Signal Processing Laboratory (ELC 433): Lab 2B: Discrete-Time Signals in the Frequency-Domain using LabVIEW

Stephanie Fournier*
Bryan Guner*
Haley Scott*

* *Electrical and Computer Engineering Department, The College of New Jersey, (e-mail: fournis1@tcnj.edu gunerb1@tcnj.edu scotth3@tcnj.edu)*

Abstract: In this lab, students were required to construct composite signals of varying frequencies and display them into LabVIEW. In order to construct these signals, students were to use LabVIEW's Virtual Instruments and then develop and display frequency transformations, such as Discrete-Fourier Transform, of the discrete time signals.

1. INTRODUCTION

In this lab, we were tasked with building a VI to compute and display the Fourier transform of a 512-point composite signal consisting of four components each having 128 samples. Given the frequencies of the four signals, we were to properly append each signal one after another, and display the waveform with Mathscript as well as virtual instruments. The appended signal functions as the desired composite signal. Once we had taken the fourier transform of the composite signal, we were to display the subsequent waveform.

2. METHODS

In order to output the desired composite waveform, we were to use Virtual Instruments to represent each of the four component signals. Within the VI, we had to input the specifications of the component signals. For the first three signals, we had to input the frequency, sampling frequency, and sample size. For the fourth signal, we used a MathScript to create a chirp signal and specified the frequency range, sampling frequency, and sample size of the signal. These four sinusoids are shown in (1) below. Next we used the append block and wired two signals at a time, in order from lowest to largest frequency, and output the resulting waveform.

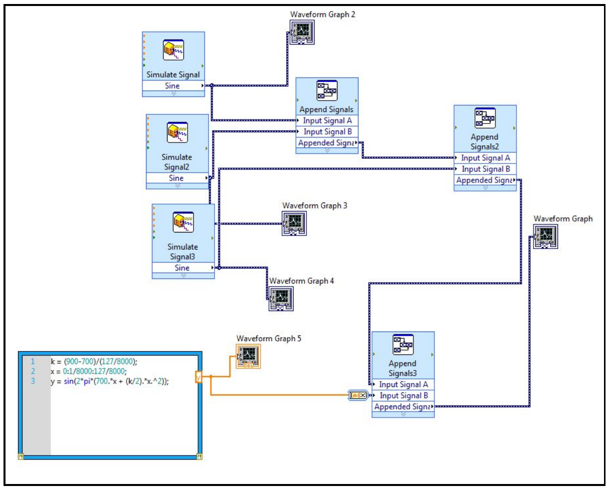

Figure 1: Block Diagram of Four Appended Sinusoids

We then took the discrete fourier transform of the appended signal, through use of the fast fourier transform block, as well as basic arithmetic blocks to ensure graphical functionality. We needed to ensure a proper horizontal axis scale, and used a bundle block to do so. We also utilized a one-dimensional array to successfully graph the fourier transform. The block diagram for our fourier transform function is shown in (3)

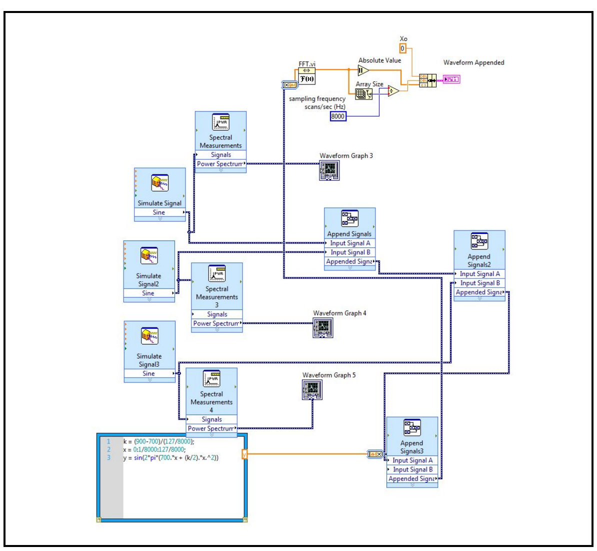

Figure 3: Block Diagram of Appended Signals' Fourier Transform

3. RESULTS

Our team successfully created four sinusoidal signals through the use of both simulate sine blocks, as well as Math Scripts. Our team then sent the appended signals' output to a time domain waveform graph, shown below in (2).

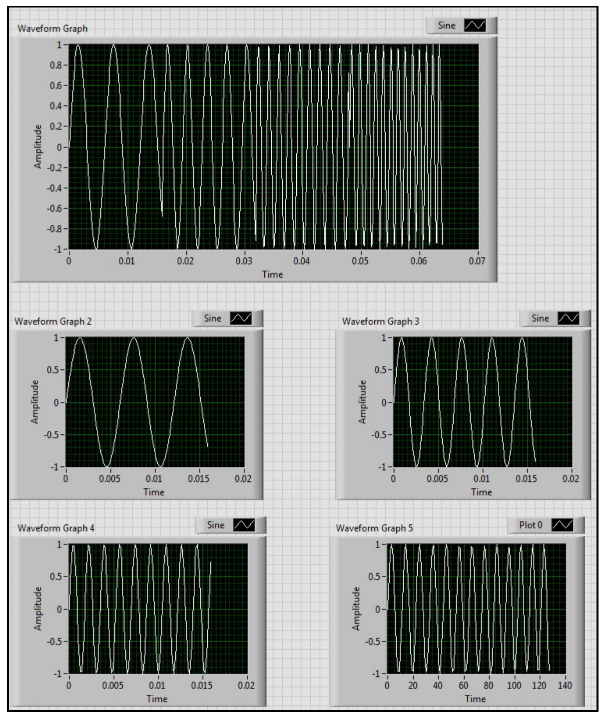

Figure 2: Output of Individual Sinusoids and Appended Signal

We were able to take the fourier transform of the appended signal, and graph the output. The frequency domain results are shown on our front panel view below (4).

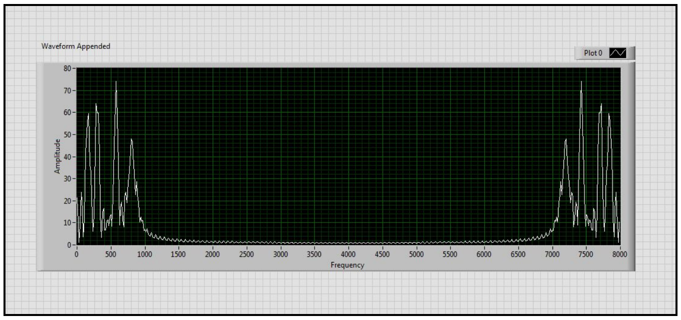

Figure 4: Frequency Domain Output Graph of Appended Signal

4. DISCUSSION

Through the use of LabVIEW's digital signal processing tools, our team was able to produce and analyze a fourier transform of an appended signal. Because the lab was instructional and did not present a 'problem' to be solved, there were no theoretical values to compare with our results. We were able to verify that the signal's output aligned with some characteristic qualities of a frequency domain sinusoid, such as the higher peaks at 8,000Hz and at the various frequencies of each signal.

5. CONCLUSION

This lab was successful in that we were able to complete each task to the required specifications. Our team was able to create and display the composite waveform from the four component signals using only Virtual Instruments and MathScript blocks. We also computed, displayed, and verified the Fourier transform of the appended signal.

7. REFERENCES

Winser, Alexander. Williams, Cranos. (2016). *Digital Signal Processing: Principles, Algorithms and System Design*. Academic Press, Cambridge, MA.