



C based Modelling of Embedded Mixed Signal Systems

Institut für
Computertechnik

ICT

Institute of
Computer Technology

Ch. Grimm¹, K. Einwich²

¹TU Vienna

²Fraunhofer IIS/EAS Dresden

Motivations for C-Based approaches

- DSP methods include an increasing amount of control flow
 - SPUC, Ptolemy, ...
 - HetSC, ...
- DSP methods and AMS circuits are often functionally interwoven and can only be analyzed as a whole
 - MixSigC, ASC, SEAMS
 - SystemC-AMS

C based Modelling of Embedded Mixed Signal Systems

- **Before (or without) SystemC**
- SystemC based
- What's next?

Very early approaches ...

- Many companies had in-house approaches, often starting in the 90s and maybe even earlier
- Motivation:
 - Simulator coupling backplane
 - Accelerated system simulation
- Example:
AVSL [Meise et al.]

Signal Processing using C++ (SPUC, '93-'05)

- From TI/HP stuff??? (Name of company has been removed ...)
- Motivation
 - Faster System level simulation
 - Allows modelling of complex control flow in C/C++
- Open Source (sourceforge!)
- Most notably a DSP library in C/C++, no analog stuff
 - Modeling Digital Filter Functions, ...

Ptolemy [Lee, Messerschmidt; '90-'97]

- Used to study interaction of different simulators (MoC), Modelling/Simulation/Design of DSP systems
- Discrete-event, process networks, synchronous/reactive, and finite-state machine models of computation in C++
- Generates C-Code (or Assembler) for DSP realization
- Commercial Products: Agilent ADS, Cadence SPW
- → Ptolemy II



Ptolemy [Lee, Messerschmidt; UCB]

The screenshot displays the Ptolemy II simulation environment with several key components:

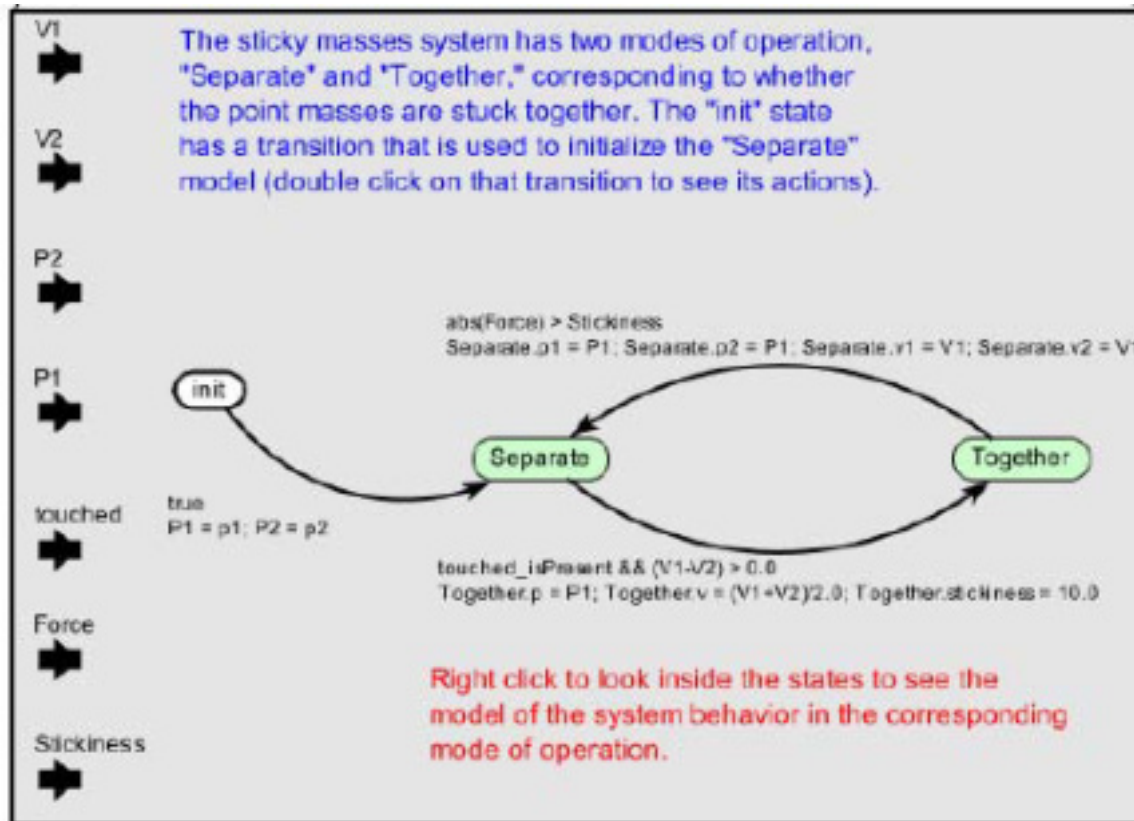
- Block Diagram (Top Left):** Shows a signal flow starting from a 'D.C.' source, passing through an adder (+) and a multiplier (x), and ending at a sink.
- Waveform Analyzer (Top Right):** Displays a digital signal waveform with a time axis from 0 to 200 ns. The signal transitions between high and low states.
- Data Display Window (Middle Left):** Titled 'DSP50002', it shows a table of data points for a simulation. The table includes columns for time (t), value (v), and other parameters.
- Circuit Diagram (Bottom Right):** A detailed schematic of a digital circuit. It features two 'U9458000' components (likely DSP50000 processors) connected to a central bus system. The bus is labeled with 'F', 'NC', and 'HT'. The circuit also includes various control signals and data paths.

At the bottom of the window, a status bar indicates 'SIMULATION IS PROGRESS' and provides instructions: 'Enter Ctrl-C to stop simulation'.

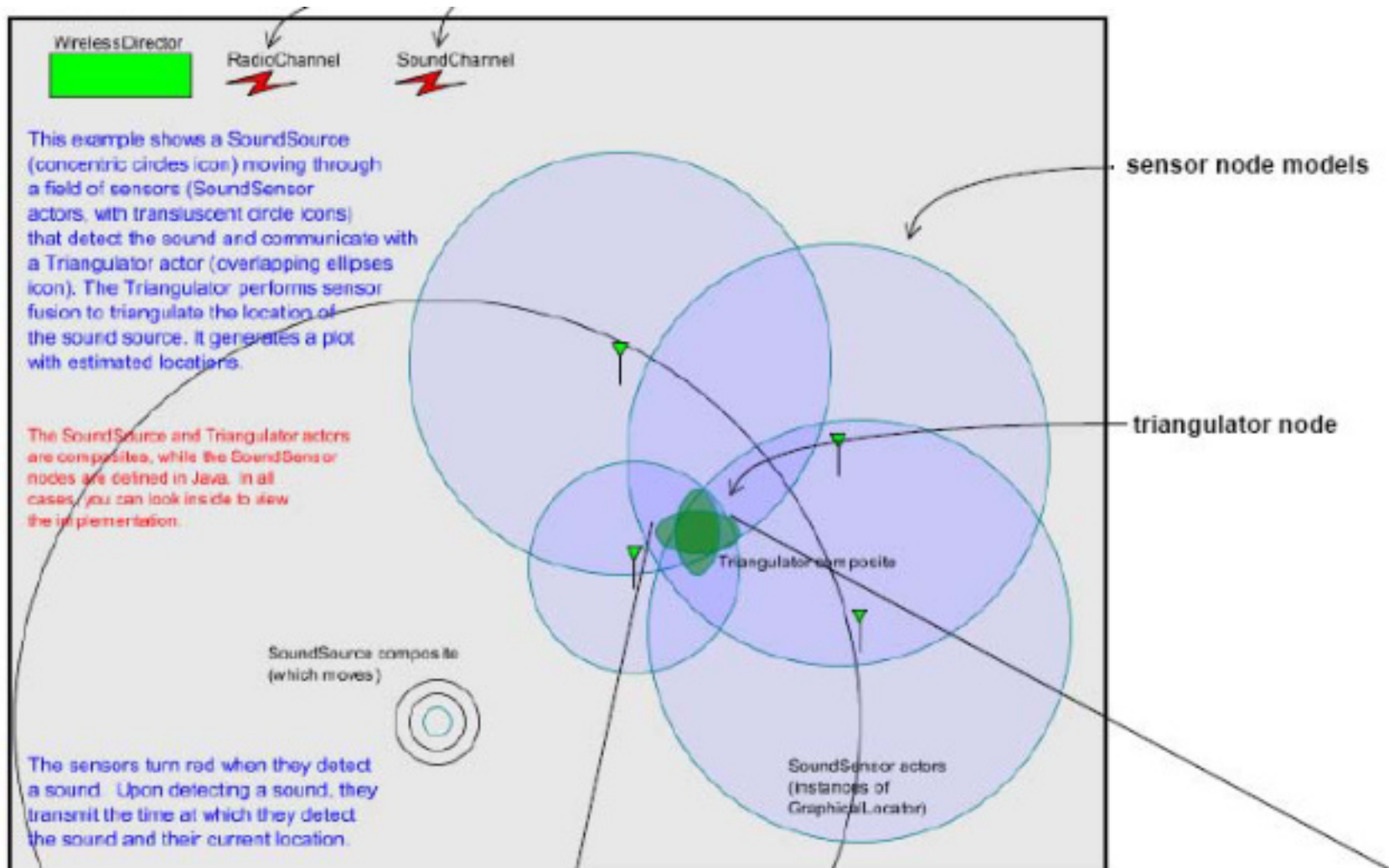
Ptolemy II [Lee et al, 1996-now]

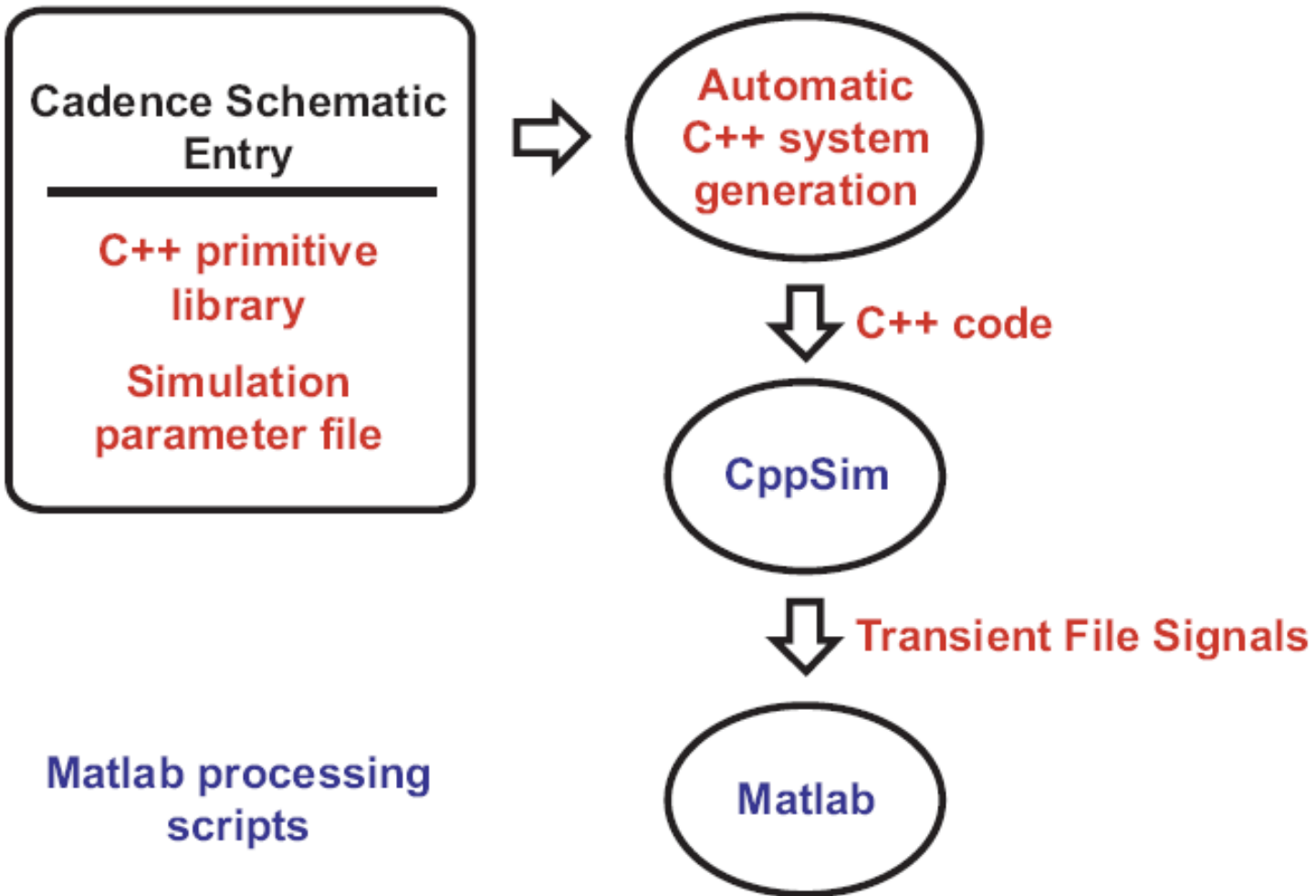
- **Focus: Functional level modelling of DSP and other abstract heterogeneous systems**
- Java based
- CT domain
 - DAE, interaction with DE and FSM domain
- Frequent use of polymorphism to allow use of components in different simulation scenarios

CT + „Mixed-Signal“ MoC in Ptolemy II ...



„Wireless“ MoC in Ptolemy II





C based Modelling of Embedded Mixed Signal Systems

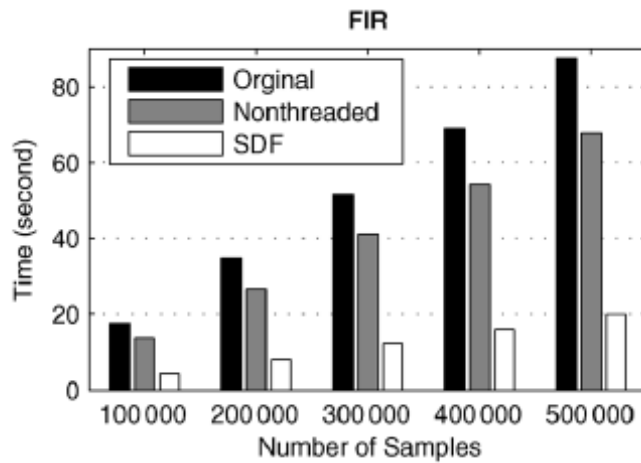
- **Before (or without) SystemC**
- **SystemC-based**
 - DE simulation kernel used
 - Extended kernel capabilities
- **What's next?**

Behavioral modeling and simulation of mixed-signal system-on-a-chip using SystemC [Bonnerud, Ytterda2001]

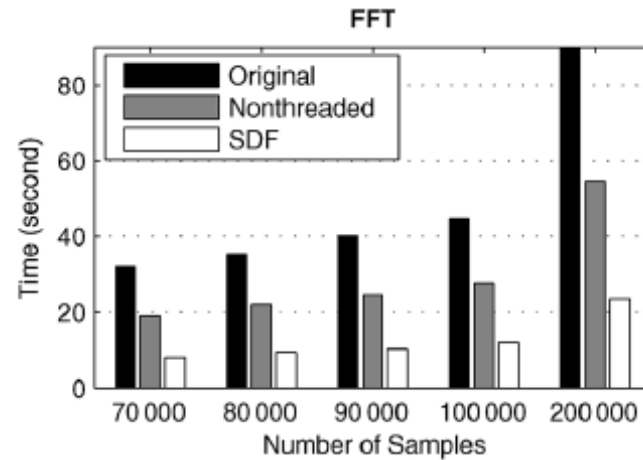
- Purpose:
Behavioural modelling of Mixed-Signal Systems
- No kernel extension
- Block diagrams with components for A/D Converter
Design such as Sample & Hold, ...
- No Netlists, transfer functions
- Use cases: Simulation of a A/D Converter

SystemC Kernel Extensions for Heterogeneous System Modeling [Patel, Shukla; 2002-2004]

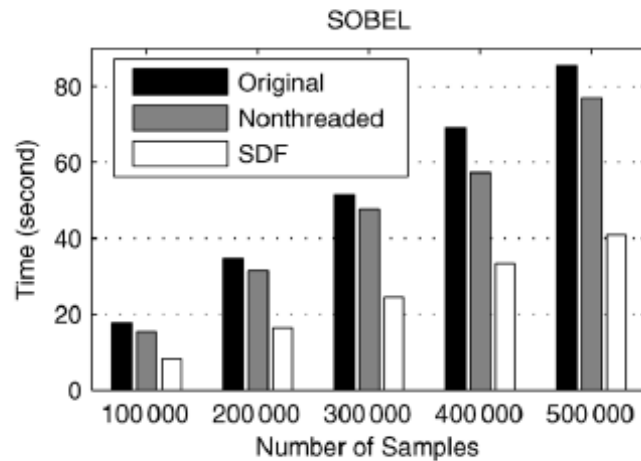
- Purpose: Scientific work ...
- Extends SystemC kernel capabilities
- SDF, FSM, CSP MoC integrated in SystemC-AMS
- MoC can be structured hierarchically
- Lacks support for CT domain
- ... **interesting**: Comparison SDF with + without kernel extension (diophantine solver) →



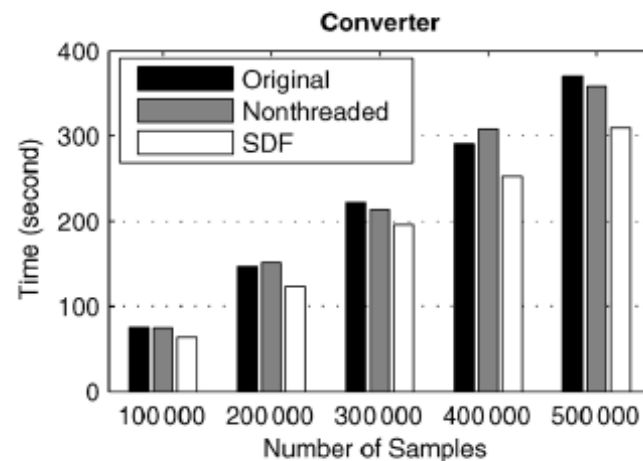
(a)



(b)



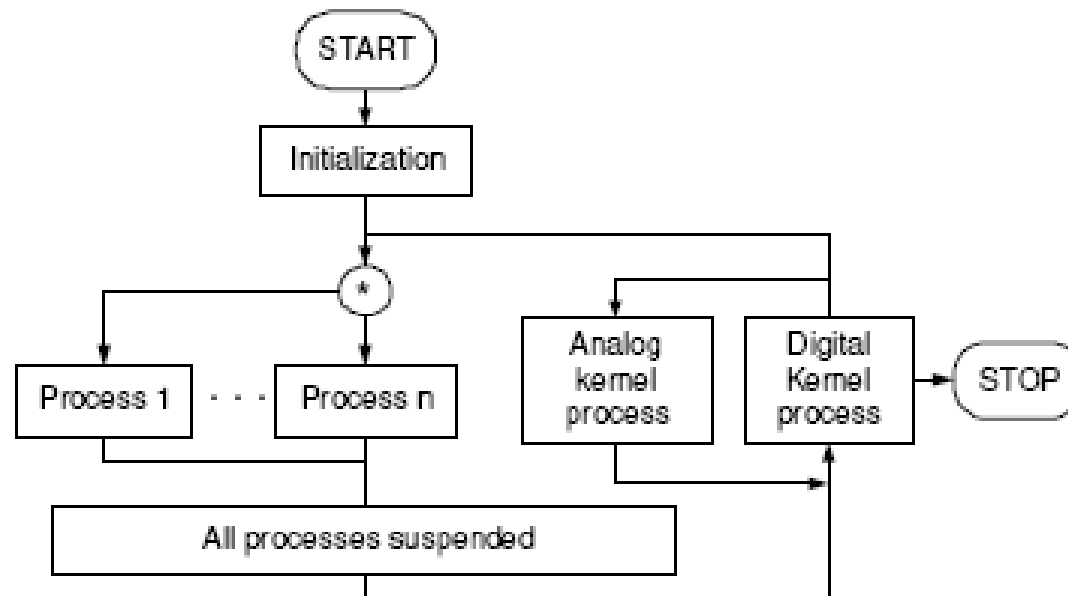
(c)



(d)

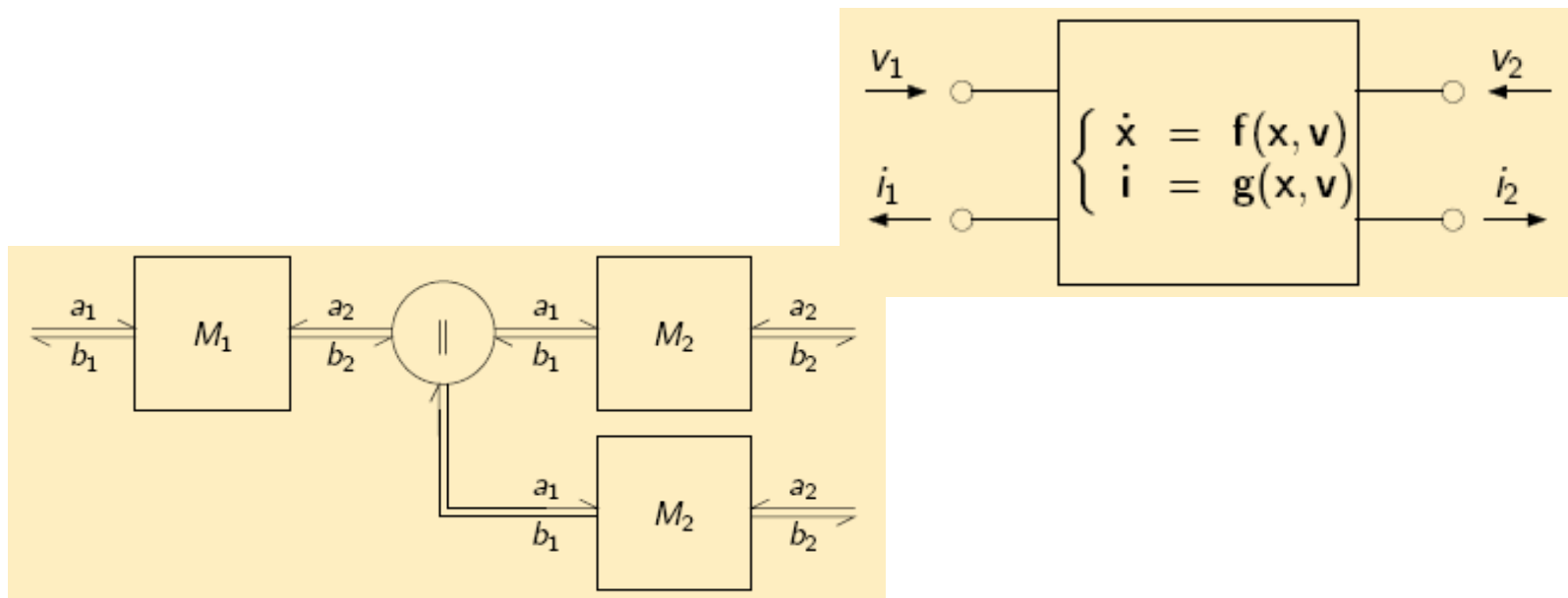
SEAMS [T. Kazmierski et al, 2004]

- Purpose: Modelling + Simulation of AMS systems
- SystemC Environment with integrated AMS solver
- Network elements, Nodes
- <http://eprints.ecs.soton.ac.uk/9413/>



SystemC-WMS [Orcioni,Conti, 2003-now]

- Hard to classify ... distributed solver, mostly DE
- Two-ports connected via wavechannels solve equation systems in DE simulator.
- <http://www.deit.univpm.it/systemc-wms/>



- Application:
 - HW/SW Codesign, SW synthesis
- Class library of MoCs based on the DE MoC
- **DE MoC implements**
 - SDF, SR, DDF, ... MoC
- Channels can convert MoCs where necessary



- started 2001-2006, 2007-
- Symbolic preprocessing with MAPLE
 - reads SABER netlist
 - linearized equations in different workpoints
 - symbolically solves equation system
 - generates C-Code that
 - Computes output iff output signal is read
 - Computes states iff workpoint is left
 - Really fast ...

ASC library [Grimm, Waldschmidt; 2001-2004]

- Analog/Mixed-Signal SystemC
 - Block diagrams, transfer functions, step width control + refinement of implicit control to control signals
 - Manual scheduling or 1 delay/block ...
 - Focus:
 - Investigation of Refinement Methodology for AMS and multi MoC systems
 - OO-Design
 - Interface Classes allow designer to change MoC
 - Signals in this case adapt automatically (Polymorphic signals)

MixSigC [Einwich et al., 2000-2003]

- Cooperation Infineon + FhG-IIS/EAS Dresden
- SDF Engine + Linear Solver
- Mostly C-Based, no OO Design
- 2003 re-implementation with improved OO design, interface classes as evaluated in ASC
- ~2003 initial SystemC-AMS SG prototype
- 2005 submitted to OSCI

SystemC-AMS SG prototype

- Joint effort by SystemC-AMS SG
 - Base on re-implementation of MixSigC
 - New, innovative OO design with some features from ASC
- Starting point for SystemC-AMS WG effort

C based Modelling of Embedded Mixed Signal Systems

- Before or without SystemC
- SystemC-Based
- What's next?

Open issues (besides standardization ...)

