情報保護學會論文誌
第 22 卷 第 2 號, 2012. 4

고속 연산이 가능한 파이프라인 구조의 SATA HDD 암호화용 FPGA 설계 및 구현

구 본 석,[1†] 임 정 석[1], 김 춘 수[1], 윤 이 중[1], 이 상 진[2‡]
[1]국가보안기술연구소, [2]고려대학교 정보보호대학원

High-Speed FPGA Implementation of SATA HDD Encryption Device based on Pipelined Architecture

Bonseok Koo,[1†] Jeongseok Lim[1], Choonsoo Kim[1], E joong Yoon[1], Sangjin Lee[2‡]
[1]National Security Research Institute,
[2]Graduate School of Information Security, Korea University

요 약

본 논문에서는 단일 FPGA를 이용한 SATA 하드디스크용 Full Disk Encryption 연산기를 제안하고, 해당 연산기를 FPGA기반 테스트용 보드에 구현하여 실험한 결과를 제시한다. 제안하는 연산기는 크게 디스크 암호화 표준 알고리즘인 IEEE P1619 (XTS-AES) 연산블록과, SATA Host (PC)와 Device (HDD)간의 정합 기능을 담당하는 SATA 인터페이스 블록으로 구성된다. 고속 암복호 연산기능을 담당하는 XTS-AES 암호 연산블록은 암복호 기능추가로 인한 속도저하를 최소화하기 위해 매 4 클록 사이클마다 1 블록 암호화를 처리하도록 4단 파이프라이닝 구조로 설계하여 최대 4.8Gbps의 암복호 성능을 가진다. 또한 전체 연산기를 Xilinx사의 ML507 FPGA 개발보드에 구현하여, Windows XP 32비트 환경에서 SATA II 하드디스크(7200rpm)에 대해 암호화 장치없이 직접 연결했을 때와 동등한 속도인 최대 140MB/sec 읽기/쓰기 성능을 나타내었다. 따라서, 제안하는 연산기는 단일 FPGA를 이용하여 속도저하 없는 Full Disk Encryption 기능 구현이 가능함을 확인하였다.

ABSTRACT

This paper addresses a Full Disk Encryption hardware processor for SATA HDD in a single FPGA design, and shows its experimental result using an FPGA board. The proposed processor mainly consists of two blocks: the first block processes XTS-AES block cipher which is the IEEE P1619 standard of storage media encryption and the second block executes the interface between SATA Host (PC) and Device (HDD). To minimize the performance degradation, we designed the XTS-AES block with the 4-stage pipelined structure which can process a 128-bit block per 4 clock cycles and has 4.8Gbps (max) performance. Also, we implemented the proposed design with Xilinx ML507 FPGA board and our experiment showed 140MB/sec read/write speed in Windows XP 32-bit and a SATA II HDD. This performance is almost equivalent with the speed of the direct SATA connection without FDE devices, hence our proposed processor is very suitable for SATA HDD Full Disk Encryption environments.

Keywords: SATA, HDD, FDE, FPGA, Encryption, Pipelining, Block Cipher, XTS-AES

접수일(2011년 8월 17일), 수정일(2011년 12월 14일),
게재확정일(2011년 12월 30일)

† 주저자, bskoo@ensec.re.kr
‡ 교신저자, sangjin@korea.ac.kr

I. 서 론

FDE (Full Disk Encryption)[1]은 파일단위 암호화 방식과 달리 하드디스크에 저장되는 모든 데이터를 암호화하여, 저장 데이터에 대한 비인가적인 접근을 차단하는 기술이다. FDE 구현기술은 크게 소프트웨어를 이용한 암호화 방식과 하드웨어 기반 암호화 방식으로 구현할 수 있는데, 소프트웨어 방식은 하드웨어 구현 방식에 비해 시스템 성능을 저하시키고, 암호화용 키 정보가 CPU 주변 메모리 등 안전하지 못한 환경에서 관리되는 단점이 있다[2]. 아울러, 최근에는 전원이 Off된 상태에서 컴퓨터 내부 메모리로부터 중요 정보를 추출해내는 'Cold Boot Attack'[3]과 같은 공격기술들이 꾸준히 발표되고 있으므로, 일정 수준 이상의 안전성이 요구되는 환경에서는 하드웨어 기반의 FDE 구현 방식이 바람직하다.

한편, 하드디스크와 같은 대용량 저장매체에 대한 암호화에는 SISWG (Security in Storage Working Group)에서 제안하여 IEEE 1619-2007 표준으로 채택된 XTS-AES 알고리즘이 가장 널리 사용되고 있다[7]. 또한, 최근에 NIST (National Institute of Standards and Technology)는 Special Publication 800-38E 문서를 발표하여 암호모듈 구현에 XTS- AES 알고리즘의 사용을 권장하고 있다[8].

본 논문에서는 최근 가장 많이 사용되는 하드디스크용 인터페이스인 SATA 인터페이스를 가지는 FDE 암호화장치에 대한 설계 및 구현내용을 제시한다. 본 논문에서 제시하는 FDE 암호화 장치는 단일 FPGA로 구현되며 크게 XTS-AES 연산기와 SATA 인터페이스 블록으로 구성되는 SATA 컨트롤러 기반의 하드웨어 방식 암호화 장치이다. 실제적으로 SATA 인터페이스 블록은 IntelliProp 사의 SATA Bridge 반도체 IP[17] (Intellectual Property)를 이용하여 호스트 PC 및 디바이스 하드디스크간의 정합 기능을 담당하도록 구현하였다. 또한, XTS-AES 연산기는 4단 파이프라이닝 구조로 설계하여 암호화로 인한 성능저하 없이 SATA Bridge IP에서 제공하는 최대 송수신 성능인 2.4Gbps를 만족하도록 설계하였다. 아울러, 최종 설계한 FDE 암호화장치를 Xilinx사의 ML507 FPGA 개발키트 보드[18]에 실제 구현하고, 하드디스크 벤치마크 프로그램을 통해 하드디스크 읽기/쓰기 성능을 측정한 결과를 제시한다.

본 논문은 다음과 같이 구성된다. 2장에서는 하드웨어 기반의 FDE 구현에 관해 발표된 기존 연구들을 살펴보고, 3장과 4장에서는 XTS-AES 알고리즘과 SATA Host 및 Device와의 인터페이스를 담당하는 IntelliProp사의 SATA Bridge 반도체 IP의 개요에 대해 각각 서술하며, 5장에서는 SATA Bridge IP와 연동하여 XTS-AES 암복호 동작을 수행하는 암복호 연산블록의 아키텍처 및 구현방법에 대해 상세히 설명하고, 6장은 제안하는 연산기를 실제 FPGA로 구현한 결과에 대해 기존 결과들을 비교하여 서술하며, 마지막 7장에서는 결론을 맺는다.

II. 관련분야의 연구

하드웨어 기반의 FDE 구현 방식은 하드디스크 내부에서 암호화를 처리하는 디스크 내부 암호화 방식과, 메인보드와 하드디스크 사이의 컨트롤러에서 암호화를 처리하는 컨트롤러 기반 암호화 방식으로 구분된다. 최근에는 주요 하드디스크 제조회사들이 디스크 내부 암호화 방식의 HDD (Hard Disk Drive) 및 SSD (Solid-State Drive) 제품들을 출시하고 있다[4-6]. 이러한 디스크 내부 암호화 방식은 하드디스크에 내장된 전용칩에서 암복호 동작을 수행하므로 부가적인 장치없이 모든 저장 데이터를 고속으로 암호화할 수 있는 장점이 있다. 하지만, 이러한 기능을 제공하는 제품들은 특정 하드디스크 모델로 한정되며, 암호알고리즘, 접근제어, 키관리 등의 정보보호방식 또한 하드디스크 제조사들에서 제공하는 솔루션을 그대로 사용해야 한다는 한계를 가진다.

한편 XTS-AES 알고리즘은 AES 블록암호 알고리즘을 기반으로 XTS (XEX-based Tweaked codebook mode with ciphertext Stealing) 운용모드를 사용하므로, 기존에 논문으로 발표된 XTS-AES 연산기들은 핵심 연산블록인 AES 암호화 연산기 자체의 최적화에 대한 내용보다는 XTS 운용모드를 일반적인 구조의 AES 암호화 연산기 주변에 부가적으로 추가하는 방식으로 구현하여 그 결과를 제시하였다[9-11]. 따라서, 연산기의 아키텍처 설계 측면에서 기존의 AES 연산기 설계방식[9-16]에서 진보된 기술을 적용한 결과를 찾아보기는 어려운 실정이다.

실제로 하드웨어 기반 FDE 시스템을 구현하기 위해서는 XTS-AES 연산기 코어 뿐만 아니라 호스트 PC와 하드디스크 사이의 ATA (Advanced Technology Attachment) 또는 Serial ATA (이하 SATA)

와 같은 인터페이스와의 정합 블록을 필수적으로 구현해야 하며, 아울러 이러한 인터페이스에서 사용하는 통신 프로토콜에 맞춰 암호연산기의 동작을 제어할 수 있는 컨트롤 블록의 설계 기술이 요구된다. 하지만, 현재까지 논문으로 발표되었거나 반도체 디자인 회사에서 제품으로 제공하는 XTS-AES 연산기[9-16]는 대부분 XTS-AES 알고리즘 자체만을 구현한 암호연산기이다. 따라서, 외부 인터페이스와의 정합 기능과 인터페이스와 연동하는 동작 컨트롤에 대한 구현 부분은 설계자가 자체적으로 구현해야 하는 부담이 있을 뿐만 아니라, 해당 연산기를 실제 하드웨어 보드 상에 구현했을 때의 실제적인 성능을 예측하기 어렵다.

III. XTS-AES 알고리즘

XTS는 P. Rogaway가 제안한 XEX (Xor-Encrypt -Xor) 운용모드[19]를 기반으로 하여 CTS (Cipher- Text Stealing) 특징을 추가한 블록암호 알고리즘의 운용모드이며, 128-비트 한 블록 데이터에 대한 XTS -AES 암호화 연산은 다음과 같이 표현된다.

$$C \leftarrow XTS-AES-blockEnc(Key, P, i, j)$$

- Key : 256비트 또는 512비트 XTS-AES 키
- P : 128-비트 한 블록의 평문 데이터
- i : 128-비트 tweak 값
- j : 입력 데이터를 128-비트 블록 단위로 정렬했을 때의 순서 번호
- C : 128-비트 한 블록의 평문 데이터에 대한 128-비트 암호문

여기서, XTS-AES 연산에 사용되는 키 정보 (Key)는 동일한 사이즈의 Key_1과 Key_2를 연접한 형태이다 (즉, $Key = Key_1 \,||\, Key_2$).

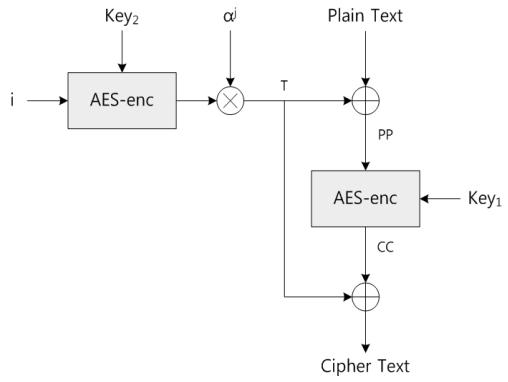
이때, XTS-AES 암호화 ($XTS\text{-}AES\text{-}blockEnc(\ Key,\ P,\ i,\ j)$)는 다음과 같은 과정으로 계산된다.

1) $T \leftarrow AES-enc(Key_2, i) \otimes \alpha^j$
2) $PP \leftarrow P \oplus T$
3) $CC \leftarrow AES-enc(Key_1, PP)$
4) $C \leftarrow CC \oplus T$

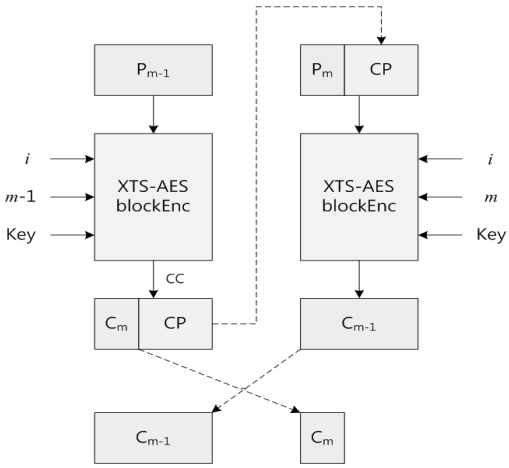
여기서, AES-enc(K, P)는 FIPS-197 표준[20]에 따라 평문 P와 키 입력 K에 대한 AES 암호화 연산을 나타낸다. 또한, 단계 1에서 α^j는 $GF(2^{128})$ 상의 원시 원소 α를 j번 곱해서 계산되는 값이다.

[그림 1]에는 XTS-AES 암호화 과정을 나타내었다. 한편 XTS-AES 복호화 과정은 단계 3에서의 AES-enc(Key_1, PP) 대신에 AES-dec(Key_1, PP)로 대체되는 점 이외에 다른 과정은 XTS-AES 암호화 과정과 동일하다.

[그림 1]의 암호화 과정은 입력 데이터의 길이가 128 비트의 배수인 경우에만 정상적인 암호화 동작이 가능하므로, XTS-AES는 입력 데이터의 길이가 128-비트 블록의 배수가 아닌 경우에는 [그림 2]와 같이 CTS 방식으로 동작하여 입력 평문 데이터와 출력 암호문 데이터의 길이가 항상 동일하게 유지되는 특징을 가진다.



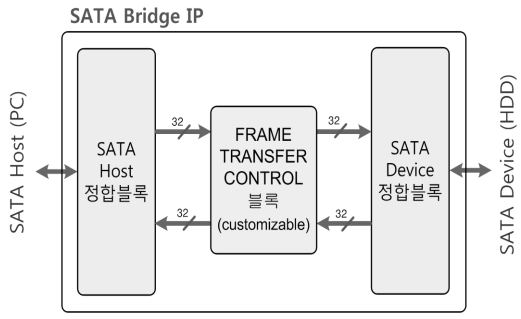
[그림 1] XTS-AES 암호화 과정



[그림 2] 128-비트 길이 미만인 블록의 XTS-AES 암호화 과정

IV. SATA Bridge IP의 개요

SATA 컨트롤러 기반의 FDE 장치를 구현하기 위해서는 SATA 호스트(PC)와의 정합과 디바이스(하드디스크)와의 정합 기능이 모두 필요한데, SATA 인터페이스용 반도체 디자인 IP 제품들[17,21,22]은 다양하지만, 이러한 두 가지 기능을 모두 지원하는 단일 제품은 IntelliProp 사의 SATA Bridge IP가 거의 유일하다. IntelliProp 사의 SATA Bridge IP는 SATA I 및 SATA II 규격[23]을 지원하며, [그림 3]에는 그 내부 구성도를 나타내었다. 그림에서 'SATA Host 정합블록'은 호스트(PC)와의 인터페이스를 담당하고, 'SATA Device 정합블록'은 디바이스(하드디스크)와의 인터페이스를 각각 담당하며, 'FRAME_ TRANSFER_CONTROL 블록'은 'SATA Host 정합블록'과 'SATA Device 정합블록' 간에 전달되는 모든 FRAME 데이터를 동기식 FIFO 인터페이스 방식으로 송수신하는 기능을 담당한다. 'FRAME_ TRANSFER_CONTROL 블록'은 설계자의 목적에 맞게 임의 변경이 가능한 블록이며, IntelliProp 사에서는 이 블록에 대한 간단한 레퍼런스 코드만을 제공한다. 따라서, 하드디스크 암호화 장치를 구현하기 위해서는, 이 블록을 수정하여 호스트와 디바이스 간에 송수신되는 SATA Frame 정보를 파싱하고 데이터 읽기/쓰기 동작과 관련된 FRAME 정보가 전송되는 경우, SATA 규격에 맞게 송수신 데이터를 암호화 및 복호화하는 동작이 가능하도록 구현할 수 있다. 한편, 'FRAME_TRANSFER_CONTROL' 블록의 FIFO 인터페이스는 32-비트 단위 데이터를 75MHz 클록 속도로 매 클록 입출력이 가능하므로, 최대 2.4Gbps (= 32bit x 75MHz)의 성능을 가진다.



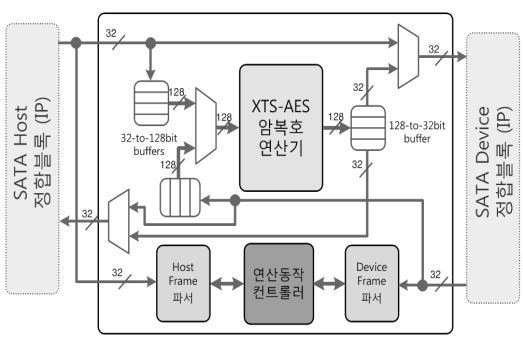
[그림 3] SATA Bridge IP의 내부 구성도

V. 암복호 연산블록의 설계

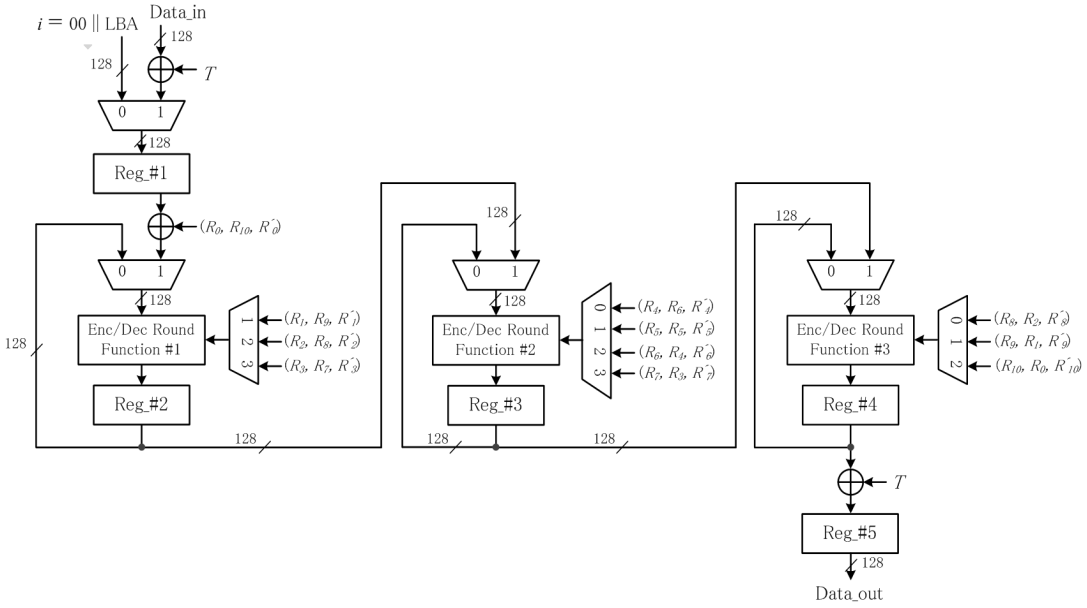
암복호 연산블록은 앞서 설명한 SATA Bridge IP의 'FRAME_TRANSFER_CONTROL' 블록에 구현되며, 내부 구성도를 [그림 4]에 나타내었다. 그림에서 보듯이 암복호 연산블록은 크게 'Host Frame 파서', 'Device Frame 파서', '연산동작 컨트롤러', 'XTS -AES 암복호 연산기' 로 구성되고, 각 구성 블록이 담당하는 기능은 다음과 같다.

- Host Frame 파서: SATA Host가 Device측으로 전달하는 모든 Frame 정보를 파싱하여, 데이터 읽기 및 쓰기 동작과 관련된 Frame이 전달되는 경우, 암복호 동작 제어에 필요한 정보를 '연산동작 컨트롤러'로 전달하는 역할을 담당
- Device Frame 파서: SATA Device가 Host측으로 전달하는 모든 Frame 정보를 파싱하여, 암복호 동작 제어에 필요한 정보를 '연산동작 컨트롤러'로 전달하는 역할을 담당
- 연산동작 컨트롤러: Host Frame 파서 및 Device Frame 파서로부터 전달되는 정보를 이용하여, XTS-AES 암복호 연산 및 SATA Frame 전달 동작 전반을 제어하는 역할을 담당, 유한상태기계 (Finite State Machine, FSM) 방식으로 구현됨
- XTS-AES 암복호 연산기: 128-비트 키 길이, 128-비트 입/출력 데이터 방식으로 XTS-AES 암복호 연산을 처리하는 블록

한편, Host Frame 파서 및 Device Frame 파서는 Frame 단위의 SATA Transport Layer 통신 프로토콜[24]에 맞춰 Frame 정보를 파싱하도록 설계하였으며, 그 세부적인 설계 내용은 본 고에서는 논외로 한다.



[그림 4] 암복호 연산블록의 구성도



〔그림 5〕 제안하는 XTS-AES 암복호 연산기의 하드웨어 구조

5.1 XTS-AES 암복호 연산기의 구조 및 동작 타이밍

암복호 연산블록은 SATA Bridge IP의 'FRAME_TRANSFER_CONTROL' 기능 블록의 위치에 구현되므로, 최대 2.5Gbps(= 32bit x 75MHz) 성능으로 양방향 데이터 송수신이 가능한 구조이다. 한편, SATA는 Command/Response 방식으로 동작하므로, Host → Device와 Device → Host 방향의 데이터 전달이 동시에 발생하지 않으므로, 암호화와 복호화 동작이 동시에 가능하도록 연산기를 설계할 필요는 없다. 따라서, 본 논문에서 제안하는 XTS-AES 연산기는 암호화 또는 복호화 동작을 선택적으로 수행할 수 있으며, 암호화 및 복호화에 의한 성능저하가 발생하지 않도록 128-비트 한 블록을 4 클록 사이클에 처리할 수 있는 파이프라이닝 방식으로 설계하였다.

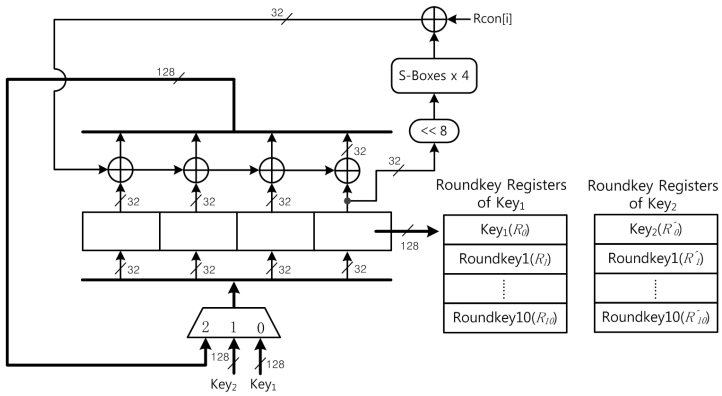
〔그림 5〕에서는 제안하는 XTS-AES 암복호 연산기의 하드웨어 구조를 나타내었다. 제안하는 연산기는 총 3개의 128-비트 블록을 한꺼번에 처리할 수 있는 파이프라이닝 구조로 설계하였으며, 4 클록 사이클마다 한 블록에 대한 암호화 및 복호화를 처리할 수 있는 특징을 가진다. 그림에서 $\{R'_0, \ldots, R'_{10}\}$은 마스터 키 Key_1으로부터 생성된 라운드 키이며, $\{R_0, \ldots, R_{10}\}$은 Key_2부터 생성된 라운드 키를 나타낸다. 한편 LBA(Logical Block Address)는 SATA 프로토콜을 통해 전달되는 48-비트 섹터 주소 정보이며, 128-비트 tweak값 i는 LBA 정보에 나머지 상위비트를 0으로 패딩시킨 00 || LBA이 사용된다. 또한 (R_x, R_y, R_z)는 평문에 대한 암호화 동작 시에는 Rx, 암호문에 대한 복호화 동작 시에는 R_y, 그리고 〔그림 1〕의 T 값을 계산하기 위한 $AES-enc(Key_2, i)$ 동작 시에는 R_z가 선택됨을 나타낸다.

한편, SATA 통신 상에서 입출력 데이터의 최소 단위인 섹터는 항상 512 바이트 또는 그 배수의 크기를 가지므로, 제안하는 연산기에서는 한 블록(=32바이트)의 배수가 아닌 입출력 데이트에 대한 암복호 기능인 CTS(〔그림 2〕)의 구현은 생략하였다.

〔표 1〕에는 제안하는 XTS-AES 연산기의 파이프라이닝 동작 방식을 설명하기 위해, 암복호 동작 타이밍을 중간 결과값이 저장되는 레지스터(〔그림 5〕의 Reg_#1 ~ Reg_#5)와 클록 사이클을 기준으로 나타내었다. 〔표 1에〕 나타낸 바와 같이 제안하는 연산기는 한 블록에 대한 암호화 또는 복호화를 수행하는

〔표 1〕 제안하는 XTS-AES 연산기의 동작 타이밍

사이클	1	2	3	4	5	6	7	8	9	10	11	12
1st 블록	#1	#2	#2	#2	#3	#3	#3	#3	#4	#4	#4	#5
2nd 블록					#1	#2	#2	#2	#3	#3	#3	#3
3rd 블록									#1	#2	#2	#2



〔그림 6〕 제안하는 XTS-AES 연산기의 라운드 키확장 회로 구조

데 총 12 클록 사이클이 소요되지만, 4 클록 사이클마다 다음 블록에 대한 연산을 시작할 수 있으므로, 입력 데이터가 연속적으로 입력될 때는 매 4 클록 사이클마다 한 블록에 대한 연산을 처리할 수 있는 장점이 있다.

5.2 라운드 키 확장블록의 설계

일반적인 암복호 통신 환경에서는 암복호용 키 정보가 키교환 프로토콜 등을 통해 수시로 변경되므로, 암복호 처리가 필요한 데이터가 입력될 때마다 매번 라운드 키 확장 연산을 수행하는 on-the-fly 키 확장 방식의 연산기 기법이 일반적이다. On-the-fly 키 확장 설계 방식은 각 라운드 연산에 맞춰 라운드 키가 갱신되므로 자원 효율성 측면에서 장점이 있지만, 복호화 동작의 경우 맨 첫 번째 라운드에 사용되는 최종 라운드키를 연산하기 위해서 초기에 약 10 클록 사이클이 부가적으로 소모되는 단점이 있다[20].

반면에, FDE 시스템은 하나의 암복호용 키를 고정해서 사용하므로, 제안하는 연산기는 시스템이 부팅되면 정상적인 암복호 동작을 시작하기 전에 모든 라운드 키 정보 (Key_1: $\{R'_0, \ldots, R'_{10}\}$, Key_2: $\{R_0, \ldots, R_{10}\}$)를 계산하여 레지스터에 미리 저장하도록 설계하였다. 〔그림 6〕에는 최종 설계한 라운드 키 확장 회로의 구조를 나타내었다.

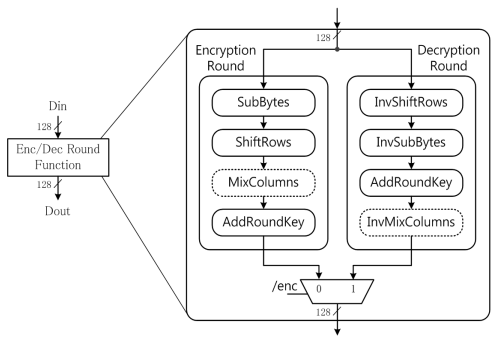
5.3 라운드 함수 블록의 설계

AES 복호화는 암호화의 역순으로 수행되며, 암호화 라운드 함수 연산을 구성하는 SubBytes, ShiftRows, MixColumns 변환 대신에 각 변환의 역함수인 InvSubBytes, InvShiftRows, InvMixColumns 변환이 사용된다. 〔그림 7〕에는 라운드 연산 블록 (〔그림 5〕의 'Enc/Dec Round Function #1 ~ #3')의 내부 구조를 나타내었다. 그림에서 보듯이 라운드 연산 블록은 암호화 라운드 연산 회로와 복호화 라운드 연산 회로를 모두 구현하여, 암호화 및 복호화 동작을 선택적으로 처리할 수 있도록 설계하였다. 또한 〔그림 5〕의 'Enc/Dec Round Function #3' 연산 블록은 최종 라운드 연산 시에는 MixColumns 변환 및 InvMix- Columns 변환을 거치지 않도록 선택하는 회로를 부가적으로 추가하여 구현하였다.

VI. 구현 및 실험 결과

6.1 구현 결과

제안하는 SATA 하드디스크 FDE용 연산기를 Xilinx사의 XC5VFX70T FPGA를 타겟으로 컴파일하여 (ISE 12.4 사용) 내부 구성 요소별 하드웨어



〔그림 7〕 라운드 함수처리 블록의 구조

〔표 2〕 SATA 하드디스크 FDE용 연산기의 하드웨어 복잡도

구성 요소	Area(Slices)	전체면적 대비 %
SATA Bridge IP	2,710	38
XTS-AES 연산기	3,099	44
연산컨트롤러 등 주변로직	1,280	18
전체 FDE 연산기	7,089	100

면적을 〔표 2〕에 나타내었다. 제안하는 SATA FDE용 연산기는 총 7,089 Slices의 하드웨어 자원을 사용하며, 핵심연산 블록인 XTS-AES 연산기는 전체 연산기 면적의 44% 수준인 3,099 Slices 자원을 사용한다.

〔표 3〕에는 동일 계열의 FPGA(Virtex5)를 사용한 기존 XTS-AES 연산기들의 결과와 제안하는 XTS-AES 연산기와의 성능을 비교하여 나타내었다. 앞서 설명한 바와 같이 본 논문에서 제안하는 연산기는 XTS-AES 연산기 뿐만 아니라, SATA Bridge IP와 연산 컨트롤러 등의 주변 로직을 포함하는 반면, 기존에 발표된 연산기들은 대부분 XTS-AES 연산기 부분만을 설계 및 구현한 결과이므로, 여기서는 보다 정확한 비교를 위해 XTS-AES 연산기 자체의 성능만을 나타내었다. 표에서 보는 바와 같이, 기존 연산기들은 128-비트 한 블록을 처리하는데 10 클록 사이클 정도를 소요하는 반면에, 제안하는 연산기는 4 클록 사이클을 사용하므로, 동일한 클록 속도를 기준으로 약 2.5배 이상의 높은 성능을 기대할 수 있다.

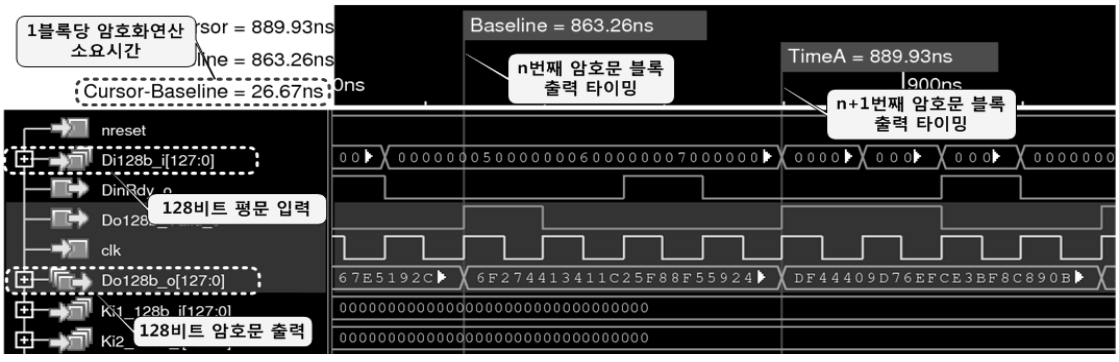
연산기의 동작 및 성능 검증을 위해 Cadence사의 NC-Sim 시뮬레이터〔24〕를 이용하였으며, 〔그림 8〕에는 XTS-AES 연산기의 암호화 시뮬레이션 결과파형을 나타내었다. 그림에서 보듯이 제안하는 연산기는 n 번째 128-비트 암호문 블록이 출력된 다음, 4 클록

〔표 3〕 제안하는 연산기와 기존 연산기의 성능 비교

디자인	동작 Freq. (MHz)	Clock cycles/ 블록	성능 (Gbps)	면적 (Slices)
Proposed	150	4	4.8	3,099
〔9〕	209	10	2.8	1,470
〔14〕	264	12	2.8	1,147

사이클 후에 n+1 번째 암호문을 출력하고, 이에 소요되는 시간은 150MHz 클록 속도를 기준으로 약 26.67ns이므로, 128 bit x 150MHz / 4 clock cycles = 4.8Gbps의 성능을 가짐을 확인할 수 있다.

〔표 3〕에 나타낸 바와 같이 제안하는 XTS-AES 연산기의 최대 동작 클록속도는 Placement & Routing 단계의 컴파일까지 완료했을 때 150MHz 이다. 반면에, 비교대상의 연산기들은 200MHz 이상의 속도를 나타내는데, 이러한 최대 클록 속도는 타겟 FPGA의 Speed grade, 컴파일 버전 및 옵션 등에 따라 상당히 달라질 수 있다. 참고로 Xilinx사 ML507 개발키트 보드에 탑재된 XC5VFX70T FPGA는 Speed Grade가 제일 느린 등급인 '-1'이다. 이와 같은 이유로 제안하는 XTS-AES 연산기는 기존 연산기들에 비해 약 30~45% 정도 느린 150MHz 클록 속도로 동작함에도 불구하고 약 4.8Gbps의 성능을 나타내며, 이는 기존 결과들에 비해 1.7배 이상 높은 수치이다. 한편, 하드웨어 면적 측면에서는 1 라운드 반복형 구조의 기존 연산기들에 비해, 제안하는 연산기는 3 라운드 로직을 사용한 파이프라이닝 구조로 설계하였으므로 약 2.1~2.7배 하드웨어 자원을 사용한다. 하지만, 타겟 FPGA인 XC5VFX70T가 총 44,800개의 Slice 자원을 가지므로, 이는 타겟 FPGA의 전체 자원 관점에서는 5%



〔그림 8〕 XTS-AES 연산기의 암호화 시뮬레이션 파형

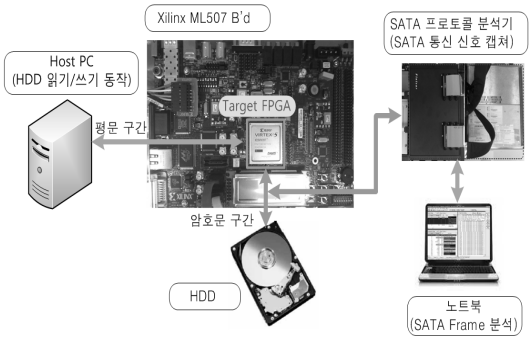
미만의 차이에 해당한다.

실제로 SATA 하드디스크 FDE용 암호칩을 구현함에 있어 XTS-AES 연산기 자체의 성능보다는 입/출력 인터페이스를 포함한 암호칩 전체의 성능이 가장 중요한 수치이다. 본 논문에서 채용한 IntellipProp사의 SATA Bridge IP는 XTS-AES 연산기를 포함한 내부 블록에 75MHz 속도의 클록을 입력한다. 따라서, 실제 구현된 FPGA의 XTS-AES 암복호 처리속도는 XTS-AES 연산기 자체 성능의 절반 수준인 2.4Gbps가 되지만, 이는 SATA Bridge IP가 지원하는 최고 속도에 해당하므로 속도 저하없이 암복호 기능을 구현할 수 있다. 한편, 한 블록당 소요 사이클 수가 10 이상인 기존 연산기들을 본 연산기와 동일한 조건인 75MHz 클록을 사용하도록 구현한다면, 연산성능이 1Gbps 미만으로 떨어지므로 암복호로 인한 속도가 저하가 발생하게 된다.

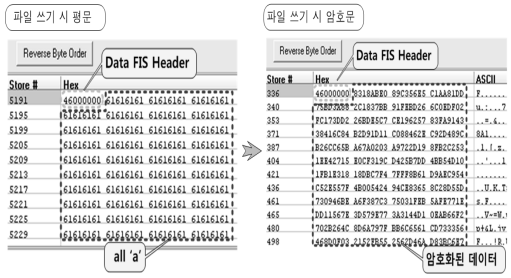
6.2 실험 결과

SATA FDE용 연산기의 시험환경을 〔그림 9〕에 나타낸 바와 같이, 호스트 PC, 하드디스크, Xilinx ML507 보드, 그리고 SATA 프로토콜 분석기 및 노트북으로 구성하였다. 이때, SATA 프로토콜 분석기는 Data Transit사의 SATA Pod와 Bus Doctor Analyzer〔26〕를 사용하였고, 호스트 PC의 사양은 인텔 코어 i7-960(@3.2GHz) CPU〔27〕, ASUS사의 P6X58D -E 마더보드〔28〕, 3GB 메모리, Windows XP 32비트이다.

호스트 PC에서 하드디스크에 쓰기 동작을 발생했을 때의 평문과 암호문 데이터를 SATA 프로토콜 분석기에서 캡쳐한 내용을 〔그림 10〕에 나타내었다. 그림에서 보듯이 문자 'a'로만 구성된 텍스트 파일을 하



〔그림 9〕 SATA FDE 동작 시험 환경



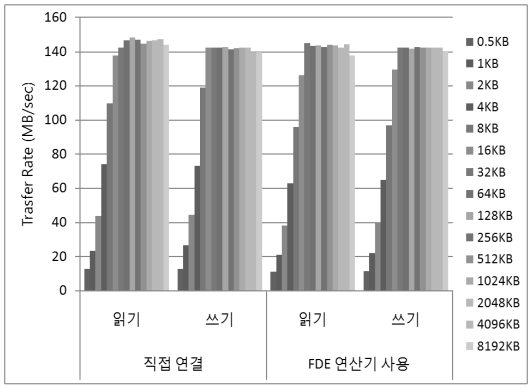
〔그림 10〕 파일 쓰기 시, 평문/암호문 쌍 내용

드디스크에 저장했을 때, 데이터 Frame 헤더정보인 'Data FIS Header' 이외의 모든 데이터는 암호화되어 전송됨을 확인하였다.

〔그림 11〕에는 암호화 장치없이 SATA 케이블에 직접 연결한 경우와, FDE 암호화 장치를 연결했을 때의 삼성 HD103SJ(1TB, 7200rpm)〔29〕 하드디스크에 대한 읽기/쓰기 속도를 측정하여 비교하였으며, 속도 측정에는 ATTO Disk Benchmark 프로그램〔30〕을 이용하였다. 그림에서 보듯이 제안하는 FDE 암호화 장치는 하드디스크에 읽기/쓰기 동작을 수행하는 데이터 단위 크기가 16KB 이상일 때는 암호화 장치없이 직접 연결했을 때와 대등한 140 MB/sec 이상의 성능을 가지므로, 속도저하 없이 FDE 장치 사용이 가능하다는 장점이 있다.

VII. 결론

본 논문에서는 SATA 하드디스크용 FDE 연산기를 제안하고, 이를 FPGA에 구현하고 실제 하드웨어 보드를 이용한 실험 결과를 제시하였다. 제안하는 SATA FDE 연산기는 SATA 인터페이스 기능과



〔그림 11〕 FDE 암호화 장치 연결 시, 하드디스크 전송 속도

IEEE 1619-2007 표준인 XTS-AES 암복호 기능을 단일 FPGA에 구현가능한 장점이 있다. 또한 XTS-AES 연산기는 암복호화로 인한 성능저하가 발생하지 않도록 4단 파이프라이닝 구조로 설계하여 기존 연산기들의 성능에 비해 약 1.7배 개선된 4.8Gbps의 성능을 가진다. 또한, 제안하는 FDE 연산기를 하드웨어 보드에 구현했을 때, Windows XP 32비트 환경에서 SATA II 하드디스크에 대해 최대 140 MB/ sec 이상의 성능을 가지며, 이는 암호화 장치없이 직접 연결했을 때와 동등한 성능을 나타내므로, 실제 시스템 상에서도 속도저하 없는 FDE 동작이 가능함을 확인하였다. 아울러, 제안하는 XTS-AES 연산기는 일반적인 동기식 FIFO와 정합하도록 설계되었으므로, 외부 인터페이스용 IP만을 변경함으로써 SATA 외에 다른 인터페이스를 지원하는 하드디스크에도 손쉽게 적용이 가능하다는 장점이 있다.

참고문헌

〔1〕 M. A. Alomari and K. Samsudin, "A STUDY ON ENCRYPTION ALGORITHMS AND MODES FOR DISK ENCRYPTION", International Conference on Signal Processing Systems, pp. 793-797, May 2009.

〔2〕 K. Scarfone, "Guide to Storage Encryption Technologies for End User Devices", NIST, vol. Special Publication pp. 800-111, Nov. 2007.

〔3〕 J. A. Halderman, S. D. Schoen, N. Heninger, W. Clarkson, W. Paul, J. A. Calandrino, A. J. Feldman, J.Appelbaum, and E. W. Felten, "Lest We Remember: Cold Boot Attacks on Encryption Keys", Proceedings of 17th Usenix Security Symposium, pp. 45-60, July 2008.

〔4〕 Hitachi Self-Encrypting Disk (SED), http://www.hitachigst.com/internal-drives/self-encrypting-drives

〔5〕 Seagate Secure™ Encrypting Drives, http://www.seagate.com/www/en-us/products/self-encrypting-drives

〔6〕 Samsung Solid State Drives with Full Encryption, http://www.samsung.com/global/business/semiconductor/file/media/SSi_SSD_ds_final-1.pdf

〔7〕 IEEE Std P1619-2007, "The XTS-AES Tweakable Block Cipher", published 18 April 2008.

〔8〕 NIST Special Publication 800-38E, "Recommendation for Block Cipher Modes of Operation: The XTS-AES Mode for Confidentiality on Storage Devices", January, 2010.

〔9〕 E. HATZIDIMITRIOU, A. P. KAKAROUNTAS, A. MILIDONIS, "Exploration and Enhancement of P1619-Based Crypto-Cores for Efficient Performance", IEEE International Conference on Consumer Electronics (ICCE), pp. 361-362, Jan. 2011.

〔10〕 E. Hatzidimitriou, A. P. Kakarountas, "Implementation of a P1619 Crypto-Core for Shared Storage Media", Proceedings of IEEE Mediterranean Electrotechnical Conference, MELECON 2010, Valetta, Malta, pp. 597-601, April 2010.

〔11〕 C. Mancillas-Lopez, D. Chakraborty, F. Rodriguez-Henriquez, "Reconfigurable Hardware Implementations of Tweakable Enciphering Schemes", Cryptology ePrint Archive: Report 2007/437, http://eprint.iacr.org.

〔12〕 Elliptic Technologies Inc., "CLP-47: Configurable XTS-AES Cipher Core", IP core, http://www. ellipticsemi.com/products-clp-47.php.

〔13〕 Elliptic Technologies Inc., "CLP-33: XTS-AES Cipher Core", IP core, http://www.ellipticsemi.com/products-clp-33.php.

〔14〕 IPCores Inc., "XTS-AES IEEE P1619 Core Families XTS2 and XTS3", IP cores, http://www.ipcores.com/AES_XTS_IP_core.htm.

〔15〕 Helion Technologies Ltd, "Fast AES XTS/CBC Core for Xilinx FPGA (XEX-based

Tweaked Codebook with Ciphertext Stealing)", IP Core, http://www. heliontech. com/aes_xex.htm.
〔16〕 Algotronix Ltd., "XTS-AES Core - Encryption of Stored Data", IP core, http://www.algotro-nix-store.com/AES_Core_XTS_p/aes-core- xts.htm.
〔17〕 IntelliProp Inc. Serial ATA(SATA) Bridge, http://intelliprop.com.
〔18〕 Xilinx Inc. "ML507 Evaluation Platform Documentation", http://www.xilinx.com /pro- ducts/boards/ml507/docs.html.
〔19〕 P. Rogaway, "Efficient Instantiations of Tweakable Block ciphers and Refinements to Modes OCB and PMAC", Advances in Cryptology - Asiacrypt, vol. 3329 of Lecture Notes in Computer Science, pp. 16 - 31, Sep. 2004.
〔20〕 NIST, "FIPS-197, Announcing the ADVANCED ENCRYPTION STANDARD (AES)", Nov 2001, http://csrc.nist.gov/pub - lications/fips/fips197/fips-197.pdf.
〔21〕 Asic Ws, SATA Host/Device Controller IP Core, http://www.asic.ws.
〔22〕 Hitech Global, SATA Host IP Core, http:www.hitechglobal.com/ipcores/sata.htm.
〔23〕 Serial ATA, http://en.wikipeida.org/wiki /Serial_ ATA.
〔24〕 SATA Storage Technology, http://www. mind- share.com.
〔25〕 구본석, 유권호, 장태주, 이상진, "자원공유기법을 이용한 AES-ARIA 연산기의 효율적인 설계," 정보보호학회 논문지, 제18권, 제6호, pp. 39-49, 2008년 12월.
〔26〕 Data Transit Inc. Serial ATA(SATA) Pod for Bus Doctor Analyzer, http://data-transit. com/products/busdr_pod_sata.html.
〔27〕 Intel CoreTM i7-960 Processor, http:// ark. intel.com/products/37151.
〔28〕 ASUSTek Computer Inc., http://www.asus.com/ Motherboards/Intel_Socket_13366/P6X58DE.
〔29〕 삼성전자 내장 HDD, http://www.samsung. com/sec/consumer/it/harddiskdrives/internal-drives/index.idx ?pagetype=subtype.
〔30〕 ATTO Disk Benchmark, http://www.attotech. com/products/product.php?sku=Disk_Benchmark.
〔31〕 NC-Sim Simulator, http://www.cadence. com

〈著者紹介〉

사 진

구 본 석 (Bonseok Koo) 정회원
1998년 2월: 경북대학교 전자공학과 학사
2000년 2월: 포항공과대학교 전자전기공학과 석사
2009년 2월: 고려대학교 정보보호대학원 박사
2000년 2월~9월: LG정보통신 중앙연구소 연구원
2000년 9월~현재: 국가보안기술연구소 선임연구원
〈관심분야〉 암호칩 설계, 공개키암호 구현, 부채널 공격

사 진

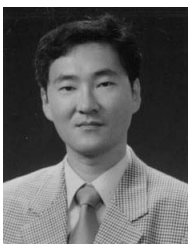
임 정 석 (Jeongseok Lim) 정회원
1987년 2월: 한양대학교 전자통신공학과 학사
1989년 2월: 한양대학교 전자통신공학과 석사
2007년 2월: 한양대학교 전자통신공학과 박사
1989년 2월~2000년 1월: 국방과학연구소 선임연구원
2000년 2월~현재: 국가보안기술연구소 책임연구원(실장)
〈관심분야〉 통신용 보안시스템 설계 및 구현, 유무선 통신시스템, 오류정정부호

사 진

김 춘 수 (Choonsoo Kim) 정회원
1987년 2월: 숭실대 전기공학과 학사
1989년 2월: 숭실대 대학원 전기공학과 석사
1996년 2월: 숭실대 대학원 전기공학과 박사
1990년~1999년: 한국전자통신연구원 팀장
2004년~2005년: 미국 상무성 소속 NIST 객원연구원
2000년~현재: 국가보안기술연구소 본부장
〈관심분야〉 암호장비 개발, 정보보호시스템 평가, 군 사이버전

사 진

윤 이 중 (E joong Yoon) 종신회원
1988년 2월: 인하대 전자계산학과 학사
1990년 2월: 인하대 대학원 전자계산학과 석사
2002년 2월: 충남대 대학원 컴퓨터과학 박사
1990년~2001년: 한국전자통신연구원 부장
2003년: 미국 미네소타주립대학 객원연구원
2001년~현재: 국가보안기술연구소 선임연구본부장
〈관심분야〉 인증시스템, 사이버보안시스템, 금융보안, 정보보호관련법제



이 상 진 (Sangjin Lee) 종신회원
1987년 2월: 고려대학교 수학과 학사
1989년 2월: 고려대학교 수학과 석사
1994년 8월: 고려대학교 수학과 박사
1989년 2월~1999년 2월: 한국전자통신연구원 선임연구원
1999년 2월~2001년 8월: 고려대학교 자연과학대학 조교수
2000년 2월~현재: 고려대학교 정보보호대학원 부교수
〈관심분야〉 대칭키 암호의 분석 및 설계, 정보은닉이론, 컴퓨터 포렌식