

# MIMO-OFDM 시스템을 위한 V-BLAST의 설계 및 구현

최 용 우, 박 인 철

한국과학기술원 전자전산학과 전기 및 전자공학전공  
전화 : 042-869-4406 / 핸드폰 : 011-9418-4576

## Design and Implementation of V-BLAST for MIMO-OFDM Systems

Yong-Woo Choi and In-Cheol Park

Division of Electrical Engineering, Electrical Engineering and Computer Science, KAIST  
E-mail : ywchoi@ics.kaist.ac.kr

### Abstract

This paper describes a VLSI implementation of BLAST detection for MIMO-OFDM systems. To achieve high speed requirement, we propose the fully pipeline architecture for BLAST structure. This design is implemented using 0.18 $\mu$ m CMOS technology. For a 4-transmit and 4-receive antennas system, it takes 7.5 $\mu$ s to calculate nulling vector and detection order from 48 channel matrixes.

### I. 서론

지난 수년간 근거리 무선 통신용 모뎀은 지속적인 기술의 발전에 힘입어 그림 1과 같은 발전 양상을 보여 왔다.[1] 이러한 발전의 단계 중 3세대를 거치고 있는 현재, 지속적인 멀티미디어 서비스에 대한 요구는 지금의 최고 전송속도에서의 약 5배 이상의 향상을 기대하고 있다.

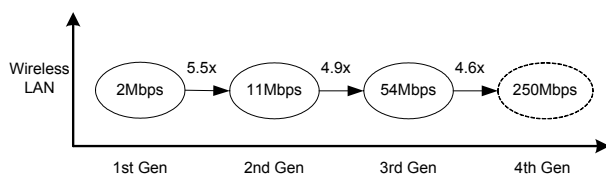


그림 1 : 무선 LAN 표준의 진화

이러한 전송속도의 증가를 이룰 기반 기술 중 현재 가장 주목을 받고 있는 분야가 다중 안테나를 사용하는 MIMO(Multiple Input Multiple Output) 시스템이다. 현재 이 MIMO 시스템은 3가지 계열로 나누어지고 있다.[2] 첫째로 Spatial diversity를 최대화하여 파워 효율을 증가시키는 것을 목적으로 하는 것으로 delay diversity, STBC(space-time block codes)와 STTC (space-time trellis codes)를 포함하는 계열이 있다. 둘째로, 송신기에서 채널의 정보를 가지고 있는 경우로 채널 계수 행렬을 SVD(singular value decomposition)을 사용하여 분해한 후 송신기와 수신기에 각각 필터를 사용하여 채널의 전송속도를 극대화시키는 방법이다. 마지막으로 전송속도를 증가시키기 위해 계층별 접근을 시도하는 경우로 BLAST(Bell Laboratories Layered Space-Time) 계열이 있다.

OFDM(Orthogonal Frequency Division Multiplexing)은 IEEE 802.11a LAN standard에서 채택되어 사용되고 있는 기술로 신호 전달시 발생하는 ISI(Inter Symbol Interference)를 감소시키는 역할을 수행한다.[3]

본 논문에서는 무선 통신용 모뎀에 MIMO 시스템을 적용한 MIMO-OFDM의 구현을 위해 기존의 무선통신용 모뎀을 수정한 시스템을 제시한다. 다음으로 BLAST 알고리즘과 이를 고속으로 처리하기 위한 파이프라인 방식의 하드웨어 구조를 제시한다. 다음으로 하드웨어 블록의 설계과정에 대해서 제시하고 마지막으로 최종 구현 후 얻어진 속도와 면적을 기술한다.

## II. 시스템과 알고리즘

OFDM 방식을 사용하는 IEEE 802.11a에 MIMO (Multiple Input Multiple Output)을 적용한 시스템은 그림 2와 같다.

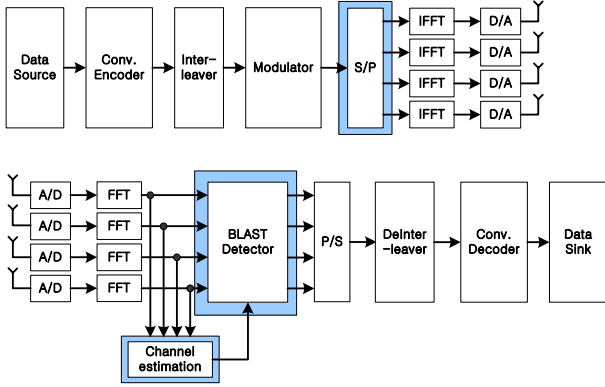


그림 2 MIMO-OFDM 시스템

기존의 시스템에서 새롭게 추가된 부분은 송신기에 S/P(Serial to Parallel) 블록과 수신기에 채널 추정 블록과 BLAST Detector 블록이다. 이중 본 논문은 가장 핵심적인 부분이 되는 BLAST Detector 블록에 대해서만 고려한다.

BLAST detector를 위해 제시되는 다중 안테나 시스템(송신단의 안테나 수  $M$ 개, 수신단의 안테나 수  $N$ 개)의 baseband equivalent model은 다음과 같다.

$$r = Hs + v \quad (1)$$

여기서,  $s = [s_1, s_2, \dots, s_M]^T$ 은 전송된 심볼 벡터로, 각각의 값은 QPSK와 같은 complex constellation로부터 독립적으로 얻어진 값이다.  $r = [r_1, r_2, r_3, \dots, r_M]^T$ 은 수신 심볼 벡터,  $v = [v_1, v_2, v_3, \dots, v_M]^T$ 는 잡음 벡터이다. 다음으로 다중 산란 환경을 가정할 경우,  $H$ 는 복소수값을 원소로 같은  $N \times M$  행렬이 된다. 그리고 여기에 OFDM을 추가하면 OFDM에서 사용하는 subcarrier의 수만큼 독립적인 채널이 생성된다고 보면 된다. 즉,  $N$ -point FFT를 사용하는 OFDM인 경우  $N$ 개의 독립적인 위 (1)식의 시스템이 형성되는 것으로 간주된다. 다음으로 본 논문의 이후에서  $H$ 는 완전하게 수신기에서 채널 추정을 통해서 알고 있는 것으로 가정한다.

MIMO 시스템의 구현방법으로 BLAST 알고리즘[4]이 제안된 후 실제 하드웨어 구현에 적합한 형태로의 알고리즘의 변화가 여러 논문에 의해서 시도되었다. 이중 현재까지 하드웨어로 실제 구현된 적이 있는 알고리즘으로 Hassibi's의 알고리즘(Square Root Algorithm)이 있다.[5][6] Hassibi에 의해서 제안된 알고리즘은 아래와 같다.

1)  $P^{1/2}$ ,  $Q_a$ 의 계산 (for  $i = 1, 2, \dots, N$ )

$$\begin{pmatrix} 1 & (H)_i A_{i-1}^{M \times M} \\ 0^{M \times 1} & A_{i-1}^{M \times M} \\ -e_i^{N \times 1} & B_{i-1}^{N \times M} \end{pmatrix} \Theta_i = \begin{pmatrix} \times 0^{1 \times M} \\ \times A_i^{M \times M} \\ \times B_i^{N \times M} \end{pmatrix}$$

$$A_0 = \beta I^{M \times M}, B_0 = 0^{N \times M}$$

여기서  $e_i$ 는 크기가  $N$ 인 단위행렬,  $\Theta_i$ 는 unitary transformation matrix,  $\beta$ 는 전송 심볼당 SNR의 제공근이다.  $N$ 번의 반복 수행 후에 최종 결과는 다음과 같다.

$$P^{1/2} = A_N, Q_a = B_N$$

2) 최적의 검출 순서와 nulling vector의 결정 (for  $i = M, M-1, \dots, 1$ )

-  $P_i^{1/2}$ 의 행(row)중 minimum length를 찾고, 이것을 마지막 행과 바꾼다.  $s$ 에 대해서 동일한 작업을 수행한다.

$$- P_i^{1/2} \Sigma_i = \begin{pmatrix} P_{i-1}^{1/2} & \times^{(i-1) \times 1} \\ 0^{1 \times (i-1)} & p_i \end{pmatrix}$$

-  $Q_a = Q_a \Sigma_i$ ,  $i$ 번째 전송신호에 대한 nulling vector는  $w_i = p_i(Q_a)_i^*$ 이다.

3) Nulling과 Cancellation을 수행 (for  $i = M, M-1, \dots, 1$ )

-  $y_i = w_i r$ 를 계산한 후,  $i$ 번째 전송신호를 아래와 같이 추정한다.

$$\hat{s}_i = \arg \min_{\hat{s} \in \Omega} \|\hat{s} - y_i\|^2$$

- 남아있는 신호에서 검출된 신호에 의한 간섭을 제거한다.

$$r = r - \hat{s}_i(H)_i$$

## III. VLSI 아키텍처

Square Root Algorithm에 기반한 하드웨어 구조는 [5]에 제시되어 있다. 그러나 [5]에서 제시한 구조는 3세대 이동통신에서 HSDPA (High Speed Downlink Packet Access) system을 위해서 설계된 것으로 송신 안테나와 수신 안테나가 각각 4개일 때 하나의 채널행렬로부터  $P^{1/2}, Q_a$ 를 계산하는데 총  $8\mu s$ 가 소요된다. 이것은 64 point FFT를 사용하는 OFDM의 경우 총  $512 \mu s$ 의 시간동안 들어오는 data symbol의 약 절반을 저장하고 있어야 한다. 이것은 FFT로부터 얻는 신호의 bit width를 11bit로 가정할 경우, IEEE802.11a에서 한

데이터의 심볼 구간이 4 $\mu$ s이고, 64의 subcarrier 중 48개가 데이터 전송을 위해 사용되는 것을 고려할 때 약 66kbit의 SRAM을 필요로 하게 된다. 따라서, 고속으로 채널 행렬로부터  $P^{1/2}$ ,  $Q_a$  및 nulling vector, detection order를 얻어내는 것이 요구된다. 고속처리를 위해서 기존에 제시되었던 구조를 파이프라인 형태도 재구성했다. 그림 3은 송신기와 수신기의 안테나 수가 각각 4개인 경우의 이전구조와 새롭게 제안된 구조이다.

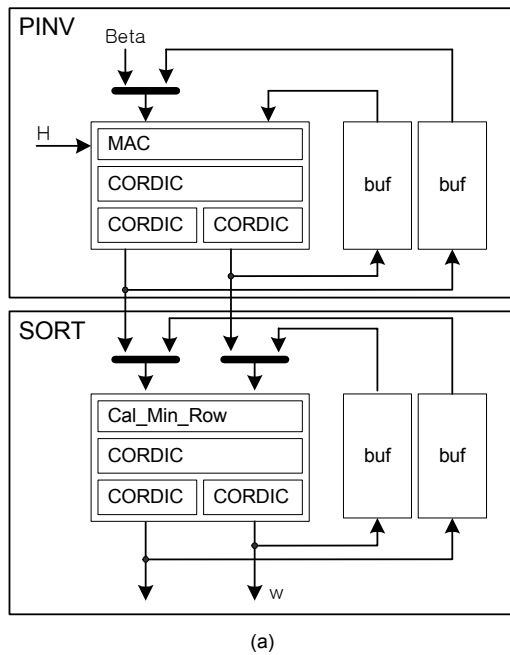


그림 3 (a) 기존 구조

새롭게 제안된 구조로 하드웨어를 설계하기 위해서 내부 bit field를 C언어로 작성한 simulator를 사용하여 표 1과 같이 할당 하였다. 이때의 768만 입력 비트에 대한 시뮬레이션 결과는 그림 4와 같다. 이때, SNR이 20dB일때 BER이 약 9.25% 정도 증가하였다.

다음으로 CORDIC[7]의 내부 단수는 역시 C언어로 작성한 simulator를 사용하여 PINV, SORT 각각을 13, 9단으로 결정했다. 본래 회로의 성능을 높이기 위해서 파이프라인을 보다 세밀하게 설정하여 내부 동작 주파수를 높이는 것이 일반적인 추세이다. 하지만, Square Root Algorithm에서는 회로의 내부에 부귀환이 걸리는 형태로 부귀환이 걸리는 부분의 파이프라인 단수가 입력 데이터의 주기의 최소값을 결정하기 때문에 이부분의 단수를 줄이는 것이 속도 향상에 더 유리하게 된다. 따라서 본 논문에서는 기존의 논문들[5]과는 달리 CORDIC 블록의 설계시 실제 수행은 13단을 수행하는 것과 동일하지만, 내부 파이프라인 단수는 입력데이터의 최대 입력 주기인 9로 조정했다. 이 경우 critical

path의 증가로 회로의 동작 주파수가 낮아지게 되지만, 실제 회로에서 critical path를 형성하는 부분은 CORDIC 내부의 adder 부분이 아닌 복소수 곱셈기 부분이기 때문에 파이프 라인의 조정에 따른 동작 주파수의 감소는 무시할 만하다.

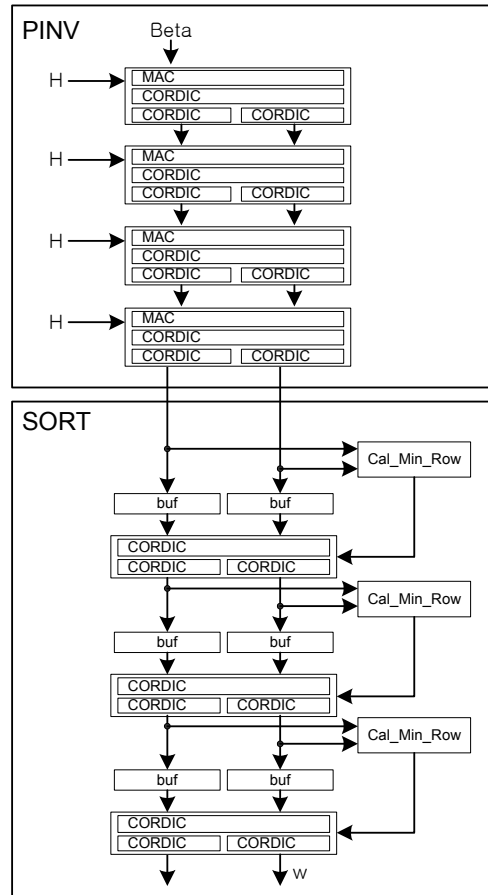


그림 3 (b) 제안된 구조

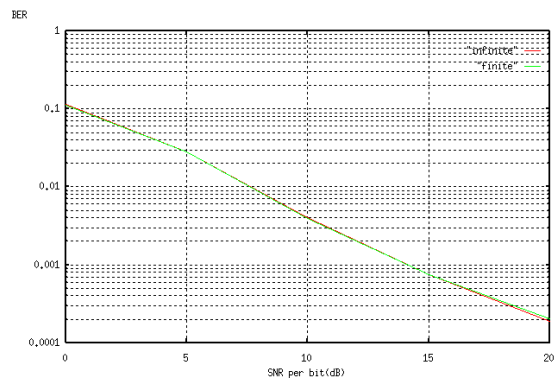


그림 4 부동소수점 vs, 고정소수점 실험결과

이와 같은 조정을 통해서 48개 4x4 채널 행렬의 계산을 위해 필요한 클럭수가 2732clks에서 1982clks로 750clks 감소하게 된다.

변수	정수부	소수부
Beta	8	12
PQ	8	12
H	4	5
s	1	12
r	4	7
w	5	8
y	3	7

표 1 양자화 정리표

#### IV. 합성 및 시뮬레이션

회로를 Verilog HDL을 통해서 RTL로 기술한 후 삼성 0.18 $\mu$ m CMOS 공정을 사용해서 Synopsis의 Design Compiler를 이용하여 합성하였다. 합성한 결과를 살펴보면 Critical path가 3.79ns로 최대 260MHz로 동작하였다. 위의 최대 동작 주파수를 고려할 때 소요시간은 다음과 같다. 첫번째 행렬의 Nulling vector와 detection ordering을 계산하는데 걸리는 시간은 총 290clk이다. 파이프 라인 구조로 제작되었기 때문에 다음 행렬이 얻어지는데 걸리는 시간은 36clk이다. 따라서 총 48 채널행렬을 모두 계산하는데 걸리는 시간은 총 1982clk로 7.5 $\mu$ s가 요구된다. 이 시간은 OFDM 시스템에서 입력 데이터가 4 $\mu$ s마다 한번씩 48개씩 들어오기 때문에 원래 필요한 1kbits정도의 SRAM에 추가로 1kbits를 추가시키는 것을 필요로 한다. 이것은 처음에 제시한 66kbits에 비교할 때 불필요하게 요구되는 SRAM이 거의 사라지게 된 것을 알 수 있다. 합성 결과 총 게이트는 48만개이다. 표 2는 합성 결과를 정리한 내용이다.

사용 공정	0.18 $\mu$ m CMOS 공정
동작 주파수	260MHz
Nulling vectors와 detection ordering 계산시 소요되는 클럭 및 시간	290 clks / 1.099 $\mu$ s
48개의 4x4 행렬의 총 계산시 소요되는 클럭 및 시간	1982 clks / 7.5 $\mu$ s
Equivalent gate의 수	48만 gates

표 2 합성 결과

#### V. 결론

본 논문에서는 MIMO-OFDM을 BLAST계열로 구현하기 위하여 하드웨어 설계에 적합한 SRA를 사용하였다. 그리고 OFDM에서 MIMO를 구현할 경우 생기는

고속 처리의 요구를 만족시키기 위해서 SRA의 파이프 라인 방식 구조를 제안했다. 그리고 보통의 CORDIC 블록과는 다르게 내부에 사용되는 CORDIC block의 단수를 최대 성능을 얻도록 조절하였다. 0.18 $\mu$ m CMOS 공정에서 합성한 결과 회로의 크기는 48만 게이트이고, 7.5 $\mu$ s안에 48개의 4x4 채널행렬의 계산을 할 수 있다.

#### 참고문헌

- [1] Syed Aon Mujtaba, "MIMO Signal Processing - The Next Frontier for Capacity Enhancement", IEEE Custom Integrated Circuits Conference, Sept. 2003.
- [2] Stuber, G.L., "Broadband MIMO-OFDM Wireless Communications", Proc. of the IEEE, Feb. 2004
- [3] IEEE 802.11a standard, ISO/IEC 8802-11:1999/ Amd 1:2000(E)
- [4] P.W. Wolniansky, "V-BLAST : An Architecture for Realizing Very High Data Rates Overs the Rich-Scattering Wireless Channel", ISSSE 98. Sept. 1998.
- [5] Zhan Guo, "A VLSI Implementation of MIMO Detection for Future Wireless Communications", PIMRC 2003, Sept. 2003.
- [6] Hassibi, B., "An Efficient Square-Root Algorithm for BLAST", ICASSP '00., June 2000.
- [7] Milos D. Ercegoavac, "Digital Arithmetic", Morgan Kaufmann Publishers, 2004.