A Novel Ultra-Compact FPGA-Compatible TRNG

Under the guidance of : Mr S.Sadiq Vali,M.tech.,Ph.D ,
Assistance Professor, ECE & Sanskrithi School Of Engineering , Puttaparthi
1. TS Tabrej ECE & Sanskrithi School Of Engineering , Puttaparthi
2. M Lokesh ECE & Sanskrithi School Of Engineering , Puttaparthi
3. P Kiran Kumar ECE & Sanskrithi School Of Engineering , Puttaparthi
4. BS Yousuf ECE & Sanskrithi School Of Engineering , Puttaparthi

---

Abstract - True random number generators (TRNGs) are widely used in cryptographic applications such as key generation, random padding bits, and generation of challenges and nonces in authentication protocol. This paper proposes a new and efficient method to generate true random numbers in XILINX by utilizing the random jitter of free running oscillators as a source of randomness.The main advantage of the proposed true random number generator utilizing programmable delay lines is to reduce correlation between several equal length oscillator rings, and thus improve the randomness qualities. Generalised FIFO is used to store generated sequence of patterns.clock gating architecture to limit the switching activity of the address decoder which improves the power efficiency of the proposed number generator. element structure is adapted to evaluate the clock cycle to the present ring counter block and to release the clock pulse to the next ring counter block.

Key Words: TRNGs, field programmable gate array (FPGA), ring oscillators

1.Introduction

Computer systems and telecommunications play an important role in modern world technology. The communication and data transfer through computers touches almost every aspect of life, i.e. transferring data, tracking personal data, trading over the internet, online banking and sending emails. As more vital information is transferred through wire or wireless means, the need to safeguard all this data from hackers is growing. All these security concerns emphasize the importance of developing methods and technology for the transformation of data to hide its information content, prevent its modification, and prevent unauthorized use.

2 . Random Number Generators

In addition to cyber security, random number generators (RNGs) are a vital ingredient in many other areas such as computer simulations, statistical sampling and commercial applications like lottery games and slot machines. Random numbers are needed in some areas in computer science, such as authentication, secret key generation, game theory, and simulations. In these applications, particularly numbers should have good statistical properties and be unpredictable and non-reproducible. The number generation in the literature is performed in two different ways as deterministic and nondeterministic.

PRNGs (Pseudo Random Number Generators), which are deterministic random number generators, generate numbers with fast, easy, inexpensive, and hardware independent solutions. The statistical qualities of these numbers produced are close to the ideal. PRNGs must meet the ideal requirements specified to be used especially for authentication and key generation. Therefore, nondeterministic functions are added to the output functions of PRNGs to guarantee these requirements.

TRNGs (True Random Number Generators), which are nondeterministic random number generators, present slower, more expensive, and hardware-dependent solutions compared to PRNGs. Contrary to PRNGs, there is no need to include extra components in the TRNG system designs for R2, R3, and R4 requirements. Because of the unpredictability of random numbers generated by the use of high noise sources with high entropy in TRNGs, it is assumed that the R2 requirement is met. If the R2 requirement is satisfied, then it is assumed that the R3 and R4 requirements are also satisfied.

# of Bits	Length of Loop		Taps
2	3	*	[0,1]
3	7	*	[0,2]
4	15		[0,3]
5	31	*	[1,4]
6	63		[0,5]
7	127	*	[0,6]
8	255		[1,2,3,7]
9	511		[3,8]
10	1,023		[2,9]
11	2,047		[1,10]
12	4,095		[0,3,5,11]
13	8,191	*	[0,2,3,12]
14	16,383		[0,2,4,13]
15	32,767		[0,14]
16	65,535		[1,2,4,15]
17	131,071	*	[2,16]
18	262,143		[6,17]
19	524,287	*	[0,1,4,18]
20	1,048,575		[2,19]
21	2,097,151		[1,20]
22	4,194,303		[0,21]
23	8,388,607		[4,22]
24	16,777,215		[0,2,3,23]
25	33,554,431		[2,24]
26	67,108,863		[0,1,5,25]
27	134,217,727		[0,1,4,26]
28	268,435,455		[2,27]
29	536,870,911		[1,28]
30	1,073,741,823		[0,3,5,29]
31	2,147,483,647	*	[2,30]
32	4,294,967,295		[1,5,6,31]

Table 1.2 : LFSR Statistical Requirements

3 . Existing Technique

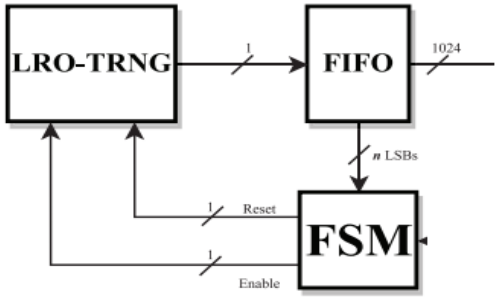

Fig. Block scheme of the TRNG validation testbed

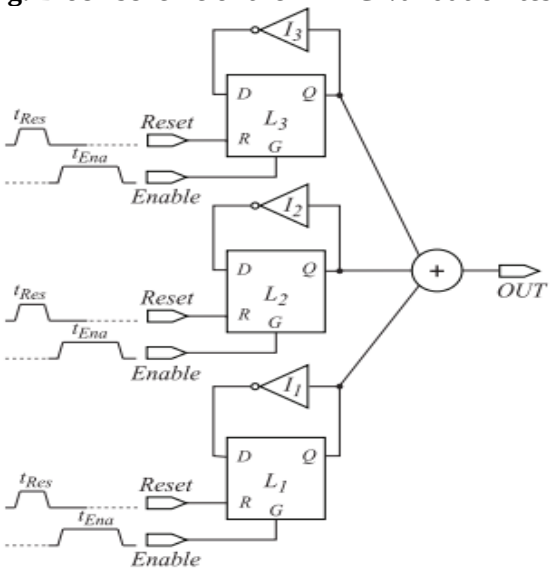

Fig. Existing TRNG Architecture.

The block scheme of the proposed TRNG architecture is shown in Fig. 4.1 The single TRNG cell exploits three latches L1, L2 and L3 closed in a ring oscillator configuration through three inverters I1, I2 and I3 respectively. When a logic '0' is applied to the Gate inputs (G) of the D-Latches, they are in the hold state and are insensitive to variations on the D inputs. On the other hand, when the G inputs of the latches are set to logic '1', the latches become transparent and their outputs Q follow the variations on the D inputs. According to this behavior, when the latches are transparent a free running oscillation comes out, whereas, when they enter in the hold state, the logic value of the output bit is sampled. Since the propagation delay from the D inputs to the Q outputs tDQi, (i = 1, 2, 3) of D-Latches L1, L2 and L3, and the propagation delay tIVi, (i = 1, 2, 3) of the inverters I1, I2 and I3, are dependent on the physical implementation and on the delay of the routing path of the three latches, each RO exhibits an oscillation period (denoted as TROi) equals to:

$$T_{ROi} = \frac{1}{2 \cdot \left(t_{DQi} + t_{IVi}\right)} \quad (1)$$

The operation of the proposed TRNG requires the following excitation sequence:

1) Set the Reset signals to logic '1' for a time tRes in order to reset the outputs of the three D-Latches to logic '0';
2) Set a logic '0' on the Reset inputs and set the Enable signals to logic '1' for a time tEna thus enabling the three ROs;
3) After a time tEna, set a logic '0' on the Enable input in order to sample the output random bit. Each bit has to be generated by means of the above Reset and Enable sequence. It is evident, from these considerations, that the overall throughput (TP) is limited by the time tRes + tEna and the bit-sequence throughput is given by the following equation:

TP=1tRes+tEna

bit s (2) As will clarified in the next sections, the entropy of the output random sequence is directly related to the accumulated jitter and therefore to the excitation time tEna of the ROs. In fact, since an increase of tEna results in an increase of the entropy and in a reduction of the through put, the proposed TRNG architecture requires to optimize the trade-off between these two figures of merit. For this purpose, it is important to remark that the behavior of the proposed LRO-TRNG is quite different from the one of conventional RO-TRNGs.

In fact, previous works exploit the jitter of the ROs and the metastability of D-flip-flops by sampling the stream started in a single instant. It has to be noticed that in the proposed LRO-TRNG, when the gates are closed (i.e., the G inputs are at logic '0' and the data is sampled) a metastable state can be captured, thus increasing the entropy due to the metastability of D-Latches.

However, since process, supply voltage and temperature (PVT) variations affect the oscillation frequencies of the three LROs of a single cell in the same way (i.e., are seen as common mode variations), the XOR operation between the output of the three D-Latches L1, L2 and L3 greatly improves the resilience of the proposed TRNG to PVT variations, thus providing very good statistical performances in spite of working condition variations as will be shown in Section IV-B.

Another important point to remark is that, if the oscillation frequencies of the three LROs are extremely close to each other, locking phenomena can occur , and the statistical performances of the TRNG can be worsened, thus requiring additional postprocessing to perform on the output sequence. To avoid these issues, the oscillation frequencies of the three LROs have to be properly unbalanced during the implementation phase by exploiting the different delays available through the different FPGA blocks and routing resources.

A Lookup Table, as the name suggests, is an actual table that generates an output based on the inputs. Here is an example for a lookup table that is implementing the function of an xor gate.

Input A	Input B	Output C
0	0	0
0	1	1
1	0	1
1	1	0

Table : xorgate truth Table

Try now to image that this table is stored in a small RAM. Inputs A and B are the address pins and C is the data pin. Every time your address pins are changing they are pointing at a different address entry and they are "reading out" the result which is 0 or 1 based on the inputs.

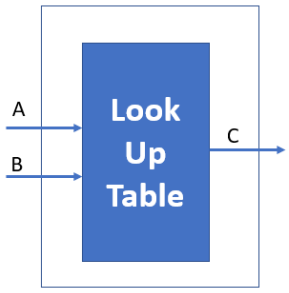

Fig : Lookup Table for xor gate

4. Proposed Architecture

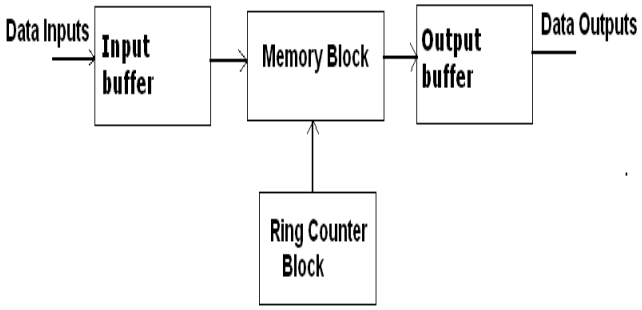

Fig : Proposed Block Of Memory Organisation

S	R	Q(t-1)	Q
0	0	0	0
0	0	1	1
0	1	1	0
0	1	1	0
1	0	1	1
1	0	1	1
1	1	0	X
1	1	1	X

Table : SR Flip Flop Truth Table

Loading binary 1000 into the ring counter, above, prior to shifting yields a viewable pattern. The data pattern for a single stage repeats every four clock pulses in our 4-stage example. The waveforms for all four stages look the same, except for the one clock time delay from one stage to the next. See figure below.

The full form of RAM is Random Access Memory. The information stored in this type of memory is lost when the power supply to the PC or laptop is switched off. The information stored in RAM can be checked with the help of BIOS. It is generally known as the main memory or temporary memory or cache memory or volatile memory of the computer system.

A memory unit is a collection of storage cells together with associated circuits needed to transform information in and out of the device. Memory cells which can be accessed for information transfer to or from any desired random location is called random access memory(RAM).

5. Conclusions

A novel TRNG architecture that leverages the benefits of both the jitter of ring oscillators and the D-Latches' metastability has been proposed. A feedback strategy to randomly set the 4 LSBs of the control word defining the excitation time has been exploited to enhance the randomness and increase the entropy. The randomness of the raw response, shows that also when supply voltage variations are considered the TRNG is able to ful-fill the test requirements

6. Future Scope

The bit-swapping LFSR generates a random test sequence with low switching power by finding hamming distance between two adjacent patterns and minimizing that distance by using combinational logic. To further reduce the average power, dual threshold voltages are assigned. By using this method and finding out the critical and non-critical paths present in BIST and then assigning a low threshold voltage for critical path, and high threshold voltage for non-critical path, a further reduction in total power, especially leakage power, can be obtained.

7. References

[1] Riccardo Della Sala , Davide Bellizia , and Giuseppe Scotti,"A Novel Ultra-Compact FPGA-Compatible TRNG Architecture Exploiting Latched Ring Oscillators" in IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS, VOL. 69, NO. 3, MARCH 2022 [2] M. A. Qureshi and A. Munir, "PUF-rake: A PUF-based robust and lightweight authentication and key establishment protocol," IEEE Trans. Dependable Secure Comput., early access, Feb. 16, 2021, doi: 10.1109/TDSC.2021.3059454.
[3] S. Larimian, M. R. Mahmoodi, and D. B. Strukov, "Lightweight integrated design of PUF and TRNG security primitives based on EFLASH memory in 55-nm CMOS," IEEE Trans. Electron Devices, vol. 67, no. 4, pp. 1586–1592, Apr. 2020.
[4] B. Yang, N. Mentens, M. Grujic, N. Mentens, and I. Verbauwhede, "ES-TRNG: A high-throughput, low-area true random number generator based on edge sampling," IACR Trans. Cryptograph. Hardw. Embedded Syst., vol. 2018, no. 3, pp. 267–292, 2018.
[5] C. Tokunaga, D. Blaauw, and T. Mudge, "True random number generator with a metastability-based quality control," IEEE J. Solid-State Circuits, vol. 43, no. 1, pp. 78–85, Jan. 2008.