Final Project: 2-Channel General-Purpose Data Acquisition Program

ME 4380/5380 Introduction to Data Acquisition and Signal Processing
Spring 2022
Instructor: Dr. Steven R. Anton

Final Project Description

Goal: To develop and fully document a general-purpose data acquisition program capable of continuously acquiring up to 2 channels of analog voltage data, scaling and displaying the data appropriately, and saving the data to a text file. The code should be usable by a LabVIEW novice.

LabVIEW Program Requirements: The LabVIEW data acquisition program has several requirements as follows:

- **Hardware:** The code must be compatible with any NI DAQ device that has multiple AI channels. That said, you should use your myDAQ to test the code's functionality.
- **Required Measurements:** The program must acquire data from up to 2 analog input channels.
  - The channels will be from the same DAQ device, but do not have to be sequential (e.g. could be Dev1/ai2 and Dev1/ai5).
  - The data must be displayed graphically and plotted against current time.
- **Required Data Processing:**
  - The program must have the option to individually scale the data *on a per-channel basis* as follows:
    - Scale the measured voltage signal to millivolts prior to plotting/saving the data.
    - [5380 Students only] In addition to scaling the measured voltage signal to millivolts, the user must also have the option to scale to the following 6 Engineering Units (EUs): acceleration in m/s^2, acceleration in g's, sound pressure in mPa, sound pressure in Pa, force in mN, and force in N prior to plotting/saving the data.
  - The program must calculate and display the Fast Fourier Transform (FFT) of the measured signal(s).
  - [5380 Students only] The program must have an optional IIR filter that can be applied to the data *on a per-channel basis* before the FFT, with user controls to select the filter topology, type, order, cutoff frequency, passband ripple, and stopband attenuation individually for each channel.
- **General Program Functionality:** The program must operate based on the following:
  - Once running, the program must wait for the user to press a button and then begin gathering and displaying data on a graph – CPU usage should be negligible before the user presses the button to request the acquisition to begin.
  - The program must allow the user to collect from either of the two ai channels, or both.
  - The program must have two modes of acquiring data:
    - Continuously acquire data until a user presses a button to stop collecting data.
    - Collect data for a user-defined amount of time (in seconds). In this mode, the program will stop acquiring data either 1) after the specified time has elapsed, or 2) when the user presses a button before the set time has fully elapsed.
  - The program must ask the user if they wish to save the data immediately after data acquisition has ended. Depending on their answer, the data either gets saved to disk at a location of the user's choice, or discarded. If discarded, you must ensure that the data is completely removed from memory or temporary storage.

    - The program must continuously run until the user presses a button which will end the VI. This means the program will allow for multiple data acquisition "runs" without having to be restarted.
- **User Interface Functionality:** The user interface (LabVIEW front panel) must include several features:
    - User input to select which channel(s) to measure from, and the sampling rate of the DAQ (control(s) must be disabled after data acquisition has begun).
    - User input for the scale factor(s) (4380: V or mV; 5380: V, mV, m/s^2, g, mPa, Pa, mN, or N) (control(s) must be disabled after data acquisition has begun). Recall, scale factors in sensor datasheets are often given in terms of "output"/"input", i.e. mV/EU (EU = Engineering Unit).
    - User input to control the data acquisition process. Appropriate controls should be available for the user to choose whether to continuously acquire data or acquire data for a user-defined amount of time, and to stop the acquisition in either case (appropriate control(s) must be disabled after data acquisition has begun).
    - The y-axis label of the graph(s) must update to reflect the proper units of the data, based on the selected scale factor(s).
    - Indicator lights to show if the program is idling or recording data, or other states of the program.
    - The graph must be cleared the first time the VI is run (before acquiring data the first time).
    - A popup box after data acquisition stops that gives the option to save or discard the data.
    - [5380 Students only] User input that allows the user to select whether or not to include a column of time data in the saved text file.
    - [5380 Students only] A progress bar that is visible only if a user-defined amount of time for acquiring data is chosen – the bar must fill as the program acquires data (filling completely when the total amount of time has elapsed).
- **Required Data File Contents:** The saved data file must be tab delimited and include the following:
    - Useful header information. This includes the: name of the VI, date/time of measurement, name of the operator making the test (obtained from a user input), name/description of the item being tested (obtained from a user input), notes about the test (obtained from a user input), channel label(s), channel units, the sampling rate, and any other information about the measurement you deem useful.
    - The raw voltage data and the scaled data for each channel, organized as columns of data.
    - [5380 Students only] A column of time data (if the user selected to write this data to the file).
- **General Requirements**
    - The program must use State Machine, Producer/Consumer, and Event Driven User Interface design patterns.
    - The code must be organized within a LabVIEW project.
    - The code must include at least one subVI (but can have as many as you see fit).
    - The code should be usable by a LabVIEW novice who is well-versed in data acquisition (i.e. someone who understands sampling/aliasing/FFT *theory*), but who has never used LabVIEW before.

Deliverables

1. A LabVIEW project file and all associated VIs zipped into a single file and submitted to iLearn.
2. A "user manual" for the code submitted to iLearn.
    - This should be written so a LabVIEW novice can use your code.
    - It must fully document the use of your code, including useful screenshots of the front panel.

3. A final 10-min presentation for the class where you describe your program's operation and perform a live demonstration of your code to the class.

Due Date

Deliverables 1 & 2 are due by 5pm on Monday, May 2nd.
Deliverable 3 will be presented live during our final exam time: Thursday, May 5th, 10:30am-12:30pm.