

VHDL-AMS Modeling for Harmonic Balance Circuit Simulation.

Gennady Serdyuk*, Boris Shelkovnikov #

Abstract – This paper present researches in analog VHDL-AMS modeling for purposes of Harmonic Balance circuit simulation and formulates frequency-domain extensions, which increase language modeling capabilities and makes possible to simulate circuits contained complex frequency-dependent components .

I. INTRODUCTION

During last decades substantial efforts spent in field of developing hardware description languages. For discrete circuit simulation progress has been achieved quite long time ago with developing of VHDL and Verilog hardware description languages. The other situation is in analog realm. Meanwhile SPICE input format is de-facto standard, there are a lot of dialects developed as long as proprietary languages for special purposes, including RF and MW modeling. That makes hard to switch between simulators during design process while no one simulator proposes all necessary features and analyses for hardware designers. Even more, simulators shared common language (say SPICE-like) often propose different device models, which make hard to compare results of simulations. That stimulates development of analog hardware description languages and two analog HDLs get status of standard – VHDL-AMS [1] and Verilog-A. That was important achievement, as opened way to standardization of device libraries and development of model exchange space [3]. Both languages are defined as languages for time-domain modeling and does not support frequency domain, required by most RF&MW tasks. VHDL-AMS is designed to define digital and mixed (digital-analog) circuits and does not contain strictly defined analog subset.

This paper aims to show, that there is limited subset of VHDL-AMS, which is sufficient to describe pure analog circuits and to propose simple VHDL-AMS extension to get advantages of frequency-domain modeling and simulation.

II. HB TECHNIQUE

Let us formulate continuous (analog) circuit equation:

$$i(v(t)) + \frac{dq(v(t))}{dt} + \int_{-\infty}^t y(t-\tau)v(\tau)d\tau + u(t) = 0 \quad (1)$$

or

$$f(v, t) = 0 \quad (2)$$

Where $v(t)$, $u(t)$, $y(t)$, $i(v(t))$, $q(v(t))$ and $f(v, t) \in R^N$, $V(t)$ and $u(t)$ are assumed to be T periodic, $y(t)$ is linear response of linear subcircuit and $i(v)$ and $q(v)$ are only nonlinear members.

Considering both v and f for (2) as Fourier series and making necessary transformations, Harmonic Balance (HB) equations may be written as the following:

$$F(V) = I(V) + \Omega Q(V) + YV + U = 0 \quad (3)$$

Gennady Serdyuk – Ridgetop Group, Inc., Kiev office, Shamrila St, 6, suite 78, Kiev, 04112, Ukraine, e-mail: gserdyuk@ridgetop-group.com

Boris Shelkovnikov – Institute of Telecommunicatin systems NTTU «KPI», Industrial per., 2, Kiev, 03056, Ukraine, e-mail: shelk@ukr.net

Where $F(V), I(V), Q(V), V, U \in C^{(K+1)N}$, K number of considered frequencies, Y, Ω are square block matrices:

$$\Omega_m = \text{diag}\{0, \dots, j\omega_n, \dots, j\omega_K\} \quad (4)$$

$$Y_m = \text{diag}\{Y(0), \dots, Y(\omega_n), \dots\} \quad (5)$$

In Equation (3) first term represents currents of nonlinear devices, second one represents nonlinear capacitors and third one is currents of linear subcircuit. U is image of excitation sources.

As most nonlinearities are modeled in time domain, first two members of Eq. (3) are usually computed via time domain using Fast Fourier Transformations (FFT) in the following manner:

$$I(V) = \Phi \{ i(\Phi^{-1} \{ X \}) \} \quad (6)$$

Same for $Q(V)$:

$$Q(V) = \Phi \{ q(\Phi^{-1} \{ X \}) \} \quad (7)$$

Equation (3), including (4-7) is solved by exact or inexact Newton method as described by many authors [4,6,7]:

$$\|F'(X_c)s_N + F(x)\| \leq \eta_c \|F(X_c)\| \quad (8)$$

Where X_c – current point, F' – Jacobian matrix and s_N – inexact Newton step from current point.

To calculate $F'(X_c)s_N$ iterative linear solvers are used.

III. VHDL-AMS ANALOG SUBSET

While VHDL-AMS standard [1] does not separate language onto digital and analog parts, it is possible to distinguish three main language parts, intended to serve different purposes: describe digital (discrete) behavior of components, describe analog (continuous) behavior of components and describe circuit (system) structure itself. Last part handles circuit decomposition into components. It supports construction ability of the language.

Discrete part is implemented using signals and concurrent statements. Analog part is implemented through quantities (say – unknowns of circuit), terminals (special kind of quantities, supports conservation semantic) and simultaneous statements (represent equations). Third, circuit descriptive part, is represented with component instantiation statement and intended to define topology of the circuit.

Any analog (sub)circuit may be described using two approaches: construct it from building blocks (usual for EDA users way) or define equations to describe behavior of such block. Surely, at lowest level all components should be described using equations. This, equation-level representation creates a modeling basis for design of component libraries. Designer can build equations of any complexity, calling functions as necessary, branching calculations, using both simple (one line) simultaneous statement and procedural one, which creates equation from sequential calculations. Designer can create own function using other languages and use them in VHDL-AMS also.

Given the equation support, designers can build higher-level blocks mixing lower-level blocks and equations, using quantities and terminals to interconnect blocks.

Consider example. To define a diode:

```
entity diode is
    port (terminal a, c: electrical);
    generic (I0, a: real);
end entity;
-- entity interface
-- I-V parameters

-- entity may have many different architectures
architecture BEHAV_1 of diode is
    quantity U across Id through a to b;
begin
```

```

Id == I0*(exp(a*U)-1);          -- equation
end;

```

Somewhere may be used:

```

...
terminal t1, out2: electric;
...
R1: entity diode (BEHAV_1)
    port map (t1, out2) generic map(1.e-12,35);
...

```

In addition to all former, language support some attributes, which, being applied to quantities allow to get their derivatives, integrals and delays. For example, if it is defined:

```

quantity a: real;

```

somewhere in code may be used:

```

...a'dot...          -- derivative of a
...a'integ...       -- integral of a
...a'delayed(1.e-6)... -- a delayed onto 1 us

```

For example, model of nonlinear capacitor may look like that:

```

entity NL_Cap is          -- entity interface
port(terminal a,b: electrical);
generic (tau, I0, a,: real); -- parameters
end entity;

-- entity may have many different architectures
architecture BEHAV_1 of NL_Cap is
quantity U across Ic through a to b;
begin
Ic == tau*I0*(exp(a*U)-1)*a*U'dot; -- equation
end;

```

All above mentioned makes quite attractive usage of VHDL-AMS subset as modeling language for Harmonic Balance simulator. That subset allows description of nonlinear dynamic behavior in time domain, correspondent to nonlinear integro-differential equations with delays. Applying Equations (6) and (7) one may transfer it to frequency domain and use directly in HB equation.

Extension statements, allowing frequency-domain modeling are described in the next section

IV. FREQUENCY DOMAIN MODELING EXTENSION

As VHDL-AMS is defined in time domain entirely, it does not contain means to define behavior in frequency-domain. Meanwhile, a lot of circuits in microwave and RF field shows complex frequency dependent behavior hard to be modeled in time domain. Most transmission lines, including strip-, microstrip-, slotlines, stubs and other resonators, T- and X- junctions, directional couplers show complex frequency-domain behavior unable to be modeled in time domain.

To extend language capabilities, additional attribute is proposed, which represent image of certain quantity in frequency domain.

Let quantities i_c and u_c are defined and represent current and voltage over capacitor. Then i_c' FD and u_c' FD will represent spectrum of i_c and u_c respectively and may be used in equations:

```

i_c'FD == 2*math_j*math_p*FREQUENCY*C*u_c'FD;

```

Where FREQUENCY is predefined language function returning value of current frequency.

More general:

```
x'FD==func(FREQUENCY,a,b,c)*y'FD;
```

where func() may be user defined.

Similar (even more flexible) approach is available using VHDL-AMS procedural simultaneous statements:

```
procedural is
variable imp: complex;
begin
a := func1();           -- A and b may be calculated
b := func2();           -- using conventional
imp := func(FREQUENCY,a,b,c); -- sequential statements.
x'FD :=imp*y'FD;       -- Here is implied equation.
end procedural;
```

That syntax allows construct equations in sequential manner, conventional for programmers used to procedural languages.

Circuit impedance or admittance values may be calculated during simulation or picked up from external source – as file or complex database.

It is necessary to state, that one assumption is made here – it is assumed that of any simultaneous statement which use quantity with attribute 'FD indeed is defined in frequency domain too, so it is impossible to mix domains inside one statement (equation). That does not contradict with usual practice, as (as best of authors knowledge) models are defined in one domain only.

Former extension corresponds to 3-rd term of Eq.(3) and makes broader the class described circuits, including distributed parameters circuits and other linear circuits, which may be described in frequency-domain.

V. CONCLUSION

VHDL-AMS subset suitable for RF analog circuit simulation by Harmonic Balance technique is described along with possible frequency domain modeling extensions.

Approach gives increasing flexibility in RF&MW component modeling and broadens the class of tasks to be described by language allowing frequency domain description of circuit components.

That set is called VHDL-AMS/FD and implemented in “Rincon” RF circuit simulator [5].

IV. ACKNOWLEDGEMENT

Parts of present work performed under contract with Ridgetop Group, Inc.

REFERENCES

- [1] VHDL-AMS Language Reference Manual, IEEE Standard No.: 1076.1-1999
- [2] G. Serdyuk, D. Goodman, “VHDL Approach Improves Nonlinear Simulation”, *Microwaves & RF*, November 2001, pp. 76-102.
- [3] M. Mierzwinsky et al, “Changing the Paradigm for Compact Model Inegration in Circuit Simulators Using Verilog-A”, <http://www.tiburon-da.com/NanoTech2003.pdf>
- [4] Ken Kundert, “Simulation Methods for RF Integrated Circuits”, ICCAD-94.
- [5] Rincon User Manual, Ridgetop Group, Inc., <http://www.ridgetop-group.com>
- [6] C.T. Kelley, “Iterative Methods for Linear and Nonlinear Equations”; *Frontiers in Applied Mathematics*, vol 16, SIAM, 1995.
- [7] V. Rizzoli et al, “Fast and Robust Inexact Newton Approach to the Harmonic-Balance Analysis of Nonlinear Microwave Circuits.” 1997 *Microwave and Guided Wave Letters* 7.10 (Oct. 1997 [MGWL]): 359-361.