Scalability Simulations for Nanomemory Systems Integrated on the Molecular Scale

MATTHEW M. ZIEGLER,[a,b] CARL A. PICCONATTO,[a] JAMES C. ELLENBOGEN,[a] ANDRÉ DEHON,[c] DELI WANG,[d] ZHAOHUI ZHONG,[d] AND CHARLES M. LIEBER[d]

[a]*Nanosystems Group, The MITRE Corporation, McLean, Virginia, USA*

[b]*ECE Department, University of Virginia, Charlottesville, Virginia, USA*

[c]*The California Institute of Technology, Pasadena, California, USA*

[d]*Harvard University, Cambridge, Massachusetts, USA*

ABSTRACT: Simulations were performed to assess the prospective performance of a 16 Kbit nanowire-based electronic nanomemory system. Commercial off-the-shelf microcomputer system modeling software was applied to evaluate the operation of an ultra-dense storage array. This array consists of demonstrated experimental non-volatile nanowire diode switches, plus encoder–decoder structures consisting of demonstrated experimental nanowire-based nanotransistors, with nanowire interconnects among all the switching devices. The results of these simulations suggest that a nanomemory of this type can be operated successfully at a density of 10^{11} bits/cm^2. Furthermore, modest device alterations and system design alternatives are suggested that might improve the performance and the scalability of the nanomemory array. These simulations represent early steps toward the development of a simulation-based methodology to guide nanoelectronic system design in a manner analogous to the way such methodologies are used to guide microelectronic system design in the silicon industry.

KEYWORDS: nanotechnology; nanoelectronics; nanocircuits; nanomemory; molecular electronics; very-large-scale integration; crossbar architectures; computer-aided design

INTRODUCTION

Recently, great effort has been dedicated to the design and fabrication of extended electronic circuit systems integrated on the molecular scale. Within only the last few years, the field of molecular electronics has seen the demonstration of single-molecule electrical devices[1–4] and the subsequent construction of functioning simple molecular-scale circuits,[5–12] including a few small nanomemory arrays.[13–16] Additional attempts to produce extended circuit systems are underway. As one such example, the U.S. Government's Defense Advanced Research Projects Agency (DARPA) is attempting to develop and build ultradense computer memories and processors from molecular-scale components.[17] More specifically, DARPA's Moletronics Program is intent on delivering a 16-kilobit nanomemory with a density of

Address for correspondence: Dr. James Ellenbogen, The MITRE Corporation, Mail Stop N230, 7515 Colshire Dr., McLean, VA 22102, USA. Voice: 703-883-7530; fax: 703-883-5963. ellenbgn@mitre.org

Ann. N.Y. Acad. Sci. **1006: 312–330 (2003).**
doi: 10.1196/annals.1292.022

10^{11} bits/cm^2 by the year 2005. By comparison, the International Technology Roadmap for Semiconductors (ITRS) estimates commercial DRAM densities will just exceed 10^9 bits/cm^2 in 2004 and be slightly less than 3×10^{10} bits/cm^2 by 2016.

Despite these aggressive goals, it is still an open question whether such a densely integrated, extended memory system can be made to operate if it is built using existing nanodevices and nanofabrication technologies. This paper addresses that question and answers it in the affirmative.

Specifically, this paper presents simulations and analysis of the experimental nanomemory being developed by Harvard University and the California Institute of Technology (Caltech) under the DARPA Moletronics Program. These simulations suggest that the planned 16-Kilobit nanomemory array will operate effectively.

The simulation methodology consists of constructing device models of experimental molecular-scale electronic devices, for example, diodes and transistors, that already have been demonstrated at Harvard University. These novel device models are then incorporated into conventional very large-scale integration computer aided design (VLSI CAD) simulations of the complete nanomemory system. The primary focus of this simulation research has been to explore the impact on nanomemory function as the bit count is scaled from that of presently fabricated prototypes, with only a few bits, to the 16-Kbit array size required by the Moletronics Program. In addition, a variety of other design parameters, for example, capacitance to ground and banking topology, have also been investigated in order to provide information to the developers at Harvard and Caltech in advance of actual experiments.

This paper provides an overview of these modeling and simulation investigations for the Harvard–Caltech nanomemory. We start with a summary of the crossbar nanomemory structure of interest[19,20] plus the key metrics of its performance. The device models and simulation methodology are then discussed. Simulation results and their implications for nanomemory banking topologies and nanomemory area are presented in subsequent sections followed by our conclusions and future objectives.

NANOMEMORY SYSTEM OVERVIEW AND EVALUATION METRICS

The Harvard–Caltech nanomemory system design consists of three major subsystems: a nanowire crossbar memory array and two decoders, one for the array rows and one for the columns. FIGURE 1 shows an illustration of these subsystems and a circuit schematic of a notional 10×10 nanomemory based on the architectural design developed by Caltech.[20] The crossbar array physically stores the data whereas the decoders select and access individual bits. In FIGURE 1 A, the nanowires forming the crossbar array are represented by thin black lines, as are the nanowires in the decoders. The decoders also contain microwires (wires on the size scale of conventional microelectronics), which are represented by thick gray lines. Finally, microwire power supplies, represented by thick black lines, are connected to each decoder. These power supplies are used to read or write a bit to the individual nanowire junction selected by the decoders.

Each nanowire–nanowire crosspoint in the crossbar array forms a bistable nonvolatile nanowire (NVNW) diode[16,21] that can store one bit of information each. The microwire–nanowire crosspoints in the decoders form field-effect transistors, similar

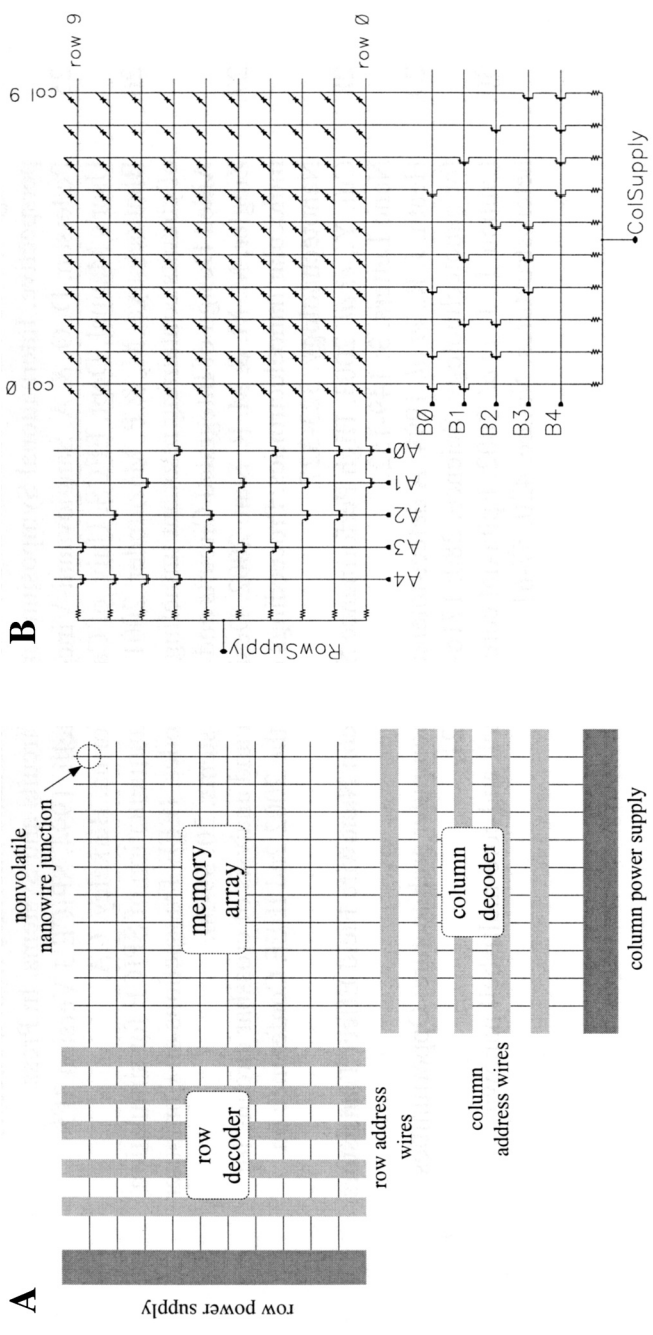

FIGURE 1. A sketch of the structure (**A**) and a circuit schematic (**B**) for the nanomemory design, which consists of a crossbar nanowire memory array composed of nonvolatile nanowire diodes, plus two decoders composed of top-gated nanowire FETs.

to the crossed nanowire transistors described by Huang *et al.*[6] These transistors permit the selection of individual rows and columns in the memory array. Additional details about these nanowire diodes and top-gated nanowire field-effect transistors (TGNW-FETs) are provided in the next section.

Addressing the nanomemory array is accomplished with a "2-hot" encoding scheme.[20] This encoding scheme differs from the binary schemes typically used in complementary metal-oxide–semiconductor (CMOS) circuitry[22] in that the 2-hot scheme requires asserting exactly two microwires to select a unique nanomemory address, no matter the size of the address space. In contrast, binary encoding would require asserting $\log_2 N$ microwires to select a unique address from an address space of size N. The 2-hot addressing scheme increases defect tolerance. With 2-hot addressing, any address line failure impacts significantly fewer bits than a comparable failure in a binary scheme.[20] Second, the 2-hot encoding scheme requires the selected address line to encounter only two TGNW-FETs in series per decoder, whereas binary encoding would meet $\log_2 N$ TGNW-FETs in series per decoder. Reducing the number of series (stacked) TGNW-FETs is beneficial to ensure that an ample amount of the supply voltage is dropped across the selected diode junction, rather than across the decoders.

A variety of metrics are considered for evaluation of nanomemory operation and performance. First and foremost, the nanomemory must function properly. For our purposes, functionality is defined as the ability to correctly read and write information. To evaluate read operations, we focus on the output current differences (ΔI_{out}) between reading logic "1" and reading logic "0" for the worst-case memory configurations. For write operations, we determine that the selected diode, and only the selected diode, is subjected to the appropriate voltage bias for storing the logic value.

Another important design goal of the nanomemory is ultrahigh density. The DARPA Moletronics Program goal of a 16-kilobit memory with a density of 10^{11} bits/cm^2 means that the nanomemory can occupy an area of only 16 μm^2. Therefore, analysis of the nanomemory footprint is warranted. Simple area equations are used to evaluate these parameters.

Finally, consideration is given to the nanomemory speed and power consumption. Output current switching times are analyzed, and methods are suggested to improve speed and reduce power. However, a thorough treatment of these issues requires additional information about the peripheral circuitry surrounding the nanomemory. Such studies are already underway.

SIMULATION METHODOLOGY AND DEVICE MODELING

The simulation of devices and complex circuit systems can be performed at a number of different levels of design abstraction.[22] The appropriate design abstraction depends largely on the goals of the simulation, capturing the important behaviors of the overall system, while disregarding immaterial details. There are three general categories of design abstraction: the device level, the circuit level, and the architectural level. The device level focuses on a single device, for example, a diode or transistor, in great detail. Such simulations provide information about the operation and physics of individual devices, but generally do not consider the interactions

among devices in a circuit. At the other end of the spectrum, the architectural level considers very large systems, but typically does not include the physics or the behavior of individual devices. In contrast, the circuit level bridges these two approaches and considers relatively large systems (on the order of tens of thousands of devices), while still retaining a connection to the underlying physical behavior. The simulations described here take place at this level.

Many concepts and techniques in conventional microelectronics are borrowed to simulate the nanoelectronic memory. For example, the commonly used commercial Cadence Spectre VLSI CAD software tool is our primary simulation program. One reason for applying such commercial off-the-shelf software tools from the microelectronics industry is the obvious timesaving and reliability associated with the use of readily available, well-tested software. This software also incorporates powerful features, such as modeling languages and graphics, developed specifically for the flexible modeling of extended circuitry. Finally, using conventional VLSI tools provides a seamless approach to the design and simulation of both the nanomemory and the peripheral microelectronic circuitry that it requires for operation and communication with the outside world.[23,24]

The work presented here also relies heavily on the conventional microelectronics concept of the device model, which captures the essential properties and response behavior of a circuit element. Models of the experimentally observed behavior of all of the devices contained in the nanomemory are required. In particular, the current–voltage transfer characteristics (I–V curves) are necessary for steady-state (DC) simulation, and the capacitance–voltage transfer characteristics (C–V curves) are required for time-varying, or transient, simulation.

Typical device models for microelectronics consist of compact equations based upon the underlying physics of a device that has been extensively studied and is well understood. However, this physics based approach is not workable, at present, for simulations involving molecular-scale devices, because the fundamental physics of most molecular-scale devices is not well known. Thus, in this work we were forced to develop empirical models based on measured device characteristics.

Incorporating new models into conventional circuit simulators can be difficult. The addition of a new model can often require modifying proprietary source code. Open source simulators do exist, for example, SPICE3,[25] but adding new device models is tedious.[26] Furthermore, these open-source simulators lack the robustness and simulation speed that are required to model large circuit systems, and which are found in many commercial simulators.

Thus, to develop and simulate models for molecular-scale devices efficiently, we chose to use a commercial simulator and to fit or describe the empirical behavior of the devices using the analog hardware description language (analog HDL) Verilog-A. This modeling approach is similar to one described elsewhere,[27] except that the empirical equations derived here are tailored to the devices employed in the Harvard–Caltech nanomemory system. The empirical equations were incorporated into the Spectre circuit simulator from Cadence, which supports cosimulation of both Verilog-A components and conventional SPICE-level devices. The combination of the empirical device modeling with Verilog-A, and the ease of incorporation of these models into the Spectre simulator, made it possible to model quickly novel

molecular-scale devices, simulate circuits composed of these new models, and incorporate state-of-the-art conventional microelectronic device models, as necessary.

The simulations of the Harvard–Caltech crossbar nanomemory required three device models. The first two are models of the distinct nanowire devices described in the previous section: the crossed nanowires that form the nonvolatile nanowire (NVNW) diodes and the top-gated nanowire FETs (TGNW-FETs). The third is a model of the nanowire interconnects of the nanomemory.

Nonvolatile Nanowire (NVNW) Diode Model

FIGURE 2 shows a sketch of the NVNW diodes being fabricated at Harvard University,[16,21] along with a circuit schematic that models the behavior of the device. The model consists of two diode rectifiers in parallel with a capacitor (C_{jdiode}). The model can be switched between a high current state (on-state) and a low current state (off-state) by switching the diode connected to the circuit. This reproduces the hysteretic *I–V* behavior seen in the experimental device. The measured *I–V* characteristics of the actual NVNW diodes and the corresponding model *I–V* curve are shown in FIGURE 3. The measured *I–V* curves were fitted to empirical equations to produce the model. The apparatus used to collect the experimental data shown in FIGURE 3A was limited to measuring currents of up to 1,000 nA, a limit that the device attains at a bias of approximately 3 V. In the model, values for the current passing through the diode at bias potentials greater than 3 V were extrapolated from the available data.

The NVNW diode switches from the on-state to the off-state when a reverse bias is applied across the diode that is more negative than V_{thresOFF}. In a similar fashion, a bias greater (i.e., more positive) than V_{thresON} switches the device from the off-state to the on-state. Device threshold values for the experimental diodes are −2.75 V and 3.80 V for V_{thresOFF} and V_{thresON}, respectively. One issue for this simulation research is whether the device behavior for the NVNW diode switches, for example, the threshold values, is optimal from the perspective of designing and building an

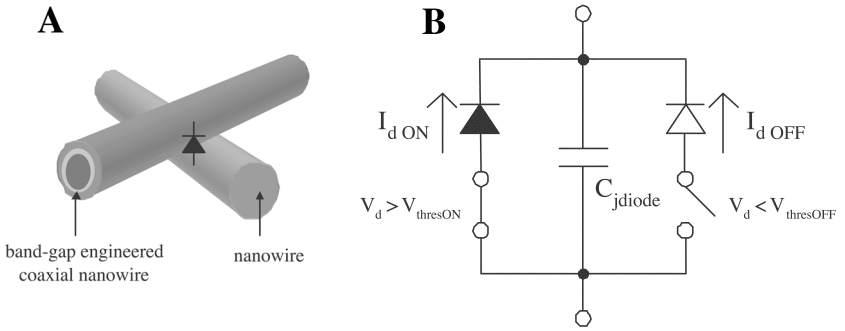

FIGURE 2. A sketch of the structure **(A)** and a circuit schematic **(B)** for a nonvolatile nanowire diode formed by crossing a band-gap engineered coaxial nanowire[30] and a standard nanowire. The device model in the schematic consists of two diodes in parallel with a capacitor. The two individual diodes model the high current state (on-state) and low current state (off-state), respectively.

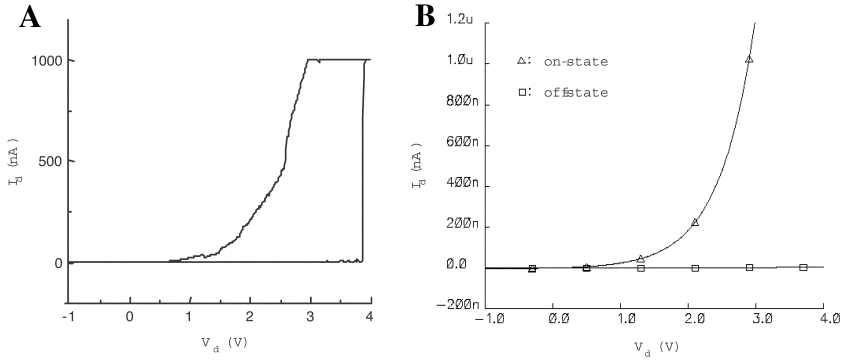

FIGURE 3. Hysteretic *I–V* curve for the nonvolatile nanowire diode. **A.** The measured *I–V* curve for an experimentally fabricated nonvolatile nanowire diode. **B.** The simulated *I–V* curve for the nonvolatile nanowire diode model.

extended memory circuit system, and whether this device behavior might not be improved for that purpose. This question is addressed in the following section.

Although this diode switch appears to exhibit relatively simple behavior, the hysteretic *I–V* curve creates a complicated device modeling task. A smooth transition between curves occurs when switching from the on-state to the off-state at V_{thresOFF}, but the device experiences an abrupt jump in current when switching from the off-state to the on-state at V_{thresON}. This discontinuity in the current requires special provisions in the mathematical models used in the simulation. We avoid any possible difficulties at the discontinuity by simply recording when V_{thresON} has been surpassed, without actually changing the underlying state of the device. This is sufficient for the purposes of the work describe here, because the memory array is simulated for only one configuration at a time. Thus, it is necessary only to determine which of its constituent diodes has crossed its switching threshold. Subsequent analysis of the nanomemory system with the diodes in the switched state is not required.

This technique is not suitable in all situations. Multiconfiguration simulations, for example, those that calculate power consumption during write-read combinations, require continuous processing as the diode switches between the ON and OFF states. This cannot be modeled with the methodology described above. Such simulations, and the more complex models they require, are being developed. Nevertheless, the single configuration simulations presented here are sufficient to determine whether the proposed memory system, constructed from the presently fabricated devices, can be made to operate.

In addition to the *I–V* curve, information concerning the device capacitance is needed for time-dependent simulation. Ideally, we would obtain a transfer curve relating capacitance to voltage in a manner similar to that of obtaining the curve describing the *I–V* behavior. However, experimental data is not yet available to describe the change in capacitance versus voltage. Instead, we used a constant value of 1 aF for the NWNV diode junction capacitance (C_j).[16] In the absence of detailed data, it is hoped that this first-order estimate will provide sufficient information to

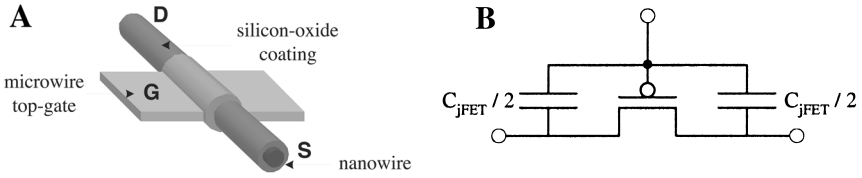

FIGURE 4. A sketch of the structure **(A)** and a circuit schematic **(B)** for a top-gated nanowire FET formed by depositing a microwire over a silicon-oxide coated nanowire. The device model in the schematic consists of a PFET transistor and two capacitors.

estimate overall memory behavior. Nonetheless, the simulations developed in this work can incorporate more detailed capacitance characteristics as they are measured.

Top-Gated Nanowire Field Effect Transistor (TGNW-FET) Model

The decoders are composed of TGNW-FETs that are constructed by crossing a microscale wire over a nanowire covered with silicon oxide. The silicon oxide isolates the microwire from the nanowire and allows the device to behave like a field-effect transistor, with the microscale wire acting as the gate. Changing the voltage on the microwire gate controls the current flow through the nanowire channel. These field-effect devices are similar to the crossed nanowire FETs (cNW-FETs) described by Huang *et al.*[6] An illustration of a TGNW-FET and a circuit schematic of the device model are shown in FIGURE 4. The experimental *I–V* characteristics for *p*-type silicon nanowires coated with silicon oxide and the corresponding TGNW-FET simulation model are shown in FIGURE 5. The device behaves as a *p*-channel MOSFET (PFET), where applying a positive voltage to the gate reduces the conductivity of the channel.[22] The *I–V* equations for the model are modified versions of the typical first-

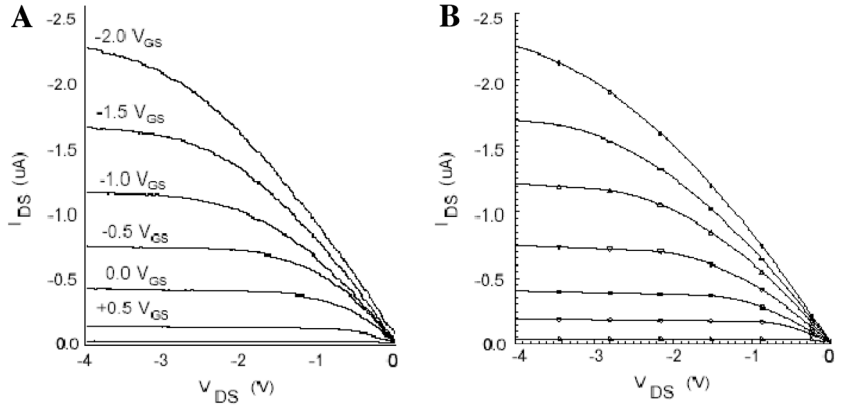

FIGURE 5. *I–V* curves for the top-gated nanowire FETs as a function of gate voltage. **A.** Measured *I–V* curves for a *p*-silicon top-gated nanowire FET. **B.** Simulated *I–V* curves for the top-gated nanowire transistor model.

order MOSFET *I–V* equations. The modifications to the MOSFET equations involve scaling the input voltages and adding an error correction term. These modifications are empirical in nature and remove any direct connection to the underlying physics. However, this is sufficient for the purposes of the present research. The objective is not to represent the underlying physics, but simply to mimic the experimental behavior of the device.

In addition, a capacitance between the nanowire and microwire is present in the model (C_{jFET}). There is not yet an accurate experimental measurement for this value. Instead, we assume the capacitance between the nanowire and microwire in the TGNW-FET is similar to that for the NVNW-diode junction and set $C_{\text{jFET}} = C_{\text{jdiode}}$. In general, the ratio of total NVNW-diode junction capacitance to total TGNW-FET gate capacitance should increase as the memory array size grows. Therefore, C_{jFET} should decline in importance as the nanomemory array grows and improve the validity of our assumption of a small, constant value.

Nanowire Interconnect Model

In conventional microelectronics, there is a clear-cut distinction between the devices and the wires that connect them. This distinction does not exist in the crossbar nanomemory considered here. Nanowires in the nanomemory form the devices, as well as connect the devices to one another. To deal with these two roles, an interconnect model was developed that incorporates an artificial division between the two functions. FIGURE 6 shows an illustration and a circuit schematic of the interconnect model. This is a Π model[22] composed of a resistor and two capacitors. The figure details a unit crossbar (i.e., two crossing nanowires), each the length of the nanowire pitch. The resistance of the unit crossbar determines the values of the resistances (R_{NW}) in FIGURE 6, whereas the capacitors (C_{NWsub}) model the capacitances to the substrate below.

An important resistance in the interconnect model that is not shown in the figure is the contact resistance R_c between the microwire power supplies and the nanowires.

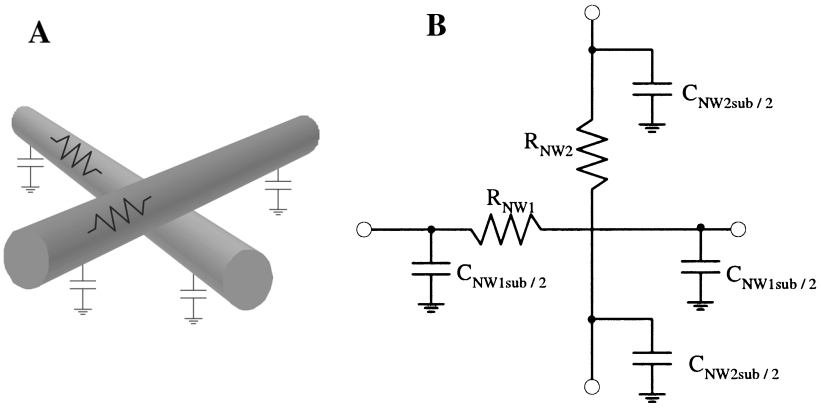

FIGURE 6. A sketch of the structure **(A)** and a circuit schematic **(B)** for the nanowire interconnect model, which consist of networks of resistors and capacitors.

This value is approximately 1 MΩ in the present devices.[28] Furthermore, R_c is dominant in comparison to R_{NW}. Thus, we assume the nanowire resistance is negligible in our simulations. Although the parameter R_{NW} is not employed in the simulations presented in this paper, including it in the interconnect model provides the capability to account for the nanowire resistance when improvements in the fabrication techniques reduce R_c to a value where the two resistances are comparable.

The capacitance to the substrate C_{NWsub} is treated as a variable parameter. The actual experimental value can be altered by the changing the nanomemory distance from the substrate and/or by changing the insulating dielectric between the crossbar and substrate. This value has an important influence on the system performance, as is shown in the next section.

There are two additional parasitic influences that may play a role in nanoscale systems and that were not included in the nanowire interconnect model. Neither crosstalk capacitance between neighboring wires nor parasitic inductances along the wires have been incorporated into the model, thus far. These two effects may manifest themselves in systems with small wire pitches or in systems with long and narrow wires operating at high frequencies, respectively. However, we believe that these effects will not influence strongly the functionality of a low speed, low frequency prototype nanomemory, such as we consider. In other words, whereas these two parasitics may impact the operating conditions and efficiency of the nanomemory, they will not alter whether or not the system can be made to operate. Nevertheless, since these effects may influence the optimization of the nanomemory, simulations that explicitly incorporate these effects are being developed and will be included in future research efforts.

NANOMEMORY SIMULATION AND ANALYSIS

The nanomemory is accessed by placing an address on the row and column decoders and then adjusting the supply voltages to force either a read or write operation. The address asserts a row and a column by turning on the TGNW-FETs in the selected row and column, while turning off at least one TGNW-FET in each non-selected row and column. This procedure isolates a unique point or address in the nanomemory array.

When the TGNW-FETs are turned off, they create an open circuit and leave the voltage upon the non-selected rows and columns "floating", in the absence of a connection to a strong power supply. Allowing the rows and columns to float in this manner risks having non-selected diode junctions inadvertently reprogrammed if these diodes are subjected to voltages from elsewhere in the array that exceed programming thresholds. To help control the voltages across the non-selected rows and columns, a precharge signaling scheme is used. The precharge places a fixed charge on all of the non-selected diodes prior to evaluation. This limits the voltage difference across them.

Each operation is thereby divided into a precharge phase and an evaluation phase. FIGURE 7 shows the waveforms of the input and output signals of these two phases for a read operation on the 10×10 nanomemory shown in FIGURE 1. The simulation first reads diode (8,8), that is, the diode in row 8 and column 8, followed by a read

of diode (9,9). It is of particular importance to be able to simulate the reading of diode (9,9) because it is the worst-case diode for both read and write operations, that is, it is the farthest from the power supplies. Simulation of the reading of diode (8,8) provides an example of the precharge scheme over successive memory accesses. In principle, any address location would do.

The precharge phase asserts all address lines and places a voltage on all the rows and columns. Then, during the evaluation phase, only the selected row and column are asserted. The junction and parasitic capacitances on the non-selected lines hold the precharge voltage while they are isolated from the rest of the circuit. During the evaluation phase, at least one TGNW-FET in each non-selected row and column is turned off, leaving the only path between the row supply and column supply through the selected diode, enabling the reading or writing of a single bit.

When reading a bit from memory, voltages are placed on the row and column supplies such that the selected diode is forward biased, allowing the output current of the nanomemory to reflect the resistance of the selected diode. It is particularly important to choose operating voltages that only forward bias the selected diode. Forward biasing non-selected diodes will cause them to contribute, inadvertently, to the overall output current. In the worst-case memory configuration for reading a logic "0" bit (i.e., when the selected diode is in the off-state and the rest of the diodes

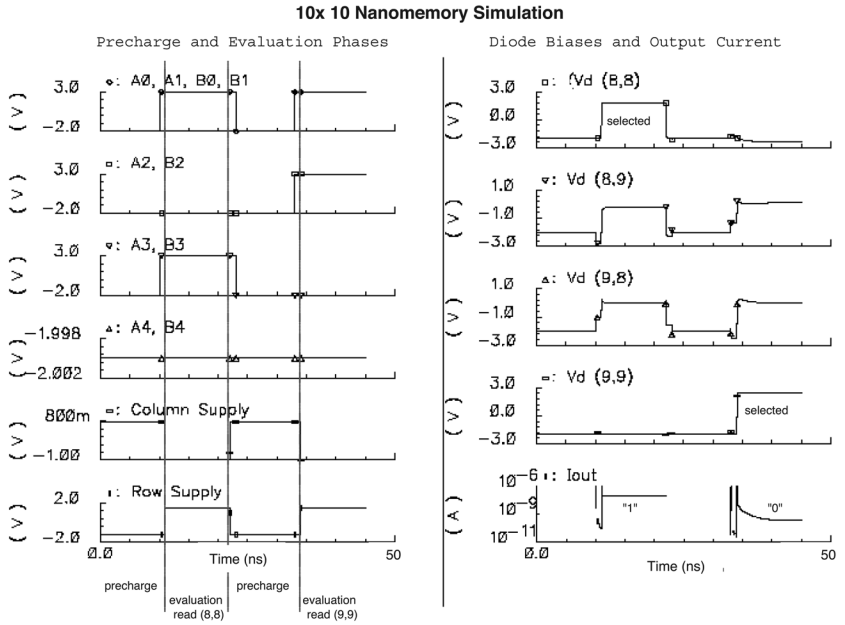

FIGURE 7. Input and output waveforms for the precharge and evaluation phases of two sequential read operations. The *left* half of the figure shows the input signals for a read of diode (8,8) followed by a read of diode (9,9). The voltage biases across the selected diodes, (8,8) and (9,9), their neighboring diodes, (8,9) and (9,8), and the memory output current are shown in the *right* half of the figure.

are in the on-state), even a slight forward biasing of the non-selected junctions may make the state of the selected diode unreadable. This problem increases with the size of the array since there are more non-selected diodes that can contribute to the overall current.

To avoid this interference from non-selected diodes, we choose precharge and evaluation voltages for reading the memory that force non-selected diodes into a reverse bias or near zero bias. This strategy prevents non-selected diodes from contributing to the output current. The right half of FIGURE 7 shows simulation results for the strategy described above. The memory configuration is set to the worst case for reading logic "0". The worst-case diode, that is, diode (9,9), is set to logic "0" and the rest of the diodes are set to logic "1". The top four waveforms are the voltage biases across the diodes being read and two neighboring diodes. The simulation results show that the diodes in non-selected rows and columns are either reverse biased or have a very small forward bias during the evaluation phase.

Although placing non-selected diodes under a reverse bias is effective for reducing unwanted current contributions to the output current, this scheme does run the risk of inadvertently programming on-state devices to off-state devices if the reverse bias exceeds V_{thresOFF}. Therefore, it is necessary to use supply voltages that are small enough to ensure V_{thresOFF} is not surpassed. This, in turn, limits the bias that can be placed across the selected diode.

Nevertheless, in the simulation it is possible to achieve excellent ON/OFF current differences for a variety of different memory arrays, as is shown in FIGURE 8. Similarly, TABLE 1 provides details of the output currents I_{out} for worst-case read operations for both logic "1" and logic "0", as well as the current difference ΔI_{out} and current ratios "1"/"0" current ratio between them. These differences are sufficient to read each memory successfully. Furthermore, the high current ratios suggest that

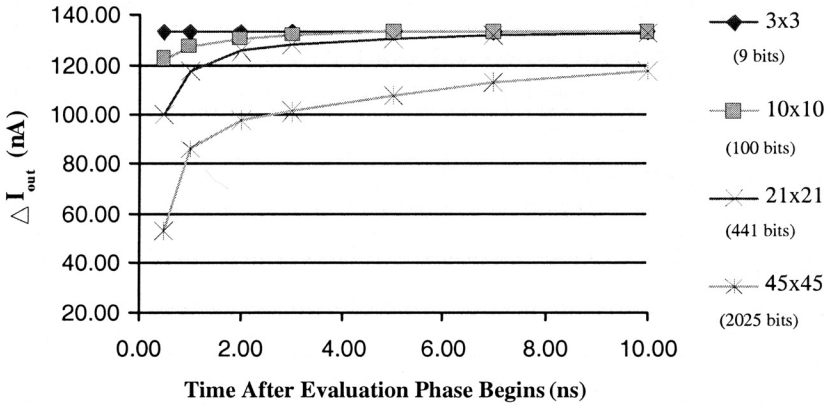

FIGURE 8. Plot of ΔI_{out} versus time. $T = 0$ corresponds to the beginning of the evaluation phase. The time it takes for ΔI_{out} to reach its maximum value has implications on the speed of the memory. The simulations above use zero capacitance between the nanowires and the substrate.

TABLE 1. Simulation results for a read operation performed on the 10 × 10 nanomemory shown in FIGURE 1

Nanomemory Array Size	I_{out} (nA)		ΔI_{out} (nA)	"1"/"0" Current Ratio
	logic "1"	logic "0"		
3×3	134	0.8	133	168
10×10	134	0.9	133	149
15×15	134	1.1	133	122
21×21	134	1.6	132	84
45×45	134	16.7	117	8

NOTE: The simulations are performed with zero capacitance to ground and the reported values occur 10 nsec after the evaluation phase begins (see text for details).

read operations can be performed successfully in memory arrays that have been scaled up to include even more rows and columns.

Data is written to the nanomemory by subjecting the selected diode to a bias exceeding the switching threshold. As discussed in the previous section, a diode in the on-state is switched to the off-state at $V_{thresOFF} \approx -2.75\,V$ and a diode in the off-state is switched to the on-state at $V_{thresON} \approx 3.8\,V$. As with the read operations, care must be taken to avoid inadvertently programming non-selected diodes. However, simulations performed in this work suggest it is feasible to write either logic value to the memory. It was always possible to identify operating conditions that programmed the selected diode without subjecting non-selected diodes to voltages that exceeded thresholds.

The simulations shown in FIGURE 8 and TABLE 1 assume no capacitance between the nanowires and the substrate, that is, $C_{NWsub1} = C_{NWsub2} = 0$. This is a reasonable approximation and can be realized experimentally by raising the crossbar nanomemory sufficiently high above the substrate and/or using a low-k dielectric between the nanomemory and substrate. Likewise, it should be possible to add a controlled amount of capacitance to the nanowires by reducing the height above the substrate and/or using an alternative dielectric. Recent experiments have shown that the capacitances between the memory cell of interest and the substrate may be estimated to be approximately 1 aF. Thus, the simulations described above were repeated with this small capacitance to ground added to each unit crossbar in the nanowire interconnect model, that is, $C_{NWsub1} = C_{NWsub2} = 1\,aF$. As is shown in FIGURE 9, adding capacitance to ground reduces the ΔI_{out} settling times, particularly for the larger arrays. This reduction in settling times occurs because the capacitance to ground provides a better environment for holding the precharge. Without capacitance to ground, the junction capacitance dominates and capacitive coupling to crossing wires can reduce the effectiveness of the precharge.

The simulations developed in this work also can evaluate the effects of varying design parameters on specific aspects of nanomemory performance or evaluate the tradeoffs between traditionally disparate design goals, such as high speed versus low power. For example, the output current difference ΔI_{out} can be improved by either shifting $V_{thresOFF}$ to a lower voltage (more negative voltage) and/or by increasing

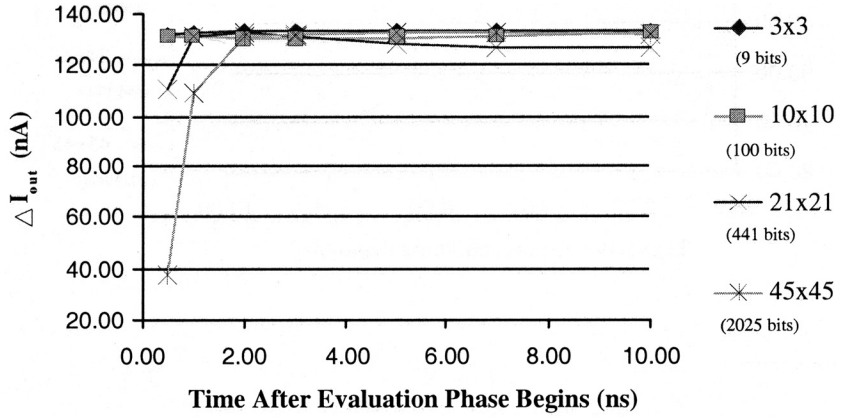

FIGURE 9. Same as FIGURE 8 except that the simulations above have 1 aF of capacitance between the nanowires and the substrate. These simulations show a small amount of capacitance produces shorter ΔI_{out} settling times.

$V_{thresON}$. Increasing ΔI_{out} should lead to increased speed and array size. However, altering the programming threshold in this manner requires more energy during write operations. This, of course, increases power consumption. Simulation is an effective way to examine these tradeoffs in a quantitative manner. Work is ongoing to identify optimal operating parameters for various design goals.

For all of the simulations performed to date, the voltage swing for the input signals is relatively large, requiring the address lines to vary by 5 V, while the row supply and column supply vary by 2.75 volts and 1.75 volts, respectively. These large voltage swings will most likely consume significant dynamic power and require level shifting circuits to interface with conventional electronics. Thus, reducing the signal swing should be an experimental goal. This will reduce power consumption and ease integration with conventional circuits. However, achieving this goal may require smaller diode thresholds. This may reduce the memory speed and could affect functionality. Additional simulations that explicitly incorporate external CMOS circuits are required to explore this issue more fully.

Nevertheless, the simulation results thus far suggest that a 45×45 nanomemory will be functional using the realistic system parameters derived from present experiments. Simulations of larger memory arrays up to 125×125 are planned, but the general trends of the results suggest that these larger memories will be functional as well. Furthermore, as is shown in the next section, the ability to assemble 45×45 nanomemories could be of considerable utility, because their use in a banked topology provides a route to realizing a 16-kilobit nanomemory, which is the DARPA development goal.

Although the simulations to date have suggested that the memory is scalable and will function under present device and design parameters, other factors should be considered in future simulations. For example, as the size of the nanomemory array grows, so does the capacitance and resistance of the rows and columns, which can hamper memory performance. FIGURE 8 shows the time dependence of ΔI_{out} for

simulations of four different memory sizes. The figure shows that increasing the size of the memory also increases the time needed for ΔI_{out} to reach its maximum value. This settling time may reduce the speed of the memory. However, detailed information about the connection of the nanomemory to conventional micron-scale CMOS circuitry, in this case signal amplifiers, is necessary for any realistic estimation of the memory speed.

BANKING TOPOLOGIES AND AREA ESTIMATES

As the previous section suggests, simply increasing the size of a single nanomemory array may not be the most effective approach for producing memories with very high bit counts. As the size of a memory arrays increases, so do the resistances and capacitances associated with the array, which increase delay and power consumption. Ultimately, this may threaten functionality.

In addition, large memory arrays are more susceptible to fabrication defects, since a single defect in a wire can render all the memory cells along it unusable. Reducing the vulnerability of nanomemories to defects is important. This is because, based on statistical and thermodynamic arguments, it is anticipated that the hierarchical self-assembly strategies being pursued for molecular-scale electronic circuits may produce a significant fraction of defective devices or devices that are imprecisely positioned.[29]

To increase defect and fault tolerance, instead of using a single large array to achieve a high bit count, banks of smaller memories might be employed. FIGURE 10 illustrates the notion of banking by showing how a one-kilobit memory array can be represented as a single 32×32 array or four 16×16 arrays. This strategy allows for the same level of defect tolerance with less redundancy, since any single defect impacts a smaller number of individual memory bits. Generally speaking, as the degree of banking increases, the amount of required redundancy should decrease, since smaller arrays pay a lower price per defect.

Adopting a banking strategy also increases the overall data throughput for the memory. First, the lower resistances and capacitances of the shorter nanowires in the smaller arrays allow faster access times. Second, banked arrays can be accessed in parallel (i.e., a bit can be accessed from each bank simultaneously) significantly increasing memory performance. Although banked architectures can create more complex fabrication patterns, the regularity of the banks would seem to provide a feasible route to nanomemory assembly. Harvard University already has made significant progress in the parallel fabrication of multiple arrays in a tiled pattern.

The one significant tradeoff generally associated with employing a banking strategy is an increase in area per usable bit. This occurs because each additional bank requires additional wires for encoding and decoding the memory array. Although some of these wires can be shared among the banks (see FIG. 10), banking always results in an increase in the number of address wires. Thus, an optimal banking strategy will employ moderately sized arrays that take advantage of the coding scheme to increase density, but also achieve the requisite degree of defect tolerance, parallel access, and/or other design goals.

Despite the various possible banking topologies for producing a 16-kilobit nanomemory, first-order area calculations suggest that the target nanowire pitch

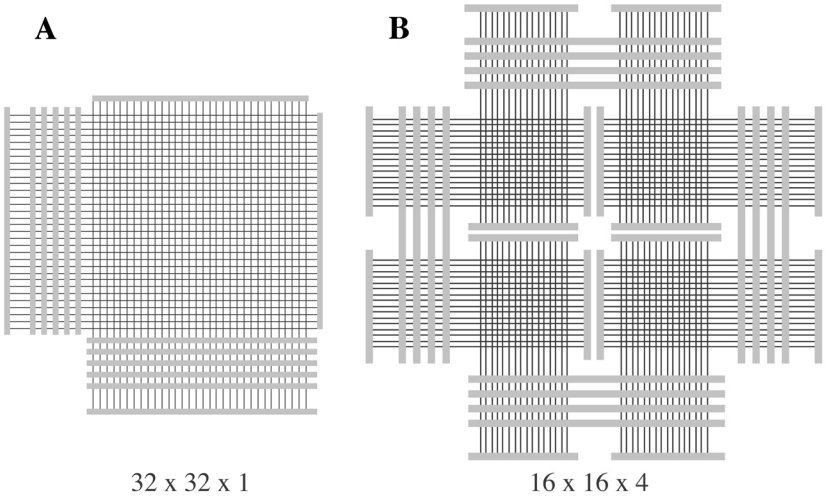

FIGURE 10. Illustration of two different topologies for realizing a 1-kilobit memory. **(A)** A single array and **(B)** a bank of four arrays with an equivalent number of bits.

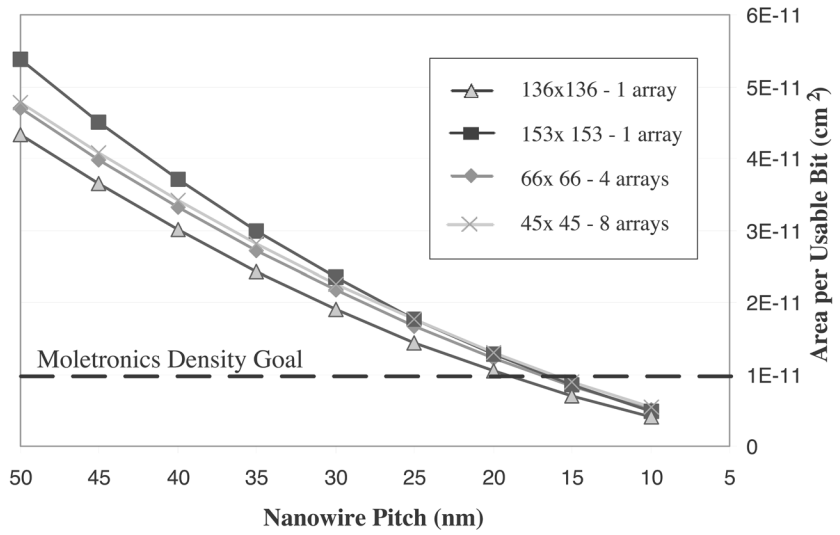

FIGURE 11. Plot of estimated area per usable bit versus nanowire pitch for four memory arrangements targeting the 16-kilobit DARPA nanomemory. The calculations assume only 16-kilobits of data can be accessed and the remaining memory locations are reserved for redundancy. The microwire pitch is set at 100 nm for all four arrays.

TABLE 2. Estimated nanomemory area for four different memory arrangements targeting the 16-kilobit DARPA nanomemory

Memory Arrangement	Total Locations	Percent Redundancy	Total Area (square microns)		
			20 nm pitch	15 nm pitch	10 nm pitch
136×136 – 1 array	18,496	15.6%	16.6	11.1	6.5
153×153 – 1 array	23,409	46.3%	20.4	13.5	7.8
66×66 – 4 arrays	17,424	8.9%	19.6	13.4	8.1
45×45 – 8 arrays	16,200	1.3%	20.9	14.4	8.8

should be similar for a variety of topologies. FIGURE 11 shows the estimated bit density for three different banking strategies as a function of nanowire pitch. We also consider two different amounts of redundancy for a single array implementation. These area estimations are premised on the goal of providing 16-kilobits of accessible memory, where any additional memory locations are assumed to be used only as replacements for faulty bits. In other words, the area per usable bit is calculated by dividing the total area for each topology by 16,000, regardless of the actual number of bits. The microwire pitch was set to 100 nm for all of the area calculations. Details of these four memory arrangements are given in TABLE 2.

The four different memory arrangements described in FIGURE 11 and in TABLE 2 all reach the Moletronics Program density goal (10^{11} bits/cm^2) when the nanowire pitch is approximately 15 nm. However, a more appropriate measure of the nanotechnology employed in the fabrication of the nanomemory might ignore the area occupied by the microwires and just consider the area occupied by the nanowires. In that case, a nanowire pitch of approximately 30 nm would suffice to achieve the density goal. Combining these results with functional 45×45 nanomemory simulations suggests that the 16-kilobit DARPA nanomemory is an attainable goal. Furthermore, as we continue simulation efforts to explore still larger functional nanomemories, we believe a variety of topological strategies will be viable.

SUMMARY AND FUTURE WORK

Simulations performed on the experimental nanomemory being developed by Harvard University and Caltech suggest that if such a system were built, it would operate. Results thus far suggest that a 45×45 nanomemory array constructed from presently fabricated experimental devices in the proposed design would function properly. Furthermore, such arrays could be banked to achieve the DARPA Moletronics Program objective of a 16 Kbit molecular-scale electronic nanomemory with a bit density of 10^{11} bits/cm^2.

The simulations described in this paper were focused on projecting the likely functionality of larger arrays of such a nanomemory than have been built and tested to date. The read operation is evaluated by comparing the difference between the worst-case output currents when reading a logic "1" versus a logic "0". The simulated output currents suggest the nanomemory should function properly. The writing of

both logic "1" and logic "0" to memory was also simulated with a similar prediction for successful operation. Finally, results concerning the settling times of the output currents were presented. These results give an indication of the memory speed and suggest that adding a small amount of capacitance between the nanowires and the substrate may the increase speed of the memory. Furthermore, a small amount of substrate capacitance creates a better environment for holding the precharge and increases the stability of the memory.

Equations were derived to estimate the area of the nanomemory and its potential density. Multiple banking topologies were considered with the conclusion that a nanowire pitch of 15 nm or less will achieve the Moletronics Program goal of 10^{11} bits/cm^2, if one includes in the density estimate the area occupied by the microwires, as well as that occupied by the nanowires. If only the area occupied by the nanowires is considered, then a nanowire pitch of approximatelyt 30 nm will suffice.

The work in this paper represents the first steps towards a detailed modeling and simulation methodology for extended electronic circuits integrated on the molecular scale. Many additional steps are required in future work. These include more accurate device and interconnect models and the ability to model variations in devices, both in terms of global variations in devices (i.e., corner analysis) and in terms of variations among individual devices. Also, it is important to incorporate features that permit the design and simulation of nanocircuits with random defects, in order to evaluate fault tolerance.

Reduced circuit models are required to analyze very large nanomemories having device counts beyond the capabilities of SPICE-level simulation. In addition, the capability to simulate nanoscale circuits along with the interfacing conventional electronics is required. This is essential for determining the performance of the system as a whole.[23,27] This is one motivation for adopting a design paradigm that allows seamless integration of both technologies. Finally, it is important to continue to search for new circuits and architectures for the design and operation of very extensive arrays of ultradense memory using molecular-scale devices.

ACKNOWLEDGMENTS

The authors thank Travis Markley and Mark Cannarsa of the MITRE Corporation for their helpful suggestions and assistance with the area calculations and device models. Professors Mircea Stan and Garrett Rose from the University of Virginia gave generously of their time for informative discussions on a number of topics associated this work. This research was supported by the DARPA Moletronics Program.

REFERENCES

1. Bumm, L.A., *et al.* 1996. Are single molecular wires conducting? Science **271:** 1705–1707.
2. Reed, M.A., *et al.* 1997. Conductance of a molecular junction. Science **278:** 252–254.
3. Reed, M.A. 1999. Molecular-scale electronics. Proc. IEEE **87:** 652–658.
4. Metzger, R.M., *et al.* 1997. Unimolecular electrical rectification in hexadecylquinolinium tricyanoquinodimethanide. J. Am. Chem. Soc. **119:** 10455–10466.

5. COLLIER, C.P., *et al.* 1999. Electronically configurable molecular-based logic gates. Science **285:** 391–394.
6. HUANG, Y., *et al.* 2001. Logic gates and computation from assembled nanowire building blocks. Science **294:** 1313–1317.
7. DERYCKE, V., *et al.* 2001. Carbon nanotube inter- and intramolecular logic gates. Nano Lett. **1:** 453–456.
8. BACHTOLD, A., *et al.* 2001. Logic circuits with carbon nanotube transistors. Science **294:** 1317–1320.
9. LIU, X.L., *et al.* 2001. Carbon nanotube field-effect inverters. Appl. Phy. Lett. **79:** 3329–3331.
10. CUI, Y. & C.M. LIEBER. 2001. Functional nanoscale electronic devices assembled using silicon nanowire building blocks. Science **291:** 851–853.
11. HUANG, Y., *et al.* 2001. Directed assembly of one-dimensional nanostructures into functional networks. Science **291:** 630–633.
12. MELOSH, N.A., *et al.* 2003. Ultrahigh-density nanowire lattices and circuits. Science **300:** 112–115.
13. LUO, Y., *et al.* 2002. Two-dimensional molecular electronics circuits. Chem. Phys. Chem. **3:** 519–525.
14. CHEN, Y., *et al.* 2003. Nanoscale molecular-switch crossbar circuits. Nanotechnology **14:** 462–468.
15. KUEKES, P.J., J.R. HEATH & R.S. WILLIAMS, inventors. Hewlett-Packard, assignee. 2000. USA Patent 6128214. Date of application: March 29, 1999.
16. LIEBER, C.M., *et al.* 2002. Design and hierarchical assembly of nanowire-based moletronics. DARPA Moletronics PI Meeting.
17. KWOK, K.S. & J.C. ELLENBOGEN. 2002. Moletronics: future electronics. Mater. Today **5:** 28–37.
18. ASSOCIATION, S.I. 2001. International Technology Roadmap for Semiconductors, 2001 edit.
19. DEHON, A. 2002. Array-based architecture for molecular electronics. Proceedings of the First Workshop on Non-Silicon Computation. **1:** 1–8.
20. DEHON, A. 2003. Array-based architecture for FET-based, nanoscale electronics. IEEE Trans. Nanotech. **2:** 23–32.
21. WANG, D., *et al.* 2003. Science. Submitted for publication.
22. RABAEY, J.M., A. CHANDRAKASAN & B. NIKOLIC. 2003. Digital Integrated Circuits: A Design Perspective. Prentice-Hall, Upper Saddle River.
23. ZIEGLER, M.M. & M.R. STAN. 2002. A case for cmos/nano co-design. IEEE/ACM International Conference on Computer Aided Design. 348–352.
24. ZEIGLER, M.M. & M.R. STAN. 2003. The cmos/nano interface from a circuits perspective. International Symposium on Circuits and Systems. **4:** 904–907.
25. PEDERSON, D.O. & A. SANGIOVANNI-VINCENTELLI. 1994. SPICE 3 version 3F5 User's Manual. Department of EECS, University of California, Berkeley, CA.
26. BHATTACHARYA, M. & P. MAZUMDER. 2001. Augmentation of SPICE for simulation of circuits containing resonant tunneling diodes. IEEE Trans. Computer-Aided Design Integrated Circuits Systems **20:** 39–50.
27. ZIEGLER, M.M. & M.R. STAN. 2002. Design and analysis of crossbar circuits for molecular nanoelectronics. Proceedings of the 2002 2nd IEEE Conference on Nanotechnology. 323–327.
28. CUI, Y., *et al.* 2003. High performance silicon nanowire field effect transistors. Nano Lett. **3:** 149–152.
29. HEATH, J.R., *et al.* 1998. A defect-tolerant computer architecture: opportunities for nanotechnology. Science **280:** 1716–1721.
30. LAUHON, L.J., *et al.* 2002. Epitaxial core-shell and core-multishell nanowire heterostructures. Nature **420:** 57–61.