

Bloomy VI Corner – Simple State Machine Template

by Keith Brainard, Project Engineer, Certified LabVIEW Developer

The simple state machine template virtual instrument (VI) provides a great jump start on new projects. For top-level VIs and almost any looping VI, it is good practice to use a state machine as it allows you to have a state for each action you wish to perform. Using a flat or linear construct can be extremely time consuming when it is necessary to reorder actions or modify the data or process flow. Developed by Bloomy Controls project engineer Keith Brainard, this VI (**Figure 1**) includes all of the most basic components that any top-level VI needs.

The state machine template consists of a while loop, a case structure, a dozen pre-wired shift registers, a queue for setting up states, and an enumerated data type with some predefined states. These predefined states serve as reminders for states that should exist in any application, and include No Action, Blank, Initialize, Error, and Shut Down. No Action is a state that is used to handle user interface events with the event structure. The event structure enqueues states to execute in response to detected events. The pre-wired shift registers and the Blank case combine nicely to provide a great deal of time savings throughout application development. In order to add a state to the state machine, simply duplicate the Blank state. This saves time in wiring shift registers in new states.

This VI easily is accessed by a feature on the functions palette called "Merge VI." When a VI is added to the functions palette, the user can right click on that VI and select the Merge VI option. This causes the VI to be included on the palette as a code snippet, not as a sub VI. The result is that when the VI is selected from the palette, it is included directly as code in the target VI. There is no need to save the template VI as a VI with a new name.

The simple state machine template is something that each user can customize, or it can be a standard throughout a company when a uniform look and feel is desired for all code produced. It saves considerable time in starting a new program or even a new user interface screen. Furthermore, it serves as a gentle reminder to include some important common elements, such as error handling and using a queue and event structure to control program flow. ↪

If you would like help using state machines for your applications, contact Charles Wimberley at (860) 298-9925 or info@bloomy.com.

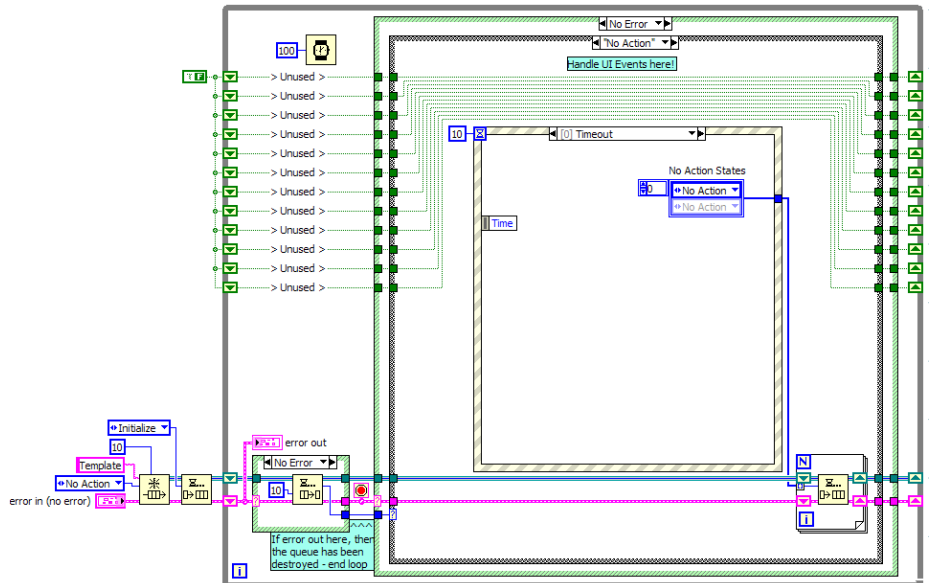


Figure 1. State machine template

Newsletter Staff

Editor, Phia Pascua

phia.pascua@bloomy.com

© 2003 Bloomy Controls Inc. All rights reserved. Product and company names listed are trademarks or trade names of their respective companies.

