

# 2019-12-05 K-Meeting (sdo)

## Meeting Details

- Moderator: ssm
- Protocol: sdo
- Attendees:
  - als
  - nre
  - rvh
  - ssm
  - lgr
  - cds
  - peu
  - kolja
  - sdo
- Start: 16:00
- End: 17:30

## Agenda

- [Debugging code generated from SCCharts \(peu\)](#)
- [New Dataflow Synthesis \(circuit\) \(kolja\)](#)
- [Host code and OO \(als\)](#)
- [Text in SVGs in KEITH \(nre\)](#)

## Debugging code generated from SCCharts (peu)

- count delay is not supported by state-based lean Java code generation
- Comment on initial state may be not correctly visualized or implemented, this is incremental update
- On edge labels do not work for multiple labels (cds checks this)
- Comment nodes are now also on transitions that are part of transitions of the original model
- Original transition can be seen as a comment in the generated code
- Current state/transition in generated code is visualized in green, they are inferred by the Javadoc/annotations on state methods and comments on transitions
- The active SCCharts states (red) are inferred by the current state variable
- Different kind of breakpoints for transition checked and transition taken
- Breakpoints should be set in the debug diagram view, since it is difficult to find the correct line in the generated code
- The diagram is only updated whenever a breakpoint is reached (stepping is an optional goal)
- A stepping mechanism in the SCCharts level might be helpful (rvh) for stepping through transition checks and transitions, new states
  - add breakpoints on the next location the execution will continue (next state, next transition to check) and remove it afterwards to simulate stepping
- Optional: Step into host code calls
- Check more example with immediate transitions and more than one outgoing transition
- It is intended to use a different view for debugging than for general diagram
- Debug perspective for SCCharts might be helpful
- Focus and context debugger is currently not active. It can be added to the Highlighter (als)

## New Dataflow Synthesis (circuit) (kolja)

- Registers are now connected correctly
- Refactor Hide local variables to Show local variables
- Old circuit synthesis is refactored into different steps to make it maintainable
- Assignments can be annotated as sequential
- Constraint for sequential equations can be visualized in the diagram
- This can also be seen in the separate assignments
- The "seq" keyword allows to override IUR dependencies, this feature should be used for experimenting with sequentiality in dataflow
- Arrays can be visualized in dataflow
- Port order is not correctly (fix this)
- Arrows in dataflow (try this) with option
- Green line between the same instances of some referenced model
- First actual bug in dataflow was found and documented (TODO add picture)
- Dataflow highlighting shows what wires are true or false (true is highlighted in red)

## Host code and OO (als)

- new pragma "resource" to add new resources for compilation and in runtime for simulation
- Host classes can be added, this can also be used to have Lists in SCCharts
- A list can be used together with scheduling directives to behave deterministically
- Scheduling directives in host classes can be accessed from outside the class, via <variableName>.<nameOfDirective>
- Classes can be directly implement in SCCharts as SCL code

- In C this is translated into structs and in Java into classes where the methods are inlined (inlining is optional and not possible for for-loops)
- SCCharts are handled as objects/classes
- Using the reference an SCChart can be used same as the hostcode class
- Instead of a class definition an SCChart can be used
- All of this works together with regions and inheritance
- State-based does not work with the class feature
- Sausage folding option has be moved to the correct category

## Text in SVGs in KEITH (nre)

- Every renderer renders font differently
- SVG has option to squish the letters together (textLenght)
- SVG has another option to squish them better (lengthAdjust)
- Since the SVGs are rendered in Chrome they are always rendered correctly in Chrome
- Test this for Windows and other SVG-viewers