# Higher Order Differential Attack of *Camellia*(II)

Yasuo Hatano, Hiroki Sekine, and Toshinobu Kaneko

Department of Electrical Engineering,
Tokyo University of Science
2641, Yamazaki, Noda, Chiba, 278-8510, Japan
{j7302656,j7301625}@ed.noda.tus.ac.jp
kaneko@kaneko01.ee.noda.tus.ac.jp

**Abstract.** *Camellia* is a 128-bit block cipher, proposed by NTT and Mitsubishi in 2000. It has been shown that 10 round variant without FL function under a 256-bit secret key is attackable by Higher Order Differential Attack and even if FL function is included, 9 round variant is attackable by Square Attack. In this paper, we present a new attack of *Camellia* using 16-th order differential and show that 11 round variant without FL function is attackable. Moreover, we show that 11 round variant with FL function is attackable, if we use chosen ciphertexts for this attack.

## 1 Introduction

*Camellia*[1] is a 128-bit block cipher proposed by NTT and Mitsubishi in 2000. Designers have evaluated it's strength against various attacks and insist that it is secure against Truncated Differential and Truncated Linear Cryptanalysis, if it consists of at least 12 rounds (without FL function) or 11 rounds (with FL function). Also, Shirai et. al. have shown that it is secure against these cryptanalysis in the case of 11 round variant (without FL function) or 10 round variant (with FL function) in FSE2002[12].

On the other hand, we have already shown that 10 round variant without FL function, under a 256-bit secret key, is attackable by a new attack, which we call "Controlled Higher Order Differential Attack"[7][1]. Furthermore, Yeom et. al. have shown that 9 round variant can be attacked by Square Attack, even if FL function is included[15].

In this paper, we present an attack of *Camellia* using 16-*th* Order Differential and show that 11 round variant without FL function, under a 256-bit secret key, is attackable. Moreover, we show that 11 round *Camellia* with FL function is attackable, if we use chosen ciphertexts. Table.1(a)[15] summarizes known attacks of *Camellia* and Table.1(b) shows our results in this paper.

This paper is organized as follows. Section 2 shows the structure of *Camellia*. Section 3 describes Higher Order Differential and leads an attack equation for *Camellia* without FL function. Section 4 shows the results of computer experiments and these analyses. In section 5, we conduct a basic attack on *Camellia*

[1] We call that paper "Higher Order Differential Attack of *Camellia* (I)"

K. Nyberg and H. Heys (Eds.): SAC 2002, LNCS 2595, pp. 129–146, 2003.

**Table 1.** The necessary number of chosen plaintexts and complexity for attacks

(a) Previous results

| Round | FL | Method | Plaintexts | Complexity | Authors |
|---|---|---|---|---|---|
| 5R | $\times$ | SA | $2^{10.3}$ | $2^{48}$ | Y.He et.al[4] |
| | $\times$ | SA | $2^{16}$ | $2^{16}$ | Y.Yeom et.al.[15] |
| 6R | $\times$ | HODA | $2^{17}$ | $2^{19.4}$ | T.Kawabata et.al.[7] |
| | $\times$ | SA | $2^{11.7}$ | $2^{112}$ | Y.He et.al[4] |
| | $\times$ | SA | $2^{56}$ | $2^{56}$ | Y.Yeom et.al.[15] |
| 7R | $\times$ | HODA | $2^{19}$ | $2^{51.2}$ | T.Kawabata et.al.[7] |
| | $\times$ | TDC | $2^{82.6}$ | 192 | S.Lee et.al.[10] |
| | $\circ$ | SA | $2^{58.5}$ | $2^{80.2}$ | Y.Yeom et.al.[15] |
| 8R | $\times$ | HODA | $2^{20}$ | $2^{126}$ | T.Kawabata et.al.[7] |
| | $\times$ | TDC | $2^{83.6}$ | $2^{55.6}$ | S.Lee et.al.[10] |
| | $\circ$ | SA | $2^{59.7}$ | $2^{138.6}$ | Y.Yeom et.al.[15] |
| 9R | $\times$ | HODA | $2^{21}$ | $2^{190.8}$ | T.Kawabata et.al.[7] |
| | $\circ$ | SA | $2^{50.5}$ | $2^{202.2}$ | Y.Yeom et.al.[15] |
| 10R | $\times$ | HODA | $2^{21}$ | $2^{254.7}$ | T.Kawabata et.al.[7] |

HODA : Higher Order Differential Attack
SA : Square Attack
TDC : Truncated Differential Cryptanalysis
Complexities are based on the number of encryptions.

(b) Our results[‡]

| Round | ET | $K1/K2/K3$ | Plaintexts | Complexity | FL |
|---|---|---|---|---|---|
| 6R | $+1$R | 8/0/0 | $2^{17}$ | $2^{18*}$ | $\times$ |
| 7R | $+2$R | 48/40/0 | $2^{19}$ | $2^{57}$ | $\times$ |
| | $-1+1$R | 24/0/16 | $2^{34}$ | $2^{34**}$ | $\times$ |
| 8R | $+3$R | 112/104/0 | $2^{20}$ | $2^{120}$ | $\times$ |
| | $-1+2$R | 64/40/16 | $2^{36}$ | $2^{71}$ | $\times$ |
| | $-2+1$R | 80/0/72 | $2^{92}$ | $2^{93}$ | $\times$ |
| 9R | $+4$R | 176/168/0 | $2^{21}$ | $2^{188}$ | $\times/\circ$ |
| | $-1+3$R | 128/104/16 | $2^{37}$ | $2^{136}$ | $\times$ |
| | $-2+2$R | 120/40/72 | $2^{92}$ | $2^{111}$ | $\times$ |
| 10R | $+5$R | 240/232/0 | $2^{21}$ | $2^{252}$ | $\times$ |
| | $-1+4$R | 176/168/16 | $2^{37}$ | $2^{201}$ | $\times/\circ$ |
| | $-2+3$R | 184/104/72 | $2^{92}$ | $2^{186}$ | $\times$ |
| 11R | $-2+4$R | 248/168/72 | $2^{93}$ | $2^{255.6}$ | $\times/\circ$ |

Computer Simulation $*: 0.2[sec]$ , $**: 1.5[h]$

[‡] Complexities are based on the number of encryptions. ET column shows an "Elimination Technique", in which we call $+n$R when we guess the last $n$ round keys and $-n$R when we guess the first $n$ round keys for the attack. $K1$ denotes the total number of guessed key bits. $K2$ and $K3$ denote the number of guessing key bits, for which we perform simple brute-force search, in the last $(n-1)$ round, and the first $n$ round, respectively.

without FL function using 16-*th* order differential, and expand the attack in section 6,7. Section 8 shows some computer experiments, which gives an attack of 11 round *Camellia* with FL function by the chosen ciphertext. Section 9 summarizes this paper.

## 2 *Camellia*[1]

*Camellia* is a 128-bit block cipher and supports 3 kinds of secret key size, 128, 192, and 256 bits. It's number of rounds are 18 (128-bit secret key) and 24 (192, 256-bit secret key), respectively. It has a Feistel structure with SPN type round function, called F function. Additionally, FL/FL$^{-1}$ function is inserted every 6 round.

Fig.1,2 and Fig.3 shows the main structure of *Camellia* and its components. For simplicity, we call *Camellia* without FL function as *Camellia*. Note that we omit key inputs of $KW_i$ in the following explanation, since these have no influence on our attacks.

Let $P_L, P_R$ be the left and right half of a plaintext $P$, and $C_L, C_R$ be those of the ciphertext, respectively. Let $X_{Li}, X_{Ri}$ be the left and right half of an *i-th* round input variables, and $Y_{Li}, Y_{Ri}$ be these outputs, respectively. Note that in $r$ round *Camellia*,

$$X_{Li} = Y_{L(i-1)} \ , X_{Ri} = Y_{R(i-1)} \tag{1}$$

$$X_{L1} = P_L \ , X_{R1} = P_R \tag{2}$$

$$C_L = Y_{Lr} \ , C_R = Y_{Rr}. \tag{3}$$

Let $X_i, Y_i$ be the input and output variable of *i-th* round F function, respectively. And $K_i$ denotes an input key to the function.

$$Y_i = F(X_i; K_i) \tag{4}$$

$$\begin{cases} X_i = {}^t(x_{i1}, \cdots, x_{i8}) \ x_{ij} \in \mathrm{GF}(2)^8 \ (j = 1 \sim 8) \\ Y_i = {}^t(y_{i1}, \cdots, y_{i8}) \ y_{ij} \in \mathrm{GF}(2)^8 \ (j = 1 \sim 8) \\ K_i = {}^t(k_{i1}, \cdots, k_{i8}) \ k_{ij} \in \mathrm{GF}(2)^8 \ (j = 1 \sim 8) \end{cases} \tag{5}$$

Let $Z_i$ be the intermediate variable in the function.

$$Z_i = {}^t(z_{i1}, \cdots, z_{i8}) \ \ z_{ij} \in \mathrm{GF}(2)^8 \ \ (j = 1 \sim 8) \tag{6}$$

Fig.2 illustrates these variables in *i-th* round F function. F function is composed of two function layers. One is S function, the other is P function.
[S Function]

$$Z_i = S(X_i \oplus K_i) \tag{7}$$

$$\begin{cases} z_{i1} = s_1(x_{i1} \oplus k_{i1}) \\ z_{i2} = s_2(x_{i2} \oplus k_{i2}) \\ z_{i3} = s_3(x_{i3} \oplus k_{i3}) \\ z_{i4} = s_4(x_{i4} \oplus k_{i4}) \\ z_{i5} = s_2(x_{i5} \oplus k_{i5}) \\ z_{i6} = s_3(x_{i6} \oplus k_{i6}) \\ z_{i7} = s_4(x_{i7} \oplus k_{i7}) \\ z_{i8} = s_1(x_{i8} \oplus k_{i8}) \end{cases} \tag{8}$$

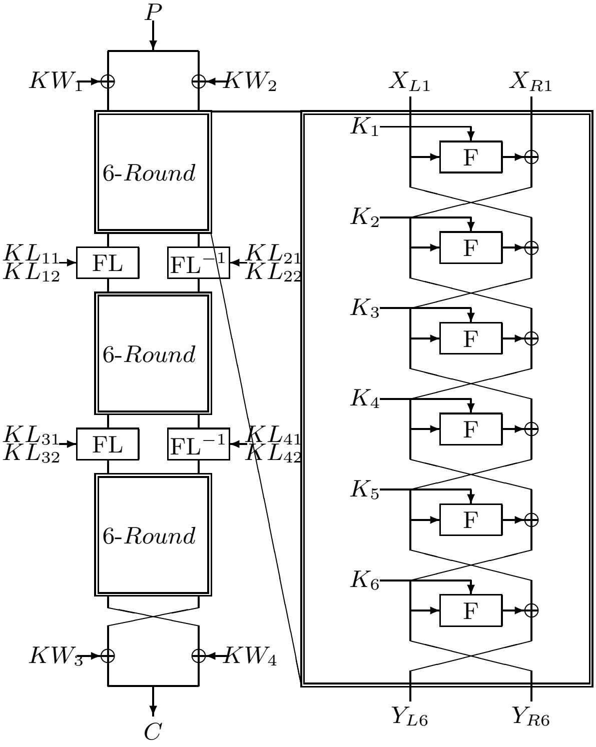


**Fig. 1.** *Camellia*(128-bit secret key)

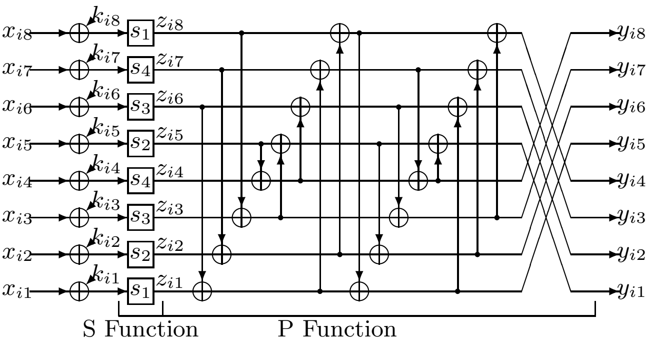
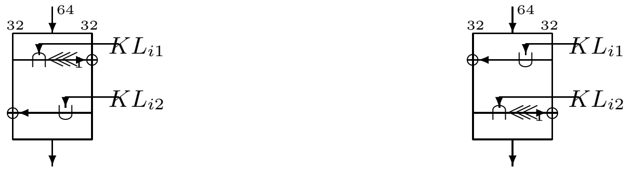


**Fig. 2.** F function



(a) FL

(b) FL$^{-1}$

**Fig. 3.** FL function

where $s_1(), s_2(), s_3()$, and $s_4()$ denote S-Boxes, which are bijective functions over GF$(2^8)$.
[P Function]

$$Y_i = \mathbf{P} Z_i, \tag{9}$$

where $\mathbf{P}$ is a regular matrix as follows.

$$\mathbf{P} = \begin{pmatrix} 1\,0\,1\,1\,0\,1\,1\,1 \\ 1\,1\,0\,1\,1\,0\,1\,1 \\ 1\,1\,1\,0\,1\,1\,0\,1 \\ 0\,1\,1\,1\,1\,1\,1\,0 \\ 1\,1\,0\,0\,0\,1\,1\,1 \\ 0\,1\,1\,0\,1\,0\,1\,1 \\ 0\,0\,1\,1\,1\,1\,0\,1 \\ 1\,0\,0\,1\,1\,1\,1\,0 \end{pmatrix} \tag{10}$$

# 3 Higher Order Differential Attack

## 3.1 Higher Order Differential[9]

Let $E()$ be a function that transforms an input $X \in \mathrm{GF}(2)^n$ to the output $Y \in \mathrm{GF}(2)^m$ under a key $K \in \mathrm{GF}(2)^s$.

$$Y = E(X; K) \tag{11}$$

Let $\{A_1, \cdots, A_i\}$ be a set of linear independent vectors in $\mathrm{GF}(2)^n$ and $V^{(i)}$ be the vector sub-space spanned by these vectors. Then, *i-th* order differential is defined as follows.

$$\varDelta^{(i)}_{V^{(i)}} E(X; K) = \bigoplus_{A \in V^{(i)}} E(X \oplus A; K), \tag{12}$$

where $\bigoplus_{A \in V^{(i)}}$ denotes the ex-OR sum over $V^{(i)}$. In the following, we denote $\varDelta^{(i)}_{V^{(i)}}$ as $\varDelta^{(i)}$, when it is clearly understood.

In this paper, we use the following properties of Higher Order Differential:

**[Property 1]**

If the degree of $E(X;K)$ equals $N$, then

$$deg_X E(X;K) = \begin{cases} \Delta^{(N+1)} E(X;K) = 0 \\ \Delta^{(N)} E(X;K) = const. \end{cases}$$

**[Property 2]**

Higher order differential operation has a linear property.

$$\Delta^{(i)} \{E_1(X;K_1) \oplus E_2(X;K_2)\} = \Delta^{(i)} E_1(X;K_1) \oplus \Delta^{(i)} E_2(X;K_2)$$

**[Property 3]**

If a set of $2^n$ vectors in GF$(2)^n$, which are outputs of $E(X;K)$, has the following properties, the value of *n-th* order differential becomes to 0.

*all* : each possible output value appears only once.
*balance*[2] : every output value appears even times.
*constant* : the output remains a constant value.

### 3.2 Attack Equation

Fig.4 shows the structure of 6 round Feistel type block cipher with SPN type round function. In the figure, the following equation holds.

$$\begin{aligned} & P_L \oplus Y_2 \oplus Y_4 \oplus Y_6 = C_L \\ \Longleftrightarrow\ & P_L \oplus C_L = \mathbf{P} Z_2 \oplus \mathbf{P} Z_4 \oplus \mathbf{P} Z_6 \\ \Longleftrightarrow\ & \mathbf{P}^{-1}\{P_L \oplus C_L\} = Z_2 \oplus Z_4 \oplus Z_6 \end{aligned} \quad (13)$$

For an *i-th* byte, the above equation is

$$\{\mathbf{P}^{-1} P_L\}_i \oplus \{\mathbf{P}^{-1} C_L\}_i = z_{2i} \oplus z_{4i} \oplus z_{6i}, \quad (14)$$

where $\{\bullet\}_i$ denotes the *i-th* byte of variable $\bullet$.

Let $V^{(N)}$ be a vector sub-space in $P_R$. Consider *N-th* order differential of Eq.(14) with respect to $V^{(N)}$. Since $P_L$ has *constant* property, from Property2 and 3, it can be calculated as

$$\bigoplus_{P_R \in V^{(N)}} \{\mathbf{P}^{-1} C_L\}_i = \Delta^{(N)} z_{2i} \oplus \Delta^{(N)} z_{4i} \oplus \Delta^{(N)} z_{6i}. \quad (15)$$

On the other hand, by guessing the key $k_{6i}$, *N-th* order differential of $z_{6i}$ with respect to $V^{(N)}$ can be calculated as

$$\Delta^{(N)} z_{6i} = \bigoplus_{P_R \in V^{(N)}} s_j(c_{Ri} \oplus k_{6i}), \quad (16)$$

where $c_{Ri}$ denotes *i-th* byte of right half of the ciphertext $C_R$(see Fig.4).

[2] This definition is different from that in Square Attack[15].

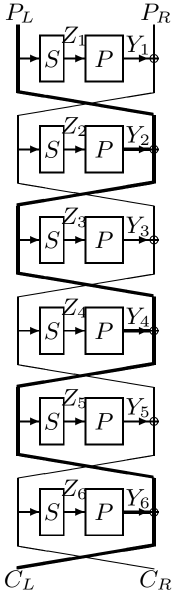


**Fig. 4.** Feistel Type Block Cipher with SPN Type Round Function

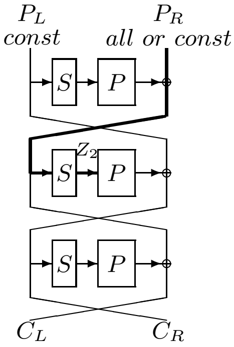


**Fig. 5.** Path : $P_R \rightarrow Z_2$

From Eq.(15) and Eq.(16), the following equation holds.

$$\bigoplus_{P_R \in V^{(N)}} \{\mathbf{P}^{-1}C_L\}_i = \bigoplus_{P_R \in V^{(N)}} s_j(c_{Ri} \oplus k_{6i}) \oplus \Delta^{(N)} z_{2i} \oplus \Delta^{(N)} z_{4i} \qquad (17)$$

If the following condition holds

$$\Delta^{(N)} z_{2i} \oplus \Delta^{(N)} z_{4i} = 0, \qquad (18)$$

then we have the following attack equation.

$$\bigoplus_{P_R \in V^{(N)}} \{\mathbf{P}^{-1}C_L\}_i = \bigoplus_{P_R \in V^{(N)}} s_j(c_{Ri} \oplus k_{6i}) \qquad (19)$$

The above equation(19) always holds when the guessed key $k_{6i}$ is true, and holds probabilistically when it is false. Thus, we can determine the true key $k_{6i}$ by an adequate number of $N$-*th* order differential. Table.2 summarizes our basic attack.

In the explanation above, we guess the last one round key and check its correctness by the attack equation(19). We call this +1R elimination technique. In the same manner, we call +$n$R elimination technique if we guess the last $n$ round keys and $-n$R elimination technique if we guess the first $n$ round keys.

**Table 2.** Basic 6R Attack

| | |
|---|---|
| Chosen Plaintext | $P_L$ has a *constant* property.<br>A variable for which Eq.(18) holds are chosen in $P_R$. |
| Guessing Key | $k_{6i}$ |
| Attack Equation | $\bigoplus_{P_R \in V^{(N)}} \{\mathbf{P}^{-1}C_L\}_i = \bigoplus_{P_R \in V^{(N)}} s_j(c_{Ri} \oplus k_{6i})$ (Eq.(19)) |

## 4 Effective Chosen Plaintext

### 4.1 Search for the Effective Chosen Plaintext by Computer Experiment

*Camellia* encrypts a plaintext in a byte oriented manner. So we perform the byte wise search for the variable sub-block with which Eq.(18) holds.

Let's consider 8-*th* order differential with respect to one byte of $P_R$. As shown in Fig.5, if we choose $i_1$-*th* byte of $P_R$ as a variable, $z_{2i_1}$ has *all* property, and $z_{2i}$ $(i \neq i_1)$ has *constant* property. Thus $\Delta^{(8)} z_{2i} = 0$. Therefore if $\Delta^{(N)} z_{4i} = 0$, then Eq.(18) holds. So we conducted a computer search for such variable sub-blocks. However, we could not find any one-byte variable to meet the condition.

If we select two bytes as variables for Higher Order Differential, we still have $\Delta^{(16)} z_{2i} = 0$. So we searched for such byte pairs by a computer experiment that make $\Delta^{(16)} z_{4i} = 0$. We show the result in Table.3. For example, the first entry in the table (1,2) means that if we choose the first and second byte pairs as variable sub-blocks for 16-*th* order differential, then $\Delta^{(16)} z_{46} = 0$. With 16-*th* order differential, the attack equation to determine a 6-*th* round key is as follows.

$$\bigoplus_{(i_1,i_2)} \{\mathbf{P}^{-1}C_L\}_i = \bigoplus_{(i_1,i_2)} s_j(c_{Ri} \oplus k_{6i}) \qquad (20)$$

where $\bigoplus_{(i_1,i_2)}$ denotes the sum over the variable, $i_1$-*th* and $i_2$-*th* byte in $P_R$.

### 4.2 The Pattern Whose 16-*th* Order Differential Equal to 0

If we select pairs $(i_1, i_2)$ as variable sub-blocks, only $z_{2i_1}, z_{2i_2}$, which are outputs of S-Boxes in the second round, are affected by these variables. In the third round, the input of each S-Boxes are among the following, $c$, $z_{2i_1} \oplus c$, $z_{2i_2} \oplus c$, $z_{2i_1} \oplus z_{2i_2} \oplus c$, where $c$ are some constant values. These outputs in third round S-Boxes are process by P function to make fourth round inputs. These inputs are converted by the corresponding S-Boxes in the fourth round to make $z_{4i}$.

We analyzed these processes. We found that the condition for $\Delta^{(16)} z_{4i} = 0$ is classified into following three patterns:
$[pattern1]$

$$z_{4i} = s_{j_1}(s_{j_2}(z_{2i_1} \oplus c_1) \oplus f(z_{2i_1} \oplus z_{2i_2}) \oplus c_2)$$

**Table 3.** Input variables sub-block byte pairs for $\Delta^{(16)} z_{4i} = 0$

| $(i_1 i_2)$ | observation byte |
|---|---|
| $(1,2)$ | $z_{46}[1]$ |
| $(1,4)$ | $z_{45}[1]$ |
| $(1,6)$ | $z_{43}[1], z_{47}[1], z_{48}[2]$ |
| ○ $(1,7)$ | $z_{42}[1], z_{43}[3], z_{45}[1], z_{46}[1]$ |
| $(2,3)$ | $z_{47}[1]$ |
| $(2,7)$ | $z_{44}[1], z_{45}[2], z_{48}[1]$ |
| ● $(2,8)$ | $z_{43}[1], z_{44}[3], z_{46}[1], z_{47}[1]$ |
| $(3,4)$ | $z_{48}[1]$ |
| ○ $(3,5)$ | $z_{41}[3], z_{44}[1], z_{47}[1], z_{48}[1]$ |
| $(3,8)$ | $z_{41}[1], z_{45}[1], z_{46}[2]$ |
| $(4,5)$ | $z_{42}[1], z_{46}[1], z_{47}[2]$ |
| ● $(4,6)$ | $z_{41}[1], z_{42}[2], z_{45}[1], z_{48}[1]$ |

[ ] denotes the *pattern*.
Two ● pairs (or two ○ pairs) gives the minimum number of chosen plaintexts to solve all 6-*th* round keys.

[*pattern*2]

$$z_{4i} = s_{j_1}(s_{j_2}(z_{2i_1} \oplus c_1) \oplus s_{j_2}(z_{2i_1} \oplus c_2) \oplus f(z_{2i_1} \oplus z_{2i_2}) \oplus c_3)$$

[*pattern*3]

$$z_{4i} = s_{j_1}(s_{j_2}(z_{2i_1} \oplus c_1) \oplus s_{j_2}(z_{2i_2} \oplus c_2) \oplus f(z_{2i_1} \oplus z_{2i_2}) \oplus c_3) \qquad (j_1, j_2 = 1,2,3,4),$$

where $c_1, c_2, c_3$ are constant values, calculated from round keys and plaintext bytes except $i_1$-*th* and $i_2$-*th* in $P_R$. $f()$ denotes some function having $z_{2i_1} \oplus z_{2i_2}$ as an input.

These patterns are shown in Table.3 as a number in the bracket[ ]. For example, when we select pairs $(1,2)$ as variable sub-blocks, $z_{46}$ is expressed as follows.

$$\begin{aligned} z_{46} = s_3(&s_1(z_{21} \oplus c_1) \oplus s_2(z_{21} \oplus z_{22} \oplus c_2) \oplus \\ &s_2(z_{21} \oplus z_{22} \oplus c_3) \oplus s_3(z_{21} \oplus z_{22} \oplus c_4) \oplus c_5) \end{aligned} \tag{21}$$

Then, if we choose $f()$ as

$$f(z_{21} \oplus z_{22}) = s_2(z_{21} \oplus z_{22} \oplus c_2) \oplus s_2(z_{21} \oplus z_{22} \oplus c_3) \oplus s_3(z_{21} \oplus z_{22} \oplus c_4), \tag{22}$$

$z_{46}$ is classified to *pattern*1.

The proofs for $\Delta^{(16)} z_{4i} = 0$ are described in Appendix.

# 5 Attack of 6 Round *Camellia* (Basic Attack)

Based on the previous discussion, we can derive an attack equation to determine the 6-*th* round key. For example, when we choose pairs $(1,2)$ as variable sub-blocks, the attack equation is as follows.

$$\bigoplus_{(1,2)} s_3(c_{R6} \oplus k_{66}) = \bigoplus_{(1,2)} \left\{\mathbf{P}^{-1} C_L\right\}_6 \tag{23}$$

Eq.(23) is a vector equation over $GF(2)^8$. It holds with probability $2^{-8}$ for a false key $k_{66}$, and always holds for the true key. Let $K1$ be the number of key bits on which we must perform a brute-force search in the attack. To remove all false keys, the necessary number $M$ of 16-*th* order differential is the one satisfying the following.

$$(2^{-8})^M \times 2^{K1} \ll 1 \tag{24}$$

Here, we have $K1 = 8$. So we choose $M = 2$. This attack requires $2^{16} \times M = 2^{17}$ chosen plaintexts.

In the straight forward calculation, the computational complexity to determine the left side of Eq.(23) is $2^{16}$ S-Box operations for each supposed key $k_{66}$. However we can reduce its cost to $2^8$ S-Box operations by an occurrence table for $c_{R6}$. Because an even time ex-OR sum always becomes 0.

We have $M$ sets of 16-*th* order differentials for checking the correctness of supposed key. On the first check, we can expect $2^8$ possible values of $k_{66}$ to check. On the *i-th* check, there are $2^8 \times 2^{-8i}$ survived false key values, which we must check for its correctness. Thus the complexity is

$$T_s = \sum_{i=0}^{M-1} \left(2^8 \times 2^8 \times 2^{-8i}\right) < 2^8 \times 2^8 \times 2 = 2^{17} \tag{25}$$

S-Box operations.

In addition, we have to consider the complexity to prepare the occurrence table [3]. It equals the complexity of encrypting $2^{16}$ plaintexts in 6 round *Camellia*. So it is estimated as

$$T_b = M \times 2^{16} \times 8 \times 6 \simeq 2^{23} \tag{26}$$

S-Box operations. The complexity to complete this task is $T = \dfrac{T_s + T_b}{8 \times 6} < 2^{20}$ encryptions. We did computer experiments. It(CPU:alpha 21264A 667MHz) took about 0.1[*sec*], which is an average value for 10,000 experiments.

To determine all the 6-*th* round keys, we need 8 attack equations. It needs $8 \times T$ encryptions. If we choose pairs (1,7) and (3,5) as variables, which are marked ○ in Table.3, we can have 4 attack equations for each pairs. So the necessary number of chosen plaintexts are $2 \times M \times 2^{16} = 2^{18}$. We can also use two ● made pairs to make such attack equations.

[3] In general, it can be ignored because it is far smaller than the complexity to solve the attack equation. In this case, however, we can not ignore it.

## 6 Expansion of Attack (I)

We expand the attack described in the previous section. Let's consider the +2R elimination technique. In this case, we have to guess 5 byte keys in 7-*th* round and 1 byte key in 6-*th* round. So $K1 = 48$.

From Eq.(24), we choose $M = 7$ and $m = 2^{16} \times M \simeq 2^{19}$. As previously mentioned, the complexity to determine $k_{6i}$ can be reduced if we use an occurrence table method. In this paper, for the first test, we check the all-possible value of 6-*th* round key $k_{6i}$ using the occurrence table under one guessed keys in 7-*th* round. In the consecutive test, we check the survived keys by a brute-force search, because the expected number of survived keys $k_{6i}$ are $|k_{6i}| = 2^{-8} \times 2^8 = 1$ and the occurrence table method does not work effectively.

The complexity to prepare all ciphertexts for this attack is $T_e = M \times 2^{16} \times 7 \times 8 < 2^{25}$ S-Box operations, because each ciphertext requires the encryption of 7 round *Camellia*. And the complexity for the brute-force search for each guessed key in 7-*th* round is $T_{b1} = 2^{16} \times 5$ and that for each guessed key in 6-*th* and 7-*th* round is $T_{b2} = 2^{16} \times 6$. Let $K2$ be the number of key bits which we have to guess the last $n$ round except the round in which we apply the occurrence table method. In this case, $K2 = 40$. And let $T_1$ be the complexity for the first test, and $T_2$ be the complexity for consecutive test. These are calculated as follows.

$$T_1 = 2^{K2} \times (T_{b1} + 2^8 \times 2^8) = 2^{K2} \times T_{b2} \tag{27}$$

$$T_2 = \sum_{i=0}^{M-2} \left\{2^{K2} \times |k_{6i}| \times T_{b2} \times 2^{-8i}\right\} < 2^{K2} \times |k_{6i}| \times T_{b2} \times 2 \tag{28}$$

The complexity to complete this task is

$$T_s = T_1 + T_2 + T_e < 3 \times 2^{K2} \times T_{b2} \simeq 2^{61}. \tag{29}$$

S-Box operations. This is $T = T_s/(8 \times 7) = 2^{57}$ encryptions.

Similarly, we can conduct +3R elimination technique. In this case, $K2 = 104$, because we have to guess 64-bit round keys at 8-*th* round in addition to the guessed keys on +2R elimination technique. Therefore, we choose $M = 15$ from Eq.(24) and $m = 2^{16} \times M = 2^{20}$. It's complexity is

$$T_{b2} = 2^{16} \times 14 \tag{30}$$

$$T_s < 3 \times 2^{K2} \times T_{b2} < 2^{126} \tag{31}$$

S-Box operations, which is equal to $T = T_s/(8 \times 8) = 2^{120}$ encryptions.

The necessary number of plaintexts and complexity for +4R,+5R elimination technique can also be calculated similarly(see Table.1).

## 7 Expansion of Attack (II)

Our attack can also be improved by eliminating the first $n$ round. For *Camellia*, this kind of technique was used by Yeom et. al.[15].

### 7.1 Attack of 7 Round *Camellia* Using −1R Elimination Technique

In this section, we present an attack of 7 round *Camellia*, which needs to guess 2 bytes of first round keys. Although we can adopt any sub-block pairs in Table.3, we choose pairs (1,2) as variables as an example.

Let's review the attack of 6 round using +1R elimination technique. In that attack, to determine the 6-*th* round key $k_{66}$, we use the plaintext, which has *constant* property except first and the second byte in $P_R$, which has *all* property. We express this condition as

$$\begin{cases} P_L = (\text{"}c\text{"}, \cdots, \text{"}c\text{"}) \\ P_R = (\text{"}v_1\text{"}, \text{"}v_2\text{"}, \text{"}c\text{"}, \cdots, \text{"}c\text{"}) \end{cases} \tag{32}$$

where "$v_1$", "$v_2$" denote a variable sub-block, and "$c$" denote a constant sub-block.

To apply −1R elimination technique, the input of the second round $X_{L2}, X_{R2}$ must satisfy the following condition.

$$\begin{cases} X_{L2} = (\text{"}c\text{"}, \cdots, \text{"}c\text{"}) \\ X_{R2} = (\text{"}v_1\text{"}, \text{"}v_2\text{"}, \text{"}c\text{"}, \cdots, \text{"}c\text{"}) \end{cases} \tag{33}$$

Let $v_1, v_2$ be the actual values of "$v_1$", "$v_2$", respectively. Note that $X_{R2}$ is also an input for the first round F function. The output of the first round F function can be expressed as follows by using $k_{11}, k_{12}$, which are the first and second byte of the first round keys, and some constants $c_i (i = 0 \sim 8)$.

$$\begin{cases} y_{11} = s_1(v_1 \oplus k_{11}) \oplus c_1 \\ y_{12} = s_1(v_1 \oplus k_{11}) \oplus s_2(v_2 \oplus k_{12}) \oplus c_2 \\ y_{13} = s_1(v_1 \oplus k_{11}) \oplus s_2(v_2 \oplus k_{12}) \oplus c_3 \\ y_{14} = s_2(v_2 \oplus k_{12}) \oplus c_4 \\ y_{15} = s_1(v_1 \oplus k_{11}) \oplus s_2(v_2 \oplus k_{12}) \oplus c_5 \\ y_{16} = s_2(v_2 \oplus k_{12}) \oplus c_6 \\ y_{17} = c_7 \\ y_{18} = s_1(v_1 \oplus k_{11}) \oplus c_8 \end{cases} \tag{34}$$

In the above formula, independent variables are

$$\begin{cases} \alpha_1 = s_1(v_1 \oplus k_{11}) \\ \alpha_2 = s_2(v_2 \oplus k_{12}) \\ \alpha_3 = s_1(v_1 \oplus k_{11}) \oplus s_2(v_2 \oplus k_{12}). \end{cases} \tag{35}$$

From the guessed key $k_{11}, k_{12}$, the right half of the plaintext $P_R$ can be calculated as follows.

$$P_R = (\alpha_1, \alpha_3, \alpha_3, \alpha_2, \alpha_3, \alpha_2, c_9, \alpha_1) \tag{36}$$

If the guessed keys $k_{11}, k_{12}$ are true, the input of the second round satisfies the condition(Eq.(33)) and the attack equation is

$$\bigoplus_{v_1,v_2} s_3(c_{R6} \oplus k_{76}) = \bigoplus_{v_1,v_2} \left\{\mathbf{P}^{-1} C_L\right\}_6, \tag{37}$$

where $\bigoplus_{v_1,v_2}$ denote the sum over variables $v_1, v_2$.

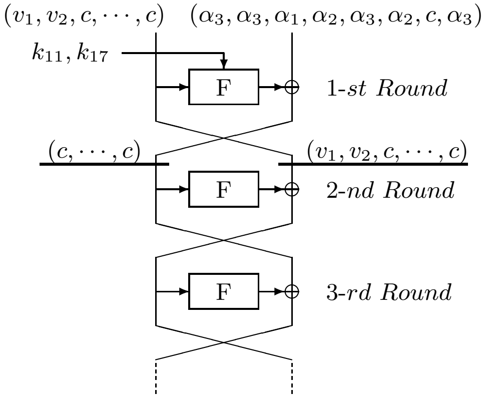

$(v_1, v_2, c, \cdots, c)$
$(\alpha_3, \alpha_3, \alpha_1, \alpha_2, \alpha_3, \alpha_2, c, \alpha_3)$
$k_{11}, k_{17}$
F
1-*st Round*
$(c, \cdots, c)$
$(v_1, v_2, c, \cdots, c)$
F
2-*nd Round*
F
3-*rd Round*

**Fig. 6.** −1R elimination technique

### 7.2 The Number of Plaintext and Complexity

Let $K3$ be the number of key bits to determine the chosen plaintext. In this attack, we have to guess keys, $k_{11}, k_{12}$ in first round and $k_{76}$ in the attack equation. Therefore $K1 = 24, K3 = 16$, respectively. The number of 16-*th* order differential pairs $M$ have to satisfy $(2^{-8})^M \times 2^{K1} \ll 1$ from Eq.(24). Here, we choose $M = 4$. We need $2^{16} \times M$ chosen plaintexts for each candidate keys in the first round. So the necessary number of chosen plaintexts are $m = 2^{K3} \times 2^{16} \times M = 2^{36}$.

Since it needs 2 S-Box operations to choose the plaintext, the complexity to make the occurrence table is $T_b = 2^{16} \times (8 \times 7 + 2) < 2^{22}$ S-Box operations. At the beginning, we must determine a chosen plaintext by guessing first round keys $k_{11}, k_{12}$. Since we have to conduct a brute-force search for those 2 byte keys, the complexity for this attack is

$$T_s < 2^{K3} \times \left(M \times T_b + 2^8 \times 2^8 \times 2\right) \simeq 2^{40} \tag{38}$$

S-Box operations, and $T = T_s/(8 \times 7) < 2^{34}$ encryptions. We conducted computer experiments. It took about 1.5[$h$], which is an average value of 30 trials(CPU:alpha 21264A 667MHz).

### 7.3 −2R Elimination Technique

It is possible to expand the above technique to −2R elimination one. In this case, we have to guess first round keys, $k_{11}, \cdots, k_{16}, k_{18}$ and first and second byte keys $k_{21}, k_{22}$ in the second round, and control the plaintext so that the input of the second round equals to the plaintext for +1R elimination technique(Fig.6). As shown in Fig.7, let $\beta_1, \cdots, \beta_8$ be sub-blocks, which are calculated from $\alpha_1, \alpha_2, \alpha_3$, when we guess the first round keys, $k_{11}, \cdots, k_{16}, k_{18}$. The chosen plaintext can be expressed as

$$\begin{cases} P_R = (\alpha_1, \alpha_3, \alpha_3, \alpha_2, \alpha_3, \alpha_2, c_7, \alpha_1) \\ P_R = (\beta_1, \cdots, \beta_8). \end{cases} \tag{39}$$

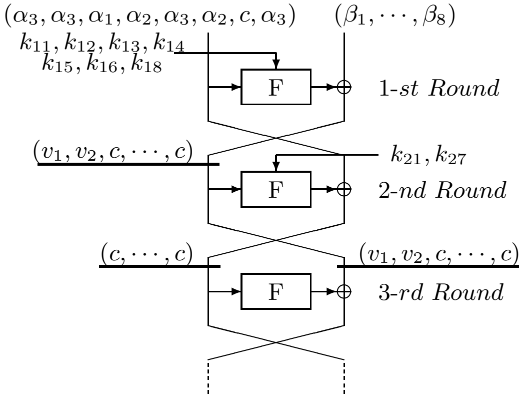

$(\alpha_3, \alpha_3, \alpha_1, \alpha_2, \alpha_3, \alpha_2, c, \alpha_3)$
$(\beta_1, \cdots, \beta_8)$
$k_{11}, k_{12}, k_{13}, k_{14}$
$k_{15}, k_{16}, k_{18}$
F
1-st Round
$(v_1, v_2, c, \cdots, c)$
$k_{21}, k_{27}$
F
2-nd Round
$(c, \cdots, c)$
$(v_1, v_2, c, \cdots, c)$
F
3-rd Round

**Fig. 7.** −2R elimination technique

In this attack, $K1 = 80, K3 = 72$, because it needs to guess 9 byte keys in the first and the second round, and 1 byte key in the 7-*th* round. Therefore $M = 11$, because it must hold $(2^{-8})^M \times 2^{80} \ll 1$. The necessary number of plaintexts are $m = 2^{72} \times 2^{16} \times M = 2^{93}$. And the complexity for this attack is

$$T_b = 2^{16} \times (8 \times 8 + 9) < 2^{23} \tag{40}$$

$$T_s < 2^{K3} \times (M \times T_b + 2^8 \times 2^8 \times 2) < 2^{99} \tag{41}$$

S-Box operations, and $T = T_s/(8 \times 8) = 2^{93}$ encryptions.

### 7.4 +*n*R and −*n*R Elimination Technique

−$n$R technique can be used with +$n$R elimination technique simultaneously. From Eq.(24), the number $M$ of 16-*th* order differential to complete an attack satisfies $(2^{-8})^M \times 2^{K1}$ and it needs $m = 2^{K3} \times 2^{16} \times M$ chosen plaintexts. For each supposed key in the first $n$ round, we must conduct +$n$R elimination technique. Thus the complexity to complete this attack is

$$T_s < 3 \times 2^{K2+K3} \times T_{b2} \tag{42}$$

S-Box operations.

Now, we estimate the attack of 11 round *Camellia*. When −2+4R elimination technique is applied, it needs to guess 9 byte keys in the first 2 rounds and 22 byte keys in the last 4 rounds. Thus $K1 = 248, K2 = 168, K3 = 72$. From Eq.(24), we choose $M = 32$ and $m = 2^{16} \times 2^{K3} \times M = 2^{93}$ chosen plaintexts. The complexity to calculate the left side of the attack equation from ciphertexts is $T_{b2} = 2^{16} \times (8 \times 2 + 5 + 1) = 2^{16} \times 22$ S-Box operations. The complexity to complete this attack is

$$T_s < 3 \times 2^{K2+K3} \times T_{b2} \simeq 2^{262.1} \tag{43}$$

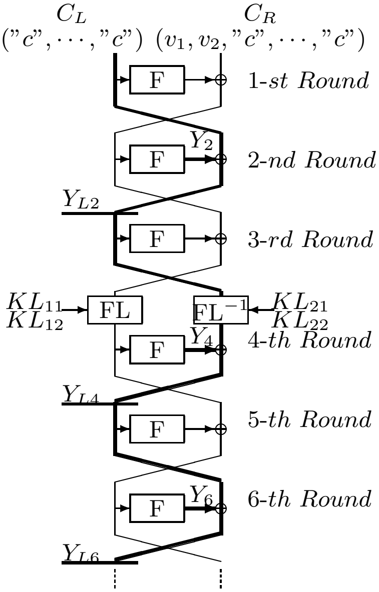


**Fig. 8.** 9 round *Camellia*(From a ciphertext)

S-Box operations and so $T = T_s/(8 \times 11) = 2^{255.6}$ encryptions[4]. Therefore, 11 round *Camellia* under a 256 bit secret key can be attacked with less complexity than brute-force search for a secret key.

## 8 Attack of *Camellia* with FL Function Using Chosen Ciphertexts

In this section, we analyze *Camellia* with FL function using chosen ciphertexts. Let $KL_{11}, KL_{12}, KL_{21}$, and $KL_{22}$ be keys, which are inputs to FL/FL$^{-1}$ function as shown in Fig.3 and 8. $X_{Li}, X_{Ri}$ denote *i-th* round inputs, and $Y_{Li}, Y_{Ri}$ denote these outputs.

Let's consider +4R elimination technique. Then, FL function is inserted between the third and the fourth round since it is inserted every 6 round from the plaintext.

[4] If we adopt precise value for $T_2$ in Eq.(28) as

$$T_2 = \sum_{i=0}^{M-2} \left\{ 2^{K2} \times |k_{6i}| \times T_{b2} \times 2^{-8i} \right\},$$

this complexity is slightly less than $2^{255}$ encryptions.

**Table 4.** Input variable sub-block bytes pair for $\Delta^{(16)} z_{4i} = 0$ (With FL function)

| $(i_1 i_2)$ | observation byte ! ! | $(i_1 i_2)$ | observation byte ! ! |
|---|---|---|---|
| $(1,6)$ | $z_{48}$ | $(3,5)$ | $z_{47}$ |
| $(1,7)$ | $z_{45}$ | $(3,8)$ | $z_{46}$ |
| $(2,7)$ | $z_{45}$ | $(4,5)$ | $z_{47}$ |
| $(2,8)$ | $z_{46}$ | $(4,6)$ | $z_{48}$ |

As shown in Fig.8, the following equation holds.

$$\begin{aligned} & Y_{L6} = Y_6 \oplus Y_4 \oplus FL^{-1}(Y_{L2}; KL_{21}, KL_{22}) \\ \Longleftrightarrow\ & Z_6 \oplus Z_4 = \mathbf{P}^{-1}\{Y_{L6} \oplus FL^{-1}(Y_{L2}; KL_{21}, KL_{22})\} \\ \Longleftrightarrow\ & z_{6i} = z_{4i} \oplus \{\mathbf{P}^{-1} Y_{L6}\}_i \oplus \{\mathbf{P}^{-1} FL^{-1}(Y_{L2}; KL_{21}, KL_{22})\}_i \\ & \qquad (i = 1 \sim 8) \end{aligned} \tag{44}$$

Since $z_{6i} = s_j(x_{6i} \oplus k_{6i})$, when we choose pairs $(i_1, i_2)$ as a variable sub-blocks and calculate 16-*th* order differential of the above equation, it gives as

$$\begin{aligned} \bigoplus_{(i_1,i_2)} s_j(x_{6i} \oplus k_{6i}) = \bigoplus_{(i_1,i_2)} \left\{\mathbf{P}^{-1} Y_{L6}\right\}_i \oplus \Delta^{(16)} z_{4i} \oplus \\ \Delta^{(16)} \left\{\mathbf{P}^{-1}\{FL^{-1}(Y_{L2}; KL_{21}, KL_{22})\}\right\}_i, \end{aligned} \tag{45}$$

where

$$Y_{L2} = Y_2 \oplus C_L = \mathbf{P} Z_2 \oplus C_L. \tag{46}$$

Let's consider 8-*th* order differential for *i-th* byte in $C_R$. Then each byte of $Z_2$ has *all* or *constant* property and $C_L$ has *constant* property. Since P and FL functions are linear functions, the third term in Eq.(45) become 0 for any 16-*th* order differential. Thus if $\Delta^{(16)} z_{4i} = 0$, we have

$$\bigoplus_{(i_1,i_2)} s_j(x_{6i} \oplus k_{6i}) = \bigoplus_{(i_1,i_2)} \left\{\mathbf{P}^{-1} Y_{L6}\right\}_i. \tag{47}$$

This is the same attack equation as Eq.(19), which is the attack equation without FL function.

We searched for variable sub-block pairs, which satisfy $\Delta^{(16)} z_{4i} = 0$. Eight pairs satisfy such condition, which are shown in Table.4. Therefore, with Eq.(47) and +4R elimination technique, we can conduct the attack of 9 round *Camellia* with FL function when we choose one of these pairs as a variable. The complexity and the necessary number of chosen plaintexts are the same as that for without FL function because Eq.(47) is the same attack equation for the case of without FL function. Using $-1, -2$R elimination technique, 10 round and 11 round *Camellia* with FL function is attackable with the same complexity as the attack of variant without FL function, respectively(see Table.1).

## 9 Conclusion

In this paper, we present a new attack of *Camellia* using 16-*th* order differential. We have shown that 11 round *Camellia* without FL function can be attacked. Moreover, we did computer experiments of attacks for 6 round and 7 round *Camellia*. They took about 0.2[*sec*] and 1.5[*h*], respectively. Using chosen ciphertexts, we have shown that 11 round *Camellia* with FL function is attackable with less complexity than a brute-force search for a 256-bit secret key.

## References

1. K.Aoki, T.Ichikawa, M.Kanda, M.Matsui, S.Moriai, J.Nakajima, and T.Tokita, "The 128-Bit Block Cipher *Camellia*," IEIEC Trans. Fundamentals, Vol.E85-A, No.1, pp.11-24, Jan, 2002.
2. K.Aoki, T.Ichikawa, M.Kanda, M.Matsui, S.Moriai, J.Nakajima, and T.Tokita, "*Camellia* −A 128-Bit Block Cipher *Camellia*,"Technical Report of IEICE, ISEC2000.
3. N.Furguson, J.Kelsey, S.Luck, B.Schneier, M.Stay, D.Wagner, and D.Whiting, "Improved Cryptanalysis of Rijndael," Seventh Fast Software Encryption Workshop, 2000.
4. Y.He, and S.Quing, "Square Attack on Reduced Round *Camemllia* Cipher," ICISC 2001, LNCS 2229, Springer-Verlag, pp.213-230, 2000.
5. T.Iwata, and K.Kurosawa,"Probabilistic Higher Order Differential Attack and Secure Boolean Functions",The 2000 Symposium on Cryptography and Information Security, SCIS2000-A-46, Okinawa, Japan, Jan.26-28, 2000.
6. M.Kanda, and T.Matsumoto, "On the Security of Feistel Cipher with SPN Round Function against Differential, Linear, and Truncated Differential Cryptanalysis," IEIEC Trans. Fundamentals, Vol.E85-A, No.1, pp.25-37, Jan, 2002.
7. T.Kawabata, and T.Kaneko, "A Study on Higher Order Differential Attack of *Camellia*," Second Open, NESSIE Workshop, Londom, U.K, Sep. 2001.
8. Lar R. Knudsen "The Interpolation Attack on Block Cipher," Fast Software Encryption 4-th International Workshop, LNCS.1008, Springer-Verlag. Berlin, 1996.
9. X.Lai, "Higher Order Derivatives and Differential Cryptanalysis," Communications and Cryptography, pp.227-233, Kluwer Academic Publishers, 1994.
10. S.Lee, S.Hong, S.Lee, J.Lim, and S.Yoon, "Truncated Differential Cryptanalysis of *Camellia*," ICISC2001.
11. T.Shimoyama, S.Moriai, and T.Kaneko, "Higher Order Differential Attack of a CAST Cipher," Fast Software Encryption 4-th International Workshop, LNCS.1372, Springer-Verlag. Berlin, 1996.
12. T.Shirai, S,Kanamaru, and G.Abe, "Improved Upper Bounds of Differential and Linear Characteristic Probability for *Camellia*" Fast Software Encryption 2002, FSE2002, pp.123-137, Lenven, Belgium, Feb, 2002.
13. M.Takeda, and T.Kaneko, "A Study for Controled Higher Order Differential Cryptanalysis of *Camellia*," The 2002 Symposium on Cryptography and Information Security, SCIS2002, Shirahama, Japan, Jan.29-Feb.1, 2002. (in Japanese).
14. H.Tanaka, K.Hisamatsu, and T.Kaneko, "Strength of MISTY1 without FL function for Higher Order Differential Attack," Applied Algebra, Algebraic Algorithm and Error Correcting Codes Symposium(AAECC13), LNCS.1719 pp.221-230, 1999.

15. Y.Yeom, S.Park, and I.Kim, "On the Security of *Camellia* against the Square Attack," Fast Software Encryption 2002, FSE2002, pp.84-93, Lenven, Belgium, Feb, 2002.

## A Proof of $\boldsymbol{\Delta z_{4i}^{(16)} = 0}$

The reason for $\Delta z_{4i}^{(16)} = 0$ can be shown as follows.
**Proof for** *pattern*1

When we choose $i_1, i_2$-*th* byte in the plaintext $P_R$ as variable sub-blocks, 16-*th* order differential of $z_{4i}$ can be expressed as follows.

$$\Delta^{(16)} z_{4i} = \bigoplus_{(i_1,i_2)} \{s_{j_1}(s_{j_2}(z_{2i_1} \oplus c_1) \oplus f(z_{2i_1} \oplus z_{2i_2}) \oplus c_2\}, \tag{48}$$

where

$$z_{2i_1} = s_j(p_{Ri_1} \oplus c_8), z_{2i_2} = s_j(p_{Ri_2} \oplus c_9).$$

Since S-Boxes of *Camellia* are bijective functions, the sum over $i_1, i_2$-*th* byte in $P_R$ $\bigoplus_{(i_1,i_2)}$ equals to the sum over $z_{2i_1}, z_{2i_2}$. Futhermore, let

$$\alpha = z_{2i_1} \oplus c_1, \beta = z_{2i_1} \oplus z_{2i_2}. \tag{49}$$

We can replace the sum over $z_{2i_1}$ and $z_{2i_2}$ with the sum over $\alpha$ and $\beta$.

$$\begin{aligned} \Delta^{(16)} z_{4i} &= \bigoplus_{\alpha,\beta} \{s_{j_1}(s_{j_2}(\alpha) \oplus f(\beta) \oplus c_2\} \\ &= \bigoplus_{\beta} \left\{ \bigoplus_{\alpha} s_{j_1}(s_{j_2}(\alpha) \oplus f(\beta) \oplus c_2) \right\} \end{aligned} \tag{50}$$

Let's estimate the value in parentheses{ }. For a constant $\beta$, this is the sum over $\alpha$ for the function $g(\alpha, \beta)$ expressed as follows.

$$g(\alpha, \beta) = s_{j_1}(s_{j_2}(\alpha) \oplus f(\beta) \oplus c_2) \tag{51}$$

From Eq.(49), $\alpha$ has *all* property as inputs $s_{j_2}()$. Thus the value equals to 8-*th* order differential of $g()$ with respect to $\alpha$.

$$\bigoplus_{\alpha} \{s_{j_1}\left(s_{j_2}(\alpha) \oplus f(\beta) \oplus c_2\right)\} = \Delta^{(8)} g(\alpha, \beta) \tag{52}$$

Since $\beta, c_2$ are constant values and S-Boxes of *Camellia* are bijective functions, the output from $g()$ has *all* property. From Property3, we have $\Delta^{(8)} g(\alpha, \beta) = 0$. Therefore, we have the following.

$$\begin{aligned} \Delta^{(16)} z_{4i} &= \bigoplus_{\beta} \left\{ \Delta^{(8)} g(\alpha, \beta) \right\} \\ &= 0 \end{aligned} \tag{53}$$

□

Due to the limited space, we omit the proof for *pattern*2 and *pattern*3. With a similar procedure, they can also be easily proven.