FSM standards and code generation

Lab 7, ESDM

Objective

Using the Model Advisor and the Code Generation tools for model-based development in Simulink.

Theoretical aspects

TBD. See the Lectures.

Exercises

1. Design a FSM in Stateflow with two inputs `MotorOn` and `LatchReached` and one output `LiftgateClosed`, for the following requirements:
    1. The liftgate shall be considered open (`LiftgateClosed = FALSE`) always when `MotorOn = TRUE`.
    2. The liftgate shall be considered closed (`LiftgateClosed = TRUE`) when `MotorOn = FALSE`, if the input `LatchReached` becomes `TRUE` within `CP_MaxLatchDelay` after `MotorOn` has become FALSE.
    3. If the input `LatchReached` becomes `TRUE`, but the motor was not started anytime within `CP_MaxLatchDelay` prior to this moment, it shall be ignored and the liftgate shall be considered open.
2. Redesign the finite state machine using a separate state for the timer operation, in a parallel state (AND decomposition / parallel decomposition).
3. Test your design: put appropriate inputs and observe the output signals.
4. Run the Model advisor tool (Analysis -> Model Advisor -> Model Advisor), select and run the "Modeling Standards for MAAB" checks. Observe the warnings/failures and fix some of them.

5. Generate C code from the model (Code -> C/C++ Code -> Build Model). Locate the code files, open them and identify the implementation of the state machine. How is it implemented (with which C instructions)?