경량 블록 표준 암호화 기술의 하드웨어 구현

임나리, 최소연, *유호영
충남대학교 전자공학과
e-mail : *nrim.cas@gmail.com, sychoi.cas@gmail.com, hyyoo@cnu.ac.kr*

Hardware Implementation of Lightweight Block Cipher Standards

Nari Im, Soyeon Choi, *Hoyoung Yoo
Chungnam National University

Abstract

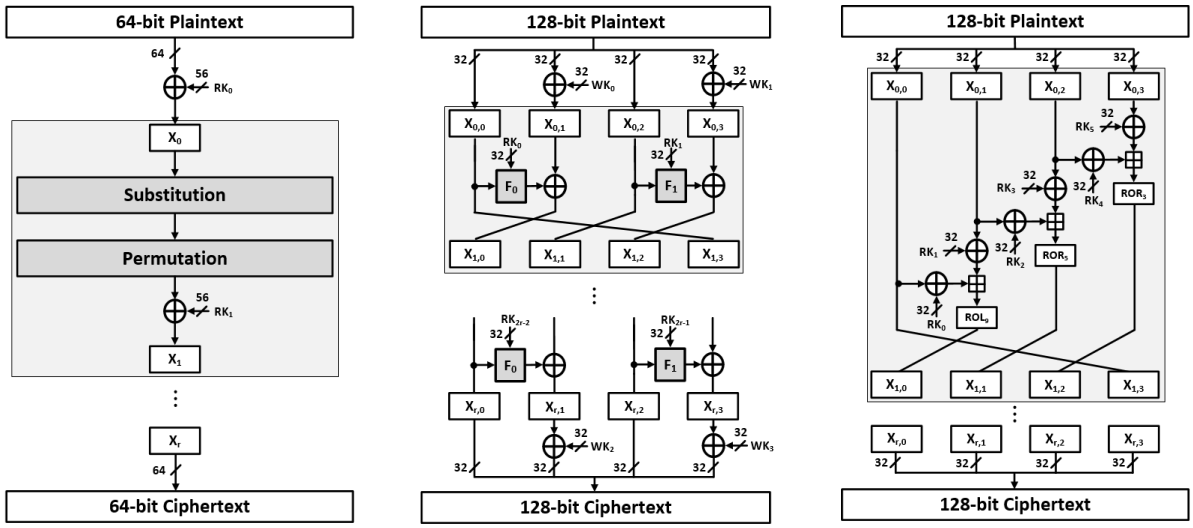
As wireless applications such as WSN(Wireless Sensor Network), RFID(Radio-Frequency Identification), and IoT(Internet of Things) are rapidly growing, the importance of information security and privacy has been increasing. To improve security and privacy given a highly constrained circumstance, lightweight cryptography techniques such as lightweight block ciphers have been proposed recently. In this paper, we analyze the algorithms of PRESENT, CLEFIA, and LEA, which are lightweight block cipher standards, and compare the hardware implementation results.

I. 서론

사물인터넷 (Internet of Things), 무선 센서 네트워크 (Wireless Sensor Network), RFID(Radio-Frequency Identification) 등의 기술발전으로 무선망으로 연결된 단말기 간의 데이터 공유가 증가하고 있다. 단말기 간의 데이터를 전송하는 과정에서 데이터의 유출 및 조작, 프라이버시 침해 등에 노출될 위험이 있으므로 정보 보안의 중요성이 대두되고 있다[1]. 전통적으로 정보 보안과 개인 정보 보호를 위하여 데이터에 AES[2]와 같은 전통적인 암호화 기술을 적용하고 있다. 그러나 이와 같은 암호화 기술은 IoT나 RFID를 사용하는 환경과 같이 공급되는 전력이 적고 공간적 제약이 큰 환경에는 부적합하다. 따라서 하드웨어 면적과 전력이 제한된 환경에서 정보 보안을 높일 수 있는 경량 암호화 기술(Lightweight Cryptography)이 제안되었고[3-5], 국제 표준화 기구인 ISO (International Organization for Standardization)와 IES (International Electrotechnical Commission)에서 경량 암호화 기술에 대한 표준을 ISO/IEC 29192로 규정하였다[6].

경량 암호화 기술은 작은 센서 및 디바이스에서 통신하기 위하여 전력 소모가 낮고, 제한된 메모리 공간에서 암호화가 가능한 암호화 기술이다. 경량 암호화 기술은 암호화/복호화를 진행하는 단위에 따라 블록 단위로 암호화/복호화를 진행하는 경량 블록 암호화 기술과 비트 단위로 암호화/복호화를 진행하는 경량 스트림 암호화 기술로 나뉜다.

일반적으로 블록 암호화 기술은 암호화와 복호화 과정에서 사용되는 키가 동일하다는 특징을 가지고 있다[2]. 블록 암호화 기술이 경량 블록 암호화 기술로 지정되기 위한 조건은 작은 블록 사이즈, 작은 키 사이즈, 단순한 암호화 연산, 단순한 키 생성 등이 있다[6]. 본 논문에서는 ISO/IEC 29192에서 표준으로 지정한 경량 블록 암호화 기술인 PRESENT[3], CLEFIA[4], LEA[5]를 하드웨어 측면에서 분석하고 구현하여 비교하였다.



(a) (b) (c)

그림 1. 암호화 알고리즘 (a) PRESENT-80 [3], (b) CLEFIA-128 [4], (c) LEA-128 [5]

II. 본론

2.1 PRESENT

PRESENT[3]는 2012년에 ISO/IEC에서 표준으로 지정한 최초의 경량 블록 암호화 기술로 입력된 평문/암호문을 64 비트 단위로 80 비트 또는 128비트의 키를 사용하여 암호화 한다. 이는 SPN (Substitution Permutation Network) 구조를 기반으로 31번의 라운드 함수를 이용하여 암호문/평문을 생성하며, 암호화 키가 80 비트인 경우의 암호화 과정은 그림 1의 (a)과 같다. PRESENT[3]의 r번째 라운드의 출력 값 X_r을 생성하기 위한 라운드 함수는 r번째 라운드 키 RK_r ($1 \le r \le 31$)를 더하는 과정, S-box를 사용하여 4비트 비선형 변환을 수행하는 S-box 대체 (substitution), 64 비트의 치환을 수행하는 치환 (permutation) 과정으로 구성된다. 라운드 함수의 S-box 대체 과정에 사용되는 S-box는 4 비트의 입력과 4 비트의 출력을 가지며, 이 과정에 필요한 16개의 S-box는 모두 동일한 값을 갖는다. 복호화 과정은 암호화 과정과 동일한 키를 사용하여 그림 1의 (a)의 역순으로 수행된다.

2.2 CLEFIA

CLEFIA[4]는 2012년에 ISO/IEC에서 표준으로 지정한 경량 블록 암호화 기술로 입력된 평문/암호문을 128 비트 단위로 128 비트, 192 비트 또는 256 비트의 키를 사용하여 암호화한다. CLEFIA[4]는 GFN (Generalized Feistel Network) 구조를 기반으로 키의 비트 수에 따라 18번, 22번 또는 26번의 라운드 함수를 이용하여 암호문/평문을 생성하며, 암호화 키가 128 비트인 경우의 암호화 과정은 그림 1의 (b)과 같다. r번째 라운드의 출력 값 $X_{r,i}$를 생성하기 위해 사용되는 키로는 화이트 키 WK_i, 라운드 키 RK_i가 있다. CLEFIA[4]는 4개 혹은 8개의 branch를 갖는 두 개의 GFN으로 구성되며, 사용되는 branch 수는 키 비트 수에 따라 다르다. 키의 비트 수가 128 비트인 경우, 4-branch GFN만을 이용하여 암호화한다. 4-branch GFN은 입력된 128 비트를 4개의 32 비트 블록으로 나누어서 라운드가 진행되고, 이 구조는 라운드 함수 뿐만 아니라 라운드 키를 생성하기 위한 중간 키를 생성에도 사용된다.

CLEFIA[4]의 r번째 라운드의 출력 값 $X_{r,i}$를 생성하기 위한 라운드 함수는 F-함수 F0, F1, F-함수의 결과를 더하는 과정 (addition), 4개 블록의 치환을 수행하는 치환 (permutation) 과정으로 구성되고, 마지막 라운드에서는 치환 과정 없이 수행된다. F-함수는 F0, F1으로 r번째 라운드 키 RK_r ($1 \le r \le 18$)를 더하는 과정, S-box를 사용하여 4 비트 비선형 변환을 수행하는 S-box 대체 (substitution), 매트릭스의 곱셈으로 구성된다. F-함수의 S-box 대체 과정에서 사용되는 S-box는 4 비트의 입력과 4 비트의 출력을 가지며, 이 과정에서 필요한 4개의 S-box는 서로 다른 2개의 값으로 구성된다. CLEFIA[4]는 라운드 함수를 수행하기 전, 후에 1번째 블록과 3번째 블록에 화이트 키 WK_i를 더하는 과정을 수행하여 최종적으로 암호문/평문을 생성한다. 복호화 과정은 암호화 과정과 동일한 키를 사용하여 그림 1의 (b)의 역순으로 수행된다.

Physical Metrics	AES [2]	PRESENT [3]	CLEFIA [4]	LEA [5]
Block bits	128	64	128	128
Key bits	128	80	128	128
Rounds	10	31	18	24
Clock Cycle	10	31	30	24
Equivalent Gate Count	43.5 K	2.4 K	22.1 K	8.6 K
Critical Path Delay	4.9 ns	4.5 ns	4.9 ns	4.9 ns

표 1. AES-128 및 PRESENT, CLEFIA, LEA의 하드웨어 구현 결과

2.3 LEA

LEA[5]는 2019년에 ISO/IEC에서 표준으로 지정한 경량 블록 암호화 기술로 입력된 평문/암호문을 128비트 단위로 128 비트, 192 비트 또는 256 비트의 키를 사용하여 암호화 한다. 이는 ARX (Addition, Rotation, eXclusive-or) 구조를 기반으로 키의 비트 수에 따라 24번, 28번 또는 32번의 라운드 함수를 이용하여 암호문/평문을 생성하며, 암호화 키가 128 비트인 경우의 암호화 과정은 그림 1의 (c)와 같다. LEA[5]의 r번째 라운드의 출력 값 $X_{r,l}$를 생성하기 위한 라운드 함수는 r번째 라운드 키 RK_r ($1 \le r \le 31$)를 더하는 과정, 비트의 순환을 수행하는 비트 순환 이동 (rotation), 32비트 덧셈을 수행하는 모듈로 덧셈 (addition)으로 구성된다. 복호화 과정은 암호화 과정과 동일한 키를 사용하여 그림 1의 (c)의 역순으로 수행된다.

Ⅲ. 구현

블록 사이즈가 128인 AES[2]와 경량 블록 암호화 기술 PRESENT[3], CLEFIA[4], LEA[5]를 하드웨어로 구현하여 65nm 공정을 사용하여 동작 주파수 200MHz로 합성한 결과를 비교하였다. 표 1은 구현한 결과로 하드웨어 면적 (Equivalent Gate Count)과 임계 경로 지연 시간 (Critical Path Delay)을 나타낸 것이다. 표 1에서 나타난 바와 같이 PRESENT[3]는 하드웨어 면적과 임계 경로 지연 시간이 가장 작았으며, LEA[5]는 암호문 생성에 필요한 클럭 사이클이 가장 작았다. 이때, CLEFIA[4]의 경우 중간 키를 생성하기 위한 과정이 라운드 함수 생성 과정과 동일하기 때문에 12번의 라운드 동안 중간 키를 생성한 후에 18번의 라운드 함수를 수행하도록 설계했다. PRESENT[3]는 블록의 크기가 세 가지 암호화 기술 가운데 64 비트로 가장 짧아 평문/암호문의 길이가 길 경우에 암호화/복호화 하는 시간이 길어진다는 단점이 있다.

Ⅳ. 결론

본 논문에서는 IoT 등 소형 장치에 적용 가능한 경량 블록 암호화 기술인 PRESENT[3], CLEFIA[4], LEA[5]를 구현하고, 그 결과를 하드웨어 측면에서 비교하였다. 하드웨어 면적이 중요한 경우에는 PRESENT[3]를 사용하는 것이 바람직하고, 동작 시간이 중요한 경우 필요한 클럭 사이클 수가 가장 짧은 LEA[5]를 사용하는 것이 바람직한 것을 확인하였다. 추후, 암호화 성능 분석을 위하여 경량 암호화 표준 기술에 대한 공격과 방어 대책을 연구할 계획이다.

참고문헌

[1] 정영훈, 송정환, " ISO/IEC JTC 1/SC 27 WG2 경량 암호기술 국제 표준화 동향, " 정보보호학회지, 25권 4호, pp. 11-17, 2015.

[2] Advanced Encryption Standard (AES), NIST FIPS Standard 197, 2001.

[3] Bogdanov, Andrey, et al. "PRESENT: An ultra-lightweight block cipher." International workshop on cryptographic hardware and embedded systems. Springer, Berlin, Heidelberg, 2007.

[4] Shirai, Taizo, et al. "The 128-bit blockcipher CLEFIA." International workshop on fast software encryption. Springer, Berlin, Heidelberg, 2007.

[5] D. Hong, J.-K. Lee, D.-C. Kim, D. Kwon, K. H. Ryu, and D. G. Lee, "LEA: A 128-bit block cipher for fast encryption on common processors," In Information Security Applications, WISA 2013, Jeju, pp. 3-27, 2013.

[6] MCKAY, Kerry, et al. "Report on lightweight cryptography," National Institute of Standards and Technology, 2016.