Boston University
Electrical & Computer Engineering
EC463 Senior Design Project

Second Prototype Test Report

On-Board-Sound-Intensity Data Acquisition System (OBSIDAS)

by

Team 8
Volpe

Team Members

Muhamed Abdalla muhabda@bu.edu
Aurojit Chakraborty aurojit@bu.edu
Daniel Cardosi dcardosi@bu.edu
Javier Gonzalez Santamaria javiergs@bu.edu
Joseph Lucatorto III jlucator@bu.edu

March 7, 2024

1. Required Materials

Hardware:

- Personal computer (Macbook Air, Lenovo-Windows)
- Mobile phone (iPhone)
- Desktop computer (located in PHO113)

Software:

- Microsoft Windows (Desktop ECE-SD-T08)
- LabVIEW Community, Version 23.3.1 (Macbook)
- Sonic (iPhone)
- LabVIEW 2017 (Desktop ECE-SD-T08)
  - Sound and Vibration Toolkit
  - NI-DAQmx 20.7

2. Set up

The purpose of this test was to ensure the functionality of the data-collection (excluding the interface with the NI USB-4431 DAQ unit), data-processing (including UI-specific processing), and data-export stages of our software in its current state. All aspects of the set-up followed the previously written test plan, reproduced in the subsections hereafter.

2.1 Computation Stage (non-UI-specific)

The procedure will be twofold: tested first will be the LabVIEW-2017 (Windows) collection/processing virtual instrument (VI), and tested subsequently will be the LabVIEW-2023 (Mac) collection/processing VI, functionally identical to the former but allowing for more convenient sound capture via the onboard microphone of the Macbook. For the LabVIEW-2017 collection/processing VI, the first preliminary step is to configure the input-file paths (displayed in schematic form in Figs. 1 and 2) to specify the (local) locations of the Volpe-provided .wav files, which are calibration signals (1-kHz pure tones) as recorded by each of the four microphones: the inside microphone of the leading probe, the outside microphone of the leading probe, the inside microphone of the trailing probe, and the outside microphone of the trailing probe. Secondarily, it should be verified that the “Call External Library” node is configured to call the “process(...)” function contained in the “mainLib.dll” shared library, which has been compiled from the C++ script responsible for all numerical computations and a statically built version of the FFTW3 DFT library. For the LabVIEW-2023 collection/processing VI, the first step is to configure the “Acquire Sound” subVI (see Fig. 3) to utilize the onboard microphone of the Macbook for the capture of external sounds, the same single-channel input being routed to all four library-node input slots. The second step is functionally identical to that above, though the shared library takes instead the form of a *.dylib file.

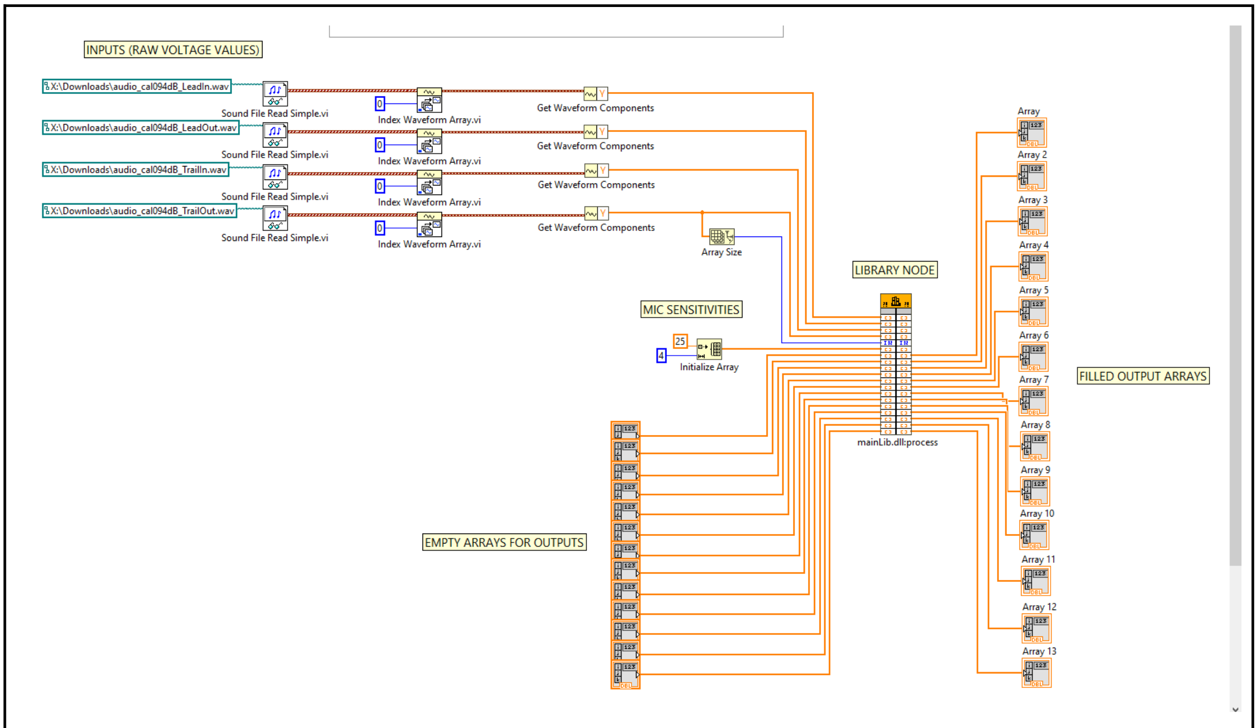


Figure 1. Block diagram of LabVIEW-2017 (Windows) collection/processing VI.

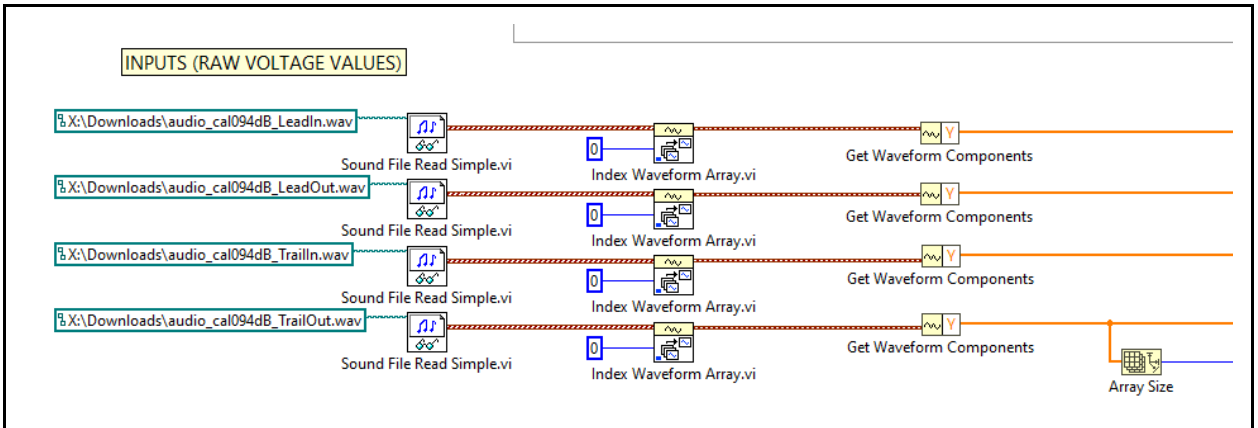


Figure 2. Detail: Input architecture of LabVIEW-2017 collection/processing VI.

2.2 Computation Stage (UI-specific)

Tested will be a single LabVIEW VI containing a prototyped user interface that includes graphical displays of recorded input data, the underlying architecture for which is partly shown in Fig. 4. Data in this test will be simulated by the Volpe-provided .wav files, for which the local file paths should be pre-specified in the VI as above (see section 2.1).

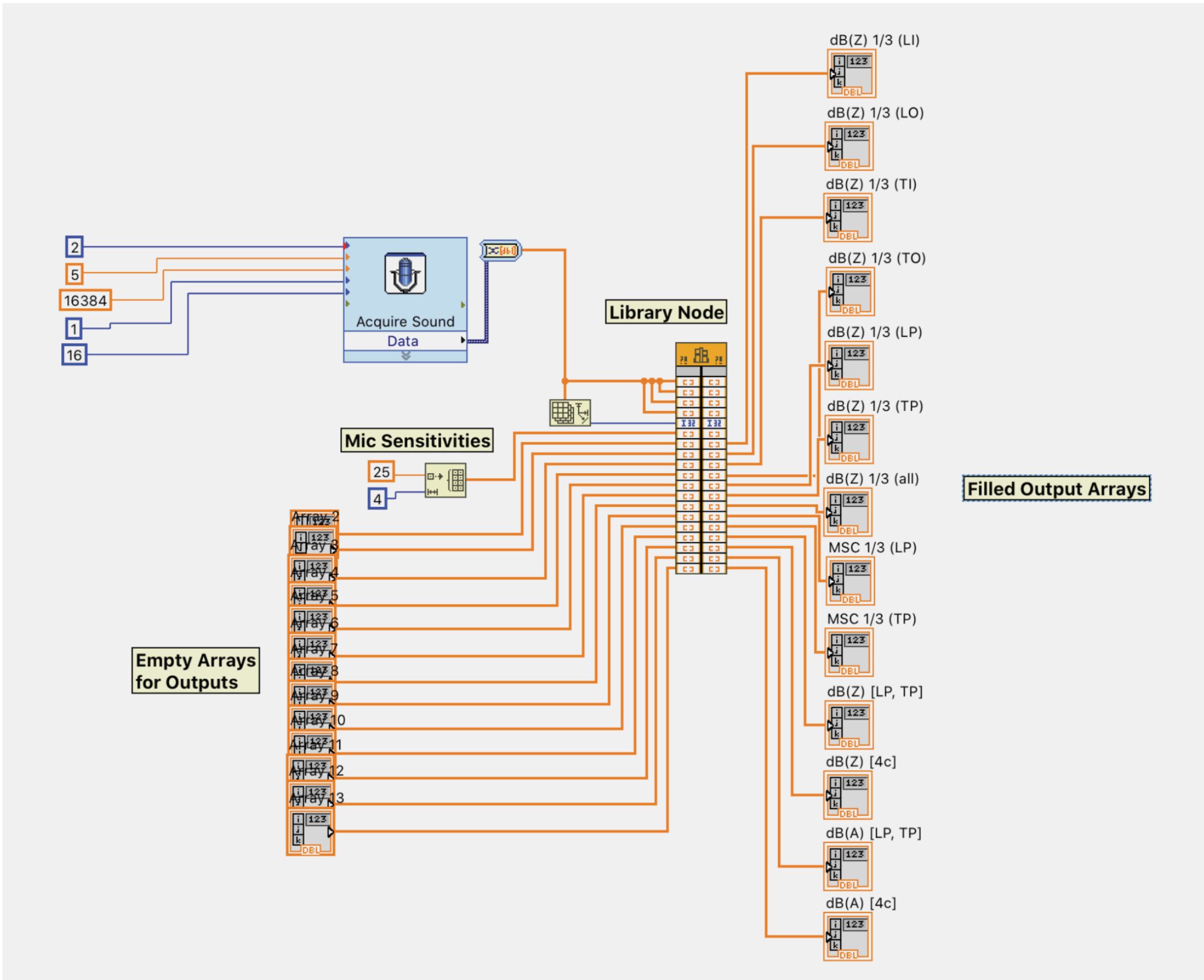


Figure 3. Block diagram of LabVIEW-2023 (Mac) collection/processing VI.

2.3 Software Datapath Infrastructure

For the purposes of this test, data will be simulated by dummy (i.e., random) numerical values. The only required step prior to testing is to select the “Device Acquisition” option (rather than the “File Input” option) under the “Acquisition Source” heading, as is visible in Fig. 5.

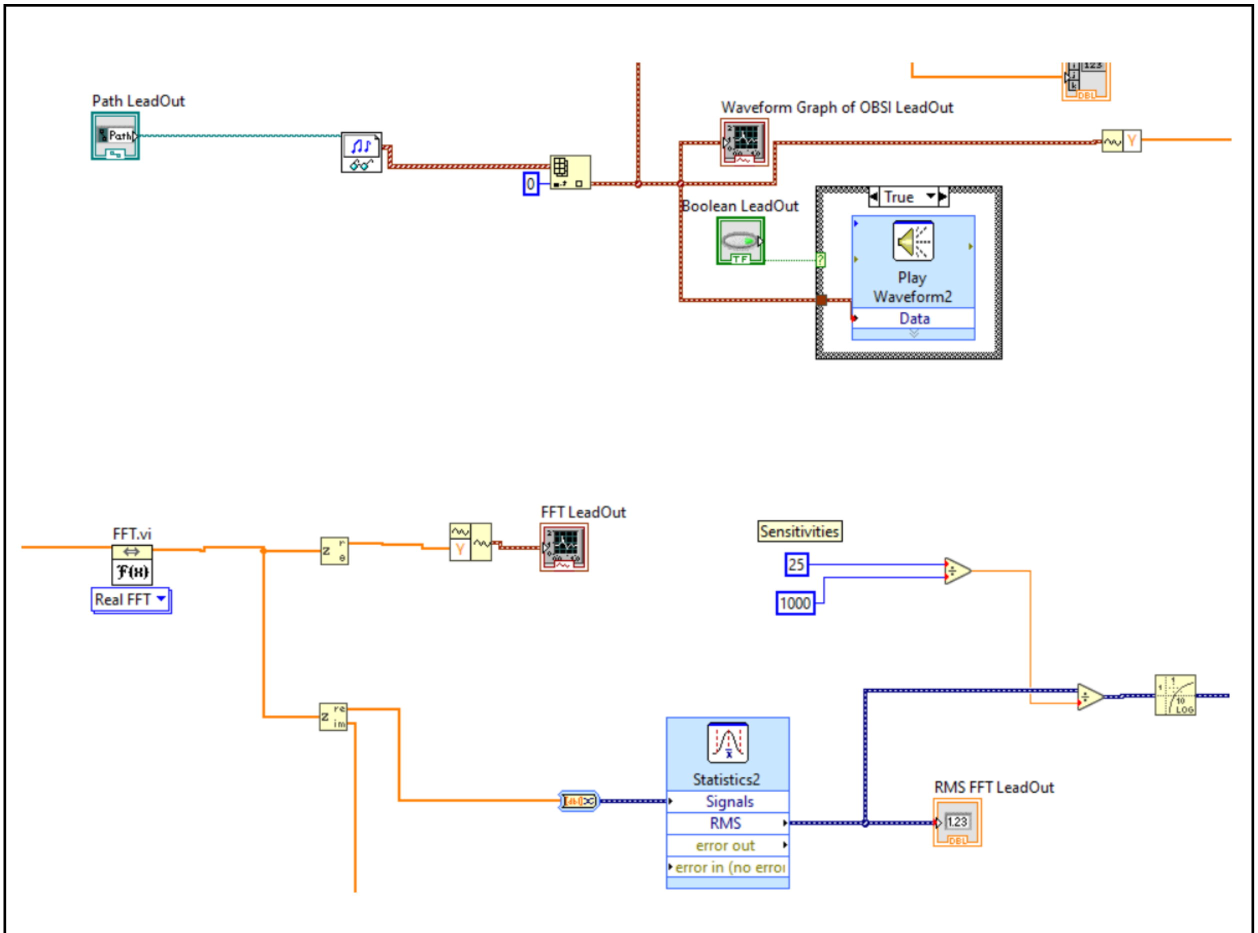


Figure 4. Sections of block diagram of UI-processing VI

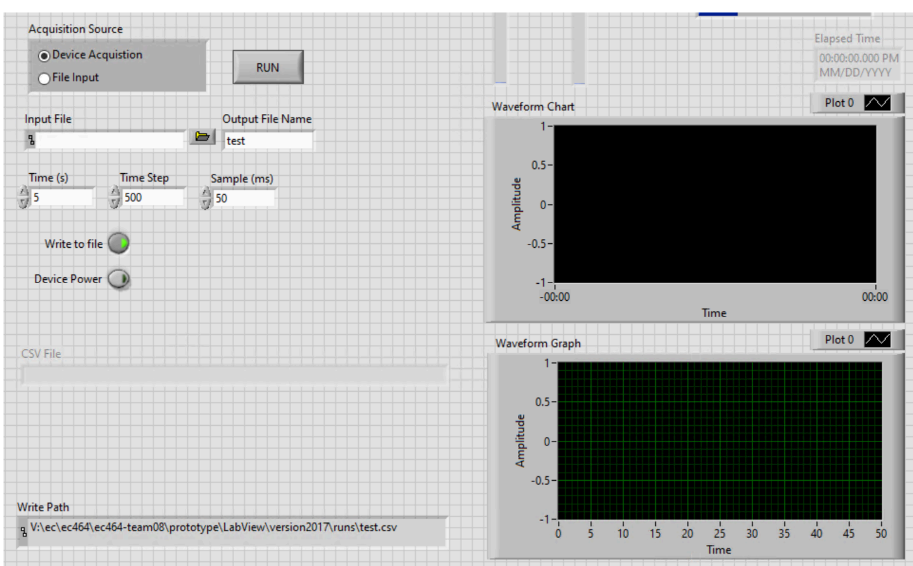


Figure 5. Temporary interface for datapath-control VI

3. Testing Procedure:

After completing the described setup, the testing steps for the collection/processing VIs were as follows:

1. Open and run LabVIEW-2017 VI; verify that outputted values appear on front panel.
2. Open Sonic application on iPhone.
3. Select tone frequency between 222 and 5612 Hz.
4. Determine third-octave band containing selected frequency.
5. Play tone at moderate level, with speaker positioned near microphone (on axis).
6. Open and run LabVIEW-2023 VI.
7. Find maximum-energy frequency bin in outputs by manual searching with index selector.
8. Compare bin indices corresponding to third-octave bands identified in (2) and (6).
9. Repeat (2)–(7) for two more tones, selecting different frequencies such that all those selected together span at least one octave.
   The LabVIEW-2017 collection/processing VI and data-export VI may be tested by completing the above described setup procedures and clicking the buttons labeled “Run”.

The steps for testing the UI-processing VI were as follows.

1. Open and run VI.
2. Verify graphical outputs for each microphone/channel by selecting corresponding tabs on front panel.

The steps for testing the datapath-control VI were as follows.

1. Open VI.
2. Specify filename of file to be outputted.
3. Specify measurement parameters (variable names: “time”, “step”, “sample)
4. Enable “Write to File” for write-back
5. Click “RUN” Button

4. Measurable Criteria

The criteria for the outputted numerical results are as follows:

1. Each of the single-band and entire-spectrum dB(SPL) values should be nonnegative and bounded above by 194 dB.
2. Each of the subband magnitude-squared-coherence values should be 1.

3. Each maximum value of the third-octave-level arrays should be approximately 40 dB greater than the value of either adjacent element (slightly more for tone frequencies above 1 kHz and slightly less for tone frequencies below 1 kHz).
4. Each bin-index comparison (as per step 7) should find no discrepancy between indices.

The criteria for the graphical displays are as follows:

1. For each of the four channels, the plots of amplitude across time should resemble qualitatively those of pink noise, given the nature of tire/pavement noise.
2. For each of the four channels, the plots of DFT coefficients should show Hermitian (conjugate) symmetry, given the real-valued inputs.

The criteria for the exported data file are as follows:

1. The file must have the extension *.csv.
2. The file must show the correct timestamp.
3. The file must be formatted with headers.
4. The file must contain all numerical data intended for export.

5. Results

All aforementioned criteria were met for all tested components. Representative results for each sub-stage (non-UI-specific numerical computation, raw-signal display, DFT display, and data export) are shown in Figs. 6, 7, 8, and 9 respectively.

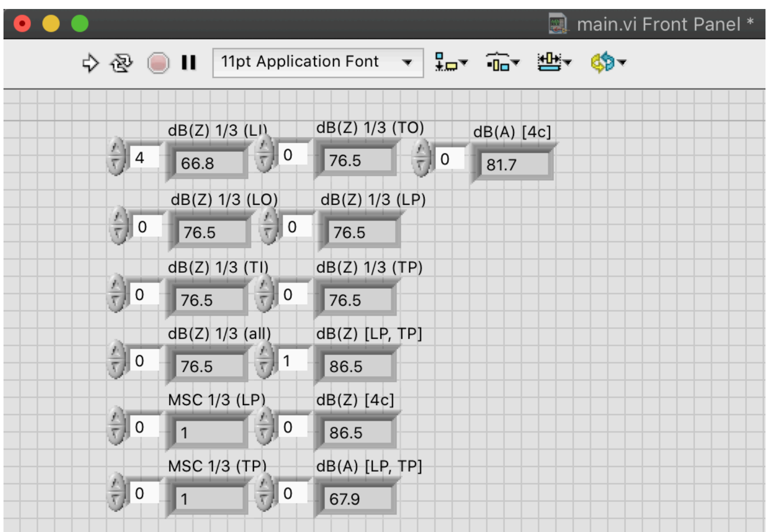


Figure 6. Representative post-trial front panel of collection/processing VI.

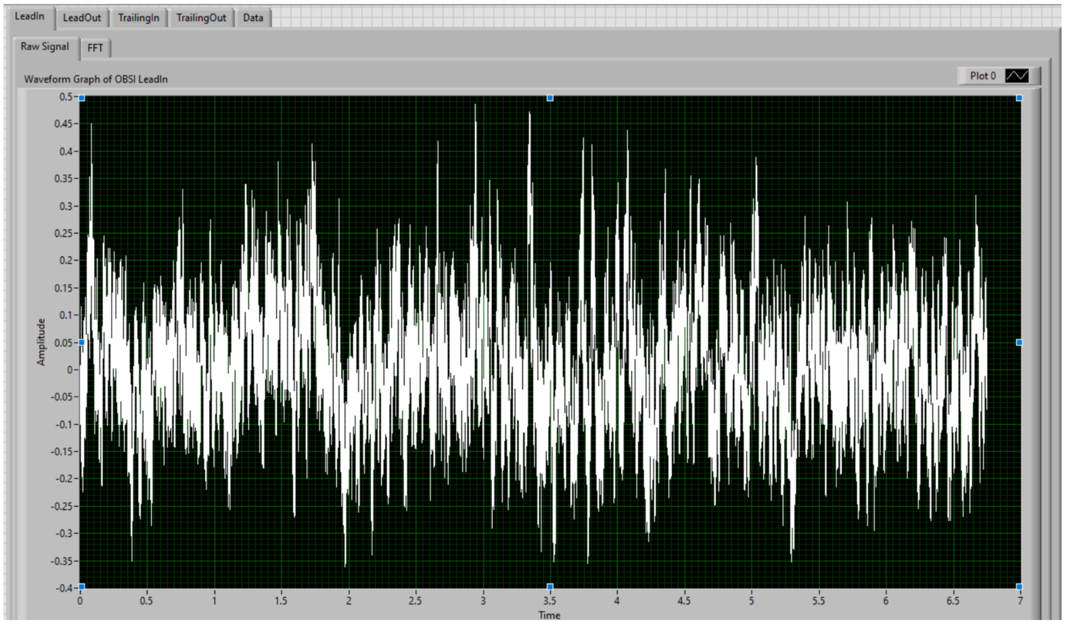


Figure 7. Representative graph of raw input (leading inside microphone)

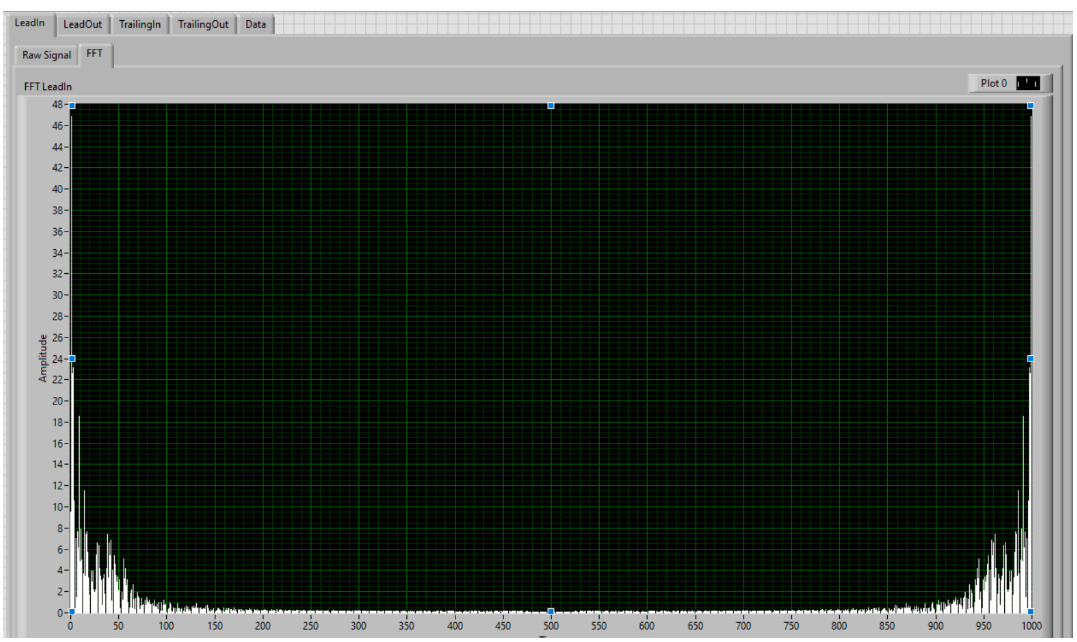
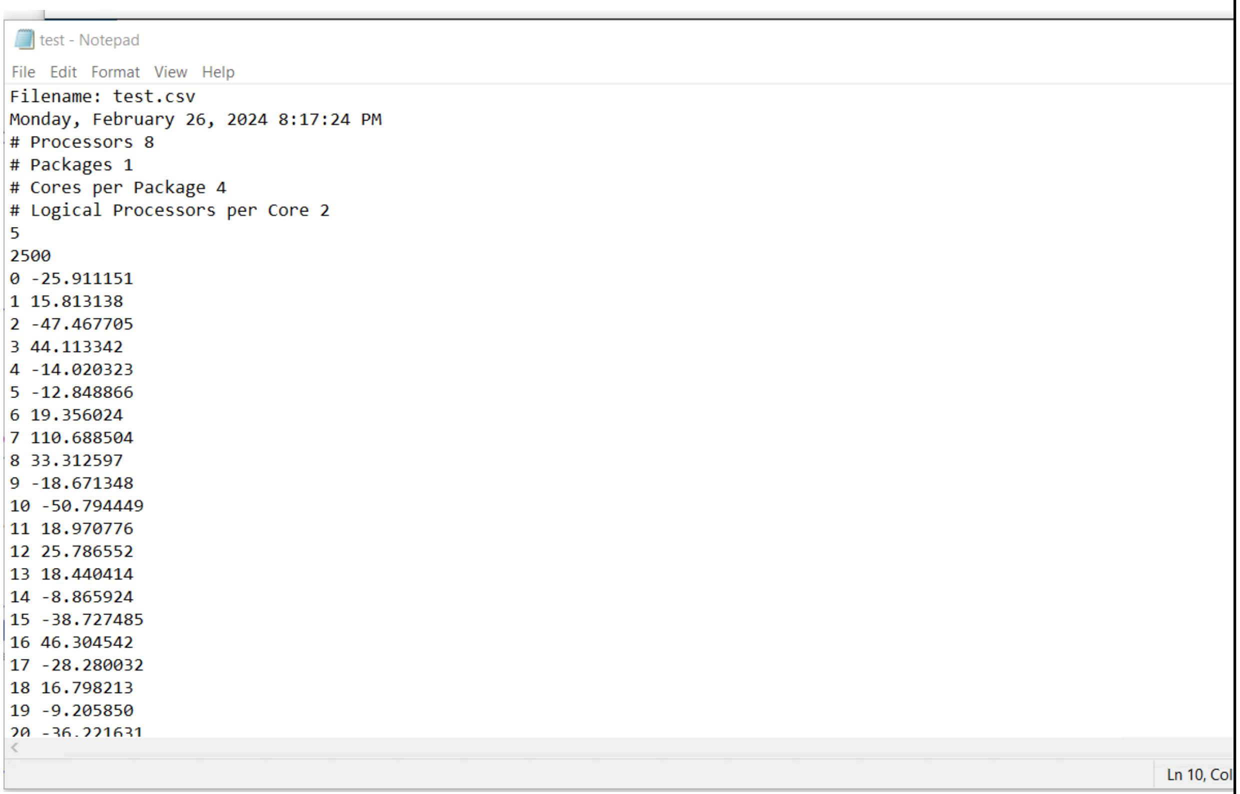


Figure 8. Representative graph of DFT (leading inside microphone)



```
test - Notepad
File  Edit  Format  View  Help
Filename: test.csv
Monday, February 26, 2024 8:17:24 PM
# Processors 8
# Packages 1
# Cores per Package 4
# Logical Processors per Core 2
5
2500
0 -25.911151
1 15.813138
2 -47.467705
3 44.113342
4 -14.020323
5 -12.848866
6 19.356024
7 110.688504
8 33.312597
9 -18.671348
10 -50.794449
11 18.970776
12 25.786552
13 18.440414
14 -8.865924
15 -38.727485
16 46.304542
17 -28.280032
18 16.798213
19 -9.205850
20 -36.221631
Ln 10, Col
```

Figure 9. Dummy data as formatted in output CSV

6. Conclusions

The testing described herein for the On-Board-Sound-Intensity Data Acquisition System (OBSIDAS) prototype was completely successful, validating the functionality of the software components thus far developed.

The single-trial computation VI, utilizing a custom C++-based external library developed previously by our team, was confirmed to correctly return all client-specified metrics (OB14 denoting a collection of fourteen values, each computed in one of the fourteen third-octave-bands with center frequencies ranging from 250 to 5040 Hz):

1. Single-microphone unweighted SPL (OB14); four per trial
2. Single-probe unweighted SPL (OB14); two per trial
3. Across-probe-averaged unweighted SPL (OB14); one per trial
4. Across-probe-averaged A-weighted SPL (OB14); one per trial
5. Inter-microphone magnitude-squared coherence; two per trial
6. Entire-spectrum unweighted single-probe SPL; two per trial

7. Entire-spectrum unweighted across-probe-averaged SPL; one per trial
8. Entire-spectrum A-weighted across-probe-averaged SPL; one per trial

Likewise, the higher-level system-control VI was confirmed to correctly output single-trial input data in *.csv form, and the prototyped user interface was confirmed to correctly generate graphical displays of single-trial input data. The results altogether demonstrate the separate functionalities of the OBSIDAS' major components and thus indicate these components' readiness to be combined into the final client-deliverable LabVIEW program.