GF(2^m) 상에서 디지트 시리얼 AB^2 시스톨릭 구조 설계

김남연° 유기영
경북대학교 컴퓨터공학과 정보보호연구실
knyeun@hanmail.net°, yook@knu.ac.kr

Digit-serial AB^2 Systolic Architecture in GF(2^m)

NamYeun Kim° KeeYoung Yoo
Dept. of Computer Engineering, Kyungpook National University

요 약

본 논문에서는 유한 필드 GF(2^m) 상에서 AB^2 연산을 수행하는 디지트 시리얼(digit-serial) 시스톨릭 구조를 제안하였다. 제안한 구조는 디지트 크기를 적당히 선택했을 때, 비트-패러럴(bit-parallel) 구조에 비해 적은 하드웨어를 사용하고 비트-시리얼(bit-serial) 구조에 비해 빠르다. 또한, 제안한 디지트 시리얼 구조에 파이프라인 기법을 적용하면 그렇지 않은 구조에 비해 m=160, L=2 일 때 공간-시간 복잡도가 10.9% 적다.

1. 서 론

최근 유한 필드상의 연산은 에러-교정 코드[1], 암호학[2, 3, 4], digital signal processing [5]등의 분야에서 주목을 받고 있다. 그리고 공개키 암호 시스템[3, 5, 6]에서의 기본 연산인 GF(2^m)상에서의 많은 구조들은 기저들을 달리하여 개발되어져 왔는데, 본 논문에서는 기저 변환이 필요없는 표준 기저 구조를 사용한다.

AB^2 연산은 GF(2^m)상에서 공개키 암호 시스템을 위한 효율적인 연산이다. 예를들면, 고속 회로를 디자인할 때 AB^2 연산은 곱셈과 곱셈의 역원($A/B = AB^{-1}$)을 이용한 나눗셈 연산을 수행하는데 효과적으로 쓰인다. 이 때 역원 연산은 $B^{-1} = B^{2^m-1} = (B(B(B \cdot B(B(B)^2)^2 \cdot)^2)^2)^2$과 같이 지수의 계산을 통해 얻어질 수 있다.

GF(2^m)상에서 파워썸(AB^2+C) 연산을 수행하기 위한 표준 기저 시스톨릭 어레이에 대한 연구가 이미 이루어져 왔다[6, 7]. 그러나 이러한 시스톨릭 파워썸 구조들은 하드웨어 복잡도가 높고 지연 시간이 길어 암호 시스템의 응용에는 적합하지 못하다. 그러므로 효율적인 연산에 대한 연구가 필요하다.

본 논문에서는 표준 기저를 사용한 GF(2^m)상에서의 AB^2 연산에 대한 새로운 알고리즘을 제안하고, 이 알고리즘을 통해 디지트 시리얼 시스톨릭 구조를 유도한다. 제안한 알고리즘은 병렬성을 제공하기 위해 MSB-first 구조를 사용하였고 하드웨어 복잡도와 지연 시간은 전통적인 구조들에 비해 효율적이다. 덧붙여 이 구조는 VLSI 구현에 적합하고 역원 구조에 쉽게 적용이 가능하다.

2. 알고리즘

유한필드 GF(2^m)은 2^m 개의 원소를 가진다. 본 논문에서는 GF(2^m)상의 모든 (2^m-1)개의 0이 아닌 원소들을 표준 기저 방식으로 표현한다. GF(2^m)상의 두 원소 A, B가 있다고 가정하고 이를 다항식 x로 나타내면 $A(x)=\sum_{i=0}^{m-1} a_i x^i$, $B(x)=\sum_{i=0}^{m-1} b_i x^i$ 이 때 a_i , $b_i \in$ GF(2) ($0 \le i \le m-1$). GF(2^m)상의 유한 필드 원소들은 GF(2)상의 m 차수의 원시 다항식으로 표현된다. $F(x)$를 기약 다항식으로 표현하면 다음과 같다.

$F(x) = x^m + \sum_{i=0}^{m-1} f_i x^i$ 이 때, $f_i \in$ GF(2) $(0 \le i \le m-1)$.

2.1 MSB-first AB^2 알고리즘

$A(x)B^2(x) \bmod F(x)$연산을 하기 위하여 다음의 식을 따른다.

$$
\begin{aligned}
R(x) &= A(x)B^2(x) \bmod F(x) \\
&= \sum_{j=0}^{m-1} A(x) b_j x^{2j} = \sum_{i=0}^{m-1} r_i x^i \\
&= \{(\cdots(\cdots((A(x)b_{m-1})x^2 + A(x)b_{m-2})x^2 + \cdots + A(x)b_{m-}.)x^2 + \\
&\quad \cdots + Ab_1)x^2 + Ab_0\} \bmod F(x)
\end{aligned}
$$

위 순환식에서 $A(x)b_{m-1}x^2 \bmod F(x)$을 첫번째 항으로 하여 마지막 항까지 계산을 하는데,

이 때, x^2 연산을 한 단계 앞에서 함으로써 마지막 항은 모듈러 리덕션 과정 없이 Ab_0 연산만을 하게 된다.
본 논문에서는 [8]에서 제안된 비트 레벨 MSB-first AB^2 알고리즘을 사용한다.

일반적인 항의 경우($1 \le i \le m-1$)

$$\begin{cases} d_k^i = p_k^{i-1} + a_k b_{m-i}; \\ p_k^i = d_{m-1}^i f_k' + d_{m-2}^i f_k \quad , for\ k = 0,\ 1; \\ p_k^i = d_{m-1}^i f_k' + d_{m-2}^i f_k + d_{k-2}^i \quad , for\ k = 2, 9, m-1; \end{cases}$$

마지막 항의 경우 ($i = m$)

$$\begin{cases} d_k^m = p_k^{m-1} + a_k b_0; \\ p_t^m = d_t^m = R(x); \end{cases}$$

2.2 자료의존 그래프

위 알고리즘의 수행을 2차원 평면에 표현한 그래프는 그림 1과 같다 [8]. 이 때, m=40이고, 각 인덱스 점 (index point) (i, k)은 i = 1, 2, ···, m and k = m-1, m-2, ···, 1, 0 와 같다. 또한, 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.

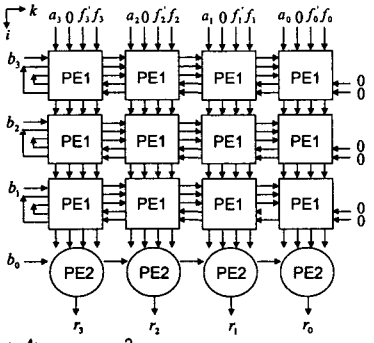

그림 1 GF(2^4)상의 AB^2 연산을 위한 자료 의존 그래프

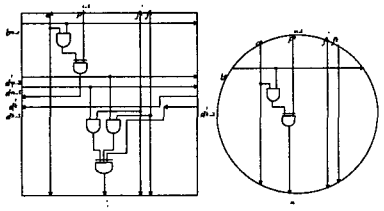

그림 2 (a) PE1 (b) PE2

3. 디지트 시리얼 시스톨릭 곱셈기

디지트 크기를 L 이라고 하자. 본 논문에서는 디지트 크기가 L 인 디지트 시리얼 시스톨릭 구조를 만들기 위해, 일반적으로 계산점을 $L \times L$ 로 묶는 방법을 사용한다. 이러한 방법을 그림 1 의 자료의존 그래프에 적용할 경우 수평 방향으로 양방향 데이터 흐름이 있기 때문에, 오른쪽으로 투영시킬 수 없다. 이러한 문제를 해결하기 위해 먼저, 마지막 행의 셀들을 제외한 나머지 셀들의 연산을 d_k^i 의 계산부분과 p_k^i 계산부분으로 분리하였다.
다음은 그림1의 자료의존 그래프 인덱스를 변환한 후, 인덱스 변환된 자료의존 그래프를 디지트 크기로 분리하였다.
그림1의 자료의존 그래프에서 수평 방향으로의 양방향 데이터 흐름을 피하기 위해 셀의 분할 후 d_k^i 을 계산하는 셀들은 셀 인덱스 (i, k)를 $(i, -2i+k+2)$으로 , 그리고 p_k^i 을 계산하는 셀들은 (i, k)를 $(i, -2i+k)$으로 인덱스 변환을 한다.
인덱스 변환된 자료의존 그래프는 그림1의 자료의존 그래프와 동일한 기능을 수행한다. 그림 1을 분할하여 인덱스 변환을 시키면 그림 3과 같다.

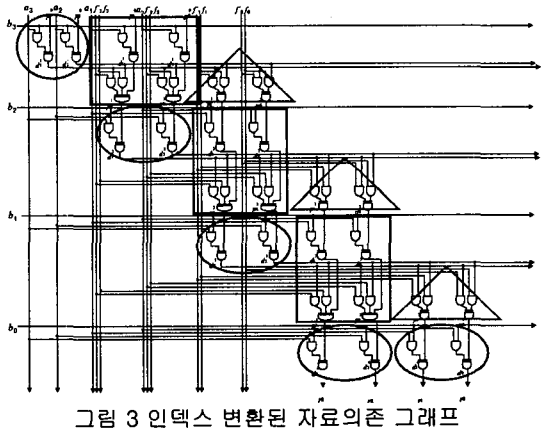

그림 3 인덱스 변환된 자료의존 그래프

그림 3은 $(m^2+2m-2)/2$개의 셀들로 구성되는데, 그것은 $(3m-2)/2$개의 PE_A cells, $(m^2-3m+2)/2$개의 PE_B 셀들 그리고 m-1개의 PE_C 셀들로 구성된 것이며 그림에서 PE_A, PE_B and PE_C 셀들은 각각 원, 사각형, 삼각형으로 표현된다.
디지트 크기가 L인 GF(2^m) 상의 디지트 시리얼 시스톨릭 곱셈기를 설계하기 위해, 먼저 그림 3을 수평방향으로 m/L 부분으로 분할한다. 이 때, 각 부분은 L 개의 열과 $(m+2L)$개의 행으로 구성된다. 그러나, 마지막 블록은 L 개의 열과 $(m+2(L-1))$개의 열로 구성된다. 다음으로 수직방향으로 $\lceil (m+2L)/L \rceil$개의 영역으로 분할을 한다.
[9]의 투영 절차에 따라 오른쪽으로 그림 3의 자료의존 그래프를 투영시키고 컷-셋 시스톨릭화 기법(cut-set systolization techniques)[10]을 이용하여, 그림 4와 같은 디지트 시리얼 시스톨릭 어레이를 유도할 수 있다.

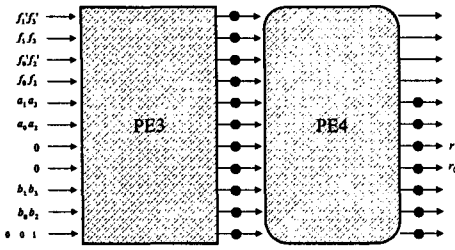

그림 4 GF(2^4) 상에서 L=2인 AB^2 연산을 위한 디지트-시리얼 시스톨릭 구조

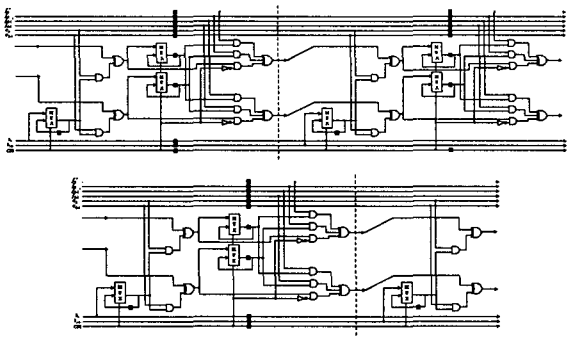

그림 5 그림 4의 PE3, PE4 회로

그림 4의 디지트 시리얼 시스톨릭 구조에 대해, 최대 임계경로는 $T_{max} = L\,(T_{AND2}+T_{NOT}+T_{XOR2}+T_{XOR3}+T_{MUX})$인데, 이때, T_{ANDi}, T_{XORi}, T_{NOT}, T_{MUX}는 각각 i-입력 AND 게이트, i-입력 XOR 게이트, NOT 게이트 그리고 2-to-1 멀티플렉서를 각각 나타낸다.

그러나, 디지트 크기 L이 커지면, 임계경로도 길어진다는 단점이 있으므로, 이를 극복하기 위해 각 셀에 파이프라인 기법을 적용하였다. [11]의 컷 이론을 적용하여, 그림 5의 점선 라인에 각각 $5L$+1개의 한 비트 래치를 추가하여 쉽게 파이프라인 시켰다. 그 결과, 지연시간은 $(5m-4)/2$가 되었고, 최대 지연시간은 $T'_{max} = T_{AND2}+T_{NOT}+T_{XOR2}+T_{XOR3}+T_{MUX}$ 이므로 파이프라인된 디지시리얼 시스톨릭 구조는 그렇지 않은 구조에 비해 Area-Time product 복잡도가 m=1600이고 L=8인 경우, 10.9% 낮다.

5. 결론

본 논문에서는 디지트 시리얼 입/출력 시스톨릭 AB^2 구조를 제안하였다. 관련있는 구조들과의 비교에 있어서 제안된 구조는 Area-Time product 복잡도에서 효율적이며 디지트 크기를 적당히 선택했을 때, 제안된 구조는 특정의 응용에 있어서, 최소의 하드웨어로 기존 구조에 비해 효율성을 보였다.

표1 시스톨릭 AB^2 구조들의 비교

Circuit Item	Wang *et al* [11]	DSPM	Pipelined-DSPM
Architecture	Systolic	Systolic	Systolic
I/O format	Bit-parallel	Digit-serial	Digit-serial
Number of cells	$m^2/2$	$\lceil m/L \rceil$	$\lceil m/L \rceil$
Function	AB^2+C	AB^2	AB^2
Throughput	1	L/m	L/m
Maximum cell delay	$T_{AND2}+3T_{XOR}$	$L\,(T_{AND2}+T_{NOT}+T_{XOR2}+T_{XOR3}+T_{MUX})$	$T_{AND2}+T_{NOT}+T_{XOR2}+T_{XOR3}+T_{MUX}$
Latency	$2m+m/2$	$\lceil (m+2(L-1))/L \rceil +3(m/L-1)$	$(5m-4)/2$
Circuit Complexity			
AND gates	$3m^2$	$\lceil m/L \rceil \cdot 4L^2-3$	$\lceil m/L \rceil \cdot 4L^2-3$
XOR gates	$3m^2$	$\lceil m/L \rceil \cdot 3L^2-2$	$\lceil m/L \rceil \cdot 3L^2-2$
Latches	$8.5m^2$	$\lceil m/L \rceil (3L^2+5L)-2$	$\lceil m/L \rceil (7L^2+5L-3)-4L-2$
Mux		$\lceil m/L \rceil \cdot 3L-2$	$\lceil m/L \rceil \cdot 3L-2$
NOT gates		$\lceil m/L \rceil L^2-L$	$\lceil m/L \rceil L^2-L$
No. of CS	-	1	1

참고문헌

[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] R.L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-key Cryptosystems," *Comm. ACM*. Vol. 21, pp. 120-126, 1978.

[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*)," *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: 226-229, 1994.

[8] N.Y.Kim, H.S.Kim and K.Y.Yoo, " Computation *AB2* multiplication in GF(2m) using low-complexity systolic architecture," IEE Proc. □ Circuits Devices Syst., Vol. 150, No. 2, April 2003.

[9] S.Y.Kung, *VLSI array processors*, Prentice Hall, Englewood Cliffs, NJ, 1988.

[10] Kung, H.T., and LAM, M., ' Fault tolerant and two level pipelining in VLSI systolic arrays,' Proceedings of MIT conference on *Advanced res. VLSI*, Cambridge, MA, January 1984, pp.74-83.

[11] C.L.Wang and J.H.Guo, ' New systolic arrays for *C+AB2*, inversion, and division in F(2m),' *IEEE Trans. Computers*, 29, pp. 1120-1125, 2000.