경량 블록 암호 CLEFIA-128/192/256의 FPGA 구현

배기철* · 신경욱*

*금오공과대학교

An FPGA Implementation of Lightweight Block Cipher CLEFIA-128/192/256

Gi-Chur Bae* · Kyung-Wook Shin*

*Kumoh National Institute of Technology

E-mail : bae921216@kumoh.ac.kr

요 약

본 논문은 128/192/256-비트의 마스터키 길이를 지원하는 경량 블록 암호 알고리즘 CLEFIA-128/192/256의 FPGA 설계에 대하여 기술한다. 라운드키 생성을 위한 중간키 생성과 라운드 변환이 단일 데이터 프로세싱 블록으로 처리되도록 설계하였으며, 변형된 GFN(Generalized Feistel Network) 구조와 키 스케줄링 방법을 적용하여 데이터 프로세싱 블록과 키 스케줄링 블록의 회로를 단순화시켰다. Verilog HDL로 설계된 CLEFIA 크립토 프로세서를 FPGA로 구현하여 정상 동작함을 확인하였다. Vertex5 XC5VSX50T FPGA에서 1,563개의 LUT FilpFlop pairs로 구현되었으며, 최대 112 Mhz 81.5/69/60 Mbps의 성능을 갖는 것으로 예측되었다.

키워드

CLEFIA, Cryptography, Block Cipher, Security

Ⅰ. 서 론

유·무선 통신 시스템의 보편화에 따라 통신망을 통한 정보의 유통이 급격하게 증가하고 있으며, 이에 따라 통신망을 통해 유통되는 정보가 제삼자에게 유출되거나 위·변조 되지 못하도록 하는 정보보안의 중요성이 날로 높아지고 있다. 특히, 무선 환경에서는 기지국 영역 내에 있는 모든 단말기들이 다른 사람의 정보를 수신할 수 있으므로, 허가된 수신자 이외에 제 3자가 정보를 알지 못하게 하는 데이터 기밀성과 사용자인증 등 정보보안 기술이 필수적으로 요구된다.[1]

CLEFIA는 디지털 저작권 관리(DRM) 시스템을 위해 소니에 의해 개발된 블록 암호 알고리듬이며, 128/192/256-비트 마스터키를 사용한다.[2] CLEFIA는 선형공격, 불능 차분공격 등의 보안공격에 대한 안전성이 입증되었고, 경량 구현이 가능하여 RFID, IoT의 보안에 적합한 것으로 평가되고 있다.

본 논문에서는 블록암호 CLEFIA를 IoT 환경에 적합하도록 저면적으로 설계하였으며, FPGA 구현을 통해 하드웨어 동작을 검증하였다. 암호화/복호화와 중간키 생성을 위한 하드웨어 자원의 공유를 통해 설계를 최적화하였다.

Ⅱ. 경량 블록암호 알고리듬 CLEFIA[2]

CLEFIA는 128-비트의 블록길이와 세 가지 마스터키 길이(128/192/256-비트)를 지원하는 대칭키 블록암호이다. 세 가지 마스터키 길이에 따라 18/22/26회의 라운드 변환을 통해 암호화/복호화가 이루어지며, 라운드 변환과 중간키 생성은 변형된 Feistel 구조인 GFN (Generalized Feistel Network)을 기반으로 한다. GFN은 4-branch와 8-branch의 두 가지 형태가 사용되며, 4-branch GFN은 라운드 변환과 128-비트의 중간키 생성에 사용되고, 8-branch GFN은 192/256-비트의 마스터키로부터 256-비트 중간키 생성에 사용된다.

CLEFIA의 암호화/복호화 데이터 프로세싱은 그림 1과 같이 4-branch GFN인 $GFN_{4,r}$로 구성된다. 암호화와 복호화는 역순으로 이루어지며, 라운드키 가산 순서와 순환이동의 방향도 반대로 이루어진다. 암호화/복호화 라운드 변환의 F-함수에는 라운드키 RK_i가 사용되며, 라운드키 RK_i는 마스터키를 GFN으로 처리하여 만들어지는 중간키, 키 스케줄러 내부에서 생성되는 상수값 그리고 마스터키의 XOR 연산에 의해 생성된다. 중간키는 마스터키를 평문처럼 GFN으로 처리하여 생성되며, 이때 키 스케줄러 내부에서 생성되는 상

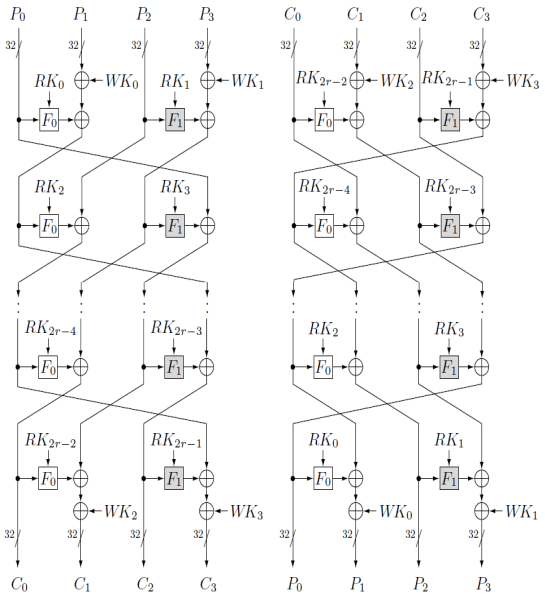

그림 1. CLEFIA의 암호화/복호화 프로세싱 구조
Fig. 1. Encryption/decryption processing of CLEFIA

수값은 F-함수에서 라운드키로 사용된다.

데이터 프로세싱 블록을 구성하는 F-함수 F_0, F_1은 라운드키 가산을 위한 XOR, 4개의 비선형 S-box, 확산 매트릭스 M_0, M_1의 곱셈으로 구성된다. S-box 출력과 M_0, M_1의 곱셈은 기약다항식 $p(z)=z^8+z^4+z^3+z^2+1$으로 정의되는 유한체 $GF(2^8)$에서 연산된다.

키 스케줄러는 GFN에 의해 생성된 중간키를 저장한 뒤 더블 스왑(double swap)을 통해 갱신해가며 마스터키와 라운드 상수값의 XOR 연산으로 라운드키를 생성한다. 더블 스왑은 순환이동의 한 형태이며, 상수값 생성 블록은 마스터키 길이에 따라 정해지는 초기값을 갱신해가며 on-the-fly 방식으로 상수값을 생성한다.

III. CLEFIA 크립토 프로세서 설계

128-비트의 평문/암호문 블록을 암호화/복호화하여 128-비트 암호문/평문을 생성하는 CLEFIA 크립토 프로세서를 설계하였다. 설계된 프로세서는 128/192/256-비트의 세 가지 길이의 마스터키를 지원하며, 하드웨어 자원의 최소화를 위한 다양한 방법을 적용하였다. 프로세서의 전체 구조는 그림 2와 같으며, 데이터 프로세싱 블록, 키 스케줄러, 제어 블록으로 구성된다.

3.1 데이터 프로세싱 블록

데이터 프로세싱 블록은 암호화/복호화 과정의 라운드 변환을 위한 4-branch GFN과 중간키 생

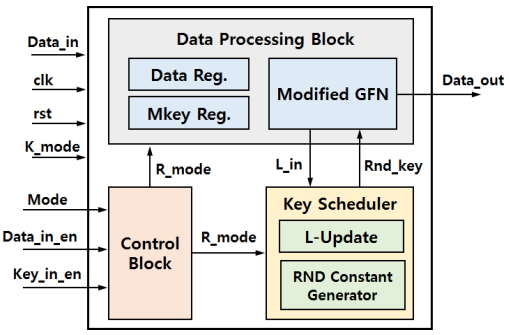

그림 2. CLEFIA 크립토 프로세서의 구조
Fig. 2. Architecture of CLEFIA crypto-processor

성을 위한 8-branch GFN 기능을 수행한다. 하드웨어 복잡도를 최소화하기 위해 다음과 같은 방법을 적용하여 설계하였다.

본 논문에서는 문헌 [3]에서 제안된 방법을 적용하여 그림 3과 같이 F-함수를 8-비트 데이터 패스로 축소하여 설계하였으며, 이를 통해 확산 매트릭스 M_0, M_1 곱셈과 S-box에 의한 하드웨어를 최소화하였다. 각각의 반쪽 라운드에 4 사이클이 소요되므로, 한 라운드가 8 사이클에 처리된다. 암호화/복호화에 소요되는 사이클 수는 늘어났지만, GFN에 필요한 F-함수의 개수, F-함수에 필요한 S-box, XTIME, XOR 등이 최소화되었다. 설계된 데이터 프로세싱 블록은 그림 4과 같은 구조를 갖는다. 8개의 32-비트 레지스터 $R_0 \sim R_7$와 256-비트의 마스터키 레지스터 $MKReg$, 매트릭스 곱셈기, S0/ S1-box 그리고 마스터키 및 라운드키 가산을 위한 XOR 게이트 등으로 구성된다.

8비트 데이터 치환을 수행하는 S-box는 조합회로를 기반으로 설계하였다. S-box S_0는 4-비트 비선형 S-box SS_x 4개와 XTIME 블록 2개 그리고 XOR를 사용하여 구현하였다. 이와 관련된 모든 연산은 기약다항식 $q_0(z)=z^4+z+1$로 정의되는 유한체 $GF(2^4)$에서 연산된다. 한편, S-box S_1은 유한체 $GF(2^8)$ 기반의 연산 대신에 $GF((2^4)^2)$의 복합체(composite field) 연산을 이용하여 설계함으로써 하드웨어가 최소화되도록 하였다. S-box S_1의 연산은 기약다항식 $q_0(z)=z^4+z+1$으로 정의되는 유한체 $GF(2^4)$와 $q_1(z)=z^2+z+\lambda$로 정의되는 유한체 $GF((2^4)^2)$에서 연산된다.

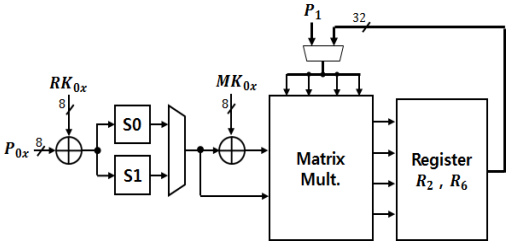

그림 3. 통합 F-함수 구조
Fig. 3. Unified F-function structure

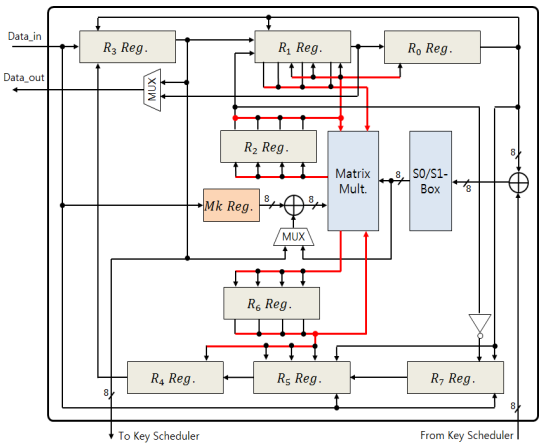

그림 4. 데이터 프로세싱 블록
Fig. 4. Data processing block

3.2 키 스케줄러 블록

CLEFIA의 라운드키는 마스터키로부터 생성되는 중간키와 라운드에 따라 생성되는 상수값을 XOR 연산하여 생성된다. 중간키는 마스터키를 GFN으로 처리하여 생성되며, 마스터키가 128-비트인 경우에는 4-branch GFN로부터 128-비트 중간키가 생성되고, 마스터키가 192/256-비트인 경우에는 8-branch GFN에 의해 256-비트 중간키가 생성된다.

본 논문에서는 GFN 공유 개념을 적용하여 데이터 프로세싱 블록의 GFN이 중간키 생성에도 사용되도록 함으로써 하드웨어 최적화를 이루었다. 설계된 키 스케줄러는 중간키 레지스터, 더블스왑 블록, 라운드에 따라 상수값을 생성하는 상수생성기 그리고 중간키와 라운드 상수값을 연산하는 XOR 게이트 등으로 구성된다.

Ⅳ. 기능 검증 및 FPGA 검증

Verilog HDL로 설계된 CLEFIA 크립토 프로세서는 FPGA 구현을 통해 하드웨어 동작을 검증하였다. FPGA 보드, UART 인터페이스, 구동 소프트웨어로 구성된 검증시스템은 그림 5와 같으며, Xilinx Virtex5 XC5VSX50T FPGA 디바이스가 사용되었다. PC에서 입력된 비밀키와 평문/암호문 데이터가 RS232C 통신을 통해 FPGA로 입력되고, FPGA에서 출력되는 암호문/평문 데이터가 표시된다. 평문을 암호화하고, 암호문을 복호화하여 원래의 평문과 일치하는 복호결과가 출력되어 FPGA에 구현된 CLEFIA 프로세서가 올바르게 동작함을 확인하였다.

설계된 CLEFIA 프로세서는 FPGA 합성결과 1,563개의 LUT-FF pairs로 구현되었다. 112 MHz 클록 주파수로 동작할 때, 128/192/256비트의 3가지 마스터키 길이에 따라 81.5/69/60 Mbps 성능을 갖는 것으로 평가되었다.

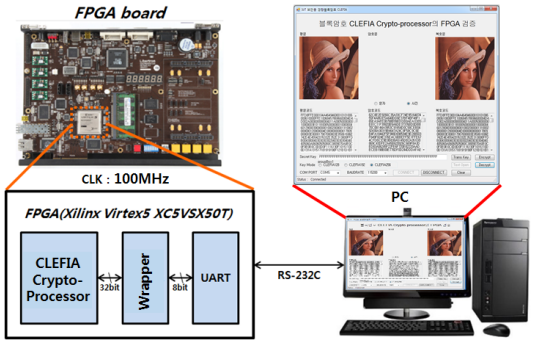

그림 5. CLEFIA 크립토 프로세서의 FPGA 검증
Fig. 5. FPGA verification of CLEFIA crypto-processor

Ⅴ. 결 론

소니에서 개발되어 ISO IEC 국제표준으로 승인된 128-비트 블록암호 CLEFIA를 FPGA로 구현하여 동작을 확인하였다. 암호화/복호화 라운드 연산과 중간키 생성이 단일 데이터 프로세싱 블록에서 처리되도록 설계하여 하드웨어 복잡도를 최소화였다. 설계된 CLEFIA 크립토 프로세서는 IoT 및 RFID 환경 등과 같이 경량화가 요구되는 응용분야의 정보보호 코어로 활용이 가능하다.

감사의 글

※ 반도체설계교육센터(IDEC)의 CAD Tool 지원에 감사드립니다.

참고문헌

[1] W. Stalling, *Cryptography and Network Security*, Prentice Hall, 1999.

[2] The 128-bit Block Cipher CLEFIA : Algorithm Specification, Sony Corp., 2007.

[3] T. Akishita and H. Hiwatari, "Very Compact Hardware Implementations of the Block Cipher CLEFIA," in Selected Areas in Cryptography－SAC 2011, ser. LNCS, vol. 7118, pp. 278-292, Springer-Verlag, 2012.