Exploring active manipulation attacks on the TERO random number generator

Yang Cao, Vladimir Rožić, Bohan Yang, Josep Balasch and Ingrid Verbauwhede
KU Leuven, ESAT/COSIC and iMinds
Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium
Email: {firstname.lastname}@esat.kuleuven.be

***Abstract*—True random number generators (TRNGs) are critical components in security systems used to generate session keys, challenges for authentication protocols and masks for secret sharing. Unfortunately, TRNGs are vulnerable to a wide class of physical attacks ranging from passive observation of generated numbers to active manipulation. In this work we investigate the susceptibility of the Transition Effect Ring Oscillator (TERO) TRNG to active manipulation attacks. In particular we perform underpower and low temperature attacks on an implementation of the TERO running on a Xilinx Spartan 6 FPGA and experimentally evaluate the effectiveness of four online tests as countermeasure.**

I. Introduction

True random number generators (TRNGs) are at the core of cryptographic applications. They are used for generating session keys, digital signatures, masks and challenges in authentication protocols. Cryptographic applications rely on the unpredictability of random numbers, which makes implementations of TRNGs crucial for security. All TRNGs have to produce output numbers that are uniformly distributed and unpredictable. On FPGA platforms, TRNGs have the additional design constraint that they have to be implemented using only digital logic. For this reason, TRNGs implemented on FPGAs are based either on timing jitter or metastability. A TRNG should be additionally resistant against physical attacks and faults induced by aging or by changing operating conditions. Several successful attacks [1], [2], [3] that manipulate the TRNG output have been presented in the literature in recent years. Robustness against such attacks can be achieved by implementing online tests [4], [5], [6] that monitor the behavior of the entropy source and trigger an alarm signal in case a malfunction is detected.

This paper focuses on the experimental evaluation of a TRNG based on the Transition Effect Ring Oscillator (TERO) originally proposed in [7]. This construction is based on a latch-like circuit that can be forced into a metastable state by an external signal. A random bit is extracted from the number of transitions that a circuit makes before resolving into one of the 2 stable states. An analysis of the robustness of this TRNG against voltage variations has been presented in [8].

In this paper, we make the following contributions:

1) We extend the evaluation presented in [8] using a wider range of power supply voltage variations.
2) We use an attack-oriented design methodology to develop statistical tests for raw random numbers.

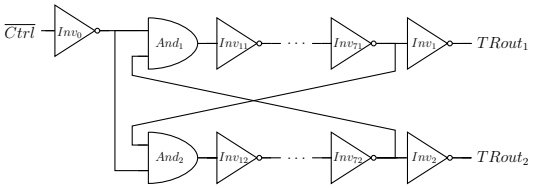


Fig. 1. Designed TERO structure with input and output buffers.

3) We experimentally verify the effectiveness of online tests.
4) We perform an analysis of a low temperature attack using freezing spray.

II. Implementation and Evaluation of TERO TRNG

Our TERO implementation uses two paths. For each path, the output of one control gate propagates through 7 inverters. The oscillation frequency is determined by the length of the inverter chain. As shown in Figure 1, the TERO structure has one input and two outputs. An inverter is used to buffer the input. The two outputs are connected to an inverter and a dummy inverter, respectively, in order to balance the load capacitance. Because our target platform is an FPGA, all gates are implemented using lookup tables.

We implemented the TERO loop on a Xilinx Spartan 6 FPGA from a Digilent Atlys board. In order to have good design symmetry and make sure that the synthesis tool does not optimize out middle inverters, we utilized the LUT-6 primitive from Xilinx to represent each logic element in the TERO. In total, 7 slices are used for one TERO. In addition, connections between all look-up tables of the TERO are manually routed with Xilinx FPGA editor to improve its symmetry.

The oscillation frequency of the TERO is higher than the system clock. Therefore, an asynchronous counter is used to count how many rising edges appear for every oscillation of the TERO. The counted value is defined as the *oscillation occurrence*. The least significant bit (LSB) of the count is used as the random bit.

The random bit generated after the oscillation stop is detected using an edge detector. After enabling the TERO loop, if there is no transition in 5 consecutive system clock cycles, we consider that the TERO has stopped. The random bit is then extracted and the TERO is reset.

We examined the oscillation frequency results of our TERO TRNG implementation with different placement constraints.

This paper has been accepted at IEEE 59th International Midwest Symposium on Circuits and Systems (http://events.kustar.ac.ae/mwscas2016/). The final publication is available at the proceedings of the IEEE Midwest Symposium on Circuits and Systems (ISSN: 1548-3746).

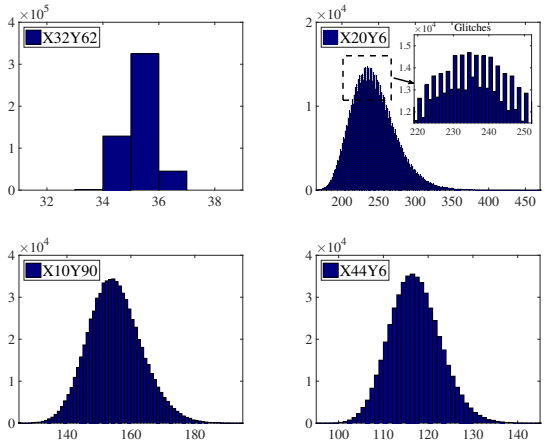


Fig. 2. Oscillation frequency results of different locations on FPGA.

TABLE I. RESULTS OF STATISTICAL TESTS.

Test	X10Y90	X44Y6
Relative frequency	0.500	0.499
Average autocorrelation coefficient (0 shift)	0.499	0.500
Minimum entropy (4-bit variable)	3.998	3.997
NIST SP800-22 tests	Success	Success
Average throughput	1.29Mbps	1.25Mbps

The location of Inv_0 is used as the reference. Results from different placement locations are presented in Figure 2. Not all of them are suitable to be used as TRNG. For example, X32Y62 represents a set of positions at which the implemented TERO has less than 40 transitions. Position X32Y62 is the extreme case with only 4 possible transition counter values. For positions such as X20Y6, the number of transitions is larger. However, glitches of the histogram indicate the existence of statistical defects. X10Y90 and X44Y6 are good positions, because they have more than 100 transitions and the histogram envelope of the transitions is close to a normal distribution. In what follows, we focus on implementations the TERO TRNG at positions with this type of distribution.

We evaluated our implemented TERO TRNG with the statistical test suite from NIST SP800-22 [9]. We also calculated the relative frequency, average autocorrelation efficient (0 shift) and minimum entropy of 4-bit segments of random sequence. As shown in the Table I, the implemented TERO TRNG has good statistical properties. Under a system clock of 100MHz, the TERO implementations at X10Y90 and at X44Y6 require on average 77 and 80 cycles, respectively, to generate one random bit. In other words, they achieve a throughput of 1.29Mbps and 1.25Mbps, respectively.

III. ACTIVE MANIPULATION ATTACKS

In this section we examine the robustness of our TERO TRNG implementations against temperature and underpower attacks. For each attack an *isolation test* is performed before testing the TRNG. This is a test suite which contains Block RAMs, shift registers, UART communication module and asynchronous counters. It is used to distinguish the attack impact on the TERO from the failure of other components on FPGA.

TABLE II. EXPERIMENTAL RESULTS UNDER FREEZING COMPARED TO NORMAL SITUATION.

	X10Y90		X44Y6	
Test	normal	freezing	normal	freezing
Average oscillation occurrence	155	159	116	111
Average random number	0.500	0.501	0.499	0.501
Average auto-correlation coefficient	0.499	0.500	0.500	0.500
Minimum entropy	3.997	3.995	3.997	3.979

TABLE III. ISOLATION TEST RESULT OF UNDERPOWERING.

System clock	Block RAM	Shift register	Counter	UART
100MHz	0.86V	0.86V	0.90V	0.86V
50MHz	0.70V	0.70V	0.73V	0.70V
20MHz	0.67V	0.67V	0.67V	0.67V
10MHz	0.67V	0.67V	0.67V	0.67V

A. Temperature attack

We performed low temperature attacks by using a freezing spray. Our isolation test module writes values 0000-FFFF to a 16-bit width Block RAM. Data is subsequently shifted into a 16-bit shift register, counted using an asynchronous counter, and finally sent to the PC using UART. We have verified that no error occurs during the low temperature attack, thus concluding that freezing does not influence the data collecting setup.

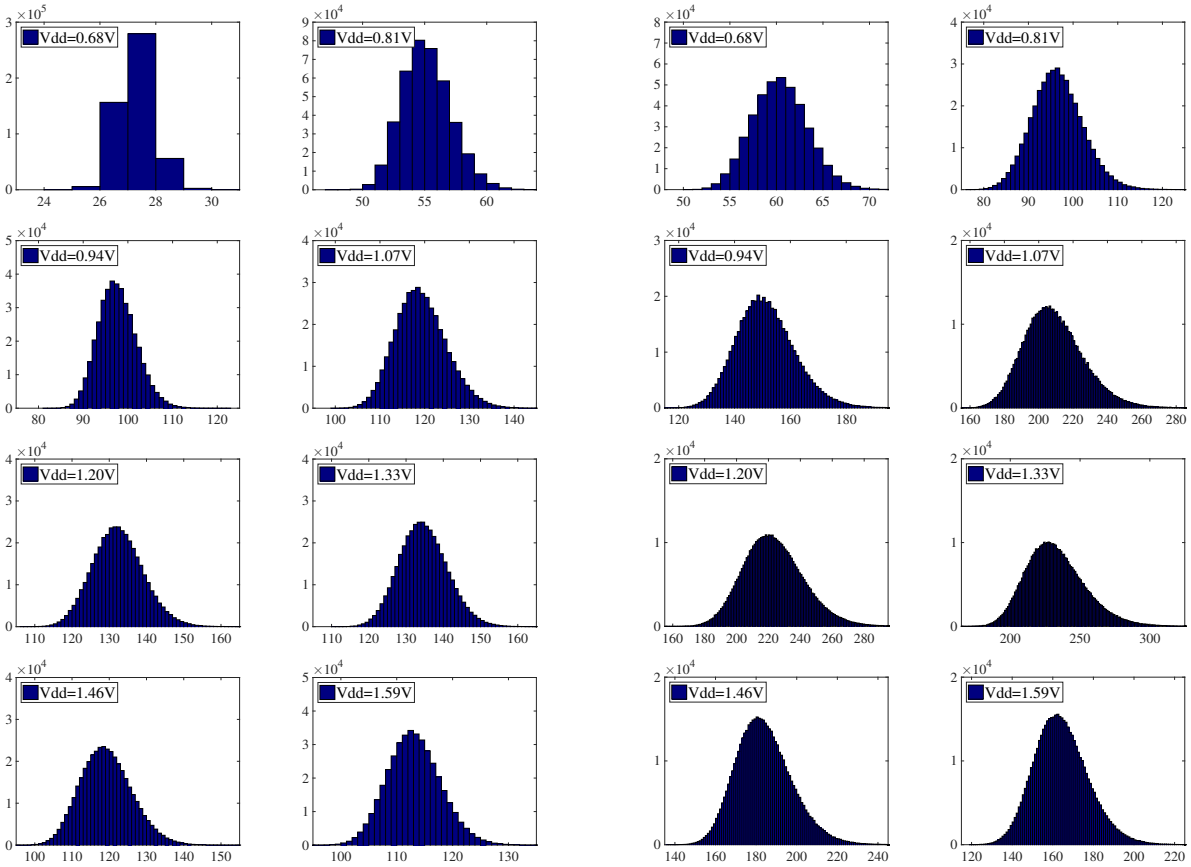
We then applied freezing on our TERO implementation. Due to limited time exposure of the FPGA to the freezing spray, it was not possible to obtain large amount of random bits *under* attack. For this reason, NIST tests could not be applied to random sequences under freezing. As shown on Table II, the observed difference between normal operation and low temperature attack was quite small. Based on this, we concluded that our use of freezing spray has only a limited influence on TERO.

B. Underpower attack

In the underpower attack experiment, we used a modified Atlys Spartan 6 FPGA board in which the voltage regulator of a FPGA is bypassed to directly connect to a power supply. We verified that location X10Y90 on this new board is still suitable for the TERO. However, location X44Y6 did not produce enough oscillations. In order to overcome this, we used a new location X44Y108 with characteristics similar to X10Y90.

The isolation test of the underpower attack was performed by recording the lowest allowed voltage for each module. Note that the lower voltage slows down the circuit, so the *critical voltage* depends on the system clock frequency. We performed the test using different system clock frequencies. The critical voltage decreases when reducing the system clock, but remains the same below 20MHz. We therefore selected 20MHz in this design to assess the influence of underpowering on the TERO. Table III lists the isolation test result of underpowering.

We first checked the lowest allowed V_{dd} of each TERO implementation. The TRNG stopped working at 0.67V, thus we selected 0.68V as the minimum V_{dd}. We also considered higher V_{dd}, which may influence the TRNG. We finally determined a range of (0.68V,1.59V) with a step size of 0.13V for these experiments.



(a) X10Y90 (b) X44Y108

Fig. 3. Oscillation test result at different V_{dd}.

Figure 3 shows the oscillation test results for different V_{dd} values. All obtained distributions are similar to the normal distribution. However, the number of oscillations is reduced when V_{dd} decreases from 1.20V to 0.68 and when V_{dd} increases above 1.33V. In particular for X10Y90 with 0.68V V_{dd}, the number of oscillations is reduced to below 30.

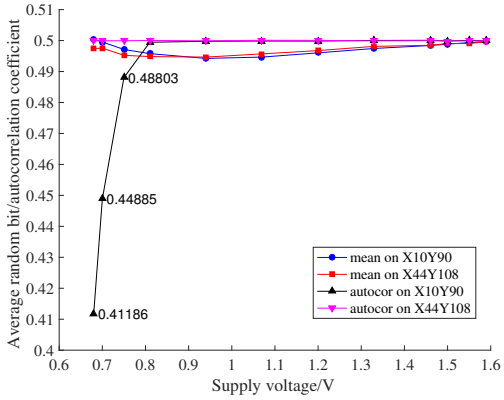
Figure 4 illustrates the results of different statistical tests. To improve accuracy, we added several test points close to interesting boundaries, e.g. 0.70V, 0.75V, 1.50V and 1.55V. On X44Y108, both average random bit and correlation coefficient are near to 0.5. Changing V_{dd} seems to have little influence on the TRNG. On X10Y90, the average random bit is also near 0.5. But the average correlation coefficient falls quickly when V_{dd} is below 0.8V. This means that the randomness of the TRNG is affected by a low V_{dd}. Note however that a higher V_{dd} has no impact on either location. We additionally calculated both Shannon entropy and min-entropy. The results of this test show that both entropies on the the TERO on X44Y108 are always higher than 3.9. But on X10Y90, both Shannon and min-entropy decrease quickly when V_{dd} is below 0.8V. The min-entropy even decreases from over 3.95 at 1.20V to 3.08 at 0.68V. Considering the oscillation test results, we believe that such decrease of randomness is caused by the decrease of number of oscillations. X44Y108 has a number of oscillations around 220 at 1.20V and higher than 50 at 0.68V. Thus randomness is only slightly influenced. X10Y90 has only around 130 oscillations at 1.20V. Such number is reduced below 30 at 0.68V, which most likely causes the randomness drop.

IV. On-the-fly Test

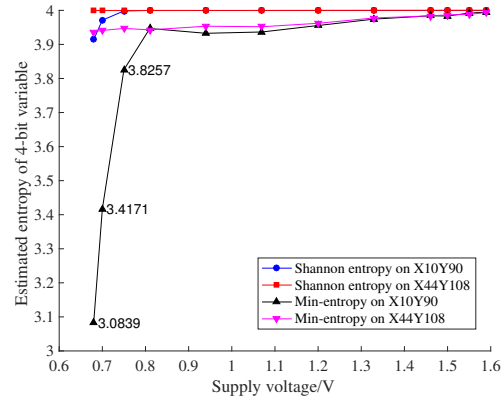
In order to detect threats to TRNG, we propose an online test module shown in Figure 5. This module performs some basic statistical tests: average random bit, autocorrelation coefficient, minimum entropy and oscillation check.

The test of average random bit checks the number of ones in a sequence. Similarly, the autocorrelation test checks the number of ones in a sequence generated by xoring adjacent bits of the input sequence. Accumulated results are compared with an upper bound and a lower bound. Such bounds are set to $(\mu-3\sigma, \mu+3\sigma)$ of the normal distribution estimated from a random sequence observed at the normal operating condition. If the result is out of range, the test module triggers the alarm.

The entropy test is the estimation of min-entropy. The test counts frequencies of all possible 4-bit segments in the test sequence. The min-entropy is estimated from the maximal detected frequency. We set the minimum allowed entropy to 3.88. The corresponding maximal frequency is pre-computed and an alarm is triggered if the detected frequency is above this bound.



(a) Average random bit and auto-correlation coefficient.



(b) Entropy.

Fig. 4. Statistical characteristics curves.

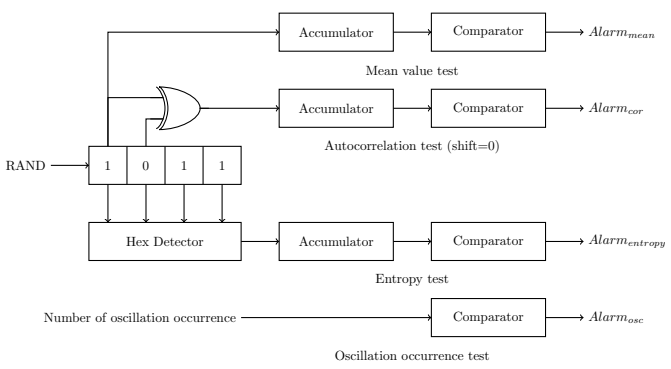


Fig. 5. Online test module

The oscillation test is simple. We directly retrieve the number of oscillation occurrence from digitalization module and compare it with a lower bound to detect the threat. The critical bound is determined as 75 (50 with 50% margin). Unlike other tests, this test is performed for every generated bit while other tests require a sequence of bits.

Table IV summarizes the effectiveness of each test for different sequence lengths.

TABLE IV. ONLINE TEST RESULT (%).

V_{dd}	Test	Sequence size					
		512	1024	2048	4096	8192	16384
1.20V	Mean	0.26	0.19	0.45	0.80	1.49	1.32
	Correlation	0.26	0.21	0.29	0.39	0.31	0.21
	Entropy	100	100	98.8	95.8	75.6	29.4
	Oscillation	0.00	0.00	0.00	0.00	0.00	0.00
0.75V	Mean	6.7	11.7	5.01	5.91	15.6	41.2
	Correlation	11.4	35.4	22.9	56.6	94.7	100
	Entropy	100	100	100	100	100	100
	Oscillation	100	100	100	100	100	100
0.70V	Mean	32.4	65.4	54.9	75.6	74.9	81.3
	Correlation	53.6	99.7	99.5	100	100	100
	Entropy	100	100	100	100	100	100
	Oscillation	100	100	100	100	100	100

V. CONCLUSIONS

In this work, we evaluated the TERO TRNG robustness against different placement locations on an FPGA and against active attacks. In addition, we implemented 4 online tests for attack detection. It was found that this TRNG is sensitive to placement locations, robust against low temperature attacks using freezing spray, and vulnerable to underpower attack. The oscillation test appears as the most useful of the 4 implemented tests for attack detection.

ACKNOWLEDGMENT

This work is supported in part by the Flemish Government through FWO G.0130.13N and FWO G.0876.14N, the KU Leuven through the C16/15/058 project, the Hercules Foundation AKUL/11/19, and through the Horizon 2020 research and innovation programme under grant agreement No 644052 HECTOR.

REFERENCES

[1] A. T. Markettos and S. W. Moore, "The frequency injection attack on ring-oscillator-based true random number generators," in *CHES*, 2009, pp. 317–331.

[2] P. Bayon, L. Bossuet, A. Aubert, V. Fischer, F. Poucheret, B. Robisson, and P. Maurine, "Contactless electromagnetic active attack on ring oscillator based true random number generator," in *COSADE*, 2012, vol. 7275, pp. 151–166.

[3] H. Martín, T. Korak, E. S. Millán, and M. Hutter, "Fault attacks on strngs: Impact of glitches, temperature, and underpowering on randomness," *IEEE TIFS*, vol. 10, no. 2, pp. 266–277, 2015.

[4] B. Yang, V. Rožić, N. Mentens, W. Dehaene, and I. Verbauwhede, "Embedded HW/SW platform for on-the-fly testing of true random number generators," in *DATE*, 2015, pp. 345–350.

[5] B. Yang, V. Rožić, N. Mentens, and I. Verbauwhede, "On-the-fly tests for non-ideal true random number generators," in *ISCAS*, 2015, pp. 2017–2020.

[6] B. Yang, V. Rožić, N. Mentens, W. Dehaene, and I. Verbauwhede, "TOTAL: TRNG on-the-fly testing for attack detection using lightweight hardware," in *DATE*, 2016, pp. 127–132.

[7] M. Varchola and M. Drutarovský, "New high entropy element for FPGA based true random number generators," in *CHES 2010*, pp. 351–365.

[8] M. Varchola, "New FPGA based TRNG principle using transition effect with built-in malfunction detection," in *CryptArchi*, 2009, pp. 150–155.

[9] A. Rukhin et al., "A statistical test suite for random and pseudorandom number generators for cryptographic applications." Special-Pub:800-22 NIST, August 2008.