|  |  |
| --- | --- |
| Faculty Member:\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| Semester:\_\_\_\_\_\_\_\_\_\_\_\_\_ | Section: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

Department of Electrical Engineering

EE 383: Instrumentation and Measurements

LAB 02: LabVIEW Programming Environment, Structures and Basic Operations

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Name** | **Reg. No** | **Conduct of**  **Experiment** | **Analysis of data in Lab Report** | **Modern Tool Usage** | **Ethics**  **and**  **Safety** | **Individual and Team Work** |
|  |  | **5 Marks** | **5 Marks** | **5 Marks** | **5 Marks** | **5 Marks** |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |

### Introduction

In this Lab, you will learn about the LabVIEW programming environment. You will also write a simple Virtual Instrument (VI) to incorporate basic operations and programming structures in LabVIEW. The structures featured include For Loops, While Loops, Case Structures, Sequence Structures, and Formula Nodes.

### Objectives

* Learn the three parts of a VI.
* Learn the three palettes.
* Learn how data is passed in LabVIEW.
* Distinguish between controls and indicators on the front panel and block diagram.
* Learn how the Case Structure executes.

### Theory

**LabVIEW Programming Basics**

**Introduction**

LabVIEW is a graphical programming language that uses icons instead of lines of text to create applications. In contrast to text-based programming languages, where instructions determine program execution, LabVIEW uses dataflow programming, where the flow of data determines execution.

In LabVIEW, you build a user interface by using a set of tools and objects.

The user interface is known as the front panel. You then add code using graphical representations of functions to control the front panel objects.

The block diagram contains this code. In some ways, the block diagram resembles a flowchart.

LabVIEW programs are called virtual instruments, or VIs, because their appearance and operation imitate physical instruments, such as oscilloscopes and multimeters. Every VI uses functions that manipulate input from the user interface or other sources and display that information or move it to other files or other computers.

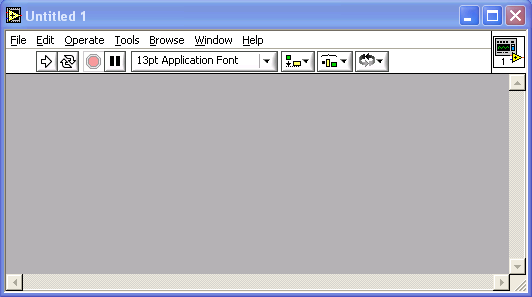
A VI contains the following three components:

• **Front panel**—Serves as the user interface.

• **Block diagram**—Contains the graphical source code that defines the functionality of the VI.

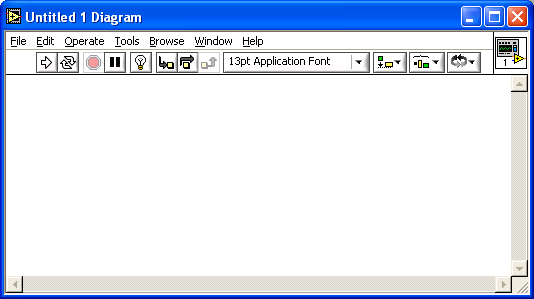
• **Icon and connector pane**—Identifies the VI so that you can use the VI in another VI. A VI within another VI is called a subVI. A subVI corresponds to a subroutine in text-based programming languages.

**Front Panel**



The front panel is the user interface of the VI. You build the front panel with controls and indicators, which are the interactive input and output terminals of the VI, respectively. Controls are knobs, pushbuttons, dials, and other input devices. Indicators are graphs, LEDs, and other displays. Controls simulate instrument input devices and supply data to the block diagram of the VI. Indicators simulate instrument output devices and display data the block diagram acquires or generates.

**Block Diagram**



After you build the front panel, you add code using graphical representations of functions to control the front panel objects. The block diagram contains this graphical source code. Front panel objects appear as terminals on the block diagram.

Additionally, the block diagram contains functions and structures from built-in LabVIEW VI libraries. Wires connect each of the nodes on the block diagram, including control and indicator terminals, functions, and structures.

**LabVIEW Palettes**

LabVIEW palettes give you the options you need to create and edit the front panel and block diagram.

The **Tools** palette is available on the front panel and the block diagram.

A tool is a special operating mode of the mouse cursor. When you select a tool, the cursor icon changes to the tool icon. Use the tools to operate and modify front panel and block diagram objects.

Select **Window»Show Tools Palette** to display the **Tools** palette. You can place the **Tools** palette anywhere on the screen.

If automatic tool selection is enabled and you move the cursor over objects on the front panel or block diagram, LabVIEW automatically selects the corresponding tool from the Tools palette.

The **Controls** palette is available only on the front panel. The **Controls** palette contains the controls and indicators you use to create the front panel. Select **Window»Show Controls Palette** or right-click the front panel workspace to display the **Controls** palette. You can place the **Controls** palette anywhere on the screen.

The **Functions** palette is available only on the block diagram. The **Functions** palette contains the VIs and functions you use to build the block diagram. Select **Window»Show Functions Palette** or right-click the block diagram workspace to display the **Functions** palette. You can place the **Functions** palette anywhere on the screen.

**Dataflow Programming**

LabVIEW follows a dataflow model for running VIs. A block diagram node executes when all its inputs are available. When a node completes execution, it supplies data to its output terminals and passes the output data to the next node in the dataflow path.

**Equations**

**Converting ºC to ºF**

The formula for converting degrees Celsius to degrees Fahrenheit is as follows:

°F = (1.8 \* °C) + 32

For example, to convert a Celsius temperature of 100 degrees into degrees Fahrenheit, first multiply the Celsius temperature reading by 1.8 to get 180. Then add 32 to 180 and get 212 degrees Fahrenheit.

**Slope of a Line**

The formula for the slope of a line is as follows:

Slope = (Y2 – Y1) / (X2 – X1)

where (X1, Y1) and (X2, Y2) are points on the line.

### Procedure

**Part 1. Converting ºC to ºF**

In Part 1, you will create a VI that can be used as a sub-VI.

1. Launch LabVIEW from **Start»Programs»National Instruments LabVIEW 6.1**. Click **New VI** to open a new front panel.

2. (Optional) Select **Window»Tile Left and Right** to display the front panel and block diagram side by side.

3. Create a numeric digital control. You will use this control to enter the value for degrees Centigrade.

a. Select the digital control on the **Controls»Numeric** palette. If the **Controls** palette is not visible, right-click an open area on the front panel to display it.

b. Move the control to the front panel and click to place the control.

c. Type deg C inside the label and click outside the label or click the **Enter** button on the toolbar, shown at left. If you do not type the name immediately, LabVIEW uses a default label. You can edit a label at any time by using the Labeling tool, shown at left.

4. Create a numeric digital indicator. You will use this indicator to display the value for degrees Fahrenheit.

a. Select the digital indicator on the **Controls»Numeric** palette.

b. Move the indicator to the front panel and click to place the indicator.

c. Type deg F inside the label and click outside the label or click the **Enter** button.



LabVIEW creates corresponding control and indicator terminals on the block diagram. The terminals represent the data type of the control or indicator. For example, a DBL terminal, shown at left, represents a double-precision, floating-point numeric control or indicator.

5. Display the block diagram by clicking it or by selecting **Window»**

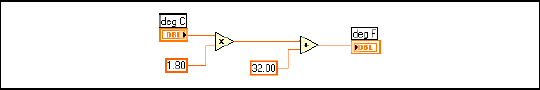
**Show Diagram**.

6. Select the Multiply and Add functions on the **Functions»Numeric** palette and place them on the block diagram. If the **Functions** palette isnot visible, right-click an open area on the block diagram to display it.

7. Select the numeric constant on the **Functions»Numeric** palette and place two of them on the block diagram. When you first place the numeric constant, it is highlighted so you can type a value.

8. Type 1.8 in one constant and 32.0 in the other.

If you moved the constants before you typed a value, use the Labeling tool to enter the values.



9. Use the Wiring tool to wire the icons as shown in the block diagram.

• To wire from one terminal to another, use the Wiring tool to click the first terminal, move the tool to the second terminal, and click the second terminal, as shown in the following illustration. You can start wiring at either terminal.

• You can bend a wire by clicking to tack the wire down and moving the cursor in a perpendicular direction. Press the spacebar to toggle the wire direction.

• To identify terminals on the nodes, right-click the Multiply and Add functions and select **Visible Items»Terminals** from the shortcut menu to display the connector pane. Return to the icons after wiring by right-clicking the functions and selecting **Visible Items»Terminals** from the shortcut menu to remove the checkmark.

• When you move the Wiring tool over a terminal, the terminal area blinks, indicating that clicking will connect the wire to that terminal and a tip strip appears, listing the name of the terminal.

• To cancel a wire you started, press the <Esc> key, right-click, or click the source terminal.

10. Display the front panel by clicking it or by selecting **Window»Show Panel**.

11. Save the VI because you will use this VI later in the course. Select **File»Save**. Type Convert C to F.vi in the dialog box. Click the **Save** button.

12. Enter a number in the digital control and run the VI.

a. Use the Operating tool or the Labeling tool to double-click the digital control and type a new number.

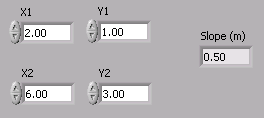
b. Click the **Run** button to run the VI.

c. Try several different numbers and run the VI again.

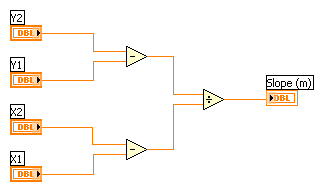
**Part 2. Calculating the Slope of a Line**

Using the techniques you learned in Part 1, create a VI to calculate the slope of a line. Then, transform the code into a subVI using the **Edit»Create SubVI** method. Save the VI as Slope.vi. Save the subVI as SlopeSub.vi. The equation for slope is given in the Theory section.

Open a new VI and complete the front panel and block diagram as follows:

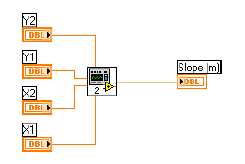


Front Panel



Block Diagram before subVI

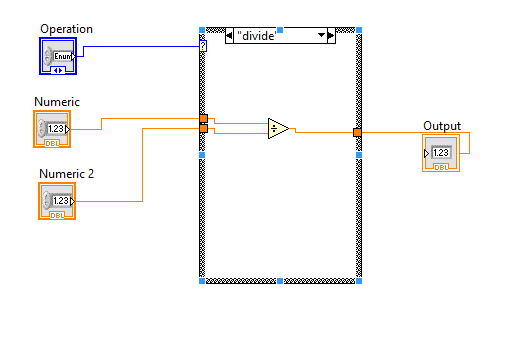
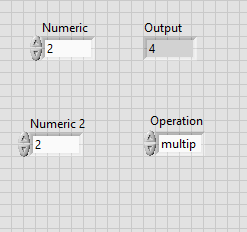
After creating the subVI, your block diagram will resemble the one below.



Block Diagram with subVI

**Part 3. Calculator VI**

Write a VI that adds, subtracts, multiplies, divides, and averages two input numbers and displays the results on the front panel. Save your VI as Calculator.vi.



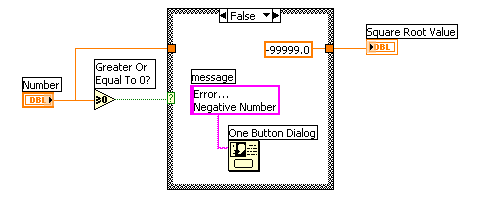
### Square Root VI

Complete the following steps to build a VI that checks whether a number is positive. If it is, the VI calculates the square root of the number. Otherwise, the VI returns an error message.

1. Open a new VI and build the following front panel.



2. Build the following block diagram.



a. Place a Case structure located on the **Functions»Structures** palette.

b. Click the decrement or increment arrow button to select the FALSE case.

c. Place the Greater or Equal to 0? function located on the **Functions»Comparison** palette. This function returns TRUE if **Number** isgreater than or equal to 0.

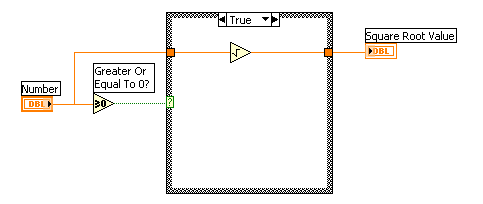
d. Right-click the numeric constant and select **Format & Precision** from the shortcut menu. Set **Digits of Precision** to 1, select **Floating Point Notation**, and click the **OK** button. This ensures there is nodata conversion between the constant and the numeric indicatoroutside the Case structure.

e. Place the One Button Dialog function located on the **Functions»Time & Dialog** palette. This function displays a dialog box thatwill contain the message Error...Negative Number.

f. Right-click the **message** terminal of the One Button Dialog function, select **Create»Constant** from the shortcut menu, type

Error...Negative Number, and press the <Enter> key.

g. Select the TRUE case and place the Square Root function located on the **Functions»Numeric** palette, as shown in the following block diagram. This function returns the square root of **Number**.



3. Save the VI as Square Root.vi.

4. Display the front panel and run the VI.

If **Number** is positive, the VI executes the TRUE case and returns the square root of **Number**. If **Number** is negative, the VI executes the FALSE case, returns –99999.0, and displays a dialog box with the message Error...Negative Number.

5. Close the VI.

**Questions**

1. What is a VI? What are the three main parts of a VI? Briefly describe each.

2. What are the three palettes? Briefly describe each.

3. How is data passed in LabVIEW?

4. How can you tell the difference between controls and indicators on the front panel? On the block diagram?

5. How does a Case Structure execute?

### References

* Getting Started with LabVIEW. November 2001. Part Number 321527E-01. <http://www.ni.com/manuals>.
* LabVIEW User’s Manual. November 2001. Part Number 320999D-01. <http://www.ni.com/manuals>.
* LabVIEW Student Edition. <http://www.ni.com/labviewse>.
* [LabVIEW Introduction Course - Six Hours](http://zone.ni.com/devzone/learningcenter.nsf/03f7c60f17aad210862567a90054a26c/55974411828f779086256ce9007504bd?OpenDocument).
* [LabVIEW Introduction Course - Three Hours](http://zone.ni.com/devzone/learningcenter.nsf/03f7c60f17aad210862567a90054a26c/60c2782788a811c986256cd50001a0a6?OpenDocument).