

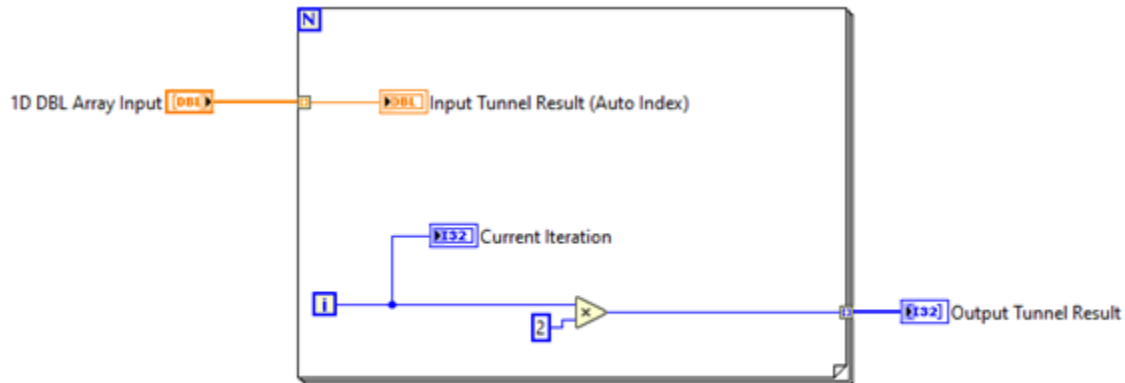
Exercise 8-6: Exploring Auto-Indexing Tunnels

Goal

- Explore the behaviour of auto-indexing input and output tunnels.

Instructions

1. Open `C:\Exercises\LabVIEW Core 1\Auto-Indexing\Auto-Indexing.lvproj`.
2. From the **Project Explorer** window, open the Auto-Indexing Tunnels VI.
3. Modify the block diagram, as shown in the following figure, to create an auto-indexing input tunnel and auto-indexing output tunnel to the For Loop.



- Wire the **1D DBL Array Input** control to the left border of the For Loop, which by default creates an auto-indexing input tunnel. Then, wire the input tunnel to the **Input Tunnel Result** indicator.



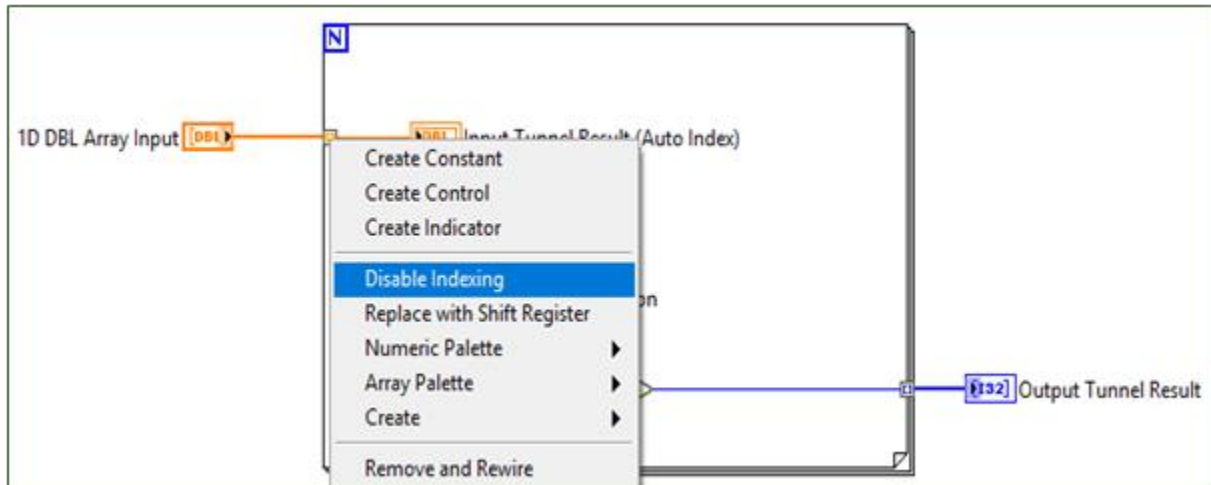
Note: If you wire an array to a For Loop or While Loop, you can read and process every element in that array by enabling indexing. When you wire an array from an external node to an input tunnel on the loop border and enable indexing on the input tunnel, elements of that array enter the loop one at a time, starting with the first element.

- Wire the output of the Multiply function to the right border of the For Loop, which by default creates an indexing output tunnel. Then, wire the output tunnel to the **Output Tunnel Result** indicator.

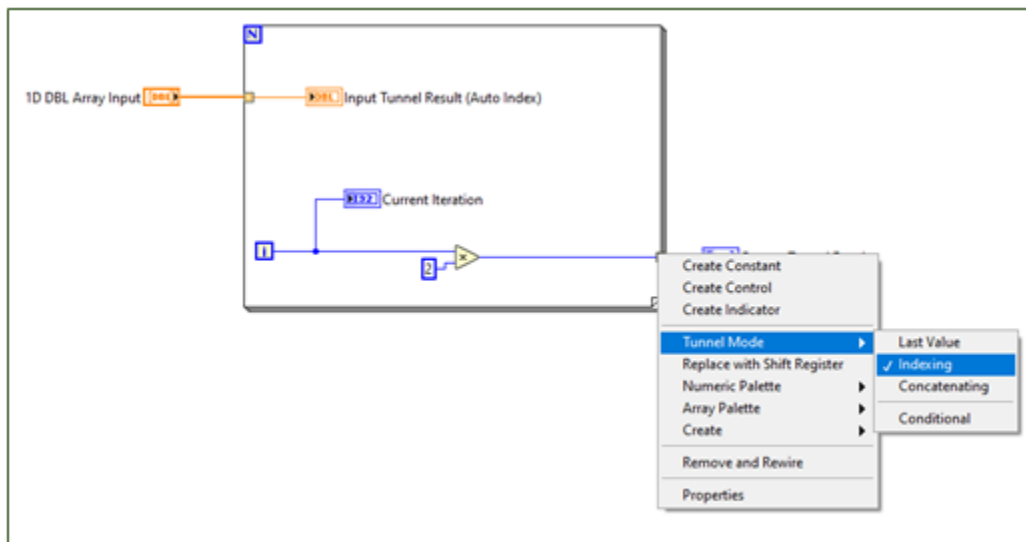


Note: For indexing output tunnels, scalar elements accumulate sequentially into 1D arrays, 1D arrays accumulate into 2D arrays, and so on.

4. Right-click the **indexing input tunnel**. In the appeared menu you can set the behaviour of any loop input tunnel to enable or disable indexing.



5. Right-click the **indexing output tunnel**. In the **Tunnel Mode** section, notice that **Indexing** is selected. This is where you can set the behaviour of any loop output tunnel to **Last Value**, **Indexing**, or **Concatenating**.



6. Turn on Highlight Execution.

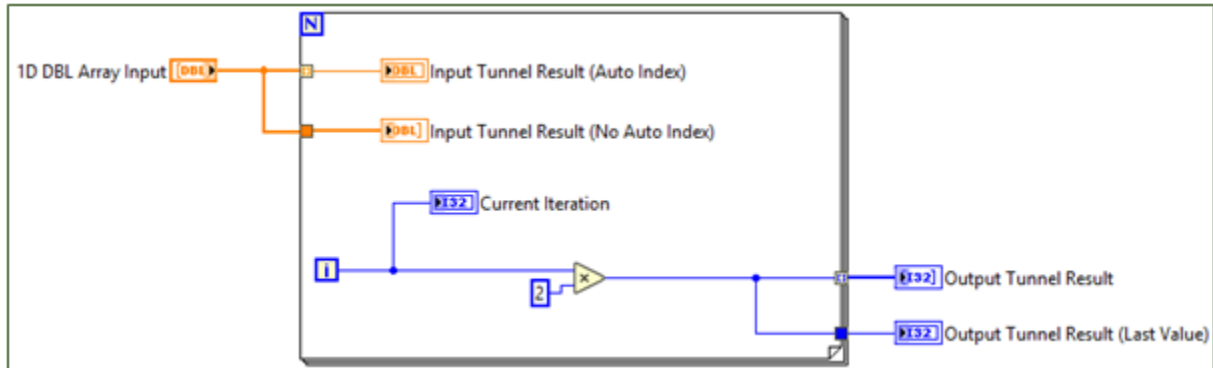
7. Run the VI.

- On the front panel, observe the behavior of each control/indicator.
- On the block diagram, observe how elements of the 1D DBL Array enter the loop one at a time, starting with the first element.
- Observe how the **Output Tunnel Result** indicator is a 1D array that consists of every result passed into the indexing output tunnel while the For Loop executed.
- Notice how the For Loop output tunnel does not return anything until the entire For Loop has finished executing.
- Run the VI multiple times with Highlight Execution turned on to make all the above observations.
- Stop the VI when finished.

8. Save the VI.

Comparing the Behavior of Auto-Indexing vs. Keeping Last Value

1. Open Compare Auto-Index and Last Value VI.
2. Explore the front panel and the block diagram.
 - Notice that for each For Loop input tunnel and output tunnel, there is an indexing indicator and a last value indicator.
 - On the block diagram, select each **tunnel**. Notice how each tunnel is configured.



- Turn on Highlight Execution.
3. Run the VI.
 - Observe the differences in data type and behavior for the input tunnels.
 - Observe the differences in data type and behavior for the output tunnels.
 - Stop the VI when finished.
 4. Save the VI.

End of Exercise 8-6