論文 98-23-6-4

AC-3와 MPEG-2 오디오 공용 복호화기의 설계

正會員 고 우 석*, 유 선 국**, 박 성 욱*, 정 남 훈*, 김 준 석*, 이 근 섭*, 윤 대 희*

A Design of Dual AC-3 and MPEG-2 Audio Decoder

Woo-Suk Ko*, Sun-Kook Yoo**, Sung-Wook Park*, Nam-Hoon Jung*,
Joon-Seok Kim*, Keun-Sup Lee*, Dae-Hee Youn* *Regular Members*

요 약

본 논문에서는 실시간으로 AC-3와 MPEG-2 복호화 과정을 모두 수행할 수 있는 공용 복호화기를 구현하였다. 설계된 공용 복호화기는 연산량을 줄이고 두 알고리듬의 공통성을 얻기위해 FFT를 이용한 고속 변환 알고리듬을 사용한다. 그리고 각 복호화 과정을 효율적으로 수행하기 위해서 간단한 코덱용 DSP 프로세서와 공용 합성필터로 구성되었다. 프로세서는 각 복호화 과정의 제어중심부분을 수행하여 하드웨어의 복잡도를 줄이고 알고리듬의 확장성을 증가시키며, 공용 합성필터는 연산중심부분을 수행하여 규칙적인 연산의 수행을 고속화한다.

공용 복호화기는 VHDL로 상위수준 설계한 후 기능적 검증을 위해 Synopsys상에서 모의실험하였고, 0.6 μm 3ML 표준셀 기술로 합성한 후 Compass상에서 모의실험을 통해 최대 지연시간 상황에서 실시간 동작을 확인하였다.

ABSTRACT

The thesis presents a dual audio decoder which can decode both AC-3 and MPEG-2 bitstream. The MPEG-2 synthesis process is optimized via FFT to establish the common data path with AC-3's. A dual audio decoder consists of a DSP core which performs the control-intensive part of each algorithm and a common synthesis filter which performs the computation-intensive part.

All the components of the dual audio decoder have been described in VHDL and simulated with a SYNOPSYS tool. The software modeling of the DSP core was used for functional validation. After being synthesized using 0.6 μm-3ML technology standard cell, the dual audio decoder was simulated at gate-level with a COMPASS tool for hardware validation.

* 연세대학교 전자공학과
** 연세대학교 의용공학과
論文番號:98108-0312
接受日字:1998年 3月 12日

I. 서 론

디지탈 오디오는 기존의 아날로그 오디오에 비해 CD 수준의 고음질, 넓은 대역폭 그리고 큰 동적영역을 제공하는 우수한 특성을 지니고 있다. 그러나 아날로그 오디오 신호를 표본화(sampling)하여 고음질의 디지탈 오디오 신호를 얻게 되면 데이터량이 크게 증가하므로이것을 효과적으로 압축하는 방식에 대한 연구가 진행되어 왔다. 현재 세계적으로 널리 사용되고 있는 압축방식은 유럽의 MUSICAM 방식에 기반을 둔 MPEG-2 알고리듬[1]과 미국의 Dolby Lab.에서 제안한 AC-3 알고리듬[2]이다.

이와 같은 두가지 압축 알고리듬이 표준안으로서 공존하므로 이제까지는 주로 각 알고리듬의 복호화기에 대한 연구가 독립적으로 수행되어 왔다[5, 6]. 그러나 두 표준안이 공용되는 응용분야가 존재하므로 두 알고리듬을 모두 지원하는 공용 복호화기의 설계에 대한 관심이 증가하고 있다[7-9]. 효율적인 구조를 통해 소형화된 공용 복호화기는 공존하는 두 표준안을 하나의 시스템으로 지원하므로 비용절감의 잇점을 얻을 수 있다.

AC-3와 MPEG-2 복호화 과정을 분석해 보면 특성에 따라 공통적으로 제어중심부분과 연산중심부분으로 나뉘어질 수 있다. 제어중심부분은 비트열로부터 비트정보를 추출하여 부호화된 샘플을 복원하는 과정이며, 다양한 연산과 복잡한 흐름제어가 필요하다. 따라서 프로세서에 작업을 할당하여 하드웨어의 복잡도를 줄이고 확장성(flexibility)을 증가시키는 것이 바람직하다. 연산중심부분은 복원된 샘플을 PCM 샘플로 역변환하거나 합성하는 과정이며, 제어가 간단하고 규칙적인 연산을 많이 수행하는 부분이다. 따라서 작업을 하드와이어드 로직(hardwired logic)에 할당하여 복호화 과정수행을 고속화시키는 것이 바람직하다.

두 표준안의 연산중심부분은 서로 다른 구조를 가지므로, 공용 복호화기를 설계하기 위해서는 동일한 구조를 갖도록 하는 것이 핵심이 된다. 본 논문에서는 MPEG-2 합성 필터링 과정을 32-point FFT를 이용하는 고속 알고리듬으로 대체하여 AC-3의 고속 알고리듬과 동일한 연산구조를 갖도록 하였다.

본 논문에서는 간단한 코덱용 DSP 프로세서와 고속 공용 합성필터로 구성된 AC-3와 MPEG-2 오디오 공용 복호화기를 설계하였다. 공용 복호화기의 각 모듈은 VHDL을 이용하여 상위수준으로 설계되었다. 모델링을 사용하였고, 공용 합성필터의 기능적 검증 프로세서를 기능적으로 검증하기 위해 C 언어에 의한 모델링을 사용하였고, 공용 합성필터의 기능적 검증은 C 언어에 의한 부동/고정 소수점 모의실험을 통해 수행되었다. 마지막으로 설계된 공용 복호화기를 Compass Tool을 이용하여 0.6 μm-3ML 표준셀(standard cell) 기술로 합성한 후 최대 시간지연 상황에서도 정상적으로 동작함을 검증하였다.

본 논문의 구성은 다음과 같다. 2장에서 AC-3와 MPEG-2 알고리듬의 특징을 간단히 고찰하고, 3장에서는 각 복호화 과정을 효과적으로 수행할 복호화기의 구조를 제안하였다. 4장에서는 공용 복호화기의 구조를 제안하고 이를 설계하였으며, 5장에서는 설계된 공용 복호화기에 대한 검증을 통하여 성능평가를 수행하였다. 마지막으로 6장에서는 본 연구의 결론부분을 서술하고 추후 연구사항에 대하여 언급하였다.

II. 오디오 알고리듬 분석

2.1 AC-3 복호화 알고리듬

AC-3 오디오 복호화 알고리듬은 크게 나누어서 비트정보 추출과정, 지수부 복호화 과정, 심리음향 모델을 통한 비트할당 과정, 가수부 복호화 과정, 다채널 처리 과정, 역변환 과정 등으로 이루어진다. AC-3 오디오 복호화 과정을 그림 2-1에 나타내었다.

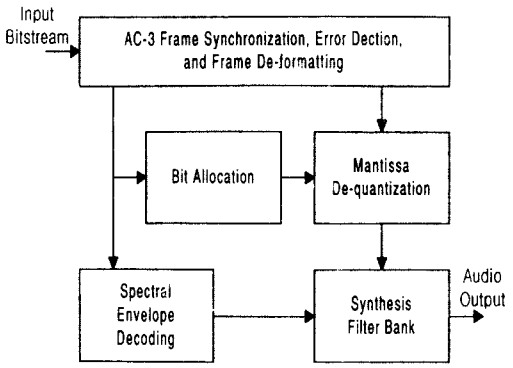


그림 2-1. AC-3 오디오 복호화 과정

그림 2-1의 비트정보 추출과정은 입력 비트열에 대해서 동기(synchronization)를 맞추고 에러를 검출(error

detection)한 후, 비트열에 대한 여러가지 정보를 추출해내는 과정이다. 이렇게 추출된 비트정보를 참조하여 지수부를 복호화(spectral envelope decoding)한다. 복호화된 지수부를 이용하여 비트할당(bit allocation) 정보를 계산한 후에, 가수부를 비트열로부터 추출하여 복원(mantissa de-quantization)한다. 마지막으로 복원된 지수부와 가수부를 모두 이용하여 합성 필터링 과정(synthesis filter bank)을 거치면 복호화된 오디오 신호를 얻을 수 있다.

AC-3 오디오 복호화 과정에 필요한 연산의 종류와 연산량 분석을 고려하면 복호화 과정이 두개의 서로 다른 성격을 지니는 부분으로 분류될 수 있음을 알 수 있다. 즉 비트 정보 추출 과정과 지수부 복호화 과정, 비트 할당 과정, 가수부 복호화 과정, 채널 디커플링 과정등은 연산량이 많지는 않지만 연산의 종류가 비교연산과 쉬프트, 표 참조 연산 등 다양한 연산을 포함하고, 조건 판단 등에 의해서 복호화 과정의 흐름이 제어되므로 제어가 매우 중요한 제어중심부분이다. 그러나 역변환 과정과 윈도우를 취하고 중첩 가산시키는 과정은 규칙적 연산을 반복적으로 많이 실행하여 연산량이 매우 많은 연산중심부분으로 간단한 제어만을 필요로 한다.

2.2 MPEG-2 복호화 알고리듬

MPEG-2 오디오 복호화 과정은 크게 비트정보 추출과정, 비트할당 정보 추출과정, 크기인자(scalefactor) 정보 추출과정, 샘플 역 양자화 과정(de-quantization), 샘플 역 정규화 과정(rescaling), 다채널 처리과정과 합성 필터링 과정으로 나누어진다. MPEG-2 오디오 복호화 과정을 그림 2-2에 나타내었다.

그림 2-2의 비트할당 복호화 과정(bit allocation decoding)은 입력 비트열로부터 비트할당 정보를 추출하여 복원하는 과정이며, 크기인자 복호화 과정(scalefactor decoding)은 비트열로부터 크기인자 정보를 추출하여 복원하는 과정이다. 복호화된 비트할당 정보를 이용하여 부호화된 샘플을 추출(requantization)해내며, 이미 복원된 크기인자를 참조하여 샘플의 크기를 역 정규화(rescaling)시킨다. 이렇게 복원된 샘플이 최종적으로 합성필터링(synthesis filtering) 과정을 거치면 출력 오디오 신호가 복호화된다.

복호화 과정에 대한 분석과 그에 따른 연산량 분석을 살펴보면, MPEG-2 오디오 복호화 과정도 AC-3

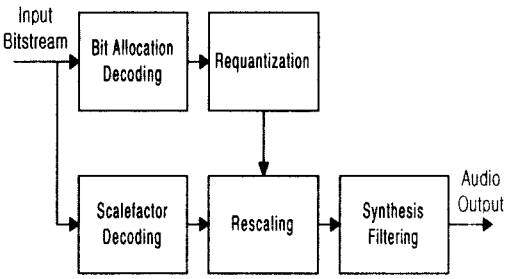


그림 2-2. MPEG-2 오디오 복호화 과정

오디오 복호화 과정과 마찬가지로 크게 두 종류로 분류할 수 있음을 알 수 있다. 비트정보 추출과정과 비트할당 정보 추출과정, 크기인자 정보 추출과정, 샘플 역 양자화 과정, 샘플 역 정규화 과정, 그리고 채널 매트리싱 과정과 다채널 처리 과정등은 다양한 연산과 조건판단 등에 근거한 복잡한 제어를 필요로 하는 제어중심부분이고, 합성 필터링 과정은 규칙적인 연산을 반복적으로 많이 수행하는 연산중심부분에 해당하며 간단한 제어만을 필요로 한다.

III. 공용 복호화기의 설계

3.1 알고리듬 수준 설계

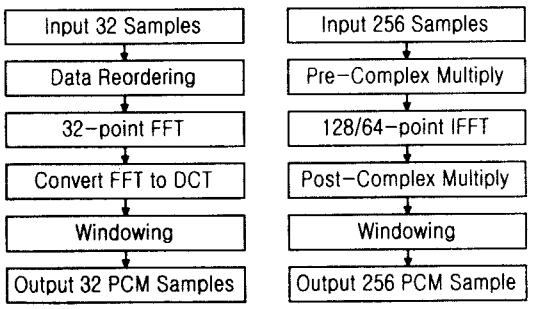
2장에서 수행한 AC-3와 MPEG-2 오디오 복호화 과정에 대한 분석을 살펴보면 두 복호화 과정이 공통적으로 제어중심부분과 연산중심부분으로 분류됨을 알 수 있다. AC-3와 MPEG-2 오디오 공용 복호화기의 설계라는 관점에서 각 복호화 과정의 제어중심부분과 연산중심부분을 비교해 보면 다음과 같다.

각 복호화 과정의 제어중심부분은 주로 비트열로부터 비트정보를 추출하고 이를 이용하여 부호화된 각 샘플을 복호화하는 과정으로 구성된다는 공통점을 가진다. 따라서 각 제어중심부분을 수행하는데 필요한 연산도 기본적인 산술연산과 논리연산을 비롯하여 최대값과 최소값을 찾는 비교연산, 특정수의 비트를 추출하는 연산 등이 공통적으로 필요하다.

각 복호화 과정의 연산중심부분을 살펴보면 곱셈과 덧셈으로 이루어지는 규칙적 연산을 반복적으로 많이 수행한다는 공통점을 가진다. 그러나 연산중심부분을 이루는 AC-3의 합성 필터링 과정은 고속 푸리에 변환과 복소수 곱셈으로 이루어지는 데이터 처

리과정으로 이루어진 반면, MPEG-2의 합성 필터링 과정은 MAC을 기본연산으로 하는 매트리싱 과정과 누적합산(accumulation) 과정으로 이루어져 있다. 따라서 공용 복호화기를 설계하기 위해서는 서로 다른 특성을 갖는 이러한 연산중심부분을 서로 공통점을 갖는 알고리듬으로 재구성해야만 한다.

본 논문에서는 128/64-point IFFT을 이용한 AC-3 고속 역변환 과정과 동일한 연산구조를 갖도록 MPEG-2 합성필터링 과정을 32-point FFT를 이용한 고속 알고리듬으로 대체하였다. 고속 알고리듬을 구성하기 위해 먼저 MPEG-2 합성 필터링 과정을 Konstantinos가 제안한 DCT를 이용한 고속 알고리듬으로 대체하였다[3]. 다음으로 DCT 과정을 Narasimha와 Peterson이 제안한 FFT를 이용한 고속 알고리듬[4]으로 대체함으로써 최종적인 MPEG-2 고속 합성 필터링 과정을 구성할 수 있다. 동일한 연산구조를 갖도록 변경된 AC-3와 MPEG-2 합성 필터링 과정의 고속 알고리듬을 그림 3-1에 나타내었다.



(a) AC-3 (b) MPEG-2

그림 3-1. 고속 합성필터링 알고리듬

MPEG-2 합성 필터링 과정을 그림 3-1의 고속 알고리듬으로 대체하면 다음과 같은 잇점을 얻는다. 첫째, FFT를 사용하므로 IFFT를 사용하는 AC-3 고속 역변환 알고리듬과 동일한 연산구조를 갖게 되어 공용 합성필터를 설계에 직접 적용할 수 있다. 둘째, 고속 알고리듬이므로 기존의 매트리싱 과정에 필요한 연산량을 약 1/4정도로 감소시킨다. 마지막으로 그림 3-1에 보여진 AC-3와 MPEG-2 고속 알고리듬이 공통적으로 선처리 과정과 후처리 과정을 수반하는 FFT/IFFT 과정으로 구성되므로 유사한 연산흐름을 갖게 되어 공용 합성필터의 설계가 용이해진다.

3.2 아키텍쳐 수준 설계

3.2.1 공용 복호화기의 구조

효율적인 공용 복호화기를 구성하려면 2장에서 분석한 복호화 과정중에서 제어중심부분을 프로세서에 할당하고 연산중심부분을 하드와이어드 로직(hardwired logic)에 할당하는 것이 바람직하다.

다양한 연산과 복잡한 흐름(flow)에 대한 제어가 필요한 제어중심부분을 프로세서에 할당하여 프로그램으로 복호화 과정을 구현하면 각 복호화 과정을 위한 별도의 하드웨어를 설계하지 않아도 되므로, 하드웨어의 복잡도를 줄이면서 제어를 효과적으로 수행할 수 있다. 또한 복호화 알고리듬이 갱신되어도 기존의 프로그램만 수정하면 새로운 알고리듬을 쉽게 구현할 수 있으므로 확장성(flexibility)을 확보할 수 있다.

반면에 규칙적인 연산을 반복적으로 많이 수행하는 연산중심부분은 제어가 간단하므로 하드와이어드 로직에 할당하면 연산을 고속으로 수행하도록 설계할 수 있어 전체 복호화 시간을 감소시킬 수 있다. 또한 스테이트 머신(state machine)으로 제어로직을 구현하면 면적을 감소시키면서 효과적인 제어를 수행할 수 있다. 이와 같은 분석에 근거하여 본 논문에서 제안한 효율적인 공용 오디오 복호화기의 구조를 그림 3-2에 나타내었다.

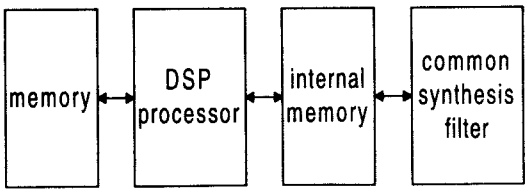


그림 3-2. 효율적인 공용 복호화기의 구성

그림 3-2에서 공용 복호화기를 구성하는데 사용된 프로세서는 오디오 복호화기의 목적으로 설계된 RISC (Reduced Instruction Set Computer) 형태의 DSP 프로세서로서 3단계 파이프라인 기법을 사용하고 하버드 구조(Harvard architecture)를 취하고 있다. 프로세서의 명령어의 구성은 DSP 연산에 필요한 기본적인 산술연산 및 논리연산 명령어들과 MAC, SHIFT등을 포함하며, 특히 오디오 복호화 과정에 적합하도록 최대값 또는 최소값을 찾는 비교명령어와 비트열로부터 특정수의 비트를 추출해내는 UNPACK

이란 명령어를 포함한다.

그림 3-2의 공용 합성필터는 그림 3-1의 AC-3와 MPEG-2 고속 합성 필터링 과정을 효율적으로 수행하는 구조를 갖도록 다음과 같은 조건을 만족한다. 첫째, 두 고속 알고리듬의 연산량중 대부분을 차지하는 FFT/IFFT를 빠르게 수행하는 구조이다. 둘째, FFT/IFFT 외에도 선처리, 후처리 과정 연산의 대부분을 이루는 복소수 곱셈을 빠르게 수행하는 구조이다. 마지막으로 MPEG-2 고속 합성 필터링 과정중에 발생하는 데이터 이동을 최소화하고, 데이터의 주소를 효율적으로 발생시키도록 모듈로 어드레싱(modulo addressing)과 인덱스 어드레싱(index addres- sing)을 지원한다.

내부 메모리(internal memory)의 구조는 실수부와 허수부로 나뉘어진 구조로서, 복소수 연산 수행시 복소 데이터에 대한 참조를 빠르게 수행하기 위한 것이다. 또한 실수부, 허수부 각각의 메모리는 쌍(pair)으로 존재하여 입력 복소 데이터를 읽고, 계산된 출력 복소 데이터를 메모리에 쓸때 메모리에 대한 충돌이 일어나지 않도록 하였다.

3.2.2 공용 합성필터의 설계

본 논문에서 설계된 공용 합성필터는 그림 3-1의 두 고속 알고리듬을 효과적으로 수행하는 구조를 갖도록 3.2.1절에서 고찰한 세가지 조건을 만족하도록 설계되었으며, 연산수행 로직인 데이터 프로세서의 내부구조는 그림 3-3과 같다.

그림 3-3을 보면 설계된 데이터 프로세서는 3.2.1절에서 고찰한 두번째 조건에 따라 복소수 곱셈 연산을 빠르게 수행할 수 있도록 2개의 곱셈기와 1개의 덧셈기를 직렬로 연결한 구조를 갖는 MMA(Multiply-Multiply-Add) 로직을 두었다. 두 복소수 x_1과 x_2을 곱셈한 결과 y는 다음과 같다.

$$Re(y) = Re(x_1) * Re(x_2) - Im(x_1) * Im(x_2)$$
$$Im(y) = Re(x_1) * Im(x_2) + Im(x_1) * Re(x_2) \quad (3\text{-}1)$$

식 (4-1)에서 보듯이 결과값 y의 실수부와 허수부를 얻기 위해서는 각각 2번의 실수곱셈과 1번의 실수 덧셈 또는 뺄셈연산이 필요하다. MMA 로직은 이러한 실수연산들을 각각 2개의 곱셈기와 1개의 덧셈기에 할당하여 경로 (1)을 사용해 2번의 실수곱셈과 1번의 실수덧셈 연산이 클럭에 의한 시간지연(clock delay) 없이 수행되도록 한다. 결과적으로 복소수 곱셈 결과의 실수부 또는 허수부를 계산하는데 1 클럭 사이클밖에 소요하지 않으며 이것은 3.2.1 절에서 살펴본 두번째 조건을 만족시키기에 충분한 속도이다.

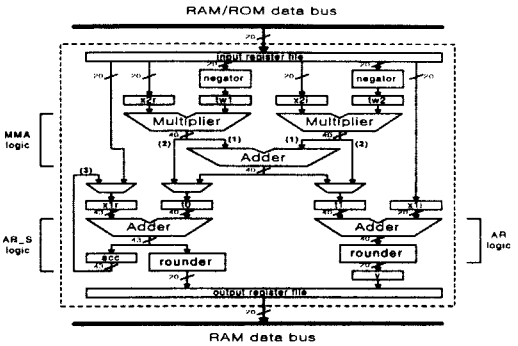


그림 3-3. 설계된 데이터 프로세서의 내부구조

데이터 프로세서 내 MMA 로직은 AC-3 합성 필터링 과정의 역 푸리에 변환 과정(IFFT), 선처리 과정(pre-complex multiply step), 후처리 과정(post-complex multiply step)과 MPEG-2 합성 필터링 과정의 푸리에 변환 과정(FFT), 후처리 과정(converting FFT to DCT) 등 복소수 곱셈이 주된 연산을 이루는 과정들을 고속으로 수행할 수 있도록 한다.

MMA 로직을 살펴보면 곱셈기의 출력이 덧셈기의 입력으로 들어가지 않는 경로 (2)가 존재하는데 이것은 AC-3의 윈도우를 취하는 과정을 효과적으로 수행하기 위한 것이다. 즉 실수곱셈 연산만을 필요로 하는 윈도우를 취하는 과정에서 덧셈기를 거치지 않은 두 곱셈기의 결과는 2개의 샘플에 2개의 윈도우를 동시에 취하는 셈이 되어 윈도우 과정(windowing)을 효율적으로 수행하도록 한다.

MPEG-2의 경우는 윈도우를 취하는 과정과 32개의 PCM 샘플을 얻는 과정을 동시에 수행하는데 MMA 로직은 경로 (2)를 사용하여 이 과정들을 매우 효율적으로 처리한다. 즉 2개의 곱셈기를 이용하여 2개의 샘플에 대해 동시에 2개의 윈도우를 취한 후 덧셈기를 이용하여 각 결과값들의 부분합을 구하며, 이렇게 구해진 부분합은 최종적으로 PCM 샘플을 얻기 위한 누적합산값에 더해진다.

설계된 데이터 프로세서는 또한 첫번째 조건에 따라 버터플라이(butterfly) 연산을 빠르게 수행하기 위

해서 덧셈기와 반올림기(rounder)를 직렬로 구성한 구조를 가지는 2개의 AR(Add-Round) 로직을 가지고 있다. 두개의 복소수 입력 x_1, x_2와 트위들 계수(twiddle factor) u에 대하여 행해지는 버터플라이 연산의 출력 y_1, y_2는 다음과 같다.

$$Re(y) = Re(x_1) * Re(x_2) - Im(x_1) * Im(x_2)$$
$$Im(y) = Re(x_1) * Im(x_2) + Im(x_1) * Re(x_2) \quad (3\text{-}2)$$

식 (4-2)에서 보듯이 버터플라이 연산 출력의 실수부와 허수부는 각각 복소수 x_2와 u의 곱셈 결과의 실수부 또는 허수부에 또다른 입력 x_1의 실수부 또는 허수부를 더하거나 빼서 얻는다. 이때 복소수 곱셈 결과는 앞서 설명한 MMA 로직에서 수행하므로 출력 y_1 또는 y_2를 얻기 위해서는 실수부와 허수부에 각각 1번의 덧셈 또는 뺄셈이 추가로 필요하게 된다. 2개의 AR 로직은 이러한 추가적인 덧셈 연산을 수행하며, 버터플라이 연산을 파이프라이닝(pipelining) 형식으로 처리할 경우 연산을 고속화시킨다. 결과적으로 1번의 버터플라이 연산 출력 y_1, y_2을 계산하는데 총 2 클럭 사이클이 소요되며, 이것은 첫번째 조건을 만족시키기에 충분한 속도이다.

AR 로직중 AR_S(Add-Round-Sum) 로직은 덧셈기의 출력을 다시 덧셈기의 입력에 연결시키는 귀환경로(feedback path) (3)을 가지는데 이것은 MPEG-2에서 최종적인 PCM 샘플을 얻기위해 누적합산하는 과정을 위한 것이다. 즉 MMA 로직을 통해 2개의 샘플에 윈도우를 취하고 결과를 더한 부분합을 다음번 부분합과 누적합산하기 위한 것으로, 누적합산한 결과는 누적레지스터(accumulator)에 저장된다. 이때 1개의 PCM 샘플을 얻기 위해서는 16번의 누적합산, 즉 8번의 부분합에 대한 누적합산이 필요하므로, 누적레지스터의 길이는 가드 비트(guard bits)를 포함해서 43 비트의 길이가 된다.

AR 로직의 반올림기(rounder)는 최종적인 40 비트 결과값을 20 비트 메모리에 저장하기 위해 반올림할 때 발생하는 반올림 오차(rounding error)를 최소화하기 위한 것이다. 또한 메모리 버스로부터 입력 레지스터로의 경로에 있는 부호 반전기(negator)는 AC-3의 IFFT 과정과 MPEG-2의 FFT 과정을 모두 지원하기 위한 것으로, 두 변환쌍(transform pair)은 트위들 계수의 부호만 다르고 연산구조는 똑같기 때문이다. 따라서 부호 반전기는 ROM으로부터 트위들 계수를 읽어온 후 모드에 따라서 부호를 반전시킨다.

3.2.3 DSP 프로세서 설계

프로세서 코어는 파이프라인 방식으로 명령어를 처리하며, 하버드 구조를 사용하여 파이프라인 운영의 효율을 높였다. 또한 소스 연산자(source operand)로 레지스터는 물론 메모리 값을 사용하며, 결과 연산자(destination operand)는 레지스터만을 사용하는 메모리-레지스터 머신(memory-register machine)이다. 즉, 소스 연산자에 데이터 메모리의 어드레스를 지정하여 이 어드레스의 값을 읽어 연산하여 레지스터에 저장한다. 메모리에 값을 적기 위해서는 WRITE라는 명령을 통해서만 가능하다. 명령어 처리과정은 프리페치(prefetch), 디코딩/리드(decoding/read), 실행(execution)의 3단계로 분할된다.

명령어 프리페치 단계에서는 프로그램 메모리로부터 명령어를 읽어 온다. 디코딩/리드 단계에서는 프리페치된 명령어를 디코딩하여 실행단계에 필요한 모든 제어 신호를 생성하며, 데이터 메모리 값이 필요한 경우 메모리로부터 읽어온다. 마지막으로 실행 단계에서는 지정된 연산을 수행하거나, 메모리에 값을 저장하는 WRITE 명령어의 경우 메모리에 값을 저장한다. 각 파이프라인 단계는 모두 1 사이클만에 수행된다.

3단계 파이프라인의 각 단계를 독립적으로 실행하기 위해서 각각의 유닛이 존재한다. 먼저 프리페치단계를 실행하기 위한 프로그램 시퀀서(program sequencer)가 있으며, 디코딩/리드 단계를 수행하기 위한 제어 및 주소발생기(control & data address generator) 유닛이 있고, 마지막으로 실행단을 처리하기 위한 ALU가 있다.

하버드 구조를 채택하였으므로 두 개의 메모리, 즉 프로그램 메모리와 데이터 메모리가 독립적으로 존재한다. 이들 메모리의 어드레스는 14비트로 설정하였다. 프로그램 메모리는 ROM만을 사용하도록 설계하였으므로 값을 읽기만 할 수 있다. 데이터 메모리는 크게 내부메모리와 외부 메모리로 구분된다. 내부 메모리는 ROM과 RAM으로 구성되는데 각각 2k씩을 할당하였다. 외부메모리는 총 14비트의 어드레스 레

인지중 내부메모리인 롬2k, 램 2k를 제외한 나머지 영역을 사용하도록 되어있다.

그림 3-4에 설계된 DSP 프로세서의 내부구조를 나타내었으며 다음과 같은 연산을 수행한다. 첫째, 산술연산으로서 기본적으로 덧셈, 뺄셈, 곱셈을 수행한다. 이 연산기는 양수 및 음수 데이터를 취급하므로 절대값을 구하는 연산도 수행하며, 곱셈기와 덧셈기를 직렬로 연결하여 MAC연산을 수행할수 있다. 둘째, 비트(bit) 조작과 시스템제어 신호를 만들기 위한 AND, OR, NOT의 논리연산과 SHIFT 연산을 수행한다. 마지막으로, 오디오 디코딩을 위한 특별한 연산이다. 설계된 프로세서 코어는 AC-3, MPEG2 복호화기를 구현하기 위해 설계되었으므로 일반적인 DSP Core에는 존재하지 않는 특수한 연산들을 지원한다. 먼저, 입력 비트열로부터 특정 비트수만큼을 추출해내는 UNPACK 이란 연산을 지원하며, 두 개의 값을 비교하여 큰값, 혹은 작은 값을 유지하도록 하는 비교연산과 두 개의 절대값을 비교하여 큰값, 혹은 작은값을 찾도록 하는 연산등을 지원한다.

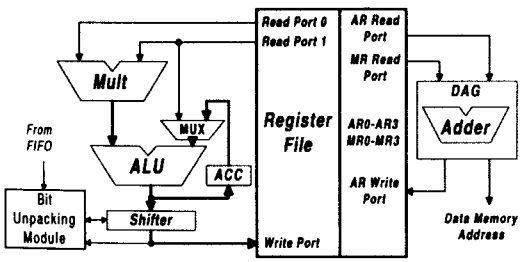


그림 3-4. 설계된 DSP 프로세서의 내부구조

IV. 결 과

4.1 기능적 검증

본 논문에서 설계된 공용 복호화기에 대한 기능적 검증은 세 단계로 수행되었다. 첫째로 상위수준 프로그래밍 언어(high-level language)인 C 언어로 부동 소수점 및 고정 소수점 모의실험을 수행하여 공용 복호화기에서 채택한 각 알고리듬에 대한 기능적 검증을 수행하였다. 두번째로 HDL(Hardware Description Language) 언어인 VHDL로 프로세서와 공용 합성필터를 설계하여 Synopsys Tool상에서 각각에 대해 상위수준에서 기능적 검증을 수행하였다. 마지막으로 상위수준에서 설계한 프로세서와 공용 합성필터를 0.6 μm-3ML 표준셀 기술(technology)을 이용해 합성한 후 Compass Tool상에서 모의실험을 수행하여 게이트(gate)에 의한 시간지연 등을 고려한 기능적 검증을 수행하였다.

그림 4-1과 그림 4-2는 설계된 공용 복호화기의 기능적 검증 결과를 나타내며, C 언어에 의한 모의실험 결과값과 compass상에서 합성한 공용 복호화기에 대해 모의실험을 수행한 결과값을 비교한 것이다. 그림

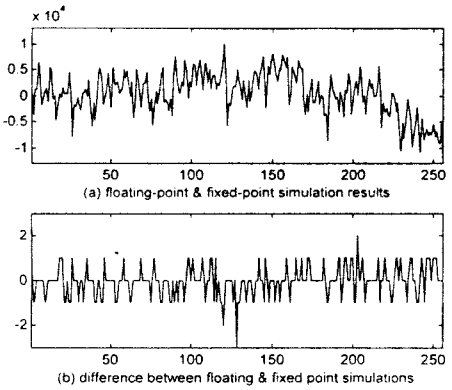


그림 4-1. AC-3 복호화 과정에 대한 부동/고정 소수점 모의실험

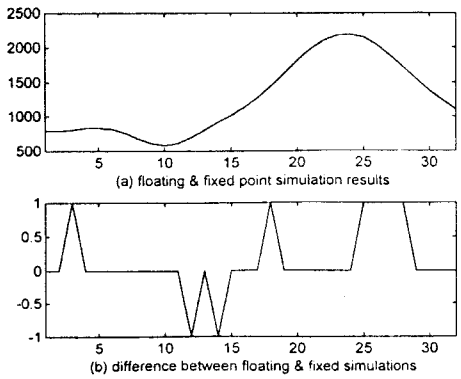


그림 4-2. MPEG-2 복호화 과정에 대한 부동/고정 소수점 모의실험

4-1과 4-2는 각각 AC-3와 MPEG-2에 대한 실험결과를 나타내며, 각각의 (a)는 두 모의실험 결과를, (b)는 두 결과값의 차이를 나타낸 것이다. 그림에서 보듯이 AC-3의 경우는 두 결과값이 최대 3정

도의 차이밖에 나지 않으며, MPEG-2의 경우는 최대 2정도의 차이밖에 나지 않아 서로가 거의 일치함을 알 수 있다. 따라서 설계된 공용 복호화기는 두 표준안 모두에 대해서 정상적으로 복호화 과정을 수행함을 알 수 있다.

4.2 실시간 구현 검증

4.2.1 AC-3 복호화 과정

최대 48 KHz 표본화 주파수(sampling frequ- ency)와 20MHz의 동작 주파수를 고려하면, AC-3 비트열(bitstream)의 1 프레임 데이터를 처리하는 시간은 20M*256*6/48,000=640,000 사이클이 된다.

모의실험을 수행한 결과 프로세서가 제어중심 부분을 수행하는데 소요한 사이클 수는 표 4-1과 같고, 공용 합성필터가 연산중심 부분을 수행하는데 소요한 사이클 수는 표 4-2와 같은 결과를 얻었다. 표 4-1과 4-2를 보면 복호화 과정을 수행하는데 소요한 총 사이클 수는 340,197+55,050=395,247이므로 할당된 640,000 사이클보다 작아 실시간으로 수행할 수 있음을 확인하였다.

표 4-1. 프로세서가 소요하는 사이클 수(비트율=448 Kbps)

복호화 과정	소요된 사이클 수
비트정보 추출과정	75
지수부 복호화 과정	25134
비트 할당 과정	169038
가수부 복호화 과정	137460
채널 디커플링 과정	8490
합 계	340197

표 4-2. 공용 합성필터가 소요하는 사이클 수

복호화 과정	소요된 사이클 수
pre-multiply step	7680
IFFT step	27990
post-multiply step	7680
window and overlap/add	11700
합 계	55050

4.2.2 MPEG-2 복호화 과정

최대 48 KHz 표본화 주파수(sampling frequency)와 20MHz의 동작 주파수를 고려하면, MPEG-2 비트열(bitstream)의 1 프레임 데이터를 처리하는 시간은 20M*1,152/48,000=480,000 사이클이 된다.

모의실험을 수행한 결과 프로세서가 제어중심 부분을 수행하는데 사용한 사이클 수는 표 4-3와 같고, 공용 합성필터가 연산중심 부분을 수행하는데 사용한 사이클 수는 표 4-4과 같은 결과를 얻을 수 있었다. 표 4-3와 4-4에서 보듯이 복호화 과정을 수행하도록 할당된 사이클 수는 480,000인 반면, 실제로 복호화 과정을 수행는데 소요된 총 사이클 수는 207,539+89,100=296,639이므로 설계된 공용 복호화기는 실시간으로 MPEG-2 복호화 과정을 수행할 수 있음을 검증하였다.

표 4-3. 프로세서가 소요하는 사이클 수(비트율=128 Kbps)

복호화 과정	소요 사이클 수
MPEG-1 비트정보 추출과정	3903
MPEG-1 샘플 역양자화 과정	51630
MPEG-2 비트정보 추출과정	20963
MPEG-2 샘플 역양자화 과정	77475
리매트리싱 과정	53568
합 계	207539

표 4-4. 공용 합성필터가 소요하는 사이클 수

복호화 과정	소요 사이클 수
32-point FFT step	33660
converting FFT to DCT step	6840
window and decimate/add step	48600
합 계	89100

4.3 물리적 검증

본 논문에서 설계된 공용 복호화기의 물리적 평가는 합성된 프로세서와 공용 합성필터에 대해 게이트에 의한 시간지연(gate delay)에 대한 분석과 게이트 수에 의한 시스템 평가로 행하였다.

물리적 평가중 게이트에 의한 시간지연은 플립플롭(flip-flop)의 출력에서 다른 플립플롭에의 입력까지 걸리는 시간으로 공용 복호화기의 최대 동작주파수를 결정하는 중요한 파라미터이다. 설계된 프로세서와

공용 합성필터의 최대 시간지연(maximum delay)은 각각 28.29 ns, 24.98 ns이므로 20 MHz의 동작 주파수에서 공용 복호화기가 정상적으로 동작함을 알 수 있다. 또한 프로세서와 공용 합성필터를 모두 고려할 때, 최대 지연 시간이 28.29 ns 이므로 30%의 여유(margin)을 고려하면 설계된 공용 복호화기의 최대 동작주파수가 약 27 MHz가 됨을 알 수 있다.

게이트 수에 의한 시스템 평가는 설계된 시스템의 전체 면적을 측정하기 위한 것으로 시스템의 경제성을 평가하는 중요한 척도중의 하나이다. 본 논문에서 구현되어 합성된 프로세서와 공용 합성필터의 게이트 수는 각각 22,890, 21,541이며, 비교적 적은 시스템 비용으로 AC-3와 MPEG-2 오디오 표준안을 모두 효율적으로 지원함을 알 수 있다. 설계된 공용 복호화기는 기존의 독립적인 AC-3 또는 MPEG-2 오디오 복호화기[5,6]보다 그다지 크지 않은 면적을 차지하면서 두 알고리듬을 모두 지원한다. 또한 공용 합성필터를 사용하지 않은 공용 복호화기보다는 약 2/3 정도로 면적을 감소시킬 수 있다.

V. 결 론

본 논문에서는 AC-3와 MPEG-2 오디오 복호화 과정을 모두 지원하는 공용 복호화기를 구현하였다. 이를 위해서 MPEG-2의 매트리싱 과정을 FFT를 이용한 고속 알고리듬으로 대체하여 IFFT를 이용하는 AC-3 고속 역변환 과정과 동일한 연산구조를 갖도록 하였다. 또한 설계된 공용 복호화기는 각 복호화 과정을 효율적으로 수행할 수 있도록 간단한 CODEC용 DSP 프로세서와 고속 공용 합성필터로 구성하였다.

설계된 프로세서는 각 복호화 과정의 제어중심부분을 효과적으로 수행할 수 있도록 기본적인 산술연산과 논리연산을 비롯해서 MAC, 비교연산, 비트추출 연산 등과 같은 특별한 명령어를 가진다. 이러한 명령어는 3 단계 파이프라이닝을 통해 효율적으로 수행되며, 하버드 구조를 사용하여 파이프라이닝의 효율을 높였다. 공용 합성필터는 각 복호화 과정의 연산중심부분을 빠르게 수행할 수 있도록 2개의 곱셈기와 1개의 덧셈기를 직렬로 연결한 MMA(Multiply-Multiply-Add) 로직과 덧셈기와 반올림기를 직렬로 연결한 AR (Add-Round) 로직으로 구성하였다.

제안한 공용 복호화기에 대한 설계와 검증은 다음과 같이 수행하였다. 우선 C 언어 프로그래밍에 의한 부동/고정 소수점 모의실험을 통해 대체된 알고리듬에 대한 기능적 검증을 수행하였다. 프로세서와 공용 합성필터는 VHDL로 상위수준 설계한 후 Synopsys Tool상에서 모의실험을 수행하였다. 마지막으로 VHDL로 설계된 각 모듈을 0.6 μm-3ML 표준셀 기술로 합성한 후 Compass Tool상에서의 모의실험을 통해 최대 지연시간 상황에서도 정상적으로 동작함을 검증하였다.

본 논문에서 설계한 공용 복호화기는 AC-3와 MPEG-2 오디오 압축 알고리듬을 사용하는 모든 분야에서 사용될 수 있다. 따라서 디지탈 오디오 방식의 저장매체나 디지탈 오디오 방송 등 두 표준안이 공용되는 분야에서 단일 시스템으로 두 복호화 과정을 모두 지원하므로 복호화기를 구성하는데 필요한 비용을 줄일 수 있다.

참 고 문 헌

1. 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)"
2. Advanced Television Systems Committee (ATSC) Standard Doc. A/52, "Digital Audio Compression Standard(AC-3)", Nov. 1994
3. K. Konstantinides, "Fast Subband Filtering in MPEG Audio Coding", IEE Signal Processing, Letters vol.1, no.2, pp.26-28, 1994
4. M. J. Narasimha and A. Peterson, "On the Computation of the Discrete Cosine Transform", IEEE Trans. Commun., vol. COM-26, pp. 934-946, Jun. 1978
5. S. Venon, "Design and Implementation of AC-3 Decoders", IEEE Trans. on Consumer Electronics, vol. 41, No. 3, Aug. 1995
6. S. -C. Han, et al. "An ASIC Implementation of the MPEG-2 Audio Decoder." IEEE Trans. on Consumer Electronics," pp.540-545, Aug. 1996.
7. 고우석, 유선국, 박성욱, 정남훈, 김준석, 이근섭, 윤대희, "AC-3와 MPEG-2 오디오 디코더 공용 합

성필터의 설계", 신호처리 합동 학술대회, pp. 299-302, Sep., 1997
8. Winnie Lau, Alex Chwu, "A Common Transform Engine for MPEG & AC3 Audio Decoder", IEEE Trans. on Consumer Electronics, Vol. 43, No. 3, pp. 559-566, Aug. 1997
9. Y. Jhung and S. Park, "Architecture of Dual Mode Audio Filter for AC-3 and MPEG", IEEE Trans. on Consumer Electronics, Vol. 43, No. 3, pp. 575-585, Aug. 1997

고 우 석(Woo-Suk Ko) 정회원
1973년 1월 2일생
1996년:연세대학교 전자공학과(공학사)
1998년:연세대학교 전자공학과(공학석사)
1998년~현재:연세대학교 전자공학과 박사과정
※주관심분야:오디오 신호처리, VLSI 신호처리 등
e-mail:xlover@radav.yousei.ac.kr

유 선 국(Sun-Kook Yoo) 정회원
1959년 1월 2일생
1981년:연세대학교 전기공학과(공학사)
1985년:연세대학교 전기공학과(공학사)
1989년:연세대학교 의용공학과(공학박사)
1989년~1995:순천향대학교 조교수
1995년~현재:연세대학교 의용공학과 조교수
※주관심분야:생체신호처리, 고속정보 시스템, VLSI 신호처리 등

박 성 욱(Sung-Wook Park) 정회원
1971년 3월 27일생
1993년:연세대학교 전자공학과(공학사)
1995년:연세대학교 전자공학과(공학석사)
1995년~현재:연세대학교 전자공학과 박사과정
※주관심분야:오디오 신호처리, VLSI 신호처리, 설계 자동화

정 남 훈(Nam-Hoon Jeong) 정회원
1973년 12월 7일생
1995년:연세대학교 전자공학과(공학사)
1997년:연세대학교 전자공학과(공학석사)
1997년~현재:연세대학교 전자공학과 박사과정
주관심분야:디지털 신호처리, 오디오 신호처리, VLSI 신호처리 등

김 준 석(Joon-Seok Kim) 정회원
1972년 10월 23일생
1995년:연세대학교 전자공학과(공학사)
1997년:연세대학교 전자공학과(공학석사)
1997년~현재:연세대학교 전자공학과 박사과정
※주관심분야:ASIP, 오디오 신호처리 등

이 근 섭(Keun-Sup Lee) 정회원
1975년 2월 1일생
1997년:연세대학교 전자공학과(공학사)
1997년~현재:연세대학교 전자공학과 석사과정
※주관심분야:오디오 신호처리, VLSI 신호처리 등

윤 대 희(Dae-Hee Youn) 정회원
1951년 5월 25일생
1977년:연세대학교 전기공학과(공학사)
1979년:미국 Kansas 주립대학교 전기공학과(공학석사)
1982년:미국 Kansas 주립대학교 전기공학과(공학박사)
1982년~1985년:미국 Kansas 주립대학교 조교수
1985년~현재:연세대학교 전자공학과 정교수
※주관심분야:적응 신호처리, 음성/오디오 신호처리, 레이다/소나 신호처리, VLSI 신호처리