Journal of The Korea Knowledge Information Technology Society

ISSN 1975-7700

http://www.kkits.or.kr

Design of Two-times Faster Bit-Serial Multiplier Using Normal Basis of $GF(2^m)$

Yong-Suk Cho, Kyoung-Il Min*

Department of IT & Securities, U1 University

ABSTRACT

Finite field arithmetic has recently gained growing attention due to its wide range of applications in signal processing, error control coding and especially in cryptography. In these applications, there is a need to design low complexity finite field arithmetic units. One important factor that could greatly affect computation performance is the basis in which finite field elements are represented. Among the basis, representation of fields elements using a normal basis is quite attractive for hardware implementation since the squaring operation over $GF(2^m)$ can be performed by only one-bit cyclic shift to left. In this paper, we propose a new architecture of two-times faster bit-serial finite field multiplier using a normal basis. In the proposed multiplier, the bits of an operand are grouped into the two parts and each part is implemented simultaneously by bit-serial multiplier. Therefore, the proposed multiplier takes $\lceil m/2 \rceil$ clock cycles, to finish one multiplication operation in a binary field of size m. The proposed architecture is two-times faster than bit-serial architectures but with lower area complexity than bit-parallel ones. It is shown that the new design has higher regular architecture compared to other similar proposals and therefore, well-suited for VLSI implementation. The main advantage of the proposed architecture is that a trade-off between hardware complexity and delay time can be achieved. Therefore, the proposed multipliers are more suitable for resource constrained cryptographic systems where the value of m is large but space is of concern.

KEYWORDS: Finite fields, Galois fields, Normal Basis, Bit-serial multipliers, Cryptography

ARTICLE INFO: Received 24 April 2017, Revised 9 June 2017, Accepted 9 June 2017.

*Corresponding author is with the Department of IT & Securities, U1 University, 52-70 Yeonamsan-ro Eumbong-myeon Asan-si Chungcheongnam-do KOREA.
E-mail address: kyilmin@u1.ac.kr

1. 서 론

유한체(finite fields)는 컴퓨터와 디지털 통신 시스템 등과 같은 여러 분야에서 널리 사용되고 있다. 특히 암호화(cryptography)와 오류정정부호(error correcting codes)에서는 유한체 상의 연산이 시스템의 구현에 매우 중요한 요소가 된다[1],[2].

유한체 연산 중에서 덧셈은 비트별 Exclusive OR (XOR)로 쉽게 구현할 수 있는 반면에 곱셈과 나눗셈은 상당히 복잡한 연산이다. 유한체의 곱셈과 나눗셈 중에서 나눗셈은 곱셈을 반복하여 수행할 수 있으므로 곱셈이 가장 중요한 연산이 된다[3].

유한체의 곱셈기는 비트병렬(bit-parallel) 방법과 비트직렬(bit-serial) 방법으로 구현할 수 있다. 비트병렬 곱셈기는 조합논리 회로로 구현되므로 곱셈의 결과는 통과하는 논리 게이트의 지연 후에 출력된다. 반면에 비트직렬 곱셈기는 순서논리 회로로 구현되므로 일반적으로 m 클럭 만큼의 시간 지연 후에 결과를 출력한다. 비트병렬 곱셈기는 연산속도는 비트직렬 곱셈기에 비해 빠르지만 회로 면적이 커지므로 유한체 $GF(2^m)$에서 m이 매우 큰 암호 분야의 응용에는 적합하지 않다[4],[5]. 반면에 비트직렬 곱셈기는 회로는 간단하지만 일반적으로 m 클럭 만큼의 시간 지연 후에 결과를 출력한다[6],[7].

이러한 문제점을 해결하기 위한 방법이 직병렬 혼합 곱셈기이다. 직병렬 혼합 곱셈기는 회로의 복잡도와 지연 시간 사이의 적절한 절충을 꾀하는 것이다. 직병렬 혼합 곱셈기는 기존의 비트직렬 곱셈기보다는 짧은 지연시간에 결과를 얻을 수 있으며, 비트병렬 곱셈기보다는 적은 하드웨어로 구현할 수 있다[8]-[12].

본 논문에서는 직병렬 혼합 방법을 사용하여 기존의 비트직렬 곱셈기에 비해 2배로 빨리 곱셈의 결과를 출력할 수 있는 2배속 비트직렬 곱셈기를 설계한다.

유한체 $GF(2^m)$ 상의 연산에서 어떠한 기저를 사용할 것인가 하는 문제는 연산기 구현의 효율성에 중대한 영향을 미친다. 따라서 여러 가지 기저를 사용한 연산 방법들이 연구되고 있다. 주로 사용되는 기저로는 다항식기저(polynomial basis)[13], 쌍대기저(dual basis)[14], 정규기저(normal basis)[15] 등이 있다. 이 중에서 정규기저를 사용하면 한 비트 순회치환(cyclic shift)만으로 제곱 연산을 구현할 수 있다. 따라서 정규기저는 여러 가지 타원곡선 암호시스템의 표준에서 널리 사용되고 있다[16],[17]. 본 논문에서는 정규기저에서 동작하는 곱셈기를 설계한다.

본 논문의 구성은 다음과 같다. 먼저 2.에서 유한체 $GF(2^m)$의 정규기저를 이용한 비트직렬 곱셈 알고리즘을 분석한다. 그리고 3.에서 직병렬 혼합 방법을 사용하여 새로운 2배속 비트직렬 정규기저 곱셈기를 설계한다. 또한 실제 예로써, 유한체 $GF(2^5)$의 정규기저 상에서 3 클럭만에 곱셈의 결과를 출력하는 2배속 비트직렬 곱셈기를 설계한다. 그리고 4.에서 결론을 맺는다.

2. 유한체 $GF(2^m)$의 정규기저 표현과 비트직렬 곱셈기 설계

유한체 $GF(2^m)$은 2^m개의 원소를 가지고 있으며 그 원소들은 2진 계수를 갖는 $m-1$차 이하의 다항식으로 표현할 수 있다. 즉 유한체 $GF(2^m)$은 2진체(binary field)인 $GF(2)$의 m차원 벡터공간(vector space)이 된다. 여기에서 선형독립인 m개의 벡터는 모두 기저(basis)가 된다.

유한체 $GF(2^m)$에서 다음과 같은 m개의 서로 독립인 원소들을 유한체 $GF(2^m)$의 정규기저라고

한다. 여기에서 β는 $GF(2^m)$의 한 원소이다.

$$\{\beta,\ \beta^2,\ \beta^{2^2},\cdots\cdots,\ \beta^{2^{m-1}}\} \tag{1}$$

유한체 $GF(2^m)$의 임의의 한 원소 A는 정규기저를 이용하면 다음과 같이 표현할 수 있다.

$$\begin{aligned} A &= \sum_{i=0}^{m-1} a_i \beta^{2^i}, \quad a_i \in GF(2) \\ &= a_0\beta^{2^0} + a_1\beta^{2^1} + \cdots\cdots + a_{m-1}\beta^{2^{m-1}} \end{aligned} \tag{2}$$

여기에서 $\beta^{2^m} = \beta$이므로, 다음과 같이 된다.

$$A^2 = a_{m-1}\beta + a_0\beta^2 + \cdots + a_{m-2}\beta^{2^{m-1}} \tag{3}$$

따라서 왼쪽으로 한 비트 순회치환(cyclic shift)하면 제곱 연산을 구현할 수 있다.

유한체 $GF(2^m)$의 임의의 두 원소를 A와 B라 하고 이 두 원소의 곱을 C라고 하면, C는 식 (2)에 따라 다음과 같이 쓸 수 있다.

$$C = A \cdot B = A \cdot \left(\sum_{i=0}^{m-1} b_i \beta^{2^i} \right) \tag{4}$$

식 (4)를 풀어 쓰면 다음과 같이 된다.

$$\begin{aligned} C &= b_0(A\beta) + b_1(A\beta^2) + \cdots\cdots \\ &\quad + b_{m-1}(A\beta^{2^{m-1}}) \\ &= b_0(A\beta) + b_1(A^{2^{-1}}\beta)^{2^1} + \cdots\cdots \\ &\quad + b_{m-1}(A^{2^{-(m-1)}}\beta)^{2^{m-1}} \end{aligned} \tag{5}$$

식 (5)는 다음과 같이 다시 정리할 수 있다.

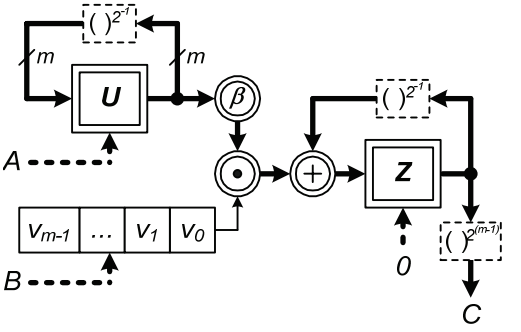

그림 1. $GF(2^m)$의 비트직렬 정규기저 곱셈기
Figure 1. Bit-serial normal basis multiplier over $GF(2^m)$

$$\begin{aligned} C &= (C^{2^{-(m-1)}})^{2^{(m-1)}} \\ &= ((\cdots((b_0\beta A)^{2^{-1}} + b_1\beta A^{2^{-1}})^{2^{-1}} \cdots \\ &\quad + b_{m-2}\beta A^{2^{-(m-2)}})^{2^{-1}} \\ &\quad + b_{m-1}\beta A^{2^{-(m-1)}})^{2^{(m-1)}} \end{aligned} \tag{6}$$

식 (6)을 회로로 구현하면 <그림 1>과 같이 된다. <그림 1>에서 □는 m 비트 레지스터를 표현한 것이다. 또한 ⊕는 m 개의 2입력 XOR 게이트이고, ⊙는 m개의 2입력 AND 게이트이다. Ⓑ는 $GF(2^m)$의 한 원소 β를 곱하는 상수 곱셈기이며, 굵은 선은 m 비트 버스이다. 점선으로 된 사각형은 2의 멱승을 수행하는 것으로 정규기저 상에서는 결선만 바꾸면 된다. 정규기저 표현에서 2^{-1}승은 각 계수를 왼쪽으로 한 번 순회치환하면 되고 $2^{(m-1)}$ 승은 각 계수를 오른쪽으로 $m-1$번 순회치환하는 것이다.

<그림 1>의 회로는 초기 상태에서 레지스터 U에는 입력 A를 로드시키고, 레지스터 V에는 입력 B를 로드시킨다. 그리고 레지스터 Z는 클리어시킨 다음 각 레지스터를 m번 치환시키면 곱셈의 결과가 레지스터 Z에 저장된다. 따라서 m 클럭 시간 후에 결과를 얻을 수 있다.

3. 유한체 $GF(2^m)$ 상의 2배속 비트직렬 정규기저 곱셈기 설계

<그림 1>과 같은 비트직렬 곱셈기는 m 클럭 지연 후에 곱셈의 결과를 출력한다. 지연 시간을 줄이기 위하여 먼저 식 (5)를 2개로 분할한다. 그리고 각각을 비트직렬 곱셈기로 구현하고 병렬로 더하면 속도를 2배로 향상시킬 수 있다.

$$C = C_0 + C_1 \tag{7}$$

C_0와 C_1은 다음과 같이 쓸 수 있다. 여기에서 w는 $m/2$ 보다 큰 최소 정수, 즉 $\lceil m/2 \rceil$ 이다.

$$C_0 = b_0(A\beta) + b_1(A\beta^{2^1}) + \cdots\cdots + b_{w-1}(A\beta^{2^{w-1}}) \tag{8}$$

$$C_1 = b_w(A\beta^{2^w}) + b_{w+1}(A\beta^{2^{w+1}}) + \cdots\cdots + b_{m-1}(A\beta^{2^{m-1}}) \tag{9}$$

식 (8)은 식 (5)와 동일한 구조를 가지고 있으므로 다음과 같이 정리할 수 있다.

$$C_0 = ((\cdots((b_0\beta A)^{2^{-1}} + b_1\beta A^{2^{-1}})^{2^{-1}}\cdots + b_{w-2}\beta A^{2^{-(w-2)}})^{2^{-1}} + b_{w-1}\beta A^{2^{-(w-1)}})^{2^{(w-1)}} \tag{10}$$

같은 방법으로 식 (9)를 정리하면 다음과 같이 된다.

$$C_1 = (b_w(A^{2^{-w}}\beta) + b_{w+1}(A^{2^{-w}}\beta^{2^1}) + \cdots\cdots + b_{m-1}(A^{2^{-w}}\beta^{2^{w-1}}))^{2^w} \tag{11}$$

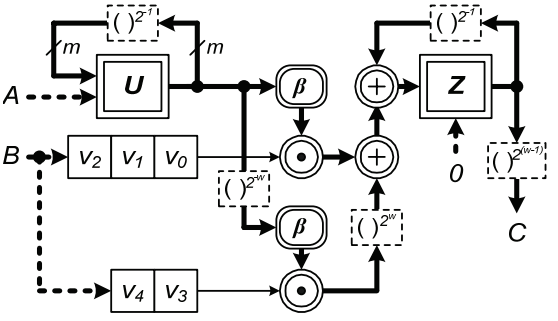

그림 2. $GF(2^m)$ 상의 2배속 비트직렬 정규기저 곱셈기
Figure 2. The two times faster bit-serial normal basis multiplier over $GF(2^m)$

식 (11)을 살펴보면 A 대신에 $A^{2^{-w}}$를 대입하면 식 (8)과 동일한 구조가 된다. 따라서 식(10)과 식 (11)을 이용하여 2배속 비트직렬 정규기저 곱셈기를 설계하면 <그림 2>와 같이 된다.

원시다항식이 $p(x) = 1 + x^2 + x^5$ 인 유한체 $GF(2^5)$에서 임의의 두 원소 A와 B의 곱 C는 다음과 같이 정리할 수 있다.

$$C = A \cdot B = b_0(A\beta) + b_1(A\beta^{2^1}) + b_2(A\beta^{2^2}) + b_3(A\beta^{2^3}) + b_4(A\beta^{2^4}) \tag{12}$$

식 (12)를 식 (7)과 같이 2개로 나누면 다음과 같이 된다.

$$C_0 = b_0 A\beta + b_1 A\beta^2 + b_2 A\beta^{2^2} \tag{13}$$

$$C_1 = b_3 A\beta^{2^3} + b_4 A\beta^{2^4} + 0\,A\beta^{2^5} \tag{14}$$

식 (13)을 식 (10)과 같이 정리하면 다음과 같이 된다.

$$C_0 = (((b_0\beta A)^{2^{-1}} + b_1\beta A^{2^{-1}})^{2^{-1}} + b_2\beta A^{2^{-2}})^{2^2} \quad (15)$$

여기에서 w는 $\lceil 5/2 \rceil = 3$이므로, 식 (14)는 다음과 같이 정리할 수 있다.

$$C_1 = (b_3(A^{2^{-3}})\beta + b_4(A^{2^{-3}})\beta^2 + 0(A^{2^{-3}})\beta^{2^2})^{2^3} \quad (16)$$

$GF(2^5)$의 임의의 한 원소 A에 $\beta(=\alpha^5)$를 곱하면

$$A \cdot \beta = a_1\beta + (a_0 + a_3)\beta^2 + (a_3 + a_4)\beta^4 + (a_1 + a_2)\beta^8 + (a_2 + a_4)\beta^{16} \quad (17)$$

가 된다. 여기에서는 식 (18)과 같은 유한체 $GF(2^5)$의 타입 2 가우시안 정규기저(Gaussian Normal Basis) 상의 곱셈 행렬(multiplication matrix)을 사용하였다[16].

$$\mathbf{M} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} \quad (18)$$

따라서 식 (15)~(17)을 이용하면 <그림 3>과 같이 $GF(2^5)$ 상의 2배속 워드병렬/비트직렬 정규기저 곱셈기를 설계할 수 있다. $\lceil 5/2 \rceil = 3$이므로 <그림 3>의 곱셈기는 3 클럭만에 곱셈의 결과를 출력한다.

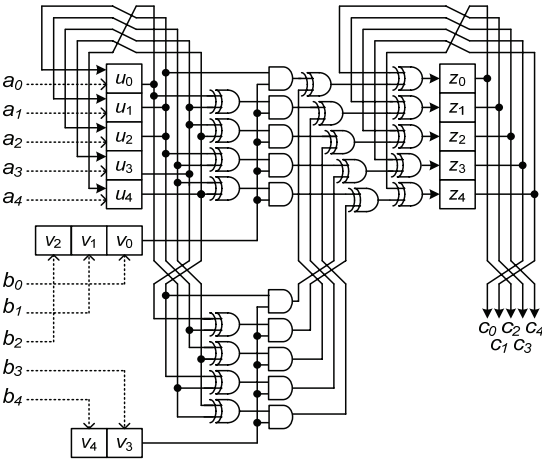

그림 3. $GF(2^5)$ 상의 2배속 비트직렬 정규기저 곱셈기
Figure 3. The two times faster bit-serial normal basis multiplier over $GF(2^5)$

4. 결 론

본 논문에서는 유한체의 정규지저 상에서, 직병렬 혼합 방법을 사용하여 기존의 비트직렬 곱셈기에 비해 2배로 빨리 곱셈의 결과를 출력하는 새로운 2배속 비트직렬 곱셈기를 설계하였다.

본 곱셈기는 2개의 비트직렬 곱셈기를 병렬로 사용하여 $\lceil m/2 \rceil$ 클럭만에 곱셈의 결과를 얻을 수 있다. 또한 본 곱셈기는 기존의 곱셈기에 비해서 매우 규칙적인 구조를 가지고 있기 때문에 VLSI 구현 등에 적합한 장점을 가지고 있다.

References

[1] M. Y. Rhee, *Error-correcting coding theory*, Mcgraw-Hill, 1989.

[2] D. Hankerson, A. Menezes, and S. Vanstone, *Guide to elliptic curve cryptography*, Springer-Verlag, 2004.

[3] R. Lidl, and H. Niederreiter, *Introduction to finite fields and their applications*, Cambridge

University Press, 1994

[4] C. K. Koc, and B. Sunar, *Low-complexity bit-parallel canonical and normal basis multipliers for a class of finite fields*, IEEE Transactions on Computers, Vol. 47, No. 3, pp. 353-356, 1998.

[5] A. Reyhani-Masoleh, and M. A. Hasan, *A new construction of massey-omura parallel multiplier over* $GF(2^m)$, IEEE Transactions on Computers, Vol. 51, No. 5, pp. 511-520, 2002.

[6] T. Beth, and D. Gollman, *Algorithm engineering For public key algorithms*, IEEE J. Selected Areas in Communications, Vol. 7, No. 4, pp. 458-466, 1989.

[7] G. B. Agnew, R. C. Mullin, I. M. Onyszchuk, and S. A. Vanstone, *An implementation for a fast public-key cryptosystem*, Journal of Cryptology, Vol. 3, No. 2, pp. 63-79, 1991.

[8] A. Reyhani-Masoleh and M. A. Hasan, *Efficient Digit-serial Normal Basis Multipliers over Binary Extension Fields*, ACM Transactions on Embedded Computing Systems, Vol. 3, No. 3, pp. 575-592, 2004.

[9] S. Kwon, K. Gaj, C. H. Kim, and C. P. Hong, *Efficient linear array for* m*ultiplication in* $GF(2^m)$ *using a normal basis for elliptic curve cryptography*, Proceedings of Workshop Cryptographic Hardware and Embedded Systems (CHES), pp. 76-91, 2004.

[10] A. H. Namin, H. Wu, and M. Ahmadi, *Comb architectures for finite field multiplication in* F_{2^m}, IEEE Transactions on Computers, Vol. 56, No. 7, pp. 909-916, 2007.

[11] A. H. Namin, H. Wu, and M. Ahmadi, *A word-level finite field multiplier using* n*ormal basis*, IEEE Transactions on Computers, Vol. 60, No. 6, pp. 890-895, 2010.

[12] Y-S. Cho, J-Y. Choi, *A LSB-first word-level normal basis multiplier over* $GF(2^m)$, Journal of The Korea Knowledge Information Technology Society, Vol. 9, No. 5, pp. 571-578, 2014.

[13] S. Lin, and D. Costello, *Error control coding: Fundamentals and applications*, Pearson, Prentice-Hall, 2004, 2nd ed.

[14] E. R. Berlekamp, *Bit-Serial Reed-Solomon encoders*, IEEE Transactions on Information Theory, Vol. 28, No. 6, pp. 869-874, 1982.

[15] J. K. Omura, and J. L. Massey, *Computational method and apparatus for finite field arithmetic*, U.S. Patent #4, 587, 627, 1986.

[16] *National institute of standards and technology, digital signature standard*, FIPS Publications 186-2, 2000.

[17] IEEE Std 1363-2000. *IEEE Standard specifications for public-key cryptography*, 2000.

$GF(2^m)$의 정규기저를 이용한 2배속 비트직렬 곱셈기 설계

조용석, 민경일

유원대학교 정보통신보안학과

요 약

유한체 연산은 최근 관심이 집중되고 있는 분야로, 신호처리와 오류정정부호, 특히 암호이론 등에서 광범위하게 응용되고 있다. 이와 같은 응용 분야에서는 저복잡도의 유한체 연산기를 설계할 필요가 있다. 유한체 연산의 성능에 큰 영향을 미치는 하나의 중요한 요소는 유한체의 원소를 표현하는 방법인 기저이다.

이 중에서 정규기저를 사용하면, 유한체 $GF(2^m)$의 제곱 연산이 왼쪽으로 한 비트 순회치환만으로 가능하기 때문에 연산기의 하드웨어 구현에 매우 유리하다. 본 논문에서는 정규기저를 이용한 새로운 2배속 비트직렬 곱셈기의 구조를 제안한다. 제안한 곱셈기는 곱셈의 한 원소를 2개의 부분으로 나눈 다음, 각각의 부분을 동시에 비트직렬 곱셈기로 구현한다. 따라서 본 곱셈기는 크기가 m인 2진 유한체에서 한 번의 곱셈 연산을 수행하는데 $\lceil m/2 \rceil$ 클럭이 소요된다. 그러므로 제안된 곱셈기는 기존의 비트직렬 곱셈기에 비해 2배 빠르며, 비트병렬 곱셈기에 비해서는 더 낮은 복잡도를 가진다. 본 곱셈기는 기존의 곱셈기에 비해서 구조가 매우 규칙적이므로 VLSI 구현에 적합하다. 제안된 곱셈기의 중요한 장점은 회로의 복잡도와 지연시간 사이에 적절한 절충이 가능하다는 것이다. 따라서 본 곱셈기는 m 값이 크지만 회로의 면적을 고려하여야 하는 자원이 한정된 암호 시스템과 같은 응용에 더 적합하다.

Yong-Suk Cho received the B.S., M.S., and Ph.D. degree in the Department of Electronic Communication Engineering from Hanyang University in 1986, 1988 and 1998, respectively. From 1989 to 1996, he was a researcher at Korea Telecom. He has been a professor in the Department of IT & Securities at U1 University since 1996. His current research interests include finite field arithmetic, cryptography, and error-control coding.

E-mail address: yscho@u1.ac.kr

Kyoung-Il Min received the B.S. degree in the Department of Electronic Engineering from the Ulsan University in 1977. He received the M.S. degree and the Ph.D. degree in the Department of Electronic Engineering from Chungnam University in 1984 and 1995, respectively. He has been a professor in the Department of IT & Securities at U1 University since 1996. His current research interests include logic design and cryptography. (Corresponding author of this paper)

E-mail address: kyilmin@u1.ac.kr