論文2003-40SC-6-4

시스템 복잡도를 개선한 GF(2^m)상의 병렬 AB^2+C 연산기 설계

(Low System Complexity Bit-Parallel Architecture for Computing AB^2+C in a Class of Finite Fields GF(2^m))

卞 基 寧 *, 金 興 壽 **

(Gi-Young Byun and Heung-Soo Kim)

요 약

본 논문에서는 m차 기약 AOP를 적용하여 시스템 복잡도를 개선한 GF(2^m)상의 새로운 AB^2+C 연산기법과 그 하드웨어 구현회로를 제안하였다. 제안된 회로는 병렬 입출력 구조를 가지며, CS, PP 및 MS를 모듈로 하여 구성되며 이들은 각각 AND와 XOR 게이트의 규칙적인 배열구조를 갖는다. 제안된 회로의 시스템 복잡도는 $(m+1)^2$개의 2-입력 AND게이트와 $(m+1)(m+2)$개의 2-입력 XOR게이트의 회로복잡도와 연산에 소요되는 최대 지연시간은 $T_A+(1+\lceil \log_2 m \rceil)T_X$이다. 제안된 연산기의 시스템 복잡도와 구성상의 특징을 타 연산기를 표로 비교하였고, 그 결과 상대적으로 우수함을 보였다. 또한, 단순하면서도 정규화된 소자 및 결선의 구조는 VLSI 구현에 적합하다.

Abstract

This study focuses on the arithmetical methodology and hardware implementation of low-system-complexity AB^2+C operator over GF(2^m) using the irreducible AOP of degree m. The proposed parallel-in parallel-out operator is composed of CS, PP, and MS modules, each can be established using the array structure of AND and XOR gates. The proposed multiplier is composed of $(m+1)^2$ 2-input AND gates and $(m+1)(m+2)$ 2-input XOR gates. And the minimum propagation delay is $T_A+(1+\lceil \log_2 m \rceil)T_X$. Comparison result of the related AB^2+C operators of GF(2^m) are shown by table, it reveals that our operator involve more lower circuit complexity and shorter propagation delay then the others. Moreover, the interconnections of the our operators is very simple, regular, and therefore well-suited for VLSI implementation.

Keywords : finite field, all one polynomial, standard basis, GF(2^m) multiplier, power-sum

I. 서 론

유한체(Finite Field)는 Galois체, 또는 간단히 GF라 하며, 오류정정부호, 스위칭이론, 컴퓨터 구조 및 암호화 등의 분야에 적용되고 있는 연산체계이다[1, 2]. 유한체를 구성하는 원소들은 표준, 정규, 쌍대기저 등에 의해 각 형식에 따른 다항식으로 표현되며, 각 기저의 특성에 따라 연산의 효율성과 회로구현의 용이성이 달라진다[3]. 일반적으로 표준기저의 경우 타 기저에 비하여 기약다항식의 선택이 자유롭고, 호환성을 갖춘 범용 유한체 하드웨어의 구현이 용이한 장점이 있다.

* 正會員, 가톨릭大學校 情報通信電子工學部
(School of Information, Communication & Electronics Eng., Catholic Univ.)
** 正會員, 仁荷大學校 電子工學科
(Dept. of Electronic Eng., InHa Univ.)
接受日字:2003年8月11日, 수정완료일:2003年10月20日

표준기저를 적용한 다양한 유한체 연산들 중 가산과 승산은 여타 연산의 기반이 되는 연산으로 활용되고 있다. 특히 오류정정부호의 분야 중 이진 BCH 코드나 RS 코드의 복호과정에서 자주 발견되는 product-sum (AB+C) 연산이나, power-sum (AB^2+C) 연산은 유한체 가산 및 승산이 반복 적용되는 대표적 연산이다[4]. 유한체 가산 연산은 유한체 성질에 의해 가산 후 발생하는 자리올림은 고려되지 않으므로 매우 쉽고 단순하게 이루어진다. 그러나, 승산을 포함한 이외의 유한체 연산에서는 기약다항식에 의한 모듈러 환원의 과정이 수반되므로 보다 복잡하게 이루어진다. 따라서, 고속 및 대용량의 신호 처리능력과 함께 소형, 경량화와 저전력 특성으로 대변되는 VLSI의 구현을 위해 효율적인 연산 기법의 개발 및 최적화된 회로의 구현은 최근까지 관심의 대상이 되어왔다.

1971년 Laws[5]의 병렬 입출력 셀배열 승산기가 제안된 이후 구성소자의 수에 따른 회로복잡도와 연산지연시간으로 일컬어지는 시스템 복잡도의 개선을 위한 다양하고 많은 연구들이 진행되었다. 그 중 1989년 Itoh와 Tsujii[6]는 기약 AOP의 조건을 활용하여 시스템 복잡도를 개선한 GF(2^m) 병렬 승산기를 제안하였고, 이후 이 분야에 중요한 진전을 이룬 많은 연구가 있어 왔다[7, 8]. 유한체 승산구현을 위한 연구와 함께 효율적인 AB^2+C 연산회로를 개발하기 위한 연구가 진행되었고, 최근 Wei[9], Guo[10], Lee[11]등이 그 연산회로를 보였다.

이러한 연구동향을 토대로 본 논문에서는 기약 AOP를 적용하여 시스템 복잡도를 개선한 새로운 AB^2+C 연산기법과 구현회로를 제안하였다. 본 논문에서 제안한 회로는 $(m+1)^2$개의 2-입력 AND 게이트, $(m+1)(m+2)$개의 2-입력 XOR 게이트로 구성되며, 메모리나 스위치 같은 별도의 소자를 필요로 하지 않는다. 또한 입력된 신호로부터 최종 출력에 이르기까지 소요되는 연산지연시간은 $T_A+(1+\lceil \log_2 m \rceil)T_X$이다. 본 논문에서 제안한 회로는 회로의 정규성을 가지면서 기존의 연구에 비해 시스템 복잡도를 보다 개선하였다. 또한, AND와 XOR 게이트의 규칙적인 배열구조를 가지면서 각 게이트들의 배선구조 또한 동일성과 규칙성을 유지하므로 VLSI에 매우 유리하다.

본 논문의 구성을 간략히 소개하면 다음과 같다. Ⅰ장의 서론에 이어, Ⅱ장에서는 유한체의 성질을 간략히 논의하였고, 이를 토대로 기약 AOP를 적용한 새로운 GF(2^m)상의 승산전개 기법 및 AB^2+C 연산기법을 보였다. Ⅱ장의 논의를 바탕으로 Ⅲ장에서는 새로운 GF(2^m)상의 병렬 AB^2+C 연산기를 설계하였다. Ⅳ장에서는 본 논문과 타 논문의 구성을 각 항목별로 비교하였으며, 결론으로 본 논문의 끝맺음을 하였다.

Ⅱ. GF(2^m)상의 승산전개

1. 유한체상의 원소표현과 가산 연산

유한체 GF(2^m)[1, 2]은 양의 정수 m에 대하여 2^m개의 원소들로 구성된 수 체계이며, 그 원소들간의 연산이 사칙연산에 대하여 닫혀있다. GF(2^m)은 0과 1을 원소로 갖는 기초체 GF(2)를 m차원으로 확장한 확장체이며, GF(2^m)상의 모든 연산은 모듈로(modulo) 2 연산을 기반으로 이루어진다. 0을 제외한 GF(2^m)상의 모든 원소들은 원시원소 α에 의해 표현되며, α는 기약다항식 $F(x)=f_0+f_1x+\cdots+f_{m-1}x^{m-1}+x^m$의 근이 된다. 따라서, $F(\alpha)=0$이 되며, $\alpha^m=f_{m-1}\alpha^{m-1}+\cdots+f_1\alpha+f_0$이 성립한다. 이에 따라 GF($2^m$)상의 모든 원소들은 m보다 낮은 차수를 갖는 α의 다항식으로 표현되며 다항식의 각 기저들, α^{m-1}, …, α, $\alpha^0=1$을 표준기저라 한다.

표준기저를 적용하여 다항식으로 표현한 GF(2^m)상의 임의의 원소를 A라 할 때, 이는 $A=a_0+a_1\alpha+\cdots+a_{m-1}\alpha^{m-1}$과 같이 표현되며, 각 기저의 계수들, a_0, a_1, …, a_{m-1}은 모두 GF(2)상의 원소이다. 한편, GF(2^m)상의 원소 A의 각 계수들에 ⊕1을 취하여 $A_i(=a_i\oplus 1,\ 0\le i\le m-1)$를 정의하면 A는 식 (1)와 같이 표현된다.

$$A=A_0+A_1\alpha+\cdots+A_{m-1}\alpha^{m-1}+A_m\alpha^m \quad (1)$$

식 (1)에서 사용된 기저들, $\{\alpha^m, \alpha^{m-1}, \cdots, \alpha, 1\}$을 표준기저의 확장기저라 하며, $A_m=1$이다.

확장기저로 표현된 GF(2^m)상의 또 다른 원소를 $B=B_0+B_1+\cdots+B_{m-1}\alpha^{m-1}+B_m\alpha^m$로 표현할 때, 두 유한체 원소들의 가산은 식 (2)와 같다.

$$A+B=(A_0\oplus B_0)+(A_1\oplus B_1)\alpha+\cdots+(A_{m-1}\oplus B_{m-1})\alpha^{m-1}+(A_m\oplus B_m)\alpha^m \quad (2)$$

식 (2)에서 사용한 ⊕는 모듈러 가산의 기호이며, 그 결과는 GF(2)상의 원소가 된다. 본 논문에서는 +는 선형결합의 의미로 ⊕와 구분하여 사용하였다.

식 (2)와 같이 GF(2^m)상의 가산연산은 가산 후 발생

하는 자리올림을 고려하지 않으므로 매우 쉽고 간단하게 이루어지나, 승산을 비롯한 이외의 연산에서는 기약다항식을 적용한 모듈러 환원의 과정이 적용되므로 그 연산과정이 보다 복잡하게 이루어진다.

2. 기약 AOP과 순환이동

모든 계수가 1인 다항식을 AOP(All One Polynomial)라 하며, 그 중 $m+1$이 소수가 되는 GF(2^m)상의 기약다항식들을 기약 AOP[6]라 한다. 이에 해당하는 m은 2, 4, 10, 12, 18, 28, 36, 52, 58, 60, 66, 82, 100, … 등이다.

유한체와 기약 AOP의 성질로부터 GF(2^m)상의 원소 $\alpha^m=\alpha^{m-1}+\alpha^{m-2}+\cdots+\alpha+1$로 표현되며, 양의 정수 i에 대하여 $\alpha^{m+i}=\alpha^{i-1}$이 성립한다.

예제 1. $\alpha^m=\alpha^{m-1}+\alpha^{m-2}+\cdots+\alpha+1$과 $\alpha^{m+i}=\alpha^{i-1}$의 관계식을 적용하여, 식 (1)과 같이 확장기저로 표현된 GF(2^m)상의 원소 A에 α를 누승한 결과를 보이면 다음과 같다.

$$A=A_0+A_1\alpha+\cdots+A_{m-1}\alpha^{m-1}+A_m\alpha^m$$
$$\alpha A=A_0\alpha+A_1\alpha^2+\cdots+A_{m-1}\alpha^m+A_m\alpha^{m+1}$$
$$=A_m+A_0\alpha+A_1\alpha^2+\cdots+A_{m-1}\alpha^m$$
$$\alpha^2A=A_m\alpha+A_0\alpha^2+\cdots+A_{m-2}\alpha^m+A_{m-1}\alpha^{m+1}$$
$$=A_{m-1}+A_m\alpha+A_0\alpha^2+\cdots+A_{m-2}\alpha^m$$
$$\vdots$$
$$\alpha^iA=A_{m-i+1}+A_{m-i+2}\alpha+\cdots+A_{m-i}\alpha^m$$

예제 1에서 기약 AOP를 모듈러 환원에 적용할 때, A에 α를 누승한 결과는 A의 각 계수들이 누승과 동일한 횟수만큼 상위차수로 이동하며, $m+1$ 차수의 계수는 최저차수로 순환 이동함을 알 수 있다.

정의 1[8]. 확장기저를 적용한 GF(2^m)상의 임의의 원소 A에 대하여 계수들의 순환이동을 식 (3)으로 정의한다.

$$^{(1)}A=A_m+A_0\alpha+\cdots+A_{m-2}\alpha^{m-1}+A_{m-1}\alpha^m \quad (3)$$

이러한 순환이동을 i번(i는 양의 정수) 실행하면 A의 각 계수들은 상위 차수로 i번 이동되며, 이를 $^{(i)}$A로 표현하기로 한다.

정리 1. 예제 1과 정의 1로부터 확장기저로 표현된 A에 대한 α의 누승을 연산식과 순환이동으로 표현하면 식 (4)와 같다.

$$\alpha^iA=A_{\langle m-i+1\rangle}+A_{\langle m-i+2\rangle}\alpha+\cdots+A_{\langle m-i\rangle}\alpha^m$$
$$=\sum_{j=0}^{m}A_{\langle j-i\rangle}\alpha^j,\ i=0,1,\cdots,m$$
$$=^{(i)}A \quad (4)$$

식 (4)에서 $A_{<\theta>}$의 아래첨자 $<\theta>$는 $m+1$에 대한 모듈러 연산의 결과로 $0\le<\theta>\le m$인 양의 정수이다. 또한, 기호의 표현에 있어 $^{(0)}$A=A이다.

(증명) 정리 1의 증명을 위해 확장기저로 표현된 GF(2^m)상의 원소 A에 α를 승산한 결과는 식 (5)와 같다.

$$\alpha A=A_0\alpha+A_1\alpha^2+\cdots+A_{m-1}\alpha^m+A_m\alpha^{m+1}$$
$$=A_m+A_0\alpha+A_1\alpha^2+\cdots+A_{m-1}\alpha^m \quad (5)$$

기약 AOP를 적용한 모듈러 환원에서 $\alpha^{m+1}=\alpha^0=1$이므로, $\alpha A=^{(1)}A$이 된다. 이와 동일하게 A에 α^i을 승산한 후 이를 정리하면, 결국 A의 각 계수들이 순환이동으로 표현되며 그 결과를 정리하면 식 (4)와 같다.

3. GF(2^m) 상의 AB^2+C 연산

AB^2+C 연산기법을 논의하기에 앞서, 확장기저로 표현된 GF(2^m)상의 원소 B의 제곱은 유한체의 성질에 의해 식 (6)과 같다.

$$B^2=(B_0+B_1\alpha+\cdots+B_{m-1}\alpha^{m-1}+B_m\alpha^m)^2$$
$$=B_0+B_1\alpha^2+\cdots+B_{m-1}\alpha^{2(m-1)}+B_m\alpha^{2m} \quad (6)$$

본 논문에서는 식 (6)의 B^2과 A의 승산, $P=AB^2$을 식 (7)과 같이 전개하였다.

$$P=AB^2=\left(\sum_{l=0}^{m}Al\alpha l\right)\left(\sum_{k=0}^{m}B_k\alpha^{2k}\right)$$
$$=\sum_{k=0}^{m}Bk\left(\sum_{l=0}^{m}A_l\alpha^l\right)\alpha^{2k}$$
$$=\sum_{k=0}^{m}Bk\left(\sum_{j=0}^{m}A_{\langle j-2k\rangle}\alpha^j\right) \quad (7)$$

식 (7)의 연산을 정리 1에서 논의한 순환이동을 적용하여 표현하면 식 (8)과 같다.

$$P=AB^2=\sum_{k=0}^{m}B_k(^{(2k)}A) \quad (8)$$

식 (7)을 전개하여 유도한 P의 각 계수들, P_0, P_1, …,

P_m을 A와 B의 계수들로 나타내면 식 (9)와 같다.

$$P_k = \sum_{n=0}^{m} B_n A_{\langle k-2n \rangle} \quad (9)$$

식 (9)에서 k는 $0 \le k \le m$의 범위를 갖는 정수이다.

지금까지의 논의를 토대로 AB^2+C연산을 $GF(2^4)$상의 예로 보이면 예제 1과 같다.

예제 1. $GF(2^4)$상의 두 원소 A와 B에 대하여, $A=A_0+A_1\alpha+A_2\alpha^2+A_3\alpha^3+A_4\alpha^4$와 $B^2=B_0+B_1\alpha^2+B_2\alpha^4+B_3\alpha^6+B_4\alpha^8$로 나타내었다. 이때, $P=AB^2=P_0+P_1\alpha+P_2\alpha^2+P_3\alpha^3+P_4\alpha^4$를 식 (7)에 적용하여 전개하면 식 (10)과 같다.

$$\begin{aligned}
P &= \sum_{k=0}^{4} B_k \left(\sum_{j=0}^{4} A_{\langle j-2k \rangle} \alpha^j \right) \\
&= B_0(A_{\langle 0-0 \rangle} \oplus A_{\langle 1-0 \rangle}\alpha \oplus A_{\langle 2-0 \rangle}\alpha^2 \oplus A_{\langle 3-0 \rangle}\alpha^3 \oplus A_{\langle 4-0 \rangle}\alpha^4) \\
&+ B_1(A_{\langle 0-2 \rangle} \oplus A_{\langle 1-2 \rangle}\alpha \oplus A_{\langle 2-2 \rangle}\alpha^2 \oplus A_{\langle 3-2 \rangle}\alpha^3 \oplus A_{\langle 4-2 \rangle}\alpha^4) \\
&+ B_2(A_{\langle 0-4 \rangle} \oplus A_{\langle 1-4 \rangle}\alpha \oplus A_{\langle 2-4 \rangle}\alpha^2 \oplus A_{\langle 3-4 \rangle}\alpha^3 \oplus A_{\langle 4-4 \rangle}\alpha^4) \\
&+ B_3(A_{\langle 0-6 \rangle} \oplus A_{\langle 1-6 \rangle}\alpha \oplus A_{\langle 2-6 \rangle}\alpha^2 \oplus A_{\langle 3-6 \rangle}\alpha^3 \oplus A_{\langle 4-6 \rangle}\alpha^4) \\
&+ B_4(A_{\langle 0-8 \rangle} \oplus A_{\langle 1-8 \rangle}\alpha \oplus A_{\langle 2-8 \rangle}\alpha^2 \oplus A_{\langle 3-8 \rangle}\alpha^3 \oplus A_{\langle 4-8 \rangle}\alpha^4) \\
&= B_0(A_0 + A_1\alpha + A_2\alpha^2 + A_3\alpha^3 + A_4\alpha^4) \\
&+ B_1(A_3 + A_4\alpha + A_0\alpha^2 + A_1\alpha^3 + A_2\alpha^4) \\
&+ B_2(A_1 + A_2\alpha + A_3\alpha^2 + A_4\alpha^3 + A_0\alpha^4) \\
&+ B_3(A_4 + A_0\alpha + A_1\alpha^2 + A_2\alpha^3 + A_3\alpha^4) \\
&+ B_4(A_2 + A_3\alpha + A_4\alpha^2 + A_0\alpha^3 + A_1\alpha^4) \quad (10)
\end{aligned}$$

식 (10)의 결과에 대하여 식 (9)을 적용하여 P의 각 기저들에 대하여 표현하면 식 (11)과 같다.

$$\begin{aligned}
P_0 &= B_0A_0 \oplus B_1A_3 \oplus B_2A_1 \oplus B_3A_4 \oplus B_4A_2 \\
P_1 &= B_0A_1 \oplus B_1A_4 \oplus B_2A_2 \oplus B_3A_0 \oplus B_4A_3 \\
P_2 &= B_0A_2 \oplus B_1A_0 \oplus B_2A_3 \oplus B_3A_1 \oplus B_4A_4 \\
P_3 &= B_0A_3 \oplus B_1A_1 \oplus B_2A_4 \oplus B_3A_2 \oplus B_4A_0 \\
P_4 &= B_0A_4 \oplus B_1A_2 \oplus B_2A_0 \oplus B_3A_3 \oplus B_4A_1 \quad (11)
\end{aligned}$$

power-sum 연산을 위해 식 (11)의 결과에 또 하나의 유한체 원소 C를 가산 연산하여 최종 AB^2+C 연산을 이룬다. 서론에서 논의한 바와 같이 유한체 가산은 가산 후 발생하는 자리올림을 고려하지 않으므로 식 (11)의 각 Pk에 C의 각 계수들을 모듈러 가산함으로써 AB^2+C 연산을 완성할 수 있다.

III. $GF(2^m)$상의 AB^2+C 연산기 구성

앞에서 논의한 바와 같이, 본 논문에서는 AOP를 기반으로 하여 확장기저로 표현된 $GF(2^m)$상의 두 원소 A와 B에 대하여 AB^2 연산을 피 승산항(A)의 순환이동과 승산항(B^2)의 각 계수를 순차적이고 반복적으로 승산한 후 동일 차수의 계수들을 모듈러 가산함으로써 이루었다. 이러한 승산전개의 회로구현을 위해 피 승산항의 계수들에 대한 순환이동(Cyslic Shift, CS) 연산모듈과 그 결과에 승산항의 각 계수들을 승산하는 부분곱(Partial Product, PP) 연산모듈, 그리고 동일한 차수의 계수들과 C를 모듈러 가산하는 모듈러 가산(Modular Summation, MS) 연산모듈들이 각각 필요하며, 연산회로의 구성을 <그림 1>에 도시하였다.

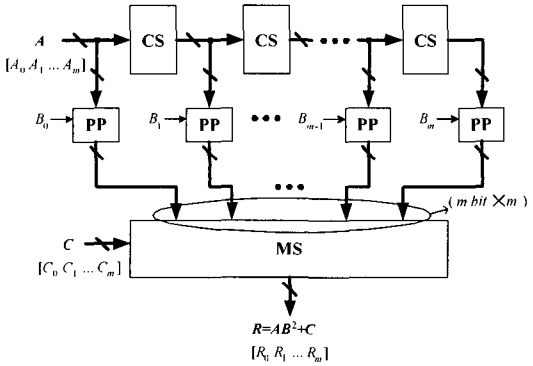
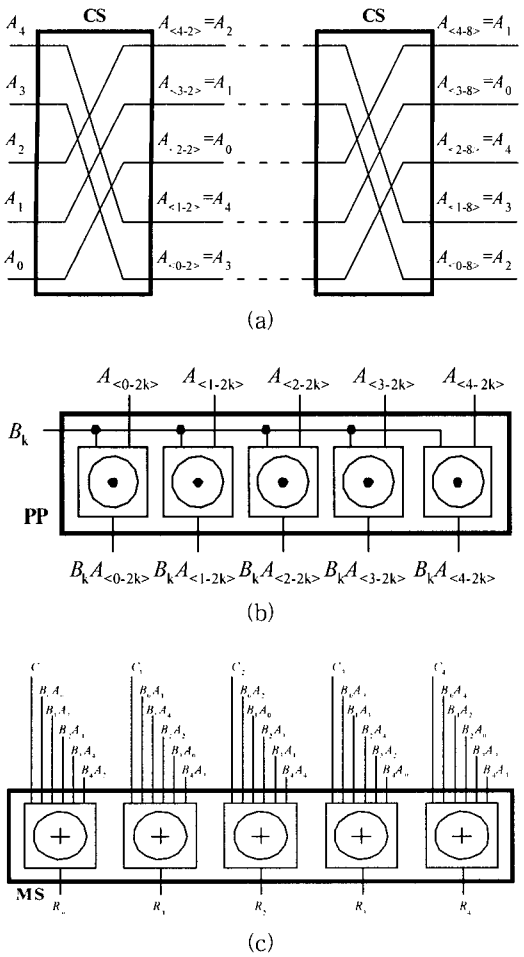


그림 1. 제안된 $GF(2^m)$ 병렬 승산기의 구성도
Fig. 1. Block diagram of proposed $GF(2^m)$ parallel multiplier.

<그림 1>에서 굵은선 화살표는 m비트 병렬 신호흐름을 나타낸다. 예제 1에서 논의한 $GF(2^4)$상의 두 원소들에 대한 AB^2 연산회로를 구현하기 위해 필요한 CS, PP, MS 연산모듈을 각각 <그림 2>에 보였다.

<그림 2(a)>의 CS 연산모듈은 게이트를 사용하지 않고 결선에 의해서만 이루어지므로 연산에 필요한 게이트 및 지연시간은 없다. (b) PP 연산모듈은 승산항의 계수 B_k와 피 승산항들의 계수들, $A_0, \cdots A_m$이 부분곱을 이루는 연산모듈로 2-입력 AND 게이트를 m개 배열함으로써 구현될 수 있다. 본 논문에서는 AND 게이트를 ⊙로 기호화하였고, 2-입력 AND 게이트에서 발생하는 지연시간을 T_A라 하였다. 단위 PP 연산모듈의 회로복잡도는 $(m+1)$개의 2-입력 AND 게이트이며, 지



(a)

(b)

(c)

그림 2. 각 연산 모듈, (a) CS, (b) PP, (c) MS
Fig. 2. Each operational modules, (a) CS, (b) PP, and (c) MS.

연시간은 T_A이다. (c) MS 연산모듈은 각 PP 연산모듈로 부터 연산된 AB^2의 결과들 중 동일 차수의 계수들과 C에 대한 모듈러 가산을 이루는 블록으로 XOR 게이트들을 배열함으로써 구현될 수 있다. 본 논문에서는 XOR 게이트 ⊕로 기호화하였고, 2-입력 XOR 게이트에서 발생하는 지연시간을 T_X라 하였다. 간략화된 표현을 위해 <그림 2(c)>에서는 6-입력 XOR로 표현하였다. 타 승산기와의 비교를 위해 2-입력 XOR로 구현하면, $(m+1)(m+2)$개의 게이트가 필요하며, MS 연산블록에서 발생하는 최소 지연시간은 $(1+\lceil \log_2 m \rceil)T_X$이다. 이를 종합하여 본 논문에서 제안한 GF(2^m)상의 병렬 승산 회로는 $(m+1)$2개의 2-입력 AND 게이트와 $(m+1)(m+2)$개의 2-입력 XOR가 사용되며, 입력된 신호로부터 연산과정을 거쳐 최종 신호가 출력되기까지의 지연시간은 $T_A+(1+\lceil \log_2 m \rceil)T_X$ 이다.

Ⅳ. 비교 및 검토

GF(2^m)상의 AB^2+C 연산회로에 대하여 본 논문과 타 논문에 대한 각 항목별 비교와 고찰을 하였고, 그 결과를 <표 1>에 정리하였다.

① 연산회로의 구조(Architecture)

Wei[9], Wang[10], Lee[11]의 연산회로에서 채택한 시스토릭 구조는 일반적으로 이는 매우 큰 m 상의 연산시 소자들에 의해 발생하는 전파지연시간의 축적을 방지하고, 파이프라인 연산이 가능하므로 고속 및 대용량의 연산시스템에 유리한 특성을 갖는다. 그러나, 별도의 메모리 소자와 그 동기신호가 필요하므로 소자의 수와 그 결선의 면적이 증가되어 시스템 복잡도의 개선에 불리한 단점을 갖는다. 본 논문에서 논의한 새로운 AB^2+C 연산기법 및 그 구현회로는 회로의 소자수 및 연산시 발생하는 지연시간을 고려할 때 시스토릭 구조에 비해 시스템 복잡도 개선이 우월하다 할 수 있다.

표 1. GF(2^m)상의 AB^2+C 연산기의 구성 비교
Table 1. Comparisons of the related multipliers for computing AB^2+C over GF(2^m).

Item \ Multiplier		Wei[9]	Wang[10]	Lee[11]	proposed
Architecture		systolic	systolic	systolic	non-systolic
Generating polynomial		General polynomial	General polynomial	Irreducible AOP	Irreducible AOP
Basis used		Standard	Standard	Extended Standard	Extended Standard
circuit complexity	No. of 2-input AND	$3m^2$	$3m^2$	$(m+1)^2$	$(m+1)^2$
	No. of 2-input XOR	$3m^2$	$2m^2$	$(m+1)^2$	$(m+1)(m+2)$
	1-bit latch	$13m^2$	$(17/2)m^2$	$3(m+1)^2$	-
Latency (unit=clock cycles)		$4m$	$(5/2)m$	$m+1$	-
Propagation delay through on cell		$T_A+2T_X+T_L$	$T_A+3T_X+T_L$	$T_A+T_X+T_L$	$T_A+(1+\lceil \log_2 m \rceil)T_X$
Note :		T_A, T_X, and T_L are the propagation delay of one 2-input AND gate, 2-input XOR gate, and 1-bit latch, respectively.			

② 기약 다항식(Generating polynomial)

유한체 연산시 모듈러 환원의 과정에서 적용되는 기약다항식 또는 생성다항식의 경우, Wei와 Wang은 m에 대한 일반적인 기약다항식의 적용이 가능한 반면

Lee와 본 논문에서는 기약 AOP로 제한하였다.

③ 적용 기저(Basis used)

서론에서 논의한 바와 같이 유한체 연산기에 적용되는 기저 표현 기법들 중 Wei와 Wang은 표준기저를, Lee와 본 논문에서는 표준기저의 각 계수들에 ⊕1을 취하여 새롭게 정의한 확장기저를 각각 채택하였다.

④ 회로복잡도 - AND 게이트

일반적으로 k개의 input을 갖는 XOR 소자는 $(k-1)$개의 2-input XOR 소자로 대치될 수 있으며 본 논문에서는 비교의 단순성과 명료성을 위해 모든 AND와 XOR 게이트를 2-input 소자를 기준하여 계수하였다.

Wei의 회로에서는 하나의 단위 셀에 3개의 AND 게이트와 3개의 XOR 게이트와 13개의 1-bit latch가 사용된다. AB^2+C 연산회로의 완성을 위해 m^2개의 셀이 사용되므로 Wei의 회로가 갖는 회로 복잡도는 <표 1>에 보인 바와 같다. Wang의 회로에서는 하나의 단위셀에 6개의 AND 게이트와 6개의 XOR 게이트와 17개의 1-bit latch가 사용된다. 이러한 단위셀이 $m^2/2$개 사용되므로 Wang의 회로가 갖는 회로 복잡도는 <표 1>에 보인 바와 같다. Lee의 회로에서는 하나의 단위 셀에 각각 1개씩의 AND와 XOR 게이트가 사용되며, 3개의 1-bit latch가 사용된다. 전체회로의 완성을 위해 $(m+1)^2$개의 단위셀이 사용되며 그 회로 복잡도는 <표 1>에 보인 바와 같다. 본 논문에서 제안한 연산회로는 각각 $(m+1)^2$개의 AND와 XOR 게이트가 사용되며, 이외의 메모리소자를 필요로 하지 않는다.

⑤ 연산지연시간(Latency & Propagation Delay)

Wei, Wang, Lee의 회로는 시스토릭 구조를 이므로 각 셀 내부의 소자에 의해 발생하는 전파지연시간과 셀들간의 연산 동기시간을 제어하기 위해 필요한 Latency가 필요하다. 본 논문에서 제안한 회로에서는 메모리 소자를 사용하지 않으므로 제어신호 및 Latency가 필요하지 않다. 또한, XOR에 의해 발생하는 전파지연시간의 경우 $\log_2 m$의 지연시간을 가지므로 상당히 큰 m에 대해서도 그 지연시간이 크지 않다. 따라서, 전파지연시간과 Latency를 함께 고려할 때 매우 빠른 연산시간을 갖는다 할 수 있다.

V. 결 론

본 논문에서는 $GF(2^m)$상의 새로운 AB^2+C 연산기법 및 그 연산회로를 제안하였다. 이를 위해 유한체 원소를 확장기저로 표현하였고, 모듈러 환원시 기약 AOP가 갖는 특성을 도출하여 순환이동 특성을 적용한 멱승 및 승산전개기법을 보였다. 제안된 연산전개 기법으로부터 회로구성을 위해 CS, PP, MS 연산모듈들을 정의하였고, 이들로부터 $GF(2^m)$ AB^2+C 연산기를 설계할 수 있음을 $GF(2^4)$의 예로 보였다. 본 논문에서 제안한 연산기를 회로의 구성소자 수와 지연시간 등으로 타 연산기와 비교하였고, 그 결과 시스템 복잡도의 개선에 보다 유용함을 확인하였다. 또한, AND 또는 XOR로만 구성된 배열구조와 각 게이트들을 연결하는 동일한 배선구조, 그리고 m의 증가에 따른 각 연산모듈의 규칙적인 증가가 갖는 정규성은 VLSI 회로구현에 매우 유리하다 할 수 있다.

참 고 문 헌

[1] S.Lin, Error Control Coding, Prentice-Hall, Inc. New Jersey, 1983.

[2] 이만영, BCH부호와 Reed-Solomon부호, 민음사, 1990.

[3] I.S.Hsu, T.K.Troun, L.J.Deutsch, and I.S.Reed, "A Comparison of VLSI Architecture of Multipliers using Dual, Normal, or Standard Bases," IEEE Trans. Comput., vol. C-37, pp. 735-739, 1988.

[4] H. Okano and H. Imai, "A Construction method of high-speed decoders using ROM's for Bose-Chaudhuri-Hocquenghem and Reed-Solomon codes," IEEE Trans. Comput., vol. C-36, pp. 1165-1171, 1987.

[5] B.A.Laws and C.K.Rushford, "A Cellular-Array Multiplier for $GF(2^m)$" IEEE Trans. Comput., vol. C-20, no. 12, pp. 1573-1578, Dec. 1971.

[6] T.Itoh, and S.Tsujii, "Structure of Parallel Multipliers for a Class of Fields $GF(2^m)$," Information and Computation, vol. 83, pp. 21-40, 1989.

[7] M.A.Hasan, M.Z.Wang, and V.K.Bhargava, "Modular Construction of Low Complexity Parallel Multipliers for a Class of Finite Fileds $GF(2^m)$," IEEE Trans. Comput., vol. 41, no. 8,

pp. 962-971, Aug. 1992.

[8] C.Y.Lee, E.H.Lu, and J.Y.Lee, "Bit-Parallel Systolic Multipliers for GF(2^m) Fields Defined by All-One and Equally Spaced Polynomials," IEEE Trans. Comput., vol. 50, no.5, pp.385-393, May 2001.

[9] S.W.Wei, "A Systolic power-sum circuit for GF(2^m)," IEEE Trans. Comput., vol. 43, pp. 226-229, Feb. 1994.

[10] C.L.Wang and J.H.Guo, "New systolic arrays for C+AB2, inversion, and division in GF(2^m)," IEEE Trans. Comput., vol. 49, pp. 1120-1125, Oct. 2000.

[11] C.Y.Lee, E.H.Lu, and L.F.Sun, "Low-Complexity Bit-Parallel Systolic Architecture for Computing AB^2+C in a class of Finite Field GF(2^m)," IEEE Trans. Circuit & Systems-II: Analog and Digital Signal Processing, vol. 48, no. 5, pp. 519-523, May 2001.

저 자 소 개

卞 基 寧(正會員)

1988년 3월~1994년 2월 : 인하대학교 전자공학과 공학사. 1996년 3월~1998년 8월 : 인하대학교 전자공학과 공학석사. 1999년 3월~2003년 2월 : 인하대학교 전자공학과 공학박사. 1994년 1월~1996년 8월 : (주)LG전자 VCR사업부 회로설계연구원. 2003년 3월~현재 : 가톨릭대학교 정보통신 전자공학부 강의전담교수. 현재 : IEEK, KICS 정회원. <주관심분야 : 정보이론, 부호이론, 논리시스템설계, 컴퓨터 구조, 유한체이론의 응용 및 VLSI 회로구현 등>

金 興 壽(正會員) 第40卷 SC編 第3號 參照

현재 : 인하대학교 전자공학과 교수