2019-03-15 Semantics Meeting (lgr)

Meeting Details

- Moderator: als
- Minutes: lgr
- Attendees:
    - nir
    - aas
    - ssm
    - sdo
    - peu
    - als
    - rvh
- Start: 9:45
- End: 10:34

Agenda

New Annotations for Model Checking (aas)

- NuSmv allows to define preconditions for the checking to cut paths from the analysis
- Specifing invariants yields condition to be true
- '@Assume' and '@AssumeRange' can be used now, defines ranges for a number

Unit Tests for Model Checking (aas)

- implemented JUnit Tests, one model is checked with all algorthims (and parameters)
- result is written into a csv file
- time for the call is logged
- failing models take more time
- bmc takes a lot of memory and time
- ask in Bamberg why some comnfigurations are inefficient
- bigger models? -> traffic light examples yields problems, so bigger examples with longer runtimes are needed
- a motor and a traffic light example should be included so good comparisons can be made
- high parallelism (example with parallel writing of integers) causes spin to be overloaded

Current State of the RaceYard Project (peu)

- goal: ECU for motor control should be simulated without the need of a board
- Cannoo allows to simulate virtual busses
    - a database with board and input/output, bytes etc in board can be extracted -> used for analysis of values from car
    - cannoo can work with the database too
- Functional Mockup interface used for coupling of ECU unit to other components on CAN bus (tell cannooo how to use the given dll)
- interface is defined in xml file
- cannoo can read inputs and write outputs within running simulation
- only scalars can be defined in interface, no arrays
- variable naming convention: structured: allows cannoo to understand special characters [,], .
- in hardware receive and transmit boxes are used -> also try to use it in interface
- there would be too many messages, so the transmission data is reduced to not sending the message but the values of the message
- allows combining more messages into one
- ECU acually needs a tick every milli second, every tick may produce up to 10 messages back to cannoo (scaling by 1000 still does not yield better performance)
- messages are buffered on side on cannoo, if too many messages are produced, its a ring buffer and ignores old messages
- only the newest value of each message is important
- if frequency of message sending changes, the simulation must be adjusted too (-> possible errors)
- Simulation layer is used instead of Hardware Abstraction Layer
- There already is a virtual car in CarMaker -> allows import of simulink Controler -> can be linked to Cannoo via FMI
    - a phyically correct model of the car is needed (only standard car is available)

- in Cannoo test cases can be defined, so simple test cases can be defined to test the ECU, next step is CarMaker (atm no licence available)
- database with datalog has same format as virtal can bus, so can be used as input for the ECU -> ECU cannot be verified well