프로그램 가능한 셀룰라 오토마타를 이용한 곱셈기 설계

박혜영° 전준철 유기영
경북대학교 정보보호학과° 컴퓨터공학과
{thewise° jcjeon33}@infosec.ac.kr
yook@knu.ac.kr

Design of Multiplier based on Programmable Cellular Automata

Hyeyoung Park° Juncheol Jeon Keeyoung Yoo
Dept. of Information Security, Kyungpook National University°
Dept. of Computer Engineering, Kyungpook National University

요 약

본 논문에서는 프로그램 가능한 셀룰라 오토마타(Programmable Cellular Automata, PCA)를 이용한 곱셈기를 제안한다. 본 논문에서 제안한 구조는 연산 후 늘어나는 원소의 수를 제한하기 위하여 이용되는 기약다항식(irreducible polynomial)으로서 All One Polynomial(AOP)을 사용하며, 주기적 경계 셀룰라 오토마타(Periodic Boundary Cellular Automata, PBCA)의 구조적인 특성을 사용함으로써 정규성을 높이고 하드웨어 복잡도와 시간 복잡도를 줄일 수 있는 장점을 가지고 있다. 제안된 곱셈기는 시간적, 공간적인 면에서 아주 간단히 구성되어 지수연산을 위한 하드웨어 설계나 오류 수정 코드(error correcting code)의 연산에 효율적으로 이용될 수 있을 것이다.

1. 서 론

최근 암호학에서 다양한 분야의 기법들이 유한체 GF(2^m)상에서 이루어지고 있다[1]. 공개키 암호화 시스템으로 암호와 디지털 서명을 위해 많은 제품과 표준에 사용되는 Diffie-Hellman과 Elgamal, 이에 경쟁하는 시스템인 ECC(Elliptic Curve Cryptosystem)에서는 유한체상의 나눗셈이나 지수연산 및 곱셈의 역원과 같은 연산들을 요구한다[2]. 이러한 연산들은 유한체상에서 모듈러 지수 연산을 기본으로 하고 지수연산은 AB 또는 AB^2 연산의 반복으로 구현할 수 있다.

곱셈 연산은 다른 연산과는 달리 연산 후 늘어나는 원소들의 수를 제한하기 위해 모듈로(modulo) 연산을 필요로 한다. 이를 기약다항식(irreducible polynomial)이라고 하는데 기약다항식으로는 항이 세 개인 Trinomials와 다섯 개인 Pentanomials, 그리고 모든 항의 계수가 1인 AOP가 많이 쓰인다.

많은 연구에서 CA의 구조에 기반한 효율적인 구조를 제안하고 있다. Von Neumann에 의해 소개된 셀룰라 오토마타는 수리적 이론에서의 많은 문제와 병렬처리 연산처럼 다양한 응용에서 사용되고 있다[3]. Zhang은 프로그램 가능한 셀룰라 오토마타를 이용한 곱셈기를 제안하였으며 Jeon은 1-AND+1-XOR의 셀 복잡도를 가진 MSB 우선 곱셈기를 제안하였다[4][5]. 본 논문에서는 PCA를 기반한 효율적인 곱셈기를 제안한다. 또한 모듈러로써 AOP를 적용함으로써 정규성이 뛰어난 구조를 유도한다.

본 논문은 다음과 같이 구성되어 있다. 2장에서 유한체에 대한 기본적인 개념과 특성을 살펴보고, 3장에서 셀룰라 오토마타와 프로그램 가능한 셀룰라 오토마타에 대해 알아본다. 제안된 곱셈기의 구조를 4장에서 설명하고 5장에서 기존의 구조와 비교, 분석한다. 마지막으로 6장에서는 결론을 맺는다.

2. 유 한 체

유한필드 혹은 갈로아 체(Galois Field, GF)로 불리는 유한체는 교환, 결합, 분배 법칙에 대해 닫혀 있고, 덧셈, 뺄셈, 곱셈, 나눗셈 연산이 가능한 유한개의 원소들의 집합이다. 유한체에서 원소들을 표기하기 위해서 정규기저(normal basis) 표기법, 이원기저(dual basis) 표기법 그리고 다항식기저(polynomial basis) 표기법등이 있다. 본 논문에서는 기저의 변환 단계가 필요 없는 다항식 기저 표기법으로 원소를 표시한다.

유한체 GF(2)의 유한 확대체인 GF(2^m)은 2^m개의 원소를 가진다[6]. GF(2)의 원소를 계수로 갖는 m차의 기약다항식을 $f(x)$라고 할 때, m+1이 소수이고 2가 모듈로 m+1의 원시근일 때, 다항식의 계수가 모두 '1'인 다항식 $f(x)=1+x+\ldots+x^{m-1}+x^m$을 AOP(All One Polynomial)이라고 한다. 이 방정식의 근을 α라 하면 $f(\alpha)=\alpha^{m+1}+1=0$의 속성을 가진다[6].

이때, 유한체 상의 한 원소 A는 $A=a_0+a_1\alpha+\ldots+a_{m-2}\alpha^{m-2}+a_{m-1}\alpha^{m-1}$ 이고, $a_i(0\le i\le m-1)$는 GF(2)의 원소이다. 또한 {1, α, ..., α^{m-2}, α^{m-1}, α^m}은 GF(2^m)상의 표준기저(standard basis)에서 확장된 기저이며, 원소 A는 다음과 같이 표현될 수 있다.

$$A=a_0+a_1\alpha+\ldots+a_{m-2}\alpha^{m-2}+a_{m-1}\alpha^{m-1}+a_m\alpha^m,\ (a_m=0) \quad (1)$$

본 논문에서는 AOP의 속성, $f(\alpha)=\alpha^{m+1}+1=0$을 효과적으로 이용하기 위하여 식(1)과 같이 하나 확장된 기저로 원소를 표현한다. 이러한 속성을 곱셈연산을 수행하

는데 있어 정규성을 제공하고 하드웨어 복잡도를 보다 효과적으로 감소한다.

3. 셀룰라 오토마타

본 장에서는 셀룰라 오토마타(Cellular Automata, CA)와 프로그램 가능한 셀룰라 오토마타(Programmable Cellular Automata, PCA)의 기본적인 특성과 법칙, 구조에 대하여 알아본다.

3.1 셀룰라 오토마타

규칙적으로 상호 연결된 많은 셀들로 구성되어져 있는 유한 상태 머신(Finite State Machine, FSM)에서 각각의 셀들은 적용된 법칙과 자신과 연결된 이웃의 현재 상태 값에 따라 새로운 값으로 갱신된다. CA를 구성하는 중요한 요소는 각 셀의 상태갱신에 적용되는 법칙과 여기에 직접적으로 관여하여 셀의 갱신에 직접적으로 영향을 미칠 수 있는 이웃 셀의 개수이다[7].

본 논문에서는 두 가지 상태를 가진 3-이웃 1차원 CA를 고려한다. 표 1은 3-이웃으로 가능한 모든 셀의 상태와 여러 가지 법칙을 보여준다.

표. 1 2-상태 3-이웃 1차원 CA

	111	110	101	100	011	010	001	000
90	0	1	0	1	1	0	1	0
150	1	0	0	1	0	1	1	0
240	1	1	1	1	0	0	0	0

법칙 90 : 왼쪽 이웃⊕오른쪽 이웃 ⇒ 상태갱신
법칙 150 : 왼쪽 이웃⊕자신⊕오른쪽 이웃 ⇒ 상태갱신
법칙 240 : 왼쪽 이웃 ⇒ 상태갱신

CA는 적용된 법칙에 따라서 선형 CA(Linear CA)와 비선형 CA(Nonlinear CA)로 구분된다. XOR연산만으로 이루어진 것을 선형CA, XOR연산 이외의 연산으로 이루어진 것을 비선형CA라고 한다. 그 중에서 XOR과 XNOR연산이 사용되는 CA는 추가적 CA(Additive CA)라고 한다. 또한 셀에 적용된 법칙의 수에 의해 구분할 수도 있다. 한 가지 법칙을 셀에 적용한 것을 균등 CA(Uniform CA), 두 가지 이상의 법칙이 사용된 CA를 하이브리드 CA(Hybrid CA)라고 한다. 그 외에 셀들의 배열 구조에 따라 1차원, 2차원, 3차원 CA로 구분될 수 있다.

CA를 구성하는 셀 중에서 가장 오른쪽 셀의 오른쪽 이웃과 가장 왼쪽 셀의 왼쪽 이웃이 존재하지 않으므로 이를 결정하는 것은 경계조건이라고 한다. 본 논문에서는 가장 오른쪽의 셀과 가장 왼쪽의 셀이 이웃한 것으로 간주하는 PBCA의 구조를 기반으로 한다.

3.2 프로그램 가능한 셀룰라 오토마타

PCA는 연산시 입력 값을 컨트롤하여 셀에 적용된 법칙을 제어할 수 있는 CA를 말한다. 아래의 그림 1은 기본적인 3-이웃 PCA구조를 나타내고 있다.

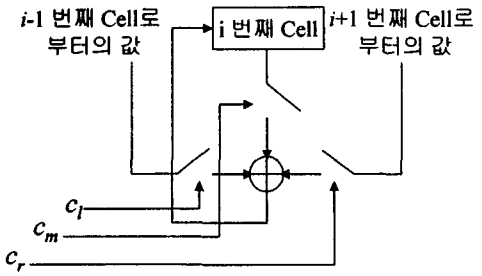

그림 1. 3개의 이웃을 가진 PCA 구조

그림 1에서 C_l은 왼쪽 이웃을 제어하는 컨트롤 값, C_m은 자기 자신을 제어하는 컨트롤 값, C_r은 오른쪽 이웃을 제어하는 컨트롤 값이다. 예를 들어 표 1에서 설명된 법칙 90은 왼쪽 이웃과 오른쪽 이웃을 XOR하여 자신의 상태를 갱신하는데 이를 구현하기 위해서 C_l=1, C_m=0, C_r=1을 컨트롤 값으로 사용하면 된다. 이와 같이 PCA 구조는 컨트롤 값을 적절히 조절하여 모든 법칙들 CA구조에 적용할 수 있다.

4. PCA를 이용한 곱셈기

본 장에서는 PCA 구조를 기반으로 AOP의 속성을 이용한 곱셈기를 제안한다. 또한 곱셈구조의 입력을 제어하는 컨트롤들의 값을 조정함으로써 효과적인 곱셈기를 설계한다. 아래의 그림은 GF(2^4)상에서 PCA 곱셈기의 구조를 나타낸다.

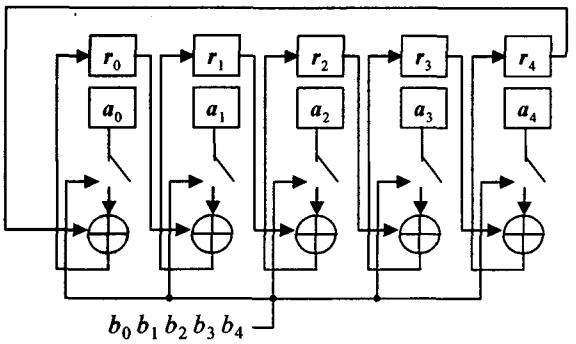

그림 2. GF(2^4)상에서 제안된 PCA 구조

GF(2^4)상에서 제안된 곱셈기는 CA의 특성 중에서 PBCA의 구조와 법칙 240을 이용한다. 그러므로 각 셀들은 자신의 왼쪽 셀 값을 항상 이용하고 오른쪽 셀의 값은 이용하지 않는다. 따라서 왼쪽 셀을 제어하는 컨트롤 값, C_l은 항상 1이 되고 오른쪽 셀을 제어하는 컨트롤 값, C_r은 항상 0이 된다. 이 때 컨트롤 값이 항상 1이 되면 연결된 선으로, 반대로 항상 0이면 연결되지 않은 선으로 나타낼 수 있으므로 제안된 곱셈기에서는 단지 한 개의 컨트롤 시그널, C_m, 만이 필요하다. C_m 컨트

를 값에는 곱셈에서 승수(Multiplier)의 값이 각 클럭마다 하나씩 입력된다. a_i ($0 \le i \le m$)에는 피승수(multiplicand)의 값이 입력되어 있고 r_i ($0 \le i \le m$)는 0으로 초기화 된다.

클럭마다 입력되는 승수(multiplier), b_i ($0 \le i \le m$)의 값에 따라 XOR연산에 입력되는 값들이 달라지는데, b_i=0 이면 각각의 셀들은 자신의 왼쪽 셀의 값에 의해서 r_i 의 값이 갱신되고 b_i=1이면 자신의 왼쪽 셀과 자신의 셀의 값을 XOR연산하여 r_i 의 셀의 값을 갱신하게 된다. 제안된 구조에서는 매 클럭 사이클마다 b_i 값에 상관없이 우측 순환 시프트를 수행한다. 이는 AOP의 속성인 $f(\alpha)=\alpha^{m+1}+1=0$를 사용하여 모듈로 감소연산을 수행하는 역할을 하게 된다. 제안된 구조의 알고리즘은 다음과 같다.

입력: $A = (a_0, a_1, \ldots, a_{m-1}, a_m)$
$B = (b_0, b_1, \ldots, b_{m-1}, b_m)$
출력: $AB = R \bmod f(\alpha) = (r_0, r_1, \ldots, r_{m-1}, r_m)$

단계 1: $(r_0, r_1, \ldots, r_{m-1}, r_m) = (0, 0, \ldots, 0, 0)$
단계 2: FOR i=0 TO m
단계 3: FOR k=0 TO m
단계 4: IF b_i=0 THEN $r_k = r_{k-1 \bmod(m+1)}$
단계 5: ELSE IF b_i=1 THEN $r_k = r_{k-1 \bmod(m+1)} \oplus a_k$

5. 분 석

본 논문에서는 효과적인 비교를 위해 Zhang이 제안한 PCA 구조와 Jeon이 제안한 CA 구조를 분석한다.

표. 2 기존 곱셈기와의 비교

	Zhang[4]	Jeon[5]	제안된 구조
기본연산	AB	AB	AB
기본 셀 수	m	m+1	m+1
셀 복잡도	3-AND + 2-XOR	1-AND + 1-XOR	1-XOR + 1-SWITCH
레지스터	4m	3(m+1)	2(m+1)
AND게이트	3m	m+1	None
XOR게이트	2m	m+1	m+1
지연시간	m	m+1	m+1

Zhang[4]이 제안한 PCA 기반의 구조는 m개의 셀마다 각각 3개의 AND 게이트와 2개의 XOR 게이트를 가지며 4m개의 레지스터가 필요하다. 이에 비해 Jeon[5]이 제안한 CA 기반의 MSB 곱셈구조는 m+1개의 셀마다 각각 1개의 AND 게이트와 1개의 XOR를 가지며 3(m+1)개의 레지스터가 필요하다.

본 논문에서 제안된 곱셈기 구조에서는 Zhang의 구조에 비해서 1개의 클럭 사이클이 더 필요하지만 단지 m+1개의 XOR 게이트와 m+1개의 SWITCH, 그리고 2(m+1)개의 레지스터만이 사용된다. 이는 Zhang의 CA 구조에 비하여 약 50% 정도의 하드웨어 복잡도를 줄인 것이다.

6. 결 론

본 논문에서는 기약다항식으로 AOP를 이용하고 PBCA의 특성에 기반한 효율적인 곱셈기 구조를 제안하였다. 제안된 구조는 PBCA의 구조적인 특성을 AOP의 특성과 조화시킴으로써, 단지 한 개의 컨트롤 시그널로 PCA의 구조에 적합한 곱셈기를 설계하였다. 또한 AND 게이트 없이 m+1개의 XOR게이트와 m+1개의 SWITCH, 그리고 2(m+1)개의 레지스터로 구성되어 있어, 기존에 제안된 구조에 비하여 약 50% 정도의 하드웨어 복잡도를 줄였다. 제안된 PCA의 구조에 기반한 곱셈기는 정규성이 뛰어나고 구조 복잡도와 시간 복잡도가 낮아 오류수정코드나 지수연산을 위한 하부구조로써 효율적으로 이용될 것으로 기대한다.

참 고 문 헌

[1] E. R. Berlekamp, Bit-serial Reed-Solomon encoders, IEEE Trans. IT-28, Vol. 6, pp. 869~874, 1982.

[2] T. R. N. Rao and E. Fujiwara, Error- Control Coding for Computer Systems, Engle-wood Cliffs, NJ: Prentice-Hall, 1989.

[3] J. Von Neumann, The theory of self-reproducing automata, University of Illinois Press, Urbana and London, 1966.

[4] C. N. Zhang and M. Y. Deng, and R. Mason, A VLSI Programmable Cellular Automata Array for Multiplication in GF(2^m), PDPTA 99 International Conference.

[5] 전준철, 김현성, 이형목, 유기영, GF(2^m)상의 셀룰라 오토마타를 이용한 VLSI 구조, *정보보호학회논문지*, Vol. 12, No. 3, pp. 87~94, June. 2002.

[6] Lidl R. and Niederreiter H. An introduction to finite field and their applications, CUP, Cambridge, 1994.

[7] A. K. Das, P. Pal. Chaudhuri, Efficient characterization of cellular automata, IEE Proceedings, Vol. 137, Part. E, pp. 81~87, January 1990.