

논문 2005-42SD-11-5

지상파 DMB 모뎀용 R2SDF/R2SDC 하이브리드 구조의 FFT/IFFT 코어 설계

(A Design of FFT/IFFT Core with R2SDF/R2SDC Hybrid Structure
For Terrestrial DMB Modem)

이진우*, 신경욱**

(Jin-Woo Lee and Kyung-Wook Shin)

요 약

본 논문에서는 지상파 DMB 단말기 모뎀의 핵심 기능블록으로 사용되는 FFT/IFFT 코어(FFT256/2k)를 설계하였다. 설계된 코어는 Eureka-147 전송 규격에 명시된 4가지 전송모드를 지원할 수 있도록 256/512/1204/2048점 FFT/IFFT를 선택적으로 수행하도록 설계되었다. R2SDF와 R2SDC 구조를 혼합하여 적용함으로써 메모리 용량을 최소화 하였으며, R2SDC 단일 구조로 구현한 경우에 비해 메모리 크기를 약 62% 감소시켰다. 또한 TS_CBFP(Two Step Convergent Block Floating Point)를 사용하여 SQNR를 향상시켰으며, 50MHz@2.5-V로 동작하는 경우 2048점 FFT/IFFT 연산에 41- μ s가 소요되었다. Verilog-HDL로 설계된 코어는 0.25- μ m CMOS Cell 라이브러리로 합성한 결과 약 68,400개의 게이트와 58,130 비트의 메모리로 구현되었으며, switching activity를 산출하여 전력소모를 측정된 결과 2048점 FFT의 경우 113-mW의 전력을 소모하는 것으로 추정되었다. 설계된 코어를 FPGA에 구현하여 동작시킨 결과 정상 동작을 검증하였으며, 전체 평균 50-dB 이상의 SQNR 성능을 보였다.

Abstract

This paper describes a design of FFT/IFFT Core(FFT256/2k), which is an essential block in terrestrial DMB modem. It has four operation modes including 256/512/1204/2048-point FFT/IFFT in order to support the Eureka-147 transmission modes. The hybrid architecture, which is composed of R2SDF and R2SDC structure, reduces memory by 62% compared to R2SDC structure, and the SQNR performance is improved by TS_CBFP(Two Step Convergent Block Floating Point). Timing simulation results show that it can operate up to 50MHz@2.5-V, resulting that a 2048-point FFT/IFFT can be computed in 41- μ s. The FFT256/2k core designed in Verilog-HDL has about 68,400 gates and 58,130 RAM. The average power consumption estimated using switching activity is about 113-mW, and the total average SQNR of over 50-dB is achieved. The functionality of the core was fully verified by FPGA implementation.

Keywords : FFT/IFFT, DMB, CBFP, Eureka-147, OFDM

I. 서 론

국내의 디지털 라디오 방송은 초기의 음성 및 데이터

서비스에서 동영상 멀티미디어 서비스까지 확대되어 2003년 초기에 명칭도 '디지털오디오방송(DAB)에서 '디지털멀티미디어방송(DMB)'으로 변경되었으며, 기술기준 및 송수신 정합표준도 이에 부합되도록 추진되어 오고 있다. DMB는 "CD 수준의 음질과 데이터 또는 영상 서비스 등이 가능하고 우수한 고정 및 이동수신 품질을 제공하는 디지털 방식의 멀티미디어 방송"으로 정의된다. 국내에서는 올해부터 각 방송사에서 본 방송을 실시함에 따라 그 관심도 급증하고 있다. 국내 지상파 DMB 전송 표준은 TTA(한국정보통신기술협회)에서

* 준회원, ** 정회원, 금오공과대학교 전자공학부
(School of Electronic Eng., Kumoh National
Institute of Technology)

※ 2004년도 IT SoC 핵심설계인력양성 사업의 SoC
전공실습프로젝트지원에 의한 연구결과의 일부임.

※ 반도체설계교육센터(IDEC)의 CAD Tool 지원에
감사드립니다.

접수일자: 2005년6월15일, 수정완료일: 2005년11월3일

2003년 10월 24일에 '초단파 디지털 라디오방송 송수신 정합표준'을 제정 하였다. 제정된 표준은 국제표준인 'ETSI EN 300 401 V1.3.3'을 기준으로 작성 되었으며, 이 국제표준은 유럽형 방식인 Eureka-147 표준 중 하나이다.^[1] 제정된 표준의 전송 규격은 4가지의 전송 모드가 있으며, 각 모드에 따라 전송 캐리어의 개수와 시간은 다음과 같다. Mode I인 경우, 캐리어의 개수는 1,536개이며 전송 시간은 1.246-ms이다. Mode II이면 캐리어의 개수는 384개, 전송 시간은 312-us 이며, Mode III인 경우는 캐리어의 개수는 192개, 전송 시간은 156-us이다. 그리고 Mode IV인 경우 캐리어의 개수는 768개, 전송 시간은 623-us이다. 그러므로 각 전송 모드를 지원하기 위해서는 256, 512, 1024, 2048점 FFT/IFFT를 연산 할 수 있는 코어가 필요하다.

Eureka-147은 전송 방식으로 OFDM (Orthogonal Frequency Division Multiplexing)을 사용 하고 있다. OFDM 방식은 여러 개의 반송파를 사용하는 다수 반송파 전송의 일종으로 반송파의 수만큼 각 채널에서의 전송주기가 증가하게 된다. 따라서 자연 환경에서의 산이나 도심지의 빌딩 등에서 반사되는 전파가 일으키는 다중경로에 의한 왜곡이나 페이딩 환경에 강인한 성능을 보인다. OFDM은 FEC(Forward Error Correction) coding, interleaving, modulation, IFFT(Inverse Fast Fourier Transform)/FFT 등의 기능 블록으로 구성되며, 그 중 FFT/IFFT는 다수 반송파의 변복조를 수행하는 핵심 블록이다.^[2]

국내의 경우 SIPAC에 등록 되어 있는 FFT IP들은 총 15개가 등록이 되어있으며 2003년도 이후 등록된 IP는 12개이다. 등록된 IP는 대부분이 IEEE 802.11a를 목표로 설계하였으며, 알고리즘은 Mixed Radix를 사용하였다. 본 논문에서 설계한 FFT256/2k 코어와 같이 멀티모드로 동작하는 IP는 DVB-T System을 목표로 R4SDC와 R4SDF 구조를 갖는 Hybrid 8k/2k FFT/IFFT IP가 있다.

국외의 경우 상용화된 FFT IP를 제공하는 회사들이 있으며 크게 Altera, RF Engines Ltd., Xilinx, QuickLogic, IcommTech 등이 있다. Altera에서는 자사에서 생산되는 device에 맞도록 'FFT Megacore Function'이라는 생성기 프로그램을 제공 하고 있다.^[3] 이 생성기는 Radix-4를 기반으로 하고 있으며 여러 가지 옵션을 사용자가 할 수 있도록 설계 되어 있다. 그 기능을 살펴보면 64~8192점까지 가능하며 또한 입출력, 격자계수, 내부 비트를 조절 할 수 있다. 성능면에서는 동작 연산 속도를 증가시키기 위해 BF(Butterfly)연

산 후 격자 계수를 곱할 때 복소수 승산기의 개수를 1개 또는 3개를 사용 할 수 있도록 하고 있으며, SNR(Signal Noise Ratio)를 높이기 위해 BFP(Block Floating Point)를 적용 하였다. RF Engines Ltd.에서는 High Speed, Guard Speed, High Length, Mixed Radix 등으로 구분하여 제공하고 있으며, 나머지 Xilinx, QuickLogic, IcommTech 에서는 각 포인트별로 FFT Core를 IP로 제공 하고 있다.

본 논문에서는 국내 지상파 DMB용 FFT/IFFT 코어를 설계했다. 제정된 표준의 각 모드에 따라 4가지 모드로 동작하는 FFT/IFFT 코어가 필요하며, 2048점 FFT/IFFT인 경우 약 1.2-ms 내에 동작을 하여야 한다. 따라서 본 논문에서는 256, 512, 1024, 2048점 FFT/IFFT를 선택적으로 처리하는 구조를 가지는 FFT/IFFT 코어를 설계하였다. 설계된 코어는 2048점 FFT/IFFT 코어인 경우 41-us내에 동작이 가능하므로 전송 규격의 요구사항을 충분히 만족하는 것으로 판단 되었다.

II. FFT 알고리즘과 아키텍처

1. FFT 알고리즘

데이터 x_n 에 대한 N 점 이산 푸리에 변환(Discrete Fourier Transform; DFT)은 식 (1)과 같이 정의된다.

$$X_k = \sum_{n=0}^{N-1} x_n \cdot W_N^{nk} \quad (1)$$

$$\text{단, } W_N^{nk} = e^{-j\frac{2\pi}{N}nk}, \quad k = 0, 1, 2, \dots, N-1$$

DFT는 N^2 번의 복소수 승산과 $N(N-1)$ 의 복소수 가산으로 계산되므로 N 이 증가함에 따라 연산량이 지수적으로 증가한다. FFT는 식(1)의 연산량을 줄인 고속 연산 알고리즘이며, 대표적인 알고리즘으로는 Cooley-Turkey FFT 알고리즘, Winograd FFT 알고리즘, Prime Factor 알고리즘, Split-Radix FFT 알고리즘 등이 있다. FFT 알고리즘은 크게 나누어 radix-2, radix-4, radix-8과 같은 fixed-radix 알고리즘과 radix-2/4, radix-2/8, radix-2/4/8 등과 같은 mixed-radix 알고리즘으로 구분할 수 있다. 표 1은 FFT 알고리즘들을 연산량과 연산의 규칙성 측면에서 비교한 것이다. fixed-radix 알고리즘은 연산구조가 규칙적이므로 VLSI 구현에 적합하다는 장점은 있지만, mixed-radix 알고리즘에 비해 많은 연산량을 필요로

표 1. FFT 알고리즘의 비교

Table 1. Comparison of FFT algorithms.

FFT Algorithms	Complex Multiplications	Constant Multiplications	Spatial Regularity
Radix-2	98	-	O
Radix-4	76	-	O
Radix-2 ²	76	-	O
Split-radix	72	-	X
Radix-2/8	44	38	X
Radix-2 ³	48	32	O
Radix-2/4/8	48	32	O

한다. 반면에 mixed-radix 알고리즘은 연산량은 작으나 연산구조가 불규칙적이므로 하드웨어 구현이 어렵다.

FFT 알고리즘들은 각기 고유의 연산구조와 연산량을 가지며, 이들은 연산시간, 칩 면적, 그리고 전력소모 등에 직접적인 영향을 미친다. 따라서 효율적인 FFT 프로세서의 구현을 위해서는 연산량과 연산 흐름의 규칙성이 함께 고려되어야 한다^[4-5].

본 논문에서는 연산량 측면에서 최적에 가깝고, 규칙적인 연산 흐름을 가져 VLSI 구현에 적합하며, radix-2 나비연산 회로로 구현될 수 있는 radix-2/4/8 알고리즘을 선택하였으며, CBFP 스케일링의 효율적인 적용을 위해 나비연산 후에 격자계수 곱셈이 이루어지는 DIF 방식을 채택하였다.

2. FFT 아키텍처

FFT의 구현 방식은 시스토크 배열(systolic array) 구조, 파이프라인 구조, 그리고 column FFT 구조 등 다양한 하드웨어 아키텍처로 구현 될 수 있다.

파이프라인 FFT 구조는 각 연산 stage에 하나 또는 그 이상의 나비 연산기를 할당하고 연산 stage들을 파이프라인방식으로 동작시키는 구조이다. 이 방식은 다른 구조에 비해 비교적 적은 하드웨어로 높은 처리율을 얻을 수 있고 순차적인 입·출력을 제공하므로 고속 연산이 요구되는 분야에 많이 사용된다. 파이프라인 FFT 구조는 지연 메모리의 구현 방식에 따라 delay feedback(DF) 방식과 delay commutator(DC) 방식으로 구분되며, FFT 알고리즘의 radix와 병렬성에 따라 R2MDC, R2SDF, R2SDC, R4MDC, R4SDF, R4SDC, R22SDC 등 다양한 형태로 구현이 가능하다. DF 구조는 적은 메모리를 필요로 하나, 나비 연산기의 효율이 작고 CBFP 스케일링의 구현이 용이하지 않다. 반면에 DC 구조는 DF 구조에 비해 많은 메모리를 필요로 하지만 나비 연산기의 이용 효율이 높고 CBFP 스케일링

의 구현이 용이하다는 장점을 갖는다.

본 논문에서는 적은 메모리로 구현이 가능한 R2SDF 구조와 나비 연산기의 이용 효율이 높고, 데이터 흐름 구조가 단순하여 CBFP 스케일링의 구현이 용이한 R2SDC 구조를 혼합한 hybrid 구조를 채택함으로써, 메모리 용량을 최소화하고 성능을 높였다. CBFP가 적용되는 첫 번째 stage는 R2SDC 구조를 나머지 stage는 R2SDF 구조를 사용했다. 또한 버퍼 메모리를 사용하지 않는 TS_CBFP 방법^[6]을 적용함으로써 적은 메모리만으로도 CBFP 스케일링이 구현되도록 하였다.

III. FFT 코어 설계

1. 아키텍처 설계

설계된 FFT/IFFT 코어는 Eureka-147 규격의 4가지 전송모드를 지원하도록 256, 512, 1024, 2048점 FFT/IFFT를 선택적으로 처리하는 구조로 설계되었으며, 이를 위해 radix-2, radix2/4, radix-2/4/8 구조를 복합적으로 사용하였다. 그림 1은 FFT256/2k 코어의 아키텍처를 보인 것이다. 256점 FFT 연산 블록을 기본으로 하고, 3개의 radix-2 연산 stage를 선택적으로 동작시켜 512, 1024, 2048점 FFT를 처리하는 구조로 설계되었다. FFT256/2k 코어의 기본 연산 블록인 256점 FFT는 그림 2와 같이 구현되며 2개의 8점 FFT 연산 블록, 1개의 4점 FFT 연산 블록, 그리고 2개의 복소수 승산기로 이루어진다. 8점 FFT 연산 블록은 radix-2/4/8 알고리즘에 의해 얻어진 연산 흐름도 그림 3-(a)를 구현하는 블록이며, 그림 3-(b)와 같이 파이프라인 구조로 구현된다. 그리고 4점 FFT 연산 블록은 8점 FFT 연산 블록에서 상수곱셈기를 포함하는 stage_B를 제거하고 stage_A와 stage_C로 구현 된다.

FFT256/2k 코어의 전체 구조를 블록별로 표현하면 그림 4와 같다. 점선 박스 내의 블록이 기본 연산 단위의 256점 FFT연산 블록이며, 앞단의 3개의 radix-2 연산블록과 멀티플렉서를 이용하여 256, 512, 1024, 2048

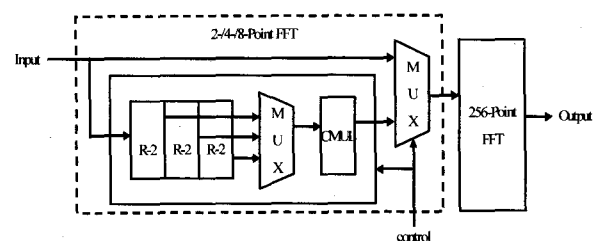


그림 1. FFT256/2k 코어의 아키텍처

Fig. 1. Overall architecture of FFT256/2k core.

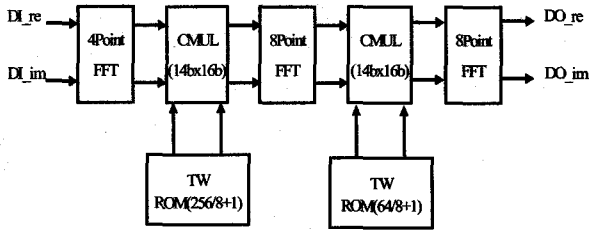
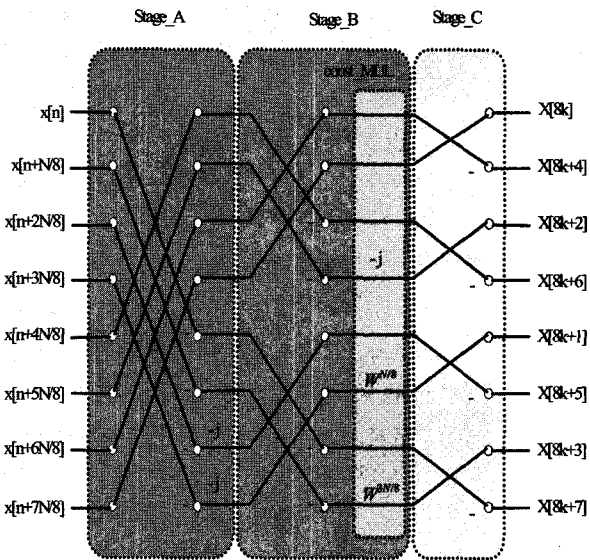
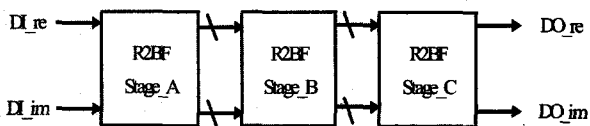


그림 2. 256점 FFT 코어의 구조
Fig. 2. Architecture of 256-point FFT core.



(a) signal flow graph



(b) pipelined implementation

그림 3. 8점 FFT의 신호 흐름도와 파이프라인 구현
Fig. 3. Signal flow graph and its pipelined implementation of 8-point FFT block.

점 FFT/IFFT 연산을 구현한다. FFT256/2k 코어는 11개의 radix-2 나비연산 블록과 3개의 Multiplexer, 3개의 복소수 승산기, 그리고 FFT와 IFFT를 선택적으로 연산하기 위한 complex conjugator 블록 등으로 구성된다. 한편, CBFP 스케일링을 처리하기 위해서 복소수 승산기의 출력을 입력으로 받는 radix-2 연산블록은 R2SDF 구조로는 설계가 불가능하여 R2SDC 구조로 설계되었다. CBFP 스케일링을 구현하기 위해서는 각 연산 블록의 데이터를 메모리에 저장하고 있어야 하나 R2SDF 구조는 연산 블록의 1/2 만큼의 데이터만 메모리에 저장하므로 CBFP구현이 불가능하다. 그리고 나머지 연산 블록은 메모리의 감소를 위해 R2SDF 구조로 설계되었다.

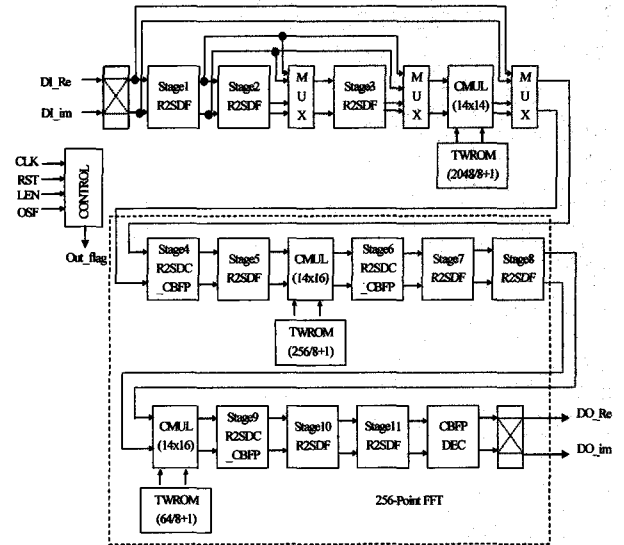


그림 4. FFT256/2k 코어의 상세 구조
Fig. 4. Detailed architecture of FFT256/2k core.

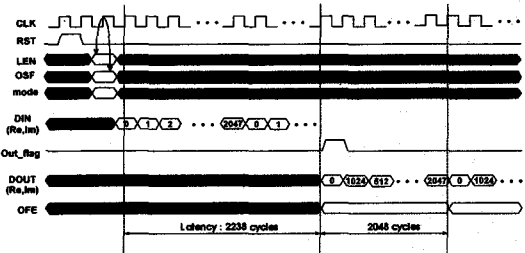


그림 5. FFT256/2k 코어의 2048점 동작 타이밍도
Fig. 5. Timing diagram of FFT256/2k core. (2048-point mode)

그림 5는 FFT256/2k 코어의 동작 모드 중 2048점 FFT의 동작 타이밍을 나타낸 것이다. 초기에 리셋신호 (rst)가 인가된 후, 동작모드를 결정하는 mode 신호 (mode=0이면 IFFT, mode=1 이면 FFT)와 FFT 길이를 지정하는 신호 (len=0 이면 256점, len=1 이면 512점, len=2 이면 1024점, len=3 이면 2048점)가 로드 된다. 첫 번째 데이터가 입력 된 후 2,238 클럭 주기의 latency를 지나 결과가 출력되기 시작하며, 데이터 출력의 시작을 알리는 신호 (out_flag)가 외부로 출력된다.

2. R2SDF 연산 Stage

그림 4의 상세구조에서 CBFP를 적용하지 않은 radix-2 연산 stage(즉, stage1, stage3, stage5, stage8, stage11)는 R2SDF 구조를 바탕으로 그림 6과 같이 구현하였으며, 공통적으로 나비 연산과 -j 곱셈을 포함한다. 이때 -j 곱셈은 복소수 데이터의 공액을 구한 후, 허수부의 부호 반전에 의해 연산될 수 있으며, 본 논문에서는 이를 radix-2 나비 연산에 포함시켜 처리함으로써 회로를 단순화 하였다.

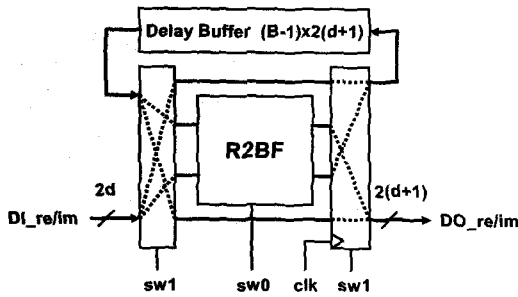


그림 6. R2SDF stage의 블록도
Fig. 6. Block diagram of R2SDF stage.

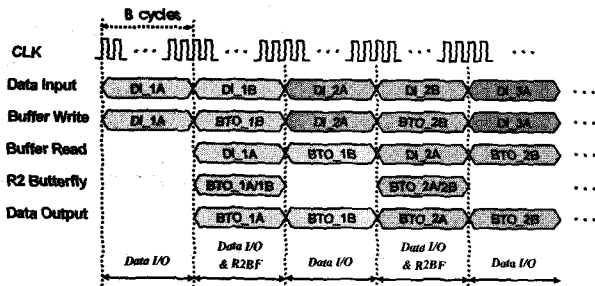


그림 7. R2SDF stage의 동작 타이밍
Fig. 7. Timing diagram of R2SDF stage.

그림 7은 R2SDF 연산 stage의 동작 타이밍을 나타낸 것이다. 타이밍 도에서 볼 수 있듯이, 나비 연산기는 전체 동작기간의 절반 동안에만 동작하고 나머지 기간에는 데이터 입·출력 동작만 일어나게 된다. 이와 같은 R2SDF stage의 동작 특성을 토대로 데이터 입·출력 동작만 일어나는 기간에는 나비 연산기가 동작을 멈추도록 설계함으로써 불필요한 스위칭 동작을 제거하여 전력소모가 최소화되도록 하였다.

3. R2SDC 연산 Stage

그림 4의 상세 구조에서 stage4, stage6, stage9는 CBFP 스케일링을 갖는 R2SDC이며, 그림 8과 같이 구성된다. 일반적인 CBFP는 연산 stage의 블록 크기에 비례하는 버퍼 메모리가 필요하다. 그러나 본 논문에서는 별도의 버퍼 메모리를 사용하지 않는 TS_CBFP 스케일링 방법을 적용함으로써 메모리 용량을 크게 감소시켰다.

TS_CBFP 스케일링의 기본 방법은 다음과 같다. CBFP 예비 스케일러에 입력되는 28비트 데이터의 실수부와 허수부에서 부호비트와 동일하면서 연속된 '0' 또는 '1'의 개수 즉, NLB(Number of Leading Bits)를 찾아 이를 PSI(Pre-Scaling Index) 레지스터에 저장된 값과 비교하여 작은 값으로 레지스터를 갱신한다. 한편, 각 데이터의 실수부와 허수부는 PSI에 14비트로 예비 스케일링된 후 해당 PSI 값과 함께 지연 변환기에 저장

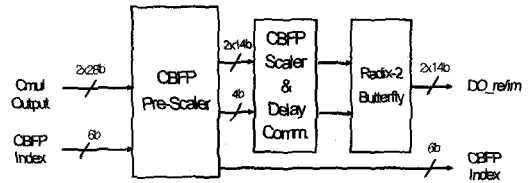


그림 8. TS_CBFP 스케일링을 갖는 R2SDC stage
Fig. 8. R2SDC stage with TS_CBFP scaling.

된다. 데이터 블록내의 모든 데이터에 대해 이와 같은 예비 스케일링이 완료되면 해당 데이터 블록의 BSI(Block Scaling Index) 값이 결정되고, 지연 변환기의 메모리에 저장된 데이터는 해당 PSI 값의 차이만큼 스케일링 보정된 후, 나비 연산기로 입력된다.

4. CBFP 디코더

각 연산 단계에서 적용된 CBFP 스케일링은 데이터 블록내의 가장 큰 데이터를 기준으로 정렬 증가 (scaling-up)시키는 과정이다. 따라서 최종 stage를 거친 데이터는 최종 CBFP 지수 값만큼 정렬 감소 (scaling-down)시켜 출력해야 한다. 이와 같은 CBFP 디코딩 과정은 CBFP 지수를 이용한 오른쪽 쉬프트로써 데이터의 최상위 비트를 확장한 만큼 최하위 비트쪽을 잘라 버리는 과정이다. 한편, 최하위 비트를 잘라내면 최종 출력의 정확성이 떨어져 SQNR이 감소하므로, 표 2와 같이 외부로부터 출력 이득조정 신호(OSF)를 인가하여 스케일링 되도록 함으로써 최종 출력의 정확성을 향상시킬수 있다. 출력 이득조정을 갖는 CBFP 디코더는 그림 9 와 같이 설계되었으며, 감산기를 통해 CBFP 지수와 이득조정 신호의 차를 구한 후, 스케일러에서 해당 비트만큼 쉬프트시켜 최종 데이터를 출력한다. 한편, OSF 값이 너무 커서 부호 비트 수 이상의 쉬프트가 이루어지는 경우 오버플로 감지회로에서 이를 감지하여 출력오버플로신호(OFE)를 발생 시킨다.

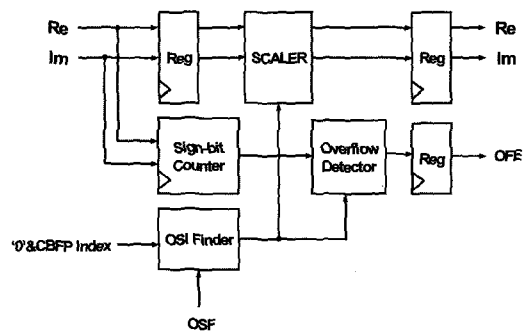


그림 9. 출력 이득 조절을 갖는 CBFP 디코더
Fig. 9. CBFP decoder with output gain control.

표 2. OSF에 따른 출력 스케일링 상수
Table 2. Output scaling factor according to OSF.

OSF	Additional left shifting(bits)	Scaling Factor
000	0	2^{-7}
001	1	2^{-6}
010	2	2^{-5}
011	3	2^{-4}
100	4	2^{-3}
101	5	2^{-2}
110	6	2^{-1}
111	7	2^0

IV. 설계 검증 및 성능 분석

FFT256/2k 코어는 Verilog-HDL로 모델링 되었으며, ModelSim과 Matlab을 사용하여 논리검증과 성능평가를 수행하였다. 그림 10은 성능평가 과정을 보인 것이며, 2진 랜덤 신호를 생성하여 변조(QPSK, 16-QAM, 64-QAM)한 후, 부동점(floating-point)연산을 갖는 이상적인 IFFT와 이득 조정을 거쳐 10비트로 양자화된 데이터를 시뮬레이션 입력으로 사용하였다.

SQNR은 식 (2)에 의해 계산 되었으며 수식에서 A는 Matlab을 이용한 부동소수점으로 연산된 결과 값을 나타내며 B는 설계된 FFT256/2k 코어로 연산된 결과 값을 나타낸다. 그림 11-(a)는 2048점 FFT 모드의 시뮬레이션 결과(출력이득이 8인 경우)이다. 시뮬레이션에 사용된 입력 데이터는 좌측 상단의 성상도와 같으며, 좌측 하단의 성상도는 Matlab에서 얻어진 이상적 FFT 출력이고, 우측 하단의 성상도는 설계된 FFT256/2k 코어에서 얻어진 FFT 출력이다. 연산 오차는 우측 상단의 성상도와 같이 매우 작은 범위이며, 따라서 설계된 FFT256/2k 코어의 연산 정확도가 우수함을 알 수 있다. 연산 정확도는 SQNR 분석을 통해서도 알 수 있으며, 그림 11-(b)는 연산오차와 SQNR 특성을 주파수 별로 보인 것이다. 2048점 FFT는 출력이득이 8인 경우 약 58-dB 정도의 성능을 보이고 있으며, 전체 주파수 영역에서 우수한 성능을 나타내고 있다.

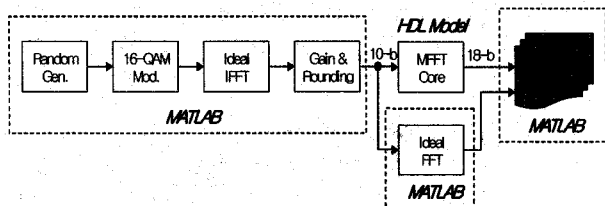
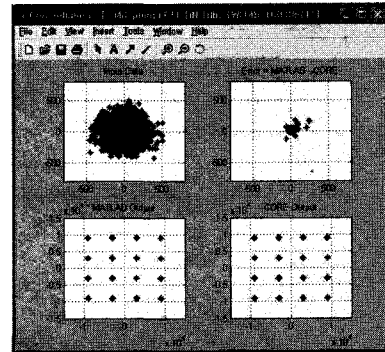
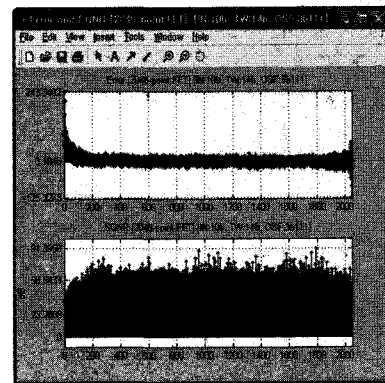


그림 10. FFT256/2k 코어의 성능 평가 방법
Fig. 10. Performance evaluation method of FFT256/2k core.



(a) constellations



(b) error and SQNR

그림 11. FFT256/2k 코어의 성능 평가
Fig. 11. Performance evaluation of FFT256/2k core.

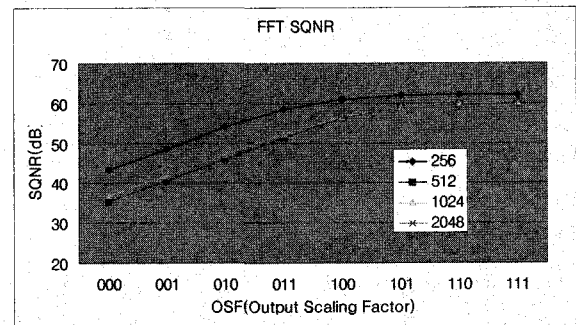


그림 12. 출력이득에 따른 SQNR 특성
Fig. 12. SQNR according to output scaling characteristics.

$$SQNR = \frac{\sum [Re(A)]^2 + \sum [Im(A)]^2}{\sum [Re(A) - Re(B)]^2 + \sum [Im(A) - Im(B)]^2} \quad (2)$$

그림 12는 FFT256/2k 코어의 출력이득에 따른 SQNR 특성을 분석한 결과이다. FFT256/2k 코어는 출력이득이 8인 경우, 각 모드에서 모두 58-dB 이상의 SQNR이 얻어졌으며, FPGA 보드에 구현하여 얻은 결과값과 동일한 결과가 얻어졌다.

표 3. FFT256/2k 코어의 메모리 사용량 비교
Table 3. Comparison of memory requirement for FFT256/2k.

Stage	this paper	R2SDC Structure	R2SDF Structure
Stage1	20,480	69,632	20,480
Stage2	12,288	37,888	12,288
Stage3	7,168	22,016	7,168
Stage4	11,776	11,776	2,560
Stage5	2,304	7,040	2,304
Stage6	2,944	2,944	1,088
Stage7	512	1,568	512
Stage8	288	880	288
Stage9	368	368	134
Total	58,128	154,112	46,822

표 4. FFT 길이에 따른 전력 소모 결과
Table 4. Power consumption according to FFT length.

FFT Length	256	512	1024	248
Total Power(mW)	95.32	109.71	115.56	113.03

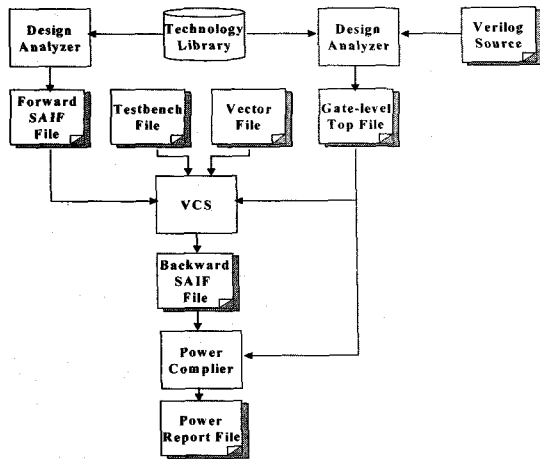


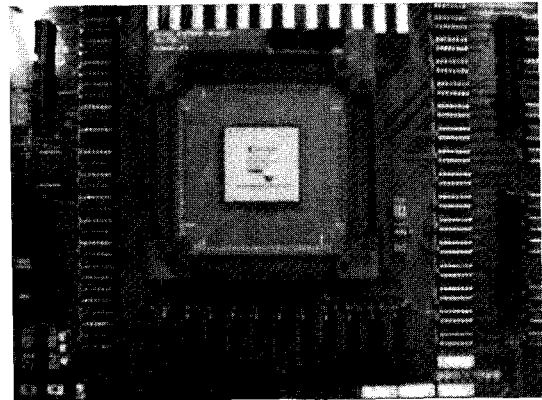
그림 13. 전력 소모 측정 방법
Fig. 13. Estimation of power consumption.

설계된 FFT256/2k 코어를 0.25- μ m CMOS 셀 라이브러리로 합성한 결과 약 68,400 게이트와 58,130 비트의 RAM으로 구현되었다. R2SDF/R2SDC hybrid 구조와 TS_CBFP 스케일링을 적용하여 설계된 FFT256/2k 코어는 2048점 FFT인 경우 메모리 사용량은 표 3과 같으며 R2SDC 단일구조의 메모리 사용량이 154,112 비트인 반면, 58,128비트로 구현되어 약 62%의 메모리 감소가 얻어졌다.

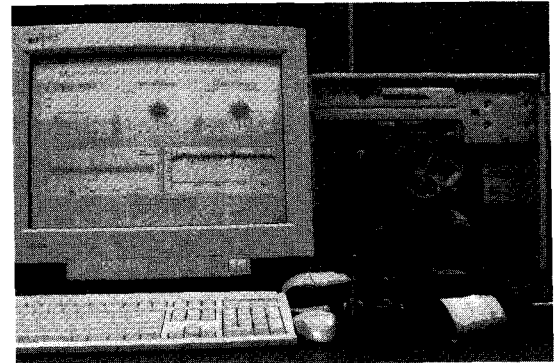
그림 13은 전력 소모 측정 방법을 나타낸 것이다. CMOS 셀 라이브러리와 설계한 코어를 Design Analyzer를 사용하여 forward switching activity 파일과 Verilog 로된 gate 레벨 top 파일을 생성한다. 생성된 두 파일과 testbench 파일, 입력 벡터 파일 총 4개의

표 5. FFT 프로세서의 비교
Table 5. Comparison of FFT Processors.

	Ref.[7]	Ref.[8]	Ref.[9]	This Paper
FFT Length	1024	512	1024	2048
Input Data Width	16	16	16	10
Twiddle Factor Width	16	16	16	14
Slices	?	7,225	7,365	13,473
Latency	4,096	?	1,099	2,238



(a) 검증용 FPGA 보드



(b) 검증 시스템

그림 14. FPGA를 이용한 FFT256/2k 코어의 검증
Fig. 14. Verification of FFT256/2k core using FPGA.

파일을 VCS를 사용하여 backward switching activity 파일을 생성한다. backward switching activity 파일과 gate 레벨 top 파일을 Power Compiler를 사용하여 최종 전력 소모를 측정하였다. 측정된 전력소모는 표 4와 같으며, 2048점 FFT인 경우 113-mW 로 측정되었다.

타이밍 시뮬레이션 결과 최대 지연시간은 약 14-ns로 나타났으며, 레이아웃 후 배선에 의한 지연을 고려하더라도 2.5V 전원전압에서 50-MHz 클럭으로 안전하게 동작 가능할 것으로 예측되었다. FFT256/2k 코어는 256 모드에서 443 cycles, 512 모드에서 699 cycles,

1024 모드에서 1212 cycles, 2048 모드에서 2,238 cycles 의 latency를 갖는다.

표 5는 FFT256/2k 코어를 Xilinx사의 ISE 툴을 이용하여 합성된 결과를 문헌 [7]~[9]에 발표된 결과들과 비교한 것이다.

FFT256/2k 코어는 최종적으로 FPGA 보드에 구현하여 검증하였으며, 검증시스템은 FPGA 보드, PC 및 PCI 인터페이스 보드, 구동 소프트웨어 등으로 구성된다. 그림 14-(a)는 검증을 위해 제작한 FPGA 보드이다. 사용된 디바이스는 Xilinx XCV2000e bg560-6이며, 외부 I/O 핀으로 데이터를 송수신 할 수 있다. 그림 14-(b)는 제작한 보드와 PC를 이용한 검증시스템이며, 사용한 프로그램은 C 프로그램이다.

V. 결 론

OFDM 기반 지상파 DMB 단말기 모델의 핵심기능 블록으로 사용되는 FFT/IFFT 코어를 설계하였다. 설계된 FFT256/2k 코어는 국내 DMB 표준으로 채택된 Eureka-147 전송규격에 명시된 4가지 전송모드를 지원할 수 있도록 256/512/1024/2048-point FFT/IFFT를 처리하도록 설계하였다. 전력소모를 감소시키기 위해 R2SDF 구조에서 발생하는 불필요한 내부 switching activity를 제거하고, 나비연산 회로를 가산기와 감산기가 결합된 형태로 설계하였다. 또한 메모리 절감형 TS_CBFP 방식 및 hybrid 구조를 적용함으로써 기존의 방식에 비해 메모리 크기를 감소시켰다.

50MHz@2.5-V로 동작하는 경우 2048점 FFT/IFFT 연산에 41- μ s가 소요되어 DMB 시스템의 사양을 만족하는 것으로 평가되었으며, 합성결과 약 68,400 게이트와 58,130 비트의 RAM으로 구현되었다. 본 논문에서 개발된 FFT256/2k 코어는 지상파 DMB 단말기용 핵심 부품 개발에 활용될 수 있을 것이다.

참 고 문 헌

- [1] TTA 저널, "지상파 DMB 기술", 제94호
- [2] 조용수, "무선 멀티미디어 통신을 위한 OFDM 기초", 대영사, 2001.
- [3] "FFT MegaCore Function V1.2.0", <http://www.altera.com/mysupport>, 2004.
- [4] L. Jia, Y. Gao, J. Isoaho and H. Tenhuen, "A New VLSI-Oriented FFT Algorithm And Implementation" *Proceeding of 1998 IEEE ASIC Conference*, pp. 337-341, 1998.

- [5] Y. J. Hongil and J. Kim, "New efficient FFT algorithm and pipeline implementation results for OFDM/DMT applications", *IEEE Trans. on Consumer Electronics*, vol. 49, no. 1, pp. 14-20, Feb., 2003.
- [6] 신경욱, "R4SDF/R4SDC Hybrid 구조를 이용한 메모리 효율적인 2k/8k FFT/IFFT 프로세서 설계", *한국해양정보통신학회 논문지*, 제8권 2호, pp. 430-439, 2004.
- [7] Sacet, "64/256-Point Complex FFT/IFFT", Jan. 2002.
- [8] S. K. Bahl, "A hardware efficient architecture for fast Fourier transform", *In Proc. GSPx and international Signal Processing Conference (ISPC'03)*, Dallas, Texas, Apr. 2003.
- [9] S. Sukhsawas, K. Benkrid, "A high-level Implementation of a High Performance Pipeline FFT on Virtex-E FPGAs", *Proceeding of the IEEE Computer Society Annual Symposium on VLSI Emerging Trends in VLSI Systems Design (ISVLSI'04)*, 2004.

저 자 소 개



이진우(준회원)

2004년 2월 금오공과대학교
전자공학과 졸업
2004년 3월~현재 금오공과
대학교 전자공학과
석사과정 재학 중

<주관심분야 : 마이크로프로세서 설계, SoC 설계, 통신 및 신호처리용 집적회로 설계>



신경욱(정회원)

1984년 2월 한국항공대학교
전자공학과 졸업
1986년 2월 연세대학교 대학원
전자공학과(공학석사)
1990년 8월 연세대학교 대학원
전자공학과(공학박사)

1990년 9월~1991년 6월 한국전자통신연구소
반도체 연구단 선임연구원
1995년 7월~현재 금오공과대학교 전자공학부
교수

1995년 8월~1996년 7월 University of Illinois
at Urbana-Champaign 방문교수

2003년 1월~2004년 1월 University of California
at San Diego 방문교수

<주관심분야 : 통신 및 신호처리용 SoC 설계, 정보보호 SoC 설계, 반도체 IP 설계>