MOS-AK Workshop
May 7th, 2004, Stuttgart

Handling of Device Models in VHDL-AMS

Joachim Haase
Fraunhofer Institute Integrated Circuits
Branch Lab Design Automation (EAS)
Dresden

Joachim.Haase@eas.iis.fhg.de
Outline

- Starting point and objectives
- Recap of VHDL-AMS basics
- Spice compatibility in behavioral languages
- Proposal to handle Spice-like models in VHDL-AMS
- Problems and further direction
Starting Point

- WG „Simulation of heterogeneous systems using VHDL-AMS“
  http://www.bausch-gall.de/vhdl-ams.htm

- Exchange of models between suppliers and OEM’s
  - Behavioral models of parts of the electrical system
    (battery, loads, sensors, actors, ...)
  - Usage of different commercial VHDL-AMS simulators

- Problem:
  Re-use of parameterized Spice models in VHDL-AMS netlists

- Proposal „Spice Components in VHDL-AMS“
Objectives

- Possibility to exchange a VHDL-AMS netlist with Spice-like models between different simulation programs
- Specification of the interface of Spice-like models
- Parameters in accordance with Spice3f
- Base for the discussion on implementation of Spice-like models in commercial VHDL-AMS simulation tools
- Consideration of handling Spice primitives in Verilog-AMS
VHDL-AMS is a superset of VHDL. Concepts and language constructs of digital VHDL stay valid.
Supported Modeling Approaches

Mathematical description

- Conservative networks (Kichhoffian networks)
- Nonconservative systems (Signal-flow diagrams)

Possible mixed descriptions are:
- analog-digital
- elektrical–non-electrical
- conservative-nonconservative
- ...

Implementation of models

- VHDL-AMS
- Verilog-AMS
- ...
Solution of Conservative Systems

Configuration space
(potential voltages and currents)

Kirchhoff space
(set of branch voltages and currents that fulfill KVL and KCL)

Voltage-current relation
(considers all branch constitutive relations)

\[ F(x(t), x'(t), t) = 0 \]
Element of Configuration Space - Quantity

Values of type
- REAL
- REAL_VECTOR

Analog waveforms are piecewise continuous

Additional conditions to calculate initial values
- in the quiescent domain (determination of operating point)
- after discontinuities

Calculation of values at analog solution points (ASP) by the analog solver

Time line of type REAL
library IEEE;
use IEEE.ELECTRICAL_SYSTEMS.all;

entity RESISTOR is
  generic (R : REAL);
  port (terminal P, N : ELECTRICAL);
end entity RESISTOR;

architecture A0 of RESISTOR is
  quantity V across I through P to N;
begin
  V == R * I;
end architecture A0;

Model of a Conservative System

context clause

branch quantity declaration

simultaneous statements

V = R \cdot I

VHDL-AMS Basics
Free Quantities in VHDL-AMS Models

entity DIODE is
  ... 
  port (terminal A, C : ELECTRICAL);
end entity DIODE;

architecture A0 of DIODE is
  ...
  terminal I : ELECTRICAL;
  quantity V_RI across I_RI through A to I;
  quantity V_D across I_CI, I_ID, I_GI through I to C;

  quantity CI : REAL;

begin
  CI == CalcCI(V_D, m_dN, m_dTT, m_dM, m_dFC, m_dCJO_T, m_dIS_T, m_dVBKORR, m_dVJ_T, m_dVT_T);
  I_CI == CI * V_D'DOT;
end architecture A0;
Entity/Architecture Concept

entity entity_name is
  generic (paramter_list)
  port (port_list)
  ...
end entity entity_name;

architecture nameA of entity_name is
  { declaration_part}
begin
  { concurrent_statement }
end entity entity_name;

architecture nameN of entity_name is
  { declaration_part}
begin
  { concurrent_statement }
end entity entity_name;
Further Possibilities in VHDL(-AMS)

- **Functions** to handle complex behavior
  - Determine values in a sequential manner

- **Type declaration**
  - User-declared types are possible

- **Overloading** of functions and operators
  - Same names for functions with different numbers or types of parameters

- **Packages**
  - Collection of related declarations (types, constants, functions, ...)
  - External view in package header
  - Implementation in package body
Logical Libraries

- Working library (WORK)
- STD
- IEEE
- DISCIPLINES
- "..."

Resource libraries:
- Company-wide libraries
- User-defined libraries
Spice Compatibility Problems

- Spice is not a single language – family of related languages
- Modification of the language and of the device parameters by many EDA vendors
- A great deal of incompatibility among the Spice languages
  - Names of built-in primitives (can) differ
  - Names of parameters (can) differ
  - Names of ports (can) differ
- Different Spice languages should be supported

Some Requirements in Device Modeling

- Handling of model parameters
  - Defined on a .model card in Spice

- Model initialization
  - Parameter defaulting and range checking

- Handling of instance parameters
  - Defined on device instance

- Instance initialization
  - Parameter defaulting and range checking

- Interaction with Simulator Variables/Algorithms

Approach to Handle Models in VHDL-AMS

package SPICE_PARAMETERS

- Model initialization
- Parameter defaulting and range checking

package SPICE_COMPONENTS

- Parameter defaulting
- Port names

Compilation into logical library SPICE2VHD
Example

Spice netlist

```
.MODEL VERTNPN NPN BF=80 IS=1E-18 RB=100 VAF=50 + CJB=3PF CJC=2PF CJS=2PF TF=0.3NS TR=6NS

.SUBCKT diffPair c1 b1 e c2 b2
Q1 c1 b1 e vertNPN
Q2 c2 b2 e vertNPN
.ENDS
```

Spice-like models in VHDL-AMS
library IEEE, SPICE2VHD;
use IEEE.ELECTRICAL_SYSTEMS.all;
use SPICE2VHD.SPICE_PARAMETERS.all;
use SPICE2VHD.SPICE_COMPONENTS.all;

entity diffPair is
  port (  
    terminal c1, b1, e, c2, b2 : ELECTRICAL 
  );
end entity diffPair;
architecture a0 of diffPair is

constant vertNPN : BJT_DATA := SET_BJT_DATA (MODEL => NPN, BF => 80.0, ISS => 1.0E-18, RB => 100.0, VAF => 50.0, CJE => 3.0E-12, CJC => 2.0E-12, CJS => 2.0E-12, TF => 0.3E-9, TR => 6.0E-9);

begin

Q1: BJT generic map (vertNPN)
    port map (c1, b1, e, electrical_ref);

Q2: BJT generic map (MDATA => vertNPN)
    port map (NC => c2, NB => b2,
               NE => e, NS => electrical_ref);

end architecture a0;

Spice-like models in VHDL-AMS
architecture a1 of diffPair is

constant row1    : STRING   :=
".MODEL VERTNPN NPN BF=80 IS=1E-18 RB=100 VAF=50";
constant row2    : STRING   :=
"+ CJB=3PF CJC=2PF CJS=2PF TF=0.3NS TR=6NS";
constant vertNPN : BJT_DATA := SET_BJT_DATA (row1 & row2);

begin

Q1: BJT generic map (vertNPN)
    port map (c1, b1, e, electrical_ref);

Q2: BJT generic map (MDATA => vertNPN)
    port map (NC => c2,
               NB => b2,
               NE => e,
               NS => electrical_ref
    );

end architecture a1;
VHDL-AMS - Version 3 (1)

File modeldata.inc

```
.MODEL VERTNPN NPN BF=80 IS=1E-18 RB=100 VAF=50
+ CJB=3PF CJC=2PF CJS=2PF TF=0.3NS TR=6NS
```

Spice-Netzliste

```
.include modeldata.inc

.SUBCKT diffPair c1 b1 e c2 b2
Q1 c1 b1 e vertNPN
Q2 c2 b2 e vertNPN
.ENDS
```
architecture a2 of diffPair is

constant vertNPN : BJT_DATA := SET_BJT_DATA (MNAME => "VERTNPN", FILENAME => "modelfdata.inc");

begin

Q1: BJT generic map (vertNPN)
port map    (c1, b1, e, electrical_ref);

Q2: BJT generic map (vertNPN)
port map    (c2, b2, e, electrical_ref);

end architecture a2;
Package SPICE_PARAMETERS (1)

- Types for declaration of constants with model card data
  - RESISTOR_DATA
  - DIODE_DATA
  - BJT_DATA
  - ...

- Functions to initialize constants with model data
  - SET_RESISTOR_DATA
  - SET_DIOODE_DATA
  - SET_BJT_DATA
  - ...

Spice-like models in VHDL-AMS
Package SPICE_PARAMETERS (2)

type MODEL_TYPE is (
    UNDEF, -- model type is not defined
    R, -- semiconductor resistor model
    C, -- semiconductor capacitor model
    SW, -- voltage controlled switch
    CSW, -- current controlled switch
    URC, -- uniform distributed rc model
    LTRA, -- lossy transmission line model
    D, -- diode model
    NPN, -- npn BJT model
    PNP, -- pnp BJT model
    NJF, -- n-channel JFET model
    PJF, -- p-channel JFET model
    NMOS, -- n-channel MOSFET model
    PMOS, -- p-channel MOSFET model
    NMF, -- n-channel MESFET model
    PMF -- p-channel MESFET model
);

Type declaration corresponds to different .model cards
Package SPICE_PARAMETERS (3)

type BJT_DATA is array (NATURAL) of REAL;  -- implementation dependent

function SET_BJT_DATA (  
  constant MODEL : MODEL_TYPE;  -- type of BJT (NPN|PNP)  
  constant ISS  : REAL := 1.0E-14;  -- saturation current (in A)  
  constant BF   : REAL := 100.0;  -- ideal maximum forward BETA  
  constant NF   : REAL := 1.0;  -- forward current emission coefficient  
  constant VAF  : REAL := REAL'HIGH;  -- forward Early voltage (in V)  
  ...  
) return BJT_DATA;

function SET_BJT_DATA (  
  constant MODELCARD : STRING  -- model card information as string  
) return BJT_DATA;

function SET_BJT_DATA (  
  constant MNAME  : STRING;  -- model card searched in FILENAME  
  constant FILENAME  : STRING  
) return BJT_DATA;

Initialization of VHDL-AMS models w.r.t. Spice3f

Initialization for tool-dependent implementations
Package SPICE_COMPONENTS (1)

- Interface descriptions
  - Instance parameter and initial values correspond to SPICE3 Version 3f3 User’s Manual (May, 1993)

- Compilation of both packages into logical library SPICE2VHD
Package SPICE_COMPONENTS (1)

```vhdl
component BJT

  generic (
    MDATA    : BJT_DATA    := MDATA_DEFAULT; -- model data
    AREA     : REAL       := 1.0;          -- area factor
    START    : START_TYPE := UNDEF_VALUE;  -- starting condition
    IC_VBE   : REAL       := UNDEF_VALUE;  -- initial condition for VBE
    IC_VCE   : REAL       := UNDEF_VALUE;  -- initial condition for VCE
    TEMP     : REAL       := AMBIENT_TEMPERATURE
  );

  port ( 
    terminal NC : ELECTRICAL;
    terminal NB : ELECTRICAL;
    terminal NE : ELECTRICAL;
    terminal NS : ELECTRICAL );

end component BJT;
```

Handling of optional terminals not quite clear
But: Terminals may be unconnected
Further Approaches

- No access to simulator variables/algorithms
  - TSTEP => 0.0 or 1.0E-15
  - TSTOP => REAL‘HIGH
  - 1/TSTOP => 0.0
  - Infinite => REAL‘HIGH
  - TEMP => AMBIENT_TEMPERATURE from Package MATERIAL_CONSTANTS

- Handling of model and instance parameter
  - Default value => can be overwritten
  - No default value => value assignment required
  - Detect specification => default to UNDEF
    (REAL‘LOW if parameter is REAL)
Expected Spice Models in HDL‘s

- Basic elements
  - RESISTOR, SEMICONDUCTOR_RESISTOR
  - CAPACITOR, INDUCTOR

- Controlled sources and lossless line
  - VCVS, VCCS, TLINE

- Independent voltage sources
  - VDC, VEXP, VPULSE, VPWL, VSINE, VSFFM

- Independent current sources
  - IDC, IEXP, IPULSE, IPWL, ISINE, ISFFM

- Device models
  - DIODE, BJT, JFET
  - MOSFET, MESFET
Further Directions

- Improvement of the proposal to implement „Spice Components in VHDL-AMS“

- Implementation of selected (simplified) models in VHDL-AMS on the base of Spice3f5

- Discussion on access to Spice models in existing tools