A Transmission Gate Physical Unclonable Function and On-Chip Voltage-to-Digital Conversion Technique

R. Chakraborty, C. Lamech
Intel Corp.

D. Acharyya
AdvanTest Inc.

J. Plusquellic
University of New Mexico

ABSTRACT

A physical unclonable function (PUF) is an embedded integrated circuit (IC) structure that is designed to leverage naturally occurring variations to produce a random bitstring. In this paper, we evaluate a PUF which leverages resistance variations which occur in transmission gates (TGs) of ICs. We also investigate a novel on-chip technique for converting the voltage drops produced by TGs into a digital code, i.e., a voltage-to-digital converter (VDC). The analysis is carried out on data measured from chips subjected to temperature variations over the range of -40^{o}C to +85^{o}C and voltage variations of +/- 10% of the nominal supply voltage. The TG PUF and VDC produce high quality bitstrings that perform exceptionally well under statistical metrics including stability, randomness and uniqueness.

Categories and Subject Descriptors

K.6.5 [**Management of Computing and Information Systems**]: Security and Protection -- *Authentication.*

General Terms

Security

Keywords

Hardware security, unique identifier, process variations

1.INTRODUCTION

Physical Unclonable Functions (PUFs) continue to gain momentum as an alternative to embedding secrets in fuses or non-volatile memory of integrated circuits. PUFs derive 'secrets' from variations that occur in the physical parameters of the on-chip wires and transistors. These variations are unique to each chip and, depending on the parameter, can be leveraged to produce large numbers of random bits. PUFs can produce repeatedly random bit strings on the fly, and without the need to store these secrets in digital form on specialized memory structures on chip.

A PUF produces a bit string by applying a set of "challenges" to specially designed circuit primitives and measuring the corresponding "responses". The challenges are typically 'digital' and therefore can be generated on-chip from a pseudo-random number generator such as a linear feedback shift register (LFSR). The challenges are used to configure one or more PUF circuit primitives prior to the application of a stimulus. The stimulus elicits an analog response from the PUF primitives, which is measured and digitized by other components of the PUF circuit. The digitized responses are then compared in a variety of combinations to produce a digital bit string.

The PUF response is analog in nature, e.g., it can be a voltage drop or the propagation delay of a signal through the PUF primitive. The analog nature of the underlying random variable make the PUF sensitive to environmental variations such as temperature and power supply noise. Several important applications of a PUF require that they produce the same bit string for a fixed challenge. Therefore, PUF architectures must be both random and resilient to noise sources.

In this paper, we investigate a PUF primitive that leverages resistance variations that occur in transmission gates (TGs). Hardware experiments are carried out on a set of chips at 9 temperature-voltage (**TV**) corners, using all combinations of the temperatures -40^{o}C, 25^{o}C and 85^{o}C and voltages 1.08 V, 1.2 V and 1.32 V. A novel embedded test structure called a **voltage-to-digital converter (VDC)** is also evaluated under these environmental conditions. The VDC is used to digitize the voltage drops produced by the TG PUF.

Beyond these novel aspects of this work, we also investigate several noise resilient bit-flip avoidance schemes, that are designed to increase the probability that the bitstring can be reproduced under varying environmental conditions. The first technique derives a voltage threshold from a chip's voltage drop distribution profile that is used to decide whether a given voltage comparison generates a **strong** bit or a **weak** bit. A second triple-module-redundancy (TMR-based) scheme is proposed for fixed length bitstrings that further improves bit-flip resilience. Although these techniques discard a significant fraction of bits, they provide several significant advantages. The public (helper) data associated with these methods reveals nothing about the secret bitstrings that they encode. Second, for applications where the PUF responses are made public, the difficulty of model building is significantly increased (assuming the public data is obfuscated) because bitstrings are constructed using only a subset of the possible voltage pairings. These techniques are investigated on data obtained from 40 copies of a test chip fabricated in a 90 nm technology (NOTE: results from 60 chips will be presented in final paper if accepted).

2.BACKGROUND

Random bit strings form the basis for encryption, identification, authentication and feature activation in hardware security. The introduction of the PUF as a mechanism to generate random bit strings began in [3], although their use for chip identifiers began a couple years earlier [2]. Since their introduction, there have been

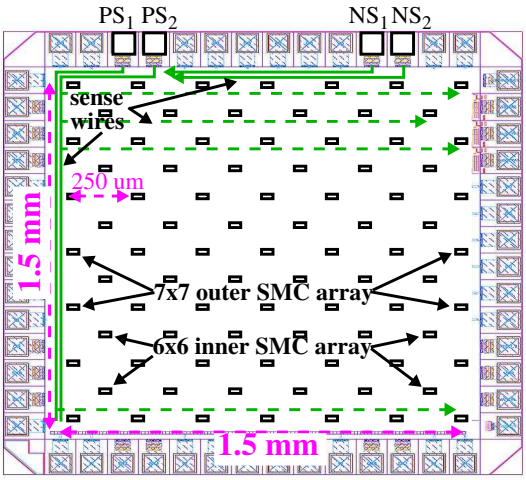

Fig. 1. Block diagram of 90 nm chips with 85 embedded stimulus-measure circuits (SMCs).

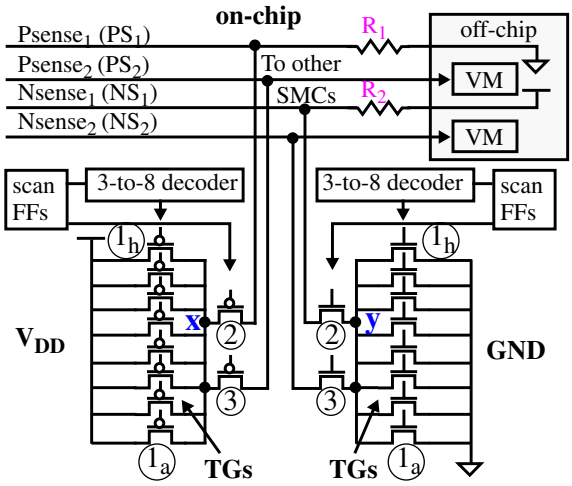

Fig. 2. SMC schematic in 90 nm chips.

many proposed architectures that are promising for PUF implementations, including those that leverage variations in transistor threshold voltages [2], in speckle patterns [3], in delay chains and ROs [4-7], in SRAMs [8], in metal resistance [9][10], in sensors [11], and many others. The TG PUF proposed in this research is also based on resistance variations as in [10]. However, this paper for the first time investigates the reproducibility of the bitstrings across 9 industrial range TV corners after digitization using an on-chip VDC.

3.EXPERIMENT SETUP

3.1 TG Array, TGVs and TGVDs

Fig. 1 gives a block diagram of the 90 nm test chip architecture. The chip padframe consists of 56 I/Os, and surrounds a chip area of approx. 1.5 mm x 1.5 mm. Four PADs labeled PS_1, PS_2, NS_1 and NS_2 refer to *voltage sense* connections, the 'P' version for sensing voltages near V_{DD} and the 'N' version for voltages near GND. These four terminals wire onto the chip and connect to 85 copies of a *Stimulus/Measure circuit* (SMC). The SMCs are distributed across the entire chip (see small rectangles) as two arrays, a 7x7 outer array and a 6x6 inner array. Although not shown, a controlling scan chain connects serially to each of the SMCs.

The schematic diagram of the SMC is shown in Fig. 2. A set of 20 'pseudo' pass gates (hereafter referred to as transmissions gates or **TGs**) serve as both the PUF primitives and voltage sensing elements. Eight of the TGs, labeled 1_a through 1_h, connect to the V_{DD} grid, as shown on the left side of Fig. 2, while the other eight connect to the GND grid. Two additional TGs, labeled as *2* and *3*, connect to the drains of the $1_{a\text{-}h}$ TGs. Separate scan FFs control their connection to the chip-wide wires that route to the P/NS_x pins shown in Fig. 1. The PS_1 and NS_1 sense wires are connected off-chip to GND and V_{DD}, resp., to create the stimulus condition described below. PS_2 and NS_2 are routed to off-chip Agilent 34401A voltmeters (VMs).

A voltage drop measurement is carried out by enabling three TGs, both of those labeled *2* and *3* and one from the group 1_a through 1_h. For example, using the PFET TGs, enabling TGs 1_a and *2* create a short between the V_{DD} grid on-chip and a GND node off-chip. The voltage drop falls across the two TGs as well as the PS_1 wire. The voltage on the node **x** between TG 1_a and *2* can be sensed with TG *3*[1]. The on-resistances of the TGs (and the resistance of the PS_1 wire) determine how much of the 1.2 V falls across each of TG 1_a and *2*. Random variations in the on-resistances of the TGs 1_a through 1_h produce different voltage drops as each is enabled. We refer to these voltages at node **x** as **TGVs**.

The component of the TGV that falls across the sense wires represents a bias because the length of the sense wires is different for each SMC in the array. The bias is eliminated by creating **TGV differences** (TGVDs) using the 8 TGVs measured within each SMC, separately for NFETs and PFETs. The TGVDs are obtained by subtracting the TGVs under all possible pairings, yielding 8*7/2 = 28 TGVDs. The total number of TGVDs obtained per chip is 2,380 for each of the PFETs and NFETs, obtained as 85 SMCs * 28 TGVDs/SMC.

The NFET and PFET TGVDs, in turn, can be compared under all combinations to produce bitstrings of length 2,380*2,379/2 * 2 = **5,662,020** bits. The NFET and PFET TGVDs cannot be compared with each other primarily because of channel width differences (PFETs are 2.5x wider than the NFETs) and mobility variations with doping (NFET variations are larger than PFET variations). As a consequence, PFET voltage variations are only about half as large as the NFET variations.

In our experiments, the order in which the comparisons are made is randomized using *srand(seed)* and *rand()* from the C programming library. This operation is easily implemented on chip using an LFSR and a seed.

3.2 Voltage-to-Digital Converter (VDC)

In addition to analyzing the TG voltage drops directly, we also analyze a digital representation of them that is produced by an on-chip VDC, similar to designs described in [12]. The architecture of the VDC is shown in Fig. 3. The VDC is designed to 'pulse shrink' a negative input pulse as it propagates down a current-starved inverter chain. As the pulse moves down the inverter chain, it activates a corresponding set of latches to record the passage of the pulse, where activation is defined as storing a '1'. A

1. Only a negligible amount of current flows through TG *3* to the voltmeter so the voltage on node **x** is nearly identical to that at the voltmeter.

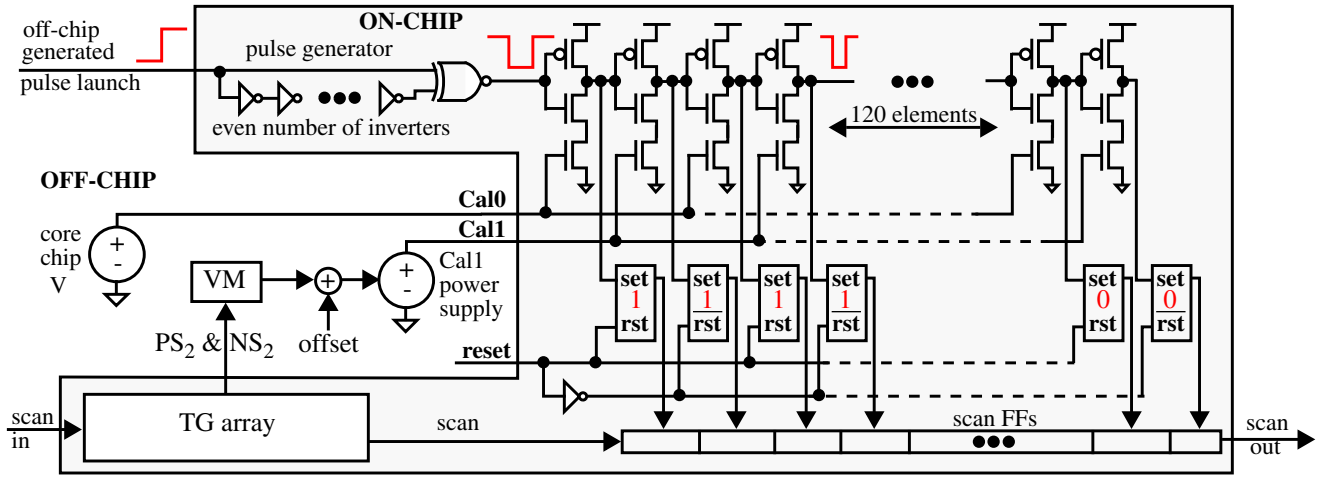

Fig. 3. Voltage-to-Digital Converter (VDC). On the left side is off-chip instrumentation that measures a voltage from the TG array, adds an offset and programs a power supply to drive the Cal1 input of the VDC on the right.

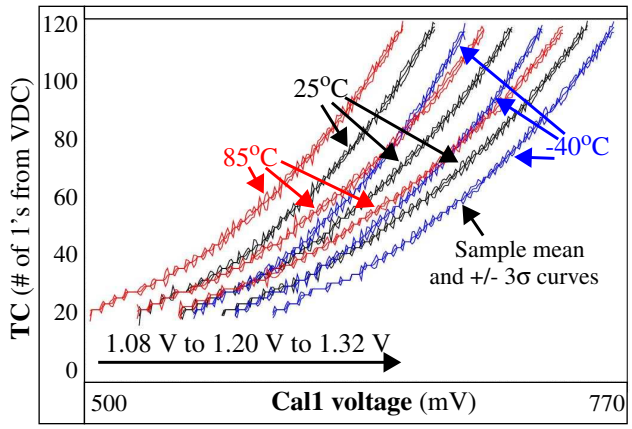

Fig. 4. VDC Cal1 vs. thermometer code (TC) curves across 9 temperature/voltage corners on one chip.

thermometer code, i.e., a sequence of '1's followed by a sequence of '0's, represents the digitized voltage.

The voltage-to-digital conversion is accomplished by introducing a fixed-width (constant) input pulse, which is generated by the pulse generator shown on the left side of the Fig. 3. Two analog voltages, labeled Cal0 (which is held constant) and Cal1 (the voltage to be digitized) connect to a set of NFET transistors in the inverter chain, with Cal0 connecting to NFETs in even numbered inverters and Cal1 to the NFETs in odd numbered inverters. The propagation speed of the two edges associated with the pulse are controlled separately by these voltages. The pulse will eventually die out at some point along the inverter chain when the trailing edge of the pulse 'catches up' to the leading edge. This is ensured by fixing Cal0 at a voltage higher than Cal1. A digital representation of the Cal1 voltage can then be obtained by counting the number of '1's in the latches.

In order to enable this type of pulse shrinking behavior, Cal1 needs to be set to a value between 500 mV and 800 mV. The voltage-divider (series) arrangement of the identically-sized TGs shown in Fig. 2 should provide voltages at the midpoint of the supply voltage, e.g., approx. 600 mV. This is not the case, however, because a significant portion of the voltage falls across the NS_1 and PS_1 sense wires, which is represented by the resistors labeled R_1 and R_2 in Fig. 2. As a consequence, the range of the TGVs observed in our experiments at node **x** (Fig. 2) for PFETs is between 950 mV to 1050 mV, and at node **y** for NFETs is 150 mV to 250 mV. In order to move Cal1 into the 600 mV range, an **offset** voltage is added (subtracted) to the voltages measured by the VM as shown in Fig. 3 for NFETs (PFETs). This offset voltage is computed using a calibration process described below.

The calibration process is needed because the required offset voltage changes as a function of temperature and voltage. The curves in Fig. 4 depict the behavior of the VDC over the 9 TV corners for one chip. The graph plots Cal1 on the x-axis against the number of '1' bits in the thermometer code, referred to as **TC**, on the y-axis. The mean and 3 σ curves are superimposed. The average 3 σ, computed using the individual 3 σ in each curve, is less than 1 for all curves. The small non-linearity in the curves does not degrade the statistical properties of the bitstrings, as shown below. The sensitivity of the VDC is approx. 1 TC bit per millivolt change in Cal1. The TGVs for a typical chip vary over the range of 40 to 60 mVs so no more than half of the 120 bit range of the VDC is used in our experiments.

Although the VDC remains stable across the TV corners, the shift of the curves along the x-axis causes overflow in the VDC; a situation where the pulse propagates through all 120 delay chain elements of the VDC. A calibration process is carried out that tunes the 'offset' at each TV corner, and effectively eliminates the adverse effects of the curve shift. The calibration process tests a distributed set of 9 TGs, e.g., of the 680 NFET TGs, and uses binary search to find an offset voltage that produces a 'target' TC, separately for each of the 9 tests. We set the target TCs for NFET and PFET TGVs to 65 and 85, resp. These targets worked well to prevent overflow in all of the 1,360 TG measurements, across the 9 TVs in all 40 chips in our population. The **median offset** from the 9 calibration tests is used as the offset during the subsequent data collection process. This calibration procedure only *approximates* the best offset, but does not need to be precise because the goal of the process is only to prevent overflow in the VDC. A more detailed explanation of the process is given in Section 6.1.

We plan to integrate the instrumentation used to measure the TGVs, to add an offset and to control the Cal1 voltage, as shown on the left side of Fig. 3, in the next version of the chip. The Cal1 offset voltages can be derived using a resistor-ladder network [13],

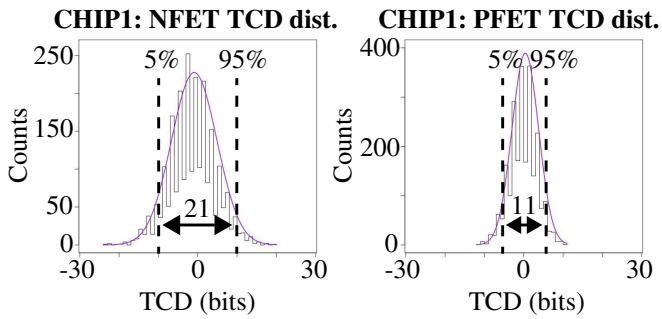

Fig. 5. Enrollment NFET (left) and PFET (right) TCD distributions with 2,380 components from one chip.

and added to the TG voltage using a voltage subtractor/adder circuit [14]. The offset only needs to be accurate to approx. 5 mVs, which significantly reduces the area overhead of the ladder network. With the availability of these on-chip components, a state machine can be easily designed to carry out the calibration process described above.

3.3 Data Collection Process

The calibration process described above is used to select an offset voltage, separately for the PFET and NFET elements on each of the 40 chips. Each of the 680 components are then enabled, one at a time, and the corresponding TGV is measured using the VM as shown in Fig. 3. The Cal1 power supply is programmed with this TGV plus the offset and 11 TC samples are collected from the VDC. This process is repeated for both the NFET and PFET components. The median value of the 11 samples is used to compute a 'difference' value, synonymous to the TGVDs described above. We use the term **TCD** to refer to these thermometer code differences in the remainder of this paper.

3.4 Overhead

Each SMC occupies an area of approx. 500 um^2, so the total area occupied by the array of 85 SMCs is approx. 42,500 um^2. If the SMCs are placed adjacent to each other (instead of being distributed as in Fig. 1), the array would occupy a 206 um x 206 um region. The VDC occupies an area of 136 um x 60 um. The area of the digital components, i.e., the LFSR and bit generation engine, is estimated at 300 um x 300 um. On-chip memory requirements for the array of 680 NFET and PFET TGs is approx. 2,380 bytes.

3.5 Voltage Thresholding Technique

As discussed above, TCDs are computed by subtracting TCs within the same SMC as a means of eliminating the voltage bias introduced by the sense wires. Computing differences also has the benefit of significantly increasing the number of bits that can be produced from each chip. For example 2,380 TCDs are produced from the 680 NFET TCs.

Using difference values, however, has two main drawbacks. First, subtracting two TCs reduces the signal-to-noise ratio because the noise from two separate measurements is combined in the difference. More importantly, TCDs 're-use' the base entropy of the array, which is defined by the 1,360 NFET and PFET TCs for each chip. Therefore, re-use makes model building attacks possible.

We propose a voltage thresholding technique as a means of dealing with model-building attacks and preventing information leakage in the public helper data. Our voltage thresholding technique discards voltage comparisons that are susceptible to producing bit flips in the bitstring. Bit flips occur when the relative ordering of a pair of TCDs defined during enrollment reverse order during regeneration. This is much more likely to occur for pairs of TCDs that are similar in magnitude. We show in our experimental results that it is possible to define a voltage threshold that filters all TCD pairings that introduce bit flips during regeneration at one or more of the TV corners. The threshold is derived using the distribution characteristics of TCs obtained during **enrollment**, which is carried out in our experiments at 25°C and 1.20 V[1].

Fig. 5 shows the TCD enrollment distributions for NFETs and PFETs from one of our chips. It is clear from the spread of the distributions that the NFET TCDs have more variation than the PFET TCDs as discussed in Section 3.1. The objective is to derive a threshold from these distributions that serves three primary goals: 1) avoids bit flips under different TV conditions in the subsequent bit generation phase, 2) preserves as many strong bits as possible for each chip and 3) makes the number of strong bits as consistent as possible across chips, i.e., scales with the range of variation that occurs on each chip. We define **strong bits** as those generated by TCD comparisons where the difference of the TCDs exceeds the threshold.

In our experiments, we found the limits defined by the two vertical lines labeled 5% and 95% in Fig. 5 achieve these goals. These limits capture the spread of the distribution while ignoring the outliers on the tails of the distributions, which, when included, introduce large variations in the number of strong bits preserved across the chip population, i.e., they degrade criteria 3 above. We then scale the range given by the distance between these limits (by a constant factor for all chips) to define the thresholds for the chip.

Figs. 6(a) and (b) provide an illustration of the thresholding process applied using TCD data from one of the chips. The graphs plot bit number along the x-axis against the **differences** of the TCDs being compared. Only the first 170 strong bits (of the 286,000) are shown. The horizontal lines at 15 and -15 delineate the voltage threshold boundaries for the NFET TCDs, which are derived from Fig. 5 using a scaling factor of 0.72.

Fig 6(a) shows those TCD differences which produce strong bits during enrollment. In addition to generating the secret bitstring, a **thresholding bitstring** is also constructed during enrollment which indicates which comparisons produce strong bits and which produce **weak bits**. The thresholding bitstring is recorded in public data storage, and using techniques such as run-length encoding, is proportional in size to the secret bitstring (see Section 6.3). This type of public data reveals nothing about the secret bitstring, and represents the public helper data for our PUF.

Fig. 6(b) superimposes the TCD difference data points generated under the remaining 8 TV corner experiments, which represent the regeneration components of our experiments. The public data is consulted to ensure regeneration uses the same comparisons as enrollment. The data points associated with the regenerations appear above and below the enrollment data points. Only those that move toward 0 line are problematic. Although none occur in these plots, points that move over the 0 line from above or below indicate the relative ordering has changed in the TCD pair-

1. It is important to derive the threshold voltage using data collected only during enrollment because it is not practical to vary environmental conditions in typical PUF usage scenarios.

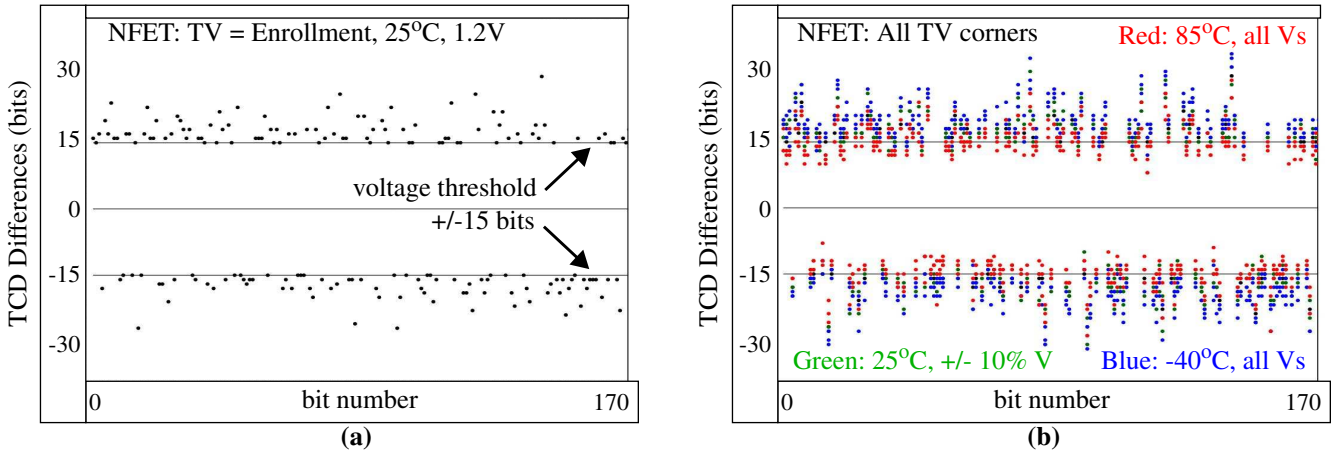

Fig. 6. Voltage threshold method showing the first 170 strong bit comparisons during (a) enrollment and (b) regeneration across 8 TV corners.

ing. A bit flip during regeneration will occur if this condition is met.

The TCD differences plotted in the figure span a larger range than the TCDs used to compute the limits from Fig. 5 because they represent **TCD differences**. Despite their larger range, only about 10% of the 2,831,010 possible comparisons, i.e., approx. 286,000 bits, survive the thresholding. A similar analysis using the TGVDs shows more than 30% surviving the thresholding, which suggests that the digitization process adds substantially to the noise. This is even more dramatic in the PFET analysis, where approx 20% of the TGVDs survive but only about 1% of the TCDs survive. The smaller variation in the PFET TCDs reduces the signal-to-noise for the VDC even further. However, the 320,000 bits for this chip that survive are reproducible across the TV corners and exhibit excellent statistical characteristics as we show below.

3.6 Fixed Length Bitstrings and TMR

In actual applications, only a fixed number of bits are needed. With encryption, the values vary between 128 to 4096 bits, depending on the encryption algorithm. The large number of bits available from the PUF can be beneficial, however, by allowing a set of fixed-length secret keys to be generated over time during successive enrollments.

A second possible usage scenario leverages this large pool of strong bits to further increase the resiliency to bit flip failures, i.e., beyond that provided by voltage thresholding. We propose a bitstring replication method that mimics a popular scheme used in fault tolerance called triple-module-redundancy or TMR. In this technique, a fixed length, e.g., 1,024-bit, bitstring is generated as described above. TMR is then applied to generate two more copies of the bitstring. The two copies are generated by parsing the strong bit sequence until a match is found to each bit in the first bitstring. During regeneration, a majority voting scheme is applied to each of the columns in the three identical regenerated bitstrings as a means of avoiding single bit flip failures. In other words, the final bitstring is constructed by using the majority of the 3 column bits as the final bit for each bit position, i.e., a '1' is assigned in the final bitstring when 2 or more of the 3 bits in the column are '1', and a '0' otherwise. An illustrative example is given in Section 6.2.

A PUF that is able to generate strong bit sequences that are locally random (a quality measured by the NIST tests [1] presented in the Section 4) ensures that a match occurs for each bit during the generation of the two copies every 2 bits on average. Under these conditions, it follows that a TMR-based bitstring, and its public data, consumes approx. 5x the number of strong bits than a non-TMR-based bitstring. The benefit, on the other hand, is a significant decrease in the 'probability of failure', i.e., the likelihood of a bit flip occurring during regeneration, as we show in Section 4. Moreover, this scheme offers flexibility by allowing the tolerance to bit flips and the size of the public data to be traded-off.

4.EXPERIMENTAL RESULTS

In this section, we evaluate the several important statistical properties of the TGV and TC-derived bitstrings including randomness, uniqueness and probability of bit flips, e.g., failures to regenerate the bitstring under different environmental conditions. As discussed in Section 3.2, the process of digitizing the voltages using the VDC adds noise and reduces the number of corresponding strong bits. The penalty of the digitization process is evaluated by carrying out the same analysis using the TGVs directly, and serves to illustrate the best that can be achieved in the absence of digitization noise.

Fig. 7a) gives the inter-chip hamming distance (HD) distribution using the TGVs while Fig. 7b) shows the distribution using TCs. The graphs plot HD along the x-axis against the number of instances on the y-axis[1]. With 40 chips, the total number of instances is 40*39/2 = 780. The distributions are 'fitted' with Gaussian curves to illustrate the level of conformity they exhibit to this distribution.

Since HDs must be computed across bitstrings of equal length, it was necessary to truncate the bitstrings used in Fig. 7 to the length obtained for the chip with the fewest number of strong bits. Truncation reduced the lengths to 1,408,294 for the TGV analysis and 265,302 for the TC analysis, which are approx. 25% and 4.7%, resp., of the maximum possible length, i.e., 5,662,020 bits. The chip with the longest bitstring, in comparison, uses 26.5% of the maximum for the TGV analysis and 6.6% for the TC analysis. The term **truncated bitstrings** is used to refer to the shorter, equal-length bitstrings.

The actual average inter-chip HDs listed in Fig. 7 are nearly equal to the ideal value of 50%. In contrast, the average inter-chip

1. HD is computed by counting the number of bits that are different in the bitstrings from two chips.

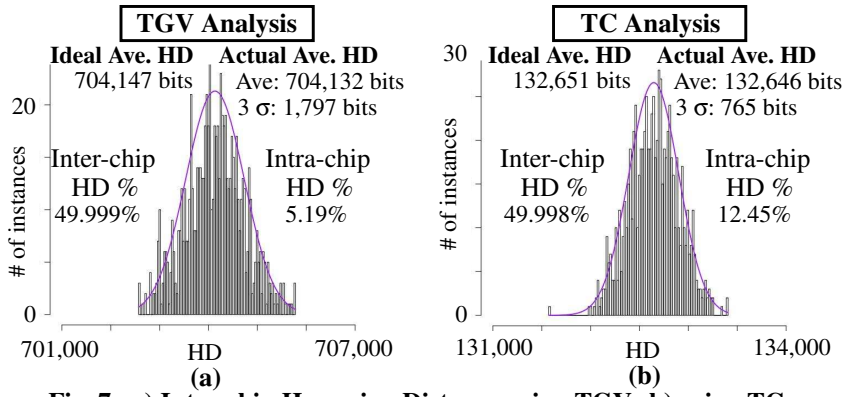

Fig. 7. a) Inter-chip Hamming Distance using TGVs b) using TCs.

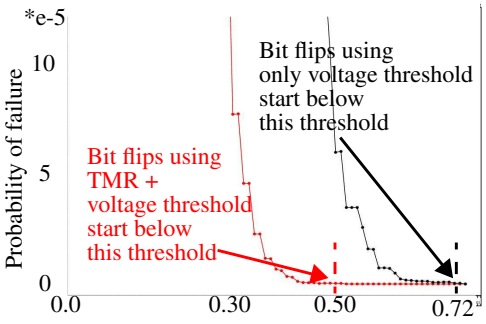

Fig. 8. NFET noise margin threshold vs. probability of failure (y-axis).

HDs for the bitstrings of length 5,662,020, i.e., those with the weak bits included (not shown), is 48.7% and 48.9% for TGV and TC, resp., so removing the weak bits improves the inter-chip HDs. The 3 σ values shown in the figure are derived from the Gaussian curves and represent the spread of the distributions (where smaller is better). These values are small relative to the length of the bitstrings, e.g., they are only 0.12% and 0.29% of the lengths for the TGV and TC analysis, resp.

The voltage thresholds are set to 0.43 (NFET) and 0.56 (PFET) for the TGV analysis and 0.72 (NFET) and 1.19 (PFET) for the TC analysis. These values were derived by analyzing the bitstrings across all 9 TV corners and tuning the values until no bit flips occurred (Section 6.2 discusses how this can be done in practice). Therefore, the intra-chip HD is technically 0 in both analyses. However, the underlying noise levels can be measured by disabling the thresholding technique. The values given in Fig 7 are the intra-chip HDs under these conditions. Intra-chip HD increases from 5.19% to 12.45% for the TC analysis, and reflects the noise added by the VDC digitization process.

We applied the NIST statistical tests [1] to the truncated bitstrings of the 40 chips at a significance level of 0.01 (the default). The TGV and TC bitstrings **pass all tests**, with no fewer than 37 passing chips per test (the number required by NIST for the test to be considered 'passed'). Moreover, all tests passed the **Pvalue-of-the-Pvalues** metric, even though the number of bitstrings used in the analysis is less than the number recommended by NIST.

Fixed-length bitstrings were also created using the TMR-based scheme proposed in Section 3.6. In our experiments, we were able to create, on average, 283 1024-bit TMR-based bitstrings per chip using TGV data, and 63 on average using TC data. Although not shown, the statistical test results are similar to those discussed above for the longer bitstrings.

As discussed in Section 3.6, the TMR scheme improves resiliency to bit flips over the voltage thresholding scheme alone. The curves shown in Fig. 8 illustrate the improvement. The voltage threshold used for NFETs (the PFET threshold is also changed proportionally) is plotted along the x-axis against the probability of failure on the y-axis. The probability of failure is computed at each threshold by dividing the number of bit flips that occur in all 40 chips by the total number of strong bits produced. The curve on the left is the result obtained using the TMR + voltage threshold technique, while the curve on the right uses only voltage thresholding. Both curves are exponential in shape (see Section 6.2 for curve fits and further analysis). However, from the positions of the curves, it is clear that the TMR scheme requires a lower threshold, 0.50 vs. 0.72, before any bit flips occur. Using 0.72 as the threshold, the probability of failure is 1.4e-6 with voltage thresholding but improves significantly to 3.3e-11 after adding TMR.

5.CONCLUSIONS

A transmission gate (TG) PUF and on-chip voltage-to-digital conversion circuit are evaluated on 40 copies of a 90 nm chip, at 9 temperature-voltage (TV) corners. Voltage thresholding and triple-module-redundancy techniques are proposed as a means of avoiding bit flips. Results from statistical tests confirm that cryptographic quality bitstrings are obtained using either the TG voltages or their digitized representations. The proposed bit flip avoidance schemes allow the user to trade-off the probability of failure with helper data overhead for applications requiring bitstring regeneration.

6.REFERENCES

[1] NIST: Computer Security Division, Statistical Tests, http://cs-rc.nist.gov/groups/ST/toolkit/rng/stats_tests.html

[2] K. Lofstrom, *et al.*, "IC Identification Circuits using Device Mismatch," *SSCC*, 2000, pp. 372-373.

[3] R. S. Pappu, *et al.*, "Physical One-Way Functions," *Science*, 297(6), 2002, pp. 2026-2030.

[4] B. Gassend, *et al.*, "Controlled Physical Random Functions," *Conference on Computer Security Applications*, 2002.

[5] M. Majzoobi, *et al.*, "Lightweight Secure PUFs", *ICCAD*, 2008.

[6] G. Qu and C. Yin, "Temperature-Aware Cooperative Ring Oscillator PUF", *HOST,* 2009, pp. 36-42.

[7] A. Maiti and P.Schaumont, "Improving the Quality of a Physical Unclonable Function using Configurable Ring Oscillators", *FPLA*, 2009. pp. 703-707.

[8] J. Guajardo, *et al.*, "Physical Unclonable Functions and Public Key Crypto for FPGA IP Protection," *FPLA*, 2007, 189-195.

[9] R. Helinski, *et al.*, "Physical Unclonable Function Defined Using Power Distribution System Equivalent Resistance Variations", *DAC*, 2009, pp. 676-681.

[10] J. Ju, R. Chakraborty, R. Rad, J. Plusquellic, "Bit String Analysis of Physical Unclonable Functions based on Resistance Variations in Metals and Transistors", *HOST*, 2012, pp. 13-20.

[11] K. Rosenfeld, *et al.*, "Sensor Physical Unclonable Functions", *HOST*, 2010, pp. 112-117.

[12] L. Guansheng, Y.M. Tousi, A. Hassibi and E. Afshari, "Delay-Line-Based Analog-to-Digital Converters,", Trans. on *CAS II*, Volume: 56 , Issue: 6, 2009, pp. 464-468.

[13] Dan O'Sullivan & Tom Igoe, "Physical Computing: Sensing and Controlling the Physical World with Computers," Thomson Course Technology Publishers, 2004, pp 388-391.

[14] R. Fried and C. C. Enz, "Simple and Accurate Voltage Adder/Subtractor," *Electronics Letters*, vol. 33, 1997, pp. 944-945.

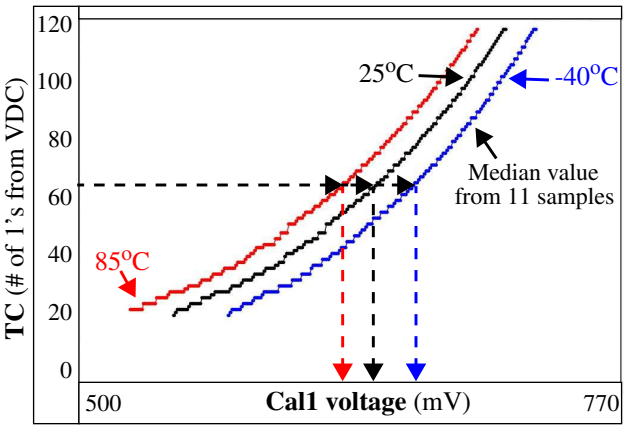

Fig. 9. VDC calibration curves from one chip at 85°C, 25°C and -40°C at 1.2 V illustrating offset calculation process.

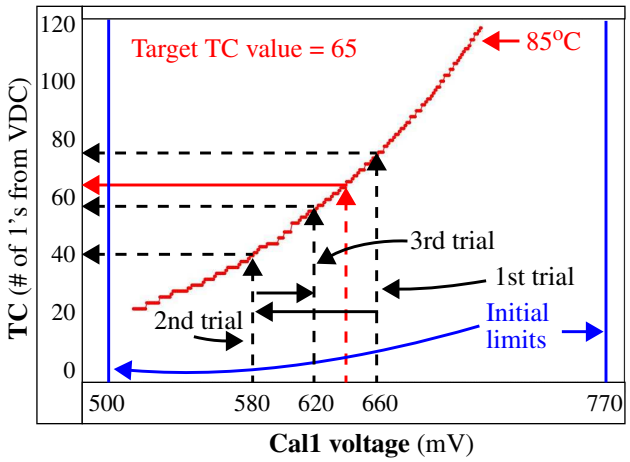

Fig. 10. Illustration of the binary search process used during calibration at 85°C.

SUPPLEMENTARY MATERAL

6.1 S1: VDC Calibration Process

The calibration process described in Section 3.2 is further illustrated using the Cal1 vs. TC curves shown in Fig. 9. The basic idea of calibration is to find an appropriate Cal1 voltage offset that prevents overflow in the VDC for any of the TGVs that will be measured subsequently. In order to minimize the time taken by the calibration process, only a subset of 9 TGs are used in the process. We found that the offset voltage obtained from this small subset of TGs serves as a suitable predictor for the entire population.

The overall objective is to select an offset voltage such that at any TV corner, the TG-under-test produces the same TC value. This objective is illustrated in Fig. 9 with the horizontal dashed line at TC = 65. The 3 curves shown represent the median values produced by the VDC on one of our chips as the Cal1 voltage is swept across a range of values (similar to the process described in Section 3.2 in reference to Fig. 4) at 3 different temperatures. The different positions of the dashed vertical lines from each curve make it clear that the offset voltage needs to change in order to maintain a value of 65 in the VDC. Note that the TGV itself measured from the TG-under-test will also change as a function of temperature. This situation is handled by using the TGVs directly in the calibration process (as opposed to using a special voltage source).

Calibration is carried out by enabling each of a select, distributed group of TGs, one at a time, and performing a binary search. The search process varies the Cal1 voltage offset until the TG-under-test produces a specific TC value. The process is illustrated in Fig. 10 using the 85°C Cal1-TC curve from Fig. 9. The initial limits are set to 500 mV and 770 mV. The 1st trial selects the midpoint between these limits, i.e., 660 mV. Note this midpoint voltage is the sum of the TGV and the offset voltage that is being tuned in the search. The 1st trial produces a TC of approx. 75, which is larger than the target. Therefore, the next trial uses 660 mV as the upper limit and the new midpoint voltage becomes 580. The 2nd trial produces a TC of 40, so 580 is used as the lower limit for the new midpoint. The process continues until an offset is found that produces a TC of 65. The binary search process is repeated using 9 TGs as a means of obtaining a value that best approximates the mean behavior. The median value from the 9 calibration tests is used as the final offset, which is added to all subsequent TGVs measured at this TV corner.

6.2 S2: Voltage Thresholding & TMR-based scheme

Voltage thresholding and TMR-based schemes are proposed in Sections 3.5 and 3.6. This section of the Supplementary Material is designed to clarify this process with an example. The voltage thresholding scheme shares characteristics with the shielding function proposed in [15] but is simpler because it is based entirely on strong bits, referred to as 'robust' bits in the reference. This fact changes the nature of the public data and eliminates information leakage that, although unlikely, is possible with shielding functions.

Fig. 11 illustrates the proposed voltage threshold and TMR-based scheme using data from a hypothetical chip. The x-axis plots a sequence of comparisons that would be used to generate a bitstring, while the y-axis plots the difference between the two TCDs that are being compared. The difference reflects the relative ordering of the two TCDs, e.g., positive difference values indicate that the first TCD is larger than the second. The voltage threshold values are labeled +VT and -VT. Strong bits require that the TCD difference data point lie above the +VT or below the -VT. This condition is recorded in the Public Data bitstring as a '1', as shown by the illustration below the data points. Weak bits, on the other hand fall within the limits and are indicated with a '0'. The bold (and blue) '0's indicate strong bits that are skipped under the TMR scheme described below.

As discussed in Section 3.6, the TMR-based method constructs 3 identical bitstrings during enrollment. The example shown in Fig. 11 shows fixed length bitstrings of length 4. The left-most bitstring labeled 'Secret BS' is generated from the first 4 strong bits encountered as the sequence of data points is parsed from left to right. The second bitstring labeled 'Redundant BS_1' is produced from the next sequence of data points but has the additional constraint that each of its bits must match the first bitstring. During its construction, it may happen in the continued left-to-right parsing of the data points that a strong bit is encountered that does not match the corresponding position in the 'Secret BS'. In the example, this occurs at the position indicated by the left-most bold '0' in the Public Data bitstring. Here, we encountered a strong bit with a value of '0'. But the 'Secret BS' requires the first bit to be a '1', so this strong bit cannot be used. The same process is used to gen-

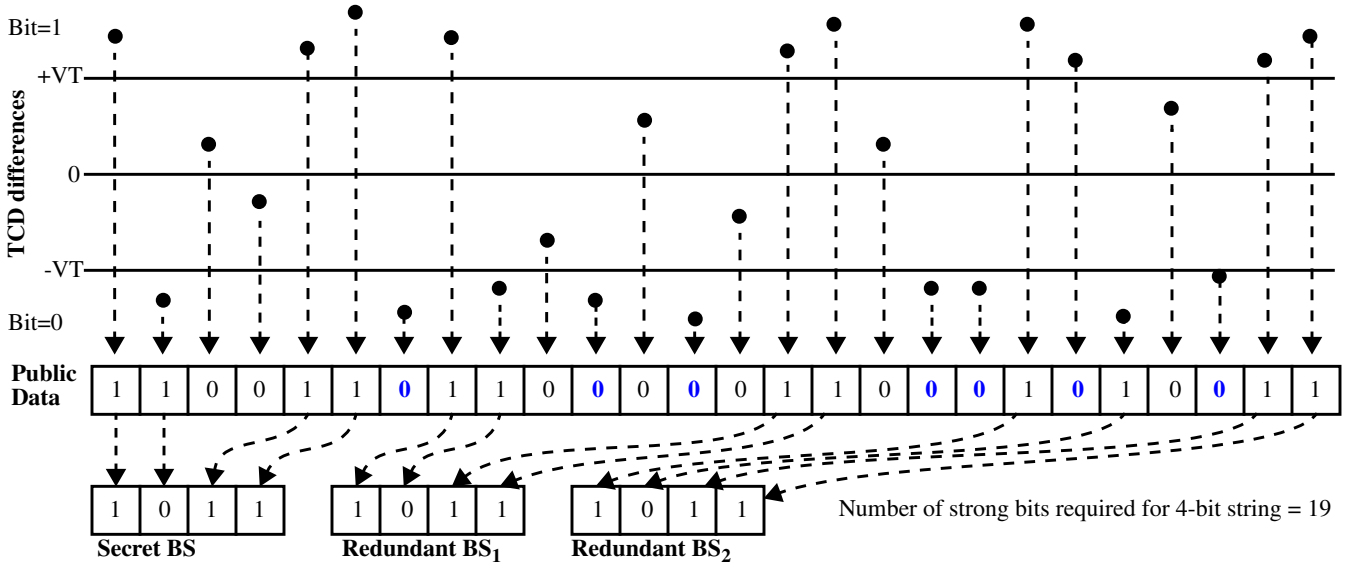

Fig. 11. Illustration showing the generation of a secret bitstring using the proposed voltage threshold and TMR-based method.

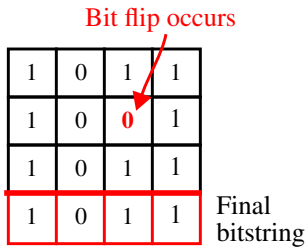

Fig. 12. Bit flip avoidance illustration using example from Fig. 11.

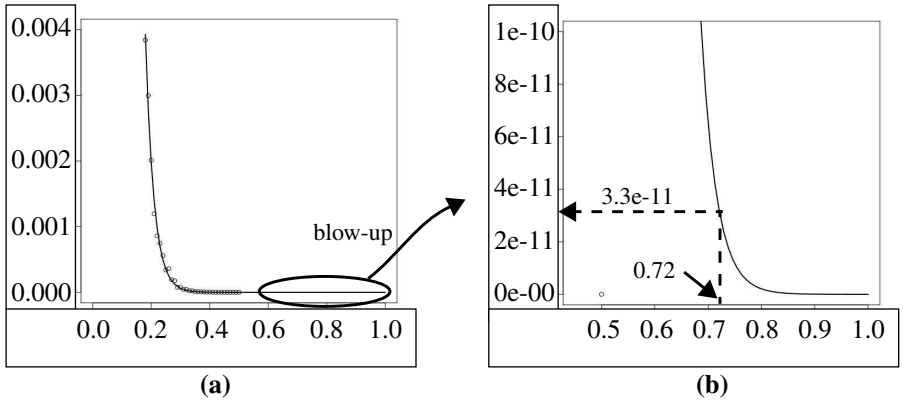

Fig. 13. a) TMR probability of error curve and b) blow-up of the designated region. The discrete curve is fitted with a superimposed exponential function.

erate the bitstring labeled 'Redundant BS_2'.

The number of strong bits required to generate a secret bit string of length 4 is approx 5x or 20. From the example, this is evaluated by counting the number of '1's and bolded '0's in the Public Data bitstring, which is given as 19. The benefit of creating these redundant bitstrings is improved tolerance against bit flips. For example. during regeneration, the three bitstrings are again produced, but this time using the Public Data to indicate which data points are to be used. Note that during regeneration, the positions of the data points shown in Fig. 11 may be displaced upward or downward because of temperature or voltage variations. Therefore, it may happen that data points that were outside the VT limits during enrollment are now inside the limits. Therefore, the Public Data must be used (instead of the VT limits) to determine which data points were used to generate the secret bitstring during enrollment.

In scenarios where the voltage threshold is set too low, it is possible that a strong data point used in enrollment is displaced across both the VT limit and the '0' line because of different temperature or voltage conditions in regeneration. This produces a bit flip, i.e., the bit generated during enrollment is opposite to the one produced during regeneration. However, with TMR, a bit flip can be avoided if no more than 1 bit flip occurs in a single column of the matrix of bits created from the 3 bitstrings. For example, the matrix of bits in Fig. 12 is constructed during regeneration in a similar way to those shown in Fig. 11 for enrollment. The first three rows are the Secret BS, Redundant BS_1 and Redundant BS_2 bitstrings as identified in Fig. 11, while the bottom row is constructed by using a **majority vote** scheme (in the spirit of TMR). The bit flip shown in the third column has no effect on the final bitstring because the other two bits in that column are '1', and under the rule of majority voting, the column bit is therefore defined as '1'.

In Section 4, the probability of failure using voltage thresholding alone and in combination with TMR was discussed, with the latter improving significantly on the former, from 1.4e-6 to 3.3e-11. These values were obtained by fitting the discrete-valued curves produced from repeatedly running the analysis at different

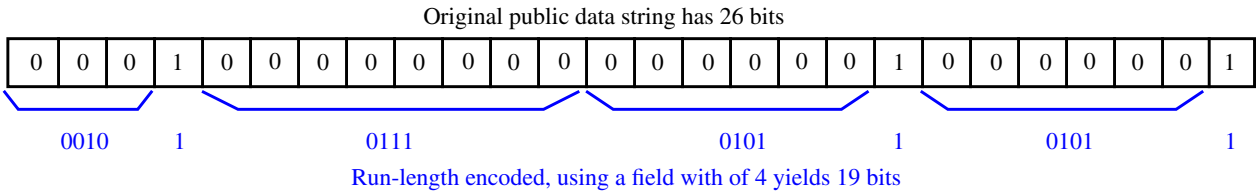

Fig. 14. Examples of run-length encoding as a compression technique to reduce public data size.

voltage thresholds with exponential functions. Fig. 13(a) shows the data for the TMR + voltage thresholding curve in Fig. 8 with the fitted exponential curve. The exponential is clearly a good fit to the data points from the discrete curve. Fig. 13(b) shows a blow-up of the region around the NFET threshold of 0.72 from which the estimate of 3.3e-11 was derived.

6.3 S3: Run-Length Encoding of Public Data

The size of the public (helper) data under the voltage thresholding and TMR-based schemes can be reduced using compression techniques such as run-length encoding. The benefit of run-length encoding is its simplicity. Fig. 14 shows an example public bit-string with 26 bits. The long strings of 0's can be run-length encoded by simply counting them and replacing the 0 sequence with a field which represents the number in each sequence. In the example, the run-length encoded public data uses 19 bits instead of 26. The longer the sequences of 0's, the more efficient the scheme becomes. The best choice for the field width depends on the nature of the public data, i.e., the average length of the 0 strings.

The public data for the TC analysis from Section 4 indicates that approx. 5% of the bits survive the thresholding, and even fewer, approx. 1%, are considered strong when TMR is added. The public data is therefore expected to contain strings of 0's with average lengths of approx. 100 under voltage thresholding + TMR. Therefore, a field width between 6 and 7 (which allows counting upto 64 and 128, resp.) should be optimal. We found that a field width of 6 is slightly better than 7, and provides a 69% reduction on average to the size of the original public data string. We plan to explore other compression techniques in future work.

6.4 S4: Supplementary Material References

[15] B. Skoric, P. Tuyls, W. Ophey, "Robust Key Extraction from Physical Uncloneable Functions", Chapter in Applied Cryptography and Network Security, 2005.