유한체 $GF(3^m)$상의 고속 병렬 승산기의 구성

최용석* · 박승용** · 성현경***

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

Yong-Seok Choi* · Seung-Yong Park** · Hyeon-Kyeong Seong***

요 약

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 승산 알고리즘을 제시하였으며, 제시된 승산 알고리즘을 이용하여 고속의 병렬 입-출력 모듈구조의 승산기를 구성하였다. 제시한 승산기의 구성은 $(m+1)^2$개의 동일한 기본 셀들로 설계되었으며, 기본 셀은 1개의 mod(3) 가산 게이트와 1개의 mod(3) 승산 게이트로 구성하였다. 셀에 래치를 사용하지 않았으므로 회로가 가장 간단하며, 셀당 지연시간도 $T_A + T_X$로서 가장 적다. 본 연구에서 제안한 승산기는 규칙성과 셀 배열에 의한 모듈성을 가지므로 m이 큰 회로의 확장이 용이하며 VLSI회로 실현에 적합할 것이다.

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 compose the multiplier with parallel input-output module structure using the presented multiplication algorithm. The proposed multiplier is designed $(m+1)^2$ same basic cells that have a mod(3) addition gate and a mod(3) multiplication gate. 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.

키워드

유한체, 승산알고리즘, 병렬승산기, 기약다항식

Key word

Finite fields, Multiplicative algorithm, Parallel multiplier, Primitive polynomials

* 정회원 : 상지대학교 컴퓨터정보공학부 (교신저자, choi-ys72@hanmail.net)
** 정회원 : 재능대학 컴퓨터정보과
*** 종신회원 : 상지대학교 컴퓨터정보공학부

접수일자 : 2010. 10. 25
심사완료일자 : 2011. 02. 17

Ⅰ. 서 론

유한체(Galois field)는 스위칭 이론, 디지털 신호처리, Reed-Solomon 부호기, 화상처리, 오류정정부호, 디지털 통신의 암호화 및 해독화를 요하는 보안 등에 널리 응용되고 있다. 특히 $GF(2^m)$은 신호처리와 화상처리분야에서 특별한 계산을 요하거나 범용컴퓨터 분야의 고속화를 보조하는 고성능컴퓨터 설계에 효과적이며, VLSI 설계에도 응용되고 있다[1,2]. 이들 중 오류정정부호의 경우 유한체 $GF(2^m)$상의 연산에서 실제로 부호기 및 복호기 설계 시 전체 시스템의 규모와 성능에 절대적인 영향을 미치므로 회로경로의 연결, 시스템 구조의 복잡성과 동시성 등의 문제점을 개선하기 위한 연구가 진행되어 왔다[3,4].

유한체에서 중요한 연산은 가산, 승산, 멱승, 제산, 승법적 역원 등이다. 가산은 매우 간단하여 유한체의 원소들이 다항식의 형태로 표현되는 경우 극히 간단한 회로로 구성할 수 있다. 반면에 승산은 암호화 및 해독화 알고리즘에 자주 사용되며, 제산과 멱승, 승법적 역원 등은 승산을 반복적으로 적용하여 수행할 수 있기 때문에 승산은 가장 중요하다. 따라서 회로의 저복잡성이 용이하게 실현될 수 있는 빠른 승산 알고리즘 개발이 중요하다.

최근 빠른 처리속도와 복잡도를 고려한 VLSI 구현에 있어 규칙성과 모듈화가 매우 중요시되면서 이에 적합한 승산기 설계에 관한 연구가 활발히 진행되고 있으며 꾸준히 발전하고 있다. 병렬 승산기 구조의 경우 회로는 복잡하지만 빠른 연산처리 능력을 가지고 있으므로 요즘 많이 연구 되고 있다. 또한 많은 계산량이 요구되는 승산은 작은 규모와 고성능을 가진 VLSI화에 적합하게 할 목적으로 연구되고 있으며, 이러한 이유로 기저를 달리하는 다양한 연산 방법들이 도입되고 있다. Yeh 등[5]은 표준기저를 사용하여 유한체 $GF(2^m)$상에서 $AB+C$ 연산을 수행하는 병렬 입-출력 시스토릭(systolic)구조의 승산기를 개발하였다. 이 승산기는 하나의 셀(cell)에 2개의 2 입력 AND 게이트와 2개의 2 입력 XOR 게이트를 사용하였다. 그러나 이 승산기는 VLSI화에는 적합하였으나 데이터가 역류하는 현상을 갖는다. Itoh 등[6]은 시스템의 복잡성을 줄이기 위하여 $GF(2^m)$상에서 다항식의 계수가 모두 1인 m차 기약 AOP(All One Polynomial)와 m차 기약 ESP(Equally Spaced Polynomial)를 기반으로 하는 모듈구조의 저복잡성 승산기를 설계하였다. Wang 등[7]은 Yeh 등이 제안한 시스토릭 승산기의 회로 복잡성을 개선하기 위하여 2개의 2 입력 XOR 게이트 대신에 1개의 3 입력 XOR 게이트를 사용하였다.

Halbutogullari 등[9]는 일반적인 기약다항식에 대한 Mastrovito 승산기를 제안하였다. 위에서 제안된 저복잡성 승산기들이 보안 및 암호 시스템 응용에 적합 할지라도 시스토릭 기술을 이용하여 설계된 것이 아닌 경우에는 m이 클 경우 $GF(2^m)$상의 승산에 대한 지연시간은 매우 크다.

Lee 등[10]은 유한체 $GF(2^m)$상에서 기약 AOP를 기반으로 하는 순환이동과 내적이라는 두 연산을 이용한 승산 알고리즘을 제안하였다. 그리고 그 알고리즘을 기반으로 저복잡성 비트-병렬 시스토릭 승산기를 구성하였다.

본 논문에서는 Lee 등이 제시한 AOP를 기반으로 하는 유한체 $GF(2^m)$상에서의 승산 알고리즘을 $GF(3^m)$상으로 확장하여 모든 항에 0이 아닌 계수가 존재하는 기약다항식의 두 원소에 대한 승산 알고리즘을 제안하였다. $GF(3^m)$상에서는 m이 홀수인 경우와 짝수인 경우에 대한 조건을 정리를 통하여 증명한 후 알고리즘을 구현하고, $GF(3^4)$상에서 승산 예를 들었으며 제시된 승산기는 $(m+1)^2$개의 동일한 셀로 구성하였다.

Ⅱ. $GF(3^m)$상에서의 승산 알고리즘

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

2.1 $GF(3^m)$의 연산

$GF(3)$은 $\{0, 1, 2\}$의 원소로 유한체를 구성하며, 이에 대한 가산과 승산에 대한 연산은 식 (1)과 식(2)이며, 이 연산에 의한 가산표 및 승산표를 표 1에서 보였다.

$$F(x, y) = (x + y) \bmod (3) \quad (1)$$

$$F(x, y) = (x \cdot y) \bmod (3) \quad (2)$$

표 1. $GF(3)$의 연산표 (a) 승산표 (b) 가산표
Table 1. Arithmetic table for $GF(3)$
(a) Multiplication table (b) Addition table

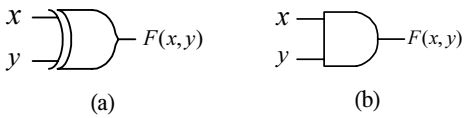
(a)

·	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

(b)

+	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

그림 1(a)는 식 (1)의 함수를 수행하는 2입력 mod(3) 가산 게이트이고, 그림 1(b)는 식 (2)의 함수를 수행하는 2입력 mod(3) 승산 게이트이다.



(a) (b)

그림 1. mod(3) 게이트.
(a) 가산 게이트 (b) 승산 게이트
Fig. 1. mod(3) Gate
(a) Multiplication gate (b) Addition gate

2.2 $GF(3^m)$상에서의 승산 알고리즘

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

2.2.1 $GF(3^m)$상에서 m이 홀수인 승산 알고리즘

유한체 $GF(3^m)$상에서 m이 홀수인 경우에 대한 승산 알고리즘을 제시하며, $GF(3^m)$은 m이 양의 정수인 3^m개의 원소를 갖는다.

[정리 1] 유한체 $GF(3^m)$상에서 모든 항이 존재하는 기약다항식이 식 (3)과 같이 표현될 때, $GF(3^m)$상에서 m이 홀수인 경우에 대한 승산 알고리즘이 성립한다.

$$F(x)=f_0+f_1x+f_2x^2+\cdots+f_{m-1}x^{m-1}+2x^m \quad (3)$$

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.

(증명) 식 (3)에서 최고차 항의 계수 2는 $GF(3^m)$상에서 식 (5)를 성립시키기 위한 계수이다. $F(x)=0$이므로 식 (4)와 같이 표현된다.

$$F(x)=f_0+f_1x+f_2x^2+\cdots+f_{m-1}x^{m-1}+2x^m=0 \quad (4)$$

식 (4)를 다시 쓰면 식 (5)와 같다.

$$-2x^m=f_0+f_1x+f_2x^2+\cdots+f_{m-1}x^{m-1} \quad (5)$$

식 (5) 좌변의 $-2x^m$에서 -2는 유한체 성질에 의해서 1과 같으므로 식 (6)과 같이 x^m의 식으로 나타낼 수 있다.

$$x^m=f_0+f_1x+f_2x^2+\cdots+f_{m-1}x^{m-1} \quad (6)$$

여기서 $f_i\in GF(3)$이다. 식 (6)과 같이 유한체 $GF(3^m)$상의 각 원소들은 차수가 $m-1$ 이하의 x의 다항식으로 표현된다. 두 다항식을 승산하였을 때, x^m보다 큰 차수들에 대하여 알아보기 위하여 먼저 x^{m+1}에 대한 식을 구하면 식 (7)과 같다.

$$\begin{aligned}x^{m+1}&=x^m\cdot x\\&=f_0x+f_1x^2+f_2x^3+\cdots+f_{m-2}x^{m-1}+f_{m-1}x^m\end{aligned} \quad (7)$$

식 (7)에 식 (6)을 대입하면 식 (8)과 같다.

$$\begin{aligned}x^{m+1}=&f_0x+f_1x^2+f_2x^3+\cdots+f_{m-2}x^{m-1}\\&+f_{m-1}(f_0+f_1x+f_2x^2+\cdots+f_{m-1}x^{m-1})\end{aligned} \quad (8)$$

정리하면 다음과 같다.

$$\begin{aligned}x^{m+1}=&f_0f_{m-1}+(f_0+f_1f_{m-1})x+(f_1+f_2f_{m-1})x^2\\&+\cdots+(f_{m-2}+f_{m-1}f_{m-1})x^{m-1}\end{aligned} \quad (9)$$

식 (9)에서 $x^{m+1}=1$이 되기 위해서는 식 (10)이 성립하여야 한다.

$$\begin{aligned}x^{m+1}=&f_0f_{m-1}+(f_0+f_1f_{m-1})x+(f_1+f_2f_{m-1})x^2\\&+\cdots+(f_{m-2}+f_{m-1}f_{m-1})x^{m-1}=1\end{aligned} \quad (10)$$

따라서 식 (10)을 만족하기 위한 각 항의 계수들을 나타내면 식 (11)과 같이 표현할 수 있다.

$$f_0 f_{m-1} = 1 \tag{11a}$$
$$f_0 + f_1 f_{m-1} = 0 \tag{11b}$$
$$f_1 + f_2 f_{m-1} = 0 \tag{11c}$$
$$\vdots$$
$$f_{m-2} + f_{m-1} f_{m-1} = 0 \tag{11d}$$

이제, [단계 1]~[단계 4]의 과정을 통하여 식 (3-11)이 성립되기 위한 $f_i \in GF(3)$ 인 f_0부터 f_{m-1}까지의 계수들을 구한다.

[단계 1] 식 (11a)에서 $f_0 f_{m-1} = 1$이 만족되기 위해서는 $f_0 = 1, f_{m-1} = 1$이어야 한다.

[단계 2] 식 (11b)에서 $f_0 + f_1 f_{m-1} = 0$이 되기 위해서는 단계 1에서 구한 $f_0 = 1, f_{m-1} = 1$ 을 대입하면 $f_1 = 2$이어야 한다.

[단계 3] 식 (11c)에서 $f_1 + f_2 f_{m-1} = 0$이 되기 위해서는 단계 2에서 구한 $f_1 = 2$를 대입하면 $f_2 = 1$이어야 한다.

[단계 4] 식 (11d)에서 $f_{m-2} + f_{m-1} f_{m-1} = 0$이 되기 위해서는 단계 1에서 구한 $f_{m-1} = 1$을 대입하여 구하면 $f_{m-2} = 2$이어야 한다.

그러므로 [단계 1]~[단계 4]에 의해서 구한 $f_0 \sim f_{m-1}$의 값을 정리하면 식 (12)와 같다.

$$f_0 = 1, f_1 = 2, f_2 = 1, f_3 = 2, \cdots, f_{m-2} = 2, f_{m-1} = 1 \tag{12}$$

따라서, x^{m+1}의 식 (9)는 상수 항 $f_0 f_{m-1}$만 1이고, 나머지 x항의 계수들은 모두 0이 되어 x^{m+1}은 식 (13)과 같이 된다.

$$x^{m+1} = x^m \cdot x = 1 \tag{13}$$

식 (13)을 이용하여 x^{m+2}, x^{m+3}, $\cdots$,x^{m+i}, $\cdots$, x^{2m}를 구하면 다음의 결과를 얻을 수 있다.

$$x^{m+2} = x^{m+1} \cdot x \tag{14a}$$
$$x^{m+3} = x^{m+2} \cdot x = x^2 \tag{14b}$$
$$\vdots$$
$$x^{m+i} = x^{m+i-1} \cdot x = x^{i-1} \tag{14c}$$
$$\vdots$$
$$x^{2m} = x^{m+m-1} \cdot x = x^{m-1} \tag{14d}$$

x가 유한체 $GF(3^m)$상에서 m차 기약 다항식의 근이라 할 때, $GF(3^m)$상의 두 원소인 승산 다항식 $A(x)$와 피승산 다항식 $B(x)$는 식 (15)와 같이 표현된다.

$$A(x) = \sum_{i=0}^{m} a_i x^i = a_0 + a_1 x + a_2 x^2 + \cdots + a_m x^m$$
$$B(x) = \sum_{i=0}^{m} b_i x^i = b_0 + b_1 x + b_2 x^2 + \cdots + b_m x^m \tag{15}$$

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

승산 알고리즘을 유도하기 위하여 다항식 $A(x)$, $B(x)$를 승산하면 식 (16)과 같다.

$$\begin{aligned} A(x) \cdot B(x) &= (a_0 + a_1 x + a_2 x^2 + \cdots + a_m x^m) \\ &\quad \cdot (b_0 + b_1 x + b_2 x^2 + \cdots + b_m x^m) \\ &= \left(\sum_{i=0}^{m} a_i x^i\right) \cdot \left(\sum_{i=0}^{m} b_i x^i\right) \end{aligned} \tag{16}$$

두 다항식의 승산식인 (16)을 $D(x)$로 놓으면, 식 (17)과 같이 표현할 수 있다.

$$\begin{aligned} D(x) &= d_0 + d_1 x + d_2 x^2 + \cdots + d_{2m} x^{2m} \\ &= \sum_{i=0}^{2m} d_i x^i \end{aligned} \tag{17}$$

식 (17)을 m차 항을 기준으로 2개의 항으로 나누어 다시 쓰면 식 (18)과 같다.

$$D(x) = \left(\sum_{i=0}^{m} d_i x^i\right) + \left(\sum_{i=m+1}^{2m} d_i x^i\right) \tag{18}$$

식 (18)의 두 번째 항 $\sum_{i=m+1}^{2m} d_i x^i$는 식 (13), (14)를 이용하여 $\sum_{i=0}^{m-1} d_{m+i+1} x^i$로 표현할 수 있으며, 식 (18)은 식 (19)와 같이 쓸 수 있다.

$$D(x)=\left(\sum_{i=0}^{m}d_i x^i\right)+\left(\sum_{i=0}^{m-1}d_{m+i+1}x^i\right) \tag{19}$$

식 (19)에서 x^m 항을 따로 빼서 다시 정리하면 식 (20)과 같이 쓸 수 있다.

$$\begin{aligned}D(x)&=\sum_{i=0}^{m-1}d_i x^i+d_m x^m+\sum_{i=0}^{m-1}d_{m+i+1}x^i\\&=\sum_{i=0}^{m-1}(d_i+d_{m+i+1})x^i+d_m x^m\end{aligned} \tag{20}$$

식 (20)에서 $d_i+d_{m+i+1}=D_i$, $d_m=D_m$ 이라 놓으면 다음과 같은 형태로 표현된다.

$$D(x)=\sum_{i=0}^{m-1}D_i x^i+D_m x^m=\sum_{i=0}^{m}D_i x^i \tag{21}$$

이상과 같이 유도된 승산 알고리즘을 $GF(3^m)$상에서 $m=5$인 경우로 적용한 예가 다음과 같다.

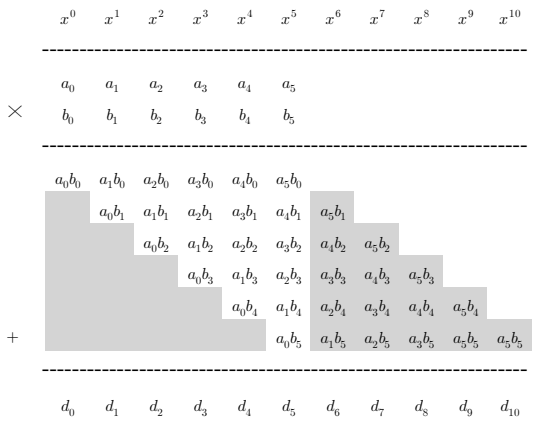
[예 1] $GF(3^m)$상에서 $m=5$인 경우의 승산 다항식 $A(x)$와 피승산 다항식 $B(x)$가 다음과 같이 표현될 때, $GF(3^5)$상에서 두 다항식 $A(x)$, $B(x)$를 승산하면 그림 2(a)와 같다.

$$\begin{aligned}A(x)&=a_0+a_1x+a_2x^2+a_3x^3+a_4x^4+a_5x^5\\B(x)&=b_0+b_1x+b_2x^2+b_3x^3+b_4x^4+b_5x^5\end{aligned}$$

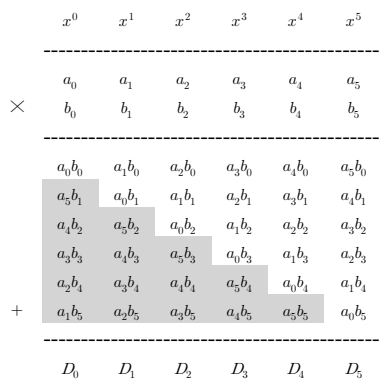
여기서, $a_i, b_i \in GF(3)$이다.

그림 2(a)에서 두 다항식 $A(x)$, $B(x)$의 승산을 정리하면 식 (22)과 같다.

$$\begin{aligned}d_0&=a_0b_0\\d_1&=a_1b_0+a_0b_1\\d_2&=a_2b_0+a_1b_1+a_0b_2\\d_3&=a_3b_0+a_2b_1+a_1b_2+a_0b_3\\d_4&=a_4b_0+a_3b_1+a_2b_2+a_1b_3+a_0b_4\\d_5&=a_5b_0+a_4b_1+a_3b_2+a_2b_3+a_1b_4+a_0b_5\\d_6&=a_5b_1+a_4b_2+a_3b_3+a_2b_4+a_1b_5\\d_7&=a_5b_2+a_4b_3+a_3b_4+a_2b_5\\d_8&=a_5b_3+a_4b_4+a_3b_5\\d_9&=a_5b_4+a_4b_5\\d_{10}&=a_5b_5\end{aligned} \tag{22}$$



(a)



(b)

그림 2. $GF(3^5)$상에서의 승산
(a) 단계 1 (b) 단계 2
Fig. 2. Multiplication over $GF(3^5)$
(a) Step 1 (b) Step 2

식 (14c)의 $x^{m+i}=x^{i-1}$로부터 $5<m\le 10$의 값을 변환하면, $i=1$인 경우 $x^6=x^0$, $i=2$인 경우 $x^7=x^1$, $i=3$인 경우 $x^8=x^2$, $i=4$인 경우 $x^9=x^3$, $i=5$인 경우 $x^{10}=x^4$이다.

따라서, x^6항 이상의 계수들은 $x^0\sim x^4$항의 계수들과 가산하여 그림 2(b)와 같이 구할 수 있으며, $m=5$이므로 식 (21)의 m에 5를 대입하여 $A(x)$, $B(x)$의 승산결과인 $D(x)$를 구하면 식 (23)과 같다.

$$\begin{aligned}D(x)&=\sum_{i=0}^{5}D_i x^i\\&=D_0+D_1x+D_2x^2+D_3x^3+D_4x^4+D_5x^5\end{aligned} \tag{23}$$

식 (23)에서 식 (20)과 식 (21)을 이용하여 다음과 같이 쓸 수 있다.

$$\begin{aligned} D_0 &= d_0 + d_6 \\ D_1 &= d_1 + d_7 \\ D_2 &= d_2 + d_8 \\ D_3 &= d_3 + d_9 \\ D_4 &= d_4 + d_{10} \\ D_5 &= d_5 \end{aligned} \tag{24}$$

식 (24)에 식 (22)를 대입하여 $D_0 \sim D_5$를 구하면 식 (25)와 같다.

$$\begin{aligned} D_0 &= a_0b_0 + a_5b_1 + a_4b_2 + a_3b_3 + a_2b_4 + a_1b_5 \\ D_1 &= a_1b_0 + a_0b_1 + a_5b_2 + a_4b_3 + a_3b_4 + a_2b_5 \\ D_2 &= a_2b_0 + a_1b_1 + a_0b_2 + a_5b_3 + a_4b_4 + a_3b_5 \\ D_3 &= a_3b_0 + a_2b_1 + a_1b_2 + a_0b_3 + a_5b_4 + a_4b_5 \\ D_4 &= a_4b_0 + a_3b_1 + a_2b_2 + a_1b_3 + a_0b_4 + a_5b_5 \\ D_5 &= a_5b_0 + a_4b_1 + a_3b_2 + a_2b_3 + a_1b_4 + a_0b_5 \end{aligned} \tag{25}$$

2.2.2 $GF(3^m)$상에서 m이 짝수인 승산 알고리즘

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

[정리 2] 유한체 $GF(3^m)$상에서 모든 항이 존재하는 기약다항식이 식 (3)과 같이 표현될 때, $GF(3^m)$상에서 m이 짝수인 경우에 대한 승산 알고리즘이 성립한다.

$$F(x) = f_0 + f_1x + f_2x^2 + \cdots + f_{m-1}x^{m-1} + 2x^m \tag{26}$$

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.

(증명) 두 다항식을 승산하였을 때, x^m보다 큰 차수들에 대하여 알아보기 위하여 먼저 x^{m+1}에 대한 식을 구하면 식 (8)과 식 (9)와 같다. 식 (9)를 다시 쓰면 식 (27)과 같다.

$$\begin{aligned} x^{m+1} = {} & f_0f_{m-1} + (f_0 + f_1f_{m-1})x + (f_1 + f_2f_{m-1})x^2 \\ & + \cdots + (f_{m-2} + f_{m-1}f_{m-1})x^{m-1} \end{aligned} \tag{27}$$

여기서, m이 짝수인 경우에 성립할 수 있는 조건을 구하기 위하여 $x^{m+1} = 2$로 놓으면, 식 (10)은 식 (28)과 같이 표현된다.

$$\begin{aligned} x^{m+1} = {} & f_0f_{m-1} + (f_0 + f_1f_{m-1})x + (f_1 + f_2f_{m-1})x^2 \\ & + \cdots + (f_{m-2} + f_{m-1}f_{m-1})x^{m-1} = 2 \end{aligned} \tag{28}$$

따라서 식 (28)을 만족하기 위해 각항의 계수들은

$$f_0f_{m-1} = 2 \tag{29a}$$
$$f_0 + f_1f_{m-1} = 0 \tag{29b}$$
$$f_1 + f_2f_{m-1} = 0 \tag{29c}$$
$$\vdots$$
$$f_{m-2} + f_{m-1}f_{m-1} = 0 \tag{29d}$$

이 되어야 한다. 이제 [단계 1]~[단계 4]의 과정을 통하여 식 (29)가 성립되기 위한 $f_i \in GF(3)$인 f_0부터 f_{m-1}까지의 계수들을 구한다.

[단계 1] 식 (29a)에서 $f_0f_{m-1} = 2$가 되기 위해서는 $f_0 = 2, f_{m-1} = 1$이어야 한다.

[단계 2] 식 (29b)에서 $f_0 + f_1f_{m-1} = 0$이 되기 위해서는 단계 1에서 구한 $f_0 = 2, f_{m-1} = 1$ 을 대입하면 $f_1 = 1$이어야 한다.

[단계 3] 식 (29c)에서 $f_1 + f_2f_{m-1} = 0$이 되기 위해서는 단계 2에서 구한 $f_1 = 1$을 대입하면 $f_2 = 2$이어야 한다.

[단계 4] 같은 방식으로 대입하여 구하면 $f_{m-2} = 2$이어야 한다.

[단계 1]~[단계 4]에 의해서 $f_0 \sim f_{m-1}$의 값을 정리하면 식 (30)과 같다.

$$f_0 = 2, f_1 = 1, f_2 = 2, f_3 = 1, \cdots, f_{m-2} = 2, f_{m-1} = 1 \tag{30}$$

따라서 x^{m+1}의 식 (28)은 상수 항 f_0f_{m-1}만 2이고, 나머지 x항의 계수들은 모두 0이 되어 x^{m+1}은 식 (31)과 같이 된다.

$$x^{m+1} = x^m \bullet x = 2 \tag{31}$$

식 (31)을 이용하여 x^{m+2}, x^{m+3}, …,x^{m+i}, …, x^{2m}를 구하면 다음과 같이 $GF(3^m)$상에서 m이 짝수인 경우는 홀수인 경우의 우변에 모두 2가 곱해진다.

$$x^{m+2} = x^{m+1} \bullet x = 2x \tag{32a}$$
$$x^{m+3} = x^{m+2} \bullet x = 2x^2 \tag{32b}$$
$$\vdots$$
$$x^{m+i} = x^{m+i-1} \bullet x = 2x^{i-1} \tag{32c}$$
$$\vdots$$
$$x^{2m} = x^{m+m-1} \bullet x = 2x^{m-1} \tag{32d}$$

두 다항식 식 (15)의 승산식인 식 (18)의 두 번째 항 $\sum_{i=m+1}^{2m} d_i x^i$는 식 (31)과 식 (32)를 이용하여 $\sum_{i=0}^{m-1} 2d_{m+i+1}x^i$로 표현할 수 있으며 식 (33)과 같이 나타낼 수 있다.

$$D(x) = \left(\sum_{i=0}^{m} d_i x^i\right) + \left(\sum_{i=0}^{m-1} 2d_{m+i+1}x^i\right) \tag{33}$$

식 (33)에서 x^m 항을 따로 빼서 다시 정리하면 식 (34)와 같이 쓸 수 있다.

$$\begin{aligned} D(x) &= \sum_{i=0}^{m-1} d_i x^i + d_m x^m + \sum_{i=0}^{m-1} 2d_{m+i+1}x^i \\ &= \sum_{i=0}^{m-1} (d_i + 2d_{m+i+1})x^i + d_m x^m \end{aligned} \tag{34}$$

식 (34)에서 $d_i + 2d_{m+i+1} = D_i$, $d_m = D_m$라 놓으면 식 (35)와 같이 $GF(3^m)$상에서 m이 홀수인 경우의 승산 알고리즘 승산식과 같은 형태로 표현된다.

$$D(x) = \sum_{i=0}^{m-1} D_i x^i + D_m x^m = \sum_{i=0}^{m} D_i x^i \tag{35}$$

이상과 같이 유도된 승산 알고리즘을 $GF(3^m)$상에서 $m=4$인 경우에 대하여 예를 들었다.

[예 2] $GF(3^m)$상에서 $m=4$인 경우의 승산 다항식 $A(x)$와 피승산 다항식 $B(x)$가 다음과 같이 표현될 때, $GF(3^4)$상에서 두 다항식 $A(x)$와 $B(x)$를 승산하면 그림 3과 같다.

$$A(x) = a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4$$
$$B(x) = b_0 + b_1 x + b_2 x^2 + b_3 x^3 + b_4 x^4$$

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

식 (32c)의 $x^{m+i} = 2x^{i-1}$로부터 x^5, x^6, x^7, x^8은 다음과 같이 x^0, x^1, x^2, x^3으로 변환할 수 있으며, $x^5 = 2x^0$, $x6 = 2x^1$, $x^7 = 2x^2$, $x^8 = 2x^3$로 변환된다.

따라서 x^5항 이상의 계수들은 $x^0 \sim x^3$항의 계수들과 가산하여 구할 수 있으며, $m=4$이므로 식 (35)의 m에 4를 대입하여 $A(x)$, $B(x)$의 승산결과인 $D(x)$를 구하면 식 (36)과 같다.

$$\begin{aligned} D(x) &= \sum_{i=0}^{4} D_i x^i \\ &= D_0 + D_1 x + D_2 x^2 + D_3 x^3 + D_4 x^4 \end{aligned} \tag{36}$$

	x^0	x^1	x^2	x^3	x^4	x^5	x^6	x^7	x^8
	a_0	a_1	a_2	a_3	a_4				
×	b_0	b_1	b_2	b_3	b_4				
	a_0b_0	a_1b_0	a_2b_0	a_3b_0	a_4b_0				
		a_0b_1	a_1b_1	a_2b_1	a_3b_1	a_4b_1			
			a_0b_2	a_1b_2	a_2b_2	a_3b_2	a_4b_2		
				a_0b_3	a_1b_3	a_2b_3	a_3b_3	a_4b_3	
+					a_0b_4	a_1b_4	a_2b_4	a_3b_4	a_4b_4
	d_0	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8

그림 3. $GF(3^4)$상에서의 승산
Fig. 3. Multiplication over $GF(3^4)$.

식 (36)에서 식 (34)와 식 (35)를 이용하여 다음과 같이 쓸 수 있다.

$$\begin{aligned} D_0 &= d_0 + 2d_5 \\ D_1 &= d_1 + 2d_6 \\ D_2 &= d_2 + 2d_7 \\ D_3 &= d_3 + 2d_8 \\ D_4 &= d_4 \end{aligned} \tag{37}$$

각 항의 계수인 $D_0 \sim D_4$를 구하면 식 (38)과 같다.

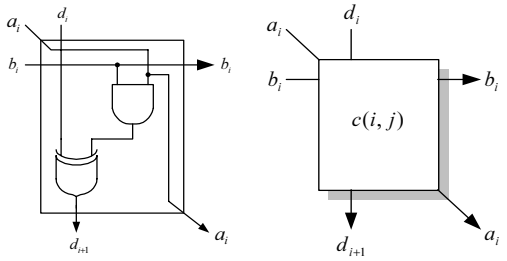
$$\begin{aligned} D_0 &= a_0b_0 + 2(a_4b_1 + a_3b_2 + a_2b_3 + a_1b_4) \\ D_1 &= a_1b_0 + a_0b_1 + 2(a_4b_2 + a_3b_3 + a_2b_4) \\ D_2 &= a_2b_0 + a_1b_1 + a_0b_2 + 2(a_4b_3 + a_3b_4) \\ D_3 &= a_3b_0 + a_2b_1 + a_1b_2 + a_0b_3 + 2a_4b_4 \\ D_4 &= a_4b_0 + a_3b_1 + a_2b_2 + a_1b_3 + a_0b_4 \end{aligned} \quad (38)$$

Ⅲ. $GF(3^m)$상의 승산기 구성

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

3.1 $GF(3^m)$상에서 m이 홀수인 승산기

앞장에서 제시한 $GF(3^m)$상에서의 승산 알고리즘에 대한 승산기를 구성한다. $GF(3^m)$상에서 m이 홀수일 경우의 승산 알고리즘은 $GF(2^m)$상의 승산 알고리즘을 그대로 적용할 수 있다. $GF(3^m)$상에서 m이 홀수일 경우의 승산기를 구성하기 위해서 먼저, 그림 1에서 보인 1개의 2입력 mod(3) 가산 게이트와 1개의 2입력 mod(3) 승산 게이트를 이용하여 기본 셀을 구성하였으며, 기본 셀의 회로와 기호를 그림 4에서 보였다.



(a) (b)

그림 4. $GF(3^5)$상에서 m이 홀수인 승산기의 기본 셀 (a) 회로 (b) 기호
Fig. 4. The basic cell of multiplier with odd m on $GF(3^5)$ (a) circuit (b) symbol

그림 4에서 셀의 회로는 식 (39)를 수행하며, a_i와 b_i는 각각 승산 다항식 $A(x)$와 피승산 다항식 $B(x)$의 계수들을 의미한다.

$$d_{i+1} = d_i \oplus (a_i \cdot b_i) \bmod(3) \quad (39)$$

그림 4에서 d_i는 셀의 입력으로서 앞단 셀의 출력이며 d_{i+1}은 셀의 출력을 의미한다. 그림 5는 기본 셀들을 이용하여 $GF(3^5)$상의 승산기를 구성한 회로이다. 그림 5에서 $a_0 \sim a_5$는 승산 다항식 $A(x)$에 대한 각 항의 계수들을 의미하며, $b_0 \sim b_5$는 피승산 다항식 $B(x)$에 대한 각 항의 계수들을 의미한다. 최하단의 $D_0 \sim D_5$는 승산결과에 대한 각 항의 계수이다. 최상단에 위치한 셀들에 입력되는 0은 셀에 대한 초기값이다.

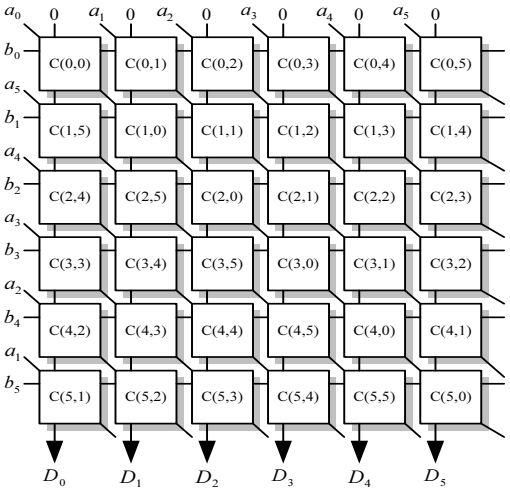
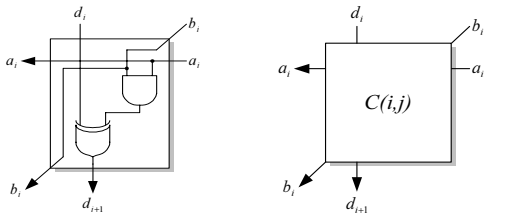


그림 5. $GF(3^5)$상의 제안된 승산기
Fig. 5. The proposed multiplier on $GF(3^5)$

3.2 $GF(3^m)$상에서 m이 짝수인 승산기

이절에서는 앞장에서 제시한 승산 알고리즘을 이용하여 m이 짝수일 때 $GF(3^m)$상의 승산기를 구성한다. 먼저 [예 2]에서 구한 $GF(3^4)$의 승산기를 구성하기 위하여 그림 1에서 보인 2입력 mod(3) 가산 게이트와 2입력 mod(3) 승산 게이트를 사용하여 그림 6과 같이 m이 짝수인 승산기의 기본 셀을 구성하였다. 그림 6의 기본 셀은 m이 홀수일 때의 그림 4와 비교해 볼 때 a_i와 b_i의 위

치가 바뀌면서 대각선의 방향이 우측에서 좌측으로 내려가는 형태임을 알 수 있다.



(a) (b)

그림 6. $GF(3^m)$상에서 m이 짝수인 승산기의 기본 셀 (a) 회로 (b) 기호
Fig. 6. The basic cell of the multiplier with even m on $GF(3^4)$ (a) circuit (b) symbol

그림 7은 m이 짝수인 경우에 대하여 제시된 승산 알고리즘을 이용하여 구현한 $GF(3^4)$에서의 승산기 구성도이다. 그림 7의 승산기에서 기본 셀 사이에 있는 4개의 "2"는 m이 짝수일 경우에 대한 승산 알고리즘을 구현할 때 3치에 따른 식 (38)에서의 값이다.

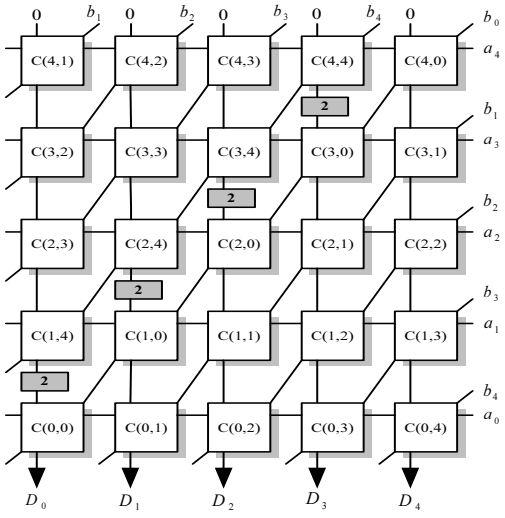


그림 7. $GF(3^4)$상의 제안된 승산기
Fig. 7. The proposed multiplier on $GF(3^4)$.

Ⅳ. 비교 및 검토

본 논문에서는 유한체 $GF(3^m)$상에서 모든 항의 계수가 0이 아닌 기약다항식의 두 원소를 승산하는 새로운 승산 알고리즘을 제시하였다. 제시된 승산 알고리즘을 이용하여 모듈 구조의 병렬 입-출력 승산기를 구성하였다. 표 2는 Yeh[5]와 Wang[7], Wei[8], 그리고 Lee[10] 등의 논문들과 승산기의 구성 형태, 각 셀당 사용된 게이트 수, 승산기에 사용된 전체 게이트 수, 그리고 셀당 지연시간, 전체 지연시간 등에 대하여 비교하였다.

셀당 게이트 수에서 mod(3) 승산 게이트의 수는 Yeh와 Wang의 논문에서는 2개, Wei의 논문에서는 3개, Lee와 본 논문에서는 1개가 사용되었다. 또한 전체 mod(3) 승산 게이트의 수는 m이 증가함에 따라 Yeh가 제시한 승산기는 $2m^2$으로 증가하고, Wang과 Wei의 승산기는 $4m^2$과 $3m^2$으로 증가하나 Lee와 본 논문의 승산기에서는 $(m+1)^2$로 증가한다.

셀당 mod(3) 가산 게이트의 수를 비교하면 Yeh의 승산기는 2 입력 mod(3) 가산 게이트 2개, Wang의 승산기는 3 입력 mod(3) 가산 게이트 1개, Wei의 승산기는 2 입력 mod(3) 가산 게이트 1개, 3 입력 mod(3) 가산 게이트 1개, Lee와 본 논문에서는 2 입력 mod(3) 가산 게이트 1개가 사용되었다.

또한 mod(3) 가산 게이트의 전체 갯수는 m이 증가함에 따라 Yeh의 승산기에서는 $2m^2$로 증가하고, Wang의 승산기에서는 m^2으로 증가하며, Wei가 제안한 승산기에서는 3입력 mod(3) 가산 게이트를 하나 더 사용하므로 $2m^2$으로 증가한다. Lee의 첫 번째 승산기와 본 논문의 승산기에서는 $(m+1)^2$으로 증가하며, Lee의 두 번째 승산기에서는 출력단에 1개씩 더 사용하므로 $(m+1)(m+2)$로 증가한다.

셀당 래치의 사용 갯수는 Yeh와 Wang의 승산기에서는 7개씩 사용되었으며, Wei의 승산기는 10개, Lee의 첫 번째 승산기에서는 3개, 두 번째 승산기에서는 4개가 사용되었다. 일반적으로 회로가 클럭에 의해 동작할 경우 회로 동작의 안정을 위하여 래치를 사용하나, 본 논문에서는 승산결과를 얻기 위하여 최종결과가 출력될 때까지를 하나의 클럭으로 동작한다.

셀당 지연시간은 Yeh와 Wei, 그리고 Lee의 첫 번째 논

문에서는 $T_A + T_X + 2T_L$이며, Wang과 본 논문의 승산기에서는 $T_A + T_X$이며, Lee의 두 번째 논문에서는 $T_A + T_L$이다. 전체 지연시간에 있어서는 Yeh와 Wang, 그리고 Wei의 승산기는 $3m$시간이 필요하며, Lee와 본 논문의 승산기는 $m+1$의 전체 지연시간이 필요하다.

표 2. 병렬 승산기의 비교
Table 2. The comparison for parallel multipliers.

승산기 / 항목	Yeh[5]	Wang[7]	Wei[8]	Lee[10](1)	Lee[10](2)	본 논문
셀당 게이트 수(개)	2	2	3	1	1	1
2 입력 AND 게이트	2	0	1	1	1	1
2 입력 XOR 게이트	0	1	1	0	0	0
3 입력 XOR 게이트	7	7	10	3	4	0
1 비트 래치						
전체 게이트 수(개)	$2m^2$	$4m^2$	$3m^2$	$(m+1)^2$	$(m+1)^2$	$(m+1)^2$
2 입력 AND 게이트	$2m^2$	0	m^2	$(m+1)^2$	$(m+1)(m+2)$	$(m+1)^2$
2 입력 XOR 게이트	0	m^2	m^2	0	0	0
3 입력 XOR 게이트	$7m^2$	$7m^2$	$10m^2$	$\approx 4(m+1)^2$	$\approx 5(m+1)^2$	0
1 비트 래치						
셀당 지연시간	$T_A+T_X+2T_L$	T_A+T_X	$T_A+T_X+2T_L$	$T_A+T_X+T_L$	T_X+T_L	T_A+T_X
전체 지연시간	3m	3m	3m	m+1	m+1	m+1

[주] TA = 2 입력 AND 게이트의 지연시간
TX = 2 입력 XOR 게이트의 지연시간
T3X = 3 입력 XOR 게이트의 지연시간
TL = 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 승산 알고리즘을 제시하였으며, 제시된 승산 알고리즘을 이용하여 고속의 병렬 입-출력 모듈구조의 승산기를 구성하였다.

본 논문에서 제안한 $GF(3^m)$상의 승산 알고리즘은 m이 홀수 및 짝수인 조건에 따라 $x^{m+1} = 3-1$을 만족하는 기약다항식의 승산 알고리즘을 이용하여 두 원소들의 승산결과를 얻을 수 있음을 보여 준다. 승산기의 구성은 $(m+1)^2$개의 동일한 셀로 설계되었으며, 기본 셀은 1개의 2입력 mod(3) 가산 게이트와 1개의 2입력 mod(3) 승산 게이트로 구성하였다. $GF(3^m)$상에서 m이 짝수인 경우의 승산기는 회로 구성에 있어서 m이 홀수인 경우와 비교해 볼 때, 승산기의 해당하는 셀 아래 위치에 2값을 곱해 주어야 하므로 m이 홀수인 경우와는 다르게 대각선 연결선의 방향은 우측에서 좌측으로 내려가는 방향으로 구성하였다.

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

참고문헌

[1] 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.

[2] C. C. Wang, T. K. Truong, H. M. Shao, L. J. Deutsch, J. K. Omuro, and I. S. Reed, "VLSI Architecture for Computing Multiplications and Inverses in $GF(2^m)$," *IEEE Trans. Computers,* vol. C-34, pp. 707-717, Aug. 1985.

[3] P. E. Scott, S. E. Tavares, L. E. Peppard, "A Fast VLSI Multiplier for $GF(2^m)$," *IEEE Journal Selected Areas in Comunications, SAL-4,* no. 1, pp. 62-65, Jan. 1986.

[4] N. Iliev, J. E. Stine and N. Jachimiec, " Parallel Programmable Finite Field $GF(2^m)$ Multiplier," in Proc. *IEEE Int. Symp. VLSI Emerging Trends in VLSI Systems Design,* Sept. 2004.

[5] C. S. Yeh, I. S. Reed, and T. K. Truong, "Systolic Multipliers for Finite Fields $GF(2^m)$," *IEEE Trans. Computers,* vol. 33, no. 4, pp. 357-360, Apr. 1984.

[6] T, Itoh and S. Tsujii, "Structure of Parallel Multipliers for a Class of Fields $GF(2^m)$," Inform. Comp., vol. 83, pp. 21-40, 1989.

[7] 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.

[8] S. W. Wei, "A Systolic Power-Sum Circuit for $GF(2^m)$," *IEEE Trans. Computers,* vol. 43, no. 2, pp. 226-229, Feb. 1994.

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

[10] 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.

최용석 (Yong-Seok Choi)

2002년 상지대학교 전산학과 이학사
2004년~2006년 (주)인터스피아 기술 이사
2007년 상지대학교 컴퓨터정보공학과 공학석사
2009년~현재 상지대학교 컴퓨터정보공학과 박사 수료 (주)명진 전산 과장
※관심분야 : 바이오인식, 유비쿼터스 컴퓨팅 보안, Cryptography Theory & Security, RFID/WSN 설계 및 응용 등

박승용 (Seung-Yong Park)

1979년 인하대학교 전자공학과 졸업
1982년 인하대학교 대학원 전자공학과 공학석사
2002년 인하대학교 대학원 정보공학 공학박사
1985 ~ 현재 : 재능대학 컴퓨터정보과 교수
※관심분야 : 컴퓨터시스템, 컴퓨터네트워크, 컴퓨터 운영관리

성현경 (Hyeon-Kyeong Seong)

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