論文2001-38SC-1-3

$GF(2^m)$상의 기약 3 항식을 이용한 승산기 설계

(A Design of Multiplier Over $GF(2^m)$ using the Irreducible Trinomial)

黃 鍾 學*, 沈 載 煥*, 崔 在 碩**, 金 興 壽*

(Jong-Hak Hwang, Jai-Hwan Sim, Jai-Sock Choi, and Heung-Soo Kim)

요 약

$GF(2^m)$상의 원시 기약 3항식인 x^m+x+1을 이용한 승산기 알고리즘은 Mastrovito에 의해 제안되었다. 본 논문에서는 기약 3항식 x^m+x^n+1에서 $1<n<m/2$을 만족하는 승산기를 구성하였으며, 승산 연산부와 원시 기약다항식 연산부, mod 연산부로 구성하였다. 승산 연산부는 타 논문과 비교하기 위하여 기존 논문의 알고리즘을 이용하였으며 mod 연산부는 원시 기약 다항식 연산부의 연산 결과를 적용할 수 있게 구성하였다. 특히 원시 기약 다항식은 승산 연산부의 연산 과정에서 필연적으로 발생되는 m차 이상의 항을 $m-1$차 이하의 항으로 표현하기 위하여 필요하다. 따라서 본 논문에서는 $GF(2^m)$상의 원시 기약 3 항식을 전개하여 회로를 간략화 하였으며, 제안된 승산기 설계는 규칙적이며 모듈러 구조, 그리고 간단한 제어신호를 요하기 때문에 VLSI 실현이 용이하다고 사료된다.

Abstract

The multiplication algorithm using the primitive irreducible trinomial x^m+x+1 over $GF(2^m)$ was proposed by Mastrovito. The multiplier proposed in this paper consisted of the multiplicative operation unit, the primitive irreducible operation unit and mod operation unit. Among three units mentioned above, the primitive irreducible operation was modified to primitive irreducible trinomial x^m+x^n+1 that satisfies the range of $1<n<m/2$. The multiplicative operation unit was adopted from an existing algorithm. The results of the primitive irreducible operation unit and the multiplicative operation unit were used for computing the mod operation unit. The primitive irreducible polynomial would be better if the size of the result of multiplication operation unit in the process of converting $x^m, \cdots, x^{2m-2}$ to $x^{m-1}, \cdots, x^0$ is reduced. In this paper, the primitive irreducible polynomial was reduced to the primitive irreducible trinomial proposed. As a result of this reduction, the primitive irreducible trinomial reduced the size of circuit. In addition, the proposed design of multiplier was suitable for VLSI implementation because the circuit became regular and modular in structure, and required simple control signal.

Ⅰ. 서 론

유한체(Galois field)는 스위칭 이론, 오진 정정 부호, 디지털 신호 처리 및 화상 처리, 디지털 통신의 암호화 및 해독화를 요하는 보안 통신등에 많이 응용되고 있다. 특히, $GF(2^m)$은 신호 처리와 화상처리 응용 분야에서 특별한 계산을 요하거나 범용 컴퓨터 계산의 고속화를 보조하는 고성능 전용 컴퓨터의 설계에 효과적이며, VLSI 설계에 응용되고 있다[1-3]

* 正會員, 仁荷大學校 電子工學科

(Dept. of Electrical Eng., Inha University)

** 正會員, 仁德大學 메카트로닉스科

(Dept. of Mechatronics, Induk Institute of Technology)

接受日字 : 2000年8月11日, 수정완료일 : 2000年12月12日

유한체상에서 가산과 승산은 관용 2진 산술 연산과는 현저하게 다르므로 실제적으로 유용성과 단순성에 기인하여 유한체 $GF(2^m)$에 관한 연구가 활발히 진행되고 있다. 유한체상의 가산은 직접적이고 비트 독립적인 mod(2)연산으로 관용 2진 가산보다 쉬운 반면 승산은 관용 2진 승산 보다 어렵고 복잡한 계산을 요한다.[7,8] VLSI설계에서 모듈 구조와 규칙적 상호연결이 중요한 설계 객체이다. 유한체상의 승산을 위한 알고리즘이 지난 십 수년간 제안되어 왔으나 불행하게도 이들 알고리즘은 불규칙한 회선 경로 선택, 복잡한 제어 문제, 비모듈화 구조 및 병발성의 부족 때문에 VLSI구조의 설계에 부적합하였다.[9,10]

최근 Yeh등[2]은 표준 기저 표현식을 사용하여 유한체상의 승산을 실현하는 직렬 입력/직렬 출력 시스토릭 배열 구조와 병렬 입력/병렬 출력 시스토릭 배열 구조의 승산기를 개발하였다. Scott등[5]은 표준 기저로 표현된 각 원소들의 유한체 승산을 실행하는 고속 승산기를 제시하였고, Wang등[10]은 Scott등이 제안한 유한체상의 승산 알고리즘을 이용하여 시스토릭 배열의 승산기를 제시하였다. 그러나 이 들이 제시한 승산기는 레지스터를 이용하기 때문에 클럭시간이 필요로 한다. 또한 Mastrovito[19,20]에 의해서 기약 3항식 x^m+x+1에 대한 승산 알고리즘이 제안되었으며, Sunar[18]에 의하여 기약 3항식 x^m+x^n+1을 이용한 승산기를 제안하였으나 n을 정하여 행렬식으로 수식을 전개하는 과정을 필요로 한다.

본 논문에서는 $GF(2^m)$상에서 두 다항식을 승산하였을 경우 승산연산부에서는 최고차 항이 $2m-2$차 항까지 발생한다. 이중 m차 이상의 항은 $m-1$이하의 항으로 다시 표현하여야 하는데, 이는 원시 기약 다항식을 이용하여야 한다. 그러나 일반적인 원시 기약다항식을 적용하여 변환하였을 때 회로 소자가 m^2-2m XOR와 m^2-2m AND게이트를 필요로 한다. 그러나 본 논문에서 제시한 $GF(2^m)$상의 3 항식 x^m+x^n+1에서 $1<n<m/2$인 경우 $m/2$ XOR 게이트만 필요하고 또한 회로를 축약하기 위하여 사전에 수식 계산을 필요치 않다. $m/2\le n$ 경우는 $3m-2n$차 항을 표현할 때 m차 항이 발생할 수 있으므로 다시 $m-1$차 항으로 표현하기 위하여 회로소자를 필요로 한다. 그러나 본 논문에서는 원시 기약 다항식 연산부의 축약에 중심으로 승산기를 구성하였다. 그리하여 타 논문과 비교하기 위하여 승산 연산부는 Hwang[17]에 의해 제안한 승산 연산부를 이용하여 구성하였다.

Ⅱ. 유한체의 승산 알고리즘[5,10,11,14]

$GF(2^m)$에서 표준기저 승산 동작은 종종 다항식 승산과 모듈러 환산인 두 단계에서 성취된다. $a(x), b(x), c(x)\in GF(2^m)$인 다항식들과 원시기약다항식인 $p(x)$와 관계식은 식 (1)과 같이 표현된다

$$c(x)=a(x)b(x)\bmod p(x) \tag{1}$$

식 (1)을 다시 $c(x)=d(x)\bmod p(x)$로 표현하면 $d(x)$와 $p(x)$는 식 (2), 식 (3)과 같이 표현할 수 있다.

$$d(x)=a(x)b(x)=\left(\sum_{i=0}^{m-1}a_ix^i\right)\left(\sum_{i=0}^{m-1}b_ix^i\right) \tag{2}$$

$$p(x)=x^m+\left(\sum_{i=0}^{m-1}p_ix^i\right) \tag{3}$$

식 (3)에서 3 항식이 만들기 위하여 x^n과 x^0의 계수는 1이고 그 외의 계수는 0이 되어야한다. 즉 $p_n=p_0=1$이고, 그 외 p_i는 0이다.

식 (2)에서 $a(x)$와 $b(x)$을 곱하여 전개하면, 0차 항에서 $2m-2$차 항까지 얻는다. 이 중에서 m차에서 $2m-2$차 항까지의 항들은 $m-1$차 이하의 항으로 표현하기 위하여 식 (3)의 모닉다항식 $p(x)=0$에 의해 다시 표현된다. 즉 m차의 항은 식 (4)과 같이 표현된다.

$x^m=\left(\sum_{i=0}^{m-1}p_ix^i\right)$ 또는

$$\begin{aligned}x^m=\;&p_{m-1}x^{m-1}+p_{m-2}x^{m-2}+\cdots\\&+p_nx^n+\cdots+p_2x^2+p_1x^1+p_0\end{aligned} \tag{4}$$

식 (4)을 이용하여 $x^{m+1}, x^{m+2},\cdots, x^{2m-n-1}$차 항은 식 (5)과 같이 표현된다.

$$\begin{aligned}x^{m+1}&=p_{m-2}x^{m-1}+p_{m-3}x^{m-2}+\cdots+p_nx^{n+1}\\&\quad+\cdots+p_2x^3+p_1x^2+p_0x^1\\x^{m+2}&=p_{m-3}x^{m-1}+p_{m-4}x^{m-2}+\cdots+p_nx^{n+2}\\&\quad+\cdots+p_2x^4+p_1x^3+p_0x^2\\&\vdots\\x^{2m-n-1}&=p_nx^{m-1}+p_{n-1}x^{m-2}+\cdots\\&\quad+p_2x^{m-n+1}+p_1x^{m-n}+p_0x^{m-n-1}\end{aligned} \tag{5}$$

식 (4)과 식 (5)에서 임의의 k을 이용하여 표현하면 식 (6)과 같이 표현된다.

$$x^{m+k} = p_{m-1-k}\,x^{m-1} + p_{m-2-k}\,x^{m-2} + \cdots + p_n\,x^{n+k} + \cdots + p_2\,x^{k+2} + p_1\,x^{k+1} + p_0\,x^k = \sum_{i=k}^{m-1} p_{m-1-i}\,x^{m-1+k-i} \tag{6}$$

여기서 $0 \le k \le m-n-1$(단, k는 정수)이다.

n이 $1 < n < \frac{m}{2}$일 경우에 x^{2m-n}은 식 (7)과 같이 표현된다.

$$x^{2m-n} = p_n x^m + p_{n-1}x^{m-1} + \cdots + p_2 x^{m-n+2} + p_1 x^{m-n+1} + p_0 x^{m-n} \tag{7}$$

식 (7)에 x^m차 항이 발생하므로 식 (4)을 대입하여 전개한다. 여기서 $p_n = p_0 = 1$이고 그 외의 계수는 0이므로 식 (8)과 같이 다시 쓸 수 있다.

$$x^{2m-n} = p_n(p_{m-1}x^{m-1} + p_{m-2}x^{m-2} + \cdots + p_2x^2 + p_1x + p_0) + p_{n-1}x^{m-1} + \cdots + p_2x^{m-n+2} + p_1x^{m-n+1} + p_0x^{m-n} \tag{8}$$

식 (8)은 다시 식 (9)과 같이 전개할 수 있다.

$$\begin{aligned} x^{2m-n} = {} & p_np_{m-1}x^{m-1} + p_np_{m-2}x^{m-2} + \cdots + \\ & p_np_{m-n}x^{m-n} + \cdots + p_np_nx^n + \cdots + \\ & p_np_2x^2 + p_np_1x + p_np_0 + p_{n-1}x^{m-1} + \cdots \\ & + p_2x^{m-n+2} + p_1x^{m-n+1} + p_0x^{m-n} \end{aligned} \tag{9}$$

식 (9)을 정리하여 식 (10)에 다시 표현하였다.

$$\begin{aligned} x^{2m-n} = {} & (p_np_{m-1} + p_{n-1})x^{m-1} + (p_np_{m-2} + p_{n-2})x^{m-2} + \cdots + \\ & (p_np_{m-n} + p_0)x^{m-n} + p_np_{m-n-1}x^{m-n-1} + \cdots + \\ & p_np_{n+1}x^{n+1} + p_np_nx^n + \cdots + p_np_2x^2 + p_np_1x + p_np_0 \end{aligned} \tag{10}$$

식 (10)에서 3 항식 조건에 의하여 p_n과 p_0만 1이고 그 외의 p_i는 0이다. 따라서 식 (10)에서 1의 계수를 갖는 항은 x^{m-n}, x^n, x^0이다. 즉, $p_np_{m-n}+p_0$, p_np_n, p_np_0는 1이다. 그러므로 $p_np_{m-n}+p_0$는 $p_{m-n}\oplus p_n$, p_np_n는 $p_n\oplus p_{m-n}$, p_np_0는 p_0으로 표기해도 같은 결과를 얻는다. 그리고 0의 계수를 갖는 차수의 항은 $x^{m-1}, \cdots, x^{m-n+1}$과 $x^{m-n-1}, \cdots, x^{n+1}, x^{n-1}, \cdots, x$이다. 즉 $p_np_{m-1}+p_{n-1}, \cdots, p_np_{m-n+1}+p_1$과 $p_np_{m-n-1}, \cdots, p_np_{n+1}, p_np_{n-1}, \cdots, p_np_1$는 0이다. 그러므로 $p_{m-1}\oplus p_1, \cdots, p_{m-n+1}\oplus p_{n-1}$과 $p_{m-n-1}\oplus p_{n+1}, \cdots, p_{n+1}\oplus p_{m-n-1}, p_{n-1}\oplus p_{m-n+1}, \cdots, p_1\oplus p_{m-1}$으로 표기해도 같은 결과를 얻는다. 이를 정리하여 식 (11)과 같이 다시 표현할 수 있다.

$$x^{2m-n} = p_{m-1}\oplus p_1\,x^{m-1} + p_{m-2}\oplus p_2\,x^{m-2} + \cdots + p_2\oplus p_{m-2}\,x^2 + p_1\oplus p_{m-1}\,x^1 + p_0 \tag{11}$$

식 (11)을 간략화하기 위하여 식 (12)과 같이 다시 표현한다.

$$x^{2m-n} = P_{m-1}\,x^{m-1} + P_{m-2}\,x^{m-2} + \cdots + P_2\,x^2 + P_1\,x^1 + p_0 \tag{12}$$

여기서 $P_k = p_{m-k}\oplus p_k$이다.

식 (12)에서 $P_{m-1} = P_1, P_{m-2} = P_2, \cdots, P_{m-k} = P_k$,인 관계가 성립하므로 식 (13)과 같이 표현된다. 여기서 k는 ⌊ $m/2$ ⌋이고, 기호 ⌊ c ⌋는 c보다 크지 않은 제일 큰 정수이다.

$$\begin{aligned} x^{2m-n} &= P_1x^{m-1} + P_2x^{m-2} + \cdots + P_2x^2 + P_1x^1 + p_0 \\ &= P_1(x^{m-1} + x^1) + P_2(x^{m-2} + x^2) + \cdots \\ &\quad + P_k(x^{m-k} + x^k) + p_0 \quad (m\text{이 홀수}) \\ &= P_1(x^{m-1} + x^1) + P_2(x^{m-2} + x^2) + \cdots + P_kx^k + p_0 \\ &\quad (m\text{이 짝수}) \end{aligned} \tag{13}$$

x^{2m-n+1}항에서 x^{2m-2} 항까지는 식 (13)을 이용하여 식 (14)과 유도할 수 있다.

$$\begin{aligned} x^{2m-n+1} &= P_2x^{m-1} + P_3x^{m-2} + \cdots + P_{k-1}x^{k+2} + P_kx^{k+1} + P_{k-1}x^k \\ &\quad + \cdots + P_2x^3 + P_1x^2 + p_0x \\ x^{2m-n+2} &= P_3x^{m-1} + P_4x^{m-2} + \cdots + P_{k-1}x^{k+3} + P_kx^{k+2} + P_{k-1}x^{k+1} \\ &\quad + \cdots + P_2x^4 + P_1x^3 + P_0x^2 \\ &\vdots \\ x^{2m-2} &= P_{n-1}x^{m-1} + P_nx^{m-2} + \cdots + P_{k-1}x^{k+n-1} + P_kx^{k+n-2} + P_{k-1} \\ &\quad x^{k+n-3} + \cdots + P_1x^{n-1} + p_0x^{n-2} \end{aligned} \tag{14}$$

식 (14)에서 임의의 $0 \le k \le n-2$(단, k는 정수)일 때, 식 (15)과 같이 표현된다.

$$x^{2m-n+k} = P_{(k+1)}\,x^{m-1} + P_{(k+2)}\,x^{m-2} + \cdots + P_1\,x^{k+1} + p_0\,x^k$$

$$= \sum_{i=k+1}^{m-1} P_{i-k}x^i + p_0 x^k \quad (15)$$

여기서 $P_{m-k} = P_k$이다.

$m=7$, $n=3$인 원시 기약다항식 $p(x)=x^7+x^3+1$인 경우를 본 논문에서 제시한 알고리즘으로 예를 들어 설명하여보자. 식 (4)에서 x^m은 $x^7 = p_6x^6 + p_5x^5 + p_4x^4 + p_3x^3 + p_2x^2 + p_1x^1 + p_0$ (단, $p_3 = p_0 = 1$이고, 그 외의 계수는 0이다)으로 표현 할 수 있다. 그리고 식 (5)에 의하여 x^{m+1}에서 x^{2m-n-1}까지 구할 수 있다. $x^8 = p_5x^6 + p_4x^5 + p_3x^4 + p_2x^3 + p_1x^2 + p_0x^1$, $x^9 = p_4x^6 + p_3x^5 + p_2x^4 + p_1x^3 + p_0x^2$, $x^{10} = p_3x^6 + p_2x^5 + p_1x^4 + p_0x^3$이다. 그리고 x^{2m-n}에서 x^{2m-2}까지 식 (13), 식 (14)에 의해 구할 수 있다. $x^{11} = p_{7-1}\oplus p_1x^6 + p_{7-2}\oplus p_2x^5 + p_{7-3}\oplus p_3x^4 + p_{7-4}\oplus p_4x^3 + p_{7-5}\oplus p_5x^2 + p_{7-6}\oplus p_6x^1 + p_0 = P_1x^6 + P_2x^5 + P_3x^4 + P_3x^3 + P_2x^2 + P_1x^1 + p_0$ 이다. 그리고 $x^{12} = P_2x^6 + P_3x^5 + P_3x^4 + P_2x^3 + P_1x^2 + p_0x^1$이다.

Ⅱ. 3 항식을 이용한 $GF(2^m)$상에서 병렬 승산기 구현

이 장에서는 $GF(2^m)$상의 승산 $c(x)=a(x)\cdot b(x) \bmod p(x)$을 실행하는 병렬 입출력 승산기의 구성을 논한다. 그림 1은 $GF(2^m)$상의 두 원소들의 승산을 실행하는 승산기의 구성도 이다.

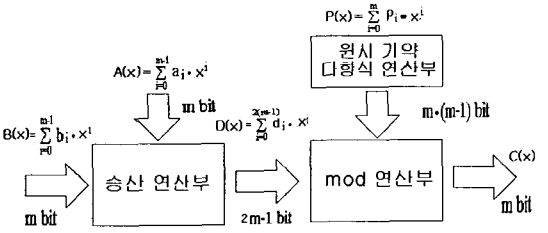
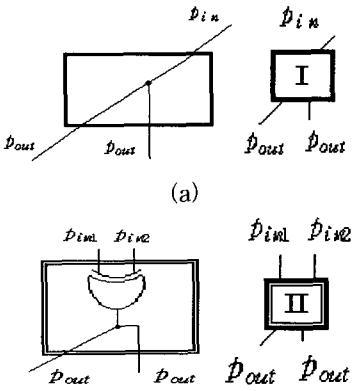


그림 1. $GF(2^m)$상의 승산기
Fig. 1. A multiplier in $GF(2^m)$.

그림 1의 승산기는 $GF(2^m)$상의 두 원소들의 승산을 실행하는 승산 연산부와 승산 연산부의 출력을 입력으로 하여 원시기약 다항식에 의한 mod 연산을 행하는 mod 연산부와 원시기약 다항식을 산술연산 처리하는 기약 다항식 연산부로 구성되어 있다.

1.원시기약다항식 연산부

$GF(2^m)$상의 두 원소들 승산을 행하였을 경우 m차 이상의 항들이 발생한다. 따라서 원시기약 다항식을 이용하여 m차 이상의 항들을 $m-1$이하의 항으로 표현하여야 한다. 앞장에서 제시한 알고리즘으로 회로도를 구성을 하기 위하여 그림 2에 기본 셀과 그 기호를 나타내었다.



(a)

(b)

그림 2. $GF(2^m)$상의 원시 기약다항식 연산부 기본 셀 (a) 셀 I 과 기호 (b) 셀 II 과 기호
Fig. 2. The basic cell of primitive irreducible operation part in $GF(2^m)$ (a) Cell I and its symbol (b) Cell II and its symbol.

식 (2)에서 식 (12)까지의 전개된 식을 이용하여 m차에서 $2m-n-1$차 항까지 회로도를 그림 3에 나타내었으며, 각 셀들은 그림 2에 있는 셀 기호를 적용하였다.

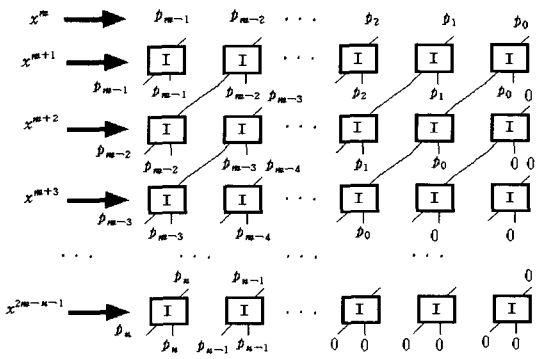


그림 3. 기약 3 항식 x^m+x^n+1을 차 항에서 $2m-n-1$차 항까지 원시 기약다항식 연산부
Fig. 3. The primitive irreducible operation part having degrees ranging from m to $2m-n-1$ using the irreducible trinomial.

그림 3에서 보는 바와 같이 회로도의 구성은 게이트가 쓰이지 않고 내부 결선으로만 구성되어 있다. 따라서 회로 공간 복잡도는 0이 된다. 그리고 식 (13)과 식 (14)를 이용하여 $2m-n$차에서 $2m-2$차 항까지의 회로도는 그림 4와 같이 구성되고 그림 2의 셀 기호를 사용한다.

그림 4에서 보는 바와 같이 맨 위 내부는 $m/2$개의 XOR게이트로 구성되고 그 외에는 내부 결선으로 구성되어 있다. 따라서 회로의 공간 복잡도는 $m/2$개의 XOR게이트로 구성된다. 그래서 원시기약 다항식 전체 복잡도는 $m/2$개의 XOR게이트로 구성된다.

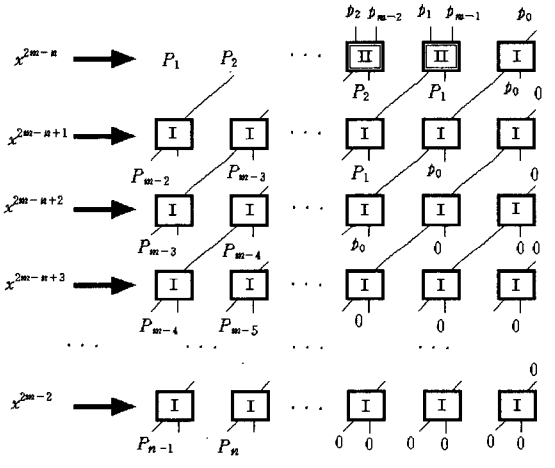


그림 4. 기약 3 항식 x^m+x^n+1을 이용한 $2m-n$차 항에서 $2m-2$차 항까지 원시 기약다항식 연산부

Fig. 4. The primitive irreducible operation part having degrees ranging from $2m-n$ to using $2m-2$ the irreducible trinomial.

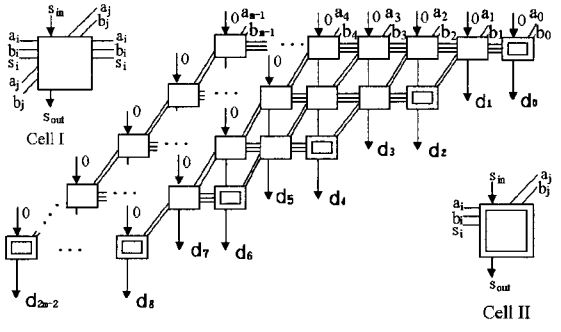


그림 5. $GF(2^m)$상의 멀티플렉서를 이용한 승산기

Fig. 5. The proposed multiplexer-based parallel multiplier in $GF(2^m)$.

2. 승산 연산부

$GF(2^m)$상의 승산은 식 (2)에서 $d(x)=a(x)b(x)$를 계산하는데 있어 Hwang[17]가 제안한 MUX와 XOR, AND게이트로 구성된 회로를 이용한다. 회로도는 그림 6, 7과 같다.

그림 5에서 *Cell I*과 *Cell II*의 회로 구성을 그림 6에 나타내었다. 소자는 $m^2/2+3m/2$ XOR와 m AND 게이트가 필요하고, MUX는 $m^2/2+m/2$가 필요하다.

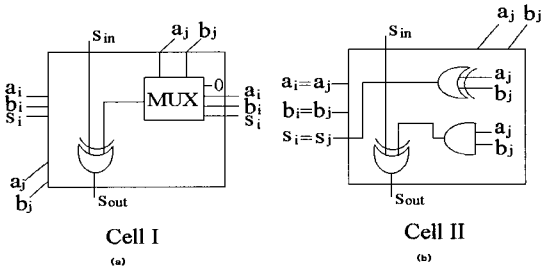


그림 6. (a) 제안된 승산기에 사용된 *Cell I*의 내부 회로도 (b) 제안된 승산기에 사용된 *Cell II*의 내부 회로도

Fig. 6. (a) First type cell (CELL I) used in the proposed array multiplier. (b) Second type cell(CELL II) used in the proposed array multiplier.

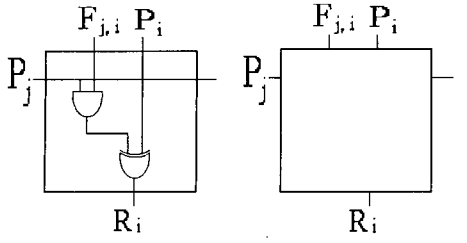
3. mod 연산부

승산 연산부와 기약 다항식 연산부를 통하여 출력된 결과 값을 최종적으로 mod연산을 하여야 $GF(2^m)$상에서 승산이 완료된다. mod 연산부의 수식은 식 (16)와 같이 표현된다.

$$c(x)=\sum_{i=0}^{m-1} d_i x^i \oplus \left[\sum_{i=m}^{2(m-1)} d_i x^i\right] \tag{16}$$

식 (16)을 이용하여 회로도를 구성하면 그림 7, 그림 8과 같다.

식 (16)은 두 번째 항인 $\sum_{i=m}^{2(m-1)} d_i x^i$은 승산 연산부에서 발생된 m차 이상의 항을 $m-1$차 이하의 항으로 표현하기 위하여 원시 기약 다항식 연산부에서 구한 $x^{2m-2},\cdots,x^m$과 AND를 하고 XOR한다. 이것은 그림 8에서 $d_{2m-2},\cdots,d_m$부분에 해당 된다. 그리고 첫 번째 항인 $\sum_{i=0}^{m-1} d_i x^i$은 승산 연산부에서 발생된 $m-1$ 값들과 식 (16)의 두 번째 항과 XOR를 한다.



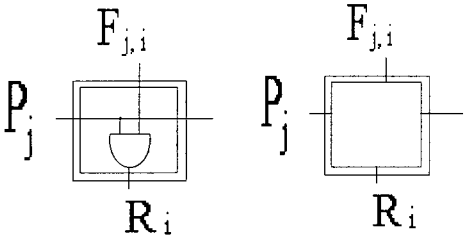


그림 7. $GF(2^m)$상의 mod 연산부의 기본 셀 (a)셀 I 및 기호 (b) 셀 II 및 기호

Fig. 7. The basic cell of modular operation part in $GF(2^m)$ (a) Cell I and it's symbol (b) Cell II and it's symbol.

mod 연산부의 소자수는 m^2-m-x XOR와 m^2-m-y AND 게이트가 필요하다. 여기서 $x = n(n-2)/2 + (m-n-1)(m-n)/2 + n-2$이고 $y = n(n-2)/2 + (m-n-1)(m-n)/2$이다.

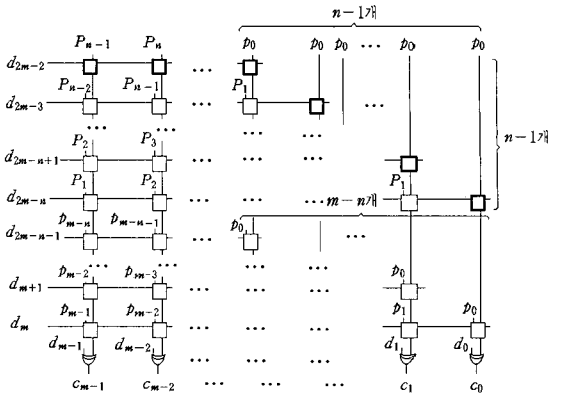


그림 8. $GF(2^m)$상의 mod 연산부

Fig. 8. The modularity operation part in $GF(2^m)$ mod연산부 게이트

Ⅳ. 비교 및 검토

본 장에서는 제시한 Mastrovito의 3 항식을 보다 확장하여 승산기를 구성하였고, 타 논문과 비교하여 표 1에 나타내었다.

타 연구의 승산기들과 본 논문에서 제시한 원시 기약 3 항식를 이용한 연산기를 각 연산 회로별로 비교하면 다음과 같다. 클럭 시간과 전달 지연 시간도 단일 연산 회로에 전달되는 시간이 같다고 가정하여 단위 시간으로 계산하였다.[14]

표 1. 비교표

Table 1. The table of comparison.

		Yeh[12]	Wang[10]	Scott[8]	Koc[15]	Paar[16]	Hwang[11]	This paper
AND	1-D	$3m$	$3m$	·	m^2	$9m^2$	$2m^2-2m$	m^2-y
	2-D	$2m^2$	$2m^2$					
XOR	1-D	$2m$	$2m$	$2m$	m^2-1	$9m^2+21m-9$	$5m^2/2-m/2$	$3/2m^2+m-1-x$
	2-D	$2m^2$	$2m^2$					
REGISTER	1-D	$10m+2$	$9m$	$4m+1$	·	·	·	·
	2-D	$7m^2+16$	$7m^2$					
INVERTER	1-D	·	·	2	$2m$	·	·	·
	2-D	·	·					
SWITCH	1-D	m	·	$8m$	·	·	·	·
	2-D	·	·					
MUX	1-D	·	·	·	·	·	$m^2/2+m/2$	$m^2/2+m/2$
	2-D	·	·					
CK TIME	1-D	$3m$	$3m$	m	$2m$	$2m+5$	$m-1$	$m-1$
	2-D	$3m$	$3m$					

단, $x = n(n-2)/2 + (m-n-1)(m-n)/2 + n-2$, $y = n(n-2)/2 + (m-n-1)(m-n)/2$

표 1에서 보는 바와 같이 Yeh[2], Wang[10]과 Scott[8]은 회로를 간략화 하기 위하여 레지스터를 사용하였다. 그리하여 레지스터를 사용하므로서 앞에서 연산된 결과 값을 저장하고 저장된 결과 값을 그 다음 연산을 하는 반복 연산이다. 이 반복 연산을 하기 위하여 클럭을 인가하여 반복 연산에 동기를 신호로 사용하고 있다. 따라서 클럭이 0V에서 5V으로 상승할 때 안정 시간과 5V에서 0V로 하강할 때 회로 안정시간이 부가적으로 더 필요하다. 그러나 본 논문에서는 클럭을 요하지 않는다. 그리고 Koc[15]는 인버터소자를 사용하였고, 회로 지연 시간이 $2m$이 소요된다. Paar[16]은 AND와 XOR게이트 소자가 $9m^2$에 비례하여 소요되며 지연 시간도 $2m$에 비례한다. Hwang[17]은 MUX를 사용하여 승산하는 계수들을 병렬 연산할 수 있도록 하였고 사용되는 MUX는 승산하는 계수들에 의하여 계산에 필요한 값을 선택하며 $1/2(m^2-m)$개가 소요된다. 그리고 $2m^2-2m$개 AND 게이트와 $5m^2/2-m/2$개의 XOR 게이트가 요한다, 본 논문에서는 원시 기약 다항식 중 3항식을 이용하여 AND게이트와 XOR게이트를 각각 m^2와 $3/2m^2+m-1$로 축소하였다. 제안된 원시 기약다항식중 3항식을 이용한 $GF(2^m)$상의 승산기는 회로 설계시 차수 m이 증가함에 따라 기본 셀을 부가하므로 설계가 용이한 모듈성과 회로 소자수가 규칙적으로 증가하는 규칙성을 가지므로 VLSI 실현에 적합할 것으로 생각된다.

V. 결 론

본 논문에서는 유한체 $GF(2^m)$상에서 두 원소들의 승산을 실현하는 Mastrovito 원시 기약 3항식 이용한 승산기를 제시 하였다. 이 승산기는 승산 연산부, mod 연산부, 원시 기약다항식 연산부로 구성된다.

승산 연산부는 AND와 XOR, MUX게이트로 설계한 기본 셀에 의하여 구성되며, mod 연산부는 AND와 XOR게이트의 기본 셀에 의하여 구성된다. 또한 원시 기약다항식 연산부는 AND와 XOR게이트들을 사용하여 구성하였다. 이 승산기의 소요 소자수는 m^2 AND 게이트와 $3/2m^2+m-1$ XOR게이트, $m^2/2+m/2$ MUX개이다. 동작시간은 승산 연산부와 mod 연산부가 동시에 실행되므로 $m-1$지연 시간이 소요된다.

본 논문에서 제시한 승산기는 회선 경로 선택의 규칙성, 간단성, 배열의 모듈성, 병발성의 이점을 가지며, 특히 차수 m이 증가하는 유한체의 두 원소들의 승산에서 확장성을 가지므로 VLSI실현에 적합하다고 사료된다.

참 고 문 헌

[1] H.M. Shao, T.K.Truong, L.J. Deutsch, J.H. Yaeh and I.S. Reed, "A VLSI design of a pipeling reed-solomon decoder," *IEEE Trans. Comput.*, vol. C-34, pp. 393-403, May 1985.

[2] C.S. Yeh, I.S Reed and T.K. Truong, "Systolic multipliers for finite field GF(2^m)," *IEEE Trans. Comput.*, vol. C-33, pp. 357-360, Apr. 1984.

[3] C.C. Wang, T.K. Truong, H.M. Shao, L.J. Deutsch, J.K. Omura and I.S. Reed, "VLSI architecture for computing multiplications and inverses in GF(2^m,)," *IEEE Trans. Comput.,* vol. C-34, pp. 709-717, Aug. 1985.

[4] K.C Smith. "The prospect for multivalued logic : A technology and applications view." *IEEE Trans. Comput.*, vol. C-30, pp. 619-634, Sept. 1981.

[5] S.L. Hurst, "Multiple-valued logic-its future," *IEEE Trans. Comput.*, vol. C-33, pp. 1161-1179, Dec. 1984.

[6] J.T. Butler, "Multiple-valued logic in VLSI," *IEEE Computer Soc. Press*, 1991.

[7] H. K. Seong and H.S. Kim, "A construction of celluar array multiplier over GF(2^m)," *KITE,* vol. 26, no. 4, pp. 81-87. April 1989.

[8] P. A. Scott, S.E. Tarvares and L.E. Peppard, "A fast multiplier for GF(2^m)," *IEEE J.Select. Areas Commun.*, vol. SAC-4, Jan. 1986.

[9] S. Bandyopadhyay and A. Sengupta, "Algorithms for multiplication in Galois field for implementation using systolic arrays," *IEE Proc.*, vol. 135. PT. E. no. 6, pp. 336-339, Nov. 1988.

[10] C.L. Wang and J.L. Lin, "Systolic array implementation of multipliers for finite fields

GF(2^m)," *IEEE Trans. Circuits and Systems*, vol. 38, no. 7, July 1991.

[11] J. T. Butler and H. G. Kerkhoff, "Multiple -valued CCD circuits," *IEEE Comput.*, pp. 58-67. Apr. 1988.

[12] M.H. Abd-El-Barr and Z. G. Vranesic, "Cost reduction in the CCD Realization of MVMT functions," *IEEE Trans. Comput.*, vol. C-39, no. 5, May 1990.

[13] Kiamal Z. Pekmestzi, "Multiplexer-Based Array Multipliers," *IEEE Trans. Compt.*, vol. 48, no. 1, pp.15-23, Jan. 1999.

[14] H.K. Seong and K.S. Yoon, "A Study on Implementation of Multiple-Valued Arithmetic Processer using Current Mode CMOS," *KITE*, vol. 36, no. C-4, pp. 35-45. Aug. 1999.

[15] C. K. Koc and B. Sunar, "Low-complexity bit-parallel canonical and normal basis multipliers for a class of finite fields," *IEEE Trans. Comput.*, vol. c-47, no. 3, pp. 353-356, March 1998.

[16] C. Paar, P. Felischmann, and P. Roelse, "Efficient multiplier architectures for Galois fields GF(2^{4n})," *IEEE Trans. Comput.*, vol. C-47, no. 2, pp. 162-170, Feb. 1998.

[17] J.H. Hwang, S.Y. Park, B.S. Shin, and H.S. Kim, "Multiplexer-Based Array Multipliers over $GF(2^m)$," *KITE*, vol. 37, no. SC-4, pp. 35-41. July 2000.

[18] B. Sunar and C.K. Koc, "Mastrovito Multiplier for All Trinomials," *IEEE Trans. Comput.*, vol. c-48, no. 5, pp. 522-527, May 1999.

[19] E. D. Mastrovito, "VLSI Architectures for Multiplication Over Finite Field $GF(2^m)$," *Applied Algebraic Algorithms, and Error-Correcting Code*, Proc. Sixth Int'l Conf., AAECC-6, T.Mora,ed., pp. 297-309, Rome, July 1988.

[20] E.D. Mastrovito, "VLSI Architectures for Computation in Galois Fields," PhD thesis, Linkoping Univ., Dept. of Electrical Eng., Linkoping, Sweden, 1991.

저 자 소개

黃 鍾 學(正會員) 第 37卷 SC編 第 4號 參照
현재 체육과학연구원 시스템개발팀 선임 연구원

崔 在 碩(正會員)
1964년 6월 8일생. 1998년 인하대학교 전자공학과 졸업. 1990년 인하대학교 전자공학과 대학원 졸업(공학석사). 1997년 인하대학교 전자공학과 대학원 졸업(공학박사). 1990년~1995년 기아정보시스템 연구소. 1996년~1998년 우진전자통신(주) 연구소. 1999년~현재 인덕대학 메카트로닉스과 전임강사. 주관심분야는 Multiple-Valued Logic. Logic Design. Microprocessor 응용 등임

沈 載 煥(正會員)
1976년 2월 인하대학교 전자공학과 졸업(공학사). 1982년 2월 숭실대학교 대학원 전자공학과 졸업(공학석사). 1996년 8월~1999년 8월 인하대학교 전자공학과 박사과정 수료. 현재 시립 인천 전문대학 통신과 교수. 주관심분야는 부호이론 등

金 興 壽(正會員) 第 37卷 SC編 第 4號 參照