通信情報保護學會誌
第1卷, 第3號, 1991. 12

招請特輯

스트림(逐字) 暗號시스템에 關한 硏究
Study on the Stream Cipher Systems
(3)

李 晩 榮*

第2號에 이어 本稿에서는 同期逐字暗號시스템(synchronous stream ciphers)에서의 誤謬傳播(error propagation) 特性과 이에 대한 對策으로 誤謬訂正方法(error control method)에 대해 記述한다. 特히, 自動키 暗號시스템, 暗號文 歸還 暗號시스템, 平文 歸還 暗號시스템으로 구분하여 內部 誤謬制御(internal error control) 및 外部誤謬制御(external error control)技法의 實現例를 詳細히 다루고자 한다. 第3號를 끝으로 本連載가 讀者 諸位에게 조금이나마 도움이 되길 바라며 招請特輯을 마친다.

目 次

* 종신회원, 漢陽大學校 名譽教授, 本 學會 會長

4. 스트림 暗號시스템에서의 誤謬傳播特性

블럭 暗號(block cipher)는 平文을 일정한 크기의 블럭으로 나눈 뒤, 각 블럭을 동일한 키를 가지고 독립적으로 暗號化하고 復號한다. 暗號文의 각 비트는 심볼간 연관성에 의하여 平文과 키의 모든 비트에 영향을 받는 複合 函數이므로 平文은 暗號文에 직접적으로 나타나지 않는다. 블럭 暗號는 본래 스트림 暗號 보다 빠르지 못하기 때문에 고속을 요하는 경우에는 스트림 暗號의 응용이 적절할 것이다. 그러나 스트림 暗號는 DES와 같은 블럭 暗號보다 더 쉽게 暗號해독자의 공격을 받을 수 있다는 단점을 갖는다.

블럭 暗號에서 한 暗號文 블럭에 생긴 誤謬는 다른 블럭에 영향을 끼치지 않는다. 실제로 블럭 暗號는 블럭의 삽입이나 삭제에 대해 취약하지 않은데, 그 이유는 어떤 暗號文 블럭에 가해진 변조가 주변 블럭에는 영향을 미치지 않기 때문이다. 認證(authentication)은 공격자가 채널에 잘못된 데이타를 집어 넣거나 暗號化된 데이타를 변조하는 일이 없도록 하는 것을 의미한다. 이런 점에서, 블럭 暗號에서의 誤謬傳播는 認證에 사용될 수 있다.

스트림 暗號는 暗號器의 초기 상태에 의존하는 방식으로 平文을 暗號化 한다. 실제로, 동일한 平文을 暗號化하더라도 출력 暗號文의 형태는 다르게 된다. 따라서, 한 블럭이 傳送誤謬에 의해 변조되면 정확히 한 블럭이 誤謬가 있는 블럭으로 復號된다. 그러나, 自己 同期 스트림 暗號는 각 키 비트가 전체 暗號文이나 平文과 함수적으로 연관되어 있으므로 誤謬傳播가 불가피하다. 특히, 暗號文 自動키 시스템에서는 한개의 誤謬 또는 한개의 소실된 暗號文 성분이 復號된 平文에서 일정량의 誤謬를 발생시키며, 일정 시간후에는 그 誤謬의 영향으로 부터 벗어나는 특성을 갖는다. 그러나, 平文 自動키 시스템에서는 誤謬가 있는 하나의 暗號文 블럭으로 復號된 平文에서는 일정치 않은 誤謬傳播가 이어지게 된다. 키 自動키 시스템에서는 전송로상의 잡음으로 인해 변조된 暗號文 각 블럭은 단지 한 개의 잘못된 블럭만을 생성시킨다. 誤謬傳播에 대한 상세한 특성을 다음의 例題들로 설명한다.

[例題 19] 例題 12와 15에서 다룬 暗號文 歸還自己 同期 暗號를 다시 고려해보자. 平文 X=(11101001)에 대한 暗號文은 Y=(00100011)이다. 탭 계수가 T=(g_1, g_2, g_3, g_4)=(0101)이고 초기상태가 S=(s_1, s_2, s_3, s_4)=(0011)인 그림 17을 이용하자. 그림 17의 暗號文에 하나의 誤謬가 생겨 Y=(01100011)이 입력되었다고 가정하자. 이와 같이 誤謬가 포함된 暗號文 Y를 復號하여 만든 平文 X에서 誤謬가 傳播되는 것에 주목할 수 있으며, 이를 분석하면 다음과 같다.

$x_0 = y_0 + s_2 + s_4 = 0 + 0 + 1 = 1$ (誤謬)

$x_1 = y_1 + s_1 + s_3 = 1 + 0 + 1 = 0$

$x_2 = y_2 + y_0 + s_2 = 1 + 0 + 0 = 1$ (誤謬)

$x_3 = y_3 + y_1 + s_1 = 0 + 1 + 0 = 1$

$x_4 = y_4 + y_2 + y_0 = 0 + 1 + 0 = 1$ (誤謬)

$x_5 = y_5 + y_3 + y_1 = 0 + 0 + 1 = 1$

$x_6 = y_6 + y_4 + y_2 = 1 + 0 + 1 = 0$

$x_7 = y_7 + y_5 + y_3 = 1 + 0 + 0 = 1$

表 7. 暗號文 歸還法에서 Y에 발생된 單一誤謬로 인한 X에서의 誤謬傳播

入力 暗號文 Y	單一 誤謬가 있는 暗號文 Y	復號된 平文에서의 誤謬 分布 X	올바른 平文 X
00100011	*10100011	***01000001	11101001
00100011	*01100011	***10111101	11101001
00100011	*00000011	***11000011	11101001
00100011	*00110011	***11111100	11101001
00100011	*00101011	**11100011	11101001
00100011	*00100111	**11101100	11101001
00100011	001000001*	*11101011	11101001
00100011	001000010*	*11101000	11101001

* : 誤謬 위치 표시

그러므로 誤謬가 있는 平文 X=(10111101)로 復號된다. 원래의 平文 X=(11101001)와 비교해 보면 3개의 誤謬가 있음을 알 수 있고 이는 Y에서 발생한 하나의 誤謬에 의한것이다.

誤謬가 포함된 暗號文 입력 Y에 의해 復號된 平文 X로 부터 誤謬가 傳播되는 형태를 보면 暗號文 歸還法에서의 誤謬傳播는 表 7과 같다.

[例題 20] 例題 16과 18에서 논의된 平文 歸還 自己 同期 暗號의 誤謬傳播 문제에 대해 생각하자. 그림 19를 보면, LFSR에서의 탭 계수는 T=(0101)이고, 초기 상태는 S=(0011)이다. 그림 19에서 誤謬가 포함된 暗號文 Y를 입력하여 復號된 平文의 誤謬傳播 형태를 살펴보자. 그림 18을 보면 平文 X=(11101001)에 의한 暗號文은 Y=(00011101)이다. Y의 두번째 비트에 誤謬가 발생하여 Y=(01011101)이라고 가정하면, 平文 비트는 다음과 같이 復號된다.

$x_0=y_0+s_2+s_4=0+0+1=1$
$x_1=y_1+s_1+s_3=1+0+1=0$ (誤謬)
$x_2=y_2+x_0+s_2=0+1+0=1$
$x_3=y_3+x_1+s_1=1+0+0=1$ (誤謬)
$x_4=y_4+x_2+x_0=1+1+1=1$
$x_5=y_5+x_3+x_1=1+1+0=0$
$x_6=y_6+x_4+x_2=0+1+1=0$
$x_7=y_7+x_5+x_3=1+0+1=0$ (誤謬)

그러므로 誤謬가 포함된 平文系列은 2, 4, 8번째에 3개의 誤謬가 생긴 X=(10111000)으로 復號된다.

單一 誤謬가 발생한 Y에 의한 X에서의 誤謬傳播는 表 8과 같다.

이제, (1) 暗號文 自動키 시스템에서는 誤謬가 포함된 暗號文 블럭과 대응되는 平文 블럭 이외에는 誤謬傳播가 없다는 것과 (2) 平文 自動키 시스템에서는 暗號文 블럭에 발생한 誤謬 비트들이 復號된 平文에서 誤謬傳播를 일으킨다는 것을 설명한다.

表 8. 平文 歸還法에서 Y에 발생된 單一 誤謬에 대응되는 X에서의 誤謬傳播

入力 暗號文 Y	單一 誤謬가 있는 暗號文 Y	復號된 平文에서의 誤謬分布 X	올바른 平文 X
00011101	*10011101	***01001011	11101001
00011101	0*011101	1***10111000	11101001
00011101	00*11101	11**000001	11101001
00011101	000*1101	111*1101	11101001
00011101	0001*101	1110**11	11101001
00011101	00011*01	11101**0	11101001
00011101	000111*1	11101*11	11101001
00011101	0001110*	11101000*	11101001

*：誤謬 위치 표시

[例題 21] 平文 입력이 X=(11101001 10101110 11101011 10111011 10110111 00111010)인 平文 自動키 暗號시스템을 고려해 보자. X를 暗號化하면 暗號文 Y=(00011101 01011111 10111111 11101110 11100001 10000111)를 얻을 수 있다. 誤謬가 없는 暗號文 Y를 復號하면 올바른 平文 X가 복구될 것이다. 그러나 Y의 8비트로 이루어진 심볼 중 임의의 한 심볼에 한 비트의 誤謬가 생기면, 復號된 X에서는 誤謬가 발생한 해당 심볼로부터 시작하여 表 9와 같이 일정하지 않게 誤謬가 傳播된다.

[例題 22] 平文 X=(11101001 10101110 11101011 10111011 10110111 00111010)가 입력되는 暗號文 自動키 暗號器에 대해 생각하자. 暗號文 歸還으로 생성되는 키 數例로 X를 暗號化하면 暗號文은 Y=(00100011 01001000 01110000 10010111 00000110 11101111)가 된다. Y의 임의의 심볼 블럭에 誤謬가 한 비트 생겼다고 가정하자. 그러면 復號된 平文 X는 誤謬를 포함하게 될 것이다. 그러나 誤謬傳播는 表 10에서와 같이 8비트 심볼블럭 하나에만 국한된다.

表 9. 平文 歸還法에서의 平文 X에 대한 誤謬傳播

Y의 첫번째 심볼 블럭 (00011101)에 생긴 誤謬	X에서의 誤謬傳播						48비트 중 誤謬 갯수
*10011101	01001011	00100100	11000011	00011001	00111101	00010010	15
*01011101	10111000	11101011	11111111	11101010	11110010	00101110	16
*00111101	11000001	00001100	01100001	10010011	00010101	10110000	16
*00001101	11111101	11111111	10101110	10101111	11100110	01111111	16
*00010101	11100011	10000110	01001001	00110001	10011111	10011000	15
*00011001	11101100	10111010	10111010	11111110	10100011	01101011	15
*00011111	11101011	00100100	11000011	00011001	00111101	00010010	14
*00011100	11101000	11101011	11111111	11101010	11110010	00101110	14

* : 誤謬 위치 표시

表 10. 暗號文 歸還法에서의 平文 X에 대한 誤謬傳播

Y의 첫번째 심볼 블럭 (00011101)에 생긴 誤謬	X에서의 誤謬傳播					
*10100011	01000001	10101110	11101011	10111011	10110111	00111010
*01100011	10111101	10101110	11101011	10111011	10110111	00111010
*00000011	11000011	10101110	11101011	10111011	10110111	00111010
*00110011	11111100	10101110	11101011	10111011	10110111	00111010
*00101011	11100011	10101110	11101011	10111011	10110111	00111010
*00100111	11101100	10101110	11101011	10111011	10110111	00111010
*00100001	11101011	10101110	11101011	10111011	10110111	00111010
*00100010	11101000	10101110	11101011	10111011	10110111	00111010

* : 誤謬 위치 표시

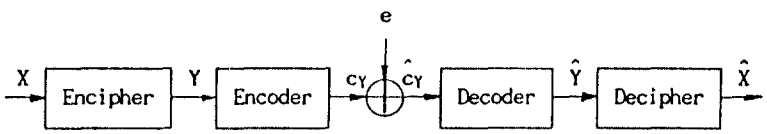
결론적으로 暗號文 歸還 自動키 暗號시스템은 誤謬傳播가 없다는 장점을 갖고 있다. 하나의 심볼 블럭 Y에 영향을 주는 傳送誤謬는 復號된 平文에서 誤謬 위치 다음의 심볼 블럭에는 영향이 없다. 반면에 平文 歸還 自動키 暗號시스템은 復號된 平文 X에서 일정치 않은 誤謬傳播가 일어나는 단점이 있다.

5. 스트림 暗號시스템에서의 誤謬訂正

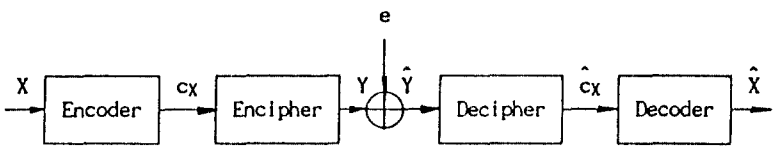
誤謬訂正符號(error correcting codes)는 전송로상에서 발생하는 誤謬(error)를 訂正하기 위한 기법으로 디지틀 통신시스템 및 컴퓨터 주기억장치 등에서 널리 사용되고 있다. 그러나, 暗號 관련시

스템에서도 誤謬訂正技術의 필요성이 인정되고 강조되고 있으나 이를 위한 연구논문 발표는 그리 많지 않다. 따라서, 본 장에서는 이와 관련하여 誤謬訂正符號를 스트림 暗號시스템에 도입하므로서 暗號文 전송중 전송로상에서 발생하는 誤謬를 制御하기 위한 기법에 대해서 소개한다.

暗號시스템에 誤謬訂正符號를 함께 사용하여 구현하고자 할 때, 그림 20과 같이 2종류의 시스템으로 구성할 수 있다. 즉, 첫번째 형태로 平文(plaintext)을 먼저 暗號化(enciphering)한 다음 符號化(encoding)시키는 것으로, 전송후 수신측에서는 誤謬訂正을 수행한 다음 원래의 平文으로 復號化(deciphering)시키는 內部誤謬制御(internal error control) 技法(그림 20(a))이 있으며, 두번째 형태로 平文을 符號化한 다음 暗號化시켜 전송하고, 수신측에서는 우선 暗號文을 復號化한 다음 誤謬訂正을 수행하는 外部誤謬制御(external error control) 技法(그림 20(b))이 있다.



(a) 內部誤謬制御



(b) 外部誤謬制御

그림 20 誤謬制御를 도입한 暗號시스템

본 장에서는, 우선 誤謬訂正符號로 訂正能力이 강력한 Reed-Solomon(이하, RS)符號에 대해 소개하고, 그 제원과 가장 일반적이며 효율적인 復號技法인 Peterson-Gorenstein-Zieler(PGZ) 알고리듬에 대해 간략히 기술한다. 또한, 이를 內部 및 外部誤謬制御로 구분하여 同期(synchronous) 및 自己同期(self-synchronizing) 스트림 暗號시스템에 도입하여 각각의 誤謬傳播特性 및 誤謬訂正 수행에 따른 결과를 비교·분석한다.

5. 1 RS符號의 復號를 위한 PGZ알고리듬

非 2元 BCH符號의 한 부류에 속하는 RS符號에 관한 符號 알고리듬은 치환레지스터를 이용한 復號, Euclid 알고리듬을 이용한 復號, 有限體 Fourier 變換을 이용한 復號 등 다각적으로 연구되어 왔다. 본 절에서는 Peterson이 2元 BCH符號의 復號를 위해 창안한 기법을 Gorenstein과 Zierler가 개선시킨 알고리듬에 대해 개략적인 기술을 한다. 이 알고리듬은 t차원 정방행렬을 이용한 것으로 誤謬訂正能力 t가 비교적 적을 경우 효과적이다.

RS符號의 復號는 受信多項式 r(x)에 대한 誤症(syndrome) s의 계산으로 부터 시작된다. 실제로, v, $0 \le v \le t$개의 誤謬가 l_1, l_2, …, l_v의 위치에서 발생하였다면 誤謬多項式은

$$e(x) = el_1 X^{l1} + el_2 X^{l2} + \cdots + el_v x^{lv} \cdots\cdots (53)$$

로 표현된다. 여기서, el_i, $l<i<v$는 $GF(2^m)$의 원소이다.

誤謬値(error value)를 $Y_i = el_i$, 誤謬位置番號

(error locator number)를 $Z_i=\alpha^{n_i}$라 치환하면 誤症要素 s_k, $1\leq k\leq 2t$는 식(53)을 이용하여

$$s_\lambda=\sum_{k=1}^{v} Y_k Z_k^{\lambda}, \quad 1<k<2t \quad \cdots\cdots (54)$$

즉,

$$\begin{aligned} s_1&=Y_1Z_1+Y_2Z_2+\cdots+Y_vZ_v\\ s_2&=Y_1Z_1^2+Y_2Z_2^2+\cdots+Y_vZ_v^2\\ &\cdots\cdots (55)\\ s_{2t}&=Y_1Z_1^{2t}+Y_2Z_2^{2t}+\cdots+Y_vZ_v^{2t} \end{aligned}$$

와 같은 2t개의 방정식으로 표현가능하다.

또한, 誤謬位置多項式(error locator polynomial) $\sigma(x)$를

$$\begin{aligned} \sigma(x)&=\prod_{i=1}^{v}(1+Z_ix)\\ &=1+\sigma_1x+\sigma_2x^2+\cdots+\sigma_vx^v \quad \cdots\cdots (56) \end{aligned}$$

라 정의할때, 식(55)와 (56)으로 부터

$$\begin{aligned} &s_{j+v}+\sigma_1s_{j+v-1}+\sigma_2s_{j+v-2}+\cdots+\sigma_vs_j\\ &=0, \quad 1\leq j\leq v \quad \cdots\cdots (57) \end{aligned}$$

과 같은 Newton의 항등식을 얻을 수 있고, 식 (57)을 행렬로 표시하면

$$\begin{bmatrix} \sigma_v\\ \sigma_{v-1}\\ \cdot\\ \cdot\\ \cdot\\ \sigma_1 \end{bmatrix} = M^{-1} \begin{bmatrix} S_v\\ S_{v+1}\\ \cdot\\ \cdot\\ \cdot\\ S_{2v} \end{bmatrix} \quad \cdots\cdots (58)$$

가 된다. 여기서 M은 誤症要素로 구성된 행렬이다. 즉,

$$M=\begin{bmatrix} S_1 & S_2 & \cdots & S_v\\ S_2 & S_3 & \cdots & S_{v+1}\\ \cdot & & & \cdot\\ \cdot & & & \cdot\\ \cdot & & & \cdot\\ S_v & S_{v+1} & \cdots & S_{2v-1} \end{bmatrix} \quad \cdots (59)$$

이다. 受信多項式 r(x)로 부터 계산된 誤症을 식(58)에 대입하여 誤謬位置多項式의 계수를 결정할 수 있다면 이를 이용하여 誤謬位置番號 Z_i를 구할 수 있게 된다. 誤謬値 Y_k, $1\leq k\leq v$, 역시, 誤症要素 s_λ, $1\leq\lambda\leq 2t$와 誤謬位置番號 Z_k, $1\leq k\leq v$를 이용하면 쉽게 결정할 수 있다. 이제, 스트림 暗號시스템에서의 誤謬制御를 위해 위에서 논의된 內部誤謬制御와 外部誤謬制御로 구분해서 有限體 $GF(2^5)$에서의 DSEC (31, 27) RS符號를 다음 세가지 부류의 暗號시스템, 즉, (1) 키 自動키 暗號시스템, (2) 暗號文 歸還 暗號시스템, (3) 平文 歸還 暗號시스템에 적용시킨다. 우선, 다음과 같은 메세지로 구성된 平文을 고려하자.

> The aim of cryptography is to hide the clear form of plaintext by making it unreadable.

暗號化 과정은 일반적으로 平文내의 각 文字들을 2진수(binary)로 변환하는 것으로 부터 시작된다. 일부 暗號시스템에서는 ASCII 符號를 사용하는 것이 편리하므로, 여기에서도 위의 平文을 ASCII 符號를 이용하여 변환하도록 하자.

表 11. ASCII 부호로 표현된 平文 내용

01010100 01101000 11100101 00100000 01100001 11101001
01101101 00100000 11101111 11100110 00100000 11100011
11110010 01111001 01110000 11110100 11101111 01100111
11110010 01100001 01110000 01101000 01111001 00100000
11101001 01110011 00100000 11110100 11101111 00100000
01101000 11101001 01100100 11100101 00100000 11110100
01101000 11100101 00100000 11100011 11101100 11100101
01100001 11110010 00100000 11100110 11101111 11110010
01101101 00100000 11101111 11100110 00100000 01110000
11101100 01100001 11101001 01101110 11110100 11100101
11111000 11110100 01100010 01111001 00100000 01101101
01100001 01101011 11101001 01101110 01100111 00100000
11101001 11110100 00100000 01110101 01101110 11110010
11100101 01100001 01100100 01100001 01100010 11101100
11100101 10101110

5. 2 內部誤謬制御

스트림 暗號시스템에서의 誤謬制御를 위해서는 內部誤謬制御가 이상적이다. 만일, 外部誤謬制御方式을 도입하여 暗號文을 전송할 경우 전송로상에서 발생된 誤謬의 영향은 復號과정중 誤謬傳播현상으로 인하여 많은 誤謬를 발생시킬 것이다. 5. 2. 1절에서는 DSEC (31, 27) RS符號를 이용한 키 自動키 暗號시스템의 內部誤謬制御 특성에 대해 분석한다.

5. 2. 1. 키 自動키 暗號시스템을 위한 內部制御

그림 20(a)과 같이 구성된 內部誤謬制御方式에 의해 平文 X는 먼저 그림 21로 부터 생성된 키 數例 K에 의해 暗號化되고, 이어서 t=2인 RS符號의 生成多項式 $g(x)=(x+\alpha)(x+\alpha_2)\cdots(x+\alpha^{2t})$에 의해 符號化된다. 그림 21은 초기상태가 S=(11110)인 키 생성기로 키 數例은 $2^5-1=31$ 비트의 週期(1001101001000010101110110001111)를 갖는다. 그러므로, 暗號化를 위한 키 數例은 表 12와 같다.

表 12. 暗號化를 위한 키 數例

10011010 01000010 10111011 00011111 00110100 10000101
01110110 00111110 01101001 00001010 11101100 01111100
11010010 00010101 11011000 11111001 10100100 00101011
10110001 11110011 01001000 01010111 01100011 11100110
10010000 10101110 11000111 11001101 00100001 01011101
10001111 10011010 01000010 10111011 00011111 00110100
10000101 01110110 00111110 01101001 00001010 11101011
01111100 11010010 00010101 11011000 11111001 10100100
00101011 10110001 11110011 01001000 01010111 01100011
11100110 10010000 10101110 11000111 11001101 00100001
01011101 10001111 10011010 01000010 10111011 00011111
00110100 10000101 01110110 00111110 01101001 00001010
11101100 01111100 11010010 00010101 11011000 11111001
10100100 00101011 10110001 11110011 01001000 01010111
01100011 11100110

表 11과 表 12로 부터 대응되는 각 심볼간의 2원합을 수행하면, 表 13과 같은 暗號文으로 平文을 暗號化 할 수 있다.

表 13. ASCII 부호로 표현된 暗號文

11001110 00101010 01011110 00111111 01010101 01101100
00011011 00011110 10000110 11101100 11001100 10011111
00100000 01101100 10101000 00001101 01001011 01001100
01000011 10010010 00111000 00111111 00011010 11000110
01111001 11011101 11100111 00111001 11001110 01111101
11100111 01110011 00100110 01011110 00111111 11000000
11101101 10010011 00011110 10001010 11100110 00001001
00011101 00100000 00110101 00111110 00010110 01010110
01000110 10010001 00011100 10101110 01110111 00010011
00001010 11110001 01000111 10101001 00111001 11000100
10100101 01111011 11111000 00111011 10011011 01110010
01010101 11101110 10011111 01010000 00001110 00101010
00000101 10001000 11110010 01100000 10110110 00001011
01000001 01001010 11010101 10010010 00101010 10111011
10000110 01001000

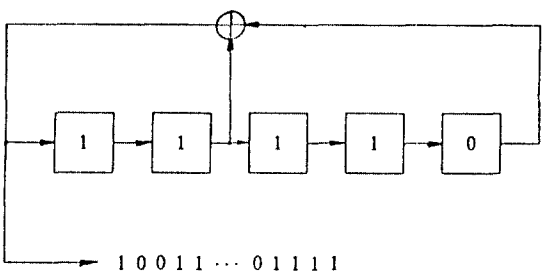


그림 21 탭 계수가 T=(01001)인 키 數例 生成器

暗號文을 다시 符號化하기 위해 $GF(2^5)$상의 심볼들로 구성된 DSEC(31, 27) RS符號를 고려하자. m=5와 t=2이므로 k=31−2×2=27개의 暗號文 심볼을 입력정보로 하여 符號器에 입력시킨다. 이를 위해, 우선, 表 13의 暗號文 내용을 5비트 심볼로 블럭화 해야 한다. 설명을 돕기 위하여, 符號化하고자 하는 暗號文 중 밑줄친 부분을 表 14에서와

表 14. 符號化 하고자 하는 暗號文

10100	11000	10000	11100	10010
00111	00000	11111	10001	10101
10001	10011	11001	11011	10111
10011	10011	10011	10011	10011
11101	11100	11101	11001	10010
01100	10111			

같이, 5비트로 구성된 27개의 심볼로 다시 정돈하였다.

表 14의 27개 심볼들은 $Y=(y_0, y_1, \cdots, y_{26})$로 表現되는 暗號文 심볼들이다. 符號長(code length)이 $2^5-1=31$ 심볼인 이중 심볼 誤謬訂正 RS符號의 生成多項式은 다음과 같다.

$$g(x)=(x+\alpha)(x+\alpha^2)(x+\alpha^3)(x+\alpha^4)$$
$$=\alpha^{10}+\alpha^{29}x+\alpha^{19}x^2+\alpha^{24}x^3+x^4 \quad \cdots\cdots (60)$$

식 (60)을 이용하면, 暗號文 Y는 表 15와 같이 暗號文 c_Y로 符號化된다.

表 15. 符號化된 暗號文 c_Y

10100	11000	10000	11100	10010
00111	00000	11111	10001	10101
10001	10011	11001	11011	10111
10011	10011	10011	10011	10011
01100	11100	11101	11001	10010
01100	10111	10001	10110	10000
01000				

符號化된 暗號文 $c_Y=(c_{30}, c_{29}, \cdots, c_4, c_3, c_2, c_1, c_0)$의 검사심볼(parity-check symbol)은 마지막 4개의 심볼블럭이다. 즉, $c_3=(10001)$, $c_2=(10110)$, $c_1=(10000)$과 $c_0=(01000)$가 된다. 暗號文 c_Y가 전송된 후 $r_Y=(r_{30}, r_{29}, \cdots, r_0)$가 수신되어 表 16과 같다고 가정하자.

表 16. 수신된 暗號文 r_Y

10100	11000	10000	11100	10010
00111	00000	11111	10001	10101
10001	10011	01000	11011	10111
10011	10011	10011	10011	10011
01100	01110	11101	11001	10010
01100	10111	10001	10110	10000
01000				

그리고, 誤謬多項式은 誤謬위치가 x^9과 x^{18}에서 誤謬値가 α^{29}와 α^{10}인 $e(x)=\alpha^{29}x^9+\alpha^{10}x^{18}$이라 할때, $e_Y=c_Y+r_Y$으로 부터 $e_9=c_9+r_9=(11100)+(01110)=(10010)=\alpha^{29}$가 되고, $e_{18}=c_{18}+r_{18}=(11001)+(01000)=(10001)=\alpha^{10}$이므로 정확한 誤謬형태 표현임을 입증할 수 있다. 수신백터의 誤症要素는 誤謬多項式 $e(x)$에서 x 대신에 α, α^2, α^3, α^4을 각각 대입함으로써 계산될 수 있으며 이는 다음과 같다.

$$s_1=e(\alpha)=\alpha^7+\alpha^{28}=\alpha=(01000)$$
$$s_2=e(\alpha^2)=\alpha^{16}+\alpha^{15}=\alpha^2=(00100)$$
$$s_3=e(\alpha^3)=\alpha^{25}+\alpha^2=\alpha^{14}=(10111) \quad \cdots\cdots (61)$$
$$s_4=e(\alpha^4)=\alpha^3+\alpha^{20}=\alpha^2=(00100)$$

(31, 27) RS 符號는 二重 심볼 誤謬訂正符號이므로, 식 (55)의 오증요소행렬식에 의한 表現은 다음과 같다.

$$M=\begin{bmatrix} s_1 & s_2 \\ s_2 & s_3 \end{bmatrix}=\begin{bmatrix} \alpha & \alpha^2 \\ \alpha^2 & \alpha^{14} \end{bmatrix} \quad \cdots\cdots (62)$$

따라서, 誤謬位置多項式 $\alpha(x)$의 계수는 다음과 같이 식(58)을 이용하여 결정할 수 있다.

$$\begin{bmatrix} \sigma_2 \\ \sigma_1 \end{bmatrix}=\begin{bmatrix} s_1 & s_2 \\ s_2 & s_3 \end{bmatrix}^{-1}\begin{bmatrix} s_3 \\ s_4 \end{bmatrix} \quad \cdots\cdots (63)$$

$M=\alpha^{23}\neq 0$이므로, σ_2와 σ_1는

$$\sigma_2=(s_3^2+s_2s_4)/(s_1s_3+s_2^2)$$
$$\sigma_1=(s_2s_3+s_1s_4)/(s_1s_3+s_2^2) \quad \cdots\cdots (64)$$

으로 부터 쉽게 구할 수 있으며, 식(61)을 식(64)에 대입함으로써, $\sigma(x)$의 계수는 다음과 같이 결정된다.

$$\sigma_2=(s^{28}+\alpha^4)/\alpha^{23}=\alpha^{27}$$
$$\sigma_1=(s^{16}+\alpha^3)/\alpha^{23}=\alpha^{25} \quad \cdots\cdots (65)$$

결국, 誤謬位置多項式 $\sigma(x)$는

$$\sigma(x)=1+\alpha^{25}x+\alpha^{27}x^2 \quad \cdots\cdots (66)$$

가 되고, $x=\alpha^{13}$과 $x=\alpha^{22}$에 대하여 $\sigma(x)=0$이므로, 誤謬位置番號는 각각

$$Z_1=1/\alpha^{22}=\alpha^9$$
$$Z_2=1/\alpha^{13}=\alpha^{18} \quad \cdots\cdots (67)$$

이 됨을 알 수 있다. 또한, $v=2$일때 식(55)는

$$s_1=Y_1Z_1+Y_2Z_2$$
$$s_2=Y_1Z_1^2+Y_2Z_2^2 \quad \cdots\cdots (68)$$

이므로 식(68)로 부터, 誤謬値 Y_1와 Y_2는 다음과 같이 표현된다.

$$Y_1=(s_2Z_2+s_2)/(Z_1Z_2+Z_1^2)$$
$$Y_2=(s_1Z_2+s_2)/(Z_1Z_2+Z_1^2) \quad \cdots\cdots (69)$$

식(67)로 부터 Z_1과 Z_2를 식 (61)로 부터 s_1과 s_2를 식(69)에 각각 대입하면, 誤謬値 Y_1과 Y_2는

$$Y_1=(\alpha\ \alpha^{18}+\alpha^2)/(\alpha^8\ \alpha^{18}+(\alpha^9)^2)=\alpha^{29}$$
$$Y_2=(\alpha\ \alpha^9+\alpha^2)/(\alpha^9\ \alpha^{18}+(\alpha^{18})^2)=\alpha^{10}$$

이 된다. 그러므로, 誤謬多項式 $e(x)$는

$$e(x)=\alpha^{29}x^9+\alpha^{10}x^{18} \quad \cdots\cdots (70)$$

으로 이에 대한 가정이 올바르다는 사실이 입증되었다. 결국, 수신된 暗號文 $r_Y=(r_{30}, r_{29}, \cdots, r_0)$은 DSEC (31, 27) RS 符號를 이용하여, 符號化된 暗號文 $c_Y=(c_{30}, c_{29}, \cdots c_0)$으로 誤謬訂正이 수행된다. r_9과 r_{18}에서 발생된 誤謬는 $r_9=(01110)\rightarrow c_9=(11100)$과 $r_{18}=(01000)\rightarrow c_{18}=(11001)$와 같이 각각 誤謬訂正에 의해 완전히 복구된다. 마지막으로, 表 12와 같은 동일한 키 數例를 이용한 키 자동키 復號器는 表 11에 나타난 것과 동일한 ASCII 平文으로 복구한다. 따라서, 복구된 ASCII 平文을 平文원문으로 변환하므로써 다음과 같은 완벽한 형태가 만들어지게 된다.

> The aim of cryptography is to hide the clear form of plaintext by making it unreadable.

5. 2. 2. 暗號文 歸還 暗號시스템을 위한 內部制御

탭 계수가 T=(01001)이며, 그림 22와 같이 m=5인 暗號文 歸還 생성기의 경우를 고려해 보자. 또한, 전송도중 $e_{23}=(11011)=\alpha^{16}$과 $e_{22}=(11111)=\alpha^{15}$으로 발생된 심볼 誤謬를 정정하기 위해 DSEC (31, 27) RS符號를 이용한다. 만일 ASCII 平文이 5. 1절에서의 表 11과 같다면, 그림 22으로 부터 생성된 키 비트 數例은 表 17과 같이 된다.

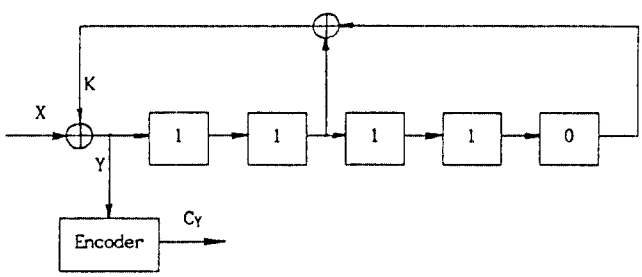


그림 22 暗號文 歸還 키 生成器

表 17. 暗號文 歸還 方式에서의 키 數例

10001001 10010001 10010111 00011110 01110101 10110101
11001101 00001011 10000000 10101001 10010001 11000001
10001100 01110010 10101000 11001111 00101110 01000000
11111010 01001011 11110101 01101000 00011011 10110000
10011110 01111010 00000001 01101011 00000101 11100111
11010011 00101111 10001101 00111001 11011001 10010010
10001110 10100011 10011011 00101101 11110110 01110001
10011001 11001110 11011001 10010111 11000010 00010110
00110100 10101111 10101011 00011000 01100010 10100010
00100100 01001011 11011100 11000111 00111100 01100100
01101000 10011000 01100000 10100000 10101111 10000101
01001011 11110010 10010100 10010000 01110100 01000011
11010110 00000010 00110101 11000000 11000111 00111101
10101001 01100000 01000000 00110111 00100010 00110001
10111100 10000010

表 11의 平文을 表 17의 키 數例로 暗號化하면 表 18과 같다.

表 18. 暗號文 歸還 方式에 의한 暗號文

11011101 11111001 01110010 00111110 00010100 01011100
10100000 00101011 01101111 01001111 10110001 00100010
01111110 00001011 11011000 00111011 11000001 00100111
00001000 00101010 10000101 00000000 01100010 10010000
01110111 00001001 00100001 10011111 11101010 11000111
10111011 11000110 11101001 11011100 11111001 01100110
11100110 01000110 10111011 11001110 00011010 10010100
11111000 00111100 11111001 01110001 00101101 11100100
01011001 10001111 01000100 11111110 01000010 11010010
11001000 00101010 00110101 10101001 11001000 10000001
10010000 01101100 00000010 11011001 10001111 11101000
00101010 10011001 01111101 11111110 00010011 01100011
00111111 11110110 00010101 10110101 10101001 11001111
01001100 00000001 00100100 01010110 01000000 11011101
01011001 00101100

만일 表 18의 밑줄친 부분과 일치하는 暗號文 Y가 생성다항식 $g(x)=\alpha^{10}+\alpha^{29}x+\alpha^{19}x^2+\alpha^{24}x^3+x^4$에 의해 符號化 된다면 暗號文 c_Y는 表 19와 같다.

表 19. 表 18의 밑줄친 부분에 대해 符號化된 暗號文

10010	01110	00010	00001
01010	10000	10100	00000
00110	00001	00100	00011
10111	00001	00100	10000
11001	11111	11010	10110
00111	10111	01111	00011
01110	10011	10111	01100
01110	11001	00111	

만일 전송로상에서 $c_{23}=(00000)=0$와 $c_{22}=(00110)=\alpha^{20}$에서 심볼誤謬 $e_{23}=(11011)=\alpha^{16}$과 $e_{22}=(11111)=\alpha^{15}$으로 발생하였다면, 수신된 해당심볼들은 $r_{23}=(11011)=\alpha^{16}$과 $r_{22}=(11001)=\alpha^{17}$이 될 것이다. 여기서, 誤謬多項式이 $e(x)=\alpha^{15}x^{22}+\alpha^{16}x^{23}$의 형태로 발생되었다고 가정하자. 발생된 誤謬를 訂正하기 위해서는 우선, 수신된 벡터 r_Y으로 부터 誤症을 계산해야 하며, e(x)에 α, α^2, α^3과 α^4을 대입함으로써 誤症要素 s_i, $1\leq i\leq 4$를 모두 구할 수 있다.

$$s_1=e(\alpha)=\alpha^6+\alpha^8=\alpha^{11}=(11100)$$
$$s_2=e(\alpha^2)=\alpha^{28}+1=\alpha^{26}=(11101) \quad \cdots\cdots\cdots\cdots (71)$$
$$s_3=e(\alpha^3)=\alpha^{19}+\alpha^{23}=\alpha^{29}=(10010)$$
$$s_4=e(\alpha^4)=\alpha^{10}+\alpha^{15}=\alpha^{12}=(01110)$$

또한, $v=2$이므로 오증요소행렬 M은

$$M=\begin{bmatrix} s_1 & s_2 \\ s_2 & s_3 \end{bmatrix}=\begin{bmatrix} \alpha^{11} & \alpha^{26} \\ \alpha^{26} & \alpha^{29} \end{bmatrix} \quad \cdots (72)$$

와 같이 된다. 식(64)를 이용하면, 誤謬位置多項式 $\sigma(x)$의 계수는

$$\sigma_2=(\alpha^{27}+\alpha^7)/(\alpha^9+\alpha^{21})=\alpha^{15}/\alpha=\alpha^{14}$$
$$\sigma_1=(\alpha^{24}+\alpha^{23})/(\alpha^9+\alpha^{21})=\alpha^{10}/\alpha=\alpha^9 \quad \cdots\cdots (73)$$

이 되므로 誤謬位置多項式 $\sigma(x)$는

$$\sigma(x)=1+\alpha^{9}x+\alpha^{14}x^{2} \quad \cdots\cdots (74)$$

이다. $\sigma(x)=0$이 되는 $\sigma(x)$의 근은 α^{3}와 α^{9}이므로 誤謬位置番號는 다음과 같다.

$Z_1=1/\alpha^{8}=\alpha^{23}=(11110)$
$Z_2=1/\alpha^{9}=\alpha^{22}=(10101)$

誤謬値 Y_1과 Y_2는 식(69)로 부터

$Y_1=(\alpha^{2}+\alpha^{26})/(\alpha^{14}+\alpha^{15})=\alpha^{17}/\alpha=\alpha^{16}$
$=(11011)=e_{23}$
$Y_2=(\alpha^{3}+\alpha^{26})/(\alpha^{14}+\alpha^{13})=\alpha^{15}/1=\alpha^{15}$
$=(11111)=e_{23}$

임을 알 수 있다. 결국, 誤謬多項式은 식(75)와 같다.

$$e(x)=\alpha^{15}x^{22}+\alpha^{16}x^{23} \quad \cdots\cdots (75)$$

$c=e+r$이므로, $c_{22}=e_{22}+r_{22}=Y_2+r_{22}=(11111)+(11001)=(00110)$이며, $c_{23}=e_{23}+r_{23}=Y_1+r_{23}=(11011)+(11011)=(00000)$이 된다. 이와같이 表 19의 符號化된 暗號文은 誤謬정정 방식을 사용하여 완전하게 복구된다. 마지막으로, 검사심볼을 제거하고, 表 17의 키 수열을 이용하여 暗號文을 復號하게 되고, 表 11에 나타난 平文의 완벽한 형태로 복구할 수 있다.

5. 2. 3. 平文 歸還 暗號시스템을 위한 內部制御

(31, 27) RS符號를 이용하여, 平文 歸還 시스템에 대한 誤謬訂正 문제를 본절에서 논의한다. 平文 歸還 시스템에 대한 키 생성기는 그림 23과 같다. 심볼誤謬가 $c_{24}=Y_1=(01100)=\alpha^{9}$과 $c_7=Y_2=(10100)=\alpha^{5}$에서 발생되었다고 가정하자. 그리고 平文 歸還 생성기에서 생성된 키 數例은 表 20과 같다.

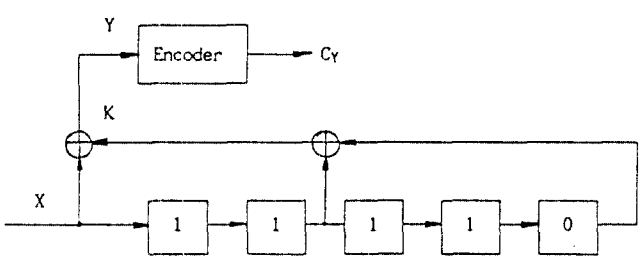


그림 23 平文 歸還 키 生成器

表 20의 키 數例을 이용하여 表 11에서의 平文을 暗號化하면 表 21과 같은 暗號文 심볼들을 얻을 수 있다.

앞에서 가정한 바와 같이 表 21의 밑줄친 부분에 대한 暗號文은 前절에서 주어진 것과 같은 生成多項式 $g(x)$을 이용하여 符號化하며 그 결과 符號化된 暗號文 c_Y는 表 22와 같다.

만일 전송로상에서 심볼誤謬가

$c_{24}=(11111)\rightarrow r_{24}=(01100)$
$c_7=(01010)\rightarrow r_7=(10100)$

와 같은 위치에서 발생하였다면, 誤謬値는 $e_{24}=Y_1=(10011)=\alpha^{25}$과 $e_7=Y_2=(11110)=\alpha^{23}$가 되며 誤謬多項式은 $e(x)=\alpha^{23}x^{7}+\alpha^{25}x^{24}$이 된다. 이 경우,

表 20. 平文 歸還 方式에 의한 키 數例

10101111 10111001 01111110 01100001 00011011 01110101
00010000 00100001 00111100 10000110 10111001 00111111
11100011 00001101 10010111 10111010 10011100 10100010
11000011 00001011 01010111 10011001 01011101 10000001
00111101 00010111 01010001 00111010 10011100 10110001
00011001 01111101 00010010 00011110 01100001 00111010
10111001 01111110 01100001 00111111 11100100 01011110
01110011 01110011 00011001 00111110 10001100 10000011
00001000 00100001 00111100 10000110 10111001 00011111
10111100 01111011 01110101 00010000 11001010 10011110
01010001 11111010 10111011 10001101 10000001 00011000
00110011 01010001 10100101 00010000 11101010 11110001
00111101 00110010 10101001 00011110 11110000 11001011
00101110 01110011 01010010 00111011 01010011 10101100
01011110 01000110

表 21. 平文 歸還 方式에 의한 暗號文

11111011 11010001 10011011 01000001 01111010 10011100
01111101 00000001 11010011 01100000 10011001 11011100
00010001 01110100 11100111 01001110 01110011 11000101
00110001 01101010 00100111 11110001 00100100 10100001
01110001 10010100 01110110 11111011 01000001 11001110
01110001 10010100 01110110 11111011 01000001 11001110
11010001 10011011 01000001 11011100 00001000 10111011
00010010 10000001 00111001 11011000 01100011 01110001
01100101 00000001 11010011 01100000 10011001 01101111
01010000 00011010 10011100 01111110 00111110 01111011
10101001 00001110 11011001 11110100 10100001 01110101
01010010 00111010 01001100 01111110 10001101 11010001
11010100 11000110 10001001 01101011 10011110 00111001
11001011 00010010 00110110 01011010 00110001 01000000
10111011 11101000

誤症要素는 다음과 같이 e(x)로 부터 계산된다.

$s_1=e(\alpha)=\alpha^{30}+\alpha^{18}=\alpha^{10}=(10001)$

表 22. 表 21의 밑줄친 부분에 대해 符號化된 暗號文

11100 01010 01100 01011 01010
00100 11111 11000 10010 01001
01000 01110 10100 01100 10001
11000 11100 11100 11100 11100
10001 01110 00110 01010 00111
01101 11110 10011 11100 10000
00010

$s_2=e(\alpha^2)=\alpha^6+\alpha^{11}=\alpha^8=(10111)$ ····· (76)

$s_3=e(\alpha^3)=\alpha^{13}+\alpha^4=\alpha^{20}=(00110)$

$s_4=e(\alpha^4)=\alpha^{20}+\alpha^{28}=\alpha^9=(01011)$

v=2이므로 식(59)로 부터 誤症要素로 구성된 행렬은 다음과 같이 표현된다.

$$M=\begin{bmatrix} s_1 & s_2 \\ s_2 & s_3 \end{bmatrix}=\begin{bmatrix} \alpha^{10} & \alpha^8 \\ \alpha^8 & \alpha^{20} \end{bmatrix} \cdots (77)$$

σ(x)의 계수는 식(63)과 식(77)을 이용하면

$\sigma_2=(\alpha^9+\alpha^{17})/(\alpha^{30}+\alpha^{16})=\alpha^{29}/\alpha^{29}=1$

$\sigma_1=(\alpha^{28}+\alpha^{19})/(\alpha^{30}+\alpha^{16})=\alpha^4/\alpha^{29}=\alpha^6$ ····· (78)

으로 결정된다. 따라서, 誤謬位置多項式 σ(x)을 구할 수 있다.

$\sigma(x)=1+\alpha^6x+x^2$ ····· (79)

$x=\alpha^7$과 α^{24}에서 σ(x)=0이므로, 근의 역을 취함으로써 誤謬位置番號는 쉽게 찾을 수 있고 다음과 같다.

$Z_1=1/\alpha^7=\alpha^{24}=(01111)$

$Z_2=1/\alpha^{24}=\alpha^7=(00101)$ ····· (80)

또한, 식(69)를 이용하여 誤謬値를 계산하면

$Y_1=(\alpha^{17}+\alpha^8)/(1+\alpha^{17})\alpha^{24}/\alpha^{30}=\alpha^{25}=(10011)$

$Y_2=(\alpha^3+\alpha^8)/(1+\alpha^{14})\alpha^5/\alpha^{13}=\alpha^{23}=(11110)$

····· (81)

이 됨을 알 수 있다. 결국, 誤謬多項式은 예상했던 대로 다음과 같다.

$$e(x)=\alpha^{23}x^7+\alpha^{25}x^{24} \quad \cdots\cdots (82)$$

일단 誤謬심볼들이 (31, 27) RS符號에 의해 訂正되기만 하면, 檢査심볼을 제거한 후의 復號된 暗號文은 表 20의 數例를 사용하여 쉽게 復號되고 表 11에서와 같은 ASCII 平文 형태로 완전히 복구된다.

지금까지는 平文 귀환 暗號시스템에서의 內部誤謬制御 방식에 대하여 분석하였고, 誤謬制御가 잘 적용됨을 보여 주었다.

5. 3 外部誤謬制御

본 절에서는 符號化(encoding)과정이 暗號化(enciphering)과정보다 먼저 행하여 지고, 復號(deciphering)과정이 誤謬정정(decoding)과정보다 앞선 경우를 고려한다. 이러한 종류의 시스템 구성을 外部誤謬制御라고 한다. 앞 절에서 논의된 바와 같이, DSEC(31, 27) RS부호를 外部誤謬制御用으로 사용한다. 또한, 본 절에서 사용하는 平文은 5. 2 절에서 사용된 것과 동일한 문장을 사용한다.

5. 3. 1. 키 自動키 暗號시스템을 위한 外部制御

키 自動키 暗號器에서는 暗號文 복호과정으로 인한 어떠한 誤謬傳播도 없다. 그러므로 키 自動키 暗號器에 대하여, 內部 혹은 外部誤謬制御의 결과는 동일하다. 그림 24에서 키 數例 K는 탭 계수 T=(01001)와 초기치 벡터 S=(11110)를 갖는 키 自動키 生成器로 부터 생성된다. 또한, 키 비트 數例는 平文과 독립적으로 생성되며, 同期 스트림 暗號시스템은 誤謬가 傳播되지 않는다는 장점을 갖기 때문에, 한 비트에 발생된 전송 誤謬는 연속되는 비트 數例에 전혀 영향을 주지 않는다. 따라서, 同期 스트림 暗號시스템은 오류정정부호와 함께 사용될 때 어떠한 인증효과도 제공하지 못한다.

DSEC을 위한 (31, 27) RS符號의 生成多項式 g(x)로, 表 23의 밑줄친 부분을 符號化하는 문제를 고려해보자.

表 23. 符號化될 부분(밑줄친 부분)을 포함한 ASCII 平文

01010100 01101000 11100101 00100000 00110001 11101001
01101101 00100000 11101111 11100110 00100000 11100011
11110010 01111001 01110000 11110100 11101111 01100111
11110010 01100001 01110000 01101000 01111001 00100000
11101001 01110011 00100000 11110100 11101111 00100000
01101000 11101001 01100100 11100101 00100000 11110100
01101000 11100101 00100000 11100011 11101100 11100101
01100001 11110010 00100000 11100110 11101111 11110010
01101101 00100000 11101111 11100110 00100000 01110000
11101100 01100001 11101001 01101110 11110100 11100101
11111000 11110100 01100010 01111001 00100000 01101101
01100001 01101011 11101001 01101110 01100111 00100000
11101001 11110100 00100000 01110101 01101110 11110010
11100101 01100001 01100100 01100001 01100010 11101100
11100101 10101110

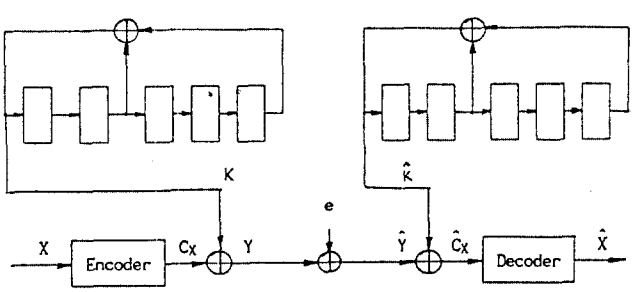


그림 24 키 自動키 暗號器에서의 外部誤謬制御

각 심볼은 GF(2^5)상의 원소들로 구성되기 때문에, 심볼당 5비트 단위로 밑줄친 부분은 재구성하면, 平文 $X=(x_0, x_1, x_2, \cdots, x_{26})$는 表 24와 같은 27개의 심볼로 나타낼 수 있다.

表 24. 재구성후의 平文 X

10110 01111 11100 10011 00001 01110
00001 10100 00111 10010 01000 00111
01001 01110 01100 10000 01111 01001
11011 11001 00000 01101 00011 10100
10110 01001 11001

符號化된 平文 $c_x=(c_{30}, c_{29}, \cdots, c_1, c_0)$ 表 25에서와 같이 열거된다.

表 25. 符號化된 平文 c_x

10110 01111 11100 10011 00001 01110
00001 10100 00111 10010 01000 00111
01001 01110 01100 10000 01111 01001
11011 11001 00000 01101 00011 10100
10110 01001 11001 01001 01001 10110
11010

表 25에서, 檢査심볼은 $c_3=(01001)$, $c_2=(01001)$, $c_1=(10110)$, $c_0=(11010)$가 된다. 그림 24의 키 自動키 생성기에서 생성된 키 數例 $K=(k_{30}, k_{29}, \cdots, k_0)$ 역시 5비트 단위로 재구성하면 表 26과 같다.

表 26. 키 自動키 暗號化에 사용되는 키 數例

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

暗號化 과정은 表 25의 符號化된 平文과, 表 26의 키 數例과의 2원 연산으로 행해진다. 따라서, 그 결과 暗號文 $Y=(y_{30}, y_{29}, \cdots, y_1, y_0)$는 表 27과 같다.

表 27. 暗號化된 暗號文 Y

00101 00110 11101 11000 10111 01001
11000 00000 10111 00111 10011 00100
10101 10100 00100 11010 10010 11000
00101 10100 00100 01000 01101 01100
01001 01111 01011 01011 11110 11010
10101

이제, 수신된 暗號文을 $r=(r_{30}, r_{29}, \cdots, r_1, r_0)$이라 하고, Y를 전송하는 동안에 $y_{18}=(10101)\rightarrow r_{18}=(00100)$과 $y_9=(01000)\rightarrow r_9=(11010)$으로, 2개의 심볼誤謬가 발생되었다고 가정한다. 수신된 暗號文은 表 28과 같다.

表 28. 키 自動키 暗號기로 부터 수신된 暗號文 r

00101 00110 11101 11000 10111 01001
11000 00000 10111 00111 10011 00100
00100 10100 00100 11010 10010 11000
00101 10100 00100 11010 01101 01100
01001 01111 01011 01011 11110 11010
10101

表 28과 같이 수신된 暗號文 r을 表 26의 키 비트 數例로 復號하면, 表 29에서와 같이 符號化된 平文 c_x'를 구할 수 있다.

表 29. 2개의 심볼誤謬가 포함되어 復號된 c_x'

10110 01111 11100 10011 00001 01110
00001 10100 00111 10010 01000 00111
11000 01110 01100 10000 01111 01001
11011 11001 00000 11111 00011 10100
10110 01001 11001 01001 01001 10110
11010

復號과정을 거친 후에, $c_{18}=(01001)\rightarrow c_{18}'=(11000)$이며, $c_9=(01101)\rightarrow c_9'=(11111)$로 되어 있음을 表 29로 알 수 있다. 원래의 平文으로 복구하기 위해서는 DSEC (31, 27) RS符號를 사용하여 復號한다.

內部誤謬制御와 外部誤謬制御方法을 비교하기 위하여, 5. 2. 1절에서 사용되었던 것과 같은 형태의 誤謬多項式인 $e(x)=\alpha^{29}x^9+\alpha^{10}x^{18}$을 사용하기로 한다. 그러므로 符號化된 平文 c_x의 誤症要素는 $s_1=\alpha$, $s_2=\alpha^2$, $s_3=\alpha^{14}$, $s_4=\alpha^2$이다. 이 경우 誤謬位置多項式 $\sigma(x)$는 그 계수가 $\sigma_1=\alpha^{25}$과 $\sigma_2=\alpha^{27}$이므로, $\sigma(x)=1+\alpha^{25}x+\alpha^{27}x^2$가 된다. $\sigma(x)$의 근의 역수를 취함으로써 결정되는 誤謬位置番號는 각기 $Z_1=\alpha^{24}$와 $Z_2=\alpha^7$이다. 따라서, s_1, s_2, Z_1과 Z_2를 식(69)에 대입시키면, 誤謬値 $Y_1=\alpha^{29}$과 $Y_2=\alpha^{10}$을 계산할 수 있다. 이렇게 해서 誤謬多項式 $e(x)$는 가정한 바와 같이 구해진다.

訂正된 심볼은 $c_i=c_i'+e_i=c_i'+Y_i$로써 표시되는데, 이를 i=9와 18에 적용시키면, $c_9=c_9'+Y_1=(11111)+(10010)=(01101)$과 $c_{18}=c_{18}'+Y_2=(11000)+(10001)=(01001)$으로 표시된다. 결국, 2개의 심볼 誤謬는 완전하게 訂正되며, 원래의 平文은 表 24에서 열거된 4개의 검사심볼을 제거함으로써 완전복구된다.

키 自動키 暗號器의 분석결과로써, 誤謬訂正方法으로는 內部制御를 사용하든지 혹은 外部制御를 사용하든지는 그리 중요하지 않다. 그러므로, 誤謬訂正 목적을 위해서는 두가지 방법 중의 어느 방법을 사용해도 동일하다고 결론지을 수 있다.

5. 3. 2. 暗號文 歸還 暗號시스템을 위한 外部制御

暗號文 歸還 暗號器에서의 外部誤謬制御시스템은 그림 25과 같다. 지금까지의 분석내용과 비교하기 위하여, 키 생성기의 탭 계수 T=(01001)과 초기치백터 S=(11110)를 다시 사용 하기로 한다. 앞절의 경우와 마찬가지로, 동일한 平文을 사용하기 때문에, 27개의 심볼로 구성된 平文은 表 25와 같이 31개의 심볼들로 符號化된다.

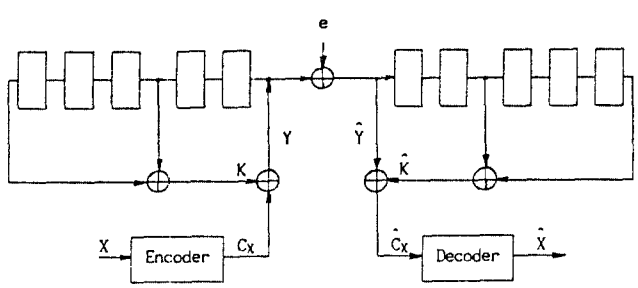


그림 25 暗號文 歸還 外部誤謬制御 시스템

暗號文 Y는 暗號文에 의해 여기 되는 키 생성기로부터 발생되는 키 수열 K와 符號化된 平文 c_x를 暗號化함으로써 얻어진다. 對稱形態의 暗號시스템(symmetric cryptosystem)에서의 송신자(sender)와 수신자(receiver)에 의해 공유되는 키 數例은 表 30과 같다.

表 30. 暗號시스팀의 양단이 공유하는 키 數例

01011 10010 10111 10011 00000 01000
10010 01101 10101 00111 11001 11111
11101 10011 10010 01000 11100 01011
10000 10001 01010 11110 01001 11001
00000 00101 01000 11100 11001 10001
11110

符號化된 平文 c_x를 키 수열 K에 의하여 暗號化하면 表 31과 같은 暗號文이 생성된다.

表 31. 平文 c_x를 K로 暗號化한 暗號文 Y

11101 11101 01011 00000 00001 00110
10011 11001 10010 10101 10001 11000
10100 11101 11110 11000 10011 00010
01011 01000 01010 10011 01010 01101
10110 01100 10001 10101 10000 00111
00100

전송도중 暗號文 Y에 $y_{23}=(11001)\rightarrow y_{23}'=(00110)$과 $y_{22}=(10010)\rightarrow y_{22}'=(01001)$와 같이 2개의 심볼誤謬가 발생하였다고 가정하자. 그러므로 수신측에서 생성되는 키 數例 K는 表 30과는 다른 형태로 생성될 것이다. 키 數例에 발생한 誤謬는 表 32에서 표시된 바와 같이 3개의 심볼에 나타나 있다.

表 32. 수신측에서 생성된 키 數例 K

01011 10010 10111 10011 00000 01000
10010 01010 10100 00100 11001 11111
11101 10011 10010 01000 11001 11111
10000 10001 01010 11110 01001 11001
00000 00101 01000 11100 11001 10001
11110

이와같이 3개의 심볼 誤謬가 포함된 키 수열 K와 暗號文 Y의 2원합으로 表 33에서 표시된 바와같은 平文 c_x'이 구해진다.

表 33. 復號된 平文 c_x'

10110 01111 11100 10011 00001 01110
00001 01100 11101 10001 01000 00111
01001 01110 01100 10000 01111 01001
11011 11001 00000 01101 00011 10100
10110 01001 11001 01001 01001 10110
11010

表 25의 c_x와 表 33의 c_x'를 비교하면 다음과 같은 3개의 심볼 誤謬가 존재함을 알 수 있다.

$c_{23}=(10100)\rightarrow c_{23}'=(01100)$
$c_{22}=(00111)\rightarrow c_{22}'=(11101)$
$c_{21}=(10010)\rightarrow c_{21}'=(10001)$

誤謬 심볼은 $e_i=c_i+c_i'$으로 표시 되므로,

$e_{23}=c_{23}+c_{23}'=(11000)=\alpha^{18}$
$e_{22}=c_{22}+c_{22}'=(11010)=\alpha^{27}$
$e_{21}=c_{21}+c_{21}'=(00011)=\alpha^{21}$

가 되어, 誤謬多項式은

$$e(x)=\alpha^{21}x^{21}+\alpha^{27}x^{22}+\alpha^{18}x^{23} \quad \cdots\cdots\cdots\cdots (82)$$

으로 표현됨을 알 수 있다. 또한, 식(82)를 이용하여 오중요소는 식(83)과 같이 얻어진다.

$$\begin{aligned} s_1&=e(\alpha)=\alpha^{11}+\alpha^{18}+\alpha^{10}=\alpha^{22}=(10101)\\ s_2&=e(\alpha^2)=\alpha+\alpha^9+\alpha^2=\alpha^{13}=(00111)\\ s_3&=e(\alpha^3)=\alpha^{22}+1+\alpha^{25}=\alpha^8=(10110) \quad \cdots\cdots (83)\\ s_4&=e(\alpha)=\alpha^{12}+\alpha^{22}+\alpha^{17}=\alpha^3=(00010)\end{aligned}$$

따라서, v=2인 오중요소행렬은

$$M=\begin{bmatrix} s_1 & s_2 \\ s_2 & s_3 \end{bmatrix}=\begin{bmatrix} \alpha^{22} & \alpha^{13} \\ \alpha^{13} & \alpha^{8} \end{bmatrix} \quad \cdots (84)$$

이며, 여기서, $|M|=\alpha^5\neq 0$이 된다. 식(83)에서 구한 誤症要素를 식(64)에 대입하면, $\sigma(x)$의 계수는

$$\begin{aligned} \sigma_2&=(\alpha^{16}+\alpha^{16})/\alpha^5=0=(00000)\\ \sigma_1&=(\alpha^{21}+\alpha^{25})/\alpha^5=\alpha^{26}=(11101) \quad \cdots\cdots\cdots (85)\end{aligned}$$

와 같다. 그러므로 誤謬位置多項式 $\sigma(x)$는

$$\sigma(x)=1+\alpha^{26}x \quad \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots (86)$$

이 되며, $\sigma(x)$의 근은 α^5뿐이다. 따라서, 誤謬位置番號는

$$Z_1=1/\alpha^5=\alpha^{26}=(11101)$$

이고, 식(69)를 이용하여, 誤謬値를 계산하면

$Y_1=(s_1Z_2+s_2)/(Z_1Z_2+Z_1^2)$
$=\alpha^{13}/\alpha^{52}=\alpha^{23}=(11110)$

와 같다. 결국, $Z_2=0$이므로 誤謬多項式은

$$e(x)=\alpha^{23}x^{26} \quad \cdots\cdots (87)$$

으로 표현됨을 알 수 있다. 식(82)와 (87)을 이용하여, x^{21}, x^{22}, x^{23}, x^{26}의 위치에서, 4개의 심볼 誤謬들이 존재함을 알 수 있고, $x_i=e_i+c_i$이므로 실제로 復號化된 심볼 x_i는 다음과 같다.

$x_{21}=e_{21}+c_{21}=(00011)+(10001)=(10010)$
$x_{22}=e_{22}+c_{22}=(11010)+(11101)=(00111)$
$x_{23}=e_{23}+c_{23}=(11000)+(01100)=(10100)$
$x_{26}=e_{26}+c_{26}=(11110)+(11111)=(00001)$

그러나, 表 34에서와 같이 復號된 誤謬値는 예상과는 다른 $x_{21}=(10001)$, $x_{22}=(11101)$, $x_{23}=(01100)$, $x_{26}=(11111)$로 나타난다.

表 34. 잘못 復號된 平文 x

10110 01111 11100 10011 11111 01110
00001 01100 11101 10001 01000 00111
01001 01110 01100 10000 01111 01001
11011 11001 00000 01101 00011 10100
10110 01001 11001 01001 01001 10110
11010

表 34의 檢査심볼 (x_3, x_2, x_1, x_0)을 제거하고, x_{30}를 x_0로, x_{29}을 x_1으로 대체하여 $X=(x_0, x_1, \cdots, x_{25}, x_{26})$와 같이 역순에 의해 다시 배열하면, 최종적으로 復號화된 平文은 表 23의 밑줄친 부분과 같다.

表 35. 檢査심볼을 제거한 후 복구된 平文

10110 01111 11100 10011 11111 01110
00001 01100 11101 10001 01000 00111
01001 01110 01100 10000 01111 01001
11011 11001 00000 01101 00011 10100
10110 01001 11001

表 23의 밑줄친 내용을 表 35의 내용으로 대체하면, 表 36과 같이 복구된 형태의 ASCII 平文을 구할 수 있다.

表 36. 복구된 ASCII 平文

01010100 01101000 11100101 00100000 01100001 11101001
01101101 00100000 11101111 11100110 00100000 11100011
11110010 01111001 01110000 11110100 11101111 01100111
11110010 01111111 01110000 01011001 11011000 10100000
11101001 01110011 00100000 11110100 11101111 00100000
01101000 11101001 01100100 11100101 00100000 11110100
01101000 11100101 00100000 11100011 11101100 11100101
01100001 11110010 00100000 11100110 11101111 11110010
01101101 00100000 11101111 11100110 00100000 01110000
11101100 01100001 11101001 01101110 11110100 11100101
11111000 11110100 01100010 01111001 00100000 01101101
01100001 01101011 11101001 01101110 01100111 00100000
11101001 11110100 00100000 01110101 01101110 11110010
11100101 01100001 01100100 01100001 01100010 11101100
11100101 10101110

表 36을 表 23과 비교해 보면, 表 36이 4곳에서 잘못 復號되었다는 것을 쉽게 알 수 있고, 표 36과 일치되는 복구된 平文은 아래와 같다.

The aim of cryptogr△pYX is to hide the clear form of plaintext by making it unreadable.

결과적으로, 暗號文 歸還 暗號器을 이용하여 外部誤謬制御方式을 채택할때 전송로 상에서 발생한 二重 誤謬의 경우 DSEC (31, 27) RS符號를 사용하여 復號 과정을 거친다 할지라도 4개의 심볼 誤謬가 訂正되지 않기 때문에 誤謬制御方法으로는 부적합 하다는 사실을 알 수 있다. 반면에 內部誤謬訂正方法은 5. 2. 2절에서 논의된 바와 같이 誤謬를 制御하는데 매우 효율적이다.

5. 3. 3. 平文 歸還 暗號시스템을 위한 外部制御

平文 歸還에 의한 暗號器를 外部誤謬制御方式으

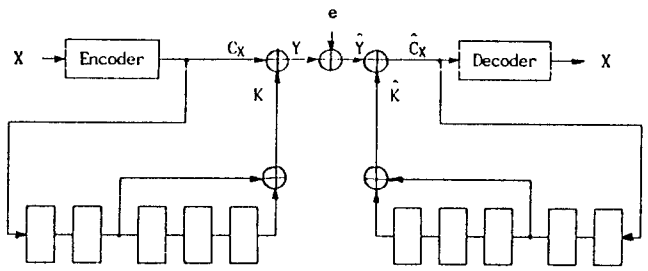


그림 26 平文 歸還 暗號器에서의 外部誤謬制御

로 구성하면 그림 26과 같다. 이 경우, 앞에서 사용했던 것과 동일한 平文과 탭 계수 T=(01001)와 초기치벡터 S=(11110)를 갖는 동일한 키 생성기를 사용하기로 하자. 符號化된 平文 c_x는 平文 c_x의 귀환과 暗號文 Y에 의해 생성된 키 數例 K로 $Y=c_x+K$에 의해 暗號化된다.

表 37. 송신자측의 키 數例 K

00100 00101 10000 11000 01011 01010
11110 01100 10101 11011 00000 01001
11101 00010 11101 01000 10011 10101
00111 00101 10001 00011 00101 11110
10001 00100 00111 10011 00011 00100
00000

符號化된 平文 c_x를 키 K로 暗號化하면 表 38과 같다.

表 38. 平文 歸還에 의한 暗號文 Y

10010 01010 01100 01011 01010 00100
11111 11000 10010 01001 01000 01110
10100 01100 10001 11000 11100 11100
11100 11100 10001 01110 00110 01010
00111 01101 11110 11010 01010 10010
11010

이제, 暗號文 Y를 전송로를 통해 전송하는 도중, $y_{24}=(11111)$와 $y_7=(01010)$ 위치에서 誤謬가 발생하여 각각 $y_{24}=(01100)$과 $y_7=(10100)$으로 변경되었다고 하자. 수신측에서의 키 생성기로부터 발생된 키 수열 K는 表 38에서 보여준 바와 같이 $K=Y+c_x$에 의해 구해진다. 물론, 키 생성기는 復號된 平文 c_x에 의해 생성된다.

表 39. 수신부에서의 키 數例

00100 00101 10000 11000 01011 01010
11011 01011 01100 01111 10000 11100
00110 00001 00001 10010 11011 11111
11010 10100 01111 01110 00001 11101
00000 11010 01010 10111 00110 01010
11000

表 40은 수신측에서 暗號文 Y와 키 數例 K의 2원합에 의해 復號된 平文 c_x 결과를 나타낸 것이다.

表 40. 수신부에서 復號된 平文 c_x

10110 01111 11100 10011 00001 01110
10111 10011 11110 00110 11000 10010
10010 01101 10000 01010 00111 00011
00110 01000 11110 00000 00111 01001
00111 10111 10100 01101 01100 11000
00010

復號된 平文 c_x는 이어서 誤謬訂正을 위한 DSEC (31, 27) RS符號의 復號器에 의해 復號가 수행된다. 수신측에서는 원래의 ASCII 平文으로 복구되기를 원하지만, 表 40의 c_x를 表 25의 c_x와 비교하면 맨 상단열을 제외하고는 상당한 차이점이 있음을 발견할 수 있다. 이러한 사실은 平文 歸還 시스템이 무한한 誤謬傳播 현상을 유발하기 때문에 平文에 대한 어떠한 형태의 外部誤謬制御 暗號시스템도 부적합하다는 것을 의미한다. 따라서, 완전한 平文을 복구시키는 시도는 무의미하지만, 그 영향을 분석하기 위해 이에 대한 復號과정을 살펴보기로 한다.

c_x와 c_x'을 이용하여, 誤謬심볼을 $e=c_x+c_x'$로 표시하면, 誤謬多項式은

$$e(x)=\alpha^{18}+\alpha^{12}x+\alpha^{7}x^{2}+\cdots+\alpha^{17}x^{22}+\alpha^{13}x^{23}+\alpha^{8}x^{24} \quad \cdots\cdots (88)$$

으로 되며, 식(88)에 x 대신 α^i, $1\leq i\leq 4$를 대입하면, 誤症要素는

$s_1=e(\alpha)=\alpha^{10}=(10001)$,
$s_2=e(\alpha^2)=\alpha^{8}=(10110)$,
$s_3=e(\alpha^3)=\alpha^{21}=(00011)$,
$s_4=e(\alpha^4)=\alpha^{25}=(11110)$.

와 같다. 또한 이 경우 誤謬위치다항식 $\sigma(x)$의 계수는 $\sigma_1=(01001)=\alpha^{30}$과 $\sigma_2=(10001)=\alpha^{10}$이 된다. 그러나 심볼 誤謬가 c_x의 서로 다른 25개의 위치에서 발생되었기 때문에, 誤謬位置番號는 무의미하다. 결과적으로 DSEC (31, 27) RS符號는 誤謬訂正없이 25개의 誤謬 심볼들을 통과 시킨다. 檢査심볼들을 제거한 후, 復號된 심볼系列은 表 41과 같다.

表 41. 檢査심볼을 제거한 후의 復號된 平文

10110 01111 11100 10011 00001 01110
10111 10011 11110 00110 11000 10010
10010 01101 10000 01010 00111 00011
00110 01000 11110 00000 00111 01001
00111 10111 10100

마지막으로 완전히 복구된 ASCII 平文 X는 表 42와 같다.

表 42. 잘못 復號된 ASCII 平文 K

01010100 01101000 11100101 00100000 01100001 11101001
01101101 00100000 11101111 11100110 00100000 11100011
11110010 01111001 01110000 11110100 11101111 01100111
11110010 01100001 01110101 11100111 11100011 01100010
01010010 01101100 00010100 01110001 10011001 00011110
00000001 11010010 01111011 11010010 11000110 01100100
11000110 00100010 11101101 11000010 10110001 01101010
11111011 10110000 10011011 11111001 11011011 01110111
00011011 00011110 10000000 00101011 00000001 00101101
01100011 11111011 10101011 11010101 11101011 11010001
01111101 10000010 01011100 00010000 00101010 10000001
00011101 10110100 01110011 00101100 11011100 00111111
11011101 01110001 01010110 01001011 00000111 11111000
00001001 00011101 10110110 01110100 10111010 00010101
01011010 10011010

表 42의 내용을 올바른 平文 X로 변환하기를 원한다 해도, 원래의 平文 X를 복구는 불가능하고, 다음과 같이 단지 쓸모없는 패키지로 만들어진다.

The aim of cryptograugcbR/q | → ┌R{RFdF"
mB1j{0[w→ +┌┌c{+UkQ}┐

전송로상에서 발생한 誤謬制御를 위해 여러가지 방식간의 비교 연구한 결과, 키 自動키 스트림 暗號器는 內部誤謬制御方式이나 外部誤謬制御方式에 공히 誤謬傳播의 영향을 받지 않는 시스템이라는 것을 알 수 있다. 또한, 暗號文 歸還 스트림 暗號器는 外部誤謬制御方法이 誤謬訂正에 다소 미흡한 점도 있으나, 次善의 시스템이 될 수 있다. 마지막으로 平文 歸還 스트림 暗號器는 무한한 誤謬傳播 특성으로 인하여 가장 부적합한 방법이라 할 수 있다.

(끝)

* 지난 號(第1卷 第2號)의 誤記를 아래와 같이 바로 잡습니다.

面 段 行	誤	正
11 左 5	B-B	B-B'
12 左 6	逐者暗號	逐字暗號
19 右 16	回期逐字暗號	同期逐字暗號
19 右 21	回期逐字暗號	同期逐字暗號
19 右 23	會期逐字暗號	同期逐字暗號
20 左 7	電送誤謬	傳送誤謬

□ 筈者紹介

李 晩 榮(正會員)

1924年 11月 30日生

서울大學校 電氣工學科 工學士(BSEE)

美國 Colorado大學校 工學碩士(MSEE) 및 工學博士(Ph.D.)

美國 Virginia州立大 工大教授

美國 California Institute of Technology, JPL 研究員

國防科學研究所 第1副所長／韓國電子通信 社長／三星半導體通信社長／漢陽大 副總長／現 漢陽大 名譽教授／韓國通信情報保護學會 會長

著書： Error Correcting Coding Theory, McGraw-Hill, New York, 1989.