2007

Nano scale based model development for MEMS to NEMS migration

Andres Lombo Carrasquilla
University of South Florida

Nano Scale Based Model Development for MEMS to NEMS Migration

by

Andres Lombo Carrasquilla

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
Department of Electrical Engineering
College of Engineering
University of South Florida

Major Professor: Wilfrido Moreno, Ph.D.
James Leffew, Ph.D.
Sanjukta Bhanja, Ph.D.
Kimon Valavanis, Ph.D.
Fernando Falquez, Ph.D.

Date of Approval:
November 7, 2007

Keywords: nanoelectronics, quantum mechanics, integrated modeling, vhdl-ams, scorm

DEDICATION

To my family and everyone who loves knowledge and uses it in peace and harmony.

ACKNOWLEDGMENTS

I wish to thank Dr. Wilfrido Moreno for his continuous guidance during this research. I also thank the Centro de Investigaciones y Desarrollo Cientifico at Universidad Distrital Francisco Jose de Caldas for its partial support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NANO SCALE BASED MODEL DEVELOPMENT FOR MEMS TO NEMS MIGRATION

Andres Lombo Carrasquilla

ABSTRACT

A novel integrated modeling methodology for NEMS is presented. Nano scale device models include typical effects found, at this scale, in various domains. The methodology facilitates the insertion of quantum corrections to nanoscale device models when they are simulated within multi-domain environments, as is performed in the MEMS industry. This methodology includes domain-oriented approximations from ab-initio modeling. In addition, the methodology includes the selection of quantum mechanical compact models that can be integrated with basic electronic circuits or non-electronic lumped element models.

Nanoelectronic device modeling integration in mixed signal systems is reported. The modeling results are compatible with standard hardware description language entities and building blocks. This methodology is based on the IEEE VHDL-AMS, which is an industry standard modeling and simulation hardware description language. The methodology must be object oriented in order to be shared with current and future nanotechnology modeling resources, which are available worldwide.

In order to integrate them inside a Learning Management System (LMS), models

were formulated and adapted for educational purposes. The electronic nanodevice models were translated to a standardized format for learning objects by following the Shareable Content Object Reference Model (SCORM). The SCORM format not only allows models reusability inside the framework of the LMS, but their applicability to various educational levels as well. The model of a molecular transistor was properly defined, integrated and translated using SCORM rules and reused for educational purposes at various levels. A very popular LMS platform was used to support these tasks. The LMS platform compatibility skills were applied to test the applicability and reusability of the generated learning objects.

Model usability was successfully tested and measured within an undergraduate nanotechnology course in an electrical engineering program. The model was reused at the graduate level and adapted afterwards to a nanotechnology education program for school teachers. Following known Learning Management Systems, the developed methodology was successfully formulated and adapted for education.

CHAPTER 1

INTRODUCTION

Nanostructures, nanodevices or devices which are termed mesoscopic, can be built with dimensions smaller than the appropriate mean free path. The prefix "meso" is used to indicate that a device is larger than atomic scale devices but smaller than the macroscopic scale, where Boltzmann transport theory has been demonstrated to be valid. The Nano Electro-Mechanical Systems, or NEMS, integrate nano scale electromechanical sensors and actuators in the same way and as well as Micro Electro-Mechanical Systems, MEMS, integrate micro scale electro-mechanical sensors and actuators. Both systems must be specified, designed, modeled and simulated in an integrated hybrid environment in order to achieve reliable and well designed prototypes. These prototypes can be fabricated with existing fabrication processes. Furthermore, they can be integrated within commercial products to further economic development.

NEMS characterization and simulation involves integrated environments taking into account mechanical, electrical (analog and digital), and optical properties within a single experiment or simulation session. As the dimensions of integrated circuit devices continue to shrink, the finite dimensions of the atoms within the structures will lead to statistical variations in critical dimensions and thus in device properties. As a consequence, physical properties will deviate from the bulk properties used for MEMS.

Such deviations arise from quantum mechanical and mean free path effects. The effects of statistical variations may be even more pronounced in multi-layer structures with components only several atoms thick. Statistical variations require that intrinsic defects and the quantum mechanical effects of confined structures be accounted for in order to characterize a device and predict circuit performance for large-scale integration.

With the National Nanotechnology Initiative (NNI), new concepts and design methodologies are needed to create new nano-scale devices, synthesize nano-systems and provide for their integration into architectures for various operational environments. Design methodologies also require multiple layers of abstraction and various mathematical models to represent component behavior at the different layers. On the other hand, the emergence of new processes in nanostructures, nanodevices and nano-systems create an urgent need for theoretical development, modeling, simulation and new design tools in order to understand, control and accelerate development at the scale of these regimes. Quantum mechanics and quantum chemistry, multi-particle simulation, molecular simulation, grain and continuum-based models, stochastic methods and nano mechanics must be combined in order to accomplish all of the required development.

Nano-scale modeling and simulation processes have always been computationally expensive. High performance computer clusters have been arranged to run atom by atom models and to obtain exact particle response from each material. These kinds of simulations provide a detailed description of carrier relations inside the atom, molecule or interface between atoms or molecules. Depending on the number of particles involved and the nature of the response to be extracted from the simulation, algorithms for these kinds of tasks have very high complexity orders. On the other hand, current MEMS

modeling and simulation tools can be run on relatively cheap workstations without high performance requirements. Lumped element models simplify multi-physics experiments and provide fast and complete system response to the designer. This characteristic is enforced by a growing market with high demands, which is looking for reliable systems compatible with existing CMOS and Bi-CMOS manufacturing lines.

Industry standard hardware description languages such as Verilog and VHDL have demonstrated their applicability in top-down design methodologies for MEMS. However, the nanotechnology industry has not used this approach due to the absence of affordable models that fit with the existing VHDL models of microscale devices. Additional tools are required to complete the whole design methodology. Digital, analog, mechanical and other domains enforce the requirement for the addition of specific tools. Additionally, cross domain verifications must be addressed in order to couple results from each design stage. To date, hardware description languages have not been used to describe nanoscale devices or to integrate such devices in a top-down design methodology. Results from molecular dynamic simulations or quantum mechanics modeling tools have not yet been directly applied to a design flow. The primary reason for the absence of the application of such tools is their complexity and incompatibility with existing Integrated Circuits (IC), and MEMS tools, which apply continuous theory rules. The statistical nanodevice nature complicates the task of expressing it by means of standard hardware description language statements. Therefore, a coupled or approximated solution, which permits systems, subsystems and devices modeling and simulation, has been pursued. Any strategies for modeling and simulation must be accessible to the scientific community by means of an open language platform, which

ensures their readability and reusability. This fact is particularly important when educational processes involve nano scale devices and other modeling and simulation tools are also involved.

Nanodevice models must also be presented as educational objects inside a learning platform. Current learning platforms have been successfully integrated by means of traditional computational environments as well as mobile computing devices. Advances in programming languages, operating systems and computing hardware and applications are leveraging the opporunities for integration of complex computing models into user friendly learning platforms. These platforms commonly support graduate and undergraduate studies in the nanotechnology area. Modeling and simulation platforms specifically dedicated to the nanotechnology area are available on line from various educational institutions around the world.

CHAPTER 2

NANO SCALE MODELING

The coupling between quantum models, continuum models and the quasi-continuum models as a middle point between the first two is currently an important development area related to the modeling and simulation market. Some methods have been proposed to analyze defects in solids [1]. These methods are intended for application to multiscale (nano and micro scale), modeling of crystalline materials under mechanical loads. Other methods have proposed two ways of coupling using a sequential method and a coupled method between the continuum and the nanoscale domain for microfluidic applications [2]. Yu and collaborators have proposed a quantum mechanical correction method, which is used for simulation of logic circuits and silicon MOS devices operation [3]. Yu's method uses a one-dimensional solution of the Schrodinger and Poisson equations, which employs a Density Gradient approach for quantum modeling. Most researchers have been working from the typical Natahnson´s approach of a resonant gate transistor, which was published in 1967. Natahnson´s approach employs a "lumped" mass-spring model where small sets of electric circuit elements represent the behavior of devices [4]. This approach has been tested with many microelectromechanical devices. However, it has been shown to be limited when nanodevices co-exist with microdevices. These limitations are mainly encountered when no-steady state solutions must be

analyzed, dynamical information must be collected or a specific nano-scale region must be solved. S.Ai and J. Pelesko provide an analysis of the viscosity dominated and time harmonically forced mass-spring model [5]. They demonstrate the applicability of this approach when inertial forces are both negligible and non-negligible.

A more detailed work with respect to modeling at the nanoscale has been performed for electronic devices. Zhiping Yu et al. have discussed the inclusion of quantum mechanical corrections to the classical transport model for devices and circuits [3]. Initially, the Hansch model, the Van Dort model and a hybrid model were compared. The first model considers the repulsive boundary condition for channel carriers at a Si/SiO2 interface and introduces a shape function, which is imposed upon the carrier concentration in the transverse direction. The second model utilizes the fact that energy quantization increases the bandgap at the substrate surface under the gate. Both methods have drawbacks that could be solved by a hybrid model, which combines the Hansch and Van Dort models.

Probably the most important effort in the modeling and simulation of nanodevices for the non-equilibrium condition has been reported by Professor Mark Lundstrom's group at Purdue University. J. Rew, from the Purdue group, developed a study to understand essential physics of quasi-ballistic transport and its implications to nanoscale device simulation based on macroscopic transport models [9]. R. Venugopal worked the non-equilibrium Green's function (NEGF) formalism into model quantum transport in nanoscale silicon transistors [9]. The objectives of these works have been:

- Implement the appropriate physics and methodology for nanoscale device modeling,

- Develop new TCAD (technology computer aided design) tools for quantum scale device simulation,
- Examine and assess new features of carrier transport in future developments in nanoscale transistors.

P. Damle presents an approach to model quantum transport in nanoscale electronic devices [9]. Damle's approach is based on the non-equilibrium Green's function (NEGF) formalism method. Damle treats a few nanoscale devices of current interest. The devices treated by Damle include a dual-gate silicon nanotransistor (effective mass model) a three terminal molecular device (semi-empirical atomic orbital model) and two terminal molecular wires (rigorous ab initio atomic orbital model). Results from these investigations provide useful insights into the underlying physics in these devices. Several important features such as charge transfer, self-consistent band lineup, I-V characteristics and voltage drop were analyzed and explained.

The NEGF approach is very consistent with a low temperature operation point. However, other techniques are appropriate when higher temperatures and external driving fields are involved in the modeling and simulation process. The density matrix approach and the Wigner distribution method are available when those special conditions are encountered.

Adequate construction of nanodevice models is necessary to develop the theory of transport in nanostructures. The first approach to this phenomenon is the ballistic (coherent or unscattered) transport. To characterize ballistic transport, all the dimensions must be comparable to or less than the inelastic electron mean free path. For GaAs at 300

°K this factor is approximately 120 nm. This consideration allows for the extraction of the energy relaxation length and the gate length, which are very important dimensions. The energy relaxation length is given by:

$$le = v_F \tau_{e,} \tag{1}$$

where v_F is the Fermi velocity and τ_e is the energy relaxation time. When the gate length is comparable to the inelastic mean free path, phase interference effects appear in the transport phenomena, which makes the Boltzmann equations non-applicable.

When phase interference effects appear, carriers move with no scattering along the coherence length. This length can be defined as the distance over which the electrons lose their phase memory. Loss of phase memory yields an average broadening of the energy levels, ΔEa, which can be related to the number of states contributing to the current (I). The current is given by:

$$I = eV/(dn/dE) \tag{2}$$

through the equation:

$$\frac{\Delta E_a}{dE/dn} = \frac{\hbar}{e^2} \sigma L^{d-2}, \tag{3}$$

where L is the diffusion length. Each state contributes with a current proportional to one electron per second.

Variables to be observed and measured from nano scale systems are primarily the phase interference, conductance fluctuations, resistance, carrier heating and scattering time [7]. Other variables related to the thermal and magnetic behavior of the device are of secondary importance.

2.1. **Mesoscopic Observables in Nanostructures**

In order to develop an adequate characterization of nanostructures certain key parameters must be considered. Most of these parameters can be related to their corresponding values in the continuum domain where Boltzmann's equations can normally be used. However, in the temperature regime, when conditions are restricted to low temperatures, these parameters cannot be related to normal processing and fabrication steps. Under conditions of low temperature they must be related to normal operation conditions. Most of the parameters look familiar but they must be related to the quantum regime. The parameters to be considered are:

- Density: In particular, the sheet density of carriers in the quasi 2D electron gas at any interface.
- Mobility: Specifically at the interfaces.
- Scattering time: Derived from the corresponding mobility.
- Fermi wave vector: Determined by the density through:

$$k_F = (2\pi n_s)^{1/2}. \tag{4}$$

- Fermi velocity: Derived from the expression:

$$v_F = \hbar k_F / m. \tag{5}$$

- Elastic mean free path: Derived from the expression:

$$l_e = v_F \cdot \tau_{sc}, \tag{6}$$

where τ_{sc} is the scattering time. This expression is only valid at low temperatures.

- Inelastic mean free path: Derived from the expression:

$$l_{in} = v_F \cdot \tau_\varphi , \tag{7}$$

where τ_φ is the inelastic mean free time or phase breaking time.

- Diffusion constant: Derived from the expression:

$$D = v_F^2 \tau_{sc} / d . \tag{8}$$

- Phase coherence length or Thouless length: Derived from the expression:

$$l_\varphi = \left(D \tau_\varphi\right)^{1/2}, \tag{9}$$

this parameter can be used as the diffusion length. On the other hand, the inelastic length is the distance traveled by an electron ballistically in τ_φ time.

- Thermal length: Derived from the expression:

$$l_T = \left(\frac{\hbar D}{k_B T} \right)^{1/2} . \tag{10}$$

- Magnetic length: Derived from the expression:

$$l_m = \left(\frac{\hbar}{eB} \right)^{1/2} . \tag{11}$$

All of these calculations will be affected by the potential barriers between each quantum box, quantum dots, nano-wires, or any other device involved. It is also valid when connections between each part of the system are performed by means of an electro-magnetical effect, which are very usual when micro-scaled systems are scaled down to the nano-regime. The actual potential shape and values depend on many-body effects within each box or system. These systems can be classified as non-equilibrium systems.

They are usually formulated using the non-equilibrium Green's function in real time. At times these systems may require classification as far-from equilibrium. Formalism has not been formulated for far-from equilibrium systems. An equilibrium approach to this kind of devices can be also acceptable if working conditions matches operationals requirements for system operation. In any case this matching must be considered for conserving quantum behavior effects inside and outside of each potential barrier.

2.1.1. **Ballistic Transport**

Some basic definitions must be stated to characterize transport in nanostructures. All of the definitions are related to the phenomenon called ballistic transport. The term ballistic is used to characterize the transport condition under which the traveling distance of the carriers is comparable or lesser than the mean distance between scattering events. In consequence, many carriers can travel from the injection point to the point of extraction without any scattering, which is a behavior similar to projectiles or electrons in a vacuum tube [8],[46]. Their associated charge flow is known as ballistic transport. Ballistic transport can be formulated from the Landauer formula for conductance. At any part of the material, which can be viewed as a two-port network, the carrier concentration variation is related to conductance variation by:

$$G = \frac{I}{\partial V} = \frac{e^2}{\hbar}\frac{T}{R}\frac{dn}{dE}\frac{dE}{dK_x}. \tag{12}$$

2.1.2. **Phase Interference**

The phase of an electron in the presence of a vector potential can be calculated

using the Peierl's substitution:

$$K \to \tfrac{(p+eA)}{\hbar}, \tag{13}$$

then

$$\phi = \phi_0 + \frac{(p+eA)}{\hbar} \cdot r . \tag{14}$$

Therefore, the phase difference is given by:

$$\partial\phi = 2\pi \frac{\phi}{\phi_0} = \frac{e}{\hbar} \int_{ring} B \cdot nds , \tag{15}$$

where

$$\phi_0 = h/e \tag{16}$$

is the quantum unit of flux, and ϕ is the magnetic flux coupled through the ring. According to the Aharonov – Bohm effect, this result can be used to calculate the resistance or conductance as the periodic oscillation of the flux.

2.1.3. **Universal Conductance**

Time independent oscillations are periodic in ϕ, equation (16), in systems whose size scale is of the order of the phase breaking length. They occur with variations in the Fermi energy. Therefore, these systems are sample-dependent. For a FET device, the gate voltage or magnetic field characterizes the random interference of the trajectories inside a sample, if the sample size is comparable with the phase coherence length. Thus the solution of Landauer's formula when the applied gate voltage is more negative yields step variations of conductance of the order of $(25.812{,}8 \text{ ohms})^{-1}$, [32]. A similar situation is experienced for the quantum Hall effect.

2.1.4. **Weak Localization**

This phenomenon can be measured when the electron tends to return to its original position, interfering with itself, and possessing a velocity, which is negatively correlated to its original velocity.

2.1.5. **Carrier Heating**

The quantum kinetics, not considered in Boltzman equations, can be calculated using a reduction of the Liouville-von Neumann equation for the density matrix ρ with boundary conditions, as:

$$i\hbar \frac{\partial \rho}{\partial t} = [H_0, \rho] + [V, \rho] + [F, \rho]. \tag{17}$$

In equation (17), the first term on the right side corresponds to the carriers, phonons and impurities Hamiltonian, the second term corresponds to the electron-scatterer interaction and the last term corresponds to the electron coupling Hamiltonian.

These definitions provide for a proper formulation of the transport processes through mesoscopic devices and yield important conclusions about the behavior of nanodevices, specifically those involved in NEMS.

2.2. **Mathematical Description of Transport in Nanodevices**

In this section, a description of the mathematical models for nanodevices is presented. In addition, the main software tools available are explored and conclusions

are obtained in order to find an adequate way to model and simulate NEMS.

In order to develop an adequate mathematical formulation for a generic nanodevice, the Hamiltonian operator facilitates a matrix representation of the effective mass equation. This approach must be used because of the variations of each quantity at a particular spatial coordinate. Solutions for the Schrodinger equation when a periodic potential is considered and the effective mass is included, can be written as:

$$i\hbar\frac{\partial}{\partial t}\Psi(r,t) = -\frac{\hbar^2}{2m^*}\nabla^2\Psi(r,t) + E_C(r,t)\Psi(r,t) + U_S(r,t)\Psi(r,t), \quad (18)$$

where E_C is the band edge energy and U_S is the scattering potential. H` is the Hamiltonian operator defined as:

$$H` = H + \mathrm{U}_S(r,t) \quad (19)$$

and

$$H = -\frac{\hbar^2}{2m^*}\nabla^2 + E_C(r). \quad (20)$$

Then the effective mass equation can be written as:

$$i\hbar\frac{\partial}{\partial t}\Psi(r,t) = H`\Psi(r,t), \quad (21)$$

or can be also written with matrix notation using a set of orthonormal functions as:

$$i\hbar\frac{d}{dt}\{\psi(t)\} = [H(t)]\{\psi(t)\}, \quad (22)$$

where $\{\psi(t)\}$ is a column vector not explicitly dependent on r and a matrix is used instead of a differential operator. This concept is widely used in multi-particle systems where a wavefunction is almost impossible to formulate. The state vector belonging to an N-dimensional state space can be written as:

$$\Psi(r,t) = \sum_{i=1}^{N} \psi_i(t) u_i(r), \tag{23}$$

where the set of unit vectors u_i are orthonormal and $\psi_i^* \psi_i$ is the probability of an electron being in state i. The electron density can be formulated as:

$$\mathrm{n(r,t)} = \sum_{all\_e} \psi^*(r,t)\psi(r,t). \tag{24}$$

The summation is performed over all the particles (electrons). For instance, the current density can be calculated as:

$$J(r,t) = -\frac{iq\hbar}{2m^*} \sum (\nabla\psi)^*\psi - \psi^*(\nabla\psi). \tag{25}$$

Equation (25) proposes a connection with the continuity equation. However, at this time non-equilibrium conditions enforce independent calculation of the electron density at each port of any nanodevice. Non-equilibrium conditions can arise from many situations such as through shining light, or maintaining a temperature, or potential gradient at the nanodevice.

Transport in nanostructures needs to be expressed by means of a non-equilibrium system whose energy is affected by the potential barriers between each quantum box (e.g. quantum dots or any other nanodevice), which is known as the many-body effect. Finite element techniques or perturbation techniques are useful for single particle problems. However, in most practical computations, different techniques must be combined [9]. The non-equilibrium Green's function (NEGF), method is one of the most popular solution methods for the problem. The Density Matrix method and the Hartree-Fock method are more sophisticated methods for solution of electron-electron interactions. These methods are usually applied to quantum chemistry calculations. The Density

Matrix method needs to work with the atomic orbital coordinates more than the real space coordinates. The Density Matrix approach can be used to compute the electron density or the current itself.

The Wigner distribution method has been studied and applied to the solution of electron-electron interactions by various research groups around the world. With an adequate computing facility, Monte Carlo methods provide an affordable method to solve the many-body Schrodinger equation. A spectral-domain method demonstrating good accuracy and faster response than other second-order finite-difference methods is described in [10]. Its effectiveness must be tested with 2D and 3D problems. Those methods are reviewed in the next paragraphs.

2.2.1. **Non-Equilibrium Green's Function Method**

In order to explain this method a typical Metal-Oxide-Semiconductor Field-Effect transistor (MOSFET), is modeled. It can be viewed, as presented in Figure 1, as three regions forming two contact regions with a potential difference of μ_1- μ_2 [9]. The particles (electrons or holes) coming from the two contacts, from left to right or viceversa, have rates of $\frac{\gamma_1}{\hbar}$ and $\frac{\gamma_2}{\hbar}$, respectively. As it can be seen on Figure 1, the net current at left side is given by:

$$I_L = q\frac{\gamma_1}{\hbar}(N_1 - N), \tag{26}$$

and the net current at the right side is given by:

$$I_R = q\frac{\gamma_2}{\hbar}(N - N_2). \tag{27}$$

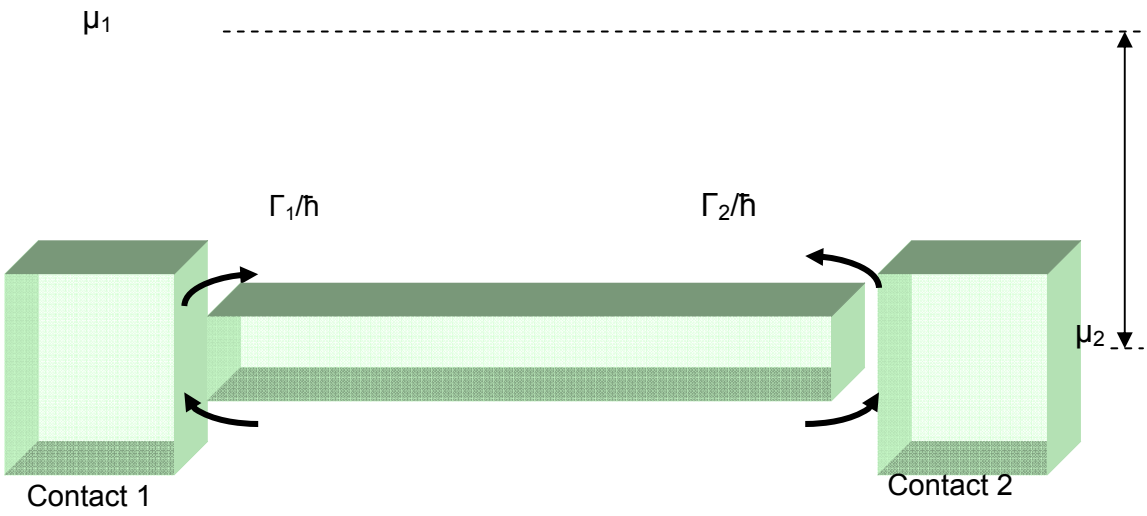

Figure 1: Sketch of Transistor Regions and Relations for MOSFET Modeling and Simulation; Adapted from [9]

Assuming no other current sources,

$$I=I_L=I_R, \tag{28}$$

so

$$I = \frac{2q}{\hbar}\frac{\gamma_1\gamma_2}{\gamma_1+\gamma_2}\left[f_1(E)-f_2(E)\right], \tag{29}$$

where

$$N_1 = 2f_1(E) = 2(1+\exp\frac{E-\mu_1}{k_BT})^{-1} \tag{30}$$

and

$$N_2 = 2f_2(E) = 2(1+\exp\frac{E-\mu_2}{k_BT})^{-1} \tag{31}$$

are the number of electrons. The quantities f_1 and f_2 are the Fermi functions at the contacts. The instantaneous number of particles responsible of conduction, N, varies from N_1 to N_2 for non-equilibrium conditions. After some algebraic manipulations [42],

N is found to be given by:

$$N = 2\frac{\gamma_1 f_1(E) + \gamma_2 f_2(E)}{\gamma_1 + \gamma_2} \tag{32}$$

and the total current is given by:

$$I = \frac{2q}{\hbar}\frac{\gamma_1\gamma_2}{\gamma_1 + \gamma_2}\left[f_1(E) - f_2(E)\right]. \tag{33}$$

At this point it is necessary to include the average broadening of energy states, which affects the total current across the device. Using the definitions given in, [9], the self-consistent energy function at each side of the device is found to by given by:

$$U_{SC} = U[N\text{-}N_0] = U[N\text{-}2f_0(E_f)], \tag{34}$$

which yields the total energy inside as:

$$\varepsilon = \varepsilon_0 + U_{SC}, \tag{35}$$

where ε_0 is the closest molecular level to E_f. For instance, a single broadened energy level can be described as:

$$D(E) = \frac{1}{2\pi}\frac{\gamma}{(E-\varepsilon)^2 + (\gamma/2)^2}. \tag{36}$$

Consequently, the new expressions for the amount of electrons and the total current are given by:

$$N = 2\int_{-\infty}^{\infty} dED(E)\frac{\gamma_1 f_1(E) + \gamma_2 f_2(E)}{\gamma_1 + \gamma_2} \tag{37}$$

and

$$I = \frac{2q}{\hbar}\int_{-\infty}^{\infty} dED(E)\frac{\gamma_1\gamma_2}{\gamma_1 + \gamma_2}\left[f_1(E) - f_2(E)\right]. \tag{38}$$

The Green function is formulated as:

$$G(E) = \left(E - \varepsilon + i\frac{\gamma_1 + \gamma_2}{2} \right)^{-1}, \tag{39}$$

using the above definitions for a single broadened energy level. However, it must be generalized for a multiple level condition which commonly happens inside a molecular device according to the all previous considerations. A secondary spectral function can be defined as:

$$\mathrm{A(E)} = \mathrm{-2Im\{G(E)\}} \tag{40}$$

and the broadening as

$$D(E) = \frac{A(E)}{\pi}, \tag{41}$$

which results in new definitions for electron density and current as:

$$N = \frac{2}{2\pi} \int_{-\infty}^{\infty} dE \left(\left|G(E)\right|^2 \gamma_1 f_1(E) + \left|G(E)\right|^2 \gamma_2 f_2(E) \right) \tag{42}$$

and

$$I = \frac{2q}{\hbar} \int_{-\infty}^{\infty} dE \gamma_1 \gamma_2 \left|G(E)\right|^2 \left[f_1(E) - f_2(E) \right]. \tag{43}$$

It is more exact to use a matrix formulation for the Green function:

$$G(E) = \left(ES - H - \Sigma_1 - \Sigma_2 \right), \tag{44}$$

where terms have been replaced by their corresponding matrices, which contain the variables for each particle-particle interaction. S is the identity matrix. The single energy level ε_0 is replaced by the Hamiltonian matrix H. In order to be consistent, the broadening terms γ are replaced by the complex energy-dependent self-energy matrices named Σ [9].

Clearly, this approach needs a large computation facility since the complexity of the calculations increases dramatically when many interacting particles are considered. The solution to the problem has to be followed through an iterative process to converge when the solutions of the NEGF are applied to the matrix density. These solutions are approximately the same as those obtained from Poisson's equation. The process can be outlined as follows:

- Depending on the device to be modeled, (e.g. a molecular one or a bigger one such as a MOSFET), select an appropriate matrix representation. Either an atomic orbital basis or a real space discrete lattice basis, [43].
- Write down a suitable Hamiltonian matrix for the device.
- Calculate the contact self-energy functions, according to the device.
- Select a value of the self-consistent potential to begin the iterations.
- Solve the NEGF to obtain the density matrix.
- Calculate the new self-consistent potential from the density matrix and iterate from the previous step until a proper convergence can be confirmed.
- Use the final density matrix to calculate the electron density, the current and other necessary functions.

Depending on whether the device is molecular or any other mesoscopic device, the method can vary and other variables can be included.

2.2.2. **Density Matrix Method**

The Density Matrix Method is based primarily in the Hartree-Fock variational principle. In the variational principle many system wave-functions are a product of

antisymmetrized single particle wave-functions. The Hartree-Fock equation states that the single particle potential is given by:

$$v_S^{HF} = v_{ext} + \frac{1}{2}\int d^3r' \frac{n(r)}{(r-r')}, \tag{45}$$

where v_{ext} represents the states for the external potential applied to the particle. However, equation (45) does not account for any correlation between the particles. If particle interaction is to be taken in account, the Density Functional Theory is applied [14]. This theory includes all exchange and correlation effects. There exists, almost always, one external potential that, when doubly occupied by two non-interacting electrons, yields the exact density of a H_2 molecule. The exact form of the exchange-correlation functional is unknown. The simplest approximation is the Local Density Approximation (LDA). A local functional provides information about the function at a single point contributing to the final solution. However, this approximation needs to be optimized by gradient terms through a process such as the Lagrange method or the Generalized Gradient method.

Application of the Density Matrix Method to an N electron system requires a set of 3N spatial coordinates, (r_i), and N spin coordinates, (σ_i), where:

$$x_i = (r_i, \sigma_i)\,. \tag{46}$$

The electron probability is written as:

$$\int dx_1 \ldots \int dx_N \left|\psi(x_1, \ldots, x_N)\right|^2 = 1\,, \tag{47}$$

where each integral is applied over all space. The electronic density is obtained as a summation, including both spins and is given by:

$$n(r) = N\sum_{\sigma}\int dx_2 \ldots \int dx_N \left|\psi(r, \sigma, x_2 \ldots x_N)\right|^2\,. \tag{48}$$

This procedure is very computationally expensive, since many methods have to be tested in order to solve this system. Even though the original formulation divides the electron density into subsystems, the computational expense is large.

Approximations to the density functional lead to linear scaling methods, mainly for metals, but can be generalized to any material. Each orbital contains $2f_i$ electrons where $0 \le f_i \le 1$, and the electronic density can be simplified to:

$$n(r) = 2\sum_i f_i \int |\psi_i(r)|^2 \tag{49}$$

and the non-interacting kinetic energy functional is given by:

$$T_S^J[n] = \min_{\{f_i\},\{|\psi_i|\}\to n} 2\sum_i f_i \psi_i^*(r)(-\frac{1}{2}\nabla^2)\psi_i(r)\,. \tag{50}$$

If all the interactions are included, then the total functional is given by:

$$2\sum_i f_i \int dr \psi_i^*(r)(-\frac{1}{2}\nabla^2)\psi_i(r) + E_H[n] + E^J{}_{XC}[n] + \int dr \cdot n(r)V_{ext}(r)\,; \tag{51}$$

however, equation (51) needs to be minimized with respect to the occupation numbers, f_i, and the orbitals, $\{|\psi_i|\}$. Using a common Lagrange method, a set of Schrodinger-like equations is obtained as:

$$\left(-\frac{1}{2} f_i \nabla^2 + f_i V(r)\right)\psi_i(r) = \lambda_i \psi_i(r)\,, \tag{52}$$

where:

$$\lambda_i = f_i \varepsilon_i \tag{53}$$

and equation (52) simplifies to:

$$\left(-\frac{1}{2}\nabla^2 + V(r)\right)\psi_i(r) = \varepsilon_i \psi_i(r)\,. \tag{54}$$

Equation (54) is multiplied by $\psi^*_i(r)$ and integrated, which yields the expression:

$$\int dr \cdot \psi^*_i(r)(-\frac{1}{2}\nabla^2)\psi_i(r) + \int dr \cdot |\psi_i(r)|^2 V(r) = \varepsilon_i, \quad (55)$$

which is related to the energy levels that any orbital can occupy.

This procedure is currently used inside the ABINIT tool to find the total energy, charge density and electronic structure of systems made of electrons and nuclei (molecules and periodic solids), using pseudopotentials and a planewave basis. ABINIT also includes options to optimize the geometry, perform a molecular dynamics simulation or generate dynamical matrices for Born effective charges and dielectric tensors. Control of the computational costs of this method is implemented through the use of computational clusters, which is the method currently utilized by USF.

2.2.3. **Wigner Transport Equation**

The Wigner Transport Equation method is based in the use of a phase space defined function, f(k,r,t), for electrons, which is given by:

$$f(\vec{k},\vec{r},t) = \sum_j P_j \int_{-\infty}^{\infty} \phi_j(\vec{r}+\frac{\vec{u}}{2},t)\cdot\phi^*_j(\vec{r}-\frac{\vec{u}}{2},t)\exp(-i\vec{k}\vec{u})du\,. \quad (56)$$

This function satisfies the transport equation given by:

$$\frac{\partial f}{\partial t} + \frac{h\vec{k}}{m^*}\cdot\nabla f - \nabla U \cdot \frac{1}{\hbar}\frac{\partial f}{\partial k} - C_Q(\vec{k},\vec{r},t) = \left(\frac{\partial f}{\partial t}\right)_C. \quad (57)$$

The parameter P_j in equation (56) is the occupancy probability of any electron at state j. The third term on the left side of equation (57) corresponds to the drift term and

the fourth term corresponds to the quantum correction. The term on the right side of the equality in equation (57) is the collision term.

Assuming no phonon occurrence [12] the relaxation time must be less than the carrier transition time and the electron drift energy much lesser than the thermal energy. These assumptions allow equation (57) to be simplified. The simplified version of equation (57) is given by:

$$\frac{\partial f}{\partial t}+\frac{\hbar\vec{k}}{m^*}\cdot\nabla f-\frac{1}{\hbar}\nabla\left(U-\frac{\hbar^2\nabla^2\ln(n)}{12m^*}\right)\cdot\frac{\partial f}{\partial k}=\left(\frac{\partial f}{\partial t}\right)_C. \tag{58}$$

A slightly different approach to evaluation of the Wigner function is presented in [13]. The approach in [13] uses a truncated form of the Wigner potential, which is given by:

$$V(\vec{r},\vec{k})=\int\frac{ds}{2\pi i\hbar}\exp(-ik\cdot s)(V(r+\frac{s}{2})-V(r-\frac{s}{2})+qs\cdot E). \tag{59}$$

This form of the Wigner potential assumes positive and negative values. Therefore, it cannot be considered as a probability density. This fact makes numerical methods such as the Monte Carlo method, which might be applied to solve the Wigner unstable potential since the particle weight (positive or negative) grows exponentially and the variance would also grows exponentially. However, as in [13], this method has been tested with acceptable results for devices of 1D at room and at low temperatures where scattering is present. However, the coherence length must be sufficiently large compared to feature sizes, otherwise the resonant peaks cannot be resolved properly. On this way, a conclusion of most of the methods presented can be yield. Drawbacks reside mainly in the computational complexity that a solution for a nanodevice can present. The expense

of computational complexity is huge when large-scale integration circuits need to be solved. A point in favor of the NEGF method is that it can be applied to the contacts between devices, asumming that nanoscale analysis can be performed there and that nanoscale effects can be partially ignored for the rest of the device. It is also based in a superposition principle, where partial solutions for each part contribute linearly to the solution.

A simplified formulation method needs to be found for systems at nano or micro scales. At the scales of NEMS and MEMS, numerous contacts are involved inside the analysis. In addition, what the microelectronics industry is currently applying as modeling tools does not perform these types of considerations. Therefore, an evident requirement is for the development of a solution system that will provide for integration between the nano and the micro scale´s approach to modeling and simulation.

At this point, the nanometric system abstraction level arises as a key point to consider for any computational system. A deeper analysis of modeling and simulation approaches is performed in next paragraphs. The aim is to facilitate a valid abstraction of quantum mechanics principles applied to systems, which include interfaces with micro scale and nano scale devices.

2.3. **Nanodevice Modeling and Simulation**

There are various research initiatives and groups whose primary concern is nanodevices modeling and simulation [33-39]. Most of research is oriented toward finding an adequate computational algorithm, which represents the quantum behavior of structures [45]. Currently available software tools run on relatively onerous and time-

consuming computational platforms. PROPHET, MOSES, SIMON and SETTTRANS represent some of the tools, which have been created to involve quantum calculations within the micro and nano-electronic design process. NEMO 3-D, from the Jet Propulsion Laboratory of NASA and CalTech, arises as one of the best and complete efforts to simulate semiconductor nanodevices. This group is working on a tool that can be applied to the simulation of optical properties of quantum dots. Initial studies have been oriented to metallic quantum dots in various implementations. These implementations include lateral quantum dots, through electrical gating of heterostructures, vertical quantum dots, through wet etching of quantum well structures, pyramidal quantum dots, through self-assembled growth, and trench quantum wires. The NEMO 3-D tool excludes band structure effects from the electron scattering simulation mainly because of the computational expense associated with the inclusion of these effects. Benchmarking has been performed using single and multiple CPUs with shared memory. Machines such as the Sun E450 Ultra-Sparc 2 running at 300 MHz, the SGI Origin 2000 running at 200 MHz, the HP V class PA 8000 at 200 MHz, clusters of the HP/Convex SPP-2000 – 256 CPUs at 180 MHz, the SGI Onyx – 4 CPUs at 200 MHz, and the Intel Pentium II – 16 CPUs at 200 MHz running the LINUX operating system, were incorporated in the benchmarking.

In Germany, the Walter Schottky Institute and personnel from the University of Rome developed Nanoext3, which is a software suite for 3D nano-device simulation that solves an 8-band momentum and the Schrodinger-Poisson equation using a library of III-V materials. Nanoext3 can calculate the structure of 3D heterostructure quantum devices under bias and its current density close to equilibrium. Nanoext3 uses a mixture of

methods. These methods range from the totally quantum mechanical solvers, to find the electronic structure, to a semi-classical approach of local Fermi levels, to find the current. In order to perform these calculations Nanoext3 assumes that the carriers are in local equilibrium.

Nanohub is a web-based initiative spearheaded by the NSF-funded Network for Computational Nanotechnology (NNC), which includes seven universities. Nanohub provides on-line simulation services and utilizes tools such as 2DS, Schred 2.0, NanoMOS and TBGreen.

- 2DS is a tool for solving Schrodinger's equation in a 2D quantum well with infinite potential barriers and an arbitrary, user defined, potential field inside the well. The tool discretizes Schrodinger's equation through the use of a Finite Cloud Method (FCM). 2DS solves the eigenvalue problem by using ARPACK.
- Schred 2.0 calculates the envelope wavefunctions and corresponding bound-state energies in a typical MOS structure. These calculations involve solutions of the one-dimensional Poisson and Schrodinger equations. Schred 2.0 assumes certain conditions for the quantum simulation. The Si/SiO2 interface is assumed parallel to the [100] plane. The conduction band is represented by the six equivalent valleys. Then the effective masses are calculated from the valley curvature. The valence band is represented by the heavy-hole band and the light-hole band and uses the same masses. Schred 2.0 is written in Fortran 77. It takes about 10 seconds per bias point calculation in quantum mode on a SPARC-5 workstation. However, only about 2 to 3 minutes are required for bulk calculations where subband energies crowds together.

- NanoMOS is a 2D simulator for thin body, (less than 5 nm), fully depleted, double-gated n-MOSFETs. NanoMOS implements five different transport models. Two of the models, quantum ballistic, [46] and quantum diffusive, are related to quantum transport.
- TBGreen calculates the transmission and reflection coefficients at any port using a tight-binding Green's function method. No benchmarks or comparisons have been reported using this tight-binding method.

On the other hand, NANOTCAD is a project funded by the European Commission [58]. The project resides within the Nanotechnology Information Devices initiative of the Information Society Technologies (IST) Program. NANOTCAD's main objectives are the development and validation of a software package for the simulation and the design of a wide spectrum of devices. The devices are based both on semiconductors and on transport through single molecules. In addition, NANOTCAD, [54] proposes demonstration of a procedure for the realization of prototype nano-scale devices based on detailed modeling.

Other approaches to NEMS modeling and simulation have been oriented to a proper representation and interchange of nanodevice designs. NanoTITAN Inc., has been a leader in this area when publishing nanoML, which is a data markup language for systematic organization, representation and interchange of nanodevice designs. Nanodevice designs include the molecular components, structure and information about the properties, interoperability, operational characteristics, display and legal status of nanodevices. A recent release of this tool, called nanoXplorer IDE, provides for molecular dynamics simulations.

However, the complexity in simulation environments becomes a problem when the system involves mixed signals designs. Problems also arise with a mixing of nano and micro scale devices where a multiphysics analysis is required to be performed. The complexity is also represented by the computational resources needed by nanoXplorer IDE when it performs nanodevices visualization graphics since graphic acceleration hardware and/or software are required.

According to the software tools reviewed, it can be concluded that all the tools are primarily oriented to find an adequate computational algorithm, which represents the quantum behavior of structures. However, most of the new tools lack portability and compatibility with existing processing tools. The new tools must be integrated with existing tools in order to perform a fabrication process. In addition, these new tools must be able to accept an existing and reliable model of the whole system at micro and nano scales, [55] , which solve customer requirements in a flexible way. An operating system, expensive computational processing capabilities, fast processor, and/or processor clusters and large memory capacities are intensively required by all the tools explored, [56]. This fact is very important when designers and fabrication facilities must be in concordance for a particular production process. The industry wants the fastest and most reliable tool to integrate with its production lines. The designer wants the more accurate tool to be sure that the design is in compliance with what the customer desires. However, accuracy, speed and reliability must meet at one point where the final product quality must be satisfied. In the next chapters an object oriented solution to this matter is formulated and successfully applied.

CHAPTER 3

OBJECT ORIENTED MODELING

This section defines the object oriented approach, which was applied to the modeling and simulation of nanodevices. Emphasis is placed on the results obtained from a typical nanoelectronic device. The results have been exported as learning objects to be used by engineering students who are users of a Learning Management System. Conclusions with respect to the applicability and limitations of this kind of solution are presented. The benefits for a learning process in nanotechnology at the undergraduate and graduate level are also presented.

Nanodevices can be considered primarily as analog devices inside a VHDL-AMS framework. However, these devices are closely related to digital systems if the designer is concerned with logical gates. In this case the mixed signal approach proves to be superior. This chapter presents a framework for nanosystems definition from the modeling and simulation point of view. This formulation is needed in order to facilitate the description of component relationships, the definition of simulation domains, and the corresponding influences that a variable from one domain can have over a variable in another domain.

From the systems point of view, a nanosystem can be considered as a well organized set of nanodevices and interfaces. The devices and interfaces are sufficiently

variable in number and type to allow a designer to form an interpretation of the whole system within one or all of the domains, [40], which encompass electronic, mechanical, optical, fluidic, thermal or electromagnetic characteristics. The system must be evaluated from the lowest level, which is the device level, to the highest one, which is the functional level. The framework for a VHDL-AMS formulation is the design entity, which consists of an (entity declaration)-(architecture body declaration) pair, which is more commonly referred to simply as an entity-architecture pair or design entity. The design entity represents the instance of an object when applied to the construction of complex nanosystems. This formalism is presented in Figure 2.

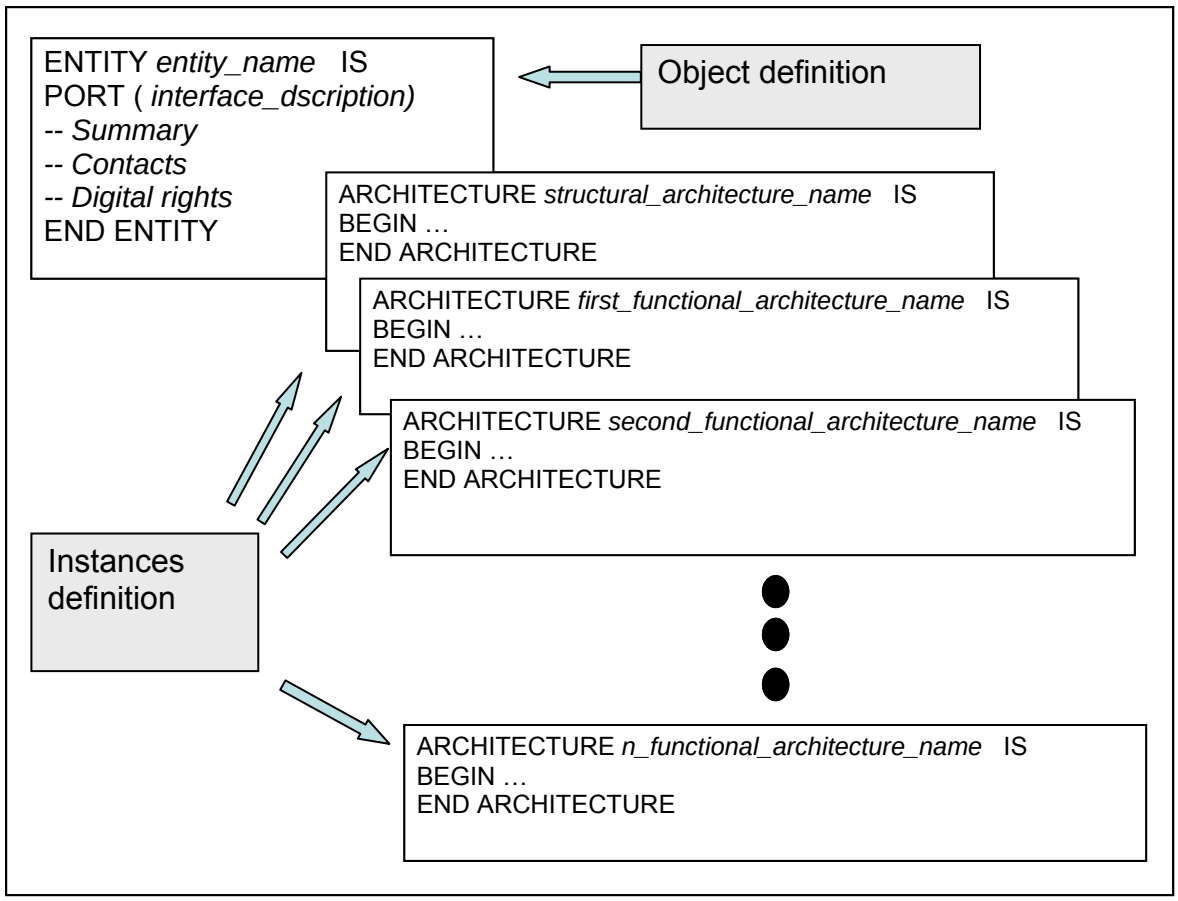

Figure 2: Entity-Architecture Pairs Used as Object Instances for a Nano-System

In order to adopt an object oriented modeling strategy, systems, subsystems and their connections must be addressed by an object. They must have properties to be inherited by other objects when they take part of a nanodevices object library. Each object, (nanodevice), must be characterized by:

- Entity Declaration, (Entity): Consists of a description containing the nanodevice's name, the set of interfaces that allow nanodevice connections, a device summary, contacts and any digital rights associated with elements of the design.
- Architecture Body Declaration, (Architecture): The architecture can represent structure, which enables precise control over the selected nanodevice's elements. The architecture can also be functional, which allows the nanodevice's physical, electromagnetic, chemical and optical characteristics and properties to be represented as a set of formulas and algorithms. In accordance with the objectives of this research, the preferred architectures must be the functional and the primary domain evaluated is the electrical domain. It is not complicated to make use of a similar set of functions in other domains. In addition, recognized microelectronics industry standard modeling tools have been stated. A brief equivalence among various domains is clearly formulated in Figure 3, which was extracted from an industries' modeling tool. It each domain, an equivalent effort of flow variable complies with the same relationship inside a particular object. With this fact in mind a proper definition of a nanodevice entity can be formulated. Furthermore, the

entity can be translated to other domains while maintaining the same set of relations and interpreting them inside a different conceptual framework.

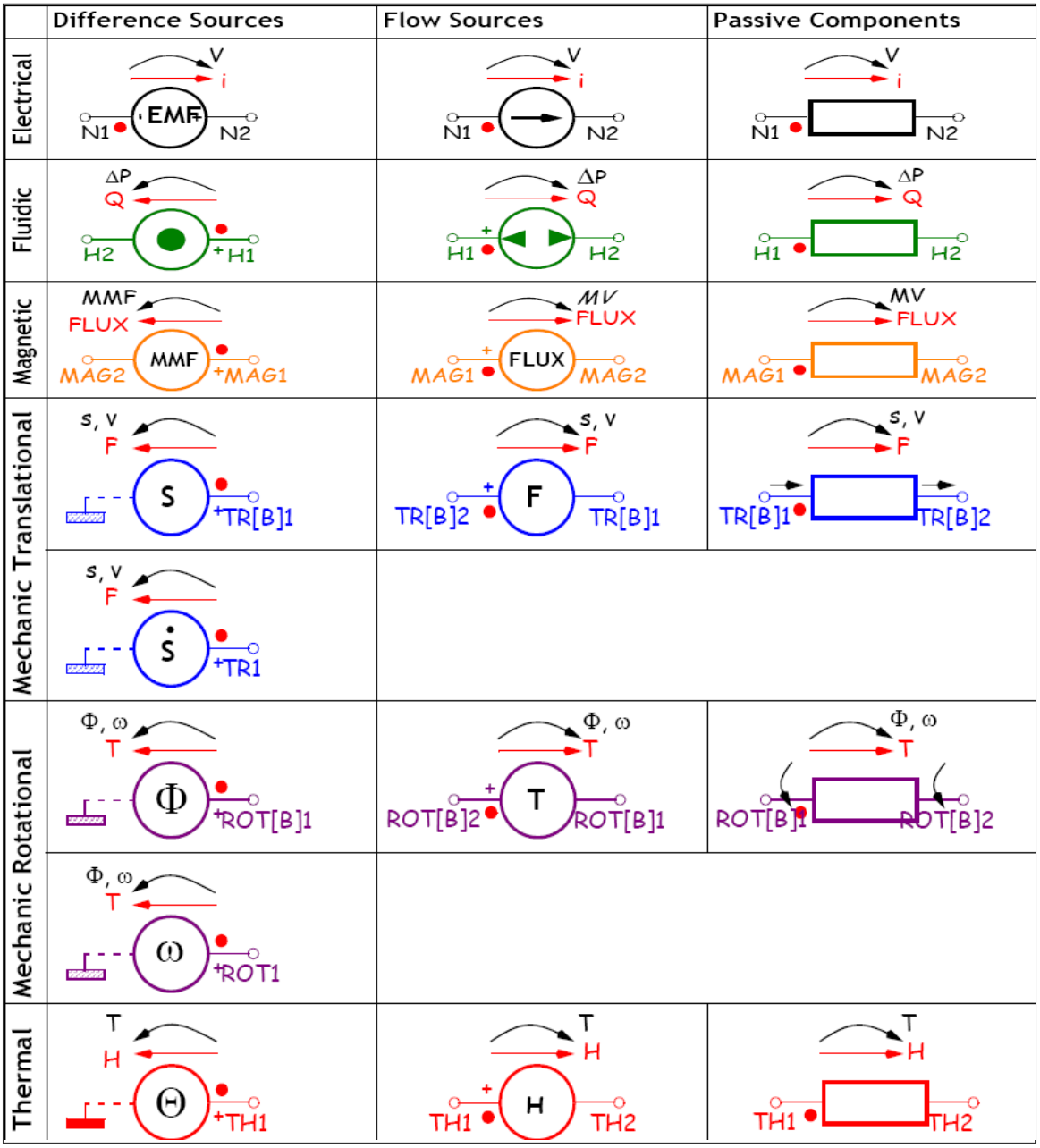

Figure 3: Component Variable Equivalences among Various Domains; Adapted from Ansoft Corporation's Simplorer 7.0 VHDL-AMS Tutorial, 2004

This structure must be applied to common electrical nanodevice model formulations for a VHDL-AMS platform [50-53]. Once available VHDL-AMS software packages have been evaluated, proper formulation of the nanodevice is executed.

3.1. **The Nanosystem**

In order to formulate a nanosystem the designer must collect a set of requirements, which must be organized within an abstract system formalism. This formalism allows the designer to propose a collection of components that fit with the original requirements from the structural and functional point of view. The designer must formulate or, in the structural case, organize and adapt a set of primitive components that complies as exactly as possible with the user demand.

There must be a formal model, which is applied to specify (write) the requirements of each nanosystem as well as each nanodevice. In fact, each nanodevice is a nanosystem by itself. The smallest nanodevice that can be analyzed is the hydrogen atom. However, it is better to create a systematic framework in order to accelerate the time to market in the nano-related industry. A nanosystem can be decomposed in a number of subsystems according to specific design goals [40].

3.2. **Connections**

A connection must be established between a pair of systems or subsystems. A connection can be modeled in two ways. The two types of connections comprise the rigorous metallic connection and the electromagnetic coupling, which permits particle transport between two reservoirs. These connections can be considered as point-to-point

as well as multipoint depending on their geometry or the kind of molecules used in performing the connection. It is important that a designer define what kinds of relationships are described inside a particular NEMS design. This requirement arises due to the possibility of multiphysics analysis and correlated measurements, which can be performed over the same connection between two or more nanodevices.

3.3. **Subsystems**

Molecular systems are the most populated group of devices with current applications within the industry. Consequently these systems can be easily modeled and simulated. In addition, the solid state electronic nanodevices are important for the industry. In the solid state electronic group there are one, two or many terminal devices. Examples of this group include quantum dots, resonant tunneling devices (diodes and transistors) and single electron transistors.

A hierarchical view of the most common nanodevices is presented in Figure 4. It shows that a system is composed of subsystems and their connections. At the nano scale those connections can be considered in two ways. One way to form a physical contact is between materials of different molecular structure, which are usually known as metal contacts. Another connection is formed when there is very close proximity between the regions with different or similar molecular structure. The latter situation performs the connections by electromagnetic coupling. On the other hand, subsystems, which can be considered as a complete system, can be classified depending on their structure. A molecular device is composed by a unique pattern of molecules that can be behaviorally isolated at the model. The Schrodinger equation can be solved independently for each

molecule, except at the boundaries [41]. Depending on the specific behavior exhibited, it is possible to classify the device differently in a particular domain. A molecular device might be classified as electrochemical, photoactive or electro-mechanical as a function of the principal behavior exhibited in a particular domain. Solid state nanodevices are also classified based on its particular behavior inside a system. It is also possible to include more additional classifications depending on a specific domain to be modeled and simulated such as microfluidics, thermodynamics, and mechanics [41].

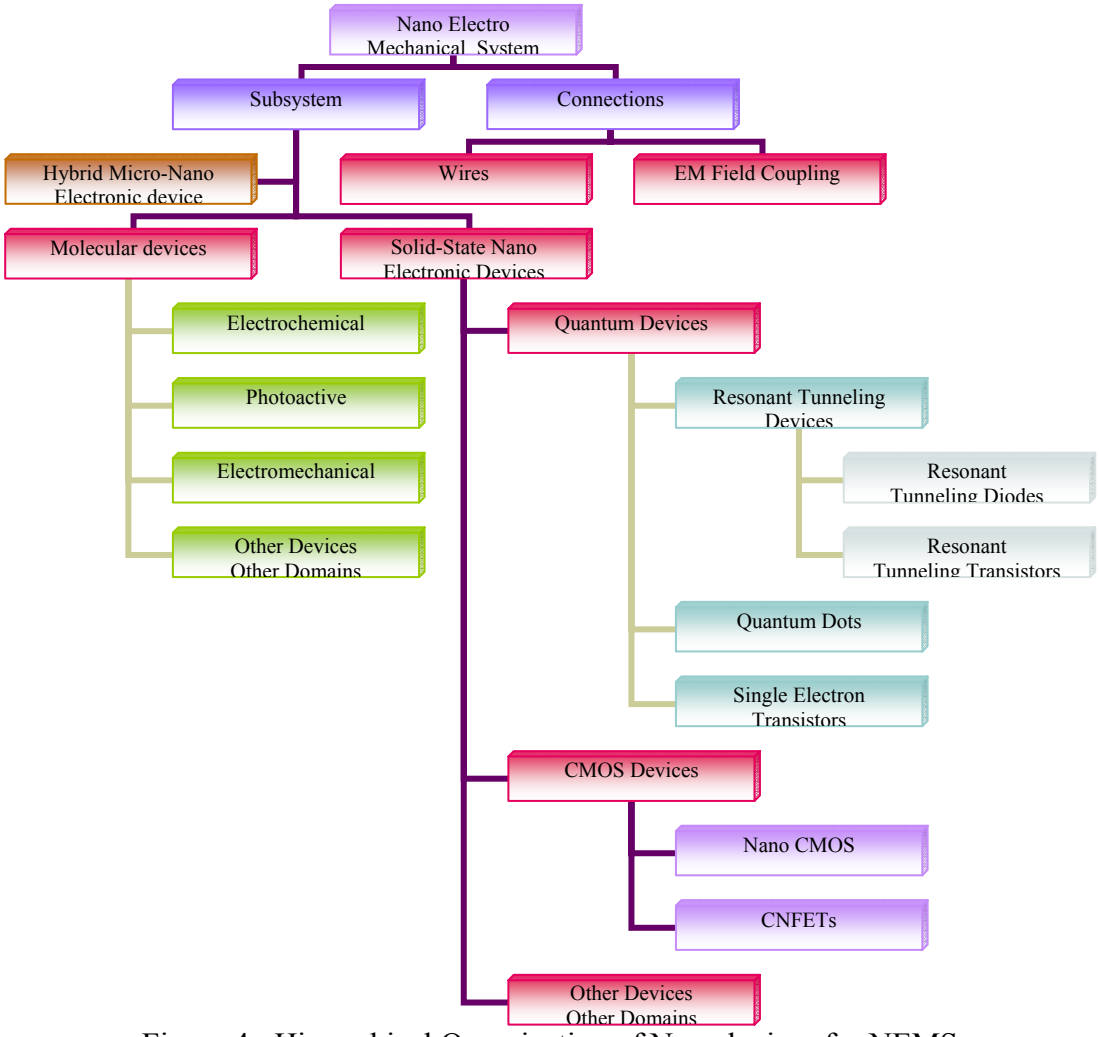

Figure 4: Hierarchical Organization of Nanodevices for NEMS

In order to adopt an object oriented modeling strategy, systems, subsystems and their connections must be addressed. Object properties must be inherited from/to other objects when they are part of a nanodevices object library. Each object must be characterized by [28]:

- A description containing the nanodevice's name, summary, contacts and any digital rights associated with elements of the design.
- A structural architecture, which enables precise control over the selected nanodevice's elements.
- A functional architecture, which contains a set of the nanodevice's physical, electromagnetic, chemical and optical properties for easy reference.

This structure must be applied to common electrical nanodevices model formulation for a VHDL-AMS platform. All nanodevices have been well characterized, in many reports, from the continuum theory point of view. However, probabilistic behavior must be included in common tools such as the available VHDL-AMS software packages.

From the hierarchical view some devices can be modeled as a function of their computational complexity. It can be evaluated from the dimensional point of view. However, many other aspects of the model must be considered. Models can be considered as:

- Zero-dimensional: such as quantum dots,
- One-dimensional: such as quantum wires,
- Two-dimensional: such as quantum wells (inside a molecular transistor, a single-electron transistor or tunneling devices such as diodes or transistors),

- Three-dimensional: such as quantum bulks (nano-cantilevers and nano-tools) [41].

The hierarchical representation of nanoscale devices presented in Figure 4 shows that a system can collect as many subsystems and/or devices as exist in the abstraction levels where they are related. Fully elaborated VHDL-AMS designs can use XML as an intermediate way of representation [27]. XML can extract complete static semantical information, which is inherent to VHDL-AMS and dynamic simulation related information such as the current values of signal drivers or the dynamic equation sets.

3.4. **SCORM – Shareable Content Object Reference Model**

Further development of the XML structure yields standardized knowledge objects, which are written following worldwide standards such as SCORM, (Shareable Content Object Reference Model). SCORM provides for the description and deliverance of e-learning content in different software platforms. The importance of SCORM lies in the ability to represent educational contents which can be shared and in the interface between these contents and the e-learning platforms that use them. Multiple platforms, either commercial or open-source, support the SCORM specification called ADL 2004.

The main SCORM components are:

- The CAM (Content Aggregation Model), which defines a model for packaging learning content. CAM deals with Assets, Shareable Content Objects (SCO), and Content Aggregation Packages. Assets are single individual objects such as HTML pages. SCOs are collections of assets.

They should be independent of the learning context and are intended to be subjectively small units such that potential reuse across multiple learning objectives is feasible. Content Aggregation Packages comprise one or more SCOs or assets. Therefore SCOs comprise one or more learning objects. SCOs should be structured in such a way that they are ready for delivery to a student.

- The RTE (Run Time Environment), defines an interface for enabling communications between learning content and the system that launches it such as a LMS. The RTE deals with an API (Application Programming Interface) adapter and a RTE service routine. The API adapter enables communications between learning content and the LMS from which it is launched. The RTE service routine is provided by the LMS and is responsible for providing the user nterface for the student [57].

CHAPTER 4

VHDL-AMS CAPABILITIES TO MODEL AND SIMULATE NANODEVICES

According to the IEEE standard 1076.1-1999, VHDL-AMS is a superset of the IEEE 1076-1993 standard language with capabilities for modeling and simulation of analog and mixed-signals designs. This can be accomplished by including a capability for representing and analyzing non-linear ordinary differential and algebraic equations. The models can follow the energy conservative principle, using nodes as a TERMINAL, or non conservative principles, using nodes termed a QUANTITY. In the case of QUANTITY, inputs are only mathematically modified and presented at outputs. Additionally, the unknowns can denote any waveform or a time series of values.

In order to achieve an adequate modeling and simulation of nanodevices or any multi-particle device at the nano scale, those quantities have to be written by means of a set of quantum correlated matrices. All iterations to be performed are followed by a "break" statement, which informs the analog solver to schedule an appropriate solution point and to determine a new initial solution for the next continuous functional segment or piece. No analog solver has been fixed an IEEE standard. Therefore, each implementer can choose the appropriate method for the solutions of equations. However, it is not yet clear which method is the best when modeling and simulating nanodevices.

Another factor to be considered is that software platforms, which include VHDL-

AMS, vary in the way of implementing the language standard. Some platforms exclude certain capabilities formulated by the IEEE or limit the capabilities of certain primitives to certain types of variables. This fact can severely decrease the capabilities of each nanodevice implementation inside a particular software tool.

Multiple experiences have been reported about formulation of VHDL-AMS models for MEMS, [24] –26], [40]. None has been reported, which include nano scale devices involving the quantum corrections mentioned previously. Describing a partial differential equation (PDE) using VHDL-AMS requires a proper PDE definition. The PDE definition must include all its parameters, its boundary conditions and a contact interface with the rest of the system [25]. VHDL-AMS does not directly support PDEs. However, the equation can be discretized with respect to spatial variables, which leaves the time derivatives to the language itself.

A more complicated situation arises when multiple domains are involved in a systems simulation [26]. Reduced order modeling of linear systems can be achieved, including non linear systems. However, the interface of analog components may use non-conservative nodes (QUANTITY), which can be connected to conservative nodes (TERMINAL). However, this type of connection is not allowed by the language. Therefore, it is necessary to modify the system, subsystem or component model interfaces in each design. Implementing this idea, in practice, leads to multiple architecture modeling (MAM) through the use of TERMINAL nodes instead of QUANTITY nodes when low or high abstraction design levels are modeled and simulated. This idea must be similarly applied when modeling nanoscale devices and their connections with microscale devices.

The final consideration that needs to be attended is that VHDL-AMS is not an object oriented language. However, the instantiation of VHDL-AMS entities is completely valid as a modeling formalism.

This type of formalism uses a hierarchical fashion to create new devices and relate them to their parent devices while maintaining original properties and adding others.

4.1. **Constructing Models**

A set of preliminary models inside the electrical domain, was developed using existing simulation tools. The models behave in accordance with the quantum mechanical theory and display the expected response from a circuit theory point of view. More models, where other domains are involved, such as the electromechanical domain, must be proposed and tested. The nano-cantilever would seem to be a likely candidate for investigation. In the following sections a description of models developed is presented and explained. Detailed coding of the models is presented in Appendix A.

4.2. **Molecular Transistor Model**

The Molecular Transistor model is based on quantum mechanical approximations of a molecular transistor. It was pointed out in Chapter 2 that conductance fluctuations are periodic in h/e. The gate voltage determines those fluctuations. The model was formulated using the theory presented and explained in Chapter 2. An exact model was simulated using the University of Purdue nanohub facilities using the approach proposed in [9]. The nanohub simulation used an exact solution with variables in a matrix

notation, which made the solution of high electron populated models very expensive but more exact than others.

A simplified model was developed wherein scalar quantities are used. The model was initially presented in Matlab and later translated to VHDL-AMS code. Steady state simplifications were performed according to reported experiences in [31]. Parametric simulations were designed and structured outputs were obtained. Four kinds of response can be displayed, depending on the simplification level of the model:

- One energy level (the lowest ~ the most probable) response, without broadening effects and spin effects.
- One energy level with broadening but without spin effects.
- One energy level with spin effects, which is also called the "unrestricted" model.
- Two energy levels: taking into account the two main energy levels E0 and E1.

A schematic view of the model is presented in Figure 5.

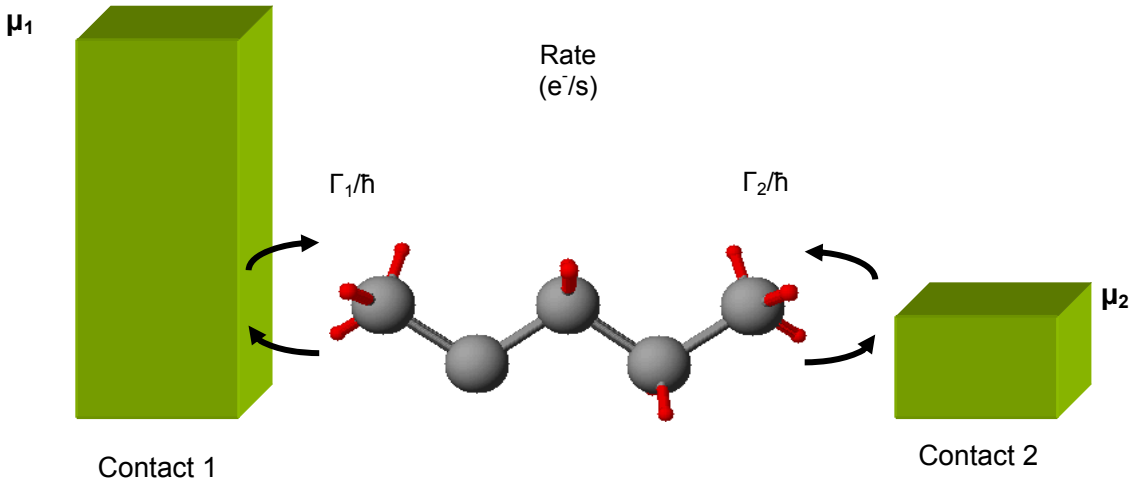

Figure 5: Schematic View of a Molecular Transistor Model

The transistor was modeled as two bulk regions with potential energy μ_1 and μ_2; this potential can be viewed as the electrochemical potential related with the Fermi function. Consider a typical transistor with grounded source and an applied drain voltage V_D. For such a transistor the relationship:

$$\mu_1 - \mu_2 = qV_D \tag{60}$$

is valid. $\Gamma_1/\hbar$ and $\Gamma_2/\hbar$ are the rate at which any charged particle (electrons) can escape from or to the bulk regions and depend on coupling with the gate molecule. Broadened density of states at contacts is modeled using a Lorentzian function centered at ε. The Lorentzian function is given by:

$$D(E) = \frac{1}{2\pi} \frac{\gamma}{(E-\varepsilon)^2 + (\gamma/2)^2}. \tag{61}$$

A generic molecule forms the channel region. If the contact bulk region is metallic the states distribution is continuous. However, if the material is semiconductor, effects such as negative differential resistance and other related effects can be present. The parameters chosen are summarized in Table 1. External parameters are related to the surrounding circuit to be connected. Internal parameters depend on transistor gate molecular composition. Mixed parameters involve the two previous conditions. Primarily, the broadening effect depends on the modification of molecular energy levels when it makes contact with the source and drain bulk.

As a primer approach, Matlab code describing the equations used in this model is presented. A complete coding is presented in Appendix A. The code was organized as follows. An initial parameter definition was written. It states molecular and device operating conditions, defines the molecular charging energy, the molecular potential

energy and coupling conditions with the microscale regions inside the transistor, as well as the condition for equilibrium through the device.

Table 1: Parameters for Simulation of the Molecular Transistor Model

External parameters:	Range
Fermi Energy. (depending on contact materials)	{-7 eV ,.. -3 eV}
Temperature	{50 K,.. 1000 K}
Internal parameters	
Molecular conduction energy levels, and charging energy (depending on molecular composition)	{-8 eV,.. -2eV} {0,..4 eV/electron}
Mixed parameters (internal-external)	
Broadening factors Γ1 and Γ2	{0.025, ..1}

Then, an energy grid was defined in order to initiate simplified Hamiltonian calculations where the parameter NE indicates the density of the grid. NE determines how exact will be the solution obtained. NE also determines the complexity of the calculations and the length of the computation time. The next step was to define biasing conditions. At this point it was important to insure that the real voltage range for transistor operation corresponded with the range expected for the actual transistor used. Most of the reported experiments deal with a short voltage range (-0.8v to 0.8 v). Therefore, this range was included in the code. The parameter IV was increased in value in order to achieve a more exact solution. This parameter also directly determines the computational time. The rest of the code describes the computation of Fermi functions and the corresponding current values. Table 2 shows the complete Matlab code.

Table 2: Matlab Code of a Molecular Transistor.

```
% Parameters definition
U0 = 0.25;                                  % charging energy in eV
kT = 0.025;                                 %energy in eV %at room temp T=300K
mu = 0;
ep = 0.2;                                   % in eV
N0 = 0;
Alphag = 0;                                 % molecular coupling
alphad = 0.5;                               % molecular coupling
%Energy grid definitions
NE = 501;
E = linspace(-1,1,NE);
dE = E(2) - E(1);
g2 = 0.005*ones(1,NE);                      %gamma 2
g1 = g2;                                    %gamma 1
g = g1 + g2;                                % absolute broadening factor
%Bias definitions
IV = 101;
VV = linspace(-0.8,0.8,IV);                 % applied voltage
for iV = 1:IV
  Vg = 0;                                   % gate voltage
  Vd = VV(iV);
  Vg = VV(iV);
  mu1 = mu;
  mu2 = mu1 - Vd;
  UL = -(alphag*Vg) - (alphad*Vd);
  U = 0;                                    %Self-consistent field
  dU = 1;
  while dU > 1e-6
    f1 = 1./(1 + exp((E - mu1)./kT));
    f2 = 1./(1 + exp((E - mu2)./kT));
    D = (g./(2*pi))./(((E - ep - UL - U).^2) + ((g./2).^2));
    D = D./(dE*sum(D));
    N(iV) = dE*2*sum(D.*((f1.*g1./g) + (f2.*g2./g)));
    Unew = U0*(N(iV) - N0);
    dU = abs(U - Unew);
    U = U + 0.1*(Unew - U);
  end
  I(iV) = dE*2*I0*(sum(D.*(f1 - f2).*g1.*g2./g));
end
hold on
h = plot(VV,I);
grid on
```

In this model, the larger the vector distance NE, the more exact will be the solution. However, the resulting computational time also increases. Results from this initial model are presented in Figure 6. This code was translated into VHDL-AMS entity architecture pairs with appropriate selection of parameters.

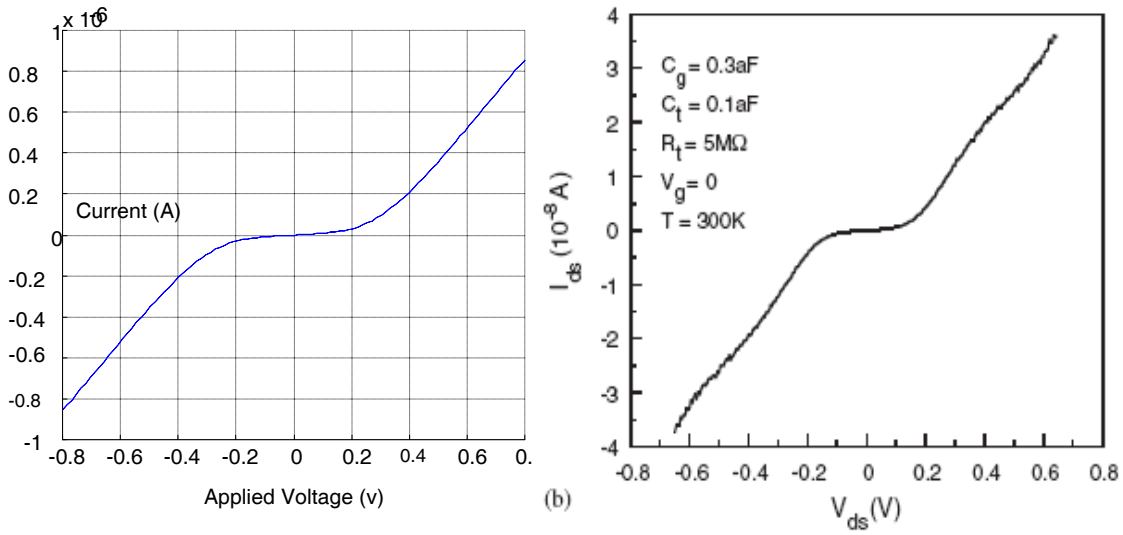

Figure 6: Molecular Transistor I-V Characteristic from the Matlab Code (Left) Compared with Results Obtained in the Arizona State Experiments (Right) [31]

These results show a very close agreement with those reported by Goodnick and Gerousis, from Arizona State University, [31]. However, they use a different simulation package called SIMON 2.0, which is a single-electron circuit simulator based on the Monte Carlo method. SIMON 2.0 includes quantum corrections across multiple junctions. However, it is a very high time and hardware consuming tool. While the simplified model simulation time is approximately two seconds, running on a PC platform, SIMON requires approximately 100 seconds running on a cluster with parallel computation.

In order to complete the results validation process, a different tool for modeling the same device, was tested. The tool Molctoy runs at the University of Purdue's nanohub facility. Figure 7 presents a detailed I-V curve obtained from the Molctoy tool.

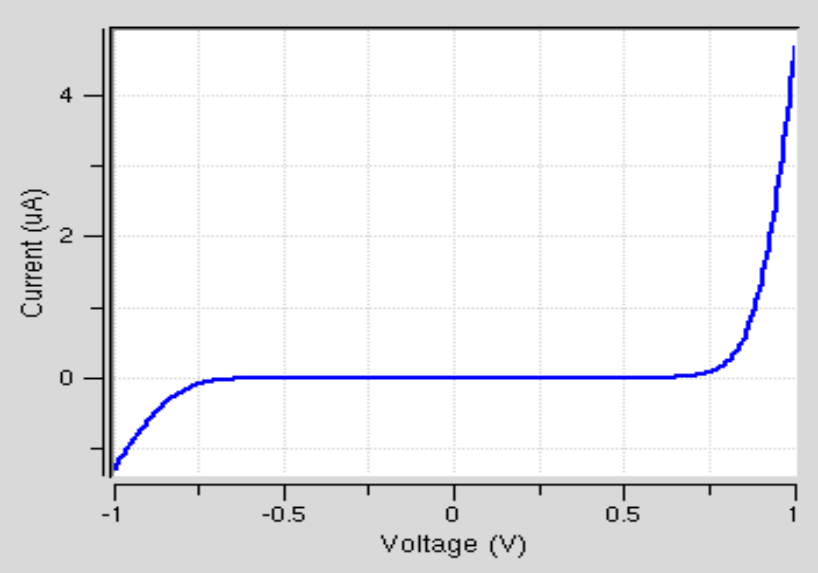

Figure 7: I-V Curve Obtained from the Molctoy Tool

Molctoy formulates a simplified toy model molecular transistor. The Matlab response agrees more with results in, [31], because of the simplifications already made inside Molctoy. In fact, Molctoy also performs more calculations. As in Arizona State's simulations, Molctoy spent approximately 100 seconds to perform the required calculations. Figure 8 presents current, conductance and number of electrons variations when an external voltage is applied to this model. The discrete quantum response for conductance is affecting the electrical current variations. Energy broadening effects can be determined from smooth shapes in all the three plots.

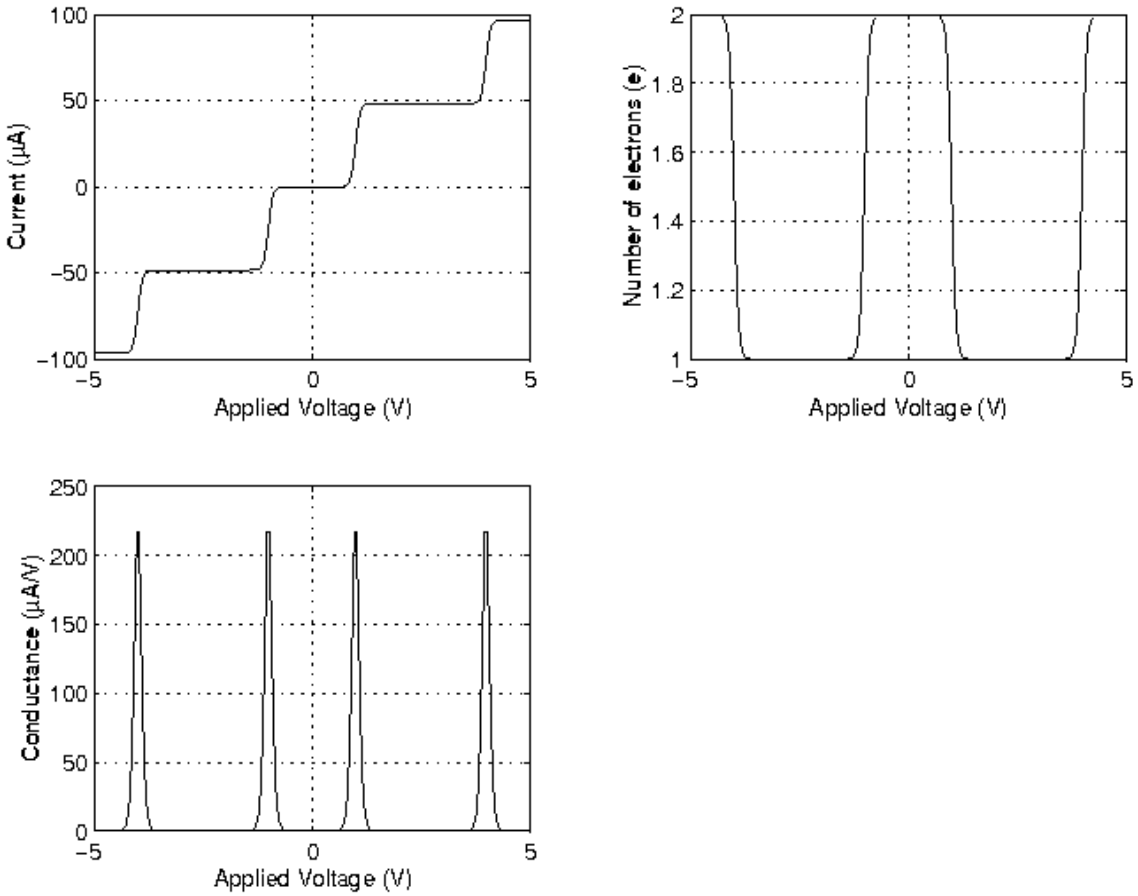

Figure 8: Plots of the Molecular Transistor Response using Molctoy; A Simplified Quantum Model

Further considerations must be taken into account when the device is considered as a subsystem inside a more complicated system. Applied voltage ranges are usually shorter than the range chosen at nanohub's simulations. Restricting the applied voltage to (-0.8v, 0.8v) displays a more accurate system behavior.

The next step in the validation process was to produce an equivalent model with code rewritten in VHDL-AMS. The VHDL-AMS code was run using the hAMSter Simulation System Version 2.0 from Ansoft Corporation on a PC equipped with an Intel Pentium M processor at 1.5 GHz. Simulation time for most of the hAMSter simulations was 20 milliseconds. VHDL-AMS coding adjustments must be performed according to

the limitations present of the available version of the hAMSter software. The model developed and simulated applied the ballistic principle. The VHDL-AMS model assumed no scattering, a constant Fermi level, used a grounded contact 1, low bias and a minimum broadening of molecular energy levels. A very important simplification was included regarding the Green's function solution. In the Matlab code the function is solved approximately. A value of NE, the energy grid parameter, and dU, the energy differential increment were selected as high as possible in order to obtain more iterations, (in the presented case dU>1e-6). In VHDL-AMS, the Green's function equation simplification was peformed by means of the fullfilment of an equilibrium condition defined by the statement given by:

$$IL=IR. \tag{62}$$

Partial results computed while this condition is not verified are not valid solutions for the transistor current. Simulation results show proper results in accordance with the quantum conductance definition. Discrete changes in conductance and its corresponding change in the transistor current were also in accordance with expectations derived from theory.

The VHDL-AMS code is shown in Table 3. Figures 9 through 15 present the results simplified molecular transistor behavior obtained from simulations of the VHDL-AMS code.

In Figure 9 the x-axis presents variations in the applied energy, which are equivalent to variations in the applied voltage as in the previous Matlab and nanohub models.

Table 3: VHDL-AMS Code of a Molecular Transistor

```
-- Model name: Molecular Transistor Level-0
--          This is a discrete model of a molecular transistor. The molecular resistance is associated
--          with the interface between the narrow wire and the wide contacts. Ballistic
--          transport model, zero scattering is assumed. Contact 1 is grounded
--          Low bias and minimum broadening of the molecular energy levels is assumed
--          This code is optimized to be simulated with the hAMSter tool by Ansoft Corporation
--          A constant Fermi level was assumed
--          The output voltage obtained represents only the positive part
LIBRARY IEEE;
USE IEEE.MATH_REAL.ALL;
-- entity definition including real quantities for scalar simplified model
ENTITY molectranslev0 IS
          QUANTITY mu1, mu2: REAL;
          QUANTITY eVg: REAL;
          QUANTITY USC: REAL;
          QUANTITY N, N0, N1, N2: REAL;
          QUANTITY ep: REAL;
          QUANTITY f1, f2: REAL;
          QUANTITY IL, LR: REAL;
          QUANTITY I, G: REAL;
   CONSTANT eta: REAL := 0.5;            -- charging coefficient: could be 0<eta<1
   CONSTANT ep0: REAL := -5.5;           -- <eV> molec-potential energy level in equlibrium
   CONSTANT Ef: REAL := -5.0;            -- <eV> Fermi level
   CONSTANT hbar: REAL := 1.1356e-15; --Planck's constant
   CONSTANT g1: REAL := 0.1;             -- Broadening coefficient gamma1
   CONSTANT g2: REAL := 0.1;             -- Broadening coefficient gamma2
   CONSTANT U: REAL := 0.001;            -- charging constant
   CONSTANT kT: REAL := 0.025;           -- Boltzman constant at room temperature
   CONSTANT q: REAL := 1.602e-19;        -- electron charge
END ENTITY molectranslev0;
ARCHITECTURE behav OF molectranslev0 IS
BEGIN
  N0  = =  2.0/(1.0 + exp((ep0 + Ef)/kT));   -- electrons in equilibrium state
  eVg = = now;                               -- (electronvolts) applied energy level
  USC = = eta*evg;                           -- charging voltage effect over the molecule
  mu1 = = Ef - (1.0 - eta)*eVg;              -- first contact energy level
  mu2 = = Ef + (eta*eVg);                    -- second contact energy level
  ep = = ep0 + USC;                          -- molecular energy level
  f1 = = 1.0/(1.0 + exp((ep-mu1)/kT));       -- Fermi level at the first contact
  f2 = = 1.0/(1.0 + exp((ep - mu2)/kT));     -- Fermi level at the second contact
  N1 = = 2.0*f1;                             -- number of charge carriers at first contact
  N2 = = 2.0*f2;                             -- number of charge carriers at second contact
  IL = = (2.0*g1*q/hbar)*(N1 - N);           -- right current
  IR = = (2.0*g2*q/hbar)*(N-N2);             -- left current
  IF (IL = IR) USE                           -- Equilibrium conditions verification routine
      N = = (2.0*((g1*f1)+(g2*f2))/(g1 + g2));
      I = = (2.0*q/hbar)*(g1*g2/(g1 + g2))*(f1 - f2);
   ELSE
     N = = ((USC/U)+2.0*N0) - N;             -- charging condition
     I = = IR-IL;
  END USE;
END ARCHITECTURE behav;
```

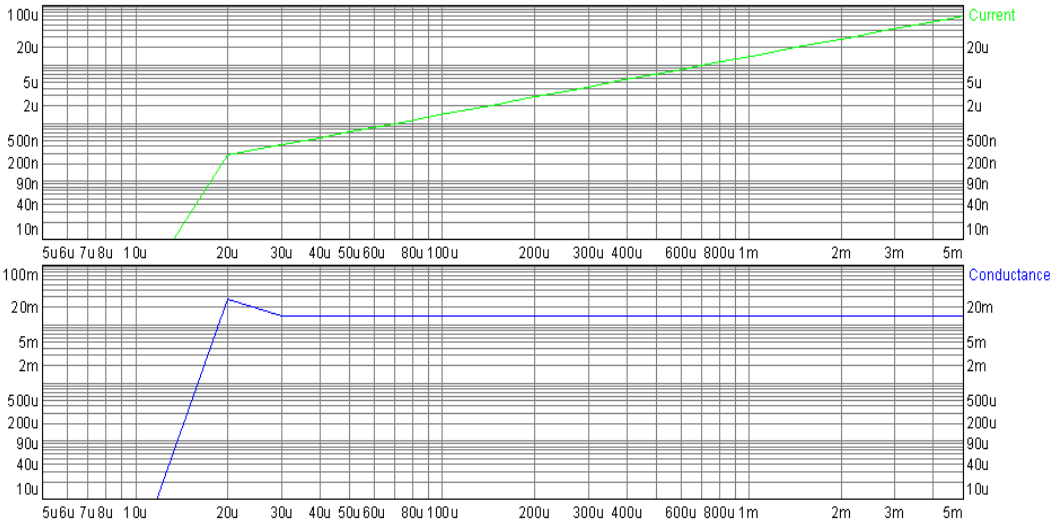

Figure 9: Conductance and Current Variations from the VHDL-AMS Model Level-0

Regarding the adequate scales and the independent variable selected, these results show a proper concordance with the other models. A discrete variation in conductance is reflected in a corresponding variation in the current. The main conclusion is that the molecular device is not showing the classical continous response. Instead, the response is discrete following the quantized approach revealed in the theory.

The VHDL-AMS simulations displayed a difference with respect to the other simulations, which is evident in simulation time and the amount of computational resources required. This simulation was performed in 20 milliseconds using a laptop equipped with an Intel Pentium M processor at 1.5 GHz.

Analysis from the VHDL-AMS entity-architecture pair was performed. Different aspects are considered:

- A charging coefficient equal to zero means no current flow. A non-linear response was displayed when the charging coefficient was increased from 0 to 1, which corresponded to an increasing current. Current variations are not

directly related to charging coefficient variations. These non-linearities are effective in a remarkable manner when current values are small. No significant differences can be noted for currents higher than 100 mA, which is not the case of most of nanodevices. These results are presented in Figure 10.

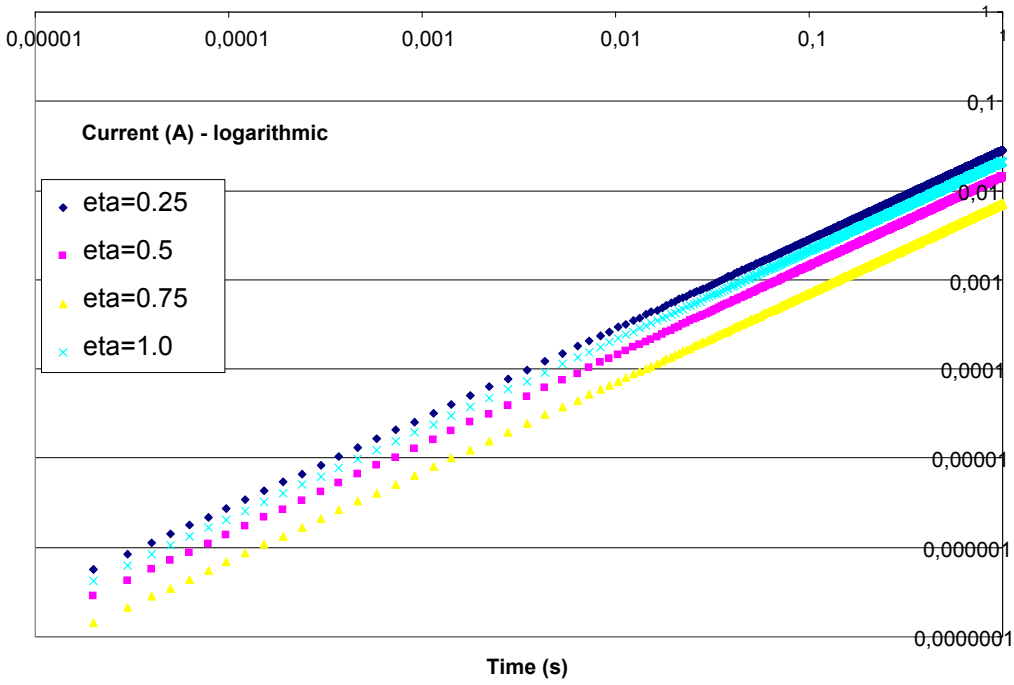

Figure 10: Current Variations with the Charging Coefficient from the VHDL-AMS model Level-0

- The results for molecular nature and its potential energy level, (ep0 in the VHDL-AMS coding) are related to the consideration of the potential energy. They are presented in Figure 11. Significant differences in the current flow can be noted when the molecular energy level is less than the Fermi level. Starting at 20 uA, current differences of approximately two orders of magnitude can be discerned between simulations with $e_{p0}<E_f$ and $e_{p0}>E_f$.

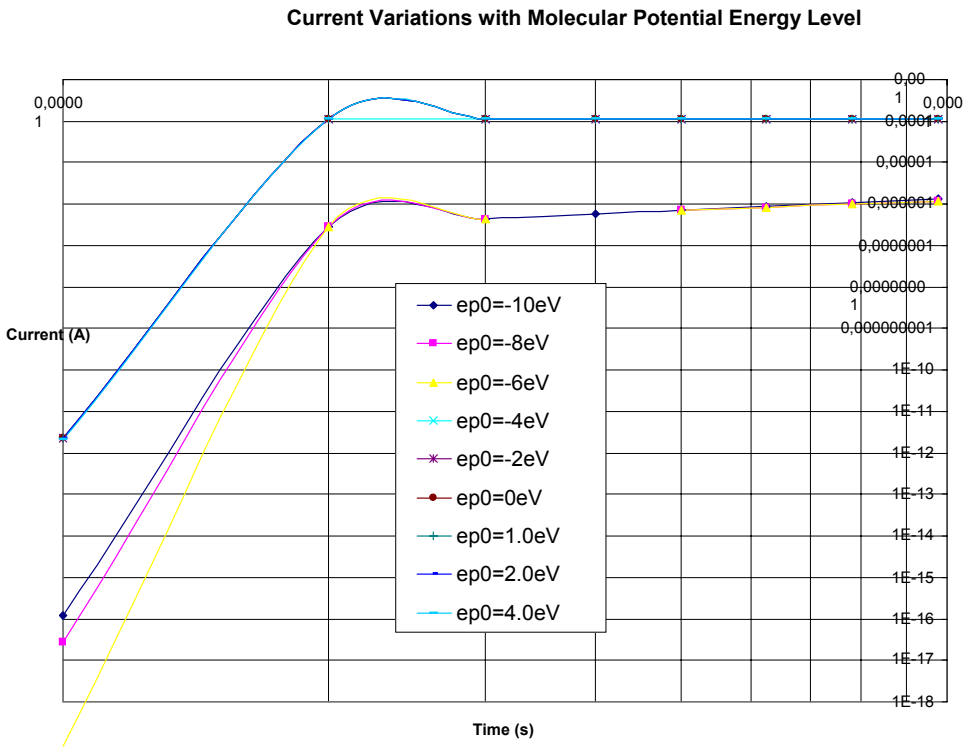

Figure 11: Current Variations with the Molecular Potential Energy Level from the VHDL-AMS Model Level-0

- A similar situation can be noted with variations of the Fermi energy level with respect to the molecular potential energy level. Results of these investigations are presented in Figure 12. While maintaining e_{p0} constant at -5.5 eV, reduction of the Fermi energy to a level more negative than e_{p0} yields a higher transistor current of approximately one or two orders of magnitude. As with previous observations, this fact was verified at low current levels. For currents higher than 10 mA, this situation was not verified.

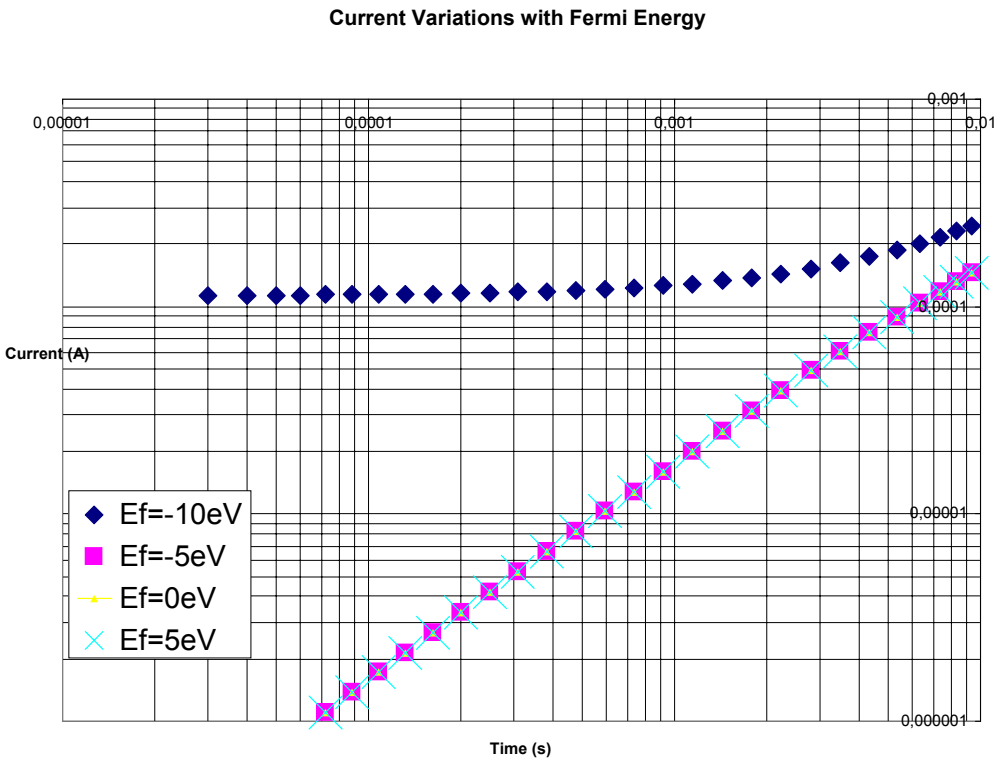

Figure 12: Current Variations with the Fermi Energy from the VHDL-AMS Model Level-0

- As theoretical principles reveal, the effect of broadening corresponds to the coupling of the molecular energy levels with the source and drain regions. For the case of symmetrical transistors where the broadening effect can be considered the same at both interfaces, this broadening produces higher charge carrier amounts flowing through the transistor. Increasing of the gamma coefficient was followed by proportional current increasings. After approximately 10 mA, this situation was maintained only for gamma values less than 0.01. The results of these investigations are presented in Figure 13.

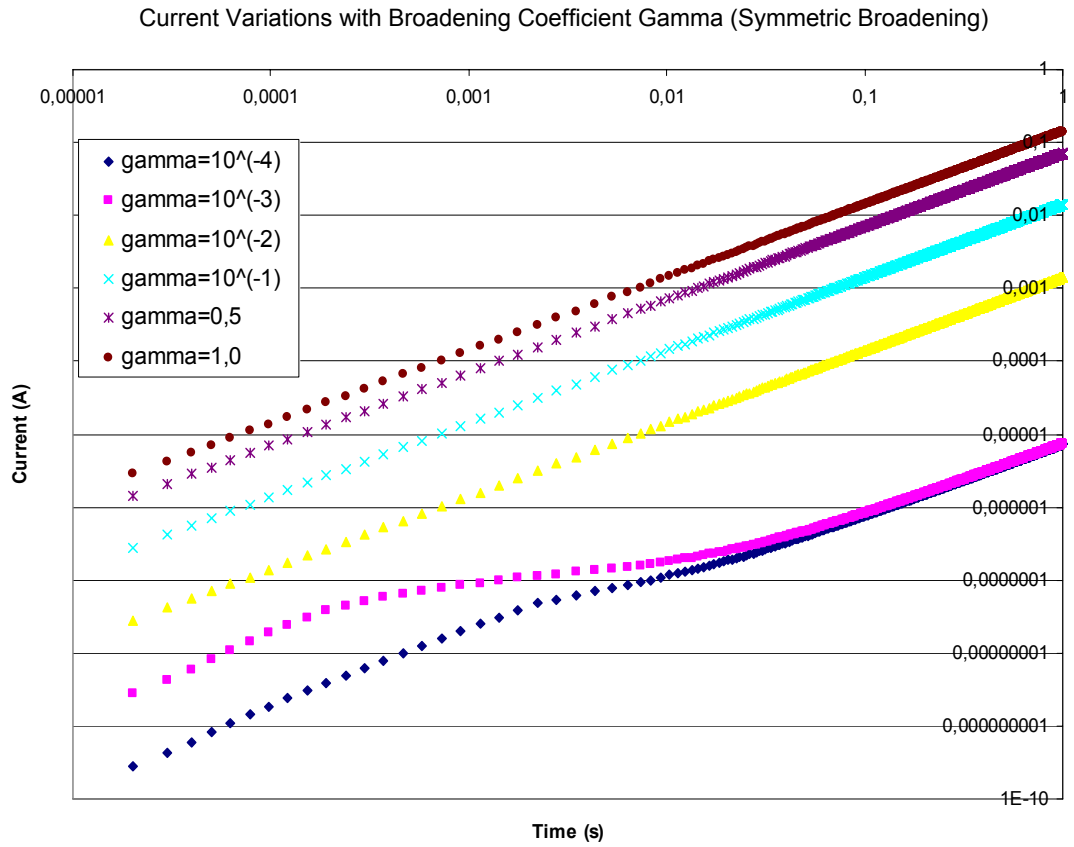

Figure 13: Current Variations with Broadening Coefficient Gamma (Symmetric Case) from the VHDL-AMS Model Level-0

- In the case of a non-symmetric transistor, which means that gamma 1 and gamma 2 are different, similar current variations were obtained when gamma 1 and gamma 2 values were interchanged. This fact reinforces the idea of the absolute broadening factor, which was included in the Matlab code as:

$$g=g1+g2. \tag{63}$$

  The effect of broadening at each contact was linearly added to the broadening effect of the other.

- Temperature effects were also analyzed. The results of these investigations are presented in Figure 14. The logarithmic dependence of the current with respect to

the temperature is displayed. An additional current increase was obtained in high temperature regimes and for current ranges no greater than 1 mA. A difference of hundreds of nanoamperes was displayed at high temperaures. According to the applied model very small differences can be obtained for higher current ranges.

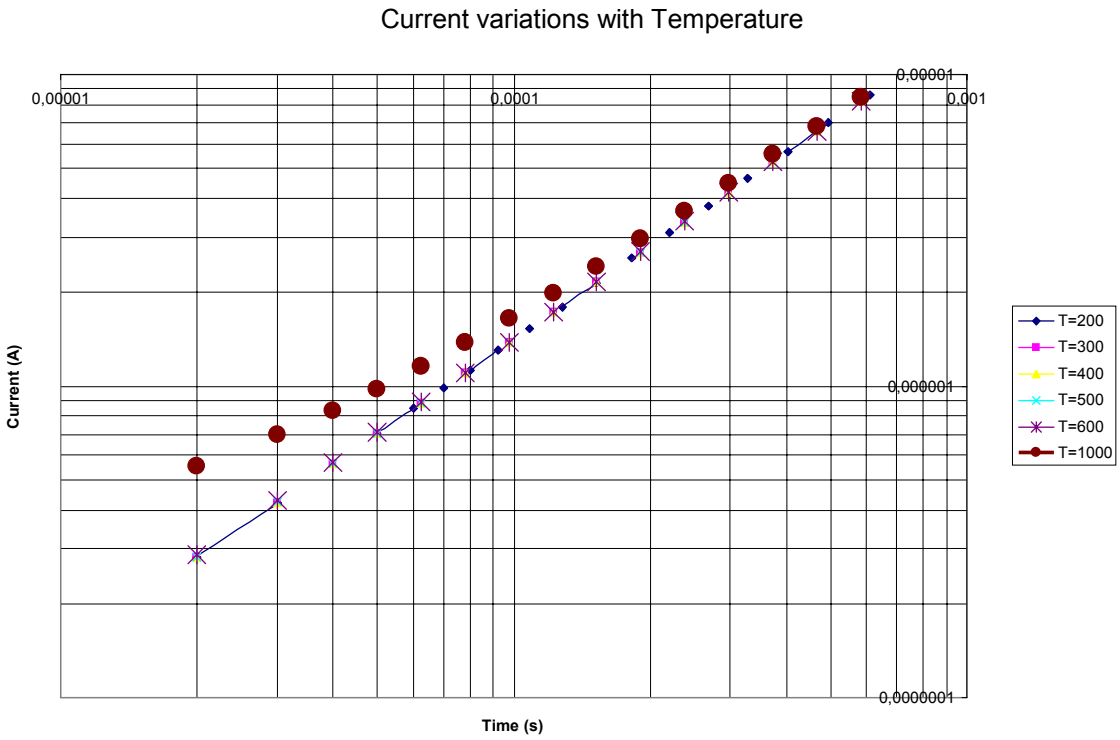

Figure 14: Current Variations with Temperature - VHDL-AMS Model Level-0

The next question with respect to the model was the verification of noise being generated inside the device. The results of this investigation are presented in Figure 15. It was verified that with a minimum amount of evg, up to 7.5 e-21 eV, the noise current was not greater than 2.5 e-48 A.

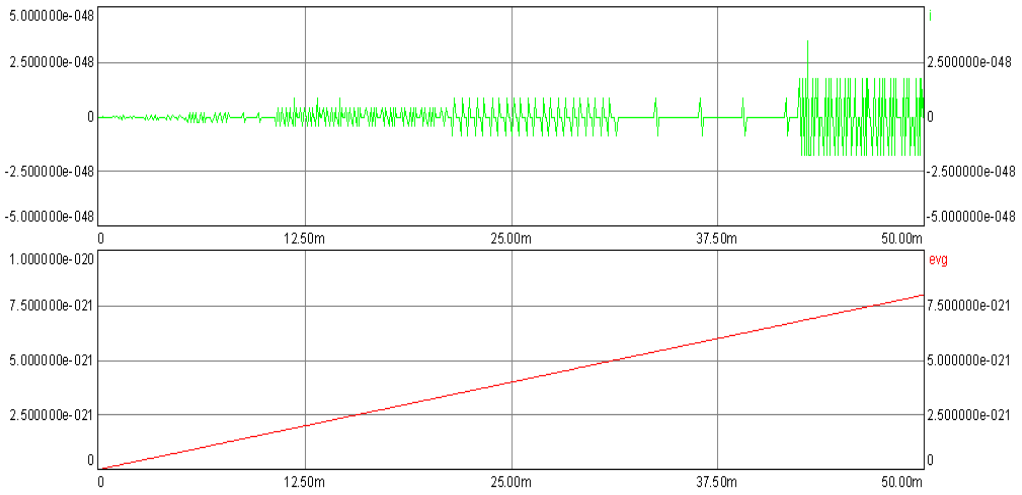

Figure 15: Noise Generation Inside the VHDL-AMS Model Level-0

Finally, the molecular transistor model was exported, with a SCORM utility, to be converted to a shareable learning object. Figure 16 presents the appearance of the model using a standard internet browser.

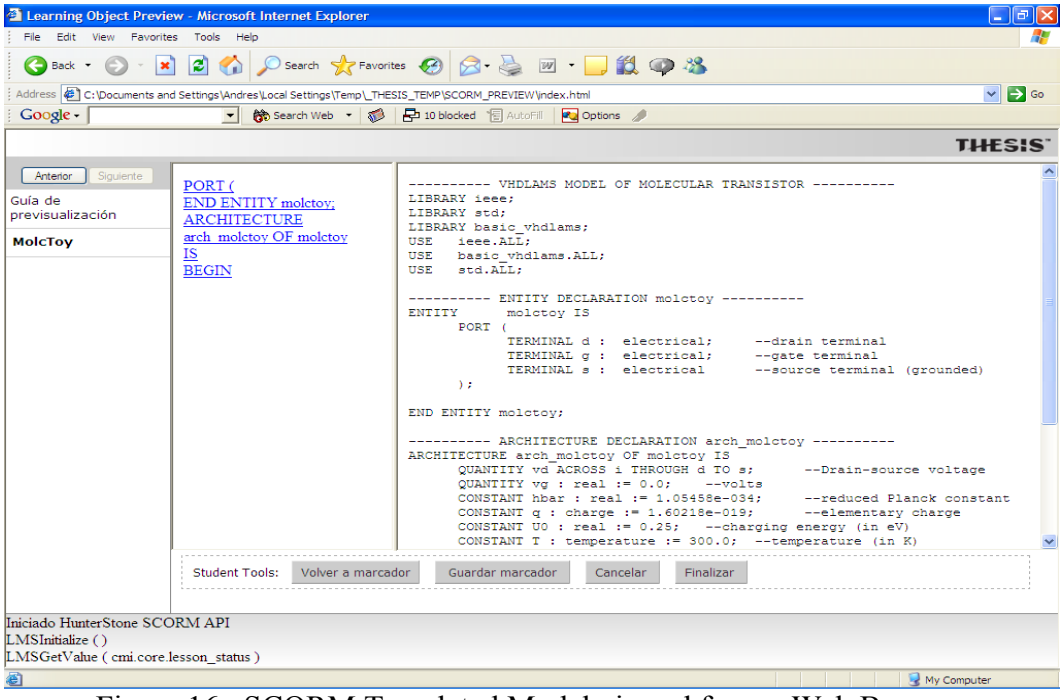

Figure 16: SCORM Translated Model viewed from a Web Browser

For an adequate model translation to SCORM format a main document was taken as a basis for the Hypertext Markup Language (HTML) and Extensible Markup Language (XML) files. The HTML file appears exactly like the original VHDL-AMS code, which was simulated with the hAMSter tool. The XML file contains a set of metadata definitions needed to translate the original format and to achieve a proper presentation of the information inside a Learning Management System (LMS). Appendix C presents the XML coding required for presentation of the VHDL-AMS model for the LMS. An open source LMS, called Moodle, was used in order to demonstrate the applicability of the nanodevice models at undergraduate and graduate engineering courses at Universidad Distrital Francisco Jose de Caldas, Bogota, Colombia.

From these results, more complicated structures can be modeled and simulated. Simplifications can be modified in order to obtain a more accurate response. However, simulation time and computational resources may be higher, which will make their incorporation with other CAD tools and hardware description languages expensive.

4.3. **Circuits Including the Proposed Model**

The next step in the validation process was to construct electrical circuits where the proposed model was included. The purpose of this investigation was to demonstrate applicability of the model inside more complex designs, demonstrate model interoperability with other existing common devices and demonstrate model validity for other integration scales. The proposed circuits are simple but allow verifying the applicability of the original nanodevice within a more complex architecture.

4.3.1. Analog Circuits Test Bench

Figure 17 provides a schematic view of an analog circuit including the already tested device.

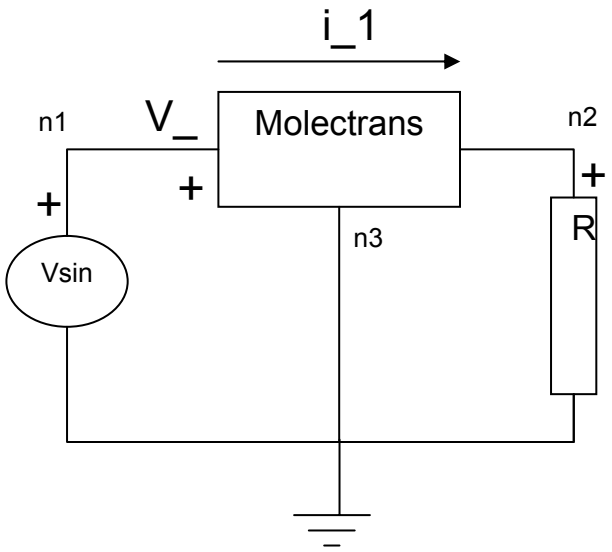

Figure 17: Analog Circuit Test Bench Including the Original Model

The VHDL-AMS code was organized as follows. Initially, the original molecular transistor entity was included. Following the proposed modeling methodology, the initial molecular transistor device model needed to be related with other entities. Following the code presented in the following paragraphs it should be noted that the original entitiy now has an explicit line coding the interface ports. Prior formulation of the transistor had a purely functional architecture without major concerns for its connectivity. Now, the entity-architecture pairs need additional properties in order to connect with other entities. This is verified by the inclusion of PORT statements at the beginning of the entity definition. Then, the other entities related to micro or macroscale devices are defined. It is ensuring the reusability and compatibility of the proposed nanodevice models. Proper

PORT MAP statements need to be written in order to achieve an electrical connectivity among circuit elements. The test bench concludes the statements of the complete circuit. Table 4 shows the complete coding.

Table 4: VHDL-AMS Code Integrating the Models

```
LIBRARY IEEE, DISCIPLINES;
USE IEEE.MATH_REAL.ALL;
USE DISCIPLINES.ELECTROMAGNETIC_SYSTEM.ALL;
ENTITY molectrans IS
   PORT(TERMINAL gate, drain, source: ELECTRICAL);  -- Interface ports
END ENTITY molectrans;
ARCHITECTURE behav OF molectrans IS
   QUANTITY Vgate ACROSS gate TO electrical_ground;
   QUANTITY Vdraingate ACROSS Idrain THROUGH drain TO gate;
   QUANTITY eVg, mu1, mu2, USC, N, ep, f1, f2: REAL;
   QUANTITY N0, N1, N2, IL, IR, Vdrain, Vsource: REAL;
   CONSTANT eta: REAL := 0.5;                -- charging coefficient. it could be 0<eta<1
   CONSTANT ep0: REAL := -5.5;               -- <ELECTRONVOLTS> molecular
                                             -- potential energy level in equlibrium
   CONSTANT Ef: REAL := -5.0;                -- <ELECTRONVOLTS> Fermi level
   CONSTANT hbar: REAL := 1.1356e-15;        --Planck's constant
   CONSTANT g1: REAL := 0.1;                 -- Broadening coefficient gamma1
   CONSTANT g2: REAL := 0.1;                 -- Broadening coefficient gamma2
   CONSTANT U: REAL := 0.001;                -- charging constant
   CONSTANT kT: REAL := 0.083;               -- Boltzman's constant at room temperature
   CONSTANT q: REAL := 1.602e-19;            -- electron charge
BEGIN
          Vsource = = 0.0;                   -- source grounded
          Vdrain = = Vdraingate - Vgate;
          N0 = = 2.0/(1.0 + exp((ep0 + Ef)/kT)); -- electrons in equilibrium state
          eVg = = Vgate;                     -- (electronvolts) applied energy level
          USC = = eta*evg;                   -- charging voltage effect over the molecule
          mu1 = = Ef-(1.0 - eta)*eVg;        -- first contact energy level
          mu2 = = Ef + (eta*eVg);            -- second contact energy level
          ep = = ep0 + USC;                  -- molecular energy level
          f1 = = 1.0/(1.0 + exp((ep - mu1)/kT));  -- Fermi level at the first contact
          f2 = = 1.0/(1.0 + exp((ep - mu2)/kT));  -- Fermi level at the second contact
          N1  = =  2.0*f1;                   -- number of charge carriers at first contact
          N2 = = 2.0*f2;                     -- number of charge carriers at second contact
          IL = = (2.0*g1*q/hbar)*(N1-N);     -- right current
          IR = = (2.0*g2*q/hbar)*(N - N2);   -- left current
-- Equilibrium conditions verification routine
  IF (IL = IR) USE
       N = = (2.0*((g1*f1) + (g2*f2))/(g1 + g2));
       Idrain = = (2.0*q/hbar)*(g1*g2/(g1 + g2))*(f1 - f2);
  ELSE
     N = = ((USC/U) + 2.0*N0) - N;           -- charging condition
     Idrain = = IR-IL;
  END USE;
END ARCHITECTURE behav;
```

Table 4 (Continued)

```
LIBRARY IEEE;
USE IEEE.MATH_REAL.ALL;
ENTITY vsin IS
   PORT(QUANTITY v_in: REAL);
END ENTITY vsin;

ARCHITECTURE behav OF vsin IS
BEGIN
   v_in = = sin(1.0e12*now);
END ARCHITECTURE behav;
-----------------------------------------------------------------------------------------------
LIBRARY DISCIPLINES;
USE DISCIPLINES.ELECTROMAGNETIC_SYSTEM.ALL;
ENTITY resistor IS
   GENERIC(resistance: REAL);                    -- resistance value given as a generic parameter
   PORT (TERMINAL p,m: ELECTRICAL);              --Interface ports
END ENTITYresistor;

ARCHITECTURE behav OF resistor IS
   QUANTITY r_e ACROSS r_i THROUGH p TO m;
BEGIN
   r_i = = r_e/resistance;
END ARCHITECTURE behav;
```

```
-- The test bench is the mechanism employed to simulate a VHDL-AMS design entity.
-- Test bench
LIBRARY DISCIPLINES, IEEE;
USE DISCIPLINES.ELECTROMAGNETIC_SYSTEM.ALL;
USE IEEE.MATH_REAL.ALL;
ENTITY Test_bench_Level0 IS
END ENTITY Test_bench_Level0;
ARCHITECTURE behav OF Test_bench_Level0 IS
          TERMINAL n1, n2, n3: ELECTRICAL;
          QUANTITY v_input: REAL;
          QUANTITY v_1 ACROSS n1 TO n3;
          QUANTITY i_1 THROUGH n2 TO n1;
BEGIN
   VSource: ENTITY vsin(behav) PORT MAP(v_in => v_1);
   T1: ENTITY molectrans(behav) PORT MAP(gate => n1, drain => n2, source => n3
   );
   R1: ENTITY resistor(behav) GENERIC MAP(resistance => 500.0)
                                    PORT MAP(p => n2, m => electrical_ground);
   v_1 = = v_input;
END ARCHITECTURE behav;
```

Without regard to the circuit gain, which depends on the size of the connected load, a proper current-voltage response at the load was obtained.

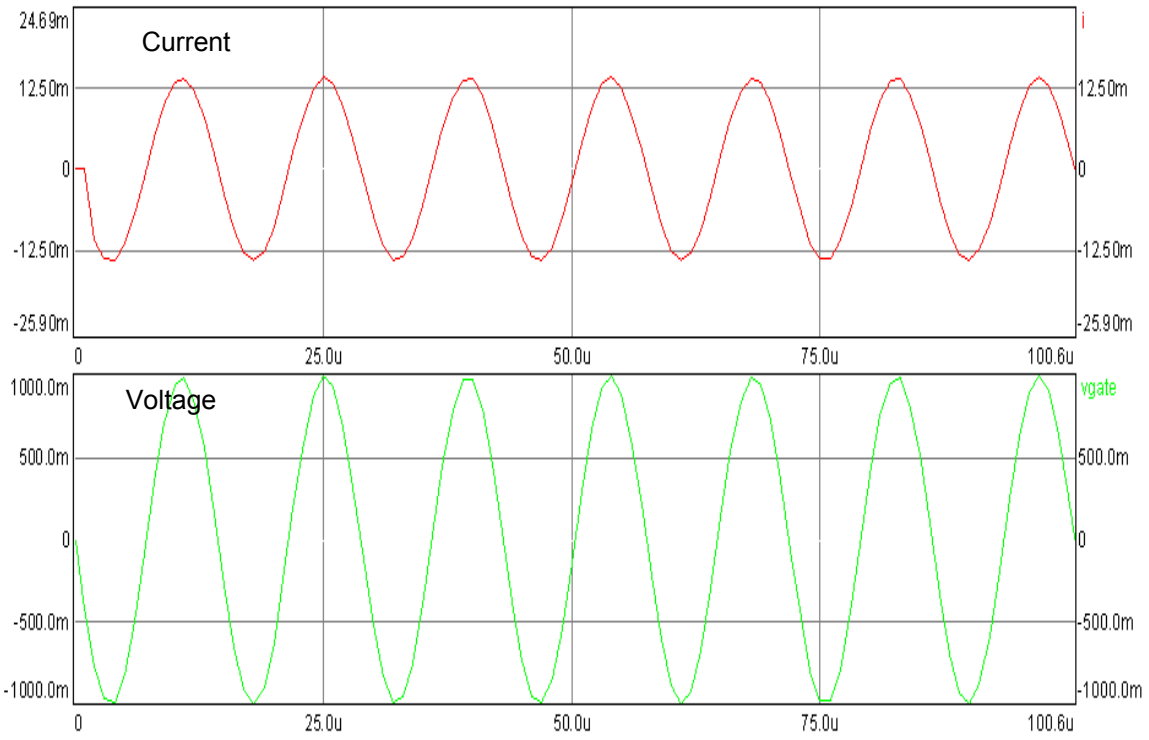

Figure 18: Analog Circuit Test Bench Response

4.3.2. Digital Circuits Test Bench

A very simple two input NAND gate circuit using two molecular transistors has been formulated. Figure 19 shows the schematic circuit corresponding to this gate.

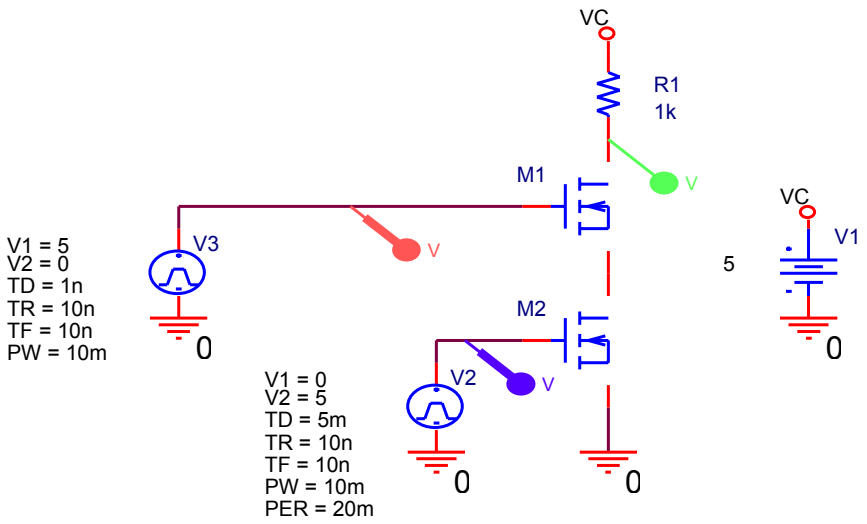

Figure 19: Basic NAND2 Gate Schematic using Two Molecular Transistors

The NAND2 gate is following the true table as was expected. As it can be seen on Figure 20, the two inputs (brown and blue traces) yield the output (green trace).

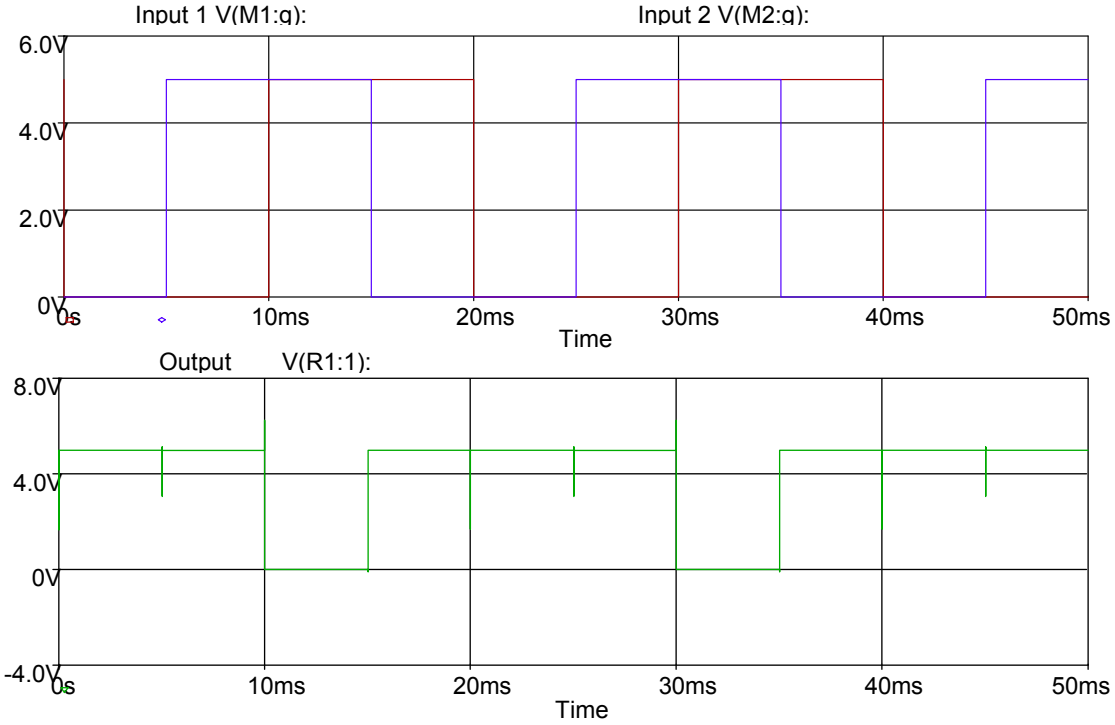

Figure 20: Input – Output Behavior for the NAND2 Gate

The VHDL-AMS coding for the two input NAND gate is presented in Table 5. The next step was to model and simulate a four transistor NAND gate. The schematic circuit is presented in Figure 21 and the VHDL-AMS coding for the circuit is presented in Table 6.

Table 5: VHDL-AMS Code for a Simple Two-Input NAND Gate

```
   LIBRARY IEEE;
   USE ieee.all;
   USE work.all;
   ENTITY Test_bench_Level0 IS
   END Test_bench_Level0 ;
   ARCHITECTURE behav1 OF Test_bench_Level0  IS
        TERMINAL d,g,s,d1,g1,s1,vc: ELECTRICAL;
        QUANTITY VXTOG ACROSS IXTOG THROUGH g TO electrical_ref;
                -- Applied signal at M1 Gate
        QUANTITY VXTOG1 ACROSS IXTOG1 THROUGH g1 TO electrical_ref;
                -- Applied signal at M2 Gate
        QUANTITY a,b: real;
   BEGIN
        sino: ENTITY v_sin PORT MAP(seno=>a,coseno=>b);  -- square signals mapping
        vgs: ENTITY v_constant GENERIC MAP (level=>5.0) PORT MAP
(pos=>vc,neg=>electrical_Ref);
        nmos: ENTITY molectrans PORT MAP (drain=>d1,gate=>g1,source=>s1);  -- Transistor M1
        nmos1: ENTITY molectrans PORT MAP (drain=>s1,gate=>g,source=>electrical_ref);
                -- Transistor M2
        re: ENTITY resistor GENERIC MAP (resistance => 1000.0) PORT MAP (p=>vc, m=>d1);
          -- Resistor R1
   IF a>0.0 USE vxtog==5.0;
          ELSE vxtog==0.0;
          END USE;
   IF b>0.0 USE vxtog1==0.0;
          ELSE vxtog1==5.0;
          END USE;
END ARCHITECTURE behav1;
```

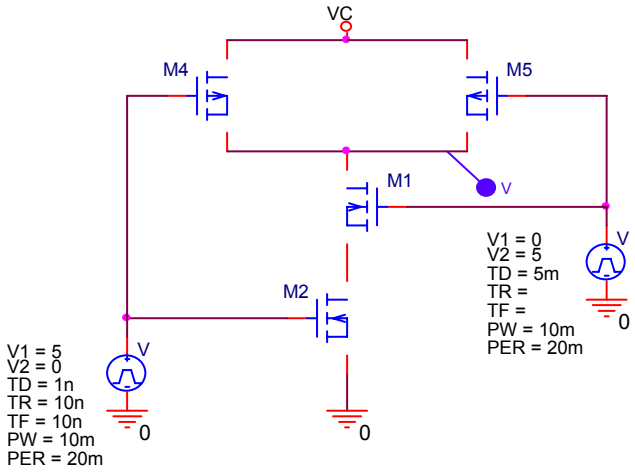

Figure 21: Two - Input NAND Circuit with
Four Molecular Transistors

Table 6: VHDL-AMS Code for a Two-Input, Four-Transistor NAND Gate

```
LIBRARY IEEE;
USE IEEE.MATH_REAL.ALL;
LIBRARY IEEE;
USE IEEE.ELECTRICAL_SYSTEMS.ALL;
USE work.all;
ENTITY Test_bench_Level0 IS
END Test_bench_Level0 ;
ARCHITECTURE behav1 OF Test_bench_Level0  IS
    TERMINAL g1,g2,g3,g4,vc,s1,s2: electrical;
    QUANTITY VXTOG1 ACROSS IXTOG1 THROUGH g1 TO electrical_ref;
    QUANTITY VXTOG2 ACROSS IXTOG2 THROUGH g2 TO electrical_ref;
    QUANTITY VXTOG3 ACROSS IXTOG3 THROUGH g3 TO electrical_ref;
    QUANTITY VXTOG4 ACROSS IXTOG4 THROUGH g4 TO electrical_ref;
    QUANTITY a,b: real;
BEGIN
    sino: ENTITY v_sin PORT MAP(seno=>a,coseno=>b);
    vgs: ENTITY v_constant GENERIC MAP (level=>5.0) PORT MAP (pos=>vc, neg=>electrical_Ref);
    T1: ENTITY molectrans PORT MAP (drain=>vc,gate=>g1, source=>s1);
    T2: ENTITY molectrans PORT MAP (drain=>vc,gate=>g2, source=>s1);
    T3: ENTITY molectrans PORT MAP (drain=>s1,gate=>g3, source=>s2);
    T4: ENTITY molectrans PORT MAP (drain=>s2,gate=>g4, source=>electrical_ref);
          vxtog4==5.0-vxtog2;
          vxtog3==5.0-vxtog1;
  IF a>0.0 USE vxtog1==5.0;
          ELSE vxtog1==0.0;
          END USE;
IF b>0.0 USE vxtog2==0.0;
          ELSE vxtog2==5.0;
          END USE;
END ARCHITECTURE behav1;
```

A similar response to the two-transistor circuit was obtained in this case. It shows that both formulations can be used, but the second one is electrically more appropriate than the first one.

CHAPTER 5

CONCLUSIONS

The methodology applied to model nanoscale devices and systems simplifies calculations for integrated simulation environments. In addition the methodology also incorporates consideration of quantum effects, which are always present in these kinds of systems. Results were comparable with other reports obtained when using more complex computational facilities and more elaborated mathematical models. Results were also in accordance with experimental results from other research groups.

The applied methodology was implemented through various steps using different tools. However, the implementation can be reordered depending on the initial conditions of the particular system definition. The application of quasi-continuum models as a middle point between quantum and continuum models was validated. The validation process can require different tools in order to achieve a better selection of system parameters and depends on the specific analysis performed. The modularity of the proposed methodology ensures an efficient validation process and the interaction from different stages of the process to various verification tools.

"Lumped" models, where small sets of electric circuit elements represent the behavior of devices, were tested. These models were shown to be of limited use when nano-devices coexist with microdevices, specifically when beam thickness is large

compared to features thickness. Other properties can be analyzed in order to apply the models in other domains.

Nano-device models can be represented using common hardware description languages such as VHDL-AMS. The use of VHDL-AMS provides affordable results, which can be applied to common design engineering environments with typical operating conditions. The application of VHDL-AMS is compatable with current requirements imposed by industry and will evolve to always be compatable with industry demands.

Nano-device models can be properly translated to standard object oriented formats in order to be shareable as a web resource. In accordance with the feedback obtained from educational experiences, nanodevice models have been successfully reused inside various simulation environments. In addition, these models were used in more complex designs, which were more easily understood by undergraduate students in electrical engineering programs.

Working conditions of most molecular nanotransistors were properly translated into a VHDL-AMS modeling and simulation environment. The VHDL-AMS environment yielded affordable results in accordance with common operating conditions reported for current electronic nanodevices.

The nano-device models developed were properly translated to standard object oriented formats in order to be shareable from standard Learning Management Systems. As a consequence, the models can be shared as a web resource and can be reused in environments such as collaborative research, development groups and various educational situations.

REFERENCES

[1]. Shilkrot L.E., and Miller R.E., "A Coupled Atomistic-Continuum Model of Defects in Solids", Journal of the Mechanics and Physics of Solids, Vol 50, pp 2085 - 2106, 2002

[2]. Jenkins J.W., Sundaram S. and Makhijani V.B., "Coupling between Nanoscale and Microscale Modeling for Microfluidic devices", CFD Research Corporation

[3]. Yu Z., Dutton R. and Kiehl R., "Circuit/Device Modeling at the Quantum Level", IEEE Transactions on Electron Devices, Vol. 47, No. 10, October 2000

[4]. Senturia S., "Microsystem Design", Kluwer Publishers. ISBN: 0-7923-7246-8, 2001

[5]. Ai S. and Pelesjo J., "Dynamics of a Canonical Electrostatic MEMS/NEMS System", School of Mathemathics and CDSNS, Georgia Institute of Technology

[6]. Lurie S., Belov P., Volkov-Bogorodsky D. and Tuchkova N., "Nanomechanical Modeling of the Nanostructures and Dispersed Composites", Computational Materials Science, vol 28, pp. 529 - 539, 2003

[7]. Ferry D. and Goodnick S., "Transport in Nanostructures", Cambridge Univ. Press, 1999

[8]. Pierret R., "Advanced Semiconductor Fundamentals", Modular Series on Solid State Devices, Addison-Wesley, 1989

[9]. Damle P., "Nanoscale Device Modeling: from MOSFETS to Molecules", Purdue University Ph.D. Thesis, at http://falcon.ecn.purdue.edu:8080/publications/#theses, 2003

[10]. Liu W., "Computational Nanomechanics of Materials", Handbook of Theoretical and Computational Nanotechnology, American Scientific Publishers, 2005

[11]. Liu W., "The Spectral Grid Method: A Novel Fast Schrödinger-Equation Solver for Semiconductor Nanodevice Simulation", IEEE Transactions on Computer-Aided Design Of Integrated Circuits and Systems, Vol. 23, No. 8, August 2004

[12]. Sano N., “Device Modeling and Simulations Toward Sub-10nm Semiconductor Devices”, IEEE Transactions on Nanotechnology, Vol. 1, No. 1, March 2002

[13]. Kosina H., “Comparison of Numerical Quantum Device Models”, Internat. Conf. on Simulation of Semiconductor Processes and Devices, (SISPAD), pp. 171 - 174, 2003

[14]. Schweizer W., “Numerical Quantum Dynamics”, Kluwer Academic Publishers, 2001

[15]. Mello P. and Kumar, N., “Quantum Transport in Mesoscopic Systems: Complexity and Statistical Fluctuations”, Oxford University Press, 2004

[16]. Belsky V., Beall M., Fish J., Shephard M. S. and Maa S.G., “Computer-Aided Multiscale Modeling Tools For Composite Materials and Structures”, Computing Systems in Engineering, Vol. 6, No. 3, pp. 213 - 223, 1995

[17]. Voigt G., Schrag G. and Wachutka P., “Microfluidic System Modeling using VHDL-AMS and Circuit Simulation”, Microelectronics Journal, Vol 29, pp. 791-797, 1998

[18]. Dewey A., Srinivasan V., Icoz E. “Visual Modeling and Design of Microelectro-mechanical System Transducers”, Microelectronics Journal, Vol 32, pp. 373 - 381, 2001

[19]. Gregoire O., Souffland D., Gauthier, S. and Schiestel, R., “Towards Multiscale Modeling of Compressible Mixing Flows”, Academic Science, T. 325, Series II b, pp. 631-634, Paris, 1997

[20]. Endeman A., and Dunnigan M., “System Level Simulation of a Double Stator Wobble Electrostatic Micromotor”, Sensors and Actuators, A 99, pp. 312 – 320, 2002

[21]. Voigt P., Schrag G. and Wachutka G., “Electrofluidic Full-System Modeling of a Flap-Valve Micropump Based on Kirchhoffian Network Theory”, Sensors and Actuators, A 66, pp. 9 - 14, 1998

[22]. Garcia S., “Mixed-Mode System design: VHDL-AMS”, Microelectronic Engineering, vol 54, pp. I71 - I80, 2000

[23]. Solomon P. and van Heeren H., “Worldwide Services and Infrastructure for Nano and Microproduction. Nanotsunami”, at http://www.voyle.net/Guest%20Writers/Patric%20Salomon/Patric%20Salomon%202004-0001.htm

[24]. Wilson P., “Multiple Domain Behavioral Modeling using VHDL-AMS”, IEEE, ISCAS, 2004

[25]. Nikitin P., “Modeling Partial Differential Equations in VHDL-AMS”, IEEE, 2003

[26]. Schlegel M., “Analyzing and Simulation of MEMS in VHDL-AMS Based on Reduced-Order FE Models”, IEEE, 2005

[27]. Karayannis T., “Using XML for Representation and Visualization of Elaborated VHDL-AMS Models”, IEEE, 2000

[28]. http://nanotitan.org

[29]. Barniol N., “A Mass Sensor with Attogram Sensitivity using Resonating Cantilevers”, Poster, 10th MEL-ARI/NID Workshop, Helsinki-Finland, July 1-3, 2002

[30]. Born M. and Huang H., “Dynamical theory of crystal lattices”, Oxford University Press, 1954

[31]. Gerousis J., “Nanoelectronic single-electron transistor circuits and architectures” International Journal of Circuit Theory and Applications, Vol 32, pp. 323 - 338, 2004

[32]. Albella J., Martinez J. and Agullo, F., “Fundamentos de Microelectronica, Nanoelectronica y Fotonica”, Pearson-Prentice Hall. Madrid, 2005

[33]. Stan M., Franzon P., Goldstein S., Lach J. and Ziegler M. “Molecular Electronics: From Devices and Interconnect to Circuits and Architecture”, IEEE Invited Paper 0018-9219, DOI 10.1109/JPROC, 2003

[34]. Repinsky S., “Organized Molecular Assemblies: Creation and Investigation of their Functional Properties”, e-Journal of Surface Science and Nanotechnology, Vol.1, pp. 7 - 19, 2003

[35]. Henderson S., Johnson E., Janulis J. and Tougaw P., “Incorporating Standard CMOS Design Process Methodologies into the QCA Logic Design Process”, IEEE Transanctions on Nanotechnology, Vol. 3, No. 1, March, 2004

[36]. Snider G.and Williams S., “Nano/CMOS Architecture using a Field-Programmable Nanowire Interconnect”, Nanotechnology, Vol. 18, pg. 11, 035204 DOI: 10.1088/0957-4484/18/3/035204, 2007

[37]. Ma X., Strukov D., Lee J. and Likharev K. "Afterlife for Silicon: CMOL Circuit Architectures", Proceedings of the 5TH IEEE Conference on Nanotechnology, 2005

[38]. Strukov D. and Likharev K., "CMOL FPGA: A Reconfigurable Architecture for Hybrid Circuits with Two-Terminal Nanodevices", Nanotechnology, Vol. 16, pp. 888-900, DOI: 10.1088/0957-4484/16/6/045, 2005

[39]. Winfree E., "Algorithmic Self-Assembly of DNA", Proceedings of the International Conference on Microtechnology In Medicine and Biology, DOI: 1-4244-0338-3/06, 2006

[40]. Pecheux F., Lallement C. and Vachoux A., "VHDL-AMS and Verilog-AMS as Alternative Hardware Description Languages for Efficient Modeling of Multidiscipline Systems", IEEE Transactions on Computer-Aided Design, DOI: 10.1109/TCAD.841071, 2004

[41]. Lyshevski S., "Nanotechnology and Super High-Density Three-Dimensional Nanoelectronics and NanoICs", IEEE, DOI: 0-7803-7976-4/03, 2003

[42]. Lundstrom M.," Nanotransistors: A Bottom-Up View", IEEE, DOI: 1-4244-0301-4/06, 2006

[43]. Ravariu C., "A NOI-Nanotransistor", IEEE, DOI: 0-7803-9214-0/05, 2005

[44] Pregaldiny F., Kammerer J. and Lallement C., "Compact Modeling and Applications of CNTFETs for Analog and Digital Circuit Design", IEEE, DOI: 1-4244-0395-2/06, 2006

[45]. Drexler E., "Toward Integrated Nanosystems: Fundamental Issues in Design and Modeling", Journal of Computational and Theoric Nanoscience, Vol. 3, pp. 1-10, 2006

[46]. Mylvaganam K. and Zhang L., "Ballistic Resistance Capacity of Carbon Nanotubes", Nanotechnology, Vol. 18, pg. 4, 475701 DOI: 10.1088/0957-4484/18/47/475701, 2007

[47]. Schlegel M., Bennini F., Mehner J., Herrmann G., Müller D. and Dözel W., "Analyzing and Simulation of MEMS in VHDL-AMS Based on Reduced Order FE-Models", IEEE Sensors, 2nd IEEE Intern. Conf. on Sensors, Toronto, Canada, 2003

[48]. Schlegel M., Herrmann G. and Müller D., "Application of a Multi-Architecture Modeling Design Method in System Level MEMS Simulation", DTIP 2003, Mandelieu-La Napoule, France, 2003

[49]. Karray M., Desgreys P. and Charlot J., "VHDL-AMS Modeling of VCSEL Including Noise", IEEE, DOI: 0-7803-8135-1, 2003

[50]. Ewing R. and Cabrera E., "Scaling Issues Addressed for Mixed-Signal Design by use of Dimensional Analysis Methodology and VHDL-AMS", IEEE, DOI: 0-7803-5491-5/99, 1999

[51]. Wilson P., Ross, J. Brown A. and Rushton A., "Multiple Domain Behavioral Modeling Using VHDL-AMS", IEEE 2004, 0-7803-8251-X/04, Proceedings of the 2004 International Symposium on Circuits and Systems ISCAS '04, Vol. 5, pp. 23-26 May 2004

[52]. Trofimov M. and Mosin S., "The Realization of Algorithmic Description on VHDL-AMS", Proceedings on Modern Problems of Radio Engineering, Telecommunications and Computer Science, 2004, TCSET' 2004

[53]. Nikitin P., Shi R. and Wan B., "Modeling Partial Differential Equations in VHDL-AMS", IEEE 2003, DOI: 0-7803-8182-3/03, Proceedings of IEEE International SOC Conference, pp. 345 - 348, 17-20 Sept. 2003

[54]. Rhew J., Ren Z. and Lundstrom M. "Benchmarking Macroscopic Transport Models for Nanotransistor TCAD", Journal of Computational Electronics, Vol. 1, pp.385 - 388, KluwerAcademicPublishers, 2002

[55]. Rengel R., Pardo D. and Martin M., "Towards the Nano Scale: Influence of Scaling on the Electronic Transport and Small-Signal Behavior of MOSFETs", Nanotechnology, Vol. 15, pp. 276 - 282, 2004.

[56]. Kumar S., Kumar A. and Choudhury S., "Soft Computing Tools for the Simulation of Efficient Nanodevice Models", IEEE, DOI: 0-7803-7749-4/03, 2003

[57]. Shear L., Singleton C., Haertel G., Mitchell K. and Zaner S., "Measuring Learning: A Guidebook for Gathering and Interpreting Evidence", Center for Technology in Learning, SRI International, 2007

[58]. NANOMAT Project No., ETIS-CT-2003-508695, Deliverable 3.1, European State of Art Report, 2003

APPENDICES

Appendix A: Matlab Code of Molecular Transistor Model

```
% MATLAB CODE OF MOLECULAR TRANSISTOR MODEL
clear all

% Constants definition
Hbar = 1.055e-34;
q = 1.602e-19;
I0 = q*q/hbar;                          % maximum conductance

% Parameters definition
U0 = 0.25;                              % charging energy in eV
kT = 0.025;                             % energy in eV %at room temp
T=300K
mu = 0;
ep = 0.2;                               % in eV
N0 = 0;
Alphag = 0;                             % molecular coupling
Alphad = 0.5;                           % molecular coupling
%Energy grid
NE = 501;
E = linspace(-1,1,NE);
dE = E(2) - E(1);
```

Appendix A: (Continued)

```
g2 = 0.005*ones(1,NE);                          % gamma 2
g1 = g2;                                        % gamma 1
g = g1 + g2;                                    % absolute broadening factor
%Bias
IV = 101;
VV = linspace(-0.8,0.8,IV);                     % applied voltage
for iV = 1:IV
  Vg = 0;                                       % gate voltage
  Vd = VV(iV);
  Vg = VV(iV);
  mu1 = mu;
  mu2 = mu1 - Vd;
  UL = -(alphag*Vg) - (alphad*Vd);
  U = 0;                                        % self-consistent field
  dU = 1;
  while dU>1e-6
    f1 = 1./(1 + exp((E - mu1)./kT));
    f2 = 1./(1 + exp((E - mu2)./kT));
    D = g./(2*pi))./(((E – ep – UL - U).^2)+((g./2).^2));
    D = D./(dE*sum(D));
    N(iV) = dE*2*sum(D.*((f1.*g1./g) + (f2.*g2./g)));
```

Appendix A: (Continued)

```
    Unew = U0*(N(iV) - N0);
    dU = abs(U - Unew);
    U = U + 0.1*(Unew - U);
  end
  I(iV) = dE*2*I0*(sum(D.*(f1 - f2).*g1.*g2./g));
  end
hold on
h = plot(VV,I);
grid on
```

Appendix B: VHDL-AMS Code of Molecular Transistor Model

The VHDL-AMS coding for the Molecular Transistor Model is presented in this appendix. A complete entity-architecture pair for the molecular transistor is presented.

```
-- VHDL-AMS MODEL OF A MOLECULAR TRANSISTOR
-- University of South Florida
-- College of Engineering
-- "Nano Scale Based Model Development for Mems To Nems Migration", Ph.D. Thesis
-- in Electrical Engineering
-- Copyright Andres Lombo-Carrasquilla

-- Model name: Molecular Transistor Level-0
-- This is a discrete model of a molecular transistor
-- The molecular resistance is associated with the interface between the
-- narrow wire and the wide contacts
-- Ballistic transport model:
        -- No scattering is assumed
        -- Contact 1 is grounded
        -- Low bias is assumed
        -- Minimum broadening of molecular energy levels is assumed
        -- This code is optimized to be simulated with Hamster tool by Ansoft
        -- Corporation
        -- Constant Fermi level assumed
```

Appendix B: (Continued)

```
-- The output voltage obtained represents only the positive part
LIBRARY IEEE;
USE IEEE.MATH_REAL.ALL;
-- entity definition
ENTITY MOLCtoy IS
        QUANTITY mu1: REAL;
        QUANTITY mu2: REAL;
        QUANTITY eVg: REAL;
        QUANTITY USC: REAL;
        QUANTITY N: REAL;
        QUANTITY ep: REAL;
        QUANTITY N0: REAL;
        QUANTITY f1: REAL;
        QUANTITY f2: REAL;
        QUANTITY N1: REAL;
        QUANTITY N2: REAL;
        QUANTITY IL: REAL;
        QUANTITY IR: REAL;
        QUANTITY I: REAL;
        QUANTITY G: REAL;
        CONSTANT eta: REAL := 0.5; -- charging coefficient, which could be 0<eta<1
```

Appendix B: (Continued)

```
    CONSTANT ep0: REAL := -5.5;          -- <electronvolts> molecular
                                         -- potential energy level in
                                         -- equlibrium
    CONSTANT Ef: REAL := -5.0;           -- < electronvolts > Fermi level
    CONSTANT hbar: REAL := 1.1356e-15;   --Planck constant
    CONSTANT g1: REAL := 0.1;            -- Broadening coefficient gamma1
    CONSTANT g2: REAL := 0.1;            -- Broadening coefficient gamma2
    CONSTANT U: REAL := 0.001;           -- charging constant
    CONSTANT kT: REAL := 0.025;          -- Boltzman constant at room
                                         -- temperature
    CONSTANT q: REAL := 1.602e-19;       -- electron charge
END ENTITY MOLCtoy;
ARCHITECTURE Level-0 OF MOLCtoy IS
BEGIN
    N0 == 2.0/(1.0 + exp((ep0 + Ef)/kT));  -- electrons in equilibrium state
    eVg = = now;                           -- (electronvolts) applied energy level
    USC = = eta * evg;                     -- charging voltage effect over the
                                           -- molecule
    mu1 = = Ef-(1.0 - eta) * eVg;          -- first contact energy level
    mu2 = = Ef + (eta * eVg);              -- second contact energy level
    ep = = ep0 + USC;                      -- molecular energy level
```

Appendix B: (Continued)

```
        f1 = = 1.0/(1.0 + exp((ep - mu1)/kT));    -- Fermi level at the first contact
        f2 = = 1.0/(1.0 + exp((ep - mu2)/kT));    -- Fermi level at the second contact
        N1 = = 2.0 * f1;                          -- number of charge carriers at first contact
        N2 = = 2.0 * f2;                          -- number of charge carriers at second
                                                  -- contact
        IL = = (2.0 * g1 * q/hbar) * (N1 - N);    -- right current
        IR = = (2.0 * g2 * q/hbar) * (N - N2);    -- left current
        G = = I'dot;                              -- conductance equilibrium conditions
                                                  -- verification routine
        IF (IL = IR) USE
          N = = (2.0 * ((g1 * f1) + (g2 * f2))/(g1 + g2));
          I = = (2.0 * q/hbar) * (g1 * g2/(g1 + g2)) * (f1 - f2);
        ELSE
          N = = ((USC/U) + 2.0 * N0) - N;         -- charging condition
          I = = IR - IL;
        END USE;
END ARCHITECTURE Level-0;
--END OF VHDL-AMS MODEL OF A MOLECULAR TRANSISTOR
```

Appendix C: XML Code of Molecular Transistor Model


```
<?xml version="1.0" encoding="iso-8859-1" ?>
- <lom xmlns="http://www.imsglobal.org/xsd/imsmd_rootv1p2p1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.imsglobal.org/xsd/imsmd_rootv1p2p1
imsmd_rootv1p2p1.xsd">
+ <general>
- <title>
  <langstring>Molecular Transistor Model</langstring>
  </title>
- <catalogentry>
  <catalog>Predeterminado</catalog>
- <entry>
  <langstring>Entrada de catálogo predeterminada</langstring>
  </entry>
  </catalogentry>
  <language>sp</language>
- <description>
  <langstring>Predeterminado</langstring>
  <langstring>Predeterminado</langstring>
  </description>
- <keyword>
  <langstring>Palabra clave predeterminada</langstring>
```

Appendix C: (Continued)

</keyword>

</general>

- <lifecycle>

- <version>

<langstring>**1.1**</langstring>

</version>

- <status>

- <source>

<langstring xml:lang="x-none">**LOMv1.0**</langstring>

</source>

- <value>

<langstring xml:lang="x-none">**Final**</langstring>

</value>

</status>

</lifecycle>

- <metametadata>

<metadatascheme>**ADL SCORM 1.2**</metadatascheme>

</metametadata>

- <technical>

<format>**text/html**</format>

<location>**molctoy.htm**</location>

Appendix C: (Continued)

</technical>

- <educational>

- <learningresourcetype>

- <source>

<langstring xml:lang="x-none">**LOMv1.0**</langstring>

</source>

- <value>

<langstring xml:lang="x-none">**Simulation**</langstring>

</value>

</learningresourcetype>

</educational>

- <rights>

- <cost>

- <source>

<langstring xml:lang="x-none">**LOMv1.0**</langstring>

</source>

- <value>

<langstring xml:lang="x-none">**no**</langstring>

</value>

</cost>

- <copyrightandotherrestrictions>

Appendix C: (Continued)

- <source>

<langstring xml:lang="x-none">**LOMv1.0**</langstring>

</source>

- <value>

<langstring xml:lang="x-none">**no**</langstring>

</value>

</copyrightandotherrestrictions>

- <description>

<langstring>**Model Description**</langstring>

</description>

</rights>

- <classification>

- <purpose>

- <source>

<langstring xml:lang="x-none">**LOMv1.0**</langstring>

</source>

- <value>

<langstring xml:lang="x-none">**Educational Objective**</langstring>

</value>

</purpose>

- <description>

Appendix C: (Continued)

<langstring>**Molecular transistor model**</langstring>

</description>

- <keyword>

<langstring>**Simplified Non-equilibrium Green function**</langstring>

</keyword>

</classification>

</lom>

ABOUT THE AUTHOR

Andres Lombo Carrasquilla received a Bachelor's Degree in Electrical Engineering from Universidad Distrital Francisco Jose de Caldas in 1994 and a M.S. in Electrical Engineering from Universidad de Los Andes in 1998. Andres started as a faculty member at the College of Engineering at Universidad Distrital in 1997, where he joined the Optoelectronics and Microelectronics Research Group at this university in 1994. Andres is currently the Principal Investigator of the group.

Andres was supported by Universidad Distrital during his Ph.D. program at the University of South Florida, in Tampa, Florida, where he was involved in research activities in MEMS and nanotechnology related areas. Andres has made several paper presentations in Colombia, Latin America, and other international meetings related to the nanotechnology area.