The Journal of The Institute of Internet, Broadcasting and Communication (IIBC)
Vol. 21, No. 1, pp.1-6, Feb. 28, 2021. pISSN 2289-0238, eISSN 2289-0246

https://doi.org/10.7236/JIIBC.2021.21.1.1
JIIBC 2021-1-1

IoT 보안을 위한 AES 기반의 암호화칩 설계

Design of AES-Based Encryption Chip for IoT Security

강민섭*

Min-Sup Kang*

요 약 본 논문은 하드웨어 자원이 제한되는 사물인터넷 시스템의 보안을 위하여 AES 기반의 효율적인 암호화칩 설계를 제안한다. ROM 기반의 S-Box는 메모리를 액세스하는데 많은 메모리 공간이 필요함과 동시에 지연문제가 발생하게 된다. 제안한 방법에서는 저면적/고성능의 암호화 칩 설계를 위해 합성체 기반의 고속 S-Box를 설계하여 보다 빠른 연산 결과를 얻도록 한다. 또한, 각 라운드 변환과정 및 키 스케쥴링 과정에서 사용되는 S-Box를 공유하도록 설계하여 보다 높은 처리율 및 적은 지연을 갖도록 한다. 설계된 AES 암호프로세서는 Verilog-HDL를 사용하여 회로동작을 기술하였으며, Xilinx ISE 14.7 툴을 이용하여 논리 합성을 수행하였다. 또한, 설계 검증은 Modelsim 10.3 툴을 이용하였으며, Xilinx XC6VLX75T FPGA 소자를 사용하여 하드웨어 동작을 검증하였다.

Abstract The paper proposes the design of AES-based encryption chip for IoT security. ROM based S-Box implementation occurs a number of memory space and some delay problems for its access. In this approach, S-Box is designed by pipeline structure on composite field GF((22)2) to get faster calculation results. In addition, in order to achieve both higher throughput and less delay, shared S-Box are used in each round transformation and the key scheduling process. The proposed AES crypto-processor is described in Veilog-HDL, and Xilinx ISE 14.7 tool is used for logic synthesis by using Xilinx XC6VLX75T FPGA. In order to perform the verification of the crypto-processor, the timing simulator(ModelSim 10.3) is also used.

Key Words : IoT security, AES chip, composite field, pipelined S-Box, Verilog HDL

Ⅰ. 서 론

최근, IT 기술과 인터넷 환경의 발달에 따라, 많은 정보통신 관련 기업들은 다양한 서비스를 제공하기 위해 IoT(Internet of Things) 기반의 임베디드 시스템 개발에 많은 비용을 투자하고 있다[1]. 이와 같이 IoT나 USN을 기반으로 하는 다양한 서비스들은 사용자들에게 편리성 및 유용성 등을 제공하지만, 관련 정보들을 이용, 수집하는 과정에서 다양한 형태의 해킹이나 정보를 보호하기 위한 보안기술이 필요하다[2]. IoT 시스템은 하드웨어나 소프트웨어 자원이 제한되어 있어 한정된 리소스를 요구하는 IoT 환경에 적합한 저비용, 저면적의 효율적인 화칩 설계가 요구된다.

암호 시스템은 크게 대칭(symmetric) 키 방식과 비대칭(asymmetric) 키 방식으로 분류된다. 대칭키 방식은 암호화와 복호화를 위해 동일한 키를 사용하며, 비대칭키

*정회원, 안양대학교 컴퓨터공학과
접수일자 2020년 12월 28일, 수정완료 2021년 1월 28일
게재확정일자 2021년 2월 5일

Received: 28 December, 2020 / Revised: 28 January, 2021 / Accepted: 5 February, 2021
*Corresponding Author: mskang@anyang.ac.kr
Dept. of Compter Engineering, Anyang University, Korea

방식은 암호화와 복호화를 위해 서로 다른 2개의 키를 사용한다.

대칭키 방식의 DES는 안전성 문제가 보고되어 미국 국립표준기술연구소(NIST)는 데이터 블록의 암호 및 복호 표준으로 RIJNDAEL을 AES로 선정하였다[3]. AES 알고리듬은 효율성, 보안성, 성능, 그리고 유연성 등 다양한 면에서 기존 암호화 알고리듬에 비해 성능 및 안정성이 탁월하다.

암호화와 복호화를 수행하는데 많은 데이터 연산 및 비교 처리를 실행하는데 빠른 계산속도가 요구된다. 이러한 암호 시스템은 소프트웨어나 하드웨어로 구현이 가능하지만, 소프트웨어의 경우 처리 속도에 있어 한계가 있다. 따라서 향상된 성능을 보장하고 보안성 및 안전성 향상을 위해 하드웨어 기반의 암호화 엔진이 요구되고 있다[4-5].

ROM 기반의 S-Box는 많은 저장 공간이 필요함과 동시에 지연문제를 해결해야 하는 문제점을 가진다. 이러한 문제점을 개선하기 위하여 합성체를 기반으로 하는 S-Box의 VLSI 구조가 제안되었다[5-6].

본 논문은 하드웨어 자원이 제한되는 사물인터넷 시스템의 보안을 위하여 AES 기반의 효율적인 암호화칩 설계를 제안한다. 제안한 방법에서는 저면적/고성능의 암호화 칩 설계를 위해 합성체 기반의 고속 S-Box를 설계하여 보다 빠른 연산 결과를 얻도록 한다. 또한 라운드변환 블록과 키 생성기에서 사용하는 S-Box 공유를 통해 지연시간을 감소시켜 암호화 프로세서의 성능을 향상시킨다.

II. 관련 연구

AES 암호 알고리듬은 128 비트의 데이터 블록을 암호화하며, 3 가지 킷값인 128비트, 192비트 그리고 256 비트를 사용한다. AES는 3 가지 다른 라운드 동작을 사용한다. 표 1은 AES에서 사용하는 라운드의 수에 따른 키값의 크기를 나타낸다[3].

AES 알고리듬의 시작 단계에서 128비트의 평문값과 첫 번째 라운드 키에 대해 덧셈 연산을 수행하게 되며, 라운드 변환은 4바이트×4바이트의 크기로 구성된다. 그 다음에는 4 종류의 변환 연산, 즉 SubBytes, ShiftRows, MixColumns 및 KeyAdd 연산이 순차적으로 진행된다. 라운드의 마지막 단계에서는 MixColumns 연산만 제외한 3종류의 변환을 수행하여 암호화된 결과를 얻는다. S-Box는 상기한 SubBytes 연산을 실행하는데 필요하며, S-Box는 8비트의 입력을 사용하며 8비트의 출력을 얻는다.

AES 알고리즘에서 S-Box는 곱셈역원(multiplicative inverse) 을 구하기 위한 모든 연산은 Galois Field 상에서 이루어진다. GF(2^8) 상에서 곱셈역원을 구하기 위해서 GF(2^8)을 계수가 GF(2^4)인 1차 다항식으로 변환하여 사용한다. GF(2^8)상의 임의의 다항식은 x^2+Ax+b 형태의 기약 다항식(irreducible polynomial)을 사용하여 $bx+c$의 형태로 표현할 수 있다[2].

III. 개선된 AES S-Box 설계

AES S-Box는 8비트를 입력으로 하여 8비트의 출력을 가지며, 키 길이가 128비트 인 경우 SubBytes 연산에 16개의 S-Box가 사용되고, 키의 생성에는 4개의 S-Box가 사용된다. S-Box는 암·복호화 과정에서 가장 큰 면적을 요구하므로 효율적인 S-Box 구현은 매우 중요하다.

GF((2^2)^2)상의 4비트 유한체 곱셈회로의 구현에서 k{k_3, k_2, k_1, k_0}, q{q_3, q_2, q_1, q_0}, w{w_3, w_2, w_1, w_0} 일 때 곱셈기의 출력 값 k=qw로 정의 된다. 입력 값 4비트 x에서 상위 2비트는 xH, 하위 2비트는 xL로 표기하면 식 (1)을 얻을 수 있다[2].

$$\begin{aligned} k &= (qHx+qL)(wHx+wL) \\ &= (qHwH)x^2+(qHwL+qLwH)x+qLwL \end{aligned} \quad (1)$$

여기에서는 $x^2=x+\varphi$를 사용하므로 정리하면 식 (2)과 같은 결과를 얻을 수 있다[3, 8].

$$k = (qHwH)(x+\varphi)+(qHwL+qLwH)x+qLwL \quad (2)$$

식 (2)를 $bx+c$의 형태로 표현하여 GF((2^2)^2) 의 곱셈기를 얻을 수 있다[6]. 이 곱셈기는 GF(2^2) 상에서 유한체 곱셈을 수행하기 위해서는 3개의 GF(2^2) 곱셈기와 파이(φ) 연산 모듈, 그리고 3개의 덧셈(XOR) 연산기 등으로 구성된다.

기존의 경우는 식 (2)를 이용한 $GF((2^2)^2$ 유한체 곱셈기를 구현하였으나[4], 본 논문에서는 조합회로로 구성된 합성체기반의 Bit-GF((2^2)^2) 곱셈기를 구현한다. 식 (3)은 Verilog HDL 언어로 기술된 로 최상위 출력 비트 Bit_GF(0)을 계산하기 위한 부울식을 나타낸다.

Bit_GF(0)=
(q(0)&w(0))⊕(q(1)&w(1))⊕(q(2)&w(3))⊕((w(3)&q(3)&⊕w(2))) (3)

식 (3)에서 연산자 &와 ⊕는 각각 비트 AND 연산과 덧셈 연산을 실행한다. 그림 1은 2-단(two-stages) 파이프라인 구조를 갖는 제안한 S-Box의 구조를 나타낸다.

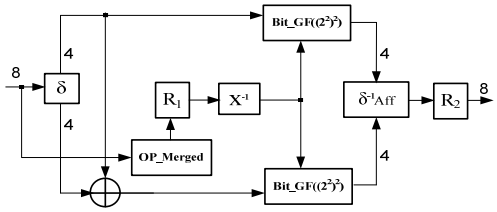

그림 1. 개선된 파이프라인구조의 S-Box 연산기
Fig. 1. Structure of proposed pipelined S-Box

그림 1에서 R1과 R2는 2단의 파이프라인 레지스터를 나타낸다. 그리고 X^{-1}모듈은 $GF((2^2)^2$ 상에서 역수를 계산하게 되는데, 여기에서는 효율성을 고려하여 LUT를 사용하여 구성한다. 제안한 S-Box는 기존의 방법[6]과 달리 X^2모듈, λ 모듈, $GF((2^2)^2$ 곱셈 모듈, 그리고 ⊕ 모듈을 한 개의 모듈로 통합하여 비트 단위로 연산을 실행하게 된다.

식 (4)에서 알 수 있듯이 제안한 유한체 곱셈기의 출력인 Bit_GF()은 입력된 비트값 단위로 & 연산과 덧셈 연산을 실행한다.

제안한 방법을 이용하면 기존의 방법과 달리 통한된 모듈에서 비트 단위로 연산을 실행하기 때문에 사용면적은 및 연산속도 등에서 성능이 개선된다.

IV. 개선된 AES 암호프로세서 설계

AES 에서는 3 종류의 암호화 키를 사용할 수 있지만, 128비트 길이의 암호 강도인 경우도 네트워크 환경에서 안전하다고 알려져 있기 때문에 본 논문에서는 128비트 키를 사용하여 통합된 암호프로세서를 설계한다.

그림 2는 제안하는 AES 암호프로세서의 main module인 Encrypt() 의 일부분을 나타낸다.

```
module Encrypt(input CLK, input [127:0] IN, input[127:0]
                KEY, output [127:0] OUT, output [127:0] K_OUT);
      //Round0 (Pre-round transformation)
AddRoundKey add(IN, KEY, round0);
      //Round1
KeyScheduling k1(CLK, 1, KEY, RK1);
SubBytes sub(CLK, round0, subs1);
ShiftRows shift(subs1, shifted1);
MixColumns mix(shifted1, mixed1);
AddRoundKey add1(mixed1,RK1 ,round1);
      ……
      ……
      //Round10
KeyScheduling k10(CLK,10, RK9, RK10);
SubBytes sub10(CLK, round9, subs10);
ShiftRows shift10(subs10, shifted10);
AddRoundKey add10(shifted10,RK10, round10);
assign OUT = round10;
assign K_OUT = RK10;
endmodule
```

그림 2. 제안하는 AES 암호프로세서의 Main module
Fig. 2. Main module of proposed AES processor

AES 암호화 모듈에서 암호화 과정은 먼저 Pre-round transformation(변환) 즉, AddRoundKey 연산이 실행된다. 다음 라운드부터는 SubBytes, ShiftRows, MixColumns 및 KeyScheduling 등 4종류의 연산이 순차적으로 수행되고, 라운드의 마지막 단계에서는 MixColumns 연산만 제외한 3종류의 변환을 수행하여 암호화된 결과를 얻게 된다.

그림 3은 AES 암호화 모듈과 통신 모듈을 통합하여 연결한 AES 암호시스템의 전체 블록도를 나타낸다. 그림 3에서 알 수 있듯이 제안하는 암호시스템은 크게 6 모듈로 구성되어 있다.

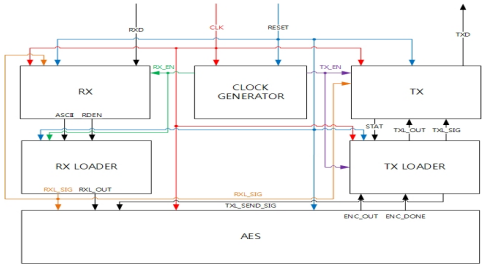

그림 3. AES 암호시스템의 전체 블록도
Fig. 3. Overall Block diagram of AES cryptosystem

여기서 AES 모듈은 그림 2에 나타난 알고리즘의 Main 모듈에 해당된다. 그림 3에서 설계한 각 모듈들은

통신과정에서 충돌을 방지하기 위해 상호간에 제어신호를 주고받는다. 메인 클록을 분할하여 송수신 듈 및 암호화 모듈에 맞게 클록을 동기시켜 최대 38,400의 통신 속도를 사용 할 수 있도록 설계 하였다.

CLOCK GENERATOR 모듈은 시스템 클록을 받아서 TX 클록, RX 클록을 생성하는 기능을 수행한다. RX 모듈은 RXD를 통해 송신 시스템으로부터 1Bit의 데이터를 수신 한다. RX 모듈측에서는 수신완료를 위해 RDEN제어신호를 RX_LOADER로 보낸다.

RX_LOADER는 RX로부터 받은 데이터를 저장하기 위한 일종의 버퍼이다. 이때 RX클록은 RX_EN에 의해 RX 모듈에 맞추어 동작을 하며 수신된 1Byte의 ASCII 코드와 완료 신호(RDEN)을 전달 받게 된다.

TX_LOADER는 AES 암호화 모듈로부터 암호화 된 128Bit 값을 받아 저장 한 후 이를 8Bit의 블록 단위로 나누어 TX로 전송하는 모듈이다. 시스템 클록과TX_EN이 인가될 때 AES에서의 암호화 된 값과 암호화 완료 신호인 ENC_DONE이 모듈의 입력으로 인가된다.

TX_LOADER에서 생성된 128Bit의 데이터는 8Bit단위로 분할하여 TXL_OUT 통해 TX로 전송하게 된다. 이때 1Byte를 송신할 때 마다 TXL_SIG가 인에블 되고, TX_LOADER의 모든 데이터에 대해 전송여부를 확인하기 위한 제어신호(TXL_SIG)를 출력신호로 사용한다.

TX 모듈은 외부의 대상 시스템으로 1Bit씩 값을 전송하는 모듈로서 외부로 데이터를 송신할 때 시스템 클록에 의해 제어된다.

V. 암호시스템 구현 및 성능평가

설계된 AES 암호시스템은 Verilog-HDL을 사용하여 모델링을 수행하였다. 또한, Xilinx ISE 14.7툴을 기반으로 하여 논리합성을 수행하였고, XC6VLX75T 소자를 이용하여 구현하였다. 그리고 ModelSim PE 10.3툴을 활요하여 타이밍 시뮬레이션은 수행하여 회로 동작을 검증하였다.

그림 4는 설계된 암호시스템에서의 암호화 및 복화과정을 검증하기 위한 시뮬레이션 결과를 나타낸다.그림 4에서 모든 모듈은 파이프라인으로 구성 되어 있으며, “round0”는 입력된 평문과 키를 가지고 XOR 연산을 수행하는 “AddRoundKey”을 나타낸다.

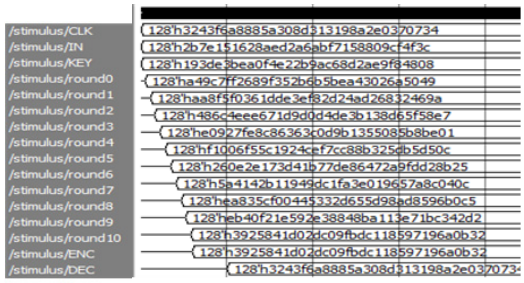

그림 4. 암/복호화를 위한 시뮬레이션 결과
Fig. 4. Simulation result for encryption/decryption

그리고 “ENC“ 라운드를 10회 수행하여 얻어진 암호화된 결과이다. 검증을 위한 복호화 프로세서는 별도로 구현 하였으며, “DEC“은 복호화를 수행한 결과이다. 그림 4의 결과를 통해 동일한 킷값을 통해 암호화 및 복호화 동작이 정상적으로 수행됨을 알 수 있다.

표 1은 그림 4의 시뮬레이션 결과를 정리한 것이다.

표 1. 암호화와 복호화 결과
Table 1. Result of encryption and decryptoion

stimulus/IN	3243F6A8885A308D313198A2E0370734
stimulus/KEY	2B7E151628AED2A6ABF7158809CF4F3C
stimulus/ENC	3925841D02DC09FBDC118597196A0B32
stimulus/DEC	3243F6A8885A308D313198A2E0370734

표 1에서 입력 값인 평문(stimulus/IN)과 키(stimulus/KEY)를 통해 암호화 되어 암호화된 결과(stimulus/ENC)를 출력한다. 이 암호화된 결과는 FIPS 197의 AES 표준[3] 에서 제시한 결과와 일치함을 확인하였다. 그리고 암호화된 결과를 복호화 하여 결과(stimulus/DEC)를 비교하면 입력된 평문과 일치함을 확인 할 수 있다.

표 2. AES 암호프로세서들의 성능 비교
Table 2. Performance comparison of AES crypto-processors

Items / Methods	Number of Slices	Frequency (MHz)	Throughput (Gbps)
McLoone[4]	1,589	89.76	1.035
Saurabh[5]	15,870 (gates)	100	1.280
Edwin[6]	15,544	139.31	1.783
Proposed	18,442	231.51	2.963

표 2는 기존의 방법들과 제안한 방법에 대한 AES 암호프로세서의 성능비교를 나타낸다.

표 2의 성능 비교결과에서 알 수 있듯이 제안한 방법(Proposed)은 Edwin[6] 방법에 비해 Slice는 약 16% 증가하였으나 동작속도(Frequency)는 약 40% 정도 개선되었다.

표 3은 본 논문에서 제안된 복호화 프로세서에 대한 합성결과를 나타낸다.

표 3. 제안된 복호화 프로세서의 합성결과
Table 3. Synthesis result for decryption processor

Items / Method	Number of Slices	Frequency (MHz)	Minimum period (ns)
Decryption processor	21,966	191.4	1.035

표 3의 합성 결과에서 알 수 있듯이 복호화를 위한 하드웨어 요구량은 암호화에 비해 다소 증가했으며, 처리속도 또한, 약 19% 정도 저하하고 있음을 확인할 수 있다.

표 4는 합성체 S-Box와 3 종류의 2단-파이프라인 방식에 대한 S-Box의 성능비교를 나타낸다. 논리합성을 위한 FPGA 소자는 Spartan Xc3s1500L을 사용하였다. 표 4에서 Composite S-Box는 합성체 기반으로 설계된 S-Box이다.

표 4. 2단-파이프라인 S-Box 기반의 성능 비교
Table 4. Performance comparison of three methods

Items / Methods	Number of Slices	4 input LUTs	Frequency (MHz)
Composite S-Box	42	74	37.282
2Stage S-Box[5]	47	86	93.791
2Stage S-Box[6]	43	80	102.965
Proposed	43	78	107.481

2-단 파이프라인 구조의 S-Box 면적 비교에서 제안한 방법은 기존의 방법[5] 보다 약 9% 감소하였고, 동작속도(최대 주파수)는 약 13% 정도의 속도가 개선되었다.

V. 결 론

IoT 장치는 개인 정보에 민감한 개인 정보뿐만 아니라 보안 및 안전에 방대한 양의 주요 데이터를 생성하고, 처리하는 역할을 하게 된다. IoT 시스템의 안전한 보안성을 갖추기 위해서는 데이터의 악의적인 변조나 가로채기 등의 무결성 및 기밀성을 보장되어야 한다 [5].

본 논문에서는 IoT 시스템에서의 기밀성을 보장하기 위해 개선된 S-Box를 기반으로 한 AES 암호화 프로세서의 설계하고, FPGA를 이용하여 하드웨어로 구현하였다. 본 논문에서 제안된 AES 암호화 칩은 Edwin[6] 방법에 비해 Slice는 약 16% 증가하였으나 동작속도(Frequency)는 약 40% 정도 개선되었다.

본 논문에서 제안한 S-Box를 이용한 AES 암호화 프로세서는 Method[3]에 비해 Slice는 증가한 반면, 주파수와 처리량이 개선되었다. 구현된 암호화 프로세서는 Verilog-HDL을 사용하여 동작을 기술하였고, Xilinx사의 ISE 14.7과 Modelsim 10.3을 이용하여 합성 및 시뮬레이션을 수행하였다.

본 논문에서는 소프트웨어 기반의 보안 알고리즘을 구현 시 문제가 되고 실시간 처리 능력을 개선하여 IoT 관련 시스템 구현에 적합한 하드웨어 기반의 암호화 칩을 설계하였다. 구현된 암호시스템은 자원이 제약되는 임베디드 시스템이나 사물인터넷과 같은 환경에서 데이터처리 속도는 물론 보다 높은 보안강도를 제공한다.

향후, IoT나 임베디드시스템 환경에서 요구되고 있는 소형 및 경량화, 저전력, 실시간 처리 등에 적합한 최적화된 암호화 솔루션 개발이 기대된다[9, 10].

References

[1] CISCO, "IoT", https://www.cisco.com/c/en/us/solutions/internet-of-things/overview.html#~stickynav=1(accessed Jan., 10, 2017)

[2] M. Katagi and S. Moriai, "Lightweight Cryptography for the Internet of Things", sony corporation, 2011.

[3] FIPS 197, "Advanced Encryption Standard (AES)", November 26, 2001.

[4] M. McLoone and J. V. McCanny, "Rijndael FPGA Implementation Utilizing Look-up Tables," IEEE Workshop on Signal Processing Systems, pp. 349-360, 2001.

[5] Saurabh Kumar, V.K. Sharma and K. K. Mahapatra, "Low Latency VLSI Architecture of S-Box for AES Encryption" Int'l Conf. on ICCPCT, 2013.

[6] Edwin NC Mui, "Practical Implementation of Rijndael S-Box Using Combinational Logic", Custom R&D Engineer Texco Enterprise Pvt.Ltd.

[7] Young-Gap You, Seung-Youl Kim, Yong-Dae Kim, Jin-Sub Park, "Low Power Cryptographic Design based on Circuit Size Reduction", Journal of the Korea Contents Association, Vol. 11, No. 2, pp. 92-99, 2007.

[8] Jong-Won Kim, Min-Sup Kang "Design of Advanced Multiplicative Inverse Operation Circuit for AES Encryption", The Journal of The Institute of Internet, Broadcasting and Communication, Vol. 20, No. 4, pp.1-6, Aug. 31, 2020.
DOI: 10.7236/JIIBC.2020.20.4.1

[9] Jung-Hwan Min, Young-Gon Kim, "End-to-end MQTT security protocol using elliptic curve cryptography algorithm", The Journal of The Institute of Internet, Broadcasting and Communication, Vol. 19, No. 5, 2019.
DOI: 10.7236/JIIBC.2019.19.5.1

[10] Sung-hwan Kim, Young-gon Kim, "A Study on Light Weight Authentication Method of DistributedCluster-based IoT Devices", Journal of Institute of The Internet, Broadcasting and Communication Vol. 19, No. 2, 2019.
DOI: 10.7236/JIIBC.2019.19.2.103

저 자 소 개

강 민 섭(정회원)

- 1979 : BS degree in Dept of Telecomm. Eng., Kwangwoon University.
- 1984 : MS degree in Dept of Electro. Eng., Hanyang University
- 1992 : Ph. D. Degree in Dept of Electro. Eng., Osaka University
- 1984 ~ 1992 : Senior researcher, ETRI
- 2001 ~ 2002 : Visiting scholar in Dept of electric. & comput. Eng., University of California, Irvine
- 1993 ~ present : Professor, Dept of Comput. Eng., Anyang University
- Research interests : ASIC design, Embedded system, Cryptosystem design, Network security, IOT security