

효율적인 파이프라인 구조와 스케줄링 기법을 적용한 고속 8-병렬 FFT/IFFT 프로세서

준회원 김은지*, 종신회원 선우명훈*

High Speed 8-Parallel Fft/iffT Processor using Efficient Pipeline Architecture and Scheduling Scheme

Eun Ji Kim* Associate Member, Myung Hoon Sunwoo* Lifelong Member

요 약

본 논문에서는 고속 데이터 전송을 위해 OFDM 시스템에 적용 가능한 고속 FFT/IFFT 프로세서를 제안하였다. 제안하는 프로세서는 높은 데이터 처리율을 만족하기 위해서 MDC 구조와 다중 병렬 처리 기법을 채택하였다. 하드웨어 복잡도를 줄이기 위해서 본 논문에서는 연산에 필요한 연산기의 수를 줄이는 구조로 버터플라이 연산기의 수를 줄인 MRMDC 구조와 효율적인 스케줄링 기법을 적용하여 복소 곱셈기의 수를 줄이는 구조를 제안한다. 제안하는 구조를 적용함으로써 연산 사이클을 증가시키지 않고 하드웨어 복잡도를 줄일 수 있다. UWB, WiMAX, O-OFDM과 같은 고속 OFDM 시스템을 위해 제안하는 프로세서는 128-포인트와 256-포인트 두 가지 모드를 지원 가능하다. 제안하는 프로세서는 IBM 90nm 공정으로 합성하여 메모리를 제외한 전체 게이트 수가 760,000개를 보이며, 동작속도는 430MHz를 나타내었다.

Key Words : O-OFDM, UWB, FFT, Mixed-radix, MDC

ABSTRACT

This paper presents a novel eight-parallel 128/256-point mixed-radix multi-path delay commutator (MRMDC) FFT/IFFT processor for orthogonal frequency-division multiplexing (OFDM) systems. The proposed FFT architecture can provide a high throughput rate and low hardware complexity by using an eight-parallel data-path scheme, a modified mixed-radix multi-path delay commutator structure and an efficient scheduling scheme of complex multiplications. The efficient scheduling scheme can reduce the number of complex multipliers at the second stage from 88 to 40. The proposed FFT/IFFT processor has been designed and implemented with the 90nm CMOS technology. The proposed eight-parallel FFT/IFFT processor can provide a throughput rate of up to 27.5Gsample/s at 430MHz.

1. 서 론

OFDM(Orthogonal Frequency Division Multiplexing)

전송 방식은 다중 경로 채널에서의 고속 데이터 전송을 위해 제안되었다. OFDM 번복조는 입력 데이터를 부반송파의 수만큼 직,병렬 변환하여 각각에

※ 본 연구는 지식경제부 및 한국산업기술평가위원회의 산업원천기술개발사업(정보통신)의 일환으로 수행하였음. [KI002145, 차세대 광통신용 디지털 신호처리 기반 초고속 CMOS회로 설계 기술]

* 아주대학교 정보통신대학 전자공학부 SoC 연구실(sunwoo@ajou.ac.kr)

논문번호: KICS2011-01-021, 접수일자: 2011년 1월 17일, 최종논문접수일자: 2011년 2월 17일

대응되는 부반송파로 변조하는 방식이다. OFDM 전송 방식은 고속 데이터 통신을 위한 방식으로 각 광 받고 있으며, IEEE 802.11a/g/n^[1], WiMAX (Worldwide Interoperability For Microwave Access)^[2], UWB (ultra-wideband)^[3], WPAN (wireless personal area network)^[4] 등의 표준에 채택되었다. 뿐만 아니라 차세대 광통신 시스템의 표준으로 채택이 활발히 논의되고 있다^[5,6].

이러한 OFDM 시스템은 DFT (Discrete Fourier Transform)를 이용하여 구현하며, 실제 하드웨어 설계에는 연산량을 줄이기 위해 FFT (Fast Fourier Transform) 알고리즘을 이용한다. FFT 프로세서는 OFDM 시스템에 있어 가장 큰 복잡도를 가지며 고속 연산이 요구되어 구현이 까다로운 부분이다. 이러한 고성능 요구하는 분야를 위해 다양한 프로세서들이 제안되고 있다^[7-15]. 작은 하드웨어 크기를 유지하기 위해서 단일 메모리 구조와 버터플라이 연산부를 사용하는 프로세서들이 제안되었다^[7,8]. 그러나 이 구조들은 많은 연산 사이클이 요구되어 높은 처리 속도를 얻는데 어려움이 있으며 높은 동작 주파수를 요구한다. 고속 동작을 요구하는 분야에서는 이러한 단점을 극복하고 높은 처리 속도를 얻기 위해 파이프라인 구조가 주로 사용된다^[9-13,15]. 파이프라인 구조는 MDC(multi-path delay commutator), SDC (single-path delay commutator), SDF(single-path delay feedback), MDF(multi-path delay feedback) 등으로 분류할 수 있다. 각각의 구조에 따라 전체 구조의 하드웨어 복잡도와 데이터 처리율이 결정된다. 최근에는 파이프라인 구조와 함께 병렬 처리 기법을 적용하여 데이터 처리율을 높이는 구조들이 제안되고 있다^[9-11]. 병렬 경로의 수가 증가할수록 각각의 경로에서의 데이터 샘플링 주파수는 감소하지만 동시에 연산을 수행하는데 필요한 연산기와 메모리가 증가하게 되므로 하드웨어 비용은 현저히 증가한다. 따라서 요구되는 데이터 처리율과 하드웨어 복잡도를 고려하여 적합한 구조와 병렬 구조의 수를 결정하여야 한다.

본 논문에서는 파이프라인 구조와 8-병렬 데이터 경로를 사용함으로써 FFT/IFFT 프로세서의 데이터 처리율을 향상시켰다. 또한 연산 사이클의 증가없이 연산기의 수를 줄일 수 있는 연산기 스케줄링 기법을 제안하였고 이를 통해 제안하는 FFT/IFFT 프로세서의 하드웨어 복잡도를 감소시킬 수 있다. 제안하는 프로세서는 UWB, WiMAX, O-OFDM (Optical OFDM) 시스템을 위해 128, 256-포인트 FFT/IFFT

연산을 지원한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 FFT 구조에 대해 설명하고 3장에서는 제안하는 FFT 구조에 대해 기술한다. 4장에서는 제안하는 FFT 프로세서의 구현 및 성능 평가를 기술하며 마지막으로, 5장에서 결론을 맺는다.

II. 기존의 고속 FFT 프로세서

고속 동작을 위해서 메모리 구조 FFT 프로세서^[7]는 WPAN 시스템을 위해 다중 메모리 뱅크 구조와 여러 개의 버터플라이 연산부들을 병렬로 두어 동시에 연산이 가능한 구조를 제안하였다. 또한 높은 Radix 알고리즘인 Radix-16 알고리즘을 적용하여 연산에 필요한 스테이지 수를 줄였다. 그 결과, 2.5Gsample/s의 데이터 처리율을 가진다. 하지만 WPAN과 같이 512-포인트 FFT 연산을 요구하는 시스템에서만 사용 가능하다. 또한 메모리 구조이기 때문에 하드웨어 복잡도는 낮지만 병렬 가능한 연산기의 수와 메모리 액세스 대역폭의 한계로 인해 그 이상의 데이터 처리율을 필요로 하는 다른 고속 통신 시스템에 적용하기에는 부적합하다.

4-병렬 Radix-24 FFT/IFFT 프로세서^[10]는 UWB 시스템을 위한 구조로 4개의 병렬 경로와 MDF 파이프라인 구조를 적용하여 데이터 처리율을 최대 1.8Gsample/s까지 향상시켰다. 또한, 8-병렬 Radix-2³/2⁴ FFT 프로세서^[11]는 WPAN 시스템을 위한 구조로 8개의 병렬 경로와 MDF 구조를 채택 하였으며 데이터 처리율은 2.4Gsample/s이다. 1Gsample/s 이상의 고속 동작을 위해서, 기존의 구조들^[9-11]은 주로 MDC구조와 SDF구조를 결합한 MDF 구조를 적용하였다. 데이터 처리율은 파이프라인의 구조와 병렬 경로의 수에 따라 결정된다. 따라서 그 이상의 데이터 처리율이 필요한 경우에는 병렬 경로의 수를 증가시키거나 다른 파이프라인 구조를 사용해야 한다. 하지만 병렬 가능한 경로의 수는 FFT 길이에 따라 제한되며, 병렬 경로의 수가 증가하면 그 만큼 하드웨어 복잡도가 증가하게 된다. 따라서 병렬 경로를 증가시켜 얻을 수 있는 데이터 처리율은 제한된다. 또한, 기존의 고속 FFT 프로세서들은 다양한 FFT 연산을 지원하지 않기 때문에 다른 FFT 연산을 요구하는 OFDM 시스템에 그대로 적용할 수 없는 단점이 있다.

III. 제안하는 FFT/IFFT 프로세서

본 절에서는 높은 데이터 처리율과 낮은 하드웨어 복잡도를 가지는 8개의 병렬 경로를 가지는 MRMDC (Mixed-radix Multi-path Delay Com-mutator) FFT/IFFT 프로세서를 제안한다. MDC 구조는 다른 구조들에 비해 하드웨어 복잡도는 크지만 데이터 처리율이 높고 제어가 간단하다. MDC 구조와 함께 8개의 병렬 경로를 사용하여 데이터 처리율을 더욱 향상시킬 수 있다.

제안하는 FFT/IFFT 프로세서의 구조를 그림 1에 나타내었다. 전체 8개의 병렬 경로이며 각각의 경로마다 3개의 스테이지를 가지고 있다. 8개의 입력 데이터가 각각의 서로 다른 경로로 입력되어 총 3번의 스테이지 연산을 통해 얻은 출력은 각각 8개의 병렬 경로에서 8개의 데이터로 총 64개의 데이터가 출력된다. 고속 수행에 적합하도록 높은 Radix 알고리즘인 Radix-8 알고리즘을 사용하였다. UWB, WiMAX, O-OFDM과 같은 고속 OFDM 시스템에서 사용하는 FFT 길이는 주로 128과 256이다^[2,3,5,6]. Radix-8 알고리즘은 128, 256-포인트 FFT와 같이 8°으로 표시할 수 없는 FFT 길이에 적용될 수 없으므로 Radix-2 또는 Radix-4 알고리즘을 함께 사용하는 Mixed-radix 알고리즘을 사용한다. 제안하는 프로세서는 128, 256-포인트 FFT 연산을 지원함으로써 다양한 OFDM 시스템에 적용가능하다.

OFDM 시스템에서 FFT 연산은 수신기에서 요구되며, IFFT 연산은 송신기에서 요구된다. 제안하는 프로세서에서 FFT와 IFFT 연산은 제어 신호

sel_FFT에 의해서 결정된다. IFFT 연산은 그림 1과 같이 FFT 연산의 입출력 데이터에 추가적인 연산을 통하여 IFFT 연산의 결과를 얻을 수 있다.

128-포인트 FFT 연산과 256-포인트 FFT 연산은 첫 번째 스테이지에서 수행되는 버터플라이 연산만 다르고 나머지 스테이지에서의 버터플라이 연산은 Radix-8 알고리즘으로 동일하다. 256-포인트 FFT 연산의 경우에는 Radix-4 버터플라이 연산을, 128-포인트 FFT 연산의 경우에는 Radix-2 버터플라이 연산이 첫 번째 스테이지에서 필요하다. 따라서 첫 번째 스테이지에서 수행될 버터플라이 연산은 FFT 길이에 의해서 결정될 수 있다.

256-포인트 FFT 연산을 수행할 때 기존의 MRMDC 구조와 제안하는 MRMDC 구조를 그림 2과 그림 3에 나타내었다. 기존에 제안된 MRMDC 구조는 모든 스테이지 사이의 병렬 데이터의 수가 같다^[13,15]. 따라서 입력 데이터는 스위치에 의해서 8개의 병렬 데이터 열로 나누고 지연소자를 이용하여 8개의 데이터의 거리를 조정한다. 처음 입력 데이터가 256-포인트 FFT 연산 중 첫 번째 버터플라이 연산을 수행하는 데까지 28 사이클이 필요한 것을 알 수 있다. 첫 번째 스테이지의 출력이 지연소자와 commutator를 거쳐 두 번째 스테이지에 도달하기까지 3 사이클이 필요하다.

그림 3는 256-포인트 FFT 연산을 위해 제안하는 MRMDC 구조를 표현하였다. 제안하는 구조는 기존의 구조와 달리 모든 스테이지에서 병렬 데이터의 수가 같지 않다. 따라서 입력 데이터는 그림 3 처럼 스위치에 의해서 4개의 병렬 데이터 열로 나

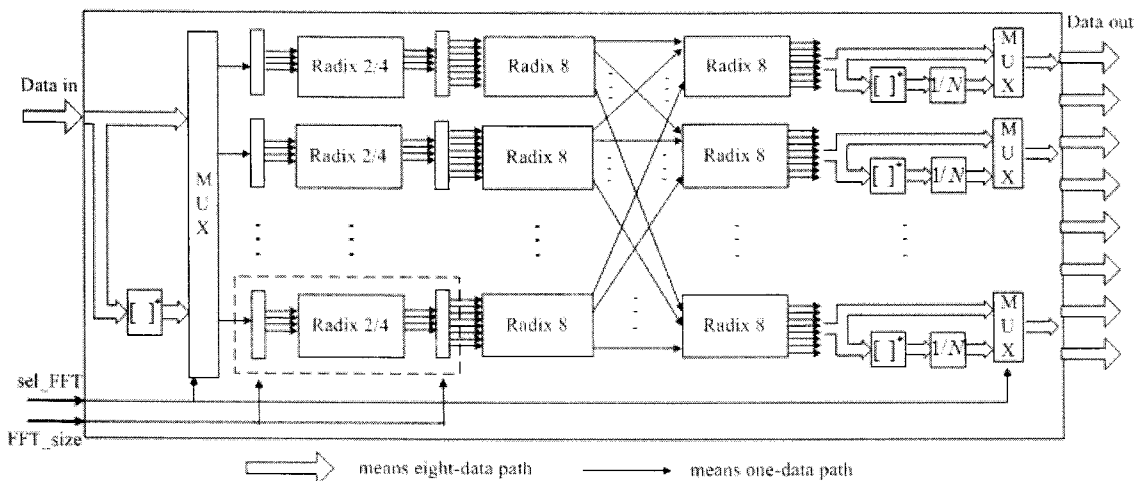


그림 1. 제안하는 FFT/IFFT 프로세서 구조

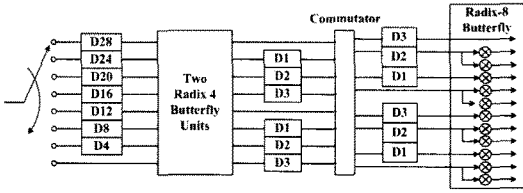
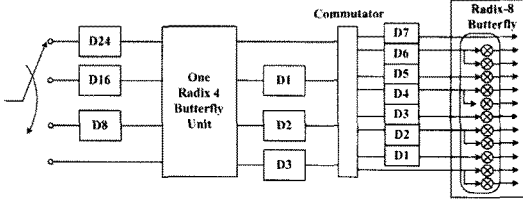
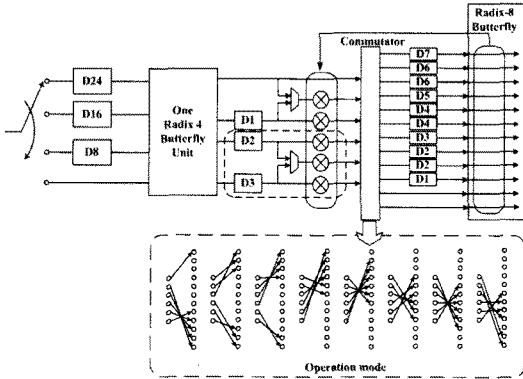


그림 2. 256-포인트 FFT를 위한 기존 MRMDC 구조



(a) 버터플라이의 수를 줄인 MRMDC 구조



(b) 버터플라이와 복소 곱셈기의 수를 줄인 MRMDC 구조

그림 3. 256-포인트 FFT를 위한 제안하는 MRMDC 구조

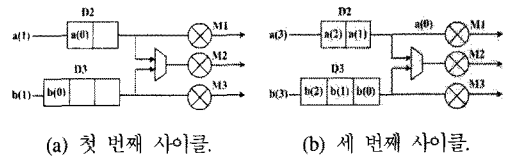
누고 지연소자를 사용하여 데이터들의 거리를 조정한다. 입력 데이터가 들어온 지 24 사이클 후에는 256-포인트 연산 중 첫 번째 버터플라이 연산을 수행할 수 있다. 첫 번째 스테이지 연산을 수행한 후 출력은 다음 스테이지에 입력되기 전에 교환기와 지연소자에 의해 4개의 병렬 데이터 열을 다음 스테이지 연산을 위해 8개의 병렬 데이터 열로 나누어진다. 그런 다음 8개의 데이터는 두 번째 스테이지로 입력된다. 제안하는 구조에서는 첫 번째 스테이지의 출력이 두 번째 스테이지에 도달하는 데 7 사이클이 필요하다. 두 번째 스테이지부터는 기존의 구조와 제안하는 구조는 같은 구조를 가진다. 따라서 제안하는 구조는 기존의 구조와 비교하였을 때 FFT 연산에 필요한 사이클은 증가하지 않고 첫 번째 연산에 필요한 버터플라이 연산부의 수를 줄임으로써 하드웨어 복잡도를 줄일 수 있다.

제안하는 FFT 프로세서는 고속 동작을 위해 파

이프라인구조에 적합한 버터플라이 구조^[14]를 사용한다. 또한 첫 번째 스테이지에서 Radix-2 버터플라이 수행을 위해 Radix-2/4 버터플라이 구조^[8]를 적용하였다. 따라서 제안하는 프로세서는 128, 256-포인트 FFT 연산이 모두 처리 가능하다.

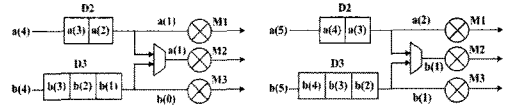
제안하는 프로세서는 버터플라이 연산부 뿐만 아니라 복소 곱셈기의 수도 줄이는 구조를 적용하였다. 그림 3(b)는 곱셈 스케줄링 기법을 적용한 구조를 나타내었다. 고속 Radix-8 버터플라이^[14]는 연산을 수행하기 위해 11개의 복소 곱셈기와 28개의 복소 덧셈기/뺄셈기가 필요하며, 8개의 입력 중에서 4개의 입력은 동시에 서로 다른 twiddle factor와 곱해진다. 복소 곱셈기는 전체 하드웨어 복잡도의 큰 부분을 차지한다. 면적의 효율을 향상시키기 위해 그림 3(a)처럼 연산에 필요한 11개의 복소 곱셈기를 그림 3(b)처럼 5개의 복소 곱셈기로 줄이는 구조를 제안하였다. 제안하는 구조에서는 두 번째 스테이지에서 수행되는 곱셈 연산을 교환기 이전에 미리 연산하고 그 출력은 교환기와 지연소자를 거쳐 두 번째 스테이지로 입력된다.

그림 4는 그림 3(b)에서 각각의 경로에 위치하는 지연소자 D2, D3의 입출력과 곱셈기 M1, M3와 서로 다른 두 개의 경로가 공유하는 곱셈기 M2에 들어오는 입력 데이터 순서를 나타낸다. 입력 데이터가 어떻게 서로 다른 두 개의 twiddle factor와 곱셈연산을 수행하는 지 나타낸다. 그림 4(a)는 첫 번째 데이터가 입력되었을 때 곱셈기에 도달하기 전에 지연소자에 입력된다. 첫 번째 입력이 들어온 지 2 사이클 이후에는, 그림 4(b)과 같이 맨 처음에 들어온 데이터 $a(0)$ 가 지연소자로부터 빠져나와 곱셈기 M1에서 곱셈연산을 수행한다. 3번째 사이클에서는 $a(1)$ 와 $b(0)$ 가 지연소자를 빠져나오고, 그림 4(c)와 같이 $a(1)$ 은 멀티플렉서에 의해 선택되어 곱



(a) 첫 번째 사이클.

(b) 세 번째 사이클.



(c) 네 번째 사이클.

(d) 다섯 번째 사이클.

그림 4. 복소 곱셈 연산의 스케줄링 기법.

셈기 M1, M2에서 연산을 수행한다. 선택되지 않은 $b(0)$ 는 곱셈기 M3에서 twiddle factor와 곱해진다. 그 다음 4번째 사이클에서는 $a(2)$ 와 $b(1)$ 가 지연소자를 거쳐 곱셈기의 입력으로 들어온다. 그림 4(d)에서처럼 $a(2)$ 는 곱셈기 M1에서 twiddle factor와 곱해지고 $b(1)$ 은 멀티플렉서에 의해 선택되어 곱셈기 M2와 M3에서 곱셈 연산을 수행한다.

128-포인트 FFT의 경우, 256-포인트 연산 마찬가지로 버터플라이와 곱셈기의 수를 줄인 구조로 연산을 수행할 수 있다. 기존 MRMDC 구조에서는 첫 번째 스테이지 연산을 수행하기 위해 2개의 버터플라이 연산부가 필요한 반면에 제안한 구조는 하나의 연산부로 첫 번째 스테이지 연산이 가능하다. 제안하는 프로세서는 128-포인트 연산에서도 효율적인 스케줄링 기법을 적용하여 복소 곱셈 연산을 수행한다.

곱셈 연산이 수행한 데이터들은 교환기를 거쳐서 두 번째 스테이지에 입력된다. 교환기는 파이프라인 구조에서 데이터들을 다음 연산에 적합한 순서로 재정렬하는 역할을 한다. 지연 소자는 데이터열 간의 거리를 조정하며, 교환기는 다음 버터플라이 연산부로 데이터들을 전달한다. 256-포인트 FFT 연산을 수행할 때, 교환기는 그림 3(b)와 같이 총 8가지 동작 모드를 가지며, 128-포인트 연산은 4가지 동작 모드를 가진다. 교환기를 거친 데이터들은 다시 지연 소자에서 두 번째 스테이지 연산을 위해 재정렬된다.

Twiddle factor 곱셈 연산을 마친 데이터들은 나머지 Radix-8 버터플라이 연산을 수행하기 위해서 두 번째 스테이지로 입력된다. 기존의 Radix-8 버터플라이 구조^[4]는 8개의 입력을 가진 반면, 제안하는 구조에서는 8개의 입력과 추가로 4개의 입력, 총 12개의 입력을 가진다. 두 번째 스테이지에서는 twiddle factor가 곱해진 12개의 입력을 받아 곱셈 연산을 제외한 나머지 버터플라이 연산을 수행한 뒤, 그 출력들은 다음 스테이지에 입력된다.

그림 1에서 세 번째 스테이지의 구조는 두 번째 스테이지 구조와는 다르다. 두 번째 스테이지에서는 각각의 병렬 경로 내의 데이터들이 입력으로 들어와 버터플라이 연산을 수행하였다. 그러나 세 번째 스테이지는 연산을 수행하기 위해서 그림 1처럼 또 다른 병렬 경로의 데이터들을 입력으로 한다. 또한, 두 번째 스테이지에선 곱셈 연산을 미리 수행한 12개의 입력으로 버터플라이 연산을 수행한 반면, 세 번째 스테이지에선 8개의 데이터를 입력으로 받는다.

동시에 8개의 데이터를 입력으로 받아, 마지막 Radix-8 버터플라이 연산을 수행한다.

IV. 성능 비교

제안한 FFT/IFFT 프로세서는 Verilog HDL 언어를 이용하여 하드웨어로 구현하고 IBM 90nm 공정 라이브러리를 이용하여 논리합성을 수행하였다. 내부 워드 길이는 실수부와 허수부를 각각 10비트로 하였다. 128-포인트 연산의 경우 32dB, 256-포인트의 경우 31dB의 SQNR 성능을 보였다. 논리합성 수행 결과, 메모리를 제외한 전체 FFT/IFFT 프로세서가 약 760,000개의 게이트 수를 보였다.

제안하는 구조들을 단계별로 성능 비교한 결과를 표 1에서 나타내고 있다. 표 1에서 기존 구조는 기존의 MRMDC 구조에 8-병렬 경로, 고속 버터플라이 구조^[4]를 적용한 구조이다. 제안하는 구조 (1)은 기존 구조에 버터플라이를 줄인 구조를 적용한 구조이다. 제안하는 구조 (2)은 제안하는 구조 (1)에 곱셈기 스케줄링 기법을 적용하여 버터플라이와 복소 곱셈기 수를 줄인 제안하는 구조이다. 버터플라이와 복소 곱셈기 수를 줄이는 두 구조를 함께 적용하면 기존의 구조에 비해 20%가량 감소시킬 수 있다. 제안하는 구조 (1)을 적용하면 첫 번째 스테이지의 병렬 데이터열의 수가 4개로 감소하기 때문에 이 구조에 스케줄링을 적용하면 필요한 복소 곱셈기의 수는 5개로 줄일 수 있다. 따라서 제안하는 구조의 gate count는 기존의 구조에 비해 20% 가량, 제안하는 구조(1)와 비교해서는 16% 가량으로 감소시킬 수 있다.

표 2는 제안하는 구조와 기존의 FFT 구조^[7,10,11]와의 성능 비교이다. 기존의 메모리 구조^[7]는 파이

표 1. MRMDC 구조의 FFT 프로세서의 성능 비교

	기존 구조	제안하는 구조(1)	제안하는 구조(2)
Radix-2/4 버터플라이	16	8	8
두 번째 스테이지의 복소곱셈기	11	11	5
복소곱셈기	176	176	128
동작 속도	430MHz	430MHz	430MHz
전체 gate count	945,000 (100%)	907,000 (96%)	760,000 (80.4%)

표 2. 제안하는 FFT 프로세서와 기존 FFT 프로세서의 성능 비교

	제안하는 구조	[7]	[10]	[11]
설계 공정	90nm	90nm	180nm	90nm
구조	파이프라인 구조	메모리 구조	파이프라인 구조	파이프라인 구조
FFT 길이	128, 256	512	128	2048
내부 워드	10 bits	12 bits	10 bits	9 bits
SQNR	32dB	-	33dB	32.8dB
Clock rate	430MHz	324MHz	450MHz	300MHz
Throughput	27.5 GS/s	2.5 GS/s	1.8 GS/s	2.4 GS/s
전체 gate count (메모리 제외)	760,000	-	130,000	-

프라인 구조에 비해 하드웨어 복잡도가 낮지만 데이터 처리율은 한계가 있다. MDF 구조를 적용한 파이프라인 구조^[10,11]은 병렬 경로를 늘리면 데이터 처리율을 향상시킬 수 있다. 하지만 병렬 경로가 증가함에 따라 하드웨어 복잡도가 크게 증가하게 된다. 제안하는 구조는 SDF 구조나 MDF 구조보다 하드웨어 복잡도는 크지만 높은 데이터 처리율을 가지는 MDC 구조를 적용하였다. 8개의 병렬 경로를 통해 데이터 처리율을 더욱 향상시켰으며, 연산에 필요한 연산기의 수를 줄여 하드웨어 복잡도를 감소시켰다. 따라서 표 2와 같이 제안하는 구조는 기존의 구조^[7,10,11]와 비교하였을 때 하드웨어 복잡도는 6배가 크지만 데이터 처리율은 11배에서 최대 15배까지 향상된 것을 알 수 있다.

제안하는 구조는 MDC구조를 적용하여 MDF 구조보다 한 번에 더 많은 데이터를 처리할 수 있다. 제안하는 구조와 기존 구조^[10]의 동작속도는 비슷하므로 제안하는 프로세서는 병렬 경로의 수와 MDC 구조의 적용으로 2x8=16배 향상된 데이터 처리율을 가진다. 또한 기존 구조^[11]와 제안하는 구조는 모두 8개의 병렬 경로의 수로 구성된다. 하지만 제안하는 구조의 동작 속도가 1.43배 높으며 MDC 구조를 적용하여 기존 구조의 MDF 구조보다 8배 높은 데이터 처리율을 가진다. 따라서 제안하는 구조는 기존 구조^[11]보다 1.43x8=11.4배 높은 데이터 처리율을 가진다.

기존의 구조^[7,10,11]는 다양한 길이의 FFT 연산을 수행할 수 없기 때문에 다른 FFT 길이를 사용하는 OFDM 시스템에 적용할 수 없다. 그에 비해 제안하는 구조는 제안하는 Radix-2/4 버터플라이를 사용하기 때문에 128, 256-포인트 FFT 연산을 모두 지원할 수 있다. 뿐만 아니라 최대 27.5 Gsample/s인 높은 데이터 처리율은 UWB, WiMAX, O-OFDM와

같은 고속 OFDM 시스템의 요구를 만족한다. 따라서 제안하는 구조는 128 또는 256-포인트 연산을 사용하는 여러 고속 OFDM 시스템에 적용할 수 있다.

V. 결 론

본 논문에서는 고속 데이터 전송을 위한 OFDM 시스템에 적용 가능한 고속 FFT/IFFT 프로세서를 제안하였다. 본 논문의 구조는 여러 FFT 연산을 위한 FFT의 구현 방안을 제시하였다. 제안하는 구조는 128과 256-포인트 FFT 연산이 모두 가능하며, UWB, WiMAX, O-OFDM과 같은 OFDM 시스템에 적용할 수 있다. 본 논문은 높은 데이터 처리율을 만족하기 위해서 파이프라인 구조와 병렬 처리 기법을 채택하였다. 파이프라인 구조와 병렬 처리 기법으로 인해 증가하는 하드웨어 복잡도를 감소시키기 위해서 버터플라이 연산부의 수를 줄인 MRMDC 구조와 복소 곱셈기의 수를 줄이는 구조를 제안하였다. 제안한 MRMDC 구조와 효율적인 스케줄링 기법을 적용하여 연산기의 수를 줄임으로써, 연산 사이클을 증가시키지 않고 하드웨어 복잡도를 줄일 수 있었다.

설계한 FFT/IFFT 프로세서는 IBM 90nm 공정으로 합성하였으며 메모리를 제외한 전체 게이트 수가 약 760,000개를 보였다. 동작속도는 430MHz로 256-포인트 연산을 85.84ns에 처리 가능한 구조이다. 제안하는 MRMDC 구조는 기존 MRMDC 구조에 비해 하드웨어 복잡도를 약 20%를 줄일 수 있었다. 기존의 FFT 구조들^[7,10,11]과 비교하면 하드웨어 복잡도는 6배 크지만 데이터 처리율은 최대 15배까지 향상되었다. 제안한 고속 FFT 프로세서는 향후 UWB, WiMAX, O-OFDM 등과 같은 OFDM 변복조 방식의 통신 시스템 개발에 활용될 수 있다.

참 고 문 헌

- [1] M. K. A. Aziz, P. N. Fletcher, and A. R. Nix, "Performance analysis of IEEE 802.11n solutions combining MIMO architectures with iterative decoding and sub-optimal ML detection via MMSE and zero forcing GIS solutions," *IEEE Wireless Communications and Networking Conference*, 2004, Vol.3, pp.1451 - 1456, Mar. 2004.
- [2] WiMAX Forum, Mobile WiMAX-Part I: A technical overview and performance evaluations, Feb. 21, 2006.
- [3] T. Domain, "UWB applications, demonstration & regulatory update," Sept 2001 Workshop, March 2001.
- [4] IEEE 802.15.3c-2009: Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPANs).
- [5] W. Shieh, Q. Yang, and Y. Ma, "107 Gb/s coherent optical OFDM transmission over 1000-km SSMF fiber using orthogonal band multiplexing," *Opt. Express*, Vol.16, No.9, pp.6378-6386, Apr. 2008.
- [6] S. L. Jansen, I. Morita, T. C. W. Schenk, N. Takeda, and H. Tanaka, "Coherent optical 25.8-Gb/s OFDM transmission over 4160-km SSMF," *IEEE J. Lightw. Technol.*, Vol.26, No.1, pp.6-15, Jan. 2008.
- [7] S. Huang and S. Chen, "A green FFT processor with 2.5-GS/s for IEEE 802.15.3c (WPANs)," in *Proc. Int. Conf. Green Circuits and Systems*, Jun. 2010, pp.9-13.
- [8] B. G. Jo and M. H. Sunwoo, "New continuous-flow mixed radix (CFMR) FFT using novel in-place strategy," *IEEE Trans. Circuits Syst.*, Vol.52, pp.911-919, May. 2005.
- [9] Y. W. Lin, H. Y. Liu and C. Y. Lee, "A 1 GS/s FFT/IFFT processor for UWB applications," *IEEE J. Solid-State Circuits*, Vol.40, pp.1726-2005.
- [10] M. Shin, H. Lee, "A high-speed four-parallel radix-2⁴ FFT/IFFT processor for UWB applications," in *Proc. IEEE Int. Symp. Circuits and Systems*, May 2008, pp.960-963.
- [11] Song-Nien Tang, Jui-Wei Tsai, and Tsin-Yuan Chang, "A 2.4GS/s FFT Processor for OFDM-Based WPAN Applications," *IEEE Trans. Circuits Syst.*, Vol.57, No.6, pp.451-455, Jun. 2010.
- [12] He Shousheng and M. Torkelson, "Designing pipeline FFT processor for OFDM (de)modulation," in *Proc. Int. Symp. Signals, Systems, and Electronics*, 29 Sep.-2 Oct. 1998, pp.257-262.
- [13] 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, 2003.
- [14] M. Jaber and D. Massicotte, "A New FFT Concept for Efficient VLSI Implementation: Part I - Butterfly Processing Element," in *Proc. IEEE Int. Conf. Digital Signal Processing*, Jul. 2009, pp.1-6.
- [15] Chen-Ming Chung, Shuenn-Shyang Wang and Chien-Sung Li, "Area-efficient architectural design of Radix-4 Pipeline Fast Fourier Transform processor," in *Proc. Workshop on Consumer Electronics and Signal Processing*, Nov. 2004, pp.1-4.

김 은 지 (Eun Ji Kim)

준회원



2009년 2월 아주대학교 정보통신대학 전자공학부 졸업
2009년 3월~현재 아주대학교 정보통신대학 전자공학과 석사

<관심분야> 통신용 SoC 설계

선우명훈 (Myung Hoon Sunwoo)

중신회원



1980년 2월 서강대학교 전자공학과 졸업

1982년 2월 한국과학기술원 전자공학과 석사

1982년 3월~1985년 8월 한국 전자통신연구소 (ETRI)

1985년 9월~1990년 8월 Univ. of Texas at Austin 전자공학과 박사

1992년 8월~1996년 10월 아주대학교 전기전자공학부 조교수

1996년 10월~2001년 9월 아주대학교 전자공학부 부교수

2001년 10월~현재 아주대학교 전자공학부 교수
<관심분야> VLSI 및 Parallel Architecture, 통신 멀티미디어용 DSP 칩 및 SoC 설계