

**The University of Jordan  
School of Engineering  
Electrical Engineering Department**

**EE 449  
Instrumentation and Control Lab**

**EXPERIMENT 7  
DATA ACQUISITION (DAQ)  
& LABVIEW**

Prepared by: Prof. Mohammed Hawa

# EXPERIMENT 7

## DATA ACQUISITION (DAQ) AND LABVIEW

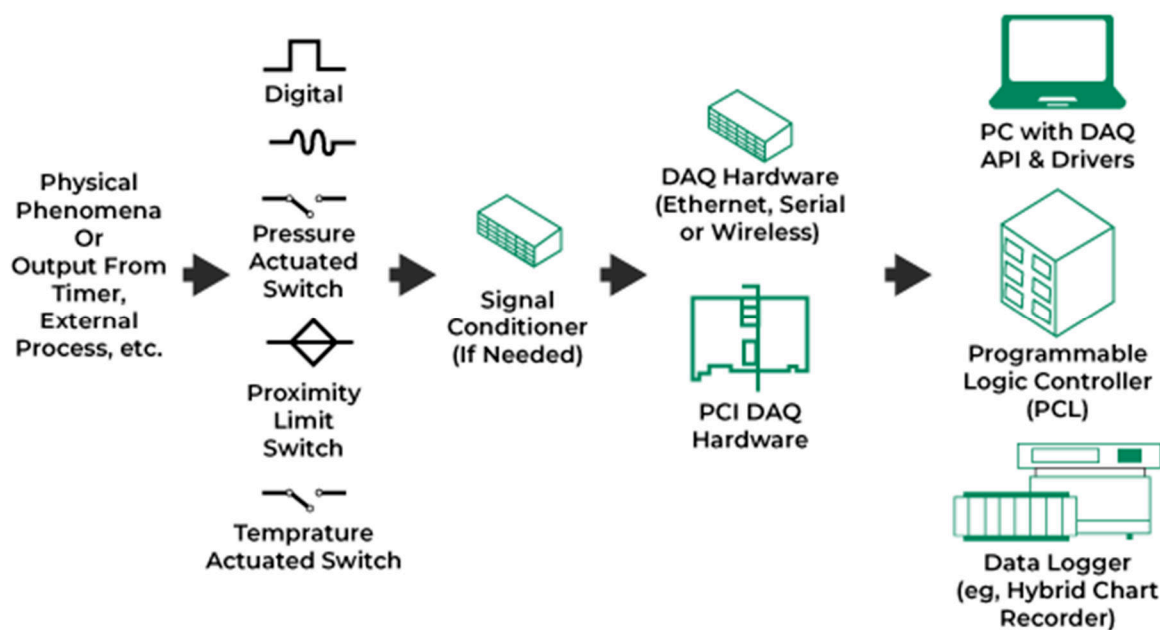
### OBJECTIVES

1. To understand the concept and benefits of Data Acquisition (DAQ) systems.
2. To become familiar with LabVIEW software and the concept of virtual instruments.

### DISCUSSION

Data acquisition (DAQ) systems are very popular in industrial and research settings. They are systems that convert analog waveforms (captured by sensors) into digital values for processing on a computer or a controller. The components of a data acquisition system include:

- Sensors, to convert physical phenomena into electrical signals.
- Signal conditioning circuitry. This can include isolation, amplification, filtering, etc.
- Analog-to-digital converters, to convert the conditioned sensor signals into digital values.



DAQ hardware is what usually interfaces between the conditioned signal and the PC. It typically contains analog-to-digital converters (ADC) and digital-to-analog converters (DAC). It could be in the form of a module that connects to a computer's port (USB, serial, Ethernet, etc) or a card connected to slots on the PC motherboard (ISA, PCI, PCI-E, etc.) Sometimes adapters are needed, in which case an external breakout box can be used.

National Instruments is an international company that builds DAQ hardware, and it also provides a popular software for this hardware. This software is called LabVIEW (Laboratory Virtual Instrument Engineering Workbench). LabVIEW is a graphical programming language that uses icons instead of lines of text to create software programs to read the acquired signals, process them, and use them for controlling industrial systems.

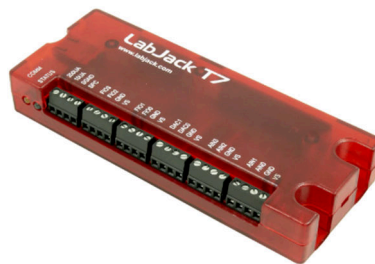
The DAQ hardware can be used to measure real-world analog signals (just like an oscilloscope) and can also be used to generate analog signals (just like a function generator). In between signal measurement and signal generation, LabVIEW is able to provide monitoring, automating and controlling functionality to build very complex control systems.

You write a software program in LabVIEW by designing a *virtual instrument* (VI). In this experiment, you will build simple VIs in order to get acquainted with the capabilities of LabVIEW and typical DAQ systems.

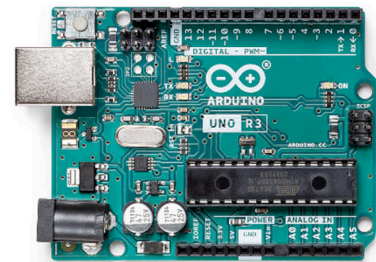
For future work you can download the latest version of LabVIEW from National Instruments' website and use an affordable DAQ module (such as an NI USB-6009 or LabJack T7) or even a simple Arduino UNO to build your own automation system. In a factory or a lab setting, the same concepts apply, but you will probably need a much more capable (and hence much more expensive) DAQ hardware.



NI USB-6009 Data Acquisition Module



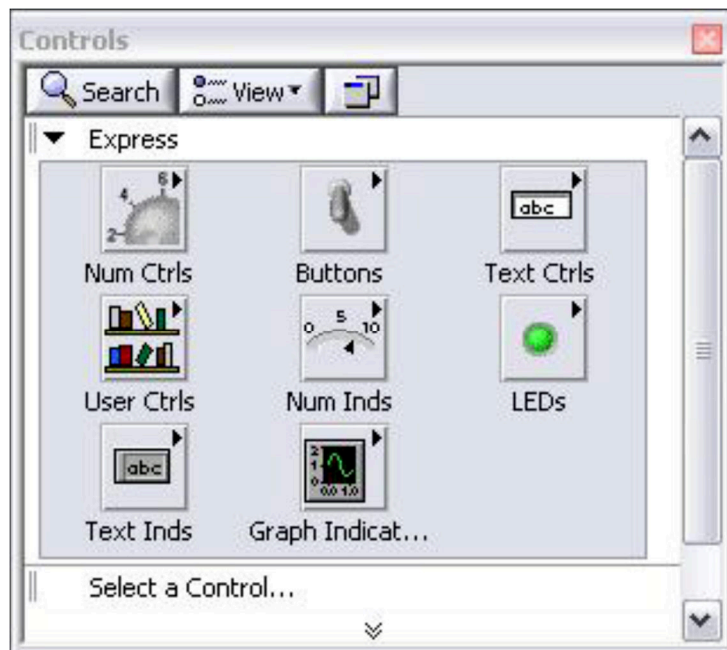
LabJack T7 Data Acquisition Module



Arduino UNO

## PROCEDURE A: GENERATE AND DISPLAY A SIGNAL

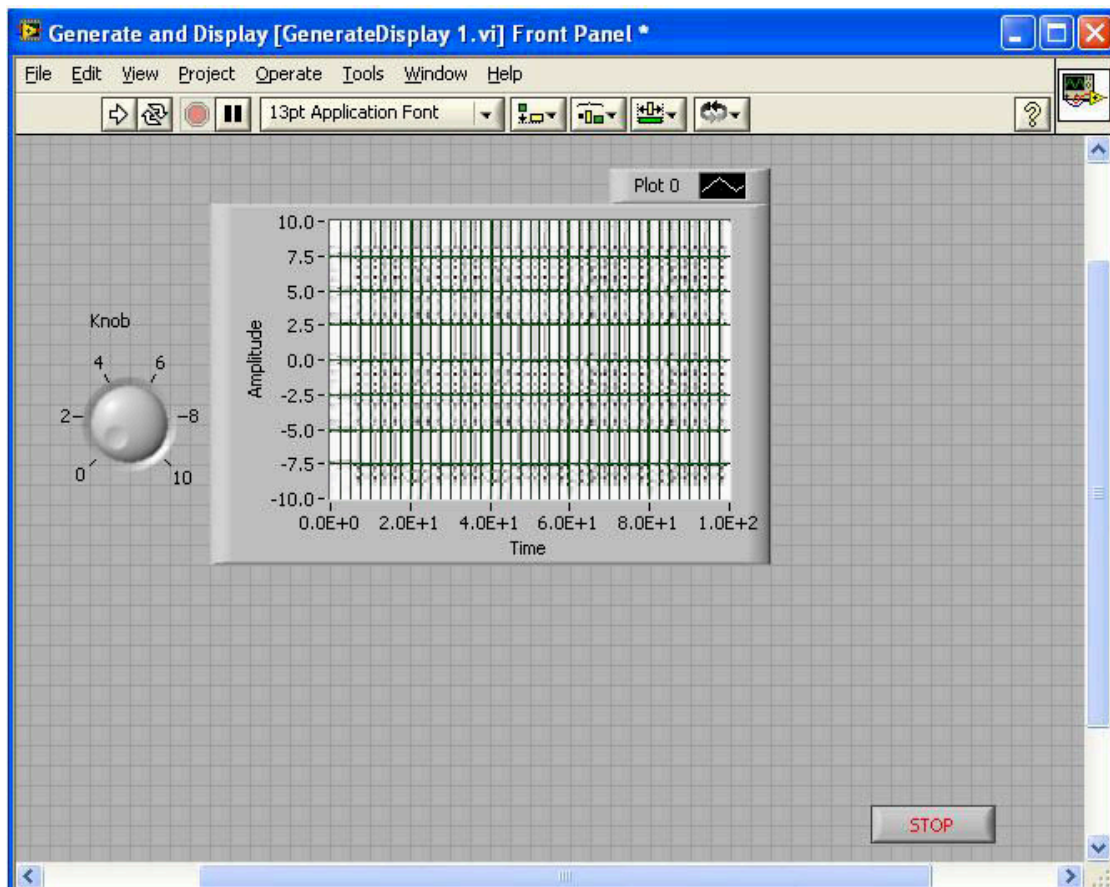
1. Start the LabVIEW application software.
2. In the Getting Started window, select **Files » New** to display the **New** dialog box. From the **Create New** list, select **VI » From Template » Tutorial (Getting Started) » Generate and Display**, then press **OK**.
3. Two windows will be opened; they are called **Front Panel** and **Block Diagram**.
4. To view the Controls palette for the **Front Panel**, right click anywhere in the **Front Panel**, which displays the **Controls Palette**. Alternatively, you can use select **View » Controls Palette**.
5. Select the **Express** tab on the **Controls Palette** to display the **Express** sub palette (see figure below).



6. From the **Express** sub palette, click on the **Num Ctrls** icon to display the **Numeric Controls** palette.

7. Click the **knob control** and then drag and drop it on the Front Panel. This knob will be used later on in an exercise to control the amplitude of a signal.

8. Click the **graph indicator** and then drag and drop it on the Front Panel (say to the right of the **knob control**). You should get something similar to what is shown below.



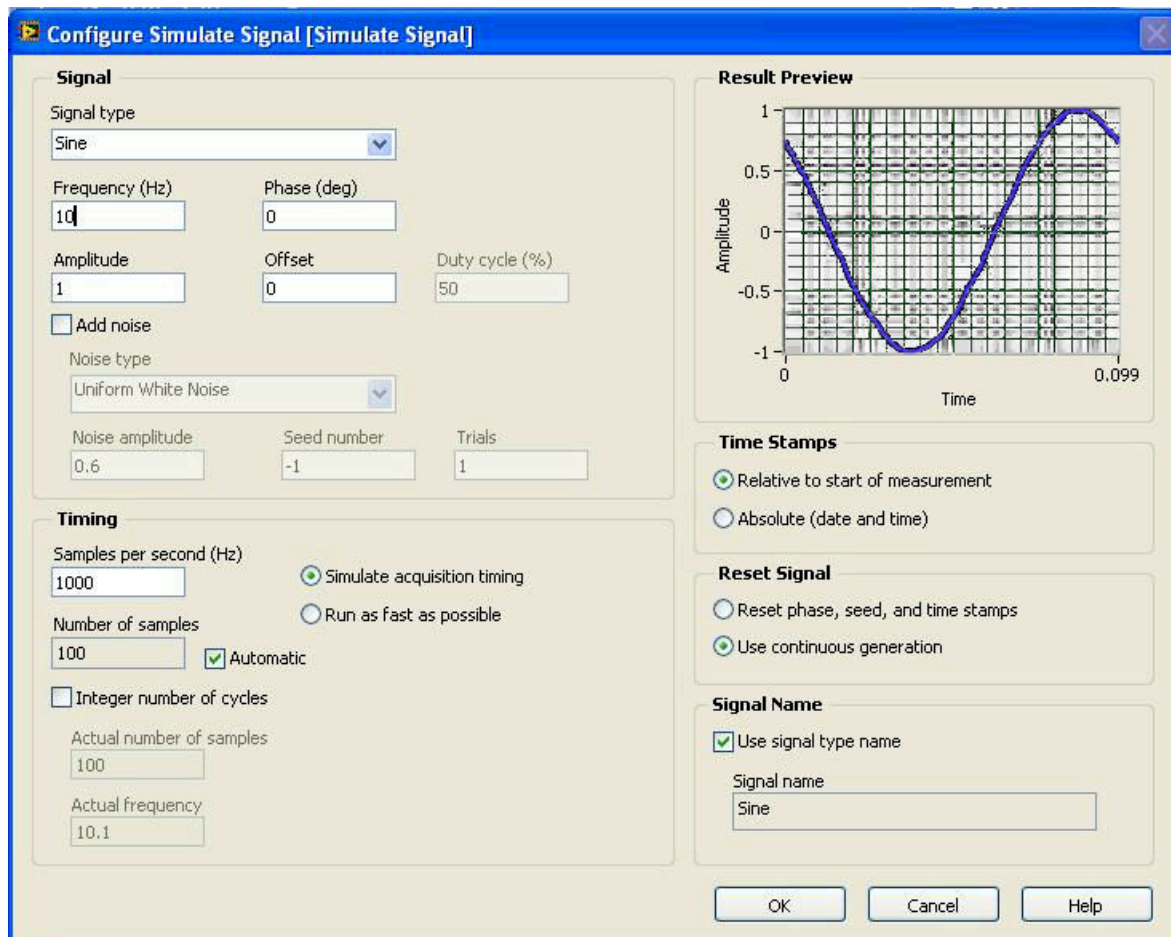
9. Select **File » Save As** and save the VI as “part1.vi”.

10. In the **Block Diagram**, right click anywhere in the window. The **Functions** palette shows up. At the bottom, select the **Express** tab, then select **Signal Analysis » Simulate Signal**, and drag and drop it into the **Block Diagram**.

11. Move the cursor over the down arrows at the bottom of the **Simulate Signal Express VI**. When a double-headed arrow appears, as shown below, click and drag the border of the Express VI to add two rows. Release the border, the Amplitude input appears. Double click the Simulate Signal Express VI to configure it.

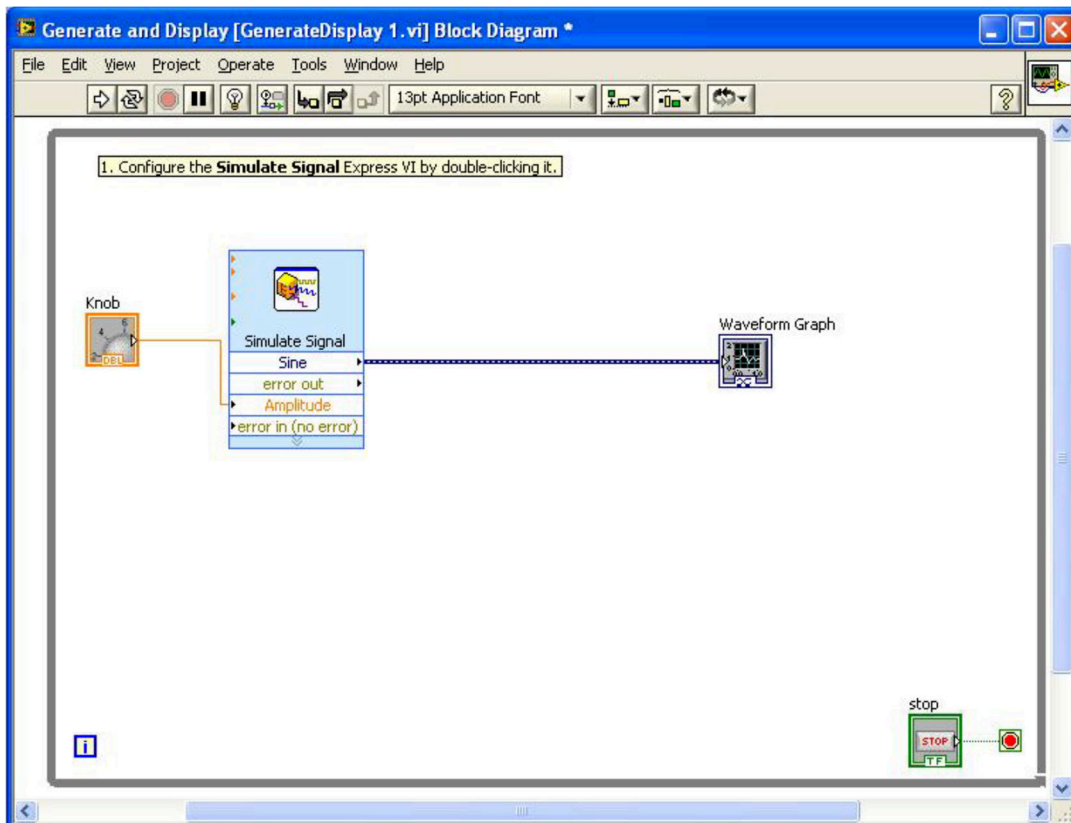
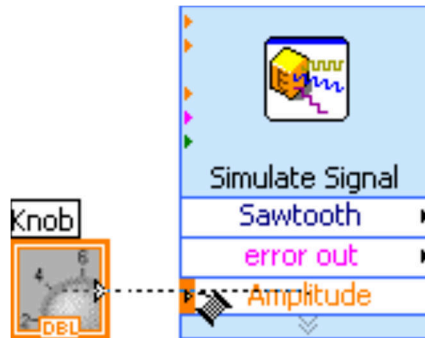


12. In the **Configure Simulate Signal** window, from **Signal type** select **Sine**, and set the **Frequency** to 10Hz, as shown below.



13. Make sure to put the knob to the left of the **Simulate Signal** Express VI and inside the gray loop, the terminals inside the loop are representations of **Front Panel**.

14. Move the cursor over the arrow on the **Knob** terminal. The cursor becomes a wire spool, as shown below. Click the arrow on the **Knob** terminal and then click the arrow on the **Amplitude** input of the **Simulate Signal** Express VI, to connect the two objects together, as shown below.



15. Select **File** » **Save** to save the VI.

16. Display the **Front Panel** to run the VI.

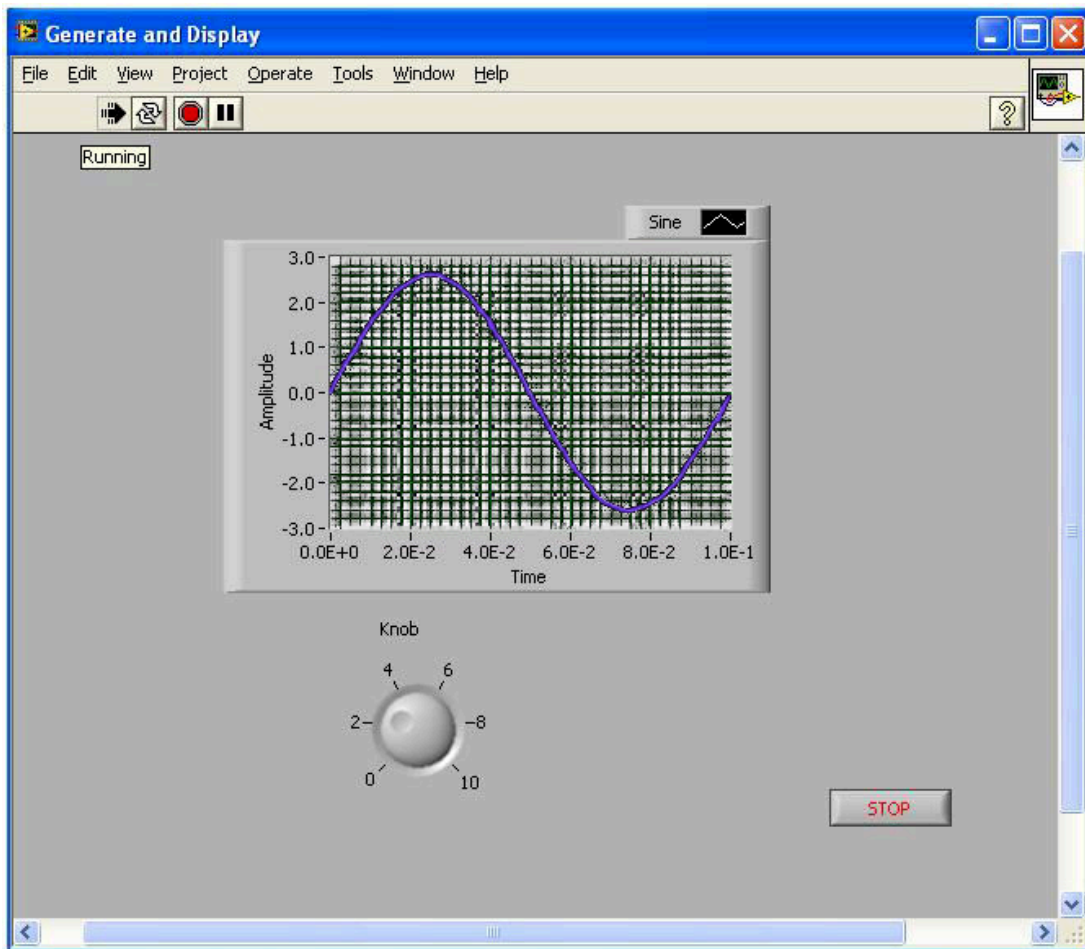
17. Run the VI by selecting **Operate** » **Run**, or click the **RUN** button.



18. Move the cursor over the knob. The cursor becomes a hand. Turn the knob to adjust the amplitude of the sine wave.



19. Notice the change of the amplitude of the sinewave which appears on the signal plotter in the **Front Panel**, as shown below.



20. Stop the VI by selecting **Operate » Stop**, or click the **STOP** button.



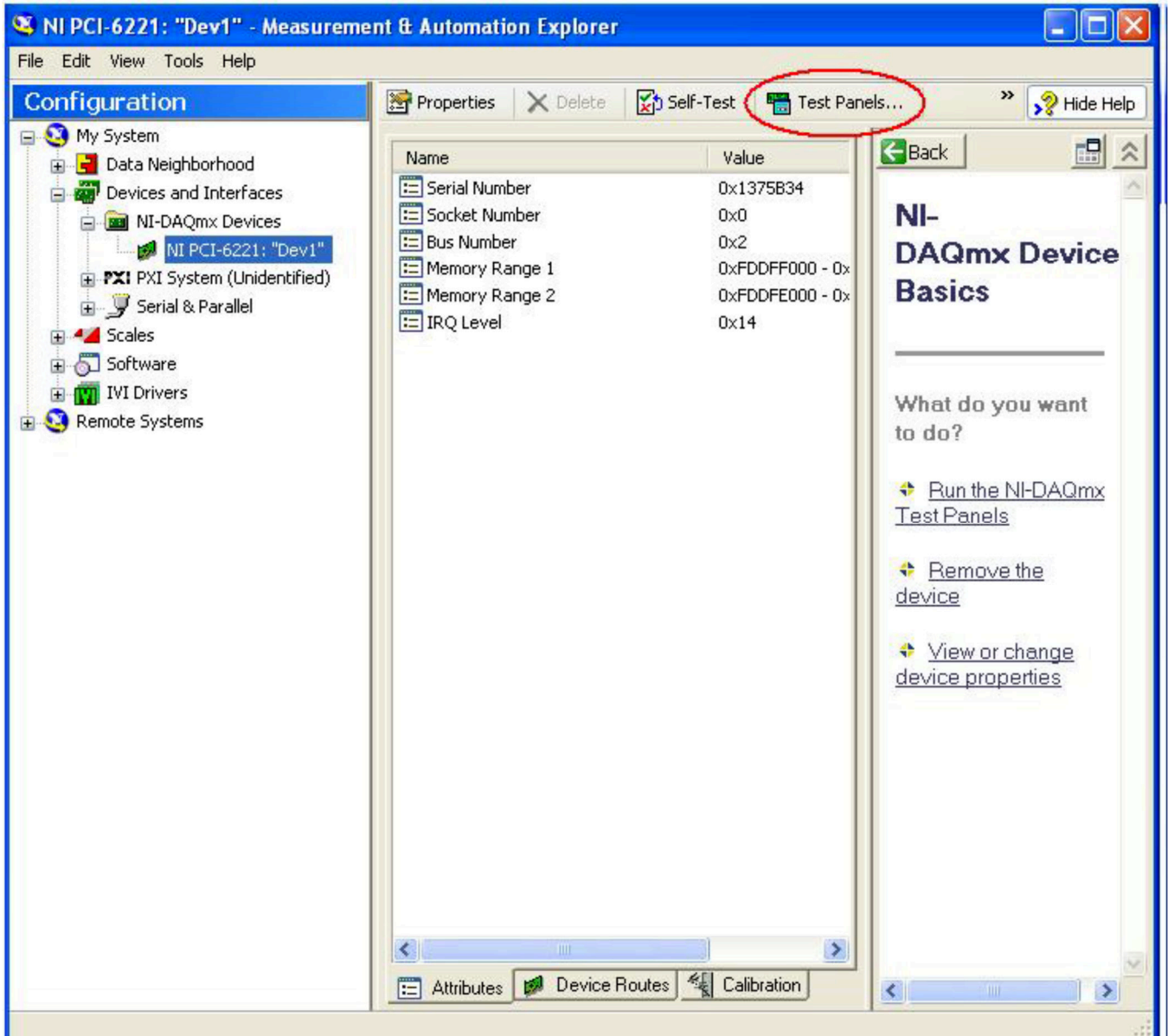
21. Make sure to check the VI program by the Engineering staff. Print the screen and save it with the name set to your group and section number.

22. Make sure to show the results to the Lab supervisor.

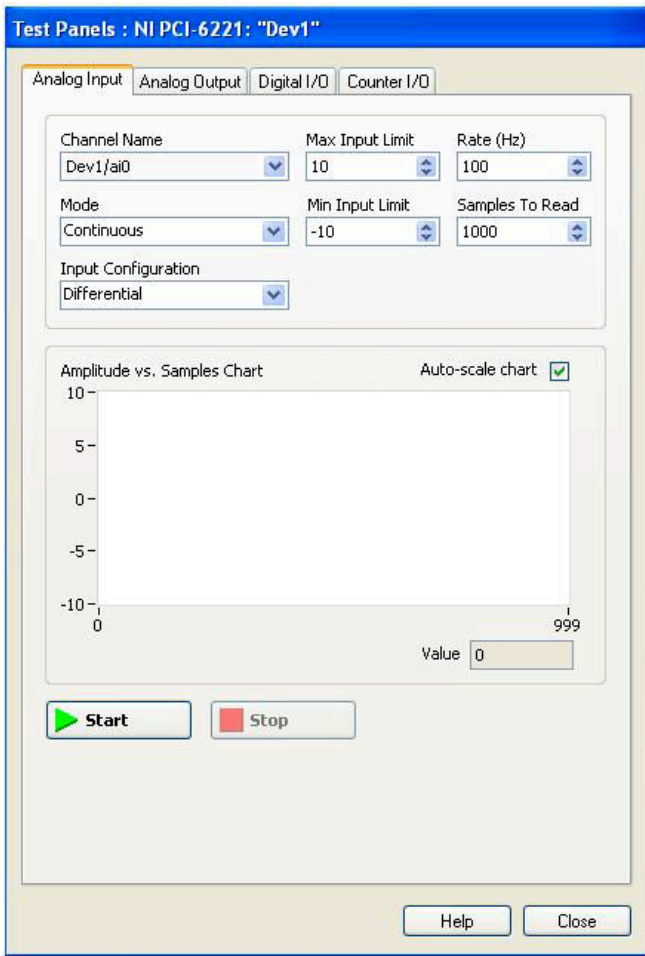
23. Close all windows.

## PROCEDURE B: MEASUREMENT & AUTOMATION EXPLORER (MAX)

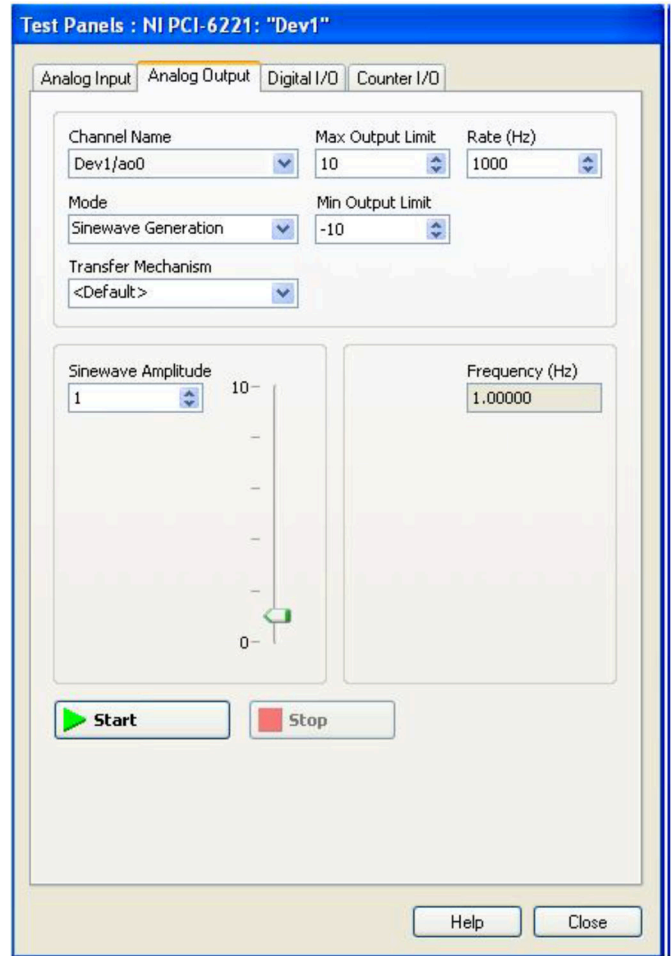
1. Start the Measurement & Automation Explorer.
2. On the **Configuration** window, select **Devices and Interfaces » NI-DAQmx Devices » NI PCI-6221:"Dev1"**, as shown below.



3. Select **Test Panel** (see above). The **Test Panels** window will open as shown below.
4. Now, test the panel by generating a sinewave analog signal in the analog output of the DAQ, then get it back to the analogue input of the DAQ to see it on the chart.
5. To generate a signal, click on the Analogue Output tab, from MODE select Sinewave Generation, and then set the **Rate** to 1000Hz, and the Sinewave **Amplitude** to 1.



Analog Output



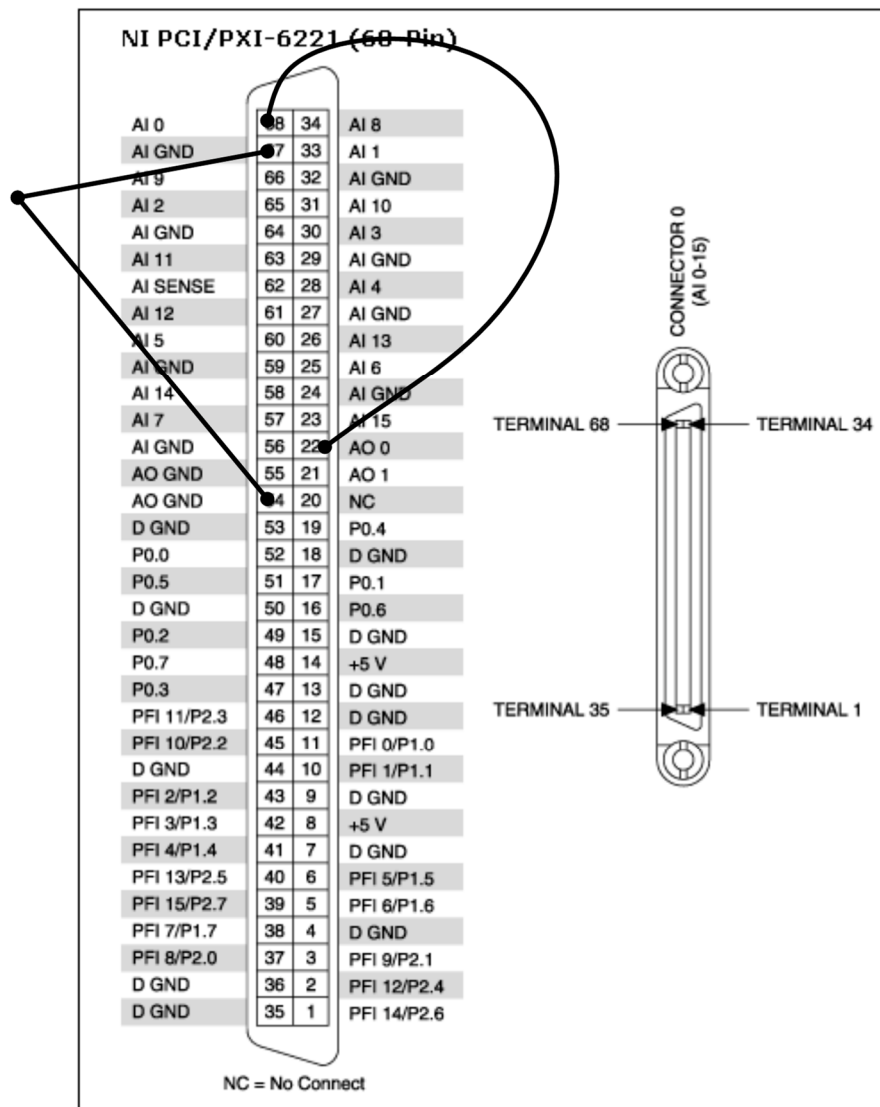
Analog Input

6. To receive the signal, click on the Analog Input Tab, then from **MODE** select **Continuous**. From **Input Configuration** select **Differential**, then set the **Rate** to 100Hz, and then set the **Samples To Read** to 1000.

7. Follow the pin connection on the DAQ as shown below, where the Differential signal from the analog output **AO 0** is sent to the Differential signal at the analog input **AI 0**. Make sure to let the Lab supervisor check the connection.

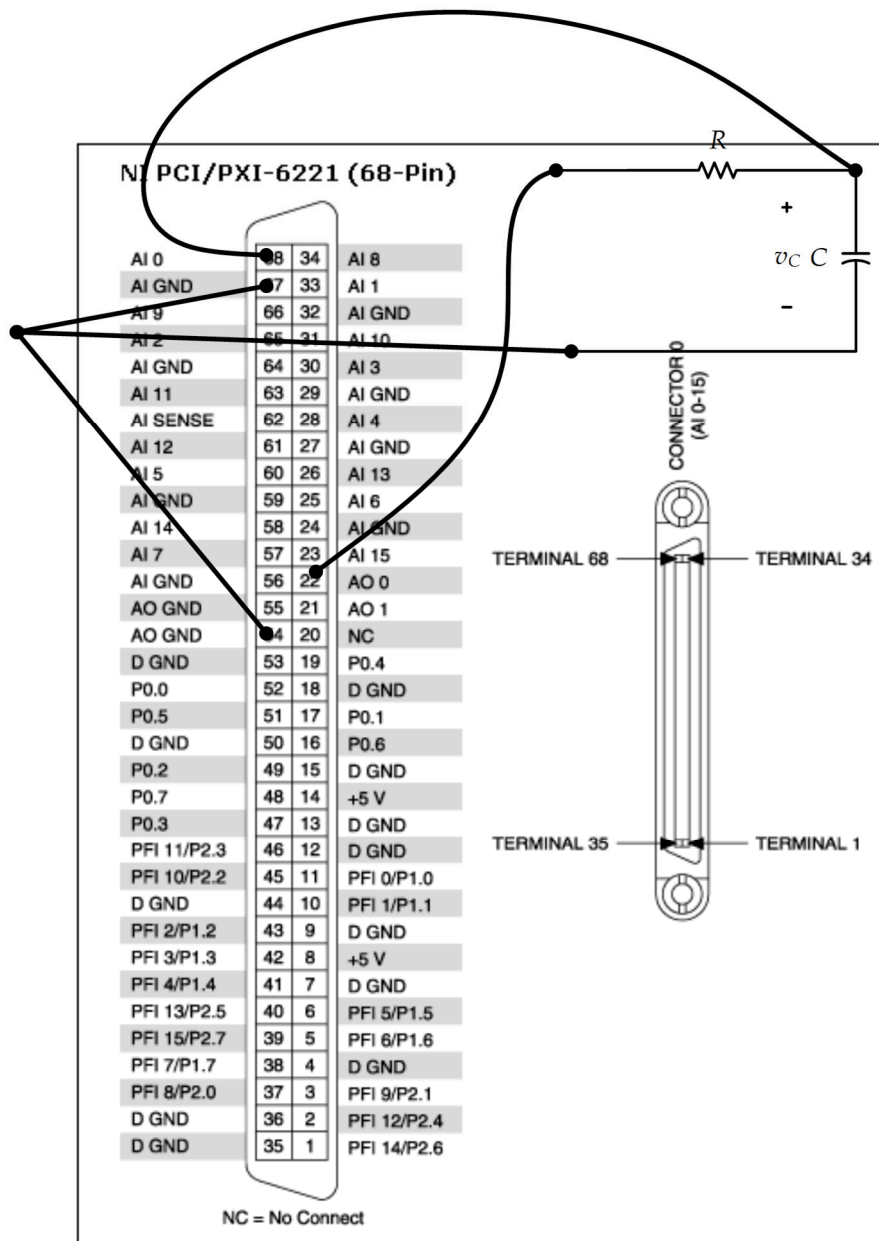
8. Click **START** to start the Analog Output, and then Click **START** to turn the Analog Input on to get the result.

9. Click **STOP** in both the Analogue Input and Analog Output.



## PROCEDURE C – USING DAQMX TO MEASURE A SIGNAL FROM AN RC CIRCUIT

1. In this procedure, you want to generate a square waveform from LabVIEW, send it to an external RC circuit (which is acting as a LPF), then read the output voltage, and display that output as a waveform using LabVIEW.
2. Connect a 0.1  $\mu\text{F}$  capacitor and a 10 k $\Omega$  resistor and wire them on a breadboard, then connect them to the DAQ hardware as shown below. Note that the output from the DAQ is the input to the RC circuit, and the output from the RC circuit is the Input to the DAQ hardware.

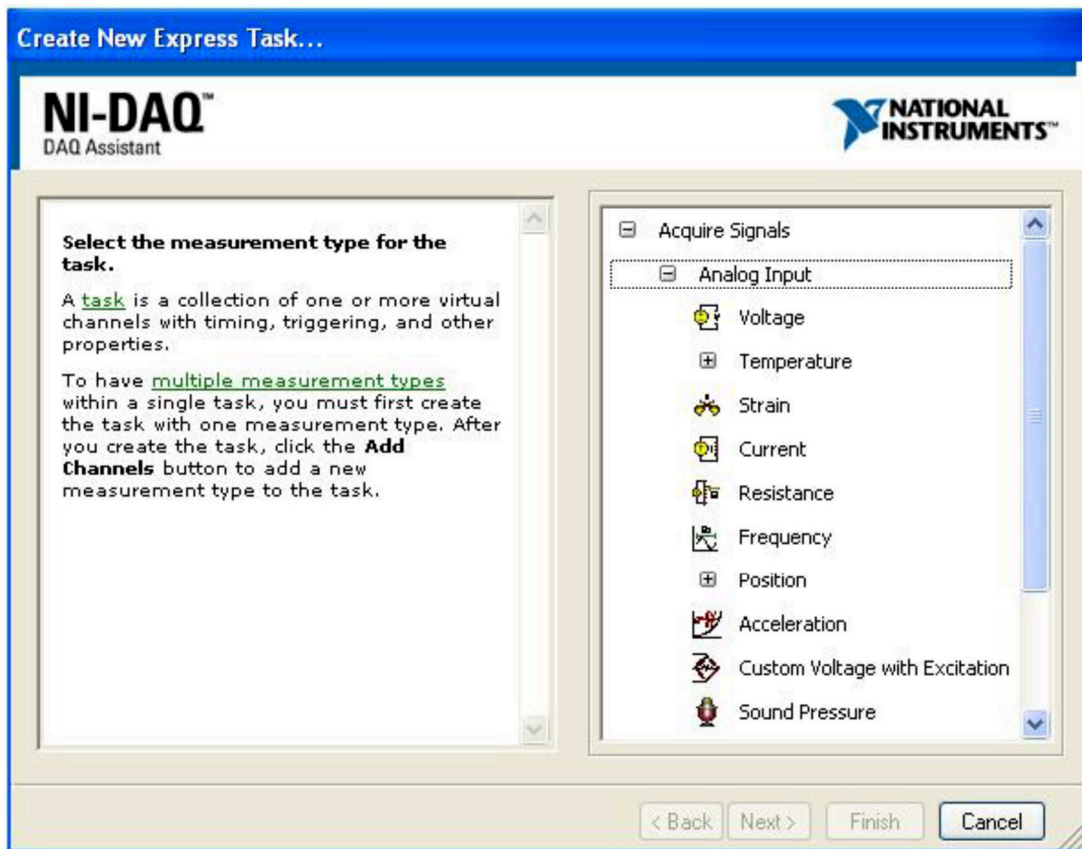


2. Launch LabVIEW again, and select Blank VI.

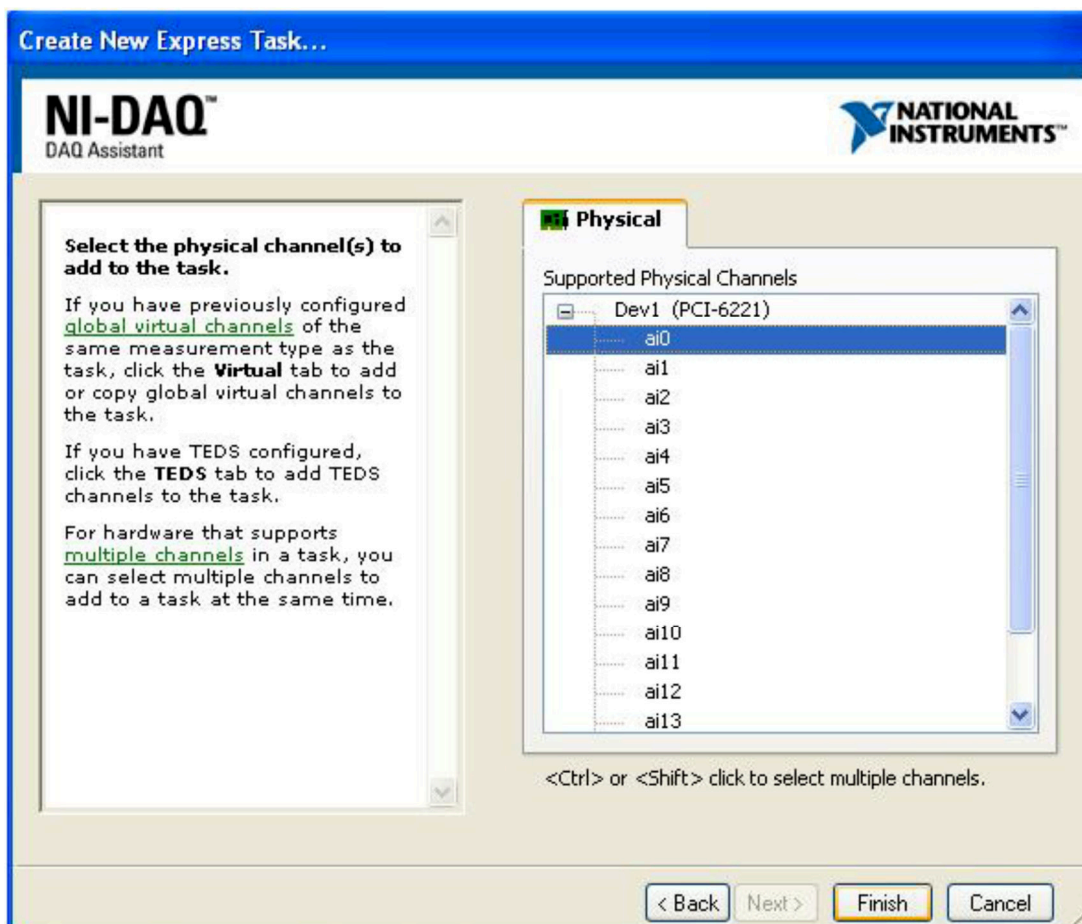
3. In the Block Diagram, inside the Function palette select Measurement I/O » NI-DAQmx » DAQ Assistant, and drag and drop it on the Block Diagram.



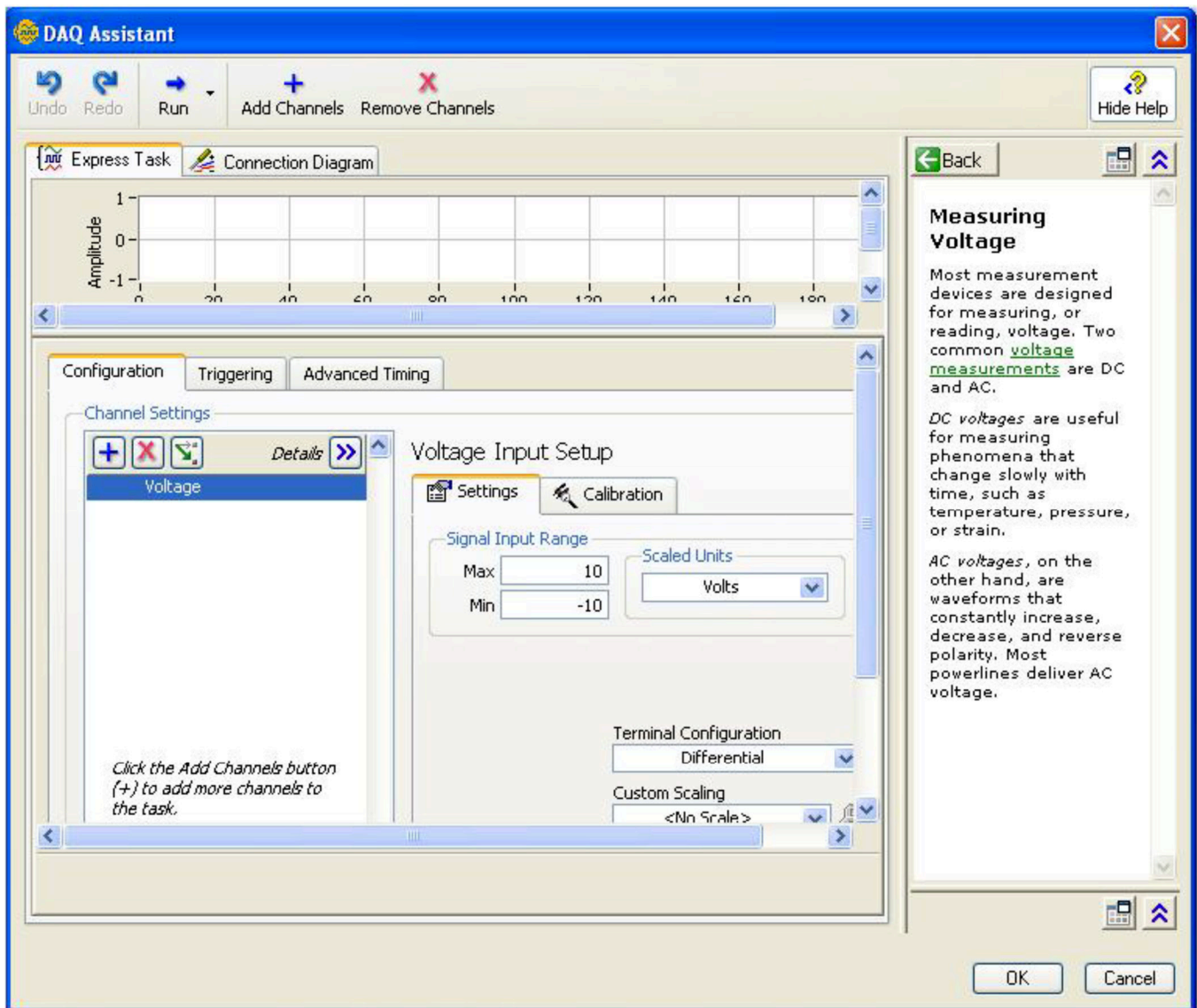
4. The DAQ Assistant will start initializing; when it finishes, configure it to read the output voltage of the RC-circuit. Select Acquire signals » Analog Input » Voltage as the measurement type for this channel, as shown below.



5. Select ai0 » Finish from the NI-DAQmx simulated device, as shown below.



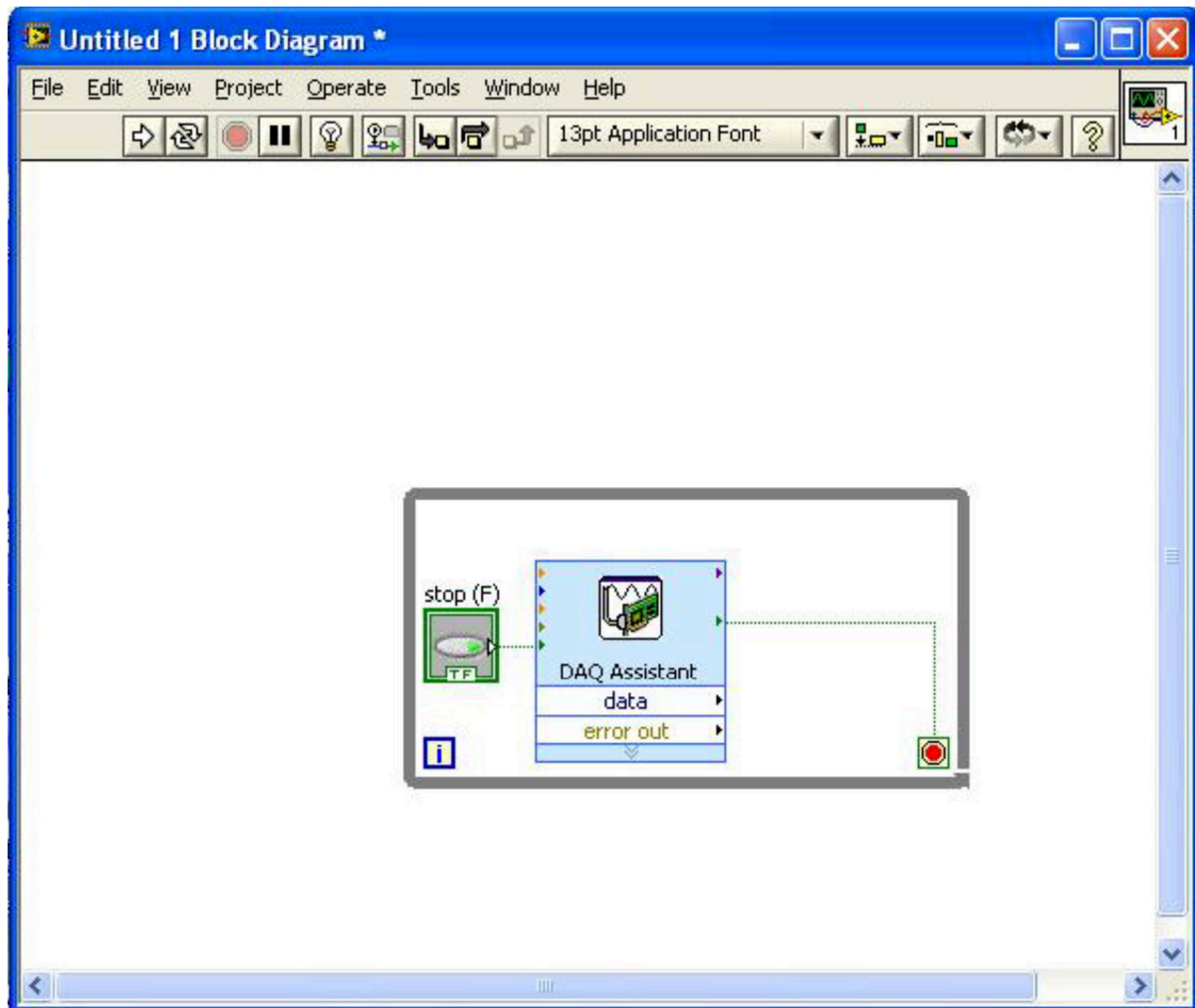
6. A DAQ Assistant window opens to add timing and terminal configuration to the channel as well as add custom scaling, and so on, see figure below.



7. Select Continuous Sample in the Acquisition Mode and Differential in Terminal Configuration, and click OK. Go to Terminal configuration and select RSE mode.

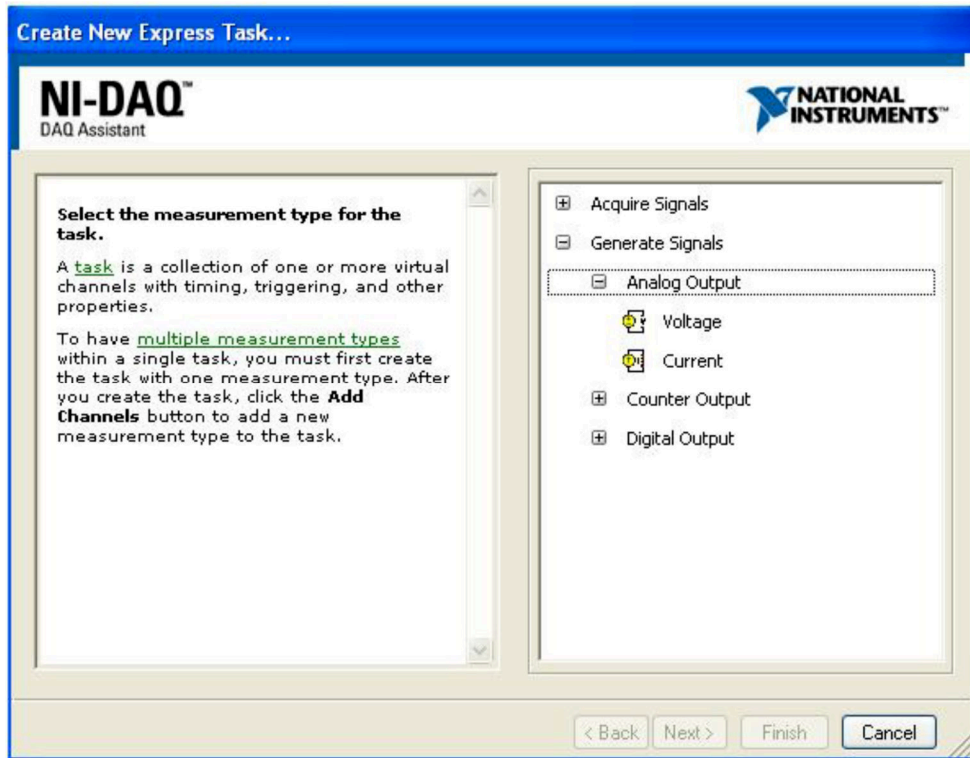
8. A confirms Auto Loop Creation window will ask to place the DAQ Assistant Express VI in a loop due to continuous generation of the signal, so create a loop by clicking Yes button.

9. The Block Diagram will be as shown below.



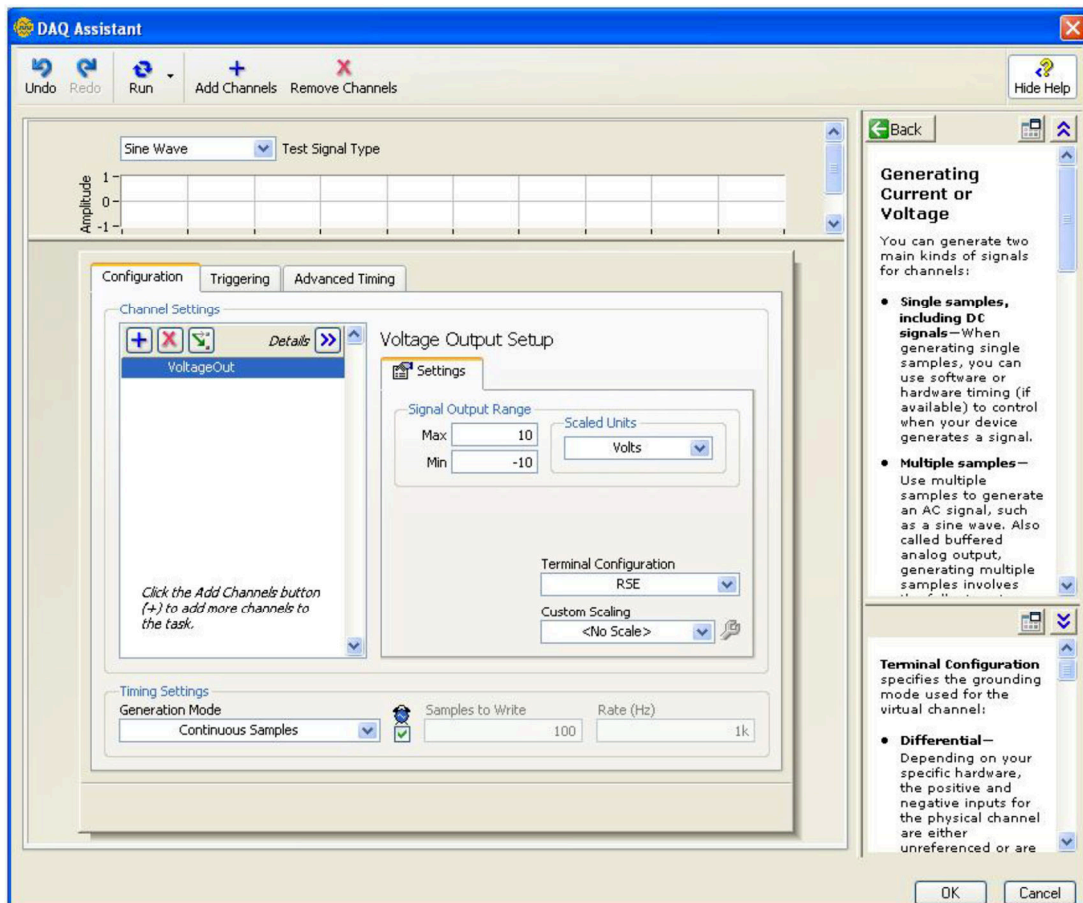
10. Go back to the Block Diagram again, select View » Function Palette, then in the Function Palette select Measurement I/O » NI-DAQmx » DAQ Assistant, and drag and drop it inside the gray loop on the Block Diagram.

11. DAQ Assistant will start initializing; when it finishes, configure it to generate signal to be used as input to the RC circuit. Select Generate signals » Analog Output » Voltage as the measurement type for this channel, as shown below.



12. Select ao0 » Finish from the NI-DAQmx simulated device.

13. A DAQ Assistant window opens to add timing and terminal configuration to the channel as well as add custom scaling, and so on, see the figure below.

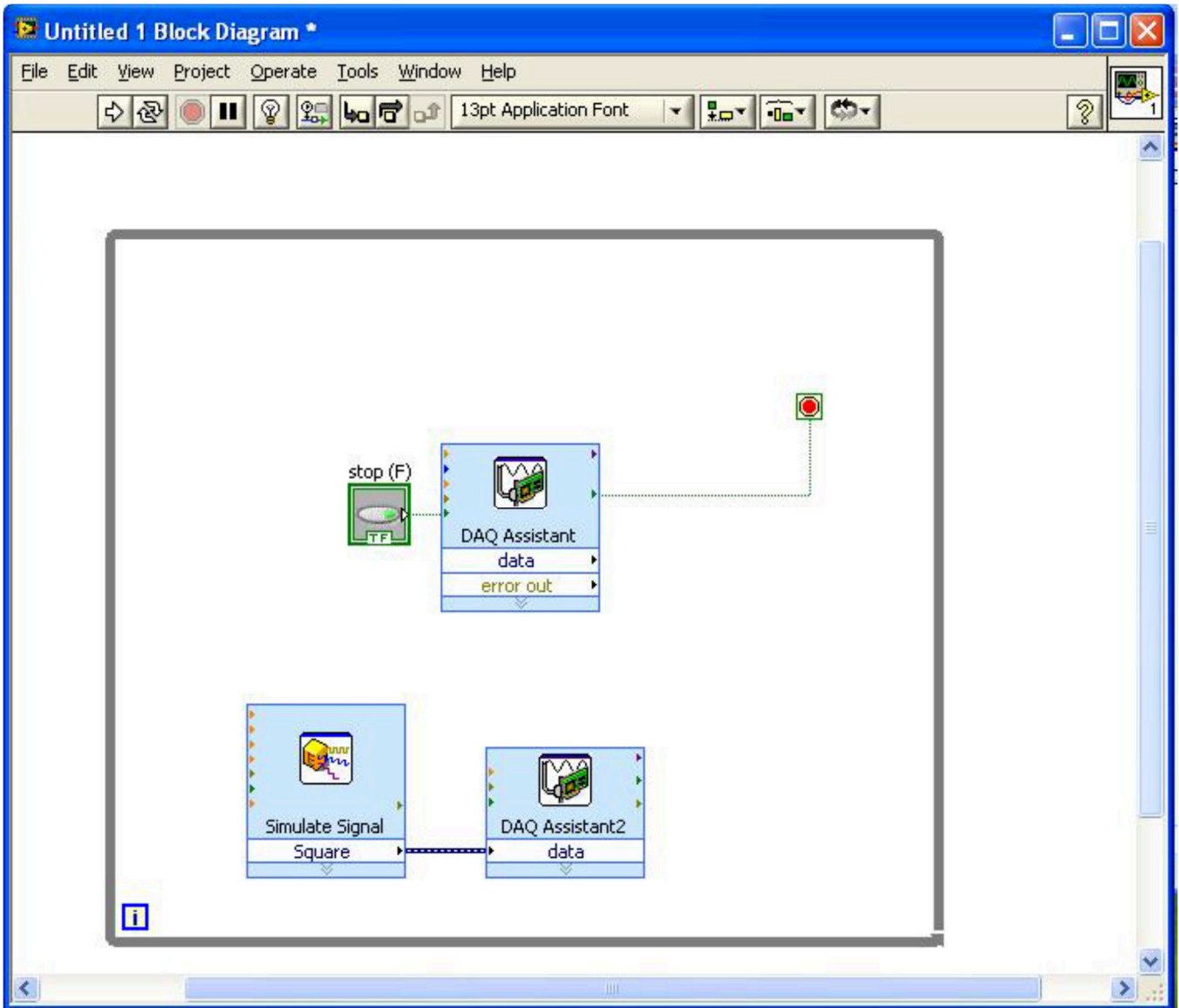


14. Select Continuous Sample in the Generation Mode and RSE in Terminal Configuration, and click OK.

15. To simulate the input signal, in the Function palette in the Block Diagram, select Express » Simulate Signal, and drag and drop it to the left of the second DAQ Assistant.

16. Double click the Simulate Signal, and configure the signal but select a square wave, set the frequency to 50Hz.

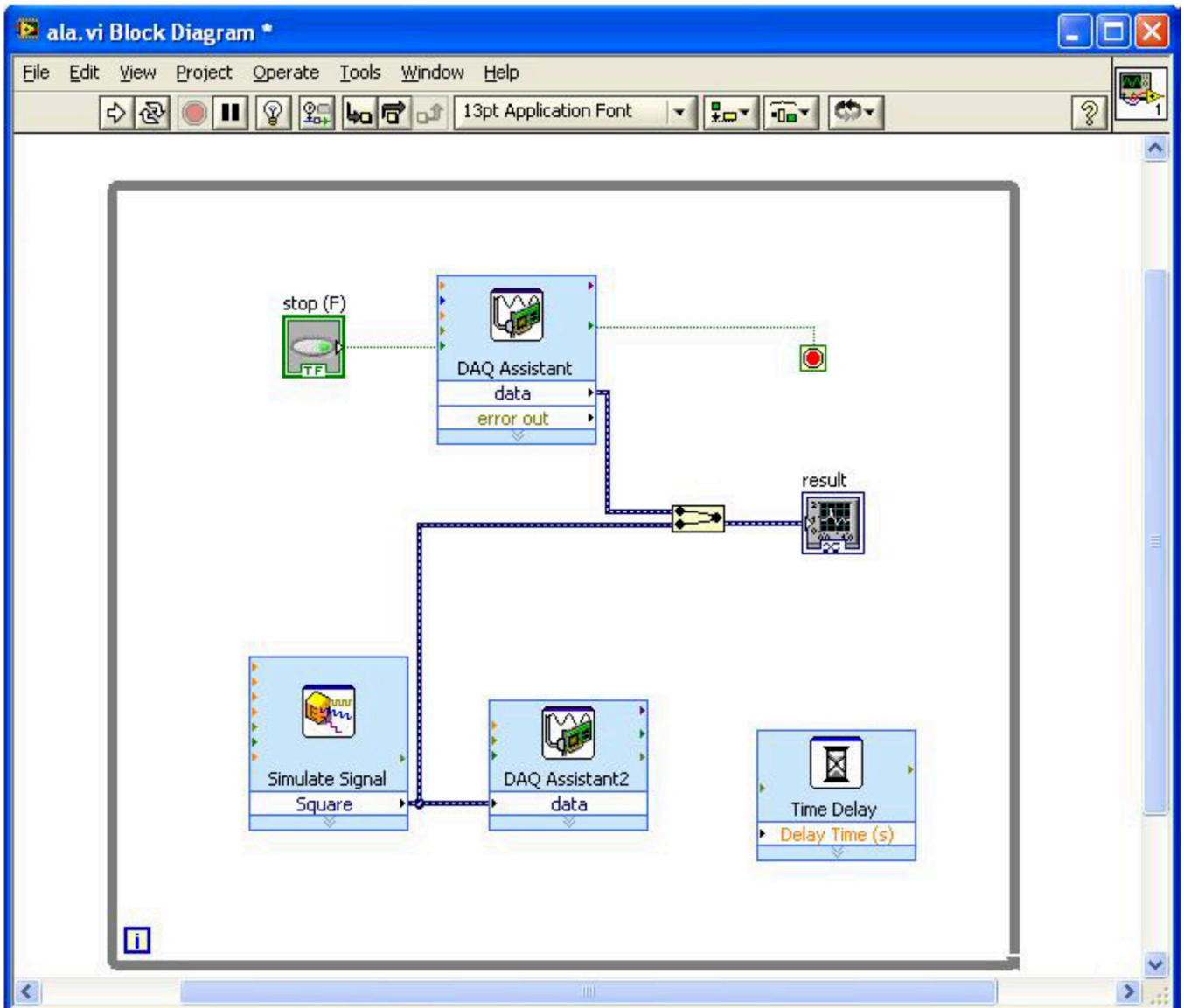
17. Connect the Simulate Signal to the second DAQ Assistant together, as shown below.



18. On the output of the simulate signal, move the cursor over the square arrows, right click then select Create, then select Graph Indicator, to view the output signal.

19. Wire the output of the first DAQ Assistant with the graph indicator to show the input signal at the same graph.

20. The Block Diagram will be as shown below.



21. Save all your work in a file with the name set to your group and section number.

22. Back to the Front Panel to run the program.

23. Make sure to show the results to the Lab supervisor.

**Q.** What does a **for** loop inside a VI look like? Explain the steps to include one on your own, rather than getting one added for you automatically. You can use Google or YouTube to find out how.

.....

.....

.....

**Q.** Explain how you would design a VI to draw a Bode diagram representing the frequency response function of the first-order RC circuit (LPF) you worked with. Just explain the main blocks you are going to use and how the software program should work in general terms. There is no need to implement it on LabVIEW, nor give step-by-step tutorial for building the VI.

.....

.....

.....

.....

**\*\* End \*\***