IEICE TRANS. INF. & SYST., VOL.E103–D, NO.2 FEBRUARY 2020

PAPER Special Section on Security, Privacy, Anonymity and Trust in Cyberspace Computing and Communications

New Pseudo-Random Number Generator for EPC Gen2

Hiroshi NOMAGUCHI[†], **Chunhua SU**[†a)], ***Nonmembers***, ***and*** **Atsuko MIYAJI**[††,†††b)], ***Member***

SUMMARY RFID enable applications are ubiquitous in our society, especially become more and more important as IoT management rises. Meanwhile, the concern of security and privacy of RFID is also increasing. The pseudorandom number generator is one of the core primitives to implement RFID security. Therefore, it is necessary to design and implement a secure and robust pseudo-random number generator (PRNG) for current RFID tag. In this paper, we study the security of light-weight PRNGs for EPC Gen2 RFID tag which is an EPC Global standard. For this reason, we have analyzed and improved the existing research at IEEE TrustCom 2017 and proposed a model using external random numbers. However, because the previous model uses external random numbers, the speed has a problem depending on the generation speed of external random numbers. In order to solve this problem, we developed a pseudorandom number generator that does not use external random numbers. This model consists of LFSR, NLFSR and SLFSR. Safety is achieved by using nonlinear processing such as multiplication and logical multiplication on the Galois field. The cycle achieves a cycle longer than the key length by effectively combining a plurality of LFSR and the like. We show that our proposal PRNG has good randomness and passed the NIST randomness test. We also shows that it is resistant to identification attacks and GD attacks.

key words: *NLFS (Non-linear feedback shift register), SLFSR (Skip-Linear feedback shift register), pseudo-random number generator, RFID, EPC Gen2*

1. Introduction

1.1 BackGround

A smart device having a communication function is one of the main components of IoT. In particular RFID [1] are considered various usages and applications, it is expected that one of the smart device that is responding to various needs. Cryptographic primitives can provide secure communications between the RFID reader and tag by using elaborately generated cryptographic keys. These unpredictable and irreproducible secret keys determine the communication security, which are created by a pseudo-random number generator (PRNG). Under such background, the importance of RFID orientated PRNG is on the rise. The regular PRNG is difficult directly to be applied to RFID tags. Therefore, it is needed to be developed to pseudo-random number generator with sufficient security can be used as a primitive for possible saving resources mounted on smart device operates in the power saving. In this paper, we focus on the extremely light-weight pseudo-random number generator for EPC Gen2 RFID tags. The following conditions were set for development. The key length is 80 or more [6], [7]. Circuit scale is less than or equal to 2000GE [8], [9]. Two of the above, is the most severe conditions in which we were in the eye.

1.2 Our Job

In the present study, it have a key length of the need to use as security of the core primitive, a sufficient security, and propose a pseudo-random number generator that also has excellent statistical evaluation. Our contributions are summarized as follows:

- Based on Wabler construction, we improve the security by extending the key length (at least 80bit), which is more secure with larger key space.
- We consider the implementation on the real-world EPC Gen2 RFID tags. The scale of the circuit is less than or equal to 2000GE which is outperform the existing PRNG scheme for EPC Gen2 tags under the same security level.
- Based on our experimental analysis using the NIST pseudo-randomness test package, we show that our proposed PRNG pass all 16 tests and does not have bias.
- Resistant to existing attacks.
- It should be able to operate independently without using external random numbers.

In this paper, we propose a new model that satisfies the above conditions. This is a development of the model previously announced at TrustCom 2017 of IEEE. This new model consists of LFSR, DLFSR and Skip-control feedback shift register (hereafter, SLFSR), and it has sufficient cycle and safety as a pseudo random number generator for RFID. It is pseudo-random number generator for the existing smart devices based on NLFSR and the SLFSR. We used NIST SP 800-22 for the evaluation of statistical random number characteristics. As an evaluation of existing attacks, we evaluated by applying identification attack and GD attack. This paper shows the structure of the proposed model and that there are no problems as a result of their evaluation.

Manuscript received March 9, 2019.
Manuscript revised October 6, 2019.
Manuscript publicized November 14, 2019.

[†]The authors are with Division of Computer Science, University of Aizu, Aizu-Wakamatsu-shi, 965–8580 Japan.
[††]The author is with Osaka University, Suita-shi, Osaka, 565–0871 Japan.
[†††]The author is with Japan Advanced Institute of Science and Technology, Nomi-shi, 923–1292 Japan.
a) E-mail: chsu@u-aizu.ac.jp
b) E-mail: miyaji@comm.eng.osaka-u.ac.jp

DOI: 10.1587/transinf.2019INP0009

2. Related Works

As a pseudo-random number generator for the EPC Gen2, J3Gen [4], Warbler (32,2,5,6) [2] and (62,3,5,6) [3], LAMED [10], AKARI [11], Grain [12] has been proposed. These are the performance of the pseudo-random number indicated by the EPC Gen2 meets. However, looking security and privacy that is beginning to be newly requested to RFID has not been done well security evaluation. Further, there is even the security evaluation by the author, also those attacks as J3Gen [5] and Grain has been reported. There is a Warbler as one of those who have not been reported for about attack and have the security assessment to the author, but the key length is less than 80bit. In addition, Warbler may not provide enough security under powerful adversarial setting. For example, the internal state of the NLFSR6 can be recovered form the output and there is a bias in the input and output of the PRNG. In order to solve these problems, we proposed IEEE TrustCom 2017 pseudo random number generator with external random number, NLFSR and DLFSR. Because it uses external random numbers, it is affected if the random number generation speed of the implementation environment is slow.

3. Security Analysis for Wabler and Our Previous Proposal Model

EPC Gen2 tag is required to be operated with restricted computing resources such as memory and power consumption. In the existing research, there are two major PRNG constructions for EPC Gen2 RFID tag. The Warbler is based on multiplied and NLFSR using Trace. Our previous proposal model is based on NLFSR, DLSFR and external random number r. Notation is defined as follows.

$\boxplus$	: Addition
$+$	: Exclusive OR
f	: Primitive polynomial
f_j	: The j-th primitive polynomial
Select	: Location of the primitive polynomial to be used in a feedback
Select (j)	: Choose from the j-th primitive polynomial Select
ζ, ϵ, μ	: NLFSR of register. It shows the index i the register number
a	: NLFSR6 register. Each register is 5 bits
lf	: Register number of DFSR
l	: The period of the switching of primitive polynomial of DFSR and random number r, rm
t	: 5 bits memory. Subscript denotes the i-th bit

3.1 Warbler

This section describes the Warbler (62,3,5,6). Algorithm is shown in Fig. 1. Warbler is configured to the three NLFSR of 1 bit 1 register, one NLFSR of five bits of 1 register and use the extension field. The internal state of the Warbler (62,3,5,6) is a 92-bit, the key length is 60 bits. The feature of Warbler is NLFSR which guarantees the maximum possible cycle and NLFSR 6 which multiplies the extension field.

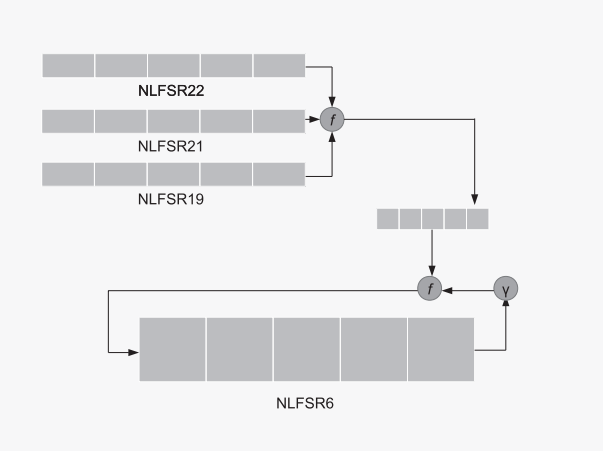

Fig. 1 Warbler (62,3,5,6).

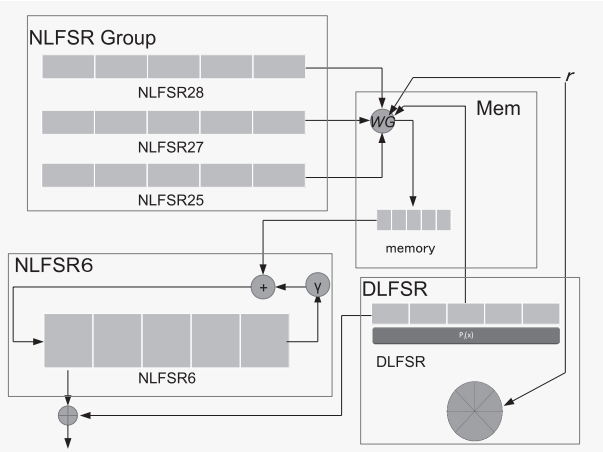

Fig. 2 Previous proposal mode.

Since the explanation of the details is not our work, we omit it.

3.2 Our Previous Proposal Mode

The previous proposal model consists of NLFSR group and DLFSR, and NLFSR 6 using multiplication on the Galois field. It has a key length of 96 bits and has shown in the past that it has statistical random number characteristics, resistance to identification attacks and GD attacks. The algorithm is as shown in Fig. 2.

3.3 Security Analysis of Warbler

Gangqiang Yang says Warbler has claimed to have a tolerance to the next attack. Algebraic attack, Resistance Against Algebraic Attacks and its development attack, Weak Internal States and Fault Injection Attacks. Currently attack has not been reported, but the following features are disturbing. The output of the Warbler is as follows.

$$O_{k+1} = WG(a^3_{k+5})$$

Table 1 A truth table of f.

		x_2	
		0	1
(x_0, x_1)	(0, 0)	0	0
	(0, 1)	1	0
	(1, 0)	1	0
	(1, 1)	1	1

If you know the output O_{k+1} (0 or 1) at the time $k+1$ round, you can identify the internal state of the NLFSR6 by order 2^4. This is because the outputs in the form of compression of the internal state of NLFSR6. In addition, if the attacker has to guess the NLFSR6 and memory $t_{K+1,K+2,K+3,K+4}$, 1 bit is determined to be applied to new in memory t_{K+5} from the output of the next time $K+1$. For this reason, the value associated with the state transition of NLFSR6 is better higher computational security. Next, attention is paid to the NLFSR19, 21, 22. The output of the NLFSR19, 21, 22 obtain an output by enter the following formula to f.

$$f(x_0, x_1, x_2) = x_0x_1 + x_1x_2 + x_0x_2 + x_0 + x_1 \ (x_{0,1,2} \in F_2)$$

A truth table of the input-output relationship shown in Table 1. From Table 1, if f outputs 0, $x_2 = 1$ a is the probability of 3/4, is the probability $x_2 = 1$ is 1/4. If f is outputs 1 is 3/4 $x_2 = 0$ of probability, is a 1/4 probability that $x_2 = 1$. There is a bias in this way. f is considered to be the effect of hiding a portion of the input at that time by ignoring any of the x_0, x_1, x_2. However, such bias is likely to be used in the attack. Therefore, it considered that it is desirable to replace the another function to keep the deviation without computational security. Next, consider the circuit scale. Compared to the Warbler (32,2,5,6) is a circuit scale 937GE, the circuit scale 1238GE of Warbler (62,3,5,6) that internal state has increased 27. When the future to think, than increasing the simple internal state for key length increase, there is a possibility that more than 2000GE is a restriction of the EPC Gen2. Considering that in the future be required to recommend equivalent to the key length of the common key encryption, it is to increase the simple internal state for key length increase, there is a possibility that more than 2000GE is a restriction of the EPC Gen2.

3.4 Security Analysis of Our Previous Proposal Model

The previous proposal model consists of NLFSR group, DLFSR and NLFSR6. We designed to maintain a period longer than a certain period and a key length of 96 bits and achieve a circuit scale of 2000 GE or less. The operation of each part is shown. The NLFSR group consists of three NLFSRs whose internal state has been expanded compared with those of Warbler, and the guarantee period as a whole is about 2^{80}. In addition, the output of which is non-linear reduction by the WG. WG is composed of WGP for multiplying should be definitive in the Galois field and Trace. As of the following formulas, respectively law of Galois field

$$\text{NLFSR28: } f_1 = x^5 + x^3 + x + 1,$$

$$\text{NLFSR27: } f_2 = x^5 + x^4 + x^2 + x + 1,$$

$$\text{NLFSR25: } f_3 = x^5 + x^4 + x^3 + x^2 + 1$$

Next, a description will be given of WGP to be used in the WG. Input is referred to as x ($x \in F_2^5$). WGP is expressed by the following equation.

$$\text{WGP}(x) = x + (x+1)^5 + (x+1)^{13} + (x+1)^{19} + (x+1)^{21}.$$

Next, a description will be given of Trace. Trace is shown by the following equation.

$$\text{Trace}(x) = x + x^2 + x^{2^2} + x^{2^3} + x^{2^4} \ (F_2^5 \to F_2)$$

By WGP and Trace, WG is shown as the following equation.

$$\text{WG}(x) = \text{Trace}(\text{WGP}(x^d)) \ (x \in F_2^5)$$

Next, a description will be given NLFSR to use the WG. NLFSR25 (μ), 27 (ϵ), 28 (ζ) consists of 25, 27, 28 stages each one bit of the register.

$$\zeta_{k+28} = 1 + \zeta_k + \text{WG}(x^7),$$
$$x = (\zeta_{k+3}, \zeta_{k+4}, \zeta_{k+8}, \zeta_{k+12}, \zeta_{k+20}),$$
$$\epsilon_{k+27} = 1 + \epsilon_k + \text{WG}(y^{11}),$$
$$y = (\epsilon_{k+4}, \epsilon_{k+10}, \epsilon_{k+12}, \epsilon_{k+15}, \epsilon_{k+20}),$$
$$\mu_{k+25} = 1 + \mu_k + \text{WG}(z^7),$$
$$z = (\mu_{k+3}, \mu_{k+6}, \mu_{k+14}, \mu_{k+16}, \mu_{k+18}).$$

Memory (t) is a save to keep the 5-bit memory output. The operation of the memory is as follows. (Notation as follows. d_i indicates the 16 i-th register of DLFSR. rm_i external random number at the time i).

$$s_i = WG(\zeta_i, \epsilon_i + 1, \mu_i + 1, rm_i, d_i), \ s_j = 0, \ j = 0, 1, 2, 3,$$
$$t_{i+4} = (s_i, s_{i+1}, s_{i+2}, s_{i+3}, s_{i+4})$$

In our previous proposal model, NLFSR6 operates as follows. NLFSR6 is 5-bit 6-stage NLFSR (a). NLFSR6 do multiplication in the enlarged body. Operations are shown below.

$$a_{k+6} = \gamma a_k + a_{k+1} + w_k + t_k, \ w_k = (0, 0, 0, 0, WG(a_k^{11}))$$

DLFSR is based J3Gen. Similarly J3Gen, constituted by Polynomial Selector and LFSR. Polynomial Selector has a function to switch the primitive polynomial LFSR is used. It captures the external random number r ($r \in \{0, 1\}$) for each round (l), choosing a primitive polynomial fb_i. It shows the switch to the following formula.

$$fb_i = Select(j)$$
$$j = j \boxplus r \ (\text{mod } 8)$$
$$Select = f_1, f_2, \cdots, f_8$$
$$f_j : j = 1, 2, \cdots, 8$$

LFSR (l_i) is to use a 1-bit 16-stage. The state transition are as the following equation.

$$lf_{i+1} = fb_i(lf_i)$$

By NLFSR6 and DLFSR, output (O_k) is represented by the following formula.

$$O_k = WG(a^3_{k+5}) + l_{k+15}.$$

There is no problem in circuit size and (statistics, cryptographic) safety. It is as shown in the previous paper. However, depending on the implementation environment, there is a possibility that the speed decreases due to external random numbers. For this reason, it is necessary to review the model.

4. The Proposed Scheme

Our proposed model consists of NLFSR, SLFSR and LFSR. We explain in order of period, nonlinearization. NLFSR 22, 21, 19 and LFSR 37, 31 are disjoint. Therefore, the cycle has 2^{130}. Also, because the SLFSR 35, 33, 32 are also disjoint, the period is 2^{100}. However, since the skip control is performed, the actual cycle is short. However, the overall cycle guarantees at least 2^{130}. It is long enough for the key length. Next, we explain nonlinearization. For nonlinearization, we use NLFSR, SLFSR, WG function, f function. These guarantee a sufficient calculation amount for the key length. Details are described in the section on safety assessment. Next, we explain the algorithm. NLFSR22 (δ), 21 (ϵ), 19 (ζ) are the same as those of Warbler. However, we use WG instead of f function for their output. The reason for this is that the f function has bias on the input and output, and the internal state is inferred efficiently. In order to use WG, it is necessary to be 5 inputs, so in addition to the output of NLFSR, the output of LFSR37 (η), 31 (θ) is also used. For this reason, the output from the WG to the memory (t) is multiplied as follows. (Notation as follows. k mean time.)

$$t_{i+k} = WG(\delta_{22+k}, \epsilon_{21+k}, \zeta_{19+k}, \eta_{37+k}, \theta_{31+k}).$$

NLFSR 6 is the same as the previous proposed model and operates as follows. NLFSR 6 is NLFSR (a) of 5 bits, 6 stages. NLFSR 6 performs multiplication in the extension field. The operation is shown below.

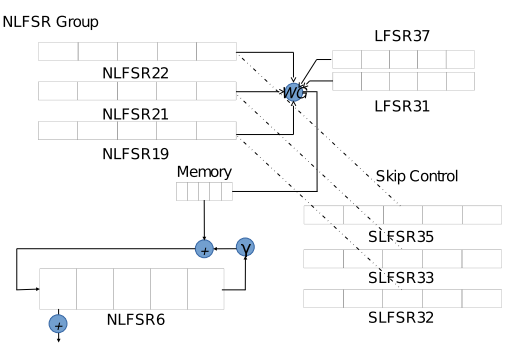

Fig. 3 Proposed scheme.

$$a_{k+6} = \gamma a_k + a_{k+1} + w_k + t_k,\ w_k = (0,0,0,0,WG(a^{11}_k)).$$

SLFSR 35 (ι), 33 (κ), 32 (λ) under the control of NLFSR 22, 21, 19 and skips. Skipping here means that state transition is performed without outputting the internal state. The operation is as follows.

$$\iota_{36+k} = \iota_{35+k} + \iota_{2+k}.$$
$$\kappa_{34+k} = \kappa_{33+k} + \kappa_{22+k} + \kappa_{13+k} + \kappa_{11+k}.$$
$$\lambda_{33+k} = \lambda_{k+32} + \lambda_{22+k} + \lambda_{2+k} + \lambda_{1+k}.$$

SLFSR operates under the control as follows after outputting 1 bit of internal state. It is used for the example of SLFSR 35. If $(\delta_{19}, \delta_{11}) = (0,0)$, it will take 2 clocks. If $(\delta_{19}, \delta_{11}) = (0,1)$, it makes three clocks. If $(\delta_{19}, \delta_{11}) = (1,0)$, do not clock. If $(\delta_{19}, \delta_{11}) = (1,1)$, it clocks 5 clocks. It is the same as the case of SLFSR 33, 32, and the bits to be extracted are $(\epsilon_{17}, \epsilon_5)$ and (ζ_{13}, ζ_7), respectively. The output from each SLFSR is output to the f function. Here, we recognize the internal state as f function fairly and are recognized as Warbler's problem. However, as with GD attacks, SLFSR is resistant to attacks that specify internal state by guess and decision, and is not affected by the problem of f function. For this reason, while obtaining the effect of nonlinearization of the f function, it does not suffer the disadvantage. Details are described in the section on security evaluation. Letting the output of f function be μ and the output of SLFSR 35, 33, 32 be $S35$, $S33$, $S32$, then

$$\mu_k = S35_k + S33_k + S32_k.$$

From the above, the output (O) in the proposed method is multiplied as follows.

$$O_k = WG(a^3_{k+5}) + \mu_k.$$

5. Processing Step

In this chapter, described divided into initialization step, a pseudo-random number generation step (Algorithm 1). In this proposed method, the key and the initial vector each use 128 bits. Express the keys as $Key = K_0, K_1, K_2, \cdots, K_{127}$, and the initial vector IV as $IV = IV_0, IV_1, IV_2, \cdots, IV_{127}$. For initialization, associate the output of NLFSR 6 with the feedback of each NLFSR, LFSR and SLFSR. In this way, in addition to the nonlinearities possessed by each NLFSR and SLFSR, nonlinearities such as multiplication in the extension field of NLFSR 6 are given respectively. Also, with respect to each LFSR, encryption can be started from an internal state having nonlinearity. Initialization should be at least 74 rounds. The reason for this is that it is assumed that the largest LFSR is 37 stages and if it makes 2 rounds, it will be sufficiently stirred. However, when the internal state of any of the registers LFSR and SLFSR is zero, it makes 37 clocks further. Place the key and initial vector as follows.

$$K_{0,\cdots,10} = \delta_{2j}\ (j = 0, \cdots, 10),$$

Algorithm 1 Initialization

$j = k = l = 0, O_0 = 1$
for $i = 0$ to 55 **do**
$\delta_{i+22} = 1 + \delta_i + \mathrm{WG}(x^7) + O_i \quad (x = \delta_{i+3}, \delta_{i+4}, \delta_{k+8}, \delta_{k+12}, \delta_{k+20})$
$\epsilon_{i+21} = 1 + \epsilon_i + \mathrm{WG}(y^{11}) + O_i \quad (y = \epsilon_{i+4}, \epsilon_{i+10}, \epsilon_{i+12}, \epsilon_{i+15}, \epsilon_{i+20})$
$\zeta_{i+19} = 1 + \zeta_i + \mathrm{WG}(x^7) + O_i \quad (x = \zeta_{i+3}, \zeta_{i+6}, \zeta_{i+14}, \zeta_{i+16}, \zeta_{i+18})$
$\eta_{i+37} = \lambda_{i+7} + \lambda_{i+18} + \lambda_{i+30}$
$\mu_{i+25} = 1 + \mu_i + \mathrm{WG}(z^7) + WG(a^3_{i+5})$
$s_i = WG(\zeta_i, \lambda_i + 1, \mu_i + 1, rm_i, d_i),\ s_j = 0,\ j = 0, 1, 2, 3$
$t_{i+4} = (s_i, s_{i+1}, s_{i+2}, s_{i+3}, s_{i+4})$
$w_i = (0, 0, 0, 0, WG(a^{11}_i))$
$a_{i+6} = \gamma a_i + a_{i+1} + w_i + t_i$
for j to $j + \delta_{19}, \delta_{11}$ **do**
$\iota_{36+j} = \iota_{35+j} + \iota_{2+j}$
end for
for k to $k + \epsilon_{17}, \epsilon_5$ **do**
$\kappa_{34+k} = \kappa_{33+k} + \kappa_{22+k} + \kappa_{13+k} + \kappa_{11+k}$
end for
for l to $l + \zeta_{13}, \zeta_7$ **do**
$\lambda_{33+l} = \lambda_{32+l} + \lambda_{22+l} + \lambda_{2+l} + \lambda_{1+l}$
end for
$\mu_i = \iota_j + \kappa_k + \lambda_l$
$O_i = WG(a^3_{i+5}) + \mu_i$
end for

$$
\begin{aligned}
&IV_{0,\cdots,10} = \delta_{2j+1}\ (j = 0, \cdots, 10),\\
&K_{11,\cdots,21} = \epsilon_{2j}\ (j = 0, \cdots, 10),\\
&IV_{11,\cdots,20} = \epsilon_{2j+1}\ (j = 0, \cdots, 9),\\
&K_{22,\cdots,30} = \zeta_{2j+1}\ (j = 0, \cdots, 8),\\
&IV_{21,\cdots,30} = \zeta_{2j}\ (j = 0, \cdots, 9),\\
&K_{31,\cdots,47} = \eta_{2j}\ (j = 0, \cdots, 16),\\
&IV_{31,\cdots,46} = \eta_{2j+1}\ (j = 0, \cdots, 15),\\
&K_{48,\cdots,62} = \theta_{2j+1}\ (j = 0, \cdots, 14),\\
&IV_{47,\cdots,62} = \theta_{2j}\ (j = 0, \cdots, 15),\\
&K_{63,\cdots,80} = \iota_{2j}\ (j = 0, \cdots, 17),\\
&IV_{63,\cdots,79} = \iota_{2j+1}\ (j = 0, \cdots, 16),\\
&K_{81,\cdots,96} = \kappa_{2j+1}\ (j = 0, \cdots, 15),\\
&IV_{80,\cdots,96} = \kappa_{2j}\ (j = 0, \cdots, 16),\\
&K_{97,\cdots,112} = \lambda_{2j+1}\ (j = 0, \cdots, 15),\\
&IV_{97,\cdots,112} = \lambda_{2j}\ (j = 0, \cdots, 15),\\
&K_{113,\cdots,127} = \alpha_{2j\%5,\ j\ (\mathrm{Mod}\ 5)}\ (j = 0, \cdots, 14),\\
&IV_{113,\cdots,127} = \alpha_{2j\%5+1,\ j\ (\mathrm{Mod}\ 5)}\ (j = 0, \cdots, 14),\\
&0 = \eta_j\ (j = 32, \cdots, 36),\ 0 = t_j\ (j = 0, \cdots, 4),
\end{aligned}
$$

6. Evaluation

For security evaluation, we evaluate in order of random number characteristics, period, distinguishing attack, GD attack.

6.1 Random Number Characteristic

The evaluation of the random number characteristics, using the NIST SP800-22. A key and IV were generated by a round function of C programming language, 100 samples of random number series were prepared and tested by NIST SP 800-22. The results are shown in Table 2. It should be noted that, for some multiple of the same item in the test described the first one of those is output.

Table 2 Result of NIST SP800-22 test.

Test name	P-VALUE	PROPORTION
Frequency	0719747	99/100
BlockFrequency	0.289667	99/100
CumulativeSums	0.275709	100/100
Runs	0.224821	99/100
LongestRun	0.759756	100/100
Rank	0.129620	100/100
FFT	0.554420	100/100
NonOverlappingTemplate	0.834308	98/100
OverlappingTemplate	0.437274	100/100
ApproximateEntropy	0.867692	99/100
RandomExcursions	0.689019	59/61
RandomExcursionsVariant	0.585209	60/61
Serial	0.699313	99/100
LinearComplexity	0.153763	98/100

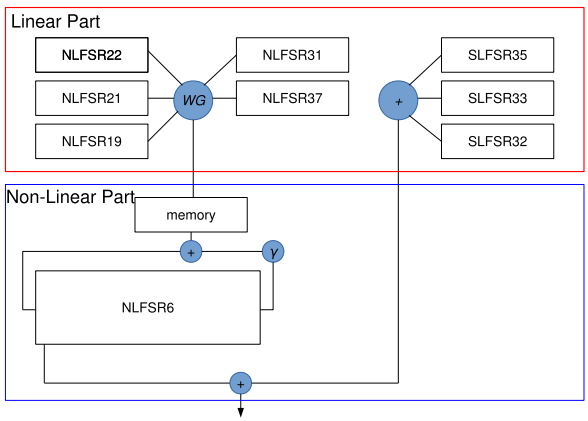

Fig. 4 Simplified model for distinguishing attack.

6.2 Period

NLFSR and LFSR each have a maximum period. At this time, the cycle when these are combined is the least common multiple. Therefore, it has a cycle of 2^{130}, which exceeds 2^{128}, so it has a sufficient cycle.

6.3 Distinguishing Attack

In order to evaluate the security against distinguishing attacks, in this paper we divided the linear part and the nonlinear part like Fig. 4 and examined the attack.

As shown in Fig. 4, NLFSR group and SLFSR are set to linear part, memory and NLFSR 6 are made nonlinear part. This is based on attacks on SNOW. In fact, NLFSR and SLFSR have nonlinearity, but this time we also add NLFSR and SLFSR to the linear part to simplify the evaluation. In order for an identification attack to be successful, the output must be biased. In the pseudo-random number generator composed of the linear part and the nonlinear part as in this model, one of the following is required for a discrimination

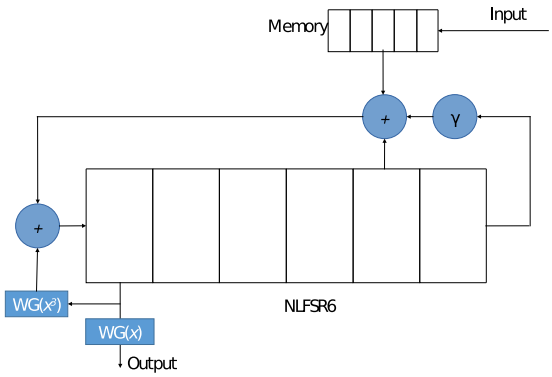

Fig. 5 Search for linear approximation.

attack to be established.

1. There is a bias in the linear part that is input to the nonlinear part.
2. A linear approximation formula must be established in the nonlinear part.

First, let's talk about parts that are handled as linear parts. The linear part consists of LFSR, NLFSR, and SLFSR. Since LFSR and NLFSR have the maximum period, their outputs are not biased. Since SLFSR is for skip control of LFSR, the output is estimated to be unbiased. Consider the memory which is a nonlinear part and NLFSR 6. NLFSR 6 is regarded as a memory with shift function, multiplication of extension field is added when performing state transition. This is the same as the *S* function of SNOW, as the output is one to one correspondence to the input. For this reason, NLFSR 6 can be regarded as an S function and a shifted memory, so it is a nonlinear part. Next, a linear approximation formula of the nonlinear part is obtained. As shown in Fig. 5, the input to the linear part to the nonlinear part was 1 bit added to every round memory, and the output was 1 bit after passing through WG of NLFSR 6. Do not use the exclusive OR of the output 1 bit (compressed 5 bit) of register 0 of NLFSR 6, which is the original output, and SLFSR in order to facilitate the evaluation. Since input and output are both 1 bit, $\Gamma = 1$, we did a full search. As a result, we measured 2, 3, 4 null, but neither bias was detected. From the above, it was found that the linear part is not biased and the nonlinear part is not biased, and the linear approximation formula is not valid. Therefore, our proposed method was shown to be resistant to distinguishing attack.

6.4 GD Attack

To simplify the evaluation, simplify the operation of the proposed scheme. Evaluate from SLFSR as an evaluation of resistance to GD attack. From the output of the scheme, the result of exclusive OR from the three SLFSRs (assumed to be output 1) can not be obtained directly but it is assumed that it can be done. Fix the operation of SLFSR as follows. Every SLFSR depends on NLFSR clock, but evaluates it as having only one clock. The SLFSRs remaining from inference and output 1 of the registers of the two SLFSRs are determined. In this case, it is better to have a smaller number of registers inferred, so SLFSR 32 and SLFSR 31 are estimated. The calculation amount for this is $2^{32+31} = 2^{63}$. In this case, by setting the SLFSR 35 to 35 clocks, the internal state is determined. Similarly, a GD attack is applied to NLFSR 22, 21, 19 and LFSR 37, 31. At this time, similarly to the evaluation of SLFSR, it is assumed that an input to the NLFSR 6, that is, an output of the WG function (assumed to be output 2) is obtained in order to simplify the evaluation. As with SLFSR, when applying attacks to NLFSR and LFSR, it is better for smaller guess inside registers, so LFSR 37 is decided and others are guessed. The computational complexity at this time is 2^{93}, so it is necessary to make 37 clock. The amount of calculation for inferring the internal state so far is A. $2^{93+63} = 2^{156}$. In fact, the output from our proposed scheme consists of the outputs of NLFSR 6 and SLFSR. In order to attack each register, it is necessary to separate the output of NLFSR 6 and SLFSR. For this purpose, it is necessary to estimate in the same way. Also, there are some simplified parts such as the SLFSR clock and WG output being known to the attacker for easy evaluation. Therefore, the actual calculation amount becomes larger than 2^{156}. Therefore, we believe that the proposed scheme has resistance to GD attacks.

7. Conclusion

In this paper, we proposed a pseudorandom number generator for resource saving such as EPC Gen 2 Class 2. The first reason is that existing block ciphers and stream ciphers do not work with less resources because the circuit scale becomes large. The second use is because the existing pseudo-random number generator for resource saving does not meet the required safety. In the development, we aimed for a 128-bit key length equivalent to the security set by a pseudo-random number generator for computers with relatively resources. In addition, it was also required to work with resource-saving devices such as EPC Gen2 for Class2. In addition, it was designed to work with resource-saving devices by choosing lightweight parts. The model proposed in this paper is a further development of the model proposed in TrustCom 2017. The first is not to rely on external random numbers. Second, the proposed model has 128 bits of safety including the period and can prove its resistance to existing attacks. The long period necessary to expand the key length was achieved by combining NLFSR and LFSR and the measure against existing attack was solved by combining NLFSR and SLFSR. In evaluating this scheme, we explained the period, the statistical random number characteristics, the identification attack, the speculation decision attack and showed that there is no problem in performance. In the future, we going to implement it in the device and measure power consumption, speed and circuit scale.

Acknowledgments

This work is partially supported by CREST (JPMJCR1404) at Japan Science and Technology Agency and Innovation Platform for Society 5.0 at MEXT.

References

[1] EPCglobal, EPC Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID, Protocol for Communications at 860–960 MHz.
[2] G. Yang, M.D. Aagaard, and G. Gong, "Efficient hardware implementations of the Warbler pseudorandom number generator," NIST Lightweight Cryptography Workshop 2015.
[3] K. Mandal, X. Fan, and G. Gong, "Design and implementation of Warbler family of lightweight pseudorandom number generators for smart devices," ACM Trans. Embedded Computing Systems, vol.15, no.1, Article 1, Feb. 2016.
[4] J. Melià-Seguí, J. Garcia-Alfaro, and J. Herrera-Joancomartí, "J3Gen: A PRNG for low-cost passive RFID," Sensors, vol.13, no.3, pp.3816–3830, 2013.
[5] A. Peinado, J. Munilla, and A. Fúster-Sabater, "EPCGen2 pseudo-random number generators: Analysis of J3Gen," Sensors, vol.14, no.4, pp.6500–6515, 2014.
[6] C. Paar, A. Poschmann, and M.J.B. Robshaw, "New designs in lightweight symmetric encryption," RFID Security, P. Kitsos and Y. Zhang, Eds., pp.349–371, Springer, Boston, MA, 2008.
[7] A. Bogdanov, L.R. Knudsen, G. Leander, C. Paar, A. Poschmann, M.J.B. Robshaw, Y. Seurin, and C. Vikkelsoe, "PRESENT: An ultra-lightweight block cipher," Cryptographic Hardware and Embedded Systems, CHES 2007, Lecture Notes in Computer Science, vol.4727, pp.450–466, Springer, Berlin, Heidelberg, 2007,
[8] A. Juels, "RFID security and privacy: A research survey," IEEE J. Sel. Areas Commun., vol.24, no.2, pp.381–394, 2006.
[9] S.A. Weis, S.E. Sarma, R.L. Rivest, and D.W. Engels, "Security and privacy aspects of low-cost radio frequency," Security in Pervasive Computing, Lecture Notes in Computer Science, vol. 2802, pp.201–212, Springer, Berlin, Heidelberg, 2004.
[10] P. Peris-Lopez, J.C. Hernandez-Castro, J.M. Estevez-Tapiador, and A. Ribagorda, "LAMED—A PRNG for EPC Class-1 Generation-2 RFID specification," Computer Standards & Interfaces, vol.31, no.1, pp.88–97, Jan. 2009.
[11] H. Martín, E.S. Millán, P. Peris-Lopez, and J.E. Tapiador, "Efficient ASIC implementation and analysis of two EPC-C1G2 RFID authentication protocols," IEEE Sensors J., vol.13, no.10, pp.3537–3547 Oct. 2013.
[12] H. Zhang and X. Wang, Cryptanalysis of Stream Cipher Grain Family.
[13] S. Kiyomoto, T. Tanaka, and K. Sakurai, "Experimental analysis of guess-and-determine attacks on clock-controlled stream ciphers," IEICE Trans. Fundamentals, vol.E88-A, no.10, pp.2778–2791, Oct. 2005.
[14] P. Hawkes and G.G. Rose, "Guess-and-determine attack on SNOW," International Workshop on Selected Areas in Cryptography, Lecture Notes in Computer Science, vol.2595, pp.37–46, Springer, Berlin, Heidelberg, 2003.
[15] L.E. Bassham, A.L. Rukhin, J. Soto, J.R. Nechvatal, M.E. Smid, S.D. Leigh, M. Levenson, M. Vangel, N.A. Heckert, and D.L. Banks, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," Special Publication 800-22 Revision 1a (2014.04).

Hiroshi Nomaguchi received the M. information science from Japan Advanced Institute of Science and Technology in March 2017. He received the 16th IEEE Trustocm 2017 Best Paper Award Admission.

Chunhua Su received the B.S. degree for Beijing Electronic and Science Institute in 2003 and received his M.S. and Ph.D. of computer science from Faculty of Engineering, Kyushu University in 2006 and 2009, respectively. He is currently working as an Associate Professor in Division of Computer Science, University of Aizu. He has worked as a research scientist in Cryptography & Security Department of the Institute for Infocomm Research, Singapore from 2011–2013. From 2013–2016, he has worked as an Assistant professor in School of Information Science, Japan Advanced Institute of Science and Technology. From 2016–2017, he worked as Assistant Professor in Graduate School of Engineering, Osaka University. His research interests include cryptanalysis, cryptographic protocols, privacy-preserving technologies in data mining and IoT security & privacy.

Atsuko Miyaji received the B.Sc., the M.Sc., and the Dr.Sci. degrees in mathematics from Osaka University, Osaka, Japan in 1988, 1990, and 1997 respectively. She joined Panasonic Co., LTD. from 1990 to 1998 and engaged in research and development for secure communication. She was an associate professor at the Japan Advanced Institute of Science and Technology (JAIST) in 1998. She joined the computer science department of the University of California, Davis from 2002 to 2003. She has been a professor at Japan Advanced Institute of Science and Technology (JAIST) since 2007 and the director of Library of JAIST from 2008 to 2012. She has been a professor at Graduate School of Engineering, Osaka University since 2015. Her research interests include the application of number theory into cryptography and information security. She received Young Paper Award of SCIS '93 in 1993, Notable Invention Award of the Science and Technology Agency in 1997, the IPSJ Sakai Special Researcher Award in 2002, the Standardization Contribution Award in 2003, Engineering Sciences Society: Certificate of Appreciation in 2005, the AWARD for the contribution to CULTURE of SECURITY in 2007, IPSJ/ITSCJ Project Editor Award in 2007, 2008, 2009, 2010, 2012, 2016, and the Director-General of Industrial Science and Technology Policy and Environment Bureau Award in 2007, Editorial Committee of Engineering Sciences Society: Certificate of Appreciation in 2007, DoCoMo Mobile Science Awards in 2008, Advanced Data Mining and Applications (ADMA 2010) Best Paper Award, The chief of air staff: Letter of Appreciation Award, Engineering Sciences Society: Contribution Award in 2012, Prizes for Science and Technology, The Commendation for Science and Technology by the Minister of Education, Culture, Sports, Science and Technology, International Conference on Applications and Technologies in Information Security (ATIS 2016) Best Paper Award, and The 16th IEEE Trustocm 2017 Best Paper Award, and IEICE milestone certification in 2017. She is a member of the International Association for Cryptologic Research, the Information Processing Society of Japan, and the Mathematical Society of Japan.