유한체 GF(2^8) 상의 직/병렬 혼합 승산기 설계

°조 용 석*, 박 상 규**

*영동공과대학교 전자공학부, **한양대학교 전자통신공학과

A Design of Serial-Parallel Multiplier over GF(2^8)

°Yong Suk Cho*, Sang Kyu Park**

*Faculty of Electronics Eng., Youngdong Institute of Technology

**Dept. of Electronic Communication Eng., Hanyang University

1. 서 론

유한체(finite field or Galois field) 상의 연산은 오류정정 부호이론 및 암호이론 등의 분야에서 중심적인 역할을 하는 것으로, 이들 분야에서 널리 응용되고 있다. 특히 오류정정부호 중 BCH 부호나 Reed-Solomon 부호와 같은 블럭부호에서는 모든 연산이 유한체 GF(2^m) 상에서 이루어진다. 따라서 유한체 GF(2^m) 상의 연산을 얼마만큼 효율적으로 수행할 것인가 하는 문제는 부호화 및 복호화의 속도와 부/복호기의 하드웨어 량에 직접적으로 영향을 미치는 매우 기본적이고 중요한 요소이다.

유한체 GF(2^m)은 2^m개의 원소(elements)를 가지고 있으며, 이들 각 원소들은 다항식표현(polynomial representation)과 지수표현(exponential representation)의 2가지 방법으로 표현할 수 있다[1]. 지수표현을 이용하면 곱셈과 나눗셈은 각각 2진수의 덧셈과 뺄셈으로 대치되므로 쉽게 수행할 수 있는 반면에 덧셈이 복잡해지며, 다항식표현을 이용하면 덧셈은 각 비트 별 2원합(modulo-2 sum)으로 간단하게 수행되지만 곱셈과 나눗셈이 어려워지는 문제점이 있다. GF(2^m)의 위수(order) m이 작은 경우에는 지수표현을 이용한 연산이 더 간단한 반면, m이 커지면 지수표현 보다는 다항식표현을 이용한 연산이 더 적은 하드웨어로 구현할 수 있으며 고속처리가 가능하다.

따라서 다항식표현을 이용하여 곱셈과 나눗셈을 효율적으로 실행하기 위한 방법들이 집중적으로 연구되고 있다. 대표적인 것으로, 기존에 사용되던 표준기저(standard basis) 대신에 쌍대기저(dual basis)를 사용한 Berlekamp[2]의 곱셈 알고리즘과, 정규기저(normal basis)를 사용한 Massey와 Omura[3]의 곱셈 알고리즘을 들 수 있다. 이 알고리즘들은 다항식 기저를 적절히 변환하여 소요되는 하드웨어 및 지연시간을 줄이고자 하는 방법들로, 이들의 개선에 관한 많은 연구들이 발표되고 있다[4]-[7].

유한체 GF(2^m) 상의 승산기는 조합회로를 사용한 병렬 승산기와 순서회로를 사용한 직렬 승산기로 구현할 수 있다. 병렬 승산기는 연산속도는 빠른 반면에 회로가 복잡해지며, 직렬 승산기는 회로는 간단하지만 m 클럭 시간의 지연이 불가피 해진다.

본 논문에서는 이러한 문제점을 해결할 수 있는 한 가지 방법으로, 직렬 승산기와 병렬 승산기를 혼합한 직/병렬 승산기를 제안한다. 본 방법은 유한체 GF(2^m)의 부분체(subfield) 상의 병렬 승산기를 이용하여 유한체 GF(2^m) 상의 직렬 승산기를 구현하는 것이다. 이는 회로의 복잡도와 지연시간 사이의 적절한 절충을 꾀하는 것이다.

유한체 GF(2^8) 상의 승산기는 CD나 DAT와 같은 정보저장 시스팀과 Voyager 등의 위성통신 시스템, 최근 방송을 시작한 디지털 TV 등에서 사용되고 있는 실용상 매우 중요한 것이다. 본 논문에서는 유한체 GF(2^2) 상의 병렬 승산기를 사용하여 유한체 GF(2^8) 상의 승산기를 설계한다.

먼저 2장에서 GF(2^2) 상에서 직렬 승산기와 병렬 승산기를 설계한다. 3장에서는 유한체 GF(2^8)을, 그것의 부분체인 GF(2^2)로 표현하고 2장에서의 논의를 기초로 GF(2^8) 상의 직/병렬 혼합 승산기를 설계한다. 끝으로 4장에서 결론을 맺는다.

2. 유한체 GF(2^2) 상의 승산기

α를 유한체 GF(2^m)의 원시원(primitive element)이라 할 때 영원(zero element)을 제외한 2^m-1개의 모든 원소들은 α의 멱(power)으로 표현할 수 있다[1]. 또 이 α가 차수가 m인 원시다항식(primitive polynomial)

$$f(x) = 1 + f_1 x + \cdots\cdots + f_{m-1}x^{m-1} + x^m, \quad f_i \in \mathrm{GF}(2),\ 1 \le i \le m-1 \qquad (1)$$

의 근(root)이라고 하면, $f(\alpha) = 0$ 이므로

$$\alpha^m = 1 + f_1\alpha + \cdots\cdots + f_{m-2}\alpha^{m-2} + f_{m-1}\alpha^{m-1} \qquad (2)$$

가 되어 유한체 GF(2^m)의 각 원소들은 m-1차 이하인 α의 다항식으로 표현할 수 있다. 이와 같은 표현방법을 다항식표현이라고 하며 이 다항식들의 계수만으로 나타내는 방법을 벡터표현이라고 한다. 지수표현은 α의 멱을 2진수로 나타내는 것이다. 여기에서

$$\{1,\ \alpha,\ \alpha^2,\ \cdots\cdots,\ \alpha^{m-2},\ \alpha^{m-1}\} \qquad (3)$$

를 유한체 GF(2^m)의 표준기저라고 한다.

예를 들어 유한체 GF(2^2)의 원시원을 β라 하고, 원시다항식을

$$f(x) = 1 + x + x^2 \qquad (4)$$

라 하면 $f(\beta) = 0$ 이므로 다음과 같이 된다.

$$\beta^2 = 1 + \beta \qquad (5)$$

따라서 유한체 GF(2^2)의 원소들을 다항식표현과 지수표현으로 나타내면 표 1과 같이 된다.

멱표현	다항식표현	벡터표현	지수표현
$\beta^{'}$	β^{0} β^{1}	β^{0} β^{1}	2 1
0	0	0 0	1 1
β^{0}	1	1 0	0 0
β^{1}	β	0 1	0 1
β^{2}	1 + β	1 1	1 0

표 1 $f(x) = 1 + x + x^2$ 일 때 GF(2^2)의 원소들의 표현

그러므로 유한체 GF(2^2)의 0이 아닌 임의의 두 원소 A와 B는

$$A = a_0 + a_1\beta = \sum_{i=0}^{1} a_i\beta^i, \quad a_i \in GF(2) \tag{6}$$

$$B = b_0 + b_1\beta = \sum_{i=0}^{1} b_i\beta^i, \quad b_i \in GF(2) \tag{7}$$

로 표현할 수 있으며, 이 두 원소의 곱 Z는 다음과 같이 쓸 수 있다.

$$\begin{aligned} Z &= A \cdot B = A \cdot \left(\sum_{i=0}^{1} b_i\beta^i \right) = \sum_{i=0}^{1} b_i(A\beta^i) \\ &= b_0A + b_1(A\beta) \end{aligned} \tag{8}$$

식 (8)을 이용하면 유한체 GF(2^2) 상의 승산기를 구성할 수 있다.

먼저 임의의 한 원소 A에 원시원 β를 곱하는 회로를 구성해 보자. 식 (6)과 같은 GF(2^2)의 임의의 한 원소 A에 β를 곱하면 다음과 같이 된다.

$$\begin{aligned} A\beta &= (a_0 + a_1\beta)\beta = a_0\beta + a_1\beta^2 = a_0\beta + a_1(\beta + 1) \\ &= a_1 + (a_0 + a_1)\beta \end{aligned} \tag{9}$$

식 (9)를 조합회로로 구성하면 그림 1의 (a)와 같이 되며, 순서회로로 구현하면 그림 1의 (b)와 같이 된다. 그림 1에서 ⊕는 GF(2) 상의 덧셈기로 1개의 2입력 Exclusive-OR(이하 XOR)로 구현할 수 있다.

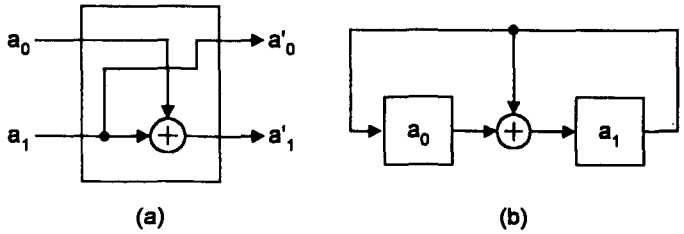

그림 1 GF(2^2) 상의 임의의 한 원소에 β를 곱하는 회로

식 (8)과 그림 1의 (a)를 이용하면 그림 2와 같은 병렬 승산기를 구성할 수 있다. 그림 2에서 ⊠는 GF(2) 상의 승산기로 1개의 2입력 AND 게이트로 구현할 수 있다. 따라서 $GF(2^2)$ 상의 병렬 승산기는 2입력 AND 게이트 4개와 2입력 XOR 게이트 3개로 구현할 수 있다.

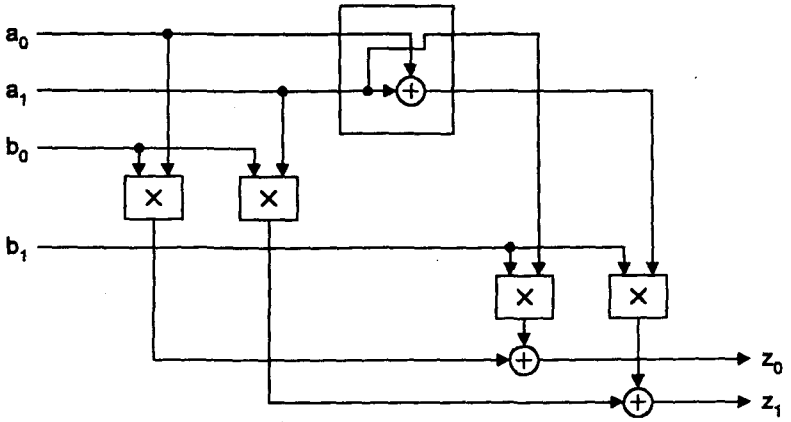

그림 2 $GF(2^2)$상의 병렬 승산기

또 식 (8)과 그림 1의 (b)를 이용하면 그림 3과 같은 직렬 승산기를 구성할 수 있다. 그림 3에서 초기에 곱하고자 하는 두 수를 레지스터 A와 B에 로드하고 레지스터 Z는 클리어 시킨 다음, 클럭을 한 번 인가하면 레지스터 Z에는 b_0A가 나타나고, 두 번 인가하면 $b_0A + b_1(A\beta)$가 나타나므로, 두 번 인가한 다음의 레지스터 Z의 값이 식 (8)과 같은 두 수를 곱한 결과가 된다.

그림 2와 그림 3의 승산기를 비교하면 그림 2와 같이 조합회로로 구현한 병렬 승산기는 지연시간이 짧은 반면에 m이 커짐에 따라 회로가 급격히 복잡해지며, 그림 3과 같은 직렬 승산기는 m이 커져도 회로가 비교적 간단한 반면 m클럭 시간의 지연 후에 결과가 나오게 된다.

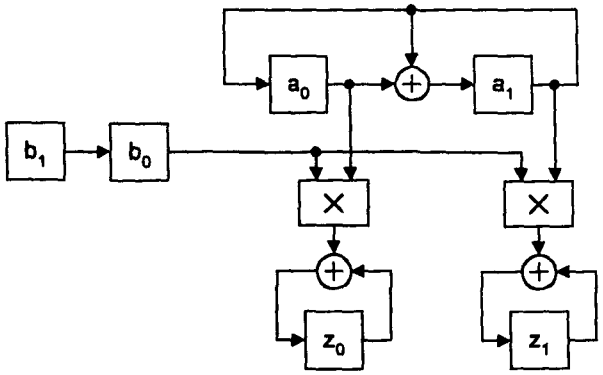

그림 3 $GF(2^2)$ 상의 직렬 승산기

3. 유한체 GF(2^8) 상의 직/병렬 혼합 승산기 설계

유한체 GF(2^8)의 원시원을 α라 하고, 원시다항식을

$$f(x) = 1 + x^2 + x^3 + x^4 + x^8 \tag{10}$$

이라 하면, $f(\alpha) = 0$ 이므로 다음과 같이 된다.

$$\alpha^8 = 1 + \alpha^2 + \alpha^3 + \alpha^4 \tag{11}$$

따라서 유한체 GF(2^8)의 0이 아닌 255개의 원소는, 다음과 같이 7차 이하인 α의 다항식으로 나타낼 수 있다.

$$\begin{aligned}
\alpha^0 &= 1 \\
\alpha^1 &= \alpha \\
&\vdots \\
\alpha^8 &= 1 + \alpha^2 + \alpha^3 + \alpha^4 \\
\alpha^9 &= \alpha + \alpha^3 + \alpha^4 + \alpha^5 \\
&\vdots \\
\alpha^{253} &= 1 + \alpha + \alpha^2 + \alpha^6 \\
\alpha^{254} &= \alpha + \alpha^2 + \alpha^3 + \alpha^7
\end{aligned} \tag{12}$$

즉 유한체 GF(2^8)의 임의의 한 원소 U는 다음과 나타낼 수 있다.

$$U = u_0 + u_1\alpha + u_2\alpha^2 + u_3\alpha^3 + u_4\alpha^4 + u_5\alpha^5 + u_6\alpha^6 + u_7\alpha^7, \quad u_i \in \mathrm{GF}(2) \tag{13}$$

GF(2^8)의 원시원 α와 GF(2^8)의 부분체인 GF(2^2)의 원시원 β는

$$\beta^3 = 1 = \alpha^{255} \tag{14}$$

과 같은 관계가 성립하므로 β를 GF(2^8) 상의 원소로 나타내면 다음과 같이 된다.

$$\beta = \alpha^{85} \tag{15}$$

따라서 유한체 GF(2^8)의 부분체인 GF(2^2)의 0이 아닌 3개의 원소는 $\beta^0 = \alpha^0$, $\beta = \alpha^{85}$, $\beta^2 = \alpha^{170}$이 된다. 식 (15)는 일반적으로는 $\beta = \alpha^{85k}$이며, 여기에서는 $k = 1$을 선택한 것이다. 여기에서 k는 255와 서로 소(relative prime)인 임의의 정수이다.

유한체 GF(2^8)은 GF(2^2) 상의 4차 원시다항식을 이용하여 구성할 수 있다. α가 $f(x)$의 근이므로 α의 공액(conjugate)인 $\alpha^{2^{m-1}}$, 즉 α^2, α^4, α^8, α^{16}, α^{32}, α^{64}, α^{128}도 $f(x)$의 근이 된다. 따라서 4차의 원시다항식 $f(x)$는 다음과 같이 쓸 수 있다.

$$f(x) = (x + \alpha)(x + \alpha^4)(x + \alpha^{16})(x + \alpha^{64}) \tag{16}$$

식 (12)를 이용하여 식 (16)을 풀어서 정리하면 다음과 같이 된다.

$$f(x) = \alpha^{85} + \alpha^{85}x + \alpha^{85}x^2 + x^3 + x^4 \tag{17}$$

식 (15)에 따라 $\alpha^{85} = \beta$ 이므로 $f(x)$를 GF(2^2) 상의 다항식으로 표현하면

$$f(x) = \beta + \beta x + \beta x^2 + x^3 + x^4 \tag{18}$$

가 된다. 또 $f(\alpha) = 0$ 이므로 다음과 같이 쓸 수 있다.

$$\alpha^4 = \beta + \beta\alpha + \beta\alpha^2 + \alpha^3 \tag{19}$$

따라서 식 (19)와 표 1을 이용하면 유한체 GF(2^8)의 0이 아닌 모든 원소는, 다음과 같이 GF(2^2) 상의 3차 이하인 α의 다항식으로 표현할 수 있다.

$$\begin{aligned}
\alpha^0 &= 1 \\
\alpha^1 &= \alpha \\
&\vdots \\
\alpha^4 &= \beta + \beta\alpha + \beta\alpha^2 + \alpha^3 \\
\alpha^5 &= (\beta + \beta\alpha + \beta\alpha^2 + \alpha^3)\alpha = \beta\alpha + \beta\alpha^2 + \beta\alpha^3 + \alpha^4 \\
&= \beta\alpha + \beta\alpha^2 + \beta\alpha^3 + (\beta + \beta\alpha + \beta\alpha^2 + \alpha^3) \\
&= \beta + (\beta + 1)\alpha^3 = \beta + \beta^2\alpha^3 \\
\alpha^6 &= (\beta + \beta^2\alpha^3)\alpha = \beta\alpha + \beta^2\alpha^4 = \beta\alpha + \beta^2(\beta + \beta\alpha + \beta\alpha^2 + \alpha^3) \\
&= \beta\alpha + \beta^3 + \beta^3\alpha + \beta^3\alpha^2 + \beta^2\alpha^3 = \beta\alpha + 1 + \alpha + \alpha^2 + \beta^2\alpha^3 \\
&= 1 + \beta^2\alpha + \alpha^2 + \beta^2\alpha^3 \\
&\vdots \\
&\vdots
\end{aligned} \tag{20}$$

그러므로 유한체 GF(2^8)의 0이 아닌 임의의 두 원소 U와 V는

$$U = \bar{u}_0 + \bar{u}_1\alpha + \bar{u}_2\alpha^2 + \bar{u}_3\alpha^3, \qquad \bar{u}_i \in \mathrm{GF}(2^2) \tag{21}$$

$$V = \bar{v}_0 + \bar{v}_1\alpha + \bar{v}_2\alpha^2 + \bar{v}_3\alpha^3, \qquad \bar{v}_i \in \mathrm{GF}(2^2) \tag{22}$$

로 표현할 수 있으며 이 두 원소의 곱 Z는 다음과 같이 쓸 수 있다.

$$\begin{aligned}
Z &= U \cdot V \\
&= U \cdot (\bar{v}_0 + \bar{v}_1\alpha + \bar{v}_2\alpha^2 + \bar{v}_3\alpha^3) \\
&= \bar{v}_0 U + \bar{v}_1(U\alpha) + \bar{v}_2(U\alpha^2) + \bar{v}_3(U\alpha^3)
\end{aligned} \tag{23}$$

앞에서와 마찬가지로 먼저 임의의 한 원소 U에 원시원 α를 곱하는 회로를 구성하여 보자. 식 (21)과 같은 임의의 한 원소 U에 원시원 α를 곱하면

$$\begin{aligned}
U \cdot \alpha &= \bar{u}_0\alpha + \bar{u}_1\alpha^2 + \bar{u}_2\alpha^3 + \bar{u}_3\alpha^4 \\
&= \bar{u}_0\alpha + \bar{u}_1\alpha^2 + \bar{u}_2\alpha^3 + \bar{u}_3(\beta + \beta\alpha + \beta\alpha^2 + \alpha^3) \\
&= \bar{u}_3\beta + (\bar{u}_0 + \bar{u}_3\beta)\alpha + (\bar{u}_1 + \bar{u}_3\beta)\alpha^2 + (\bar{u}_2 + \bar{u}_3)\alpha^3
\end{aligned} \tag{24}$$

가 되며, 이를 순서회로로 구현하면 그림 4와 같이 된다.

그림 4에서 모든 선은 2비트 선이며, □는 2비트 레지스터이고, ⊕는 $GF(2^2)$ 상의 가산기로 2개의 2입력 XOR 게이트로 구현할 수 있다. 또 [β]는 $GF(2^2)$ 상의 상수 β를 곱하는 회로로 그림 1의 (a)와 같은 회로이다.

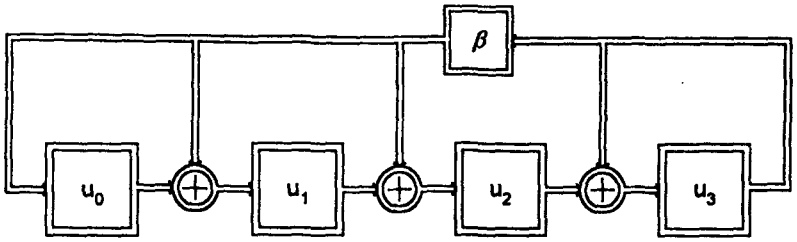

그림 4 $GF(2^8)$ 상에서 β를 곱하는 회로

따라서 식 (24)와 그림 4를 이용하면 그림 5와 같은 $GF(2^8)$ 상의 직/병렬 승산기를 구성할 수 있다. 그림 3에서와 마찬가지로 그림 5에서도 초기에 곱하고자 하는 두 수를 레지스터 U와 V에 로드하고 레지스터 Z는 클리어 시킨 다음, 클럭을 한 번 인가하면 레지스터 Z에는 $\overline{v}_0 U$가 나타나고, 두 번 인가하면 $\overline{v}_0 U + \overline{v}_1(U\alpha)$가, ……, 네 번 인가하면 $\overline{v}_0 U + \overline{v}_1(U\alpha) + \overline{v}_2(U\alpha^2) + \overline{v}_3(U\alpha^3)$가 나타나므로, 4 클럭 타임만에 결과가 나오게 된다.

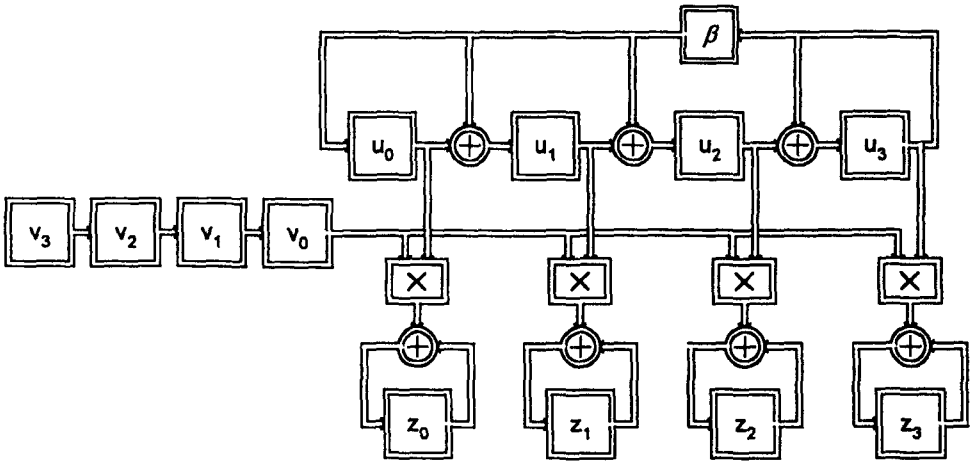

그림 5 $GF(2^8)$ 상의 직/병렬 혼합 승산기

그림 5에서 ⊠는 $GF(2^2)$ 상의 병렬 승산기로 그림 2와 같은 회로이다. 그러므로 그림 5와 같은 $GF(2^8)$ 상의 직/병렬 혼합 승산기의 회로 규모는 표 2와 같이 정리할 수 있다. $GF(2^8)$ 상의 병렬 승산기는 일반적으로 1클럭 타임만에 결과를 얻을 수 있는 반면에 2입력 AND 64개와 2입력 XOR 73개가 소요된다[8]. 직렬 승산기는 레지스터 24개, 2입력 AND 8개, 2입력 XOR 11개로 구현할 수 있으나 8 클럭 타임만에 결과가 나온다.

본 논문에서 제안한 직/병렬 혼합 승산기는 회로규모와 지연시간간을 적절히 절충하여 표 2와 같이 레지스터 24개, 2입력 AND 16개, 2입력 XOR 27개로 구현할 수 있으며 4 클럭 타임만에 결과를 얻을 수 있다.

표 2 $GF(2^8)$ 상의 직/병렬 혼합 승산기의 회로 규모

레지스터	4 × 3 × 2 = 24
$GF(2^4)$ 상의 병렬 승산기	2입력 AND : 4 × 4 = 16 2입력 XOR : 3 × 4 = 12
$GF(2^4)$ 상의 가산기	2입력 XOR : 2 × 7 = 14
β 승산기	2입력 XOR : 1
합 계	레지스터 : 24 2입력 AND : 16 2입력 XOR : 27

식 (21)에서 $GF(2^2)$의 원소로 표현된 $\overline{u}_i$를 GF(2)의 원소로 바꾸어 쓰면, $\beta = \alpha^{85}$ 이므로 다음과 같이 쓸 수 있다.

$$\begin{aligned} U &= \overline{u}_0 + \overline{u}_1\alpha + \overline{u}_2\alpha^2 + \overline{u}_3\alpha^3 \\ &= (\overline{u}_0 + \overline{u}_1\beta) + (\overline{u}_2 + \overline{u}_3\beta)\alpha + (\overline{u}_4 + \overline{u}_5\beta)\alpha^2 + (\overline{u}_6 + \overline{u}_7\beta)\alpha^3 \\ &= \overline{u}_0 + \overline{u}_1\alpha^{85} + \overline{u}_2\alpha + \overline{u}_3\alpha^{86} + \overline{u}_4\alpha^2 + \overline{u}_5\alpha^{87} + \overline{u}_6\alpha^3 + \overline{u}_7\alpha^{88} \end{aligned} \quad (25)$$

또 이를 식 (12)를 이용하여 정리하면 다음과 같이 된다.

$$\begin{aligned} U &= \overline{u}_0 + \overline{u}_1(\alpha+\alpha^2+\alpha^4+\alpha^6+\alpha^7) + \overline{u}_2\alpha + \overline{u}_3(1+\alpha^4+\alpha^5+\alpha^7) + \overline{u}_4\alpha^2 + \overline{u}_5 \\ &\quad (1+\alpha+\alpha^2+\alpha^3+\alpha^4+\alpha^5+\alpha^6) + \overline{u}_6\alpha^3 + \overline{u}_7(\alpha+\alpha^2+\alpha^3+\alpha^4+\alpha^5+\alpha^6+\alpha^7) \\ &= (\overline{u}_0+\overline{u}_3+\overline{u}_5) + (\overline{u}_1+\overline{u}_2+\overline{u}_5+\overline{u}_7)\alpha + (\overline{u}_1+\overline{u}_4+\overline{u}_5+\overline{u}_7)\alpha^2 \\ &\quad + (\overline{u}_5+\overline{u}_6+\overline{u}_7)\alpha^3 + (\overline{u}_1+\overline{u}_3+\overline{u}_5+\overline{u}_7)\alpha^4 + (\overline{u}_3+\overline{u}_5+\overline{u}_7)\alpha^5 \\ &\quad + (\overline{u}_1+\overline{u}_5+\overline{u}_7)\alpha^6 + (\overline{u}_1+\overline{u}_3+\overline{u}_7)\alpha^7 \end{aligned} \quad (26)$$

따라서 식 (26)을 이용하면 다음과 같이, 표준기저로 표현된 $GF(2^8)$의 원소들을 $GF(2^2)$ 상의 기저 표현으로 변환할 수 있다.

$$
\begin{aligned}
u_0 &= \overline{u}_0 + \overline{u}_3 + \overline{u}_5 \\
u_1 &= \overline{u}_1 + \overline{u}_2 + \overline{u}_5 + \overline{u}_7 \\
u_2 &= \overline{u}_1 + \overline{u}_4 + \overline{u}_5 + \overline{u}_7 \\
u_3 &= \overline{u}_5 + \overline{u}_6 + \overline{u}_7 \\
u_4 &= \overline{u}_1 + \overline{u}_3 + \overline{u}_5 + \overline{u}_7 \\
u_5 &= \overline{u}_3 + \overline{u}_5 + \overline{u}_7 \\
u_6 &= \overline{u}_1 + \overline{u}_5 + \overline{u}_7 \\
u_7 &= \overline{u}_1 + \overline{u}_3 + \overline{u}_7
\end{aligned}
\tag{27}
$$

또한 식 (27)을 선형 조합하면 다음과 같이 역변환도 할 수 있다.

$$
\begin{aligned}
\overline{u}_0 &= u_0 + u_6 + u_7 \\
\overline{u}_1 &= u_4 + u_5 \\
\overline{u}_2 &= u_1 + u_6 \\
\overline{u}_3 &= u_4 + u_6 \\
\overline{u}_4 &= u_2 + u_6 \\
\overline{u}_5 &= u_4 + u_7 \\
\overline{u}_6 &= u_3 + u_4 + u_5 + u_6 \\
\overline{u}_7 &= u_5 + u_6 + u_7
\end{aligned}
\tag{28}
$$

식 (27) 및 식 (28)과 같은 변환 및 역변환은 각각 10개의 2입력 XOR로 구현할 수 있다.

4. 결 론

본 논문에서는 표준기저 상에서 GF(2^2) 상의 병렬 승산기를 이용하여 GF(2^8) 상의 직/병렬 혼합 승산기를 설계하였다. 제안된 승산기는 직렬 승산기의 긴 지연시간과 병렬 승산기의 복잡한 회로 사이를 적절하게 절충함으로써, 직렬 승산기 보다는 짧은 지연시간에 결과를 얻을 수 있으며 병렬 승산기 보다는 적은 회로로 구현할 수 있는 장점이 있다.

본 논문에서 사용한 유한체 상의 부분체를 이용한 연산 방법은 유한체 상의 승산뿐만 아니라 역원의 계산에도 매우 효율적으로 응용될 수 있을 것으로 생각된다.

참 고 문 헌

[1] 이 만 영, *BCH 부호와 Reed-Solomon 부호*, 민음사, 1988

[2] E. R. Berlekamp, "Bit-Serial Reed-Solomon Encoders," *IEEE Trans. Information Theory*, Vol.28, pp.869~874, Nov. 1982.

[3] J. L. Massey and J. K. Omura, "Computational Method and Apparatus for Finite Field Arithmetic," U. S. Patent Application, Submitted 1981.

[4] T. K. Truong, L. J. Deutsch, I. S. Reed, I. S. Hsu, K. Wang, and C. S. Yeh, "The VLSI Implementation of a Reed-Solomon Encoder Using Berlekamp's Bit-Serial Multiplier Algorithm," *IEEE Trans. Computers*, Vol.33, No.10, pp.906~911, Oct. 1984.

[5] M. A. Hasan and V. K. Bhargava, "Division and Bit-Serial Multiplication over GF(q^m)," *IEE Proc. E.*, Vol.139, pp.230~236, May 1992.

[6] C. C. Wang, T. K. Truong, H. M. Shao, L. J. Deutsch, J. K. Omura, and I. S. Reed, "VLSI Architectures for Computing Multiplications and Inverses in GF(2^m)," *IEEE Trans. Computers*, Vol.34, No.8, pp.709~716, Aug. 1985.

[7] S. T. J. Fenn, M. Benaissa and D. Taylor, "Fast normal basis inversion in GF(2^m)," *Electronics Letters*, Vol.32, No.17, pp.1566~1567, Aug. 1996.

[8] A. M. Patel, "On-the-fly decoder for multiple byte errors," *IBM J. RES. DEVELOP.*, Vol. 30, No.3, pp.259~269, May 1986.