

ARM920T 를 이용한 Dolby AC-3 decoder 의 실시간 구현

하호진, 김수현

삼성전자, 디지털 미디어 연구소, 소프트웨어 솔루션 팀

Real-time implementation of the Dolby AC-3 decoder using ARM920T

Jin-Ho Ha, Su-Hyun Kim

Software Solution Team, Digital Media R&D Center, Samsung Elec.

e-mail) hjini@samsung.com, hyunii@samsung.com

요 약

본 논문에서는 미국에서 고 선명(High Definition) TV 를 위한 오디오 표준인 ATSC 에 의해서 채택되고, DVD 의 오디오 코덱으로 널리 채택된 Dolby AC-3 복호화기를 고정 소수점으로 변환한 후, ARM920T 를 기반으로 실시간 구현하였다.

실시간 구현에 사용된 ARM920T 는 저전력 소비형 32-비트 고정 소수점 RISC 구조이며, 명령어 Cache 와 데이터 Cache, MMU(Memory Management Unit)로 구성 되어 있다. 그리고 200MIPS 의 성능을 가지고 있어, 음성/오디오, 통신, 디지털 셀룰라폰에 적용할 수 있고, ASIC 화 할 수 있는 장점을 가지고 있다.

구현된 AC-3 복호화기는 약 67MIPS(6 채널, 640kbps, 48KHz)의 복잡도를 나타내며, 사용된 메모리량은 프로그램 ROM 23.3K Bytes, 데이터 ROM(table) 4.62K Bytes 및 RAM 45K Bytes 이다. 구현된 AC-3 복호화기는 Dolby 사에서 제공하는 40 개의 test 벡터들을 bit-exact 하게 통과하였다.

1. 서 론

디지털 오디오 방송, 위성 TV 및 멀티미디어의 다양한 응용 분야에서, CD 수준의 오디오 신호를 낮은 비트율에서 전송 및 저장하기 위해서, 오디오 신호의 압축은 필수적이다. AC-3 오디오 기술은 다양한

오디오 채널(6 채널)과 저 비트율로 디지털 오디오를 압축할 수 있는 방식이다. 채널의 범위는 8 채널까지 가능하고, 일반적으로 모노, 스테레오, 그리고 6 채널을 포함하는 서라운드 포맷이 있다. 그리고 전송률은 32bps - 640kbps, 그리고 샘플링 레이트는 48, 44.1, 32KHz 까지 지원된다. CD 수준의 오디오 신호를 12:1 이상의 압축으로 복원음을 재생할 수 있는 고품질 오디오 부호화 기술이다[1]. 이러한 오디오 신호 압축의 핵심은 인간의 청각적 지각 특성을 이용하여 잡음을 은폐 시키는 심리음향 모델과 오디오 신호의 중복성을 이용하는 방법이라고 할 수 있다. 특히 AC-3 는 부호화 효율을 향상시키기 위해서, 주파수 커플링 기법과 여러 채널에 대해 효과적인 비트 할당 방법을 이용한다[3].

Dolby AC-3 오디오 부호화기의 기본 구조는 그림 1 에 나타나 있다. 입력 오디오 데이터는 50% overlap 되는 512 point MDCT(Modified Discrete Cosine Transform)을 사용하여 주파수 영역으로 옮겨진다. 이때 프리에코(Pre-echo)와 같은 영역을 다루기 위해서 블록 스위칭 기술을 사용한다. 이 경우에는 256 point MDCT 를 사용하게 된다. 이러한 주파수 영역의 계수에 대해서 지수(exponent)와 가수(mantissa)로 나누어, 먼저 지수에 대해서 양자화가 이루어 진다. 계수의 분포를 이용하여, 지수에 대한 양자화 방법(D15, D25, D45)을 선택한다. 이 결과와 심리 음향 모

델을 근거로 하여, 가수의 양자화 방법으로 파라미터 비트 할당 방법을 이용 한다. 이때 사용되는 심리음향모델은 마스킹 임계값을 사용하여 인간 청각계의 특성에 의한 지각적인 중복성을 제거한다. 마스킹 임계값을 근거로 주관적인 잡음이 들리지 않도록 양자화가 이루어지게 된다[2]. 결과적으로 잡음이 최소화된 양자화 비트는 부가 정보와 함께 비트열로 만들어 지게 된다.

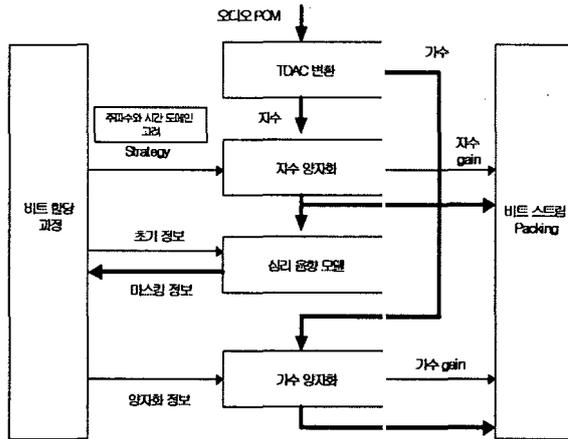


그림 1. AC-3 부호화기의 기본 구조

본 논문에서 사용한 ARM920T 칩은 ARM9TDMI를 기반으로 MMU, 각각 16K Byte 명령어 Cache 와 Data Cache 를 사용하고 200MIPS 의 성능을 가지고 있다.

본 논문의 구성은 2 장에서 AC-3 복호화기의 기본구조와 특성에 대하여 살펴보고 3 장에서는 실시간 구현에 관해 논하며 마지막으로 4 장에서 결론을 맺도록 하겠다.

2. AC-3 복호화기

AC-3 복호화기는 3 부분으로 크게 나눌 수 있다. 프레임 헤더정보 디코딩, 지수, 가수 역 양자화, 합성 필터 뱅크이다. 개략적인 블록도는 그림 2 와 같다.

2.1 프레임 헤더정보 디코딩

AC-3 비트 스트림은 그림 3에서와 같은 구조로 되어있다. 싱크 정보(Sync info.), 비트 스트림정보(Bit Stream Info.), 6 개의 오디오 블록 정보, 부가정보, 그리고 CRC 체크로 구성되어 있다. CRC 체크는 프레임의 5/8 부분과 나머지 3/8 의 2 부분을 체크 하게 된다. 싱크정보와 비트 스트림 정보에는 한 프레임을 제어할 수 있는 전체적인 정보를 포함한다. 비트

를, 인코딩된 채널 수, 프레임 크기, 샘플링 주파수 등에 관한 정보가 할당되어 있다.

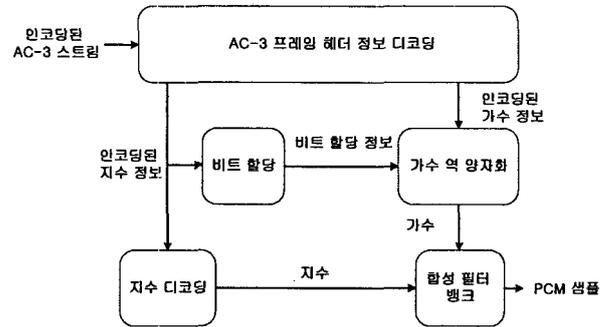


그림 2. AC-3 복호화기의 기본 구조

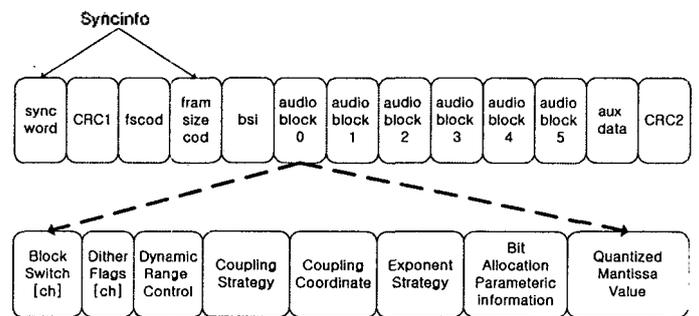


그림 3. AC-3 프레임 및 오디오 블록 기본 구조

AC-3 는 6 개의 오디오 블록으로 구성되어 있고, 이를 구성하기 위해서 각각의 오디오 블록은 블록 스위치 정보, 커플링, 가수 와 지수, 비트 할당 정보를 포함하고 있다. 그리고 각 오디오 블록의 정보는 데이터 공유가 가능하여, 블록 0 의 정보가 다른 블록에서도 사용이 가능하다. 커플링(Coupling)기술은 데이터의 압축률을 높이기 위해서 높은 주파수에 대한 인간의 청각적인 특성을 이용한 기술이다. 하지만 이 방법은 음질의 손상을 가져올 수 있기 때문에 낮은 비트율에서는 제한된다. 그리고 AC-3 는 출력 채널의 수를 조정하기 위해서, Down mixing 처리를 한다. Dynamic Range Control 은 소리의 레벨을 조정 함으로써, 소비자의 의도에 맞게 소리의 범위를 조정할 수 있는 요소이다. Dual mono 의 경우에 대해서 적용된다[1].

2.2 지수, 가수의 역 양자화

가수는 각각의 오디오 블록에서 공유가 가능하다. 즉 AB0(Audio Block 0)의 가수는 AB1 에서 AB5 까지 공유(REUSE)가 가능하다. 하지만 AB0 의 가수는 항상 전송된다. 지수는 D15, D25, D45 의 3 가지

방법으로 디코딩이 되어 진다. 중간 1,2,4는 공유되는 지수의 개수를 의미하는 것이고, 마지막의 5는 양자화 레벨을 의미한다. 주파수 도메인의 계수가 높은 주파수 해상도를 필요로 할 경우에는 D15가 사용되고, 낮은 경우에는 D45가 사용된다[3].

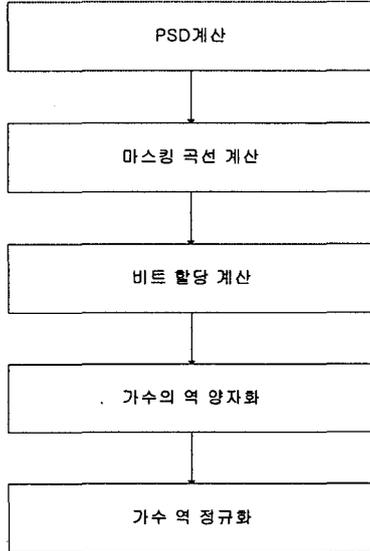


그림 4. 가수의 역 양자화

처음의 지수는 독립적으로 4 비트로 보내지고, 나머지는 -2 - 2의 변화 범위를 가지면서, 3개씩 그룹으로 묶어지고, 7 비트(25*4+5*4+4 = 124)로 전송된다. 지수에 대한 분해가 끝난 후, 가수에 대한 역 양자화가 수행된다. Bap(Bit Allocation Pointer)에 따라, 정해진 레벨의 역 양자화가 수행된다. Bap가 5보다 작을 경우에는, 대칭(Symmetric) 역 양자화 레벨을 갖고, bap가 5보다 클 경우에는 비대칭(Asymmetric) 역 양자화를 수행한다. bap가 12일 경우에는 2개씩 역 그룹핑 방법을 사용하고, bap가 4일 경우에는 2개의 가수에 대해서 역 그룹핑을 수행한다. 그림 4는 가수 역 양자화에 대한 블록도이다[4][5][6][7].

2.3 합성 필터 뱅크

역양자화를 거쳐 얻어진 주파수 도메인 값을 시간 도메인으로 변환하기 위해서 IMDCT(Inverse MDCT)와 윈도잉 하는 과정을 수행한다. 역 양자화를 통하여 나온 256개의 값을 long window에서는 512point IMDCT를 수행하고, short window에서는 256point IMDCT를 수행한다. 하지만 AC-3에서는 빠른 IMDCT를 수행하기 위해서, 이 과정을 Pre-IFFT, IFFT, Post-IFFT의 3부분을 나누어 이 과정을

수행하게 된다. 이 과정에 사용되는 식은 다음과 같다.

$$Z[k] = (X[N/2 - 2k - 1] + jX[2k]) + (\cos l[k] + j \sin l[k]) \quad (\text{식 1})$$

$$z[n] = \sum_{k=0}^{N/4-1} Z[k] \times e^{j \frac{8\pi n k}{N}} \quad n = 0 \dots \frac{N}{4} - 1 \quad (\text{식 2})$$

$$y[k] = (z[k]) + (\cos l[k] + j \sin l[k]) \quad (\text{식 3})$$

$$\cos l[k] = -\cos(2\pi(8k+1)/(8N)) \quad (\text{식 4})$$

$$\sin l[k] = \sin(2\pi(8k+1)/(8N))$$

여기서 long 윈도우는 N이 512이고, short 윈도우는 256이다. IMDCT를 수행 후, 윈도잉 과정을 수행하게 된다. 이 과정에서 512개의 시간도메인 샘플이 생성되고, 처음 256 샘플은 그 이전 블록 후반부의 256 샘플과 더해져서 최종적인 PCM 샘플이 출력되게 된다. 이 과정을 그림 5와 같다.

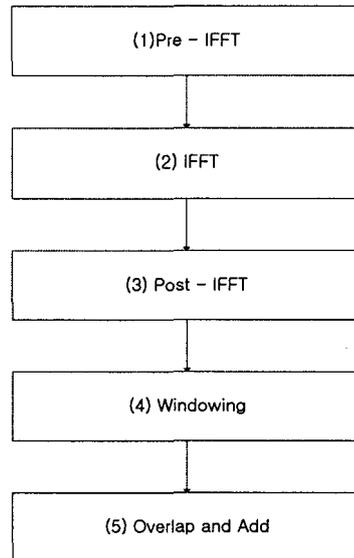


그림 5. IMDCT, windowing, Overlap and Add 과정

3. 실시간 구현

AC-3의 실시간 구현에 사용된 보드는 ARM9TDMI를 기반으로 설계된 ARM920T 칩이 탑재되어 있다. 또한 ARM920T 칩은 128K Bytes의 RAM과 128K Bytes의 ROM의 메모리를 확장할 수 있다.

DSP 프로그램의 개발은 준비 단계, 코딩 그리고 검증 단계의 3가지 부분으로 나누어서 수행하였다. 우선 준비 단계에서는 알고리즘을 분석하고 부동 소수점 C 소스를 고정 소수점 ARM9TDMI에서 작동되

도록 고정소수점 C 소스로 변환하는 작업을 수행하였다. 그리고 AC-3 디코더에 필요한 모듈을 파악하고 call tree 를 작성하였다. 준비 단계에서는 프로그램의 형식, 각 모듈의 depth 에 따른 메모리의 위치, 지역 변수와 전역 변수의 지정 그리고 함수 호출 시 인자의 전달방법을 미리 프로그래머 상호간에 규약을 정해서 프로그램을 개발하였다. AC-3 복호화기의 코딩은 프레임 헤더정보 디코딩, 지수, 가수 역 양자화, 합성 필터 뱅크의 3 가지로 구분 지을 수 있다. 프로그램을 코딩할 때에는 어느 정도 관련이 있는 모듈들로 나누어서 하는 것이 효과적이므로 알고리즘 분석을 통해서 프로그램의 partition 을 나누었다. 본 논문에서는 프로그램은 코딩작업의 효율성을 고려하여, 프레임 헤더정보 디코딩 부분은 C 함수로 cross-compile 을 수행하였으며, 나머지 두 부분은 프로그램 복잡도를 결정하는 중요한 모듈이기 때문에 어셈블리어로 프로그램 되어있으며[8], 프로그램된 각각의 모듈은 test 벡터들을 이용한 검증을 통해 확정되었다. 프레임 헤더정보 디코딩 부분은 그 내부의 모듈이 서로 유기적으로 연결되어있기 때문에 각 모듈을 먼저 compile 한 후, test 벡터를 통하여 검증하였다. 나머지 모듈은 각각을 프로그램한 후, 계속 연결시키면서 test 벡터를 검증하였다. Test 벡터는 Dolby 사에서 제공하는 40 개의 벡터 값을 사용하였으며, 구현된 프로그램은 test 벡터와 bit-exact 하게 일치 하였다. 이러한 과정을 통해 구현된 AC-3 복호화기의 평균 MIPS 를 표 1 에 나타내었다. 사용된 메모리량은 프로그램 ROM 23.3K Bytes, 데이터 ROM(table) 4.62K Bytes, RAM 45K Bytes 이다.

표 1. 구현된 AC-3 복호화기의
각 모듈별 MIPS(6 채널, 640ktps, 48KHz)

모듈	MIPS
헤더 프레임 분해	1.5
지수, 가수 역 양자화	46
합성 필터 뱅크	14
윈도링	5.4
합계	66.9

4. 결론

본 논문에서 사용한 ARM920T 칩은 저전력 소비형 32-비트 고정 소수점 RISC 구조로서 ARM9TDMI를 기반으로 MMU, 각각 16K Byte 명령어 Cache 와

Data Cache 를 사용하고 200MIPS 의 성능을 가지고 있다. 본 구현을 통해 AC-3 오디오 복호화기가 실시간 동작되도록 구현되었으며, test 벡터의 검증을 통하여 C 시뮬레이션 결과와 bit-exact 함을 확인 하였다.

구현된 AC-3 복호화기는 약 67 MIPS 의 복잡도를 나타내고 있으며 사용된 메모리량은 프로그램 ROM 23.3K Bytes, 데이터 ROM(table) 4.62K Bytes, RAM 45K Bytes 이다. 본 논문에서 구현된 AC-3 복호화기는 ARM920T 의 저전력 소비, ASIC 구현의 용이함, 편리한 디버깅 환경과 더불어 휴대형 CD 재생기, 카세트 재생기를 대체할 수 있는 차세대 휴대형 디지털 오디오 기기 등 다양한 응용에 부합하는 경쟁력 있는 시스템을 구축할 것이라 사료된다.

[참고 문헌]

- [1] Advanced Television Systems Committee, "Digital Audio Compression (AC-3) Standard", Doc. A/52, Nov, 92.
- [2] D. Pan, "A tutorial on mpeg/audio compression," IEEE Trans. on Multimedia, Vol.2, No.2, pp.64-74, 1995.
- [3] Yu-Chi Chen, Chien-Wu Tsai, and Ja-Ling Wu, "Fast time-frequency transform algorithm and their applications to real-time software implementation of Ac-3 audio codec," IEEE Transactions on Consumer Electronics, Vol. 44, No. 2, March 16, 1998.
- [4] Chi-Min Liu, Szu-Wei Lee, and Wen-Chieh Lee, " Bit allocation method for AC-3 Encoder," IEEE Transactions on Consumer Electronics, Vol. 44, No. 3, August, 1998.
- [5] Steve Vernon, " Design and implementation of AC-3 coders," IEEE Transactions on Consumer Electronics, Vol. 41, No. 3, August, 1995.
- [6] Marina Bosi, Grant Davidson, " High quality, low-rate audio transform coding for transmission and multimedia application," Audio Engineering Society at the 93rd convention, October, 1993.
- [7] Craig C. Todd, Grant A. Davidson, Mark F. Davis, " Flexible perceptual coding for audio transmission and storage," Audio Engineering Society at the 96th convention, February, 1994.
- [8] ARM, "ADS Assembler Guide," Revision 1.1, November 2000.