A Closer Look at the Delay-Chain based TRNG

Miloš Grujić, Vladimir Rožić, Bohan Yang and Ingrid Verbauwhede
imec-COSIC, KU Leuven
Kasteelpark Arenberg 10, B-3001 Leuven-Heverlee, Belgium
Email:{milos.grujic, vladimir.rozic, bohan.yang, ingrid.verbauwhede}@esat.kuleuven.be

***Abstract*—This paper presents a refined stochastic model of the delay-chain based true random number generator (DC-TRNG) and its application. DC-TRNG is a true random number generator for FPGAs that utilizes time-to-digital conversion (TDC) to accurately determine the position of the ring-oscillator jittery signal edge. Our stochastic model employs precise time characterization of the carry-chains that are used for TDC in the DC-TRNG. In order to determine lower bounds of the estimated min-entropy, the binary probabilities are calculated by applying the stochastic model. Based on these computed probabilities, we perform optimizations of the DC-TRNG parameters on two different FPGAs – Xilinx Spartan 6 and Intel Cyclone IV, in order to achieve the highest possible throughput of the DC-TRNG.**

I. Introduction

Random number generation is one of the crucial issues in modern-day hardware security. In particular, random number generators (RNGs) are used for generating secret keys and challenges for authentication protocols. Most cryptographic primitives rely on the uniformity and unpredictability of the random numbers for their security.

True random number generators (TRNGs) are a class of RNGs that inherit randomness from physical, non-deterministic phenomena such as thermal noise or metastability. The state-of-the-art approach for design and evaluation of TRNGs requires not only that a TRNG passes a variety of statistical tests (e.g. NIST 800-22 [1]) but also to have an accompanying stochastic model that is used for estimating the unpredictability of the output [2]. Min-entropy is used as the relevant metric for quantifying this unpredictability. As part of the design procedure, a TRNG developer is required to provide a lower bound on the min-entropy. Any simplification used in the stochastic model should be analyzed with respect to the worst-case impact on the security.

With the advent of reconfigurable computing in the last decades, the design of fully-digital TRNGs for FPGA platforms has emerged as a new challenging problem. In this paper, we look into the delay-chain based TRNG (DC-TRNG) [3] which relies on the timing jitter of a ring-oscillator for generating randomness. The accumulated timing jitter is sampled by a high-precision time-to-digital converter (TDC) implemented using CARRY4 primitives on Xilinx FPGAs. We focus on the most compact version of the DC-TRNG consisting of a single-LUT ring-oscillator followed by a single-line TDC. Figure 1 shows the generic DC-TRNG architecture that we used for implementations on a Xilinx Spartan 6 [4] and Intel Cyclone IV [5] FPGAs.

The contributions of this paper are as follows:

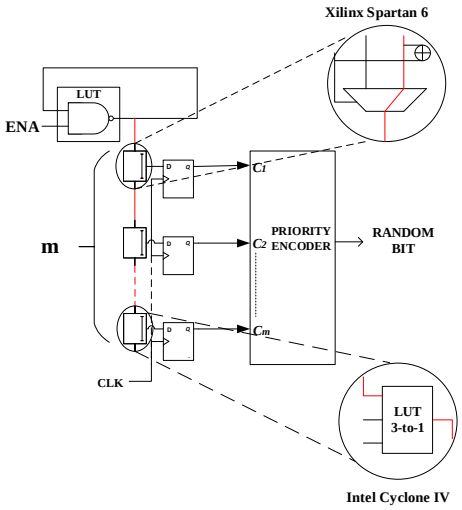


Fig. 1. Architecture of DC-TRNG.

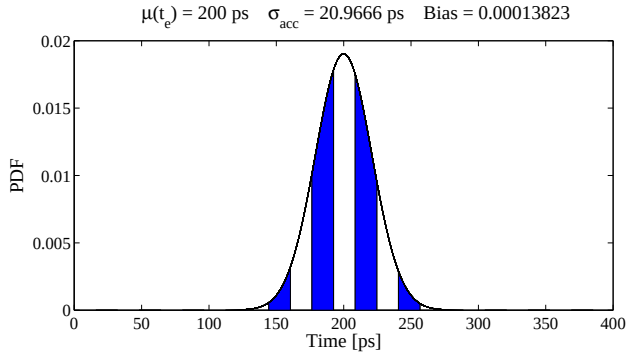
- A refined stochastic model of the DC-TRNG that accounts for the non-linearity of the time-to-digital conversion.
- Application of the new model to estimate the min-entropy of the DC-TRNG implementations on Xilinx Spartan 6 and Intel Cyclone IV FPGAs.
- A throughput optimization procedure to guide the choice of the design parameters.
- A comparison between the two models in terms of security and maximal obtainable throughput.

II. The Stochastic Model

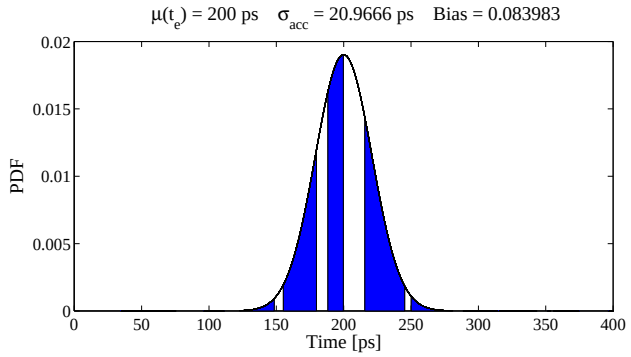
DC TRNG harvests randomness from the time uncertainty of the signal edge position of a ring-oscillator. After enabling oscillations the timing jitter starts to accumulate. The ring-oscillator output is connected to a delay-chain, which, together with the corresponding flip-flops, forms a time-to-digital converter. After a fixed number of system clock cycles N_A, the noisy ring-oscillator output is sampled by the TDC to precisely determine the position of the signal edge. This position is encoded by a priority encoder, and the LSB of the output is used as a raw random bit of the DC-TRNG.

We use the following notation for the cumulative distribution function of the normal distribution with the mean μ and the standard deviation σ:

$$F_{\mu,\sigma}(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{x} e^{-\frac{(t-\mu)^2}{2\sigma^2}} dt\,. \qquad (1)$$



(a) Sampled Gaussian distribution – stochastic model in [3].



(b) Sampled Gaussian distribution – stochastic model in this paper.

Fig. 2. Sampling process difference between the old and the new stochastic model.

The original stochastic model of the DC-TRNG [3] uses the assumption that all time bins of the delay-chain have equal widths and binary probabilities are estimated using the following method. The accumulated timing jitter is modeled by a Gaussian distribution with the mean μ_{acc} and the standard deviation σ_{acc}. The probability of sampling a bit value "1" is equal to the probability that the jittery signal edge is captured in odd time bins. This probability is given by:

$$P_1(\mu_{acc}, \sigma_{acc}) = \sum_{i=-\infty}^{\infty} \left(F_{\mu_{acc},\sigma_{acc}}((2i+1)d) - F_{\mu_{acc},\sigma_{acc}}(2id) \right), \quad (2)$$

where d denotes the width of the delay-chain time bins.Finally, the min-entropy can be calculated as

$$H_\infty = -log_2(max(P_1(\mu_{acc}, \sigma_{acc}), 1-P_1(\mu_{acc}, \sigma_{acc}))) . \quad (3)$$

To estimate the min-entropy using Equations (2) and (3), the designer has to know the values of σ_{acc} and μ_{acc}. The value of σ_{acc} depends on the jitter accumulation time $N_A \cdot T_{CLK}$. According to the central limit theorem, the variance σ^2_{acc} accumulates linearly over time. The accumulation rate has to be measured prior to design, for example by using one of the procedures proposed in [6], [7], [8]. On the other hand, the distribution offset μ_{acc} cannot be reliably determined at design time because it is affected by the unpredictable processes such as the power supply noise and the low frequency noise. In order to provide a safe lower bound, the min-entropy is computed by sweeping the parameter μ_{acc} across its domain – from the beginning until the end of a time bin. The entropy claim has to be given for the worst case value. In this model, the lowest min-entropy is achieved for $\mu_{acc} = d/2$.

One important limitation of the stochastic model that uses the assumption of equal widths of all time bins in the delay-chain is illustrated in Figure 2. Figure 2a shows sampling normal distribution using a linear TDC. The blue areas under the normal distribution represent the probability of sampling bit value "1" and their positions and widths correspond to the positions and widths of the time bins which are encoded as "1" by the priority encoder. For the chosen parameters, the estimated output bias is very low, around 0.01 %. However, in practice a TDC is not linear, and a more realistic result is shown in Figure 2b. Sources of the non-linearities of the TDC on the FPGA are the time skew of the clock signal and uneven propagation delays in the FPGA structure, that occur due to technology process variations and the layout of FPGA primitives. Sampling the same distribution using a non-linear TDC results in a bias of more than 8 %. Therefore, the assumption of linear TDC leads to an overestimation of generated randomness and may potentially result in an unsecured TRNG design. For this reason, we propose a refinement of the stochastic model that accounts for the non-linearity of the TDC. The new estimator of probability is given by:

$$P_1(\mu_{acc}, \sigma_{acc}) = \sum_{i=-\infty}^{+\infty} \sum_{j=1}^{N/2} \left[F_{\mu_{acc},\sigma_{acc}} \left(\sum_{k=1}^{2j} d_k - i \cdot T_0 \right) - F_{\mu_{acc},\sigma_{acc}} \left(\sum_{k=1}^{2j-1} d_k - i \cdot T_0 \right) \right], \quad (4)$$

where d_k are the widths of the TDC bins and $T_0 = \sum_{l=1}^{N} d_l$ is the period of the ring oscillator. Steps d_k have to be measured on the implementation platform. Min-entropy has to be evaluated for μ_{acc} within the range $[0, T_0)$ and the lowest value is used as the conservative entropy claim.

III. Application of the Model

The refined stochastic model presented in Section II is employed to determine the lower bounds of the estimated min-entropy after calculating relevant platform parameters – period of the ring-oscillator T_0, the rate of the jitter accumulation σ^2_A/t_A and widths of the individual time bins d_k in the delay-chain. In Figure 3, blue rectangles depict experimentally obtained widths of the time bins in the delay-chain which are encoded as "1" using a priority encoder. The widths of the time bins are measured by applying the code density test [8]. This is performed by sampling the ring-oscillator signal in the delay-chain every clock cycle, recording the state of the delay-chain after each sample, and then calculating the number of detected ring-oscillator signal edges in each time bin. The proportion of detected edges in each time bin corresponds to its width because there is equal probability of the ring-oscillator signal edge occurring at any point during the clock period.

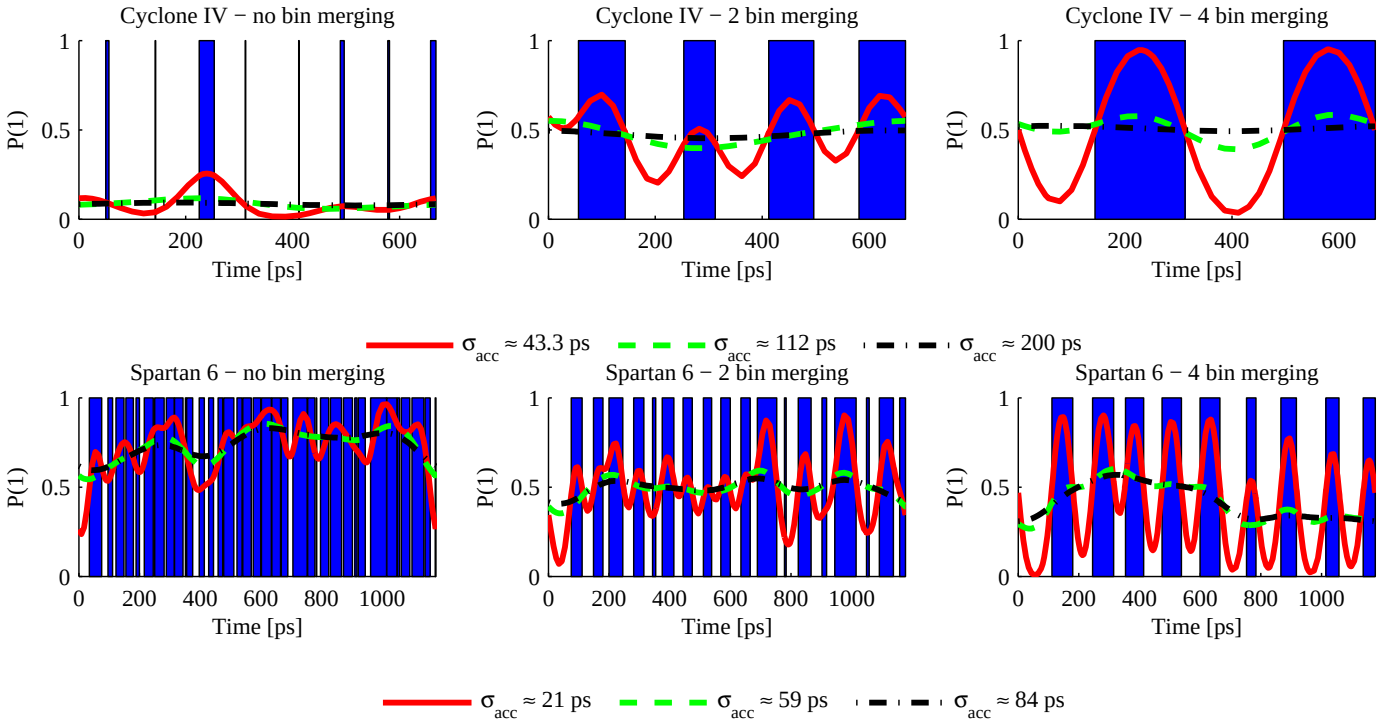


Fig. 3. Probability of generating "1" depending on the mean time position of the ring-oscillator signal edge in the new model.

Figure 3 shows probabilities of the output bit being "1", depending on where in the delay-chain the signal edge of the ring-oscillator is expected – μ_{acc}. The probabilities are calculated for three different values of the accumulated jitter: starting from the jitter approximately equal to the average value of the time bin width in the delay-chain, and then gradually increasing it until the probabilities stop to be considerably different. The three diagrams in the first row show the results for the Intel Cyclone IV, while the three diagrams in the second row show the results for the Xilinx Spartan 6. The probabilities are also calculated for three delay-chain configurations: without merging consecutive time bins, with merging two consecutive bins and with merging four consecutive bins. Merging time bins, which the priority encoder encodes as the same output bit, in a delay-chain improves the linearity of the delay-chain because the resulting time bins have more uniform widths. This optimization has the cost of reduced precision and consequently requires higher jitter accumulation time. We observe that in the case of plain delay-chains – without bin merging, the increase of the accumulated jitter only reduces variations of "1" probabilities, but does not remove bias significantly. Furthermore, the "1" probabilities for certain mean time positions of the signal edge are very close to 0 (in case of Intel Cyclone IV) or to 1 (in case of Xilinx Spartan 6), implying that at these positions almost no entropy can be obtained from the DC-TRNG. It can be seen that for both FPGAs, bin merging significantly improves probabilities of generating "1" compared to configurations without bin merging for higher values of the accumulated jitter. This effect is due to reduced differences of the time widths between newly formed consecutive bins. For lower values of accumulated jitter, as expected, bin merging does not contribute to balancing "1" and "0" probabilities.

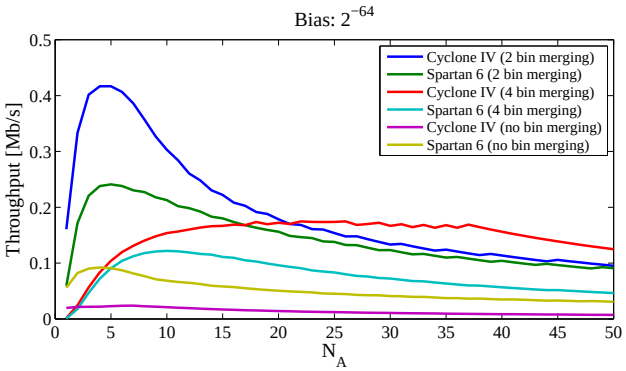


Fig. 4. Throughput of the DC-TRNG depending on the jitter accumulation time for 2^{-64} bias after the optimization.

IV. Throughput Optimization

In order to obtain the highest achievable throughput of the DC-TRNG for a targeted level of bias of the output bit sequence, we applied the throughput optimization procedure for both Intel Cyclone IV and Xilinx Spartan 6 FPGAs. The optimization parameters in our procedure are: the jitter accumulation time – expressed in number of clock periods N_A, the parity filter order – n_f and the number of merged bins. For three different levels of bias we gradually increased

TABLE I
IMPLEMENTATION PARAMETERS AND PERFORMANCES

Xilinx Spartan 6									
No bin merging				2 bin merging			4 bin merging		
Bias	N_A	**PF order**	**Throughput [Mb/s]**	N_A	**PF order**	**Throughput [Mb/s]**	N_A	**PF order**	**Throughput [Mb/s]**
2^{-8}	4	31	0.806	6	8	2.083	12	8	1.042
2^{-16}	4	65	0.384	5	20	1	11	18	0.505
2^{-64}	4	271	0.092	5	83	0.241	10	82	0.122
Intel Cyclone IV									
No bin merging				2 bin merging			4 bin merging		
Bias	N_A	**PF order**	**Throughput [Mb/s]**	N_A	**PF order**	**Throughput [Mb/s]**	N_A	**PF order**	**Throughput [Mb/s]**
2^{-8}	6	78	0.214	4	7	3.571	22	3	1.515
2^{-16}	7	143	0.099	6	10	1.667	23	6	0.724
2^{-64}	7	598	0.024	5	48	0.417	22	26	0.175

TABLE II
COMPARISON BETWEEN OLD AND NEW STOCHASTIC MODEL

Xilinx Spartan 6 (no bin merging)				
N_A	**PF order**	**Throughput [Mb/s]**	**Bias Old Model**	**Bias New Model**
1	11	9.09	2^{-64}	0.3821
Intel Cyclone IV (no bin merging)				
N_A	**PF order**	**Throughput [Mb/s]**	**Bias Old Model**	**Bias New Model**
13	2	3.85	2^{-64}	0.2025

the jitter accumulation time ($t_A = N_A \cdot T_{clk}$) and for each N_A we calculated the smallest order of the parity filter n_f required to obtain the targeted bias. The levels of bias of the DC-TRNG before applying the parity filter are calculated based on the stochastic model described in Section II, and the biggest bias values are used as inputs to our optimization procedure. The throughput for each step of N_A is then calculated as $T = f_{clk}/(n_f \cdot N_A)$. We performed this procedure for three different delay-chain configurations: no bin merging, two bin merging and four bin merging. The results of the throughput optimization procedure for targeted bias of 2^{-64} are shown in Figure 4. In this figure, we can observe that there exists a clear maximum of the throughput for all three delay-chain configurations, and the same observation can be made for other targeted levels of bias.

Table I shows the implementation parameters and achieved throughput of the DC-TRNG for three different levels of bias after the optimization procedure on both Xilinx Spartan 6 and Intel Cyclone IV FPGAs. In all three delay-chain configurations, the throughput decreases with higher security level. Configurations with two consecutive bins merged for both FPGAs increase throughput for all levels of targeted bias. However, further increase in the number of consecutive bins merged into a new time bin decreases maximum achievable throughput due to much wider new bins and therefore higher requirements for the accumulation time. The DC-TRNG on Xilinx Spartan 6 in delay-chain configuration without bin merging achieves almost four times better throughput compared to Intel Cyclone IV for all levels of bias. The reason for this is in substantially higher width differences of consecutive time bins in delay-chains of Intel Cyclone IV FPGA (see Figure 3). On the other hand, it can be observed that in the delay-chain configurations with two bin merging, the throughput of the DC-TRNG is almost two times higher on Intel Cyclone IV. The width differences of every second time bin in the carry-chain of Intel Cyclone IV are much smaller than in the carry-chains of Xilinx Spartan 6. Therefore, two bin merging benefits more to throughput improvement of Intel Cyclone IV, as the resulting time bins have more uniform widths.

In order to evaluate our new stochastic model of the DC-TRNG, we compare it with the one presented in [3] by applying the throughput optimization procedure on the old stochastic model. We then used the obtained parity filter (PF) order and accumulation time (N_A) to recalculate the levels of bias according to the model in this paper. The results for the delay-chain configuration without bin merging are shown in Table II. We observe that the bias calculated according to the stochastic model in [3] is much lower than the bias calculated for the model in this paper. This implies that using the stochastic model in [3] significantly overestimates the min-entropy of the random bits, and thus overestimates security claims of the DC-TRNG.

V. CONCLUSIONS

In this paper, we proposed a refined stochastic model of the DC-TRNG that accounts for the non-linearity of the delay-chain. This model was applied to TRNG implementations on Xilinx Spartan 6 and Intel Cyclone IV FPGAs. Our experimental study showed that the non-linearity of the TDC on these platforms significantly reduces the min-entropy of the output sequence and thus should be taken into account when making the entropy claim. In addition, we investigated the throughput optimization strategy based on the three techniques: merging bins of the TDC, increasing the jitter accumulation time and applying a parity filter for post-processing.

ACKNOWLEDGMENTS

This work is supported in part by the Research Council KU Leuven: C16/15/058. In addition, this work is supported in part by the Flemish Government through G.0130.13N and FWO G.0876.14N, the Hercules Foundation AKUL/11/19, and through the Horizon 2020 research and innovation programme under grant agreement No 644052 HECTOR and Cathedral ERC Advanced Grant 695305.

REFERENCES

[1] A. Rukhin et al., "A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications," *SP 800-22, NIST*, 2010.

[2] W. Killmann and W. Schindler, "A proposal for: Functionality classes for random number generators, version 2.0," [Available online: https://www.bsi.bund.de/EN/Home/home_node.htm], 2011.

[3] V. Rožić, B. Yang, W. Dehaene and I. Verbauwhede, "Highly Efficient Entropy Extraction for True Random Number Generators on FPGAs," *2015 52nd Design Automation Conference (DAC)*, pp.1-6, 2015.

[4] "Spartan-6 FPGA Configurable Logic Block," *Xilinx*,February 2010.

[5] "Cyclone IV Device Handbook,"*Intel*, 2016.

[6] V. Fischer, F. Bernard, N. Bochard and M. Varchola, "Enhancing security of ring oscillator-based TRNG implemented in FPGA," *2008 International Conference on Field Programmable Logic and Applications (FPL)*, pp. 245-250, 2008.

[7] P. Haddad, Y. Teglia, F. Bernard and V. Fischer, "On the assumption of mutual independence of jitter realizations in P-TRNG stochastic models," *Proceedings of 2014 Design, Automation and Test in Europe (DATE) Conference*, pp.1-6, 2014.

[8] B. Yang, V. Rožić, M. Grujić, N. Mentens and I. Verbauwhede, "On-chip jitter measurement for true random number generators," *2017 Asian Hardware Oriented Security and Trust Symposium (AsianHOST)*, 6 pages, 2017.