Journal of the Korea Academia-Industrial cooperation Society
Vol. 12, No. 5 pp. 2311-2316, 2011

DOI: 10.5762/KAIS.2011.12.5.2311

바이트 가변 연산기능을 가진 블록 암호시스템 설계에 관한 연구

이선근[1*]
[1]전북대학교 화학공학부

A Study on the Block Cryptosystem Design with Variable Byte Operation

Seon-Keun Lee[1*]
[1]**Faculty of Materials & Chemical Engineering, Chonbuk National University**

요 약 정보통신과 네트워크의 발전으로 인하여 정보에 대한 보안의 중요성이 날이 갈수록 심화되고 있다. 이러한 시점에서 암호시스템들이 발전하고 있으나 비례적으로 크랙 및 해킹의 기술도 발전되고 있다.
그러므로 본 논문에서는 바이트 가변연산을 이용하여 블록암호시스템을 설계하였다. 설계된 바이트 기반 블록암호시스템은 고정된 DC 및 LC를 발생시키지 않으므로 이러한 공격방법으로부터 안전하다는 장점을 가진다. 또한 기존 대칭형 암호시스템이 가지고 있던 처리속도와 비대칭 암호시스템이 가지고 있던 인증기능을 모두 포함함으로서 네트워크 기반에서의 정보통신발달에 많은 도움을 주리라 사료된다.

Abstract With development of information communications and network environments security importance to the informations deepen as time goes. In this viewpoint, cryptosystem is developing but proportionally cracking and hacking technology is developing.
Therefore in this paper we proposed and designed block cryptosystem with byte variable operation. Designed cryptosystem based on byte operation is safe than existed cryptosystem because it is not generate the fixed DC and LC characteristics. Additionally, proposed cryptosystem have high processing rate and authenticated operation. Therefore proposed cryptosystem is considered to many aid in the network fields.

Key Words : DC, LC, Authentication, Asymmetric, Block cipher

1. 서론

네트워크 환경 및 멀티미디어의 기반 위에서 성장한 IT(Information Technology)산업으로 인하여 현대사회는 기술적으로 매우 윤택한 생활을 영위하게 되었다. 그러나 IT 산업의 실생활에 대한 응용 확대는 정보보호의 필요성이 전제조건이 된다. 이러한 정보보호는 IT 산업과 더불어 매우 많은 발전을 이룩하였지만 실시간 처리 및 비화성 유지등은 아직도 해결해야 하는 걸림돌이다[1].

2000년에 발표된 Rijndael은 DES 대체용 블록 암호시스템으로서 NIST(National Institute of Standard and Technology)에서 차세대 AES (Advanced Encryption Standard)로 결정하였다.

본 논문에서는 표준화된 AES인 Rijndael에 대하여 비도 증가 및 처리율 증가를 위하여 바이트 단위로 연산을 수행하는 새로운 암호알고리즘을 제안하였으며 Rijndael 및 다른 AES와 성능분석을 수행하였다[2-4].

2. AES Rijndael 암호 알고리즘

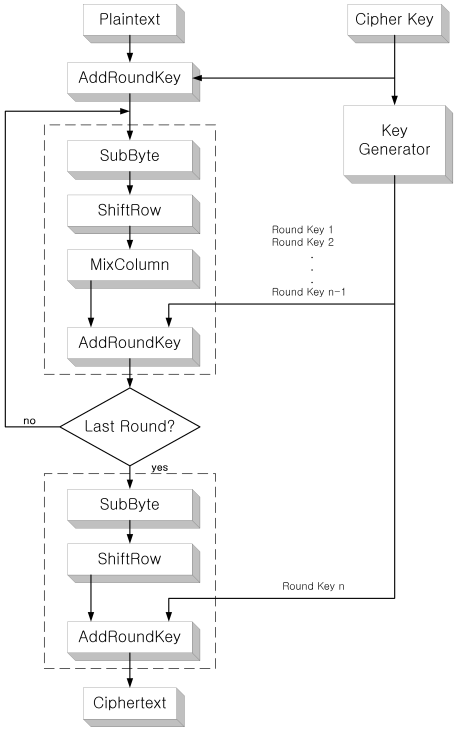
AES 암호알고리즘은 ATM, HDTV, 음성과 위성방송 등 여러분야에 적용이 가능하도록 하기 위하여 고안된 블록 암호알고리즘으로서 데이터 길이는 128 비트이며

*교신저자 : 이선근(caiserrisk@googlemail.com)
접수일 11년 03월 30일 수정일 11년 05월 11일 게재확정일 11년 05월 12일

키 길이는 가변적이 되도록 하였다. 또한 취약키(weak key)가 없어야 하며 3-DES보다 안전하며 효율적인 설계와 구현이 가능하고 알려진 공격방법에 대하여 강해야 한다. 이와 같은 전제조건을 만족하는 후보 알고리즘으로는 MARS, RC6, Rijndael, Serpent, Twofish 등이 있다. Rijndael은 Joan Daemen과 Vincent Rijnmen이 공동으로 만든 암호알고리즘으로서 선형 치환 변환이라는 연산을 사용한다. 여기에 사용되는 키 크기는 가변적이며 키 크기에 따라 라운드 횟수가 결정된다. Rijndael은 다른 암호알고리즘에 비하여 매우 효율적인 특성을 가진다.

AddRoundKey 연산을 포함한 모든 연산과정이 별도의 프로세서 없이 매우 높은 효율을 가지지만 암호화와 복호화를 동시에 수행할 수 없다는 단점을 가진다.

Rijndael은 바이트 치환(SubByte), 행 이동(ShiftDiagonal), 바이트 혼합(MixColumn), 라운드 키 덧셈(AddRoundKey)의 네 단계를 거치는 바이트 단위의 변환으로 구성된 라운드를 이용하여 암호화 및 복호화를 수행한다. 평문 입력은 State 배열에 저장된다. State 배열은 초기 라운드 키와 덧셈을 수행한 후 라운드 과정을 수행하게 된다. 모든 라운드가 실행되면 State 배열은 출력 배열에 저장되어 암호화를 마치게 된다. 이러한 Rijndael에 대한 전체적인 흐름은 그림 1과 같다.



[그림 1] Rijndael 암호알고리즘

SubByte, ShiftDiagonal, MixColumn, AddRoundKey의 단계를 거치면서 암호화를 수행하게 되는 Rijndael 암호화 과정은 마지막 라운드에서 MixColumn 단계를 생략한다. 암호화는 각 라운드를 수행하면서 State 배열을 처리하고 AddRoundKey 변환에서 사용되는 라운드 키는 키 확장 루틴을 이용하여 얻어진 4 바이트 워드를 이용한다. SubByte 변환은 S-box를 이용하여 State의 각 바이트에 독립적으로 작용하는 비선형적인 바이트를 생성하게 된다. SubByte 변환에 사용되는 S-box는 역변환(inverse transformation)이 가능하다. ShiftDiagonal 변환은 State 배열의 행단위로 변환이 이루어진다. 행번호가 0인 첫행은 shift 되지 않고 행번호가 1인 행은 한번 shift 되며 행번호가 2인 행은 두 번 shift 되며 행번호가 3인 행은 세 번 shift 된다.

이와 같은 변환은 같은 행에 있는 바이트들이 열의 번호가 낮은 위치로 이동하는 결과를 가져오며 열 번호가 낮은 위치의 바이트는 상위열의 위치로 이동하게 된다. MixColumn 변환은 고정된 다항식인 $a(x)$를 곱함으로써 변환되며 전체 변환식은 곱셈이 기본이 된다. 즉 변환된 함수 $s'(x)$는 변환전 함수인 $s(x)$에 대하여 $a(x)$를 곱한 형태를 가지게 된다.

AddRoundKey 변환은 라운드 키와 State 배열을 더하는 연산만을 수행한다. 각 라운드 키는 키 스케쥴로부터 Nb 개의 워드로 구성되며 Nb 워드들은 식 (1)과 같은 연산을 수행한다.

$$[s'_{0,c}, s'_{0,c}, s'_{0,c}, s'_{0,c}] = [s_{0,c}, s_{0,c}, s_{0,c}, s_{0,c}] \oplus [w_{\partial * Nb+c}] \quad (1)$$

여기에서 $0 \le c < Nb$이며 $[w_i]$는 키 스케쥴 워드이며 라운드 범위는 $0 \le round \le Nr$의 범위 안에 존재한다.

Rijndael 암호알고리즘에서 초기 라운드 키 덧셈은 라운드가 0일 경우이며 AddRoundKey 변환은 $1 \le round \le Nr$의 범위를 가지고 있을 경우이다.

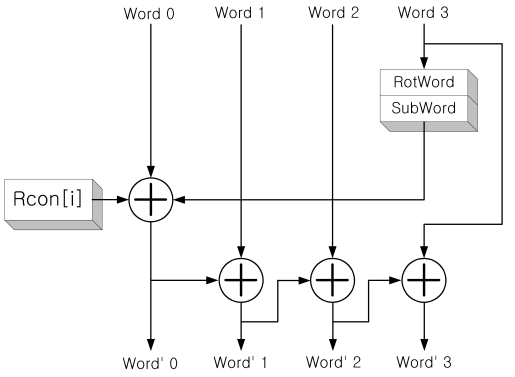
Rijndael 암호알고리즘은 처음 입력되는 암호키로부터 각 라운드에서 연산할 라운드 키를 생성하기 위하여 키 확장 루틴을 수행한다. 키 확장 루틴은 모든 Nb(Nr+1) 만큼의 4 바이트 워드들을 생성한다. 키 확장 루틴의 결과 4 바이트 워드들은 선형배열로 이루어지고 $[w_i]$로 표현된다. 이때 i는 $0 \le i \le Nb(Nr+1)$의 범위를 가진다. SubWord 변환, RotWord 변환, Rcon[i] 변환이 적용되는 키 확장 루틴은 각 라운드에서 사용하게 되는 라운드 키를 생성하게 된다. SubWord 변환은 4 바이트 입력워드로부터 출력워드를 생성하기 위하여 S-box를 이용하여 치

환기능을 수행한다. RotWord 변환은 입력워드에 대하여 순환적 치환을 수행하는 기능을 한다. 즉, 식 (2)와 같이 rotation 기능을 수행하게 된다.

$$RotWord\{word[a_0, a_1, a_2, a_3]\} = word[a_1, a_2, a_3, a_0] \quad (2)$$

라운드 상수에 대한 워드 배열인 Rcon[i]는 $[x^{i-1}, \{00\}, \{00\}, \{00\}]$로써 x는 $\{02\}$를 의미하며 x^{i-1}은 유한테 GF(2^8) 범위안의 값이 된다.

확장된 키의 첫 번째 Nk 워드들은 처음 입력되는 암호키로 채워지며 다음에 오는 $w[i]$는 이전 워드인 $w[i-1]$과 Nk 만큼의 전 워드인 $w[i-Nk]$와 배타적 논리합을 수행한다. 즉 입력 키는 $w[i-1]$과 배타적 논리합 연산을 수행한 후 RotWord 변환을 수행하고 4 바이트들에 대한 S-box 치환인 RotWord 변환수행 후 Rcon[i]와 배타적 논리합 연산을 수행한다.



[그림 2] 라운드 키 생성 스케쥴

그림 2는 키 확장 루틴을 수행하는 라운드 키 생성과정으로써 SubWord, RotWord, Rcon[i] 변환을 수행한 후 각 라운드에 맞는 키 데이터를 산출한다. 256 비트 암호키에 대한 키 확장 루틴은 128 비트 혹은 196 비트 암호키에 대한 루틴과 약간 다르다. 즉, $Nk = 8$이고 $i-4$가 Nk의 배수이면, SubWord 변환을 우선 적용하고 w $[i-1]$와 배타적 논리합 연산을 수행한다.

3. 제안된 바이트 단위 대칭형 암호시스템

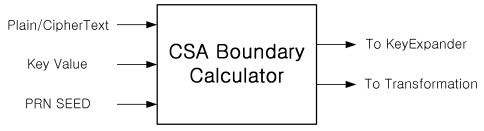
제안된 바이트 단위 대칭형 기반 암호알고리즘은 기본 연산자로 배타적 논리합을 사용하며 처리 수행단위는 바이트를 사용하였다. 바이트 연산은 처리속도를 매우 높게 수행할 수 있다는 장점과 더불어 역추적이 어렵기 때문에 비도 증가에도 매우 우수한 특성을 가진다. 또한 기존 Feistel 구조와 SPN을 사용하여 암호화를 수행할 때, 암호화와 복호화의 동시 수행이 가능하기 때문에 암호화를 수행하는 시스템에서 Rijndael 또는 Serpent 암호알고리즘과 같은 효율 저하가 발생하지 않는다. 이러한 특징은 AES에 대한 충분한 전제조건을 만족함과 동시에 AES 다음 버전에 대한 내용을 제시할 수 있다. 제안된 암호알고리즘에 사용되는 입력블록과 출력블록의 크기는 128 비트이며 키 크기도 128 비트로서 평문, 암호문 그리고 키의 크기는 1:1:1이 된다[5].

제안된 암호알고리즘은 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 수행기능 (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 초기 비선형 특성을 가진 상태를 의미한다. CSA는 식 (3) 및 그림 3과 같이 외부에서 주어지는 파라미터를 가지고 현재상태를 결정하며 결정된 현재상태는 불확실한 미래상태를 형성하는 기준값으로 설정된다.

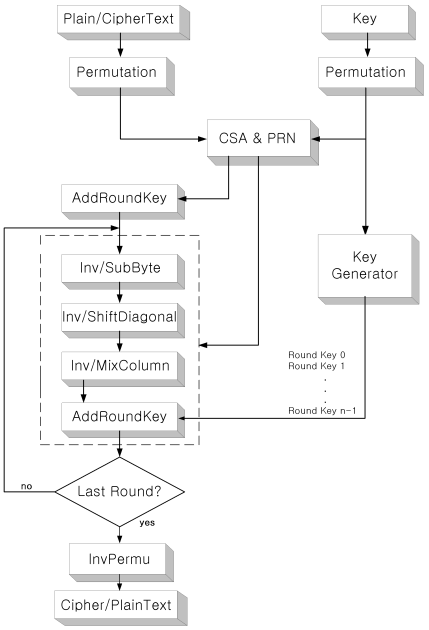
$$CSA_{next} <= CSA_{present}(prn_{seed} \bmod 8) \quad (3)$$



[그림 3] CSA 특성

제안된 암호알고리즘의 가장 큰 특징은 그림 4에서 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. Inv/SubByte, Inv/ShiftDiagonal, Inv/MixColumn, Inv/AddRoundKey는 라운드를 수행하는 기본단위가 된다.

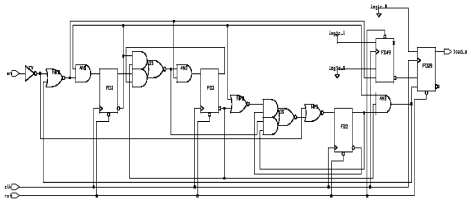


[그림 4] 제안된 바이트 단위 암호알고리즘

여기에서 네 가지 변환을 SOLO라고 정의하면 식 (4)와 같다.

$$SOLO_{n-round} <= \quad (4)$$
$$Inv/SubByte(odd)_n + Inv/ShiftDiagonal(odd)_n +$$
$$Inv/MixColumn(odd)_n + AddRoundKey(odd)_n$$

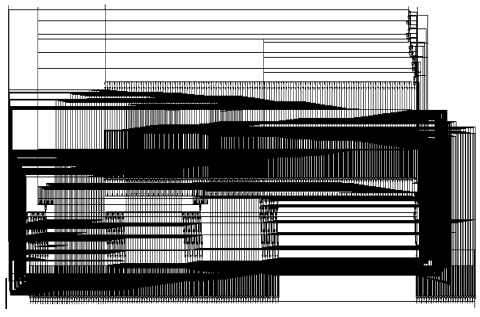
기존 AES들은 데이터와 키의 길이를 128, 192, 256 비트들로 가변시키며 변화하는 길이에 따라 최적화된 라운드 수를 결정하게 된다. 그러므로 기존 AES인 경우 데이터의 블록길이에 따라 라운드 수가 결정된다. 이러한 결과로 인하여 데이터의 심볼 크기를 파악하게 되는 경우 라운드 수를 파악할 수 있으며 라운드 수와 키 및 데이터와의 DC 및 LC에 의하여 크래킹이 가능해진다. 그러나 제안된 암호알고리즘의 경우 고정된 블록 및 키 크기를 가지고 있어도 내부적으로 라운드 수에 따라 데이터 내용이 변화되므로 라운드 수를 데이터 심볼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



[그림 5] 데이터 저장 제어 블록

그림 5는 S-box에 저장되는 정보를 받아들이는 동시에 S-box 값을 저장하고 있는 기능을 수행할 수 있도록 제어하는 기능 블록이다.

32 비트씩 입력데이터들이 레지스터 블록으로 유입됨과 동시에 S-box로 저장된다. 이때 각 저장 블록들에 대한 제어를 수행하는 블록인 데이터 저장 제어 블록은 S_box와 내부 메모리, 데이터 입/출입 인터페이스에 관련된 동기신호 발생 및 제어기능을 수행한다.



[그림 6] 키 스케쥴러 블록

그림 6은 입력되는 키 정보를 이용하여 PRN을 구동시키고 발생된 랜덤 키 값을 입력데이터와 상태변환을 수행하는 블록으로서 키에 대한 스케쥴을 수행하게 된다. 또한 S-box에 대한 저장 데이터를 관리하는 블록으로서 데이터 상태변환에 관련된 변환기능을 수행한다. 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 AddRoundKey 블

록의 입력으로 사용된다. 이때 제안된 암호시스템의 round는 10회로 결정되어 있다. 10회 연산이 이루어진 후 round 판별과정을 거친 후 마지막 치환과정을 거친후 복호문 또는 암호문을 생성하게 된다. 이와같이 SOLO 블록에 대한 키 정보를 입력시켜주며 랜덤한 키 정보를 생성하는 블록이 그림 6의 keyscheduler 블록이다.

표 1은 기존 암호시스템과 제안된 암호시스템을 상호 비교 분석한 표이다.

[표 1] 블록 암호시스템 성능분석표

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

표 1에서 기존 대칭형 블록 암호시스템에 비하여 제안된 암호시스템이 처리율면에서 1.03배의 특징을 가짐을 확인하였다. 또한 라운드 횟수와 비도에 의한 암호 효율 측면에서 제안된 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. 그러나 시스템 면적이 기존 블록 암호시스템에 비하여 10배 증가됨을 알 수 있다. 그러나 최근 SoC에 대한 구현론에서 굳이 면적상의 문제점은 고려되고 있지 않는 것이 현 실정이므로 전체적인 시스템 효율은 기존 블록 암호시스템에 비하여 제안된 암호시스템이 2배의 성능을 가짐을 알 수 있다. 이와 같이 제안된 암호시스템은 DES 또는 Rijndael과 같은 블록 암호시스템에 비하여 암호 프로세서 효율면에서 우수함을 알 수 있다.

5. 결 론

네트워크 환경 및 멀티미디어에 대한 연계서비스의 만족을 위해서는 다수의 사용자에 대하여 보다 안전하며 실시간 처리가 가능하고 대용량의 데이터를 전송시켜야 한다. 이러한 전제조건을 만족하기 위하여 2000년도에 대칭형 암호알고리즘으로 AES의 한 후보였던 Rijndael이 채택되었다. 그러나 AES의 후보 알고리즘들은 구현 또는 비도 측면에서 많은 단점을 가지고 있다. 그러므로 본 논문에서는 이러한 AES들에 대한 단점들을 없애고자 새로운 블록 암호알고리즘을 제안하여 설계하였다.

제안된 새로운 암호알고리즘은 평문 또는 암호문의 일부분과 ID를 이용하여 암호 키를 생성함으로서 네트워크 환경에서 실시간 처리 및 높은 비도를 설정할 수 있다. 또한 암호화를 수행하는 자원으로 자체 정보만을 가진다. 자체정보는 변환블록에서 4가지 종류의 변환을 수행하게 되는데 이때 4 가지의 변환은 동시다발적으로 수행되며 정보에 대한 event 발생시마다 라운드 변환으로 취급하기 때문에 각 라운드는 정보변환에 대한 기준가치로서 판단할 수 있다.

모의실험 결과, 대칭형 암호시스템인 DES는 동작주파수가 40㎒일 경우 416Mbps의 처리율을 가지며, Rijndael 암호시스템은 동작주파수가 50㎒일 경우 612Mbps의 처리율을 가진다. 제안된 암호시스템의 전체 게이트 수는 10K이며 동작주파수가 40㎒일 때 128 비트에 대한 처리율은 430Mbps, 50㎒일 때 128 비트에 대한 처리율은 630Mbps였다. 본 논문에서 제안한 암호알고리즘은 기존 암호알고리즘에 비하여 매우 높은 전송률 및 시스템 효율을 가지며 특정 길이의 키 결정을 수행할 필요가 없으며 구조적 기반 알고리즘이기 때문에 초고속 연계 무선망 발달에 따른 실시간 처리 및 대용량 데이터의 암호화에 매우 적합한 암호알고리즘으로 사료된다.

참고문헌

[1] M. Salmasizadeh, J. Dj. Goli´c, E. Dawson, L. Simpson, "A Systematic Procedure for Applying Fast Correlation Attacks to Combiners with Memory", In Workshop on Selected Areas in Cryptography(SAC 97), pp. 102-115, 1997.

[2] L. Brown and J. Seberry, "On the Design of Permutation P in Des Type Cryptosystem", Abstract of AUSCRYPT'90, 1990.

[3] C. Ding, V. Niemi, A. Renvall, and A. saloma, "Twoprime : A Fast Stream Ciphering Algorithm", Fast Software Encryption'97, Springer-Verlag, pp. 82-96, 1997.

[4] B. Schneier, "Applied Cryptography : Protocols, Algorithms, and Source Code in C", John Wiley & Sons, Inc., New York, USA, 1994.

[5] Third AES candidate conference, "AES3 Proceedings, http://csrc.nist.gov/encryption/aes/round2/conf3/papers/", pp. 44-54, April, 2000.

이 선 근(Seon-Keun Lee) **[정회원]**

- 1997년 8월 : 원광대학교 전자공학과 (공학석사)
- 2003년 2월 : 원광대학교 전자공학과 (공학박사)
- 2011년 4월 ~ 현재 : 전북대학교 화학공학부 겸임교수

<관심분야>
프로세서 설계, 암호알고리즘, 보안시스템설계