

Editing and Debugging in LabView

1. Editing Tools

- a. How to create controls, indicators, and constants directly on the block diagram
 - i. By using the right click button on a function icon near an input or output and selecting the appropriate variable
 - ii. Use the “change to” or “replace” option by right clicking to change the variable or function
- b. Use the positioning tool on the tools palette to select and move, create a selection window by dragging, use shift key for multiple selections, use the cut/paste, copy/paste, drag/drop, or control and drag to create another instance of a variable or function
- c. You can import text or figures by copy and paste
- d. Resize objects by pulling the object handles and enlarge or reduce
- e. The labeling tool is used for create (double click anywhere on screen) or modifying a label
- f. Use text settings in the toolbar to modify font, font size, style
- g. How to work with wires?
 - i. Select and delete wires with the selection tool. Double or triple clicking selects wire segment or wire branch
 - ii. Wires can be selected and stretched or moved
 - iii. Broken wires are present when you see a broken “run” symbol
 1. Double click the broken wire to see a list of all errors and warnings
 2. Double click an error to see where it is located and to learn more information about the type of error. Typical problems are wire type, wire dimension, element conflicts, multiple wire sources, no wire sources, or wire cycles.
 - iv. Use ctrl-B to remove broken wires
 - v. Right click on wire to clean up a wire branch
- h. Use the align/distribute/resize options in the toolbar to align, distribute or resize objects on the screen to control their layout on the screen.
- i. Use the coloring tools on the tools palette to control the foreground and background color, T makes for transparent colors.
- j. Find editing examples in **Example Finder > Browse Task > Building User Input > General > Editing Exercise, Resizing Objects**

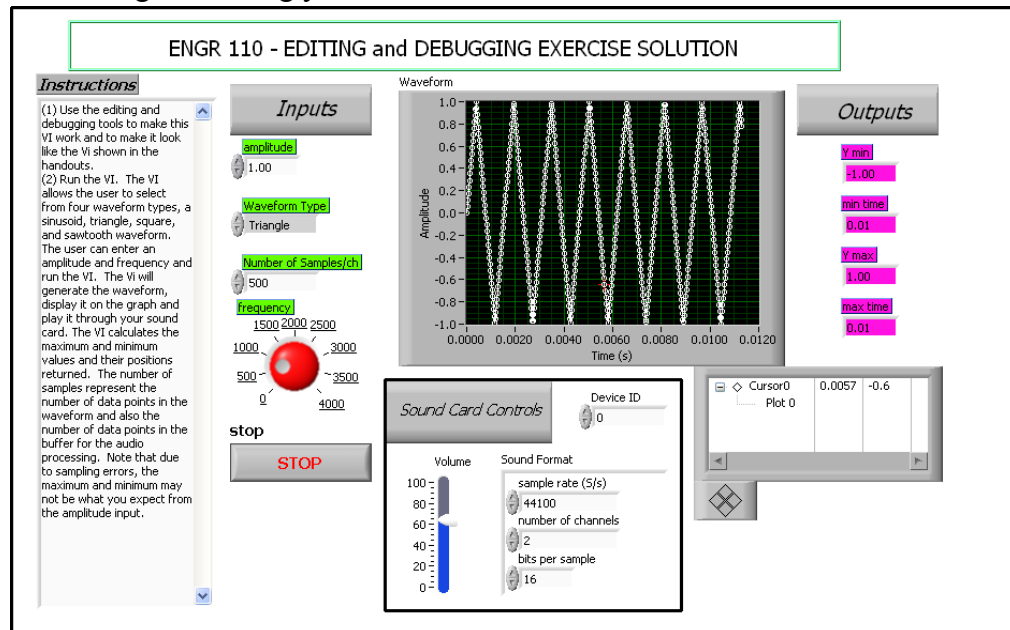
2. Debugging tools

- a. Find errors by clicking on the broken run icon
 - i. Errors are listed
 - ii. Activate an error by clicking on it

- b. Highlight execution button allows you to follow the execution of the program. This will slow down the execution.
- c. Single stepping, stepping into, stepping over, stepping out are the traditional debugging tools found in any program language
- d. Use breakpoints to halt execution at a particular location in the program, and use probes to evaluate data as it flows through the block diagram. Use conditional probe to built in conditions.
- e. Operational: see what the effect of sampling rate is on the discretization of the waveform.
- f. Find debugging examples in **Example Finder > Browse Task > Fundamentals > Debugging > Using Supplied Probes**

3. Exercise for Editing and Debugging

- a) Open up the VI Waveform_edit.vi. This VI is not complete. It does contain all inputs and outputs. The VI allows the user to generate four different types of waveforms, sinusoidal, square, triangular, and sawtooth. The user specifies the frequency (number of cycles per second) and the amplitude of the wave. The program will then display the wave and play it through the sound card of the computer.
 - i) Open the VI.
 - ii) Use the editing and wiring tools to complete the VI.
- b) After finishing the editing your VI should look like this.



- c) When the VI is operational, run it and experiment with the different type of waveform inputs. Also vary the amplitude and frequency of the wave. You can see whether the volume input to the computer speaker is operational. In addition see what the effect of sampling rate is on the operation. In addition see what the effect of sampling rate is on the discretization of the waveform.

