Exercise 2-3: Producer/Consumer (Events)

Goal

- Explore an example using the Producer/Consumer (Events) design pattern.

Hardware Setup

(Hardware) In the exercises where we work with Analog Input/Output channels, we use PCI-6221/USB-6212 multifunction I/O device paired with the BNC-2120 shielded connector block. Analog Input 2 should be connected to the Sine/Triangle BNC connector. Analog Input 3 should be connected to the TTL Square Wave BNC connector. The Sine/Triangle waveform switch should be set to Sine.
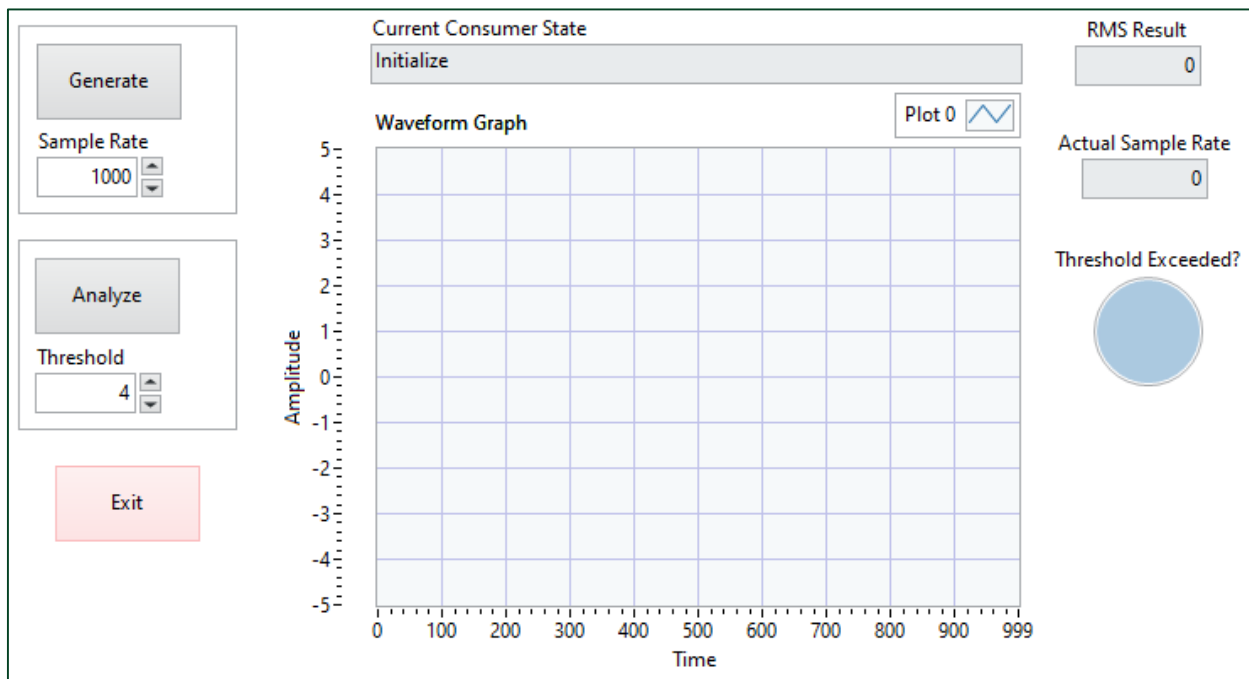
Scenario

You have a VI that uses the Producer/Consumer (Events) design pattern to process messages.

Guided Instructions

Part-1 Disable Controls

1. Open `C:\Exercises\LabVIEW Core 2\Producer Consumer (Events)\Producer Consumer (Events).lvproj`.
2. From the **Project Explorer** window, open the Main VI.

3. Run the VI.
4. Set the **Sample Rate** control value.

5. Click the **Generate** button.
   - Notice the **Current Consumer State** indicator displays Generate Signal message.
   - Notice that the **Waveform Graph** indicator displays the generated signal and the **Actual Sample Rate** indicator displays the sample rate of the generated signal.
   - Notice that the **RMS Result** and the **Threshold Exceeded** indicators haven't updated yet.
6. Change the **Sample Rate** control value and click the **Generate** button again.
   - Notice that the **Actual Sample Rate** indicator updated to show the sample rate that you set.
7. Set the **Threshold** control value.
8. Click the **Analyze** button.
   - Notice that **Current Consumer State** indicator displays the Analyze Signal message.
   - Notice that the **RMS Result** indicator shows the RMS value of the generated signal.
   - Notice that depending on the **Threshold** control value, the **Threshold Exceeded** indicator will also update to show if the RMS value is over the threshold value or not.
9. Click the **Exit** button to stop the execution.
10. Inject an error in the Producer Loop, and see what happens.
    - Navigate to the "**Generate Signal": Value Change** event case.
    - Add an **Error Cluster From Error Code VI** on the block diagram, as shown in the following figure.
    - Right-click the **error code (0)** input and select **Create» Constant**.
    - Set the constant to −1.
    - Wire the **error out** output to the output tunnel of the Event structure in the "**Generate Signal": Value Change** event case. The **Error Cluster From Error Code VI** will now pass an error with a code of -1 on its **error out** output.
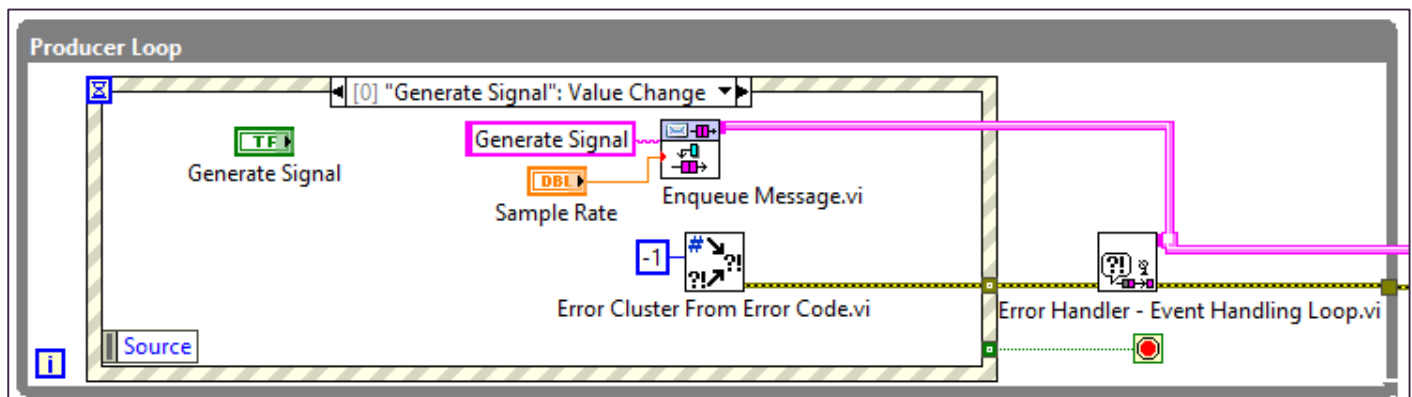
- Run the VI again and click the **Generate** button.
- Notice that the VI does not behave properly anymore. This is because an error in the Producer Loop has caused the Consumer Loop to shut down.
- Because the Consumer Loop was doing the bulk of the work, the VI no longer behaves correctly. The Producer Loop is still running.

11. Close the VI and the project. Do not save your changes.
This VI does not include any error handling.

End of Exercise 2-3