Modeling a Floating-Gate Memristive Device for Computer Aided Design of Neuromorphic Computing

L. Danial[1], *Student Member, IEEE*, V. Gupta[1,2], E. Pikhay[3], Y. Roizin[3], and S. Kvatinsky[1], *Senior Member, IEEE*
[1]*The Andrew and Erna Viterbi Faculty of Electrical Engineering, Technion - Israel Institute of Technology, Haifa, Israel*
[2]*School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, USA* [3]*TowerJazz, Migdal HaEmek, Israel*
Email: sloaidan@campus.technion.ac.il

***Abstract*— Memristive technology is still not mature enough for the very large-scale integration necessary to obtain practical value from neuromorphic computing. While nonvolatile floating-gate "synapse transistors" have been implemented in very large-scale integrated neuromorphic systems, their large footprint still constrains an upper bound on the overall performance. A two-terminal floating-gate memristive device can combine the technological maturity of the floating-gate transistor and the conceptual novelty of the memristor using a standard CMOS process. In this paper, we present a top-down computer aided design framework of the floating-gate memristive device and show its potential in neuromorphic computing. Our framework includes a Verilog-A model, small-signal schematics, a stochastic model, Monte-Carlo simulations, layout, DRC, LVS, and RC extraction.**

Keywords—Floating-gate, neuromorphic computing, CAD, device modeling, artificial neural networks, memristors, flash memory, VLSI

I. Introduction

Neuromorphic computing is an emerging engineering discipline that strives to reproduce the brain's cognitive and adaptive abilities in hardware by mimicking its architectural and dynamical properties [1]. To implement this novel paradigm, researchers have turned to machine learning for inspiration, as it has already achieved adaptive and error-tolerant training in software for a variety of applications. Artificial Neural Networks (ANNs) are an example of such trainable architectures [2]. Notably, the difficulties in scaling current computing platforms to meet the future demands, combined with the looming end of Moore's law, is spurring widespread interest in neuromorphic computing.

The concept of using nonvolatile memories in analog neuromorphic networks is at least 30 years old [3]. The speed and energy efficiency of these memories exceed those of digital circuits of the same functionality because the vector-matrix-multiplication (VMM), considered the most computationally intensive task performed in ANNs, is implemented by means of Ohm's law and Kirchhoff's law. The key component of such networks is essentially an analog nonvolatile memory device with adjustable conductance, mimicking the biological synapse. Such devices were implemented mostly as floating-gate "synapse transistors", due to their compatibility with the standard CMOS process. However, these devices had relatively large areas and low retention time in old technology nodes, and they led to long time delays and high energy consumption.

In the last decade, the field of nanoelectronic memory devices has been revived by new developments. Of particular importance are the nonvolatile two-terminal resistive switching devices, known as RRAM or memristors [4]. These passive devices have a small footprint, new computational capabilities, and diverse applications. A number of memristor-based neuromorphic circuits with various technologies for synaptic devices (*e.g.*, PCM, ferroelectric, and STT-MRAM) [5] have been realized. Impressive examples of neuromorphic networks [6] were based on the so-called 1T1R technology, in which every memory cell is coupled to a select transistor. Networks based on passive 0T1R devices have so far shown only limited functionality, in part due to the strict requirement for their *I–V* uniformity. Functionality has thus been demonstrated only for small networks [7], for which better density, and performance were also shown. However, the fabrication of memristive technology still has a long way to go before it is ready for the large-scale integration necessary to obtain practical value from neuromorphic computing.

In recent years, the nonvolatile floating-gate memory cells have been optimized, scaled down, and embedded in CMOS integrated circuits. A prototype of a three-layer mixed-signal neuromorphic network was implemented, using digital-input-analog-weight arrays of floating-gate cells redesigned from a commercial 180nm NOR flash memory [8]. Their main advantage is the mature fabrication technology. The memory arrays have been redesigned to allow for individual, precise adjustment of the memory state of each device. The floating-gate cells were operated at the energy-efficient subthreshold domain with exponential I-V characteristics. However, the large footprint, high switching voltages, number of terminals, analog data retention, and the exponential I-V characteristics of the device still limit the training performance. Thus, further optimizations of these cells should be investigated.

A novel scalable memristive floating-gate device was previously proposed and experimentally demonstrated in a standard CMOS process [9]. The memristive device is gradually switched as a floating-gate memory device but with lower voltages and shorter times. It operates at the subthreshold domain, similar to a "synaptic transistor" but linearized for small-signal voltages as a resistor (linear I-V). In this paper, we provide a neuromorphic-oriented computer aided design (CAD) framework of the floating-gate memristive device. The framework includes a Verilog-A SPICE model, small-signal equivalent schematics, a stochastic model, Monte-Carlo simulations, layout, LVS, DRC, and RC extraction.

The remainder of this paper is organized as follows. In Section II, background on the floating-gate memristive device is provided. Section III introduces the CAD framework. In Section IV, memristive arrays are designed using the framework and robustly analyzed. In Section V, basic ANN principles, applied on the array, are briefly introduced to show the feasibility of neuromorphic computing by using the proposed framework. The paper is summarized in Section VI. We believe that such a framework will spur the development of large-scale neuromorphic integrated circuits in the near future.

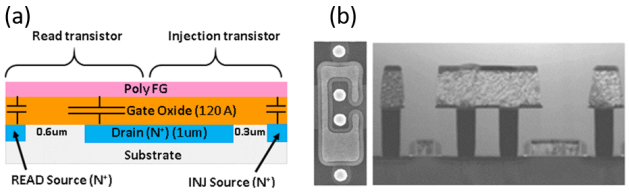

Fig. 1. (a) Single poly Y-Flash device cross-sectional structure in a standard CMOS process, using two NMOS transistors, with asymmetrical coupling gate-drain to gate-source capacitance ratio, where the gate-drain capacitance is made larger. (b) SEM photography of the Y-Flash device: top and cross-section view.

II. Floating-Gate Memristive Device

A. Device Structure and Physics

The floating-gate memristive device, based on the Y-flash technology, comprises two NMOS transistors coupled with a common floating gate (FG) [9] as depicted in Fig. 1(a). The device is manufactured using a standard CMOS process flow and requires no additional masks. The FG potential is controlled by a capacitive coupling between the FG and the common drain junction. This control is obtained by the Miller capacitance of the drain, which is made larger than the source using a customized layout. Top-view and cross-section images of the Y-flash taken by a scanning electron microscope (SEM) are shown in Fig. 1(b). When a positive voltage is applied to the drain, with terminal connections as listed in Table I, a fraction of the drain voltage is transferred to the FG. If the transferred voltage exceeds the threshold voltage of NMOS (V_{th}), the Y-flash cell conducts in the saturation region. Otherwise, it conducts in the subthreshold region. When a higher voltage is applied to the drain terminal, hot electrons are generated in the drain junction by means of the channel hot electron (CHE) mechanism [10]. Some of these electrons are injected into the FG, leading to an increase in V_{th} and programming of the cell. To erase the Y-flash, high positive voltage is applied to the source injection terminal, while the drain and substrate are as configured in Table I. Hot holes are generated by band-to-band tunneling (BTBT) in the source of the injection transistor and are injected into the FG, thus decreasing V_{th} of the device [11].

The device is asymmetric in nature, where the injection transistor is made with a shorter length than the readout transistor and is optimized with p-type LDD doping in the junction to enhance hot carrier injection. The readout transistor is optionally added to eliminate CHI disturbances while the internal state is being read. This readout transistor has lower V_{th} and aids in lowering the applied drain voltages for similar dynamic ranges of output current.

B. Large-Signal Functional Model

Here, we introduce and model the operational modes of the Y-flash memristive device: read, reset, and set.

a. Read mode

The memristive device is obtained by externally shorting the sources of the read and injection transistors. This converts the three-terminal Y-Flash memory cell into a two-terminal memristive device, as illustrated in Fig. 2. The internal memory state V_{th} of the Y-flash can be determined during its operation in the subthreshold region. This is done by supplying a read voltage $(e.g., 2V)$ across the drain as per Table I, and then measuring the current through the device.

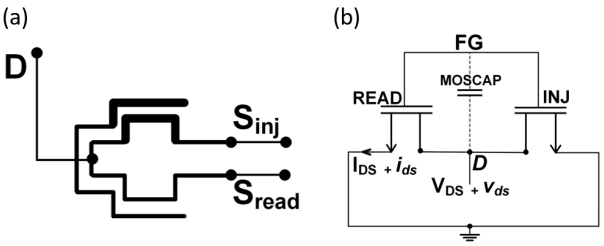

Fig. 2. (a) Schematic symbol of a three-terminal Y-flash. (b) The Y-flash memristive device in read mode, as schematically modeled using three NMOS transistors: parasitic (drain to gate coupling capacitor), read and injection. The current via the injection transistor is negligible in subthreshold.

TABLE I: Operational Modes of Y-Flash.

Mode	Terminal name			
	Drain	Source Read	Source Injection	Sub
Read	$2V$	0	0/High-Z	0
Program	$4.5V/5V$	0/High-Z	0	0
Erase	0/High-Z	0/High-Z	$8V$	0

TABLE II: Programming Model Parameters.

Parameter	Model	
	Programming	
	Short channel	Long channel
$P(E_v)$	$e^{-\frac{E_v}{KT_e}}$	$e^{-\frac{E_v}{q\lambda E_m}}$
K'	$\frac{\mu_n C_{ox} W}{2}$	$\frac{\mu_n C_{ox} W}{2L^2 E_{sat}}$
n	1	2

The subthreshold I-V relationship emulates Ohm's law:

$$I_{DS} = I_{read} \underbrace{e^{CR\frac{V_{DS}}{mV_T}}}_{V(V_{DS})} \underbrace{e^{-\frac{V_{th}}{mV_T}}}_{G(V_{th})}, \quad (1)$$

where m is a technology constant called the subthreshold slope factor, CR is the coupling ratio between gate-drain and gate-source capacitances of the asymmetric device, and $I_{read} = 1nA$ is the current at voltage equal to V_{th}.

The current dynamic range is three orders of magnitude $[1nA: 1\mu A]$, and the state variable range is determined as $\Delta V_{th} = \frac{mV_T}{CR}\ln(I_{on}/I_{off})$. Therefore, $V_{th} \in [V_r - \Delta V_{th}: V_r] = [1V: 2V]$. Intermediate levels are gradually obtained by applying program/erase pulses with specific time width.

b. Programming/RESET mode

The RESET process is modeled by the "lucky electron" model of CHE injection [10]:

$$\frac{dV_{th,RESET}}{dt} = \frac{I_{inj}}{C_{fr}} \approx \overbrace{\frac{K'}{C_{fr}}(CR \cdot V_{DS} - V_{th})^n \cdot P(E_V)}^{f(V_{DS}, V_{th,RESET})}, \quad (2)$$

where C_{fr} includes the floating-gate oxide and the fringing gate-drain capacitance, I_{inj} is the equivalent current of injected hot electrons to the shared floating gate via the injection transistor and equals $I_{DS} \cdot P(E_V)$, and $P(E_V)$ is the probability of a hot electron traveling a sufficient distance to gain energy E_V without a collision. K' is a technology parameter and is a function of

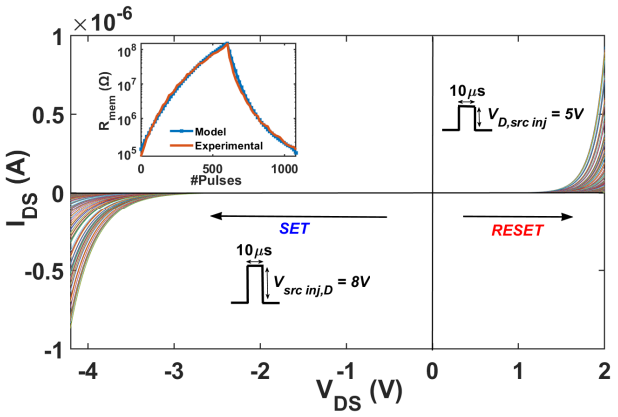

Fig. 3. I-V hysteresis of the proposed memristive model. Reading R_{mem} by a voltage sweep at every state which varies as a function of pulses number for programming voltage of $5V/10\mu s$ and erasing voltage of $8V/10\mu s$. Inset illustrates modeled memristance fitted to the small signal measurements of fabricated DUTs in [9].

transistor width W, mobility μ, gate oxide capacitance C_{ox}, saturation velocity field E_{sat}, and the channel length L (short/long channel modulation). I_{DS} is the transistor channel current with saturated velocity:

$$I_{DS} = \frac{\mu C_{ox}}{2} \cdot \frac{W}{L}(V_{GS} - V_{th})V_{DS,sat}, \quad (3)$$

where $V_{DS,sat}$ is the saturation velocity as a result of high drain voltage and is equal to $V_{DS,sat}(L) \approx (V_{GS} - V_{th}) \parallel (L \cdot E_{sat})$. The Y-flash characteristics can be determined either by the long channel model for $4.5V/10\mu s$ pulses or by the short-channel model for $5V/10\mu s$ pulses as depicted in (2) and listed in Table II. We use the short channel model for both cases instead of the long channel model, although both achieve similar results, because it is much simpler and requires fewer parameters. Since K', $P(E_v)$ and CR are constants, and n =1, for simplicity we represent their multiplication as a constant. Furthermore, for a specific programming voltage, *i.e.*, $4.5V$ or $5V$, the parameter $CR \cdot V_{DS}$ remains constant. Therefore, (2) is reduced to:

$$\frac{dV_{th,RESET}}{dt} = b\left(a - V_{th,RESET}\right), \quad (4)$$

where a and b are constants as listed in Table III, determined by fitting the programming measurements of a fabricated Y-flash device under test (DUT) in [9] to the proposed model. The obtained expression is in the form:

$$V_{th,RESET} = a - e^{c-bt}, \quad (5)$$

where c is the integral constant and t is the input pulse period.

c. Erase/SET mode

Analogously, the SET process is based on BTBT hot holes injection modeled using gate-induced drain leakage current (GIDL) [11]:

$$I_{GIDL} = A \cdot E_s e^{-\frac{B}{E_s}},\ E_s = \frac{CR \cdot V_S - V_{th} + V_{ox}}{t_{ox}}, \quad (6a)$$

$$\frac{dV_{th,SET}}{dt} = \frac{I_{GIDL}}{C_{dep}} = \underbrace{\frac{A \cdot (CR \cdot V_s + V_{ox} - V_{th})}{C_{dep} t_{ox}} e^{-\frac{Bt_{ox}}{CR \cdot V_s + V_{ox} - V_{th}}}}_{f(V_{DS}, V_{th,SET})}, \quad (6b)$$

where A and B are constants for indirect phonon-assisted tunneling, E_s is the vertical electrical field at the silicon surface, t_{ox} is the oxide thickness in the overlap region, $V_{ox} = -Q_{dep}/C_{ox}$ is the oxide voltage, and C_{dep} is the depletion layer capacitance. Simulated annealing is used to determine optimized parameters for the data extracted from erasing measurements in [9] to fit (6b). By considering $CR \cdot V_S + V_{ox}$ and $\frac{A}{C_{dep} t_{ox}}$ as constants, (6b) can be reduced to:

$$\frac{dV_{th,SET}}{dt} = a \cdot (b - V_{th}) e^{-\frac{c}{b - V_{th}}}, \quad (7)$$

where a, b, and c are constants listed in Table III. The state variable dynamics and the corresponding measured current in SET/RESET of fabricated DUTs in [9] are fitted to the physical models, in addition to the corresponding current which is read after applying each programming/erasing pulse, exhibiting an I-V hysteresis as shown in Fig. 3. Notably, the analogies between (2), (6b) and the abstract kinetics that model the state variable evolution of any memristor [12] prove the memristive characteristics of the device.

TABLE III: FITTING FOR THE Y-FLASH FUNCTIONAL MODEL

Fitting parameter constants	
Parameter	**Value**
Programming	
$a = CR \cdot V_{DS}$	$2.16, 2.4$ (for $4.5V, 5V$)
$b = \frac{K'}{C_{fr}} \cdot P(E_v)$	$2.1 * 10^{-4}$
Erasing	
$a = \frac{A}{C_{dep} t_{ox}}$	$4.643 * 10^{-4}$
$b = CR.V_S + V_{ox}$	0.9531
$c = Bt_{ox}$	0.07

C. *Small-Signal Functional Model*

To show full equivalence to a memristive device, analogy to Ohm's law should be exhibited. (1) can only emulate it with unit-less functions. Thus, a small-signal model is proposed for this purpose. As per (1), a biasing voltage V_{DS} produces a large signal current I_{DS} through the Y-flash. The current flow through the injection transistor is negligible and can be ignored. If the bias point V_{DS} is perturbed with a small-signal voltage v_{ds}, the current i_{DS} varies as:

$$i_{DS} = I_{DS} + i_{ds} = I_{read} e^{\frac{CR \cdot (V_{DS} + v_{ds})}{mV_T}} e^{-\frac{V_{th}}{mV_T}} = I_{DS} e^{\frac{CRv_{ds}}{mV_T}}. \quad (8)$$

Taylor series expansion across V_{DS} yields [13]:

$$I_{DS} + i_{ds} = I_{DS}\left(1 + \frac{CRv_{ds}}{mV_T} + \left(\frac{CRv_{ds}}{mV_T}\right)^2 + \cdots\right). \quad (9)$$

If $CRv_{ds} \ll mV_T$, second- and higher-order terms in (9) are negligible. From large-signal fitting of the model, $\frac{mV_T}{CR} = 144mV$, giving a maximum magnitude (input dynamic range) of $v_{ds} = 30mV$ for $\frac{CRv_{ds}}{mV_T} = 0.208$, which allows to safely ignore the second- and higher-order terms. The I-V characteristics are linearized around the biasing point as:

$$I_{DS} + i_{ds} = I_{DS}\left(1 + \frac{CRv_{ds}}{mV_T}\right), \quad (10)$$

$$g_m = \frac{\Delta I_{DS}}{\Delta V_{DS}} = \frac{I_{DS} CR}{mV_T}, \quad (11)$$

where g_m is defined as the dynamic/incremental conductance for small-signal changes. Therefore, the small signal current expression can be written in an Ohm's law manner:

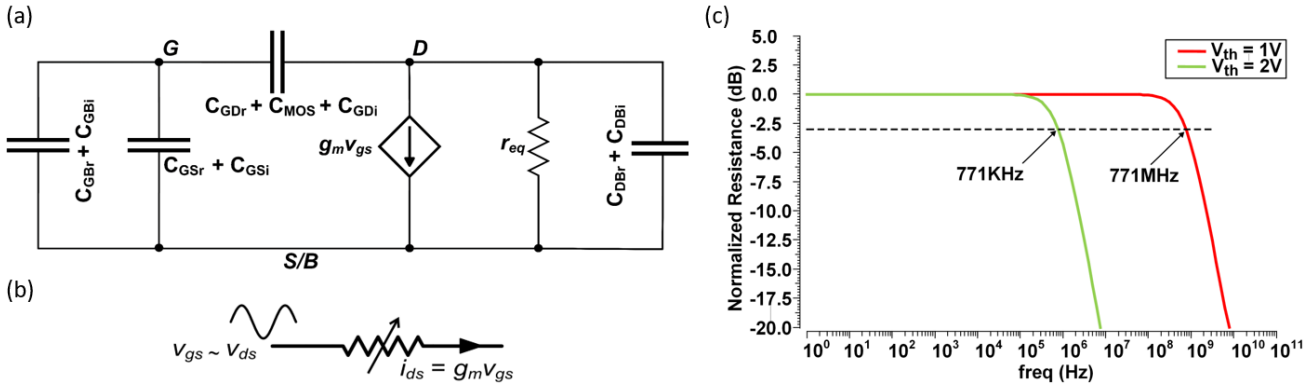

Fig. 4. (a) Subthreshold small-signal schematic of the Y-flash device developed from two parallel-NMOS and parasitic transistors. Since the two NMOSs have the same terminal voltages, they can be folded to develop a single schematic in parallel. Note: $C_{GD} \gg C_{GS}$. (b) An approximate small-signal schematic of the Y-flash (passive) device. (c) 3-dB frequency range, which demonstrates the maximum frequency up to which the Y-flash device works as a linear resistor in response to small-signal changes.

$$i_{ds} = \frac{I_{DS}CR}{mV_T}\, v_{ds} = g_m v_{ds}, \tag{12}$$

$$R_{mem} = \frac{1}{g_m} = \frac{mV_T}{CRI_{read}} e^{\frac{-CR \cdot V_{DS}}{mV_T}} e^{\frac{V_{th}}{mV_T}}, \tag{13}$$

where R_{mem} is defined as the incremental resistance, or the small-signal memristance, with values $\in [145K\Omega : 145M\Omega]$, where $R_{ON} = 145K\Omega$ and $R_{OFF} = 145M\Omega$ (Fig. 3 inset).

A small-signal equivalent schematic considering various capacitors, as listed in Table IV, is developed and illustrated in Fig. 4(a). The maximum frequency of the small-signal input v_{ds} is limited by the coupling capacitor and other capacitors of the two transistors. We performed spectral analysis on the circuit in Fig. 4(a), by biasing the Y-flash in subthreshold and applying a small-signal v_{ds} (within the dynamic range). The gate-source voltage v_{gs} is determined as $v_{gs} = \frac{C_{GD}}{C_{GS}+C_{GD}} v_{ds}$. The total C_{GS}, which includes the parasitic capacitance, is relatively lower than C_{GD}, which is a combination of the parasitic and the additional MOS capacitor. Asymptotically, the gate is floating and shorted to the drain. Therefore, the small-signal schematic can be simplified as a variable resistor, illustrated in Fig. 4(b), ignoring the bulk parasitic capacitance. At higher frequencies, the Y-flash operation is dominated by additional capacitors, transitioning it to behave as a capacitor rather than a resistor. Spectral analysis as illustrated in Fig. 4(c) gives a maximum operating frequency range of $[770KHz : 770MHz]$ for a dynamic range of V_{th} from $[1V : 2V]$.

D. Stochastic Model

Rather than enforcing deterministic dynamics, several applications take advantage of the non-deterministic memory for native stochastic computing, where the required randomness is intrinsic to the device and fundamentally changes its operating principle. Due to the stochastic nature of the injection process [14], the expected statistical spread of the device in $\Delta V_{th} = q\Delta n / C_{fr}$ (q is the electronic charge, and n is the number of injected electrons to obtain V_{th}) is:

$$\sigma_{\Delta V_{th}} = \frac{q\sqrt{\overline{\Delta n}}}{C_{fr}} = \sqrt{\frac{q}{C_{fr}} \overline{\Delta V_{th}}}. \tag{14}$$

Assume n is ruled by a Poisson process [14]. Therefore, the V_{th} statistical spread in Y-Flash after programming/erasing is fitted to Poisson or normal distribution, as measured for DUTs in [9] and characterized in Fig. 5. Notably, the resistive state statistical spread after programming/erasing is log-normally distributed. This finding is consistent with the exponential I-V relationship in subthreshold mode. Assume the average time required for the injection of an electron into the floating gate after the beginning of the program operation is τ_i, which is related to I_i (injection current) by the relation $\tau_i = q/I_i$. Then, the time required for the electron injection (ΔT) is exponentially distributed [14]:

$$P_{\Delta T} = \frac{1}{\tau} e^{-\frac{\Delta T}{\tau}}, \tag{15}$$

where $P_{\Delta T}$ is the probability density function of ΔT. The general model is modified considering the stochastic dynamics to:

$$dV_{th} = f(V_{DS}, V_{th})dt + \sigma_{\Delta V_{th}} P_{\Delta T} \cdot \delta(\Delta T - \tau)|_{\Delta T = dt}, \tag{16}$$

where the first term refers to the deterministic dynamics in (2) and (6b), and the second term refers to the stochastic dynamics [15] dominated by the Poisson process of electron injection/tunneling to/from the floating gate.

TABLE IV: CAPACITANCE SPEC OF THE Y-FLASH DEVICE

Capacitance type	Injection transistor (aF)	Read transistor (aF)
C_{GB}	80.14	164.15
C_{GS}	48.51	49.29
C_{GD}	254.19	746
C_{DB}	320.079	320.737
C_{Gate}	$\sim 2-3\ fF/\mu m^2$	

III. FLOATING-GATE MEMRISTIVE DEVICE CAD

A. Verilog A Model

We developed a Verilog-A model based on the fitted parameters as described in Section II and used it for circuit design and simulations of different applications. This model allows the designer to explore preliminary results based on the Y-flash based circuit topologies, similar to the memristor model discussed in [16]. A link to the model can be found in [19].

The noise characterization methodology for robust trainable circuits is not as stringent as for standard circuit design, where precise noise models of electric devices should be supplied based on mass measurements. Here, non-ideal sources of PVT at CMOS peripheral circuits are calibrated and compensated for by configuring the analog non-volatile memory cells [17]. Moreover, the small-signal differential model makes the device insensitive to temperature and voltage variations.

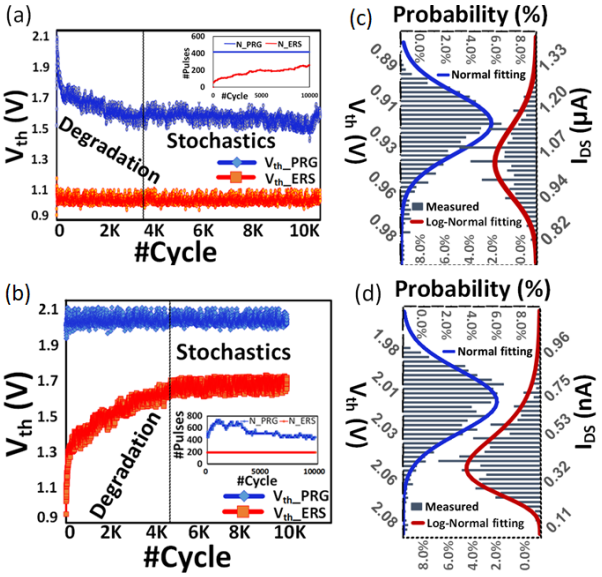

Fig. 5. (a-b) RESET/SET degradation as a result of injected electrons/holes into the floating gate. In the beginning, injection is destructive, and the V_{th} level is decreased/increased under the same number of programming/erasing pulses, while (adaptively) varying the number of pulses to get LRS/HRS, as shown in the insets. (c-d) After the degradation has almost stopped, the measured switching stochastics in [9] are modeled and fitted to normal and log-normal distributions of V_{th} and I_{DS}, respectively in LRS/HRS.

Furthermore, the Y-flash switching variability can be precisely controlled by the programming/erasing procedure until the desired resistive level is achieved. Thus, we allow the user to supply the distribution margins of a statistical parameter added to the code driven by a Poisson process (different processes could be easily added to the code by the user). In addition, we allow the user to supply a parametric spread of each of the device parameters given in the code for behavioral PVT sweep simulations by a graphic user interface (GUI). We believe that this feature adds a high level of flexibility for the user to behaviorally validate the robustness of the design in different extreme cases. As such, it provides the same advantages as Monte Carlo simulations.

B. *Backend Design Tools*

We set up the design environment of the process, including the library setup and the parasitic parameter extraction flow. We also installed the required files for the process design kit (PDK), *e.g.*, TowerJazz 180nm power management technologies (TSLPM). All required libraries, including verification tools, were fully installed. In the installation process, the workload included the integration of the PDK into the pre-installed Cadence environment. The TowerJazz PDK provided the GUI symbol in Virtuoso, the SPICE model, and the configuration protocols for each standard cell. TowerJazz also provides the Assura (Cadence) toolset, which includes multiple functions such as Design Rule Check (DRC), LVS (Layout vs. Schematic, shown in Fig. 6 and Fig. 2(b), respectively), and RC extraction for post-layout verification. By applying these rules appropriately during the test, we validated the design correctness meeting the manufacturing process requirements. The use of standard process facilitates incorporating our models into commercial CAD tools. The most significant effort was

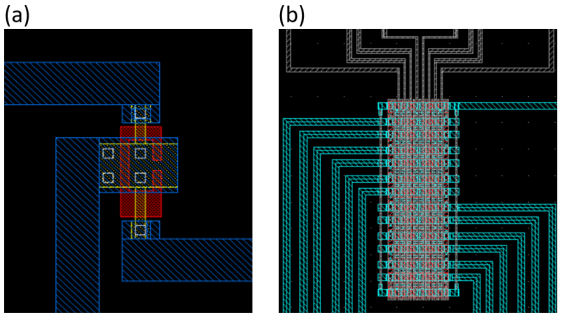

Fig. 6. Layout of a (a) single Y-flash cell (b) 12x8 memristive array. These layouts were checked using DRC and LVS.

spent first on performing LVS using a temporary file holding the Y-flash geometry based on NMOS transistors (Fig. 2(b)), which are easily recognized by the tool. Then, the environment variables were set up and the configuration files were loaded.

IV. Floating-Gate Memristive Arrays

A. *0T1R Memristive Arrays*

Sneak current paths coupled with factors such as half-select, program disturb and crosstalk [18] have been debilitating in the development of large-scale crossbar arrays. Many solutions propose using extra circuit elements to counter these non-idealities. Use of 1T1R (or 1D1R) configurations in particular have enabled significant advances [6] for in-memory computing based on memristive architectures. However, incorporating additional gating elements comes with reduced density and power trade-offs. Thanks to Y-flash's non-linear and asymmetric properties, the non-ideal effects in 0T1R memristive arrays can be circumvented. Thus showing a promising lead for building large-scale dense neuromorphic arrays comprising up to one million cells with high yields [9].

B. *Array Reliability Validation*

In this section, we discuss key reliability issues in crossbars based on the floating-gate memristive arrays. The designer needs to address these issues and validate the array design while developing robust topologies for applications.

Sneak Paths – Fig. 7(a) illustrates various sneak currents while reading a selected device under test (DUT) in a 3x3 array, while keeping the unselected cells floating. The shortest sneak-path within the array that contains three serial exponential devices connected in opposite polarities allows maximal current in the sensed path. Analysis using the developed models was performed to show the negligible effect of sneak paths on the desired sensed current, due to the opposite polarity of the middle device. Furthermore, simulations carried out on 12x8 arrays, considering a worst-case scenario of a fully erased array except for the DUT, determined a $0.5nA$ variation of the typical sensed current in the range $[1nA{:}\,1\mu A]$.

Similarly, Fig.7(b) illustrates sneak paths while programming a DUT. Analysis shows that the saturation current in the injection transistors of unselected cells is suppressed by the asymmetry and non-linearity of the device. In M_{32}, which is part of a sneak path that contains three devices, simulations validate that the voltage drop is kept below erasing voltage ($V_{Src_{inj}D} \ll 8V$) due to the opposite polarity of the device and its high resistance. A worst-case analysis wherein the devices are fully erased (at the threshold) shows a negligible sneak current of $\sim 30pA$.

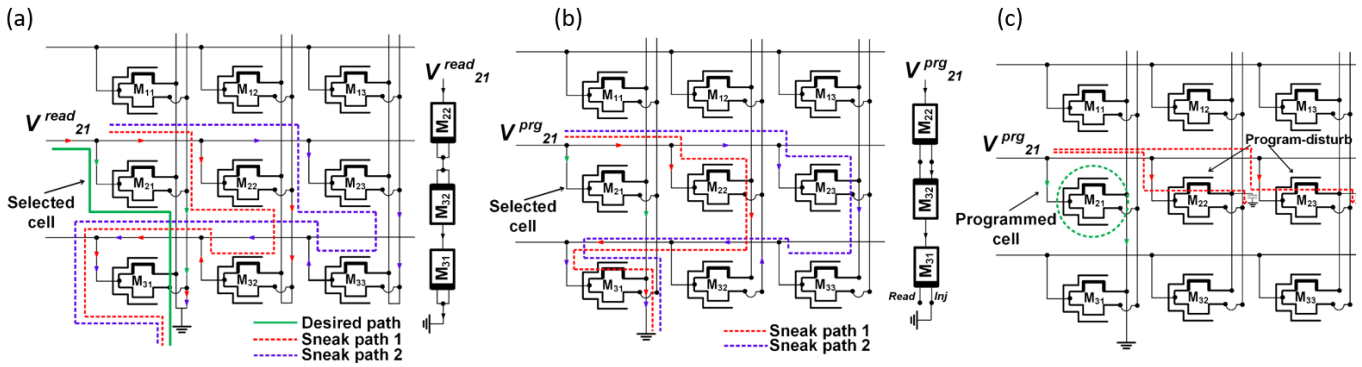

Fig. 7. A 3x3 representative crossbar illustrating sneak currents (a) while reading, and (b) while programming. The insets in (a) and (b) depict the shortest sneak path topology in the array. (c) While programming, the state of the adjacent unselected devices may be disturbed because of substrate leakage current via the injection transistor.

Program Disturb – It is a major error source in Y-flash arrays and can lead to variation in the V_{th} of unselected cells. While programming, the source terminals of the unselected cells are floating. However, as illustrated in Fig. 7(c), current leakages from the source line of the injection transistor to the substrate via a capacitor disturb the unselected devices in the same row. The model allows the designer to incorporate the variation margin and hence analyze its effect on the overall performance.

V. A Neuromorphic-Oriented Framework

Vector-matrix-multiplication (VMM) is a key function in ANNs [7]. By using our framework, we show that VMM can be inherently and robustly implemented within Y-flash memristive arrays, shown in Fig. 7, where the integration of analog resistive memory devices enables efficient analog realization by utilizing Ohm's and Kirchhoff's laws [9]. VMM is represented as $Y_m = \sum_n W_{n,m} X_n$ (where X is an input vector with n rows, and W is the synaptic weight matrix, with m rows and n columns) and implemented via the small-signal I-V operation of a Y-flash memristive array described in (12) by $i_m = \sum_n W_{n,m} v_n$, where $W_{n,m} = 1/R_{mem_{n,m}}$ is the small-signal conductance for a Y-flash memristive device at the n,m node, and v_n is the small-signal input voltage. Furthermore, training simulations are validated on the model-based Y-flash array using the gradient descent algorithm and spike-time-dependent plasticity according to [9].

VMM is an atomic operation executed simultaneously in the entire array using two successive steps (large- and small-signal). While the programming is done sequentially for DUTs in a specific column, the erasing is performed in parallel for entire columns. A differential synapse model, comprising two Y-flash devices, is also used to overcome spatial variations, avoid the power-starved segmented erase, and obtain negative synaptic weights. Correspondingly, the effective weight at node i,j is:

$$W_{ij} = G_{ij}^{+} - G_{ij}^{-}. \qquad (17)$$

Such neuromorphic basics are easily applied by our framework.

VI. Conclusions

In this paper, we present a functional model of a two-terminal floating-gate memristive device fabricated in standard 180nm CMOS technology. The model is implemented in Verilog-A and describes the large-signal operation in the sub-threshold region and the RESET/SET kinetics using CHE/CHH injection. A small-signal schematic was validated for the memristive read operation. Stochastic dynamics were modeled and added to the model along with PVT, enabling Monte-Carlo simulations of the floating-gate memristive device. Layouts of a single-cell and arrays were implemented and checked using DRC followed by LVS and RC extraction. Finally, case-studies based on SPICE simulations were performed to validate the reliable functionality of disturb-free memristive arrays to be used in the future as a building block for training ANNs to perform VMM. We believe that this framework will spur the development of industry-level large-scale neuromorphic circuits.

References

[1] C. Mead, "Neuromorphic electronic systems," in *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1629-1636, Oct. 1990.

[2] J. Mao *et al.*, Artificial neural networks, vol. 350, 1997.

[3] P. Hasler, C. Diorio, B. A. Minch, and C. Mead, "Single transistor learning synapses," in *Proceedings of the International Conference on Neural Information Processing Systems*, Cambridge, MA: MIT Press, Jan. 1994, pp. 817–824.

[4] D. B. Strukov, G. S. Snider, D. R. Stewart, and R. S. Williams, "The missing memristor found.," *Nature*, vol. 453, no. 7191, pp. 80–3, May 2008.

[5] H.-S. P. Wong, and S. Salahuddin, "Memory leads the way to better computing," *Nature Nanotechnology*, vol. 10, no. 3, pp 191–194, Mar. 2015.

[6] G. W. Burr *et al.*, "Experimental demonstration and tolerancing of a large-scale neural network (165,000 synapses), using phase-change memory as the synaptic weight element," in *IEEE International Electron Devices Meeting*, San Francisco, CA, pp. 29.5.1-29.5.4, Dec. 2014.

[7] M. Prezioso *et al.*, "Training and operation of an integrated neuromorphic network based on metal-oxide memristors," *Nature*, vol. 521, no. 7550, pp. 61–64, May 2015.

[8] F. Merrikh-Bayat *et al.*, "High-performance mixed-signal neurocomputing with nanoscale floating-gate memory cell arrays," *IEEE Transactions On Neural Networks And Learning Systems*, vol. 29, no. 10, pp. 4782-4790, Oct. 2018.

[9] L. Danial *et al.*, "*Two-Terminal Floating-Gate Transistors with a Low-Power Memristive Operation Mode for Analogue Neuromorphic Computing*," *Nature Electronics*, in press.

[10] P. K. Ko, C. Hu, and S. Tam, "Lucky-electron model of channel hot-electron injection in MOSFETs," in *IEEE Transactions on Electron Devices*, vol. 31, no. 9, pp. 1116–1125, Sept. 1984.

[11] T. Y. Chan *et al*, "The impact of gate-induced drain leakage current on MOSFET scaling," in *International Electron Devices Meeting*, pp. 718-721, Dec. 1987.

[12] L.O. Chua, "Memristor—the missing circuit element," *IEEE Transactions on Circuit Theory*, vol. 18, no. 5, pp. 507–519, Sept. 1971.

[13] S. M. Sharroush, Y. S. Abdalla, A. A. Dessouki, and E.-S. A. El-Badawy, "Subthreshold MOSFET transistor amplifier operation," in *Proceedings of the International Design and Test Workshop (IDT)*, pp. 1–6, Nov. 2009.

[14] C. M. Compagnoni *et al.*, "Analytical model for the electron-injection statistics during programming of nanoscale NAND flash memories," *IEEE Transactions on Electron Devices*, vol. 55, no. 11, pp. 3192–3199, Nov. 2008.

[15] M. Al-Shedivat, R. Naous, G. Cauwenberghs, and K. N. Salama, "Memristors empower spiking neurons with stochasticity," *IEEE Journal on Emerging and Selected Topics in Circuits And Systems*, vol. 5, no. 2, pp. 242–253, June 2015.

[16] S. Kvatinsky, M. Ramadan, E. G. Friedman, A. Kolodny, "VTEAM: a general model for voltage-controlled memristors," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 62, no. 8, pp. 786–790, May 2015.

[17] Kam-Chuen Jim, C. L. Giles, and B. G. Horne, "An analysis of noise in recurrent neural networks: convergence and generalization," *IEEE Transactions on Neural Networks*, vol. 7, no. 6, pp. 1424–1438, Nov. 1996.

[18] M. Zidan *et al.*, "Memristor-based memory: the sneak paths problem and solutions," *Microelectronics Journal*, vol. 44, no. 2, pp. 176–183, Feb. 2013.

[19] "Y-flash Verilog A model," [Online]. Available: https://github.com/yflash