

Timing Functions

Goal

Use the Tick Count and Wait timing functions and use shift registers on While Loops to store data between iterations.

Scenario

Create a VI that will use the Tick function and a shift register to measure the time between iterations of a While Loop as well as the time since the VI began.

Design

The finished VI is shown in Figure 1 and Figure 2.

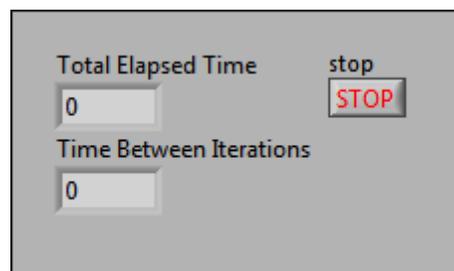


Figure 1. Elapsed Time VI Front Panel

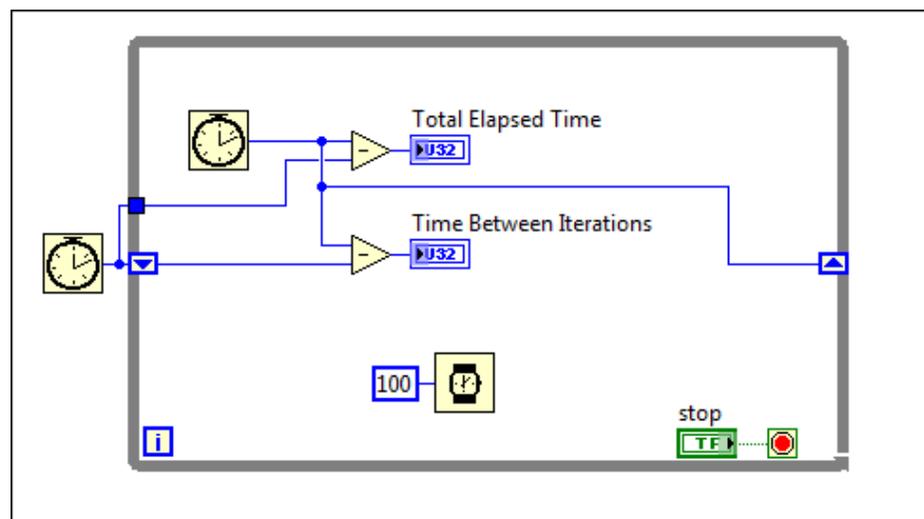


Figure 2. Elapsed Time VI Block Diagram

Implementation

The files that you need to complete this exercise are here:

<NI eLearning>\LV Core 1\Timing Functions\Exercise.

1. Open a blank VI.
2. Save the VI as `Elapsed Time.vi` in the <Exercise> directory.
3. Add two Numeric Indicators to the front panel, as shown in Figure 1.
 - Rename the first indicator `Total Elapsed Time` and the second indicator `Time Between Iterations`.
 - Right-click the indicators and select **Representation»U32**.
4. Place a While Loop.
 - Switch to the block diagram.
 - Add a **While Loop** to the block diagram containing the indicators.
5. Set up the loop to calculate the total elapsed time and the time between iterations.
 - Add two **Tick Count (ms)** functions to the block diagram.
 - Place the first Tick Count function outside the loop and the second Tick Count function inside the loop.
 - Add two **Subtract** functions inside the loop, one above the other, as shown in Figure 2.
 - Add a shift register to the loop by right-clicking the edge and selecting **Add Shift Register**.
 - Wire the Tick Count function located outside the loop to the left shift register and the y input of the top Subtract function.
 - Wire the left shift register to the y input of the bottom Subtract function.
 - Wire the Tick Count function located inside the loop to the right shift register and the x input of both Subtract functions.



- Wire the output of the top Subtract function to the Total Elapsed Time numeric indicator.
 - Wire the output of the bottom Subtract function to the Time Between Iterations numeric indicator.
6. Set up the end condition to finish when the stop button is pressed.
 - Right-click the condition terminal and select **Create»Control**.
 - On the front panel, move the stop Boolean control to the right of the indicators.
 7. Place a Wait (ms) function inside the loop to control the rate at which the loop will run.



- Add a **Wait (ms)** function inside the loop.
- Right-click the milliseconds to wait input and select **Create»Constant**.
- Change the value of the constant to 100.

Test

1. Run the VI. Notice that the Total Elapsed Time is increasing in increments of 100.
2. Click the **Stop** button.



3. On the block diagram, right-click the Wait (ms) function and select **Replace»Timing Palette»Wait Until Next ms Multiple**. This loop will now run synchronized with the millisecond timer.
4. Run the VI. Notice that the Total Elapsed Time is increasing in increments of 100, with a small offset which is constant.
5. Save the VI and close it when you are finished.

End of Exercise

Notes
