JKIICE Journal of the Korea Institute of Information and Communication Engineering

한국정보통신학회논문지 Vol. 24, No. 2: 268~275, Feb. 2020

피드백 구조를 갖는 Self-Timed Ring 기반의 경량 TRNG

최준영[1] · 신경욱[2*]

A Self-Timed Ring based Lightweight TRNG with Feedback Structure

Jun-Yeong Choe[1] · Kyung-Wook Shin[2*]

[1]Graduate Student, Department of Electronic Engineering, Kumoh National Institute of Technology, Gumi, Kyungbuk 39177, Korea
[2*]Professor, School of Electronic Engineering, Kumoh National Institute of Technology, Gumi, Kyungbuk 39177, Korea

요 약

정보보안 응용에 적합한 self-timed 링 (ring) 기반 TRNG (true random number generator)의 경량 하드웨어 설계에 관해 기술한다. TRNG의 하드웨어 복잡도를 줄이기 위해 피드백 구조의 엔트로피 추출기를 제안하였으며, 이를 통해 링 스테이지 수를 최소화 하였다. 본 논문의 FSTR-TRNG는 동작 주파수와 엔트로피 추출 회로를 고려하여 링 스테이지 수가 11의 배수가 되도록 결정되었으며, 링 발진기가 등간격 모드로 진동할 수 있도록 토큰 (token)과 버블 (bubble) 개수의 비를 결정하였다. FSTR-TRNG는 FPGA 디바이스에 구현하여 난수 생성 동작을 검증하였다. Spartan-6 FPGA 디바이스에 구현된 FSTR-TRNG로부터 2,000만 비트의 데이터를 추출하여 NIST SP 800-22에 규정된 통계학적 무작위성 테스트를 수행한 결과, 15개의 테스트가 모두 기준을 만족하는 것으로 확인되었다. Spartan-6 FPGA 디바이스로 합성한 FSTR-TRNG는 46 슬라이스로 구현이 되었으며, 180 nm CMOS 표준셀로 합성하는 경우에는 약 2,500 등가 게이트로 구현되었다.

ABSTRACT

A lightweight hardware design of self-timed ring based true random number generator (TRNG) suitable for information security applications is described. To reduce hardware complexity of TRNG, an entropy extractor with feedback structure was proposed, which minimizes the number of ring stages. The number of ring stages of the FSTR-TRNG was determined to be a multiple of eleven, taking into account operating clock frequency and entropy extraction circuit, and the ratio of tokens to bubbles was determined to operate in evenly-spaced mode. The hardware operation of FSTR-TRNG was verified by FPGA implementation. A set of statistical randomness tests defined by NIST 800-22 were performed by extracting 20 million bits of binary sequences generated by FSTR-TRNG, and all of the fifteen test items were found to meet the criteria. The FSTR-TRNG occupied 46 slices of Spartan-6 FPGA device, and it was implemented with about 2,500 gate equivalents (GEs) when synthesized in 180 nm CMOS standard cell library.

키워드 : 순수 난수 발생기, TRNG, 셀프 타임 링, 찰리 효과, 정보보안

Key word : True random number generator, TRNG, Self-timed ring, Charlie effect, Information security

Received 15 November 2019, **Revised** 18 November 2019, **Accepted** 4 December 2019

*** Corresponding Author** Kyung-Wook Shin(E-mail:kwshin@kumoh.ac.kr, Tel:+82-54-478-7427)
Professor, School of Electronic Engineering, Kumoh National Institute of Technology, Gumi, Kyungbuk 39177, Korea

Open Access http://doi.org/10.6109/jkiice.2020.24.2.268 print ISSN: 2234-4772 online ISSN: 2288-4165

Ⅰ. 서 론

정보보안 시스템에서 다양한 용도로 사용되는 난수(random number)는 대칭키 암호 시스템의 비밀키 생성, 공개키 암호 시스템의 개인키와 공개키 생성, 그리고 전자서명과 인증 프로토콜 등에 필수적으로 사용된다 [1]. 난수는 통계적으로 독립적이고 값이 균일하게 분포되며 예측할 수 없는 수를 의미한다. 이러한 난수의 특성은 암호화 시스템에 대한 보안 공격으로부터 내성을 증가시켜준다. 난수를 생성하는 난수 발생기는 크게 나누어 의사 난수 발생기 (pseudo-random number generator; PRNG)와 순수 난수 발생기 (TRNG)로 구분된다. 의사 난수 발생기는 결정론적 알고리듬을 사용하는 난수 발생기로 다양한 알고리듬들이 사용된다. 높은 처리량을 가져 비교적 짧은 길이의 키를 임의의 길이를 가지는 긴 시퀀스로 확장이 가능하다. 그러나 생성되는 난수가 일정한 주기를 가지기 때문에 난수의 예측 불가능성을 충족하지 못한다 [2, 3].

순수 난수 발생기는 물리적인 현상에서 발생하는 무작위성을 이용하는 난수 발생기로 엔트로피 소스, 엔트로피 추출 회로, 후처리 회로로 구성된다. 엔트로피 소스는 회로 내의 열, 샷 잡음 등과 같은 물리적 프로세스에 존재하는 무작위성을 이용하여 예측불가능 데이터인 엔트로피를 생성하는 역할을 한다. 엔트로피 소스에는 링 발진기 (ring oscillator), PLL (phase-locked loop), 셀룰러 오토마타 (cellular automata) 등이 사용된다. 엔트로피 추출 회로는 엔트로피 소스로부터 가능한 많은 엔트로피를 얻어낼 수 있도록 설계되어야 한다. 난수의 예측 불가능성을 엔트로피 소스에 두기 위해서는 엔트로피 추출 회로는 간단하게 설계하여 분석이 용이해야 한다. 후처리 회로는 엔트로피 소스와 엔트로피 추출 회로의 결함을 숨기거나 환경의 변화 및 변조 시 내성을 제공하기 위해서 사용된다. 후처리 회로를 사용하지 않고 난수를 생성할 수 있다면 설계자의 판단에 따라 이 회로를 사용하지 않고 순수 난수 발생기를 설계할 수 있다 [4]. 후처리 회로는 Von Neumann correction, LFSR (linear feedback shift register), XOR 축약 (reduction) 등이 사용된다. 무작위성과 엔트로피 수치 등은 선택된 기술에 의해 크게 좌우되며 표준이나 권장 순수 난수 발생기는 존재하지 않는다.

본 논문에서는 self-timed 링 기반의 순수 난수 발생기의 하드웨어 복잡도를 감소시키는 설계방법에 관해 기술한다. Ⅱ장에서는 self-timed 링 구조와 동작에 대해 간략히 설명하고, Ⅲ장에서는 엔트로피 추출 회로에 피드백 구조를 삽입한 self-timed 링 기반의 순수 난수 발생기 설계에 관해 설명한다. Ⅳ장에서는 설계된 FSTR-TRNG의 FPGA 검증 및 성능 평가에 관해 기술하고, 마지막으로 Ⅴ장에서 결론을 맺는다.

Ⅱ. Self-timed 링

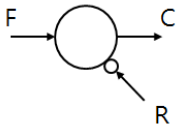
2.1. Self-timed 링의 구조

그림 1은 self-timed ring (STR)를 구성하는 링 스테이지의 기본 구조 및 진리표이다. 링 스테이지는 Muller 게이트와 인버터로 구성되며, 두 입력 F와 R이 동일한 값이면 출력 C는 이전값을 유지하고, 두 입력이 다른 값이면 입력 F의 값을 출력 C로 내보내는 동작을 한다.

STR의 구조는 그림 2와 같으며 마이크로 파이프라인과 2-위상 핸드셰이크 (handshake) 프로토콜을 기반으로 한다 [5]. 입력 F는 이전 링 스테이지의 출력이고 입력 R은 다음 링 스테이지의 출력이다. 현재 링 스테이지의 출력을 만들기 위해서 이전과 다음 링 스테이지의 출력이 사용된다. 2-위상 핸드셰이크 프로토콜의 첫 번째 단계에서는 현재 링 스테이지의 출력이 입력 R을 요청하는 신호로 사용됨과 동시에 출력이 다음 링 스테이지의 입력 F로 사용 가능함을 알려준다. 두 번째 단계에서는 이전 링 스테이지의 출력을 저장하고 그 데이터가 사용되었음을 알리는 신호로 링 스테이지의 출력이 사용된다. 이 신호는 이전 링 스테이지의 입력 R이 된다.

2.2. 토큰과 버블 [6]

토큰(T)과 버블(B)은 현재 링 스테이지 i와 다음 링 스테이지 $i+1$간에 어떤 상관관계가 있는지를 알려주



F	R	C
0	0	Previous C
	1	0
1	0	1
	1	Previous C

Fig. 1 Basic structure of ring stage and its truth table.

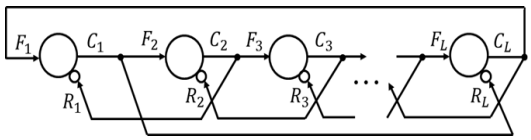


Fig. 2 Architecture of self-timed ring.

는 정보이다. 식 (1)과 같이 현재 링 스테이지의 출력 C_i와 다음 링 스테이지의 출력 C_{i+1}이 상이할 경우에는 현재 링 스테이지는 토큰을 포함한다. 만약 식 (2)와 같이 두 출력 C_i와 C_{i+1}의 값이 동일한 경우에는 현재 링 스테이지는 버블을 포함하게 된다.

$$C_i \neq C_{i+1} \Rightarrow stage_i = T \tag{1}$$

$$C_i = C_{i+1} \Rightarrow stage_i = B \tag{2}$$

토큰과 버블은 STR이 진동하는 조건에도 영향을 끼치며 그에 대한 조건은 다음과 같다. ① $NB \geq 1$ (NB는 버블의 개수), ② $NT\%2 = 0$ (NT는 토큰의 개수), ③ $L \geq 3$ (L은 링 스테이지의 개수). 버블은 토큰을 이동시켜 데이터를 전파시키기 때문에 하나 이상이 존재해야 하며 전파가 균일하게 진행되기 위해 토큰의 수는 반드시 짝수여야 한다. 따라서 STR의 링 스테이지는 3개 이상이어야 한다는 조건을 갖는다.

데이터의 전파 경로는 토큰과 버블의 상관관계에 의해 정해진다. 전파 경로의 규칙은 마이크로 파이프라인 구조에서 토큰이 전파될 데이터가 존재하는 단계이고 버블은 새로운 데이터를 받아들이는 단계라는 것을 기초로 한다. 만약, 현재 링 스테이지가 토큰을 포함하고 다음 링 스테이지가 버블을 포함하고 있다면 C_{i+1}은 C_i가 되고 토큰과 버블은 서로 교환된다. 그림 3-a는 2개의 토큰을 가지는 3-스테이지 STR의 데이터 전파 경로를 보인 것이며, 그림 3-b는 각 스테이지의 출력이 진동하고 있음을 보여주고 있다.

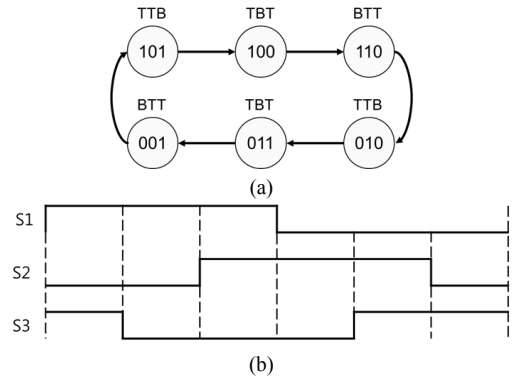


Fig. 3 (a) State transition graph of 3-stage ring with two tokens (b) Timing diagram of 3-stage ring with two tokens.

2.3. Self-timed 링의 진동 모드

STR의 진동 모드는 버스트 (burst) 모드와 등간격 (evenly-space) 모드가 있으며, Charile 효과에 영향을 받는다. 각 모드에 대한 STR의 동작은 그림 4와 같다. 버스트 모드의 경우 링 주변에 클러스터를 형성하면서 전파가 진행되며, 등간격 모드는 링 주위로 퍼져나가며 전파가 일어난다 [7]. Charlie 효과는 입력간 시간 차이가 Muller 게이트의 지연에 영향을 주는 현상으로 시간 차이가 짧을수록 전파 지연이 길어진다. 두 입력이 짧은 시간 간격으로 입력될 때 링 스테이지에서 생긴 지연은 현재 링 스테이지 출력과 바로 다음 스테이지의 입력에 영향을 주게 되며, 이때 발생한 지연 증가로 인해 링 스테이지의 출력들이 서로 밀려나게 된다. 이러한 현상은 STR이 등간격 모드로 동작하게 하여 일정한 간격으로 진동이 일어나도록 한다. STR의 진동 모드와 그에 대한 작동 지점은 디자인 매개 변수에 따라 달라지며, 등간격 모드로 동작하기 위한 조건은 식 (3)과 같다. D_{ff}는 정적 포워드 지연 (static forward delay)을 의미하고 D_{rr}은 정적 리버스 지연 (static reverse delay)을 의미한다. 시뮬레이션 결과를 통해 식 (4)와 같은 비율을 유지한다면 STR이 등간격 모드로 동작할 뿐만 아니라 Charlie 효과를 최대화 할 수 있음이 알려져 있다 [6].

$$\frac{N_T}{N_B} \simeq \frac{D_{ff}}{D_{rr}} \tag{3}$$

$$\frac{D_{ff}}{D_{rr}} = \frac{2}{3} = \frac{N_T}{N_B} \tag{4}$$

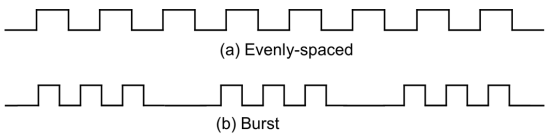


Fig. 4 Evenly-spaced and burst propagation modes in self-time ring.

Ⅲ. 피드백 구조를 갖는 STR 기반의 TRNG 설계

3.1. 지터 기반의 무작위성 숫자 생성

본 설계에서는 무작위성을 가지는 숫자를 만들기 위해 지터 (jitter)를 이용하였다. 지터는 신호가 발생하는 이상적인 시간에서 열잡음, 샷 잡음, 전원공급 장치로부터의 잡음 등과 같은 여러 가지 현상에 의해 매우 짧은 범위 내에서 신호가 시간적인 차이를 가지고 나타나는 현상이다. 신호가 지터에 의해 시간적으로 변화되어 나올 수 있는 시간 범위를 지터 경계라고 한다. 그림 5는 Muller 게이트와 인버터로 구성된 링 스테이지의 출력 C에 발생한 지터 경계를 샘플링 주파수로 샘플링 하였을 때의 결과를 나타낸 것이다. 출력 C가 천이되는 구간에 존재하는 지터 경계에서 샘플링을 진행했을 때 샘플링된 값이 어떤 값을 가지는지 알 수 없기 때문에 무작위성을 가지게 된다. 하지만 샘플링 된 값을 직접 난수 발생기의 출력으로 사용한다면 무작위성이 떨어지는 숫자가 얻어질 가능성이 크다. 지터 경계는 보통 주기의 1% 미만으로 나타나기 때문에 샘플링 시 샘플링 신호가 지터 경계를 벗어나지 않도록 설계를 진행하는 것이 어렵기 때문이다.

링 발진기 기반의 순수 난수 발생기는 그림 6과 같은 구조를 가지며, N-개의 링 발진기를 다수의 인버터로 구성하고, 링 발진기의 출력들을 XOR 한다. 그리고 그 값을 플립플롭을 이용하여 샘플링하여 무작위성을 갖는 값을 만들어낸다. 이러한 구조는 XOR 게이트를 제외한 구조에서 링 발진기의 출력이 지터 경계를 벗어나 샘플링 되었을 경우 샘플링 된 값이 무작위성을 갖지 않는다는 단점을 보완하기 위해 제안되었다. 모든 링 발진기의 출력값이 XOR 된다면 N-개의 링 발진기 중 N-1개의 링

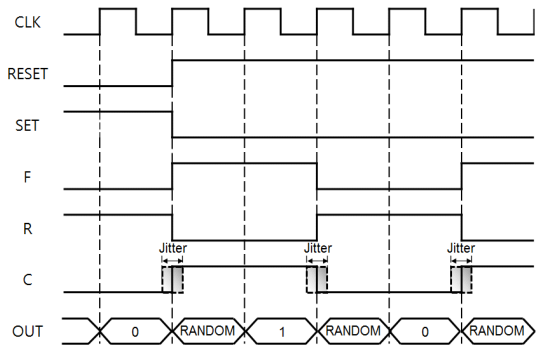


Fig. 5 Timing diagram of the self-timed ring outputs.

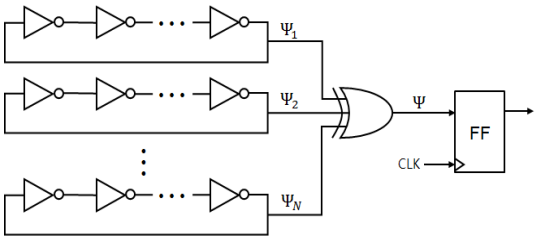


Fig. 6 General architecture of the ring oscillator based TRNG [3].

발진기 출력이 지터 경계를 벗어나 샘플링 되더라도 하나의 출력이라도 무작위성을 가진다면 XOR 게이트의 출력도 무작위성을 갖게 되어 최종 출력도 무작위성을 갖게 된다.

하지만 이러한 구조는 다수의 링 발진기 출력을 필요로 하며 링 스테이지를 인버터로 구성할 경우 많은 인버터가 사용되어야 한다. 또한, 인터버로 구성되는 링 발진기는 지터가 링 스테이지의 수에 따라 가변적이기 때문에 지터 경계를 예측하기가 어렵다는 단점을 갖는다. 이에 반해 STR은 Charlie 효과로 인하여 누적된 지터가 링 스테이지의 출력에 나타나는 것이 아니라 해당 링 스테이지에서 발생하는 지터가 각 링 스테이지의 출력에 나타나며, 따라서 지터 경계에서 샘플링 되도록 설계하기가 용이하다.

3.2. FSTR-TRNG의 구조

본 논문에서 설계된 FSTR-TRNG는 그림 7과 같은 구조를 가지며, 엔트로피 소스로는 STR이 사용되고 엔트로피 추출회로는 D 플립플롭과 피드백 형식의 SIPO (serial-input parallel-output) 시프트 레지스터가 사용된다. STR은 마이크로 파이프라인 구조로 설계되었으며, 각각의 4-입력 링 스테이지는 논리게이트 대신에 룩-업 테이블 (look-up table)을 사용하여 표 1과 같은 동작을 수행하도록 설계하였다. 링 스테이지 수 및 토큰과 버블 개수의 비율은 가변적으로 디바이스 환경이나 엔트로피 추출 회로 등에 따라 설계자가 결정하게 된다. 본 논문의 설계에서는 문헌 [8]의 사례를 참고하여 동작주파수, 토큰과 버블 개수의 비율 (N_T/N_B), 진동모드 등의 파라미터를 결정하였다.

엔트로피 추출 회로는 그림 8의 구조를 가지며, SIPO 시프트 레지스터, L개의 멀티플렉서와 D 플립플롭, L+2개의 XOR 게이트로 구성된다. 링 스테이지의 각 출력

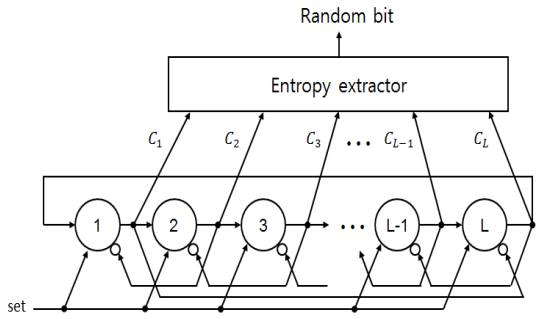


Fig. 7 Architecture of the FSTR-TRNG.

은 플립플롭으로 샘플링되어 XOR 게이트와 멀티플렉서의 입력으로 사용된다. 이때 샘플링 주파수의 상승 모서리와 STR의 출력에 천이가 발생하였을 때의 지터 경계를 고려하여 설계를 진행하였다. SIPO 시프트 레지스터의 초기값은 멀티플렉서의 출력값을 각각의 링 스테이지 초기값으로 설정하기 위하여 2^L-1로 결정하였다. FSTR-TRNG의 최종 출력으로 나오는 1-비트 값이 SIPO 시프트 레지스터의 입력으로 인가되며, 만약 이 값이 무작위성을 갖는 값이라면 L-클록 동안의 최종 출력값들은 무작위성을 갖는다. SIPO 시프트 레지스터의 출력값은 멀티플렉서의 활성화 신호로 사용되도록 설계하였다. 멀티플렉서의 출력들은 최종 출력값을 만들어내기 위한 XOR 게이트의 입력으로 사용되거나 링 스테이지의 출력과 XOR 되어 다시 멀티플렉서의 입력으로 사용된다. 따라서 멀티플렉서의 출력이 무작위성을 갖는다면 전체 회로의 출력 또한 무작위성을 갖는다.

Table. 1 Truth table of 4-input ring stage.

RESET	SET	F	R	C_{i-1}	C_i
0	-	-	-	-	0
1	1	-	-	-	Intial value
	0	0	0	0	0
				1	1
			1	0	0
				1	
		1	0	0	1
				1	
			1	0	0
				1	1

Ⅳ. 하드웨어 동작 검증 및 성능 평가

4.1. FPGA 구현을 통한 하드웨어 동작 검증

설계된 FSTR-TRNG를 Spartan-6 FPGA 디바이스에 구현하고, 무작위성 테스트를 통해 하드웨어 동작을 확인하였다. FPGA에 구현된 FSTR-TRNG로부터 2×10^7 비트의 랜덤 데이터를 추출하여 NIST SP 800-22에 규정된 15가지 테스트 [9]를 실시하였다. FSTR-TRNG가 25 MHz의 정수배 동작주파수로 동작할 수 있도록 문헌 [8]을 참고하여 링 스테이지 수를 11의 배수로 결정하고 등간격 진동 모드로 동작하도록 설정하였다. 최적의 링 스테이지 수와 N_T/N_B 비를 결정하기 위해 해당 값들을 가변시키며 FSTR-TRNG의 하드웨어 동작에 대한 무작위성 테스트를 실시하였다. 최적의 링 스테이지 수와 N_T/N_B 비를 결정하기 위한 NIST 800-22 테스트 결과는 표 2와 같다. 링 스테이지 수가 33인 경우에는 무작위성 테스트를 통과하지 못하는 것으로 나타났으며, 링 스

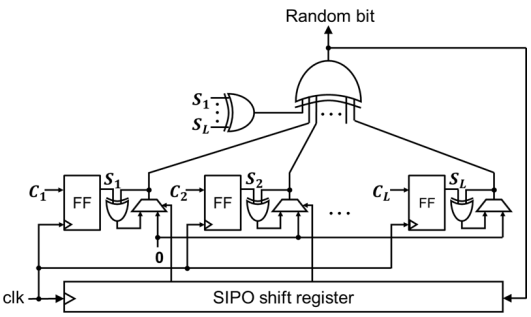


Fig. 8 Proposed entropy extractor for FSTR-TRNG.

Table. 2 NIST 800-22 test results of FSTR-TRNG for determining the optimal number of ring stages and N_T/N_B ratio.

FPGA Device	Number of stages	N_T/N_B	Results
Spartan-6	33	6T / 5B	FAIL
		8T / 3B	FAIL
	44	2T /3B	FAIL
		6T / 5B	FAIL
		8T / 3B	PASS
	55	2T /3B	FAIL
		6T / 5B	FAIL
		8T / 3B	PASS

테이지 수가 44 이상이고 $N_T/N_B = 8T/3B$인 경우에 무작위성 기준을 통과하여 최적의 조건임을 확인하였다. 이와 같은 FPGA 구현을 통한 하드웨어 동작 검증 및 최적조건 결정을 토대로, 본 논문에서는 링 스테이지 수를 44, $N_T/N_B = 8T/3B$로 FSTR-TRNG를 구현하였다. Spartan-6 FPGA에 구현된 FSTR-TRNG로부터 2×10^7 비트의 랜덤 스트림을 생성하여 NIST 800-22 테스트를 수행한 항목별 결과는 표 3과 같으며, 15개의 테스트 항목이 모두 기준값을 만족하였다. P-value 값은 NIST SP 800-22 테스트에서 제시하는 기준값인 0.01을 적용하였다. Proportion 값은 테스트에 사용되는 데이터 개수에 의존하며, 테스트에 사용되는 데이터 개수가 2×10^7 비트인 경우에 Proportion 기준값은 0.9233이 된다. P-value와 Proportion 값이 주어진 기준값 이상이면 무작위성 테스트를 통과하는 것으로 판정한다.

설계된 FSTR-TRNG는 보안 시스템온칩 (System-on-Chip; SoC)에 무작위 난수 발생 IP (intellectual property)로 사용될 수 있음을 그림 9와 같은 SoC 검증 플랫폼을

Table. 3 NIST 800-22 test results of FSTR-TRNG with 44 ring stages and $N_T/N_B = 8T/3B$.

Statistical test	P-value(0.01)	Proportion	Results
Frequency (Monobit)	0.9642	1	PASS
Block Frequency	0.0487	1	PASS
Cumulative Sums	0.3504 0.9642	1	PASS
Runs	0.9114	1	PASS
Longest Runs of Ones	0.1223	1	PASS
Binary Matrix Rank	0.9642	1	PASS
FFT	0.6371	0.95	PASS
Non-Overlapping Template Matching	0.0668 0.9914	0.9895	PASS
Overlapping Template Matching	0.2757	1	PASS
Universal Statistical	0.7399	1	PASS
Approximate Entropy	0.4372	1	PASS
Random Excursions	0.1626 0.8343	1	PASS
Random Excursions Variant	0.0487 0.9642	1	PASS
Serial	0.0668 0.0909	1	PASS
Linear Complexity	0.6371	1	PASS

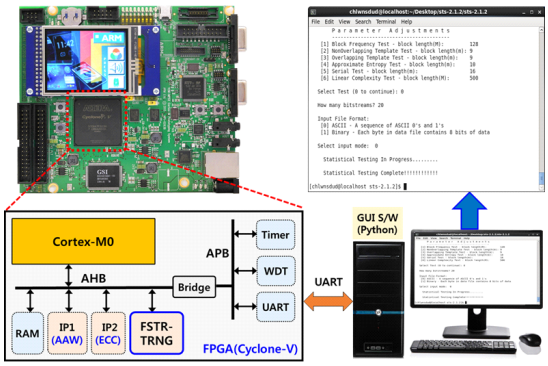


Fig. 9 SoC verification of FSTR-TRNG used as an IP.

이용하여 검증하였다. FSTR-TRNG는 대칭키 블록암호 IP (AAW), 공개키 암호 IP (ECC) 등과 함께 Cortex-M0에 AHB로 인터페이스 되고, Cortex-M0의 제어에 의해 FSTR-TRNG에서 무작위 난수가 생성되어 AAW, ECC 등의 암호 IP에 비밀키 또는 개인키로 사용되어 보안 SoC가 올바로 동작함을 검증하였다.

설계된 FSTR-TRNG를 Spartan-6 FPGA 디바이스로 합성한 결과, 46 슬라이스로 구현되었으며, 180 nm CMOS 표준셀로 합성한 결과 약 2,500 GEs로 구현되었다. 표 4는 본 논문의 FSTR-TRNG와 문헌에 발표된 TRNG의 비교를 보인 것이다. 본 논문의 FSTR-TRNG는 엔트로피 추출기에 피드백 구조를 도입하여 링 발진기의 스테이지 수를 감소시켰으며, 따라서 TRNG 전체의 하드웨어 복잡도가 가장 작음을 확인할 수 있다. 동일한 FPGA 디바이스를 사용하는 문헌 [10]의 TRNG와 비교하여 본 논문의 FSTR-TRNG가 약 30% 적은 하드웨어를 사용하면서도 랜덤 비트 생성율이 약 7배 높아 우수한 것으로 평가되었다.

Table. 4 Comparison with different TRNGs.

Designs	FPGA device	Resources	Throughput [Mbps]
Ref [10]	Spartan 6	67 slices	14.3
Ref [11]	Virtex 5	220 LUTs	30
Ref [12]	Virtex 5	47 slices	150
This work	Spartan 6	46 slices	100*

* at 100MHz clock frequency

Ⅴ. 결 론

사물인터넷 (IoT)과 같이 제한된 하드웨어 자원을 갖는 분야의 보안 하드웨어 구현에 적합한 저면적 무작위 난수발생기를 설계하였다. 하드웨어 복잡도 최소화를 위해 엔트로피 추출회로에 피드백 구조를 적용하였으며, 이를 통해 링 스테이지 수를 감소시켰다. 설계된 FSTR-TRNG를 Spartan-6 FPGA 디바이스에 구현하여 NIST 800-22 테스트를 수행한 결과, 모든 항목이 기준값을 만족함을 확인하였다. Spartan-6 디바이스로 합성한 결과 46 슬라이스로 구현되었으며, 180 nm CMOS 표준셀로 합성한 결과 약 2,500 GEs의 저면적으로 구현되었다. 설계된 FSRT-TRNG는 타원곡선 암호 (ECC) 코어, SHA-3 해시 함수 코어, AES 대칭키 암호 코어와 함께 보안 SoC에 난수발생기로 내장되어 타원곡선 전자서명 (EC-DSA), EC-ElGamal 프로토콜 구현에 사용되어 유용성이 입증되었으며, 다양한 정보보안 응용분야에서 비밀키 생성 및 무작위 난수 발생 용도로 사용될 수 있다.

ACKNOWLEDGEMENT

- This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (No. 2017R1D1A3B03031677)
- This research was supported by the KIAT(Korea Institute for Advancement of Technology) grant funded by the Korea Government(MOTIE : Ministry of Trade Industry and Energy). (No. N0001883, HRD Program for Intelligent semiconductor Industry)
- Authors are thankful to IDEC for EDA tool support

REFERENCES

[1] M. Bakiri, C. Guyeux, J.-F. Couchot, and A. K. Oudjida, “Survey on hardware implementation of random number generators on FPGA: Theory and experimental analyses,” *Computer Science Review*, vol. 27, pp. 135-153, 2018.

[2] J. Balasch, F. Bernard, V. Fischer, M. Grujic, M. Laban, O. Petura, V. Rozic, G. V. Battum, I. Verbauwhede, M. Wakker, and B. Yang, “Design and testing methodologies for true random number generators towards industry certification,” *2018 IEEE 23rd European Test Symposium (ETS)*, Bremen, pp. 1-10. 2018.

[3] M. Stipcevic, and C.K. Koc, “True Random Number Generator,” In Koc (eds) Open Problems in Mathematics and Computational Science, Springer, Cham, 2014.

[4] O. Petura, U. Mureddu, N. Bochard, V. Fischer, and L. Bossuet, “A survey of AIS-20/31 compliant TRNG cores suitable for FPGA devices,” *2016 26th International Conference on Field Programmable Logic and Applications (FPL)*, Lausanne, Switzerland, pp. 1-10, 2016.

[5] I. E. Sutherland, “Micropipelines,” *Communications of the ACM*, vol. 32, no. 6, pp. 720-738, Jun. 1989.

[6] J. Hamon, L. Fesquet, B. Miscopein, and M. Renaudin “High-Level Time-Accurate Model for the Design of Self-Timed Ring Oscillators,” *2008 14th IEEE International Symposium on Asynchronous Circuits and Systems*, Newcastle upon Tyne, pp. 29-38, Apr. 2008.

[7] A. Winstanley, and M. Greenstreet, “Temporal properties of self-timed rings,” In: *Margaria T., Melham T. (eds) Correct Hardware Design and Verification Methods. CHARME 2001. Lecture Notes in Computer Science*, vol. 2144, Springer, Berlin, Heidelberg, 2001.

[8] O. Elissati, E. Yahya, L. Fesquet, and S. Rieubon, “Oscillation Period and Power Consumption in Configurable Self-Timed Ring Oscillators,” *2009 Joint IEEE North-East Workshop on Circuits and Systems and TAISA Conference*, Toulouse, pp. 1-4, 2009.

[9] L. Bassham, A. Rukhin, J. Soto, J. Nechvatal, M. Smid, E. Barker, S. Leigh, M. Levenson, M. Vangel, D. Banks, N. Heckert, and J. Dray, “A Statistical Test Suite For Random and Pseudorandom Number Generators for Cryptographic Applications,” *NIST Special Publication 800-22*, Apr. 2010.

[10] V. Rozic, B. Yang, W. Dehaene, and I. Verbauwhede, “Highly efficient entropy extraction for true random number generators on FPGAs,” *2015 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, San Francisco, CA, pp. 1-6, 2015.

[11] C. Li, Q. Wang, J. Jiang, and N. Guan, “A metastability-based true random number generator on FPGA,” *2017 IEEE 12th International Conference on ASIC (ASICON)*, Guiyang, pp. 738-741, 2017.

[12] Y. Zhsnh, Q. Wang, J. Jiang, and N. Guan, “A Self-Timed Ring Based True Random Number Generator on FPGA,”

2018 14th IEEE International Conference on Solid-State and Integrated Circuit Technology (ICSICT), pp. 1-3, 2018.

최준영(Jun-Yeong Choe)

2019 : BS degree in Electronic Engineering, Kumoh National Institute of Technology
2019~ Graduate student, Kumoh National Institute of Technology

신경욱(Kyung-Wook Shin)

1984 : BS degree in Electronic Engineering, Korea Aerospace University
1986 : MS degree in Electronic Engineering, Yonsei University
1990 : Ph.D. degree in Electronic Engineering, Yonsei University
1990~1991 : Senior Researcher, Semiconductor Research Center, Electronics and Telecommunications Research Institute (ETRI)
1991~ : Professor in School of Electronic Engineering, Kumoh National Institute of Technology
1995~1996 : University of Illinois at Urbana- Champaign (Visiting Professor)
2003~2004 : University of California at San Diego (Visiting Professor)
2013~2014 : Georgia Institute of Technology (Visiting Professor)