유비쿼터스 컴퓨팅 보안을 위한 경량 블록 암호 구현

김성환* · 김동성* · 송영덕** · 박종서*

요 약

본 논문에서는 유비쿼터스 컴퓨팅 보안을 위한 128비트 Reversible Cellular Automata(RCA) 기반 경량 블록 암호를 설계하고 구현한다. 유비쿼터스 컴퓨팅이 요구하는 하드웨어 제약조건을 충족하기 위하여 높은 임의성을 제공하는 Cellular Automata를 기반으로 블록구조를 설계하였다. 구현된 블록 암호기법은 암호화 과정동안 704 클럭 사이클로 동작하고 2,874 게이트수를 보였다. 구현 결과 기존의 AES나 NTRU보다 처리속도가 31% 향상되었고, gate수는 20%만큼 절감되었다. 차분 분석(Differential Cryptanalysis)과 Strict Avalanche Criterion(SAC)을 수행함으로써 구현된 블록 암호 알고리즘의 안정성을 검증하였다.

Implementation of Lightweight Block Cipher for Ubiquitous Computing Security

Sung-Hwan Kim* · Dong Seong Kim*
Young Deog Song** · Jong Sou Park*

ABSTRACT

This paper presents a 128-bit Reversible Cellular Automata (RCA) based lightweight block cipher for Ubiquitous computing security. To satisfy resource-constraints for Ubiquitous computing, it is designed as block architecture based on Cellular Automata with high pseudo-randomness. Our implementation requires 704 clock cycles and consumes 2,874 gates for encryption of a 128-bit data block. In conclusion, the processing time outperformed that of AES and NTRU by 31%, and the number of gate was saved by 20%. We evaluate robustness of our implementation against both Differential Cryptanalysis and Strict Avalanche Criterion.

Key words : Cellular Automata, Lightweight Block Cipher, Ubiquitous Computing, Security

* 한국항공대학교 컴퓨터공학과
** 한서대학교 정보보호공학과

1. 서 론

유비쿼터스 컴퓨팅은 주변 환경 정보(온도, 습도 등)를 감지할 수 있는 매우 작은 유비쿼터스 센서들이 네트워크를 구성하고 있어서, 사용자들이 누구나 쉽게 언제, 어디서, 어떤 객체에서든지 원하는 정보를 얻을 수 있는 장점이 있다. 그러나 이러한 유비쿼터스 컴퓨팅 환경에 편재되어 있는 작은 장치들로 인해, 사용자들은 익명의 다른 사용자에게 위치 추적, 사용자 정보 누출 등의 감시를 받을 수 있기 때문에 보안에 심각한 위협을 받는다. 따라서 안전한 유비쿼터스 컴퓨팅을 위해 다른 보안 기술과 더불어 암호 알고리즘의 사용이 필수적이다.

유비쿼터스 환경을 실현하기 위한 핵심 기술로 RFID(Radio Frequency Identification)와 USN (Ubiquitous Sensor Network)가 대표적이다. 먼저 RFID 태그의 표준 가격은 평균 5센트이며 이로 인해 게이트 수는 5K~10K이하로 설계되어야 한다[8]. 또한 USN의 경우 센서 노드의 제한된 메모리 용량과 배터리 수명을 연장하기 위하여 저전력의 경량화된 암호 알고리즘의 설계 및 구현이 요구된다. 기존의 전통적인 네트워크에 적용된 DES나 AES와 같은 블록 암호 알고리즘은 유비쿼터스 컴퓨팅의 제약사항을 충족하기 어렵다.

따라서 본 논문에서는 이러한 유비쿼터스 컴퓨팅 환경의 제약 조건을 만족하는 동시에 우수한 보안성을 제공하는 암호 알고리즘을 제안한다. CA는 우수한 임의성과 복잡성 때문에 다양한 응용분야에 적용되어 왔다. CA는 Von Neumann [1]에 의해 처음 소개되었으며, Wolfram[2]에 의해 수학적 기초를 마련하였고 암호학에도 처음 적용되었다. 이후 부울 방정식의 해법, BIST 구조, 의사 랜덤 수열 생성기, 암호 알고리즘 등과 같은 많은 응용 분야에 활용되었다[3-7]. 한편 CA는 매우 경량화된 구조이므로 효율적인 하드웨어 설계가 가능하다. 본 논문에서는 Cellular Automata(CA) 기반의 경량 암호 알고리즘을 설계하고 VHDL로 구현한다. CA 기반의 암호 알고리즘은 F. Bao[19]와 F. Seredynski 외[12]에 의해서 제안되었다. 본 논문에서는 Reversible Cellular Automata(RCA) 개념을 이용하여 효율적인 구조로 F. Bao[19]가 제안한 기존의 CA 기반 암호 알고리즘[20-23]보다 경량화된 방식으로 암·복호화가 가능하도록 설계하였다. 한편, M. Seredynski 외[9]가 제시한 1024-bit RCA 기반 블록 암호기법의 전체 구조와 설계 방식을 도입하였지만, 유비쿼터스 컴퓨팅 환경에 적합하게 설계할 뿐만 아니라 새로운 암호 안정성이 우수한 룰들을 사용함으로써 확장 가능성을 제공한다. 본 논문에서 구현한 경량 블록 암호 알고리즘은 전체 128비트 평문의 암호화 과정 동안 하나의 8-bit 데이터 블록으로 16라운드를 가지는 구조이다. 또한 16라운드를 통한 암호화를 수행하는 시간이 704 클럭 사이클만큼 소요되고 이를 위해 2,874 게이트수가 필요하다. 구현된 경량 암호 알고리즘의 우수성을 검증하기 위해 속도와 게이트 용량 면에서 AES 및 NTRU와 비교 분석하며, 암호 안정성을 증명하기 위해 차분 분석[10]과 Strict Avalanche Criterion(SAC)[11]을 수행한다.

본 논문은 다음과 같이 구성된다. 2장은 배경 지식에 대해 살펴보고, 3장은 CA와 RCA, 그리고 기본 블록 암호 블록 다이어그램, 전체 블록 암호 다이어그램에 대해 알아본다. 4장은 차분 분석 및 SAC와 관련된 암호 분석과 구현 결과를 비교, 분석한다. 그리고 5장은 결론에 대해 논의할 것이다.

2. 배경 지식

CA는 1비트 크기의 비트 열로 구성되며, 특정한 rule에 따라 중심 셀과 그 이웃 셀 간의 국소

작용에 의해 자동으로 전개되는 구조로 되어있다. 그리고 이러한 CA에 단일한 rule이 적용될 때, "uniform CA"라고 하며, 여러 다양한 rule이 적용 될 때, "nonuniform CA"라고 한다. 또한 CA에 적용되는 rule은 그 중심 셀과 이웃 셀 사이의 거리 r에 따라 여러 종류로 구분된다. 각각의 rule들은 특정한 임의성과 복잡성을 가지고 있으며, 이러한 rule들 중에서 보다 우수한 임의성과 복잡성을 지닌 rule의 선별 작업이 필요하다. F. Seredynski 외[12]는 중심 셀과 이웃 셀 간의 거리 r이 1일 때의 256개 rule들과 r=2일 때의 2^{32}개 rule들에 대해서, Cellular Programing[13]기법과 uniform CA에 대한 entrophy, x^2, serial correlation과 FIPS 140-2 standard[14]의 monobit, poker, runs, long runs 들을 실험하였고, nonuniform CA에 대해 Marsaglia 실험을 수행하였다. 이를 통해 총 8개의 rule이 선택 되었으며, 키 암호 알고리즘으로서 충분히 만족할 만한 우수한 임의성과 복잡성을 지니고 있다고 제안하였다. 해당되는 8개의 rule은 r=1일 때, rule 86, 90, 101, 105, 150, 153, 165, 그리고 r=2일 때, rule 1436194405이다.

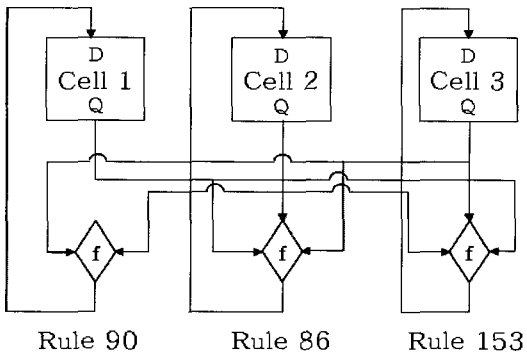
또한, M. Seredynski 외[9]는 이러한 RCA 기반의 1024-bit 블록 암호 알고리즘을 제안하였다. 그리고 Feng Bao[19]는 기존의 여러 유비쿼터스 컴퓨팅 환경에 적합한 CA기반 블록암호 알고리즘[20-23]이 암호 안정성면에서 취약함을 지니고 있다고 소개하였다. 따라서 본 논문에서 제안하는 블록 암호 알고리즘의 암호 안정성에 대한 검증이 필요하고 AES[15]와 NTRU[18]과 같은 기존의 블록 암호 알고리즘과의 게이트 수와 클록수의 비교를 통해 하드웨어 설계 효율성 및 속도 면에서의 우수성을 증명한다.

3. RCA기반 암호알고리즘 설계

본 장에서는 키 암호 알고리즘으로서 CA와 RCA 알고리즘들의 기본적인 특성과 법칙, 구조들을 살펴보고 RCA8와 RCA6, RCA4L, 그리고 RCA4R에 적용되는 기본 블록 암호 설계 구조에 대해 알아본 후에 전체 RCA기반 블록 암호의 구조에 대해서 설명한다.

3.1 Cellular Automata

CA는 시간 t에서의 중심 셀과 이웃하는 셀들의 값에 따라 일종의 rule을 적용하여 시간 t+1로 새로운 값을 갱신하는 이산 시간 구조로 구성된다. 2장에서 설명한 것처럼, CA는 uniform과 nonuniform 형태가 존재하고, 거리 r에 따라서 다양한 rule로 구분할 수 있다. 다음 (그림 1)은 1차원이고 각각 1비트 크기인 3셀 격자 배열로 구성된 거리 r=1일 때의 간단한 nonuniform CA의 하드웨어 구조를 보여준다.



(그림 1) 1차원 Cellular Automata 구조

D-플립플롭 각각은 1비트의 셀을 의미하며, f는 상태 천이 함수를 말한다. CA는 이산 시간 모델로서, 각각의 셀이 1 또는 0의 값을 저장한 이전 상태에서 다음 상태로 갱신될 때, 상태 천이 함수 f에 의해 결정된다. 이를 식으로 표현하면 식 (1)과 같다.

$$Q_i^{t+1} = f(Q_{i-1}^t, Q_i^t, Q_{i+1}^t) \qquad (1)$$

여기서, Q_i^t는 시간 t일 때 i번째 셀의 값 Q를 의미하며, 시간 t에서 3비트 이웃 셀인 Q_{i-1}^t, Q_i^t, Q_{i+1}^t에 의해 시간 t+1의 i번째 셀의 값이 결정된다. 또한, 상태 천이 함수 f는 상태 천이 법칙인 rule에 의해 결정된다. <표 1>은 여러 상태 천이 법칙의 예들을 보여준다. 가로 열의 000에서 111까지의 이진수는 시간 t에서의 가능한 모든 3비트 이웃 셀의 값을 나타내고 가로열의 각 rule에 대한 값들은 이진수의 값에 의해 갱신된 시간 t+1에서의 일정한 결과 값들을 의미한다.

〈표 1〉 상태천이 법칙(rule)의 예

셀 / rule	111	110	101	100	011	010	001	000
90	0	1	0	1	1	0	1	0
165	1	0	1	0	0	1	0	1
153	1	0	0	1	1	0	0	1
101	0	1	1	0	0	1	0	1

이러한 상태 천이 법칙을 부울 대수로 표현하면 식 (2)와 같다.

$$Rule90: \quad Q_i^{t+1} = Q_{i-1}^t \oplus Q_{i+1}^t$$

$$Rule165: \quad Q_i^{t+1} = (Q_{i-1}^t) \cdot (Q_{i+1}^t) + \overline{(Q_{i-1}^t)} \cdot \overline{(Q_{i+!}^t)}$$

$$Rule153: \quad Q_i^{t+1} = (Q_i^t) \cdot (Q_{i+1}^t) + \overline{(Q_i^t)} \cdot \overline{(Q_{i+1}^t)}$$

$$Rule101: \quad Q_i^{t+1} = (Q_i^t) \cdot \overline{(Q_{i+1}^t)} + \overline{(Q_{i-1}^t)} \cdot \overline{(Q_{i+1}^t)} + (Q_{i-1}^t) \cdot \overline{(Q_i^t)} \cdot (Q_{i+1}^t) \quad (2)$$

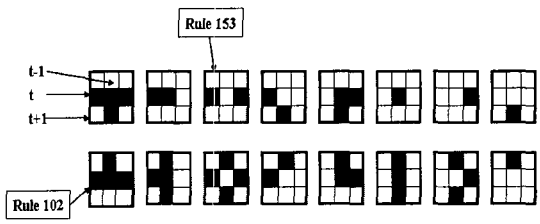
{ • }는 and, {+}는 or, {¯}는 not, 그리고 {⊕}는 xor를 표시한다. rule의 부울 방정식이 xor 혹은 xnor만으로 구성되어 사용된 CA를 선형 CA라 하고, 이와 반대일 경우에는 비선형 CA라고 정의한다. 그리고 단순히 선형 CA만 적용하는 것보다 비선형 CA를 함께 적용하면 출력 블록의 비선형성을 더욱 높여준다[24]. 또한, CA는 유한 상태 머신이므로, 셀 격자의 좌측 끝단과 우측 끝단이 중심 셀일 경우에 이웃한 셀의 적용 여부를 고려해야 한다. 본 논문은 Cyclic Boundary Condition[12]을 적용한다.

3.2 Reversible Cellular Automata

일반적인 CA는 이산 시간 모델로서, 시간 t와 갱신된 시간 t+1일 때만 고려하지만 RCA는 시간이 t-1일 때까지 확장하여 구성되는 키 암호 알고리즘이다. RCA에서 사용되는 rule을 "reversible rule"이라고 정의한다[9]. 하나의 reversible rule안에는 두 개의 rule이 공존하고 두 rule에 대한 상관관계는 식 (3)과 같이 정의 된다[9].

$$R_2 = 2^d - R_1 - 1 \quad (3)$$

R_2와 R_1은 reversible rule을 의미하며, d는 <표 1>의 000에서 111까지의 3비트 이웃 셀이 표현할 수 있는 모든 셀의 종류를 의미하고, 따라서 d=8이 된다. 결국 R_2나 R_1중에서 하나만 알면 다른 나머지 rule을 구할 수 있으며, 그 두 rule이 상호 보완적으로 reversible rule이 되는 것이다.



(그림 2) 1차원 Rule 153R(R=reversible)

(그림 2)는 1차원 rule 153R(R=reversible)의 구조를 보여준다. 셀의 검은색은 1, 흰색은 0을

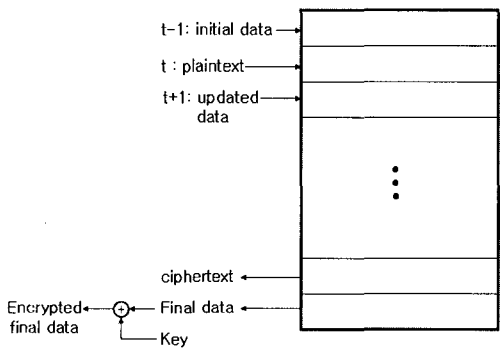
의미하며, rule 153과 rule 102(예를 들어, 식 (3)의 R_2와 R_1)로 구성된다. 이것은 <표 1>에서 설명된 법칙과 유사한 구조이며, 다만 시간 t-1일 때, 셀의 상태 값이 선택조건이 된다. 즉, RCA 알고리즘은 시간 t-1일 때의 셀의 값에 의해 선택되고 다음과 같이 요약할 수 있다.

1. *시간 t-1일 때, 값이 0(또는 1)이면,*
2. *적용되는 상태 천이 법칙의 부울 대수는 rule 153(또는 rule 102)이다.*

시간 t-1 일 때와 t일 때의 셀 상태가 입력 값이 되고, t+1일 때의 셀이 결과 값이 된다. 그리고 이와 같은 알고리즘은 다음 절에서 보여주는 기본 블록 암호 구조에 그대로 적용된다.

3.3 기본 블록 암호 구조

본 절에서는 반복 구조를 갖는 전체 경량 블록 암호 알고리즘의 기본 블록 다이어그램에 대해 살펴본다. 다음 (그림 3)은 기본 블록 암호 알고리즘의 구조를 보여준다.



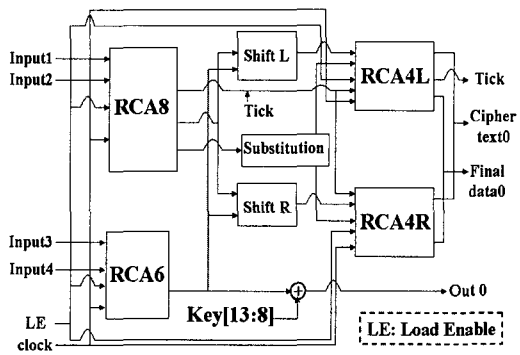
(그림 3) 블록 암호 구조 블록 다이어그램

3.2절에서 설명한 시간 t-1과 t일 때의 두 입력 값이 블록 암호의 입력이 된다. 시간 t-1일 때, initial data가 입력 값이 되고 시간 t에서는 평문이 입력되고, 시간이 지남에 따라 시간 t+1에서 일정한 reversible rule에 의해 갱신된다. 그리고 다시 시간 t와 t+1의 두 값들이 입력 값으로서 반복된다. 이러한 반복 과정을 통하여 정해진 횟수만큼 최종 암호화가 끝난 후에 암호문과 final data가 출력되고 그중에서 final data는 key와 함께 xor 연산을 거쳐 다시 암호화 된다. 암호화된 final data값은 복호화할 때, 다시 xor연산을 거친 후에 시간 t-1의 셀 열로 입력되고, 마찬가지로 암호문은 시간 t일 때의 셀 열에 입력된다. 따라서 복호화 과정은 암호화 과정의 역순이 된다. (그림 3)의 블록 암호 구조는 다음 장의 RCA8과 RCA6, RCA4L, 그리고 RCA4R에 적용된다.

다음 절에서는, 2장에서 설명한 8개의 rule (rule 86, 90, 101, 105, 150, 153, 165, 1436194405)과 3.2절에서 설명한 reversible rule의 특성, 그리고 블록 암호 알고리즘 기본 구조를 적용한 전체 설계 과정에 대해 살펴본다.

3.4 경량 블록 암호 설계

전체 암호화 과정은 16라운드로 이루어지며 8비트 크기의 블록이 반복되는 구조이다. 따라서 전체 128비트의 평문이 암호화된다. 전체 키는 22비트로서 RCA8의 8비트와 RCA6의 6비트, 그리고 RCA4L 및 RCA4R의 4×2비트로 구성된다.

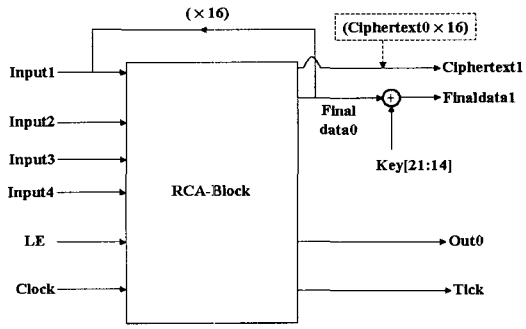


(그림 4) 단일 라운드 암호화 블록 다이어 그램 (RCA-Block)

(그림 4)는 단일 라운드에서의 암호화 과정인 "RCA-Block"을 보여준다. 이것은 크게 RCA8과 RCA4L, RCA6, RCA4R로 구성되고 중간 단계에서 shift-left 또는 shift-right 연산 과정과 치환 과정(substitution)을 거친다. Input1은 시간 t-1일 때의 initial data를, 그리고 Input2는 시간 t의 평문 값을 나타내며, Input3과 Input4는 shift 연산의 횟수를 임의로 결정하는 초기 데이터 값으로서 각각 Input1, Input2와 같은 기능을 수행한다. Tick은 RCA8과 같은 블록, 또는 단일 라운드의 암호화 과정과 같은 반복 구조에서 한번의 라운드가 끝났을 때의 동기 신호를 의미한다. Ciphertext0는 단일 라운드의 8비트 암호문이고 Finaldata0는 최종 출력 블록 데이터이다. Out0는 복호화시에 shift연산을 역으로 적용할 때 사용되는 데이터 값이다. 단일 라운드의 블록 알고리즘은 신호 clock과 LE에 의해 동기화 된다. 각 블록 RCA8, RCA4L, RCA6, RCA4R의 역할은 다음과 같이 표현된다.

- RCA8 : 평문을 initial data와 함께 정해진 횟수인 19번 동안 반복해서 rule 153R(rule 153과 102)과 함께 암호화한다. 하드웨어 동기를 위해 암호화가 끝났을 때의 신호 Tick가 블록 RCA4L에 입력되며, (그림 3)의 cipher text와 final data에 해당하는 8비트의 셀 열이 4비트씩 나눠진 후, 각각 shift 연산((그림 4)의 Shift L 또는 Shift R)과 치환 연산을 한다.
- RCA6 : 6비트 크기의 데이터가 rule 165R (rule165와 90)과 함께 19번 동안 암호화 되며, 오직 출력된 final data((그림 3) 참조)의 가운데 3비트 값으로부터 얼마나 shift 연산을 수행하는지에 대한 횟수를 결정한다. 예를 들어, 이동 횟수를 결정하는 비트의 크기는 3비트 이므로, 최소 000_2=0에서 최대 111_2=7 횟수만큼 shift 연산한다.
- RCA4L(혹은 RCA4R) : 4비트 크기의 데이터가 rule 90R(혹은 RCA4R일 경우, rule 101R)과 함께 22번 동안 암호화된다. RCA4L과 RCA4R의 결과 값들이 다시 8비트 크기의 데이터 블록, Ciphertext0와 Finaldata0((그림 3)의 encypted final data 참조)으로 출력된다.

그리고 이러한 단일 라운드의 암호화 과정은 16번 동안 반복되고 다음 (그림 5)에서 전체 경량 블록 암호화 구조를 보여주고 이를 "RCA cipher"라고 한다.



(그림 5) 16 라운드의 암호화 블록 다이어그램 (RCA cipher)

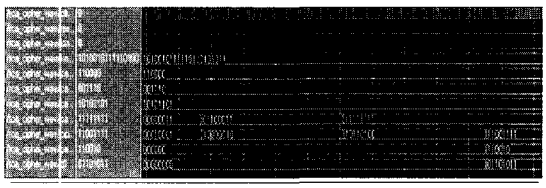
clock 신호와 함께, Load-Enable 신호와 연결된 Tick 신호는 다음 라운드를 위한 동기 신호이며, (그림 4)의 Ciphertext0는 16번 동안 반복된 다른 값들이 128비트 크기의 Ciphertext1로 출력되고, 최종 Finaldata0이 key값 중에서 15번째와 22번째의 비트 값들과 함께 xor연산을 통해 Finaldata1로 암호화되어 출력된다. 한편, 전체 복호화 과정은 (그림 5)의 블록 다이어그램의 역순으로 구성된다.

4. 결과 및 암호 분석

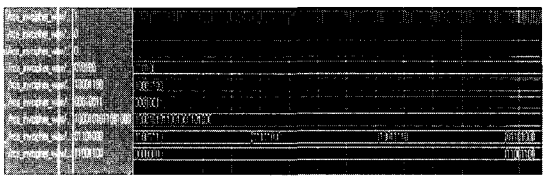
4.1 구현 결과 분석

본 논문에서 구현된 암호 알고리즘은 최대 주

파수 81.686MHz에서 동작하며, 128비트의 데이터를 암호화 및 복호화하는 동안 처리속도가 704 클럭 사이클이었다. 그리고 암호화 과정에서 2,874 게이트 수와 복호화 과정에서는 2,282 게이트 수로 측정되었다. 본 논문에서는 VHDL과 FPGA Xillinx를 이용하여 구현하였고, (그림 6)은 시뮬레이션 결과 파형들을 보여준다.



(a) 암호화 결과 파형



(b) 복호화 결과 파형

(그림 6) 시뮬레이션 결과 파형

<표 2>는 NTRU[18] 및 AES[15]의 8비트 크기 블록으로 구현된 블록 암호 알고리즘들과의 하드웨어 측정 결과의 비교를 보여준다.

〈표 2〉 NTRU 및 AES와의 결과 비교(E:암호화, D:복호화)

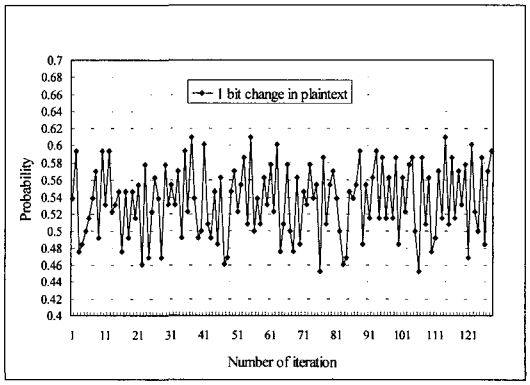
8비트 구현	게이트 수	클럭 사이클
RCAcipher	2,874E(2,282D)	704E(704D)
AES [15]	3,595E	1016E
NTRU [18]	2,975E	722,401E

분석 결과, 암호화 과정에서 AES와 비교했을 때, 20% 게이트수의 절감과 31%의 클럭 사이클 만큼 향상된 처리시간을 보여줬다. NTRU와 비교 했을 때, 약 100개의 게이트수가 줄었고 클럭 사이클에서 상당한 차이를 보였다. 이것은 본 논문의 RCA기반 경량 블록 암호 알고리즘이 다른 두 블록 암호 알고리즘보다 어려운 제약조건에서도 구현이 가능하다는 것을 의미한다[8].

4.2 암호 해독

본 절에서는 구현된 결과에 대한 암호 해독을 수행함으로써 암호 공격에 대한 암호 안정성을 테스트하는데 목적을 둔다.

Strict Avalanche Criterion(SAC)은 일반적인 암호 알고리즘에 대한 암호 안정성을 평가하기 위해 사용되는 기법이다. A. F. Webster외 [11]가 처음 제안하였는데, "하나의 함수가 SAC를 만족한다면, 그 함수의 입력 값 중에서 한 비트를 변경했을 때, 결과 값들이 기존 결과와 비교하면 절반정도가 변한다."라고 제안하였다. (그림 7)은 SAC 결과 값들에 대한 그래프를 보여준다.



(그림 7) SAC 시뮬레이션 결과

SAC의 결과 값은 전체 128비트 가운데, 0.535의 평균값과 최소 값 0.453, 그리고 최대 값 0.609를 보여줬다. 이 결과는 SAC를 충분히 만족하며, 따라서 우수한 암호 안정성을 보여준다[9].

차분분석(Differential Cryptanalysis)은 전형적인 블록 암호 알고리즘의 암호 해독에 사용되

는 기법으로서 Bilham 외[17]가 처음 제안하였다. 본 논문에서는, DES의 S-box와 유사한 역할을 수행하는 RCA8, RCA6, RCA4L, RCA4R의 4개 블록에 대해 차분 분석을 실행하였다. 각 블록 구조들은 8비트와 6비트, 4비트의 크기로 이루어져 있기 때문에 각각 256 ($=2^8$), 64 ($=2^6$), 16 ($=2^4$), 16 ($=2^4$)번 동안 차분 분석을 수행하였다.

<표 3>은 initial data값이 18일 때, RCA8의 차분 분석의 결과 값을 보여준다. 차분 분석을 위해서 256개의 initial data를 대입하여 실험하였다. 세로 열은 입력 차분이고 가로 열은 출력 차분을 나타낸다. 각각의 값은 입력 차분과 출력 차분에 대한 차분 횟수를 의미한다. 차분 횟수를 합한 총 차분 분석의 합계가 13,323개였으며, 평균 12,950개로 나타났다. 이것은 전체 66536(=256×256)개에서 20.024%와 평균 18.582%를 차지한다. <표 4>와 <표 5>, <표 6>은 각각 RCA4L, RCA4R, RCA6에 대한 차분 분석의 결과를 보여준다. RCA4L에서는 평균 57.422%였고, RCA4R은 24.220%, 그리고 RCA6은 22.950%의 결과를 보여줬다. 이는 전체적으로 차분 분석의 조건을 만족한다[10].

〈표 3〉 RCA8의 차분 분석

Initial Data : (00010010)						
입력 \ 출력	0	1	…	254	255	차분 횟수
0	256	0	…	0	0	1
1	2	16	…	0	0	28
…	…	…	…	…	…	…
254	0	2	…	0	0	28
255	2	16	…	0	0	28
합 계						13323

〈표 4〉 RCA4L의 차분 분석

Initial Data : (0111)						
입력 \ 출력	0	1	…	14	15	차분 횟수
0	16	0	…	0	0	1
1	1	2	…	1	0	13
…	…	…	…	…	…	…
14	0	1	…	0	0	11
15	1	2	…	1	0	13
합 계						147

〈표 5〉 RCA4R의 차분 분석

Initial Data : (00010010)						
입력 \ 출력	0	1	…	14	15	차분 횟수
0	16	0	…	0	0	1
1	0	8	…	0	0	4
…	…	…	…	…	…	…
14	0	0	…	0	0	3
15	0	8	…	0	0	4
합 계						62

〈표 6〉 RCA6의 차분 분석

Initial Data: (00010010)						
입력 \ 출력	0	1	…	62	63	차분 횟수
0	64	0	…	0	0	1
1	0	0	…	0	0	10
…	…	…	…	…	…	…
62	0	0	…	0	0	11
63	0	0	…	0	0	10
합 계						940

5. 결 론

유비쿼터스 컴퓨팅 환경은 기존의 네트워크보다 보안에 매우 취약한 특징을 가지고 있다. 또한 유비쿼터스 환경의 핵심 기술 중의 하나로서 RFID의 하드웨어 경량화 구현을 위한 효율적인 설계가 필요하며 이것은 설계 비용의 절감을 가져온다. 본 논문에서는 RCA 기반의 경량 블록 암호 알고리즘을 구현하였다. 구현된 블록 암호

는 AES[15]와 NTRU[18]등과 비교했을 때, 하드웨어적인 측면에서 20%의 게이트수가 절감되었고, 클럭 사이클이 감소함으로써 처리 속도는 31%만큼 향상되었다. 그리고 SAC와 차분 분석을 수행한 결과, 암호 해독에 대한 암호 안정성을 만족하였다. 결론적으로 본 논문에서 제안한 RCA 기반 블록 암호의 구현은 기존 블록 암호 알고리즘보다 유비쿼터스 환경을 실현하기 위한 효율적인 하드웨어 설계에 있어서 우수함을 증명한다.

참 고 문 헌

[1] J. Von Neumann, "Theory of self-reproducing automata", University of Illinois, Press Urbana, 1966.

[2] S. Wolfram, "Cellular Automata and Complexity", Addison Wesely Publishing Company, 1994.

[3] P. P. Chaudhuri, A. R. Chowdhury, S. Nandi, and S. Chattopadhyay, "Additive Cellular Automata, Theory and Applications", IEEE Computer Society Press, Vol. 1, 1997.

[4] S. Bhattacharjee, S. Sinha, C. Chattohyay, and P. P. Chaudhuri, "Cellular Automata based Scheme for Solution of Boolean Equations", IEEE Proceedings, Computer and Digital Techniques, Vol. 143, No. 3, 1996.

[5] M. Mihaljevic and H. Imai, "A Family of Fast Keystream Generations based on Programmable Linear Cellular Automata over GF(q) and Time-Varient Table", IEICE Transactions on Fundamentals, Vol. E82-A, No.1, pp. 32-39, 1999.

[6] M. Mihaljevic, Y. Zhang, and 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.

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

[8] S. E. sarma, S. A. Weis, and D. W. Engels, Radio-Frequency Identification, "Security Risks and Challenges", RSA Laboratories Cryptobytes, Vol. 6, No. 1, pp. 2-9, Spring 2003.

[9] M. Seredynski, K. Pienkosz, and P. Bouvry, "Reversible Cellular Automata Based Encryption", Network and Parallel Computing, IFIP International Conference, NPC 2004.

[10] J. Lee, H. Jang, and K. Rhee, "A Block Cipher Algorithm based on Cellular Automata", Journal Multimedia, 2002.

[11] A. F. Webster and S. E. Tavares, "On the Design of S-Boxes, Advances in Cryptology", Crypto'85 proceedings, Springer, 1986.

[12] F. Seredynski, P. Bouvry, and Albert Y. Zomaya, "Cellular Automata Computations and Secret Key Cryptography", Parallel Computing, Vol. 30, pp. 753-766, 2004.

[13] M. sipper and M. Tomassini, "Generating Parallel Random Number Generators by Cellular Programming", International Journal of Modern Physics C, Vol. 7, No. 2, pp. 181-190, 1996.

[14] National Institute of Standards and Technology, Federal Information Processing Standards Publication 140-2 : Security Requirements for Cryptographic Modules, US Government Printing Office, Washington,

1999.

[15] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, "Strong Authentication for RFID Systems Using the AES Algorithm", CHES 2004, LNCS 3156, pp. 357-370, 2004.

[16] A. F. Webster and S. E. Tavares, "On the Design of S-boxes", Advances in Cryptology: Crypto'85 proceedings, Springer, 1986.

[17] E. Bilham, A. Shamir, "Differential Cryptanalysis of DES-like Cryptosystems", Proceedings at CRYPTO'90 Conference, 1990.

[18] C. M. O'Rourke, "Efficient NTRU Implementations", M. S. Thesis, Electrical & Computer Engineering, Worcester Polytechnic Institute, 2002.

[19] F. Bao, " Cryptanalysis of a Partially Known Cellular Automata Cryptosystem", IEEE Trans. Computers, Vol. 53, No. 11, November 2004.

[20] P. Guan, "Cellular Automata Public Key Cryptosystem", Complex System, Vol. 1, pp. 51-57, 1987.

[21] S. Nandi, B. Kar, and P. Chaudhuri, "Theroy and Applications of Cellular Automata in Cryptography", IEEE Trans, Computers, Vol. 43, No. 12, pp. 1346-1357, Dec. 1994.

[22] N. Ganguly, A. Das, B. Sikdar, and P. Chaudhuri, "Cellular Automata Model for Cryptosystem", Proc. Cellular Automata Conf., 2000.

[23] S. Sen, C. Shaw, R. Chowdhrui, N. Ganguly, and P. Chaudhuri, "Cellular Automata based Cryptosystem (CAC)", Proc. Fourth Int'l Conf. Information and Comm. Security (ICICS02), pp. 303-314, Dec. 2002.

[24] B. Srisuchinwong, T. A York, and Ph. Taslides, "A Symmetric Cipher using Autonomous and Non-autonomous Cellular Automata", IEEE Global Telecommunications Conference, GLOBECOM '95, pp. 1172-1177, 1995.

김 성 환

2004년 한국항공대학교 전자공학과(공학사)

2004년~현재 한국항공대학교 컴퓨터공학과 석사과정

김 동 성

2001년 한국항공대학교 전자공학과(공학사)

2003년 한국항공대학교 컴퓨터공학과(공학석사)

2003년~현재 한국항공대학교 컴퓨터공학과 박사과정

송 영 덕

1986년 수원대학교 전자계산학과(이학사)

2003년 청운대학교 전산전자공학과(공학석사)

2003년~현재 한서대학교 정보보호공학과 박사과정

박 종 서

1983년 한국항공대학교 항공통신학과(공학사)

1986년 노스캐롤라이나대학교 전기컴퓨터공학과 (공학석사)

1994년 펜실베이나주립대학교 컴퓨터공학부(공학박사)

1994년~1996년 펜실베니아주립대학교 컴퓨터공학과 조교수

1996년~현재 한국항공대학교 컴퓨터공학과 부교수