

Report: Functional modelling in Simulink and Stateflow

Task Summary

This lab assignment was challenging but exciting. I enjoyed working with Simulink and Stateflow and seeing the system come to life when simulated. I was able to model the traffic lights controller system according to the high-level requirements defined in lab one and develop assertion flows for the requirements given.

There were some issues I encountered which I will summarise below:

1. For the system initialisation requirement, I could not simulate that the system starts only after a delay if the controller's off period is within a certain threshold.
2. When the `callButtonReady` variable is not true, the system does not record button presses. This means that all call requests are ignored, and the button will have to be clicked again when `callButtonReady` is true.
3. It was pretty challenging to model the interaction between the two carriageways. I settled for sending the outputs in states which isn't exactly ideal.
4. Point 2 above affects the `checkPCTimeFrame` assertion. Suppose the button is pressed before `callButtonReady` is valid; it throws an error. A fix I tried out was adding a condition that transitions to the next state when the call button is pressed, and the `callButtonReady` signal is equal to 1. However, I found out after coverage analysis that it never resolves to true and so doesn't go into its intermediate state, "handleAssertionDuring".

Coverage Analysis

For the coverage analysis, I ran the simulation a couple of times and made some edits to the model to achieve more coverage. However, there were some issues encountered which I will summarise below:

1. The assertion blocks never reach their `handleAssertionExit` state, which is intended behaviour because we do not want errors to be thrown. However, I think there may have been better ways to handle the assertion logic. I am just not sure how.
2. For substate exit on state "on", I found out that I would have to manually turn off the power button on each substate to achieve a 100%. I tested it out with two other states other than "vehiclesPassing". They both show coverage.
3. Coverage analysis flagged the issue mentioned in point 1 in the task summary above. The condition specified never resolves to true.
4. The carriageway folder in "slcov_output" seems like it shows coverage analysis for the carriageway model on a "standalone" basis. Due to this, the model does not get the necessary data to operate parts of the model. It relies on the power and call button press of the controller.