An Oscillator-Based True Random Number Generator with Jitter Amplifier

Takehiko Amaki, Masanori Hashimoto and Takao Onoye
Graduate School of Information Science and Technology, Osaka University   JST CREST
1-5 Yamadaoka, Suita, Osaka, 565-0871 Japan
Email: {amaki.takehiko, hasimoto, onoye}@ist.osaka-u.ac.jp

***Abstract*— This paper presents an oscillator-based TRNG (true random number generator) with jitter amplifier. The proposed jitter amplifier fabricated in a 65nm CMOS process archives 8.4x gain at $25\ ^\circ\mathrm{C}$, and significantly improves randomness of output bitstream. The TRNG with the jitter amplifier enhances throughput per area by 94 % compared to a TRNG with frequency dividers. The prototype TRNG occupies 6,300 μm^2, generates 2 Mbps random bitstreams, and passes FIPS 140-2 randomness tests and 12 tests in NIST test suite.**

I. Introduction

High-quality random number generation is essentially demanded for security. True random numbers are produced from physical random sources. Each bit of the bitstreams is independent from the other bits and the probabilities of 1/0 occurrences are identical. Because true random numbers cannot be predicted by computational methods, they are highly desirable for security purposes. For example, they are used as keys and initial vectors of CBC (cipher block chaining) mode in a common key cryptosystem. Challenge-response authentication also requires true random numbers as challenges.

Although there are many kinds of physical random phenomena in the nature, a limited number of phenomena are usable for on-chip TRNG since special hardware and materials (e.g. [1][2]) are mostly unacceptable due to their cost. Therefore, TRNGs which use internal random noises are popular and widely studied.

Oscillator-based TRNG[3], which utilizes random period jitter of oscillators as random source, is one of the popular circuits for generating truly random numbers. Though the oscillator-based TRNG can be easily implemented with CMOS gates or FPGA[4], the amount of the internal noises, that is, the jitter of the oscillators is so small that highly random bitstreams cannot be generated. A long inverter chain or a frequency divider provides the sufficient jitter at the enormous sacrifice of area or throughput, which will be discussed in Sec. V. Therefore, the TRNG is often accompanied by a post-processor that consumes additional area and power dissipation. Even with the post-processor, the randomness of the TRNG is still a concern for high-quality random number generation.

This paper proposes an oscillator-based TRNG with jitter amplifier. Test chip measurements demonstrate that the proposed jitter amplifier improves the randomness with small increase in area without throughput degradation.

II. Oscillator-based TRNG on silicon

Figure 1 illustrates the structure and the operation of the proposed TRNG. Two oscillators, one of which is fast and the other of which is slow, are exploited, and the fast oscillating signal (D in Fig. 1) is sampled with the jittery slow clock whose jitter is amplified with the jitter amplifier (CK in Fig. 1). An output being determined by a time when the clock rises, the randomly fluctuating rise edges of the slow signal result in a random bitstream.

Figure 2 shows χ of poker test[5] and ApEn (approximate entropy)[6] of simulated random bitstreams when the jitter of slow signal is amplified[7]. The periods of the fast and the slow oscillators are 220 ps and 7.3 ns, and the jitter of the fast oscillator is set to zero for simplification. Lower χ and higher ApEn mean better randomness. We can see that increase in the jitter of the slow oscillator reduces χ and enlarges ApEn, which means that the jitter amplification of the slow oscillator is expected to improve the quality of random bitstream. It should be noted that the randomness monotonously improves as the jitter of the slow clock increases in Fig. 2, and hence the gain magnitude of the jitter amplifier gain is important yet its stability is not demanded.

In this paper, jitter is random period jitter of an oscillating signal, originating from internal random noises such as thermal noise, shot noise, random telegraph noise, and it is defined as the standard deviation of periods.

III. Jitter Amplifier

A. Concept of Jitter Amplification

Figure 3 illustrates the concept of jitter amplification. Here, the jitter of the input oscillating signal in is amplified. The right figure explains the operation of Latency-Controllable buffer (LC buffer), which is the main part of jitter amplifier. Each element delay of the LC buffer can be enlarged by changing an operation mode from fast mode to slow mode with cont signal. A rising edge of cont is given while in is propagating through the buffer. Here, the rise timings of cont are independent from in.

Figure 4 exemplifies latencies of the normal buffer and the LC buffer as a function of t_{in}, which is rise timing of in.

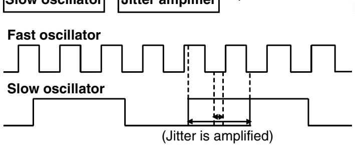

Fig. 1. Proposed oscillator-based TRNG with jitter amplifier.

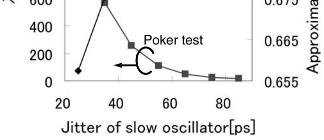

Fig. 2. Randomness vs. jitter.

978-1-4244-9472-9/11/$26.00 ©2011 IEEE

t_{cont}, which is the rise timing of cont, is also appended to the normal buffer for comparison in Figs. 3 and 4. The latency of the LC buffer gets longer as t_{in} increases as long as t_{in} is in the appropriate range colored gray in Fig. 4, because the time period of fast mode decreases as t_{in} increases, whereas the latency of the normal buffer is constant. This means that the later rising edge of in causes the larger latency of the LC buffer. Therefore, the time interval between the early and the late rise timings at in is intensified at out by the variable latency of the LC buffer. Thus, the jitter of in is amplified.

B. Implementation

Figure 5 depicts the implemented jitter amplifier which consists of LC buffer and timing generator. When in rises, a ring oscillator is enabled (en_e/o) and its corresponding counter starts increment after initialization. Every time the counter value (cnt_e/o) exceeds predefined values, a pulse generator produces rise and fall edges. For generating cont for every in rise edge, the timing generator has two (even and odd) paths because the increment of the counter starts at a rising edge of in and ends after the next rising edge. The LC buffer is designed so that each buffer delay t_d can be changed by cont from t_{df} to t_{ds}, where $t_{df} < t_{ds}$. That is, the buffer operates in fast mode until cont rise edge arrives, and after that it works in slow mode. To change t_d, VDD of the buffer (VDBUF) is varied from high voltage (VDBUFH) and low (VDBUFL) by PMOS switches according to cont.

Figure 6 shows the behavior of the signals in jitter amplification process. A rise edge of in is delayed awhile (in this figure, for about one cycle) by the ring oscillators and the counters in the timing generator and appears as a rise edge of cont just while the next rise edge of in is propagating through the LC buffer. The next rise edge of in is jittered in the slow oscillator while the rise edge of cont is jittered in the timing generator.

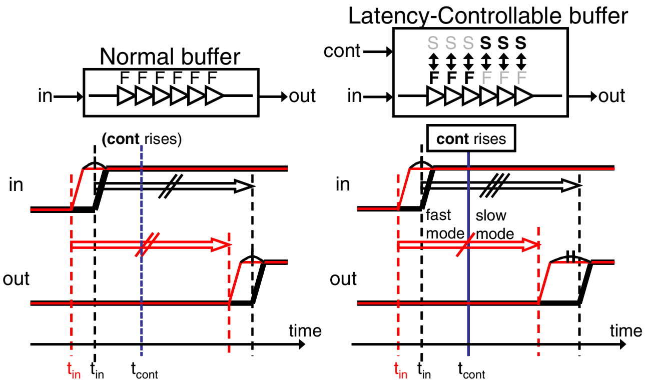

Fig. 3. Concept of jitter amplification.

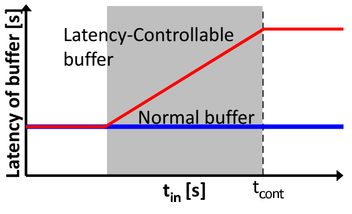

Fig. 4. Latencies of normal buffer and Latency-Controllable buffer.

Therefore, the jittery rise timings of cont are independent from those of in. Consequently, when the rise edge of in is late relatively to that of cont, the buffer operates longer in fast mode and out edge arises earlier. From the opposite point of view, when cont rise edge is early, slow mode operation becomes longer and out edge arrives later. This namely means the timing fluctuation of in rise edge, strictly speaking the time interval fluctuation between successive two rise edges of in, is amplified.

C. Gain of Jitter Amplification

The total latency from in to out, T_{BUF}, is given by

$$T_{BUF} = \left(T_{st} - T_{lag}\right) + \left(N - \frac{T_{st} - T_{lag}}{t_{df}}\right) t_{ds}, \quad (1)$$

where N is the number of the buffer element stages, T_{st} is the ideal time of fast mode without any timing fluctuation, and T_{lag} is the fluctuation of time interval between in and cont rise edges. Therefore, $T_{st} - T_{lag}$ is the actual time during fast mode, and the second term of Eq. (1) is the time of slow mode. From Eq. (1), jitter gain Gain $= \{\sqrt{\sigma(T_{BUF})^2 + \sigma_{in}^2}\}/\sigma_{in}$ is:

$$\begin{aligned} \text{Gain} &= \left\{\sqrt{\left\{-\sigma_{lag}\left(1 - \frac{t_{ds}}{t_{df}}\right)\right\}^2 + \sigma_{in}^2}\right\} \frac{1}{\sigma_{in}}, \\ &= \sqrt{a^2\left(1 - \frac{t_{ds}}{t_{df}}\right)^2 + 1}, \quad (2) \end{aligned}$$

where σ_{lag} is the standard deviation of T_{lag}, and σ_{in} is the standard deviation of the time interval between successive rise edges of in. a is the ratio of $(\sigma_{lag}/\sigma_{in})$ and is expressed as $(a = \sqrt{1 + \sigma_{tg}^2/\sigma_{in}^2})$, where σ_{tg} is the standard deviation of the delay time from rise edge of in to corresponding rise edge

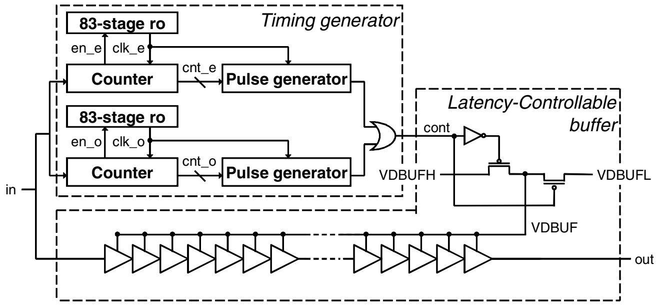

Fig. 5. Implemented jitter amplifier.

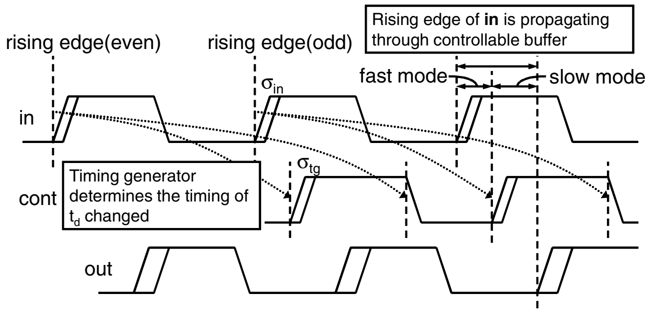
rising edge(even)
rising edge(odd)
Rising edge of in is propagating through controllable buffer
fast mode
slow mode
σ_{in}
in
σ_{tg}
Timing generator determines the timing of t_d changed
cont
out

Fig. 6. Behavior of jitter amplifier.

of cont. Using σ_{in} and σ_{tg}, σ_{lag} is expressed as: $\sigma_{lag}^2 = \sigma_{in}^2 + \sigma_{tg}^2$. Eq. (2) indicates that the gain is determined by t_{ds}/t_{df} and mostly independent of other design parameters.

IV. Experiment

The proposed circuit was fabricated in 65nm CMOS (Fig. 7) and measured using a real-time oscilloscope and a logic analyzer. 31-stage ring oscillator and 251-stage ring oscillator with 64- frequency divider are fast and slow oscillators. Interaction between the oscillators can be ignored because the VDD for the fast oscillator is independent from the other circuits. The jitter amplifier (timing generator and LC buffer) occupies 3,300 μm^2 and the total area of the TRNG macro-cell is 6,300 μm^2. Figure 8 shows the amplified jitter of the slow oscillator at 25 °C when the voltage difference (VDBUFH-VDBUFL) is changed by decreasing VDBUFL while keeping VDBUFH 1.2V. As the voltage difference becomes larger, the jitter increases. Figure 9 shows the jitter gain at various temperatures. $a = 1.9$ is used for gain estimation with Eq. (2). The proposed circuit achieves 8.4x gain at 25 °C. It can be seen that the gain increases as the temperature decreases, because the sensitivity of t_d to supply voltage becomes larger at lower temperature. As described in Sec. II, the instability of jitter gain, here in terms of the temperature, is not a serious problem as long as the gain is larger than the requirement. The gain estimation of Eq. (2) is also plotted. The estimation reproduced the similar tendency to the measurement result.

Figure 10 shows the randomness evaluation with poker test and ApEn, where the duty cycle of the fast oscillator is adjusted to within 50±0.3 % by body biasing, VDBUFH is fix to 1.2 V and the temperature is 25 °C. In order to clarify the effect of jitter amplifier, the VDD for the fast oscillator is lowered to 0.9 V while the VDD for the slow oscillator and the sampler is 1.2 V. 1,000 streams of 20k bits and 128 streams of 2^{18} bits were measured and used for poker test and ApEn calculation, respectively. χ becomes smaller and ApEn gets larger, i.e. the randomness improves as the voltage difference increases. Table I lists the results of FIPS140-2 and NIST randomness tests, where the duty cycle of the fast oscillator was adjusted to within 50±0.2 %, VDBUFH and VDBUFL are 1.2 V and 0.7 V, the VDD for the fast oscillator is 0.9 V, and the temperature is 25 °C. Note that FFT test was not executed because a problem of FFT test is reported by NIST. With this configuration, 2 Mbps of throughput is achieved. The TRNG without jitter amplification failed 10 tests of NIST test suite. On the other hand, the number of passed tests in NIST test suite increases from 4 to 12 thanks to the jitter amplifier.

Figure 11 shows the comparison of throughput and area with previously reported TRNGs. Points at the upper left mean high throughput per area. The proposed TRNG with jitter amplification attains high throughput per area in addition to high randomness.

V. Discussion

We discuss the efficiency of jitter amplification using throughput per area as a metric for comparing three jitter enlargement techniques. Here, the fluctuations of inverter delays (σ_{inv}) are assumed to be independent from each other. The jitter of rising timings of slow oscillator from the viewpoint of rising edges of fast oscillator is defined as equivalent jitter. Assuming that the jitter of two oscillators are independent from each other, the equivalent jitter is calculated as the square root of the variance of slow periods added by the variance of

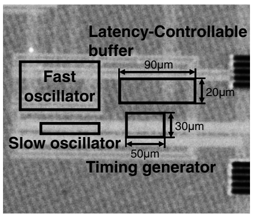

Fig. 7. Chip photo.

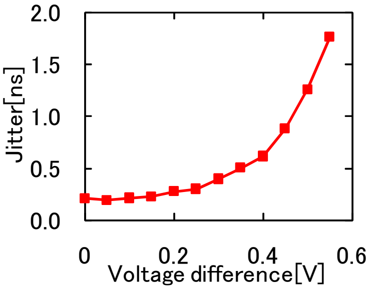

Fig. 8. Jitter vs. voltage difference

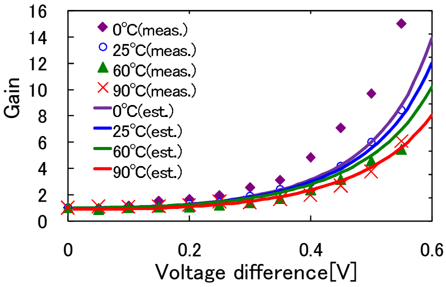

Fig. 9. Gain vs. voltage difference at various temperatures.

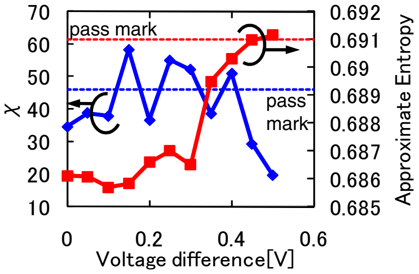

Fig. 10. Randomness evaluation.

TABLE I

FIPS140-2 and NIST randomness test results.

Randomness Tests	Jitter amp.		Improved?
	Without	With	
FIPS140-2 randomness tests (4 tests)			
Monobit Poker Runs Longruns	PASS	PASS	-
NIST test suite (14 tests)			
Frequency CumulativeSums Rank RandomExcursionsVariant	PASS	PASS	-
BlockFrequency LongestRun NonOverlappingTemplate OverlappingTemplate Universal ApproximateEntropy RandomExcursions Serial	FAIL	PASS	YES
Runs LinearComplexity	FAIL	FAIL	-

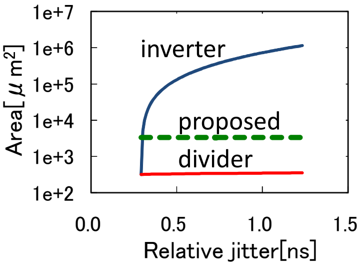

Fig. 12. Area vs. required equivalent jitter.

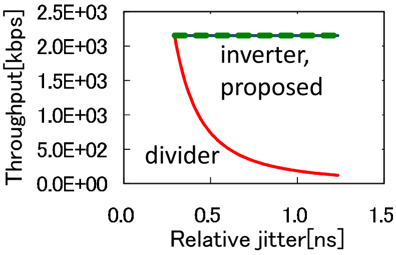

Fig. 13. Throughput vs. required equivalent jitter.

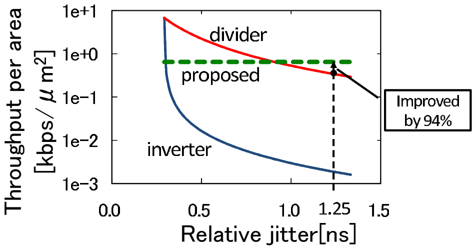

Fig. 14. Throughput per area vs. required equivalent jitter.

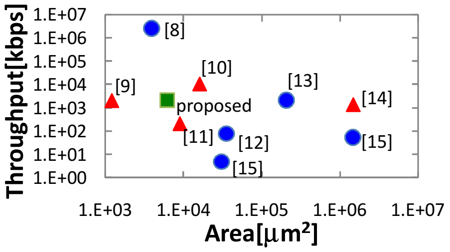

Fig. 11. Comparison with other TRNGs. Circle symbols passes especially severe randomness tests: NIST test suite or DIEHARD[16].

fast periods accumulated during a slow cycle.

As Table I indicates, a TRNG consisting of 31-stage fast oscillator and 251-stage slow oscillator with 64-divider requires larger jitter to pass NIST tests. Three methods for obtaining sufficient jitter are considered: (1) Adding a long inverter chain to the slow oscillator output. $k\sigma_{inv}^2$ is added to the period variance, where k is the number of inverters. (2) Dividing the slow oscillator output with a frequency divider. When $m-$divider is employed, the standard deviation of the period is multiplied by $\sqrt{m}$, whereas the throughput becomes $1/m$ times smaller. (3) Using the proposed jitter amplifier.

Figures 12-14 show the required area, throughput and throughput per area in the case that the required jitter in x-axis is obtained by the three methods. The average periods of the fast and slow oscillators are 930 ps and 460 ns, and the standard deviations are 9.2 ps and 210 ps, respectively. Figure 12 demonstrates that a TRNG with long inverter chain ("inverter") requires larger area as the required jitter grows while the area of a TRNG with dividers ("divider") and with the proposed jitter amplifier ("proposed") are insensitive to the necessary jitter values. Figure 13 shows that the throughput of "divider" sharply decreases as the required value of jitter rises. On the other hand, the throughputs of "inverter" and "proposed" are constant. Figure 14 shows that the throughput per area of "proposed" is independent from the required jitter gain[1]. This means that the proposed jitter amplifier is suitable when large equivalent jitter is required. Here, let us assume 1.25 ns of equivalent jitter is necessary, which is the equivalent jitter value needed to obtain the results of Table I. This jitter value is the measurement value when the voltage difference is 0.5 V in Fig. 8 and the average and standard deviation of the periods of the oscillators referred above in this section. In this case, throughput per area with the proposed jitter amplifier is as 1.94 times large as that with divider.

[1] Rigidly speaking, if the required gain is larger than the attainable gain, cascaded jitter amplifiers are necessary and the required area increases.

VI. Conclusion

An oscillator-based TRNG with jitter amplifier is proposed. Thanks to the jitter amplifier, the TRNG fabricated in 65nm process occupying 6,300 μm^2 generates 2 Mbps bitstream which passes FIPS140-2 tests and 12 tests of NIST test suite. The TRNG with the jitter amplifier improves throughput per area by 94% compared to the TRNG using frequency divider.

Acknowledgment

The VLSI chip in this study has been fabricated in the chip fabrication program of VDEC of Univ. Tokyo in collaboration with STARC, e-Shuttle, Inc., and Fujitsu Ltd.

This work is supported by SCOPE program of Ministry of Internal Affairs and Communications.

References

[1] Q. Zhou, et. al, "True random number generator based on mouse movement and chaotic hash function," *information sciences*, pp. 3442–3450, Sep., 2009.

[2] J. F. Dynes, et. al, "A high speed, postprocessing free, quantum random number generator," *applied physics letters*, pp. 031109(1–3), Jul., 2008.

[3] B. Jun and P. Kocher, "The Intel random number generator," cryptography research inc., white paper prepared for Intel corp., Apr., 1999.

[4] K. H. Tsoi, et. al, "High performance physical random number generator," *IET computers & digital techniques*, vol. 1, no. 4, pp. 349–352, 2007.

[5] "Security requirements for cryptographic modules," FIPS pub. 140-2, May, 2001.

[6] "A statistical test suite for the validation of random number generators and pseudorandom number generators for cryptographic applications," NIST, pub. 800-22, May, 2001.

[7] T. Amaki, et. al, "A design procedure for oscillator-based hardware random number generator with stochastic behavior modeling," *WISA*, 2010.

[8] S. Srinivasan, et. al, "2.4GHz 7mW all-digital PVT-variation tolerant true random number generator in 45nm CMOS," *VLSI Circuit*, pp. 203–204, 2010.

[9] M. Matsumoto, et. al, "1200μm^2 physical random-number generators based on SiN MOSFET for secure smart-card application," *ISSCC*, pp. 414–624, 2008.

[10] M. Bucci, et. al, "A high-speed oscillator-based truly random number source for cryptographic applications on a smart card IC," *IEEE transactions on computers*, vol. 52, no. 4, pp. 403–409, Apr., 2003.

[11] R. Brederlow, et. al, "A low-power true random number generator using random telegraph noise of single oxide-traps," *ISSCC*, pp. 1666–1675, Feb., 2006.

[12] C. Tokunaga, et. al, "True random number generator with a metastability-based quality control," *ISSCC*, pp. 404–611, Feb., 2007.

[13] V. Kaenel and T. Takayanagi, "Dual true random number generators for cryptographic applications embedded on a 200 million device dual CPU SoC," *CICC*, pp. 269–272, Sep., 2007.

[14] C. S. Petrie, J. A. Connelly, "A noise-based IC random number generator for applications in cryptography," *IEEE transactions on circuits and systems*, vol. 47, no. 5, pp.615–621, MAY, 2000.

[15] J. Holleman, et. al, "A 2.92 μW hardware random number generator," *ESSCIRC*, pp. 134–137, 2006.

[16] G. Marsaglia, "DIEHARD statistical tests," Florida state university, (http://www.stat.fsu.edu/pub/diehard/).