Trusted Integrated Circuits: A Nondestructive Hidden Characteristics Extraction Approach

Yousra Alkabani[1], Farinaz Koushanfar[1,2], Negar Kiyavash[3], and Miodrag Potkonjak[4]

[1] Rice University CS Dept.
[2] Rice University ECE Dept.
[3] UIUC CS Dept.
[4] UCLA CS Dept.

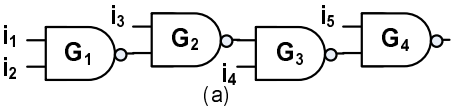
Abstract. We have developed a methodology for unique identification of integrated circuits (ICs) that addresses *untrusted fabrication* and other security problems. The new method leverages *nondestructive gate-level characterization* of ICs post-manufacturing, revealing the hidden and *unclonable* uniqueness of each IC. The IC characterization uses the externally measured leakage currents for multiple input vectors. We have derived several optimization techniques for gate-level characterization. The probability of collision of IDs in presence of intra- and inter-chip correlations is computed. We also introduce a number of novel security and authentication protocols, such as *hardware metering*, *challenge-based authentication* and *prevention of software piracy*, that leverage the extraction of a unique ID for each IC. Experimental evaluations of the proposed approach on a large set of benchmark examples reveals its effectiveness even in presence of measurement errors.

1 Introduction

Recently, manufacturing variability (MV) emerged as a mechanism for providing IC security [11,12]. It was used for tasks such as authentication [4,10,19]. So far, the work in this area has relied on addition of new circuitry or specialized processes to achieve security. In order to address this limitation, we have developed a method for accurate characterization of an arbitrary IC at the gate level. Characterization is done in a nondestructive way, without the need for additional circuitry or special processes. The extracted characteristics can be translated in a unique and unclonable ID for each IC and form the starting point for the creation of variety of security protocols.

Probably the best way to introduce the new security approach is to consider a small example shown in Figure 1 that consists of 4 NAND gates with two inputs (NAND2). Table (b) shows the leakage current of a nominal NAND2 gate for different sets of inputs. However, in deep-submicron technologies, the current greatly varies from one IC to another due to MV. For example, in 65nm technology, the leakage currents of the same gate on two different ICs may scale by a factor of 20 [14]. Table (c) shows an example of possible scaling factors for

K. Solanki, K. Sullivan, and U. Madhow (Eds.): IH 2008, LNCS 5284, pp. 102–117, 2008.



(a)

NAND2	
Input	Leakage (nA)
00	37.84
01	100.3
10	95.7
11	454.5

(b)

Gate	Scaling factor	
	IC 1	IC 2
G_1	0.5	2.4
G_2	1.3	0.6
G_3	2.1	4
G_4	3	0.9

(c)

Input vector	Leakage (nA)	
	IC 1	IC 2
00011	1391	2055
10101	2082	1063
01110	1243	2150
11001	1841	1905

(d)

Fig. 1. (a) A design consisting of 4 NAND2 gates, (b) leakage current vs. input for NAND2,(c) scaling factors of gates on two ICs, and (d) total leakages of ICs for different input vectors

the four gates in two circuits, denoted by IC1 and IC2. Table (d) shows leakage power in IC1 and IC2 for different input vectors.

It is easy to see that from these measurements, we can calculate the leakage power of each gate in both ICs. Once the scaling factor of each gate is extracted, we can use it as the circuit's ID. For instance, if we standardize that a scaling factor larger than 1 is denoted by one in the pertinent ID and zero otherwise, the ID of IC1 is 0111 and the ID of IC2 is 1010. For a more realistic IC with millions of gates, limited number of primary inputs and outputs, and a limited number of scanned flip-flops, the gate characterization task is much more challenging, but the chances for collision of IDs for any two ICs are much lower. In this paper, we introduce and analyze such characteristic extraction, ID assignment and uniqueness evaluation techniques.

We have two key conceptual and technical goals. The first is to demonstrate how nondestructive techniques for ICs can be used to extract unique and unclonable IDs from each chip for a given design. The second strategic objective is to introduce a new spectrum of security protocols that leverage the unique and irreproducible IDs by connecting them to the functionality of hardware or software executed on the IC. To achieve the two goals, one needs to address a system of demanding engineering, optimization, and modeling tasks. In the remainder of the manuscript, we show that the tasks can be solved in very elegant and efficient ways.

2 Related Work

The idea of adding circuitry that uses the manufacturing variability for generating a unique random ID for each IC with a single mask was proposed by a number of authors [11,12,18]. The IDs do not provide a measure of trust, as they are separated from the functionality and thus, are easy to tamper and remove.

A group of MIT researchers has focused on the idea of utilizing the variability-based delays for security and authentication purposes [4,10,19]. They add a circuitry that implements the physically unclonable function (PUF) which maps a set of challenges to a set of responses. PUFs are unique since the process variations result in significant fluctuations in delays of chips coming from the same mask. PUFs ensure that each IC has a unique set of outputs for each input vector. The method stores a database of the challenge-response pairs for each IC. An IC is authenticated when it correctly responds to the output of one or more challenge inputs.

Note that even though the PUF-based and other random ID generation circuits find usage in authentication devices and other security scenarios, they are radically different from the novel scheme proposed in this paper. Unlike the earlier approaches [11,12,18,4,10,19], the new method does not require addition of any circuitry, database of challenge-response sets, or special process technology. All what is needed is storing the unique ID for each chip. Furthermore, we will show how noninvasive ID extraction from the I/O measurements from the external pins can be used in new security and protection mechanisms.

In FPGA and other programmable platforms adding unique programmable fingerprints for each IC was proposed [9], but the techniques are not directly applicable to application specific integrated circuits (ASICs). In ASICs, giving a unique ID to each IC by adding a small programmable part to the control path of the design post-fabrication was pursued as well [8]. The technique does not exploit the manufacturing variability and only leverages the equivalence of various synthesized control paths to identify each circuit.

The prior work in trusted IC also consists of a number of watermarking schemes [21,15,7,16]. A comprehensive survey of fingerprinting and watermarking schemes can be found in [16]. Note that, watermarking is a radically different problem than unique identification. Watermarking addresses the problem of uniquely identifying each intellectual property (IP) core and not each IC.

3 Preliminaries

3.1 Background

Circuit model. The full specification of the circuit's functionality from input/output is assumed to be publicly available. The designer has post-synthesis design knowledge, including the exact mapping of the logic to gates in the technology library. The table that specifies leakage current values for each library gate versus the input state is also available.

Model of variations. Intensive scaling of CMOS to its physical limitations results in high variability of circuit-level parameters such as delay and leakage. Variations may be temporal or spatial. Our method leverages the spatial variations while it alleviates the temporal ones by introducing robust measures. Spatial variations are divided into two categories: (i) inter-chip variations, or the chip-to-chip fluctuations; and (ii) intra-chip variations, present inside one

chip. We employ the variation model described in [17] where the multivariate normal distribution (MVN) is used for modeling all random components across the chip and intra-chip correlations. Furthermore, the grid model that partitions the space into grids is used; devices within the same grid are highly correlated and devices in further grids are correlated proportional to their distances.

Model of the leakage current. Leakage current is also a function of the process fluctuations. The leakage model we use here was proposed in [1]: the model considers the subthreshold leakage (I_{sub}) and the gate tunneling leakage (I_{gate}) for each gate. Both currents are modeled as exponential functions that can be approximated by a lognormal distribution. The full-chip leakage distribution is the sum of the lognormal distributions of all gates considering spatial correlations. The sum is not theoretically known to have a closed form, but can be well approximated as a lognormal distribution using Wilkinson's method [1].

3.2 Flow

The proposed unique chip identification has three phases: (i) gate-level characterization method and improving its performance; (ii) translation of gate characteristics into IDs.

Phase (i). The gate level characteristics that are subject to variations are extracted, e.g., such as the gate delays on a specific IC or its leakage current. In this paper we use the leakage current because of its properties such as high variability, coverage of all gates in each measurements, and suitability for treatment using provably optimal optimization techniques of polynomial complexity such as linear programming. It is also possible to add interconnect leakages but we did not include it since it has much less visible fluctuations.

The first phase itself has four steps: (i) input vector generation applying multiple inputs to the circuit; (ii) execution and analysis of measurements where the goal is to conduct measurements in the most effective way and to characterize errors; (iii) solving the systems of equations obtained by using the measurements. Depending on the error model and the structure of equations formed, different methods may be used;and (iv) statistical analysis to evaluate results' stability.

A key observation is that not all gate scaling factors have to be extracted for ID creation. In a number of relatively common designs, it is impossible to extract the characteristics of all gates. Note that there are a few techniques that indirectly alleviate requirements for very accurate measurements. For instance, if one increases the supply voltage or temperature, the leakage increases making the relative errors of measurements lower for the same measurement equipment.

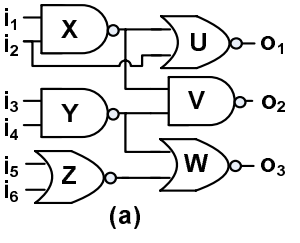
Phase (ii). In this phase, the gate characteristics are translated (coded) into the corresponding IDs. We have developed two techniques that emphasize different trade-offs between the resiliency against measurement errors and circuit variations on one side and probability of collision on the other side.

Phase (iii). Once we have gate level characteristics for a sufficiently large number of gates and the coded IDs, we address potential security attacks by analyzing the technique with respect to the likelihood of ID collision.

Note that many new cryptographical, security, IPP and DRM protocols are enabled by the proposed method. The main idea is to leverage intrinsic and unclonable unique chip ID to create a task that can be easily completed in a short period of time on the IC, and requires many orders of time longer if one does not have access to the IC. In interest of brevity, we omit the discussion in this paper.

4 Nondestructive Extraction of IC Characteristics

The goal is to extract the gate-level properties of an IC in a noninvasive way (by external I/O measurement) such that the overhead in terms of the time and resources is minimized, while the method does not require a specific instrumentation other than the classical test equipment. Note that the destructive measurement of ICs for the purpose of characteristics extraction is done in industrial practice, but the method renders the IC unusable after testing [3]. Other noninvasive methods such as X-ray imaging are nonefficient and slow.



Input	Leakage (nA)	
	NAND2	NOR2
00	37.84	250.6
01	100.3	227.2
10	95.7	213
11	454.5	92.05

(b)

Fig. 2. (a) A small circuit consisting of NAND2 and NOR2 gates, (b) leakage current vs. input of the gates

Perhaps an example would be the best way to illustrate the nondestructive extraction method by using the leakage current measurements. In Figure 2(a) we show a small circuit consisting of 6 NAND2 and NOR2 gates X, Y, Z, U, V, and W. The circuit has 6 primary inputs and three primary outputs. The table in Figure 2(b) presents the leakage current of the two gate types versus the input to the gate, where the gate leakage has a strong dependence on its inputs [1].

Assume that all the gate inputs are set to 0, i.e., $\{i_1,i_2,i_3,i_4,i_5,i_6\}$=000000. On each chip one can measure the leakage current resulting from this input, denoted by I_{leak}(000000). The error in this measurement is denoted by e_1. The leakage current of the circuit can be written in terms of the individual gates. Let us denote the leakage currents of the nominal-sized 2-input NOR and NAND gates by $I_{NAND}(.)$ and $I_{NOR}(.)$ respectively, where the arguments inside the parentheses are the inputs to the gate. Because of MV the exact gate leakage

[1] Such tables are standard and readily available. The table in Figure 2(b) is directly taken from the leakage current measurements by Yuan and Qu [22].

values deviate from the nominal value. We use a scaling coefficient times the nominal value to express the exact leakage of the gate. For the gates in Figure 2(a), we denote the scaling factors by s_X, s_Y, s_Z, s_U, s_V, and s_W.

$$\begin{aligned} I_{leak}(000000) + e_1 &= s_X I_{NAND}(00) + s_Y I_{NAND}(00) \\ &\quad + s_Z I_{NOR}(00) + s_U I_{NOR}(01) \\ &\quad + s_V I_{NAND}(11) + s_W I_{NOR}(11); \\ I_{leak}(010101) + e_2 &= s_X I_{NAND}(01) + s_Y I_{NAND}(01) \\ &\quad + s_Z I_{NOR}(01) + s_U I_{NOR}(01) \\ &\quad + s_V I_{NAND}(00) + s_W I_{NOR}(01); \\ I_{leak}(\ldots) + e_i &= \ldots; \qquad i = 4 \ldots, M \end{aligned} \tag{1}$$

Similarly, one can apply M different inputs to the circuit and write M linear relationships for the measured leakage as demonstrated in the Equation Set 1. To find the unknown scaling factors, we form an optimization framework, where the equations are the constraints of the optimization and the objective function (OF) is to optimize a specific norm of the measurement error. Let $f(E)$ denote a function for measuring a metric of errors, where $E{=}\{e_i\}_{i=1}^{M}$. The OF is written as min $f(E)$. Common forms of the f that are used in minimization are the L_p norms of error that are defined as: $L_p{=}(\sum_{m=1}^{M} w_m|e_m|^p)^{1/p}$ for $1 \leq p < \infty$; and $L_p{=}\max_{m=1}^{M} w_m|e_m|$ if $p = \infty$.

Independence of the Equations. While in principle one would be able to solve for and to find the scaling factor of each gate, it is possible to have ambiguous gates. The ambiguous gates are those whose combination always gets the same ratio of coefficients and are indistinguishable. As a simple example for an ambiguous combination take three inverters A, B, and C with scaling factors s_A, s_B, and s_C that are placed in series in the middle of the circuit. No matter which input combination is used, the term $s_A I_{inv}(0) + s_B I_{inv}(1) + s_C I_{inv}(0)$ or the term $s_A I_{inv}(1) + s_B I_i nv(0) + s_C I_{inv}(1)$ would be present in the circuit. It is impossible to distinguish between s_A and s_C since they always have the same coefficient. Even though the three inverter example is an extreme case, in real circuits, ambiguous cases occur (largely because of reconvergent fanout) and should be taken into account. We add a check that scans the equations and figures out the gate combinations that are ambiguous and consolidates them into one entity. The characteristics of the ambiguous gates will not be used in the final identification scheme.

Solving the Optimization Problem. The optimization problem may take many different formats depending on the form of the objective function. Generally speaking, the L_p error norms and the maximum likelihood determine a nonlinear OF with linear constraints which requires a nonlinear optimization method. Although many nonlinear solvers exist, it is well-known that the general non-linear optimization with uncertainty is prone to get stuck at local minima

and may not be optimally solved. However, for a common class of the nondestructive extraction problems it is possible to cast the problems as linear, quadratic, or convex optimization that are easier to solve:

1. In case of minimizing L_1 norm, the optimization problem can be written in form of a linear program as follows: min $\sum_{m=1}^{M} |e_m|$, subject to the M constraints presented in Equation Set 1. The absolute function is nonlinear, but we convert it to a linear one by introducing m auxiliary variables e_m^+ and adding $2m$ constraints, i.e., for each m, $e_m^+ \geq e_m$ and $e_m^+ \geq -e_m$. The linear objective function is: min $\sum_{m=1}^{M} e_m^+$.
2. In case of the L_2 norm, the OF is: min $\sqrt{\sum_{m=1}^{M} e_m^2}$, whose minimization is equivalent to min $\sum_{m=1}^{M} e_m^2$. The optimization is a quadratic program.
3. To minimize the L_∞ norm, we define a new variable e_{max}. The OF is then min e_{max}. We also add constraints that satisfy the L_∞ requirements, i.e., $e_{max} \leq e_m$ for m=1, ..., M that can be also solved by a linear program.
4. If one assumes the errors are i.i.d Gaussian distribution $\mathcal{N}(0, \sigma^2)$, The log-likelihood function would be: $\max \sum_{m=1}^{M} log(\exp -\frac{e_m^2}{2\sigma^2}) \equiv \max \sum_{m=1}^{M} -e_m^2 \equiv \min \sum_{m=1}^{M} e_m^2$. This is equivalent to the quadratic program that minimizes the L_2 norm of the errors.

5 Formation and Analysis of IDs

5.1 Robust Coding for IDs

We form digital IDs using the extracted characteristics of the gates in the following way. The starting points are the analog values of the scaling factors. The values are quantized and concatenated to form digital IDs. We enforce a number of desiderata on this process, including low computational requirement, low probability of collision of two IDs, and stability and robustness against IC aging [6] and change in environmental conditions such as supply voltage and temperature. Note that the gate ordering and the coding methods must be easy to check and standardized. To have a suitable and standard notation, we introduce an *indicator string* for each IC. The length of the indicator string is equal to the number of gates in the design's netlist. The order of the gates in the netlist corresponds to their x and y placement coordinates. If a gate has a lower x coordinate, it has a lower position. If two gates have equal x coordinates, the gate with the smaller y, has a lower position. We observed that not all the characteristics of all ICs were extractable. Luckily, not all of them are needed for identifying one IC because on modern designs there are many gates. If a gate is used for identification, its corresponding indicator bit would be 1. Otherwise, it is 0. To improve stability and robustness, we calculate the normalized characteristics against leakage power of all gates on the pertinent IC. It has been experimentally proven that the environmental variations mostly change the characteristics of all the affected gates in the same way [20].

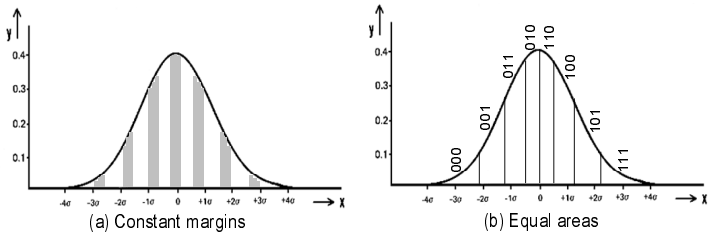


Fig. 3. Selecting identification codes that are robust: (a) margin coding with constant margins, and (b) equiareal coding with equal areas

Robust Translation into Codes. When converting the analog values into binary, it is important to consider the distribution of the analog values. After extracting the IC characteristics, we create a histogram of of the extracted values. We use two different methods for binary coding: (i) margin coding, and (ii) equiareal coding. Both schemes use the probability density function (PDF) of the extracted characteristics, obtained by smoothing of the histogram.

Margin coding. This scheme is illustrated in Figure 3(a). It finds a robust binary conversion of the analog codes by partitioning the PDF into regions. The regions are either approved (shown as white on the figure) or banned (shown as dark). The white and dark regions are interleaved. If the characteristics of a gate falls into the banned region, it will not be considered in the ID formation and its corresponding bit would be set to zero in the indicator string. If an analog value falls in the approved region, the corresponding value in the indicator string is 1. Each approved analog value obtains a binary code of its partition. The length of the margin codes depend on the number of partitions. The large number of partitions increases the length of IDs and decreases the robustness of margins. Thus, the number of partitions should provide a good trade-off between the length of the IDs and robustness. Note that, the dark partitions have uniform size as shown in Figure 3(a). If we denote the width of one partition by w, then we can detect any changes in the characteristics of amplitude w or less and correct the changes of the maximum size $\frac{w}{2}$.

Equiareal coding. This coding scheme partitions the region into segments so that the area under the PDF curve of all segments are equal. Thus, the probability of having a characteristic belonging to one area is the same for all segments in order to maximize the entropy of IDs. An example is shown in Figure 3(b) where the PDF is partitioned into 8 segments and thus, a 3-bit code can be used to identify each segment. All segments are used for ID creation. The robustness is achieved by taking into account the order of segments. Hence, changing the value by a relatively small amount translates into shifting to previous or the next segment. The encoding scheme takes into account such changes. For example, assume that in Figure 3(b), the scale factor of a gate was encoded as 011. For ID verification, the scale factor of the same gate is considered to be equal to

any gate that has code 001, 011, or 010. So, the assigned IDs are robust against variations as long as the characteristics stays in the adjacent segment.

5.2 Probabilistic Analysis

One of the best-known probabilistic problems frequently used in cryptography is the birthday problem [2] which can be abstracted as: *What is the probability that among K possible objects drawn from the population* $\{1, \ldots, n\}$ *at least two have the same value.* Our problem requires solving generalized version of birthday problem with non-uniform probabilities that is defined as: *Given a collection of K binary sequences, each of length M; what is the probability of collision among the K strings?*

The above problem is a generalization of the birthday problem in the sense that, the strings are not required to be equally likely. There are total of $n = 2^M$ binary length M sequences. Let P_i denote the probability that the ID of the IC is sequence i and let $\mathbf{P} = (P_1, \ldots, P_n)$ be the collection of probabilities of all $n = 2^M$ sequences. In case of birthday problem all the probabilities P_i are equally likely and we can represent them by $\mathbf{n}^{-1} = (\frac{1}{n}, \ldots, \frac{1}{n})$. Then probability of no match between K sequences is given by $P(M, K, \mathbf{n}^{-1}) = \frac{2^M!}{K^{2^M}(2^M-K)!}$, and the probability of collision is $P(collision) = 1 - \frac{2^M!}{K^{2^M}(2^M-K)!}$. When the sequences are not equally likely, probability of collision is given by

$$\begin{aligned} P(collision) &= 1 - P(M, K, \mathbf{P}) \\ &= 1 - K! \sum_{1 \le \nu_1 < \ldots < \nu_K \le n} P_{\nu_1} P_{\nu_2} \ldots P_{\nu_K}, \end{aligned} \tag{2}$$

where $P(M, K, \mathbf{P})$ is the complimentary probability that no collision occurs.

As long as probability of the sequences $\mathbf{P}$ is specified, (2) gives the collision probability of the IDs. Depending on the problem formulation, we consider the following three cases for specifying $\mathbf{P}$:

- **Case 1:** The bits in each sequence are independent and identically distributed (i.i.d.) with P(any bit is 1)= π, P(any bit is 0)= 1-π. Then, P_i is $P_i = (1-\pi)^{n_{0i}} \pi^{n_{1i}}$, where n_{0i} denotes the total number of zeros in sequence i and n_{1i} denotes the total number of its ones.
- **Case 2:** The bits in each sequence are independent but not identically distributed with P(bit m is 1)= π_m, Prob (bit m is 0)= 1-π_m. Let us define the following function

$$I(b_m) = \begin{cases} \pi_m, & b_m = 1 \\ 1 - \pi_m, & b_m = 0 \end{cases},$$

  then P_i is $P_i = \prod_{m=1}^{M} I(b_m)$.
- **Case 3:** The bits in each sequence are correlated and their Cumulative Distribution Function (CDF) is $\mathbf{P}$.

The problem with (2) is that it involves sums of exponential number of terms. For a large number unique IDs, the exact solution of (2) is intractable. Hence,

we suggest Nunnikhoven's approximation [13] details of which are omitted in interest of brevity for calculating the collision probability of (2).

5.3 Statistical Analysis

While the analysis of Section 5.2 takes into account the situations where the probability of IDs are related because of inter-chip correlations, it ignores the existence of the intra-chip correlations. In general, modeling the intra-chip spatial correlation is a complex problem. Hence, we perform nonparametric statistical Monte Carlo (MC) simulations to estimate the collision probability.

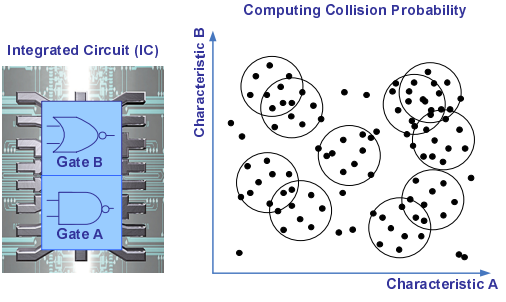


Fig. 4. An example of using 2 gates to identify the IC characteristics. The probability of collision between the ICs is calculated using the MC simulation on the 2D plot.

Figure 4 illustrates the MC-based simulations when only characteristics of two gates are used. The extracted analog characteristic (scaling factor) of each gate is used as one dimension of our analysis. Thus, we have a 2D space in this example. In the general case of using M' characteristics, we have a M'-dimensional space. Now, assume that we take K such ICs, and we would like to calculate the probability of collision. For computing the probability, we randomly position a sphere in the M' dimensions multiple times.

The center of each sphere is selected at the position of a new IC. Assuming that the coding that is done on the analog characteristics is robust, the points that fall within one circle will be assigned the same IDs. Since we repeat the MC experiment (i.e., circle generation) multiple times, we can count the number of possible points inside each circle because of the finite resolution of the measurements. Using the assumption and the multiple measurements, one can easily calculate the probability of collision for the whole space as follows,

$$P(collision) = \sum_{n=1}^{N_{MC}} P(collision|C_n)P(C_n) \tag{3}$$

where N_{MC} denotes the number of MC-simulation runs, n is the index of each simulation, and C_n is the generated spheres for one experiment. The parameters

of the MC analysis are the radius of the sphere and the number of random runs (N_{MC}) that must be carefully selected. In our experiments, we select the radius to be equal to the robust margin of the codes.

6 Experimental Results

In this section, we present the experimental evaluations of the new approach. We first show the results for the nondestructive extraction of chip characteristics and in the second subsection shows the probabilistic and statistical analysis of the collision of keys and robustness.

6.1 Nondestructive Extraction of Chip Characteristics

Our approach was tested on circuits from the MCNC'91 benchmarks. We used SIS and CPLEX to perform synthesis and LP solving. We used the variation model from Section 3.1. Leakage values of 0.18μm reported by [22] are used. The spatial correlation values decrease with the distance between the grids. There are 20 grids. On each chip, we randomly selected 5 center grids where the variations are the highest and the variations of the other grids are computed by correlations to those centers. The 3σ of the parameter variations of T_{ox} and L was set to 25% (corresponding to the 90nm technology). The levels of inter-chip and intra-chip variations were equal.

Table 1. The characterization error for different error distributions

	characterization error (%)		
error	**Uniform**	**Triangle**	**Gaussian**
0%	0	0	0
5%	0.99	0.75	0.11
10%	2.03	1.48	0.23
15%	3.25	2.34	0.35
20%	4.32	2.92	0.47
25%	5.21	3.7	0.6
30%	6.04	4.46	0.74
35%	7.49	5.2	0.86
40%	8.4	5.63	0.93
45%	9.34	6.06	1.06
50%	9.42	7.62	1.24

Table 1 shows the characterization error when we consider Uniform, Triangular, and Gaussian error measurement distributions for the alu2 circuit. The first column shows the absolute value of relative measurement error (percentage). The next three columns show the average extracted characterization error (percentage) for a measurement error of Uniform, Triangular, and Gaussian distributions respectively. We see that the characterization error is the smallest

when the measurement error has a Gaussian distribution, and largest the for a Uniform distribution. For 25% relative absolute error, the Gaussian distribution leads to 0.6% error in characterization, whereas, Uniform and Triangular distributions yield 5.21% and 3.7% respectively.

Table 2 presents the characterization errors for 15 MCNC'91 benchmarks. The measurement error distribution is Gaussian. The first column shows the benchmark, the next two columns show the number of primary inputs and the number of gates after technology mapping to a subset of the MCNC library. The next two sets of three columns show the error when we optimized the average and the sum of squares of the measurement errors.

For every circuit, we generate a number of equations that is equal to the $\min\{2^{PI}, 3G\}$, where PI is the number of primary inputs, and G is the number of gates in the benchmark. We also tried minimization of the maximum error (L_∞) and it gave similar results.

Table 2. The characterization error for different benchmarks and different absolute relative errors

			L_1(%)			L_2(%)		
BM	**PI**	**gates**	**1%**	**5%**	**10%**	**1%**	**5%**	**10%**
9symml	9	166	0.03	0.14	0.26	0.03	0.13	0.24
alu2	10	356	0.02	0.12	0.25	0.02	0.12	0.24
C1908	33	615	0.01	0.05	0.09	0.01	0.04	0.09
C432	36	200	0.01	0.07	0.12	0.01	0.05	0.11
C8	28	164	0.01	0.04	0.07	0.01	0.04	0.08
C880	60	354	0.01	0.05	0.07	0.01	0.04	0.07
f51m	8	136	0.02	0.09	0.17	0.02	0.1	0.21
i6	138	340	0.01	0.04	0.06	0.01	0.04	0.06
i7	199	405	0.01	0.03	0.05	0	0.03	0.05
lal	26	116	0.01	0.05	0.1	0.01	0.07	0.13
term1	34	363	0.01	0.04	0.06	0.01	0.05	0.09
too_lrg	38	582	0.01	0.04	0.07	0.01	0.04	0.09
ttt2	24	207	0.01	0.03	0.05	0.01	0.04	0.08
x1	51	295	0.01	0.05	0.1	0.01	0.06	0.13
x3	135	742	0.01	0.03	0.05	0.01	0.02	0.05

To study the effect of the number of constraints used for characterization, we generate four different sets of constraints. Figure 5 shows the percentage error in characterization when considering different number of constraints for benchmark lal. The first column shows the relative absolute error used. The rest of the columns show the error in characterization for 500, 1000, 1500, and 2000 constraints. It can be seen that although the error in characterization is very small in the case of using 500 constraints, the error is reduced by going from 500 constraints to 1000 constraints. However, there is almost no improvement for constraints more than 1000.

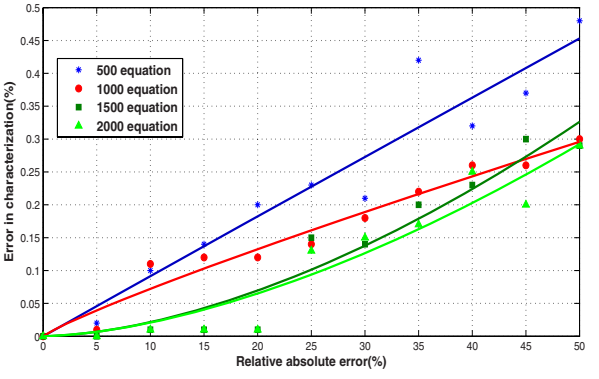


Fig. 5. Percentage characterization error for different number of constraints (la1 benchmark)

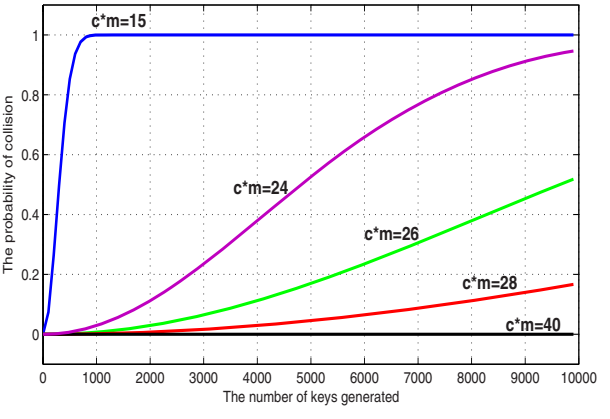


Fig. 6. The probability of collision of different number of keys for different values of code length (c) and number of circuit characteristics (m)

6.2 Evaluation of the Collision Probability

Figure 6 shows the probability of collisions for equiareal coding. The probability of collision is decreased when we increase the product $M = c.m$, where c is the number of code bits and m is the number of gates used for the key generation. In the case of margin coding the probability of the assigned codes were nonuniform and we used the Nunnikhoven's approximation to compute the probabilities. Our studies revealed that for a larger value of $c.m$ this distribution approached the Uniform distribution where the various codes have the same probabilities. Thus, the curves are similar to those in Figure 6 for large $c.m$. The only difference is that in margin coding for c bits codes the corresponding equiareal code with the equal number of partitions will have $c + 1$ bits.

Table 3. Robustness for the equiareal and equidistance (margin) coding schemes

	EQ-AREA			EQ-DIST		
BM	1 bit	2 bits	3 bits	1 bit	2 bits	3 bits
alu2	0.76	0.18	0.09	3.5	1.75	0.86
c8	0.77	0.17	0.08	3.7	1.87	0.93
lal	0.71	0.15	0.07	4.23	2.11	1.06
too_large	0.71	0.2	0.08	3.65	1.82	0.92
x1	0.75	0.23	0.09	3.84	1.92	0.96

Table 3 shows the robustness for the margin coding and equiareal coding for 5 benchmarks. The first column shows the benchmark. The next three columns show the robustness for equiareal codes (EQ-AREA). The last three columns show the robustness for using margin codes (EQ-DIST). We see that the second method increases the robustness at the expense of having shorter IDs.

For the statistical analysis of the probability of collision, we performed the MC simulations on five benchmarks. The number of runs was determined using bootstrapping techniques [5]: for each run number, we did multiple random MC simulations for calculating the probability of collisions. If the results were similar over multiple random instances we stopped increasing the run number since the confidence over the bootstrapped results was high. Table 4 presents the results. The first column shows the benchmark, and the remaining columns show the probability of collisions when considering 25, 50, 75, and 100 gates. For 100 characteristics, the probability of collision was always less than 0.005. It is interesting to compare the results from the statistical analysis to that of the probabilistic, even though one is for the collision of the analog characteristics and the other one for the coded characteristics. For example, for c.m=26 that is comparable to 25 characteristics and 1000 ICs, the probability of collision is much lower in number for Figure 6, as expected.

Table 4. Statistical analysis of collision probability

	Number of characteristics			
BM	25	50	75	100
alu2	0.027	0.011	0.005	0.002
c8	0.014	0.003	0.001	0
lal	0.007	0.001	0	0
too_large	0.032	0.015	0.008	0.0046
x1	0.023	0.007	0.002	0.001

7 Conclusions

We have developed a method for extraction of unique unclonable IDs from each IC of a given design by exploiting deep submicron manufacturing variability. The approach is applicable to legacy designs, and induces no power, area and timing

overhead while enabling a spectrum of novel security and IPP protocols. Experimental results on a large set of industrial benchmark instances demonstrate the efficiency of the proposed methods. The characteristics of an IC could be extracted within 2% accuracy in presence of 50% relative measurement errors. We also demonstrated that for all industrial design, the probability of collision rapidly approached zero.

Acknowledgment

This work is partly supported by the DARPA/MTO Young Faculty Award W911NF-07-1-0198, NSF CT-0716674, and NSF CCF-0729061.

References

1. Chang, H., Sapatnekar, S.: Full-chip analysis of leakage power under process variations, including spatial correlations. In: DAC, pp. 523–528 (2005)
2. Coppersmith, D.: Another birthday attack. In: Williams, H.C. (ed.) CRYPTO 1985. LNCS, vol. 218, pp. 14–17. Springer, Heidelberg (1986)
3. Friedberg, P., et al.: Modeling within-die spatial correlation effects for process-design co-optimization. In: ISQED, pp. 516–521 (2005)
4. Gassend, B., et al.: Identification and authentication of integrated circuits. In: Concurrency and Computation: Practice and Experience, vol. 16, pp. 1077–1098. John Wiley & Sons, Chichester (2004)
5. Hastie, T., et al.: The Elements of Statistical Learning. Springer, Heidelberg (2001)
6. Bernstein, K., et al.: High-performance CMOS variability in the 65-nm regime and beyond. IBM Journal of Research and Development 50(4/5), 433–450 (2006)
7. Kirovski, D., Potkonjak, M.: Local watermarks: methodology and application to behavioral synthesis. IEEE Trans. CAD 22(9), 1277–1283 (2003)
8. Koushanfar, F., et al.: Intellectual property metering. In: IHW, pp. 81–95 (2001)
9. Lach, J., et al.: Fingerprinting digital circuits on programmable hardware. In: IHW, pp. 16–32 (1998)
10. Lee, J., et al.: A technique to build a secret key in integrated circuits for identification and authentication applications. In: Symposium of VLSI, pp. 176–179 (2004)
11. Lofstrom, K., et al.: IC identification circuits using device mismatch. In: ISSCC, pp. 372–373 (2000)
12. Maeda, S., et al.: An artificial fingerprint device (AFD): a study of identification number applications utilizing characteristics variation of polycrystalline silicon TFTs. IEEE Trans. Electron. Devices 50(6), 1451–1458 (2003)
13. Nunnukhoven, T.: A birthday problem solution for nonuniform birthday frequencies. The American Statistician 46(4), 270–274 (1992)
14. Vijaykrishnan, N., Xie, Y.: Reliability concerns in embedded system designs. IEEE Computer 39(1), 118–120 (2006)
15. Oliveira, A.: Techniques for the creation of digital watermarks in sequential circuit designs. IEEE Trans. on CAD 20(9), 1101–1117 (2001)
16. Qu, G., Potkonjak, M.: Intellectual Property Protection in VLSI Design. Kluwer Academic Publishers, Dordrecht (2003)

17. Srivastava, A., et al.: Statistical Analysis and Optimization for VLSI: Timing and Power. Series on Integrated Circuits and Systems. Springer, Heidelberg (2005)
18. Su, Y., et al.: A 1.6 J/bit stable chip ID generating circuit using process variations. In: ISSCC, 406–407 (2007)
19. Suh, G., et al.: Design and implementation of the aegis single-chip secure processor using physical random functions. In: ISCA, pp. 25–36 (2005)
20. Thompson, A., Layzell, P.: Evolution of robustness in an electronics design. In: ICES, pp. 218–228 (2000)
21. Torunoglu, I., Charbon, E.: Watermarking-based copyright protection of sequential functions. JSSC 35(3), 434–440 (2000)
22. Yuan, L., Qu, G.: A combined gate replacement and input vector control approach for leakage current reduction. IEEE Trans. on VLSI 14(2), 173–182 (2006)