

Recursive 구조를 이용한 MPEG-2 BC/AAC 오디오 공용 합성 필터

정회원 강 명 수*, 박 세 기*, 오 신 범*, 이 채 욱*

A Common Synthesis Filter for MPEG-2 BC/AAC Audio Using Recursive Structure

Myung-Soo Kang*, Se-Ki Park*, Shin-Bum Oh*, Chae-Wook Lee** *Regular Members*

요 약

MPEG 오디오 표준은 고품질 오디오를 제공하기 위해 인간 청각계의 지각현상을 이용한 오디오 압축 알고리즘이다. 다수의 표준안 중 MPEG-2 BC와 MPEG-2 AAC에 대한 공용 복호화 시스템은 아직 발표되어 있지 않다. 본 논문에서는 BC/AAC의 공용 시스템을 구축하여 복호화 과정에서 공통된 구조로 연산을 수행하고 면적감소, 비용절감 등을 목표로 한다.

본 논문에서는 공통된 연산구조를 가지기 위해서 recursive 구조를 이용하여 복호화 과정중 합성 필터링 과정을 공통된 구조로 연산을 수행할 수 있는 공용 합성 필터 구조를 제안한다. 제안한 구조는 연산시간의 개선을 위하여 고속 알고리즘을 적용하였으며 기존의 MPEG-2 AAC 합성 필터링에 대한 recursive 알고리즘을 2단 recursive 구조로 변경, 적용하였고 그 결과 같은 구조로 MPEG-2 BC의 서브밴드 합성필터링도 연산 가능함을 보였다.

Key Words : common synthesis filter, recursive, MPEG-2 BC, MPEG-2 AAC

ABSTRACT

MPEG Audio compression algorithm is the international standard for the digital compression of high quality audio using mechanism of the perceptual coding based on psychoacoustic masking. It is necessary to discuss the constraints on designing of common filter banks for MPEG-2 BC and MPEG-2 AAC decoder system, which is not known yet, mapping audio signals from the time domain into the frequency domain.

In this paper, we present an architecture of common synthesis filter which can be used for MPEG-2 BC and MPEG-2 AAC decoder using recursive structure. The proposed algorithm is based on recursive architecture that effectively performs common computation.

I. 서 론

디지털 필터 뱅크는 신호의 성분을 서로 겹치지 않는 주파수 대역으로 분할하거나 혹은 분할되어 있는 서로 다른 대역의 주파수 성분을 갖는 신호를 합쳐서 새로운 신호를 만들어 내기 위한 디지털 대역 통과 필터의 조합을 말한다^[1]. 따라서 디지털 뱅

크로 신호를 분할하게 되면 서로 다른 주파수 대역에 있는 신호로 분리해 낼 수 있고 반대로 분할된 신호를 합쳐서 다시 원래의 신호를 만들어 낼 수도 있다.

이와 같은 원리를 이용한 대표적인 멀티채널 오디오 압축 알고리즘인 MPEG-2 BC(Backward Compatible), MPEG-2 AAC(Advanced Audio

* 대구대학교 정보통신공학과 디지털신호처리 연구실(cwlee@daegu.ac.kr)

논문번호 : 040057-0207, 접수일자 : 2004년 2월 7일

**본 연구는 산업자원부의 지역혁신인력양상사업의 지원으로 수행되었습니다.

Coding)는 시간-주파수 변환을 위하여 서브밴드 필터뱅크 및 MDCT 필터뱅크와 같이 서로 다른 CMFB(Cosine Modulated Filter Banks)를 사용한다^[2].

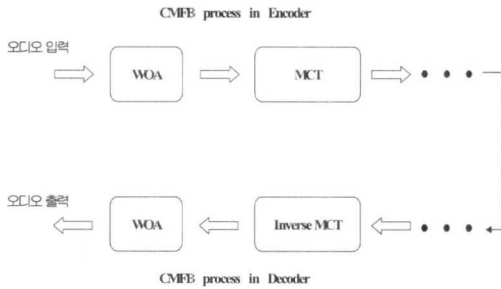


그림 1. 오디오 부호화기와 복호화기의 CMFB

그림 1은 CMFB를 보인 것으로써 WOA (Window and Overlapping Addition)와 MCT (Modulated Cosine Transform)의 두 단계 과정으로 나누어진다. MCT 과정은 덧셈과 곱셈으로 이루어져 있으며, 그 복잡성으로 인하여 FFT(Fast Fourier Transform), FCT(Fast Cosine Transform)와 같은 고속 알고리즘이 널리 사용되고 있으며 고속 MCT의 하드웨어나 소프트웨어를 설계할 경우, 산술 복잡성, 구조의 규칙성, 수치 정확도가 고려되어야 한다.

본 연구에서는 MPEG-2 BC와 MPEG-2 AAC의 복호화 과정 중 합성 필터링 과정의 알고리즘을 분석하여 공통된 구조로 연산을 수행할 수 있는 MPEG-2 BC/AAC 공용 오디오 복호화기를 제안한다. 제안한 구조는 recursive 구조를 이용하였으며 FFT를 사용한 공용 합성 필터구조에 필요한 전처리 및 후처리 과정을 고려해주지 않아도 되고 복소수 연산이 아닌 실수 연산이 되어 단순한 하드웨어 구조를 가지게 하였다^[3]. 제안한 2단 Recursive 구조를 갖는 MPEG-2 BC/AAC 공용오디오 복호화기 성능을 검증하기 위해 기존의 알고리즘을 사용하여 공용오디오 복호화기를 구현하였을 때와 연산시간, 하드웨어의 복잡도를 비교하였으며, 컴퓨터시뮬레이션을 통해 제안한 알고리즘의 기능함을 입증하였다.

II. MPEG-2 BC/AAC 합성 필터링

1. MPEG-2 BC

MPEG에서는 CD수준의 디지털 오디오를 1.5Mbit/s 급의 디지털 저장 매체에 압축, 저장할 수 있는 MPEG-1 표준안을 91년 확정하였다^[4]. 이후 HDTV와 같은 방송매체에 적용할 멀티채널, 음성 다중 등의 응용 목적에 적합한 MPEG-2 표준안도 확정하였다^{[5][6]}.

MPEG 오디오 표준안은 구현의 복잡성 및 비트율에 따라 계층-I, 계층-II, 계층-III으로 나뉘어지며 흔히 MPEG-2 BC는 MPEG-2 계층-II를 일컫는다^[7].

MPEG-2의 각 채널별 부호화 방식은 MPEG-1과 동일하고 MPEG-1에서 규정한 세가지 표본화 주파수(48kHz, 44.1kHz, 32kHz) 이외에도 24kHz, 22.05kHz, 16kHz의 표본화 주파수를 첨가하여 64kbit/s 이하의 낮은 비트율에서도 좋은 음질을 얻을 수 있도록 하였다. MPEG-2 BC 오디오 복호화 과정은 다음 그림과 같다.

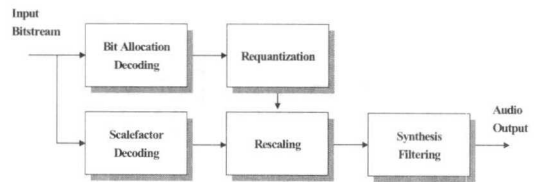


그림 2. MPEG-2 BC 오디오 복호화 과정

그림 2에서 MPEG-2 BC 오디오 복호화 과정은 프레임 동기를 맞추고 여러 가지 헤더 정보를 추출하는 과정인 비트정보 추출과정, 크기인자 정보 추출과정, 역 양자화 과정, 정규화 과정 그리고 다채널 처리과정과 합성 필터링 과정으로 나누어진다.

MPEG-2 BC 오디오 복호화 과정 중 합성 필터링 과정은 매우 많은 연산량을 차지하는 부분으로 크게 나누어서 샘플 벡터 형성 과정, 매트릭싱 과정, 간축을 통한 샘플 벡터 형성과정, 윈도우를 취하는 과정, 32개의 PCM 샘플을 얻어내는 과정으로 이루어져 있다. 먼저 새로운 32개의 서브밴드 샘플로부터 매트릭싱 과정을 통해서 64개의 매트릭싱된 샘플을 얻은후 1024 버퍼에 넣고 간축을 통해 512 크기의 샘플벡터를 형성한 후, 합성 윈도우를 취해서 32개마다 하나의 샘플씩을 간축하여 총 16개의 샘플을 합산하면 1개의 PCM 샘플을 복원하게 된다. 그리고 합성 필터링 과정의 매트릭싱 과정은 곱셈과 덧셈이 주된 연산을 이루고 있으며 이를 반복 실행하는 과정으로써 다음과 같다.

$$V(i) = \sum_{k=0}^{31} S(k) \cos \left[\frac{\pi}{64} (2k+1)(i+16) \right] \quad (1)$$

for $i = 0, 1, \dots, 63$

여기서 $S(k)$ 는 서브밴드 샘플이다.

논문 [8]에서 제안한 알고리즘에 따르면 식(1)은 코사인함수의 대칭성을 이용해서 64개의 $V(i)$ 대신에 다음과 같이 32개의 $V'(i)$ 값만으로 연산이 가능하다.

$$V'(i) = \sum_{k=0}^{31} S(k) \cos \left[\frac{\pi}{64} (2k+1)(i) \right]$$

for $i = 0, 1, \dots, 31$

(2)

DCT 연산을

$$Y(k) = \sum_{i=0}^{N-1} y(i) \cos \left[\frac{\pi}{2N} (2i+1)k \right]$$

for $k = 0, 1, \dots, N-1$

(3)

로 나타내었을 때 recursive 커널은 다음과 같다 [9][10]

$$P_m \sin(x) = \sum_{i=m}^{N-1} y(i) \sin[(i-m+1)x] \quad (4)$$

식 (3)에서 N 은 32이고 $Y(k)$, $y(i)$ 는 식 (2)의 $V'(i)$, $S(k)$ 에 해당된다.

식 (4)를 전개하여 정리하면 P_m 은 다음과 같다.

$$P_m = y(m) + 2\cos(x)P_{m+1} - P_{m+2}$$

$m = N-1, \dots, 1, 0$

(5)

그리고 식 (3)에 $\sin(x)$ 을 양변에 곱한 후 정리하면 다음과 같이 된다.

$$Y(k) = y(0) \cos\left(\frac{\pi k}{2N}\right) + \cos\left(\frac{3\pi k}{2N}\right)P_1 - \cos\left(\frac{\pi k}{2N}\right)P_2 \quad (6)$$

그리고 식 (5)와 (6)의 관계를 이용하게 되면 다음과 같다.

$$Y(k) = P_0 \cos\left(\frac{\pi k}{2N}\right) - P_1 \cos\left(\frac{\pi k}{2N}\right) \quad (7)$$

그림 3과 4는 식 (6)과 (7)의 recursive 필터 구조를 나타내었다.

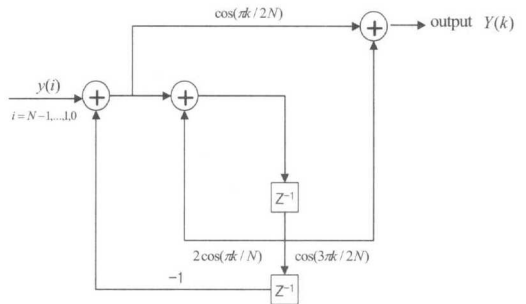


그림 3. 식 (6)에 의한 Chau의 recursive 필터 구조

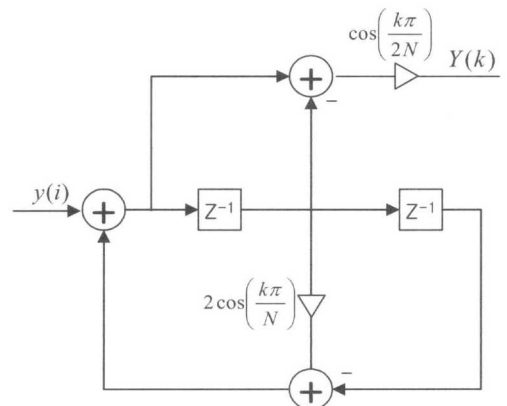


그림 4. 식 (7)에 의한 Chau의 recursive 알고리즘 구조

1. MPEG-2 AAC

MPEG에서는 MPEG-1 스테레오 포맷과의 호환이 필요없는 응용 분야로의 활용을 위한 새로운 표

준화 작업으로 멀티채널 오디오 부호화 방식의 MPEG-2 AAC 국제 표준을 발표하였다[11]. MPEG-2 AAC는 매우 낮은 비트율에서 방송 음질 수준의 오디오를 제공하기 위해서 고해상도 필터뱅크(High-resolution Filter Bank), 예측기법(Prediction), 허프만 부호화(Huffman Coding) 등을 결합하여 사용한다. 또한 MPEG-2 AAC는 국제 표준안으로서 완성된 MPEG-4 오디오의 한 부분으로 채택되었다[7].

MPEG-2 AAC 복호화 과정은 다음 그림과 같다.

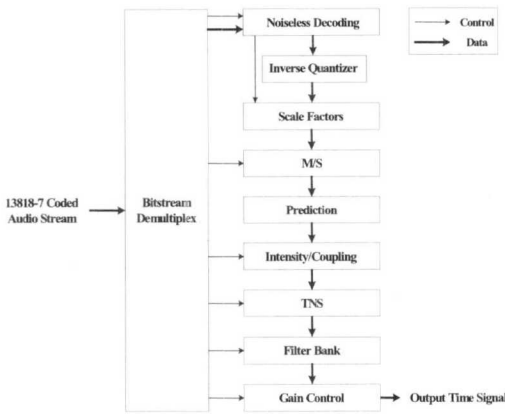


그림 5. MPEG-2 AAC 오디오 복호화 과정

MPEG-2 AAC 시스템은 음질, 메모리, 전력 요구량 사이의 손익을 고려해서 세 가지 profile을 지원하고 각각의 profile마다 사용되는 틀은 다르다. 그림 5에서 필터뱅크는 세 가지 profile에서 모두 사용하며 시간영역의 신호로 변환하는 역할을 수행한다. 그리고 시간영역 에일리어싱 제거(Time-Domain Aliasing Cancellation; TDAC)라 불리는 기법이 적용된다.

MPEG-2 AAC 복호화기에서 사용하는 IMDCT는 다음과 같다.

$$x(n) = \sum_{k=0}^{M-1} X(k) \cos \left[\cos \frac{\pi}{M} \left(k + \frac{1}{2} \right) \left(n + \frac{M+1}{2} \right) \right]$$

$$n = 0, 1, \dots, N-1$$

(8)

여기서

- n = 샘플 인덱스
- k = 스펙트럼 계수 인덱스
- N = 윈도우 길이
- M = $N/2$

이다.

논문 [12]에서 제안한 IMDCT recursive 필터 구조를 그림 6에 나타내었다.

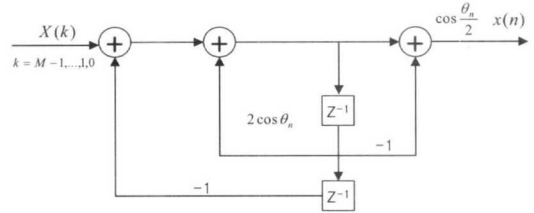


그림 6. Nikolajevic의 IMDCT recursive 필터 구조

여기서 $\theta_n = \left(n + \frac{M+1}{2} \right) \frac{\pi}{M}$ 이다.

III. 제안하는 공용 합성 필터 구조

앞에서 살펴본 MPEG-2 BC와 MPEG-2 AAC의 recursive 알고리즘은 입력 값 하나에 대하여 입력 개수만큼 순환을 계속하여 곱셈과 덧셈을 한 후 값이 출력되게 된다. 그림 4와 그림 6에서 MPEG-2 BC와 MPEG-2 AAC의 recursive 알고리즘의 구조는 지연소자 2개, 덧셈소자 3개, 곱셈소자 2개로 동일함을 알 수 있다. 또한, 동일한 데이터 경로를 가진다. 그러나 곱해지는 코사인 계수값은 MPEG-2 BC와 MPEG-2 AAC에 따라 다르다. 그리고 recursive 공용 합성 필터 구조를 설계할 때 MPEG-2 AAC 같은 경우 많은 연산량을 가지기 때문에 공용 합성 필터 구조를 실제로 하드웨어로 구현하기 위해서는 연산량 및 계산의 복잡도가 문제가 된다.

한편 합성 필터링 과정에서 MCT연산은 그림 4와 그림 6처럼 입력은 역순으로 들어가고 출력은 순차적으로 나오게 되는 구조를 가지게 되며 WOA과정이 필요하다. 즉 MPEG-2 BC와 MPEG-2 AAC에 따라 적절한 윈도우를 취한 후에 MPEG-2 BC는 중첩 가산 방법을 이용하여 32개씩 36번 반복하여

1152개의 PCM sample을 출력하고, MPEG-2 AAC 도 중첩 가산 방법을 이용하여 long window의 경우는 1024개, short window의 경우는 128개씩 8번 반복하여 1024개의 PCM sample을 출력한다. 그림 6에서 $x(n)(n = 0, 1, \dots, N - 1)$ 값은 코사인 계수의 대칭성으로 인하여 절대치가 동일한 값을 중복 계산하는 구조를 가지고 있다. 따라서 불필요한 연산이 소모됨으로 많은 연산시간의 문제점이 있다.

따라서, 본 논문에서는 이러한 점을 고려하여 다음과 같이

$$x(M - 1 - n) = -x(n) \tag{9}$$

$$n = 0, 1, \dots, \frac{N}{4} - 1$$

$$x(N - 1 - n) = x(M + n)$$

$$n = 0, 1, \dots, \frac{N}{4} - 1 \tag{10}$$

설정하여 $\frac{N}{4}$ 길이의 $x(n)$ 및 $x(M+n)$ 을 각각 구하면 식 (9), (10)에 의해 $x(n)$ 을 구할 수 있다. 따라서 IMDCT의 recursive 구조를 그림 7과 같이 2단으로 구성하면 병렬로 연산되는 구조가 되어 연산시간을 대폭 줄일 수 있다. 2단으로 구성하였을 경우 지연소자 4개, 덧셈소자 6개, 곱셈소자 4개가 필요하다.

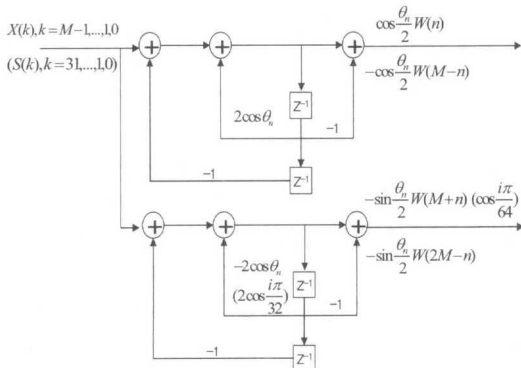


그림 7. 제안하는 MPEG-2 BC/AAC 합성필터 구조

그림 7에서 윗부분이 $x(n)(n = 0, 1, \dots, \frac{N}{4} - 1)$ 의 연산부이고, 아랫부분이

$x(M+n)(n = 0, 1, \dots, \frac{N}{4} - 1)$ 에 대한 연산부이다. 그리고 $W(\cdot)$ 은 윈도우를 취하기 위한 계수이다. 코사인 계수와 윈도우 계수를 미리 곱한 값을 메모리에 저장해두고 사용하도록 한다. 아랫부분의 $x(M+n)$ 에 필요한 코사인 계수 값인 $2\cos\theta_{M+n}$ 와 $\cos\frac{\theta_{M+n}}{2}$ 는 다음과 같이 나타낼 수 있다.

$$2\cos\theta_{M+n} = 2\cos\left[\left(M+n + \frac{M+1}{2}\right)\frac{\pi}{M}\right]$$

$$= 2\cos(\pi)\cos\left[\left(n + \frac{M+1}{2}\right)\frac{\pi}{M}\right]$$

$$= -2\cos\theta_n \tag{11}$$

$$\cos\frac{\theta_{M+n}}{2} = \cos\left[\left(M+n + \frac{M+1}{2}\right)\frac{\pi}{2M}\right]$$

$$= -\sin\left(\frac{\pi}{2}\right)\sin\left[\left(n + \frac{M+1}{2}\right)\frac{\pi}{2M}\right] \tag{12}$$

$$= -\sin\left(\frac{\theta_n}{2}\right)$$

그림 7에서 $x(M+n)$ 에 필요한 코사인 계수값은 식 (11)처럼 같은 클럭에서 $x(n)$ 의 연산시 필요한 코사인 계수값의 부호만 반전시켜주면 된다. 그 결과로 메모리량을 줄일 수 있다. 그리고 Overlap-Add 과정을 위하여 아랫부분의 계산 결과값은 1K 워드 메모리에 저장하도록 하고 윗부분의 계산값과 1K 워드 메모리에 저장된 값을 더해 Overlap-Add 과정을 끝낸후 PCM sample 값을 출력한다.

그리고 MPEG-2 BC의 recursive 과정을 그림 7에 적용할 경우 아랫부분에서 $V(i)$ ($i = 0, 1, \dots, 31$)를 연산하도록 한다. 그리고 연산한 결과인 $V(i)$ 데이터를 적절하게 배치하면 원하는 결과인 $V(i)(i = 0, 1, \dots, 63)$ 를 구할

수 있다. 이때 곱셈에 필요한 코사인 계수값인 $2\cos\left(\frac{i\pi}{32}\right)$ 및 $\cos\left(\frac{i\pi}{64}\right)$ 은 메모리에서 읽어오도록 한다.

Window Overlap-Add 과정은 MPEG-2 AAC와는 다르다. 하지만 MPEG-2 AAC의 WOA 과정에 필요한 1K 워드 메모리내에서 연산이 가능하고 $\cos\left(\frac{i\pi}{64}\right)$ 와 합성 윈도우 계수값을 미리 곱한 결과를 사용하여 연산을 하면 곱셈 횟수를 줄일 수 있다. 그리고 Overlap-Add 연산을 거친 후 32개의 오디오 샘플값을 출력한다.

따라서, 그림 7의 구조는 MPEG-2 BC/AAC의 합성 필터링 과정에서 공통적으로 recursive 구조를 가지게 되어 FFT를 사용한 방법처럼 전처리 및 후처리 과정을 고려하지 않는 실수연산을 함으로써 효율적인 공용 합성 필터 구조의 구현이 가능해진다. 다만, MPEG-2 BC와 AAC의 WOA 과정은 서로 상이함으로 별개의 연산 구조를 가지게 된다.

IV. 실험결과

공용 합성 필터 구조의 검증은 위하여 Matlab을 사용하여 컴퓨터 시뮬레이션을 하였다. 시뮬레이션 블록도는 다음 그림과 같다.



그림 8. 제안 구조 검증 시뮬레이션 블록도

샘플데이터는 16비트, 44.1kHz로 샘플링 된 타악기 cowbell 및 8비트, 22.05kHz로 샘플링된 타악기 conga 이며 기준신호와 제안한 구조의 출력신호를 원본데이터와 비교하였다. 제안하는 구조의 출력신호는 식 (2)와 식 (9), 식 (10)의 고속 알고리즘을 이용하고 그림 7의 recursive 구조로 matlab 코드화한 것이며 기준신호는 표준문서에서 제공해준 합성 필터블록으로 실행한 것으로 AAC는 1024 샘플,

BC는 1152 샘플 결과값을 비교하였다.

그림 9과 그림 10은 MPEG-2 AAC의 분석필터를 거친 신호를 이용, 기준신호와 출력신호를 구한 것으로 출력신호는 기준신호와 샘플 데이터와 비교하였을때 동일한 결과가 나옴을 확인하였고 그림 11과 그림 12는 MPEG-2 BC의 분석필터를 거친 신호를 이용, 기준신호와 출력신호를 구한 것으로 약간의 지연을 제외하고는 동일한 결과가 나옴을 확인하였다. 따라서, 공통적인 연산을 수행할 수 있는 구조인 recursive 공용 합성 필터 구조는 구현 가능하다.

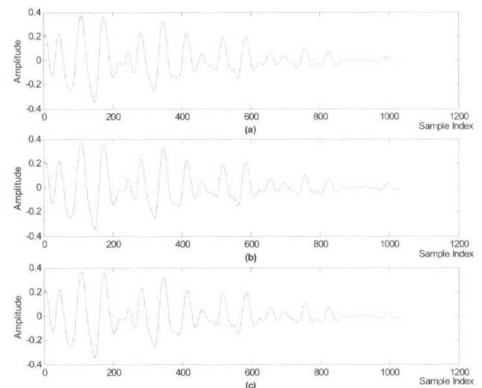


그림 9. AAC의 cowbell 샘플데이터에 대한 실행결과 (a) 샘플신호 (b) 기준신호 (c) 출력신호

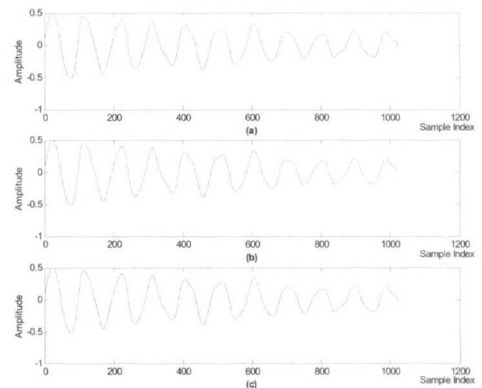


그림 10. AAC의 conga 샘플데이터에 대한 실행 결과 (a) 샘플신호 (b) 기준신호 (c) 출력신호

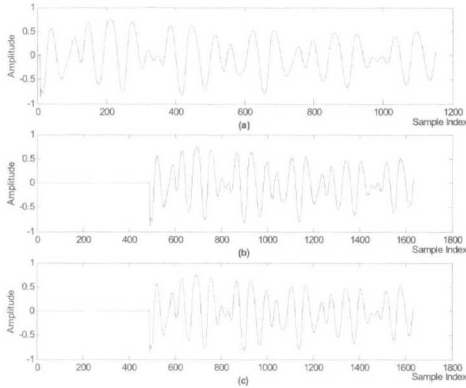


그림 11. BC의 cowbell 샘플데이터에 대한 실행 결과
(a) 샘플신호 (b) 기준신호 (c) 출력신호

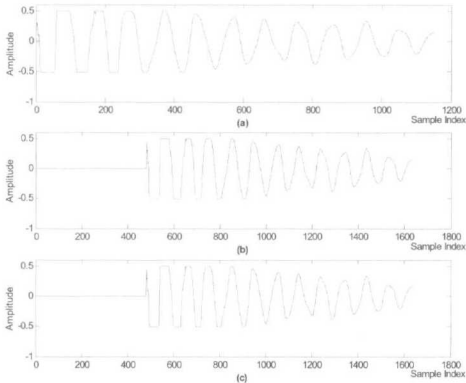


그림 12. BC의 conga 샘플데이터에 대한 실행 결과
(a) 샘플신호 (b) 기준신호 (c) 출력신호

다음은 제안한 2단 Recursive 구조를 갖는 MPEG-2 BC/AAC 오디오 공용복호화기와 기존의 알고리즘을 연산시간, 하드웨어의 복잡도를 비교하였다.

표 1에서 제안한 2단 Recursive 구조를 갖는 MPEG-2 BC/AAC 오디오 공용복호화기 계수를 저장하기 위해 필요로 하는 메모리량을 나타내었다. 표 2와 표3에서 MPEG-2 AAC와 MPEG-2 BC에 대한 각 알고리즘의 Recursive 모듈을 하드웨어로 설계할 때 필요한 하드웨어와 제안한 알고리즘에 대한 하드웨어 복잡도를 비교하였다. 먼저 MPEG-2 AAC 연산시간을 보면 Modified Regressive 구조와 제안하는 2단 Recursive 구조는 Chiang의 Regressive 구조와 Nikolajevic의 Recursive 구조의

1/4의 연산시간만을 필요로 한다. 하지만 하드웨어의 복잡도에서는 Chiang의 Regressive 구조와 Nikolajevic의 Recursive 구조에 비해 Modified Regressive 구조는 4배정도 증가하였으며, 2단 Recursive 구조는 2배가 증가하였다. MPEG-2 BC에서 대해서는 표3에 보이는 것처럼 각 알고리즘의 Recursive 연산시간과 하드웨어 복잡도에 대해서 기존 알고리즘과 제안한 알고리즘이 서로 같음을 알 수 있다.

따라서 연산시간과 하드웨어 복잡도, 메모리를 모두 고려하였을 경우, 제안하는 2단 Recursive 구조를 갖는 MPEG-2 BC/AAC 공용오디오 복호화기는 기존 알고리즘을 공용으로 했을 때 보다 연산시간을 75% 단축시킬 수 있다.

표 1. 제안 구조의 계수 메모리량

계 수 / 구 분	AAC		BC	Total
	N=256	N=2048		
$2 \cos \theta_n$ $(2 \cos \frac{i\pi}{32})$	64	512	32	608
$\cos \frac{\theta_n}{2} W(n)$ $-\sin \frac{\theta_n}{2} W(n + \frac{i\pi}{64})$	128*2	1024*2	64	2368
Total	320	2560	96	2976

표 2. 하드웨어의 복잡도 비교(AAC)

구 분 / 류	IMDCT			
	Mpy	Add	Z ⁻¹	clock cycles
Chiang의 Regressive	1	3	3	1024*2048
Nikolajevic의 Recursive	1	3	2	1024*2048
Modified Regressive	4	10	8	512*1024
2단 Recursive	2	6	4	1024*512

표 3. 하드웨어의 복잡도 비교(BC)

구분 \ 분류	BC			
	Mpy	Add	Z ⁻¹	clock cycles
chau의 recursive al-algorithm	1	3	2	32*32
chau의 recursive al-algorithm	1	3	2	32*32
제안구조	1	3	2	32*32

V. 결론

본 논문에서는 MPEG-2 BC/AAC 오디오 복호화에 공용으로 사용될 수 있는 공용 합성필터의 구조를 제안하였다. MPEG-2 BC와 MPEG-2 AAC의 합성 필터링 과정은 서로 다른 MCT 형태를 가지고 있으나 공통적인 recursive 구조로 유도하여 같은 구조에서 연산이 가능하도록 하였다. 제안한 구조는 서브밴드 필터링 과정과 TDAC에 바탕을 둔 IMDCT의 공통된 연산구조를 유도하기 위하여 recursive 알고리즘을 사용하였고, Nikolajevic의 MPEG-2 AAC recursive 구조를 변형한 2단 recursive 구조를 사용하였다. 그리고 여기에 MPEG-2 BC의 서브밴드 필터링처리를 할 수 있는 구조가 되도록 하여 MPEG-2 BC와 MPEG-2 AAC 합성 필터링 과정을 모두 효과적으로 처리할 수 있도록 하였다.

제안한 2단 Recursive 구조를 갖는 MPEG-2 BC/AAC 공용오디오 복호화기와 기존 알고리즘을 이용한 공용오디오 복호화기에 대하여 연산시간, 하드웨어 복잡도를 비교하였으며, 그 결과 본 논문에서는 제안한 2단 Recursive 구조를 갖는 MPEG-2 BC/AAC 공용오디오 복호화기는 기존 알고리즘을 이용한 공용오디오 복호화기에 비해 연산시간을 75% 단축시킬 수 있었다.

제안한 알고리즘은 다양한 Window Switching을 포함한 가변길이의 IMDCT 필터뱅크에 적합하며, 두 가지 이상의 알고리즘을 복호화할 수 있는 공용 복호화기의 합성필터에도 적용가능 할 것이다.

앞으로의 과제는 고정소수점에서의 오차에 대한 값을 최소화하는 것과 제안된 알고리즘을 실제 복호화기에 적용하여 설계하는 연구가 필요하다.

참고 문헌

- [1] Bose. Tamal, "Digital Signal and Image Processing", John Wiley & Sons Inc, pp. 337-358, 2003.
- [2] Ch.M. Liu, W.Ch. LEE, "A unified fast algorithm for cosine modulated filter banks in current audio coding standards", J. Audio Eng. Soc. 47(12), pp. 1061-1075, 1999
- [3] 고우석, 유선국, 박성욱, 정남훈, 김준석, 이근섭, 윤대회, "AC-3와 MPEG-2 오디오 디코더 공용 합성필터의 설계", 신호처리 합동 학술대회, pp. 299-302, 1997.
- [4] ISO/IEC JTC1/SC29/WG11 No.71 "Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s - CD 11172-3(Part3. MPEG-Audio)".
- [5] ISO/IEC JTC1/SC29/WG11 No.703 "Generic Coding of Moving Pictures and Associated Audio - CD 13813-3 (Part3. MPEG-Audio)", Mar. 1994.
- [6] ISO/IEC JTC1/SC29/WG11 No.1519 "Generic Coding of Moving Pictures and Audio - CD 13813-3(Part3. MPEG-Audio)", 2nd Edition, Feb., 1997.
- [7] M. Bosi, R. Goldberg, "Introduction to Digital Audio Coding and Standards", Kluwer Academic Publishers, pp. 256-369, 2003.
- [8] K. Konstantinides, "Fast Subband Filtering in MPEG Audio Coding", IEEE Signal Processing, Letters vol.1, no.2, pp. 26-28, 1994.
- [9] L.-P. Chau and W.-C. Siu, "Recursive algorithm for the discrete cosine transform with general lengths", ELECTRONICS LETTERS, VOL. 30, NO. 3, Feb 1994.
- [10] Lap-Pui CHAU, Wan-Chi SIU, "Recursive Algorithm for the realization of the Discrete Cosine Transform", ISCAS 2000 - IEEE International Symposium on Circuits and Systems, May 28-31, 2000.
- [11] ISO/IEC JTC1/SC29/WG11 No.1650 "IS

13818-7 (MPEG-2 Advanced Audio Coding, AAC)", Apr., 1997.

[12] Vladimir Nikolajevic, "Computation of Forward and Inverse MDCT Using Clenshaw's Recurrence Formula", *IEEE Trans, SIGNAL PROCESSING, VOL. 51, NO.5*, May 2003.

이 채 욱(Chae-wook Lee)

정회원



1980년 2월 : 한국항공대학교
통신공학과 졸업

1987년 3월 : 동경공업대학
전기전자공학과 석사

1990년 3월 : 동경공업대학
전기전자공학과 박사

1990년 3월~현재 :

대구대학교 정보통신공학과 교수

<관심분야> 신호처리, 광통신시스템

강 명 수(Myung-Soo Kang)

정회원



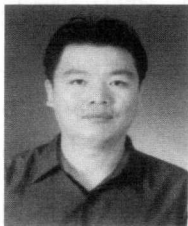
2002년 2월 : 대구대학교
정보 통신공학과 졸업

2004년 2월 : 대구대학교
정보 통신공학과 석사

<관심분야> 디지털신호처리, 적응신호처리

박 세 기(Se-Ki Park)

정회원



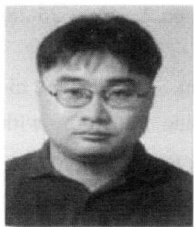
2002년 2월 : 대구대학교
정보 통신공학과 졸업

2004년 2월 : 대구대학교
정보 통신공학과 석사

<관심분야> 디지털신호처리, 적응신호처리

오 신 범(Shin-bum Oh)

정회원



1997년 2월 : 대구대학교
정보 통신공학과 졸업

1999년 8월 : 대구대학교
정보 통신공학과 석사

2004년 2월 : 대구대학교
정보통신공학과 박사

<관심분야> 디지털신호처리, 적응신호처리