PUF-ROMS: Accelerating and Optimizing PUF Design using SPICE and Reduced Order Modeling

Ian Wilcox, *Member, IEEE*, Jenilee Jao, *Student Member, IEEE*, Jim Plusquellic, *Senior Member, IEEE*, Biliana Paskaleva, *Member, IEEE*, Pavel Bochev

***Abstract*—We formulate a non-intrusive model order reduction (MOR) framework, called PUF-ROMS, to accelerate and optimize the design and analysis of physical unclonable functions (PUFs). The primary goals of PUF-ROMS are to estimate entropy and temperature-voltage noise (TV-noise) of circuit structures used in the PUF's design in an accelerated evaluation environment to enable designers to explore different architectures with the goal of maximizing entropy and minimizing the adverse impact of TV-noise on accessing this entropy. PUF-ROMS starts with the development of reduced order models (ROMs) for the logic cell primitives used in the PUF circuit structure. These models are based on the canonical Hammerstein model architecture and are trained using SPICE transistor-level simulation data of the cell primitives collected *offline*. The cell primitive ROMs are then used in SPICE system-level Monte Carlo (MC) simulations to enable efficient exploration of the PUF design space. PUF-ROMS is developed and demonstrated using an IBM 90nm PDK, a standard cell library and hardware data collected from a variant of the Arbiter PUF. Our evaluation shows that delay PUF designs can nearly double the level of entropy by using a specific subset of the standard cells, and by instantiating them with transistor options normally used in low-power design. The performance, memory requirements and effectiveness of the PUF-ROMS evaluation methodology is compared with an alternative SPICE-level strategy. The assessment accounts for the time taken to calibrate the standard cell ROM models to SPICE-level simulation results, where calibration utilizes Monte Carlo simulations of local device mismatch and simulations using process-voltage-temperature (PVT) corner models.**

***Index Terms*—Physical Unclonable Functions, Hardware Security, Reduced Order Modeling, Within-die variations**

I. Introduction

PHYSICAL Unclonable Functions (PUFs) are hardware security functions capable of generating encryption keys and random bitstrings for authentication protocols. In microelectronic implementations of PUFs, the sources of randomness (entropy) leveraged by PUFs are variations introduced by limitations in the manufacturing process, which result in small random differences among identical devices, e.g., in the delay of signal paths, transistor leakage current, and the capacitance and resistance of conductors. PUFs are designed to measure and digitize these small differences to produce a sequence of binary 0's and 1's. The ability of the PUF to generate and reproduce bitstrings of high statistical quality is related to the randomness and magnitude of the electrical signals being measured by the PUF, as well as their stability over adverse environmental conditions.

The PUF architecture defines a circuit structure that is used as the source of randomness. In nearly all cases, the circuit structure is composed of a set of identically designed primitives. One of the most popular primitives is the Ring Oscillator (RO) [1], which consists of an odd sequence of inverters connected in a cyclic circuit structure. The first inverter is usually replaced with a NAND gate to enable external control over when the RO is enabled, and the oscillation frequency is typically measured using an on-chip counter. Manufacturing imperfections cause the frequency of identical instances of the RO to vary randomly.

Although the RO is relatively simple and straightforward to describe, there are many possible ways to construct it in a layout. Given that the security properties of the RO's bitstrings are significantly impacted by choices made regarding the types and sizes of the transistors used for the logic gates that define the ring, exploration and evaluation of different layout configurations is essential for the design of strong PUFs.

However, due to the small impact of within-die variations, and the difficulty of capturing them in a model, exploration of the PUF design space is typically performed by carrying out large numbers of Monte Carlo SPICE-level simulations across process, voltage and temperature (PVT) conditions. Given the simplicity of the RO, this type of evaluation is possible, albeit time-consuming, and is, in fact, carried out by engineers for purposes of yield learning and process characterization. Unfortunately, for more complicated PUF architectures, e.g., the glitch PUF [2] and SiRF PUF [3], which use large area test structures as the source of entropy, this type of full circuit SPICE-level Monte Carlo exploration for maximizing entropy and minimizing TV-noise effects is not practical.

Model order reduction (MOR) techniques provide an opportunity to reduce the computational burden of full order model (FOM) SPICE-level Monte Carlo simulations, by replacing transistor-level circuit descriptions by computationally efficient Reduced Order Models (ROMs).

In this paper we develop and demonstrate a non-intrusive behavioral MOR methodology for rapid assessment of PUF architectures, which is able to capture signal characteristics at the layout level of abstraction in a FOM. Our approach, termed PUF-ROMS, starts with the development of accurate ROMs for standard cells primitives, based on the canonical

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration contract number DE-NA0003525.

Hammerstein model architecture. These ROM primitives are identified using training data gathered *offline* by transistor-level SPICE-level simulations. The ROMs are then used *online* to accelerate full circuit evaluations of entropy and TV-noise levels under different configurations. Our cell primitive ROMs are implemented in Verilog-A [4] and enable speed ups to a maximum factor of 20x, relative to transistor-level SPICE simulations. At the same time, these ROMs are accurate enough to provide assessments of entropy and TV-noise of the PUF circuit layout that are consistent with SPICE transistor-level simulations. Our key contributions are as follows:

- Development of the PUF-ROMS approach, using an IBM 90nm PDK and a design fabricated by MOSIS, and demonstration of its ability to accurately represent entropy and TV-noise effects.
- A performance analysis of the PUF-ROMS approach, relative to a SPICE transistor-level exploration of a PUF design space.
- A cell variability analysis is proposed which can be used to determine which foundry-specific transistor options yield the best trade-off of entropy and TV-noise resilience.

The remainder of this paper is organized as follows. Section II reviews the relevant MOR work, motivates PUF-ROMS and outlines the basic idea of the approach. In Section III we present background on the research related to PUF structures. Sections IV and V investigate standard cell applications of the ROM models, which includes analysis related to device mismatch and environmental conditions. Section VI explores standard cell variants and the impact of environmental noise. Section VII draws a comparison between the ROM developed in this work and the Level 1 MOSFET model calibrated to the IBM 90nm technology node. Section VIII demonstrates using ROM to explore Arbiter PUF options and provides select results for PUF acceleration. Section IX highlights the performance results for the Arbiter PUF circuit assembled using the BSIM4v4.3, Level 1 and ROM options for circuit simulation. Section X demonstrates the ROM developed in this work to a PUF circuit described in literature.

II. Motivation and Background

PUF designs can be improved by modeling the variable response of digital circuits due to device variations. Device variability modeling can also be instrumental in understanding bias sources in SRAM PUFs [5], which are detrimental to their performance. For example, [6] uses SPICE templates to enable a series of simulations to gather and evaluate results for variations of a TCO PUF [7] arranged in different M × N arrays. This work demonstrates optimization of PUF metrics by selecting the optimal structure for the design using a series of transistor-level SPICE simulations with Monte Carlo driven randomness for MOSFET parameters.

However, multi-query SPICE simulations for Monte Carlo analysis based on transistor-level full order circuit models (FOMs) can quickly become untenable, both in terms of setup and computation, as the circuit size increases. Indeed, such FOMs are built from compact device models using the modified nodal analysis (MNA) technique [?]. MNA applies Kirchhoff's current law (KCL) at each circuit node to combine compact device models into circuit FOMs given by systems of differential algebraic equations (DAEs) with dimensions proportional to the number of devices in the circuit. Numerical solution of these systems can be challenging and time consuming for larger circuits.

One option to improve circuit solver performance is to use domain decomposition [?], [?], or FastSpice techniques [8], both of which are based on partitioning of the circuit into blocks to increase concurrency. However, the former are primarily designed for large-scale linear circuits such as power grids, whereas the latter sacrifice accuracy for speed when compared with SPICE FOM simulations and are better suited for analyzing worst case statistics of large circuits. This makes such techniques less applicable to PUF modeling which involves nonlinear circuits and requires FOM-like accuracy to capture subtle within-die variations.

A second option is to consider a model order reduction (MOR) approach, which replaces the transistor-level circuit FOM by a computationally efficient reduced order model (ROM) that is also sufficiently accurate for the desired analysis tasks. MOR is a broad concept that includes techniques ranging from intrusive projection-based ROMs [?], to non-intrusive operator learning methods such as Dynamic Mode Decomposition (DMD) [9], and operator inference [?]. The breadth of this topic makes a comprehensive review of MOR impossible within the limited space of this paper. Instead we focus on MOR techniques relevant to circuit simulations and use these examples to motivate our approach.

One of the earliest such techniques is *macro-modeling* (MM), [?] which aims to obtain a simplified circuit model that approximates the functionality of the full circuit. Macro-modeling is largely a heuristic process that relies on designer's intuition and requires intimate understanding of the circuit operation. Development of efficient circuit ROMs can be significantly simplified if the circuit's FOM is a linear time invariant system (LTI). MOR for such circuits exploits the fact that an LTI system is completely characterized by its transfer function (TF) and so, a ROM can be constructed by approximating this function. The latter is usually obtained by matching the moments of the TF at the DC operating point, either explicitly [?], or implicitly as in the PRIMA (passive reduced-order macro-models for linear RLC systems) algorithm [?]. Moment matching can be extended to circuits whose FOMs have polynomial nonlinearities by using Volterra theory, with the nonlinear model order reduction method (NORM) [?], [?] being one of the typical examples. A further extension to the larger class of quadratic-linearizable nonlinear systems is provided by QLMOR (model order reduction via quadratic-linear systems) approach [?].

The approaches described above are examples of *structure-exploiting* MOR because they leverage information about the mathematical structure of the FOM to construct the ROMs. In contrast, projection-based MOR obtains the ROM by projecting the FOM onto a reduced order basis, derived from a collection of snapshots by using Proper Orthogonal Decomposition (POD) [?]. POD-based ROM has been applied to circuits of interest to us, such as ring oscillators (RO), in [10]. Projection-based MOR can also be combined with structure exploiting approaches by first projecting the FOM onto a reduced basis and then performing moment matching using the resulting ROM; see, e.g., [?]. This approach has been used to develop a ROM for RO circuits in [11].

In this work we are interested in a MOR approach that can reliably capture process variations essential for characterization of the PUF design space. Because the circuit structures of interest to us are nonlinear, the approach should also be applicable to general nonlinear systems. These requirements rule out from the onset structure-exploiting ROMs designed for LTI or weakly nonlinear systems. While QLMOR can handle more general systems, it does so by embedding them into larger DAEs with possible increase in the DAE index, which can lead to numerical complications. On the other hand, POD-based MOR are well-suited for PUF structures, particularly delay-based PUFs, which use a set of identically designed circuit structures. These techniques scale well with RO circuits and have been demonstrated for RO circuits with 10000 stages, far longer than typical PUF designs; see [10]. However, individually tuned values for the resistance and capacitance chosen in that work may not be well-suited for modeling inherent PVT variations essential in PUF assessment nor would they be easily adaptable to other technology nodes.

In this paper we propose an alternative *behavioral* MOR approach that starts with a selection of a suitable model architecture for the ROM. This architecture should be expressive enough to capture behavior details relevant to the exploration of the PUF design space, yet be simple enough to yield computationally efficient models. In this work we choose to work with the canonical Hammerstein model architecture comprising a static nonlinear block in series with a dynamic LTI block. This choice is motivated by the fact that the nonlinear behavior of the cell primitives can be accurately represented by a nonlinear DC model with the LTI block providing the necessary "corrections" that capture transient phenomena. An early example of utilization of Hammerstein architectures for circuit modeling is [12]. More recent work [?] uses a similar approach to develop ROMs for differential amplifier and operational amplifier circuits [?].

The proposed approach, that we call PUF-ROMS, is an example of a non-intrusive MOR that can be inferred solely from input-output data. This is advantageous when the circuit FOM, necessary for projection-based and structure-exploiting MOR is not available due to, e.g., proprietary technologies. Also, in contrast to traditional macro-modeling our approach does not rely on SME analysis of the circuit's design, and only requires understanding of its characteristic behaviors.

To conclude this section let us mention that recent approaches based on neural network (NN) regression have attracted attention for circuit modeling. For example, neural networks trained on the behavior of BSIM-CMG simulations are explored in [13] to achieve faster turn around time, improved numerical stability, and increased capability for incorporating process variation effects, when compared to the look-up table (LUT) based models from [14]. [15] expand on the work of [13] to include local process variations in the NN models to achieve $>$ 10x reduction in simulation time for relevant circuits including 17-stage Ring Oscillator and SRAM cell with 1-2% error. Similar to the ROM in our work, the NN models from [15] are applied using Verilog-A. However, evaluation of NN models requires extensive use of matrix multiplications, which complicates their implementation in Verilog-A. In contrast, our approach can be easily implemented through the standard functionality in Verilog-A.

The impact of V_{th} variations on critical path delays is studied by [16] with an emphasis on the impact to yield. Statistical Static Timing Analysis [17] offers an approach for generating a distribution of the timing results for a series of paths given device and interconnect variations, and, with modifications, may be applicable to analysis of delay PUFs.

Methods for modeling variability in digital circuits and standard cells is typically applied to reduce the impact of variation on circuit designs while increasing the tolerance to local variations [18]. In contrast, our work focuses on the development of a ROM and an analysis of variability in standard cells as a means of maximizing the path delay variations (entropy) of a given circuit structure.

III. PUF Structures

PUF circuit architectures are designed to leverage the natural, random variations that occur in integrated circuit (IC) structures. Examples include PUFs that leverage variations in path delays [19]–[21], MOSFET threshold voltage variations [22], and digital memory power-up behavior [23]–[26]. These circuits are designed to reliably produce a unique sequence of random bits, denoted as the bitstring, for cryptographic applications. Additional support circuitry is included in some cases to enable on-chip measurements and digitized representations of, e.g., delay, which is then used in data post-processing operations. Error Correction Code circuitry [27] enables robust key generation across a range of environmental variations including temperature, aging and supply voltage. Alternatively, error-avoidance-based helper data can be generated to improve reliability [21].

In this work, we focus on modeling to support design optimization and acceleration of path delay PUFs. In particular, the modeling will be applied to two versions of path delay PUFs: Ring Oscillator and Arbiter.

A. Ring Oscillator PUF

The RO PUF is a path delay PUF architecture with a long history of research in the PUF community [1], [28]. The repeated instances of inverters is well-suited for an analysis which compares different modeling techniques including both FOM and ROM. In this work, we utilize 7, 31 and 127-stage RO implementations for the development and analysis of our ROM rather than demonstrating the scale of ROM efficiency as in [10]. Moreover, nearly all RO implementations [28] utilize a NAND gate in the sequence of inverters to control the run state of the ROs, as we do in our implementations.

B. Arbiter PUF

The Arbiter (ARB) PUF, introduced in [19], is designed to leverage delay variations that occur in identically configured delay chains. A schematic of the first 8-stages of the ARB PUF is presented in Fig. 1. A vector of digital challenges is used as the select bits to 2-to-1 multiplexers (MUXs) in the delay chains. A response bit is generated by an on-chip arbiter (not shown) which measures the difference in the propagation delays of the two delay chains. A series of challenges are applied to obtain a set of challenge-response pairs, which define a unique bitstring for each device.

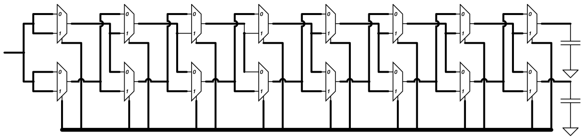


Fig. 1. Schematic for 8-segment Arbiter PUF

IV. Reduced Order Modeling

In this section, we describe the process used to develop the ROM for standard cells, which begins with an automated calibration process and then proceeds with development and analyses of a model for the static, nonlinear DC behavior cascaded with model for the dynamic transient behavior of the cell via linear RC network model and using PDK-informed parameters. The electrical behavior of the CMOS standard cell is captured in a cell level ROM model which is defined using these DC and transient behaviors. We then evaluate the performance characteristics of the ROM on standard cells and PUF circuits and compare it to a SPICE-level model defined within the PDK. Lastly, we incorporate environmental conditions in the ROM model and perform a similar assessment.

A. Model Development Process

The development of the ROM is analogous to the tool flow for a standard cell ASIC design, and is demonstrated here using a set of standard cells that are required to construct an ARB PUF. In prior work, we built an ARB PUF in an IBM 90nm technology which is composed entirely of NAND and Inverter (INV) standard cells [29]. Therefore, the ROMs for these standard cells are developed, and later used to emulate and simulate the ARB PUF circuit, e.g., the 2-to-1 MUXs used as the ARB switch boxes are constructed using NAND and INV standard cells.

The ROM uses the Hammerstein model structure consisting of a static nonlinear block followed by a dynamic LTI block to model the behavior of a standard cell. The input nonlinearity captures the DC behavior of the cell while the linear block models the transient response of the circuit. The block diagram for the Hammerstein model architecture is shown in Fig. 2.

B. Static Nonlinear Block Development Process

The process used for the development of the static nonlinear block for each standard cell utilizes Eq. 1, first proposed by [30]. For the Inverter, Eq. 1 is used directly in the nonlinear block calibration process. Other gates use a modified version of the equation as the corresponding transfer characteristics include multiple inputs. The calibration process tunes the parameters α and β in the equation to match the DC behavior of the cell when simulated with Cadence Spectre [31] using the Matlab Curve Fitting Toolbox [32].

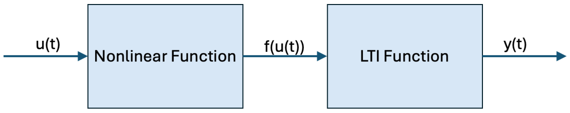


Fig. 2. Hammerstein model architecture used in ROM

$$\frac{-\text{Vdd}}{2}\Big(tanh(\alpha\text{V(IN,Vss)}+\beta)\Big)+\frac{\text{Vdd}}{2} \tag{1}$$

In Eq. 1, the $tanh()$ function provides the DC transfer characteristic for the cell. This equation is multiplicatively scaled by $\frac{\text{Vdd}}{2}$ to account for the range of the output and then additively scaled by $\frac{\text{Vdd}}{2}$ to set the output response to the logic low and high levels of the technology. Within the $tanh()$ function, parameters α and β adjust the drive strength cell, calibrated to PDK results.

C. Dynamic LTI Block Development Process

The development process of the LTI block utilizes an RC network connected to the nonlinear block model. This RC network, with computed values for R_{out} and C_{out}, connect the output of the ROM, ROM_{out}, to the output of the nonlinear block at node DC_{out}. The schematic for both the Inverter ROM and the NAND ROM are included in Fig. 3.

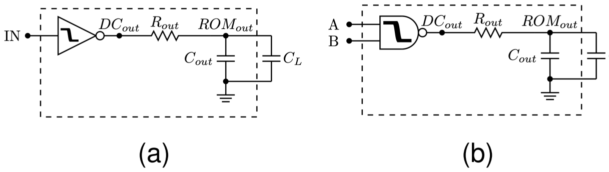


Fig. 3. ROM Schematic for (a) Inverter and (b) NAND gate. Both the inverter and NAND symbols contain a $tanh()$ function to illustrate these cells are modeled with a the nonlinear block presented in this work. The dashed line represents the boundary of the ROM. Devices outside this boundary are load circuitry external to the ROM.

The computed values for R_{out} and C_{out} determine the dynamic change in current through the LTI block, I_{ROM}, from node DC_{out} to the load C_L. This output current is modeled with Eq. 2 under the assumption that logic cells contain an even number of MOSFETs and half of the devices, $N_{devices}$, are in strong inversion and are conducting current.

$$I_{ROM} = \frac{N_{devices}}{2}(I_{DS}) \tag{2}$$

Strong inversion is defined using the saturation current equation (I_{DS}) for a MOSFET, Eq. 3, which includes Gate-to-Source bias (V_{GS}), Drain-to-Source bias (V_{DS}), threshold voltage (V_{th}), transconductance (K) and channel length modulation (λ). Under this assumption, high-frequency switching effects and bias conditions at the individual transistor level are not captured in the ROM.

$$I_{DS} = K(V_{GS} - V_{th})^2(1 + \lambda V_{DS}) \tag{3}$$

Equation 4, models the MOSFET transconductance (K) and maps the ROM to the physical PDK parameters such as oxide thickness (t_{ox}), carrier mobility (μ) and geometry (W, L).

$$K = \frac{1}{2}\mu C_{ox}^{'}\frac{W}{L} \tag{4}$$

For each ROM, we fix V_{GS} = V_{DS} = 0.6V, in Eq. 3, under the assumption half the devices are conducting and the other half are off. The threshold voltage component of the model, V_{th}=0.475V, is computed with the linear extrapolation method [33] and fixed throughout the simulation. It is increased to 0.5V for the high V_{th} version of the ROM. This simplification is necessary to map the behavior of a cell to an equation for an individual transistor as there are multiple gate and drain voltages throughout the devices in a logic cell.

The load capacitance, C_L, is fixed to 1 fF for all ROM and does not include the additional capacitance and fan-out configuration of downstream gates. This value was chosen as an approximation C_{ox} of a device under 0 V bias, connected as a load to the cell. The resistance of the RC network, R_{out}, applies the current equation, for I_{ROM}, using Eq. 5.

$$R_{out} = \frac{V(DC_{out}) - V(ROM_{out})}{I_{ROM}} = \frac{V_{ROM}}{I_{ROM}} \tag{5}$$

The PDK parameters based on geometry and mobility determine the magnitude of I_{ROM} using the resistance definition in Eq. 5 for the RC circuit. The values of width (W), length (L), oxide thickness (t_{ox}), junction capacitance (C_j), sidewall capacitance (C_{sw}) and overlap capacitance (C_{ov}) are derived from values in the PDK.

The ROM output capacitance, C_{out}, is computed using Eq. 6, which includes the overlap, sidewall and junction capacitance from the PDK and the calculated oxide capacitance when the device is biased in the linear mode of operation. The number of MOSFET devices in the logic cell, $N_{devices}$, scales the total capacitance for each ROM.

$$C_{out} = N_{devices}\left[\frac{1}{2}(C_{ox}\frac{W}{L}) + C_{ov} + C_{sw} + C_j\right] \tag{6}$$

Within-die process variations are central to the randomness (entropy) leveraged by PUFs. Our ROM incorporates variations in process parameters using the parameter K given in Eq. 4. To model local mismatch in the ROM, $\hat{K}$ is defined as a random variable, as shown in Eq. 7:

$$\hat{K} = K\big(1 + K_{mc} \cdot MC_{cal}\big) \tag{7}$$

In our ROM, K_{mc} is a random variable selected during each MC run from a normal distribution $\mathcal{N}(\mu = 0, \sigma = 1)$, and MC_{cal} is a constant scalar term, calibrated to the observed variation from Spectre MC simulations of each standard cell. For the ROM, the best results are obtained when calibrating the NAND, Inverter, NOR and AND gates with MC_{cal} = 6%, 5%, 10% and 20%, respectively.

D. Cell Level Model

The nonlinear and LTI components of the ROM define the behavior of the standard cells based on PDK parameters. Application of the ROM to model the Inverter, NAND gate and 2-to-1 MUX are described later in this section, as well as an evaluation of the ROM under different environmental conditions. Additional gates within the PDK are modeled using the same process, but with different parameter values.

1) Inverter Model: The ROM implementation of the Inverter logic cell follows the nonlinear block development process to optimize values of α and β for Eq. 1 using PDK simulation results for the INVX1 cell. The calibrated DC model, from Eq. 1, shown in Fig. 4, is connected to an RC network as depicted in Fig. 3a and calibrated with the PDK parameters using the LTI block process.

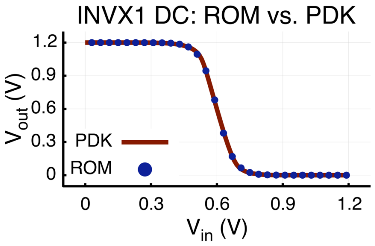


Fig. 4. Inverter 1x cell (INVX1) DC Simulation with comparison between PDK and ROM using parameters calibrated from Curve Fitting Toolbox with resulting coefficient of determination of $R^2 = 0.9999$.

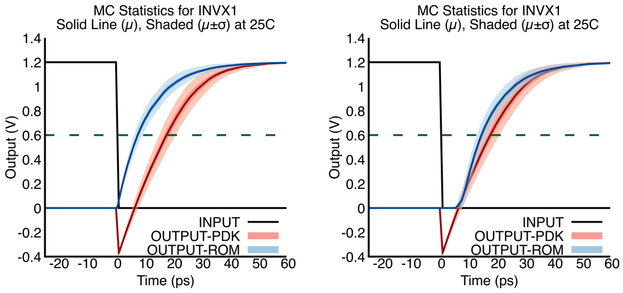
The simulations of the gates modeled using parameters from the PDK exhibit a negative transient in the output voltage V. This negative voltage is caused by gate-drain capacitive coupling as the inputs of the NFET and PFET devices are switched. Our ROM does not include modeling components that represent these parasitic components, and as a result, the output voltage behavior of the ROM does not portray the additional delay introduced by the BSIM4 models from the PDK. We found that using two INVX1 ROMs in series accounts for the time difference, as shown in Fig. 5(b), which can be used as a time compensation technique in cases where actual propagation delays are important.

2) NAND Model: The nonlinear block implementation for a NAND2X1 logic cell is similar to the INVX1 cell but has two inputs, and therefore, requires a separate model for each input, which are given by Equations 8 and 9. The output behavior of the NAND gate is represented by Eq. 10.

$$f(A) = \frac{\text{Vdd}}{2}tanh\Big(\alpha(V(A) - \frac{\text{Vdd}}{2}) + \beta\Big) + \frac{\text{Vdd}}{2} \tag{8}$$

$$f(B) = \frac{\text{Vdd}}{2}tanh\Big(\alpha(V(B) - \frac{\text{Vdd}}{2}) + \beta\Big) + \frac{\text{Vdd}}{2} \tag{9}$$

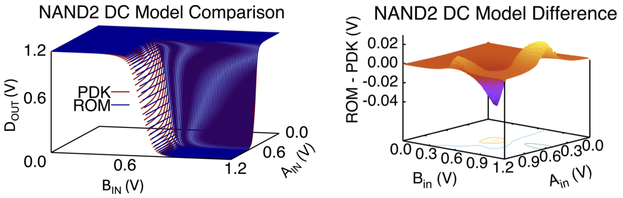
$$\text{NAND}_{DC} = max\Big(f(A), f(B)\Big) \tag{10}$$



(a) (b)

Fig. 5. INVX1 cell ROM and PDK model simulation results for (a) Standard ROM and (b) Standard ROM with a single delay buffer

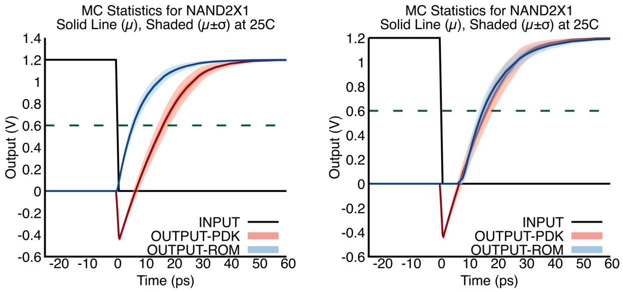
Equation 10, plotted as a function of inputs A and B, produces a surface which corresponds to the analog, DC behavior of the NAND2X1 gate, similar to a continuous version of the corresponding discrete boolean logic truth table. The accuracy of the nonlinear block is demonstrated in a comparison with the SPICE PDK results in Fig. 6. This model is fit using the Matlab Curve Fitting toolbox with a Coefficient of Determination value, R^2, of 0.98763.



(a) (b)

Fig. 6. (a) DC simulation results for 2-input NAND gate ROM comparison with PDK model and (b) corresponding error from ROM-PDK

The LTI block for the NAND gate uses the same structure as the inverter with the notable exception of $N_{devices} = 4$, used in Equations 2 and 6, as a means of accounting for the 4 MOSFET devices in the NAND. The results for the NAND ROM compared to PDK simulations are provided in Fig. 7.



(a) (b)

Fig. 7. Transient Monte Carlo simulations ($25°C$, 1.2V) for NAND2X1 cell t_{pLH} when inputs A and B (black) are set to 0V including ROM (blue) and PDK (red) for (a) Standard and (b) Standard with buffered output. Statistics displayed with μ solid and $\pm\sigma$ shaded. Comparison of the mean delay of the curves has an R^2 value of 0.9565.

3) V_{th} Variant Modeling: Without knowledge of the specific processing techniques used to develop the low V_{th} (LVT) and high V_{th} (HVT) variants to the FETs, appropriate modifications to the ROM are challenging. For the HVT version of the NFET, PDK simulations demonstrate V_{th} increases by 0.025V in comparison to the nominal value.

Methods for increasing V_{th} include channel doping concentration increases and material changes in the gate stack to manipulate the metal work function. From the work on FinFETs by the authors of [34], the relationship between V_{th} and doping can be modeled as presented in Eq. 11.

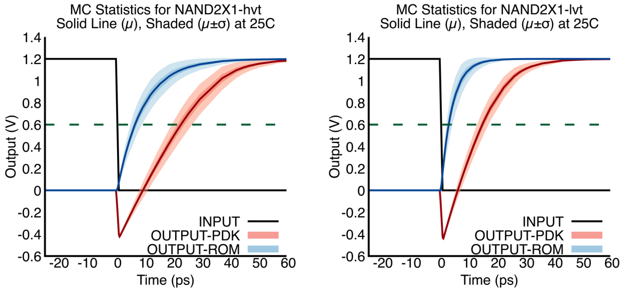
$$V_{th} = V_{FB} + 2_{\psi F} + \frac{\sqrt{2q\epsilon_s N_A}}{C_{ox}}\sqrt{2_{\psi F}} \tag{11}$$

The depletion width and corresponding junction capacitance, C_j, are proportional to channel doping, $N_A \propto \sqrt{C_j}$ [35], through the decrease in the depletion width, W_{dep}. The impact of doping (N_A, N_D) on depletion width and corresponding junction capacitance is modeled in Eq. 12.

$$C_j = \frac{\epsilon A}{W_{dep}} = \frac{\epsilon A}{\sqrt{\frac{2\epsilon}{q}(\frac{N_A+N_D}{N_A N_D})V_j}} \tag{12}$$

Given the limited information on device fabrication, we choose to scale the total capacitance from the base ROM by a factor of $\sqrt{2}$ to account for the additional capacitance in the HVT variants of each cell. The inverse scale is applied to the capacitance for each LVT variant.

Results for modeling the LVT variant, shown in Fig. 8(b), include delay variations of 0.68 ps and 1.194 ps for the BSIM, measured at 0.6V and 1.08V For the ROM, the same output voltages produce delay variations of 0.36 ps and 1.187 ps. When the NAND gate contains HVT versions of the MOSFETs, the simulated delay variations are 1.20 ps and 2.04 ps for the BSIM and 0.73 ps and 2.42 ps for the ROM, measured at outputs of 0.6V and 1.08V as plotted in Fig. 8(a).



(a) (b)

Fig. 8. Statistics for 100 MC simulations for (a) HVT and (b) LVT version of NAND2X1 cell using ROM (blue) and PDK (red).

Table I details the number of parameters in the equations associated with the BSIM4v4.3, Level 1 MOSFET Model and ROM developed in this work. One can infer that the ROM will scale efficiently for large circuits comprised of the standard cells modeled by the ROM due to the absence of repeated model instances compared to the transistor based models (BSIM4v4.3 and Level 1). For this specific comparison, it

TABLE I
PARAMETER COUNT FOR BSIM4V4.3, LEVEL 1 AND ROM

Circuit	BSIM 4v4.3	Level 1	ROM
INV-1X	224 (x2)	33 (x2)	12
NAND2-1X	224 (x4)	33 (x4)	12
NOR2-1X	224 (x4)	33 (x4)	12
AND2-1X	224 (x6)	33 (x6)	12 (x2)
OR2-1X	224 (x6)	33 (x6)	12 (x2)

is important to note the standard cell design for the NAND gate (NAND2X1) contains four transistor instances and the inverter cell (INVX1) contains two. Each ROM uses a different nonlinear DC model for the I/O behavior, but the same parameter set for the transient behavior, captured in the LTI block, thus requiring 10 total parameters for each logic cell. In addition, the ROMs for the AND2X1 as well as the OR2X1 are easily created using a ROM with complementary logic connected to the INVX1 ROM.

4) Multiplexer Model: The conventional implementation of a 2-to-1 multiplexer gate utilizes three instances of 2-input NAND gates and a single inverter, Fig. 9(a). Typically, the 2-input NAND gate is designed using 4 MOSFETs, yielding an efficient realization of a 2-to-1 multiplexer. The smaller size of the layout is noticeable by comparing Fig. 9(b) with an alternative approach shown in Fig. 10(b). However, the NAND version may not be optimal for entropy generation.

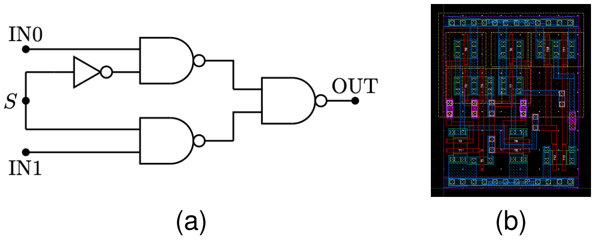


Fig. 9. 2-to-1 multiplexer implementation using 2-input NAND gates (a) schematic (b) custom layout

The alternative implementation of the 2-to-1 multiplexer shown in Fig. 10 uses two instances of 2-input AND gates, an Inverter and a single OR gate. Both implementations of the 2-input AND gate and the 2-input OR gate are typically designed using 6 MOSFETs, therefore yielding a version of a 2-to-1 MUX with higher area utilization. However, this version has the potential to produce additional delay variation compared with the NAND gate version, as the AND gate version of the 2-to-1 MUX has 8.28 ps of delay variation in the output rising pulse compared to 4.18 ps for the NAND gate version. Simulation results for the ROM and PDK implementations of each version are presented in Fig. 11.

E. Ring Oscillator Modeling

In this section, we carry out experiments that apply the ROM to several different RO structures. The RO is unique in that it can be implemented solely with Inverters, and can be scaled arbitrarily, which provides an ideal validation platform for the Inverter ROM. Additionally, we can test both the NAND2X1 and INVX1 cells with the version of the RO which contains an enable signal as shown in Fig. 12. Here, we compare the results of applying the ROM to a 7-stage and a 31-stage RO.

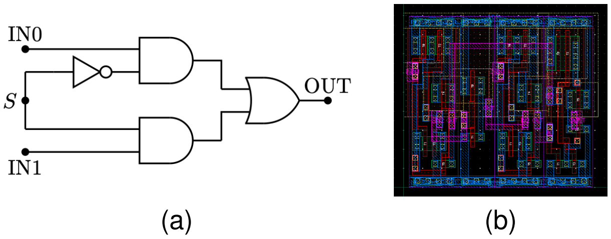


Fig. 10. 2-to-1 multiplexer implementation using 2-input AND gates with 2-input OR gate at output (a) schematic (b) layout using standard cells

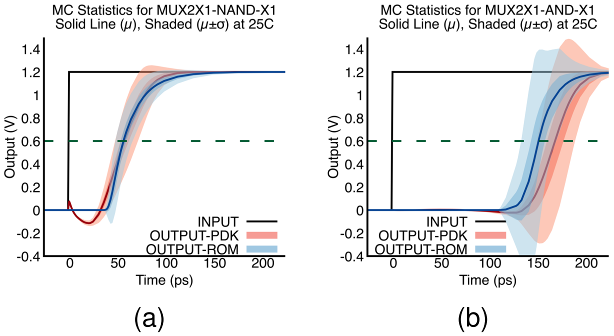


Fig. 11. Simulation results for 2-to-1 multiplexer implementation using (a) 2-input NAND gates (b) 2-input AND gates with 2-input OR gate at output.

The voltage behavior of the first stage input of a 7-stage RO are shown in Fig. 12(a), which are derived from Monte Carlo PDK-based simulations with local device mismatch enabled compared to the ROM. There is a small discrepancy between the ROM and PDK, which is more noticeable after a couple oscillations. Interestingly, the error is not dominated by RO circuit length, as illustrated in the simulation results for the 31-stage RO, as shown in Fig. 12(b).

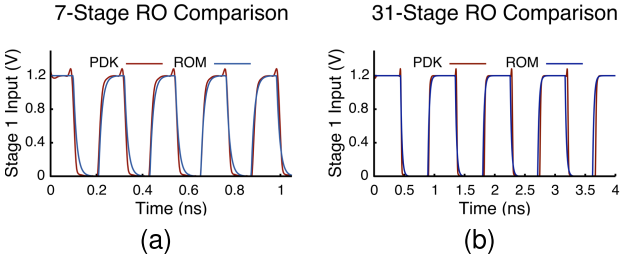


Fig. 12. PDK and ROM comparison for (a) 7-stage RO and a (b) 31-stage RO. The PDK results are plotted in red and the ROM in blue.

F. Arbiter Modeling

The Arbiter PUF is implemented using NAND2X1 and INVX1 gates, and versions of length 8, 16, 32, 64 and 128 segments are constructed and extracted into RCC (resistance and coupling capacitance) netlists. The extracted netlists are simulated with Spectre using several different challenges, and across several temperatures and supply voltages. A comparison of the ROM and the SPICE PDK simulation results for the 8-stage Arbiter at T_{nom}=25°C are shown in Fig. 13(a), and box plots portraying the distribution of path delay differences are shown in 13(b). The ROM performs well in comparison to the PDK results with errors on order of 5%.

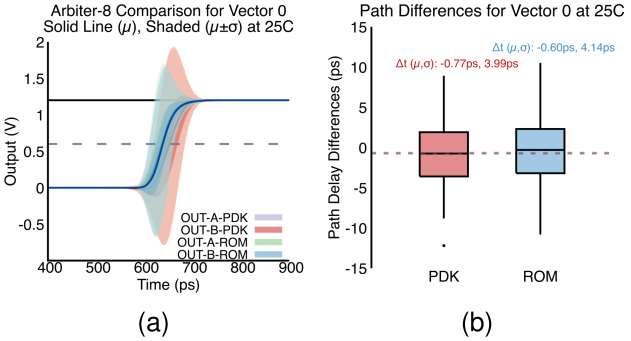


Fig. 13. PDK and ROM Comparison for 8-segment Arbiter using NAND2X1 gates simulated at 25°C for Vector 0 including (a) Transient simulations with statistics with an input pulse applied at time=0s and (b) Box plots with statistics for path delay differences.

G. Environmental Variations

PUFs must exhibit reliable performance under adverse environmental conditions. Reliability testing is commonly performed across supply voltage variations of ±10% of nominal and across a temperature range of $-40°C$ to $100°C$, which can be expanded for military applications to $-55°C$ to $125°C$.

1) Temperature Variations: Changes in temperature between enrollment and regeneration [27] can cause bit flip errors during bitstring regeneration. Mitigation options include error correction codes [27] and error avoidance schemes [36]. When modeling a PUF, it is important to consider temperature effects to understand the robustness of a design.

Sze at al. [35] present a proportional relationship between carrier mobility and temperature using Eq. 13, which is derived from the Boltzmann transport equation where η is driven by scattering mechanisms.

$$\mu_{Si}(T) \propto T^{\eta} \tag{13}$$

To calculate mobility for a given temperature, $\mu(T2)$, one can use this proportional relationship for mobility at a known result for a specific temperature, T1, typically 300K, detailed in Eq. 14. Empirical measurements of n-type and p-type doped silicon at 300K at common device doping levels have an average proportionality factor of $\eta = -2.3$.

$$\mu(T2) = \mu(T1)(\frac{T2}{T1})^{\eta} \tag{14}$$

Modeling the effect of temperature on carrier mobility in a semiconductor device within the range of temperatures described in this work would conventionally utilize $\eta = -2.3$ in Eq. 14. We define this as Temperature Model 0, or TM_0, for our simulations. This model is designed for individual transistors, and therefore, may not be the best model for representing the change in the output of the cell across temperature, as is the goal of our ROM.

Using the output voltage ranges across temperature for the PDK results at 0.6V in Table II, we adjust the model to calibrate the impact of temperature on path delay in the NAND ROM. This process finds good agreement of $< 5\%$ error, in mean (μ) delay, for $\eta = -0.75$, denoted as Temperature Model 1, TM_1, for all of our ROM.

TABLE II
TEMPERATURE MODEL COMPARISON OF NAND GATE DELAYS

Model	μ delay (ps) T=125°C	μ delay (ps) T=−55°C	Range (ps)
PDK	17.48	14.9	2.58
ROM (TM_0)	11.28	3.16	8.12
ROM (TM_1)	7.36	4.85	2.51

The results for the ROM of the 8-stage Arbiter PUF compared with the PDK simulations at 25°C are plotted in Fig. 13(a). Using Eq. 14, with $\eta = -0.75$ for both the NAND and INV gates, we compare the ROM for the Arbiter PUF across a range of temperatures. Included are the plots for the temperature corners −55°C and 125°C, in Fig. 14(a) and Fig. 14(b). The changes due to temperature are well-represented in the ROM using TM_1 after initial calibration at T=25°C.

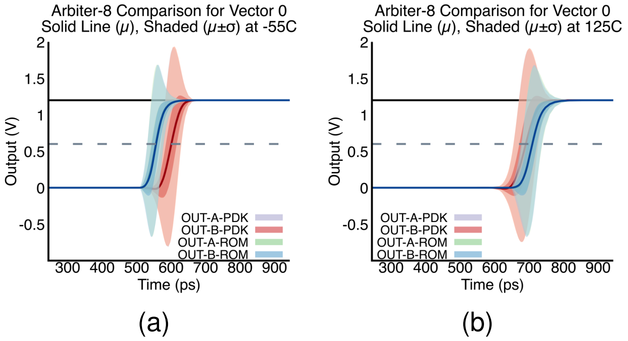


Fig. 14. 8-segment Arbiter simulations at 1.20V for Vector 0 at (a) $-55°C$ and (b) $125°C$. The rising edge input pulse (black) is shifted to time=0s. The corresponding ensemble of outputs on Path A and Path B for the PDK results are plotted in red and purple. For the ROM, the two paths are blue and green. For both sets of plots, the solid line corresponds to the mean of 100 MC simulations with $\pm\sigma$ contained in the dark shaded region and $\pm 3\sigma$ in the light shaded region.

2) Supply Voltage Variations: Similar to temperature, changes in the supply voltage on an IC between enrollment and regeneration can lead to bit flip errors. Accurately modeling the relationship between device behavior and supply voltage is essential to assess the reliability of the PUF design. For our ROM, simulation results from the PDK are used to calibrate parameters in our ROM across supply voltage inputs.

We observe three changes in cell behavior due to increases in the supply voltage and discuss the modeling strategy for each in the following section.

- Increase in maximum output voltage
- Decrease in propagation delay
- Decrease in variability

The output voltage relationship is modeled using the term $\frac{\text{Vdd}}{2}$ as a coefficient in Eq. 1. Additionally, the terms α and β from Eq. 1 have voltage dependence, which is calibrated using PDK simulation results and the Curve Fitting Toolbox. The propagation delay is modified using a multiplicative scale factor on the output capacitance, C_{out}. Lastly, the variability relationship with supply voltage is modeled using a multiplicative scale factor, related to the supply voltage, on the mismatch term (MC_{cal}), for K, from Eq. 7, to formulate a voltage dependent transconductance parameter.

$$K = K\Big[1 + MC_{cal}(\frac{\text{Vdd}_{nominal}}{\text{Vdd}})\Big] \tag{15}$$

Simulating the NAND2X1 gate with Vdd=1.08V, we observe delay variations of 0.841 ps for the PDK using the BSIM and 0.728 ps for the ROM. When biased at 1.32V, the simulated delay variations are 0.567 ps for the PDK and 0.325 ps for the ROM. The results for the ROM at Vdd=1.08V and Vdd=1.32V are included in Fig. 15(a) and Fig. 15(b).

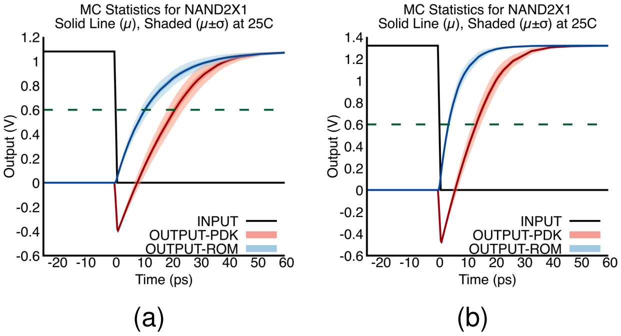


Fig. 15. Comparison plots of 100 MC runs for NAND2X1 (PDK red, ROM blue) with supply voltage variations at $25°C$ for Vdd at (a) 1.08V (b) 1.32V.

V. Standard Cell and Entropy Modeling

Digital IC design tool flows synthesize behavioral descriptions into netlists composed of standard cells for physical placement [37]. Efficiency with regards to size, weight, area and power (SWAP) drives design decisions. For PUFs, maximizing available entropy is desirable, running counter to conventional circuit design focused on SWAP and reliability. This section explores design strategies that maximize path delay variations (entropy) in common logic structures. We study both circuit level schematic options and standard cell options, and, in particular, investigate the impact of utilizing MOSFETs configured with high and low V_{th} process options.

1) Cell Variations: The complementary configuration of multiple devices used to construct standard cells, each with inherent variations, adds complexity to the modeling of their delay variations. We simulated the standard cells using Spectre to develop the reference variation model, but did so with only the device mismatch switch enabled in the PDK process models, i.e., global process variations were disabled. Device mismatch, also called Across Chip Local Variation (ACLV), introduces random variations to components of the transistor models including FET doping and geometry, and to proximity and orientation of polysilicon lines. Statistical results are calculated after 100 MC simulations in Spectre using IBM 90nm CMOS LP PDK with local device mismatch representing the source of entropy under a fixed process corner. We extracted the standard cell layouts with parasitic resistance and capacitance enabled to provide more accurate representation of cell variability and circuit performance. The results for these simulations are summarized in Tables III and IV for nominal standard cell models and the high and low V_{th} variants.

TABLE III
Nominal Cell Variability Comparison

Cell	μ delay (ps)	σ (ps)
AND2X1	64.2816	3.0066
INVX1	17.7897	0.9649
NAND2X1	16.5143	0.6961
NOR2X1	36.1721	1.4596
OR2X1	37.6820	1.6480

From the results shown for the NAND2X1, AND2X1, and OR2X1 gates in Tables III, IV, we observe the standard deviation of the delay for the AND2X1 and NAND2X1 increases by a factor of 2.09 and 1.781 for the high V_{th} variants, respectively. Delay is computed in the standard manner as the Δ-Time (ΔT) between the 50% point on rising input edge to the 50% point on the output edge (t_{pHL}). We use 0.6 V as the 50% point even when the supply voltage is set to $\pm10\%$ of the nominal 1.2 V, when modeling the impact of supply voltage.

TABLE IV
Comparison of HVT and LVT Cell Variability

Cell	μ delay (ps)	σ (ps)
AND2X1-HVT	98.9045	6.4091
AND2X1-LVT	56.0492	2.3927
INVX1-HVT	24.9737	2.0102
INVX1-LVT	16.1045	0.9002
NAND2X1-HVT	22.7983	1.2423
NAND2X1-LVT	14.9263	0.6073
NOR2X1-HVT	54.1233	2.7397
OR2X1-HVT	53.1382	3.1551
OR2X1-LVT	33.9994	1.4941

The input stimulus for each gate is chosen such that a rising transition occurs on the output of the gate. Most of the gates simulated exhibit a ~2x increase in the standard deviation (σ) of delay when the V_{th} is changed from nominal to high, which is notable by comparing the corresponding gates values in Tables III and IV. The mean and variability in delay also changes when the V_{th} is changed from nominal to low, but to a lesser extent.

2) Circuit Variations: The cell variations study can be extended to elements within PUF structures such as 2-to-1 multiplexers. Table V presents the mean delay and standard deviation for multiple variants of the 2-to-1 multiplexer.

To determine the impact in PUF circuit variability of path delay, two variants of the 8-segment Arbiter are compared in Table VI. The first variant uses the version of the Arbiter designed with three NAND2X1 cells and an INVX1 cell. The second variant uses the HVT FETs for both the NMOS and PMOS devices in both sets of NAND2X1-HVT and INVX1-HVT gates. The results indicate an increase in the standard

TABLE V
2-TO-1 MULTIPLEXER PATH DELAY VARIABILITY

Circuit	μ delay (ps)	σ (ps)
MUX2X1-AND-X1	170.6893	8.2806
MUX2X1-AND-X1-LVT	149.1170	7.5629
MUX2X1-AND-X1-HVT	260.6888	15.2517
MUX2X1-NAND-X1	58.7117	4.1281
MUX2X1-NAND-X1-LVT	49.1362	2.8463
MUX2X1-NAND-X1-HVT	106.3358	10.6941

deviation of the output path delay differences by ∼2x across all simulated input vectors.

TABLE VI
STATISTICS FOR 8-SEGMENT ARBITER PATH DELAY DIFFERENCES (ΔT)

Vector	Circuit	μ ΔT (ps)	σ (ps)
00000000	ARB8-NAND2X1	-0.7724	3.9918
11111111	ARB8-NAND2X1	-0.9959	4.2986
01010101	ARB8-NAND2X1	-0.9463	4.2991
00000000	ARB8-NAND2X1-HVT	-0.9209	7.4466
11111111	ARB8-NAND2X1-HVT	-2.2835	8.8921
01010101	ARB8-NAND2X1-HVT	-1.9465	8.7785
00000000	ARB8-NAND2X1-LVT	-0.6109	3.2454
11111111	ARB8-NAND2X1-LVT	-0.8592	3.5208
01010101	ARB8-NAND2X1-LVT	-0.8221	3.5663

We investigate two methods of increasing the level of variability (entropy) within PUF circuit structures. The first method involves utilizing high V_{th} MOSFET models, and the second involves modifying the Arbiter PUF circuit structure to utilize standard cells which exhibit higher levels of variability.

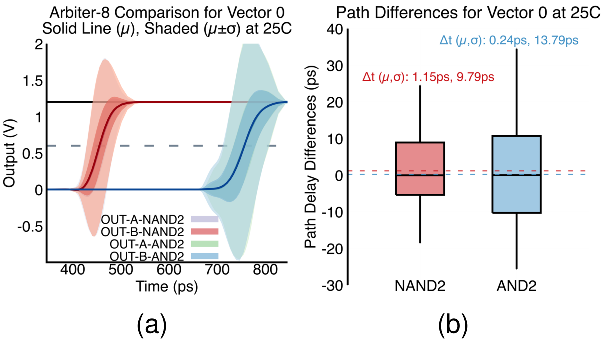


Fig. 16. Simulation comparison between 8-segment ARB PUF circuit utilizing 2-to-1 MUX comprised of NAND gates vs. AND gates, without parasitics, at $25°C$, 1.2V and Vector 0 (a) Transient simulations with a rising edge applied at t=0s and (b) Box plots for the statistics of path delay differences.

As an example, the second method utilizes the version of the 2-to-1 multiplexer given in Fig. 10(a) rather than the version shown in Fig. 9(a). The simulated delay variations associated with these two versions of the Arbiter PUF are shown in Fig. 16 where it is clear that the AND gate version exhibits a substantial increase in path delay variability compared with the NAND gate version. Additionally, we observe mean path delay differences of $\mathcal{N}(\mu = 1.15\text{ps}, \sigma = 9.79\text{ ps})$ for the NAND2 implementation, compared to $\mathcal{N}(\mu = 0.24\text{ ps}, \sigma = 13.79\text{ ps})$ for the AND version, which yields a 1.41x increase variability over the NAND version.

VI. NOISE STUDY

There are many sources of this entropy for PUFs in an IC, including line edge roughness (LER) [38], random dopant fluctuation (RDF) [39], oxide thickness [40], polysilicon variations, gate grain granularity [40], random defects and traps, and STI stress [41]. The impact of these sources of variation can affect variability in transistor performance through changes in V_{th} which impacts drive current, capacitance and correspondingly delay.

In this section, we focus on the adverse effects of temperature-voltage (TV) noise as they act to reduce the effectiveness of a PUF architecture to fully access the entropy that exists within circuit structures. Our analysis investigates TV-noise effects on propagation delay, t_{pLH}, in standard logic cells and PUF structures. Here, we characterize the full extent of delay variations using the range -3σ to $+3\sigma$ for a given cell or circuit. Although we recognize that each cell can potentially exhibit a different mean delay for t_{pLH} and t_{pHL} [42], for this work, we focus on comparisons of t_{pLH} for both SPICE and ROM simulations.

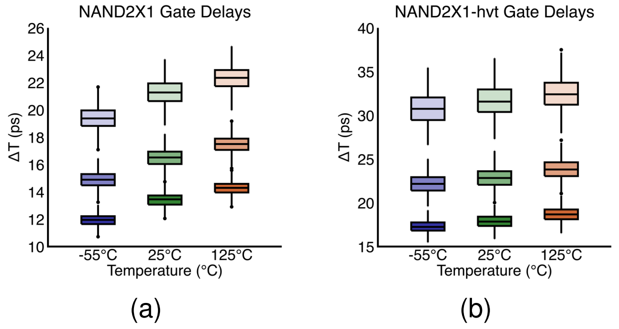
A. Noise Analysis

This section will analyze the impact of noise on both the nominal and HVT version of the NAND2X1 standard cell to determine if the level of entropy is larger than magnitude of TV-noise. Fig. 17 presents the distribution of path delays for both the standard and HVT version of the NAND2X1 gate across a range of TV values. We standardize the variant data using Eq. 16 for the results for the specific gate at nominal temperature and voltage ($25°C$, 1.2V). This standardization process, first proposed in [43], significantly reduces the adverse effects of TV-noise, improving the ability of a PUF architecture to access the entropy associated with variations in path delays. We refer to the standardized delay value as the standard score, z, from Eq. 16.

$$z = \frac{x - \mu}{\sigma} \tag{16}$$

Performing this analysis helps provide the noise component introduced by temperature and voltage variations to the delay through a gate. For the standard NAND2X1 gate, the maximum standard score across all temperature and voltage variations is 1.402, representing a ±40.2% noise addition from temperature and voltage.

For the HVT version of the NAND2X1 gate, the maximum standard score is 1.6735, which corresponds to a ±67.35% addition to the variability due to noise. Therefore, the HVT cells exhibit an additional ±27.15% of delay due to noise. Including both results determines the total standard score for both the nominal and HVT cells across all noise sources. The maximum standard score for both the nominal and HVT cells across all TV noise sources is 1.72, representing an increase in path delay variations of ±72% when using the HVT cells. Next, we subtract the noise additions generated by the HVT cells, ±27.15%. The remaining ±44.85% indicates the net path delay variations, or entropy source, for the NAND2X1-HVT cells compared to the nominal NAND2X1 cells.

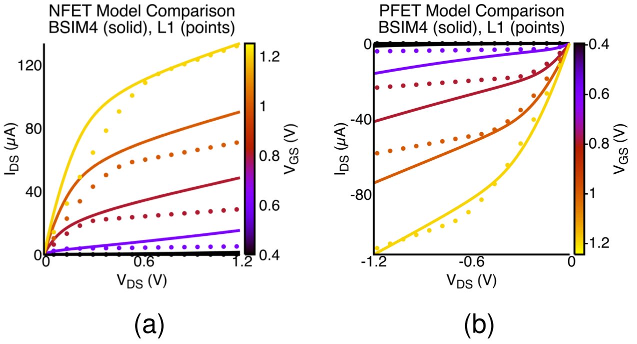


(a) (b)

Fig. 17. Delay statistics across TV variations for (a) NAND2X1 and (b) NAND2X1-HVT gates. Temperature values on the x-axis and fill opacity of the boxes correspond to supply voltages: 1.08V, 1.20V and 1.32V.

VII. Transistor Models: BSIM4 vs. Level 1

Semiconductor manufacturers provide models for the specific technology PDK, typically using the formulation of the BSIM model provided by UC Berkeley [44]. Standard SPICE simulators also have source code definitions for the Level 1 MOSFET model [45], which do not contain modeling elements for many of the physical components of modern short channel devices. However, for a study of model complexity reduction, it is worthwhile to consider the Level 1 formulation to understand the impact of our ROM both in model accuracy and computational time. This section describes our work on comparing simulation results using the SPICE Level 1 MOSFET model with the BSIM4 model provided in the PDK.

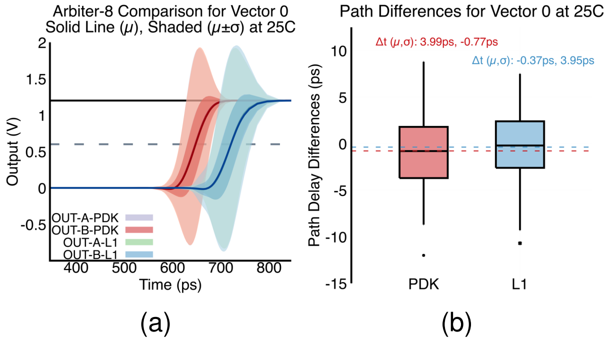


(a) (b)

Fig. 18. Comparison of MOSFET models between SPICE Level 1 and BSIM4 (from IBM PDK) for DC IV curves for (a) NFET and (b) PFET devices.

A. Level 1 Model Simulations Results and Comparisons

We calibrated the Level 1 model formulations for the NFET and PFET devices using simulations from the BSIM4 PDK models, focusing on the higher values of V_{GS} due to the digital nature of the circuit application. The calibration results are presented in Fig. 18 and demonstrate an accurate representation of the PDK results at the highest bias values for both V_{GS} and V_{DS}. Available capacitance parameters are leveraged from the PDK values with minor adjustments after simulation. Additionally, for Monte Carlo simulations for PUFs, we must include randomness in the model parameters. The authors of [6] varied t_{ox} and V_{th} parameters by 30% to simulate process design variability. To achieve the goal of incorporating random process variability in the Level 1 model, we took the Gaussian process variables from the PDK models and applied them to the Level 1 model parameters for oxide thickness (TOX), threshold voltage (VTO) and transconductance (KP) using 3.5% for the standard deviation. These parameters were calibrated to the path delay difference response of Arbiter when modeled with the PDK.



(a) (b)

Fig. 19. Simulation comparison results between Level 1 (blue) and PDK (red) for 8-segment Arbiter at $25°C$, 1.2V and Vector 0 for (a) Transient simulations with input pulse applied at time=0s and (b) Box plots for the statistics of path delay differences.

The Level 1 model does not adequately represent the physics of modern short-channel devices making it difficult to calibrate across both DC and transient effects. We observe the impact of the calibration misalignment for the Level 1 model in the results presented in Fig. 19(a) as the Level 1 model overestimates the Arbiter propagation delay. The mismatch parameters for the Level 1 model perform well as the standard deviation for the path delay differences is 3.95 ps for Vector 0 at ($25°C$, 1.2V) compared to 3.99 ps for the PDK simulations under identical environmental conditions as seen in Fig. 19(b).

VIII. PUF Design Acceleration

The ROM is utilized to model several design options for the Arbiter PUF using the range of variations of path delay differences as the key performance metric. We present here the comparisons for the Arbiter PUF with design options including standard cell alternatives and V_{th} variants.

The ROM simulation results for select Arbiter PUF design options are presented in Table VII, modeled using Vector 0 with TV conditions ($25°C$, 1.2V). Circuits in which every MOSFET is replaced with its HVT variant produced the largest level of variation in path delay differences. Similar trends are observed when 1 or 2 sets of multiplexers are replaced with HVT variants, e.g. circuits which include HVT1 or HVT2 in the name. Circuit designs leveraging alternative standard cells produce additional entropy with the maximal entry belonging to the Arbiter PUF design using AND/OR gates. However, this design configuration also produces the most bias, which could be detrimental to PUF design. Alternatively, designs with the LVT variant of a specific circuit configuration could be beneficial to PUF design as these variants yield reduced bias compared to corresponding HVT option. The optimal PUF design should consider both variations in path delay and bias.

TABLE VII
PATH DELAY DIFFERENCE (ΔT) STATISTICS FOR ARB-PUF DESIGNS

Circuit	μ ΔT (ps)	σ (ps)
ARB8-NAND2X1	0.5424	4.14
ARB8-NAND2X1-HVT1	-0.3395	4.81
ARB8-NAND2X1-HVT2	0.3591	5.54
ARB8-NAND2X1-HVT8	0.8326	6.58
ARB8-AND2X1	1.2062	9.48
ARB8-AND2X1-HVT8	4.3008	22.79
ARB8-AND2X1-LVT8	2.678	13.58

IX. ROM PERFORMANCE COMPARISON

The ROM provides significant efficiency improvements for time and memory compared with instantiating a BSIM for every transistor in the circuit not only because the base standard cell ROM implementations require fewer parameters and map to an entire cell, but also from the capability of these ROM to model parasitic layout connections as well. This enables the ROM to model the behavior of many resistors and capacitors used to accurately represent the physical layout of the circuit. By lumping elements, the ROM cannot capture all of the effects found in the FOM, but it can model first order effects related to delay while providing a solution which will scale efficiently for large circuits. This section presents comparisons of circuit inventory as well as computational time and memory footprint for the ROM, Level 1 and BSIM versions of the Ring Oscillator and Arbiter PUF.

A. Ring Oscillator Performance Comparison

The comparison of circuit inventory for the RO PUF is presented in Table VIII. As expected, the number of devices scale linearly with the increase in circuit size.

TABLE VIII
CIRCUIT INVENTORY FOR 7-SEGMENT AND 127-SEGMENT RING OSCILLATOR CIRCUIT MODELS FOR PUF AND ROM IMPLEMENTATIONS

Circuit	Nodes	BSIM4	Capacitors	Resistors
RO7-PDK	306	16	705	292
RO7-ROM	11	0	1	0
RO127-PDK	4249	256	10773	4374
RO127-ROM	131	0	1	0

Table IX includes the comparison of performance between the Level 1 model, PDK and ROM for the RO PUF circuit. Each implementation is simulated with 100 MC steps with the wall time of the simulation reported in the time column. The PDK speedup is a multiplicative factor used to illustrate the fraction of time required by the Level 1 model and ROM compared with the PDK simulation wall time.

B. Arbiter Performance Comparison

Provided in Table X are comparisons of the circuit inventory and performance metrics for the NAND version of the Arbiter PUF. Similar to the RO PUF, the number of devices scale linearly with circuit size. Additionally, the time and memory comparisons for 100 MC simulations of the 8 and 128 segment versions of the NADN ARB PUF are presented in Table XI.

TABLE IX
TIME AND MEMORY COMPARISON FOR MODELS SIMULATING RO-PUF

Model	Stages	Time (s)	PDK Speedup	Memory (MB)
PDK	7	12.8	-	156
L1	7	6.1	2.1x	157
ROM	7	8.14	1.57x	152
PDK	127	759	-	431
L1	127	415	1.83	369
ROM	127	38.5	19.7x	314

TABLE X
CIRCUIT INVENTORY FOR 8-SEGMENT AND 128-SEGMENT ARBITER CIRCUIT MODELS FOR PUF AND ROM IMPLEMENTATIONS

Circuit	Nodes	BSIM4	Capacitors	Resistors
ARB8-PDK	2767	224	206	2756
ARB8-ROM	79	0	2	2
ARB128-PDK	44227	3584	3206	44096
ARB128-ROM	1159	0	2	2

In terms of time and memory, the ROM outperforms both the Level 1 representation and the BSIM4 from the PDK. Efficiency improvements are observed with increases in circuit size. The trends scale linearly for additional MC simulations, observed up to 10000 MC runs.

X. PUF COMPARISON

This section provides an application of our PUF-ROMS technique applied to a 32-bit implementation of the TCO-PUF introduced in [7]. We modified our ROM to the unit cell structure of the TCO-PUF and compared simulation time of the circuit using the ROM and the IBM 90nm PDK in Spectre.

Table XII presents the results for these simulations which include a 23.94% reduction in simulation time of the circuit modeled using the ROM compared with the BSIM4. The reduction in both time and memory is lower than the reduction of the RO and Arbiter PUF simulated with the ROM.

TABLE XII
TCO-PUF COMPARISON USING BSIM AND ROM

Model	Size	Time (s)	PDK Speedup	Memory (MB)
PDK	8x4	18.12	-	216
ROM	8x4	14.62	1.24	154

We suspect there are two primary drivers for these results. The first, and likely most significant being the unit cell structure in the TCO-PUF contains no logic cells, but rather parallel MOSFETs. Computationally, the ROM is advantageous when representing multiple devices with a static non-linear function. For the unit cell in the TCO-PUF, the ROM contains only the LTI function, modeling each state of the transistor behavior with no assumption for the complementary behavior of CMOS. The second discrepancy lies in the circuit itself. Consistent with the results of the papers for the TCO-PUF, the modeled circuit does not contain any parasitic devices in the network, thus reducing the speedup ratio of the ROM.

XI. CONCLUSION

Through the development of a ROM at the standard cell level, we have demonstrated the scaling of these models to

TABLE XI
TIME AND MEMORY COMPARISON FOR MODELS SIMULATING ARB-PUF

Model	Segments	Time (s)	PDK Speedup	Memory (MB)
PDK	8	64.1	-	363
L1	8	45.7	1.40x	322
ROM	8	7.09	9.04x	172
PDK	128	2200	-	2640
L1	128	638.4	3.45x	2300
ROM	128	147	14.97x	890

digital circuits applied to PUF structures in a similar manner to how circuits are constructed using full order standard cell models. This implies broad applicability to a range of digital circuits. ROM development can provide performance improvement with regards to time and memory compared with both Level 1 model and BSIM4 calibrated to PDK. Both the ROM and Level 1 model performed well after calibration and are well-suited for modeling path delay differences, producing similar results as the BSIM4. While the calibrated ROM provides fast and accurate results, it is important to note the ROM is not intended to supplant the BSIM4 or PDK, but rather supplement it by facilitating design exploration prior to final design verification. For example, the comparison of delay variations can be made over a range of Arbiter segment lengths or different circuit configurations, as in Table VII, without the time required for layout, extraction and SPICE simulation for each circuit. Additionally, the ROM structure could be applied to other PUFs constructed from digital standard cells.

Design modifications including different logic cells or leveraging cell variants such as high V_{th} FETs can be used to increase available entropy in a PUF design, with penalties in both noise and bias. However, the added entropy potentially outweighs the added noise making these cells a viable option for PUF design, provided the additional bias can be mitigated through post-processing techniques. While the techniques discussed in this paper provide options for increasing circuit entropy, they do not explore mitigation for model building attacks on delay PUFs or reliability, which could be explored in future work in addition to model improvements focused on scaling variability to larger circuits.

ACKNOWLEDGMENTS

This work was supported by the Sandia National Laboratories (SNL) Laboratory-directed Research and Development (LDRD) program, and the U.S. Department of Energy, Office of Science, and Office of Advanced Scientific Computing Research under Award Number DE-SC-0000230927.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energys National Nuclear Security Administration contract number DE-NA0003525. This written work is authored by an employee of NTESS. The employee, not NTESS, owns the right, title and interest in and to the written work and is responsible for its contents. Any subjective views or opinions that might be expressed in the written work do not necessarily represent the views of the U.S. Government. The publisher acknowledges that the U.S. Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce the published form of this written work or allow others to do so, for U.S. Government purposes. The DOE will provide public access to results of federally sponsored research in accordance with the DOE Public Access Plan.

REFERENCES

[1] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas, "Silicon physical random functions," in *Proceedings of the 9th ACM Conference on Computer and Communications Security*. New York, NY, USA: Association for Computing Machinery, 2002, p. 148–160. [Online]. Available: https://doi.org/10.1145/586110.586132

[2] D. Suzuki and K. Shimizu, "The glitch puf: a new delay-puf architecture exploiting glitch shapes," in *Proceedings of the 12th International Conference on Cryptographic Hardware and Embedded Systems*, ser. CHES'10. Berlin, Heidelberg: Springer-Verlag, 2010, p. 366–382.

[3] J. Plusquellic, "Shift register, reconvergent-fanout (sirf) puf implementation on an fpga," *Cryptography*, vol. 6, no. 4, 2022. [Online]. Available: https://www.mdpi.com/2410-387X/6/4/59

[4] (2024) Verilog-ams language reference manual. [Online]. Available: https://www.accellera.org/images/downloads/standards/v-ams/VAMS-LRM-2-4.pdf

[5] S. Masoumian, R. Maes, R. Wang, K. K. Yerriswamy, G.-J. Schrijen, S. Hamdioui, and M. Taouil, "Modeling and analysis of sram puf bias patterns in 14nm and 7nm finfet technology nodes," in *2023 IFIP/IEEE 31st International Conference on Very Large Scale Integration (VLSI-SoC)*, 2023, pp. 1–6.

[6] A. Xynos and V. Tenentes, "Metaspice: Metaprogramming spice framework for the design space exploration of puf circuits," in *2023 12th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, 2023, pp. 1–4.

[7] M. S. Mispan, B. Halak, Z. Chen, and M. Zwolinski, "Tco-puf: A subthreshold physical unclonable function," in *2015 11th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME)*, 2015, pp. 105–108.

[8] F. Bizzarri, A. Brambilla, and G. Storti Gajani, "Fastspice circuit partitioning to compute dc operating points preserving spice-like simulators accuracy," *Simulation Modelling Practice and Theory*, vol. 81, pp. 51–63, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1569190X17301727

[9] P. J. SCHMID, "Dynamic mode decomposition of numerical and experimental data," *Journal of Fluid Mechanics*, vol. 656, p. 5–28, 2010.

[10] S. Singh, M. A. Bazaz, and S. A. Nahvi, "Reduced order modeling of ring oscillator system with pod-deim," in *2018 4th International Conference on Computing Communication and Automation (ICCCA)*, 2018, pp. 1–5.

[11] H. Aridhi, M. H. Zaki, and S. Tahar, "Towards improving simulation of analog circuits using model order reduction," in *2012 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2012, pp. 1337–1342.

[12] K. Narendra and P. Gallman, "An iterative method for the identification of nonlinear systems using a hammerstein model," *IEEE Transactions on Automatic Control*, vol. 11, no. 3, pp. 546–550, 1966.

[13] J. Wang, Y.-H. Kim, J. Ryu, C. Jeong, W. Choi, and D. Kim, "Artificial neural network-based compact modeling methodology for advanced transistors," *IEEE Transactions on Electron Devices*, vol. 68, no. 3, pp. 1318–1325, 2021.

[14] R. A. Thakker, C. Sathe, A. B. Sachid, M. Shojaei Baghini, V. Ramgopal Rao, and M. B. Patil, "A novel table-based approach for design of finfet circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, no. 7, pp. 1061–1070, 2009.

[15] C.-T. Tung and C. Hu, "Neural network-based bsim transistor model framework: Currents, charges, variability, and circuit simulation," *IEEE Transactions on Electron Devices*, vol. 70, no. 4, pp. 2157–2160, 2023.

[16] M. Eisele, J. Berthold, D. Schmitt-Landsiedel, and R. Mahnkopf, "The impact of intra-die device parameter variations on path delays and on the design for yield of low voltage digital circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 5, no. 4, pp. 360–368, 1997.

[17] H. Chang and S. Sapatnekar, "Statistical timing analysis under spatial correlations," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 9, pp. 1467–1482, 2005.

[18] S. Fabrie, J. Diego Echeverri, M. Vertreg, and J. P. de Gyvez, "Standard cell library tuning for variability tolerant designs," in *2014 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2014, pp. 1–6.

[19] D. Lim, J. Lee, B. Gassend, G. Suh, M. van Dijk, and S. Devadas, "Extracting secret keys from integrated circuits," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 13, no. 10, pp. 1200–1205, 2005.

[20] G. E. Suh and S. Devadas, "Physical unclonable functions for device authentication and secret key generation," in *2007 44th ACM/IEEE Design Automation Conference*, 2007, pp. 9–14.

[21] J. Aarestad, J. Plusquellic, and D. Acharyya, "Error-tolerant bit generation techniques for use with a hardware-embedded path delay puf," in *2013 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, 2013, pp. 151–158.

[22] Y. Su, J. Holleman, and B. P. Otis, "A digital 1.6 pj/bit chip identification circuit using process variations," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 1, pp. 69–77, 2008.

[23] D. E. Holcomb, W. P. Burleson, and K. Fu, "Power-up sram state as an identifying fingerprint and source of true random numbers," *IEEE Transactions on Computers*, vol. 58, no. 9, pp. 1198–1210, 2009.

[24] M. Bhargava, C. Cakir, and K. Mai, "Reliability enhancement of bistable pufs in 65nm bulk cmos," in *2012 IEEE International Symposium on Hardware-Oriented Security and Trust*, 2012, pp. 25–30.

[25] R. Maes, P. Tuyls, and V. I., "Intrinsic pufs from flip-flops on reconfigurable devices," in *Workshop on Information and System Security (WISSec 2008)*, 2008, p. 17.

[26] P. Simons, E. van der Sluis, and V. van der Leest, "Buskeeper pufs, a promising alternative to d flip-flop pufs," in *2012 IEEE International Symposium on Hardware-Oriented Security and Trust*, 2012, pp. 7–12.

[27] M.-D. Yu and S. Devadas, "Secure and robust error correction for physical unclonable functions," *IEEE Design & Test of Computers*, vol. 27, no. 1, pp. 48–65, 2010.

[28] A. Maiti, J. Casarona, L. McHale, and P. Schaumont, "A large scale characterization of ro-puf," in *2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, 2010, pp. 94–99.

[29] M. Martin and J. Plusquellic, "An on-chip high resolution measurement structure for measuring path delays in an arbiter puf," 2013. [Online]. Available: http://ece-research.unm.edu/jimp/pubs/ARB_PUF.pdf

[30] A. Demir and J. Roychowdhury, "A reliable and efficient procedure for oscillator ppv computation, with phase noise macromodeling applications," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 22, no. 2, pp. 188–197, 2003.

[31] Cadence Design Systems, Inc., *Cadence Spectre Circuit Simulator User Guide*, Cadence Design Systems, Inc., San Jose, CA, 2024, version 23.1. [Online]. Available: https://www.cadence.com

[32] The Mathworks, Inc., *Curve Fitting Toolbox*, The Mathworks, Inc., Natick, MA, 2024. [Online]. Available: https://mathworks.com/help/curvefit/index.html

[33] L. Dobrescu, M. Petrov, D. Dobrescu, and C. Ravariu, "Threshold voltage extraction methods for mos transistors," in *2000 International Semiconductor Conference. 23rd Edition. CAS 2000 Proceedings (Cat. No.00TH8486)*, vol. 1, 2000, pp. 371–374 vol.1.

[34] F. Adamu-Lema, X. Wang, S. M. Amoroso, C. Riddet, B. Cheng, L. Shifren, R. Aitken, S. Sinha, G. Yeric, and A. Asenov, "Performance and variability of doped multithreshold finfets for 10-nm cmos," *IEEE Transactions on Electron Devices*, vol. 61, no. 10, pp. 3372–3378, 2014.

[35] S. Sze and K. Ng, "Physics of semiconductor devices," in *Physics of Semiconductor Devices*. John Wiley & Sons, Ltd, 2006.

[36] W. Che, M. Martinez-Ramon, F. Saqib, and J. Plusquellic, "Delay model and machine learning exploration of a hardware-embedded delay puf," in *2018 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, 2018, pp. 153–158.

[37] M. A. Boussadi, T. Tixier, A. Landrault, and J.-P. Derutin, "Teaching advanced digital asic designs by a complex case study," in *10th European Workshop on Microelectronics Education (EWME)*, 2014, pp. 112–115.

[38] A. Asenov, S. Kaya, and A. Brown, "Intrinsic parameter fluctuations in decananometer mosfets introduced by gate line edge roughness," *IEEE Transactions on Electron Devices*, vol. 50, no. 5, pp. 1254–1260, 2003.

[39] P. Stolk, F. Widdershoven, and D. Klaassen, "Modeling statistical dopant fluctuations in mos transistors," *IEEE Transactions on Electron Devices*, vol. 45, no. 9, pp. 1960–1971, 1998.

[40] S. K. Saha, "Compact mosfet modeling for process variability-aware vlsi circuit design," *IEEE Access*, vol. 2, pp. 104–115, 2014.

[41] Y. Sheu, K. Doong, C. Lee, M. Chen, and C. Diaz, "Study on sti mechanical stress induced variations on advanced cmosfets," in *International Conference on Microelectronic Test Structures, 2003.*, 2003, pp. 205–208.

[42] Y. Wang and M. Zwolinski, "Analytical transient response and propagation delay model for nanoscale cmos inverter," in *2009 IEEE International Symposium on Circuits and Systems*, 2009, pp. 2998–3001.

[43] D. Ismari and J. Plusquellic, "Ip-level implementation of a resistance-based physical unclonable function," in *2014 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, 2014, pp. 64–69.

[44] I. University of California, Berkeley, *BSIM 4.3.0 Manual*, University of California, Berkeley, Inc., Berkeley, CA, 2003, version 4.3. [Online]. Available: https://ewh.ieee.org/r5/denver/sscs/References/2003_BSIM4v30_manual.pdf

[45] H. Shichman and D. Hodges, "Modeling and simulation of insulated-gate field-effect transistor switching circuits," *IEEE Journal of Solid-State Circuits*, vol. 3, no. 3, pp. 285–289, 1968.