EECE 5698 Fall 2025
Lab 2: Acoustic Injections Against Accelerometers Exploiting Signal Aliasing

Instructor: Prof. Kevin Fu
TAs: Hui Zhuang, Nuntipat Narkthong
Last updated: 09/22/2025, 09 AM

Submission Deadline: Oct 6, 2025 by 11:45 AM.
Submit your report as a group on Canvas.

Equipment

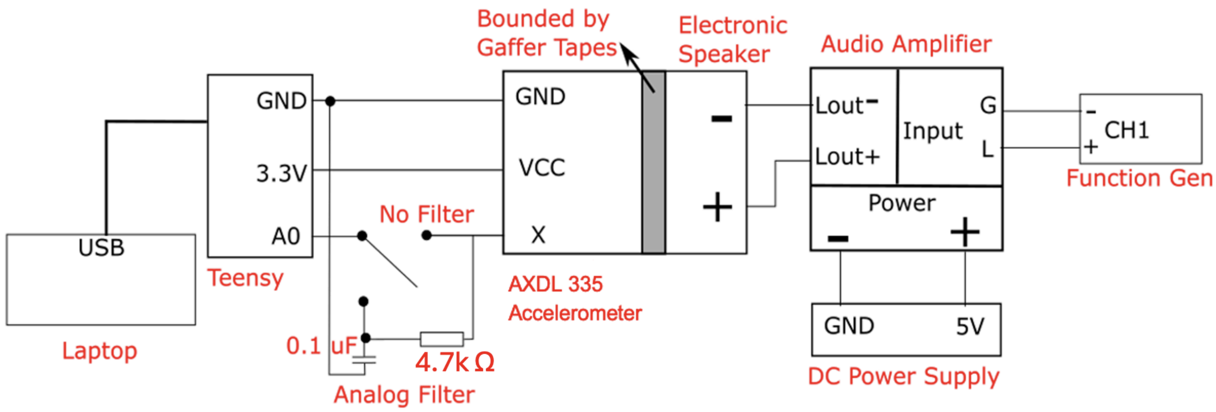

Fig. 0. Connection Diagram

- An electronic speaker tightly coupled with an AXDL335 MEMS analog output accelerometer (#1 in Fig. 1; you need to make it by yourself; Fix them on the breadboard.)
- An audio amplifier (#2 in Fig. 1; 5V power supply; use left channels only; use maximum output level)
- Teensy 4.0 board for reading accelerometer output voltage @ 50 Hz sample rate (#3 in Fig. 1, specs and pinouts can be found here)
- A 100 nF capacitor for low pass filtering (#4 in Fig. 1)
- A 4.7K Ohm resistor for low pass filtering (#5 in Fig. 1)
- Function Generator Agilent 33220A (Use Sine 1Vpp output)
- Your laptop, DC power supply,
- Noise-reduction earplugs

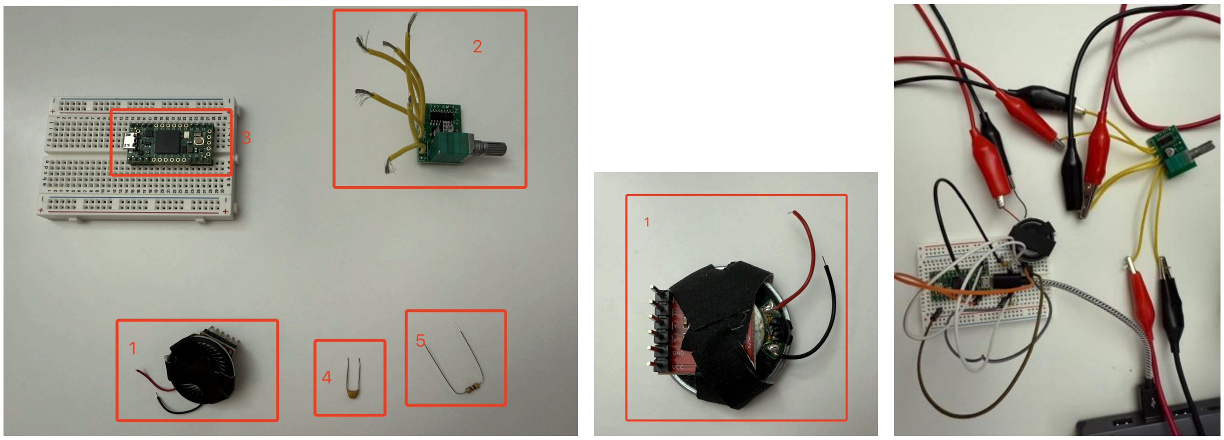

Fig. 1. Main Components

Notes

- In this lab, we define resonance points as any visually salient local maximum in the sensor readings deviations under injection attack.
- Make sure you wear noise reduction ear plugs to protect your hearing from the loud noise.
- Data logged in this lab has a sample rate of 50 Hz, which is the configured sample rate of the Teensy board.
- The Teensy analog input pin A0 is not pin 0!
- **Only use 1 Vpp output for the function generator throughout the lab. If you go higher, you can fry your amplifier board!**

Part 1: Acoustic Injection & Analog Filtering

In this lab, you will understand the relationship between sound and mechanical vibrations by using electronic speakers to change the measurements of accelerometers. You will also realize the importance of analog filtering in removing unwanted sensor signals.

Problem 1. Resonance Characterization

Step 1: Bond the speaker with the accelerometer

To demonstrate the impact of sound on accelerometer sensors with portable electronics and a reasonable level of noise in the lab, you need to first tightly bond the electronic speaker and the accelerometer. To this end, use gaffer tapes to tape them together, as shown in Fig. 1. Attach the metal side of the speaker with the side of the AXDL335 dev board that does not have the sensor module on it.

Remember that you will need them taped as tight and rigid as possible. The tighter they are, the more energy generated by the speaker can propagate to the accelerometer and change the sensor readings.

At the end of this lab session, **you will also need to remove the tapes so that students from the next session can repeat the process themselves.**

Step 2: Setup up the remaining things

Now you need to finish the remaining connections as shown in Fig. 0. Basically, you output sine wave signals to your audio amplifier, which drives the speaker and thus change the accelerometer readings acquired by the Teensy 4.0 board. This lab only cares about data of the accelerometer's X axis.

In problem 1, you need to connect the X axis output of the accelerometer directly to the A0 port of Teensy to inspect the frequency response and resonance characteristics of the accelerometer without any filtering. The impact of RC low pass filtering will be investigated later.

You also need to connect the Teensy board with your laptop through USB to display and/or save the sensor data. The Arduino script can be downloaded from here.

There are two ways we handle the sensor data transmitted to your laptop from Teensy through USB serial in this lab. (1) When you need to save the sensor data, run this Python program as introduced in the prelab. The Python program also displays the temporal and spectral sensor readings, but the display is slow. (2) When you need to get a sensor reading graph that updates rapidly (e.g., for tuning your injection frequency in problem 3), you can use Arduino IDE's built-in serial plotter.

When it's all set up, use the function generator to do a frequency sweep from 50 Hz to 10k Hz in 30s (linear frequency increase). Log and the sensor readings.The sensor reading collected will be similar to the one shown in Fig. 2. You need to plot the logged sensor readings in the saved csv file using the script you wrote in the prelab. **Note that you can take a quick look at the data you get by plotting a rough graph and leave the other time-consuming plotting adjustments required by submission after the lab. This way, you can save time for in-lab experiments.**

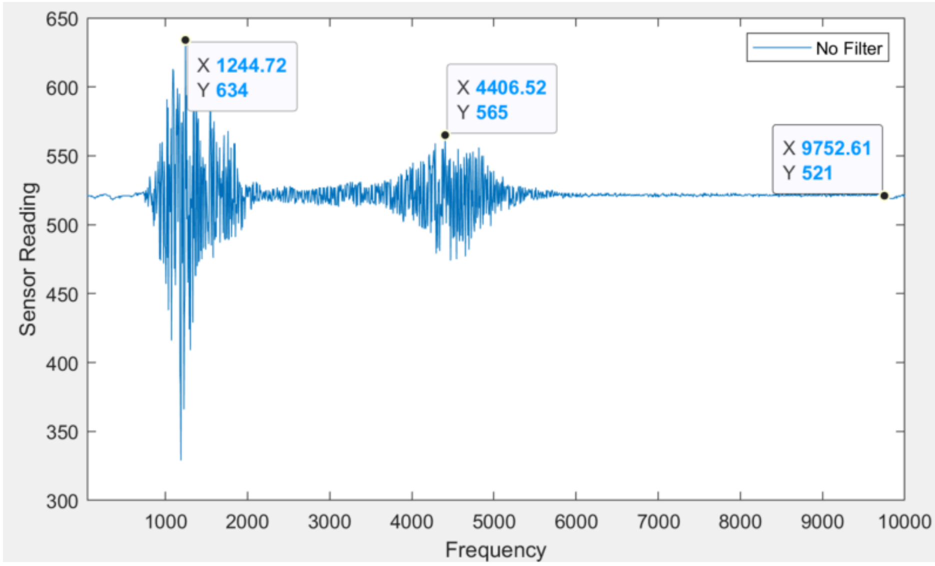

Fig. 2

Question 1:

- Submit a figure similar to Fig. 2 when you do a frequency sweep without analog filtering. Once you complete this figure, please ask the TA to check it off during lab sessions or open hours.
- You need to convert the X axis from time to frequency of the signal as shown in Fig. 2. The X axis is approximately ranging from 50 to 10k Hz.
- You also need to clearly label on the figure the resonance points generating the largest positive deviations as well as a point that represents no-attack sensor readings, as shown in Fig. 2. (Although Fig. 2 has two such resonance points, you may have more and fewer.)

Problem 2. Defense of Analog Filtering

Now you need to insert the RC analog filter between the X axis output and Teensy input port (see the switch in Fig. 0) to get rid of the high-frequency injection signals before they pass through Teensy's ADC. The RC filter has a cut-off frequency around 340 Hz. When it's set up, repeat the same frequency sweep and logging procedure as problem 1 (50 Hz to 10k Hz in 30s). With the new data, you will be able to produce a figure like Fig. 3. Note that you may observe a DC offset between the two traces which is fine.

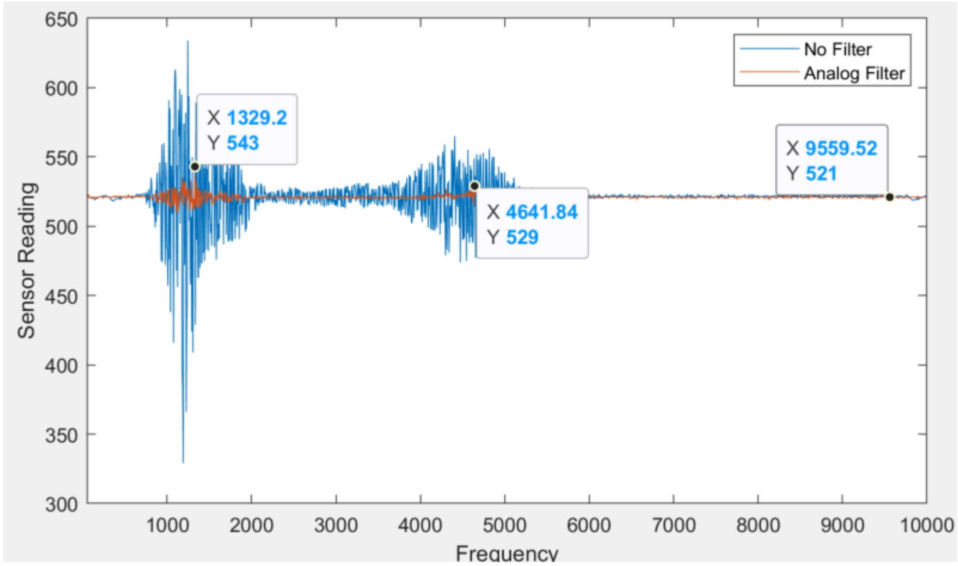

Fig. 3

Question 2:

- Submit a figure similar to Fig. 3 by showing sensor readings both with and without analog filtering simultaneously. Once you complete this figure, please ask the TA to check it off during lab sessions or open hours.
- You need to convert the X axis from time to frequency of the signal. The X axis is approximately ranging from 50 to 10k Hz.
- You also need to clearly label on the figure the resonance points **of the analog-filtering data** as well as a no-attack point with legends, as shown in Fig. 3.

Part 2: Acoustic Injection & Analog Filtering

Note that you do not use the filter in part 2. After finding the resonance points, you can start exploring output biasing.

Problem 3. Output Biasing Exploiting ADC Aliasing

You need to approach output biasing by adjusting the frequency of your injection signal. Note you may not get complete DC outputs due to the instability of the cheap microcontroller's ADC. Try to inject signals as close to DC as possible. It is recommended that you start with a frequency close to the largest resonant frequency you found in question 1. For example, you can start with ~4400 Hz by referring to Fig. 2.

To demonstrate output biasing after you find a reasonable frequency, switch on and off the function generator output for 5 cycles. Each time, keep the injection signal on for at least 3s. You have a limit of 60 seconds to finish these 5 cycles. Log your sensor readings. You will get some readings similar to the ones shown in Fig. 4. **Note that your goal is to continuously**

cause positive or negative deviations (i.e., always larger or smaller than the sensor readings without attack signals) for at least 1.5 second. In Fig. 4, the first attack achieved 10.12-7.76=2.36 seconds.

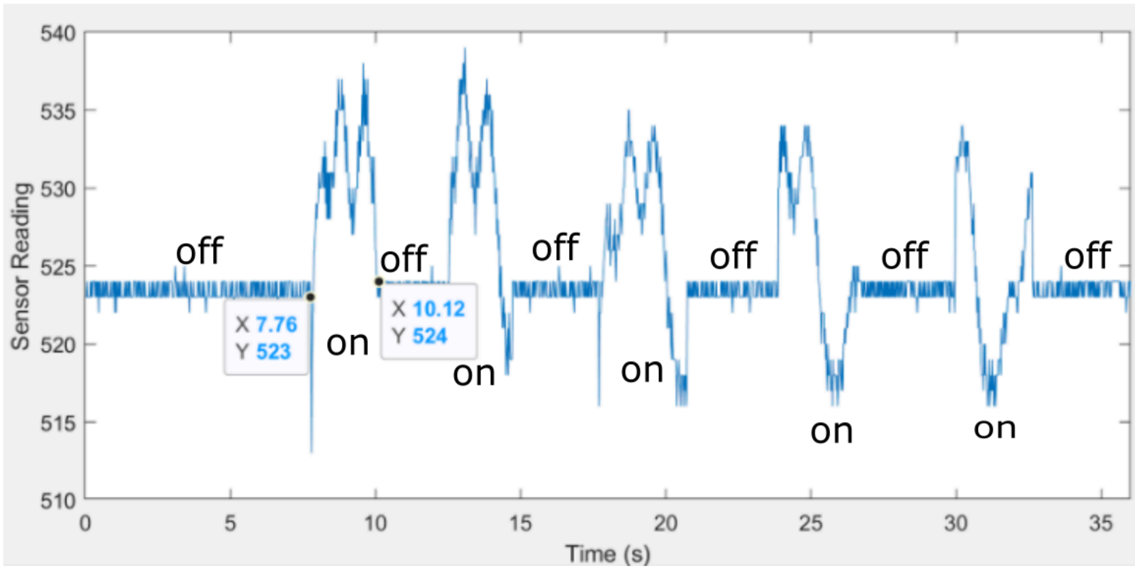

Fig. 4

Question 3:

- Submit a figure similar to Fig. 4 by showing the sensor readings with and without injection attack for 5 times. Once you complete this figure, please ask the TA to check it off during lab sessions or open hours.
- Use time (seconds) as your X axis of the figure. **Note that the data has a sample rate of 50 Hz when you create the time axis.** The figure you submit must have a duration less than 60s.
- You also need to clearly label on the figure two points that show the longest duration of your continuous positive or negative deviation among the 5 attack cycles, as shown in Fig. 4.

Question 4:

- What did you do to achieve output biasing? Describe the process.
- What is the final injection signal frequency you used?