

# PH2150

## Introduction to LabVIEW (2)

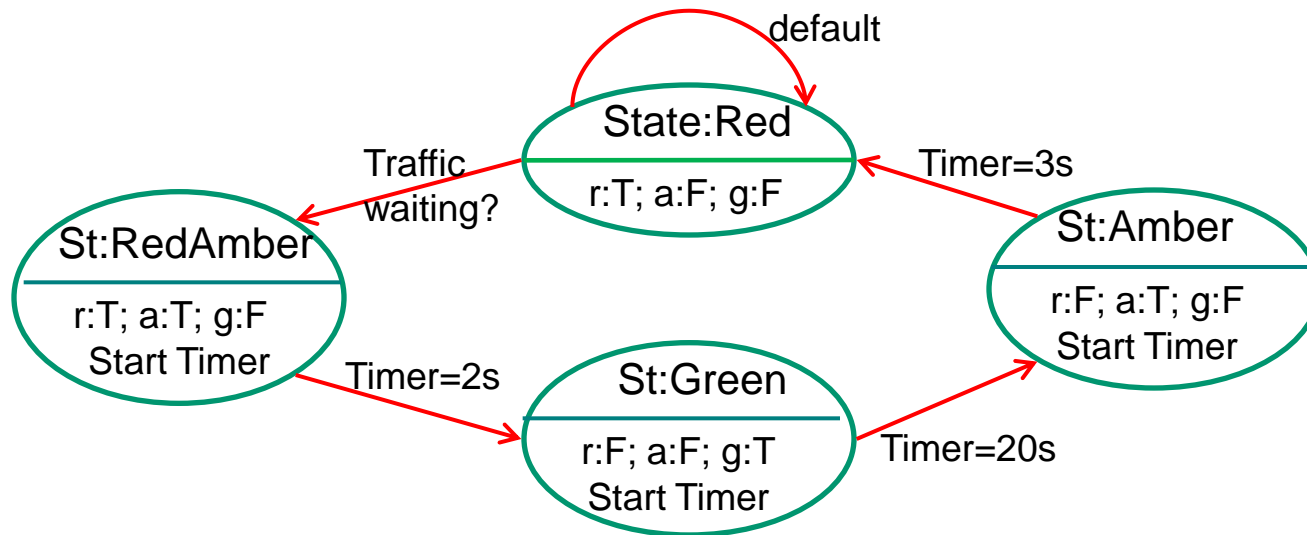


Gary Boorman

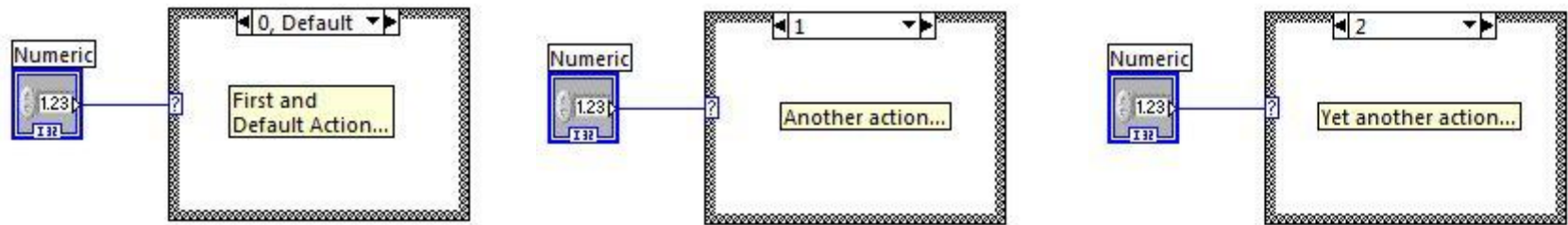


# LabVIEW – State Machines: The Essence of Control

A State Machine (SM) is a model of behaviour containing a set of states, transitions between states and actions



# LabVIEW – CASE Structure

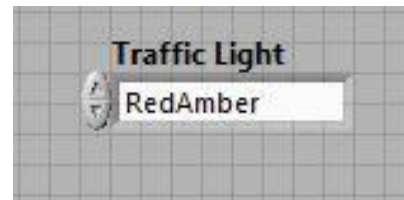


The CASE structure can have different data-types wired to the selector, not just Booleans. A 'default' case is required unless all selector values have a case.

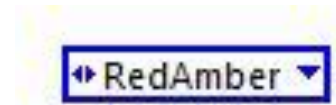


# LabVIEW – Enumerated Type

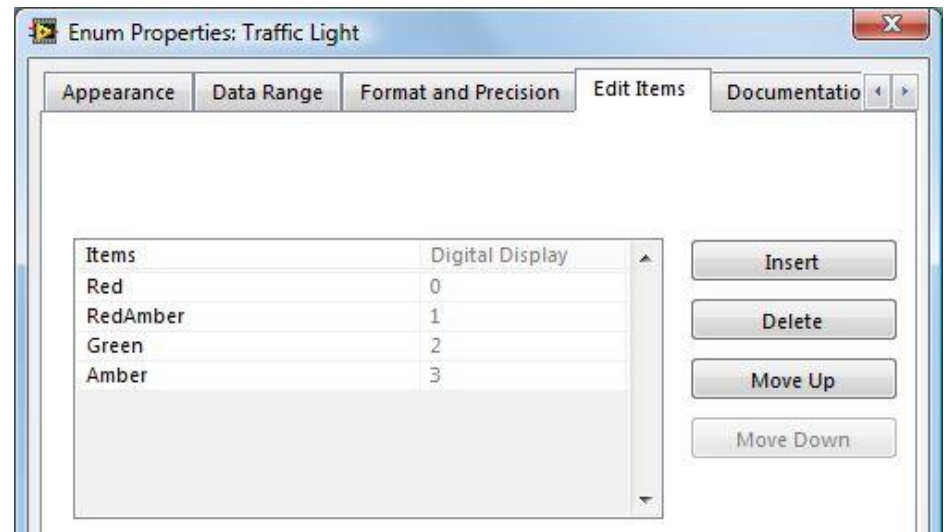
The Enumerated type (enum) associates an integer with readable text. Items (text) can be added to the enum, and an integer can be automatically associated with the text.



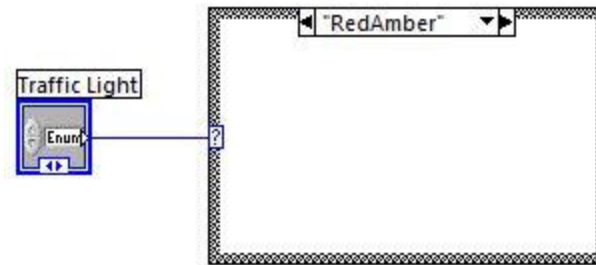
enum: Control



enum: Constant



# LabVIEW – CASE and ENUM

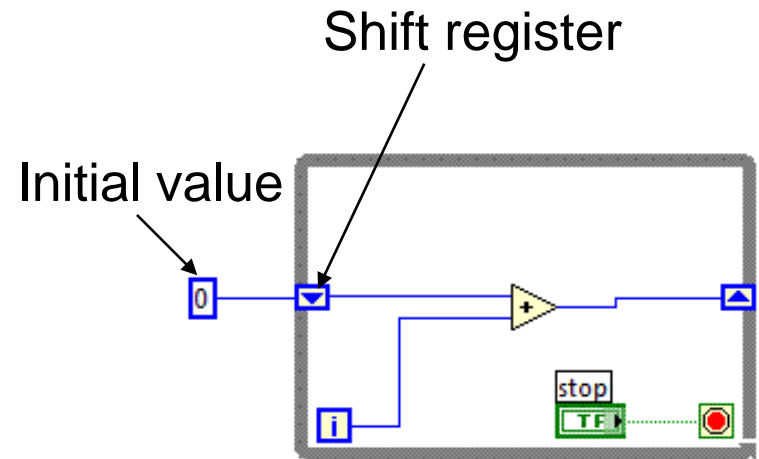


The CASE structure can have an ENUM wired to the selector. Wire ENUM to selector then right-click on label and ‘add case for every value’. The ‘default’ text should then be deleted.

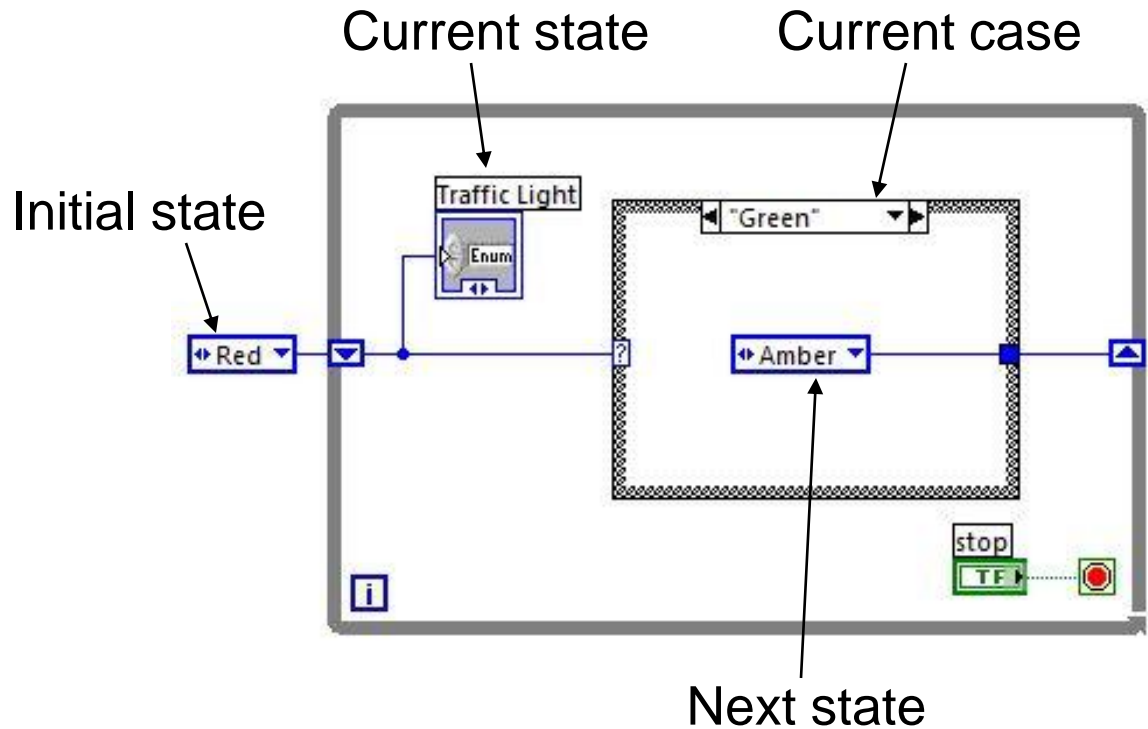


# LabVIEW – Shift Registers

- Shift registers store and pass data between successive iterations of For and While loops
- Can be initialised to a value
- If not initialised, value from previous call of loop is used (can be unexpected values!)

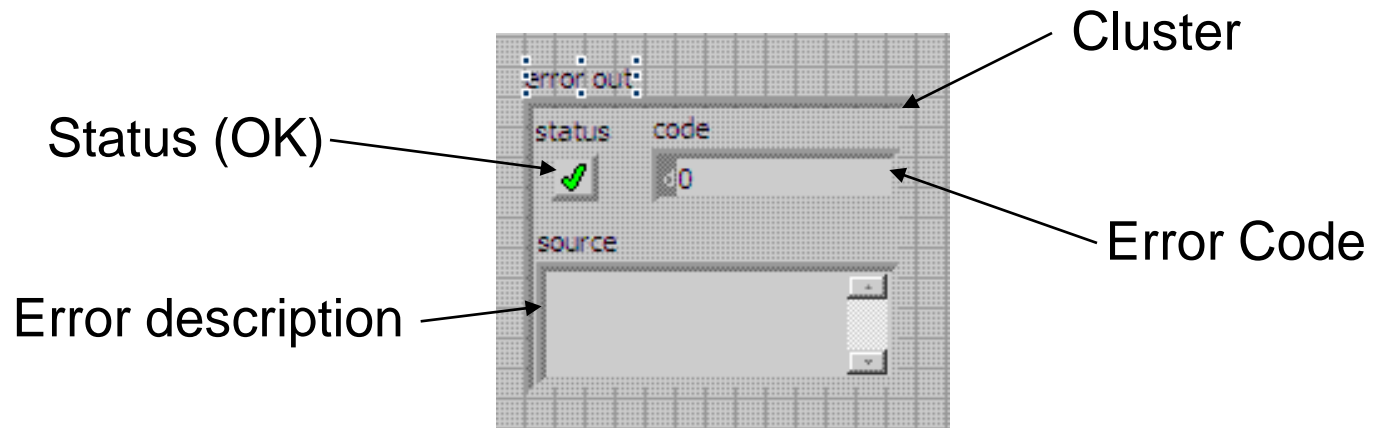


# LabVIEW – A State Machine



# LabVIEW – Clusters (1)

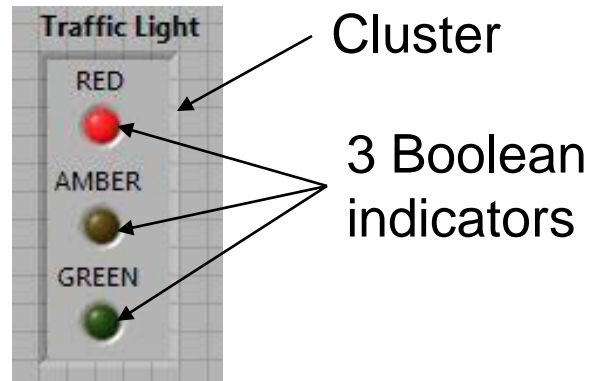
- Clusters are similar to arrays but can consist of mixed data-types
- The error cluster is probably the most common cluster – consists of a Boolean, Integer and String



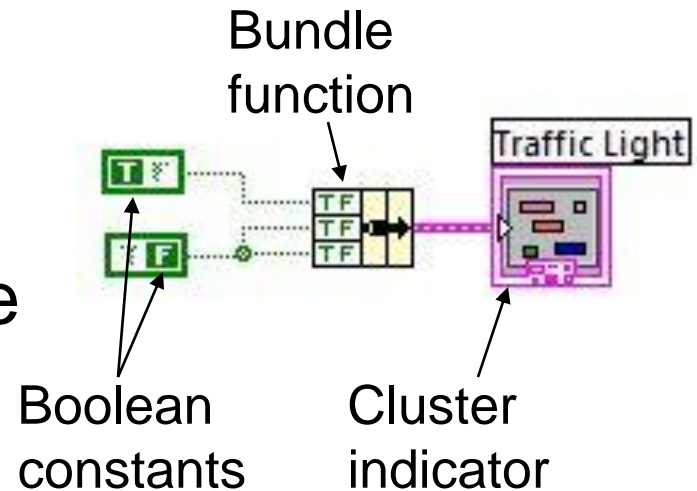


# LabVIEW – Clusters (2)

A cluster can hold elements that are the same.

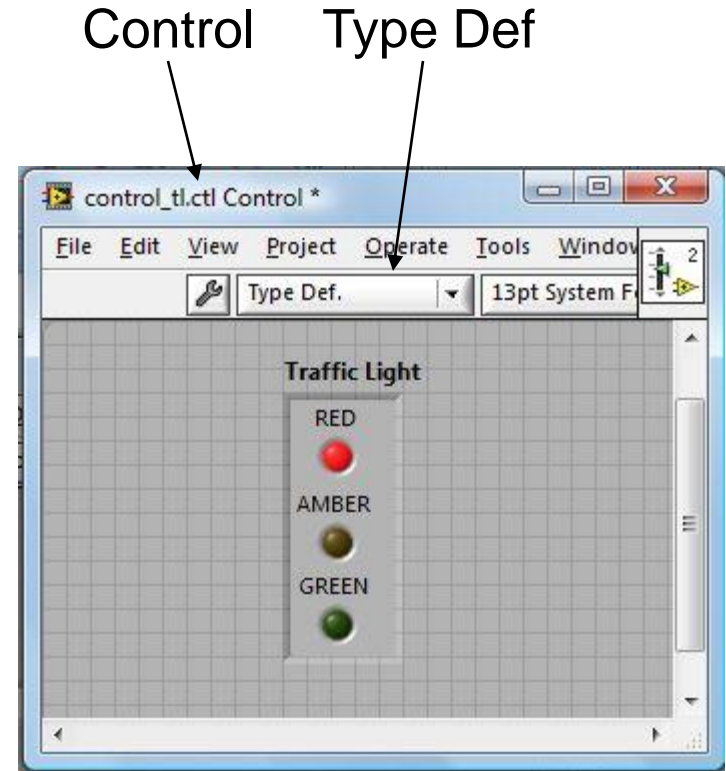


Individual elements are 'bundled' together.  
Cluster controls can have elements 'unbundled'.



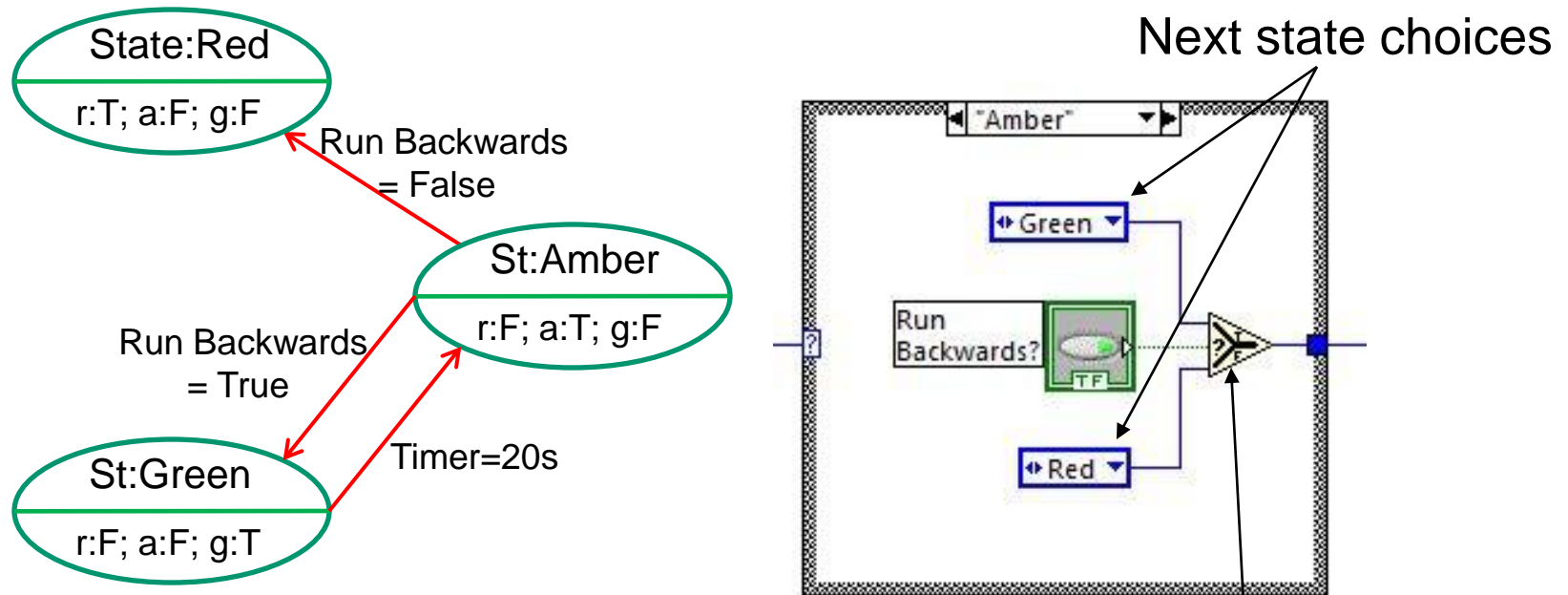
# LabVIEW – Custom Controls

Customs Controls are useful when needing many copies of a Control/Indicator/Constant. Do File>>New...>>Custom Control to open a front panel for the new control. Save as Type Def to update all versions of the control used.





# LabVIEW – State Machine Decision Making



If more than two options for the next state, then use a CASE structure within the current state.

