Sequential Control Example in LabVIEW

What is Sequential Control? The aim of this example is to develop a better understanding of sequential control and using state diagrams and State Machines in LabVIEW.

In this example you will learn:

- The basic concepts of Sequential Control
- The basic concept of a state machine and how to create a state machine in LabVIEW. In LabVIEW you use a State Machine to create a Sequential Control application

There are different techniques in use when it comes to design and implementation of Sequential Control, such as:

- Sequential Function Charts (SFC)
- State Diagrams

Both these techniques are very similar. They have states (in SFC the term steps is used) with actions to be executed when the state is active. They have transitions from one active state to another taking place if the transition condition is logically true.

Task: In this Example we will implement a simple Sequential Control (Start/Stop an Engine) in LabVIEW using the State Machine principle in LabVIEW. The State Machine approach in LabVIEW uses a Case structure inside a While loop to handle the different states in the program, and the transitions between them. A Shift Register is used to save data from and between the different states. Below we see the State Diagram for the Example:

A State Diagram consists of:
- Steps
- Transitions
- Actions
The User Interface (Front Panel) could look something like this:

When the user push the “Start Engine” button, the light gets on and when the user push the “Stop Engine” button, the light gets off. When the user pushes the “Exit” button, the program will stop.

The Procedure is as follows:

**Step 1:** Create a **New VI** (File→New VI) (Blank VI)

**Step 2:** Create your **Front Panel/User Interface** as shown above.

**Step 3:** Add a **While Loop** to your VI

Your application should now look like:

**Step 4:** Add a **Case Structure** to your VI inside the While Loop

Your application should now look like:
**Step 5:** Add a **Shift Register** to your While Loop by right-click on the While Loop and select “Add Shift Register”.

Your application should now look like:
Step 6: Create the Init State according to the State Diagram. Your application should now look like:

![State Diagram Image]

Step 7: Add a Case called, e.g., “Wait” in your Case Structure. Create an Event Structure inside this case.

![Event Structure Image]

Add Events (right-click on the Event Structure) for your 3 different buttons; “Start Engine”, “Stop Engine” and “Exit”. Your application should now look like:

![Event Structure with Events Image]
Step 8: Add the Start Engine, Stop Engine and Exit cases (states) and create the Actions according to the State Diagram.

Your application should now look like:

Start Engine:

Stop Engine:
Step 8: Run the Application and check if it works as expected.

Turn on "Highlight Execution" and see how the flow in your application is.

Further reading:

If you want to learn more about LabVIEW, Sequential Control and State Diagrams, you should take a look at:

- Tutorial: Introduction to LabVIEW by Hans-Petter Halvorsen
- Tutorial: Introduction to State-based Applications in LabVIEW by Hans-Petter Halvorsen

Hans-Petter Halvorsen, M.Sc.

E-mail: hans.p.halvorsen@hit.no
Blog: http://home.hit.no/~hansha
Phone: 35 57 51 58
Office: B-237a