Polynomial Metamodel Integrated Verilog-AMS for Memristor-Based Mixed-Signal System Design

Geng Zheng*, Saraju P. Mohanty†, Elias Kougianos‡, and Oghenekarho Okobia§
University of North Texas, Denton, TX 76203, USA, N¶¶§
Email: gengzheng@my.unt.edu*, saraju.mohanty@unt.edu†, elias.kougianos@unt.edu‡, and oo0032@unt.edu§

Abstract—This paper proposes a two-level framework for memristor based mixed-signal design exploration. First, a
Verilog-A memristor model is proposed which is not source-type
dependent and has advantages over existing SPICE memristor
models. It includes the threshold-type behavior and nonlinear
dynamics while retaining memristor parameters that are useful
for circuit design. Second, a Polynomial Metamodel integrated
Verilog-AMS (Verilog-AMS-POM) is proposed to enable fast
circuit-accurate system-level design space exploration of such
circuits. The metamodeling technique is proposed to assist their
design, modeling, and higher-level integration. A memristor
based programmable Schmitt trigger oscillator is presented as
a case study. Polynomial metamodels are created to facilitate
the analysis and verification of the programmable oscillator. The
coefficients of determination of the proposed metamodels demon-
strate excellent fidelity. Verilog-AMS-POM simulation achieves
over 30,000× speedup compared to SPICE simulations.

I. INTRODUCTION

Many novel memristor based integrated circuits and nano-
electronic systems have emerged [1]–[3]. As in any circuit,
memristor based integrated circuit design requires proper
device and circuit models. For system-level integration of
such circuit blocks, efficient block-level representations with
reasonable accuracy are necessary for high-level design space
exploration. This paper proposes a two-level framework to
support memristor based circuit design. First a Verilog-A
memristor model is presented. It includes the memristor
threshold-type behavior and nonlinear dynamics while retaining
the parameters that are useful for circuit design. Second, a
Polynomial integrated Verilog-AMS (Verilog-AMS-POM) is
proposed to support fast circuit-accurate system-level design
space exploration. Verilog-AMS-POM embeds polynomial
metamodels into a Verilog-AMS module to form an efficient
and parametric memristor based circuit block representation.
A programmable memristor oscillator design is proposed as a
case study. The programming scheme proposed in our paper
does not require extra capacitors or negative and high voltages.
Metamodels are used for oscillator circuit model creation and
design analysis.

There have been increasing research interest on designing
digital as well analog systems with memristors as core circuit
elements [1], [4]. The use of memristors for digital systems
is at a very early stage and representing ON and OFF for
digital systems with High-Resistance and Low-Resistance of
memristor needs more research. In analog systems, memristors
are employed to enable circuit programmability.

The novel contributions of this paper are summarized as:
1) A Verilog-A model which includes the threshold and
nonlinear dopant drift for titanium dioxide (TiO2) thin-film
memristors is presented.
2) A memristor programmable Schmitt trigger oscillator
design and a programming scheme are proposed.
3) Polynomial metamodels for the proposed memristor-based
programmable oscillator are presented for fast simulation.
4) A Verilog-AMS model for the programmable oscillator
embedding the polynomial metamodels.

The rest of this paper is organized as follows: Section II
presents the proposed memristor device model in Verilog-A.
Section III describes a memristor based programmable oscillator
design. Section IV presents the polynomial metamodels for the
programmable oscillator. Section V concludes the paper.

II. VERILOG-A DEVICE-LEVEL MEMRISTOR MODEL

Several models have been proposed to address the highly
nonlinear dynamics of memristors [5], [6]. The coupled
variable-resistor model of a titanium dioxide (TiO2) thin film
memristor is of the following form [7]:

\[
V_M = \frac{\mu_v}{D^2} I_M, \quad \frac{dI_M}{dt} = \frac{1}{R_{on} R_{off}} (1-x) I_M.
\]

where \( x \in [0, 1] \) is an internal state variable, \( \mu_v \) is the dopant
mobility, and \( D \) is the semiconductor film thickness. This
model does not include the threshold-type behavior observed
in fabricated devices [8]. One of the major limitations of
the memristor modeling approach arises from its boundary
conditions. When the value of the state variable \( x \) is pushed to
its boundaries (0 or 1) by an external source, this value should
stay unchanged (0 or 1) until the applied voltage/current
changes its polarity. A window function may prevent the state
variable from returning from its boundaries even after the
polarity of the applied source has changed [6]). This issue
is termed as hard-switching or terminal-state problem [6].
Moreover, it appears that the nonlinearity generated by the
window function does not conform to the characteristics of the
available fabricated device. Non-inclusion of an explicit
threshold further limits the use of existing models.

Polynomial Metamodel Integrated Verilog-AMS for Memristor-Based Mixed-Signal System Design

Geng Zheng*, Saraju P. Mohanty†, Elias Kougianos‡, and Oghenekarho Okobia§
University of North Texas, Denton, TX 76203, USA, N¶¶§
Email: gengzheng@my.unt.edu*, saraju.mohanty@unt.edu†, elias.kougianos@unt.edu‡, and oo0032@unt.edu§

Abstract—This paper proposes a two-level framework for memristor based mixed-signal design exploration. First, a
Verilog-A memristor model is proposed which is not source-type
dependent and has advantages over existing SPICE memristor
models. It includes the threshold-type behavior and nonlinear
dynamics while retaining memristor parameters that are useful
for circuit design. Second, a Polynomial Metamodel integrated
Verilog-AMS (Verilog-AMS-POM) is proposed to enable fast
circuit-accurate system-level design space exploration of such
circuits. The metamodeling technique is proposed to assist their
design, modeling, and higher-level integration. A memristor
based programmable Schmitt trigger oscillator is presented as
a case study. Polynomial metamodels are created to facilitate
the analysis and verification of the programmable oscillator. The
coefficients of determination of the proposed metamodels demon-
strate excellent fidelity. Verilog-AMS-POM simulation achieves
over 30,000× speedup compared to SPICE simulations.

I. INTRODUCTION

Many novel memristor based integrated circuits and nano-
electronic systems have emerged [1]–[3]. As in any circuit,
memristor based integrated circuit design requires proper
device and circuit models. For system-level integration of
such circuit blocks, efficient block-level representations with
reasonable accuracy are necessary for high-level design space
exploration. This paper proposes a two-level framework to
support memristor based circuit design. First a Verilog-A
memristor model is presented. It includes the memristor
threshold-type behavior and nonlinear dynamics while retaining
the parameters that are useful for circuit design. Second, a
Polynomial integrated Verilog-AMS (Verilog-AMS-POM) is
proposed to support fast circuit-accurate system-level design
space exploration. Verilog-AMS-POM embeds polynomial
metamodels into a Verilog-AMS module to form an efficient
and parametric memristor based circuit block representation.
A programmable memristor oscillator design is proposed as a
case study. The programming scheme proposed in our paper
does not require extra capacitors or negative and high voltages.
Metamodels are used for oscillator circuit model creation and
design analysis.

There have been increasing research interest on designing
digital as well analog systems with memristors as core circuit
elements [1], [4]. The use of memristors for digital systems
is at a very early stage and representing ON and OFF for
digital systems with High-Resistance and Low-Resistance of
memristor needs more research. In analog systems, memristors
are employed to enable circuit programmability.

The novel contributions of this paper are summarized as:
1) A Verilog-A model which includes the threshold and
nonlinear dopant drift for titanium dioxide (TiO2) thin-film
memristors is presented.
2) A memristor programmable Schmitt trigger oscillator
design and a programming scheme are proposed.
3) Polynomial metamodels for the proposed memristor-based
programmable oscillator are presented for fast simulation.
4) A Verilog-AMS model for the programmable oscillator
embedding the polynomial metamodels.

The rest of this paper is organized as follows: Section II
presents the proposed memristor device model in Verilog-A.
Section III describes a memristor based programmable oscillator
design. Section IV presents the polynomial metamodels for the
programmable oscillator. Section V concludes the paper.

II. VERILOG-A DEVICE-LEVEL MEMRISTOR MODEL

Several models have been proposed to address the highly
nonlinear dynamics of memristors [5], [6]. The coupled
variable-resistor model of a titanium dioxide (TiO2) thin film
memristor is of the following form [7]:

\[
V_M = \frac{R_{on} x + R_{off}(1-x)}{M} I_M, \quad \frac{dI_M}{dt} = \frac{1}{R_{on} R_{off}} (1-x) I_M,
\]

where \( x \in [0, 1] \) is an internal state variable, \( \mu_v \) is the dopant
mobility, and \( D \) is the semiconductor film thickness. This
model does not include the threshold-type behavior observed
in fabricated devices [8]. One of the major limitations of
the memristor modeling approach arises from its boundary
conditions. When the value of the state variable \( x \) is pushed to
its boundaries (0 or 1) by an external source, this value should
stay unchanged (0 or 1) until the applied voltage/current
changes its polarity. A window function may prevent the state
variable from returning from its boundaries even after the
polarity of the applied source has changed [6]). This issue
is termed as hard-switching or terminal-state problem [6].
Moreover, it appears that the nonlinearity generated by the
window function does not conform to the characteristics of the
available fabricated device. Non-inclusion of an explicit
threshold further limits the use of existing models.
A. The Proposed Memristor Model

We present a coupled variable-resistor model to take into account the threshold-type behavior. The modification is based on the exponential dopant drift model [9] for thin-film memristors where the drift velocity was characterized using the following expression:

\[ u \approx \begin{cases} \mu_v E & \text{if } E \ll E_o, \\ \mu_v E_o e^{E/E_o} & \text{if } E \sim E_o, \end{cases} \tag{3} \]

where \( E \) is the electric field across the device and \( E_o \) is the characteristic high field that introduces the nonlinear effect. A variable-mobility model was proposed in [1] for \( \mu_v \), in Eqn. (2) to incorporate the memristor threshold based on this nonlinear drift equation. \( \mu_v \), however, should be kept the same and the rest of Eqn. (2) should be modified. Assuming a uniform field, the memristor threshold \( V_o \) is related to \( E_o \) through \( E_o = V_o/D \). There can be two threshold values, \( V_p \) and \( V_n \), for different polarities of \( V_M \). Using Eqn. (3), Eqn. (2) can be extended to the following model:

\[ \frac{dx}{dt} = \begin{cases} \mu_v V_p e^{R_{on} I_p} & \text{if } V_M \geq V_p, \\ \mu_v V_n e^{R_{on} I_p} & \text{if } V_M \leq V_n, \\ \mu_v R_{on} I_M & \text{otherwise}, \end{cases} \tag{4} \]

The \( I - V \) relation in Eqn. (1) is still valid. This way the memristor nonlinear dynamic is included in the new model while parameters such as \( D, R_{on}, R_{off} \) are kept.

B. Proposed Verilog-A memristor model

SPICE memristor models are source-type dependent, i.e. voltage-controlled based and cannot be driven by a current source and vice versa. This restricts their application. In contrast, a Verilog-A implementation is not restricted and is more suitable for mixed-signal simulation. The memristor device-level model implementation in this paper is based on the coupled variable-resistor relation from Eqn. (1) and the memristor nonlinear dynamic described in Eqn. (4). They can be described in physical modeling languages such as Verilog-A and MATLAB Simulink/Simscape. Fig. 1 shows the time-domain simulation for both implementations. A Verilog-A model used in this paper for circuit-level simulation. The memristor dynamic is implemented in integral form. A variable \( \text{integ} \) has been used to control the integral operation such that the boundary conditions are satisfied and the hard-switching problem is avoided.

III. MEMRISTOR-BASED PROGRAMMABLE OSCILLATOR

A. Programmable Schmitt trigger oscillator

Schmitt trigger oscillators are used in many applications such as capacitive and inductive sensing [10] and pressure sensing [11]. They are also used in implantable devices, e.g., in [12]. An investigation of three types of digitally controlled oscillators—differential ring oscillators, Source-coupled multivibrator, and Schmitt trigger oscillator—for a phase-locked loop was conducted in [13]. It demonstrated that the Schmitt

![Fig. 1: Time-domain simulation of memristor. A 2-V, 40-Hz sine wave is applied to a memristor: \( R_{on} = 1 \, \text{k} \Omega, R_{off} = 10 \, \text{k} \Omega, \mu_v = 10^{-14} \, \text{m}^2/\text{Vs}, V_p = 1.7 \, \text{V}, V_n = -1.7 \, \text{V}. \)](image)

![Fig. 2: The conventional Schmitt trigger oscillator and the proposed one with added programmability.](image)
The width \((W)\) for transistors M1–M4 for generating \(I_p\) is 4 \(\mu\text{m}\). Transistors M13 and M14 for creating hysteresis have a \(W\) of 2.5 \(\mu\text{m}\) and 1.5 \(\mu\text{m}\), respectively. For all other PMOS transistors, \(W = 2 \mu\text{m}\); for all other NMOS transistors, \(W = 1 \mu\text{m}\). The programming current \(I_p\) is set to 100 \(\mu\text{A}\) and \(C = 200 \text{fF}\). The memristor \(R_{on}\) and \(R_{off}\) are 10 k\(\Omega\) and 100 k\(\Omega\), respectively. Table I lists the resultant characteristics of the memristor based programmable oscillator at three states.

It can be seen from Table I that replacing the regular resistor with the memristor does not lead to noticeable increase in power consumption. Device noise is the major source of the simulated jitter, although the device noise of the memristor was not included in the simulations. The current flow through the memristor can cause slight fluctuation on the memristor state. This temporal variation produces additional jitter to the programmable oscillator. For a large resistance value, \(R = 100 \text{k}\(\Omega\)\), the resistor thermal noise is dominant and thus the conventional Schmitt trigger oscillator exhibits more jitter. For \(R = 55 \text{k}\(\Omega\)\), the resistor thermal noise is reduced and the memristor state variation becomes a larger jitter source. For \(R = 10 \text{k}\(\Omega\)\), the resistor thermal noise is further reduced. However, the jitter contributed by the memristor state variation is reduced by a greater amount since the memristor state is insensitive to high-frequency signals. Therefore the programmable oscillator exhibits less jitter than a CMOS only realization. The jitter exhibited by both oscillators is expected to be comparable if the memristor device noise is added to the simulation.

IV. METAMODELS FOR THE MEMRISTOR BASED PROGRAMMABLE OSCILLATOR

Metamodels are surrogate models that attempt to approximate the characteristic response surfaces of systems. A polynomial metamodel (POM) has the following form [14]:

\[
    f_{POM}(\mathbf{v}) = \sum_{i=0}^{N_B-1} \beta_i \prod_{j=0}^{N_d-1} v_{ij}^{p_{ij}},
\]

where \(\mathbf{v} = \{v_1, v_2, \ldots, v_{N_D}\} \) are circuit design variables, \(N_B\) is the number of basis functions of this POM, \(\beta_i\) is the coefficient of the \(i\)th basis function, \(p_{ij}\) is the power term for the \(j\)th design variable in the \(i\)th basis function, and \(N_D\) is the number of design variables. The signal propagation delay of a simple RC circuit can be approximated as \(t_d = 0.7RC\). Using metamodeling, a 1st order POM describes the delay as:

\[
    t_d, POM(R, C) = \beta_0 R C^0 + \beta_1 R^1 C^1 + \beta_2 R^2 C^0 + \beta_3 R^1 C^1.
\]

It has two design variables \((N_D = 2)\) and four basis functions \((N_B = 4)\). The coefficients \(\beta_i\) are determined by sampling the circuit design space and then fitting the POM to the sampled responses by tuning the coefficient values.

<table>
<thead>
<tr>
<th>Variable</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>(t_{set})</td>
<td>0 ms</td>
<td>100 ms</td>
</tr>
<tr>
<td>(I_p)</td>
<td>80 (\mu\text{A})</td>
<td>120 (\mu\text{A})</td>
</tr>
<tr>
<td>(R_{on})</td>
<td>8 (\text{k}(\Omega))</td>
<td>12 (\text{k}(\Omega))</td>
</tr>
<tr>
<td>(R_{off})</td>
<td>80 (\text{k}(\Omega))</td>
<td>120 (\text{k}(\Omega))</td>
</tr>
<tr>
<td>(W_P)</td>
<td>1.6 (\mu\text{m})</td>
<td>2.4 (\mu\text{m})</td>
</tr>
<tr>
<td>(W_N)</td>
<td>0.8 (\mu\text{m})</td>
<td>1.2 (\mu\text{m})</td>
</tr>
</tbody>
</table>

\(x_{POM}(\mathbf{v}) = x_{POM}(t_{set}, I_p, R_{on}, R_{off}, W_P, W_N)\)

In this paper, two POMs were created. The first POM, \(x_{POM}(\mathbf{v})\), approximates the response surface of the memristor state \(x\). A 2nd order POM was created from 500 SPICE-simulated samples with Latin hypercube sampling (LHS). The six included variables \((N_D = 6)\) and their ranges for this POM are listed in Table II. To visualize the effect of \(t_{set}\) and \(I_p\) on memristor state setting, a response surface of the programmed state \(x\) as a function of \(t_{set}\) and \(I_p\) is constructed in Fig. 5 using \(x_{POM}(\mathbf{v})\) while other variables are fixed. The same response surface constructed using SPICE simulations is also shown as the true response surface for comparison. It is also worth mentioning that although only 6 variables were created, the circuit nonidealities such as the parasitics of the programming switches M5–M8 have been included in the POM during the LHS process through SPICE simulation. Hence, it is feasible to use a POM to improve the programming accuracy. This can be done by using a POM, \(x_{set, POM}(x, I_p, R_{on}, R_{off}, W_P, W_N)\), to compute the required value for \(t_{set}\) for a given \(x\). The computed \(t_{set}\) value automatically includes the compensation.
for the circuit nonidealities, \( t_{set, POM}(v) \) can be coded into the programming unit so the memristor state is software-defined.

The second POM, \( f_o, POM(v) \), approximates the response surface of the oscillator output frequency but is not shown here due to space restrictions. This POM is useful for fast estimation of \( f_o \) variation due to various effects and for construction of a model for high-level design simulation. The 500 simulation samples in the first POM were re-used to create a 4th order \( f_o, POM(v) \). For a 20% variation in \( W_P \) and \( W_N \), the output frequency \( f_o \) can vary from 54 MHz to 66 MHz.

In order to assess the POM accuracy, a verification set of 500 SPICE-simulated samples were generated. The values predicted by the POMs are compared with the verification samples. Metrics adopted to assess the POM accuracy include: coefficient of determination \( (R^2) \), Relative maximum absolute error \( (RMAE) \), and RMSE.

In order to evaluate the speedup achieved by employing Verilog-AMS-POM, the runtimes for the Verilog-AMS-POM and SPICE simulations with various memristor state variable values are compared in Table III. For each simulation run, the simulation time was set to be equal to 1000 oscillation cycles. Over 30000× speedups were observed. The oscillator Verilog-AMS-POM bypasses the complexity of the memristor and the nanometer transistor models while including the circuit nonidealities through polynomial metamodels. With the parameterization inherited in the metamodels, the Verilog-AMS-POM is not only suited for verification tasks but also for system-level design optimization. Verilog-AMS-POM in particular can make the mixed-signal system simulation, verification, and optimization faster than existing design methods. This in turn will reduce the design cycle time, design cost and improve chip yield.

### Table III: Runtime of memristor-based oscillator.

<table>
<thead>
<tr>
<th>( x )</th>
<th>Verilog-AMS-POM</th>
<th>SPICE</th>
<th>Speedup</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>10.72 ms</td>
<td>360.31 s</td>
<td>33704×</td>
</tr>
<tr>
<td>0.3</td>
<td>11.60 ms</td>
<td>335.96 s</td>
<td>28962×</td>
</tr>
<tr>
<td>0.5</td>
<td>11.77 ms</td>
<td>349.10 s</td>
<td>29660×</td>
</tr>
<tr>
<td>0.7</td>
<td>11.73 ms</td>
<td>348.08 s</td>
<td>29674×</td>
</tr>
<tr>
<td>0.9</td>
<td>10.91 ms</td>
<td>355.97 s</td>
<td>32628×</td>
</tr>
</tbody>
</table>

### V. Conclusions

A memristive device model that relates the memristor physical operation and its circuit-oriented parameters has been presented. This model is implemented in Verilog-A for higher efficiency and flexibility. Polynomial metamodels have been created for the proposed memristor based programmable oscillator. They are accurate and efficient tools for analyzing memristor based circuits. Verilog-AMS-POM construction of the memristor based oscillator has been demonstrated. Verilog-AMS-POM boosts the simulation speed of the memristor based circuit. It is thus a prominent candidate to support system-level integration, verification, and design exploration.

### References


