[4-7] # GF(2⁸)상에서 부분체를 이용한 승산기와 정규기저표현을 이용한 승산기의 비교

박 영 옥 김 재 문 *조 용 석 이 만 영
한양대학교 전자통신공학과 *한국전기통신공사

A Comparison of finite field multiplier using subfield and normal basis representation

Park, Young Ok Kim, Jae Moon *Cho, Yong Seok Rhee, Man Young
Dep.of Electronic Comm.Engineering, Hanyang Univ. *Korea Telecommunication Authority

ABSTRACT

The algorithm for finite field operations, such as addition, mutiplication, and division is very important, because it is greatly related to the size and throughput of encoders and decoders.

In this paper, the mutipliers over GF(2^8) utilizing subfield and normal basis representation are presented and compared. In subfield multiplier, the elements of GF(2^8) are expressed in terms of elements over GF(2^4), and multiplication of those elements is primarily performed on the subfield GF(2^4). The multiplier using the normal basis can be implemented by serial-input method, which is suitable for VLSI circuits.

1. 서 론

오류정정부호이론(error correcting coding theory)과 암호이론(cryptography)은 최근들어 주목받고 있는 컴퓨터 보안, CDP나 DAT 등의 가전제품, 또는 위성통신 분야에 이용되는 중요한 학문이며, 이것은 유한체의 이론을 그 바탕으로 하고있다. 이중 오류정정부호이론의 경우 유한체 상에서 이루어지는 연산은 실제 부호기 및 복호기 설계시 전체 시스템의 규모나 성능에 절대적인 영향을 미치므로 회로경로연결(wire routing), 구조의 복잡성(complexity), 및 동시성(concurrence) 등의 문제점을 개선하기 위한 연구가 진행중이다. 특히 GF(2^m)상의 연산 알고리즘에 관한 연구가 활발히 진행되어 왔는데, 유한체상의 원소를 표현하는 방법으로기저(basis)가 사용되며 그 기저에 따라 다양한 방법의 연산이 가능하게 된다. 본 논문에서는 부분체(subfield)를 갖는 유한체의 승산을 그 부분체에서의 연산으로 처리하는 승산기와, 제곱계산이 용이한 정규기저(normal basis)상에서 치환레지스터를 이용하는 승산기를 소개하고 있다. 특히 실제 응용분야에서 가장 많이 쓰이고 있는 GF(2^8)상에서, 이 두 승산기의 승산알고리즘을 분석하고 논리회로를 제시하였으며, 사용되는 소자들의 규모와 계산시간을 검토함으로써 그 특징들을 제시, 비교하였다.

2. 부분체 GF(2^4)를 이용한 GF(2^8)상의 승산

2.1 부분체에 의한 표현

일반적으로 GF(2^m)에 있어서 m이 합성수 일 때, 즉

$$m = s_0 s_1 \cdots s_t \qquad (1)$$

로 표현될 때 GF(2^m)은 GF(2^{s_i}), $0 \le i \le t$, 를 그 부분체로 갖는다. 따라서, $8 = 4 \cdot 2 \cdot 1$ 이므로 GF(2^4)는 GF(2^8)의 부분체가 되어, GF(2^8)의 원소는 다음과 같은 GF(2^4)상의 기저로 표현될 수 있다.

$$\{ 1, \beta \} \qquad (2)$$

여기서 $\beta \in$ GF(2^8) 이며, 반드시 $\beta \notin$ GF(2^4)이어야 한다. 그러면 GF(2^8)의 임의의 한 원소 δ를

$$\delta = a_0 + a_1\beta, \quad a_0, a_1 \in GF(2^4) \qquad (3)$$

로 나타낼 수 있을 때, $\{ 1, \beta \}$가 GF(2^8)의 GF(2^4)상의 기저가 될 수 없다고 가정하면,

$$a_0 + a_1\beta = c_0 + c_1\beta, \quad a_0 \neq c_0, \; a_1 \neq c_1 \qquad (4)$$

이고 식(1.4)로부터

$$\beta = \frac{a_0 + c_0}{a_1 + c_1} \qquad (5)$$

가 된다. 식(5)에서 a_0, a_1, c_0, c_1은 모두 GF(2^4)의 원소이므로 β도 또한 GF(2^4)의 원소가 된다. 즉, $\{ 1, \beta \}$가 GF(2^8)의 기저가 되기 위해서는 β는 GF(2^4)의 원소가 아니어야만 한다.

이러한 기저 $\{ 1, \beta \}$를 구하기 위해 우선 원소들의 승산을 고려해 보자. GF(2^8)의 임의의 원소 A, B, C에 대하여 $C = A \cdot B$ 라고 하면

$$C = A \cdot B = (a_0 + a_1\beta)(b_0 + b_1\beta)$$
$$= a_0b_0 + (a_0b_1 + a_0b_1)\beta + a_1b_1\beta^2 \quad (6)$$

이 된다. 그리고 식(6)의 β^2은 다시 $\{1, \beta\}$의 기저로 표현해야 하므로

$$\beta^2 = d_0 + d_1\beta, \quad d_0, d_1 \in GF(2^4) \quad (7)$$

로 나타내어 보자. 여기서 $d_1 = 1$ 이라면 연산을 하드웨어적으로 구현하는데 있어 보다 간단할 뿐 아니라 식(7)은 선형화다항식(linearized polynomial)과 $GF(2^4)$의 원소로 이루어진 이차방정식의 형태를 갖게 되므로 β를 결정하는데 있어 용이하다.[4] 그래서, $d_1 = 1$ 로 하면 식(7)은

$$\beta^2 + \beta + d_0 = 0 \quad (8)$$

이 되고 β는 결국

$$X^2 + X + d_0 = 0, \quad d_0 \in GF(2^4) \quad (9)$$

의 근이라고 할 수 있다. 그런데, 식(9)가 $GF(2^4)$에서 근을 갖지 않고 확대체인 $GF(2^8)$에서 근을 갖기 위해서는 다음을 성립해야 한다.

$$Tr_2^4(d_0) = 1 \quad (10)$$

여기서, $Tr_q^n(x) = \sum_{i=0}^{n-1} x^{q^i}$

따라서 β는 식(8)을 만족하고, 이 때 d_0는 식(10)을 만족해야 한다.

이러한 β를 원시다항식이 $P(X) = X^8 + X^4 + X^3 + X^2 + 1$ 인 $GF(2^8)$에서 구해 보겠다. 원시원을 α라 하면

$$\alpha^8 + \alpha^4 + \alpha^3 + \alpha^2 + 1 = 0 \quad (11)$$

이다. 그리고, $\gamma^4 + \gamma^3 + 1 = 0$ 이 되고 $Tr_2^4(\gamma) = 1$ 이 되는 $GF(2^8)$의 임의의 한 원소 γ를 찾으면

$$\gamma^4 + \gamma^3 + 1 = (\alpha^{119})^4 + (\alpha^{119})^3 + 1 = 0 \quad (12)$$

$$Tr_2^4(\alpha^{119})$$
$$= (\alpha^{119})^{2^0} + (\alpha^{119})^{2^1} + (\alpha^{119})^{2^2} + (\alpha^{119})^{2^3}$$
$$= 1 \quad (13)$$

으로부터

$$\gamma = \alpha^{119} \quad (14)$$

이 되어, γ는 원시다항식이 $P(X) = X^4 + X^3 + 1$ 인 $GF(2^4)$의 한 원소이면서 식(10)을 만족한다. 따라서,

$$\beta^2 + \beta + \alpha^{119} = 0 \quad (15)$$

을 만족하는 β는, $\beta = \alpha^7$ 이 되어 $GF(2^8)$의 $GF(2^4)$상의 기저는

$$\{1, \alpha^7\} \quad (16)$$

이 된다.

2.2 부분체를 이용한 승산

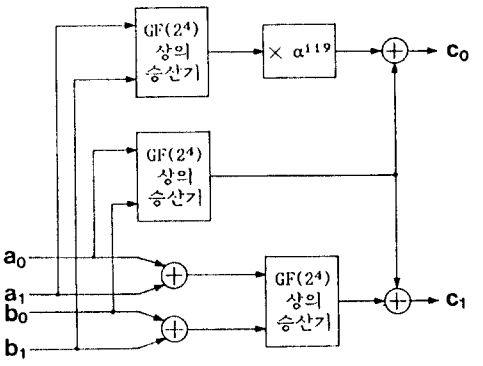
식(6)과 (15)로 부터

$$C = c_0 + c_1\beta$$
$$= A \cdot B = (a_0 + a_1\beta)(b_0 + b_1\beta)$$
$$= (a_0b_0 + a_1b_1\alpha^{119}) + (a_0b_1 + a_0b_1 + a_1b_1)\beta \quad (17)$$

이다. 그리고 $GF(2^8)$의 임의의 한 원소 $A = a_0 + a_1\beta$ 는 a_0과 a_1가 $GF(2^4)$의 원소이므로

$$A = a_0 + a_1\beta$$
$$= (x_0 + x_1\gamma + x_2\gamma^2 + x_3\gamma^3) + (y_0 + y_1\gamma + y_2\gamma^2 + y_3\gamma^3),$$
$$\begin{cases} x_i, y_i \in GF(2), \ 0 \le i \le 3 \\ \gamma \text{는 } P(X) = X^4 + X^3 + 1 \text{ 의 근} \end{cases} \quad (18)$$

로 표현할 수 있다. 결국 식(17)에서 $c_0 = a_0b_0 + a_1b_1\alpha^{119}$, $c_1 = a_0b_1 + a_0b_1 + a_1b_1$ 은 모두 $GF(2^4)$상의 계산으로 구할 수 있으며, 결국 부분체표현을 통해서 $GF(2^8)$의 승산을 $GF(2^4)$상의 연산으로 실행할 수 있다.



그림<1> 부분체를 이용한 $GF(2^8)$에서의 승산기

그림<1>에서 'GF(2⁴)상의 승산기'의 연산은

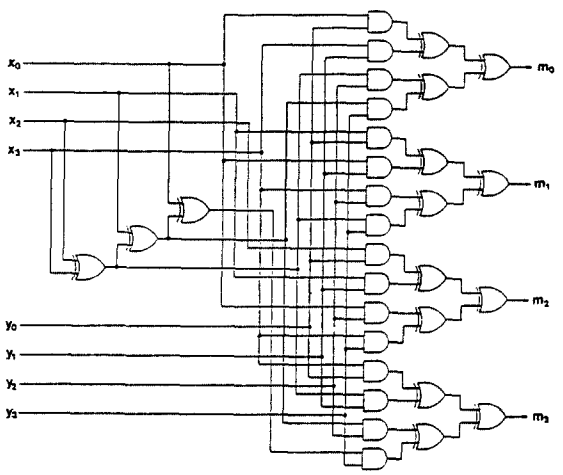
$$m_0 + m_1\gamma + m_2\gamma^2 + m_3\gamma^3$$
$$= (x_0 + x_1\gamma + x_2\gamma^2 + x_3\gamma^3)(y_0 + y_1\gamma + y_2\gamma^2 + y_3\gamma^3) \quad (19)$$

$$\begin{cases} m_0 = x_0y_0 + x_3y_1 + (x_2 + x_3)y_2 \\ \qquad + (x_1 + x_2 + x_3)y_3 \\ m_1 = x_1y_0 + x_0y_1 + x_3y_2 + (x_2 + x_3)y_3 \\ m_2 = x_2y_0 + x_1y_1 + x_0y_2 + x_3y_3 \\ m_3 = x_3y_0 + (x_2 + x_3)y_1 + (x_1 + x_2 + x_3)y_2 \\ \qquad + (x_0 + x_1 + x_2 + x_3)y_3 \end{cases}$$

이며 이 연산은 그림<2>의 논리회로로 구성할 수 있다. 그리고, 그림<1>의 α^{119} 승산은 $\gamma = \alpha^{119}$ 이므로, $X = x_0 + x_1\gamma + x_2\gamma^2 + x_3\gamma^3$ 이라고 할 때

$$\begin{aligned} \gamma \cdot X &= x_0\gamma + x_1\gamma^2 + x_2\gamma^3 + x_3\gamma^4 \\ &= x_3 + x_0\gamma + x_1\gamma^2 + (x_2 + x_3)\gamma^3 \end{aligned} \qquad (20)$$

이 되므로 EX-OR 게이트 1개로 구현할 수 있다.



그림<2> GF(2^4)상의 승산기

3. 정규기저표현에 의한 GF(2^8)상의 승산

일반적으로 GF(2^m)의 한 원소를 β라 하고 GF(2)상의 정규기저를

$$\{ \beta^{2^0}, \beta^{2^1}, \beta^{2^2}, \cdots, \beta^{2^{m-1}} \} \qquad (21)$$

라 하면, 이들은 선형독립(linearly independence)을 이루므로

$$\beta^{2^0} + \beta^{2^1} + \beta^{2^2} + \cdots + \beta^{2^{m-1}} = Tr_2^m(\beta) = 1 \qquad (22)$$

이 된다. 원시다항식이 $P(X) = X^8 + X^4 + X^3 + X^2 + 1$ 인 GF(2^8)의 원시원을 α라 하면 $\alpha^8 + \alpha^4 + \alpha^3 + \alpha^2 + 1 = 0$ 이 되며, $Tr_2^8(\beta) = 1$ 이 되는 GF(2^8)의 한 원소 β를 찾으면

$$\begin{aligned} Tr_2^8(\alpha^5) &= \sum_{i=0}^{7} (\alpha^5)^{2^i} \\ &= \alpha^5 + \alpha^{10} + \alpha^{20} + \alpha^{40} + \alpha^{80} + \alpha^{160} + \alpha^{65} + \alpha^{130} \\ &= 1 \end{aligned} \qquad (23)$$

이므로, $\beta = \alpha^5$ 이 된다. 따라서 GF(2^8)은

$$\begin{aligned} &\{ \beta^{2^0}, \beta^{2^1}, \beta^{2^2}, \beta^{2^3}, \beta^{2^4}, \beta^{2^5}, \beta^{2^6}, \beta^{2^7} \} \\ &= \{ \alpha^5, \alpha^{10}, \alpha^{20}, \alpha^{40}, \alpha^{80}, \alpha^{160}, \alpha^{65}, \alpha^{130} \} \end{aligned} \qquad (24)$$

을 정규기저로 갖는다. 그러므로 GF(2^8)의 임의의 원소 A는

$$A = a_0\beta^{2^0} + a_1\beta^{2^1} + a_2\beta^{2^2} + \cdots + a_7\beta^{2^7} \qquad (25)$$

으로 표현되며, A^2은 A를 오른쪽으로 한번 순회치환한

$$A^2 = a_7\beta^{2^0} + a_0\beta^{2^1} + a_1\beta^{2^2} + \cdots + a_6\beta^{2^7}$$

이 된다.

GF(2^8)의 임의의 두 원소를 정규기저표현으로 나타내어

$$A = a_0\beta^{2^0} + a_1\beta^{2^1} + a_2\beta^{2^2} + \cdots + a_7\beta^{2^7}$$
$$B = b_0\beta^{2^0} + b_1\beta^{2^1} + b_2\beta^{2^2} + \cdots + b_7\beta^{2^7}$$

이라 하고, $C = A \cdot B$ 라고 하자. 그리고, 정규기저를 이용한 승산법을 제시하기 위해 다음을 정의한다.

$$\begin{aligned} A_i &= a_{7-i}\beta^{2^0} + a_{7-(i-1)}\beta^{2^1} + \cdots + a_6\beta^{2^{i-1}} + a_7\beta^{2^i} \\ B_i &= b_{7-i}\beta^{2^0} + b_{7-(i-1)}\beta^{2^1} + \cdots + b_6\beta^{2^{i-1}} + b_7\beta^{2^i} \\ C_i &= A_i \cdot B_i \\ &\qquad \text{for } 0 \le i \le 7 \end{aligned} \qquad (26)$$

그러면 $B_i = B_{i-1}^2 + b_{7-i}\beta$, $A_i = A_{i-1}^2 + a_{7-i}\beta$ 가 되어 B_i는 B_{i-1}을 오른쪽으로 한번치환한 뒤, β^{2^0} 위치에 b_{7-i}를 첨가시키는 것과 같으며, 이는 치환레지스터를 이용해 구현할 수 있다. 식(26)으로부터

$$\begin{aligned} C_0 &= a_7b_7\beta^2 \\ C_i &= A_i \cdot B_i = (A_{i-1}^2 + a_{7-i}\beta)(B_{i-1}^2 + b_{7-i}\beta) \\ &= C_{i-1}^2 + (A_{i-1}^2 \cdot b_{7-i} + B_{i-1}^2 \cdot a_{7-i})\beta \\ &\quad + a_{7-i}b_{7-i}\beta^2 \end{aligned} \qquad (27)$$

이 되며, $i = 7$ 이 되었을 때 $A_7 = A$, $B_7 = B$, $C_7 = C$ 이 되어 승산이 완결된다. 식(27)에서 C_i를 구하기 위해서는 $\beta \cdot (A_{i-1}^2 \cdot b_{7-i} + B_{i-1}^2 \cdot a_{7-i})$의 계산이 필요하며 이는 GF($2^8$)의 임의의 한 원소 D에 대해 $\beta \cdot D$ 를 실현하는 방법을 제시함으로써 가능하다.

$$D = d_0\beta^{2^0} + d_1\beta^{2^1} + d_2\beta^{2^2} + \cdots + d_7\beta^{2^7}$$

라 하면 $\beta = \alpha^5$ 이므로 $\beta \cdot D$ 는 다음과 같이 이루어 진다.

$$\begin{aligned}
\beta \cdot D &= f_0\beta^{2^0} + f_1\beta^{2^1} + f_2\beta^{2^2} + \cdots + f_7\beta^{2^7} \\
&= \alpha^5 \cdot (d_0\alpha^5 + d_1\alpha^{10} + d_2\alpha^{20} + d_3\alpha^{40} \\
&\quad + d_4\alpha^{80} + d_5\alpha^{160} + d_6\alpha^{65} + d_7\alpha^{130}) \\
&= d_0\alpha^{10} + d_1\alpha^{15} + d_2\alpha^{25} + d_3\alpha^{45} \\
&\quad + d_4\alpha^{85} + d_5\alpha^{165} + d_6\alpha^{70} + d_7\alpha^{135} \\
&= d_0\alpha^{10} + d_1(\alpha^{10} + \alpha^{20} + \alpha^{160}) \\
&\quad + d_2(\alpha^5 + \alpha^{20} + \alpha^{40} + \alpha^{80}) \\
&\quad + d_3(\alpha^5 + \alpha^{10} + \alpha^{80} + \alpha^{160} + \alpha^{65} + \alpha^{130}) \\
&\quad + d_4(\alpha^{10} + \alpha^{40} + \alpha^{160} + \alpha^{130}) \\
&\quad + d_5(\alpha^5 + \alpha^{10} + \alpha^{20} + \alpha^{40} + \alpha^{80} + \alpha^{160}) \\
&\quad + d_6(\alpha^5 + \alpha^{10} + \alpha^{20} + \alpha^{65}) \\
&\quad + d_7(\alpha^5 + \alpha^{10} + \alpha^{80}) \\
&= (d_2 + d_3 + d_5 + d_6 + d_7)\alpha^5 \\
&\quad + (d_0 + d_1 + d_4 + d_5 + d_6 + d_7)\alpha^{10} \\
&\quad + (d_1 + d_2 + d_5 + d_6)\alpha^{20} \\
&\quad + (d_2 + d_3 + d_4 + d_5)\alpha^{40} \qquad (28) \\
&\quad + (d_2 + d_3 + d_5 + d_6)\alpha^{80} \\
&\quad + (d_1 + d_3 + d_4 + d_5)\alpha^{160} \\
&\quad + (d_3 + d_6)\alpha^{65} \\
&\quad + (d_3 + d_4)\alpha^{130}
\end{aligned}$$

식(28)은 EX-OR 게이트를 이용한 논리회로로 구현한다. 그림<3>은 레지스터와 논리게이트로 식(27)을 이용하여 직렬입력-병렬출력 승산기를 나타낸 것이다. A와 B는 각각 a_7과 b_7 부터 한 클럭에 한 비트씩 입력되며 8클럭 후에 $C = A \cdot B$가 완성된다. 식(28)의 계산은 그림<3>의 중간부분에서 이루어지는데, $A_{i-1}{}^{2} \cdot b_{7-i} + B_{i-1}{}^{2} \cdot a_{7-i} = d_0\beta^{2^0} + d_1\beta^{2^1} + d_2\beta^{2^2} + \cdots + d_7\beta^{2^7}$ 이라고 하면, d_0는 항상 "0"이 되므로 그림<3>에서 d_0는 생략되었다.

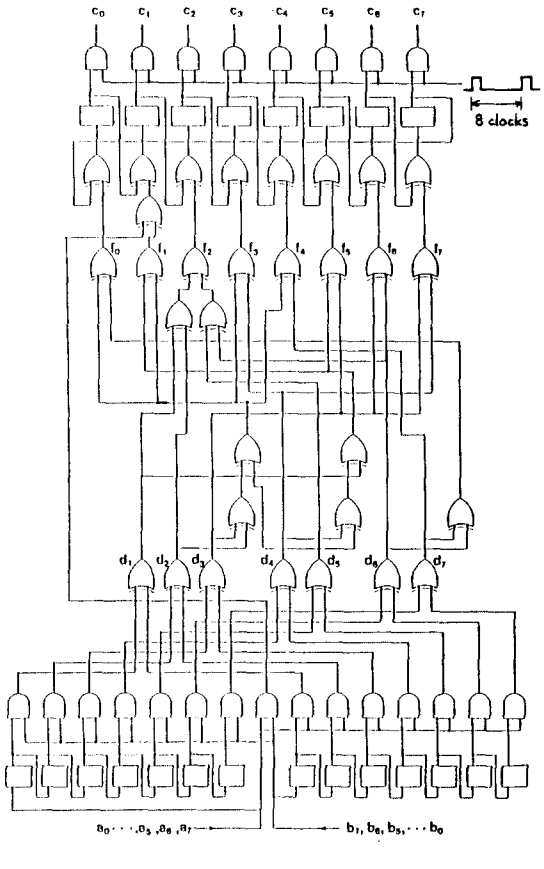
4. 비교·검토

그림<1>에서 주어진 승산기는 $GF(2^8)$의 원소들을 $GF(2^4)$상의 기저로 표현함으로써 $GF(2^8)$상의 승산을 $GF(2^4)$상의 연산으로 구할 수 있다. 이것을 위해 먼저, GF(2)상의 표준기저(standard basis)로 표현된 원소를 { 1, β }를 이용한 표현으로 변환하는 과정이 필요하다. $GF(2^4)$의 GF(2)상의 표준기저는

$$\{\ 1,\ \gamma,\ \gamma^2,\ \gamma^3\ \} \qquad (29)$$

이며 $GF(2^8)$의 $GF(2^4)$상의 기저는 { 1, β } 이므로, $GF(2^8)$의 GF(2)상의 기저는

$$\begin{aligned}
&\{\ 1,\ \gamma,\ \gamma^2,\ \gamma^3,\ \beta,\ \beta\gamma,\ \beta\gamma^2,\ \beta\gamma^3\ \} \\
&= \{\ 1,\ \alpha^{119},\ \alpha^{238},\ \alpha^{102},\ \alpha^{7},\ \alpha^{126},\ \alpha^{245},\ \alpha^{109}\ \}
\end{aligned} \qquad (30)$$



그림<3> 정규기저표현에 의한 $GF(2^8)$상의 직렬입력-병렬출력 승산기

이 된다. 그래서, 표준기저로 표현된 $GF(2^8)$의 한 원소는 다음식을 통해 변환된다.

$$\begin{aligned}
&a_0 + a_1\alpha + a_2\alpha^2 + a_3\alpha^3 \\
&\quad + a_4\alpha^4 + a_5\alpha^5 + a_6\alpha^6 + a_7\alpha^7 \\
&= (a_0 + a_1 + a_5) + (a_1 + a_4 + a_5)\alpha^{119} \\
&\quad + (a_2 + a_3 + a_6)\alpha^{238} + (a_1 + a_3 + a_4 + a_6)\alpha^{102} \\
&\quad + (a_1 + a_2 + a_4 + a_5 + a_6 + a_7)\alpha^{7} \\
&\quad + (a_2 + a_5 + a_6)\alpha^{126} \\
&\quad + (a_1 + a_2 + a_3 + a_4 + a_5 + a_6)\alpha^{245} \\
&\quad + (a_1 + a_3 + a_4 + a_5)\alpha^{109}
\end{aligned} \qquad (31)$$

또한, 그 역변환은

$$\begin{aligned}
&b_0 + b_1\alpha^{119} + b_2\alpha^{238} + b_3\alpha^{102} \\
&\quad + b_4\alpha^{7} + b_5\alpha^{126} + b_6\alpha^{245} + b_7\alpha^{109} \\
&= (b_0 + b_1 + b_2 + b_6 + b_7) + (b_1 + b_2 + b_5)\alpha
\end{aligned}$$

$$+ (b_3 + b_5 + b_7)\alpha^2 + (b_2 + b_6 + b_7)\alpha^3$$
$$+ (b_1 + b_7)\alpha^4 + (b_5 + b_6 + b_7)\alpha^5$$
$$+ (b_3 + b_5 + b_6)\alpha^6 + (b_1 + b_4 + b_6 + b_7)\alpha^7 \quad (32)$$

으로 이루어진다. 식(31)의 변환회로와 식(32)의 역변환회로는 각각 13개의 EX-OR 게이트로 구현할 수 있다.

이렇게 변환된 GF(2^8)의 원소는 그림<1>의 승산기에서 연산된다. 이 승산기는 EX-OR 게이트 65개, AND 게이트 45개로 구성되며, 표<1>에서 보다 상세히 보여주고 있다. 이것은 GF(2^8)에 대하여 식(19)와 같은 형태의 연산을 했을 때 EX-OR 게이트 73개, AND 게이트 64개가 소요되는 데에 비교하여 크게 축소된 것이며, 논리게이트 만으로 구성함으로써 지연없이 한 클럭 내에 처리될 수 있다.

	GF(2^4) 승산	$\times\alpha^{110}$	modulo 2 연산	총
EX-OR	16 × 3 = 48	1	4 × 4 = 16	65
AND	15 × 3 = 45			45

표<1> GF(2^4)를 이용한 GF(2^8)의 승산기에 사용되는 게이트 수

그림<3>의 승산기는 치환레지스터와 논리게이트로 구성하였으며, CP(clock pulse)와 레지스터의 클리어(clear)입력은 그림상에서 생략했다.

소 자	갯수
EX- R	31
AND	23
D flip-flop	22

표<2> 정규기저표현을 이용한 승산기에 사용되는 소자 수

사용되는 소자의 수는 표<2>에 나타나 있는데, 이는 그림<1>의 승산기에 비해 적은 수이지만 앞의 승산기가 한 클럭 안에 승산을 끝낼 수 있는 것에 반하여, 레지스터의 사용으로 8클럭 후에야 승산이 완결되므로 속도면에서 불리함이 있다. 그렇지만, 입력은 직렬로, 출력은 병렬로 이루어지기 때문에 일반적인 병렬연산에서 필요한 S/P 전환이 필요없으므로, 정규기저표현으로 연산하는 승산기, 제산기, 가산기 등이 연결된 회로에서 제일 첫 단의 승산기로 사용되면 유리하다.

5. 결 론

본 논문에서는 GF(2^8)상에서의 두가지 승산법을 제시, 비교하였다. 부분체를 이용한 승산기는 GF(2^8)이 GF(2^4)을 부분체로 가지며, GF(2^8)의 모든 원소를 이 부분체 GF(2^4) 상의 기저로 표현될 수 있다는 것을 이용하여 GF(2^8)의 승산을 GF(2^4)상의 연산으로 처리함으로써 승산기 구성에 사용되는 논리게이트의 수를 크게 줄일 수 있었고, 논리게이트 만으로 승산기를 구성하여 한 클럭에 승산이 처리되는 장점이 있다. 정규기저를 이용한 승산기는 직렬입력방법과 치환레지스터를 이용하여 VLSI가 가능한 작은 규모로 구현할 수 있으며 8 클럭만에 승산이 이루어진다. 따라서, 부분체를 이용한 승산기는 정규기저를 이용한 승산기에 비해 규모는 크지만 지연이 없다는 이점을 가지며 정규기저를 이용한 승산기는 8 클럭의 지연이 필요한 반면 VLSI에 적합하며 직렬입력의 특성을 이용할 수 있다.

[References]

(1) M. Y. Rhee, *Error Correcting Coding Theory*, McGraw-Hill, New York, 1989.

(2) F. J. MeWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland, Amsterdam, The Netherland, 1977.

(3) R. J. McEliece, *Finite Fields for Computer Scientists and Engineers*, Kluwer Academic Publishers, 1987.

(4) E. R. Berlekamp, *Algebraic Coding Theory*, McGraw-Hill, New York, 1968.

(5) M. Morii and M. Kasahara, "Efficient Construction of Gate Circuit for Computing Multiplicative Inverses over GF(2^m)", *Trans. IEICE*, Japan, vol. E-72, no.1, Jan. 1988.

(6) G. L. Feng, "A VLSI Architecture for Fast Inversion in GF(2^m)", *IEEE Trans. Comput.*, vol.38, no.10, Oct. 1989.