# CA를 이용한 ElGamal 서명기법

이 준 석* 장 화 식** 이 경 현***
* 부경대학교 전자계산학과
** 대덕대학 인터넷정보기술계열
***부경대학교 전자컴퓨터정보통신공학부

# An ElGamal Signature Scheme using Cellular Automata

Kyung Sung*, Sang-Yong Choi*, Woo-Young Soh*

**ABSTRACT**

In this paper, we propose a multiplication scheme based on cellular automata and propose high speed multiplication scheme and exponentiation scheme using a optimal normal basis. And then ElGamal signature scheme is implemented by proposed schemes. A proposed multiplication and exponentiation scheme based on cellular automata can be used in restricted computing environments such that basis is frequently changed and cryptosystem and multimedia applications that are required high speed operations.

## Ⅰ. 서 론

공개키 암호 알고리즘을 소프트웨어로 구현할 경우 현저하게 그 속도가 느리며, 또한 이를 보완하기 위해 하드웨어로 구현할 경우 구성 복잡도가 높아 그 비용면에서 비효율적이다. 또한 대부분의 암호 및 멀티미디어 응용분야에서 유한체내의 연산을 이용하고 있다. 가장 잘 알려져 있는 최적정규기저(optimal normal basis) 표현을 이용한 Massey-Omura 곱셈기의 구현은 XOR 게이트의 수가 $2(m-1)$를 요구한다[1][2]. 하지만 본 논문에서 제안하는 셀룰라 오토마타(cellular automata)를 이용한 최적정규기저 표현에서의 곱셈기는 셀룰라 오토마타가 가지는 기본적인 특성인 모듈러 성질(modularity), 단순성(simplicity), 규칙적 상호연결(regular interconnection), 등에 의해 고속연산을 구현할 수 있으며, 하드웨어 설계 시 AND 게이트의 요구없이 단지 $(m-1)$개의 XOR 게이트만을 요구하기 때문에 하드웨어로 구현함에 있어서 매우 용이한 구조라 할 수 있다[3][4][5][6]. 또한 정규기저 표현에서의 임의의 원소의 자승은 단지 1비트 순환 쉬프트를 통해서 간단하게 구현할 수 있기 때문에 비용과 활용도 면에서 매우 뛰어나다고 할 수 있다.

본 논문의 구성은 2장에서 최적정규기저 표현에서의 곱셈 연산을 위한 알고리즘과 제안 알고리즘을 구현하기 위한 기본 구조로 사용할 셀룰라 오토마타의 기본 개념에 대하여 소개한다. 3장에서 변형된 프로그램 가능한 셀룰라 오토마타(improved programmable cellular automata : IPCA)를 제안하고 이를 이용한 새로운 곱셈기를 설계하고, 제안된 곱셈기를 이용하여 멱승 알고리즘을 보인다. ElGamal 서명 기법에 제안된 알고리즘을 적용하여 그 결과를 4장에서 보이고, 5장에서 결론을 맺는다.

## Ⅱ. 최적정규기저 표현과 셀룰라 오토마타

### 1. 최적정규기저(optimal normal basis)

임의의 $m$차 기약다항식 $p(x)$의 근을 $\alpha$ 라 하면 확장 필드 $GF(2^m)$의 임의의 원소 $a$ 는 다음과 같이 표현할 수 있다.

$$a = a_0 + a_1\alpha + a_2\alpha^2 + \cdots + a_{m-1}\alpha^{m-1} \quad (1)$$

여기서, $a_i \in GF(2)$이다. 이때 $(\alpha^0, \alpha^1, \alpha^2, \cdots, \alpha^{m-1})$를 표준기저 (standard basis)라고 한다. 또한 적절한 원소 $\beta$를 선택하면 $(\beta^{2^0}, \beta^{2^1}, \beta^{2^2}, \cdots, \beta^{2^{m-1}})$로써 확장 필드의 임의의 원소 $b$ 를 다음과 같이 표현할 수 있다.

$$b = b_0\beta + b_1\beta^2 + b_2\beta^{2^2} + \cdots + b_{m-1}\beta^{2^{m-1}} \quad (2)$$

이때, $(\beta^{2^0}, \beta^{2^1}, \beta^{2^2}, \cdots, \beta^{2^{m-1}})$를 정규기저 (normal basis)라고 한다. 또한 정규기저 표현의 두 원소의 곱에서 곱셈 항이 최소가 되도록 하는 $\beta$ 를 최적정규기저(optimal normal basis)라 한다[7][8]. 최적정규기저 표현은 하드웨어 복잡도를 최소화 할 수 있는 매우 좋은 특성을 나타낸다. 특히 정규기저 표현의 임의의 원소의 자승은 단순히 1비트 순환 쉬프트로써 수행 가능하다. 즉, 정규기저로 표현된 임의의 원소 $c$ 를 다음과 같이 표현한다면,

$$\begin{aligned} c &= c_0\beta + c_1\beta^2 + c_2\beta^{2^2} + \cdots + c_{m-1}\beta^{2^{m-1}} \\ &= (c_0, c_1, c_2, \cdots, c_{m-1}) \end{aligned} \quad (3)$$

$c^2$은 아래와 같이 표현할 수 있다.

$$\begin{aligned} c^2 &= c_{m-1}\beta + c_0\beta^2 + c_1\beta^{2^2} + \cdots + c_{m-2}\beta^{2^{m-1}} \\ &= (c_{m-1}, c_0, c_1, \cdots, c_{m-2}) \end{aligned} \quad (4)$$

따라서, 정규기저를 이용하여 필드 원소를 표현한다면 곱셈 연산의 복잡도를 줄일 수 있다.

$x, y \in GF(2^m)$를 다음과 같이 정규기저로 표현된 임의의 두 원소라 하자.

$$x = x_0\beta + x_1\beta^2 + x_2\beta^{2^2} + \cdots + x_{m-1}\beta^{2^{m-1}} = (x_0, x_1, x_2, \cdots, x_{m-1}) \quad (5)$$

$$y = y_0\beta + y_1\beta^2 + y_2\beta^{2^2} + \cdots + y_{m-1}\beta^{2^{m-1}} = (y_0, y_1, y_2, \cdots, y_{m-1}) \quad (6)$$

이 두 원소의 곱을 $z = x \times y$라 하면 $z$ 는 다음과 같이 계산할 수 있다.

$$z = x \times y = \sum_{x_i=0}^{m-1} \sum_{y_j=0}^{m-1} (x_i\beta^{2^i} \cdot y_j\beta^{2^j}) \quad (7)$$

이것은 또한 다음과 같이 나타낼 수 있다.

$$z = x \times y = \sum_{k=0}^{m-1} x \cdot y_k\beta^{2^k} \quad (8)$$

즉, 여기서 $y_k \in GF(2)$이므로 $z = x \times y$를 수행하기 위해서는 $x \cdot \beta^{2^k}$ $(1 \le k \le m-1)$를 계산할 필요가 있으며, 정규기저의 연산 특성을 이용하여 단지 $x \cdot \beta$만의 구현으로 각 $k$ 에 대한 값들을 계산해 낼 수 있다.

## 2. 셀룰라 오토마타(cellular automata)

셀룰라 오토마타는 Von Neumann에 의하여 처음 소개되었고, Wolfram에 의해서 수학적 기초를 마련하였다[9]. 또한 Wolfram은 암호학에 셀룰라 오토마타를 처음으로 도입하였다[10]. 이후 셀룰라 오토마타에 대한 많은 분석과 연구가 이루어졌으며[11][12] 부울 방정식의 해법, BIST 구조, 의사랜덤 수열생성기, 암호알고리즘, 등과 같은 많은 응용분야에 셀룰라 오토마타가 활용되었다[13][14][15][16][17].

특히 Chaudhuri 등은 다차원 셀룰러 오토마타를 제안하였으며, 그룹 셀룰라 오토마타(Group CA)를 구성할 수 있는 선형 법칙(Linear Rules)을 이용한 하이브리드 셀룰라 오토마타(Hybrid CA)를 구성함으로써 같은 길이의 사이클을 이용하여 여러 개의 기본변환(Fundamental Transformations)을 정의하고, 이를 연속적으로 적용함으로써 $n$-비트 메시지 블록을 암호화하는 블록 암호 알고리즘을 제안하였다. 또한, Muzio 등은 최대 주기를 갖는 LFSR에 대응하는 셀룰라 오토마타를 특별한 선형 법칙을 이용한 PCA(Programmable CA)로 구성할 수 있음을 보고하였다[18][19]. 최근에는 Phase shifter를 가지는 LFSR과 셀룰라 오토마타에 대한 비교 분석 결과가 보고되어 있다[20][21]. 기존 연구 결과에 따르면 랜덤성 관점에서 셀룰라 오토마타는 LFSR에 비하여 매우 복잡한 천이과정을 가짐으로써 우수한 랜덤성을 가지는 것으로 알려져 있다.

### 1. 셀룰라 오토마타의 정의 및 종류

셀룰라 오토마타(cellular automata)는 특별한 법칙에 의해 동시에 국소적인 상호작용을 가지는 동일한 셀들이 규칙적으로 배열되어져 있는 유한상태 머신(finite state machine)이다. [그림 1]은 1차원 배열을 가지는 1-D(one-dimensional) 셀룰라 오토마타의 예를 보여준다. 여기서, 각 셀들의 차기 상태는 천이함수 또는 법칙에 의존하여 갱신된다.

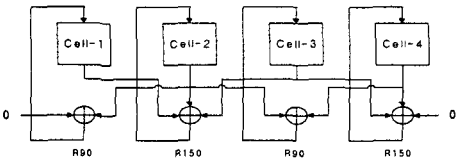



그림 1 4-셀 1-차원 셀룰라 오토마타의 구조

Figure 1. Structure of 1-Dimensional 4-Cell CA

각 셀에 적용된 상태 천이법칙은 아래 식으로 표현할 수 있다.

$$s_i^{t+1} = f(s_{i-1}^t, s_i^t, s_{i+1}^t) \quad (9)$$

여기서, $s_i^t$는 시간 $t$에서 $i$번째 셀의 상태 값을 의미하고, $f$는 상태 천이함수를 나타낸다. 그러므로, 3-이웃 셀룰라 오토마타일 경우 시간 $t$+1에서의 $i$번째 셀의 상태 값은 시간 $t$에서의 $i$-1, $i$, $i$+1번째 셀의 상태 값에 의존하여 결정된다. 상태 천이함수는 일반적으로 아래 표와 같이 법칙으로써 표현한다.

<표 1> 상태천이 법칙의 예
Table 1. Examples of State Transition Rule

| | 111 | 110 | 101 | 100 | 011 | 010 | 001 | 000 |
|---|---|---|---|---|---|---|---|---|
| 법칙 90 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 0 |
| 법칙 150 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 |
| 법칙 60 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 |
| 법칙 102 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 |

<표 1>에서 첫 번째 행은 3개의 이웃으로 $2^3$개의 가능한 상태를 나타낸다. 두 번째 행 이하는 몇 가지 특별한 천이법칙에 대한 상태 값을 나타내고 있다. 따라서, 3-이웃 셀룰라 오토마타에서 가능한 법칙의 개수는 $2^{2^3}$이다. 또한 법칙 90에서 표현되는 90은 2진 상태 값에 대응하는 10진 값과 동일하다. 또한 이들 법칙은 부울 방정식으로 표현할 수 있다. 위 [표 1]에서 보여진 법칙들에 대한 부울 방정식은 아래 식과 같다.

$$\begin{aligned} &\textit{Rule } 90 && s_i^{t+1} = s_{i-1}^t \oplus s_{i+1}^t \\ &\textit{Rule } 150 && s_i^{t+1} = s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t \\ &\textit{Rule } 60 && s_i^{t+1} = s_{i-1}^t \oplus s_i^t \\ &\textit{Rule } 102 && s_i^{t+1} = s_i^t \oplus s_{i+1}^t \end{aligned} \quad (10)$$

부울방정식에 적용된 연산에 따라서 셀룰라 오토마타를 선형 셀룰라 오토마타, 비선형 셀룰라 오토마타, 등으로 구분하고 있다. 선형 셀룰라 오토마타란 적용된 연산이 XOR/XNOR만으로 구성된 것을 의미하고, 그 외의 연산이 적용된 셀룰라 오토마타는 비선형 셀룰라 오토마타이다. 또한 모든 셀에 동일한 법칙이 적용된 셀룰라 오토마타를 Uniform 셀룰라 오토마타라고 하고, 적용된 법칙이 2개 이상일 경우 Hybrid 셀룰라 오토마타라고 부른다.

각 셀의 상태 천이에서 고려해야 할 또 다른 것은 셀룰라 오토마타를 구성하는 양끝의 셀에서 존재하지 않는 이웃에 대한 가정이다. 즉, 1-차원 셀룰라 오토마타에 대하여, 첫 번째 셀의 왼쪽 이웃과 마지막 셀의 오른쪽 이웃이 존재하지 않기 때문에 이에 대한 가정을 정의해야 한다. 이를 경계조건(Boundary Condition)이라고 한다. 경제조건에 따라 CA를 NBCA(Null Boundary CA), PBCA(Periodic Boundary CA), IBCA(Intermediate Boundary CA)로 분류한다. 또한 셀룰라 오토마타의 구성에 따라 1차원, 2차원, 3차원 셀룰라 오타마타로 구분할 수 있다.

## 2. 프로그램 가능한 셀룰라 오토마타

Chaudhuri 등은 [13]에서 3-이웃 셀룰라 오토마타의 응용에 대하여 논하였다. 특히, PCA(programmable cellular automata)를 제안하여 스트림 및 블록 암호 알고리즘에 적용하였다. PCA 구조는 매 갱신 시간마다 새로운 법칙으로써 셀의 상태를 갱신함으로써 셀룰라 오토마타의 상태를 보다 랜덤하게 할 수 있는 구조

이다. 다음 [그림 2]는 PCA의 구조를 나타낸다.

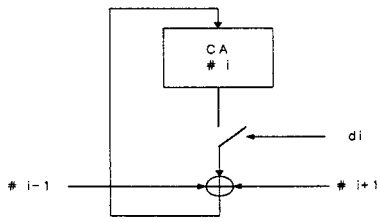



그림 2 PCA의 셀 구조
Figure 2. Cell Structure of PCA

여기서, $di$ 는 $i$ 번째 셀의 갱신법칙을 제어하는 제어신호이다. 즉, $di$ 의 값에 따라서 아래 식과 같이 $i$ 번째 셀의 갱신법칙이 달라진다.

$$s_i^{t+1} = \begin{cases} f(s_{i-1}^t \oplus s_i^t \oplus s_{i+1}^t) : di=1, & Rule\,150 \\ f(s_{i-1}^t \oplus s_{i+1}^t) \quad : di=0, & Rule\,90 \end{cases} \quad (11)$$

## III. 셀룰라 오토마타를 이용한 연산 구조

본 장에서는 PCA를 개선하여 곱셈 알고리즘에 적용할 수 있는 셀룰라 오토마타 기반 곱셈기를 제안한다. [그림 2]에서 보여진 PCA의 구조를 확장하여 제어신호를 3개의 이웃 모두에게 부여함으로써 곱셈기를 구현할 수 있다. 제안된 셀룰라 오토마타 기반 곱셈기는 3개의 제어신호 ($Cx$, $Cy$, $Cp$)를 이용하여, $z = x \times y$를 구현할 수 있다. 다음 [그림 3]은 이를 위한 곱셈기의 셀 구조를 보인다.

여기에서, $Cy$는 다항식 $y$의 계수이며, $Cp$는 다항식 $p$의 계수이다. 또한 $Cl$는 항상 1 이다. 이것은 이전 셀을 쉬프트하기 위함이다. 결과적으로 $n$번의 사이클 이후 셀룰라 오토마타의 상태 값이 $z$의 결과 값이 될 것이다. 제안된 구조는 $Csel$과 $Cy$, $Cp$, $Cl$을 적절하게 조절함으로써 간단하게 연산 구조를 변경할 수 있다. 즉, $Csel$=1 일 경우 MUX의 출력값은 각각 $Ix$와 $Ip$ 값이 된다. 이것은 곱셈기를 구현하기 위한 입력 값들이다. 그리고, $Csel$=0 일 경우, 전통적인 3-이웃 셀룰라 오토마타를 구성한다.

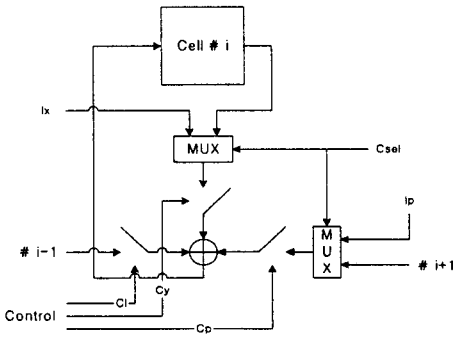



그림 3 개선된 확장 PCA의 셀 구조
Figure 3. Cell Structure of Advanced EPCA

또한, 다항식 $p$의 변화는 하드웨어적인 구조의 변화를 의미하게 되지만 제안된 구조는 $Cp$의 제어신호를 간단히 변경함으로써 구현 가능하다. 따라서, 제안된 셀룰라 오토마타 기반의 곱셈기 구조는 기저가 자주 변경되는 연산 환경에 적절하게 적용할 수 있다.

### 1. 곱셈 알고리즘

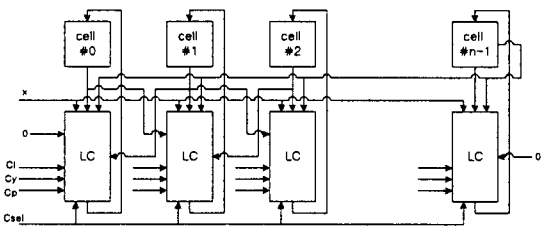



그림 4 곱셈기 구조
Figure 4. Structure of Multiplier

[그림 4]는 제안된 IPCA를 이용하여 곱셈 연산을 구현하기 위한 구조를 보인다. 또한 이를 정규기저(normal basis)에 적용함으로써 연산을 보다 효율적으로 할 수 있다. 정규기정 표현에서의 임의의 원소의 자승은 정규기저 표현에서의 순환 쉬프트를 의미한다. 따라서 제안된 IPCA를 이용하여 쉬프트 레지스트와 곱셈기를 구현하고 이를 이용하여 $m$ 번의 사이클 동안 연산의 결과를 얻을 수 있는 효율적인 곱셈기를 구현할 수 있다. 변형된 곱셈 알고리즘은 다음과 같다.

**[곱셈 알고리즘]**

단계 1] 레지스터 Y와 Z에 각각 $y$와 $z$을 로드한다. Z의 초기값은 0(all zero)이다.

단계 2] 만약 $y_i = 1$ 이면 $x \cdot \beta$ 를 수행한다. 그렇지 않으면 단계 4]로 이동한다.

단계 3] 연산 결과를 Z와 bit-wise XOR 한다.

단계 4] Y와 Z를 1비트 왼쪽 순환 쉬프트한다.

단계 5] $0 \le i \le m-1$ 까지 단계 2~4를 반복한다.

최종적으로 Z에 남아 있는 값이 $z = x \times y$ 이다.

최적정규기저 표현을 이용하여 $x \cdot \beta$를 구현할 경우 셀들 간의 의존도는 항상 2이하이다. 특히 첫 번째 셀의 경우 의존도는 항상 1이 된다[5]. [그림 5]와 pseudo-code는 $p = (11000000001) = 1 + x + x^{10}$, $\beta = \alpha^{93}$ 일 경우 $x \cdot \beta$를 셀룰라 오토마타로 구성한 연산회로이다.

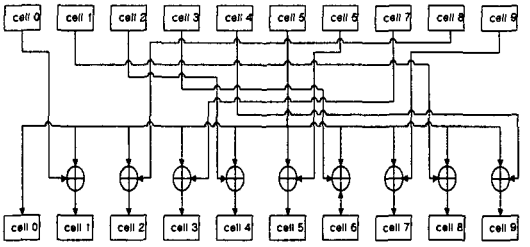



그림 5 셀룰라 오토마타를 이용한 $x \cdot \beta$의 구현 예

Figure 5. Example of $x \cdot \beta$ operation using cellular automata

```
multiplication(int x, int y)
{
  Initialize buffer;
  for(int k=0; k<=m-1; k++){
    if( ISBitSet(y,k) )  buffer ^= Fbox(x);
      x = left_cyclic_shift(x);
      buffer = left_cyclic_shift(buffer);
    }
    return buffer;
}
```

[그림 6]는 제안된 알고리즘을 이용하여 임의의 두 원소 $x$, $y$의 곱을 소프트웨어로 구현한 결과 화면을 보인다.

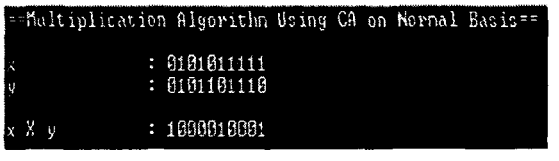



그림 6 정규기저 표현의 곱셈기 구현 예

Figure 6. Example of implementation of multiplication on normal basis

### 2. 멱승 연산과 역원

앞 절에서 설명한 정규기저 표현의 특성과 곱셈 알고리즘을 이용하여 전통적인 square-multiplication 방법을 이용하여 멱승 연산을 수행할 수 있다. 이에 대한 연산 알고리즘은 아래와 같다.

```
exponentiation( int x , int power){
Initialize squ_buffer;
Initialize bitmask;
Initialize mul_buffer;
  if(power==0)  return mul_buffer;
  if(power==1)  return x;
  temp=x;
  for(k=0; k<=maxbit; k++){
    if(k==0){squ_buffer = mul_buffer;
      if(ISBitSet(power,k))  mul_buffer =
multiplication(x, squ_buffer);
    }
    else{
      temp = right_cyclic_shift(temp);
      squ_buffer=temp;
    }
    if(ISBitSet(power,k))  mul_buffer =
multiplication(mul_buffer, squ_buffer);
  }
  return mul_buffer;
}
```

또한 임의의 원소 $x$에 대한 곱셈에 대한 역원은 $x^{2^m-1}=1$임이 명백하기 때문에, 임의의 원소 $x$에 대하여 $2^m-2$번의 멱승을 수행함으로써 $x$에 대한 역원을 구할 수 있다.

아래 [그림 7]은 제안한 멱승 알고리즘을 이용하여 임의의 원소 $x$의 역원을 구하고 $x$와 곱하여 그 결과가 곱셈에 대한 항등원이 나타남을 보이고 있다. 정규기저 표현에서 곱셈에 대한 항등원은 모든 비트가 1인 경우이다.

```
--Exponentiation Algorithm Using CA on Normal Basis --

x              : 0101011111
x^1022         : 0110010110

x X x^-1       : 1111111111
```

그림 7 정규기저 표현의 멱승기 구현 예
Figure 7. Example of implementation of exponentiation on normal basis

## Ⅳ. ElGamal 서명 기법

본 장에서는 Ⅲ 에서 기술한 곱셈 및 멱승 알고리즘을 이용하여 ElGamal 서명 기법을 구현한다. 기존의 ElGamal 서명 기법은 정수 연산이기 때문에 이를 수정없이 적용하기에는 곤란한 점이 있다[9][10]. 따라서 정규기저 표현에서 적용될 수 있는 알고리즘을 다음과 같이 제안한다.

### 1. 키 생성

단계 1] $m$차 기약다항식 $p(x)$와 생성자 $g(x)$를 선택한다.

단계 2] 비밀키 $a$를 선택하고 $y=g(x)^a$를 계산한다.

단계 3] 사용자 A의 공개키는 $p(x)$, $g(x)$, $y$ 이고 비밀키는 $a$이다.

### 2. 서명 생성

단계 1] $\gcd(k, 2^{m-1}) = 1$인 임의의 $k$를 선택하고, $r = g(x)^k$를 계산한다.

단계 2] 서명값 $s$ 를 다음과 같이 구한다.

$$s = k^{-1}(h(m) + a^{-1} \cdot r) \bmod (2^m-1)$$

여기서, $k^{-1}$은 $k$에 대한 곱셈상의 역원, $a^{-1} = 2^m - 1 - a$, h(m)은 메시지 m(x)에 대한 해쉬값이다.

단계 3] $r$ 과 $s$ 를 전송

### 3. 서명 검증

단계 1] 수신된 $r$ 과 $s$ 를 이용하여 다음을 계산

$$\delta = g(x)^{h(m)}, \quad \gamma = y^r \cdot r^s$$

단계 2] 만일 $\delta = \gamma$이면, 정당한 서명으로 인정

$m$ 차 기약다항식 $p(x)$에 의해 생성되는 정규기저 표현에서 0(모든 비트가 0인 원소)을 제외한 원소가 $2^m - 1$개 이므로 서명 $s$ 의 생성 시 법 $2^m - 1$을 취하였으며, 따라서 임의의 값 $k$ 에 대한 역원이 존재하기 위한 조건은 $\gcd(k, 2^{m-1}) = 1$ 이다. 그러므로 서명 $s$ 에 대한 안전성을 높이기 위해 정규기저를 생성할 때 $2^m - 1$이 소수가 되는 $m$차 기약다항식 $p(x)$를 선택하는 것이 바람직할 것이다.

[그림 8]는 제안된 서명 절차를 이용하여 임의의 메시지에 대해 서명 생성 및 서명 검증을 수행한 결과를 보여준다. 여기서 선택한 $h(m)$, $g(x)$, $a$, $k$는 다음과 같다.

$h(m)$ : (00010 11111)
$g(x)$ : (10011 01111)
$a$ : 649
$k$ : 625($k$의 역원은 676)


그림 8 정규기저 표현의 ElGamal 서명기법의 구현 예

Figure 8 Example of ElGamal signature scheme on normal basis

## Ⅳ. 결 론

본 논문에서는 최적정규기저 표현에서의 효율적인 곱셈 및 멱승 연산을 정의하고 이를 셀룰라 오토마타로 구현하였다. 곱셈기의 경우 Massey-Omura 곱셈기보다 비트당 XOR 게이트의 수를 약 1/2로 현저하게 줄였을 뿐 아니라 하드웨어 구현 시 효율적으로 구성할 수 있음을 보였다. 또한 구현된 곱셈 및 멱승기를 이용하여 정수 기반 ElGamal 서명 기법을 정규기저 표현에서 적용할 수 있는 새로운 기법을 제안하였다. 하지만 최적정규기저 원소를 결정하는 효율적인 알고리즘의 개발이 뒤따라야 할 것으로 생각된다.

제안된 셀룰라 오토마타 기반의 곱셈기 및 멱승 연산 구조는 암호 알고리즘 및 멀티미디어 데이터에서 요구되는 고속 연산 알고리즘에 적합하다. 뿐만 아니라 기저가 자주 변경되는 제한된 연산 환경에서 보다 효율적으로 적용될 수 있을 것이다. 또한 본 논문에서 제한하고 있는 셀룰라 오토마타 기반의 새로운 연산 구조는 기존의 연산 구조를 획기적으로 변형하는 새로운 구조가 될 것으로 생각한다.

## 참고문헌


[1] Y.R. Shayan, T. Le-Ngoc, "The least complex parallel Massey-Omura multiplier and its LCA and VLSI designs", Circuit, Devices and Systems, *IEE Proceedings G*, vol.136, iss.6, Dec. 1989.

[2] G. Drolet, "Massey-Omura type adder for elements of finite fields $GF(2^m)$ in logarithmic representation", *IEE Electronics Letters*, vol.35, iss.5, pp.368-369, March 1999.

[3] S. Nandi, B.K. Kar, P.P. Chaudhuri, "Theory and Application of Cellular Automata in Cryptography", *IEEE Transactions on Computers*, vol.43, iss.12, pp.1346-1357, Dec. 1994.

[4] A.K. Das, A. Ganguly, A. Dasgupta, S. Bhawmik, P.P. Chaudhuri, "Efficient characterization of cellular automata", *Computers and Digital Techniques, IEE Proceedings E*, vol.137, iss.1, pp.81-87, Jan. 1990.

[5] P.Pal. Choudhury, R. Barua, "Cellular Automata Based VLSI Architecture for Computing Multiplication and Inverses In GF( $2^m$)", *IEEE Proceedings of the Seventh International Conference on VLSI Design*, pp.279-282, 1994.

[6] Shuhong Gao, *Normal Bases over Finite Fields*, Ph.D Thesis, Combinatorics and Optimization, University of Waterloo, 1993.

[7] B. Sunar, C.K. Koc, "An Efficient Optimal Normal Basis Type II Multiplier", *IEEE Transactions on Computers*, vol.50, iss.1, Jan. 2001.

[8] R. Barua, S. Sengupta, "Architectures for Arithmetic over $GF(2^m)$", *IEEE Proceedings of Tenth International Conference on VLSI Design*, pp.465-468, 1997.

[9] S. Wolfram, *Cellular Automata and Complexity*, Addison Wesely Publishing Company, 1994.

[10] S. Wolfram, "Cryptography with Cellular Automata", *in Advances in Cryptology: Crypto '85 Proceedings, Lecture Notes and Computer Science*, vol.218, pp.429-432 (Springer-Verlag, 1986)

[11] A.K. Das, A. Ganguly, A. Dasgupta, S. Bhawmik, P.P. Chaudhuri, "Efficient characterization of cellular automata", *IEE Proceeding E, Computer and Digital Techniques*, vol.137, no.1, pp.81-87, Jan. 1990.

[12] K. Cattell, M. Serra, "Analysis of One-Dimensional Multiple-Value Linear Cellular Automata", *IEEE Proceedings of the 20th International Symposium on Multiple Valued Logic*, pp.402-409, 1990.

[13] P.P. Chaudhuri, A.R. Chowdhury, S. Nandi, S. Chattopadhyay, *Additive Cellular Automata: Theory and Applications, Volume 1*, IEEE Computer Society Press, 1997.

[14] S. Bhattacharjee, S. Sinha, C. Chattopadhyay, P.P. Chaudhuri, "Cellular automata based scheme for solution of Boolean equations", *IEE Proceedings E, Computer and Digital Techniques*, vol.143, no.3, 1996.

[15] M. Mihaljevic, H. Imai, "A Family of Fast Keystream Generations based on

Programmable Linear Cellular Automata over GF(q) and Time-Variant Table", *IEICE Transactions on Fundamentals*, vol.E82-A, no.1, pp.32-39, 1999.

[16] M. Mihaljevic, Y. Zhang, H. Imai, "A Fast and Secure Stream Cipher based on Cellular Automata over GF(q)", *IEEE Global Telecommunications Conference, GLOBECOM '98*, vol.6, pp.3250-3255, 1998.

[17] B. Srisuchinwong, T.A. York, Ph. Taslides, "A Symmetric Cipher using autonomous and non-autonomous cellular automata", *IEEE Global Telecommunications Conference, GLOBECOM '95*, pp.1172-1177, 1995.

[18] K. Cattell, J. Muzio, "Analysis of One-Dimensional Linear Hybrid Cellular Automata over GF(q)", *IEEE Transactions on Computers*, vol.45, no.7, pp.782-792, 1996.

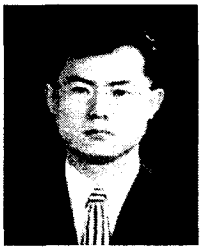
[19] K. Cattell, "Synthesis of One-Dimensional Linear Hybrid Cellular Automata", *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol.15, no.3, pp.325-335, 1996.

[20] J. Rajski, G. Mrugalski, J. Tyszer, "Comparative Study of CA-based PRPGs and LFSRs with Phase Shifters", *IEEE Proceedings of 17th VLSI Test Symposium*, pp.236-245.

[21] P.S. Cardoso, M. Strum, J.R. Amazonas W.J. Chau, "Comparison between Quasi-Uniform Linear Cellular Automata and Linear Feedback Shift Registers as Test Pattern Generators for Built-In Self Test Applications", *IEEE Proceedings of 12th Symposium on Integrated Circuits and Systems Design*, pp.198-201, 1999.

[22] 박승안, *대수학과 암호학*, 경문사, 1999.

[23] 이임영, 송유진, *현대암호*, 생능출판사, 1999.



이준석(Jun-Seok Lee)

1995년 2월 : 동의대학교 전자통신공학과 졸업

1998년 2월 : 동의대학교 전자공학과 석사

2001년 2월 : 부경대학교 전자계산학과 박사수료

관심분야 : 셀룰라 오토마타, 정보보호, 암호이론, 부호이론

장화식(Hwa-Sik Jang)

1993년 2월 : 계명대학교 통계학과 졸업

1995년 2월 : 부경대학교 대학원 전자계산학과 석사

2000년 2월 : 부경대학교 대학원 전자계산학과 박사수료

1996년 3월 ~ 1999년 8월 : 제주관광대학 사무자동화과 전임강사

2000년 3월 ~ 현재 : 대덕대학 인터넷정보기술계열 전임강사

관심분야 : 컴퓨터보안, 정보보호, 암호학, 그룹키관리

이 경 현
(Kyung-Hyune Rhee)

1982년 2월 : 경북대학교 수학교육과 졸업

1985년 2월 : 한국과학기술원 응용수학과 석사

1992년 8월 : 한국과학기술원 수학과 박사

1985년 2월 ~ 1993년 2월 : 한국전자통신연구소 연구원, 선임연구원

1993년 3월 ~ 현재 : 부경대학교 전자컴퓨터정보통신공학부 전임, 조교수, 부교수

관심분야 : 암호학, 암호프로토콜, 네트워크보안, 이동네트워크, 그룹키 관리