Introduction to LabVIEW and Computer-Based Measurements

ni.com/daq

NATIONAL INSTRUMENTS™

March 2014 Edition

Eastern Europe

Copyright

© 2014 National Instruments Corporation. All rights reserved.
Under the copyright laws, this publication may not be reproduced or transmitted in any form, electronic or mechanical, including photocopying, recording, storing in an information retrieval system, or translating, in whole or in part, without the prior written consent of National Instruments Corporation.

National Instruments respects the intellectual property of others, and we ask our users to do the same. NI software is protected by copyright and other intellectual property laws. Where NI software may be used to reproduce software or other materials belonging to others, you may use NI software only to reproduce materials that you may reproduce in accordance with the terms of any applicable license or other legal restriction.

Trademarks

National Instruments, NI, ni.com, and LabVIEW are trademarks of National Instruments Corporation. Refer to the *Terms of Use* section on ni.com/legal for more information about National Instruments trademarks.

Product and company names mentioned herein are trademarks or trade names of their respective companies.

Members of the National Instruments Alliance Partner Program are business entities independent from National Instruments and have no agency partnership, or joint-venture relationship with National Instruments.

Patents

For patents covering National Instruments products/technology, refer to the appropriate location: **Help»Patents** in your software, the patents.txt file on your media, or the *National Instruments Patent Notice* at ni.com/patents.

Contents

Exercise 1 - Hardware Configuration

Goals

- Learn how to detect and configure hardware
- Carry out a test measurement to confirm that the setup is connected correctly

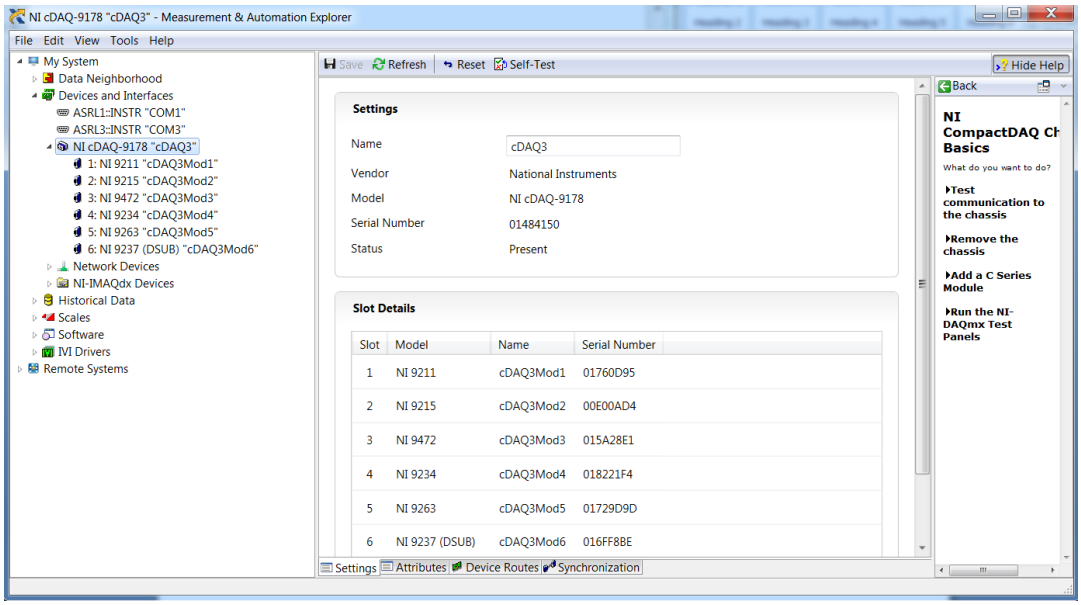
Software and hardware configuration

Estimated time: 20 minutes

Similar to the Windows Device Manager, which manages all peripherals connected to a Windows PC, MAX (Measurement and Automation Explorer) manages all NI hardware and software. This application is installed with most NI software packages. In this exercise, you will look at the most used features of MAX including **Software** and **Devices and Interfaces**.

1. Once your hardware is connected and powered on, open MAX by double clicking the icon on the desktop or navigating to **Start » All Programs » National Instruments » Measurement & Automation Explorer**.

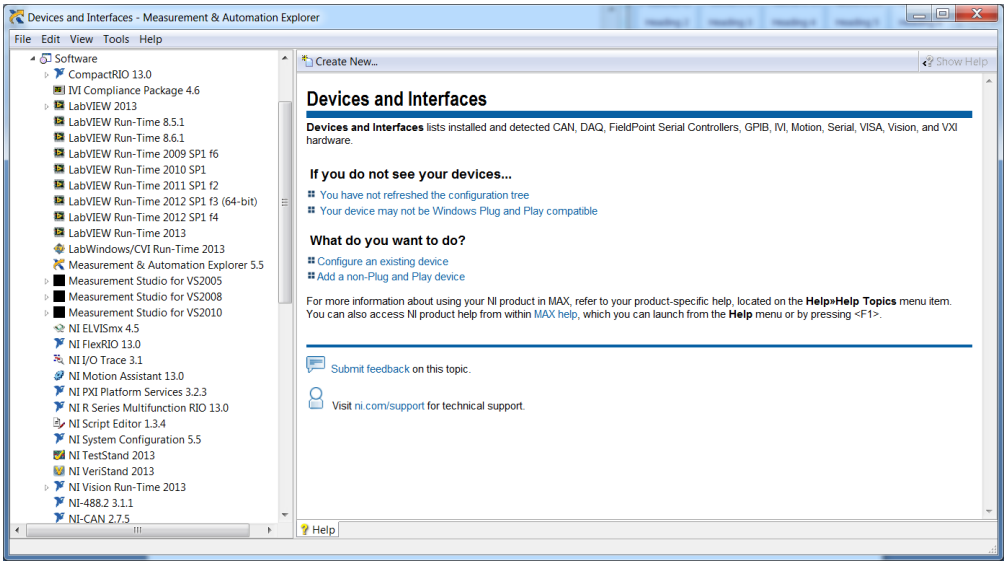
2. After MAX launches, expand **My System » Devices and Interfaces**. You should be able to see any hardware attached to your local machine:



   Within MAX, you can ensure that your hardware is properly connected, check external connections, rename your devices, and even simulate devices for developing code without attached hardware.

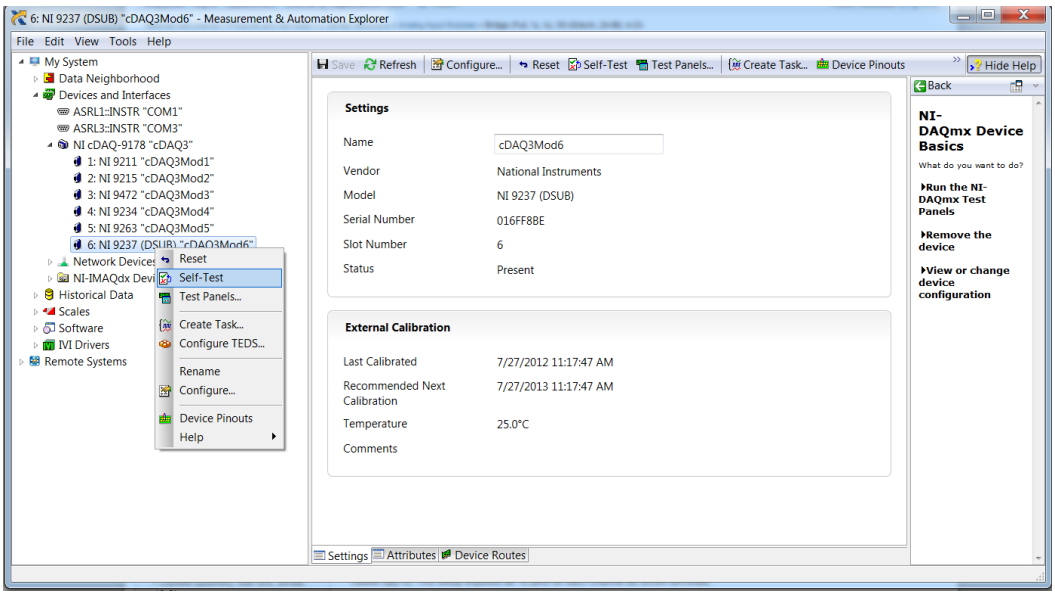
3. In addition to hardware, you can see all the software installed on the

machine by expanding **My System » Software**. Here you can check the driver and software versions:

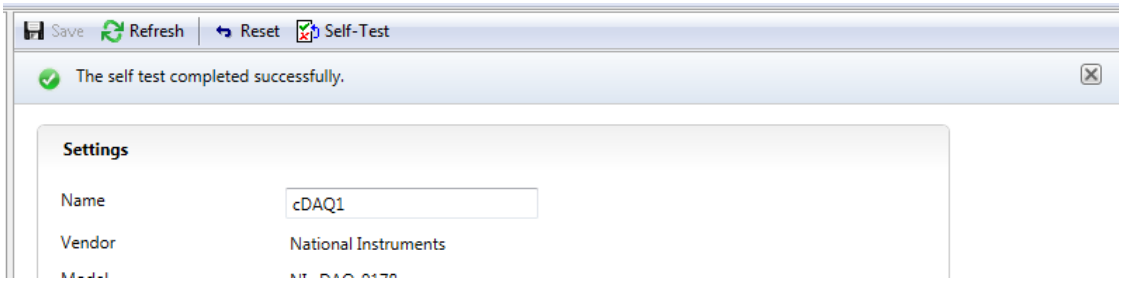


Underneath the software section, you should notice that both LabVIEW and NI-DAQmx are installed – NI-DAQmx is the driver that lets your PC communicate with your hardware and LabVIEW is the graphical programming environment that we will use to program during today's exercises.

4. Now that you have navigated through and understand the basics of MAX, you can test the connectivity of the NI CompactDAQ chassis. **Right-click** the chassis (cDAQ-917x) under the Devices and Interfaces section and select **Self-Test**.



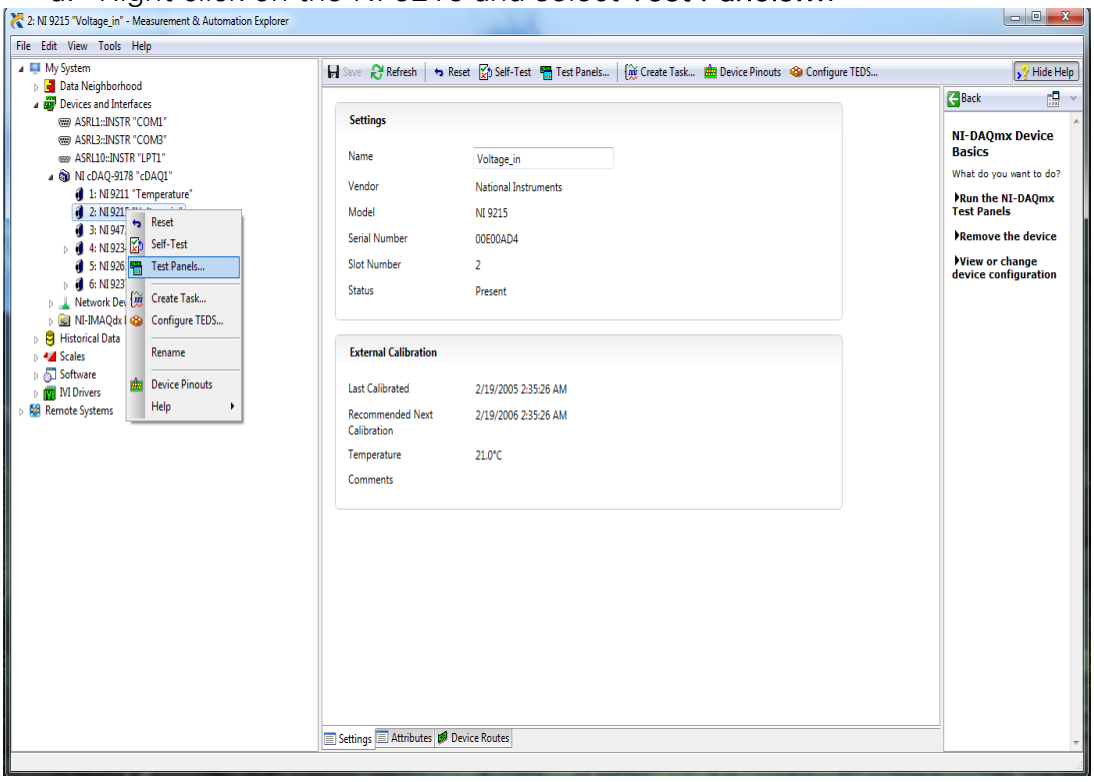
If your chassis is properly connected and able to communicate with your PC, then you will see the following text in the top of MAX window:



Self-testing the chassis is always a good first step after installing new hardware or software.

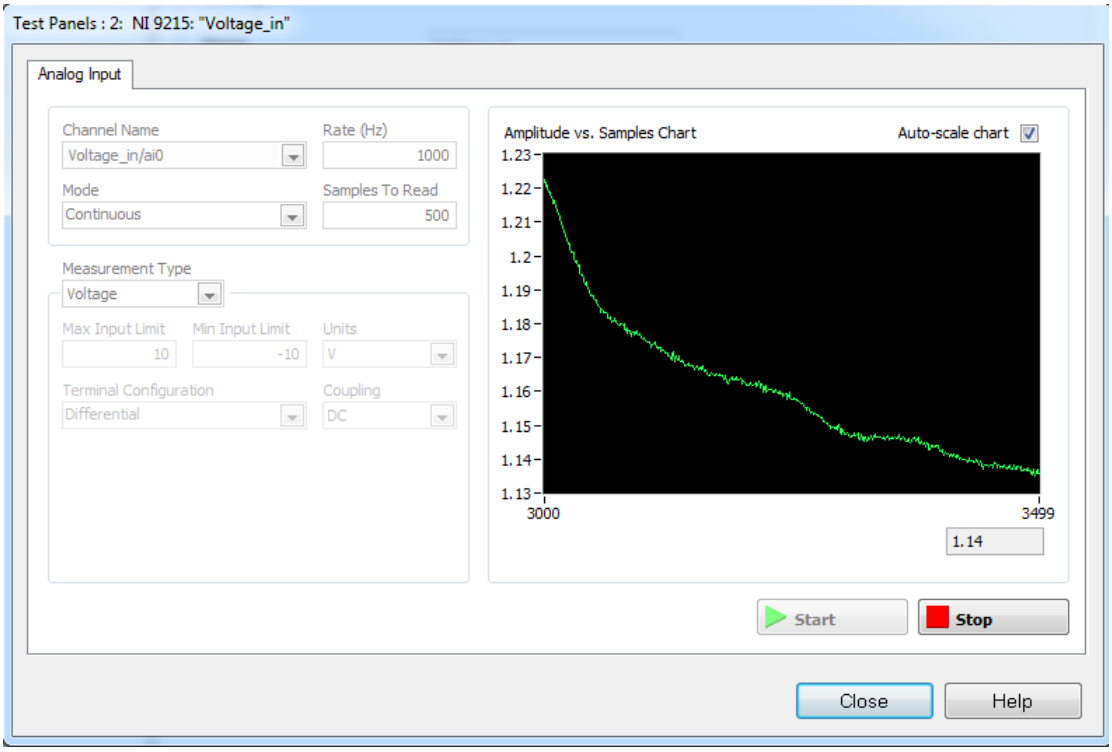
5. To check the signal connections to the NI CompactDAQ chassis, you will use the NI-DAQmx Test Panels. As an example, we will check the signal connectivity of the NI 925 analog input module. We will measure the voltage generated by the solar cell connected to the analog input.

    a. Right click on the NI 9215 and select **Test Panels…**:



    b. Selecting Test Panels brings up a simple utility to check the signals on each of the channels of the module. We have to specify the mode

of acquisition. In our case we will be acquiring the data continuously at sampling rate 1000 Hz and we will collect 500 points at once and show it on the Test Panel graph. Set the measurement according to the and press **Start**. You should see similar pattern as on the figure below.



c. Cover the solar cell by your hand. You will see that the voltage is dropping..

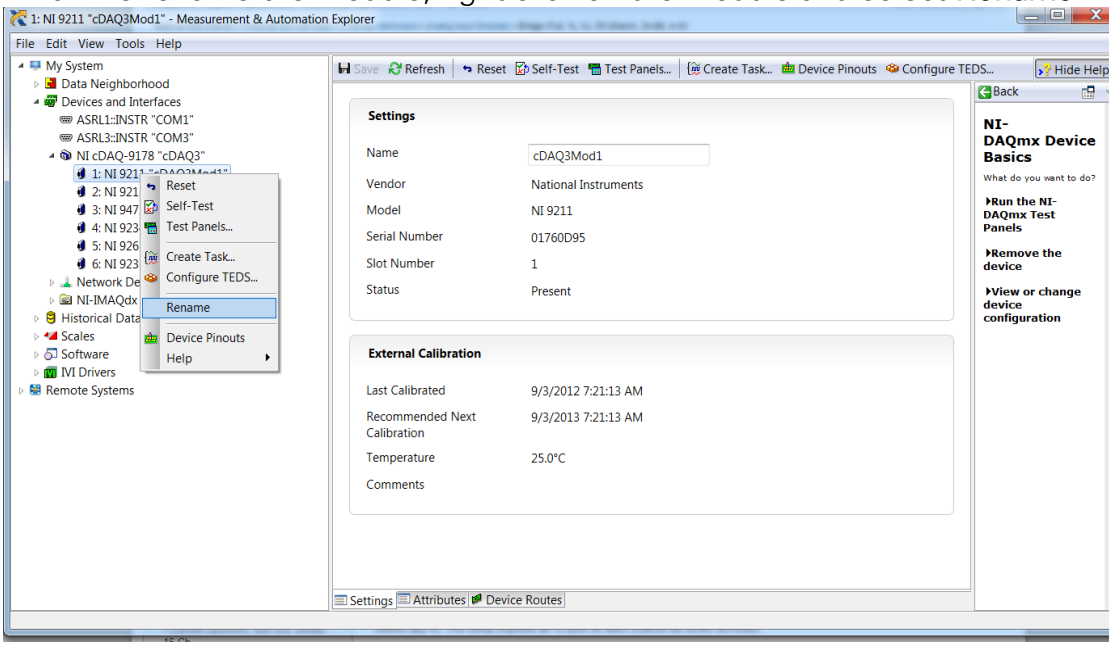
d. Every time you set up a new measurement or measurement system, it is a good idea to confirm that the wiring is correct and that all software is installed and working correctly. MAX provides the insight into your system and setup to help you eliminate mistakes early in your development process.

e. Take a moment to practice with the test panels for the other modules in the chassis. Notice that test panels are slightly different, each providing the inputs for measurements supported by the module.

f. When finished, close any open Test Panel window by selecting **Close**.

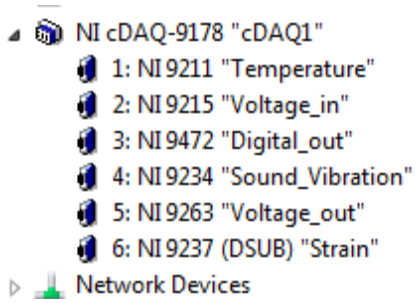
6. For many applications, simply using the default assigned names is not sufficient. Within MAX, you can very easily rename your modules to match the module type or measurement.

    a. To rename the module, right-click on the module and select **Rename**.



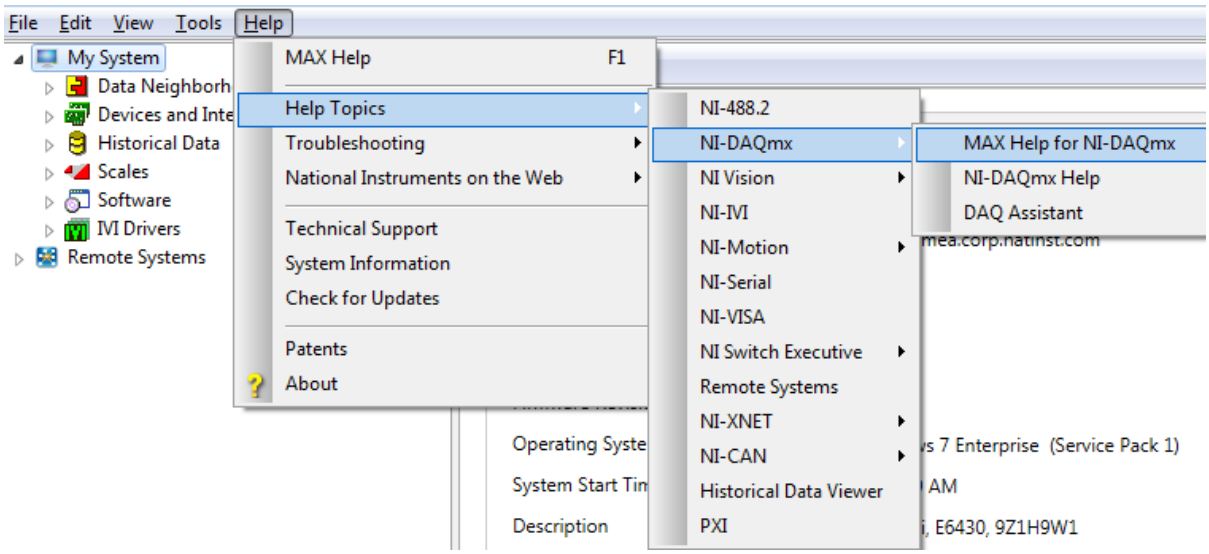
    b. Rename each of the channel types to descriptive names.

    c. Use the following descriptive names to identify your modules



Challenge

Go to **Help » Help Topics » NI-DAQmx » MAX Help for NI-DAQmx** and search for *Simulated Devices* and create your own simulated device. You can test your data acquisition application without real hardware now.



<End of Exercise>

Exercise 2 - Example Finder

Goals

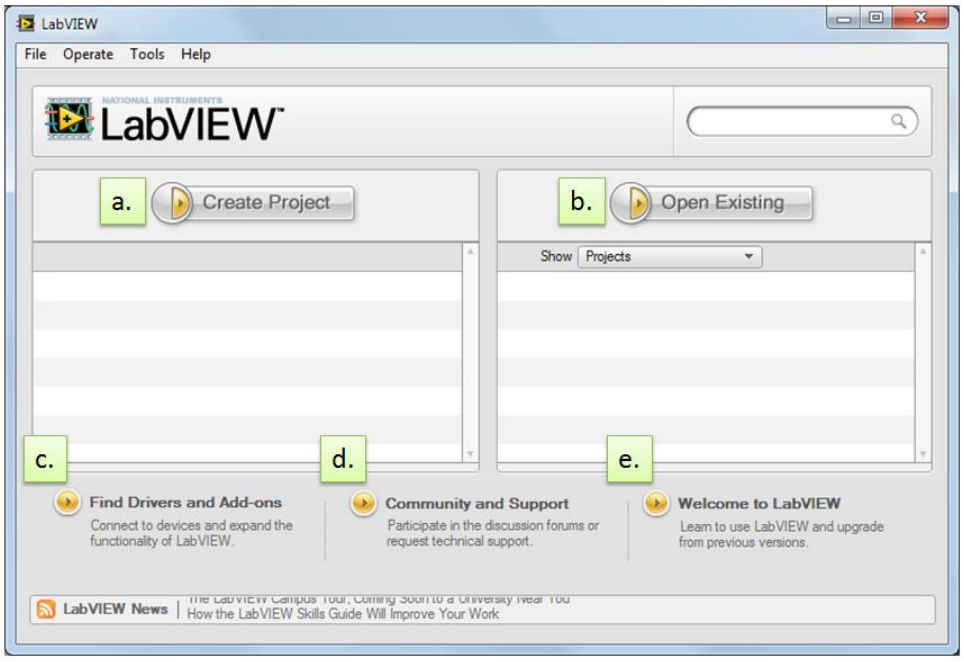
- Become familiar with the NI-DAQmx examples
- Run a pre-built example program to acquire your first measurement

Part A **Open LabVIEW and Explore the Example Finder**

Estimated time: 10 minutes

Installed with NI-DAQmx, the NI-DAQmx examples provide you with a proven starting point for creating data acquisition applications. By utilizing an example program, you can eliminate several sources of errors and save time by utilizing existing code.

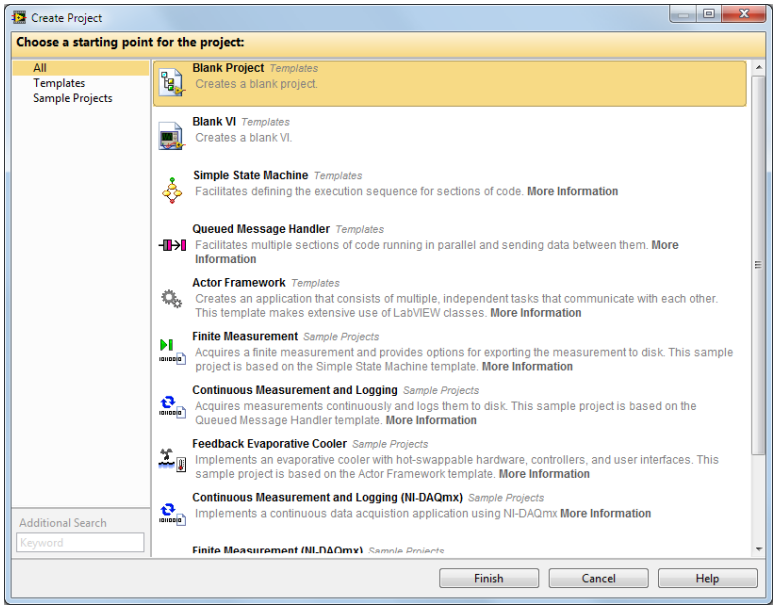
1. To access the examples, open LabVIEW by navigating to **Start » All Programs » National Instruments » LabVIEW 2014** Upon launching, you will see the following Getting Started screen:



2. The opening screen provides paths for developing code, recalling previously opened programs and finding support or help resources.

    a. *Create Project* – Selecting Create Project will launch a screen with options to create new code from existing templates and sample projects. Code architectures like the State Machine and Actor Framework are included in the templates and you can quickly get up and running by creating them from this dialog box.

The sample projects are ready-to-run code built using best practices for scalable development. Using the templates and sample projects, you should never have to start a LabVIEW application from scratch.



b. *Open Existing* – As you work on programs today, you can quickly recall projects that you have worked on within the Open Existing section or press the Open Existing button to browse for your code.

c. *Find Drivers and Add-ons* – For most hardware connected to your PC, you will need to use a hardware driver. Today, you will be using NI CompactDAQ, which utilizes the NI-DAQmx driver. This same driver works for hundreds of data acquisition devices, enabling you to reuse code from device to device and application to application.

d. *Community and Support* – For those times when you get stuck, NI has a vibrant community of LabVIEW developers and data acquisition experts. The forums are a free support resource that is monitored by both NI Applications Engineers and NI enthusiasts to ensure that you will get the support you need. For more immediate support, you can call or email NI Applications Engineers directly as one of the many benefits of the LabVIEW Standard Service Program (SSP).

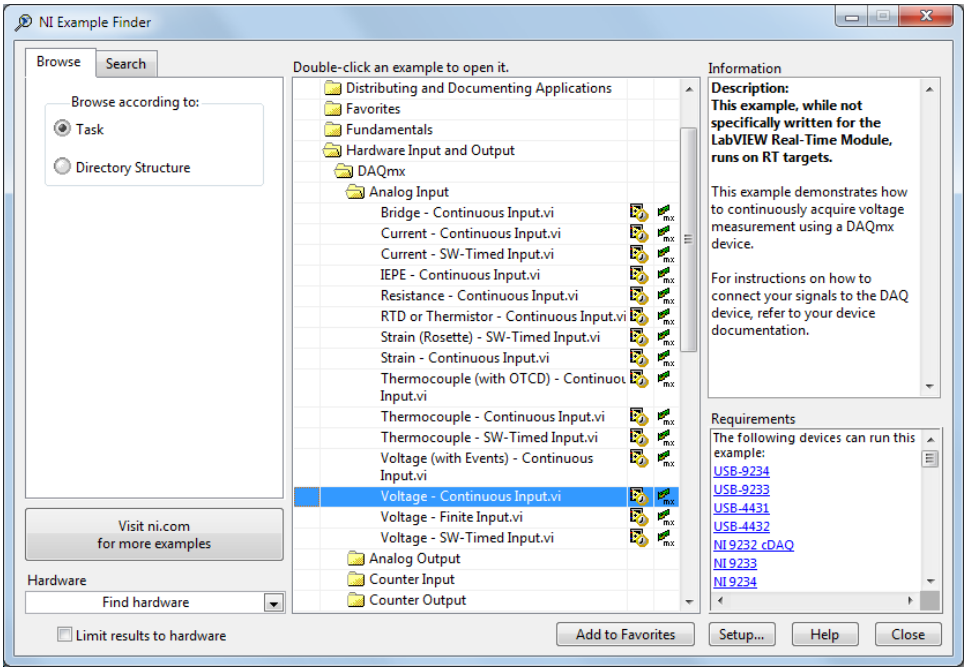
e. *Welcome to LabVIEW* – After today, you will be familiar with using the LabVIEW environment. To move beyond simple

programming tasks and access more in-depth proficiency training, refer to the help and training resources in the Welcome to LabVIEW link.

3. Today, we will be using several examples as starting points for our code. To open the NI-DAQmx examples directory, select **Help » Find Examples** from the LabVIEW Getting Started screen.

4. Once the NI Example Finder launches, navigate to **Hardware Input and Output » DAQmx**.

5. Navigate the DAQmx directory and read some of the capabilities of the examples. Each example has a descriptive name, but you can find out more capabilities of the program by reading the descriptions in the Information section.



Each section of the directory includes examples for performing certain tasks. For example, analog input includes examples for measuring simple voltage, current, strain, and temperature, whereas the analog output section includes examples for controlling either voltage or current.

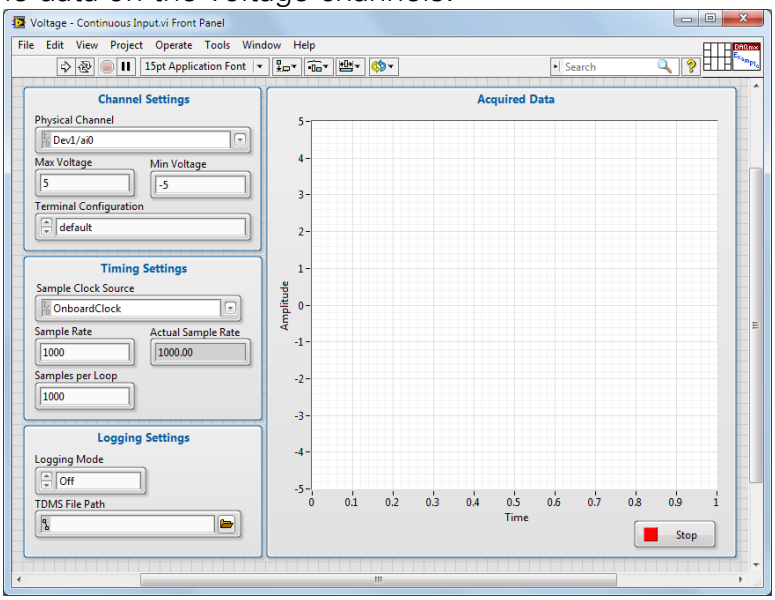
Part B Run a Pre-Built Example

Estimated time: 10 minutes

Now that you are familiar with the different measurement types provided by the NI-DAQmx examples, you can open and run several that use some of the hardware in your hands-on kit. We will start with a simple voltage measurement.

1. In the NI Example Finder, navigate to **Hardware Input and Output » DAQmx » Analog Input** and double click **Voltage – Continuous Input.vi**. This will launch a VI to continuously measure a voltage channel.

2. When the VI launches, the front panel will be displayed. This is the user interface and includes controls and indicators for interacting with your code. In this particular example, notice the controls for selecting the channel, voltage input ranges, timing parameters and logging location. Additionally, the front panel includes a graph indicator for instantly viewing the data on the voltage channels.

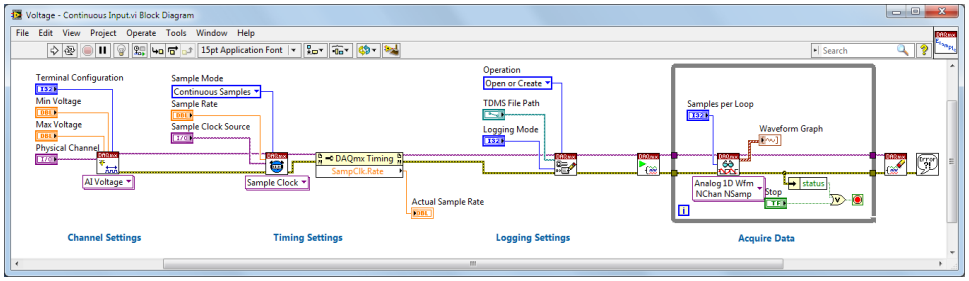


3. To view the code behind the scenes of this front panel, select **Window » Show Block Diagram** or use the keyboard shortcut **<Ctrl+E>**.

4. Expand the block diagram window so that you can see more of the source code.

   You will see a series of VIs used to program your DAQ device to acquire a voltage signal. LabVIEW uses dataflow to pass information along wires to sequential VIs. This programming paradigm enables you to better

program like you think.



5. The block diagram is organized into the following sections:

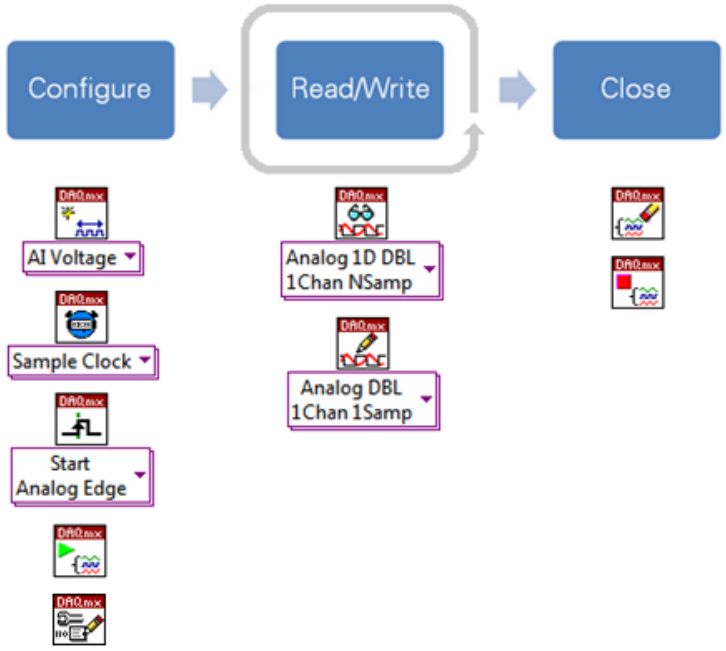
    1. Channel Settings – In this section, the *DAQmx Create Channel VI* configures the channel on the DAQ device that you intend to use. In this example, you can configure the range of the device and terminal configuration (single ended versus differential).

    2. Timing Settings – In the Timing Settings section, the *DAQmx Timing VI* is used to configure the sample clock on the DAQ device. Using this VI, you can configure the sample rate, the sample clock source, and the sampling mode you intend you use. In this example, the default settings use the on-board clock to sample continuously at 1000 Hz.

    3. Logging Settings – The *DAQmx Configure Logging VI* is used to set the location of the logged data file as well as the logging mode.

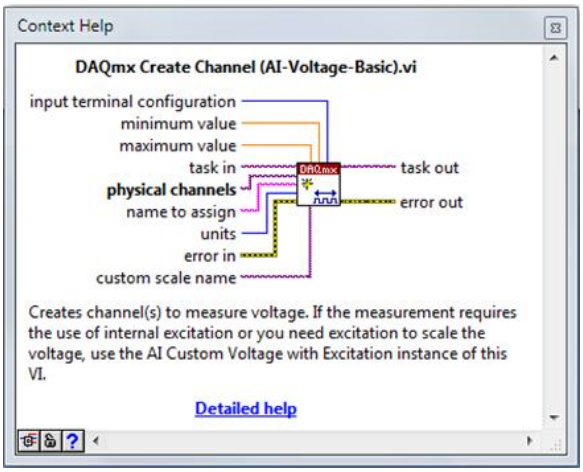
    4. Acquire Data – In this section, the *DAQmx Read VI* is called inside a While Loop, allowing this code to continuously acquire data from the DAQ device until the Stop button is pressed.

Most data acquisition applications will follow this flow –



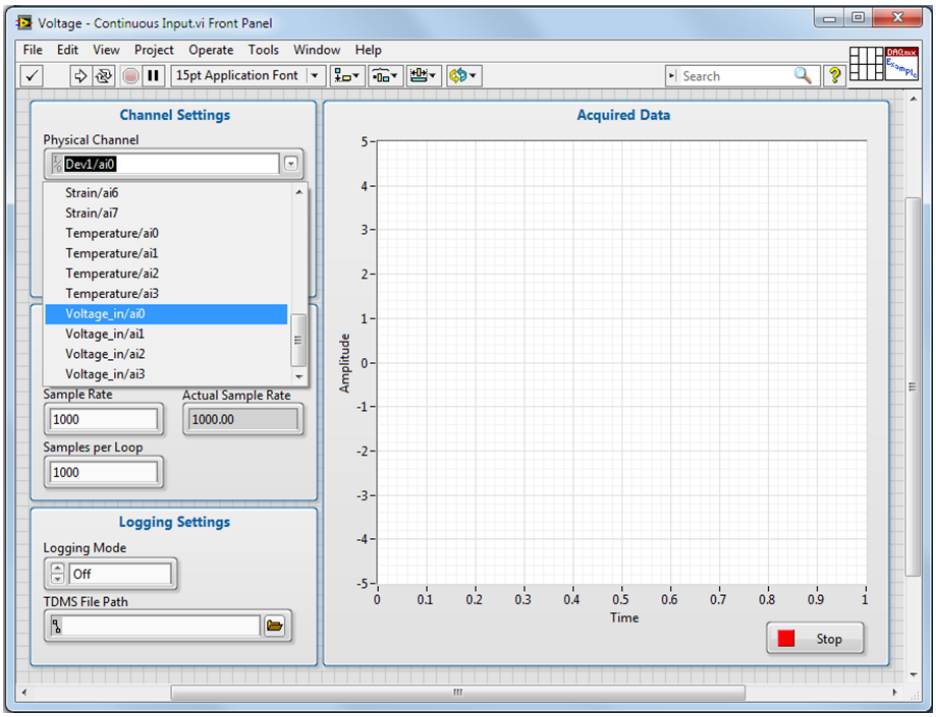
6. Compare the example program block diagram to the VIs shown in Figure 6. If you open other DAQmx example programs, notice that most follow this exact flow.

7. For more information about the elements on your block diagram and front panel, press <Ctrl+H> to bring up the Context Help window. By hovering over elements in your code, you can see brief descriptions and the inputs and outputs of each VI. Required input labels are shown as bold in the Context Help. For more detailed descriptions, you can press the Detailed help link. Try hovering over a few elements on your block diagram.



8. Now that you are familiar with the parts of a basic program, you are ready to acquire data using this example. To do this, switch back to the front panel by selecting **Window » Show Front Panel** or by pressing **<Ctrl+E>**.

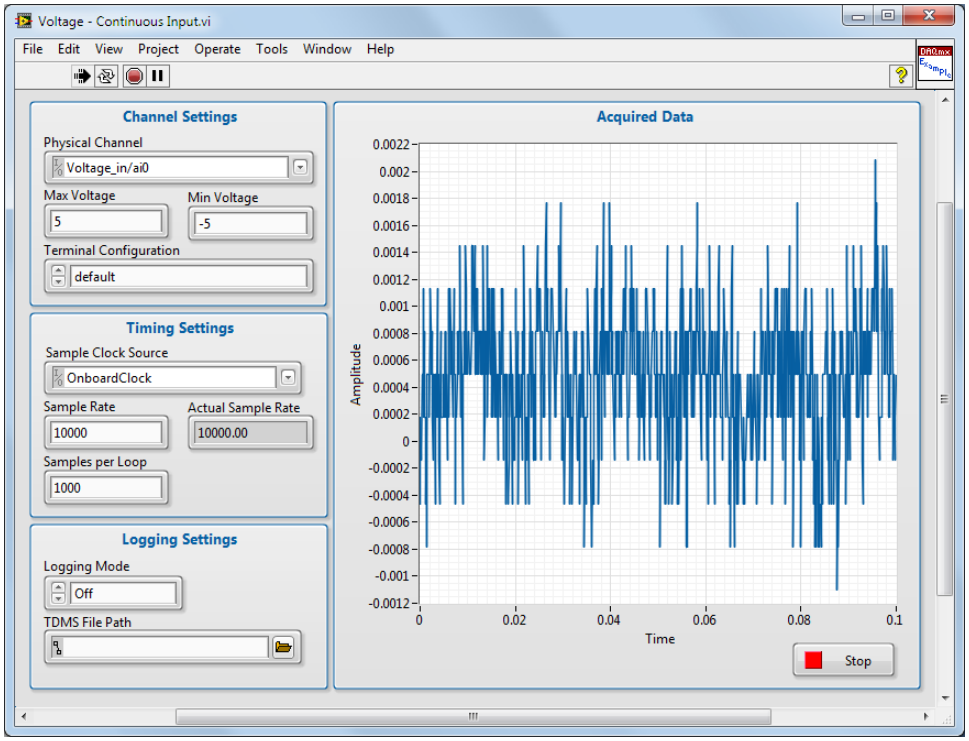
9. Change the Physical Channel to **Voltage_in/ai0**.



10. Change the **Sample Rate** control to **10000** and press the Run arrow () to begin acquiring data.

    Because your signal is connected to the light sensor in the box, you will see the output of the light sensor. This signal will have some noise from

sources in the room but you should be able to see the effects of placing your hand over the sensor. You are sampling the signal along with the noise in the room at 10 kHz - what could be some sources of noise in the room around you? What could you do to get a better idea of what is happening to the signal of interest?



11. Press **Stop** to stop acquiring data.

12. Exit the program without saving.

<End of Exercise>

Exercise 3 – Sound Acquisition and Octave Analysis

- The first goal is to quickly acquire a set of analog (audio) signals data using the DAQ Assistant
- The second goal is to perform analysis on that data and display the result of measurements

Setup

In the following exercises you will configure data acquisition equipment, generate a signal and connect it to an input, AUDIO IN CH0. Then you will acquire the signal from the AUDIO OUT CH0, analyze it using the FFT, calculate total harmonic distortion and noise in the signal (THD, SINAD). Later you will measure sound pressure level and number of additional parameters that are determined by human perception (octave analysis, tone analysis, etc).

Connections:

Please confirm that:

- Channel 0 of the 9234 module is connected to "AUDIO OUT CH0", and channel 0 of the 9263 module is connected to "AUDIO IN CH0";
- Simulator is powered, the switch "SPEAKER" in "ON" position, and "DISTORTION" switch in the position "OFF";

Part A Sound acquisition

Estimated time: 15 minutes

The purpose of this exercise is to use LabVIEW and NI CompactDAQ to quickly

set up a program to acquire audio signal from Analog Input module (NI 9234 or NI 9233).

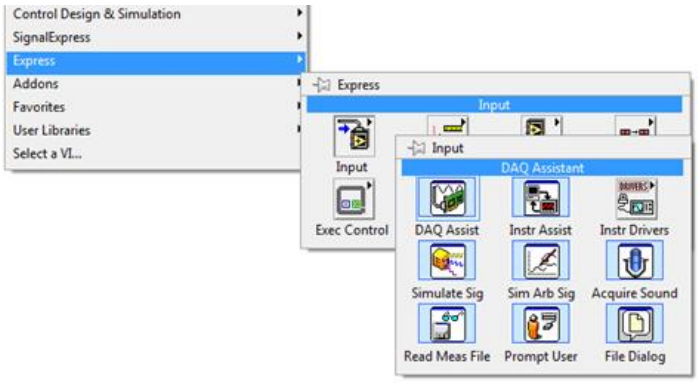
1. From LabVIEW getting started Window click **File » New VI**. Or use **<Ctrl+N>.**

2. Press **<Ctrl+T>** to tile front panel and block diagram windows.

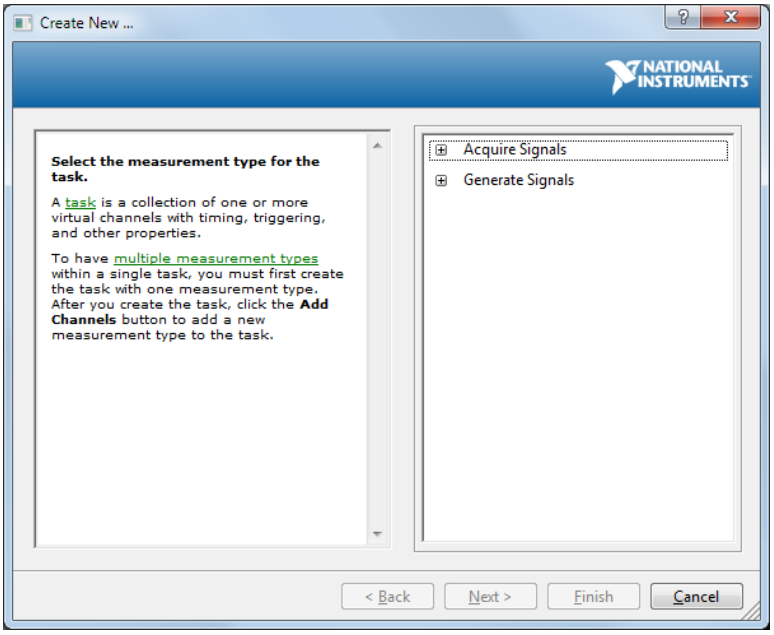
3. Save this VI as **Sound.vi** into folder C:\LabVIEW Workshop\Exercises\Sound Acquisition and Analysis

4. Pull up the Functions Palette by right-clicking the white space on the LabVIEW block diagram window.

5. Move your mouse over the **Express » Input** palette and click the *DAQ Assistant Express VI*. Left-click the empty space to place it on the block diagram.



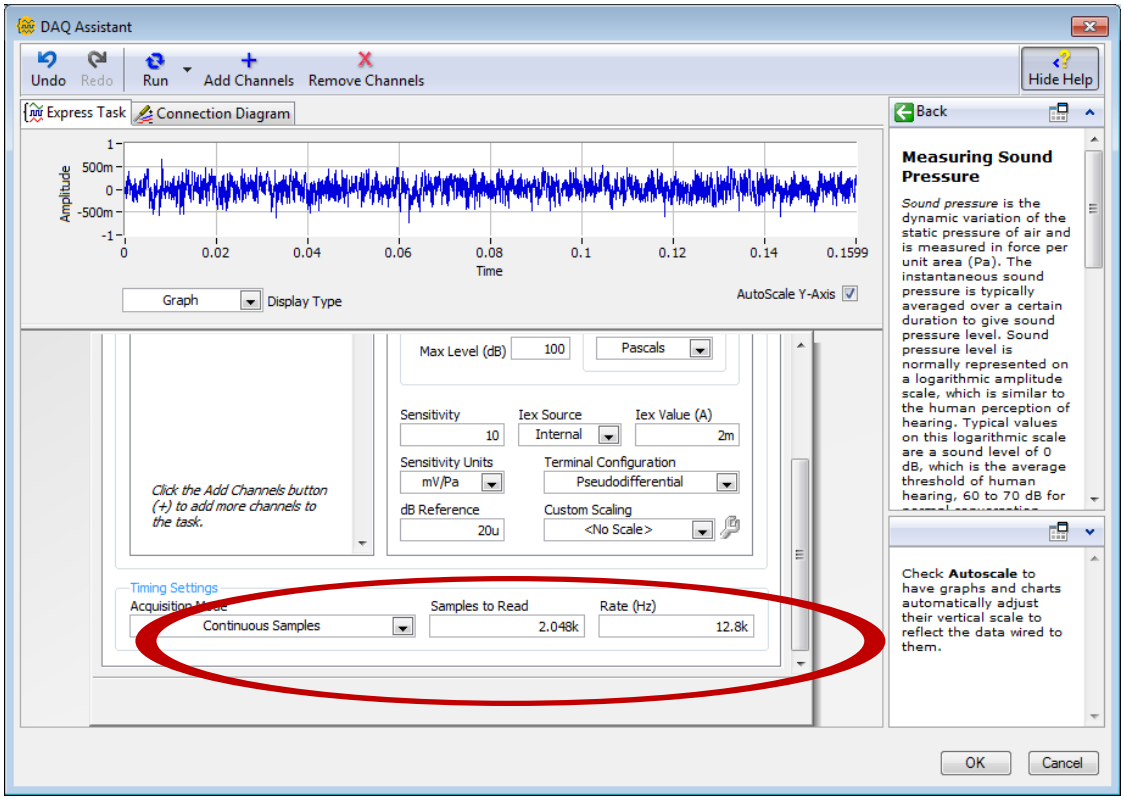
6. The *Create New…* window then appears:



7. To configure a sound measurement application, click **Acquire Signals » Analog Input » Sound Pressure**.
   Click the **+** sign next to the module named **Sound_Vibration (NI 9234/3)**, highlight channel **ai0**, and click **Finish**. This adds a physical channel to your measurement task.

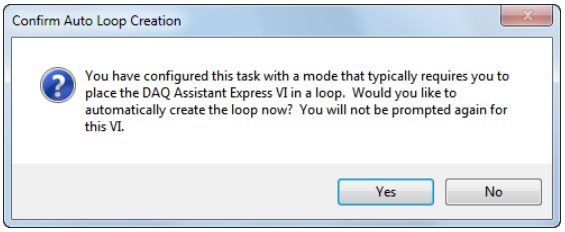
8. Within the DAQ Assistant configuration dialog, change Acquisition Mode to **Continuous Samples**. Change the *Samples to Read* to **2048**.Change the *Rate (Hz)* to **12.8k**.

9. Click the **Run** button.

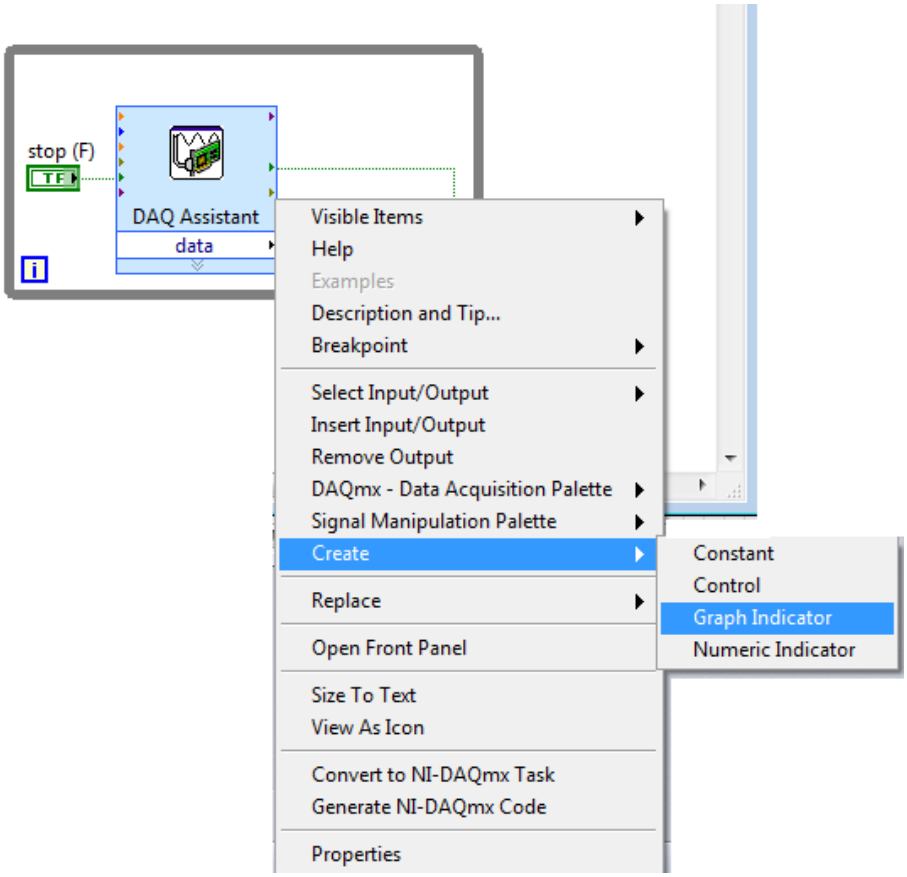


10. Click **Stop** and then click **OK** to close the Express block configuration window to return to the LabVIEW block diagram.

11. LabVIEW automatically creates the code for this measurement task. Since we select a Continuous Samples measurement, it makes sense for the code to be placed within a While Loop, which LabVIEW informs us. Click **Yes** to automatically create a While Loop.



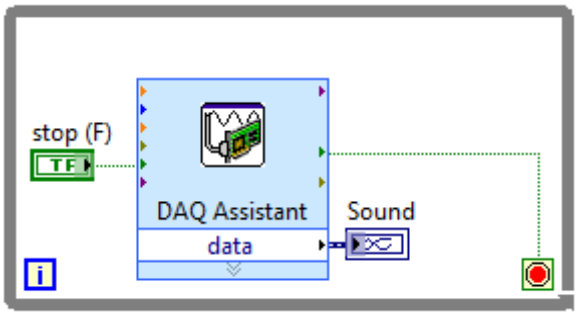
12. Right-click the data terminal output on the right side of the DAQ Assistant Express VI and select **Create » Graph Indicator**.



13. Rename the new Waveform Graph indicator from *Data* to *Sound* by double-clicking on **Data** Caption

14. Notice that a corresponding graph indicator is placed on the front panel.

15. Your block diagram should now look like the figure below. The While Loop automatically adds a Stop button to your front panel that allows you to stop the execution of the loop.



16. Feel free to **Run** the VI to test its functionality. Press **Stop** when you are finished.

17. **Save** the VI.

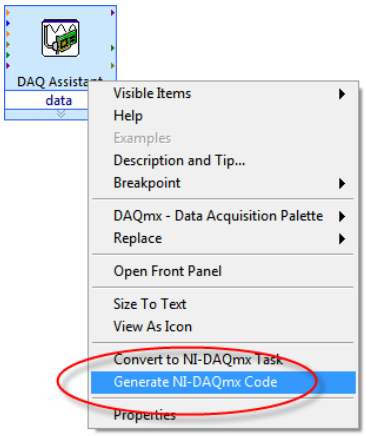
Additional Steps

Express VIs make creating basic applications very easy. Their configuration dialogs allow you to set parameters and customize inputs and outputs based on your application requirements. However, to optimize your DAQ application's performance and allow for greater control, you should use standard DAQmx driver VIs. The DAQ Assistant can automatically generate standard DAQmx code; not all Express VIs have this functionality.

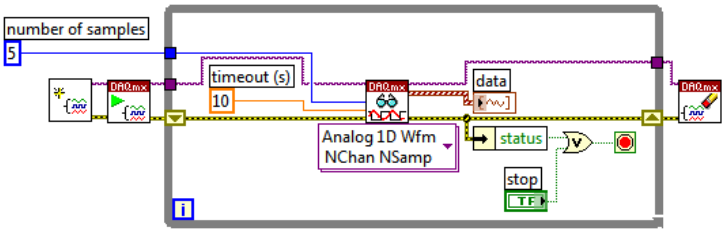
Before you generate standard DAQmx code from the DAQ Assistant, you need to remove any unnecessary code that was automatically created by the Express VI.

18. Right-click the While Loop and select **Remove While Loop**. Then click the **Stop** button and press the **<Delete>** key to remove the Stop button. Repeat these actions for the Temperature Graph as well as any additional wires that may remain. You can press **<Ctrl+B>** to remove all unconnected wires from a block diagram.

19. When you are left with only a DAQ Assistant, right-click on the DAQ Assistant Express VI you created in this exercise and select **Generate NI-DAQmx Code**.



Your block diagram should now appear something like this:



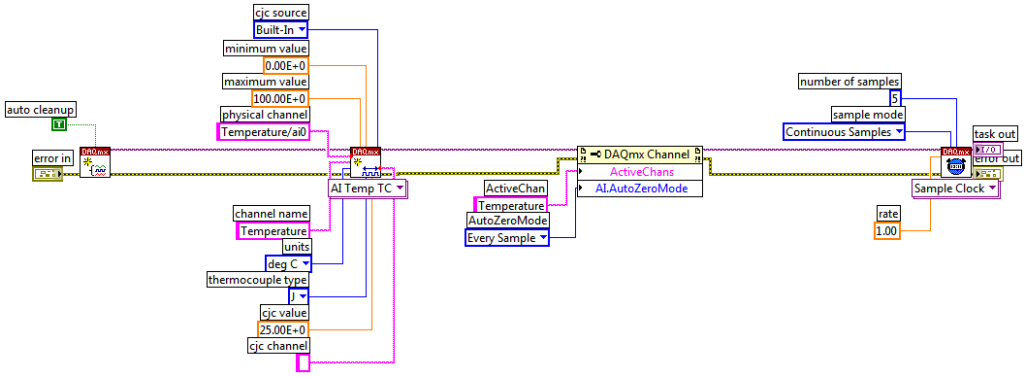
The Express VI has been replaced by four VIs. We'll examine their functionality in the following steps.

20. Open Context Help by clicking the **Context Help** icon () on the upper-right corner of the block diagram.

21. Hover your cursor over each VI and examine their descriptions and wiring diagram.

22. *DAQmx Read.vi* reads data based on the parameters it receives from the *Untitled VI* on the far left.

23. Double-click the *Untitled VI* and open that VI's block diagram (code shown below).



All the parameters that are wired as inputs to the different DAQmx setup VIs reflect the settings you originally configured in the DAQ Assistant Express VI.

Note: By moving these parameter and setup VIs onto the block diagram, you could now programmatically change their values without having to stop your application and open the Express VI configuration dialog. This can save development time and possibly optimize performance by eliminating unnecessary settings depending on your application.

24. Close the *Untitled VI*. When prompted, **do not save**.

25. Save the created Sound.vi. .

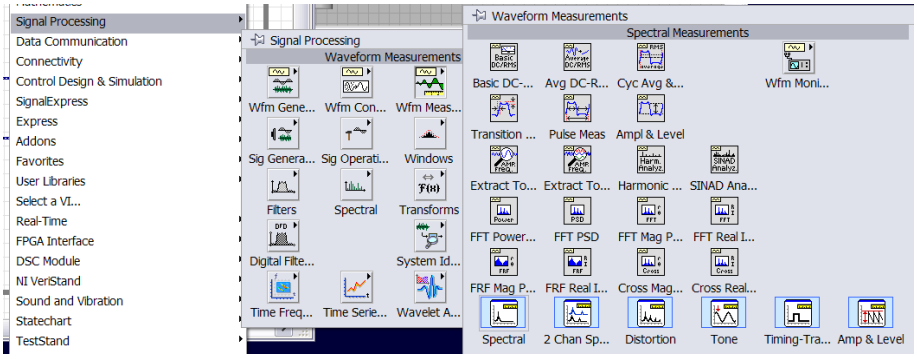
Part B. Data Analysis

Estimated time: 15 minutes

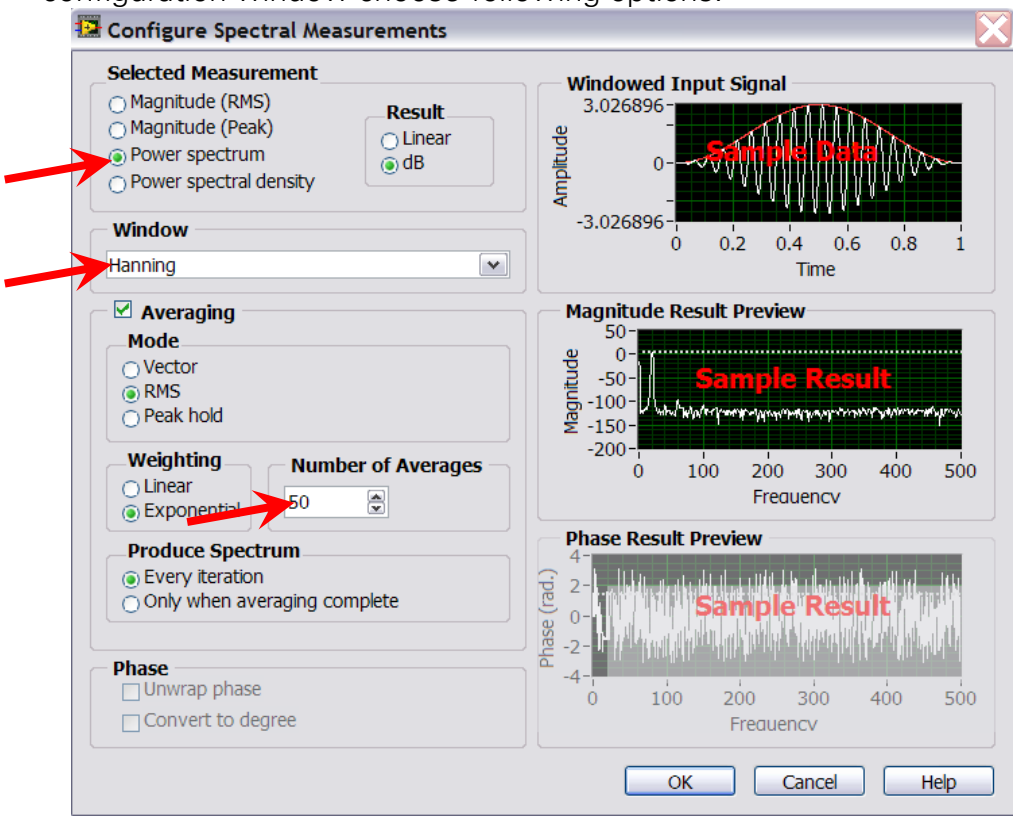
1. Use the previously created Sound VI for acquisition of data. You will

acquire the data on the channel 0 of the 9234/3 module.

2. Now we will add the Power Spectrum calculation to our code. Switch to Block Diagram by pressing Ctrl+E. By right-clicking on Block Diagram open Functions palette. Choose Signal Processing -> Waveform Measurements -> Spectral.



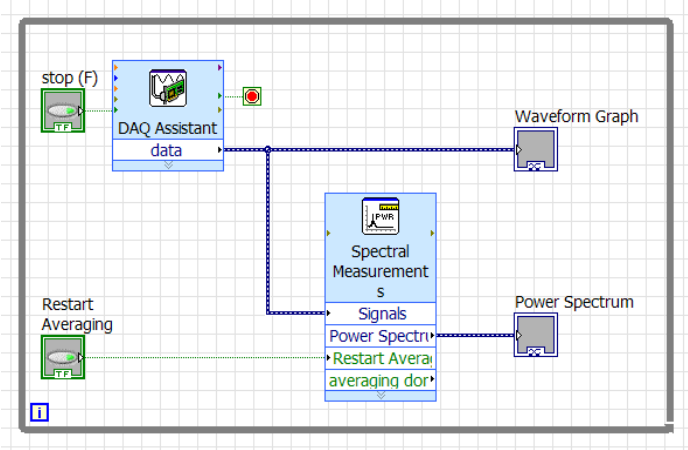
3. Place "Spectral Measurements.vi" on Block Diagram and in the configuration window choose following options:



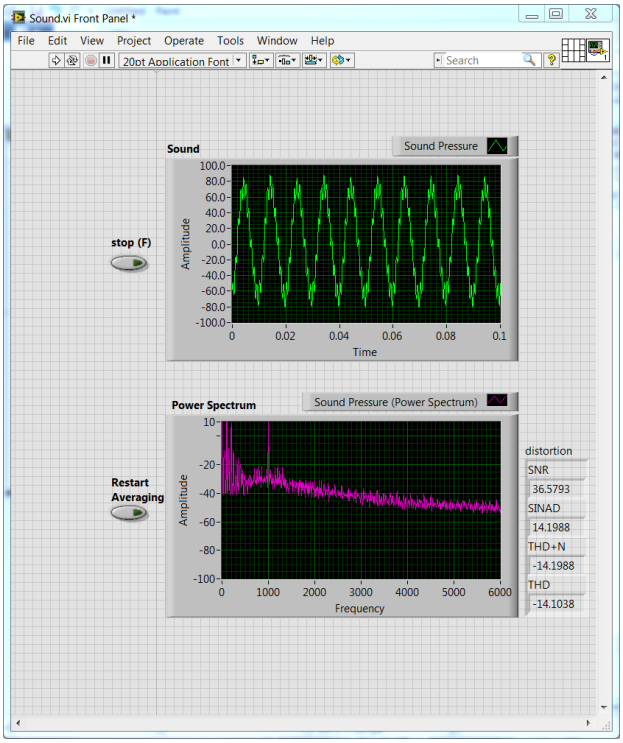
4. Click "OK"
5. Connect the wire containing measured signal to the "Signals" input of the "Spectral Measurements" function.

6. Finally, plot the power spectrum on a graph (right-click Power Spectrum of Spectral Measurements and select Create -> Graph Indicator).

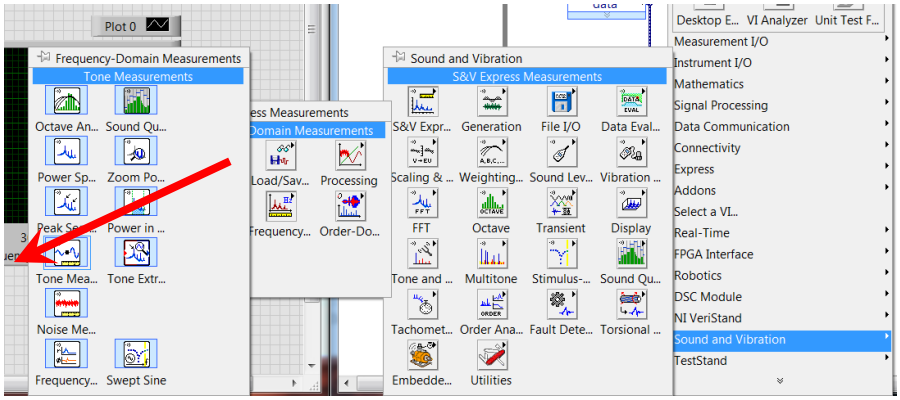
7. Create a Boolean Control (right-click on "Restart Averaging" input, Create -> Control). Rearrange the block diagram so that it looks similar as on the figure below:



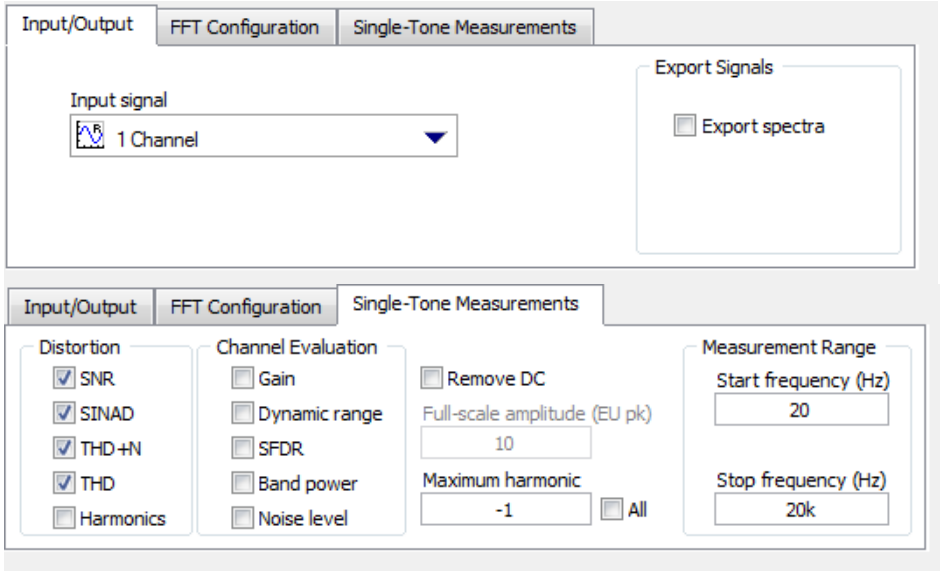
8. Rearrange the Front Panel, disable the auto-scaling of X axis (right-click on the axis and uncheck Autoscale X).
9. Adjust the range of the X axis based on the values of measured signal (double-click on the value on axis scale and write the new value):



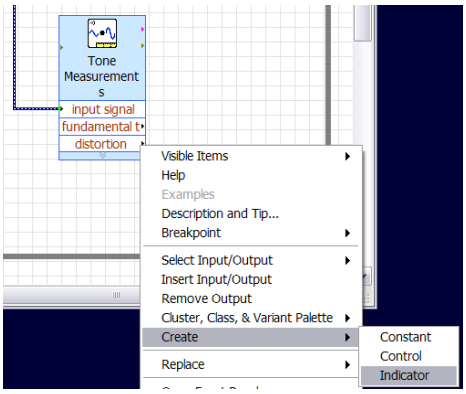
10. Now go to the Functions palette and choose Sound and Vibration -> S&V Express -> Frequency Domain -> Tone Measurements.vi and place it into the While Loop.



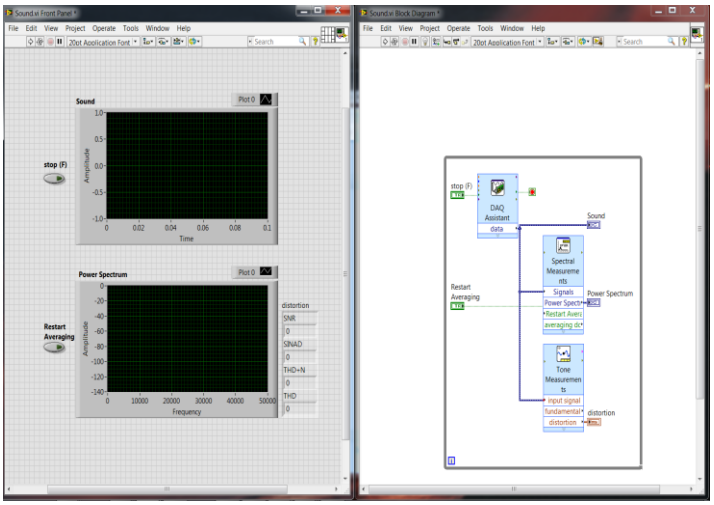
11. In the configuration window switch to "Input/Output" tab and clear "Export spectra", in the tab "Single-Tone Measurements", select: SNR, SINAD, THD, THD+N, clear SFDR and click "OK". Connect the wire containing measured signal with the "input signal" input .



12. Create an indicator for the output "distortion" (right-click, Create -> Indicator).



13. Rearrange both Front Panel and Block Diagram to get a similar look as on the figure below.



14. Open and runGenerator.vi that is located in folder C:\LabVIEW Workshop\Exercises\Sound Acquisition and Analysis. This VI will generate a testing signal on analog output of the 9263 module.

15. Run the Sound.vi and experiment with settings of the Generator.vi.
16. Save the VI.

Part C. Save the data into TDMS file and analyze them using NI DIAdem

Estimated time: 15 minutes

In this part of exercise, you will use the Express VI to save your data into TDMS file. TDMS is a binary file format that can be used to store measurement data with additional information, such as unit, name of the channel, name of operator, etc.

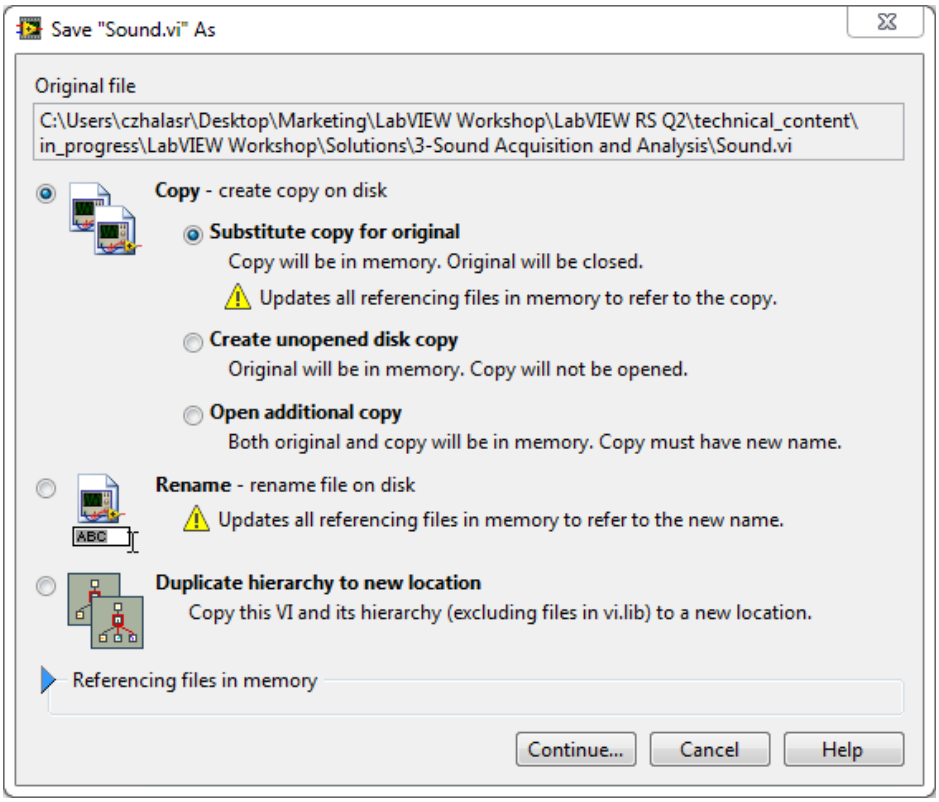
TDMS can be easily viewed using Microsoft Excel plugin or even better using NI

DIAdem software. NI DIAdem is a standalone application that can be used to search through large data set distributed in different location, open multiple types of files, make analysis on the data and automatically generate reports.

Let's use the **Sound.vi** that you have already created and add the functionality of data logging:

1. Open the block diagram of the Sound.vi

2. Go to **File -> Save As** and choose **Copy – Substitute copy for original**

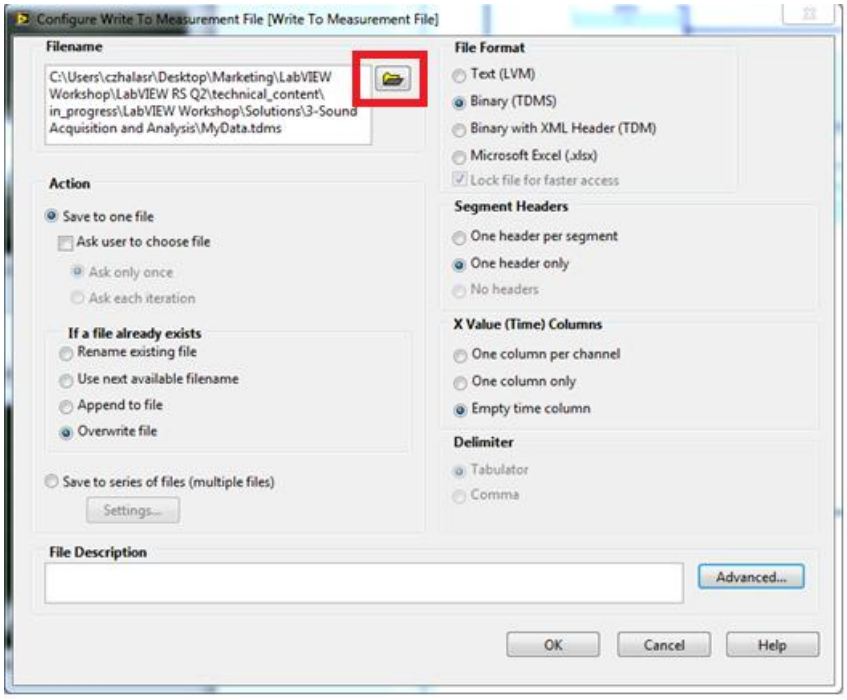


3. Click Continue

4. Save the VI as **Sound_logdata.vi**, this will create a VI with new name. This VI will be loaded in memory

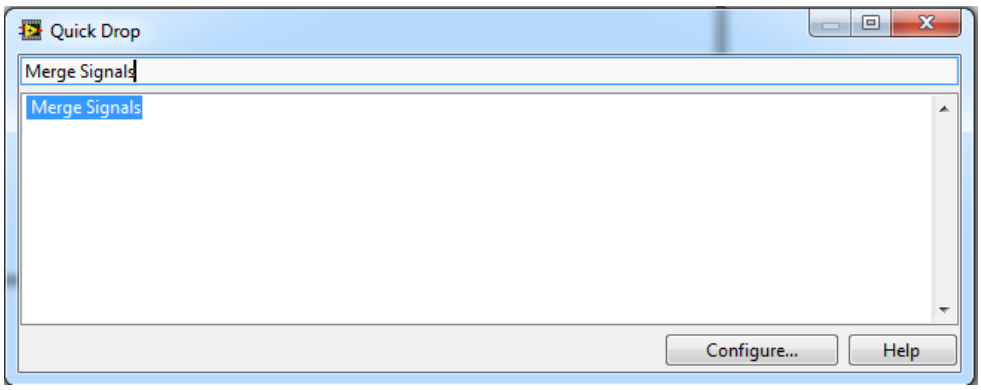
5. Go to the Functions Palette (right-click anywhere on block-diagram) and choose **Express -> Output -> Write To Measurement File**

6. Place the Express VI on block diagram and configure it in following way. Use the browse button to specify path for the file to be saved. Save the file into the exercise folder.



7. Click OK

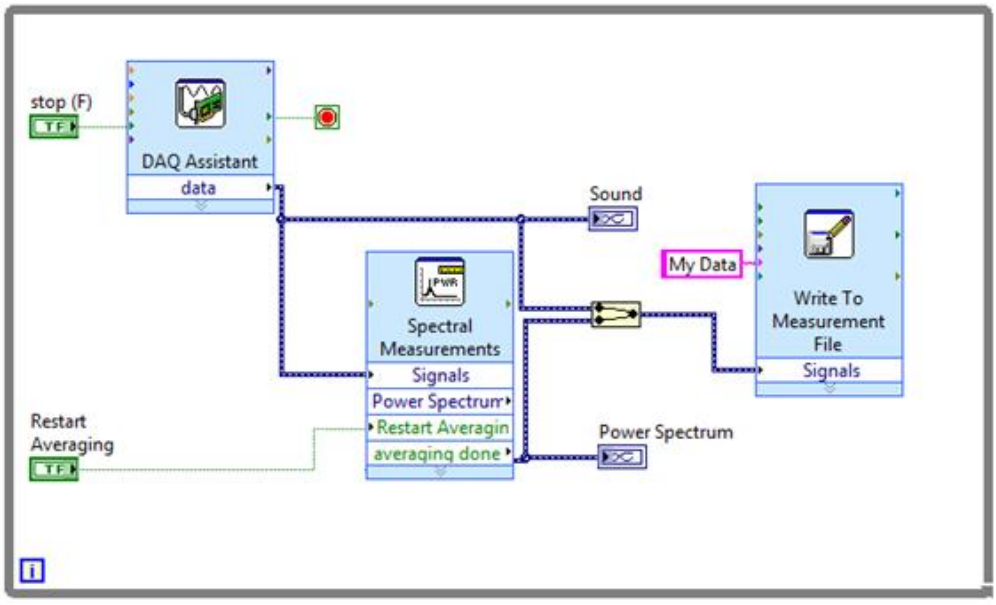
8. Now we will use function Merge Signals to merge measured signal and it's FFT. When you know the name of the function you are looking for, you can use "Quick Drop". Just press **<Ctrl+Space>** , type **Merge Signals** and press Enter.



9. Click on the block diagram where you want to place the function. LabVIEW will automatically place it on that place.

10. Right-click on the Comment input of the **Write to Measurement File** function and select **Create -> Constant**. Type "My Data" into the string constant. Rearrange the block diagram to resemble following figure.

11. Save the VI.



12. Run the Generator.vi and then Sound_logdata.vi. You are acquiring data and saving it into TDMS file. After few seconds press the Stop button on front panel of your acquisition VI. Stop the Generator.vi as well.

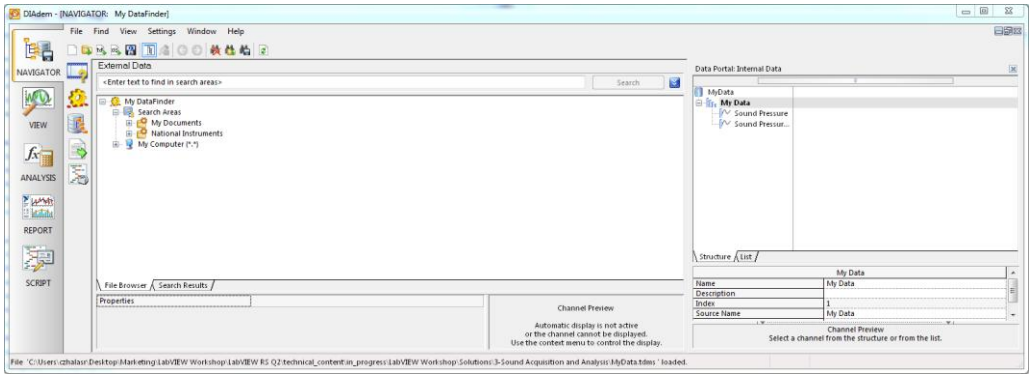
    Useful Tip

    You can open your TDMS file using Microsoft Excel or OpenOffice Calc. Just go to www.ni.com page and search for TDM Excel Add In or TDM Open Office Add In. After installing the plugin, you can view your TDMS data.

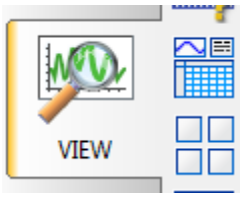
13. From the Windows Explorer right-click the TDMS file and select Open with -> DIAdem 2014.

14. Your data will open in the software for data analysis and report generation NI DIAdem.

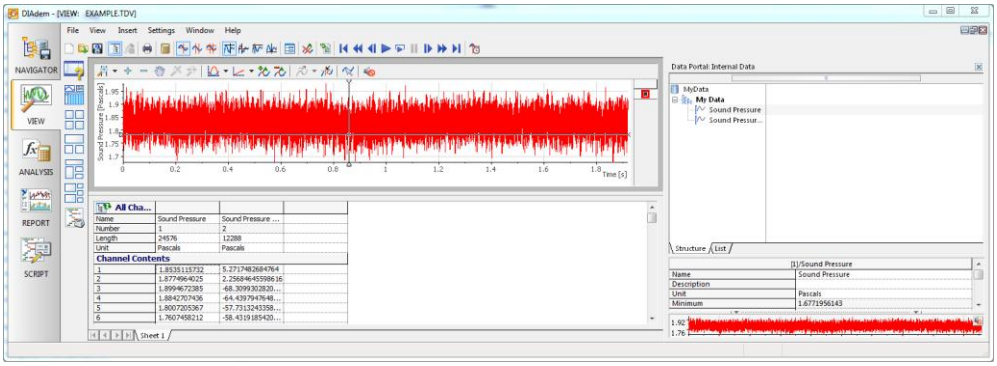
15. You can see your data in the right part called Data portal



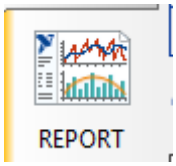
16. Click on the VIEW icon in the left part of DIAdem window. DIAdem will switch into viewing mode.



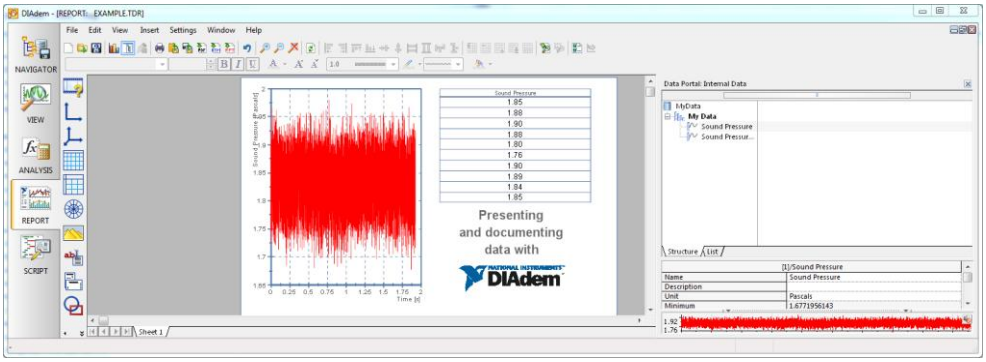
17. You should see your signal in the VIEW window. You can use cursors and zoom tools to observe the signal. Take some time to experiment with viewing tools.



18. Click on the REPORT tab and switch the view into REPORTwindow.



19. Drag and drop your Sound Pressure signal into the graph and also into the table.



20. You have created a report containing graph and values in table. You used just one out of many predefined DIAdem templates. You can also create your own templates and use built-in scripting to automatically analyze data and generate report. For more information refer to DIAdem Help files.

21. Locate the **PDF Export** icon in the top toolbar and click on it.

22. Select the path to your exercise folder and save your report as **MyReport.pdf**.

23. Locate the PDF report in the exercise folder and examine it.

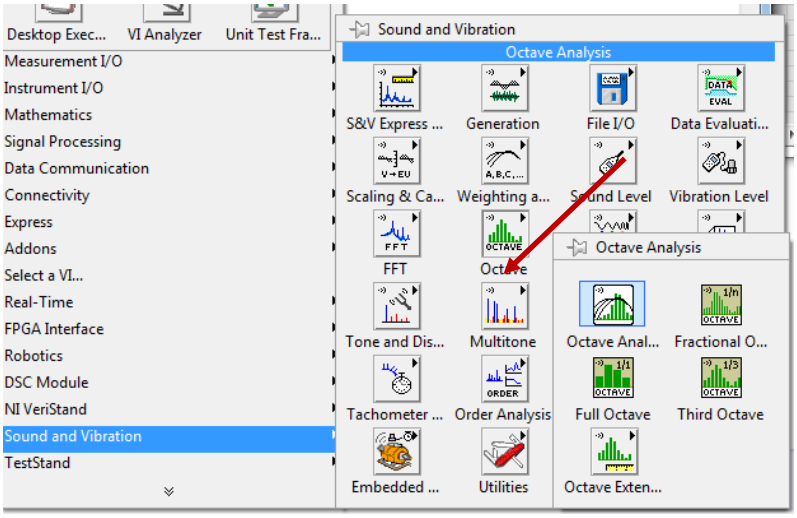
24. Close DIAdem and all VI's that are opened.

Part D. Octave analysis of the audio signal

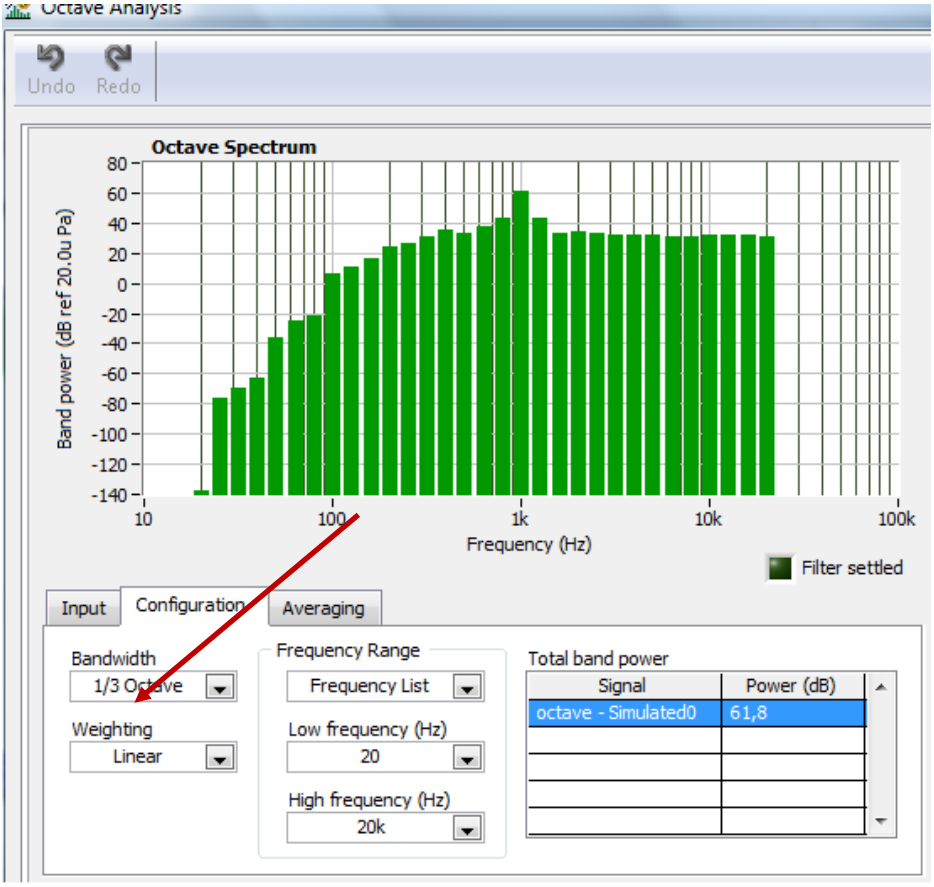
Estimated time: 15 minutes

Octave analysis processes the analog audio signal in a similar way as human ear. Time-domain signal is processed through a series of band-pass filters and the average value in each of the bands is calculated. We can see the results in the form of a bar graph. Features in the Sound and Vibration Toolkit used with National Instruments DSA acquisition devices are capable of octave analysis, ranging from a full octave to 1/24 octave and compliant with ANSI and IEC standards.

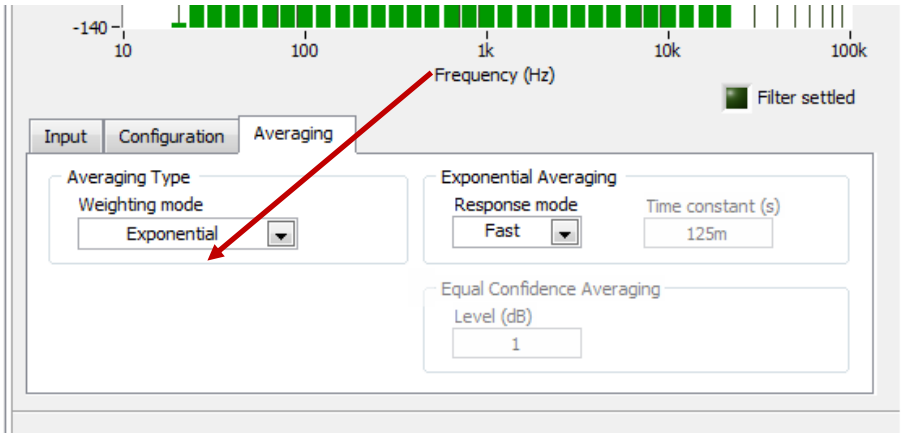
1. Let's add octave analysis to our VI. Go to the folder **C:\LabVIEW Workshop\Exercises\3-Sound Acquisition and Analysis** and open the Octave_Analysis.vi. This VI is uses DAQ Assistant to measure signal from channel 0 of 9234/3 module.
2. Go to Functions Palette and choose **Sound and Vibration->Octave -> Octave Analysis**



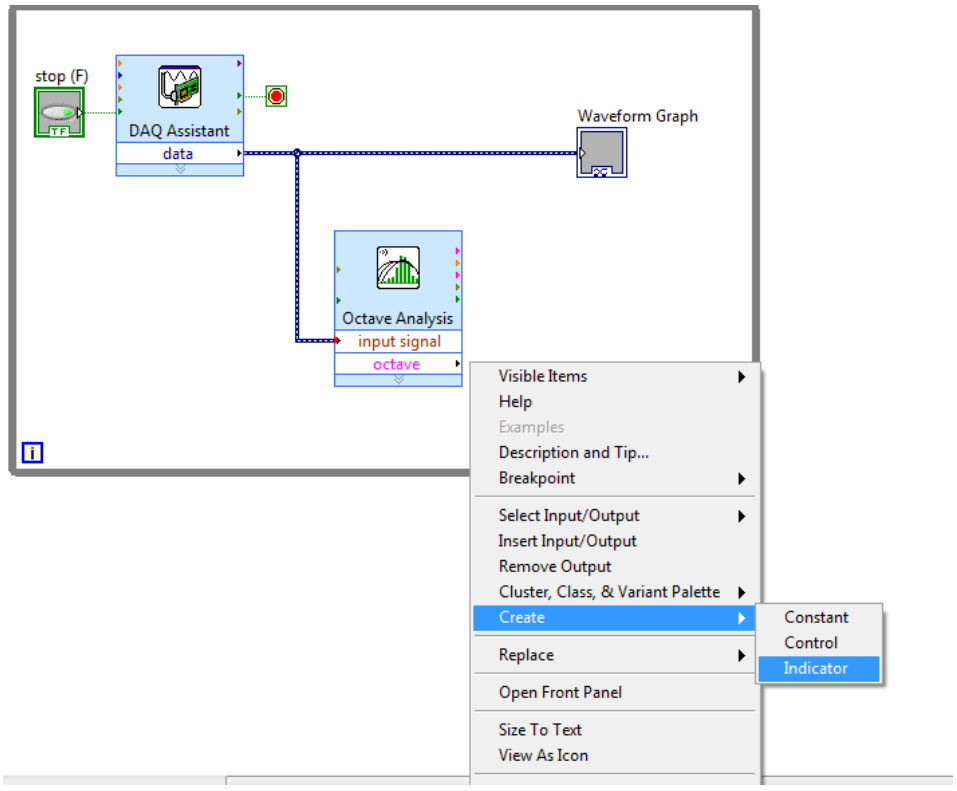
3. Place the VI on Block Diagram. In the configuration window choose **1/3 octave bandwidth**, and linear weighting filter (**Weighting->Linear**)



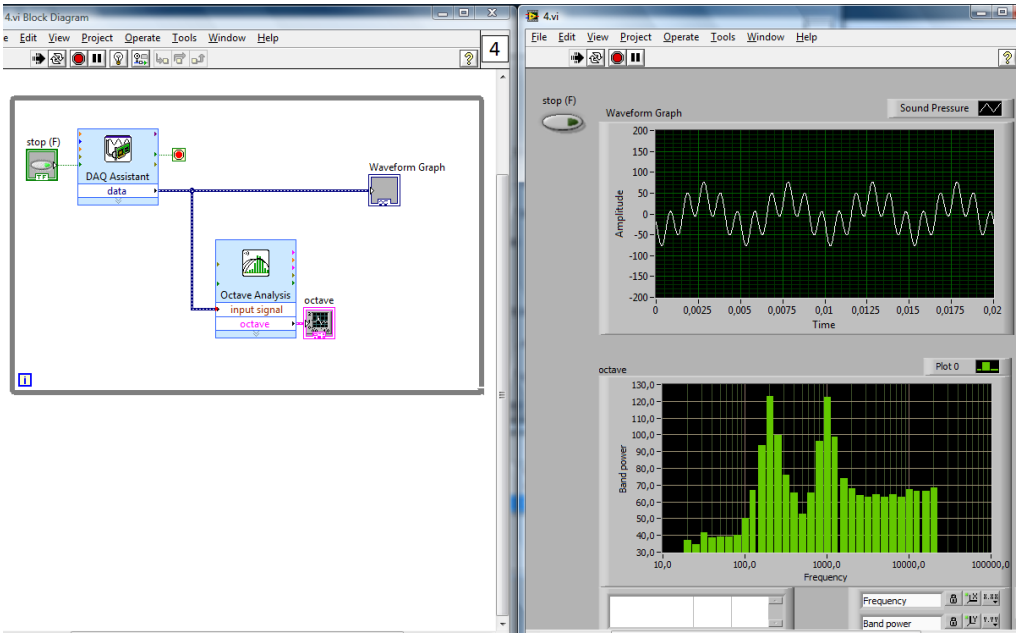
4. In the Averaging tab choose Exponential.



5. Connect the output signal from **DAQ assistant** to the input of the "**Octave Analysis**". Create an indicator for the "octave" output. Rearrange the Block Diagram according to the figure below:



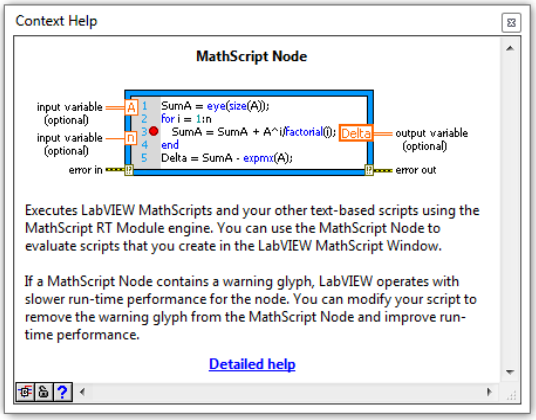
6. Run both the" Generator.VI" and " Octave Analysis.vi" and experiment with different setting of generator.

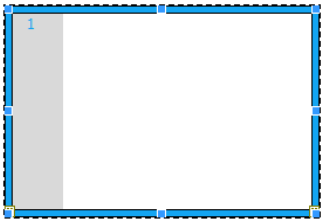


7. Experiment with configuration of the" Octave Analysis" block (double-click on the block to get to configuration window).
    - Change the weighting filter for the A filter
    - Change the analysis bandwidth for 1/24 octave
    - Confirm the changes in "Octave Analysis.vi"
    - Run the VI and change the settings of generator

Challenge

In LabVIEW, you can use MathScript Node structure to use textual language for calculations or directly import commonly used m. files. Create a MathScript Node structure on block diagram of your VI, go to help and study the functionality. Use MathScript Node to add offset of 1 to your data and calculate mean value of your measured signal. Hint: to add inputs or outputs you need to right-click on the border of MathScript Node.





Useful Tips

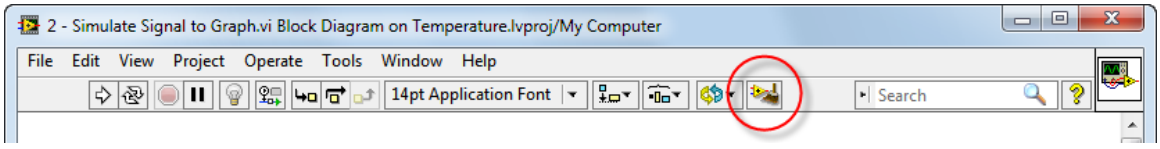
LabVIEW provides several tools that can help you develop your applications. The next few steps will show how to use some of the most important programming assistance tools.

Block Diagram Cleanup

As you program, and especially as you learn how to program in LabVIEW, you are not always thinking about layout and readability. This can result in a poorly organized block diagram.

LabVIEW's Block Diagram Cleanup is a built-in tool that organizes your code, making it easier for you and others to understand how your program functions.

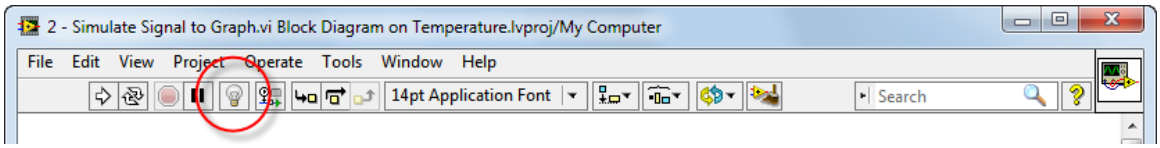
1. Press the Block Diagram Cleanup icon found on the menu bar.



Your block diagram should now be organized, with cleaner wires and an even distribution of code elements.

Highlight Execution

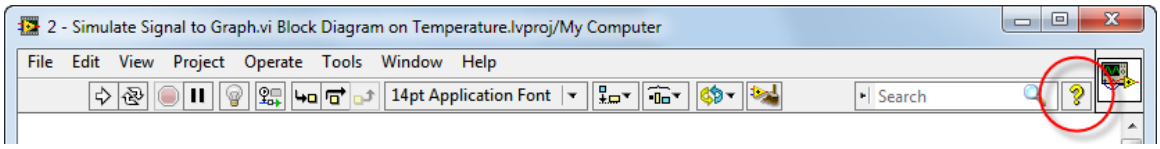
2. Press the **Highlight Execution** button on the menu bar. Notice that the light bulb icon now appears to be on.



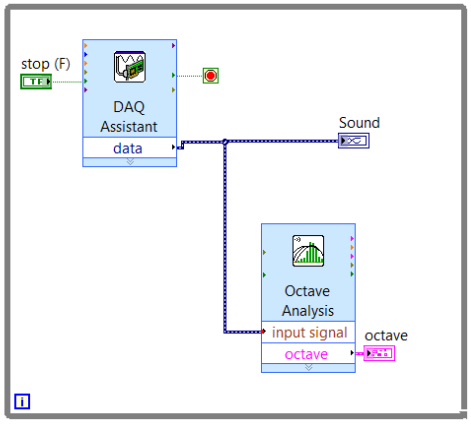
3. Run your application with Highlight Execution turned on. Press the **Run** arrow and watch as your code executes step-by-step. While not always necessary for simple applications, the Highlight Execution tool is a powerful resource for troubleshooting complex programs and determining if your code performs as expected.

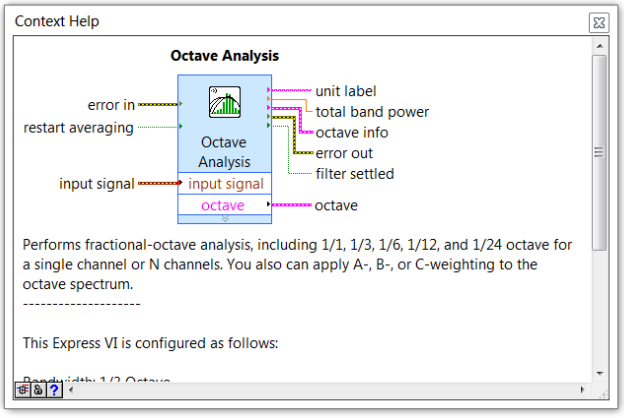
Context Help

4. Press the **Context Help** button in the upper right portion of the block diagram.



5. With Context Help active, hover your cursor over different objects on the block diagram and front panel of Octave Analysis.vi. As you do so, the Context Help Window provides details including descriptions and wiring diagrams.





6. Right-click the block diagram and navigate around the palettes. Notice that

the Context Help window provides details on the objects while they are in the palettes. Also notice that for some objects, the Context Help window provides a link for Detailed Help. This link will open the *LabVIEW Help* and give you more information.

7. Save *Octave Analysis.vi* and close.

Congratulations! You have just finished the LabVIEW Workshop exercise set.

Now you have a set of basic knowledge that is needed for LabVIEW programming, so you can download evaluation version of LabVIEW from www.ni.com/trylabview and start your first real project!

However, LabVIEW environment covers so many areas that there is always something new to explore and learn. If you want to take a professional training with NI Instructor, go to the webpage ww.ni.com/training, or even better – contact your local Sales representative, who can advise you which course is suitable for your needs.

Thank you for your participation!

National Instruments

National Instruments Eastern Europe

National Instruments Hungary Kft.
H-1117 Budapest, Neumann János utca 1/E. 2. emelet
Tel: 06 1 481 1400, Fax: 06 1 203 3490
E-mail: ni.hungary@ni.com
Web: hungary.ni.com
06 80 204 704

National Instruments,
Instrumentacija, avtomatizacija in upravljanje procesov d.o.o.
Kosovelova ulica 15, 3000 Celje, Slovenija
Tel: 080 88 64 | Tel: + 386 3 425 4270 | Fax: +386 3 425 4212
E-mail: ni.slovenia@ni.com | ni.serbia@ni.com | ni.croatia@ni.com | ni.see@ni.com
Web: slovenia.ni.com | serbia.ni.com | croatia.ni.com | see.ni.com |

National Instruments Poland Sp. z o.o.
Salzburg Center
ul. Grójecka 5
02-025 Warszawa
Tel: +48 22 328 90 10
Fax: +48 22 331 96 40
E-mail: ni.poland@ni.com
Web: poland.ni.com
Infolinia: 800 889 897

National Instruments (Czech Republic), s.r.o.
Sokolovská 136D
186 00 Praha 8
Česká republika
Tel: +420 224 235 774
Fax: +420 224 235 749
E-mail: ni.czech@ni.com
Web: czech.ni.com
CZ: 800 267 267 | SK: 00 800 182 362

SC National Instruments Romania SRL
B-dul Corneliu Coposu, nr. 167A, et.I, Cluj Napoca, CP 400228
Tel.: 0800 070 071
E-mail: ni.romania@ni.com
Web: romania.ni.com