Welcome to LabVIEW — graphical programming for data acquisition, instrument I/O, measurement analysis, and visualization. This card provides information about getting started with LabVIEW quickly.

**Installing LabVIEW**

Complete the following steps to install LabVIEW.
1. Run the setup program on the CD.
2. Be sure to register online on [www.ni.com/register](http://www.ni.com/register). You also can fill out the LabVIEW Product Registration Card and return it to National Instruments. Registering qualifies you for support, upgrades, and other important product information.
3. If you are new to LabVIEW, complete the LabVIEW Tutorial by starting LabVIEW and clicking the LabVIEW Tutorial button.

Read and complete the exercises in the Getting Started with LabVIEW manual. For data acquisition or instrument I/O examples, click the DAQ Solutions button. The DAQ Solutions button is available on Windows and Macintosh only. For other examples, click the Search Examples button.

**Where to Go Next**

While you are in LabVIEW, select Help»Show Context Help to display the Context Help window. Select Help»Contents and Index to display the LabVIEW Help. Refer to the following manuals for additional LabVIEW information:

- Data Acquisition : LabVIEW Measurements Manual
- Instrument Control : LabVIEW Measurements Manual
- LabVIEW programming : LabVIEW Help or LabVIEW User Manual

Refer to [www.ni.com/library](http://www.ni.com/library) for LabVIEW books and other resources.

© Copyright 1997, 2000 National Instruments Corporation. All rights reserved. Product and company names listed are trademarks or trade names of their respective companies.
LabVIEW Environment

The front panel contains controls that supply data to the block diagram of the VI. Indicators display data the block diagram acquires or generates.

The block diagram is the source code for the VI. You build the block diagram by wiring together objects that send or receive data, perform specific tasks, and control the flow of execution.

The VI Hierarchy window displays a graphical representation of the calling hierarchy for all VIs in memory. Select Browse→Show VI Hierarchy to display the VI hierarchy.

Wiring Techniques

1. Click to tack down a wire
   - tack down wire

2. Double-click to tack down a temporary wire
   - tack down a temporary wire

3. Use space bar to change direction of wire
   - press space bar

4. Remove last tack point
   - Windows: Ctrl-left-click
   - Macintosh: Option-click
   - UNIX: Middle-click

5. Highlight wire segments
   - single-click
   - double-click
   - triple-click
### VI Navigation

<table>
<thead>
<tr>
<th>Description</th>
<th>Technique</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Find terminals, local variables, references, invoke nodes,</td>
<td>Right-click the control on the front panel and select <strong>Find</strong> from the shortcut menu to locate the terminal, local variable, reference, invoke node, or property node on the block diagram.</td>
</tr>
<tr>
<td>and property nodes on the block diagram associated with a front panel</td>
<td>-----------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>control</td>
<td>-----------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>2. Find text and objects in memory</td>
<td>Select <strong>Edit»Find</strong> or: Windows: <strong>Ctrl-f</strong> Macintosh: <strong>Command-f</strong> UNIX: <strong>meta-f</strong></td>
</tr>
<tr>
<td>3. Find a VI, global variable, or type definition in the VI hierarchy</td>
<td>Select <strong>Browse»Show VI Hierarchy</strong> then select <strong>Edit»Find</strong> or type object name.</td>
</tr>
<tr>
<td>4. Open subVI front panel</td>
<td>Double-click subVI</td>
</tr>
<tr>
<td>5. Open subVI block diagram</td>
<td>Double-click subVI while pressing: Windows: <strong>Ctrl</strong> Macintosh: <strong>Option</strong> UNIX: <strong>meta</strong></td>
</tr>
</tbody>
</table>

### Debugging Techniques

<table>
<thead>
<tr>
<th>Technique</th>
<th>Icon</th>
<th>Description</th>
<th>Shortcut Keys</th>
</tr>
</thead>
<tbody>
<tr>
<td>Probe Tool</td>
<td><img src="https://example.com/probe_icon.png" alt="Probe Icon" /></td>
<td>Displays intermediate values on a wire in a running VI.</td>
<td></td>
</tr>
<tr>
<td>Breakpoint Tool</td>
<td><img src="https://example.com/breakpoint_icon.png" alt="Breakpoint Icon" /></td>
<td>Specifies node you pause on during execution.</td>
<td></td>
</tr>
<tr>
<td>Execution Highlighting</td>
<td><img src="https://example.com/highlight_icon.png" alt="Highlight Icon" /></td>
<td>Animates the movement of data on the block diagram using bubbles that move along the wires.</td>
<td></td>
</tr>
<tr>
<td>Pause</td>
<td><img src="https://example.com/pause_icon.png" alt="Pause Icon" /></td>
<td>Temporarily stops execution to debug a portion of VI.</td>
<td></td>
</tr>
<tr>
<td>Step Into</td>
<td><img src="https://example.com/step_into_icon.png" alt="Step Into Icon" /></td>
<td>Single-steps into a subVI or structure to debug it.</td>
<td>Windows: <strong>Ctrl-0</strong> Macintosh: <strong>Command-0</strong> UNIX: <strong>Meta-0</strong></td>
</tr>
<tr>
<td>Step Over</td>
<td><img src="https://example.com/step_over_icon.png" alt="Step Over Icon" /></td>
<td>Executes a subVI or structure and resumes single-stepping in next main function.</td>
<td>Windows: <strong>Ctrl-→</strong> Macintosh: <strong>Command-→</strong> UNIX: <strong>Meta-→</strong></td>
</tr>
<tr>
<td>Step Out</td>
<td><img src="https://example.com/step_out_icon.png" alt="Step Out Icon" /></td>
<td>Executes a subVI or structure and resumes single-stepping in calling VI or structure.</td>
<td>Windows: <strong>Ctrl-←</strong> Macintosh: <strong>Command-←</strong> UNIX: <strong>Meta-←</strong></td>
</tr>
<tr>
<td>Call Chain</td>
<td><img src="https://example.com/call_chain_icon.png" alt="Call Chain Icon" /></td>
<td>Lists the chain of callers from the top-level VI down to the opened subVI.</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>When you choose a VI from the ring control, the block diagram of the VI</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>opens. The ring control is in the toolbar when the subVI executing or running</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>is in Execution Highlighting on single-step mode. If a subVI has multiple</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>instances, you can observe which instance is executing.</td>
<td></td>
</tr>
</tbody>
</table>
### Block Diagram Navigation

<table>
<thead>
<tr>
<th>Description</th>
<th>Technique</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Create subVI from selected block diagram objects.</td>
<td>Select block diagram objects and select Edit/Create SubVI.</td>
</tr>
<tr>
<td>2. Create constant, control, or indicator on block diagram</td>
<td>Right-click terminal and select Create Constant, Control, or Indicator from the shortcut menu.</td>
</tr>
<tr>
<td>Create a constant on the block diagram</td>
<td>Drag front panel control to block diagram.</td>
</tr>
<tr>
<td>Create a control on the front panel</td>
<td>Drag block diagram constant to front panel.</td>
</tr>
<tr>
<td>3. List errors</td>
<td>Right-click broken wire and select List Errors from shortcut menu -or- click the Run arrow that appears broken.</td>
</tr>
</tbody>
</table>
| 4. Delete broken wires | Windows: Ctrl-b  
Macintosh: Command-b  
Sun: Meta-b  
UNIX: Meta-b  
HP-UX: Alt-b  
-or- select Edit/Remove Broken Wires. |

### Wire Types

<table>
<thead>
<tr>
<th>Scalars</th>
<th>1D Array</th>
<th>2D Array</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Control" /></td>
<td><img src="image" alt="Indicator" /></td>
<td><img src="image" alt="Control" /></td>
</tr>
<tr>
<td>Data Source</td>
<td>Data Display</td>
<td>Data Display</td>
</tr>
</tbody>
</table>

**NOTE:** Array wires are thicker than scalar wires, and array terminals have [] around base type

**Terminal Data Types** wire styles and colors are unique for each data type

<table>
<thead>
<tr>
<th>Signed Integers</th>
<th>8-bit</th>
<th>16-bit</th>
<th>32-bit</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="8-bit integer" /></td>
<td><img src="image" alt="16-bit integer" /></td>
<td><img src="image" alt="32-bit integer" /></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unsigned Integers</th>
<th>8-bit</th>
<th>16-bit</th>
<th>32-bit</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="8-bit integer" /></td>
<td><img src="image" alt="16-bit integer" /></td>
<td><img src="image" alt="32-bit integer" /></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Real Floating-Point</th>
<th>Single</th>
<th>Double</th>
<th>Extended</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Single precision" /></td>
<td><img src="image" alt="Double precision" /></td>
<td><img src="image" alt="Extended precision" /></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Complex Floating-Point</th>
<th>Single</th>
<th>Double</th>
<th>Extended</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Single precision" /></td>
<td><img src="image" alt="Double precision" /></td>
<td><img src="image" alt="Extended precision" /></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Boolean</th>
<th>String</th>
<th>Path</th>
<th>Variant</th>
<th>Waveform</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Boolean" /></td>
<td><img src="image" alt="String" /></td>
<td><img src="image" alt="Path" /></td>
<td><img src="image" alt="Variant" /></td>
<td><img src="image" alt="Waveform" /></td>
</tr>
</tbody>
</table>

**NOTE:** The Polymorphic Terminal Data Type represents a terminal to which multiple data types can be input or output.
**Keyboard Shortcuts**

**File**
- Ctrl-N: New VI (skips New dialog box)
- Ctrl-O: Opens file
- Ctrl-W: Closes file
- Ctrl-S: Saves VI
- Ctrl-P: Prints
- Ctrl-I: Displays VI properties
- Ctrl-Q: Quits LabVIEW

**Edit**
- Ctrl-V: Pastes object
- Ctrl-Shift-F: Displays search results
- Ctrl-B: Removes broken wires
- Ctrl-C: Copies an object
- Ctrl-D: Allows you to redraw (VI Hierarchy window only)
- Ctrl-F: Finds a terminal, local variable, reference, invoke node or property node
- Ctrl-X: Cuts object
- Ctrl-Z: Allows you to undo last action
- Ctrl-Shift-Z: Allows you to redo last action

**Tools**
- Ctrl-Y: Adds to VI Revision History

**Window**
- Ctrl-E: Displays block diagram/front panel
- Ctrl-L: Displays error list
- Ctrl-T: Tiles the block diagram and front panel windows
- Ctrl-/: Adjusts window to full size

**Help**
- Ctrl-H: Displays context help
- Ctrl-?: Displays help contents and index
- Ctrl-Shift-L: Locks context help

**Font**
- Ctrl-0: Displays Font dialog box
- Ctrl-1: Changes Application font
- Ctrl-2: Changes System font
- Ctrl-3: Changes Dialog font
- Ctrl-4: Changes Current font

**Operate**
- Ctrl-R: Runs VI
- Ctrl-M: Changes to run/edit mode
- Ctrl-.: Aborts VI

**Other Shortcuts**
- Ctrl-A: Adds a comment (VI Revision History window only)
- Shows all VIs (VI Hierarchy window only)
- Performs last alignment

---

**LabVIEW Web Resources**

**Support**
- www.ni.com/support
  - KnowledgeBase – searchable database of tips, common questions, and more
  - Troubleshooting Wizards
  - Application notes and white papers
  - Wishlist (online suggestions)

**Training**
- www.ni.com/custed
  - Course schedules, descriptions, and registration information
  - Self-paced training information

**Consulting**
- (Alliance Program Members)
  - www.ni.com/alliance

**Instrument Drivers**
- www.zone.ni.com/idnet

**Additional LabVIEW-Related Sites**
- www.vimarket.com
- www.ltrpub.com

**Developer Resources**
- zone.ni.com
  - Resource Library – example programs, technical presentations, and tutorials
  - Developer Exchange
  - Product Advisor
  - Measurement Glossary