A Self-Repairable TRNG

Honorio Martin, Giorgio Di Natale, and Pedro Peris-Lopez

I. Abstract

True Random Number Generators (TRNGs) are one of the basic cryptographic primitives commonly used in security protocols. Mainly these generators are employed to generate keys, nonces, padding plain-text or even like a countermeasure against side channel attacks. Due to their importance for the security, they must be subjected to on-line testing in order to guarantee true randomness at their outputs.

Nevertheless there is an open question in this topic: what should be done with a particular generator, if some test fails?. To the best of our knowledge, all the aforementioned tests raise an alarm but they provide no specification about appropriate actions to recovers in case of a faulty state. In this work we attempt to provide some general guidelines in order to guide designers to create more dependable TRNGs. In addition, we present a case of study where these guidelines are applied.

The mentioned guidelines comprehend a number of recommendations that are valid for general-purpose digital circuits but are also specific guidelines for TRNGs. It is important to note that the applied hardening strategies will depend on the application (and its corresponding security levels), the specific TRNG (e.g., entropy extraction mechanism, post-processing technique, etc.) and the available resources. We have split these guidelines into three groups:

- **Prevention:** In this group typical guidelines to design more reliable systems such as the use of one-hot coding for the FSMs implemented in the statistical tests are included. On the other hand, as some TRNGs are quite sensitive to the surrounding conditions, we propose to check, if possible, the main magnitudes that could affect the randomness (temperature, voltage, frequencies, etc.).
- **Detection & Evaluation:** In order to accomplish the self-repairability for the TRNG, it is a key aspect to be able to detect different kind of faults. Typical scenarios considered in the scientific literature only include the evaluation of faults in the digitized noise source (source of randomness and entropy extractor). We propose that the post-processing block and the different embedded tests are included as possible fault targets. Normally on-the-fly tests or ad-hoc solutions could allow the detection of faults on the digitized noise source or on the post-processing. Redundancy might be applied to detect faults on the on-board supported tests. It is crucial that the detection strategies answer the following questions: *Where is located the fault? How bad/harmful is the fault?*
- **Correction:** Once the fault has been detected and located, the TRNG must be stopped if deemed necessary. After this first action, different measures can be used to recover from the faulty state. These further actions will highly depend on the available resources and the TRNG itself.

Conclusively, we have shown the effectivenest of these guidelines using a case of study. We have defined an experimental framework where a RO-TRNG is used to generate nonces and padding plain-texts –certain degradation on the entropy source is allowed. We have added support for some of the aforesaid guidelines (redundancy, one-hot coding,control main magnitudes, etc.) to the system. In Fig1 is depicted the proposed TRNG scheme. We have duplicated the entropy source, the tests and post-processing blocks. In addition we have added and LFSR that can be used to check the different tests and as a post-processing block. The checker and controller block will handle the test results and the control different blocks.

Finally, using a flexible open-source fault simulator (in particular, LIFTING), we have performed several fault injection campaigns using different fault models (bit-flip, stuck-at-0/1 and transient stuck-at-0/1). The results of these injection show that the proposed enhanced TRNG is able to self-recover from the different fault scenarios considered.

H. Martin is with Department of Electronics Technology, Universidad Carlos III de Madrid,(e-mail: hmartin@ing.uc3m.es)

G. Di Natale is with the LIRMM, Universite Montpellier II,(e-mail: DiNatale@lirmm.fr)

P. Peris-Lopez is with the Computer Security (COSEC) Lab, Universidad Carlos III de Madrid, (e-mail: pperis@inf.uc3m.es)

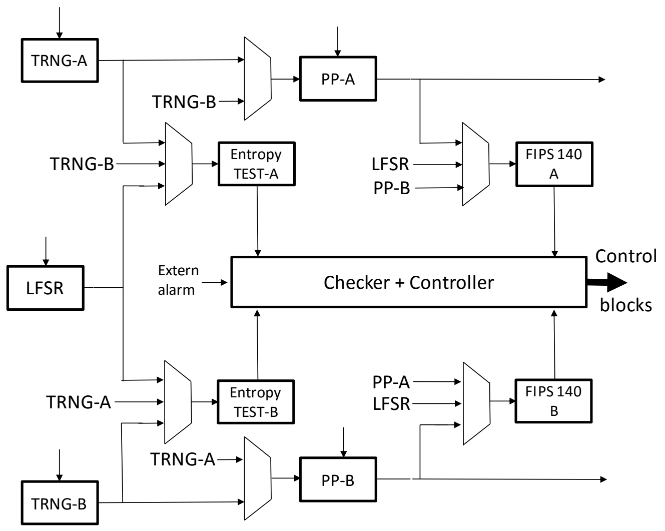

Fig. 1. Self-repairable TRNG scheme