유한체 GF(2^m)상의 비트-병렬 곱셈기의 설계

성 현 경*

Design of Bit-Parallel Multiplier over Finite Field GF(2^m)

Hyeon-Kyeong Seong*

이 논문은 2006년도 상지대학교 교내연구비 지원에 의해 연구되었음

요 약

본 논문에서는 $GF(2^m)$상에서 표준기저를 사용한 두 다항식의 곱셈을 비트-병렬로 실현하는 새로운 형태의 비트-병렬 곱셈기를 제안하였다. 곱셈기의 구성에 앞서, 피승수 다항식과 기약다항식의 곱셈을 병렬로 수행한 후 승수 다항식의 한 계수와 비트-병렬로 곱셈하여 결과를 생성하는 VCG를 구성하였다. VCG의 기본 셀은 2개의 AND 게이트와 2개의 XOR 게이트로 구성되며, 이들로부터 두 다항식의 비트-병렬 곱셈을 수행하여 곱셈결과를 얻도록 하였다. 이러한 과정을 확장하여 m에 대한 일반화된 회로의 설계를 보였으며, 간단한 형태의 곱셈회로 구성의 예를 $GF(2^4)$를 통해 보였다. 또한 제시한 곱셈기는 PSpice 시뮬레이션을 통하여 동작특성을 보였다. 본 논문에서 제안한 곱셈기는 VCG의 기본 셀을 반복적으로 연결하여 구성하므로, 차수 m이 매우 큰 유한체상의 두 다항식의 곱셈에서 확장이 용이하며, VLSI에 적합하다.

ABSTRACT

In this paper, we present a new bit-parallel multiplier for performing the bit-parallel multiplication of two polynomials in the finite fields $GF(2^m)$. Prior to construct the multiplier circuits, we consist of the vector code generator(VCG) to generate the result of bit-parallel multiplication with one coefficient of a multiplicative polynomial after performing the parallel multiplication of a multiplicand polynomial with a irreducible polynomial. The basic cells of VCG have two AND gates and two XOR gates. Using these VCG, we can obtain the multiplication results performing the bit-parallel multiplication of two polynomials. Extending this process, we show the design of the generalized circuits for degree m and a simple example of constructing the multiplier circuit over finite fields $GF(2^4)$. Also, the presented multiplier is simulated by PSpice. The multiplier presented in this paper use the VCGs with the basic cells repeatedly, and is easy to extend the multiplication of two polynomials in the finite fields with very large degree m, and is suitable to VLSI.

키워드

Finite fields $GF(2^m)$), Bit-parallel multiplier, Systolic multiplier, irreducible polynomial)

* 상지대학교 컴퓨터정보공학부

접수일자 2008. 02. 15

Ⅰ. 서론

유한체는 오류정정부호, 디지털신호처리 및 화상처리, 디지털통신의 암호화 및 해독화를 요하는 보안통신 등에서 많이 응용되고 있다. 특히 유한체 $GF(2^m)$는 신호처리와 화상처리 분야에서 특별한 계산을 요하거나 범용 컴퓨터계산의 고속화를 보조하는 고성능 전용 컴퓨터의 설계에 주목을 받고 있으며 Reed-Solomon 부호기 및 복호기의 VLSI 설계에 응용되고 있다[1-5]. 이들 중 오류정정부호의 설계는 유한체 $GF(2^m)$상의 연산을 사용한다. 실제로 오류정정부호기 및 복호기 설계는 전체 시스템의 규모와 성능에 절대적인 영향을 미치므로 유한체 $GF(2^m)$상의 연구는 회로경로의 연결, 시스템 구조의 복잡성과 동시성 등의 문제점을 개선하기 위하여 진행되고 있다[6-8].

유한체 $GF(2^m)$에서 연산은 디지털 2진 산술연산과 현저하게 다르다. 유한체상의 덧셈은 매우 간단하여 유한체의 원소들이 다항식의 형태로 표현되는 경우 XOR 게이트에 의한 비트별 연산으로 회로를 간단히 구성할 수 있으며, 비교적 적은 비용으로 빠른 속도 특성을 가지는 구현이 가능하다. 이와 달리 곱셈은 유한체상에서 중요한 연산으로 연산과정이 복잡할 뿐만 아니라 구현에 따른 비용이 크다. 따라서 회로가 복잡하지 않으면서 용이하게 연산을 실현할 수 있는 빠른 곱셈 알고리즘의 개발이 중요하다.

$GF(2^m)$의 원소를 표현할 때 표준 기저 표현법을 사용할 경우 곱셈 알고리즘은 승수의 처리 순서에 따라 LSB 우선과 MSB 우선 방법으로 구분되며, 일반적으로 LSB 우선 곱셈 알고리즘이 MSB 우선 곱셈 알고리즘에 비해 적은 계산 지연시간을 갖는다. 또한 $GF(2^m)$상의 곱셈기는 비트-병렬 및 비트-직렬 구조 곱셈기로 구분할 수 있으며, 일반적으로 비트-병렬형은 비트-직렬형에 비해 데이터 처리율이 높지만, 하드웨어가 복잡해지는 단점이 있다. 최근 빠른 처리속도와 복잡도를 고려한 VLSI 구현에 있어 규칙성과 모듈화가 매우 중요시되면서 이에 대한 적합한 유한체 곱셈기 설계에 관한 연구가 활발히 진행되고 있으며, 병렬 곱셈기 구조의 경우 회로는 복잡하지만 빠른 연산처리 능력을 가지고 있으므로 최근에 많이 연구되고 있다[9,10].

Yeh 등[11]은 유한체 $GF(2^m)$상에서 표준기저를 사용하여 $A \cdot B + C$ 연산을 수행하는 병렬 입-출력 시스토릭 구조의 곱셈기를 개발하였다. 이 곱셈기는 하나의 셀에 2개의 2입력 AND 게이트와 2개의 2입력 XOR 게이트를 사용하여 VLSI화에 적합하도록 설계하였으나 데이터의 역류 현상을 갖는 단점이 있다. Wang 등[12]은 Sunar와 Koc에 의해 제안된 병렬형 곱셈기로부터 유도된 직렬형과 병렬형 곱셈기의 일종인 타입 II 최적 정규기저에서 수행하는 새로운 종류의 곱셈기를 개발하였다. 이 곱셈기는 ModelSim 도구로 시뮬레이션하였고, Xilinx의 ISE로 합성하였다. Xilinx의 FPGA 장비로 실현된 회로기판 실험이 회로기판에서 80MHz에서 동작한다. Petra 등[13]은 Mastrovito 곱셈기의 첫 번째 블록의 최소-영역 실현과 Matrovito 곱셈기의 두 번째 블록의 고속 지연 유도 나무구조를 이용하여 새로운 곱셈기를 개발하였으며, 곱셈기의 복잡성과 지연시간에 대하여 여러 다항식을 분석적으로 평가하였다. 제안된 곱셈기는 실제 응용에서 사용할 수 있도록 (255, 239) Reed-Solomon 디코더를 해석할 수 있도록 0.25㎛ CMOS 기술로 실현함으로써 증명하였다. Wu 등[14]은 유한체에서 기약 AOP(All One Polynomial)와 기약 ESP(Equally Spaced Polynomial)를 기반으로 하는 약한 이중 기저를 이용한 저 복잡성 비트-병렬 곱셈기를 제안하였으며, Halbutogullari 등[15]은 일반적인 기약다항식에 대한 병렬 곱셈기를 제안하였다. 이들이 제안한 유한체상의 곱셈기들은 보안 및 암호시스템 응용에 적합하다 할지라도 시스토릭 기술을 이용하여 설계된 것이 아닌 경우에는 m이 클 경우 $GF(2^m)$상의 곱셈에 대한 지연시간은 매우 큰 것이 단점이다. 이들의 연구 이외에도 많은 연구결과들이 도출되어 왔으며, 이들은 각각 독특한 회로설계 알고리즘과 회로구성으로 그 효용성을 입증 받았으며, 보다 개선된 회로구현을 위한 연구는 계속될 것으로 전망된다.

본 논문에서는 유한체 연산에 관한 기존의 연구결과를 토대로 $GF(2^m)$상의 표준기저를 이용하여 두 다항식의 곱셈을 실현하는 비트-병렬형의 곱셈기를 제안하였다. 제안된 비트-병렬형 곱셈기는 피승수 다항식과 기약다항식의 곱셈을 병렬로 수행한 후 승수 다항식의 한 계수와 비트-병렬로 곱셈하여 결과를 생성하는 벡터코드 생성기(Vector Code Generator; VCG)로 구성된다. VCG의 기본 셀은 2개의 AND 게이트와 2개의 XOR 게이트를 사용하여 구성하였다. 본 논문에서 제안한 VCG

는 벡터의 각 비트들의 병렬연산에 의해 동작되며, 각 VCG 간에 2개의 메모리 소자를 사용하여 회로 동작의 안정성 갖는다. 또한 회로 구성의 모듈화, 블록화 함으로서 m에 대한 확장과 VLSI에 유리하도록 하였다.

Ⅱ. $GF(2^m)$상의 덧셈과 곱셈 알고리즘

1. 유한체상의 덧셈

유한체 $GF(2^m)$은 p가 소수(prime number)이고 m이 양의 정수인 p^m개의 원소들을 갖는다. 유한체 $GF(2^m)$은 2개의 원소들을 갖는 기초체(ground field) $GF(2)$의 확대체이다. 즉, 유한체 $GF(2)$는 {0,1}의 원소들을 구성한다[3,19,20]. $GF(2^m)$에서 모든 산술연산은 그 결과를 mod(2) 연산을 함으로써 이루어진다. $GF(2^m)$의 0이 아닌 모든 원소들은 원시원소 α에 의해 생성되며, α는 $GF(2^m)$의 원시 기약 다항식 $F(x)=0$의 근이다.

$$F(x)=\sum_{i=0}^{m} f_i \cdot x^i \tag{1}$$

여기서 $F(x)$는 최고 차수 m의 계수 $f_m=1$인 모닉 다항식(monic polynomial)이다. 또한 $GF(2^m)$의 0이 아닌 원소들은 α의 승(power)으로서 표현이 가능하며 식 (2)와 같다.

$$GF(2^m)=\left\{0, \alpha^1, \alpha^2, \cdots, \alpha^{2^m-2}, \alpha^{2^m-1}=1\right\} \tag{2}$$

원시 기약 다항식 $F(\alpha)=0$ 임으로 식 (3)과 같이 구할 수 있다.

$$F(\alpha)=\alpha^m+f_{m-1}\cdot\alpha^{m-1}+\cdots+f_1\cdot\alpha^1+f_0=0$$
$$\alpha^m=\sum_{i=0}^{m-1} f_i \cdot \alpha^i \tag{3}$$

그러므로 $GF(2^m)$상의 원소들은 m보다 더 낮은 차수를 갖는 α의 다항식으로 식 (4)와 같이 표현할 수 있다.

$$GF(2^m)=\sum_{i=0}^{m-1} a_i \cdot \alpha^i \ ; \alpha_i \in GF(2) \tag{4}$$

유한체 $GF(2^m)$에서 임의의 다항식 $A(x)$는 식 (5)와 같이 표현할 수 있다.

$$A(x)=a_{m-1}\cdot x^{m-1}+a_{m-2}\cdot x^{m-2}+\cdots+a_1\cdot x^1+a_0$$
$$=\sum_{i=0}^{m-1} a_i \cdot x^i \tag{5}$$

또한 임의의 다항식 $B(x)$는 식 (6)과 같이 표현할 수 있다.

$$B(x)=b_{m-1}\cdot x^{m-1}+b_{m-2}\cdot x^{m-2}+\cdots+b_1\cdot x^1+b_0$$
$$=\sum_{i=0}^{m-1} b_i \cdot x^i \tag{6}$$

임의의 두 다항식 $A(x)$와 $B(x)$의 덧셈은 식 (7)과 같이 나타낼 수 있다[3].

$$S(x)=A(x)+B(x)$$
$$=s_{m-1}\cdot x^{m-1}+s_{m-2}\cdot x^{m-2}+\cdots+s_1\cdot x^1+s_0$$
$$=\sum_{k=0}^{m-1} s_k \cdot x^k \tag{7}$$

여기서 $s_k=(a_i+b_i)\bmod(2)$이고, $0\le k\le m-1$이다.

$GF(2^m)$상에서 임의의 두 다항식의 덧셈은 모듈러-2 덧셈을 $\oplus$기호로 나타낼 때, 식 (7)의 각 계수 $s_i=a_i\oplus b_i$ $(0\le i\le m-1)$와 같이 간단히 구할 수 있다. 그러므로 유한체 $GF(2^m)$상의 덧셈회로는 m개의 비트 독립적인 XOR 게이트들에 의해 쉽게 구현된다.

2. 곱셈 알고리즘

유한체 $GF(2^m)$상의 곱셈은 덧셈에 비해 매우 복잡하게 구현되며, 곱셈의 전개방식에 따라 다양한 회로구현이 가능하다. 유한체 $GF(2^m)$상에서 두 다항식 $A(x)$와 $B(x)$의 곱셈 결과를 $P(x)$는 식 (8)과 같이 나타낼 수 있다.

$$P(x)=\{A(x)\cdot B(x)\}\bmod(F(x))$$
$$=p_{m-1}\cdot x^{m-1}+p_{m-2}\cdot x^{m-2}+\cdots+p_1\cdot x^1+p_0$$
$$=\sum_{i=0}^{m-1} p_i \cdot x^i \tag{8}$$

식 (8)을 자세히 표현하기 위하여 기약다항식을 $F(x)$라 하였을 때 비트-병렬을 수행하는 곱셈 알고리즘은 다음과 같다.

$$\begin{aligned} P(x) &= \{A(x) \cdot B(x)\} \bmod (F(x)) \\ &= \left\{ A(x) \cdot \left(\sum_{i=0}^{m-1} b_i \cdot x^i \right) \right\} \bmod (F(x)) \\ &= \left\{ \sum_{i=0}^{m-1} b_i \cdot (A(x) \cdot x^i) \right\} \bmod (F(x)) \\ &= \sum_{i=0}^{m-1} b_i \cdot \left\{ \left(\sum_{k=0}^{m-1} a_k^{(i)} \cdot x^{(i+k)} \right) \right\} \bmod (F(x)) \quad (9) \end{aligned}$$

식 (9)의 오른쪽 항인 $\left\{ \left(\sum_{k=0}^{m-1} a_k^{(i)} \cdot x^{(i+k)} \right) \right\} \bmod (F(x))$은 식 (2)의 $F(x)$에 의하여 계수항의 연산을 식 (10)과 같이 나타낼 수 있다.

$$\begin{aligned} a_k^{(i+1)} &= \left(f_k \cdot a_{m-1}^{(i)} \right) \oplus a_{k-1}^{(i)} \quad (1 \le k \le m-1) \\ &= f_k \cdot a_{m-1}^{(i)} \quad (k=0) \end{aligned} \quad (10)$$

식 (10)에서 $k=0$인 경우 $a_{k-1}^{(i)} = 0$이다.

그리고 식 (9)에 식 (10)을 대입하면 식 (11)과 같이 나타낼 수 있다.

$$P(x) = \sum_{i=0}^{m-1} \sum_{k=0}^{m-1} \left\{ \left(b_i \cdot a_k^{(i+k)} \right) \oplus p_{k-1}^{(i)} \right\} \cdot x^i \quad (11)$$

식 (11)에서 $i=0$일 때 오른쪽 항 $p_{k-1}^{(i)}$은 0이다.

식 (11)에서 곱셈결과 $P(x)$의 계수항을 나타내면 식 (12)와 같다.

$$\begin{aligned} p_k^{(i+1)} &= \left(b_i \cdot a_k^{(i+1)} \right) \oplus p_{k-1}^{(i)} \\ &= \left\{ b_i \cdot \left(f_k \cdot a_{m-1}^{(i)} \right) \oplus a_{k-1}^{(i)} \right\} \oplus p_{k-1}^{(i)} \end{aligned} \quad (12)$$

그러므로 식 (10)은 다항식 $A(x)$와 기약다항식 $F(x)$을 병렬로 연산을 구현하고, 식 (11)에서 i를 0에서 $m-1$까지 순차적으로 대입하여 반복적으로 구할 수 있다.

Ⅲ. 유한체 $GF(2^m)$상의 비트-병렬 곱셈기의 설계

이 장에서는 앞장에서 논한 $GF(2^m)$상의 곱셈 알고리즘 $P(x) = \{A(x) \cdot B(x)\} \bmod (F(x))$를 실행하는 비트-병렬 곱셈기의 설계를 논한다. 유한체상의 피승수 다항식과 기약다항식의 계수항만을 병렬로 연산하고, 승수 다항식의 한 계수와 비트-병렬로 연산을 수행하는 벡터코드 생성기(Vector Code Generator; VCG)의 기본 셀은 그림 1과 같다.

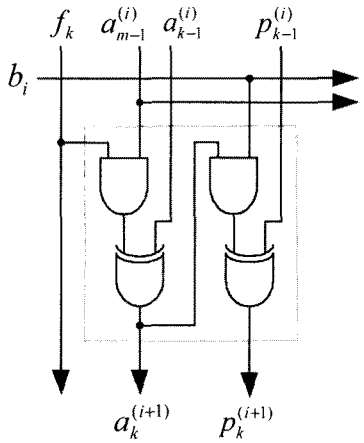

그림 1. VCG의 기본 셀 회로
Fig. 1. Basic Cell circuit of VCG.

그림 1의 VCG의 기본 셀은 2개의 AND 게이트와 2개의 XOR 게이트로 구현하였으며, 식 (13)을 수행한다.

$$\begin{aligned} p_k^{(i+1)} &= \left(b_i \cdot a_k^{(i+1)} \right) \oplus p_{k-1}^{(i)} \\ &= \left\{ b_i \cdot \left(f_k \cdot a_{m-1}^{(i)} \right) \oplus a_{k-1}^{(i)} \right\} \oplus p_{k-1}^{(i)} \end{aligned} \quad (13)$$

그림 1의 VCG의 기본 셀을 사용하여 식 (11)에서 $m=4$인 유한체 $GF(2^4)$상의 다항식 $B(x)$의 임의의 계수 b_i를 구하는 VCG를 구현하면 그림 2와 같다. 그림 2의 VCG는 기약다항식과 피승수 다항식 $A(x)$를 병렬로 연산을 수행한 결과와 승수 다항식 $B(x)$의 한 계수와 병렬로 연산을 수행한다. 그림 2는 VCG의 1비트 연산회로를 보였다.

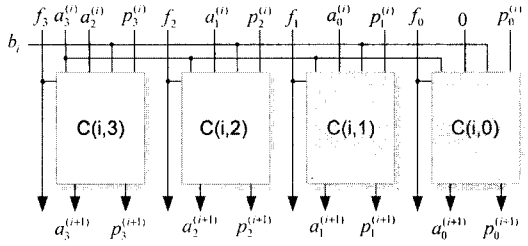

그림 2. $GF(2^4)$의 VCG circuit.
Fig. 2. VCG circuit of $GF(2^4)$

그림 3은 유한체 $GF(2^4)$상의 VCG 회로의 시뮬레이션 결과를 보인 것이다. $GF(2^4)$의 기약다항식 중 $F(x)=x^4+x+1$를 사용하였다. 그림 3에서 시간이 60μs에서 $A(x)$=(1 0 1 1)이고, $B(x)$= (1)의 1 비트만을 곱하였을 경우 시뮬레이션 결과 $P(x)$= (1 0 1 1)를 보인다.

그림 3의 유한체 $GF(2^4)$의 VCG 회로를 사용하여 유한체 $GF(2^4)$상의 임의의 두 다항식에 대한 비트-병렬 곱셈기 회로를 구성하면 그림 4와 같다. 그림 2의 VCG는 피승수 다항식 $B(x)$의 계수항 만큼 반복적으로 사용되며, 유한체 $GF(2^4)$상의 비트-병렬 곱셈 연산을 위해 소요되는 기본 셀은 $m=4$인 경우 $4\times4(m\times m)$개가 필요하며, 메모리 소자는 $2m\times(m-1)$개인 24개가 요구된다.

그림 4의 비트-병렬 곱셈기 회로는 $GF(2^4)$의 기약다항식 중 $F(x)=x^4+x+1$를 사용하며, 이를 통해 $GF(2^4)$의 모든 원소들은 x^3, x^2, x^1, x^0의 기저들로 표현될 수 있으며, 이를 표 1에 정리하였다.

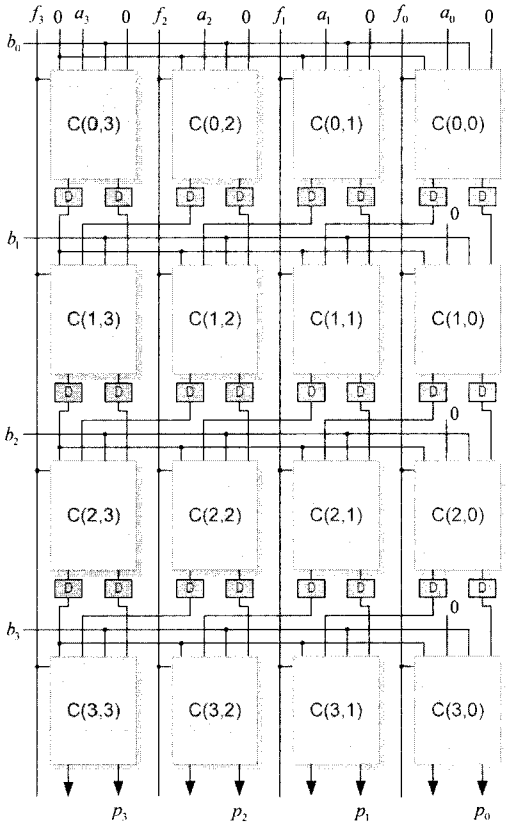

그림 4. $GF(2^4)$의 비트-병렬 곱셈 회로
Fig. 4. Bit-parallel multiplier circuit of $GF(2^4)$

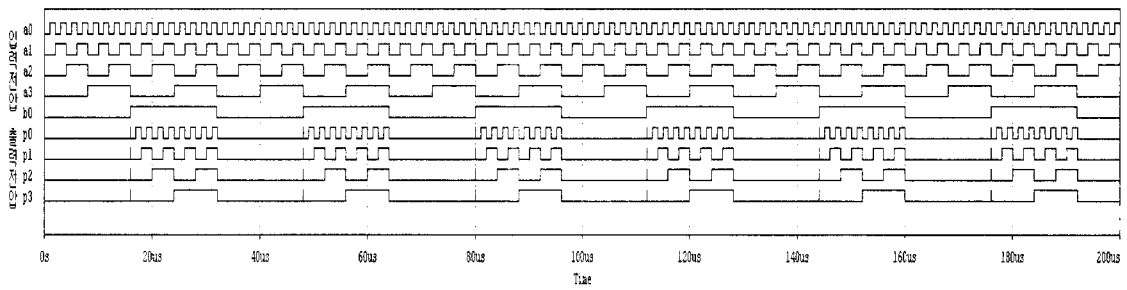

그림 3. $GF(2^4)$의 VCG 회로 시뮬레이션 결과
Fig. 3. Simulation result of VCG circuit over $GF(2^4)$

표 1. 표준기저에 의한 $GF(2^4)$상의 원소들
Table 1. Elements of $GF(2^4)$ using standard basis.

원소	다항식 표현	벡터표현
		$x^3 x^2 x^1 x^0$
x^{15}	0	0000
x^0	$x^0=1$	0001
x^1	x^1	0010
x^2	x^2	0100
x^3	x^3	1000
x^4	x^1+x^0	0011
x^5	x^2+x^1	0110
x^6	x^3+x^2	1100
x^7	$x^3+x^1+x^0$	1011
x^8	x^2+x^0	0101
x^9	x^3+x^1	1010
x^{10}	$x^2+x^1+x^0$	0111
x^{11}	$x^3+x^2+x^1$	1110
x^{12}	$x^3+x^2+x^1+x^0$	1111
x^{13}	$x^3+x^2+x^0$	1101
x^{14}	x^3+x^0	1001

그림 4에서 승수 다항식 $B(x)$의 첫 번째 계수 b_0가 입력으로 들어가는 VCG 회로에서 곱셈 결과 다항식의 계수항 $p_k^{(0)}$는 모두 0이 가해진다. 이는 전단으로부터 곱셈 결과가 없기 때문이다.

그림 5는 $GF(2^4)$의 비트-병렬 곱셈기 회로에 대한 시뮬레이션 결과이다. 그림 5에서 비트-병렬 곱셈기 회로의 동작은 48μs에서 피승수 입력전압 $A(x)$는 (1 1 0 0)이고, 승수 입력전압 $B(x)$의 첫 번째 비트 b_0는 48μs에서 (0)이고, b_1은 52μs에서 (0), b_2은 56μs에서 (1)이고, b_3는 60μs에서 (1)의 값인 $B(x)$는 (1 1 0 0)을 가하면 출력전압 $P(x)$는 60μs에서 (1 1 1 1)을 보인다. 그러므로 $A(x)$가 가해진 후 3클럭 후에 출력전압의 결과 $P(x)$를 보인다.

Ⅳ. 비교 및 검토

본 논문에서 제안한 비트-병렬 곱셈회로를 포함하여 참고문헌의 곱셈회로들은 저마다의 독특한 성질과 장점을 갖는다. 일반적으로 사용되는 회로비교의 척도들은 간략화된 회로구성, 빠른 동작속도, 저전력 등이다. 회로의 간략화를 평가하기 위해서는 구성소자의 개수 및 소자 간 결선의 수, 입출력 단자의 수, 기타 부속회로 및 게이트의 개수, VLSI 구현시 필요한 면적 등을 고려하여야 한다. 또한 동작속도는 입력이 인가되면서 회로의 동작출력이 나타나기까지의 소자에 의한 지연시간과 클럭시간 등이 중요한 고려 요소이다. 이외에도 주변 회로 블록과의 호환 및 신호전달의 적합성, 예를 들면 부호기, 복호기의 필요 여부 등 다양한 항목을 통해 종합적으로 평가될 수 있다. 적용하고자 하는 목적에 따라 일부 항목에 대한 트레이드-오프 조건을 고려할 수도 있다. 따라서 일부 항목만의 단편적 비교를 통해 구성회로의 우열을 논하기는 쉽지 않은 문제이다. 그러나 대략적으로 회로의 비교를 위해 여러 참고문헌들은 구성회로의 소자 수와 시간지연에 대한 비교를 행하고 있으며, 본 논문에서도 이에 따라 비교하였다.

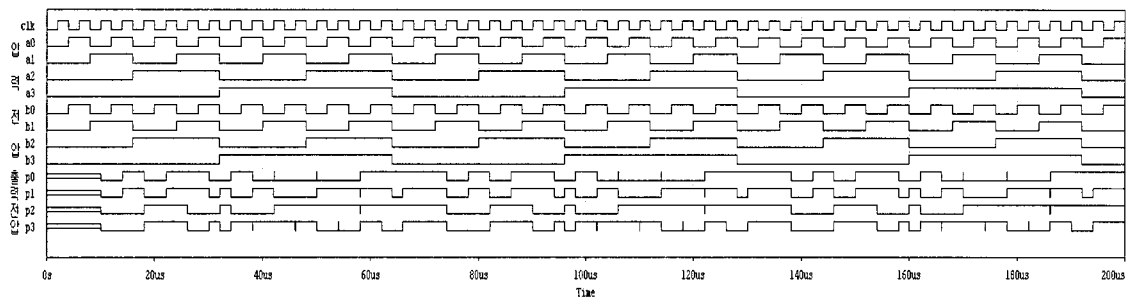

그림 5. 비트-병렬 곱셈기 회로의 시뮬레이션 결과
Fig. 5. Simulation result of bit-parallel multiplier circuit.

본 논문에서 제시한 비트-병렬 곱셈기의 구성과 참고문헌의 곱셈기들의 구성을 표 2에 정리하였다. 표 2에서 보인 것처럼 Mastrovito[18], Koc[7], Masoleh[21], Petra[13] 및 본 논문에서는 유한체상의 두 다항식 A와 B의 곱셈함수는 시스토릭 구조를 갖지 않기 때문에 전단에서 들어오는 초기치 C가 없어서 $P=A \cdot B$이며, Yeh 등[11], Kumar[27]와 Namin[29]의 곱셈기는 시스토릭 구조를 갖고 동작하기 때문에 곱셈함수 $P=A \cdot B+C$이다.

$GF(2^4)$상의 기약다항식 $F(x)$는 x^4+x+1, x^4+x^3+1와 $x^4+x^3+x^2+x+1$ 등이 있으며, Koc, Masoleh, Namin은 AOP로서 $F(x)=x^4+x^3+x^2+x+1$를 적용하여 회로를 구성하였으며, Kumar, Petra는 Trinomial을 기약다항식을 사용하였고, 본 논문과 Yeh, Mastrovito의 곱셈기는 $F(x)=x^4+x+1$을 적용하여 회로를 구성하였다. I/O 형식은 Yeh는 직렬형(1D)과 병렬형(2D)의 곱셈기를 제안하였으며, 본 논문과 비교의 일관성을 위해 병렬형 및 비트-병렬형 곱셈기에 대해서 논의하였다. 다만, 직렬형의 예를 보인 것은 전체 동작시간이 상당히 증가함을 보이기 위한 것이다. 유한체상에서 곱셈은 기약다항식에 따라 계산량이 많아지거나 적어진다. 그러므로 임의의 기약다항식에서 동작할 수 있는 일반성을 갖는 곱셈기를 설계하는 것이 연구의 목적이다. 그러므로 본 논문은 AND 게이트와 XOR 게이트를 사용하여 기본 셀들이 일반성을 갖게 설계하였다.

곱셈기를 구성하는 게이트의 수를 비교하면 m=4인 경우 AND 게이트는 Masoleh의 논문은 16개로 우수하며, 타 연구와 본 연구는 32개로 다소 증가한다. XOR 게이트는 타 논문의 경우 25개로 우수하며, 본 연구는 32개로 약간 증가한다. Koc, Masoleh의 논문은 D 플립플롭을 전혀 사용하지 않으며, Mastrovito, Kumar, Namin, Petra 및 본 논문은 많은 수의 D 플립플롭이 필요하다. 동작시간은 D 플립플롭을 사용하지 않는 Koc, Masoleh의 논문이 가장 우수하며, 본 논문도 타 연구에 비해 D 플립플롭을 다소 적게 사용한다. AOP 기약다항식은 수많은 기약다항식 중에서 특수한 기약다항식이며, 본 논문의 경우 AOP 기약다항식을 사용할 경우도 동일하게 소자들이 소요되는 장점이 있다.

곱셈기의 구조를 비교하면 Koc와 Masoleh는 D 플립플롭을 사용하지 않는 간단한 AND 와 XOR 의 배열 구

표 2. $GF(2^4)$상의 곱셈기들의 비교표
Table 2. Comparison table of multipliers on $GF(2^4)$

Multiplier Item	Yeh[11] 1-D	Yeh[11] 2-D	Mastrovito [18]	Koc[7]	Masoleh [18]	Kumar [22]	Namin [24]	Petra [13]	This Paper
1. Function	AB+C	AB+C	AB	AB	AB	AB+C	AB+C	AB	AB
2. F(x)	x^4+x+1	x^4+x+1	x^4+x+1	AOP	AOP	Trinomial	AOP	Trinomial	x^4+x+1 or AOP
3. I/O Format	Serial	Parallel	Bit-Parallel	Parallel	Parallel	Bit-Parallel	BSWP	Bit-Parallel	Bit-Parallel
4. AND	3m (12)	$2m^2$ (32)	$2m^2$ (32)	$2m^2$ (32)	m^2 (16)	$2m^2$ (32)	$2m^2$ (32)	$2m^2$ (32)	$2m^2$ (32)
5. XOR	2m (8)	$2m^2$ (32)	$(m+1)^2$ (25)	$(m+1)^2$ (25)	$(m+1)^2$ (25)	$(m+1)^2$ (25)	$2m^2$ (32)	$(m+1)^2$ (25)	$2m^2$ (32)
6. D Flip-Flop	10m+2 (42)	$7m^2+16$ (128)	$(m+1)^2$ (25)	-	-	$(m+1)^2$ (25)	$(m+1)^2$ (25)	2m(m-1) (24)	2m(m-1) (24)
7. Minimum clock period	$D_A+D_X+4D_L$	$D_A+D_X+2D_L$	$D_A+3D_X+5D_L$	D_A+3D_X	D_A+2D_X	$D_A+D_X+5D_L$	$D_A+2D_X+5D_L$	$D_A+2D_X+4D_L$	$D_A+2D_X+3D_L$
Comment	D_A = the propagation delay of one 2-input AND gate D_X = the propagation delay of one 2-input XOR gate D_L = the propagation delay of one latch ( ) = the total gate number of generalization for degree m=4 AOP means All One Polynomial of degree m BSWP = Bit-Serial Word-Parallel								

조로 구성되어 있으며 모듈성이 있으나 규칙성이 없어 소자가 증가하는 단점과 각 소자들 간의 연결이 매우 복잡한 단점이 있다. 반면에 Mastrovito, Kumar, Namin, Petra는 비트 시스토릭 구조로 동작하며, AND-XOR 셀 배열의 모듈성과 규칙성이 있으나 게이트 수가 증가하는 단점이 있다. 본 논문은 각 2개의 AND-XOR 셀 배열로 구성되어 있어 배열의 모듈성과 규칙성을 가지며, 소자간의 연결이 간단하고, 확장성이 용이한 장점이 있으며, 동작속도가 빠르다. 또한 AOP 기약 다항식을 사용하는 경우도 동일한 회로가 사용되므로 임의의 기약다항식에서도 동일한 동작속도를 갖는 장점이 있다. 또한 PSpice를 사용하여 제안한 고속 병렬 곱셈기의 동작을 확인하기 위해 시뮬레이션을 하였으며, 곱셈기가 정상적으로 동작함을 보였다.

V. 결 론

본 논문에서는 유한체상에서 곱셈을 수행하는 여러 가지 방법 중에서 한 가지 방법인 유한체 $GF(2^4)$ 상에서 두 다항식의 곱셈을 실현하는 비트-병렬을 갖는 곱셈기를 제시하였다. 이 곱셈기는 먼저 피승수 다항식과 기약다항식의 곱셈을 병렬로 수행한 후 승수 다항식의 한 계수와 비트-병렬로 곱셈하여 결과를 생성하는 VCG을 제안하였다. VCG의 기본 셀은 2개의 AND 게이트와 2개의 XOR 게이트로 구성되며,

이들로부터 두 다항식의 비트-병렬 곱셈을 수행하여 곱셈결과를 얻도록 설계하였다. 또한 PSpice에 의한 시뮬레이션을 통하여 제안한 비트-병렬 곱셈기가 정상적으로 동작함을 보였다.

제시한 비트-병렬 곱셈기는 $m=4$인 경우 AND 게이트가 32개, XOR 게이트의 수가 32개 소요된다. 비트-병렬 곱셈기의 VCG는 1 단위시간(클럭시간)이 소비되며, 전체 지연시간은 $m-1$ 단위시간이 소비된다. 그러므로 제시한 비트-병렬 곱셈기의 전체 시스템 동작시간은 $m-1$ 단위시간이 소요되어 타 연구의 곱셈기보다 전체 지연시간이 빠른 장점이 있다. 또한 유한체상에서 수많은 기약다항식 중 특수한 기약다항식인 AOP 기약다항식을 사용할 경우도 동작속도는 변화가 없는 장점이 있다.

본 논문에서 제시한 비트-병렬 곱셈기는 각 2개의 AND-XOR 기본 셀들의 배열로 구성되기 때문에 회선경로선택의 규칙성, 단순성, 배열의 모듈성, 병렬 동작의 이점을 가지며 특히 차수 m이 증가하는 유한체상의 두 다항식의 곱셈에서 확장성을 갖이므로 다양한 유한체 연산회로에 적용할 수 있을 것이다.

참고문헌

[1] B. A. Laws and C. K. Rushforth, "A Cellular Array Multiplier for $GF(2^m)$," IEEE Trans. Computers, vol. C-20, pp. 1573-1578, Dec. 1971.

[2] H. M. Shao, T. K. Truong, L. J. Deutsch, J. H. Yaeh and I. S. Reed, "A VLSI Design of a Pipelining Reed-Solomon Decoder," IEEE Trans. Computers, vol. C-34, pp. 393-403, May 1985.

[3] C. C. Wang, T. K. Truong, H. M. Shao, 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. C-34, pp. 709-717, Aug. 1985.

[4] P. A. Scott, S. E. Tarvares and L. E. Peppard, "A Fast Multiplier for $GF(2^m)$," IEEE J. Select. Areas Communications, vol. SAC-4, no. 1, pp. 707-717, Jan. 1986.

[5] I. S. Hsu, T. K. Truong, L. J. Deutsch and I. S. Reed, "A Comparison of VLSI Architecture of Finite Field Multipliers Using Dual, Normal, or Standard Bases," IEEE Trans. Computers, vol. C-37, no. 6, pp. 735-739, Jun. 1988.

[6] 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.

[7] 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.

[8] Kiamal Z. Pekmestzi, "Multiplexer-Based Array Multipliers," IEEE Trans. Computers, vol. 48, no.1, pp. 15-23, Jan. 1999.

[9] 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, Nov. 1998.

[10] J. J. Wonziak, "Systolic Dual Basis Serial Multiplier," IEE Proceeding Computers and Digital Technology, vol. 145, no. 3, pp.237-241, July 1998.

[11] C. S. Yeh, I. S. Reed and T. K. Truong, "Systolic Multipliers for Finite Field $GF(2^m)$," IEEE Trans. Computers, vol. C-33, pp. 357-360, Apr. 1984.

[12] Y. Wang, Z. Tian, X. Bi and Z. Niu, "Efficient Multiplier over Finite Field Represented in Type II Optimal Normal Basis," Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications (ISDA '06), 2006.

[13] N. Petra, D. de Caro and A. G.M. Strollo, "A Novel Architecture for Galois Fields $GF(2^m)$ Multipliers Based on Mastrovito Scheme," IEEE Trans. Computers, vol. 58, no. 11, pp.1470-1483, Nov. 2007.

[14] H. Wu and H. A. Hasan and L. F. Blake, "New Low-Complexity Bit-Parallel Finite Fields Multipliers Using Weekly Dual Basis," IEEE Trans. Computers, vol. 47, no. 11, pp. 1223-1234, Nov. 1998.

[15] A. Halbutogullari and C. K. Koc, "Mastrovito Multiplier for General Irreducible Polynomials," IEEE Trans. Computers, vol. 49, no. 5, pp. 503-518, May 2000.

[16] E. D. Mastrovito, "VLSI Design for Multiplication on Finite Field $GF(2^m)$," Proc. International Conference on Applied Algebraic Algorithms and Error-Correcting Code, AAECC-6, Roma, pp. 297-309, July 1998.

[17] S. B. Wicker and V. K. Bhargava, Error Correcting Coding Theory, McGraw- Hill, New York, 1989.

[18] A. R. Masoleh and M. A. Hasan, "A New Construction of Massey-Omura Parallel Multiplier over $GF(2^m)$," IEEE Trans. Computers, vol. 51, no. 5, pp. 511-520, May 2002.

[19] H. Wu, "Bit-Parallel Finite Field Multiplier and Squarer Using Polynomial Basis," IEEE Trans. Computers, vol. 51, no. 7, pp. 750-758, July 2002.

[20] H. Fan and Y. Dai, "Fast Bit-Parallel $GF(2^m)$ Multiplier for All Trinomials," IEEE Trans. Computer, vol. 54, no. 4, pp.485-490, Apr.. 2005.

[21] A. K. Daneshbeh, and M. A. Hasan, "A Class of Unidirectional Bit Serial Systolic Architectures for Multiplicative Inversion and Division over $GF(2^m)$," IEEE Trans. Computer, vol. 54, no. 3, pp.370-380, Mar. 2005.

[22] S. Kumar, T. Wollinger and C. Paar, "Optimum Digit Serial $GF(2^m)$ Multipliers for Curve-Based Cryptography," IEEE Trans. Computers, vol. 55, no. 10, pp.1306-1311, Oct. 2006.

[23] J. Imana, J. M. Sanchez and F. Tirado, "Bit-Parallel Finite Field Mutlipliers for Irreducible Trinomials," IEEE Trans. Computers, vol. 55, no. 5, pp.520-533, May 2006.

[24] A. H. Namin, H. Wu and M. Ahma야, "Comb Architectures for Finite Field Multiplication in IF_{2^m}," IEEE Trans. Computers, vol. 56, no. 7, pp.909-916, July 2007.

[25] K. Sakiyama, L. Batina, B. Preneel and I. Verbauwhede, "Multicore Curve-Based Cryptoprocessor with Reconfigurable Modular Arithmetic Logic Units over $GF(2^m)$," IEEE Trans. Computers, vol. 56, no. 9, pp.1269-1282, Sep. 2007.

저자소개

성현경(Hyeon-Kyeong Seong)

1982년 인하대학교 전자공학과 공학사
1984년 인하대학교 대학원 전자공학과 공학석사
1991년 인하대학교 대학원 전자공학과 공학박사
2005년~2006년 미국 Naval Postgraduate School 방문교수
1991년~현재 상지대학교 컴퓨터정보공학부 교수
※관심분야 : Multiple-Valued Logic Design, Computer Architecture & VLSI 설계, Information & Coding Theory, Cryptography Theory & Security, RFID/WSN 설계 및 응용 등