A Lightweight Stream Cipher WG-7 for RFID Encryption and Authentication

Yiyuan Luo*[1], Qi Chai*, Guang Gong* and Xuejia Lai†
*Department of Electrical and Computer Engineering, University of Waterloo.
Email: {yiyuan, q3chai, ggong}@uwaterloo.ca
†Department of Computer Science and Engineering, Shanghai Jiao Tong University
Email: {laix}@sjtu.edu.cn

***Abstract*—The family of WG stream ciphers has good randomness properties. In this paper, we parameterize WG-7 stream cipher for RFID tags, where the modest computation/storage capabilities and the necessity to keep their prices low present a challenging problem that goes beyond the well-studied cryptography. The rigorous security analysis of WG-7 indicates that it is secure against time/memory/data trade off attack, differential attack, algebraic attack, correlation attack and Discrete Fourier Transform (DFT) attack. Furthermore, we offer efficient implementation of WG-7 on the 4-bit microcontroller ATAM893-D and the 8-bit microcontroller ATmega8 from ATmel. The experimental results show that WG-7 outperforms most of previous proposals in terms of throughput and implementation complexity. Moreover, we propose a mutual authentication protocol based on WG-7, which provides the untraceability, resistance of tag impersonation and reader impersonation. With its verified cryptographic properties, low implementation complexity and ideal throughput, WG-7 is a promising candidate for RFID applications.**

I. Introduction

Radio Frequency Identification (RFID) is one of the most promising technologies in the field of ubiquitous computing which allows for remote identification of objects automatically. Although it offers many catching and exclusive characteristics, security risks associated are not easy to address, examples are tracking, inventorying and counterfeiting [6]. On the other hand, passive tags' modest computation/storage capabilities and the necessity to keep their prices low present a challenging problem that goes beyond the well-studied problems of cryptography.

The phenomenon that conventional ciphers such as DES or AES seems to be overkill, leads intensive studies toward designs and implementations of lightweight cipher for RFID tags. In the kingdom of block ciphers, HIGHT, mCrypton, SEA, PRESENT, KATAN, KTANTAN and Hummingbird are proposed, analyzed and implemented. (The survey of them can be found in [7], [6].) However, except eSTREAM candidates Grain and Trivium, lightweight stream ciphers does not receive much attention to date. For the existing lightweight stream ciphers, the design of Grain heavily relies on Nonlinear Feedback Shift Registers (NFSRs) for which the theoretic exploration is still ongoing. Therefore, the security depends on the difficulty of analyzing the design itself (i.e., the unknown period/linear complexity/autocorrelation). Trivium is designed to explore how far a stream cipher can be simplified without sacrificing its security and speed. However, several attacks [15] already come close, which could recover the internal state as long as less than the standard number of rounds of initialization is used.

In this paper, we parameterize a lightweight stream cipher WG-7 for RFID tags, which is a variant of the WG stream cipher [17] as submitted to the eSTREAM. WG-7 is a synchronous stream cipher that includes a 23-stage Linear Feedback Shift Register (LFSR) with each stage over the finite field $\mathbb{F}_{2^7}$ and a nonlinear filtering function which is realized by Welch-Gong (WG) transformation [10]. The keysize of WG-7 is 80 bits while the IV is specified to be 81 bits. The special property, namely ideal two-level autocorrelation, offered by WG-7 provides tamper resistance and resistance to side-channel attacks, since the power spectral density of the keystream sequence is flat. Besides, this property can also be used to perform tag/reader anti-collision in RFID systems at the signal-transmission level for free which is out of the scope of this paper. The security analysis shows that WG-7 is secure against time/memory/data trade off attack, differential attack, algebraic attack, correlation attack and Discrete Fourier Transform (DFT) attack. To provide fair bases for the comparison with other lightweight candidates implemented by practitioners on different platforms, WG-7 is implemented and synthesized in Atmel's 4- and 8-bit microcontrollers. Our experimental results show that this newly-proposed lightweight cipher achieves approximately 5, 3, 22 times in throughput with even smaller memory size when compared with the state-of-the-art lightweight ciphers PRESENT, speed optimized Hummingbird and Grain on the same platforms, respectively.

Moreover, we propose a mutual authentication protocol based on WG-7, which provides the untraceability, resistance to tag impersonation, reader impersonation and denial of service attack. With this protocol, the adversary cannot get enough consecutive keystream bits for each

[1] Yiyuan Luo is a visiting Ph.D student at University of Waterloo from Shanghai Jiao Tong University from September 2009 to August 2010.

978-1-4244-5638-3/10/$26.00 ©2010 IEEE

IV, which makes WG-7 secure against linear span attack.

The rest of this paper is organized as follows. In Section 2, the design of the WG-7 stream cipher is presented after the introduction of basic terms and notations. In Section 3, we give the detailed security analysis of WG-7 in terms of different attacks. In Section 4, the implementations are considered. In Section 5, a mutual authentication protocol based on WG-7 with the achieved security properties are given. Section 6 concludes the paper.

II. Description of WG-7

In the following we will introduce some basic terms and notations which will be used in the rest of this paper.

- $\mathbb{F}_q = GF(q)$, a finite field with q elements; $\mathbb{F}_q^*$, the multiplication group of $\mathbb{F}_q$; $Tr(x) = x + x^2 + \ldots + x^{2^{n-1}}$, the trace function from $\mathbb{F}_{2^n}$ to $\mathbb{F}_2$.
- $\mathbb{F}_2^n = \{(x_0, x_1, \ldots, x_{n-1}) \mid x_i \in \mathbb{F}_2\}$, a vector space over $\mathbb{F}_2$ with dimension n.
- For a positive integer $r = r_0 + r_1 2 + \ldots + r_{n-1}2^{n-1}, r_i \in \mathbb{F}_2, wt(r) = |\{0 \leq i < n \mid r_i = 1\}|$ is called the Hamming weight of the integer r.
- A boolean function $f(\mathbf{x}) = f(x_1, \ldots, x_n)$ is said to be balanced if and only if for a random variable $\mathbf{y} \in \mathbb{F}_2^n$, $Pr[f(\mathbf{y}) = 0] = Pr[f(\mathbf{y}) = 1] = 1/2$.
- Let $X_1, X_2, \ldots, X_n$ be independent binary random variables with equal probability of 0 and 1. A boolean function $f(\mathbf{x}) = f(x_1, \ldots, x_n)$ is said to be tth order correlation immune, if for each subset of t variables $X_{i_1}, X_{i_2}, \ldots, X_{i_t}$, $1 < i_1 < \ldots < i_t < n$, the random variable $Z = f(X_1, X_2, \ldots, X_n)$ is statistically independent of the random vector $X_{i_1}, X_{i_2}, \ldots, X_{i_t}$. A balanced boolean function with tth order correlation immune is called a tth order resilient function.
- Let $\mathbf{a} = \{a_i\}, a_i \in \mathbb{F}_2$ be a periodic binary sequence with period v and $C_{\mathbf{a}}(\tau) = \sum_{i=0}^{v-1}(-1)^{a_i + a_{i+\tau}}, 0 \leq \tau \leq v-1$, the periodic autocorrelation function of $\mathbf{a}$. If
$$C_{\mathbf{a}} = \begin{cases} v & \text{if } \tau \equiv 0 \bmod v \ , \\ -1 & \text{otherwise,} \end{cases}$$
then $\mathbf{a}$ is said to have ideal two-level autocorrelation.
- The linear complexity, or linear span, LC, of a binary sequence is defined as the size, in bits, of the shortest LFSR required to generate that sequence.

WG-7 is a synchronous stream cipher, which usually has two steps: the key/IV initialization step and the keystream generation. WG-7 has 80-bit key and 81-bit initial value (IV).

A. Keystream Generation

As shown in Fig.1, WG-7 consists of a 23 stage linear feedback shift registers (LFSR) over $\mathbb{F}_{2^7}$ and a WG nonlinear transformation. The finite field $\mathbb{F}_{2^7}$ is defined by the primitive polynomial $g(x) = x^7 + x + 1$. The characteristic polynomial of the LFSR is primitive over $\mathbb{F}_{2^7}$ and is given by: $f(x) = x^{23} + x^{11} + \beta$, where β is a root of $g(x)$.

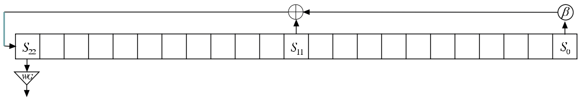

Fig. 1. Keystream Generation Diagram of WG-7.

We define $S_0, S_1 \ldots, S_{22} \in \mathbb{F}_{2^7}$ to be the internal state of the LFSR. We denote the output of the LFSR as $b_i = S_i, i = 0, 1, \ldots, 22$. Then for $i \geq 23$, we have $b_i = b_{i-12} + \beta b_{i-23}$.

The non-linear WG transformation $WG7$, $\mathbb{F}_{2^7} \to \mathbb{F}_2$, is applied to generate the keystream from the LFSR. Let $h(x) = x + x^{33} + x^{39} + x^{41} + x^{104}$ and $t(x) = h(x+1) + 1$, the original WG transform over $\mathbb{F}_{2^7}$ is defined in [10] as

$$f(x) = Tr(t(x)), x \in \mathbb{F}_{2^7}.$$

We use the cubic decimation of the original WG transform, is defined as follows. It has the same ideal two-level autocorrelation property as the original WG transform and offers better security against algebraic attack.

$$WG7(x) = f(x^3) = Tr(x^3 + x^9 + x^{21} + x^{57} + x^{87}), x \in \mathbb{F}_{2^7}.$$

B. Key/IV initialization and re-synchronization

The key/IV initialization is shown in Fig.2.

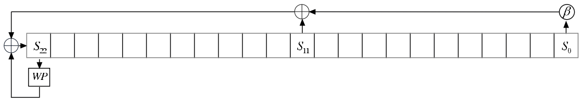

Fig. 2. Key/IV Initialization Diagram of WG-7.

The state of the LFSR is represented as S_0, S_1, …, S_{22}. Each S_i has 7 bits, and we represent the key bits as $K_{0\ldots m}$, $m < 80$ and IV bits as $IV_{0\ldots n}$, $n < 81$. The key and the IV are loaded into the LFSR as follows: For $0 \leq i \leq 10$, $S_{2i} = (K_{7i,7i+1,7i+2,7i+3}, IV_{7i,7i+1,7i+2})$, $S_{2i+1} = (K_{7i+4,7i+5,7i+6}, IV_{7i+3,7i+4,7i+5,7i+6})$ and $S_{22} = (K_{77,78,79}, IV_{77,78,79,80})$.

Once the LFSR has been loaded with the key and IV, it runs for 46 clock cycles with a nonlinear permutation feedback, WP. If the outputs of LFSR are denoted by $b_i = S_i, i = 0, 1, \ldots, 22$, then for $23 \leq i \leq 68$, $b_i = WP(b_{i-1}) + b_{i-12} + \beta b_{i-23}$. After the key initialization step, the LFSR is clocked once and the 1 bit output of the WG transform gives the first bit of the keystream.

The WP transformation is a permutation over $\mathbb{F}_{2^7}$ generated by $g(x)$ and is defined as $WP(x) = t(x^3), x \in \mathbb{F}_{2^7}$.

C. Resiliency basis for WG-7

The boolean function that corresponds to the $WG7$ transformation depends on the basis used for the computation over $\mathbb{F}_{2^7}$. The polynomial basis over $\mathbb{F}_{2^7}$ defined

978-1-4244-5638-3/10/$26.00 ©2010 IEEE

by $g(x) = x^7 + x + 1$ is $(1, \alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^6)$ where $\alpha^7 + \alpha + 1 = 0$. We found the resiliency basis for the $WG7$ transformation by a modified version of the algorithm in [10]. The new resiliency basis is $(\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6) = (1, \alpha, \alpha^2, \alpha^3, \alpha^4, \alpha^5, \alpha^6) \cdot A^{-1}$, where

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

Under the basis $(\beta_0, \beta_1, \beta_2, \beta_3, \beta_4, \beta_5, \beta_6)$, it can be verified the boolean function $WG7(x_0, x_1, x_2, x_3, x_4, x_5, x_6)$ has the 1-order resiliency.

D. Mathematical Properties of WG-7

The WG-7 stream cipher has following good cryptographic properties:

1) **The long period property**. The period of the keystream of WG-7 can be determined by the period of the LFSR, which is $2^{161} - 1$.
2) **The balance property**. The number of 0s is only one less than the number of 1s in one period of the keystream.
3) **First order resiliency property**. With the basis defined above, the boolean function $WG7$ has the first order resiliency property.
4) **Ideal two-level autocorrelation property**. It is proved that the WG transform is an orthogonal transform; thus, by applying the WG transform to the output of LFSR, the result sequence is an ideal two-level autocorrelation sequence [10]. It can be verified that any decimation of the WG sequence over $\mathbb{F}_{2^7}$ also has the ideal two-level autocorrelation. When using the cubic decimation WG transform to the LFSR, the result sequence also has the ideal two-level autocorrelation property.
5) **Acceptable linear complexity for lightweight cryptography**. The keystream of this generator can be regarded as a composition of WG transform and an m-sequence. The linear complexity of the keystream can be determined exactly. Since $WG7(x) = Tr(x^3 + x^9 + x^{21} + x^{57} + x^{87}), x \in \mathbb{F}_{2^7}$, the index set $I = \{3, 9, 21, 57, 87\}$ and l denote the number of internal states of the LFSR, the linear complexity of WG-7 is $LC = n \times \sum_{i \in I} l^{wt(i)} = 7 \times \sum_{i \in I} 23^{wt(i)} \approx 2^{25.5}$.

III. Security of WG-7

The security of WG-7 is based on the assumption that if the adversary cannot obtain 2^{24} consecutive keystream bits, then the best attack of WG-7 is the exhaustive search. In this section, we show that WG-7 is secure against correlation attack, differential attack, cube attack, algebraic attack and Discrete Fourier Transform attack.

A. Correlation Attacks on WG-7

It is well known that there are mainly two correlation attacks on synchronous stream ciphers. One type is the correlations between the keystream bits and the output of the LFSRs. The other type is the correlations between the keystream bits themselves. The first attack, discussed in [22], is especially applicable to combining generators. The $WG7$ transformation used in WG-7 is first order resilient, i.e., the output of the $WG7$ transformation or the keystream is not correlated to any single input bit of the LFSR output. This suggests that WG-7 is secure against this kind of correlation attack.

However, the case where $WG7$ transformation is approximated by linear boolean functions must also be considered. The nonlinearity of a boolean function $f(x_0, x_1, \ldots, x_{n-1})$, denoted by N_f, is defined as $N_f = \min_{a \in A} d(f, a)$ where $a(x_0, x_1, \ldots, x_{n-1})$ is an affine boolean function with n variables, and A is the set consisting of all affine boolean functions with n variables and $d(f, a) = | \{\mathbf{x} \in \mathbb{F}_2^n \mid f(\mathbf{x}) \neq a(\mathbf{x})\} |$. The nonlinearity of a boolean function denotes the minimum distance from all affine boolean functions. A high nonlinearity of boolean function can prevent the stream cipher against various types of correlation attacks. The nonlinearity of $WG7$ is 52.

In the fast correlation attack [2], the keystream is regarded as a noisy version of the LFSR output. This reduces the problem of finding the internal stage of the LFSR to a decoding problem where the keystream is the received codeword and LFSR internal state is the original message. These linear approximations can be used to derive a generator matrix of a linear code. The decoding can then be performed by a Maximum Likelihood (ML) decoding algorithm to recover the internal state of the LFSR. We use some of the theoretical bounds given in [2] to estimate the complexity of these attacks on the WG-7 cipher. Let g be a linear function with minimum hamming distance of $WG7$. Then $P[WG7(x) = g(x)] = \frac{2^7 - N_{WG7}}{2^7} = 0.59375$. Using the results given in [2] with the parameter $t = 3$, the required length N of the observed sequence in the attack is given by $N \approx 1/4 \cdot (k \cdot 12\ ln2)^{1/3} \cdot \epsilon^{-2} \cdot 2^{\frac{l-k}{3}}$, and the decoding complexity is given by $C_{dec} = 2^k \cdot k \cdot \frac{2ln2}{(2\epsilon)^6}$, where $l = 161$ is the bit size of the internal state of the LFSR, $\epsilon = P[WG7(x) = l(x)] - 0.5 = 0.09375$ and k is the number of LFSR internal state bits recovered. If we choose k to be very small (i.e. $k = 3$), then the amount of the keystream required in the attack is approximately 2^{58}, which is not realistic. If we choose a large k (i.e., $k = 80$), the amount of the keystream required is approximately 2^{33}, and the complexity of the decoding phase is approximately 2^{89}, which exceeds the complexity of the key exhaustive search. If the adversary gets less keystreams, then the complexity of the decoding will become much larger. This analysis shows that the WG-7 cipher is secure against this kind of correlation attacks.

Due to the ideal two-level autocorrelation property, WG-7 is naturally resistant against the second type of attack.

B. Time-Memory-Data Trade-off Attack on WG-7

In [1], a generic time-memory-data trade-off attack on stream ciphers is proposed. The generic attack on a stream cipher with n internal state bits will cost $O(2^{n/2})$. In WG-7, the bit size of the internal state is 161, thus the expected complexity of a time-memory-data trade-off attack is not lower than $O(2^{80})$, which makes time-memory-data trade-off attack infeasible.

C. Differential Attack and Cube Attack on WG-7

The original WG stream cipher submitted to the eSTREAM project has been carefully analyzed. At that time, there was a flaw in the key/IV initialization that made one easily built a distinguisher by differential cryptanalysis. The reason is that the WP transform is applied to the first block of the internal states (the rightmost of the LFSR) and the key/IV mixing is not enough. In the later version of the WG stream cipher, the WP transform is applied to the last block of the internal states, and then differential cryptanalysis doesn't work.

The WP permutation has good differential distribution. The maximum count for $\{\Delta x = a \Rightarrow \Delta y = b\}$ in the differential distribution table of WP is 8 and provides a maximum 2^{-4} possibility for differential characteristics. In the key/IV initialization step of WG-7, the LFSR runs for 46 cycles, thus WP is applied 46 times. For $23 \leq i \leq 68$, $b_i = WP(b_{i-1}) + b_{i-12} + \beta b_{i-23}$, if we change any one bit of the LFSR, then after 11 cycles, b_{33} will be affected; after 34 cycles, any cell of the LFSR internal state will be affected. Any invariable doesn't exist after 46 cycles. Thus WG-7 is secure against the differential attack.

Cube attack was successful against reduced-round variants of Trivium [5]. The success probability of cube attack is high if the degree is low. For the WG-7 stream cipher, the degree grows quickly. After 46 cycles, the degree can be very high, which is not of the effective range of Cube Attack.

D. Algebraic attacks on WG-7

In [4], the algebraic attack is introduced. The algebraic immunity of $WG7(\mathbf{x})$ is 4, which is optimal for seven variables boolean functions. According to [4], the time complexity of the algebraic attack that recovers the internal state of the generator is approximately $7/64 \cdot \binom{161}{4}^{log_2^7} \approx 2^{66.1}$, and the data complexity is approximately $\binom{161}{4} \approx 2^{24.7}$. In [3], the fast algebraic attack is introduced. To launch the fast algebraic attack, the adversary needs more keystream bits than the original algebraic attack.

E. Discrete Fourier Transform Attack on WG-7

In [19], Ronjom and Helleseth propose a new algebraic attack on binary filtering generators. The attack uses D keystream bits with complexity $O(D)$, where D is the linear complexity of the keystream or $D = \sum_{i=1}^{d} \binom{n}{i}$ and d is the algebraic degree of the boolean function f, after a pre-computation of complexity $O(D(log_2 D)^3)$. Later in [20], they extend the attack from binary filter generator to filter generator over $\mathbb{F}_{2^m}$. The complexity of this attack is the same as the original attack on binary filtering generators. Actually, all of the attacks can be viewed as the discrete fourier transform attack (DFT attack), as explained in [9]. For WG-7, the attacker needs $2^{25.5}$ keystream bits with a complexity of $O(2^{25.5})$ after a pre-computation with a complexity of $O(2^{39.5})$. If the length of keystream obtained is less than 2^{25}, the attacker has to randomly guess $2^{25.5} - 2^{25} > 2^{23}$ unknown bits to launch the DFT attack, which is impossible in practice.

Based on the above security analysis, we can safely conclude that if the adversary cannot obtain 2^{24} consecutive keystream bits, the best attack to WG-7 is the exhaustive search.

IV. Microcontroller Implementation of WG-7

We select two platforms for the implementation, including the 4-bit MARC4 ATAM893−D microcontroller (with current consumption of 1μA – 1mA) and the 8-bit AVR microcontroller ATmega8 (the modest one in the ATmega family in terms of computation/storage capability) from Atmel. The common fact of low power consumption and relative high data throughput make them the interesting platforms for simulating passive low-cost RFID tags [18][8][24][6].

As one can see, extremely simple arithmetic and logic operations are employed in WG-7, namely 7-bitwise XOR and shift (which can be performed efficiently without much optimization). Furthermore, since all of the finite field computations are over $\mathbb{F}_{2^7}$, it is ideal to use lookup table design. To be specific, three lookup tables are pre-computed and loaded to the microcontrollers for the $WG7$ function (7-bit input and 1-bit output), the permutation WP (7-bit input and 7-bit output) and the multiplication of β (7-bit input and 7-bit output) respectively. In all, 240 bytes are desired to store the tables. To access the tables efficiently, for ATAM893−D, `ROMByte@`, which occupies 10 cycles, is executed to fetches data from ROM onto the top of the stack, whereby the 12-bit ROM address is on the Expression Stack; for ATmega8, `pgm_read_byte`, taking 2 cycles, is called to read a byte from the program space with a 16-bit address.

A comparison is illustrated in Table I between our implementations and typical implementations of other

978-1-4244-5638-3/10/$26.00 ©2010 IEEE

TABLE I
COMPARISON OF WG-7 AND OTHER LIGHTWEIGHT CIPHERS

	Cipher	Cost of Resource		Init. [cycles]	Thru.put [bits/sec]
		Code	EXP/RET		
a	PRESENT@2MHz[24]	841	25/4	230	2,297
	PRESENT@0.5MHz[24]				574
	HB@2MHz[8]	1,532	9/7	22,949	5,543
	HB@0.5MHz[8]				1,386
	WG-7@2MHz	1,097	7/4	10,084	9,852
	WG-7@0.5MHz				2,463
		Flash	SRAM		
b	AES@8MHz[16]	6,664	88	7,149	81,432
	Salsa20@8MHz[16]	3,842	258	318	83,688
	XTEA@8MHz[18]	820	0	–	51,655
	PRESENT@8MHz[6]	2,398	528	–	53,361
	Size+ HB@8MHz[6] [c]	1,308	0	14,735	34,934
	Speed+ HB@8MHz[6]	10,918	0	8,182	91,494
	GRAIN@8MHz[18]	778	20	107,366	12,966
	Trivium@8MHz[18]	424	36	775,726	12,030
	WG-7@8MHz	1,100	0	10,074	280,087

[a] 4-bit microcontroller MARC4 ATAM893−D. The resource is measured by line of codes and depth of Expression (EXP) and Return (RET) stacks.

[b] 8-bit microcontroller ATmega family. The unit of Flash/SRAM is byte.

[c] Hummingbird has two implementations, sized-optimized (denoted as size+) and speed-optimized (denoted as speed+).

ciphers on each of the chosen platforms in the public literature. Thanks to the simple structure of WG-7, the resource demanded is much less than that of the other ciphers except Grain/Trivium and XTEA. However, the aforementioned drawback of Grain/Trivium and the reported attack towards XTEA [12] make them less appealing. Furthermore, in terms of throughput [23][11], WG-7 achieves up to 5, 3, 22, 3 times in throughput compared with that of PRESENT, speed optimized Hummingbird, Grain and the compact implementation of AES, where the former three are the state-of-the-art ultra-lightweight ciphers designed specifically for RFID tags. The achieved high throughput property is particularly expected in the RFID communication since the tag needs to convey its message to the interrogator in a fraction of a millisecond.

Note that the hardware implementations of WG-7 on Altera Stratix II and ST Microelectronics 90 nm CMOS cell library (CORX90GPHVT) by Lam *et al.* [13] shows the similar results.

Based on the above evaluation, when taking both resource consumption and throughput into consideration, it is confirmed that WG-7 is suitable for RFID applications.

V. THE MUTUAL AUTHENTICATION PROTOCOL

Many authentication protocols for RFID in current literature are based on block ciphers or hash functions, whereas very few are based on stream ciphers. Here we propose an authentication protocol based on WG-7. Assume that a RFID system consists of a reader and N RFID tags, where each tag carries an 80-bit secret ID, i.e., t_i, and a unique 80-bit secret key, i.e., k_i, and both are shared with the reader. Concerning the length of t_i, an exhaustive search to find t_i is computationally infeasible. The protocol acts as follows:

1. The reader first sends a QUERY together with an 80-bit random number IV_1 to a tag.

2. Upon receiving the QUERY and IV_1, the tag generates an 80-bit random number IV_2, computing $M_1 = t_i \oplus IV_2$. The tag also uses $IV_1 \oplus IV_2$, and the 80-bit key shared with the reader to initialize the 161-bit internal state registers of WG-7, the remaining one bit is set to 1. After that, the tag outputs the 80-bit keystream M_2 and sends (M_1, M_2) to the reader.

3. The reader tries every (t_i, k_i) stored in the database, computes $IV_2' = M_1 \oplus t_i$ and initializes the LFSR with $IV_1 \oplus IV_2'$ and the key k_i and performs a brute force search to identify the tag privately. More specifically, the reader computes the first bit of the keystream, and compares it with the first bit of M_2, if the first bit does not match, the reader stops and chooses another (t_i, k_i) and go on. If a match is found for the M_2, then the tag is identified and authenticated. If no records found, the reader rejects the tag.

4. Once the tag is identified and authenticated , the reader continues to run WG-7 for 80 clock cycles, outputs 80-bit M_3, and sends M_3 to the tag.

5. When M_3 is received, the tag continues to runs WG-7 for 80 clock cycles, outputs 80-bit M_3'. If $M_3 = M_3'$, the tag accepts the reader. The reader and tag are now mutual authenticated.

The protocol has the following privacy and security properties:

- **Tag Untraceability**. The ID of each tag is stored in the database of the reader and is assumed to be secure. Only a legitimate reader can extract a tag t_i from the pair (M_1, M_2) sent by the tag t_i. The responses of a tag are anonymous and random for unauthorized readers, thus, untraceability is realized.
- **Tag Impersonation.** To impersonate a tag t_i, the attacker must be able to compute a valid response (M_1, M_2) to a reader query. Since the attacker doesn't know (t_i, k_i), it is hard to compute such a valid pair.
- **Reader Impersonation.** A legitimate reader responds with M_3 to a tag in order to enable the tag to authenticate the reader. The attacker is unable to create a valid M_3 without knowing (t_i, k_i). Thus it is hard for the attacker to impersonate a legitimate reader.

In fact, this protocol is a challenge-response protocol. In a successful mutual authentication between the tag and the reader, the adversary can obtain at most 160 consecutive keystream bits (M_2, M_3). For a chosen IV attack, the adversary can get at most 80 keystream bits for

978-1-4244-5638-3/10/$26.00 ©2010 IEEE

each IV, thus it is impossible for the adversary to obtain 2^{24} consecutive keystream bits in this protocol. Therefore, WG-7 stream cipher can be used in this protocol.

The protocol has also been implemented on the same microcontrollers as well as the laptop (IBM Thinkpad T43 with P4 1.8, 2G RAM) to evaluate tag's and reader's performances respectively. The resulting timing of each step in the protocol is approximately 8000μs for 4-bit microcontroller and 200μs for 8-bit microcontroller. Concerning the (Application-Specific Integrated Circuit) ASIC implementation for commercial EPC tags is least 1000 faster, our protocol safely meets the timing constraints for EPC Gen2 tags. For the reader, this protocol could also been performed efficiently, i.e., the experiments show that our protocol could be executed less than 3ms when the population of tags is 10^4.

VI. Conclusions

In this paper, we show a lightweight stream cipher WG-7, which can be implemented efficiently for RFID tags while preserving all cryptographic properties of its full version. A rigorous security analysis of the WG-7 cipher indicates that unless the adversary obtains enough consecutive keystream bits, the best attack is exhaustive key search. We compare WG-7 with other ciphers in different platforms and the overall performance of WG-7 shows its feasibility for RFID applications. At last, we present the mutual authentication protocol based on WG-7, which realized the assumption that the adversary is impossible to get enough consecutive keystream bits. With the cryptographic properties and low implementation complexity, WG-7 is a novel and competitive candidate for RFID applications.

Acknowledgment

The research is supported by NSERC Discovery and Strategic Project Grant. The authors wish to thank Dr. Honggang Hu for his providing of the parameters of the LFSR of degree 23 over the extension field and many valuable and helpful discussions for this research.

References

[1] A. Biryukov and A. Shamir, Cryptanalytic time/memory/data tradeoffs for stream ciphers, *Advances in Cryptology-Asiacrypt'00*, Lecture Notes in Computer Science, vol. 1976. Springer-Verlag, 2000, pp. 1-13.

[2] V. Chepyzhov, T. Johansson and B. Smeets, A simple algorithm for fast correlation attacks on stream ciphers, *Fast Software Encryption'00*, Lecture Notes in Computer Science, vol. 1978, Springer-Verlag, 2001, pp. 181-195.

[3] N. Courtois, Fast algebraic attacks on stream ciphers with linear feedback, *Advances in Cryptology-Crypto'03*, Lecture Notes in Computer Science, vol. 2729, pp. 176-194, Springer-Verlag, 2003.

[4] N. Courtois and W. Meier, Algebraic attacks on stream ciphers with lienar feedback, *Advances in Cryptology-Eurocrypt'03*, Lecture Notes in Computer Science, vol. 2656, pp. 345-359, Springer, 2003.

[5] I. Dinur, A. Shamir, Cube attacks on tweakable black box polynomials. *Advances in Cryptology-Eurocrypt'09*. Lecture Notes in Computer Science, vol. 5479, pp. 278C299. Springer, 2009.

[6] D. Engels, X. Fan, G. Gong, H. Hu, and E. M. Smith, Hummingbird: ultra-lightweight cryptography for resource-constrained devices, To appear in the Proceedings of the 14th International Conference on Financial Cryptography and Data Security-FC 2010, January 25-28, 2010, Tenerife, Canary Islands, Spain.

[7] T. Eisenbarth, S. Kumar, C. Paar, A. Poschmann, and L. Uhsadel, A Survey of Lightweight-Cryptography Implementations, *IEEE Design & Test of Computers*, vol. 24, no. 6, pp. 522-533, 2007.

[8] X. Fan, H. Hu, G. Gong, E. M. Smith and D. Engels, Lightweight implementation of Hummingbird cryptographic algorithm on 4-Bit Microcontrollers, *Proceedings of the 1st International Workshop on RFID Security and Cryptography 2009 (RISC'09)*, pp. 838-844, 2009.

[9] G. Gong, S. Ronjom, T. Helleseth and H. Hu. *Fast Linear Subspace Attacks on Stream Ciphers*, CACR Technical Report. Avaiable at http://www.cacr.math.uwaterloo.ca/techreports/2009/cacr2009-04.pdf.

[10] G. Gong and A. Youssef, Cryptographic properties of the Welch-Gong transformation sequence generators, *IEEE Transactions on Information Theory*, vol. 48, no. 11, pp. 2837-2846, 2002.

[11] T. Good and M. Benaissa , AES on FPGA from the Fastest to the Smallest, *Lecture Notes in Computer Science*, vol. 3659, pp. 427-440, 2005.

[12] Y. Ko, S. Hong, W. Lee, S. Lee and J.S. Kang, Related key differential attacks on 27 rounds of XTEA and full-round GOST, *Fast Software Encryption'04*, Lecture Notes in Computer Science, vol 3017, Springer-Verlag, 2004, pp. 299-316.

[13] C.H. Lam, M. Argaard and G. Gong, Hardware Implementation of MOWG (preprint).

[14] W. Meier, E. Pasalic and C. Carlet, Algebraic attacks and decomposition of boolean functions, In *Advances in Cryptology-Eurocrypt'04*, Lecture Notes in Computer Science, vol. 3027, pp. 474-491, Springer, 2004.

[15] A. Maximov and A. Biryukov, Two trivial attacks on trivium, *Selected Areas in Cryptography*, Lecture Notes in Computer Science, vol 4876, Springer-Verlag, 2007, pp. 36-55.

[16] G. Meiser, T. Eisenbarth, K. Lemke-Rust and C. Paar, Software implementation of eSTREAM profile I ciphers on embedded 8-bit AVR microcontrollers, *Workshop Record State of the Art of Stream Ciphers (SASC 07)*, 2007. also submitted to *The eSTREAM Project*

[17] Y. Nawaz and G. Gong, WG: A family of stream ciphers with designed randomness properties, *Information Science*, vol. 178, no. 7, pp. 1903-1916, 2008.

[18] D. Otte, AVR-Crypto-Lib, http://www.das-labor.org/wiki/AVR-Crypto-Lib/en, 2009.

[19] S. Ronjom and T. Helleseth, A new attack on the filter generator, *IEEE Transactions on Information Theory*, vol. 53, no. 5, pp. 1752-1758, 2007.

[20] S. Ronjom and T. Helleseth, Attacking the filter generator over $GF(2^m)$, *Arithmetic of Finite Fields, First International Workshop, WAIFA 2007, June 2007, Lecture Notes on Computer Science*, vol. 4547, pp. 264-275, 2007.

[21] S. Ronjom, G. Gong and T. Helleseth, *On attacks on filtering generators using linear subspace structures, Sequences, Subsequences and Consequences, Lecture Notes in Computer Science*, S.W. Golomb *et al.* (Eds.), vol. 4893, pp. 204-217, 2007.

[22] T. Siegenthaler, Correlation-immunity of nonlinear combining functions for cryptographic applications, *IEEE Transactions on Information Theory*, vol. 30, no. 5, pp. 776-780, 1984.

[23] P. Schaumont and I. Verbauwhede., Hardware/software codesign for stream ciphers, *submitted to ESTREAM Project*, 2007.

[24] M. Vogt, A. Poschmann, and C. Paar, Cryptography is Feasible on 4-Bit Microcontrollers - A Proof of Concept, *2009 IEEE International Conference on RFID*, pp. 241-248, 2009.