Journal of Sensor Science and Technology
Vol. 24, No. 2 (2015) pp. 124-130
http://dx.doi.org/10.5369/JSST.2015.24.2.124
pISSN 1225-5475/eISSN 2093-7563

WSN을 위한 128비트 확장된 데이터 블록을 갖는 고성능 HIGHT 설계

김승열[1] · 이제훈[2,+]

High Performance HIGHT Design with Extended 128-bit Data Block Length for WSN

Seong-Youl Kim[1] and Je-Hoon Lee[2,+]

Abstract

This paper presents a high performance HIGHT processor that can be applicable for CCM mode. In fact, HIGHT algorithm is a 64-bit block cipher. However, the proposed HIGHT extends the basic block length to 128-bit. The proposed HIGHT is operated as 128-bit block cipher and it can treat 128-bit block at once. Thus, it can be applicable for the various WSN applications that need fast and ultralight 128-bit block cipher, in particular, to be operated in CCM mode. In addition, the proposed HIGHT processor shares the common logics such as 128-bit key scheduler and control logics during encryption and decryption to reduce the area overhead caused by the extension of data block length. From the simulation results, the circuit area and power consumption of the proposed HIGHT are increases as 40% and 64% compared to the conventional 64-bit counterpart. However, the throughput of the proposed HIGHT can be up to two times as fast. Consequently, the proposed HIGHT is useful for USN and handheld devices based on battery as well as RFID tag the size of circuit is less than 5,000 gates.

Keywords: HIGHT, USN, WSN, RFID, Block cipher

1. 서 론

무선 통신 기술 및 무선 네트워크의 발전으로 무선 환경에서의 정보 보안 기술이 대두되고 있다. 무선 환경에서의 통신은 불특정 다수에게 정보가 방송되고 있기 때문에 이를 방지하기 위해 정보를 보호할 수 있는 암호화 기술을 필요로 한다. 최근, WSN (wireless sensor network)을 기반으로 이동형 장치의 소형화와 퍼베이시브 컴퓨팅, 무선 센싱 기술이 주목을 받고 있다. IEEE 802.15.4는 WSN에 적용할 수 있는 낮은 전송율과 저전력을 지원하는 표준이다. WSN 노드 및 장치는 대부분 보안을 신뢰할 수 없는 환경에 노출되어 있고 다양한 보안 공격에 민감하다. 따라서 IEEE 802.15.4는 보안을 위해 CTR (counter)모드와 CBC-MAC (cipher block chaining-message authentication code)모드가 더해진 CCM (CTR with CBC-MAC) 모드를 사용하고 있다[1-3].

IEEE 802.15.4에 사용되는 암호 회로는 CCM모드와 128비트 데이터 블록을 갖는 블록 암호 AES (advanced encryption standard)를 이용한 AES-CCM이다. CCM 모드에 적용하기 위한 블록 암호의 연구는 주로 AES를 저 전력 및 소형화를 위해 진행되었다[1,2]. AES 암호 회로의 소형화는 128비트 데이터 블록의 사이즈를 32비트 및 8비트 구조로 나누어 반복함으로써 회로의 크기를 소형화하였다. M. Feldhofer등은 3,400 게이트 크기로, 1.5 V동작 전압에서 4.5 uW@100kHz의 전력을 소비하는 8비트 데이터 버스 구조를 갖는AES 구조를 제안하였고 A. Satoh등은 5,400게이트를 갖고 32비트 데이터 버스 구조를 갖는 AES 구조를 제안하였다[4,5].

국내에서는 USN (ubiquitous sensor network) 및 RFID 시스템에 적용할 수 있는 초 경량 암호 알고리즘 HIGHT (high security and light weight)를 개발하였다. D. Hong 등은 USN 또는 RFID tag에 적용할 수 있는 3,048 게이트를 갖는 초경량 HIGHT 회로를 제안하였다[6]. 하지만 HIGHT암호 알고리즘은 64비트 데이터 블록 암호이기 때문에 IEEE 802.15.4 WSN에서 사용하는 CCM 모드를 사용할 수 없다. CCM모드는 128비트 블록 암호

[1] 충북대학교 정보통신공학부 (School of Informatioin and Communication Engineering, Chungbuk National University)
Chugbuk National University, 1 ChungDae-ro, Seowon-Gu, Cheongju, 362-763, Korea
[2] 강원대학교 전자정보통신공학부 (Div. of of Electronics, Information and Communication Engineering, Kangwon National Unversity)
Samcheock Campus, Kangwon National University, 1 Jooangang-ro, Samcheock, Gangwon, 245-711, Korea
[+]Corresponding author: jehoon.lee@kangwon.ac.kr
(Received : Mar. 23, 2015, Accepted : Apr. 2, 2015)

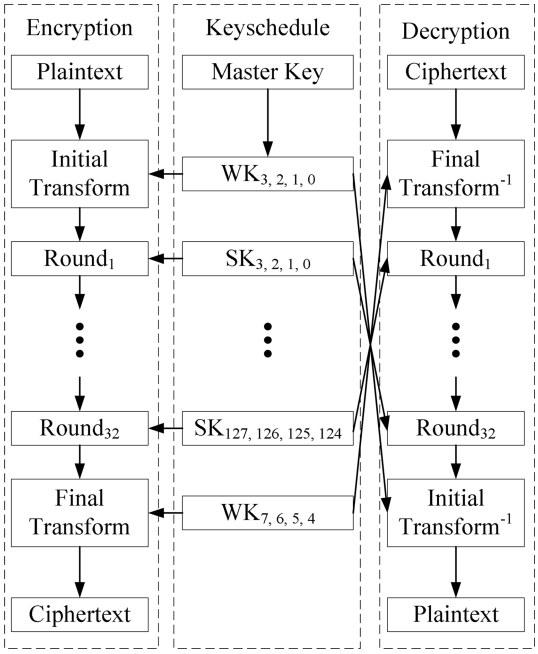

Fig. 1. HIGHT Block Diagram.

만을 지원한다[3].

본 논문에서는 HIGHT 알고리즘의 64비트 데이터 블록 크기를 128비트 데이터 블록으로 확장함으로써 CCM 모드의 사용을 가능하게 한다. 또한 키 스케줄과 제어 회로를 공유함으로써 회로의 크기를 절약하고 데이터 블록의 크기를 확장함으로써 처리량을 두 배 향상시켰다.

본 논문의 2장에서는 기존 HIGHT 블록 암호 알고리즘을 설명한다. 3장은 제안된 HIGHT 암호 알고리즘의 구조에 대해 설명하고, 4장에서는 성능 평가 및 분석을 한다. 마지막으로 5장에서 결론을 맺는다.

2. HIGHT 블록 암호 알고리즘

본 장에서는 기존의 HIGHT 블록 암호 알고리즘에 대해 소개한다. HIGHT 구현을 위한 전체 구조와 각각의 모듈의 동작 설명 및 구조를 설명한다.

2.1 HIGHT 전체 구조

HIGHT 암호 알고리즘은 변형된 Feistel 구조로 되어 있다. 이 알고리즘은 128비트의 키 길이와 64비트의 데이터 블록 길이를 갖고 있고 XOR(⊕), 모듈러 2^8덧셈(⊞) 및 좌측 순환 이동(⋘)과 같은 단순한 동작을 하기 때문에 저 전력, 저 비용, 초 경량 회로로 구현될 수 있다. HIGHT 블록 암호회로는 Fig. 1과 같이 암호화 블록, 복호화 블록 키 스케줄 블록으로 구성되어 있다. 암호화 블록은 초기변환, 라운드함수 그리고 최종변환으로 구성되고 복호화 블록은 역 최종변환, 라운드함수, 그리고 역 초기변환으로 구성된다. 키 스케줄 블록은 화이트닝 키(whitening key)와 서브 키(subkey)로 구성된다. 암호화 및 복호화 블록의 초기변환, 최종변환, 역 초기변환 그리고 역 최종변환은 각각 1회 수행되고 라운드함수는 32회 반복 수행된다. 키 스케줄 블록은 8개의 8비트 화이트닝 키(WK)와 128개의 8비트 서브 키(SK)를 생성한다[6].

2.2 키 스케줄

HIGHT 암호 알고리즘은 두 종류의 키를 사용한다. 그것은 각각 마스터 키 (MK)의 값을 직접 사용하는 화이트닝 키 와 키 생성 알고리즘을 통하여 생성되는 서브 키다[6].

화이트닝 키는 식(1)을 이용하여 생성한다. 생성된 키는 8개의 8비트, 즉 8바이트(WK_0, …, WK_7)로 구성되어 있으며 초기변환, 역 초기변환과 최종변환, 역 최종변환에서 사용된다.

$$WK_i = \begin{cases} MK_{i+12}, & 0 \le i \le 3 \\ MK_{i-4}, & 4 \le i \le 7 \end{cases} \tag{1}$$

서브 키는 마스터 키와 상수 값 δ_i를 필요로 한다. 상수 값 δ_i 는 LFSR (left feedback shift register) h를 이용하여 생성한다. LFSR h의 연결 다항식(x^7+x^3+1)을 이용한 δ_i의 생성식은 식(2)와 같다. 그리고 초기값 $\delta_0 = (s_6, s_5, s_4, s_3, s_2, s_1, s_0)$는 1011010_2으로 고정된다. h의 주기는 $2^7-1=127$ 이고 δ_0와 δ_{127}의 값은 같다.

$$\begin{aligned} &s_{i+6} = s_{i+2} \oplus s_{i-1}, \\ &\delta_i = (s_{i+6}, s_{i+5}, s_{i+4}, s_{i+3}, s_{i+2}, s_{i+1}, s) \\ &(1 \le i \le 127) \end{aligned} \tag{2}$$

서브키 생성 방법은 식(3)과 같다. 서브키는 128개의 상수 δ_i를 마스터 키와 모듈러 2^8 덧셈을 반복적으로 수행하여 생성된다. 생성된 키는 128개의 8비트, 즉 128 바이트(SK_0, …, SK_{127})로 구성되어 있으며 라운드함수에서 사용된다.

$$\begin{aligned} &\text{For i = 0 to 7} \\ &\quad \text{For j = 0 to 7} \\ &\qquad SK_{16 \cdot i+j} \leftarrow MK_{j-im\ od8} \boxplus d_{16 \cdot i+j}; \\ &\text{For j = 0 to 7} \\ &\quad SK_{16 \cdot i+j+8} \leftarrow MK_{(j-im\ od8)+8} \boxplus d_{16 \cdot i+j+8}; \\ &\delta_i (0 \le i \le 127) \end{aligned} \tag{3}$$

2.3 암호화 과정

HIGHT의 암호화 과정은 Fig. 1에서 나타낸 것과 같이 초기변환, 라운드함수, 최종변환으로 이루어져 있다[6]. 초기변환은 Fig. 2와 같이 스케줄러에서 생성된 화이트닝 키 WK_0,

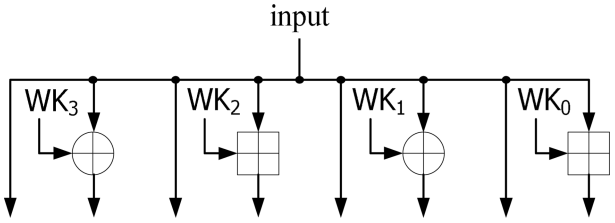

Fig. 2. Initial Transform.

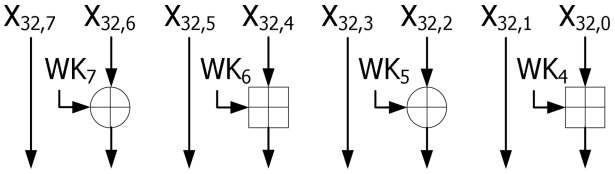

Fig. 3. Final Transform.

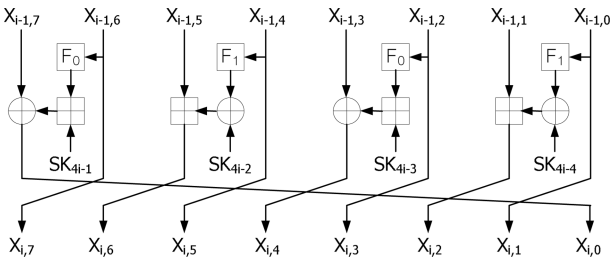

Fig. 4. Round Function, $Round_i$, i=1, …, 31.

WK_1, WK_2, WK_3 과 평문을 XOR연산과 모듈러 2^8덧셈을 수행한다. 또한 최종변환은 Fig. 3과 같이 화이트닝 키 WK_4, WK_5, WK_6, WK_7과 32번째 라운드 함수의 출력 X를 입력 받아 초기변환과 같이 XOR연산과 모듈러 2^8덧셈을 동일하게 수행한다.

라운드함수는 Fig. 4와 같이 XOR, 모듈러 2^8덧셈, F함수(F_0, F_1)로 구성되어 있다. F함수는 식(4)와 같다. 이 함수는 라운드함수의 입력 X에 대하여 좌측 순환 이동과 XOR연산을 수행한다.

$$F_0(X) = X^{\lll 1} \oplus X^{\lll 2} \oplus X^{\lll 7}$$
$$F_1(X) = X^{\lll 3} \oplus X^{\lll 4} \oplus X^{\lll 6} \quad (4)$$

2.4 복호화과정

HIGHT의 복호화 과정은 그림1과 같이 암호화 과정의 역순으로 이루어진다. 암호화 과정의 초기변환과 최종변환의 화이트닝 키가 반대로 입력되고 모듈러 2^8덧셈은 모듈러 2^8뺄셈으로 대체된다. 또한 라운드 함수에서 서브 키가 입력되는 모듈러 2^8덧셈을 모듈러 2^8뺄셈으로 대체한다. 라운드함수의 F함수는 복호화 과정과 동일하다. 서브키는 역순으로 라운드 함수에 입력된다.

Table 1. Gate count of conventional HIGHT

Component	HIGHT[6](0.25 um) (Gate Count)
RoundFunction	838
Key Schedule	1648
Control Logic	562
Total	3048

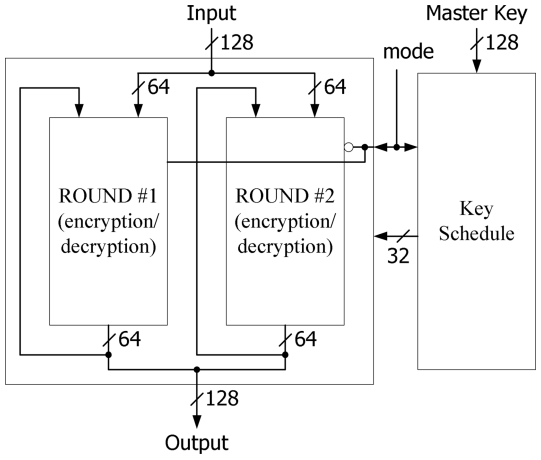

Fig. 5. Proposed HIGHT Block Diagram.

3. 제안된 HIGHT 암호회로

본 장에서는 제안된 HIGHT 암호회로의 구조 및 설계 방법을 소개한다. HIGHT 암호회로를 구성하는 주요 모듈의 상세한 구조와 동작에 대해 설명한다.

3.1 제안된 HIGHT 암호회로 구조

HIGHT 암호 알고리즘은 USN, RFID 등과 같은 장치에 사용되어야 하기 때문에 저전력 초 경량의 암호회로로 설계될 필요가 있다. 기존의 HIGHT는 하나의 라운드함수 블록과 하나의 키 스케줄을 사용한다. 설계된 회로의 모듈 별 크기는 Table 1에서 보는 것과 같이 키 스케줄 블록의 크기가 라운드 함수 블록의 크기보다 40% 이상 크다는 것을 알 수 있다.

제안된 HIGTHT 암호회로는 Fig. 5와 같이 기존 HIGHT 암호회로와 키 스케줄을 공유하며 두 개의 라운드 함수 블록을 사용한다. 두 개의 라운드 함수 블록을 갖는 제안된 회로는 라운드 함수 블록보다 30% 이상 큰 키 스케줄을 공유하기 때문에 하나의 라운드 함수 블록을 추가 하였을 때 기존 회로 구조와 비교하여 면적이 크게 증가 하지 않을 것이라는 것을 알 수 있다.

각각의 라운드 함수는 암호화 및 복호화를 모두 사용할 수 있는 구조로 되어 있다. Fig. 5에서 라운드#1과 라운드#2의 동작 모드는 4가지로 구성되어 있다. Table 2는 모드에 따른 라운드 블록의 암호화 또는 복호화 동작을 나타낸다.

Table 2. Operation mode of ROUND block

	Mode0	Mode1	Mode2	Mode3
Round#1	Encrypt	Encrypt	Decrypt	Decrypt
Round#2	Encrypt	Decrypt	Encrypt	Decrypt

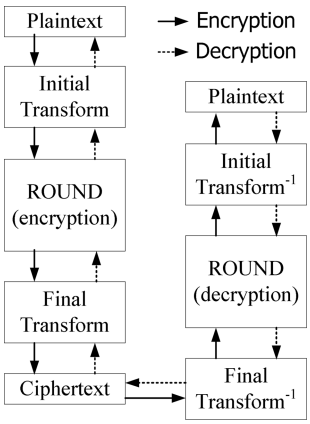

Fig. 6. Operational flow of encryption/decryption.

Fig. 6은 암호화 및 복호화 과정에서 데이터의 흐름을 보여준다. HIGHT 알고리즘은 암호화 블록 또는 복호화 블록 어느 것을 먼저 실행 시켜도 입력 값과 출력 값이 같다는 것을 알 수 있다. 따라서 Table 2의 모드1과 모드2를 사용하였을 때 암호화 및 복호화가 정상적으로 이루어 짐을 알 수 있다.

제안된 HIGHT 암호회로는 AES와 같이 128비트 데이터 블록을 갖는 암호 운용모드CCM을 지원할 수 있도록 구조를 제시하였다. 64비트 라운드 블록을 두 개 사용함으로써 128비트의 데이터 블록을 갖는 회로로 재구성하였다. 이 회로는 독립된 라운드 블록을 가지고 있으며 키 스케줄과 제어회로를 공유하고 있기 때문에 모드0 또는 모드3은 동일한 입력 값에 대하여 동일한 출력 값을 갖는다. 그러나 모드1과 모드2는 동일한 키 스케줄에 의한 동일한 키 값과 제어 신호를 사용하지만 라운드 블록의 동작이 암호화와 복호화로 른 출력 값을 갖는다. 따라서 모드0 또는 모드3을 사용하는 것보다 높은 보안성을 제시한다. 그러나 모드1과 모드2는 ECB 동작 모드에서 비 표준이며 모드0과 모드3은 표준이다. CCM 운용모드에서는 모두 비 표준이다. 하지만 제안된 HIGHT 알고리즘은 저전력 및 고성능 동작이 가능하여 AES와 비교하여 USN, WSN 및 RFID에서 효과적으로 사용할 수 있다.

3.2 키 스케줄 구조

키 스케줄 블록은 128비트의 마스터 키를 입력 받아 초기변환, 라운드 함수, 최종변환에 사용되는 화이트닝 키와 서브 키를 생성한다. 화이트닝 키는 초기변환, 최종변환에 사용되고 서브 키는 라운드 함수에 사용된다. 필요한 키를 매 클럭마다 생성하기 위하여 제안된 회로는 on-the-fly 방식을 사용한다.

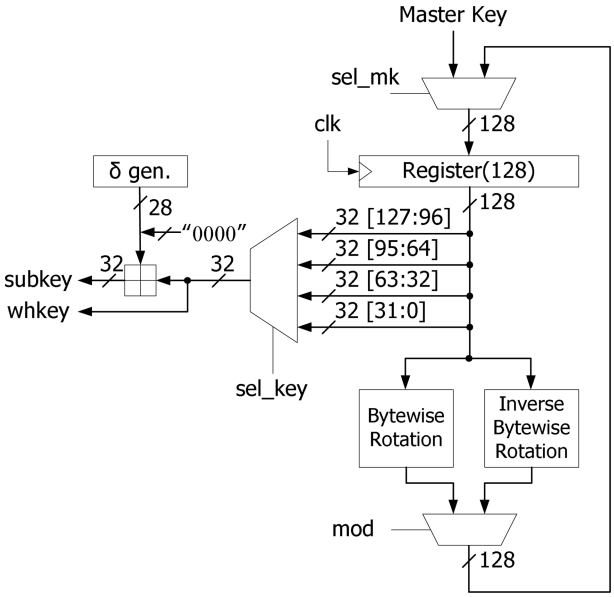

Fig. 7. Architecture of a key scheduler.

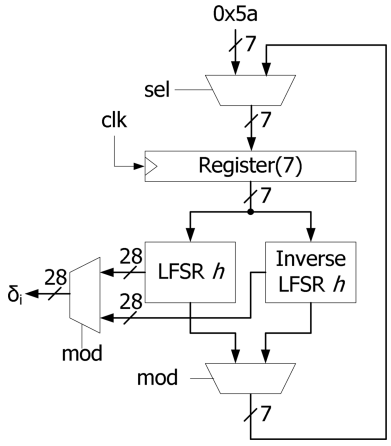

Fig. 8. Architecture of δ_i generator.

Fig. 7은 키 스케줄 블록의 구조를 나타낸다. 128비트의 마스터 키를 입력 받아 화이트닝 키를 마스터 키로부터 직접 생성한다. 서브키는 δ_i 생성기에서 생성된 상수 값과 마스터 키 값을 모듈러 2^8덧셈을 통하여 생성한다. δ_i 값은 28비트 출력을 하기 때문에 각각의 모듈러 2^8덧셈기의 최상위 비트인 8번째 비트에 '0'을 입력시킴으로써 32비트 덧셈을 수행하도록 한다. 서브 키에 사용되는 마스터 키 값은 Bytewise Rotation 블록에 의하여 바이트 단위로 순환이동이 이루어 지고 그 값이 레지스터에 저장된다. 바이트 단위의 순환 이동은 4 클럭마다 이루어진다. 라운드 함수의 반복이 모두 수행된 후 순환이동에 의해 위치가 변경된 레지스터의 값은 초기 입력된 마스터 키 값과 같아진다. 따라서 최종변환에 사용하는 화이트닝 키를 마스터 키 값에서 직접 생성할 수 있다. 순환이동은 와이어로 구성되기 때문에 게이트 카운트에 영향을 주지 않는다.

LFSR *h*를 사용한 δ_i 생성기는 서브 키를 생성하기 위해 매

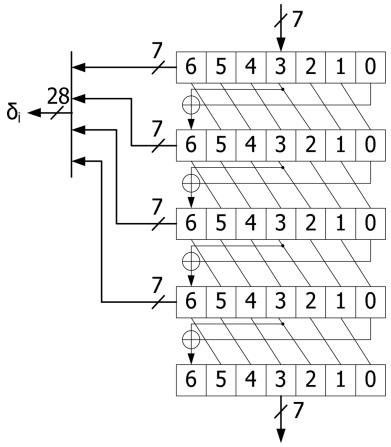

Fig. 9. Block diagram of LFSR *h* for encryption.

클럭마다 28비트의 상수 값을 생성한다. Fig. 8은 δ_i 생성기 구조를 나타낸다. δ_i 는 7비트의 레지스터와 LFSR *h*를 통하여 생성된다. δ_i 는 입력 받은 초기 값 1011010_2 (0x5a)이다. 암호화를 위한 LFSR *h*는 Fig. 9와 같이 비트 순환 이동과 XOR연산으로 이루어진다. 각각 7비트씩 총 28비트를 출력하도록 되어 있다. 복호화를 위한Inverse LFSR *h* 는 암호화의 역방향으로 이루어진다. 비트 순환 이동은 와이어로 구성되기 때문에 게이트 카운트에 영향을 주지 않는다.

3.3 제안된 라운드 블록 회로 구조

제안된 라운드 블록 회로는 초기변환, 라운드 함수, 최종변환이 모두 포함되어 있다. Fig. 10은 제안된 라운드 블록 회로의 구조를 나타낸다. 이 회로는 초기변환과 최종변환에 사용되는 XOR, 모듈러 2^8덧셈, 모듈러 2^8뺄셈을 라운드 함수의 회로와 공유한다. 라운드 함수의 회로를 공유하기 위해 데이터 입력을 와이어를 이용하여 초기변환을 할 수 있도록 교차 입력할 수 있도록 한다.

암호화 과정에서 초기변환에 입력되는 Input의 값은 다음과 같이 와이어를 통해 바이트 단위로 교차하여 입력한다.

$$\text{Input} \Rightarrow in_6 \| in_7 \| in_4 \| in_5 \| in_2 \| in_3 \| in_0 \| in_1$$

교차 입력된 Input 값은 라운드 함수의 XOR, 모듈러 2^8덧셈을 수행하고 멀티플렉서를 이용하여 다시 교차 시킨 후 출력한다. 초기변환이 끝난 Input 값은 라운드 함수를 반복 연산하고 최종변환 후 라운드 함수에서 사용한 좌측 순환 이동을 우측 순환 이동 후 출력한다.

복호화 연산은 암호화 연산과 역순으로 동작을 수행한다. 따라서 복호화 연산은 역 최종변환을 시작으로 라운드 함수, 역 초기변환으로 진행한다. 복호화 과정의 라운드 함수는 우측 순환 이동이 이루어 지고 역 초기변환 후 라운드 함수에서 사용한 우측 순환 이동을 좌측 순환 이동 후 출력한다.

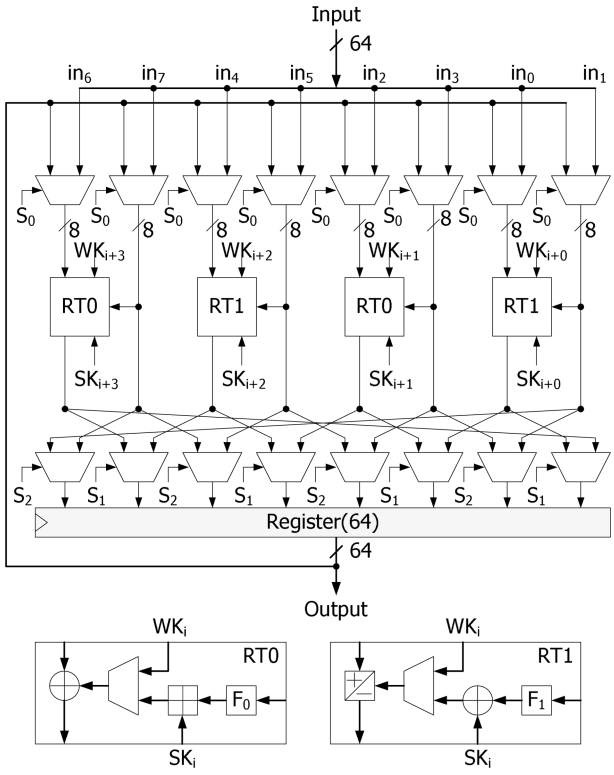

Fig. 10. Architecture of proposed round block circuit.

Table 3. Gate count of proposed HIGHT

		HIGHT (data_block: 64bit)	HIGHT (data_block: 128bit)
HIGHT Size (endecryption)		3,148(100%)	4,478(142%)
Size	Control	190(100%)	190(100%)
	Key Schedule	1,629(100%)	1,630(100%)
	Round	1,329(100%)	2,658(200%)
Technology		CMOS 0.25um	CMOS 0.25um

4. 성능분석

제안된 HIGHT암호회로는 CMOS 0.25um 공정을 사용하여 구현하였다. 합성과 전력측정은 각각 시납시스 사의 Design Compiler와 NanoSim을 이용하였다. Table 3은 제안된 HIGHT 암호회로의 크기를 나타낸다. 키 스케줄과 제어회로를 공유한 128비트 데이터 블록을 갖는 제안된 HIGHT의 크기가 64비트 데이터 블록을 갖는 HIGHT와 비교하여 크기가 42% 증가하는 것을 알 수 있다. 하지만 데이터 블록이 두 배 증가하였기 때문에 처리량은 두 배 증가하였다.

Table 4는 기존의 HIGHT와 AES 그리고 제안된 HIGHT의 성능을 비교 분석하였다. 64비트 HIGHT 암호회로는 기존의 HIGHT와 크기가 유사함을 알 수 있다. HIGHT[7]은 2,608 게이트 카운트로 가장 작은 회로 사이즈를 나타내고 있다. 이 회

Table 4. Comparison results of size and throughput

	Cycle	Size (GE)	Throughput (@80MHz)
This work(128 bit) (0.25 um)	34	4,478	301.2 Mbps
This work(64 bit) (0.25 um)	34	3148	150.6 Mbps
HIGHT[6] (0.25 um)	34	3048	150.6 Mbps
HIGHT[7] (0.35 um)	34	2,608	150.6 Mbps
AES-32[5] (0.11 um)	54	5400	189.6 Mbps
AES-32[1] (0.11 um)	54	7732	189.6
AES-8[1] (0.11 um)	160	4023	64.8
AES-8[4] (0.35 um)	1032	3400	9.9 Mbps

Table 5. Comparison results of Power and Energy

	Operation Voltage	Power (@100KHz)	Energy
This work(128 bit) (0.25 um)	2.5 V	17.5 uW	5.95 nJ
This work(64 bit) (0.25 um)	2.5 V	10.6 uW	3.6 nJ
HIGHT[6] (0.25 um)	-	-	-
HIGHT[7] (0.35 um)	2.5 V	10.8 uW	3.67 nJ
AES5-32[5] (0.11 um)	-	-	-
AES-32[1] (0.25 um)	1.8 V	37.3 uW	20.l nJ
AES-8[1] (0.25 um)	1.8 V	10.6 uW	16.9 nJ
AES-8[4] (0.35 um)	1.5 V	4.5 uW	46.44 nJ

로의 키 스케줄 블록 게이트 카운트는 1,591이다. 제어회로와 라운드함수 블록의 크기는 1,017 게이트 카운트를 갖는다. 따라서 제어회로를 제외한 라운드함수 블록의 게이트 카운트에 64비트 데이터 블록 레지스터가 포함되어 있지 않음을 알 수 있다. 레지스터가 포함되면 제안된 회로의 라운드함수 크기와 유사할 것이다. 제안된 64비트 HIGHT 및 기존의 HIGHT 암호회로는 AES-32와 비교하여 크기는 우세하지만 처리량이 부족함을 알 수 있다. 하지만 제안된 128비트 HIGHT는 크기 및 처리량 모두 우세함을 알 수 있다. AES-8은 64비트 HIGHT와 비교하여 크기는 유사하지만 처리량이 매우 낮음을 볼 수 있다. 하지만 Table 5에서 보듯이 소비전력이 매우 작음을 확인 할 수 있다.

제안된 128비트 HIGHT암호회로는 크기가 4,478 게이트로 5,000 게이트 이하로 구현할 수 있다. RFID tag에 사용하기 위해 암호회로의 크기는 5,000 게이트 이하로 만들어져야 한다. 따라서 제안된 회로는 RFID tag에 적용이 가능하다. 또한 이 회로는 표5에서 보듯이 소비 에너지 측면에서 64비트 HIGHT와 비교하여 전력은 64% 증가한 반면 에너지는 21% 절약 하였고 처리량은 두 배 증가하였음을 알 수 있다.

AES-8은 낮은 처리량을 갖는 저전력 RFID tag에서 HIGHT 암호회로보다 유리함을 알 수 있다. 하지만 USN, WSN 및 사물인터넷 등 배터리가 기반이 되는 핸드헬드형 기기에는 적합하지 않다[4]. 동일한 데이터를 처리하기 위해 AES-8은 16.9, 46.44 nJ을 사용하고 있고 HIGHT 암호회로는 5.95 nJ, 3.6 nJ을 사용하고 있다.

5. 결 론

본 논문은 USN, RFID와 같은 저전력, 초 경량 및 제한된 리소스를 요구하는 장치에 적용할 수 있는 고 성능의 128비트 데이터 블록을 갖는 HIGHT 암호회로를 제안 하였다. 제안된 암호 회로는 128비트 데이터 입력과 출력을 갖고 128비트의 키를 사용한다. 이 회로는 암호화 및 복호화 회로를 모두 내장하고 있고 34cycle에 동작을 완료한다.

제안된 회로는 기존의 HIGHT 알고리즘의 64비트 데이터 블록을 128비트 데이터 블록으로 확장함으로써 회로의 크기가 40% 증가하였다. 하지만 128비트로 데이터 블록이 확장되었기 때문에 처리량은 두 배 향상되었다. 그리고 전력 측면에서 64비트 블록 구조보다 전력이 64% 증가하였지만 에너지를 21% 절약 하였다. 또한 IEEE 802.15.4표준을 사용하는 WSN에서 사용하는 CCM 모드는 128비트 데이터 블록을 갖는 암호 회로만 사용이 가능하다. 따라서 데이터 블록을 128비트로 확장하였기 때문에 CCM 모드의 사용을 가능하게 하였다. 또한 소비 에너지를 줄임으로써 배터리 기반의 모바일 기기에 적합하게 배터리 소모를 줄일 수 있고 고속의 데이터 처리가 가능하도록 하였다.

Acknowledgement

본 연구는 교육부와 한국연구재단의 지역혁신창의 인력양성 사업으로 수행된 연구결과임(No. NRF-2012H1B8A2026055).

REFERENCES

[1] L. Huai, X. Zou, Z. Liu, and Y. Han, “An Energy-Efficient AES-CCM Implementation for IEEE802.15.4 Wireless Sensor Networks”, *2009 International Conf. on Networks*

Security, Wireless Communications and Trusted Computing, pp. 394-397, 2009

[2] F. Bin, Q. De-yu, and H. Han, "Parallel and multiplex architecture of AES-CCM coprocessor Implementation for IEEE802.15.4 Wireless Sensor Networks", *2013 Fourth International Conf. on Emerging Intelligent Data and Web Technologies*, pp. 149-153, 2013

[3] D. Whiting, R. Housley, and N. Ferguson, "Counter with CBCMAC(CCM)", RFC 3610, 2003

[4] M. Feldhofer, J. Wolkerstorfer, and V. Rijmen, "AES implementation on grain of sand", *IEE Proc. Information Security*, Vol. 152, No. 1, pp. 13-20, 2005.

[5] A. Satoh, S. Morioka, K. Takno, and S. Munetoh, "A compact rijndael hardware architecture with S-Box optimization", *Proc. ASIACRYTP2001 LNCS 2248*, pp. 239-254, 2001.

[6] D. Hong, J. Sung, S. Hong, J. Lim, S. Lee, B. S. Koo, C. Lee, D. Chang, J. Lee, K. Jeong, H. Kim, J. Kim, and S. Chee, "HIGHT: A new block cipher suitable for low-resource device", *Proc. CHES2006 LNCS 4249*, pp. 46-59, 2006.

[7] Y. I. Lim, J. H. Lee, Y. You, and K. R. Cho, "Implementation of HIGHT cryptic circuit for RFID tag", *IEICE Electronic Express*, Vol.6, No.4, pp. 180-186, 2009.