SEED 암호 보조 프로세서의 CPLD 구현

CPLD Implementation of SEED Cryptographic Coprocessor

최병윤, 김진일
Byeong-Yoon Choi and Jin-IL Kim

요 약

본 논문에서는 SEED 알고리즘을 구현하는 암호 보조 프로세서를 CPLD로 구현하였다. 속도 와 면적 사이의 상반 관계를 고려하여, 암호 보조 프로세서는 1 라운드 동작을 3개의 부분 라운드로 나누고, 클록마다 하나의 부분 라운드를 수행하는 구조를 갖는다. 동작속도를 향상시키기 위해서 암호 및 복호 동작의 라운드 키를 온라인 사전 계산 기법을 사용하여 계산하였으며, 다양한 분야에 응용할 수 있도록 4가지 동작 모드를 지원한다. 설계한 암호 프로세서는 알테라사 EPF10K100GC503-3 디바이스에 구현하고, PC ISA 버스 인터페이스를 통한 문서 파일에 대한 암 · 복호화 동작을 통해 올바른 동작이 이루어짐을 확인하였다. 설계된 회로는 약 29,300개의 게이트로 구성되며 CPLD상에서 약 18Mhz의 동작 주파수를 가지며, ECB 동작 모드에서 약 44 Mbps의 암 · 복호율의 성능을 얻을 수 있었다.

ABSTRACT

In this paper CPLD design of cryptographic coprocessor which implements SEED algorithm is described. To satisfy trade-off between area and speed, the coprocessor has structure in which 1 round operation is divided into three subrounds and then each subround is executed using one clock. To improve clock frequency, online precomputation scheme for round key is used. To apply the coprocessor to various applications, four operating modes such as ECB, CBC, CFB, and OFB are supported. The cryptographic coprocessor is designed using Altera EPF10K100GC503-3 CPLD device and its operation is verified by encryption or decryption of text files through ISA bus interface. It consists of about 29,300 gates and performance of CPLD chip is about 44 Mbps encryption or decryption rate under 18 Mhz clock frequency and ECB mode.

I. 서 론

인터넷과 컴퓨터망 기술의 발달에 따라, 정보 공유라는 긍정적인 측면과 정보의 유출 가능성이 높아지는 부정적인 측면이 존재한다. 따라서 정보화 사회를 구현하기 위해 정보 보호 및 보안 서비스에 대한 연구가 필수적이다. 특히 전자 상거래 및 인터넷을 통한 정보 서비스를 사용자들이 신뢰하며 사용하기 위해서는 정보 시스템의 보안과 처리 속도가 우선적으로 보장되어야 한다[1]. 정보 보호를 위한 소프트웨어 구현 방식은 하드웨어 방식에 비해 암호화 속도와 키 관리의 안전성 측면에서 문제를 야기할 가능성이 높다. 그러므로 고속 통신 시스템에 암호화를 적용하거나, 키의 보다 안전한 관리를 위해서는 암호 알고리즘의 하드웨어 구현이 필요하다. 최근에 FPGA(Field Programmable Gate Array)와 CPLD(Complex Programmable Logic Device) 기술의 발달로 수십만 게이트 규모의 회로를 구현하는 것이 가능하게 되었다. 일반적으로 이러한 프로그램 가능한 디바이스는 내부 구조상의 특성으로 고속 동작은 불가능하지만, 고속 동작

이 필요하지 않은 응용 분야나 실험실에서 최종 단계 이전의 시제품(prototype)을 제작하는데 널리 활용되고 있다. 현재 보편적으로 널리 사용되고 있는 DES(Data Encryption Standard) 암호 알고리즘은 고속 프로세서의 개발로 알고리즘 자체의 안전성에 위협이 되고 있는 상황이다[2]. 따라서 미국에서는 조만간 DES를 대신할 새로운 대칭형 암호 표준 AES(Advanced Encryption Standard)[3]가 선정되어 사용될 예정이다. 이러한 추세에 맞추어 한국에서도 독자적인 128 비트 SEED 암호 알고리즘을 개발하여 표준으로 정하였다[4]. 그러나 SEED 알고리즘은 DES에 비해 안전성은 크게 증가하였지만, 내부 구조의 복잡성으로 하드웨어가 복잡하고 속도가 크게 떨어지는 결점이 있다. 따라서 SEED를 전자 상거래 등 다양한 응용 분야에 적용하기 위해서는 면적 과 속도 측면에서 우수한 성능을 갖는 SEED 암호 프로세서 개발이 필요하다. 기존 SEED 알고리즘을 구현한 연구[5][6]는 면적을 감소시키기 위해, 많은 면적을 갖는 S-박스를 암호칩 외부에 두거나, 모든 라운드 키를 SEED 알고리즘 연산 시작 전에 미리 계산해 두는 기법을 취하고 있다. 따라서 이러한 방식은 외부 메모리 접근에 따른 입출력 문제와 라운드 키의 사전 계산 시간을 포함할 경우 암·복호율이 떨어지는 문제가 있다.

본 논문에서는 라운드 키의 온라인 파이프라인 계산 기법과 모든 S-박스를 칩 내부에 유지하면서 하드웨어 공유 기법을 통해, 하드웨어 크기와 암호화 처리 속도를 개선시킨 SEED 암호 알고리즘 구현 구조를 제안하고, 이를 CPLD 디바이스로 구현한 후, 하드디스크 파일 암호 보드에 적용하여, 그 성능을 분석하였다. 본 논문의 구성은 2장에서는 SEED 암호 알고리즘의 특징을 간단히 기술하고, 3장에서는 제안한 SEED 암호 알고리즘에 대한 하드웨어 구현 방안을, 4장에서는 보조 프로세서의 하드웨어 설계를 다루며, 5장에서는 암호 보조프로세서의 CPLD 구현 및 성능 분석을 기술하였으며, 6장에서는 결론을 제시하였다.

II. SEED 암호 알고리즘

SEED 암호 알고리즘은 그림 1과 같이 128 비트 평문을 2개의 64 비트 블록($L_0(64)$, $R_0(64)$)으로 나눈 후, 16쌍의 라운드 키 $K_i(K_{i,1}, K_{i,0})$와 함께 Feistel 구조의 16 라운드 동작을 수행한 후, 최종 128 비트 데이터($L_{16}(64)$, $R_{16}(64)$)를 출력하는 구조를 갖고 있다. SEED의 동작 특성을 결정하는 그림 2의 F 함수는 Ri(64)값과 라운드 키 K_i ($K_{i,1}$, $K_{i,0}$)를 입력으로 받아, 64 비트 블록 (D', C')을 생성한다. SEED 복호 동작은 암호 동작과 유사하지만 라운드 키 적용 순서가 암호 과정과 반대이다. G 함수는 SEED 암호 알고리즘의 안전성을 구현하는 핵심 부분으로 2쌍의 S1, S2 박스에 대한 테이블 룩업 동작, 마스킹 동작과 XOR 연산을 통해, 32 비트 출력을 생성하는 회로이다. 암호 동작에 사용되는 라운드 키 생성 알고리즘은 128 비트의 마스터 키 또는 중간 라운드의 조정된 마스터키 값을 받아 64 비트씩 좌우로 나눈 후, 이들을 8 비트씩 좌 또는 우로 회전 이동시켜 새로이 조정된 마스터키를 생성함과 동시에 중간 라운드의 4개의 입력 값 (D, C, B, A)에 대해 32 비트 모듈로 덧셈, 뺄셈 및 G 함수를 적용하여 라운드 키(K_i)를 생성한다. 그림 3은 암호 동작에 대한 라운드 키 생성 알고리즘을 나타낸다. 그림에서 ||, <<, 와 >> 표현은 각각 병렬 접속, 좌 또는 우측 순환 이동 동작을 나타낸다.

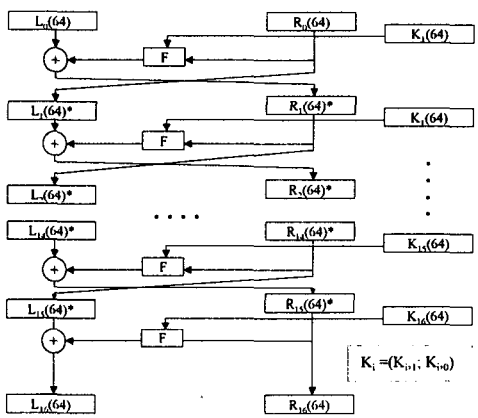


그림 1. SEED 암호 알고리즘의 Feistel 구조
Fig.1. Feistel structure of SEED encryption

키 생성부에 사용되는 라운드 상수 KCi는 식(1)과 같이 결정된다. 여기서 i 값은 암호 동작에 대한 라운드를 나타낸다.

$KC_1 = 0x9e3779b9$
algorithm $KC_i = KC_{i-1} \ll 1$ for $2 \le i \le 16$
여기서 $0x$는 16진수 데이터를 지시

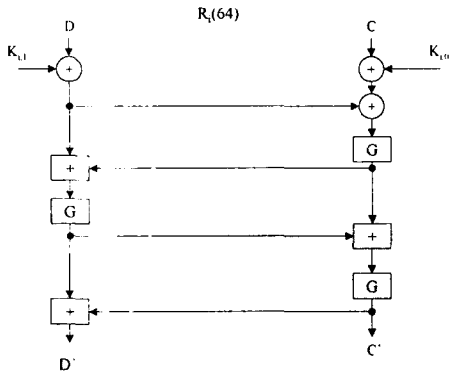


그림 2. F 함수의 구조
Fig.2. Structure of F function

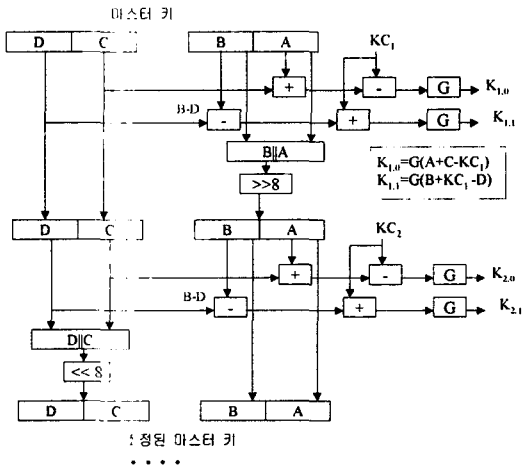


그림 3. 암호 동작에 대한 라운드 키 생성 알고리즘
Fig.3. Round key generation algorithm for encryption operation

III. SEED 암호 알고리즘에 대한 하드웨어 구현 기법

본 장에서는 SEED 암호 알고리즘을 효율적으로 구현하기 위해, 제안한 하드웨어 구현 기법을 기술한다.

1. 라운드의 다중 사이클 처리와 라운드 키의 사전 계산 기법

그림 1과 그림 2에 따르면 SEED 알고리즘의 각 라운드 수행 동작은 라운드 키 계산과 3쌍의 G함수와 모듈로 덧셈의 직렬 연산으로 구성된다. 이러한 동작을 하나의 클록동안 구현하게 되면 라운드 동작 구현에 많은 시간이 소요되고, 특히 하드웨어 공유가 불가능하므로 많은 하드웨어가 소요된다. 이러한 문제를 해결하기 위해 본 연구에서는 SEED의 각 라운드 동작을 그림 4와 같이 3개의 부분 라운드로 분할한 후, 각 부분 라운드를 단일 클록으로 구현하는 방식(1 round/3 clocks)을 사용하였다. 단, 라운드 키는 이전 라운드에 온라인 방식으로 미리 계산하는 기법을 사용한다. 이 방식의 경우 각 부분 라운드 내에 한 쌍의 G함수와 32 비트 모듈라 덧셈기가 존재하므로 하드웨어 공유가 극대화될 수 있다. 그리고 각 부분 라운드 구현 시 연결 관계를 단순화시키기 위해서, 각 부분 라운드의 출력을 엇갈리게 하는 형태를 채택하였다. 이러한 방식을 사용할 경우, 하드웨어 공유가 가능해지고, 라운드 키 계산에 따른 지연 시간 제거로 클록 주파수가 4배 이상 향상되는 장점이 있다. 그림 5는 제안한 기법에 대한 하드웨어 구현을 나타내며, 부분 라운드 1 과 부분 라운드 2의 중간 결과를 저장하기 위해 T 레지스터(T1, T0)를 사용한다. Sub_r_23 신호는 두 번째와 세 번째 부분 라운드에서, 중간 부분 라운드 결과를 저장하고 있는 T 레지스터 값을 입력으로 선택하는 동작을 수행한다.

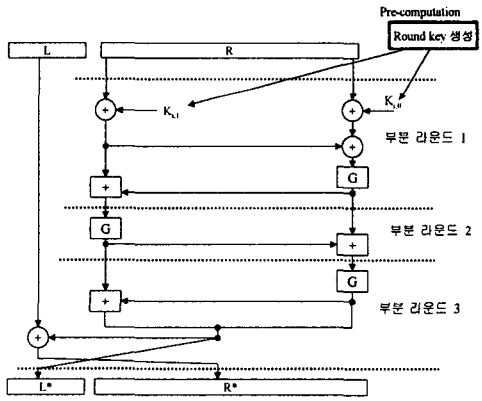


그림 4. 1 라운드를 3개의 부분 라운드로 분할하는 기법
Fig.4. Scheme which divides 1 round into three rounds

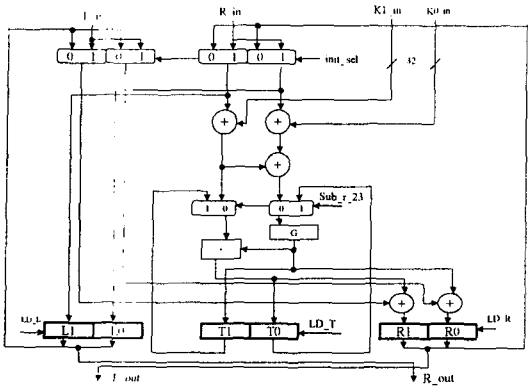


그림 5. 라운드 데이터패스
Fig.5. Round datapath

2. 라운드 키의 파이프라인 계산 기법

라운드 키 계산은 라운드 동작 진행과 병행하여 계산하는 온라인 계산 방식을 사용한다. 16개의 라운드 키를 암호 또는 복호 동작 전에 모두 계산하는 오프라인(off-line) 방식은 라운드 키 저장에 필요한 다수개의 레지스터와 라운드 키 계산에 따른 연산 시간 증가 문제로 본 연구에서는 배제하였다. 라운드 키의 계산은 이전 라운드에 파이프라인 방식으로 미리 계산하는 방식을 사용한다. 그림 4와 같이 라운드 동작을 3개의 클록 내에 수행하므로, 라운드 키의 계산에 활용할 수 있는 시간이 3개의 클록 길이라는 동작 특성을 활용하여, 라운드 키 계산을 병렬 계산 방식이 아닌 파이프라인 기법을 통해 그림 6과 같이 수행하였다. 파이프라인 계산 방식의 경우 $K_{i,1}$과 $K_{i,0}$를 동시에 계산하는 병렬 계산 방식에 비해 32 비트 파이프라인 레지스터와 캐리 보존 가산기(carry save adder)가 추가되었지만, G함수 1개와 모듈로 가산기가 3개 감소되어 하드웨어 감소 측면에서 약 2배 효과가 있다. 단, 모듈로 가산기는 고속 동작을 위해 32 비트 CLA(carry lookahead adder) 가산기 구조를 사용하였다.

3. 암호 라운드 생성 하드웨어를 사용한 복호 라운드 키 생성 기법

복호 동작과 암호 동작 시에 라운드 키가 적용되는 순서가 반대이므로, 온라인 방식으로 라운드 키를 생성할 경우, 암호 및 복호 동작에 대

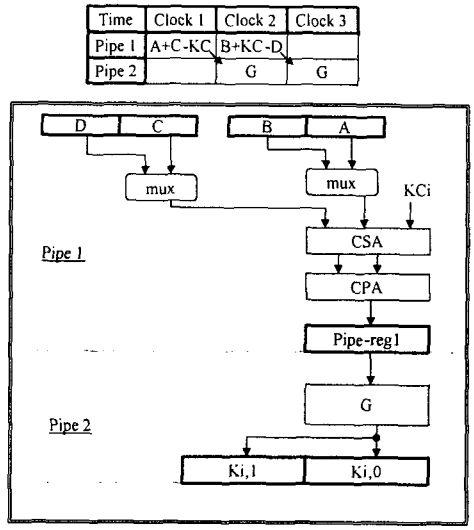


그림 6. 파이프라인 계산을 사용한 라운드 키의 사전 계산 방법
Fig.6. *Precomputation method of round key using pipelined processing*

한 라운드 키 생성을 위해 2개의 별도 하드웨어가 필요하다. 이러한 문제를 해결하기 위해 본 연구에서는 복호 동작의 라운드 생성 동작 분석을 통해, 기존 암호 동작에 대한 라운드 키 생성회로에 일부 기능을 추가하면, 복호 동작에 대한 라운드 키 생성이 가능함을 알 수 있었다. 그림 7은 복호 동작에 대한 라운드 키 생성 회로를 나타낸다. 그림에 보는 바와 같이 첫 번째 라운드 동작에 대한 라운드 키 생성 시 초기 정렬 동작이 추가로 필요하고, 나머지 동작은 암호 동작과 반대 방향으로 중간 결과를 순환 이동시킴에 의해 복호 동작에 대한 라운드 키 생성이 가능하다.

IV. SEED 암호 보조 프로세서의 하드웨어 설계

3장에서 제안한 기법을 바탕으로 암호 프로세서 설계 시 고려한 사항은 다음과 같다.

첫째, 암호 프로세서를 호스트 프로세서에 대한 보조 프로세서 형태로 설계한다.

둘째, 4가지 암호 동작 모드, 즉 ECB, CBC, CFB 및 OFB 방식을 모두 지원하도록 한다[7].

셋째, 입출력 동작과 암호 동작을 동시에 수행

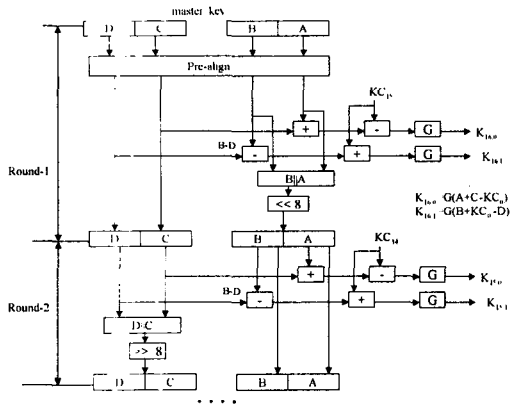


그림 7. 복호 동작에 대한 라운드 키 생성 기법
Fig.7. Round key generation scheme for decryption operation

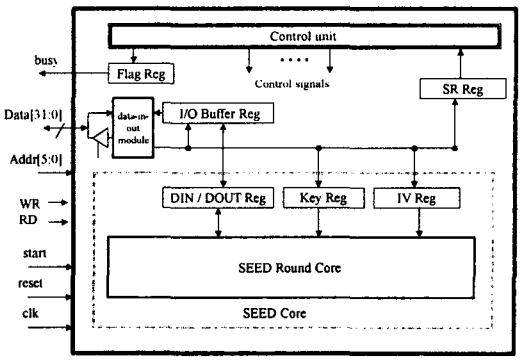


그림 8. SEED암호 보조 프로세서의 블록 구성도
Fig.8. Block diagram of SEED cryptographic coprocessor

하도록 하여, 입출력에 따른 시간 지연이 전체 연산 시간에 영향을 미치지 않도록 한다.

위의 3가지를 조건을 만족시키도록 설계된 암호 보조 프로세서의 블록도는 그림 8과 같다. 외부의 데이터 버스를 통해 마스터 키와 CBC, CFB, OFB 모드에서 사용되는 초기값(IV) 데이터를 입력한다. 단, 위의 세 번째 고려 사항에 따라 암호 동작과 외부 입출력 동작을 동시에 수행할 수 있도록 하여 입출력 시간에 따른 성능 저하를 방지하기 위해, 입·출력 버퍼 레지스터(I/O buffer Reg)와 내부 암호 코어의 입출력 레지스터 (DIN/DOUT Reg.)을 분리시켰다. 즉, 암호 동작 중에 입·출력 버퍼 레지스터를 통해 입출력 동작을 수행할 수 있도록 하였으며, 새로운 암호 동작에 대한 시작 신호 발생 시 DIN/DOUT 레지스터에 담긴 암·복호 결과와 I/O 버퍼 레지스터에 저장된 외부 입력 데이터를 서로 교환하는 동작을 수행한다. SR(status register) 레지스터는 동작 모드, 외부 인터페이스 방안, CFB와 OFB 모드 시 암· 복호 단위를 지시하는 필드 등을 갖고 있다

.SEED 코어는 그림 9에 보이는 바와 같이 입출력 데이터, 마스터 키, 초기 값을 담고 있는 레지스터, 배럴 시프터(L_shift_1, L_shift_2)와 SEED 라운드 코어로 구성된다. DIN_OUT 레지스터는 암·복호 동작을 수행할 128 비트 입력 데이터를 담는 기능과 암·복호 동작의 결과 값을 저장하는 역할을 함께 수행한다

배럴 시프터 회로는 128 비트 좌측 이동 또는 j 비트 좌측 이동 동작을 통해 4가지 동작 모드에 대한 초기 값과 중간 결과의 좌측 이동 동작 또는 갱신 동작을 지원한다. 여기서 j 비트는 CFB 또는 OFB 모드에서 암·복호 단위를 나타내는 값으로 SR 레지스터에 그 값이 정의된다. SEED 라운드 코어는 16 라운드 동작을 구현하는 라운드 데이타패스와 라운드 키를 생성하는 키 생성부 부로 나뉘어 진다. SEED 라운드 코어 설계 시 귀환 출력이 사용되는 CBC, CFB 및 OFB 모드의 지원에 따라 파이프라인 구조는 불가능하므로, 1 라운드 동작에 필요한 하드웨어를 갖추고, 이를 16 라운드 반복 사용하여 암·복호 동작을 구현하는 구조를 채택하였다. 제어 회로는 외부 호스트 프로세서가 제공하는 입출력 명령을 해독하여, 필요한 입출력 동작을 수행하는 입출력 제어부와 호스트 프로세서가 제공하는 시작신호를 받아서 SEED 암호 알고리즘을 구현하는 코어 제어부로 나뉜다. SEED 암호 보조 프로세서가 외부 호스트와 별도의 클록으로 동작하므로, 호스트가 제공하는 명령 중 시작신호는 SEED 암호 프로세서의 동기 클록과 동기화가 필요하다. 따라서 외부에서 제공되는 시작 신호는 보조 프로세서 내부에 포함된 동기화 회로를 거쳐 코어 제어부에 제공되도록 함에 의해서, 외부 명령이 비실행 또는 다중 실행되는 것을 방지하도록 하였다. 코어 제어부는 FSM(finite state machine)부, 카운터부, 제어신호 생성부로 구성된다. 코어 제어 회로 설계는 데이터패스의 동작

흐름을 ASM(Algorithmic State Machine) 차트로 표현한 후, 이를 제어 회로로 변환하는 방법을 사용하였다.

V. CPLD칩 구현을 통한 설계 검증 및 성능 분석

그림 10은 본 연구에서 설계한 암호 프로세서의 설계 및 검증 흐름도를 나타낸다. 먼저 SEED 암호 알고리즘을 C 언어로 모델링한 후, 이를 Verilog HDL[8] 언어로 변환하여 2가지 동작이 일치하는지 확인하는 과정을 사용하였다. 검증 과정에 ECB 모드의 경우 정보 보호 센터 보고서[4]에서 명시한 테스트 벡터를 올바로 만족함을 확인하였다. 그리고 나서 설계한 회로는 Max-Plus II 소프트웨어[9]를 사용하여 합성된 회로를 검증 한 후, HBE-DTK -200K 보드에 장착된 EPF10K100GC503-3 디바이스에 다운로드 하였다. 설계된 칩이 암·복호 동작을 올바로 수행하는지 검증하기 위해 그림 11과 같은 테스트환경을 구현하여 PC상에서 ISA 버스 인터페이스를 통해 일반 문서 파일을 암호 칩에 전달한 후, 암호동작을 지시하여 암호화된 파일을 생성하여 메모리에 저장하였다. 그리고 나서 암호화된 파일을 다시 암호 보조프로세서 칩에 보내 복호 동작을 하여 원래 문서 파일이 만들어지는지 확인하였다.

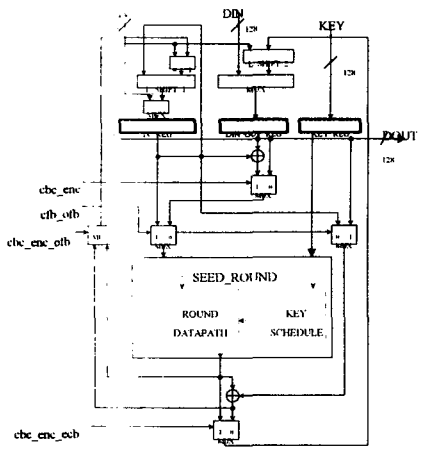


그림 9. SEED 코어의 블록도
Fig. 9. Block diagram of SEED core

CPLD 칩으로 합성한 결과 암호 보조 프로세서 칩의 최대 동작 주파수는 약 18Mhz이었으며, 총 게이트 수는 약 29,300이었다. 그림 12는 설계한 암호 프로세서의 ECB 모드에 대한 동작 타이밍을 나타낸다. 라운드 1 이전에 라운드 키의 사전 계산을 위한 3개의 클록을 포함해 총 51개의 클록으로 구성된다. 단, 16 라운드의 동작 수행 후에 SEED 라운드 최종 결과가 그림 5의 {L,R} 레지스터에 위치가 반대로 담기게 되므로, 외부 호스트로 전달을 하기 위해서는 {L,R} 레지스터 값을 엇갈리게 DIN_DOUT 레지스터로 이동시키는 추가의 사이클이 필요하다. 따라서 실제 ECB 동작 구현에 소요된 사이클 수는 (48+1)+3=52이다. 반면 CBC, CFB, OFB 모드의 경우, 이러한 추가의 동작 사이클 과정에 XOR 게이트와 L_shift 회로를 활용하여, IV 레지스터와 DIN_DOUT 레지스터 값을 적절히 갱신한다. 따라서 ECB 모드의 경우 암·복호율은 식(2)에 따라 18Mhz 클록 조건에서 약 44 Mbps 이다. 반면 CFB와 OFB 모드의 경우 암·복호율은 식(3)과 같이 정의된다.

$$ECB\text{와 }CBC\text{모드에 대한 암·복호율} = (128 \div 52) \times f \quad (2)$$

여기서 f 주파수

$$CFB\text{와 }OFB\text{모드에 대한 암·복호율} = \frac{128}{49 \times \frac{128}{f} + 3} \times f \quad (3)$$

여기서 f는 암·복호 단위(비트)

표 1은 암호 프로세서의 특성을 나타낸다. S 박스에 대한 구현 방안으로 식의 논리 최소화를 이용하는 방식은, 테이블 룩업 방식에 비해 회로 개선 효과가 크지 않고 배선을 복잡하게 하는 문제가 있어서, 본 연구에서는 이진 룩업 테이블 형태로 구현하는 방식을 사용하였다. 표 2는 본 SEED 암호 프로세서와 여러 가지 SEED 암호 프로세서의 구현 방안을 아키텍처 측면에서 비교한 결과이다. 단, 최종 {L, R} 레지스터 값의 DIN_DOUT 레지스터로의 이동 동작은 하드웨어 구현 측면의 동작 특징이므로 계산식에 포함시키지 않았다. 여기서 1 round/4 clocks 방식은 라운드 키를 사전에 계산하지 않고, 라운드 키 계산을 첫 번째 부분 라운드로 할당한 방식을 나타낸다. 표 2에 따르면 본 암호 프로세서는 면적과 속도 측면에서 효율적임을 알 수 있다. 즉 1 round/clock 방식에 비해 소요되는 클록 수가 많지만, 라운드 키의 사전 계산 특성에 의해 동작 주파수가 4배정도 빠르므로, 실제 전체 연산은

1.25배 빠르게 수행된다. 그리고 면적 측면에서 비교할 때 약 3배정도 면적 개선 효과가 있다. 표 3은 기존 DES 프로세서와 구현 결과를 비교한 결과이다. 본 연구의 SEED 프로세서는 전용 반도체 칩이 아닌 게이트당 지연시간이 3 ns인 저속 CPLD 칩으로 구현했으므로, 기존 프로세서에 비해 낮은 암·복호율을 갖는다. 그러나 고속 CPLD 칩을 사용하여 40 Mhz의 동작 주파수를 갖게 되면 약 96Mbps의 암·복호율을 얻을 수 있으므로, 본 연구와 유사하게 FPGA 칩을 사용한 FPGA DES[12] 칩에 비해 높은 암·복호율을 가짐을 알 수 있다. 그리고 본 연구의 프로세서를 전용 반도체 칩으로 구성하면 100 Mhz 이상의 동작 주파수를 얻을 수 있으므로, 식(2)을 적용할 경우 수 백 Mbps의 암·복호율을 얻을 수 있다. 이러한 특성을 고려할 때 본 연구의 SEED 암호 프로세서는 SEED 암호 알고리즘을 보안 모듈로 사용하는 시스템에 효율적으로 장착될 수 있을 것으로 판단된다.

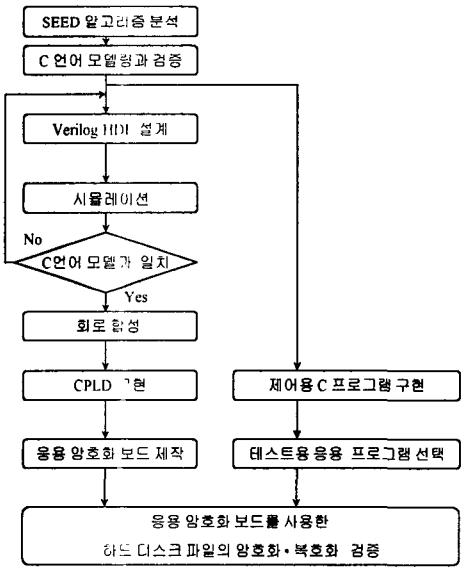
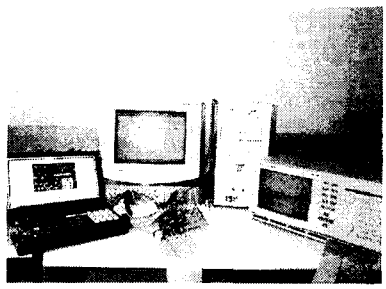
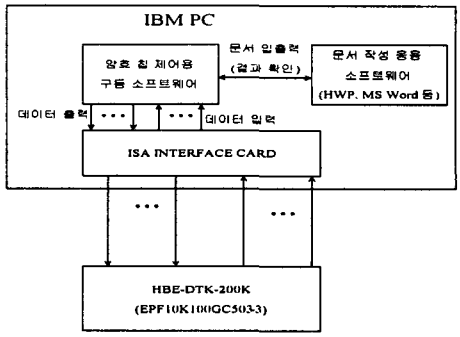


그림 10. 암호 프로세서의 설계 및 검증 흐름도
Fig.10. Design and verification flow for cryptographic coprocessor



(a)



(b)

그림 11. SEED 프로세서의 검증 환경
(a) 검증 환경 사진
(b) 검증 환경 구성도
Fig.11. Test environment of SEED processor
(a) photography of test environment
(b) block diagram of test environment

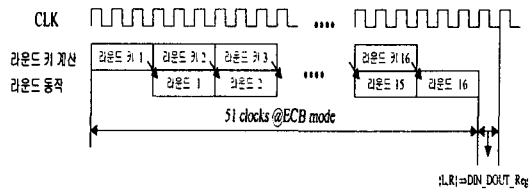


그림 12. ECB 모드에 대한 타이밍 분석
Fig.12. Timing analysis for ECB mode

표 1. 암호 보조 프로세서의 특징
Table 1. Characteristics of cryptographic coprocessor

지원 동작 모드	ECB, CBC, CFB, OFB
라운드당 함수수	3
게이트 수	약 29,300
동작 주파수	18 Mhz
라운드 키 계산 방식	온라인 파이프라인 사전 계산 방식
암·복호율	44 Mbps @ ECB, CBC mode 44 Mbps @ CFB, OFB mode (J=128)
입출력 방식	background input/output
외부 인터페이스	8-bit/16-bit/32-bit
암·복호화 단위(J) (@CFB, OFB)	8, 16, 32, 64, 128 비트

표 2. SEED 암호 프로세서 구현 방식 비교
Table 2. Comparisons of various implementations for SEED algorithm

방 식	G함수 수	S박스 수	모듈라 가산기 수	성능(클럭수) @ECB	클럭 주기	연산 시간
1 round/ 1 clock	5	20	7	16	T	16T
1 round/ 4 clocks	3	12	4	4×16=64	T/4	16T
본 연구	2	8	2	3×16 + 3=51	T/4	12.75T

VI. 결 론

본 논문에서는 면적 측면과 속도 측면에서 효율적인 SEED 암호 보조 프로세서를 CPLD 칩으로 구현한 후, 응용 암호화 보드를 활용하여 설계된 칩이 올바로 동작함을 확인하였다. 설계한 SEED 암호 보조 프로세서는 4가지 동작 모드를 모두 지원하며, 1 라운드 동작을 3개의 부분 라운드로 분할 처리, 라운드 키의 온라인 파이프라인 계산 기법을 통해 하드웨어 공유를 극대화시켜, 기존의 단일 클록에 하나의 라운드를 구현하는 방식에 비해 약 1/3의 하드웨어와 1.25배 개선된 연산 시간 특성을 얻을 수 있다. 설계한 SEED 암호 보조 프로세서는 약 29,300개의 게이트로 구성되며, 알테라 EPF10K100GC503-3 상에서 약 18 Mhz의 동작 주파수를 가지며, ECB 모드에서 약 44 Mbps의 암·복호율를 얻을 수 있었다. 설계된 암호 보조프로세서를 CPLD 칩이 아닌 반도체 공정을 사용하여 집적회로로 구현할 경우 수백 Mbps이상의 성능을 얻을 수 있을 것으로 예상되므로, 본 논문의 암호 보조 프로세서는 SEED 암호 알고리즘이 적용되는 네트웍, 전자 상거래 시스템 등의 보안 모듈로 사용될 수 있을 것으로 판단된다.

표 3. 여러 가지 대칭키 암호 프로세서의 구조 비교
Table 3. Architectural comparisons for various symmetric key processors

프로세서	알고리즘	동작 모드	성 능	동작 주파수	구현 형태
GaAs DES[10]	DES	ECB, CBC	1 Gbps	350 Mhz	GaAs
PCC 101[11]	DES, TDES	ECB, CBC, CFB, OFB	132 Mbps	33 Mhz	CMOS custom IC
FPGA DES[12]	DES	ECB	88 Mbps	40 Mhz	FPGA
본 연구	SEED	ECB, CBC, CFB, OFB	44 Mbps (96 Mbps)	18 Mhz (40Mhz)	CPLD

접수일자 : 2000. 9. 14. 수정완료 : 2000. 10. 25.
이 논문은 한국 전자 지불 연구원의 연구비에 의해 수행 되었음

참고문헌

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

[2] Jenes-Peter Kaps, *High Speed FPGA Architecture for the Data Encryption Standard*, Master Thesis, May, 1998.

[3] Kris Gaj, "Comparison of the hardware performance of AES candidates using reconfigurable hardware", Third AES candidate Conference, April, 2000.

[4] 한국 정보 보호 센터, *128 비트 블록 암호 알고리즘(SEED) 개발 및 분석 보고서*, 1998. 12.

[5] 신 종 호, 강 준 우, "SEED 블록 암호 알고리즘의 단일 칩 연구", 2000년도 대한 전자 공학회 하계 종합 학술 발표 대회 논

문집, pp. 165-168, 2000.

[6] 전 신 우, 정 용진, "128 비트 SEED 암호 알고리즘의 고속 처리를 위한 하드웨어 구현", 2000년도 한국 통신 학회, 하계 학술 발표대회 논문집, pp.1307-1310, 2000

[7] National Bureau of Standards, *DES Modes of Operation, Federal Information Processing Standards Publication FIPS PUB 81*, December 1980.

[8] IDEC 반도체 설계 교육센터, *Cadence Tool 교육 강좌 자료*, 1999.5.

[9] 이승호, Altera Max-Plus II를 사용한 디지털 시스템 설계, 북두출판사, 1999

[10] Hans Eberle, "A High-speed DES Implementation for Network Applications", CRYPTO '92, pp.521-539, 1993, Springer-Verlag.

[11] PIJNENBURG, PCC101 DES/3DES Data Encryption Device, http://www.pijnenburg.nl/pcc101.htm

[12] Jenes-Peter Kaps, "High Speed FPGA Architecture for the Data Encryption Standard", Master Thesis, WPI. 1998

최병윤(Byeong-Yoon Choi)
正會員
1985년 연세대학교 전자공학과 (공학사)
1987년 연세대학교 전자공학과(공학석사)
1992년 : 연세대학교 전자공학과(공학박사)
1997년~1998년 : 일리노이 주립대 연구 교수
1993년~현재 : 동의대학교 부교수
관심 분야 RISC 마이크로프로세서 설계, 컴퓨터 연산 회로 설계, 통신 및 암호 회로의 VLSI 설계

김진일(Jin-IL Kim)
正會員
1980년 경희대학교 전자공학과(공학사)
1982년 경희대학교 전자공학과(공학 석사)
1994년 서강대학교 전자계산학과(공학박사)
1982년-1983년 미국 Bon Scours 시스템, 맥도널다글라스 우주 항공 회사 등 연구원
1996년-1997년 미국 Purdue Univ. 전기 및 컴퓨터공학부 연구교수
1988년-현재 동의대학교 컴퓨터공학과 부교수
위성탐사 영상처리 연구회(한국과학재단) 회장
관심분야 패턴인식, 퍼지 논리, 위성영상 응용, 컴퓨터보안 인공지능 및 컴퓨터응용분야