論文2000-37SC-4-5

멀티플렉서를 이용한 $GF(2^m)$상의 승산기
(Multiplexer-Based Array Multipliers over $GF(2^m)$)

黃 鍾 學*, 朴 承 用*, 申 富 植**, 金 興 壽*
(Jong-Hak Hwang, Seung-Yong Park, Boo-Sik Shin, and Heung-Soo Kim)

요 약

본 논문에서는 유한체 $GF(2^m)$상에서 두 다항식의 승산 알고리즘을 제시하였다. 이 알고리즘은 반복적인 배열로 병렬 승산을 효과적으로 실현하며, 동일한 시간에 고속 동작을 실현한다. 제시된 승산기는 승산연산부와 mod연산부, 원시 기약다항식연산부로 구성하였다. 승산연산부는 멀티플렉서, X-OR게이트, AND게이트, MUX로 구성하였으며, mod연산부는 AND게이트, X-OR게이트로 구성하였다. 또한 본 논문에서 제시한 승산에는 효과적인 파이프형을 도입하였다. 도출된 모든 승산기는 고속 동작하며, 회로 복잡성이 감소한다. 셀들의 내부결선도는 VLSI 실현에 적합하도록 규칙적으로 구성되었다.

Abstract

In this paper, the multiplicative algorithm of two polynomals over finite field $GF(2^m)$ is presented. The proposed algorithm permits an efficient realization of the parallel multiplication using iterative arrays. At the same time, it permits high-speed operation. This multiplier is consisted of three operation unit: multiplicative operation unit, the modular operation unit, the primitive irreducible operation unit. The multiplicative operation unit is composed of AND gate, X-OR gate and multiplexer. The modular operation unit is constructed by AND gate, X-OR gate. Also, an efficient pipeline form of the proposed multiplication scheme is introduced. All multipliers obtained have low circuit complexity permitting high-speed operation and interconnection of the cells are regular, well-suited for VLSI realization.

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]

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 VLSI 구조의 설계에 부적합하였다.[9,10]

최근 Yeh등[2]은 표준 기저 표현식을 사용하여 유한

* 正會員, 仁荷大學校 電子工學科
(Inha University, Dept. of Electronic Engineering)
** 正會員, 安山 1大學 인터넷情報科
(AnSan College, Dept. of Computer Information)
接受日字:1999年 11月11日, 수정완료일:2000年 6月22日

체상의 승산을 실현하는 직렬 입력/직렬 출력 시스토릭 배열 구조와 병렬 입력/병렬 출력 시스토릭 배열 구조의 승산기를 개발하였다. Scott등[5]은 표준 기저로 표현도니 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.

본 논문에서는 Pekmestzi[13]등이 제시한 멀티플렉서를 이용한 승산 알고리즘을 $GF(2^m)$상의 승산 알고리즘으로 확장하여 병렬 입-출력 모듈구조의 다치 승산기를 제시하였다. 이 다치 승산기의 기본 셀은 승산 연산부, mod 연산부, 원시 기약 다항식 연산부로 구성된다. 승산 연산부는 AND, XOR, MUX 게이트로 구성하며, mod 연산부는 AND와 XOR 게이트로 구성한다. 원시 기약 다항식은 AND와 XOR게이트로 이루어져 있다. 그리고 원시 기약 다항식 연산부는 시작할 때 한번만 수행하면 되므로 출력결과 수행 시간에 영향을 주지 않는다.

Ⅱ. 유한체의 승산 알고리즘[5,10,11,13]

유한체 $GF(p^m)$은 p가 소수이고 m이 양의 정수인 p^m개의 원소들을 가지며 p개의 원소들을 갖는 기초체 $GF(p)$의 확대체이다. 유한체 $GF(p^m)$는 {0, 1, 2, …, $p-1$}의 원소들로 구성된다. $GF(p^m)$에서 모든 산술 연산은 mod(p)연산으로 이루어지며, $GF(p^m)$의 0이 아닌 모든 원소들은 원시 원소 α에 의해 생성된다.

유한체 $GF(2^m)$상에서 피승산 다항식을 $A(x)$, 승산 다항식을 $B(x)$, 원시 기약 다항식을 $F(x)$라 하고, 다음과 같이 전개하여 표현할 수 있다.

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

$$B(x) = b_{m-1}\cdot x^{m-1} + \cdots b_1\cdot x + b_0 \quad (2)$$

$$F(x) = x^m + f_{m-1}\cdot x^{m-1} + \cdots f_1\cdot x + f_0 \quad (3)$$

여기서 $F(x)$는 최고 차수가 m이고 계수 $f_m=1$인 모닉 다항식이고, 계수 a_i, b_i, f_i는 {0, 1}의 값을 갖는다.

그리고 두 다항식 $A_{m-1}(x)$와 $B_{m-1}(x)$는 식 (4), (5)와 같이 표현한다.

$$A_{m-1}(x)=a_{m-2}\cdot x^{m-2}+\cdots a_1\cdot x + a_0 \quad \text{이고}$$
$$A(x) = a_{m-1}\cdot x^{m-1}+A_{m-1}(x) \quad (4)$$

$$B_{m-1}(x)=b_{m-2}\cdot x^{m-2}+\cdots b_1\cdot x + b_0 \quad \text{이고}$$
$$B(x) = b_{m-1}\cdot x^{m-1}+B_{m-1}(x) \quad (5)$$

이다.

두 다항식 $A(x)$와 $B(x)$의 승산 알고리즘은 이들 다항식의 계수들을 곱하여 mod $F(x)$ 연산을 수행하므로 구할 수 있다. $GF(2^n)$상에서 $A(x)$와 $B(x)$의 승산은 식 (6)과 같다.

$$R = \{A(x)\cdot B(x)\} \bmod F(x) \quad (6)$$

여기서 $P=A(x)\cdot B(x)$라 놓으면, P는 다음과 같이 전개할 수 있다.

$$\begin{aligned} P &= A(x)\cdot B(x) \\ &= [a_{m-1}\cdot x^{m-1}+A_{m-1}(x)] \\ &\quad \cdot [b_{m-1}\cdot x^{m-1}+B_{m-1}(x)] \\ &= a_{m-1}\cdot b_{m-1}\cdot x^{2m-2}+[a_{m-1}\cdot B_{m-1}(x) \\ &\quad + A_{m-1}(x)\cdot b_{m-1}]\cdot x^{m-1} \\ &\quad + A_{m-1}(x)\cdot B_{m-1}(x) \end{aligned} \quad (7)$$

$P_{m-1}=A_{m-1}(x)\cdot B_{m-1}(x)$이라 정의하면,

$$P_j = A_j(x)\cdot B_j(x) \quad (8)$$

라 놓을 수 있다.

P_j는 순환 함수로 식 (9)과 같이 표현할 수 있다.

$$\begin{aligned} P_j &= A_j(x)\cdot B_j(x) \\ &= a_{j-1}\cdot b_{j-1}\cdot x^{2j-2}+[a_{j-1}\cdot B_{j-1}(x) \\ &\quad + b_{j-1}\cdot A_{j-1}(x)]\cdot x^{j-1}+A_{j-1}(x)\cdot B_{j-1}(x) \\ &= a_{j-1}\cdot b_{j-1}\cdot x^{2j-2}+[a_{j-1}\cdot B_{j-1}(x) \\ &\quad + b_{j-1}\cdot A_{j-1}(x)]\cdot x^{j-1} + P_{j-1} \end{aligned} \quad (9)$$

따라서 $P=\{A(x)\cdot B(x)\} \bmod F(x)$는 식 (10)과 같이 표현된다.

$$\begin{aligned} P &= \sum_{j=0}^{m-1} a_j\cdot b_j\cdot x^{2j} \\ &\quad + \sum_{j=1}^{m-1}[a_j\cdot B_j(x)+b_j\cdot A_j(x)]x^j \end{aligned} \quad (10)$$

여기서,

$$Z_j = a_j\cdot B_j(x)+b_j\cdot A_j(x) \quad (11)$$

라 놓으면

$$P = \sum_{j=0}^{m-1} a_j \cdot b_j \cdot x^{2j} + \sum_{j=1}^{m-1} Z_j \cdot x^j \qquad (12)$$

이다.

Z_j는 표 1에서 보는 바와 같이 변수 a_j와 b_j에 의하여 논리적으로 얻을 수 있다.

표 1. Z_j의 값
Table 1. The Values of Z_j.

a_j	b_j	Z_j
0	0	0
0	1	$A_j(x)$
1	0	$B_j(x)$
1	1	$A_j(x)+B_j(x)=S_j$

변수 a_j와 b_j가 모두 1일 때 Z_j의 값은 S_j이다. 그때 S_j는 다음 식 (13)에서와 같이 j-1번째에서 구해진다.

$$S_j = S_{j-1} \oplus a_{j-1} \cdot x^{j-1} \oplus b_{j-1} \cdot x^{j-1} \qquad (13)$$

여기서 ⊕는 modular 연산이다.

식 (13)에서 $GF(2^n)$상의 같은 차수의 계수 덧셈은 X-OR 게이트로 연산할 수 있으므로, 그림 1과 같다.

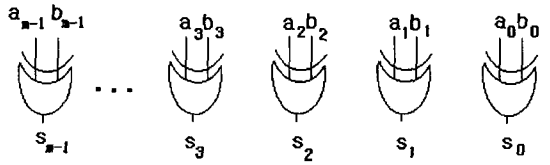


그림 1. $GF(2^n)$상의 S_j값 생성 회로도
Fig. 1. Adder for the generation of S_j in $GF(2^n)$.

또한 표 1은 제어 입력 비트 a_j와 b_j인 4X1 멀티플렉서를 이용하여 회로를 구성할 수 있다. 다음절에서 이제까지 전개된 알고리즘을 이용하여 병렬 승산기를 구성한다.

Ⅲ. 멀티플렉서를 이용한 $GF(2^m)$상에서 병렬 승산기 구현

이 장에서는 $GF(2^m)$ 상의 승산 $R=A \cdot B \bmod F(x)$을 실행하는 병렬 입출력 승산기의 구성을 논하였다. 그림 2는 $GF(2^m)$ 상의 두 원소들의 승산을 실행하는 승산기의 구성도이다.

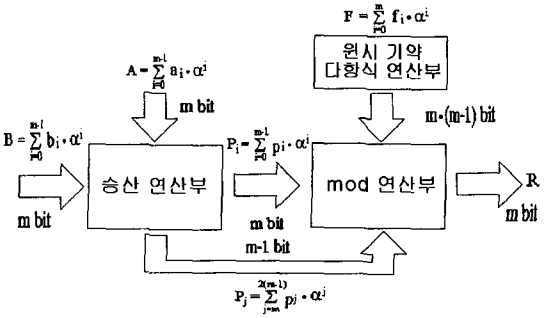


그림 2. $GF(2^m)$상의 승산기
Fig. 2. A multiplier in $GF(2^m)$.

이 승산기는 $GF(2^m)$상의 두 원소들의 승산을 실행하는 승산 연산부와 승산 연산부의 출력을 입력으로 하여 원시 기약 다항식에 의한 mod 연산을 행하는 mod 연산부와 원시기약 다항식을 산술연산 처리하는 기약 다항식 연산부로 구성되며, 그림 2와 같다.

1. 승산연산부

$GF(2^m)$상의 두 원소들의 승산을 실행하는 승산연산부는 식 (12)에서 $Z_j \cdot x^j$항과 $a_j \cdot b_j \cdot x^{2j}$항을 그림 3

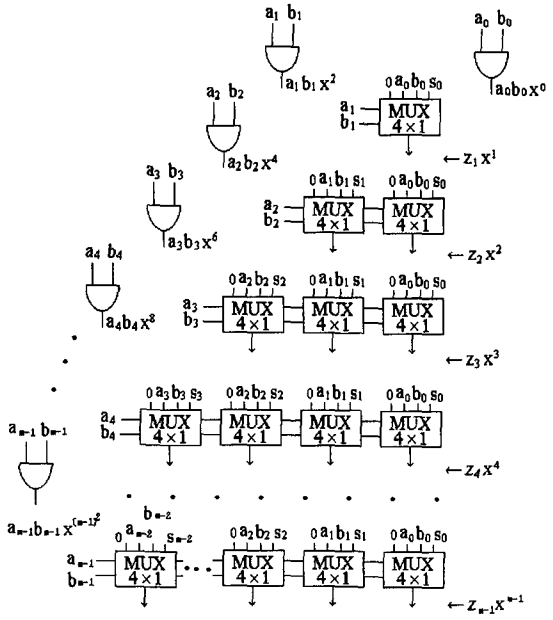


그림 3. 제안된 승산기 알고리즘을 이용한 $Z_j \cdot x^j$항과 $a_j \cdot b_j \cdot x^{2j}$항 회로도
Fig. 3. The implementation of the terms $Z_j \cdot x^j$ and $a_j \cdot b_j \cdot x^{2j}$ used in the proposed multiplication algorithm.

에서 보는 바와 같이 MUX와 AND게이트로 표현할 수 있다.

그림 3에서 구현된 $Z_j \cdot x^j$ 항과 $a_j \cdot b_j \cdot x^{2j}$ 항을 구현한 소자들에 의하여 그림 4와 같이 $GF(2^m)$상의 승산 연산부를 구현할 수 있다.

그림 4에서 *Cell I*과 *Cell II*의 회로 구성을 그림 5에 나타내었다. 소자는 $m^2/2+3m/2$ XOR와 m AND 게이트가 필요하고, MUX는 $m^2/2+m/2$가 필요하다.

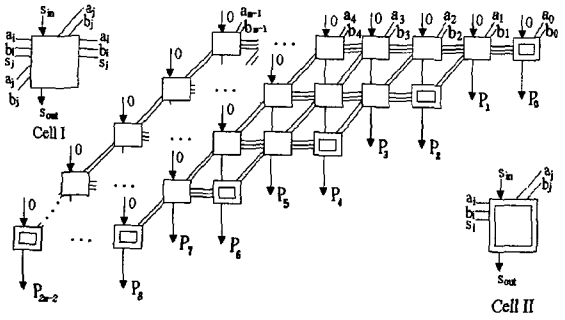


그림 4. $GF(2^m)$상의 멀티플렉서를 이용한 승산기
Fig. 4. The proposed multiplexer-based parallel multiplier in $GF(2^m)$.

2. mod 연산부

$GF(2^m)$상에서 승산기 구현을 식 (6)에와 같이 승산연산부를 통하여 출력된 P을 mod$F(x)$를 취하여 승산 결과를 얻을 수 있다.

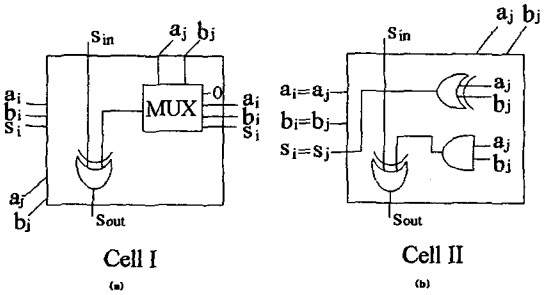


그림 5. (a) 제안된 승산기에 사용된 *Cell I*의 내부 회로도 (b) 제안된 승산기에 사용된 *Cell II*의 내부 회로도
Fig. 5. (a) First type cell (CELL I) used in the proposed array multiplier. (b) Second type cell (CELL II) used in the proposed array multiplier.

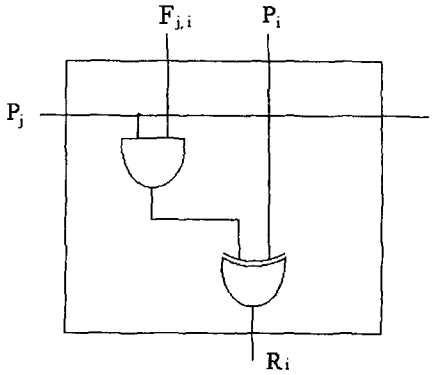


그림 6. $GF(2^m)$상의 mod연산부의 기본 셀
Fig. 6. The basic cell of modular operation part in $GF(2^m)$.

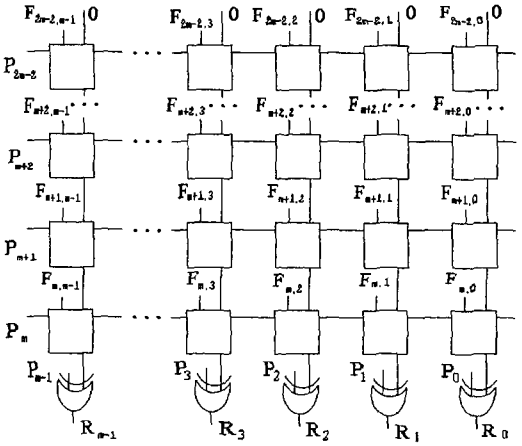


그림 7. $GF(2^m)$상의 mod연산부
Fig. 7. The modularity operation part in $GF(2^m)$.

즉, 승산 연산부의 출력을 입력하는 mod 연산부는 AND와 XOR게이트로 셀계된 기본 셀의 배열에 의해 구성된다. 그림 6은 기본 셀의 회로도이며, 그림 7과 같이 구현된다. 이 셀의 출력 R_i

$$R_i=(P_j \cdot F)\oplus P_i \tag{14}$$

이다. 여기서 P_i, P_j는 승산 연산부의 출력이고, F는 원시 기약다항식 연산부의 출력이다. 또한 i는 $\{0, 1, \cdots, m-1\}$이고 j는 $\{m, m+1, \cdots, 2m-2\}$이다.이 기본 셀의 배열에 의한 $GF(2^m)$상의 승산 연산부와 원시 기약다항식 연산부의 출력을 입력으로 하는 mod 연산부는 그림 7과 같다. 여기서 상측은 원시기약다항식 연산부의 출력과 승산 연산부의

출력 $P_i = \sum_{i=0}^{m-1} P_i \cdot \alpha^i$ 의 계수 원소들이, 좌측은 $P_j = \sum_{j=0}^{2m-2} P_j \cdot \alpha^j$ 의 계수 원소들이 각각 가해지고, 우측은 $R = \sum_{k=0}^{m-1} R_k \cdot \alpha^k$ 의 계수 원소들이 출력된다. 이때 mod 연산부의 동작을 원시 기약다항식 연산부에서 출력된 $F_{j,i}$ 값들이 입력되어 회로소자 지연시간이 지나면 출력된다.

mod연산부의 소자수는 m^2 XOR와 m^2-m AND 게이트가 필요하다.

3. 원시 기약다항식연산부

$GF(2^m)$상의 원시기약다항식 $F(\alpha)=0$이고 모닉 다항식이므로 $\alpha^m = \sum_{i=0}^{m-1} f_i \cdot \alpha^k$이다.

이 다항식에 의해 두 원소들의 승산에서 m이상의 차수를 m-1이하로 감소시킨다. $GF(2^m)$상의 원시 기약다항식 연산부 기본 셀은 그림 8과 같으며, 원시 기약다항식 연산부는 그림 9와 같이 구현된다. 이 연산부의 셀 내부는 한 개의 AND게이트와 한 개의 XOR 게이트에 의해 구성되었다. 회로 동작은 클럭신호(clock signal)에 의해 연산되는 것이 아니고 회로의 각 소자 지연시간에 의한 연산 결과가 출력된다. 그리고 기약 다항식 연산은 반복적인 실행이 아니라 처음 시작할 때 한번 만 실행하면 되므로 연산 결과시간은 중요치 않다.

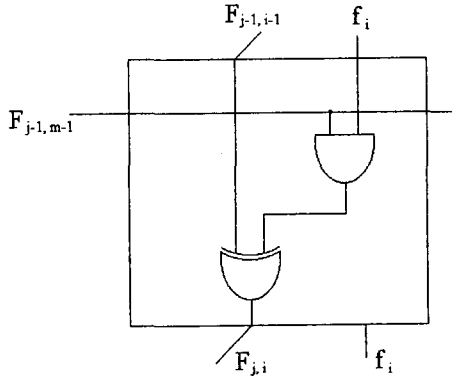


그림 8. $GF(2^m)$상의 원시 기약다항식 연산부 기본셀
Fig. 8. The basic cell of primitive irreducible operation part in $GF(2^m)$.

원시 기약다항식연산부는 m^2-2m XOR와 m^2-2m AND 게이트가 필요하다. 따라서 승산연산부, mod연산부, 원시 기약다항식연산부에 소요되는 모든 소자를 합하면 $5m^2/2-m/2$ XOR와 $2m^2-2m$ AND 게이트가 필요하고, MUX는 $m^2/2+1/2m$ MUX가 필요하다.

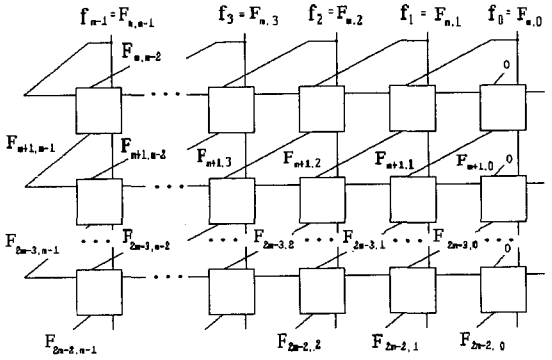


그림 9. $GF(2^m)$상의 원시 기약다항식 연산부
Fig. 9. The primitive irreducible operation part in $GF(2^m)$.

Ⅳ. 비교 및 검토

본 장에서는 제시한 멀티플렉서를 이용한 승산기를 타 논문의 승산기들과 비교하였으며 비교 결과는 표 2와 같다.

타 연구의 승산기들과 본 논문에서 제시한 멀티플렉서를 이용한 다치 연산기를 각 연산 회로별로 비교하면 다음과 같다. 클럭 시간과 전달 지연 시간도 단일 연산 회로에 전달되는 시간이 같다고 가정[15]하여 단위 시간으로 계산하였다.

표 2에서 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 안정 시간이 부가적으로 더 필요하다. 그러나 본 논문에서는 클럭을 요하지 않는다. 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. 본 논문은 MUX를 사용하여 승산하는 계수들을 병렬 연산할 수 있도

표 2. 비교표
Table 2. The table of comparison.

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

록 하였다. 그리고 본 논문에서 사용되는 MUX는 승산하는 계수들에 의하여 계산에 필요한 값을 선택하며 $1/2(m^2-m)$개가 소요된다. 제안된 멀티플렉서를 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 멀티플렉서를 이용한 승산기를 제시하였다. 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 동작시간은 승산 연산부에서 $m-2$개의 XOR게이트 소자 지연 시간과 mod 연산부에서 $m-1$개의 XOR게이트 소자지연시간이 소요되나 승산 연산부와 mod연산부가 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.

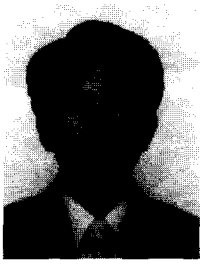
[15] C.K. Koc and B. Sunar, "Low-complexity bit-parallel canonical and normal basis multipliers for a class of finite fields," *IEEE Trans. Comput.*, vol. c-47, No. 3, pp.353-356, March 1998.

[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.

저 자 소 개

黃 鍾 學(正會員)

1964년 6월 12일 1988년 인하대학교 전자공학과 졸업(학사). 1990년 인하대학원 전자공학과 졸업(석사). 1996년~현재 인하대학원 전자공학과 박사과정. 1990년~1991년 (주)필코 부설연구소 연구원. 1992년~1995년 나우정밀 중앙연구소 전임연구원. 1996년~현재 체육과학연구원 선임연구원. 관심분야는 이동통신, 체육 기자재, FPGA설계



朴 承 用(正會員)

1954년 4월 2일생. 1979년 인하대학교 전자공학과 졸업(학사). 1982년 인하대학원 전자공학과 졸업(석사). 1996년~현재 인하대학교 전자공학과 박사과정. 1985년~현재 재능대학 정보계열 교수. 관심분야는 컴퓨터 시스템 및 네트웍

申 富 植(正會員) 第36卷 C編 第5號 參照

金 興 壽(正會員) 第36卷 C編 第5號 參照