

# Struktur einer toolunabhängigen Modellbibliothek in VHDL-AMS

Vortrag beim ASIM-Fachgruppentreffen  
"Simulation Technischer Systeme"  
am 11. März 2003 in Ulm

Hans Gall, BAUSCH-GALL GmbH, München

# Struktur einer toolunabhängigen Modellbibliothek in VHDL-AMS

- Einleitung
- Ziele
- Stand der VHDL-AMS-Modellierung
- Anforderungen an die Modellbibliothek
- Rahmenbedingungen
- Realisierungsvorschlag
- weitere Arbeiten
- Zusammenfassung und Ausblick

# Einleitung

- IEEE-Standard VHDL-AMS
- Tool-Hersteller unterstützen VHDL-AMS
- AG VHDL-AMS unterstützt frei verfügbare Modellbibliothek
- Recherche durchgeführt
- Vorbereitungsphase abgeschlossen
- Kodierung der Modelle beginnt

## Ziele

- VHDL-AMS-Nutzung verbreiten
- Aufbau und Pflege einer Modellbibliothek
- öffentliche Bibliothek verfügbar
- Modellaustausch ermöglichen
- keine Verwendung graphischer Symbole
- Niveaustufen für Architectures, schrittweise
- Modelle möglichst bald bereitstellen

## Stand der VHDL-AMS-Modellierung

- Internet-Recherche durchgeführt
- mehrere hundert Links durchgesehen
- wenig verwertbare Modelle gefunden
- Beispiel Lauritzen-Diode
- Beispiel Cooper
- Modelle entsprechen kaum industriellen Anforderungen

## Stand der VHDL-AMS-Modellierung, verwertbare Informationen

- [www.mentor.com/systemvision](http://www.mentor.com/systemvision)
- VeriasHDL-Packages von Synopsys
- Tutorial von Prof. Huss, TU Darmstadt
- Kapitel "Systemsimulation" von Dr. Peter Schwarz im "Handbuch der EDA"
- Lastenheft "Generatormodell" erhalten von Herrn Sterner

## verwertbare Informationen (Forts.)

- Buch "The Designer's Guide to Analog & Mixed-Signal Modeling – Illustrated with VHDL-AMS and MAST" (Fa. Avant!)
- Skripten des Saber-User-Group-Treffens
- Modelica-Bibliothek von Dr. Clauss und Dr. Schneider (FhG Dresden)

# Anforderungen an die Modellbibliothek, Inhalt

- Grundmodelle SPICE3
- Modelle für das Kfz-Bordnetz
  - Batterie
  - Leitung
  - Schalter
  - DC-/DC-Wandler
  - Generator



## Anforderungen an die Modellbibliothek, Inhalt (Forts.)

- mechanische Grundmodelle
  - Feder-Masse-Dämpfer (SMD)
  - Feder-Masse-Dämpfer mit Kennlinie
- Modelle für den Kfz-Antriebsstrang

## Inhalt (Forts.): parametrisierbare Modelle (Templates)

- Elektrik: Schalter, Relais, Lampen, Sicherungen
- thermische Grundmodelle
- Grundmodelle der Hydraulik
- Grundmodelle der Regelungstechnik für Signale  
(nicht-konservativ)
- Grundmodelle der digitalen Regelungstechnik

# Inhalt (Forts.): parametrisierbare Modelle (Forts.)

- Schnittstellen
- Koppелеlemente (z.B. elektrisch – mechanisch)
- Quellen: elektrisch, mechanisch, hydraulisch, Signale, von Datei
- DC/DC-Wandler
- elektrische Motoren und Generatoren
- elektrische Stecker
- Leitungsmodelle

## Anforderungen an die Modellbibliothek, Vorgaben für Modellentwickler

- Zusammenarbeit mehrerer Modellentwickler erfordert einige Richtlinien
- Modellentwickler muß bei Übergabe liefern
  - VHDL-AMS-Modell
  - aussagekräftige Dokumentation (sh. unten)
  - VHDL-AMS-Testumgebung
  - Vergleichsrechnungen, falls sinnvoll

## Anforderungen an die Modellbibliothek, Modelldokumentation

- Dokumentation in HTML oder XML
- druckbare Dokumentation als pdf-Datei
- Entwickler dokumentieren nur ungern und erfahrungsgemäss unzureichend, daher muss die Erstellung der Dokumentation weitmöglichst automatisch erfolgen

# Beispiel für Dokumentation

```
-- Constant Thermal Resistance
-- the rtherm template models a constant thermal resistance across two
-- thermal pins (th and tl). The product of this quantity and the power
-- flow (in W) between these pins is the temperature (in C) across
-- them.</DESCRIPTION>
```

```
-- Developer: Delphi Deutschland GmbH (PES)
-- <DEVELOPER>Thomas Reinders</DEVELOPER>
```

```
LIBRARY ieee;
USE ieee.math_real.all;
LIBRARY disciplines;
USE disciplines.thermal_systems.all;
USE disciplines.all;
```

# Beispiel für Dokumentation (Forts.)

```
ENTITY rtherm IS
  GENERIC (rth: real := 1.0); -- thermal resistance (must be
    nonzero)
  PORT (TERMINAL th,t1 : THERMAL);
END ENTITY rtherm;

ARCHITECTURE simple OF rtherm IS QUANTITY delta_t ACROSS
  heatflux THROUGH th TO t1;
BEGIN
  delta_t == rth * heatflux;
END ARCHITECTURE simple;
```

# Rahmenbedingungen

- Bibliothekskode soll auf allen VHDL-AMS-Simulatoren laufen
- Simplorer (hAMSter)
- VeriasHDL (Synopsys)
- SystemVision (Mentor)
- weitere
- keine toolunabhängigen Graphikeingabeprogramme
- bei Modelldefinition Eigenschaften der Nicht-VHDL-AMS-Simulatoren berücksichtigen



# Dokumentation

- so einfach wie möglich für den Modellentwickler
- Vermeidung doppelten Codes
- einfache Wartbarkeit
- Formate: PDF + in Browser sichtbar (XML, HTML)

## Weitere Arbeiten

- Zur Zeit Kodierung einer Prototypbibliothek
- Untersuchung der SystemVision-Modelle
- Erstellen eines Templates für Modellentwickler
- Befragung von Experten

## Zusammenfassung und Ausblick

- Vorbereitungsarbeiten beendet
- Kodierung, Sammeln und Testen der Modelle kann beginnen
- Vorgehen bei Dokumentation jetzt klar
- Informationen über geeignete Modelle?