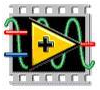
LabVIEW – Synchronisation

Gary Boorman

Queues and Notifiers

- A **Queue** is used to communicate data between sections of the Diagram or between other VIs
- A **Notifier** is similar to a Queue, but can only communicate one piece of data at a time
- **Queues** are equivalent to **FIFOs**
- **Notifiers** equivalent to single-element FIFO

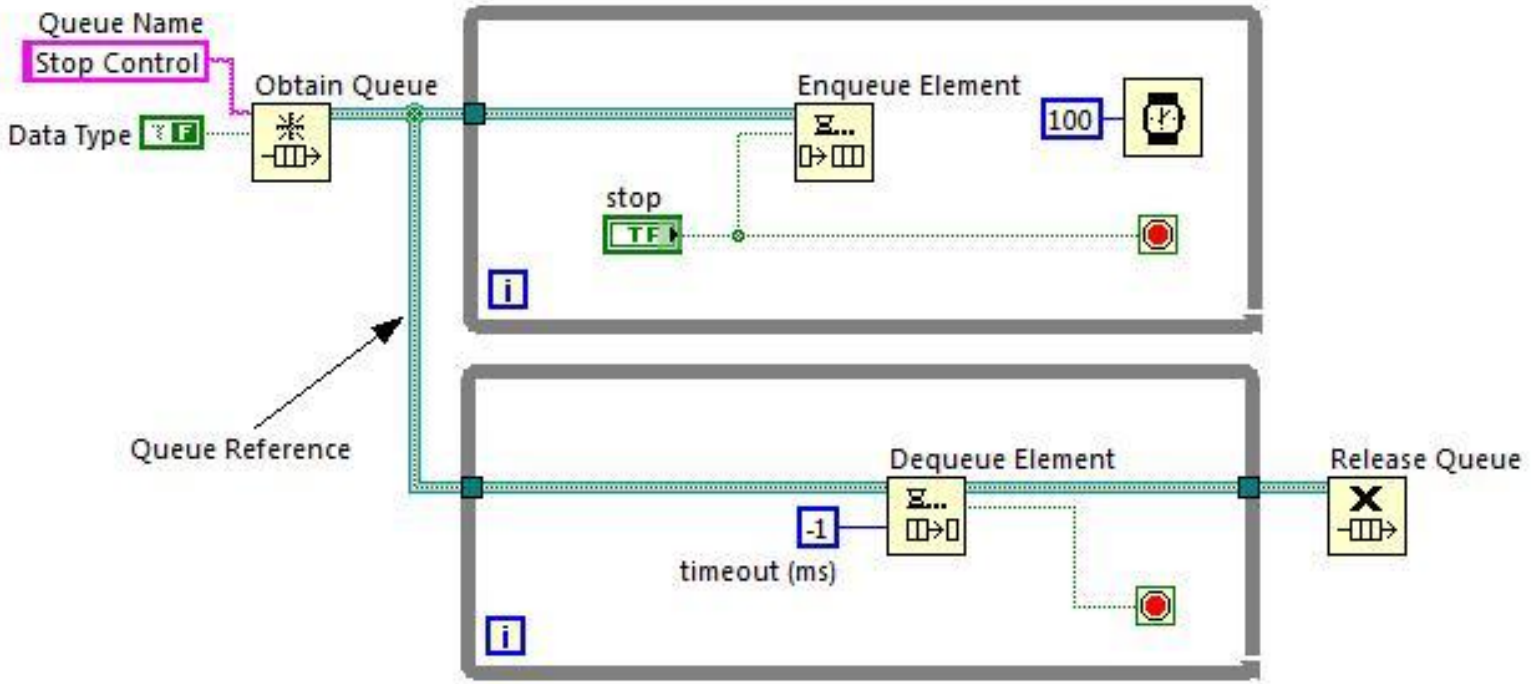


Queues I

- A Queue has to be *obtained* before it can be used. A Queue must have a *Data-type*, with an optional name and size
- A reference is used to access the Queue – eg to place data into (*enqueue*), read data from *(dequeue*), empty (*flush*), preview etc
- A Queue must be *released* when finished with

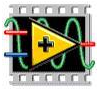
Queues II

Example of using a Queue to communicate between loops in the same VI



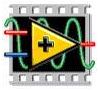
Queues III

- To Queue data between VIs, obtain a Queue with same name in each VI. The data-types must match!
- If the Queue size is -1, the Queue length is unlimited (as long as the memory allows)
- To Queue data within the same VI, no name is required, but is recommended



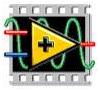
Queues IV

- Queues can be *previewed* to see the value of the element without actually *de-queuing* it – useful for monitoring purposes
- Data can be put on to the front-end (*enqueue at opposite end*) – can make a *Stack* rather than a *FIFO*
- The *status* of the queue can be read, to see the number of elements in the queue etc



Notifiers I

- A Notifier has to be *obtained* before it can be used. A Notifier must have a *Data-type*, with an optional name
- A reference is used to access the Notifier – eg to place data into *(send*), read data from *(status*), wait for data to be put into Notifier (*wait on notification*)
- A Notifier is *released* when finished with



Notifiers II

Example of using Notifiers between Loops