An Efficient Stream Cipher Algorithm for Data Encryption

Majid Bakhtiari [1] Mohd Aizaini Maarof [2]

[1] Department of Computer Science & Information Systems, University Technology Malaysia, City Campus Jalan Semarak, 54100 Kuala Lumpur, Malaysia

[2] Department of Computer Science & Information Systems, University Technology Malaysia, Skudai Johor Bahru, 81310 Malaysia

Abstract

Nowadays the data telecommunication security has been provided by most of well-known stream cipher algorithms which are already implemented in different secure protocols such as GSM, SSL, TLS, WEP, Bluetooth etc. These algorithms are A5/1, A5/2, E0 and RC4. On the other hand, these public algorithms already faced to serious security weakness such that they do not provide enough security of proportional plain data in front of cryptanalysis attacks. In this paper we proposed an efficient stream cipher algorithm which generates 23 random bits in each round of processing by parallel random number generator and 115 bits of Initial Vector. This algorithm can implement in high speed communication link more than 100Mb/s and it has passed all of standard cryptographic tests successfully, also it can resist in front of well-known attacks such as algebraic and correlation.

Keywords: *Stream Ciphers, GSM, SSL, WEP, A5/1, A5/2, E0, data telecommunication, cryptanalysis attacks.*

1. Introduction

Stream cipher algorithms are being used in a wide range of information processing applications. This kind of cryptography is symmetric encryption primitives which are widely applied for providing the confidentiality of different networks. Currently, the public communication security is supported by well-known secure protocols such as GSM, WEP, SSL, TLS, Bluetooth, etc. These protocols are supported by four stream cipher algorithms which are A5/x in GSM networks [1] , E0 in Bluetooth standard [2] and RC4 in SSL, TLS and WEP (802.11 wireless LAN standard) [3]. On the other hand, there are many practical attacks discovered on all mentioned encryption algorithms [4-6].

Stream ciphers are always faster than block ciphers but due to the nature of random number generators which have been used in well-known stream ciphers, there are confronting with many threatening problems that permits unauthorized persons to easily access on public privacy. On the other hand, it is impossible to have infinite state random number generator to generate a truly random sequence, since the finiteness forces the random sequence to be periodic. Therefore, the best that can do is using very long period sequences that called pseudo-random sequences.

With consider that all of linear random number generators are not enough strong in front of algebraic and correlation attacks, it is necessary to notice that the linear part of algorithm should isolate from the output part of algorithm which generates key stream. However, some encryption algorithms are not implemented that part of algorithm like as A5/x, E0, RC4. Currently, the mentioned algorithms have tried to solve the linearity weaknesses by applying nonlinear clocking to those cryptosystems. As a result those algorithms cannot resist in front of algebraic and correlation attacks. It should notice that A5/1, A5/2 and E0 do not completely protect the linear part of random generator which is threatening stream ciphers.

Basically, stream ciphers should have one part as internal state and some of update function to update internal state for each round of process. The internal state, mostly initialized by secret key and initial vector (IV) key, then generates long key-stream, that known as a pseudo-random sequence. The internal state must be located behind of output part of the algorithm that generates random sequence for plain data engagement. This consideration is an important subject that some of current stream ciphers do not follow. The non-compliance with this rule causes to generate serious security problems for those stream ciphers which do not follow up like as A5/x, E0 and RC4 faced to. However, the Boolean functions in the output part of random generators must have good non linearity properties in order to resist in front of many cryptanalysis attacks such as algebraic, correlation and known IV attacks [7].

Another problem that the most of stream ciphers faced to is that each of them generates just one random bit in each round of process as the output stream of cryptosystem. This feature increases the risk of algebraic and

correlation attacks against those cryptosystems. In this paper, an efficient stream cipher algorithm designed in such a way that can generate 115 random bits in one round of process. This feature increases the resistance in front of Berlekamp-Massey, algebraic and correlation attacks.

In this paper, three public stream ciphers algorithms (A5/1, A5/2 and E0) are explained briefly, then a new stream cipher algorithm has designed in two sections as Parallel Random Number Generator and Read Out Combiner Function. The designed algorithm can generate 23 random bits in a round of processing, also it can resist in front of algebraic and correlation attacks. The designed algorithm can easily implement by software and hardware. This algorithm has passed successfully all of important cryptographic tests that mentioned in National Institute of Standards and Technology (NIST).

2. Background

Some of the most popular stream cipher algorithms which now cover more than 80% of the world of telecommunication and cyber space are A5/1, A5/2, E0 and RC4. These algorithms are weak in front of different kinds of cryptanalysis attacks such as correlation and algebraic attacks and etc. In this section their structure of those algorithms which are working on the base of LFSRs briefly explained.

2.1 A5/1 Stream Cipher Algorithm

The A5/1 is one of the stream cipher algorithm that currently is using by the most countries around the world in order to ensure privacy of conversations on GSM mobile phones. The A5/1 consists of 3 shift registers named R1, R2 and R3 with method of majority clocking as shown in Figure 1. The initialization of registers will be done by 64-bit K_c and 22-bit frame number which these are first shifted into the left side of all 3 registers and XORed with the feedbacks. Then A5/1 is clocked by using the majority clocking for 100 cycles to initial mix the bits. Then, the next 114-bits of output from A5/1 is XORed with the plain-text to encrypt/decrypt.

There are several kinds of attacks are listed on A5/1 in section 1 in [8]. One of them is the method of Biryukov [9]. He found a known-key stream attack on A5/1 requiring about two second of the key stream and recovers K_c in a few minutes on a personal computer. The second one is the method of Barkan [8]. He proposed a cipher-text-only attack on A5/1 that can recover K_c by using only four frames of cipher text.

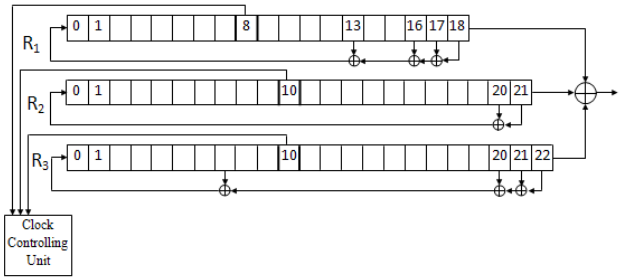

Figure 1: The A5/1 stream cipher algorithm.

Alex Biryukov, Adi Shamir and David Wagner presented that it is possible to find the A5/1 key in less than a second on a single PC, by analyzing the output of the A5/1 algorithm in the first two minutes of the conversation [9]. It is because the output key stream generated by just linear function named XOR.

In 2008, Timo Gendrulis and his team presented a guess-and-determine attack on the A5/1 stream cipher by running on the special-purpose hardware device named COPACOBANA [10]. It reveals the internal state of the cipher in less than 6 hours on average needing only 64 bits of known key stream [11].

2.2 A5/2 Stream Cipher Algorithm

The A5/2 is the 2nd stream cipher algorithm that currently support by GSM protocol in many countries. In 2006 Elad Barkan, Eli Biham and Nathan Keller demonstrated attacks against A5/1 and A5/2, that allow attackers to tap GSM mobile phone conversations and decrypt them either in real-time, or at any later time. The protocol weaknesses of GSM allow to recovery of the secret key. According to survey on the attacks against A5/2 stream cipher algorithm, it has been determined that exist linear relations among the output sequence bits and the vast majority of the unknown output bits can be reconstructed. Furthermore, some researcher have shown the time complexity of the attack is proportional to 2^{17} [12]. While according on GSM declaration the complexity of A5/2 should be 2^{64}.

In 2007 Ian Goldberg and David Wagner of the University of California at Berkeley published an analysis of the weaker A5/2 algorithm showing a work factor of 2^{16}, or approximately 10 milliseconds. Elad Barkhan, Eli Biham and Nathan Keller of Technion, the Israel Institute of Technology, have presented a cipher-text-only attack against A5/2 that requires only a few dozen milliseconds of encrypted off-the-air traffic. They also described new attacks against A5/1 and A5/3 [13].

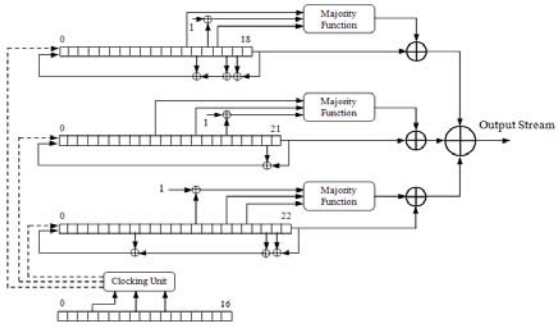

Figure 2: The A5/2 stream cipher algorithm.

With point of comparison to the previous algorithms, the description of KASUMI is public and based on the block cipher MISTY. However, the A5/3 algorithm is so far believed to be stronger than A5/1 and A5/2 but an attack successfully has done by Biham [8]. He presented that the key could be found faster than exhaustive key search [1].

2.3 E0 Stream Cipher Algorithm

Bluetooth protocol is an open standard for short-range digital radio. The goal of Bluetooth is to connect devices (PDAs, cell, phones, printers, faxes, etc.) together wirelessly in a small environment such as an office or home. The Bluetooth has three different encryption modes to support the confidentiality service as follows:

Mode 1: No encryption is performed on any data.
Mode 2: Broadcast traffic is not encrypted, but the individually addressed traffic is encrypted according to the individual link keys.
Mode 3: All traffic is encrypted according to the master link key.

Bluetooth is working on the base of E0 algorithm. Until now, there are many known attacks on the encryption scheme E0 are available that can threaten the security of Bluetooth. The most well-known of them are algebraic attacks [14] and correlation attacks [15-16].

E0 generates a bit using four shift registers with differing lengths (25, 31, 33, 39 bits). The Figure 3 shows the involved algorithm use in the Bluetooth standard.

However, in E0 like A5/1 and A5/2, the last function that generates key stream is simple XOR. Due to the linear properties of XOR, the output key stream has linear relation with its inputs that it may threaten the whole of algorithm.

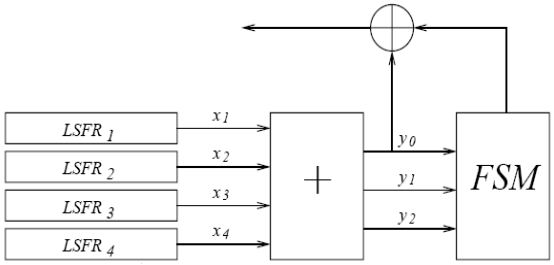

Figure 3: The encryption algorithm used in Bluetooth.

3. New Stream Cipher Algorithm

One of the important parameter that A5/1, A5/2 and E0 suffering from is using from XOR at the last section of algorithm to generate key stream. This method of random bit generation has caused that those algorithm faced to big security problems. In this paper, a new algorithm has designed in such a way that key stream can resist in front of correlation and algebraic attacks.

3.1 Parallel Random Number Generator

Linear feedback shift registers (LFSRs) are very applicable in parallel random number generators. Due to the simplicity of implementing the LFSRs in structure of hardware and software, LFSRs are use in many of random number generators. LFSRs can generate different sequences with good statistical properties and large length of period. With notice that, the equation of polynomial feedback plays very important role in LFSRs. If the feedback polynomial equation is primitive, it is means that an LFSR with length n can generate maximal length of sequence equal to $2^n - 1$. Furthermore, due to feature of output linearity stream, the output sequences of LFSR are easily expectable and if the designers want to trap more than one stream as outputs of sequence, each bit is exactly equal to others bits with time delay (maximum delay is length of LFSR-bit or n). This problem is threatening the system from different viewpoint especially from correlation attack. In this paper, one model of LFSR has designed in such a way that can solve this big problem; hence it is important in cryptography.

Designing a parallel random number generator (PRNG) by using one LFSR has the feature that one can construct a linear sequential system which is correctly initialized and for each clock cycle generates different consecutive stream of the sequences, while the normal LFSR would generate just one stream sequence. In fact, each bit-output of the finite state machine can be XORed together to form the key-stream output.

LFSRs are defined by characteristic polynomials which determine all properties of the sequences produced by an LFSR. Parallel Random Number Generators (PRNGs) are defined by very specific polynomials. The most properties of this kind of generators are that those have been used in practice and at large scale of encryption in symmetric cryptography. The estimation of the number of primitive polynomials in PRNGs related to LFSRs can be calculated from Eq. (1), where v is the number of sub-registers.

$$P(x) = \frac{2^{1+2\left\lfloor \frac{n}{v} \right\rfloor} . \varphi(2^n - 1)}{n.2^n} \quad (1)$$

This class of primitive polynomials is very strong properties on the parallel implementation of an LFSR. The basic idea of a parallel generator consists in generating the sub-sequences of a given sequence in parallel. However, this is a basic technique for taking advantage of parallel computer.

Let $S = (S_0, S_1, S_2, \dots)$ be an unlimited binary sequence with period T, thus $S_j \in \{0,1\}$ and $S_{j+T} = S_j$ or all $j \geq 0$. For a given integer d, a v-decimation of S is the set of sub-sequences defined in Eq. (2).

$$S_v^i = \left(S_i, S_{i+v}, S_{i+2v}, \dots, S_{i+jv}, \dots\right) \quad (2)$$

where $i \in \{0, d-1\}$ and $j = 0,1,2, \dots$. Consequently, the sequence S is completely described by the sub-sequences as follows:

$$\begin{aligned} S_v^0 &= (S_0, S_v, \dots) \\ S_v^1 &= (S_1, S_{v+1}, \dots) \\ S_v^2 &= (S_2, S_{v+2}, \dots) \\ &\vdots \\ S_v^{v-2} &= (S_{v-2}, S_{2v-2}, \dots) \\ S_v^{v-1} &= (S_{v-1}, S_{2v-1}, \dots) \end{aligned}$$

Usually the strong random generators are structured by combining of more than one LFSR which are working together with different methods to provide non-linearity in output stream. This paper do not follows the classical methods of generation such as n-sequences that explained by Colomb. Mostly, the classical methods for generating n-sequences is using a primitive polynomial to select those taps of an n-cell shift register which, if their contents are added modulo 2 and the summation used as input to the shift register, will result in a cycle length of $2^n - 1$ steps. In this paper, the special combination of LFSR has designed in such a way that each output traps are not equal to others outputs. With consider that, this paper need 115 separated random sequences with maximum period of length which should be isolated from each others. This paper designed a LFSR in such a way that can provide 115 separated random stream sequences. On the other hand, the speed of processing is important parameters that it is possible to implement designed PRNG by software and hardware to obtain suitable speed of processing. In this regard, the Eq. (3) initially designed as a primitive candidate polynomial equation that can be satisfied the form of Eq. (4).

$$\begin{aligned} P(x) = {} & x^{257}+x^{254}+x^{251}+x^{249}+x^{244}+x^{243}+x^{242}+x^{238}+ \\ & x^{237}+x^{233}+x^{232}+x^{230}+x^{228}+x^{226}+x^{225}+x^{221}+x^{220}+ \\ & x^{218}+x^{216}+x^{214}+x^{213}+x^{209}+x^{208}+x^{204}+x^{203}+x^{202}+ \\ & x^{197}+x^{195}+x^{192}+x^{190}+x^{187}+x^{185}+x^{180}+x^{179}+x^{178}+ \\ & x^{174}+x^{173}+x^{169}+x^{168}+x^{166}+x^{164}+x^{162}+x^{161}+x^{157}+ \\ & x^{156}+x^{154}+x^{152}+x^{150}+x^{149}+x^{145}+x^{144}+x^{140}+x^{139}+ \\ & x^{138}+x^{133}+x^{131}+x^{128}+x^{126}+x^{123}+x^{121}+x^{116}+x^{115}+ \\ & x^{114}+x^{110}+x^{109}+x^{105}+x^{104}+x^{102}+x^{100}+x^{98}+x^{97}+ \\ & x^{93}+x^{92}+x^{90}+x^{88}+x^{86}+x^{85}+x^{81}+x^{80}+x^{76}+x^{75}+x^{74}+ \\ & x^{69}+x^{67}+x^{64}+x^{62}+x^{59}+x^{57}+x^{52}+x^{51}+x^{50}+x^{46}+x^{45}+ \\ & x^{41}+x^{40}+x^{38}+x^{36}+x^{34}+x^{33}+x^{29}+x^{28}+x^{26}+x^{24}+x^{22}+ \\ & x^{21}+x^{17}+x^{16}+x^{12}+x^{11}+x^{10}+x^{5}+x^{3}+1 \end{aligned} \quad (3)$$

According to the output of parallel random number generator by LFSRs which should be on the base of Eq. (4), after analyzing the Eq. (3), we find that it is equal to Eq. (5). So, both of equation are equal to each other from period of length and sequence bit-randomness point of view.

$$P(x) = x^n + (x^m + 1)^k * (x^j + 1)^l * (x^i + 1)^y \; ; \; \left((m * k) + (j * l) + (i * y)\right) < n \quad (4)$$

$$P(x) = x^{257} + (x^2 + 1)^{100}(x^3 + 1)^{13}(x^5 + 1)^3 \quad (5)$$

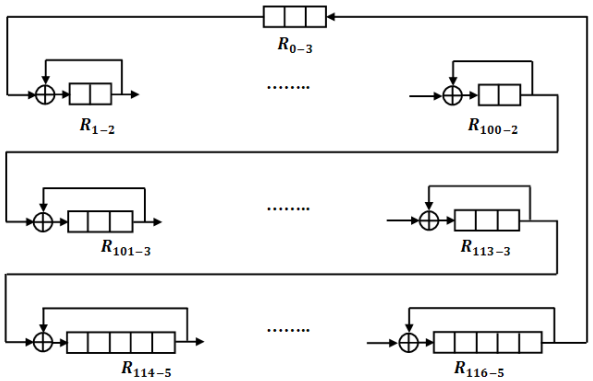

Figure 4: Diagram of Parallel Random Number Generator.

On the other hand, we need 23 random bit-streams that should be isolated from each other as the output of PRNG. The Eq. (5) has potential of parallel random bit-stream generation and it can generate 116 isolated bit-stream sequences. The period length of each sequence is equal to $2^{257} - 1$. It is because the characteristic polynomial equation of diagram that has shown in Figure 4 is primitive. In this regard, we have simulated Eq. (5) in

Figure 4. In fact, the diagram that shown in Figure 4 is equal to Eq. (5) or Eq. (3). Therefore, we can select just 115 traps as bit-stream output from Figure 4. However, it is advisable to implement a multiplexor for sequence selection to increase the nonlinearity of each stream. Finally, each sequence can be selected according to Eq. (6) as follows:

$$S_v^{R_i} = \left(S_v^{R_i}, S_{v+1}^{R_i}, \dots\right) \;;\; 1 \le i \le 31 \;;\; 1 \ge v \ge 3$$
$$S_w^{R_j} = \left(S_w^{R_j}, S_{w+1}^{R_j}, \dots\right) \;;\; 1 \le j \le 23 \;;\; 1 \ge w \ge 5$$
$$S_u^{R_k} = \left(S_u^{R_k}, S_{u+1}^{R_k}, \dots\right) \;;\; 1 \le k \le 5 \;;\; 1 \ge u \ge 7 \quad (6)$$

It is important to notice that if someone wants to implement one multiplexer to increase the degree of nonlinearity of stream sequence. It should apply on parameters of i, j, k in Eq. (6).

3.2 Read Out Combiner Function

The read out combiner as an important part of algorithm, plays very critical role in front of different kinds of attacks. This part is first part of algorithm that located in front of cryptanalyst. This part of algorithm designed in such a way that can resist in faced to strong methods of attacking such as correlation and algebraic attacks.

The first step is designing the truth table of read out combiner function. The truth table should be designed in such a way that can satisfy all of features related to cryptography point of view such balanced-ness, correlation immunity, algebraic degree and non-linearity. The truth table of this part of algorithm has shown in Appendix as Table 1 and the proportional Boolean function Eq. (7) is as follows:

$$\begin{aligned} f(x) = {} & \bar{A}\bar{B}\bar{C}\bar{D}\bar{E} + \bar{A}\bar{B}\bar{C}\bar{D}E + \bar{A}\bar{B}\bar{C}DE + \bar{A}\bar{B}C\bar{D}E + \\ & \bar{A}\bar{B}C\bar{D}\bar{E} + \bar{A}\bar{B}CDE + \bar{A}BC\bar{D}\bar{E} + \bar{A}BCDE + \\ & A\bar{B}\bar{C}\bar{D}\bar{E} + A\bar{B}\bar{C}D\bar{E} + A\bar{B}C\bar{D}E + A\bar{B}CD\bar{E} + \\ & A\bar{B}\bar{C}DE + ABC\bar{D}\bar{E} + ABC\bar{D}E + ABED\bar{E} \end{aligned} \quad (7)$$

After designing the truth table as an output of Boolean function, the characteristic of designed table has simplified as an equation that has shown in Eq. (8).

$$\begin{aligned} f(x) = {} & \bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}E(C \oplus D) + \bar{A}B\bar{C}D + \\ & \bar{A}BC\left(\overline{D \oplus E}\right) + A\bar{B}\bar{C}\bar{E} + AC(D \oplus E) + \\ & AB\bar{C}DE + ABC\bar{D}\bar{E} \end{aligned} \quad (8)$$

It should be notice that in this paper, the read out combiner consists of 23 functions that all of them are same together but all of their inputs are different from each others. Each function has five separate inputs from PRNG. Therefore, 115-bits stream from PNRG after XOR with Initial Vector (IV), feed to functions of read out combiner to generate 23 bits as output of algorithm. So, each bit of key stream output is simplifies as Eq. (8).

For convenient explanation of Eq. (8), the Figure 5 shows the Eq. (8) as function box with totally 115-bits. In fact, the functionality of Figure 5 is exactly Eq. (8). Therefore, the functionality of $f(x)$ is a function with 5 input variables and 1-bit output that operates instead of Eq. (8). The important statistical cryptography tests have applied on $f(x)$.

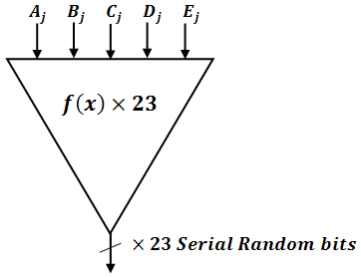

Figure 5: Read-Out Combiner Function.

3.2.1 Balance Check

With consider that, a binary sequence is called balanced if its truth table has the same number of 1's and 0's. According to Table 1 in Appendix, the function of $f(x)$ is balanced as shown in Eq. (8). The balanced-ness of a Boolean function is a significant cryptography property in the manner that the output of function should not leak any statistical information related to the crypto system.

3.2.2 Nonlinearity Check

Both non-linear and linear functions are significance for block and stream ciphers. Non-linear functions are usually used to achieve confusion, while linear functions are employed to achieve diffusion. Non-linear functions are useful in protecting a cipher system from a differential cryptanalysis, and determining the key by solving equations and so on. The non-linearity is the number of bits which must change in the truth table of a Boolean function to reach the closest affine function.

There are different kinds of nonlinearity measurement methods available. In this paper, the non-linearity of Eq. (8) has calculated from affine function and Walsh spectrum. Therefore, the non-linearity of the Eq. (8) with 5-variable Boolean function N_f calculated as follows:

$$N_f = 2^{n-1} - \frac{1}{2} max\left|\omega_f(a)\right| \;,\; a \in \{0,1,2,\dots 2^{n-1}\}$$
$$W_f = 16,4,4,4,-4,4,-4,0,0,0,4,0,0,4,0,0,-4,4,4,$$
$$0,0,-4,4,4,0,-4,0,0,0,0,-4,0$$

$$N_f \leq 2^{n-1} - 2^{\frac{n}{2}-1} \Longrightarrow N_f = 12 \quad (9)$$

The higher non-linearity in Boolean function means, that the designed function can protecting the cipher system in faced to some methods of attacks particularly algebraic attack. With consider that the non-linearity of designed function Eq. (8) that has shows as the read out combiner function in Figure 5, has calculated by Eq. (9). Also the maximum degree for five variables should be equal to 12, it is means that the designed function can protect the cipher system from two serious attacks such as correlation and algebraic. Currently well-known stream cipher algorithms are suffering from these kinds of attacks.

3.2.3 Correlation Immunity Check

Cryptographers care about correlation immunity because its absence in Boolean functions which are used in a cryptosystem can allow effective attacks on the system. According to the test result of designed function from probability point of view, the result for all of variables is equal to $\frac{1}{2}$. Therefore the designed function is correlation immune.

$$CI(A) = \frac{1}{2}\ , \quad CI(B) = \frac{1}{2}\ , \quad CI(C) = \frac{1}{2}\ ,$$
$$CI(D) = \frac{1}{2}\ , \quad CI(E) = \frac{1}{2} \quad (10)$$

As it has shown in Eq. (10), which have derived from Table 1 (in Appendix), the result check of correlation immunity is excellent for designed function. On the other hand, from correlation calculations point of view, we have calculated all of possibility of designed function. All of results are equal to zero. It is because the correlation coefficients have boundaries of -1 and +1. A value of +1 indicates perfect positive linear relationship between two sequences, while -1 is a perfect negative linear relationship between them. A value of zero indicates no correlation between input variables or independent. In designed function, the correlations for five variables are excellent. Therefore highly non-linear balanced Boolean function with an excellent Correlation-Immunity is enough strong in faced to correlation attack.

3.2.4 Algebraic Degree Check

The algebraic degree is one of the nonlinearity measures of Boolean function. The Boolean functions with small algebraic degree are in general considered to be less suitable for cryptographic applications than those with higher degree. However there are large classes of cryptographically strong Boolean functions with small algebraic degree such as quadratic bent functions. It is important that almost every balanced Boolean function has maximal or almost maximal algebraic degree.

The algebraic degree of Eq. (8) is equal to 4 which is the maximum level for 5 variables. Therefore, each output random bit of this function can successfully resist in faced to algebraic attack and Berlekamp-Massey attacks.

4. Practical Statistical Tests

According to National Institute of Standards and Technology (NIST), all important cryptography tests (Frequency test, Serial test, Run, Long Run test, Poker test, Auto-Correlation test, Maurer's Universal Test) have applied on designed stream cipher algorithm. All of tests passed successfully.

5. Conclusion

In this paper we introduce some weaknesses of well-known stream cipher algorithms in current industrial world which are threatening public interests in different cyber space networks. According to many sources and serious security weaknesses in well-known stream cipher algorithms which are already implemented in GSM, SSL, TLS, WEP, Bluetooth and so on, it is strongly advise not to rely on E0, A5/x and RC4 in field of data security communication.

Furthermore, an efficient designed stream cipher algorithm can be implemented in GSM, WEP, SSL, TLS and Bluetooth protocols. The new algorithm has designed base on parallel random number generator with the high speed of processing which can be implemented in high speed data/voice link of communication and it can resist in front of different kinds of attacks such as correlation and algebraic.

The designed algorithm has passed all of cryptographic tests in NIST standard successfully. The designed new algorithm can support the encryption/decryption with rate of 100 MB/s. The key variety of designed algorithm is equal to 2^{257} and the length key of IV is equal to 2^{115}. It can be implemented easily by hardware and software.

This paper designed a new stream cipher algorithm with key variety of 2^{257} and 115-bit IV that is more secure than other public one from speed of processing and others viewpoint of security.

Appendix

Table 1: Truth table of Nonlinear Function

Input variety	*Output*
00000	1
00001	1
00010	0
00011	1
00100	0
00101	1
00110	0
00111	0
01000	0
01001	0
01010	1
01011	1
01100	1
01101	0
01110	0
01111	1
10000	1
10001	0
10010	1
10011	0
10100	0
10101	1
10110	1
10111	0
11000	0
11001	0
11010	0
11011	1
11100	1
11101	1
11110	1
11111	0

References

[1] Briceno, M., I. Goldberg, and D. Wagner, *A pedagogical implementation of A5/1*. URL: http://www.scard.org/gsm/a51.html.

[2] Bluetooth, S., *Specification of the Bluetooth system.* Core, version, 2005. **1**: p. 2005-10.

[3] Rivest, R., *The RC4 Encryption Algorithm. RSA Data Security.* Inc., March, 1992. **12**.

[4] Maximov, A., T. Johansson, and S. Babbage, *An Improved Correlation Attack on A5/1*, in *Selected Areas in Cryptography*, H. Handschuh and M. Hasan, Editors. 2005, Springer Berlin / Heidelberg. p. 1-18.

[5] Stubblefield, A., J. Ioannidis, and A. Rubin, *A key recovery attack on the 802.11 b wired equivalent privacy protocol (WEP).* ACM transactions on information and system security (TISSEC), 2004. **7**(2): p. 319-332.

[6] Golić, J., V. Bagini, and G. Morgari, *Linear Cryptanalysis of Bluetooth Stream Cipher*, in *Advances in Cryptology — EUROCRYPT 2002*, L. Knudsen, Editor. 2002, Springer Berlin / Heidelberg. p. 238-255.

[7] Meier, W. and O. Staffelbach, *Nonlinearity Criteria for Cryptographic Functions*, in *Advances in Cryptology — EUROCRYPT '89*, J.-J. Quisquater and J. Vandewalle, Editors. 1990, Springer Berlin / Heidelberg. p. 549-562.

[8] Barkan, E., E. Biham, and N. Keller, *Instant Ciphertext-Only Cryptanalysis of GSM Encrypted Communication.* Journal of Cryptology, 2008. **21**(3): p. 392-429.

[9] Biryukov, A., A. Shamir, and D. Wagner, *Real Time Cryptanalysis of A5/1 on a PC*, in *Fast Software Encryption*, G. Goos, *et al.*, Editors. 2001, Springer Berlin / Heidelberg. p. 37-44.

[10] Gendrullis, T., M. Novotný, and A. Rupp, *A Real-World Attack Breaking A5/1 within Hours*, in *Cryptographic Hardware and Embedded Systems – CHES 2008*, E. Oswald and P. Rohatgi, Editors. 2008, Springer Berlin / Heidelberg. p. 266-282.

[11] Kumar, S., *et al.*, *Breaking Ciphers with COPACOBANA –A Cost-Optimized Parallel Code Breaker*, in *Cryptographic Hardware and Embedded Systems - CHES 2006*, L. Goubin and M. Matsui, Editors. 2006, Springer Berlin / Heidelberg. p. 101-118.

[12] Petrovic, S. and A. Fuster-Sabater. *An improved Cryptanalysis of the A5/2 Algorithm for Mobile Communications*. 2002.

[13] Barkan, E., E. Biham, and N. Keller, *Instant Ciphertext-Only Cryptanalysis of GSM Encrypted Communication*, in *Advances in Cryptology - CRYPTO 2003*. 2003, Springer Berlin / Heidelberg. p. 600-616.

[14] Armknecht, F. and M. Krause, *Algebraic Attacks on Combiners with Memory*, in *Advances in Cryptology - CRYPTO 2003*. 2003, Springer Berlin / Heidelberg. p. 162-175.

[15] Hermelin, M. and K. Nyberg, *Correlation Properties of the Bluetooth Combiner*, in *Information Security and Cryptology - ICISC'99*, J. Song, Editor. 2000, Springer Berlin / Heidelberg. p. 17-29.

[16] Lu, Y. and S. Vaudenay. *Faster correlation attack on Bluetooth keystream generator E0*. 2004: Springer.