# On Metrics to Quantify the Inter-Device Uniqueness of Physically Unclonable Functions

Linus Feiten, Matthias Sauer, and Bernd Becker

Chair of Computer Architecture, University of Freiburg
Georges-Köhler-Allee 51, 79110 Freiburg, Germany
{feiten,sauerm}@informatik.uni-freiburg.de

**Abstract.** Physically Unclonable Functions (PUFs) have been an emerging topic in hardware security and trust in recent years, and many different kinds of PUFs have been presented in the literature. An important criterion is always the diversity of PUF responses for different devices, called inter-device uniqueness. A very popular uniqueness metric consists of calculating the pairwise hamming distance between the response bit-strings of all devices, assuming that all response bits are uncorrelated. Such correlations, however, should be regarded when a statement about inter-device uniqueness is made. We therefore propose a novel correlation metric to fulfil this requirement. Furthermore, we show that the hamming distance metric is actually redundant when at the same time the also popular bit-aliasing metric is applied.

**Keywords:** PUF, Physically Unclonable Function, Uniqueness, Metric, Hamming Distance, Bit Aliasing, Correlation

## 1 Introduction

Physically unclonable functions (PUFs) are an emerging topic in hardware security. They provide an alternative to storing cryptographic keys in non-volatile memory that might be susceptible to manipulation or key extraction. A PUF, on the other hand, generates a unique signature for each device it is implemented on, based on the device's physical characteristics. Hence, any physical tampering attack to extract the signature may already lead to changes in these characteristics and thereby alter the signature.

Many different ways to implement PUFs on integrated circuits have been proposed in recent years; e.g. arbiter PUFs [10], butterfly PUFs [9], ring oscillator (RO)-PUFs [14,16,17], TERO-PUFs [2] or SRAM-PUFs [6,1]. For all these PUFs it is important to quantify their ability to uniquely identify a device. I.e. the generated signature should be unique for each device and knowing the signature of one device should not allow for any conclusions about other devices' signatures.

Some very popular metrics to quantify this inter-device uniqueness are presented in [12] and have since been used in several publications, e.g. [4,7,13,5,8]. Section 2 recapitulates these metrics providing a novel analysis of their extreme values. Section 3 then shows that the so-called hamming distance metric is actually redundant in the face of the bit-aliasing metric, that is mostly applied in addition to the former. Section 4 shows that neither of these two metrics is able to recognise correlations between signature bits, resulting in misleading judgements of inter-device uniqueness. To overcome this, a new metric is suggested that takes correlated bits into account. Section 5 concludes the paper.

## 2 Traditional Uniqueness Metrics

Let $m$ be the number of different devices and let $n$ be the total number of PUF response bits generated per device. While a single response bit $r_{i,j}$ with $1 \leq i \leq m$ and $1 \leq j \leq n$ can have different outcomes on a single device (poor intra-device reliability), we define the *signature bit* $sig_{i,j}$ to be the bit's outcome irrespective of poor reliability. The signature bit can thus be considered the outcome after an error correction has been applied. When analysing inter-device uniqueness, only the signature bits outcomes need to be regarded.

### 2.1 Inter-device hamming distance

When $Sig_i = (sig_{i,1}, sig_{i,2}, \ldots, sig_{i,n})$ is the $n$-bit signature of a device $i$, [12] defines the Hamming Distance (HD) metric as

$$HD_{metric} = \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} \frac{HD(Sig_u, Sig_v)}{n}$$

where

$$HD(Sig_u, Sig_v) = \sum_{j=1}^{n} (sig_{u,j} \oplus sig_{v,j})$$

Thus, with $HD(Sig_u, Sig_v) = 0$ for $Sig_u$ being bitwise equal to $Sig_v$, and $HD(Sig_u, Sig_v) = n$ for $Sig_u$ being the bitwise inverse of $Sig_v$, the minimum outcome of $HD_{metric} = 0.0$ is achieved when all devices generate the same signature (worst inter-device uniqueness). The maximal outcome of $HD_{metric}$ converges to 0.5 with growing number of devices $m$. To understand this, we first realise that $HD_{metric}$ can be calculated by regarding each signature bit $j$ separately:

$$\begin{aligned}
&HD_{metric}\\
&= \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} \frac{HD(Sig_u, Sig_v)}{n}\\
&= \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} \frac{1}{n} \sum_{j=1}^{n} (sig_{u,j} \oplus sig_{v,j})\\
&= \frac{1}{n} \sum_{j=1}^{n} \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} (sig_{u,j} \oplus sig_{v,j})
\end{aligned}$$

The maximum of $HD_{metric}$ is thus achieved, when $\sum_{u=1}^{m-1} \sum_{v=u+1}^{m} (sig_{u,j} \oplus sig_{v,j})$ is maximal for each bit $j$. This is the case, when exactly $\frac{m}{2}$ devices generate $sig_{i,j} = 0$ and the other $\frac{m}{2}$ devices generate $sig_{i,j} = 1$ (without loss of generality let $m|2$). The sum is incremented by 1 for each XOR of a "0" bit with a "1" bit. As there are $\frac{m}{2}$ many "0" bits and $\frac{m}{2}$ many "1" bits, the sum's maximal outcome is:

$$\sum_{u=1}^{m-1} \sum_{v=u+1}^{m} (sig_{u,j} \oplus sig_{v,j}) = \frac{m}{2} \cdot \frac{m}{2} = \frac{m^2}{4}$$

Inserting this into the formula of $HD_{metric}$ we get:

$$\frac{1}{n} \sum_{j=1}^{n} \frac{2}{m(m-1)} \cdot \frac{m^2}{4} = \frac{1}{2} \cdot \frac{m}{(m-1)}$$

Thus, the more devices are involved, the closer the maximal $HD_{metric}$ outcome is to 0.5 – indicating optimal inter-device uniqueness. For smaller device populations, however, greater $HD_{metric}$ outcomes must be achieved to demonstrate the same quality of uniqueness. Table 1 shows the maximal outcomes for different numbers of devices.

**Table 1.** Maximal $HD_{metric}$ outcome for different numbers of sampled devices.

| Number of devices $m$ | Maximal $HD_{metric}$ outcome |
|---|---|
| 2 | 1.0000 |
| 4 | 0.6667 |
| 8 | 0.5714 |
| 14 | 0.5385 |
| 20 | 0.5263 |
| 40 | 0.5128 |
| 60 | 0.5085 |
| 100 | 0.5051 |

### 2.2 Bit-Aliasing

Another metric introduced in [12] is bit-aliasing. Here, each signature bit $j$ is analysed individually to detect whether its outcome is biased toward either 0 or 1 over all $m$ devices:

$$BA_j = \frac{1}{m} \sum_{i=1}^{m} sig_{i,j}$$

An unbiased bit should have a $BA_j$ outcome close to 0.5, which is the case when half of the devices generate the bit as 0 and the other half generate it as 1. An outcome of 0.0 (or 1.0) means that the bit was 0 (or 1) on all devices, which is an indicator for poor inter-device uniqueness.

## 3 Redundancy of Hamming Distance Metric in the Face of Bit-Aliasing Metric

In this work, we show that $HD_{metric}$ is in fact equivalent to the $BA_j$. I.e. if the $BA_j$ value of some or all bits indicates poor uniqueness, $HD_{metric}$ indicates poor uniqueness as well, and vice versa. There is no case in which one metric indicates something that is not also indicated by the other.

This is why $HD_{metric}$ can be considered redundant, especially as $BA_j$ is more expressive identifying single signature bits with poor uniqueness and indicating their bias toward 0 or 1, whereas $HD_{metric}$ only produces a single number between 0 and 0.5 for all bits. If such a single number is desired, it could also be derived from the $BA_j$ values as follows. First, a *normalised* version of $BA_j$ is calculated that does not differentiate any more between biases toward 0 or 1:

$$BA_j^{norm} = 0.5 - |0.5 - BA_j|$$

Then, the average over all signature bits is calculated:

$$BA_{metric} = \frac{1}{n} \sum_{j=1}^{n} BA_j^{norm}$$

Next, we show how $HD_{metric}$ and $BA_{metric}$ do actually differ in just a slight manner. First notice, that both metrics are in fact averages over all signature bits. This is obvious for the above definition of $BA_{metric}$ and has been shown for $HD_{metric}$ in 2.1 by rearranging the sums to:

$$HD_{metric} = \frac{1}{n} \sum_{j=1}^{n} \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} (sig_{u,j} \oplus sig_{v,j})$$

Thus, both metrics do not consider correlations between signature bits, such that for an analysis of the metrics only one single signature bit $j$ needs to be regarded.

Let $0 \leq \gamma \leq m$ be the number of devices for which the regarded signature bit's outcome is 1; for the remaining $m - \gamma$ devices it is 0. For $\gamma = 0$ ($\forall i : sig_{i,j} = 0$) and $\gamma = m$ ($\forall i : sig_{i,j} = 1$), both metrics produce the outcome 0.0. For $\gamma = \frac{m}{2}$ (optimal distribution of 0s and 1s) both assume $\sim 0.5$. The slight difference is in how both metrics are mapping the $\gamma$ values in between.

Figure 1 shows the results of both metrics (y-axis) for the range of $\gamma$ values (x-axis). With $BA_j = \frac{1}{m}\sum_{i=1}^{m} sig_{i,j} = \frac{\gamma}{m}$, the plot for $BA_{metric}$ is:

$$BA_{plot}(\gamma) = 0.5 - |0.5 - \frac{\gamma}{m}|$$

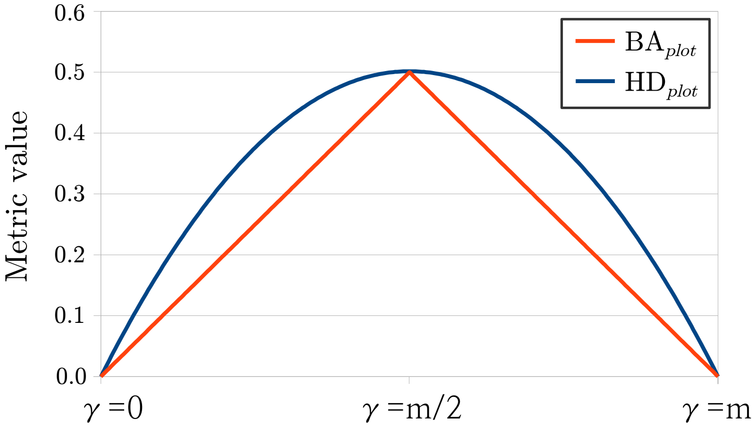

**Fig. 1.** The results of $HD_{metric}$ and $BA_{metric}$ when applied to a single signature bit. $0 \leq \gamma \leq m$ is the number of devices for which this bit is 1.

To construct the plot for $HD_{metric}$, its definition must first be rewritten to:

$$\begin{aligned}
&\frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} (sig_{u,j} \oplus sig_{v,j}) \\
&= \frac{2}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} (sig_{u,j} + sig_{v,j} - 2(sig_{u,j} \cdot sig_{v,j})) \\
&= \frac{2}{m(m-1)} \Bigg( \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} sig_{u,j} + \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} sig_{v,j} \\
&\qquad\qquad - \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} 2(sig_{u,j} \cdot sig_{v,j}) \Bigg)
\end{aligned}$$

$$
\begin{aligned}
&= \frac{2}{m(m-1)} \Bigg( \sum_{u=1}^{m-1} (m-u) \cdot sig_{u,j} + \sum_{v=2}^{m} (v-1) \cdot sig_{v,j} \\
&\qquad\qquad - \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} 2(sig_{u,j} \cdot sig_{v,j}) \Bigg) \\
&= \frac{2}{m(m-1)} \Bigg( \sum_{w=1}^{m} (m-w) \cdot sig_{w,j} + \sum_{w=1}^{m} (w-1) \cdot sig_{w,j} \\
&\qquad\qquad - \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} 2(sig_{u,j} \cdot sig_{v,j}) \Bigg) \\
&= \frac{2}{m(m-1)} \Bigg( \sum_{w=1}^{m} \Big( \big((m-w) + (w-1)\big) \cdot sig_{w,j} \Big) \\
&\qquad\qquad - \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} 2(sig_{u,j} \cdot sig_{v,j}) \Bigg) \\
&= \frac{2}{m(m-1)} \sum_{w=1}^{m} (m-1) \cdot sig_{w,j} \\
&\quad - \frac{4}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} (sig_{u,j} \cdot sig_{v,j}) \\
&= \frac{2}{m} \sum_{w=1}^{m} sig_{w,j} - \frac{4}{m(m-1)} \sum_{u=1}^{m-1} \sum_{v=u+1}^{m} (sig_{u,j} \cdot sig_{v,j})
\end{aligned}
$$

With

$$\sum_{w=1}^{m} sig_{w,j} = \gamma$$

and

$$\sum_{u=1}^{m-1} \sum_{v=u+1}^{m} (sig_{u,j} \cdot sig_{v,j}) = \frac{\gamma(\gamma-1)}{2}$$

we get:

$$HD_{plot}(\gamma) = 2 \cdot \left( \frac{\gamma}{m} - \frac{\gamma(\gamma-1)}{m(m-1)} \right)$$

As Figure 1 shows, the only difference between $HD_{metric}$ and $BA_{metric}$ is that $HD_{metric}$ penalises deviations from $\gamma = \frac{m}{2}$ less than $BA_{metric}$. In fact, the value of $HD_{metric}$ can even be directly calculated from the $BA_j$ values by transforming the $BA_{plot}$ values of each signature bit $j$ into the corresponding $HD_{plot}$ value, and then taking the average of all bits:

For each bit $j$ the value of $BA_j$ is transformed into $BA_j^{norm}$ as defined above:

$$BA_j^{norm} = 0.5 - |0.5 - BA_j|$$

Then the inverse function of $BA_{plot}$ calculates the corresponding $\gamma_j$ for that $BA_j^{norm}$ (as the plots are symmetric, it does not matter that we only get values $\gamma_j \leq \frac{m}{2}$):

$$\gamma_j = m \cdot BA_j^{norm}$$

This $\gamma_j$ is then inserted into $HD_{plot}$ for each bit $j$ and the average of the results is calculated:

$$HD_{metric} = \frac{1}{n} \sum_{j=1}^{n} \left( 2 \cdot \left( \frac{\gamma_j}{m} - \frac{\gamma_j(\gamma_j - 1)}{m(m-1)} \right) \right)$$

## 4 New Correlation Sensitive Metric (CSM)

As mentioned in the previous section, the traditional uniqueness metrics only analyse one signature bit at a time and – if applicable – calculate the average over all bits. Thus, correlations between signature bits are not registered. The undesirable effect of that can be illustrated by the following example. Let $\frac{m}{2}$ devices have the signature $11\ldots1100\ldots00$ and the other $\frac{m}{2}$ devices have the signature $00\ldots0011\ldots11$. Both $HD_{metric}$ and $BA_j$ would produce their optimal values of $\sim 0.5$, while the signatures are not actually unique.

Hence, good outcomes of $HD_{metric}$ or $BA_j$ are only necessary conditions for inter-device uniqueness but not sufficient – unless the a priori assumption is made that the signature bits are uncorrelated. In our view, however, a metric quantifying inter-device uniqueness should not rely on such an assumption, because unforeseen correlations might always be there. This is why we are presenting a new alternative metric.

### 4.1 Definition of New Metric

To identify correlated signature bits, we suggest to compare the outcomes of the $n$ signature bits of each device $i$ with one another, resulting in $\frac{n\cdot(n-1)}{2}$ pairings:

$$\begin{array}{cccc}
cor^i_{1,2} & cor^i_{1,3} & \cdots & cor^i_{1,n} \\
 & cor^i_{2,3} & \cdots & cor^i_{2,n} \\
 & & \cdots & \cdots \\
 & & & cor^i_{n-1,n}
\end{array}$$

with

$$cor^i_{j,k} = \begin{cases} 1, & \text{if } sig_{i,j} = sig_{i,k} \\ -1, & \text{otherwise} \end{cases}$$

The $cor^i_{j,k}$ values for each pairing are then summed up over all $m$ devices:

$$cor_{j,k} = \frac{1}{m} \sum_{i=1}^{m} cor^i_{j,k}$$

such that $cor_{j,k} = 1.0$ means a positive correlation ($sig_{i,j} = sig_{i,k}$ on all devices) and $cor_{j,k} = -1.0$ means a negative correlation ($sig_{i,j} \neq sig_{i,k}$ on all devices). $cor_{j,k} \approx 0.0$, on the other hand, indicates the absence of systemic correlations between the respective bits $j$ and $k$.

There are $2m+1$ many possible outcomes for each of the $\frac{n\cdot(n-1)}{2}$ many $cor_{j,k}$ values:

$$-1.0 = \frac{-m}{m}, \frac{-m+1}{m}, \ldots, \frac{-1}{m}, 0, \frac{1}{m}, \ldots, \frac{m-1}{m}, \frac{m}{m} = 1.0$$

To evaluate these outcomes, we suggest a plot as shown in Figure 2. The x-value of a dot represents how often the y-value occurred as $cor_{j,k}$. To estimate the PUF uniqueness quality, a plot gained from empirical experiments is compared with the theoretically ideal plot that would be expected from completely uncorrelated bits. The ideal plot is generated as follows.

For completely uncorrelated signature bits, the outcome of $cor^i_{j,k}$ is like a coin toss, so we have $P(cor^i_{j,k} = 1) = P(cor^i_{j,k} = -1) = 0.5$. Hence, the probabilities for the outcome of each $cor_{j,k}$ value are:

| $cor_{j,k}$ | $P$ |
|---|---|
| $-1.0 = \frac{-m}{m}$ | $\binom{0}{m} \cdot 0.5^m$ |
| $\frac{-m+1}{m}$ | $\binom{1}{m} \cdot 0.5^m$ |
| $\frac{-m+2}{m}$ | $\binom{2}{m} \cdot 0.5^m$ |
| ... | ... |
| $\frac{-1}{m}$ | $\binom{\frac{m}{2}-1}{m} \cdot 0.5^m$ |
| 0 | $\binom{\frac{m}{2}}{m} \cdot 0.5^m$ |
| $\frac{1}{m}$ | $\binom{\frac{m}{2}+1}{m} \cdot 0.5^m$ |
| ... | ... |
| $\frac{m-2}{m}$ | $\binom{m-2}{m} \cdot 0.5^m$ |
| $\frac{m-1}{m}$ | $\binom{m-1}{m} \cdot 0.5^m$ |
| $1.0 = \frac{m}{m}$ | $\binom{m}{m} \cdot 0.5^m$ |

The fact that

$$\sum_{k=0}^{m} \binom{m}{k} \cdot 0.5^m = 2^m \cdot 0.5^m = 1.0$$

confirms this consideration.

### 4.2 Evaluation Capabilities of New Metric

When the occurrences of all empirically observed $cor_{j,k}$ outcomes are to be compared to the ideal distribution, the ideal probabilities can be multiplied by $\frac{n\cdot(n-1)}{2}$ (amount of $cor_{j,k}$ values) to get the expected amounts of the empirical outcomes. Alternatively, the observed outcomes can be divided by $\frac{n\cdot(n-1)}{2}$, which is preferable as it allows the comparison of PUFs with different $m$ and $n$.

Figure 2 shows the plot from an actual study done by the authors with $m = 72$ FPGA and a ring-oscillator (RO)-PUF producing $n = 3160$ signature bits. The red × plot stems from a flawed implementation in which some ROs suffered from systemic biases. As a result, several signature bits are the same on all devices. This poor bit-aliasing is indicated by the high occurrences of $cor_{j,k} = -1$ and $cor_{j,k} = 1$, because a pair of bits with bit-aliasing – that are either 0 or 1 on all devices – is always correlated. The blue + plot stems from an improved implementation overcoming the RO biases. The improvement is immediately visible in the plot.

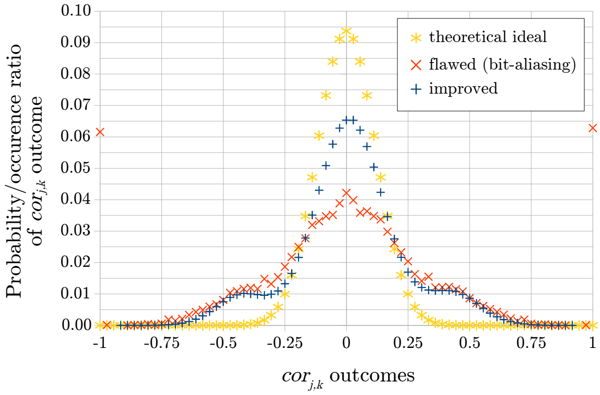

**Fig. 2.** The evaluation of the new metric is done with a plot. Each dot represents how often (x-value) a $cor_{j,k}$ (y-value) was observed. For completely uncorrelated bits, the occurrences are expected to be in a bell curve around zero.

The theoretical ideal of completely uncorrelated signature bits (plot of yellow stars) is not achieved, because of another interesting phenomenon made visible by the new metric: There are more occurrences around $cor_{j,k} = \pm\frac{1}{3}$ than in the ideal case. The reason for this is that the signature bits of RO-PUFs are generated by comparing two ROs per bit. Some bits use the same RO as others, such that the frequency ranking of that RO has an influence on both bits, giving them a slightly increased overall correlation.

Finally, we revisit the example from the beginning of this section, where the signatures of $\frac{m}{2}$ devices have been the bitwise inverse of the other $\frac{m}{2}$ devices' signatures. While $BA_j$ and equivalently $HD_{metric}$ do not register this undesirable anomaly, the new metric produces a plot showing that there are only occurrences of $cor_{j,k} = -1$ and $cor_{j,k} = 1$, because each “0” bit is 100% positively correlated with every other “0” bit, and 100% negatively correlated with every “1” bit.

## 5 Conclusion

We showed, that the CSM is not only able to identify the same anomalies as the traditional metrics, but that it furthermore detects correlations between signature bits the traditional metrics cannot detect. However, $BA_j$ is still a valuable addition to CSM, because $BA_j$ allows to single out signature bits with poor bit-aliasing directly. The plot visualisation of CSM does not allow to immediately recognise which bits are correlated, it only shows if there are any. However, the generation algorithm can easily be setup to write down the indices $j$ and $k$ of all bits with suspicious $cor_{j,k}$ outcomes.

For future work, the authors intent to take into account further means of quantifying the uniqueness of PUF signatures, e.g. Shannon Entropy [11] or the effectiveness of data compression algorithms [15,3].

## References

1. C. Bohm, M. Hofer, and W. Pribyl. A microcontroller SRAM-PUF. In *Network and System Security (NSS), 2011 5th International Conference on*, pages 269–273, Sept 2011.
2. L. Bossuet, X. T. Ngo, Z. Cherif, and V. Fischer. A PUF based on a transient effect ring oscillator and insensitive to locking phenomenon. *Emerging Topics in Computing, IEEE Transactions on*, 2(1):30–36, March 2014.
3. M. Claes, V. Leest, and A. Braeken. *Information Security Technology for Applications: 16th Nordic Conference on Secure IT Systems, NordSec 2011, Tallinn, Estonia, October 26-28, 2011, Revised Selected Papers*, chapter Comparison of SRAM and FF PUF in 65nm Technology, pages 47–64. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
4. S. Eiroa and I. Baturone. An analysis of ring oscillator PUF behavior on FPGAs. In *Field-Programmable Technology (FPT), 2011 International Conference on*, pages 1–4, Dec 2011.
5. L. Feiten, A. Spilla, M. Sauer, T. Schubert, and B. Becker. Implementation and analysis of ring oscillator PUFs on 60 nm Altera Cyclone FPGAs. *Information Security Journal: A Global Perspective*, 22(5-6):265–273, 2013.
6. J. Guajardo, S. S. Kumar, G.-J. Schrijen, and P. Tuyls. FPGA intrinsic PUFs and their use for IP protection. In *Proceedings of the 9th International Workshop on Cryptographic Hardware and Embedded Systems*, pages 63–80, Berlin, Heidelberg, 2007. Springer-Verlag.
7. B. Habib, K. Gaj, and J. P. Kaps. FPGA PUF based on programmable LUT delays. In *Digital System Design (DSD), 2013 Euromicro Conference on*, pages 697–704, Sept 2013.
8. R. Kumar and W. Burleson. On design of a highly secure PUF based on non-linear current mirrors. In *Hardware-Oriented Security and Trust (HOST), 2014 IEEE International Symposium on*, pages 38–43, May 2014.

9. 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. D. Lim, J. Lee, B. Gassend, G. Suh, M. van Dijk, and S. Devadas. Extracting secret keys from integrated circuits. *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, 13(10):1200–1205, Oct 2005.
11. R. Maes, A. Herrewege, and I. Verbauwhede. *Cryptographic Hardware and Embedded Systems – CHES 2012: 14th International Workshop, Leuven, Belgium, September 9-12, 2012. Proceedings*, chapter PUFKY: A Fully Functional PUF-Based Cryptographic Key Generator, pages 302–319. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
12. A. Maiti, J. Casarona, L. McHale, and P. Schaumont. A large scale characterization of RO-PUF. In *Proceedings of the 2010 IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, pages 94–99, June 2010.
13. D. P. Sahoo, D. Mukhopadhyay, and R. S. Chakraborty. Design of low area-overhead ring oscillator PUF with large challenge space. In *Reconfigurable Computing and FPGAs (ReConFig), 2013 International Conference on*, pages 1–6, Dec 2013.
14. G. E. Suh and S. Devadas. Physical unclonable functions for device authentication and secret key generation. In *Proceedings of the 44th Annual Design Automation Conference*, pages 9–14, 2007.
15. V. van der Leest, G.-J. Schrijen, H. Handschuh, and P. Tuyls. Hardware intrinsic security from D flip-flops. In *Proceedings of the Fifth ACM Workshop on Scalable Trusted Computing*, STC '10, pages 53–62, New York, NY, USA, 2010. ACM.
16. C.-E. Yin and G. Qu. Temperature-aware cooperative ring oscillator PUF. In *Proceedings of the 2009 IEEE International Workshop on Hardware-Oriented Security and Trust*, pages 36–42, 2009.
17. H. Yu, P. Leong, H. Hinkelmann, L. Moller, M. Glesner, and P. Zipf. Towards a unique FPGA-based identification circuit using process variations. In *International Conference on Field Programmable Logic and Applications, 2009. FPL 2009.*, pages 397–402, Aug. 2009.