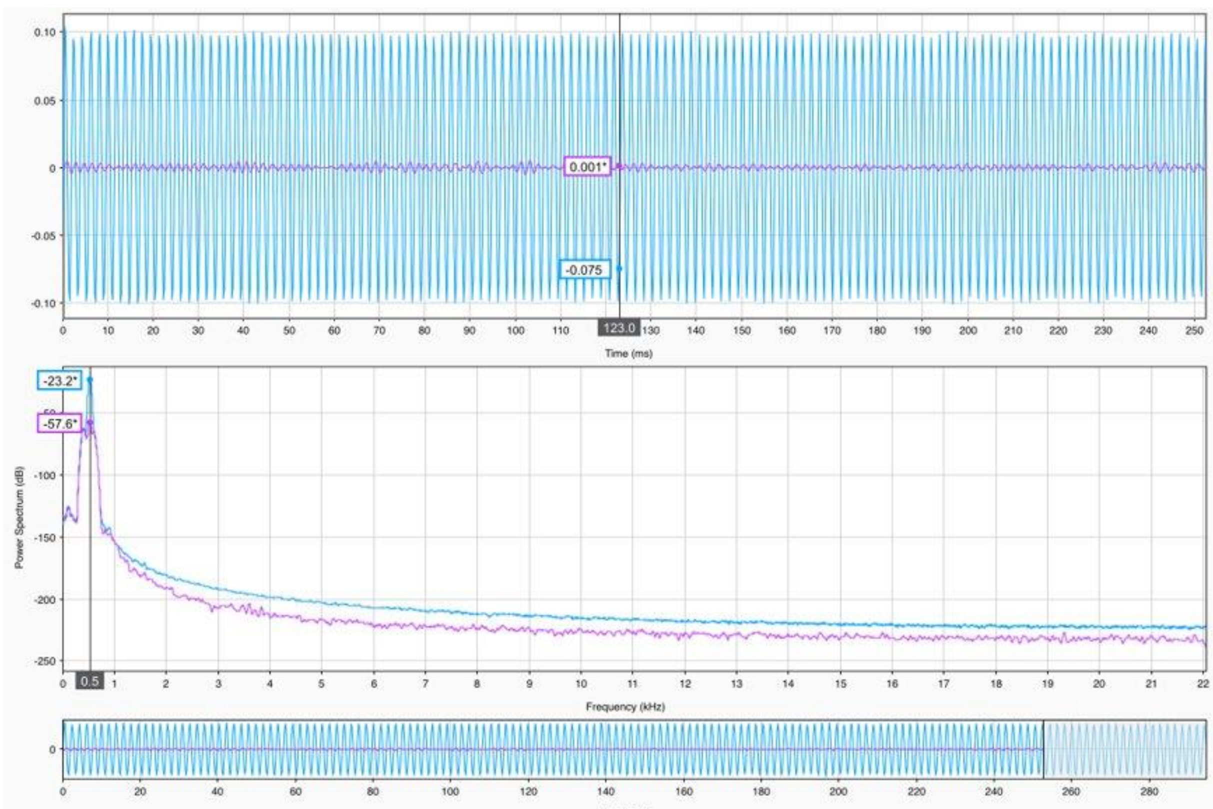

Figure 18: Initial vs Steady state amplitude comparison for ANC example 2 (top graph) visualized in MATLAB's signal analyzer. Fourier transformed signal with an applied bandpass filter between 450 and 600 Hz (bottom graph).

The final testing phase comprised adjusting the code to include multitone noise. Frequencies at 528 and 392 Hz were cancelled using the updated algorithm to a lesser extent than single tone noise. Figure 19 plots the multitone noise over time for a 10 second sample while Figure 20 shows the frequency component reduction between the beginning and end of the signal. The noise reduction is not consistent between either frequency. The 392 Hz and 528 Hz signals experienced a -10.9 dB and -1.9 dB reduction respectively.

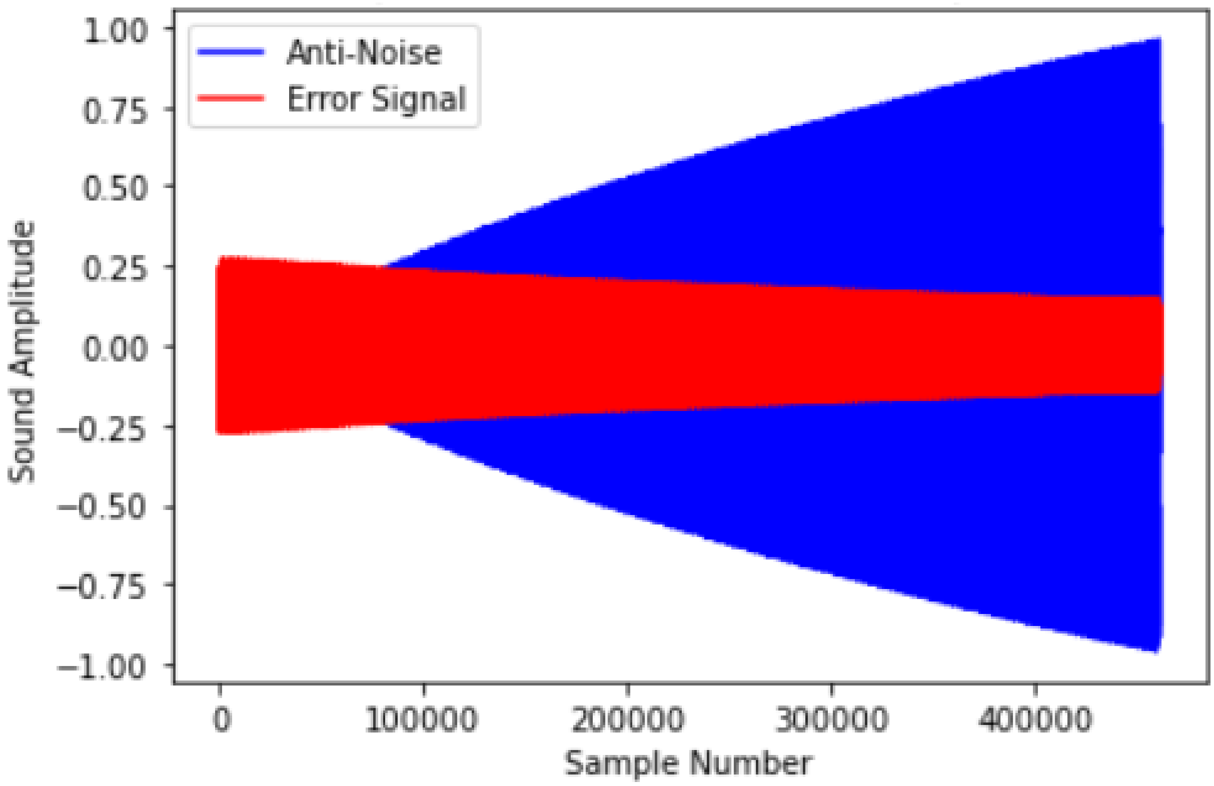

Figure 19: Multitone noise cancellation at 528 Hz and 392 Hz over a 10 second sample. The red and blue curves denote the error microphone and speaker output signal respectively.

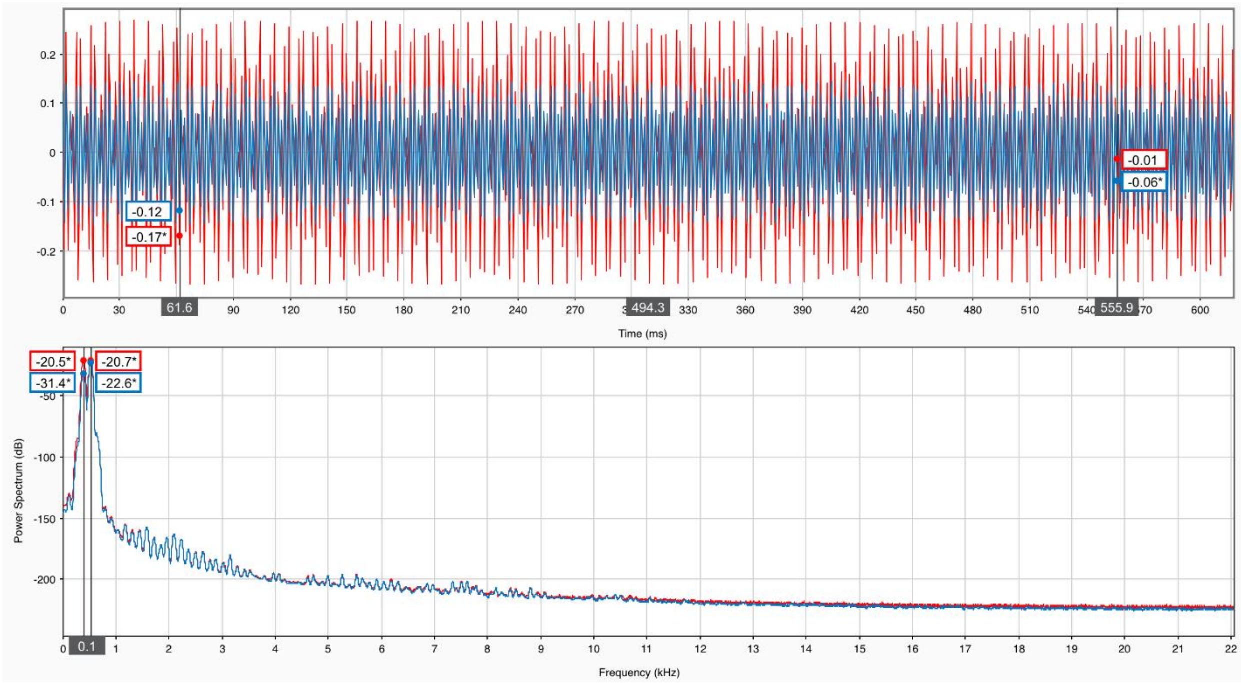

Figure 20: Initial vs Steady state amplitude comparison for multitone ANC (top graph) visualized in MATLAB's signal analyzer. Fourier transformed signal with an applied bandpass filter between 300 and 600 Hz (bottom graph). Two cursors identify the 528 and 392 Hz signals and the decibel drop between initial and final output.

The final product successfully achieved 3 out of the 6 revised project outcomes, including the cancellation of single and multitone noise by more than 1 dB and implementation of an automatic frequency finder. The current code base could be expanded to reach steady state in a lower time frame with minor adjustments. Incomplete objectives include the cancellation of white noise and a sine swept signal as these objectives would require the implementation of a feedforward microphone due to their time dependent variations.

Table 1: Table comparing initial goals to achieved goals

Initial Goals	MVP Results	Final Results
Single tone steady state cancellation of 1dB or 20% (power scale)	Cancels over 20db inconsistently	Cancels over 20db consistently
Multitone cancellation of 1dB	Untested	Cancels either tone by more than 1 dB
Automated Frequency Finder	Prototyped but not integrated	Accurate identification and transfer function extraction for N discrete signals
White noise cancellation by 1dB	Incomplete	Not implementable with current code
Reach steady state within 1 second.	Incomplete	Unfinished due to time constraints
Sinusoidal sweep noise control by 1 dB	Incomplete	Not implementable with current code

Safety, Ethical and Environmental Considerations

When performing an experiment safety is paramount, but ethical and environmental considerations are major factors that must be accounted for to ensure a holistically beneficial process.

Safety Considerations

The main safety concerns during this project were the construction and transport of the sound chamber, the use of electric devices, and prolonged exposure to loud noises. To fabricate the sound chamber power tools and woodworking machines were used. To ensure sufficient safety the team went through a machine shop training module before use and always wore the appropriate protective equipment, such as eyewear and steel toed boots. Electrical hazards are always present when handling electrical equipment. In this project the only exposure to electrical hazards were low current well insulated connections used in food and drink prohibited areas. This well controlled environment and the team's prioritization of cable management created a very low electrical risk. Finally, due to the nature of the project, experimentation consisted of extended loud noises, which can cause damage to one's hearing [15]. As a safety precaution and, the sound chamber constructed for the initial in-pane cancellation was repurposed and further insulated to contain the experimental setup, dampening the noise reaching experimenters and others nearby.

Ethical Considerations

Even with the implementation of the sound chamber to reduce noise from experiments, the noise produced by the team was unfair to others in the same environment. The steady high frequency noises could cause irritation and distraction while they worked on their own projects [16]. To ameliorate this the team relocated the experimental setup to a more private location where any noise pollution would be far removed from other students.

Environmental Considerations

The largest environmental cost of this project was undoubtedly the manufacture and transportation of components ordered online. The team attempted to minimize this, but components for the initial in-pane noise cancellation method were ordered before the change in scope to open air cancellation. These piezoelectric parts became useless for the project, creating unnecessary environmental strain which is worsened by the fact that they may not be used much by other students in the future.

Discussion

The system was able to autonomously identify and reduce steady single discrete frequency noise by 28dB in roughly 10 seconds, overshooting initial expectations for noise reduction. Proof of concept was achieved for multi tone noise cancellation, but performance was inconsistent between the cancelled frequencies suggesting further iteration is necessary. This level of functionality achieved three of the six goals set out at the start of the design process. The goals that were not met were white noise cancellation, 1 second cancellation time, and the cancellation of a frequency sine sweep. White noise was not cancelled because the code can only handle discrete frequencies, not a continuous spectrum of frequencies as is found in white noise; the cancellation time was not reduced below 1 second due to poor saturation control leading to errors in the anti-noise signal if the learning rate got too high; and the sine sweep was not cancelled because the change in scope to open air noise control led to implementing a soundboard that did not have the number of ports necessary for hybrid feedback/forward control. All these goals are implementable with the current system but could not be completed withing the time constraints of the semester.

Additionally, there were several unforeseen limitations to the system, such as high sensitivity to noise external to the source signal and an unstable transfer function. The external noise sensitivity was problematic because conversation level noise near the sound chamber could lead to reduced cancellation efficacy or even amplification. To address this noise was carefully controlled during experimentation, but this is unrealistic in a real-world application. Furthermore, the variable error in the transfer function mandated a short period of source silence to calibrate the secondary path before every cancellation cycle, a process which is impractical for a final product as there will be no control over the source noise outside of the testing environment. A currently undefined contributor to system error is the frequency finding function used to identify noise for cancellation. Without high precision equipment the exact difference between the calculated and real frequencies of the source noise was not quantified, which could have been a significant contributor to error especially later in cancellation cycles when dephasing would be greatest.

Economic Analysis

This project was given a budget of one thousand dollars ($1000) to be completed and was completed under budget at a total cost of $423.91 for all actual expenditures. Of the $424 spent, actual project expenditures were equipment and experimental testing based, with 13% being spent on prototyping parts, 21% on the experimental setup, and 66% on the final design. Labour cost has not been accounted for in this number, though an approximate 624 hours (8 hours a week for 13 weeks, for 6 individuals) was spent on the project. A high-level budget of project expenditure can be found in Table 2 below. A pie chart of project expenditure of total budget given can be found in Figure 21 below.

Table 2: Table outlining a breakdown of costs during the project and total expenditure

#	Item Name	Price
1	Sound Proof Foam x3	$ 89.97
2	Focusrite Scarlett 2i2	$ 281.37
3	Piezoelectrics Sensors x5	$ 13.79
4	ADC	$ 27.74
5	DAC	$ 11.04
TOTAL SPENT DURING PROJECT (incl. HST)		**$ 423.91**

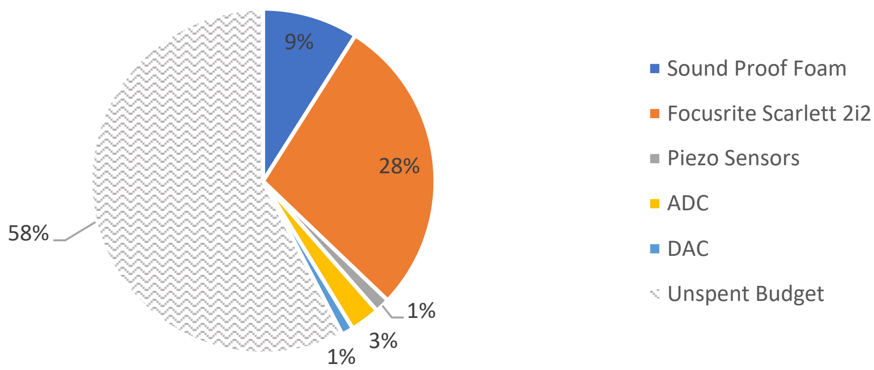

Figure 21: Pie chart of product expenditure as related to total provided budget of $1000

13% of spent budget went towards purchasing DAC and ADCs necessary for investigating the use of Piezoelectric sensors as an interface. As discussed in Design Decisions, the purchased equipment was found to be too ineffective for the desired purpose. Upon investigation, acquiring suitable equipment would have been out of the projects allocated budget. These purchases can still be considered essential and worthwhile as they allowed for quantifiable demonstration that the initial solution was ineffective, which would not have been possible without the equipment.

For the product itself, equipment such as microphones and speakers were sourced at no cost to the team. To recreate a product identical to the team's would cost an estimated $940, not including a computer to run the code. However, the product could realistically be designed and produced more cheaply considering the product functionally only requires: one microphone, one speaker, and hardware to interface and run python code. A high-level breakdown of equipment and associated estimated cost can be found in Table 3 below.

Table 3: Table outlining a breakdown of product cost

#	Item Name	Price
1	Focusrite Scarlett 2i2	$ 281.37
2	Rode Microphones	$ 259.00
3	Bose Speakers	$ 400.00
TOTAL COST OF PRODUCT		**$ 940.37**

The product is currently not market viable, considering the narrow use case, unreliability of product, and expensive product parts. To bring this product to market would require further development time and expenditure to become competitive with existing products such as ANC headphones and equipment testing to procure suitable and inexpensive microphone and speakers. To list two potential opportunities, this product could be realised as a software solution for customers and businesses to run on already owned microphones and speakers, or alternatively, sold as a pre-integrated product requiring minimal user setup, including selected equipment. To date, there are no identical products on the market for open-air noise cancellation, however, noise cancelling devices such as ANC headphones on market cost in the range of $100 - $400 and have become widely accepted by consumers worldwide.

Conclusion

The team began the project by aiming to create a vibroacoustic noise cancelling device to reduce the transmission of sound through a windowpane as a partial solution to the adverse effects of noise pollution on civilian life. As this initial aim was found to be unrealistic given the constraints on the project, the team pivoted to open air noise control. The final design was able to achieve the minimum viable goal of the project by identifying and reducing steady, discrete, and multi-tone noise by more than 1dB.

Despite this success, the system requires substantial improvement before it satisfies all initial goals. In a real-world application, the noise targeted by the system will be time-variant over a continuous frequency spectrum and signal processing will need to occur on the timescale of 10s of milliseconds.

Next steps to improve on the systems design include:

1. Upgrading the system to time-variant control by replacing the soundcard with an ADC/DAC unit with three ports instead of two. This is to accommodate the additional microphone necessary for feedforward control. This reference microphone will extract incoming noise to be processed and be outputted as anti-noise before passing the anti-noise to the speaker, enabling dynamic control.
2. Reducing the processing time of the system, likely by implementing the algorithm on a custom PCB (which could include the ADC/DAC ports mentioned previously).
3. Eliminating the variable phase delay observed in the experimentally recorded transfer functions of the system. Likely to be achieved with custom hardware, allowing for high precision calibration of the transfer function during setup to be used for every subsequent operation, instead of the current recalibration necessary before each cancellation process.
4. Improving the code for continuous frequency spectrum cancellation. Implementing a frequency finding function is compatible with the currently employed FxLMS algorithm, though will require alterations to the information encoding system currently in place.

These improvements address the start of the necessary changes to achieve commercially viable functionality. The remaining challenges would also include packaging, fabricating, and marketing the system into a cohesive easily installable product.

References

[1] "Applied Ecology and Sustainable Environment - Shri V.K Jain - Google Books." https://books.google.ca/books?id=qRYsEAAAQBAJ&pg=PA283&lpg=PA283&dq=%E2%80%9CWorldwide,+noise-induced+hearing+impairment+is+the+most+prevalent+irreversible+occupational+hazard,+and+it+is+estimated+that+120+million+people+worldwide+have+disabling+hearing+difficulties.%E2%80%9D&source=bl&ots=T1SpWM_5qQ&sig=ACfU3U0tX294f0Xdd5QJVVgxHwqjqx83xg&hl=en&sa=X&ved=2ahUKEwjjwfeMy_T7AhU6kIkEHTfNDNUQ6AF6BAgIEAM#v=onepage&q&f=false (accessed Dec. 12, 2022).

[2] "Noise pollution is a major problem, both for human health and the environment — European Environment Agency." https://www.eea.europa.eu/articles/noise-pollution-is-a-major (accessed Dec. 12, 2022).

[3] "(PDF) The study of the effectiveness of sound insulation of gypsum wall boards with 'Shimast' coating." https://www.researchgate.net/publication/345401068_The_study_of_the_effectiveness_of_sound_insulation_of_gypsum_wall_boards_with_Shimast_coating (accessed Dec. 12, 2022).

[4] J. D. Quirt, "Sound transmission through windows," *Natl. Res. Counc. Can. Inst. Res. Constr.*, vol. 240, Feb. 1988, doi: https://doi.org/10.4224/20330959.

[5] "A review of the application of active noise control technologies on windows: Challenges and limitations - ScienceDirect." https://www.sciencedirect.com/science/article/abs/pii/S0003682X20308574?casa_token=fIjWvgkh-8UAAAAA:4-avPOA9vf5D_m9zUTu2wCxFzyvmt3dynYAB7xhfRYVgvw8taRGIETfuqFn-dNqJyEZDJB0h (accessed Dec. 12, 2022).

[6] P. Lueg, "Process of Silencing Sound Oscillations." https://patentimages.storage.googleapis.com/97/93/fc/19408ed05d22d6/US2043416.pdf (accessed Dec. 12, 2022).

[7] "Bose Global Press Room - Bose QuietComfort 25 Acoustic Noise Cancelling Headphones." https://www.bose.ca/en_ca/pressroom/archive/2015/bose-quietcomfort-25-acoustic-noise-cancelling-headphones.html (accessed Dec. 12, 2022).

[8] V. Varadan, "United States Patent." https://patentimages.storage.googleapis.com/3e/2d/1c/da5999f9352cc7/US5668744.pdf (accessed Dec. 12, 2022).

[9] A. Tadeu and D. Mateus, "Sound transmission through single, double and triple glazing. Experimental evaluation," *Elsevier Appl. Acoust.*, vol. 62, no. 3, pp. 307–325, Mar. 2001, doi: 10.1016/S0003-682X(00)00032-3.

[10] "Look to windows, walls and ceilings to shut out noise | The Seattle Times." https://www.seattletimes.com/life/lifestyle/look-to-windows-walls-and-ceilings-to-shut-out-noise/ (accessed Dec. 12, 2022).

[11] I. T. Ardekani and W. H. Abdulla, "FxLMS-based Active Noise Control: A Quick Review," p. 10.

[12] J. Cline, J. Hamann, M. Lui, and H. Pham, “Active Noise Control: System Design, Implementation, and Evaluation Report.” https://workstory.s3.amazonaws.com/assets/670544/ANC_Final_Report_original.pdf (accessed Dec. 11, 2022).
[13] C. House and J. Cheer, “Efficient Raspberry PI Implementation of a SISO Active Noise Control System Using Python.” https://eprints.soton.ac.uk/428207/1/Efficient_Raspberry_Pi_Implementation_of_a_SISO_Active_Control_System_Using_Python.pdf (accessed Dec. 11, 2022).
[14] M. Moazzam and M. Shoaib Rabbani, “Performance Evaluation of Different Active Noise Control (ANC) Algorithms for Attenuating Noise in a Duct.” http://bth.diva-portal.org/smash/get/diva2:830747/FULLTEXT01.pdf (accessed Dec. 11, 2022).
[15] “High Frequency Hearing Loss: Know the Symptoms and Treatment.” https://www.hoffmannaudiology.com/blog/high-frequency-hearing-loss-symptoms-and-treatment (accessed Dec. 12, 2022).
[16] “Noise Pollution | National Geographic Society.” https://education.nationalgeographic.org/resource/noise-pollution (accessed Dec. 12, 2022).

Appendix

```
[ ]: #Intialising Libraries
     import pyaudio
     import wave
     import time
     import numpy as np
     from matplotlib import pyplot as plt
     import scipy.signal as sig
     import soundcard as sc
```

```
[ ]: # get a list of all speakers:
     speakers = sc.all_speakers()
     print(speakers)
     # get the current default speaker on your system:
     default_speaker = sc.default_speaker()
     print(default_speaker)
     # get a list of all microphones:
     mics = sc.all_microphones()
     print(mics)
     # get the current default microphone on your system:
     default_mic = sc.default_microphone()
     print(default_mic)

     # search for a sound card by substring:
     sc.get_speaker('Scarlett')
     one_mic = sc.get_microphone('Scarlett')


     p = pyaudio.PyAudio()
     info = p.get_host_api_info_by_index(0)
     numdevices = info.get('deviceCount')
     print(numdevices)

     for i in range(0, numdevices):
         print("Device id ", i, " - ", p.get_device_info_by_host_api_device_index(0, i).get('name'))
```

Figure 22: Setup code to import all libraries and check which input and output indices to use when plugged into the soundcard.

```
[ ]: #ANC parameters
CHUNK = 2048
FORMAT = pyaudio.paFloat32
CHANNELS = 1
RATE = 44100

#now that we have the secondary path implement the ANC
p = pyaudio.PyAudio()

NumFreqs = 2 #Number of frequencies to analyse
x=np.arange(0,CHUNK,1) #for calculating the reference each iteration

#secondary path arrays
response_frames = np.array([])
control_frames = np.array([])
control_frames = np.append(control_frames, np.zeros(CHUNK))
count = 0

#ANC parameters
frames = np.array([])
output = np.zeros(CHUNK)
outputs = np.array([])
response = np.zeros(CHUNK)
responses = np.array([])

#Learning rate of ANN
LR = 0.00002

#high pass filter
normal_cutoff = 50/(0.5 * RATE)
b, a = sig.butter(6, normal_cutoff, btype='high', analog=False)

#create weights array <<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<CURRENTLY SET TO TWO FREQ
W = np.zeros((4,1))
```

Figure 23: Parameters for the ANC code including secondary path measurement and noise cancelling. Note that this is for two tone cancellation.

```
def main():

    #open the audio stream
    stream = p.open(format=FORMAT,
                    channels=CHANNELS,
                    rate=RATE,
                    output=True,
                    input=True,
                    input_device_index = 1,
                    output_device_index = 4,
                    frames_per_buffer=CHUNK,
                    stream_callback=callback)

    #callback function is called while this loop waits
    stream.start_stream()
    while stream.is_active():
        time.sleep(20)
        stream.stop_stream()
    stream.close()

    #plot the error over time
    numpydata = np.hstack(frames)

    #plot the outputs over time
    numpydata1 = np.hstack(outputs)
    plt.plot(numpydata1, 'b-', label = 'Anti-Noise')
    plt.plot(numpydata, 'r-', label = 'Error Signal')
    plt.title('Error Microphone Noise vs. Anti-Noise Output at 528 Hz')
    plt.xlabel('Sample Number')
    plt.ylabel('Sound Amplitude')
    plt.legend()
    plt.show()

    p.terminate()
```

Figure 24: Main method that opens the audio stream and sets up a delay of 20 seconds to perform the secondary path measurement and noise cancelling. After the delay, the error signal and output signal are plotted.

```python
#called everytime there is new audio to record or play
def callback(in_data, frame_count, time_info, flag):
    global response_frames, control_frames, output_sin, frames, outputs, responses, x, count, S, Freqs

    input_audio = in_data

    #measure the transfer function for the first 200 frames
    if (count < 100):
        #record the audio through the microphone
        response_frames = np.append(response_frames, np.frombuffer(input_audio, dtype=np.float32))

        #play combination of 392Hz and 528Hz
        output_sin = 0.25*(np.sin(2*np.pi*392*(1/RATE)*x)+np.sin(2*np.pi*528*(1/RATE)*x))
        x = x+CHUNK

        output_audio = output_sin
        control_frames = np.append(control_frames, output_audio)

        count = count + 1
        y_out = output_audio
```

Figure 25: For the first 100 frames, the transfer function is measured.

```python
    #calculate the transfer function in this frame
    elif (count == 100):

        #don't care about inputs and outputs
        input_audio = in_data
        y_out = np.zeros(CHUNK)

        #fourier transforms of input and output
        nfft = 16384 #frequency range for fft

        #cross spectral density between the control output and microphone input is the Fourier Transform of their convolution
        [f,sXY] = sig.csd(control_frames, response_frames, fs=RATE, window='hanning', nperseg=nfft, noverlap=nfft/2, nfft=nfft)
        [f,sXX] = sig.welch(control_frames, fs=RATE, window='hanning', nperseg=nfft, noverlap=nfft/2, nfft=nfft)

        #the transfer function is H = Sxy/Sxx
        H = sXY/sXX

        #INDEX AS ARRAY <<<<<<<<<<
        Freqs = FreqFinder(response_frames[-5*CHUNK:-1], NumFreqs = NumFreqs)
        Freqs = np.sort(Freqs)
        S = np.zeros((len(Freqs), 2))
        for i in range(len(Freqs)):
            index = (np.abs(f-Freqs[i])).argmin() #select the indices of the transfer function closest to the frequency of the noise
            S[i][0] = np.real(H[index])
            S[i][1] = np.imag(H[index])

        np.savetxt('Filter.txt', S)
        count = count + 1
```

Figure 26: After measuring, take one frame to calculate the transfer function and save it to a file.

```python
    #delay to start the disturbance signal
    elif (count > 100 and count < 200):

        #don't care about inputs and outputs
        y_out = np.zeros(CHUNK)
        count = count + 1

    #do the noise cancelling FREQ 1 and FREQ 2
    else:

        #record the error through the microphone
        error = input_audio
        frames = np.append(frames, np.frombuffer(error, np.float32))
        errorData = sig.lfilter(b, a, np.frombuffer(error, dtype=np.float32))

        #new reference signals
        x_reference_sin0 = np.sin(2*np.pi*392*(1/RATE)*x)
        x_reference_cos0 = np.cos(2*np.pi*392*(1/RATE)*x)

        x_reference_sin1 = np.sin(2*np.pi*528*(1/RATE)*x)
        x_reference_cos1 = np.cos(2*np.pi*528*(1/RATE)*x)
        x = x+CHUNK

        #frequency response - recall that multiplying by i is the same as a 90 degree phase shift to the right (isin -> cos, icos -> -sin)
        filtered_sin0 = S[0][0]*x_reference_sin0 + S[0][1]*x_reference_cos0  # y = (H_R + iH_I)(sin(wt)) = H_r sin(wt) + iH_I sin(wt)
        filtered_cos0 = S[0][0]*x_reference_cos0 - S[0][1]*x_reference_sin0  # y = (H_R + iH_I)(cos(wt)) = H_r cos(wt) + iH_I cos(wt)
        filtered_sin1 = S[1][0]*x_reference_sin1 + S[1][1]*x_reference_cos1  # y = (H_R + iH_I)(sin(wt)) = H_r sin(wt) + iH_I sin(wt)
        filtered_cos1 = S[1][0]*x_reference_cos1 - S[1][1]*x_reference_sin1  # y = (H_R + iH_I)(cos(wt)) = H_r cos(wt) + iH_I cos(wt)

        #update the weights
        W[0] = W[0] - (LR)*np.sum(errorData*filtered_sin0)
        W[1] = W[1] - (LR)*np.sum(errorData*filtered_cos0)
        W[2] = W[2] - (LR)*np.sum(errorData*filtered_sin1)
        W[3] = W[3] - (LR)*np.sum(errorData*filtered_cos1)

        #references of next frame
        x_sin_next0 = np.sin(2*np.pi*392*(1/RATE)*x)
        x_cos_next0 = np.cos(2*np.pi*392*(1/RATE)*x)

        x_sin_next1 = np.sin(2*np.pi*528*(1/RATE)*x)
        x_cos_next1 = np.cos(2*np.pi*528*(1/RATE)*x)

        #output calculated with the weights and the next reference frame
        output = W[0]*x_sin_next0 + W[1]*x_cos_next0 + W[2]*x_sin_next1 + W[3]*x_cos_next1

        outputs = np.append(outputs, output.astype(np.float32))
        y_out = output

    return (y_out.astype(np.float32).tobytes(), pyaudio.paContinue)
```

Figure 27: 100 frames of delay to start the disturbance signal, then the remaining frames are used to cancel the two-tone signal. The main method is called at the end of the code.

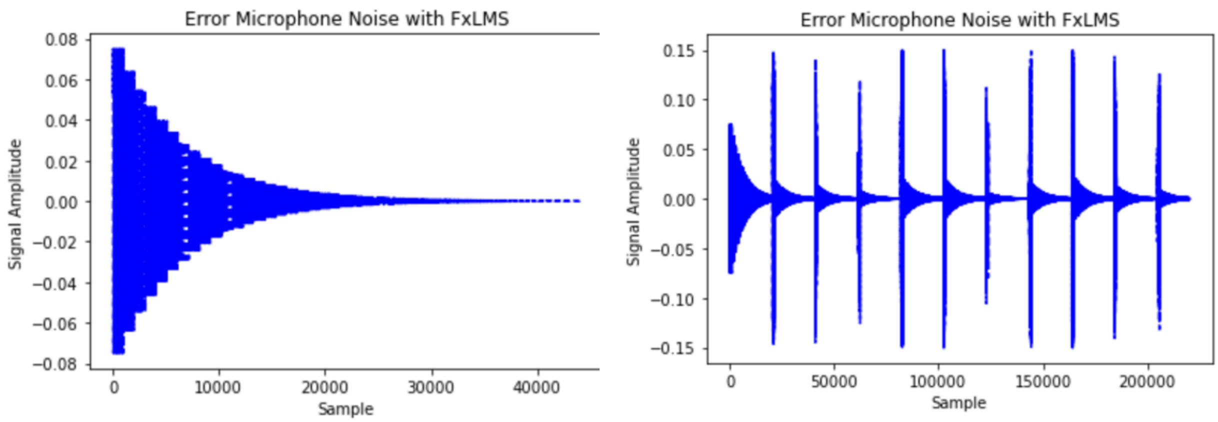

Figure 28: FxLMS simulations for a constant frequency and a periodically increasing frequency.

All code can be downloaded from the following GitHub repository:
https://github.com/MatthewBoxer/SISO-Active-Noise-Controller