SEED 블록 암호 알고리즘의 단일 칩 연구

신 종 호, 강 준 우
한국외국어대학교 전자제어공학과
전화 : (0335)330-4502 팩스 : (0335)330-4120

Study of one chip SEED block cipher

Jong-Ho Shin, Jun-Woo Kang
Dept. of Electronic and Control Engineering
Hankuk University of Foreign Studies
E-mail : shinjks@san.hufs.ac.kr

Abstract

A hardware architecture to implement the SEED block cipher algorithm into one chip is described. Each functional unit is designed with VHDL hardware description language and synthesis tools. The designed hardware receives a 128-bit block of plain text input and a 128-bit key, and generates a 128-bit cipher block after 16-round operations after 8 clocks. The encryption time is within 20 nsec.

I. 서 론

인터넷을 통한 전자상거래의 활성화로 개인 신상 관련 자료나 중요 정보에 대한 보호 수단이 필요하게 되었다. 그러나 정보 보호에 관한 연구는 몇몇 선진국에서만 행하여 지고 있는 실정으로, 자국의 정보 보호에 관한 기술 개발은 앞으로 전개될 인터넷 시대에 국가 경쟁력 확보의 중요 수단이 될 것이다. 따라서, 국내에서도 한국정보보호센터를 중심으로 1998년 SEED 블록 암호 알고리즘을 개발하여 이를 공개하고 표준화 시켜 국내의 전자상거래등에 응용할 계획이다.[1-3]

본 논문에서는 디지털 정보 보호를 위한 SEED 블록 암호 알고리즘의 하드웨어 구조를 설계하여 단일 칩으로 구현하는 방안을 제시하였다. SEED 블록 암호 알고리즘은 128비트로 고정된 한 블록의 평문을 128비트의 비밀 키를 이용하여 같은 비트 블록의 암호문으로 암호화하며 동일한 비밀 키를 이용하여 복호화 시키는 비밀 키 암호 알고리즘이다. SEED 블록암호알고리즘을 단일 칩으로 구현하기 위하여 칩의 전체 구조와 각종 기능 블록을 설계하고, 설계된 칩의 구조와 각 모듈을 기능 수준에서 하드웨어 기술 언어를 사용하여 모델링 한 후, 시뮬레이션을 통하여 그 결과를 확인하였다.

II. SEED 암호 알고리즘

SEED는 대칭 키 암호 알고리즘으로서, 블록 단위로 메시지를 처리하는 블록암호알고리즘이다. 대칭 키 블록암호알고리즘은 기밀성을 제공하는 암호시스템의 중요 요소이다. n비트 블록암호알고리즘이란 고정된 n비트 평문을 같은 길이의 n비트 암호문으로 바꾸는 함수를 말한다. 이러한 변형과정에 암호키가 작용하여 암호화와 복호화를 수행한다.

대부분의 블록 암호알고리즘은 Feistel구조로 되어있다. Feistel 구조란 각 t비트인 L0, R0블록으로 이루어진 2t비트 평문 블록(L0, R0)을 r라운드를 거쳐 암호문(Lr, Rr)을 내는 반복 구조를 말한다. 이 구조는 라운드 함수에 관계없이 역변환이 가능하며, 두 번의 수행만으로도 블록간의 완전한 Diffusion이 이루어진다. 그리고 알고리즘의 수행 속도가 빠르며, 소프트웨어나 하드웨어로의 구현이 용이하다는 특징도 있어 다른 암호화 과정에서도 많이 사용되고 있는 구조이다.[2, 11] SEED는 128비트의 평문 블록 단위당 128비트 키로부터 생성된 64비트의 라운드 키(16개)를 입력으로 받아 총 16라운드를 거쳐 128비트 암호문 블록을 출력으로 낸다. 내부 연산은 Modular adder와 Exclusive-OR만을 사용하였고 암호화에 사용된 기본 함수는 F-함수와 G-함수가 있으며, 라운드 키 생성에도 같은 연산과 함수를 사용한다.

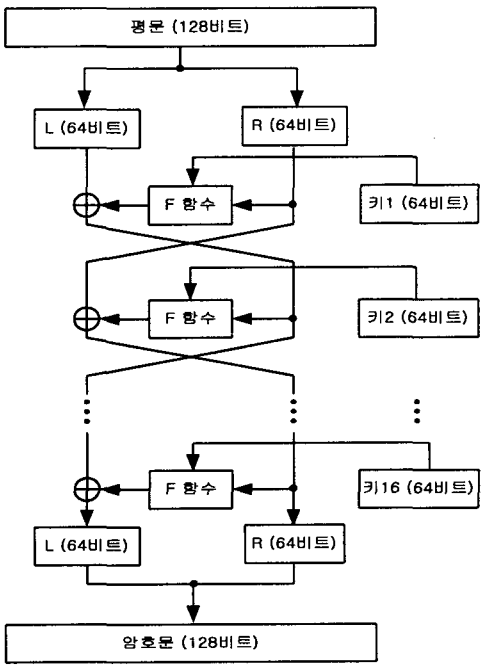

그림 1. SEED 암호 알고리즘의 전체 구조도

(그림 1)에 Feistel 구조를 바탕으로 한 SEED 암호 알고리즘의 전체 구조도를 나타내었다.

III. SEED칩의 구조설계

SEED 알고리즘은 그 바탕이 Feistel 구조이므로, 그대로 하드웨어로 구현한다면 단일 블록이 반복되는 구조로 설계 될 수 있다. 이 구조는 DES (Data Encryption Standard) [4] 암호 알고리즘의 경우와 마찬가지로 파이프라인의 구조로 하드웨어를 구성할 수도 있지만, 파이프라인 구조에서는 먼저 사용된 스테이지의 하드웨어가 다음 입력이 공급될 때까지 사용되지 않아서 칩 면적의 낭비를 초래할 수 있으므로, 128비트 Register와 반복되는 하나의 블록을 사용하여 매 클럭 마다 카운터를 증가시키면서 16 클럭을 수행한 후 출력을 내보내는 구조로 설계할 수 있다. 본 논문에서는 특성이 다른 키 생성 블록을 하나의 클럭에 동작하도록 하여 면적에 손해를 보더라도 속도를 높일 수 있게 하였다. (그림 2)는 SEED 알고리즘을 하드웨어로 구현한 전체 블록도를 나타낸 것이다. 제어블록은 매 클럭 마다 입력으로 받은 128비트 평문값과 128비트 라운드 키 값을 다음 클럭에 암호화 블록인 SEED Unit 블록과 라운드 키 생성 블록의 입력으로 보내는 기능과 총 16라운드 수행 후 암호화된 블록을 내보내는 기능을 수행한다.

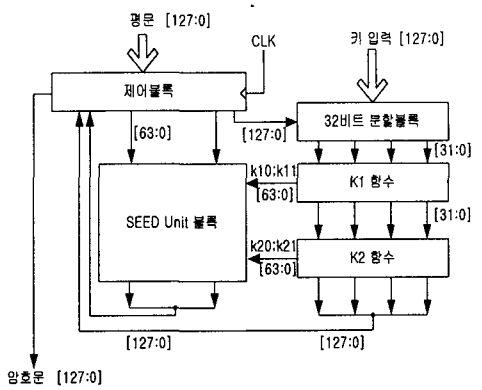

그림 2. SEED하드웨어의 전체 블록도

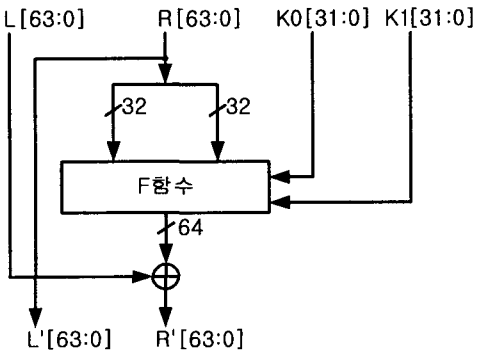

그림 3. SEED Unit블록의 구조도

32비트 분할 블록은 입력된 128비트를 알고리즘에 따라 32비트 4개의 블록으로 나누는 기능을 수행한다. 다음 (그림3)은 SEED Unit블록의 내부 구조를 나타낸 것으로 F함수로 구성된 하나의 단계를 수행하는 암호화 블록이다. SEED 알고리즘을 구성하고 있는 F함수나 G함수는 알고리즘의 설계 단계에서부터 하드웨어로의 구현이 용이하도록 설계되었다. (그림4)의 G함수의 경우, 입력 32비트를 8비트씩 나눈 다음, Truth Table형태로 구현한 S-Box의 입력으로 넣어주고, 결과값의 Permutation은 출력선을 서로 바꾸어 연결시키는 방법으로 구현하였다. S_Box는 비선형 변환을 일으키는 부분으로 암호화에서 가장 중요한 역할을 하는 부분이다. SEED 알고리즘에 사용되는 S_Box는 8비트의 입력값 (0~255)을 받아 같은 비트의 함수의 결과값을 가지고 있으며, 본 설계에서는 S_Box를 PLA (Programmable Logic Array)를 이용하여 구현하였으나, 구현된 S_Box의 면적이 너무 커서 SEED를 단일 칩으로 구현하기가 어려웠다.

SEED를 단일 칩으로 구현하기 위하여 S_Box를 키 생성 블록을 포함한 암호화 부분과 분리시켜 외부에 두었다. 외부에 구현된 S_Box는 32-bit의 입/출력을 하나의 port로 처리하는 총 10개의 port로 구성되어 있다. 그 이유는 한 라운드 동안에 총 10번의 G 함수 출력이 있기 때문이며, 이렇게 구현된 S_Box 블록은 10번의 비선형 처리를 한 라운드에 수행할 수 있다. S_Box의 구현 방법에는 앞에서처럼 Truth Table로 구현하는 방법, 변환에 사용된 함수를 하드웨어로 구현하는 방법, 그리고 Decoder와 Encoder를 이용하여 구현하는 방법 등이 있다.[6]

(그림 5)는 F-함수를 나타낸 것이다. 여기서 가산기는 CLA(Carry Look-ahead Adder)를 사용하여 구현하였는데[10], 이는 보다 빠른 속도로 연산을 수행하여 전체 시스템에서 더 빠르게 결과 값을 얻기 위함이다.

라운드 키 블록의 구현에 사용된 덧셈기는 F함수의 경우와 마찬가지로 CLA를 사용하였으며 KCi값은 S-Box와 마찬가지로 Table로 작성된 것을 그대로 사용하였다.

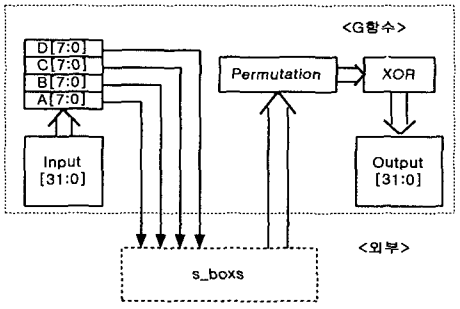

그림 4. G함수의 구현 구조

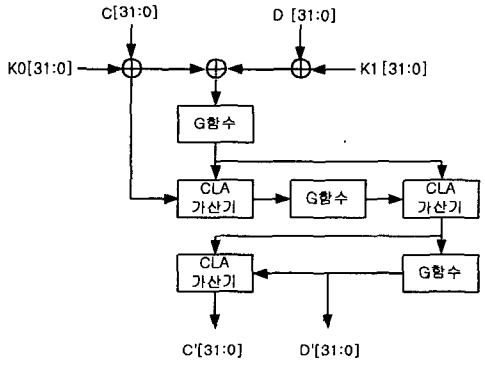

그림 5. F함수의 구현

여기서 사용된 KCi값을 처음 값 (KC0) 인 "0x9e3779b9"를 1비트씩 shift하여 생성하는 방법도 있으나 이 방법은 매 라운드마다 shift연산을 수행하여야 하기 때문에 지연시간이 더 길어지는 단점이 있다.[11] 다음 (그림6)은 하드웨어 구현 시에 사용된 두개의 라운드 키 생성 블록인 K1, K2 중에서 K1블록을 나타낸 것이다.

IV. 시뮬레이션 결과 및 분석

SEED Unit 블록, K1,K2함수 블록등 각종 기능 블록들을 Hardware Description Language로 기술하여 시뮬레이션을 통해 결과값을 확인한 후, 제어 블록을 포함한 Top 블록을 완성하였다. Top블록에 대해서는 먼저 RTL수준에서 시뮬레이션 결과값을 확인하고, LG 0.6마이크론 셀 라이브러리를 사용해 합성하였다. 합성 조건으로 지연시간과 면적을 최소화하는데 중점을 두었으며, 합성회로의 전력값은 평균이 되도록 하였다. 하드웨어로 구현된 SEED구조에서는 각 라운드마다 키 값을 출력시키며, 평문의 길이와 암호화된 문장의 길이는 128비트로 이루어져 있다. (그림 7)은 라운드 키 K1, K2함수 중에서 K1함수의 시뮬레이션 결과 값을 나타낸 것이며, (그림 8)은 SEED블록 전체의 시뮬레이션 결과 값을 나타낸 것이다.

(그림7)의 시뮬레이션 결과는 라운드키 생성 1단계를 거친 후의 시뮬레이션 결과로 암호키를 128비트의 0값을 주었을때이다. "KEY_COUNT" 라는 신호의 값이 16단계 중 첫 단계임을 나타낸다. 입력 값은 128비트가 32비트씩 나누어 입력된 값이고 출력도 32비트 4개로 나누어 출력된 값이다. (그림8)의 시뮬레이션 결과는 16단계를 거친 완전히 암호화 된 결과 값이다. "OUTP" 라는 신호가 암호화된 결과 값을 나타낸다. 처음 암호화 블록은 8클럭이 지난 후 생성되었다. 확인을 위해 비교한 결과값은 정보보호센터에서 제공한 소프트웨어로 구현된 SEED알고리즘의 결과값이다.

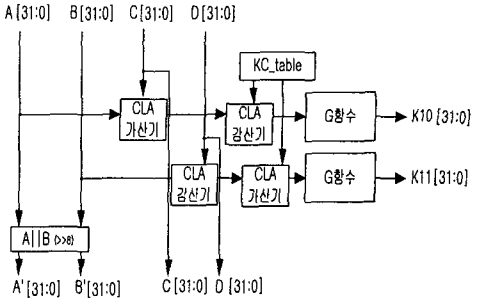

그림 6. 라운드 키 K1 생성 블록

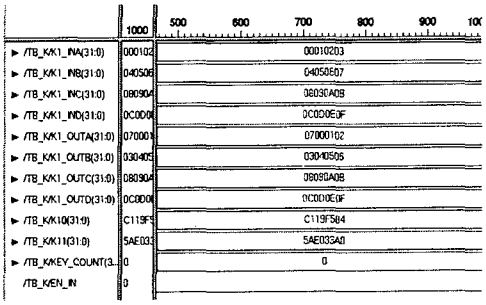

그림 7. 라운드 키 K1의 시뮬레이션 결과

SEED알고리즘에서는 4개의 S_Box가 한 라운드에서 5개의 G 함수 내에 각각 존재하여 전체 시스템의 지연시간에 결정적인 역할을 한다. 즉, S_Box의 경우에는 적은 면적으로 구현하는데 많은 어려움이 있다. 또한, 가산기 및 감산기를 CLA로 구현하여 속도의 향상을 제공받더라도 많은 면적을 차지하는 단점을 가지고 있음을 알 수 있다. 따라서, 이러한 면적에 관한 문제의 보완을 위해서 SEED알고리즘은 수정해야 할 필요성이 있다.

(그림 9)에 나타낸 바와 같이, 암호화부와 라운드 키 생성부를 포함한 합성된 하드웨어의 총 면적은 14,300 게이트로서, 단일 칩 으로 구현될 수 있다. 향후 S_Box를 비동기 ROM으로 구현하면 더 빠른 결과를 얻을 수 있을 것이다.

V. 결 론

본 논문에서는 SEED 블록 암호 알고리즘을 VHDL을 이용하여 하드웨어로 설계하고 검증하였으며, 단일 칩으로 구현될 수 있는 구조를 제안하였다. 검증된 결과는 Synopsys 도구를 사용하여 합성하였다. 합성된 하드웨어의 성능을 보면, 암호화에 소요된 클럭 수는 8클럭이고, 처리시간은 19.12 ns이다.

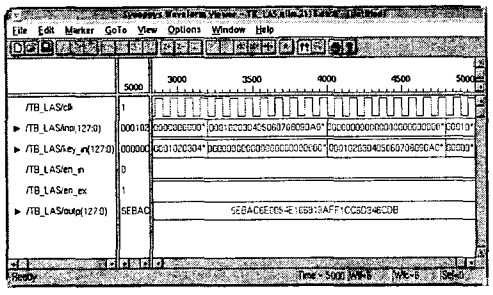

그림 8. SEED 블록 전체의 시뮬레이션과 결과

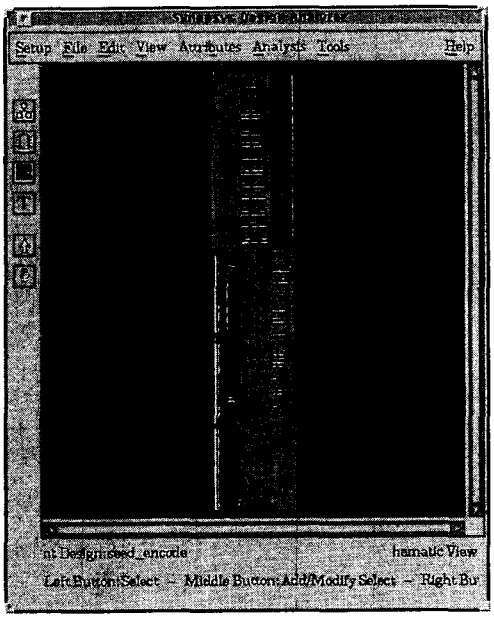

그림 9. SEED 칩의 합성 결과

참고문헌

[1] 김 철, 암호학의 이해, 영풍문고, 1996년10월.

[2] A Design and Analysis of SEED, 한국정보보호센터, 1998년12월.

[3] 한국정보보호센터, http://www.kisa.or.kr/

[4] "Data Encryption Standard", National Bureau of Standards (U.S.) FIPS PUB 46-1, National Technical Information Service, Springfield VA, 1988.

[5] M. J. Wiener, "Efficient DES Key Search", Rump session of Crypto'93, Aug. 1993.

[6] 강준우, "DES알고리즘의 고속 단일칩 구현 연구", 한국외국어대학교 논문집 제31집, pp. 493-509, 1999년 6월.

[7] 암호화 칩 설계, 반도체설계교육센터, 2000년2월.

[8] C. E. Shannon, "Communication theory of secrecy systems", Bell System Journal , vol. 28, pp.656-715, Oct. 1949.

[9] 128-bit Symmetric Block Cipher, 한국정보통신기술협회, 1999년4월.

[10] K. Hwang, Computer Arithmetic, John wiley & sons Inc., 1979.

[11] 염동복, 박종서, "PLD를 이용한 블록 암호화 알고리즘 SEED구현", IDEC 5th MPW Proceedings, 1998년2월.