

Experiment 2: Digital data acquisition and analysis

In this lab you will learn the basic techniques of digital data acquisition and analog to digital conversion. In many physics experiments, a measurement is made, which results in an analog voltage signal proportional to the observable of interest. In general, one would like to be able to measure this signal as a function of time at a specific data acquisition rate, which depends on the nature of the observed voltage. If the signal is changing rapidly in time, fast (and expensive) electronics must be used to record the signal. Signals that change on the time scale of milliseconds are fairly easy to record, while faster signals require additional effort. A signal that is fast in the time domain has a high frequency, whereas slow signals can be observed at lower frequencies. To quantify this, spectrum analyzers are often used to take a signal observed in the time domain and perform a Fourier transform to produce the frequency spectrum of the signal.

High frequency spectrum analyzers are very expensive, but at frequencies in the kHz range the analog to digital conversion capabilities of our National Instruments data acquisition (DAQ) cards can be used to perform a Fast Fourier Transform and transform a time-varying voltage signal into the frequency domain. Because this tool is used very often in experimental physics, we will develop a virtual instrument to display and analyze the spectrum of your input signal.

1. Programming in Labwindows

Before you begin, create a new folder in the Student Directory with your name and your partner's name in the title. Save all newly created files to this directory.

In order to read data from your DAQ card, you must first write a computer program that will allow you to interface with the card. The programming environment we will use is Labwindows, which was already introduced in Experiment 1.

Before designing your code to access the DAQ card, you will need to design your virtual instrument. To do this, think about what the user will input into the program and how to display the results of a measurement. The user interacts with the program by entering numbers into boxes and by pushing buttons and changing switches. The first object you should create is a quit button. This button will end the program when activated by the user. Next, the user will need to tell the program when to acquire data and how to acquire that data. This will require an "acquire data" button as well as several numerical input boxes that specify the time over which to acquire the data and how frequently the data is acquired. Finally, after data acquisition is complete, the program should display the data for the user to evaluate. Because we will be acquiring the data in the time domain, taking its Fourier Transform, and displaying its frequency spectrum, you will need two graphs to display the time and frequency data.

To learn how to acquire data in Labwindows, you will need to search through the Labwindows manual for information on data acquisition. Labwindows has built-in drivers for the DAQ card, so you will find specific functions in Labwindows that will

allow you to acquire data. Because we are now using Windows Vista, the “Traditional NI-DAQ” functions are no longer available. Therefore, we will need to use the new National Instrument DAQmx functions to access the data from the card. This is because the traditional commands required direct hardware access, and newer versions of Windows do not allow direct hardware access for security and stability reasons.

In general, data acquisition programming with DAQmx involves the following steps:

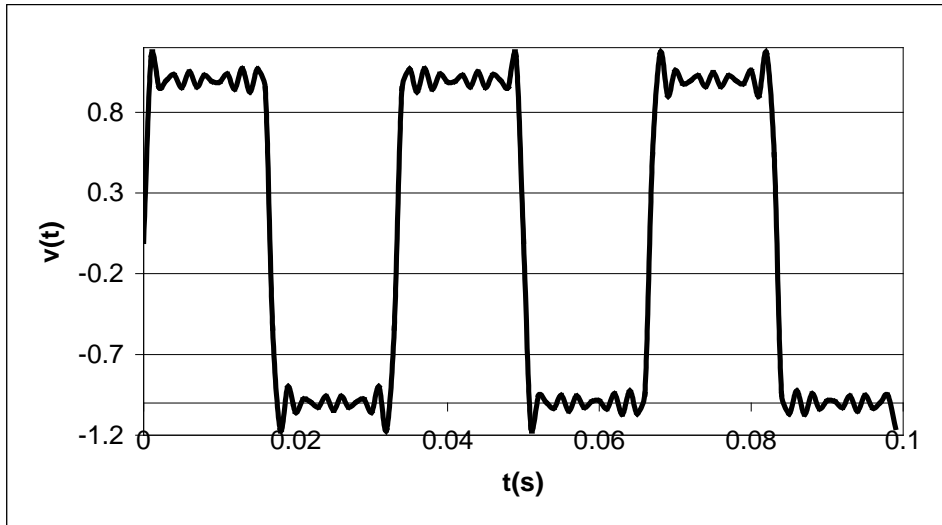
- Create a Task
- Create Virtual Channels
- Configure the Timing Parameters
- Start the Task
- Perform a Read or Write operation from the DAQ card
- Stop and Clear the Task

The functions for performing these tasks can be found in the DAQmx library and they all have self-explanatory names. For example, `DAQmxCreateTask` creates the task with a variable assigned to the task to be used throughout the process, `DAQmxCreateAIVoltageChan` sets up the parameters for analog voltage input, `DAQmxStartTask` loads the task with these settings, and `DAQmxReadAnalog64` reads the analog voltage into an array using the previously specified settings for the task.

Using the appropriate commands, create a virtual instrument that allows a user to acquire voltage data over a user-specified time at a user-specified frequency. The program should display two graphs containing the data and its Fourier Transform, and the graph should refresh each time data is acquired. The Fourier Transform is a class of special functions that you should be able to find in the Labview libraries. It is easiest to use a function that displays the magnitude of the voltage, such as the power spectrum function.

2. The Fast Fourier Transform

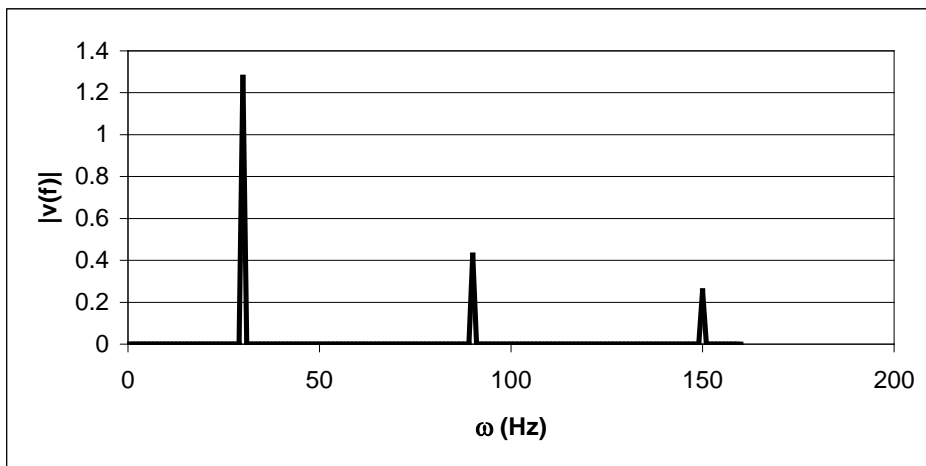
Within your program, the data will be input into a function called the Fourier Transform. This function transforms your data from the time domain into the frequency domain. For example, consider the following square wave and its Fourier Transform



This “square wave” was generated from the function

$$v(t) = \frac{4}{\pi} \sum_{n \text{ odd}} \frac{\sin(n\omega t)}{n}, \text{ where only the first 6 terms were used.}$$

The frequency spectrum will look like



3. Application of the spectrum analyzer: Fourier analysis of a sine, triangle, and square wave.

Connect the output of a function generator to an analog input of your DAQ card on the junction box. Set the output voltage to 1 V, the wave type to a sine wave, and the frequency to 100 Hz. Set your spectrum analyzer program to acquire data for 10 msec at a sampling frequency of 1 kHz. Your program should display the sine wave and its Fourier Transform. If it worked, while your program is active press “alt-print screen” and a picture of your results will be saved to the clipboard in Windows XP. Open an

MSWord document and paste the picture into your document by choosing paste from the edit menu. The picture you have created is referred to as a “screenshot”. This is your proof that your application works. Make sure the application has your name and your partner’s name on the front panel so we know it was your application. Increase the sampling time to observe several periods of the sine wave. Does your FFT signal change? Do the same for a triangle and square wave using your function generator. Take screenshots in each case and save them.

4. Data extraction: writing your data to a file

When taking data in experimental physics, it is never enough to have a program that displays the raw data. You need to create a file that contains the raw data so you can take that data later and manipulate it using MS Excel or some other analysis program. Here you will modify your program by adding a “save to file” command to your virtual instrument. Refer to the Labwindows documentation for a function that will allow you to save your array to a file. Write your program so that it outputs two columns for each data set. For the time data, these columns will be time and voltage. For frequency, it will be frequency and $v(f)$. In Excel, each column is separated by a tab. Repeat the experiments above for one type of signal and plot the data in Excel. Label your axes and add a title. Save this file to import into your lab report later.

5. Data sampling and aliasing

If the time between sampled data points is given by τ , the Nyquist frequency is defined as

$$f_{Nyquist} = \frac{1}{2\tau}$$

This is the maximum frequency that can be determined from data taken at a given sampling rate. If data is input at the Nyquist frequency, the observed frequency will be correct, but the amplitude information cannot be obtained. Data input at twice the Nyquist frequency results in the loss of both amplitude and frequency information (i.e. sampling at exactly the signal frequency gives a constant signal as a function of time.)

This means that for a given sample rate, all components of a signal that occur at a higher frequency will not be properly sampled. In fact, they will appear to be a much lower frequency. This is referred to as aliasing. In practice, one must always filter out all signals that occur at frequencies higher than the Nyquist frequency for a given sampling rate. Experiment with different sampling rates that illustrate aliasing. Be sure to note the input frequencies you are using, the sampling rates, and resulting FFT signal. Save this data for your report. In your report, explain the mathematical relationship between the observed frequency, input frequency, and sampling frequency (or Nyquist frequency).

6. Writing your lab report

One of the most important skills for an experimental physicist is the ability to communicate your experimental results. Your lab report should describe your experiment step by step and relate the experimental results and methods quantitatively using words

and graphic illustration. This report should contain an introduction describing the goals of the experiment and the methods used. This should be followed by a “Results and Discussion” section, with a description of the methods used and the data acquisition program you wrote. It should contain the text of the program and a screenshot of the virtual instrument you have created. Screenshots of the results when acquiring data for sine, triangle, and square waveforms should be presented in your report. Graphs made in excel showing these results should also be presented. You should discuss the values obtained in the graph and how they relate to the expected behavior of a Fourier Transform. Give three examples of aliasing from your experiments and explain what is happening, comparing the observed signal with the input signal. Finally, your conclusion should state what you have accomplished in the experiment and what values you measured.