셀룰라 오토마타를 이용한 GF(2^m)상의 곱셈기

(Modular Multiplier based on Cellular Automata Over GF(2^m))

이 형 목 † 김 현 성 †† 전 준 철 ††† 유 기 영 ††††

(Hyoung-Mok Lee) (Hyun-Sung Kim) (Jun-Cheol Jeon) (Kee-Young Yoo)

요 약 본 논문에서는 유한 체 GF(2^m)상에서 셀룰라 오토마타 (Cellular Automata)의 구조에 적합한 곱셈기 구조를 제안한다. 제안된 LSB 우선 곱셈 구조는 AOP(All One Polynomial)를 기약 다항식으로 사용하며, m+1의 지연시간과 1-D_{AND}+1-D_{XOR}의 임계경로를 갖는다. 특히 정규성, 모듈성, 병렬성을 가지기 때문에 VLSI구현에 효율적이고 나눗셈기, 지수기 및 역원기를 설계하는 데 기본 구조로 사용될 수 있다. 또한, 이 구조는 유한 체 상에서 Diffie-Hellman 키 교환 프로토콜, 디지털 서명 알고리즘, 및 ElGamal 암호화와 같이 잘 알려진 공개키 정보 보호 서비스를 위한 기본 구조로 사용될 수 있다.

키워드 : 유한체, 공개키 암호화, 셀룰라 오토마타, 곱셈기, 기약다항식

Abstract In this paper, we propose a suitable multiplication architecture for cellular automata in a finite field GF(2^m). Proposed least significant bit first multiplier is based on irreducible all one polynomial, and has a latency of (m+1) and a critical path of (1-D_{AND}+1-D_{XOR}). Specially it is efficient for implementing VLSI architecture and has potential for use as a basic architecture for division, exponentiation and inverses since it is a parallel structure with regularity and modularity. Moreover our architecture can be used as a basic architecture for well-known public-key information service in GF(2^m) such as Diffie-Hellman key exchange protocol, Digital Signature Algorithm and ElGamal cryptosystem.

Key words : Finite field, Public-key cryptography, Cellular automata, multiplication architecture, Irreducible polynomial

1. 서 론

암호학[1,2]의 응용에서 갈로아 체(Galois Field, GF)[3,4]의 연산은 아주 중요하다. 이러한 연산은 효율적이고 적은 복잡도의 암호화 시스템의 구현에 필요하다. 특히 유한 체 중에서GF(2^m)은 암호학에서 가장 관심을 가지는 유한 체이다. 또한GF(2^m)은 0과 1의 비트 스트링(bit string)으로 이루어진2m 개의 원소를 가지기 때문에 컴퓨터 구조의 구현을 위한 계산에 적당하다. 이러한 유한 체 GF(2^m) 상에서 기저의 표현 방법은 세 가지 즉, 표준 (Standard), 정규(Normal), 이원(Dual)기저가 있다. 본 논문에서는 기저의 변환이 필요 없는 표준기저(Standard Basis)에 초점을 맞추었다.

1984년에 Yeh, *et al.*[5]는 일반적인 GF(2^m)상에서 AB+C의 연산을 수행하여 병렬 시스 톨릭 어레이 구조를 구현하였다. 표준기저 상에서 구현한 세미시스톨릭 에레이 구조가 논문 [6]에 제시되었다. 또한 논문 [7]에서는 정규기저 상에서 곱셈과 곱셈 역원을 계산 하기 위한 구조를 제안하였다. 논문 [8]에서는 AB^2+C를 계산하기 위한 세시스톨릭 회로 가 제안되었다. 그리고 그 후에도 많은 비트 단위 병렬 세미시스톨릭 곱셈기들이 제안 되었으나 시스템의 복잡도 때문에 이러한 곱셈기들은 암호화 시스템구성에 효과적 이지 못했다. 이러한 시스템의 복잡도를 줄이기 위해서 Itoh와 Tsujii [9]가 GF(2^m)상에서 m차의 기약 다항식 AOP에 기초한 곱셈기와 m차의 기약 다항식 ESP에 기초한 곱셈기 를 설계하였다. 이렇게 설계된 두 개의 곱셈기는 작은 시스템 복잡도를 가졌다. 이 후에 Hasan, *et al.*[10]은 AOP에 기초한 작은 크기의 병렬 곱셈기를 제안하고, 이를 이용하여 ESP에 기초한 병렬 곱셈기로 발전시켰다.

· 본 연구는 대학 IT연구센터 육성지원사업의 연구 결과로 수행되었음

† 비 회 원 : (주) 모빌랩 연구원
hmlee@mobilab.co.kr
†† 종신회원 : 경일대학교 컴퓨터공학과 교수
hskim@kiu.ac.kr
††† 학생회원 : 경북대학교 컴퓨터공학과
jcjeon33@infosec.knu.ac.kr
†††† 종신회원 : 경북대학교 컴퓨터공학과 교수
yook@knu.ac.kr

논문접수 : 2001년 12월 1일
심사완료 : 2003년 9월 4일

1950년대에 John Von Neuman에 의해 처음으로 소개된 셀룰라 오토마타 (CA)는 복잡 하고 물리적인 시스템의 시뮬레이션을 위해 적당한 모델로서 채택 되었다 [11]. 또한, 논문 [12]에서 Choudhury는 CA를 이용한 LSB방식의 곱셈기를 제안하였다. 지금까지 연구에서 효율적인 구조들이 제안되었지만 보다 더 효율적인 구조의 설계에 관한 연구가 필요하다.

본 논문에서는 기약 다항식으로 AOP의 속성을 사용한 새로운 모듈러 곱셈기 구조를 제안한다. 곱셈을 LSB 방식으로 계산하는 제안된 구조의 기본 셀 구조는 하나의 AND와 하나의 XOR로 이루어진다. 제안된 구조는 전체 지연시간으로 m+1을 갖고 임계경로로는 1-D_{AND}+1-D_{XOR}를 갖는다. 또한 제안된 구조는 시간과 공간 복잡도 면에서 기존의 구조보다 효율적이다. 제안된 구조는 보다 효율적인 지수기, 나눗셈기 및 역원기를 위한 기본 구조로 사용될 수 있다. 또한 암호화 시스템의 보안을 위해서 사용될 수 있다. 더욱이 구조 자체가 정규성, 모듈성, 병렬성을 가지기 때문에 VLSI구현에 효율적이다.

2. 셀룰라 오토마타

셀룰라 오토마타(Cellular Automata, CA)는 규칙적으로 상호 연결된 많은 셀들로 구성 되어져 있는 유한 상태 머신 (Finite State Machine)이다. CA의 각 셀들은 상호 연결된 이웃 의 현재 상태 값과 특별한 법칙에 따라 이산적 시간에 동시에 새로운 상태 값으로 갱신 되어진다. CA는 이웃 셀을 이용하여 셀을 갱신하기 위한 함수 즉 법칙과 자신을 포함 하여 셀을 갱신 하는데 직접적으로 관여하는 이웃의 개수에 의해 이루어진다.

예제. 2-상태 3-이웃 1-차원 CA.

이웃의 상태 :	111	110	101	100	011	010	001	000	
상태 계수 :	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
다음 상태 :	0	1	0	1	1	0	1	0	(rule 90)
다음 상태 :	1	1	1	1	0	0	0	0	(rule 240)

위의 진리 테이블에서 첫 번째 행은 시간 t에서 3-이웃으로 셀을 나타낼 수 있는 모든 상태를 보여준다. 두 번째 행은 이와 관련된 상태 계수이다. 반면에 세 번째 행과 마지막 행은 두 개의 다른 법칙 즉, 법칙 90과 법칙 240에 대한 시간 t+1에서 i 번째 셀에 대응되는 상태이고 이들 계수의 십진 표현이 법칙의 숫자가 된다. 세 번째 행은 왼쪽 이웃과 오른쪽 이웃을 XOR한 결과로서 자신의 상태를 갱신한다. 이의 십진 표현은 90이고, 이것을 법칙 90이라 한다. 마지막 행의 십진 표현은 240, 즉 법칙 240이고 자신의 왼쪽 이웃 값에만 의존한다. 이처럼 CA는 현재의 상태와 법칙에 의해 다음 상태가 결정되어진다. 시간 t에서 이전, 현재 그리고 다음 상태를 s_{i-1}, s_i, 그리고 s_{i+1}라 하자. 그러면 두 법칙 90과 240은 다음의 식과 같이 표현할 수 있다.

rule 90 : $s_i^+ = s_{i-1} \oplus s_{i+1}$

rule 240 : $s_i^+ = s_{i-1}$

여기서 s_i^+는 시간 t+1에서 현재 상태를 나타내고, '⊕'는 XOR 연산을 나타낸다.

또한 CA는 셀 사이에 적용된 법칙이 어떤 연산으로 이루어지는가에 따라서 선형 셀룰라 오토마타(Linear CA), 비선형 셀룰라 오토마타(Non-Linear CA), Additive CA로 이루어 진다. 선형 셀룰라 오토마타는 각 셀들간의 다음 상태를 갱신 하기 위한 법칙이 선형 연산인 XOR연산 만으로 이루어진 것을 말한다.

정의 1. 모든 상태 s와 u에 대해 $f(s+u)=f(s)+f(u)$이고 모든 상태 s와 유한 체 GF(P)의 모든 원소 α에 대해 $f(\alpha s)=\alpha f(s)$이면, 이 때 함수 f를 선형(linear)라 한다.

비선형 셀룰라 오토마타는 그 이외의 연산으로 이루어진 것을 말한다. 비선형 셀룰라 오토마타 중 XOR와 XNOR연산 만으로 이루어진 CA를 Additive CA라 한다. 그리고 CA는 적용된 법칙들이 동일한 것으로 이루어지는 Uniform CA와 두개 이상의 법칙들로 이루어진 Hybrid CA로 구분된다. 또한 각 셀들의 배열 구조에 따라서 1-차원, 2-차원, 3-차원 CA가 있다.

CA를 구성하는데 있어서 다른 중요한 요소 중의 하나는 경계 조건이다. CA는 가장 왼쪽 셀의 왼쪽 이웃과 가장 오른쪽 셀의 오른쪽 이웃이 존재하지 않는다. 이를 해결하기 위한 Null, Periodic, Intermediate 경계 조건이 존재한다. 이 경계 조건을 각각 적용한 CA를 NBCA(Null Boundary CA), PBCA(Periodic Boundary CA), IBCA(Intermediate Boundary CA)라 부른다. NBCA는 가장 왼쪽 셀과 가장 오른쪽 셀에서의 입력을 0으로 부여하는 것이고, PBCA는 가장 왼쪽 셀과 가장 오른쪽 셀이 이웃한 것으로 간주함으로써 두 입력을 부여하는 것이다. IBCA는 가장 왼쪽(오른쪽) 셀의 왼쪽(오른쪽) 이웃을 두 번째 오른쪽(왼쪽) 이웃으로 간주하여 값을 입력하는 것이다.

n개의 셀을 가지는 CA의 현재 상태는 n-백터, 즉 $v=(v_1,v_2,\cdots,v_n)$라 하고 특성화 행렬을 T라 하자. 이 때 시간 t에서 CA의 상태를 v^t라 하면, 시간 t+1에서 CA의 상태는 다음과 같이 표현된다.

$$v^{t+1} = Tv^t \quad (1)$$

결국 CA의 특성화 행렬의 i번째 행은 i번째 셀의 이웃에 의존도를 나타낸다. 법칙 240 즉, 자기 자신의 왼쪽 이웃을 1로 표현하고 나머지 이웃은 0으로 표현하는 PBCA의 (4×4)특성화 행렬은 다음과 같이 나타낼 수 있다.

$$\begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \quad (2)$$

그림 1은 법칙 <90, 150, 90, 150>을 가지는 4-셀 2-상태 3-이웃 1-차원 CA를 보여준다. 여기서 #i는 CA의 각 셀을 나타낸다.

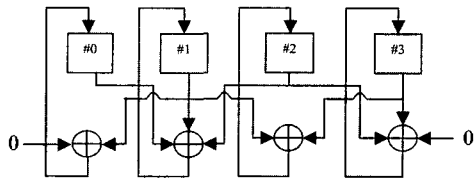


그림 1 법칙 <90, 150, 90, 150>을 가지는 4-셀 2-상태 3-이웃 1-차원 CA

3. 유한 체와 All One Polynomial(AOP)

유한 체 GF(2)의 유한 확대 체를 GF(2^m)이라 하자. 먼저 유한 확대 체 GF(2^m)의 원소를 표현하는 방법은 표준, 정규, 이원기저의 세 가지 기저가 있다. 첫 번째, 표준기저{1, α, α^2, ⋯, α^{m-1}}에 의해서 유한 체 GF(2^m)상의 임의의 원소 A를 나타내면 $A = a_{m-1}\alpha^{m-1} + a_{m-2}\alpha^{m-2} + \ldots + a_1\alpha + a_0$로 나타낼 수 있다. 두 번째, 정규기저 {$\alpha$, α^2, ⋯, α^{2m-2}}에 의해서 GF(2^m)상의 임의의 원소 A를 나타내면 $A = a_{m-1}\alpha^{2m-2} + a_{m-2}\alpha^{2m-4} + \ldots + a_1\alpha^2 + a_0$로 나타낼 수 있다.

마지막으로 이원기저 {$\mu_0, \mu_1, \ldots, \mu_{m-1}$}에 의해서 임의의 원소 A를 나타내면 $A = a_{m-1}\mu_{m-1} + a_{m-2}\mu_{m-2} + \ldots + a_1\mu_1 + a_0\mu_0$로 나타난다. 단, 각각의 a_i(i = 0, 1,⋯,m-1)는 유한 체 GF(2)상의 원소이다. 다항식 $f(x)$의 근을 α라 하자. 상에서 $f(x)$를 $f(x) = f_m f^m + f_{m-1} f^{m-1} + \ldots + f_1 x + f_0$라 할 때, 만약 f_i=1 (i = 0,1,⋯,m)이면 이 $f(x)$를 m차의 AOP(all one polynomial)이라고 한다. 이 다항식 AOP에서 m+1이 소수이고 2가 모듈러 m+1에 대해 원시 근이 되면 기약 다항식이 된다.

그리고 위의 AOP $f(x)$의 근 α에 의해 생성된 집합 {1, α, α^2, ⋯, α^{m-1}}은 유한 체 GF(2^m)의 표준 기저가 되고 유한 체 GF(2^m)상의 원소 A는 $A = a_{m-1}\alpha^{m-1} + a_{m-2}\alpha^{m-2} + \cdots + a_1\alpha + a_0$로 표현된다. 이 표준기저에서 확장된 기저를 {1, α, α^2, ⋯, α^{m-1}, α^m}이라 하면, 확장된 기저 상에서 유한 체 GF(2^m)상의 원소 A는 $A = A_m\alpha^m + A_{m-1}\alpha^{m-1} + \cdots + A_1\alpha + A_0$ ($A_i = A_m + a_i$, $0 \le i \le m$)로 표현된다. 그러므로 GF(2^m) 상의 원소 A는 두 가지 다른 표현을 가진다.

$F(x) = x^m + x^{m-1} + \ldots + x + 1$을 m차의 기약 다항식 AOP라 하고 $F(x)$의 근을 α라 하자. 즉, $F(\alpha) = \alpha^m + \alpha^{m-1} + \cdots + \alpha + 1 = 0$이다. 그러면 $\alpha^m = \alpha^{m-1} + \ldots + \alpha + 1$로 나타낼 수 있고 양변에 α를 곱하고 정리하면 다음 방정식을 만족한다.

$$\alpha^{m+1} = 1 \quad (3)$$

유한 체 GF(2^4)상의 두 원소 A와 B를 확장된 기저로 표현하고 확장된 체 상에서 AOP 속성이 적용된 P^* = α^5+1를 모듈러로 사용해서 A와 B의 곱 즉, AB mod P^*(LSB방식)을 다음과 같이 표현할 수 있다.

$AB \bmod P^*$

$$= A(b_0 + b_1\alpha + b_2\alpha^2 + b_3\alpha^3 + b_4\alpha^4) \bmod P^*$$

$$= (Ab_0 + Ab_1\alpha + Ab_2\alpha^2 + Ab_3\alpha^3 + Ab_4\alpha^4) \bmod P^*$$

이 곱셈 연산을 수행하는데 있어서 위에서 언급한 AOP의 속성이 모듈러로서 사용된다. 즉, 모듈러 연산은 원소 A의 각 비트들을 한 비트 순환 시프트(Circular shift) 함으로써 수행된다. 그림 2에서 보여주는 바와 같이 GF(2^4)상에서 법칙 240이 적용된 (5×5)특성화 행렬을 가지는 3-이웃 1-차원 PBCA구조를 이용하여 $A^{\alpha} \bmod P^*$ 연산을 수행할 수 있다.

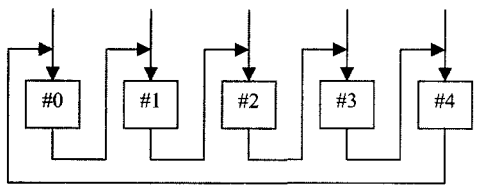


그림 2 법칙 240이 적용된 PBCA

4. 효율적인 셀룰라 오토마타 곱셈기 설계

이 장에서는 셀룰라 오토마타에 기반하고 기약 다항

식으로 AOP의 특성을 사용하는 곱셈기 구조를 제안한다. GF(2^m)상의 두 원소 A와 B를 확장된 기저상에서 표현해서 A와 B의 LSB방식의 곱셈 즉, $AB \bmod P$ ($P = \alpha^{m+1}$)을 다음과 같이 표현할 수 있다.

$$
\begin{aligned}
& AB \bmod P \\
&= A(b_0 + b_1\alpha + \ldots + b_{m-1}\alpha^{m-1} + b_m\alpha^m) \bmod P \\
&= (Ab_0 + Ab_1\alpha + \ldots + Ab_{m-1}\alpha^{m-1} + Ab_m\alpha^m) \bmod P \\
&= (r_0 + r_1\alpha + \ldots + r_{m-1}\alpha^{m-1} + r_m\alpha^m) \qquad (4)
\end{aligned}
$$

위 식은 크게 두 가지 연산으로 나뉘어진다. CA를 이용해서 승수의 자리 수, $A\alpha \bmod P$를 구하는 연산과 구해진 자리 수를 이용해서 실제 곱셈 연산인 $R_i = R_i + Ab_i$ ($0 \le i \le m$)를 구하는 연산이다.

먼저 CA를 이용한 $A\alpha \bmod P$의 연산은 레지스터에서 원소 A의 각 비트들을 한 비트 순환 시프트 함으로써 얻을 수 있다. 이 때 AOP의 이런 특성을 CA에 적용하면 왼쪽 이웃의 값에만 의존적인 법칙 240을 따라야 한다. 그리고 마지막 셀과 처음 셀이 이웃한 것으로 간주해서 입력을 처리하는 PBCA를 이용함으로써 $A\alpha \bmod P$를 계산할 수 있다. (5)식은 (4)식을 계산하기 위해 법칙 240을 적용한 PBCA의 $(m+1)\times(m+1)$특성화 행렬이다.

$$
\begin{pmatrix}
0 & 0 & 0 & ⓳ & 0 & 1 \\
1 & 0 & 0 & ⓳ & 0 & 0 \\
0 & 1 & 0 & ⓳ & 0 & 0 \\
⓱ & ⓱ & ⓱ & ⓱ & ⓱ & ⓱ \\
0 & 0 & 0 & ⓳ & 0 & 0 \\
0 & 0 & 0 & ⓳ & 1 & 0
\end{pmatrix} \qquad (5)
$$

그리고 이렇게 계산된 결과 값, 즉 $A\alpha \bmod P$와 b_i 값을 AND 연산하고 그 결과 값을 누적 변수 R_i에 누적하면 원하는 곱셈 결과를 얻을 수 있다.

그림 3의 (a)에서는 셀룰라 오토마타를 이용한 제안된 곱셈기 구조를 보여주고 각 셀들은 (b)구조를 기반으로 한다. 단, #i ($0 \le i \le m$)는 PBCA의 각 셀을 나타낸다.

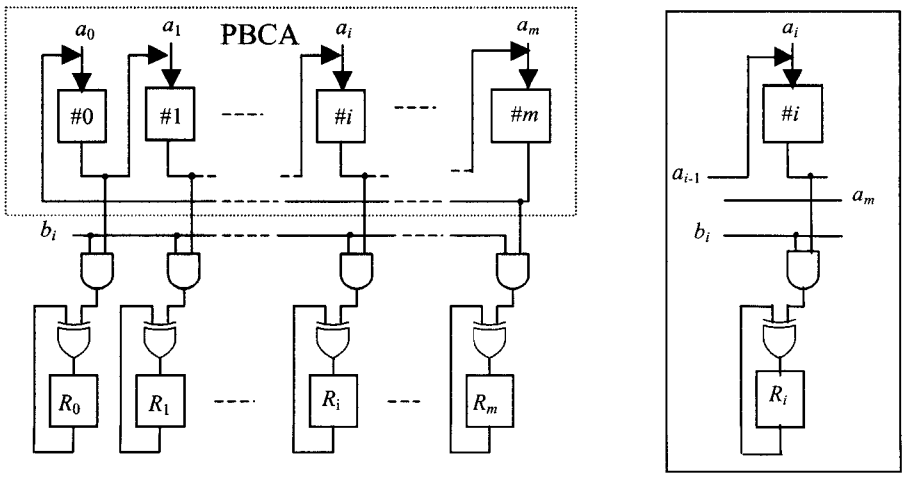
5. 비교 및 분석

기존의 모듈러 멱승 알고리즘은 대부분 선형 귀한 시프트 레지스트(Linear Feedback Shift Register: LFSR)을 이용하여 구현되었다. 본 장에서는 효과적인 비교를 위하여 Choudhury[12]와 Fenn[13]의 구조를 비교 대상으로 선정하였다. Choudhury의 구조는 $AB+C$ 연산을 위해 CA구조를 이용한 LSB우선 곱셈 방식의 비트 병렬 곱셈기이며, Fenn의 구조는 AB 연산을 행렬의 곱셈을 이용하여 LSFR방식으로 구현한 MSB우선 곱셈 방식의 비트 직렬 구조이다.

표 1 구조 비교

구조 / 항목	Choudhury [12]	Fenn [13]	제안된 구조
기 능	$AB+C$	AB	$AB+C$
셀 수	m	m	$m+1$
셀 복잡도	2 AND 2 XOR	1 AND 1 XOR	1 AND 1 XOR
지연 시간	m	$2m+1$	$m+1$
임계 경로	$2\text{-}D_{AND}+2\text{-}D_{XOR}$	$1\text{-}D_{AND}+1\text{-}D_{XOR}$	$1\text{-}D_{AND}+1\text{-}D_{XOR}$

위의 표에서 D_{AND}와 D_{XOR}를 각각 AND와 XOR 게



(a) 곱셈기 구조 (b) 곱셈기의 기본 셀

그림 3 GF(2^m)상의 PBCA를 이용한 곱셈기

이트의 지연시간이라 할 때, 논문 [12]에서 지연시간은 m이고 임계경로는 $2\text{-}D_{AND}+2\text{-}D_{XOR}$이다. 제안된 구조에서는 비록 연산 시간은 1증가하지만 $1\text{-}D_{AND}+1\text{-}D_{XOR}$의 임계경로를 가진다.

따라서 논문 [12]의 구조보다 임계경로 면에서 더 효율적이다. 논문 [13]에서는 지연 시간이 $2m+1$이고 임계경로는 제안된 구조와 같다. 제안된 구조는 논문 [13]과 비교할 때, 같은 임계경로를 가지지만 시간 면에서는 m 만큼 더 효율적이다. 결국 본 논문에서 제안된 구조가 기존의 구조인 논문 [12]와 [13]에 비해서 각각 임계경로와 지연시간 면에 서 더 효율적임을 알 수 있다. 제안된 구조는 $GF(2^m)$상에서 효율적인 나눗셈기, 지수기 및 역원기를 설계하는 데 기본 구조로 사용될 수 있을 것이다.

6. 결 론

본 논문에서는 유한 확대 체 $GF(2^m)$상의 셀룰라 오토마타를 이용한 곱셈기 구조를 제안하였다. 그리고 모듈러 연산을 위해 일반 기약 다항식을 사용하지 않고 AOP의 특성을 사용함으로써 $m+1$의 지연시간과 $1\text{-}D_{AND}+1\text{-}D_{XOR}$의 임계경로를 가졌다. 본 논문에서 제안된 구조는 기존의 구조보다 시간과 임계경로 면에서 더 효율적인 장점을 제공한다. 그래서 제안된 구조는 $GF(2^m)$상에서의 효율적인 나눗셈기, 지수기 및 역원 기를 설계하는데 기본 구조로 사용될 수 있다. 더욱이 제안된 구조 자체가 정규성, 모듈성, 병렬성을 가지기 때문에 VLSI구현에 효율적이다.

참 고 문 헌

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

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

[3] E. R. Berlekamp, *Algebraic Coding Theory*, New York: McGraw-Hill, 1986.

[4] R. J. McEliece, *Finite fields for Computer Scientists and Engineers*, New York: Kluwer-Academic, 1987.

[5] C. S. Yeh. S. Reed, and T. K. Truong, "Systolic multipliers for finite fields $GF(2^m)$," *IEEE Trans. on Computers.* Vol. 33, pp.357~360, Apr. 1984.

[6] S. K. Jain and L. Song, "Efficient Semisystolic Architectures for finite field Arithmetic," *IEEE Trans. on VLSI Systems*, Vol. 6, No. 1, Mar. 1998.

[7] J. L. Massey and J. K. Omura, *Computational method and apparatus for finite field arithmetic,* U. S. Patent application, submitted 1981.

[8] S. W. Wei, "A systolic power-sum circuit for $GF(2^m)$," *IEEE Trans. Comput.*, Vol. 43, pp.226~229, Feb. 1994.

[9] T. Itoh and S. Tsujii, "Structure of parallel multipliers for a class of finite fields $GF(2^m)$," *Info. Comp.* Vol. 83, pp.21~40, 1989.

[10] M. A. Hasan, M. Z. Wang and V. K. Bhargava, "Modular Construction of low complexity parallel multipliers for a class of finite fields $GF(2^m)$," *IEEE Trans. on Computers.* Vol.8. pp.962~971, Aug. 1992.

[11] J. V. Newmann, *The theory of self-reproducing automata*, Univ. of Illinois Press, Urbana (London, 1966.

[12] P. P. Choudhury, "Cellular Automata Based VLSI Architecture for Computing Multiplication And Inverses In $GF(2^m)$," *IEEE 7th International Conference on VLSI Design*, pp.279~282. Jan. 1994.

[13] S. T. J. Fenn et al, "Bit-serial Multiplication in $GF(2^m)$ using irreducible all one polynomials," *IEE. Proc. Comput. Digit. Tech.*, Vol. 144. No. 6. Nov. 1997.

이 형 목

2001년 계명대학교 이과대학 수학과(이학사). 2003년 경북대학교 공과대학 대학원 컴퓨터공학과(공학석사). 2003년~현재 (주) 모빌랩 연구원. 관심분야는 암호연산, 병렬처리, 암호화 프로토콜

김 현 성

1996년 경일대학교 공과대학 컴퓨터공학과(공학사). 1998년 경북대학교 공과대학 대학원 컴퓨터공학과(공학석사). 2002년 경북대학교 공과대학 대학원 컴퓨터공학과(공학박사). 2002년~현재 경일대학교 공과대학 컴퓨터공학과 교수. 관심분야는 암호연산, 병렬처리, 암호화 프로토콜

전 준 철

2000년 금오공과 대학교 공과대학 컴퓨터공학과(공학사). 2003년 경북대학교 공과대학 대학원 컴퓨터공학과(공학석사) 2003년~현재 경북대학교 공과대학 대학원 컴퓨터공학과(박사과정). 관심분야는 암호연산, 병렬처리, 접근제어

유 기 영

1976년 경북대학교 이과대학 수학교육과(이학사). 1978년 한국 과학 기술원 컴퓨터공학과(공학석사). 1993년 New York Rensselaer Polytechnic Institute 컴퓨터과학과(이학박사). 1978년~현재 경북대학교 공과대학 컴퓨터공학과 교수. 관심분야는 암호연산, 병렬처리, 암호화 프로토콜, 정보보호