Random numbers from metastability and thermal noise

D.C. Ranasinghe, D. Lim, S. Devadas,
D. Abbott and P.H. Cole

Pseudorandom number generators are algorithmic and thus, predictable. Ideally cryptography, simulation and modelling applications require a source of true random numbers. Presented is a true random number generator that exploits metastablity and thermal noise. The novelty is that the low-cost design can be fully integrated with standard CMOS technology.

Introduction: Random number generators are important in a number of applications, however the most significant is their application in cryptography, such as the generation of keys [1]. Cryptographic requirements and large Monte Carlo simulations continue to advance development and research into true random number generators. While the output sequences of pseudorandom number generators are predictable, physical phenomena such as the decay of radioactive isotopes can be used to exploit and generate random numbers devoid of predictability. However, such techniques cannot be conveniently integrated on standard CMOS. In this Letter we present a fully integrable generator based on metastability and thermal noise as sources of randomness, demonstrating results from a fabricated circuit.

Metastability: When a signal violates a bistable device's signal setup and hold timing requirements, the device output becomes unstable [2]. This undesirable phenomenon where the final state of the device is unpredictable and is determined by thermal noise is known as metastability [2]. Previous attempts [3, 4] to use metastability as a source of randomness have dealt poorly with the sensitivities of the occurrence of a metastable condition to temperature and voltage variations since these factors affect gate delays, signal propagation times and thermal noise intensity. We present a novel repeatable method of creating metastability to produce random sequences in a practical range of environmental conditions.

Generator design: A secret key extraction technique from the manufacturing variation in ICs [5] provides a suitable solution to create metastability on a recurring basis. The technique employs a physically unclonable function (PUF) circuit which has an exponential number of delay path configurations determined by a challenge input. The observation of the results of a PUF reveals that for certain challenges, the setup and hold time violation of an arbiter (D-latch) leads to unpredictable responses as the arbiter enters into a metastable condition. The responses from these challenges can be used to generate a random bit stream. Experimental results show that approximately 10 out of 10 000 challenges produce random responses within a temperature tolerance of 5°C. Based on the performance of PUF circuits, it takes 0.5 s to test the randomness of 10 000 challenges by 1000 repeated measurements [5].

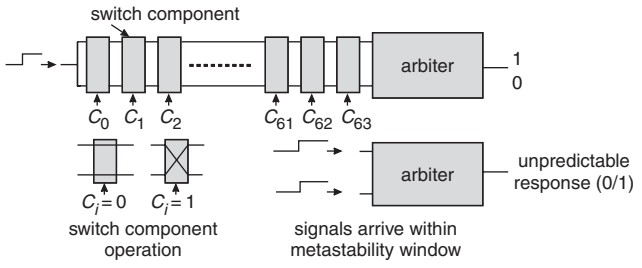

Fig. 1 *PUFRNG circuit*
Chip built in TSMC's 0.18 μm, single poly, six-level metal process with standard cells. PUFRNGs mounted on circuit board and interfaced to PC using JTAG interface for testing

The block diagram in Fig. 1 depicts the structure of a PUF random number generator (PUFRNG) circuit. The circuit accepts an n bit challenge $c_0, c_1, c_2, \ldots, c_{n-1}$ to form two delay paths in 2^n different configurations. To generate a response bit, two delay paths are excited simultaneously to allow the transitions to race against each other. The arbiter block at the end of the delay paths determines which rising edge arrives first and sets its output to 0 or 1. The actual implementation of arbiter-based PUFs in [5] uses 64 bit challenges. When two transitions violate the setup and hold times of the arbiter, the arbiter becomes metastable and generates random responses. The switch components are implemented using a pair of two-to-one multiplexers.

The total power consumption of a PUFRNG circuit is about 130 μW in our implementation. However, input and output functions of the generator are responsible for most of the power consumption and the power consumption of the generator core is relatively small.

Post-processing: The sensitivity of the generator to the temperature of the surrounding environment can be minimised by the use of eight different PUFRNG circuits each calibrated at intervals of approximately 10°C, increasing the tolerance of operation to around 80°C. The experiments were run in an oven with thermostat control to provide cyclic temperature changes. The output bit stream was then a result of an exclusive OR operation on each of the individual bit streams.

It is necessary to process the original bit stream to remove bias. This was conducted by passing the bit stream through an entropy distillation process. The method adopted is detailed in [1], and it involves parsing the bits in pairs, and then transforming them according to the scheme outlined in [1]. This method resulted in a typical reduction in the original PUFRNG bit stream in the range of 65 to 75%. However this is only a general estimate which will tend to depend on the environmental conditions. The PUFRNG was used to obtain 4.5 million random bits after post-processing of the output sequence for testing.

Evaluation: To evaluate the random number generator the output was subjected to dynamic system analysis to analyse the complex system used for random number generation to investigate whether the system exhibits behaviour that is either random or deterministic. The bit stream subdivided into 32 bit blocks provided the random numbers for the following analysis.

Taken's embedding theorem [6], allows the reconstruction of the phase space of the system dynamics to determine underlying patterns. The reconstruction of the phase space can be performed from a finite time series of observed random numbers (observation of a single variable). Employing Taken's theorem requires the determination of the delay τ and the embedding dimension d. The popular average mutual information algorithms used provided a value of unity for τ. Cao's [7] $E_2(d)$ metric was applied to identify whether the system is random. Here, $E_2(d)$ remains at unity for all values of d, suggesting a random system and hence the system is not chaotic. Fig. 2 provides visual confirmation of the lack of structure in the 3-D phase space reconstruction.

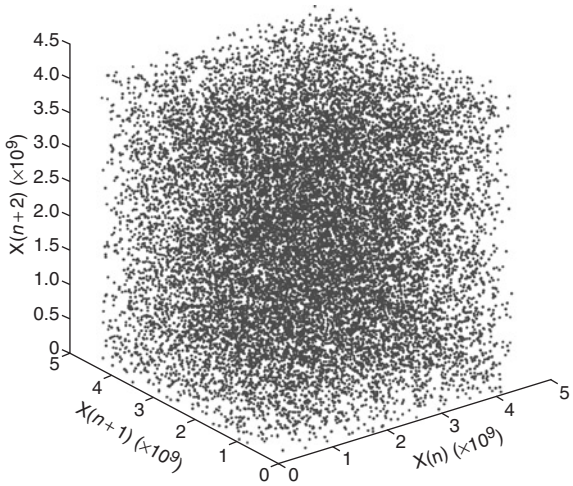

Fig. 2 *3-D phase space plot of PUFRNG output using unity delay*
Weak attractor is result of finite block size (32 bits)

Statistical testing: The PUFRNG output was subjected to a battery of benchmark statistical tests used by the National Institute of Standard and Technology (NIST). The following tests were used in the testing process: frequency test, frequency block test (block size of 128 bits),

the runs test, test for the longest run once in block, the binary matrix rank test, the non-overlapping template matching test (template size of 9 bits), the overlapping template matching test (template size of 9 bits), the serial test (block size of 16 bits), the approximate entropy test (block size of 10 bits) and the cumulative sums test. A detailed discussion of these tests and their interpretations can be found in [8]. The bit stream subdivided into 115 sequences of 38912 bit blocks were used for all the above tests.

Each NIST statistical test assesses a binary sequence to establish whether there is significant evidence to suggest that the null hypothesis (H_0—the sequence is random) should be rejected in favour of the alternative hypothesis (H_1—the sequence is not random), based on a P-value. The P-value indicates the strength of the evidence against the null hypothesis. A significance level of 0.05 was used for this work. Hence a P-value < 0.05 implied that H_0 should be rejected in favour of H_1.

The bit stream successfully passed all the tests used from the NIST test suite. However to further assess the validity of the sequences the proportion of sequences passing each test should be greater than 0.95, while the distribution of P-values should be approximately uniform [8]. The former can be evaluated by calculating a P-value of P-values. A significance level of 0.0001 was used to assess the uniformity of P-values [8]. These additional tests are readily passed by the sequences generated from the PUFRNG.

Conclusion: The assessment of the PUFRNG output qualifies the unpredictable random nature of the bit stream under varying temperature changes. The PUFRNG is a cost-effective, power-conservative solution to produce millions of random bits rapidly by fabricating a number of PUFRNGs on a single IC using standard CMOS technology. A 64-stage PUF circuit costs less than 1000 gates. Additionally, various kinds of low power techniques such as subthreshold logic design and multi-threshold CMOS designs can be utilised to further reduce the power consumption.

However, the PUFRNG requires some overhead due to post-processing and calibration prior to its use. Nevertheless the unique ability to calibrate the PUFRNG very rapidly allows the generator to adapt to external influences and to fine-tune the generator for greater performance. Future work will involve investigations into the effects of supply voltage on the performance of the PUFRNG.

28 April 2005
Electronics Letters online no: 20051559
doi: 10.1049/el:20051559

D.C. Ranasinghe, D. Abbott and P.H. Cole (*School of Electrical and Electronic Engineering, University of Adelaide, SA 5005, Australia*)

E-mail: damith@eleceng.adelaide.edu.au

D. Lim and S. Devadas (*Massachusetts Institute of Technology, 77 Massachusetts Ave., Cambridge, MA, 02139, USA*)

References

1 Menezes, A., Van Oorschot, P., and Vanstone, S.: 'Handbook of applied cryptography' (CRC Press, 1996)
2 Couranz, G.R., and Wann, D.F.: 'Theoretical and experimental behaviour of synchronizers operating in the metastable region', *IEEE Trans. Comput.*, 1975, **24**, (6), pp. 604–616
3 Walker, S., and Foo, S.: 'Evaluating metastability in electronic circuits for random number generation'. IEEE Computer Society Workshop on VLSI, 2001, pp. 99–102
4 Bellido, M.J., Acosta, A.J., Valencia, M., Barriga, A., and Huertas, J.L.: 'A simple binary random number generator', *Electron. Lett.*, 1992, **28**, (7), pp. 617–618
5 Lee, J.W., Lim, D., Gassend, B., Suh, G.E., Van Dijk, M., and Devadas, S.: 'A technique to build a secret key in integrated circuits for identification and authentication applications'. Symp. on VLSI circuits, Honolulu, 2004, pp. 176–179
6 Taken, F.: 'Detecting strange attractors in turbulence', Dynamical Systems and Turbulence, *Lect. Notes Math.*, Springer-Verlag, 1981, **898**, pp. 365–381.
7 Cao, L.: 'Practical method for determining the minimum embedding dimension of a scalar times series', *Physica D*, 1977, **110**, pp. 43–50
8 Rukhin, A. *et al.*: 'A statistical test suite for random and pseudorandom number generators for cryptographic applications'. NIST Special Publication 800-22, 2001.