

고속 무선 LAN 시스템을 위한 저전력/저면적 MIMO-OFDM 기저대역 프로세서 설계

정회원 임준하*, 조미숙*, 종신회원 정윤호**, 정회원 김재석***

Design of Low-Power and Low-Complexity MIMO-OFDM Baseband Processor for High Speed WLAN Systems

Junha Im*, Misuk Cho* *Regular Members,*

Yunho Jung** *Lifelong Member,* Jaeseok Kim*** *Regular Member*

요약

본 논문에서는 휴대용 고속 무선 LAN 시스템에 적합한 저전력/저면적 MIMO-OFDM 기저대역 프로세서의 효율적인 하드웨어 구조를 제시한다. 고속 무선 LAN 시스템은 최대 수백 Mbps의 데이터 속도를 처리해야 하기 때문에 높은 시스템 클럭과 다중경로 구조를 사용하게 되는데, 이는 소모 전력과 구현 면적을 상승시키는 결과를 초래한다. 따라서 본 논문에서는 저전력으로 동작하면서도 동시에 하드웨어 부담을 줄인 고속 무선 LAN 시스템용 기저대역 프로세서의 하드웨어 구조를 제시한다. 이를 위해서 비트 병렬 처리 구조로 설계된 송신단 PLCP (TX-PLCP) 프로세서와 연산 복잡도를 효과적으로 감소시킨 심볼 검출기를 제안한다. 제안된 TX-PLCP 프로세서 구조는 비트 병렬 처리를 통해 동작 주파수를 감소시킴으로써 전력소모를 낮추는 효과를 얻을 수 있고, PMD 프로세서에서 가장 큰 면적을 차지하는 심볼 검출기는 수식 변형을 통해서 나눗셈 연산 및 제곱근 연산을 제거함으로써 저면적 설계를 가능하게 한다. 제안된 하드웨어 구조를 적용한 기저대역 프로세서는 Verilog HDL을 통해 설계 및 검증되었으며, 0.18um CMOS 공정을 통해 합성되었다. 합성결과, 병렬처리 구조를 적용한 TX-PLCP 프로세서는 비트 직렬 처리 구조에 비해 약 81% 감소된 전력에서 동작함을 확인하였고, 제안된 심볼 검출기는 나눗셈 및 제곱근 연산을 포함하는 심볼 검출 기법에 비해 약 18% 정도 하드웨어 복잡도가 감소함을 확인하였다.

Key Words : WLAN, MIMO-OFDM, low-power, low-complexity, baseband processor

ABSTRACT

This paper presents a low-power, low-complexity design and implementation results of a high speed multiple-input multiple-output orthogonal frequency division multiplexing (MIMO-OFDM) wireless LAN (WLAN) baseband processor. The proposed processor is composed of the physical layer convergence procedure (PLCP) processor and physical medium dependent (PMD) processor, which have been optimized to have low-power and reduced-complexity architecture. It was designed in a hardware description language (HDL) and synthesized to gate-level circuits using 0.18um CMOS standard cell library. As a result, the proposed TX-PLCP processor reduced the power consumption by as much as 81% over the bit-level operation architecture. Also, the proposed MIMO symbol detector reduced the hardware complexity by 18% over the conventional SQRD-based architecture with division circuits and square root operations.

* 본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었고, CAD Tool은 IDEC으로부터 지원받았음.

* 연세대학교 전기전자공학과 정보통신융SoC설계 연구실({adric, misuk_cho}@yonsei.ac.kr).

** 한국항공대학교 항공전자및정보통신공학부 조교수(yjung@kau.ac.kr), *** 연세대학교 전기전자공학과 교수(jaekim@yonsei.ac.kr)

논문번호 : KICS2008-09-385, 접수일자 : 2008년 9월 2일, 최종논문접수일자 : 2008년 10월 7일

I. 서 론

무선 통신의 발달로 인하여 모바일 기기를 통한 멀티미디어 서비스 시장이 확대되는 가운데, 고속 무선 데이터 전송에 대한 요구를 충족시킬 수 있는 방안으로 무선랜 (WLAN) 시스템이 주목을 받고 있다. IEEE 802.11a/b/g와 같은 현재의 무선랜 시스템은 최대 54Mbps에 이르는 전송 속도를 지원한다. 하지만, 초고속 인터넷, 무선 비디오/멀티미디어 서비스와 같은 최근의 어플리케이션은 더 높은 데이터 전송률을 요구하고 있다. 이러한 초고속 데이터 전송을 충족시킬 수 있는 차세대 무선랜 시스템으로써, 송수신단에 여러 개의 안테나를 사용하는 다중안테나(MIMO, Multiple-Input Multiple-Output) 방식과 다중경로 페이딩 채널환경에서 우수한 성능을 보이는 직교 주파수 분할 다중접속 (OFDM, Orthogonal Frequency Division Multiplexing) 방식을 채택한 IEEE 802.11n 표준이 IEEE 802.11 Working Group에 의해서 진행되고 있다^[1]. IEEE 802.11n 시스템은 최대 4개의 송수신 안테나를 사용하여 최고 600Mbps의 전송률까지 지원가능하다.

하지만, MIMO-OFDM 시스템은 다중안테나로 데이터를 전송하기 위해 병렬적으로 데이터를 처리하기 때문에, 데이터 전송률이 높아지는 대신 하드웨어 복잡도 역시 증가하는 단점을 가지고 있다. 따라서 MIMO-OFDM 시스템을 구현함에 있어서 낮은 하드웨어 복잡도를 갖도록 설계하는 것은 매우 중요한 고려사항이다. 특히 모바일 기기는 시스템 특성상 저전력 및 저면적 설계를 요구하는데, MIMO-OFDM 시스템의 하드웨어 복잡도는 데이터 경로 수에 따라 증가하기 때문에 일정 수 이상의 데이터 경로를 가지는 구조로 설계하는 것은 모바일 기기에 적합하지 않다. 일반적으로 모바일 기기의 크기를 고려했을 때 2개의 송수신 안테나를 갖는 구조가 가장 적절하며^[2], 이에 본 논문에서는 2x2 MIMO-OFDM 기반 무선랜 시스템을 위한 저전력, 저면적 기저대역 모뎀 프로세서의 하드웨어 구조를 제시한다.

제안하는 기저대역 프로세서는 비트단위로 동작하는 PLCP (Physical Layer Convergence Procedure) 프로세서와 심볼단위로 동작하는 PMD (Physical Medium Dependent) 프로세서로 구분된다. PLCP 프로세서의 경우 처리할 수 있는 데이터양이 시스템 클럭 주파수에 매우 의존적이기 때문에 고속의 전송률을 지원하려면 클럭 주파수를 증가시킬 수밖에

에 없다. 그러나 높은 클럭 주파수를 사용하게 되면 회로의 동적 파워 (dynamic power)가 증가하기 때문에 전체적인 전력소모가 커지게 된다^[3]. IEEE 802.11n 시스템에서는 2개의 데이터 경로를 가질 경우 최대 130Mbps의 데이터를 처리하기 위해서 160MHz의 속도로 동작해야 하므로, 프로세서의 클럭 주파수가 매우 높아져 전력소모가 증가하게 된다. 이러한 문제를 해결하기 위해서 본 논문에서는 낮은 클럭 주파수에서 동작할 수 있는 송신단 PLCP (TX-PLCP) 프로세서 구조를 제안한다. 수신단 PLCP (RX-PLCP)의 경우에는 동적 파워를 줄이기 위해 병렬 처리 구조를 사용하게 되면 면적이 크게 증가하는 문제가 있는 반면, TX-PLCP는 큰 면적 증가 없이 동적 파워를 줄일 수 있는 구조를 사용할 수 있다. 제안된 구조는 전송 모드별로 메모리를 가장 적게 사용할 수 있는 최적의 데이터 패킷을 찾아냄으로써 데이터 비트를 병렬적으로 묶어서 처리하게 된다. 이는 클럭 당 처리할 수 있는 데이터양이 증가하는 것을 의미하기 때문에, 결과적으로 클럭 주파수를 낮춰서 저전력으로 동작 가능한 프로세서를 설계할 수 있게 된다.

PMD 프로세서는 심볼단위로 동작하고, 높은 연산량을 요구하는 블록들로 구성되어 있다. 따라서 연산량 감소를 통해 소모전력 감소와 하드웨어 복잡도 감소의 효과를 동시에 얻을 수 있다. 특히, 전송률 증가를 위해 사용된 MIMO-SDM 심볼 검출기는 전체 시스템에서 하드웨어적으로 큰 비중을 차지하기 때문에 MIMO-SDM 심볼 검출기의 효율적인 설계는 전체 시스템의 하드웨어 부담을 줄이는 데 큰 역할을 한다^[4]. 따라서 본 논문에서는 저복잡도를 가지는 SDM (Space Division Multiplexing)^[5] 심볼 검출 기법을 제안한다. 제안하는 심볼 검출 기법은 성능적인 측면과 하드웨어 복잡도를 종합적으로 고려하여 MMSE-SQRD (Minimum Mean Square Error - Sorted QR Decomposition)^[6]을 기반으로 하는 기법을 선택하였고, 검출과정에서 수식변형을 통해서 연산량이 높은 나눗셈 및 제곱근 연산을 제거한다. 또한 다양한 근사화 기법을 적용하여 추가적인 연산 복잡도 감소 효과를 얻기 때문에, 기존의 방식보다 적은 연산량을 갖는 심볼 검출기를 구현할 수 있다.

본 논문의 구성은 다음과 같다. 먼저 II장에서는 2x2 MIMO-OFDM 기저대역 모뎀 프로세서의 구성에 대해서 언급한다. III장에서는 제안된 TX-PLCP 프로세서의 병렬 처리 구조 및 데이터 패킷을 살펴보고, IV장에서는 SDM-OFDM 심볼 검출기의 효율

적인 설계 방안을 제시한다. V장에서는 하드웨어 구현 결과를 분석하고, 마지막으로 VI장에서 결론을 맺는다.

II. MIMO-OFDM 기저대역 프로세서

그림 1은 2x2 MIMO-OFDM 기저대역 모델 프로세서의 블록도이다. 기저대역 프로세서는 비트 단위로 동작하는 PLCP 프로세서와 심볼 단위로 동작하는 PMD 프로세서로 구성된다.

송신모드일 경우, 기저대역 프로세서는 MAC 프로세서로부터 데이터 비트를 받아서 스크램블링과 FEC 인코딩을 수행한다. FEC 인코더는 R=1/2인 길쌈부호기 (convolutional encoder)와 펀처러 (puncturer)로 구성되어 있으며, 1/2, 2/3, 3/4, 5/6 등의 다양한 부호화율 (coding rate)을 제공한다. 인코딩을 거치면 채널 상에서 전송되는 데이터를 효과적으로 랜덤 분산하기 위해서 인터리빙을 수행하고, 심볼 변조를 통해서 다양한 형태의 심볼로 변환된다. 변조된 심볼들은 전송 모드에 따라서 SISO (Single-Input Single Output) 심볼로 전송되거나 또는 MIMO 인코더에서 STBC (Space-Time Block Coding)^[7]나 SDM 인코딩을 거치게 된다. 단, 송신 안테나가 최대 2개이기 때문에 2개의 데이터 경로를 사용하는 전송 모드에서는 STBC 인코딩이 불가능하다. 전송 모드에 따른 변조기법은 표 1에 정리되어 있다. 전송모드 0-7은 SISO 또는 STBC 모드로 동작할 수 있고, 전송모드 8-15는 SDM 모드로 동작한다. MIMO 인코더를 거친 데이터는 128-point IFFT를 거쳐서 시간대역 신호로 변조되고, 마지막으로 CP (Cyclic Prefix) 삽입과 파워 앰프의 비선형성 (non-linearity)에 대한 영향을 낮추기

표 1. 전송모드별 기저대역 파라미터

전송모드	변조기법	부호화율	전송률(Mbps)	MIMO
0	BPSK	1/2	6.5	STBC
1	QPSK	1/2	13.0	
2	QPSK	3/4	19.5	
3	16QAM	1/2	26.0	
4	16QAM	3/4	39.0	
5	64QAM	2/3	52.0	
6	64QAM	3/4	58.5	
7	64QAM	5/6	65.0	SDM
8	BPSK	1/2	13.0	
9	QPSK	1/2	26.0	
10	QPSK	3/4	39.0	
11	16QAM	1/2	52.0	
12	16QAM	3/4	78.0	
13	64QAM	2/3	104.0	
14	64QAM	3/4	117.0	
15	64QAM	5/6	130.0	

위한 clipping 과정을 거쳐서 RF단으로 전달된다.

수신모드일 경우에는 AGC와 시간, 주파수 동기를 거친 신호가 FFT를 거쳐서 주파수대역 신호로 복조된다. 채널추정부에서는 무선 환경에 대한 채널정보를 획득한 후, 그 값을 심볼 검출기에 전달한다. 심볼 검출기는 1개 혹은 2개의 안테나를 통해서 수신된 신호로부터 송신된 심볼들을 검출하는 기능을 하며, SISO, MRC (Maximal Ratio Combining), STBC, SDM모드를 모두 지원한다. 심볼 검출기로부터 검출된 심볼과 채널 상태 정보 (CSI: Channel State Information)를 전달받아 연판정 (soft decision) 심볼 복조과정을 거치면 데이터 비트가 복조된다. 이 데이

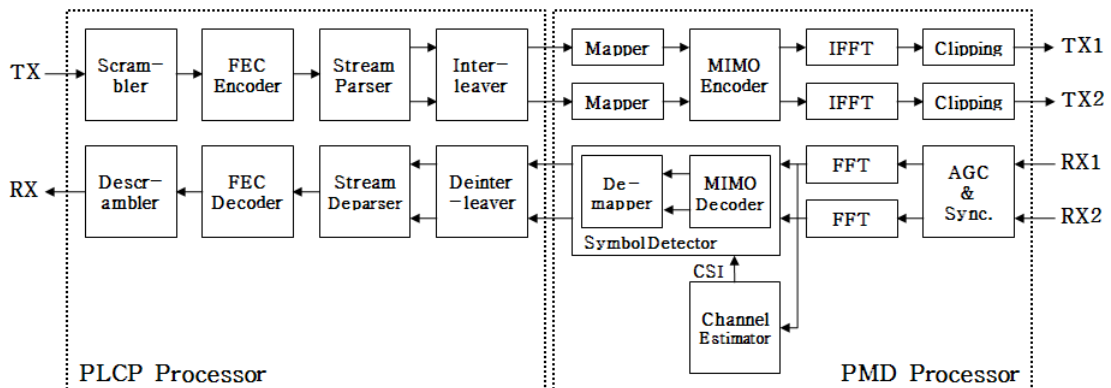


그림 1. 2x2 MIMO-OFDM 시스템용 기저대역 모델 프로세서의 블록도

터는 디인터리버와 비터비 디코더, 디스크램블러를 거쳐서 오류를 정정한 후에 MAC단으로 전달된다.

III. TX-PLCP의 비트 병렬 처리 구조

3.1 비트병렬처리 구조의 기본 입력 패턴

TX-PLCP 프로세서에서 데이터를 1비트씩 입력 받아서 처리하면 최대 전송률인 130Mbps를 지원하기 위해서 160MHz의 클럭주파수를 필요로 한다. 그러나 클럭주파수가 높아지면 타이밍 마진이 줄어들어 설계도 어려워질 뿐만 아니라, 회로의 동적 파워 증가로 인하여 전력소모도 커지게 된다. 하지만 데이터를 비트 단위가 아닌 바이트 (8비트) 단위로 입력받아서 처리할 수 있다면 40MHz의 클럭주파수로도 130Mbps의 전송률을 지원할 수 있게 동작할 수 있다. 그러나 바이트 단위로 동작하는 비트 병렬 처리 구조의 TX-PLCP를 설계하려면 각 블록의 제어 회로가 복잡해지고 추가적인 버퍼가 필요하기 때문에, 최적의 데이터 패턴 및 하드웨어 구조를 찾는 것이 중요하다.

바이트 단위로 입력 데이터를 받는 TX-PLCP 프로세서에서 만약 매 클럭마다 입력을 받게 되면, TX-PLCP 프로세서의 최종단인 인터리버의 출력 비트가 64QAM 모드일 때 최대 6비트이므로, 인터리버의 출력 버퍼에는 매 클럭마다 2비트씩 계속해서 데이터가 누적되게 된다. 이는 더 큰 사이즈의 버퍼를 사용해야함을 의미하기 때문에 하드웨어의 복잡도 증가를 야기한다. 이러한 현상을 방지하기 위해서는 입력 데이터와 출력 데이터의 속도를 동일하게 맞춰야 한다. 따라서 데이터가 입력되지 않는 기간이 필요한데, 3클럭마다 한 클럭씩 쉬도록 패턴을 구성하는 것이 하나의 방법이 될 수 있다. 이렇게 입력패턴을 구성하면 시스템 요구사항인 4us 이내에 한 OFDM심볼만큼의 데이터를 처리할 수 있으므로, 전체 시스템 스펙을 만족시키면서 클럭주파수를 낮출 수 있는 효율적인 설계를 할 수 있다.

3.2 TX-PLCP의 통합 구조

비트 병렬 처리 구조에서는 데이터가 기본 처리 단위인 바이트의 정수배가 아닐 경우가 발생할 수 있다. 따라서 TX-PLCP 프로세서의 기능 블록들을 개별적으로 설계하면, 제어 회로의 구조가 복잡하고 추가적인 데이터 버퍼도 필요하게 되어서 하드웨어 복잡도가 증가하는 문제가 발생한다. 그러므로 개별 블록들의 하드웨어 복잡도 증가를 최소화할 수 있

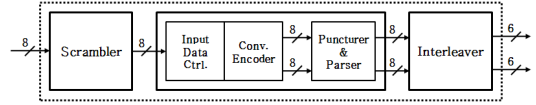


그림 2. 비트병렬처리 구조를 갖는 통합 TX-PLCP의 블록도

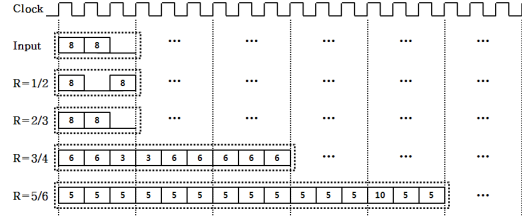


그림 3. 통합 TX-PLCP를 위한 데이터 패턴

는 구조가 필요하다. 이는 각각의 블록을 통합하여 단일 블록으로 설계하고, 입력부에 간단한 데이터 제어 블록을 사용함으로써 추가적인 버퍼 없이 해결할 수 있다. 인코더 블록부터 parser 블록까지를 통합시킨 TX-PLCP 프로세서의 블록도는 그림 2와 같다. 스크램블러로부터 바이트 단위로 데이터를 입력받으면 통합 TX-PLCP의 입력 데이터 제어 블록에서 전송모드에 따라서 알맞은 패턴으로 데이터를 펄처리와 parser 블록으로 전달하게 된다.

통합 TX-PLCP의 입력 제어 블록에서 데이터를 정리하여 나열하는 방식은 부호화율에 따라서 달라진다. 즉, 입력 데이터는 모든 전송모드에서 3클럭 단위로 8-8-0의 패턴을 갖게 되지만, 인코딩을 거친 출력 데이터는 부호화율마다 달라진다. 각 부호화율에 따른 데이터 정렬 패턴은 그림 3과 같다. 이와 같은 데이터 제어는 쉬프트 레지스터와 XOR 게이트를 사용하여 구현할 수 있는데, 그 블록도는 그림 4와 같다. 3클럭마다 입력되는 16비트의 데이터는 왼쪽의 레지스터에 저장되어 되고, 카운터가 증가하면 부호화율에 따라서 데이터 개수를 맞춰서 오른쪽 레지스터로 전달된다. 이 과정에서 인코딩 polynomial에 해당하는 위치에 있는 레지스터끼리 XOR를 취해서 출력하면 인코딩을 동시에 수행할 수 있다. 이처럼 데이터를 출력하면 각 출력마다 다음 클럭으로 넘어가는 데이터가 없어지기 때문에 puncturing 과정과 parsing 과정에서 바이트 단위로 데이터를 처리하기 위해서 개별적으로 저장해 둘 필요가 없어진다. 따라서 추가적인 버퍼가 필요 없어지므로 구현 면적의 감소 효과를 얻을 수 있다.

3.3 인터리버의 구조

비트 병렬 처리 구조를 가지는 인터리버의 효율

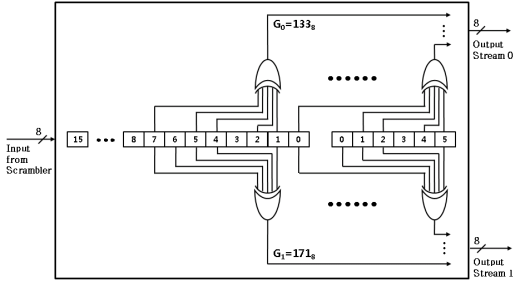


그림 4. 통합 TX-PLCP의 데이터 제어 및 encoding 회로

표 2. 각 경로 (stream)의 최대 입력 비트 사이즈

BPSK	QPSK		16QAM		64QAM		
1/2	1/2	3/4	1/2	3/4	2/3	3/4	5/6
8	8	4	8	4	6	6	6

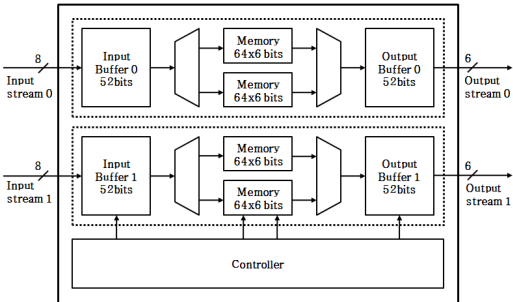


그림 5. 인터리버의 블록도

적인 구조는 그림 5와 같다. 최대 2개의 데이터 경로를 지원하고, 데이터는 매 클럭마다 인터리버의 각 경로 (stream)로 입력된다. 클럭 당 입력 비트 수는 최대 8비트이지만, 전송 모드에 따라 가변적이다. 표 2는 변조기법과 부호화율에 따른 인터리버 각 경로의 클럭 당 최대 입력 비트 수를 나타낸다.

인터리버는 입력 데이터를 섞는 동작을 수행하기 때문에 인접한 입력 데이터 비트는 동일한 클럭에 출력될 수 없다. 따라서 동일한 클럭에 출력되는 데이터를 셔플링 (shuffling)을 통해서 서로 인접하게 위치시켜야 한다. 주어진 구조에서의 셔플링은 식 (1)과 식 (2)의 두 단계에 걸쳐서 이루어진다.

$$j = ((i \bmod 52) \times 4) \bmod 52 + \lfloor (i \bmod 52) / 13 \rfloor + \lfloor (i \bmod 52) / 52 \rfloor + \lfloor i \bmod 52 \rfloor \times 52 \quad (1)$$

$$k = (j + s \times N_{CG}) \bmod 52 + \lfloor j / 52 \rfloor \times 52, \quad (2)$$

$$i, j, 0 \sim N_{CBPS}, \quad s = \begin{cases} 32, & \text{if } j > (N_{CBPS} / 52) \\ 28, & \text{else} \end{cases}$$

위의 두 식은 기본 인터리빙 수식을 52비트 단위로 동작하도록 변형한 것이며, N_{CBPS} 는 단위 심볼 당 부호화된 비트 수 (coded bit per symbol)를 의미한다. N_{CBPS} 가 전송모드에 따라서 52 또는 그 정수배의 값을 가지기 때문에 52비트 단위로 셔플링을 수행하는 것이 데이터를 가장 효과적으로 제어할 수 있다. 이렇게 52비트씩 셔플링된 데이터는 메모리에 저장된다.

하나의 OFDM 심볼을 구성하는 데이터 비트가 메모리에 모두 저장되면 메모리로부터 데이터를 읽어서 출력 버퍼에 저장한다. 연속된 심볼을 처리하기 위해서 각 경로 당 두 개의 메모리가 필요하다. 첫 번째 메모리에서 데이터를 읽어서 출력 버퍼에 저장하는 동안 새로 입력되는 일련의 데이터는 입력 버퍼에서 셔플링이 된 다음에 다른 메모리에 저장된다. 메모리의 크기는 변조기법이 64QAM일 때를 기준으로 두 데이터 경로에서 입력되는 연속된 심볼을 처리할 수 있도록 할당되었다.

식 (1)과 식 (2)를 통해서 셔플링을 수행하게 되면, 동일한 시간에 출력되어야 하는 데이터는 메모리의 두 라인에 나누어 저장된다. 출력 버퍼에서는 메모리로부터 12비트의 데이터를 읽어서 그 중에서 현재 출력 될 6비트를 선택하게 된다. 출력 버퍼에서 메모리로부터 읽어온 데이터를 비트끼리 상호교환 (flipping)한 후에 심볼 변조기 (mapper)로 출력시키면 인터리버의 기능을 모두 수행하게 된다.

IV. 효율적인 SDM-OFDM 심볼 검출 기법

데이터를 컨트롤하는 비중이 높은 PLCP 프로세서의 동작에 비해서 PMD 프로세서는 신술 연산이 대부분이기 때문에 연산량을 줄이는 것이 복잡도 감소 측면이나 전력소모 측면에서 훨씬 효과적이다. 특히, 연산량이 많은 MIMO-SDM 심볼 검출기의 효율적인 설계는 프로세서 전체의 복잡도 감소와 직결되는 중요한 고려사항이다.

4.1 시스템 모델

송수신 안테나가 각각 2개씩인 2x2 MIMO-OFDM 시스템의 k-번째 부반송파에 대한 SDM 심볼 수신 벡터는 식 (3)과 같이 표현할 수 있다.

$$\mathbf{Y} = \mathbf{H} \cdot \mathbf{X} + \mathbf{N} = \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_2 \end{bmatrix} \cdot \mathbf{X} + \mathbf{N} = \begin{bmatrix} h_{11}^k & h_{12}^k \\ h_{21}^k & h_{22}^k \end{bmatrix} \cdot \begin{bmatrix} x_1^k \\ x_2^k \end{bmatrix} + \begin{bmatrix} n_1^k \\ n_2^k \end{bmatrix} \quad (3)$$

식 (3)에서 $\mathbf{Y} = [y_1^k \ y_2^k]^T$ 는 수신된 심볼 벡터를 의미하고, \mathbf{H} 는 2x2 페이딩 채널을 의미한다. \mathbf{h}_1 , \mathbf{h}_2 는 \mathbf{H} 의 열벡터를 의미하고, \mathbf{H} 의 성분 h_{ij}^k 는 i -번째 수신 안테나와 j -번째 송신 안테나 사이의 주파수 응답을 의미한다. $\mathbf{X} = [x_1^k \ x_2^k]^T$ 는 서로 독립된 두 안테나를 통한 송신 심볼 벡터이고, 파워를 1로 정규화한 출력 신호이다. \mathbf{N} 은 평균은 0이고, σ^2 의 분산을 가지는 AWGN (Additive White Gaussian Noise)을 의미한다.

4.2 SQRD 기반 SDM 심볼 검출 기법

최근에, 하드웨어 복잡도를 낮추고 연산 속도를 향상시킨 MIMO-SDM 심볼 검출 기법들이 많이 연구되었다. 그 중에서 SQRD (sorted QR decomposition) 기반의 심볼 검출 기법^[6]은 반복된 역행렬 연산을 수행하지 않기 때문에 복잡도가 높지 않을 뿐만 아니라, MMSE (minimum mean square error)와 조합되었을 경우 좋은 성능을 유지할 수 있기 때문에 하드웨어로 구현하기에 매우 적합한 기법이다.

SQRD 기반의 심볼 검출 기법은 표 3과 같다. 여기서 $Q(\cdot)$ 는 성상도에 따른 적절한 값으로 양자화하는 함수이다. 기존의 SQRD 기반 검출 기법은 QR 분해과정과 심볼 추정과정에서 나눗셈 연산을 필요로 하고, 놈 (norm) 값을 구하는 과정에서는 제공근 연산을 필요로 한다. 나눗셈기 회로의 경우, 논리 지연 시간이 클 뿐만 아니라 하드웨어 복잡도도 크기 때문에 나눗셈 회로를 포함한 SDM 심볼

검출기를 설계하는 것은 쉽지 않은 일이다. 이러한 이유 때문에, 수정된 성상도 상에서 심볼을 추정함으로써 나눗셈 연산을 제거한 심볼 검출 기법이 [8]에서 제안되었다. 따라서 [8]의 기법을 사용하여 심볼 추정과정에서 필요로 하는 나눗셈 연산을 제거할 수 있다. 그러나 QR 분해과정에서 여전히 나눗셈 연산을 필요로 하고, 놈값을 구하기 위해서 하드웨어 복잡도가 큰 제공근 회로를 포함해야 하기 때문에, 주어진 심볼 검출기를 설계하기에는 여전히 큰 하드웨어 부담이 있다.

4.3 저복잡도 심볼 검출 기법

본 논문에서 사용한 검출 기법은 QR 분해과정의 수식 변형을 통해서 나눗셈 연산과 제공근 연산을 제거함으로써 저복잡도를 가지면서 성능에는 차이가 없는 심볼 검출기 구현을 가능하게 한다.

QR 분해과정의 수식 변형을 위해서 새로운 2x2 보조 벡터 \mathbf{A} 를 식 (4)와 같이 정의한다.

$$\mathbf{A} = \begin{bmatrix} r_{11}^k & 0 \\ 0 & \|\mathbf{h}_1\|^2 r_{22}^k \end{bmatrix} \tag{4}$$

그러면 또 다른 2x2 보조 벡터 $\mathbf{B} = \mathbf{Q} \cdot \mathbf{A}$ 를 다음과 같이 나타낼 수 있다.

$$\mathbf{B} = [\mathbf{b}_1 \ \mathbf{b}_2] = \begin{bmatrix} \mathbf{q}_1 r_{11}^k & \mathbf{q}_2 \|\mathbf{h}_1\|^2 r_{22}^k \\ \mathbf{h}_1 & \|\mathbf{h}_1\|^2 \mathbf{h}_2 - \mathbf{h}_1^H \mathbf{h}_2 \mathbf{h}_1 \end{bmatrix} \tag{5}$$

식 (5)에서 $\mathbf{q}_1, \mathbf{q}_2$ 는 \mathbf{Q} 의 열벡터를 의미한다. 심볼을 검출하기 위해 \mathbf{B}^H 를 식 (3)의 양변에 곱해서 정리하면 다음과 같다.

$$\begin{aligned} \tilde{\mathbf{Z}} &= \mathbf{B}^H \cdot \mathbf{Y} \\ &= \mathbf{B}^H \cdot (\mathbf{H} \cdot \mathbf{X} + \mathbf{N}) \\ &= (\mathbf{Q} \cdot \mathbf{A})^H \cdot (\mathbf{Q} \cdot \mathbf{R}) \cdot \mathbf{X} + \tilde{\mathbf{N}} \\ &= \mathbf{A}^H \cdot \mathbf{R} \cdot \mathbf{X} + \tilde{\mathbf{N}} \\ &= \begin{bmatrix} (r_{11}^k)^2 & r_{11}^k r_{12}^k \\ 0 & \|\mathbf{h}_{11}\|^2 (r_{22}^k)^2 \end{bmatrix} \cdot \begin{bmatrix} x_1^k \\ x_2^k \end{bmatrix} + \begin{bmatrix} n_1^k \\ n_2^k \end{bmatrix} \end{aligned} \tag{6}$$

식 (6)에서 각 성분은 다음과 같이 구할 수 있다.

$$(r_{11}^k)^2 = \|\mathbf{h}_1\|^2 \tag{7}$$

$$r_{11}^k r_{12}^k = \mathbf{h}_1^H \mathbf{h}_2 \tag{8}$$

$$\|\mathbf{h}_1\|^2 (r_{22}^k)^2 = \|\mathbf{h}_1\|^2 \|\mathbf{h}_2\|^2 - |\mathbf{h}_1^H \mathbf{h}_2|^2 \tag{9}$$

위의 식 (7)-(9)에서 보는 바와 같이, 제공근 연산과 나눗셈 연산이 모두 제거된 형태로 QR 분해

표 3. SQRD 기반 심볼 검출 기법^[6]

과정	동작
1	채널 정렬 $\hat{\mathbf{H}} = [\mathbf{h}_2 \ \mathbf{h}_1], p=1$ (if $\ \mathbf{h}_1\ > \ \mathbf{h}_2\ $) $\hat{\mathbf{H}} = [\mathbf{h}_1 \ \mathbf{h}_2], p=0$ (otherwise)
2	QR 분해 $\mathbf{H} = \mathbf{Q} \cdot \mathbf{R} = \begin{bmatrix} q_{11}^k & q_{12}^k \\ q_{21}^k & q_{22}^k \end{bmatrix} \cdot \begin{bmatrix} r_{11}^k & r_{12}^k \\ 0 & r_{22}^k \end{bmatrix}$
3	$\mathbf{Q}^H \cdot \mathbf{Y}$ $\mathbf{Z} = \mathbf{Q}^H \cdot \mathbf{Y} = \mathbf{R} \cdot \mathbf{X} + \mathbf{N}'$ $\begin{bmatrix} z_1^k \\ z_2^k \end{bmatrix} = \begin{bmatrix} r_{11}^k & r_{12}^k \\ 0 & r_{22}^k \end{bmatrix} \cdot \begin{bmatrix} x_1^k \\ x_2^k \end{bmatrix} + \begin{bmatrix} (n_1^k)' \\ (n_2^k)' \end{bmatrix}$
4	x_2 추정 $\hat{x}_2^k = Q(z_2^k, r_{22}^k)$
5	SIC $\hat{z}_1^k = z_1^k - r_{12}^k \hat{x}_2^k$
6	x_1 추정 $\hat{x}_1^k = Q(\hat{z}_1^k, r_{11}^k)$
7	재정렬 p 값에 의거하여 \hat{x}_1^k, \hat{x}_2^k 순서 재정렬

표 4. 중복잡도 심볼 검출 기법

과정	동작
1	채널 정렬 $\hat{\mathbf{H}} = [\mathbf{h}_2 \ \mathbf{h}_1], p = 1$ (if $\ \mathbf{h}_1\ > \ \mathbf{h}_2\ $) $\hat{\mathbf{H}} = [\mathbf{h}_1 \ \mathbf{h}_2], p = 0$ (otherwise) ($\ \mathbf{h}_i\ \approx \mathbf{h}_{1i} + \mathbf{h}_{2i} $)
2	변형된 QR 분해 $\mathbf{B} = \mathbf{Q} \cdot \mathbf{A} = \begin{bmatrix} q_{11}^k & q_{12}^k \\ q_{21}^k & q_{22}^k \end{bmatrix} \cdot \begin{bmatrix} r_{11}^k & 0 \\ 0 & \ \mathbf{h}_1\ ^2 r_{22}^k \end{bmatrix}$
3	$\mathbf{B}^H \cdot \mathbf{Y}$ $\begin{bmatrix} \hat{z}_1^k \\ \hat{z}_2^k \end{bmatrix} = \begin{bmatrix} (r_{11}^k)^2 & r_{11}^k r_{12}^k \\ 0 & \ \mathbf{h}_1\ ^2 (r_{22}^k)^2 \end{bmatrix} \cdot \begin{bmatrix} x_1^k \\ x_2^k \end{bmatrix} + \begin{bmatrix} (n_1^k)^H \\ (n_2^k)^H \end{bmatrix}$
4	x_2 추정 $\hat{x}_2^k = Q(\hat{z}_2^k, \ \mathbf{h}_1\ ^2 r_{22}^k)$
5	SIC $\hat{z}_1^k = \hat{z}_1^k - (r_{11}^k r_{12}^k) \hat{x}_2^k$
6	x_1 추정 $\hat{x}_1^k = Q(\hat{z}_1^k, (r_{11}^k)^2)$
7	재정렬 p 값에 의거하여 \hat{x}_1^k, \hat{x}_2^k 순서 재정렬

과정이 진행되는 것을 알 수 있다. 이는 새로 정의한 벡터인 \mathbf{A} 와 \mathbf{B} 를 통해서 놈의 제곱을 이용한 수식 전개가 가능하고, 또한 나눗셈 연산의 분모로 사용되는 성분들이 곱해진 형태로 최종 결과를 도출했기 때문이다. 따라서 변형된 성상도 기반의 심볼 추정 기법^[8]을 적용하면, QR 분해 이후의 과정은 기존의 심볼 검출 기법과 동일하게 진행된다. 표 4는 본 논문의 검출 과정을 정리한 것이다.

표 4의 정렬과정(과정 1)에서 사용하는 놈값은 단순히 크기를 비교하기 위함이기 때문에 식 (10)과 같이 실수부와 허수부의 절대값의 합으로 근사화할 수 있다.

$$\|\mathbf{h}_1\| \approx |Re(\mathbf{h}_1)| + |Im(\mathbf{h}_1)| \quad (10)$$

심볼 검출 이후에는 연판정 복조 (soft demapping) 과정을 거치는데, LLR (Log Likelihood Ratio)을 기반으로 하는 기법이 가장 좋은 성능을 보여준다^[9]. 이 때 검출된 심볼과 기준 심볼과의 유클리디안 거리를 구하는 과정에서 놈 연산을 수행하는데, 정확한 거리를 비교해야 하기 때문에 식 (10)과 같이 근사화시키면 오차가 커지게 된다. 하지만 식 (11)과 같은 근사화 방식을 사용하면 성능 열화가 거의 없으면서 연산량을 감소시킬 수 있기 때문에 중복잡도 설계에 적합한 방식이다^[10].

$$\|\mathbf{y}_n\| \approx \frac{3}{8} (|Re(\mathbf{y}_n)| + |Im(\mathbf{y}_n)|) + \frac{5}{8} MAX(|Re(\mathbf{y}_n)|, |Im(\mathbf{y}_n)|) \quad (11)$$

4.4 연산량 비교 및 근사화 성능 평가

앞에서 언급한 것처럼, 제안된 SDM 심볼 검출 기법은 제공근 연산과 나눗셈 연산을 완전히 제거하기 때문에 기존의 SQRD 기반 검출 기법보다 적은 연산 복잡도를 가진다. 우선 채널 벡터를 정렬하는 과정에서 놈 연산 근사화를 통해서 곱셈 연산이 모두 덧셈 연산으로 대체되고, QR 분해과정에서 제공근 연산과 나눗셈 연산이 제거되어 연산량이 크게 감소된다. 또한, LLR 기반 연판정 복조 과정의 놈 연산 근사화는 연산 복잡도를 추가로 감소시킨다. 표 5는 기존의 검출 기법과 제안된 기법의 단계별 연산량을 비교한 것이다. 표에서 알 수 있듯이, 제안된 기법은 기존의 기법에서 필요한 제공근 연산 및 나눗셈 연산이 모두 제거되기 때문에, 효율적인 하드웨어 구현이 가능하다.

또한 놈 연산 근사화를 통한 성능 열화 정도를 평가하기 위해서 모의실험을 수행하였다. MIMO-SDM 기법이 사용된 전송모드 8~15에 대해서 시간 및 주파수 동기는 완벽하다는 가정 하에 1000바이트의 패킷을 사용하여 모의실험을 수행한 결과, 모

표 5. 연산 복잡도 비교

과정	SQRD 기반 기법			제안 기법		
	곱셈	나눗셈	제공근	곱셈	나눗셈	제공근
채널 정렬	8	-	-	-	-	-
QR 분해	24	8	2	34	-	-
LLR 기반 복조	8	-	-	6	-	-
합계	40	8	2	40	-	-

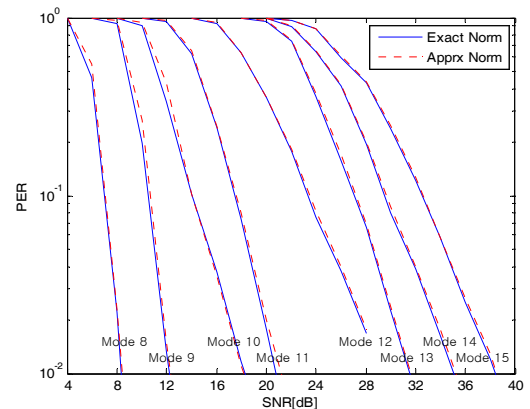


그림 6. 놈 연산 근사화에 의한 성능 열화 평가

든 전송모드에서 높 연산 근사화에 의한 성능 열화가 거의 없는 것으로 확인되었다. 각 모드별 SNR 대비 패킷오류율(PER, Packet Error Rate)은 그림 6과 같다. 모든 전송모드에서 정확한 높 연산을 수행했을 경우와 근사화된 높 연산을 수행했을 경우에 성능 차이가 거의 없는 것을 확인할 수 있다.

V. 하드웨어 구현 결과

비트 병렬 처리 구조를 갖는 TX-PLCP 프로세서와 제안된 심볼 검출 기법을 적용한 PMD 프로세서로 구성된 MIMO-OFDM 무선랜 기저대역 프로세서에서는 Verilog HDL을 이용하여 설계하고 매그나 칩 0.18um CMOS 공정을 이용하여 합성하였다.

표 6은 비트 병렬 처리 TX-PLCP 프로세서의 전력소모 측정 및 논리 합성 결과이다. 파워 시뮬레이션은 Synopsys사의 Power Compiler™를 사용하여 수행하였고, 클럭 주파수는 40MHz로 설정하였다. 결과를 비교하기 위해서 비트 직렬 처리 TX-PLCP 프로세서를 설계하였고, 동일한 전송률을 지원하도록 160MHz에서 동작시켰다. 64QAM 심볼로 구성된 1,000개의 패킷을 입력하여 전력소모를 비교한 결과, 제안된 구조는 비트 직렬 처리 구조에 비해서 약 80.5%의 감소 효과를 얻었다.

제안된 MIMO 심볼 검출기의 합성 결과는 표 7

표 6. 비트 병렬 처리 TX-PLCP의 구현 결과

	직렬 구조 (160MHz)	제안 구조 (40MHz)	증감율
전력소모	23.1mW	4.5mW	- 80.5%
게이트 수	27K	29K	+ 7.4%

표 7. MIMO 심볼 검출기의 구현 결과

	SQRD 구조 (w/ div. & sqrt)	제안 구조 (w/o div. & s qrt)	감소율
게이트 수	286K	234K	18.2%

표 8. 제안된 시스템의 게이트 수 및 전력소모 결과

게이트 수	PLCP	252 K
	PMD	601 K
	기타	80 K
	합계	933 K
전력 소모 (mW)	TX	62
	RX	284

에 정리되어 있다. 제안된 심볼 검출기는 나눗셈 및 제곱근 연산 제거와 높 연산 근사화 과정을 통해서 하드웨어 복잡도를 줄인 것이므로, 해당 연산들을 포함하는 심볼 검출기와의 복잡도 비교를 수행하였다. 그 결과, 연산정 복잡기를 포함하는 제안 심볼 검출기는 234K개의 게이트로 구현되었고, 약 18%의 하드웨어 복잡도 감소 효과를 얻었다.

기저대역 프로세서 전체의 구현 결과는 표 8과 같다. 제안된 구조를 적용한 결과, 총 933K의 게이트로 구현되었고, 전력소모는 송신시 62mW, 수신시 284mW로 측정되었다.

또한 제안된 구조로 설계된 프로세서의 실시간 기능 검증을 위해서, 기저대역 프로세서를 40MHz에서 동작하는 FPGA 칩을 사용하여 구현하였다. 테스트 결과, 칩이 정상적으로 동작함을 확인하였고, 모의실험을 통한 출력벡터와 FPGA 칩을 거친 출력 데이터가 일치함을 확인하였다.

VI. 결론

본 논문에서는 휴대용 고속 무선 LAN 시스템을 위한 MIMO-OFDM 기저대역 프로세서의 저전력, 저면적 하드웨어 구조를 제시하였다. 전력 소모를 줄이기 위해서 클럭 주파수를 기존의 160MHz에서 40MHz로 낮춘 비트 병렬 구조의 TX-PLCP 프로세서를 제안하였다. 또한 MIMO-OFDM 시스템에서 하드웨어 복잡도 측면에서 큰 비중을 차지하는 MIMO 심볼 검출기를 효율적으로 설계하기 위하여 검출기법 수식을 변형함으로써 연산 복잡도를 낮춘 심볼 검출기 구조를 제안하였다. 제안된 기저대역 프로세서는 HDL로 설계하였고, CMOS 0.18um 공정으로 합성되었다. 그 결과, 40MHz에서 동작하는 비트 병렬 구조의 TX-PLCP 프로세서는 160MHz에서 동작하는 직렬 구조보다 약 81%의 전력 소모 감소 효과를 얻었고, 제안된 구조로 설계된 MIMO 심볼 검출기는 나눗셈 및 제곱근 연산을 포함한 구조보다 하드웨어 복잡도가 약 18%만큼 감소하는 것을 확인하였다. 따라서 본 논문에서 제안한 기저대역 프로세서는 저전력, 저면적 설계가 중요한 고려사항으로 인식되는 휴대용 고속 무선 LAN 시스템에서 활용할 수 있는 유용한 구조로 판단된다.

참고 문헌

[1] IEEE 802.11 Task Group n, "Draft Amend-

ment-Part II: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Enhancements for Higher Throughput,” Mar. 2006.

- [2] W. Sun, “Maximizing MIMO effectiveness by multiplying WLAN radios”, www.atheros.com
- [3] A. P. Chandrakasan and R.W. Brodersen, “Minimizing Power Consumption in Digital CMOS Circuits,” *Proceedings of the IEEE*, Vol.83, No.4, Apr. 1995.
- [4] S. Noh, Y. Jung, S. Lee and J. Kim, “Low-Complexity Symbol Detector for MIMO-OFDM Based Wireless LANs,” *IEEE Trans. Circuits Syst. II*, Vol.53, No.12, pp.1403-1407, Dec. 2006.
- [5] P. W. Wolniansky, G. J. Foschini, G. D. Golden and R. A. Valenzuela, “V-BLAST: an architecture for realizing very high data rates over the rich-scattering wireless channel,” in *Proc. ISSSE*, 1998.
- [6] D. Wubben, R. Bohnke, J. Rinas, V. Kuhn, K. D. Kammeyer, “Efficient algorithm for decoding layered space-time codes,” *IEE Electron. Lett.*, Vol.37, No.22, pp.1348-1350, Oct. 2001.
- [7] V. Tarokh, H. Jafarkhani, and A. R. Calderbank, “Space-time block codes from orthogonal designs,” *IEEE Trans. Inform. Theory*, Vol.45, No.5, pp.1456-1467, Jul. 1999.
- [8] E. Cavus, B. Daneshrad, “A computationally efficient algorithm for space-time block decoding,” in *Proc. ICC*, Vol.4, pp.1157-1162, Jun. 2001.
- [9] F. Tosato, P. Bisaglia, “Simplified soft-output demapper for binary interleaved COFDM with application to HIPERLAN/2,” in *Proc. ICC*, vol.2, pp.664-668, May 2002.
- [10] A. Adjoudani et al, “Prototype experience for MIMO BLAST over third-generation wireless system,” *IEEE J. on Selected Areas in Comm.*, Vol.21, No.3, Apr. 2003.

임 준 하 (Junha Im)

정회원



2005년 연세대학교 전기전자공학과 졸업
 2007년 연세대학교 전기전자공학과 석사
 2007년~현재 연세대학교 전기전자공학과 박사과정
 <관심분야> MIMO/OFDM 시스템, 모뎀SoC설계

조 미 숙 (Misuk Cho)

정회원



2003년 대전대학교 전자공학과 졸업
 2004년~현재 연세대학교 전기전자공학과 석박사통합과정
 <관심분야> 통신시스템의 SoC 구현, 신호처리

정 윤 호 (Yunho Jung)

중신회원



1998년 연세대학교 전자공학과 졸업
 2000년 연세대학교 전기전자공학과 석사
 2005년 연세대학교 전기전자공학과 박사
 2005년~2007년 삼성전자 책임연구원

2007년~2008년 연세대학교 연구교수

2008년~현재 한국항공대학교 조교수

<관심분야> MIMO/OFDM 통신시스템, VLSI신호처리, 모뎀SoC설계

김 재 석 (Jaeseok Kim)

정회원



1977년 연세대학교 전자공학과 졸업
 1979년 연세대학교 전기전자공학과 석사
 1988년 Rensselaer Polytechnic Institute, NY. 박사
 1988년~1993년 AT&T Bell Lab. 연구원

1993년~1996년 한국전자통신연구원 책임연구원

1996년~현재 연세대학교 전기전자공학과 교수

<관심분야> 통신SoC설계, 고속멀티미디어IP설계