韓國 컴퓨터情報學會 論文誌
第8卷 第2號, 2003. 6.

2003-8-2-2-7

PRN을 이용한 키 스케쥴러 블록암호시스템 설계에 관한 연구

김 종 협* 김 환 용**

A Study on the Design of Key Scheduler Block Cryptosystem using PRN

Jong-Hyup Kim* Hwan-Yong Kim**

요 약

정보보호 및 암호기술은 IT 산업과 더불어 매우 많은 발전을 이룩하였지만 실시간 처리 및 비화성 유지 등은 아직도 해결해야 하는 문제점이다. 그러므로 본 논문에서는 표준화된 AES인 Rijndael에 대하여 비도 증가 및 처리율 증가를 위한 새로운 PRN-SEED 암호알고리즘을 제안하였으며 Rijndael 및 다른 AES와 비교하여 성능분석을 수행하였다. PRN-SEED 암호알고리즘의 구현은 Synopsys Design Analyser Ver. 1999.10과 삼성 KG75 library 그리고 Synopsys VHDL Debegger를 사용하였다. 모의실험 결과, 대칭형 암호시스템인 DES는 동작주파수가 40㎒일 경우 416Mbps의 처리율을 가지며, Rijndael 암호시스템은 동작주파수가 50㎒일 경우 612Mbps의 처리율을 가진다. PRN-SEED 암호시스템의 전체 게이트 수는 10K이며 동작주파수가 40㎒일 때 128 비트에 대한 처리율은 430Mbps, 50㎒일 때 128 비트에 대한 처리율은 630Mbps였다.

Abstract

Information protection and cryptography technology is developed with IT but solved problem of real time processing and secret maintain. Therefore this paper is proposed new PRN-SEED(Pseudo-Random Number-SEED) for the increasing secret rate and processing rate perform performance analysis with existed other cryptography algorithms. Proposed new PRN-SEED crypto-algorithm increase in the processing rate than existed algorithms use bit and byte mixed operation with RNG(Random Number Generator). PRN-SEED that performs simultaneous operations have higher 1.03 in the processing rate and 2 in the cryptosystem performance than existed cryptosystems. Implementation for PRN-SEED use Synopsys Design Analyser Ver. 1999.10, samsung KG75 library and Synopsys VHDL Debegger. As a simulation result, symmetric cryptosystem DES operate 416Mbps at the 40㎒ and Rijndael operate 612Mbps at the 50㎒. PRN-SEED cryptosystem have gate counting 10K and operate 430Mbps at the 40㎒ and 630Mbps at the 50㎒.

* 동아인재대학 인터넷정보처리과 부교수, ** 원광대학교 전자공학과 교수

Ⅰ. 서론

이동 및 무선통신망, 전자상거래 등은 IT 산업에 대한 하나의 실례가 되는 동시에 실생활에 없어서는 안 되는 매우 중요한 생활의 일부가 되었다. 그러나 실시간 처리 및 비화성 유지 등은 아직도 해결해야 하는 걸림돌이다 [1][2][3][4][5].

2000년도에 발표된 Rijndael은 DES에 대한 보안성 약화로 인한 대체용 블록 암호시스템으로서 NIST에서 차세대 AES(Advanced Encryption Standard)로서 결정하였다. 제안된 Rijndael 시스템은 기본적으로 DES와 처리율 면에서 유사하기 때문에 기존 대칭형 암호방식에 적용되는 크랙방식이 적용되므로 비도 및 처리율 향상을 기대하기는 어렵다. 본 논문에서는 표준화된 AES인 Rijndael에 대하여 비도 증가 및 처리율 증가를 위한 새로운 PRN-SEED (Pseudo-Random Number SEED) 알고리즘을 제안하였으며 Rijndael 및 다른 AES와 비교하여 성능분석을 수행하였다. 제안된 새로운 PRN-SEED 암호알고리즘은 기존 대칭형 암호알고리즘에 비하여 처리속도가 더욱 빨라졌으며 비트 연산과 바이트 연산을 혼합함과 동시에 RNG(Random Number Generator)를 사용하였기 때문에 비도 측면에서 매우 우수한 특성을 보였다. 또한 ECB 모드에서 동작하더라도 동일한 평문에 대한 동일한 암호문이 생성되지 않도록 하였다.

Ⅱ. 기존 암호 알고리즘

1. DES 암호알고리즘

1977년 DES(Data Encryption Standard) 표준안이 결정된 이후 군사분야, 외교분야 및 금융 네트워크와 같은 상업분야로 폭넓게 사용되고 있다. 블록 암호시스템에서 사용되는 암호방식은 일반적으로 암호화를 수행하는 방식에 의하여 ECB, CBC, CFB, OFB로 분류된다 [13][7][8][9].

DES는 기본적으로 페이스텔 구조와 SPN (Substitution and Permutation Network)를 사용하기 때문에 i 번째 라운드 결과를 $X_i=(L_i, R_i)$라고 할 경우, L과 R에 대한 표현은 식 (1)과 같은 형태를 가진다.

$$L_i = R_{i-1}$$
$$R_i = L_{i-1} \oplus f(R_{i-1}, K_i) \quad \cdots\cdots\cdots (1)$$

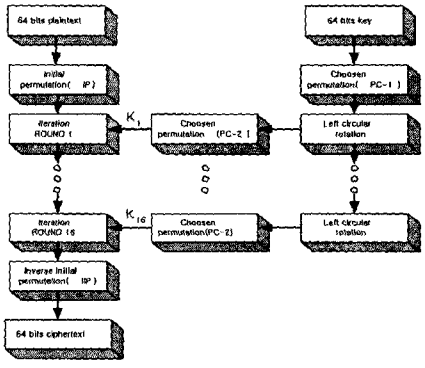

그림 1. DES 암호알고리즘 구성
Fig. 1 Structure of DES crypto-algorithm

식 (1)은 <그림 1>과 같은 페이스텔 구조와 SPN을 이용하여 암호화를 수행하기 때문에 발생되는 수식으로서 L, R로 분리된 좌우의 데이터가 페이스텔 구조 내부의 F 함수에서 키 데이터와 비선형 결합을 수행함으로써 나타나게 된다.

키 스케줄은 64 비트 키 중 8 비트의 패리티비트 (parity bit)를 제외한 56 비트를 선택재배열하고, 28 비트씩 좌/우 두개로 나누어 키 스케줄에 의해 각각 1 비트 또는 2 비트씩 미리 정해진 표에 의하여 좌측이동 동작을 수행하게 된다. 이후 56 비트 키는 두번째 선택재배열을 수행하게 되며 미리 정해진 표에 의하여 48 비트를 선택하고 Feistel 구조의 F 함수에 대한 각 라운드의 서브키로 사용된다. 치환과 재배열과정으로 구성된 암호함수 F 내부의 비선형 함수인 S-box는 암호화 강도를 높이는 기능을 수행하게 된다. DES에서 S-box는 8개로 구성되어 있으며 각 S-box는 6 비트의 입력을 받아 4 비트의 비선형 출력값을 산출한다[10][11][12].

키 스케줄은 최초 사용자가 제공하는 64 비트의 키에서 8,16,24,32,40,48,56,64번째 비트를 제거하고 남은 56 비트의 키가 암호화에 사용되고 8 비트는 패리티비트로 활용된다. 〈그림 2〉와 같이 56 비트 키를 각 비트의 위치를 바꾸어 재배열하고 재배열된 56 비트 키를 28 비트씩 양분(C, D)한 다음 각 라운드별로 정해진 룰(rule)에 의하여 1 또는 2 비트씩 왼쪽순환이동 시키고, 각각의 48 비트에서 24 비트를 선택하여 재배열한다. 이와 같이 결정된 48비트가 암호함수의 서브키로 사용된다[7][8][9].

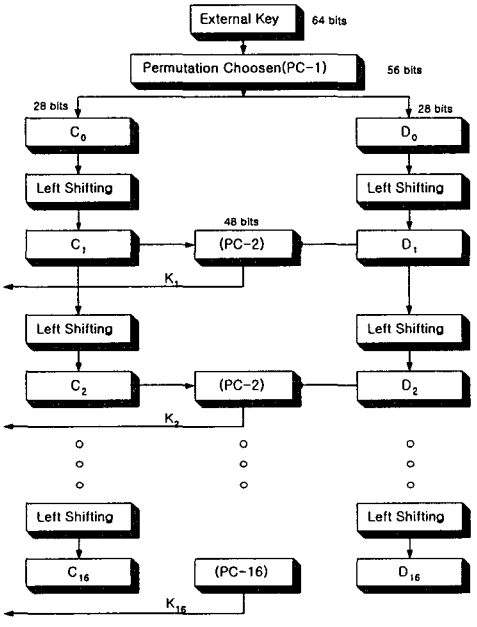

그림 2. DES의 키 스케줄
Fig. 2 DES key schedule

2. 스트림 암호알고리즘

스트림 암호시스템(stream cipher system)은 두 가지 형태로 분류된다.

첫째, 동기 스트림 방식(synchronized stream cipher)은 키 비트 스트림(key bit stream)과 평문이 독립적인 것으로써 암호문을 성공적으로 복호화해서 올바른 평문을 산출하는 것으로서 키 비트 스트림과 암호문 사이에 동기가 필요하다. 둘째, 자동동기 스트림 암호방식(self-synchronized stream cipher)은 키 비트 스트림과 평문 또는 암호문에 대하여 종속적인 함수로 구성되어 있는 것으로써 암호문에 삽입 또는 제거된 오류 비트(error bits)로 인하여 몇 개의 한정된 오류가 복호된 평문에 발생하지만 연속되는 암호문에 오류가 발생되지 않는 경우 올바른 평문으로 다시 복귀되는 특성을 가진다[6].

스트림 암호방식은 비트 단위로 암호화를 수행하며 기본연산은 배타적 논리합과 지연소자이다. 그러므로 LFSR은 의사난수와 동일하다. 의사난수(Pseudo Random Number : PRN)는 비밀키 암호시스템의 세션키(session key) 또는 초기값(Initialize Vector : IV) 생성, 공개키 암호나 디지털 서명의 공개키/비밀키 및 시스템 변수의 생성, SSL(Secure Socket Layer)과 같은 보안 통신 프로토콜에서 사용되는 Random Challenge 등 각종 암호시스템 전반에 걸쳐 사용되고 있다. PRN은 컴퓨터 프로그래밍으로 계산되며 결정적인 성질을 갖는다.

대칭형 암호알고리즘이나 단방향 해쉬알고리즘(hash algorithm)이 랜덤한 값을 생성한다고 가정할 때, 의사난수에 대한 안전성은 입력으로 사용되는 seed 값에 대한 엔트로피(entropy)와 seed 및 state 정보가 비인가자로부터의 안전한 보관 및 유지에 달려 있다. 즉 LFSR의 초기값 및 키의 안전한 관리가 필요하다.

유한체 $GF(2)=0,1$에서 정의되는 스트림 수열은 식 (2)와 같이 정의된다.

$$s_{j+n}=(c_0 s_{j+n-1}+c_1 s_{j+n-2}+\cdots+c_{n-1}s_j) \bmod 2 \quad \cdots\cdots (2)$$

여기에서 $s_0, s_1, s_2, \cdots, s_{n-1}$은 초기상태를 정의하게 된다.

일반적으로 유한체 위에 정의된 다항식의 특성은 그 다항식의 차수(degree)와 위수(order)에 의하여 결정된다. 특히, 위수는 다항식에 의하여 생성되는 수열의 주기와 밀접한 관련이 있다. n차 다항식 $f(x)$에 의하여 생성되는 수열의 주기는 $f(x)$의 위수의 약수이며 이때, $f(x)$가 기수인 경우 $f(x)$의 위수와 동일하고 이러한 위수는 2^n-1의 약수를 가진다. 위수가 정확히 2^n-1인 다항식을 원시다항식(primitive polynomial)이라 부른다.

3. Rijndael 암호알고리즘

DC(Differential Cryptanalysis), LC(Linear Cryptanalysis)와 같은 암호해석방법의 발달로 인하여 DES에 대한 안전성 문제가 심화됨에 따라 Rijndael과 같은 DES 대체용 블록암호알고리즘이 2000년에 새로운 표준안으로 NIST에 의하여 채택되었다. Rijndael 암호알

고리즘은 DES와 같은 블록암호알고리즘이지만 데이터 블록 및 키의 크기는 128, 192, 256 비트들로 가변이 가능하도록 되어 있으며 비선형 결합에 의존하던 DES와 달리 유한체 GF를 사용하여 S-box를 구성하고 있는 점이 DES와 구별되는 점이다[14][15].

Rijndael 암호알고리즘에서 입력블록과 출력블록 및 State의 크기는 각각 128, 192, 256 비트 중 하나이며 Nb=4, 6, 8로 표현되고 State에서 32 비트 word들인 열의 수를 나타낸다. 키 크기는 128, 192, 256 비트로써 Nk=4, 6, 8로 표현되며 암호키에서 32 비트 word들인 열의 수를 나타낸다.

〈표 1〉은 Rijndael에서 Nb, Nk 크기에 따라 일정한 비도를 유지하기 위해서 필요한 라운드 수를 나타낸다.

표 1. 블록 및 키 길이에 따른 필요 라운드 수
Table 1. Necessary round number related block and key length

필요 라운드 수		블록 크기		
		Nb=4	Nb=6	Nb=8
키 크기	Nk=4	10	12	14
	Nk=6	12	12	14
	Nk=8	14	14	14

Rijndael은 바이트 단위의 변환으로 구성된 라운드를 이용하여 암호화 및 복호화를 수행한다.

i) State 배열의 S-box를 이용하는 바이트 치환 (SubByte)
ii) State 배열의 행 이동(ShiftDiagonal)
iii) State 배열의 각 열에 있는 바이트들의 혼합 (MixColumn)
iv) State 배열과 라운드 키의 덧셈(AddRoundKey) 평문 입력은 State 배열에 저장된다. State 배열은 초기 라운드 키와 덧셈을 수행한 후 라운드를 수행하게 된다. 모든 라운드가 실행되면 State 배열은 출력배열에 저장되어 암호화를 마치게 된다. 이러한 Rijndael에 대한 전체적인 흐름은 그림3과 같다.

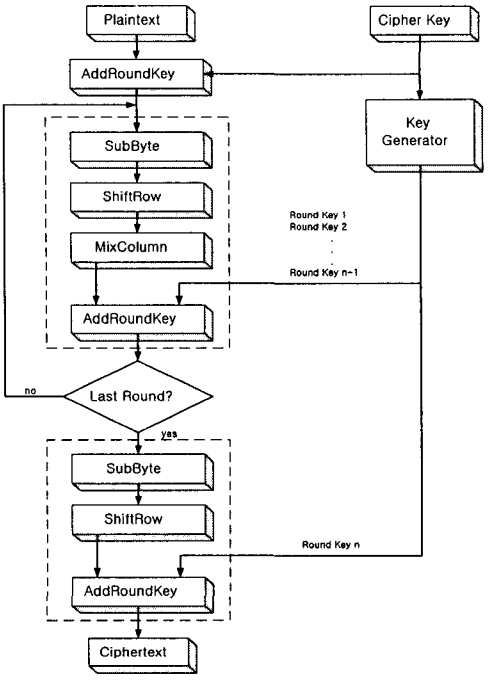

그림 3. Rijndael 암호알고리즘
Fig. 3 Rijndael crypto-algorithm

Ⅲ. PRN을 포함하는 PRN-SEED 암호알고리즘

대칭형 암호시스템에 대한 처리시간 및 비도 증대 측면을 위하여 본 논문에서는 처리시간 및 비도, 그리고 구현상의 문제점을 보다 용이하게 해결하기 위하여 PRN-SEED 암호알고리즘을 새롭게 제안하였다. 제안된 PRN-SEED 암호알고리즘은 배타적 논리합을 사용하며 처리 수행단위는 바이트를 사용하였다. 바이트 연산은 처리속도를 매우 높게 수행할 수 있다는 장점과 더불어 역추적이 어렵기 때문에 비도 증가에도 매우 우수한 특성을 가진다. 또한 기존 페이스텔 구조와 SPN을 사용하여 암호화를 수행할 때, 암호화와 복호화의 동시 수행이 가능하기 때문에 암호화를 수행하는 시스템에서 Rijndael 또는 Serpent 암호알고리즘과 같은 효율 저하가 발생하지 않는

다. 이러한 특징은 AES에 대한 충분한 전제조건을 만족함과 동시에 AES 다음 버전에 대한 내용을 제시할 수 있다. 이러한 특징으로 인하여 PRN-SEED 암호알고리즘은 실시간 처리 및 비도 그리고 구현상의 문제점을 해결할 수 있다.

PRN-SEED 암호알고리즘에 사용되는 입력블록과 출력블록의 크기는 128 비트이며 키 크기도 128 비트로서 평문, 암호문 그리고 키의 크기는 1:1:1이 된다. PRN-SEED 암호알고리즘은 다른 AES 암호알고리즘들과 마찬가지로 다음과 같은 네 가지 기능블록을 포함하며 각 단계를 거치는 동안 바이트 단위의 변환으로 구성된 라운드를 이용하여 암호화 및 복호화를 수행한다.

i) 조건 상태 배열(CSA : Condition State Array) 기능을 가진 S-box를 이용하여 바이트 치환 수행 기능(Inv/SubByte)
ii) CSA에 대한 행 방향 이동기능(Inv/ShiftDiagonal)
iii) CSA의 각 열에 해당하는 바이트들의 혼합기능 (Inv/MixColumn)
iv) CSA와 라운드 키에 대한 1:1 덧셈기능 (AddRoundKey)

평문 및 암호문 128 비트의 입력은 CSA 상태로 초기 저장된다. CSA는 PRN-SEED 암호알고리즘을 형성하기 위한 비선형 특성을 가진 상태를 의미한다. CSA는 식 (3) 및 〈그림 4〉와 같이 외부에서 주어지는 파라미터를 가지고 현재상태를 결정하며 결정된 현재상태는 불확실한 미래상태를 형성하는 기준 값으로 설정된다.

$$CSA_{next} <= CSA_{present}(prn_{seed} \bmod 8) \cdots (3)$$

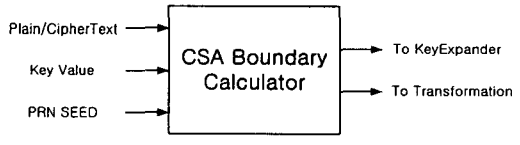

그림 4. CSA 특성
Fig. 4 CSA characteristic

CSA는 입력으로 사용되는 여러 가지 파라미터를 이용하여 키 값을 생성하는데 필요한 자료를 만드는 동시에 암/복호화하는 과정 중에 필요한 치환변환을 제어하는 기능을 수행한다.

식 (3)과 같이 mod 8은 입력 데이터들에 대한 모듈러 연산을 의미하는 것으로서 입력 값들에 대한 포맷을 바이트로 변환하기 위한 과정이다. 또한 PRN(Pseudo Random Number) SEED는 식 (4)와 같이 입력 데이터와 키 값을 이용하여 생성한다.

$$SEED = INPUT \oplus KEY_{7,15,\cdots,111,119,127} \cdots (4)$$

식 (4)에 의하여 생성된 SEED는 식 (5)와 같은 PRN을 통하여 2 바이트의 출력 값을 산출하게 된다.

$$PRN = PRN_{odd} \parallel PRN_{even}$$
$$PRN_{odd} = x^{16} + x^{13} + x^{12} + x^{11} + x^{7} + x^{6} + x^{5} + x^{4} + 1$$
$$PRN_{even} = x^{16} + x^{14} + x^{10} + x^{9} + x^{8} + x^{6} + 1 \quad \cdots\cdots (5)$$

산출된 2 바이트 중 odd 정보는 PRN-SEED의 내부 바이트 치환 제어에 사용되어지며 even 정보는 키 확장 제어에 사용된다.

odd와 even 정보는 예측 불가 함수를 생성하는 기능을 가짐으로서 내부 치환정보가 외부로 유출될 가능성이 적으며 입력 데이터와 키 정보만을 가지고 생성된 것이기 때문에 별도의 프로세싱이 필요 없게 된다.

식 (5)는 이동통신망에서 사용되는 PRN과 동일한 방정식으로서 2 바이트를 하나의 방정식으로 간주하여 출력 값을 산출하게 된다. 이러한 PRN 방정식을 사용하는 이유는 멀티미디어의 발전으로 인하여 연계서비스의 증가와 응용 서비스가 무선통신망을 기준으로 확산되는 추세이기 때문이다. CSA는 초기 라운드 키와 배타적 논리합을 수행한 후 n번의 라운드를 수행하게 된다. 모든 라운드가 실행되면 암호화 및 복호화를 마치게 된다. 이러한 PRN-SEED 암호알고리즘에 대한 전체적인 흐름은 그림 4와 같다.

PRN-SEED 암호알고리즘의 가장 큰 특징은 그림 5에서 보는바와 같이 암호화 및 복호화가 동시에 수행된다는 점이다. 제어신호에 의하여 암호화 모드, 복호화 모드가 결정되며 처리되어지는 연산은 순서만 역으로 동작한다.

각 라운드마다 Inv/SubByte, Inv/ShiftDiagonal, Inv/MixColumn, Inv/ AddRoundKey에 대한 데이터 값들은 CSA & PRN에 의하여 별개로 동작하게 된다. 그러므로 라운드 수에 따라서 비도가 결정된다. 그러므로 라운드 수와 비도와는 비례관계를 가진다.

이러한 네 가지 변환을 SOLO라고 정의하면 식 (6)과 같다.

기존 AES들은 데이터와 키의 길이를 128, 192, 256 비트들로 가변시키며 변화하는 길이에 따라 최적화된 라운드 수를 결정하게 된다. 그러므로 기존 AES인 경우 데이터의 블록길이에 따라 라운드 수가 결정된다. 이러한 결과로 인하여 데이터의 심볼 크기를 파악하게 되는 경우 라운드 수를 파악할 수 있으며 라운드 수와 키 및 데이터와의 DC 및 LC에 의하여 크래킹이 가능해진다. 그러나 PRN-SEED인 경우 고정된 블록 및 키 크기를 가지고 있어도 내부적으로 라운드 수에 따라 데이터 내용이 변화되므로 라운드 수를 데이터 심볼 크기만을 가지고 파악할 수 없다는 장점이 있다.

일반적인 암호알고리즘은 키 생성 알고리즘 및 키 스케줄링 작업을 수행한다. 이러한 키 스케줄링 작업은 보다 복잡한 키를 생성하기 위한 수단으로서 알고리즘의 안전도에 절대적인 역할을 수행한다.

$$
\begin{aligned}
SOLO_{n-round} \\
&<= Inv|SubByte(odd)_n + \\
&\quad Inv|ShiftDiagonal(odd)_n + \\
&\quad Inv|MixColumn(odd)_n + \\
&\quad AddRoundKey(odd)_n \\
SOLO_{(n-1)-round} \\
&<= Inv|SubByte(even)_{n-1} + \\
&\quad Inv|ShiftDiagonal(even)_{n-1} + \\
&\quad Inv|MixColumn(even)_{n-1} + \\
&\quad AddRoundKey(even)_{n-1}
\end{aligned}
\quad \cdots\cdots (6)
$$

암호알고리즘의 분류도 키의 생성방법에 따라 분류할 정도로 키 생성은 매우 중요한 정보보안의 파라미터이다.

PRN-SEED 암호알고리즘의 경우, 키 스케줄링을 라운드에 따라 변화하는 방법을 사용하지 않고 단지 CSA & PNR 방법을 사용하여 생성한다. 암호문을 생성하고자 하는 평문의 일부를 이용하여 PNR의 SEED로 사용하고 PNR은 각 event가 발생할 때마다 라운드과정으로 인식하여 암호화에 필요한 키를 생성하게 된다. 이러한 키 생성은 각 라운드마다 필요한 별개의 키를 효율적으로 생성할 수 있으며 생성된 키를 이용하여 암호화를 수행할 경우 라운드를 구별하는 기준점을 별도로 설정할 필요성이 없어지게 된다. 키 생성은 PNR의 event 발생 때마다 변화하는 값을 이용하며 내부의 암호문은 네 단계의 변환을 수행할 때마다 변화하는 동시에 PNR의 값이 변환을 조절하게 된다. 이러한 이유로 다른 암호알고리즘에서는 키와 암호문과의 크기 조절을 위하여 키 길이의 확장 및 축소과정을 거치게 되지만 PRN-SEED 암호알고리즘은 이러한 키 길이의 조작을 수행할 필요성이 없다.

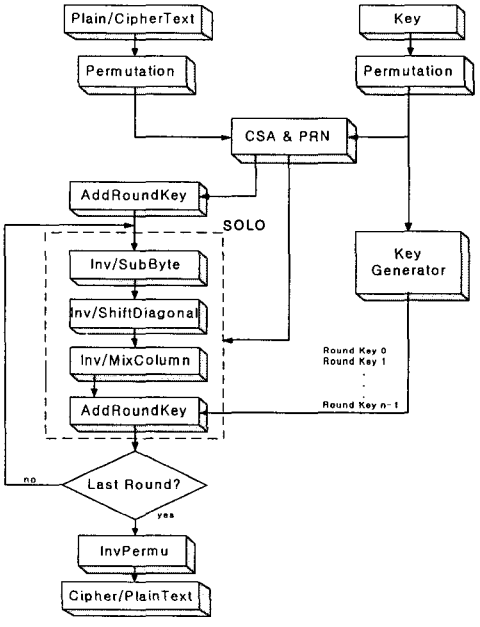

그림 5. PRN-SEED 암호알고리즘
Fig. 5 PRN-SEED cryptoalgorithm

Ⅳ. 시스템 설계 및 모의실험

CSA & PRN 처리부는 제안된 PRN-SEED 암호알고리즘의 핵심부분으로서 실제적인 PRN- SEED 암호알고리즘을 수행하게 되는 부분이다. CSA 블록은 입력 데이터들과 PRN-SEED 암호알고리즘에 대한 상태 조건을 바탕으로 조건상태를 산출하게 되며 산출된 조건 상태 배열들인 CSA는 바이트 치환을 수행하게 된다. 식 (3)과 같

이 CSA에 대한 상태 변화는 의사랜덤 발생기와 mod 연산을 통하여 변화하게 된다. 이때 입력으로 사용되는 데이터들은 암호문 또는 평문과 키 값 그리고 의사랜덤 발생기의 SEED값이다. 입력 데이터 128 비트에 대하여 키 값 역시 128 비트이며 키 값 128 비트 중 일부 데이터는 입력데이터와 2진곱 연산을 수행한 후 의사랜덤 발생기 SEED의 값으로 사용된다. 〈그림 6〉은 의사랜덤 발생기 SEED값 발생을 수행하는 기능 블록으로서 키와 키의 일부가 입력으로 사용되어진다.

그림 6. 의사랜덤 발생기 SEED 발생부
Fig. 6 PRNG SEED generator

암호모드, 복호모드에 따라서 PRNG의 even, odd가 결정되며 결정된 PRNG는 키 정보를 SEED값으로 받아 랜덤한 키 정보를 출력한다. 이때 출력되어지는 키 정보는 암호문 또는 복호문의 일부와 전처리 SEED 연산을 수행하게 된다. 이때 SCS에 의한 PCS 정보를 산출한다. 산출된 PCS 데이터는 SEED 포맷과정을 거치게 된다. SEED 포맷과정을 거친 키 정보는 SOLO 블록 내부의 라운드 키의 덧셈블록의 입력으로 사용된다. 이때 PRN-SEED 암호시스템의 라운드는 10회로 결정되어 있다. 10회 연산이 이루어진 후 round 판별과정을 거친 후 마지막 치환과정을 거친후 복호문 또는 암호문을 생성하게 된다. 이와 같이 SOLO 블록에 대한 키 정보를 입력시켜 주며 랜덤한 키 정보를 생성하는 블록이 그림 7의 키 스케줄러 블록이다.

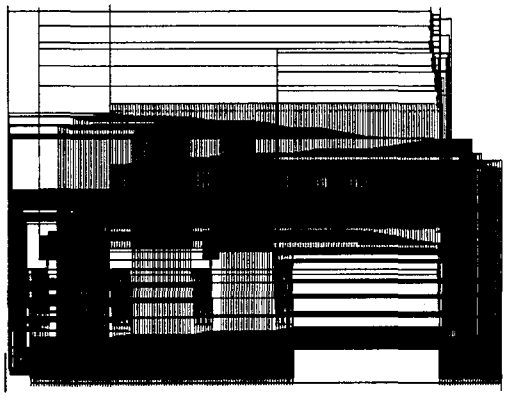

그림 7. 키 스케줄러 블록
Fig. 7 Key scheduler block

〈그림 8〉은 PRN-SEED 암호시스템의 전체 회로도이다. PRN-SEED 암호시스템은 SCS 블록과 PCS 블록 그리고 키 정보에 대한 상태변환을 수행하는 키 스케줄러 블록, 입력 데이터들에 대한 포맷 및 랜덤변수 발생을 수행하는 CSA & PRN 블록, 입력 정보들에 대한 실제적인 암호프로세싱인 상태변환을 수행하는 SOLO 블록, 128 비트를 32 비트별로 분리하여 처리되도록 하며 치환과정을 수행하는 전/후처리 블록, 각 블록에 대한 제어신호를 발생시키는 control 블록으로 분류된다.

PRN-SEED 암호시스템은 입력 128 비트에 대하여 출력 128 비트를 산출하며 암호문 또는 평문에 대하여 평문 또는 암호문을 출력하게 된다.

암/복호화를 수행하는데 있어 필요한 정보인 키 정보는 실제적으로 암/복호화를 수행할 때는 필요 없으며 단지 새로운 키 정보를 생성하기 위한 기본 자료로 활용될 뿐이다.

일정 라운드 과정이 마무리되면 출력단자 중 done_o 단자가 활성화되며 이때의 출력데이터가 암호화/복호화 된 데이터가 된다.

입력 데이터 128 비트, 키 데이터 128 비트가 입력되면 암호화 과정을 수행한 후 출력되어지는 출력 데이터는 암호화된 데이터 128 비트와 done 데이터 1 비트이다.

입력 데이터로 주어진 정보는 "1234567890ABCDEF1234567890ABCDEF"이며 키 정보는 "10101010101010101F1F1F1F1F1F1F1F"일 때 출력정보는 "1A3C2011F AED19635A24B88DE2E1FA39"이다.

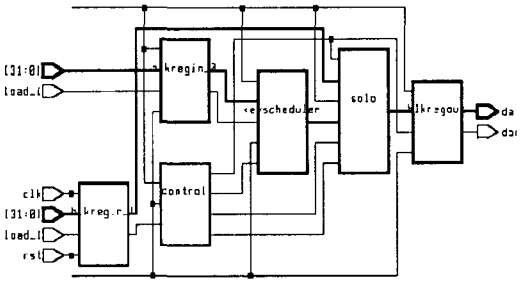

그림 8. PRN-SEED 암호시스템
Fig. 8 PRN-SEED cryptosystem

〈그림 9〉는 PRN-SEED 암호시스템의 전체 모의실험 결과 파형을 보여준다.

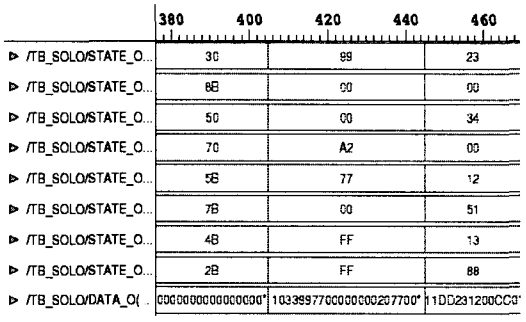

그림 9. PRN-SEED 암호시스템에 대한 모의실험 결과
Fig. 9 Simulation result for the PRN-SEED cryptosystem

〈표 2〉는 입출력 데이터들에 대한 PRN-SEED 암호시스템 관계표이다. 표에서 보는바와 같이 입력 데이터와 키 데이터 그리고 출력 데이터는 모두 128 비트로서 1:1 암호처리를 수행한다.

이러한 1:1 암호처리는 데이터의 증감을 가져오지 않으면서 DC에 대한 정보를 획득하기 어렵게 만든다. 그러므로 일반적인 암호시스템의 경우 1:1 처리가 되도록 설계된다.

표 2. PRN-SEED 암호시스템 입출력 관계
Table 2. In/output relation for PRN-SEED cryptosystem

입력 데이터	1234567890ABCDEF1234567890ABCDEF
키 데이터	10101010101010101F1F1F1F1F1F1F1F
출력 데이터	1A3C2011FAED19635A24B88DE2E1FA39

표 3은 기존 암호시스템과 제안된 PRN-SEED 암호시스템을 상호 비교 분석한 표이다.

표 3에서 기존 대칭형 블록 암호시스템에 비하여 PRN-SEED 암호시스템이 처리율면에서 1.03배의 특징을 가짐을 확인하였다. 또한 라운드 횟수와 비도에 의한 암호 효율 측면에서 PRN-SEED 암호시스템은 암호화에 사용되어지는 키 정보가 내부 PCS와 PRN에 의하여 생성되며 암호화와 복호화가 동일한 시스템에서 동시에 실행 가능함으로서 기존 블록 암호시스템에 비하여 2배의 효율을 가짐을 알 수 있다.

표 3. PRN-SEED 성능분석표
Table 3. PRN-SEED performance analysis

	시스템 구조	gate count(0.5㎛)	처리율 (40@MHz)
DES (Block cryptosystem)	1 라운드	1,090	416Mbps
	16 라운드	6,159	416Mbps
RSA (Asymmetric cryptosystem)	R-L architecture	186k	94kbps
	NTT 1994	105k	20kbps
	STI Tech. 2000	?	124kbps
PRN-SEED (Block cryptosystem)	10 라운드	10,158	430Mbps

그러나 시스템 면적이 기존 블록 암호시스템에 비하여 10배 증가됨을 알 수 있다. 그러나 요사이 SoC에 대한 구현론에서 굳이 면적상의 문제점은 고려되고 있지 않는 것이 현 실정이므로 전체적인 시스템 효율은 기존 블록 암호시스템에 비하여 PRN- SEED 암호시스템이 2배의 성능을 가짐을 알 수 있다. 이와 같이 PRN-SEED 암호시스템은 DES 또는 Rijndael과 같은 블록 암호시스템에 비하여 암호 프로세서 효율면에서 우수함을 알 수 있다.

V. 결 론

현대사회는 고정망, 이동망 및 근거리 통신망의 급격한 발전으로 인하여 네트워커 환경에서 상호 연계서비스를 강력히 요구하고 있는 추세이다. 이러한 연계서비스 만족을 위해서 보다 안전하며 실시간 처리가 가능하고 대용량의 데이터를 전송시켜야 한다는 전제조건이 필요하다. 이러한 전제조건을 만족하기 위하여 2000년도에 대칭형 암호알고리즘으로 Rijndael이 채택되었다. 그러나 AES의 후보 알고리즘들은 구현 또는 비도 측면에서 많은 단점을 가지고 있다. 특히 Rijndael 암호알고리즘의 경우 암/복호화가 동시에 수행할 수 없으며 다수의 사용자를 상대로 암호화 과정을 진행하는 것이 거의 불가능하다는 단점을 가지고 있다. 본 논문에서는 이러한 AES들에 대한 단점들을 없애고자 새로운 블록 암호알고리즘인 PRN-SEED 암호알고리즘을 개발하여 설계하였다.

제안된 PRN-SEED 암호시스템은 암호화를 수행하는 자원으로 자체 정보만을 가진다. 이러한 자체정보는 변환 블록에서 4가지 종류의 변환을 수행하게 되는데 이때 4가지의 변환은 순차적으로 수행되며 정보에 대한 event 발생을 라운드 변환으로 취급하기 때문에 각 라운드는 정보변환에 대한 기준가치로서 판단할 수 있다. 이러한 판단 기준은 비례적으로 비도와 직결된다. 처리시간 및 비도는 이러한 동시다발적인 연산으로 인하여 기존 대칭형 암호시스템에 비하여 처리율면에서 1.03배, 암호 효율면에서 2배의 처리율 향상을 가져왔다. 또한 암호화와 복호화를 하나의 시스템으로 처리 가능하므로 전체적인 시스템 효율면에서 2배의 성능을 가짐을 확인하였다.

PRN-SEED 암호알고리즘의 구현은 VHDL을 이용하여 탑다운 방식으로 진행하였으며 회로합성은 Synopsys Design Analyser Ver. 1999.10을 이용하였고 사용된 라이브러리는 삼성 KG75 library를 사용하였다. 모의실험에 사용된 툴은 Synopsys VHDL Debegger를 사용하였다.

모의실험 결과, 대칭형 암호시스템인 DES는 동작주파수가 40㎒일 경우 416Mbps의 처리율을 가지며, Rijndael 암호시스템은 동작주파수가 50㎒일 경우 612Mbps의 처리율을 가진다. PRN-SEED 암호시스템의 전체 게이트 수는 10K이며 동작주파수가 40㎒일 때 128비트에 대한 처리율은 430Mbps, 50㎒일 때 128 비트에 대한 처리율은 630Mbps였다. 그러므로 PRN-SEED은 DES 또는 Rijndael에 비하여 103%의 성능향상을 보임을 확인하였다.

참고문헌

[1] W. Stallings, "Cryptography and Network Security", Prentice Hall, 1998.

[2] E. Dawson, W. Millan and L. Simpson, "Methods for Designing Boolean Functions for Cryptographic Applications", In 58th Workshop on General Algebra Conference (AAA58) Vienna, June 1999.

[3] Neal Koblitz, "A Course in Number Theory and Cryptography", Springer-Verlag, 1994.

[4] E. Dawson and L. Nielsen, "Automated cryptanalysis of XOR Plaintext Strings", Cryptologia, Vol. XX, No. 2, pp. 165-181, April 1996.

[5] L. Simpson, J. Dj. Goli´c, M. Salmasizadeh and E. Dawson, "A Fast Correlation Attack on Multiplexer Generators", In Information Processing Letters, pp70, 1999.

[6] M. Kimberley, "Comparison of Two Statistical Tests for Keystream Sequences", Electronics Letters, Vol. 23, No. 8, pp. 365-366, April 1987.

[7] L. Brown and J. Seberry, "Key scheduling in DES type Cryptosystems", abstract of AUSCRYPT'90, 1990.

[8] E. Biham and A. Shamir, "Differential Cryptanalysis of DES-like Cryptosystems", Journal of CRYPTOLOGY Vol. 4 No. 1, 1991.

[9] E. Biham and A. Shamir, "Differential Cryptanalysis of the Full 16-Round DES", Proc. of CRYPTO'92, 1992.

[10] J. D. Golic, "On the linear complexity of Functions of Periodic GF(q) Sequences", IEEE Trans. on Information Theory, vol. 35. No. 1, pp. 69-75, Jan. 1989.

[11] M. Matsui, "Linear Cryptanalysis Method for DES Cipher", EUROCRYO'93, Extended Abstracts, 1993.
[12] L. Brown and J. Seberry, "On the Design of Permutation P in Des Type Cryptosystem", Abstract of AUSCRYPT'90, 1990.
[13] NIST, "Draft FIPS for the AES", http://csrc. nist.gov/publications/drafts.html, Feb. 2001.
[14] R. Rueppel, "Stream Ciphers", Contemporary Cryptology : The science of Infor. Integrity, New York, IEEE Pres, pp. 65-134, 1991.
[15] Burton S. Kaliski Jr., "An Overview of the PKCS Standards", RSA Data Security, Inc. 1993.

저 자 소 개

김 종 협

1992 원광대학교 공과대학 전자공학과 졸업(공학사)
1994 원광대학교 대학원 전자공학과 졸업(공학석사)
1998 원광대학교 대학원 전자공학과 (박사수료)
1994 ~ 현재 동아인재대학 인터넷정보처리과 교수

김 환 용

1973 전북대학교 공과대학 전기공학과 졸업(공학사)
1978 전북대학교 대학원 전기공학과 졸업(공학석사)
1984 전북대학교 대학원 전기공학과 졸업(공학박사)
1979 ~ 현재 원광대학교 공과대학 전자공학과 교수