고속 모뎀에서의 AES-CCM 보안 모드 구현에 관한 연구

논 문
60P-4-17

Research on the Implementation of the AES-CCM Security Mode in a High Data-Rate Modem

이 현 석† · 박 승 권*
(Hyeon-Seok Lee · Sung-Kwon Park)

Abstract - In high data-rate communication systems, encryption/decryption must be processed in high speed. In this paper, we implement CCM security mode which is the basis of security. Specifically, we combine CCM with AES block encryption algorithm in hardware. With the combination, we can carry out encryption/decryption as well as data transmission/reception simultaneously without reducing data-rate, and we keep low-power consumption with high speed by optimizing CCM block.

Key Words : CCM, AES, CTR, CBC-MAC, Security Mode

1. 서 론

최근 고속 무선 통신을 목표로 하여 UWB, IEEE802.11n 등이 개발되고 있다. UWB는 STB와 평판 DTV같은 영상기기에서 HD 영상 신호를 전송하는 무선 AV 관련 시장과 휴대성, 이동성에 중심을 둔 PC 주변기기 관련 시장을 목표로 하고 있다. 또한, IEEE802.11n 기술은 무선랜에서 가장 단점으로 지적되어 온 통신 속도에 대한 한계를 극복하기 위해 최대 500Mbps의 속도를 목표로 하며, 무선랜 관련 선진업체를 중심으로 칩셋을 출시하고 있다.[1]

이 기술들은 공통적으로 물리계층에서의 고속화를 주 목표로 하고 있지만, 이에 못지 않게 효율적인 MAC 구조 설계의 의한 시스템 성능 최적화와 무선 통신의 취약부분인 보안이 중요한 이슈로 남아 있다. 데이터 통신에서 암호화 속도와 보안 능력을 향상시키기 위하여 미국의 NIST (National Institute of Standards and Technology)는 2001년 DES(Data Encryption Standard)를 대체하기 위한 AES(Advanced Encryption Standard)로 Rijndael 알고리즘을 선정하였다.[2] 무선 통신의 취약 부분 중 하나인 보안을 강화하기 위하여 UWB, IEEE802.11n에서는 블록암호 알고리즘인 AES를 사용한 CCM(CTR+CBC-MAC) 보안 모드를 사용한다.[3-4]

AES의 중요성이 증가함에 따라 AES의 ASIC 및 FPGA 구현에 관한 많은 연구들이 발표되었다. 특히 AES 암호화 블록의 게이트 수를 감소시키기 위해 S-box를 체(Field) 변환 방법을 사용하여 LUT(Look-Up Table)를 대체하는 방법과 S-box의 개수를 줄이고 순차적으로 S-box를 최적화하는 folded architecture에 대한 연구가 이루어졌다.[5-6]

† 교신저자, 정회원 : 한양대 전자통신컴퓨터공학부 박사과정, 전자부품연구원 선임연구원
E-mail : hslee75@keti.re.kr
* 비 회 원 : 한양대 전자통신컴퓨터공학부 교수 · 공박
접수일자 : 2011년 8월 30일
최종완료 : 2011년 11월 24일

그러나, 보안 기능을 구현하는데 있어 통신 모뎀과 연계하여 시스템 버스 사용 효율을 최적화하기 위한 방법에 대한 연구는 미비하였다.

본 논문에서는 multi-band OFDM alliance(MBOA) UWB 고속 통신을 위한 효율적인 AES 블록 암호 알고리즘을 사용한 CCM 보안 모드의 설계 방법과 시스템 버스 사용 효율을 높여 칩의 소모 전력을 저감하기 위한 클럭 최적화 방법론을 설명한다.

본 논문은 다음과 같이 구성되어 있다. 먼저 2장에서는 MBOA UWB 규격에서의 CCM 보안 모드의 기본적인 구조에 대해 설명하고, 3장에서는 시스템 버스 사용율과 클럭 최적화를 고려한 CCM 구현방법과 결과를 설명하고 마지막으로 4장에서 본 논문의 결론을 맺는다.

2. CCM 구성

2.1 CCM 개요

CCM 동작모드는 미국국가기술표준국(NIST : National Institute of Standards and Technology)에서 제시한 AES의 동작모드 중 하나로써 CBC 모드와 CTR 모드를 결합하여 데이터의 무결성과 은닉성을 동시에 보장한다.[7] AES의 CBC 동작모드를 이용하여 메시지의 무결성을 위한 MIC(Message Integrity Code)를 생성하고, CTR 모드를 이용하여 메시지의 은닉성을 위한 암호/복호 연산을 수행한다. 암호화는 보안 페이로드의 전부 또는 일부에 적용되며, 무결성 코드는 MAC헤더와 보안 페이로드 전체에 대해서 생성한다.

CCM은 M(무결성 코드의 바이트 수)과 L(길이 필드를 나타내는 바이트 수)의 두 가지 입력 파라미터를 가진다. MBOA UWB 규격에서는 M=8, L=2의 값을 가진다. CCM은 보안 프레임을 보호하기 위해 갱신 가능한 암호키와 매 프레임마다 다른 값을 가지는 Nonce 사용을 요구한다. CCM nonce는 동일한 키를 사용하여 같은 값을 사용하면 암호학적으로 결점이 생기므로 같은 값을 재사용하면 안 된다. 그

래서 키의 갱신 주기는 CCM nonce가 반복되지 않도록 결정되어야 한다. MBOA UWB에서 CCM nonce는 13 옥텟으로 구성된다. Nonce는 인증 블록 B_0의 구성요소가 되어, CBC-MAC의 입력값이 된다. 또한, 암호화 블록 A_i의 구성요소가 되어 카운터 모드의 입력값이 된다.

옥텟 : 2	2	3	6
송신스테이션 주소	수신스테이션 주소	TKID	SFN

그림 1 CCM Nonce 입력의 구조
Fig. 1 Nonce input to the CCM algorithm

2.2 CBC-MAC 모드

CCM 무결성 코드 생성 블록 구조는 그림 2와 같다. 무결성 블록 B_0은 CBC-MAC 무결성 코드 생성의 첫 번째 블록이며, CBC-MAC의 플래그와 Nonce, 암호화된 데이터의 길이 l(m)으로 구성된다. 무결성 블록 B_1은 CBC-MAC 무결성 코드의 두번째 블록이며, 추가 무결성 검사 데이터의 길이 l(a)와 MAC헤더, 암호화 오프셋으로 구성된다. 그리고, 무결성 블록 B2, ..., B_N은 암호화되지 않는 페이로드 부분과 암호화 되는 페이로드로 구성된다. 암호화되지 않는 페이로드의 무결성 블록들은 16바이트 블록 길이로 만들기 위해 "0"으로 채워넣기를 해야한다. 마찬가지로 암호화되는 페이로드의 무결성 블록들도 16바이트 블록 길이로 만들기 위해 "0"으로 채워넣기를 해야 한다. 그러나, "0"으로 채워넣기를 한 부분은 실제로 전송되지는 않는다.

옥텟 : 1	13	2	2	10	2	1	1	EO	0~15	P~EO	0~15
Flags(=0 x59)	Nonce	L(m) : 암호화된 데이터의 길이	L(a) : 추가 무결성 검사 데이터 길이	MAC 헤더	암호화 오프셋	예약됨	0	암호화되지 않는 페이로드	'0'으로 패딩	암호화되는 페이로드	'0'으로 패딩
B_0			B_1					B_2, ···, B_(M-1)		B_M, ···, B_N	

그림 2 CBC-MAC 무결성 코드의 입력
Fig. 2 Input to CCM authentication blocks

CBC-MAC은 다음의 수식에 의해 무결성 코드 T를 생성한다.

$$X_1 := E(K, B_0) \quad (1)$$

$$X_{i+1} := E(K, X_i \oplus B_i) \text{ for } i = 1, \ldots, n \quad (2)$$

$$T := first - M - octets(X_{n+1}) \quad (3)$$

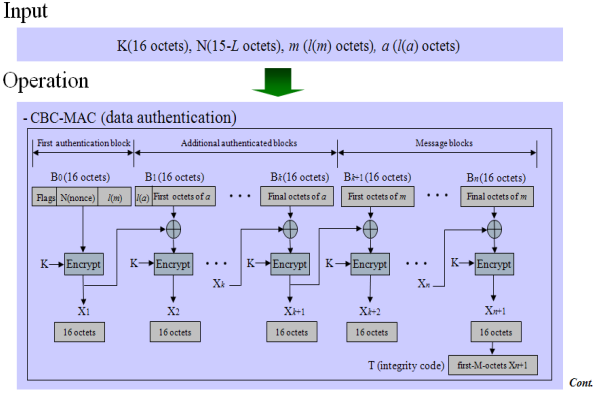

그림 3 CBC-MAC 동작 구조
Fig. 3 CBC-MAC Operation

CCM 무결성 생성 코드는 CBC-MAC 모드를 사용하며, CCM 생성 블록의 구조는 그림 3과 같다.

2.3 CTR(카운터) 모드

CCM은 보안 페이로드 부분을 암호화하고 CBC-MAC 무결성 코드를 암호화하기 위해 A_0, A_1, ..., A_m 암호화 블록을 사용한다. 암호화 블록 A_i의 구조는 그림 4와 같다. 데이터 암호화는 카운터 모드를 사용한다.

13	13	2
Flags = 0x01	Nonce	카운터 i

그림 4 암호화 블록 A_i의 구조
Fig. 4 Format of A_i blocks

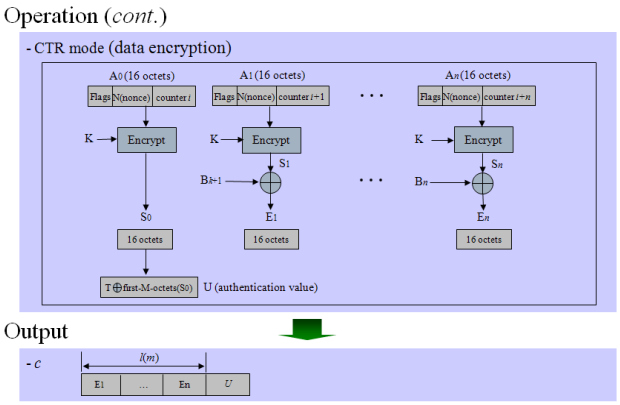

그림 5 카운터 모드의 동작 구조
Fig. 5 CTR operation

카운터는 보안 프레임 전송을 시작할 때 "0"으로 초기화되어, 매 암호화 블록마다 "1"씩 증가한다.

무결성 코드 U는 S0를 이용하여 계산된다.

$$S_i := E(K, A_i) \text{ for } i = 0, 1, 2, \ldots. \quad (4)$$

$$U := T \oplus first - M - octets(S_0) \quad (5)$$

2.4 복호화

메시지를 복호화하기 위해서는 대칭키, Nonce, 암호화되지 않은 보안 메시지, 암호화된 보안 메시지, 무결성 코드가 필요하다.

복호화는 암호화된 메시지로부터 원래의 메시지를 복원한 다음 무결성 코드를 계산하여 수신된 무결성 코드와 비교하여 해당 프레임의 암호학적 무결성을 검사한다.

3. CCM 구현 및 결과

3.1 MAC과 CCM의 인터페이스 설계

데이터 송/수신과 동시에 CCM 동작을 수행할 수 있도록 CCM을 MAC전용 DMA와 MAC사이에 위치시켰다. AHB1은 CPU 및 모뎀이 버스 마스터로서 버스를 점유하고, AHB2는 호스트 인터페이스가 버스 마스터로서 AHB2의 버

스를 점유하도록 한다. 시스템 버스 리소스를 독립적으로 운영토록 구현함으로써 마스터로 하여금 버스 점유를 위한 불필요한 대기 시간을 최소화할 수 있다. 그리고, 절약된 시간을 CPU가 사용할 수 있게 된다. 즉 모뎀의 전송구간에서 호스트는 AHB2 버스를 점유하여 Data Queue를 Access하고 동시에 모뎀은 AHB1 버스를 이용하여 Data Queue를 Acess하여 시간적으로 동시에 발생 가능한 마스터들의 버스 점유의 충돌을 피한다.

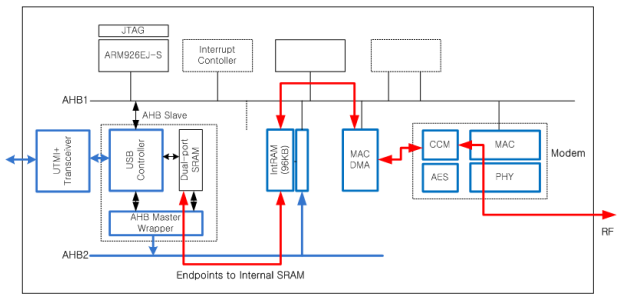

그림 6 송수신에 따른 데이터 경로
Fig. 6 Data path during transmission and reception

다음은 MAC과 보안 관련 블록과의 인터페이스를 도시한 그림이다.

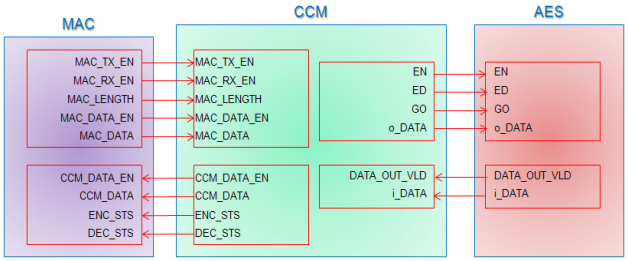

그림 7 보안 블록에서의 MAC, CCM, AES간의 인터페이스
Fig. 7 Interface structure of security related blocks such as MAC, CCM and AES

송신시 MAC은 MAC헤더와 MAC페이로드를 포함한 송신 데이터를 CCM 블록에게 전송을 하고, 이에 따라 CCM 블록은 해당 키와 Nonce를 생성하여 위에서 설명한 Encryption과 무결성 코드를 생성하며, 수행 결과를 ENC_STS신호를 통해서 MAC에게 알려준다.

수신시에는 MAC이 PHY로부터 수신한 MAC헤더와 MAC페이로드를 CCM블록에게 전송하면, CCM 블록은 송신시에 사용한 키와 Nonce를 추출하여 Decryption과정을 수행한 후 그 결과를 DEC_STS신호를 통해 MAC에게 알려준다. 이 모든 과정은 송/수신과 동시에 실시간으로 이루어지므로 송/수신 암호화 관련 F/W의 제어를 최소화할 수 있고 고속 동작이 가능하다.

3.2 암호화 블록의 설계

고속의 CCM 암/복호화 기능을 처리하기 위하여 하드웨어 기반의 CCM 처리 블록을 설계하였으며 블록 암호 알고리즘은 128bits AES를 사용하였다.

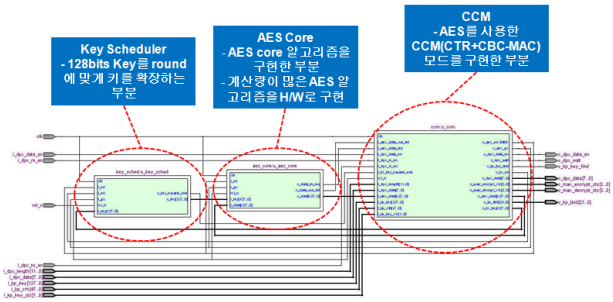

그림 8 보안 블록의 하드웨어 구조
Fig. 8 Hardware structure of security block

3.2.1 Key Scheduler

AES 키 확장 알고리즘은 128비트의 초기 키(K0)를 입력받아 이를 seed로 하여 매 라운드 연산에 사용되는 키를 생성하며, i번째 라운드 키 (Ki)가 (i-1)번째 라운드 키 (Ki-1)로부터 생성되는 체인형태의 연산구조를 갖는다. Key Scheduler를 하드웨어로 구현함으로써 별도의 소프트웨어 작업을 줄였으며, 그림 9에 나타난바와 같이 키 확장을 시작하여 44클럭후에 11개의 라운드 키 생성을 완료한다.

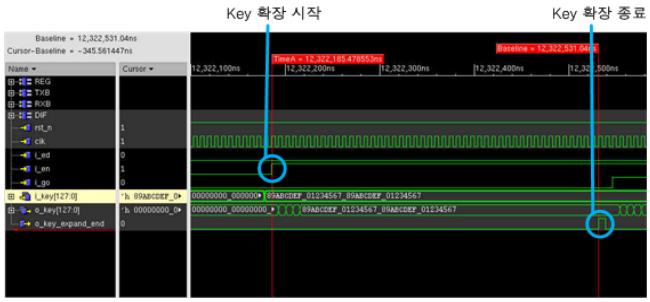

그림 9 키 확장 타이밍도
Fig. 9 Timing diagram of the key expansion

3.2.2 AES Core

AES 코어는 블록 길이와 키 길이가 모두 128비트이며, 10번의 라운드 연산으로 구성되는 AES 알고리즘을 구현한다. AES 알고리즘의 암호 연산과정은 초기 라운드 키 가산 후, 9번의 반복 라운드 변환과 최종 라운드 변환의 과정으로 처리된다. 128 비트 입력 데이터를 암호화하는데 총 12클럭이 소요된다. 즉 라운드당 1클럭, 총 11 라운드에 11클럭과 제어용 1클럭을 더해서 총 12클럭이다.

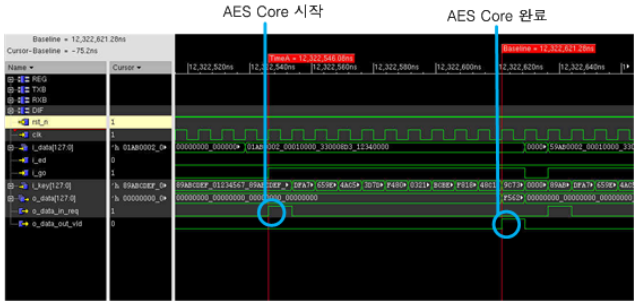

그림 10 AES Core 동작 타이밍도
Fig. 10 Timing diagram of the AES core

3.2.3 CCM

CCM(CTR+CBC-MAC) 모드를 구현한 부분으로써, 암/복호화 과정이 실시간으로 처리될 수 있도록 하였으며, 예외사항으로 Key 불일치 에러나 데이터 무결성 에러등을 처리한다. TDMA방식에서는 송/수신이 Half Dulpex 방식으로 동작하므로 AES Core와 Key Scheduler, CCM 블록을 각각 1개씩 사용하여 게이트 수를 최소화하도록 설계하였다. 아래 그림에서 보는 바와 같이 CBC-MAC 동작을 통하여 무결성 코드를 먼저 생성하고, 이어서 CTR 모드 동작을 통해서 암호화 과정을 수행한다.

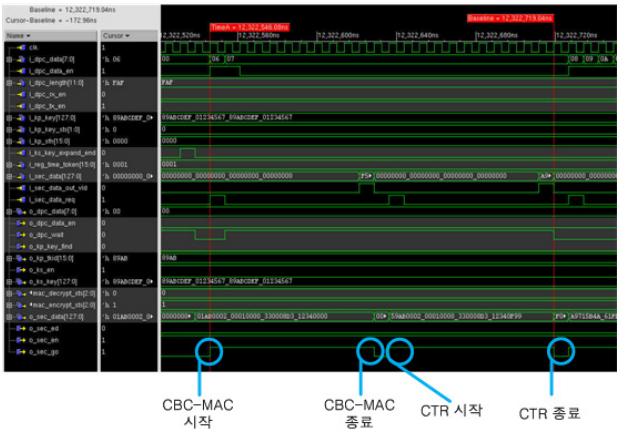

그림 11 CCM 동작 타이밍도

Fig. 11 Timing diagram of the CCM operation

3.3 CCM 블록의 성능

표 1 CCM 연산 파라미터

Table 1 Parameters of the CCM operation

기호	설명	값	비고
N_{block}	암호화 단위 블록의 길이	16	단위:octet (128 bits)
$l(m)$	암호화 할(된) 메시지의 길이	가변	단위:octet
$l(a)$	무결성 검사할 메시지의 길이	가변	단위:octet
T_{KS}	Key Scheduling을 수행하는데 소요되는 클럭 수	44	단위:clock
T_{B_0}	CBC-MAC(B0) : B0블록의 CBC-MAC 동작을 수행하는데 소요되는 클럭 수	12	단위:clock
T_{B_1}	CBC-MAC(B1) : B1블록의 CBC-MAC 동작을 수행하는데 소요되는 클럭 수	12	단위:clock
T_{B_i}	CBC-MAC(Bi) : Bi블록의 CBC-MAC 동작을 수행하는데 소요되는 클럭 수	12	단위:clock
T_{A_0}	CTR(A0) : A0블록의 CTR 동작을 수행하는데 소요되는 클럭 수	12	단위:clock
T_{A_i}	CTR(Ai) : Ai블록의 CTR 동작을 수행하는데 소요되는 클럭 수	12	단위:clock

$$Number\ of\ CTR(A_i):\ N_{ctr(A_i)} = ceiling(l(m), N_{block}) / N_{block} \quad (6)$$

카운터 모드를 사용하는 16바이트 블록의 개수는 식(6)과 같이 암호화할 메시지 $l(m)$을 16바이트 기준 배수 올림(ceiling)을 통해서 계산한다.

$$Number\ of\ CBC-MAC(B_i):\ N_{cbc-mac(B_i)} = N_{ctr(A_i)} + ceiling(l(a) - l(m), N_{block}) / N_{block} \quad (7)$$

CBC-MAC 모드를 사용하는 16바이트 블록의 개수는 순수 CBC-MAC만 수행하는 블록의 16바이트 기준 배수 올림(ceiling)과 카운터 모드를 사용하는 블록 개수의 합이다.

$$Total\ Number\ of\ Clock = T_{KS} + T_{B_0} + T_{B_1} + T_{B_i} \times N_{cbc-mac(B_i)} + T_{A_0} + T_{A_i} \times N_{cbc-mac(B_i)} \quad (8)$$

따라서, CCM 모드를 수행하는데 소요되는 총 클럭의 수는 식(8)과 같이 계산된다.

위의 수식을 바탕으로 $l(m)$과 $l(a)$를 변경하면서 CCM 연산을 수행하기 위해 필요한 클럭수와 수행연산비트 대 클럭 비율(bits/clock)을 계산하였다.

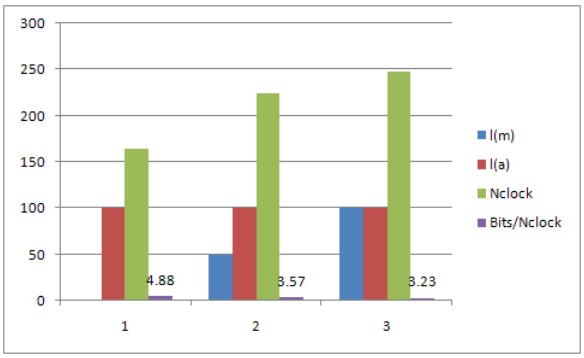

그림 12 패킷 길이가 짧은(100바이트) 비콘이나 제어 프레임의 경우

Fig. 12 In the case of the beacon or command frame with short packet size

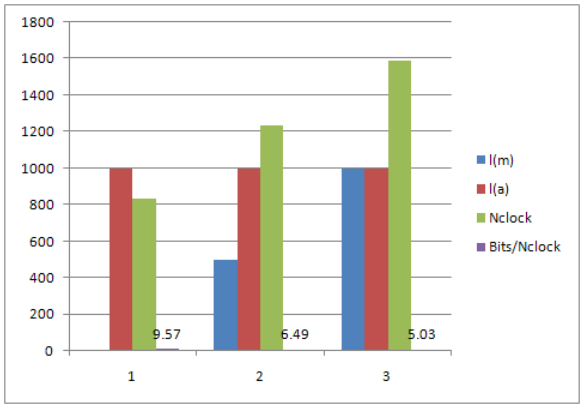

그림 13 패킷 길이가 1,000바이트인 경우

Fig. 13 In the case of data frame with packet length of 1,000 bytes

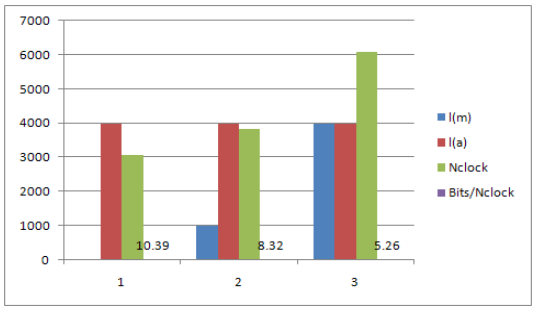

그림 14 패킷 길이가 4,000바이트인 경우
Fig. 14 In the case of data frame with packet length of 4,000 bytes

위의 그림 12,13,14에서 보듯이 $l(m)$, $l(a)$의 길이에 따라 Bits/clock값이 3.23~10.39의 값을 나타낸다.

$l(m)$과 $l(a)$의 값이 동일한 경우, 즉 모든 패킷에 대해 암호화와 무결성 코드를 생성하는 경우 수행연산비트 대 클럭 비율(bits/clock)값은 3.23~5.26의 값을 나타낸다.

만약 전송속도가 500Mbps이라고 가정하면, 하드웨어 클럭 동작속도는 그림 12의 경우에는 $(500Mbits/\sec)/(3.23bits/clk) \cong 155MHz$보다 커야 한다.

그림 13의 경우에는 $(500Mbits/\sec)/(5.03bits/clk) \cong 100MHz$보다 커야 한다. 마지막으로 그림 14의 경우에는 $(500Mbits/\sec)/(5.26bits/clk) \cong 96MHz$보다 커야 한다.

그러나, 패킷 길이가 짧은 비콘이나 제어 프레임은 일반적으로 낮은 전송속도로 전송하거나, 모든 프레임에 대해 암/복호화와 무결성 코드를 생성하는 경우가 아니므로, 구현상의 여유값을 고려하여 수행연산비트 대 클럭 비율(bits/clock)을 5로 선정하여 구현 가능하다.

4. 결 론

본 논문에서는 고속의 통신 속도를 지원하는 CCM 설계 방법에 대한 고찰을 하였으며, RTL 시뮬레이션과 수식적 계산을 통해 그 성능을 검증하였다.

하드웨어 형태로 CCM과 AES 블록암호 알고리즘을 설계하여 데이터 송/수신과 동시에 암/복호화 연산을 수행함으로써, 통신 속도의 저하없이 고속 처리가 가능하였다. 그리고, 다양한 패킷 형태에 따른 MAC 최저 동작 속도에 대한 분석을 통해서 "물리계층 통신속도" 대비 "MAC 클럭" 비율이 최대 6bits/clock까지 가능하다는 결론을 도출하였다. 가령 480Mbps의 물리계층 통신속도를 지원하기 위해서는 MAC 클럭이 최소 96MHz 이상이 되어야 한다.

특히, 본 논문에서 제안한 CCM 설계 부분은 앞으로의 고속 통신 칩 설계에 적용 가능하며, MAC 하드웨어의 동작 클럭을 낮추어 저전력 설계에 중요한 역할을 할 것으로 기대한다.

감사의 글

본 연구는 지식경제 프론티어 기술개발사업의 일환으로 추진되고 있는 지식경제부의 유비쿼터스컴퓨팅및 네트워크원천기반기술개발사업의 지원에 의한 것으로써, 관계부처에 감사 드립니다.

참 고 문 헌

[1] 정창모, 김용석 "UWB 표준화 현황", 정보처리학회지, v.16, no.3, pp.12-17, 2009.
[2] National Institute of Standards and Technology, "Specification for the Advanced Encryption Standard (AES)", FIPS Publication 197, 2001.
[3] WiMedia Alliance, "Distributed Medium Access Control(MAC) For Wireless Network, Release 1.0", MBOA, 2005.
[4] IEEE, "802.11n-2009 IEEE Standard for Information Technology, Wireless LAN Medium Access Control(MAC) and Physical Layer(PHY) Specifications Amendment 5 : Enhancements for Higher Throughput", 2009.
[5] 김종환, 신경욱, "A Design of AES-based WiBro Security Processor", 대한전자공학회 v.44 no.7, pp.71-80, 2007.
[6] 이동호, "무선 내장형 시스템을 위한 저비용 AES의 구현", 대한전자공학회 v.41 no.12, pp.67-74, 2004.
[7] D.Whiting, R.Housley, "RFC 3610 on Counter with CBC-MAC(CCM)", IETF, 9월, 2003.
[8] Block, John Rogaway, Phillip, "CBC MACs for Arbitrary-Length Messages : The Three-Key Constructions", Journal of cryptography, v.18 no.2, pp.111-131, 2005.

저 자 소 개

이 현 석 (李 鉉 錫)
2000년 2월 한양대학교 전자통신·전파공학과 졸업(학사), 2002년 2월 한양대학교 전자통신공학과 졸업(석사), 2006년 ~ 현재 한양대학교 전자통신공학과 박사과정, 2003년~현재 : 전자부품연구원 통신네트워크센터 선임연구원
Tel : 02) 6388-6663
FAX : 02) 6388-6679
E-mail : hslee75@keti.re.kr

박 승 권 (朴 承 權)
1982년 2월 한양대학교 전자공학과 졸업(학사), 1983년 8월 Stevens Institute of Technology, 전자공학과 졸업(석사), 1987년 12월 Rensselaer Polytechnic Institute, 전자공학과 졸업(박사), 1987년 9월~1992년 8월 Rensselaer Polytechnic Institue, Electrical, Computer and Systems Engineering Dept., 조교수, 1993년 3월~현재 한양대학교 전자통신컴퓨터공학부, 교수
Tel : 02) 2294-0366
FAX : 02) 2281-9912
E-mail : sp2996@hanyang.ac.kr