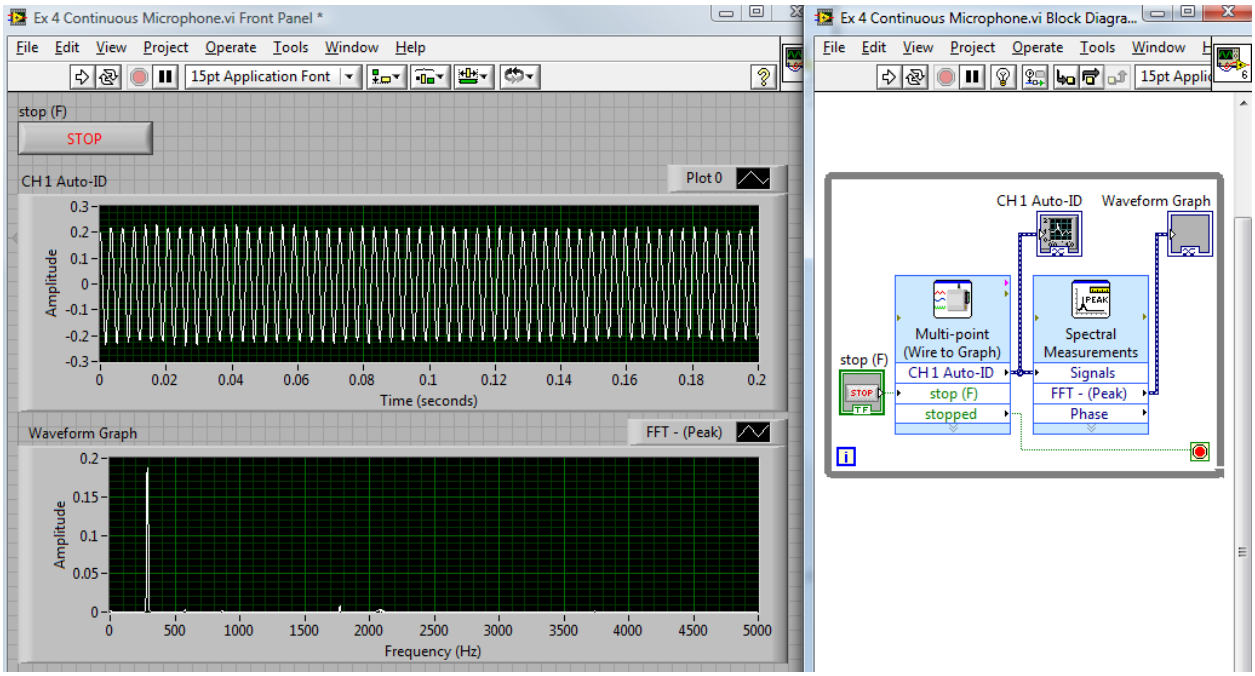
Exercise
4

Continuously Read and Analyze Microphone Data



Completed front panel and block diagram

In this exercise, you will create a simple program using the Analog Express VI and the Spectral Measurements Express VI to continuously collect microphone data and continuously perform an FFT analysis on the data. A Fast Fourier Transform (FFT) is a mathematical algorithm commonly used in digital signal processing to determine the dominant frequencies in a complex waveform.

OBJECTIVES

In this experiment, you will

- Create a LabVIEW VI to continuously collect and analyze data.
- Incorporate LabVIEW Express Analysis functions.
- Use a While Loop to keep collecting and analyzing data until stopped by the user.

MATERIALS

SensorDAQ or LabQuest interface
USB cable
computer
LabVIEW
Vernier Microphone
tuning fork

PROCEDURE

Part I Connect Equipment

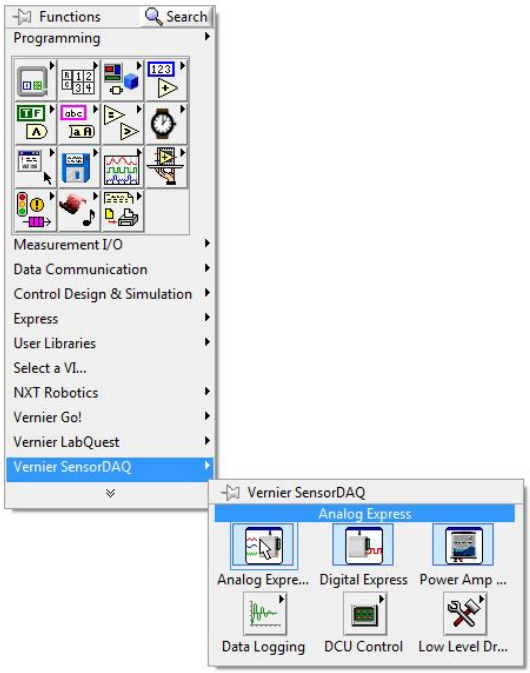
1. Connect the USB cable to the SensorDAQ or LabQuest interface.
2. Connect the other end of the USB cable to any available USB port on your computer. If you are using a LabQuest interface with a power button, turn it on.
3. Connect the Microphone to Ch. 1.

Part II Start LabVIEW and Create a VI to Collect Data

4. Start LabVIEW.
5. In the Getting Started window, click the "Blank VI" link under the New category.
6. View the block diagram by choosing Show Block Diagram from the Window menu (or use the <Ctrl-E> shortcut).
7. Place an Analog Express VI in the block diagram workspace.

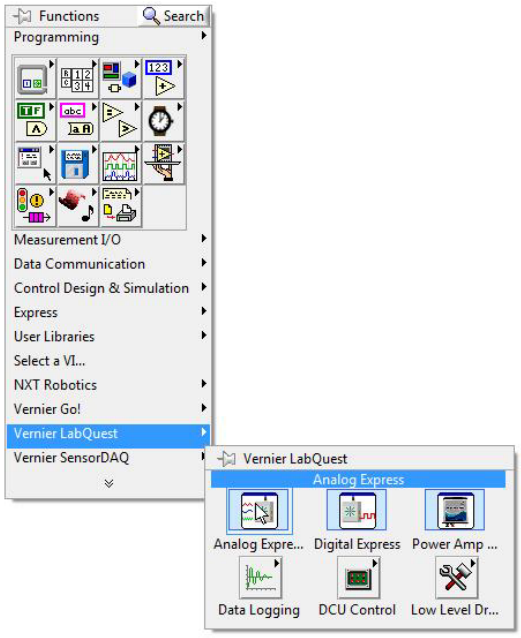
SensorDAQ

If you are using a SensorDAQ, right-click in the block diagram workspace and select Vernier SensorDAQ from the Functions palette. Click and drag the Analog Express VI to the block diagram workspace.



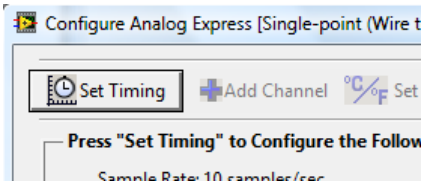
LabQuest Interface

If you are using a LabQuest interface, right-click in the block diagram workspace and select Vernier LabQuest from the Functions palette. Click and drag the Analog Express VI to the block diagram.

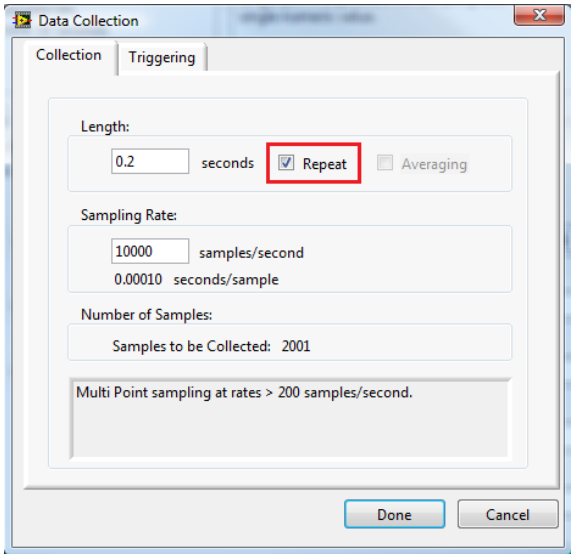


8. After dragging the Express VI from the palette to the block diagram workspace, the Express VI's configuration dialog will open. Note that this step can be slow, depending on your computer.

9. Click the Set Timing button, located in the top-left corner of the configuration dialog.



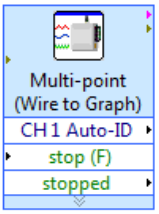
10. First check the Repeat option, and then set the timing with a length of 0.2 second and a sample rate of 10,000 samples/second.



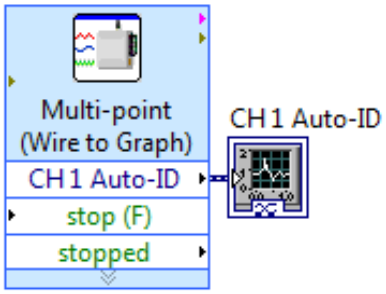
Tip: Repeat means that data collection is repeated. In this case, because the sample rate is greater than 200 samples/second, the Express VI returns data as a multi-point packet containing all of the data points. Checking Repeat configures the interface to stream 2001 samples every 0.2 seconds. This configuration requires that you place the Express VI in a loop to continually read those samples. If Repeat was not checked, the interface would collect the 2001 samples and then stop.

11. Click Done to close the Set Timing window. The Express VI Configuration should now be updated with the new settings.

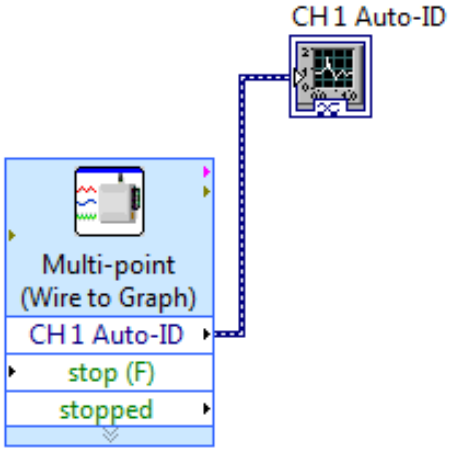
12. Select OK to close the Express VI's Configuration dialog. The Analog Express VI will now be located in your block diagram workspace.



13. Right-click the Express VI's "CH 1 Auto-ID" output terminal and select Create ► Graph Indicator from the shortcut menu.

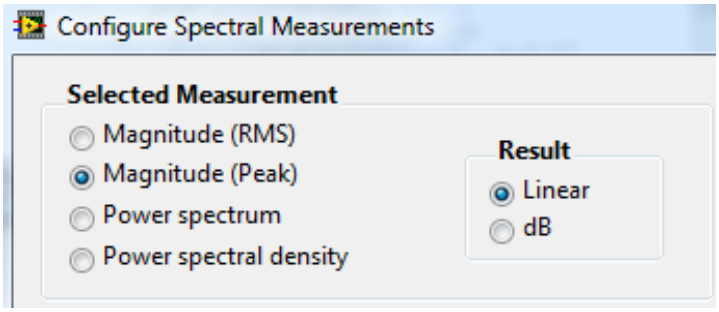


14. Move the graph terminal (click and drag to move an object) to make room to branch off of the wire.

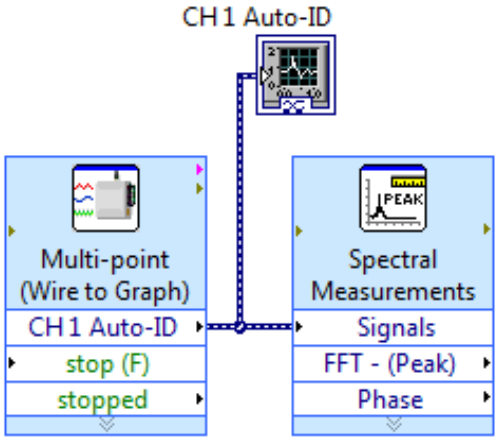


15. Place the Spectral Measurements Express VI to the right of the Analog Express VI on the block diagram. From the Functions palette, select Express ► Signal Analysis ► Spectral Measurements and place it on the block diagram.

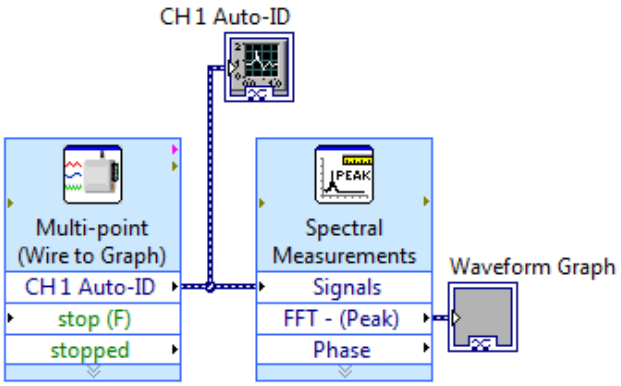
16. In the Configure Spectral Measurements window, select Magnitude (Peak) for the Selected Measurement and Linear for the Result. Leave the rest of the configuration window as the default values. Select OK to close the Express VI.



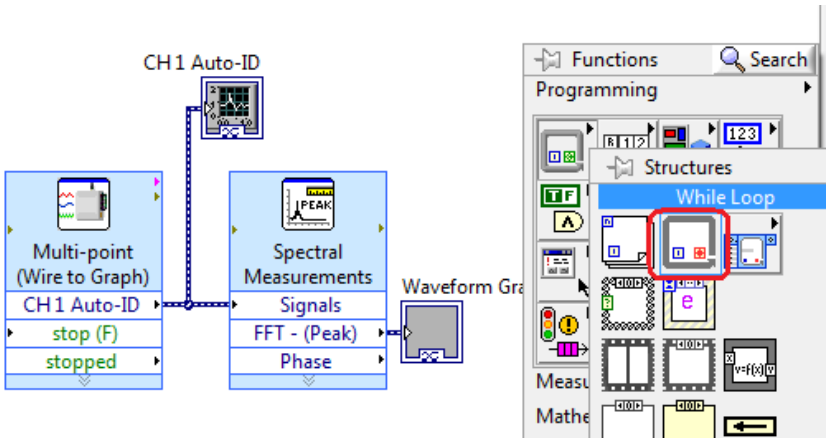
17. Create a wire branch from CH 1 Auto-ID to the Signals input terminal.



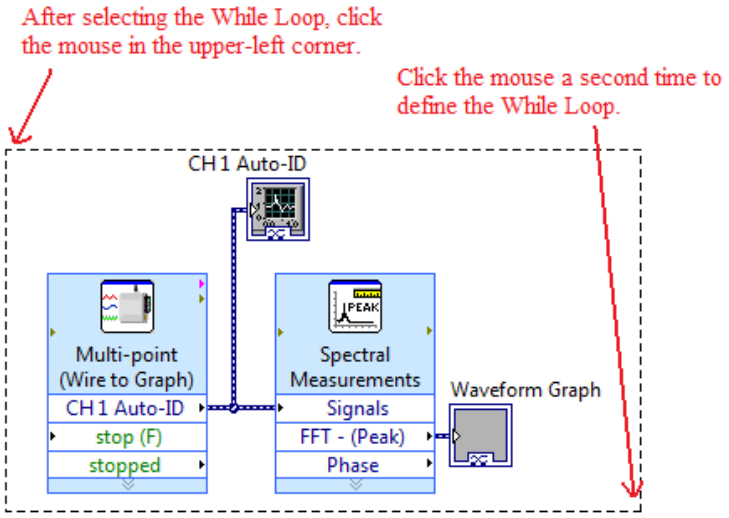
18. Create a Graph Indicator for the FFT output terminal (right-click on the FFT-(Peak) terminal to access the shortcut menu).



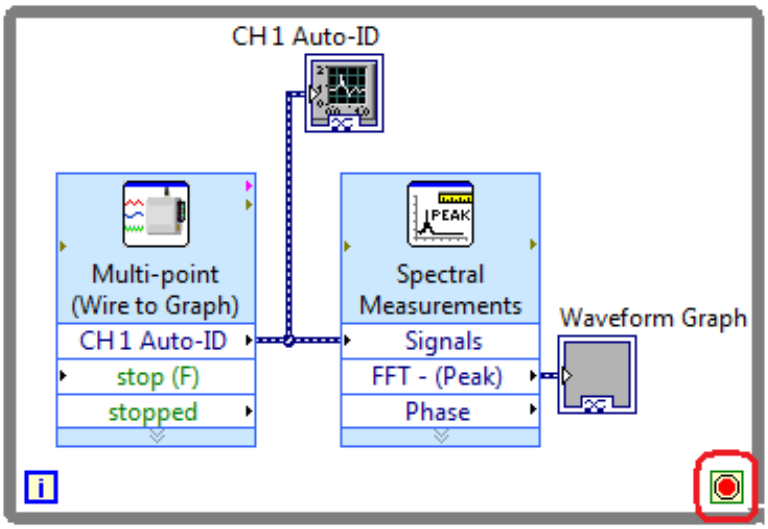
19. Select the While Loop from the Programming ► Structures palette.



20. The cursor will become a special pointer that you use to enclose the code. Click the mouse to define the upper-left corner. Drag the mouse to the location of the bottom-right corner, and click again to create the While Loop.

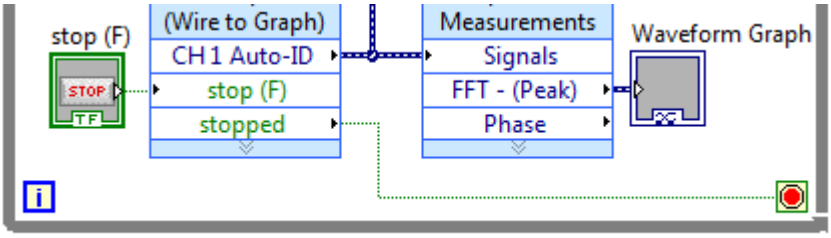


21. The While Loop generates a conditional terminal that is set with the default condition to Stop if True.



22. Right-click the Express VI's "stop (F)" terminal and select Create ► Control. You may have to resize the While Loop to provide room for this control.

23. Wire the Express VI's "stopped" terminal to the While Loop's conditional terminal.

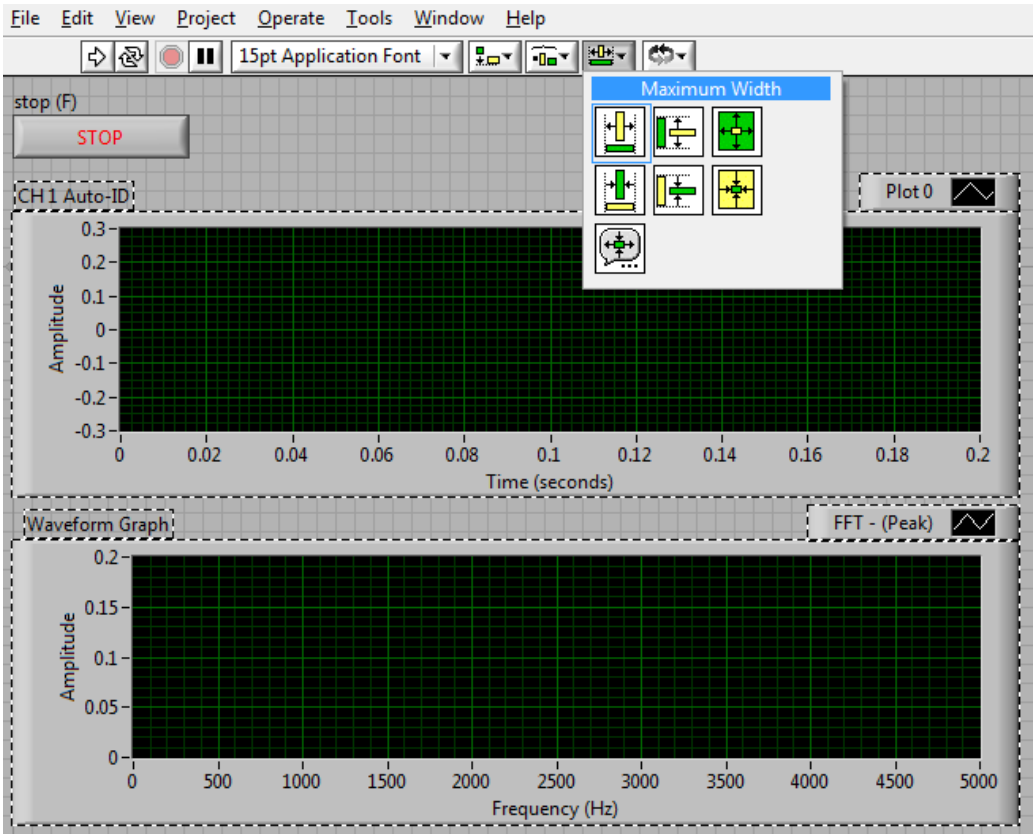


Note: It is important to stop the Express VI and the While Loop in this manner. The data flow of this program insures that the Analog Express VI is stopped before the While Loop is terminated. If the STOP button were wired directly to the While Loop's conditional terminal, the loop would be terminated properly, but the Express VI would not have any input telling it to shut down the interface hardware. This could cause problems with the device the next time it is run; therefore, the order of the program must first stop the Analog Express VI and then stop the While Loop.

24. View the front panel using the <Ctrl-E> shortcut.

25. Rename the x-axis legend of the Waveform Graph to "Frequency (Hz)" and move the graphs and STOP button, if necessary.

26. Resize the two graphs to make them longer. Highlight the two graphs and use the Resize Objects toolbar button to make them the same size.



27. Run the VI. Strike the tuning fork against a soft object (sole of a shoe, for example), and hold it near the Microphone.

28. Click the STOP button to end the program.

EXTENSIONS

1. Analyze your voice while saying the vowels “e” and “o,” and also while singing the vowel sounds.

2. Create an indicator to monitor the number of iterations made by the loop. Recall that the iterations of a loop start at zero, so you must add one to the iteration count.

3. Modify the program to execute the code exactly 25 iterations using a For Loop. To properly stop the program, the Express VI must be stopped prior to ending the program. Do not use a stop button to stop the Express VI; instead use the iteration count and the Equal? Function to stop the Express VI on the very last iteration of the For Loop.

4. In Extension 3, the count input for the For Loop is a constant. Change this constant to a front panel control.