Optimum Power Controller for Random Number Generator in the Crypto Module of Ubiquitous Computing Environment *

Jinkeun Hong[1], Kihong Kim[2], Nohbok Lee[3], Miyoung Kwon[3], Yonghyun Kim[3]

[1] Division of Information & Communication Engineering, Baekseok University
115, Anseo-dong, Cheonan-si, Chungnam, 330-704, South Korea
jkhong@bu.ac.kr

[2] Network & Communication Security Division, ETRI
P. O. Box 1, Yuseong, Daejeon, 305-600, South Korea
hong0612@hanmir.com

[3] Agency for Defense Development
P. O. Box 132, Songpa, Seoul, 138-600, South Korea
nblee@gsm.kaist.edu, kmyadd@paran.com, yonghyun@add.re.kr

Abstract. Critical cryptography applications require the production of an unpredictable and unbiased stream of binary data derived from a fundamental noise mechanism, which is quite difficult to create with a stable random bit stream, as required for statistical randomness, when using a random generator with only a hardware component. However, since all electronic systems are influenced by a finite bandwidth, $1/f$ noise, and other non-random influences, perfect randomness cannot be preserved by any practical system. Thus, when generating random numbers using an electronic circuit, a low-power white noise signal is amplified, then sampled at a constant sampling frequency. Yet, it is quite difficult to create an unbiased and stable random bit stream, as required for statistical randomness, when using a random generator with only a hardware component and in especially it has occur the drift phenomena of input power. Therefore if the randomness of output bit stream is beyond limits range, it is applied the regulation of input power range to take the output bit stream, through the evaluation of randomness by constant period of output bit stream. Accordingly, this paper proposes a method for stabilizing the input power of a random number generator using optimum power control mechanism in crypto module hardware. As such, the proposed scheme is designed to reduce the statistical property of a biased bit stream and optimize the input power to a random number generator engine in crypto module engine for ubiquitous computing.

1 Introduction

In recent years, ubiquitous computing advocates the construction of massively distributed computing environments that consumer electronics, sensors, global

* This paper is sponsored by the ADD project of DUTC (Dual Use Technology Center), Korea.

positioning system (GPS) receives. Bluetooth originally thought of as a “serial cable replacement” for small computer peripherals, and 802.11, originally developed as a wireless LAN system for mobile devices (laptop, PDA) [1] [2] [3]. In this environment, ubiquitous computing imposes peculiar constraints computational power and energy budget, which make this case significantly different from those contemplated by the canonical doctrine of security in distributed systems. There are many security issues in the ubiquitous environment, including authentication, authorization, accessibility, confidentiality, integrity, and non repudiation. And other issues include convenience, speed, and so on. A H/W random number generator uses a non-deterministic source to produce randomness, and more demanding random number applications, such as cryptography, smart card crypto engine, and statistical simulation, benefit from sequences produced by a random number generator, a cryptographic system based on a hardware component [1]. As such, a number generator is a source of unpredictable, irreproducible, and statistically random stream sequences, and a popular method for generating random numbers using a natural phenomenon is the electronic amplification and sampling of a thermal or Gaussian noise signal.

However, since all electronic systems are influenced by a finite bandwidth, $1/f$ noise, and other non-random influences, perfect randomness cannot be preserved by any practical system. Thus, when generating random numbers using an electronic circuit, a low-power white noise signal is amplified, then sampled at a constant sampling frequency. Yet, it is quite difficult to create an unbiased and stable random bit stream, as required for statistical randomness, when using a random generator with only a hardware component. The studies reported in [2] [3] [4] show that the randomness of a random stream can be enhanced when combining a real random number generator, LFSR number generator, and hash function. However, the randomness of this combined method is still dependent on the security level of the hash function and LFSR number generator.

Therefore, controlling a stable input voltage for a random number generator is an important aspect of the design of a random number generator. In previous studies, Peiris and Annakkage examined the use of logic modulation for damping power system oscillations [5], while Zang and Phillis proposed the use of logic to solve the admission control problem in two simple series paralleled networks [6]. Plus, logic has also been applied to admission control in communication networks [8]. If it is occurred the transition of input power due to circumstance effects, temperature, transition of time, it is not guaranteed the stable output bit stream and the randomness of randomness number generator output bit stream is not guaranteed. Therefore when it is occurred the drift of input power deviation, it is needed to design the mechanism, which can be guaranteed the randomness of output bit stream. Accordingly, this paper proposes a optimum power approach to ensuring a stable input power for a random number generator engine. The stability of the input power is a very important factor in the randomness of a random number generator engine. Thus, to consistently guarantee the randomness of an output sequence from a random number generator, the origin must be stabilized, regardless of any change of circumstance elements. Therefore, a random number generator is proposed that applies power logic control, thereby providing the best input power supply. Additionally we use measure of randomness test to decide DB base and its measure is provided the efficiency, which is fast and not weighty due to use test bits of 200,000bits, when it is evaluated the randomness of output stream.

Hereinafter, section 2 reviews the framework of power logic control. Then, section 3 examines a case study, experimental results and some final conclusions are given in section 4.

2 Framework of Optimum Power Controller (OPC) in Crypto Module

Most crypto module microcomputer chips are consists of CPU, ROM, RAM, I/O, EEPROM, etc. The ROM contains the chip operating system and the RAM is the process's working memory.

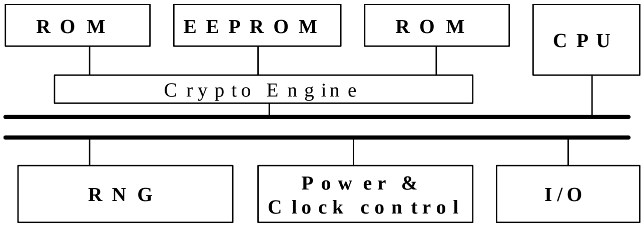

Fig. 1. Microcomputer architecture of crypto module

In the EEPROM memory, data and program can be written to and read from the EEPROM under the control of OS. Within the card, data are passed through a bus under the security logic's control. Crypto module has some form of power and clock control circuitry, BUS, and I/O interface.

The H/W random number generator includes common components for producing random bit streams, classified as follows: characteristics of the noise source, amplification of the noise source, and sampling for gathering the comparator output [10] [11]. The applied noise source uses Gaussian noise, which typically results from the flow of electrons through a highly charged field, such as a semiconductor junction [12] [13] [14] [15].

Ultimately, the electron flow is the movement of discrete charges, and the mean flow rate is surrounded by a distribution related to the launch time and momentum of the individual charge carriers entering the charged field. The Gaussian noise generated in a PN junction has the same mathematical form as that of a temperature-limited vacuum diode. The noise seems to be generated by the noise current generator in parallel with the dynamic resistance of the diode. The probability density $f(x)$ of the Gaussian noise voltage distribution function is defined by Eq. (1).

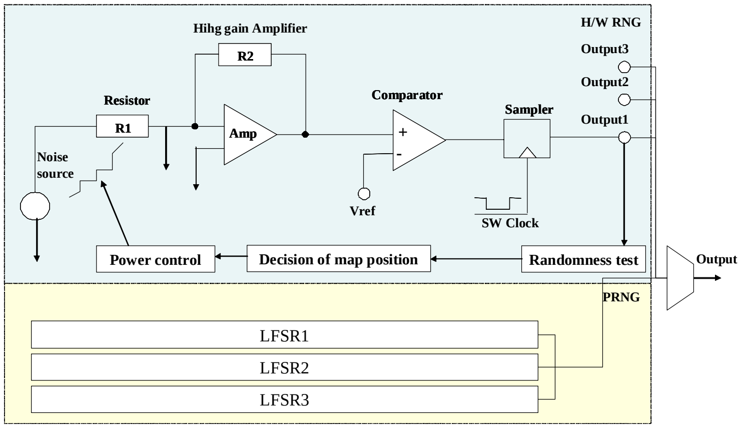

Fig. 2. RNG (H/W RNG & PRNG) module architecture

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{x^2}{2\sigma^2}} \tag{1}$$

Here, σ is the root mean square value of Gaussian noise voltage. However, for designed Gaussian noise random number generator, the noise diode is used the diode with white Gaussian distribution. The power density for noise is constant with frequency from 0.1Hz to 10MHz and the amplitude has a Gaussian distribution. $Vn(rms)$ is the rms value of noise standard deviation of distribution function. The noise must be amplified to a level where it can be accurately threshold with no bias by a clocked comparator. Although the rms value for noise is well defined, the instantaneous amplitude of noise has a Gaussian normal distribution.

$$V_n(rms) = \sqrt{4kTRB} \tag{2}$$

Here, k is Boltzmann constant $(1.38 \times 10^{-23} J/deg.K)$, T is absolute temperature (deg. Kelvin), B is noise bandwidth (Hz), R is resistor (ohms). If $4kT$ is 1.66×10^{20} and R is $1K$, B is $1Hz$, then $V_n(rms) = \sqrt{4kTRB} = 4nV/\sqrt{Hz}$. The applied voltage is $15Vdc$, and current limiting resistor is $16k\mathit{\Omega}$. Noise comes from agitation of electrons within a resistance, and it sets a lower limit on the noise present in a circuit. When the frequency range is given, the voltage of noise is decided by a factor of frequency. The crest factor of a waveform is defined as the ratio of the peak to the rms value. A crest value of approximately 4 is used for noise.

However, for the proposed real random number generator, the noise diode is a noise diode with a white Gaussian distribution. The noise must be amplified to a level where it can be accurately thresholded with no bias using a clocked comparator.

This section provides a short description of the framework of a FLC [5] [6] [7] [8] as follows: the input power source (1), generate engine that generates random numbers (2), random test process (3), decision of voltage map position (4), DB map table (5), and regulation of power control (6). The proposed optimum power control framework is consists of three components, such as decision of map position, and management of voltage map table.

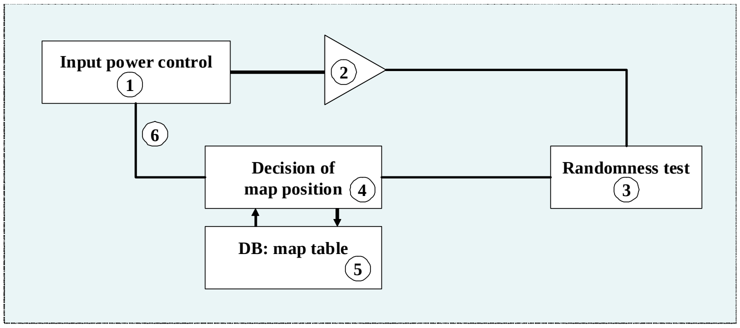

Fig. 3. Optimum power control framework used to generate random numbers

- *Generate engine that generates random numbers and randomness test block*: A generating engine that generates random numbers includes common components for producing random bit streams. It can be characterized as encompassing the following: A Gaussian noise process, a source amplification process, and a sampling process [10] [11]. The cryptographic modules that implement a random number generator engine also incorporate the capability to perform statistical tests for randomness.
- *Decision of map position, management of voltage map table*: To set up the position of a voltage map, a map DB is managed, and the parameters in the map DB consist of the current VP value, the LST VP, the DVP, the lower bound value, and the upper bound value, as shown in Table 1.

Table 1. Voltage map table to decide voltage position

Parameter	Current VP	LST VP	DVP	Lower bound	Upper bound
Voltage	V_c	V_{lst}	+ or -	V_D	V_U

Here, VP is voltage point value, $LST\ VP$ is the last voltage point value, and DVP is the direction value of the voltage point. The current VP is set at V_c, and the decision voltage value, $LST\ VP$, is set at V_{lst} after the test evaluation of the last randomness of the output bit stream. If it is increased to the value of $LST\ VP$, then the value of DVP is positive, and the last decision value V_{lst} is increased in reference of the current $VP\ V_c$, as shown in Eq. (3) and Fig. 4: Here, ΔV is an acceptable level of voltage for voltage regulation .

$$V_{lst} = V_c + (DVP)\Delta V \tag{3}$$

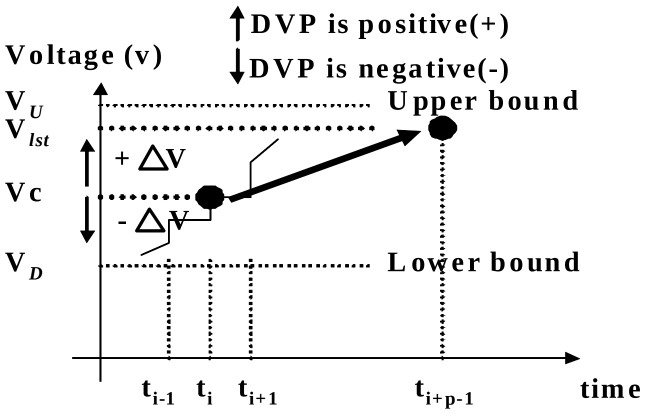

Fig. 4. Optimum power control setting process of output bit stream

In Fig. 4, to decide at the point of the optimum time controlled interval, the threshold level is set up and Vc at the current time point t_i, V_{lst} at the next time point t_{i+p-1} results. In Fig. 5, when the randomness of the output bit stream is evaluated, if it is found to deviate from the threshold level of randomness, then it is considered as a failed region during the period of time interval T_p. The value of the count is summed, and if it is more than that of threshold level, the optimum power control can be operated.

$$E_p = \{E_{i-1},\ E_i,\ E_{i+1},\ ...,\ E_{i+p-1}\} \tag{4}$$

Here, i is $1, 2, 3, ..., n$, and E is the result of the randomness evaluation test during the each period time (T_p).

$$T_p = \{t_{i-1},\ t_i,\ t_{i+1},\ ...,\ t_{i+p-1}\} \tag{5}$$

In addition, δ is a decision factor; it is also a threshold level and reference condition that is used for verifying the success rate of the randomness factor.

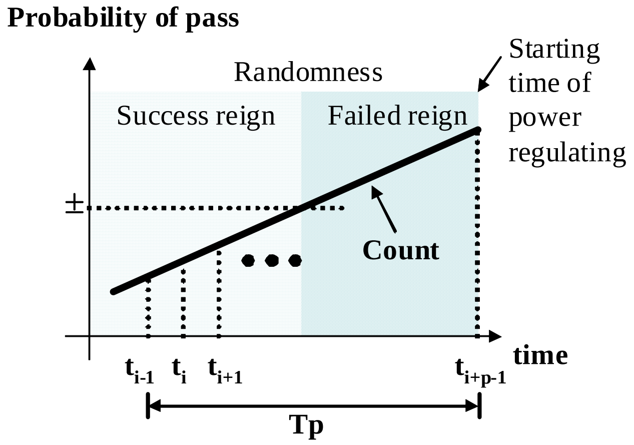

Fig. 5. Setting up of optimum power control time

The OPC algorithm evaluates the randomness test for the output bit stream after the interval of the period T_p. When the iteration result of the randomness evaluation is greater than the value of the threshold level, then the regulation of the input power level is determined and the optimum power control process is controlled as follows:

Each of the following random number tests is implemented to test a sequence length of 200,000bits. In Fig. 6, the frequency test determines whether the number of ones and zeros in a sequence approximate the number expected for a truly random sequence.

The upper bound value of the threshold level is not greater than 3.841. The serial test is the frequency of each and every overlapping m bit pattern; this is used to compare the frequency of overlapping blocks of two consecutive/adjacent lengths (m and $m+1$) against the expected result for a random sequence. In this case, the value of the threshold level is under the outer bound of 5.991. A poker test is used to divide the sequence into k non-overlapping m length sequences. The value of the threshold level is under 14.067 for a length of 3. These sequences are compared to a 2^m space, and with each match the value total increases. An autocorrelation test checks for the correlation between the current sequence and the shifted sequence. The value of the threshold level is under 0.05.

3 Experimental results

The decision of the optimum power map position is converted by a value based on a DB map table. When the input power remains within the border area, the output random number sequence maintains stable randomness. When five levels of input power are given, the randomness of the output random number sequences is as shown in Table 2.

```
Algorithm: process of optimum power control
Optimum_Controller() ::

1. Let threshold level of randomness $\gamma$ ;
2. Given RNGSequence size: $w = i \times 200000, i = \{0, ..., n\}$;
3. for $i = \{0, ..., n\}$ times do
4.    Result = EvaluationTest(w);
5.    if (Result == 'False') then count++;
      if (count > $\delta$) then Regulate_Power_Control();
6. End for

Evaluation_Randomness(width) ::

1. width {
2. If $\|D\| <= \delta$, then D[width] is PassBitStream, SaveBit Stream=D[width], return
   'True';
3. Else, D[width] is discarded, return 'False'; }

Regulate_Power_Control()::

1. DB_map_check: $V_c$, $V_lst$, $V_U$, $V_D$, $DVP$;
2. Check randomness of controller output stream after regulation voltage according
   to DB map value;
3. Decision of $V_lst$ to the direction value DVP(+: increased direction, -: decreased
   direction);
```

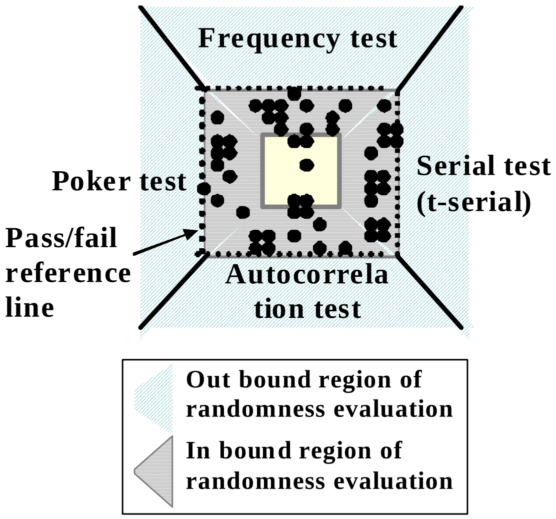

Fig. 6. Bound lines in the randomness evaluation of output bit streams

Table 2. Relationship between the result of randomness test and input power

Voltage	9.65V	9.83V	10.0V	10.15V	10.3V
Pocker test (block=4)	7.8	12.1	13.0	21.5	15.7
(X < 24.9)	PASS	PASS	PASS	PASS	PASS
Pocker test (block=5)	28.6	41.2	30.5	44.0	24.1
(X < 44.7)	PASS	PASS	PASS	PASS	PASS
t-serial test (block=4)	4.1	5.0	11.4	4.8	4.8
(X < 15.5)	PASS	PASS	PASS	PASS	PASS
t-serial test (block=5)	16.3	23.3	18.9	26.4	22.5
(X < 26.3)	PASS	PASS	PASS	PASS	PASS

The randomness of the output random number sequence reacted sensitively whenever the input power supply was changed. Therefore, the experimental model was shown to highlight the relationship between the randomness and variations in the input power, where the randomness of the output random number sequences was found to depend on the input power, and a threshold value could be used to determine the randomness of the output random number sequence engine. Therefore, modifications in the input power controlled by the proposed OPC were used to stabilize this interdependence between the input power and the randomness of the output random number sequences. In Table 3, the initial input power was set between 9.6V and 10.4V, and the result of the randomness evaluation passes in a given time interval. After a lapse of a specific time, due to the drift of the surrounding conditions, such as drift of the input power level or in the specific circuitry, the randomness security level of output bit stream is not always guaranteed in the case of a generally stable designed input power range condition.

Table 3. The pass/fail condition according to tolerant input power range in general case

Voltage	$V_{lst} < 9.6V$	$9.6V < V_{lst} < 10.4$	$V_{lst} > 10.4$
Frequency test	Fail	Pass	Fail
Serial test	Fail	Pass	Fail
t-serial test	Fail	Pass	Fail
Pocker test	Fail	Pass	Fail
Autocorrelation test	Fail	Pass	Fail

If the result of the randomness evaluation passes at a level of 90% with a generated random number speed 200kbps and a 1 day T_p value, the number of collected bit streams that pass the randomness evaluation is 1.56×10^{10} bits. Additionally, the number of discarded bit streams that fail is 1.73×10^{9} bits. The condition of T_p is set at 1 hour with the OPC; the number of collected bit streams is 6.48×10^8 bits and the number of discarded bit streams is 7.2×10^7 bits.

Table 4. Pass/fail bits in condition of 200kbps/2Mbps (pass rate = 90%)

T_p	200kbps Pass bits	200kbps Fail bits	2Mbps Pass bits	2Mbps Fail bits
1sec	1.8×10^5	2.0×10^4	1.8×10^6	2.0×10^5
10min	1.08×10^8	1.2×10^7	1.08×10^9	1.2×10^8
1hrs	6.48×10^8	7.2×10^7	6.48×10^9	7.2×10^8
1day	1.56×10^{10}	1.73×10^9	1.56×10^{11}	1.73×10^{10}

Otherwise if the OPC is not applied, the output bit stream of the random number generator cannot guarantee randomness. In Table 5, although the pass probability is degraded at 80% due to the state of random number generator, if the RNG is applied with the OPC, a guaranteed 1.56×10^{10} bits will be determined, which is at least 90%. If the OPC is not applied, the result is then a guaranteed 1.38×10^{10} bits, approximately. If the management of the random number generator is neglected, the pass probability of the output bit stream is degraded, and the security characteristics and stability of the random number generator can no longer be guaranteed.

Table 5. Pass/fail bits in condition of 200kbps (pass rate = 80%)

T_p	Without OPC Pass bits	Without OPC Fail bits	With OPC Pass bits	With OPC Fail bits
1sec	1.6×10^5	4.0×10^4	1.8×10^5	2.0×10^4
10min	9.6×10^7	2.4×10^7	1.08×10^8	1.2×10^7
1hrs	5.76×10^8	1.44×10^8	6.48×10^8	7.2×10^7
1day	1.38×10^{10}	3.46×10^9	1.56×10^{10}	1.73×10^9

In Fig. 7, the collected bits are compared with the OPC and without OPC. Here, the randomness level is satisfied, in terms of a variable T_p, such as a variable pass probability.

If the T_p value is 1 day, although the degradation of the random number generator occurs, indicating that the pass probability has been degraded, and if the OPC is applied, it can be guaranteed that a stable and secure output bit stream will function continuously. Otherwise, if the OPC is not adopted, as in the degradation state of the random number generator, the collected number of guaranteed bit streams is reduced, which satisfies the randomness condition. Moreover, if the period of the test interval T_p becomes short enough, the time

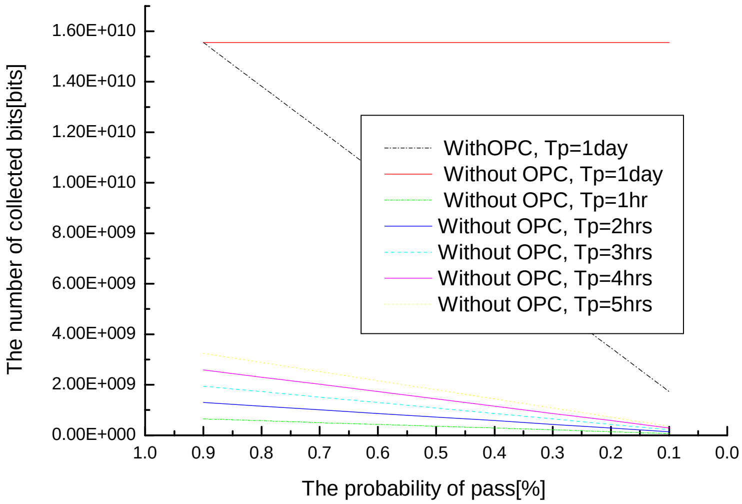

Fig. 7. Bound lines in the randomness evaluation of output bit streams

consumed during the test of randomness is enhanced and the period for the detection rate is short. Otherwise, if the period of the test interval is relatively long, the time consumed for the test of the randomness is reduced. However, if the state of the random number generator fails due to the drift of input power, the generated bits stream during the interval must be discarded. Therefore, it is necessary to study the optimum power control in addition to the related period.

4 Conclusion

In ubiquitous computing, a smart card consists of a chip and an integral operating system. The chip contains the CPU, ROM, RAM, I/O functions, and the EEPROM. Some smart card microprocessors use a RNG and cryptographic processors. An optimum power controller was proposed and applied to the input power of a random number generator engine in crypto-processor of crypto module. A random number generator uses a non-deterministic source to produce randomness, and more demanding random number applications, such as cryptography and statistical simulation, benefit from sequences produced by a random number generator, a cryptographic system based on a hardware component in a smart card. Nevertheless, the stability of the input power is very important in ensuring the randomness of a random number generator engine. Therefore, to guarantee the randomness of the output sequences from a random number generator consistently, a method that can stabilize the origin quickly, regardless of any changes in the circumstance elements, is presented. Tests showed hat the proposed optimum power controller using a length of 200,000bits is effective and rapid in stabilizing the input power of a random number generator engine in a crypto module.

References

1. H. Alireza and V. Ingrid, "High-Throughput Programmable Crypto-Coprocessor," *IEEE Computer Society*, 2004.
2. A. M. Jalal, R. Anand, C. Roy, and M. D. Dept, "Cerberus: A Context-Aware Security Scheme for Smart Spaces," *IEEE PerCom'03*, 2003.
3. N. O. Attoh-Okine and L. D. Shen, "Security Issues of Emerging Smart Cards Fare Collection Application in Mass Transit," 1995.
4. WiTness: Interaction of SIM based WiTness Security Functions and Security Properties of Mobile Devices and Communication Channels, *Information society*, 2003
5. H. J. C. Peiris, U. D. Annakkage, and N. C. Pahalawaththa, "Generation of Fuzzy Rules to Develop Fuzzy Logic Modulation Controllers for Damping of Power System Oscillations," *IEEE Trans. on Power System*, Vol.14, No.4, 1999.
6. R. Zang and Y. A. Phillis, "Admission Control and Scheduling in Simple Series Paralleled Networks Using Fuzzy Logic," *IEEE Trans. on Fuzzy Systems*, Vol.9, No.2, 2001.
7. George J. Klir and Bo Yuan, *Fuzzy Sets and Fuzzy Logic Theory and Applications*, Prentice-Hall International Inc., 1995.
8. Q. Le and G. M. Knapp, "Incorporating Fuzzy Logic Admission Control in Simulation Models," *Winter Simulation Conference*, 2003.
9. M. Kimberley, "Comparison of Two Statistical Tests for Keystream Sequences," *IEE Electronics Letters*, Vol. 23, No. 8, 1987.
10. C. S. Petrie and J. A. Connelly, "A Noise-Based Random Bit Generator IC for Applications in Cryptography," *ISCAS'98*, 1998.
11. M. Delgado-Restituto, F. Medeiro, and A. Rodriguez-Vasquez, "Nonlinear Switched-Current CMOS IC for Random Signal Generation," *IEE Electronic Letters*, Vol. 29, 1993.
12. http://www.io.com/∼ritter/RES/NOISE.HTM.
13. http://www.clark.net/pub/cme/P1363/ranno.html.
14. http://webnz.com/robert/true_rng.html.
15. Boris Ya, Ryabko, and E. Matchikina, "Fast and Efficient Construction of an Unbiased Random Sequence," *IEEE Trans. on Information Theory*, Vol. 46, No. 3, 2000.
16. W. Timothy Holman, J. Alvin Connelly, and Ahmad B. Dowlatabadi, "An Integrated Analog/Digital Random Noise Source," *IEEE Trans. on Circuits and System I: Fundamental Theory and Applications*, Vol.44, No.6, 1997.
17. FIPS 140-1: Security Requirements for Cryptographic Modules, Federal Information Processing Standards Publication 140-1, *U.S. Department of Commerce/NIST[National Technical Information Service]*, 1994.
18. http://csrc.ncsl.nist.gov/fips/fips 1401.htm.
19. http://stat.fsu.edu/∼ geo/diehard.html.