

MPEG-2 AAC 디코더의 DSP 구현에 관한 연구

정중훈, 김정근, 이재식, 장태규, *장홍엽
 중앙대학교 전자전기공학부, *삼성전자

DSP based implementation of MPEG-2 AAC decoder

J.H.Jeong, J.K.Kim, J.S.Lee, T.G.Chang, *H.Y.Jang

Department of Electrical and Electronics Engineering, *Samsung Electronics

Email : tgchang@jupiter.cie.cau.ac.kr

요약

본 논문에서는 MPEG-2 AAC 디코더의 DSP 구현에 관한 연구결과로서, ISO/IEC 13818-7 표준에 의거 구현된 MPEG-2 AAC 디코더의 각 세부 기능블럭들의 구성 및 동작원리에 대하여 요약 기술하고, DSP연산에 최적화된 연산구조의 연구를 바탕으로 16bit 고정소수점 연산구조를 가지는 DSP상에서 구현된 MPEG-2 AAC 디코더의 시스템의 하드웨어 및 소프트웨어 구성에 관하여 간략한 기술하였다. 구성된 디코더의 성능평가를 통하여 MPEG-2 AAC 비트스트림을 디코딩하기 위하여 필요로 하는 연산량 및 소요 메모리의 양을 측정하고, 디코더 성능의 중요 척도인 음질평가를 수행하였다. 수행방법으로서 conformance test에 의거하여 PSNR을 측정함으로써 객관적인 성능 지표의 제시와 함께, 주관적인 음질 평가도 병행하여 수행하였다.

1. 서론

MPEG-2 AAC(Advanced Audio Coding)는 다채널, 고음질, 고압축률의 특징으로 인하여 사용범위가 점점 넓어지고 있는 디지털 오디오 압축표준이다. MPEG-1 오디오가 128kbps의 단일 또는 스테레오 채널의 지원하는데 비하여, MPEG-2 AAC는 320kbps의 전송속도에서 5.1채널의 매우 우수한 음질을 지원한다. 또한 지원 시스템의 사양에 따라 Main, LC, SSR의 세가지 프로파일을 제공하고, ADIF, ADTS의 두가지 프레임 구성방법을 제공함으로써 최적의 시스템 구현이 가능하다.

본 논문에서는 ISO/IEC 13818-7 표준에 의거 구현된 MPEG-2 AAC의 세부 기능블럭들의 동작원리 분석을 통하여 독자 디코더 구현모델을 도출하였다. 또한 DSP연산에 최적화된 연산구조의 연구를 바탕으로 16bit 고정소수점 연산구조를 가지는 DSP상에서 실시간 동작 가능한 MPEG-2 AAC 디코더를 구현하고 동작을 확인하였다. 또한 구현된 디코더의 음질평가를 위하여 ISO/IEC 13818-4의 기준에 근거, Conformance test를 통한 성능평가를 수행하였다. [1][2]

2. MPEG-2 AAC 디코더의 구성 및 동작원리

MPEG-2 AAC의 주요한 압축 원리는 인간의 청각 특성을 고려한 심리음향모델(psychoacoustic model)에 근거를 두고 있다. MPEG-2 AAC의 인코딩 과정중 대부분의 데이터 압축은 양자화 과정에서 이루어지는데 이때 청각 심리음향모델을 고려하여 양자화 잡음의 분포를 조절함으로써 음질의 손실을 최소화하면서 높은 압축률을 얻을 수가 있게 된다.

2.1 MPEG-2 AAC 디코더 구성

그림 1에서는 MPEG-2 AAC 디코더의 처리 과정 흐름도를 나타내고 있다.

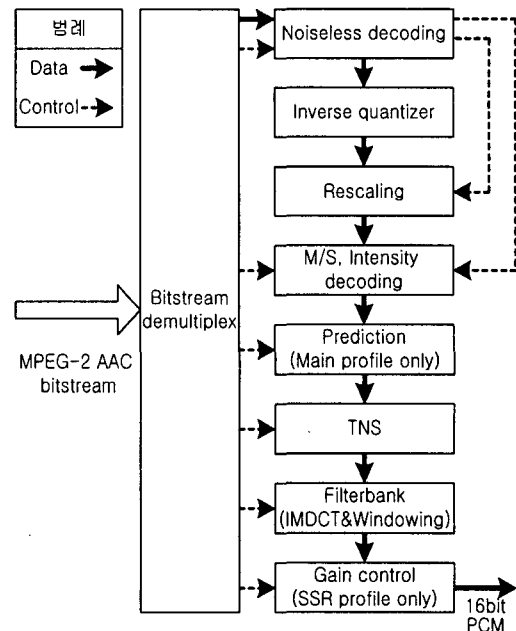


그림 1 MPEG-2 AAC 디코더의 기능블럭도

ISO/IEC 13818-7의 표준에 따라 압축된 데이터는 헤더 및 기타 부가정보에 따라 디코딩을 수행하게 된다. Noiseless decoding과정에서는 주로 허프만 디코딩을 수행한다. 허프만 코딩과정에서는 주파수 대역의 데이터 분포의 통계적 특성에 따라 11개의 서로 다른 허프만 테이블 중 최적의 테이블을 선택하여 사용한다. 일반적으로 허프만 디코딩 과정은 그 수행방법에 따라 소요되는 디코딩 시간의 차이가 매우 크다. 본 논문에서는 이진트리 방법에 기초한 새로운 허프만 디코딩 방법을 도입함으로써 디코딩 시간의 단축 및 소요 메모리를 최소화하도록 하였다. [3][4]

Inverse quantizer에서는 스펙트럼 데이터들의 분포를 고려하여 식(1)과 같이 비선형 양자화를 수행한다.

$$x_invquant = \text{Sign}(x_quant) \times x_quant^{\frac{4}{3}} \quad (1)$$

이때 발생된 양자화 잡음은 scalefactor를 적용하는 rescaling과정과, TNS(Temporal Noise Shaping)과정에서 양자화 잡음을 조절한다. 전자는 주파수 영역에서, 후자는 시간 영역에서 각각 양자화 잡음의 분포를 조절하는 역할을 수행한다. Scalefactor 값들은 주파수상의 인접 데이터들의 차분신호들을 허프만 코드화하여 압축을 수행한다.

MPEG-2 AAC 채널중 CPE(Channel Pair Element)의 경우는 좌/우측 음의 연관성을 고려하여 압축을 수행한다. 저주파의 경우는 식(2)와 같이 두 신호의 합과 차를 고려한 M/S(Mid/Side)코딩을, 고주파의 경우는 식(3)과 같이 좌/우측의 음의 세기의 차이를 고려한 intensity stereo 코딩을 수행한다.

$$\begin{bmatrix} l \\ r \end{bmatrix} = \begin{bmatrix} +1 & -1 \\ +1 & +1 \end{bmatrix} \begin{bmatrix} m \\ s \end{bmatrix} \quad (2)$$

$$r[g][spec] = scale \times l[g][spec] \quad (3)$$

Prediction 기능블럭은 Main profile 경우에만 사용되는 기능블럭으로서 인코더에서는 프레임들간의 스펙트럼 데이터의 유사성을 이용하여 다음 프레임의 예측신호와 실제 신호의 잔차 신호를 전송하고 디코더에서는 이 잔차신호로부터 원신호를 복원하는 과정을 수행한다.

Filterbank 기능블럭은 주파수 영역의 데이터를 시간영역의 데이터로 변환시키는 DCT(Discrete Cosine Transform)과정과 블럭연산과정에서 발생하는 블럭간의 불연속성을 제거하기 위한 windowing을 수행한다. 이때 음의 전이구간에서 발생하는 pre-effect 현상을 줄이기 위하여 long window(2048 sample)과 short window로 구분하여 변환을 수행한다. 이 filter

bank는 변환과정에서 많은 연산량이 소요되기 때문에 최적 연산 알고리즘을 수행하여야 한다. [5][6]

Gain control 기능블럭은 SSR프로파일의 경우에만 사용되는 기능블럭으로서 가변적인 샘플링주파수를 선택할 수 있도록 한다.

2.2 MPEG-2 AAC 디코더의 최적연산 알고리즘 구현

MPEG-2 AAC 디코더의 최적화는 수행속도의 증가를 위한 최적화, 소요 메모리의 감소를 위한 최적화, 고정소수점 연산 수행 시 유한비트 근사화의 영향을 최소화하기 위한 최적화로 구분할 수 있다.

본 논문에서는 식 (4)의 경우와 같이, 연산량을 줄이기 위한 검색테이블 작성과 최적 수치연산 기법의 적절한 조합으로 처리속도의 증가와 더불어 메모리의 소모를 최소화 할 수 있도록 하였다.

$$\begin{aligned} x_rescal &= x_invquant \times 2^{0.25 \cdot sf} \\ &= x_invquant \times 2^{\ln((sf/4) + (sf \% 4))} \quad (4) \\ &= x_invquant \times 2^{sf \gg 2} \times 2^{sf \% 4} \\ &= x_invquant \ll (sf \gg 2) \times 2^{sf \% 4} \\ &\approx 2^{sf \% 4} \rightarrow 2^{0.4}, 2^{1.4}, 2^{2.4}, 2^{3.4} \end{aligned}$$

유한비트 근사화의 영향을 최소화하기 위한 최적화는 inverse quantizer의 경우처럼 수치 표현의 정밀도가 매우 중요하거나, filter bank의 경우와 같이 반복적인 수치 연산에 의하여 결과를 도출하는 기능블럭의 경우에 있어 수치 표현의 범위를 정밀하게 산출후 최대의 표현범위를 나타내도록 하였다.

3. MPEG-2 AAC 디코더의 DSP 구현

본 논문에서는 DSP기반의 실시간 동작 가능한 MPEG-2 AAC 디코딩 시스템을 구현하고, 동작확인 및 성능검증을 수행하였다.

3.1 MPEG-2 AAC 디코딩 시스템 구현

그림 2에서는 본 연구에서 구성한 DSP기반의 MPEG-2 AAC 디코딩 시스템의 하드웨어 구성을 간략한 블록도를 이용하여 나타내었다.

시스템 구성을 위하여 16bit 고정소수점 연산구조를 가지는 TeakLite DSP를 기반으로 디코딩 시스템을 구현하였다. 디버거 호스트 PC는 MPEG-2 AAC 디코더의 컴파일 및 대상 DSP로의 프로그램 다운로드를 수행하고, 디코딩 수행 중에는 DSP의 레지스터값의 동작 상태를 모니터링하고, 동작제어 및 상태값들을 알려주는 역할을 수행한다. 데이터 호스트 PC는 DSP로부터의 데이터 요청 신호에 맞추어 MPEG-2 AAC 비트스트림을 I/O 포트를 통하여 DSP로 전송하는 역할을 수행한다.

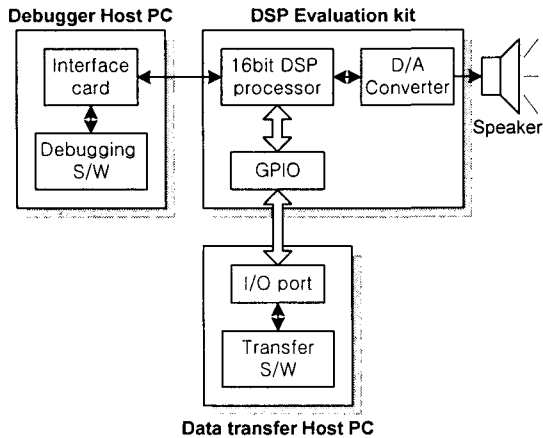


그림 2 MPEG-2 AAC 디코더의 하드웨어 구성도

이때 요청신호의 수락 및 데이터 전달과정은 인터럽트 신호에 의하여 동작함으로써, 상태파악을 위한 부담을 줄임으로써 원활한 동작을 수행할 수 있도록 한다. 대상 DSP에서는 데이터 호스트 PC로부터 전송받은 비트스트림을 사용하여 디코딩을 실시하고, 생성된 오디오 신호를 D/A변환기로 전송함으로써 실시간 디코딩을 수행하게 된다.

3.2 MPEG-2 AAC 디코딩 프로그램 구성

그림 3은 MPEG-2 AAC 디코딩 프로그램의 신호 흐름을 나타내고 있다.

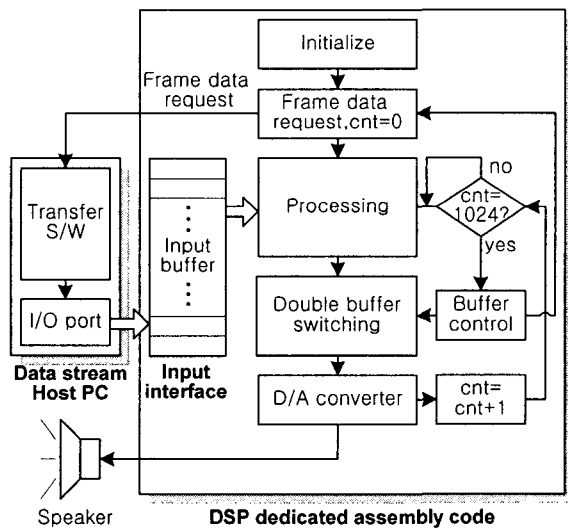


그림 3 MPEG-2 AAC 디코더 소프트웨어 구성도

디코딩 프로그램의 동작 시작과 함께 DSP의 메모리 및 I/O 구성 및 D/A 변환기의 초기화를 수행한다. 또한 버퍼 및 내부 변수들의 초기화도 함께 수행한다.

초기화를 마친 후 DSP에서는 데이터 호스트 PC측으로 디코딩을 수행하기 위한 비트스트림 데이터를 요청한다. 이 신호는 DSP와 연결된 GPIO(General Purpose I/O)를 통하여 데이터 호스트 PC의 I/O포트를 전달된다. 데이터 호스트 PC측에서는 데이터 요청 인터럽트신호에 반응하여 한프레임 데이터를 I/O포트를 이용하여 전송하고, 이와 함께 데이터 전송 인터럽트 신호를 보낸다. DSP는 입력버퍼에 받아놓았던 비트스트림을 이용하여 디코딩을 수행중 데이터 전송 인터럽트 신호에 반응하여 인터럽트 서비스 루틴을 수행하고, 입력 원형버퍼에 전달 받은 데이터들을 저장한다. 디코딩 중에는 이 원형 버퍼로부터 비트스트림 데이터를 추출하여 디코딩을 수행하게 된다. 데이터의 출력은 double buffer switching기법을 사용하는데, 이것은 두 개의 출력버퍼 공간을 마련하여 하나의 버퍼에서는 D/A 변환기측으로 디코딩을 마친 16bit PCM 오디오 데이터를 전달하여 스피커로 출력을 수행하고, 다른 하나의 출력버퍼에서는 디코딩을 마친 데이터를 저장하는 과정을 수행한다. 한 프레임의 재생 및 디코딩이 끝난 후에는 출력버퍼는 오디오 신호의 저장버퍼로, 오디오 신호의 저장버퍼는 출력버퍼로 두 개의 버퍼가 가지는 기능을 교환하게 된다. 이와 같이 double buffer switching 기법을 도입함으로써 음의 출력과 디코딩 과정을 동시에 수행함으로써 오디오 신호 출력 중 음의 단절 없이 재생이 가능토록 하였다.

4. 수행결과

본 절에서는 ISO/IEC 13818-7의 표준에 기준 하여 구성된 MPEG-2 AAC 디코더의 메모리 및 MIPS 소요량을 측정하고, 생성된 오디오 데이터의 음질 평가에 대한 결과를 요약하여 기술하였다.

MPEG-2 AAC는 프로세서의 성능 및 지원 가능한 메모리의 크기에 따라 Main, LC, SSR의 세가지 프로파일의 선택이 가능하다. 본 논문에서 대상으로한 DSP는 최대 100MIPS의 연산능력을 가지고 있으며, 지원가능한 메모리는 64kbyte(내부 20kbyte + 외부 44kbyte)이다. 본 논문의 소스코드는 Main 및 LC 프로파일의 디코딩이 가능하며, DSP기반의 코드는 LC프로파일을 대상으로 구성하였다.

표 1에서는 DSP기반의 MPEG-2 AAC 디코더의 리소스 소요량을 나타내고 있다. 컴파일된 디코더 프로그램을 저장하기 위한 공간은 3.6kword, 디코딩 과정 중 각종 변수 및 입출력 데이터들의 저장을 위한 데이터 메모리 공간은 35.6kword가 소요되어 총 39.2kword의 메모리 공간이 소요되었다. 디코딩을 위한 MIPS 소요량은 인코딩된 비트 스트림의 구성 및 압축정도, 샘플링 주파수에 따라 많은 차이가 발생하는데 44.1kHz의 샘플링 주파수와 96kbps의 전송량을 가지는 스테레오 채널의 비트스트림을 디코딩시 평균 28.1MIPS가 소요되었다.

표 1 MPEG-2 AAC 디코더의 메모리 및 소요 연산량

기능블록	자원	메모리 (kword)		연산량 (MIPS)
		Program	Data	
bitstream demultiplex		1.1	0.3	2.4
noiseless decodig		0.5	3.5	2.4
inverse quantizer		0.1	4.0	1.2
scalefactor		0.1	7.4	1.2
M/S.intensity coding		0.2	0.4	2.4
TNS		0.4	1.7	-
filterbank		0.9	13.3	18.5
I/O buffering		0.3	5.0	-
total		3.6	35.6	28.1

오디오 디코딩에 있어서는 소요 리소스뿐만 아니라 디코딩된 신호의 충실도 또한 매우 중요한 요소중의 하나이다. 본 논문에서는 MPEG-2 AAC 디코더의 성능 평가 기준인 ISO/IEC TR 13818-4에서 제시하는 기준에 의거 성능평가를 수행하였다. 디코더의 성능은 식 (5)와 같이 평균 SNR로 측정한다.

$$\sqrt{\frac{1}{N} \sum_{n=1}^N \text{diff}(n)^2} \quad (5)$$

이 식을 통한 성능평가의 합격선으로는 $s \geq \frac{1}{2^{13} \times 2^{0.5}}$

로서 평균 SNR이 100.3[db]이상을 만족하여야 한다. 또한 디코딩된 전체 구역중 1bit이하의 bit error만이 존재하여야 한다.

성능평가를 위한 비교대상으로는 MPEG에서 제공하는 테스트 비트스트림과, 본 논문에서 구성한 디코더를 통해 생성된 PCM 오디오 데이터를 표준안에서 제시하는 평가식을 이용하여 비교 및 판단을 수행하였다. 평가 결과는 표 2와 같고, 부동소수점 연산구조의 경우 평균 약108[db], 16비트 고정소수점 연산구조의 경우 약 83[db]의 결과를 얻을 수 있었다. 고정소수점 연산의 경우 conformance test를 만족시키지는 못하였으나, 16bit 고정소수점 연산을 통한 연산 결과임을 고려시 만족할만한 결과라 할 수 있겠다.

표 2 MPEG-2 AAC 디코더의 PSNR 측정결과

테스트 파일	PSNR [db]		비교
	floating point	fixed point	
test01.aac	109.10	73.29	ADIF, mono
test02.aac	108.44	62.63	ADIF, mono
test03.aac	109.38	73.57	ADIF, mono
test04.aac	108.42	79.27	ADIF, stereo
test05.aac	101.63	84.21	ADIF, stereo
test06.aac	108.29	90.43	ADTS, stereo
test07.aac	106.18	91.56	ADTS, stereo
test08.aac	115.55	92.12	ADTS, stereo
test09.aac	116.09	92.07	ADTS, stereo
test10.aac	102.11	88.41	ADTS, stereo
평균	108.52	82.76	-

5. 결 론

본 논문에서는 MPEG-2 AAC 디코더의 DSP구현에 관한 연구결과로서 각 세부 기능블럭들의 구성 및 동작원리에 대하여 요약 기술하고, MPEG-2 AAC 비트스트림의 실시간 디코딩을 수행하기 위한 시스템의 하드웨어 및 소프트웨어의 구성 및 동작과정에 대하여 기술하였다.

16bit 고정소수점 연산구조를 가지는 DSP를 기반으로 디코딩 시스템 구현결과 28.1MIPS의 연산량의 소요 및 3.6kword의 프로그램 메모리 공간, 35.6kword의 데이터 메모리 공간이 소요되었다. 또한 디코더 성능의 중요한 평가 기준인 음질 테스트 결과 부동소수점 연산구조 적용시 108.52[db], 고정소수점 연산구조 적용시 82.76[db]의 PSNR출력으로 만족할 만한 결과를 나타내었으며, 청각에 의한 주관적 평가에서도 만족할만한 수준의 음질을 확인할 수 있었다.

본 연구를 통하여 MPEG-2 AAC 디코딩 시스템의 구현에 관한 기술습득을 통하여, 차후 다양한 분야에 적용이 예상되는 오디오 압축기술의 요소기술확보에 기여될 수 있을 것이라 사료된다.

참고문헌

- [1] ISO/IEC JTC1/SC29/WG11 N1650 "IS 13818-7 (MPEG-2 Advanced Audio Coding, AAC)"
- [2] ISO/IEC JTC1/SC29/WG11 N2201 "ISO/IEC TR 13818-4. Conformance test"
- [3] J. H. Jeong, B. I. Kim, T. G. Chang, "Huffman decoding method based on bit-wise comparison", IEEK Summer conference, Vol4, pp. 131-134, 2001
- [4] S.HO and P.LAW, "Efficient hardware decoding method for modified Huffman code", Electronics letters, Vol.27, pp.855-856, 1991
- [5] P. Duhamel, Y. Mahieux, "A fast algorithm for the implementation fo filter banks based on time domain aliasing cancellation", IEEE, pp 2209-2212, 1991
- [6] A.N.Skodras, "Efficient computation of the split-radix FFT", IEE proceedings, Vol.139 pp.60 1992