Introductory Motion and Control
Data Acquisition Software for Step Excitation

Introduction

- These notes describe LabVIEW software that can be used for ***data acquisition***. The overall software characteristics are described.
- This software was written specifically to interface with electro-hydraulic trainers. It is designed to interface with National Instruments (NI 6052E or NI 6251) data acquisition cards to ***send*** a ***step voltage*** command to a ***linear proportional valve*** and to ***read*** the ***command signal*** and the ***valve spool*** and ***hydraulic cylinder positions***.

Front Panel

- The front panel of the LabVIEW ***data acquisition*** (open-loop control) program for a ***step excitation*** is in Fig. 1. The ***Signal Output AO*** box (upper left) specifies the ***magnitude*** of the ***step voltage*** command to be sent to the valve.
- The program assumes this command is then available on ***analog input Channel 2***, and the ***hydraulic cylinder*** and ***valve spool*** position signals are available on ***analog input Channels 0*** and ***1***, respectively.

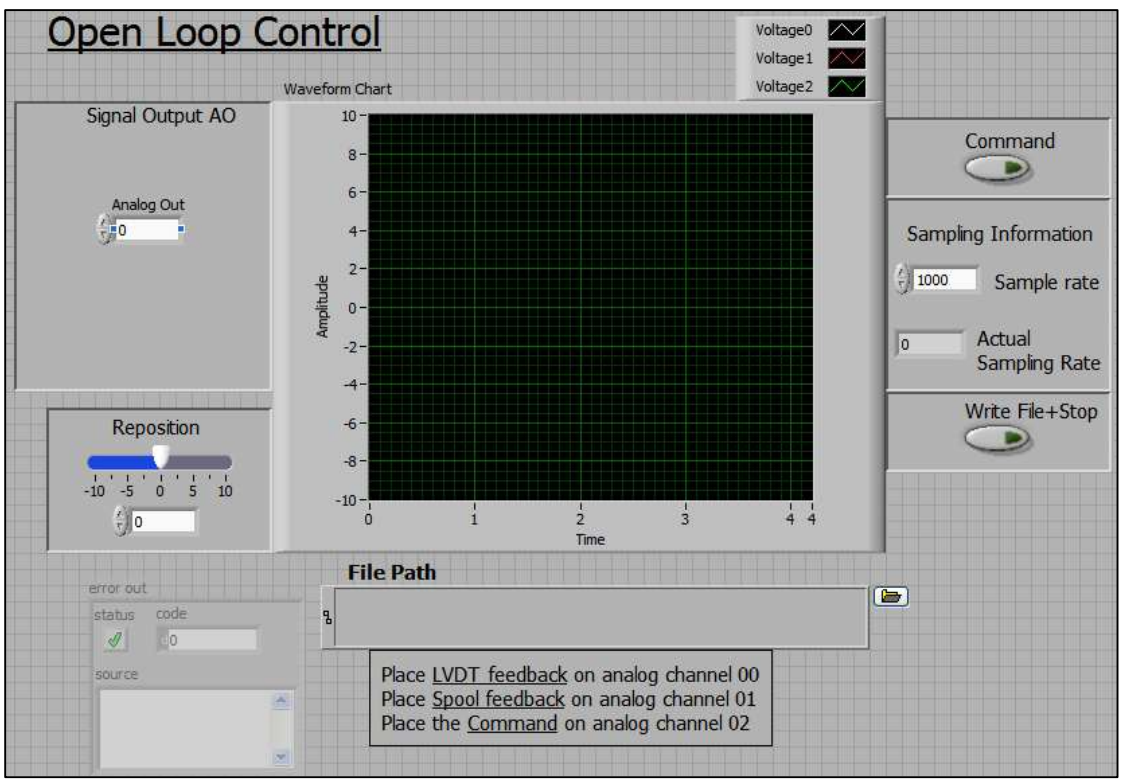

Fig. 1. The Front Panel (Graphical User Interface)

- The ***rate*** at which the analog input channels are ***sampled*** (in samples/sec) is shown in the ***Sampling Information*** box on the right. Just below this input box is the ***actual*** sampling rate achieved as determined by the LabVIEW program.
- The collected data is ***stored*** in a ***text file*** as specified in the File Path input box. The ***file folder icon*** on the right of the box allows the user to browse for a file location. The ***size*** of the files generated is determined by the sampling rate and the total time duration of the measurement.
- Once the information for ***signal output***, ***sampling rate***, and ***file path*** is entered, the user ***starts*** the program ***execution***. The command signal will be sent, and sampling of data will begin only after the user clicks on the ***Command*** button (turning it ***ON***). Data collection will continue until the user clicks the ***Command*** button a second time (turning it ***OFF***).
- When the ***Command*** button is ***OFF***, the program will send the constant ***Reposition*** voltage as indicated on the front panel, allowing the user to reposition the hydraulic cylinder to a convenient starting point.
- After repositioning the hydraulic cylinder, clicking on the ***Write File + Stop*** button will allow the data to be ***written*** to a ***file*** and program ***execution*** to ***stop***.

The Block Diagram

- The LabVIEW ***code*** is depicted in the form of a block diagram. Descriptions of the basic operation of the various segments of code follow.

<u>Analog Input Channel Initialization</u>

- At the writing of these notes, the most recent form of data acquisition available in LabVIEW is called ***DAQmx***. LabVIEW includes many subroutines (VI's) to support this feature. Fig. 2 shows the first segment of the code that utilizes ***DAQmx*** VI's to ***create*** a data acquisition ***task*** and ***initializes*** the analog ***input channels***.
- The ***initialization*** of each channel includes defining the ***physical device*** (NI 6052E or NI 6251), the ***channels*** (in this case, analog input channels *0, 1,* and *2*), the ***minimum*** and ***maximum*** voltages for each channel, the ***type*** of measurement system (reference single-ended (***RSE***) in this case), the ***sampling rate***, and the ***sample mode***.
- The ***sample mode*** determines whether the sampling is based on ***software*** or ***hardware timing***. (The most accurate form of timing is hardware timing.) This code utilizes continuous sampling mode which is ***hardware timed***. The ***DAQmx Timing*** block measures the ***actual*** sampling rate achieved. This rate is ***displayed*** in the output box in the Sampling Information section on the right side of the front panel.

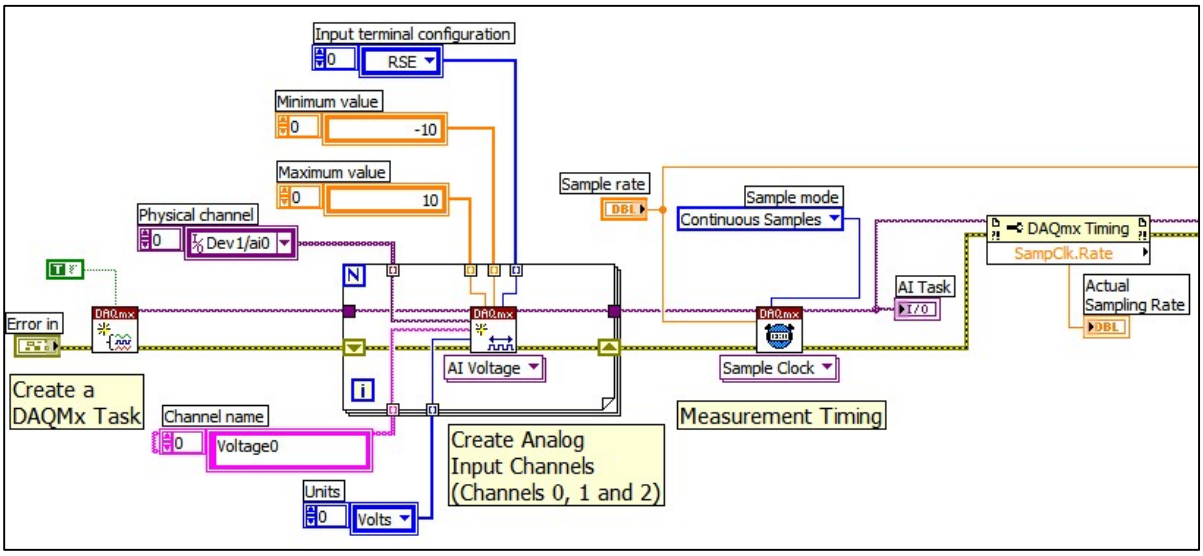

Fig. 2. Segment to Create DAQmx Task and Initialize Analog Input Channels

<u>Analog Output Channel Initialization</u>

- The ***initialization*** of the analog voltage ***output channel*** includes defining the ***physical device*** (NI 6052E or NI 6251), the ***channels*** (in this case, just channel *0*), the ***sampling rate***, and the ***sample mode***. The sampling rate is ***carried forward*** from the analog input initialization, and as before, continuous sampling mode is chosen to guarantee hardware timing.
- As wired, the ***sample clocks*** for the analog input and analog output tasks are ***shared*** so the two tasks will be ***synchronized***. The analog ***output task*** is then ***started***; however, since it is ***tied*** to the sample clock for analog input, analog output will not occur until the analog input task is started.
- Note also in this section that the ***data file*** is ***opened*** (or created) before the data acquisition is started. The block diagram for this section of the block diagram is shown in Fig. 3.

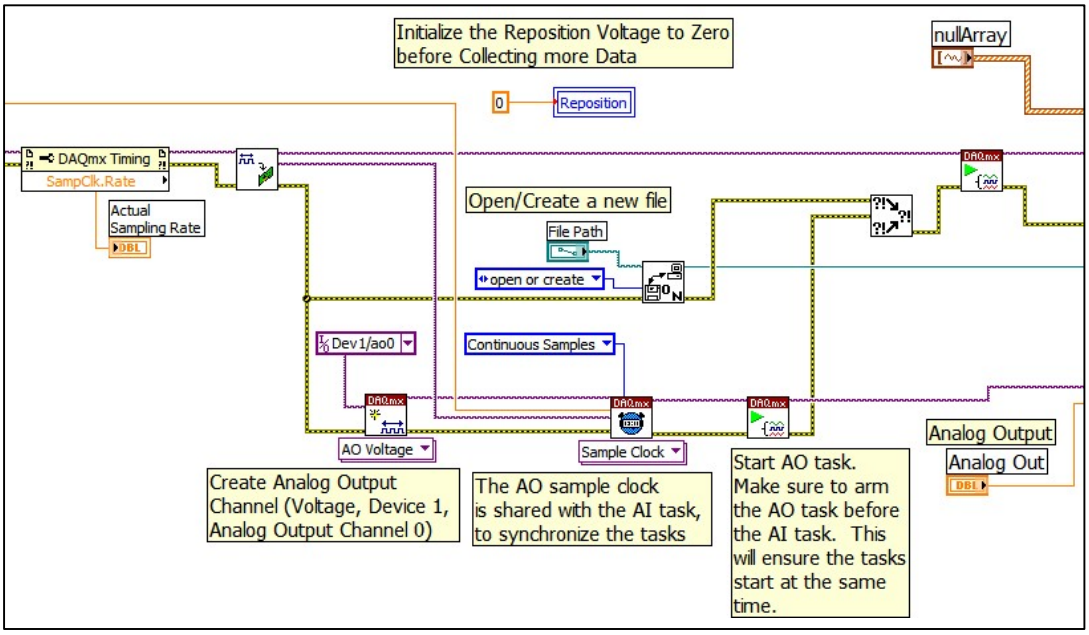

Fig. 3. Segment to Initialize the Analog Output Channel and Synchronize it with the Analog Input Channels

Data Acquisition and Cylinder Repositioning

- Just before the data acquisition ('while loop") loop is executed, the analog ***input task*** (and consequently the analog output task) is ***started***, a ***null array*** is defined to store the collected data, and the ***analog output*** value from the front panel is ***identified***.
- As execution enters this loop, the ***Command*** button should be ***OFF***, so the ***reposition*** voltage is sent to the analog output channel, and data is collected and shown on the ***Waveform Chart*** on the front panel. The block diagram of this segment with the Command button OFF is shown in Fig 4.
- Clicking the ***Command*** button (to turn it ***ON***) ***sends*** out the analog output voltage ***command*** and ***collects*** the response data. The data is ***appended*** to the ***null array***. Data is collected by the DAQ card using ***hardware timing*** and stored in a ***memory buffer*** on the card. Each time the "while loop" is executed, the software reads a certain number of samples from that buffer.
- The ***number of samples*** read from the buffer is listed just above the ***analog read task icon***. From our experience, that number should be about ***one-tenth*** of the ***sampling rate***. If this number is ***too small*** or ***too large*** relative to the sampling rate, ***data will be lost***, and the data acquisition process will be compromised.

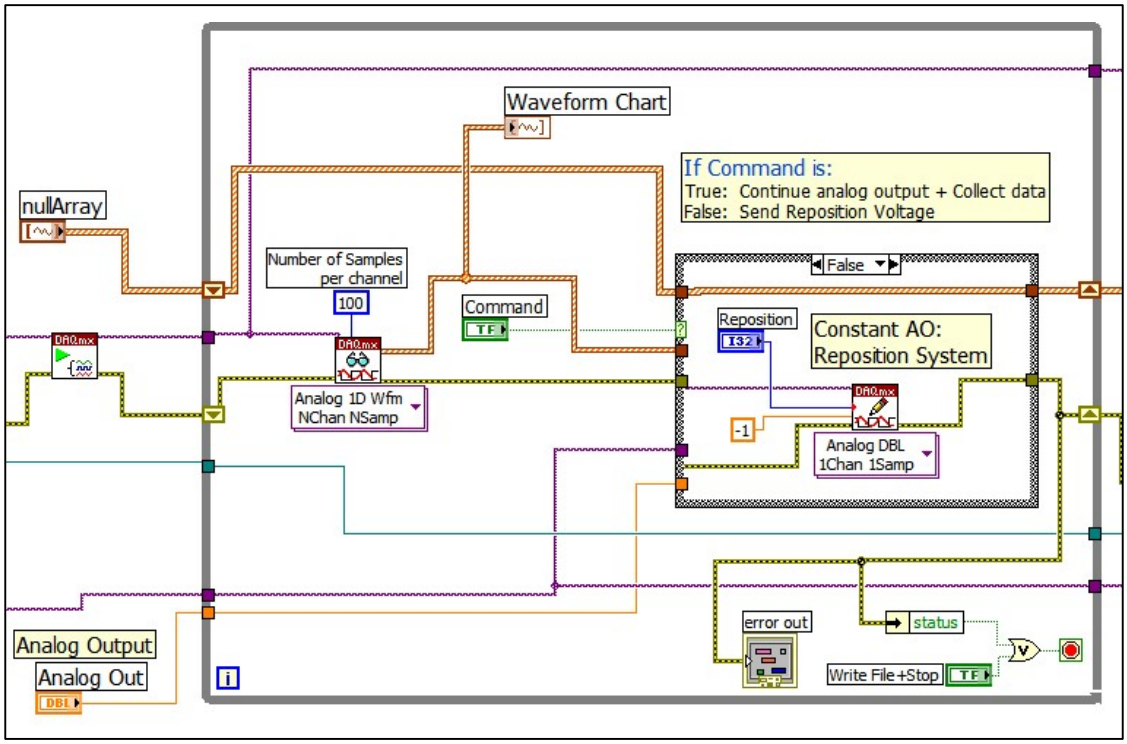

Fig. 4. The Data Acquisition Segment with the Command Button OFF

- The ***amount of data*** collected (and size of the data file) is dependent on the amount of time the user leaves the Command button ON. The block diagram of this segment with the Command button ON is shown in Fig. 5.
- Click the ***Command*** button ***OFF*** to ***stop data acquisition***. Then use the reposition voltage to reposition the hydraulic cylinder. After repositioning the cylinder, the reposition voltage should be reset to zero.
- Click the ***Write File + Stop*** button on the right of the front panel to ***end*** the "while loop", ***write*** the data to a file, and ***stop*** program execution.

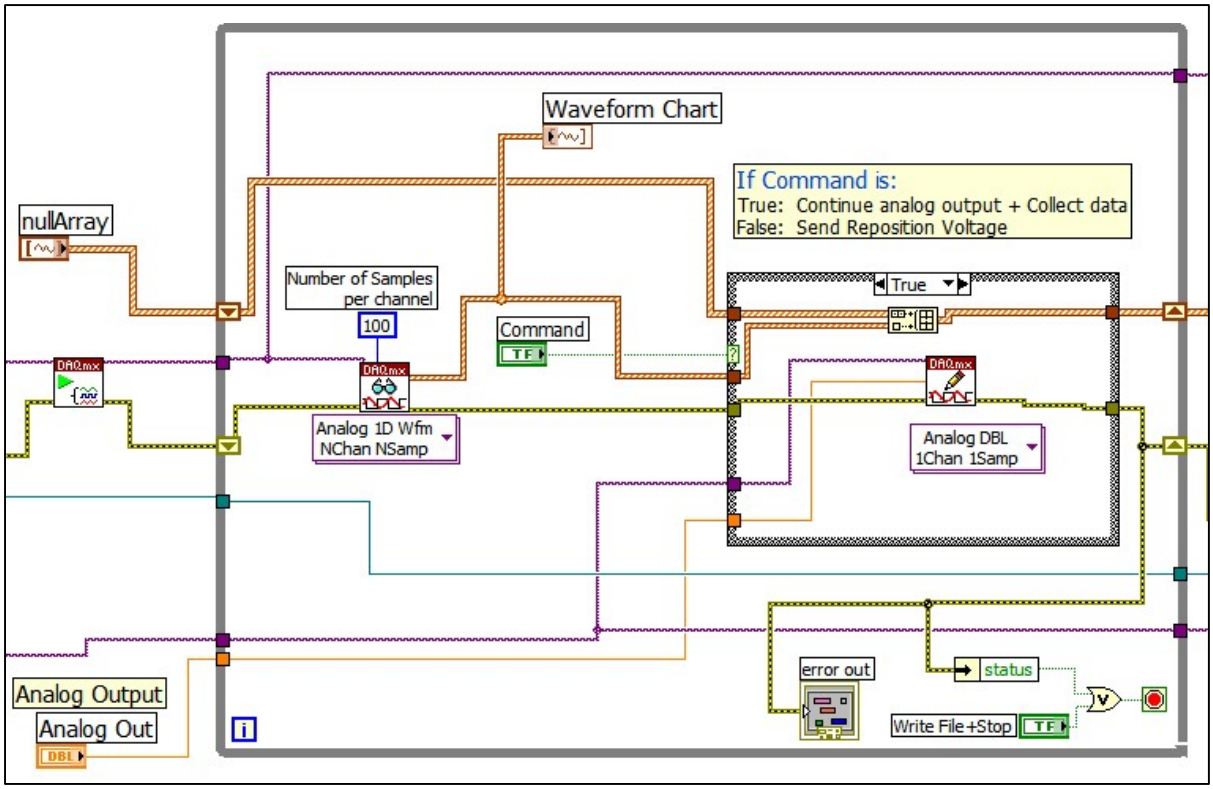

Fig. 5. The Data Acquisition Segment with the Command Button ON

<u>Storing the Data</u>

- After the user clicks the ***Write File + Stop*** button, the ***"while loop" stops***, the ***input*** and ***output tasks*** are ***stopped*** and ***cleared***, the ***data*** is ***written*** to a spreadsheet file, and the ***file*** is ***closed***. Before converting the data to a spreadsheet format, the time values are modified so the ***first value*** in that data is ***zero***. This segment of the block diagram is shown in Fig. 6.

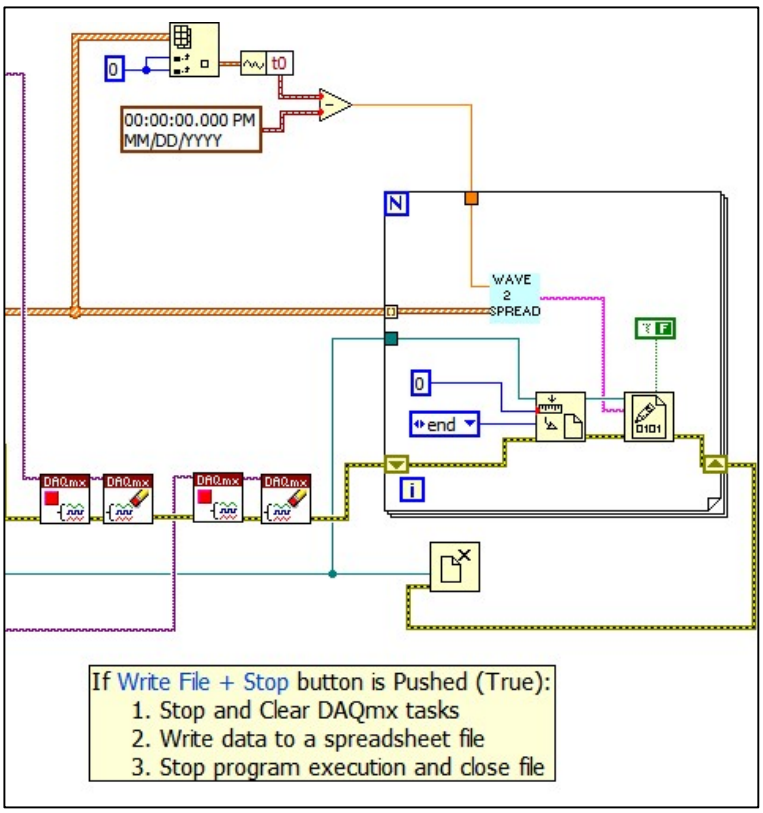

Fig. 6. Segment to Record Data and Stop Execution