韓國通信情報保護學會誌
第1卷, 第1號, 1991. 4

招請特輯

스트림 暗號 시스템에 關한 硏究
Study on the Stream Cipher Systems
(1)

李 晩 榮*

스트림 暗號 시스템(stream cipher system)은 同期方式과 自動同期方式으로 區分된다. 本稿에서는 各種 暗號 시스템을 細分化하고 그의 定義, 理論的 解釋, 暗號解讀을 爲해 要求되는 條件 및 暗號文 誤謬訂正 등을 다음 目次에 따라 連載로 記述하고자 한다.

目 次

* 정회원, 漢陽大學校 名譽教授, 本 學會 會長

1. 序 論

스트림 暗號 시스템에는 두가지 形態가 있다. 하나는 키 비트 스트림(key bit stream)이 平文(plaintext)과 獨立的인 것이고 다른 하나는 키 비트 스트림이 平文 또는 暗號文(ciphertext)의 函數로 되는 것이다. 前者는 暗號文을 성공적으로 復號(decipherment)해서 올바른 平文을 뽑아내기 위해 키 비트 스트림과 暗號文 사이에 同期(synchronization)가 必要함으로 同期 스트림 方式(synchronous stream cipher)이라 한다. 後者는 暗號文에 插入(injection) 또는 削除(deletion)된 誤謬 비트(error bit)로 因하여 몇개의 한정된 誤謬가 復號된 平文에 發生하지만 후속되는 暗號文에 異常이 없는 한 올바른 平文으로 다시 복귀되기 때문에 自動同期 스트림 暗號方式(self-synchronizing cipher system)이라 한다.

근래 스트림 暗號 시스템은 거의 軍事 및 外交用으로 汎用되고 있으나 一部 商用으로도 使用되고 있다. 線形歸還置換 레지스터(linear feedback shift register : LFSR)에 의해 만들어지는 符號化 系列인 키 비트 스트림(key bit stream)은 暗號文을 만들기 爲해 平文과 2元合(modulo-2 sum) 된다. 스트림 暗號 시스템에서 LFSR의 役割은 暗號 키를 生成하는 符號器로 사용되기도 하고, LFSR의 入力이 平文인 경우 出力은 直接 暗號文으로 構成해 주는 暗號器(enciphering or deciphering device) 役割도 한다. LFSR은 쉽게 구현될 수 있고 또 比較的 저렴하기 때문에 商用 스트림 暗號 시스템에 널리 쓰여진다. LFSR에서 生成되는 키 스트림이 적어도 擬似亂數(pseudo-random number) 系列이 되도록 하는 이유는 盜聽者가 이 暗號 알고리즘에 대해 統計的 공격을 어렵게 하기 위해서다.

Gilbert Vernam은 平文과 키 비트 系列(key bit sequence)을 2元合해서 暗號文으로 또 暗號文을 같은 키 비트 계열과 2元合해서 平文으로 還元하는 暗號化(encryption) 및 復號化(decryption)하는 技法을 처음으로 考案하였다. Vernam은 1918년경에 美國 電信電話公社(AT&T)에 근무시 32文字 符號에 基礎하여 電氣通信用 暗號器를 設計하였다. 個個의 文字는 mark("on"으로 標記)와 space("off"로 標記)의 組合으로 표현하고 各各의 文字는 GF(2) 上의 5비트로 變換하는 方法을 썼다. 平文은 通常 많은 알파벳으로 構成된다. 平文을 傳送하기 爲해 각 文字를 비트(bit)들의 集合으로 變換할 必要가 있다. 만약 英文 알파벳으로 된 平文을 傳送하려면 26개의 文字외에 Letters, Figures, Space, Null, Carrige return, Line feed 등 6개의 文字가 더 必要하다. 따라서 文字集合은 32개로 構成됨으로 각 文字는 $\log_2 32$비트 즉 한 文字당 5비트의 길이를 갖는다.

2次大戰후 暗號通信이 發達하게된 가장 중요한 要因은 컴퓨터, VLSI 기술, 數學的 難易性(rigor)의 發展 등이다. 스트림 暗號의 效能은 傳送路의 人爲的 또는 自然發生 雜音의 妨害가 甚한 경우 블럭(block) 暗號 보다 優越하다. 一例로 DES(Data Encryption Standard)와 같은 블럭 暗號 시스템과는 달리 傳送도중 暗號文에 發生한 單一誤謬(single error)는 復號(decryption)후 復元된 平文 該當 블럭내 單一誤謬만이 생기기 때문이다. 블럭 暗號와 스트림 暗號 兩 시스템의 差異點을 다음과 같이 要約할 수 있다.

(1) 블럭 暗號는 일정 길이의 데이타블럭으로 分轄되어 獨立的으로 暗號化 되지만 스트림 暗號는 暗號化와 復號化되는 過程에 키 비트 系列과 順次的인 비트별 2元 演算이 必要하다 ; (2) 블럭 暗號에서는 모든 暗號文(ciphertext) 비트가 相互 從屬關係를 갖는 平文과 키 비트의 複合函數(complex function)로 決定되나 스트림 暗號에서는 모든 暗號文 비트가 $Y=E_z(X)=(y_1, y_2, y_3, \cdots)$, 여기서 $y_i=x_i+z_i(\mathrm{mod}\ 2)$, $i=1, 2, \cdots$, 에 의해 決定된다 ; (3) 블럭 暗號는 初期條件이 必要치 않지만 스트림 暗號의 경우는 初期內容(initial contents or seed vector)이 반드시 必要하다 ; (4) 블럭 暗號 시스템보다 스트림 暗號 시스템의 우수한 長點의 하나는 誤謬傳播(error propagation)가 없다는 점이다 ; (5) 블럭 暗號 시스템은 相互從屬性(intersymbol dependence) 때문에 誤謬檢出(error detection)이나 認證(authentication)面에서 스트림 暗號 시스

템보다 有利하다.

2. 同期 스트림 暗號 시스템 (Synchronous Stream Ciphers)

同期 스트림 暗號 시스템에서 LFSR은 실제로 외부 키(external key)가 그 LFSR에 入力되어 無限한 키 비트 계열을 出力으로 生成시키는 키 符號器(key encoder) 역할을 하는 것을 指稱한다. 그림 1의 同期 스트림 暗號 시스템은 同一한 키 비트계열이 暗號化하려는 平文 또 復號化하려는 暗號文과 獨立的임을 나타내고 있다. 보통 우리는 暗號解讀術(cryptographic attack)에 버틸 수 있는 強한 暗號 시스템을 일컬어 無條件 完璧(unconditionally secure)한 시스템이라 부른다. 이 部類에 屬하는 가장 理想的인 暗號化 시스템을 one time

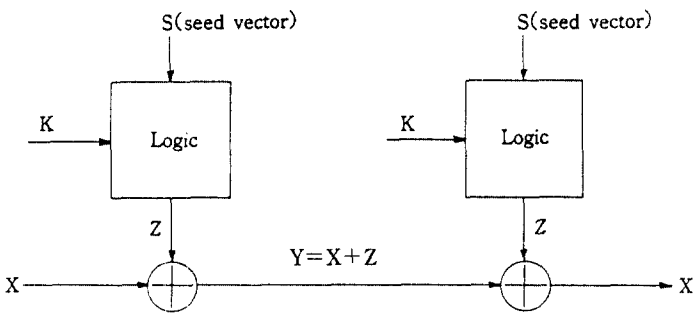

그림 1. 스트림 暗號 시스템의 一般形

표 1. 國際電信用 文字規格(ITA 2)

Character No.	Code bits	British teleprinter keyboard	Character No.	Code bits	British teleprinter keyboard
1	1 1 0 0 0	A −	17	1 1 1 0 1	Q 1
2	1 0 0 1 1	B ?	18	0 1 0 1 0	R 4
3	0 1 1 1 0	C :	19	1 0 1 0 0	S '
4	1 0 0 1 0	D WRU	20	0 0 0 0 1	T 5
5	1 0 0 0 0	E 3	21	1 1 1 0 0	U 7
6	1 0 1 1 0	F %	22	0 1 1 1 1	V =
7	0 1 0 1 1	G @	23	1 1 0 0 1	W 2
8	0 0 1 0 1	H £	24	1 0 1 1 1	X /
9	0 1 1 0 0	I 8	25	1 0 1 0 1	Y 6
10	1 1 0 1 0	J BELL	26	1 0 0 0 1	Z +
11	1 1 1 1 0	K (	27	0 0 0 1 0	CARRIGE RETURN
12	0 1 0 0 1	L)	28	0 1 0 0 0	LINE FEED
13	0 0 1 1 1	M .	29	1 1 1 1 1	LETTERS
14	0 0 1 1 0	N ,	30	1 1 0 1 1	FIGURES
15	0 0 0 1 1	O 9	31	0 0 1 0 0	SPACE
16	0 1 1 0 1	P 0	32	0 0 0 0 0	NULL

pad라고 부르는데 平文과 無限週期 키 스트림을 비트별로 2元合함으로써 暗號文을 生成하는 것이다. 그러나 無限週期의 키 스트림을 生成 또는 利用한다는 것이 事實上 實用이 不可能하다. 通常 平文과 키 계열 共히 GF(2)上의 數字集合을 사용한다. 예컨데 International Telegraphy Alphabet No.2 (ITA 2)는 元來 Teletypewriter 文字를 2進數로 變換하는 符號規格으로 한 文字當 5비트씩 符號化한 表를 指稱한다.

[例題 1] ITA 2 表를 利用해서 平文(message) X 즉 "START WITH 1990"를 2進數로 表現해 보자.

X=Letter, START, Space, Letter, WITH, Space, Figure, 1990
=11111 10100 00001 11000 01010 00001 00100 11111 11001
01100 00001 00101 00100 11011 11101 00011 00011 01101

키 비트 系列 Z를 다음과 같이 假定했을때

Z=R W B O Z V G M D H A X Q Y B I F J
=01010 11001 10011 00011 10001 01111 01011 00111 10010
00101 11000 11001 11101 10101 10011 01100 10110 11010

暗號文 Y는

Y=X+Z
=Y P D Figure Figure C V A G L W U W C C V Y X
=10101 01101 10010 11011 11011 01110 01111 11000 01011
01001 11001 11100 11001 01110 01110 01111 10101 10111

따라서 平文은 同一한 키 비트 系列 Z를 再使用함으로써 다음과 같이 復號된다.

X=Y+Z
=11111 10100 00001 11000 01010 00001 00100 11111 11001
01100 00001 00101 00100 11011 11101 00011 00011 01101

스트림 暗號 시스템의 完璧한 保安을 지키기 위해서는 키 비트 스트림 즉 수행 키(running key)는 어느 누구도 豫測할 수 없어야 한다. 非豫測性(unpredictability)을 위해서는 키 비트 스트림의 週期가 길어야 하며 수행 키가 豫測될 수 없기 위해서는 線形複雜度(linear complexity)가 커야 한다. 따라서 키 비트 계열의 非豫測性을 確立하기 위한 주된 觀點은 線形複雜度에 있다. 항상 계열의 주기는 線形複雜度가 커지면 길어지므로 線形複雜度가 크다는 것은 週期가 길다는 것을 의미한다. 물론 LFSR에 의해 배출하는 수행 키를 非線形結合法(nonlinear combination)으로 非週期 形態의 키 비트 스트림으로 改造할 수 있다. 非線形 結合에 관한 硏究는 Groth(1971), Geffe(1973), Siegenthaler (1985), Rueppel(1986) 등 많은 사람들에 의해 꾸준한 硏究가 수행되고 있다.

暗號文에서 디지트의 插入 및 消去(傳送중 디지트가 添加되거나 削除되는 것)는 同期의 실패를 초래하는 직접적인 原因이 된다. 受信側에서 同期를 回復하기 위해서는 모든 가능한 수단으로 送信側에서 키 스트림을 發生한 LFSR의 初期値 벡터를 再調整하거나 受信側에서 平文으로 復元하기 전에 誤謬訂正(error correction)을 해야 한다.

2.1 LFSR에 의한 키 符號化

그림 2 및 3에 나타난 바와 같이 g_1으로부터 g_m까지의 탭 係數(tap or feedback coefficients)와 m個의 플립플롭(flip-flop)으로된 LFSR를 생각해 보자. LFSR로부터 만들어지는 키 비트 스트림(符號化된 系列)이 統計的으로 無作爲性을 보인다 하더라도 元來의 外部 키와 그에 對應되는 키 비트 스트림 사이에는 線形性이 存在함은 무시할 수 없다. 그러므로 이러한 키 비트 스트림은 平文 保安을 유지하기 위한 暗號 키로서는 適合치 못하다. 실제로 알고 있는 平文과 暗號文의 2m 비트만으로도 탭 係數와 LFSR의 初期內容(initial contents or initial seed vector)을 決定할 수 있기 때문에 키 비트 스트림을 探知하기란 그리 어려운 문제가 아니다. 모든 正數 m에 대해서 有限體 GF(2^m)上의 原始多項式 p(x)가 주어져 있음으로 週期가 2^m-1인 m段 LFSR을 構成할 수 있다. 結局, 어떤 값 m에 대해서도 탭 係數는 쉽게 구해지므로 LFSR로부터 符號化된 키 스트림은 거의 不安全하고 아주 작은 m에 대해서는 미니 컴퓨터로 수 초만에 깨어질 수 있다.

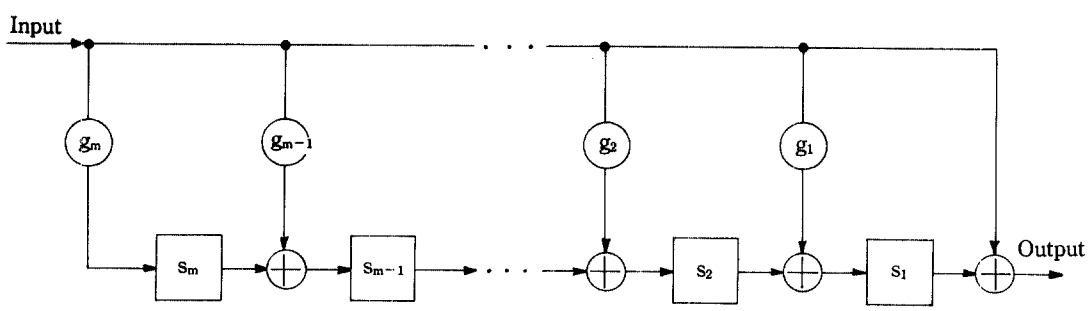

그림 2. 탭 係數가 g_i, $1 \leq i \leq m$, 인 m段 線形歸還置換 레지스터(LFSR)

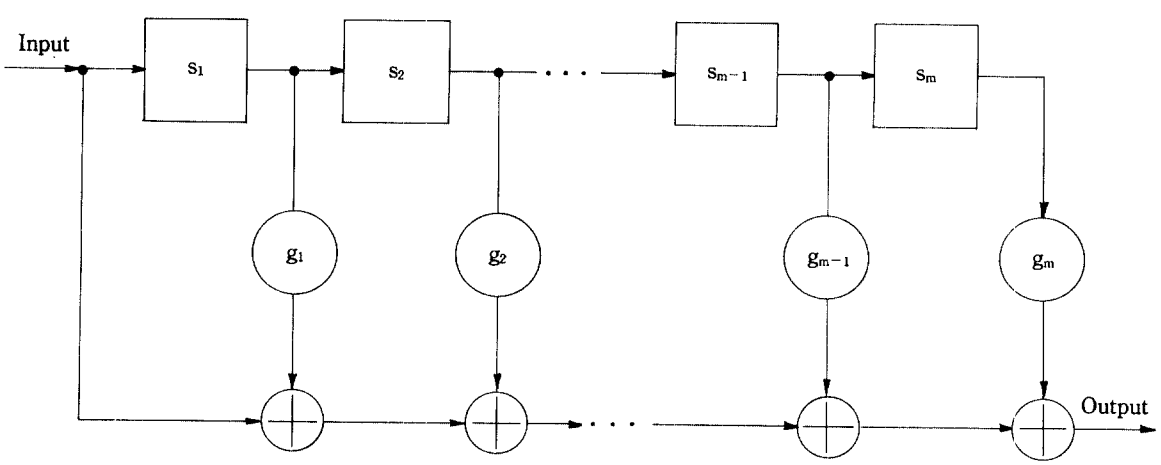

그림 3. 탭 係數가 g_i, $1 \leq i \leq m$, 인 變形된 m段 LFSR

그림 2와 3에서 보듯이 LFSR의 初期狀態는 s_1, s_2,…, s_m, 여기서 $s_i \in GF(2)$로 表示되고, 탭 係數는 g_1, g_2,…, g_m으로 g_i가 1 또는 0에 따라 접속 또는 단락된다. 이런 種類의 LFSR 配列을 置換 레지스터 符號化 方式이라 하며, 行列 T로 表現될 수 있다. 外部 키 벡터를 $K=(k_0, k_1,\cdots, k_{m-1})$로 表示하면 LFSR를 통해 符號化된 키 스트림 Z는

$$Z=T\cdot K \qquad (1)$$

가 된다. 여기서 T를 遷移行列(transfer matrix)이라 한다.

LFSR 符號化는 이미 설명한 바와 같이 線形이므로 式(1)은 解釋的으로 遲延素子 D에 의해

$$Z(D)=T(D)K(D) \qquad (2)$$

로 表現된다. 여기서

$$K(D)=\sum_{i=0}^{\infty} k_i D^i \qquad (入力\ 키)$$

$$Z(D)=\sum_{j=1}^{m} [K(D)D^j]g_j \quad (符號化된\ 出力\ 키)$$

로 表現될 수 있다. $K=(k_0, k_1,\cdots, k_{m-1})$를 元來의 入力 키, $Z=(z_0, z_1,\cdots, z_{m-1})$를 對應되는 符號化 키 블럭이라 하자. LFSR 키 符號器는 탭 係數 g_1, g_2,…, g_m와 初期內容(initial seed vector) $S=(s_1, s_2,\cdots, s_m)$가 貯藏된 m段 레지스터로 構成된다. LFSR의 탭 係數와 初期狀態를 빠르게 알아내기 위해서는 2m 비트의 키 系列과 이에 對應되는 符號化 키 系列을 알기만 하면 된다는 것은 흥미로운 일이다. 일단 이런 條件을 알 수만 있다면 m段 LFSR로 만들어진 키 비트 스트림은 깨어질 수 있고, 따라서 完全한 元來의 키는 復號될 수 있다. 끝으로 아래 例題를 통하여 充分한 설명을 記述한다.

[例題 2] 그림 4에서 보인 4段 LFSR 키 符號器를 생각하자. LFSR의 탭 係數는 g_1, g_2, g_3, g_4로 設定하고 키 符號器의 初期內容은 s_1, s_2, s_3, s_4로 하자. 따라서 2m=8이므로 入力 키 系列은 $K=(k_{70}, k_1,\cdots, k_{77})$이며 符號化된 出力 키 系列은 $Z=(z_0, z_1,\cdots, z_7)$이다. 法(modulo) 2 演算에서는 $k_j+s_j=z_{7j}$가 $z_{7j}+k_j=s_j$로 表現될 수 있음으로 4段 키 符號器를 解釋하면 다음과 같은 式이 만들어진다.

$$\begin{aligned}
z_0+k_0&=s_1\\
z_1+k_1&=g_1k_0+s_2\\
z_2+k_2&=g_1k_1+g_2k_0+s_3\\
z_3+k_3&=g_1k_2+g_2k_1+g_3k_0+s_4\\
z_4+k_4&=g_1k_3+g_2k_{72}+g_3k_1+g_4k_0\\
z_5+k_5&=g_1k_4+g_2k_3+g_3k_2+g_4k_1\\
z_6+k_6&=g_1k_5+g_2k_4+g_3k_3+g_4k_2\\
z_7+k_7&=g_1k_6+g_2k_5+g_3k_4+g_{74}k_3
\end{aligned} \qquad (3)$$

式(3)은 다시 아래와 같은 行列形態로도 表現할 수 있다.

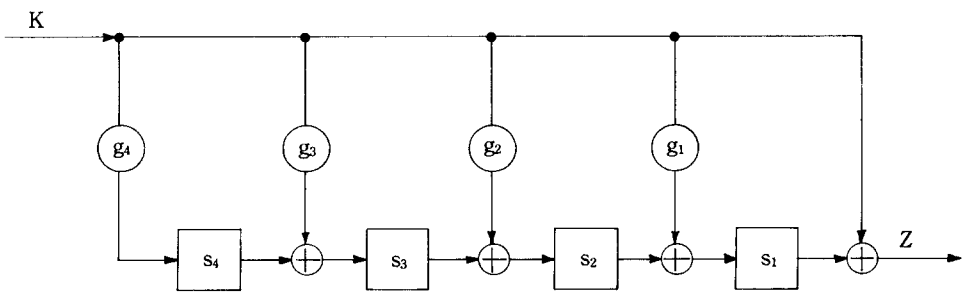

그림 4. 4段 LFSR 키 符號器

$$\begin{bmatrix} z_0+k_0 \\ z_1+k_1 \\ z_2+k_2 \\ z_3+k_3 \\ z_4+k_4 \\ z_5+k_5 \\ z_6+k_6 \\ z_7+k_7 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ k_0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ k_1 & k_0 & 0 & 0 & 0 & 0 & 1 & 0 \\ k_2 & k_1 & k_0 & 0 & 0 & 0 & 0 & 1 \\ k_3 & k_2 & k_1 & k_0 & 0 & 0 & 0 & 0 \\ k_4 & k_3 & k_2 & k_1 & 0 & 0 & 0 & 0 \\ k_5 & k_4 & k_3 & k_2 & 0 & 0 & 0 & 0 \\ k_6 & k_5 & k_4 & k_3 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \\ s_1 \\ s_2 \\ s_3 \\ s_4 \end{bmatrix} \quad (4)$$

레지스터 段의 모든 初期內容이 零이라고 假定하면, 즉 모든 레지스터 段이 初期에 리셋트(reset)되었다고 假定하면, 이는 $s_i=0 \leq i \leq 4$임을 나타낸다. 이 경우 式(4)는 다음과 같이 簡略하게 된다.

$$\begin{bmatrix} z_4+k_4 \\ z_5+k_5 \\ z_6+k_6 \\ z_7+k_7 \end{bmatrix} = \begin{bmatrix} k_3 & k_2 & k_1 & k_0 \\ k_4 & k_3 & k_2 & k_1 \\ k_5 & k_4 & k_3 & k_2 \\ k_6 & k_5 & k_4 & k_3 \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{bmatrix} \quad (5)$$

한 例로써 그림 5에 보인 4段 키 符號器(LFSR)를 살펴보자. 入力 키 K와 符號化된 出力 키 Z 사이에 線形關係가 있음을 보이고자 한다. 탭 係數가 $g_1=g_2=0$이고 $g_3=g_4=1$이기 때문에 符號化 回路의 遷移函數는 $T(D)=D^4+D^3+1$로 誘導된다. 入力 키 벡터가 K=(11010001)이라고 假定하면 그의 多項式 表現은 $K(D)=1+D+D^3+D^7$이다. 그림 5의 符號器는 法 2(modulo 2) 多項式 곱셈기이므로 多項式 形態로 出力 키를 表現하면

$$\begin{aligned} Z(D) &= T(D)K(D) \\ &= (D^4+D^3+1)(1+D+D^3+D^7) \\ &= 1+D+D^5+D^6+D^{10}+D^{11} \end{aligned}$$

이 된다. 여기서 D^i, $i>7$의 項을 無視함으로써 $Z(D)=1+D+D^5+D^6$ 또는 Z=(11000110)가 된다. Z와 K를 비트별로 더하면

$$Z+K=(00010111)$$

를 얻을 수 있고 이것으로부터

$$\begin{aligned} z_4+k_4&=0 \\ z_5+k_5&=1 \\ z_6+k_6&=1 \\ z_7+k_7&=1 \end{aligned}$$

임을 알 수 있다. 이 값을 式(5)에 代入하면 歸還係數 g_1에서 g_4까지를 다음과 같이 決定할 수 있다.

$$\begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{bmatrix}$$

이 式에서

$$\begin{aligned} g_1+g_3+g_4&=0 \\ g_2+g_4&=1 \\ g_3&=1 \\ g_4&=1 \end{aligned}$$

를 얻을 수 있으며 이 聯立方程式으로부터 g_i, $1 \leq i \leq 4$에 대한 解를 구하면 $g_1=0$, $g_2=0$, $g_3=1$, 그리고 $g_4=1$이 된다.

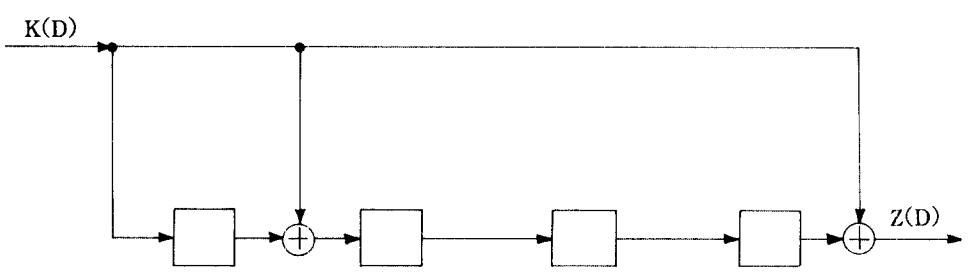

그림 5. 遷移函數가 $T(D)=1+D^3+D^4$인 4段 LFSR 키 符號器

그런데 式(5)은 入力 키 K가 주어졌을때 탭 係數 g_i, $1\leq i\leq 4$를 決定할 수 있는 唯一한 關係式을 表現한 것은 아니다. 그 外에 3가지 聯立方程式群을 풀어도 똑같은 탭 係數 g_i을 決定할 수 있다. 다시말해서 初期內容이 零(即 $s_i=0$, $1\leq i\leq 4$)일때 式(4) 即 式(3)으로부터 4個의 方程式을 뽑아내는 組合은 모두 네가지의 경우가 있다. 그 中 하나가 式(5)이고 나머지 3개의 聯立方程式群은 다음과 같다.

$$\begin{bmatrix} z_1+k_1 \\ z_2+k_2 \\ z_3+k_3 \\ z_4+k_4 \end{bmatrix} = \begin{bmatrix} k_0 & 0 & 0 & 0 \\ k_1 & k_0 & 0 & 0 \\ k_2 & k_1 & k_0 & 0 \\ k_3 & k_2 & k_1 & k_0 \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{bmatrix}, \quad \begin{bmatrix} z_2+k_2 \\ z_3+k_3 \\ z_4+k_4 \\ z_5+k_5 \end{bmatrix} = \begin{bmatrix} k_1 & k_0 & 0 & 0 \\ k_2 & k_1 & k_0 & 0 \\ k_3 & k_2 & k_1 & k_0 \\ k_4 & k_3 & k_2 & k_1 \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{bmatrix}, \quad \begin{bmatrix} z_3+k_3 \\ z_4+k_4 \\ z_5+k_5 \\ z_6+k_6 \end{bmatrix} = \begin{bmatrix} k_2 & k_1 & k_0 & 0 \\ k_3 & k_2 & k_1 & k_0 \\ k_4 & k_3 & k_2 & k_1 \\ k_5 & k_4 & k_3 & k_2 \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{bmatrix}$$

위 3個의 聯立方程式群중 어느 하나를 選擇해도 入力 키 K=(11010001)와 出力 키 Z=(11000110)를 使用한다면 탭 係數 g_i, $1\leq i\leq 4$는 $g_1=g_2=0$, $g_3=g_4=1$로 計算됨을 알 수 있다.

앞에서 說明한 절차는 단지 그림 2와 같은 回路에만 適用된다. 그림 3의 變形된 回路는 다음 例題를 通하여 논의될 수 있다. 그러나 그림 4와 그림 6의 곱셈기가 똑같다는 것을 認知하기는 그리 어렵지 않다. 實際로 그림 2의 回路構成이 그림 3의 構成보다 좀 더 빠른 速度로 作動할 수 있다. 왜냐하면 그림 2는 歸還經路에서 傳播遲延이 좀 더 적기 때문이다.

[例題 3] 그림 6에 圖示된 4段 키 符號器를 變形한 構造를 살펴보자. 앞에서처럼 LFSR 키 符號器의 初期內容을 GF(2)上의 元素인 s_1, s_2, s_3, s_4로 表示하여 이 4段 키 符號器를 解釋하면 다음 式들이 얻어진다.

$$\begin{aligned} z_0+k_0 &= g_1s_1+g_2s_2+g_3s_3+g_4s_4 \\ z_1+k_1 &= g_1k_0+g_2s_1+g_3s_2+g_4s_3 \\ z_2+k_2 &= g_1k_1+g_2k_0+g_3s_1+g_4s_2 \\ z_3+k_3 &= g_1k_2+g_2k_1+g_3k_0+g_4s_1 \\ z_4+k_4 &= g_1k_3+g_2k_2+g_3k_1+g_4k_0 \\ z_5+k_5 &= g_1k_4+g_2k_3+g_3k_2+g_4k_1 \\ z_6+k_6 &= g_1k_5+g_2k_4+g_3k_3+g_4k_2 \\ z_7+k_7 &= g_1k_6+g_2k_5+g_3k_4+g_4k_3 \end{aligned} \qquad (6)$$

이 線形方程式群을 行列로 表現하면 아래와 같다.

$$\begin{bmatrix} z_0+k_0 \\ z_1+k_1 \\ z_2+k_2 \\ z_3+k_3 \\ z_4+k_4 \\ z_5+k_5 \\ z_6+k_6 \\ z_7+k_7 \end{bmatrix} = \begin{bmatrix} s_1 & s_2 & s_3 & s_4 \\ k_0 & s_1 & s_2 & s_3 \\ k_1 & k_0 & s_1 & s_2 \\ k_2 & k_1 & k_0 & s_1 \\ k_3 & k_2 & k_1 & k_0 \\ k_4 & k_3 & k_2 & k_1 \\ k_5 & k_4 & k_3 & k_2 \\ k_6 & k_5 & k_4 & k_3 \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{bmatrix} \qquad (7)$$

萬若 모든 레지스터段이 初期에 리렛트 되었다면 初期値 s_i, $1\leq i\leq 4$는 모두 零이 되고 結果的으로 式(7)은 아래와 같이 간단한 行列表現으로 式(5)와 同日하게 된다.

$$\begin{bmatrix} z_4+k_4 \\ z_5+k_5 \\ z_6+k_6 \\ z_7+k_7 \end{bmatrix} = \begin{bmatrix} k_3 & k_2 & k_1 & k_0 \\ k_4 & k_3 & k_2 & k_1 \\ k_5 & k_4 & k_3 & k_2 \\ k_6 & k_5 & k_4 & k_3 \end{bmatrix} \begin{bmatrix} g_1 \\ g_2 \\ g_3 \\ g_4 \end{bmatrix} \qquad (8)$$

歸還係數 $g_1=g_2=0$과 $g_3=g_4=1$로 그림 6은 그림 7과 같이 그려질 수 있다.

그림 7을 살펴보면 $z(t)=k(t)[(D^3+1)+D^4]$이므로 키 符號器의 遷移函數는 $T(D)=D^4+D^3+1$로 誘導되고 例題 2에서 얻은 式과 同一하다. 앞 例題와 같은 假定下에 $K(D)=1+D+D^3+D^7$라 하면 出力函數가 $Z(D)=T(D)K(D)=1+D+D^5+D^6$이 된다. 따라서 Z와 K를 結合하여 式(8)을 利用하면 탭(歸還) 係數 g_i, $1\leq i\leq 4$,는 앞에서 구한것과 같은 $g_1=g_2=0$과 $g_3=g_4=1$로 決定된다. 이 두 例題를 통하여 단지 알고 있는 키 文(keytext) 2m=8 비

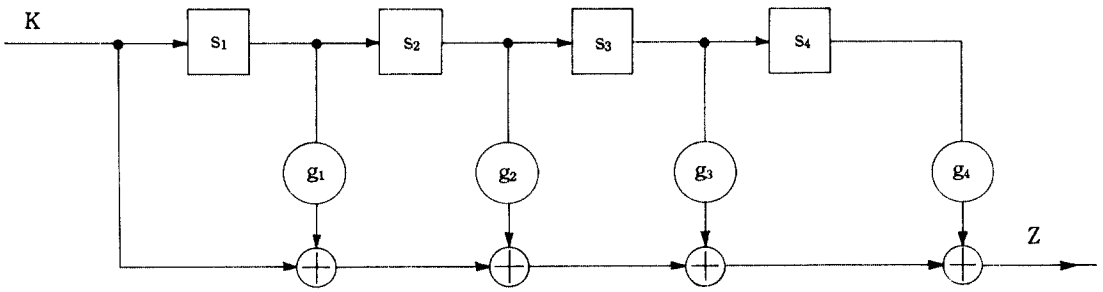

그림 6. 變形된 4段 LFSR 키 符號器

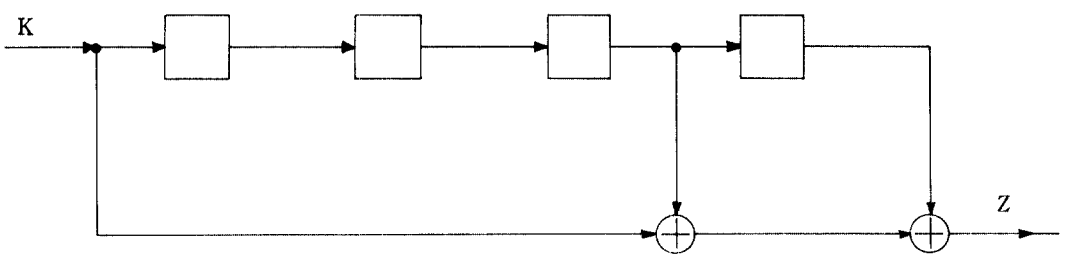

그림 7. 탭(歸還) 係數가 $g_1=g_2=0$, $g_3=g_4=1$인 4段 키 符號器

트와 그에 對應되는 符號化된 키 스트림만으로 LFSR 符號器의 탭 係數가 決定됨을 알 수 있다. 이 시점에서 그림 2의 LFSR 키 符號器를 言及하는 것은 의미있는 일이다. 만약 入力 키 $K=(k_0, k_1, \cdots, k_{2m-1}, \cdots)$가 平文 $X=(x_1, x_2, \cdots, x_{2m}, \cdots)$으로 代置되고 符號化된 出力 키 $Z=(z_0, z_1, \cdots, k_{2m-1}, \cdots)$가 暗號文 $Y=(y_1, y_2, \cdots, y_{2m}, \cdots)$로 代置되며 또 LFSR의 歸還係數가 설정키(key setting)로 代置되는 경우 그림 2의 키 符號器는 完全한 暗號器가 된다. 그러나 그림 2 또는 그림 3의 構成은 脆弱한 暗號 시스템이다. 왜냐하면 알고있는 平文 2m 비트와 이에 對應되는 暗號文 2m 비트만으로 설정키를 決定할 수 있고 결국 盜聽者(opponent)는 推定된 키스트림으로 暗號를 解讀하게 되어 完全한 平文을 얻을 수 있기 때문이다. 따라서 이런 種類의 同期式 스트림 暗號 시스템에는 弱點이 있고 쉽게 깨어질 수 있다. (다음 號에 계속)

參 考 文 獻

1. Golomb, S.W. : Shift Register Sequence, Honlden-Day, San Francisco, CA, 1967.

2. Hurd, W.J. : "Efficient Generation of Statistically Good Pseudonoise by Linearly Interconnected Shift Registers," IEEE Trans. Computers, vol. C-23, pp. 146-152, Feb. 1974.

3. Pane, W.H. and T.G. Lewis : "Generalized Feedback Shift Register Pseudorandom Number Algorithms," J. Assoc. Comput. Math., vol. 20, pp. 456-468, July, 1973.

4. Rhee, M.Y. : Error Correcting Coding Theory, McGraw-Hill, New York, pp. 98-103, 1989.

5. Siegenthaler, T. : "Decrypting a Class of Stream Ciphers Using Ciphertext Only," IEEE Trans. Computers, vol. C-34, no. 1, January, 1985.

(著者紹介는 p. 23 참조)