Vol.2013-DPS-154 No.6
Vol.2013-CSEC-60 No.6
2013/3/14

単純化ストリーム暗号K2のクロック制御を無効化したGD攻撃

伊藤 竜馬[1,a)]　岩切 宗利[1,b)]

概要：K2 は，128 ビット秘密鍵と 128 ビット初期ベクトルの計 256 ビットを入力とし，1 サイクル当たり 64 ビットの鍵ストリームを出力するストリーム暗号である．既存研究では，K2 の構成要素である置換関数と転置関数を省略した単純化モデルに対して，GD(Guess and Determine) 攻撃に基づくビットスライス解読手法が提案されている．本研究では，従来手法に加えて，K2 のクロック制御ビットを推定することなく解読する手法について検討した．提案手法を用いることで，従来手法による計算量を削減できた．

キーワード：ストリーム暗号，K2，GD 攻撃

Guess and Determine Attack without Clock Control Estimate on Simplified Stream Cipher K2

Ryoma Ito[1,a)]　Munetoshi Iwakiri[1,b)]

Abstract: K2 is a stream cipher, outputs a 64-bit keystream on each cycle with a 256-bit input which consists of a 128-bit secret key and a 128-bit initial vector. In the previous study, Ooshima and Iwakiri suggested the bit slice cryptanalysis based on Guess and Determine Attack on simplified K2 which made except for Substitution and Permutation Functions. In this study, we investigated a method of Guess and Determine Attack without clock control estimate, could reduce computational complexity as compared with the previous method.

Keywords: Stream Cipher，K2，Guess and Determine Attack

1. はじめに

スマートフォンを始めとする携帯型情報通信端末の急速な発達に伴い，大容量データを高速に処理できる暗号アルゴリズムの重要性が高まっている．このような背景のもと，共通鍵暗号方式の一種であるストリーム暗号 [1–6] が注目されている．代表的なストリーム暗号は，秘密鍵を入力して生成される擬似乱数と平文の排他的論理和をとることで暗号文を出力する方式である．通常は，ビット単位もしくはバイト単位で逐次暗号化できるため，高速な暗号通信が可能となっている．

K2 は，2006 年に清本，田中，櫻井によって提案されたストリーム暗号 [7–9] であり，128 ビット秘密鍵と 128 ビット初期ベクトルの計 256 ビットを入力とし，1 サイクル当たり 64 ビットの鍵ストリームを生成する．本報告に示す K2 とは，文献 [7] に示されたストリーム暗号である．

K2 の特徴として，線形フィードバックシフトレジスタ（LFSR:Linear Feedback Shift Register）とクロック制御が挙げられる．K2 のような LFSR 型ストリーム暗号は LFSR から得られる出力値を非線形関数で処理することにより鍵ストリームを生成する [6]．代表的な LFSR 型ストリーム暗号として，K2 のほかに SNOW [1] がある．クロック制御は，LFSR のフィードバック値を不規則に変化させる処理であり，LFSR 型ストリーム暗号の安全性向上に有効で

[1] 防衛大学校情報工学科
National Defense Academy,
Yokosuka, Kanagawa 239-8686, Japan
[a)] f12010@nda.ac.jp
[b)] iwak@nda.ac.jp

あることが示されている [7, 10, 11]．代表的なクロック制御型ストリーム暗号として，K2 のほかに A5 [2, 6] がある．

ストリーム暗号への攻撃手法として識別攻撃，相関攻撃等があり，これらの攻撃を用いて K2 の安全性評価が行われているが，未だ完全に解読できたという報告はない．

これらの攻撃手法の一種である GD（Guess and Determine）攻撃は，既知平文攻撃を前提として，内部状態の一部を推測（Guess）し，残りの部分を内部状態の更新関数，出力関数，鍵ストリーム等を用いて決定（Determine）する攻撃である [11–15]．ビットスライス解読手法とは，各レジスタが 2 ビット以上の値を持つ暗号に対し，1 ビット毎順に解読する手法である．これらの手法を用いることで，文献 [16] の研究では，K2 の構成要素である置換関数と転置関数を省略した単純化モデルの K2 を解読している．

本研究では，文献 [16] で提案された手法に加え，K2 のクロック制御を無効化する手法について検討した．その結果，クロック制御ビットを推測することなく解読することができ，従来手法 [16] に比べて計算量を 2^{-6} 削減できた．

2. K2 の概要

K2 は，128 ビット秘密鍵と 128 ビット初期ベクトルの計 256 ビットを入力とし，1 サイクル当たり 64 ビットの鍵ストリームを生成するストリーム暗号である．各レジスタは 32 ビットである．K2 の鍵ストリーム生成部を**図 1** に示す．

K2 は，2 つの LFSR（LFSR-A，LFSR-B），2 つのメモリ（M1，M2），非線形関数（置換関数，転置関数）で構成され，LFSR-A の出力値を LFSR-B の制御に用いるためのクロック制御部を有する．

LFSR-A と LFSR-B の既約多項式は，

$$f_A(x) = \alpha_0 x^5 + x^2 + 1 \tag{1}$$

$$f_B(x) = \alpha_1^{c1} x^{11} + x^9 + x^6 + \alpha_2^{c2} x^3 + 1 \tag{2}$$

である．時刻 t におけるクロック制御部からの出力値 $c1$，$c2$ は，

$$c1_t = 1 - A_{t+2}[30] \tag{3}$$

$$c2_t = A_{t+2}[31] \tag{4}$$

である．ここで $A_{t+2}[30]$ と $A_{t+2}[31]$ は，時刻 $t+2$ における LFSR-A の 30 ビット目と 31 ビット目の値を表す．31 ビット目はレジスタの最上位ビットである．クロック制御部からの出力値により，LFSR-B の既約多項式が制御される．出力鍵ストリーム Z_t は，

$$Z_t = (B_{t+4} + M2_t \oplus A_t \oplus B_{t+9}, \\ B_{t+7} + M1_t \oplus A_{t+4} \oplus B_t) \tag{5}$$

である．‘+’ は全加算，‘⊕’ は半加算を表す．Z_t の右辺第 1 項は左側から出力される 32 ビット，第 2 項は右側から出力される 32 ビットである．

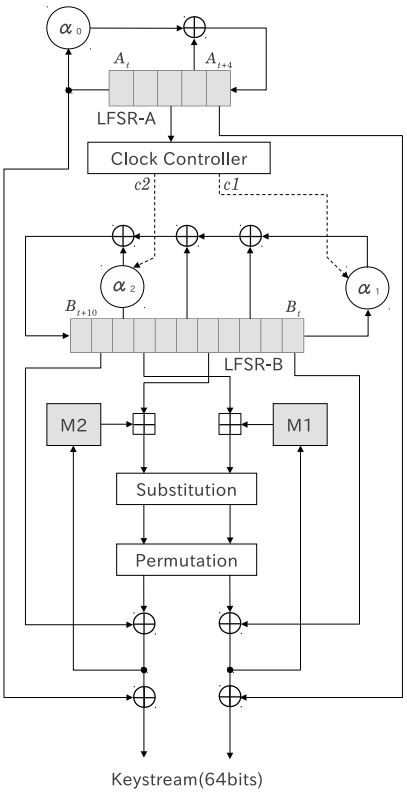


図 1 K2 の鍵ストリーム生成部

クロック制御部で制御される LFSR-B からの 2 つの出力値と M1，M2 の全加算，置換（Substitution），転置（Permutation）の処理を行う．その後，LFSR-B の出力値との半加算により M1，M2 が更新され，さらに LFSR-A の出力値との半加算により鍵ストリーム Z_t（計 $32 \times 2 = 64$ ビット）が出力される．これが K2 の 1 サイクル処理の概要である．

3. 1 ビット単純化 K2 の解読

3.1 ビットスライス解読手法

文献 [16] に，1 ビット単純化 K2 に対して GD 攻撃に基づくビットスライス解読手法が提案されている．このビットスライス解読手法の概要を**図 2** に示す．

図 2 は，レジスタ 5 個で構成された LFSR-A の例である．各レジスタの最下位ビット（LSB）のみで構成される 1 ビットスライスが図 2 （a）であり，その上位ビットが図 2 （b），各レジスタの最上位ビット（MSB）のビットスライスが図 2 （c）である．

各ビットスライスに分割された LFSR-A の中で，最下位ビットスライスから順に解読し，その結果を手がかりとして上位ビットスライスを逐次解読する．最上位ビットスライスを解読した時点で，LFSR-A の値を全て解読したこと

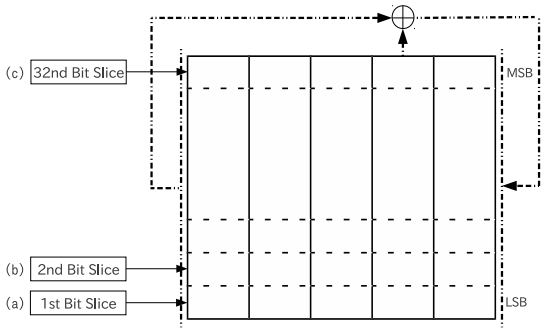


図 2 ビットスライス解読手法の概要

になる．

3.2 1 ビット単純化モデル

ビットスライス解読手法を K2 に適用するため，各レジスタが 1 ビットの単純化 K2 を次のとおり定義する．**図 3** はこの単純化モデルの処理フローである．

この単純化モデルでは，入出力の関係が 1 対 1 対応の置換関数（α_0,α_1,α_2 を含む）と転置関数を省略した．出力される鍵ストリームは，それぞれの関数に対する入力に依存するため，これらの関数を省略しても K2 の処理の流れに変化はない．また，各レジスタが 1 ビットであるため，全加算回路を半加算回路に置き換えた．

K2 は，クロック制御ビットを 2 ビット必要とするため，本研究では，文献 [16] に示されたモデルと異なり，図 3 のとおり乱数生成器（Random Number Generator）を用いて，クロック制御用の 2 ビットを生成させた．

このように構成された 1 ビット単純化 K2 では，各サイクル 2 ビットの鍵ストリームを生成できる．

LFSR-A と LFSR-B の既約多項式は，

$$f_A(x) = x^5 + x^2 + 1 \tag{6}$$

$$f_B(x) = c1x^{11} + x^9 + x^6 + c2x^3 + 1 \tag{7}$$

である．K2 の置換関数である α_0,α_1,α_2 を省略し，クロック制御ビットの処理を単純化した．乱数生成器から生成するクロック制御用の 2 ビットを $r1$，$r2$ とすると，$c1$，$c2$ の値は，

$$c1_t = r1_t \tag{8}$$

$$c2_t = r2_t \tag{9}$$

となる．出力鍵ストリーム Z_t は，

$$Z_t = (B_{t+4} \oplus M2_t \oplus A_t \oplus B_{t+9}, \\ B_{t+7} \oplus M1_t \oplus A_{t+4} \oplus B_t) \tag{10}$$

である．この右辺第 1 項は，左側から出力される 1 ビット，第 2 項は右側から出力される 1 ビットである．

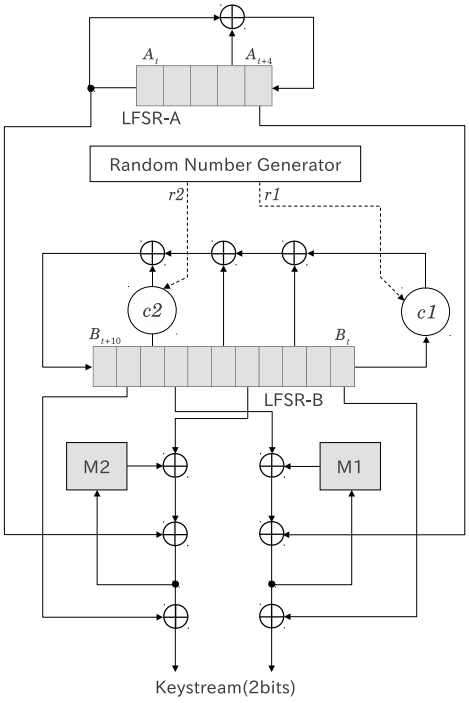


図 3 1 ビット単純化 K2（提案モデル）

3.3 提案手法による GD 攻撃

提案手法による 1 ビット単純化 K2 への GD 攻撃手法を**図 4** に示す．

step1. 図 4 (a) の上図 ($t = 0$) において，G と表記のある箇所，すなわち $\{A_t, A_{t+3}, A_{t+4}, B_t, B_{t+9}, M1_t, M2_t\}$ の 7 箇所の値を推測する．その結果，N と表記のある箇所，すなわち $\{B_{t+4}, B_{t+7}\}$ の 2 箇所が式 (10) より求められる．これを 1 サイクル動かした状態 ($t = 1$) が図 4 (a) の下図であり，式 (6) より A_{t+4} が求められる．D と表記のある箇所が決定したレジスタの内部状態であり，step1 では，7 箇所の推測で 8 箇所の値が定まることを示している．

step 2. 図 4 (b) のように，$\{A_t, B_t, B_{t+9}\}$ を推測することで，$\{B_{t+4}, B_{t+7}\}$ の 2 箇所が式 (10) より求められる．これを 1 サイクル動かす ($t = 2$) と，式 (6) より A_{t+4} が求められる．したがって，step2 では，3 箇所の推測で計 12 箇所の値が定まる．

step 3. 図 4 (c) のように，$\{A_t, B_{t+9}\}$ を推測することで，$\{B_t, B_{t+4}\}$ の 2 箇所が式 (10) より求められる．これを 1 サイクル動かす ($t = 3$) と，式 (6) より A_{t+4} が求められる．したがって，step3 では，2 箇所の推測で計 15 箇所の値が定まる．

step 4. 図 4 (d) のように，N と表記された箇所，すなわち $\{B_t, B_{t+9}\}$ は，式 (10) より求められる．これを 1 サイクル動かす ($t = 4$) と，式 (6) より A_{t+4} が求められる．したがって，step4 では，推測せずに計 16 箇所の値が定まる．

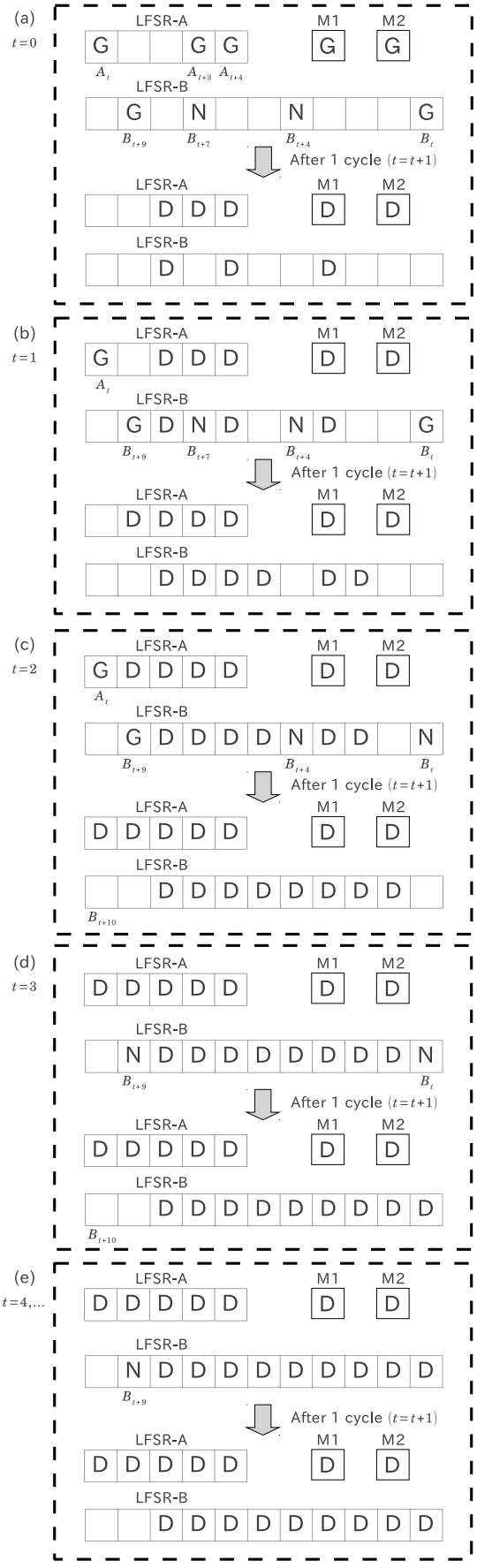


図 4 1 ビット単純化 K2 への GD 攻撃

step 5. 図 4（e）のように，N と表記された箇所，すなわち B_{t+9} は，式（10）より求められる．これを 1 サイクル動かす（$t = t + 1$）と，式（6）より A_{t+4} が求められる．したがって，step5 では，推測せずに計 16 箇所の値が定まる．

時刻 $t = 4$ 以降は，step5 の手順を繰り返すことにより，B_{t+9} が定まる．

既知平文攻撃により得た鍵ストリームと，推測した内部状態から出力した鍵ストリームを比較することで，推測した内部状態の判定を行なう．全ての鍵ストリームが一致すれば解読成功となる．その後，推測した値と決定した値を用いて，初期内部状態を復元する．提案手法では，18 ビット中 12 ビットを推測することで，残りの 6 ビットを決定した．

3.4 考察

図 3 の 1 ビット単純化 K2 と**図 5** に示した文献 [16] における単純化モデルの処理には，クロック制御ビットの生成法に違いがある．

従来モデルでは，A_{t+2} の値によりクロック制御に用いる 1 ビットを定め，$c1$，$c2$ の値を，

$$c1_t = 1 - A_{t+2} \tag{11}$$

$$c2_t = A_{t+2} \tag{12}$$

としている．このため，文献 [16] の研究では，クロック制御ビットを確定させてから攻撃する手法が示されている．

通常，LFSR-B は，式（7）のようにクロック制御ビッ

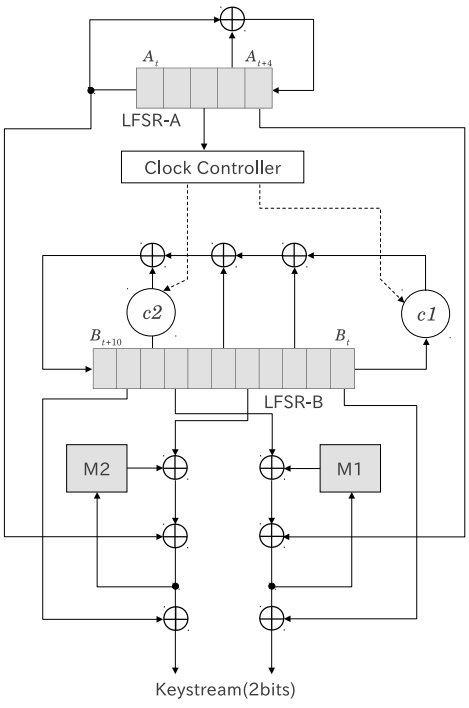


図 5 1 ビット単純化 K2（従来モデル）

トを用いて更新されるが，図 4（d）と式（10）右辺第 1 項から，$\{A_t, B_{t+4}, M2_t\}$ が定まっているため，B_{t+9} を決定できる．その後，図 4（e）を繰り返すことにより，B_{t+10} は定まらないが，クロック制御ビットの値に依存しないで LFSR-B を更新できる．すなわち，本研究では，クロック制御ビットを推定せずに攻撃する手法を示した．

ただし，1 ビット単純化 K2 においては，提案手法も従来手法 [16] も推測するビット数は同じであるため，探索する組み合わせは，どちらの手法も 2^{12} 通りである．したがって，提案手法による 1 ビット単純化 K2 に対する計算量 O_1 は，

$$O_1 = 2^{12} \tag{13}$$

となる．

4. 32 ビット単純化 K2 の解読

4.1 32 ビット単純化モデルの構成

各レジスタを 1 ビットから 32 ビットに拡張した単純化モデルを次のとおり定義する．**図 6** は，その単純化モデルの処理フローである．

1 ビット単純化 K2 との相違点は，LFSR-B からの出力と M1，M2 を半加算していた箇所が全加算に置き換わった点と，クロック制御ビットを乱数生成器から得るのではなく，K2 と同様に A_{t+2} のレジスタから 2 ビットを得る点である．

LFSR-A と LFSR-B の既約多項式，クロック制御ビット $c1$，$c2$ の値，出力鍵ストリーム Z_t は式（3）〜（7）のとおりである．

4.2 全加算回路に関する考察

半加算回路から全加算回路に置き換わることで，桁上がりが発生する．ビットスライス解読手法は，最下位ビットスライスから解読するが，最下位ビットスライスでは桁上がりを考慮する必要がないため，1 ビット単純化 K2 に対する攻撃手法を適用できる．しかし，桁上がりが発生すると上位ビットスライスに影響を及ぼすため，2 ビットスライス目以降は桁上がりを考慮した演算の場合分けが必要となる．

B_{t+4} を決定させるための桁上がりを考慮した演算の場合分けを**図 7** に示した．B_{t+4} を決定させるためには，図 7 の $\{b, c, d\}$ の 3 つの値が既知であることが前提である．また，$CarryBit$ は下位ビットスライスからの桁上がりを意味する．$\{b, c, d\}$ の 3 つの値の組み合わせから図 7 の表に示した 8 通りの場合分けにより，一意に値 a，すなわち B_{t+4} を決定できる．式で表すと，

$$a = b \oplus c \oplus d \tag{14}$$

となる．この時，上位ビットスライスへの桁上がりビット c に関しては，図 7 の表を参照して，$\{a, b, d\}$ の 3 つの値の組み合わせから一意に定まる．

このため，32 ビット単純化 K2 に拡張しても，全加算の演算は，場合分けによって対応可能である．

4.3 計算量に関する考察

32 ビット単純化 K2 は，クロック制御ビットを A_{t+2} の最上位 2 ビットから得ている．そのため，最上位ビットスライスを解読するまで，クロック制御ビットが不明確な状態にある．

従来手法 [16] では，クロック制御ビットを確定させてから攻撃する手法であるため，各ビットスライスを解読する

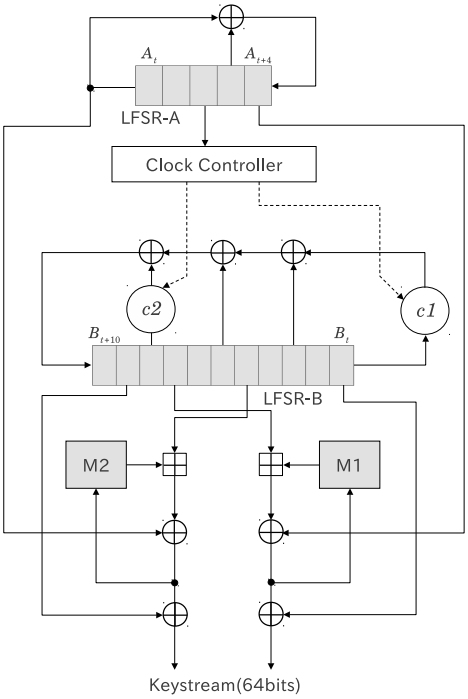
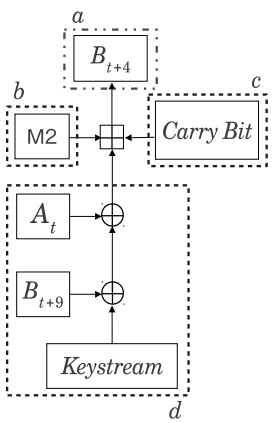


図 6 32 ビット単純化 K2 モデル



a	b	c	d
0	0	0	0
1	0	0	1
1	0	1	0
0	0	1	1
1	1	0	0
0	1	0	1
0	1	1	0
1	1	1	1

図 7 1 ビット演算の桁上がり

ためには，クロック制御ビットを推定させる必要がある．クロック制御ビットは，$(c1, c2) = (0,0), (0,1), (1,0), (1,1)$ の 4 通りに場合分けでき，3 サイクル分の推測，すなわち 2^6 通りの全数探索で解読できる．したがって，従来手法による計算量 O_2 は，

$$O_2 = O_1 \times 2^5 \times 2^6 = 2^{12} \times 2^{11} = 2^{23} \quad (15)$$

となる．O_1 は 1 ビット単純化 K2 に対して解読するための計算量，2^6 はクロック制御ビットの全数探索の計算量，2^5 はビットスライス数を表す．

提案手法による 1 ビット単純化 K2 の GD 攻撃では，クロック制御ビットを推定せずに攻撃する手法であるため，クロック制御ビットを全数探索するための計算量 2^6 を削減できる．したがって，提案手法による計算量 O_3 は，

$$O_3 = O_1 \times 2^5 = 2^{12} \times 2^5 = 2^{17} = \frac{O_2}{2^6} \quad (16)$$

となる．

5. まとめ

本研究では，従来手法 [16] をベースに，単純化 K2 のクロック制御ビットを推定することなく解読する手法について検討した．LFSR 型ストリーム暗号にとって，クロック制御は安全性向上に有効であることが示されていたが [7,10,11]，K2 の単純化モデルに対してクロック制御を無効化できることを示した．その結果，従来手法 [16] よりも計算量を 2^{-6} 削減できた．

参考文献

[1] Ekdahl, P., Johansson, T.: *A new vesion of the stream cipher SNOW*, Proc. of SAC2002, LNCS.2595, pp.47-61,Springer-Verlag(2002).
[2] Shah, J., Mahalanobis, A.: *A New Guess-and-Determine Attack on the A5/1 Stream Cipher*, available from ⟨http://eprint.iacr.org/2012/208.pdf⟩. (accessed 2013-2-8)
[3] Watanabe, D., Furuya, S., Yoshida, H., Preneel, B.: *A new key stream generator MUGI*, Proc. FSE2002, LNCS2365, pp.179-194(2002).
[4] Boesgaard, M., Vesterager, M., Pedersen, T., Christiansen, J., Scavenius, O.: *A new high-performance stream cipher*, FSE2003, LNCS2887, pp.307-329(2003).
[5] Babbage, S., Dodd, M.: *The stream cipher MICKEY 2.0*, The eSTREAM Project(2006).
[6] Ekdahl, P.: *On LFSR based Stream Ciphers -Anarysis and Design-*, LUND UNIVERSITY(2003).
[7] 清本晋作 , 田中俊昭, 櫻井幸一: 効率的なクロック制御を用いたストリーム暗号の設計, ISEC2005-166，pp85-90(2006).
[8] Kiyomoto, S., Tanaka, T., Sakurai, K.: *A Word-Oriented Stream Cipher Using Clock Control*, Proc. SASC2007, pp260-273(2007).
[9] Kiyomoto, S., Tanaka, T., Sakurai,K.: *K2:A stream cipher algorithm using dynamic feedback control*, Proc. SECRYPT2007, pp204-213(2007).
[10] 清本晋作 , 田中俊昭, 櫻井幸一: クロック制御型ストリーム暗号におけるクロック制御の効果について, ISEC2005-112，pp11-16(2005).
[11] 清本晋作 , 田中俊昭, 櫻井幸一: クロック制御型ストリーム暗号に対する推測決定攻撃とその評価, 2004-CSEC-26，pp247-254(2004).
[12] 井手口恒太, 渡辺大: 推測決定攻撃に対する安全性評価の一手法, SCIS2008，pp1-6(2008)
[13] Ahmadi, H., Eghlidos, T., Khazaei, S.: *Improver Guess and Determine Attack on SOSEMANUK*, Tehran,Iran(2006)
[14] Feng, X., Liu, J., Zhou, Z., Wu, C., Feng, D.: *A Byte-Based Guess and Determine Attack on SOSEMANUK*, ASIACRYPT2010, LNCS6477, pp.146-157(2010).
[15] Hawkes, P., Rose, G.: *Guess-and-Determine on SNOW*, SAC2002, LNCS2595, pp.37-46(2003).
[16] 大嶋崇士, 岩切宗利: 単純化ストリーム暗号 K2 のビットスライス解読手法, ISEC2008-141, p253-258(2009).