GF(q^n)상의 병렬 승산기 설계를 위한 기약다항식에 관한 연구

오진영, 김상완, 황종학, 박승용, 김흥수
인하대학교 전자공학과
인천광역시 남구 용현동 253 인하대학교
e-mail : jinga@ee.inha.ac.kr

A Study on Irreducible Polynomial for Construction of Parallel Multiplier Over GF(q^n)

Jin-Young Oo, Sang-Wan Kim, Jong-Hak Hwang, Seung-Yong Park, Heung-Su Kim
Dept. Of Electronic Eng. Inha Univ,
253 Yong-hyundong Namgu Incheon Seoul. Korea
e-mail : jinga@ee.inha.ac.kr
FAX : 82-32-868-3654

ABSTRACT

In this paper, We represent a low complexity of parallel canonical basis multiplier for GF(q^n) ,(q> 2). The Mastrovito multiplier is investigated and applied to multiplication in GF(q^n) . GF(q^n) is different with GF(2^n), when MVL is applied to finite field. If q is larger than 2, inverse should be considered. Optimized irreducible polynomial can reduce number of operation. In this paper we describe a method for choosing optimized irreducible polynomial and modularizing recursive polynomial operation.

A optimized irreducible polynomial is provided which perform modulo reduction with low complexity. As a result, multiplier for fields GF(q^n) with low gate counts and low delays are constructed. The architectures are highly modular and thus well suited for VLSI implementation.

1. 서론

최근 집적회로 기술의 비약적인 발전으로 인해 단일 칩 상에 방대한 양의 회로가 집적 될 수 있게 되었지만 복잡하고 다양한 기능을 구현하기 위해 더 많은 소자들을 더 적은 면적의 칩 속에 집적해야 하는 것이 현재 집적회로 기술이 해결해야 할 과제로 떠오르고 있다. 그것의 해결 방법으로 최근 주목받고 있는 분야가 다치논리 이론을 회로에 적용하는 것이다. 이는 하나의 신호선에 오직 두 개의 신호레벨만을 전송하는 것 보다 동일한 신호선에 더 많은 신호를 전송함으로써 내부결선의 복잡성을 감소시킬 수 있는 장점이 있다.

유한체는 컴퓨터 네트웍 및 통신 시스템등의 코딩, 암호화 등에 널리 쓰이고 있다. GF(2^m)상의 연산은 BCH부호, Reed-Solomon부호, digital signal processing, error control coding과 보안 통신에 요구되는 암호화와 복호화등에서 상용되며, 이에 따라 승산과 역원계산에 관한 연구가 많이 이루어져 왔다. [4,5,]

본 논문에서는 다치논리 이론을 유한체상에 적용시켜 승산기를 설계하였다. 승산기 설계는 Mastrovito가 제안한 알고리즘을 기반으로 하였고, 이 알고리즘에서 발생하는 기약다항식을 최적화 하는 방법에 대해서 기술하였다. 기약다항식에 따라 게이트 수의 변화를 설명하기 위해 GF(3^4)상의 기약다항식들을 이용해서 승산기를 설계하고 각각 게이트들의 개수를 비교, 검토하였다.

2. GF(q^n)상에서의 승산기

오류정정부호인 BCH부호와 Reed-Solomon부호의 복호과정에서는 GF(q^n)상의 연산을 이용하여 관련 알고리즘을 나타낼 수 있다. 유한체 연산의 간략화는 전체 복호회로의 복잡성에 매우 큰 영향을 미친다. 따라서 부호기와 복호기에 적용될 GF(q^n)상에서의 연산에 대하여 살펴보았다. 원소의 개수가 q^n개인 유한체 GF(q^n)상의 각 원소들의 표현방법은 크게 벡터 표현과 지수표현의 두 가지로 나눌 수 있는데, 벡터표현과 지수표현에 의한

GF(q^n)상의 임의의 두 원소들은 다음과 같이 나타내었다.

$$A(x) = a_{m-1}\, x^{m-1} + \ldots + a_0,\ a_i \in GF(q) \quad (1)$$
$$B(x) = b_{m-1}\, x^{m-1} + \ldots + b_0,\ b_i \in GF(q) \quad (2)$$
$$C(x) = c_{m-1}\, x^{m-1} + \ldots + c_0,\ c_i \in GF(q) \quad (3)$$

유한체에서의 원소A와 B의 승산은 다음과 같다.

$$C(x) = C'(x) \bmod P(x) = A(x) \times B(x) \bmod P(x) \quad (4)$$

여기서 사용되는 계수들의 모든 연산은 GF(q^n)내에서 수행되고 GF(q^n)에서의 원시 기약 다항식은 다음과 같이 나타내었다.

$$P(x) = x^n + p_{n-1}\, x^{n-1} + \ldots + p_0,\ p_i \in GF(q) \quad (5)$$

α가 이 식의 근이라고 가정하면 P(α) = 0 이므로, $a^n = d_{n-1}\, a^{n-1} + \ldots + d_0,\ d_i \in GF(q)\ (d_i = q - p_i)$와 같은 식이 되고 각 원소들을 차수가 n-1 이하인 a의 다항식들을 이용해서 GF(q^n)상의 각각 q^n 개의 원소를 표현하였다.

GF(q^n)상에서의 연산과 GF(2^n)상에서의 연산은 차이가 있다. 즉 2치 상에서의 연산이 아니라, 다치논리가 적용되게 된다. 기약다항식을 이용해서 원소를 생성할 때 감산이 발생하게 되는데, 이것은 덧셈에 대한 역원을 이용해서 나타내었다.

GF(q^n)상에서의 승산을 Mastrovito가 제안한 승산 알고리즘을 이용해서 유한체 상에서의 승산기를 구성하였다. GF(q^n)상에서 원소를 A(x), B(x), C(x)라 하면 승산은 C(x)=(A(x) · B(x)) mod P(x)가 되고 원소는 n 보다 작은 차수의 다항식을 가지게된다.

$$c_{n-1}\, x^{n-1} + \ldots + c_0 = (a_{n-1}\, x_{n-1} + \ldots + a_0) \times (b_{n-1}\, x^{n-1} + \ldots + b_0) \bmod P(x) \quad (6)$$

유한체에서의 승산은 위 식에서 볼 수 있는 바와 같이 일반적인 다항식 승산을 하고, 다항식의 모듈러 연산을 거치게 된다. 여기서, A(x)는 피승수이고, B(x)는 승수이다. 따라서 A(x)와 P(x)로 생성행렬 Z를 만들어 낼 수 있다. 생성행렬 Z= f(A(x),P(x))는 다음 식(7)과 같은 행렬식으로 나타낼 수 있다.

$$C = \begin{pmatrix} c_0 \\ c_1 \\ \vdots \\ c_{n-1} \end{pmatrix} = ZB = \begin{pmatrix} f_{0,0} & \cdots & f_{0,n-1} \\ \vdots & \vdots & \vdots \\ f_{n-1,0} & \cdots & f_{n-1,n-1} \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_{n-1} \end{pmatrix} \quad (7)$$

계수 $f_{ij} \in$GF(q)는 a_i와 p_i에 의해서 표현 될수 있다. 여기서 c_i는 일반적인 다항식 승산을 하고, 다항식의 모듈러 연산을 결과를 합한 것임을 알고 있다. 따라서 원시 기약다항식에 의해서 $f_{i,j}$ 가 결정되고 이것은 승산기의 구조를 좌우하게 된다.

3. GF(q^n)에서의 기약다항식

유한체 연산에서 2치 연산이 아닌 3치이상의 연산에서 고려해야 할 문제점은 원시 기약다항식을 이용해서 모듈러 연산을 할 경우, 감산에 대하여 고려해 보아야 한다. 본 논문에서는 감산의 경우 가산 역원을 이용해서 감산의 문제를 해결하였다.

즉, GF(q)상의 임의의 원소 $-p$는 유한체 특성에 의해서 $q-p$로 표현 될 수 있다. 유한체 2치 연산의 경우, -1 = 2-1 = 1 임을 알 수 있다. 따라서 오류정정 코드로 주로 사용되고 있는 GF(2^n)에서는 감산과 역원의 문제가 발생하지 않는다. 이것이 GF(q^n)상으로 확장될 경우, 다항식의 선택과 감산의 고찰은 승산기 구조, 크기, 연산 속도, 복잡도, 규칙성에 큰 영향을 미치게 된다. 본 장에서 원시기약다항식의 특성과 활용에 대해서 다항식들의 특성을 정리하였다.

원시기약다항식 (5)에 이식의 근 α를 대입하면, P(α) = 0 이 된다.

$$a^n + p_{n-1}\, a^{n-1} + \ldots + p_0 = 0,\ p_i \in GF(q) \quad (8)$$

(8)식을 a^n에 관해 정리하면 다음 식과 같이 된다.

$$a^n = d_{n-1}\, a^{n-1} + \ldots + d_0,\ d_i \in GF(q) \quad (9)$$
$$(d_i = q - p_i)$$

위 식과 같은 일반적인 형태의 다항식을 이용해서 다항식의 계수의 변화를 살펴보았다. 위 (6)식의 계수들을 다시한번 살펴보면

계수 $f_{ij}\in GF(P)$는 a_i와 q_i에 의해서 표현 될수 있다.

$j=0$; $i=0, \ldots, n-1$이면 a_i이고, $j=1, \ldots, n-1$; $i=0, \ldots, n-1$이면 $u(i-j)\, a_{i-j}+\sum_{t=0}^{i-1} d_{j-1-t},\ a_{n-1-t}$이 된다.

$u(t)=1(t\geq 0)$

일반적인 곱셈을 수행한 후 다항식이 n차 이상이 되는 경우 다음 행렬과 같이 나타낼 수 있다.

$$\begin{pmatrix} x_n \\ x_{n+1} \\ \vdots \\ x_{2n-2} \end{pmatrix} \equiv \begin{pmatrix} q_{0,0} & \cdots & q_{0,n-1} \\ \vdots & \vdots & \vdots \\ q_{n-1,0} & \cdots & q_{n-1,n-1} \end{pmatrix} \begin{pmatrix} 1 \\ x \\ \vdots \\ x_{n-1} \end{pmatrix} \bmod P(x) \quad (10)$$

Q행렬은 Z행렬을 만드는데 필요하다. n차의 기약다항식 $P(x)$을 이용하여 x^n, x^{n+1}, ... x^{2n-2},를 mod $P(x)$에서의 다항식으로 바꾸어 주는 Q행렬은 $a^n+p_{n-1}a^{n-1}+\ldots+p_0=0$, $p_i\in GF(q)$의 계수들을 이용하여 구한다.

계수 $q_{i,j}$는 $j=0$, $i=1,\ldots,n-2$인 경우 $q_{i-1,n-1}$이고, $i=1, \ldots, n-2$, $j=1, \ldots, n-2$인 경우, $q_{i-1,n-1}\ q_{i-1,n-1}\ q_{0,j}$ 가 된다. 원소 $A(x)$는 고정되어 있고 원소$B(x)$를 곱한다.

[예.1] GF(3^2)상의 원시 다항식 $P(x)=x^2+x+2$로 놓고 원시근을 α로 정의하면 $P(\alpha)=0$이다.

고정된 생성행렬(Z)을 만들기 위한 $A(x)=\alpha^6=\alpha+2$로 놓으면 생성행렬Z는 다음과 같은 형태로 나타난다.

$$C=\begin{bmatrix} c_0 \\ c_1 \end{bmatrix}=\alpha^6 B=ZB=\begin{bmatrix} 2b_0+b_1 \\ 2b_1+(b_0+2b_1) \end{bmatrix} \quad (12)$$

4. GF(3^n)에서의 승산

다치 논리에서 지금 현재 가장 활발히 연구 중인 3치에서의 유한체 승산기에 대해서 고려해 보았다.

2치(binary)의 경우 0, 1의 값을 가지고 연산을 하는 반면, 3치는 하나의 레벨을 더 추가하여 0, 1, 2의 값을 이용하여 연산을 수행하게 된다.[4,9]

3치 유한체 승산기에서 필요한 기약다항식을 살펴보게 되면 2진 연산과는 달리 복잡한 구조를 가짐을 알수 있다. [1] 기약 다항식과 승산기의 구조를 고려해 볼 때, 삼항식이 가장 효율적인 것으로 알려졌다. 하지만 3치 이상에서는 3항식이 하나이상 존재하게 된다.

예로 GF(3^4)에서의 기약다항식들을 살펴보게 되면 4차의 기약 다항식은 모두 18개가 존재하고, 그중에서 3^4개의 원소를 가지고 있어서 승산기 설계에 적합한 기약다항식은 8개가 된다. 위 (8)식에서와 같이 Q matrix는 기약다항식에 의해서 결정된다.

승산기의 복잡도의 결정도 이것에 의해서 결정되는 것을 알 수 있다. 또한 위에서 설명한 바와 같이 (8)식은 다항식의 항의 수가 적을수록, 고차항의 계수가 없을수록 간결해 진다. 2진 연산과는 달리 (9)식과 같이 $d_i=q-p_i$와 같은 결과가 발생한다.

GF(3^4)상에서의 기약다항식의 일반적인 형태로 나타내어 식 (6)에 대입해 보면 다음 식들과 같이 나타낼 수 있다.

$$f_{0,0}=a_0 \quad f_{0,1}=a_3 d_0, \quad f_{0,2}=a_3 d_3 d_0+a_2 d_0$$
$$f_{0,3}=a_3(d_3 d_3 d_0+d_2 d_0)+a_2(d_3 d_0)+a_1(d_0)$$
$$f_{1,0}=a_1 \quad f_{1,1}=a_3 d_1+a_0 \quad f_{1,2}=a_3(d_3 d_1+d_0)+a_2 d_1$$
$$f_{1,3}=a_3(d_3 d_3 d_1+d_2 d_1+d_3 d_2)+a_2(d_3 d_1+d_0)+a_1 d_1$$
$$f_{2,0}=a_2 \quad f_{2,1}=a_3 d_2+a_1 \quad f_{2,2}=a_3(d_3 d_2+d_1)+a_2 d_2+a_0$$
$$f_{2,3}=a_3(d_3 d_3 d_2+d_2 d_2+d_3 d_3+d_0)+a_2(d_3 d_2+d_1)+a_1 d_2$$
$$f_{3,0}=a_3 \quad f_{3,1}=a_3 d_3+a_2 \quad f_{3,2}=a_3(d_3 d_3+d_2)+a_2 d_3+a_1$$
$$f_{3,3}=a_3(d_3^3+d_3 d_2+d_3 d_3+d_0)+a_2(d_3 d_1+d_0)+a_1 d_1 \quad (13)$$

위 식에서 차수가 높은 계수일수록 사용 빈도가 높은 것을 알 수 있다. 다음으로 고려해야 할 것은 계수의 값이다. 계수의 값은 0, 1, 2가 될 수 있는데 기약 다항식의 계수의 값이 1의 값을 갖게 되면 $d_i=q-p_i$ 에 의해서 식 (6)에서 보면 C의 다항식에는 2의 계수가 발생한다. 즉 2를 곱해 주어야 한다.

표-1은 GF(3^4)상의 기약다항식들을 이용해서 승산기를 설계할 경우 각각의 승산기에 사용되는 연산의 수를 나타내었다. 표에서 보면 알 수 있는 바와 같이 항이 적을수록, d_i의 값이 1일 경우 복잡도가 줄어드는 것을 알 수가 있다.

표-1을 살펴보면 다항식의 계수중 1의 개수가 많을수록 승산게이트가 줄어들고, 항의 개수가 많을수록 승산과 가산게이트 늘어 날수 있다. 항의 위치에 따른 게이트의 변화를 살펴보면, 고차 항의 위치에 있을수록 전체적인 게이트의 수는 늘어 나고 있다.

표 1 GF(3^4)상의 승산기

다항식	# of ⊗	# of ⊕
10012	19	18
10022	16	18
11002	22	17
11122	27	25
11222	24	25
12002	16	19
12112	26	26
12212	24	26

5. 결론

본 논문은 유한체 승산기 즉, GF(q^n)상에서의 승산기의 설계에 중요한 역할을 하는 원시 기약다항식에 대해서 설명하였다. 승산기는 Mastrovito의 알고리즘을 사용하였다. 이 승산기는 모듈화와 규칙성이 좋기때문에 VLSI에 적합하여 많이 사용되는 승산기이다.

Mastrovito 승산기를 이용해서 연산기 게이트 개수를 줄이기 위한 방법으로 기약 다항식의 선택 방법을 고려하였다. 위 설명에서 본 바와 같이 기약 다항식에 대해서 두 가지 측면을 제시하였다.

(1) 기약다항식 항의 개수
(2) 기약다항식 항의 위치
(3) 기약다항식 항의 크기

기약다항식의 최저 항수는 삼항식이다. GF(3^4)에서 예를 들어 설명한 바와 같이 삼항식이 연산기의 개수를 줄일 수 있다. 삼항식 중에서 차수가 높은 계수는 사용빈도가 높기 때문에 차수가 낮은 계수를 선택하였다..

본 논문에서 예를 들어 설명한 유한체 승산기는 3치에서 설계하였는데, 이 경우 덧셈에 대한 역원을 고려하여야 한다. 유한체 연산에서는 이진 연산에서는 고려되지 않았던 점들이다. $d_i = q - p_i$ 의 값이 1이 되면 연산의 수를 줄일 수 있으므로 복잡도를 줄일 수 있다.

이상과 같이 유한체 승산기에서의 기약다항식과 게이트 개수와의 관계를 설명하였는데, 다치 연산기는 2진 연산기와 차이가 있고 현재 꾸준한 연구가 이루어지고 있다.

[참고 문헌]

[1] Rudolf Lidl, Harald Niederreiter, "Introduction to finite fields and their application" Cambridge Uni. press, 1994

[2] M. Kameyama and T. Higuchi, "Multiple-Valued logic and special Purpose Processors : Overview and Futurre", The 12th Int'l Symp. M.V.L pp. 289-292, May 1982

[3] K. C. Smith, "Multiple-Valued Logic : A Tutorial and Application", Com. Mag, pp.17-27, April. 1988

[4] 김태한, "GF(3^m)상의 승산기 및 역원생성기 구성에 관한 연구", 인하대학교 석사학위 청구 논문, Feb. 1990

[5] M.A. Hasan, V.K. Bhargava, "Division and Bit-serial Multiplication over GF(q^m)" IEE Proceeding-E, Vol.139, No.3, May 1992

[6] D.H.Green and I.S.Taylor,"Irreducible Polynomials over ComPosite Galois Fields and Their Appplications in Coding TechniQues," Proc.IEE,vol.121, no.,pp.935-39, Sept.1974.

[7]M.A.Hasan,M.Wang, and V.K.Bhargava, "Division and Bit-Serial Multiplication overGF(q^m),"IEEE Trans. Computers,vol.41,no.8pp962-971,Aug.1992

[8] 서근옥, '1,2-차원 배열 구조를 갖는 GF(p^m)상의 승산기 구성에 관한 연구", 인하대학교 석사학위 청구 논문, Feb. 1993.

[9] E.D. Mastrovito, "VLSI Design for Mutiplication over Finite Fields GF(2^m),"Lecture Notes in Computer Science 357,pp.297-309 Berlin:Springer-verlag,mar.1989

[10] E.D.Masrovito,"VLSI Architectures for Computati -on in Galois Fields,"PhD thesis, Dept.of Electrical Eng.,Linkoping Univ.,Linkoping,Swesen,1991