

	(19) 대한민국특허청(KR) (12) 공개특허공보(A)	(11) 공개번호 10-2009-0113138 (43) 공개일자 2009년10월29일
(51) Int. Cl. <i>G06F 7/52</i> (2006.01) <i>G06F 7/57</i> (2006.01) (21) 출원번호 10-2008-0039068 (22) 출원일자 2008년04월26일 심사청구일자 2008년04월26일	(71) 출원인 고려대학교 산학협력단 서울 성북구 안암동5가1 고려대학교 내 (72) 발명자 홍석희 서울 은평구 응암2동 451-12 202호 임종인 서울 종로구 내수동 95 삼성파크팰리스 101-901 (뒷면에 계속) (74) 대리인 현중철	

전체 청구항 수 : 총 12 항

(54) MSD first GF (3^m) 직렬 곱셈 장치, 그방법 및 이를 기록한 기록매체

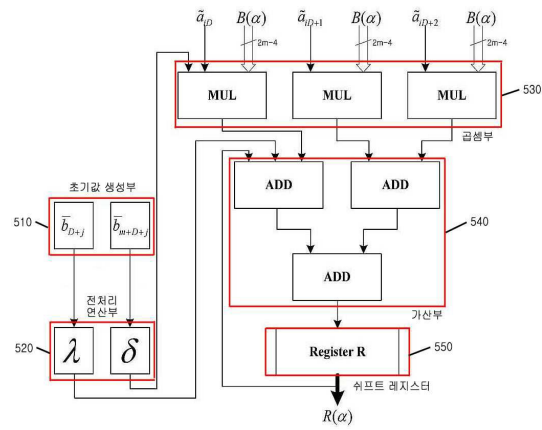
(57) 요약

MSD first GF(3^m) 직렬 곱셈 장치, 그 방법 및 이를 기록한 기록매체가 개시된다.

본 발명에 따른 MSD first GF(3^m) 직렬 곱셈 장치는, GF(3) 상에서 차수가 *m*인 삼항 기약 다항식 $f(x) = x^m + f_t \cdot x^t + f_0$ 을 사용하며, 상기 $f(x)$ 의 해가 α 일 때, 유한체 상의 두 원소 $A(\alpha)$, $B(\alpha)$ 의 곱셈을 수행하여 직렬 곱셈 결과값 $R(\alpha)$ 를 생성하는 GF(3^m)기반의 MSD first 디지털 직렬 곱셈 장치에 있어서, 루프 인덱스 *i*가 $\lceil m/D \rceil + 2$ 일 때, *D*개의 디지털의 인덱스를 *j*라 하면, $b(\alpha)$ 의 계수 b_{t+1} , b_{t+j} , 상기 삼항 기약 다항식의 계수로부터 생성되는 $-f_t$, 및 상기 삼항 기약 다항식의 계수에 따라 생성되는 $-f_t f_0$ 를 입력으로 하여 $\overline{b_{m+D+j}}$ 과 $\overline{b_{D+j}}$ 을 생성하고, 상기 $b(\alpha)$ 를 이용하여 $\overline{b(\alpha)}$ 를 생성하는 초기값 생성부; 루프 인덱스 *i*가 $\lceil m/D \rceil + 1$ 일 때, 상기 연산된 $\overline{b_{D+j}}$ 를 이용하여 제 1 치환 연산자 λ 를 생성하고, 상기 연산된 $\overline{b_{m+D+j}}$ 를 이용하여 제 2 치환 연산자 δ 를 생성하는 전처리 연산부; 상기 전처리 연산부에서 생성된 제 1 치환 연산자 δ , 상기 $\tilde{A}(\alpha)$ 의 계수 $\tilde{a}_{Di+j}$, $-f_t$, $-f_0$ 및 $B(\alpha)$ 를 입력으로 하여 $\tilde{a}_{Di+j} \overline{b(\alpha)}$ 의 결과값 $v(\alpha)$ 을 생성하는 곱셈부; 이전 루프의 MSD first 디지털 직렬 곱셈의 결과값 $R(\alpha)$, 상기 제 2 치환 연산자 λ , 및 상기 곱셈부의 결과값을 입력으로 하여 덧셈 결과값을 생성하는 가산부; 및 상기 덧셈 결과값을 라이트 쉬프트하여 상기 가산부에 피드백하는 쉬프트 레지스터를 포함한다.

본 발명에 의하면, 삼항 기약 다항식이 적용되는 페어링 기반의 암호 시스템에 있어서, 기존의 MSD-first 곱셈기에 비하여 적은 시간 지연을 가지고, 표수에 의존하지 않고, 기존의 유한체 곱셈기에 비하여 고속 동작하면서 이때 증가하는 공간 복잡도를 최소화할 수 있고, 이에 따라 페어링 기반의 암호 시스템을 더욱 효율적으로 설계할 수 있으며, 특히 삼항 기약 다항식을 사용하는 모든 유한체에 적용할 수 있기 때문에 이를 기반으로 하는 하드웨어 설계에 응용할 수 있는 효과가 있다.

대표도 - 도5



(72) 발명자

김창한

서울시 강남구 대치동 은마아파트 21-507

장남수

서울 마포구 성산2동 200-84

김태현

서울 강서구 화곡2동 846-19

특허청구의 범위

청구항 1

$GF(3)$ 상에서 차수가 m 인 삼항 기약 다항식 $f(x) = x^m + f_t \cdot x^t + f_0$ 을 사용하며, 상기 $f(x)$ 의 해가 α 일 때, 유한체 상의 두 원소 $A(\alpha)$, $B(\alpha)$ 의 곱셈을 수행하여 직렬 곱셈 결과값 $R(\alpha)$ 를 생성하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 장치에 있어서,

루프 인덱스 i 가 $\lceil m/D \rceil + 2$ 일 때, D 개의 디지털의 인덱스를 j 라 하면, $b(\alpha)$ 의 계수 b_{t+1} , b_{t+j} , 상기 삼항 기약 다항식의 계수로부터 생성되는 $-f_t$, 및 상기 삼항 기약 다항식의 계수에 따라 생성되는 $-f_t f_0$ 를 입력으로 하여 $\overline{b_{m+D+j}}$ 과 $\overline{b_{D+j}}$ 을 생성하고, 상기 $b(\alpha)$ 를 이용하여 $\overline{b(\alpha)}$ 를 생성하는 초기값 생성부;

루프 인덱스 i 가 $\lceil m/D \rceil + 1$ 일 때, 상기 연산된 $\overline{b_{D+j}}$ 를 이용하여 제 1 치환 연산자 λ 를 생성하고, 상기 연산된 $\overline{b_{m+D+j}}$ 를 이용하여 제 2 치환 연산자 δ 를 생성하는 전처리 연산부;

상기 전처리 연산부에서 생성된 제 1 치환 연산자 δ , 상기 $\tilde{A}(\alpha)$ 의 계수 $\tilde{a}_{iD+j}$, $-f_t$, $-f_0$ 및 $B(\alpha)$ 를 입력으로 하여 $\tilde{a}_{Di+j} \overline{b(\alpha)}$ 의 결과값 $v(\alpha)$ 을 생성하는 곱셈부;

이전 루프의 MSD first 디지털 직렬 곱셈의 결과값 $R(\alpha)$, 상기 제 2 치환 연산자 λ , 및 상기 곱셈부의 결과값을 입력으로 하여 덧셈 결과값을 생성하는 가산부; 및

상기 덧셈 결과값을 라이트 쉬프트하여 상기 가산부에 피드백하는 쉬프트 레지스터를 포함하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 장치.

청구항 2

제 1 항에 있어서,

상기 초기값 생성부는

하기의 식 1에 따라 상기 $b(\alpha)$ 를 이용하여 $\overline{b(\alpha)}$ 를 생성하는 것을 특징으로 하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 장치.

$$\overline{B(\alpha)} = B(\alpha) - \left(\sum_{j=0}^{D-1} b_{t+D+j} \cdot f_t \cdot F(\alpha) \alpha^{D+j} \right) \quad (1)$$

청구항 3

제 1 항에 있어서,

상기 초기값 생성부는

디지털 D 의 인덱스를 j 라 하면, $b(\alpha)$ 의 계수 b_{t+1} , b_{t+j} , 상기 삼항 기약 다항식의 계수로부터 생성되는 $-f_t$, 및 $-f_t f_0$ 에 따라 연산되는 제 1 전처리 연산자 t_j^1 에 따라 하기의 식 2에 의해 $\overline{b_{m+D+j}}$ 과 $\overline{b_{D+j}}$ 을

생성하는 것을 특징으로 하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 장치.

$$\begin{aligned} \overline{b_{m+j}} &\leftarrow b_{t+1} \cdot (-f_t), \\ \overline{b_j} &\leftarrow t_j^1 + b_j, \text{ where } t_j^1 \leftarrow b_{t+j} \cdot (-f_0) \end{aligned} \quad (2)$$

청구항 4

제 3 항에 있어서,

상기 전처리 연산부는

하기의 식 3에 의해 제 2 전처리 연산자 t_j^2 및 제 3 전처리 연산자 t_j^3 를 생성하는 것을 특징으로 하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 장치.

$$t_j^2 \leftarrow \overline{b_{m+j}} \cdot \tilde{a}_{D(i-2)+j}, \quad t_j^3 \leftarrow \overline{b_j} \cdot \tilde{a}_{D(i-2)+j} \quad (3)$$

청구항 5

제 4 항에 있어서,

상기 전처리 연산부는

하기의 식 4에 의해 상기 제 1 치환 연산자 λ 및 제 2 치환 연산자 δ 를 생성하는 것을 특징으로 하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 장치.

$$\delta_j \leftarrow t_{j+D}^2 + r_{m-D+j}, \quad \lambda_j \leftarrow t_j^3 + v_0 \quad (4)$$

청구항 6

제 1 항에 있어서,

상기 가산부는

이전 루프의 MSD first 디지털 직렬 곱셈의 결과값 $R(\alpha)$ 및 상기 $A(\alpha)$ 의 계수 a_{Di+j} 와 상기 $B(\alpha)$ 의 곱에 의해 생성되는 중간 연산값 v_j 를 이용하여 하기의 식 5에 따라 덧셈 연산의 결과값을 생성하는 것을 특징으로 하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 장치.

$$R(\alpha) \leftarrow R(\alpha) + \sum_{j=1}^{D-1} v_j \quad (5)$$

청구항 7

$GF(3)$ 상에서 차수가 m 인 삼항 기약 다항식 $f(x) = x^m + f_t \cdot x^t + f_0$ 을 사용하며, 상기 $f(x)$ 의 해가 α 일 때, 유한체 상의 두 원소 $A(\alpha)$, $B(\alpha)$ 의 곱셈을 수행하여 직렬 곱셈 결과값 $R(\alpha)$ 를 생성하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 방법에 있어서,

루프 인덱스 i 가 $\lceil m/D \rceil + 2$ 일 때, D 개의 디지털의 인덱스를 j 라 하면, $b(\alpha)$ 의 계수 b_{t+1} , b_{t+j} , 상

기 삼항 기약 다항식의 계수로부터 생성되는 $-f_t$, 및 상기 삼항 기약 다항식의 계수에 따라 생성되는 $-f_t f_0$ 를 입력으로 하여 $\overline{b_{m+D+j}}$ 과 $\overline{b_{D+j}}$ 을 생성하고, 상기 $b(\alpha)$ 를 이용하여 $\overline{b(\alpha)}$ 를 생성하는 초기값 생성 단계;

루프 인덱스 i 가 $\lceil m/D \rceil + 1$ 일 때, 상기 연산된 $\overline{b_{D+j}}$ 를 이용하여 제 1 치환 연산자 λ 를 생성하고, 상기 연산된 $\overline{b_{m+D+j}}$ 를 이용하여 제 2 치환 연산자 δ 를 생성하는 전처리 연산 단계;

상기 전처리 연산부에서 생성된 제 1 치환 연산자 δ , 상기 $\tilde{A}(\alpha)$ 의 계수 $\tilde{a}_{iD+j}$, $-f_t$, $-f_0$ 및 $B(\alpha)$ 를 입력으로 하여 $\tilde{a}_{Di+j} \overline{b(\alpha)}$ 의 곱셈 결과값 $v(\alpha)$ 을 생성하는 곱셈 결과값 생성 단계;

이전 루프의 MSD first 디지털 직렬 곱셈의 결과값 $R(\alpha)$, 상기 제 2 치환 연산자 λ , 및 상기 곱셈부의 결과값을 입력으로 하여 덧셈 결과값을 생성하는 덧셈 결과값 생성 단계; 및

상기 덧셈 결과값을 라이트 쉬프트하여 상기 가산부에 피드백하는 피드백 단계를 포함하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 방법.

청구항 8

제 7 항에 있어서,

상기 초기값 생성 단계는

하기의 식 6에 따라 상기 $b(\alpha)$ 를 이용하여 $\overline{b(\alpha)}$ 를 생성하는 것을 특징으로 하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 방법.

$$\overline{B(\alpha)} = B(\alpha) - \left(\sum_{j=0}^{D-1} b_{t+D+j} \cdot f_t \cdot F(\alpha) \alpha^{D+j} \right) \quad (6)$$

청구항 9

제 7 항에 있어서,

상기 초기값 생성 단계는

디지털 D 의 인덱스를 j 라 하면, $b(\alpha)$ 의 계수 b_{t+1} , b_{t+j} , 상기 삼항 기약 다항식의 계수로부터 생성되는 $-f_t$, 및 $-f_t f_0$ 에 따라 연산되는 제 1 전처리 연산자 t_j^1 에 따라 하기의 식 7에 의해 $\overline{b_{m+D+j}}$ 과 $\overline{b_{D+j}}$ 을 생성하는 것을 특징으로 하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 방법.

$$\begin{aligned} \overline{b_{m+j}} &\leftarrow b_{t+1} \cdot (-f_t), \\ \overline{b_j} &\leftarrow t_j^1 + b_j, \text{ where } t_j^1 \leftarrow b_{t+j} \cdot (-f_t f_0) \end{aligned} \quad (7)$$

청구항 10

제 9 항에 있어서,

상기 전처리 연산 단계는

하기의 식 8에 의해 제 2 전처리 연산자 t_j^2 및 제 3 전처리 연산자 t_j^3 를 생성하는 것을 특징으로 하는

$GF(3^m)$ 기반의 MSD first 디지털 곱셈 방법.

$$t_j^2 \leftarrow \overline{b}_{m+j} \cdot \tilde{a}_{D(i-2)+j}, \quad t_j^3 \leftarrow \overline{b}_j \cdot \tilde{a}_{D(i-2)+j} \quad (8)$$

청구항 11

제 10 항에 있어서,

상기 전처리 연산 단계는

하기의 식 9에 의해 상기 제 1 치환 연산자 λ 및 제 2 치환 연산자 δ 를 생성하는 것을 특징으로 하는 $GF(3^m)$ 기반의 MSD first 디지털 곱셈 방법.

$$\delta_j \leftarrow t_{j+D}^2 + r_{m-D+j}, \quad \lambda_j \leftarrow t_j^3 + v_0 \quad (9)$$

청구항 12

제 7 항 내지 제 11 항 중 어느 한 항의 방법을 컴퓨터에서 수행할 수 있도록 프로그램으로 기록된 기록 매체.

명세서

발명의 상세한 설명

기술 분야

- <1> 본 발명은 유한체 상의 연산기에 관한 것으로서, 특히 $GF(3^m)$ 유한체 상에서 삼항 기약다항식을 사용하여 페어링 기반의 암호시스템을 더욱 효율적으로 설계함으로써 공간을 줄인 고속 연산 시스템을 구축하고, 암호 시스템의 암호화, 복호화 및 전자서명을 빠른 속도로 수행할 수 있는 MSD first $GF(3^m)$ 디지털 곱셈 장치, 그 방법 및 이를 기록한 기록매체를 제공하는 것이다.

배경 기술

- <2> 최근 유비쿼터스 환경에서 센서 네트워크 기술이 IT839 정책의 일환으로 활성화되고 있으며, 이러한 센서 네트워크는 관리자의 부재와 무선 통신, 자원의 제약 등으로 인하여 공격자가 쉽게 접근하여 공격을 시도할 수 있다.
- <3> 한편, 암호서비스에 대한 요구가 많아지면서 대칭키 알고리즘에서 발생한 키 관리 문제와 인증 문제를 해결하기 위한 알고리즘이 많이 연구되었다.
- <4> 1976년 Diffie와 Hellman이 처음으로 공개키 암호 알고리즘의 개념을 소개하였다. 그 후 많은 알고리즘이 제안되었지만, 1978년 RSA가 소개되어 널리 사용되고 있다.
- <5> 타원 곡선은 1985년 Koblitz와 Miller에 의해 발표된 이후 꾸준히 연구되어 왔다.
- <6> 타원 곡선 암호가 발표되었을 당시에는 새로운 암호에 대한 안전성 분석이 충분히 이루어지지 않았으므로 널리 사용되지 못했다.
- <7> 이 후, 많은 연구자들에 의한 연구 결과에 따르면 같은 안전도에서 비교할 때, RSA에 비해 훨씬 빠르고 효율적이며 짧은 키 길이를 가지는 장점이 있다.
- <8> 예를 들면, 타원 곡선 암호의 160비트는 RSA의 1024비트와 같은 안전도를 가진다. 이에 따라서 타원 곡선 암호는 무선 인터넷이나 스마트카드 등 제한된 연산능력을 가지는 하드웨어에 적합한 암호 시스템으로 주목받게 되었다.
- <9> 한편, 다양한 차수 m 을 사용하는 상호 간의 거래에 있어서 Versatile 프로세서를 사용함으로써 호환성을 높

일 수 있다.

- <10> 많은 암호시스템에서는 유한체 $GF(2^m)$ 에서의 연산을 사용한다. 이는 일반적인 컴퓨터 상에서는 $GF(2^m)$ 의 성능이 $GF(p)$ 의 성능을 능가하기 때문이다.
- <11> 실제로, 유한체에서의 연산을 더욱 빠르게 하기 위하여, 많은 프로세서가 제안되고 있다. 그러나, 대부분 고정된 유한체에서의 연산을 다루고 있고, 이 경우에는 체의 변화나 파라미터 변화가 있을 경우 새로운 프로세서가 필요하게 된다.
- <12> 그러나, 최근 페어링(pairing)의 곱선 형성(bilinearity)과 같은 특성들이 새로운 암호시스템에 사용되면서 Weil 페어링과 Tate 페어링 등의 페어링의 효율성에 대한 관심과 연구가 증가하고 있다.
- <13> 페어링을 효율적으로 연산하기 위한 알고리즘은 Miller에 의해 처음 제안되었다.
- <14> 그 이후, Barreto와 3인의 "Efficient algorithms for pairing-based cryptosystems, CRYPTO 2002, LNCS 2442, pp.354-368, Springer-Verlag, 2002"와 Galbraith 외 2인의 "Implementing the Tate pairing, ANTS V, LNCS 2369, pp.324-337, Springer-Verlag, 2002"에서 표수가 2인 유한체 위에서 정의된 초특이 타원 곡선(supersingular elliptic curve)에서 Tate 페어링의 연산 속도를 향상시킬 수 있는 방법을 제안하였다.
- <15> 그리고, Duursma와 Lee는 표수가 3인 경우에 닫힌 공식(closed formula)을 제안하였고, S. Kwon은 "Efficient Tate pairing computation for elliptic curves over binary fields, ACISP 2005, LNCS 3574, pp.134-145, Springer-Verlag, 2005"에서 표수가 2인 초특이 타원 곡선으로 Duursma와 Lee의 방법을 확장하였다.
- <16> 한편, Barreto와 3인은 "Efficient Pairing Computation on Supersingular Abelian Varieties," Designs, Codes and Cryptography, Vol.42, No.3, pp.239-271, 2007"에서 초특이 abelian variety에서 페어링을 효율적으로 연산하기 위한 일반적인 접근 방법인 η_T 페어링을 제안하였다.
- <17> 한편, 최근 페어링을 구현하기 위한 많은 방법들이 제안되었다. 페어링 구현은 소프트웨어뿐만 아니라 하드웨어에서 다양한 방법으로 진행되고 있지만, 병렬기법과 파이프라이닝(pipelining) 기법 등의 속도를 향상시킬 수 있는 다양한 방법이 있기 때문에 하드웨어 구현에 대한 연구가 좀 더 활발하게 진행되고 있다.
- <18> 실제로, $GF(p)$ 와 $GF(2^m)$ 에 대한 하드웨어 연구는 많이 진행되었다. 그러나, 페어링 기반의 암호 시스템의 효율성을 높이기 위한 Duursma-Lee의 알고리즘과 η_T 페어링 등의 사용으로 인하여 유한체 $GF(3^m)$ 에 대한 하드웨어 구현 방법이 필요하게 되었다.
- <19> $f(x) = x^m + f_t \cdot x^t + f_0$ 를 $GF(3)$ 위에서 차수가 m 인 삼항 기약 다항식이라고 하고, α 를 $f(x)$ 의 해라 하자.
- <20> 그러면, 유한체 $GF(3^m)$ 은 그러면, 유한체 $GF(3^m)$ 은 $GF(3)[X]/f(x)$ 와 동형이기 때문에 $GF(3^m)$ 의 원소 $a(\alpha)$ 는 다항식 기저를 이용하여 하기의 수학식 1과 같이 표현할 수 있다.

수학식 1

<21>
$$A(\alpha) = a_{m-1}\alpha^{m-1} + \dots + a_1\alpha + a_0 = (a_{m-1} \dots a_1 a_0), \text{ where } a_i \in GF(3)$$

<22> 그리고, α 는 $f(x)$ 의 해이므로 $f(\alpha) = 0$ 에 의하여 하기의 수학식 2를 구성할 수 있다.

수학식 2

<23>
$$\alpha^m = -f_t\alpha^t - f_0$$

<24> 그리고, $A(\alpha), B(\alpha) \in GF(3^m)$ 라 하면, $R(\alpha) = A(\alpha) + B(\alpha)$ 는 하기의 수학식 3과 같이 구성할 수 있고, $a_i + b_i$ 는 $GF(3)$ 의 덧셈 연산이다.

수학식 3

<25>
$$R(\alpha) = A(\alpha) + B(\alpha) = \sum_{i=0}^{m-1} (a_i + b_i) \alpha^i$$

<26> 이하, 기존의 디지털 직렬 곱셈기(Digit Serial multiplier)에 대하여 기술한다. $GF(3^m)$ 의 곱셈은 두 원소의 캐리 전파가 없는 다항식 곱셈과 주어진 삼항 기약 다항식 $F(x)$ 를 이용한 모듈러 연산으로 구성된다.

<27> 여기서는, 일반적인 페어링 기반의 암호 시스템에서 사용하는 삼항 기약 다항식 $f(x) = x^m + f_t x^t + f_0$ 만을 고려한다.

<28> 디지털(Digit) 단위의 연산은 한번의 프로세스에 D 개의 계수를 처리하므로, 비트 단위 연산보다 고속 연산 수행이 가능하다.

<29> 상기 디지털의 크기를 D 라고 하면, $GF(3^m)$ 의 원소 $A(\alpha)$ 는 하기의 수학식 4와 같이 표현된다.

수학식 4

<30>
$$A(\alpha) = a_{\lfloor m/D \rfloor - 1} \alpha^{\lfloor m/D \rfloor - 1} + \dots + a_1 \alpha + a_0$$

$$= (a_{\lfloor m/D \rfloor - 1} \dots a_1 a_0), \text{ where } a_i \in GF(3).$$

<31> 이를 기반으로 우선, 기존에 제안된 [Low energy digit-serial/parallel finite field multiplier]에서의 MSD-first 디지털 직렬 곱셈기에 대하여 살펴보기로 한다.

<32> 이를 타입 1(Type 1)의 MSD-first 디지털 직렬 곱셈기라 하면, 이 곱셈기는 전체적으로 $\lceil m/D \rceil$ 클럭 사이클을 수행하며 i 번째 반복문에서 부분곱 D 개의 합은 하기의 수학식 5과 같이 표현된다.

수학식 5

<33>
$$U(\alpha) = \sum_{j=0}^{D-1} a_{Di+j} \cdot B(\alpha) \cdot \alpha^j \bmod F(x)$$

<34> 상기 수학식 5에서 연산된 $U(\alpha)$ 는 $m-1$ 차 다항식이다.

<35> 한편, 상기 수학식 5에 따라 연산된 $U(\alpha)$ 와 누적값 $R(\alpha)$ 를 더하여 MSD-first 디지털 직렬 곱셈의 결과값을 생성한다.

<36> 이와 같은 MSD-first 디지털 직렬 곱셈을 정리하면, 하기의 표 1에 따른 알고리즘 1과 같다.

표 1

Algorithm 1 MSD-first Digit-Serial 타입1 곱셈기

Input : $A(x) = \sum_{i=0}^{m-1} a_i x^i, B(x) = \sum_{i=0}^{m-1} b_i x^i, \quad a_i, b_i \in GF(3)$

Output: $R(\alpha) = A(\alpha) \cdot B(\alpha) = \sum_{i=0}^{m-1} r_i x^i, \quad r_i \in GF(3)$

1. $R(\alpha) = 0$

2. For $i = \lceil m/D \rceil - 1$ to 0 do

2.1 $U(\alpha) = \sum_{j=0}^{D-1} a_{Di+j} \cdot B(\alpha) \cdot \alpha^j \bmod F(x)$

2.2 $R(\alpha) = R(\alpha) \cdot \alpha^D + U(\alpha) \bmod F(x)$

3. Return $R(\alpha)$

<37>

<38> 상기 알고리즘 1에 의하면 $A(\alpha), B(\alpha) \in GF(3^m)$ 를 만족하는 $A(\alpha)$ 및 $B(\alpha)$ 를 곱하여 $R(\alpha)$ 를 생성하는 MSD-first 디지털 직렬 곱셈에 있어서, 단계 2.1에서 부분곱마다 감산 연산을 수행하므로 이를 병렬화하는 경우, $D(D-1)$ 개의 $GF(3)$ 기반의 곱셈기와 $D(D-1)/2$ 의 덧셈기가 필요하다.

<39> 여기서, 단계 2.1에서 부분곱의 모듈러 감산을 수행하는 수식인 $a_{Di+j} \cdot B(\alpha) \cdot \alpha^j \bmod F(x)$ 는 도 1의 모듈러 감산 연산부에 의해 연산될 수 있다.

<40> 한편, $U(\alpha)$ 는 D 개의 $m-1$ 차 다항식의 합이므로 $GF(3^m)$ 덧셈기가 $D-1$ 개로 구성되며, 상기 덧셈기에 따라서 $(\log \lceil D \rceil)ADD$ 의 시간 지연이 소요된다.

<41> 또한, $R(\alpha) \cdot \alpha^D \bmod F(x)$ 연산은 $U(\alpha)$ 와 병렬로 수행되므로 $R(\alpha)$ 의 연산에서 $1ADD$ 의 시간 지연이 소요되므로, 크리티컬 패스 딜레이(Critical Path Delay)는 $2 \cdot MUL + (\log \lceil D+1 \rceil + 1) \cdot ADD$ 이다.

<42> 이에 대하여 상기 표 1의 알고리즘 1을 하드웨어로 구성하면 도 2와 같다.

<43> 도 2를 참조하면, 상기 알고리즘 1에 따른 종래의 MSD-first 디지털 직렬 곱셈기는 $A(\alpha)$ 의 계수인 D 개의 a_{Di+j} 성분과 $B(\alpha)$ 의 곱셈을 수행하는데 있어서 부분곱의 연산마다 감산 연산을 수행하기 위한 모듈러 감산 연산부가 필요함을 알 수 있고, 모듈러 감산 연산을 수행한 각각의 부분곱 성분끼리 덧셈 연산하는 덧셈기가 구성되어 있음을 확인할 수 있다.

<44> 그 다음, 종래의 [FPGA accelerated Tate pairing based cryptosystem over binary fields]에서 제안된 MSD-first 디지털 직렬 곱셈기를 살펴보기로 하며, 이를 타입 2(type 2)의 MSD-first 디지털 직렬 곱셈기라고 하기로 한다.

<45> [FPGA accelerated Tate pairing based cryptosystem over binary fields]에서는 상기 타입 1의 MSD-first 디지털 직렬 곱셈기와는 달리 디지털(digit) 단위의 프로세스 $U(\alpha)$ 연산에서 모듈러 감산을 수행하지 않고, 누적값 $R(\alpha)$ 의 연산에서 한번에 처리한다.

<46> 따라서, 디지털의 크기를 D 라고 할 때, 매 단계의 루프 i 마다 부분곱의 합 $m+D-2$ 차 다항식의 연산은 하기의 수학적 6과 같다.

수학식 6

$$U(\alpha) = \sum_{j=0}^{D-1} a_{Di+j} \cdot B(\alpha) \cdot \alpha^j$$

<47>

<48> 그 다음은, 연산된 $U(\alpha)$ 와 누적값인 $R(\alpha)$ 를 더하는 단계이며, 이때, $R(\alpha)$ 는 $m + D - 1$ 차 다항식이다.

<49> 타입 2의 곱셈기는 $U(\alpha)$ 의 연산에서 모듈러 감산부가 없으므로 시간지연이 상기 타입 1보다 효율적이며, 각각의 $a_{Di+j} \cdot B(\alpha) \cdot \alpha^j$ 는 상호 $m-1$ 개 항만 더해지므로, 공간복잡도의 면에서 또한 효율적이다.

<50> 물론 $R(\alpha)$ 가 $m + D - 1$ 차 이므로 이의 모듈러 감산 연산부가 증가하나, $U(\alpha)$ 의 모듈러 감산 연산부 감소에 비하여 미비하다. 따라서, $R(\alpha)$ 의 저장 공간인 D 비트를 제외한 모든 부분에서 상기 표 1의 타입 1보다 더욱 효율적이라고 할 수 있다.

<51> 한편, 상기 타입 2의 MSD-first 디지털 직렬 곱셈기는 하기의 표 2의 알고리즘 2와 같다.

표 2

Algorithm 2 MSD-first Digit-Serial 타입2 곱셈기

Input : $A(x) = \sum_{i=0}^{m-1} a_i x^i$, $a_{-i} = 0$, $B(x) = \sum_{i=0}^{m-1} b_i x^i$, $a_i, b_i \in GF(3)$

Output: $R(\alpha) = A(\alpha) \cdot B(\alpha) = \sum_{i=0}^{m-1} r_i x^i$, $r_i \in GF(3)$

1. $R(\alpha) = 0$

2. For $i = \lceil m/D \rceil - 1$ to -1 do

2.1 $U(\alpha) = \sum_{j=0}^{D-1} a_{Di+j} \cdot B(\alpha) \cdot \alpha^j$

2.2 $R(\alpha) = R(\alpha) \cdot \alpha^D + U(\alpha) \bmod F(x)$

3. Return $R(\alpha)$

<52>

<53> 상기 알고리즘 2의 경우 단계 2.1에서 부분곱의 합을 수행하므로 이를 병렬화하는 경우 $D \cdot m$ 개의 $GF(3)$ 기반의 곱셈기와 $(m-1)(D-1)$ 개의 $GF(3)$ 기반의 덧셈기가 필요하다.

<54> 여기서, $U(\alpha)$ 는 $m+D-2$ 차 다항식이고, $R(\alpha) \cdot \alpha^D$ 가 하위 D 개의 계수가 0인 $m+D-2$ 차 다항식이므로, $R(\alpha)$ 의 연산에서 $m-1$ 개의 $GF(3)$ 기반의 덧셈기가 필요하다.

<55> 시간 복잡도의 경우, $U(\alpha)$ 의 연산에서 D 개의 $m-1$ 차 다항식의 부분곱의 합이므로 $1MUL + (\log \lceil D \rceil)ADD$ 의 시간 지연이 소요된다.

<56> 또한, $R(\alpha) \bmod f(x)$ 는 $U(\alpha)$ 와 병렬로 연산이 수행되므로 $R(\alpha)$ 의 연산에서 $1ADD$ 의 시간지연이 소요된다.

<57> 따라서, 크리티컬 패스 딜레이(Critical Path Delay)는 $2MUL + (\log \lceil D \rceil + 1)ADD$ 이며, 마지막 $R(\alpha)$ 의 모듈러 연산을 위하여 1번 추가 반복된 연산이 수행된다.

<58> 상기 표 2의 알고리즘 2를 하드웨어로 구성하면 도 3과 같다.

<59> 도 3을 참조하면, 상기 알고리즘 2에 따른 종래의 MSD-first 디지털 직렬 곱셈기는 $A(\alpha)$ 의 계수인 D 개의 a_{Di+j} 성분과 $B(\alpha)$ 의 곱셈을 수행하는데 있어서 이전의 루프에서 모듈라 연산의 결과값을 감산 연산하는 모듈라 감산 연산부가 필요함을 알 수 있고, 각각의 부분곱을 수행하는 곱셈기와 곱셈 연산이 수행된 부분곱을 합하는 덧셈기가 구성되어 있음을 확인할 수 있다.

<60> 그러나, $GF(3^m)$ 과 같이 표수가 3 이상인 타원곡선 암호 시스템은 다항식의 계수들의 모듈러 연산이 따로 수행되어야 하고, 연산 과정이 더욱 복잡하기 때문에, 실제로 표수가 3 이상인 타원 곡선 암호 시스템이 표수가 2인 타원 곡선 암호 시스템보다 더욱 높은 안전성을 제공함에도 불구하고 거의 사용되고 있지 않은 문제점이 있다.

발명의 내용

해결 하고자하는 과제

<61> 본 발명이 해결하고자 하는 첫 번째 과제는 종래의 MSD-first 곱셈기에 비하여 표수에 의존하지 않고 기존의 유한체 곱셈기에 비하여 고속 동작하면서 이때 증가하는 공간 복잡도를 최소화할 수 있고, 적은 시간 지연을 가질 수 있는 MSD-first 디지털 직렬 곱셈 장치를 제공하는 것이다.

<62> 본 발명이 해결하고자 하는 두 번째 과제는 상기 MSD-first 디지털 직렬 곱셈 장치를 적용한 MSD-first 디지털 직렬 곱셈 방법을 제공하는 것이다.

<63> 한편, 본 발명이 해결하고자 하는 세 번째 과제는 상기 MSD-first 디지털 직렬 곱셈 방법을 컴퓨터에서 수행할 수 있도록 프로그램으로 기록된 기록매체를 제공하는 것이다.

과제 해결수단

<64> 상기 첫 번째 과제를 해결하기 위하여 본 발명은,

<65> $GF(3)$ 상에서 차수가 m 인 삼항 기약 다항식 $f(x) = x^m + f_1 \cdot x' + f_0$ 을 사용하며, 상기 $f(x)$ 의 해가 α 일 때, 유한체 상의 두 원소 $A(\alpha)$, $B(\alpha)$ 의 곱셈을 수행하여 직렬 곱셈 결과값 $R(\alpha)$ 를 생성하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 장치에 있어서, 루프 인덱스 i 가 $\lceil m/D \rceil + 2$ 일 때, D 개의 디지털의 인덱스를 j 라 하면, $b(\alpha)$ 의 계수 b_{i+1} , b_{i+j} , 상기 삼항 기약 다항식의 계수로부터 생성되는 $-f_1$, 및 상기 삼항 기약 다항식의 계수에 따라 생성되는 $-f_1 f_0$ 를 입력으로 하여 $\overline{b_{m+D+i}}$ 와 $\overline{b_{D+i}}$ 를 생성하고, 상기 $b(\alpha)$ 를 이용하여 $\overline{b(\alpha)}$ 를 생성하는 초기값 생성부; 루프 인덱스 i 가 $\lceil m/D \rceil + 1$ 일 때, 상기 연산된 $\overline{b_{D+j}}$ 를 이용하여 제 1 치환 연산자 λ 를 생성하고, 상기 연산된 $\overline{b_{m+D+i}}$ 를 이용하여 제 2 치환 연산자 δ 를 생성하는 전처리 연산부; 상기 전처리 연산부에서 생성된 제 1 치환 연산자 δ , 상기 $\tilde{A}(\alpha)$ 의 계수 $\tilde{a}_{iD+j}$, $-f_1$, $-f_0$ 및 $B(\alpha)$ 를 입력으로 하여 $\tilde{a}_{D+i+j} \overline{b(\alpha)}$ 의 결과값 $v(\alpha)$ 를 생성하는 곱셈부; 이전 루프의 MSD first 디지털 직렬 곱셈의 결과값 $R(\alpha)$, 상기 제 2 치환 연산자 λ , 및 상기 곱셈부의 결과값을 입력으로 하여 덧셈 결과값을 생성하는 가산부; 및 상기 덧셈 결과값을 라이트 쉬프트하여 상기 가산부에 피드백하

는 쉬프트 레지스터를 포함하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 장치를 제공한다.

<66> 한편, 상기 초기값 생성부는 수학적식 $\overline{B(\alpha)} = B(\alpha) - \left(\sum_{j=0}^{D-1} b_{t+D+j} \cdot f_t \cdot F(\alpha) \alpha^{D+j} \right)$ 에 따라 상기 $b(\alpha)$ 를 이용하여 $\overline{b(\alpha)}$ 를 생성하는 것을 특징으로 한다.

<67> 그리고, 상기 초기값 생성부는 디지털 D 의 인덱스를 j 라 하면, $b(\alpha)$ 의 계수 b_{t+1} , b_{t+j} , 상기 삼항 기약 다항식의 계수로부터 생성되는 $-f_t$, 및 $-f_t f_0$ 에 따라 연산되는 제 1 전처리 연산자 t_j^1 에 따라 수학적식 $\overline{b_{m+j}} \leftarrow b_{t+1} \cdot (-f_t)$, $\overline{b_j} \leftarrow t_j^1 + b_j$, where $t_j^1 \leftarrow b_{t+j} \cdot (-f_t f_0)$ 에 의해 $\overline{b_{m+D+j}}$ 과 $\overline{b_{D+j}}$ 을 생성하는 것을 특징으로 한다.

<68> 아울러, 상기 전처리 연산부는 수학적식 $t_j^2 \leftarrow \overline{b_{m+j}} \cdot \tilde{a}_{D(i-2)+j}$, $t_j^3 \leftarrow \overline{b_j} \cdot \tilde{a}_{D(i-2)+j}$ 에 의해 제 2 전처리 연산자 t_j^2 및 제 3 전처리 연산자 t_j^3 를 생성하는 것을 특징으로 한다.

<69> 한편, 상기 전처리 연산부는 수학적식 $\delta_j \leftarrow t_{j+D}^2 + r_{m-D+j}$, $\lambda_j \leftarrow t_j^3 + v_0$ 에 의해 상기 제 1 치환 연산자 λ 및 제 2 치환 연산자 δ 를 생성하는 것을 특징으로 한다.

<70> 그리고, 상기 가산부는 이전 루프의 MSD first 디지털 직렬 곱셈의 결과값 $R(\alpha)$ 및 상기 $A(\alpha)$ 의 계수 a_{Di+j} 와 상기 $B(\alpha)$ 의 곱에 의해 생성되는 중간 연산값 v_j 를 이용하여 수학적식 $R(\alpha) \leftarrow R(\alpha) + \sum_{j=1}^{D-1} v_j$ 에 따라 덧셈 연산의 결과값을 생성하는 것을 특징으로 한다.

<71> 상기 두 번째 과정을 해결하기 위하여 본 발명은,

<72> $GF(3)$ 상에서 차수가 m 인 삼항 기약 다항식 $f(x) = x^m + f_t \cdot x^t + f_0$ 을 사용하며, 상기 $f(x)$ 의 해가 α 일 때, 유한체 상의 두 원소 $A(\alpha)$, $B(\alpha)$ 의 곱셈을 수행하여 직렬 곱셈 결과값 $R(\alpha)$ 를 생성하는 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 방법에 있어서, 루프 인덱스 i 가 $\lceil m/D \rceil + 2$ 일 때, D 개의 디지털의 인덱스를 j 라 하면, $b(\alpha)$ 의 계수 b_{t+1} , b_{t+j} , 상기 삼항 기약 다항식의 계수로부터 생성되는 $-f_t$, 및 상기 삼항 기약 다항식의 계수에 따라 생성되는 $-f_t f_0$ 를 입력으로 하여 $\overline{b_{m+D+j}}$ 과 $\overline{b_{D+j}}$ 을 생성하고, 상기 $b(\alpha)$ 를 이용하여 $\overline{b(\alpha)}$ 를 생성하는 초기값 생성 단계; 루프 인덱스 i 가 $\lceil m/D \rceil + 1$ 일 때, 상기 연산된 $\overline{b_{D+j}}$ 를 이용하여 제 1 치환 연산자 λ 를 생성하고, 상기 연산된 $\overline{b_{m+D+j}}$ 를 이용하여 제 2 치환 연산자 δ 를 생성하는 전처리 연산 단계; 상기 전처리 연산부에서 생성된 제 1 치환 연산자 δ , 상기 $\tilde{A}(\alpha)$ 의 계수 $\tilde{a}_{iD+j}$, $-f_t$, $-f_0$ 및 $B(\alpha)$ 를 입력으로 하여 $\tilde{a}_{iD+j} \overline{b(\alpha)}$ 의 결과값 $v(\alpha)$ 을 생성하는 곱셈 결과값

생성 단계; 이전 루프의 MSD first 디지트 직렬 곱셈의 결과값 $R(\alpha)$, 상기 제 2 치환 연산자 λ , 및 상기 곱셈부의 결과값을 입력으로 하여 덧셈 결과값을 생성하는 덧셈 결과값 생성 단계; 및 상기 덧셈 결과값을 라이트 쉬프트하여 상기 가산부에 피드백하는 피드백 단계를 포함하는 $GF(3^m)$ 기반의 MSD first 디지트 직렬 곱셈 방법을 제공한다.

<73> 그리고, 상기 초기값 생성 단계는 수학적식 $\overline{B}(\alpha) = B(\alpha) - \left(\sum_{j=0}^{D-1} b_{t+D+j} \cdot f_t \cdot F(\alpha) \alpha^{D+j} \right)$ 에 따라 상기 $b(\alpha)$ 를 이용하여 $\overline{b}(\alpha)$ 를 생성하는 것을 특징으로 한다.

<74> 아울러, 상기 초기값 생성 단계는 디지트 D 의 인덱스를 j 라 하면, $b(\alpha)$ 의 계수 b_{t+1} , b_{t+j} , 상기 삼항 기약 다항식의 계수로부터 생성되는 $-f_t$, 및 $-f_t f_0$ 에 따라 연산되는 제 1 전처리 연산자 t_j^1 에 따라 수학적식 $\overline{b}_{m+j} \leftarrow b_{t+1} \cdot (-f_t)$, $\overline{b}_j \leftarrow t_j^1 + b_j$, where $t_j^1 \leftarrow b_{t+j} \cdot (-f_t f_0)$ 에 의해 $\overline{b}_{m+D+j}$ 과 $\overline{b}_{D+j}$ 을 생성하는 것을 특징으로 한다.

<75> 한편, 상기 전처리 연산 단계는 수학적식 $t_j^2 \leftarrow \overline{b}_{m+j} \cdot \tilde{a}_{D(i-2)+j}$, $t_j^3 \leftarrow \overline{b}_j \cdot \tilde{a}_{D(i-2)+j}$ 에 의해 제 2 전처리 연산자 t_j^2 및 제 3 전처리 연산자 t_j^3 를 생성하는 것을 특징으로 한다.

<76> 아울러, 상기 전처리 연산 단계는 수학적식 $\delta_j \leftarrow t_{j+D}^2 + r_{m-D+j}$, $\lambda_j \leftarrow t_j^3 + v_0$ 에 의해 상기 제 1 치환 연산자 λ 및 제 2 치환 연산자 δ 를 생성하는 것을 특징으로 한다.

<77> 상기 세 번째 과정을 해결하기 위하여 본 발명은,

<78> 상기 $GF(3^m)$ 기반의 MSD first 디지트 직렬 곱셈 방법을 컴퓨터에서 수행할 수 있도록 프로그램으로 기록된 기록 매체를 제공한다.

효 과

<79> 본 발명에 의하면, 삼항 기약 다항식이 적용되는 페어링 기반의 암호 시스템에 있어서, 기존의 MSD-first 곱셈기에 비하여 적은 시간 지연을 가지고, 표수에 의존하지 않고, 기존의 유한체 곱셈기에 비하여 고속 동작하면서 이때 증가하는 공간 복잡도를 최소화할 수 있고, 이에 따라 페어링 기반의 암호 시스템을 더욱 효율적으로 설계할 수 있으며, 특히 삼항 기약 다항식을 사용하는 모든 유한체에 적용할 수 있기 때문에 이를 기반으로 하는 하드웨어 설계에 응용할 수 있는 효과가 있다.

발명의 실시를 위한 구체적인 내용

<80> 본 발명은 $GF(3^m)$ 에 기반한 MSD-first 디지트 직렬 곱셈기를 제안한다. 본 발명에서 제안하는 곱셈기는 상기 표 2의 타입 2의 모듈러 감산 연산부의 병렬화에 기반한다.

<81> 상기 알고리즘 2에서 $R(\alpha)$ 는 하기의 수학적식 7에 의해 표현될 수 있다.

수학식 7

$$\begin{aligned}
 R(\alpha) \cdot \alpha^D \bmod F(x) &= \sum_{i=0}^{m-1} r_i \alpha^{i+D} \bmod F(x) \\
 &= \sum_{i=0}^{D-1} r_{m-1+i} \alpha^{m+i} + \sum_{i=0}^{m-2} r_i \alpha^{i+1} \bmod F(x) \\
 &= \sum_{i=0}^{D-1} r_{m-1+i} (-f_i \alpha^{t+i} - f_0 \alpha^i) + \sum_{i=0}^{m-2} r_i \alpha^{i+1} \bmod F(x)
 \end{aligned}$$

<82>

한편, 상기 알고리즘 2에서 반복문에서 i 가 $\lceil m/D \rceil - 1$ 에서 -1까지 감소하는 것을 고려하여 i 번째의 $R(\alpha)$ 를 하기의 수학식 8과 같이 표기한다.

수학식 8

$$R(\alpha)^{(i)} = \sum_{j=0}^{m-1} r_j^{(i)} \alpha^j = R(\alpha)^{(i+1)} \cdot \alpha^D + \sum_{j=0}^{D-1} a_{Di+j} \cdot B(\alpha) \cdot \alpha^j \bmod F(x)$$

<84>

상기 수학식 8에서 $R(\alpha)^{(\lceil m/D \rceil - 1)} = 0$ 의 식이 성립한다.

<85>

한편, 본 발명을 위하여 하기의 정의 및 정리에 따라 본 발명을 더욱 상세히 상술하기로 한다.

<86>

우선 아래와 같이 첫 번째 정의를 기술하기로 한다.

<87>

정의 1(Definition 1)

<88>

$A(\alpha)$ 가 $GF(3^m)$ 의 원소라 하고, 삼차 기약 다항식 $F(x)$ 에 대하여 $\widetilde{F}_i(x) = F(x) \cdot x^i$ ($D \leq i < 2D$)라 하면, 하기의 수학식 9에 의해 $\overline{A}(\alpha)$ 를 정의할 수 있다.

<89>

수학식 9

$$\begin{aligned}
 \overline{A}(\alpha) &= A(\alpha) - \sum_{i=D}^{2D-1} a_{t+i} \cdot f_t \cdot F(\alpha) \alpha^i \\
 &= A(\alpha) - \sum_{i=D}^{2D-1} a_{t+i} \cdot f_t \cdot \widetilde{F}_i(\alpha)
 \end{aligned}$$

<90>

정리 1(Theorem 1)

<91>

$A(\alpha), B(\alpha) \in GF(3^m)$ 이면, 상기 정의 1에서 두 원소 $A(\alpha), B(\alpha)$ 의 곱은 하기의 수학식 10에 의해 정의될 수 있다.

<92>

수학식 10

$$\begin{aligned}
 A(\alpha)B(\alpha) \bmod F(x) &= (A(\alpha)B(\alpha)\alpha^D \bmod F(x)x^D)/\alpha^D \\
 &= \{\widetilde{A}(\alpha)b(\alpha) \bmod \widetilde{F}_D(x)\}/\alpha^D \\
 &= \{\widetilde{A}(\alpha) \cdot \overline{B}(\alpha) \bmod \widetilde{F}_D(x)\}/\alpha^D
 \end{aligned}$$

<93>

상기 수학식 10에서 $A(\alpha)\alpha^D = \widetilde{A}(\alpha) = \sum_{i=0}^{m+D-1} \widetilde{a}_i \alpha^i$ 의 값이 성립된다. 그리고, $F(x)$ 가 삼차 기약 다

<94>

항식이면, $B(\alpha) \equiv \overline{B}(\alpha) \bmod F(x)$ 가 성립되고, $t+D$ 항부터 $t+2D-1$ 까지의 계수는 0이다. 단 $t < m-2D$ 이다.

<95> 한편, $GF(3)$ 의 0이 아닌 임의의 원소 a_i 는 자신의 역원이므로 $a_i^2 = 1$ 의 등식이 성립한다. 따라서 $\overline{B}(\alpha)$ 의 $t+D$ 항부터 $t+2D-1$ 항까지의 계수는 하기의 수학식 11에 의한다.

수학식 11

<96>
$$b_{t+i}\alpha^{t+i} - b_{t+i} \cdot f_t \cdot f_t \alpha^{t+i} = (b_{t+i} - b_{t+i})\alpha^{t+i}$$

<97> 이와 같이 상기 수학식 11에 의거하여 $\overline{B}(\alpha)$ 의 $t+D$ 항부터 $t+2D-1$ 항까지의 계수값은 0이 된다. ($D \leq i < 2D$)

<98> 따라서,
$$R(\alpha) \equiv R(\alpha) \cdot \alpha^D + \sum_{j=0}^{D-1} (A_{Di+j} \alpha^i \cdot B(\alpha)) \bmod F(x)$$
는 상기 정리 1에 의하여
$$R(\alpha) \equiv R(\alpha) \cdot \alpha^D + \widetilde{a}_i \cdot \overline{B}(\alpha) \bmod + \sum_{j=0}^{D-1} (a_{Di+j} \alpha^i \cdot B(\alpha)) \bmod F(x)$$
이며, 반복 연산을 고려하여 표현하면, 하기의 수학식 12와 같으며 $\widetilde{F}_D(x)$ 가 $m+D$ 차 이므로 $\widetilde{R}(\alpha)^{(i+1)}$ 는 $m+D-1$ 차 이다.

수학식 12

<99>
$$\begin{aligned} R(\alpha)^{(i)} &\equiv R(\alpha)^{(i+1)} \cdot \alpha^D + \widetilde{a}_i \cdot \overline{B}(x) + \sum_{j=0}^{D-1} (a_{Di+j} \alpha^i \cdot B(\alpha)) \bmod \widetilde{F}_D(x) \\ &= \widetilde{R}(\alpha)^{(i+1)} + \widetilde{a}_i \cdot \overline{B}(x) + \sum_{j=0}^{D-1} (a_{Di+j} \alpha^i \cdot B(\alpha)) \bmod \widetilde{F}_D(x) \\ &= \sum_{j=D}^{2D-1} (\widetilde{r}_{m+j}^{(i+1)} + \widetilde{a}_i \cdot \overline{b}_{(m+j)}) \alpha^{(m+j)} \\ &\quad + \sum_{j=0}^{m+D-1} \left\{ (\widetilde{r}_j^{(i+1)} + \widetilde{a}_i \overline{b}_j) + \left(\sum_{k=1}^{D-1} \widetilde{a}_i b_{j-k} \right) \right\} \alpha^j \bmod \widetilde{F}_D(x) \end{aligned}$$

<100> 상기 수학식 12에 의하여 $\widetilde{R}(\alpha)$ 반복문에서의 $R(\alpha)$ 를 정의할 수 있고, 여기서, $\widetilde{b}_j$ 에서 $\widetilde{j}$ 가 0보다 작으면 $\widetilde{b}_j = 0$ 이다.

<101> 한편, 상기 수학식 12에서 $\widetilde{r}_j^{(i+1)} + \widetilde{a}_i \overline{b}_j$ 를 $\delta^{(i,j)}$ 라고 하면, $\delta^{(i,j)} \cdot \alpha^{(m+j)}$ 는 $-\delta^{(i,j)} \cdot \widetilde{f}_{j,t+j} \alpha^{(t+j)} - \delta^{(i,j)} \cdot \widetilde{f}_{j,j} \alpha^j$ 와 같이 표현되고, 이를 상기 수학식 12에 대입하여 정리하면, $R(\alpha)^{(i)}$ 의 계수는 하기의 수학식 13과 같다. 이 때, $\widetilde{f}_{j,t+j}$ 는 $\widetilde{F}_j$ 의 $t+j$ 항의 계수이다.

수학식 13

$$r_j^{(i+1)} + \widetilde{a_i \overline{b_j}} - \delta^{(i,j)} \widetilde{f_{j,j}} + \left(\sum_{k=1}^{D-1} \widetilde{a_i \cdot b_{j-k}} \right), D \leq j < 2D$$

$$r_j^{(i)} \equiv \begin{cases} r_j^{(i+1)} + \widetilde{a_i \overline{b_j}} - \delta^{(i,j)} \widetilde{f_{j-t,j}} + \left(\sum_{k=1}^{D-1} \widetilde{a_i \cdot b_{j-k}} \right), t+D \leq j < t+2D. \\ r_j^{(i+1)} + \widetilde{a_i \overline{b_j}} + \left(\sum_{k=1}^{D-1} \widetilde{a_i \cdot b_{j-k}} \right), otherwise \end{cases}$$

<102>

<103>

정리 2(Theorem 2)

<104> $\widetilde{r_{m+j}^{(i+1)}} + \widetilde{a_i \overline{b_{m+j}}}$ 를 $\delta^{(i,j)}$ 라 하고, $D \leq j < 2D$ 일 때, $\widetilde{a_{i+1} \overline{b_{j-D}}} + \widetilde{a_i \overline{b_j}}$ 를 $\lambda^{(i,j)}$ 라 하고 하면, $R(\alpha)^{(i)}$

의 계수 $r(\alpha)^{(i)}$ 는 하기의 수학식 14에 의해 정리된다.

수학식 14

$$r_j^{(i+1)} + \widetilde{a_i \overline{b_j}} - \delta^{(i,j)} \widetilde{f_{j,j}} + \left(\sum_{k=1}^{D-1} \widetilde{a_i \cdot b_{j-k}} \right), D \leq j < 2D$$

$$r_j^{(i)} \equiv \begin{cases} r_j^{(i+1)} + \widetilde{a_i \overline{b_j}} - \delta^{(i,j)} \widetilde{f_{j-t,j}} + \left(\sum_{k=1}^{D-1} \widetilde{a_i \cdot b_{j-k}} \right), t+D \leq j < t+2D. \\ r_j^{(i+1)} + \widetilde{a_i \overline{b_j}} + \left(\sum_{k=1}^{D-1} \widetilde{a_i \cdot b_{j-k}} \right), otherwise \end{cases}$$

<105>

<106> 상기 수학식 14를 참조하여, $i+1$ 번째의 반복문에서 $\delta^{(i,j)}$ 와 $\lambda^{(i,j)}$ 를 연산할 수 있으면, i 번째의 반복문에서 $R(\alpha)^{(i)}$ 의 모든 계수는 한 번의 $GF(3)$ 곱셈과 $\lceil \log(D+1) \rceil$ 번의 덧셈 연산으로 계산된다.

<107> 여기서, $\overline{B(\alpha)}$ 의 $t+D$ 항부터 $t+2D-1$ 까지의 계수는 0이므로, $D \leq j < 2D$ 에 대하여,

$$r_{t+j}^{(i)} = \widetilde{r_{t+1}^{(i+j)}} + \widetilde{a_i \overline{b_{t+j}}} - \delta^{(i,j)} \cdot \widetilde{f_{j,t+j}} = \widetilde{r_{t+1}^{(i+j)}} - \delta^{(i,j)} \widetilde{f_{j,t+j}}$$

이고, $\widetilde{r_{j-t+j}^{(i+j)}}$ 는 $r_{j-D}^{(i+1)}$ 이므로, $\lambda^{(i,j)}$ 를

$$\widetilde{a_{i+1} \overline{b_{j-D}}} + \widetilde{a_i \overline{b_j}}$$

라고 하면, $r_j^{(i)} = \widetilde{a_{i+1} \overline{b_{j-D}}} + \widetilde{a_i \overline{b_j}} - \delta^{(i,j)} \widetilde{f_{j,j}} + \left(\sum_{k=1}^{D-1} \widetilde{a_i \overline{b_{j-k}}} \right) = \lambda^{(i,j)} - \delta^{(i,j)} \widetilde{f_{j,j}} + \left(\sum_{k=1}^{D-1} \widetilde{a_i \overline{b_{j-k}}} \right)$ 가 성립된다.

<108> 따라서, $\lambda^{(i,j)}$ 는 $i+1$ 번째 반복문에서 연산 가능하다.

<109> 또한, $\widetilde{r_{m+D}^{(i+1)}} = \widetilde{r_m^{(i+1)}} = \widetilde{r_m^{(i+2)}} + \widetilde{a_{i+1} \overline{b_m}}$ 이고, 상기 정의 1에 의하여 $\overline{b_m}$ 은 0이므로 $\delta^{(i,j)} = \widetilde{r_m^{(i+2)}} + \widetilde{a_i \overline{b_{m+j}}}$ 는 $i+1$ 번째의 반복문에서 연산 가능하다.

<110> 따라서, $R(\alpha)^{(i)}$ 의 모든 계수는 한번의 $GF(3)$ 곱셈과 $\lceil \log(D+1) \rceil$ 번의 덧셈 연산으로 연산된다.

<111> 본 발명에 따른 MSD-first 비트 직렬 곱셈기를 정리하면 도 4의 알고리즘 3과 같다.

<112> 도 4에 도시된 알고리즘 3에 대하여 기술하면 하기와 같다.

<113> 본 발명에 따른 알고리즘 3은 한번의 곱셈과 $\log \lceil D-1 \rceil$ 번의 덧셈을 $\lceil m/D \rceil + 3$ 번 반복하며 각각의 곱셈(Multiplication)과 덧셈(Addition) 연산에서 (a), (b), (c)는 각각 병렬로 동작한다.

- <114> 여기서, (a)는 $\overline{B(\alpha)}$ 를 초기화하는 과정으로 실제값은 $\overline{B(\alpha)} = B(\alpha) - \left(\sum_{j=0}^{D-1} b_{t+D+j} \cdot f_t \cdot F(\alpha) \alpha^{D+j} \right)$ 의 연산값이므로, $0 \leq j < D$ 의 범위를 가질 때, $\overline{b_{m+D+j}}, \overline{b_{D+j}}$ 는 $F(x)$ 의 계수에 의하여 갱신되며, $\overline{b_{m+j}}, \overline{b_{t+D+j}}$ 는 0이고, 나머지 $\overline{b_i}$ 는 b_i 와 같다.
- <115> 이는 도 4의 알고리즘 3에서 볼 수 있는 바와 같이, (a)initiation은 1(a), 2(a)로 구성되며, $i = \lceil m/D \rceil + 2$ 인 반복문에서의 레지스터에 $\overline{b_{m+D}}, \dots, \overline{b_{m+2D-1}}$ 과 $\overline{b_D}, \dots, \overline{b_{2D-1}}$ 을 저장한다.
- <116> 따라서, 본 발명에 따른 MSD-first 비트 직렬 곱셈기의 알고리즘 3은 $\overline{b_{m+D+j}}, \overline{b_j}$ 만 연산하면 된다.
- <117> (b)는 상기 표 2의 알고리즘 2의 $r_j^{(i)}$ 를 연산하는 부분이고, (c)는 상기 표 2의 알고리즘 2에서의 $\delta^{(i,j)}, \lambda^{(i,j)}$ 를 연산하는 부분이므로, i 번째의 $\delta^{(i,j)}, \lambda^{(i,j)}$ 는 $i+1$ 번째에서 연산되어야 한다.
- <118> 따라서, 본 발명에 따른 알고리즘 3은 i 가 $\lceil m/D \rceil + 2$ 일 때, $\overline{B(\alpha)}$ 를 초기화하고, i 가 $\lceil m/D \rceil + 1$ 일 때, 다음 사용할 $\delta^{(m,j)}, \lambda^{(m,j)}$ 를 연산하여야 하므로, $\tilde{a}_{m+3D-1}, \tilde{a}_{m+D}$ 를 0으로 할당하여 (a)와 (c)를 준비한다.
- <119> 여기서, (c)Precomputation은 상기 (a)에 저장된 값을 이용하여 1(c), 2(c)와 같이 병렬 연산을 위한 제 1 치환 연산자 δ 와 제 2 치환 연산자 λ 를 연산한다.
- <120> 이는 $i = \lceil m/D \rceil + 1$ 에서 부터 정상 작동하며, i 번째의 결과값은 $i-1$ 번째에 사용되어 진다.
- <121> 한편, 본 발명에 따른 MSD-first $GF(3^m)$ 비트 직렬 곱셈기의 블록도는 도 5와 같다.
- <122> 도 5를 참조하면, 본 발명에 따른 MSD-first $GF(3^m)$ 비트 직렬 곱셈기는 초기값 생성부(510), 전처리 연산부(520), 곱셈부(530), 가산부(540) 및 쉬프트 레지스터(550)으로 구성될 수 있다.
- <123> 한편, 본 발명에 따른 초기값 생성부(510), 전처리 연산부(520), 곱셈부(530), 가산부(540) 및 쉬프트 레지스터(550)의 구체적인 동작은 상기 도 4의 알고리즘과 동일하므로 구체적인 상술은 발명의 구성의 명확화를 위하여 과감히 생략하기로 한다.
- <124> $GF(3)$ 상에서 차수가 m 인 삼항 기약 다항식 $f(x) = x^m + f_t \cdot x^t + f_0$ 을 사용하며, 상기 $f(x)$ 의 해가 α 일 때, 유한체 상의 두 원소 $A(\alpha), B(\alpha)$ 의 곱셈을 수행하여 직렬 곱셈 결과값 $R(\alpha)$ 를 생성함에 있어서, 상기 초기값 생성부(510)는 루프 인덱스 i 가 $\lceil m/D \rceil + 2$ 일 때, D 개의 디지털의 인덱스를 j 라 하면, $b(\alpha)$ 의 계수 b_{t+1}, b_{t+j} , 상기 삼항 기약 다항식의 계수로부터 생성되는 $-f_t$, 및 상기 삼항 기약 다항식의 계수에 따라 생성되는 $-f_t f_0$ 를 입력으로 하여 $\overline{b_{m+D+j}}$ 과 $\overline{b_{D+j}}$ 을 생성하고, 상기 $b(\alpha)$ 를 이용하여 $\overline{b(\alpha)}$ 를 생성한다.
- <125> 여기서, 초기값 생성부(510)는 전술한 바와 같이, 하기의 수학식 15에 따라 상기 $b(\alpha)$ 를 이용하여 $\overline{b(\alpha)}$ 를

생성할 수 있다.

수학식 15

$$\overline{B}(\alpha) = B(\alpha) - \left(\sum_{j=0}^{D-1} b_{t+D+j} \cdot f_t \cdot F(\alpha) \alpha^{D+j} \right)$$

한편, 디지털 D 의 인덱스를 j 라 하면, $b(\alpha)$ 의 계수 b_{t+1} , b_{t+j} , 상기 삼항 기약 다항식의 계수로부터 생성되는 $-f_t$, 및 $-f_t f_0$ 에 따라 연산되는 제 1 전처리 연산자 t_j^1 에 따라 하기의 수학식 16에 의해 $\overline{b}_{m+D+j}$ 과 $\overline{b}_{D+j}$ 을 생성할 수 있다.

수학식 16

$$\begin{aligned} \overline{b}_{m+j} &\leftarrow b_{t+1} \cdot (-f_t), \\ \overline{b}_j &\leftarrow t_j^1 + b_j, \text{ where } t_j^1 \leftarrow b_{t+j} \cdot (-f_t f_0) \end{aligned}$$

그리고, 전처리 연산부(520)는 루프 인덱스 i 가 $\lceil m/D \rceil + 1$ 일 때, 상기 연산된 $\overline{b}_{D+j}$ 를 이용하여 제 1 치환 연산자 λ 를 생성하고, 상기 연산된 $\overline{b}_{m+D+j}$ 를 이용하여 제 2 치환 연산자 δ 를 생성한다.

여기서, 상기 전처리 연산부(520)는 하기의 수학식 17에 의해 제 2 전처리 연산자 t_j^2 및 제 3 전처리 연산자 t_j^3 를 생성한다.

수학식 17

$$t_j^2 \leftarrow \overline{b}_{m+j} \cdot \tilde{a}_{D(i-2)+j}, \quad t_j^3 \leftarrow \overline{b}_j \cdot \tilde{a}_{D(i-2)+j}$$

그리고, 상기 전처리 연산부(520)는 하기의 수학식 18에 의해 상기 제 1 치환 연산자 λ 및 제 2 치환 연산자 δ 를 생성한다.

수학식 18

$$\delta_j \leftarrow t_{j+D}^2 + r_{m-D+j}, \quad \lambda_j \leftarrow t_j^3 + v_0$$

그 다음, 곱셈부(530)는 상기 전처리 연산부(510)에서 생성된 제 1 치환 연산자 δ , 상기 $\tilde{A}(\alpha)$ 의 계수 $\tilde{a}_{iD+j}$, $-f_t$, $-f_0$ 및 $B(\alpha)$ 를 입력으로 하여 $\tilde{a}_{D+j} \overline{b}(\alpha)$ 의 결과값 $v(\alpha)$ 을 생성한다.

가산부(540)는 이전 루프의 MSD first 디지털 직렬 곱셈의 결과값 $R(\alpha)$, 상기 제 2 치환 연산자 λ , 및 상기 곱셈부의 결과값을 입력으로 하여 덧셈 결과값을 생성한다.

여기서, 가산부(540)는 이전 루프의 MSD first 디지털 직렬 곱셈의 결과값 $R(\alpha)$ 및 상기 $A(\alpha)$ 의 계수 a_{D+i+j} 와 상기 $B(\alpha)$ 의 곱에 의해 생성되는 중간 연산값 v_j 를 이용하여 하기의 수학식 19에 따라 덧셈 연

산의 결과값을 생성할 수 있다.

수학식 19

$$R(\alpha) \leftarrow R(\alpha) + \sum_{j=1}^{D-1} v_j$$

그러면, 마지막으로 쉬프트 레지스터(550)에서 상기 덧셈 결과값을 라이트 쉬프트하여 상기 가산부에 피드백함으로써 본 발명에 따른 MSD-first $GF(3^m)$ 비트 직렬 곱셈의 연산을 반복적으로 수행하게 된다.

여기서, 쉬프트 레지스터(550)는 $2m+2D$ 비트의 정보를 저장하며, $\delta, \lambda, \overline{b}_{D+j}, \overline{b}_{m+D+j}$ 는 각각 $2D$ 비트를 저장한다.

상기 도 4와 도 5를 참조하면, 덧셈과 곱셈의 각각의 (a)와 (c)의 구성에서 $GF(3)$ 곱셈기 $4D$ 개와 $GF(3)$ 덧셈기 $3D$ 개가 소요되며, 주 연산부와 병렬로 동작한다.

또한, $A(\alpha)\alpha^D$ 를 고려하여 동작하므로 $2m+2D$ 비트의 중간값 $R(\alpha)$ 는 마지막 사이클에서 $2m$ 비트의 결과값이 된다.

도 6은 본 발명에 따른 MSD-first $GF(3^m)$ 비트 직렬 곱셈 방법의 흐름도를 도시한 것이다.

전술한 기존의 [low energy digit-serial/parallel finite field multiplier]의 발명은 모듈러 감산 연산의 결과인 D 개의 중간 결과값과 누적값 $R(\alpha)$ 의 합이 연산되어야 하므로, 트리 구조의 덧셈 연산에서 $\log([D+1])$ 의 연산이 소요되므로, $\log([D+1])$ 의 덧셈 연산에 대한 시간 지연이 필요하게 된다.

또한 기존의 [FPGA accelerated Tate pairing based cryptosystem over binary fields]의 경우 D 개의 중간 결과값과 누적값 $R(\alpha)$ 의 모듈러 감산 연산의 합이 병렬로 연산되어야 하므로, $\log([D+1])$ 의 덧셈 연산에 대한 시간 지연이 필요하다.

그러나, 본 발명의 경우, 제 1 치환 연산자 δ 와 제 2 치환 연산자 λ 의 사전 연산을 이용하여 누적값 $R(\alpha)$ 의 모듈러 감산 연산과 $\widetilde{a_{id}} \cdot B(\alpha) + R(\alpha)$ 의 덧셈 연산을 1번의 덧셈 연산으로 수행하여 $\log([D+1])$ 의 덧셈 연산에 대한 시간 지연만으로도 연산이 가능하다.

도 6에서 본 발명에 따른 MSD-first $GF(3^m)$ 비트 직렬 곱셈 방법은 상기 도 4 및 도 5의 구성과 동일하므로 구체적인 상술은 발명의 구성의 명확화를 위하여 생략하기로 한다.

$GF(3)$ 상에서 차수가 m 인 삼항 기약 다항식 $f(x) = x^m + f_t \cdot x^t + f_0$ 을 사용하며, 상기 $f(x)$ 의 해가 α 일 때, 유한체 상의 두 원소 $A(\alpha)$, $B(\alpha)$ 의 곱셈을 수행하여 직렬 곱셈 결과값 $R(\alpha)$ 를 생성하는 $GF(3^m)$ 기반의 MSD first 디지털 곱셈을 수행하기 위하여, 우선 루프 인덱스 i 가 $\lceil m/D \rceil + 2$ 일 때, D 개의 디지털의 인덱스를 j 라 하면, $b(\alpha)$ 의 계수 b_{t+1} , b_{t+j} , 상기 삼항 기약 다항식의 계수로부터 생성되는 $-f_t$, 및 상기 삼항 기약 다항식의 계수에 따라 생성되는 $-f_t f_0$ 를 입력으로 하여 $\overline{b}_{m+D+j}$ 와 $\overline{b}_{D+j_0}$ 을

생성하고, 상기 $b(\alpha)$ 를 이용하여 $\overline{b(\alpha)}$ 를 생성한다(S610).

<148> 그 다음, 루프 인덱스 i 가 $\lceil m/D \rceil + 1$ 일 때, 상기 연산된 $\overline{b_{D+j}}$ 를 이용하여 제 1 치환 연산자 λ 를 생성하고, 상기 연산된 $\overline{b_{m+D+j}}$ 를 이용하여 제 2 치환 연산자 δ 를 생성한다(S620).

<149> 그 다음, 상기 전처리 연산부에서 생성된 제 1 치환 연산자 δ , 상기 $\tilde{A}(\alpha)$ 의 계수 $\tilde{a}_{iD+j}$, $-f_i$, $-f_0$ 및 $B(\alpha)$ 를 입력으로 하여 $\tilde{a}_{D+j}\overline{b(\alpha)}$ 의 곱셈 결과값 $v(\alpha)$ 을 생성한다(S630).

<150> 그 다음, 이전 루프의 MSD first 디지털 직렬 곱셈의 결과값 $R(\alpha)$, 상기 제 2 치환 연산자 λ , 및 상기 곱셈부의 결과값을 입력으로 하여 덧셈 결과값을 생성하는 덧셈 결과값을 생성한다(S640).

<151> 그러면, 마지막으로 상기 덧셈 결과값을 라이트 쉬프트하여 상기 가산부에 피드백한다(S650)

<152> 본 발명에 따른 $GF(3^m)$ 기반의 MSD first 디지털 직렬 곱셈 방법은 소프트웨어를 통해 실행될 수 있다. 소프트웨어로 실행될 때, 본 발명의 구성 수단들은 필요한 작업을 실행하는 코드 세그먼트들이다. 프로그램 또는 코드 세그먼트들은 프로세서 판독 가능 매체에 저장되거나 전송 매체 또는 통신망에서 반송파와 결합된 컴퓨터 데이터 신호에 의하여 전송될 수 있다.

<153> 컴퓨터가 읽을 수 있는 기록매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록 장치를 포함한다. 컴퓨터가 읽을 수 있는 기록 장치의 예로는 ROM, RAM, CD-ROM, DVD±ROM, DVD-RAM, 자기 테이프, 플로피 디스크, 하드 디스크(hard disk), 광데이터 저장장치 등이 있다. 또한, 컴퓨터가 읽을 수 있는 기록매체는 네트워크로 연결된 컴퓨터 장치에 분산되어 분산방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다.

<154> 한편, 이상에서 설명한 본 발명은, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에 있어 본 발명의 기술적 사상을 벗어나지 않는 범위 내에서 여러 가지 치환, 변형 및 변경이 가능하므로 전술한 실시예 및 첨부된 도면에 의해 한정되는 것이 아니라고 할 것이다.

도면의 간단한 설명

<155> 도 1은 종래의 모듈러 감산 연산부의 개략도를 도시한 것이다.

<156> 도 2는 종래의 MSD- first 디지털 직렬 곱셈기의 일 예를 도시한 것이다.

<157> 도 3은 종래의 MSD- first 디지털 직렬 곱셈기의 다른 예를 도시한 것이다.

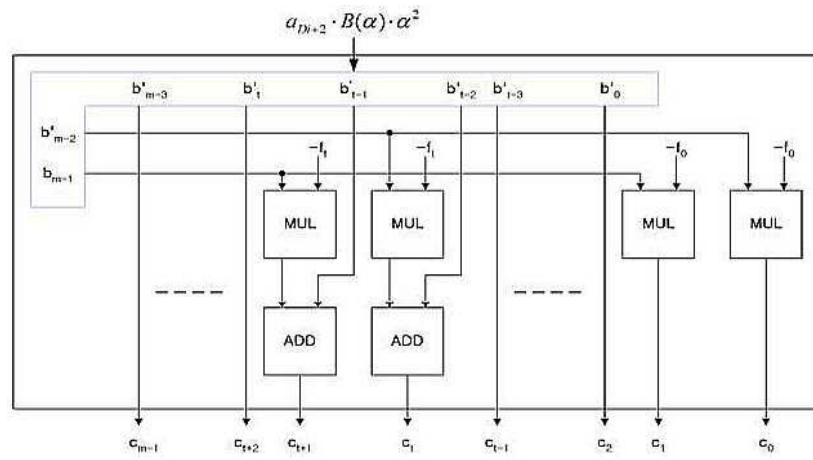
<158> 도 4는 본 발명에 따른 MSD-first 비트 직렬 곱셈의 알고리즘을 도시한 것이다.

<159> 도 5는 본 발명에 따른 MSD-first $GF(3^m)$ 비트 직렬 곱셈기의 블록도를 도시한 것이다.

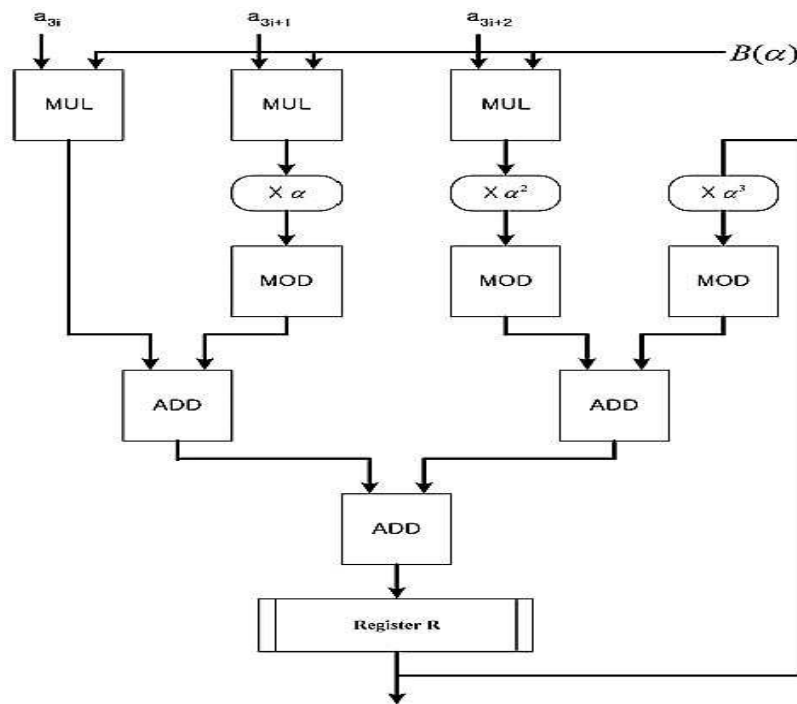
<160> 도 6은 본 발명에 따른 MSD-first $GF(3^m)$ 비트 직렬 곱셈 방법의 흐름도를 도시한 것이다.

도면

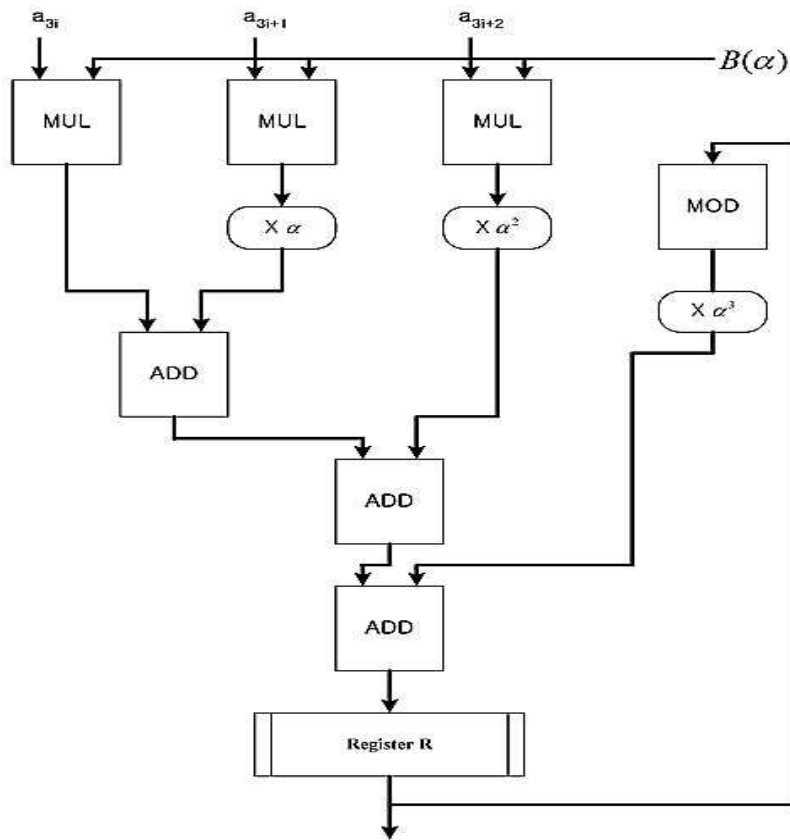
도면1



도면2



도면3



도면4

Algorithm 3 본발명에 따른 MSD-first Digit-Serial 곱셈

Input : $\tilde{A}(\alpha) = A(\alpha)\alpha^D = \sum_{i=0}^{m+3D-1} \tilde{a}_i \alpha^i = (\underbrace{0, \dots, 0}_{2D}, \dots, \underbrace{0, \dots, 0}_{\lceil m/D \rceil}, \dots, \underbrace{a_0, \dots, a_0}_D, \underbrace{0, \dots, 0}_D)$,

$$B(\alpha) = \sum_{i=0}^{m-1} b_i \alpha^i = (b_{m-1}, \dots, b_0), \quad a_i, b_i \in GF(3),$$

$$-f_t, -f_0, -f_t f_0 \in GF(3)$$

Output: $R(\alpha) = A(\alpha)B(\alpha) \bmod F(x) = \sum_{i=0}^{m-1} r_i \alpha^i, \quad r_i \in GF(3)$

For i=D to 2D-1 do

$$\delta_i \leftarrow 0, \lambda_i \leftarrow 0, \overline{b_{m+i}} \leftarrow 0, \overline{b_i} \leftarrow 0$$

For i = $\lceil m/D \rceil + 2$ to 0 do

1. Multiplication

(a) ($\overline{b}(\alpha)$ Initialization) For j=D to 2D-1 do

$$\overline{b_{m+j}} \leftarrow b_{t+1} \cdot (-f_t), \quad t_j^1 \leftarrow b_{t+j} \cdot (-f_t f_0)$$

(b) For j=1 to D-1 do

$$v_j \leftarrow a_{D+j} \cdot B(\alpha)$$

end for

$$v_0 \leftarrow \sum_{\substack{j=0 \\ 0 \leq j < D \\ 2D \leq j < t+D \\ t+2D \leq j < m}}^{m-1} a_{D+j} b_j,$$

For j=0 to D-1 do

$$v_{0,t+D+j} \leftarrow \delta_j \cdot (-f_t), \quad v_{0,D+j} \leftarrow \delta_j \cdot (-f_0)$$

end for

(c) (Precomputation) For j=D to 2D-1 do

$$t_j^2 \leftarrow \overline{b_{m+j}} \cdot \tilde{a}_{D(t-2)+j}, \quad t_j^3 \leftarrow \overline{b_j} \cdot \tilde{a}_{D(t-2)+j}$$

2. Addition

(a) ($\overline{b}(\alpha)$ Initialization) For j=D to 2D-1 do

$$\overline{b_j} \leftarrow t_j^1 + b_j$$

(b) $R(\alpha) \leftarrow \sum_{j=2D}^{m-1} (v_{0,j} + r_{j-D}) \alpha^j,$

For j=0 to D-1 do

$$r_{m+j} \leftarrow r_{m-D+j}, \quad r_{D+j} \leftarrow v_{0,D+j} + \lambda_j, \quad r_j \leftarrow v_j$$

end for

$$R(\alpha) \leftarrow R(\alpha) + \sum_{j=1}^{D-1} v_j,$$

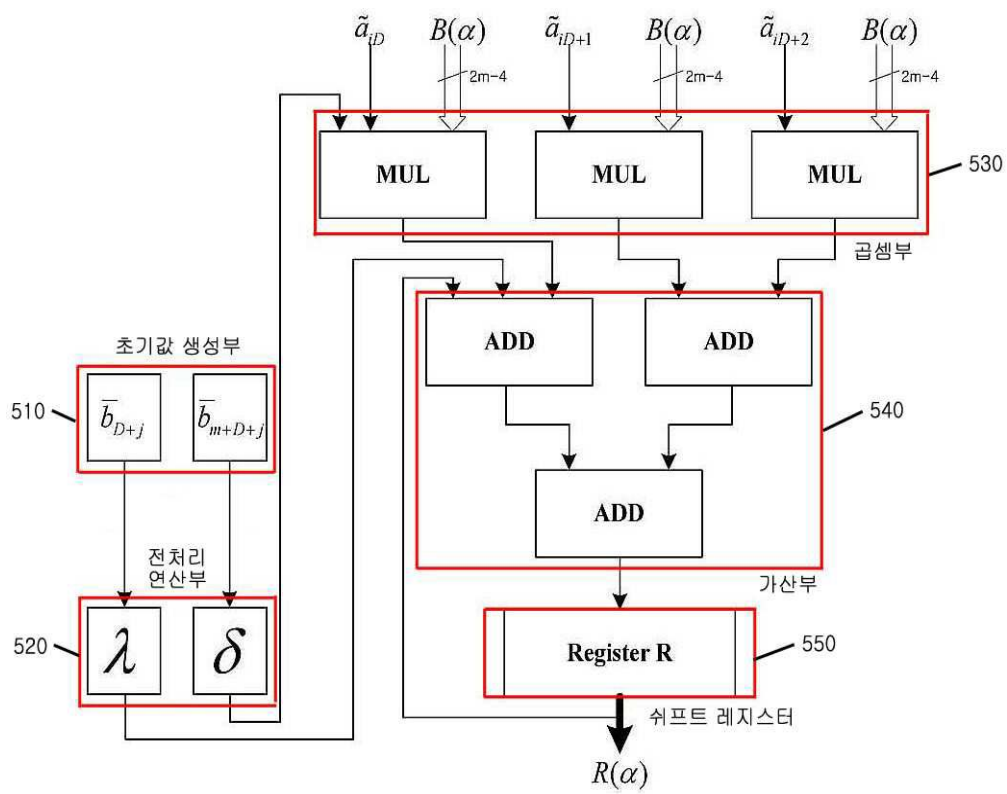
(c) (Precomputation) For j=0 to D-1 do

$$\delta_j \leftarrow t_{j+D}^2 + r_{m-D+j}, \quad \lambda_j \leftarrow t_j^3 + v_0$$

end for

Return $(R(\alpha)/\alpha^D)$

도면5



도면6

