System-level co-modeling AADL and Simulink specifications using Polychrony (and Syndex)

AADL Standards Meeting
June 6., 2011

Jean-Pierre Talpin, INRIA

Parts of this presentation are joint work with Paul, Thierry, Loïc, Huafeng, Yue (INRIA Espresso), Julien, Sandeep (VT), Dumitru, Yves, Robert (INRIA Aoste), Marc, Frédéric (N7), Martin, Mamoun (IRIT) and partly funded by the ITEA2 project OPEES, Artemisia project CESAR and the Polycore associate project.
Heterogeneity of skills, teams, tools, methods
Motivation

- CATIA
- Nastran
- Simulink
- Scade
- Rhapsody
- ...

- CAN
- Flexray
- ARINC 653
- AADL
- Profiling
- Energy
- ...

co-modeling

- analyse
- verify
- test

map

simulate
Motivation

- Simulink
- AADL

analyse
verify
test

co-modeling

map
simulate
Case study of an airplane doors control system

Functional specification

A suitable GALS model of computation

Structural specification

Simulation, verification, performance evaluation, scheduling, distribution …

ACM SAC’11 – Artemisa project CESAR
Asynchrony

Asynchronous architecture models

AADL diagrams (or UML, SysML, MARTE)  Kahn process networks, CCS, CSP, …
Synchrony

Synchronous behavior models

Elements of Simulink (or Geneauto, Scade, Ptolemy)

Synchronous data-flow, SCCS, …
Polychrony

Signal (or RT-Builder, CCSL, MRICDF)

... and Syndex (for allocation and real-time scheduling)
Polychrony

Signal (or RT-Builder, CCSL, MRICDF)

… and Syndex (for allocation and real-time scheduling)
Polychrony

Signal (or RT-Builder, CCSL, MRICDF)

... and Syndex (for allocation and real-time scheduling)
Globally asynchronous implementation of a composition of synchronous modules preserving functional correctness.
Globally asynchronous implementation of a composition of synchronous modules with multiple clocks preserving functional correctness.

**Polychrony**

$c = a \text{ default } b$

$T_c = T_a \cup T_b$

A composition of synchronous modules with multiple clocks.
Globally asynchronous implementation of a dataflow network (merge) of synchronous modules with multiple clocks.

$$c = a \text{ default } b$$

$$T_c = T_a \cup T_b$$

a dataflow network (merge) of synchronous modules with multiple clocks
Globally asynchronous implementation of a dataflow network (sampling) of synchronous modules with multiple clocks.

Polychrony

\[ T_c = T_a \cap T_b \]

a dataflow network (sampling) of synchronous modules with multiple clocks
Globally asynchronous implementation of
- latency insensitive
- scheduling independent synchronous modules
Methodology

synchronous module

synchronous module

synchronous module

Co-modeling

Architecture exploration
Case study of the A350 doors management system

System-level model of the Doors and Slides Control System (SDSCS)

Function
- Monitor doors status via sensors
- Control flight lock actuators
- Manage the residual pressure
- Inhibit incorrect cabin pressure

A safety-critical system
- High-level modeling
- Early validation & verification
- Architecture exploration
SDSCS functional model (Simulink/Geneauto)

**Simulink**
Matlab Simulink and Stateflow, a popular high-level modeling language

**Gene-Auto**
A safe subset of Simulink/Stateflow
Logical time and synchronized data-flow

**A transformation tool-chain with Polychrony**
AADL
An SAE standard for high-level, component-based, architecture modeling: application software, execution platform, composites

ARINC-653
An API for avionic software supporting the partitioned IMA approach

A transformation tool-chain with Polychrony
SDSCS simulation model

Additional models for open system simulation
A simple, non-preemptive, static scheduler
Time intervals are abstracted

Simulation clocks
- Reference clocks
- Period clocks (periodic threads)

VCD interface - Global simulation clock, interactive and of offline modes
Scheduling and distribution with Syndex

Algorithm

Syndex
- Algorithm, architecture, and adequation
- Scheduling analysis and heuristic for adequation
- Automatic code distribution
- Processor-level scheduling and communication, synchronization, ...

Architecture

Mapping
Conclusion and perspectives

Conclusion

- System-level co-modeling with AADL and Simulink/Gene-Auto
- Polychrony as a pivot model-transformation platform
  - Formal polychronous model
  - Automatic model transformations
  - Interoperability between tools
- Simulation with VCD and profiling/scheduling with Syndex

Perspective

- Scheduler synthesis/decompilation with Syndex or RTOS scheduling
- Architecture exploration (performance, energy, ...)
- Formal verification, control synthesis, fault modeling and analysis
- Automatic test case generation
SME, a synchronous modeling environment and open-source Eclipse front-end for Polychrony

A unified model of computation for architecture exploration of integrated modular avionics
• Data-flow for computation
• Mode automata for control
• Libraries for services
• Model-checking
• Controller synthesis

An eclipse interactive interface
• Open import functionalities
• High-level visual editor
• Analysis and transformation visualization and traceability

Component of the OpenEmbeDD platform and CESAR RTP v1.0
Tools and applications

RT-Builder (Geensoft)
Real-time, hardware in-the-loop, simulation of electronic equipments
Polychrony toolbox

- **Design**
  - **Simulink/Gene-Auto**
    - Functional model
  - SME
    - Eclipse Platform
    - Java, Kermeta, ATL

- **Analysis**
  - Fiacre
    - XML model
  - Sigali
    - SIGNAL process

- **Toolbox**
  - Signal Library for AADL
  - SIGNAL Toolbox
    - Compilation
    - Code distribution

- **Scheduling**
  - Signal Library for AADL
  - Syndex

- **Simulation**
  - C, C++
    - GCC
  - Test cases
  - Binaries
  - VCD files

**C communication library**