Highly Secure Strong PUF based on Nonlinearity of MOSFET Subthreshold Operation

Mukund Kalyanaraman and Michael Orshansky
Department of Electrical and Computer Engineering
The University of Texas at Austin
email:{*orshansky,mukundkm*}*@utexas.edu*

1 Abstract

Silicon physical unclonable functions (PUFs) are security primitives relying on intrinsic randomness of IC manufacturing. Strong PUFs have a very large input-output space which is essential for secure authentication. Several proposed strong PUFs use timing races to produce a rich set of responses. However, these PUFs are vulnerable to machine-learning attacks due to linear separability of the output function resulting from the additive nature of timing delay along timing paths.

We introduce a novel strong silicon PUF based on the exponential current-voltage behavior in subthreshold region of FET operation which injects strong nonlinearity into the response of the PUF. The PUF, which we term subthreshold current array (SCA) PUF, is implemented as a two-dimensional $n \times k$ transistor array with all devices subject to stochastic variability operating in subthreshold region. Our PUF is fundamentally different from earlier attempts to inject nonlinearity via digital control techniques, which could also be used with SCA-PUF. Voltages produced by nominally identical arrays are compared to produce a random binary response.

SCA-PUF shows excellent security properties. The average inter-class Hamming distance, a measure of uniqueness, is 50.3%. The average intra-class Hamming distance, a measure of response stability, is 0.6%. Crucially, we demonstrate that the introduced PUF is much less vulnerable to modeling attacks. Using a machine-learning technique of support-vector machine with radial basis function kernel for best nonlinear learnability, we observe that "information leakage" (rate of error reduction with learning) is much lower than for delay-based PUFs. Over a wide range of the number of observed challenge-response pairs, the error rate is $3 - 35$X higher than for earlier designs.

2 Introduction

Many electronic systems require solutions for security, unique identification, and authentication. As a low cost solution, physical unclonable functions (PUFs) have been proposed [1, 2]. PUFs are pseudo-random functions that exploit the randomness inherent in the scaled CMOS technologies to generate random output strings. In response to an input challenge a PUF generates a binary response. Because of the randomness of the input-to-output mapping, different PUFs generate a different response for the same challenge. The set of challenge-response pairs (CRPs) defines the behavior of a PUF and provides an ability to uniquely identify it.

Multiple realizations of PUFs have been proposed [1, 3–9]. The key distinction among different PUF constructions is between strong and weak PUFs. The distinction is based on the rate at which the number of CRPs grows with the size of the physical realization of a PUF [10]. Weak PUFs are characterized by a small number of CRPs. Strong PUFs are systems with a large number of CRPs, and in an ideal case, the CRP set size grows exponentially with the size of the PUF. The exponential size of the CRP set makes it impossible to record the responses for a PUF of a reasonable size.

Strong PUFs are essential for public authentication security protocols in which the number of CRPs needs to be large such that the same CRPs are not re-used for authentication (preventing the adversary from simply capturing the CRPs transmitted in plain text and using them for subsequent attacks). However, for a strong PUF to be an effective security primitive, the CRPs need to be unpredictable: given a certain set of known challenge-response pairs, it should not be possible to predict the unobserved CRPs with any reasonable probability. If that is not the case, an adversary can stage an attack based on building a model of the PUF. A number of strong PUFs have been proposed in the literature over the years. However, the unpredictability of responses in published strong PUFs has been shown to be limited. The earliest example of a strong silicon PUF is the arbiter-based PUF proposed in [1]. It exploits variation in path delays between gate stages in two parallel propagation paths to generate a binary response by using an arbiter. The arbiter-based PUF has been shown to be vulnerable to model-building attacks [11, 12]. In such attacks, machine-learning techniques, such as regression, neural networks, support vector machines, are used to construct a model of the internal parameters of a PUF based on the observed instances. Attempts to remediate this vulnerability resulted in several variants of the arbiter-based PUF [4, 5]. These approaches attempt to improve unpredictability by using digital techniques. In [8], an XOR gate is used to scramble outputs of two parallel arbiter-based PUFs. In [5], a feed-forward path is introduced within the arbiter-PUF circuit as a way to inject nonlinearity. Unfortunately, recent work [11] shows that all of the derivatives of the arbiter-based PUF are also vulnerable to model-building attacks, even though the improved versions require a larger number of observed CRPs for building a model.

In weak PUFs, the number of CRPs typically grows linearly with the PUF physical size. That implicitly means that each CRP depends on a single, and not shared, realization of a random physical property. That makes it impossible to build a model of such a PUF. Several constructions have been demonstrated. The SRAM-based PUF produces a unique chip signature by relying on threshold voltage mismatch between cell transistors which leads to a cell settling into a random state upon powering up the array [6, 7]. The ring oscillator PUF proposed in [8] generates a response based on the random frequency difference between pairs of ring oscillators.

This paper introduces a novel intrinsic strong silicon PUF based on the essential nonlinearity of terminal current-voltage behavior of field-effect transistors (FETs) at the nanometer scale. *The fundamental principle is reliance on the subthreshold regime of the FET operation, where current is an exponential function of threshold voltage, which exhibits strong random intrinsic variability.* An additional nonlinearity is due to the exponential dependence of threshold voltage (1) on drain-to-source voltage due to drain-induced barrier (DIBL) effect, and (2) on body-to-source voltage due to body effect. Both of these are used to create coupling between FETs in the array, further improving nonlinearity and unpredictability. The new PUF shows excellent security properties.

3 New Source of Nonlinearity: FET Subthreshold Current

We develop a principled approach to significantly improving PUF resilience against machine-learning attacks. It has been recognized that the limitations of strong arbiter-based PUFs in terms of unpredictability are due to their linear additive dependence on partial delays in generating a response. Machine-learning methods are particularly effective in constructing models of such functions. Machine-learning algorithms for classification are tasked with classifying an object given a set of its attributes. In supervised learning setting, the algorithm is first given a set of training examples in which both the attributes and the label is available. If the space being learned is naturally linearly separable, it is easy for the learning algorithm to derive a classification rule with low prediction error.

Unfortunately, the known silicon realizations of PUFs have utilized output functions that are linear, or nearly linear, in the base random variables. In fact, delay-based functions are intrinsically poorly suited for this task as (1) segment delay is near-linear in threshold voltage, and (2) path delays are naturally additive, and, thus, linear, in segment delays. Most strong silicon PUFs known thus far have been derived from the original work on arbiter PUFs for which the output can be described as a linear

function of the delays of individual stages, as formalized in [13]. Attempts to introduce nonlinearity in the arbiter-based PUF, such as using feed-forward paths or XORing the outputs introduce nonlinearity through digital means. Empirical results of model-building attacks show that the added nonlinearity helps but is insufficient in that low prediction errors can still be achieved. A distinct limitation of at least some digital techniques, those based on XORing outputs, is that PUF instability increases along with the improvement in unpredictability [11].

In order to aid the discussion, we introduce a formal distinction between the ways of injecting nonlinearity. For most silicon PUFs, a random bit is produced by evaluating $sgn(f(\mathbf{x}) - f(\mathbf{y}))$, where $\mathbf{x}$, $\mathbf{y}$ are vectors of realizations of a random physical parameter. Function $f(\cdot)$ maps the underlying realizations of physical parameters, e.g. threshold voltages, to a measurable circuit-level quantity, e.g. delay or voltage. If function $f(\cdot)$ is expressible entirely in terms of real-valued functions we call it a fully continuous random function (FCRF), otherwise we call it a mixed continuous-discrete random function (MCDRF). With that distinction in place, we point out that the above digital techniques of achieving nonlinearity still use delay races as a building block for PUFs with the underlying mechanism of generating pseudo-random behavior remaining linear. Thus, both the XOR PUF and the FF PUF start with a "native" FCRF-based PUF and ultimately use the mixed continuous-discrete random function to achieve nonlinearity. Given that the known digital techniques can be equally applied to other underlying ("native") FCRF-based PUFs, the question becomes: can strong silicon PUFs utilizing fully continuous random functions be constructed that are significantly more secure than the FCRF-based delay PUF? We provide an affirmative answer in this paper.

The key for engineering a secure silicon PUF is identifying an output function that would be nonlinear in random variables. We introduce a highly unpredictable PUF that uses the strongly nonlinear I-V terminal dependencies to generate PUF responses. Its central feature is that it moves away from the delay/digital implementation paradigm towards the current/analog one, thereby realizing the necessary degree of nonlinearity over a space of permutations. Because it doesn't rely on digital techniques for injecting the nonlinearity, it does not compromise the stability in the output response to environmental variations.

The output function should ideally have two properties: (1) be nonlinear in random parameters, and (2) introduce the coupling effect in which two or more random variables interact in producing the output. Both of these properties are enabled if the binary output is produced by comparing two voltages produced by a suitably arranged *network of FETs operating in subthreshold region.* The key to our analysis is the equation relating the subthreshold current to FET terminal voltages [14]:

$$\begin{aligned} I_{ds} &= I_S 10^{\frac{V_{gs}-V_{th}(V_{ds},V_{bs})}{S}}(1-10^{\frac{-nV_{ds}}{S}}) \\ &= I_S 10^{\frac{V_{gs}-V_{th}+\lambda V_{ds}+\gamma V_{bs}}{S}}(1-10^{\frac{-nV_{ds}}{S}}) \end{aligned} \tag{1}$$

where I_{ds} is the drain-to-source subthreshold current, I_S is the nominal current, V_{gs} is the gate-to-source voltage, V_{th} is the transistor threshold voltage, V_{ds} is the drain-to-source voltage, V_{bs} is the body-to-source voltage, and λ, γ, and n are the coefficients of drain-induced barrier lowering and body-bias, and the subthreshold coefficient respectively. Crucially, the current is exponentially dependent on the threshold voltage V_{th}. This is important because V_{th} exhibits large and spatially-uncorrelated variability due to random dopant fluctuation (RDF). In nanometer scale CMOS devices, RDF is very significant and grows with transistor scaling [14, 15]. Equation 1 also captures the impact of physical mechanisms of drain-induced barrier lowering and of body effect which lead to a dependence of V_{th} on V_{ds} and V_{bs}. In the second part of the equation, we use a linear expansion of V_{th} in terms of V_{ds} and V_{bs} to enable closed-form analysis.

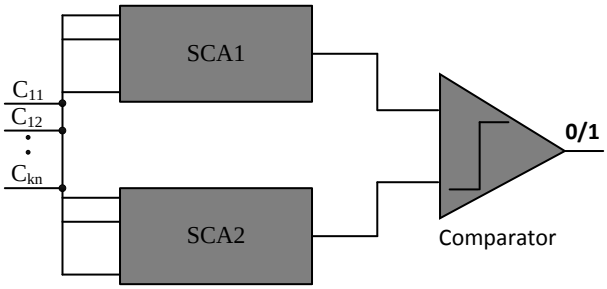


Figure 1: *PUF architecture.*

4 Subthreshold Current Array PUF

4.1 Array PUF Architecture

We now present a transistor-level realization of a subthreshold current array PUF (SCA-PUF) that exploits the above current behavior to construct a highly secure strong PUF. Figure 1 depicts the overall architecture of the SCA-PUF. The PUF is implemented as a two-dimensional transistor array with all devices subject to stochastic variability operating in subthreshold region. The 2D organization allows to maximize the reliability and security properties of the PUF, as demonstrated by experiments.

Each PUF consists of two nominally identical arrays. The array schematic is shown in Figure 2. The array is composed of k columns and n rows of a unit cell. We use the term "stochastic" transistor to refer to a device with high amount of threshold voltage variability. The unit cell consists of a stochastic subthreshold nFET, which is a transistor with a highly variable threshold voltge that always operates in the subthreshold region. A non-stochastic switch transistor is arranged in parallel to the stochastic FET. The non-stochastic transistor $M0$ acts as a load device and operates in the subthreshold region (its gate terminal is tied to ground). At the bottom of each column of cells is a footer transistor Miy controlled by $\overline{C_{i1}C_{i2}\ldots C_{in}}$. Its role is to ensure that there is never a low-impedance path to ground from V_{out}.

Both array blocks are driven with the same set of control inputs and thus in the absence of variability produce identical voltages. The randomness of transistor threshold voltages leads to the differences in two output voltages. The binary response is generated by comparing the output voltages produced by the two arrays via a comparator. The size of the CRP set is 2^{kn}, making it a strong PUF.

We now describe in greater detail the building block of the array, the unit cell. In each cell, which we identify using a column index i and a row index j, an NMOS transistor Mij always operates in the subthreshold region: its gate terminal is tied to ground. An NMOS transistor $Mijx$, in parallel with Mij, acts as a switch transistor. Careful sizing of both devices is essential for correct operation. Two requirements need to be satisfied. First, only transistor Mij is subject to significant variation of threshold voltage due to random dopant fluctuation. This is achieved by sizing tranistors Mij to their minimum size to maximize their threshold voltage variability according to Pelgrom's model [16]. Second, the subthreshold current through the switch transistor $Mijx$ needs to be negligible compared to the subthreshold current through Mij. At the same time, $Mijx$ needs to have small on-state resistance. These requirements can be met, for example, with $W = 10W_{min}$ and $L = 10L_{min}$. Because the nominal current I_S in the subthreshold region is exponentially dependent on channel length $I_{ds}(Mijx)/I_{ds}(Mij) \approx 0$ when $C_{ij} = 0$.

The role of the switch transistor is to set V_{ds} of the stochastic transistor to zero. In this case, the impact of the stochastic transistor is effectively "removed" in that its contribution to the branch current is eliminated. At the same time, when the switch transistor is off, because its subthreshold current is negligible compared to the stochastic transistor, its contribution to the total current can

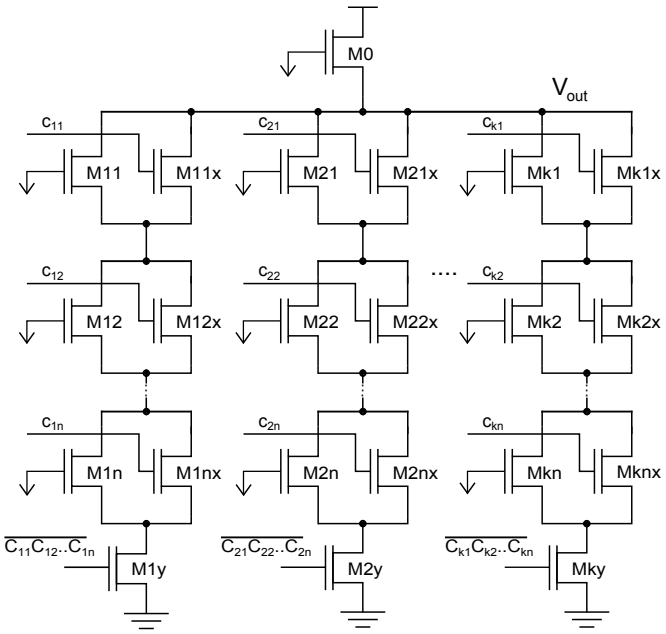


Figure 2: *Circuit schematic of the 2D subthreshold current array.*

be ignored. Depending on the control input, the stochastic transistor therefore is either part of the pull-down network and contributes current that depends on its threshold voltage, or does not impact total current flowing through a branch. Thus, each branch can have 2^n current values.

4.2 Analysis of Array Nonlinearity

The principle feature of the circuit we propose is that it has a highly nonlinear boundary between the regions of PUF 1-outputs and 0-outputs in the kn-dimensional space of V_{th}. In this section, we more formally analyze the nonlinearity of the SCA-PUF. To enable analytical treatment, we derive equations for two special cases: (a) a single-column array, (b) a single-row array, and (c) a simple 2D array with $k = 2$ and $n = 2$. We aim to bring out the form of the nonlinearity involved in each of the two special cases (a) and (b). The two special cases of the 2D array exhibit distinct forms of nonlinearity which, when combined within a 2D array structure, form a rich nonlinear space.

First, we consider the single-row (parallel-only) array with two columns ($n = 1$, $k = 2$). To be able to derive a closed-form equation relating V_{out} to threshold voltages of two "stochastic" transistors, we assume that V_{ds} >100 mV. For $n = 1$ we can also ignore the impact of the body-bias effect. With that, Equation 1 can be written as:

$$log\left(\frac{I_{ds}}{I_S}\right) = \frac{V_{gs} - V_{th} + \lambda V_{ds}}{S} \tag{2}$$

For convenience, we use a simplified notation where $V_{th,M0} = V_0$ and similar for others. Solving for V_{out}, we get:

$$V_{out} = \left(\frac{S}{1+\lambda}\right)[log(I_S) + \lambda V_{dd} - V_0 - log(I_0)] \tag{3}$$

Applying KCL at node V_{out}, $I_0 = I_{11} + I_{21}$, where I_{11}, I_{21} are the currents through $M11$ and $M21$, and describing these currents using Equation 1, we can write an equation for the terminal voltage V_{out}:

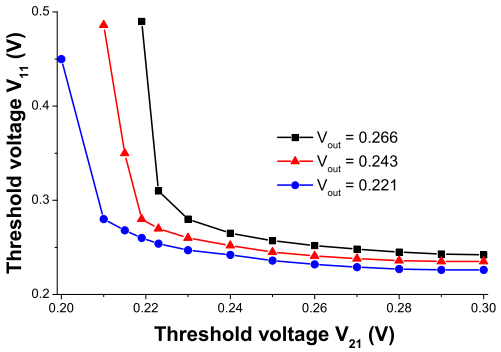


Figure 3: *Response nonlinearity in the single-row array: nonlinearity of additive subthreshold current behavior.*

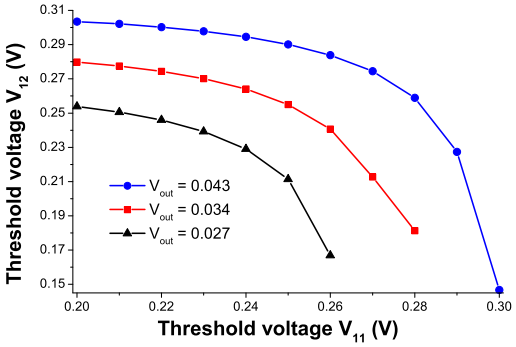


Figure 4: *Response nonlinearity in the single-column array: nonlinearity of series-connected subthreshold FETs.*

$$V_{out} = \left(\frac{S}{1+\lambda}\right)\left[\frac{\lambda V_{dd}}{S} - \frac{V_0}{S} - log\left(10^{\frac{-V_{11}+\lambda V_{out}}{S}} + 10^{\frac{-V_{21}+\lambda V_{out}}{S}}\right)\right] \tag{4}$$

Equation 4 is a transcendental equation. The key to our construction is the nonlinearity of V_{out} in terms of values of threshold voltages of transistors $M11$ and $M21$. The nonlinearity of Equation 4 is explored in Figure 3.

Next we consider the single-column array ($k = 1$) with only two rows ($n = 2$). It represents a subthreshold current array with series-only "stochastic" transistors $M11$ and $M12$. Using Equation 1 for transistors $M0$, $M11$ and $M12$ respectively, and treating the source (drain) of $M11$ ($M12$) as an intermediate node V_x, we get:

$$log\left(\frac{I_0}{I_S}\right) = \frac{-V_{out}(1+\lambda) - V_0 + \lambda V_{dd}}{S} + log\left(1 - 10^{\frac{-nV_{dd}+nV_{out}}{S}}\right) \tag{5}$$

$$log\left(\frac{I_{11}}{I_S}\right) = \frac{-V_x(1+\lambda) - V_{11} + \lambda V_{out}}{S} + log\left(1 - 10^{\frac{-nV_{out}+nV_x}{S}}\right) \tag{6}$$

$$log\left(\frac{I_{12}}{I_S}\right) = \frac{-V_{12} + \lambda V_x}{S} + log\left(1 - 10^{\frac{-nV_x}{S}}\right) \tag{7}$$

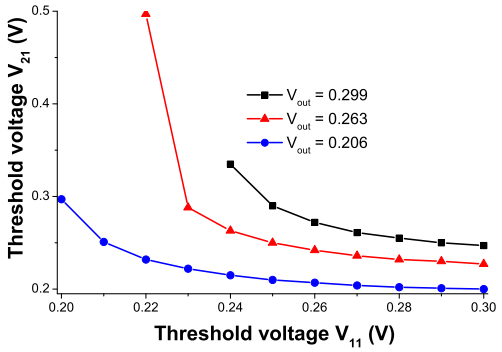


Figure 5: *Response nonlinearity in the* 2×2 *SCA.*

We also know that $I_0 = I_{11} = I_{12}$. Unfortunately, expressing V_{out} in closed form appears infeasible. By simultaneously solving the system of Equations 5, 6 and 7 numerically, we generate Figure 4 and observe the nonlinearity of the single-column (series-only) array topology. The nonlinearity is significant. Notably, while the nonlinear separating surface of the parallel-only array is convex, the surface separating 0- and 1-regions in the series-only array is concave.

Finally, we consider the 2×2 array. We treat the source (drain) of $M11$ ($M12$) as node V_x and source (drain) of $M21$ ($M22$) as node V_y. Using Equation 1 for transistors $M0$-$M22$, we get:

$$log\left(\frac{I_0}{I_S}\right) = \frac{-V_{out}(1+\lambda) - V_0 + \lambda V_{dd}}{S} + log\left(1 - 10^{\frac{-nV_{dd}+nV_{out}}{S}}\right) \tag{8}$$

$$log\left(\frac{I_{11}}{I_S}\right) = \frac{-V_x(1+\lambda) - V_{11} + \lambda V_{out}}{S} + log\left(1 - 10^{\frac{-nV_{out}+nV_x}{S}}\right) \tag{9}$$

$$log\left(\frac{I_{21}}{I_S}\right) = \frac{-V_{21} + \lambda V_x}{S} + log\left(1 - 10^{\frac{-nV_x}{S}}\right) \tag{10}$$

$$log\left(\frac{I_{12}}{I_S}\right) = \frac{-V_y(1+\lambda) - V_{12} + \lambda V_{out}}{S} + log\left(1 - 10^{\frac{-nV_{out}+nV_y}{S}}\right) \tag{11}$$

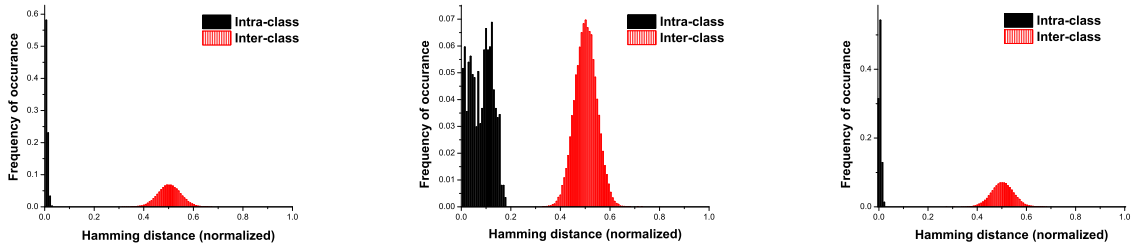
$$log\left(\frac{I_{22}}{I_S}\right) = \frac{-V_{22} + \lambda V_y}{S} + log\left(1 - 10^{\frac{-nV_y}{S}}\right) \tag{12}$$

Additionally, we have $I_{11} = I_{21}$ and $I_{12} = I_{22}$, and $I_0 = I_{11} + I_{12}$. Simultaneously solving these equations, along with the system of Equations 8-12, using a numerical package, we show the resulting nonlinearity of V_{out} in Figure 5. In the next section we show that the effect of the created strong nonlinearities is a PUF whose input-ouput behavior is difficult to model through observation of a partial list of CRPs.

5 Analysis of PUF Security Properties via Transistor-Level Simulations

The performance of the proposed SCA-PUF was simulated using SPICE, the industry-standard transistor-level circuit simulator, in a 45 nm technology node using the predictive technology models [17–19]. The source of randomness is in V_{th} variability assumed to be caused by random dopant fluctuation and therefore to be spatially uncorrelated. The threshold voltages are assumed to follow a normal

distribution with a standard deviation of 40 mV, a value consistent with ITRS data [20]. The results are based on the output random string generated by the PUF having the length of 128 bits.



(a) Single-row (parallel-only) array (b) Single-column (series-only) array (c) Square 2D array

Figure 6: *Inter-class and intra-class HD for three array topologies.*

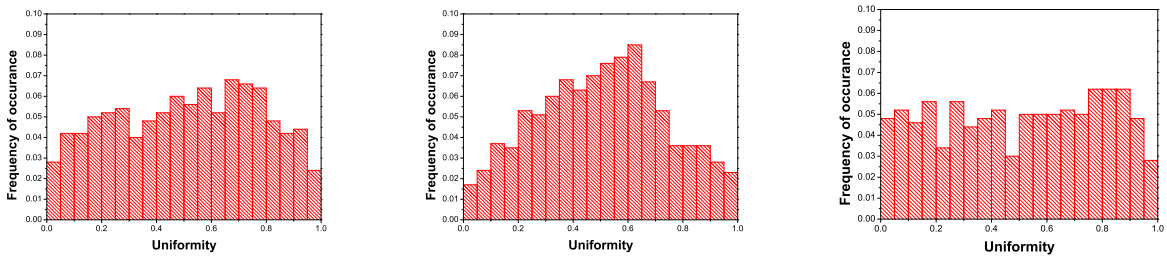
There are several commonly used metrics that quantify the goodness of a PUF [21]. The inter-class Hamming distance (HD) is a measure of the ability to differentiate two different PUFs under the same input. Ideally, each PUF produces an entirely unique response, and thus the ideal inter-class HD normalized to the total number of bits in the output is 0.5. Intra-class HD is the measure of the reliability of a PUF and quantifies how much response of a given PUF changes under a different set of environmental conditions. Ideally, the intra-class HD is 0. Reliability of the PUF responses across different environmental conditions was studied for supply voltage variation of 5% and temperature ranging from $-40\,^{\circ}\mathrm{C}$ to $85\,^{\circ}\mathrm{C}$. Monte-Carlo simulation sample size used to extract inter-class HD was 1000.

The previous section has shown that the nonlinear behavior implemented in the 2D subthreshold current array is a combination of the nonlinearities of the single-row array which represents a parallel-only combination of stochastic subthreshold FETs, and the single-column array which represents a series-only combination of stochastic subthreshold FETs. The series-only and the parallel-only combinations lead to different types of nonlinearities. In principle, the SCA-PUF can be implemented via a variety of $n \times k$ array dimensions. For example, a PUF with the same CRP space of 2^{nk} can be implemented as either $n \times k$ or $nk \times 1$ or $1 \times nk$ array. The latter two types would correspond to the parallel-only and series-only combinations. Therefore, it is important to investigate the behavior of different possible array organizations with respect to PUF security and reliability metrics.

Figure 6 shows the histograms of the normalized intra-class and inter-class HDs. The mean values are summarized in Table 1. We observe that the mean inter-class HD for all three cases is excellent and is practically indistinguishable from 0.5. Intra-class HD is excellent for a parallel-only array but is noticeably higher for the series-only array. Interestingly, when the two types of nonlinearity are combined within the 2D array, intra-class HD is still excellent (below 0.01).

Another useful measure of the goodness of a PUF is the uniformity metric defined by [21]. In an ideal PUF, the fraction of challenges that produces a response of 1 and of 0 should be equal. A useful, and closely, related metric is randomness, as defined by [22], which also quantifies uniformity but in a min-entropy sense. The histograms of the uniformity metric are shown in Figure 7. For comparison, Table 2 summarizes the figures of merit for several published PUFs.

Model-building attacks are the tool with which an adversary may attempt to overcome the authentication guarantees offered by PUFs. Therefore, the ability of a PUF to withstand model-building attacks has been suggested as the ultimate measure of their security [11]. These attacks rely on the power of machine-learning algorithms to model the inner parameters of PUFs through observation of a small set of CRPs. In this paper, the effectiveness of machine-learning attacks was investigated using a support vector machine (SVM) algorithm. Open-source LIBSVM software was used [24]. A set of challenge inputs, along with their output responses, is used as a training set to estimate the PUF model parameters. The estimated model is used to compute the predicted output response for



(a) Single-row (parallel-only) array (b) Single-column (series-only) array (c) Square 2D array

Figure 7: *Distribution of the uniformity metric for three array topologies.*

Table 1: *Average inter-class and intra-class Hamming distance, uniformity, and randomness for* $\sigma_{V_{os}} = 7$ *mV.*

Parameter	Single-row	Single-column	2D array
Inter-class HD	0.493	0.498	0.503
Intra-class HD	0.009	0.076	0.006
Uniformity	0.514	0.507	0.514
Randomness	0.386	0.382	0.439

the non-training challenge inputs and the prediction error rate ϵ is measured. The procedure is carried out for several training sample sets of different size. Figures 8 and 9 show the comparison of prediction error vs. training set size for 8 bit and 16 bit PUFs respectively. To maximize the learning ability of the SVM algorithm, we employed a nonlinear radial basis function (RBF) kernel. Using a nonlinear kernel makes SVM more effective in nonlinear classification problems. We further used a 5-fold cross-validation scheme to select the best kernel parameters. The results indicate that the SCA-PUF is significantly more secure than the delay based PUF. The prediction error is almost an order of magnitude higher than for the arbiter PUF. As we argued earlier, the digital techniques of injecting nonlinearity can be thought of as qualitatively distinct from the behavior of the "native" PUF. For that reason we focus on comparing SCA-PUF behavior to the arbiter PUF learning behavior only. We again note that digital control techniques could also be applied to SCA-PUF, and would further enhance its native nonlinearity and security.

Another practical aspect that we investigate is the influence of comparator characteristics on the overall PUF behavior. The voltage difference $|\Delta V|$ between the two outputs depends on the realizations of random threshold voltages. In an ideal comparator, the binary output matches perfectly the sign of $|\Delta V|$ at any magnitude. In practice, we need to take into account the effect of comparator offset voltage. Offset voltage effectively determines the resolution of the comparator and it may also impact the security properties of the SCA-PUF. We first investigate the distribution of the actual observed voltage differences that the comparator needs to resolve. The mean $|\Delta V|$ is 14.3 mV, 48.5 mV and 14.5 mV respectively for the parallel-only, series-only, and 2D array SCA-PUFs. Figure 10 shows the lower tail of the cumulative distribution of $|\Delta V|$ for three array topologies.

We study the impact of offset voltage on PUFs properties by assuming it follows a normal distribution with a mean of 0 mV and a standard deviation of several mVs. Figures 11 and 12 show the effect of offset voltage on intra-class Hamming distance and on the randomness measure. The inter-class Hamming distance was found to remain nearly-constant around 0.5 for all three circuit constructs. Based on this exploration, we find that a comparator that has an offset voltage of up to $\sigma_{Vos} = 7$ mV would be acceptable but a wider offset distribution would significantly deteriorate randomness. A comparator with such offset can be designed using a strong-arm sense amplifier topology proposed

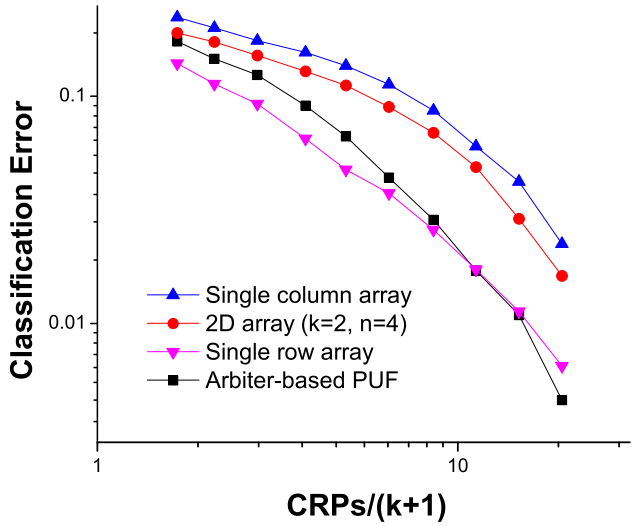


Figure 8: *Classification error from modeling an 8-bit PUF via an SVM-based attack.*

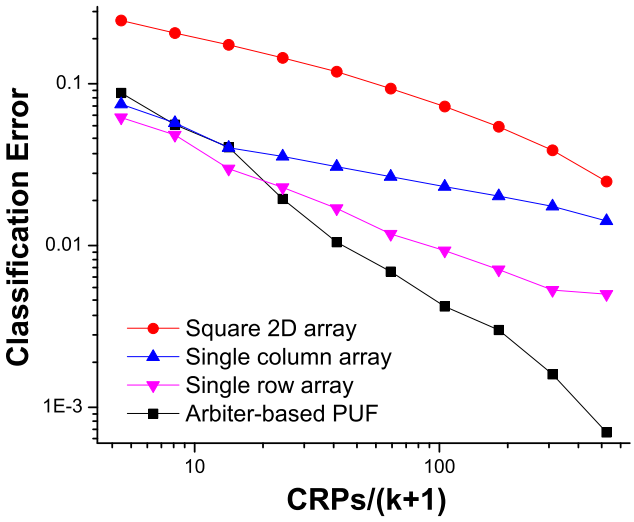


Figure 9: *Classification error from modeling a 16-bit PUF via an SVM-based attack.*

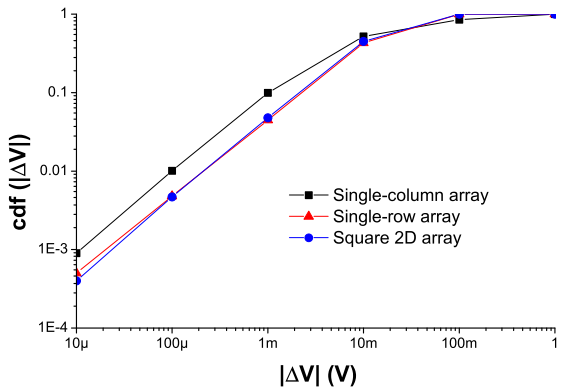


Figure 10: *Cumulative distribution of $|\Delta V|$ for three array topologies.*

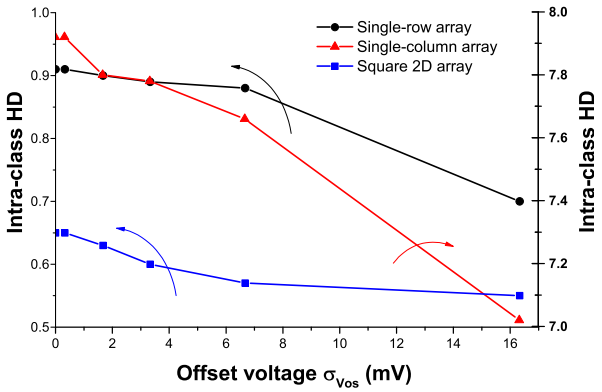


Figure 11: *Dependence of intra-class HD on offset voltage spread.*

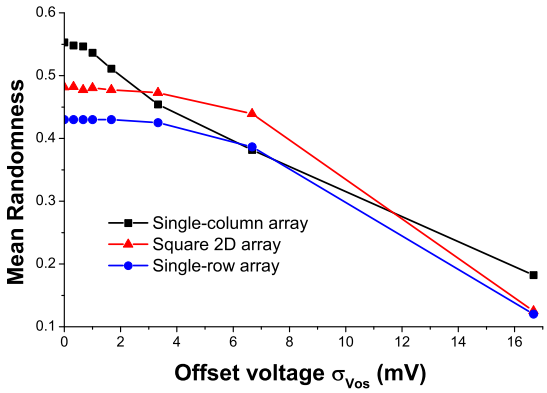


Figure 12: *Dependence of randomness on offset voltage spread.*

Table 2: *Average intra-class and inter-class HD for PUFs reported in the literature [23].*

PUF type	Inter HD (%)	Intra HD (%)
Optical PUF	49.79	25.25
Coating PUF	≈50	<5
Basic arbiter PUF	23	<5
Feed-forward arbiter PUF	38	9.8
Basic Ring Oscillator PUF	≈1	≈0.01
Ring Oscillator PUF w/ comparator	46.14	0.48
SRAM PUF	49.97	<12
Latch PUF	50.55	3.04
Butterfly PUF	≈50	<6
Sense Amplifier PUF	50.00	6

in [25]. Achieving the offset standard deviation of about 7 mV will require either sizing the active devices of the sense amplifier to about $100X$ of the minimum transistor size or using offset cancellation techniques, both of which are easily realizable.

6 Conclusion

We introduced a novel strong silicon PUF based on the essential nonlinearity of responses produced by the physics of field-effect transistors (FETs) at the nanometer scale. The PUF shows excellent security properties which are superior to those reported for other strong PUFs. We demonstrate that the introduced PUF is less vulnerable to modeling attacks and that its "information leakage" is significantly lower than for delay-based strong PUFs.

References

[1] B. Gassend, D. Clarke, M. van Dijk, and S. Devadas. Silicon Physical Random functions. In *ACM Conference on Computer and Communications Security*, pages 148–160, New York, NY, USA, 2002. ACM Press.

[2] J. W. Lee, D. Lim, B. Gassend, G. E. Suh, M. van Dijk, and S. Devadas. A technique to build a secret key in Integrated circuits for Identification and Authentication application. In *Proceedings of the Symposium on VLSI Circuits*, pages 159–176, 2004.

[3] K. Lofstrom, W.R. Daasch, and D. Taylor. IC Identification Circuit using Device mismatch. In *Proceedings of ISSCC 2000*, pages 372–373, 2000.

[4] M. Majzoobi, F. Koushanfar, and M. Potkonjak. Lightweight secure PUFs. In *ICCAD '08: Proceedings of the 2008 IEEE/ACM International Conference on Computer-Aided Design*, pages 670–673, Piscataway, NJ, USA, 2008. IEEE Press.

[5] B. Gassend, D. Lim, D. Clarke, M. van Dijk, and S. Devadas. Identification and authentication of Integrated circuits: Research articles. *Concurr. Comput. : Pract. Exper.*, 16(11):1077–1098, 2004.

[6] D.E. Holcomb, W.P. Burleson, and K. Fu. Initial SRAM state as a fingerprint and source of true random numbers for RFID tags. In *Proceedings of the Conference on RFID Security*, July 2007.

[7] J. Guajardo, S.S. Kumar, G.J. Schrijen, and P. Tuyls. FPGA intrinsic PUFs and their use for IP protection. In *Cryptographic Hardware and Embedded Systems Workshop*, volume 4727 of *LNCS*, pages 63–80, September 2007.

[8] E.S. Suh and S. Devadas. Physical Unclonable Functions for device authentication and secret key generation. In *Design Automation Conference*, pages 9–14. ACM Press, 2007.

[9] S.S. Kumar, J. Guajardo, R. Maes, G.-J. Schrijen, and P. Tuyls. The butterfly PUF protecting ip on every FPGA. In *Hardware-Oriented Security and Trust, 2008. HOST 2008. IEEE International Workshop on*, pages 67–70, June 2008.

[10] J. Guajardo, S.S. Kumar, K. Kursawe, G.J. Schrijen, and P. Tuyls. Intrinsic Physical Unclonable Functions in Field Programmable Gate Arrays. *ISSE/SECURE 2007:*, pages 1–10, 2007.

[11] U. Rührmair, F. Sehnke, J. Sölter, G. Dror, S. Devadas, and J. Schmidhuber. Modeling attacks on physical unclonable functions. In *CCS 2010: Proceedings of the 17th ACM conference on Computer and communications security*, pages 237–249, New York, NY, USA, 2010. ACM.

[12] M. Majzoobi, F. Koushanfar, and M. Potkonjak. Testing Techniques for Hardware Security. *2008 IEEE International Test Conference*, pages 1–10, October 2008.

[13] L. Daihyun. Extracting Secret Keys from Integrated Circuits. Master's thesis, MIT, USA, 2004.

[14] T. Yuan, D.A. Buchanan, C. Wei, D.J. Frank, K.E. Ismail, L. Shih-Hsien, G.A. Sai-Halasz, R.G. Viswanathan, H.-J.C. Wann, S.J. Wind, and H.-S Wong. CMOS scaling into the nanometer regime. *Proceedings of the IEEE*, 85(4):486–504, April 1997.

[15] M. Orshansky, S. Nassif, and D. Boning. *Design for Manufacturability And Statistical Design: A Constructive Approach*. Springer, 2007.

[16] M.J.M. Pelgrom, A.C.J. Duinmaijer, and A.P.G. Welbers. Matching Properties of MOS Transistors. *IEEE Journal of Solid-State Circuits*, 24(5):1433 – 1439, October 1989.

[17] W. Zhao and Y. Cao. New Generation of Predictive Technology Model for Sub-45nm Design Exploration. *Electronic Design*, pages 7–12, 2006.

[18] Y. Cao, T. Sato, M. Orshansky, D. Sylvester, and C. Hu. New Paradigm of Predictive MOSFET and Interconnect Modeling for Early Circuit Simulation. *Methodology*, pages 201–204, 2000.

[19] Y. Cao. Predictive Technology Model. Internet: http://ptm.asu.edu/.

[20] ITRS. International Technology Roadmap for Semiconductors. Internet: http://public.itrs.net.

[21] A. Maiti, V. Gunreddy, and P. Schaumont. A systematic method to evaluate and compare the performance of Physical Unclonable Functions. Cryptology ePrint Archive, Report 2011/657, 2011. http://eprint.iacr.org/.

[22] Y. Hori, T. Yoshida, T. Katashita, and A. Satoh. Quantitative and Statistical Performance Evaluation of Arbiter Physical Unclonable Functions on FPGAs. *International Conference on Reconfigurable Computing and FPGAs*, pages 298–303, December 2010.

[23] A.-R. Sadeghi and D. Naccache. *Towards Hardware-Intrinsic Security: Foundations and Practice*. Springer, 2010.

[24] C.C. Chang and C.J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[25] L. Brooks. *Circuits and Algorithms for Pipelined ADCs in Scaled CMOS Technology*. PhD thesis, Massachusetts Institute of Technology, 2008.