GF(2^m)상에서 나눗셈/역원 연산을 위한 AB^2 시스톨릭 어레이 설계 및 분석

(Design and Analysis of a AB^2 Systolic Arrays for Division/Inversion in GF(2^m))

김 남 연 † 고 대 곤 †† 유 기 영 †††

(Nam-yeun Kim) (Dae-Ghon Kho) (Kee-Young Yoo)

요 약 GF(2^m)상의 공개키 암호 시스템에서 AB^2 연산은 효율적이고 기본적인 연산으로 잘 알려져 있다. 나눗셈/역원은 기본이 되는 연산으로, 내부적으로 AB^2 연산을 반복적으로 수행함으로써 계산이 된다. 본 논문에서는 GF(2^m)상에서 AB^2 연산을 수행하는데 필요한 새로운 알고리즘과 그에 따른 병렬 입/출력 및 시리얼 입/출력 구조를 제안한다. 제안된 알고리즘은 최상위 비트 우선 구조를 기반으로 하고, 구조는 기존의 구조에 비해 낮은 하드웨어 복잡도와 적은 지연을 가진다. 이는 역원과 나눗셈 연산을 위한 기본 구조로 사용될 수 있으며 암호 프로세서 칩 디자인의 기본 구조로 이용될 수 있고, 또한 단순성, 규칙성과 병렬성으로 인해 VLSI 구현에 적합하다.

키워드 : 공개키 암호 시스템, AB^2 알고리즘, 나눗셈/역원, 시스톨릭 어레이

Abstract Among finite field arithmetic operations, the AB^2 operation is known as an efficient basic operation for public key cryptosystems over GF(2^m). Division/Inversion is computed by performing the repetitive AB^2 multiplication. This paper presents two new AB^2 algorithms and their systolic realizations in finite fields GF(2^m). The proposed algorithms are based on the MSB-first scheme using standard basis representation and the proposed systolic architectures for AB^2 multiplication have a low hardware complexity and small latency compared to the conventional approaches. Additionally, since the proposed architectures incorporate simplicity, regularity, modularity, and pipelinability, they are well suited to VLSI implementation and can be easily applied to inversion architecture. Furthermore, these architectures will be utilized for the basic architecture of crypto-processor.

Key word : Public-key Cryptosystem, Power multiplier algorithm, Division/Inversion, Systolic array

1. 서 론

최근 유한 필드 상의 연산은 에러-교정 코드[1], 암호학[2, 3, 4], 디지 시그널 프로세싱[5] 등의 분야에서 주목을 받고 있다. 그러한 분야에서의 정보처리는 주로 곱셈, 파워썸(AB^2+C), 나눗셈/역원과 지수 연산을 필요로 한다. 그 중에서도 나눗셈/역원 연산은 ElGamal, RSA, ECC 등 공개키 암호시스템에 기본이 된다. 예를 들어, 타원곡선암호시스템을 설계할 때 여러 번의 나눗셈 연산이 필요한데, 나눗셈은 곱셈과 곱셈의 역원을 통해 계산($A/B=A \cdot B^{-1}$)될 수 있다. 이 때 사용된 곱셈의 역원은 $B^{-1} = B^{2^m-2} = (B(B(B\cdots(B(B)^2)^2\cdots)^2)^2)^2$과 같이 지수의 반복을 통해 얻어질 수 있다. 다음은 역원 연산의 과정을 Fermat의 이론에 따라 알고리즘으로 나타낸 것이다.

1단계 : $R = B$;

2단계 : For $i = m-2$ downto 1

3단계 : $R = B \cdot R^2$;

· This work was supported by Mobile Network Security Technology Research Center.

† 비 회 원 : 경북대학교 컴퓨터공학과
knyeun@hanmail.net
†† 비 회 원 : 대구교육대학교 전산교육과 교수
jdkho@dnue.ac.kr
††† 종신회원 : 경북대학교 컴퓨터공학과 교수
yook@knu.ac.kr

논문접수 : 2002년 5월 16일
심사완료 : 2002년 11월 1일

4단계 : $R = R^2$;

이 때, 결과는 $R = B^{-1}$ 이고, AB^2 연산이 단계 3과 4에서 사용되고 있음을 알 수 있다.

GF(2^m)상에서의 많은 구조들은 기저들을 달리하여 개발되어져 왔는데, 그 예로 정규기저(normal), 이원기저(dual basis), 다항식기저(polynomial basis) 타입이 있다. 그 중 정규기저와 이원기저 타입의 구조들은 각각의 장점을 가지고 있으나 기저 변환을 해야 한다는 단점을 지닌다. 반면에 다항식기저 구조는 기저 변환을 필요로 하지 않는다. 따라서 본 논문에서는 다항식기저를 사용하여 GF(2^m)상에서 AB^2 연산을 하는데 중점을 두겠다.

GF(2^m)상에서 파워썸(AB^2+C) 연산을 수행하기 위한 다항식 기저 시스톨릭 어레이에 대한 연구가 이미 이루어져 왔다[6, 7, 8]. 시스톨릭 어레이는 계산 위주의 문제를 위한 특수 목적의 보조처리기(back-end processor)로 사용되는데, 전역 시계에 의해 작동되어 동기성(synchrony)을 갖고 있으므로 비교적 간편하게 이용할 수 있고, 처리 요소들의 배열에 규칙성(regularity)이 있어 확장성이 좋다. 그리고, 인접한 처리 요소들로만 연결되어 교신하고 공간 및 시간적 국부성(spatial and temporal locality)이 있어 하드웨어 구현이 간편하며 자료 흐름의 방향과 속도가 일정한 선형성(linearity)을 갖고 있어 성능 분석이 비교적 용이하다.

Wei[6]는 파워썸 시스톨릭 구조에 MUX와 DEMUX를 하나씩 덧붙여 여덟 가지 다른 타입의 연산을 할 수 있도록 하였고, Wei[7]는 또한 [6]을 바탕으로 역원과 나눗셈을 위한 구조를 제안하고 있다. Wang[8]은 GF(2^m)에서 파워썸 연산을 수행하기 위한 단방향 구조를 제안하였다. 그러나 이러한 시스톨릭 파워썸 구조들은 하드웨어 복잡도가 높고 지연 시간이 길어 암호 시스템의 응용에는 적합하지 못하다. 그러므로 효율적인 파워썸 연산에 대한 연구가 필요하다.

따라서 본 논문에서는 다항식 기저를 사용한 GF(2^m)상에서의 파워썸 연산에 대한 알고리즘[10]을 수정·보완하여 AB^2 알고리즘을 제안하고, 이 알고리즘을 통해 병렬 입/출력 및 시리얼 입/출력 구조들을 유도한다. 제안된 알고리즘은 나눗셈/역원 구조에 적용하였을 때 병렬성을 제공하기 위해 최상위 비트 우선 (MSB-first) 구조를 사용하였고 기존의 알고리즘에 비해 단방향 데이터 흐름을 가진다. 또한, 제안된 SPM(Systolic Power Multiplier)와 MSPM(Modified Systolic Power Multiplier) 구조들은 기존의 구조들[7, 8]에 비해 하드웨어 복잡도를 각각 $2m$AND+$2m$XOR+($4m^2$+$6m$-1) Latches와 $2m$AND+$2m$XOR+($5m^2$+$2m$+3) Latches를 줄일 수 있었고, 지연시간도 전통적인 구조들에 비해 효율적인데, 특히 MSPM 구조는 m=160일 때 Wang[8]의 구조에 비해 지연시간을 40%줄일 수 있었다. 덧붙여 이 구조는 VLSI 구현에 적합하고 나눗셈/역원 구조에 쉽게 적용이 가능하다. McCanny[9]에 의하면, 단방향 구조가 다방향에 비해 cascadability, fault tolerance와 wafer-scale integration에 효율적이라고 한다. 따라서 본 논문에서는 단방향 구조 MSPM과 SMSPM(Serial Modified Systolic Power Multiplier)을 얻기 위해 SPM 구조를 분리 및 병합하였다.

본 논문의 구성은 다음과 같다. 2장과 3장에서는 시스톨릭 AB^2 알고리즘과 구조를 제안하고 , 4장에서는 구조를 시뮬레이션한 결과를 보이며, 기존의 구조들과 비교 및 분석을 한다. 마지막으로 5장에서 결론을 내린다.

2. GF(2^m)상에서 제안된 시스톨릭 AB^2 곱셈기

이 장에서는 다항식 기저를 사용하여 AB^2의 새로운 알고리즘과 병렬 시스톨릭 구조인 SPM(Systolic Power Multiplier)을 제안한다 .

2.1 GF(2^m)상에서 SPM 알고리즘

유한체 GF(2^m)상의 두 원소 $A(x)$, $B(x)$는 이 다항식 기저 표기법에 따르면 차수(degree)가 m-1 보다 작거나 같은 다항식이며, 다음과 같이 표기한다.

$$A(x) = \sum_{i=0}^{m-1} a_i x^i = a_{m-1}x^{m-1} + a_{m-2}x^{m-2} + \cdots + a_1 x + a_0$$

$$B(x) = \sum_{i=0}^{m-1} b_i x^i = b_{m-1}x^{m-1} + b_{m-2}x^{m-2} + \cdots + b_1 x + b_0$$

위 식에서 계수들은 a_i, $b_i \in$ GF(2)이고, 두 원소 A와 B는 $A = (a_{m-1}\ a_{m-2} \cdots a_1\ a_0)$, $B = (b_{m-1}\ b_{m-2} \cdots b_1\ b_0)$로 나타낸다. 유한체 GF($2^m$)상의 두 원소 a와 b의 덧셈은 다항식 $a(x)$와 $b(x)$의 더하기를 함으로서 수행되어지고, 각 계수들은 필드 GF(2)상에서 더하여 진다. 이것은 a와 b가 비트별로 XOR(Exclusive OR) 연산을 하는 것과 동일하다.

유한체 GF(2^m)상의 두 원소 A와 B^2의 곱셈은 차수가 m인 기약 다항식(irreducible polynomial)이 두 개 필요한데, $F(x)$를 GF(2)상의 차수가 m인 기약 다항식이라 하자.

$$F(x) = \sum_{i=0}^{m} f_i x^i = x^m + f_{m-1}x^{m-1} + f_{m-2}x^{m-2} + \cdots + f_1 x + f_0$$

이 때, 계수 $f_i \in$ GF(2)이고, 만약 α를 $F(x)$의 근이

라고 하면 $F(\alpha) = 0$이 되고, 따라서 다음과 같은 두 식을 얻을 수 있다.

$$F(\alpha) = \alpha^m = f_{m-1}\alpha^{m-1} + f_{m-2}\alpha^{m-2} + \cdots + f_1\alpha + f_0$$

$$F'(\alpha) \equiv \alpha^{m+1} = f'_{m-1}\alpha^{m-1} + f'_{m-2}\alpha^{m-2} + \cdots + f'_1\alpha + f'_0$$

이 때, 계수는 $f_i, f'_i \in GF(2)$이다 $(0 \le i \le m-1)$. 유한체 $GF(2^m)$상의 곱 $C = A \cdot B^2$는 $C(x) = A(x)B(x)^2 \bmod F(x)$를 계산함으로서 얻을 수 있다. $C(x)$는 차수가 $m-1$ 이하인 다항식이고, $C(x) \in GF(2^m)$이다. 따라서, 유한체 $GF(2^m)$상의 AB^2 연산은 대응하는 다항식의 곱을 기약 다항식 $F(x)$와 $F'(x)$로 모듈러 연산을 함으로서 수행된다. 식 1은 AB^2 연산을 위한 순환식이다.

$$\begin{aligned} P &= AB^2 \bmod F(x) \\ &= A(b_{m-1}\alpha^{2m-2} + b_{m-2}\alpha^{2m-3} + \cdots + b_1\alpha^2 + b_0) \bmod F(x) \\ &= (Ab_{m-1}\alpha^{2m-2} + Ab_{m-2}\alpha^{2m-3} + \cdots + Ab_1\alpha^2 + Ab_0) \bmod F(x) \\ &= (\cdots((\cdots((Ab_{m-1})\alpha^2 \bmod F(x) + Ab_{m-2})\alpha^2 \bmod F(x) \\ &\quad + \cdots + Ab_{m-i})\alpha^2 \bmod F(x) + \cdots + Ab_1)\alpha^2 \bmod F(x) \\ &\quad + Ab_0 \end{aligned} \tag{1}$$

이 순환식은 효율적인 AB^2 시스톨릭 어레이 구현에 효율적으로 적용될 수 있을 것이다. 식 1에서 첫 번째 항은 $Ab_{m-1}\alpha^2$이므로 식 2와 같이 표현할 수 있다.

$$\begin{aligned} P_1 &= Ab_{m-1}\alpha^2 \bmod F(x) \\ &= \left[\sum_{k=0}^{m-1} a_k b_{m-1}\alpha^k\right]\alpha^2 \bmod F(x) \\ &= \left[\sum_{k=0}^{m-1} a_k b_{m-1}\alpha^k\alpha^2\right] \bmod F(x) \\ &= \sum_{k=0}^{m-1} d_k^1\alpha^{k+2} \bmod F(x) \\ &= (d_{m-1}^1\alpha^{m+1} + d_{m-2}^1\alpha^m + d_{m-3}^1\alpha^{m-1} + \cdots + d_1^1\alpha^3 \\ &\quad + d_0^1\alpha^2) \bmod F(x) \\ &= d_{m-1}^1(f_{m-1}x^{m-1} + f_{m-2}x^{m-2} + \cdots + f_1x + f_0) \\ &\quad + d_{m-2}^1(f_{m-1}x^{m-1} + f_{m-2}x^{m-2} + \cdots + f_1x + f_0) \\ &\quad + \cdots + d_1^1\alpha^3 + d_0^1\alpha^2) \\ &= \sum_{k=0}^{m-1} p_k^1\alpha^k \end{aligned} \tag{2}$$

이 때, 다음과 같은 식을 유도할 수 있다.

$$d_k^1 = a_k b_{m-1};$$

$$p_k^1 = d_{m-1}^1 f'_k + d_{m-2}^1 f_k + d_{k-2}^1 \quad (k = 2, \cdots, m-1);$$

$$p_k^1 = d_{m-1}^1 f'_k + d_{m-2}^1 f_k; \quad (k = 0, 1)$$

그리고, 일반적인 경우의 항은

$$\begin{aligned} P_i &= (P_{i-1} + Ab_{m-i})\alpha^2 \bmod F(x) \\ &= \left[\sum_{k=0}^{m-1}(p_k^{i-1} + a_k b_{m-i})\alpha^k\right]\alpha^2 \bmod F(x) \\ &= \left[\sum_{k=0}^{m-1}(p_k^{i-1} + a_k b_{m-i})\alpha^k\alpha^2\right] \bmod F(x) \\ &= \left[\sum_{k=0}^{m-1} d_k^i\alpha^{k+2}\right] \bmod F(x) \\ &= \sum_{k=0}^{m-1} p_k^i\alpha^k \end{aligned} \tag{3}$$

와 같은데, 이 때 다음과 같은 식을 유도할 수 있다.

$$d_k^i = p_k^{i-1} + a_k b_{m-i};$$

$$p_k^i = d_{m-1}^i f'_k + d_{m-2}^i f_k + d_{k-2}^i \quad (k = 2, \cdots, m-1);$$

$$p_k^i = d_{m-1}^i f'_k + d_{m-2}^i f_k \quad (k = 0, 1);$$

마지막 항의 경우에는,

$$\begin{aligned} P_m &= P_{m-1} + Ab_0 \\ &= \sum_{k=0}^{m-1} p_k^{m-1}\alpha_k + \sum_{k=0}^{m-1} a_k b_0\alpha_k \\ &= \sum_{k=0}^{m-1}(p_k^{m-1} + a_k b_0)\alpha^k \\ &= \sum_{k=0}^{m-1} d_k^m\alpha^k = \sum_{k=0}^{m-1} p_k^m\alpha^k \end{aligned} \tag{4}$$

이 때, 다음과 같은 식을 구해낼 수 있다.

$$d_k^m = p_k^{m-1} + a_k b_0;$$

$$p_k^i = d_k^m;$$

따라서, AB^2의 결과 P는 순환 알고리즘을 사용하여 효과적으로 계산될 수 있고, 이 때 P는 P_m과 같다.

SPM 알고리즘을 비트 표현식으로 나타내면 다음과 같다.

Input : $A(\alpha)$, $B(\alpha)$, and $F(\alpha)$
Output : $P(\alpha) = A(\alpha)B(\alpha)^2 \bmod F(\alpha)$
Step1: $P_1 = Ab_{m-1}\alpha^2 \bmod F(\alpha)$
Step2: For $i = 2$ to $m-1$
$P_i = (P_{i-1} + Ab_{m-i})\alpha^2 \bmod F(\alpha)$
Step3: $P_m = P_{m-1} + Ab_0$

2.2 GF(2^m)상에서 SPM 구조

앞서 살펴본 알고리즘으로부터, 병렬 입/출력 시스톨릭 어레이 구조를 얻을 수 있다[10, 11]. 그림 1은 $GF(2^m)$ 상에서 제안된 시스톨릭 AB^2 구조를 보여준다. A, F와 F'는 위로부터 병렬로 입력이 되고, B는 가장 왼쪽 행으로부터 입력이 된다. 결과 P는 병렬로 행렬의

아래쪽으로 산출된다. 입력 A와 B는 둘 다 최상위 비트 우선 구조로 입력이 되고 출력도 같은 구조를 가진다. 이 때 최상위 비트 우선 구조는 최하위 비트 우선 구조에 비해 지수, 나눗셈, 역원 연산처럼 반복 연산을 필요로 하는 회로에 병렬성을 제공한다는 점에서 유리하다. 그림 1에서 (i, k) 셀을 지나는 선이 있다. 이 선은 $(i-1, k-1)$셀에서 $(i, k+1)$셀로 p_k^{i-1} 시그널을 전해주는 기능을 가진다. 셀이 첫 번째 열에 위치한 경우 $(k=m-1)$, p_k^{i-1}는 p_{m-1}^{i-1}과 연결하게 되고, p_{k-1}^{i-1}는 p_{m-2}^{i-1}와 연결하게 되어 그 값은 전달받는다.

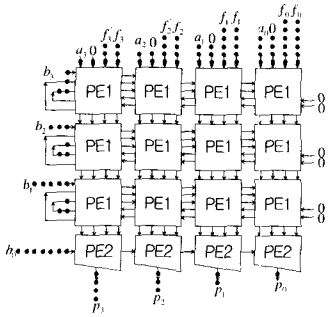
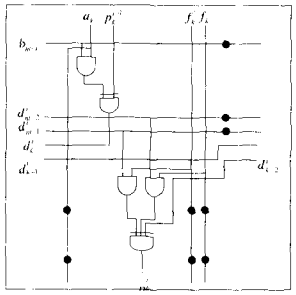
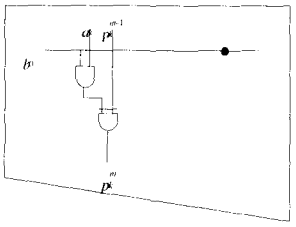


그림 1 SPM for AB^2 in GF(2^4)



(a)



(b)

그림 2 (a) Circuit for PE1(Processing Element1) (b) PE2

그림 2(a)의 PE1(Processing Element1)은 기본적인 셀들의 논리 회로를 표현하고, 그림 2(b)의 PE2는 마지막 행에 위치한 셀들의 논리 회로를 보여준다. 만일 셀이 우측 두 열에 위치하면 (즉, $k = 0$, $k = 1$), p_{k-2}^{i-1}의 값은 0이다. 마지막 행의 셀들은 모듈러 리덕션 과정없이 오직 d_k^m만 계산된다는 사실은 주목할 만하다. 그림 2에서 보는 것처럼 마지막 행에 위치한 셀들의 회로가 간단하기 때문에 이전의 구조들과 비교해 볼 때 전체 셀 복잡도를 줄일 수 있다. 마지막 행의 셀들을 제외하고 나머지 셀들의 수직 연결은 두 개씩의 지연을 필요로 하므로 전체 지연은 $3m-1$이다.

3. AB^2 연산을 위한 새로운 방식 제안

앞서 살펴본 SPM 구조는 양방향 데이터 흐름를 가진다. 역방향의 데이터 흐름을 없애기 위해 이번 장에서는 SPM 구조를 개선하여 MSPM(Modified Systolic Power Mulitplier) 과 SMSPM(Serial Modified Systolic Power Multiplier) 구조를 제안한다.

3.1 MSPM 알고리즘

개선된 AB^2 구조는 SPM을 분리·병합함으로써 얻어질 수 있다. 다음은 그림 1의 기본셀의 연산 과정을 보여준다. 아래의 식 (a)와 (b), (c)와 (d), (e)와 (f) 등의 사이에는 의존성이 없다. 즉, (i, k) 셀의 연산 d_k^i와 p_k^i는 각각 독립적이다. 반면에, (b)와 (c), (d)와 (e), …(h)와 (i) 사이에는 의존성이 있다. 다시 말하면, 식 (b)를 계산하기 위해서는 식(c)의 계산 결과가 필요하고, 식 (d)를 계산하기 위해서는 식(e)의 계산 결과가 필요하다. 이러한 사실로 (i, k) 셀의 d_k^i 연산은 $(i-1, k)$ 셀의 p_k^i연산에 의존성이 있다는 것을 알 수 있다.

$i=1$	$d_k^1 = a_k b_{m-1}$;	(a)
	$p_k^1 = d_{m-1}^1 f_k + d_{m-2}^1 f_{k+1} + d_{k-2}^1$;	(b)
$i=2$	$d_k^2 = p_k^1 + a_k b_{m-2}$;	(c)
	$p_k^2 = d_{m-1}^2 f_k + d_{m-2}^2 f_{k+1} + d_{k-2}^2$;	(d)
$i=3$	$d_k^3 = p_k^2 + a_k b_{m-3}$;	(e)
	$p_k^3 = d_{m-1}^3 f_k + d_{m-2}^3 f_{k+1} + d_{k-2}^3$;	(f)
⋮		
$i=m-1$		(g)
		(h)
$i=m$		(i)

그러므로, 구조의 성능을 향상시키기 위해서 셀의 연산들을 분리하고 병합하는 방법이 제안되었다[12,13]. 예를 들면, 식 (*a*)와 (*b*), (*c*)와 (*d*)…를 분리하고, 식(*b*)와 (*c*), (*d*)와 (*e*)를 병합하고, …식(*g*)와 (*h*)를 분리한 후, 식(*h*)와 (*i*)를 병합한다. 따라서 아래와 같은 새로운 MSPM의 순환 알고리즘을 유도할 수 있다.

i=1	$d_k^1 = a_k b_{m-1}$;	(*a*)
i=2	$p_k^1 = d_{m-1}^1 f_k' + d_{m-2}^1 f_k + d_{k-2}^1$;	(*b*)
	$d_k^2 = p_k^1+ a_k b_{m-2}$;	(*c*)
i=3	$p_k^2 = d_{m-1}^2 f_k' + d_{m-2}^2 f_k + d_{k-2}^2$;	(*d*)
	$d_k^3 = p_k^2 + a_k b_{m-3}$;	(*e*)
	$p_k^3 = d_{m-1}^3 f_k' + d_{m-2}^3 f_k + d_{k-2}^3$;	(*f*)
	⋮	
	$d_k^{m-1} = p_k^{m-2} + a_k b_1$;	(*g*)
i=m	$p_k^{m-1} = d_{m-1}^{m-1} f_k' + d_{m-2}^{m-1} f_k + d_{k-2}^{m-1}$;	(*h*)
	$d_k^m = p_k^{m-1} + a_k b_0$;	(*i*)

이 때, 제안한 알고리즘을 비트 표현식으로 나타내면 다음과 같다.

Input : $A(\alpha)$, $B(\alpha)$, and $F(\alpha)$
Output : $P(\alpha) = A(\alpha)B(\alpha)^2 \bmod F(\alpha)$
Step1: $P_1 = Ab_{m-1}$
Step2: For $i = 2$ to m
$P_i = P_{i-1} \cdot \alpha^2 \bmod F(\alpha) + Ab_{m-i}$

이로써 첫 번째 셀은 연산 $d_k^1 = a_k b_{m-1}$;만을 수행하고, 나머지 다른 셀들은 다음에 나오는 연산을 수행하게 됨을 알 수 있다.

$$p_k^{i-1} = d_{m-1}^{i-1} f_k' + d_{m-2}^{i-1} f_k + d_{k-2}^{i-1};$$

$$d_k^i = p_k^{i-1} + a_k b_{m-i};$$

식 3으로부터, 다음의 식 4를 유도할 수 있다.

$$d_k^i = d_{m-1}^{i-1} f_k' + d_{m-2}^{i-1} f_k + d_{k-2}^{i-1} + a_k b_{m-i};$$

3.2 GF(2^m)상에서 MSPM 구조

그림 3의 분리·병합 과정을 통해서 그림 5의 MSPM 구조를 얻을 수 있다.

그림 6은 그림 5에서 보여준 시스톨릭 구조의 기본적인 두 셀들을 보여주고 있다. PE3 셀은 첫 번째 행에 위치한 셀들의 연산을 보여주는 것으로 모듈러 리덕션

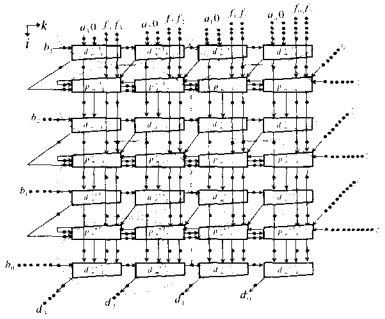


그림 3 그림 1의 분리·병합 과정

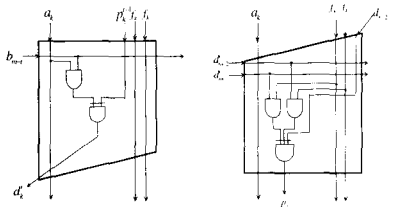


그림 4. 그림 3의 로직 회로

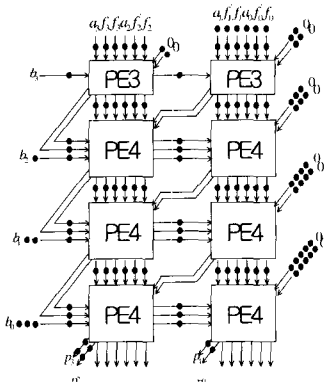
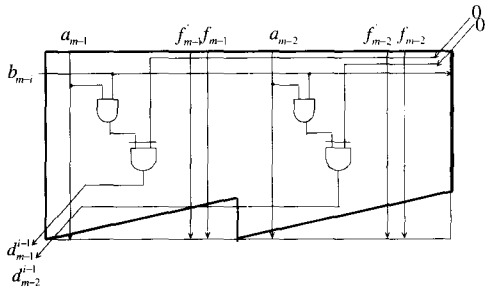
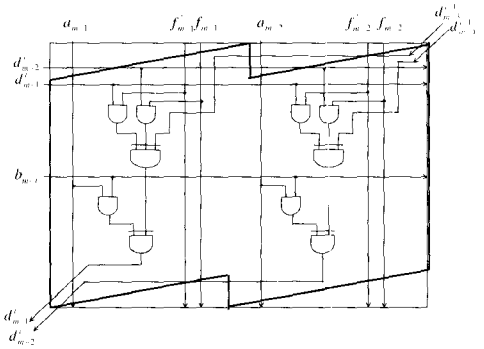


그림 5 GF(2^4) 상에서 AB^2 연산을 위한 병렬 입/출력 시스톨릭 구조



(a)



(b)

그림 6 (a) PE3 (b) PE4

과정이 없이 두 수의 곱셈 연산만이 이루어지고 있고, PE4 셀은 일반적인 셀들로 식 4 의 연산을 따르고 있다. 그림 5에서 만일 셀이 최하위 두 열에 위치한다면 (k=0, 1), d_{k-2}^{i-1}의 값은 0이다. 첫 번째 행에 위치한 셀들은 매우 간단해서 기존의 구조들보다 전체 셀 복잡도를 줄이고 있음을 보여준다. 그리고 결과값 P는 병렬로 구조의 마지막 행으로부터 얻어지고 있는데, 이 때 P는 d_m과 같다. 그림에서 보는 바와 같이, 그림 5는 그림 1 보다 데이터의 흐름이 일정하다(uni-directional)는 것을 알 수 있다.

3.3 GF(2^m)상에서 SMSPM 구조

cut-set 시스톨릭 과정 [11, 12]을 따라, 그림 1의 구조를 수평방향으로 projection하여 그림 7과 같은 시리얼 시스톨릭 구조를 얻을 수 있었다. 입력값 A, B, F와 F'는 왼쪽에서 들어가고 출력값 P는 오른쪽에서 산출된다. 이처럼, 입력과 출력값은 모두 최상위 비트 우선(MSB) 원칙을 따른다. 그림 8에서는 셀들의 논리 회로를 보여준다. 그림 8(a)는 일반적인 셀의 로직 회로를 나타내는데, 세 개의 멀티플렉서(MUXs)가 i번 째 셀에서 d_{m-1}^i, d_{m-2}^i, b_{m-1} 값을 저장하기 위해 필요하다. 두 개의 컨트롤 시그널(CSs), CS1과 CS2는 계산 도중 위의 세 값을 유지하고 리플레쉬하는데 쓰여진다. CS1은 m 개의 비트 중에 첫 비트만 1값을 가지고 나머지는 0값을 가지며, CS2는 첫 세 비트가 1의 값을 가진다. 그림 8(b)는 구조의 최우측 셀의 로직 회로를 나타내고 있는데, d_k^m 연산만 수행을 하므로 로직이 간단하다.

4. 시뮬레이션 및 성능 분석

본 장에서는 2, 3장에서 제안한 구조들을 시뮬레이션 하여 정확성을 검증하고, 그 성능을 분석하여 관련된 구조들과 비교하였다.

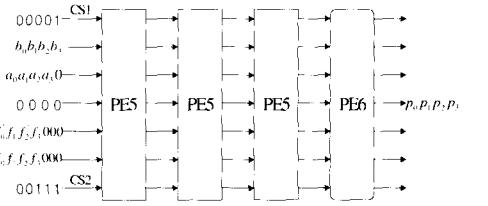
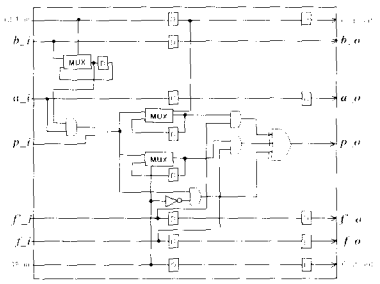
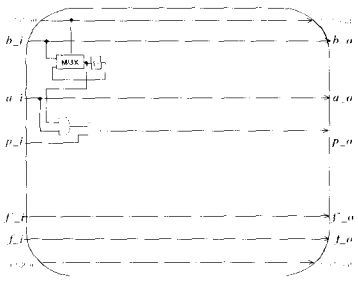


그림 7 GF(2^1) 상에서 AB^2 연산을 위한 시리얼 입/출력 시스톨릭 구조



(a)



(b)

그림 8 (a) PE5 (b) PE6

4.1 시뮬레이션

제시한 구조들의 정확성을 검증하기 위해 ALTERA 사의 MAX+PLUSⅡ시뮬레이터와 FLEX 10K 디바이스로 시뮬레이션 하였다. 그림 9와 10은 각각 GF(2^1)상에서 병렬 입/출력, 시리얼 입/출력 AB^2의 시뮬레이션 결과를 보여준다. 입력값은 A, B, F와 F'이고, 출력값은 P이다. 입력값 $A(x)=x^3+x=(1010)$와 $B(x)=x^3+x=(1010)$은 GF(2^1)의 두 원소이다. 그리고 $F(x)=x^4+x+1$ 이고, $F'(x)=x^5+x^2+x$은 GF(2^1)의 두 개의 기약 다항식이다. 만일 기약 다항식의 근이 α라면 $F(\alpha)\equiv\alpha^4+\alpha+1=(0011)$, $F'(\alpha)\equiv\alpha^5+\alpha^2+\alpha=(0110)$이 입력된다. 결과값 $P(x)$는 $AB^2 \bmod F(x) = x^3+x^2+x+1=(1111)$ 값을 얻을

수 있는데, 병렬 입/출력 구조인 MSPM은 $m+\lceil \frac{m}{2} \rceil$ = 6 clock cycle후에 모든 결과값을 얻을 수 있고, 시리얼 입/출력 구조인 SMSPM은 $3m-2$ = 10 clock cycle 후에 모든 결과값을 얻을 수 있었다.

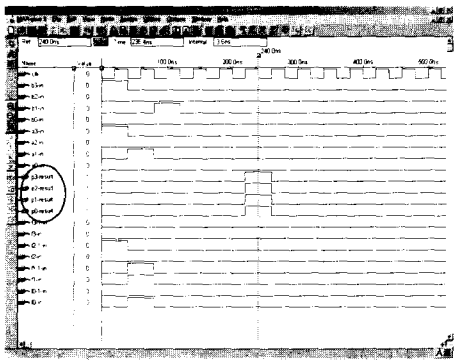


그림 9 GF(2^4)상에서 병렬 입/출력 구조인 MSPM (AB^2)의 시뮬레이션 결과

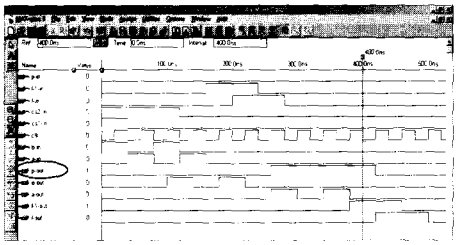


그림 10 GF(2^4)상에서 시리얼 입/출력 구조인 SMSPM (AB^2)의 시뮬레이션 결과

4.2 성능 분석

다음의 표에서는 본 논문에서 제안한 AB^2 구조와 시스톨릭 기반 파워썸 (AB^2+C) 구조들을 비교하고 있다. 이미 제안된 연산기들은 여러 구조로 구현이 되었다. 그 중 LFSR(Linear Feedback Shift Register)구조는 지연 시간이 m이고 공간 복잡도 면에서 단순하다는 장점이 있으나 곱셈기의 속성상 임계 경로가 $T_{AND2}+(\log m)T_{XOR2}$인 속성을 가지므로 길어진다. 그에 반해 시스톨릭 구조는 임계 경로가 짧고($T_{AND2}+T_{XOR4}$), 구조의 단순성(simplicity), 규칙성(regularity), 모듈성(modularity) 등의 특성을 가지므로 VLSI 구현에 적합하고 파이프라닝 (pipelinabiliy)이 가능하여 지수/역원 구현에 적합하다. 그러나, 전체 지연 시간이 LFSR에 비해 길어진다.

또한, 이미 제안된 AB^2 연산기들의 공통적인 특징을 살펴보면[14, 15, 16], 기약 다항식으로 계수가 모두 1인 m차의 기약다항식 AOP(All One Polynomial)를 주로 사용한다. AOP 기반의 구조는 하드웨어 복잡도와 지연 시간에 있어 우수성을 보이나 기약 다항식의 계수가 모두 1인 특수한 상황에서만 쓰일 수 있는 것이므로 일반적이지 못하다.

본 논문에서는 나눗셈/역원 연산을 위한 AB^2 연산기를 제안하였으나, 기존의 논문에서 일반 기약 다항식을 이용한 시스톨릭 AB^2 연산기의 구현이 미미하여 AB^2+C 구조와 비교하였다. AB^2 구조에서 AB^2+C 구조를 구현하기 위해서는 하드웨어 복잡도와 지연 시간에 영향을 미치지 않고 본 논문의 MSPM구조 우측에서 입력되는 $d_0^{-1}, d_1^{-1}, d_2^{-1}, d_3^{-1}$ 대신에 C 값을 입력하면 되므로 비교에 무리가 없다.

표1은 제안된 병렬 입/출력 구조와 그와 관련된 구조[7]를 비교하고 있다. Wang[8]의 논문에 따르면 Wei[7]의 논문은 정확하지 못한 부분이 있다. 즉, 회로의 각각의 셀에 세 개의 1비트 래치들을 추가해야 한다. 따라서 본 논문은 정확한 비교를 위해 Wang[8]의 가정을 따른다. 이 때, AND와 XOR는 2-input AND와 2-input XOR를 각기 지칭한다. 그리고 3-input XOR와 4-input XOR 게이트는 각각 두 개와 세 개의 2-input XOR 게이트로 수행될 수 있다. 표1에서 알 수 있는 바와 같이, Wei[7] 구조의 셀 복잡도는 m^2(3AND+3XOR +13Latches)인 반면, 제안된 SPM 구조는 m^2(3AND+3XOR)-m(2AND+2XOR)+($9m^2-6m-1$) Latches의 셀 복잡도를 가진다. 이것은 제안된 구조가 m(2AND+2XOR)+($4m^2+6m+1$) Latches 만큼의 셀 복잡도를 줄였음을 보여준다. 그리고 Wei 구조의 지연은 $4m$인데 반해, 제안된 구조는 $3m-1$의 지연을 가진다.

표 2에서는, Wang[8] 구조의 셀 복잡도가 m^2(3AND+3XOR+9.5Latches)인 반면, 제안된 MSPM 구조의 셀 복잡도는 m^2(3AND+3XOR)-m(2AND+2XOR)+($4.5m^2-2m-3$)Latches를 가짐을 보여준다. 따라서, MSPM은 Wang[8]에 비해 m(2AND+2XOR)+($5m^2+2m+3$)Latches의 셀 복잡도를 줄였다. 그리고, Wang[8] 구조의 지연은 $2m+m/2$인 반면, MSPM은 $m+m/2$의 지연을 가진다. 따라서, MSPM은 Wang[8]의 구조에 비해 지연 시간을 40% 줄일 수 있었다. 표 3은 Danial[17]이 정의한 게이트의 트랜지스터 수와 지연 시간에 따라 두 구조의 AT Product를 비교하고 있다.

표 4는 SMSPM 구조의 셀 복잡도가 m(4AND+3XOR+15Latch)-(3AND+2XOR+14Latch)이고, 지연은 $3m-2$임을 보여준다. Yeh[13]는 AB 연산을 하므로 본

논문의 AB^2의 연산보다는 하드웨어 복잡도가 낮다.

표 1 GF(2^m)상의 병렬 입/출력 양방향(bi-directional) 시스톨릭 구조의 비교

Item \ Circuit	Wei[7]	SPM	
No. of cells	m^2	m^2	
Throughput	1	1	
Function	AB^2+C	AB^2	
Latency	$4m$	$3m-1$	
Computation time per basic cell	T_{AND2} + T_{XOR3}	T_{AND2} + T_{XOR3}	
Cell complexity	3 2-input AND 1 2-input XOR 1 3-input XOR 13 1-bit latches	PE1	3 2-input AND 1 2-input XOR 1 3-input XOR 11 1-bit latches
		PE2	1 2-input AND 1 2-input XOR 1 1-bit latch
Algorithm Fashion	LSB	MSB	

표 2 GF(2^m)상의 병렬 입/출력 단방향(uni-directional) 시스톨릭 구조의 비교

Item \ Circuit	Wang[8]	MSPM	
No. of cells	$m^2/2$	$m^2/2$	
Throughput	1	1	
Function	AB^2+C	AB^2	
Latency	$2m+m/2$	$m+m/2$	
Computation time per basic cell	T_{AND2} + T_{XOR4}	T_{AND2} + T_{XOR3}+ T_{XOR2}	
Cell complexity	6 2-input AND 2 4-input XOR 17 1-bit latches	PE3	6 2-input AND 2 2-input XOR 2 3-input XOR 9 1-bit latches
		PE4	2 2-input AND 2 2-input XOR 7 1-bit latches
Algorithm Fashion	MSB	MSB	

표 3 Area-Time Product 비교(Φ:트랜지스터 수 Δ: 게이트 지연 시간(ns) [17])

Item \ Circuit	Wang[8]	MSPM
Area Complexity	$230m^2\Phi$	$(150m^2-60m)\Phi$
Time Complexity	$7.5m^2\Delta$	$7.5m^2\Delta$
AT Product	$1725m^4\Phi\Delta$	$(1125m^4-450m^3)\Phi\Delta$

표 4 GF(2^m)상의 시리얼 입/출력 시스톨릭 구조의 비교

Item \ Circuit	Yeh[14]	SMSPM	
No. of cells	m	m	
Throughput	$1/m$	$1/m$	
Fuction	AB	AB^2	
Latency	$3m$	$3m-2$	
Cell complexity	3 2-input AND 2 2-input XOR 11 1-bit latches 1 switch	PE5	4 2-input AND 1 2-input XOR 1 3-input XOR 14 1-bit latches 3 switches
		PE6	1 2-input AND 1 2-input XOR 1 1-bit latch 1 switch
No. control signals	2	2	

5. 결 론

본 논문은 GF(2^m)상에서 새로운 AB^2 알고리즘과 그에 따른 병렬 입/출력 시스톨릭 구조인 SPM, MSPM, 시리얼 입/출력 시스톨릭 구조인 SMSPM를 제안하였다. 제안한 알고리즘은 다항식 기저로 표현하였고, MSB 구조을 가져 나눗셈/역원 구조에 적용하였을 때 병렬성을 가진다. 또한 제안한 구조들을 기존의 구조들에 비해 셀 복잡도가 낮고 지연이 적다. 본 논문에서 제안한 MSPM과 SMSPM 구조는 역방향 데이터 흐름을 줄여서 단방향 데이터 흐름을 가진다. 덧붙여, 이 구조들은 단순성, 규칙성, 모듈성, 병렬성을 가져 VLSI 구현에 적합하고 나눗셈/역원 구조에 쉽게 적용이 가능하다.

참 고 문 헌

[1] W.W.Peterson and E.J.Weldon, *Error-correcting codes*, MIT Press, MA, 1972.

[2] D.E.R.Denning, *Cryptography and data security*, Addison-Wesley, MA, 1983.

[3] A.Menezes, *Elliptic Curve Public Key Cryptosystems*, Kluwer Academic Publishers, Boston, 1993.

[4] T.ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithms, *IEEE Trans. on Info. Theory*, vol. 31(4), pp. 469-472, July 1985.

[5] I.S.Reed and T.K.Truong, The use of finite

fields to compute convolutions, *IEEE Trans. Inform. Theory*, 21, pp.208-213, 1975.

[6] S.W.Wei, VLSI architectures for computing exponentiations, multiplicative inverses, and divisions in GF(2*m*), *Proc. IEEE Trans. Circuits and Systems*, 44, pp.847-855, 1997.

[7] S.W.Wei, A Systolic Power-Sum Circuit for GF(2m), *IEEE Trans. Computers*, 43, pp.226-229, 1994.

[8] C.L.Wang and J.H.Guo, New systolic arrays for *C+AB*2, inversion, and division in GF(2m), *IEEE Trans. Computers*, 49, pp.1120-1125, 2000.

[9] J.V.McCanny, R.A.Evans and J.G.Mcwhirter, Use of unidirectional data flow in bit-level systolic array chips, *Electron.Lett.*, 22, pp. 540-541, 1986.

[10] Nam-Yeun Kim and Kee-Young Yoo, "A Power-Sum Systolic Architecture in GF(2^m)," *Lecture Notes in Computer Science VOL 2344 Information Networking. Wireless Communications Technologies and Network Applications (LNCS 2344)*, pp. 409-417, Feb. 2002.

[11] S.Y.Kung, *VLSI Array Processors*, Prentice-Hall, 1987.

[12] K.Y.Yoo, *A Systolic Array Design Methodology for Sequential Loop Algorithms*, Ph.D. thesis, Rensselaer Polytechnic Institute, New York, 1992.

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

[14] C.H.Liu, N.F.Huang, and C.Y.Lee, 'Computation of AB^2 Multiplier in GF(2m) Using an Efficient Low-Complexity Cellular Architecture,' IEICE trans. fundamentals, Vol. E83-A, No. 12 December 2000.

[15] 이형목, 김현성, 전준철, 유기영, 'GF(2m)상에서 AB^2연산을 위한 세미시스톨릭 구조,' 정보보호학회 논문지 제 12권 제2호, 2002년 4월

[16] H.S.Kim,'Bit-Serial AOP Arithmetic Architecture for Modular Exponentiation,' Ph.D. thesis, Kyungpook National University, 2001.

[17] Daniel D. Gajski, *Principles of Digital Design*, Prentice-hall international, INC. 1997.

김 남 연
1995년 대구교육대학교 교육학사. 1999년 계명대학교 전산교육전공 석사. 2001년 ~ 현재 경북대학교 컴퓨터공학과 박사 과정. 관심분야는 Arithmetic 알고리즘, 암호학, VLSI array processors 설계

고 대 곤
1974년 연세대학교 물리학 이학사. 1981년 단국대학교 컴퓨터시뮬레이션 공학석사. 1989년 연세대학교 인공지능, CAI 공학박사. 1989년 ~ 현재 대구교육대학교 전산교육과 교수로 재직. 관심분야는 인공지능, ICAI, 컴퓨터 교육

유 기 영
1976년 경북대학교 수학교육학과 졸업(이학사). 1978년 한국과학기술원 전산학과 졸업(공학석사). 1992년 미국 Rensselarer Polytechnic Institute 졸업(이학박사). 1978년 ~ 현재 경북대학교 컴퓨터공학과 교수로 재직. 관심분야는 병렬처리, DSP array processor 설계, 암호화 등