# 4-7 Security Evaluation of Cryptographic Technology

**TANAKA Hidema**

Cryptography is the fundamental technology for information security. It plays in the function of confidential, authentication, signature in the various information security technologies. Since the status of security evaluation influences the reliability of information security, the security evaluation of cryptographic technologies is very important. In particular, the security evaluation of cryptographic technologies which is used in electrical government service now is requested to be executed by an impartial aspect. In addition it is necessary to estimate the cost of the attack and its feasibility. So it is appropriate that the National institute executes such research activity. In this paper, we show the outline of the security evaluation activity for symmetric ciphers of Security Fundamentals Group between 2006 and 2010.

***Keywords***
Symmetric cipher, Higher order differential attack, Side-channel attack, Fault based attack, Pseudo random number generator

## 1 Introduction

Cryptography is the fundamental technology for information security. It provides confidentiality of communication, authentication for validating communication partners, and signature functions for proving authenticity of data in various attack scenarios. These are based on cryptographic technology (cryptographic primitives) and designed from various points of view. Outside Japan, it is not unusual that these are designed not only by private companies but also by research organizations such as universities. In Japan, private companies and universities loosely share roles. Private companies play the role of technology development, and universities the role of academic development. Since evaluation as academic achievement (development of cryptanalysis methods) and development of cryptographic primitives are inextricably linked, at a glance it seems that development of cryptographic technology is completed by collaboration between private companies and universities; however, in order for the technology to be recognized as a trusted technique, it needs to be validated by a third party. This is because the security of public electronic services in Japan such as electronic government services and the basic residential registers network system rely on cryptographic technology, and if information is provided by only one side, there is a danger that vulnerabilities may remain. In addition, studying academic trends will not be sufficient. There is a gap between academic achievements and actual usage environments, and the arguments of a paper are not always appropriate. For the above reasons, security evaluation of cryptographic technologies should be carried out from a fair and neutral viewpoint, and in order to achieve this, initiatives of the public sector are indispensable.

In this paper, we show the outline of the security evaluation activity for symmetric ciphers of the Security Fundamentals Group between 2006 and 2010. The activity of the Security Fundamentals Group especially focused on the following points.

- Improvement against algebraic attacks for which the theory of proving security is immature

- Effectiveness of attack methods against cryptographic modules such as FPGA implementation
- Reduction of evaluation costs

We chose 64-bit block ciphers as the main subject of the security evaluation. Although 128-bit block ciphers are becoming the mainstream in recent years, in some cases 64-bit block ciphers are more advantageous in terms of implementability, and they are often used for familiar services such as electronic money type services as represented by Felica, and smart cards for the basic residential registers network system. Therefore, we consider it is important to estimate the period for which they can be used securely. We chose fault based attacks that use electromagnetic emanation to evaluate the security of cryptographic modules. Generally it is considered that attacks causing malfunction by electromagnetic emanation are relatively low cost, and attack methods based on this assumption have been proposed in academic conferences now and then; therefore, we decided it was necessary to verify the viability. The aforementioned evaluation of 64-bit block ciphers is closely related to evaluation of cryptographic modules. This is because migration of a cryptographic technology that has already come into wide use will require services to be stopped, and because migration will be a huge risk for the service provider since the mixture of new and old cryptographic primitives could reduce the security, and as a result the migration process will be slowed down. Therefore, 64-bit block ciphers and their cryptographic modules, which are still used and becoming popular, have a significant impact on the reliability of future electronic services. In addition, performance of smart cards and chips that provide the base for cryptographic modules has been improving rapidly, helping to improve the performance of a pseudo random number generator that is implemented on top of it. Pseudo random number generators are used for key generation and authentication protocols. For example, they are widely implemented and used for car keys (to lock/unlock car doors by radio signal). The security of a pseudo random number generator can be evaluated by long range periodicity, linear complexity and correlation immunity, and thanks to the improved performance of cryptographic modules, it is now possible to use a pseudo random number generator which is so large that it cannot be evaluated by computer. As a result, it has become difficult to verify the security, and thus new evaluation methods are required. The Security Fundamentals Group developed an algorithm that computes linear complexity by a linearization method that is based on an algebraic attack.

This paper first describes scenarios of evaluation of cryptographic technologies in **2**, as well as the presumption, purpose and validity of security evaluation of cryptography. In **3**, we outline a higher order differential attack against a 64-bit block cipher, MISTY1. In **4**, we describe experiments on fault based attacks that use electromagnetic emanation against FPGA implementations. In **5**, we introduce evaluation of linear complexity by a linearization method for pseudo random number generators, and **6** provides a summary.

## 2 Scenarios of evaluation of cryptographic technologies

The security of cryptography requires that confidential information such as keys should not be found more efficiently than by exhaustive search. Generally, cryptanalysis means to recover plaintext from ciphertext, and has the following two meanings.

1) Directly recover plaintext from ciphertext.
2) Recover the key from ciphertext to decrypt the ciphertext to plaintext.

As for 1), for example, $2^n$ of plaintext can be recovered from $n$ bits of ciphertext, and all of them are the candidates for the correct plaintext; therefore, it is only necessary to prevent the correct candidate from being distinguished from the incorrect ones. Although linguistic meanings of plaintext could have influences, since modern cryptography is based on the assumption that plaintext is binary information, it is considered to be sufficient if the plaintext is not distinguished from random numbers. As for 2), consider a problem where $Y=f(X,K)$ and $Y$ is given, and find $X$ and $K$. In this case, $(X,K)$ will be underspecified as it is. In order to solve this problem as equation, it needs to be set up as simultaneous equations to eliminate one of the variables, or either $X$ or $K$ must be given. In the former case, since the key is an invariable, it is only necessary to eliminate the key: this is equal to the scenario 1). In the latter case, it is trivial that $X$ can be decrypted if the key is given. Therefore, an appropriate scenario for security evaluation would be to provide information of plaintext to find the key. Thus, we evaluate the security by estimating the required cost for recovering a key from plaintext and the corresponding ciphertext.

The cost consists of computational cost and amount of data. For example, if no limit is set to computer capability (assume unlimited computational cost), the key can be found from a pair of plaintext and ciphertext by conducting an exhaustive search. This is the limit of the security of cryptography. So we conclude that even if a pair of plaintext and ciphertext that is convenient for the attacker is provided, it will be still secure if the key cannot be found more efficiently than it can be found by exhaustive search. There are two attack scenarios depending on how plaintext is provided.

- Known plaintext attack
- Chosen plaintext attack

A known plaintext attack is a method to carry out attacks under the condition that a pair of plaintext/ciphertext has been simply given. The only such method against block ciphers is linear cryptanalysis. As for the attack method against stream ciphers, which are symmetric ciphers that use pseudo random number generators, correlation attack and its advanced versions fall under this category. A chosen plaintext attack is a method to carry out attacks under the condition that attackers have chosen plaintext that is advantageous for them and obtained the corresponding ciphertext. For example, it is advantageous for the attackers if they have obtained ciphertext that correspond to numbers 0, 1, 2, and 3. The key point in this example is that all bits except the low two bits should be fixed to 0. There are a number of such attack methods against block ciphers, and typical examples include differential attacks and higher order differential attacks. As for stream ciphers, some attack methods have been proposed which are used when only the Initial Value (IV, publicly known parameters excluding the key) is changed, and the key is fixed.

Incidentally, the key size commonly used for symmetric ciphers is generally 128 bits. In the above-mentioned attack method, it is regarded that it is not secure if even one bit of the key has been determined. It is generally considered that it is not secure if one bit of an expanded key (an internal key used in cryptographic algorithm) generated from a 128-bit key has been determined. On the other hand, even the most advanced computer in the world can recover 60 bits of a key by exhaustive search, and therefore, there is a gap between the actual computer security and theoretical cryptanalysis. The gap between the limit of exhaustive search by computer and theoretical cryptanalysis can be seen as the margin of the security and the period in which the cryptography can be used securely.

# 3 Security evaluation of 64-bit block cipher MISTY1

## 3.1 64-bit block cipher MISTY1

64-bit block cipher MISTY1 is a Feistel type block cipher developed by Mitsubishi Electronics Corporation in 1996 (Fig. 1)[2]. The data length is 64 bits and the key length is 128 bits. Plaintext is divided as follows.

$$\mathrm{P}=(\mathrm{P_L}\,\|\,\mathrm{P_R})=(X_{15},\ldots,X_8\,\|\,X_7,\ldots,X_0)\rightarrow X_i\in\begin{cases}GF(2)^7: i=\text{even}\\ GF(2)^9: i=\text{odd}\end{cases}\qquad(1)$$

It generates an expanded key from a 128-bit key in accordance with Table 1.

$$\begin{aligned}&K=(K_7,\ldots,K_0),\ K_i\in GF(2)^{16}\\&K_i'=FI(K_i;K_{i+1})\end{aligned}\qquad(2)$$

Figure 2 shows the variables that will be used

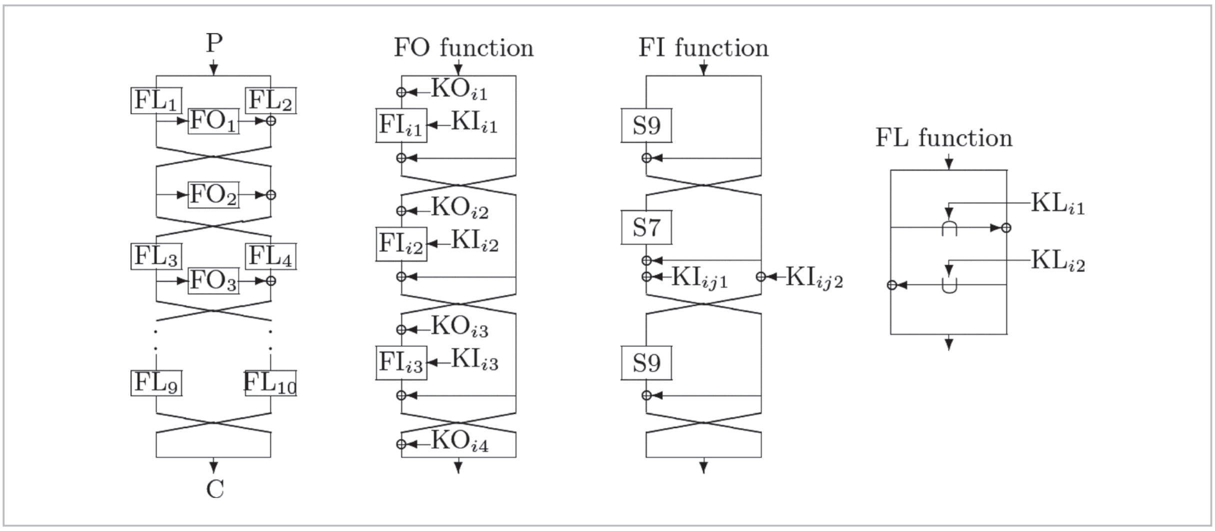


**Fig.1** 64-bit block cipher MISTY1

**Table 1** Key schedule

| Sub-key | $KO_{i1}$ | $KO_{i2}$ | $KO_{i3}$ | $KO_{i4}$ | $KI_{i1}$ | $KI_{i2}$ | $KI_{i3}$ | $KL_{i1}$ | $KL_{i2}$ |
|---|---|---|---|---|---|---|---|---|---|
| Secret key sub-block | $K_i$ | $K_{i+2}$ | $K_{i+7}$ | $K_{i+4}$ | $K'_{i+5}$ | $K'_{i+1}$ | $K'_{i+3}$ | $K_i$ (odd $i$)<br>$K'_{i+1}$ (even $i$) | $K'_{i+6}$ (odd $i$)<br>$K_{i+3}$ (even $i$) |

below.

The security of MISTY1 is characterized as providing provable security against differential attacks[3] and linear attacks[4] by performing only three rounds of FO function. In addition, low algebraic degree is chosen for the S-box (substitution table generated by non-linear function) of S7 and S9 in order to realize compact hardware implementation, and the degree of S7 is 3 and the degree of S9 is 2. High security and implementation performance has been achieved in this way, and it has been adopted as the standard 64-bit block cipher for the international standard ISO[5], CRYPTREC (Japan's e-Government recommended ciphers)[6], and NESSIE (cryptographic technology evaluation project in Europe)[7].

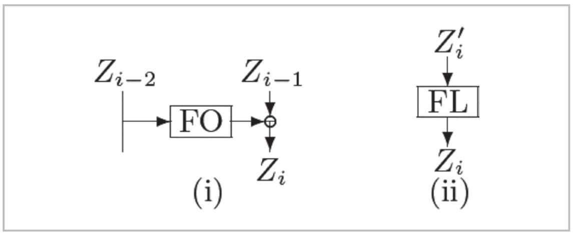


**Fig.2** Locations of variables

As described earlier, the low algebraic degree of the S-box contributes largely toward maintaining high implementation performance while providing provable security against differential attacks and linear attacks. On the other hand, the low algebraic degree could compromise the security against algebraic attacks. The purpose of our evaluation is to confirm the security against algebraic attacks. Algebraic attacks include higher order differential attacks, interpolation attacks, and integral attacks, and we chose higher order differential attacks in our research. This is because higher order differential attacks are the most basic method among algebraic attacks, and potentially the evaluation can be applied to various cases.

### 3.2 Higher order differential attack

Let $F(X;K)$ be a function of $GF(2)^n \times GF(2)^s \rightarrow GF(2)^n$.

$$\begin{aligned} &Y = F(X;K) \\ &X \in GF(2)^n,\ Y \in GF(2)^n,\ K \in GF(2)^s \end{aligned} \tag{3}$$

Let $(a_0, a_1, .., a_{N-1})$ denote linear independent vectors of $GF(2)^n$, and let $V_{[a_0, a_1, .., a_{N-1}]}$ denote the subspace spanned by them. Here, if $\Delta^{(N)}_{V_{[a_0,a_1,...,a_{N-1}]}}$ is the $N$-th order differential of $X$ of $F(X;K)$, it is computed as follows.

$$\Delta^{(N)}_{V_{[a_0,a_1,...,a_{N-1}]}} F(X;K) = \sum_{A \in V_{[a_0,a_1,...,a_{N-1}]}} F(X+A;K) \tag{4}$$

Hereafter when $V_{[a_0, a_1, .., a_{N-1}]}$ is known, $\Delta^{(N)}_{V_{[a_0,a_1,...,a_{N-1}]}}$ is abbreviated as $\Delta^{(N)}$. If $\deg_X\{F(X;K)\} = d$, the following property is satisfied.

**Property 1:**

$$\deg_X\{F(X;K)\} = d \rightarrow \begin{cases} \Delta^{(d+1)} F(X;K) = 0 \\ \Delta^{(d)} F(X;K) = \text{const.} \end{cases} \tag{5}$$

Figure 3 shows the final round of $r$-round Feistel type block cipher. $H^{(r)}(X)$ denotes output from the $(r-2)$th round, and is computed as follows,

$$H^{(r)}(X) = \tilde{F}(X; K^{(1,2,...,(r-2))}) \tag{6}$$

where, $\tilde{F}(\cdot)$ is a function of $GF(2)^n \times GF(2)^{s\times(r-2)} \rightarrow GF(2)^n$, and $K^{(1,2,...,(r-2))}$ denotes the keys from the first round to the $(r-2)$th round. Thus, $H^{(r)}(X)$ is computed from the plaintext side. On the other hand, it is also computed from the ciphertext side by assuming the key of the final round $K^{(r)}$ as follows.

$$H^{(r)}(X) = F(C_L(X); K^{(r)}) + C_R(X) \tag{7}$$

If $\deg_X\{H^{(r)}(X)\} = d$, then the following equation is satisfied.

$$\Delta^{(d)} \tilde{F}(X; K^{(1,2,...,(r-2))}) = \text{const} \tag{8}$$

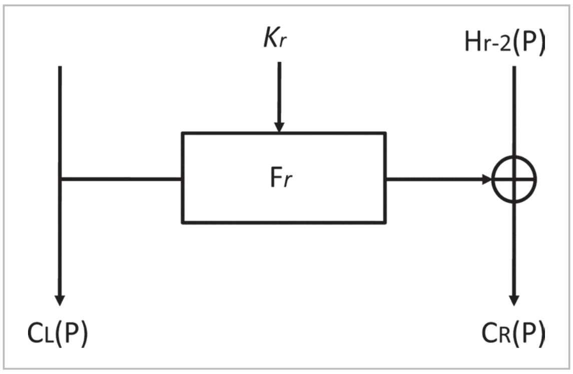


**Fig.3** *The final round of r-round Feistel type block cipher*

Equations (6), (7), and (8) yield the following equation.

$$\sum_{A \in V_{[a0,a1,...,ad-1]}} \{F(C_L(X+A); K^{(r)}) + C_R(X+A)\} = \text{const} \tag{9}$$

If the value of const has been obtained, the value of $K^{(r)}$ can be obtained by solving this equation. Therefore, hereafter this equation is called the attack equation.

Consider solutions for the attack equation by applying the linearization method described in reference [8]. This method significantly reduces computational cost by transforming the attack equation to linear equations. For further details refer to [8][9]. Let $L$ denote the total of independent unknowns that have been redefined by linearization, and let $H$ denote the width of the attack equation, then the $\left\lfloor \frac{L}{H} \right\rfloor \times 2^N$ numbers of chosen plaintext and $\left\lfloor \frac{L}{H} \right\rfloor \times 2^N \times L$ rounds of F function computation will be required. Compared to an exhaustive search, the number of required chosen plaintext will be significantly increased; however, the computational cost will be reduced to a negligible amount.

### 3.3 Higher order differential attack against MISTY1

MISTY1 is known to have the following algebraic property (reference [9]).

**Property 2:** When MISTY1 does not have FL function, computation of seventh order differential with a variable that consists of the right seven bits of FO$_i$ satisfies the following equation, regardless the value of the fixed part of plaintext and expanded key.

$$\Delta^{(7)} Z^{L7}_{i+2} = 0\text{x}6d \tag{10}$$

For example, let $X_0$ be a variable, then the following equation is satisfied:

$$\Delta^{(7)} Z^{L7}_{3} = 0\text{x}6d \tag{11}$$

Reference [9] describes attacks that utilize the above against 5-round MISTY1 without FL function. Analysis results of the characteristics of seventh order differential are also described in reference [10]. Thus, the use of low algebraic degree for the S-box causes a flaw in the security. This paper describes applied attacks based on Property 2.

Since FL function consists of AND and OR operations, if the following key is input, the output value of FL function can be controlled.

$$KL_{21} = KL_{31} = 0\text{x}0000,\ KL_{22} = KL_{32} = 0\text{x}ffff \quad (12)$$

If these conditions are satisfied, Property 2 will be satisfied even with FL function. The problem is whether there is a 128-bit key that satisfies such an expanded key. According to the key schedule (Table 1),

$$K'_3 = K_2 = 0\text{x}0000,\ K_5 = K'_8 = 0\text{x}ffff \quad (13)$$

However,

$$K'_3 = FI(K_3; K_4),\ K'_8 = FI(K_8; K_1) \quad (14)$$

From the above, if $K_1$, $K_2$, $K_3$, $K_4$, $K_5$, and $K_8$ are fixed, it is possible to carry out attacks based on Property 2 of seventh order differential characteristics, even with FL function. Figure 4 shows the estimate of formal algebraic degree in FI function, but the input of expanded key is omitted. Figure 5 shows the estimate of formal algebraic degree both when $KL_3$ is fixed and when it is not fixed.

In order to fix $KL_3$, $K_1$, $K_2$, $K_3$, $K_4$, $K_5$, and $K_8$ should be fixed, and this is not a very realistic attack condition. However, if $KL_3$ is not fixed, only the following condition is required:

$$K_5 = K'_7 = 0\text{x}ffff \quad (15)$$

Nevertheless, the value of seventh order differential is not fixed as Fig. 5 shows, and it cannot be used for attack as it is. Then, we discovered from computer results that if eighth order differential is used with one randomly chosen bit from $X_0$ and $X_1$, the following equation is satisfied.

$$\Delta^{(8)} Z_3^{L7} = 0 \quad (16)$$

Further details are described in reference [11]. The use of this property derives the attack equation shown in Fig. 6. That is,

$$\begin{aligned} A &= FL_8(C_R; KL_8)^{L7} = FL_8(C_R; K'_6, K_8) \\ B &= FO_5(C; KO_5, KI_5)^{L7} = FO_5(C; K_1, K'_2, K_4, K_5, K'_6, K_7, K'_8)^{L7} \\ C &= FL_7(C_L; KL_7) + FO_6(FL_8(C_R; K'_6, K_8); KO_6, KI_6) \\ &= FL_7(C_L; K'_2, K_4) + FO_6(FL_8(C_R; K'_6, K_8); K'_1, K_2, K'_3, K_5, K'_7, K'_8) \end{aligned} \quad (17)$$

it is not possible to solve them by applying the solution described in **3.2** as it is due to numerous variables. It is more appropriate to use algebraic solution for one part and exhaustive search for the other part to solve them.

- $K_5$, $K_7$ = fixed
- $K'_3$, $K_6$, $K_8$ = exhaustive search

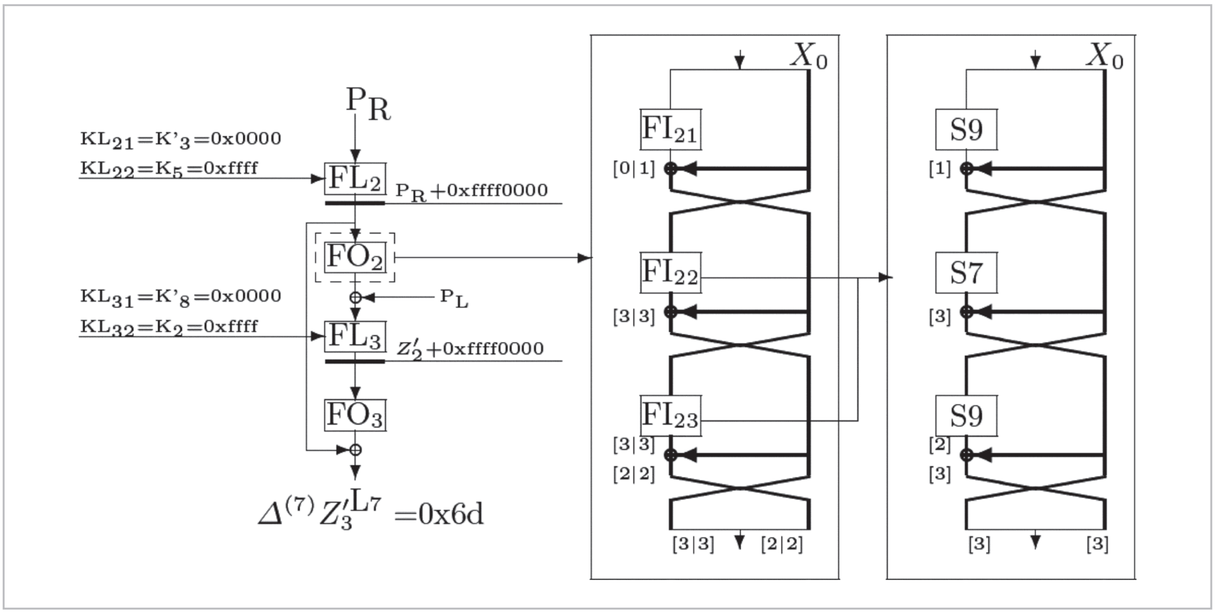


**Fig.4** Estimate of formal algebraic degree

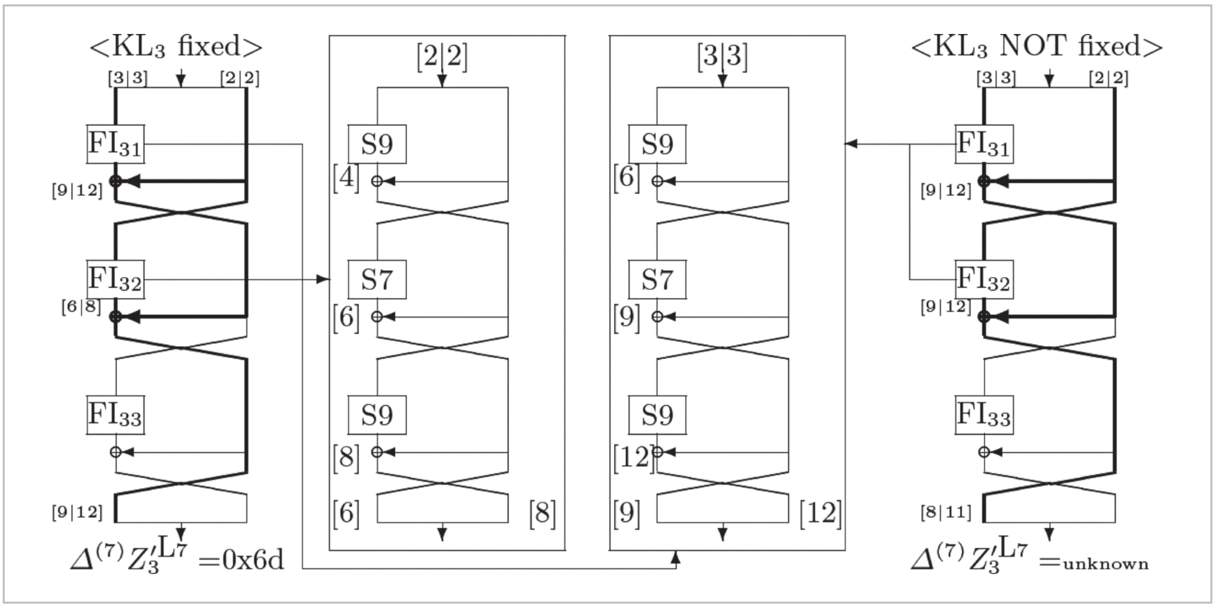


**Fig.5** Estimate of formal algebraic degree of $FO_3$ (left: $KL_3$ is fixed, right:$KL_3$ is not fixed)

- $K_7'$ = linearization method

When categorized as above, $H=7$, $L=13269+64$ are given, then the computational cost can be estimated as $2^{80.6}$, and the amount of data can be estimated as $2^{18.9}$. For further details on computation, refer to reference [11]. The unit of computational cost is the computation counts for one round of FO function, and the amount of data is the number of pairs of chosen plaintext and the corresponding ciphertext.

## 3.4 The meaning of the result of attacks

The result of attacks described in **3.3** is obtained when a key has a specific value. In such condition, it is possible to solve up to the sixth round of an 8-round structure. This means that the security that was proposed as being provable with three rounds has been compromised. From this result, it can be said that the margin of the security of MISTY1 has been reduced from five rounds to two rounds.

On the other hand, even if the given conditions were very advantageous to attackers, the attack did not result in a full spec attack against MISTY1. The above-mentioned attack will be only successful when a certain weak key is chosen as in equation (15). Sufficient security can be maintained by eliminating the use of such keys. In addition, the computational cost required for the attack is not realistic at the time of writing in 2011. Therefore, we conclude that MISTY1 is secure enough to be used in practice.

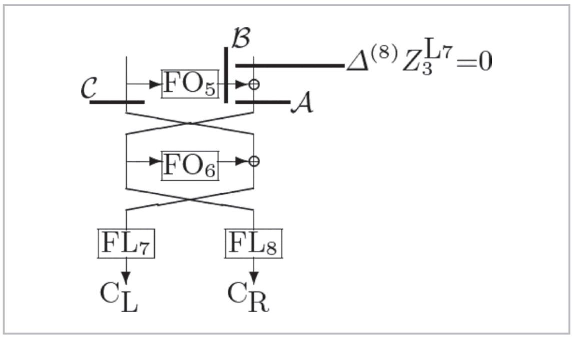


**Fig.6** Attack equation against 6-round MISTY1

Variables A, B, C are shown in the equation (17)

However, MISTY1 is not always used based on algorithms. When it is implemented on FPGA, even if legitimate keys are used and the full spec has been implemented, it could be attacked by obtaining intermediate computational information or by using some kind of interference, and the data could be altered. When taking account of such attacks, sufficient security could not be completely maintained. Therefore, we verify the feasibility of attacks against such implementation.

## 4 Fault based attacks against FPGA cryptographic modules

Electronic devices generate noise in accordance with their operations. The characteristics of the noise depend on the information processed by the device and the content of the operation. Side-channel attacks utilize such properties, and analyze confidential information by analyzing power consumption waveforms and electromagnetic emanation. In **3**, we described attacks in the case where an 8-round structure has been reduced to 6-round, but it is not realistic to consider a 6-round would be used in the actual implementation. Therefore, there is no way that an attacker can obtain intermediate computational results; however, this is possible in side-channel attack scenarios. Consequently, if attackers obtain the output from $FO_5$, it will become a realistic threat.

On the other hand, it is not known whether the input data consists of an eighth order differential structure as the attackers intended. In addition, if the key is stored in the memory, there are no such weak keys that the attackers expect. They need to tamper with the value of the expanded key or input information by some method. The following two methods are known to be used for such attacks [12].

- Invasive attack
- Non-invasive attack

In invasive attacks, the attackers alter the operation to their intended operation by tampering with circuit, and it is not possible to restore the circuit or device back to the original state. In non-invasive attacks, the attackers cause a brief malfunction but do not modify the circuit itself. Since non-invasive attacks could allow illegal modules to be produced, they are more serious threats for consumer products.

There are various methods to carry out non-invasive attacks, and we performed attacks that use electromagnetic emanation in this research. The target of attack was SASEBO, Side-channel Attack Standard Evaluation Board (Fig. 7)[13]. In addition, since our purpose was to verify whether information could be altered as intended and not to perform actual cryptanalysis, we chose circuits and data transfer in which alteration of information could be observed easily.

First, as shown in Fig. 8, we verified whether it was possible to alter signals by irradiating electromagnetic emanation directly to the board. Most of the experiments resulted in failure. This is because when the amount of irradiated electromagnetic emanation is large enough to add interference to signals, the capacitors on the power circuit will be destroyed, and as a result the board itself will stop operating. One of the other reasons of the failure was that due to the property of electromagnetic emanation, it was difficult to irradiate them to one location and they could not be controlled. We conducted other experiments by replacing the power source with batteries, but they were not effective.

Next, we verified alteration of signals using a surge (Fig. 9). For example, when an electronic lighter is ignited near the coin slot of arcade video game, it is wrongly recognized as coins were inserted and false operation will be caused. Similar to this, we experimented to alter signals by causing spike-like potential changes. We confirmed that signals had been altered by conducting the operation in the locations shown in Fig. 10. However, we could not change signals to arbitrary ones.

Finally, we developed a device that input enough signal to change the voltage directly

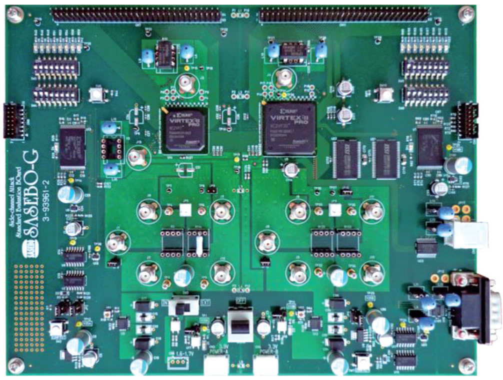


**Fig.7** Side-channel attack standard evaluation board SASEBO-G

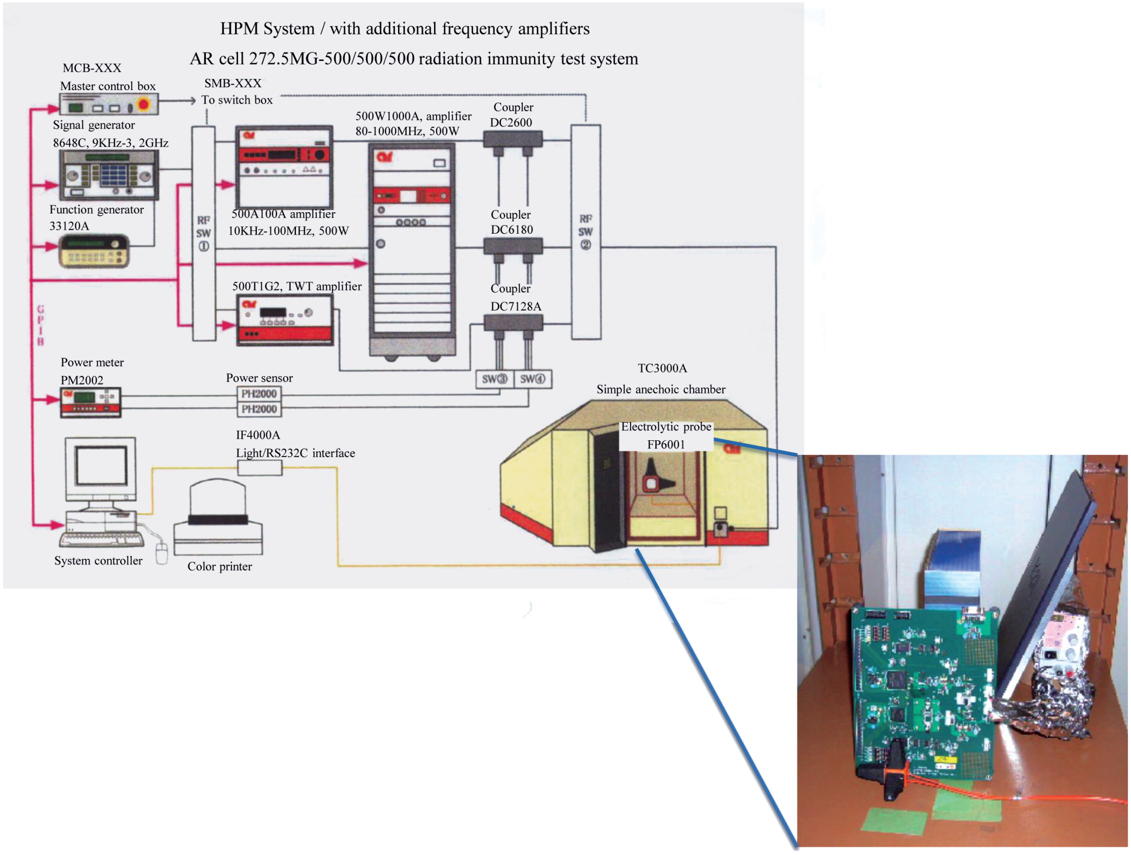


**Fig.8** Radiation experiment inside a chamber

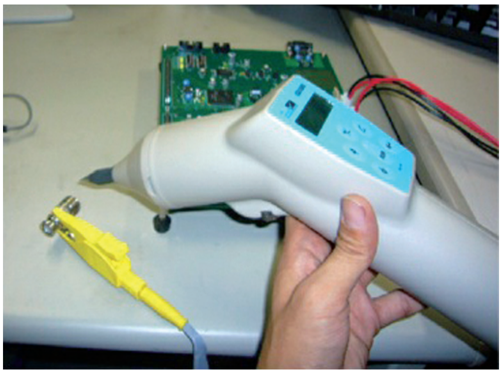


**Fig.9** Experiment of surge irradiation

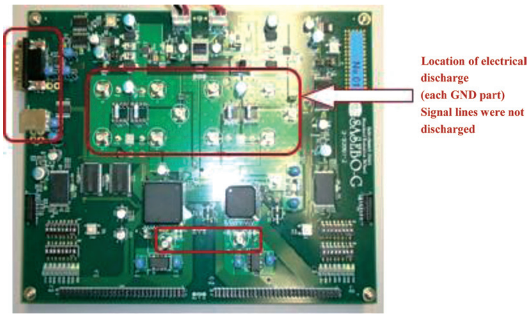
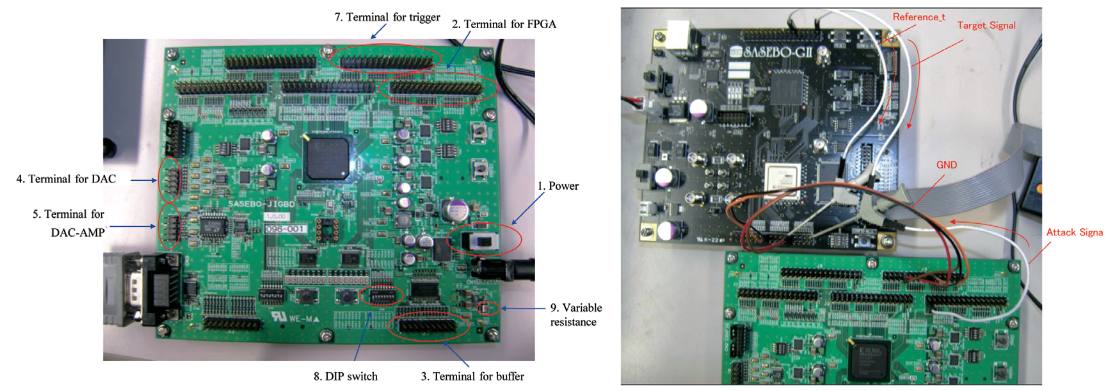


**Fig.10** Locations where signals were successfully altered by surge irradiation experiment (red frame)

into the signal lines (Fig. 11). When we used it, we were able to input arbitrary signals into the FPGA.

From the results of the experiments, we found the following.

- Simply irradiating electromagnetic emanation is a difficult method to attack FPGA and smart cards. We also conducted experiments of local irradiation using a probe, but we did not observe the phenomena described in reference [14], at least on SASEBO.
- Some fault based attacks do not require altering signals to arbitrary ones but sim-



(i) The board for carrying out attacks

(ii) Attack to SASEBO-G2

**Fig.11** The board for carrying out attacks

ply need errors to be caused [15]. Such purposes can be fully achieved by using a surge, which is very easy to execute.

- A device similar to that shown in Fig. 11 is sufficient to arbitrarily alter the input to an FPGA. Its structure is simple, and it is inexpensive. For further details, refer to [12].

We were not able to alter the data that was being processed in the FPGA by using electromagnetic emanation. Conversely, the important processes for security reasons should be processed in the FPGA. For example, if all processes to generate and read expanded keys are executed in the FPGA, it is not possible to carry out fault based attacks that use weak keys as described in **3**. On the other hand, it became clear that it is possible to prevent legitimate input and carry out chosen plaintext attacks. It is apparent that tamper resistant implementation has an importance as it prevents circuit networks from being directly operated.

## 5 Evaluation of linear complexity of pseudo random number generator

The linearization method described in **3** is an algebraic attack, which redefines non-linear variable terms of quadratic or higher order as one new variable and transforms a quadratic or higher degree equation into a linear equation in order to make it easy to solve. For example, for $f(x_1, x_2)=x_1x_2+x_1+x_2$, redefine as $y_1=x_1x_2$, $y_2=x_1$, $y_3=x_2$, then $f(x_1, x_2)\rightarrow f'(y_1, y_2, y_3)=y_1+y_2+y_3$ is obtained, thus a quadratic equation with respect to $x_1$, $x_2$ can be expressed as a linear equation with respect to $y_1, y_2, y_3$. Multi-degree equations can be expressed as linear equations, but on the other hand, redefined variables may be increased significantly. In this example, variables were increased from two to three.

When a sequence is given, linear complexity is given as the minimum required number of rounds of LFSR to generate the same sequence. From the viewpoint of linear complexity, replacing terms of quadratic or higher order that are generated by non-linear functions with new independent terms is equal to adding new registers to construct an equivalent LFSR. In addition, linear complexity is estimated by computing each AND operation, XOR operation, and NOT operation using logic operation circuit and totaling the results.

AND Operation

Consider a product sequence of the output from LFSR#1 and LFSR#2.

$$r(t)=x_1(t)x_2(t)$$

Here, when $r(t)$ is expressed in Boolean algebra equation, it can be expressed only by quadratic terms, $a_{1i}, (i=1\sim s_1)$ and $a_{2j}, (j=1\sim s_2)$. Since there are $s_1\times s_2$ types of quadratic terms, if $s_1$ and $s_2$ are coprime and generator polynomials $f_1(x)$ and $f_2(x)$ are both primitive polynomial and irreducible, the output sequence $r(t)$ can be generated by $s_1s_2$-round LFSR that has a $s_1s_2$ degree polynomial as a generator polynomial. Therefore, the linear complexity is given by $s_1s_2$ in this case.

On the other hand, from the viewpoint of the linearization method, $r(t)$ can be expressed by $s_1s_2$ quadratic terms as described above. Therefore, Property 3 is derived as follows.

**Property 3:** A product sequence of two LFSR where $s_1$ and $s_2$ are coprime and generator polynomials $f_1(x)$ and $f_2(x)$ are both primitive polynomial and irreducible satisfies the condition: linear complexity = $s_1s_2$.

XOR Operation

Consider an exclusive OR sequence of the output from LFSR#1 and LFSR#2.

$$r(t)=x_1(t)\oplus x_2(t)$$

In this case, $r(t)$ can be generated from LFSR that has the least common multiple polynomial of generator polynomials $f_1(x)$ and $f_2(x)$ as a generator polynomial. If $f_1(x)$ and $f_2(x)$ are irreducible to each other, and if $s_1$ and $s_2$ are coprime, the generator polynomial of $r(t)$ is given by $f(x)=f_1(x)f_2(x)$. Therefore, the degree of $f(x)$ is given by $s_1+s_2$, and the number of rounds is given by $s_1+s_2$.

On the other hand, from the viewpoint of linearization method, since $r(t)$ is linear, the number of variables is given by $s_1+s_2$. Therefore, Property 4 is derived as follows.

**Property 4:** An exclusive OR sequence of two LFSR where $s_1$ and $s_2$ are coprime and generator polynomials $f_1(x)$ and $f_2(x)$ are both primitive polynomial and irreducible satisfies the condition: linear complexity = $s_1+s_2$.

NOT Operation

Consider a sequence that inverts the output from LFSR#1.

$$r(t)=\overline{x_1(t)}$$

As a logic operation circuit, this operation is realized by exclusive OR sequences of a single round LFSR with the initial value 1 and LFSR#1. Therefore, in accordance with Property 2, linear complexity = $s_1+1$.

On the other hand, from the viewpoint of a linearization attack,

$$r(t)=x_1(t)\oplus 1$$

Here, only constant terms are included and variable terms are not affected. Therefore, the number of variables is given by $s_1$. Therefore, Property 5 is derived as follows.

**Property 5:** When the output from LFSR#1 is inverted by NOT operation, in terms of the output sequence, the linear complexity is computed by $s_1+1$ with the general method, and by $s_1$ with the linearization method.

As above, there is a difference in the estimated values between the linearization method and general method. This becomes clear when considering unoptimized logic circuits as shown in Fig. 12.

Figure 12 can be estimated as $s_1+s_2+2$ by the general method; however, it is clear that it is $s_1+s_2$. Therefore, the linearization method is more effective for computing more accurate linear complexity.

In addition, the required conditions for computing linear complexity, computational cost, and length of series are shown in Table 2, comparing to the Belekamp-Massey method and Games-Chan method. We confirmed that the linearization method is an effective method to compute linear complexity.

## 6 Summary

This paper outlined of the achievements of security evaluation activity for symmetric

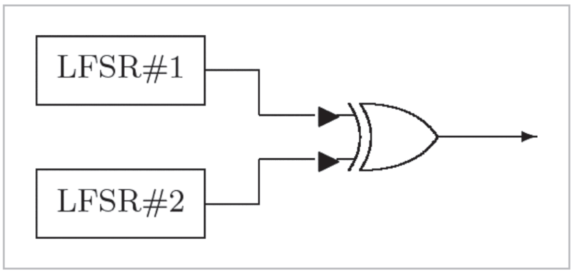


*Fig.12* *Unoptimized logic circuit*

**Table 2** *Comparison of condition, computational cost, and memory space*

| | Condition | Computational cost | Memory space cost |
|---|---|---|---|
| Berlekamp-Massey | output sequence with period $N$ | $O(N^2)$ | $N$ |
| Games-Chan | output sequence with period $N = 2^m$ | $O(N)$ | $N$ |
| **Linearization method** | algebraic expression of PRNG | $\leq O(2^n)$ | $\leq 2^n$ |

ciphers of the Security Fundamentals Group between 2006 and 2011. The activity is linked to the initiatives of CRYPTREC (Cryptography Research and Evaluation Committees) that is described in **4-9**. Although it is omitted in this paper, we also conducted security evaluation for 128-bit block cipher AES and HyRAL, 64-bit block cipher KASUMI and ICEBERG, stream cipher Multi-S01, and hash function SHA-1. However, we only managed to confirm that they were secure, and were not able to obtain better evaluation results than the previous ones; therefore, we only gave a verbal presentation regarding this matter in Japan.

There have been improvements in the theories to construct cryptographic technologies securely in recent years. We consider it is important to continue evaluation activities in order for these cryptographic technologies to be used securely, which is the mission assigned to NICT, a fair and neutral organization.

## *References*

1 Naional Institute of Standards and Technology, “ADVANCED ENCRYPTION STANDARD (AES),” Federal Information Processing Standards Publication (FIPS-PUB) 197, 2001.

2 M.Matsui, “New block encryption algorithm MISTY,” Proceedings in Fast Software Encryption 1997, LNCS. 1267, Springer-Verlag, 1997, pp. 54–68.

3 E.Biham and A.Shamir, “Differential Cryptanalysis of the Data Encryption Standard,” Springer Verlag, 1993.

4 M.Matsui, “The First Experimental Cryptanalysis of the Data Encryption Standard,” Advances in Cryptology - CRYPTO '94, 14th Annual International Cryptology Conference, LNCS. 839, Springer-Verlag, 1994, pp. 1–11.

5 International Organization for Standardization. ISO/IEC WD 18033-3: information technology—security techniques—encryption algorithms—Part 3: block ciphers, 2002.

6 CRYPTographic Research and Evaluation Committees: CRYPTREC, http://www.cryptrec.go.jp

7 NESSIE (New European Schemes for Signatures, Integrity and Encryption), https://www.cosic.esat.kuleuven.be/nessie/

8 S.Moriai, T.Shimoyama, and T.Kaneko, “Higher Order Differential Attack of a CAST Cipher,” Proceedings in Fast Software Encryption 1998, LNCS. 1372, Springer-Verlag, 1998, pp. 17–31.

9 H.Tanaka, K.Hisamatsu, and T.Kaneko, “Strength of MISTY1 without FL Function for Higher Order Differential Attack,” Applied Algebra, Algebraic Algorithms and Error-Correcting Codes, 13th International Symposium, AAECC-13, LNCS. 1719, Springer-Verlag, 1999, pp. 221–230.

10 S.Babbage and L.Frisch, “On MISTY1 Higher Order Differential Cryptanalysis,” ICISC2000, LNCS. 2015, Springer-Verlag, 2001, pp. 22–36.

11 H.Tanaka, Y.Hatano, N.Sugio, and T.Kaneko, “Security Analysis of MISTY1,” Information Security Applications, 8th International Workshop, WISA 2007, LNCS. 4867, Springer-Verlag, 2008, pp. 215–226.

12 Hidema Tanaka, “A study on experiment method of fault induction attack using electro-magnetic emanation,” Symposium on Cryptography and Information Security 2009 : SICS2009 2A3-1.

13 SASEBO Side-channel Attack Standard Evaluation Board, http://www.rcis.aist.go.jp/special/SASEBO/index-ja.html

14 D.Agrawal, B.Archambeault, J.R.Rao, and P.Rohatgi, "The EM Side-Channel(s)," CRYPTOGRAPHIC HARDWARE AND EMBEDDED SYSTEMS - CHES 2002, LNCS. 2523, Springer-Verlag, 2003, pp. 29–45.

15 J.S. Coron, A. Joux, I. Kizhvatov, and P. Paillier, "Fault Attacks on RSA Signatures with Partially Unknown Messages," CRYPTOGRAPHIC HARDWARE AND EMBEDDED SYSTEMS - CHES 2009, LNCS. 5747, Springer-Verlag, 2009, pp. 444–456.

(Accepted June 15, 2011)

***TANAKA Hidema,*** *Ph.D.*

*Director, Security Fundamentals Laboratory, Network Security Research Institute*

*Information Security, Cryptographic Technology, Information Theory*