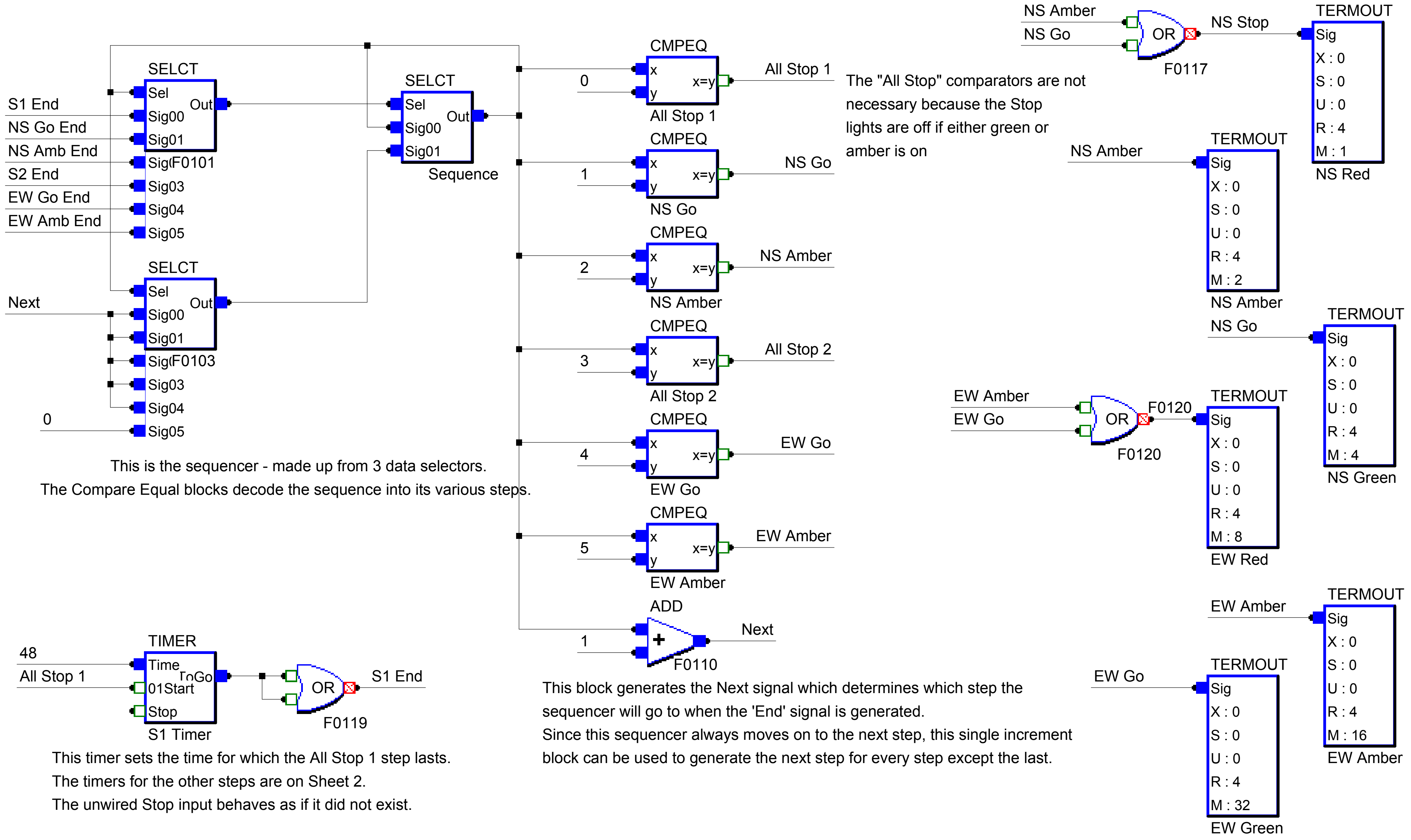
TrafficLight1 Example

The TrafficLight1 example illustrates a sequencer (state machine) which simply progresses from one state to the next, then returns to the start in an eternal cycle.
Each step is controlled by its own timer.

This is a simple crossroads light which operates according to the Australian / New Zealand light sequence and has just 6 states:
All Stop 1 = 0; North-South Go = 1; North-South Amber = 2; All Stop 2 = 3; East-West Go = 4; East-West Amber = 5

Note the simple way in which the "Next" signal is generated. Also note that F0103, the data selector which determine which step is to be executed next, has a constant 0 wired into input Sig05. This ensures that the sequence cycles back to the first state. If this input were omitted, the effect would be the same, because selecting a non-existent input of a data selector generates an output value of 0.

Generation of the red light signals is done in an economical and fail-safe way. If neither amber or green is active, then red must be switched on. This will not work with the British light sequence where there is a "Prepare to Go" state showing red and amber simultaneously.



The "All Stop" comparators are not necessary because the Stop lights are off if either green or amber is on

This is the sequencer - made up from 3 data selectors.

The Compare Equal blocks decode the sequence into its various steps.

This block generates the Next signal which determines which step the sequencer will go to when the 'End' signal is generated.

Since this sequencer always moves on to the next step, this single increment block can be used to generate the next step for every step except the last.

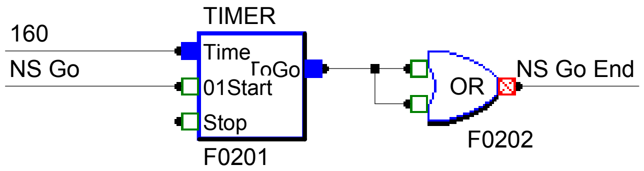
This timer sets the time for which the All Stop 1 step lasts.

The timers for the other steps are on Sheet 2.

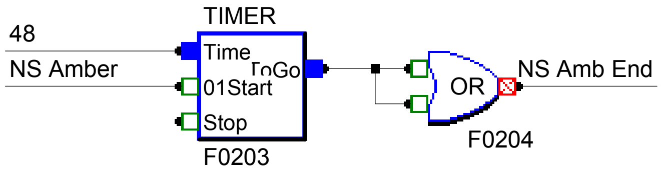
The unwired Stop input behaves as if it did not exist.

AmbiLogique Ltd

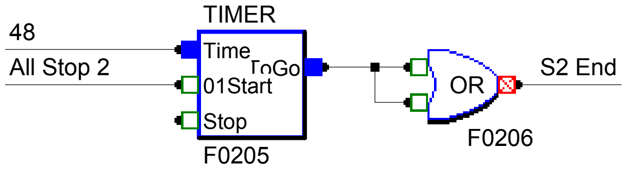
TRAFFIC LIGHTS 1 - Sheet 1



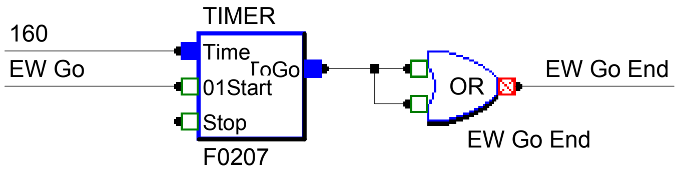
Using this technique we can set up a separate timer for each step in the sequence.
This step is timed at 160 / 16 = 10 seconds.

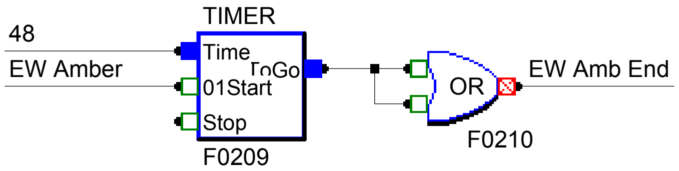


The top input, "Time" is wired from a constant. We generate the constant by right-clicking the wire, selecting "Properties" and typing a pure number into the "Signal Name" box in the dialog.



The "01Start" input is wired from an incoming cross-reference. We generate this cross-reference by drawing a wire which starts in blank space and ends at the input pin.
We then right-click it, select "Properties" and type the signal name into the "Signal Name" box in the dialog.
Note that the signal name will not show until we do a Wire Check (from the "Project" menu item). The names of all cross-references are then shown.





Outgoing cross references can be generated in one of two ways.
In each case, the cross-ref is drawn as a wire starting at an output terminal
In the case at the top, "EW Go End," the function block was named with the signal name and the wire was left unnamed. When the Wire Check was carried out, AmbiLogic named the wire automatically from the function block.
In the second case, "EW Amb End," the wire attached to the function block output was named directly using the right-click-Properties technique.
The Wire Check found the wire segment already named, and refrained from renaming it.

AmbiLogique Ltd

TRAFFIC LIGHTS 1 - Sheet 2