Exercise 6-2: Create and Debug a Stand-Alone Application

Goal

Create a build specification, build a stand-alone application (EXE) in LabVIEW, and debug the application running on the local computer.

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

Create a stand-alone application to run the Event-Driven State Machine - Main VI. After you prepare your files, you create an Application (.exe) Build Specification, and run the application. You then use LabVIEW to debug the running application.

Design

Use the Application (EXE) Build Specifications to create a stand-alone application for the event-driven state machine application.

Connect with the running application by creating a debugging session in LabVIEW.

Review the Building Applications Checklist

1. Select **Help» LabVIEW Help** to open the *LabVIEW Help*.
2. Select **Fundamentals» Building and Distributing Applications» Developing and Distributing an Application**.
3. Review the **Configuring Specifications for a Built Application** list of steps.

Guided Instructions

Creating an Application (EXE) Build Specification

1. Open C:\Exercises\LabVIEW Core 2\Creating an Executable\Event-Driven State Machine.lvproj.
2. Right-click **Build Specifications** in the **Project Explorer** window and select **New» Application (EXE)** from the shortcut menu.
3. (Optional) Place a checkmark in the **Do not prompt again for this operation** checkbox and click the **OK** button if you receive a prompt about SSE2 optimization.
4. Modify the filename of the target and destination directory for the application in the **Information** page.

   - Select the **Information** page.
   - Change the **Target filename** to Event-Driven State Machine.exe.
   - Enter C:\Exercises\LabVIEW Core 2\Creating an Executable\Executable in the **Destination directory** text box.

Tip You do not need to create the directory. LabVIEW creates any directories that you specify.

5. Specify the top-level VI for the application.
   - Select the **Source Files** page.
   - Select the **Event-Driven State Machine - Main VI** in the **Project Files** tree.
   - Click the **right arrow next** to the **Startup VIs** list box to add the selected VI to the **Startup VIs** list box.
6. Include code to allow debugging of the application.

   - Select the **Advanced** page.
   - Place a checkmark in the **Enable debugging** checkbox.
   - Click **OK**.
7. In the Project Explorer window, select **File» Save All**.
8. In the Project Explorer window, right-click the **My Application** build specification and select **Build** from the shortcut menu.
9. Click **Done** in the **Build status** window.

Running the Application

1. Close the Project Explorer window and exit LabVIEW.
2. Navigate to C:\Exercises\LabVIEW Core 2\Creating an Executable\Executable in Windows Explorer.
3. Run Event-Driven State Machine.exe.

   - Click the **Acquire** button.
   - Click the **Exit** button when done.
4. Verify that the application closed when you stopped the application and the application created a text file in the Executable folder.

Debugging the Application on the Same Computer

1. Launch LabVIEW.
2. Run the Event-Driven State Machine.exe.
3. Select **Operate» Debug Application or Shared Library** from the LabVIEW window.
4. Enter localhost in the **Machine name or IP address** text box.
5. Select **Event-Driven State Machine.exe** from the **Application or shared library** drop-down menu.

   - Click the **Refresh** button if Event-Driven State Machine.exe does not appear in the list.
6. Click the **Connect** button to create the debugging connection.
7. Start debugging the running application.

   - Open the block diagram.
   - Turn on **execution highlighting**.
   - Try using probes, breakpoints, and single-stepping.
8. Click the **Stop** button in the debugging window to stop the application.

End of Exercise 6-2