

# 직렬연결 다채널 스피커의 PC 기반 제어 시스템

이 선 응<sup>\*</sup> · 김 태 완<sup>†</sup> · 변 지 성<sup>‡</sup> · 송 문 빈<sup>‡‡</sup> · 정 연 모<sup>‡‡‡</sup>

## 요 약

본 논문에서는 최근에 연구되어 발표된 하나의 선을 사용하여 여러 채널의 음향 신호를 전송하는 기술인 다채널 직렬연결 스피커 시스템에 USB 인터페이스를 사용하여 PC 환경에서 보다 많은 채널의 음향 신호를 제어할 수 있는 시스템을 제시하였다. USB 호스트에서 음원 파일을 분석하고 처리한 후 전송 알고리즘에 맞게 패킷을 생성하여 오디오 데이터를 실시간으로 전송한다. 각 스피커에서는 해당하는 디지털 신호만을 검출하여 처리한 후 DAC를 통해 음향을 재생한다. 사용자는 PC에서 시스템을 GUI 환경을 통해서 쉽게 제어할 수 있다.

키워드 : 멀티채널, 오디오 시그널, USB 인터페이스, 직렬 전송

## PC-based Control System of Serially Connected Multi-channel Speakers

Sunyong Lee<sup>\*</sup> · Taewan Kim<sup>†</sup> · Jisung Byun<sup>‡</sup> · Moonvin Song<sup>‡‡</sup> · Yunmo Chung<sup>‡‡‡</sup>

## ABSTRACT

In this paper, we propose a system which easily controls the existing serially connected multi-channel speakers in a general personal computer by using a USB(Universal Serial Bus) interface. The personal computer as a host of the USB interface analyzes a sound source and sends audio data in a real-time fashion by the use of the isochronous transmission, one of four transmission methods provided by the USB interface. In addition, a channel is assigned by means of the bulk transmission, one of four transmission methods provided by the USB interface. Transmitted data from the USB host are sent to each speaker through compression and packet generation process. Each speaker detects corresponding digital data and regenerates audio signals through DAC(Digital-to-Analog Converter). A user can easily select a sound source file and a channel by the use of a GUI environment in a personal computer.

Keywords : Multi-Channel, Audio Signal, USB Interface, Serial Transmission

## 1. 서 론

오디오 시스템에서 사용하는 스피커의 숫자는 날로 증가하고 있다. 또한 스튜디오나 공연장 같은 경우에는 많은 수의 스피커를 사용한다. 이에 따라 스피커를 연결하는 선의 수를 관리하는 것이 어려워지고 있다. 따라서 시스템 구성이 복잡해 질 뿐만 아니라 설치에도 많은 시간이 소요된다.

이러한 문제를 해결하기 위해서 스피커의 연결을 효율적으로 할 수 있도록 이더넷을 이용한 코브라넷(Cobranet) 기술이 개발되었다. 코브라넷은 일반적인 이더넷 프로토콜을 통해 여러 채널의 음향 데이터를 전송하기 위해 디자인된 소프트웨어, 하드웨어, 네트워크 프로토콜의 결합체이다. CAT5 케이블이나 광케이블을 통해 실시간으로 64채널의

오디오를 전송할 수 있다. 하나의 케이블을 사용하여 전송하기 때문에 케이블에 들어가는 비용은 절약이 되지만, 코브라넷 신호를 인코딩, 디코딩하고 스위칭하는 장비에 많은 비용이 드는 단점이 있다.<sup>[8]</sup>

최근에 연구되고 있는 다채널 직렬연결 스피커 시스템은 모든 스피커를 하나의 선으로 연결하는 것으로 코브라넷의 고 비용을 줄일 수 있는 대안이다.<sup>[1-3]</sup> 이 시스템은 각 아날로그 채널을 디지털화 하고 각 스피커에 ID를 부여하여 ID와 샘플링 데이터로 패킷을 만들어 직렬로 수신부에 전송하는 기술이다. 그러나 이 시스템의 경우에는 여러 채널을 아날로그 신호로 각각 입력 받은 뒤에 ADC(Analog to Digital Converter)를 통해서 처리해야 할 뿐만 아니라, 입력 받은 오디오 선이 많아짐에 따라 시스템이 복잡해지는 단점이 있다. 또한 스피커의 채널 변경을 위해서 시스템에 있는 스위치를 물리적으로 작동해야 하는 번거로움이 있다.

본 논문에서는 다채널 직렬연결 스피커 시스템에 USB(Universal Serial Bus)를 이용하여 GUI(Graphical User Interface) 환경으로 PC에 저장된 여러 음원파일을 한 번에

<sup>\*</sup> 준 회원 : 경희대학교 전자공학과 석사과정  
<sup>††</sup> 정 회원 : 마린콤(주) 대표이사  
<sup>‡‡‡</sup> 종신회원 : 경희대학교 전자전파공학전공 교수  
논문접수 : 2008년 7월 3일  
수정일 : 1차 2008년 9월 8일, 2차 2008년 10월 8일  
심사완료 : 2008년 10월 8일

전송하고<sup>[5-6]</sup>, 채널도 쉽게 변경할 수 있는 시스템의 구조를 제시하였다.

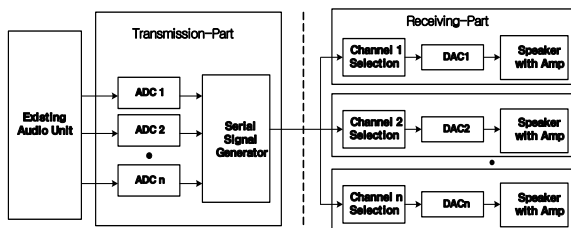
USB는 현재 널리 사용되고 있는 표준 버스 규격으로서 빠른 전송속도를 가지고 있으며 포트의 크기가 작아 휴대용 저장장치 및 미디어 기기에 널리 사용된다. USB는 제어전송, 벌크 전송, 인터럽트 전송, 그리고 등시성 전송의 네 가지 전송방법을 가지고 있다. 그 중에서도 실시간 전송에 적합한 등시성 전송을 사용하여 오디오 데이터를 전송하고, 벌크 전송을 사용하여 채널을 선택한다.<sup>[4,7]</sup>

### 2. 다채널 직렬연결 스피커 개념

기존의 다채널 직렬연결 스피커 시스템은 (그림 1)과 같은 구조를 가지고 있다.

이 시스템은 외부 오디오 시스템으로부터 여러 채널의 아날로그 음향 신호를 받아서 ADC를 통해 디지털 신호로 변환한다. 그런 후 각각의 채널 별로 압축하고 해당 채널 ID와 함께 패킷화 하는 과정을 거친다. 그리고 순서대로 직렬로 수신부에 전송한다. 수신부에서는 채널 ID와 스피커의 ID를 체크하여 두 ID가 같으면 데이터를 받아 압축을 해제하고 DAC를 거쳐 스피커로 음향 신호를 출력한다.

이 시스템의 경우에는 별도의 외부 오디오 시스템이 필요하고, 채널 수 만큼 ADC가 필요하기 때문에 시스템의 크기가 커지게 된다. 또한 처리할 수 있는 채널의 수가 하드웨어적으로 고정되어 있기 때문에 채널수를 변경하는 것이 불가능하다. 또한 원하는 음향을 재생하고 선택함에 있어 외부 오디오 시스템을 조작해야하는 번거로움이 있어 유연성이 떨어지는 단점이 있다.



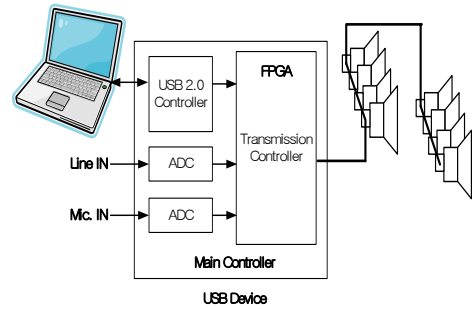
(그림 1) 기존 전송 및 수신측 블록도  
(Fig. 1) Transmission and Receiving-part block diagram of the existing system

### 3. 다채널 직렬연결 스피커의 USB 인터페이스

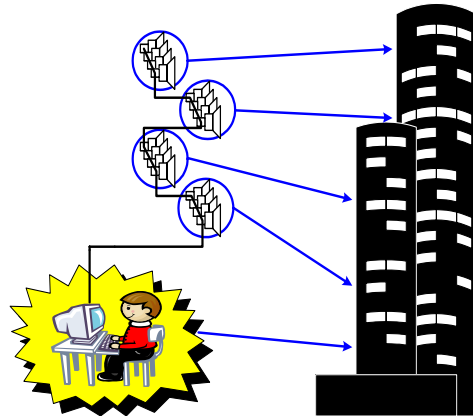
위와 같은 문제들을 해결하기 위해 본 논문에서는 (그림 2)와 같이 PC와의 USB 인터페이스를 이용하여 여러 채널의 음원을 제공받아 전송하는 방법을 제시하였다.

#### 3.1 다채널 직렬연결 스피커의 USB 인터페이스

(그림 2)와 같이 기존의 외부 오디오 시스템 대신 PC를 사용하여, 저장된 파일을 USB 2.0 프로토콜로 USB 디바이스



(그림 2) USB 인터페이스 스피커의 직렬연결 방법  
(Fig. 2) Serial connection method of speakers by a USB interface



(그림 3) 빌딩의 시스템 구현 예  
(Fig. 3) System implementation example within a building

스인 주제어부(Main Controller)로 전송한다. 이 과정에서 디지털 음향 데이터가 저장된 파일을 사용하였기 때문에 별도의 ADC과정이 필요 없게 된다. 또한 기존의 시스템과 비교했을 때, USB 컨트롤러와 USB 인터페이스를 위한 기능이 추가되었다. 이 시스템을 응용하면 사용자는 (그림 3)과 같이 빌딩의 모든 스피커 들을 PC에서 GUI 환경을 통해 제어할 수 있다.

#### 3.2 시스템 설계

본 논문에서 제시하는 시스템은 PC가 USB의 호스트 역할을 수행한다. PC에서는 USB 어플리케이션으로 음원 파일을 선택하여 전송할 수 있도록 재구성한다. 이 과정에서는 wave 파일의 지원은 물론 mp3 파일을 디코딩하는 기능까지 포함된다. 그리고 USB API 함수를 이용하여 호스트 송신 버퍼에 데이터를 채워 넣게 되면 USB 전송이 이루어진다. 여러 채널의 음원 데이터를 받은 주제어부는 각각의 채널을 압축하고 패킷화 하는 과정을 거쳐 각 스피커에 전송한다.

##### 3.2.1 시스템 설계 사양

본 논문에서 제안한 다채널 오디오 스피커의 직렬연결 시

<표 1> 개발 사양  
<Table 1> Development specifications

개발 내용	사양	비고
USB 인터페이스	2.0	High Speed 지원
USB 전송방법	등시성 전송	음원데이터 전송
	벌크 전송	채널 선택
디지털 샘플링	I <sup>2</sup> S	디지털 변환 방식
샘플링 비율	48kHz	디지털 CD 수준 이상
샘플링 크기	24bit	각 채널당
샘플링 주파수 대역	16Hz - 20kHz	가청 주파수
지원 음향 파일	wave, mp3	

시스템은 <표 1>과 같이 사양을 가지고 있다.

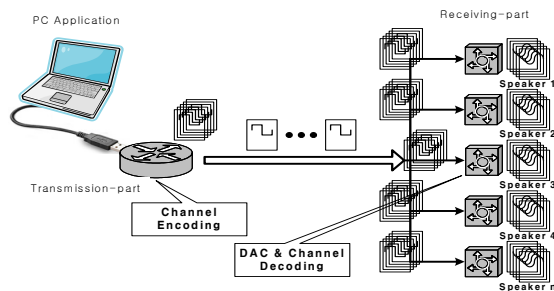
USB 2.0 High Speed 등시성 전송의 경우 최대 속도 24.5765MByte/sec를 지원하기 때문에 여러 음향 채널의 실시간 전송이 가능하다. 또한 벌크 전송으로 채널 선택이 가능하도록 구성하였다. 외부에서 아날로그 오디오 신호와 마이크 신호를 입력 받을 수 있도록 ADC를 추가하고 I<sup>2</sup>S(Inter-IC Sound) 버스 신호로 변환하여 처리 할 수 있도록 하였다. 전송부 측의 ADC와 스피커 측의 DAC(Digital to Analog Converter)에서의 샘플링 주파수는 48kHz이고 이 때 샘플링 해상도는 각 채널마다 24비트이다. 그리고 PC에서는 wave 파일과 mp3 파일을 지원한다.

3.2.2 전체 시스템 개념

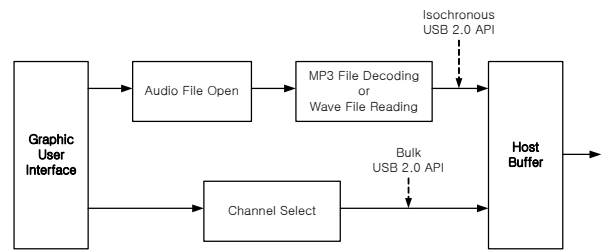
다채널 직렬연결 시스템의 전체적인 개요는 (그림 4)와 같다. 기존의 아날로그 신호 입력대신에 PC와 USB 통신을 통하여 채널을 입력받게 된다. 전체 블록도는 (그림 5)의 PC 어플리케이션과, (그림 6)의 전송측, 그리고 (그림 7)의 수신측으로 이루어진다.

PC 어플리케이션은 사용자가 원하는 음원 파일인 mp3 파일 또는 wave 파일을 읽어 온다. mp3 파일의 경우는 디코딩 과정을 거쳐 등시성 전송 API 함수를 사용하여 USB 송신 버퍼에 데이터를 써 넣는다. 또한 채널 선택을 위해 벌크 전송 API 함수를 이용하여 전송한다.

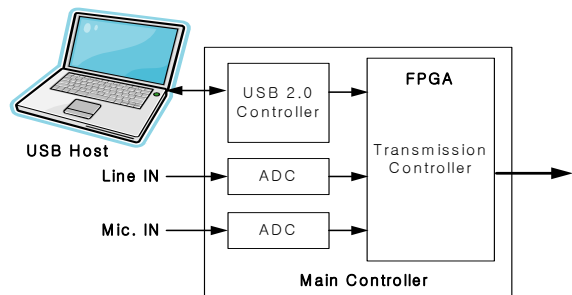
전송 측에서는 USB 호스트인 PC에서 여러 개의 음원 데



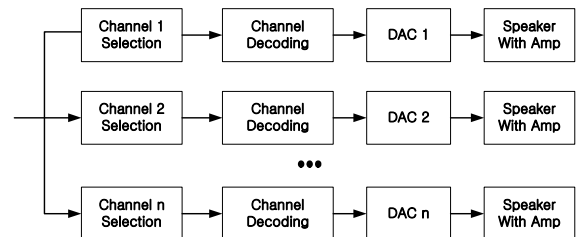
(그림 4) 시스템 개요도  
(Fig. 4) System configuration



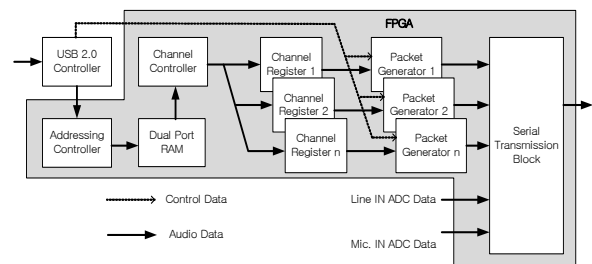
(그림 5) PC 어플리케이션 블록도  
(Fig. 5) PC application block diagram



(그림 6) 전송측 블록도  
(Fig. 6) Transmission part block diagram



(그림 7) 수신측 블록도  
(Fig. 7) Receiving part block diagram



(그림 8) FPGA의 내부 블록도  
(Fig. 8) Internal block diagram of FPGA

이터를 전송하면 USB 컨트롤러에서는 프로토콜에 맞춰 채널 데이터만 FPGA로 넘겨주게 된다. FPGA에서는 (그림 8)과 같이 USB 컨트롤러에서 받은 채널 데이터를 가지고 세부적인 과정을 수행한다. 실선으로 표시된 것은 오디오 신호의 흐름을 나타낸 것이고, 점선으로 표시된 것은 채널 선택을 위한 제어 신호를 나타낸 것이다. 어드레싱 컨트롤러에서는 USB 컨트롤러로부터 받은 샘플링 데이터를 듀얼

포트 RAM에 저장할 수 있도록 주소를 지정해 준다. 그리고 채널 컨트롤러에서는 듀얼 포트 RAM에 저장된 데이터를 샘플링 간격으로 읽어 오게 된다. 오디오신호가 계속 재생되기 위해서는 데이터가 끊이지 않아야 한다. 따라서 RAM의 값을 실시간으로 업데이트 해주고 계속 읽어 와야 한다. RAM에서 읽어온 데이터는 각각의 채널 별로 레지스터에 저장된 후, 패킷 생성기를 통해 압축 및 패킷화 과정을 거쳐 직렬로 스피커에 전송하게 된다.

스피커에서 원하는 채널을 선택하기 위해 벌크 전송을 사용하여 채널 선택할 수 있는 제어 데이터를 추가로 전송할 수 있도록 하였다. 채널 선택을 위한 제어 데이터는 패킷 생성기에서 패킷의 ID로 설정이 된다.

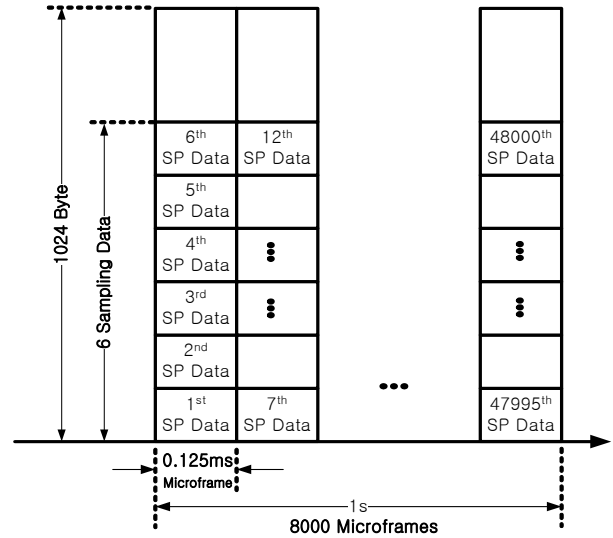
3.2.3 USB 전송

USB의 네 가지 전송방법 중에 실시간 전송에 적합한 것이 등시성 전송이다. 등시성 전송의 경우 데이터를 전송한 후, 에러가 발생을 해도 재전송을 하지 않는다. 그렇기 때문에 핸드셰이크 패킷의 전송을 생략한다. 실시간으로 재생이 되는 오디오나 비디오의 경우 한 두 비트 에러는 사람의 귀나 눈으로 확인이 불가능하기 때문에 정해진 시간에 데이터가 도착하지 않는 것이 더 큰 오류가 될 수 있다.

벌크전송의 경우 전송 타이밍이 중요하지 않은 데이터 전송에 유리하다. 다른 전송방식에 비해 버스를 양보하고 사용할 수 있을 때까지 기다리기 때문에 버스의 다른 전송을 방해하지 않고 데이터를 보낼 수 있다.

따라서 본 논문에서 제안한 시스템에서는 등시성전송으로 오디오 데이터를 전송하고, 벌크전송으로 채널 선택을 위한 제어 데이터를 전송한다.

USB 호스트 어플리케이션에서 등시성전송 API 함수를 이용하여 송신 버퍼에 보내고자 하는 채널 데이터를 써 넣는다. 시스템의 샘플링 주파수가 48kHz 이므로 하나의 채널 당 1초에 48,000개의 샘플링 데이터를 보내야 한다. USB 2.0 등시성 전송의 경우 한 번에 보낼 수 있는 최대 패킷의 크기가 1,024 바이트 이고, 그 간격이 0.125ms 이다. 따라서 1초 동안에 8,000개의 패킷을 보낼 수 있고, 하나의 패킷에 6개의 샘플링 데이터를 실어서 보낼 수 있다. (그림 9)은 PC 어플



(그림 9) 등시성 패킷 전송 과정  
(Fig. 9) Isochronous packet transmission process

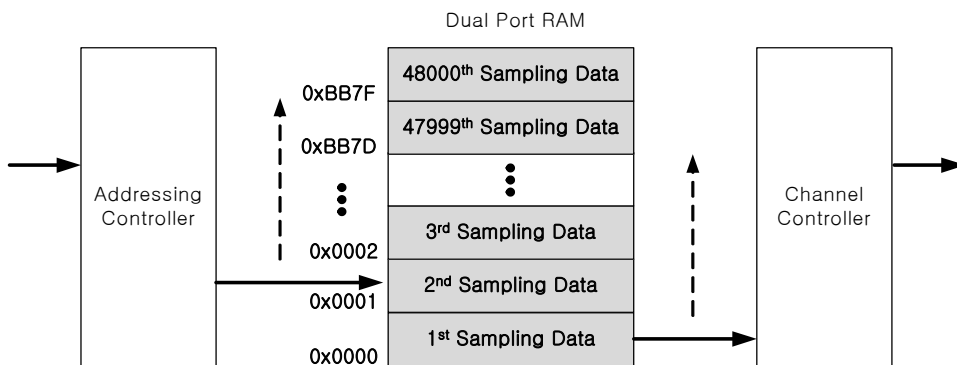
리케이션에서의 등시성 패킷 전송 과정을 보여주고 있다.

첫 번째 패킷에는 1 - 6 번째 샘플링 데이터를 실어 보내고, 두 번째 패킷에는 7 - 12 번째 샘플링 데이터를 실어 보낸다. 이런 과정을 거쳐 마지막에는 47,995 - 48,000 번째 샘플링 데이터를 보내게 된다. 지금까지의 과정이 1초 동안 이루어지고 계속 반복하게 된다.

3.2.4 USB 인터페이스

이 시스템에서 가장 중요한 것은 실시간으로 오디오 신호가 재생이 되는 것이다. 따라서 PC에서 보내 준 채널 데이터를 48kHz의 샘플링 주파수에 맞춰서 데이터의 누락 없이 스피커로 전송해야 한다. USB 2.0 등시성전송에서는 매 0.125ms 마다 1024 바이트의 데이터가 전송이 되므로 48kHz로 맞추기 위해서는 중간에 버퍼나 혹은 메모리가 있어야 한다. 실시간으로 데이터를 업데이트하고 읽어와야하므로 듀얼 포트 RAM을 사용하여 한 쪽 포트에서는 계속 쓰고, 다른 한 쪽 포트에서는 계속 읽을 수 있게 하였다.

(그림 10)은 듀얼 포트 RAM의 인터페이스를 보여주고



(그림 10) 듀얼 포트 RAM 인터페이스  
(Fig. 10) Dual port RAM interface

있다. 어드레싱 컨트롤러에서는 USB 컨트롤러에서 받은 채널 샘플링 데이터를 메모리의 0x0000번지부터 샘플링 순서대로 저장하도록 해준다. 그런 후, 채널 컨트롤러에서는 메모리의 0x0000번지부터 샘플링 주파수 간격으로 데이터를 읽어오게 된다. 메모리의 0x0000번지부터 마지막 번지인 0xBB7F까지 읽는 데에는 1초의 시간이 걸리고, 이 과정을 계속 반복하게 된다.

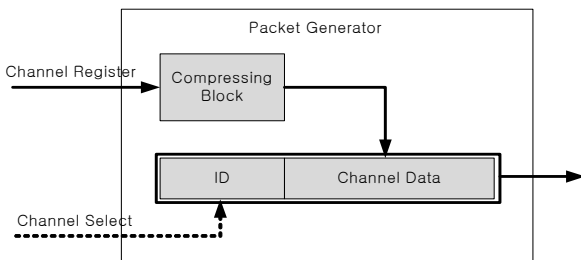
### 3.2.5 패킷 생성 및 전송

듀얼 포트 RAM에서 각각 채널별로 샘플링 데이터를 읽은 후, 직렬 전송하기 전에 압축 및 패킷화하는 과정을 거치게 된다. (그림 11)는 압축하고 패킷화하는 과정을 보여준다.

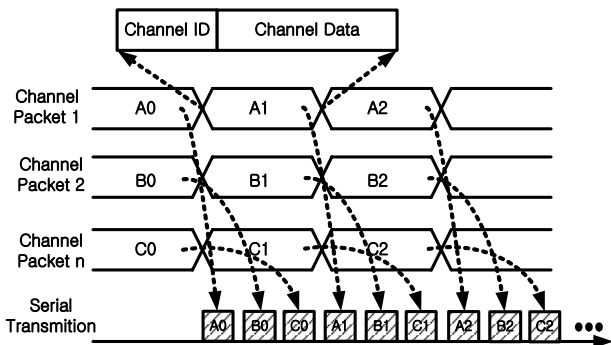
RAM에서 읽은 데이터가 채널 레지스터에 저장된다. 그리고 그 데이터가 패킷 생성기의 압축 블록을 지나 압축 과정을 거치게 된다. 음향신호를 디지털로 샘플링 한 경우에는 서로 인접하는 데이터간의 변화량이 매우 작다. 따라서 이러한 특성을 이용하면 큰 압축효율을 가질 수 있다.[3] 그리고 벌크전송을 사용하여 전송한 채널 선택 데이터가 패킷의 ID로 설정되어 압축된 데이터와 함께 패킷이 만들어진다.

직렬 전송 블록에서는 각각의 채널에서 생성된 패킷을 순차적으로 직렬 전송하게 된다. (그림 12)은 패킷을 직렬 전송 하는 과정을 보여주고 있다.

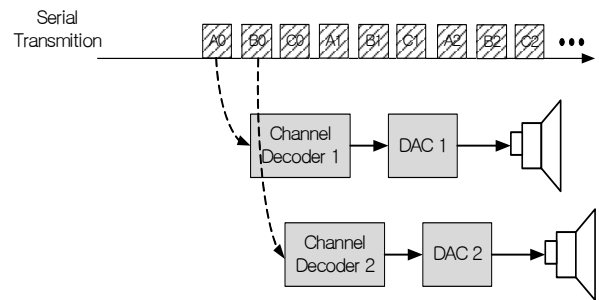
각 채널의 패킷은 샘플링 주파수인 48kHz의 간격으로 생성이 된다. 따라서 다음 패킷이 만들어지기 전까지 순차적으로 한 비트씩 전송하게 된다. 모든 스피커가 하나의 선으



(그림 11) 패킷 생성 과정  
(Fig. 11) Packet generation process



(그림 12) 패킷 전송 과정  
(Fig. 12) Packet transmission process



(그림 13) 패킷 복원 과정  
(Fig. 13) Packet regeneration process

로 연결되었기 때문에 각 스피커는 모든 패킷의 ID를 확인하여 자신과 일치하는 ID를 가진 패킷만 선택하여 읽어오게 된다. 그런 후, 압축된 데이터를 복원하고 DAC를 지나 스피커에 해당 소리가 나오게 된다. (그림 13)은 패킷의 복원 과정을 보여주고 있다.

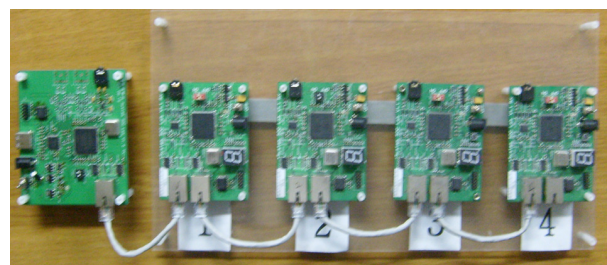
## 4. 구현 및 성능 평가

본 논문에서 구현한 다채널 직렬연결 스피커의 USB 인터페이스는 Cypress사의 8051 코어가 내장된 USB 2.0 컨트롤러와 Altera사의 Cyclone 2 FPGA를 사용하여 성능 실험을 위해서 구현하였다. 8051 코어를 동작하기 위한 펌웨어 (firmware)를 작성하여 열거과정을 수행하고, 등시성 OUT 엔드포인트와 벌크 OUT 엔드포인트를 사용하여 파이프를 구성하였다.

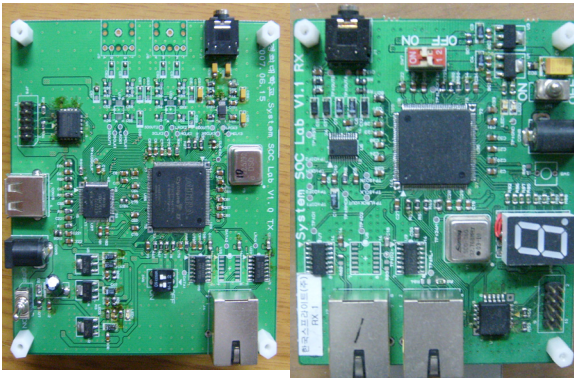
(그림 14)는 본 논문에서 제시한 시스템의 성능을 확인하기 위해서 1개의 전송부와 4개의 수신부를 연결하기 위해서 만든 보드 사진이고 (그림 15)는 확대한 보드 사진이다.

구현한 시스템에서 한 채널의 음원 파일을 직접 재생한 오디오 신호와, USB 전송으로 전송부와 수신부를 모두 거쳐 최종적으로 재생된 오디오 신호의 파형을 각각 비교한 것이 (그림 16)과 같다.

샘플링 크기는 24 비트이며, 샘플링 주파수는 48kHz이다. 그림에서 처음의 그래프는 입력으로 주어지는 음원 파일을 그래프로 나타낸 것이고, 아래의 그래프는 USB 전송 후에 압축 및 패킷처리 과정, 그리고 전송과정을 거쳐 DAC로 거치기 전의 복원된 데이터를 UART를 통해 PC로 전송하여



(그림 14) 전송부와 수신부를 연결한 사진  
(Fig. 14) Photograph for transmission and receiving connection

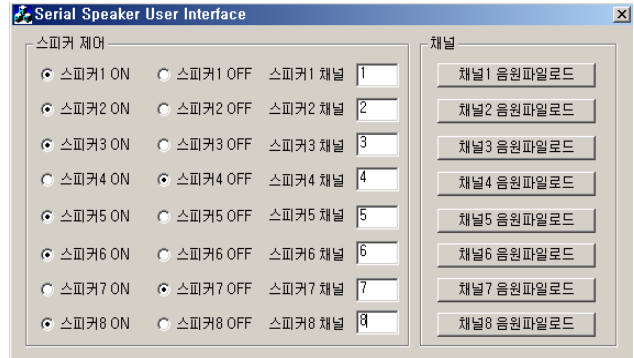


(그림 15) 보드 확대 사진(좌 : 전송부, 우 : 수신부)  
(Fig. 15) Board enlargement photographs (left : transmission part, right : receiving part)

그래프로 나타낸 것이다. 그리고 마지막 그래프는 두 그래프의 차이 값을 나타낸 것이다. 두 개의 그래프를 비교해보면 원본과 재생 신호가 유사한 것을 알 수 있다.

(그림 17)에서는 총 8개의 스피커를 제어할 수 있는 사용자 인터페이스를 나타낸 것이다. 각 스피커의 ON, OFF 및 스피커별로 다른 채널을 할당할 수 있게 하였으며 각 채널별로 음원파일을 읽어올 수 있다.

<표 2>는 코브라넷과 제안한 시스템의 실험적인 성능 비교를 나타낸 것이다. 채널 수와 해상도, 샘플링 주파수는 두 시스템이 비슷하다. 하지만 코브라넷의 경우는 다수의 네트워크 장비를 사용하기 때문에 설치 및 유지 보수 비용이 고가인 반면에 제안한 시스템의 경우는 USB 컨트롤러와 FPGA만을 사용하여 시스템이 간단하기 때문에 설치 및 유지 보수 비용이 저렴하다.<sup>[8]</sup>



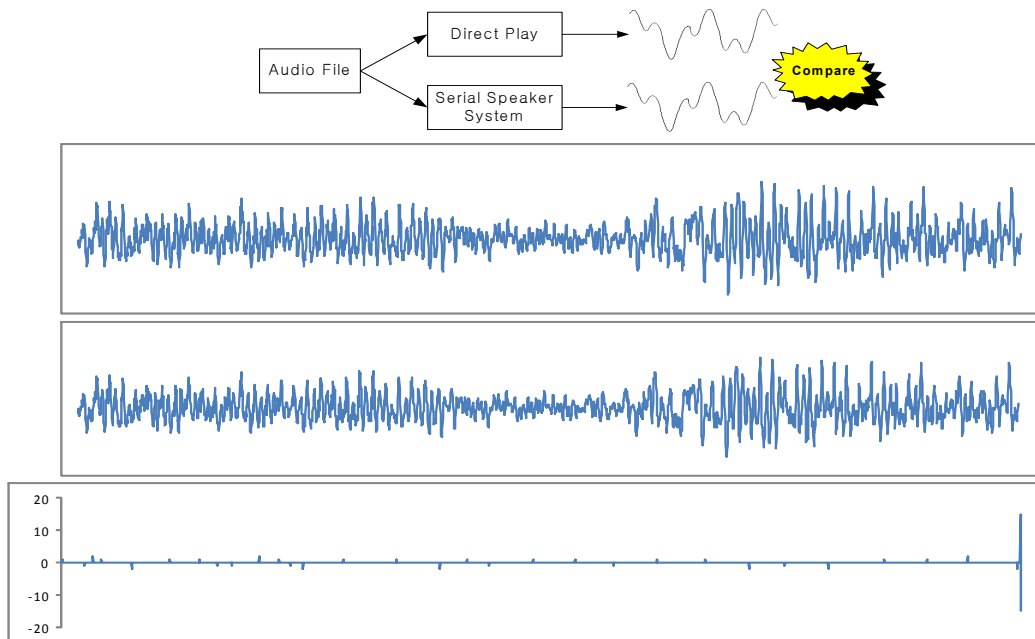
(그림 17) PC 사용자 인터페이스  
(Fig. 17) PC user interface

### 5. 결 론

최근 오디오 시스템은 여러 채널과 많은 수의 스피커를 사용하고 있어서 시스템이 복잡하고 설치 및 관리가 어려울 뿐만 아니라 비용도 많이 들게 된다. 본 논문에서 소개한 시스템은 일반 PC 환경에서 다 채널의 방송을 각 스피커로의 제어가 가능하다. PC는 주제어부에만 연결되어 있고, 주제어부로부터 모든 스피커가 하나의 선으로 연결되어 있기 때문에 쉽게 설치할 수 있을 뿐만 아니라 비용도 저렴한다는 장점이 있다.

비슷한 기능을 가진 코브라넷과 비교하였을 때에 채널의 수 등의 성능은 비슷하게 전송할 수 있다. 또한 제안한 시스템은 구조가 간단하기 때문에 설치 및 유지보수 비용면에서 코브라넷 보다 저렴한 장점이 있다.

사용자는 컴퓨터 GUI 환경을 통해 저장된 여러 음원 파



(그림 16) 원래의 오디오 신호와 재생 신호의 파형 비교  
(Fig. 16) Waveform comparison of original signal and and regenerated signal



〈표 2〉 제안한 시스템과 코브라넷의 성능 비교  
 〈Table 2〉 Performance comparison of proposed system and Cobranet

비교내용	코브라넷	제안 시스템
채널수	64개	64개
해상도	24 bit	24 bit
샘플링 비율	48kHz	48kHz
설치 비용	고가	저가
기반 기술	이더넷 네트워크 기반	음향 압축 기술, 스피커 시리얼 제어 기술
사용 장비	다수의 네트워크 장비	USB 컨트롤러, FPGA
유지보수	어려움	간단함

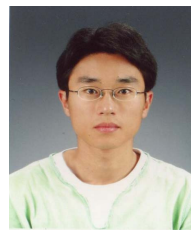
일을 채널별로 선택하여 원하는 스피커에 음원을 재생할 수 있도록 할 수 있다. 예를 들면, 백화점 같은 경우 각각의 층에 맞는 음악을 재생하는 층별 방송이 가능해진다. 따라서 회사, 학교, 병원, 공항 등 스피커가 많이 필요로 하는 곳의 중앙 통제식 방송 시스템에 적합하다.

지금까지는 실험적으로 기능의 확인을 위해서 개발을 하였지만 추가적으로 해야 할 일은 본 논문에서 제시된 시스템의 기능을 완벽하게 구현한 시작품을 만들고 검증하여 실제로 제품화 할 수 있도록 하는 것이다. 또한 이 과정에서 시작품에 많은 수의 채널을 사용하는 경우에도 성능저하가 없는지를 확인하고, 활용 과정에서 추가적으로 필요로 하는 인터페이스에 대해서도 검토 및 구현하여야 한다.

**참 고 문 헌**

[1] Moonvin Song, Doo Young Yi, Ohkyun Kwon, Yunmo Chung, "Serial Connection of Speakers for Multi-channel Audio Systems," Proceedings of IEEE 2003, pp.440-443, Japan, November, 2003.  
 [2] Moonvin Song, Ohkyun Kwon, Yunmo Chung, "A Serial Connection Technique of Speakers for Multi-channel Audio systems" IEEE Transactions on Consumer Electronics. Vol. 51, No.2, May, 2005.  
 [3] 권오균, 송문빈, 이승원, 이영원, 정연모 "다채널 스피커 시스템을 위한 오디오 신호의 직렬전송" 한국음향 학회지, 제24권, 제7호, pp. 387-394, 2005.  
 [4] 이선용, 김태완, 변지성, 송문빈, 정연모 "USB 인터페이스를 이용한 다채널 직렬연결 스피커의 통합 제어", 제15회 한국반도체학술대회 논문집, pp.687, 2008.  
 [5] 최호성, "Windows Programming," FREELEC, pp.587-628, 2006.

[6] Emmanuel C. Lfeachor, Barrie W. Jervis, "Digital Signal Processing" Prentice Hall, pp.727-804, 2002.  
 [7] Jan Axelson, "USB 완전정복", 에이콘 pp.76-81, 2006.  
 [8] <http://www.cobranet.info>



**이 선 용**

e-mail : kyyu1@hanmail.net  
 2007년 경희대학교 전자공학과(학사)  
 2007년 경희대학교 전자공학과 석사과정  
 관심분야: ARM, FPGA, SoC



**김 태 완**

e-mail : wani11@khu.ac.kr  
 2007년 경희대학교 전자공학과(학사)  
 2007년 경희대학교 전자공학과 석사과정  
 관심분야: ARM, FPGA, SoC



**변 지 성**

e-mail : 02-12@hanmail.net  
 2007년 경희대학교 전자공학과(학사)  
 2007년 경희대학교 전자공학과 석사과정  
 관심분야: ARM, FPGA, SoC



### 송 문 빈

e-mail : good52@gmail.com  
1998년 한밭대학교 전자공학과(학사)  
2002년 경희대학교 전자공학과(석사)  
2007년 경희대학교 전자공학과(박사)  
현 재 마린콤(주) 대표이사  
관심분야: ARM, FPGA, RTOS



### 정 연 모

e-mail : chung@khu.ac.kr  
1980년 경북대학교(학사).  
1982년 KAIST(공학석사)  
1982년~1987년 경제기획원 전산처리관  
1992년 미국 미시간주립대학교(공학박사)  
1992년~현 재 경희대학교 전자정보대학  
전자전파공학전공 교수

관심분야: 컴퓨터 구조, ARM, SoC