On Design of a Highly Secure PUF based on Non-Linear Current Mirrors

Raghavan Kumar, Wayne Burleson
Department of Electrical and Computer Engineering
University of Massachusetts Amherst, USA
Email: {rkumar, burleson}@ecs.umass.edu

***Abstract*—Physically Unclonable Functions (PUFs) are lightweight hardware security primitives for generating unique signatures from the unpredictable nature of silicon. However, most of the proposed PUFs have been shown to be vulnerable to modeling attacks, especially against Machine Learning algorithms. A subset of challenge-response pairs can leak the required information to break a PUF due to the presence of a linear separating boundary between the PUF responses. In this paper, we propose a strong and secure PUF based on non-linear current mirrors. The fundamental idea is to propagate a current through two identical chains of non-linear current mirrors. The current through a single stage is shifted by some amount based on the strength of the input current. As the current shift is not a fixed value anymore, strong non-linearity is introduced into the challenge-response relationship. The proposed PUF shows excellent properties upon statistical circuit simulation. The average inter-distance and intra-distance of the proposed PUF are 49.9% and 0.8% respectively. One of the most striking features of the proposed PUF is the low information leakage measured in terms of its modeling attack resistance. By employing Support Vector Machine (SVM) based attacks, we observed that the proposed PUF is almost 10-30x stronger than the delay-based PUFs. Moreover, the current mode nature of the PUF circuit enables low power operation. The proposed PUF consumes about 15% lower energy than an arbiter PUF to produce a single response bit.**

***Keywords*—PUFs, Hardware security, Modeling attacks**

I. Introduction

With the notion of embedded computing becoming ubiquitous, there is a strong pressing need to provide cryptographic protection to majority of their applications. For example, technologies like Wireless Sensor Networks (WSN) and Radio Frequency Identification (RFID) are deployed in several applications including healthcare, military etc. For these applications, providing security is extremely challenging due to power and area constraints. Lightweight cryptography is one solution for providing security. In classical cryptography, security operations are often performed by storing secret keys in the hardware, typically in a non-volatile memory. However, the permanent nature of the key poses security vulnerabilities. Moreover, non-volatile memory may impose high area and cost overheads in resource constrained platforms [1].

Physically Unclonable Functions (PUFs) have been proposed as an efficient alternative for generating signatures in integrated circuits [1]. They rely on complex and hard to control manufacturing variations for producing unique signatures. A PUF can be envisioned as a function that produces a response when queried with a challenge. A challenge associated with its response is referred to as a challenge-response pair (CRP). There exist several categories by which PUFs can be classified. One such classification depends on the number of CRPs that can be generated from a PUF. The two types of PUFs based on the number of CRPs are weak PUFs and strong PUFs [2]. Weak PUFs typically have a very few number of CRP pairs and have a strict requirement that the CRPs should never be shared with the external world. On the other hand, strong PUFs have an exponential number of CRPs to the PUF size, which provides the flexibility of using a particular CRP only once for security purposes. It is extremely difficult to collect all the CRPs of a strong PUF through physical measurements within a short amount of time.

PUFs can be used in several applications including device authentication, identification, secret key generation, etc. To enable practical usage, there are some properties to be met by a PUF [3]. First, the responses generated by an instance of a PUF should be unique. In other words, any two responses from two different PUF instances must have a significant difference. The instances may be from the same die or wafer. Second, the PUF responses should be reliable and stable across different operating conditions. Finally, the responses generated should be unpredictable for an adversary. In other words, a subset of CRP pairs should not leak any information for predicting the response of a challenge outside the subset. If there is any leakage of information, the adversary might construct a computer algorithm that can mimic the PUF under consideration.

Several silicon-based PUF implementations have been proposed in the literature [1], [5], [6]. One of the earliest versions is the delay-based arbiter PUFs. The idea is to propagate a rising pulse through two identical delay paths and exploit the variations in path delays to generate a signature bit. However, the vulnerabilities of arbiter PUFs to modeling attacks through Machine Learning (ML) algorithms are well demonstrated in the literature [2]. The vulnerability in arbiter PUFs is due to the linear behavior of CRPs, which allows a ML algorithm to estimate the delay components from a set of CRPs. In general, an arbiter PUF can be modeled through an additive delay model [7]. To introduce non-linearities in CRPs, several variants of arbiter PUFs have been proposed such as XOR, feed-forward construction etc. Though these variants introduce an improved attack resistance, they are still susceptible to model building attacks [2]. A strong PUF to improve the ML tolerance based on sub-threshold FET operation has been proposed in [4], which exploits the exponential current-voltage behavior for PUF operation.

In this paper, we propose a secure PUF based on non-linear current mirrors. A current mirror is one of the basic blocks in analog circuits whose function is to mirror the input current in the read-node to its output node. However, the amount of

978-1-4799-4112-4/14/$31.00 ©2014 IEEE

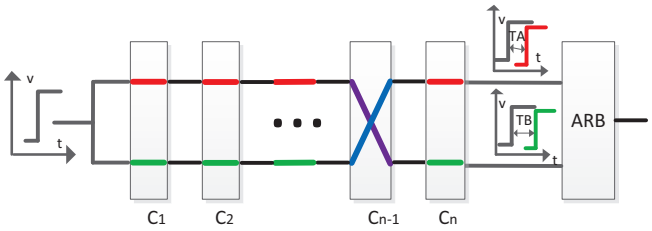

Fig. 1. Arbiter PUF architecture

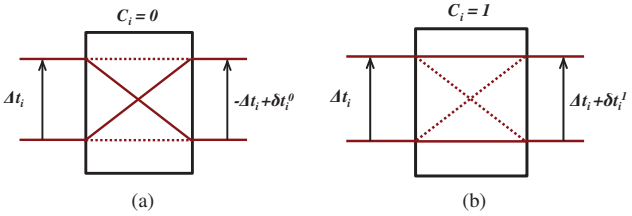

Fig. 2. Delay parameters

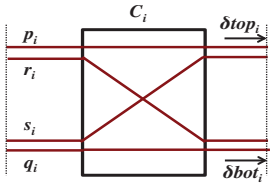

Fig. 3. Individual delay components

copied current will deviate depending on the amount of device mismatch. The basic principle in our PUF design is to use a non-linear current mirror to shift the input current by some amount depending on the strength of the input current itself. A non-linear current mirror can be designed by utilizing a constant current source along with a simple current mirror. Such a construction exhibits different piecewise transfer characteristics and also introduces a threshold to the input current. Our PUF consists of two identical non-linear current mirror chains connected at the input to a common current source. The currents through the unit blocks are then propagated through a current switching element, which accepts an external challenge bit. The construction is similar to an arbiter PUF with the exception that the unit blocks are non-linear in nature. In an arbiter PUF, the amount of delay introduced by a stage remains fixed for a challenge bit. However, in our construction, the amount of current introduced by a stage is dependent on the input current itself. This introduces a non-linearity to the CRPs, therefore making it difficult to model through ML algorithms. Moreover, the current mode nature of the circuit helps in low power applications.

Organization: The rest of the paper is organized as follows. Section II provides some background information. The proposed PUF is described in section III. Experimental methodology and results are presented in section IV. Concluding remarks are presented in section V.

II. Background

In this section, we provide some background information on modeling attacks on arbiter PUFs which closely resembles our proposed PUF. First, we begin with a description of arbiter PUFs.

A. Arbiter PUFs

An arbiter PUF exploits propagation delay variations in integrated circuits to generate a single bit response [1]. The construction of an arbiter PUF is shown in Figure 1. A rising pulse is propagated through a chain of switches. A switch decides the propagation paths of the incoming signals based on an external challenge bit C_i. For example, $C_i = 0$ switches the propagation paths and $C_i = 1$ passes the signals directly to output. An arbiter at the output decides the response based on the delay difference between the racing pulses ($\Delta t_n = T_A - T_B$).

Arbiter PUFs are susceptible to various attacks such as direct measurements, side-channel analysis, software modeling attacks [7] etc. However, we focus only on modeling attacks as they represent the most feasible attack among the group mentioned above.

B. Modeling Attacks

In this section, we provide a brief description of modeling attacks on arbiter PUFs. The attacks are based on the comprehensive analysis of breaking arbiter PUFs as proposed in [7].

As arbiter PUFs exploit the delay variations, they can be modeled through an additive delay model. The delay through a path is the sum of the delays of unit stages. So, by estimating the individual delay components, the delay of a path can be obtained. A direct measurement of individual delay components is very difficult. However, they can be estimated by applying a polynomial number of challenges to the circuit and measuring their responses. To learn the delay components, machine learning algorithms are employed [7]. It is important to note that the exact delay values are not required for modeling attacks. The delay difference through a single stage (positive or negative) can reveal the required information. The delay difference through a single stage is shown in Figure 2. As can be seen from Figure 2, a single stage of an arbiter PUF can be represented using two delay parameters that also contain the information about the challenge bits. To break down the delay parameters further, the individual delay components through a single stage are denoted using the notations as shown in Figure 3. The notations are same as in [7] for consistency. The delay parameter in terms of individual delay components for $C_i = 1$ is shown in equation 3.

$$\delta_{top}(i) = p_i + \delta_{top}(i-1), \delta_{bot}(i) = q_i + \delta_{bot}(i-1)$$

$$\begin{aligned} \Delta t_i &= \delta_{bot}(i) - \delta_{top}(i) && (1)\\ &= (q_i - p_i) + (\delta_{bot}(i-1) - \delta_{top}(i-1)) && (2)\\ &= \delta t_i^1 + \Delta t_{i-1} && (3) \end{aligned}$$

Similarly Δt_i for $C_i = 0$ can be obtained. So, the total delay difference can be represented as,

$$\Delta t_n = p_0 \Delta t_0 + \sum_{i=1}^{n} p_i \delta t_i \quad (4)$$

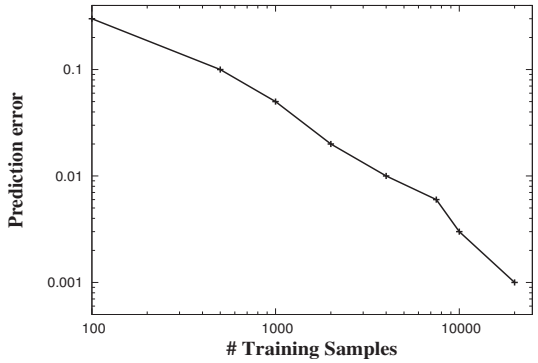

Fig. 4. Prediction error for a 64-bit Arbiter PUF

where $(p_0, p_1 ... p_n)$ represents the parity vector computed using eqn 5,

$$p_i = \prod_{j=i+1}^{n} C_j \tag{5}$$

with $p_n = 1$. To compute the parity vector, the challenge bits are mapped from (0,1) to (-1,1). So, the challenge-response relationship can be expressed as,

$$\Delta t_n \lessgtr 0 \tag{6}$$

As explained above, it is almost impractical to learn δt values through physical measurements. However, they can be estimated using machine learning algorithms. One such common algorithm being used is support vector machine (SVM). SVM uses either linear or non-linear separating surfaces for prediction. The prediction error for a 64-stage arbiter PUF designed in 32nm technology node is shown in Figure 4. We employed open-source SVM^{light} [8] for modeling attacks. It can be observed that an arbiter PUF can be broken by observing as few as 5000 CRPs to achieve a prediction accuracy of 99%. Even, non-linear variants of arbiter PUFs such as feed-forward arbiter PUFs, XOR arbiter PUFs have been shown to be vulnerable to modeling attacks. We refer to [2] for a thorough analysis of modeling attacks on the non-linear variants of arbiter PUFs.

III. Non-Linear Current Mirror based PUF Architecture

In this section, we describe in detail our proposed PUF using non-linear current mirrors. First we present the source of non-linearity in section III-A and the PUF architecture in section III-B.

A. Non-linear Current Mirrors

We use non-linear current mirrors described in [9] as building blocks for our proposed PUF circuit. Current mirrors act as diodes as the current flows only for positive input currents. However, a current mirror starts exhibiting non-linear transfer characteristics if a constant current source is used along with the current mirror. The threshold value of the input current is shifted based on the constant current source. A simple circuit implementation of a current mirror exhibiting non-linear transfer characteristics is shown in Figure 5. The non-linear current mirrors can be constructed in NMOS-only, PMOS-only and NMOS-PMOS combinations. We show only NMOS-PMOS combination for the sake of brevity. The non-linear transfer characteristic exhibited by the current mirror is shown in Figure 6.

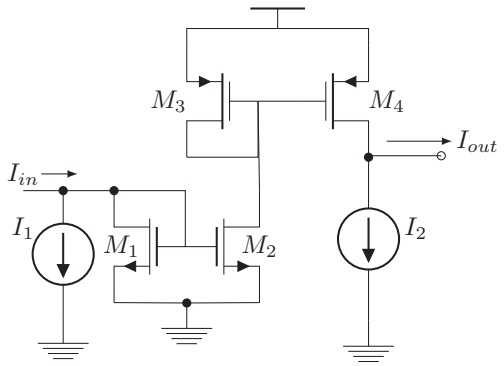

Fig. 5. Non-linear current mirror [9]

The slope of the linear transfer characteristic shown in Figure 6 depends on the relative sizing of transistors M1 and M2. By making M2 stronger than M1, the slope can be increased and vice-versa. Similar non-linear transfer characteristics can be synthesized by using a linear current mirror with a constant current source in different combinations.

1) Effect of Process Variations: As described earlier, the slope of the linear transfer characteristic region depends on the relative sizing of the transistors that make up the current mirror. The slope of the transfer characteristic shifts accordingly depending on the mismatch in transistor sizes due to process variations. We performed a Monte Carlo analysis on the current mirror shown in Figure 5 and the corresponding transfer characteristic is shown in Figure 7. To perform this analysis, we designed the circuit in a 32nm CMOS technology node using the available PTM models [10] assuming threshold voltage variations assigned from a Gaussian distribution with 3σ deviation to be 90mV [11]. For analysis purposes, we assume a constant current source. However, in real-time implementation, there might be process variations in the current source itself. The uncertainty in the transfer characteristics is exploited in our PUF construction as described in the next section.

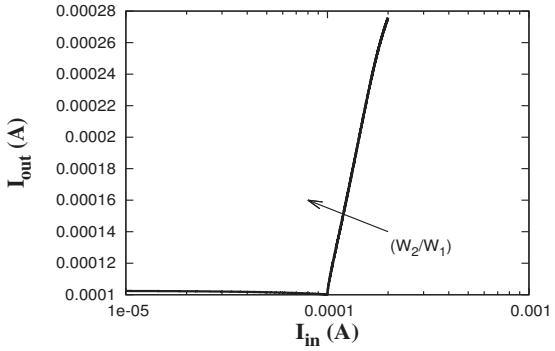

Fig. 6. Non-linear transfer characteristic of the current mirror

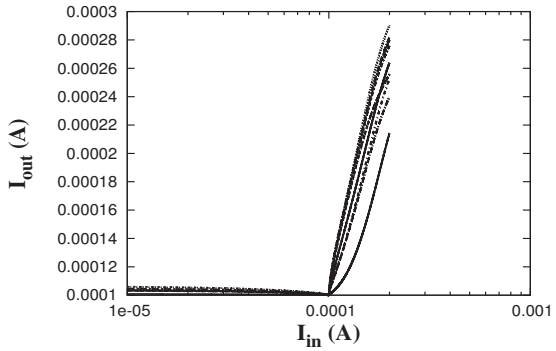

Fig. 7. Impact of process variations on the transfer characteristic of Non-linear current mirror

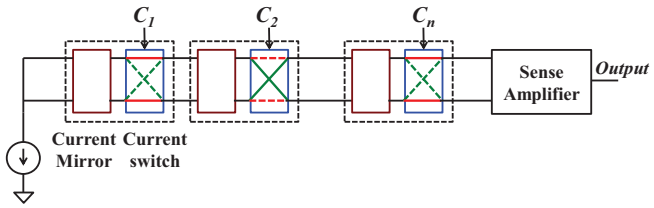

Fig. 8. Proposed PUF Architecture. I_a and I_b are the output currents that are compared to generate the response bit.

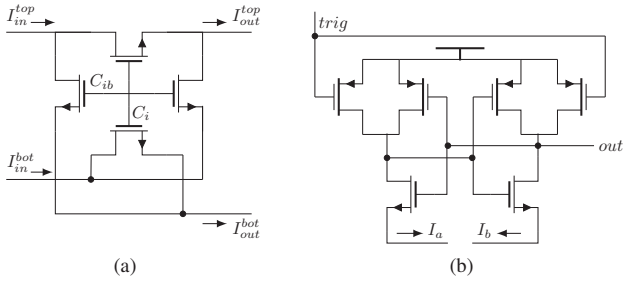

Fig. 9. (a) Current switch, (b) Sense amplifier. The input currents to the current switch are I_{in}^{top}, I_{in}^{bot} and the output currents are I_{out}^{top} and I_{out}^{bot}. C_{ib} is the inverted challenge bit ($C_{ib} = \sim C_i$).

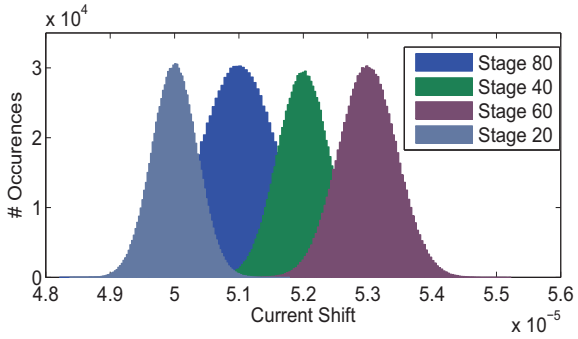

Fig. 10. Current shift distribution for different stages of the proposed PUF

B. Proposed PUF Architecture

The fundamental idea of the proposed PUF is to propagate a current through two identical chains of non-linear current mirrors and switches similar to arbiter PUFs. A switch either *passes* or *switches* the input currents based on an external challenge bit C_i. In our construction, $C_i = 0$ switches the currents and $C_i = 1$ passes the currents directly to output. The entire PUF architecture is shown in Figure 8. The non-linear current mirror shown in Figure 5 acts as the basic building block in our PUF design. However, any non-linear current mirror combination can be used. The current switch circuit is shown in Figure 9(a).

As the input current flows through the building blocks, it gets shifted by some amount depending on the block's transfer characteristic. The amount of shifting depends on the value of the input current itself unlike in arbiter PUFs. The currents at the output I_a and I_b are then compared to generate a single bit response. For comparing the currents, we use a latch-based sense amplifier shown in Figure 9(b) that generates the output bit based on which of the flowing currents is stronger. Before response generation, the *trig* signal in the sense amplifier is pulled low which pre-charges the *out* and out_b (not shown in figure) nodes to V_{dd}. Once the $trig$ signal goes high, the challenge bits are applied. Based on the strength of the flowing currents, one of the output nodes discharges quickly than the other node. This results in a positive feedback which settles the output nodes. The process is repeated again for generating further response bits.

As it is difficult to express the transfer characteristic of the entire PUF in a closed form, we show the non-linearity injected by a single stage in terms of its current shift. While a single stage in an arbiter PUF introduces a fixed delay to the racing signal independent of previous stages, the proposed PUF's single stage injects a different current shift each time depending on the state of the previous stages. For example, we show the current shift distribution of the stages {20,40,60,80} in an 80-stage PUF circuit obtained by simulating the circuit over 20,000 CRPs in Figure 10. The impact of previous stages on the current shift distribution and the source of non-linearity can be clearly inferred from the distribution.

IV. Validation of PUF's Properties

In this section, we describe the experimental results of the proposed PUF circuit obtained through circuit-level simulations. For evaluation purposes, an 80-stage PUF circuit was designed in a 32nm CMOS technology node using the available PTM models [10]. Threshold voltage variations were used as the source of randomness in circuit simulations. The variations were assigned from a Gaussian distribution with a standard deviation of 30mV [11]. For accurate results, the layout level netlist along with the parasitics were used for simulations.

There are several performance metrics to analyze a PUF as described in [12]. We use the metrics uniqueness, reliability and uniformity to analyze the proposed PUF's performance. We also analyze the security properties by performing a modeling attack over the proposed PUF.

A. Uniqueness

Uniqueness is a measure of a PUF's ability to produce a significantly different response for a particular challenge when compared to other identical PUFs. Uniqueness is often

degraded if the process variations are systematic. Random variations are often desirable for PUF's operation. To evaluate uniqueness, Hamming distance (HD) between a pair of responses can be used. We use the average inter-class HD (d_{inter}) described in [12] to evaluate our PUF's uniqueness, which is given by

$$d_{inter} = \frac{2}{m(m-1)} \sum_{p=1}^{m-1} \sum_{q=p+1}^{m} \frac{HD(R_p, R_q)}{k} * 100\%. \quad (7)$$

In eqn 7, R_p, R_q are any two responses from two different PUFs for a particular challenge, m is the number of PUF instances considered, k is the number of bits per response and $HD(.)$ is the Hamming distance of the PUF response pairs under consideration. Unless otherwise mentioned, we assume $\{m, k\}$ to be $\{1000, 128\}$. The histogram of the inter-class HD distribution is shown in Figure 11 and the average value is given in Table I.

B. Reliability

Reliability refers to the ability of a PUF to reproduce the same response under different operating conditions. Intra-class Hamming distance (d_{intra}) can be used to evaluate reliability [12], which is given in eqn 8.

$$d_{intra} = \frac{1}{s} \sum_{j=1}^{s} \frac{\text{HD}(R_i, R'_{i,j})}{k} \text{ x } 100\%. \quad (8)$$

In eqn 8, R_i refers to the k bit response obtained at nominal operating conditions ($1.1V, 25°$), $R'_{i,j}$ refers to the j^{th} sample of the response R_i obtained at different conditions and s is the number of response samples obtained. We assumed both temperature and supply voltage fluctuations ($0 - 75°$, $\pm 0.1V$) for analyzing the PUF's reliability. The histogram and the average values are shown in Figure 11 and Table I respectively.

C. Uniformity

Uniformity is a measure of the proportion of '0's and '1's in a PUF's k bit response. In other words, the number of challenges that produce '0's and '1's must be equal in an ideal scenario. Uniformity of a PUF is evaluated using,

$$(\text{Uniformity})_i = \frac{1}{k} \sum_{j=1}^{k} r_{i,j} \text{ x } 100\% \quad (9)$$

where $r_{i,j}$ is the j^{th} bit of a k bit response. We analyzed uniformity over 100,000 CRPs. The results are shown in Figure 11 and Table I respectively.

We also analyzed the proposed PUF for the presence of bit-aliasing, if any. If bit-aliasing happens, then all the PUFs will produce identical response bits. While uniqueness is analyzed over a k bit response within a PUF, bit-aliasing probability is analyzed for a particular bit over different PUF instances. The average bit-aliasing probability for our PUF is shown in Table I.

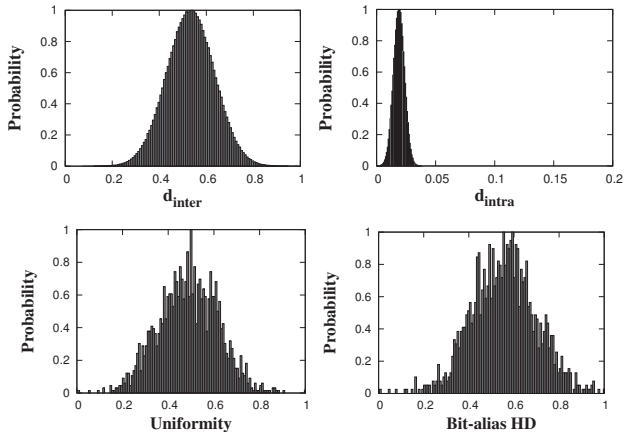

Fig. 11. PUF Performance metrics distributions. (a) Inter-class HD (b) Intra-class HD (c) Uniformity and (d) Bit-aliasing probability

TABLE I. PERFORMANCE METRICS EVALUATION RESULTS

Parameter	Average value	Expected value
Inter-class HD	0.49	0.5
Intra-class HD	0.02	0
Uniformity	0.47	0.5
Bit-aliasing HD	0.43	0.5

From Table I, we can infer that the proposed design has excellent properties required for a PUF. We also present the results on modeling attacks in the next section.

D. Security Properties

As described in earlier sections, a PUF is expected to produce unpredictable responses. However, an adversary can use modeling attacks to break a PUF as described in section II. To estimate the proposed PUF's vulnerability to modeling attacks, we employed SVM based attacks over a set of CRPs. We collected over 2 million CRPs for analysis purposes. All the experiments were executed on a 32-node cluster of Intel Xeon processors employing 8 cores each. A random subset of CRPs is chosen and then used as a training set to learn the PUF's parameters. The parameters obtained from the training process are used as the PUF model for prediction. The prediction accuracy increases with the number of training data used. SVM^{light} was employed for modeling attacks. For prediction, we used a Gaussian radial basis function (RBF) kernel for maximizing the prediction accuracy in non-linear problems. The prediction error for the proposed PUF is shown in Figure 12. We compared the security properties of our PUF with arbiter PUFs. To enable fair comparison, we designed (i) 80 stage arbiter PUF and (ii) 80 stage XOR arbiter PUF, in 32nm technology node. Even feed-forward arbiter PUFs were designed and evaluated. However, SVM based attacks on feed-forward PUFs were not efficient. As described in [2], feed-forward arbiter PUFs can be broken using evolution strategies.

Table II shows the best case prediction accuracies obtained from modeling attacks. To reflect a real-time measurement scenario, we also used error-inflicted CRPs. To deliberately introduce errors, we altered the operating conditions while simulating a PUF. From the modeling attack results, we can infer that the proposed PUF fares better than the arbiter PUF

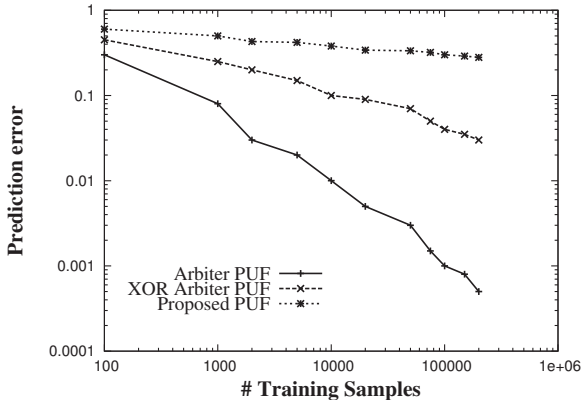

Fig. 12. Prediction errors for 80-bit Arbiter PUF, XOR-Arbiter PUF and Proposed PUF

TABLE II. MODELING ATTACK RESULTS

No. and type of CRPs	Type of PUF	Prediction error	Attack time
$2*10^6$ 0% error-inflicted	Arbiter PUF	0.001	0:21 sec
	XOR Arbiter PUF	0.03	2:20 min
	Proposed PUF	0.3	20:40 min
$2*10^6$ 2% error-inflicted	Arbiter PUF	0.007	0:21 sec
	XOR Arbiter PUF	0.038	2:20 min
	Proposed PUF	0.36	20:40 min

TABLE III. POWER CONSUMPTION COMPARISON

Super-threshold arbiter PUF	14.5 μW
Sub-threshold arbiter PUF	0.052 μW
Proposed PUF (V_{dd} =0.9V)	12.3 μW
Proposed PUF (V_{dd} =0.5V)	0.044 μW

counterparts. The prediction error is almost 10x higher than XOR arbiter PUFs and about 50x higher than simple arbiter PUFs.

E. Implementation Results

The proposed PUF was implemented in a 32 nm CMOS technology node. The PUF circuit was laid out within a $35\mu m$ x $25\mu m$ footprint. For comparison purposes, we also implemented an arbiter PUF using the same technology node and was fit within a $40\mu m$ x $40\mu m$ footprint. The proposed PUF occupied about 20% lower area than an arbiter PUF. This is mainly due to the increased transistor count in an arbiter PUF. For example, the 80-bit arbiter PUF has about 300 transistors more than the proposed PUF.

One of the obvious benefits of using the proposed PUF is its low power dissipation. We compared the power consumption of the proposed PUF to the low power arbiter PUFs proposed in [13]. The power consumption details are presented in Table III. We can infer that the proposed PUF dissipates about 20% lower power than an arbiter PUF operating at super-threshold voltage. We also measured the power consumption at low supply voltage (V_{dd} = 0.5V) and observed that the power consumption is about 15% lower than an arbiter PUF operating at sub-threshold voltage.

V. CONCLUSION

In this paper, we have demonstrated a novel and secure PUF based on non-linear current mirrors. The proposed PUF shows excellent security properties measured in terms of its resistance to machine learning attacks, while achieving good levels in other performance metrics. The proposed PUF also fares better than delay-based counterparts in terms of physical design and power consumption. Future work includes post-silicon validation of the PUF circuit in IBM 32nm SOI technology and also evaluate the PUF's tolerance against other ML algorithms.

ACKNOWLEDGEMENT

This work is supported by Semiconductor Research Corporation (SRC) task # 1836.074 through the Texas Analog Center of Excellence (TxACE) and NSF grant 0964379.

REFERENCES

[1] G. E. Suh and S. Devadas, "Physical Unclonable Functions for Device Authentication and Secret Key Generation," in *Proc. Design Automation Conference*, 2007, pp. 9–14.

[2] U. Rührmair, F. Sehnke, J. Sölter, G. Dror, S. Devadas, and J. Schmidhuber, "Modeling attacks on Physical Unclonable Functions," in *ACM conference on Computer and communications security*, 2010.

[3] I. Verbauwhede and R. Maes, "Physically Unclonable Functions: Manufacturing Variability as an Unclonable Device Identifier," in *ACM Great Lakes Symposium on VLSI'11*, 2011, pp. 455–460.

[4] M. Kalyanaraman, M. Orshansky, "Novel strong PUF based on non-linearity of MOSFET subthreshold operation, " in *IEEE Hardware Oriented Security and Trust*, 2013.

[5] 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 Applications," in *Proc. Symposium on VLSI Circuits*, 2004, pp. 176–179.

[6] L. Lin, S. Srivathsa, D. K. Krishnappa, P. Shabadi, and W. Burleson, "Design and Validation of Arbiter-based PUFs for sub-45-nm Low-power Security Applications," *IEEE Trans. on Information Forensics and Security*, vol. 7, no. 4, pp. 1394–1403, 2012.

[7] D. Lim, "Extracting Secret Keys from Integrated Circuits," Master's thesis, Massachusetts Institute of Technology, Dept. of Electrical Engineering and Computer Science, 2004.

[8] T. Joachims, "Making large scale SVM learning practical," 1999. [Online]. Available: http://svmlight.joachims.org

[9] B. Wilamowski, E. Ferre-Pikal, and O. Kaynak, "Low power, Current Mode CMOS circuits for synthesis of arbitrary nonlinear functions," in *Proc. NASA Symposium on VLSI Design*, 2000.

[10] Y. Cao, "Predictive technology model." [Online]. Available: http://ptm.asu.edu/

[11] ITRS, "International technology roadmap for semiconductors." [Online]. Available: http://public.itrs.net

[12] A. Maiti, V. Gunreddy, and P. Schaumont, "A systematic method to evaluate and compare the performance of Physical Unclonable Functions," 2013, pp. 245–267.

[13] L. Lin, D. Holcomb, D. K. Krishnappa, P. Shabadi, and W. Burleson, "Low-power Sub-threshold Design of secure Physical Unclonable Functions," in *Proc. International Symposium on Low Power Electronics Design*, Aug. 2010, pp. 43 –48.