All-One 다항식에 의해 정의된 유한체 $GF(2^m)$ 상의 효율적인 Bit-Parallel 정규기저 곱셈기

장용희[O] 권용진
한국항공대학교 정보통신공학과
{yhjang[O], yjkwon}@mail.hangkong.ac.kr

An Efficient Bit-Parallel Normal Basis Multiplier for $GF(2^m)$ Fields Defined by All-One Polynomials

Yong-Hee Jang[0] Yong-Jin Kwon
Dept. of Inform. & Telecomm. Eng., Hankuk Aviation University

요 약

유한체 $GF(2^m)$ 상의 산술 연산 중 곱셈 연산의 효율적인 구현은 암호이론 분야의 어플리케이션에서 매우 중요하다. 본 논문에서는 All-One 다항식에 의해 정의된 $GF(2^m)$ 상의 효율적인 Bit-Parallel 정규기저 곱셈기를 제안한다. 게이트 및 시간 면에서 본 논문의 곱셈기의 complexity는 이전에 제안된 같은 종류의 곱셈기 보다 낮거나 동일하다. 그리고 본 논문의 곱셈기는 이전 곱셈기 보다 더 모듈적이어서 VLSI 구현에 적합하다.

1. Introduction

유한체 $GF(2^m)$ 상의 산술 연산, 즉 덧셈, 곱셈, 제곱(squaring) 및 곱셈역원 연산 등은 부호이론, 컴퓨터 대수, 암호이론과 같은 분야의 어플리케이션에서 많이 사용된다[7]. 이들 어플리케이션이 효율적으로 구현되기 위해서는 이들 연산에 대해서 면적 및 시간적인 면에서 효율적인 알고리즘과 하드웨어 구조가 필요하다[4]. 덧셈은 매우 간단히 구현되지만, 다른 연산들은 더 복잡하다. 덧셈 연산을 제외한 다른 연산들, 즉 exponentiation, 나눗셈 및 곱셈역원 연산은 곱셈 연산을 반복 계산하여 수행되므로 곱셈 연산을 효율적으로 구현하는 것은 매우 중요하다[8].

지금까지 일반적인 기약 다항식에 의해 정의된 유한체 $GF(2^m)$ 상의 Bit-Parallel 곱셈기가 많이 제안되어 왔다. 그러나 이들 곱셈기는 시스템 복잡도가 커서 암호 분야의 어플리케이션에 비효율적이다[8]. 1989년에 Itoh와 Tsujii[1]는 시스템 복잡도를 감소시키기 위해 차수 m의 기약 All-One Polynomial(AOP)에 의해 정의된 유한체 $GF(2^m)$ 상의 low-complexity Bit-Parallel 곱셈기를 제안했다. 이 이후로 AOP에 의해 정의된 $GF(2^m)$ 상의 Bit-Parallel 곱셈기가 많이 제안되어 왔다(예 [2],[3],[4],[5],[6],[7],[8]).

본 논문에서는 정규기저를 사용하고 AOP에 의해 정의된 $GF(2^m)$ 상의 Bit-Parallel 정규기저 곱셈기를 제안한다. 본 논문의 곱셈기의 AND 게이트 및 XOR 게이트의 complexity는 각각 m^2 및 m^2-1이다. 그리고 시간 complexity는 $T_A+(1+\lceil \log_2(m-1)\rceil)T_X$으로 이전에 제안된 같은 종류의 곱셈기 보다 complexity가 낮거나 같다. 그러나 본 논문에서 제안한 Bit-Parallel 정규기저 곱셈기는 이전에 제안된 곱셈기와는 다른 구조를 가지며 몇몇 다른 장점을 가진다.

2. Preliminaries

2.1 Normal Basis Representation

유한체 $GF(2^m)$의 임의의 원소 A는 $GF(2)$상에서 정규기저(Normal Basis), β^{2^0}, β^{2^1}, $\cdots, \beta^{2^{m-1}}$ ($\beta \in GF(2^m)$)를 사용해서 표현할 수 있다. A와 B를 $GF(2^m)$의 원소라 하고 $(a_0, a_1, \cdots, a_{m-1})$과 $(b_0, b_1, \cdots, b_{m-1})$을 각각 A와 B의 위 정규기저에 관한 좌표하고 하자. 그리고 A와 B의 곱을 C라고 하면 C의 각 좌표 c_i는 아래와 같이 얻어진다.

$$c_i = \underline{a} \times M_i \times \underline{b}^T = \underline{a}^{(i)} \times M_0 \times \underline{b}^{(i)^T}, \quad 0 \le i \le m-1 \tag{1}$$

여기서 $\underline{a}^{(i)}$와 $\underline{b}^{(i)}$은 각각 $\underline{a}=[a_0, a_1, \cdots, a_{m-1}]$와 $\underline{b}=[b_0, b_1, \cdots, b_{m-1}]$의 i-fold left cyclic shift이다. 그리고 T는 vector Transposition이며 M_i는 각 성분이 $GF(2)$에 속 하는 $m \times m$ 행렬이다.

각 M_i에서 1의 개수는 모두 동일하며, 이 1의 개수를 C_N으로 보통 표시한다. M_i에서 1인 요소의 개수가 정규기저 곱셈기의 게이트 수를 결정하기 때문에, C_N을 정규기저의 complexity라 한다.

C_N은 $C_N \ge 2m-1$이라고 증명되어 있으며, $C_N=2m-1$일 때의 정규기저를 최적정규기저(Optimal Normal Basis:ONB)라고 한다. 이 ONB에는 두 종류, 즉 type-I과 type-II가 존재한다.

2.2 $GF(2^m)$ Fields Defined by All-One Polynomials

$GF(2)$ 상에서 다항식 $P(x)=p_0+p_1x+p_2x^2+\cdots+p_{m-1}x^{m-1}+p_mx^m$의 모든 계수가 1이면, 즉 $i=0,1,2,\cdots,m-1,m$에 대해서 $p_i=1$이면, 다항식 $P(x)$를 차수 m의 All One Polynomial(AOP)이라 한다. AOP가 기약 다항식일 필요충분조건은 $m+1$이 소수이고 2가 유한체 $GF(m+1)$의 생성원이어야 한다. 100이하인 정수 m에 대하여, 차수 m의 AOP가 기약이 되는 m의 값은 2, 4, 10, 12, 18, 28, 36, 52, 58, 60, 66, 82, 100이다[8].

$P(x)=1+x+x^2+\cdots+x^{m-1}+x^m$을 차수 m의 기약 AOP라 하고 β를 $P(x)$의 근이라 하자. 그러면 $1+\beta+\beta^2+\cdots+\beta^{m-1}$

* 본 논문은 과학기술부·한국과학재단지정 「한국항공대학교 인터넷정보검색연구센터」의 연구비 및 IDEC의 지원으로 수행되었음.

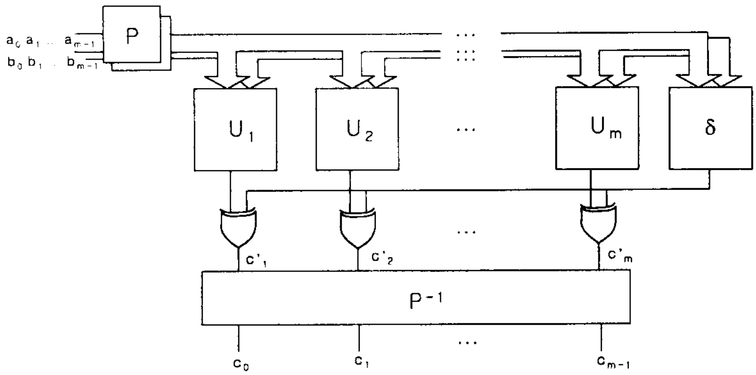

그림 1. 제안한 Bit-Parallel 정규기저 곱셈기

$+\beta^m=0$이므로 $\beta^{m+1}=1$이 된다.

차수 m의 기약 AOP의 한 근을 β라고 하면, $\beta^{2^i}(i=1,\cdots,m-1)$도 또한 근이 된다. 그리고 차수 m의 기약 AOP에 의해 정의된 $GF(2^m)$의 정규기저는 $\beta^{2^i}(i=0,1,\cdots,m-1)$에 의해 형성될 수 있으며, 이 정규기저는 type-I ONB이다. AOP의 차수 m에 대해서, 2는 법 $m+1$에 관해서 원시근이므로 $\{2^0,2^1,\cdots,2^{m-1}\}$은 법 $m+1$에 관한 기약잉여계가 된다. 따라서 기약 AOP에 의해 정의된 type-I ONB $\{\beta,\beta^{2^1},\beta^{2^2},\cdots,\beta^{2^{m-1}}\}$은

$$\{\beta,\beta^2,\beta^3,\cdots,\beta^m\} \tag{2}$$

이 된다. 집합 (2)는 또한 기저이므로 $GF(2^m)$의 임의의 원소를 표현할 수 있다. 정규기저로 표현된 $GF(2^m)$의 원소는 (2)의 기저 표현으로 쉽게 변환된다. $\beta^{m+1}=1$이므로, $GF(2^m)$의 원소 $A=\sum_{i=0}^{m-1}a_i\beta^{2^i}$는 아래와 같이 주어진 치환 P를 사용해서

$$a'_{2^i \bmod m+1}=a_i,\quad i=0,1,\cdots,m-1 \tag{3}$$

다음과 같이 변환된다.

$$A=\sum_{i=0}^{m-1}a_i\beta^{2^i}=\sum_{i=1}^{m}a'_i\beta^i \tag{4}$$

3. An Efficient Multiplier Using Normal Basis

3.1 Formulation of Multiplication

기약 AOP에 의해 정의된 $GF(2^m)$의 임의의 원소는 정규기저 대신에 (3)식의 치환 P를 이용하여 (2)의 기저 표현으로 쉽게 변환된다. 그래서 정규기저로 표현된 $GF(2^m)$의 임의의 원소 A와 B의 곱셈, 즉 $C=AB$를 수행하기 위해, 먼저 (3)식의 치환 P를 사용하여 A와 B를 아래와 같이 변환한 후

$$A=\sum_{i=1}^{m}\beta^i=a'_1\beta+a'_2\beta^2+\cdots+a'_m\beta^m.$$
$$B=\sum_{i=1}^{m}\beta^i=b'_1\beta+b'_2\beta^2+\cdots+b'_m\beta^m. \tag{5}$$

다음과 같이 곱셈을 수행한다.

$$C'=AB=(\underline{A}\times\underline{B}^T)\times(\underline{\beta}\times\underline{B}^T)=\underline{A}\times M'\times\underline{B}^T. \tag{6}$$

여기서 $\underline{A}=[a'_1,a'_2,\cdots,a'_m]$, $\underline{B}=[b'_1,b'_2,\cdots,b'_m]$, $\underline{\beta}=[\beta,\beta^2,\cdots,\beta^m]$이고, M'은 아래와 같이 정의된다.

$$M'=\underline{\beta}^T\times\underline{\beta}=[\beta^{i+j}]$$
$$=\begin{bmatrix}\beta^{1+1} & \beta^{1+2} & \cdots & \beta^{1+m}\\ \beta^{2+1} & \beta^{2+2} & \cdots & \beta^{2+m}\\ \cdot & \cdot & \cdot & \cdot\\ \cdot & \cdot & \cdot & \cdot\\ \cdot & \cdot & \cdot & \cdot\\ \beta^{m+1} & \beta^{m+2} & \cdots & \beta^{m+m}\end{bmatrix} \tag{7}$$

(7)식에서 β는 기약 AOP의 근이므로 $\beta^{m+1}=1$이다. 그래서 M'은 아래와 같이 된다.

$$M'=\begin{bmatrix}\beta^2 & \beta^3 & \beta^4 & \beta^5 & \iddots & \beta^{m-1} & \beta^m & 1\\ \beta^3 & \beta^4 & \beta^5 & \iddots & \iddots & \beta^m & 1 & \beta\\ \beta^4 & \beta^5 & \iddots & \iddots & \iddots & 1 & \beta & \beta^2\\ \beta^5 & \iddots & \iddots & \iddots & \iddots & \beta & \beta^2 & \beta^3\\ \iddots & \iddots & \iddots & \iddots & \iddots & \iddots & \iddots & \iddots\\ \beta^{m-1} & \beta^m & \iddots & \iddots & \iddots & \iddots & \iddots & \iddots\\ \beta^m & 1 & \beta & \beta^2 & \beta^3 & \iddots & \iddots & \beta^{m-2}\\ 1 & \beta & \beta^2 & \beta^3 & \iddots & \iddots & \beta^{m-2} & \beta^{m-1}\end{bmatrix} \tag{8}$$

따라서 C'는 아래와 같이 주어진다.

$$C'=\sum_{i=1}^{m}a'_ib'_{m+1-i}\cdot 1+\sum_{i=1}^{m}\Big(\sum_{j=1}^{m-i}a'_{i+j}b'_{m+1-j}+\sum_{j=1}^{i-1}a'_jb'_{i-j}\Big)\beta^i \tag{9}$$

(9)식의 $\sum_{i=1}^{m}a'_ib'_{m+1-i}\cdot 1$에서 1은 $1=\sum_{i=1}^{m}\beta^i$이다. 그래서 $\sum_{i=1}^{m}a'_ib'_{m+1-i}$을 δ로 표시하면 C'는 아래와 같이 다시 쓸 수 있다.

$$C'=\sum_{i=1}^{m}\Big(\delta+\sum_{j=1}^{m-i}a'_{i+j}b'_{m+1-j}+\sum_{j=1}^{i-1}a'_jb'_{i-j}\Big)\beta^i. \tag{10}$$

따라서 C'의 각 좌표 c'_i는

$$c'_i=\delta+\sum_{j=1}^{m-i}a'_{i+j}b'_{m+1-j}+\sum_{j=1}^{i-1}a'_jb'_{i-j},\quad 1\le i\le m \tag{11}$$

표 1. 기약 AOP에 의해 정의된 $GF(2^m)$의 Bit-Parallel 곱셈기의 비교

곱셈기	기저	#AND	#XOR	시간 지연
Itoh와 Tsujii[1]	Polynomial	$(m+1)^2$	m^2+2m	$T_A+(\lceil \log_2 m \rceil + \lceil \log_2(m+2) \rceil)T_X$
Hasan 등[2]	Polynomial	m^2	m^2+m-2	$T_A+(m+\lceil \log_2(m-1) \rceil)T_X$
Koc과 Sunar[4]	Polynomial	m^2	m^2-1	$T_A+(2+\lceil \log_2(m-1) \rceil)T_X$
Wu-Hasan[6]	Weakly dual	m^2	m^2-1	$T_A+(1+\lceil \log_2(m-1) \rceil)T_X$
Wang 등[5]	Normal	m^2	$2m^2-2m$	$T_A+(1+\lceil \log_2(m-1) \rceil)T_X$
Koc와 Sunar[4]	Normal	m^2	m^2-1	$T_A+(2+\lceil \log_2(m-1) \rceil)T_X$
Hasan 등[3]	Normal	m^2	m^2-1	$T_A+(1+\lceil \log_2(m-1) \rceil)T_X$
Reyhani-Masoleh등[7]	Normal	m^2	m^2-1	$T_A+(1+\lceil \log_2(m-1) \rceil)T_X$
본 논문	Normal	m^2	m^2-1	$T_A+(1+\lceil \log_2(m-1) \rceil)T_X$

이 된다. (10)식과 같이 주어진 C'에 P의 역치환을 수행하면 원래의 정규기저로 표현된 C를 얻게 된다.

3.2 Architecture

3.1절의 (11)식을 사용해서 얻어진 병렬 정규기저 곱셈기의 구조를 그림 1에 나타낸다. 이 구조에서 P는 (3)식에 나타낸 치환을 나타낸다. 즉, 이 치환을 통해 정규기저로 표현된 입력 A와 B는 (2)의 기저 표현으로 변환된다. 그리고 각 $U_i(i=1,2,\cdots,m)$는 $\sum_{j=1}^{m-i} a'_{i+j}b'_{m+1-j} + \sum_{j=1}^{i-1} a'_j b'_{i-j}$을 생성하고, δ는 $\sum_{i=1}^{m} a'_i b'_{m+1-i}$을 생성한다. (11)식에 의해 각 U_i의 출력 값은 δ의 출력 값과 더해져서 P의 역치환 P^{-1} 블록에 입력된다. 이 블록을 통해 원래의 정규기저 표현으로 변환되게 된다.

4. Gate and Time Complexities

그림 1의 구조에서 치환 P와 역치환 P^{-1}은 단순히 wiring에 의해 구현되므로 어떠한 게이트 자원도 필요로 하지 않는다. 따라서 그림 1 곱셈기의 전체 AND 게이트 및 XOR 게이트의 개수는 아래와 같다.

· 전체 AND 게이트 개수 = (각 U_i의 AND 게이트 개수) × (U_i의 개수) + (δ의 AND 게이트 개수)
$= (m-1)\times m+m = m^2$

· 전체 XOR 게이트 개수 = (각 U_i의 XOR 게이트 개수) × (U_i의 개수) + (δ의 XOR 게이트 개수) + m
$= (m-2)m+m-1+m = m^2-1$

그 다음으로 그림 1의 곱셈기의 시간 지연(time delay)을 조사해 보자. 그림 1의 곱셈기에서 가장 긴 입력에서 출력까지 경로는 아래와 같다.

· 입력에서 출력까지의 경로 = $P \rightarrow \delta$블록 $\rightarrow$ XOR 게이트 $\rightarrow P^{-1}$

위 경로에서 P와 P^{-1}은 wiring에 의해 구현되므로 시간 지연에 포함되지 않는다. 그리고 δ 블록의 시간 지연은 $T_A+\lceil \log_2 m \rceil T_X$이다(여기서 $T_A=$ AND 게이트의 시간 지연, $T_X=$ XOR 게이트의 시간 지연이다). 그리고 m이 짝수이면 $\lceil \log_2 m \rceil = \lceil \log_2 m-1 \rceil$이다. 본 논문에서 m은 항상 짝수이므로, δ블록의 시간 지연은 $T_A+\lceil \log_2(m-1) \rceil T_X$이 된다. 따라서 그림 1의 곱셈기의 전체 시간 지연 $T_A+(1+\lceil \log_2(m-1) \rceil)T_X$이다. 위에서 계산된 그림 1의 병렬 곱셈기의 게이트 개수와 시간 지연에 대해서 기약 AOP에 의해 생성된 같은 종류의 다른 병렬 곱셈기와 비교한 결과를 표 1에 나타낸다. 표 1의 결과처럼 본 논문의 곱셈기는 면적 및 시간적인 면에서 가장 효율적인 아키텍처 속함을 알 수 있다. 그리고 본 곱셈기는 이전 곱셈기 보다 더 모듈적이어서 VLSI 구현에 적합하다.

5. Conclusions

본 논문에서는 AOP에 의해 정의된 $GF(2^m)$ 상의 Bit-Parallel 정규기저 곱셈기를 제안하였다. 본 곱셈기의 complexity는 이전에 발표된 같은 종류의 곱셈기 보다 낮거나 동일하다. 그리고 이전 곱셈기와는 다른 구조를 가지며 VLSI 구현에 적합하다.

참고문헌

[1] T. Itoh and S. Tsujii, "Structure of Parallel Multiplier for a Class of Fields $GF(2^m)$," *Information and Computation*, vol. 83, pp. 21-40, 1989.
[2] 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. Computers*, vol. 41, no. 8, pp. 962-971, Aug. 1992.
[3] M.A. Hasan, M.Z. Wang, and V.K. Bhargava, "A Modified Massey-Omura Parallel Multiplier for a Class of Finite Fields," *IEEE Trans. Computers*, vol. 42, no. 10, pp. 1278-1280, Oct. 1993.
[4] C.K. Koc and B. Sunar, "Low-Complexity Bit-Parallel Canonical and Normal Basis Multipliers for a Class of Finites Fields," *IEEE Trans. Computers*, vol. 47, no. 3, pp.353-356, Mar. 1998.
[5] C.C. Wang, T.K. Truong, H.M. Shar, L.J. Deutsch, J.K. Omura, and I.S. Reed, "VLSI Architecture for Computing Multiplications and Inverses in $GF(2^m)$," *IEEE Trans. Computers*, vol. 34, no. 8, pp. 709-716, Aug. 1985.
[6] H. Wu and M.A. Hasan, "Low Complexity Bit-Parallel Multipliers for a Class of Finite Fields," *IEEE Trans. Computers*, vol. 47, no. 8, pp. 883-887, Aug. 1998.
[7] A. Reyhani-Masoleh and M.A. Hasan, "A New Construction fo Massey-Omura Parallel Multiplier over $GF(2^m)$," *IEEE Trans. Computers*, vol. 51, no. 5, pp. 511-520, May 2002.
[8] C.Y. Lee, E.H. Lu, and J.Y. Lee, "Bit-Parallel Systolic Multipliers for $GF(2^m)$ Fields Defined by All-One and Equally Spaced Polynomials," *IEEE Trans. Computers*, vol. 50, no. 5, pp. 385-393, May 2001.