

논문 2010-2-14

IEEE 802.16e Mobile WiMax 시스템을 위한 효율적인 FFT 프로세서 설계

Design of Efficient FFT Processor for IEEE 802.16e Mobile WiMax Systems

박윤옥*, 박종원**

Youn Ok Park, Jong-Won Park

요 약 본 논문에서는 IEEE 802.16e mobile WiMax 시스템을 위한 효율적인 FFT 프로세서 구조를 제안한다. 제안된 scalable FFT/IFFT 프로세서는 128/512/1024/2048-point FFT 연산을 가변적으로 수행할 수 있다. 또한 mixed radix (MR) 기법과 multi-path delay commutator (MDC) 구조를 사용하여 비단순 승산을 줄임으로써 기존의 설계 구조에 비해 시스템 수율 변화 없이 하드웨어 복잡도를 크게 감소시켰다. 제안된 scalable FFT/IFFT 프로세서는 하드웨어 설계 언어 (HDL)를 이용하여 설계 되었고, 0.18um CMOS 스탠다드 셀 라이브러리를 이용하여 논리 합성되었다. 논리 합성 결과 4채널 radix-2 MDC (R2MDC) FFT 프로세서와 비교시 16% 감소된 게이트 수와 27% 감소된 메모리로 구현 가능성이 확인되었다.

Abstract In this paper, an area-efficient FFT processor is proposed for IEEE 802.16e mobile WiMax systems. The proposed scalable FFT processor can support the variable length of 128, 512, 1024 and 2048. By reducing the required number of non-trivial multipliers with mixed-radix (MR) and multi-path delay commutator (MDC) architecture, the complexity of the proposed FFT processor is dramatically decreased without sacrificing system throughput. The proposed FFT processor was designed in hardware description language (HDL) and synthesized to gate-level circuits using 0.18um CMOS standard cell library. With the proposed architecture, the gate count for the processor is 46K and the size of memory is 64Kbits, which are reduced by 16% and 27%, respectively, compared with those of the 4-channel radix-2 MDC (R2MDC) FFT processor.

Key Words : IEEE 802.16e WiMax, MIMO-OFDM, MRMDC, Scalable FFT

I. 서 론

최근 장소에 구애 받지 않고 언제 어디서든지 통신 서비스를 제공받으려는 요구의 증가와 함께 광대역 무선 통신 기술 중 하나로 IEEE 802.16e mobile WiMax 시스템이 광대역 데이터 전송, 고속 이동성 지원, 광역 커버리지, 고용량 제공 등 다양한 특성을 바탕으로 주목 받고

있다[1]. 이러한 IEEE 802.16e mobile WiMax 시스템의 고속 데이터 전송을 위한 핵심 기술로 multiple-input multiple-output orthogonal frequency division multiplexing (MIMO-OFDM) 이 사용된다[2]. MIMO-OFDM은 다중 경로 페이딩 채널 환경에 강하며, 채널 대역폭을 증가시키지 않고 시스템의 용량을 증대시킬 수 있는 장점을 가지고 있어 주파수 자원 제한에 대한 해결책으로 주목 받은 반면, 다수개의 데이터 패스를 가지기 때문에 single-input single-output OFDM (SISO-OFDM) 에 비해 하드웨어 복잡도가 증가하게 되므로, 증

*정회원, 한국전자통신연구원

**정회원, 충남대학교 공과대학 정보통신공학과

접수일자 2010.03.18, 수정일자 2010.4.12

가되는 복잡도 문제를 해결하는 것이 가장 큰 이슈이다. 특히, MIMO-OFDM 시스템에 있어서 fast Fourier transformation (FFT) 프로세서는 가장 큰 복잡도를 갖는 블록 중 하나로 2048/1024/512/128-point를 가변적으로 지원해 줄 수 있는 최적의 FFT/IFFT 프로세서를 설계하는 것이 IEEE 802.16e mobile WiMax 시스템을 위한 핵심 사항이라 할 수 있다[3].

MIMO-OFDM 시스템에서 FFT 프로세서의 가장 보편적인 설계 방법은 데이터 패스 수만큼 FFT 프로세서를 구현하는 것이다. 이 경우 single-path delay feedback (SDF) 파이프라인 FFT 구조가 단일 패스에서 가장 복잡한 비단순 승산을 최소로 하는 특징으로 인해 보편적으로 사용되며[4], 논문 [3]에서는 128-point의 1-4개의 채널을 지원 가능한 FFT/IFFT 프로세서를 제안하였다. 하지만, MIMO-OFDM은 다수개의 데이터 패스를 가지기 때문에 하드웨어 복잡도가 데이터 패스 수에 따라 선형적으로 증가하는 문제가 발생하게 된다. 이를 해결하기 위해 multi-path delay commutator (MDC) 구조를 사용하여 다중 데이터 패스를 하나의 FFT 프로세서로 동시에 처리함으로써 복잡도를 줄이는 방법이 제안되었다[5]. 이 논문에서는 MIMO-OFDM 시스템을 위한 FFT 구조의 하드웨어 복잡도를 분석하였으며, k 개의 입력 데이터 패스를 갖는 MIMO-OFDM 시스템의 경우, k 개의 radix-2 SDF (R2SDF) 구조보다 radix-4 MDC (R4MDC) 구조가 더 면적 효율적임이 제시되었다. 이후, 논문 [6]- [7]을 통해 4개의 송수신 안테나를 갖는 (4x4) MIMO-OFDM 시스템을 위한 radix-4와 radix-2를 혼합 mixed-radix MDC (MRMDC) 구조가 제안되었고, MRMDC 구조가 R4MDC 구조에 비해 비단순 승산의 수를 더 줄일 수 있음이 확인되었다. 그러나 [6]-[7]의 MRMDC 구조는 단순한 radix-4/2 연산의 반복으로 인해 다양한 길이의 FFT를 지원하지 못하는 문제점을 갖는다.

본 논문에서는 IEEE 802.16e mobile WiMax 시스템을 위한 최적 FFT 프로세서의 하드웨어 구조를 제안한다. 제안된 FFT 프로세서는 최대 4 채널의 128/512/1024/2048-point 가변 길이 FFT 연산을 위해 고유의 mixed-radix 알고리즘과 MDC 구조에 기반하여 구현된다. 논문의 구성은 다음과 같다. II장에서 FFT 알고리즘 및 하드웨어 구조를 정리하고, III장에서 제안하는 FFT 프로세서의 하드웨어 구조를 설명한다. 이어서 IV장에서는 설

계한 FFT 프로세서의 설계 및 구현 결과를 제시하고, 마지막으로 V장에서 본 논문의 결론을 맺는다.

II. FFT 알고리즘 및 하드웨어 구조

고속 OFDM 기반 무선 통신 시스템을 위한 FFT 구조는 radix-2, radix-4 알고리즘을 바탕으로 하고 있다. 파이프라인 구조일 경우, radix-4는 radix-2에 비해 수열 면에서 이득을 얻을 수 있지만, 하드웨어 측면에서는 더 복잡한 구조를 가진다. 즉, radix-2 알고리즘이 radix-4 알고리즘에 비해 덜 복잡한 구조를 가짐으로서 면적 면에서의 이득을 얻을 수 있다. 일반적으로 radix가 높아질수록 복소수 승산이 적어지고 하드웨어 구조는 더 복잡해진다. 이에 높은 radix 알고리즘의 복잡도를 줄이기 위해 radix-2², radix-2³이 제안되었다[4]. radix-2² 알고리즘은 radix-2 알고리즘과 같은 버터플라이 구조를 가지면서 복잡도를 낮추었고, radix-4 알고리즘과 같은 복소수 승산을 가진다. 마찬가지로 radix-2³ 알고리즘은 복소수 승산을 단순 승산으로 교체되어 radix-2²에 비해 복소수 승산을 더 줄일 수 있는 특성을 갖는다.

파이프라인 구조는 SDF, MDC, single-path delay commutator (SDC) 으로, 크게 3가지 형태로 구분된다. 이 중 OFDM 방식에서는 입출력 구조가 OFDM 시스템 구현에 적합하다는 점과 적은 하드웨어 자원을 사용하는 특징으로 인해 SDF 방식이 보편적으로 사용되어 왔다[4]. 하지만, MIMO-OFDM 시스템에서는 다수개의 입력 데이터 패스를 가지기 때문에 SDF 방식의 FFT 프로세서를 이용하면 패스 수에 따른 복수의 프로세서를 사용하게 됨으로써 하드웨어 복잡도 증가를 가져온다. 그에 반해 MDC 방식의 FFT 프로세서는 하나의 교환기를 추가하여 하나의 FFT 프로세서로 설계할 수 있어 하드웨어 복잡도를 크게 증가시키지 않고도 다수개의 데이터 패스를 처리 할 수 있다. 따라서, MIMO-OFDM 시스템에서는 MDC 방식이 더 효율적이다[5-7].

이러한 알고리즘과 파이프라인 구조를 이용하여 효율적인 FFT 프로세서를 설계하기 위한 많은 방법들이 연구되어 왔으며, 그 중 비단순 승산을 줄임으로서 하드웨어 복잡도를 줄이는 단일 채널 MRMDC FFT 프로세서[6]와 다중 채널 MRMDC FFT 프로세서가 제안되었다[7]. 이 논문들에서는 MR 알고리즘과 MDC 구조를 적용

하여 기존의 방식에 비해 비단순 승산의 수를 줄일 수 있음이 제시 되었으며, 표 1은 각 구조에 따른 하드웨어 복잡도를 비교하여 보여준다. 상수 T 는 비단순 승산의 수를 의미하며, $[D]$ 는 D 를 넘지 않는 최대 정수를 의미한다. 정리된 결과를 통해 채널수가 4일 경우 MRMDC 방식의 FFT 프로세서가 가장 작은 하드웨어 복잡도를 갖게 됨을 알 수 있다.

표 1. 다채널 N-point FFT 프로세서의 연산량 비교
Table 1. Comparison of the computational complexity for k-MIMO channel FFT processor

	프로세서 수(P)	P당 복소곱셈수	P당 복소덧셈수
R2SDF	k	$\log_2 N - 1$	$2\log_2 N$
R4SDF	k	$\log_4 N - 1$	$8\log_4 N$
R4SDC	k	$\log_4 N - 1$	$3\log_4 N$
R2 ² SDF	k	$\log_2 N - 1$	$4\log_4 N$
R2 ³ SDF	k	$\log_8 N - 1$	$(6+2T)\log_8 N$
R2MDC	$[k/2]$	$\log_2 N - 2$	$2\log_2 N$
R4MDC	$[k/4]$	$3(\log_4 N - 1)$	$8\log_4 N$
MRMDC	$[k/4]$	$4(\log_8 N - 1)$	$(12+3T)\log_8 N$

III. 제안하는 FFT 프로세서의 알고리즘 및 하드웨어 구조

1. 제안하는 FFT 알고리즘

N 개의 이산신호 $x(n)$ 에 대한 N -point discrete Fourier transform (DFT) 의 식을 정의하면 다음과 같다.

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk}, (0 \leq n < N, 0 \leq k < N) \quad (1)$$

여기서 $W_N^{nk} = e^{-j2\pi nk/N}$ 는 twiddle factor이다.

IEEE 802.16e mobile WiMax의 표준 규격[1]을 보면 MIMO-OFDM을 위한 FFT 크기는 128/512/1024/2048-point이다. 따라서 제안하는 IEEE 802.16e mobile WiMax 시스템을 위해서는 하나의 FFT 프로세서로 128/512/1024/ 2048-point를 가변적으로 지원할 수 있어야 하므로 MR-2/2/4/2/4/2/4/2 분해 방법을 제안한다. 이 분해 방법을 이용함으로써 다양한 길이의 FFT 연산을 지원할 수 있을 뿐만 아니라, 발생하는 twiddle factor (TF) 비단순 승산을 최적화시킬 수 있다.

2048-point FFT 식을 표현하기 위해 수식 (1)의 n, k 를 수식 (2)와 같이 정리한다. 수식 (2)을 수식 (1)에 대입하여 정리하면, 2048-point의 FFT 식을 수식 (3)과 같이

$$\left\{ \begin{array}{l} n = \frac{N}{2}n_1 + \frac{N}{4}n_2 + \frac{N}{16}n_3 + \frac{N}{32}n_4 + \frac{N}{128}n_5 + \frac{N}{256}n_6 + n_7 \\ n_1 \in \{0,1\}, n_2 \in \{0,1\}, n_3 \in \{0,1,2,3\}, n_4 \in \{0,1\}, n_5 \in \{0,1,2,3\}, n_6 \in \{0,1\}, n_7 \in \left\{0, \dots, \frac{N}{256} - 1\right\} \\ k = k_1 + 2k_2 + 4k_3 + 16k_4 + 32k_5 + 128k_6 + 256k_7 \\ k_1 \in \{0,1\}, k_2 \in \{0,1\}, k_3 \in \{0,1,2,3\}, k_4 \in \{0,1\}, k_5 \in \{0,1,2,3\}, k_6 \in \{0,1\}, k_7 \in \left\{0, \dots, \frac{N}{256} - 1\right\} \end{array} \right. \quad (2)$$

$$\sum_{n=0}^{2047} x(n) \cdot W_{2048}^{nk} = \sum_{n_7=0}^7 \left(\left(\sum_{n_6=0}^2 \left(\left(\sum_{n_5=0}^3 X_{WiMax} \cdot W_4^{n_5 k_5} \right) W_8^{n_6 k_5} \right) W_2^{n_6 k_6} \right) W_{2048}^{n_7 Q_3} \right) W_8^{n_7 k_7} \quad (3)$$

$$X_{WiMax} = \left(\left(\sum_{n_4=0}^1 \left(\left(\sum_{n_3=0}^3 \left(\left(\sum_{n_2=0}^1 \left(\left(\sum_{n_1=0}^1 x_{2048} \cdot W_2^{n_1 k_1} \right) W_4^{n_2 k_1} \right) W_2^{n_2 k_2} \right) W_{256}^{P_1 Q_1} \right) W_4^{n_3 k_3} \right) W_8^{n_4 k_3} \right) W_2^{n_4 k_4} \right) W_{64}^{P_2 Q_2} \right) \quad (4)$$

$$\left\{ \begin{array}{l} x_{2048} = x [1024n_1 + 512n_2 + 128n_3 + 64n_4 + 16n_5 + 8n_6 + n_7] \\ P_1 = 16n_3 + 8n_4 + 2n_5 + n_6, P_2 = 2n_5 + n_6 \\ Q_1 = k_1 + 2k_2, Q_2 = k_3 + 4k_4, Q_3 = k_1 + 2k_2 + 4k_3 + 16k_4 + 32k_5 + 128k_6 \end{array} \right. \quad (5)$$

표현할 수 있다. 수식 (3)의 변수 X_{WiMax} 는 수식 (4)의 값을 가지며, 수식 (4)의 변수 x_{2048} , P_1 , P_2 , Q_1 , Q_2 는 수식 (5)의 값을 갖는다. 수식 (4)을 통해 2048-point FFT는 2-point FFT와 1024-point FFT로, 1024-point FFT는 2-point FFT와 512-point FFT로, 512-point FFT는 4-point FFT와 128-point FFT로 연산됨을 확인할 수 있다. 즉, 제안된 분해 방법은 처음의 2-point FFT와 복소수 승산 $W_4^{n_2k_1}$ 을 건너뛰면 쉽게 1024-point FFT를 구현할 수 있다. 마찬가지로 두 번째의 2-point FFT와 복소수 승산 $W_{256}^{P_1Q_1}$ 을 건너뛰면 512-point FFT를, 3번째의 4-point FFT와 복소수 승산 $W_8^{n_4k_3}$ 을 건너뛰면 128-point FFT를 쉽게 구현할 수 있다. 이는 하나의 FFT 프로세서를 이용하여 다양한 크기의 FFT 연산을 지원할 수 있음을 의미하며, TF $W_{256}^{P_1Q_1}$, $W_{64}^{P_2Q_2}$ 은 각 FFT 크기에 따라 변수의 값이 달라진다. 2048-point FFT일 때 TF $W_{256}^{P_1Q_1}$ 의 P_1 , Q_1 은 각각 $16n_3 + 8n_{24} + 2n_5 + n_6$, $k_1 + 2k_2$ 이며, 1024-point FFT일 때 k_1 이 0이 되어 Q_1 은 $2k_2$ 이 된다. 마찬가지로 512-point FFT일 때 TF $W_{64}^{P_2Q_2}$ 의 P_2 , Q_2 은 각각 $2n_5 + n_6$, $k_3 + 4k_4$ 이 되며, 128-point FFT일 때 k_3 이 0이 되어 Q_2 은 $4k_4$ 이 된다. 다양한 FFT 연산에 맞는 TF의 인덱스가 결정됨으로써 발생하는 TF 비단순 승산을 최적화시킨다.

$$\left\{ \begin{array}{l} W_{2048}^{n_7Q_3} = W_{2048}^{n_7(k_1 + 2k_2 + 4k_3 + 16k_4 + 32k_5 + 128k_6)} \\ W_{1024}^{n_7Q_3} = W_{1024}^{n_7(k_2 + 2k_3 + 8k_4 + 16k_5 + 64k_6)} \\ \quad = W_{2048}^{2n_7(k_2 + 2k_3 + 8k_4 + 16k_5 + 64k_6)} \\ W_{512}^{n_7Q_3} = W_{512}^{n_7(k_3 + 4k_4 + 8k_5 + 32k_6)} \\ \quad = W_{2048}^{4n_7(k_3 + 4k_4 + 8k_5 + 32k_6)} \\ W_{128}^{n_7Q_3} = W_{128}^{n_7(k_4 + 2k_5 + 8k_6)} = W_{2048}^{16n_7(k_4 + 2k_5 + 8k_6)} \end{array} \right. \quad (6)$$

또한, 수식 (3)에 나와 있는 TF $W_{2048}^{n_7Q_3}$ 은 FFT 연산에 따라 수식 (6)처럼 표현할 있다. 수식 (6)에서 알 수 있듯이 Q_3 은 2048-point FFT일 때 $k_1 + 4k_3 + 2k_2 + 16k_4 + 32k_5 + 128k_6$, 1024-point FFT일 때 $k_2 +$

$2k_3 + 8k_4 + 16k_5 + 64k_6$, 512-point FFT일 때 $k_3 + 4k_4 + 8k_5 + 32k_6$, 128-point FFT일 때 $k_4 + 2k_5 + 8k_6$ 이다. 2048-point FFT일 때 Q_3 의 k_1 값이 0이면 $W_{2048}^{n_7Q_3}$ 이 1024-point FFT의 $W_{1024}^{n_7Q_3}$ 와 같아지며, 이는 1024-point FFT TF 값이 2048-point FFT TF의 처음 1024개와 같다는 의미가 된다. 마찬가지로 $W_{2048}^{n_7Q_3}$ 은 k_1 , k_2 의 값이 0이면 512-point FFT TF $W_{512}^{n_7Q_3}$ 와 같아지고, k_1 , k_2 , k_3 의 값이 0이면 128-point FFT TF $W_{128}^{n_7Q_3}$ 와 같아진다. 즉, 하나의 2048-point FFT twiddle factor ROM (TFR)으로 모든 128/512/1024/2048-point TFR을 구현할 수 있으며, 이로 인해 하드웨어 복잡도를 줄일 수 있다.

제안된 MR-2/2/4/2/4/2/4/2 분해방법은 2개의 radix-2을 사용하여 2048/1024-point FFT를 지원해 주며, 3개의 비단순 승산을 사용함으로써 TF 비단순 승산을 최적화 시켰다. 또한, radix-4와 radix-2를 혼용하여 512/128-point FFT를 지원해 주며, 8개의 비단순 승산을 사용함으로써 TF 비단순 승산을 최적화 시켰다. 즉, 제안된 MR 알고리즘은 12개의 비단순 승산으로 구현 가능하며, 기존의 방법인 R2SDF에 비해 20개, R2MDC에 비해 4개의 비단순 승산수를 감소시킬 수 있다.

2. 제안하는 FFT 알고리즘

그림 1은 IEEE 802.16e mobile WiMax 시스템을 위한 제안된 128/512/1024/2048-point scalable FFT 프로세서의 하드웨어 구조를 도시한다. 제안된 FFT 프로세서 구조는 data mapping unit (DMU), radix-2 unit 1 (R2U1), R2U2, R2U3, R4U1, R4U2, data reordering unit (DRU)으로 구성된다. DMU은 지연소자, MUX, switch으로 구성되며, 다수개의 입력 데이터를 4개의 병렬로 재구성하여 다음 단으로 밀어준다. R2U1은 2개의 radix-2 butterfly (BF), 1개의 비단순 승산기, commutator으로, R2U2은 2개의 radix-2 BF, 4개의 비단순 승산기, TFR, commutator으로, R2U3은 2개의 radix-2 BF, commutator으로, R4U1은 radix-4 BF, 3개의 단순 승산기, commutator으로, R4U2은 radix-4 BF, 3개의 단순 승산기, commutator으로 구현된다. 처음에는 1-4개의 입력 데이터가 DMU을 통해 FFT 크기에 맞는 정확한 길이로 재구성되어 다음 단에 입력된다. 만약 2048-point

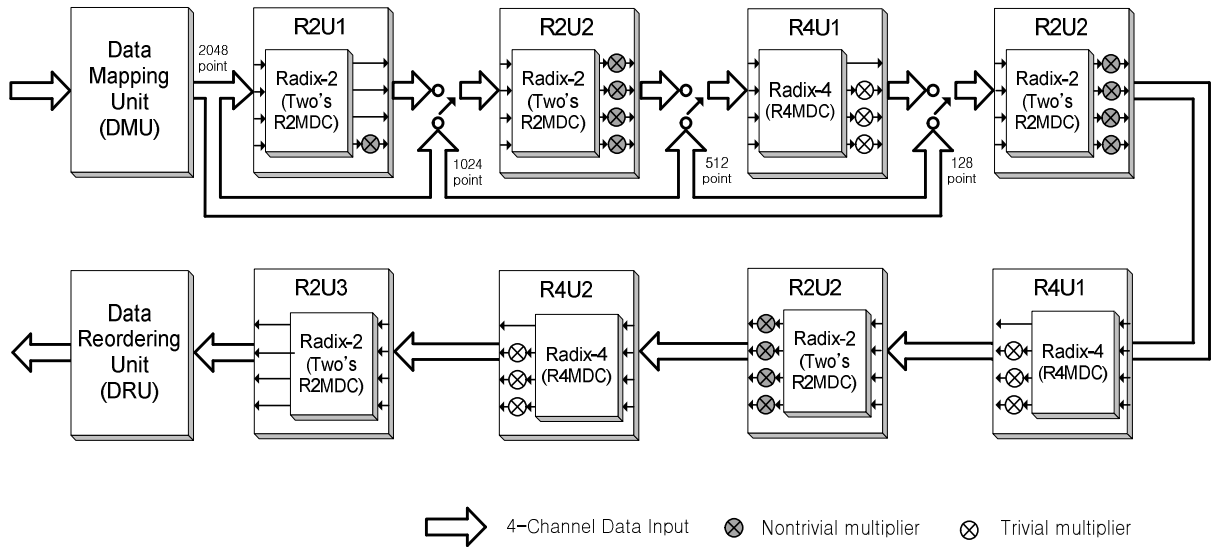


그림 1. IEEE 802.16e mobile WiMax 시스템을 위한 제안된 FFT 프로세서 블록 다이어그램.
Fig. 1. Block diagram of the proposed FFT processor for IEEE 802.16e mobile WiMax Systems.

FFT일 경우, 재구성되어진 1-4개의 데이터 MUX에 의해 R2U1에 입력되며, 2048-point FFT 연산이 이루어진다. 1024-point FFT이면, DMU의 출력 데이터는 R2U1을 지나쳐 첫 번째의 R2U2의 입력으로, 512-point FFT이면 R4U1의 입력으로, 128-point FFT이면 두 번째의 R2U2로 입력된다. DRU는 연산되어진 데이터들을 재정렬 (reordering) 해 준다.

IV. 제안된 FFT 프로세서의 설계 및 구현 결과

제안된 FFT 프로세서는 Verilog HDL을 이용하여 설계되었고, 0.18um의 CMOS 스탠다드 셀 라이브러리를 이용하여 논리합성 되었다. 합성된 결과 제안된 FFT 프로세서는 약 46K 게이트 수와 약 90K 메모리로 구현되는 것을 확인하였다. 제안된 FFT 프로세서가 하드웨어 복잡도 면에서 효율적이라는 것을 검증하기 위해, 2048-point R2MDC 방식의 하드웨어를 설계 및 구현하였고, 표 2는 비교 결과를 정리하여 보여준다. 논리합성 결과, R2MDC 방식의 FFT 프로세서는 약 55K 게이트 수와 약 123K 메모리로 구현되는 것을 확인하였다. 이 결과를 통해 MIMO-OFDM 시스템은 R2MDC 방식보다 제안된 방식이 더 효율적임을 알 수 있다.

V. 결론

표 2. Scalable FFT/IFFT 논리 합성 결과 비교
Table 2. Comparison for the logic synthesis results of the scalable FFT/IFFT processors (2048-point)

	총 게이트 수	메모리(bits)
2*R2MDC	55K (100%)	123K (100%)
Proposed	46K (84%)	90K (73%)

본 논문에서는 IEEE 802.16e mobile WiMax 시스템을 위한 scalable FFT/IFFT 프로세서 구조를 제안하고 1-4 채널 128/512/1024/2048-point FFT 프로세서를 하드웨어로 구현 및 검증하였다. MDC 방식을 이용하여 다수개의 입력 데이터를 하나의 FFT 프로세서로 처리했으며, 제안된 MR 분해 방법을 통해 복잡도 측면에서 가장 큰 비중을 차지하는 비단순 승산의 수를 줄임으로써 복잡도를 크게 낮추었다.

제안된 구조를 갖는 FFT 프로세서의 복잡도 비교를 위해 R2MDC 구조, 제안된 MRMDC 구조의 FFT 프로세서를 구현하였다. 그 결과 제안된 프로세서는 R2MDC FFT 프로세서에 비해 약 16% 감소된 게이트 수와 약 27% 감소된 메모리로 구현 가능함을 확인할 수 있었다. FFT 프로세서는 IEEE 802.16e mobile WiMax 시스템에서 가장 큰 블록 중 하나이므로 제안된 scalable FFT/IFFT 프로세서는 IEEE 802.16e mobile WiMax 시스템

의 하드웨어 복잡도를 줄이는데 큰 공헌을 할 수 있을 것으로 예상된다.

참고문헌

- [1] IEEE Std. 802.16e, "Local and Metropolitan Area Networks-Part 16: Air Interface for Fixed Broad-band Wireless Access System", Oct. 2004.
- [2] G. L. Stuber, J. R. Barry, S. W. McLaughlin, Y. Li, M. A. Ingram, and T. H. Pratt, "Broadband MIMO-OFDM wireless communications," Proc. IEEE, vol. 92, no. 2, pp. 271-297, Feb. 2004.
- [3] Y. W. Lin, and C. Y. Lee, "Design of an FFT/IFFT processor for MIMO OFDM systems," IEEE Trans. on Circuits and Systems, vol. 54, no. 4, pp. 807-815, Apr. 2007.
- [4] S. He and M. Torkelson, "Designing pipeline FFT processor for OFDM (de)modulation," Proc. IEEE URSI Int. Signals, Systems, and Electron., vol. 2, PP. 257-262, Oct. 1998.
- [5] T. Sansaloni, A. Perex-Pascual, V. Torres and J. Valls, "Efficient pipeline FFT processors for WLAN MIMO-OFDM systems," Electronics Letters, vol. 41, no. 19, pp. 1043-1044, Sep. 2005.
- [6] Y. Jung, H. Yoon, and J. Kim, "New efficient FFT algorithm and pipeline implementation results for OFDM/DMT applications," IEEE Trans. Consum., Electron., vol. 49, no. 1, pp. 14-20, Feb. 2003.
- [7] S. Lee, Y. Jung, and J. Kim, "Low complexity pipeline FFT processor for MIMO-OFDM systems," IEICE Electronics Express, vol. 4, no. 23, pp. 750-754. Dec. 2007.

저자 소개

박 윤 옥(정회원)



책임연구원

<주관심분야: 영상처리 기술, 이동통신 시스템, 안테나 및 전파전파, 통방융합기술>

- 학위(석, 박사)
- 경력
 - 1986년 한양대 전자공학과 학사
 - 1997년 충남대 컴퓨터공학과 석사
 - 2001년~현재 충남대 정보통신공학과 박사과정
 - 1987년 ~ 현재 한국전자통신연구원

박 중 원(정회원)



교수

<주관심분야: 컴퓨터구조, 영상처리, 컴퓨터비전>

- 학위(박사)
- 경력
 - 1997년 충남대 전자공학과 학사
 - 1981년 KAIST 전산학과 석사
 - 1991년 KAIST 전산학과 박사
 - 1981년~1993년 충남대 전산학과 부교수
 - 1992년 미국 Univ. of Texas 전기공학과 교환교수
 - 1994년 ~ 현재 충남대학교 공과대학 정보통신공학과 교수