Device Modeling for Neuromorphic Computing with Dynamic Time Evolution Method

Lining Zhang, Mansun Chan, Dept. of ECE, HKUST
Edge Computing System Needs

- We are entering the age of Tera-byte of memory system
- Computation switch from logic centric to memory centric
Many Memory Modeling Papers Published

Emerging Memory Technologies pp 15-50

NVSIm: A Circuit-Level Performance, Energy, and Area Model for Emerging Non-volatile Memory

Modeling for bipolar resistive memory switching in transition-metal oxides

Ji Hyun Hur, Myoung-Jae Lee, Chang Bum Lee, Young-Bae Kim, and Chang-Jung Kim
Phys. Rev. B 82, 155321 – Published 25 October 2010

Modeling of Set/Reset Operations in NiO-Based Resistive-Switching Memory Devices

Carlo Cagli, Federico Nardi, and Daniele Ielmini, Member, IEEE


A SPICE Compact Model of Metal Oxide Resistive Switching Memory With Variations

Yiming Guan, Member, IEEE, Shimeng Yu, Student Member, IEEE, and H.-S. Philip Wong, Fellow, IEEE

Compact Models for Memristors Based on Charge–Flux Constitutive Relationships

Sangho Shin, Member, IEEE, Kyungmin Kim, Member, IEEE, and Sung-Mo Kang, Fellow, IEEE
No Standard Memory Model in SPICE Yet

Your netlist can include several types of elements:

- Passive elements:
  - Resistors
  - Capacitors
  - Inductors
  - Mutual Inductors

- Active elements:
  - Diodes
  - Bipolar Junction Transistors (BJTs)
  - Junction Field Effect Transistors (JFETs)
  - Metal Semiconductor Field Effect Transistors (MESFETs)
  - Metal Oxide Semiconductor Field Effect Transistors (MOSFETs)

- Transmission lines:
  - W element
  - T element
  - U element

- No standard models in Commercial SPICE or plans from the Compact Model Coalitions
What are we modeling?

- Passive elements
- Sources
- Transistors
- Transmission lines
- Transformers
- ...

Instant parameters for geometry

```
M1 1 0 L=1u W=1u
...

.MODEL XMEM (LEVEL=54 VTO=-0.8V KP=16U
...
```

SPICE input deck

Model parameters for physical behavior

\[ I_x = f_x(V_1(t), V_2(t), \ldots, V_n(t)) \]

\[ = f_x(\bar{V}(t)) \]
Quasi-static Device behavior

\[ I_x(t) = f_x(\bar{V}(t)) + \frac{\partial Q_x(\bar{V}(t))}{\partial t} \]

steady state

charging

Device equations

\[ I_{Dlin} = \frac{\mu WC_{ox}}{L} \left[ (V_G - V_T)V_{DS} - \frac{V_{DS}^2}{2} \right] \]

\[ I_{Dsat} = \frac{\mu WC_{ox}}{2L} (V_G - V_T)^2 \]

Parameter set

\[ V_T = 0.7, \ C_{ox} = \text{xxx}, \ W = \text{xxx}, \ L = \text{xxx} \]

Constant node voltages

\[ V(t) \]

constant

constant currents

\[ I \]

constant

After reaching steady state
Memory Example: PCRAM

- A non-electrical state variable is needed to keep track of degree of crystallization (crystal fraction)

\[ C_f(t_n) = C_f(t_{n-1}) + \Delta C_f \left( \tilde{V}(t_n) \right) \]

\[ \Delta C_f \left( \tilde{V}(t_n) \right) = \int_{t_{n-1}}^{t_n} I_{\text{set}} \left( \tilde{V}(t) \right) dt \]

\[ I_{\text{pcm}} \left( \tilde{V}(t_n) \right) = f \left( \tilde{V}(t_n), C_f(t_n) \right) \]

- Indirectly introduce extra time dependent through internal node \( C_f(t) \) in addition to terminal voltages

The Johnson-Mehl-Avrami (JMA) equation

\[ C_f(t) = 1 - e^{-K \cdot t} \]

\[ K = K_0 e^{-\frac{E_a}{k_B \Delta T + T_{\text{top}}}} \]

\[ I_{\text{SET}} = K_0^{-1} e^{\frac{-E_a}{k_B \Delta T_{\text{top}}}} \]

\[ R_{\text{SET}} = K_0 e^{\frac{E_a}{k_B \Delta T_{\text{top}}}} \]

\[ C_{\text{SET}} = 1 \]

\[ V_{\text{RESET}} = \left[ 1 + e^{-\frac{H_{\text{reset}}}{h}} \right]^{-1} \]

Subcircuit (K. C. Wong et. al. EDSSC 2009)
**Dynamic System Simulation Methodology**

- Need an internal state variable to keep track on the state of the device for **each instance**

  - Instant parameters
  
  \[
  X_1 \ 0 \ L=1u \ W=1u \ \text{Cf}=0.5 \ \text{ract}=0.1 \\
  \ldots
  \]

  - Model parameters
  
  \[
  \text{.MODEL XMEM (Hg=0.4 \ Wbe=0.05 \ ...)}
  \]

  - SPICE input deck

- Some instant parameters are used as state variables to calculate the current change

  \[
  I_{Mem} = f\left((\tilde{V}(t), C_f), (\tilde{P})\right) \Rightarrow I_{Mem} = f\left((\tilde{V}(t)), (\tilde{P}, C_f)\right)
  \]

- More explicit, allow more state variables without speed penalty
Example: FLASH memory simulation

- Indirect versus direct method

\[ I_D(t) = f_P \left( \tilde{V}(t), C_f \left( \tilde{V}(t) \right) \right) + \frac{\partial Q_P}{\partial t} \left( \tilde{V}(t), C_f \left( \tilde{V}(t) \right) \right) \]

\[ I_D(t) = f_{P,C_f} \left( \tilde{V}(t) \right) + \frac{\partial Q_{P,C_f}}{\partial t} \left( \tilde{V}(t) \right) \]

- Change in internal parameter is more explicitly expressed
Defining Initial States

- How to define at the initial values of the internal state variables?
- Users maybe asked to enter the values
  - How to extract these parameters from observable measurement results?
- Let users input some physically measurable quantity like resistance and the state variables are internally extracted?
  - No guarantee it can be achieved
- Assume all cells start from one of the states and a pre-program run is needed if some cells starts with a different states
Reducing Simulation Time

- Many memory models are based on macro-model approach with many internal nodes

- There are many internal nodes

- Example, the temperature model for a PCM cell

K. Sonoda, Renesas Technology
Effects of internal nodes

- Jacobian Matrix in SPICE

\[
\begin{bmatrix}
\frac{\partial I_1}{\partial V_1} & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & \frac{\partial I_T}{\partial \Delta T}
\end{bmatrix}
\begin{bmatrix}
V_1^{i+1} \\
\Delta T^{i+1} \\
V_i
\end{bmatrix}
= -
\begin{bmatrix}
I_1^i & \cdots & 0 \\
I_T^i & \cdots & 0 \\
\frac{\partial I_T}{\partial \Delta T} & \cdots & \frac{\partial I_T}{\partial \Delta T}
\end{bmatrix}
\begin{bmatrix}
V_1^i \\
\Delta T^i \\
V_i
\end{bmatrix}
\]

- Each internal node creates one entry in the matrix
- As these nodes are isolated from other nodes, they directly increase the size of the Jacobian matrix
Eliminating Internal Nodes

- Jacobian Matrix in SPICE

\[
\begin{pmatrix}
\frac{\partial I_1}{\partial V_1} & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & \frac{\partial I_T}{\partial \Delta T} \\
\end{pmatrix}
\begin{bmatrix}
V_1^{i+1} \\
\vdots \\
\Delta T^{i+1} \\
\end{bmatrix}
- \begin{pmatrix}
I_1^i \\
\vdots \\
I_T^i \\
\end{pmatrix}
= -
\begin{pmatrix}
\frac{\partial I_1}{\partial V_1} & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & \frac{\partial I_T}{\partial \Delta T} \\
\end{pmatrix}
\begin{bmatrix}
V_1^i \\
\vdots \\
\Delta T^i \\
\end{bmatrix}
\]

- As the node is loosely coupled, it can be directly evaluated without putting into the Jacobian matrix

\[
\left( \frac{C_{th}}{h} + \frac{1}{R_{th}} \right) \Delta T_i^{n+1} = I_i \left[ \Delta T_i^n \right] V_i^n + \frac{C_{th}}{h} \Delta T_{i-1}^n
\]
Simulation Flow

- The internal node can be eliminated without making any differences.
Convergence issues

- Some models contain switches causing convergence problems.
- A generic smoothing hysteresis function can be used:

\[
S(V) = S(V)s_{low}(V) + (1 - S(V))s_{high}(V)
\]

\[
s_{low}(V) = \frac{1}{1 + \exp\left(-\frac{V - V_{low}}{k}\right)}
\]

\[
s_{high}(V) = \frac{1}{1 + \exp\left(-\frac{V - V_{high}}{k}\right)}
\]

\(S\) is a hysteresis function between 0-1 with \(V_{low}\) and \(V_{high}\) as the two triggering points.
Some Dynamic Behavior

- Some oscillation is observed with static signal input

![Diagram of a memory cell with labeled components](image)

![Graph showing voltage over time for different currents](image)
Possible Explanation

- $E > E_{\text{crit}}$, conductive filament path forms due to generation of excess carriers
- The abrupt change of conductivity causes the voltage to drop
- The high current is sustained by the excess charge and not able to disappear immediately with the voltage drop
- Some partial crystallization takes place in the active region and a lower $E$-field is needed to trigger the next event
Threshold voltage for switching

- Threshold voltage decreases with the increase in crystalized volume with less voltage for filament formation.
- Recovery voltage increases when the filament resistance is smaller than the crystallization region resistance.
Dual Resistance Model

- A dynamic resistor is added in parallel with the active region resistance which has a dynamic dependence on voltage across it.

\[ g_f = \frac{g_f}{1 + \exp\left[\alpha(t - \tau_d)\right]} S_f(V) + \frac{\sigma_f(1-C_f)}{z_{\text{ext}}}[1 - S_f(V)] \]

- To reduce simulation time, the resistance is implemented as

\[ R_{PCM} = R_C + \frac{R_AR_D}{R_A + R_D} \]

Parameter setting:
- \( E_G = 0.15\text{MV/cm} \)
- \( H_G = 100\text{nm} \)
- \( W_{BE} = 30\text{nm} \)
- \( W_G = 400\text{nm} \)
- \( x = 25\text{nm} \)
- \( \tau_0 = 0.2\mu\text{s} \)
Simulation Example

- Simulation of a PCM cell array without speed and convergence problems
# Neural Network and Neuromorphic Hardware

<table>
<thead>
<tr>
<th></th>
<th>Non-Spiking</th>
<th>Spiking</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Premises</strong></td>
<td>Mathematics</td>
<td>Biology</td>
</tr>
<tr>
<td><strong>Data</strong></td>
<td>Big</td>
<td>Sparse</td>
</tr>
<tr>
<td><strong>Knowledge Requisition</strong></td>
<td>Trained</td>
<td>Learned</td>
</tr>
<tr>
<td><strong>Model</strong></td>
<td><img src="image1.png" alt="Diagram" /></td>
<td><img src="image2.png" alt="Diagram" /></td>
</tr>
</tbody>
</table>
Modeling the Analog Characteristics of PCRAM

- Biologically spiking action potentials gradually change the synaptic plasticity
- Consecutive pulses due to SET operation of PCM
- Depression pulses from the RESET of PCM

Resetting with pulse trains


Simulation data
The Spiking Time Dependent Plasticity (STDP)

IBM circuit topology [1]

Simulation Results

Red: pre-synaptic, on STDP transistor M1
Orange: post-synaptic, on PCM terminal

Graphs showing synaptic weight change $\Delta w$ (%), spike timing $\Delta t$ (ms), and time (ms).
SPICE Simulation of Analog Neuron Circuits

Input: image

- Encoding to spike sequence
- LIF mode
- Spiking Neuron networks: first layer
  - STDP mode: Update weights of 1st layer
  - Renewal of membrane potential
  - Fire?
    - YES

Output

Axon Drivers
- 49 Inputs from pre-synaptic neurons

Initial weight

post-synaptic neurons
Simulating the Neuromorphic Computing Designs

- Fire
- STDP BL
- Voltage of Cap
- LIF Pulse
- STDP BL
- \( C_f \) of PCM

neuron2 is affected by lateral inhibition when neuron1 fire.
We demonstrated the possibility to use the PCM model to neuromorphic computing, but still lots of work to do.

Q & A

http://i-mos.org/