

OFDM 통신시스템을 위한 radix-2² MDF IFFT의 메모리 감소 기법

조경주*

Memory Reduction Method of Radix-2² MDF IFFT for OFDM Communication Systems

Kyung-Ju Cho*

요약 OFDM 기반 초고속 통신시스템을 위한 IFFT/FFT 프로세서는 저면적 저전력이면서 데이터 처리량이 높고 프로세싱 지연이 적어야 한다. 따라서, 파이프라인과 병렬처리를 적용한 radix-2k 알고리즘 기반 MDF(multipath delay feedback) 구조가 적합하다. 기존의 MDF 구조에서 입력신호의 워드길이에 비례하여 커지는 피드백 메모리는 면적과 전력소모가 크다. 본 논문에서는 OFDM 응용을 위한 radix-22 MDF IFFT 프로세서의 피드백 메모리 크기 감소 방법을 제안한다. MDF 구조에서 첫 두 스테이지의 피드백 메모리의 크기는 전체 피드백 메모리의 75%를 차지하므로 첫 두 스테이지의 피드백 메모리 크기 감소에 초점을 맞춘다. OFDM 전송에서 IFFT 입력신호는 변조데이터와 파일럿과 널 신호로 구성된다는 특징을 이용하여 변조데이터와 파일럿/널 신호를 각각 부호있는 정수로 매핑하여 입력신호의 워드길이를 감소시키는 방법을 제안한다. 시뮬레이션을 통해 제안한 방법이 기존 방법보다 피드백 메모리의 크기를 약 39%까지 감소시킬 수 있음을 보인다.

Abstract In OFDM-based very high-speed communication systems, FFT/IFFT processor should have several properties of low-area and low-power consumption as well as high throughput and low processing latency. Thus, radix-2k MDF (multipath delay feedback) architectures by adopting pipeline and parallel processing are suitable. In MDF architecture, the feedback memory which increases in proportion to the input signal word-length has a large area and power consumption. This paper presents a feedback memory size reduction method of radix-22 MDF IFFT processor for OFDM applications. The proposed method focuses on reducing the feedback memory size in the first two stages of MDF architectures since the first two stages occupy about 75% of the total feedback memory. In OFDM transmissions, IFFT input signals are composed of modulated data and pilot, null signals. In order to reduce the IFFT input word-length, the integer mapping which generates mapped data composed of two signed integer corresponding to modulated data and pilot/null signals is proposed. By simulation, it is shown that the proposed method has achieved a feedback memory reduction up to 39% compared to conventional approach.

Key Words : Feedback memory size reduction, IFFT, Multipath delay feedback (MDF), Parallel, Pipeline

1. 서론

최근 초고속 인터넷 등과 같은 멀티미디어 통신을 지원하는 휴대용 단말기의 수요가 증가함에

따라 무선통신 기술이 빠르게 발전하고 있다. OFDM 전송방식은 직렬로 입력되는 데이터 열을 서로 직교인 부반송파에 병렬로 전송하므로 다중 경로 페이딩에 강해 5G를 포함한 IEEE 802.11

This Paper was supported by Wonkwang University in 2019.

*Department of Electronic Engineering Wonkwang University

Received January 16, 2020

Revised February 14, 2020

Accepted February 17, 2020

WiFi, IEEE 802.15.3 WPAN, IEEE 802.16 WMAN 등과 같은 많은 무선통신의 기술표준으로 채택되었다[1]-[3].

OFDM을 기반으로 하는 시스템의 물리계층에서 IFFT와 FFT 연산은 복잡도가 큰 블록 중 하나이다. 따라서, 최적의 IFFT와 FFT 프로세서 설계는 초고속 통신을 구현하기 위한 핵심사항 중 하나이다. IFFT/FFT의 설계에는 radix-2 기반의 radix-2^k(k=2~5)과 radix-4, mixed radix 알고리즘 등이 사용된다[4]-[7].

IFFT/FFT의 구현 시 파이프라인 구조가 선호되며, 데이터 처리량이 높고 프로세싱 지연이 적은 응용에는 MDF(multipath delay feedback) 파이프라인 구조가 적합하다. 피드백 구조는 버터플라이 연산을 위해 입력 데이터를 지연시키기 위한 피드백 메모리가 필요하다[8]-[10]. 전체 피드백 메모리의 크기는 포인트 수(N)와 입력신호의 워드길이에 비례하며, 첫 두 스테이지의 메모리 크기는 전체 메모리의 75%를 차지한다. IFFT 입력신호의 워드길이는 시스템에서 요구하는 SQNR을 만족시키도록 정해야 하므로 쉽게 줄일 수 없어 피드백 메모리 크기를 줄이기 어렵다[8].

통상 OFDM 시스템에서 IFFT 입력은 변조데이터, 파일럿과 널(null) 신호로 구성되고, 이 중 변조데이터만 정규화 인수가 곱해져 양자화 되어 IFFT에 입력된다. 본 논문에서는 변조데이터와 파일럿/널을 구분하여 매핑한 후 IFFT에 입력시키고, 정규화 인수의 곱셈을 두 번째 스테이지 이후로 옮겨 입력신호의 워드길이를 줄여 피드백 메모리 크기를 감소시키는 방법을 제안한다.

2. 배경 지식

2.1 FFT와 IFFT의 상관관계

복소신호 $x(n)$ 에 대한 N-포인트 DFT(discrete Fourier transform)는 다음과 같다.

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

여기서, $W_N^{kn} = e^{-j2\pi nk/N}$ 은 회전인자이다.

DFT의 역변환인 IDFT는 다음과 같다.

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn}, \quad (0 \leq n \leq N-1) \quad (2)$$

위 식은 다음과 같이 표현할 수 있다.

$$x(n) = \frac{1}{N} \left(\sum_{k=0}^{N-1} X^*(k) W_N^{kn} \right)^* \quad (3)$$

여기서, *는 켤레(conjugate)를 의미한다. 입력신호 $X(k)$ 를 켤레 시키고, 출력신호를 켤레 시키면 $1/N$ 을 제외하고 DFT 식과 동일한 식이 된다. 따라서, IFFT는 FFT와 동일한 구조와 회전인자를 사용하여 구현할 수 있다[6]. 일반적으로 포인트 수(N)는 2ⁱ이므로 $1/N$ 은 오른쪽으로 i비트 쉬프트 시켜 간단히 구현할 수 있다.

Radix-2 기반 알고리즘 중 radix-2² 알고리즘이 저전력 FFT의 VLSI 구현에 적합하여 많이 사용된다[7]. 그림 1은 64-포인트 radix-2² FFT의 SDF(single-path delay feedback) 구조를 나타낸다. 박스 안에 표시된 수는 버터플라이 연산을 위해 입력신호의 지연을 위한 메모리의 크기를 나타낸다. 버터플라이 출력 중 덧셈결과는 다음 스테이지로 넘어가고, 뺄셈결과는 피드백 메모리에 저장된다. 첫 두 스테이지의 피드백 메모리는 전체 피드백 메모리의 약 75%를 차지하며, 프로세싱 지연은 (N-1)이다[1][8]. 버터플라이 BF2는

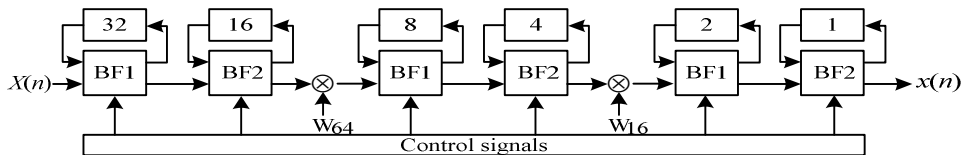


그림 1. 64-포인트 radix-2² SDF FFT 구조.

Fig. 1. 64-point radix-2² SDF FFT architecture.

BF1과 달리 $-j$ 의 곱셈을 포함한다.

Radix-2² 알고리즘에 의하면 스테이지 2의 출력은 다음과 같이 표현할 수 있다.

$$\begin{aligned} a_{o_2}(n) &= A(k_1 = 0) + B(k_1 = 0) \\ s_{o_2}(n) &= A(k_1 = 0) - B(k_1 = 0) \\ a_{o_2}(n + N/2) &= A(k_1 = 1) - jB(k_1 = 1) \\ s_{o_2}(n + N/2) &= A(k_1 = 1) + jB(k_1 = 1) \end{aligned} \quad (4)$$

여기서 a_{o_2} 와 s_{o_2} 는 버터플라이의 덧셈과 뺄셈 출력을 의미하며, A와 B는 스테이지 1의 버터플라이 출력을 의미하며, 다음과 같다.

$$\begin{aligned} A &= x(n) + (-1)^{k_1} x(n + N/2) \\ B &= x(n + N/4) + (-1)^{k_1} x(n + 3N/4) \end{aligned}$$

2.2 IFFT의 입력신호 생성

OFDM 통신에서는 사용환경과 전송속도에 따라 BPSK, QPSK, 16/64-QAM 중 적합한 변조를 선택 데이터를 변조시켜 부반송파에 싣는다. 그림 2는 OFDM 시스템의 송신과정에서 IFFT의 입력신호 생성까지의 절차를 나타낸다. 이진 직렬 데이터는 신호매핑에 의해 변조방식에 따라 BPSK는 1비트, QPSK는 1비트, 16-QAM은 2비트, 64-QAM은 3비트로 매핑된 후 inphase와 quadrature에 할당되어 BPSK, QPSK, 16/64-QAM의 성상도를 표현하는 복소수 ($I+jQ$)로 변환된다. 모든 매핑이 동일한 평균전력을 가지도록 변환된 값에 정규화 인수(K_{mod})를 곱해 최종 변조데이터 d 를 만든다.

$$d = (I + jQ) \times K_{mod} \quad (5)$$

여기서 K_{mod} 는 BPSK, QPSK, 16/64-QAM에 대해 각각 1, $1/\sqrt{2}$, $1/\sqrt{10}$, $1/\sqrt{42}$ 이다.

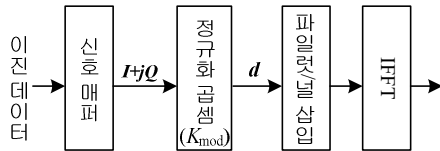


그림 2. IFFT 입력 생성 절차.
Fig. 2. Procedure of IFFT input generation.

OFDM 전송에서는 심볼 동기과 주파수 오프셋 검출을 위해 파일럿(± 1)과 널(0) 신호를 추가한다. 이 신호들은 변조데이터와 달리 정규화 인수

가 곱해지지 않는다. 따라서 변조데이터와 파일럿, 널 신호가 IFFT의 입력신호가 된다. 이 신호들은 IFFT로 입력되기 전에 시스템에서 요구하는 SQNR을 만족하도록 10비트, 12비트, 16비트 등의 워드길이를 양자화 된다.

3. 제안한 방법

이 장에서는 제안하는 방법을 적용할 MDF IFFT의 구조를 살펴보고, IFFT 입력신호의 워드 길이를 줄일 수 있는 입력매핑 기법을 제안한다.

3.1 병렬 IFFT의 구조 (MDF IFFT)

식 (2)의 N-포인트 IFFT를 경로의 수가 L인 $M(=2^k)$ -포인트 IDFT로 계산하기 위해 n 과 k 를 다음과 같이 분해한다.

$$\begin{cases} n = L \cdot m + l, m = 0, 1, \dots, M-1; l = 0, 1, \dots, L-1 \\ k = M \cdot j + i, i = 0, 1, \dots, M-1; j = 0, 1, \dots, L-1 \end{cases} \quad (6)$$

식 (6)의 변수를 적용하면 N-포인트 IFFT는 다음과 같이 2차원 형태로 표현할 수 있다.

$$x(M \cdot j + i) = \sum_{i=0}^{L-1} [x_i(i) W_N^{-il}] W_L^{-jl} \quad (7)$$

여기서 W_N^{-il} 은 회전인자이며, $x_i(i)$ 는 M-포인트 IDFT로 다음과 같다.

$$x_i(i) = \sum_{m=0}^{M-1} X(L \cdot m + l) W_M^{-im} \quad (8)$$

따라서, $N(L \times M)$ -포인트 IFFT의 계산은 세 가지 스텝으로 구분할 수 있다. 첫째 M 포인트 IFFT $x_i(i)$ 을 계산하고, 둘째 회전인자 W_N^{-il} 을 생성하며, 마지막으로 L-포인트 IFFT를 계산한다.

그림 3은 경로가 4인 256-포인트 MDF IFFT 구조이다. IFFT 입력 $X(n)$ 은 4개의 그룹 $X(4m)$, $X(4m+1)$, $X(4m+2)$, $X(4m+3)$ ($m=0, 1, \dots, 63$)으로 나뉘어져 64 포인트 IFFT에 입력되고, 각 IFFT 출력에 offset factor를 곱해 radix-4 버터플라이 연산으로 256-포인트 IFFT를 계산한다. 2.1절 FFT와 IFFT의 관계를 이용하면 64-포인트 IFFT는 그림 1의 FFT를 사용할 수 있다.

경로가 L인 MDF 구조는 $L(M-1)$ 개의 피드백

메모리가 필요하며, SDF 구조보다 처리량과 프로세싱 지연이 L배 우수하다.

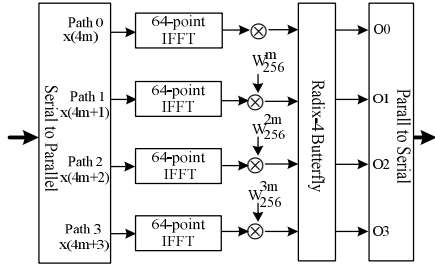


그림 3. 경로가 4인 256-포인트 MDF IFFT 구조.
Fig. 3. 256-point MDF IFFT architecture (L=4).

3.2 IFFT 입력매핑 기법

그림 1에서 알 수 있듯이, 스테이지 1과 2의 피드백 메모리는 전체 피드백 메모리의 약 75%를 차지하며, radix-2² 알고리즘에서는 스테이지 2 이후에 회전인자와의 곱셈이 존재한다. 그림 2의 일반적인 IFFT의 과정과 달리 정규화 인수와의 곱셈을 그림 4와 같이 스테이지 2 이후로 옮길 수 있다면 IFFT의 입력 워드길이를 줄여 스테이지 1과 2의 메모리를 감소시킬 수 있다. 그러나 파일럿에는 정규화 인수가 곱해지지 않으므로 정규화 인수의 곱셈 이동이 쉽지 않다[8].

본 논문에서는 IFFT 입력별 정규화 인수와의 곱셈의 여부를 구분하기 위해 입력에 대한 매핑 기법을 제안한다. 변조데이터와 파일럿/널 신호를 구분하여 다음과 같이 표현한다.

$$I + jQ = (m_{re}, pn_{re}) + j(m_{im}, pn_{im}) \quad (9)$$

여기서 m과 pn은 각각 2의 보수로 표현된 변조데이터와 파일럿/널 신호를 나타낸다. 또한, 밑첨자 re와 im은 real과 imaginary를 나타낸다.

변조데이터의 워드길이를 W_m , 파일럿/널의 워드길이를 W_{pn} 이라하면 제안한 매핑에 의한 입력 워드길이(W_p)는 다음과 같이 표현할 수 있다.

$$W_p = W_m + W_{pn} \quad (10)$$

매핑된 신호는 버터플라이 연산 시 변조데이터와 파일럿/널 신호를 구분하여 덧셈과 뺄셈을 수행한다. 그림 5는 16-QAM에 대한 제안한 매핑기법의 연산

예시 이다. 변조데이터 간에 버터플라이 연산을 할 때 오버플로우가 발생할 수 있으므로 스테이지 마다

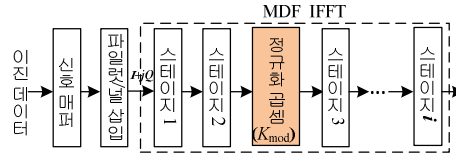


그림 4. 제안한 변조 및 MDF IFFT 과정.
Fig. 4 Proposed modulation and MDF IFFT.

부호확장	변조데이터	파일럿/널
	1 0 1	0 0 (-3s)
+	1 0 1	0 0 (-3s)
	1 0 1 0	0 0 (-6s)
		(a)
	0 1 1	0 0 (3s)
+	0 0 0	1 1 (-p)
	0 0 1 1	1 1 (3s-p)
		(b)

그림 5. 스테이지 1에서 연산 예.
Fig. 5. Example of addition in stage 1.

부호확장을 해야 한다. 그러나, 파일럿 신호는 정해진 부반송파에 일정한 거리로 배치되므로 단일경로와 달리 다중경로인 경우 파일럿 신호끼리는 스테이지 1과 2의 버터플라이에서 만나지 않으므로 부호확장이 필요 없다. 스테이지 2에서도 스테이지 1과 동일하게 버터플라이 연산을 하면 된다.

변조기법에 따라 가능한 IFFT의 입력값이 정해져 있고, 식 (4)처럼 스테이지 2까지 버터플라이 연산만 수행되므로 스테이지 2의 출력은 한정된 값을 가진다. 입력매핑 적용 시 스테이지 2의 버터플라이 결과는 $(stg2_out_{re}, pn_{re}) + j(stg2_out_{im}, pn_{im})$ 으로 표현할 수 있다. 이 값을 사용하여 실제 스테이지 2의 버터플라이 출력은 다음과 같이 계산할 수 있다.

$$stg2_out = (stg2_out_{re} \cdot K_{mod} + pn_{re}) + j(stg2_out_{im} \cdot K_{mod} + pn_{im}) \quad (11)$$

위 식에서 곱셈의 결과는 시스템에서 요구하는 SQNR을 만족시킬 수 있는 워드길이를 양자화 된다. 스테이지 2의 결과 값이 제한되어 있으므로 식 (11)에서 곱셈은 룩업테이블(look-up table)을 사용하여 구현할 수 있다. 64-QAM인 경우, 스테이지 2의 가능한 출력 값은 48개이지만, 룩업테이블 크기 감소기

법[8]을 사용하면 9개로 줄 일 수 있다. 예를 들면, K_{mod} 값이 s 라면, 스테이지 2의 출력 값($\pm s, \pm 2s, \pm 4s, \pm 16s$)은 s 를 왼쪽으로 쉬프트시켜 구할 수 있다. 이때 음수인 경우는 보수를 취하면 된다.

4. 시뮬레이션 및 성능평가

제안한 방법의 메모리 감소정도를 평가하기 위해 기존 방법과 제안한 방법(MDF 구조, 경로 2와 4)에 대한 메모리 크기를 비교하였다. 기존 방법인 SDF 구조는 입력매핑 없이 IFFT의 입력 신호 워드길이(W)를 각 스테이지에서 동일하게 유지하는 것으로 가정하였다. [8]의 SDF 구조와 제안한 MDF 구조는 가장 큰 워드길이를 요구하는 64-QAM을 가정하였다.

[8]의 SDF 구조의 경우, 스테이지 1의 입력은 6비트($W_m: 4, W_{pn}: 2$)이며, 스테이지 1의 버터플라이 출력 중 뺄셈결과가 피드백 메모리에 저장되어야하므로 부호확장을 고려하여 8비트($W_m: 5, W_{pn}: 3$), 스테이지 2의 버터플라이 출력은 9비트($W_m: 6, W_{pn}: 3$)를 가진다. 파일럿 신호는 배치 특성상 연속해서 두 스테이지 모두에서 만나지 않으므로 부호확장은 1번만 필요하다. 스테이지 3 이후부터는 기존 IFFT 방법과 동일하다.

입력경로가 2개 이상인 제안한 MDF 구조의 경우, 파일럿 신호는 배치 특성상 스테이지 1과 2에서 만나지 않으므로 [8]의 SDF 구조와 달리 부호확장이 필요 없다. 따라서 스테이지 1과 2에서 피드백 메모리의 워드길이는 각각 7비트($W_m: 5, W_{pn}: 2$), 8비트($W_m: 6, W_{pn}: 2$)이다. 입력경로가 4인 경우도, 스테이지 1과 2의 피드백 메모리의 워드길이는 경로가 2인 경우와 같으나, 실제 스테이지 2의 출력을 계산하기 위한 룩업테이블이 경로 수 차이만큼 더 필요하다.

그림 6(a)는 입력 워드길이를 12비트와 14비트로 가정했을 때, IFFT의 포인트 수(N)에 따라 일반 SDF 구조(100%)를 기준으로 [8]과 제안한 방법의 메모리 감소 정도를 비교한 것이다. 정규화 인수의 이동으로 인해 스테이지 2에서 식 (11)처럼 실제 스테이지 2의 결과 값을 계산하기 위한

룩업테이블이 필요하다. 경로 수가 증가하면 경로 수만큼 룩업테이블의 크기도 증가하므로 $N=64$ 일 때는 감소 정도가 반감되어 [8]의 방법과 큰 차이가 없으나, $N=128$ 이후부터 [8]의 방법보다 감소율이 증가한다. 제안한 방법($L=2, W=14$)은 SDF 구조보다 최대 35%까지 메모리를 감소시킬 수 있다. 그림 6(b)는 $N=512$ 일 때 워드길이의 변화에 따른 메모리 감소 정도를 비교한 것이다. 제안한 방법($L=2$)은 워드길이가 커질수록 감소율이 증가하며, 최대 39%까지 메모리를 감소시킬 수 있다. 제안한 방법은 IFFT의 크기와 입력 워드길이가 커질수록 메모리 감소율이 증가함을 알 수 있다.

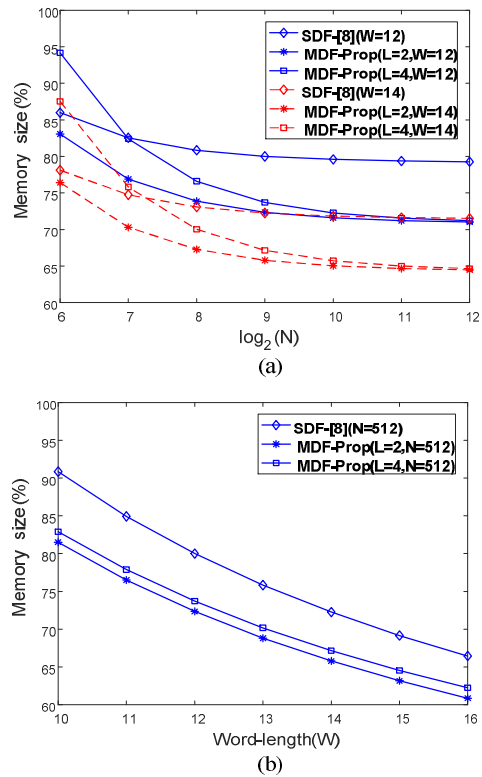


그림 6. (a) IFFT 크기와 (b) 워드길이별 메모리감소 비교
Fig. 6. Comparison of memory reduction for IFFT size and word-length.

5. 결론

OFDM 기반 초고속 통신에는 데이터 처리량이 높은 MDF IFFT가 주로 사용된다. 이 구조에서 피드백 메모리의 크기는 IFFT 크기와 입력신호의 워드길이에 따라 결정되는데, 입력의 워드길이는 시스템에서 요구하는 SQNR을 만족하도록 정해야 하므로 워드길이를 줄이는데 한계가 있다. 일반적으로 OFDM 시스템에서 IFFT 입력은 변조데이터, 파일럿과 널 신호로 구성되며, 이 신호들은 정규화 인수와 곱해져 양자화 되고 IFFT에 입력된다. 본 논문에서는 일반적인 방법과 달리 radix-2² MDF IFFT에서 첫 두 스테이지까지는 버터플라이 연산만 존재하므로 IFFT 입력신호를 변조데이터와 파일럿/널 신호로 구분하여 각각 입력매핑하고, IFFT 내에서 정규화 인수의 곱셈을 처리하여 입력신호의 워드길이를 감소시키는 방법을 제안하였다. 시뮬레이션을 통해 피드백 메모리의 크기를 39%까지 감소시킬 수 있음을 보였다.

제안한 방법을 OFDM 송신기에 적용한다면 IFFT의 메모리 감소를 감소시켜 저면적, 저전력 송신기를 구현할 수 있을 것으로 기대 된다.

REFERENCES

- [1] S. He and M. Torkelson, "Designing pipeline FFT processor for OFDM (de) modulation", *URSI Int Symp Signals Syst Electron.*, pp. 257-262, 1998.
- [2] "Wireless LAN Medium Access Control and Physical Layer Specifications", in Proc. IEEE Advanced Communication Technology, Part 15.3, pp. 804-808, Feb, 2012.
- [3] "Wireless MAC and PHY Specifications for High Rate WPANs", IEEE Std 802.15.3-2003, Local and metropolitan area networks Part 15.3.
- [4] T. S. Cho and H. H. Lee, "A high-speed low-complexity modified radix-2⁵ FFT processor for high rate WPAN applications", *IEEE Trans. VLSI Syst.*, vol. 21, no. 1, pp. 187-191, Feb, 2013.
- [5] S. J. Huang, S. G. Chen, "A high-throughput

- radix-16 FFT processor with parallel and normal input/output ordering for IEEE 802.15. 3c systems", *IEEE Tran. Circuits Sys. I*, vol. 59, pp. 1752-1765, 2012.
- [6] C. Yu, M. H. Yen and P. A. Hsiung, "A low-power 64-point pipeline FFT/IFFT processor for OFDM applications", *IEEE Trans Consum. Electron.* vol. 57, pp. 40-45, 2011.
- [7] G. K. Ganjikunta and S. K. Sahoo, "An area-efficient and low-power 64-point pipeline Fast Fourier Transform for OFDM applications", *Integration, the VLSI Journal*, vol. 57, pp. 125-131, 2017.
- [8] I. G. Jang, J. K. Cho, Y. E. Kim and J. G. Chung, "Memory size reduction technique of SDF IFFT architecture for OFDM-based applications", *IEICE Trans. Commun.*, vol. 95-B, pp. 2059-2064, 2012.
- [9] C. Yang, C. Wei, Y. Xie, H. Chen and C. Ma, "Area-efficient mixed-radix variable-length FFT processor", *IEICE Electron. Expr.*, vol. 14, no. 10, pp. 1-10, 2017.
- [10] J. Yu and K. J. Cho, "An area-efficient 256-point design for WiMAX system", *JKIIECT*, vol. 11. no. 3, pp. 270-276, 2018.
- [11] J. Yu and K. J. Cho, "A low-area and low-power 512-point pipelined FFT design using radix-2⁴-2³ for OFDM applications", *JKIIECT*, vol. 11. no. 5, pp. 475-480, 2018.

저자약력

조 경 주 (Kyung-Ju Cho)

[정회원]



- 2006년 8월: 전북대학교 정보통신공학과 박사
- 2012년 3월 ~ 현재: 원광대학교 전자공학과 교수

<관심분야> VLSI설계, SOC