

# 분산산술연산방식을 이용한 MPEG-1 오디오 계층 3 합성필터의 FPGA 구현

## An FPGA Implementation of the Synthesis Filter for MPEG-1 Audio Layer III by a Distributed Arithmetic Lookup Table

구 대 성\*, 고 성 식\*\*, 최 현 용\*\*\*, 김 종 빈\*

(Dae-Sung Ku\*, Sung-Shik Koh\*\*, Hyun-Yong Choi\*\*\*, Jong-Bin Kim\*)

\*조선대학교 전자공학과, \*\*OSAKA City University \*\*\*신한 포토닉스

(접수일자: 2004년 7월 9일; 수정일자: 2004년 10월 26일; 채택일자: 2004년 11월 18일)

반도체 기술과 멀티미디어 통신기술이 발달하면서 고품위 영상과 다중 채널의 오디오에 관심을 갖게 되었다. MPEG 오디오 계층 3 디코더는 표준안에 기반을 둔 프로세서로서 기존에 많이 구현되어 있다. MPEG-1 오디오 계층3 디코더의 합성필터는 디코더 전체에서 가장 많은 연산을 필요로하기 때문에 고속 프로세서를 설계하기 위해서는 연산량을 줄일 수 있는 새로운 방식의 합성필터를 필요로 한다. 따라서 본 논문에서는 MPEG-1 오디오 계층 3의 핵심부분인 합성필터 부분을 DALUT (distributed arithmetic look-up table)방식을 이용하여 FPGA (Field Programmable Gate Array)에 구현하였다. 고속 필터를 설계하기 위해서 승산기 대신에 DALUT 방식을 사용하였고, 파이프라인 구조를 사용하였으며, 데이터를 코사인 함수와 곱셈한 결과를 테이블로 만듦으로써 곱셈기를 제거하여 30%의 성능향상을 얻었다. 본 논문에서의 하드웨어 설계는 모두 VHDL (VHSIC Hardware Description Language)로 기술하였다. VHDL 시뮬레이션은 ALDEC사의 Active-HDL 6.1과 Model-sim 및 합성은 Synplify Pro 7.2v를 사용하였다. 대상 라이브러리는 XILINX사의 XC4010E, XC4020EX, XC4052 XL, P&R 툴은 XACT M1.4를 사용하여 구현하였다. 구현된 프로세서는 20MHz~70MHz사이에서 동작한다.

**핵심용어:** 분산산술연산방식, 합성필터, FPGA, MPEG 오디오, VHDL, XILINX

**투고분야:** 전기음향 분야 (3.2)

As the technologies of semiconductor and multimedia communication have been improved, the high-quality video and the multi-channel audio have been highlighted. MPEG Audio Layer 3 decoder has been implemented as a processor using a standard. Since the synthesis filter of MPEG-1 Audio Layer 3 decoder requires the most outstanding operation in the entire decoder, the synthesis filter that can reduce the amount of operation is needed for the design of the high-speed processor. Therefore, in this paper, the synthesis filter, the most important part of MPEG Audio, is materialized in FPGA using the method of DAULT (distributed arithmetic look-up table). For the design of high-speed synthesis filter, the DAULT method is used instead of a multiplier and a pipeline structure is used. The performance improvement by 30% is obtained by additionally making the result of multiplication of data with cosine function into the table. All hardware design of this paper are described using VHDL (VHIC Hardware Description Language). Active-HDL 6.1 of ALDEC is used for VHDL simulation and Synplify Pro 7.2V is used for Model-sim and synthesis. The corresponding library is materialized by XC4013E and XC4020EX, XC4052XL of XILINX and XACT M1.4 is used for P&R tool. The materialized processor operates from 20MHz to 70MHz.

**Keywords:** Distributed arithmetic lookup table, Synthesis filter, FPGA, MPEG-audio, VHDL, Xilinx

**ASK subject classification:** Electro-acoustics (3.2)

## I. 서론

MPEG 오디오는 고품질 스테레오 및 다 채널 부호화를 위한 ISO/IEC의 표준방식이다. 근래의 모든 DVD 및 오디오 방식들이 다 채널 및 고품질을 추구하고 있다[1].

비디오 압축과는 다르게 오디오 압축에는 3개의 layer를 두고 있다. MPEG-1 레이어 I, II에서는 polyphase 필터뱅크를 사용하지만 레이어 III에서는 MDCT (modified discrete cosine transform)를 이용하여 필터의 출력을 재처리함으로써 결점을 보완하고 있다. Polyphase 필터뱅크와는 다르게 MDCT 부분은 가역변환을 수행하고 더 나은 spectral resolution을 제공하며 polyphase 필터부에서 발생하는 에일리어싱을 부분적으로 제거할 수 있다. 그리고 두 개의 다른 블록을 갖는데 long 블록은 18 샘플이고 short 블록은 6 샘플로 구성되어 있다[2]. 이러한 블록은 연속하는 변환 윈도우간 50% 중첩을 가지므로 윈도우 사이즈는 각각 36과 12를 갖는다. 블록의 기능을 보면 long 블록 길이는 안정된 특징의 오디오 신호에서 더 나은 주파수 분해능을 갖고 short 블록 길이는 순간적 신호에서 더 나은 시간영역 분해능을 제공한다[3]. MPEG-Audio 압축 알고리즘이 제공하는 변형은 인간의 지각적인 측면에서 손실이 없을 뿐만 아니라 612 배의 압축율을 제공한다. PCM (pulse code modulation)방식에서의 오디오 샘플링 주파수는 아래 표 1과 같이 구성되고 비트율은 채널 당 32224Kbps가 주로 사용된다[4,5].

MPEG-1 오디오 계층 3의 두드러진 특징 중의 하나는 서브밴드 부호화로써 오디오 신호를 여러 대역으로 나누고 각각의 대역 신호의 샘플링율을 decimation에 의해 낮춘 후 신호의 크기에 따라 가변적으로 비트를 할당하여 신호대 잡음비를 향상시키는 부호화방식이다. 필터뱅크로는 polyphase 필터뱅크와 weighted overlap-add 필터뱅크로 나눌 수 있는데, MPEG-1 오디오 계층 3에서는 시간공간과 주파수공간에서 같은 수의 샘플로 되어 있으며 연속적인 32개의 오디오 샘플이 하나의 단위로 묶여서 처리된다. 본 논문에서의 합성필터 부분은

표 1. MPEG 오디오 샘플링 주파수  
Table 1. Sampling frequency of MPEG Audio.

Index	주파수
00	44.1 kHz
01	48 kHz
10	32 kHz
11	reversed

IMDCT를 나타낸다. MPEG-1 오디오 계층 3 디코더에서 가장 연산량이 많은 합성필터 부분을 파이프라인 구조로 설계하였다. 그리고 곱셈기에서 21비트 (입력)×21비트 (코사인 함수 출력)연산 시 많은 시간이 소요되는데 코사인 함수와 곱셈한 결과를 테이블로 만들어서 코사인 함수 발생기와 곱셈기를 제거하였다. 따라서 기존방식의 연산기보다 더욱 고속으로 동작하고, 필터부분을 DALUT 방식을 이용하여 효율적으로 하드웨어를 설계하였다.

본 논문의 구성을 보면, II장에서는 DALUT 방식의 이론적인 배경을 설명하고, III장은 합성필터의 FPGA 구현을 위하여 분산산술 LUT와 계수 저장용 LUT를 설계한 내용을 기술하였다. 그리고 IV장에서는 병렬 파이프라인 구조와 III장에서 설계된 하드웨어 구조를 시뮬레이션 및 검증하였고, 마지막으로 V장에서 결론을 맺는다.

## II. 분산산술 연산 방식

분산산술 연산방식의 시분별 시스템에서 선형으로 정의되는 수치의 산술 합은 식 (1)과 같다.

$$y(n) = \sum_{k=1}^K A_k X_k(n) \tag{1}$$

$y(n)$  : 시간 n에서의 네트워크 합.

$X_k(n)$  : 시간 n에서의 k번째 입력 변수.

$A_k$ 는 k번째에 해당하는 변수로써 모든 n에 대하여 상수가 될 수 있다. 필터링 응용에서 상수  $A_k$ 는 필터 계수로 다양한 값이고,  $X_k$ 는 진단 싱글 데이터 소스의 샘플이다. 식 (1)의 곱셈 의존적 성질은 단일 출력 값 k의 출력 누적연산 요구사항을 분석함으로써 알 수 있다. 상수  $A_k$ 는 많은 양의 데이터 워드길이를 필요로 하지 않는다. 변수  $X_k$ 는 분수 포맷으로 식 (2)와 같다[7,8].

$$X_k = -X_{k0} + \sum_{b=0}^{B-1} X_{kb} 2^{-b} \tag{2}$$

$X_{kb}$  : 이진수 값으로 0, 1을 사용한다.

부호비트는  $X_{k0}$  1비트에 의해서 결정된다. 알고리즘에서 가장 중요한 것은 타임 인덱스 n으로써, n은 연속적



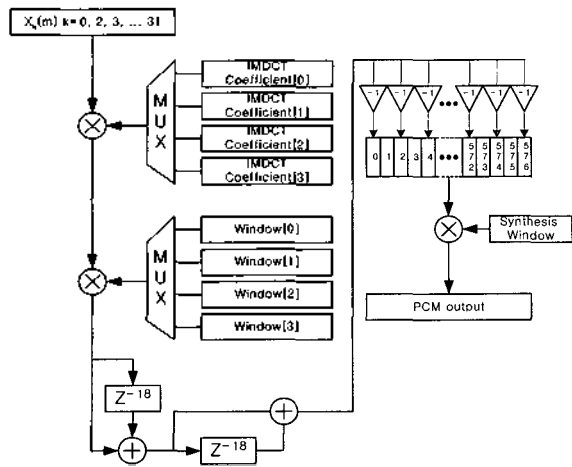


그림 3. 합성필터의 하드웨어 구성  
Fig 3. Hardware architecture of Synthesis Filter.

인 함수는 10차 이상의 다항식이 되어야 일정한 수준의 정밀도를 얻을 수 있다. 그림 3은 합성필터의 하드웨어 구조를 나타낸다.

누산기는 4비트 캐리 (carry) 전달 가산기의 캐리를 선택하는 구조로써 캐리 선택 가산기와 레지스터를 사용하여 설계하였다. 테이블 블록의 종류는 4가지가 존재하는데 long, short 블록만 독립된 테이블로 만들고 다른 블록들은 같은 테이블에 넣어 분산산술 LUT로 곱셈을 구현하였다. 따라서 입력에 대한 적절한 테이블을 찾는 테이블 선택기가 요구되는데, 테이블 선택은 샘플이 입력 순번과 입력값에 따라 결정되도록 카운터와 디코더를 사용하여 어드레스 디코더를 설계하였다.

3.1. 분산산술 LUT 설계

합성필터에서 구현된 DALUT는 속도를 높이기 위해서

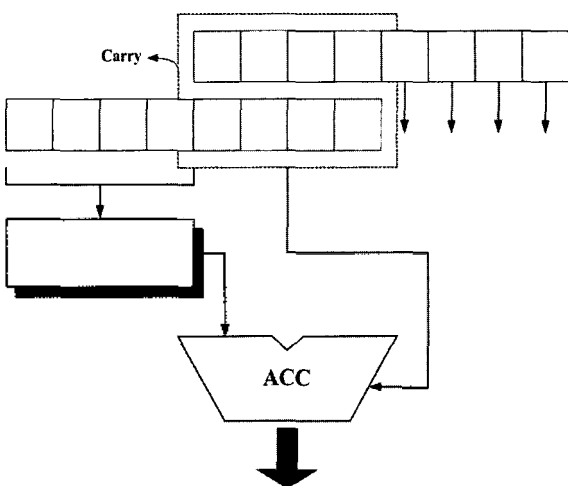


그림 5. DALUT 승산과정의 흐름도  
Fig 5. Flow of DALUT Multiplexing process.

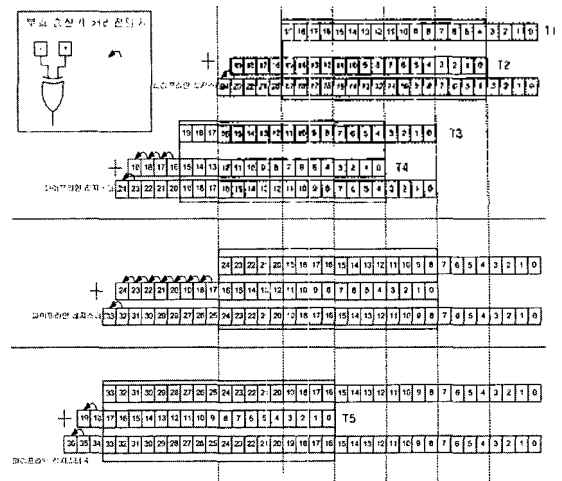


그림 4. 병렬 분산산술 LUT 승산  
Fig 4. Multiplier of parallel distributed arithmetic LUT.

4비트 각 필터계수에 16진수 1 digit를 곱한 결과를 해당 자릿수에 따라 쉬프트 가산함으로써 테이블의 크기를 "각필터계수 $\times 2^4$ "으로 줄일 수 있다. 이러한 구조를 병렬 DALUT라고 하며 병렬 분산산술 LUT의 승산 과정은 그림 4와 같다.

그림 4의 세부과정을 다시 표현하면 그림 5와 같다. 좌측 블록 음영처리 부분의 화살표는 캐리가 발생한다고 가정하고 동시에 계산을 수행하는 방식의 승산 과정의 흐름도를 나타낸다.

그림 6은 입력신호의 블록 종류에 맞는 LUT를 선택할 수 있도록 LUTS (look-up table selector)가 존재하고, 블록 종류에 따라서 데이터열의 크기는 블록 타입이 2일 때 12개, 0, 1, 3인 경우 36개의 데이터를 블록으로 취급함으로써 6비트 카운터를 이용하여 상위 어드레스를 디

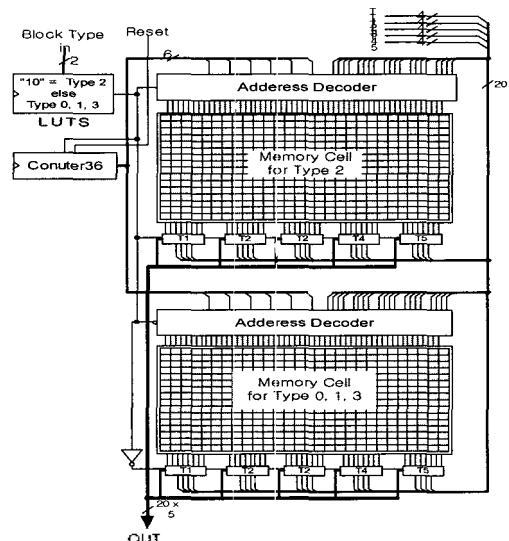


그림 6. MPEG-1 오디오 계층 3 디코더 LUT  
Fig 6. MPEG-1 Audio Layer 3 cecoder LUT.

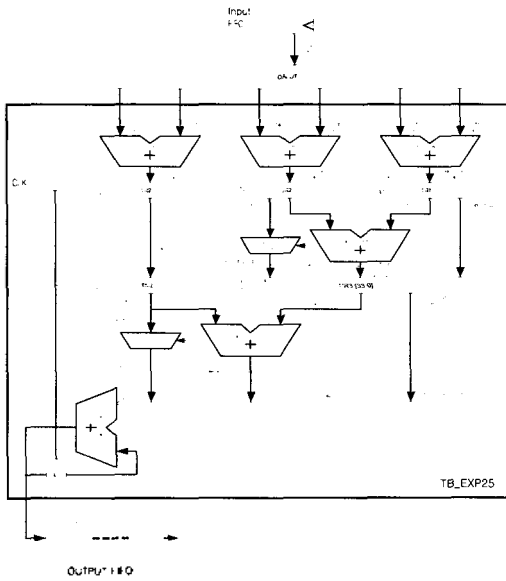


그림 7. DALUT 병렬 파이프라인 승산기  
Fig 7. Parallel pipeline multiplier using DALUT.

코드하고, 부호비트를 제외한 4비트 분산입력 5개가 같은 상위메모리에 대하여 각각의 하위 어드레스를 디코딩한다. 그리고 하위어드레스에 맞는 출력을 수행할 수 있도록 출력용 어드레스 디코더가 존재한다. 그림 6은 MPEG-1 오디오 계층 3 디코더 부분의 LUT를 나타낸 것이다.

그림 7은 분산산술 LUT, 쉬프트 가산기 그리고 누산기를 포함한 병렬 파이프라인 승산기를 나타낸다.

분산산술 LUT 출력은 클럭 동기로 동시에 5개 즉, 21개의 데이터가 출력되고 출력된 데이터를 4비트씩 쉬프트 가산하여 승산할 수 있다. 승산의 입력은 모두 양 (+)의 실수 값으로 취급하여 분산산술 LUT의 크기를 1/2로 하였다. 그리고 입력의 음 (-)은 XOR를 이용하여 별도로 부호 연산을 수행한다. 전체 블록은 클럭 동기로 동작되며 외부 컨트롤러의 제어가 간편하도록 EN (속도 제어용), RESET(초기화), B(동작중)핀을 두었다. 각각의 LUT는 모두 동일한 메모리 공간을 사용함으로써 메모리셀 1개로 구성하고 테이블과 테이블 선택기를 XC-4010E에 합성하였다. 본 블록은 합성필터에서 코사인함수의 승산을 구하기 위한 것이다. 그리고 결과는 합성원도 값을 다시 곱해야 하는데 이것 역시 같은 구조로써 단지 입력을 24비트까지 처리하여 누산 결과를 41비트까지 계산할 수 있도록 처리 비트 크기만 더 크게 설계하였다.

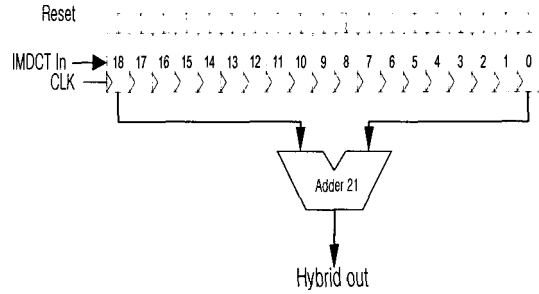


그림 8. 합성필터의 하이브리드 출력  
Fig 8. Hybrid output of Synthesis Filter.

### 3.2. 중첩가산기의 설계

중첩가산기 사용목적은 시간영역의 에일리어싱 제거를 목적으로 한다. 그림 8은 중첩가산기의 구조를 나타낸다.

합성필터는 이전 블록이 현재 블록의 1/2만큼 중첩되게 된다. 즉, 18 샘플씩 중첩가산을 수행해야 한다. 이와 같은 연산은 간단한 FIFO를 이용하여 구현할 수 있다. 복잡한 FIFO의 주소제어용 회로는 사용하지 않고 간단히 처음에 메모리를 초기화하기 위한 Reset 단자, 입출력 신호, 그리고 동기용 클럭으로 구성하였으며 각각의 레지스터를 직렬로 연결하여 구현하였다. 중첩가산은 합성필터의 처리과정으로써 최종적으로 중첩가산기 단계를 거치면 PCM 출력을 얻을 수 있다. 블록길이의 1/2만큼 레지스터를 직렬 연결하여 FIFO를 구성하였고 각각의 레지스터는 25비트로 구성되어있다.

### 3.3. 계수저장용 LUT

병렬처리를 수행함에 있어서 LUT는 메모리 셀 1개로 구성되며 읽기전용으로 다중포트 메모리로 구성하기가 비교적 용이하다. LUT에는 IMDCT를 수행하기 위한 IMDCT 계수값, IMDCT 윈도우 계수 그리고 합성필터 윈도우가 각각 독립된 LUT에 저장된다. 표 2는 IMDCT 계수 LUT의 메모리 맵을 나타낸다.

LUT는 클럭동기 방식으로 동작하며 동시에 5개의 어드레스를 입력받아 5개의 다른 주소공간을 지정할 수 있다. 그리고 5개의 데이터 버스를 두어 동시에 계수 값을 출력할 수 있고 LUT는 이산신호가 순차적으로 입력되므로 자체적인 제어회로를 내장할 수 있다. IMDCT 계수 LUT인 경우 외부입력에 의해 해당 블록 타입의 테이블을 출력하도록 2비트를 할당하였다. 입력 수열의 위치를 계산하여 LUT내의 해당 값을 출력하도록 카운

표 2. IMDCT 계수 LUT의 메모리 맵

Table 2. Memory Map of IMDCT Coefficient LUT.

	short-long	short, long-short, long
0	0x00~0x10	~0xd0
1	~0x20	~0xe0
2	~0x30	~0xf0
3	~0x40	~0x200
4	~0x50	~0x210
5	~0x60	~0x220
6	~0x70	~0x230
7	~0x80	~0x240
8	~0x90	~0x250
9	~0xa0	~0x260
10	~0xb0	~0x270
11	~0xc0	~0x280
12		~0x290
13		~0x2a0
14		~0x2b0
15		~0x2c0
16		~0x2d0
17		~0x2e0
18		~0x2f0
19		~0x300
20		~0x310
21		~0x320
22		~0x330
23		~0x340
24		~0x350
25		~0x360
26		~0x370
27		~0x380
28		~0x390
29		~0x3a0
30		~0x3b0
31		~0x3c0
32		~0x3d0
33		~0x3e0
34		~0x3f0
35		~0x400

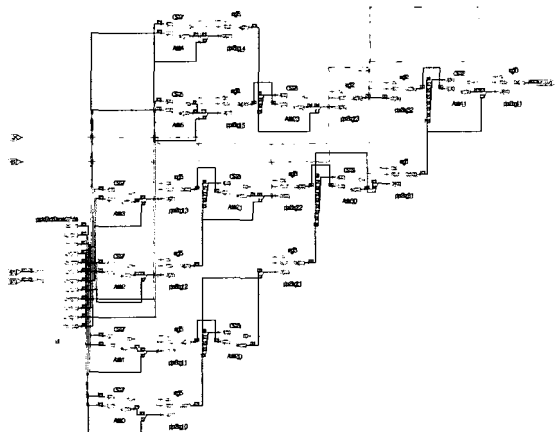


그림 9. 병렬 파이프라인 구조의 합성 블록도

Fig 9. Synthesis block diagram of parallel pipeline Architecture.

터를 가지고 있고, 초기화를 위한 리셋 신호를 포함하고 있다.

## IV. DALUT 시뮬레이션 및 검증

### 4.1. 병렬 파이프라인 구조 설계

LUT 각각의 출력결과를 *csa* (carry save adder)를 이용하여 가산한다. 병렬처리가 불가능한 곳은 중간에 파이프라인 레지스터를 두어 파이프라인 처리를 수행하도록 하였다. 수의 표현은 모두 2의 보수형태를 취했으며 가산기에 캐리 입력을 주면 1의 보수를 2의 보수로 만들 수 있다. 분산산술 LUT는 테이블 부분과 산술연산 부분이 각각 별도로 합성되었으며 산술연산 부분의 마지막 글자는 비트폭을 의미한다. 그림 9는 DALUT방식의 병렬 파이프라인 스키매틱 구조의 합성 블록도를 나타낸 것이다.

### 4.2. 계수저장용 LUT 설계

CLB (configurable logic block)단위의 스키매틱 구조를 보면 그림 10과 같다.

LUT 전체를 FPGA로 구현한다면 많은 양의 clb를 차지하기 때문에 메모리 셀은 ROM에 저장하며 LUT의 컨트롤 블록만을 FPGA에 합성하였다. 그리고 시뮬레이션의 경우 ROM의 값은 component로 제공하여 검증하였다.

### 4.3. 시스템의 합성 및 검증

MPEG-1 오디오 계층 3의 합성필터가 본 논문에서 구

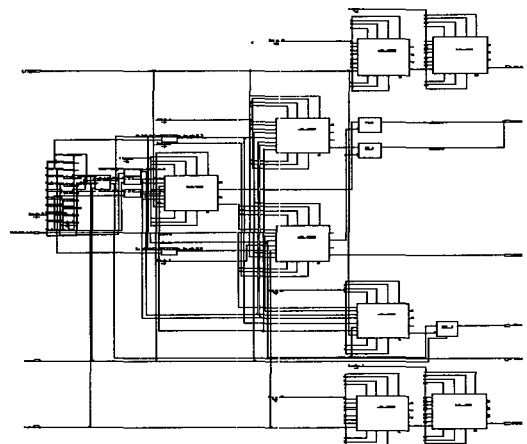


그림 10. LUT제어 유닛의 CLB 합성 블록도

Fig 10. Synthesis block of CLB in LUT control unit.

표 3. 합성필터의 FPGA 설계 자원.  
Table 3. FPGA design resource of synthesis filter.

FG Function Generators	1274
H Function Generators	253
Number of CLB cells	973
Number of Hard Macros and Other Cells	274
Number of CLBs Other Cells	322
Total Number of CLBs	1295
Number of Ports	93
Number of Clock Pads	0
Number of IOBs	0
Number of Flip Flops	1496
Number of 3-State Buffers	0
Total Number of Cells	1247

현하고자하는 부분으로써, 전단에서 설계한 모든 블록을 포트맵하여 연결하면 합성필터가 완성된다. 합성필터의 전체 구성은 하드웨어 전용 언어인 VHDL을 사용하였으

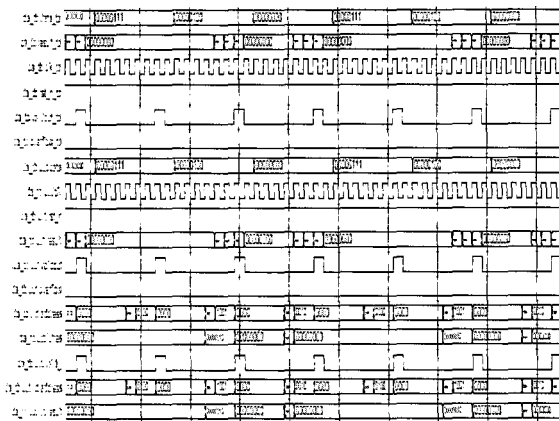


그림 11. DALUT 합성필터 결과 파형  
Fig 11. Result Waveform of DALUT Synthesis Filter.

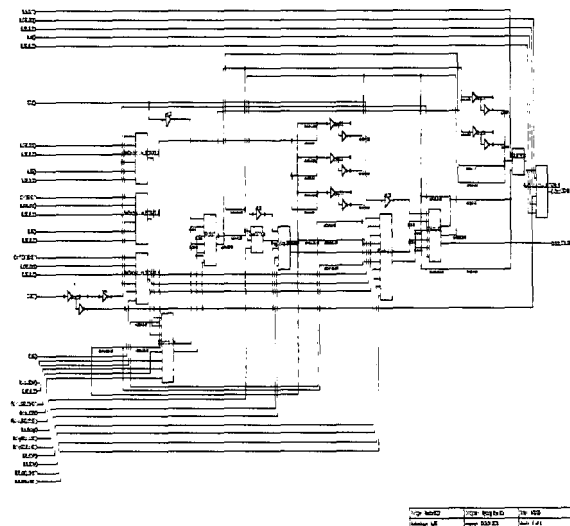


그림 12. 합성필터의 계층 구조  
Fig 12. Hierarchy architecture of synthesis filter.

며 합성은 Synplify 및 Synopsys를 이용하였다. 합성필터 전체구조의 시뮬레이션 결과는 그림 11과 같고 시뮬레이션은 Model-sim에서 수행하였다.

그림 12는 DALUT를 이용한 합성 필터 전체의 계층구조를 스카메틱 구조로 표현한 부분으로써, Synopsys를 이용하여 합성하였다.

표 3은 본 논문에서 구현한 합성필터 전체를 구성하는 FPGA 자원을 나타낸다.

### V. 결론

본 논문에서는 MPEG-1 오디오 계층 3 디코더의 핵심 부분인 합성 필터부분을 DALUT 방식을 이용하여 xilinx XC4010E, XC4020EX, XC4052XL을 사용하였고 P&R 툴은 XACT M1.4를 사용하여 FPGA에 구현하였다. 그리고 VHDL 시뮬레이션은 ALDEC사의 Active-HDL 6.1과 Model-sim 및 합성은 Synplify Pro 7.2v를 사용하였다.

고속 합성필터를 구현하기 위해서 승산기를 사용하지 않고 DALUT방식을 이용하여 실시간에서는 최소 20 MHz에서 동작하고 설계된 칩의 최대 동작 주파수는 70MHz이다. DALUT구조는 승산기를 필요치 않으므로 고속인 대신 대용량의 메모리가 추가로 요구된다. 시스템의 모든 모듈을 VHDL로 구현하였기 때문에 ASIC에 적용하기가 매우 용이하다. 구현된 시스템은 실제 여러 개의 FPGA에 나뉘어 합성되나 ASIC으로 구현할 경우 합성효율이 매우 좋아져 크기가 훨씬 작아진다. 그리고 속도가 향상되어 실제로 50채널 이상을 복호화할 수 있다. 전체 시뮬레이션을 위해서는 고급 컴퓨터 사양과 긴 시간을 요구하며 결과의 검증도 매우 어려운 작업이다. 검증은 C언어로 작성한 소프트웨어와 E-CAD Tool을 사용한 시뮬레이션 결과를 비교 검증하였다. 그리고 VHDL의 Text-IO를 사용하여 2의 보수 형태 출력을 파일로 저장하고, 소프트웨어로 이를 PC 스피커를 통해 플레이하도록 하였다. 검증결과 동작 속도면에서 30% 향상되었다.

본 논문에서 제안된 DALUT방식의 합성필터는 MPEG-1 오디오 계층 3의 사운드 Codec용 DSP 칩, MP-3 플레이어, MP-3 폰 및 실시간 디코더에서 핵심적인 필터로 적용할 수 있다.

참고 문헌

1. Davis Pan, "A Tutorial on MPEG/Audio Compression" Motorola Inc.
2. KEN C. POHLMAN, *PRINCIPLES OF DIGITAL AUDIO 3rd Edition*, (McGraw-Hill, 1997), 354-415.
3. J. Princen, A. Johnson, A. Bradley, "Subband/Transform Coding Using Filter Bank Designs Based on Time Domain Aliasing Cancellation", Proc. of the ICASSP, pp. 2161-2164, 1987.
4. The Sorer, Kh. Brandenburg, "The use of multirate filter banks for coding of high quality digital audio" 6th European Signal Processing Conference, Amsterdam, 1, pages 211-214, June 1992.
5. Ted painter, Andreas Spanias, "A Review of Algorithms for Perceptual Coding of Digital Audio Signals" Department of Electrical Engineering, Telecommunications Research Center Arizona State University.
6. "The Role of Distributed Arithmetic in FPGA based Signal Processing" www.xilinx.com
7. P.P. VAIDYANATHAN, *MULTIRATE SYSTEMS AND FILTER BANKS*, (Prentice Hall Englewood Clieffs, 1993).
8. ALEXANDER D. POULARIKAS, "The Transforms and Applications Hand Book" CRC and IEEE PRESS, 1996.
9. ISO/IEC JTC1/SC29/WG11 MPEG, International Standard IS 13818-3 (Second Edition) "Information Technology- Generic Coding of Moving Pictures and Associated Audio, Part 3: Audio", 1997.
10. Vijay K. Madisetti "VLSI digital signal processors An Introduction to Rapid Prototyping and Design Synthesis" IEEE PRESS.

저자 이력

• 구 대 성 (Ku Dae Sung)



1973년 10월 3일생  
 1999년 2월: 조선대학교 정보통신공학과 (공학사)  
 2001년 2월: 조선대학교 전자공학과 (공학석사)  
 2004년 2월: 조선대학교 전자공학과 (박사과정 수료)  
 2002년 3월~2003년 12월: (주)슈퍼테크놀로지 연구원

• 고 성 식 (Koh Sung shik)



1972년 2월 2일생  
 1994년 2월: 조선대학교 전자공학과 (공학사)  
 1996년 2월: 조선대학교 전자공학과 (공학석사)  
 2002년 2월: 조선대학교 전자공학과 (공학박사)  
 2003년 3월~2004년 8월: 조선대학교 전자공학과 겸임교수  
 2004년 9월~현재: 오사카 사립대학교 공학연구과 (박사후 해외연수 과정)

• 최 현 용 (Choi Hyun Yong)



1975년 8월 4일생  
 1997년 2월: 호남대학교 전자공학과 (공학사)  
 1999년 2월: 조선대학교 전자공학과 (공학석사)  
 2004년 2월: 조선대학교 전자공학과 (공학박사)  
 2001년 3월~현재: (주)신한포토닉스 광네트워크 연구소 연구소장  
 2004년 9월~현재: 한국산업기술평가원 기술개발기회 평가단 평가위원

• 김 중 빈 (Kim Jong Bin)



1945년 5월 4일생  
 1967년 2월: 조선대학교 전기공학과 (공학사)  
 1980년 2월: 건국대학교 전자공학과 (공학석사)  
 1993년 2월: 광운대학교 재료공학과 (공학박사)  
 1992년 1994년: 한국통신학회지부장  
 1999년 3월~2004년 2월: 조선대학교 전자부품재료 설계 인력교육센터 (EMDEC) 소장  
 1983년 3월~현재: 조선대학교 전자공학과 교수