**Windows vs. Real-Time OS Jitter Benchmarking Demo for Any Real-Time Target**

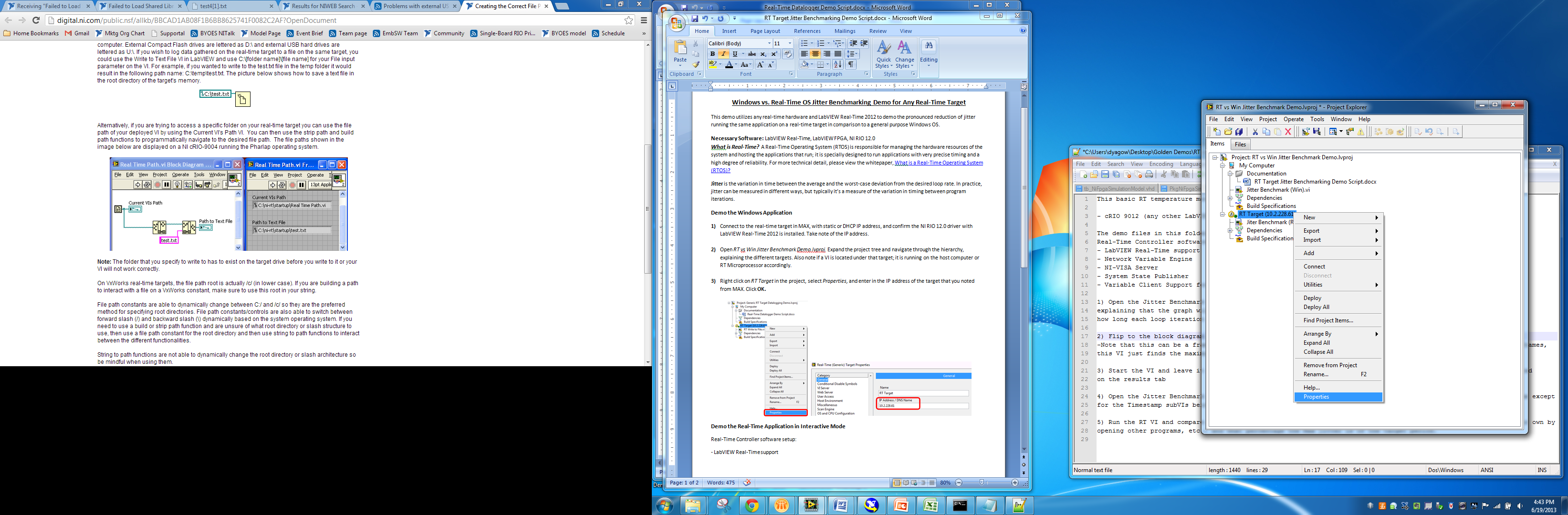
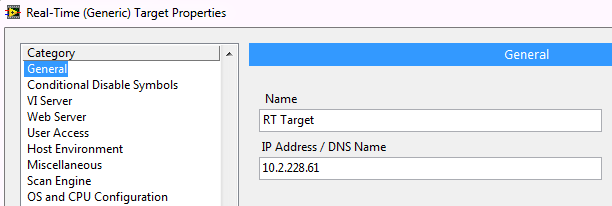
This demo utilizes any real-time hardware and LabVIEW Real-Time 2013 to demo the pronounced reduction of jitter running the same application on a real-time target in comparison to a general purpose Windows OS.

**Necessary Host Software:** LabVIEW Real-Time, LabVIEW FPGA, NI RIO 13.0   
**Necessary Target Software:** Recommended Software Set is sufficient  
 ***What is Real-Time?*** A Real-Time Operating System (RTOS) is responsible for managing the hardware resources of the system and hosting the applications that run; it is specially designed to run applications with very precise timing and a high degree of reliability. For more technical detail, please view the whitepaper, [What is a Real-Time Operating System (RTOS)?](http://www.ni.com/white-paper/3938/en)

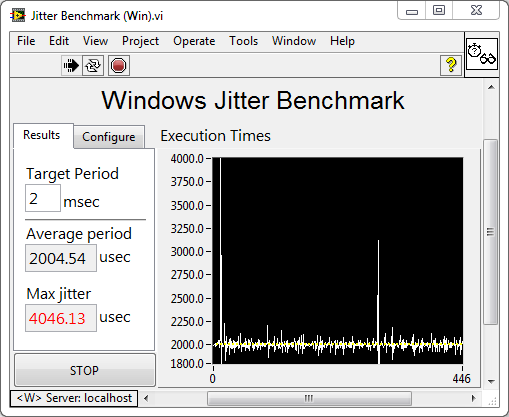
**Jitter** is the variation in time between the average and the worst-case deviation from the desired loop rate. In practice, jitter can be measured in different ways, but typically it’s a measure of the variation in timing between program iterations.

**Demo the Windows Application**

1. Connect to the real-time target in MAX, with static or DHCP IP address, and confirm the NI RIO 12.0 driver with LabVIEW Real-Time 2012 is installed. Take note of the IP address.
2. Open *RT vs Win Jitter Benchmark Demo.lvproj.* Expand the project tree and navigate through the hierarchy, explaining the different targets. Also note if a VI is located under that target; it is running on the host computer or RT Microprocessor accordingly.
3. Right click on *RT Target* in the project, select *Properties,* and enter in the IP address of the target that you noted from MAX. Click **OK.**

* *

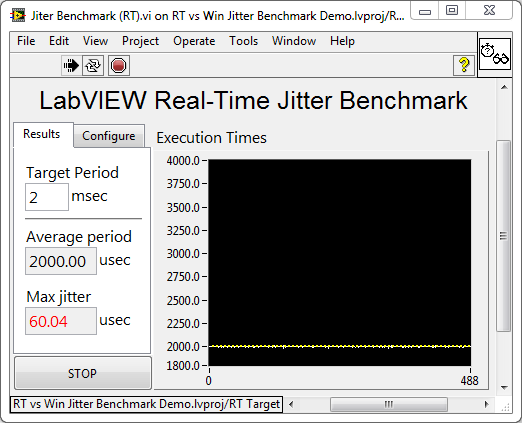
1. Open and Run *Jitter Benchmark (Win).vi*. This vi benchmarks how long each loop iteration running on your PC takes. The target period is 2msec, but Windows in practice can have unbounded jitter. The **Average Period** indicator shows the average loop iteration time in microseconds over the length of the running VI. The **Max jitter** indicator stores the loop iteration that experienced the worst jitter. If you open/close applications or even move the VI front panel window, the VI will experience high jitter, often multiple msec off from the target period, as shown below:

****

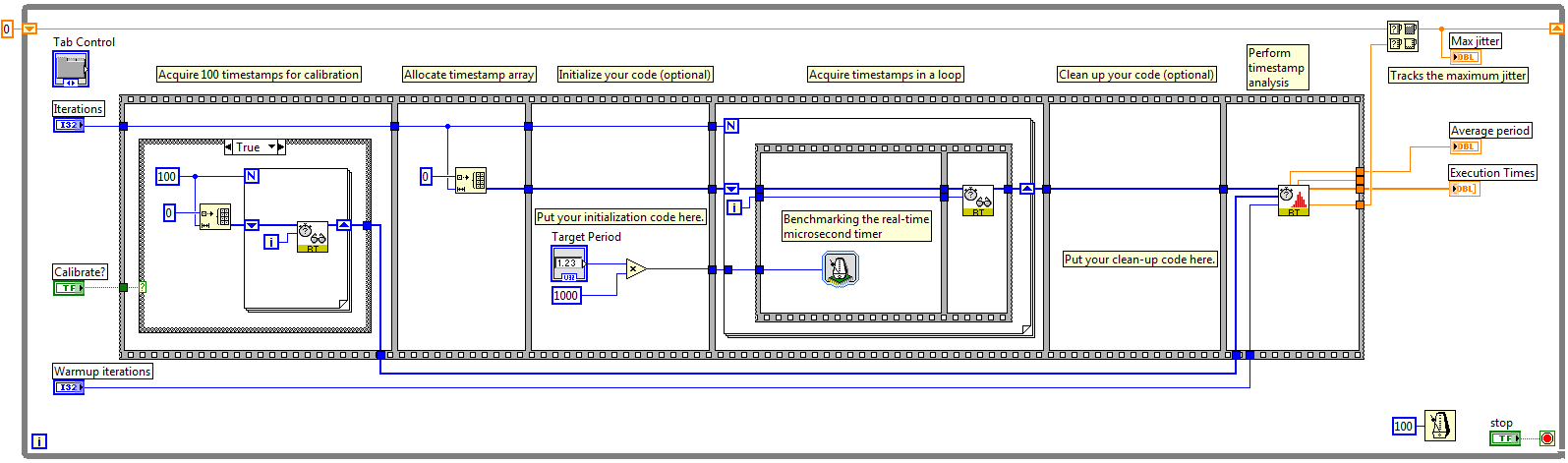
1. Press the Stop button and switch to the block diagram and walk through the benchmark code frame by frame as directed by the comments. Note that this can be a framework for customers to benchmark their own code, which would be placed in the 3rd, 4th, and 5th frames. This VI is currently set up to find the maximum jitter of running the VI itself at the target period.

**Demo the Real-Time Application in Interactive Mode**

1. Open the Jitter Benchmark (RT).vi and repeat steps 4 and 5, now contrasting the jitter experienced on the real-time system. There are no peripherals or other programs that we can modify. In time critical systems where one loop exceeding the specified jitter would be catastrophic, such period consistency is necessary.



1. Again switch to the block diagram and walk through the benchmark code frame by frame as directed by the comments. Customers can find this code to benchmark their own applications in the LabVIEW Example Finder by navigating to **Help>>Find Examples** and then searching for **Benchmark Project.lvproj** (Toolkits and Modules>>Real-Time>>Benchmarking).



Note that the code is the same, except that the Timestamp subVIs are replaced with their RTOS-specific counterparts from the RT Utilities palette.