Model standards and code generation tools

Lab 5, ESDM

Objective

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

Theoretical aspects

Model Advisor and MAAB rules

`MAAB` (or `MAB`) and `JMAAB` are sets of guidelines and recommended practices for the implementation of Stateflow models (e.g. like `MISRA-C` for C code).

Respecting these guidelines implies a high quality of the models and is often required in practice.

Some of the rules can be checked automatically with the `Model Advisor` tool.

Generating C code

Matlab can generate C code automatically from Simulink models, including Stateflow charts.

Steps:

- specify the data types of all inputs and outputs
- Right-click on chart -> Block Parameters -> Code Generation -> Function Packaging and select:

  - Reusable function
  - User specified function name (you choose name)
  - User specified file name (you choose name)

- Right-click on chart -> C/C++ Code -> Build This Subsystem

It is recommended that the delays are specified in `ticks` rather than `sec` or `msec`. In this way the generated code is simpler.

Exercises

1. Design a FSM in Stateflow for a simplified alarm system. The system has two inputs `DoorOpen` and `CodeOK` and one output `AlarmOn`, for the following requirements:

    1. When input `DoorOpen = FALSE`, the alarm is turned off (output `AlarmOn = FALSE`).
    2. The alarm is turned on after 15 seconds from the moment the door is open (`DoorOpen = TRUE`), while `CodeOK = FALSE`.
    3. The alarm is turned off when the correct passcode is entered (`CodeOK = TRUE`).

2. Test your design using another model. Inside this test model, use the `Model Reference` to reference the model under test.

3. Run the Model advisor tool (Analysis -> Model Advisor -> Model Advisor), select and run the "Modeling Standards for MAB" checks. Observe the warnings/failures and fix some of them.

    - You need to install the toolbox `Simulink Check` in Add-Ons -> Get Add-ons

4. Generate C code from the model. Locate the code files, open them and identify the implementation of the state machine.

    How is it implemented (with which C instructions)?

    How would you implement it yourself in C?