2001년 한국정보처리학회 춘계 학술발표논문집 제8권 제1호

IMT-2000을 위한 LILI-128 암호의 고속 구현에 관한 연구

이훈재*
*경운대학교 컴퓨터전자정보공학부
e-mail:hjlee@kyungwoon.ac.kr

A Study on High-Speed Implementation of the LILI-128 cipher for IMT-2000 Cipher System

Hoon-Jae Lee*
*School of Computer, Electronics and Information Communications Engineering, Kyungwoon University

요약

LILI-128 스트림 암호는 IMT-2000 무선단말간 데이터 암호화를 위하여 제안된 128-비트 크기의 스트림 암호방식이며, 클럭 조절형태의 채택에 따라 속도저하라는 구조적인 문제점을 안고 있다. 본 논문에서는 귀환/이동에 있어서 랜덤한 4개의 연결 경로를 갖는 4-비트 병렬 $LFSR_d$를 제안함으로서 속도문제를 해결하였다. 그리고 ALTERA 사의 FPGA 소자 (EPF10K20RC240-3)를 선정하여 그래픽/VHDL 하드웨어 구현 및 타이밍 시뮬레이션을 실시하였으며, 50MHz 시스템 클럭에서 안정적인 50Mbps (즉, 45 Mbps 수준인 T3급 이상, 설계회로의 최대 지연 시간이 20ns 이하인 조건) 출력 수열이 발생될 수 있음을 확인하였다. 마지막으로, FPGA/VHDL 설계회로를 Lucent ASIC 소자 (LV160C, 0.13μm CMOS & 1.5v technology)로 설계 변환 및 타이밍 시뮬레이션한 결과 최대 지연시간이 1.8ns 이하였고, 500 Mbps 이상의 고속화가 가능함을 확인하였다.

1. 서론

암호 구현에 있어서 키 분배 또는 인증 기능이 요구되는 경우 공개 키 암호가 적용되지만, 데이터 암복호화 등 고속 처리가 요구되는 응용에는 스트림 암호나 블록 암호가 많이 사용된다. 블록 암호는 소프트웨어 구현이 용이한 반면 채널 에러시 수신단에서 블록 크기만큼 에러가 확산되어 채널 효율(channel efficiency)이 떨어지며, 비도 수준에 대한 정량화가 불가능한 단점이 있다. 반면 스트림 암호는 에러 확산이 없고, 비도 수준에 대한 수학적 정량화가 가능하며, 하드웨어 구현이 용이하고, 통신 지연이 없으며, 고속 통신이 가능한 것 등의 잇점으로 인해서 이동·무선통신 전송로 구간의 링크 암호 또는 군사·외교용으로 많이 사용되고 있다[1-2].

스트림 암호 알고리즘이란 이진화된 평문과 이진 키 수열의 배타적 논리합(XOR) 연산을 실행하여 암호문을 생성하는 알고리즘을 말하며, 이 때 출력 키 수열에 대한 특성과 발생 방법이 안전도에 직접적인 영향을 미친다.

70~90년대에 발표된 대표적인 키 수열 발생기로는 비메모리 형태의 Geffe 발생기[1-2]와 메모리 형태의 Rueppel 합산 수열 발생기 (summation generator)[3-5]를 들 수 있다. 그리고 최근에는 여러 종류의 클럭 조절(clock-controlled) 형 키 수열 발생기나 비트 처리 형태의 스트림 암호와 블록 처리 형태의 블록 암호를 결합시킨 병렬형 스트림 암호 (parallel stream cipher)[6]등의 알고리즘에 관심이 집중되고 있다.

본 논문에서는 Simpson 등[7]이 IMT-2000 무선단말간 데이터 암호화를 위하여 제안한 LILI-128 암호의 하드웨어 병렬 구현에 대하여 연구한다. LILI-128 암호는 128-비트 크기의 클럭 조절형 스

트림 암호 방식으로 이러한 구조는 동기식 논리회로 구현시 속도가 저하되는 단점이 있다. 즉, 클럭 조절형인 $LFSR_d$는 외부 클럭보다 1~4 배 높은 클럭을 요구하기 때문에 동일한 시스템 클럭 하에서는 데이터 전송속도에 따른 시스템 성능이 저하된다. 본 논문에서는 귀환/이동에 있어서 랜덤한 4개의 연결 경로를 갖는 4-비트 병렬 $LFSR_d$를 제안한다. 그리고 ALTERA사의 FPGA 소자(EPF10K20RC 240-3)[8]를 선정하여 그래픽/VHDL 하드웨어 구현 및 타이밍 시뮬레이션을 실시한 다음 50MHz 시스템 클럭에서 안정적인 50Mbps (즉, 45 Mbps 수준인 T3급 이상, 설계회로의 최대 지연 시간이 20ns 이하인 조건) 출력 수열이 발생될 수 있음을 확인한다. 마지막으로, FPGA/VHDL 설계회로를 Lucent ASIC 소자 (LV160C, 0.13μm CMOS & 1.5v technology)[9]에 적합하게 설계 변환 및 시뮬레이션 실행으로 최대 지연시간이 2.0ns 이하에서 500 Mbps 이상의 고속화가 가능함을 확인코자 한다.

2. 고속화 방안 제안

LILI-128 암호[7]의 구조는 그림 1과 같으며, 사용된 선형 귀환 이동 레지스터(LFSR, linear feedback shift register)는 39단 $LFSR_c$와 89단 $LFSR_d$로 구성되어 있는데, 이 중에서 $LFSR_d$는 $LFSR_c$의 출력에 의하여 클럭 통제를 받게 된다. 통제되는 클럭 수는 통상적인 경우 랜덤하게 설정된 f_c 함수에 의하여 생성된 정수 값 (1~4 범위) 만큼 $LFSR_d$의 클럭을 이동시키며, 그 후 $LFSR_d$의 내부 값으로부터 f_d 필터 함수를 통하여 필터 수열 (filtered sequence) [7]을 발생하게 된다.

LILI-128 암호에서 39단 $LFSR_c$, 89단 $LFSR_d$의 원시다항식 (primitive polynomial) 및 각각 좌측 이동 (left shift)될 귀환 비트 (feedback bit) 조합 입력인 $c[39+t]$, $d[89+t]$는 다음과 같이 정의된다.

$$c(t) = f_c(t) = 2 \cdot c[12+t] + c[20+t] + 1$$

$$g_c(x) = x^{39} + x^{35} + x^{33} + x^{31} + x^{17} + x^{15} + x^{14} + x^2 + 1$$

$$g_d(x) = x^{89} + x^{83} + x^{80} + x^{55} + x^{53} + x^{43} + x^{39} + x + 1$$

$$c[39+t] = c[37+t] \oplus c[25+t] \oplus c[24+t] \oplus c[22+t] \oplus c[8+t] \oplus c[6+t] \oplus c[4+t] \oplus c[t]$$

$$d[89+t] = d[88+t] \oplus d[50+t] \oplus d[47+t] \oplus d[36+t] \oplus d[34+t] \oplus d[9+t] \oplus d[6+t] \oplus d[t]$$

여기서 $LFSR_c$의 레지스터 비트 탭은 좌측으로부터 $c[0], c[1], \cdots, c[37], c[38]$, $LFSR_d$ 레지스터 비트 탭은 $d[0], d[1], \cdots, d[87], d[88]$로 각각 표기되며, $\oplus$는 배타 논리합인 XOR (exclusive-or) 연산을 의미한다.

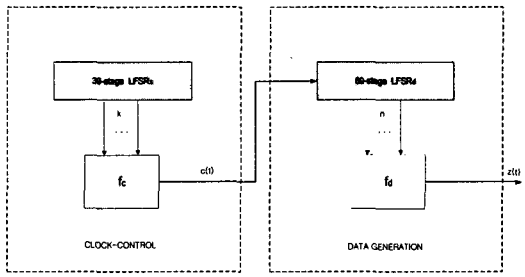

그림 1. LILI-128 스트림 암호

LFSR의 하드웨어 구현 시에는 시스템의 안정성을 고려할 때 시스템 클럭에 맞추어 레지스터 값을 좌측 이동시키는 클럭 동기식 논리 설계 (clock-synchronized logic design) 방법이 일반적으로 많이 적용된다. 그러나 이 방법으로 LILI-128 암호를 구현함에 있어서 일반형인 $LFSR_c$는 상기의 방법으로 쉽게 구현될 수 있지만, 클럭 조절형인 $LFSR_d$는 1~4배의 고속 클럭이 별도로 요구된다. 또한 별도의 고속 클럭 추가문제를 해결하고자 주파수 채배기 (frequency multiplier)를 도입할 수도 있겠지만,

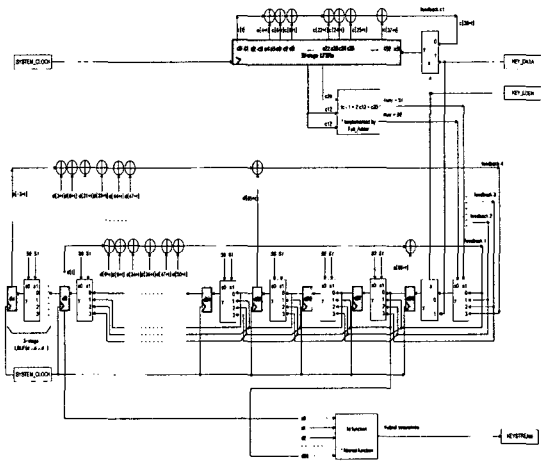

그림 2. LILI-128 고속 구현 (4-bit parallel $LFSR_d$)

고속/초고속 통신에서는 클럭 간격(clock interval)에서의 시간 여유 (time margin)가 작기 때문에 적용이 어렵다.

LILI-128이 갖는 구조적인 문제를 해결하기 위하여 비트 이동 루트가 클럭을 초월하여 1~4 비트씩 가변적으로 이동할 수 있는 4-비트 병렬 입력 $LFSR_d$의 고속 구현 방안을 그림 2와 같이 제안한다.

그림 2의 상반부에 위치한 $LFSR_c$는 일반적인 39단 이동 레지스터 및 귀환 비트 조합으로 구현이 가능하다. 그리고 출력 f_c 회로는 $LFSR_d$의 좌측 이동 클럭 수를 결정하는 것으로서 전가산기 (full adder)를 사용하면 쉽게 구현된다. 그러나 89단 $LFSR_d$ 각 비트들은 $d_0, d_1, \ldots, d_{88}$로 나타낸 레지스터에 저장된 다음 f_c가 정하는 값에 따라 1~4 비트씩 이전 값 (우측 레지스터)으로부터 4-1 멀티플렉서 (4-1 MUX) 회로를 통하여 입력된다. 이 부분에 대한 설계 아이디어를 "4-비트 병렬 $LFSR_d$ (4-bit parallel $LFSR_d$)" 라고 부르며, 고속화 구현 회로의 핵심부분이다. 예를 들면, 그림에서 d_{84} 레지스터의 경우 그 이전 4개의 레지스터들 $d_{85}, d_{86}, d_{87}, d_{88}$ 중에서 랜덤하게 어느 한 입력이 선택 ($f_c=1$일 때는 d_{85}로부터, $f_c=4$일 때는 d_{88}로부터 각각 입력)되는데 이때 선택 신호들 ($s1, s0$)은 f_c로 구현된 전가산기의 출력으로부터 얻어진다. 그리고 $LFSR_d$의 좌측에는 3-비트 LBUF가 4개의 귀환 비트 조합을 계산하기 위하여 d_0의 출력을 차례로 보관하고 있다. 4개의 귀환 비트 조합 중에서 feedback 1은 원래의 귀환 비트와 동일한 탭의 XOR 조합을, feedback 2는 feedback 1에 비하여 각각 1-비트씩 좌측 이동된 탭의 XOR 조합을, feedback 3는 각각 2-비트씩 좌측 이동된 탭의 XOR 조합을, feedback 4는 각각 3-비트씩 좌측 이동된 탭의 XOR 조합을 이룬다. 사용된 4개의 feedback 조합은 다음과 같다.

$$d[89+t] = d[88+t] \oplus d[50+t] \oplus d[47+t] \oplus d[36+t] \oplus d[34+t] \oplus d[9+t] \oplus d[6+t] \oplus d[t]$$

: *feedback* 1

$$d[88+t] = d[87+t] \oplus d[49+t] \oplus d[46+t] \oplus d[35+t] \oplus d[33+t] \oplus d[8+t] \oplus d[5+t] \oplus d[-1+t]$$

: *feedback* 2

$$d[87+t] = d[86+t] \oplus d[48+t] \oplus d[45+t] \oplus d[34+t] \oplus d[32+t] \oplus d[7+t] \oplus d[4+t] \oplus d[-2+t]$$

: *feedback* 3

$$d[86+t] = d[85+t] \oplus d[47+t] \oplus d[44+t] \oplus d[33+t] \oplus d[31+t] \oplus d[6+t] \oplus d[3+t] \oplus d[-3+t]$$

: *feedback* 4

마지막으로 LILI-128의 출력 수열은 그림 하단에 설정된 비선형 여과 함수 (nonlinear filter function) f_d로부터 얻어지는 비트 수열이 된다.

3. 설계 구현

제안된 고속화 병렬 구현 방안을 검증하기 위하여 그림 3의 회로를 설계하였으며, 이는 ALTERA사의 FPGA 소자 (EPF10K20RC240-3)를 선정하여 8개의 세부 블록으로 구현시킨 그래픽 설계 회로이다.

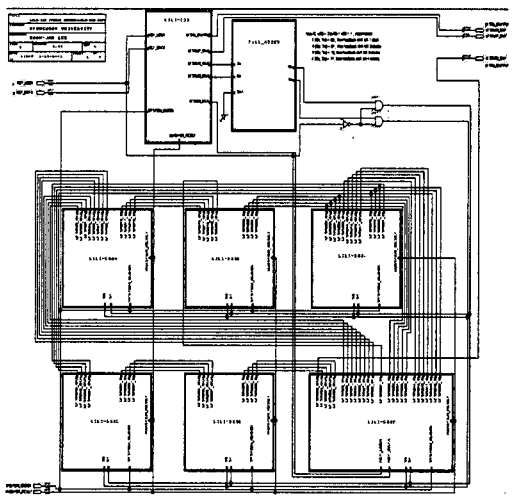

그림 3. LILI-128 스트림 암호 구현 예

표 1에서는 제안된 병렬 하드웨어 구현 방안 및 그 시뮬레이션 결과를 기존의 방안과 비교하였는데, 하드웨어 복잡도 측면에서 소규모 증가가 예상되지만 키수열 발생기의 안전성을 유지하면서도 그 성능을 최대 4배까지 고속화시킬 수 있었다. 즉, 게이트 수로 살펴본 하드웨어 복잡도가 2.5배 증가되었으며, 키 수열 출력속도가 최대 4배까지 향상 가능함을 확

표 1. LILI-128의 구현 방안 비교표

항목	기존방안	제안 방안 (병렬 구현)
구현 방법	-동기식 회로 구성 -LFSR$_d$를 직렬구성	-동기식 회로 구성 -LFSR$_d$를 병렬구성
ALTERA FPGA소자 (EPF10K20 RC240-3) at 50MHz	12.5~50 Mbps 가변속도 (시스템클럭 이동 횟수 만큼 분주된 출력 속도)	50 Mbps 고정 속도 (최장지연시간은 18ns 이하임)
Lucent ASIC 소자 (LV160C, 0.13μm) at 500MHz	-	500 Mbps 고정 속도 (최장지연시간은 1.8ns 이하임)
Gate 수 (단, 1 F/F=5 AOI gate로 환산)	688 게이트 -128 D F/Fs -2 (2-1) MUXs -14 XORs	1745 게이트 (약 2.5배) - 131 D F/Fs - 89 (4-1)MUXs - 2 (2-1) MUXs - 35 XORs

인하였다. 마지막으로 50 MHz 시스템 클럭을 인가한 경우 안정적인 키 수열 출력을 낼 수 있음을 FPGA 소자를 통하여 확인할 수 있었다. 초고속 암호 통신을 위한 ASIC 설계 변환 및 시뮬레이션에서는 상기 FPGA 의 성능을 10배 정도 향상이 가능하였기 때문에 500 Mbps의 속도가 가능함을 확인하였다. 이 보다 더 높은 통신 속도를 위해서는 참고문헌[6]에서 제시된 병렬형 스트림 암호의 적용이 요구된다.

4. 결 론

LILI-128 스트림 암호는 클럭 조절형 스트림 암호 방식으로서 이러한 형태는 동기식 논리회로에 따른 하드웨어 구현에 있어서 속도를 떨어뜨리는 구조적인 문제점을 안고 있다. 본 논문에서는 이러한 속도 저하의 문제를 해결하는 LILI-128 스트림 암호의 고속화 구현 방법을 연구하여 하드웨어 구현에 따른 구조적인 문제점을 보완하였고, 그 결과 기존의 구현 방안에 비하여 최대 4배까지 고속화가 가능함을 확인하였다. 또한 하드웨어 설계 검증을 위하여 ALTERA 사의 Max+plus II로 타이밍 시뮬레이션을 실시하였고, FPGA소자(EPF10K20RC240-3)를 선정하여 하드웨어로 구현하였다. 구현된 회로는 50MHz 시스템 클럭에서 안정적인 50Mbps 출력 수열이 발생될 수 있음을 확인하였다. 마지막으로, FPGA/VHDL 설계회로를 Lucent ASIC 소자(LV160C, 0.13μm CMOS & 1.5v technology)에 적합하게 설계 변환 및 시뮬레이션한 결과 최대 지연시간이 1.8ns 이하였기 때문에 500 Mbps 이상의 고속화가 가능함을 확인하였다. 이 보다 더 높은 통신 속도를 위해서는 병렬형 스트림 암호의 적용이 요구된다.

참고문헌

[1] B. Schneier, Applied Cryptography (2nd edition), John Wiley & Sons, Inc., 1996.

[2] A.J. Menezes, P.C. Oorschot and S.A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1997.

[3] R. A. Rueppel, "Correlation Immunity and the Summation Generator," *Advances in Cryptology, Proceedings of CRYPTO'85*, pp. 260-272, 1985.

[4] R. A. Rueppel, *Analysis and Design of Stream Ciphers*, Springer-Verlag, 1986.

[5] Hoonjae Lee, Sangjae Moon, "On An Improved Summation Generator with 2-Bit Memory," Signal Processing , Vol. 80, No.1, pp. 211~217, Jan. 2000.

[6] 이훈재, 문상재 "고속 안전 통신을 위한 병렬형 스트림 암호," 한국통신학회 논문지, 2001년 3월호 게재 예정.

[7] L. Simpson, E. Dawson, J. Dj. Golic and W. Millan, "LILI Keystream Generator," Proceedings of the Seventh Annual Workshop on Selected Areas in Cryptology-SAC'2000 to appare in Springer-Verlag LNCS, 2000.

[8] Altera technical data sheets in http://www.altera.com.

[9] Lucent technical data sheets in http://www.lucent.com.