Lab 3: Evaluation of True Randomness and TRNGs

Igli Duro
Department of Computer Science & Engineering
University of South Florida
Tampa, FL

I. Introduction

For this lab, we evaluate the randomness of two different structures of TRNGs and determine if they would be good enough to be used for crytograhpic applications. TRNGs are defined as being truly random as they contain an entropy source that allows for non-deterministic number generation. This entropy source is a physical property or phenomena of the TRNG such as thermal noise, clock jitter, atmospheric readings, etc that is harvested and processed into unpredictable bits of data. To measure the randomness of an TRNG we gathered data files containing randomly generated bytes and used the NIST Statisical Test Suite (STS)to evaluate said files. The NIST test suite is a set of 15 statistical sets aimed to test different features of binary sequences. The results of said tests are p-values that are calculated percentages in support of a given hypothesis. These tests were done with a 99

II. Methods

This project provided two bitstream files, RO_LFSR_TRNG.bit & GARO_TRNG.bit, that were used to program a CMOD-S7 FPGA. Along with that we were also provided with a NIST.py script and NIST statistical test suite. We first created the dataCollection.py script that was used to sample data from both the TRNGs by setting an amount of bytes to be generated and reading it into a .bin file when finished. From our data collection we gathered 4 samples of data, RO_1M.bin, RO_10M.bin, GARO_1M.bin, GARO_10M.bin, or 1 million and 10 million sampled bits for both TRNGs.

We then created a second script, dataAnalysis.py that was used to initialize TRNGTester objects that take in the .bin files as arguments and runs the NIST STS on the data. After running the tests a list of p-values get returned and results were graphed, one for each sample. Some tests generated multiple p-values, so we took the min each time this happened.

III. Results

A. Reading Check

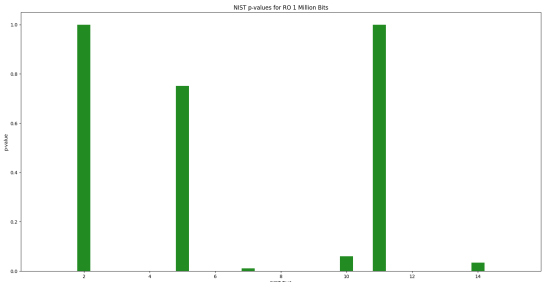
1. *What kinds of applications are TRNGs used for?*
TRNGs can be used in any scenarios where random number generation is needed. Such scenarios can be for security, cryptography, gaming or gambling etc.

2. *What are the 3 major components of a good TRNG design?*
Good TRNG designed if comprised of the entropy source, harvesting mechanism, and post-processing. The entropy source is the physical phenomena that is used to be harvested and it should be aperiodic and unpredictable. This can be thermal noise, clock jitter or other physical properties. The harvesting mechanism is the method used for extracting the entropy from its source and it should be designed in a way to not disturb the process while collecting data. Post-processing is the last step that used to mask any imperfections.

3. *How can we evaluate whether or not a TRNG is truly random?*
To evaluate if a TRNG is truely random, the NIST Statistical Test Suite(STS) can be used. The NIST STS is a set of 15 statistical sets used to test different features of binary sequences.

B. Analysis of Results

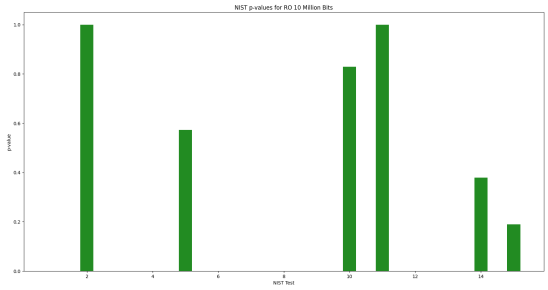
The graphs below show the results of the NIST STS across 4 different data sets. As can be seen, both the RO data sets performed very poorly. The RO 1 million experiment passed 5/15 and RO 10 million experiment passed 6/15 of the tests. However, the GARO data sets performed much better both the GARO 1 million and 10 million experiments passed all 15 tests. There are interesting features to make note of that we'll go over in Section IV.

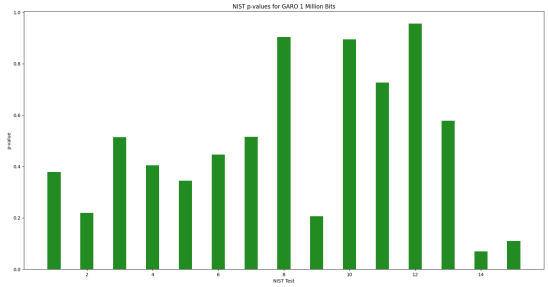


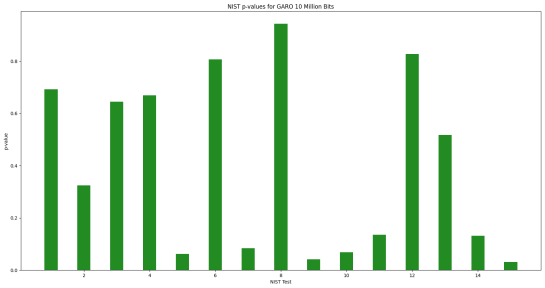
IV. Discussion

As mentioned in Section III, the RO experiments performed poorly while the GARO experiments passed all their tests. However there are some features to make note of. For the RO experiments, the 10 million bit data set did perform better than the 1 million bit data set in the majority of categories. While our reference points are small this could identify a trend of an RO LFSR TRNG being more random with larger sample sets.

For the GARO experiments, some interesting things occurred. Comparing the 10 million and 1 million bit data







sets, the 10 million bit set performed much worse in some categories while performing only slightly better in others. This could mean that with larger samples of data the GARO TRNG has some flaws with how well it can produce random numbers. If larger sets of data were sampled this may result in the GARO TRNG failing some of the NIST tests.

We would conclude that the RO LFSR TRNG is not good enough to be used in cryptographic applications as it failed to pass over half of the NIST tests. Whereas the GARO TRNG is good enough to be used in such applications as it passed all the NIST tests, but to be weary of its use in larger scale applications as it may prove to be less random over larger sets of data.

V. Conclusion

In this lab, the NIST STS was used to evaluate the randomness of two different structures of TRNGs, the RO LFSR TRNG and GARO TRNG. We sampled two sets of data from both TRNGs containing 1 million and 10 million bits. Results from these data sets being used in 15 different NIST tests were then graphed using the p-values produced. We concluded that the RO LFSR did not perform well enough to be used in cryptographic applications as it failed more than half the tests. While the GARO TRNG performed highly and can be considered good enough for cryptographic applications as it passed all the NIST tests.