

DEBUGGING TECHNIQUES*

National Instruments

This work is produced by The Connexions Project and licensed under the Creative Commons Attribution License †

Abstract

In this section, you will learn about LabVIEW debugging techniques.



If a VI does not run, it is a broken, or nonexecutable, VI. The Run button often appears broken, shown at left, when you create or edit a VI. If it is still broken when you finish wiring the block diagram, the VI is broken and will not run. Generally, this means that a required input is not wired, or a wire is broken.

1 Finding Errors

Click the broken Run button or select **Windows»Show Error List** to display the **Error list** window, which lists all the errors. Double-click an error description to display the relevant block diagram or front panel and highlight the object that contains the error.

2 Execution Highlighting



View an animation of the execution of the block diagram by clicking the **Highlight Execution** button, shown in this media. Execution highlighting shows the flow of data on the block diagram from one node to another using bubbles that move along the wires. Use execution highlighting in conjunction with single-stepping to see how data move from node to node through a VI.

NOTE: Execution highlighting greatly reduces the speed at which the VI runs.

3 Single-Stepping

Single-step through a VI to view each action of the VI on the block diagram as the VI runs. The single-stepping buttons affect execution only in a VI or subVI in single-step mode. Enter single-step mode by clicking the **Step Over** or **Step Into** button. Move the cursor over the **Step Over**, **Step Into**, or **Step Out** button to view a tip strip that describes the next step if you click that button. You can single-step through subVIs or run them normally.



If you single-step through a VI with execution highlighting on, an execution glyph, shown in this media, appears on the icons of the subVIs that are currently running.

*Version 1.1: Jul 16, 2004 2:54 pm GMT-5

†<http://creativecommons.org/licenses/by/1.0>

4 Probes



Use the **Probe** tool, shown in this media, to check intermediate values on a wire as a VFI runs. When execution pauses at a node because of single-stepping or a breakpoint, you also can probe the wire that just executed to see the value that flowed through that wire.

You also can create a custom probe to specify which indicator you use to view the probed data. For example, if you are viewing numeric data, you can choose to see that data in a chart within the probe. To create a custom probe, right-click a wire and select **Custom Probe**»**New** from the shortcut menu.

5 Breakpoints



Use the **Breakpoint** tool, shown in this media, to place a breakpoint on a VI, node, or wire on the block diagram and pause execution at that location. When you set a breakpoint on a wire, execution pauses after data pass through the wire. Place a breakpoint on the block diagram workspace to pause execution after all nodes on the block diagram execute. When a VI pauses at a breakpoint, LabVIEW brings the block diagram to the front and uses a marquee to highlight the node or wire that contains the breakpoint. LabVIEW highlights breakpoints with red borders for nodes and block diagrams and red bullets for wires. When you move the cursor over an existing breakpoint, the black area of the **Breakpoint** tool cursor appears white. Use the **Breakpoint** tool to click an existing breakpoint to remove it.