

# 휴대용 실시간 MP3 오디오 부호화기를 위한 하드웨어 가속기 설계

여창훈\*, 방경호\*, 이근섭\*, 박영철\*\*, 윤대희\*

\*연세대학교 미디어통신신호처리연구실

\*\*연세대학교 정보기술학부

## Design of Hardware Accelerator for Portable Real-time MP3 Audio Encoder

Changhoon Yeo\*, Kyoung Ho Bang\*, Keun-Sup Lee\*, Young Cheol Park\*\*, Dae Hee Youn\*

\*Media/Communication signal processing LAB., Yonsei University

\*\*The Division of Information Technology, Yonsei University

### Abstract

본 논문에서는 고정소수점 DSP로 구현한 실시간 MP3 오디오 부호화기에 사용되는 초월함수용 하드웨어 가속기 구조를 제안한다. 구현된 하드웨어 가속기는 MP3 부호화 성능을 저하시키는 초월함수 연산오차에 강인하도록 설계되었다. 제안된 가속기의 연산오차는 Q1.23 고정소수점 출력에서 2비트, 즉  $2^{-21}$  까지의 연산오차를 가진다. LAME 부호화기[5] 심리음향 모델의 SMR 오차는 테이블 보간법[4]을 사용할 경우에 비해 4dB 이상 향상되었으며, 연산량은 총 4 MIPS 감소하였다. 제안한 하드웨어 가속기는 Verilog HDL로 기술되었으며, SYNOPSIS에서 0.18 $\mu$ m CMOS 표준 셀 라이브러리 공정으로 합성되었다. 합성 면적은 7514 게이트이며 초월함수 연산에 대한 동작속도는 3 사이클이다.

위해서는 반복적인 초월함수의 연산이 필요하다. 이때 초월함수 연산 과정에서 큰 오차가 발생할 경우, 이 오차는 청각적으로 감지되어 부호화 성능을 감소시킨다. 따라서 부호화 성능을 향상시키기 위해서는 오차가 적은 초월함수 연산이 수행되어야 한다. 그러나 일반적인 고정소수점 DSP 구조에서 소프트웨어로 오차가 적은 초월함수 연산을 수행하기 위해서는 구현방법에 따라 하나의 칩으로 구현이 불가능할 정도의 과도한 기억장소가 필요하거나, 실시간 부호화가 불가능할 정도의 연산량이 필요하다.

본 논문에서는 이 문제를 해결하기 위해 초월함수 연산을 지원하는 하드웨어 가속기를 설계하였다. 가속기는 초월함수 연산 시 부호화 성능을 저하시키지 않는 오차 범위로 작동하며, 실시간 부호화에 적합하도록 빠른 동작속도를 가진다.

### I. 서론

MPEG 오디오 계층-3 (이하 MP3)은 휴대용 오디오 재생기, 인터넷 오디오 스트리밍 등에 널리 사용되고 있는 고품질 오디오 부호화 기술이다. MP3는 인간의 청각 특성을 고려한 심리음향 모델을 사용하며, 인간의 귀에 지각되지 않는 신호를 제거하여 압축률 10:1의 고품질 오디오 신호를 제공한다[1].

이와 같은 장점으로 인해 최근 많은 휴대용 MP3 복호화기가 고정소수점 DSP 구조로 구현되어 상품화되고 있다. 반면 고정소수점 DSP 구조를 이용한 휴대용 실시간 MP3 부호화기의 구현은 거의 이루어지고 않고 있으며, 그 원인 중 하나는 심리음향 모델 연산 과정의 초월함수 연산 오차이다.

MP3 부호화 과정의 심리음향 모델을 구현하기

### II. MPEG 오디오 심리음향 모델

MPEG 오디오의 압축 부호화 방식은 인간의 청각 모델링, 즉 심리음향 모델(Pschoacoustic Model)에 기반을 두고 있다. 심리음향 모델은 (1) 주파수 영역에 따라 인간의 가청 한계가 달라진다는 점과 (2) 주파수 영역에서 큰 신호가 인접한 작은 신호를 마스킹(masking)하여 인접한 작은 신호가 들리지 않는 점이며, 이 두 가지 특성을 이용하여 압축 부호화를 수행한다. 심리음향 모델의 연산 결과는 오디오 신호에 대한 마스킹의 비인 SMR(Signal to Mask Ratio)이다. MP3 부호화기는 식 (1)과 같이 SMR값과 SNR(Signal to Noise Ratio)값을 이용하여 MNR(Mask to Ratio)값을 구한 후, MNR 값을 이용하여 오디오 신호에 대한 비트를 할당한다[2]. 따라서 심리음향 모델의 오차로 인해 SMR이 정확하게 구해지지 않을 경우 비트 할당

과정에서 문제가 발생하게 되며 부호화 성능 저하가 발생하게 된다.

$$MNR(dB) = SNR(dB) - SMR(dB) \quad (1)$$

심리음향 모델의 연산 결과인 SMR은 일반 사칙 연산과 초월함수 연산의 과정으로 이루어진다[2]. 먼저 심리음향 모델의 비예측성(unpredictability, cw)를 구하기 위해서는 식 (2)의 제곱근 연산이 필요하며, 심리음향의 tonality, threshold와 같은 값을 구하기 위해서는 식 (3), (4)의 지수, 로그 연산이 필요하다. 따라서 정확한 SMR 값을 얻기 위해서는 정확한 초월함수 연산결과가 필요하다. 이 경우 일반적인 고정소수점 DSP 구조에서 초월함수는 하드웨어에서 지원되지 않으므로 소프트웨어로 구현되어야 한다.

$$cw = \frac{\sqrt{(r \cos f - \hat{r} \cos f)^2 + (r \sin f - \hat{r} \sin f)^2}}{r + abs(\hat{r})} \quad (2)$$

$$tbb = conv1 + conv2 \cdot \log(e^{ctb(b) / ecbb(b)}) \quad (3)$$

$$nbb(b) = ecbb(b) \cdot norm(b) \cdot 10^{SNR(b)/10} \quad (4)$$

고정소수점 DSP 구조에서 소프트웨어로 초월함수 연산을 수행하기 위해서는 테일러 급수, 또는 테이블에 의한 보간법을 사용할 수 있다. 테일러 급수를 이용할 경우 연산을 반복하여 높은 정밀도의 계산을 수행할 수 있으나, 많은 수의 곱셈과 덧셈을 요구하기 때문에 높은 연산량을 가지며 실시간 부호화에 사용하는 알고리즘으로는 적합하지 않다. 반면 테이블 검색법은 연산속도가 빠른 장점이 있지만 정확한 연산결과를 얻기 위해서는 매우 방대한 메모리가 필요하다. 만일 충분한 메모리를 사용하지 않을 경우 연산결과 오차가 커지는 단점이 있다. 따라서 심리음향 모델과 같이 높은 정확도가 요구되는 경우 초월함수는 고정소수점 DSP 구조에서 소프트웨어로 구현하는 것은 적합하지 않다[3].

따라서 본 논문에서는 적은 초월함수 연산오차를 가지며 실시간 부호화에 적합한 적은 연산량을 가지는 하드웨어 가속기를 제안하도록 한다.

### III. 초월함수 연산 알고리즘 및 하드웨어 가속기 구조

본 절에서는 먼저 초월함수 연산, 특히 로그연산과 지수 연산을 수행하는 알고리즘에 대해 설명하고 이 알고리즘을 수행하는 하드웨어 구조에 대해 설명한다.

#### 3.1 log 연산 알고리즘

MP3 부호화 과정에서 필요한 로그연산  $\log_{10} x$  은 식 (5)와 같이  $\log_2 x$ 에 관한 식으로 변환할 수 있다.

$$y = \log_2 x / \log_2 10 \cong 0.3010 \cdot \log_2 x \quad (5)$$

로그연산의 입력값  $x$ 는 식 (6)과 같이 지수  $2^n$ 와 실수  $\alpha$ 의 곱으로 표현 가능하다. 이 경우 지수  $2^n$ 은 하드웨어로 구현할 경우 쉬프트로 간단하게 구현할 수 있다. 따라서 로그 연산기의 입력은 실수  $\alpha$ 로 가정한다.

$$x = 2^{\frac{y}{0.3010}} = 2^n \cdot 2^f = 2^n \cdot \alpha \quad (6)$$

( $n$  = 정수,  $1 \leq \alpha < 2$ )

실수  $\alpha$ 에 대한 로그 연산은 아래 과정과 같다. 먼저 식 (7)과 같이 실수  $\alpha$ 를 1로 수렴시키는 수열  $\beta_n$ 을 정의한다. 이 경우 식 (8)와 같이 양변에 로그를 취할 경우 결과값은 0으로 수렴한다.

$$\alpha \cdot \beta_1 \cdot \beta_2 \dots \beta_n \cong 1 \quad (7)$$

$$\log_2 \alpha \cdot (\beta_1 \cdot \beta_2 \dots \beta_n) \cong \log_2 1 = 0 \quad (8)$$

따라서 식 (9), (10)과 같이  $\log_2 \alpha$ 의 값은 수열  $\beta_n$ 의 로그 값의 합과 같다[4].

$$\log_2 \alpha + \log_2 \beta_1 + \log_2 \beta_2 + \dots + \log_2 \beta_n \cong 0 \quad (9)$$

$$\log_2 \alpha \cong -(\log_2 \beta_1 + \log_2 \beta_2 + \dots + \log_2 \beta_n) \quad (10)$$

그러나 위 방법을 이용하여 하드웨어 가속기를 설계할 경우 아래와 같은 문제점이 발생한다. 먼저 식 (7)의 조건을 만족하는 수열  $\beta_n$ 의 값은 나눗셈 연산으로 구해진다. 그러나 나눗셈 연산은 일반적인 DSP 코어에서 지원되지 않으며, DSP에서 지원될 경우에도 매우 느리게 동작한다. 따라서 위 알고리즘을 하드웨어 가속기로 구현하기 위해서 곱셈기와 로그 테이블을 이용하여 로그 연산을 구현하는 알고리즘을 제안하였다.

먼저 식 (7)에서 실수  $\alpha(1 \leq \alpha < 2)$ 는 식 (11)과 같이 상수 1과 수열  $\alpha_n$ 의 조합으로 나타낼 수 있다.

$$\alpha = 1.001000\_110010\_010010 \dots (2)$$

$$\alpha_1 = 0.001000 (2)$$

$$\alpha_2 = 0.000000\_110010 (2)$$

$$\alpha_3 = 0.000000\_000000\_010010 (2) \dots \dots \quad (11)$$

다음으로 식 (12)를 만족하는  $\beta_1$  값을 메모리 테이블을 이용하여 구한다. 식 (12)의 메모리 테이블 구조는 그림 1과 같으며 수열값  $\alpha_1$ 를 메모리 주소로 사용한다. 이 때  $\beta_1$  값은 식 (13)에 의하여 항상 1보다

작거나 같은 실수이다.

$$(1 + \alpha_1) \cdot \beta_1 \cong 1 \tag{12}$$

$$\beta_1 \cong \frac{1}{(1 + \alpha_1)} \leq 1 \quad (1 + \alpha_1 \geq 1) \tag{13}$$

Addr.	$\beta_1$ Values
000000	0000000000...000
000001	0000000000...010
? ..	? ..
111111	1111111111...111

그림 1. 식 (12)를 위한 메모리 구조

식 (12)의 결과를 이용하여 식 (7)의 첫 번째 연산을 수행할 경우 연산 결과는 식 (14)과 같다. 위 식의 결과값을 새로운 수열  $\alpha_n'$ 로 새로 정의할 경우 식 (13)에 의해  $\alpha_1'$  값은 항상 0이 되는 것을 알 수 있다.

$$\alpha \cdot \beta_1 \cong (1 + \alpha_1) \cdot \beta_1 + \alpha_2 \cdot \beta_1 + \dots + \alpha_n \cdot \beta_1 \tag{14}$$

$$\cong 1 + \alpha_2' + \alpha_3' + \dots + \alpha_n'$$

$$(\alpha_2 \cdot \beta_1 = \alpha_2' \leq \alpha_2 \dots)$$

따라서 식 (11)부터 식 (14)의 과정을 n번 반복할 경우 식 (14)의 결과값은 1이 되며 이는 식 (7)와 같다. 따라서 식 (14)에 곱해진 수열  $\beta_n$ 을 이용하여 식 (10)의 연산이 가능하다. 아래 그림 2는 제안한 가속기 알고리즘의 하드웨어 구조로 n번의 연산 과정으로 로그 연산을 구현할 수 있다.

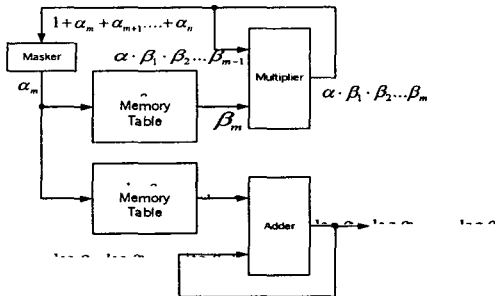


그림 2. 로그 연산을 위한 가속기 구조

### 3.2 지수 연산 알고리즘

부호화 과정에서 사용되는 지수연산  $10^{-x}$ 은 아래 식 (15)와 같이  $2^{-x}$  연산으로 구현할 수 있다.

$$10^{-x} \cong 2^{-\alpha}, \quad (\alpha \cong 0.3010 \times x) \tag{15}$$

$$\alpha = \alpha^1 + \alpha^2 + \dots + \alpha^n, \quad (0 \leq \alpha < 1) \tag{16}$$

위 식에서 입력값  $\alpha$ 를 식 (16)과 같이 수열  $\alpha^n$ 의 합으로 나타낼 경우  $2^{-x}$  연산은 식 (17)과 같이 반복된 곱셈 연산으로 나타낼 수 있다[4].

$$2^{-\alpha} = 2^{-\alpha^1} \cdot 2^{-\alpha^2} \dots 2^{-\alpha^n} \quad (0 \leq 2^{-\alpha^n} < 1) \tag{17}$$

따라서 수열  $\alpha^n$ 의 지수 테이블과 곱셈기를 이용하여 n번의 곱셈연산을 이용하여 지수 연산을 수행할 수 있다. 이 경우 식 (17)에서 곱셈에 사용되는 지수 테이블 값에 비례하여 곱셈기의 크기가 증가하게 된다.

일반적으로 연산기의 워드길이가 증가할 경우, 동작속도가 동일할 경우 가속기 면적이 증가하는 문제가 발생한다. 이런 문제점을 해결하기 위해 식 (17)는 식 (18)과 같이 MAC 연산으로 변환하여 사용한다.

$$2^{-\alpha} = 2^{-\alpha^1} \cdot (1 - \phi_2) \dots (1 - \phi_n) \tag{18}$$

$$\cong (2^{-\alpha^1} - 2^{-\alpha^1} \cdot \phi_2) \dots (1 - \phi_n)$$

$$(\phi_2 = 1 - 2^{-\alpha^2}, \dots, \phi_n = 1 - 2^{-\alpha^n})$$

이 경우  $\alpha^n$ 의 값이 0으로 수렴할 경우 지수 테이블은 1로 수렴하게 된다. 따라서  $\phi_n$  값은 0으로 수렴하게 되고 이 값을 이용하여 곱셈 연산을 수행할 경우 식 (17)에 비해 워드길이가 작은 곱셈기를 사용하여 지수 연산을 수행할 수 있다.

### 3.3 하드웨어 가속기 구조

3.1, 3.2 장에서 제안한 알고리즘을 적용한 하드웨어 가속기 구조는 그림 3과 같다. 24비트 고정소수점 DSP 코어와 독립적으로 작동하는 단일 MAC 구조로 가속기 구조를 설계하였다. 17비트의 곱셈기 레지스터와 24비트의 덧셈기 레지스터를 사용하여 중간 연산 결과를 저장하며 덧셈기에 2개의 버스, 곱셈기의 2개의 버스가 있으며 연산기에 들어가는 입력을 제어한다.

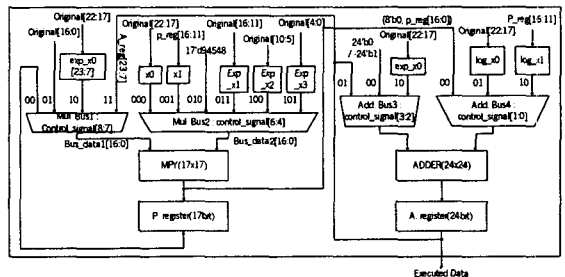


그림 3. 제안한 가속기의 블록 다이어그램

그림 2와 같이 로그 연산의 구현을 위해서 수열  $\beta_n$  과  $\log \beta_n$  의 테이블이 필요하며, 총 17비트, 64개의 데이터를 가지는  $x_m$  테이블 2개와 24비트, 64개의 데이터를 가지는  $\log x_m$  테이블 2개를 사용하였다.

또한 지수 연산을 위해 24비트, 64개의 데이터를 가지는  $2^{-n}$  테이블 1개를 사용하였으며, 또한 식 (18)의 연산을 위해 17비트, 64개의 데이터를 가지는  $\phi_m$  테이블 3개를 사용하였다.

#### IV. 부호화 성능 분석 및 하드웨어 가속기 합성 결과

본 논문에서 제안한 하드웨어 가속기 성능을 측정하기 위하여 아래와 같은 과정을 수행하였다.

- 제안한 가속기와 MATLAB의 부동소수점 연산 정확도 비교
- LAME 부호화기[5] 심리음향 모델에서 제안한 가속기와 테이블 보간법의 SMR 오차 비교
- 테이블 보간법과 제안 알고리즘의 연산량을 비교 후 LAME 부호화기[5] 심리음향 모델의 연산량 감소량 측정
- 제안 알고리즘을 구현한 하드웨어 가속기의 동작 속도 및 합성면적 측정

먼저 제안한 가속기의 오차를 측정하기 위해 MATLAB의 초월함수 연산값을 Q1.23의 고정소수점으로 변환하여 가속기 연산값과 비교하였다. 위 실험에서 연산 최대 오차는 2비트이며, 연산값은  $2^{-21}$  까지의 정확도를 가진다.

다음으로 LAME 부호화기[5]의 심리음향 모델을 이용하여 제안 기법과 테이블 보간법의 성능을 비교하였다. 그림 4는 심리음향 모델에서 제안 기법을 사용하여 얻은 SMR 오차와 테이블 보간법을 사용한 경우의 SMR 오차를 비교한 것이다.

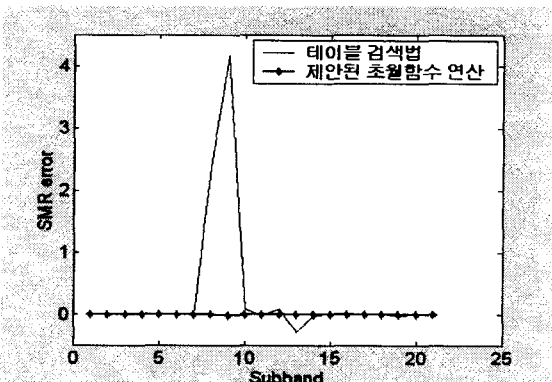


그림 4. 입력 오디오에 대한 SMR 에러 (44.1KHz, 48kbps, mono Encoding)

심리음향 모델의 SMR 오차가 -2dB-1dB 이내일 경우 음질 저하가 없는 심리음향 모델이라 할 수 있으며[6] 기존 테이블 검색법의 SMR 오차가 4dB이상인 경우에 비해 제안한 초월함수 연산 SMR 오차는 0dB에 가까우므로 기존 방법에 비해 음질 향상 효과가 있는 것을 알 수 있다.

다음으로 테이블 보간법과 제안한 알고리즘의 연산량을 비교하였다. 제안 가속기의 연산량은 3 사이클로 기존 테이블 검색기법의 25 사이클[3]에 비해 약 12%로 감소하였다. 위 가속기를 이용하여 LAME 부호화기[5]의 심리음향 모델을 구현할 경우 테이블 보간법을 사용할 경우에 비해 제공된 연산에서 약 3 MIPS의 연산량 감소와, 로그 / 지수 연산에서 1 MIPS의 연산량 감소로 총 4 MIPS의 연산량이 감소한다.

제안한 가속기는 verilog HDL로 기술되었으며, 가속기는 SYNOPSIS에서 0.18 $\mu$ m 표준 셀 라이브러리를 사용하여 게이트 수준까지 합성되었다. 동작 클럭은 10ns에서 3 사이클의 연산시간을 가지며 합성면적은 총 7514 게이트이다.

#### V. 결론

본 논문에서는 fixed-point DSP 로 구현한 실시간 MP3 오디오 부호화기에 사용되는 초월함수 하드웨어 가속기 구조를 설계하였다. 새로운 초월함수 알고리즘을 사용하여 연산 오차를 감소시켰으며 알고리즘에 최적화된 하드웨어를 설계하여 연산 속도를 증가시켰다. 구현된 하드웨어 가속기의 최대 오차는 2 비트이며 약 4dB 의 SMR 이득과 25%의 연산량 증가를 얻을 수 있었다. 0.18  $\mu$ m 공정에서 합성면적은 7514 게이트이며 10ns 의 클럭 속도에서 3 사이클의 연산시간이 소요된다.

#### 참고문헌

- [1] ISO/IEC 13818-3 "Generic Coding of Moving Pictures and Associated Audio (Part 3: MPEG-Audio)", 2nd Edition, Feb., 1997
- [2] ISO/IEC 11172-3 "Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbit/s (Part 3: MPEG-Audio)"
- [3] Keun-Sup Lee, et al, "High quality MPEG-audio layer III algorithm for a 16-bit DSP", Circuits and Systems, 2001. ISCAS 2001. The 2001 IEEE International Symposium on , Volume: 2 , 6-9 May 2001
- [4] ISRAEL KOREN, "Computer Arithmetic Algorithms", Prentice-Hall international
- [5] <http://www.mp3dev.org>
- [6] Christian Neubauer, et al, "A Compatible Family of Bitstream Watermarking Schemes for MPEG-Audio", Audio Engineering Society, 2001 May 12-15, Convention Paper 5346