JANT 한국항행학회논문지
Journal of Advanced Navigation Technology

응용 및 융합 기술

J. Adv. Navig. Technol. 23(2): 207-213, Apr. 2019

사물인터넷 응용을 위한 암호화 프로세서의 설계

Design of Crypto-processor for Internet-of-Things Applications

안 재 욱 · 최 재 혁 · 하 지 웅 · 정 용 철 · 정 윤 호*
한국항공대학교 항공전자정보공학부

Jae-uk Ahn · Jae-Hyuk Choi · Ji-Ung Ha · Yongchul Jung · Yunho Jung*
School of Electronics and Information Engineering, Korea Aerospace University, Gyeonggi-do, 10540, Korea

[요 약]

최근 IoT 산업에서 보안의 중요성이 증가하고 있으며, IoT (internet of things) 통신 산업에서는 소형의 하드웨어 칩이 필요하다. 이를 위해 본 논문에서는 대표적인 블록 암호 알고리즘인 AES (advanced encryption standard), ARIA (academy, research, institute, agency)와 CLEFIA를 통합한 저면적 암호화 프로세서를 제안한다. 제안하는 암호화 프로세서는 128 비트 기반으로 라운드 키 생성 과정과 암호화 및 복호화 과정을 하나로 공유하였으며, 각각 알고리즘의 구조를 공유시켜 면적을 축소하였다. 더불어, 경량 IoT 기기를 포함한 대부분의 IoT 기기나 시스템에 적용이 가능하도록 구현하였다. 본 프로세서는 Verilog HDL (hardware description language)로 기술되었고 65nm CMOS 공정을 통해 논리 합성하여 11,080개의 논리 게이트로 구현 가능함을 확인하였다. 결과적으로 각 알고리즘 개별 구현 대비 gate 수 총계에서 약 42%의 이점을 보인다.

[Abstract]

Recently, the importance for internet of things (IoT) security has increased enormously and hardware-based compact chips are needed in IoT communication industries. In this paper, we propose low-complexity crypto-processor that unifies advanced encryption standard (AES), academy, research, institute, agency (ARIA), and CLEFIA protocols into one combined design. In the proposed crypto-processor, encryption and decryption processes are shared, and 128-bit round key generation process is combined. Moreover, the shared design has been minimized to be adapted in generic IoT devices and systems including lightweight IoT devices. The proposed crypto-processor was implemented in Verilog hardware description language (HDL) and synthesized to gate level circuit in 65nm CMOS process, which results in 11,080 gate counts. This demonstrates roughly 42% better than the aggregates of three algorithm implementations in the aspect of gate counts.

Key word : Advanced encryption standard (AES), Academy, research, institute, agency (ARIA), CLEFIA, IoT, Low area.

https://doi.org/10.12673/jant.2019.23.2.207

Received 6 March 2019; **Revised** 5 April 2019
Accepted (Publication) 22 April 2019 (30 April 2019)

***Corresponding Author; Yunho Jung**

Tel: +82-2-300-0133
E-mail: yjung@kau.ac.kr

Ⅰ. 서 론

최근 4차 산업혁명이 대두되고 있으며 특히, 사물 간 연결망을 기반으로 방대한 정보를 수집 및 활용하는 IoT (internet of things) 시장의 규모는 점차적으로 증가하고 있는 추세이다 [1]. 이에 따라, IoT를 겨냥한 보안 위협은 기하급수적으로 증가해 IoT의 암호화의 중요성은 큰 이슈가 되고 있다 . 또한, 소형화되고 있는 IoT 기기들을 고려하여 볼 때 현재 시장에 존재하는 IoT들의 응용에 적합하도록 하기 위해선 소형의 암호화 처리기가 필요하다 [2].

이러한 암호화 처리기를 구현함에 있어서 전 세계적으로 가장 많이 사용되는 AES (advanced encryption standard)를 가장 먼저 생각 할 수 있다 [3]. 또한, 국내에서는 정부 차원에서 장려하고 있는 ARIA (academy, research, institute, agency)가 가장 보편적으로 사용되고 있으며, ISO (international organization for standardization)에서 경량 암호화 표준으로 등록된 알고리즘인 CLEFIA는 다양한 분야의 IoT 기기에 적용할 수 있을 것으로 판단된다 [4], [5]. 위 세 알고리즘을 모두 이용한다면 전반적으로 AES 알고리즘을 통해 많은 IoT 기기에서 사용 가능하며, ARIA 알고리즘을 통해 국내 기관과 이와 관련된 부처에서 사용되는 기기에 대한 보호를 할 수 있다. 또한, 나머지 기기에 대한 보호는 국제 표준기구에 등록된 CLEFIA 알고리즘을 통해 가능하므로 위 3개의 알고리즘을 모두 구현한다면, 대부분의 기기에 대해 보안 시스템을 적용 할 수 있다.

위의 세 가지 알고리즘은 각각의 연산에서 SPN (substitution -permutation network) 구조를 사용하고 있다 [6]. 이때, SPN 구조는 암호화 및 복호화 과정에 있어 원함수와 이에 대응하는 역함수 모듈이 필요하다. 즉, 기존의 단일 암호화 프로세서의 경우 하나의 알고리즘을 위해 암호화 모듈과 복호화 모듈이 동시에 필요하다. 따라서, 3개의 알고리즘을 모두 구현한다면 각각의 알고리즘에 대한 암호화와 복호화 모듈이 필요해 하드웨어 복잡도가 증가하게 된다. 이에 따라, 위의 세 알고리즘이 모두 S-box라는 표를 이용한다는 공통점을 이용해 이를 공통된 연산 모듈로 구현해 각 알고리즘 별로 다른 암호화 처리기를 구현하는 경우보다 더 소형으로 구현이 가능하다. 즉, 본 논문에서는 소프트웨어 암호화 구현 방식과 달리 보완 및 수정에 있어서 상대적으로 유연성이 부족한 하드웨어 구현 방식에서의 유연성을 강화하기 위해 암호화 알고리즘인 AES, ARIA와 CLEFIA를 모두 지원해 선택적으로 동작 시킬 수 있는 저복잡도 암호화 프로세서를 제안하고 설계 및 검증 결과를 제시한다. 본 논문의 구성은 다음과 같다. 우선 Ⅱ장에서는 사용하게 될 AES, ARIA와 CLEFIA 알고리즘에 대해 설명하고, Ⅲ장에서는 제안하는 암호화 프로세서의 하드웨어 구조 설계결과를 제시한다. Ⅳ 장에서는 제안된 하드웨어 구조에 대한 구현 결과를 제시하며, 끝으로 Ⅴ 장에서 본 논문의 결론을 맺는다.

Ⅱ. 암호화 알고리즘

2-1 AES 알고리즘

AES는 2001년 미국 NIST (National Institute of Standards and Technology)에서 DES (data encryption standard)를 대체할 알고리즘으로 개발되었으며, ISO/IEC 18033-3 국제 표준 암호 알고리즘으로 등록이 되어 미국 외에 캐나다, 브라질, 일본, 유럽 등 많은 나라에서 권장 암호 알고리즘으로 이용하고 있다 [7]. SPN (substitution-permutation network) 구조를 가지는 알고리즘으로 암호화를 위한 key의 경우 128, 192와 256 비트를 지원하며, 입력으로 들어가는 block의 크기는 128 비트로 정해져 있다. Key size에 따라 전체 round 수는 10, 12와 14로 달라지며, 각 round 마다 내부 함수로 지정되어 있는 동작을 수행하게 된다. 내부 함수로는 SubBytes, ShiftRows, MixColumns와 AddRoundKey가 있다. SubByte의 경우 기존의 데이터를 s-box를 이용하여 대치하며, ShiftRows는 4x4 크기의 block의 각 행에 대해 rotation을 수행한다. MixColumns의 경우 finite field 내에서만 이용되는 곱 연산을 수행하며, AddRoundKey는 key expansion을 통해 만들어진 round key와 데이터의 XOR 연산을 수행한다. 이러한 내부 함수를 이용한 전체적인 흐름은 그림 1과 같다.

첫 round에서는 최초로 사용하게 되는 입력 master key를 이용한 XOR 연산의 결과를 이용하여 SubBytes, ShiftRows와 MixColumns 연산이 수행된 후, Key expansion을 통해 해당되는 round key와의 XOR 연산이 이루어지며, 이는 128 비트 key 기준으로 9번이 진행된다. 마지막 round에서는 ShiftRows 연산 후의 결과를 마지막 round key와의 XOR 연산을 통해 최종 암호화문을 얻어낼 수 있다. 복호화 과정 같은 경우 내부 함수는 Inverse SubBytes, Inverse ShiftRows, Inverse MixColumns 등 역연산에 해당되는 대응 함수를 이용하여 기존 암호화 순서와 반대로 진행되며 이때 사용되는 key는 암호화 시의 round key가 역순으로 사용이 된다.

Key expansion의 경우 처음 입력으로 들어온 master key의 block data 중 4번째 열을 이용하여 확장을 시작하는데, 이를 위로 rotation, S-box를 이용한 대치 후 첫 열과의 XOR과 기존의 정해진 상수 값과의 XOR 연산을 통해 다음 round key block의 첫 번째 열의 데이터를 생성하며, 그 후로는 이전 key의 두 번째 열과 해당 round key의 첫 번째 열과의 XOR 연산을 통한 두 번째 열 생성, 같은 방법으로 세 번째, 네 번째 열의 데이터를 얻어 해당 round key block의 값을 생성해낸다. 이후의 round key 또한 이러한 방식을 통해 128 비트 key의 경우 10개의 key block, 192 비트 key의 경우 12개, 256 비트 key의 경우 14개의 데이터를 생성해낸다. 이러한 key expansion은 실제 암호화를 하는 동안 동시에 생성해 나가며 수행해 나갈 수 있는데 이러한 on-the-fly 방식 또한 많은 경우에 적용되고 있다.

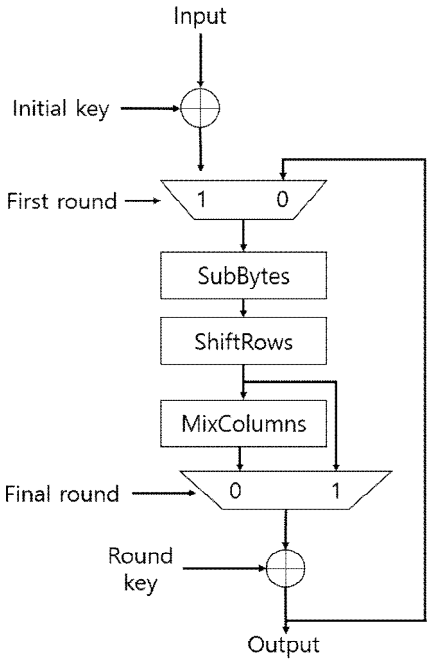

그림 1. AES 알고리즘의 흐름도
Fig. 1. Flow of AES algorithm.

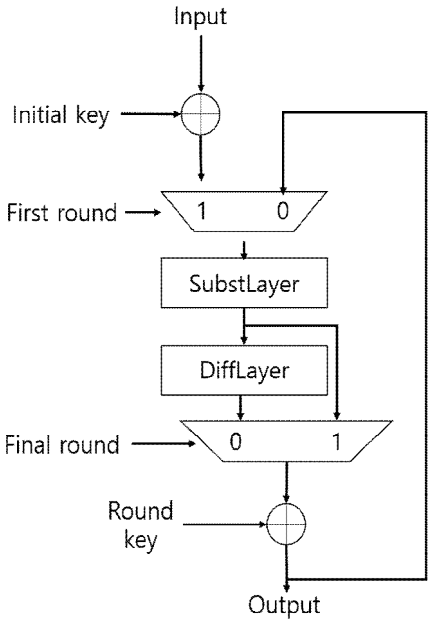

그림 2. ARIA 알고리즘의 흐름도
Fig. 2. Flow of ARIA algorithm.

2-2 ARIA 알고리즘

ARIA는 2004년에 한국 KS 1213-2 국내 표준으로 등재되어 국내 공공기관 및 기업에서 다양하게 사용되고 있다. ARIA 블록 암호 알고리즘은 128 비트 블록 암호 알고리즘이며, 암호화 과정과 복호화 과정이 같은 구조인 Involution SPN (substitution permutation network)로 되어 있으므로 하나의 설계로 두 과정 모두를 사용할 수 있는 이점을 가지고 있다 [8]. ARIA 알고리즘은 128/192/256 비트 암호 키를 사용하여 128 비트의 데이터 블록을 처리하는 알고리즘이며, 암호 키 길이에 따라 round 수가 다르며, 128/192/256 비트 별로 각각 12/14/16 라운드를 거치게 된다. 라운드 함수는 홀수 라운드 함수, 짝수 라운드 함수, 최종 라운드 함수로 구성이 되어 있다. Round 함수는 AddRoundKey, SubstLayer와 DiffLayer로 구성 되어있으며, AddRoundKey는 128 비트 round 키를 128 비트 입력과 XOR 연산을 수행한다. SubstLayer는 8 비트 입, 출력 S-Box 들로 구성되며, S-Box와 이의 역변환으로 이루어진 2개의 유형으로 구성되어 있다. 두 유형의 치환 계층은 교대로 사용되며 전체 구조가 involution 구조가 되도록 한다. DiffLayer는 상태의 바이트들을 섞는 과정으로 16×16 involution 이진 행렬을 사용한다. 입력 16 바이트에 대하여 바이트 단위의 행렬 곱을 수행하여 16 바이트의 출력을 얻어낸다. ARIA 암호화 알고리즘의 전체적인 암호화 및 복호화 과정에 대한 내용은 그림 2와 같다. 그림에서 나타내듯이 첫 번째 라운드가 이루어지기 전 AddRoundKey 연산을 한 번 수행하고, SubstLayer, DiffLayer와 AddRoundKey 세 가지 변환으로 이루어진 하나의 round를 반복한다. 또한 마지막 라운드에서는 DiffLayer를 생략하고, SubstLayer와 AddRoundKey만을 수행한다.

2-3 CLEFIA 알고리즘

CLEFIA는 2007년 일본 SONY에서 개발되어 ISO/IEC 29192-2 국제 표준 알고리즘으로 등록 되어 있으며, 경량화 기법이 적용되어 하드웨어로 구현 시 작은 면적으로 구현 가능한 장점이 있다 [9]. CLEFIA 알고리즘은 GFN (generalized Feistel network)의 구조를 가지고 있으며 입력은 128비트, key size는 128, 192와 256 비트를 지원하며, 이러한 key size에 따라 각각 18, 22와 26 rounds로 구분된다. 그림 3과 같이 GFN 구조에 따라 최초 입력 128 비트를 4개의 branch로 나누어 32비트 단위의 연산을 수행한다. 처음과 마지막 round에서 초기 입력했던 master key를 이용한 whitening key인 WK_0~WK_3이 사용되며, round key가 각각의 F_0와 F_1에 적용되어 수행되는 것을 확인할 수 있으며, F_0과 F_1의 내부는 각각 그림 4와 그림 5에서 볼 수 있다. 32 비트의 데이터를 8 비트 단위로 나누어 연산이 수행되며, 각각 바이트 단위로 구분된 round key와의 XOR 연산이 수행되며 데이터의 위치에 따라 2개의 S-box 중 S_0와 S_1에서 선택하여 대치가 이루어진다. S_0는 $GF(2^4)$ 안에서 4비트 s-box인 SS_i (i=1,2,3,4)을 이용하여 생성이 되며, S_1은 S_0와 달리 $GF(2^8)$ 안에서 affine transform을 이용하여 생성된다. F_0과 F_1에 입력되는 각각의 round key는 DoubleSwap 함수와 32비트 상수 값을 이용하여 생성된다. 초기 입력했던 master key K가 whitening key

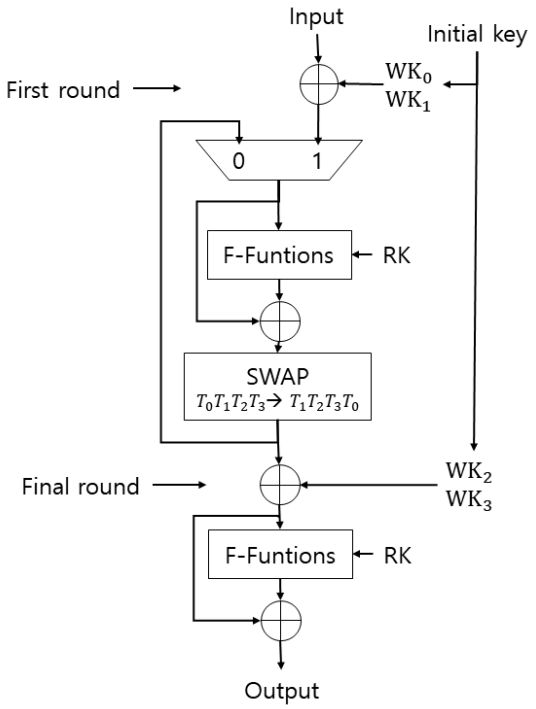

그림 3. CLEFIA 알고리즘의 흐름도
Fig. 3. Flow of CLEFIA algorithm.

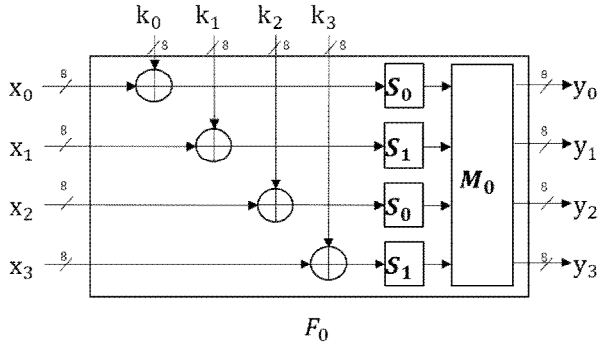

그림 4. CLEFIA의 F_0의 블록도
Fig. 4. Block diagram of F_0 of CLEFIA.

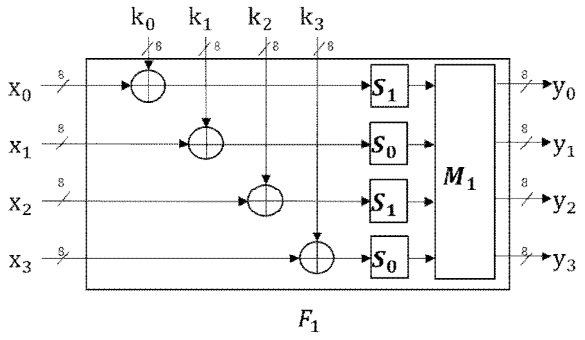

그림 5. CLEFIA의 F_1의 블록도
Fig. 5. Block diagram of F_1 of CLEFIA.

WK_i (i=1,2,3,4)을 생성한다고 하면 중간키 L은 K와 상수 값을 $GFNn,m$의 입력 값으로 한 출력 값이 된다. $GFNn,m$은 n개의 branch를 가지는 m 라운드 GFN 구조이다. 이때, 128 비트 키는 $GFN_{4,12}$를 사용하며, 192/256 비트 키는 $GFN_{8,10}$을 사용한다. Round key RK_i (i=1,⋯,2r-1)는 K와 L 값을 DoubleSwap함수와 XOR 연산을 반복하여 생성한다.

Ⅲ. 제안된 암호화 프로세서의 하드웨어 구조

AES, ARIA와 CLEFIA의 3가지 블록암호를 지원하도록 설계된 제안하는 다중 암호 프로세서의 내부 구조는 그림 6에 나타내었으며, 3가지 운영모드와 128 비트의 1가지 마스터 키 길이를 지원한다. 본 논문에서 제안하는 프로세서는 치환연산을 하는 SU (substitution unit), 확산연산을 하는 DU (diffusion unit)을 포함한 세부적인 연산들과 데이터 레지스터로 구성된다. 전반적으로 암호화 및 복호화 시키는 data processing 모듈과 key를 확장시키는 key processing 모듈을 공유시키는 구조로 구현하였다. 또한, AES 모드에서는 key scheduling과 data processing에서 공통적으로 사용되는 s-box를 포함한 substitution 연산, ARIA와 CLEFIA 모드에서는 key scheduling과 data processing에서 사용되는 모든 path와 module을 공유하였다.

전체적인 데이터 흐름은 컨트롤러에 의해 제어되며 round당 1 클록 주기가 소요된다. 초기 1 round 진행시 처음 입력된 plain text 또는 cipher text가 데이터 레지스터에 입력되지 않고 바로 연산이 진행된다. 1 round 이후 결과는 데이터 레지스터에 입력되어 다음 round가 진행될 때 이용된다. 또한, 암호화와 복호화에서 사용되는 구조를 구분하지 않고 하나의 구조로 통합하여 두 연산을 모두 처리하도록 하였다. AES 모드 암호화의 경우, substitution, shift, diffusion와 add roundkey의 순서로 진행되며, 복호화의 경우 inversion shift, inversion substitution, add roundkey와 inversion diffusion 순서로 연산이 진행되며, 암호화의 경우 21클록이 소요되며 복호화의 경우 22 클록이 소요된다. ARIA 모드의 경우 add roundkey, substitution, diffusion의 순서로 연산되며 암/복호화에 모두 29 클록이 소요된다. CLEFIA 모드의 경우 add roundkey, substitution, Diffusion의 순서로 F-function 연산이 진행되며, CLEFIA data process 블록을 통해 데이터의 암/복호화가 진행된다. AES와 ARIA와 달리 상수 값의 생성과 round key 생성에 86클록이 소요되며 암/복호화에 총 105클록이 소요된다. 치환 연산을 담당하는 SU는 그림 7에 나타낸 Substitution Unit_32 모듈 4개가 결합된 구조이며, 128 비트 입력을 32 비트 4개로 분할하여 치환을 수행한다. 32 비트 데이터를 입력받아 S-box Generator를 통과시켜 결과 값을 얻어낸다. 그림 8은 확산 연산을 담당하는 DU의 하드웨어 구조로, GF(2^8)상에서의 multiplier를 구현하는 것으로 각 알고리즘별로 행렬 곱을 연산한다.

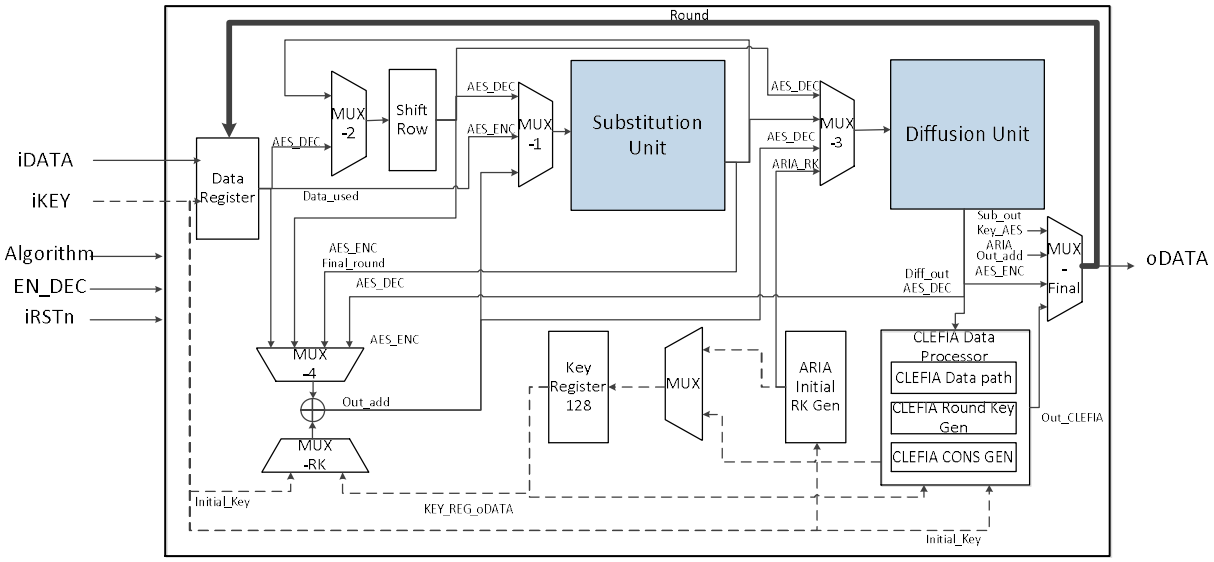

그림 6. 제안된 암호화 프로세서의 하드웨어 구조도
Fig. 6. Hardware architecture of the proposed crypto-processor.

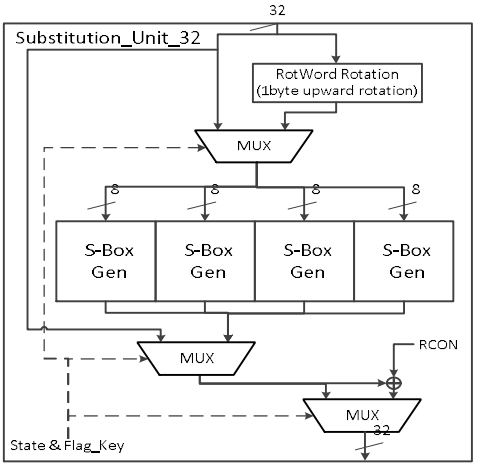

그림 7. SU의 구조도
Fig. 7. Block diagram of the SU.

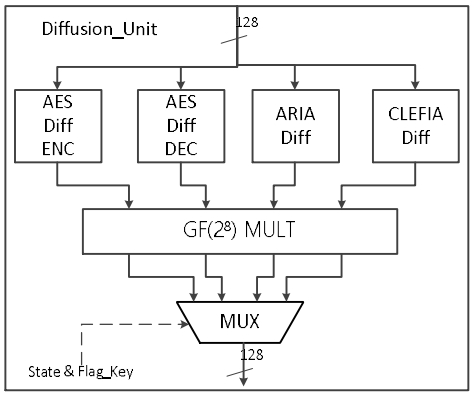

그림 8. DU의 구조도
Fig. 8. Block diagram of the DU.

Ⅳ. 구현 및 결과

본 장에선 제안하는 암호화 프로세서의 설계 및 구현 결과를 제시한다. 본 암호화 프로세서는 Verilog HDL (hardware description language)을 이용하여 RTL 설계 후, Altera Cyclone-Ⅳ FPGA 디바이스를 이용하여 구현 및 검증되었다. 그림 9는 제안하는 암호화 프로세서의 로직 시뮬레이션 결과를 보여주며, 그림 10과 11은 FPGA 플랫폼 기반 암호화와 복호화의 검증 결과를 제시한다. 각 알고리즘 별로 NIST, KISA와 SONY에서 제공하는 test-vector를 메모리에 초기화한 뒤, push-button을 통하여 start 신호를 입력받아 해당 test-vector에 대한 암호화 및 복호화가 진행되며, 그 결과를 text-LCD를 통하여 출력되도록 하였다. 그 결과 정상적으로 암호화 및 복호화가 수행되는 것을 확인하였다. 또한, 표 1은 제안된 암호화 프로세서의 65nm CMOS 공정 기반 논리 합성 결과를 보여준다. 논리 합성결과 총 게이트 수는 약 11K개이며, 기존에 존재하는 IoT 타겟의 암호화 프로세서들과 비교했을 때 제안하는 암호화 프로세서가 면적 측면에서 최대 42.3% 효율적인 것을 확인하였다 [10-12].

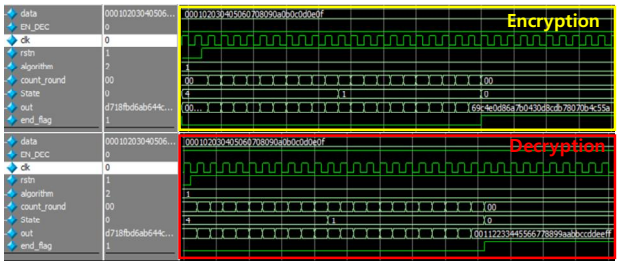

그림 9. 로직 시뮬레이션 결과
Fig. 9. Logic simulation results.

표 1. 제안된 암호화 프로세서의 논리 합성 결과
Table 1. Logic synthesis results of the proposed crypto processor.

	Algorithm	Key length	Data width	Gate Count	Reduction (%)
[10]	AES	128	128	5,398	-
[11]	ARIA	128	128	11,301	-
[12]	CLEFIA	128	128	2,488	-
[10 + 11 + 12]	AES, ARIA, CLEFIA	128	128	19,187	-
Proposed	AES, ARIA, CLEFIA	128	128	11,080	42.3

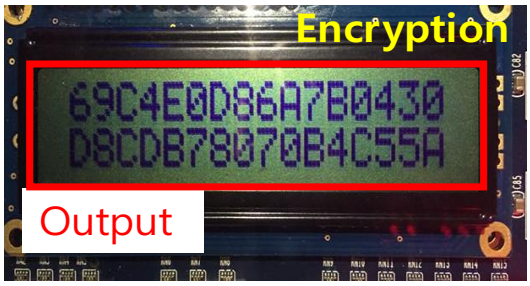

그림 10. FPGA 플랫폼 기반 구현 암호화 결과 (입력 : 00112233 4455667788 99AABBCCDDEEFF)
Fig. 10. Results of encryption based on FPGA platform (Input : 00112233445566778899AABBCCDDEEFF).

그림 11. FPGA 플랫폼 기반 구현 복호화 결과 (입력 : 69C4E0D 86A7B0430D8CDB78070B4C55A)
Fig. 11. Results of encryption based on FPGA platform (Input : 69C4E0D86A7B0430D8CDB78070B4C55A)

Ⅴ. 결 론

본 논문에서는 IoT 응용을 위하여 AES, ARIA와 CLEFIA 암호화 알고리즘을 이용한 통합형 병렬 암호화 프로세서를 설계 및 구현하였다. 제안된 암호화 프로세서는 기본적인 알고리즘의 구조를 분석 및 응용하여 라운드 키 확장부와 데이터 암호화 및 복호화를 하는 전반적인 구조를 공유해, 모듈의 재사용성을 높이며 logic element 사용을 최소화하였다. 이를 통해 제안된 프로세서는 기존의 각 알고리즘의 개별 암호화 프로세서들을 통합한 결과 대비 약 40% 이상의 면적 측면에서의 이점을 가지게 되었다. 이러한 점에서 소형화가 필수적인 IoT 시장에서 제안된 프로세서는 전반적인 IoT 기기에 통용 가능하며, 이를 통하여 다양한 분야에 이용되는 IoT 기기 및 시스템에 응용 가능할 것으로 기대된다.

References

[1] L. Columbus. A roundup of 2018 enterprise internet of things forecasts and market estimates [Internet]. Available: https://www.enterprise-cio.com/news/2018/jan/04/roundup-of-internet-of-things-forecasts-and-market-estimates-2018

[2] B. Gupta. Miniaturization, composability, and the internet of things (IoT) [Internet]. Available: https://dzone.com/articles/miniaturization-composability-and-internet-of-thin

[3] D. P. Leech, S. Ferris and J. T. Scott, The economic impacts of the advanced encryption standard 1996-2017, National Institute of Standards and Technology, Gaithersburg: MD, Technical Report NIST GCR 18-017, 2018.

[4] D. Kwon, J. Kim, S. Park, S. Sung, Y. Sohn, J. Song, Y. Yeom, E. Yoon, S. Lee, J. Lee, S. Chee, D. Han, and J. Hong, "New block cipher: ARIA," in *6th International Conference on Information Security and Cryptology*, Seoul: Korea, pp. 432-445, Nov. 2004.

[5] P. Proenca and R. Chaves, "Compact CLEFIA implementation on FPGAs," in *21st International Conference on Field Programmable Logic and Applications*, Chania: Greece, pp. 512-517, Sep. 2011.

[6] A. Pandey, M. A. Rizvi, "Comparative survey of different cryptographic algorithm," *International Journal of Scientific and Engineering Research*, Vol. 8, No. 5, pp. 41-45, May. 2017.

[7] J. Daemen and V. Rijimen. Announcing the advanced encryption standard (AES), National Institute of Standards and Technology, Gaithersburg: MD, Technical Report FIPS-197, 2011.

[8] Korea Internet & Security Agency (KISA). ARIA-specification [Internet]. Available: https://seed.kisa.or.kr/iwt/ko/bbs/EgovReferenceDetail.do?bbsId=BBSMSTR_000000000002&nttId=39&pageIndex=1&searchCnd=&searchWrd

[9] SONY Corporation. The 128-bit block cipher CLEFIA specification version 1.0 [Internet]. Available:https://www.sony. net/Products/cryptography/clefia/download/index.html

[10] A. Satoh, S. Morioka, K. Takano, and S. Munetoh, "A compact Rijndael hardware architecture with S-box optimization," in *International Conference on the Theory and*

Application of Cryptology and Information Security, Gold Coast: Australia, pp. 239-254, Nov. 2001.

[11] G. Ryu, B. Koo, S. Yang, and T. Chang, "Area efficient implementation of 32-bit architecture of ARIA block cipher using light weight diffusion layer," *The Journal of The Korea Institute of Information Security and Cryptology*, Vol. 16, No. 6, pp. 15-24, Dec. 2006.

[12] T. Akishita and H. Hiwatari, "Very compact hardware implementations of the blockcipher CLEFIA," in *International Workshop on Selected Areas in Cryptography*, Toronto: Canada, pp. 278-292, Aug. 2012.

안 재 욱 (Jae-Uk Ahn)

2013년 3월 ~ 현재 : 한국항공대학교 항공전자정보공학부

※관심분야 : 디지털 회로 설계

최 재 혁 (Jae-Hyuk Choi)

2012년 3월 ~ 현재 : 한국항공대학교 항공전자정보공학부

※관심분야 : 디지털 회로 설계

하 지 웅 (Ji-Ung Ha)

2013년 3월 ~ 현재 : 한국항공대학교 항공전자정보공학부

※관심분야 : 디지털 회로 설계

정 용 철 (Yongchul Jung)

2015년 8월 : 한국항공대학교 전자 및 항공전자공학과 (공학사)

2017년 2월 : 한국항공대학교 항공전자정보공학부 (공학석사)

2017년 3월 ~ 현재 : 한국항공대학교 항공전자정보공학부 박사과정

※관심분야 : 레이다 시스템, 레이다 SoC 설계

정 윤 호 (Yunho Jung)

1998년 2월 : 연세대학교 전자공학과 (공학사)

2000년 2월 : 연세대학교 전기전자공학과 (공학석사)

2005년 2월 : 연세대학교 전기전자공학과 (공학박사)

2005년 ~ 2007년 : 삼성전자 책임연구원

2007년 ~ 2008년 : 연세대학교 연구교수

2008년 ~ 현재 : 한국항공대학교 항공전자정보공학과 교수

※관심분야 : 무선 통신 시스템, 항공통신 시스템, 레이다 시스템, SoC 설계