$GF(2^m)$상의 AOP 기반 비-시스토릭 병렬 AB^2+C 연산기

황 운 택*

A Base AOP Bit-Parallel Non-Systolic for AB^2+C Computing Unit for $GF(2^m)$

Woon-taek Hwang*

이 논문은 2005년도 시립 인천전문대학 교내학술연구비 지원에 의해 연구되었음

요 약

본 논문은 $GF(2^m)$상의 m차 기약 AOP를 적용하여 비-시스토릭 병렬 AB^2+C 연산기를 제안한다. 본 논문에서 제안한 연산기 회로는 AND게이트와 EX-OR게이트만을 사용하여 설계되어지며, 설계된 회로는 기약 AOP의 특성을 이용하여 게이트를 사용하지 않고 결선으로만 연결되어 게이트 및 지연시간이 없는 순환이동과, m개의 AND 게이트와 m개의 EX-OR게이트를 필요로 하는 승산연산, EX-OR게이트로만 구성되어지는 멱승연산, 승산연산과 멱승연산을 이용한 파워섬연산 및 가산연산 등이 사용된다. 제안된 연산기법은 AND게이트와 EX-OR게이트만을 사용함으로 고속의 데이터 처리, 저전력 및 집적화 등의 장점이 있으며, $T_A+(1+\lceil \log_2^m \rceil)T_X$의 연산 지연시간을 갖는다.

ABSTRACT

This paper proposes a non-systolic parallel AB^2+C computing unit based on irreducible AOP order m of $GF(2^m)$. Proposed circuit have only AND gates and EX-OR gates, composes of cyclic shift operation, multiplication operation, power operation, power-sum operation and addition operation using a property irreducible AOP. Suggested operating a method have an advantage high speed data processing, low power and integration because of only needs AND gates and EX-OR gates. AB^2+C computing unit has delay-time of $T_A+(1+\lceil \log_2^m \rceil)T_X$.

키워드

AB^2+C, Cyclic Shift, Systolic, AOP, Power-sum

* 시립 인천전문대학

접수일자 : 2006. 7. 19

Ⅰ. 서 론

원소의 개수가 유한한 유한체 $GF(2^m)$은 2진수 수체계의 효용성과 확장성에 기인하여 1960년대부터 오늘날까지 지속적으로 연구개발 되어지고 있으며 통신채널 및 저장매체에서 발생하는 오류를 정정하기 위한 오류 정정 회로로부터 컴퓨터 메모리, 디지털 레이더 신호처리, 디지털 보안 및 서명, 디지털 워터 마킹 등 다양한 영역에서 상업적으로 널리 응용되고 있다[1-3].

유한체를 구성하는 원소들은 기저(Basis)의 구분에 따라 각각 표준기저, 정규기저, 쌍대기저로 구분되어지며 다항식 또는 벡터 형식으로 표현된다. 각 기저의 특성에 따라 연산별 효용성과 회로구현의 용이성이 달라지는데 일반적으로 표준기저의 경우 타 기저에 비하여 기약 다항식의 선택이 자유롭고, 하드웨어 구현이 용이한 장점이 있다[4-6]. 표준기저를 이용한 $GF(2^m)$상의 연산에서 중요하게 다루어지는 연산으로는 가산, 승산, 제산 및 승산에 대한 역원, 멱승 등이 있다.

유한체 연산에 대한 연구는 날로 대용량화, 고속화 및 소형화되고 있는 산업 환경에 맞추어 전체 시스템의 규모와 성능에 절대적인 영향을 미치는 회로경로의 연결이나 시스템 구조의 복잡성, 동시성 등의 문제점을 고려한 최적화된 연산기법과 회로를 개발하기 위한 연구가 지속되고 있다.

본 논문에서는 표준기저를 적용한 $GF(2^m)$상의 병렬 AB^2+C연산기들 중에서 AOP(All one Polynomial) 특성을 갖는 연산기법 및 회로 구현의 시스템 복잡도 개선에 관하여 논의한다. 이에 관련한 연구 동향을 간략히 살펴보면, 1984년 Yeh[5] 등은 대용량 신호처리를 위하여 표준기저를 근거로 $GF(2^m)$상의 $AB+C$ 시스토릭 구조의 승산 회로를 제안하였다. 시스토릭 구조란 메모리로부터 출력된 데이터들이 연산기의 각 처리부를 통과하며 연산된 후 메모리에 저장되는 구조로 정규화 되고 모듈형태를 가지므로 VLSI구조에 유리하다 할 수 있다. 또한, 2001년 Lee[8] 등은 $GF(2^m)$상에서 기약 AOP을 적용한 AB^2+C연산을 수행하는 병렬 입출력 시스토릭 구조를 갖는 승산기를 개발하여 시스템 복잡도를 개선하였다. 제안된 병렬 입출력 AB^2+C시스토릭 승산기법은 구조가 단순하고, 규칙성이 있으며 감소된 지연시간과 전파지연시간을 갖는 승산기를 설계할 수 있다.

이와 같은 승산기의 예를 종합하면, 이들은 각각 $GF(2^m)$의 특성을 만족하는 고유한 회로설계 알고리즘과 회로구성으로 그 효용성을 입증 받았으며, 보다 개선된 회로구현을 위한 연구는 계속될 것으로 전망된다.

본 논문에서는 AOP(All One Polynomial)의 정의를 통해 기약 AOP를 적용한 다항식간의 가산연산, 승산연산, 멱승연산, 파워섬 연산을 정의하고 각각의 정의되어진 연산을 $GF(2^4)$상의 회로로 구현하며 Verilog-HDL을 이용하여 구현한 회로를 검증한다.

Ⅱ. $GF(2^m)$상의 연산

2.1. 순환이동과 기약 AOP[7-9]

유한체 $GF(2^m)$은 0과 1만을 원소로 하는 2^m개의 원소들로 구성된 수 체계이며, 기초체 $GF(2)$로부터 m차원 확장한 확장체 이므로 모든 연산은 모듈로(modulo) 2 연산을 기반으로 수행된다. 0을 제외한 $GF(2^m)$상의 모든 원소들는 원시원소(Primitive Element) α의 멱승으로 표현되며, α가 차수가 m인 원시 기약 다항식 $p(x)=x^m+f_{m-1}x^{m-1}+\cdots+f_1x+f_0$의 근 이라고 할 때, $\{\alpha^k|0\le k\le m-1\}$은 $GF(p^m)$상의 기저집합이 된다. 이 기저를 $GF(p^m)$상의 표준기저라 하고 표준기저를 사용하여 모든 원소들을 표현할 수 있다.

$GF(2^m)$의 원소들 α^i, $\{1, \alpha^1, \alpha^2, \cdots, \alpha^{m-1}\}$에 대한 모듈러 연산의 결과는 $m-1$이하의 차수를 갖는 다항식이 되며 식 (1)과 같다.

$$A(\alpha)=a_{m-1}\alpha^{m-1}+\cdots+a_1\alpha^1+a_0 \quad (1)$$

식 (1)에서 $\alpha^{m-1}, \cdots, \alpha^1$, 1들은 다항식을 이루는 기저가 되며, 각 기저들의 계수 $a_{m-1}, a_{m-2}, \cdots, a_1, a_0$들은 모두 $GF(2)$의 원소이다. 식 (1)에서 표현한 계수들에 대한 확장기저는 $A_i=a_i\oplus 1$을 취함으로써 식 (2)와 같이 표현 될 수 있다.

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

식 (2)에서 확장된 기저들, $\{\alpha^m, \cdots, \alpha^1, 1\}$로부터

$GF(2^m)$상의 승산에서 유용한 순환이동 특성을 도출할 수 있으며 이를 정의 1에 나타내었다.

정의 1. $GF(2^m)$상의 원소 $A(\alpha)$에 확장 기저를 적용시키면 각 기저의 계수들의 순환이동을 식(3)과 같이 정의할 수 있다.

$$A^{(1)}(\alpha) = A_{m-1}\alpha^m + \cdots + A_0\alpha^0 + A_m \tag{3}$$

식(3)은 1회의 순환이동이 일어난 것을 의미하며 순환이동을 m번 실행하면, $A^{(m)}(\alpha)$, 식(3)의 계수들은 우측으로 m번 순환이동이 된다.

앞서 살펴본 모든 식에서와 같이 다항식의 모든 계수가 1인 다항식을 AOP(All One Polynomial)라고 하며, AOP 범주 안에 $m+1$이 소수가 되는 2, 4, 10, 12, 18, 26, 28, 36, 52, 58, 60, 66, 82, 100, ...등을 $GF(2^m)$상의 기약 AOP라고 한다. 기약 AOP의 성질로부터 $GF(2^m)$상의 모듈러 환원의 성질을 유도할 수 있으며, 식(4)와 같다.

$$\alpha^m = \alpha^{m-1} + \alpha^{m-2} + \cdots + \alpha^1 + 1 \tag{4-a}$$

$$\alpha^{m+i} = \alpha^{i-1} \tag{4-b}$$

식(4)는 확장기저로 표현된 $A(\alpha)$에서 α의 멱승연산 전개에 유용한 특성을 제공한다.

2.2. AOP를 적용한 다항식의 가산연산

유한체상의 연산들 중 가산연산은 연산의 형태가 가장 간략하며 다른 유한체 연산에 공통적으로 적용된다. 모듈러 연산의 정의에 의해 유한체 가산은 연산 후 발생하는 자리올림을 고려하지 않는다. 따라서, 표준기저로 표현된 원소들의 가산연산은 동일한 차수를 갖는 기저의 계수들만의 모듈러 가산으로 이루어진다.

$GF(2^m)$상에서 기약 AOP를 적용한 두 다항식 $A(\alpha)$, $B(\alpha)$가 각각 $A(\alpha)=\sum_{l=0}^{m} A_l\alpha^l$, $B(\alpha)=\sum_{k=0}^{m} B_k\alpha^k$ 일 때, 다항식 $A(\alpha)$와 $B(\alpha)$의 가산은 식 (5)과 같다.

$$A(\alpha)+B(\alpha) = \sum_{l=0}^{m} A_l\alpha^l + \sum_{k=0}^{m} B_k\alpha^k = \sum_{n=0}^{m} (A_n + B_n)\alpha^n \tag{5}$$

일반적으로 유한체 연산에서 + 기호는 각 기저들간의 선형결합을 나타내며, $\oplus$ 기호는 각 계수들 간의 모듈러 가산연산을 나타낸다. 모듈러 연산의 정의에서 유한체의 가산은 연산 후 발생되는 자리 올림을 고려하지 않기 때문에 표준기저로 표현된 유한체 원소들 간의 가산은 동일한 차수를 갖는 기저들의 계수들만의 모듈러 가산으로 이루어진다. 이로부터, 벡터표현에 의한 가산은 각 비트별로 EX-OR 함으로써 쉽게 구현 할 수 있다.

2.3. AOP를 적용한 다항식의 승산연산

유한체상의 기약 AOP를 적용한 승산연산은 다항식의 승산전개와 모듈러 연산이 함께 이루어져야 함으로 그 연산형태가 복잡하다. 다항식 $A(\alpha)$와 $B(\alpha)$가 각각 $A(\alpha)=\sum_{l=0}^{m} A_l\alpha^l$, $B(\alpha)=\sum_{k=0}^{m} B_k\alpha^k$일 때, 두 다항식의 승산전개는 식 (6)과 같다.

$$\begin{aligned} P(\alpha) &= A(\alpha)B(\alpha) \\ &= (\sum_{l=0}^{m} A_l\alpha^l)(\sum_{k=0}^{m} B_k\alpha^k) \\ &= \sum_{k=0}^{m} B_k(\sum_{l=0}^{m} A_l\alpha^l)\alpha^k \\ &= \sum_{k=0}^{m} B_k \sum_{j=0}^{m} A_{<j-k>}\alpha^j) \end{aligned} \tag{6}$$

식 (6)의 연산결과를 전개하여 순환이동을 적용한 일반식은 k가 $0 \le k \le m$일 때, 식 (7)와 같다.

$$P_k = \sum_{n=0}^{m} B_n(A_{<k-n>}) \tag{7}$$

식 (7)을 기약 AOP를 갖는 $GF(2^4)$상의 승산연산 적용한 승산회로는 그림 1과 같다. 그림 1에서 승산연산은 다항식 $B(\alpha)$의 계수들을 모두 EX-OR 연산한 후에 다항식 $A(\alpha)$의 각 원소들과 AND연산하여 수행된다.

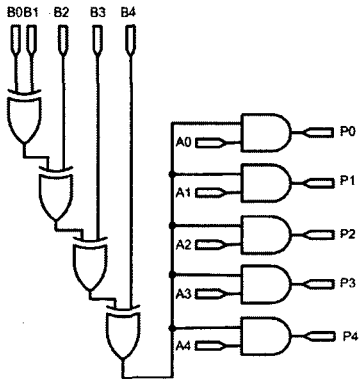

그림 1. 기약 AOP를 적용한 $GF(2^4)$상의 승산회로
Fig. 1. $GF(2^4)$ of the multiplier circuit using the irreducible AOP

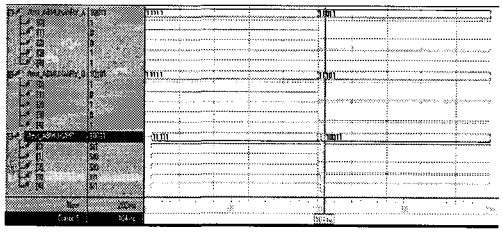

그림 2. 그림 1 회로의 모의실험 결과
Fig. 2. Simulation result of Fig. 1 circuit

그림 1의 회로를 Verilog-HDL에서 서로 다른 입력을 주었을 때 모의실험 한 결과를 그림 2에 보였다. 모의실험 결과에서 $A(\alpha)$, {10011}와 $B(\alpha)$, {10101}를 입력으로 주었을 때, $P(\alpha)$, {10011}로 출력됨을 알 수 있다. 또한 승산회로에서 EX-OR연산 지연 시간과 AND연산 지연 시간이 출력단에 영향을 준 것을 확인 할 수 있다.

2.4. AOP를 적용한 다항식의 멱승연산

유한체상의 기약 AOP를 적용한 멱승 연산은 하나의 다항식을 거듭제곱 이상으로 승산하는 연산이다. 앞서 살펴본 2.1절의 식 (4)를 이용하여 확장기저로 멱승연산을 수행할 때, $GF(2^m)$상의 다항식 $A(\alpha)$의 제곱은 식 (8)과 같다.

$$P(\alpha) = A(\alpha)A(\alpha) = \left(\sum_{l=0}^{m} A_l\alpha^l\right)\left(\sum_{l=0}^{m} A_l\alpha^l\right)$$
$$= \sum_{l=0}^{m} A_l\left(\sum_{l=0}^{m} A_l\alpha^l\right)\alpha^l$$
$$= \sum_{l=0}^{m} A_l\alpha^{2l} \qquad (8)$$

예를 들어, $GF(2^4)$상의 다항식 $A(\alpha)$의 제곱연산을 수행하는 과정은 식 (9)와 같다.

$$\begin{aligned} A(\alpha) &= A_4\alpha^4 + A_3\alpha^3 + A_2\alpha^2 + A_1\alpha + A_0 \\ P(\alpha) &= A(\alpha)A(\alpha) \\ &= A_0(A_0\oplus A_1\alpha\oplus A_2\alpha^2\oplus A_3\alpha^3\oplus A_4\alpha^4) \\ &\quad + A_1(A_0\oplus A_1\alpha\oplus A_2\alpha^2\oplus A_3\alpha^3\oplus A_4\alpha^4) \\ &\quad + A_2(A_0\oplus A_1\alpha\oplus A_2\alpha^2\oplus A_3\alpha^3\oplus A_4\alpha^4) \\ &\quad + A_3(A_0\oplus A_1\alpha\oplus A_2\alpha^2\oplus A_3\alpha^3\oplus A_4\alpha^4) \\ &\quad + A_4(A_0\oplus A_1\alpha\oplus A_2\alpha^2\oplus A_3\alpha^3\oplus A_4\alpha^4) \end{aligned} \qquad (9)$$

식 (9)에서 $P(\alpha)$의 각 계수들을 정리하면 식 (10)과 같다.

$$\begin{aligned} P_0 &= A_0\oplus A_3\oplus A_1\oplus A_4\oplus A_2 \\ P_1 &= A_1\oplus A_4\oplus A_2\oplus A_0\oplus A_3 \\ P_2 &= A_2\oplus A_0\oplus A_3\oplus A_1\oplus A_4 \\ P_3 &= A_3\oplus A_1\oplus A_4\oplus A_2\oplus A_0 \\ P_4 &= A_4\oplus A_2\oplus A_0\oplus A_3\oplus A_1 \end{aligned} \qquad (10)$$

그림 3은 식 (10)에서 도출한 $P(\alpha)$의 각 계수들을 회로로 구현한 $A^2(\alpha)$ 멱승회로이다.

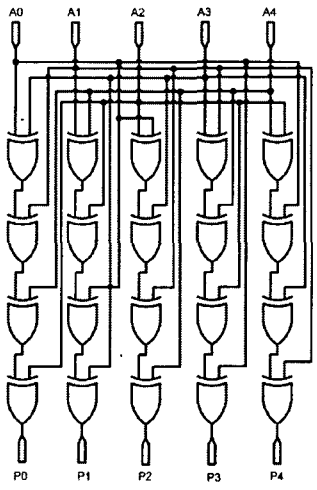

그림 3. 기약 AOP를 적용한 $GF(2^4)$ 상의 $A^2(\alpha)$ 멱승회로
Fig. 3. $GF(2^4)$ of the $A^2(\alpha)$ power circuit using the irreducible AOP

그림 3의 회로를 Verilog-HDL에서 입력 다항식 $A(\alpha)$에 같은 입력을 주었을 때의 모의실험 한 결과를 그림 4에 보였다. 모의실험 결과에서 $A(\alpha)$, {10101}를 입력으로 주었을 때, $P(\alpha)$, {11101}로 출력됨을 알 수 있다.

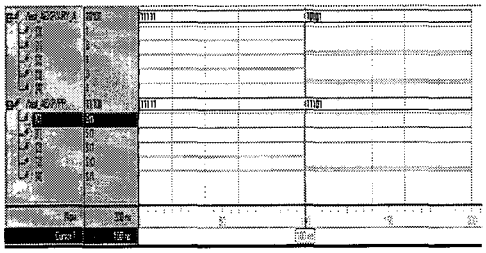

그림 4. 그림 3 회로의 모의실험 결과
Fig. 4. Simulation result of Fig. 3 circuit

2.5. AOP를 적용한 다항식의 파워섬연산

유한체상의 기약 AOP를 적용한 파워섬 연산은 앞서 다루었던 2.1절의 AOP정의와 2.3절의 승산 연산, 2.4절의 멱승연산 등을 이용한다.

기약 AOP를 적용한 파워섬(power-sum) 연산, $P(\alpha)$ $A(\alpha)B(\alpha)^2$은 다항식 $B(\alpha)$의 멱승 연산을 취한 후에 다항식 $A(\alpha)$와의 승산 연산을 하는 과정으로 이루어진다. 식 (8)을 이용하여 다항식 $B(\alpha)$의 멱승 연산 나타내면 식 (11)와 같다.

$$\begin{aligned} B^2(\alpha) &= \sum_{l=0}^{m} B_l \alpha^{2l} \\ &= B_4\alpha^8 + B_3\alpha^6 + B_2\alpha^4 + B_1\alpha^2 + B_0 \end{aligned} \tag{11}$$

다음은 승산 연산과정으로 식 (11)을 식 (6)에 적용하여 파워섬 연산을 표현하면 식 (12)와 같다.

$$\begin{aligned} P(\alpha) &= A(\alpha)B^2(\alpha) \\ &= (\sum_{l=0}^{m} A_l\alpha^l)(\sum_{k=0}^{m} B_k\alpha^{2k}) \\ &= \sum_{k=0}^{m} B_k(\sum_{l=0}^{m} A_l\alpha^l)\alpha^{2k} \\ &= \sum_{k=0}^{m} B_k(\sum_{j=0}^{m} A_{<j-2k>}\alpha^j \end{aligned} \tag{12}$$

$GF(2^4)$상의 다항식 $A(\alpha)$와 $B(\alpha)$의 파워섬 연산을 식 (12)에 적용하여 $P(\alpha)$의 계수 값을 식 (13)에 나타내었다.

$$\begin{aligned} A(\alpha) &= A_4\alpha^4 + A_3\alpha^3 + A_2\alpha^2 + A_1\alpha + A_0 \\ B(\alpha) &= B_4\alpha^4 + B_3\alpha^3 + B_2\alpha^2 + B_1\alpha + B_0 \\ P(\alpha) &= A(\alpha)B^2(\alpha) \\ P_0 &= A_0B_0 \oplus A_3B_1 \oplus A_1B_2 \oplus A_4B_3 \oplus A_2B_4 \\ P_1 &= A_1B_0 \oplus A_4B_1 \oplus A_2B_2 \oplus A_0B_3 \oplus A_3B_4 \\ P_2 &= A_2B_0 \oplus A_0B_1 \oplus A_3B_2 \oplus A_1B_3 \oplus A_4B_4 \\ P_3 &= A_3B_0 \oplus A_1B_1 \oplus A_4B_2 \oplus A_2B_3 \oplus A_0B_4 \\ P_4 &= A_4B_0 \oplus A_2B_1 \oplus A_0B_2 \oplus A_3B_3 \oplus A_1B_4 \end{aligned} \tag{13}$$

그림 5는 식 (13)에서 도출한 기약 AOP를 갖는 $GF(2^4)$ 상의 파워섬 연산 회로이다. 그림 5에서 회로의 모든 소자는 단지 2-입력 AND게이트와 EX-OR게이트들로만 표현

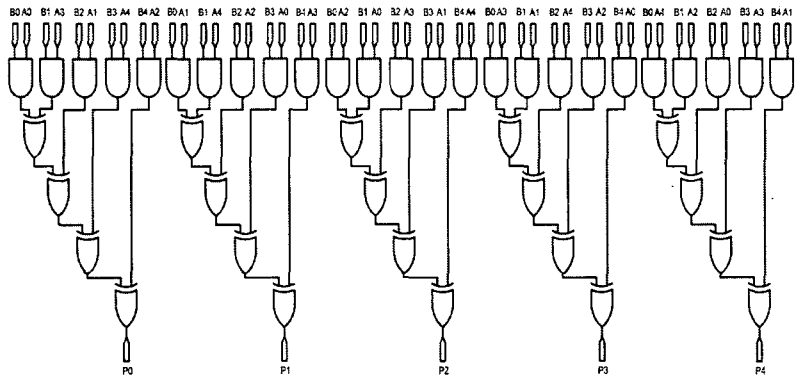

그림 5. 기약 AOP를 적용한 $GF(2^4)$ 상의 Power-Sum 연산회로
Fig 5. $GF(2^4)$ of the Power-Sum circuit using the irreducible AOP

되며 다항식 $A(\alpha)$와 $B(\alpha)$의 각각의 계수들은 앞서 정의한 멱승연산에 대한 식 (10)과 승산 연산에 대한 식 (7)의 회로 구성 특성을 포함하여 5번의 AND연산을 실행하고 4번의 EX-OR연산을 실행하여 $P(\alpha)$의 모든 계수 값들을 얻을 수 있다.

그림 5의 회로를 Verilog-HDL에서 서로 다른 입력을 주었을 때 모의실험 한 결과를 그림 6에 보였다. 그림 6에서, 5번의 AND연산과 4번의 EX-OR연산으로 20ns의 지연시간이 출력 단에 영향을 끼친 것을 확인 하였으며, 다항식

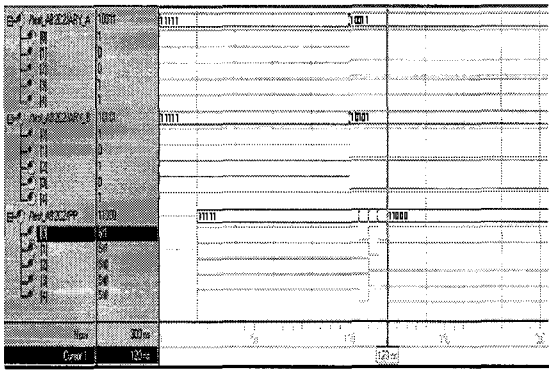

그림 6. 그림 5 회로의 모의실험 결과
Fig. 6. Simulation result of Fig. 5 circuit

$A(\alpha)$,{10011}와 $B(\alpha)$,{10101}를 입력으로 주었을 때, $P(\alpha)$,{11000}로 출력됨을 알 수 있다.

2.6. AOP를 적용한 AB^2+C연산

AOP를 적용한 AB^2+C 연산은 2.5절에서 전개한 파워섬 연산에 2.2절에서 전개한 가산 연산을 더하면 된다. 유한체 가산연산은 가산연산 후 발생하는 자리올림은 고려하지 않기 때문에 식 (12)의 각 P_k 에 C의 각 계수들을 식 (5)와 같이 가산함으로 연산을 완성할 수 있으며, 식 (14)와 같다.

$$\begin{aligned} A(\alpha) &= A_4\alpha^4 + A_3\alpha^3 + A_2\alpha^2 + A_1\alpha + A_0 \\ B(\alpha) &= B_4\alpha^4 + B_3\alpha^3 + B_2\alpha^2 + B_1\alpha + B_0 \\ C(\alpha) &= C_4\alpha^4 + C_3\alpha^3 + C_2\alpha^2 + C_1\alpha + C_0 \\ P(\alpha) &= A(\alpha)B^2(\alpha) + C(\alpha) \\ &= \sum_{n=0}^{m}((B_k(\sum_{l=0}^{m}A_l\alpha^l)\alpha^{2k}) + C_n)a^n \end{aligned} \quad (14)$$

식 (14)를 적용한 $GF(2^4)$상의 다항식 $A(\alpha)$, $B(\alpha)$, $C(\alpha)$의 AB^2+C 연산 결과는 식 (15)와 같다.

$$\begin{aligned} P_0 &= (A_0B_0 \oplus A_3B_1 \oplus A_1B_2 \oplus A_4B_3 \oplus A_2B_4) \oplus C_0 \\ P_1 &= (A_1B_0 \oplus A_4B_1 \oplus A_2B_2 \oplus A_0B_3 \oplus A_3B_4) \oplus C_1 \\ P_2 &= (A_2B_0 \oplus A_0B_1 \oplus A_3B_2 \oplus A_1B_3 \oplus A_4B_4) \oplus C_2 \\ P_3 &= (A_3B_0 \oplus A_1B_1 \oplus A_4B_2 \oplus A_2B_3 \oplus A_0B_4) \oplus C_3 \\ P_4 &= (A_4B_0 \oplus A_2B_1 \oplus A_0B_2 \oplus A_3B_3 \oplus A_1B_4) \oplus C_4 \end{aligned} \quad (15)$$

Ⅲ. 비교 및 검토

본 장에서는 승산기의 구조와 기약 다항식, AND게이트, EX-OR 게이트, 1-비트 래치, 연산 지연 시간 등의 관점에서 본 논문에서 제안한 연산 기법을 일반적인 생성 다항식 구조로 표준기저를 근거로 한 $GF(2^m)$상의 시스토릭 구조인 Yeh[5]의 승산기와 AB^2+C의 시스토릭 승산 기법에 기약 AOP를 기반으로 최적화한 Lee[8]의 승산기와 비교하여 표 1에 정리하였다.

표 1 $GF(2^m)$상의 AB^2+C 연산기의 구성 비교표
Table 1 Comparison of the computing AB^2+C in $GF(2^m)$

		Yeh[5]	Lee[8]	본 논문
구조		시스토릭	시스토릭	비-시스토릭
생성 다항식		일반	기약 AOP	기약 AOP
수식		$AB+C$	AB^2+C	AB^2+C
기저		표준	확장된 표준	확장된 표준
회로 복잡도	2입력 AND	$2m^2$	$(m+1)^2$	$(m+1)^2$
	2입력 EX-OR	$2m^2$	$(m+1)^2$	$(m+1)^2$
	1-bit 래치	$7m^2$	$3(m+1)^2$	-
지연시간		$3m$	$m+1$	-
연산지연시간		$T_A+T_X+2T_L$	$T_A+T_X+T_L$	$T_A+(1+\lceil \log_2^m \rceil)T_X$

표 1에서 Yeh[5], Lee[8]의 연산회로에서 채택한 시스토릭 구조는 일반적으로 매우 큰 m상의 연산소자들에 의해 발생하는 전파지연 시간의 축적을 방지하고 파이프라인 연산이 가능하므로 고속 및 대용량의 연산시스템에 유리한 특성을 갖는다. 그러나 별도의 메모리소자와 동기신호가 필요하다는 단점을 가지고 있다. 본 논문에서 제안한 AB^2+C 연산은 비-시스토릭 구조로 래치가 필요하지 않기 때문에 비교 회로에 대한 복잡도를 개선할 수 있는 장점을 가지고 있다. 지연시간에서도 Yeh와 Lee의 회로는 시스토릭 구조로 각 셀 내부의 소자에 의해 발생하는 전파지연시간과 셀 간의 연산 동기 시간을 제어하기 위해 필요한 지연시간이 필요하나 본 논문의 회로는 메모리를 필요로 하지 않으므로 지연시간이 필요하지 않다. 본 논문에서 제안한 연산기법으로 감소된 회로의 연산지연시간은 $\lceil \log_2^m \rceil$ 의 지연시간을 가지므로 기존의 연산지연시간에 비해 그 지연시간이 크지 않다. 따라서 전파지연시간과 지연시간을 함께 고려할 때 매우 빠른 연산시간을 갖는다.

Ⅳ. 결 론

본 논문에서는 $GF(2^m)$상의 승산전개상에 AOP 순환이동과 $GF(2^m)$ 상의 AB^2+C 연산기 구성에 대하여 논의하였다. 본 논문에서 제안한 기법은 기존의 연산기법에 비해 각 연산부분을 모듈로 정의하여 정규화가 용이하며 AND게이트와 EX-OR게이트만으로 회로 구성이 가능하므로 회로의 정규화 및 집적화, 전파 지연 시간의 감소 등의 장점을 갖는다. 따라서 AB^2+C 연산기를 이용하여 고속 대용량 연산시스템의 개발을 위한 구성소자의 수에 따른 회로복잡도와 연산지연시간을 개선한 효율적인 $GF(2^m)$상의 승산기가 고속의 연산으로 구현될 것을 기대한다.

참고문헌

[1] G. L. Feng, A VLSI Architecture for Fast Inversion in $GF(2^m)$, *IEEE Trans. on Computers*, vol. 38, no. 10, Oct. 1989.

[2] C. K. Koc, and B. Sunar, Low-Complexity Bit Parallel Canonical and Normal Basis Multipliers for a Class of Finite Fields, *IEEE Trans. on Computer*, vol. 47, no. 3, pp. 353~356. Mar. 1988.

[3] E. D. Mastrovito, *VLSI Architectures for Computation in Galois Fields, Ph.D thesis, Dept. of Electrical Eng. of Linkoping Univ.*, Sweden, 1991.

[4] B. Sunar and C. K. Koc, Mastrovito Multiplier for All Trinomials, *IEEE Trans. on Computers*, vol. C-48, No. 5, pp. 522~527, May 1999.

[5] C. S. Yeh, I.S. Reed, and T. K. Truong, Systolic Multipliers for Finite Fields $GF(2^m)$, *IEEE Trans. on Computers*, vol. 33, no. 4, pp. 357~360, Apr. 1984.

[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] C. Y. Lee, J. Y. Lee, Bit-Parallel Systolic Multipliers for $GF(2^m)$ Fields Defined by All-One and Equally Spaced Polynomials, *IEEE Trans.. Computers*, vol. 50, No. 5, pp.385-393, May 2001.

[8] C. Y. Lee, E. H. Lu, "Low-Complexity Bit-Parallel Systolic Architecture for Computing AB^2+C in a Class of Finite Field $GF(2^m)$," *IEEE Trans. Comput.*, vol. 48, No 5, May 2001.

[9] 변기영, " 시스템 복잡도 개선을 위한 AOP 기반의 병렬 유한체 승산기" *한국통신학회 논문지* 2003년 04월 Vol.29 No.3A

저자소개

황 운 택(Yoon-taek Hwang)

1973년 2월 광운대학교 전자공학과 (공학사)
1981년 2월 숭실대학교 전자공학과 (공석사)
1998년 2월 한국해양대학교 전파공학과 박사수료
1977년 2월 - 현재 시립 인천전문대학 정보통신과 교수
※관심분야 : 무선기기 전원부 설계, 이동통신 기술 개발