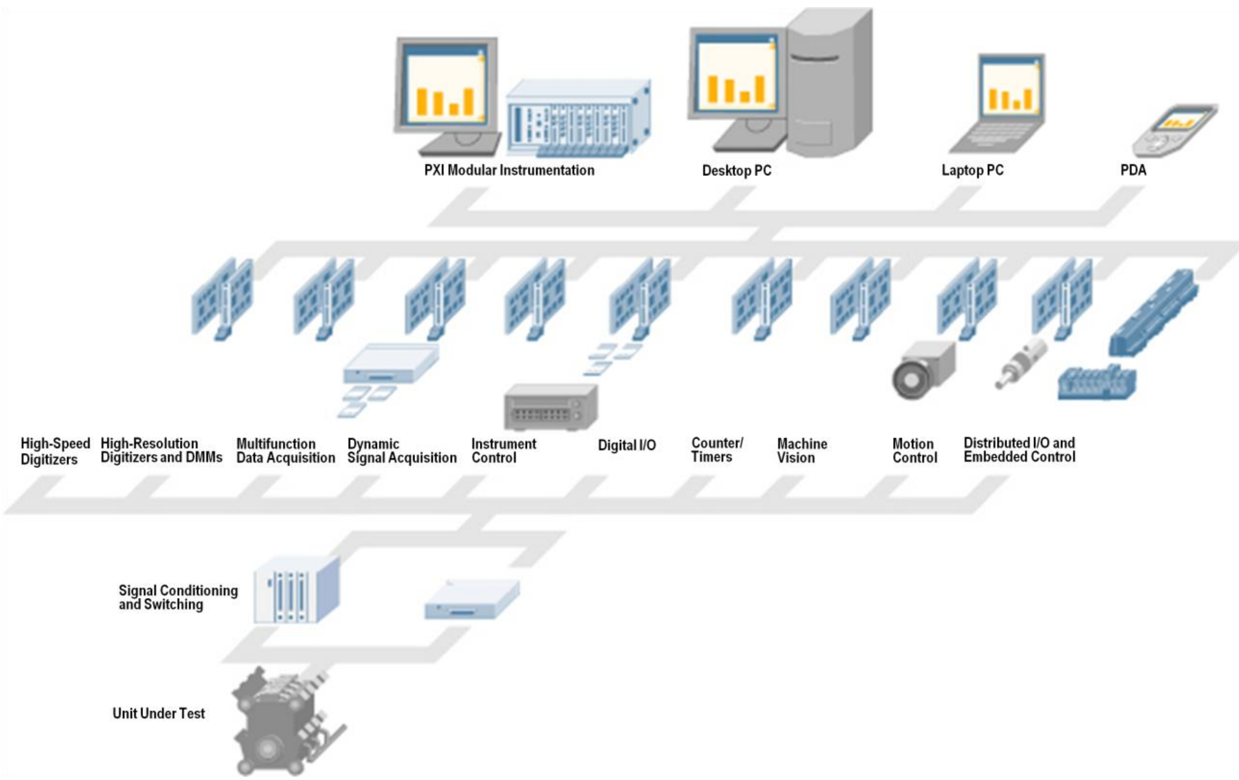
Engineering at a Glance

Hands on Manual

By

Ahmed Khalid Applications Engineer

National Instruments



Exercise 1 – Testing Your Device

Goal

Objective of this exercise is to test functionalit of our NI Elivs using MAX

Procedure

In this exercise you will use Measurement and Automation Explorer (MAX) to test your NI Elvis 2 device.

1. Launch MAX by double-clicking the icon on the desktop or by selecting **Start»Programs»National Instruments»Measurement & Automation**.
2. Expand the **Devices and Interfaces** section to view the installed National Instruments devices. MAX displays the National Instruments hardware and software in the computer.
3. The device number appears in quotes following the device name. The data acquisition VIs use this device number to determine which device performs DAQ operations. You will see your hardware listed as NI Elvis 2: "Dev x".
4. Perform a self-test on the device by right-clicking it in the configuration tree and choosing **Self-Test** or clicking "Self-Test" along the top of the window. This tests the system resources assigned to the device. The device should pass the test because it is already configured.
5. Open the test panels. Right-click the device in the configuration tree and select **Test Panels…** or click "Test Panels…" along the top of the center window. The test panels allow you to test the available functionality of your device, analog input/output, digital input/output, and counter input/output without doing any programming.
6. On the **Analog Input** tab of the test panels, change **Mode** to "Continuous" and **Rate** to 10,000 Hz.
7. On the **Digital I/O** tab notice that initially the port is configured to be all input. Observe under **Select State** the LEDs that represent the state of the input lines. Click the "All Output" button under **Select Direction**. Notice you now have switches under **Select State** to specify the output state of the different lines. Toggle line 0 and watch the LED light up. Click "Close" to close the test panels.
8. Close MAX.

-End of Exercise-

Exercise 2– Signal Acquisition and Analysis with NI Elvis

Goal

Objective of this exercise is to generate and acquire a signal using NI Elvis and then perform operations and analysis on received signal.

Procedure

Complete the following steps to create a VI that acquires data continuously from your DAQ device.

1. Launch LabVIEW.
2. In the **Getting Started** window, click the **New** or **VI from Template** link to display the **New** dialog box.
3. Open a data acquisition template. From the Create New list, select **VI**»F**rom Templat**e»DAQ»D**ata Acquisition with NI-DAQmx.vi** and click "OK".
4. Display the block diagram by clicking it or by selecting **Window»Show Block Diagram**. Read the instructions written there about how to complete the program.
5. Double-click the DAQ Assistant to launch the configuration wizard.
    1. Configure an analog input operation.
        - Choose **Analog Input»Voltage**.
        - Choose **Devx (Elvis 2)»ai0** to acquire data on analog input channel 0 and click "Finish."
        - In the next window you define parameters of your analog input operation. To choose an input range that works well with your input signal, on the settings tab enter **x Volts** for the maximum and **–x Volts** for the minimum. On the task timing tab, choose "**Continuous**" for the acquisition mode and enter **10000** for the rate. Leave all other choices set to their default values. Click "OK" to exit the wizard.
6. Place the Filter Express VI to the right of the DAQ Assistant on the block diagram. From the functions palette, select **Express»Signal Analysis**»**Filter** and place it on the block diagram inside the while loop. When you bring up the functions palette, press the small push pin in the upper left hand corner of the palette. This will tack down the palette so that it doesn't disappear. This step will be omitted in the following exercises, but should be repeated. In the configuration window under Filtering Type, choose "Low pass." Under Cutoff Frequency, use a value of 300 Hz. Click "OK."
7. Now to generate a signal which is to be inputted into analog input terminal place another DAQmx express VI on block diagram and configure it to output signal between 2 and -2.
8. Now use simulate signal express VI to generate a simulated sine wave and connect its output to input of DAQmx express VI.
9. Now create a control for frequency of sine wave of simulated signal express VI.
10. Run the code and wary the frequency terminal till the signal is completely filtered.
11. Close the VI

-End of exercise-

Circuit Design and Analysis with Elvis and Multisim

Exercise 1 – Schematic Capture

Objective

Design and capture a band-pass filter using the 741 op-amp. This exercise has been designed to introduce users to the component browser. By the end of this exercise, you should be able to open the component browser, search and find components and wire a basic circuit in Multisim.

Design

The completed bandpass filter for capture is shown in Figure E1-1. This exercise teaches you how to capture this schematic.

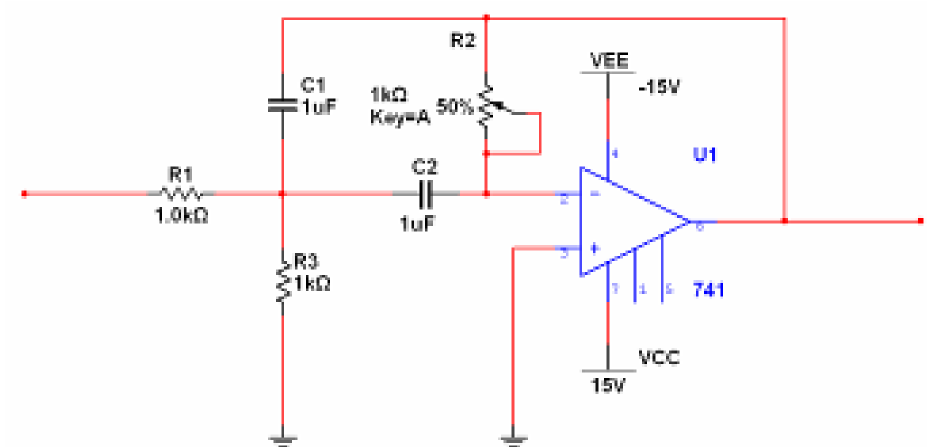


Figure E1-1. Fully Captured Bandpass Filter

Procedure

A. Capture

The first step in the process is to select and place the components for the bandpass filter.

1. Locate and place the 741 op-amp.
   a. Click **Place » Component** from the **Main menu**.
   b. From within the **Component Browser,** click **Search**.
   c. Enter "741" into the **Component** field, and click **Search.**

Multisim then runs a search and returns all the components named 741. In this search, you can also use an asterisk ('*') as a wildcard. For example, the search term "*741" returns the SML4741 component as well as the 741op-amp.

d. Accept the 741 by clicking **OK**. Notice that the **Component Browser** seen in Figure E1-2 now displays the exact location of the 741 (Analog Group, OPAMP family).

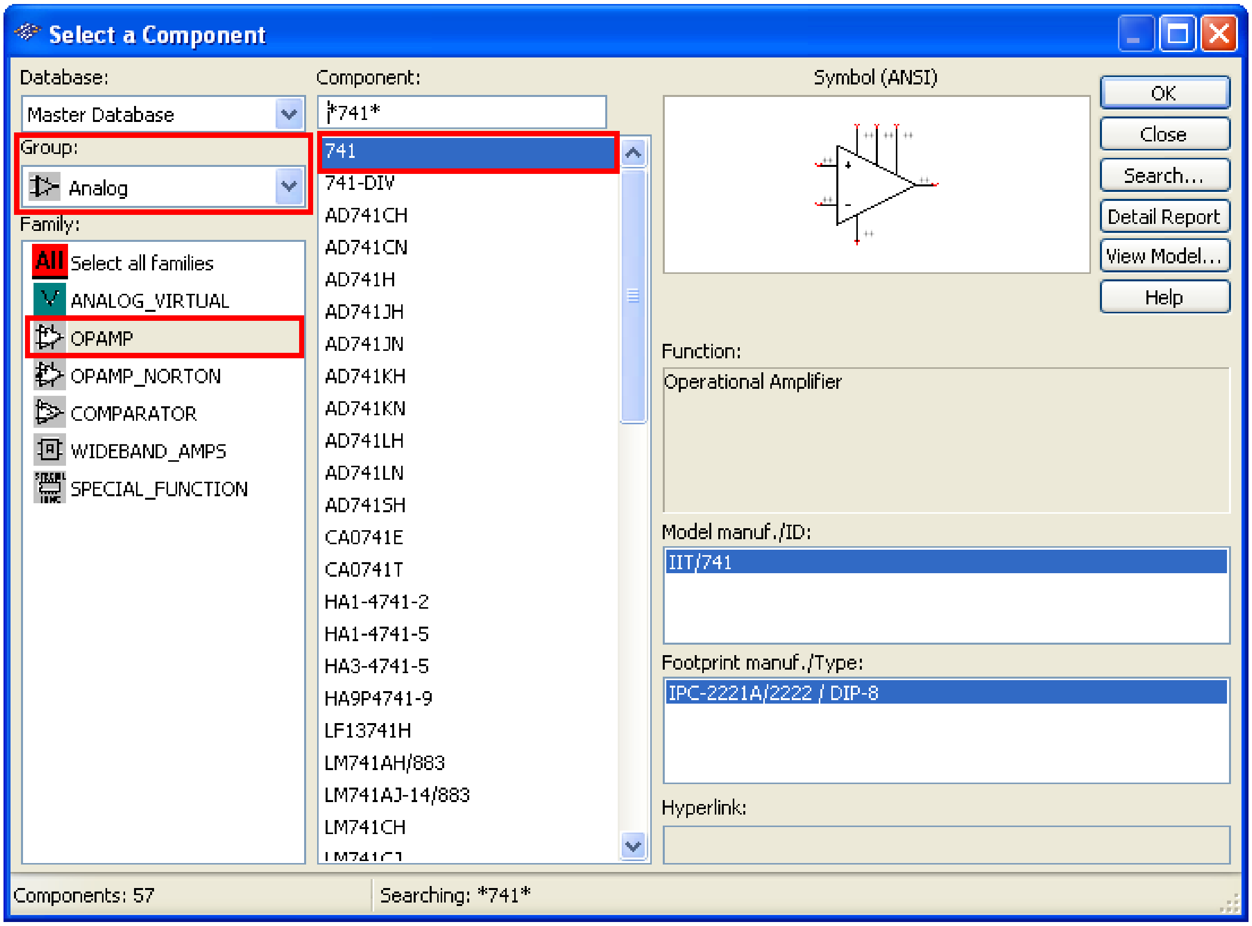


Figure E1-2. Component Browser

e. Click **OK** again, and the op-amp ghosts the mouse cursor. Click anywhere on the schematic to finalize the placement.
f. Click **CLOSE** within the Component Browser window
g. Right-click on the **741** and choose **Flip Vertically** to have the negative input on the top.

2. Place the passive components using the **basic toolbar** shown in Figure E1-3. To make the basic toolbar visible, click **View » Toolbars » Basic**.

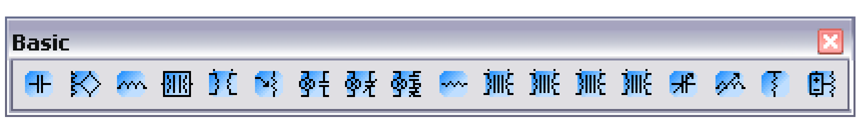


Figure E1-3. Basic Component Toolbar (Direct Placement)

a. Place the resistors R1 and R3, the capacitors C1 and C2, and the potentiometer R2 by clicking once on the corresponding **basic toolbar** icon and clicking again on the schematic. You can **rotate** components by 90 deg by pressing **Ctrl-R** while the component is **ghosting** the mouse cursor.

3. Place the VCC, VEE, and GND components using the power sources toolbar shown in Figure E1-4.

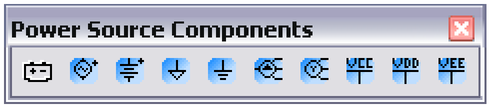
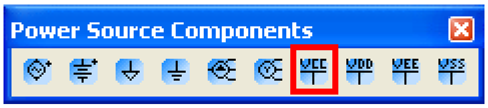
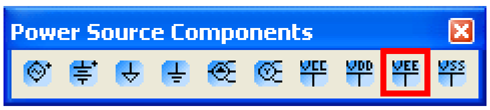


Figure E1-4. Power Source Components Toolbar (Direct Placement)

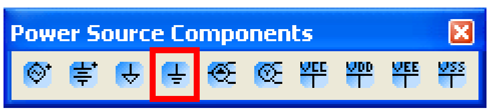
a. Select **View » Toolbars » Basic** to view the **Power Source Components Toolbar**
b. Click on the **VCC** toolbar button and place the **VCC** component on the schematic.



c. Click on the **VEE** toolbar button and place the **VEE** component on the schematic.



d. Click on the **GND** toolbar button and place the **GND** component on the schematic.



e. Double-click on the **VCC** and change its value to **15** from the **Value** tab.
f. Double-click on the **VEE** and change its value to **-15** from the **Value** tab.

Your schematic should look like the diagram in Figure E1-5.

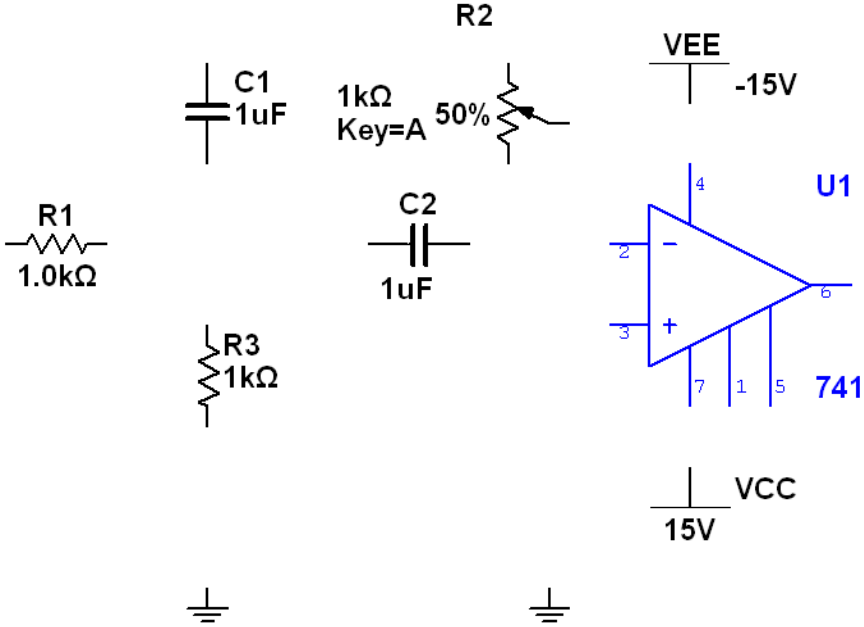


Figure E1-5. Power Source Components Toolbar (Direct Placement)

B. Wiring

The second step in the process is to wire the components previously placed in Section A. To create a wire, simply click on the source terminal and click a second time on the destination terminal. Multisim automatically routes a wire between the two terminals.

1. Connect the **input** of the circuit.
   a. Double-click on a blank area of the schematic worksheet to create a single-ended connection.
   b. Click on the input of R1 to complete the connection.
   c. Right click on the wire and select **Properties**. In the **Net name** field type in **input.**

2. Connect the **output** of the circuit.
   a. Double-click on a blank area of the schematic worksheet to create a single-ended connection.
   b. Click on the output of U1 to complete the connection.
   c. Right click on the wire and select **Properties**. In the **Net name** field type in **output**

3. Complete the remainder of the wiring by clicking source and destination terminals. The resulting circuit should match Figure E1-6.

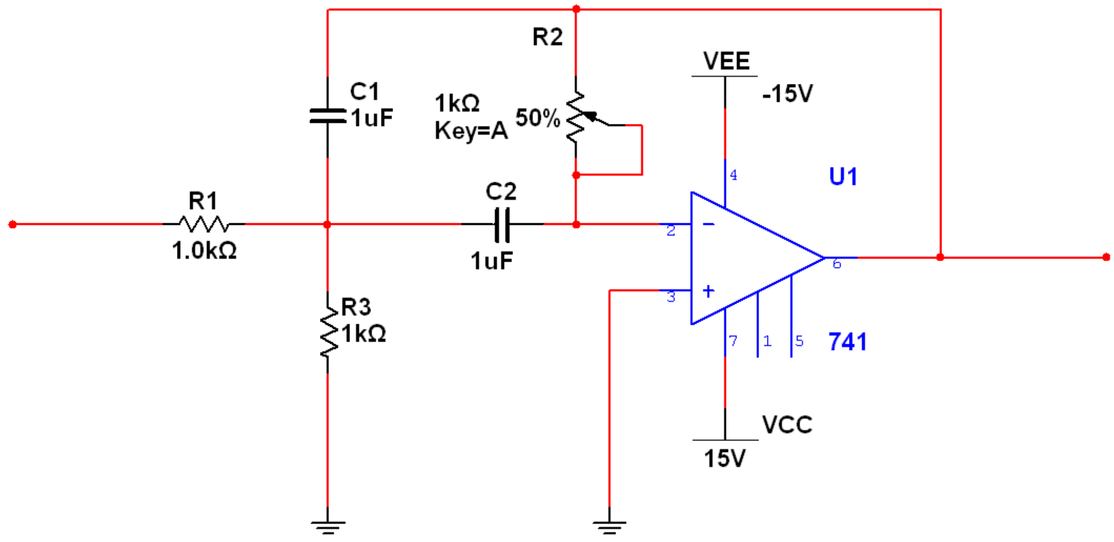


Figure E1-6. Completed Bandpass Schematic

- End of Exercise -

Exercise 2A – Simulating Circuits

Objective

Use interactive simulation to measure the gain, bandwidth, and center frequency of the bandpass filter. Interact with components while simulation is running to see the effects of changing feedback resistance on overall circuit operation.

Design

This exercise uses the function generator, oscilloscope, and Bode plotter to measure the characteristics of the bandpass filter. You can work directly with the completed schematic from Exercise 1 or open “Exercise 1– Bandpass Filter (Complete).ms10” to continue the exercise. The schematic ready for simulation is shown in Figure E2-1.

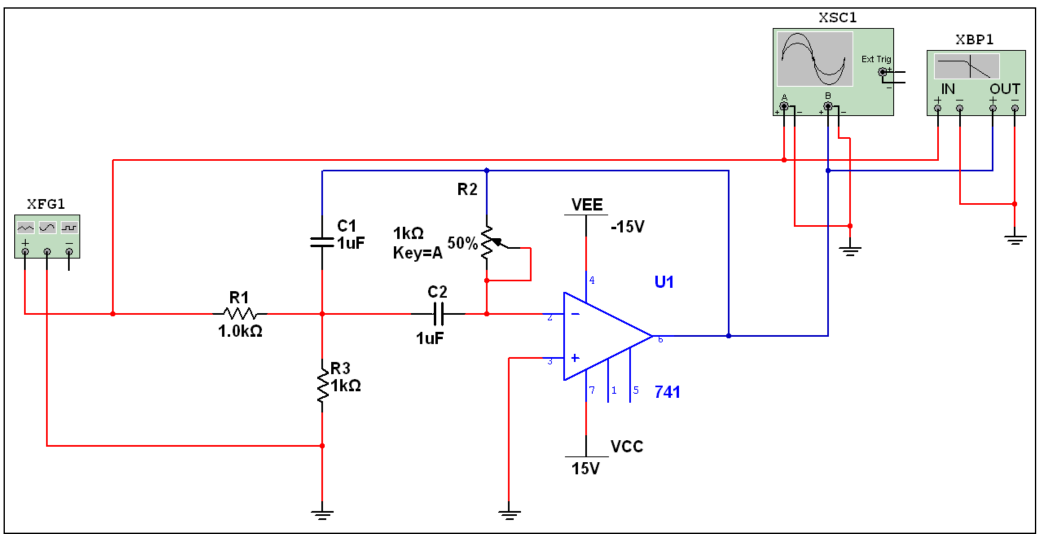


Figure E2-1. Complete Schematic for Exercise 2A

Procedure

1. Place and connect the function generator.

   a. Click first on the **function generator** from the **Instruments toolbar** circled in Figure E2-2 and then click again on the schematic to place the instrument.

Figure E2-2. Instruments Toolbar with Function Generator Highlighted

b. Connect the positive terminal of the function generator to the input of the circuit.
c. Connect the center **reference terminal** to a ground terminal.

Your schematic should look like Figure E2-3.

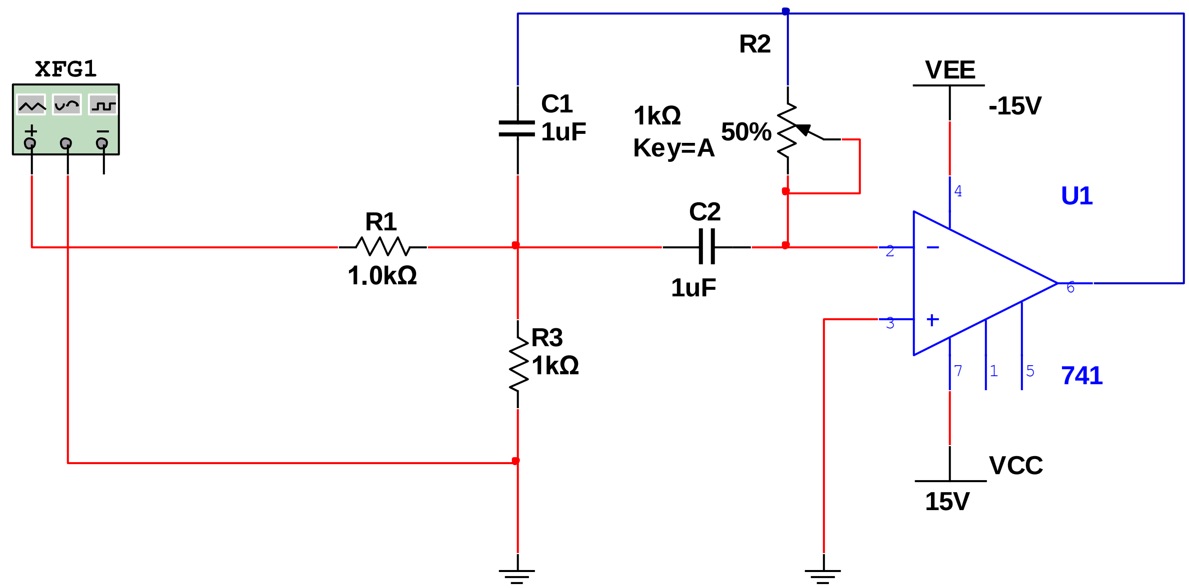


Figure E2-3. Schematic with Function Generator

2. Place and connect the **oscilloscope.**
   a. Click first on the **oscilloscope** from the **Instruments toolbar** circled in Figure E2-4 and then click again on the schematic to place the instrument.

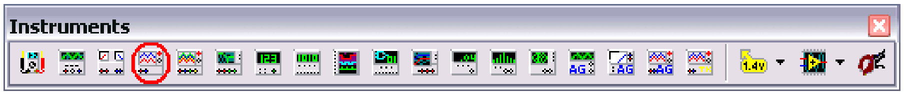


Figure E2-4. Instruments Toolbar with Oscilloscope Highlighted

   b. Connect the negative terminals of channels A and B to ground (place new ground components if needed).
   c. Connect the positive trace of Channel A to the input of the circuit.
   d. Connect the positive trace of Channel B to the output of the circuit.

Now change the color of the wire leading into the positive terminal of Channel B. The color of the wires connected to the terminals of this oscilloscope should match the color of the traces displayed on the oscilloscope's front panel.

e. Right-click on the red wire leading into Channel B+ and choose **Segment Color.** Pick a dark blue from the palette dialog.

Your schematic should look like Figure E2-5.

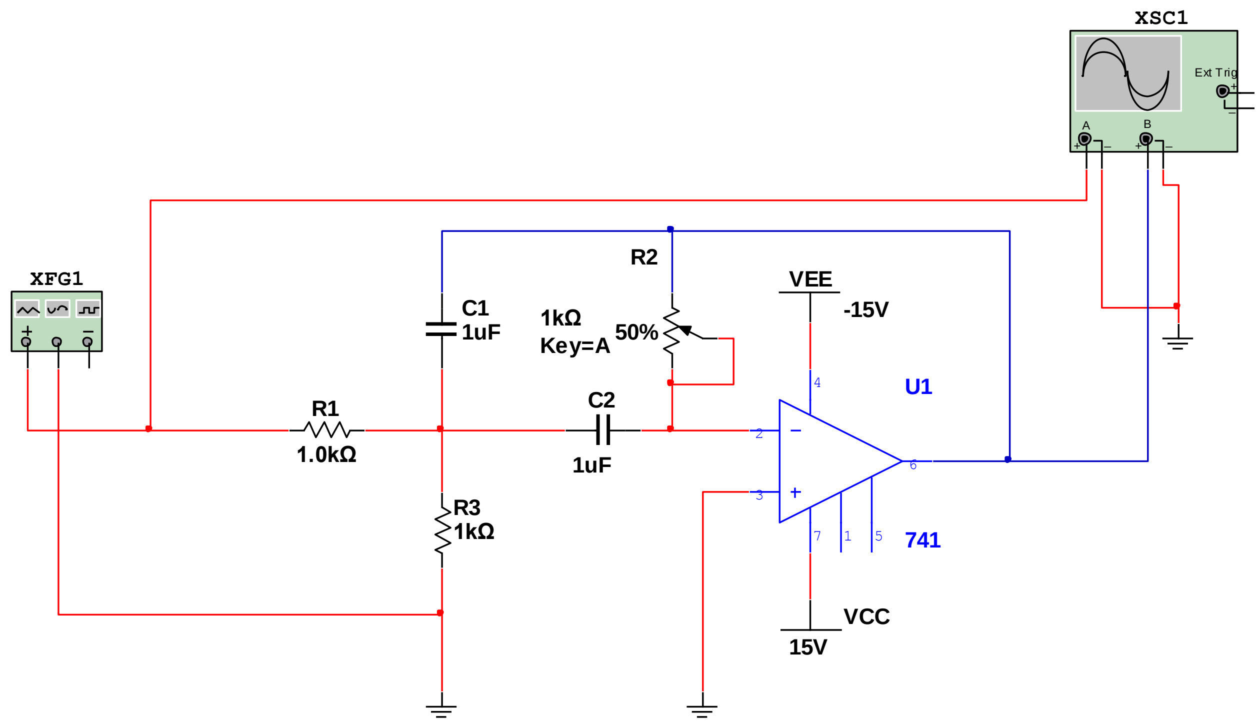


Figure E2-5. Schematic with Oscilloscope Placed

3. Place and connect the **Bode plotter.**
   a. Click first on the **Bode plotter** from the **Instruments toolbar** circled in Figure E2-6 and then click again on the schematic to place the instrument.

Figure E2-6. Instruments Toolbar with Bode Analyzer Highlighted

   a. Connect the negative terminals IN and OUT of the **Bode plotter** to a ground terminal.
   b. Connect the positive IN terminal to the input of the circuit.
   c. Connect the positive OUT terminal to the output of the circuit.

Your circuit is now ready for simulation. It should resemble Figure E2-7.

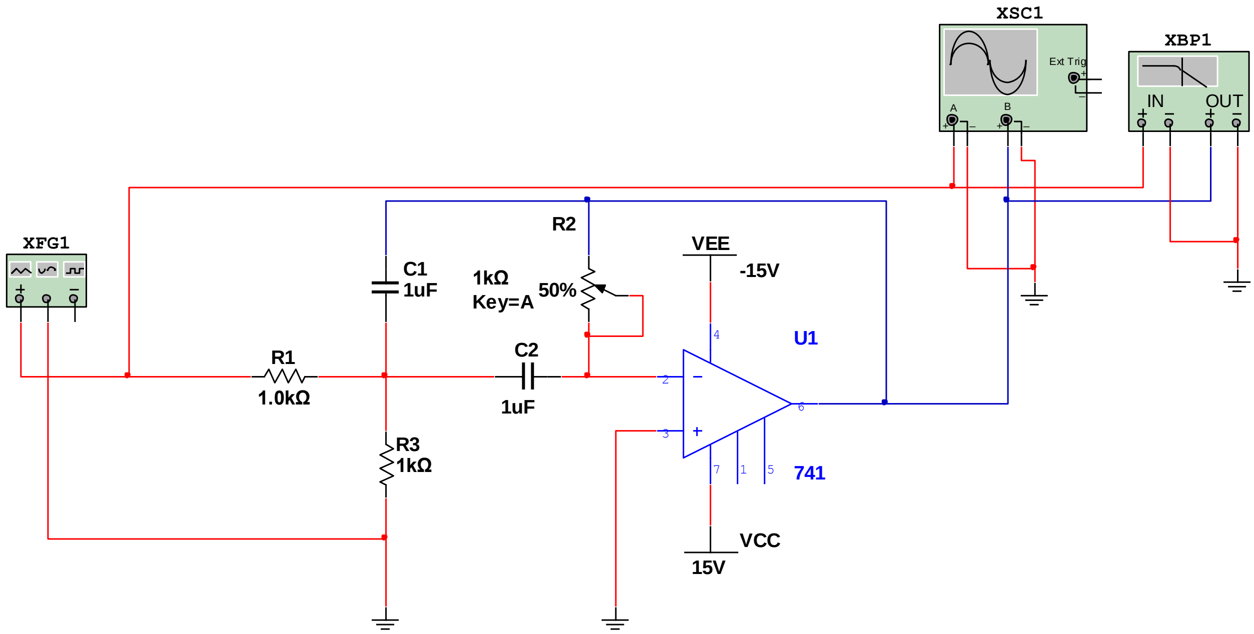


Figure E2-7. Completed Schematic Ready for Simulation

4. Use the **function generator** and **oscilloscope** to simulate the circuit and explore its transient characteristics.

   a. Start the simulation by clicking the **Simulate button** from the **Simulation toolbar** circled in Figure E2-8.

Figure E2-8. Simulation Toolbar with Simulate Button Circled

   b. Open the **function generator** and **oscilloscope** front panels by double-clicking on their schematic symbols. Click the **Reverse** button on the **oscilloscope** to turn its background color white, making it easier to see the results.

   c. Verify your circuit is operating as expected by configuring the **function generator** and **oscilloscope** as shown in Figures E2-9 and E2-10.

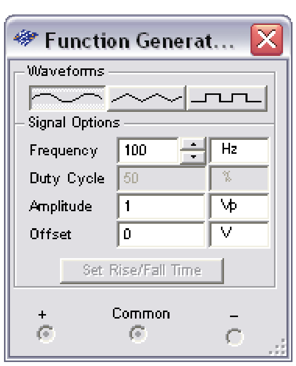


Figure E2-9. Function Generator Configuration

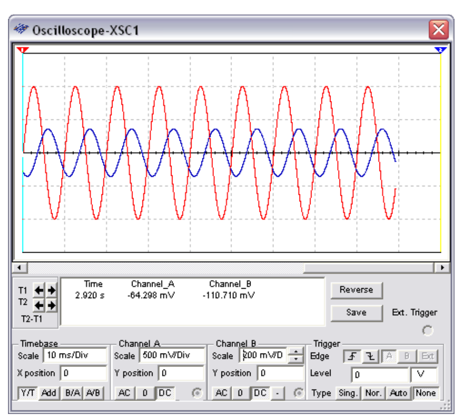


Figure E2-10. Oscilloscope Configuration

d. While the circuit is simulating, investigate how changing the resistance of R2 affects the overall gain and phase shift of the circuit. To change the resistance of the potentiometer, hover the mouse over the component and click and drag the slider.

e. Drag the slider to 100 percent, which represents the full 1 kΩ resistance of the component.

5. Using the **Bode analyzer**, measure the frequency characteristics of the circuit.

a. Stop simulation by clicking the **Stop button** from the **Simulation toolbar.**

b. Start the simulation to rerun the AC analysis to reflect the new value of 100 percent for the potentiometer R2.

c. Open the **Bode analyzer** front panel by double-clicking on it. Click the **Reverse** button to turn the background color white. The **Bode plotter** front panel should look similar to Figure E2-11.

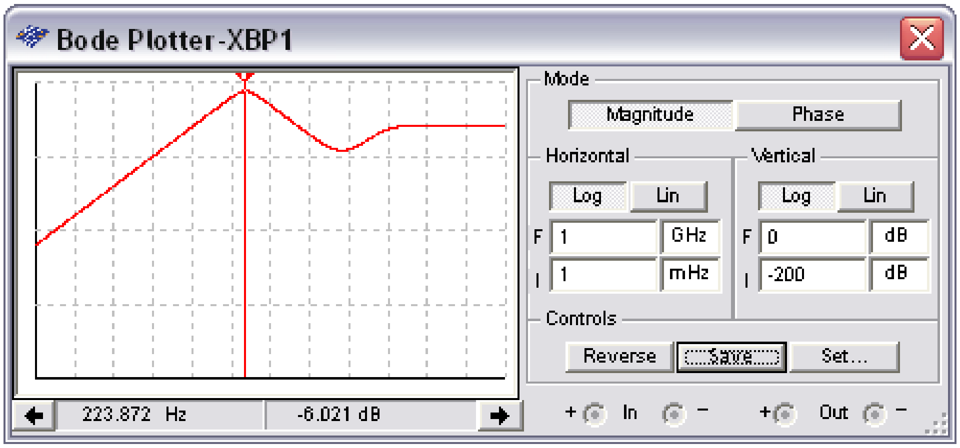


Figure E2-11. Bode Plotter Showing Center Frequency Measurement

Now measure the center frequency and gain.

d. Right-click on the cursor and choose **Go To Next Y_MAX =>.** This is the center frequency. If your center frequency is different, verify that the potentiometer is set to 100 percent and stop and restart the simulation.

Expected center frequency: **223.9 Hz**
Expected overall gain: **-6.0 dB**

Measure the -3 dB points of the circuit. The -3 dB points are the frequencies at which the gain is 3 dB below its maximum value. In this case, the -3 dB points are the frequencies where the gain of the circuit is -9 dB.

e. Drag the cursor on the front panel of the **Bode plotter** to measure the two -3 dB points of the circuit. You can also right-click on the cursor and, depending on its current location, choose **Set Y_Value =>** or **Set Y_Value <=.** Enter -9 as the value and select OK.

Expected lower -3 dB point: 117 Hz
Expected upper -3 dB point: 433 Hz

You can save the simulation results generated in this section for comparison with measurements by clicking the **Save** button from each front panel.

- End of Exercise -

Exercise 2B – SPICE Analyses

Objective

Having completed interactive simulation, we are able to utilize SPICE analyses to measure the center frequency of the bandpass filter while learning how to effectively use the grapher tool to analyze simulation data.

Design

This exercise continues to use the same bandpass filter completed in the previous exercises. You can open the **Exercise 2b - Bandpass Filter (Complete).ms10** file to continue this exercise (as seen in Figure E2-12)

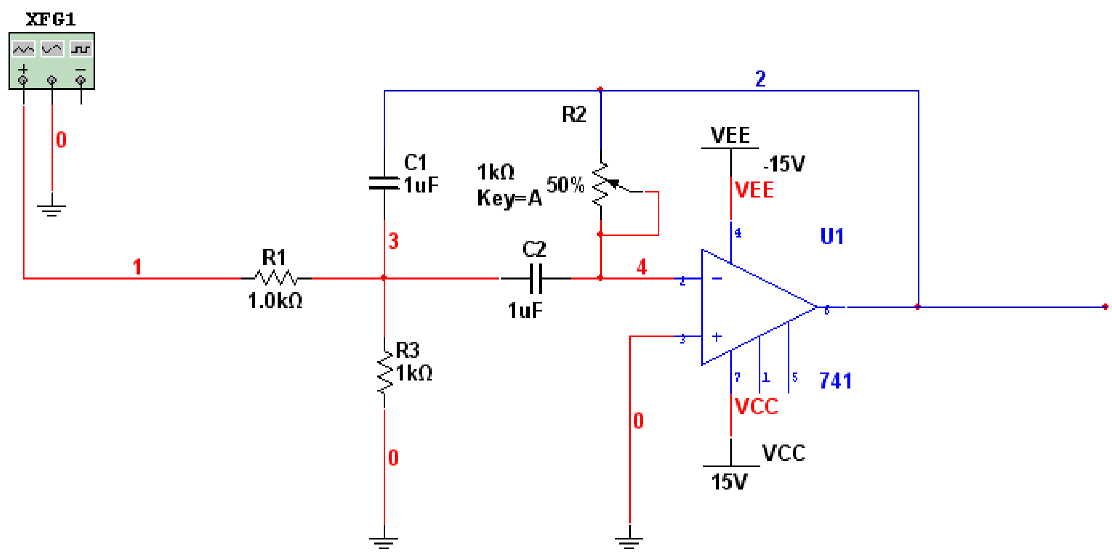


Figure E2-12. Bandpass Filter

Procedure

1. Select **Simulate » Analyses.** You will notice the various analyses that are available in NI Multisim.
2. Select **AC Analysis…**
   The AC Analysis dialog box contains 4 tabs:

   - *Frequency Parameters*
     This tab allows you to set-up the start and stop frequencies, simulation sweep type and the vertical scale units
   - *Output*
     Sets the nodes and nets at which a simulation will be performed
   - *Analysis Options*
     Sets the SPICE simulation parameters such as the custom settings for integration, error tolerance etc…

- *Summary*
  A list of the AC Analysis simulation settings chosen in the previous 3 tabs. Select the *Frequency Parameters* tab and set the following values (as seen in Figure E2-13):

Start Frequency (FSTART): **1 Hz**

Stop Frequency (FSTOP): **1 GHz**

Sweep Type: **Decade**

Number of points per decade: **100**

Vertical Scale: **Decibel**

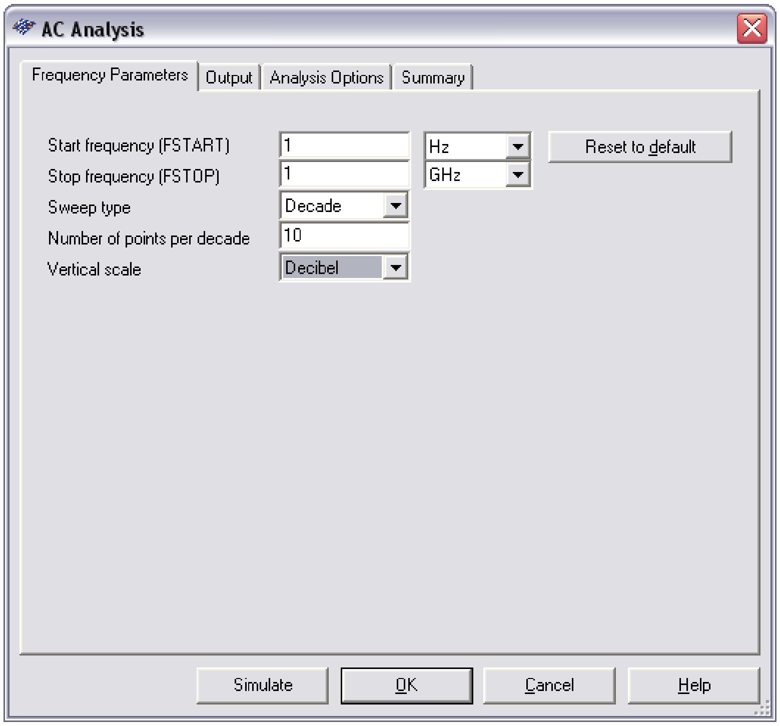


Figure E2-13. AC Analysis Frequency Parameters

3. Select the Output tab and set up the SPICE simulation to perform an AC Analysis at the output of the band-pass filter (as seen in Figure E2-14)
4. In the right-hand side of the analysis you will notice the *Variables in circuit* field. In this field you will see a list of all of nets in the design at which a simulated voltage value can be measured. These variables are listed with their names. Notice the variable **V(output)** which represents the voltage at the output of our band-pass filter.
5. Select the **V(output)** variable.

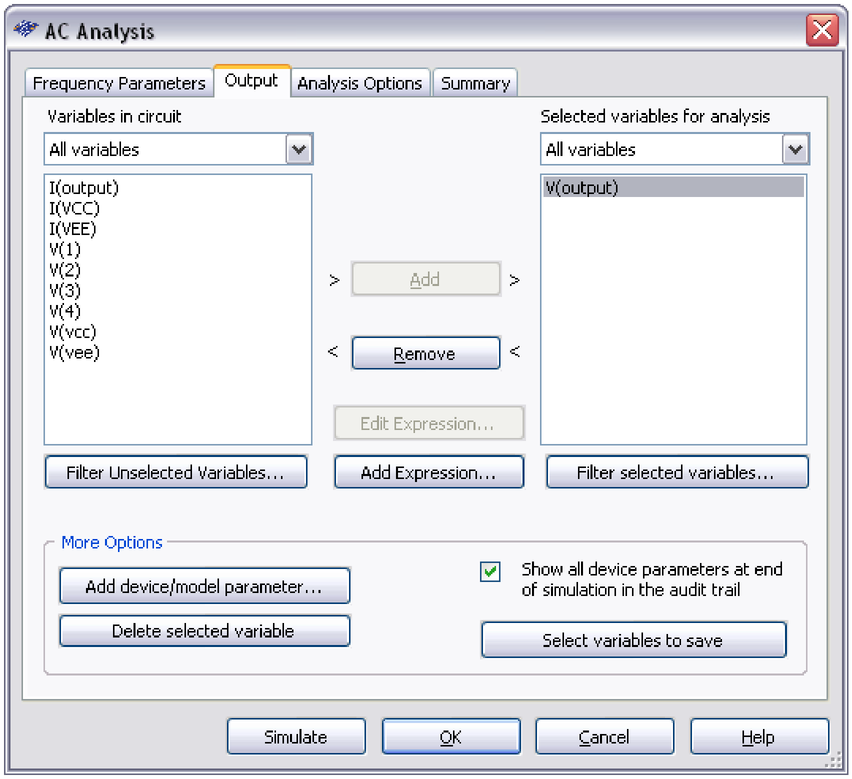


Figure E2-14. AC Analysis Output Tab

6. Click on the **Add** button. Notice that **V(ouput)** is now a *selected variable for analysis*. We are ready to simulate
7. Click on the **Simulate** button.
8. The Grapher View is now open. If you see the black background and would like to reverse the color of the background to white, click on the reverse button (as seen in Figure E2-15).

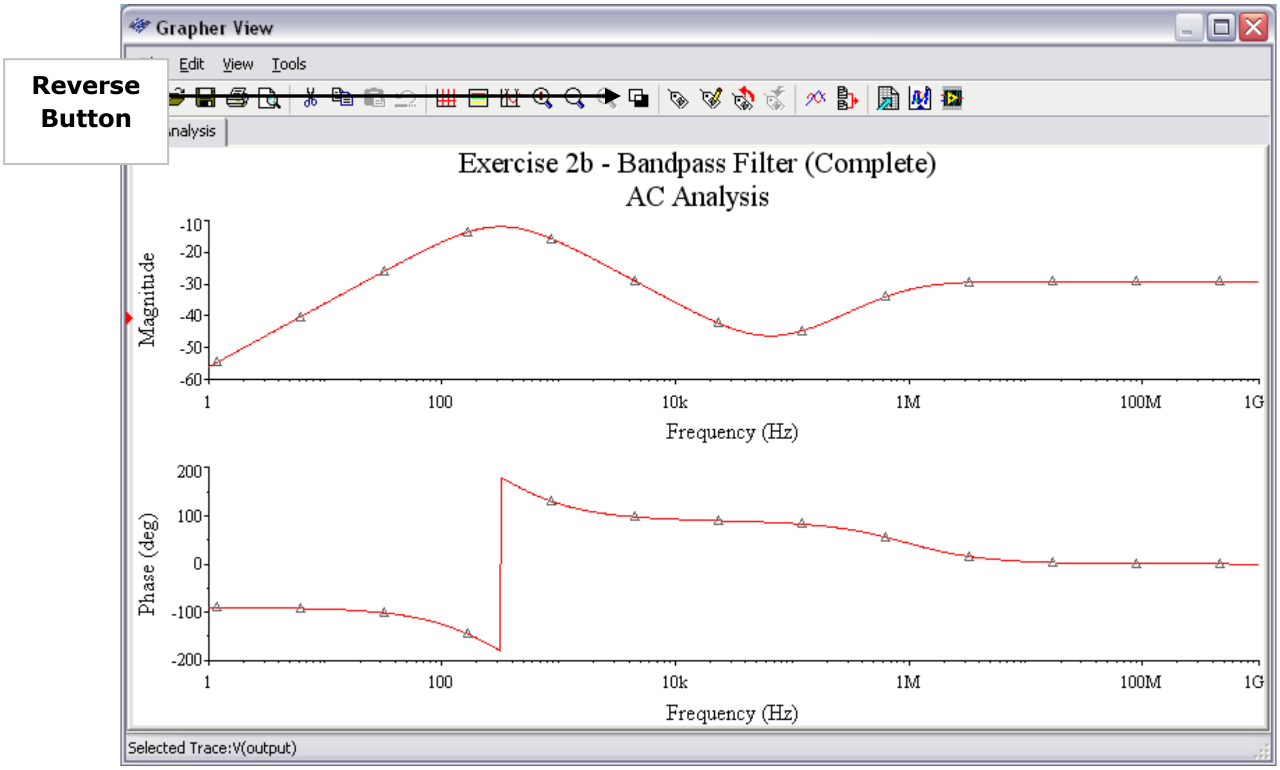


Figure E2-15. AC Analysis of Bandpass Filter

Using the Measurement Probe

We will learn now about setting up some of the advanced features of the Multisim SPICE simulation environment as well as of using the Grapher interface.

Before we return to our analysis we will discover a way in which to better identify a node of interest to measure in our simulation. Recall that previously in the analysis we selected our node of interest by using its net name, **V(output)**, which in a large design, may be difficult to find.

In this exercise we will use a measurement probe to name and identify the node we will eventually study.

A measurement probe can be used both to identify a node in an analysis as well as provide instantaneous measurement data during an interactive simulation.

1. In the instruments toolbar select the measurement probe (as seen in Figure E2-16).

Figure E2-16. Instruments toolbar with measurement probe

2. The measurement probe will now be ghosted to your mouse. Left-click the measurement probe to the output of the filter (as seen in Figure E2-17).

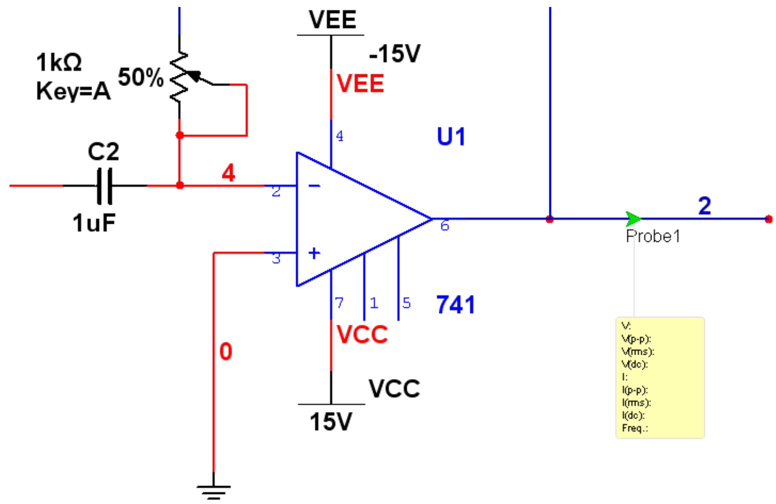


Figure E2-17. Measurement Probe at bandpass filter output

3. Click on the (interactive**) simulate button** or press **F5**. You will notice that during an interactive simulation you will be able to instantaneously access the **voltage**, **current** and **frequency** information of that node through the measurement probe (as seen in Figure E2-18).

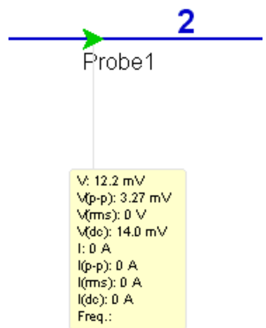


Figure E2-18. Measurement Probe during an interactive simulation

4. Stop the interactive simulation session.

Notice that the current RefDes of the probe is **Probe 1.** This name or RefDes can be changed to something which means more in terms of your design or simulation.

5. Double-click on your probe (Probe 1).
6. In the **Probe Properties** dialog box set the **RefDes** to **output** (as seen in Figure E2-19).
7. Click on the **OK** button.

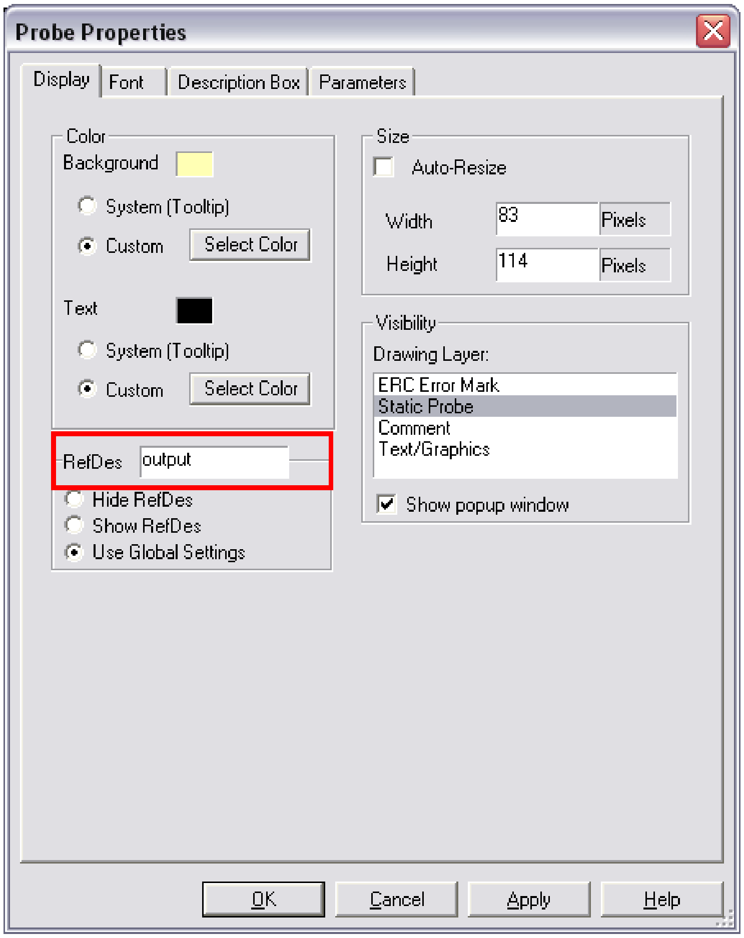


Figure E2-19. Measurement Probe Properties

We are now ready to return to our analyses.

8. Select **Simulate » Analyses » AC Analysis**.
9. Make sure that your **frequency parameter** tab is set-up as in step 3.
10. Click on the **Output** tab.

You will notice that in the *Variables in Circuit*, we can only see the static probe voltages, specifically our node **output**.

11. Select **V(output)** and click on the **Add** button.
12. Click on the **Simulate** button. The grapher again displays the bandpass filter characteristics according to an AC Analysis.

Working with the Grapher View

We will now begin working with the Grapher View to make analysis easier.

We will be using the three highlighed icons in the Grapher View menu as seen in Figure E2-20.

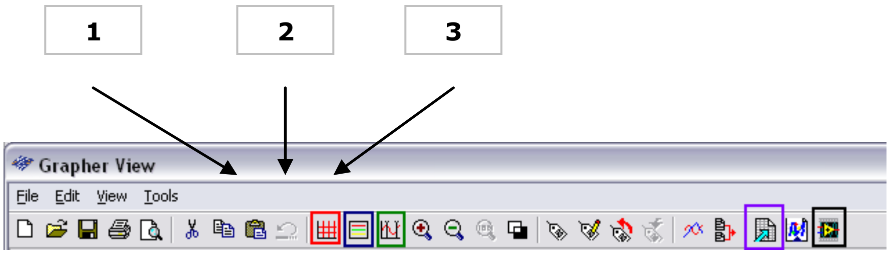


Figure E2-20. Grapher View menu toolbar

1. To show the grid on the Grapher, left-click on the magnitude plot of the AC Analysis and then click on the **Show/Hide Grid** button (Figure E2-20-1).
2. Left-Click on the phase plot of the AC Analysis and click on the **Show/Hide Grid** button (Figure E2-20-1). The Grapher view will now look like figure E2-21.

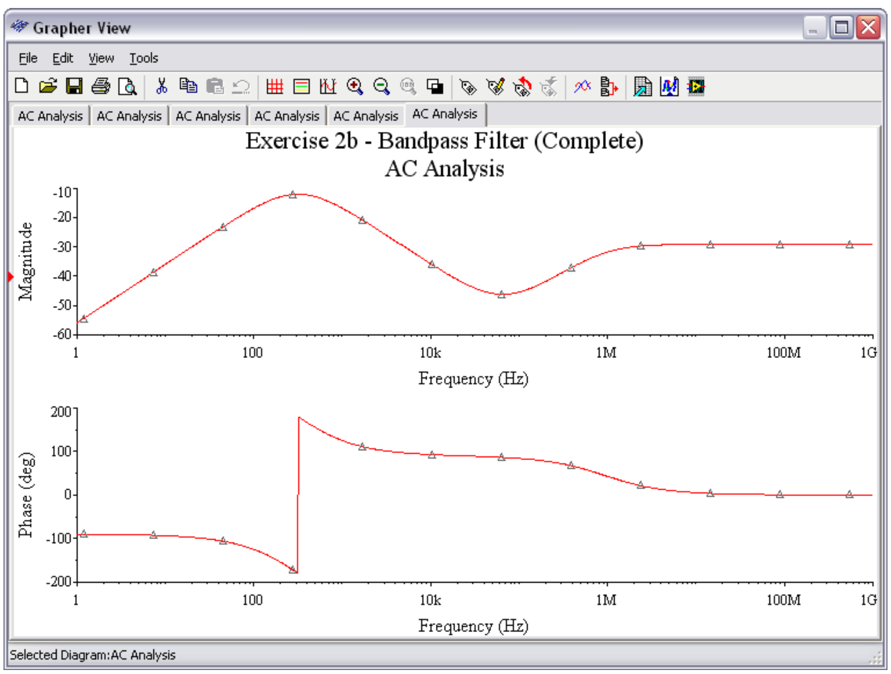


Figure E2-21. Grapher View with grid

3. Left-click on the magnitude plot. Now click on the **Show/Hide Legend button** (Figure E2-20-2).The AC Analysis legend for the magnitude plot will appear.
4. Click on the **Show/Hide Cursors** button (Figure E2-20-3).The value box will appear, displaying the various x and y co-ordinate values as measured by the cursors (Figure E2-22).

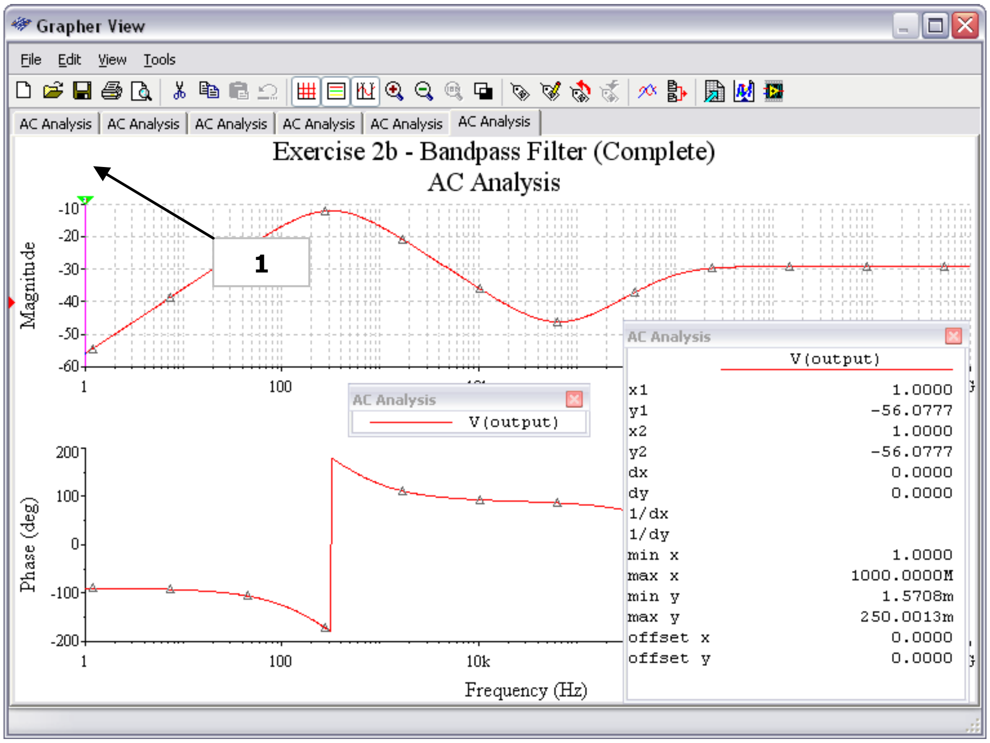


Figure E2-22. Grapher View with grid, legend and cursors

5. Click on the neon-green colored cursor (Figure E2-22-1).Move the cursor left and right across the screen and notice the values in the AC Analysis measurement box change at each location.

We can now find the center-frequency of our plot. **Note** that in this exercise our potentiometer is set at **50%** and **not 100%** as we had seen in a previous lesson. So we should expect a different center frequency and gain from exercise 2.

6. Right-click on the green cursor and select **Go to next Y_MAX »**
7. Look at the AC Analysis cursor value and you will see that the center frequency, denoted as `x1` equals `316.3378 Hz` and the gain denoted as `y1` equals `-12.0412 dB`.

It is often very useful to compare different analyses you have performed to better understand a design decision. For example in lesson 2 we had set our potentiometer to 1k Ω and in this exercise it remained at 500 Ω. Let's take a look at the difference between both of these analyses by overlaying the results on top of each other on a single set of axes.

8. Close the Grapher View.
9. Open the file **Exercise 2 – Bandpass Filter (Complete).ms10** (recall this is the file that was used in exercise 2)
10. With **Exercise 2 – Bandpass Filter (Complete).ms10** open, set the potentiometer (R2) to 100% of its value (1 k Ω)
11. Double-click on the Bode Plotter instrument
12. Set the **Horizontal** initial (**I**) frequency to **1 Hz**.
13. Click on the simulate button and view the frequency response of your filter in the Bode Plotter.
14. Stop the simulation.
15. Close the Bode Plotter instrument and select **View»Grapher**.
16. In the Grapher View select the **Bode Plotter-XBP1** tab (Figure E2-23)

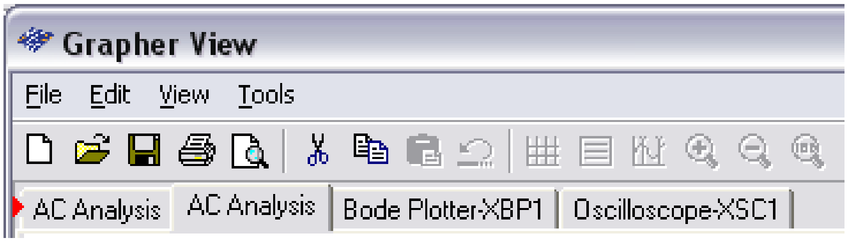


Figure E2-23. Grapher View graph tabs

17. Select the magnitude plot in the Bode Plotter graph.
18. Select **Tools»Overlay Traces**
19. In the **Select a Graph** dialog box select Graph 1 (which is the magnitude plot) in the last AC Analysis that you completed. In this case (as seen in Figure E2-24) we select **Page_2:Graph_1[Tab_Name:AC Analysis,Graph_Title:AC Analysis].**
20. Click on the **OK** button

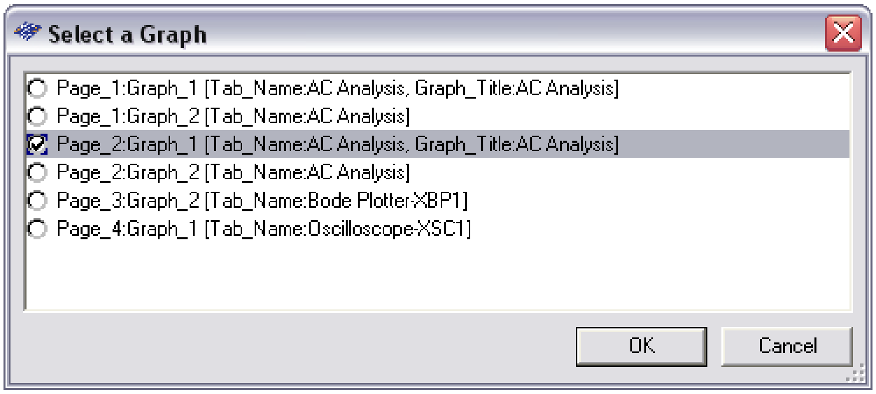


Figure E2-24. Overlaying AC Analysis magnitude over Bode Plot simulation

Multisim will merge the magnitude plot from the Bode Plot and the magnitude plot from the AC Analysis onto one set of axes (as seen in Figure E2-25)

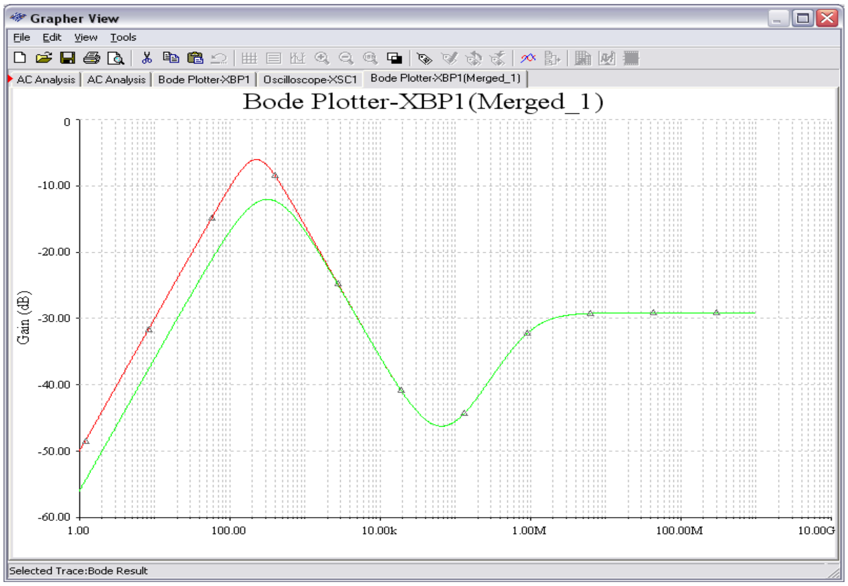


Figure E2-25. Merged analysis of Bode Plot and AC Analysis

- End of Exercise -

Exercise 3 – 3D Virtual Prototyping

Objective

Use the 3D prototyping environment to do virtual prototyping of the designed band-pass filter circuit.

Design

Multisim has a 3D NI ELVIS and NI ELVIS II prototyping environment. First, open the filter circuit from the previous exercises developed using the NI ELVIS II template. Then switch to the 3D breadboard view and prototype the circuit using NI ELVIS II.

Procedure

1. Open "**Exercise 3 – Bandpass Filter NI ELVISII.ms10.**"

The schematic that opens is the same circuit from Exercise 2, except it has been created using the NI ELVIS II schematic.

2. From the Main toolbar, click the **Breadboard** icon to open the 3D NI ELVIS prototyping environment. The icon is shown in Figure E3-1, and the 3D environment is depicted in Figure E3-2.

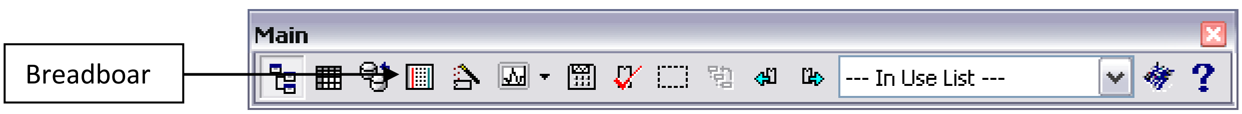


Figure E3-1. Breadboard Icon

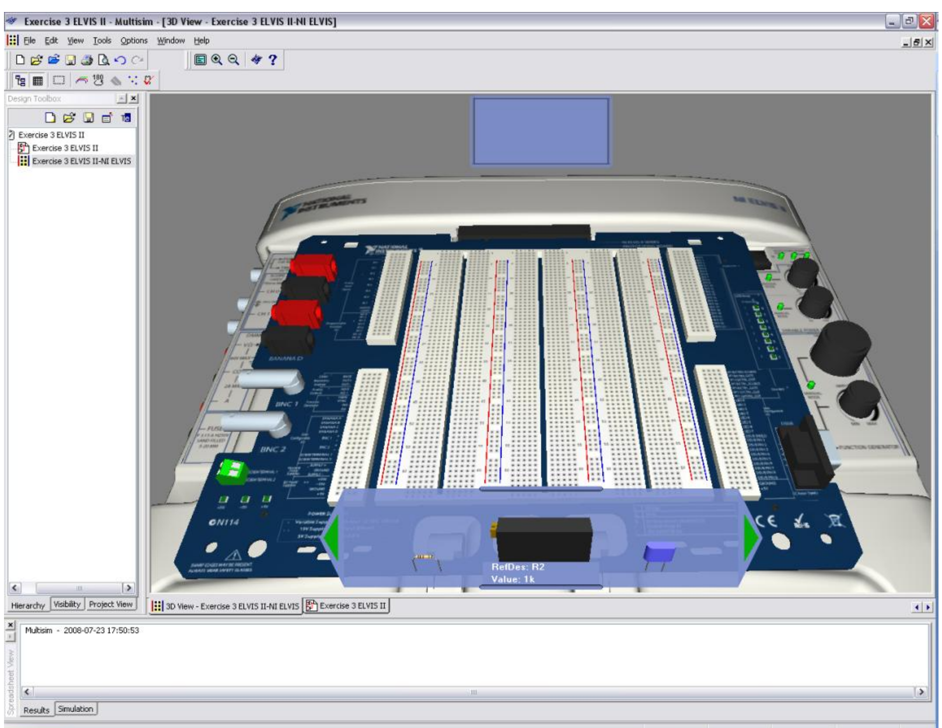


Figure E3-2. 3D NI ELVIS II Prototyping Environment

4. Place components onto the 3D breadboard by dragging them from the parts bin at the bottom of the display. Connect pins together with wires by clicking first on a source hole and then on a destination hole. Using the techniques described, prototype the bandpass filter circuit.

4. Verify the completion of your prototype.

   a. Method 1 – Click on the DRC button from within the 3D view breadboard. The DRC toolbar button is labeled 5 in Figure E3-4.

Figure E3-4. Main Toolbar in 3D Environment

   b. Method 2 – Switch from the 3D breadboard view to the schematic. All nodes and components should turn green to indicate that the prototype is complete.

After completing the virtual protyping exercise, now the design is ready to be prototyped on the NI ELVIS II hardware workstation for measurement and test.

5. Build the physical circuit on the NI ELVIS II design and prototyping hardware platform.
6. Investigate filter behavior using ELVISmx virtual instruments.
   a. Modify the earlier filter circuit by wiring in the ELVISmx Oscilloscope. Wire the positive terminal of Channel 0 of the oscilloscope to the AI1 + pin of the Analog Input Signals portion of the NI ELVIS II virtual schematic.

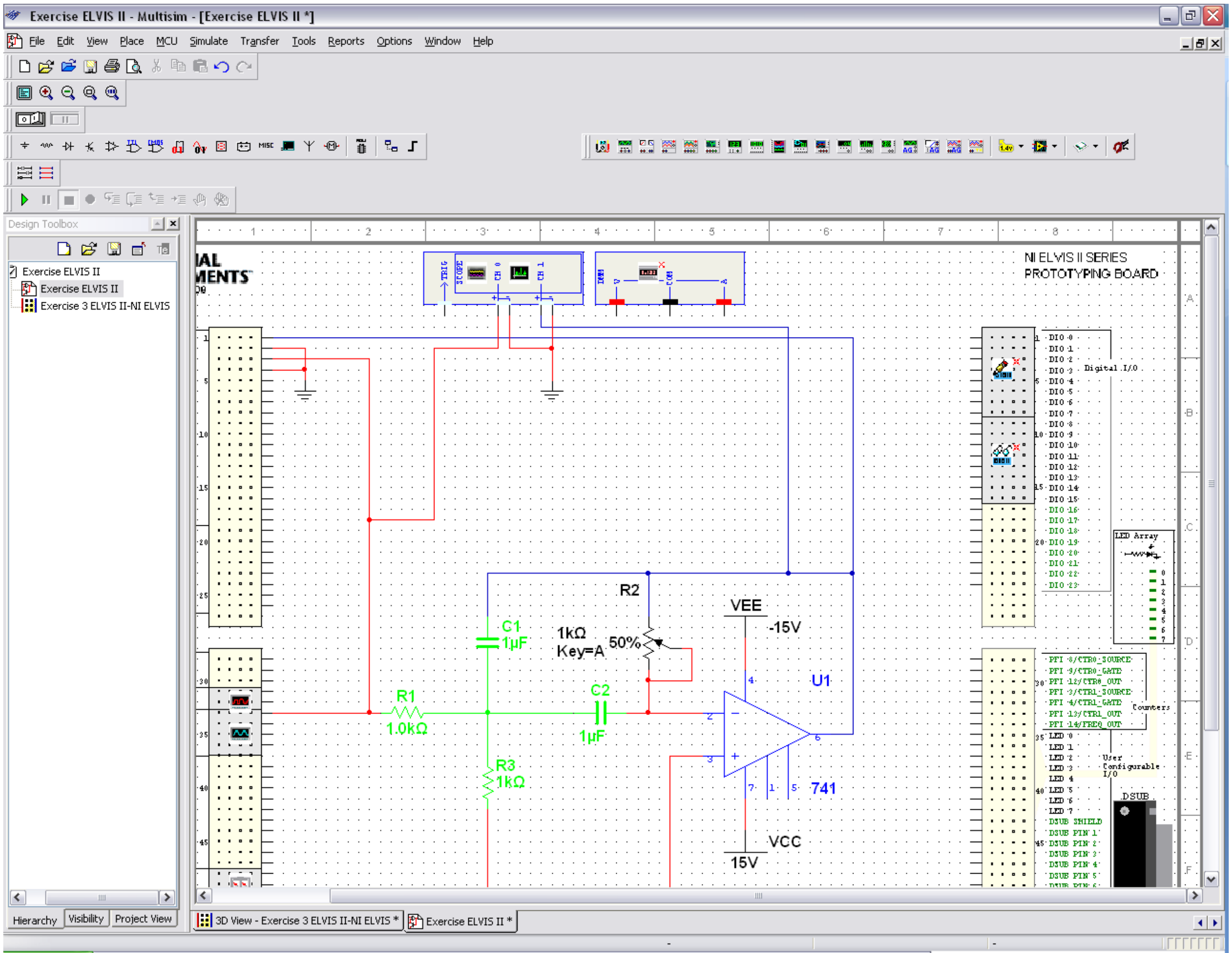


Figure E3-5. ELVISmx Virtual Oscilloscope

b. Wire the positive terminal of Channel 1 of the oscilloscope to the AI0 + pin of the Analog Input Signals portion of the NI ELVIS II virtual schematic, which is monitoring the output of the 741 op-amp.
c. Connect a ground to the negative terminals of CH0 and CH1 of the oscilloscope.
d. Double click the function generator to bring up the function generator soft front panel. The function generator icon is shown in Figure E3-6.

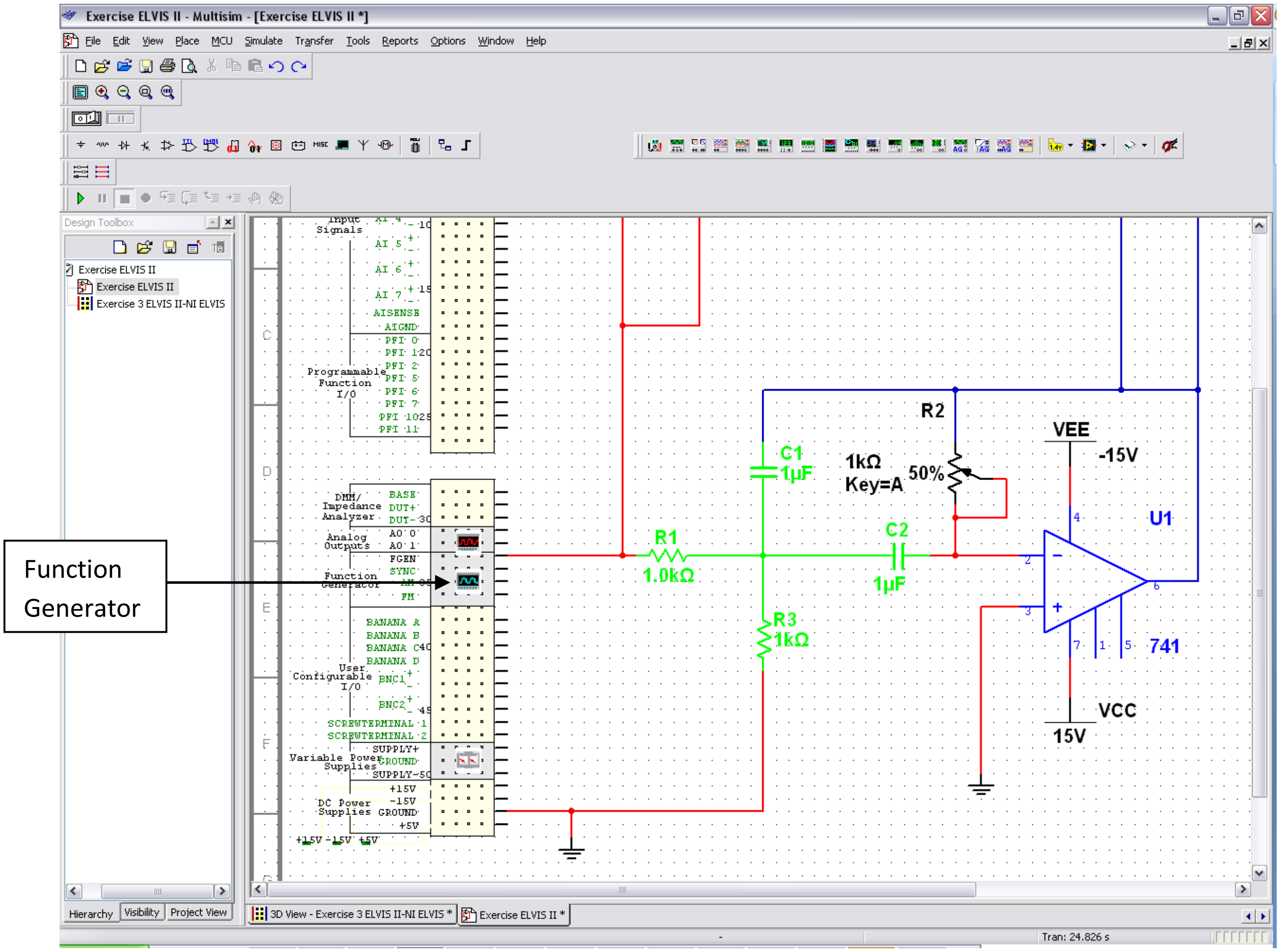


Figure E3-6. ELVISmx Virtual Oscilloscope

e. The function generator soft front panel should pop up, it should look similar to Figure E3-7 below. Change the Frequency to 230 Hz, the amplitude to 1V, and make sure the function generator is generating a sin wave.

Waveform:	**Sine Wave**
Frequency:	**230Hz**
Amplitude:	**1Vp-p**

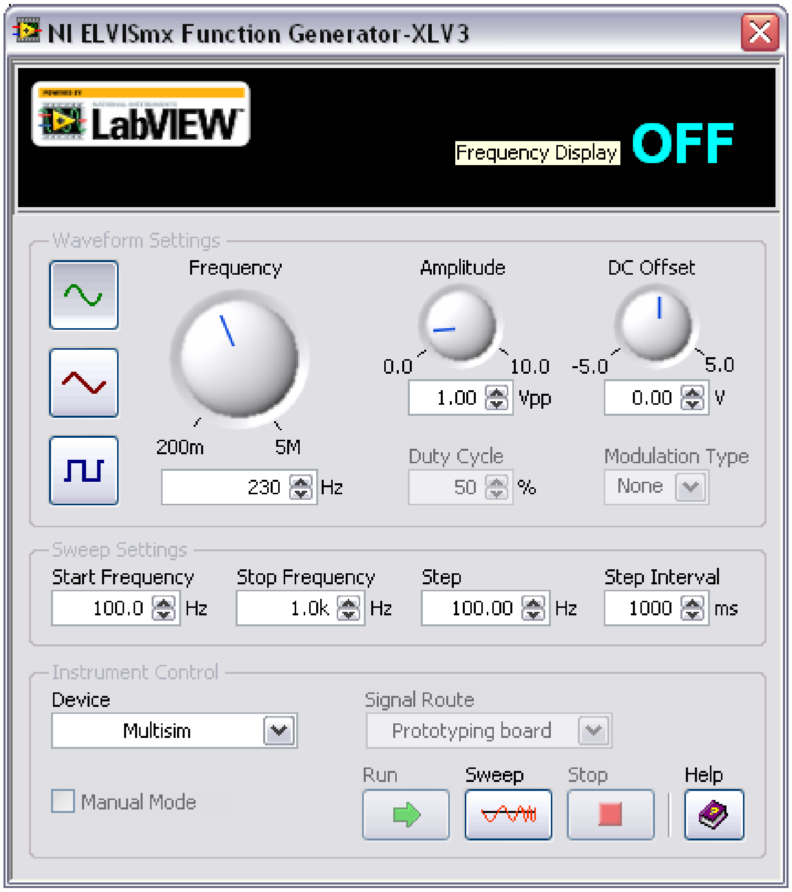


Figure E3-7. Function Generator Soft Front Panel

f. Notice that under the Instrument Control portion of the soft front panel, the Device that is selected is "**Multisim**." Multisim 10.1 allow users to compare simulated data of the captured schematic and compare it to measured data from the circuit prototyped on the the NI ELVIS II protoboard.
g. Close the **Function Generator** Soft Front Panel after configuring the settings.
h. Double click the **oscilloscope** to bring up the oscilloscope soft front panel. The oscilloscope icon is shown in Figure E3-8.

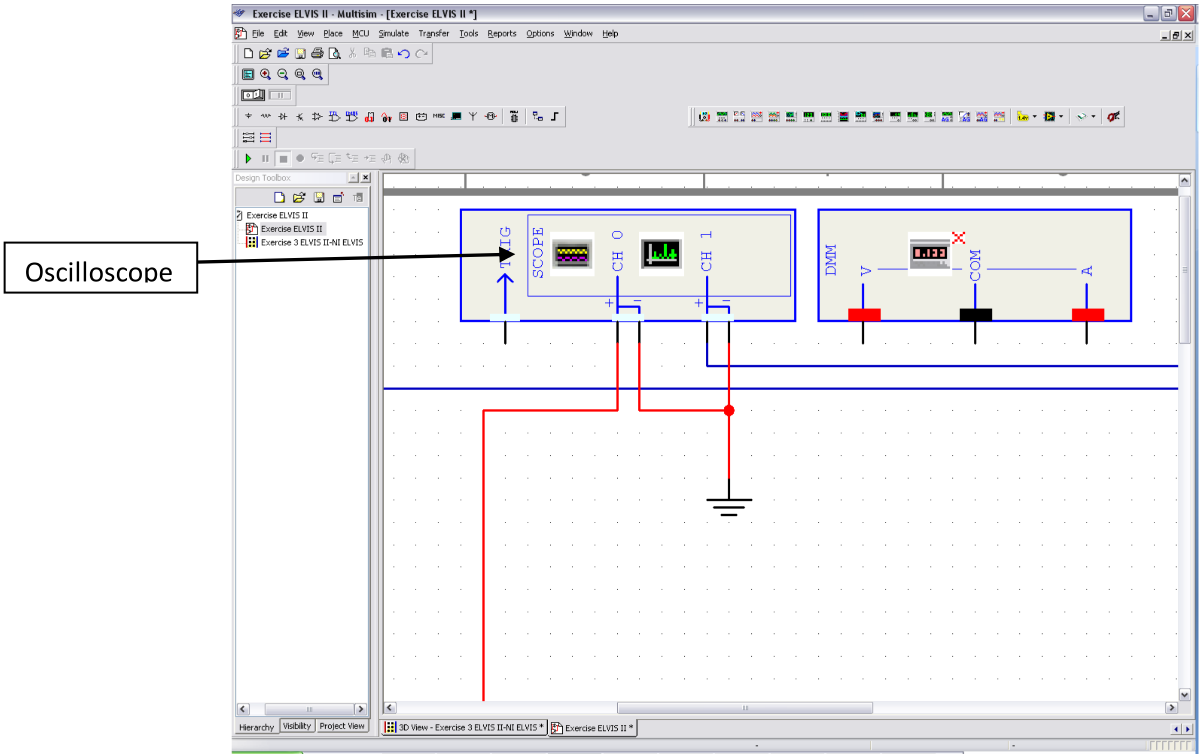


Figure E3-8. ELVISmx Oscilloscope Icon

i. The soft front panel should look similar to Figure E3-9 below. The soft front panel parameters of the oscilloscope can be adjusted on the fly while the schematic is being simulated or during actual hardware testing. Make sure the Channel 1 is enabled by clicking the enabled check box.

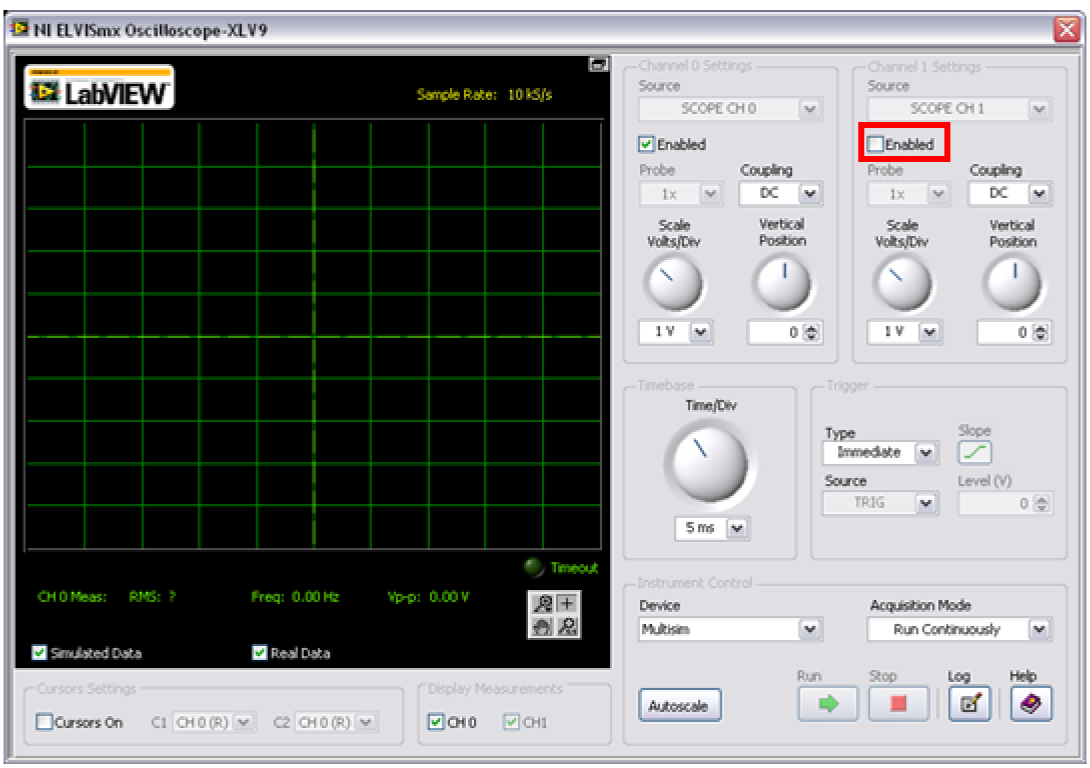


Figure E3-9. ELVISmx Oscilloscope Soft Front Panel

j. Now, click the simulate button to simulate your circuit. The oscilloscope should process the schematic and output a waveform similar to the one shown in Figure E3-10 below. This is the simulated output of the circuit that you have designed within Multisim.

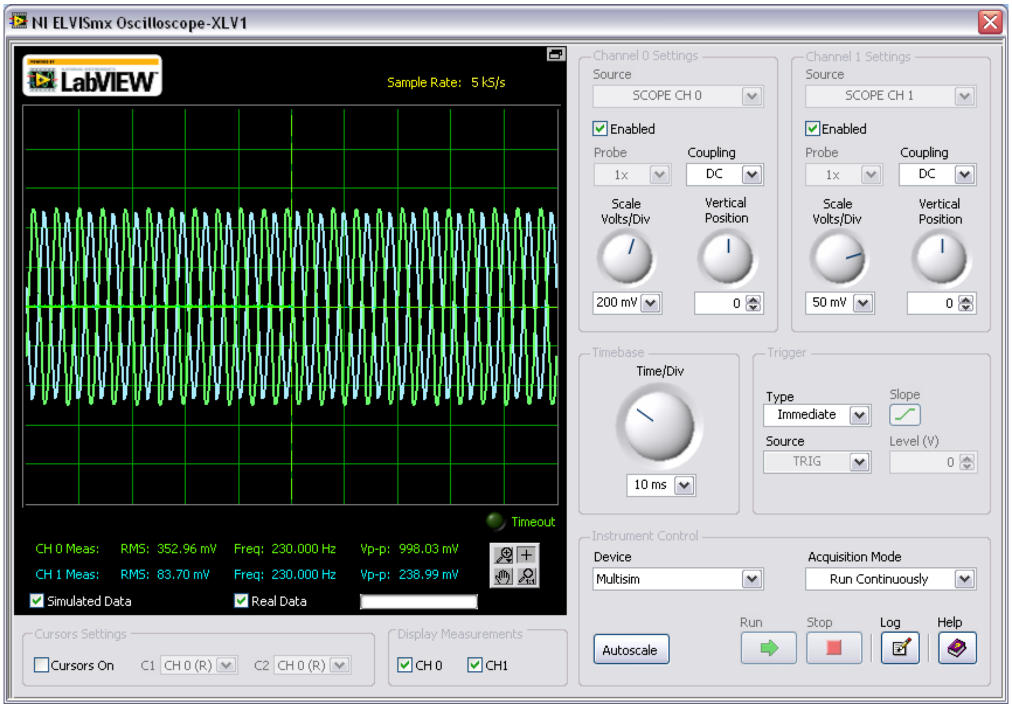


Figure E3-11. ELVISmx Oscilloscope Soft Front Panel

k. If you have already built your actual circuit on the NI ELVIS II protoboard, then we can see the output of this circuit from within the scope front panel as well. In order to do this, stop the simulation by clicking the stop simulation button.

l. Under the scope soft front panel, click the Device drop down menu and change the Device to **NI ELVIS II**. Perform the same step for the function generator as well.

m. Click the **Run** button. Ensure that the **"Simulated Data**" and the "**Real Data**" checkboxes are selected.

n. Observe the hardware output as well as the simulated output of your circuit overlaid on each other. This allows you to compare the output of your hardware implementation to your schematic design within Multisim. After the device is set to acquire real data from the NI ELVIS II, you should see output similar to the Figure E3-12 below.

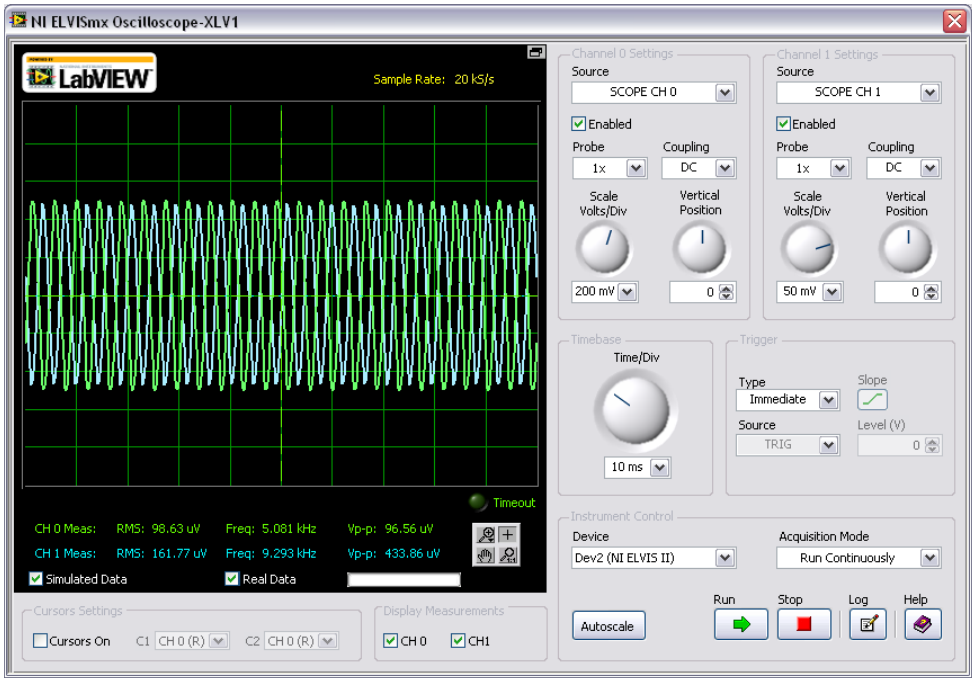


Figure E3-12. ELVISmx Oscilloscope Soft Front Panel

- End of Exercise -

Exercise 4 – Measurement and Comparisons

Objective

Compare previously saved simulation data with real-world measurements to verify bandpass filter operation.

Design

Because this exercise requires the NI ELVIS II workstation to be configured, it is recommended that the instructor leads the exercise through a demonstration. It is also assumed that the circuit from Exercise 3 is already prototyped on NI ELVISmx.

Procedure

A. Transient Verification

1. Open the NI ELVISmx instrument launcher **(Start » Programs » National Instruments » NI ELVISmx» NI ELVISmx Instrument Launcher).**
2. Open the NI ELVISmx Function Generator soft front panel by clicking on **FGEN**.
3. Verify the correct NI ELVIS device is selected in the **Device** drop down menu.
4. Set the input frequency to **230 Hz** and the input amplitude to **1** $\mathbf{V_{p\text{-}p}}$ using the fields. One variation is to check the box labeled **Manual Mode** to enable manual control of the hardware. This will allow the user to use the adjustment knobs on the front of the NI ELVIS II workstation.
5. Open the NI ELVISmx Oscilloscope soft front panel by clicking on **Scope**.
6. Click the **Run** button on both the Function Generator and the Oscilloscope.
7. Verify your circuit operates as indicated in Figure E4-1. Note the configuration of the NI ELVISmx Oscilloscope.

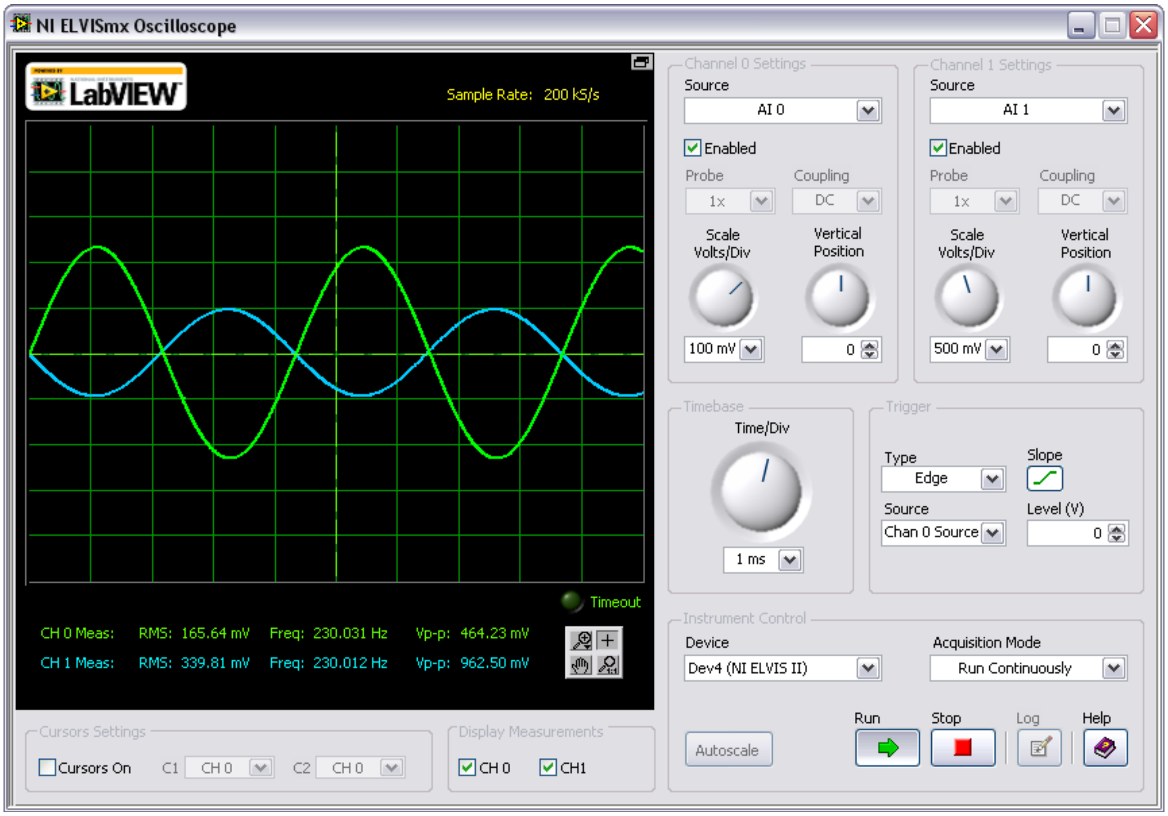


Figure E4-1. Transient Verification of Real-World Circuitry

Frequency Verification

Using the NI ELVISmx Bode analyzer, quickly verify the frequency response.

1. Turn off **Manual Mode** if this is currently on from the **FGEN** soft from panel
2. Close both the **Scope** and **FGEN** Soft Front Panels.
3. From the NI ELVISmx Instrument Launcher, click NI ELVISmx Bode analyzer. Ensure the correct are made.

   AI0: Signal

   AI1: FGEN

4. Run the sweep from **10 Hz to 1 KHz**. The results should be similar to those in Figure E4-2.

Figure E4-2. Frequency Response Verification of Real-World Circuitry

Comparisons

Using LabVIEW SignalExpress, to control NI ELVIS II to perform a Bode analysis, and load previously saved SPICE simulation results.

1. Open LabVIEW SignalExpress by clicking on S**tart » Programs » National Instruments » LabVIEW SignalExpress» LabVIEW SignalExpress x.x**
2. Click on the **"+Add Step"**, from the menu select **Load/Save Signals»Analog Signals»Load from SPICE.** See Figure E4-3 for location of the step.

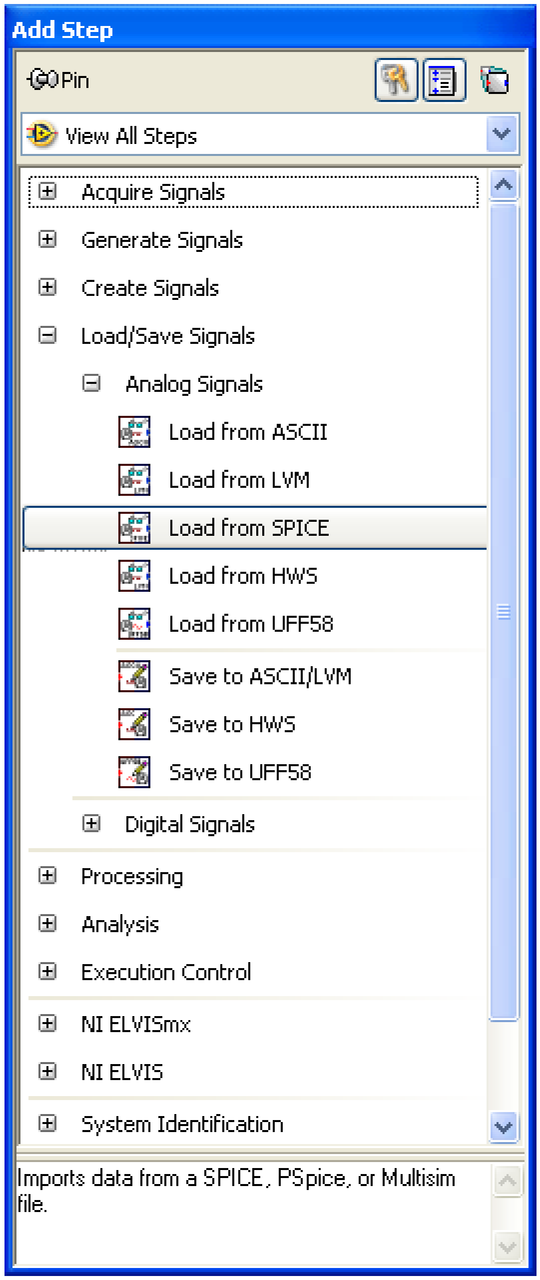


Figure E4-3. Add Step Menu

3. In the step setup, from the drop down menu select **Multisim** as the **Simulation file type**.
4. Click on the browse file button and locate the file **"Frequency Response.txt"** on your computer in the **Import file path** field.
5. Check boxes to import data from **V(Probe1)** and 20***log(mag(V(Probe1))).**
6. Select **XY Pair – Time** from the drop down menu for **Domain**.
7. From the drop down menu select Voltage:V as the Y-axis unit.
8. From the drop down menu select Time: s as the X-axis unit.
9. Add another step by selecting, **Add Step»NI ELVISmx»Analog»Acquire» NI ELVIS Bode Analyzer**
10. Using the same configuration on the Bode Analyzer in exercise 3, set the number of Steps as 10
11. In the **Data View** tab, click on the **Gain Data**, located under the NI ELVISmx Bode Analyzer task, and drag it to the top display.
12. Similarly, click and drag **Phase Data**, located under the NI ELVISmx Bode Analyzer task, and drag it to the bottom display.

13. Click **Run Once**. LabVIEW SignalExpress then run a frequency sweep using NI ELVISmx and collects 10 points per decade. **After** the results are generated, they are displayed on the graphs, along with the simulation results. The overall result is shown in Figure E4-4. The white traces are measured data, and the red tracers are simulation results.

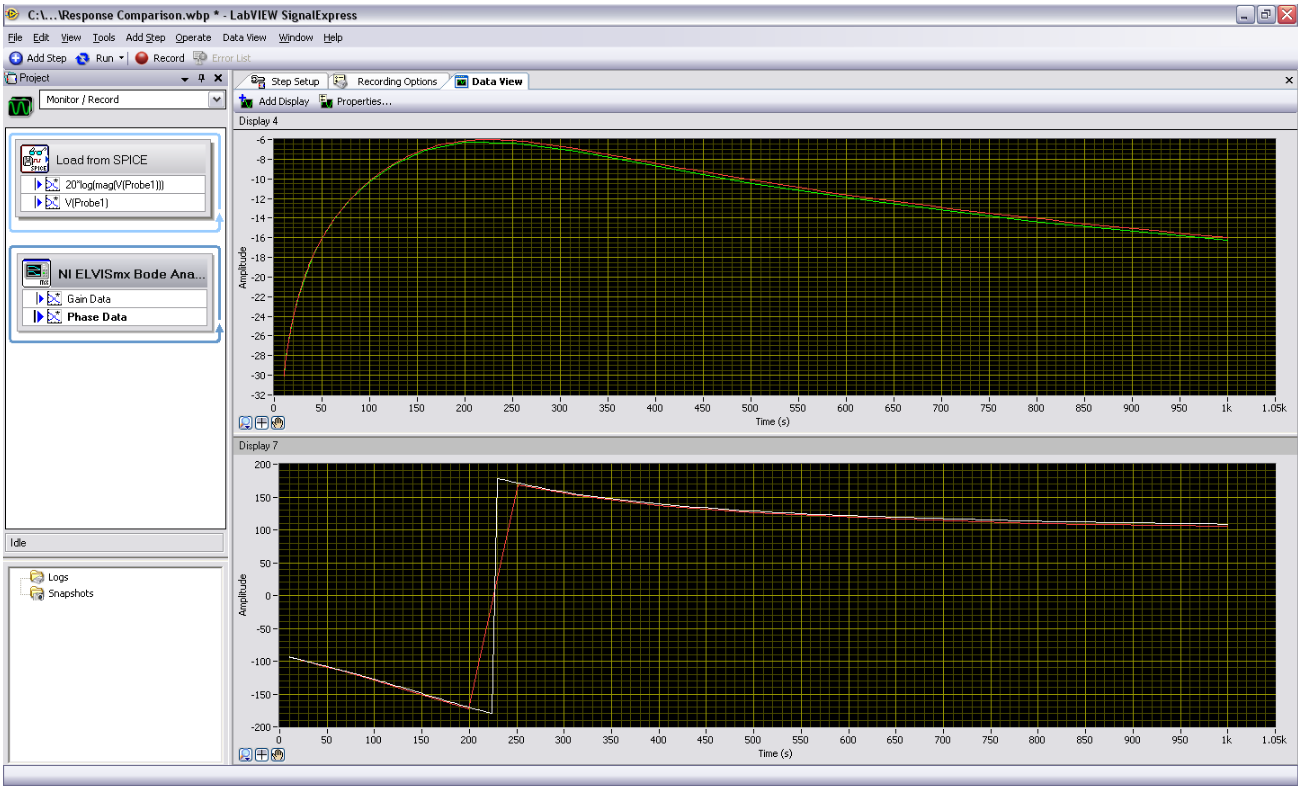


Figure E4-4. Frequency Response Verification of Real-World Circuitry

Note: The frequency data loaded from Multisim includes an additional, post-processed trace that is the magnitude response in decibels. Because of this, you can capture the gain data from LabVIEW SignalExpress to match the simulation data from Multisim. This trace was created, after first performing an AC analysis, by using the **Add Trace(s) to Plot** feature of the Grapher. The expression used was "20*log(mag(V(Probe1)))."

- End of Exercise -

Digital Logic Design with Elvis

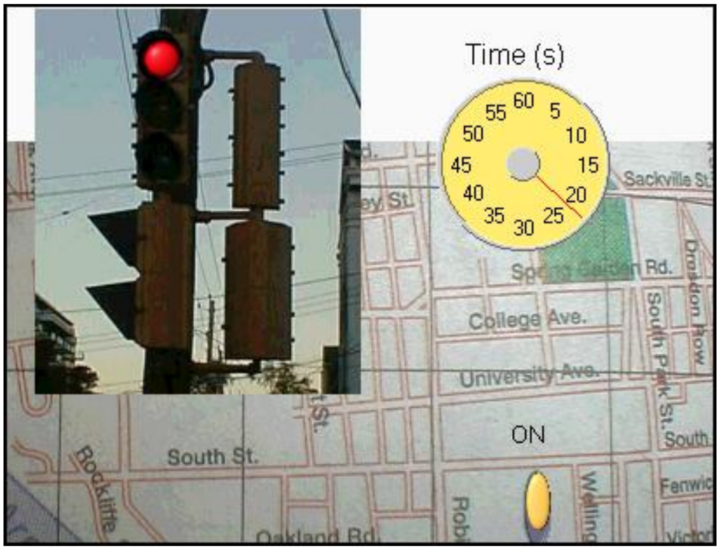


Figure 1. Stoplights with LabVIEW Indicators

Have you ever sat in your car stopped at a city intersection waiting for the stoplight to change and wondering how long the red light will last? Sometimes it seems like forever. Using a stop watch at a simple two-way intersection, you will find red lasts for 30 seconds, green lasts for 25 seconds, and yellow lasts for 5 seconds. In some states, these times may be two, three, or four times as long, but the ratios are always the same.

A property of an electronic diode is that in one direction current flows easily (forward biased), while in the other direction current flow is blocked. Light emitting diodes (LEDs) have the same property, but in the forward-biased region light is given off and in the reverse-biased region the LED is dark. Today, LEDs are used as the primary light elements in stoplights, so understanding how they operate is useful.

Goal

This lab focuses on using NI ELVIS II to illuminate diode properties, diode test methods, bit patterns for a two-way stoplight intersection, and the use of NI ELVIS II APIs in a LabVIEW program to run the stoplights automatically. A Multisim challenge encourages the reader to design a two-way stoplight intersection using discrete transistor-transistor logic (TTL) ICs.

Required Soft Front Panels (SFPs)

- Digital diode tester (DMM[▶|])
- Two-wire current-voltage analyzer (2-Wire)
- Digital writer (DigOut)

Required Components

- Silicon diode
- Six LEDs (2 red, 2 yellow, and 2 green)
- Six 220 Ω resistors

Exercise 1 Testing Diodes and Determining Their Polarity

Goal

A semiconductor junction diode is a polar device with a band on one end which indicates the cathode. The other end is called the anode. While there are many ways to indicate this polarity in the packaging of a diode, one thing is always the same—a positive voltage applied to the anode results in the diode being forward-biased so that current can flow. You can use NI ELVIS II to determine the diode polarity.

Procedure

Complete the following steps to set up NI ELVIS II for diode and polarity tests:

1. Launch the NI ELVISmx Instrument Launcher and select **DMM**.
2. Click on the diode test button [ ]. Click on **Run**.
3. Connect one of the LEDs to the workstation banana sockets DMM [VΩ] and [COM].

When you apply a positive voltage to the cathode, the diode blocks the current. The display, which reads the same value as it does when no diode is connected (open circuit), shows the word OPEN (see Figure 2).

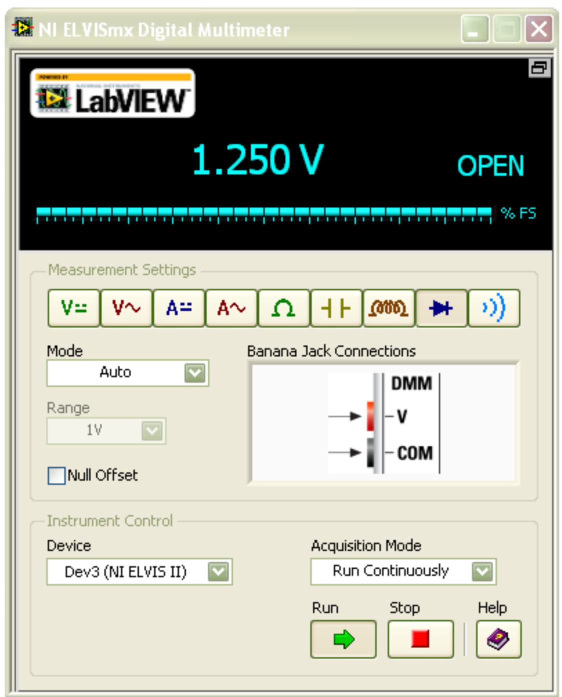


Figure 2. Reverse-biased Diode Reading

When you apply the positive voltage to the anode, the diode allows current to flow. The display reads a voltage level less than the open circuit value (1.250 V) and shows the word GOOD (see Figure 3).

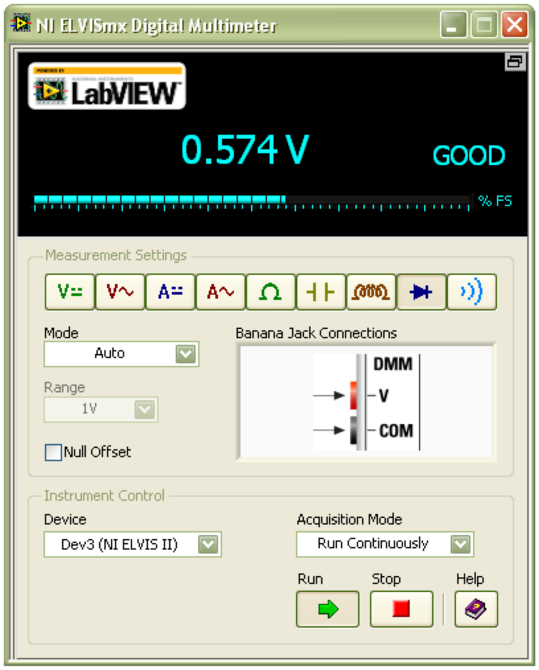


Figure 3. Forward-biased Diode Reading

For example, a silicon rectifying diode in the forward-bias direction displays a voltage ~0.6 V and shows the word GOOD. In the reverse-bias direction, the display reads the open circuit value (~1.250 V) and shows the word OPEN.

Note You can use this simple test to determine the polarity of a colored LED. Connect a red LED to your test leads. In one direction, you see light (forward-biased) and, in the other direction, no light (reverse-biased). The DMM display does not change, but there is enough current to produce some light. Check closely the LED is dimly lit and may be difficult to see with bright lights in the room. When the LED is lit, the red lead connection is the anode.

The way this works is that the display shows the voltage required to generate a small current flow of about 1 mA. In the forward-bias region, this voltage level is usually smaller than the open circuit voltage. In the reverse-bias direction, no current flows and the tester displays the open circuit voltage, about 1.250 V. For LEDs, the voltage threshold is often larger than the open circuit voltage. The 1 mA test is not sufficient to discern the forward-bias test (GOOD), but it is enough to generate a low light intensity.

End of Exercise 1

Exercise 2 Characteristic Curve of a Diode

Goal

The characteristic curve of a diode, that is, a plot of the current flowing through the device as a function of the voltage across the diode, best displays the diode's electronic properties. Objective is to display diodes electronic properties.

Procedure

Complete the following steps to display the characteristic curve of a diode:

1. Place a silicon diode across the DMM/Impedance Analyzer pin sockets DUT+ and DUT–. The anode diode pin goes to the + input. For reference, the flat side of the LED is the cathode.
2. Launch the NI ELVISmx Instrument Launcher and select the Two-Wire Current-Voltage Analyzer **(2-Wire)**. A new SFP opens so you can display the characteristic (I-V) curve for the device under test. This SFP applies a test voltage to the diode from a starting voltage level to an ending level in incremental voltage steps, all of which you can select.
3. For a silicon diode, set the following parameters: Start –2 V

Stop	+2.0 V
Increment	0.05 V

4. Set the maximum current in either direction to ensure the diode does not operate in a current region where damage may occur. Check the diode specifications.
5. Click on **Run** and see the I-V curve appear.

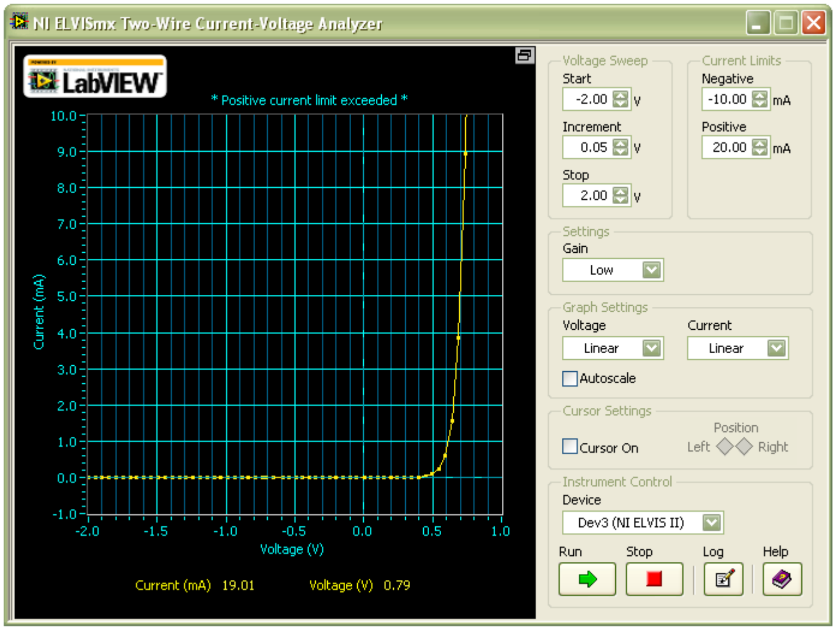


Figure 4. Current-VoltageCharacteristic Curve of a Silicon Diode

In the reverse-bias direction, the current should be very small (µA) and negative. In the forward-bias direction, you should see that above a threshold voltage, the current rises exponentially to the maximum current limit.

6. Change the **Display** buttons [Linear/Log] to see the curve plotted on a different scale.
7. Try the **Cursor** operation. It gives the (I,V) coordinate values as you move the cursor along the trace.

The threshold voltage is related to the semiconductor material of the diode. For silicon diodes, the threshold voltage is about 0.6 V, and for germanium diodes, it is about 0.3 V. One way to estimate the threshold voltage is to fit a tangent line in the forward-bias region near the maximum current (refer to Figure 5). The point where the tangent intersects the voltage axis defines the threshold voltage. Observe the (I,V) characteristic curve for a light emitting diode. For this LED, the threshold voltage given by the intersection of the tangent with the voltage axis is about 1.56 V.

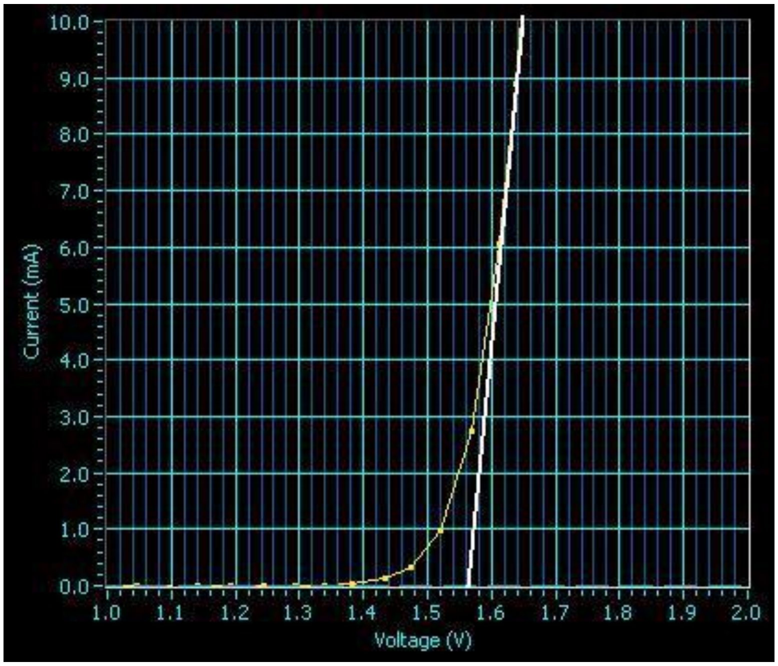


Figure 5. Current-Voltage Curve of a Red LED with Tangent Line

8. Using the **Two-Wire Current-Voltage Analyzer,** determine the threshold voltage for a red, yellow, and green LED, and complete the chart below.

   Red LED ________________________________V

   Yellow LED ________________________________V

   Green LED ________________________________V

Do you see a trend?

End of Exercise 2

Exercise 3 Manual Testing and Control of a Two-Way Stoplight Intersection

Goal

Objective of this exercise is to establish two way stoplight intersection algorithm to identify bit signals to be given to leds for its operation.

Procedure

Complete the following steps to build and manually test and control a two-way stoplight intersection.

1. Install two each of red, yellow, and green LEDs on the NI ELVIS II protoboard, positioned as a two-way stoplight intersection.

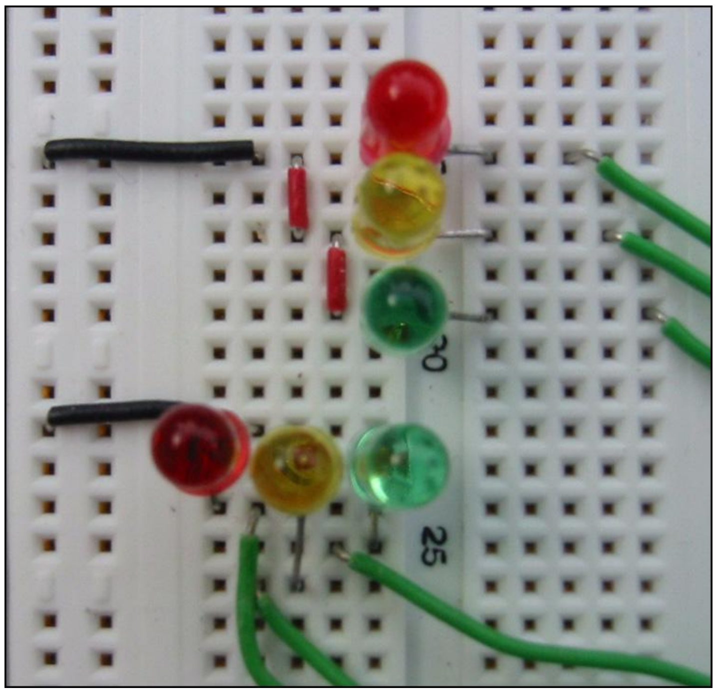


Figure 6. LED layout of a Two-way Stoplight Intersection

Each LED is controlled by one binary bit on one of the 8-bit parallel ports on the protoboard. Use digital I/O bit sockets DIO <0..7>.

2. Connect the pin socket DIO <0> to the anode of the red LED in the North-South (Up-Down) direction.

3. Connect the other end of the LED through a 220 Ω resistor to digital ground (not pictured).

Note The resistor is used to limit the current through the LED.

4. Connect the remaining colored LEDs in a similar fashion.

   Here is the complete mapping scheme.

DIO <0> Red	N-S direction	DIO <4> Red	E-W direction
DIO <1> Yellow	N-S direction	DIO <5> Yellow	E-W direction
DIO <2> Green	N-S direction	DIO <6> Green	E-W direction

5. From the NI ELVISmx Instrument Launcher, select **Digital Writer (DigOut)**.
6. Using the vertical slide switches, select any 8-bit pattern and output that pattern to the NI ELVIS II digital lines. Recall that Bit 0 is connected to the pin socket on the protoboard labeled DIO <0>.
7. Set the **Generation Mode** to **(Run Continuous)** and **Pattern** to **(Manual),** as shown in Figure 7.
8. To activate the port, click on the **Run** button.

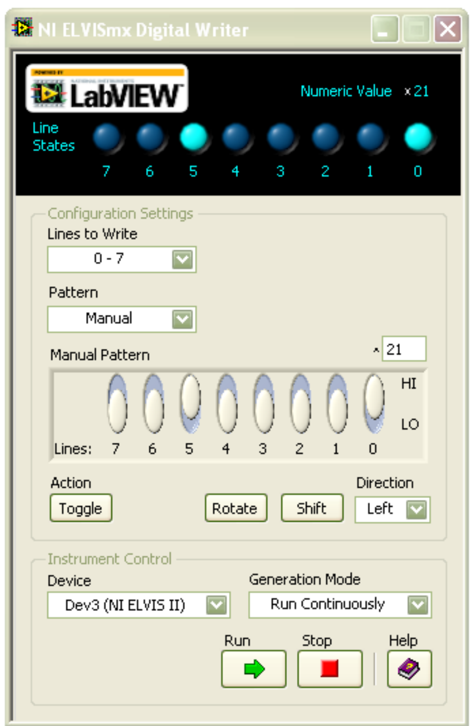


Figure 7. Digital Writer for Testing LEDs

When all switches (Bits 0-2 and 4-6) are HI, all the LEDs should be lit. When all these switches are LO, all the LEDs should be off.

You can now use these switches to find out which 8-bit codes are necessary to control the various cycles of a stoplight intersection.

Here are some clues for an intersection. The basic operation of a stoplight is based on a 60-second time interval with 30 seconds for red, followed by
25 seconds for green, followed by 5 seconds for yellow. For example, in a two-way intersection, the yellow light in the North-South direction is on while the red light in the East-West direction is on. This modifies the
30-second red timing interval to two timing intervals: a 25-second cycle followed by a 5-second cycle. There are four timing periods (T1, T2, T3, and T4) for two-way stoplight intersection operation.

9. Study the following chart to see how a two-way stoplight intersection works.

Direction	**Lights Bit#**	**N-S RYG 012**	**E-W RYG 456**	**8-Bit Code**	**Decimal Value**
T1	25 s	001	100	00010100	20
T2	5 s	010	100	__________	__________
T3	25 s	100	001	__________	__________
T4	5 s	100	010	__________	__________

10. Use the Digital Writer to determine which 8-bit codes need to be written to the digital port to control the stoplights in each of the four timing intervals.

    For example, timing period 1 requires the code 00101000. Computers read the bits in the reverse order (least significant bit on the right). This code then becomes **00010100**. In the white box above the Manual Pattern Line switches display, you can read the radix of the switch pattern in binary {**00010100**}, decimal {**20**}, or hexadecimal {**14**}.

11. Click on the black ^ to left of the white display box to change the radix.
    You can use this feature to determine the numeric codes for the other timing intervals T2, T3, and T4. If you output the 8-bit code for each of the timing intervals in sequence, you can manually operate the stoplights.

Note You can also change the radix in the Line States display by clicking on the white x beside the Numeric Value display.

Repeating this four-cycle sequence automates your intersection.

End of Exercise 3

Exercise 4 Automatic Operation of the Two-Way Stoplight Intersection

Goal

Objective of this exercise is to automate two way traffic lights by writing a small program in NI Labview.

Procedure

Complete the following steps to automate the timing cycle on the stoplight circuit.

1. Close NI ELVIS II SFPs and launch LabVIEW.
2. Open the program StopLightsMx.vi. There is only one control on the front panel a Boolean switch used to stop the operation of the stoplights.
3. Switch to the block diagram **(Window»Show Block Diagram).**
4. Observe the four-cycle sequence generated by the for loop.

   The NI ELVISmx Digital Writer API is the structure that outputs the light code to the stoplights. This API expects the input code to be an 8-bit Boolean array. For example, the first timing interval T1 requires the code 20 (twenty decimal). Its value is placed in the first element of an integer array labeled **Lights Pattern**. You must transfer the other integer codes from the table in Exercise 3 into the three blank elements of the Lights Pattern array.

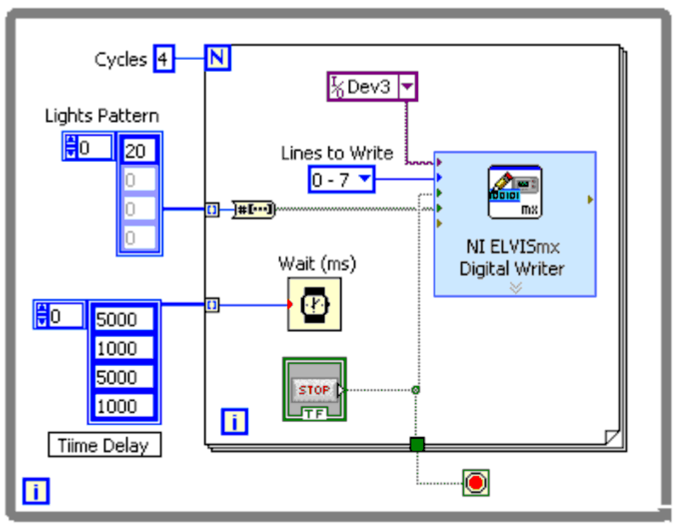


Figure 8. Block Diagram for Automated Operation of a Two-way Stoplight Intersection

In operation, one of the elements of the Lights Pattern array is selected on the boundary of the for loop (inner loop) and converted into an 8-bit Boolean array. In a similar way, the appropriate time delay is selected at the for loop boundary and passed to the Wait function.

The timing intervals are stored in the four elements of the **Time Delay** array. To speed up operation, the 25-second time interval is reduced to 5 seconds and the 5-second time interval is reduced to 1 second.

End of Exercise 4

RF Wireless Communications

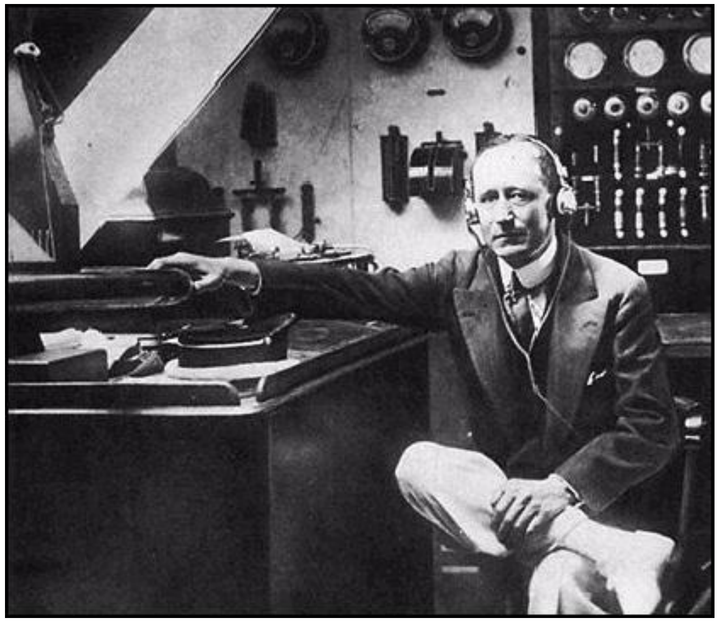


Figure 1. Guglielmo Marconi

Midday at Signal Hill near St. John's, Newfoundland, in Canada, Guglielmo Marconi pressed his ear to a telephone headset connected to an experimental wireless receiver. About 1,700 miles away at Poldhu, Cornwall, in England, his coworkers were about to send the Morse code letter s, which is three dots. Faintly, but clearly "psht-psht-psht" pause "psht-psht-psht" came through the earphone. The date was December 12, 1901, and the first transatlantic message had just been sent and received.

Goal

In this lab, use a paper clip antenna to send this classic message and other waveforms over a wireless radio frequency (RF) link. The NI ELVIS II function generator is the transmitter and a high-gain op amp is the receiver. The classic message is formulated using the NI ELVIS II arbitrary waveform generator.

Required Soft Front Panels (SFPs)

- Oscilloscope (Scope)
- Arbitrary waveform generator (ARB)

Required Components

- 1 kΩ resistor (brown, black, red)
- 100 kΩ resistor (brown, black, yellow)
- 741 op amp or field-effect transistor (FET) op amp 753
- Paper clips

Exercise 1 The Transmitter

Goal

Objective of this exercise is to make a simple transmitter by using a paper clip or a piece of wire.

Procedure

Complete the following steps to build a simple transmitter antenna from a paper clip:

1. Straighten a paper clip and cut it into a piece about 2.5 in. long.
2. Push one end of the paper clip into the output pin socket of the function generator.

   When FGEN is running, the output voltage leaks from the pin socket to the paper clip antenna and radiate a small RF signal. A similar antenna about a centimeter away can pick up this signal and amplify it to a higher signal level. Use this transmitter in Exercise 2.

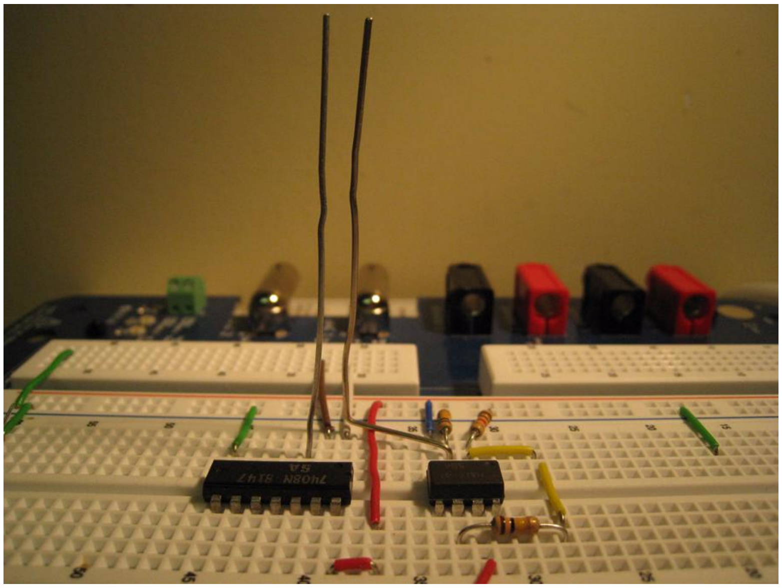


Figure 2. RF Transmitter-Receiver circuit with Antennas

3. Initially, use a sine wave to test the transmitter by setting the SFP function generator to sine waveform, 2.5 V amplitude, and 10000 Hz frequency.

End of Exercise 1

Exercise 2 The Receiver

Goal

Objective of this exercise is to build a receiver using a paper clip and then amplify the received signal with an opamp.

Procedure

Complete the following steps to build a simple receiver antenna from a paper clip:

1. Bend a second paper clip into step shape, with the long side about 2.5 in, the step height about 0.25 in., and the step width about 0.5 in.
2. Insert the short end of the paper clip into a pin socket. The midsection supports the antenna on the protoboard, so you can rotate the antenna about the short end. The long side sits vertically and is parallel to the transmitter antenna (see Figure 2).
3. Build a high-gain amplifier using a 741 op amp or 753 FET op amp in the simple inverting configuration.

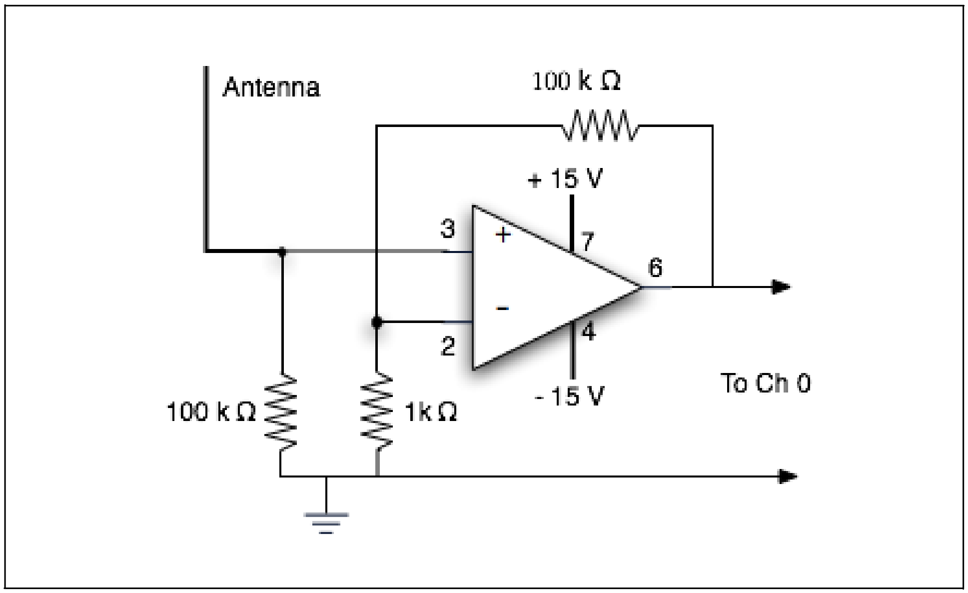


Figure 3. RF Receiver Op Amp Circuit

4. Connect a 1 kΩ resistor to the – input (pin 2).
5. Connect a 100 kΩ bias resistor to the + input (pin 3).
6. Connect the other end of the resistors to AIGND.
7. Connect a 100 kΩ resistor as the feedback resistor R_f from pin 2 to pin 6.
8. To power the circuit, connect +15 V on pin 7 and –15 V on pin 4.

Nominally the op amp has a gain of 101. You can use other resistor combinations for higher gains.

9. The receiver antenna is connected to the input (pin 3).
10. Connect the op amp output pin 6 to the oscilloscope.

End of Exercise 2

Exercise 3 Testing the RF Transmitter and Receiver

Goal

Objective of this exercise is to test functionality of Transmitter and receiver by using a random deterministic signal.

Procedure

Complete the following steps to use a sine wave signal to test the transmitter-receiver pair.

1. Check the circuit you built in Exercise 9-2 and power on the protoboard.
2. Move the receiver antenna a few millimeters from the transmitter antenna.
3. Connect the oscilloscope BNC connector channel **(CH0)** to the op amp output, pin 6, and ground.
4. Connect the oscilloscope BNC connector channel **(CH1)** to the function generator pin socket **(SYNC)**.
5. Typical oscilloscope settings are: Channel 0: 10 to 500 mV

   Channel 1: 2 V/div

   Trigger source: Channel 1
6. Decrease Channel 0 scale (V/div) until you see a sine wave.

   If you cannot see a signal right away, touch the two antenna tips with your fingertip. This simulates the high impedance of the atmosphere and allows a small signal to propagate.
7. Adjust the FGEN amplitude and frequency until you get a good signal.
8. Measure the signal level as you separate the receiver antenna from the transmitter antenna. You can easily measure the separation with a ruler.

   You can quickly get an idea of how rapidly the signal level falls off with distance; a long antenna helps in receiving distant signals. Marconi, at Signal Hill, used a kite to carry his antenna hundreds of feet up into the atmosphere.

   Now that the transmitter-receiver is working, it is time to duplicate Marconi's classic message.

   Marconi's first RF transmitter consisted of a spark gap connected to a resonant circuit and a very long antenna often carried high on a balloon or kite. When a spark is discharged between the electrodes, an intense RF pulse is generated with a short time duration of a few milliseconds.

   It takes 30,000 V to produce a spark between electrodes separated by 1 cm, and the current can be large. A single spark followed by a pause was a dot. A longer spark followed by a pause was a dash. Together, these were all the ingredients needed for Morse code transmission. The letter S is just three dots in rapid succession. The letter O is just three dashes in rapid succession. The distress call, S-O-S (save our souls), is:

   ```
   dot-dot-dot dash-dash-dash dot-dot-dot
   ```

For the first transatlantic message, Marconi chose the simpler signal dot-dot-dot.

End of Exercise 3

Exercise 4 Building a Unique Test Signal with an Arbitrary Waveform Analyzer

Goal

A dot is a signal, usually an oscillation, followed by silence (no signal). Each part lasts for about 0.1 second. A dash is just a signal lasting for the duration of three dots, or 0.3 second, followed by a pause. The encoding scheme is a simple tone burst with different duration times. The letter S is encoded as dot-dot-dot or, in binary, 101010, where 1 is the dot and 0 is the pause. A longer message consisting of multiple letters like SSS has a longer pause (0.4 second) placed between each letter. This message in binary is

101010 0000 101010 0000 101010 0000.

If you can generate this waveform on the NI ELVIS II digital-to-analog converter, or DAC (AO), then you can use the DAC output to gate the function generator on and off. The resulting tone burst signal from the FGEN can radiate the message to the world.

Procedure

Complete the following steps to build a program to produce a Morse code transmission:

1. From the NI ELVISmx Instrument Launcher, select Arbitrary Waveform Generator (ARB).

   With the arbitrary waveform generator, you can create unique waveforms, such as Marconi's first message. You can use a special program called the Waveform Editor to create all kinds of unique diagnostic and control waveforms.

2. Click the Waveform Editor button to view this feature.

   The SFP ARB provides waveform control over the AO 0 and AO 1 outputs.

3. Click on the browser icon next to the DAC0 Waveform Name box.
4. From the NI ELVIS II library folder, select the 1VSine1000.wdtfile. Enable AO output by clicking on AO 0:[box]. When you click on the Run button, a 1.0 V amplitude sine wave at 1000 Hz is applied to the AO 0 pin socket.

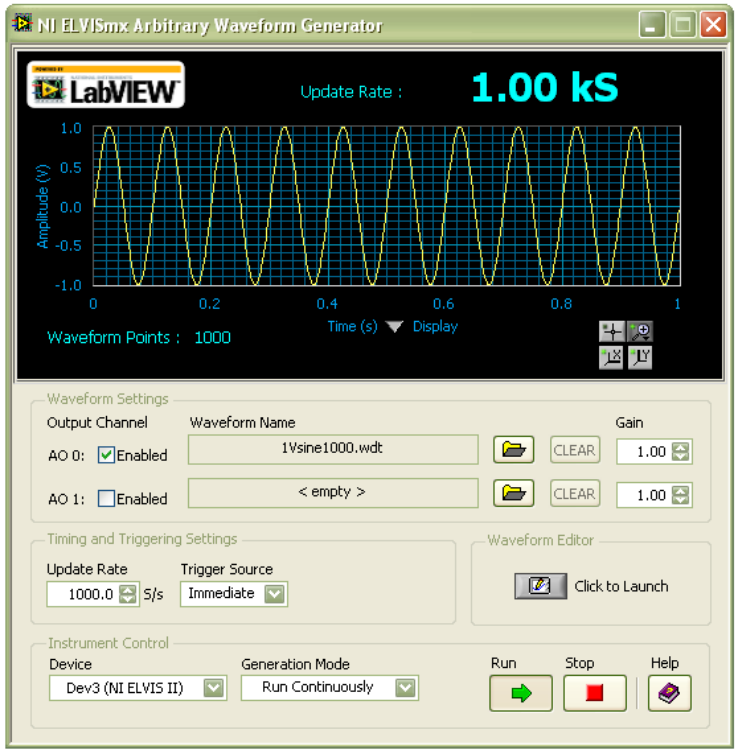


Figure 4. ARB created 1 V Sine Waveform

5. Connect the oscilloscope CH 0 BNC input to the AO 0 pin socket. Click the **Run** button and observe a 1 kHz sine wave signal on the oscilloscope window.

Note For a steady signal trace, trigger on Channel 0.

6. Return to the AO 0 browser icon, navigate to the Hands-On NI ELVIS II library folder, and select the file Morse.wdt. This file provides the waveform for the letter S in Morse code. Change the AO 0 gain to 2.5.
7. Click **Run** and observe this signal on the oscilloscope.

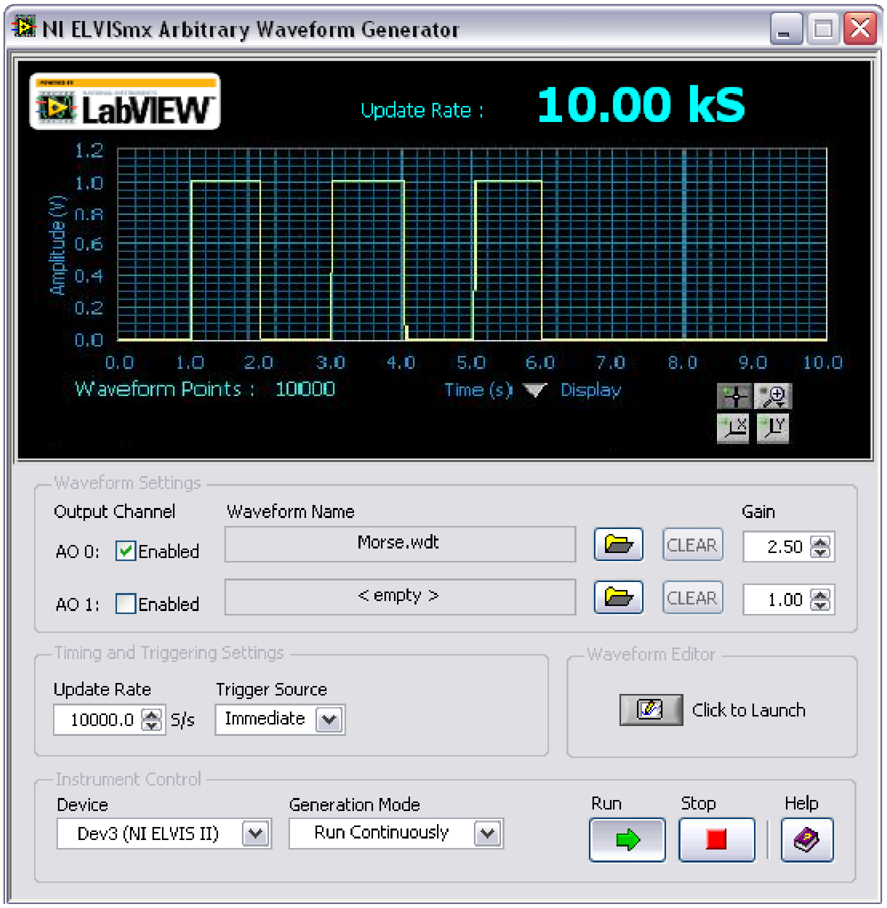


Figure 5. ARB Created Morse Code Letter S

For the real transmission, change the Update Rate box to 10000.0 S/s.

End of Exercise 4

Temperature Chamber Control Using cDAQ

Demonstration of Control Systems

Purpose of this demonstration is to introduce basic concepts of control systems and to familiarize with instrumentation using NI cDAQ.

In this demonstration we will be controlling temperature of a chamber using a bulb and a fan. Temperature of the chamber is acquired by using a thermistor. All these inputs and outputs are communicated to our processing unit through cDAQ and acquisition modules are 9201 and 9474.

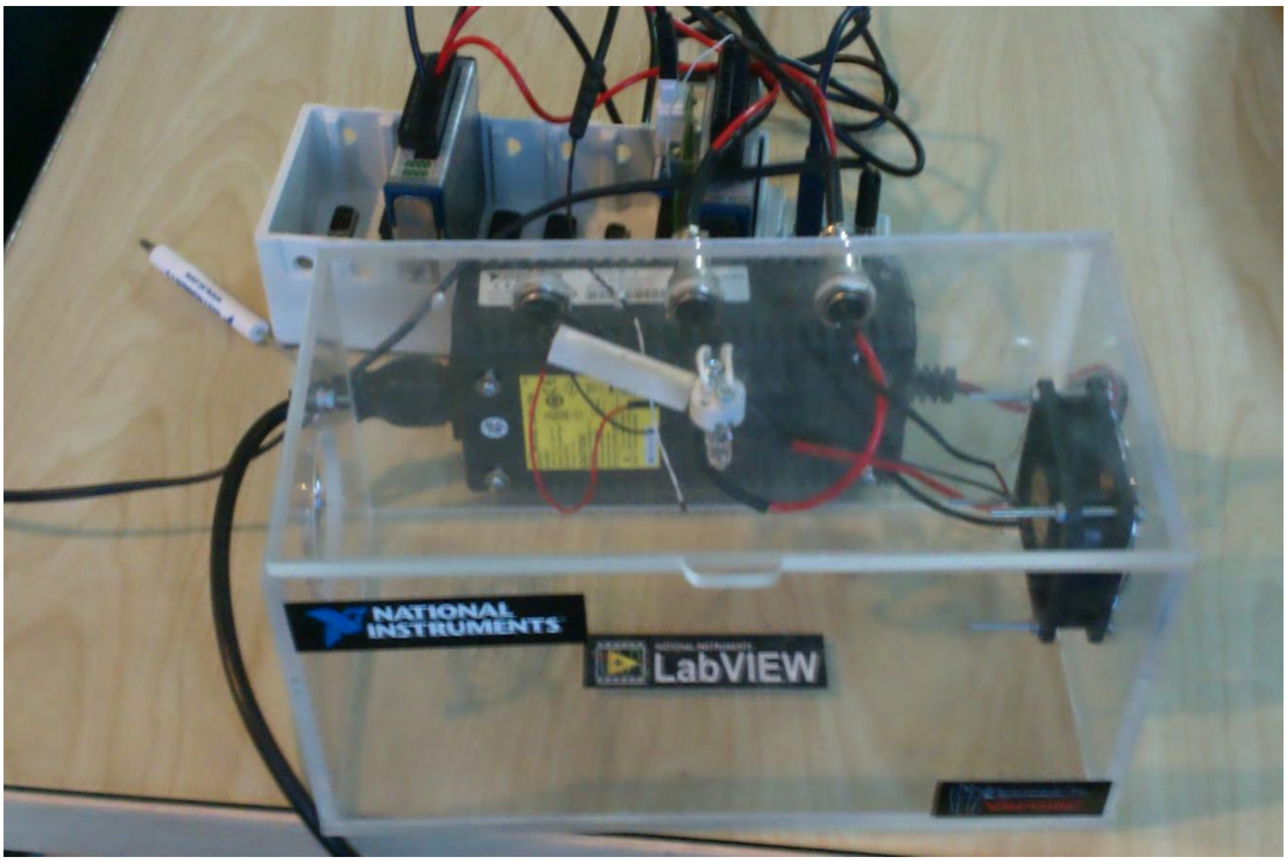


Figure 1 – Temperature Chamber Assembly