

# MPEG 오디오 복호기용 하이브리드 필터의 VHDL 설계 및 C 언어 인터페이스에 의한 기능 검증

## (VHDL Design of Hybrid Filter Bank for MPEG Audio Decoder and Verification using C-to-VHDL Interface)

鞠一鎬\*, 朴鍾鎭\*\*, 朴洵台\*\*\*, 趙源敬\*\*\*

(Il-Ho Kook, Jong-Jin Park, Won-Tae Park, and Won-Kyung Cho)

### 요 약

반도체 공정 기술의 발달은 기하 급수적인 집적도의 증가를 가져오고, 이는 한 칩에 시스템을 모두 집적시키는 시스템 온 칩(SoC : System on Chip) 설계가 가능해지고, 이에 따른 설계 방법의 변화를 요구하고 있다. Soc 설계는 시스템에서 설계 사양(Specification)의 정의가 중요한 요소가 되고 있다. 본 논문에서는 MPEG 오디오 복호기에서 사용되는 IMDCT를 시스템 수준의 실행 가능한 설계 사양(Executable Specification)에 의해 설계하였다.

### Abstract

Silicon semiconductor technology agrees that the number of transistors on a chip will keep growing exponentially, and it is pushing technology toward the System-On-Chip. In SoC Design, Specification at system level is key of success. Executable Specification reduces verification time. This Paper describes the design of IMDCT for MPEG Audio Decoder employing system-level design methodology and Executable Specification Methodology in the VHDL simulator with FLI environment.

### I. 서 론

디지털 오디오의 압축은 낮은 전송 비트 율에서 높은 음질을 유지하기 위하여 필수적인 기술이다. 이와

같은 디지털 오디오 압축은 MPEG(Moving Picture Expert Group) 규정 안으로 표준화 되어 높은 압축 율을 얻을 수 있으면서도 고 음질의 유지가 가능하여 저장 매체에 제약이 있는 휴대 장비나, 낮은 전송 율에서도 높은 음질이 요구되는 분야에 널리 응용되고 있다.<sup>[1][2][3][4]</sup> MPEG 3계층 오디오 압축은 1과 2계층에 비하여 MDCT(Modified Discrete Cosine Transform) 알고리즘을 추가하여 압축 및 음질이 향상된 반면 연산의 복잡도가 상당히 증가하여 복호기를 구성하기 위해서 높은 성능의 하드웨어를 필요로 한다.<sup>[5]</sup> MPEG 오디오 복호화 과정은 허프만 복호화(Huffman Decoding) 역 양자화(Inverse Quantization), IMDCT, 그리고 서브밴드 합성 필터(Subband Synthesis Filter)의 과정을 거쳐 PCM 데이터를 출력한다. MPEG 오디오 3

\* 正會員, 株式會社 앤스랩  
(AnsLab Co., Ltd.)

\*\* 正會員, 大林大學 電子情報通信科  
(Dept. of Electronic Information Communication,  
Daelim College)

\*\*\* 正會員, 慶熙大學校 電子工學科  
(Dept. of Electronic Engineering, Kyunghee Univ.)  
接受日:2000年9月30日, 수정완료일:2000年11月30日

계층 복호화 알고리즘에서 연산의 복잡도가 높은 IMDCT는 하드웨어 설계 시 처리 과정이 복잡하여 기능 검증에 어려움이 많다.<sup>[6][7]</sup>

MPEG 오디오 복호기에서 IMDCT 연산은 가변 길이 코드(Variable Length Code) 알고리즘인 허프만 디코딩, 비선형 역 양자화를 거친 출력과 부가 정보(Side-Information)를 처리 후에 이루어지므로 임의의 테스트 값을 이용한 기능 검증이 어렵다. 더구나, MPEG 오디오 규정의 다양한 형식의 입력 데이터를 대상으로 한 검증이 필요하므로 이를 위해서 방대한 양의 테스트 벡터를 필요로 하게 되어 검증에 많은 시간이 소요된다.<sup>[7]</sup> 이에 따라 매우 복잡한 제어 구조와 높은 수준의 연산 구조를 갖는 알고리즘의 하드웨어 설계 과정과 이의 검증에 필요한 시간적인 비용을 최소화 할 수 있는 설계 방식이 요구되며 이에 대한 연구도 활발하게 이루어지고 있다.<sup>[8][9]</sup> 최근 반도체 기술의 발달로 인하여 단위 면적 당 집적도가 급격히 증가하고 있으나, 설계의 복잡도가 증가하는 반면 더욱 단축된 Time-To-Market의 요구 등으로 인하여 Re-Usable Design, IP(Intellectual Property) Based Design 이라고 하는 새로운 개념의 설계 방법과 원활한 Team Design이 강조되고 있다.<sup>[11]</sup>

특히 설계 전 단계에서부터 Specification의 중요성과 이를 기반으로 하는 Spec-Based Design 과 검증은 성공적인 설계의 요소가 되고 있다.<sup>[9]</sup> 예전의 문서화된 "Document Specification"보다는 테스트 벤치에 바로 사용할 수 있도록 준비된 실행 가능한 설계 사양을 사용함으로써 기능 검증 시간을 단축할 수 있을 뿐만 아니라 좀더 실제에 접근한 테스트 벡터를 사용하게 됨으로써 설계물의 높은 신뢰도를 보장 할 수 있게 된다. 본 논문에서는 MPEG 3계층 오디오 복호기에서 사용될 수 있는 IMDCT 프로세서를 VHDL을 이용하여 설계하였으며 기능 검증을 위하여 C언어에 의한 HDL 인터페이스 기법을 사용함으로써 설계의 정확성을 높이고 검증에 필요한 시간적인 비용을 최소화 하였다. C언어를 이용하여 16비트 범위의 고정 소수점 연산기를 갖는 MPEG 오디오 복호기 전체를 모델링하고 설계된 IMDCT의 기능 검증을 하였다. 기능 검증에 사용된 테스트 벡터는 MPEG 1과 2의 규정에 따라 각각 3 종류의 전송 비트 율 및 8종류의 샘플 주파수 형식으로 부호화된 MPEG 오디오 3계층(MP3) 파일을 이용하였다. C언어 모델을 HDL 시뮬레이터에 인터페이스 하는 방법

으로 PLI(Procedural)와 FLI(Foreign Language Interface)가 있다. 본 논문에서는 VHDL로 설계된 IMDCT의 기능 검증에 FLI와 16비트 고정 소수점 연산 C모델을 이용하여 효과적인 설계 및 검증하였다.

## II. IMDCT 알고리즘

MPEG 오디오 3계층 알고리즘은 기본적으로 16비트 PCM 오디오 데이터를 Poly phase Filter에 의하여 32개로 주파수 영역으로 분할하며 이를 각각 MDCT하여 18개의 Spectral 계수로 표현 한다. MPEG 오디오 3계층에서 MDCT를 채용함으로써 32개의 서브밴드 계수 표현에서 576개의 spectral 주파수 분해능으로 개선하였으며, 50% 시간 영역 중첩된 윈도우를 사용하여 시간 영역의 왜곡을 줄이도록 한다. MPEG 오디오 3계층의 복호화 알고리즘은 그림 1과 같다. 부호화된 오디오 입력 신호는 허프만 디코딩과 역 양자화 되며, 32개 서브밴드사이의 천이 영역 보상을 위한 Anti-aliasing butterfly를 거쳐 IMDCT로 입력된다. 각 서브밴드 당 18개의 spectral 계수들은 IMDCT에 의하여 32 서브밴드 합성 필터의 입력 값으로 계산되어진다.

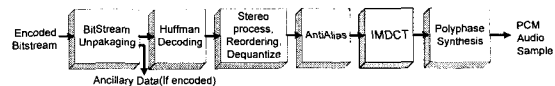


그림 1. MPEG 오디오 3계층 복호기의 처리 과정  
Fig. 1. MPEG Audio Layer II Decoder.

IMDCT 계산식은 다음 식(1)과 같다.

$$x_i = \sum_{k=0}^{\frac{n}{2}-1} X_k \cos\left(\frac{\pi}{2n}\left(2i+1+\frac{n}{2}\right)(2k+1)\right) \quad \text{for } i=0 \text{ to } n-1 \quad (1)$$

Anti-aliasing된 각 서브밴드의 18개 샘플은 IMDCT를 거쳐 36개의 샘플이 얻어지는데 이는 시간 영역에서 50%중첩된 윈도우를 사용하였기 때문이다. Windowing 계산은 식 (2)와 같다.

$$z_i = x_i \sin\left(\frac{\pi}{36}\left(i+\frac{1}{2}\right)\right) \quad \text{for } i=0 \text{ to } 35 \quad (2)$$

Windowing된 36개 데이터는 시간 상의 중복을 포함하므로 연속적인 블록 사이에 누적되어진다. 시간적 중복에 따른 누적 과정은 식 (3)과 같다.

$$result_i = z_i + z_{i+18} \text{ for } i=0 \text{ to } 17 \quad (3)$$

본 논문에서 설계한 IMDCT 프로세서는 식 (1)의 IMDCT, 식 (2)의 Windowing, 그리고 식 (3)의 중복 누산기를 포함하였다. 그림 2는 설계한 IMDCT의 프로세서의 구성을 보여준다.

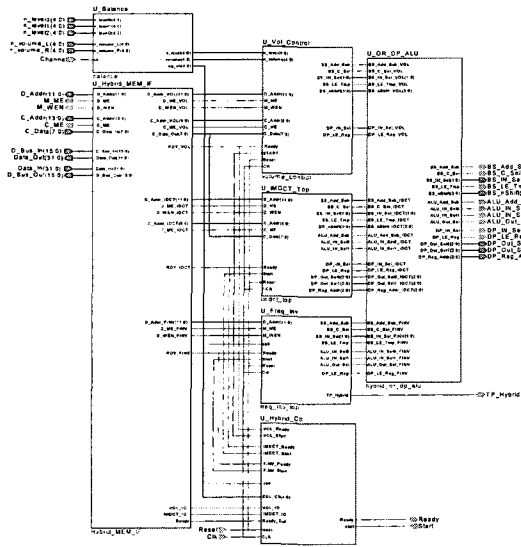


그림 2. 설계한 IMDCT 프로세서의 구성  
Fig. 2. Structure of Designed IMDCT Processor.

### III. IMDCT 프로세서 설계

실수 연산이 포함되어 있는 알고리즘을 하드웨어로 설계하는 과정에서 첫 단계는 C와 같은 고 수준의 프로그래밍 언어를 이용하여 사용 가능한 범위내의 고정 소수점 연산 모델링이 이루어져야 한다. 이와 같은 C 모델은 하드웨어 설계 시 필요한 연산 및 제어기 등의 구조를 예측할 수 있고, 설계된 회로의 기능 검증에 사용할 수 있다. IMDCT를 표현하는 식(1), (2)는 실수 계수 승산기와 가산기의 설계가 필요함을 알 수 있다. IMDCT 연산에 필요한 고정 소수점 연산기의 유효 숫자 범위를 정하기 위하여 부동 소수점 연산 알고리즘을 개발하고 그 결과를 비교 검토하였다. MPEG 오디오 복호기는 IMDCT 이전에 가변 길이 코드 알고리즘인 허프만 디코딩과 비선형 역 양자화 과정, 그리고 Anti-aliasing을 필요로 한다. 이러한 알고리즘은 산술

연산의 복잡도는 낮으나 부호기에 주어지는 부가 조건에 따라 매우 다양한 결과를 출력하게 되므로 임의의 테스트 값으로 고정 소수점 연산 모델링된 IMDCT의 검증이 곤란해진다.

따라서 본 논문에서는 MPEG 규정 안을 기준으로 하여 MPEG 오디오 복호기 시스템의 알고리즘 동작 모델(Behavioral Model)을 C언어로 구현하고 설계할 IMDCT 부분을 분할하여 고정 소수점 C 모델을 알고리즘으로 수정하였다. 고정 소수점 알고리즘은 C언어의 함수 형태로 작성되므로 쉽게 MPEG 오디오 복호기 시스템에 링크 되어 시스템 모델의 실행에 의하여 검증된다. 이와 같이 설계할 하드웨어의 사양을 실행 가능(Executable Specification)한 형태로 제공함으로써 방대한 테스트 벡터의 자동 생성이 가능하고 실제 오디오 데이터(Real-Stimuli)의 사용에 의한 하드웨어 설계의 신뢰도를 높일 수 있게 된다.

본 논문에서는 실행 가능한 설계 사양의 정의(Executable Specification)를 기반으로 하여 IMDCT를 VHDL로 설계하였다. MPEG 오디오 복호화에 사용되는 IMDCT 처리 과정은 18개의 입력 값을 갖는 IMDCT를 32회 반복하는 것으로 이루어진다. 각각의 IMDCT는 크게 변환(Transform) 처리 부분과 윈도우(Window) 처리, 그리고 시간 중복 누산 처리(Overlapping and Addition)부분으로 나눌 수 있는데,

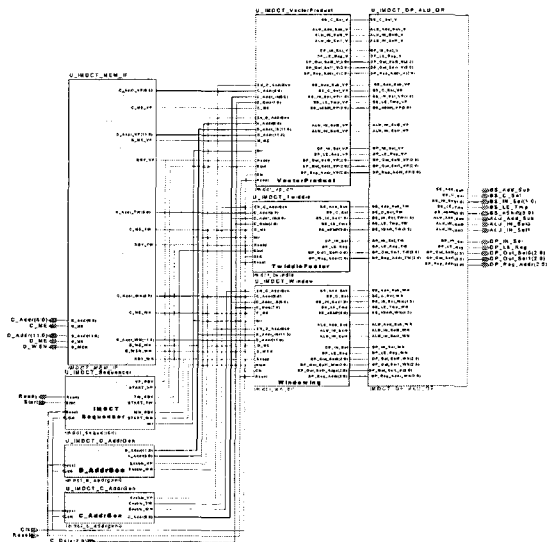


그림 3. IMDCT 세부 블록  
Fig. 3. IMDCT Block Diagram.

이들 연산 절차가 서로 상이하기 때문에 각각의 처리 부분에 대한 내부 제어기는 독립적으로 설계하였고, 메인 제어기가 세 부분을 제어하도록 계층적인 제어 구조로 구현하였다. 실행 가능한 설계 사양(Executable Specification)에서 IMDCT에 필요한 연산기는 16비트 고정 소수점으로 결정되었고 각각의 계수들은 ROM에 저장하여 각 부분별 제어기에 의한 주소 지정에 따라 곱셈이 이루어진다. 설계한 연산기의 승산은 4개 이내의 유효 비트를 갖는 SD(Signed-Digit)방식으로 표현된 계수들을 이용하여 Barrel-Shifter 및 가산기에 의하여 이루어진다. 계수들은 64개의 IMDCT 계수와 512개의 Window 계수가 사용되었다. 그림 3은 설계된 IMDCT의 세부 블록이다.

IV. Executable Specification을 이용한 기능 검증

실행 가능한 설계 사양(Executable Specification)의 가장 큰 장점은 시스템 수준에서 고도의 추상화 수준을 갖는 설계 사양(specification)을 기능 검증(Functional Verification)에 사용할 수 있다는 점이다. 즉 시스템의 부분 블록을 시스템 수준에서 검증할 수 있다는 것이다. 본 논문에서는 실행 가능한 설계 사양을 테스트 벤치(Test Bench)에서 바로 사용할 수 있도록 하기 위해서 FLI 환경에서 구현하였다. 그림 4는 FLI를 이용한 IMDCT 기능 검증 환경을 보여준다.

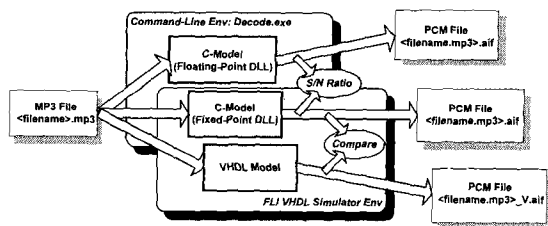


그림 4. FLI Test 환경  
Fig. 4. FLI Test Environment

FLI는 순차 수행 언어인 C언어와 병렬 처리를 기본으로 하는 VHDL 시뮬레이터 사이에서 매개체 역할을 하여 C언어로 기술된 실행 가능한 설계 사양과 VHDL 모델을 연동시켜 모의 실험하는 방법을 제공한다<sup>[5]</sup>. 그림 5는 FLI 환경에서 VHDL 모델과 C모델에서 서로 데이터를 주고받기 위한 포트 정합(Port Mapping)을

보여준다.

entity fl_iMDCT is port( filename : in string(1 to 80);  Reset : in STD_LOGIC Clk : in STD_LOGIC;  Glbl_Gain : out STD_LOGIC_VECTOR(7 downto 0); Prelflag : out STD_LOGIC Smpl_Freq : out STD_LOGIC_VECTOR(1 downto 0); Scfsc_Scl : out STD_LOGIC; Scfsc_Cmpr : out STD_LOGIC Version : out STD_LOGIC Stereo : out STD_LOGIC; Block_Type : out STD_LOGIC_VECTOR(1 downto 0); Mode : out STD_LOGIC_VECTOR(1 downto 0); Mode_Ext : out STD_LOGIC_VECTOR(1 downto 0);  IMDCT_Rd : out STD_LOGIC IMDCT_St : in STD_LOGIC  D_Addr : in STD_LOGIC_VECTOR(11 downto 0); D_Bus : in STD_LOGIC_VECTOR(15 downto 0); D_ME : in STD_LOGIC D_WEN : in STD_LOGIC ); end fl_iMDCT;	typedef struct { signalID filename;  signalID Reset; signalID Clk;  driverID Glbl_Gain; driverID Prelflag; driverID Smpl_Freq; driverID Scfsc_Scl; driverID Scfsc_Cmpr; driverID Version; driverID Stereo; driverID Block_Type; driverID Mode; driverID Mode_Ext;  driverID IMDCT_Rd; signalID IMDCT_St;  signalID D_Addr; signalID D_Bus; signalID D_ME; signalID D_WEN; } inst_rec;
--	--

그림 5. VHDL과 C모델의 Port Mapping  
Fig. 5. Port Mapping between VHDL and C.

MPEG 오디오 복호기의 테스트 벤치 구성은 그림 5의 오른쪽과 같은 엔터티(Entity)를 가지는 블록을 선언하고 아키텍처(Architecture)에는 C모델의 경로를 기술한다. 공용 객체(Shared Object)로 구현된 C모델에서는 왼쪽과 같은 스트럭처(Structure)를 선언해주면 FLI 환경에서 서로 데이터를 주고받을 수 있다. 그림 4와 같은 FLI 환경에서의 기능 검증은 많은 장점들을 제공한다. 첫 번째 장점은 실제 환경에서 사용하는 데이터(Real World Stimuli)를 이용할 수 있다. MPEG 오디오 복호기 전체를 기술한 실행 가능한 설계 사양을 기능 검증에 사용함으로써 IMDCT 이전 블록까지 복호화된 오디오 데이터(MP3)를 테스트 벡터(Test Vector)로 사용하여 설계된 IMDCT를 모의 실험할 수 있고, 처리 결과를 실행 가능한 설계 사양의 결과와 비교하여 오류를 찾아낼 수 있다. 또한 VHDL 모의 실험 결과를 실행 가능한 설계 사양(Executable Specification)에서 IMDCT 연산 후 블록까지 처리하여 완전히 복호화된 PCM 데이터를 얻을 수 있으므로 설계된 IMDCT를 이용해서 복호화된 오디오를 귀로 청취할 수 있다. 두 번째 장점은 테스트 벡터(Test Vector)를 생성하는 시간을 절약할 수 있다. 설계된 IMDCT는 MPEG I, II의 모든 표본화 주파수(Sampling Frequency)(6가지), 비트 율(bitrate)(14가지)을 지원한다. 따라서 기존의 테스트 환경에서는 모든 경우를 테스트하기 위해서는 6×14=84가지의 테스트 벡터를 파일로 만들

어야 한다. 여기에 스테레오 모드가 4가지 이상 존재하므로 300개가 넘는 test vector를 만들어야 하는 번거로움이 있다. 또한 부호화된 오디오 데이터가 프레임으로 구분되어 있고, 한 프레임이 2개의 granule로 나뉘어진다. 1초 정도의 오디오에 40개 정도의 프레임이 존재하기 때문에 80개(granule 단위로)의 test vector를 만들어야 하므로 전체적으로  $300 \times 80 = 24000$ 개가 넘는 test vector 파일을 만들어야 하므로 많은 노력과 시간이 소모된다. 하지만 FLI 환경에서 C언어로 기술한 실행 가능한 설계 사양을 이용하는 기능 검증 방법에서는 부호화된 오디오 데이터를 읽어 그래놀(Graule) 단위로 테스트 벡터를 넘겨줄 수 있기 때문에 테스트 벡터 생성 시간도 절약할 수 있다. 표 1과 2는 기능 검증에 사용된 오디오 데이터 형식의 종류를 나타내었다.

표 1. 검증에 사용된 MPEG Audio 압축 형식(MPEG1)

Table 1. Compressed MPEG Audio Layer III format for testing(MPEG1).

Bs \ Fs	32	40	48	56	64	80	96	112	128	160	192	224	256	320
32			M	M	M			J	J	J	J			
44.1			M	M	M		J	J	J	J/S	J/S	S	S	
48			M	M	M		J	J	J	J	J		S	

M : mono, J:joint-stereo, S:stereo

표 2. 검증에 사용된 MPEG Audio 압축 형식(MPEG2)

Table 2. Compressed MPEG Audio Layer III format for testing(MPEG2).

Bs \ Fs	16	24	32	40	48	56	64	80	96	112	128	144	160
16	M	M	J/M	J	J								
22.05	M	M	M/S	J	J	J	J	S					
24		M	M	J	J	J	J						

M : mono, J:joint-stereo, S:stereo

세 번째 장점은 MPEG 오디오 복호기 설계 시 기능 검증에 소모되는 시간을 절약할 수 있다. 설계 초기 단계에서 다른 부 블록과의 인터페이스 등을 고려하여 기술한 시스템 수준의 실행 가능한 설계 사양에 그대로 사용함으로써 시스템 수준에서의 기능 검증을 할 수 있다. 따라서 MPEG 오디오 복호기 설계에서는 설계된 IMDCT를 다른 부 블록과의 통합(Integration) 작업만 수행하면 된다. 실행 가능한 설계 사양을 이용한

설계 방법의 또 다른 장점은 부 블록의 설계를 동시에 수행할 수 있다는 것이다. MPEG 오디오 복호기의 각 부 블록이 순차적으로 수행되지만, 모든 부 블록을 시스템 수준에서 기술한 실행 가능한 설계 사양에 의해 기능 검증 할 수 있으므로 이전 블록의 설계에 관계없이 다음 블록의 설계도 동시에 할 수 있다. 그림 6은 MPEG 오디오 복호기 설계 시 기능 검증에 사용할 수 있는 통합 모의 실험 환경을 보여준다.

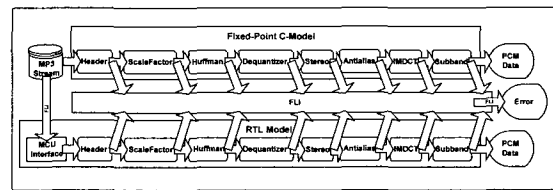


그림 6. MPEG Audio Decoder 통합 simulation 환경  
Fig. 6. Integrated Simulation Environment of MPEG Audio Decoder.

모든 부 블록의 C모델의 결과와 VHDL 모델의 모의 실험 결과를 FLI 환경에서 바로 비교할 수 있으므로 전체 시스템 통합에서 어느 블록에서 오류가 발생하는지 쉽게 찾을 수 있고 기능 검증에 소모되는 시간도 줄일 수 있다.

### V. 결론

반도체 공정 기술의 발달은 기하 급수적인 집적도의 증가를 가져오고, 이는 한 칩에 시스템을 모두 통합시키는 시스템 온 칩 설계로 설계 방법이 변화하고 있다. 시스템 온 칩 설계는 시스템 수준에서의 알고리즘의 설계 사양 정의가 시스템 설계의 중요한 요소가 되고 있다. 본 논문에서는 MPEG 오디오 복호기에서 사용되는 IMDCT를 시스템 수준의 실행 가능한 설계 사양에 의해 설계하였다. C언어를 이용하여 MPEG 오디오 복호기의 실행 가능한 설계 사양을 기술하였고, FLI을 이용하여 IMDCT의 기능 검증에 실행 가능한 설계 사양을 그대로 이용하였다. 이 방법은 실제 오디오 데이터를 이용할 수 있고, 테스트 벡터 파일을 생성하는데 소모되는 시간과 수고를 덜어주며, MPEG 오디오 복호기 설계 시 기능 검증에 소비되는 시간을 단축시켜준다. 모든 블록들이 시스템 수준으로 기술된 실행 가능한 설계 사양을 이용하여 설계하였다. 이와 같은 기능 검

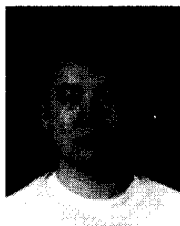
증 후 최종 시스템 설계에서는 각 블록들을 신뢰할 수 있으므로 통합 작업만 하면 된다. 시스템 통합 후, 전체 시스템의 기능 검증에도 실행 가능한 설계 사양을 사용할 수 있으므로 빠른 시간 내에 신뢰성 있는 설계를 가능하게 해 준다. 향후 과제는 시스템 수준의 C모델을 바탕으로 MPEG 오디오 3계층 복호기 설계에 이용될 수 있도록 하여야 하며 시스템 인터페이스에 대한 기준을 마련하여 신뢰도 있는 재사용 가능한 IP(Intellectual Property)가 되도록 할 예정이다.

참 고 문 헌

[1] R. K. Jurgen, "Broadcasting with Digital Audio", IEEE Spectrum, pp. 52-59, Mar. 1996.  
 [2] United States Advanced Television Systems Committee(ATSC), Audio Specialist Group (T3/S7) Doc. A/52, "Digital Audio Compression Standard (AC-3)", Nov. 1994.  
 [3] ISO/IEC JTC1/SC29/WG11 No.71, Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 M bit/s Part 3 : Audio(ISO/IEC 11172-3), Mar. 1993.

[4] K. Brandenburg et al., "ISO-MPEG-1 Audio: A Generic Standard for Coding of High-Quality Digital Audio", J. Audio Eng. Soc., pp. 780-792, Oct. 1994.  
 [5] ISO/IEC JTC1/SC29/WG11 No.803, Generic Coding of Moving Pictures and Associated Audio : Audio(ISO/IEC 13818-3), Nov. 1994.  
 [6] M. J. Narashima and A. Peterson, "On the computation of Discrete Cosine Transform", IEEE trans. Comm., vol. COM-26, pp. 934-946, Jun., 1978  
 [7] Winnie Lau and 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.  
 [8] D. Gajski, F. Vahid, S. Narayan, J. Gong, Specification and Design of Embedded Systems, New Jersey, Prentice Hall, 1994.  
 [9] A. Gerstlauer, S. Zhao, D. Gajski, A. Horak, Design of a GSM Vocoder using SpecC Methodology, SpecC Technical report.  
 [10] ModelSim User's Manual, ModelTech.  
 [11] Synopsys, CoWare, SystemC User's Guide ver. 1.0, 2000.

저 자 소 개



韓一鎬(正會員)  
 1963년 11월 21일생. 1987년 2월 경희대학교 물리학과 이학사. 1989년 2월 경희대학교 전자공학과 공학석사. 1995년 2월 경희대학교 전자공학과 공학박사 수료. 1999년 5월~현재 주식회사 앤스랩 대표이사. 주 관심분야 : 멀티미디어 통신, 휴대용 단말기(PDA), ASIC 설계

趙源敬(正會員) 第 29卷 第 2號 參照



朴鍾鎭(正會員)  
 1966년 12월 17일생. 1989년 2월 경희대학교 전자공학과 공학사. 1991년 8월 경희대학교 전자공학과 공학석사. 1995년 2월 경희대학교 전자공학과 공학박사 수료. 1999년 5월~현재 주식회사 앤스랩 ASIC 설계 교육 팀장. 주 관심분야 : 디지털 신호 처리 시스템, 통신 시스템, ASIC 설계

朴洵台(正會員)  
 1972년 7월 27일생. 1999년 2월 경희대학교 전자공학과 공학사. 2001년 2월 경희대학교 전자공학과 공학석사. 200년 3월~현재 주식회사 앤스랩. 주 관심분야 : 멀티미디어 통신, 디지털 신호 처리, ASIC 설계