Compact device modeling for established and emerging technologies with the Qucs GPL circuit simulator

M. E. Brinson
Qucs development team
Faculty of Computing
London Metropolitan University
London, UK
mbrin72043@yahoo.co.uk

S. Jahn
Qucs project manager
Munich
Germany
stefan@lkcc.org

Abstract—Current trends in circuit simulation technology suggest a strong movement towards software packages which promote equation-defined compact semiconductor device modeling and circuit macromodeling. Today, the Verilog-A subset of the Verilog-AMS hardware description language is one of the most popular choices of hardware description language for model construction. The "Quite universal circuit simulator" (Qucs) is a GPL software package supporting the MOS-AK Verilog-A standardization initiative. This paper outlines how equation centered modeling can act as a strong support vehicle for the construction of new device models and circuit macromodels. The material presented in the text explores the relationship between device model specifications, equation-defined devices and Verilog-A code generation. A number of examples demonstrate the capabilities of the model construction tools implemented by the Qucs development team.

Index Terms—**Quite universal circuit simulator (Qucs), equation-defined device modeling, compact semiconductor models, circuit macromodels, Verilog-A**

I. Introduction

High quality models are an essential prerequisite for accurate circuit simulation of established and emerging technology devices. Current trends in modeling [1] indicate a growing acceptance of the popular Verilog-A analogue subset of the Verilog-AMS [2] hardware description language as a vehicle for specifying and interchanging device and circuit models among commercial and GNU General Public License (GPL) circuit simulators. In the past semiconductor device models were often released as C code source files compatible with the application programming interface of the Berkeley SPICE 3 program [3]. Hand-coding of circuit simulation models in C includes generation of code for the partial derivatives of model currents and charges. This is by no means a trivial task. Indeed, it is often a tedious and error prone task which can be a major source of error leading to convergence problems when a new model is added to a circuit simulator. In contrast to hand-coded C, Verilog-A provides a highly expressive standardized hardware description language which embodies features for the automatic generation of partial derivatives. Once written, Verilog-A model descriptions are normally compiled to C or C++ code [4][5], and linked to a circuit simulator [6], or run as a "turn-key" application [7][8]. Traditional circuit macromodeling [9] adopts a different approach where a subcircuit is used to represent the connectivity of the circuit being modeled. The body of the subcircuit being assembled from predefined components and user defined subcircuits, whose parameters are often represented by numerical quantities rather than non-linear algebraic equations. The SPICE 3 type B controlled sources being an exception. Macromodeling supports both functional modeling and interactive testing without the need for code compilation and linking, making the technique particularly suitable for model developers who do not have a specialized knowledge of simulator application programming interfaces. This paper presents a unified modeling approach for the construction of compact semiconductor device models and circuit macromodels which retains the best features of interactive subcircuit macromodeling while promoting a straight forward procedure for the generation of high performance Verilog-A analogue code. The paper also introduces the concept of equation-defined quantity modeling, which in this context is used to indicate that the central

elements in the main body of a device model, or a circuit macromodel, are physical quantities expressed as non-linear algebraic equations rather than simple numerical values. This approach to modeling is radically different to traditional macromodeling where groups of conventional electrical components are connected to model specific electrical functions in each of the simulation domains. The proposed equation-defined modeling technique has been implemented and tested using the GPL "Quite universal circuit simulator" (Qucs) [10]. The structure and properties of a number of fundamental non-linear equation-defined components are also described in the text and their performance demonstrated using a floating-gate nMOS synapse transistor example.

II. Modelling with Qucs equation defined devices

The Qucs equation-defined device (EDD) [11] is a universal quantity element formed from one to eight two terminal components where individual branch currents, within a single EDD, can be algebraic functions of EDD branch voltages. Similarly, individual branch charges can be algebraic functions of both EDD branch voltages and currents. These current and charge quantities are given by (1) to (3).

$$I = I(V), g = \frac{dI}{dV}, \tag{1}$$

$$Q = Q(I, V), \tag{2}$$

$$C = \frac{dQ}{dV} = \frac{dQ(V)}{dV} + \frac{dQ(I)}{dI} \cdot g. \tag{3}$$

Where g is the branch admittance and C is the branch capacitance. EDD current and charge equations can be composed from constants and variables defined in Qucs equation blocks [12], subcircuit parameters, EDD branch voltages and currents (charge equations only), and the operators and functions defined in the Verilog-A analogue subset of Verilog-AMS. Similar to Verilog-A, Qucs EDD conditional branch current and charge equations can be selected by if-then-else statements written in the style of the C language ternary operator (?:) syntax. The EDD equation-defined quantity elements shown in Fig. 1 illustrate the fundamental Qucs component structures for modeling current and charge as algebraic non-linear functions of EDD branch voltages and currents. In Fig. 1(a) current Iname is a function of the voltages applied to the EDD terminals connected to branches two to eight. EDD branches with current set at zero act as high impedance voltage sensing probes. In Fig. 1(b) charge Q1 represents the stored charge in branch one. Changes in Q1 over a period of time result in a change in the current flowing in branch one. The Verilog-A code fragments listed in Fig. 1 clearly identify the strong relationship between the EDD equation-defined quantity block structures and the corresponding Verilog-A code. The EDD structures in Fig. 1 represent the simplest quantity elements. Other combinations, with different numbers of current or charge equations and voltage probes, within a single EDD are perfectly possible, yielding a range of network functions of which the most significant are noise free resistance, current to voltage conversion, and shot and flicker noise current generation when an EDD is combined with noise voltage source [13]. Combinations of such EDD blocks provide most of the additional functionality needed to model a high percentage of semiconductor devices and circuits. Models which require linear or non-linear inductance can also be easily constructed using a charge equation-defined quantity block and a gyrator [14].

III. Generating Verilog-A code form Qucs equation defined quantity models

Although the current and charge EDD blocks listed in Fig. 1 are shown as separate items the Qucs equation-defined modeling technique naturally proceeds in a top-down fashion rather than the conventional bottom-up process commonly

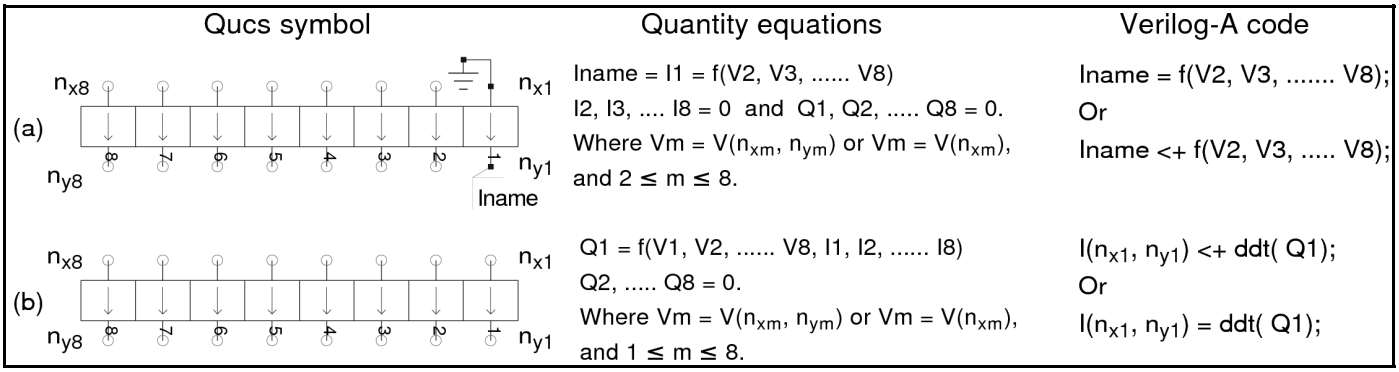

Figure 1. Qucs fundamental quantity modeling blocks: symbol, quantity equations and Verilog-A code; (a) basic current equation block, (b) basic charge equation block

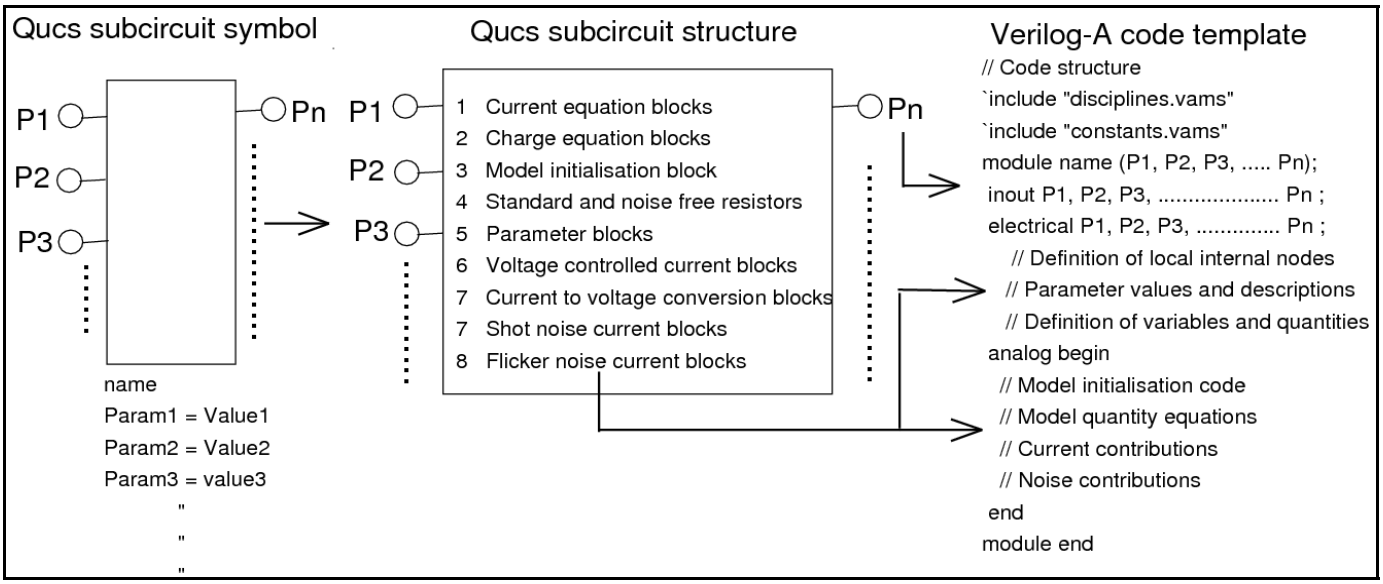

Figure 2. A block diagram illustrating the major stages in the construction of a Qucs equation defined model

adopted in discrete circuit design. Fig. 2 shows a diagram that illustrates the three principle stages in the construction of a Qucs equation-defined quantity model. Starting with a set of equations that characterize the physical properties of a device, or the functionality of a circuit, a subcircuit schematic symbol is drawn with signal connection pins and a default model parameter list. The next step in the model development sequence involves drawing a second schematic composed of EDD current and charge equation-defined blocks which represents the body of a model. Interactive testing of the model follows, allowing a full evaluation of model function. On satisfactory completion of the testing phase the Verilog-A code for a Qucs equation-defined quantity model is generated by inspection of the model symbol and the body schematic, simultaneously entering the Verilog-A code for each item in the various sections indicated as comments (lines starting with //) in the Verilog-A template given in Fig. 2.

IV. A LONG CHANNEL EPFL-EKV NMOS TRANSISTOR MODEL

Fig. 3 lists the fundamental dc and charge equations for a simplified long channel EPFL-EKV nMOS model [15], where VGprime is the effective gate voltage, VP is the pinch-off voltage, n is the slope factor, BETA is a transconductance parameter, Ispecific is the specific current, If is the forward current, Ir is the reverse current and Ids is the drain to source current. EPFL-EKV v2.6 equation numbers are given in "{ }" brackets at the left-hand side of each equation. The parameters for the long channel model are: VTO the threshold voltage, PHI the bulk Fermi potential, GAMMA the body effect parameter, THETA the mobility reduction coefficient, KP a transconductance parameter, W the channel width, L the channel length, COX the gate oxide capacitance per unit area and vt the thermal voltage at the device temperature. The schematics in Figs. 4 and 5 demonstrate how device equations are represented by a set of Qucs EDD structures where each physical quantity is calculated as a branch current or a branch charge. Currents output from an EDD, and passed as an input to other EDD are converted to voltages by setting individual input branch currents to be identical in value to their branch voltage using (4).

$$I_n = V_n . \tag{4}$$

Where subscript n is the EDD branch number in the range 1 to 8. The dynamic charge properties of the nMOS EPFL-EKV transistor model are determined by device parameter Xpart

{22} Vg=V(gate)-V(bulk), Vs=V(source)-V(bulk), Vd=V(drain)-V(bulk)
{33} VGprime=Vg-VTO+PHI+GAMMA · sqrt(PHI)
{34} VP=VGprime-PHI-GAMMA · sqrt(VGprime · (GAMMA/2)^2) - GAMMA/2
{39} n=1+(GAMMA/2) · sqrt(VP+PHI+4 · vt),
{58} BETA=KP · (W/L) · (1/(1+THETA · VP))
{44} X1=(VP-Vs)/vt, If=(ln(1+limexp(X1/2))^2
{57} X2=(VP-Vd)/vt, Ir=(ln(1+limexp(X2/2))^2
{65} Ispecific=2 · n · BETA · vt^2,
{66} Ids=Ispecific · (If-Ir)
{69} nq=1+GAMMA/(2 · sqrt(VP+PHI+1e-6)
{70} Xf=sqrt(0.25+If)
{71} Xr=sqrt(0.25+Ir)
{74} qI=-nq((4/3) · (Xf^2+Xf · Xr+Xr^2)/(Xf+Xr) - 1)
{75} qB=-(GAMMA · sqrt(VP+PHI+1e-6)/vt) -(nq-1)/nq, when VGprime > 0
qB=-VGprime/vt, when VGprime<=0
{76} qG=-qI-qB
{77} Cox=COX · W · L
{78} Q(I,B,G)=Cox · vt · q(I,B,G)

Figure 3. EPFL-EKV v2.6 long channel nMOS device equations

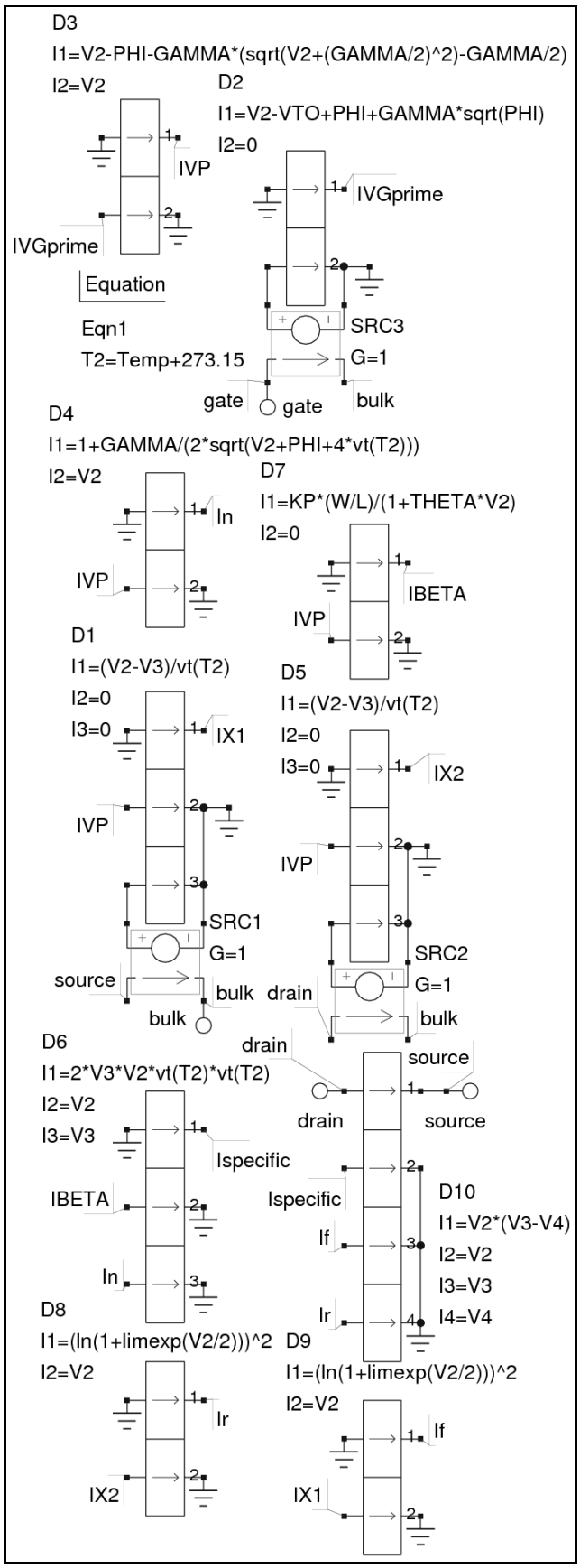

Figure 4. Qucs large signal model for a long channel nMOS transistor

which allows users to set the device charge partition ratio. The default value of Xpart is 0.6 which gives a ratio of 40/60 for QS/QD. Fig. 6 lists the Verilog-A code generated from the EDD equation-defined quantity modeling illustrated in Figs. 4 and 5 and the schematic symbol for the EPFL-EKV nMOS transistor drawn in Fig. 7.

V. An nMOS floating-gate synapse transistor model

The schematic symbol and equivalent circuit shown in Fig. 7 represent a floating-gate nMOS synapse transistor . The model consists of two EPFL-EKV long channel nMOS transistors with one operating in the subthreshold bias region and the second connected as a MOS capacitor, plus a control terminal coupling capacitance and two additional resistors which ensure that the floating gate node has a DC path to ground. Charge on the floating-gate is set by Fowler-Nordheim tunneling [16]. EDD D1 models this process, setting the tunneling current as a function of the voltage applied to the nMOS transistor gate oxide. This example demonstrates the power of the Qucs modeling tools. The currently implemented modeling facilities actively promote experimentation with physical process

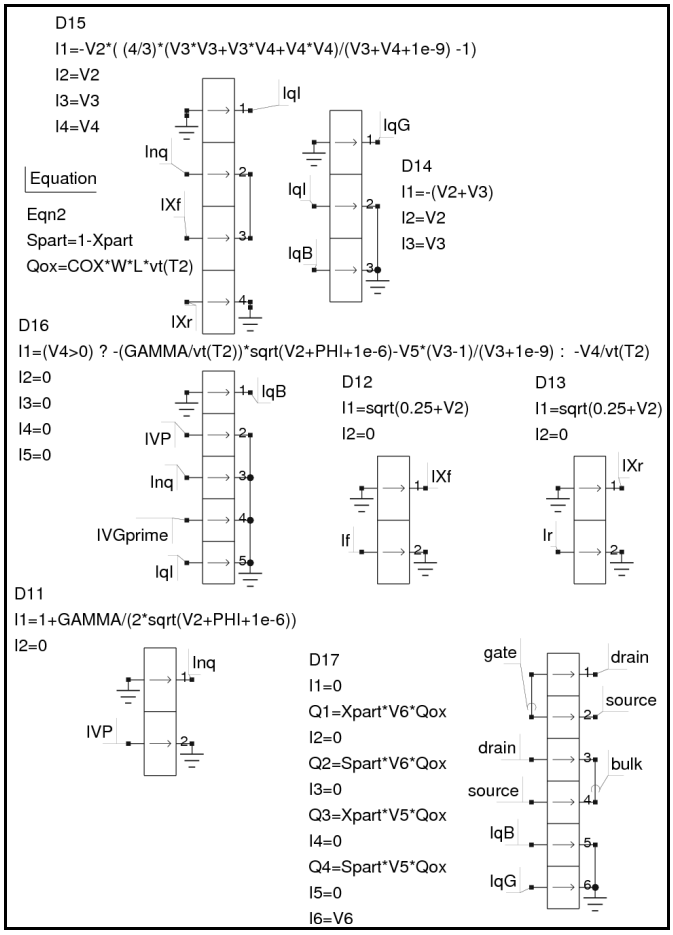

Figure 5. Qucs charge partition model for a long channel nMOS transistor

```
`include "disciplines.vams"
`include "constants.vams"
module EKVLC(drain, gate, source, bulk);
inout drain, gate, source, bulk; electrical drain, gate, source, bulk;
`define attr(txt) (*txt*)
// Parameter value and description blocks
parameter  real L=1e-6 from [1e-20 :inf]
  `attr(info="Channel length");
..
// Definition of internal variables and quantities
real VGprime,VP, n, BETA, X1, If, X2, Ir;
real nq, XF, Xr, ql, qB,qG, Ispecific, Ids;
analog begin
// Model initialization code
@(initial_model) begin
   Spart=1-Xpart; Qox=COX*W*L;
   T2=Temp+273.15; vt=`P_K*T2/`P_Q;
end
// Model quantities
Vg=V(gate)-V(bullk); Vs=V(source)-V(bulk); Vd=V(drain)-V(bulk);
VGprime=Vg-VTO+PHI+GAMMA*sqrt(PHI);
VP=VGprime-PHI-GAMMA*(sqrt(VGprime+(GAMMA/2)^2)-GAMMA/2);
n=1+GAMMA/(2*sqrt(VP+PHI+4*vt); BETA=KP*(W/L)*1/(1+THETA*VP);
X1=(VP-Vd)/vt; If=ln(1+limexp(X1/2))^2;
X2=(VP-Vs)/vt; Ir=ln(1+limexp(X2/2))^2;
Ispecific=2*n*BETA*vt*vt;  Ids=Ispecific*(If-Ir);
nq=1+GAMMA/(2*sqrt(VP+PHI+1e-6);
Xf=sqrt(0.25+If); Xr=sqrt(0.25+Ir)
ql=-nq*( (4/3)*(Xf^2+Xr*Xf+Xr^2)/(Xf+Xr)^2 -1);
if (VGprime > 0)
    qB=-GAMMA*sqrt(VP+PHI+1e-6);
else qB=-VGprime/vt;
qG=-ql-qB;
// Current contributions
I(drain,source) `  <+ Ids;
I(gate, drain)<+ddt(Xpart*qG*Qox;
I(gate, source)<+ddt(Spart*qG*Qox);
I(drain, bulk)<+ddt(Xpart*qB*Qox);
I(source, bulk)<+ddt(Spart*qB*Qox);
end
endmodule
```

Figure 6.Verilog-A code for a long channel EPFL-EKV nMOS transistor

models expressed as non-linear algebraic equation-defined components, subcircuits, macromodels and Verilog-A blocks, making model development a fast and productive activity. These innovative capabilities coupled with the extensive post-simulation data processing features implemented in the latest Qucs release support easy access to the inner workings of models, allowing extraction of circuit signals and equation-defined quantity values during simulation. The waveform plots of the tunneling and source currents and the floating gate voltage shown in Fig. 8 are typical examples of both categories of data.

VI. Discussion and conclusions

The interactive nature of macromodeling is one of the primary reasons why it has become such a popular modeling technique, particularly for constructing behavioural models of compact semiconductor devices and integrated circuits. Conventional macromodeling does have a number of fundamental weaknesses however, including limitations imposed by the lack of non-linear equation-defined components, and the slow simulation speed of macromodels. Hardware description languages address directly the question of model simulation speed, often yielding orders of magnitude improvement after compilation of model code to C or C++ and linking with a circuit simulator. However, this speed improvement does come at a cost, namely a more complex model development process that normally requires a specialised understanding of simulator application programming interfaces. The Qucs equation-defined quantity modeling technique introduced in this paper attempts to combine the best features of macromodeling and compact semiconductor model development with high level hardware description languages. By adding multi-terminal non-linear equation-defined components to macromodeling it becomes possible to interactively construct and test simulation models whose properties translate directly and simply to Verilog-A. Moreover, interactive performance testing at the macromodeling stage of a model design cycle significantly improves the reliability of the resulting Verilog-A code. The Qucs equation-defined quantity approach to compact device and circuit macromodeling also highlights the role new modeling techniques will take on as GPL simulator technology evolves in the future.

Acknowledgment

The material reported in this paper forms part of the work done by the Qucs development team in response to the "MOS Modeling and Parameter Extraction Working Group" (MOS-AK) Verilog-A standardization initiative [17].

References

[1] G. J. Coram, "How to (and How Not to) Write a Compact Model in Verilog-A", IEEE International Behavioural Modeling and Simulation Conference (BMAS), San José, California, pp. 97-106, October 2004.

[2] Accellera, "Verilog-AMS language Reference Manual, version 2.2", 2004, http://www.accellera.org, [accessed November 2008].

[3] B. Johnson, T. Quarles, A. R. Newton, D. O. Pederson D.O, and Sangiovanni-Vincentelli, "SPICE 3f User's Manual", Department of Electrical Engineering and Computer Sciences, University of California, Berkeley, California, 1992.

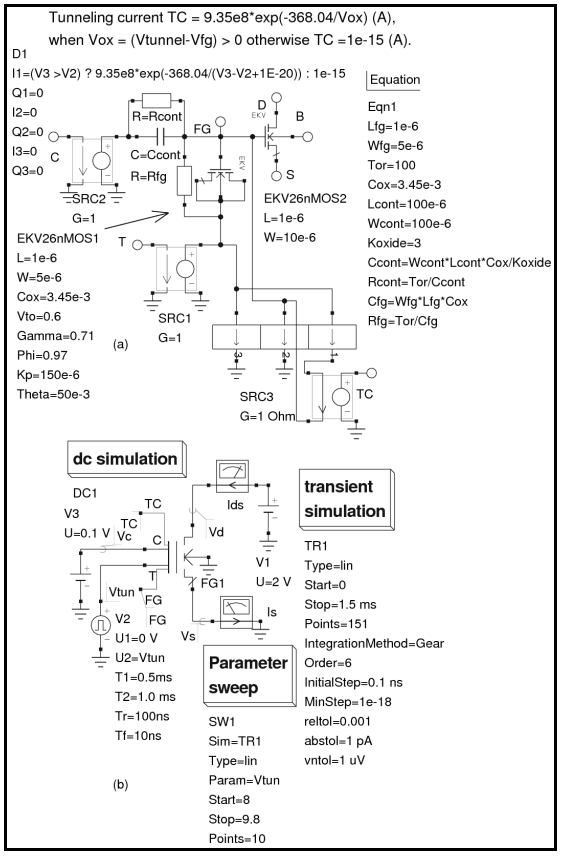

Figure 7. nMOS floating gate synapse transistor : (a) macromodel, (b) symbol and test circuit

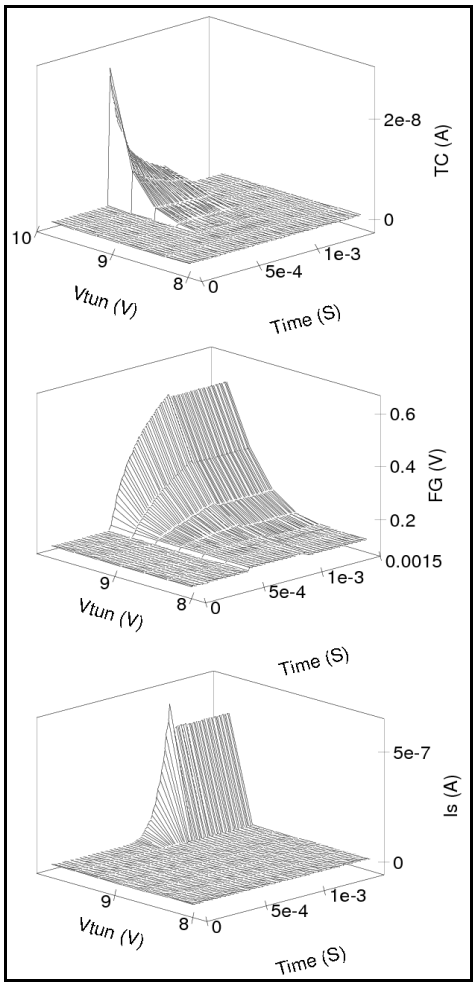

Figure 8. Tunneling current, floating gate voltage and source current plots

[4] L. Lemaitre, W. Grabinski, and C. McAndrew, "Compact Device Modeling using Verilog-A and ADMS", Electron Technology Internet Journal, 2003, vol. 35, pp. 1-5.

[5] B. Gu, and L. Lemaitre, "Recent Advancements on ADMS Development", Nanotech 2008, Boston, Massachusetts, June 2008, vol. 3, pp 787-790.

[6] S. Jahn, and H. Parruitte, "Qucs: A Description; Verilog-AMS interface", 2006, http://qucs.sourceforge.net/docs/Verilog.pdf, [accessed February 2009].

[7] D. Fitzpatrick, and I. Miller, "Analog Behavioral Modeling with the Verilog-A Language", Kluwer Academic Publishers, 1998.

[8] K. S. Kondert, and Q. Zinke, "The designers Guide To Verilog-AMS", Kluwer Academic Publishers, 2004.

[9] J. A. Connelly, and Pyung Choi, "Macromodeling with SPICE", Prentice Hall Inc., 1992.

[10] M. Margraf, S. Jahn, J. Flucke, R. Jacob, V. Habchi, T. Ishikawa, A. Gopala Krishna, M. Brinson, H. Parruitte, B. Roucaries, and G. Kraut, "Qucs (Quite universal circuit simulator)", 2003, http://qucs.sourceforge.net/index.html, [accessed February 2009].

[11] M. Brinson, "Qucs: A tutorial; Component, Compact Device and Circuit Modelling Using Symbolic Equations", 2007, http://qucs.sourceforge.net/docs.html, [accessed Febuary 2009].

[12] M. Brinson, and S. Jahn, "Qucs: A GPL Software Package for Circuit Simulation, Compact Device Modelling and Circuit Macromodelling from DC to RF and beyond", Int. J. Numer. Model, 2009, in press, DOI 10.1002/jnm.702.

[13] M. Brinson, and S. Jahn,"Modelling of high-frequency inductance with Qucs non-linear radio frequency equation defined devices, International Journal of Electronics, in press, 2009.

[14] S. Jahn, and M. Brinson, "Interactive Compact Modeling Using Qucs Equation Defined Devices", Int. J. Numer. Model. 2008, vol. 21, pp. 335-349.

[15] M. Bucher, C. Lallement, C. Enz, F. Théodoloz, and F. Krummenacher. "The EPFL-EKV model equations for simulation, v2.6", Electronics Laboratories, Swiss Federal Institute of Technology, Laussunne (EPFL), 1997.

[16] M. Lenzlinger and E. H. Snow, "Fowler-Nordheim tunneling into thermally grown SiO_2", Journal of Applied Physics, vol. 40, pp. 278-283.

[17] S. Jahn, M. Brinson, H. Parruitte, B. Ardouin, P. Nenzi, and L. Lemaitre, "GNU Simulators supporting Verilog-A Compact Model Standardization", MOS-AK meeting, Premstaetten, 2007. http://www.mos-ak.org/premstaetten/papers/MOSAK_QUCS_ngspice_ADMS.pdf, [accessed February 2009].