Exercise 5-3: Create an Execution Log File

Goal

Explore how an example application logs timestamps, state information, and error information to an execution log file.

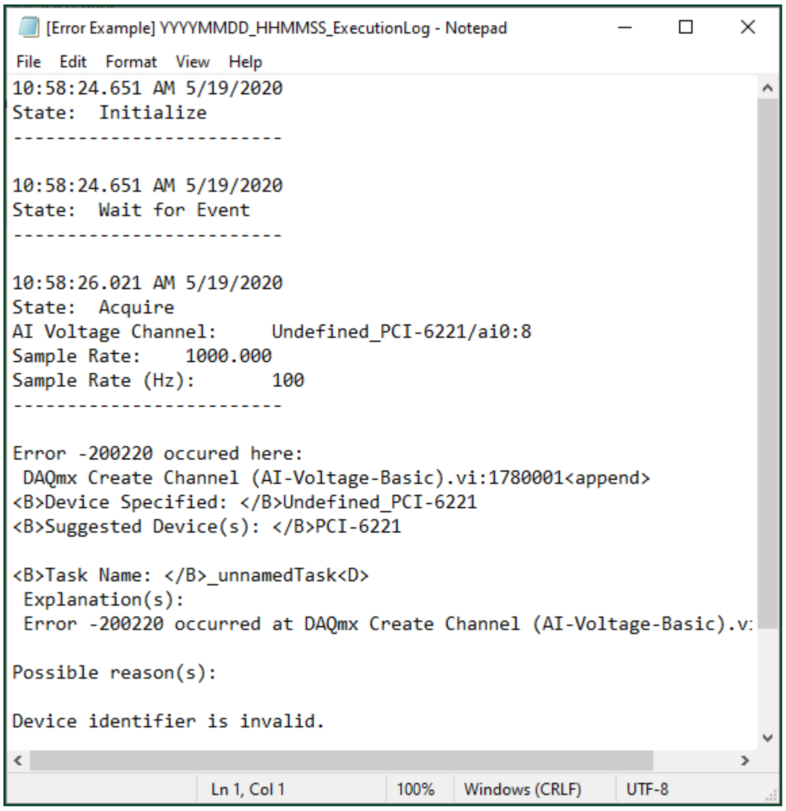
Hardware Setup:

(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.

Instructions:

Explore Example Execution Log Files

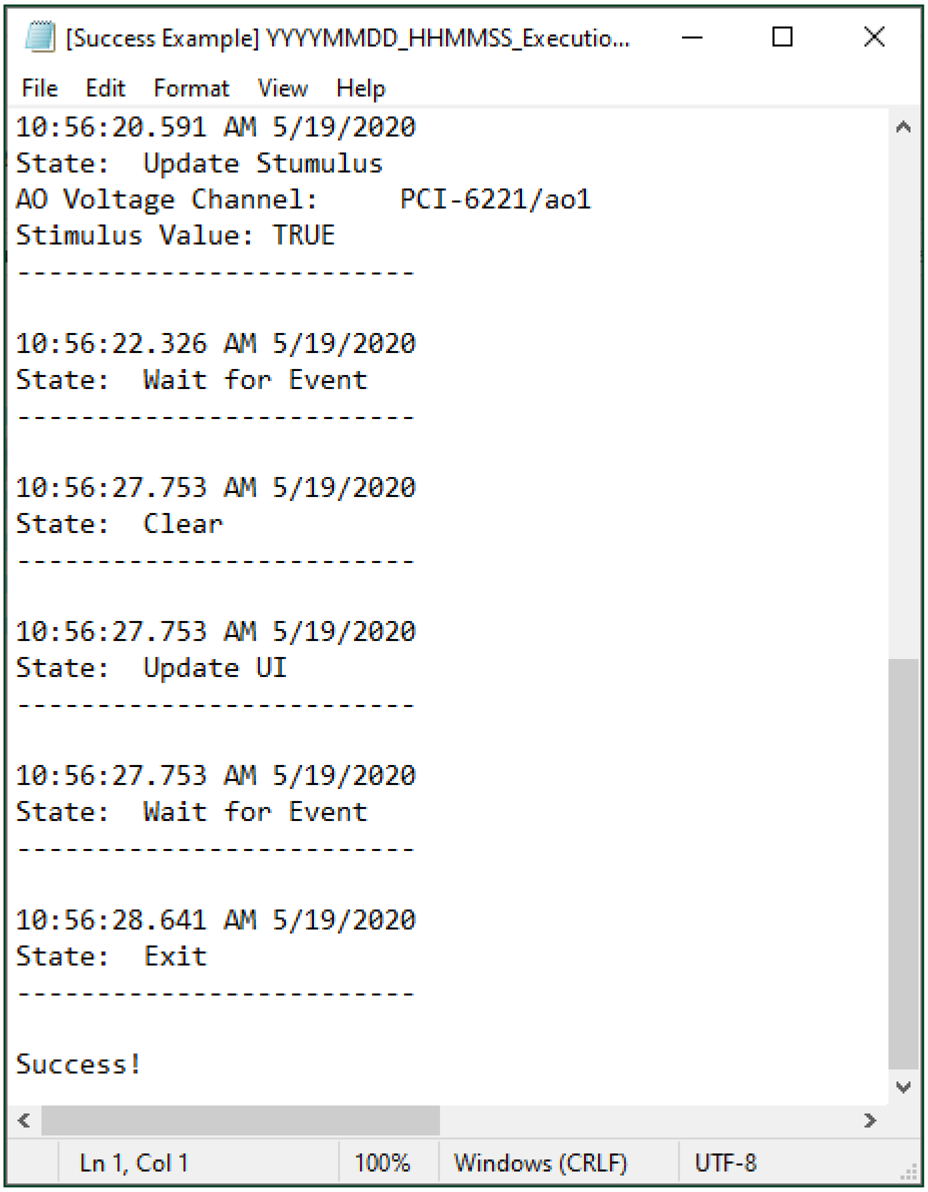
1. In Windows Explorer, go to the `C:\Exercises\LabVIEW Core 2\Execution Log` directory.
2. Explore an example of an execution log where an error occurs.
   - Double-click **[Error Example] YYYYMMDD_HHMMSS_ExecutionLog.txt** to view its contents.
   - By looking at this execution log, you can see the error information at the end of the file. You can also see the timing, states, and state data that led to this error occurring in the application.
   - Close the file when finished


[Error Example] YYYYMMDD_HHMMSS_ExecutionLog - Notepad
File Edit Format View Help
Ln 1, Col 1
100%
Windows (CRLF)
UTF-8

```
10:58:24.651 AM 5/19/2020
State:  Initialize
-------------------------

10:58:24.651 AM 5/19/2020
State:  Wait for Event
-------------------------

10:58:26.021 AM 5/19/2020
State:  Acquire
AI Voltage Channel:     Undefined_PCI-6221/ai0:8
Sample Rate:    1000.000
Sample Rate (Hz):       100
-------------------------

Error -200220 occured here:
 DAQmx Create Channel (AI-Voltage-Basic).vi:1780001<append>
<B>Device Specified: </B>Undefined_PCI-6221
<B>Suggested Device(s): </B>PCI-6221

<B>Task Name: </B>_unnamedTask<D>
 Explanation(s):
 Error -200220 occurred at DAQmx Create Channel (AI-Voltage-Basic).v:

Possible reason(s):

Device identifier is invalid.
```

3. Explore an example of an execution log where no errors occur.
   - Double-click **[Success Example] YYYYMMDD_HHMMSS_ExecutionLog.txt** to view its contents.

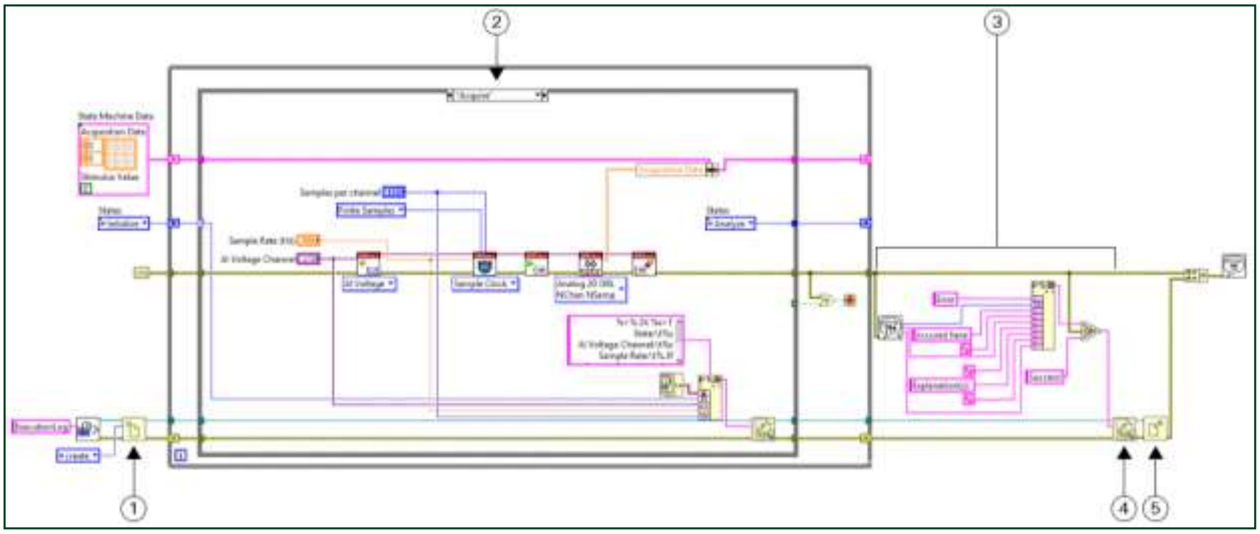
- In this example execution log, you know that no errors occurred because the end of the log file explicitly logs `Success` when the application exited.
- The information in the execution log allows you to have an idea of what happened during this run of the application.
- Close the file when finished.



```
10:56:20.591 AM 5/19/2020
State:  Update Stumulus
AO Voltage Channel:     PCI-6221/ao1
Stimulus Value: TRUE
-------------------------

10:56:22.326 AM 5/19/2020
State:  Wait for Event
-------------------------

10:56:27.753 AM 5/19/2020
State:  Clear
-------------------------

10:56:27.753 AM 5/19/2020
State:  Update UI
-------------------------

10:56:27.753 AM 5/19/2020
State:  Wait for Event
-------------------------

10:56:28.641 AM 5/19/2020
State:  Exit
-------------------------

Success!
```

Explore Example Code That Creates Execution Log Files

1. Open `C:\Exercises\LabVIEW Core 2\Execution Log\Execution Log.lvproj.`
2. From the **Project Explorer** window, open the Event-Driven State Machine (Execution Log) VI.Move all the control references into the “Acquire”: Value Change event case.
3. Explore the block diagram.



1. **Open/Replace/Create File** – Creates the execution log file.
2. Scroll through every state in the state machine. Notice that every state contains code that writes timestamp, the current state, and state data to the execution log file.
3. If an error has occurred, this part of the VI outputs a string containing the error information. If no errors have occurred this part of the VI outputs a string containing `Success!`.
4. **Write to Text File** – Writes this text to the execution log file.
5. **Close File** – Closes the file.

4. Generate an error log with no errors.
   - Run the VI.
   - Click a **few buttons** without causing an error.
   - Click the **Exit** button to stop the VI.
   - Open the new execution log created in the `C:\Exercises\LabVIEW Core 2\ Execution Log` directory. Notice that the last line of the execution log is Success!.
   - Close the execution log file.
5. Generate an error log with an error.
   - Run the VI.
   - Double-click the **text** in the AI Voltage Channel control and set it to `Undefined_cDAQ1Mod8/ai0:8`.
   - Click the **Acquire** button. This will cause an error in the Acquire case of the state machine because Undefined_cDAQ1Mod8/ai0:8 does not exist in your system. Click **Continue** on the error dialog window. The VI now exits.

- Open the new execution log created in the `C:\Exercises\LabVIEW Core 2\Execution Log` directory. Notice that the last lines of the execution log contain the error information. Notice that the Acquire state information before the error contains the incorrect AI Voltage Channel value of `Undefined_cDAQ1Mod8/ai0:8`.
- Close the execution log file.

6. Close the VI and project when finished.

On the Job

1. Would any of your applications benefit from creating an execution log? If so, what information should you write to the execution log?

   ____
   ____
   ____
   ____
   ____

End of Exercise 5-3