Queues in LabVIEW*

Felix Annan

Abstract

This describes the function of Queues in transferring data between virtual instruments in National Instruments LabVIEW

Queues, as the name suggests, queue data. This data is stored in a buffer in memory until it is read from the queue. There are generally two forms of queues available, there is the First In First Out (FIFO) queue and there is also the Last In First Out (LIFO) queue. Most of the time when you stand in a queue to receive a service or purchase something, that is a FIFO queue, the first person to get there is the first person served. LIFO queues generally do not occur often in social queue settings. When you stack tennis balls in a tube, the last ball in is the first ball you can get out, or when you stack books in a box or equipment in a van, generally the last bit to go in is the first bit that can be removed. Items can be previewed(read) from the queue without being removed from the queue or "Dequeued". However, once dequeued the data item will no more be available on the queue. Adding items to a queue is called 'Enqueuing'. The end to which data items are enqueued determines whether the queue is a LIFO queue or a FIFO queue

The Advantage:

1. They buffer and store all information written to them until it is read so nothing is lost.
2. They can be created dynamically by changing the queue name

The Disadvantage:

1. Queues must be polled continually to determine when data has arrived except when combined with a notifier

Generally a queue is very well suited for data streams like that of a waveform, especially data for which loss of information would be very undesirable.

Implementation:

A typical setup for reading from and writing to a queue is indicated in the Figure 1.

*Version 1.1: Aug 4, 2006 2:50 am GMT-5

†http://creativecommons.org/licenses/by/2.0/

Queues

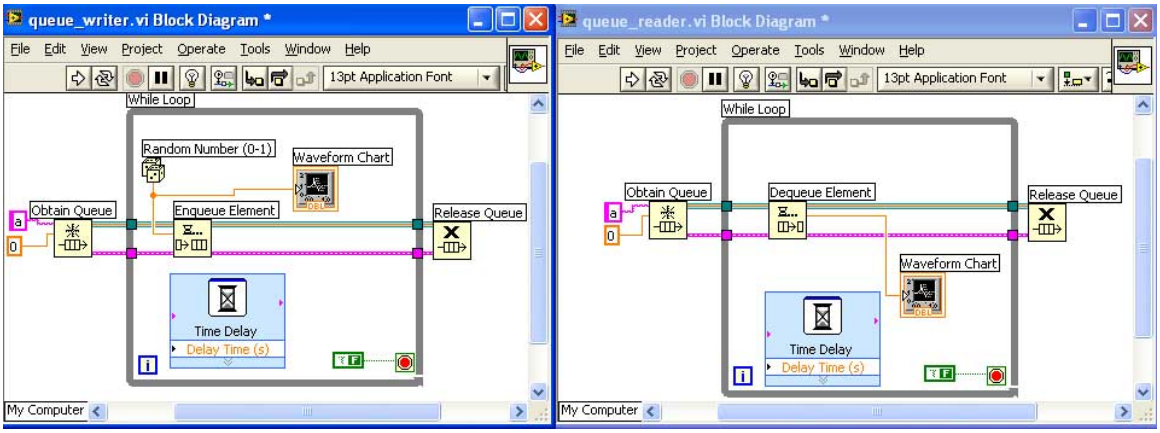

Figure 1: The VI on the left writes to a queue and the VI on the right reads from the queue