

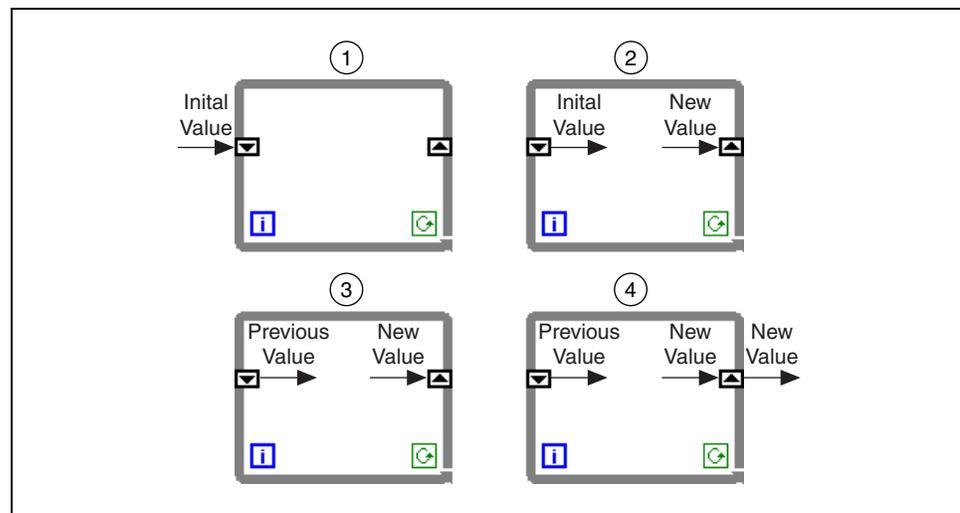
C. Shift Registers

Use shift registers on For Loops and While Loops to transfer values from one loop iteration to the next. Create a shift register by right-clicking the left or right border of a loop and selecting **Add Shift Register** from the shortcut menu.



A shift register appears as a pair of terminals, shown at left, directly opposite each other on the vertical sides of the loop border. The right terminal contains an up arrow and stores data on the completion of an iteration. LabVIEW transfers the data connected to the right side of the register to the next iteration.

A shift register transfers any data type and automatically changes to the data type of the first object wired to the shift register. The data you wire to the terminals of each shift register must be the same type. You can create multiple shift registers on a structure, and you can have more than one left terminal to remember more than one previous value, as shown in the following illustration.

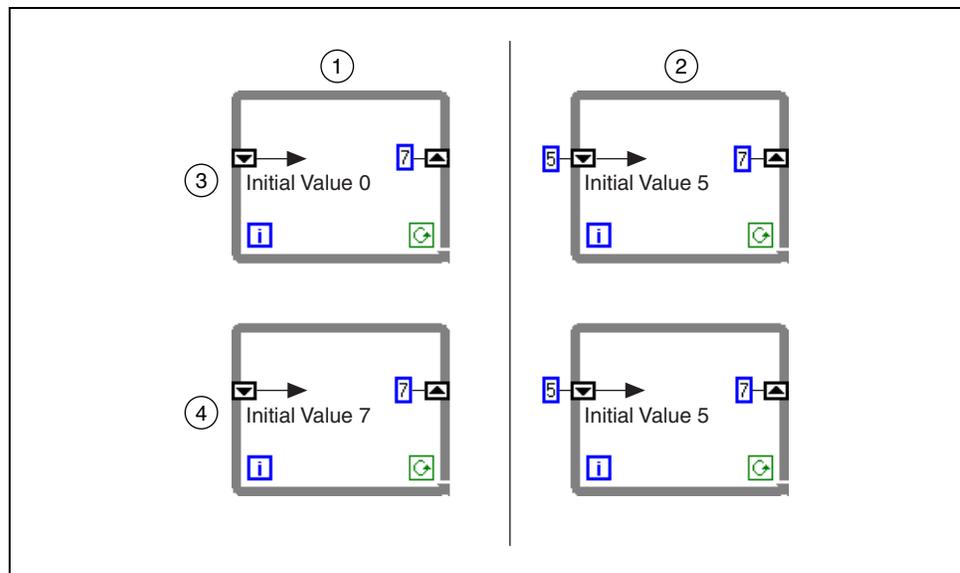


You can use shift registers to remember values from previous iterations. This technique is useful for averaging data points. To configure a shift register to carry over values to the next iteration, right-click the left terminal and select **Add Element** from the shortcut menu. For example, if you add two more elements to the left terminal, values from the last three iterations carry over to the next iteration.

Initializing Shift Registers

To initialize a shift register, wire any value from outside the loop to the left terminal. If you do not initialize the register, the loop uses the value written to the register when the loop last executed or the default value for the data type if the loop has never executed. For example, if the shift register data type is Boolean, the initial value is FALSE. Similarly, if the shift register data type is numeric, the initial value is 0.

Use a loop with an uninitialized shift register to run a VI repeatedly so that each time the VI runs, the initial output of the shift register is the last value from the previous execution. Leave the input to the left shift register terminal unwired for an uninitialized shift register to preserve state information between subsequent executions of a VI.



Exercise 4-5 Temperature Running Average VI

Objective: To use shift registers to perform a running average.

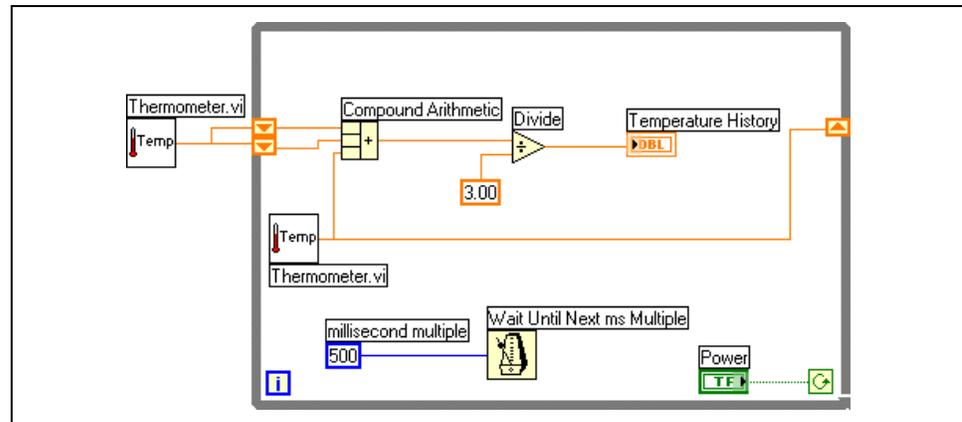
Complete the following steps to modify the Temperature Monitor VI to average the last three temperature measurements and display the average on a waveform chart.

Front Panel

1. Open the Temperature Monitor VI, which you built in Exercise 4-1.
2. Select **File»Save As** and rename the VI Temperature Running Average.vi.

Block Diagram

3. Display the block diagram.
4. Right-click the right or left border of the While Loop and select **Add Shift Register** from the shortcut menu to create a shift register.
5. Right-click the left terminal of the shift register and select **Add Element** from the shortcut menu to add an element to the shift register.
6. Modify the block diagram as follows.



- a. Press the <Ctrl> key while you click the Thermometer VI and drag it outside the While Loop to create a copy of the subVI.

(MacOS) Press the <Option> key. **(Sun)** Press the <Meta> key. **(Linux)** Press the <Alt> key.

The Thermometer VI returns one temperature measurement from the temperature sensor and initializes the left shift registers before the loop starts.



b. Place the Compound Arithmetic function located on the **Functions»Numeric** palette. This function returns the sum of the current temperature and the two previous temperature readings. Use the Positioning tool to resize the function to have three left terminals, shown at left.



c. Place the Divide function located on the **Functions»Numeric** palette. This function returns the average of the last three temperature readings.



d. Right-click the y terminal of the Divide function, select **Create»Constant**, type 3, and press the <Enter> key.



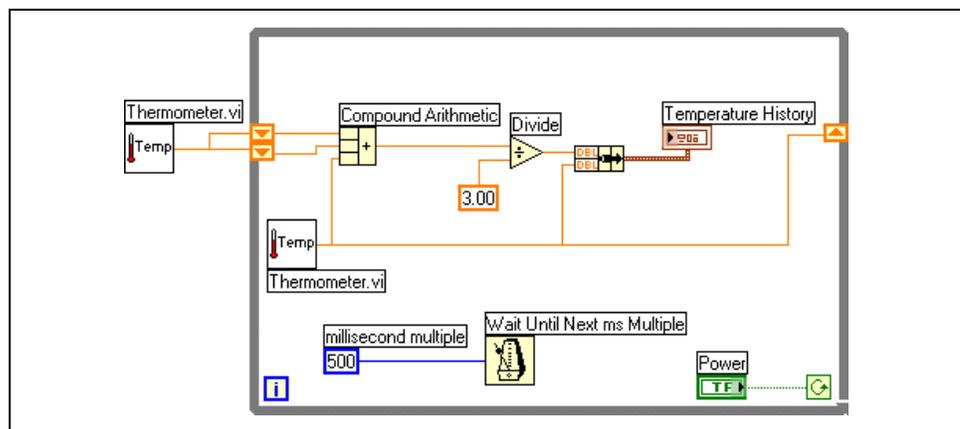
e. Use the Labeling tool to double-click the numeric constant wired to the Wait Until Next ms Multiple function, type 500, and press the <Enter> key.

7. Save the VI because you will use this VI later in the course.
8. Run the VI.

During each iteration of the While Loop, the Thermometer VI takes one temperature measurement. The VI adds this value to the last two measurements stored in the left terminals of the shift register. The VI divides the result by three to find the average of the three measurements, the current measurement plus the previous two. The VI displays the average on the waveform chart. Notice that the VI initializes the shift register with a temperature measurement.



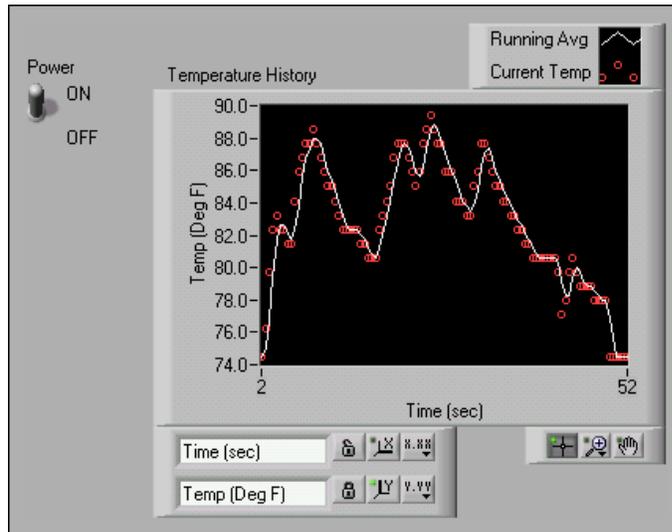
9. Place the Bundle function located on the **Functions»Cluster** palette to display both the average and the current temperature measurement on the same waveform chart. This function bundles the average and current temperature for plotting on the waveform chart.



10. Save and run the VI. The VI displays two plots on the waveform chart. The plots are overlaid. That is, they share the same vertical scale.

Customizing Charts

Complete the following steps to customize the waveform chart as shown in the following front panel. You can display a plot legend, a scale legend, a graph palette, a digital display, a scrollbar, and a buffer. By default, a waveform chart displays the plot legend.



11. Customize the y-axis.



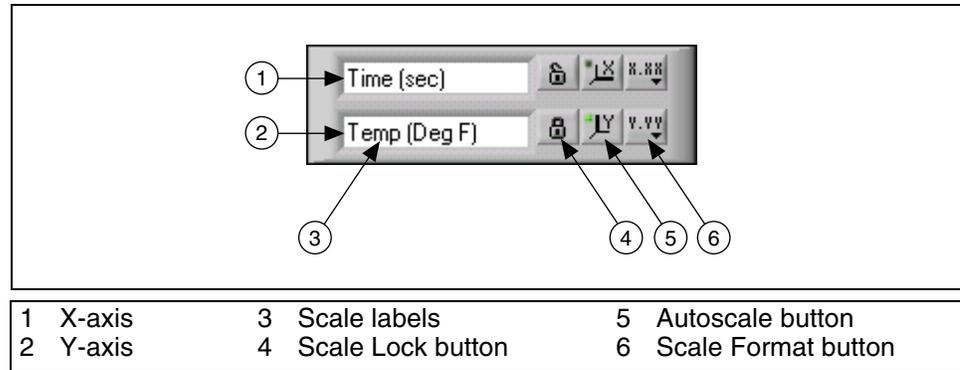
- Use the Labeling tool to double-click 70.0 in the y-axis, type 75.0, and press the <Enter> key.
- Use the Labeling tool to double-click the second number from the bottom on the y-axis, type 80.0, and press the <Enter> key. This number determines the numerical spacing of the y-axis divisions.

For example, if the number above 75.0 is 77.5, indicating a y-axis division of 2.5, changing the 77.5 to 80.0 reformats the y-axis to multiples of 5.0 (75.0, 80.0, 85.0, and so on).



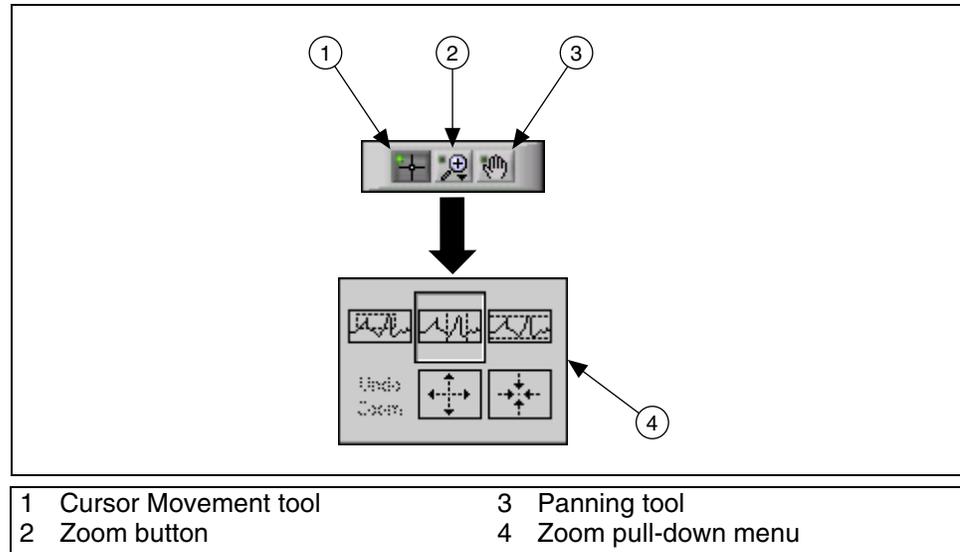
Note The waveform chart size has a direct effect on the display of axis scales. Increase the waveform chart size if you encounter problems while customizing the axis.

12. Right-click the waveform chart and select **Visible Items»Scale Legend** from the shortcut menu to display the scale legend, as shown in the following illustration. You can place the scale legend anywhere on the front panel.



13. Use the scale legend to customize each axis.
- Make sure the **Autoscale** LED is green and that the **Lock Autoscale** button appears locked so the y-axis adjusts the minimum and maximum values to fit the data in the chart.
 - Click the **Scale Format** button to change the format, precision, mapping mode, scale visibility, and grid options for each axis.
14. Use the plot legend to customize the plots.
- Use the Positioning tool to resize the plot legend to include two plots.
 - Use the Labeling tool to change Temp to Running Avg and to change Plot 1 to Current Temp. If the text does not fit, use the Positioning tool to resize the plot legend.
 - Right-click the plot in the plot legend to set the line and point styles and the color of the plot background or traces.

15. Right-click the waveform chart and select **Visible Items»Graph Palette** from the shortcut menu to display the graph palette, as shown in the following illustration. You can place the graph palette anywhere on the front panel.



Use the **Zoom** button to zoom in or out of sections of the chart or the whole chart. Use the **Panning** tool to pick up the plot and move it around on the display. Use the **Cursor Movement** tool to move the cursor on the graph.

16. Save and run the VI. While the VI runs, use the buttons in the scale legend and graph palette to modify the waveform chart.



Note If you modify the axes labels, the display might become larger than the maximum size that the VI can correctly present.

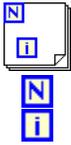
17. Use the Operating tool to click the **Power** switch and stop the VI.
18. Save and close the VI.

End of Exercise 4-5

Challenges:

1. Add the ability to output the mean temperature and standard deviation to appropriate indicators on the Front Panel after the monitor runs.
2. Add the ability to determine the maximum and minimum temperatures reached during running of the temperature monitor. Output these to appropriate indicators on the Front Panel.
3. Add the ability to write the temperature data to a spreadsheet file. Verify that the data was successfully written to the spreadsheet file by creating a VI to read the spreadsheet file and plot the data in a graph on the Front Panel.

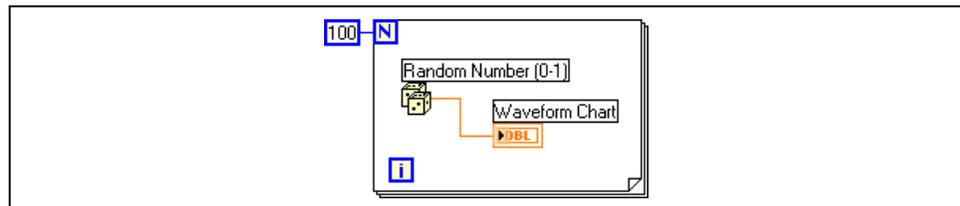
D. For Loops



A For Loop, shown at left, executes a subdiagram a set number of times. The For Loop is located on the **Functions»Structures** palette. The value in the count terminal (an input terminal), shown at left, indicates how many times to repeat the subdiagram. The iteration terminal (an output terminal), shown at left, contains the number of completed iterations. The iteration count always starts at zero. During the first iteration, the iteration terminal returns 0.

The For Loop differs from the While Loop in that the For Loop executes a set number of times. A While Loop stops executing the subdiagram only if the value at the conditional terminal exists.

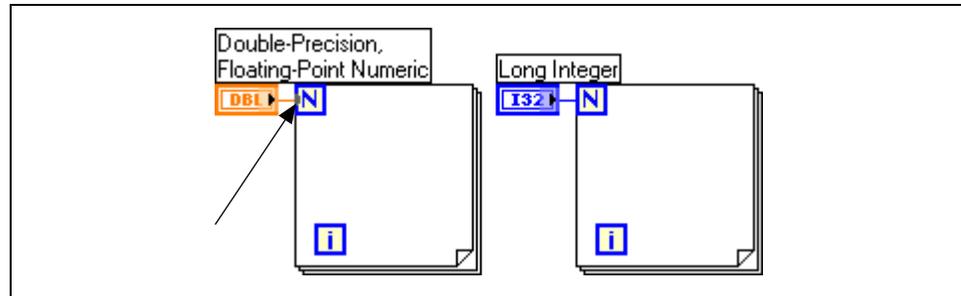
The following For Loop generates 100 random numbers and displays the points on a waveform chart.



Numeric Conversion

LabVIEW can represent numeric data types as integers (byte, word, or long), floating-point numerics (single, double, or extended precision), or complex numerics (single, double, or extended precision). When you wire two or more numeric inputs of different representations to a function, the function usually returns output in the larger or wider format. The functions coerce the smaller representations to the widest representation before execution, and LabVIEW places a coercion dot on the terminal where the conversion takes place.

For example, the For Loop count terminal is long integer. If you wire a double-precision, floating-point numeric to the count terminal, LabVIEW converts the numeric to a long integer. A gray coercion dot appears on the count terminal of the first For Loop, as shown in the following illustration.



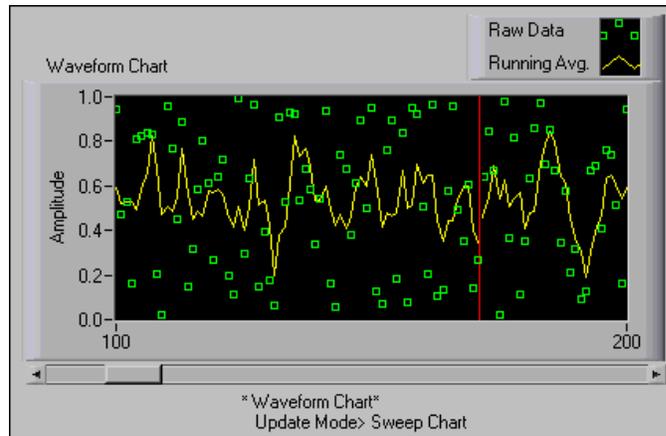
To change the representation of a numeric object, right-click the object and select **Representation** from the shortcut menu. Select the data type that best represents your data.

When LabVIEW converts floating-point numerics to integers, it rounds to the nearest integer. LabVIEW rounds $x.5$ to the nearest even integer. For example, LabVIEW rounds 2.5 to 2 and 3.5 to 4.

Exercise 4-6 Random Average VI

Objective: To build a VI that displays two plots, a random plot and a running average of the last four points, on a waveform chart in sweep update mode.

1. Build this VI, using the following tips:
 - Use a For Loop ($n = 200$) instead of a While Loop. The sweep chart should be similar to the following chart.



- Use a shift register with three left terminals to average the last four data points.
 - Use the Random Number (0-1) function located on the **Functions»Numeric** palette to generate the data.
 - Use the Bundle function located on the **Functions»Cluster** palette to group the random data with the averaged data before plotting.
2. Save the VI and name it `Random Average.vi`.
 3. Close the VI.

End of Exercise 4-6

Summary, Tips, and Tricks

- Use structures on the block diagram to repeat blocks of code and to execute code conditionally or in a specific order.
- The While Loop executes the subdiagram until the conditional terminal receives a specific Boolean value. By default, the While Loop executes its subdiagram until the conditional terminal receives a FALSE value.
- The For Loop executes a subdiagram a set number of times.
- You create loops by using the cursor to drag a selection rectangle around the section of the block diagram you want to repeat or by dragging and dropping block diagram objects inside the loop.
- The Wait Until Next ms Multiple function makes sure that each iteration occurs at certain intervals. Use this function to add timing to loops.
- The waveform chart is a special numeric indicator that displays one or more plots.
- The waveform chart has the following three update modes:
 - A strip chart shows running data continuously scrolling from left to right across the chart.
 - A scope chart shows one item of data, such as a pulse or wave, scrolling partway across the chart from left to the right.
 - A sweep display is similar to an EKG display. A sweep works similarly to a scope except it shows the old data on the right and the new data on the left separated by a vertical line.
- Use shift registers on For Loops and While Loops to transfer values from one loop iteration to the next.
- Create a shift register by right-clicking the left or right border of a loop and selecting **Add Shift Register** from the shortcut menu.
- To configure a shift register to carry over values to the next iteration, right-click the left terminal and select **Add Element** from the shortcut menu.
- Right-click a waveform chart or its components to set attributes of the chart and its plots.
- Coercion dots appear where LabVIEW coerces a numeric representation of one terminal to match the numeric representation of another terminal.