

MPEG-2 AAC 복호화 시스템의 구조 제안 및 구현

방경호, 김준석*, 윤대희

연세대학교 전기전자공학과, ㈜인타임*

Design Optimization of MPEG-2 AAC Decoder

Kyoung Ho Bang, Joon Seok Kim* and Dae Hee Youn

Dept. of Electrical and Electronic Eng. Yonsei University, InTime Corp.*

E-mail : euphony@cyclon.yonsei.ac.kr

요약

본 논문에서는 2 채널 MAIN 프로파일 MPEG-2 AAC 복호화 시스템의 구조를 제안하고 구현하였다. 복호화 알고리즘의 구조적인 모듈화에 근거하여, 시스템 설계 과정에서 전체 시스템을 3 개의 하드웨어 모듈로 분할하였다. 전체 시스템은 허프만 복호화기, 예측기, 20 비트 고정소수점 DSP 코어로 이루어져 있다. 허프만 복호화기는 주어진 작업을 1 클럭 사이클 내에 수행할 수 있는 고속의 하드와이어드 모듈이고, 예측기는 높은 해상도를 가지고 다른 모듈들과 병렬 처리가 가능한 구조를 가진 모듈이다. 구현된 시스템은 16.9 MIPS 로 2 채널의 MPEG-2 AAC 비트열을 고음질로 복호화할 수 있다.

1. 서론

AAC 시스템은 음질, 메모리, 전력 요구량 사이의 손익에 따라, MAIN 프로파일, LC 프로파일, SSR 프로파일의 세가지 프로파일을 제공한다 [1]. 그 중에서 MAIN 프로파일은 가장 뛰어난 음질을 제공한다. 그러나 이득 제어 틀을 제외한 모든 틀을 사용하기 때문에 가장 복잡도가 높다 [2]. 그러므로 MAIN 프로파일 AAC 복호화기를 구현함에 있어서 막대한 연산량과 메모리 요구량을 해결해야 한다. 이러한 문제점들은 AAC 복호화 알고리즘을 구성하는 틀 중에서, 주로 허프만 복호화, 예측, 고해상도 필터뱅크 [3]에서 발생한다. 이 세 가지 틀들의 특징을 요약하면 다음과 같다.

• 허프만 복호화 : 허프만 코드는 크기인자와 양자화된 스펙트럼 데이터의 각각 중복된 부분을 제거하기 위해서 가변 길이의 코드워드를 사용한다. AAC 에서는 하나의 크기인자 코드북과 11 개의 스펙트럼 허프만 코드북이 사용된다. 그러나 이러한 허프만 코드북을 저장하기 위해서는 4k word 가량의 많은 양의 메모리가 요구된다. 또한 코드북을 탐색하는데 많은 양의 연산량이 요구된다.

• 예측 : 예측은 신호의 중복성을 제거하는 데 사용되고, 특히 정적인 신호에 효과적인 틀이다. 2 차의

후방 적응 예측기는 이전 프레임의 양자화된 스펙트럼에 근거하여 주어진 서브밴드에서 서브밴드 샘플을 예측하여 비트율을 감소시킨다. 그러나 이전 프레임의 상태변수 값들을 저장하기 위해 많은 메모리 공간이 필요하다.

• 고해상도 필터뱅크 : MP3 의 하이브리드 필터뱅크와는 달리, AAC 는 MDCT 를 사용한다. 윈도우 길이의 증가, 채택된 윈도우 함수, 변환 블록 전환 등과 함께 MDCT 는 이전의 부호화 방식에 비해 성능이 향상되고, 필터뱅크의 주파수 선택도가 향상되었다. 그러나 MDCT 는 연산량 부담이 증가되었다.

본 논문에서는 MAIN 프로파일을 지원하는 2 채널 MPEG-2 AAC 의 최적화된 시스템 구조를 제안한다. 위에서 열거한 문제점들을 해결하기 위해서 효율적인 시스템 구조를 개발하였다.

제안된 시스템은 허프만 복호화 모듈, 예측 모듈, 프로그래머블 DSP 코어 모듈의 세 부분으로 구성되어 있다. 세 개의 하드웨어 모듈로 병렬 처리 구조를 형성하여 시스템의 효율을 최대화하였다. 또한 고해상도 필터뱅크의 연산량 부담을 덜기 위해서, Duhamel 등이 제안한 고속 IMDCT 알고리즘 [4]을 어셈블리 언어로 구현하였다.

2. 설계 최적화

2-1. 허프만 복호화 과정

AAC 복호화기에서는 허프만 코드북 선택 정보에 의해 12 개의 허프만 코드북 중의 하나가 선택된다. 허프만 코드워드는 선택된 허프만 코드북을 참조하여 비트열로부터 추출된다. 양자화된 스펙트럼 계수는 추출된 허프만 코드워드의 허프만 코드 인덱스로부터 구해진다.

허프만 복호화 과정은 12 개의 허프만 코드북을 포함하기 때문에, 이 과정에서 많은 양의 메모리가 필요하다. 허프만 코드워드를 추출하는 것은 DSP 코어의 제어를 복잡하게 만드는 다양한 연산을

요구한다.

본 논문에서는 고속의 하드웨어로 모직을 이용하여 허프만 복호화기를 설계하였다. 설계된 허프만 복호화기는 1 클럭 사이클 내에 하나의 허프만 코드 인덱스를 복호화할 수 있다. 표 1 에서 허프만 복호화기의 하드웨어 구현 방식과 소프트웨어 구현 방식을 비교하였다.

표에서 보듯이, 하드웨어 구현 방식이 소프트웨어 구현 방식에 비해 속도와 메모리 요구량 측면에서 유리하다는 사실을 알 수 있다.

표 1 허프만 복호화기의 구현 방식 비교

| | H/W 구현 방식 | S/W 구현 방식 |
|-----|--|----------------------|
| 방법 | Combinational logic (logic gate & MUX) | 코드북 탐색 (최대 289 번 탐색) |
| 속도 | 1 클럭 사이클 | 경우에 따라 가변 |
| 메모리 | 불필요 | 4098 word |

2-2. 예측 과정

AAC 는 채널 내에서 프레임 사이의 상관도를 제거하여 부호화 효율을 높이는 것으로 정적인 특성이 강한 신호의 경우에 특히 효과적이다. 예측 과정은 각 채널의 주파수 스펙트럼 성분들에 대해 과거 블록과의 상관성을 제거하는 방법으로 적용된다.

예측 과정은 각 스펙트럼 성분들에 대해 하나씩의 2 차 후방 적응 예측기를 가지고 있어서, 연속되는 이전 프레임들의 스펙트럼으로부터 상관 관계를 제거시킨다. 16kHz 의 주파수 대역까지의 각 스펙트럼 성분에 대해서 예측기를 하나씩 사용하고, LMS 알고리즘을 각 예측기에서 사용한다. 그래서, 예측 과정은 부동 소수점 곱셈과 덧셈의 결과가 누적되고, 각 예측기의 상태 변수들을 저장하기 위한 많은 양의 저장 공간을 필요로 한다.

본 논문에서는 부동 소수점 연산기와 메모리 모듈을 사용하여 예측기를 하드웨어로 로직으로 설계하였다. 예측 과정과 그 외의 과정은 서로 독립적으로 처리되므로, 복호화 시스템의 처리 속도를 향상시키기 위해서 예측기와 다른 모듈들을 병렬 처리할 수 있는 구조로 설계하였다.

2-3. 고속 T/F 변환

T/F 변환 과정은 부호화기에서 시간 영역의 신호를 내부적인 시간-주파수 표현으로 변환한 신호를 복호화기에서 이의 역변환을 수행하는 것이다. AAC 에서는 이 변환 과정을 IMDCT 를 이용한다. IMDCT 는 시간영역에서의 TDAC 기법이 적용된다.

본 논문에서는 Duhamel 등이 제안한 고속 IMDCT 알고리즘을 이용하여 필터 बैं크 과정을 구현하였다. 이 알고리즘은 N-point FFT 에 대해서 N/4-point Complex FFT 를 적용하여, 전체 시스템의

연산량을 1/10 로 감소시킬 수 있었다.

2-4. 제안된 시스템 구조

제안된 AAC 복호화 시스템의 전체 구조는 그림 1 과 같다. 전체 시스템은 전용 프로세서 코어와 두 개의 하드웨어로 로직 모듈로 구성되어 있다.

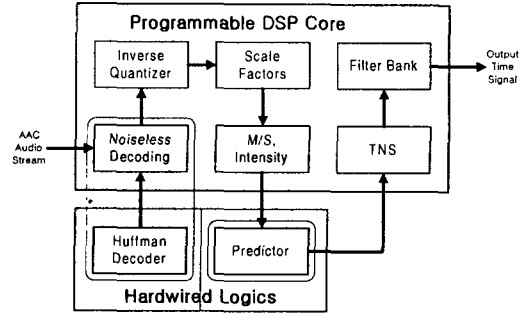


그림 1 제안된 AAC 복호화 시스템의 전체 구조

하드웨어로 로직 모듈은 허프만 복호화 모듈과 예측 모듈로 구성되어 있다. 허프만 복호화 모듈의 방대한 연산량으로 인해 허프만 복호화기의 구조는 빠른 연산을 수행하는 데 초점을 맞추어야 한다. 예측기는 연산의 정확성, 많은 연산량과 메모리를 요구하므로, 부동소수점 연산과 외부 메모리를 갖는 구조를 선택하는 것이 바람직하다.

프로세서 코어는 오디오 신호처리에 최적화된 20 비트 고정소수점 프로그래머블 DSP 코어이다 [5]. DSP 코어는 무손실 복호화 과정, 역양자화 과정, 크기인자 과정, M/S 스테레오 과정, intensity 스테레오 과정과 필터 बैं크 과정을 수행한다.

3. 하드웨어 모듈의 설계

3-1. 허프만 복호화기

허프만 복호화 모듈은 프로세서 코어의 연산량을 덜기 위해 하드웨어로 로직을 사용하여 설계하였다. 허프만 복호화 모듈의 구조는 그림 2 와 같다.

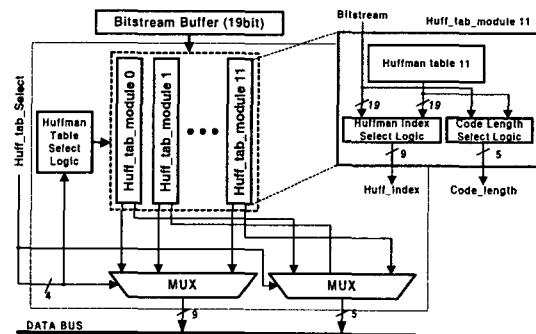


그림 2 허프만 복호화기의 구조

이 모듈은 12 개의 허프만 테이블 모듈과 이 모듈 중에 하나를 선택하는 허프만 테이블 선택 모듈로 구성되어 있다. 허프만 인덱스 선택 로직은 각 허프만 테이블과 비트열을 비교하여 비트열에 해당하는 허프만 코드 인덱스를 찾아낸다. 이러한 구조를 사용하여 허프만 복호화 과정은 1 클럭 사이클 내에 이루어진다.

3-2. 예측기

예측 과정은 각각의 스펙트럼 계수를 예측하기 위하여 2 차 후방 격자구조의 필터를 사용한다. 예측 과정에서는 고휘상도의 연산 결과를 필요로 한다. 따라서 본 논문에서는 부동소수점 연산기를 설계하여 사용하였다. 16 비트 데이터 형식을 사용하여 시스템의 전력 소모도 감소시킬 수 있었다. DAG 모듈은 예측기의 상태 변수들이 저장될 메모리의 주소를 발생시키고, 예측기 제어기는 연산기와 DAG 모듈을 제어하는 역할을 한다. 예측기의 구조는 그림 3 과 같다.

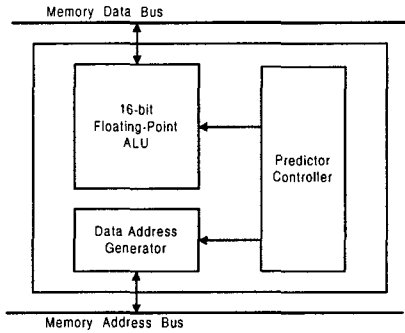


그림 3 예측기의 구조

3-3. 프로세서 코어

프로세서 코어 모듈은 오디오 신호처리에 적합한 구조를 가진 전용 프로세서이다. 성능 향상을 위해서 3 단 파이프라인을 사용하였고, 효과적인 파이프라인을 구현하기 위해서 Harvard 구조를 채택하였다. 이 프로세서 코어는 최소 40MHz 의 클럭 속도를 지원한다. DSP 코어 모듈은 명령어 페치 유닛, 명령어 디코더 유닛, 실행 유닛으로 구성되어 있다. 실행 유닛은 데이터 처리 유닛과 데이터 주소 발생기로 구성되어 있다. DSP 코어의 데이터 처리 유닛은 단일 MAC 구조를 기반으로 설계하였다. 데이터 처리 유닛의 구조는 그림 4 와 같다.

프로세서 코어는 하드웨어 MAC 을 포함한 산술/논리 연산 뿐만 아니라, UNPACK, HUFFMAN 과 같은 특수 명령어를 지원한다. 특히 UNPACK 은 bit 단위로, 정보를 추출하는 데에 유용하게 사용되는 명령어이다. 모든 명령어는 1 사이클 내에 수행된다.

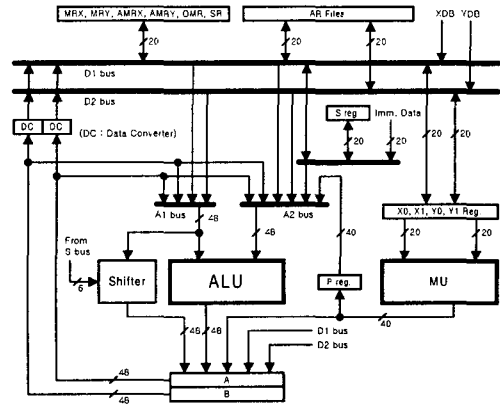


그림 4 데이터 처리 유닛의 구조

4. 시스템의 효율성

구현된 시스템을 검증하기 위해서 두 단계의 검증 과정을 거쳤다. 먼저 설계된 시스템의 알고리즘 타당성을 검증하는 것이고, 다음은 구현된 시스템의 실시간 동작 가능성을 검증하는 것이다. 두 검증 과정은 C-언어 기반으로 제작한 DSP 코어의 시뮬레이터를 사용하여 진행하였다.

먼저 고정소수점 모의실험은 시스템 출력의 음질평가를 위해 수행되었다. 고정소수점 모의실험은 유한한 길이를 가진 데이터 표현 형식에서 발생하는 오차를 감소시키기 위한 것이다.

설계된 시스템의 음질을 객관적으로 측정하기 위해서 ISO/IEC 13818-4 Compliance Test [6]를 수행하였다. 검증의 편의를 위해서 2 개의 파라미터, NL(Noise Level)과 MER(Maximum Error Rate)를 다음과 같이 정의 하였다.

$$NL = 20 \log_{10} \left(\frac{\text{Error R.M.S value}}{1} \right) \text{ dB FS}$$

$$MER = \frac{\text{Maximum Error}}{2^{-14}}$$

Compliance test 의 요구조건은 -101dB FS 이하의 NL 과 1 이하의 MER 을 만족하는 것이다. 구현된 시스템의 성능은 표 2 에 제시하였다. 여기서 Np 는 프로세싱 비트 수이고 No 는 출력 PCM 의 비트 수이다. 실험 결과에서 보듯이 20 비트의 해상도는 음질과 시스템 복잡도를 가장 잘 절충한 형태이다. 이 결과에서 AAC 복호화기를 구현함에 있어서 20 비트 DSP 코어의 사용은 타당한 선택임을 알 수 있다.

본 논문에서 사용한 프로세서 코어는 최소 40MHz 의 시스템 클럭에 대해 검증이 되었고 [5], 구현된 복호화기는 최대 48kHz 의 표본화 주파수를 지원하도록 하였다. 그러므로 하나의 오디오 샘플에 대응되는 클럭 수는 833.3 사이클이고, 한 오디오 프레임의 샘플 수는 1024 샘플이므로 한 프레임 처리에 사용 가능한 클럭 수는 853,299 사이클이다. 따라서 복호화 과정을 실시간으로 처리하기 위해서는

한 프레임에 대한 복호화 과정이 853.299 사이클 안에 수행되어야 한다. 전용 하드웨어와 프로세서 코어로 소프트웨어적으로 구현된 전체 복호화 과정의 연산량은 표 3 과 같다.

설계된 시스템은 2 채널의 MPEG-2 AAC main 프로파일 비트열을 16.9 MIPS (361.828 사이클)에 처리할 수 있다. 시스템의 복잡도는 표 4 와 같다. 제안된 시스템은 SAMSUNG 0.35 μ m 3.3V CMOS technology 의 VLSI 로 합성하였다. 시스템의 게이트 수는 표 5 에 제시하였다. 구현된 시스템은 참고문헌 [7]에서 제시한 LC 프로파일의 고정 소수점 복호화기보다 우수한 성능을 보이고 있다.

표 2 구현된 AAC 복호화기의 객관적 음질 평가

| No | No = 16 | | No = 18 | | No = 20 | |
|----|---------|------|---------|------|---------|------|
| | NL | MER | NL | MER | NL | MER |
| 16 | -85.6 | 5.05 | -85.6 | 5.05 | -85.6 | 5.05 |
| 17 | -90.3 | 3.19 | -90.5 | 3.15 | -90.5 | 3.15 |
| 18 | -94.0 | 2.33 | -94.6 | 2.27 | -94.6 | 2.27 |
| 19 | -96.5 | 1.62 | -97.6 | 1.42 | -97.7 | 1.39 |
| 20 | -97.8 | 0.25 | -100.0 | 1.17 | -100.2 | 1.11 |
| 21 | -100.0 | 1.25 | -104.0 | 0.71 | -104.5 | 0.65 |
| 22 | -100.9 | 0.75 | -108.2 | 0.64 | -109.7 | 0.64 |
| 23 | -101.2 | 0.64 | -110.9 | 0.64 | -114.5 | 0.64 |
| 24 | -101.3 | 0.64 | -111.9 | 0.64 | -117.3 | 0.64 |

표 3 구현된 시스템의 연산량

| 복호화 과정 | MIPS |
|---------|-------|
| 무손실 복호화 | 5.26 |
| 스테레오 | 0.64 |
| 예 측 | 0.65 |
| TNS | 4.94 |
| 필터뱅크 | 5.41 |
| 합 계 | 16.90 |

표 4 구현된 시스템의 복잡도

| | 2 채널(word) |
|----------|------------|
| 프로그램 메모리 | 4.1k |
| 데이터 ROM | 5.5k |
| 데이터 RAM | 7.1k |

표 5 합성된 시스템의 게이트 수

| | 게이트수(gate) |
|----------|------------|
| DSP 코어 | 22585.5 |
| 허프만 복호화기 | 3987.5 |
| 예 측기 | 5760.8 |
| 합 계 | 32333.8 |

5. 결 론

MPEG-2 AAC 시스템은 디지털 오디오의 최신 기술들이 집약되어, 기존의 MP3 시스템에 비해서 뛰어난 음질과 압축률을 지닌 기술로서 평가되고 있다. 본 연구에서는 MAIN 프로파일 MPEG-2 AAC 복호화 시스템을 구현함에 있어서, 각 과정별 연산의 특징을 분석하고, 소프트웨어와 하드웨어의 작업 할당과 스케줄링을 통하여 최적화된 구조를 제안하고 구현하였다. 제안된 시스템은 소프트웨어 작업을 위한 고정 소수점 프로세서 코어와 하드웨어 작업을 위한 2 개의 하드와이어드 모듈로 구성되어 있다. 2 개의 하드와이어드 모듈은 허프만 복호화기와 예측기이다. 전체 시스템은 VLSI 를 이용한 ASIC 으로 구현되었고, 소프트웨어는 DSP 코어의 어셈블리로 구현되었다.

설계된 시스템을 검증하기 위해서, DSP 코어의 시뮬레이터 모델을 사용하였다. ISO/IEC 13818-4 Compliance test 를 통하여, 구현된 시스템의 객관적 음질을 보장할 수 있었다.

구현된 시스템은 40MHz 로 동작하는 DSP 코어 시스템에서 16.9 MIPS (16.9MHz)로 2 채널의 MAIN 프로파일 AAC 비트열을 고품질로 복호화할 수 있어서, 기존의 연구들에 비해서 우수한 성능을 보이고 있음을 확인할 수 있었다.

참고 문헌

- [1] ISO/IEC JTC1/SC29/WG11 No. 1650 "IS 13818-7 (MPEG-2 Advanced Audio Coding, AAC)," April, 1997.
- [2] M. Bosi and et al., "ISO/IEC MPEG-2 Advanced Audio Coding," Journal of Audio Engineering Society, Vol. 45, No. 10, pp. 789-814, October, 1997.
- [3] J. P. Princen, A. W. Johnson, and A. B. Bradley, "Subband/Transform Coding Using Filter Bank Design Based on Time Domain Aliasing Cancellation," Proceeding of ICASSP, pp.2161-2164, 1987.
- [4] P. Duhamel, Y. Mahieux, and J. P. Petit, "A fast algorithm for the implementation of filter banks based on 'time domain aliasing cancellation'," Proceeding of ICASSP, pp. 2209-2212, May, 1991.
- [5] Kyoung Ho Bang, Nam Hun Jeong, Young Cheol Park and Dae Hee Youn, "Design Optimization of Main-Profile MPEG-2 AAC Decoder," Proc. IEEE ICASSP, pp.989-992, May, 2001.
- [6] ISO/IEC 13818-4 (Information Technology - Generic Coding of Moving Pictures and Associated Audio: Conformance)," March, 1996.
- [7] Matthew A. Watson and Peter Buettner, "Design and implementation of AAC decoders," IEEE Transactions on Consumer Electronics, Vol. 46, Issue:3, pp.819-824, August, 2000.