# Initial SRAM State as a Fingerprint and Source of True Random Numbers for RFID Tags

Daniel E. Holcomb, Wayne P. Burleson, and Kevin Fu

University of Massachusetts, Amherst MA 01002, USA,
{dholcomb, burleson}@ecs.umass.edu, kevinfu@cs.umass.edu
http://www.rfid-cusp.org/

**Abstract.** RFID applications create a need for low-cost security and privacy in potentially hostile environments. Our measurements show that initialization of SRAM produces a physical fingerprint. We propose a system of Fingerprint Extraction and Random Numbers in SRAM (FERNS) that harvests static identity and randomness from existing volatile CMOS storage. The identity results from manufacture-time physically random device threshold mismatch, and the random numbers result from run-time physically random noise. We use experimental data from virtual tags, microcontroller memory, and the WISP UHF RFID tag to validate the principles behind FERNS. We show that a 256byte SRAM can be used to identify circuits among a population of 160 virtual tags, and can potentially produce 128bit random numbers capable of passing cryptographic statistical tests.

## 1 Introduction

Identification and random number generation are important primitives in RFID tag circuits. The extreme constraints of passive RFID applications require that both be accomplished with minimal cost, and without sacrificing quality. A static identity is required by nearly all RFID applications, including tracking and authentication. Random numbers are essential to many cryptographic schemes; if random numbers can be guessed with any accuracy, the security of any scheme which relies on them is broken.

Our system for Fingerprint Extraction and Random Numbers in SRAM (FERNS) uses SRAM physical fingerprints for identification and generation of random numbers. The frequent powering up of passive tags is continually generating fingerprints, providing an opportunity to use memory without disrupting computation, and making SRAM a viable information source.

The FERNS approach to identification and random number generation is to extract both from the physical fingerprints of SRAM, allowing reuse of existing RAM cells. We validate FERNS through experiments on three platforms. The first is a population of 160 virtual tags. Each virtual tag is a 256byte logical segment of a 512kbyte SRAM chip [4], read out using the Altera DE2 development board [1]. The second platform is a population of 10 TI MSP430F1232 microcontrollers [19]. This particular microcontroller was chosen because its

ultra lower power design is a good match for RFID technology in general, and because it is the microcontroller used on the target application for this work, Intel's WISP wirelessly-powered platform for sensing and computation [14, 12]. The third platform is a population of 3 WISPs. The WISP is passively powered at 915MHz in the ultra high frequency band, and transmits data in 64 bit packets according to the Electronic Product Code Gen 1 specification, allowing communication with commercially available RFID readers. Because the WISP is a flexible microcontroller-based passive RFID tag with 256bytes of RAM, it is an ideal platform for FERNS. Each of the three platforms serves a purpose; the virtual tags allow for collection of a large questionably representative dataset, the MSP430s allow for collection of a modest sized dataset using a highly representative technology but are not passively powered, and the WISP provide a smaller dataset using passive power. Our experiments on these three platforms demonstrate that circuits can be identified among a population using only SRAM fingerprints, and shows that hashed fingerprints can pass basic statistical tests for randomness.

The remainder of this paper is organized as follows. Section 2 gives the related work in the fields of tag identification and random number generation. Section 3 formally introduces the FERNS system, and gives its physical foundations. Sections 4 and 5 discuss applications in fingerprint identification and true random number generation, respectively.

## 2 Related Work

The low cost of RFID circuits constrains their functionality. A typical EPC class 1 tag has 1,000-4,000 gates, with class 2 tags having several thousand more [10]. To work within these constraints, low-cost security solutions for RFID have been the subject of much research, including the notable work on light-weight cryptography in [9]. Low-cost is often accomplished through serializing computation, creating higher storage requirements, making FERNS an attractive alternative. FERNS enables low-cost implementations of cryptography by providing static identities and random numbers using existing hardware.

### 2.1 Identification

In the most general terms, RFID circuits can be identified either through the use of non-volatile memories or the use of some identifying physical characteristic, which we call fingerprinting. The non-volatile approach involves programming an identity into a tag at the time of manufacture using EPROM, EEPROM, flash, fuse, or more exotic strategies. While non-volatile identities are static and fully reliable, they have drawbacks in terms of the process cost and the area cost of supporting circuitry. Even if only a small amount of non-volatile storage is used, the process cost must be paid across the entire chip area. Additionally, supporting circuitry such as charge pumps for tunneling oxide devices, and programming transistors for fuse devices, are needed. A notable alternative is

implemented using electron beam programming and single transistor cells in 90nm SOI [21].

The fingerprint approach to identification consists of using the process variation that is inherent in the manufacture of integrated circuits for differentiation between chips. Process variation comes in many forms, including lithographic variations in effective feature size, and random threshold voltages. In terms of producing identifying characteristics, it is generally not the absolute variation that matters, but instead the mismatch between the spatially correlated devices that are implementing the function. Lithographic variations are correlated among local devices and devices occupying the same within-field position on different chips [2]. Variations in threshold voltages are due to random fluctuations in the concentration of dopant atoms, and are not spatially correlated [18]. Thus, random threshold assignment makes an ideal identifying characteristic.

Simple physical fingerprints can be used to generate identifying signatures. The circuit in [7] is designed for RFID identification using MOS device random threshold assignment as the identifying characteristic, with supporting circuitry to indirectly measure these threshold voltages. A related version of this approach for identifying RFID tags is illustrated in [15]. Here an array of 10 transistor physical functions are used, where each is operated like a cross coupled NOR cell, with the second input being used to reset the cross coupled devices. When the identity is desired, the cross coupled state nodes are pulled low. Once released, they will transition to a stable state, where the choice of stable state is determined by threshold mismatch. The physical uncloneable function (PUF) of [3] uses a physical race condition where the racing paths are selected by the applied input. The identifying output is determined by the relative delays of the two paths. The same PUF circuit is used for both random number generation and authentication in [16], by finding and then persistently applying inputs that cause races between well-matched paths, leading to each binary outcome with equal probability. The advantage to using physical fingerprints is their use of ordinary CMOS process, and the fact that no programming step is required. The most significant drawback to physical fingerprint identification is that the identities are impacted by noise. FERNS is comparable to these physical fingerprint methods. The primary difference is that FERNS harvests the identity from existing RAM arrays, instead using a dedicated circuit for this purpose.

### 2.2 Random Number Generation

The approaches to creating random numbers can be broadly classified into two main categories, True Random Number Generation (TRNG) and Pseudo Random Number Generation (PRNG). TRNGs rely on a physically random process as a source of entropy, whereas PRNGs produce outputs that have statistical properties of random numbers, yet are fully deterministic. For this reason, TRNGs are desirable for security applications. The random process that is harvested varies across TRNG designs. One physically random processes in integrated circuits is thermal noise, which describes voltage variations that exist

when a conductor is in equilibrium [8, 5]. A related physically random process is shot noise, which describes the randomness in a current as it begins to flow through a conductor [13]. To create a random number from such a physically random process, a harvesting method is required. A well-known method for harvesting this noise is through its manifestation in the jitter of free-running oscillators, shown recently in [17]. A second way to extract thermal or shot noise is by amplifying the noise to a measurable level, by use of direct amplification or through the large gain that is inherent in metastable CMOS devices [20, 6]. FERNS is comparable to these true random number generators. The primary difference is that FERNS harvests the randomness from existing RAM arrays, instead using a dedicated circuit for this purpose.

## 3 SRAM as a Physical Fingerprint

FERNS is built upon the idea that the stabilization of each SRAM cell at power-up reveals a physical fingerprint. Uninitialized SRAM is normally considered to be in a logically unknown state. By descending below the logical level of abstraction, and considering RAM to be a physical fingerprint, a wealth of information is found. With a RAM cell being the required circuitry for merely storing a bit, each SRAM cell is perhaps the smallest possible physical fingerprint capable of producing a digital output. In the remainder of this section we build up the FERNS system, starting at the circuit level by identifying why SRAM initialization serves as a physical fingerprint.

Each bit of SRAM is a 6 transistor storage cell, consisting of cross coupled inverters and access transistors. Each of the inverters drives one of the two state nodes, labeled A and B. When the circuit is unpowered, both state nodes are low (AB=00). Once power is applied, this unstable state will immediately transition to one of the two stable states, either '0' (AB=01) or '1' (AB=10). The choice between the two stable states depends on threshold mismatch and thermal and shot noise. Because the stabilization depends only on mismatch between local devices, the impacts of common mode process variations such as lithography, and common mode noise such as substrate temperature and supply fluctuations, are minimized. The sources of identity and randomness are shown in Fig 1.

RAM cells with large threshold mismatches are heavily skewed toward one state, and are immune to the small disturbances of noise. Because these cells consistently stabilize to the same state, and vary across chips, they can be used to identify the tag. RAM cells that happen to have well matched thresholds are highly sensitive to noise. In these cells, physically random noise will determine the outcome. These cells act as noise in the identity, but can be constructively used in random number generation. This is shown in Fig 2. We propose FERNS as a means of using initial SRAM state as a source of both identifying fingerprints and true random number generation. To avoid revealing entropy, the same bits cannot be used for both purposes. The following terminology is used, adopted from human fingerprinting.

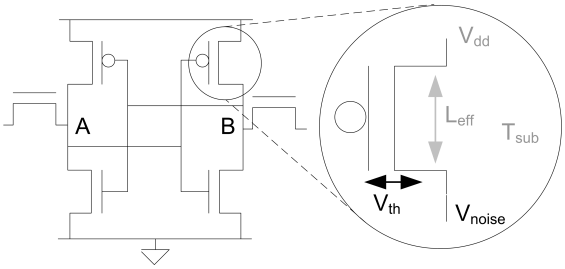


**Fig. 1.** SRAM cell with relevant process variation and noise shown. Threshold voltage mismatch is the source of ID. Noise is the source of randomness.

1. Latent Fingerprint - A fingerprint generated in RAM at initialization. A latent print represents a single data point and can be impacted by noise.
2. Known Fingerprint - An intentional fingerprint that is cataloged for matching against latent prints. Known prints are obtained by averaging many latent prints, to eliminate the effects of noise.

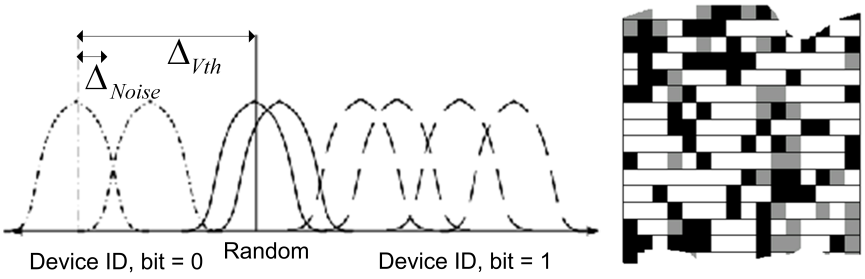


**Fig. 2.** Some bits in FERNS function as reliable identifiers, while others are random. When the skew due to process variation is large, the minor influence of noise is insufficient to sway the outcome of the bit. When the skew due to process variation is small, the influence of noise is sufficient to determine the outcome of the bit. The figure at right is created from observed SRAM physical fingerprints. Each row in the figure corresponds to a word of memory, and the pixels represent individual bits within that word of memory. Bits shaded black reliably initialize to the 0 state, unshaded bits reliably initialize to the 1 state, and those shaded gray can initialize to either state.

## 4 FERNS for Identification

FERNS extracts a usable fingerprint from the initial state of SRAM. In evaluating the identifying qualities of fingerprints, we consider three relevant quantities.

1. The distance between latent fingerprint and known fingerprint when both are generated by the same device. A close distance indicates a highly reliable fingerprint.
2. The distance between latent fingerprint and all known fingerprints generated by different tags. If this distance overlaps with the above, then it may not be possible to always determine a correct match based on a single print.
3. The distances between all known fingerprints. Because noise impacts latent fingerprints and not known fingerprints, this distance indicates how different the fingerprints would be in the absence of noise.

### 4.1 Potential Security Features of Using FERNS for Identification

Identification via the slightly randomized fingerprint of FERNS offers some potential security advantages. The set of possible latent prints that indicate a given device can be thought of as a large ID space for the device; given the number of bits used, the odds of an ID legitimately repeating are exceedingly slim. As would be expected, no tag ever generated the exact same 256byte latent print twice during testing. A reader might thus prevent replay attacks by cataloging a history of the latent prints generated by a device. The condition for authenticating a tag would then be to force the tag to produce a new latent print that closely matches a known print. This would only prevent replay attacks, as an intelligent adversary could still easily generate randomized prints himself. This is similar to human fingerprinting, where it is not impossible for an adversary to reproduce a fingerprint; it is only impossible for him to reproduce a fingerprint using another human finger.

### 4.2 Analysis of Fingerprint Matching Results

In this section, the identifying quality of the fingerprints is explored based on experimental data. A simple Hamming Distance based matching is used, with the implementation of an efficient fingerprint extractor left as an open question. For each platform, all latent fingerprints are compared against all known fingerprints, for a measure of how reliably a single latent fingerprint can be matched to a known print. We also look for pairwise similar known prints, indicating similar devices.

Virtual tags allow for testing of process corner cases. Virtual tags that occupy the same positions on different SRAM chips have correllated within-field positions, while tags from nearby locations on the same SRAM chip have correlated wafer positions, as is shown in Fig 3. Without virtual tags or custom silicon, there would be no way determine the relative wafer positions of the tags being compared. The observation that neither of these corner cases showed a

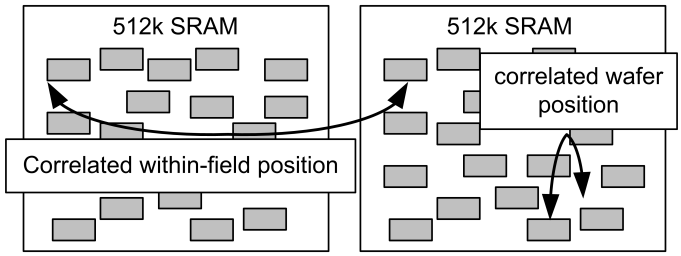


**Fig. 3.** Virtual tags allow for the testing of RAM arrays with highly correlated positions on the same wafer, and of arrays with the same within-field position on different dice.

strong correlation supports the claim that primary source of the differentiation is threshold mismatch due to dopant concentrations.

A population of 160 virtual tags are created across eight 512kbyte SRAM chips, using the same addresses on each chip. A known fingerprint is generated for each of the tags, and 800 total latent fingerprints are generated, allowing for 128,000 possible matchings between latent and known prints. All incorrect matchings differ by at least 685 of the 2,048 bits; all correct matchings differ by less than 109 bits, as shown in Fig 4. Comparing known prints to other known prints yields only slightly better results, as the latent prints are already relatively stable. Interestingly, the 5% bit error rate of the virtual tags approximates the reliability of some state-of-the-art dedicated identification circuits [15]. The spacing between known prints is suboptimal because more total bits tend towards the one state than the zero state, seeming to indicate a systematic skew in design or layout.

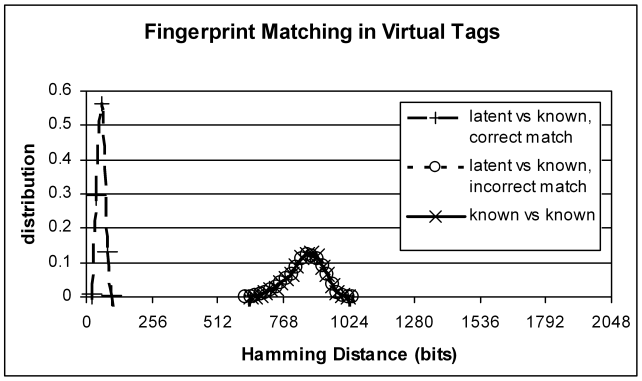


**Fig. 4.** Matching between fingerprints in virtual tags shows that fingerprints contain a reliable identity.

FERNS identification is also tested on the population of MSP430 microcontrollers. Communication and power was accomplished using a JTAG debugger. The MSP430 SRAM looks to be more susceptible to noise than the virtual tags. Among the population of MSP430s, 300 latent fingerprints are generated for comparison against the 10 known fingerprints. Among the 2,700 incorrect matchings, less than 10 came within 600 bits of each other. Among the 300 correct matches, only 4 differed by more than 425 bits, as is shown in Fig 5. The furthest distance between correct matches exceeds the closest distance between incorrect matches; this implies that an appropriate matching threshold for preventing false positive identification may not be fully reliable, and some tags could be forced to produce several fingerprints before being identified. However, the results overall indicate that there is a usable differentiation between tags. It should be noted that some latent fingerprints are extreme outliers, seeming to indicate a possible cause other than random noise.

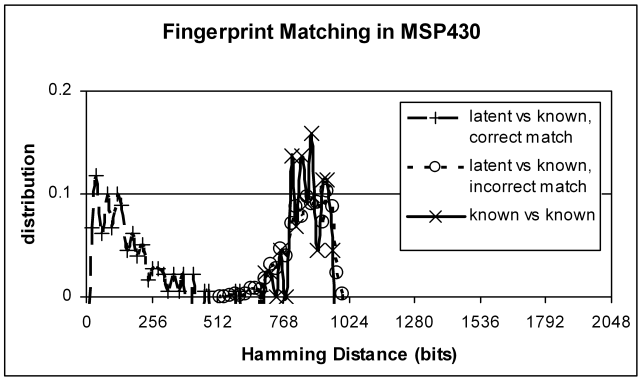


**Fig. 5.** Matching between fingerprints in a population of 10 MSP430s. Fingerprinting is reliable, but has a possibility of false positive and false negative identification, as discussed in Section 4.2.

FERNS identification is also explored on the small population of WISPs. To allow for comparisons among a larger population of identities, a fingerprint is extracted from each of 5 distinct 64 bit blocks on each WISP. This provides 15 known fingerprints, which are matched against 150 latent prints. Among the 2,100 incorrect matchings, none came within 20 bits of matching. Among the 150 correct matches, only 3 differed by more than 8 bits. No overlap is seen between incorrect matches and correct matches, indicating the existence of a reliable fingerprint in the WISP as shown in Fig 6. This result implies that passive power does not influence the fingerprints.

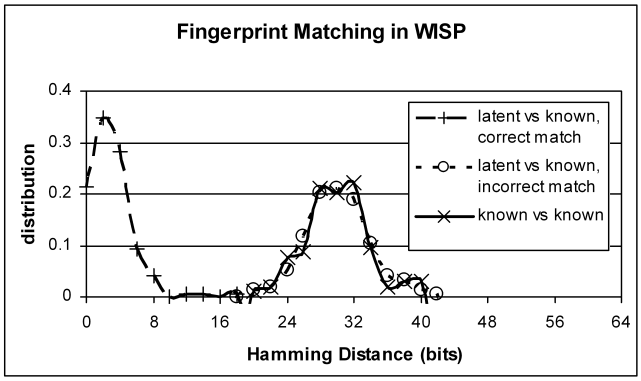


**Fig. 6.** Matching between 15 fingerprints in a population of 3 WISPs shows a reliable identity

## 5 FERNS for True Random Number Generation

FERNS allows for the capture of physically random noise on well matched devices as an entropy source for TRNG. In essence, the bits that happen to be constructed of well matched devices are tiny 6 transistor TRNGs. Because the well matched devices are randomly scattered according to dopant concentrations, the randomness is unpredictably scattered throughout the SRAM state, and must be collected by use of an entropy extractor. The parallel true random number generation of FERNS relies on the law of large numbers to ensure that entropy is harvested from within the array. This is contrary to most TRNGs, which generate predictable amounts of entropy serially.

### 5.1 Potential Security Features of Using FERNS for TRNG

This unusual TRNG has interesting security implications, and is potentially resistant to attack for several reasons. The random locations of the well matched cells in the SRAM array obscure the location of the entropy sources. If the attacker had designed the tag himself, he would not know the location of the TRNGs. If an attacker did identify the random bits, the proximity of the SRAM cells ensures that any attempt to influence one cell is also likely to influence others, making the cells difficult to predictably control by means of a directed attack. Further, the distributed parallelism of RAM provides a natural resiliency against attacks.

### 5.2 Analysis of TRNG Results

In this section, preliminary findings on the random number generation capabilities of FERNS are presented. The NIST test suite [11] is used to test the

approximate entropy of various bitstreams. We use the virtual tags as our extraction test case because they are the least random of our three experimental platforms, and thus present the most challenging extraction scenario. Using results from our experiment, we calculate the binary entropy of each bit of virtual tag memory by considering its history of initial values, and find that an average of 0.103 bits of entropy is produced per raw bit of memory. This implies that 210 bits of entropy are generated by the 2,048 raw bits, supporting the plausibility of extracting 128 random bits. The NH polynomial (PH) universal hashing function from [22], and shown in equation (1), is used as an entropy extractor. Each block of key and message that are input to the hash (2) are the raw values produced by 64 bits of memory(3). While the hashing function is currently implemented in software, it was designed for low power hardware implementation.

$$PH_K(M) = \sum_{i=1}^{8} (m_{2i-1} + k_{2i-1})(m_{2i} + k_{2i}) \tag{1}$$

$$M = (m_1, ..., m_{16}) \quad K = (k_1, ..., k_{16}) \tag{2}$$

$$m_i, k_i \in GF(2^{64}) \tag{3}$$

The approximate entropy test from the NIST suite is used to evaluate the quality of the random numbers produced. For the sake of comparison, we test the raw bits that are input to the universal hashing function, as well as the extracted random output. Based on testing 800 blocks of 128 bits each, the raw bits are shown to fail the approximate entropy test, while the hashed output passes, see Fig 7. This demonstration is preliminary; in future work the min-entropy of the raw bits will be quantified to ensure the randomness of the hashed output. The result shown illustrates that random numbers can be extracted from the initial state of SRAM by use of an entropy extracting code, supporting the TRNG aspect of FERNS.

| *dataset* | $C1$ | $C2$ | $C3$ | $C4$ | $C5$ | $C6$ | $C7$ | $C8$ | $C9$ | $C10$ | $PVAL$ | $PROP$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RAW | 790 | 8 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0.0000 | 0.0962 |
| HASHED | 100 | 91 | 71 | 73 | 73 | 79 | 65 | 92 | 73 | 83 | 0.1188 | 0.9912 |

**Fig. 7.** Output from NIST approximate entropy test. The non-uniform distribution of p-values in the raw data indicates a lack of entropy, the uniform distribution of p-values in the hashed data indicates high entropy.

## 6 Conclusions and Future Work

RFID applications present a unique set of challenges in terms of both security and cost. By introducing SRAM as a physical fingerprint we explore a system capable of producing usable fingerprints and true random numbers through reuse of existing CMOS circuitry. As technology continues to scale, RFID tags

will become more capable and include progressively larger volatile memories, making FERNS an increasingly attractive option. In future work, the quality of the TRNG needs to be further specified in terms of min entropy, under a variety of scenarios. The quality of the ID needs to be explored in varied environments, and over a longer duration. Vulnerability to side channel attacks will also be explored. Further work is currently underway.

## Acknowledgments

We thank Thomas Heydt-Benjamin for his discussion on potential applications for SRAM physical fingerprints, Intel Research and Joshua R. Smith for providing and supporting the WISP platforms, Hee-Jin Chae and Salma Mirza for their assistance with using the WISP and MSP430, and Adam Stubblefield for reviewing an early manuscript and providing feedback that has helped to guide the development of this work. This material is based upon work supported by the National Science Foundation under Grant No. 0627529.

## References

1. Altera Corporation: Altera's Development and Education Board. (2007) Referenced at http://www.altera.com/education/univ/materials/boards/unv-de2-board.html.
2. Friedberg, P., Cheung, W., Spanos, C.: Spatial variability of critical dimensions. In: VLSI/ULSI Multilevel Interconnection Conference XXII. (2005) 539–546
3. Gassend, B., Clarke, D., van Dijk, M., Devadas, S.: Silicon physical random functions. In: Proceedings of the 9th ACM Conference on Computer and Communications Security. (2002) 372–373
4. Integrated Silicon Solution, Inc.: IS61LV25616AL - 256K x 16 High Speed Asynchronous CMOS Static RAM With 3.3V Supply. (2006) Referenced at http://www.issi.com/pdf/61LV25616AL.pdf.
5. Johnson, J.: Thermal agitation of electricity in conductors. Phys. Rev. 32, 97 (1928)
6. Kinnimet, D.J., Chester, E.: Design of an on-chip random number generator using metastability. In: Proceedings of the 28th European Solid-State Circuits Conference, ESSCIRC 2002. (2002) 595–598
7. Lofstrom, K., Daasch, W., Taylor, D.: IC identification circuit using device mismatch. In: Digest of Technical Papers, 2000 IEEE International Solid-State Circuits Conference. (2000) 372–373
8. Nyquist, H.: Thermal agitation of electric charge in conductors. Phys. Rev. 32, 110 (1928)
9. Poschmann, A., Leander, G., Schramm, K., Paar, C.: New light-weight crypto algorithms for RFID. In: Proceedings of the 2007 IEEE International Symposium on Circuits and Systems. (2007)
10. Ranasinghe, D.C., Lim, D., Cole, P.H., Devadas, S.: A low cost solution to authentication in passive RFID systems. In Cole, P.H., Ranasinghe, D.C., eds.: Networked RFID Systems and Lightweight Cryptography Raising Barriers to Product Counterfeiting. Springer (2007)

11. Rukhin et al: A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST Special Publication 800-22 (revised May 15 2002) (2002)
12. Sample, A.P., Yeager, D.J., Powledge, P.S., Smith, J.R.: Design of a passively-powered, programmable platform for UHF RFID systems. In: Proceedings of IEEE International Conference on RFID, 2007. (2007) 149–156
13. Sarpeshkar, R., Delbruck, T., Mead, C.: White noise in MOS transistors and resistors. IEEE Circuits and Devices Magazine **9** (1993) 23–29
14. Smith, J.R., Sample, A., Powledge, P., Roy, S., Mamishev, A.: A wirelessly-powered platform for sensing and computation. In: Proceedings of Ubicomp 2006: 8th International Conference on Ubiquitous Computing. (2006) 495–506
15. Su, Y., Holleman, J., Otis, B.: A 1.6pJ/bit 96% stable chip ID generating circuit using process variations. In: Digest of Technical Papers, 2007 IEEE International Solid-State Circuits Conference. (2007)
16. Suh, G., O'Donnell, C., Sachdev, I., Devadas, S.: Design and implementation of the aegis single-chip secure processor using physical random functions. In: Proceedings of 32nd International Symposium on Computer Architecture. (2005) 25– 36
17. Sunar, B., Martin, W.J., Stinson, D.R.: A provably secure true random number generator with built-in tolerance to active attacks. IEEE Transactions on Computers **58** (2007) 109–119
18. Tang, X., De, V.K., Meindl, J.D.: Intrinsic MOSFET parameter fluctuations due to random dopant placement. IEEE Transactions on Very Large Scale Integration (VLSI) Systems (December 1997) 369–376
19. Texas Instruments: MSP430x11x2 MSP430x12x2 Mixed Signal Microcontroller. (2004) Referenced at http://focus.ti.com/lit/ds/symlink/msp430f1232.pdf.
20. Tokunaga, C., Blaauw, D., Mudge, T.: A true random number generator with a metastability-based quality control. In: Digest of Technical papers, 2007 IEEE International Solid-State Circuits Conference. (2007)
21. Usami, M., Tanabe, H., Sato, A., Sakama, I., Maki, Y., Iwamatsu, T., Ipposhi, T., Inoue, Y.: A 0.05x0.05mm RFID chip with easily scaled-down ID-memory. In: Digest of Technical papers, 2007 IEEE International Solid-State Circuits Conference. (2007)
22. Yuksel, K., Kaps, J.P., Sunar, B.: Universal hash functions for emerging ultra-low-power networks. In: Proceedings of The Communications Networks and Distributed Systems Modeling and Simulation Conference. (2004)