

논문 2010-47TC-10-5

# CIM(Combined Integer Mapping)을 이용한 OFDM 송신기의 IFFT 메모리 감소

(Memory Reduction of IFFT Using Combined Integer Mapping for  
OFDM Transmitters)

이재경\*, 장인걸\*, 정진균\*\*, 이철동\*\*\*

(Jae-Kyung Lee, In-Gul Jang, Jin-Gyun Chung, and Chul-Dong Lee)

## 요약

FFT(Fast Fourier Transform)는 IEEE 802.22와 같은 여러 무선표준에서 사용되는 OFDM 시스템의 주요 블록 중 하나이다. FFT의 전력소모 감소, 면적감소, 고속동작을 위해 새로운 FFT 아키텍처 개발, twiddle factor 곱셈을 위한 곱셈기의 수나 면적감소, 제어회로의 단순화 등에 초점을 둔 FFT 프로세서의 구현에 관한 연구가 지속적으로 진행되어왔다. FFT의 입력포인트 수  $N$ 이 증가함에 따라  $\log_2 N$  개의 각 FFT 스테이지 구현에 사용되는 시프트레지스터(또는, 메모리)가 차지하는 비중이 전체 FFT회로의 70%이상이며 이러한 메모리들은 FFT의 처음 두 스테이지에 집중되어 두 스테이지의 메모리가 전체 메모리의 75%를 차지한다. 본 논문에서는 OFDM 송신부의 IFFT(Inverse Fast Fourier Transform)에서 요구되는 메모리 사이즈를 감소시키기 위해 입력변조신호, 파일럿(pilot)신호, 널(null) 신호의 mapping을 IFFT와 결합하는 새로운 기법을 제안한다. Cognitive radio 시스템에 적용하기 위한 2048포인트 IFFT를 제안한 방법으로 설계하고 메모리가 차지하는 면적에서 기존의 방법과 비교하여 38.5% 이상의 이득을 가짐을 보인다.

## Abstract

FFT(Fast Fourier Transform) processor is one of the key components in the implementation of OFDM systems for many wireless standards such as IEEE 802.22. To improve the performances of FFT processors, various studies have been carried out to reduce the complexities of multipliers, memory interface, control schemes and so on. While the number of FFT stages increases logarithmically ( $\log_2 N$ ) as the FFT point-size ( $N$ ) increases, the number of required registers (or, memories) increases linearly. In large point-size FFT designs, the registers occupy more than 70% of the chip area. In this paper, to reduce the memory size of IFFT for OFDM transmitters, we propose a new IFFT design method based on a combined mapping of modulated data, pilot and null signals. The proposed method focuses on reducing the sizes of the registers in the first two stages of the IFFT architectures since the first two stages require 75% of the total registers. By simulations of 2048-point IFFT design for cognitive radio systems, it is shown that the proposed IFFT design method achieves more than 38.5% area reduction compared with previous IFFT designs.

**Keywords :** IFFT, Integer mapping, Quantization, Memory reduction, Cognitive radio

\* 학생회원, \*\* 정회원, 전북대학교 전자정보공학부, 정보통신연구소

(Div. of Electronic & Information Engineering, Information & Telecommunications Research Center, Chonbuk National University)

\*\*\* 정회원, 전자부품연구원 (KETI)

※ 본 연구는 지식경제부 및 한국산업기술평가관리원의 IT산업원천기술개발사업[B1120-0901-0002, IT특화연구소설립]과 2단계 BK21 사업 지원에 의하여 수행하였음.

접수일자: 2010년5월26일, 수정완료일: 2010년10월15일

I. 서 론

OFDM 시스템은 많은 통신시스템들에 널리 사용되며 OFDM 시스템 구현의 핵심블록 중 하나가 FFT이다. 변적, 전력, 속도면에서 효율적인 FFT 구현을 위한 많은 연구가 수행되어왔으며 파이프라인 FFT 아키텍처가 실시간 응용시스템의 VLSI 구현에 적합한 것으로 알려져 있다<sup>[1~4]</sup>.

다양한 파이프라인 FFT 구조가 제안되었으며 메모리나 제어회로 측면에서 효율적인 구조 중 하나가 Radix-22 구조이다. Radix-22 FFT 알고리즘은 기존의 Radix-2 알고리즘의 구조를 개량한 것으로 Radix-4 구조의 곱셈량과 Radix-2 구조의 간단한 연산구조를 복합 수용하여 복잡한 곱셈을 효과적으로 줄이고, 제어를 쉽게 만든 구조이다. 이와 관련해서 다양한 Radix-2i 구조와 복합적인 구조가 연구되었다<sup>[4]</sup>.

OFDM 방식에서 광대역과 이동성에 대한 요구가 높아짐에 따라 IEEE 802.22의 2048포인트 FFT와 같이 큰 포인트 사이즈의 FFT 프로세서에 대한 요구가 증가하고 있다<sup>[5]</sup>. FFT의 포인트 수  $N$ 이 증가함에 따라 스테이지 개수는 로그함수  $\log_2 N$ 으로 증가하지만 시프트 레지스터(또는, 메모리)는 선형적으로 증가하게 된다. 따라서, 전체 FFT회로에서 메모리가 차지하는 비중이 약 70% 이상이 되며 이러한 메모리들은 FFT의 처음 두 스테이지에 집중되어 두 스테이지의 메모리가 전체 메모리의 75%를 차지한다.

Radix-2<sup>i</sup> SDF(Single-Path Delay Feedback) IFFT 구조에서 변조된 입력데이터의 mapping을 고려한 IFFT메모리감소방법이 제안되었다<sup>[6]</sup>. 그러나 [6]에서는 입력신호 중 파일럿(Pilot)신호 및 널(Null)신호에 대한 고려가 제외되어있으므로 이에 대해 별도의 mapping 방법의 개발이 필요하다. 본 논문에서는 변조데이터 뿐 아니라 파일럿신호 및 널신호를 고려한 정수매핑방법을 제안하고 이를 이용해 효율적으로 IFFT의 메모리를 감소시킬 수 있음을 보인다.

II장에서는 변조방식에 따른 IFFT내 butterfly의 입·출력에 대해 설명하고, III장에서는 정수매핑을 이용한 IFFT 메모리감소방법을 제안한다. IV장에서는 제안하는 방법으로 Radix-2<sup>4</sup> SDF 구조의 CR 시스템을 위한 2048포인트 IFFT를 설계하여 결과를 비교하고, 끝으로 V장에서 결론을 맺는다.

II. 변조방식과 Butterfly 입·출력

OFDM 시스템에서 기저대역의 데이터들은 BPSK, QPSK, 16-QAM, 64-QAM 등의 다양한 변조방식을 거쳐 변조된다. 그림 1에서 보는 바와 같이 비트 인터리버(bit interleaver)의 출력은 데이터 변조를 위해 constellation mapper로 입력된다. Constellation mapper에서는 각 변조방식의 constellation point에 따라 Gray-coding constellation mapping으로 입력 데이터를 변환한다. 이때 변조되는 데이터들은 평균전력(average power)이 같도록 하기 위해 변조방식에 따라 표 1과 같은 normalization factor( $K_{mod}$ )를 곱하여 정규화 시킨다. 변조방식에 따른 각 constellation point의 실제값을 나타내는 식은 다음과 같다.

$$d = (I + jQ) \times K_{mod} \tag{1}$$

변조된 신호는 복소수로서 실수 및 허수값의 종류가 정해져 있고 서로 등 간격을 이루며 실수와 허수축을 중심으로 대칭이라는 특징이 있다. 변조된 입력과는 별

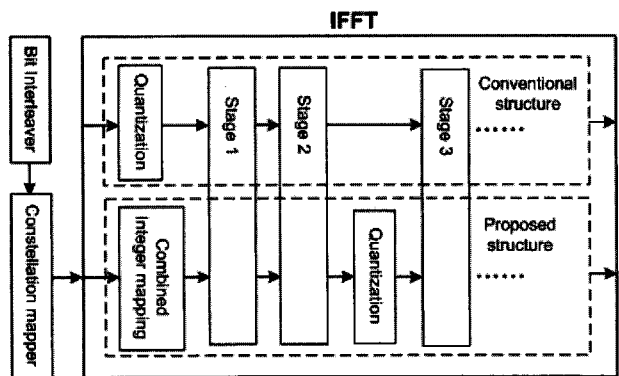


그림 1. OFDM transmitter의 IFFT 입력 및 구조 비교를 위한 블록 다이어그램

Fig. 1. Block diagram of OFDM transmitters for the comparison of IFFT input and structures.

표 1. 변조방식에 따른 normalization factor  $K_{mod}$   
Table 1. Modulation-dependent normalization factor  $K_{mod}$ .

Modulation	$K_{mod}$
BPSK	1
QPSK	$\frac{1}{\sqrt{2}}$
16-QAM	$\frac{1}{\sqrt{10}}$
64-QAM	$\frac{1}{\sqrt{42}}$

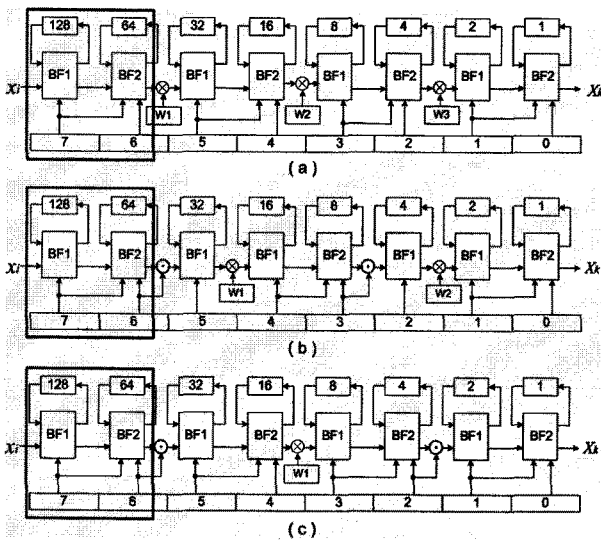


그림 2. Radix-2<sup>2</sup> SDF 256포인트 IFFT 구조:  
 (a) Radix-2<sup>2</sup> 구조, (b) Radix-2<sup>3</sup> 구조,  
 (c) Radix-2<sup>4</sup> 구조

Fig. 2. Radix-2<sup>2</sup> SDF 256 point IFFT structures:  
 (a) Radix-2<sup>2</sup> structure, (b) Radix-2<sup>3</sup> structure,  
 (c) Radix-2<sup>4</sup> structure.

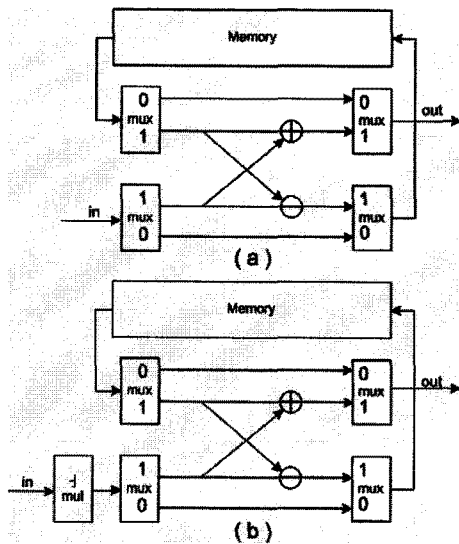


그림 3. BF1 및 BF2의 구조:  
 (a) BF1, (b) BF2

Fig. 3. BF1 and BF2 structures:  
 (a) BF1, (b) BF2.

도로 심볼의 동기를 맞추거나 보호구간 삽입 등을 위해 파일럿(pilot) 신호 및 널(null)신호가 IFFT에 입력된다.

그림 2는 Radix-2<sup>i</sup> SDF 256포인트 IFFT 구조를 보인다. 그림에서 ⊙는 복소 상수 곱셈기이고 ⊗는 프로 그래머를 복소 곱셈기를 나타낸다. 각 그림의 상단에 표시된 숫자들은 메모리의 depth를 의미한다. 그림 3은 butterfly의 구조이다. BF2에서 -j를 곱하는 블록은 실

수와 허수의 위치를 바꾸는 연산으로 실제 곱셈이 이뤄지지 않는다. 그림 2의 각 IFFT 구조는 곱셈기의 개수만 다르다는 것을 알 수 있다. 각 구조에서 박스로 표시된 부분의 메모리(192개)가 전체 IFFT의 메모리(255개)의 75%를 차지하고 있다.

OFDM 시스템에서는 송신 데이터들이 IFFT의 butterfly로 입력되기 전에 시스템에서 요구하는 SQNR에 따라 양자화 되어야한다. 높은 SQNR(Signal-to-Quantization-Noise Ratio)을 요구하는 시스템에서는 양자화로 생기는 에러를 줄이기 위해 양자화 비트 수를 증가시켜야한다.

그림 2의 박스 내 처음 스테이지의 BF1 출력이 13비트 일 경우 13 × 128 사이즈의 메모리가 필요하다. 만일, BF1의 출력을 3비트로 감소시킬 수 있다면 3 × 128 사이즈의 메모리만 사용하면 되므로 10 × 128에 해당하는 메모리 사이즈를 감소시킬 수 있다. 또한 박스로 표시된 부분에서는 회전인자와의 곱셈연산이 이루어지기 전이므로 입력의 덧셈과 뺄셈만 이루어진다. 따라서 IFFT의 변조데이터 입력과 파일럿신호 및 널신호의 종류가 정해져 있으므로 첫 번째 스테이지와 두 번째 스테이지에서 출력되는 값의 종류도 쉽게 알 수 있다.

### III. 메모리 감소를 위한 Combined Integer Mapping

많은 OFDM 시스템에서는 채널의 상황에 따라 입력 데이터의 변조방식을 다르게 선택할 수 있다. 데이터의 변조방식이 달라지면 constellation point의 종류 및 normalization factor( $K_{mod}$ )가 달라진다. IEEE 802.22에서 규정한 Cognitive Wireless RAN(Regional Area Network)에서 pilot subcarrier는 BPSK 변조방식을 따르고 normalization factor( $K_{mod}$ )는 1로 고정되어 pilot 신호는 -1과 1의 2종류를 가진다. 즉, 입력 데이터는 각각의 변조방식에 따라 그 값이 달라지지만 파일럿 신호와 널 신호는  $K_{mod}$  값이 1로 고정되어 있기 때문에 변조방식에 따라 값이 바뀌지 않고 allocation만 바뀌는 특징을 가진다.

식 (1)과 같이  $K_{mod}$ 를 미리 곱한 후 양자화를 하면 요구되는 입력 워드길이가 증가되어야하기 때문에 본 논문에서는 그림 1에서와 같이  $K_{mod}$ 를 곱하고 양자화하는 연산을 두 번째 스테이지 이후에 수행하는 방법을 제안한다. 이와 같이 함으로써 전체 메모리의 75%가

Signed Integer (Modulated Data)	Signed Integer (Pilot/Null Signal)
------------------------------------	---------------------------------------

그림 4. 변조 데이터와 파일럿/널 신호의 combined integer 표현

Fig. 4. Combined integer representation of modulated data and pilot/null signals.

집중되어있는 첫 번째와 두 번째 스테이지에서 메모리의 요구량을 감소시킬 수 있다. 하지만 처음 두 스테이지에서 butterfly를 통과할 때 덧셈과 뺄셈 연산을 수행하게 되고, 두 번째 스테이지 이후에  $K_{mod}$ 를 곱할 때 데이터인지, 파일럿/널 신호인지에 따라  $K_{mod}$  값이 달라져야하는 문제가 발생한다.

이를 해결하기 위해 변조된 데이터와 파일럿/널 신호를 직접적으로 연산하지 말고 따로 구분하여 각각 독립적인 부호있는 정수(signed integer)로 표시한 후 그림 4와 같이 두 신호의 조합으로 표시한다. Butterfly 연산 시 변조 데이터부분은 변조 데이터부분과, 파일럿/널 신호 부분은 파일럿/널 신호 부분과만 연산을 하여 서로 영향을 미치지 않게 하고 두 번째 스테이지 이후의 곱셈이 이뤄지기 전에  $K_{mod}$ 가 곱해진 실제값으로 양자화한다.

변조 데이터의 정수표현은 constellation의 실수값이나 허수값을 정수로 표현하며 가능한 최소의 워드길이를 표현한다. Constellation의 실수(또는, 허수값)의 종류가  $r$  개라면 요구되는 워드길이  $W$ 는 다음과 같다.

$$W = 1 + \lceil \log_2(r-1) \rceil \tag{2}$$

(2)에서  $\lceil x \rceil$  는 ceiling 연산을 의미한다.

예로서 16-QAM의 경우를 살펴보자. 16-QAM의 constellation point는 그림 5와 같다. 그림 5에서 식 (1)의  $K_{mod}$ 는  $s$ 로 표현하였다. 그림 5의 실수축에서  $s=1$

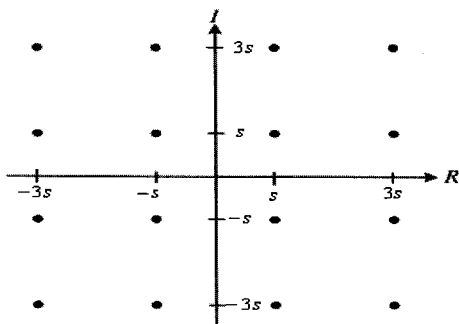


그림 5. 16-QAM의 constellation point  
Fig. 5. Constellation points of 16-QAM.

표 2. 16-QAM 입력의 signed integer 표현

Table 2. Signed integer representation for 16-QAM input signals.

	sign		
-3s	1	0	1
-s	1	1	1
s	0	0	1
3s	0	1	1

표 3. Pilot/null 신호의 signed integer 표현

Table 3. Signed integer representation for pilot and null signals.

	sign	
-p	1	1
0	0	0
p	0	1

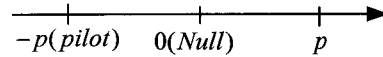


그림 6. Pilot 신호와 Null 신호

Fig. 6. Pilot and null signals

로 하면  $\{-3, -1, 1, 3\}$ 의 정수가 얻어진다. 이들 정수는 표 2에 보인 바와 같이 각각 3비트 수 (101, 111, 001, 011)로 표현할 수 있다. 허수축의 데이터도 동일한 방법으로 표현될 수 있다. 그림 6은 BPSK 변조되는 pilot 신호와 null 신호의 constellation를 나타낸다.

Normalization factor를  $p=1$ 로하고 constellation를 정수로 표현하면 표 3과 같다.

첫 번째 스테이지에서의 입력신호는 표 2와 표 3의 정수들의 조합으로 그림 4와 같이 나타낸다. 이 때 입력이 변조데이터이면 파일럿/널 신호 부분은 정수 '0'으로 표시하고 입력이 파일럿/널 신호이면 변조데이터부분을 정수 '0'으로 표시한다. 첫 번째 스테이지에서 연산할 때 앞에서 언급한 것처럼 변조데이터 부분은 변조데이터 부분과, 파일럿/널 신호 부분은 파일럿/널 신호 부분과만 연산을 한다. 각 부분들의 덧셈이나 뺄셈 연산 시 오버플로우를 피하기 위해 반드시 1비트 부호확장을 고려하여야한다.

그림 7은 첫 번째 스테이지에서 덧셈의 예를 보인다. 첫 번째 예는 변조데이터와 변조데이터간의 덧셈, 두 번째 예는 변조데이터와 파일럿/널 신호간의 덧셈에 해당된다. 16-QAM의 가능한 첫 번째 스테이지 연산의 결과는 표 4와 표 5의 조합으로 나타낼 수 있다. 두 번

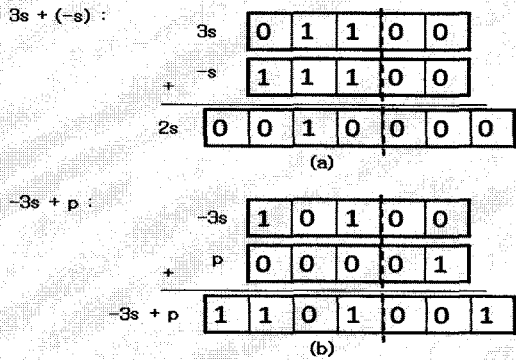


그림 7. 스테이지 1에서 덧셈 연산 예제:  
(a) 변조데이터 + 변조데이터,  
(b) 변조데이터 + 파일럿/널 신호

Fig. 7. Example of addition in Stage 1:  
(a) modulated data + modulated data  
(b) modulated data + pilot/null signal.

표 4. 스테이지 1을 통해 출력된 변조신호의 signed integer 표현

Table 4. Signed integer representation of the modulated data from stage 1.

	sign			
-6s	1	0	1	0
-4s	1	1	0	0
-3s	1	1	0	1
-2s	1	1	1	0
-s	1	1	1	1
0	0	0	0	0
s	0	0	0	1
2s	0	0	1	0
3s	0	0	1	1
4s	0	1	0	0
6s	0	1	1	0

표 5. 스테이지 1을 통해 출력된 파일럿/널 신호의 signed integer 표현

Table 5. Signed integer representation of the pilot/null signal from stage 1.

1	sign		
-2p	1	1	0
-p	1	1	1
0	0	0	0
p	0	0	1
2p	0	1	0

째 스테이지에서도 동일한 방법으로 연산이 이루어진다. 그림 8은 두 번째 스테이지에서 연산의 예를 보인다. 표 6과 표 7은 두 번째 스테이지에서 출력될 수 있는 모든 변조데이터와 파일럿/널 신호의 정수표현을 나

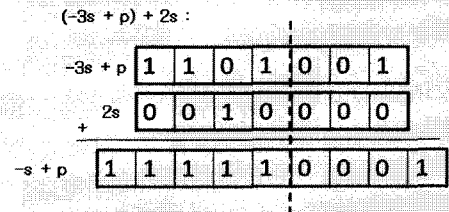


그림 8. 스테이지 2에서 덧셈 연산 예제  
Fig. 8. Example of addition in stage 2.

표 6. 스테이지 2를 통해 출력된 변조데이터 output의 signed integer 표현

Table 6. Signed integer representation of the modulated output from stage 2.

	sign				
-12s	1	0	1	0	0
-10s	1	0	1	1	0
-9s	1	0	1	1	1
-8s	1	1	0	0	0
-7s	1	1	0	0	1
-6s	1	1	0	1	0
-5s	1	1	0	1	1
-4s	1	1	1	0	0
-3s	1	1	1	0	1
-2s	1	1	1	1	0
-s	1	1	1	1	1
0	0	0	0	0	0
s	0	0	0	0	1
2s	0	0	0	1	0
3s	0	0	0	1	1
4s	0	0	1	0	0
5s	0	0	1	0	1
6s	0	0	1	1	0
7s	0	0	1	1	1
8s	0	1	0	0	0
9s	0	1	0	0	1
10s	0	1	0	1	0
12s	0	1	1	0	0

표 7. 스테이지 2를 통해 출력된 파일럿/널 신호 output의 signed integer 표현

Table 7. Signed integer representation of the pilot/null signal output from stage 2.

	sign			
-4p	1	1	0	0
-3p	1	1	0	1
-2p	1	1	1	0
-p	1	1	1	1
0	0	0	0	0
p	0	0	0	1
2p	0	0	1	0
3p	0	0	1	1
4p	0	1	0	0

타낸다. 표 2와 표 3에서 보인바와 같이 변조데이터와 파일럿/널신호는 그 종류가 정해져 있고 스테이지 1과 스테이지 2에서는 덧셈과 뺄셈연산만 수행하기 때문에 스테이지 2 연산의 결과를 표 6 및 표 7과 같이 쉽게 예측 할 수 있다. 따라서, 표 6과 표 7의 출력 신호에 변조방식에 따라 적절한 normalization factor( $K_{mod}$ )를 곱하고 시스템에서 요구하는 SQNR에 따라 양자화한 값을 look-up 테이블로 저장하여 활용할 수 있다. 스테이지 2의 최종 출력신호는 그림 9와 같이 얻어진다. 스테이지 2의 BF2 연산을 통해 얻어진 정수 출력들은 look-up 테이블의 주소로 사용되며 각 주소에는 다음 값이 저장된다.

$$\text{stored value} = Q(\text{modulated integer value} \times \frac{1}{4} \times K_{mod}) \quad (3)$$

(3)에서  $Q(x)$ 는  $x$ 의 양자화를 의미하며,  $1/4$ 은 각 스테이지에서 오버플로우를 방지하기 위해 첫번째와 두번째 스테이지에서 각각  $1/2$ 씩 스케일다운이 되었음을 의미한다.

그림 9에서 표 6에 해당하는 look-up 테이블은 표 6의 모든 경우에 대한 양자화 값을 필요로 하지만 다음의 과정을 거쳐 look-up 테이블 사이즈를 간략화 할 수 있다.

- 음수 출력( $-s, -2s, \dots, -10s, -12s$ )에 대한 양자화 값은 양수 출력에 대한 양자화 값의 보수를 취하여 구한다
- $2^i \times s$ 에 해당하는 출력들( $2s, 4s, 8s$ )의 양자화 값은  $s$ 에 해당하는 출력을  $i$ 자리만큼 쉬프트하여 구한다
- $3 \times 2^i \times s$ 에 해당하는 출력들( $6s, 12s$ )의 양자화 값은  $3s$ 에 해당하는 출력을  $i$ 자리만큼 쉬프트하여 구한다
- $5 \times 2^i \times s$ 에 해당하는 출력들( $10s$ )의 양자화 값은  $5s$ 에 해당하는 출력을  $i$ 자리만큼 쉬프트하여 구한다

이상을 종합하면 그림 9에서 표 6에 해당하는 look-up 테이블은  $s, 3s, 5s, 7s, 9s$ 에 해당하는 양자화 값만 포함하고 있으면 됨을 알 수 있다.

파일럿/널신호 부분은  $K_{mod}=1$ 로 고정되어있어서 별도의 곱셈이 필요 없으므로 그림 9에서 이에 해당하는 look-up 테이블을 제거할 수 있다. 다만,  $1/4$ 의 스케일다운을 고려하여 표 7에서 사인비트를 제거하고 그 다

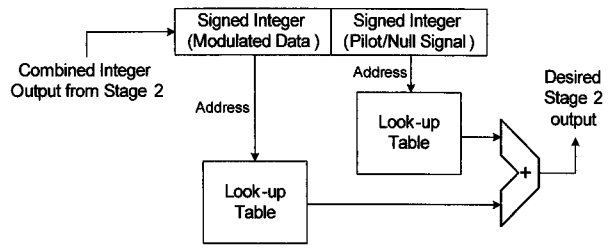


그림 9. 스테이지 2에서 desired output의 계산  
Fig. 9. Computation of desired output in stage 2.

음 비트가 MSB로 취급되며 MSB 다음에 binary point가 존재하는 것으로 간주한다. 이 때  $(4p) \times (1/4) = p$ 에 해당하는 출력은 정해진 워드길이 만큼  $0.11\dots1$ 로 대체하여야한다.

그림 1에 보인 바와 같이 스테이지 3 부터는 기존의 방법과 동일하게 연산이 이루어진다.

#### IV. 시뮬레이션 및 비교

##### 1. Verilog 시뮬레이션

그림 10의 구조와 같은 CR을 위한 Radix- $2^4$  SDF 2048포인트 IFFT를 제안한 구조와 기존의 구조로 Verilog HDL로 코딩하였고 삼성 0.13um 라이브러리를 이용해 합성하여 면적과 파워를 비교하였다. CR시스템에서는 BPSK, QPSK, 16-QAM, 64-QAM 변조방식을 다 지원해야 하며 SQNR이 70 dB이상이어야 한다. 변조된 신호는 12비트로 양자화되어 첫 번째 스테이지로 입력된다. 첫 번째 메모리와 두 번째 메모리의 워드길

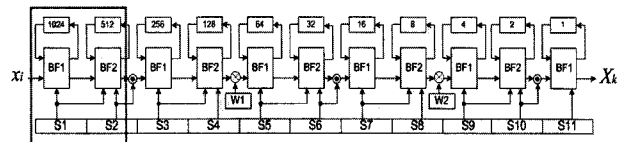


그림 10. Radix- $2^4$  SDF 2048 포인트 IFFT 구조  
Fig. 10. Radix- $2^4$  SDF 2048-point IFFT structure.

표 8. 제안한 구조와 기존구조의 결과 비교  
Table 8. Comparison of the proposed method and the existing method.

	기존	제안	감소율
면적(gates)	307,788 (100%)	201,162 (65.3%)	106,626 (34.7%)
파워(mW)	92.5453 (100%)	61.6157 (66.6%)	30.9296 (33.4%)

이는 각각 13비트, 14비트이다.

그림 10에서 박스 표시한 부분을 제안한 방법과 mapping을 사용하지 않은 기존의 방법으로 설계하여 표 8에서 비교하였으며 제안한 방법에 의해 34.7%의 면적 감소와 33.4%의 파워감소를 얻을 수 있음을 알 수 있다.

## V. 결 론

본 논문에서는 IEEE 802.22를 포함한 무선 고속 데이터 전송을 위해 사용되는 OFDM 시스템의 핵심 블록인 IFFT에서 큰 면적을 차지하는 첫 번째 스테이지와 두 번째 스테이지의 메모리를 줄이기 위해 Radix-2<sup>4</sup> SDF 구조에 적용할 수 있는 IFFT의 효율적인 설계방법을 제시하였다. 제안방법을 이용하여 Cognitive Radio용 Radix-2<sup>4</sup> SDF 2048포인트 IFFT를 설계하였다. 기존의 IFFT 구조로 설계한 경우와 비교하여 면적과 파워에서 약 34.7%, 33.4%의 감소를 얻었다.

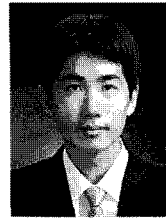
본 논문에서 제안한 방법은 높은 정밀도를 요구하여 긴 워드길이 요구되는 무선 통신시스템 및 요구되는 IFFT의 포인트가 큰 시스템에 사용될 경우 더 큰 이득을 가져 올 수 있다.

## 참 고 문 헌

- [1] S. He and M. Torkelson, "A new approach to pipeline FFT processor," in *Proc. 10th International Parallel Processing Symp. (IPPS '96)*, pp. 766-770, 1996.
- [2] S. He and M. Torkelson, "Designing pipeline FFT processor for OFDM (de)Modulation," in *Proc. IEEE URSI Int. Symp. Signals, Syst., Electron.*, pp. 257-262, 1998.
- [3] S. Yu and E. E. Swartzlander, Jr., "A pipelined architecture for the multidimensional DFT," *IEEE Transactions on Signal Processing*, vol. 49, pp. 2096-2102, 2001.
- [4] J. Y. Oh and M. S. Lim, "New radix-2 to the 4th power pipeline FFT processor," *IEICE Trans. Electron.*, vol.E88-C, no. 8, pp.1740-1746, Aug. 2005.
- [5] IEEE 802.22/ Draft Standards for Wireless Regional Area Networks Part 22: Cognitive Wireless RAN Medium Access Control(MAC) and Physical Layer(PHY) specifications, 2006.

- [6] In-Gul Jang, Yong-Eun Kim, Yi-Nan Xu and Jin-Gyun Chung, "Efficient IFFT design using mapping method," in *Proc. IEEE APCCAS 2008*, pp. 878-881, Nov. 2008.

## 저 자 소 개



이 재 경(학생회원)  
2009년 전북대학교  
전자공학과 학사 졸업  
2009년~현재 전북대학교  
전자정보공학부 석사재학  
<주관심분야 : 통신, 컴퓨터, 신호처리, 반도체>



장 인 걸(학생회원)  
2005년 전북대학교  
전자공학과 학사 졸업  
2007년 전북대학교 정보통신  
공학과 석사 졸업  
2008년~현재 전북대학교  
전자정보공학부 박사과정  
<주관심분야 : 통신, 신호처리, 반도체>



정 진 균(정회원)  
1985년 전북대학교  
전자공학 학사 졸업  
1989년 미국 미네소타 주립대학  
전기공학 석사 졸업  
1991년 미국 미네소타 주립대학  
전기공학 박사 졸업  
<주관심분야 : 통신, 컴퓨터, 신호처리, 반도체>



이 철 등(정회원)  
1977년 경북대학교  
전자공학과 학사 졸업  
1989년 한양대학교  
전자공학과 석사 졸업  
1997년 충북대학교  
컴퓨터공학과 박사 졸업  
1994년~현재 전자부품연구원  
센터장/본부장/수석  
<주관심분야 : 통신, 컴퓨터, 신호처리, 반도체>