$GF(2^m)$상의 LSD 우선 곱셈을 위한 새로운 시스톨릭 어레이

정회원 김 창 훈*, 남 인 길*°

A New Systolic Array for LSD-first Multiplication in $GF(2^m)$

Chang Hoon Kim* In Gil Nam*° ***Regular Members***

요 약

본 논문에서는 암호 응용을 위한 $GF(2^m)$상의 새로운 디지트 시리얼 시스톨릭 곱셈기를 제안한다. 제안된 곱셈기는 연속적인 입력 데이터에 대해 $\lceil m/D \rceil$ 클럭 사이클마다 곱셈 결과를 출력한다. 여기서 D는 선택된 디지트 크기이다. 기존에 제안된 구조들은 선형의존성 때문에 디지트 크기 D가 증가하면 최대 처리기 지연시간 역시 선형으로 증가하지만 제안된 곱셈기는 이진트리 형태의 내부 구조를 가지기 때문에 D에 대해 로그단위로 증가한다. 따라서 제안된 구조는 기존에 제안된 디지트 시리얼 시스톨릭 곱셈기에 비해 계산지연을 상당히 감소시킨다. 뿐만 아니라 제안된 곱셈기는 규칙성, 모듈성, 단방향 신호 흐름의 특성을 가지기 때문에 VLSI 구현에 매우 적합하다.

Key Words : Finite Field Multiplication, Digit-Serial Architecture, Systolic Array, VLSI

ABSTRACT

This paper presents a new digit-serial systolic multiplier over $GF(2^m)$ for cryptographic applications. When input data come in continuously, the proposed array produces multiplication results at a rate of one every $\lceil m/D \rceil$ clock cycles, where D is the selected digit size. Since the inner structure of the proposed array is tree-type, critical path increases logarithmically proportional to D. Therefore, the computation delay of the proposed architecture is significantly less than previously proposed digit-serial systolic multipliers whose critical path increases proportional to D. Furthermore, since the new architecture has the features of regularity, modularity, and unidirectional data flow, it is well suited to VLSI implementations.

Ⅰ. 서 론

최근 유한 필드 $GF(2^m)$상의 연산들은 오류 제어 코딩, 암호응용 등 여러 분야에서 중요한 역할을 하고 있다[1,2]. $GF(2^m)$상에서 중요한 연산은 덧셈, 곱셈, 지수, 나눗셈이 있다. $GF(2^m)$상의 덧셈은 비트별 배타적 논리합(XOR) 연산으로 적은 비용의 고속 구현이 가능하지만 다른 연산들은 상당히 복잡할 뿐만 아니라 구현에 따른 비용이 크다. 곱셈 연산은 $GF(2^m)$ 응용분야에서 가장 많이 사용되는 연산일 뿐만 아니라 곱셈의 반복적인 연산을 통하여 지수 및 나눗셈을 수행할 수 있다. 따라서 본 논문은 $GF(2^m)$상의 고속 디지트 시리얼 곱셈기 설계에 초점을 맞춘다.

* 대구대학교 컴퓨터·IT공학부(kimch@daegu.ac.kr), (ignam@daegu.ac.kr) (°:교신저자)
논문번호 : KICS2007-09-391, 논문접수 : 2007년 9월 1일, 최종논문접수일자 : 2008년 4월 18일

$GF(2^m)$상의 효율적인 곱셈기 구현에 대해 많은 연구가 이루어져 왔다[3-14]. 기존의 곱셈기들은 서로 다른 기저를 사용하였는데 가장 대표적인 $GF(2^m)$ 원소표기법에는 정규기저(normal basis)와 다항식기저(polynomial basis)가 있다. 각 기저 표기법은 장·단점을 가지는데, 정규기저 표기법을 사용할 경우 제곱연산이 쉽게 되는 반면 곱셈연산이 매우복잡하고 서로 다른 m에 대해 규칙적인 하드웨어 구조 설계가 어렵다. 이러한 이유로 $GF(2^m)$상의 곱셈에 대한 하드웨어 구현에는 다항식기저 표기법이 더 많이 사용된다.

최근 Song 등[11]은 $GF(2^m)$상에서 특별한 기약다항식($G(x)=x^m+x^k+\sum_{i=0}^{k-1}g_ix^i$, $D\le m-k$)을 사용한 고속의 저비용 디지트 시리얼/패러럴 곱셈기를 제안하였다. 여기서 D는 선택된 디지트 크기이다. 위의 조건을 만족하는 기약다항식을 사용한다면 $A(x)x^D \bmod G(x)$ 연산은 비트별 AND 게이트와 XOR 게이트의 이진트리[11]를 이용한 계산이 가능하다. 여기서 $A(x)$는 $GF(2^m)$상의 원소이다. 일반적으로 Trinomial, Pentanomial, all one polynomial과 같은 특별한 기약다항식은 저면적 및 고속의 곱셈기 구현이 가능하지만 모든 m에 대하여 존재하지 않는다[16]. 이와 달리 Song 등이 사용한 기약다항식은 D를 적당하게 선택하면 모든 m에 대해 존재한다. 예를 들면, 타원곡선 암호 시스템을 위해 NIST[16]에서 권고하는 모든 기약다항식의 경우 D의 크기는 128까지 가능하다. 따라서 실제적인 응용에 있어서는 제약조건이 거의 없다 할 수 있다.

Song 등에 의해 제안된 곱셈기가 하드웨어 면적, 전력 소모, 기약다항식의 선택 등 많은 장점을 가지지만 많은 글로벌 신호의 브로드캐스팅을 포함하기 때문에 현저한 속도 저하를 보인다.

본 논문에서는 암호 응용을 위한 $GF(2^m)$상의 새로운 디지트 시리얼 시스톨릭 곱셈기를 제안한다. LSD(Least Significant Digit) 우선 곱셈 알고리즘[11]으로부터 디지트-레벨의 새로운 자료의존 그래프를 유도하고 이에 기반한 새로운 디지트 시리얼 시스톨릭 어레이를 설계한다. 제안된 곱셈기는 연속적인 입력 데이터에 대해 매번 $\lceil m/D \rceil$ 클럭 사이클마다 곱셈 결과를 출력한다. 또한 기존에 제안된 구조들은 선형의존성 때문에 디지트 크기 D가 증가하면 최대 처리기 지연시간 역시 선형으로 증가하지만 제안된 곱셈기는 이진-트리 형태의 내부 구조를 가지기 때문에 D에 대해 로그단위로 증가한다. 따라서 제안된 구조는 기존에 제안된 디지트 시리얼 시스톨릭 곱셈기에 비해 속도측면에서 상당한 개선을 보인다. 뿐만 아니라 제안된 구조는 높은 규칙성, 모듈성, 단방향 신호 흐름의 특성을 가지기 때문에 VLSI 구현에 매우 적합하다.

Ⅱ. $GF(2^m)$상의 디지트 레벨 곱셈 알고리즘

2.1 $GF(2^m)$상의 LSD 우선 곱셈 알고리즘

$A(x)=\sum_{i=0}^{m-1}a_ix^i$, $B(x)=\sum_{i=0}^{m-1}b_ix^i$를 $GF(2^m)$상의 두 원소라 하고 $G(x)=x^m+\sum_{i=0}^{m-1}g_ix^i$를 유한체 $GF(2^m)\cong GF(2)[x]/G(x)$를 생성하는 기약다항식이라 하면, $A(x)B(x) \bmod G(x)$의 결과 값은 $P(x)=\sum_{i=0}^{m-1}p_ix^i$로 나타낼 수 있다. 또한 D는 디지트 크기이고 $N=\lceil m/D \rceil$을 디지트의 전체 개수라 하면, 디지트 $A_i(0\le i\le N-1)$는 다음과 같이 정의할 수 있다.

$$A_i=\begin{cases}\sum_{j=0}^{D-1}a_{Di+j}x^j, & 0\le i\le N-2\\ \sum_{j=0}^{m-1-D(N-1)}a_{Di+j}x^j, & i=N-1\end{cases} \quad (1)$$

디지트 B_i, G_i, P_i 역시 A_i와 유사하게 정의할 수 있으며, 곱셈 $A(x)B(x) \bmod G(x)$를 계산하기 위해 아래와 같은 LSD 우선 방식을 사용할 수 있다.

$$\begin{aligned}P(x) &= A(x)B(x) \bmod G(x)\\ &= B_0A(x)+B_1[A(x)x^D \bmod G(x)]\\ &\quad +B_2[A(x)x^{2D} \bmod G(x)]\\ &\quad +\cdots+B_{N-1}[A(x)x^{D(N-1)} \bmod G(x)]\end{aligned} \quad (2)$$

식 (2)로 부터 아래의 LSD 우선 곱셈 알고리즘[11]을 얻을 수 있다.

[알고리즘 1] : $GF(2^m)$상의 LSD 우선 곱셈 알고리즘[11]

Input : $G(x)$, $A(x)$, $B(x)$
Output : P has $P(x)=A(x)B(x) \bmod G(x)$
Initialize : $A=A^{(0)}=A(x)$, $B=B(x)$, $G=G(x)$, $P=P^{(0)}=0$
1. for $i=1$ to N do
2. $A^{(i)}=A^{(i-1)}x^D \bmod G$
3. $P^{(i)}=B_{i-1}A^{(i-1)}+P^{(i-1)}$
4. end for
5. $P=P^{(N)} \bmod G$

2.2 $GF(2^m)$상의 mod G(x) 연산

2.1절의 곱셈 알고리즘에서 가장 중요한 연산은 $A(x)x^D$ mod G(x)로서 최 고차 항은 기껏해야 m+D-1 이고 x를 기약다항식 $G(x)=x^m+x^k+\sum_{i=0}^{k-1}g_ix^i$의 근 이라하면 $x^m=x^k+\sum_{i=0}^{k-1}g_ix^i$이다. 여기서 k는 기약다항식에서 두 번째로 높은 차수이다. 따라서 $A(x)x^D$ mod G(x) 연산은 아래 식(3)과 같이 계산 할 수 있다.

$$
\begin{aligned}
&A(x)x^D \bmod G(x) \qquad (3)\\
&=(a_{m-1}x^{m+D-1}+a_{m-2}x^{m+D-2}+\ldots+a_0x^D) \bmod G(x)\\
&=a_{m-1}x^mx^{D-1}+a_{m-2}x^mx^{D-2}+\ldots+a_0x^D\\
&=a_{m-1}x^{D-1}(x^k+\sum_{j=0}^{k-1}g_jx^j)\\
&+a_{m-2}x^{D-2}(x^k+\sum_{j=0}^{k-1}g_jx^j)\\
&\vdots\\
&+a_{m-D-1}x^{m-1}+a_{m-D-2}x^{m-2}+\ldots+a_0x^D
\end{aligned}
$$

위의 식 (3)에서 a_ix^m $(m-D\le i\le m-1)$ 연산(곱셈)은 각 항의 연산 결과가 다른 항에 영향을 미치지 않기 때문에 일반적인 기약다항식을 사용하여도 패러럴한 연산 수행이 가능하다. 그러나 최종적인 연산 결과를 얻기 위해서는 각 항에 해당하는 임시결과의 덧셈이 필요하다. 이 경우 $m-D\le i\le m-1$에 대한 a_ix^{i+D}항의 임시 결과의 최 고차 항은 m보다 크거나 같기 때문에 다른 모든 항에 영향을 미친다. 즉, 패러럴한 연산 수행이 불가능하다. 따라서 [4-5]의 결과에 나타나듯이 일반적인 기약다항식을 사용하는 디지트 시리얼 곱셈기의 Critical Path는 D에 선형적으로 증가한다.

위에서 언급한 선형 의존성은 $G(x)=x^m+x^k+\sum_{i=0}^{k-1}g_ix^i$, $D\le m-k$ 같은 특별한 기약 다항식을 사용하면 쉽게 해결 될 수 있다. 위의 식 (3)에서 $D\le m-k$ 조건이 만족 한다면 임시 결과의 모든 차수는 m-1 보다 작거나 같다. 즉, 임시 결과의 덧셈은 이진트리를 이용하여 계산할 수 있다. 따라서 $D\le m-k$ 조건을 만족하는 기약 다항식을 선택한다면 디지트 시리얼 곱셈기의 Critical Path는 D에 대해 로그 단위로 증가한다. $D\le m-k$ 조건은 m이 작은 경우 D의 크기를 높게 정의할 수 없지만 암호 응용의 경우 m은 적어도 163 이상이어야 하기 때문에 위의 조건을 가지는 기약다항식은 쉽게 찾을 수 있다. 아래의 표 1에 나타나듯이 타원곡선

표 1. NIST[16] 권장 필드 크기 및 기약다항식

m	기약다항식
163	$x^{163}+x^7+x^6+x^3+1$
233	$x^{233}+x^{74}+1$
283	$x^{283}+x^{12}+x^7+x^5+1$
407	$x^{407}+x^{87}+1$
571	$x^{571}+x^{10}+x^5+x^2+1$

암호시스템을 위해 NIST[16]에서 권고하는 기약다항식의 경우 $m-k$가 매우 크기 때문에 실용적인 측면에서 고려하면 D의 선택에 대한 제약 사항은 거의 없다 할 수 있다.

Ⅲ. LSD 우선 곱셈을 위한 새로운 자료의존 그래프

3.1 핵심 연산

[알고리즘 1]에서 $A^{(i)}=A^{(i-1)}x^D \bmod G$와 $P=P^{(N)} \bmod G$는 입력값을 제외하면 완전히 동일한 연산이다. 따라서 $A^{(i)}=A^{(i-1)}x^D \bmod G$와 $P^{(i)}=B_{i-1}A^{(i-1)}+P^{(i-1)}$연산을 고려하면 된다. 위의 식에서 $D\le(m-k)$이면 $A^{(i)}=A^{(i-1)}x^D \bmod G$ 연산은 비트별 AND 게이트와 XOR 게이트의 이진 트리를 이용해서 계산할 수 있다[11]. 여기서 k는 $G(x)$의 두 번째로 가장 높은 차수이다. 그림 1은 $m=9$이고 $D=3$인 경우의 예를 보인다.

$A = a_8x^8 + a_7x^7 + a_6x^6 + a_5x^5 + a_4x^4 + a_3x^3 + a_2x^2 + a_1x + a_0$

$G = x^9 + g_5x^5 + g_4x^4 + g_3x^3 + g_2x^2 + g_1x + g_0$

$A\cdot x^3 = a_8x^{11} + a_7x^{10} + a_6x^9 + a_5x^8 + a_4x^7 + a_3x^6 + a_2x^5 + a_1x^4 + a_0x^3$

$A' = A\cdot x^3 \bmod G$

$= a_8x^2x^9 + a_7xx^9 + a_6x^9 + a_5x^8 + a_4x^7 + a_3x^6 + a_2x^5 + a_1x^4 + a_0x^3$

a_5	a_4	a_3	a_2	a_1	a_0	0	0	0
⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
0	a_8g_5	a_8g_4	a_8g_3	a_8g_2	a_8g_1	a_8g_0	0	0
⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
0	0	a_7g_5	a_7g_4	a_7g_3	a_7g_2	a_7g_1	a_7g_0	0
⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕	⊕
0	0	0	a_6g_5	a_6g_4	a_5g_3	a_6g_2	a_6g_1	a_6g_0

$A' = a'_8x^8 + a'_7x^7 + a'_6x^6 + a'_5x^5 + a'_4x^4 + a'_3x^3 + a'_2x^2 + a'_1x + a'_0$

그림 1. $m=9$, $D=3$일 때, $A(x)x^D \bmod G(x)$ 연산

위의 그림에서 기술된 바와 같이 $A'=A\cdot x^D \bmod G$라 하면 계수 A_i'는 다음 식을 사용해서 계산할 수 있다.

$$A_i'=\sum_{i=0}^{N-1}\sum_{j=0}^{D-1}\left(\sum_{k=1}^{D}(a_{m-k}g_{D(i-1)+j+k})+a_{D(i-1)+j}\right) \qquad (4)$$

여기서 $A_{-1}=G_{N-1}=G_{-1}=0$이다.

$$
\begin{array}{cccccccccccc}
p_{11} & p_{10} & p_9 & p_8 & p_7 & p_6 & p_5 & p_4 & p_3 & p_2 & p_1 & p_0 \\
\oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus \\
0 & b_2a_8 & b_2a_7 & b_2a_6 & b_2a_5 & b_2a_4 & b_2a_3 & b_2a_2 & b_2a_1 & b_2a_0 & 0 & 0 \\
\oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus \\
0 & 0 & b_1a_8 & b_1a_7 & b_1a_6 & b_1a_5 & b_1a_4 & b_1a_3 & b_1a_2 & b_1a_1 & b_1a_0 & 0 \\
\oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus & \oplus \\
0 & 0 & 0 & b_0a_8 & b_0a_7 & b_0a_6 & b_0a_5 & b_0a_4 & b_0a_3 & b_0a_2 & b_0a_1 & b_0a_0
\end{array}
$$

$$P'_{=}\ p'_{11}x^{11}+p'_{10}x^{10}+p'_9x^9+p'_8x^8+p'_7x^7+p'_6x^6+p'_5x^5+p'_4x^4+p'_3x^3+p'_2x^2+p'_1x+p'_0$$

그림 2. $m=9$, $D=3$일 때, B_iA+P 연산

이고 $D=3$인 경우 $P'=B_iA+P$의 계산 예를 나타낸다.

그림 2에 나타나듯이 P의 차수가 기껏해야 $(m+D-2)$이지만, 디지트-레벨의 동일한 기본 셀들을 얻기 위해 부가적인 1-비트를 추가한다. 따라서 $P'=B_iA+P$는 아래 식 (5)를 통하여 얻을 수 있다.

$$P_i' = \sum_{i=0}^{N}\sum_{j=0}^{D-1}\left(\sum_{k=1}^{D}\left(b_{D-k}a_{D(i-1)+j+k}\right)+p_{Di+j}\right) \quad (5)$$

여기서 $A_{-1}=G_{N-1}=G_{-1}=0$이다.

3.2 디지트 시리얼 곱셈을 위한 새로운 자료의존 그래프

식 (5)로 부터 $A^{(i-1)}x^D \bmod G$을 위한 자료의존 그래프는 아래 그림 3과 같이 얻을 수 있다.

[알고리즘 1]에서 $A^{(N-1)}x^D \bmod G$의 결과를 위한 마지막 반복 계산은 곱셈 결과에 영향을 미치지 않기 때문에 자료의존 그래프는 디지트-레벨의 $(N-1)\times N$ 기본 셀로 구성할 수 있다. 그림 3은

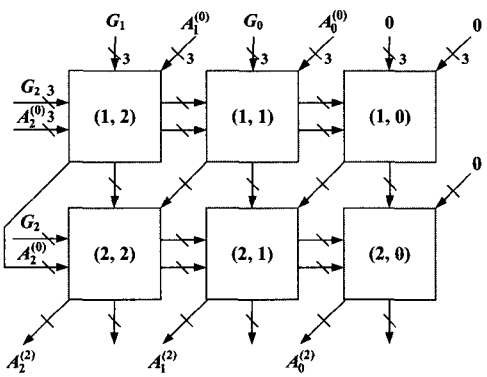

그림 3. $A^{(i-1)}x^D \bmod G$ 연산을 위한 자료의존 그래프 $(m=9,\ D=3)$

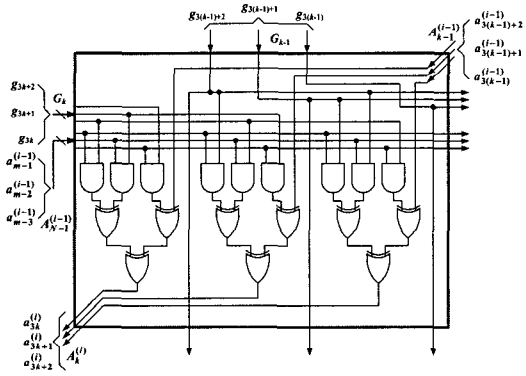

그림 4. 그림 3의 (i,k) 계산점 회로도

$m=9$, $D=3$으로 가정 하였으며, 그림 4는 그림 3의 기본 셀 구조를 나타낸다. 그림 3과 [알고리즘 1]의 비교를 통해 그림 3의 i번째 행은 [알고리즘 1]의 i번째 $A^{(i-1)}x^D \bmod G$ 연산을 수행함을 알 수 있다.

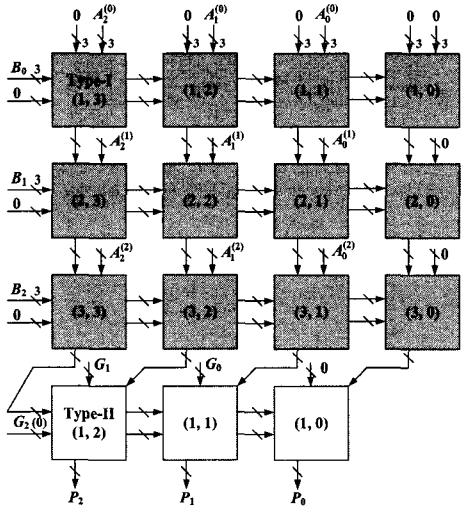

그림 5. [알고리즘 1]의 단계 3, 5를 위한 자료의존 그래프$(m=9,\ D=3)$

그림 5는 $P^{(i)}=B_{i-1}A^{(i-1)}+P^{(i-1)}$와 $P=P^{(N)} \bmod G$를 위한 새로운 자료의존 그래프를 나타낸다. 그림 5의 자료의존 그래프는 디지트-레벨의 $(N+1)\times N$개의 Type-1 셀과 N개의 Type-2 셀로 구성된다. 그림 5에서 $m=9$이고 $D=3$이다. 그림 5에 나타나듯이 i번째 행의 Type-1 셀은 $P^{(i)}=B_{i-1}A^{(i-1)}+P^{(i-1)}$를, $(N+1)$번째 행의 Type-2 셀은 $P=P^{(N)} \bmod G$를 각각 계산한다. 그림 6은 Type-1 셀의 구조를 나타낸다. 또한 $A^{(i-1)}x^D \bmod G$와 $P=P^{(N)} \bmod G$는 동일한 연산으로 그림 5의 Type-2 셀 구조는 그림 3의 기본 셀과 동일하다.

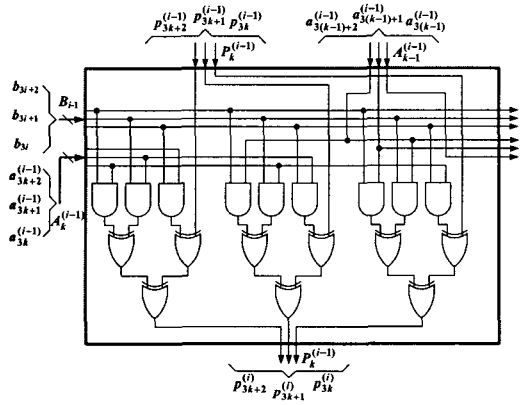

그림 6. 그림 5의 (i,k) 계산점 회로도

Ⅳ. $GF(2^m)$상의 새로운 디지트 시리얼 시스톨릭 곱셈기

그림 3과 그림 5에 나타나듯이 모든 데이터 흐름은 수평으로 단방향이다. 따라서 두 개의 자료의존 그래프는 [15]의 투영 절차를 통해 우측 방향으로 투영할 수 있다.

그림 7은 $GF(2^9)$상의 $A^{(i-1)}x^D \bmod G$를 위한 일차원 신호 흐름 그래프(Signal Flow Graph: SFG) 어레이이다. 여기서 $D=3$이고 '●'은 1-비트 1 클럭 지연소자를 나타낸다. 그림 7에서 나타나듯이 $GF(2^9)$상의 곱셈을 위한 일차원 SFG는 동일한 $(N-1)$개의 처리기(Processing Element: PE)로 구성된다.

그림 8은 그림 7의 PE 회로를 나타내며 N길이의 컨트롤 시퀀스(011…1)에 의해 제어된다. 디지트 A_i와 G_i는 MSD 우선 시리얼 형태로 어레이에 입력된다. 그림 3에 나타나듯이 $A_{N-1}^{(i-1)}$의 계수는 그림 5 자료의존 그래프의 i번째 행의 모든 기본 셀들에

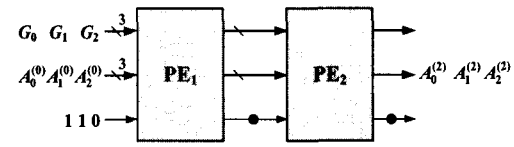

그림 7. 그림 3의 일차원 SFG 어레이

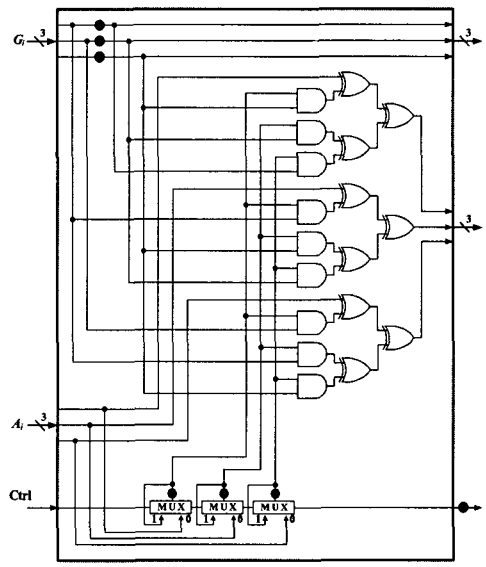

그림 8. 그림 7의 PE 회로도

브로드캐스팅 되어야 하기 때문에 추가적인 D개의 멀티플렉서와 D개의 1-비트 래치를 그림 7 SFG 어레이의 각 PE에 추가한다. 컨트롤 신호가 0이면 임시 결과는 래치된다.

유사한 방식으로 그림 5를 오른쪽으로 투영하면 그림 9와 같은 디지트-레벨의 일차원 SFG를 얻을 수 있다. 그림 9에서 PE-I은 그림 10의 회로도를, PE-II는 그림 8의 회로도를 각각 나타낸다. 그림 9에 나타나듯이 각 PE는 $(N+1)$길이의 컨트롤 시퀀스(011…1)에 의해 제어된다. 또한 그림 5의 자료의존 그래프에 나타나듯이 맨 왼쪽 셀의 0 입력 데이터 처리를 위해 각 PE-I에 D개의 2-입력 AND 게이트를 추가한다.

그림 7과 그림 9의 SFG를 결합한 후, 컷 셋 시스톨릭화 기법(cut-set systoli- zation technique)[15]을 적용하면 그림 11과 같은 $GF(2^m)$상의 완전한 디지트 시리얼 시스톨릭 곱셈기를 얻을 수 있다. 그림 11은 연속적인 입력 데이터에 대해 초기 $\lceil 3N \rceil +2$ 클럭 지연시간 후 매번 N 클럭 사이클마다 곱셈 결과를 출력한다. 즉, 곱셈 결과는 MSD 우선 디지트 형태로 곱셈기의 우측으로 출력된다.

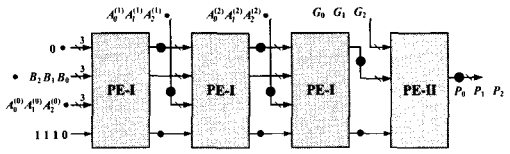

그림 9. 그림 5의 일차원 SFG 어레이

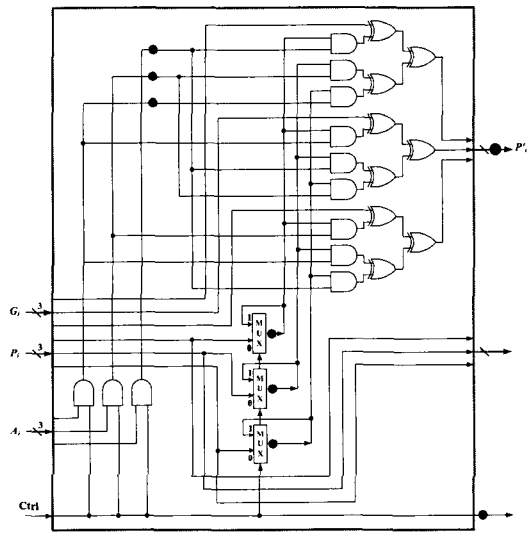

그림 10. 그림 9의 PE-I 회로도

V. 성능 분석

그림 11 곱셈기의 기능 검증을 위해 VHDL로 회로를 기술하였고 Mentor Graphics사의 VHDL-ChipSim을 이용하여 시뮬레이션 하였다. 시뮬레이션을 위한 회로 합성은 Synopsis사의 FPGA-Express(버전 2000, 11-FE3.5)에서 이루어졌으며, 타겟 FPGA 디바이스로 Altera사의 EP2A70F1508C-7을 사용하였다. 그림 11 곱셈기의 기능 검증 후, 동일한 입출력 형태를 가지는 기존의 곱셈기와 성능을 비교하였으며 표 1에 그 결과를 요약하였다. 표 1에서 3-입력 XOR 게이트와 4-입력 XOR 게이트는 두 개와 세 개의 2-입력 XOR 게이트로 구성된다고 가정하였다. 표 1에 나타나듯이 기존에 제안된 구조들은 선형의존성 때문에 디지트 크기 D가 증가하면 최대 처리기 지연시간 역시 선형으로 증가하지만 제안된 곱셈기는 이진트리 형태의 내부 구조를 가지기 때문에 D에 대해 로그단위로 증가한다. 따라서 제안된 구조는 D의 크기가 커짐에 따라 기존에 제안된 디지트 시리얼 시스톨릭 곱셈기에 비해 속도 측면에서 상당한 성능 향상을 보인다. 참고로 그림 11의 *PE-I*과 *PE-II*에 *2-to-1* 멀티플렉서를 추가하였지만 멀티플렉서의 모든 입력은 래치로부터 전달되고 모든 멀티플렉서의 출력은 래치로 전달되기 때문에 Critical Path에는 아무런 영향을 미치지 않는다.

표 1. $GF(2^m)$상의 디지트 시리얼 시스톨릭 곱셈기의 성능 비교

	Guo 등[4]	김 등[5]	그림 11
Throughput (1/cycles)	$1/N$	$1/N$	$1/N$
Latency (cycles)	$3N$	$3N$	$3N+2$
Circuit Requirement	AND_2 : $N(2D^2+D)$ XOR_2 : $2ND^2$ Latch : $10ND$ MUX_2 : $2ND$	AND_2 : $N(2D^2+D)$ XOR_2 : $2ND^2$ Latch : $10ND+4D$ MUX_2 : $2ND$	AND_2 : $N(2D^2+D)$ XOR_2 : $2ND^2$ Latch : $10ND+2D$ MUX_2 : $2ND$
Critical Path	$T_{AND2}+3T_{XOR2}+$ $(D-1)(T_{AND2}+$ $2T_{XOR2}+T_{MUX2})$	$T_{AND2}+T_{XOR2}+$ $(D-1)(T_{AND2}+$ $T_{XOR2}+T_{MUX2})$	$T_{AND2}+$ $\lceil \log_2(D+1) \rceil T_{XOR2}$
Control Signal	1	1	1

$N = \lceil m/D \rceil$
AND_2 : 2-input AND gate
XOR_2 : 2-input XOR gate
MUX_2 : 2-to-1 multiplexer
T_{AND2} : propagation delay through one AND_2 gate
T_{XOR2} : propagation delay through one XOR_2 gate
T_{MUX2} : propagation delay through one MUX_2 gate

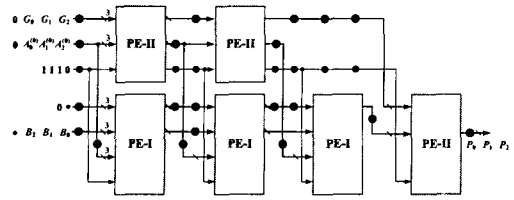

그림 11. 새로운 디지트 시리얼 시스톨릭 곱셈기

VI. 결 론

본 논문에서는 $GF(2^m)$상의 새로운 디지트 시리얼 시스톨릭 곱셈기를 제안하였다. 이를 위해 LSD 우선 곱셈 알고리즘으로부터 디지트-레벨의 자료의존 그래프를 얻은 후 투영 절차에 따라 일차원 SFG 어레이 및 PE를 유도하였고 컷 셋 시스톨릭화 기법을 적용하여 $GF(2^m)$상의 완전한 디지트 시리얼 시스톨릭 곱셈기를 구성하였다.

제안된 곱셈기는 크게 다음과 같은 두 가지 특성을 가진다. 1) 기존에 제안된 구조에 비해 거의 동일한 하드웨어를 사용하지만 훨씬 적은 계산지연을 가진다. 2) 제안된 구조가 타원곡선 암호 시스템과 같은 암호 응용에 적용된다면 다양한 디지트 크기를 선택할 수 있다. 따라서 위의 두 가지 특징으로부터 본 논문에서 제안된 $GF(2^m)$상의 곱셈기는 적절한 디지트 크기의 선택에 따라 최소한의 하드웨어 사용으로 최대한의 처리율을 만족시킬 수 있다. 뿐만 아니라 제안된 곱셈기는 높은 규칙성, 모듈성, 단방향 신호 흐름을 가지기 때문에 VLSI 구현에 매우 적합하다.

참 고 문 헌

[1] R. E. Blahut, Theory and Practice of Error Control Codes, Reading, MA: Addison Wesley, 1983.

[2] I. F. Blake, G. Seroussi, and N. P. Smart, Elliptic Curves in Cryptography, Cambridge University Press, 1999.

[3] S. K. Jain, L. Song, and K. K. Parhi, "Efficient Semisystolic Architectures for Finite-Field Arithmetic," IEEE Trans. VLSI Syst., Vol.6, No.1, pp.101-113, Mar. 1998.

[4] T. Zhang and K. K. Parhi, "Systematic Design Approach of Mastrovito Multipliers over $GF(2^m)$," Proc. of the 2000 IEEE Workshop on Signal Processing Systems (SiPS): Design and Implementation, Lafayette, LA, pp.507-506, Oct. 2000.

[5] C. S. Yeh, I. S. Reed, and T. K. Trung, "Systolic Multipliers for Finite Fields $GF(2^m)$," IEEE Trans. Comput., Vol.C-33, No.4, pp.357-360, Mar. 1984.

[6] C. L. Wang and J. L. Lin, "Systolic Array Implementation of Multipliers for Finite Field $GF(2^m)$," IEEE Trans. Circuits and Syst., Vol.38, No.7, pp.796-800, July 1991.

[7] G. Orlando and C. Paar, "A Super-Serial Galois Fields Multiplier for FPGAs and its Application to Public-Key Algorithms," Proc. of the 7th Annual IEEE Symposium on Field Programmable Computing Machines, FCCM'99, Napa Valley, California, pp.232-239, April. 1999.

[8] M. A. Hasan and V. K. Bhargava, "Bit-Serial Systolic Divider and Multiplier for Finite Fields $GF(2^m)$," IEEE Trans. Comput., Vol.41, No.8, pp.972-980, Aug. 1992.

[9] W. C. Tsai and S. J. Wang, "Two Systolic Architectures for Multiplication in $GF(2^m)$," IEE Proc. Comput. Digit. Tech., Vol.147, No.6, pp.375-382, Nov. 2000.

[10] C. Paar, P. Fleischmann, and P. Soria-Rodriguez, "Fast Arithmetic for Public-Key Algorithms in Galois Fields with Composite Exponents", IEEE Tans. Comput., Vol.48, No.10, pp.1025-1034, Oct. 1999.

[11] L. Song and K. K. Parhi, "Low Energy Digit-Serial/Parallel Finite Field Multipliers," J. VLSI Signal Processing, Vol.19, No.2, pp.149-166, June 1998.

[12] J. H. Guo and C. L. Wang, "Digit-Serial Systolic Multiplier for Finite Field $GF(2^m)$," IEE Proc. Comput. Digit. Tech., Vol.145, No.2, pp.143-148, Mar. 1998.

[13] C.H. Kim, S.D. Han and C.P. Hong, "An Efficient Digit-Serial Systolic Multiplier for Finite Fields $GF(2^m)$", Proc. on 14th Annual IEEE International Conference of ASIC/SOC, pp.361-365, 2001.

〔14〕 M.C. Mekhallalati, A.S. Ashur, and M.K. Ibrahim, "Novel Radix Finite Field Multiplier for $GF(2^m)$", J. VLSI Signal Processing, Vol.15, No.3, pp.233-245, Mar. 1998.
〔15〕 S. Y. Kung, VLSI Array Processors, Englewood Cliffs, NJ: Prentice Hall, 1988.
〔16〕 NIST, Recommended elliptic curves for federal government use, May 1999. http://csrc.nist.gov

김 창 훈 (Chang Hoon Kim) 정회원

2001년 2월 대구대학교 컴퓨터정보공학부 (학사)
2003년 2월 대구대학교 컴퓨터정보공학과 (석사)
2006년 8월 대구대학교 컴퓨터정보공학과 (박사)
2006년 9월 대구대학교 정보통신공학부 BK21, 연구교수
2007년 9월~현재 대구대학교 컴퓨터IT공학부, 전임강사
<관심분야> 암호 시스템, Embedded System, RFID/USN 보안

남 인 길 (In Gil Nam) 정회원

1978년 경북대학교 전자공학과 (공학사)
1981년 영남대학교 대학원 전자공학과(공학석사)
1992년 경북대학교 전자공학과 전산공학전공(공학박사)
1978년~1980년 대구은행 전산부
1996년~1997년 루이지애나 주립대학 교환교수
1980년~1990년 경북산업대학교 전자계산학과 부교수
1990년~현재 대구대학교 정보통신대학 컴퓨터·IT공학부 교수
<관심분야> 데이터베이스, GIS 등