Journal of The Korea Society of Computer and Information
Vol. 20, No. 2, February 2015
www.ksci.re.kr
http://dx.doi.org/10.9708/jksci.2015.20.2.001

유한체 GF(3^m)상의 고속 병렬 곱셈기의 설계

성 현 경*

Design of High-Speed Parallel Multiplier on Finite Fields GF(3^m)

Hyeon-Kyeong Seong*

요 약

본 논문에서는 유한체 $GF(3^m)$상에서 모든 항에 0이 아닌 계수를 갖는 기약 다항식에 대하여 m이 홀수 및 짝수인 경우 $GF(3^m)$상의 곱셈 알고리즘을 제시하였으며, 제시한 곱셈 알고리즘을 이용하여 고속의 병렬 입-출력 모듈구조의 곱셈기를 설계하였다. 제시한 곱셈기의 구성은 $(m+1)^2$개의 동일한 기본 셀들로 설계되었으며, 셀에 메모리를 사용하지 않았으므로 회로가 간단하며 셀당 $T_A + T_X$의 지연시간을 갖는다. 본 논문에서 제안한 곱셈기는 규칙성과 셀 배열에 의한 모듈성을 가지므로 m이 큰 회로의 확장이 용이하며 VLSI회로 실현에 적합할 것이다.

▸ Keywords : 유한체, 곱셈기, 기약다항식, 다항식, 갈로아필드

Abstract

In this paper, we propose a new multiplication algorithm for primitive polynomial with all 1 of coefficient in case that m is odd and even on finite fields $GF(3^m)$, and design the multiplier with parallel input-output module structure using the presented multiplication algorithm. The proposed multiplier is designed $(m+1)^2$ same basic cells. Since the basic cells have no a latch circuit, the multiplicative circuit is very simple and is short the delay time $T_A + T_X$ per cell unit. The proposed multiplier is easy to extend the circuit with large m having regularity and modularity by cell array, and is suitable to the implementation of VLSI circuit.

▸ Keywords : Finite fields, Multiplier, Irreducible polynomial, Polynomial, GF

•제1저자 : 성현경
•투고일 : 2014. 12. 30, 심사일 : 2015. 1. 10, 게재확정일 : 2015. 1. 21.
* 상지대학교 컴퓨터공학부(School of Computer Information Communication Eng., Sangji University)
※ 이 논문은 2013년도 상지대학교 교내연구비 지원에 의해 연구되었음.

I. 서 론

유한체(Galois field)의 연산은 디지털 신호처리, Reed-solomon 부호기, 영상처리, 오류정정부호, 디지털 통신의 암호화 및 해독화를 요하는 보안 등에 다양한 응용 분야에서 중요한 역할을 한다[1-3]. 이들 중 오류정정부호의 경우 유한체 $GF(2^m)$ 상의 연산에서 실제로 부호기 및 복호기 설계 시 전체 시스템의 규모와 성능에 절대적인 영향을 미치므로 회로 경로의 연결, 시스템 구조의 복잡성과 동시성 등의 문제점을 개선하기 위한 연구가 진행되어 왔다[4]. 특히, Galois field (GF) 연산은 Reed-Solomon 채널코딩과 디코딩 구조에 일반적으로 사용된다. Reed-Solomon(RS) 코드는 무선통신 채널에 대하여 오류검출과 정정을 제공한다. 예를 들면, 3GPP/EDGE/E-TCH 블록 부호화/복호화는 보통 $GF(2^8)$으로 구현된다[5]. 그러나 RS 부호기와 복호기는 여러 가지 유한체 승산과 가산을 요하며, 유한체 연산에서 가산은 간단하게 수행되는 반면에 승산은 상당한 계산량을 요구한다. 그러므로 승산에 대한 효과적인 구현을 갖는 것이 중요하게 되었다.

Koc와 Sunar[6]은 AOP를 기반으로 하는 저복잡성 비트-병렬 정규기저 곱셈기를 제안하였다. 제안된 정규기저 곱셈기는 m^2-1개의 XOR 게이트와 m^2개의 AND 게이트를 필요로 한다. Halbutogullari와 Koc[7]는 일반적인 기약다항식에 대한 Mastrovito 곱셈기를 제안하였다. 위에서 제안된 저복잡성 곱셈기들이 보안 및 암호 시스템 응용에 적합하다 할지라도 시스토릭 기술을 이용하여 설계된 것이 아닌 경우에는 m이 클 경우 $GF(2^m)$ 상의 곱셈에 대한 지연시간은 매우 크다.

Lee 등[8]은 유한체 $GF(2^m)$ 상에서 기약 AOP (all-one-polynomial)를 기반으로 하는 순환이동과 내적이라는 두 연산을 이용한 곱셈 알고리즘을 제안하였고, 제안한 알고리즘을 기반으로 저복잡성 비트-병렬 시스토릭 곱셈기를 구성하였다. 첫 번째 곱셈기는 1개의 2 입력 AND 게이트와 1개의 2 입력 XOR 게이트, 3개의 1 비트 래치로 이루어진 $(m+1)^2$개의 동일한 셀들로 구성하였고, 또 하나의 곱셈기는 $(m+1)^2$개의 동일한 셀들과 m개의 XOR 게이트로 구성하였는데, 각 셀은 1개의 2 입력 AND 게이트와 1개의 2 입력 XOR 게이트, 4개의 1 비트 래치로 구성하였다. 각각의 곱셈기는 각 셀에서의 지연시간이 짧기 때문에 속도가 빠르다. Kim 등[9]은 기약 AOP를 이용한 비트-병렬 시스토릭 곱셈기를 제안하였으며, 제안한 곱셈기는 공간 복잡도가 증가하는 반면에 시간 복잡도가 감소하며, VLSI 구현에 적합하다. Kim과 Jeon[10]은 유한체 상에서 다항식 기저의 세미-시스토릭 어레이를 기반으로 한 효율적인 몽고메리 곱셈기를 제안하였다. 제안한 곱셈기는 입력 비트수가 홀수인지 또는 짝수인지에 따라 다르게 수행하며 전체 시간 복잡도를 감소시킨다. Chang 등[11]은 삼항 기약다항식 기반의 $GF(3^m)$ 디지트-직렬 곱셈기를 제안하였으며, 이 곱셈기는 모듈러 감산 연산부를 병렬화하여 공간 복잡도는 기존의 결과와 같으나 시간 복잡도가 감소한다.

본 논문에서는 Chang 등[11]이 제안한 $GF(3^m)$의 디지트-직렬 곱셈기와 Lee 등[8]이 제시한 AOP를 기반으로 하는 유한체 $GF(2^m)$ 상에서의 곱셈 알고리즘을 $GF(3^m)$ 상으로 확장하여 모든 항에 0이 아닌 계수가 존재하는 기약다항식의 두 원소에 대한 곱셈 알고리즘을 제안하였다. $GF(3^m)$ 상에서는 m이 홀수인 경우와 짝수인 경우에 대한 조건을 정리를 통하여 증명한 후 알고리즘을 구현하고, $GF(3^4)$ 상에서 곱셈기를 설계하였다.

II. $GF(3^m)$상에서의 곱셈 알고리즘

본 장에서는 유한체 $GF(3)$의 연산인 가산과 곱셈에 대하여 논하고, $GF(3^m)$의 곱셈 알고리즘을 제시한다.

2.1 GF(3^m)상에서의 곱셈 알고리즘

본 절에서는 유한체 $GF(3^m)$ 상에서 모든 항에 0이 아닌 계수가 존재하는 기약다항식의 두 원소에 대하여 m이 홀수인 경우와 짝수인 경우에 대한 곱셈 알고리즘을 제시하였다.

2.1.1 GF(3^m)상에서 m이 홀수인 곱셈 알고리즘

유한체 $GF(3^m)$ 상에서 m이 홀수인 경우에 대한 곱셈 알고리즘을 제시하며, $GF(3^m)$은 m이 양의 정수인 3^m개의 원소를 갖는다. 유한체 $GF(3^m)$ 상에서 모든 항이 존재하는 기약다항식이 식 (1)과 같이 표현될 때, $GF(3^m)$ 상에서 m이 홀수인 경우에 대한 곱셈 알고리즘은 다음과 같다.

$$F(x) = f_0 + f_1 x + f_2 x^2 + \cdots + f_{m-1} x^{m-1} + 2x^m \quad (1)$$

여기서, $f_i = \begin{cases} 1 & i\text{는 짝수} \\ 2 & i\text{는 홀수} \end{cases}$ 이며, $f_i \in GF(3)$,

$0 \le i \le m-1$이다.

$F(x)=0$이므로 식 (2)와 같이 표현되며 식 (1)에서 최고차 항의 계수 2는 $GF(3^m)$상에서 식 (3)를 성립시키기 위한 계수이다.

$$F(x) = f_0 + f_1 x + f_2 x^2 + \cdots + f_{m-1}x^{m-1} + 2x^m = 0 \tag{2}$$

식 (2)를 최고차 항으로 정리하면 식 (3)과 같다.

$$-2x^m = f_0 + f_1 x + f_2 x^2 + \cdots + f_{m-1}x^{m-1} \tag{3}$$

식 (3) 좌변의 $-2x^m$에서 계수 -2는 유한체 성질에 의해 1과 같으므로 식 (3)을 식(4)와 나타낼 수 있다.

$$x^m = f_0 + f_1 x + f_2 x^2 + \cdots + f_{m-1}x^{m-1} \tag{4}$$

여기서 $f_i \in GF(3)$이다.

두 다항식을 곱셈하였을 때 x^m보다 큰 차수들에 대한 연산은 다음과 같이 수행되며, 먼저 x^{m+1}에 대한 식을 구하면 식 (5)와 같다.

$$\begin{aligned} x^{m+1} &= x^m \cdot x \\ &= f_0 x + f_1 x^2 + f_2 x^3 + \cdots + f_{m-2}x^{m-1} + f_{m-1}x^m \end{aligned} \tag{5}$$

식 (5)에 식 (4)를 대입하면 식 (6)과 같다.

$$\begin{aligned} x^{m+1} = {} & f_0 x + f_1 x^2 + f_2 x^3 + \cdots + f_{m-2}x^{m-1} \\ & + f_{m-1}(f_0 + f_1 x + f_2 x^2 + \cdots + f_{m-1}x^{m-1}) \end{aligned} \tag{6}$$

식 (6)을 정리하면 식 (7)과 같다.

$$\begin{aligned} x^{m+1} = {} & f_0 f_{m-1} + (f_0 + f_1 f_{m-1})x + (f_1 + f_2 f_{m-1})x^2 \\ & + \cdots + (f_{m-2} + f_{m-1} f_{m-1})x^{m-1} \end{aligned} \tag{7}$$

식 (7)에서 $x^{m+1}=1$ 이 되기 위해서는 식 (8)이 성립하여야 한다.

$$\begin{aligned} x^{m+1} = {} & f_0 f_{m-1} + (f_0 + f_1 f_{m-1})x + (f_1 + f_2 f_{m-1})x^2 \\ & + \cdots + (f_{m-2} + f_{m-1} f_{m-1})x^{m-1} = 1 \end{aligned} \tag{8}$$

따라서 식 (8)을 만족하기 위한 각 항의 계수들을 나타내면 식(9)과 같이 표현할 수 있다.

$$f_0 f_{m-1} = 1 \tag{9a}$$

$$f_0 + f_1 f_{m-1} = 0 \tag{9b}$$

$$f_1 + f_2 f_{m-1} = 0 \tag{9c}$$

$$\vdots$$

$$f_{m-2} + f_{m-1} f_{m-1} = 0 \tag{9d}$$

식 (9)이 성립되기 위한 $f_i \in GF(3)$인 f_0부터 f_{m-1}까지의 계수들을 구하는 과정은 다음과 같다.

〔단계 1〕 식 (9a)에서 $f_0 f_{m-1}=1$이 만족되기 위해서는 $f_0=1$, $f_{m-1}=1$이다.

〔단계 2〕 식 (9b)에서 $f_0 + f_1 f_{m-1}=0$이 되기 위해서 단계 1에서 구한 $f_0=1$, $f_{m-1}=1$ 을 대입하면 $f_1=2$이다.

〔단계 3〕 식 (9c)에서 $f_1 + f_2 f_{m-1}=0$이 되기 위해서 단계 2에서 구한 $f_1=2$를 대입하면 $f_2=1$이다.

〔단계 4〕 식 (9d)에서 $f_{m-2} + f_{m-1} f_{m-1}=0$이 되기 위해서 단계 1에서 구한 $f_{m-1}=1$을 대입하면 $f_{m-2}=2$이다.

그러므로 위의 과정에서 구한 f_0 ~ f_{m-1}의 값을 정리하면 식 (10)과 같다.

$$f_0 = 1, f_1 = 2, f_2 = 1, f_3 = 2, \cdots, f_{m-2} = 2, f_{m-1} = 1 \tag{10}$$

따라서 x^{m+1}의 식 (7)은 상수 항 $f_0 f_{m-1}$만 1이고, 나머지 x항의 계수들은 모두 0이 되어 x^{m+1}은 식 (11)과 같이 된다.

$$x^{m+1} = x^m \bullet x = 1 \tag{11}$$

식 (11)을 이용하여 x^{m+2}, x^{m+3}, $\cdots$, x^{m+i}, $\cdots$, x^{2m-2}를 구하면 다음의 결과를 얻을 수 있다.

$$x^{m+2} = x^{m+1} \bullet x = x \tag{12a}$$
$$x^{m+3} = x^{m+2} \bullet x = x^2 \tag{12b}$$
$$\vdots$$
$$x^{m+i} = x^{m+i-1} \bullet x = x^{i-1} \tag{12c}$$
$$\vdots$$
$$x^{2m-2} = x^{m+m-2} \bullet x = x^{m-1} \tag{12d}$$

x가 유한체 $GF(3^m)$상에서 m차 기약 다항식의 근이라 할 때, $GF(3^m)$상의 두 원소인 곱셈 다항식 $A(x)$와 피곱셈 다항식 $B(x)$는 식 (13)과 같이 표현된다.

$$A(x) = \sum_{i=0}^{m-1} a_i x^i = a_0 + a_1 x + a_2 x^2 + \cdots + a_{m-1} x^{m-1}$$
$$B(x) = \sum_{i=0}^{m-1} b_i x^i = b_0 + b_1 x + b_2 x^2 + \cdots + b_{m-1} x^{m-1} \tag{13}$$

여기서, $a_i, b_i \in GF(3)$이며, $0 \le i \le m$이다.

곱셈 알고리즘을 유도하기 위하여 다항식 $A(x)$, $B(x)$를 곱셈하면 식 (14)와 같다.

$$\begin{aligned} A(x) \bullet B(x) &= (a_0 + a_1 x + a_2 x^2 + \cdots + a_{m-1} x^{m-1}) \\ &\quad \bullet (b_0 + b_1 x + b_2 x^2 + \cdots + b_{m-1} x^{m-1}) \\ &= \left(\sum_{i=0}^{m-1} a_i x^i\right) \bullet \left(\sum_{i=0}^{m-1} b_i x^i\right) \end{aligned} \tag{14}$$

두 다항식의 곱셈식인 (14)를 $D(x)$로 놓으면, 식 (15)와 같이 표현할 수 있다.

$$\begin{aligned} D(x) &= d_0 + d_1 x + d_2 x^2 + \cdots + d_{2m-2} x^{2m-2} \\ &= \sum_{i=0}^{2m-2} d_i x^i \end{aligned} \tag{15}$$

식 (15)를 m차 항을 기준으로 2개의 항으로 나누어 정리하면 식 (16)과 같다.

$$D(x) = \left(\sum_{i=0}^{m-1} d_i x^i\right) + \left(\sum_{i=m}^{2m-2} d_i x^i\right) \tag{16}$$

식 (16)의 두 번째 항 $\sum_{i=m}^{2m-2} d_i x^i$는 식 (13)과 식 (14)을 이용하여 $\sum_{i=0}^{m-2} d_{m+i} x^i$로 표현할 수 있으며, 식 (16)은 식 (17)과 같이 쓸 수 있다.

$$D(x) = \left(\sum_{i=0}^{m-1} d_i x^i\right) + \left(\sum_{i=0}^{m-2} d_{m+i} x^i\right) \tag{17}$$

식 (17)에서 x^{m-1}항을 따로 빼서 정리하면 식 (18)과 같이 쓸 수 있다.

$$\begin{aligned} D(x) &= \sum_{i=0}^{m-2} d_i x^i + d_{m-1} x^{m-1} + \sum_{i=0}^{m-2} d_{m+i} x^i \\ &= \sum_{i=0}^{m-2} (d_i + d_{m+i}) x^i + d_{m-1} x^{m-1} \end{aligned} \tag{18}$$

식 (18)에서 $d_i + d_{m+i} = D_i$, $d_{m-1} = D_{m-1}$이라 놓으면 식 (19)와 같이 표현된다.

$$D(x) = \sum_{i=0}^{m-2} D_i x^i + D_{m-1} x^{m-1} = \sum_{i=0}^{m-1} D_i x^i \tag{19}$$

2.1.2 GF(3^m)상에서 m이 짝수인 곱셈 알고리즘

앞 절에서는 유한체 $GF(3^m)$상에서 m이 홀수일 경우에 대하여 모든 항의 계수가 0이 아닌 기약다항식에 대한 원소인 두 다항식의 곱셈 알고리즘을 제시하였다. 이절에서는 $GF(3^m)$에서 m이 짝수일 경우에 대한 곱셈 알고리즘을 제시한다.

유한체 $GF(3^m)$상에서 모든 항이 존재하는 기약다항식이 식 (1)과 같이 표현될 때, $GF(3^m)$상에서 m이 짝수인 경우에 대한 곱셈 알고리즘이 성립한다.

$$F(x) = f_0 + f_1 x + f_2 x^2 + \cdots + f_{m-1} x^{m-1} + 2x^m \tag{20}$$

여기서, $f_i = \begin{cases} 1 & i\text{는 홀수} \\ 2 & i\text{는 짝수} \end{cases}$ 이며, $f_i \in GF(3)$, $0 \le i \le m-1$이다.

두 다항식을 곱셈하였을 때, x^m보다 큰 차수들에 대하여 알아보기 위하여 먼저 x^{m+1}에 대한 식을 구하면 식 (6)과 식 (7)와 같다. 식 (7)을 다시 쓰면 식 (21)과 같다.

$$x^{m+1} = f_0 f_{m-1} + (f_0 + f_1 f_{m-1})x + (f_1 + f_2 f_{m-1})x^2 + \cdots + (f_{m-2} + f_{m-1} f_{m-1})x^{m-1} \tag{21}$$

여기서, m이 짝수인 경우에 성립할 수 있는 조건을 구하기 위하여 $x^{m+1} = 2$로 놓으면, 식 (21)은 식 (22)과 같이 표현된다.

$$x^{m+1} = f_0 f_{m-1} + (f_0 + f_1 f_{m-1})x + (f_1 + f_2 f_{m-1})x^2 + \cdots + (f_{m-2} + f_{m-1} f_{m-1})x^{m-1} = 2 \tag{22}$$

따라서 식 (22)을 만족하기 위하여 각 항의 계수들은 식 (23)과 같이 계산된다.

$$f_0 f_{m-1} = 2 \tag{23a}$$
$$f_0 + f_1 f_{m-1} = 0 \tag{23b}$$
$$f_1 + f_2 f_{m-1} = 0 \tag{23c}$$
$$\vdots$$
$$f_{m-2} + f_{m-1} f_{m-1} = 0 \tag{23d}$$

식 (23)이 성립되기 위한 $f_i \in GF(3)$인 f_0부터 f_{m-1}까지의 계수들을 구하는 과정은 다음과 같다.

〔단계 1〕 식 (23a)에서 $f_0 f_{m-1} = 2$가 되기 위해서 $f_0 = 2$, $f_{m-1} = 1$이다.

〔단계 2〕 식 (23b)에서 $f_0 + f_1 f_{m-1} = 0$이 되기 위해서 단계 1에서 구한 $f_0 = 2$, $f_{m-1} = 1$ 을 대입하면 $f_1 = 1$이다.

〔단계 3〕 식 (23c)에서 $f_1 + f_2 f_{m-1} = 0$이 되기 위해서 단계 2에서 구한 $f_1 = 1$를 대입하면 $f_2 = 2$이다.

〔단계 4〕 같은 방식으로 대입하여 구하면 $f_{m-2} = 2$이다.

그러므로 위의 과정에 의해 구한 f_0~f_{m-1}의 값을 정리하면 식 (24)와 같다.

$$f_0 = 2, f_1 = 1, f_2 = 2, f_3 = 1, \cdots, f_{m-2} = 2, f_{m-1} = 1 \tag{24}$$

따라서 x^{m+1}의 식 (22)는 상수 항 $f_0 f_{m-1}$만 2이고, 나머지 x항의 계수들은 모두 0이 되어 x^{m+1}은 식 (25)와 같이 된다.

$$x^{m+1} = x^m \bullet x = 2 \tag{25}$$

식 (25)을 이용하여 x^{m+2}, x^{m+3}, $\cdots, x^{m+i}$, $\cdots$, x^{2m-2}를 구하면 다음과 같이 $GF(3^m)$상에서 m이 짝수인 경우는 홀수인 경우의 우변에 모두 2가 곱해진다.

$$x^{m+2} = x^{m+1} \bullet x = 2x \tag{26a}$$
$$x^{m+3} = x^{m+2} \bullet x = 2x^2 \tag{26b}$$
$$\vdots$$
$$x^{m+i} = x^{m+i-1} \bullet x = 2x^{i-1} \tag{26c}$$
$$\vdots$$
$$x^{2m-2} = x^{m+m-2} \bullet x = 2x^{m-1} \tag{26d}$$

두 다항식 식 (13)의 곱셈식인 식 (16)의 두 번째 항 $\sum_{i=m}^{2m-2} d_i x^i$는 식 (35)와 식(26)을 이용하여 $\sum_{i=0}^{m-2} 2d_{m+i} x^i$로 표현할 수 있으며 식 (27)과 같이 나타낼 수 있다.

$$D(x) = \left(\sum_{i=0}^{m-1} d_i x^i\right) + \left(\sum_{i=0}^{m-2} 2\, d_{m+i} x^i\right) \tag{27}$$

식 (27)에서 x^{m-1} 항을 따로 빼서 다시 정리하면 식 (28)과 같이 쓸 수 있다.

$$D(x) = \sum_{i=0}^{m-2} d_i x^i + d_{m-1} x^{m-1} + \sum_{i=0}^{m-2} 2d_{m+i} x^i$$
$$= \sum_{i=0}^{m-2} (d_i + 2d_{m+i}) x^i + d_{m-1} x^{m-1} \quad (28)$$

식 (28)에서 $d_i + 2d_{m+i} = D_i$, $d_{m-1} = D_{m-1}$이라 놓으면 식 (29)와 같이 $GF(3^m)$상에서 m이 홀수인 경우의 곱셈 알고리즘의 곱셈식과 같은 형태로 표현된다.

$$D(x) = \sum_{i=0}^{m-2} D_i x^i + D_{m-1} x^{m-1} = \sum_{i=0}^{m-1} D_i x^i \quad (29)$$

III. $GF(3^m)$상의 곱셈기 설계

이 장에서는 앞장에서 제시한 $GF(3^m)$상의 곱셈 알고리즘을 이용하여 m이 홀수인 경우와 짝수인 경우에 대한 $GF(3^m)$상의 곱셈기를 구성한다.

3.1 GF(3^m)상에서 m이 홀수인 곱셈기

$GF(3^m)$상에서 $m=5$인 경우의 곱셈 다항식 $A(x)$와 피곱셈 다항식 $B(x)$은 식 (30)과 같이 표현된다..

$$A(x) = \sum_{i=0}^{m-1} a_i x^i = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4$$
$$B(x) = \sum_{i=0}^{m-1} b_i x^i = b_0 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4 \quad (30)$$

두 다항식 $A(x)$, $B(x)$를 곱셈하면 식 (31)과 같다.

$$A(x) \bullet B(x) = \left(\sum_{i=0}^{m-1} a_i x^i\right) \bullet \left(\sum_{i=0}^{m-1} b_i x^i\right)$$
$$= (a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4)$$
$$\bullet\ (b_0 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4) \quad (31)$$

여기서, $a_i, b_i \in GF(3)$이다.

두 다항식 $A(x)$, $B(x)$의 곱셈한 식 (31)의 결과는 식 (32)와 같다.

$$D(x) = \sum_{i=0}^{m-1} D_i x^i \quad (32)$$
$$= D_0 + D_1 x + D_2 x^2 + D_3 x^3 + D_4 x^4$$

식 (32)에서 계수항 만을 정리하면 식 (33)과 같다.

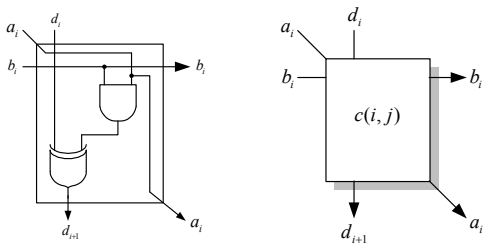
$$\begin{aligned} D_0 &= d_0 + d_5 = a_0 b_0 + a_4 b_1 + a_3 b_2 + a_2 b_3 + a_1 b_4 \\ D_1 &= d_1 + d_6 = a_1 b_0 + a_0 b_1 + a_4 b_2 + a_3 b_3 + a_2 b_4 \\ D_2 &= d_2 + d_7 = a_2 b_0 + a_1 b_1 + a_0 b_2 + a_4 b_3 + a_3 b_4 \\ D_3 &= d_3 + d_8 = a_3 b_0 + a_2 b_1 + a_1 b_2 + a_0 b_3 + a_4 b_4 \\ D_4 &= d_4 = a_4 b_0 + a_3 b_1 + a_2 b_2 + a_1 b_3 + a_0 b_4 \end{aligned} \quad (33)$$

$GF(3^m)$상에서 $m=5$인 홀수의 두 다항식의 곱셈연산 결과는 식 (33)과 같으며, 식 (33)을 수행하는 곱셈기를 설계하면 다음과 같다.

$GF(3^m)$상에서 $m=5$ 홀수일 경우의 곱셈기를 구성하기 위해서 1개의 2입력 mod(3) 덧셈 게이트와 1개의 2입력 mod(3) 곱셈 게이트를 이용하여 기본 셀을 구성하였다. 기본 셀의 회로와 기호를 그림 1에서 보였다.

그림 1의 셀의 회로는 식 (34)를 수행하며, a_i와 b_i는 각각 곱셈 다항식 $A(x)$와 피곱셈 다항식 $B(x)$의 계수들을 의미한다.

$$d_{i+1} = d_i \oplus (a_i \bullet b_i) \bmod (3) \quad (34)$$



(a) (b)

그림 1. $GF(3^5)$상에서 m이 홀수인 곱셈기의 기본 셀, (a) 회로 (b) 기호

Fig. 1. The basic cell of multiplier with odd m on $GF(3^5)$, (a) circuit (b) symbol

그림 1에서 d_i는 셀의 입력으로서 앞단 셀의 출력이며 d_{i+1}은 셀의 출력을 의미한다. 그림 2는 기본 셀들을 이용하

여 $GF(3^5)$상의 곱셈기를 구성한 회로이다. 그림 2에서 a_0 ~ a_4는 곱셈 다항식 $A(x)$에 대한 각 항의 계수들을 의미하며, b_0 ~ b_4는 피곱셈 다항식 $B(x)$에 대한 각 항의 계수들을 의미한다. 최하단의 D_0 ~ D_4는 곱셈결과에 대한 각 항의 계수이다. 최상단에 위치한 셀들에 입력되는 0은 셀에 대한 초기 값이다.

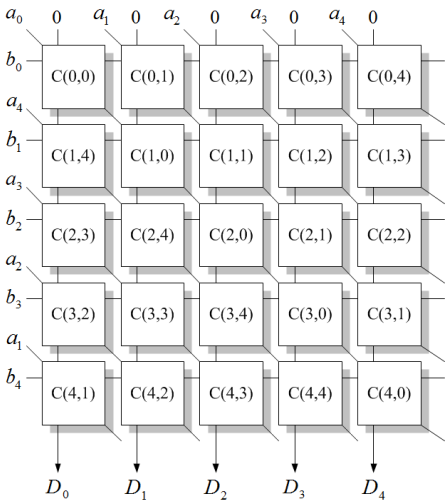


그림 2. $GF(3^5)$상의 제안된 곱셈기
Fig. 2. The proposed multiplier on $GF(3^5)$

3.2 GF(3^m)상에서 m이 짝수인 곱셈기

$GF(3^m)$상에서 $m=4$인 경우의 곱셈 다항식 $A(x)$와 피곱셈 다항식 $B(x)$가 식 (35)와 같이 표현될 때, $GF(3^4)$상에서 두 다항식 $A(x)$와 $B(x)$를 곱셈하면 식 (36)과 같다.

$$A(x) = \sum_{i=0}^{m-1} a_i x^i = a_0 + a_1 x + a_2 x^2 + a_3 x^3$$
$$B(x) = \sum_{i=0}^{m-1} b_i x^i = b_0 + b_1 x + b_2 x^2 + b_3 x^3 \qquad (35)$$

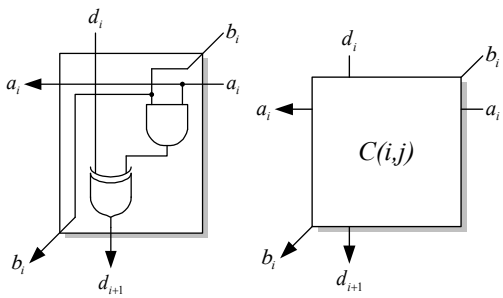
여기서, $a_i, b_i \in GF(3)$이며, $0 \le i \le m$이다.

$$D(x) = \sum_{i=0}^{m-1} D_i x^i \qquad (36)$$
$$= D_0 + D_1 x + D_2 x^2 + D_3 x^3$$

식 (36)에서 계수항 만을 정리하면 식 (37)과 같다.

$$D_0 = d_0 + 2d_4 = a_0 b_0 + 2(a_3 b_1 + a_2 b_2 + a_1 b_3)$$
$$D_1 = d_1 + 2d_5 = a_1 b_0 + a_0 b_1 + 2(a_3 b_2 + a_2 b_3)$$
$$D_2 = d_2 + 2d_6 = a_2 b_0 + a_1 b_1 + a_0 b_2 + 2(a_3 b_3) \qquad (37)$$
$$D_3 = d_3 = a_3 b_0 + a_2 b_1 + a_1 b_2 + a_0 b_3$$

식 (36)의 곱셈 알고리즘을 이용하여 $m=4$이 짝수일 때의 $GF(3^m)$상의 곱셈기를 설계한다. $GF(3^4)$의 곱셈기를 설계하기 위하여 그림 3과 같이 m이 짝수인 곱셈기의 기본 셀을 구성하였다. 그림 3의 기본 셀은 m이 홀수일 때의 그림 1과 비교해 볼 때 a_i와 b_i의 위치가 바뀌면서 대각선의 방향이 우측에서 좌측으로 내려가는 형태임을 알 수 있다.



(a) 회로 (b) 기호

그림 3. $GF(3^m)$상에서 m이 짝수인 곱셈기의 기본 셀, (a) 회로 (b) 기호
Fig. 3. The basic cell of the multiplier with even m on $GF(3^m)$, (a) circuit (b) symbol

그림 4는 m이 짝수인 경우에 대하여 제시된 곱셈 알고리즘을 이용하여 구현한 $GF(3^4)$에서의 곱셈기 구성도이다. 그림 4의 곱셈기에서 기본 셀 사이에 있는 3개의 상수 2는 m이 짝수일 경우에 대한 곱셈 알고리즘을 구현할 때 3치에 따른 식 (37)에서의 값이다.

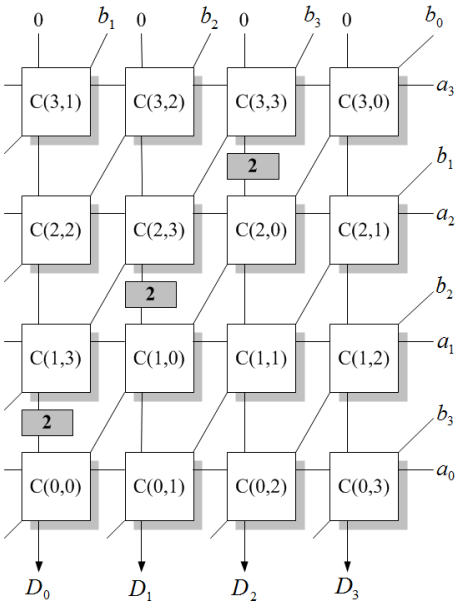


그림 4. $GF(3^4)$ 상의 제안된 곱셈기
Fig. 4. The proposed multiplier on $GF(3^4)$

Ⅳ. 성능 비교

본 논문에서는 유한체 $GF(3^m)$ 상에서 모든 항의 계수가 0이 아닌 기약다항식의 두 원소를 곱셈하는 새로운 곱셈 알고리즘을 제시하였다. 제시된 곱셈 알고리즘을 이용하여 모듈 구조의 병렬 입-출력 곱셈기를 구성하였다. 제시된 곱셈기는 Wang[12], Wei[13], Lee[8] 그리고 Kim[9] 등의 논문들과 곱셈기의 구성 형태, 각 셀당 사용된 게이트 수, 곱셈기에 사용된 전체 게이트 수, 그리고 셀당 지연시간, 전체 지연시간 등에 대하여 비교하였다.

전체 mod(3) 곱셈 게이트의 수는 m이 증가함에 따라 Wang과 Wei의 곱셈기는 $4m^2$과 $3m^2$으로 증가하나 Lee의 첫 번째 곱셈기와 본 논문의 곱셈기에서는 $(m+1)^2$으로 증가하며, Lee의 두 번째 곱셈기에서는 출력단에 1개씩 더 사용하므로 $(m+1)(m+2)$로 증가한다. Kim의 곱셈기는 $4m(m-2)$로 증가한다.

mod(3) 가산 게이트의 전체 개수는 m이 증가함에 따라 Wang의 곱셈기에서는 m^2으로 증가하며, Wei가 제안한 곱셈기에서는 3입력 mod(3) 가산 게이트를 하나 더 사용하므로 $2m^2$으로 증가한다. Kim의 곱셈기는 $4m(m-2)$로 증가한다. Lee의 첫 번째 곱셈기와 본 논문의 곱셈기에서는 $(m+1)^2$으로 증가하며, Lee의 두 번째 곱셈기에서는 출력단에 1개씩 더 사용하므로 $(m+1)(m+2)$로 증가한다.

셀당 래치의 사용 개수는 Wang의 곱셈기에서는 7개씩 사용되었으며, Wei의 곱셈기는 10개, Lee의 첫 번째 곱셈기에서는 3개, 두 번째 곱셈기에서는 4개가 사용되었다. Kim의 곱셈기는 12개 사용되었다. 일반적으로 회로가 클럭에 의해 동작할 경우 회로 동작의 안정을 위하여 래치를 사용하나, 본 논문에서는 곱셈결과를 얻기 위하여 최종결과가 출력될 때까지를 하나의 클럭으로 동작한다.

셀당 지연시간은 Wei, 그리고 Lee의 첫 번째 논문에서는 $T_A+T_X+2T_L$이며, Wang과 본 논문의 곱셈기에서는 T_A+T_X이며, Lee의 두 번째 논문에서는 T_A+T_L이다. 전체 지연시간에 있어서는 Wang, 그리고 Wei의 곱셈기는 $3m$시간이 필요하며, Kim은 $T_A+T_X+T_L$시간이 필요하다. Lee와 본 논문의 곱셈기는 $m+1$의 전체 지연시간이 필요하다.

표 1은 병렬 입-출력 곱셈기들의 비교를 보인 것이다. 본 논문에서는 셀에 래치를 사용하지 않았으며 셀당 게이트 수 및 전체 게이트의 수를 비교해 볼 때, 가장 낮은 복잡성을 갖는다. 또한 본 논문의 곱셈기는 셀당 지연시간과 전체 지연시간도 비교 논문들 중 가장 낮다.

표 1. 고속 병렬 입-출력 곱셈기의 비교
Table 1. The comparison for high-speed parallel input-output multipliers.

항목 \ 승산기	Wang[12]	Wei [13]]	Lee[8] (1)
전체 게이트 수(개)			
2 입력 곱셈 게이트	$4m^2$	$3m^2$	$(m+1)^2$
2 입력 가산 게이트	0	m^2	$(m+1)^2$
3 입력 가산 게이트	m^2	m^2	0
1 비트 래치	$7m^2$	$10m^2$	$\approx 4(m+1)^2$
셀당 지연시간	T_A+T_X	$T_A+3T_X+2T_L$	$T_A+T_X+T_L$
전체 지연시간	3m	3m	m+1
항목 \ 승산기	**Lee[8] (2)**	**Kim[9]**	**This paper**
전체 게이트 수(개)			
2 입력 곱셈 게이트	$(m+1)^2$	$4m(m-2)$	$(m)^2$
2 입력 가산 게이트	$(m+1)(m+2)$	$4m(m-2)$	$(m)^2$
3 입력 가산 게이트	0	0	0
1 비트 래치	$\approx 5(m+1)^2$	$3(m-2)^2$	0
셀당 지연시간	T_X+T_L	$T_A+T_X+2T_L$	T_A+T_X
전체 지연시간	m+1	m/2+2	m+1

(주) T_A = 2 입력 AND 게이트의 지연시간
T_X = 2 입력 XOR 게이트의 지연시간
$3T_X$ = 3 입력 XOR 게이트의 지연시간
T_L = 래치의 지연시간

V. 결 론

본 논문에서는 유한체 $GF(3^m)$상에서 모든 항에 0이 아닌 계수가 존재하는 기약 다항식에 대하여 m이 홀수 및 짝수인 경우인 $GF(3^m)$상의 곱셈 알고리즘을 제시하였으며, 제시된 곱셈 알고리즘을 이용하여 고속의 병렬 입-출력 모듈구조의 곱셈기를 설계하였다.

본 논문에서 제안한 $GF(3^m)$상의 곱셈 알고리즘에 의한 곱셈기의 설계는 $(m+1)^2$개의 동일한 셀로 설계되었으며, 기본 셀은 1개의 2입력 mod(3) 가산 게이트와 1개의 2입력 mod(3) 곱셈 게이트로 구성하였다. $GF(3^m)$상에서 m이 짝수인 경우의 곱셈기는 회로 구성에 있어서 m이 홀수인 경우와 비교해 볼 때, 곱셈기의 해당하는 셀 아래 위치에 2값을 곱해 주어야 하므로 m이 홀수인 경우와는 다르게 대각선 연결선의 방향은 우측에서 좌측으로 내려가는 방향으로 설계하였다.

본 논문에서 제시된 곱셈기는 클럭이 필요하지 않고 m개의 mod(3) 가산 게이트 소자 지연시간과 1개의 mod(3) 곱셈 게이트 소자의 지연시간만을 필요로 한다. 또한 셀에 래치를 사용하지 않았으므로 회로가 간단하며, 셀 당 게이트 수는 2개, 곱셈기에 사용된 전체 게이트의 수는 $(m+1)^2$로서 비교 논문들 중에 가장 적은 수의 게이트가 사용되었다. 또한 셀 당 지연시간도 T_A+T_X로서 가장 적으므로 곱셈기의 전체 지연시간도 적다. 본 연구에서 구성한 곱셈기는 규칙성과 셀 배열에 의한 모듈성을 가지므로 확장이 용이하며 VLSI회로 실현에 적합할 것이다.

REFERENCES

〔1〕 K. C. Smith. "The Prospect for Multivalued Logic : A Technology and Applications View," IEEE Trans. Computers, Vol. C-30, No. 9, pp. 619-634, Sept. 1981.

〔2〕 M. Kameyama and T. Higuchi, "Multiple-Valued Logic and Special Purpose Processors : Overview and Future," in Proc. IEEE Int. Symp. Multiple-Valued Logic, pp. 289-292, 1982.

〔3〕 S .L. Hurst, "Multiple-valued Logic-its Future," IEEE Trans. Computers, Vol 30, pp. 1161-1179, Dec. 1984.

〔4〕 M. A. Hasan, M. Wang, and V. K. Bhargava, "Moduler Construction of Low Complexity Parallel Multipliers for a Class of Finite Fields $GF(2^m)$," IEEE Trans. Computers, Vol. 41, No. 8, pp. 961-971, Aug. 1992.

〔5〕 3rd Generation Partnership Project., "Technical specification group GSM/EDGE radio access network; channel coding (release 5)," Tech. Rep. 3GPP TS 45.003 V5.6.0, June 2003.

〔6〕 C. K. Koc, and B. Sunar, "Low Complexity Bit-Parallel Canonical and Normal Basis Multipliers for a Class of Finite Fields," IEEE Trans. Computers, Vol. 47, No. 3, pp. 353-356, Mar. 1998.

〔7〕 A. Halbutogullari and C. K. Koc, "Mastrovito Multiplier for General Irreducible Polynomials," IEEE Trans. Computers, Vol. 49, No. 5, pp. 503-518, May 2000.

〔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. 385-392, May 2001.

〔9〕 T. W. Kim, W. J. Lee, and K. W. Kim, "Bit-Parallel Systolic Multiplication Architecture with Low Complexity and Latency in $GF(2^m)$ Using Irreducible AOP," Journal of KIIT, Vol. 11, No. 3, pp. 133-139, March 2013.

〔10〕 K. W. Kim and J. C. Jeon, "Montgomery Multiplication Architecture Based on Cellular Systolic Array over $GF(2^m)$," Journal of KIIT, Vol. 10, No. 9, pp. 1-6, Sept. 2013.

〔11〕 N. S Chang, T. H. Kim, C. H. Kim, D. G. Han, and H. W. Kim, "Digit-Serial Finite Field Multipliers for $GF(3^m)$," Journal of IEIE, Vol. 45-SD, No. 10, pp. 23-30, Oct. 2008.

〔12〕 C. L. Wang and J. L. Lin, "Systolic Array Implementation of Multipliers for Finite Fields $GF(2^m)$," IEEE Trans. Circuits and Systems,

Vol. 38, No. 7, July 1991.
[13] S. W. Wei, "A Systolic Power-Sum Circuit for $GF(2^m)$," IEEE Trans. Computers, Vol. 43, No. 2, pp. 226-229, Feb. 1994.

저 자 소 개

성 현 경
1982: 인하대학교
전자공학과 공학사.
1984: 인하대학교
전자공학과 공학석사.
1991: 인하대학교
전자공학과 공학박사.
현 재: 상지대학교
컴퓨터정보공학부 교수
관심분야: 다치논리설계, 정보 및 부호 이론.
Email : hkseong@sangji.ac.kr