

# 초경량 블록 암호 CHAM에 대한 CPA 공격과 대응기법 제안

김현준\*, 권혁동\*, 김경호\* 서화정\*  
\*한성대학교 IT융합공학부  
e-mail:khj930704@gmail.com

## Suggestion of CPA Attack and Countermeasure for Super-light Block Cryptographic CHAM

Hyun-Jun Kim\*, Hyeok-Dong Kwon\*, Kyung-Ho Kim\*, Hwa-Jeong Seo\*  
\*Divison of IT convergence engineering, Hansung University

### 요 약

초 경량암호 CHAM은 자원이 제한된 장치 상에서 효율성이 뛰어난 덧셈, 회전연산, 그리고 XOR 연산으로 이루어진 알고리즘이다. CHAM은 특히 사물인터넷 플랫폼에서 높은 연산 성능을 보인다. 하지만 사물 인터넷 상에서 사용되는 경량 블록 암호화 알고리즘은 부채널 분석에 취약할 수 있다. 본 논문에서는 CHAM에 대한 1차 전력 분석 공격을 시도하여 부채널 공격에 대한 취약성을 증명한다. 이와 더불어 해당 공격을 안전하게 방어할 수 있도록 마스킹 기법을 적용하여 안전한 알고리즘을 제안한다.

### 1. 서론

최근 가전제품, 모바일 장비, 웨어러블 디바이스 등 다양한 임베디드 시스템이 사물 인터넷 기술로 연결되어 정보를 주고받고 있다. 이러한 기술은 안전한 보안 통신이 필요하다. 이를 위한 방법으로 메모리 공간이 작으며 비용이 낮은 사물 인터넷 플랫폼의 구현 및 배포가 쉬운 ARX (Add Rotation, Xor) 기반 암호 알고리즘이 제안되고 있다. 이러한 알고리즘으로는 LEA, HIGHT, SIMON, SPECK, 그리고 CHAM이 있다 [1, 2, 3, 4].

그러나 이러한 임베디드 디바이스에 암호 알고리즘을 구현하게 되면서 발생하는 결함들로 인해 부채널 분석 (side-channel analysis)에 취약할 수 있다. 부채널 분석이란 1999년 Kocher에 의해 처음으로 제안되었으며 구현 과정에서 설계자가 고려하지 못한 정보의 누출 정보를 통해 비밀 정보를 알아내는 공격 기법이다[5]. 이 중에 가장 효과적인 공격 방법은 전력 분석 공격 (Power Analysis Attack)방법이며, ARX 기반 알고리즘 LEA, HIGHT, SIMONE, 그리고 SPECK이 해당 공격 기법에 취약한 것으로 밝혀졌다[6, 7, 8]. 이러한 부채널 공격으로부터 안전성을 갖추기 위한 기법으로 마스킹 기법이 있다. 마스킹 기법이란 발생하는 중간 값을 랜덤하게 만들어 공격자에게 필요한 정보의 누출을 막는 대응기법이다.

본 논문에서는 8비트 프로세스 상에서 CHAM-64/128에 대한 전력 분석 공격 실험을 통하여 마스터 키 값을 획득할 수 있음을 확인한다. 또한 이를 보안하기 위한 마스킹 기법을 함께 제안한다.

본 논문의 구성은 다음과 같다. 2장에서 CHAM에 대

한 소개 그리고 부채널 공격 방법론에 대해 설명한다. 3장에서는 CHAM 64/128이 가지는 보안 취약점을 이용한 공격 방법을 최초로 제시한다. 4장에서는 제안된 부채널 공격 기법을 방어하기 위한 CHAM 블록 암호에 대한 마스킹 기법을 제안한다. 마지막으로 5장에서 본 논문에 대한 결론을 내린다.

### 2. 관련 연구

#### 2.1 국내 초경량 블록암호 CHAM

2017년 국가보안기술연구소 (NSR)에서 제안한 초경량 블록 암호 알고리즘 CHAM은 4-branch Feistel 구조의 블록암호로 3가지의 암호화 모드를 제공한다. 표 1은 CHAM 패밀리 목록과 매개 변수이다. 각 암호는 블록 크기가 n 비트이고 키 크기가 k 비트 일 때 CHAM-n / k로 표시된다. r과 w는 각각 라운드 수 및 워드의 비트 길이를 나타낸다.

cipher	n	k	r	w	k/w
CHAM-64/128	64	128	80	16	8
CHAM-128/128	128	128	80	32	4
CHAM-128/256	128	256	96	32	8

<표1> List of CHAM ciphers and their parameters

CHAM은 <표1> 구조의 상태 정보를 유지하지 않는 stateless 키 스케줄과 ARX 연산을 기반으로 하여 임베디드 디바이스에서 효율적이며 특히 저사양 디바이스 상에

서 더 효율적인 알고리즘이다. 또한 라운드키의 수는 라운드 수보다 훨씬 적으며 반복적으로 재사용하여 라운드 키를 저장하는 데 필요한 메모리 크기가 줄어 든다. 암호화는 8 비트 AVR 마이크로 컨트롤러의 작업 수를 최소화하기 위해 1 비트 및 8 비트 두 가지 유형의 왼쪽 회전을 사용한다.

2.2 상관 관계 전력분석(CPA)

전력 분석 공격은 전력 소비 모델과 측정된 전력 신호의 통계적 특성을 비교, 분석하여 암호화에 사용된 키를 찾아내는 강력한 부채널 공격 방법이다. 전력 분석 공격에서 차분 전력 분석 공격(Differential Power Analysis, DPA)과 상관 계수 전력분석(Correlation Power Analysis, CPA)은 가장 대표적인 분석법으로 알려져 있다.

CPA의 과정은 암호화 알고리즘에서 추측하고자하는 비밀키와 조작 가능한 정보로 이루어진 부분을 공격지점으로 설정하고 연산이 수행되는 부분의 소비전력을 수집한다. 그리고 추측하는 비밀키의 모든 경우에 대한 공격지점의 중간값을 해밍 웨이트(Hamming Weight) 모델로 변환한다. 위와 같은 방법으로 모델링된 전력 소비와 수집한 실제 전력 소비 간의 피어슨 상관 계수를 계산하여 가장 높은 상관계수를 나타내는 추측키를 키로 결정한다.

2.3 마스크 변환 기법

CHAM과 같은 ARX 구조의 알고리즘에 마스크 기법이 적용되는 경우 불 마스크 기법과 산술 마스크를 상호 변환 하는 과정이 필요하다.

Goubin이 CHES 2001에서 제안한 방법인 불 마스크와 산술 마스크의 변환(B2A, Boolean to Arithmetic Masking Conversion)기법과 산술 마스크와 부울 마스크의 변환(A2B, Arithmetic to Boolean Masking Conversion)기법은 임의의 비트 크기에 대해 변환이 가능하다. Xor 연산 5개와 2개의 산술 뺄셈 연산을 사용하여 부울 마스크를 산술 마스크로 변환 한다.

3. CHAM 64/128에 대한 CPA 부채널 공격 기법 제안

본 논문에서는 CHAM-64/128에 대해 상관전력 분석(CPA) 기법을 사용한다. 입력 값을 알고 있다고 가정하고 공격을 시도하였다. 16bit의 라운드키를 8비트씩 나누어 추측키로 사용하여 연산을 감소 시켰다.

CHAM은 하나의 마스터키로부터 2개의 키값이 생성된다. 따라서 CHAM의 모든 라운드키를 계산하지 않고도 k/w만큼의 라운드키를 알면 모든 키를 알 수 있다. 모든 경우의 마스터키 값에 대해 생성되는 라운드키를 전 탐색 기법을 통해서 사전 연산하여 공격하면 획득한 라운드 키 값을 통해 마스터키 값을 알아낼 수 있다. 그리고 CHAM은 이전 라운드에서 연산한 값을 사용하여 연산하므로 각 라운드에 해당하는 라운드 키 값을 알아낸 후 그 값을 다음 라운드에 사용하여 계산해야 한다.

CHAM에서 공격 가능한 중간 값은 X1의 평문값, 라운드키값, XOR 연산이 이루어지는 곳에 모듈러 덧셈 연산

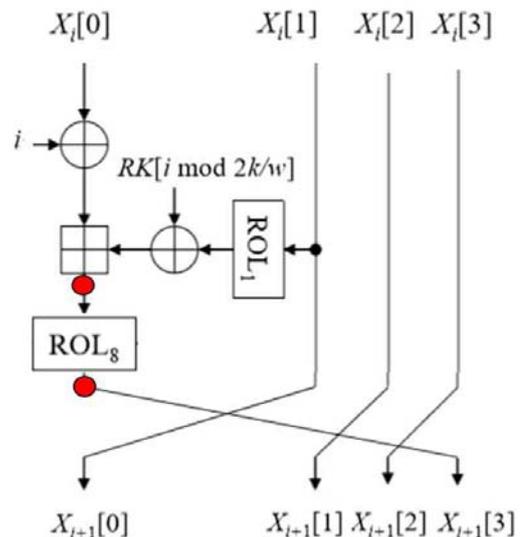
이 이루어진 값이다. 또한 왼쪽 회전 연산에서 16비트로 공격한다면 같은 해밍 웨이트 값을 나타내므로 차이가 없지만, 8비트씩 나누어 공격 하면 다른 해밍 웨이트 값을 나타내므로 중간값으로 공격이 가능하다.

CHAM은 홀수 라운드와 짝수 라운드에서 다른 연산을 반복하여 사용한다. 그렇기 때문에 홀수 라운드와 짝수 라운드를 서로 다른 방법으로 CPA공격을 수행하였다.

먼저 8비트 프로세서의 특징으로 인해 홀수 라운드의 왼쪽 회전 연산 지점을 중간값으로 하였다. 라운드키의 우측부분 8bit(RK8-15)만 추측값으로 공격을 시도하여 상관 계수가 가장 높은 추측값을 라운드 키의 좌측 8bit로 선택한다.

라운드키의 좌측 부분 또한 8bit(RK0-7) 만 추측값으로 넣고 결과값 중 가장 상관계수 값이 높은 추측값을 라운드키의 좌측 8bit로 선택하여 획득한다. 이렇게 획득한 16bit의 라운드키 값으로부터 마스터키 값을 획득 할 수 있다.

짝수 라운드의 공격은 우측부분 8bit(RK8-15)는 모듈러 덧셈 연산 결과 값을 중간 값으로 공격을 시도하고 좌측 부분에 8bit(RK0-7)는 ROL 연산 결과 값을 중간 값으로 하여 공격한다. 먼저 우측부분 8bit(RK8-15)만 추측 값으로 CPA공격을 시도하여 상관계수 값이 높은 추측값을 라운드키의 우측 8bit로 선택한다. 그리고 획득한 RK8-15 값과 RK0-7을 추측값으로 하여 CPA공격을 시도 하고 결과 값 중 상관계수 값이 가장 높은 추측 값을 라운드 키의 좌측 8bit으로 선택 한다.



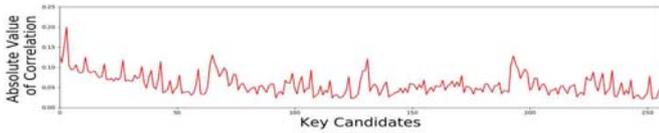
(그림 1) Attackable intermediate result of Odd Rounds

4. 공격 결과

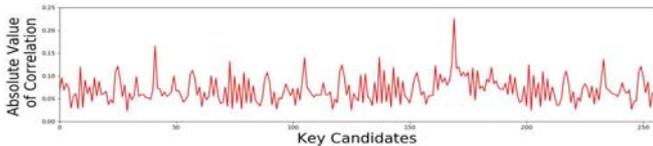
실험은 CHAM-64/128를 대상으로 8비트 프로세스 XMEGA보드에 Chipwhisperer를 사용하여 8라운드까지의 파형을 5,000개 수집하였다.

그림 2과 그림 3과 같이 1라운드연산에서 공격을 통해

나오는 가장 높은 상관관계를 가지는 좌측과 우측의 8bit 값은 (0x03, 0xA9)이다. (0x03A9)의 라운드키 값을 통해 사용된 마스터키 값의 일부인 (0x2B7E)을 알아낼 수 있다. 나머지 7개의 라운드에서도 동일한 방법으로 올바른 마스터키 값을 알아내어 공격에 성공하였다. 1~8라운드의 전력과형과 입력 값을 알고 있는 경우 5분 이내에 모든 비밀 키를 찾을 수 있었다.



(그림 2) Attack on the left side of the roundkey key in the first round. The highest correlation coefficient value is represented at (0x03).



(그림 3) Attack on the right side of the round key in the first round. It shows the highest correlation coefficient value at (0xA9).

#### 4.2 CHAM 1차 대응 기법 제안

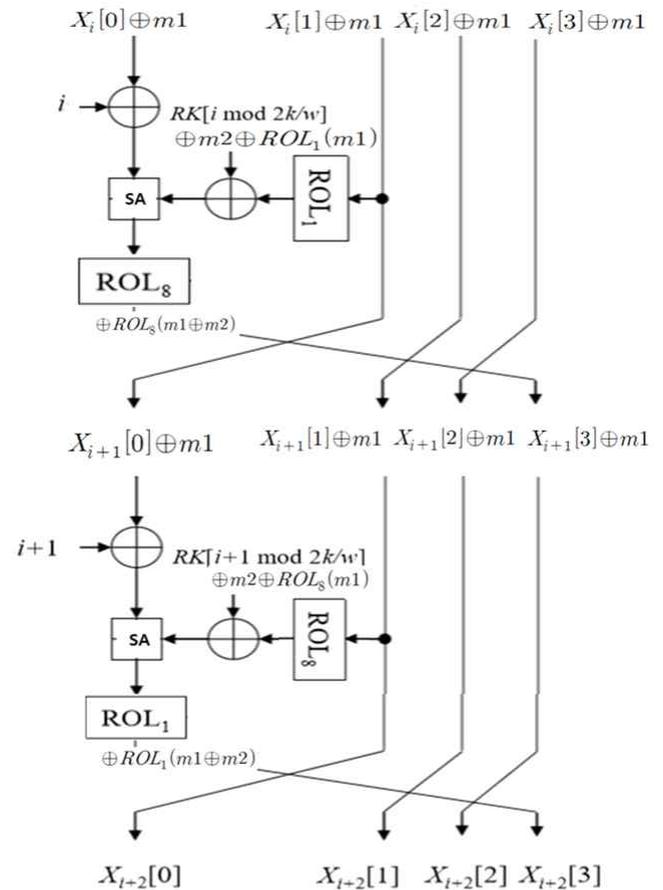
대표적인 부채널 대응기법으로 예상 가능한 중간 값을 랜덤하게 만드는 마스킹 기법이 쓰인다. 임베디드 디바이스에서 요구 되는 저용량 및 저전력 등으로 인해 고차 마스킹을 쓰는 것이 현실적이지 않기 때문에 1차 마스킹기법이 주로 쓰인다.

본 논문에서는 1차 부채널 공격에 안전한 CHAM 마스킹 기법을 제안한다. 일반 구현방법과 동일하게 라운드 함수의 입·출력 마스크 값을 동일하게 유지한다. 이때, 마스크 값은 16비트의 난수를 사용 하며, 효율성을 높이기 위해 모듈러 덧셈에 변환 기법 적용을 제외한 부분은 동일한 구조를 가진다. 그리고 B2A, A2B 연산을 동시에 수행 할 수 있는 기법 중 KRJ 기법을 사용하여 마스킹을 진행한다.

먼저 라운드키 생성 시, 마스크값  $m1$ 과 마스크값  $m2$ 를 1bit 의 왼쪽 회전 연산한 값을 짝수 라운드키 값에 XOR연산을 한다. 그리고 홀수 라운드의 라운드 키 값에는 마스크값  $m1$ 과 마스크 값  $m2$ 를 8bit 왼쪽 회전 연산한 값을 XOR 연산한다. 암호화 연산을 수행할 때 먼저 모든 평문값 에 마스크값  $m1$ 값으로 Xor 연산한다. 그 다음 연산인 입력값  $X_i[1]$ 은 라운드 키의 Xor 연산 시에  $m1$ 에서  $m2$ 값으로 마스킹 값이 변한 상태가 되며 Secure Addition 연산 이후 결과값은 ( $m1 \text{ xor } m2$ )가 마스킹 된 상태가 된다. 다음으로 왼쪽 회전 연 산 이후에 짝수 라운드 연산은( $m1 \text{ xor } m2$ ) 왼쪽 8bit 회전 연산을 수행하여 마스크값을 제거한다. 홀수 라운드의 연산은 ( $m1 \text{ xor } m2$ ) 왼쪽 8bit 회전 연산을 수행 하여 마스크 값을 제거한다.

이러한 방법은 라운드마다 반복된다.

제안 기법의 검증을 위해 III에서의 공격 과 동일한 환경과 공격을 시도였다. 그림 5와 그림 6과 같이 모든 추측 값에 대하여 낮고 특정 추측값에 대해 높은 상관관계를 나타 지 않았다. 그리고 그림 7과 그림 8과 같이 해당하는 실제값에 대한 상관관계가 드러나지 않아 안전한 마스킹 이 적용됨을 확인하였다.



(그림 4) Masking scheme suggested

#### 5. 결론

본 논문에서는 초경량 블록 암호 알고리즘 CHAM-64/128을 대상으로 저사양 8비트 AVR 프로세서 상에서 전력 분석 공격에 대한 취약성을 확인하였다. 또한 이러한 공격 대응책으로 CHAM에 효율적으로 적용할 수 있는 마스킹 구조를 제안하였다. 결과적으로 CHAM에 대한 전력 분석 공격에 대한 취약점 확인과 부채널 공격에 안전한 마스킹 기법을 제안하였다.

**참고문헌**

[1] D.S.Milojicic, V.Kalogeraki, R.Lukose,K.Nagaraja, J.Pruyne, B.Richard, S.Rollins and Z.Xu, Peer to Peer Computing, HP Laboratories Palo Alto HPL-2002-57, March, 2002.

[1] TTA, “128-bit lightweight block cipher LEA,” TTAK.KO-12.0223, Dec. 2013.

[2] D. Hong , 2006 , HIGHT : a new block cipher suitable for low-resource device, CHES 2006 , LNCS 4249 : 46 ~ 59

[3] Ray Beaulieu “The SIMON and SPECK Families of Lightweight Block Ciphers” 2013

[4] CHAM: A Family of Lightweight Block Ciphers for Resource-Constrained Devices

[5] P. Kocher, J. Jaffe, and B. Jun, “Differential power analysis,” Advances in Cryptology, CRYPTO’99, LNCS 1666, pp.388-397, 1999.

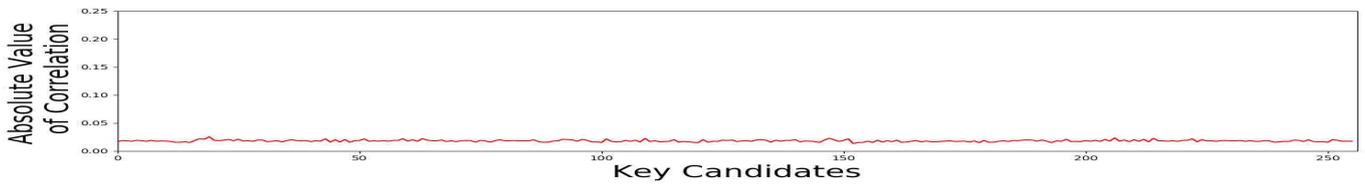
[6] J. Park, D. Hong, D. Kim, D. Kwon and H. Park,

“128-Bit Block Cipher LEA,” TTAK.KO-12.0223, Dec, 2013.

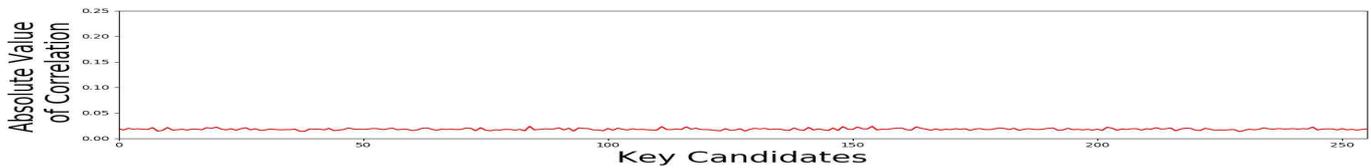
[7] Tae-jong Kim, Yoo-seung Won, Jin-hak Park, Hyun-jin An, Dong-guk Han. (2015). Side Channel Attacks on HIGHT and Its Countermeasures. Journal of the Korea Institute of Information Security & Cryptology, 25(2), 457-465.

[8] Biryukov A., Dinu D., Großschädl J. (2016) Correlation Power Analysis of Lightweight Block Ciphers: From Theory to Practice. In: Manulis M., Sadeghi AR., Schneider S. (eds) Applied Cryptography and Network Security. ACNS 2016. Lecture Notes in Computer Science, vol 9696. Springer, Cham

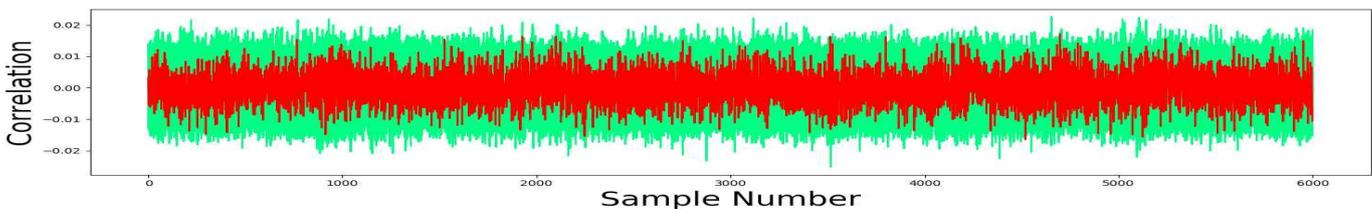
[11] Hwajeong Seo. 2018. Memory-Efficient Implementation of Ultra-Lightweight Block Cipher Algorithm CHAM on Low-End 8-Bit AVR Processors. Journal of the Korea Institute of Information Security & Cryptology 28: 545-550.



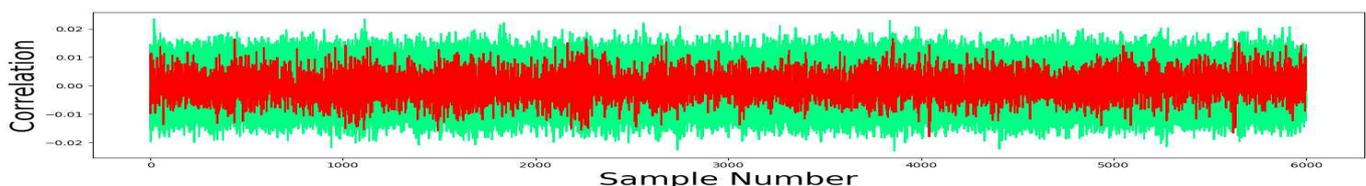
(그림 5) Attack on the left side of the roundkey key in the first round.



(그림 6) Attack on the left side of the roundkey key in the first round.



(그림 7) Attack on the left side of the roundkey key in the first round. Comparison of the actual value(red) with the estimated value(green).



(그림 8) Attack on the right side of the roundkey key in the first round. Comparison of the actual value(red) with the estimated value(green).