SIMON—A Simulator for Single-Electron Tunnel Devices and Circuits

Christoph Wasshuber, Hans Kosina, *Member, IEEE*, and Siegfried Selberherr, *Fellow, IEEE*

***Abstract*—SIMON is a single-electron tunnel device and circuit simulator that is based on a Monte Carlo method. It allows transient and stationary simulation of arbitrary circuits consisting of tunnel junctions, capacitors, and voltage sources of three kinds: constant, piecewise linearly time dependent, and voltage controlled. Cotunneling can be simulated either with a plain Monte Carlo method or with a combination of the Monte Carlo and master equation approach. A graphic user interface allows the quick and easy design of circuits with single-electron tunnel devices. Furthermore, as an example of the usage of SIMON, we discuss the essential problem of random background charge and present possible solutions.**

***Index Terms*— Cotunneling, Coulomb blockade, master equation, Monte Carlo technique, random background charge, single electron, tunneling.**

I. Introduction

THE first observations of the Coulomb blockade and thus single electronics was made by Gorter in 1951 [1]. He studied granular thin-film structures, and discovered a low-voltage suppression of the dc conductivity. He already identified the reason for the conductivity suppression due to charging of grains with single electrons. In the past decade, knowledge and understanding of single electronics grew, due to the available production processes for small structures. It was then for the first time possible to study a single grain or small tunnel junction, and thus one did not have to deal with averaged characteristics of granular films consisting of thousands of grains. A theory explaining quantitatively charging effects and the Coulomb blockade appeared at the end of the past decade [2]–[4]. Today, we see more and more publications dealing with the circuit level of single electronics [5]–[7], and the first applications have been developed [8], [9].

To support circuit level design, some simulators have been implemented or are under construction. Among them are SENECA by Fonseca *et al.* [10] at Stony Brook State University of New York, MOSES by Chen *et al.* [11] at Stony Brook State University of New York, Ancona at Naval Research Laboratory, Washington [7], Kirihara *et al.* [12] at Osaka University, Fukui *et al.* [6] and Amakawa at Tokyo University, Kuwamura *et al.* [13] at Osaka University, Kautz at NIST Colorado and Ferry at Arizona State University, Masu and Tsubouchi at Tohoku University, Amemiya at Hokkaido University, and Simon [23] at TU-Vienna.

Manuscript received December 18, 1996. This paper was recommended by Associate Editor Z. Yu.

The authors are with the Institute for Microelectronics, TU-Vienna, A-1040 Wien, Austria.

Publisher Item Identifier S 0278-0070(97)09012-X.

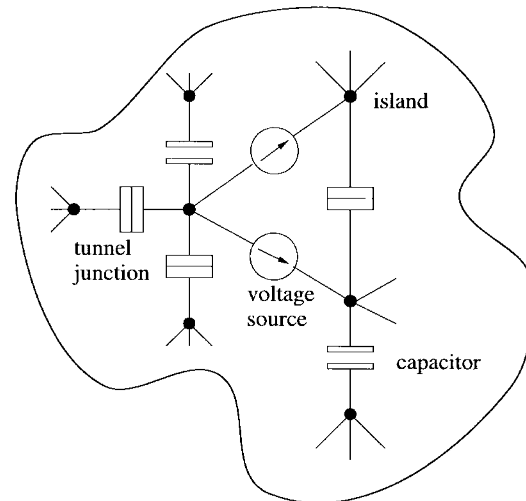

Fig. 1. Single-electron tunnel circuit consisting of tunnel junctions, capacitors, and voltage sources.

The basic working principle of single electronics is that one needs Coulomb energy E_c to charge a grain (island) with an electron

$$E_c = \frac{e^2}{2C_\Sigma} > kT \tag{1}$$

where C_Σ is the overall capacitance of the grain. If this Coulomb energy is larger than the available thermal energy, one can control the movement of electrons by controlling the available energy supplied by voltage sources.

II. Single-Electron Tunnel Circuits

We are contemplating single-electron tunnel (SET) circuits consisting of islands that are arbitrarily connected with tunnel junctions and capacitors and which are driven by voltage sources (see Fig. 1). Voltage sources are considered to be ideal; thus, their internal resistance is zero. Electrons tunnel independently from island to island through tunnel junctions, changing the charge distribution in the circuit. Electron states are localized on islands. To assure this, all tunnel resistances must be larger than the fundamental resistance

$$R_T > R_q = \frac{h}{e^2} \doteq 25\,813\ \Omega. \tag{2}$$

III. Two Different Simulation Methods: Monte Carlo and Master Equation

Besides some analytical solutions for very simple or symmetric problems [3], [14], there are currently two simulation approaches used for SET circuits. One is based on a Monte

0278–0070/97$10.00 © 1997 IEEE

Carlo method, and the other on a master equation. The Monte Carlo approach starts with all possible tunnel events, calculates their probabilities, and chooses one of the possible events randomly, weighted according to their probabilities (see Section V). This is done many times to simulate the transport of electrons through the network. Tunnel events are considered to be independent and exponentially distributed.

The master equation, on the other hand, is a description for the underlying Markov process [15] of electrons tunneling from island to island, and thus the circuit occupies different states. In order to do this, one needs the set of all possible states of the circuit. A state is defined by the set of voltages of voltage sources and the charge distribution in the circuit. Neglecting induced background charge, which can be nonintegral, every island has an integer number of elementary charges stored on it. Therefore, in the case of constant voltage sources, the set of states is a countable infinite manifold. In order to solve the master equation, only a finite number of states can be considered. This finite set of states together with the transition probabilities of every state to any other state in the set lets one solve the master equation. The result of the solution to the master equation is the state probabilities.

IV. Advantages and Disadvantages of a Monte Carlo Approach

We think a Monte Carlo method is superior to other approaches because of the following advantages.

- It gives better transient and dynamic characteristics of SET circuits because it models the underlying microscopic physics in a very direct manner. In real SET circuits, electrons tunnel from island to island as simulated by the Monte Carlo method. The master equation deals with average probabilities and transition rates, which paint a more macroscopic picture.
- It is not required to find the relevant states before one can start with the actual simulation as in the case of a master equation. For an unknown circuit, no *a priori* knowledge about relevant states is available. Thus, in the case of the master equation approach, one needs to include many more than relevant states to correctly simulate the circuit. Considering more states means longer simulation time and decreased numerical stability.
- It is easy to trade accuracy with simulation time, and therefore one can quickly achieve approximate results of very large circuits. To speed up the solution of the master equation, one can only limit the considered states. Because of the lack of knowledge of which states are dominating the behavior of the circuit, a reduction of states is very often impossible.

Nevertheless, there is one major disadvantage of the Monte Carlo method. When it comes to simulating cotunneling, a plain Monte Carlo approach has its limitations. Cotunneling is a very rare process which is difficult to resolve by a Monte Carlo method. It demands very long simulation times. We tackled this problem by implementing a new algorithm that combines the Monte Carlo and master equation methods (see Section VI).

V. Simulation Algorithm

We extract the capacitance matrix from the circuit description

$$\begin{pmatrix} q \\ Q \end{pmatrix} = \begin{pmatrix} C_a & C_b \\ C_b & C_d \end{pmatrix} \cdot \begin{pmatrix} V \\ v \end{pmatrix} \tag{3}$$

where C is the capacitance matrix, V is the known part of the node voltages, v is the unknown part of the node voltages, Q is the known part of the node charges, and q is the unknown part of the node charges. For a particular node, either its charge or its voltage is known. Hence, only a part of the node voltages and a part of the node charges is known. One has to partially invert the capacitance matrix to derive a relation between the known and unknown data

$$\begin{pmatrix} q \\ v \end{pmatrix} = \begin{pmatrix} -C_a + C_b \cdot C_d^{-1} \cdot C_b^T & -C_b \cdot C_d^{-1} \\ -C_d^{-1} \cdot C_b^T & C_d^{-1} \end{pmatrix} \cdot \begin{pmatrix} V \\ Q \end{pmatrix}. \tag{4}$$

Since the capacitance matrix is symmetric, only one half of it has to be stored, and one can use efficient algorithms for the inversion of C_d. We implemented a Cholesky factorization and an inversion of the resulting triangular matrix. Finally, the inverted triangular matrix is multiplied by its transpose. It is a good choice to explicitly invert C_d because (4) has to be evaluated very often, and usually the dimensions of C_d are small, often below 100.

To simulate the tunneling of electrons from island to island, one has to determine the rates of all possible tunnel events. The normal tunnel rate for one tunnel junction is given by [14]

$$\Gamma = \frac{\Delta F}{e^2 R_T \left(1 - \exp\left(-\frac{\Delta F}{kT}\right)\right)} \tag{5}$$

where ΔF is the change in Helmholtz's free energy, R_T is the tunnel resistance, and kT is the thermal energy. Helmholtz's free energy is defined by the difference in electrostatic energy stored in the circuit and the work done by the voltage sources

$$F = U - W = \tfrac{1}{2}(q, v) \cdot \begin{pmatrix} V \\ Q \end{pmatrix} - W \tag{6}$$

with

$$W = \sum_n \int V_n(t) i_n(t)\, dt \tag{7}$$

where U is the electrostatic energy, W the work done by the voltage sources, $V_n(t)$ the voltage of the nth voltage source, and $i_n(t)$ is the current through the nth voltage source, respectively.

Once all tunnel rates are known, the actually occurring event is determined with a Monte Carlo method. Tunnel events are considered as independent and exponentially distributed processes. Combining a Monte Carlo method with an exponential distribution leads to an expression for a concrete duration to the next tunnel event in a particular junction [16], [12]

$$\Delta t = -\frac{\ln(r)}{\Gamma} \tag{8}$$

where r is an evenly distributed random number from the interval [0, 1] and Γ is the tunnel rate. The event, among

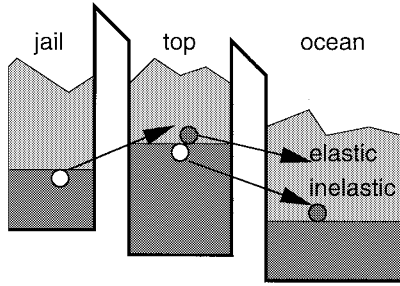

Fig. 2. Cotunneling via a virtual intermediate state.

all possible ones, with the shortest duration is taken. After a tunnel event, node charges and node voltages generally change. Consequently, the free energy changes too, and one has to calculate all possible rates again. This loop is performed many times to simulate the transport of electrons through the network.

VI. Problems when Simulating Cotunneling

The cotunnel effect is a quantum mechanical effect which allows electrons to tunnel via an intermediate virtual state, where normal tunneling would be impossible, or due to missing thermal energy very unlikely (see Fig. 2). An electron cannot tunnel directly from jail to ocean. Also, a normal tunneling from jail to top is impossible because of missing thermal energy. Nevertheless, an electron will escape to ocean via an intermediate virtual state. One could also picture this process in the following simplified way. An electron starting at jail overcomes the energy difference to top for a very short time allowed by Heisenberg's uncertainty principle. If a different electron from top tunnels in the same very short time to ocean, then overall, an electron escaped from jail to ocean. This process is called the inelastic cotunnel effect. There is another usually negligible process, which is called elastic cotunneling, where the same electron that tunneled first from jail to top tunnels further to ocean. Cotunneling is a major source of errors in single-electron logic devices.

The rate of second-order inelastic cotunneling can be written as [17], [18]

$$\Gamma = \frac{2\pi}{\hbar} T_{(1)}^2 T_{(2)}^2 \left(\frac{1}{\Delta E_1} + \frac{1}{\Delta E_2} \right)^2 \delta(E_i - E_f) \tag{9}$$

where $T_{(i)}$ represents the tunneling amplitude through barrier i, ΔE_i is the change in energy of the system for a tunnel event through barrier i, and E_i and E_f are the initial and final energies of the system.

A cotunnel event has a very rare occurrence compared to a normal tunnel event because it scales with $1/R_T^{(n-1)}$, with n as the order of cotunneling. Thus, it poses a formidable problem for a Monte Carlo (MC)-based simulator to resolve such rare events. Standard variance reducing techniques [19] do not work because in a typical MC simulation run, rare states are very likely not even visited once. And therefore, the trajectory splitting or multiplying schemes will not trigger.

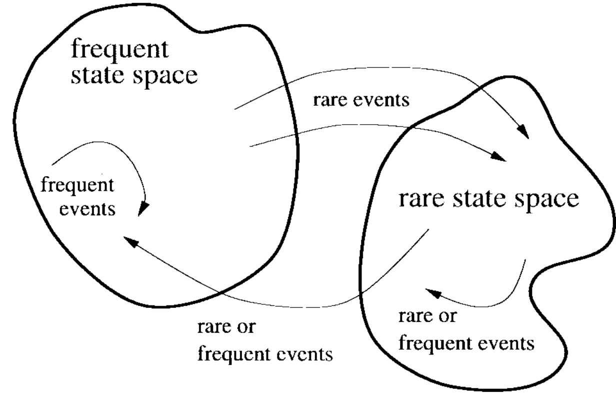

Fig. 3. Partition of the state space into a frequent state domain and a rare state domain.

The usual approach to overcome this problem is to use the master equation (ME) method. The master equation is given by

$$\frac{dP_i}{dt} = \sum_j (\Gamma_{ji} P_j - \Gamma_{ij} P_i) \tag{10}$$

where Γ_{ij} is the tunnel rate between states i and j, and P_i is the occupation probability of state i. Although the ME gives theoretically accurate results, it has many other impracticabilities that limit its accuracy and usability. The starting point of the ME is the set of all relevant states a circuit will occupy during operation. There is no straightforward way to obtain this set. In order to achieve the desired simulation goal, one has to include many more states than would be relevant, which results in extremely long simulation times and sometimes bad numerical stability. Fonseca *et al.* [10] explain an adaptive algorithm which alleviates this problem.

We have combined advantages of the MC and ME methods. Consider all possible states as divided into two subspaces, the frequent state space and the rare state space (see Fig. 3). The MC kernel simulates only the frequent state space. The convenient part is that the same MC simulation already gives the occupation probabilities of frequent states that would result from a ME calculation. The occupation probability P_i is calculated as the ratio of time T_i spent in state i to the time T_Σ spent in all states

$$T_\Sigma = \sum_i T_i \tag{11}$$

$$P_i = \frac{T_i}{T_\Sigma}. \tag{12}$$

P_i is the solution to the stationary ME for the frequent states. Instead of waiting for the MC simulator to step into the rare state space, which would result in impractically long simulation times, we directly calculate the contribution of events leading to rare states by stepping through the event tree starting at frequent states (see Fig. 4). The essential assumption is that the rare states cause only a small perturbation to the frequent state probabilities

$$\sum_j T_{j,\mathrm{rare}} \ll T_\Sigma \tag{13}$$

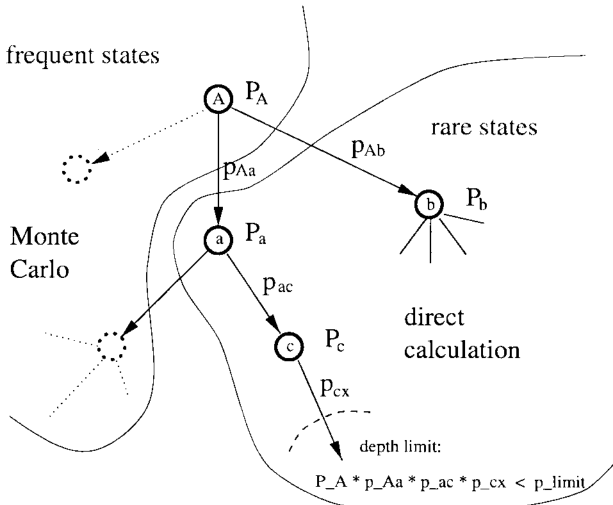

Fig. 4. Direct calculation of the contribution of rare events and states by stepping through the event tree.

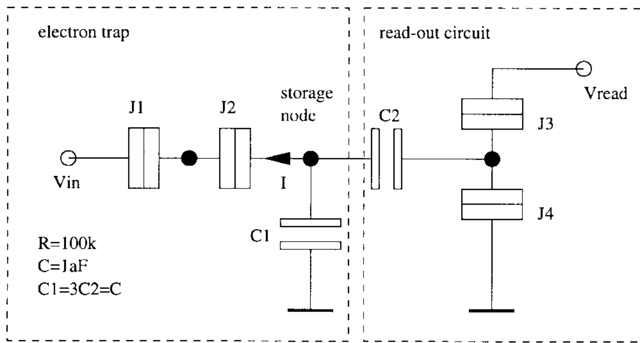

Fig. 5. Electron trap with read-out circuit.

where $T_{j,\mathrm{rare}}$ is the directly calculated time spent in the rare state j

$$T_{j,\mathrm{rare}} = \frac{\sum_i (T_i \Gamma_{ij})}{\sum_k \Gamma_{jk}}. \tag{14}$$

$T_i\Gamma_{ij}$ is the number of times the rare state would be visited from state i, and $1/\sum_k \Gamma_{jk}$ is the average time spent in state j for one visit. The occupation probability for a rare state is

$$P_{j,\mathrm{rare}} = \frac{T_{j,\mathrm{rare}}}{T_\Sigma} = \frac{\sum_i (T_i \Gamma_{ij})}{T_\Sigma \sum_k \Gamma_{jk}}. \tag{15}$$

The algorithm follows all possible events (rare and frequent) starting at frequent states. If a frequent state is encountered, the algorithm terminates the branch and continues with another one because all relevant information is already included in the known state probability. Once the probability of a state is lower than a predefined limit, no further descent from this state is made (see Fig. 4).

As an example of the benefits of this new simulation algorithm, we simulate an electron trap with a read-out circuit as shown in Fig. 5. A voltage pulse on V_{in} transports one

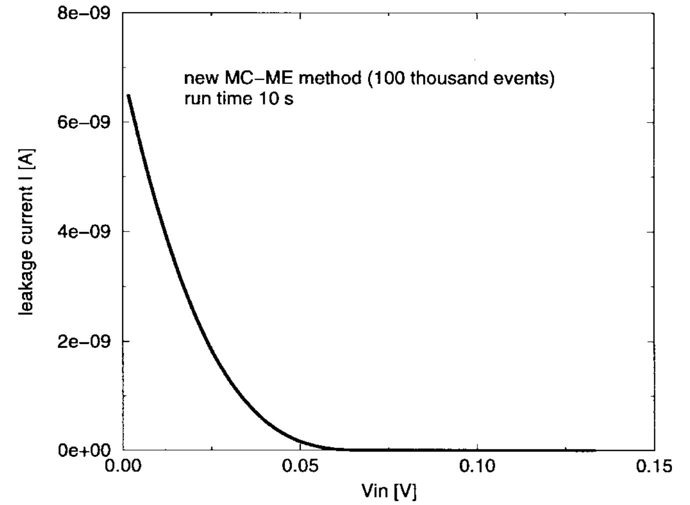

Fig. 6. Leakage current versus input voltage, simulated with our new MC–ME algorithm.

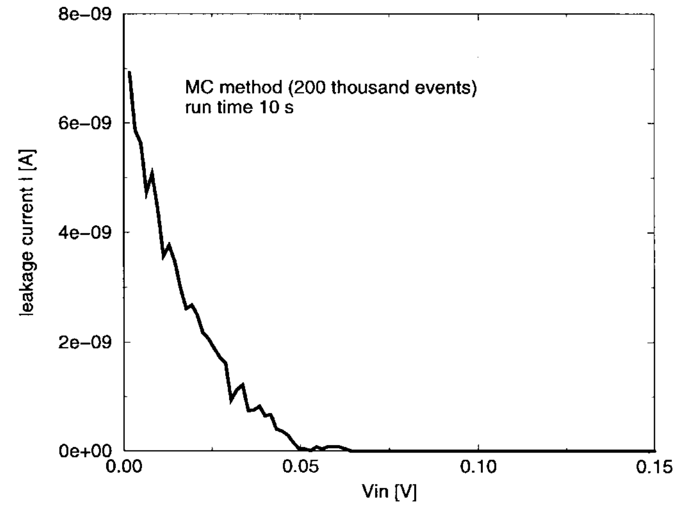

Fig. 7. Leakage current versus input voltage, simulated with a plain MC algorithm.

elementary charge onto the "storage node." This stored charge influences via C_2 the read-out circuit which is a single-electron transistor. The current through the read-out transistor depends on the charge which is stored in the trap. We are interested in the leakage current I or, which is equivalent, the lifetime of a stored electron. The lifetime t and leakage current are related by $t = e/I$, where e is the charge of an electron. The leakage current is partly due to thermal fluctuations and partly due to cotunneling. Our interest is in which simulation algorithm gives better results. Hence, we simulated the same circuit with a plain MC method and with our new MC–ME method. Figs. 6 and 7 show the leakage current as a function of V_{in}. It can be clearly seen from Fig. 6 that our new MC–ME algorithm produces in the same simulation time a much more accurate result than the plain MC method (Fig. 7). One would have to simulate many more tunnel events with a plain MC method to achieve a similar accuracy. The reason for this is that the read-out circuit consumes many tunnel events which do not contribute to the accuracy of the leakage current. In

bigger and more complicated circuits, this problem can be much more pronounced, and render a plain MC method useless for cotunneling simulations.

VII. Making the Simulator Fast

All SET circuits have an infinite state space, except some very simple circuits without an island. Nevertheless, very often, only a limited number of states determine the behavior of a circuit. This characteristic makes it possible to considerably speed up an MC-based SET simulator. We calculate a tunnel rate for every possible tunnel event. Since only a limited number of states dominate the circuit, one will repeatedly step into states that have been visited before. Instead of calculating all possible tunnel rates again and again, one can store calculated tunnel rates. Some form of hash table is a good choice because it allows a quick and almost direct access of already stored states. The hash table very often yields an acceleration of a factor between five and ten.

The CPU time of a simulation mainly depends on the number of tunnel junctions j and the number of nodes n a circuit has. For normal tunneling, $2j$ tunnel rates have to be calculated for every simulated tunnel event. For cotunneling with the order k, one needs to calculate additionally $(2j)^k$ tunnel rates. Thus, altogether $\sum_{i=1}^{k}(2j)^i$ tunnel rates per event have to be calculated. Matrix operations to calculate all node voltages and node charges scale quadratically with the node number. Hence, one could approximate the CPU time with the following expression:

$$t_{\mathrm{CPU}} = c_0 + j\big(c_1 + c_2 n + c_3 n^2\big). \tag{16}$$

The acceleration algorithm usually drastically reduces the CPU time. The CPU time is, for an accelerated simulation, also a function of temperature and bias voltages. Increasing temperature and increasing bias voltages increase the CPU time. The reason for this is that with increasing temperature and increasing bias voltages, the number of states the circuit occupies during a simulation increases too.This means that the acceleration algorithm loses its effectiveness more and more.

Fig. 8 shows typical CPU times versus the number of tunnel junctions for simulations without cotunneling.

VIII. Implementation Issues

SIMON does not impose any *a priori* limitations such as maximum number of islands, maximum number of tunnel junctions, or that voltage sources have to be grounded. It can deal with constant, piecewise linearly time-dependent and voltage-controlled voltage sources.

SIMON features a graphical user interface and a graphical circuit editor that is implemented in tcl/tk [20].

The graphical circuit editor allows a drag-and-drop assembly of SET circuits. Parameters can be changed interactively, and simulation results can be looked at in graphical form. Also, all simulation parameters, such as simulation mode, cotunnel order, and ambient temperature, are modifiable. For screen shots of SIMON, see Fig. 9.

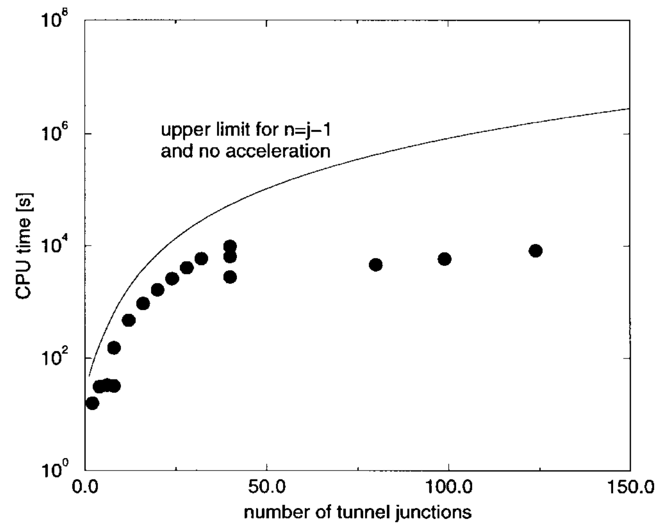

Fig. 8. Typical CPU times for 10^6 tunnel events versus number of tunnel junctions for a PC with 133-MHz Pentium processor and 48-Mbyte RAM. The solid line shows the upper limit of the CPU time for junctions arranged in one long line ($n = j - 1$) with the acceleration algorithm switched off. (The coefficients of (16) for this hardware are: $c_0 = 21$, $c_1 = 25.7$, $c_2 = 1.5$, $c_3 = 0.83$.)

IX. Example: Random Background Charge

To give an example on the study of SET devices, we address the problem of random background charge. Impurities and trapped electrons in the substrate induce charges on islands that usually destroy the correct device function. With today's processing techniques, one is not able to control the purity of materials enough to meet conditions suitable for SET device production. A single impurity atom located in an unfortunate position most likely changes the desired device behavior. Take, for example, the SET transistor (Fig. 10) at $T = 10$ K, zero gate voltage V_g, and background charge $Q_b = 0, 0.25, 0.5e$ (Fig. 11). One observes the Coulomb blockade in the I–V characteristic. The size of the blockade is

$$V_{cb} = \frac{e}{C_\Sigma} - \frac{2Q_b}{C_\Sigma}, \qquad \text{for } Q_b = [0,\ 0.5\ e] \tag{17}$$

where $C_\Sigma = C_1 + C_2 + C_g$ denotes the capacitance of the island. If we consider a background charge, the Coulomb blockade, depending on the amount of background charge, decreases and vanishes completely for $Q_b = 0.5$ e (see Fig. 11 where $R_\Sigma = R_1 + R_2$). Since one has no *a priori* knowledge of the value of the background charge, one has to consider the worst case, in which the Coulomb blockade vanishes completely, thus destroying SET device behavior.

In the case of the electron pump which is used for metrology [8], the background charge can be compensated manually. Unfortunately, impurities and trapped electrons migrate on a time scale of hours. Therefore, one has to adjust the background charges about every hour. This is, of course, no solution for large-scale devices.

As one can imagine, the random background charge problem is crucial. It will decide the commercial usability of SET devices.

Likharev and Korotkov [21] proposed a possible solution for memory cells. They used in their design the current oscillations

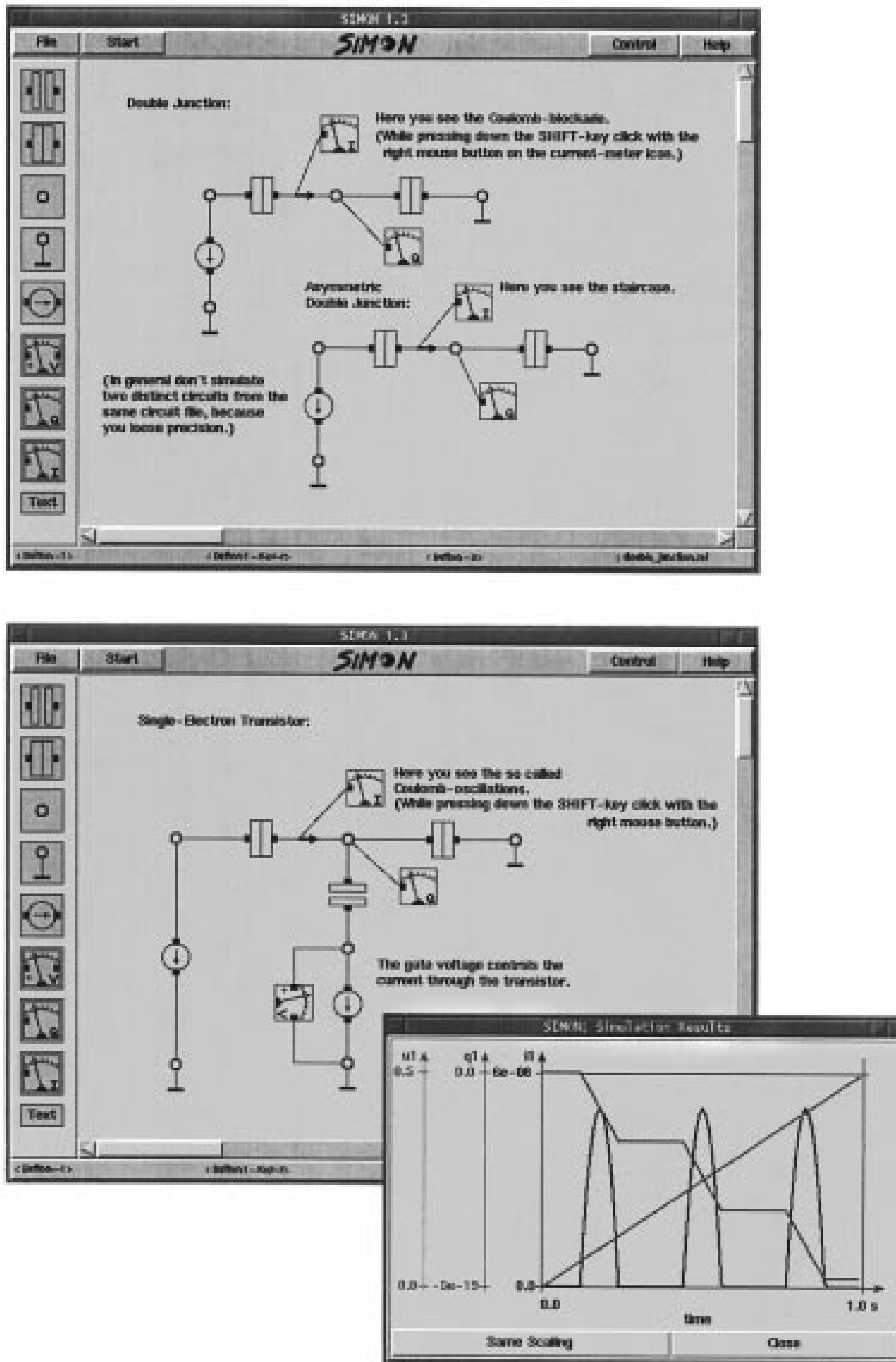

Fig. 9. Screen shots of SIMON.

of a single-electron transistor to determine the presence or absence of charge on a floating gate. The frequency and amplitude of these current oscillations are independent of random background charge. Another very promising solution is based on an array of islands (instead of only one island) for a SET transistor. Such structures show a very good immunity against random background charge. Connections of islands are not crucial, which suggests easy process technologies for granular films. We simulated a hexagonal mesh of tunnel junctions consisting of 50 islands (five times ten islands) connected by 124 junctions. This is what we call a multi-island SET transistor (see Fig. 12). To model the different sizes of grains, we randomized all capacitances and tunnel resistances by $\pm 10\%$. A comparison between the multi-island

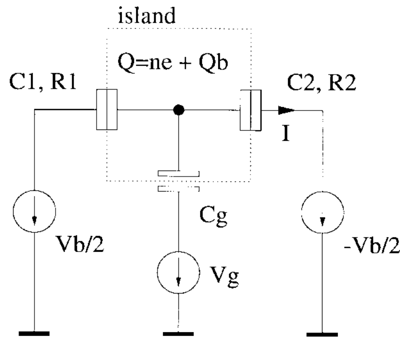

Fig. 10. Circuit diagram of a SET transistor.

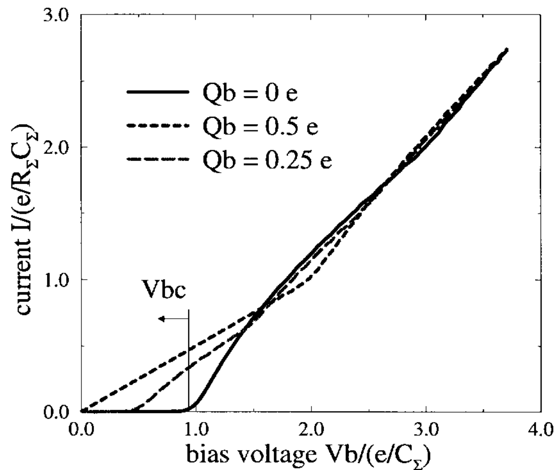

Fig. 11. I–V characteristic of a SET transistor for three different background charges. The Coulomb energy decreases with increasing Q_b. For $Q_b = 0.5e$, the Coulomb blockade vanishes completely.

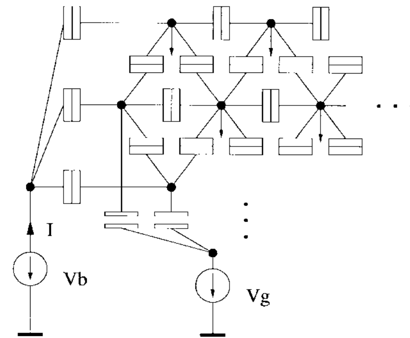

Fig. 12. Multi-island SET transistor. Every island is connected with a capacitor to the gate voltage and tunnel junctions to its nearest neighbors.

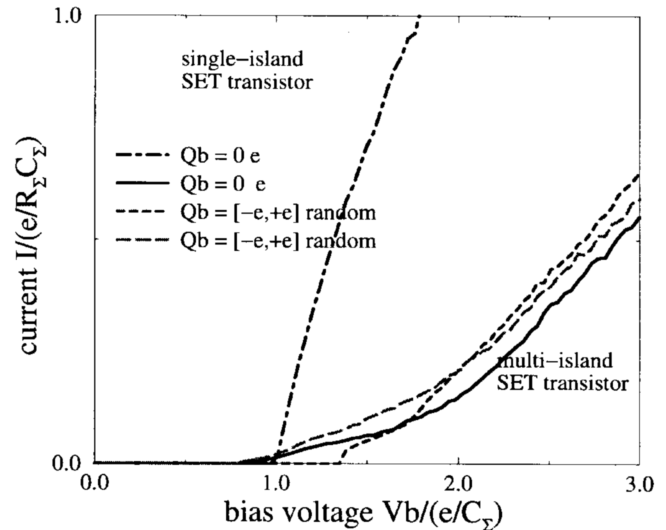

Fig. 13. Comparison of I–V characteristics of single and multi-island SET transistor with $V_g = 0$.

SET transistor and the single-island SET transistor is shown in Fig. 13. The effect of the Coulomb blockade is essentially the same; only the magnitude of the current is larger for the single-island SET transistor because the multi-island SET transistor has more resistances in series than in parallel, which reduces the current. Interesting is that, in spite of having random background charge on all islands in the range $[+e, -e]$, the Coulomb blockade does not change much. In one case, the Coulomb blockade even increased. This is a clear sign that the multi-island SET transistor is immune to randomly induced background charge. We performed similar simulations with a nonsymmetric mesh, which showed the same behavior; thus, actual connections of islands are not crucial.

Chen and Ahmed [22] produced such structures in metal with a grain size ~2 nm, and Yano *et al.* [9] used this approach for their 64-bit SET memory chip, where they deposited an undoped polysilicon batch with grains ~3 nm in size. Their batch consisted of about 2000 islands. Our simulations show that this batch could be easily reduced by a factor 10 or even 100 without destroying the immunity against random background charge. A further advantage is that at such small grain sizes, the spatial confinement of electrons results in an increased energy level spacing. Thus, the Coulomb energy E_c to add one electron is larger than a classical calculation would give, which allows higher operation temperatures [see (1)]. It is generally true for SET devices that downscaling improves their characteristics.

X. Conclusion

We presented a multipurpose SET device and circuit simulator capable of simulating transient and static behavior with and without cotunneling. Cotunneling can be accounted for with a plain Monte Carlo method or with a combination of the Monte Carlo method and master equation method. Our simulator is implemented with a graphical user interface and a graphical circuit editor. It is currently used to test new circuits, and to understand the underlying fundamental phenomena and behavior, like Coulomb blockade and SET oscillations.

An early version, SIMON 1.1, is freely available by ftp after registration. For more information, visit http://www.iue.tuwien.ac.at/. For information on the most recent versions, currently SIMON 1.3, visit http://members.magnet.at/catsmeow/index.html.

References

[1] C. J. Gorter, "A possible explanation of the increase of the electrical resistance of thin metal films at low temperatures and small field strengths," *Physica*, vol. 17, pp. 777–780, Aug. 1951.

[2] K. K. Likharev, "Correlated discrete transfer of single electrons in ultrasmall tunnel junctions," *IBM J. Res. Develop.*, vol. 32, pp. 144–158, Jan. 1988.

[3] D. V. Averin and K. K. Likharev, "Single electronics: A correlated transfer of single electrons and Cooper pairs in systems of small tunnel junctions," in *Mesoscopic Phenomena in Solids*, B. L. Altshuler, P. A. Lee, and R. A. Webb, Eds. Amsterdam, Oxford, New York, Tokyo: North-Holland, Elsevier Science B.V., 1991, ch. 6, pp. 173–271.

[4] H. Grabert and M. H. Devoret, *Single Charge Tunneling*. New York: Plenum, 1992.

[5] J. R. Tucker, "Complementary digital logic based on the 'Coulomb blockade,'" *J. Appl. Phys.*, vol. 72, pp. 4399–4413, Nov. 1992.

[6] H. Fukui, M. Fujishima, and K. Hoh, "Simple and stable single-electron logic utilizing tunnel-junction load," *Jpn. J. Appl. Phys.*, vol. 34, pp. 1345–1350, Feb. 1995.

[7] M. G. Ancona, "Design of computationally useful single-electron digital circuits," *J. Appl. Phys.*, vol. 79, pp. 526–539, Jan. 1996.

[8] J. M. Martinis, M. Nahum, and H. D. Jensen, "Metrological accuracy of the electron pump," *Phys. Rev. Lett.*, vol. 72, pp. 904–907, Feb. 1994.

[9] K. Yano, T. Ishii, T. Sano, T. Mine, F. Murai, and K. Seki, "Single-electron-memory integrated circuit for giga-to-tera bit storage," in *IEEE Int. Solid-State Circuits Conf.*, 1996, pp. 266–267.

[10] L. R. C. Fonseca, A. N. Korotkov, K. K. Likharev, and A. A. Odintsov, "A numerical study of the dynamics and statistics of single electron systems," *J. Appl. Phys.*, vol. 78, pp. 3238–3251, Sept. 1995.

[11] R. H. Chen, A. N. Korotkov, and K. K. Likharev, "Single-electron transistor logic," *Appl. Phys. Lett.*, vol. 68, pp. 1954–1956, Apr. 1996.

[12] M. Kirihara, N. Kuwamura, K. Taniguchi, and C. Hamaguchi, "Monte Carlo study of single-electronic devices," in *Ext. Abst. Int. Conf. Solid State Devices Mater.*, Yokohama, Japan, 1994, pp. 328–330.

[13] N. Kuwamura, K. Taniguchi, and C. Hamaguchi, "Simulation of single-electron logic circuits," *Trans. IEICE*, vol. J77-C-II, no. 5, pp. 221–228, 1994.

[14] G.-L. Ingold and Y. V. Nazarov, "Charge tunneling rates in ultrasmall junctions," in *Single Charge Tunneling—Coulomb Blockade Phenomena in Nanostructures*, H. Grabert and M. H. Devoret, Eds. New York and London: Plenum and NATO Scientific Affairs Division, 1992, ch. 2, pp. 21–107.

[15] W. J. Stewart, *Introduction to the Numerical Solution of Markov Chains*. Princeton, NJ: Princeton University Press, 1994.

[16] G. E. Johnson, "Construction of particular random processes," *Proc. IEEE*, vol. 82, pp. 270–285, Feb. 1994.

[17] D. V. Averin and A. A. Odintsov, "Macroscopic quantum tunneling of the electric charge in small tunnel junctions," *Phys. Lett. A*, vol. 140, pp. 251–257, Sept. 1989.

[18] D. V. Averin and Y. V. Nazarov, "Virtual electron diffusion during quantum tunneling of the electric charge," *Phys. Rev. Lett.*, vol. 65, pp. 2446–2449, Nov. 1990.

[19] A. Phillips, Jr. and P. J. Price, "Monte Carlo calculations on hot electron energy tails," *Appl. Phys. Lett.*, vol. 30, pp. 528–530, May 1977.

[20] J. K. Ousterhout, *Tcl and the Tk Toolkit*. Reading, MA: Addison-Wesley, 1994.

[21] K. K. Likharev and A. N. Korotkov, "Analysis of Q_0-independent single-electron systems," in *Int. Workshop Computational Electron.*, 1995, p. 42.

[22] W. Chen and H. Ahmed, "Fabrication and physics of ~2 nm islands for single electron devices," *J. Vac. Sci. Technol. B*, vol. 13, pp. 2883–2887, Nov. 1995.

[23] C. Wasshuber and H. Kosina, "A single-electron device and circuit simulator," *Superlattices and Microstructures*, vol. 21, no. 1, 1997.

Christoph Wasshuber received the Diplomingenieur and the Ph.D. degrees in electrical engineering from the Vienna University of Technology, Austria, in 1993 and 1997, respectively.

He was with the Asada Laboratory, Tokyo University, Japan for 1.5 years, studying single electronics. He then joined the Institut für Mikroelektronik at the Vienna University of Technology. His current interests include modeling and simulation of single-electron devices, quantum effects in semiconductor devices, and computer-aided engineering in VLSI technology in general.

Hans Kosina (S'88–M'91) was born in Haidershofen, Austria, in 1961. He received the Diplomingenieur and the Ph.D. degrees in electrical engineering from the Vienna University of Technology in 1987 and 1992, respectively.

He was with the Institut für flexible Automation for one year, and then joined the Institut für Mikroelektronik at the Vienna University of Technology in 1988. Currently, he is employed as an Assistant Professor, heading the Device Modeling Group. His current interests include the physics and technology of solid-state devices and integrated circuits.

Siegfried Selberherr (M'79–SM'84–F'93) was born in Klosterneuburg, Austria, in 1955. He received the degree of Diplomingenieur in control theory and industrial electronics from the Vienna University of Technology in 1978.

Since that time, he has been with the Institut für Allgemeine Elektrotechnik und Elektronik, previously called the Institut für Physikalische Elektronik, at the Vienna University of Technology. He finished his thesis on two-dimensional MOS transistor modeling in 1981. He has been holding the venia docendi on computer-aided design since 1984, and has been the Head of the Institut für Mikroelektronik since 1988. His current topics are modeling and simulation of problems for microelectronics engineering. He has authored and coauthored more than 200 publications in journals and conference proceedings, and he also wrote a book, *Analysis and Simulation of Semiconductor Devices*.

Dr. Selberherr is a member of the Association for Computing Machinery (1979), the Society of Industrial and Applied Mathematics (1980), and the Verband deutscher Elektrotechniker. He is Editor of *Transactions of the Society for Computer Simulation*, of *Electrosoft* (until 1992), of *Mikroelektronik*, and of the Springer-Verlag book series *Computational Microelectronics*. In 1983, he received the Dr. Ernst Fehrer Award, in 1985 the award of the Nachrichtentechnische Gesellschaft, in 1986 the Dr. Herta Firnberg Staatspreis, and in 1987 the Heinz Zemanek Award.