

# 자동 고장진단이 가능한 스피커 연결 시스템의 SoC 설계

## SoC Design of Self-Diagnosing Speaker Connection System

송 문 빈\*, 권 오 균\*, 송 태 훈\*, 정 연 모\*

(Moonvin Song\*, Ohkyun Kwon\*, The-Hoon Song\*, Yunmo Chung\*)

\*경희대학교 전자공학과

(접수일자: 2007년 7월 24일; 채택일자: 2007년 8월 15일)

디지털 기술의 발전으로 어디서나 음악을 들을 수 있는 다채널 편재형 오디오 시스템의 개발이 구체화 되고 있다. 본 논문에서는 SoC 설계 기술을 기반으로 양방향 디지털 통신을 이용하여 각 스피커를 효율적으로 직렬 연결하는 시스템을 제시한다. 특히 각 스피커는 해당하는 미드 스트림을 확인하여 아날로그 오디오 신호로 변경한다. 또한 스피커는 여러 구형파 테스트 신호의 주파수를 측정하여 스피커 자체의 고장 유무를 진단하는 기능을 가진다. 본 논문에서 제시한 시스템은 200Mhz의 속도로 작동하고 있으며, 기존의 아날로그 방식의 시스템에서는 신호가 직접 출력되지만 500 $\mu$ s 정도의 지연으로 아날로그 신호를 복원하고 있다.

**핵심용어:** 홈-오디오, 고장 진단, 직렬연결, 다 채널, 스피커

**투고분야:** 전기음향 분야 (3.1)

Pervasive Multi-channel audio systems are being realized due to advances in digital technology. This paper proposes an efficient system that serially connects individual speakers with bidirectional digital communication capability by means of SoC design. In particular, each speaker can identify the bit stream assigned to the speaker and convert it into analog audio. Furthermore, the speaker can self-diagnose the speaker functionality by utilizing the designed capability to measure frequencies of various square wave test signals. The proposed system running on 200MHz clock yielded restoration of analog output signal with latency of only 560 $\mu$ s compared to directly driving the speakers in a traditional way.

**Key words:** Home audio, Self-diagnosis, Serial connection, Multi channel, Speaker

**ASK subject classification:** Electro-Acoustic (3,4)

## I. 서론

다양한 음향 기기들이 개발되고 멀티미디어 환경이 보편화 되면서 사용자들의 요구 또한 다양해지고 있다. 따라서 일반 가정에서 사용하는 음향 기기의 수가 많아지고 있기 때문에 사용하는 스피커의 숫자도 날로 늘고 있다. 현재 일반 가정에서 음향 기기들을 사용하는 방법은 [그림 1]과 같다 [1].

디지털 TV, 라디오, CD (Compact Disk) 플레이어, MP3 (MPEG audio layer-3) 플레이어, 게임 콘솔, DVD

(Digital Versatile Disk) 플레이어, 디지털 카메라, 홈시어터, 그리고 PMP (Portable Multimedia Player) 등과 같은 다양한 음향 서비스를 제공하는 장비들이 개별적으로 사용되고 있다.

예를 들어 거실에서 CD 플레이어를 사용하여 들던 음악을 침대 방에서 듣기 위해서는 CD 플레이어를 가지고 이동하거나 CD 만을 가지고 침대 방으로 이동해서 다시 그 곳의 CD 플레이어에서 재생시켜야 한다. 부엌에서 들던 라디오 채널을 거실로 이동하면 계속 들을 수 없다 [2].

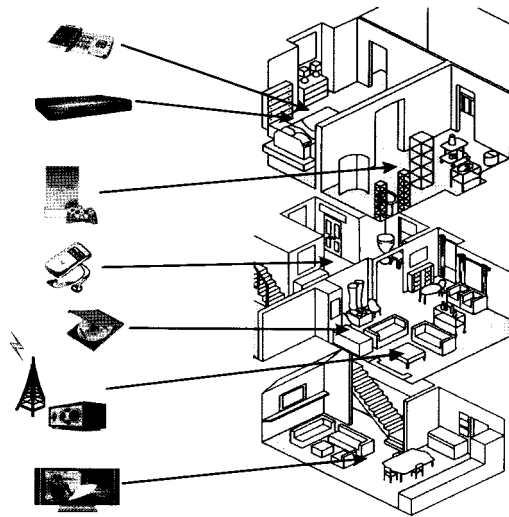


그림 1. 기존의 홈-오디오 환경  
Fig. 1. Present home-audio environment.

이런 단점을 보완하기 위해서 본 논문에서는 [그림 2]와 같이 가정 내에서 사용하는 다양한 음원들을 중앙에서 통합으로 제어하고, 각 방의 모든 스피커들을 하나의 선으로 연결하여 다양한 채널의 방송을 사용자가 원하는 대로 제어하고 자동으로 스피커의 고장을 진단하는 기능을 갖는 스피커 직렬연결 홈-오디오 시스템의 구조를 제안하고 구현하였다.

집안 내에 존재하는 다양한 음원들은 중앙 제어기에서 각 채널별로 통합하고 관리한다. 가정 내의 각 공간에는 스피커를 설치하고, 모든 스피커들은 제어기에서 하나의 연결선으로 연결한다. 제어기는 동시에 여러 채널의 음향 신호를 원하는 스피커에 선택적으로 보낼 수 있다. 제어기에서 특정 스피커의 고장을 진단하기 위해서 고장 진단 명령을 보내면 해당 스피커에서는 자동으로 고장을 진단하여 결과를 제어기로 전달한다.

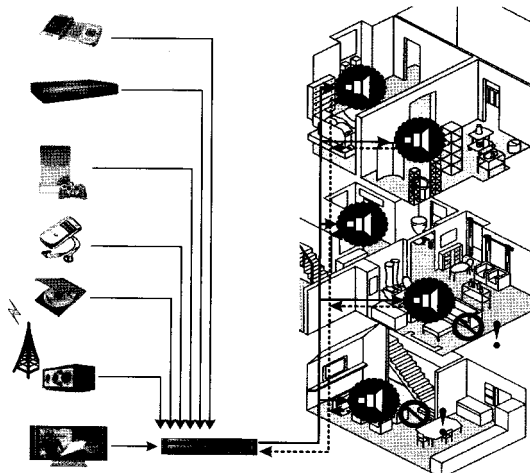


그림 2. 홈-오디오 스피커 직렬연결 시스템 개념  
Fig. 2. Home-audio serial connection speaker system.

제어기에서 스피커 까지는 신호를 패킷화된 디지털 데이터를 UTP (Unshielded Twisted-Pair) 케이블을 사용하여 전송한다. UTP 케이블은 일반적인 LAN (Local Area Network)을 설치할 때 사용하는 것으로 보편화되어 있고 가격이 저렴한 장점을 가지고 있다. 따라서 설비 비용을 대폭 줄일 수 있다 [3].

## II. 시스템 구조

SoC 기반의 자동 고장진단 기능을 갖는 스피커 직렬 연결 홈-오디오 시스템의 전체적인 개념은 [그림 3]과 같다.

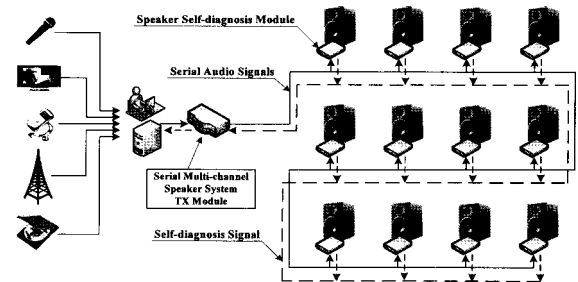


그림 3. 전체 시스템 구조  
Fig. 3. System architecture.

시스템은 여러 채널의 마이크, 공중파 채널, 그리고 MP3 와 CD 또는 DVD와 같은 디지털 저장 매체와 같은 다양한 음향 신호원은 동시에 사용할 수 있다. 여러 채널의 음향 신호는 중앙 제어장치에서 하나의 연결선을 사용하여 동시에 여러 개의 스피커들을 제어한다. 각 스피커에서는 사용자의 요구에 따라서 자동으로 고장을 진단한 후 결과를 알려 준다.

### 2.1. 데이터 변환

모든 스피커를 하나의 선에 연결하고 제어하기 위해서는 아날로그 신호로 입력되는 채널의 입력 신호를 공통된 포맷의 디지털 신호로 변환하여 전송하고 처리해야 한다. 각 스피커 에서도 자동으로 고장을 진단하는 기능은 디지털 구형파를 사용하여 주파수를 측정하는 방식을 사용한다. 시스템에서 전체적인 데이터의 변환 과정은 [그림 4]와 같다.

각 채널별로 입력되는 음향 신호는 채널 1, 2와 같은 아날로그 신호 이거나 채널 3과 같이 디지털 음원이다. 아날로그 신호의 경우에는 음향신호 전용 ADC (Analog Digital Converter)를 사용하여 [그림 5]와 같은 CD 플

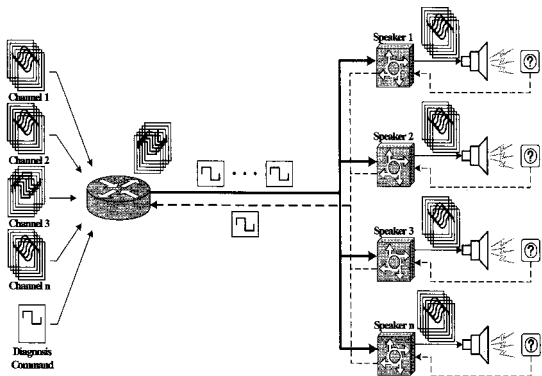


그림 4. 전체적인 데이터 변환  
Fig. 4. Data conversion.

레이어, DSP, 디지털 TV 음향 등과 같은 디지털 오디오 장치 및 기술을 위한 직렬 버스 신호인 I2S (Inter-IC Sound) 포맷으로 변환한다.

디지털로 샘플링을 수행한 각 채널별 음향 신호는 압축 과정을 거쳐서 패킷으로 만들어 진다. 이 때 특정 스피커의 고장을 진단하기 위한 명령어를 포함 할 수 있다. 패킷 신호는 모든 스피커에 전달된다.

각 스피커에서는 전달된 패킷 신호의 채널을 분석하여 해당하는 경우에 압축을 해제하는 과정을 거쳐서 원래의 음향 신호를 복원하여 DAC (Digital Analog Converter) 를 사용하여 스피커로 음향을 출력한다. 만일 패킷이 고장 진단 명령어를 가지고 있으면 스피커는 진단 모드로 변환되어 고장 진단 기능을 수행한다. 고장 진단은 특정 패턴의 신호를 스피커로 출력하고 이를 진동 센서로 검출한 후 비교하는 과정을 거친다.

디지털 오디오 직렬 버스 신호인 I2S에서 오디오 데이터인 SDATA는 클럭 신호인 LRCK, SCLK와 분리하여 처리한다. I2S는 두 채널의 음향 신호만을 표현할 수 있으며 LRCK는 두 채널을 분리하는 신호이다. 즉, LRCK가 '0'일 때는 첫 번째 채널의 신호를 나타내고, '1'일 때는 두 번째 채널의 신호를 나타낸다. SDATA는 SCLK에 동기를 맞추어 데이터를 출력한다. LRCK의 상태가 바뀌고 다음 SCLK의 하강 에지부터 음향 신호의 최상위 비트인 MSB (Most Significant Bit)를 출력하기 시작해서 매 하강 에지마다 한 비트씩 출력한다.

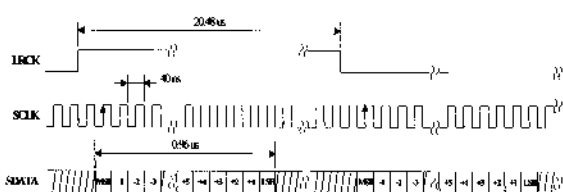


그림 5. I2S 버스 포맷  
Fig. 5. I2S bus format.

SDATA는 사용하는 모드에 따라서 다양한 크기를 가질 수 있다. 따라서 SDATA가 음향 데이터의 최하위 비트인 LSB (Least Significant Bit)를 출력하고부터 LRCK의 상태가 바뀔 때 까지는 유용한 데이터가 없다. 이렇게 출력한 MSB에서 LSB까지의 데이터는 한 채널의 음향 신호를 한 번 샘플링을 수행한 결과이다. 따라서 두 채널이 하나의 I2S를 사용한다.

## 2.2. 패킷 생성

하나의 전송 선로를 사용하여 동시에 여러 채널의 음향 신호를 동시에 보내기 위해서는 디지털로 변환한 음향 신호를 압축하고 채널별로 여러 종류의 패킷을 만들어야 한다. 따라서 I2S 신호를 가지고 [그림 6]과 같은 방법을 사용하여 직렬로 전송하기 위한 패킷을 만든다.

아날로그 음향 신호는 [그림 6]의 상단과 같이 연속적인 신호이고, 해당하는 샘플링 주파수 간격으로 샘플링을 수행하여 디지털 데이터 값을 얻는다. 패킷 데이터는 기본적으로 직전에 샘플링한 데이터와 현재 샘플링된 데이터를 xor 하여 만든다. 음향 신호의 특성상 샘플링한 데이터들의 변화량이 일반적으로 몇 비트에 지나지 않는다. 따라서 두 값으로 xor 연산을 수행한 결과를 분석하면 샘플링 값이 같은 MSB 영역은 모두 '0' 값을 가지고 있고 실제로 데이터가 변한 부분부터 '1'로 시작하는 것을 알 수 있다.

예를 들면, 24 비트의 크기로 음향 신호를 샘플링 한다고 가정하고 직전에 샘플링한 데이터가 "0000 0000 0000 0010 1111 1111" 이고 현재 샘플링한 데이터가 "0000 0000 0000 0010 1111 0011"이면, 두 값을 비트별로 xor 연산을 수행한 결과는 "0000 0000 0000 0000 1100"이다. MSB 부분부터 한 비트씩 살펴보면 '0'인 부분은 음향 데이터가 변하지 않은 부분이고 처음으로 '1'이

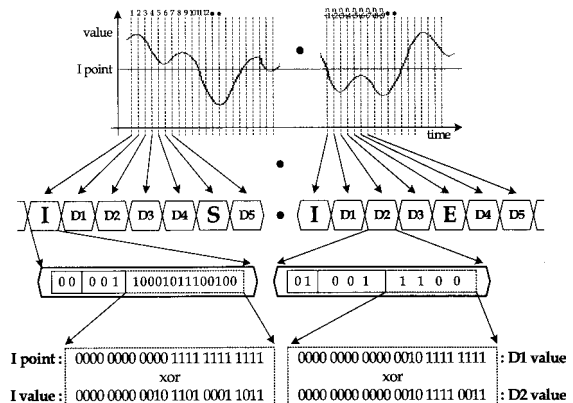


그림 6. 패킷 생성 방법  
Fig. 6. Packet creation method.

나오는 비트가 변화량을 나타낸다. 즉, 하위 네 개의 비트 값인 "1100"이 변화량에 해당한다. 따라서 패킷에서 데이터는 "1100"이다.

I point는 임의의 기준 값이다. 처음으로 시작하는 패킷의 경우에는 xor 연산을 수행하기 위해 비교할 직전에 샘플링한 데이터가 없으므로 임의의 기준 데이터가 필요하다. 이 때 I point와 xor 연산을 수행하고, 그 결과로 생성한 패킷을 I 패킷이라고 한다. D 패킷은 실제적인 직전의 데이터와 xor 연산을 수행하여 생성한 패킷이다.

전송 도중에 손상된 패킷이 있는 경우에는 이후에 전달되는 D 패킷은 유효한 정보를 복원하지 못하는 단점을 가지고 있다. 이러한 단점을 보완하기 위해 임정한 조건에 따라서 I 패킷을 전송한다. I 패킷은 직전 데이터와의 의존성이 없기 때문에 독립적으로 원래의 데이터를 복원할 수 있다.

패킷은 I 패킷, D 패킷, E 패킷, 그리고 S 패킷의 네 가지 종류가 있고 [그림 7]과 같은 구조를 갖는다.

패킷은 패킷의 종류를 나타내는 두 비트의 State와 특정 스피커를 지정하기 위한 Channel number, 그리고 실제적인 압축 데이터인 Channel data의 세 부분으로 구성한다. Channel number는 시스템에서 사용하는 스피커의 수에 따라 가변적인 값을 갖는다.

I 패킷은 기존 패킷에 의존성이 없는 독립적인 압축 데이터를 가지고 있으며 State가 "00"이다. D 패킷은 "01"로 시작하며 직전의 I 패킷이나 D 패킷의 데이터 값에 의존적이다. S 패킷은 자동 고장 진단을 명령하는 것으로 해당 채널만을 설정하고, Channel data 부분을 가지고 있지 않다. E 패킷은 "11"로 시작하며 비상시 전체 스피

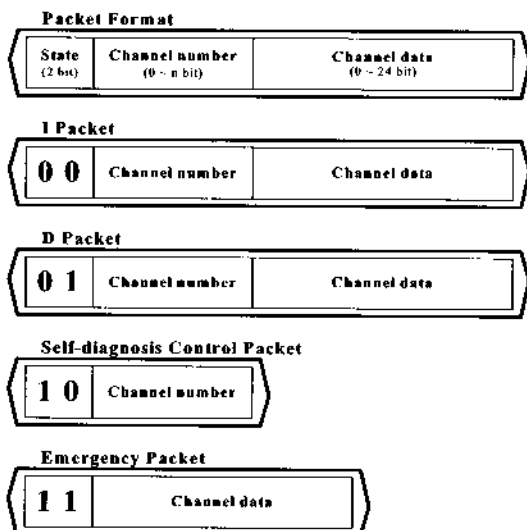


그림 7. 패킷의 구조 및 종류  
Fig. 7. Packet architecture and type.

커에 비상 방송을 위한 것으로서 Channel number가 없고 압축을 수행하지 않는다. 따라서 Channel data는 I, D 패킷과는 다르게 원래의 음향신호 데이터이다.

### III. 시스템 구현

#### 3.1. 전송 제어기 구조

전송 제어기는 [그림 8]과 같은 구조를 가지고 있으며 각 채널에서 실시간으로 입력되는 데이터들을 처리하여 각종 패킷으로 만들고 제어 신호에 따라 시스템을 제어한다. 그리고 전송선로의 부하를 검출하여 I 패킷의 전송 횟수를 제어한다.

Channel Separation 블록에서는 아날로그로 입력되는 여러 채널의 데이터를 I2S 포맷으로 변환한 데이터, 디지털 오디오용 인터페이스의 한 형식인 S/PDIF (Sony Philips Digital Interface)와 I2S 데이터, 그리고 비상 방송이나 고장 진단을 명령하기위한 Control Signal 입력 신호를 받아들인다. 여러 개의 데이터들은 채널별로 분리하여 채널 정보와 데이터를 다음 블록으로 전달한다.

D Packet Data Generator 블록에서는 메모리 블록에 저장한 직전에 생성된 데이터와 비교해서 압축한 정보를 만들고 여기에 패킷 정보를 추가하여 D 패킷을 생성한다. I Packet Data Generator 블록에서는 I point와 현재의 데이터를 가지고 I 패킷을 생성한다. E Packet Data Generator 블록은 Channel number가 필요 없고 압축 과정을 거치지 않은 원래의 음향 데이터에 제어 신호만을 추가하여 E 패킷을 만든다.

Header Generator 블록에서는 [그림 7]의 패킷 구조에서 전송할 패킷에 맞는 Status와 Channel number를 붙여서 완벽한 패킷을 만들어 전송하는 서리얼로 변환하여 출력한다. 이렇게 채널별로 만들어진 데이터들은 Signal Generator에서는 각 채널별 패킷의 전송 순서를 제어한다.

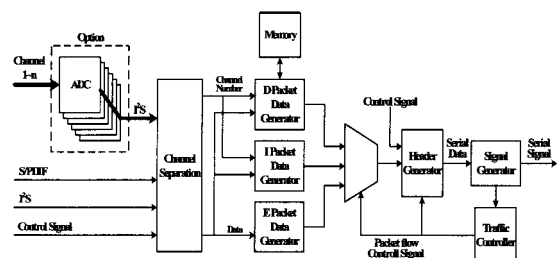


그림 8. 전송 제어기 블록 다이어그램  
Fig. 8. Transmission controller block diagram.

D 패킷, I 패킷, 그리고 E 패킷을 모두 만든 후에 Traffic Controller 블록에서는 Signal Generator 블록에서 발생하는 전송량을 분석하여 I 패킷의 전송을 제어한다.

구현한 디지털 앰프의 주파수 응답이 아날로그 앰프나 기존의 디지털 앰프보다 가청 주파수 전 대역에서 균일하다.

### 3.2. 수신 제어기

수신 제어기는 스피커마다 존재하며 원래의 음향신호를 복원하여 스피커를 제어하거나 자동으로 고장을 진단하는 기능을 수행한다. 따라서 그림 9와 같은 순서에 따라서 수신한 패킷의 Status와 Channel number를 판별하여 동작한다.

한 비트씩 입력되는 데이터를 수신 제어기에서는 패킷으로 만든 후에 제일 먼저 패킷의 State 부분을 분석하여 패킷의 종류를 판별한다. I, D 패킷의 경우에는 Number 부분을 분석하여 해당 데이터 인지를 판단한다. 맞으면 복원 과정을 거쳐 원래의 채널을 복원하고 ADC를 사용하여 스피커로 음향 신호를 재생한다.

S 패킷은 고장 진단을 명령하는 모드이다. 자동으로 고장을 진단하기 위해서 특정한 패턴의 신호를 스피커로 출력하고 진동센서를 사용하여 다시 원래의 신호와 비교하는 과정을 거쳐서 고장 유무를 판별한다. 비상 방송을 위한 E 패킷은 압축되지 않은 데이터 이므로 바로 원래 신호를 빼내어 ADC로 보낸다.

### 3.3. 패킷 복원

$A \text{ xor } B = C$  라고 가정 하면 A, B, 그리고 C의 위치를 서로 바꾸어도  $B \text{ xor } C = A$  와  $A \text{ xor } C = B$ 와 같다. 즉, xor 연산은 수식부의 두 값과, 결과 값을 어디에 놓아도 된다는 특성을 활용하여 I와 D 패킷은 원래의 데이터를

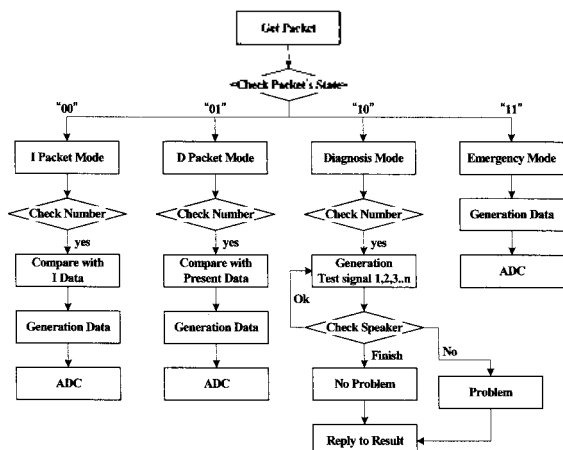


그림 9. 수신 제어기 동작 플로 차트  
Fig. 9. Receive controller flow-chart.

표 1. 복원 과정 예

Tab. 1. Example of restore process.

	Expression	Result
Compression	I point xor Data	I Packet
Decompression	I Packet xor I Point	Data

복원하는 과정을 [표 1]과 같이 역으로 xor 연산을 수행하면 된다.

전송부에서 데이터를 패킷으로 변환하는 과정에서 압축은 현재 샘플링한 데이터와 I 패킷은 I point와 D 패킷 일 경우에는 직전에 샘플링한 데이터와 xor 연산을 수행한 결과를 가지고 만든다. 따라서 복원 과정은 I 패킷의 예를 들면 I point와 I 패킷을 xor 하면 된다.

### 3.4. 고장 진단

스피커 고장 진단 명령어를 검출한 스피커의 제어기는, 여러 주파수 대역으로 구성된 특정 패턴의 디지털 구형파를 스피커로 출력한다. 주파수가 같은 아날로그 신호인 정현파와 디지털 신호인 구형파를 스피커로 출력하는 경우에는 신호의 유형에 관계없이 동일한 소리를 출력한다. 본 논문에서 구현한 스피커의 자동 고장 진단 기능은 [그림 10]과 같이 구형파와 정현파의 주파수를 비교하는 방법을 적용하여 구현한다 [4,5].

DAC의 입력으로 디지털로 샘플링한 정현파 데이터를 사용하면 특정한 주파수를 갖는 아날로그 정현파를 얻는다. 이 신호를 schmitt-trigger를 통과 시키면 입력으로 사용한 정현파와 동일한 주파수를 갖는 구형파를 만들 수 있다.

DAC의 입력으로 사용할 수 있는 최솟값을 '1', 그리고 최댓값을 '0' 이라고 하면 발생 시키고자 하는 주파수에 '0'과 '1'을 반복적으로 사용한다. 그러면 DAC는 특정한 주파수를 갖는 디지털 구형파를 출력한다.

이렇게 만든 두 구형파를 각각 카운트 하여 비교하면 주파수의 차이를 간단하게 비교할 수 있다. 위와 같은 원리를 이용하여 [그림 11]과 같이 스피커의 자동 고장 진단을 구현한다.

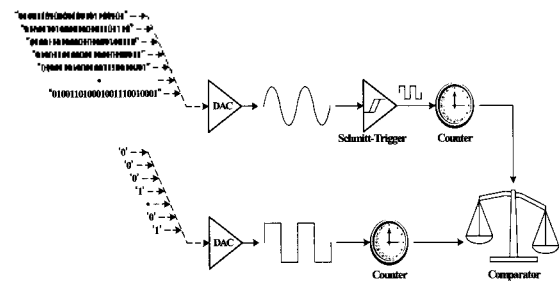


그림 10. 구형파와 정현파의 주파수 비교  
Fig. 10. Compare with sign and square wave.

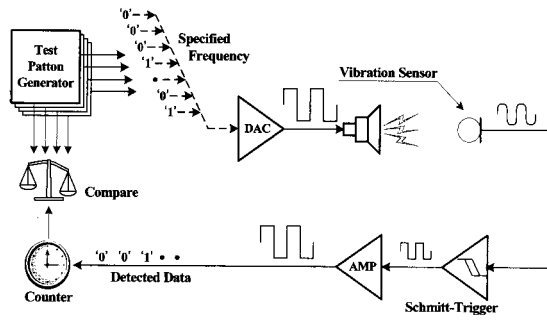


그림 11. 스피커 자동 고장 진단 방법  
Fig. 11. Speaker auto diagnosis method.

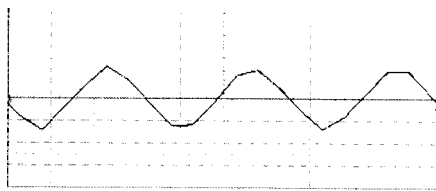


그림 12. 진동 센서 출력 파형  
Fig. 12. Vibration sensor output signal.

논문에서 구현한 스피커 시스템은 초기에 입력되는 아날로그 음원 부분을 제외하면 시스템 내부에 전체적으로 디지털 신호만 존재한다. 따라서 고장을 진단하기 위해서 스피커를 구동 시키려면 DAC를 사용하여 디지털 신호를 입력해야한다.

가청 주파수 대역은 20 Hz ~ 20 KHz 이지만 스피커의 고장을 진단하기 위해서 초대역은 사용할 필요가 없다. 따라서 Test Pattern Generator에서 50 Hz ~ 15 KHz 대역의 주파수를 일정한 패턴으로 발생 시켜서 DAC를 통해 발생하는 구형파로 스피커를 구동한다. 스피커로 출력되는 음향 신호를 진동 센서를 사용하여 검출하면 [그림 12]와 같이 구형파의 모서리가 심하게 왜곡된 파형을 얻을 수 있다 [6,7].

센서에서 검출한 신호에 schmitt-trigger 회로를 사용하여 임계값을 적용하여 구형파를 만든다. 이 구형파의 크기를 앰프를 사용하여 TTL (Transistor Transistor Logic) 수준의 신호 수준으로 증폭하여 카운트 하면 주파수를 계산할 수 있다. 이렇게 검출한 주파수와 Test Pattern Generator의 주파수와 비교하여 스피커의 고장 유무를 판단한다.

#### IV. 성능 분석

구현한 시스템은 다 채널의 음향 신호를 제어하기 위하여 [표 2]와 같은 사양을 가지고 있다. 아날로그로 입력

표 2. 구현 사양  
Tab. 2. Development specifications.

개발 내용	사양	비고
디지털 샘플링 방법	I <sup>2</sup> S	디지털 변환 방식
샘플링 비율	44.1 KHz	디지털 CD 수준
샘플링 크기	24 bit	채널당
샘플링 주파수 대역	20Hz - 20KHz	가청 주파수 대역
동시 전송 채널 수	23 개	
최고 전송 속도	10 Mbit/sec	

되는 채널의 음향신호는 ADC를 사용하여 24 비트의 크기를 갖는 CD 수준인 44.1KHz로 샘플링 하여 I2S 신호로 변환하여 시스템에서 처리한다. 샘플링 주파수 대역은 가청 주파수인 20Hz - 20KHz 이다. 동시에 23 개의 채널이 전송 가능하고, 사용 가능한 스피커의 숫자에는 제한이 없다.

시스템의 송수신에 따른 왜곡 및 전송 지연을 측정하기 위해서 [그림 13]과 같은 burst 신호를 한 채널의 입력으로 사용하여 전송 후에는 [그림 14]와 같은 결과를 얻었다. Burst 신호는 5 주기 동안 신호를 발생하고 20 주기 동안 신호를 발생시키지 않는 패턴을 가지고 있다.

Burst 신호는 전송 제어기의 ADC에서 I2S 신호로 샘플링 되어 패킷으로 만들어진다. 패킷을 전송 선로를 통하여 수신 제어기에서 복원 과정을 거쳐 [그림 14]와 같이 560 μs의 지연을 갖는 동일한 파형을 출력한다.

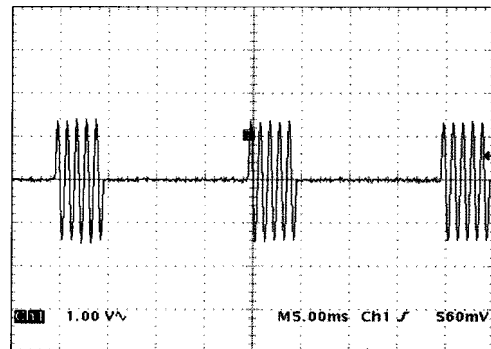


그림 13. 테스트 입력 신호  
Fig. 13. Test input signal.

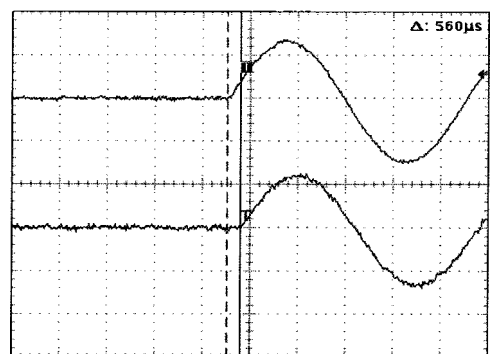


그림 14. 입, 출력 파형 비교  
Fig. 14. Compare with input and output signal.

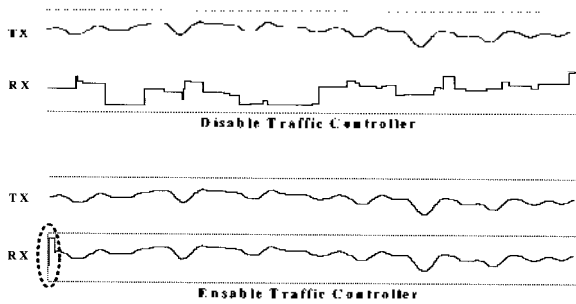


그림 15. Traffic Controller 적용 결과  
Fig. 15. Result of using traffic controller.

여러 채널의 입력 신호를 동시에 처리하는 경우에는 패킷의 종류에 따라서 사용하는 전송 선로의 대역폭 보다 전송할 데이터의 양이 큰 경우가 발생한다. 전송 상태를 확인하기 위하여 전송 선로상의 데이터를 아날로그 파형으로 나타내고, 표 1(그림 8)과 같은 전송 제어기 구조의 Traffic Controller를 적용한 결과는 [그림 15]와 같다.

Traffic Controller가 동작하지 않는 경우에는 전송한 결과가 심하게 왜곡되었다. 그러나 Traffic Controller가 동작하면서 초기의 시스템 안정화 시간을 거치 이후에는 정상적으로 전송되는 것을 알 수 있다.

## V. 결론

본 논문에서 제시하고 구현한 SoC 기반의 자동 고장진단 기능을 갖는 스피커 직렬연결 홈-오디오 시스템은 날로 다양해지는 멀티미디어 서비스 환경에서 가정 내의 음원들을 중앙에서 통합으로 제어하고 사용하는 모든 스피커를 하나의 선으로 연결하여 다양한 채널의 방송을 사용자가 원하는 대로 제어하는 기술이다.

스피커의 자동 고장진단 기능은 오늘날과 같이 가정에 다양한 멀티미디어 기기가 보급되면서 많은 수의 스피커를 동시에 사용하는 상황에서 유용한 기술이다. 동시에 22 개 채널을 제어할 수 있으며 자동으로 스피커의 고장을 진단할 수 있다.

## 감사의 글

본 논문은 2006년도 경희대학교 지원에 의한 결과임.

## 참고 문헌

1. Moonvin Song, Ohkyun Kwon, Yunmo Chung, "A Serial Connection Technique of Speakers for Multi-channel Audio systems" IEEE Transactions on Consumer Electronics. 51 (2), May 2005.
2. B. Forouzan, *Data Communications and Networking Third Edition*, (McGraw Hill, 2003)
3. C. Busbridger, Y. Huang, and P. A. Fryer, "Crossover Systems in Digital Loudspeakers," AES Journal, 50 (10), 791, 2002.
4. K. Parfi, *VLSI Digital Signal Processing Systems : Design and Implementation*, (Material)
5. P. Rashinkar, P. Paterson, L. Singh, *System-On-a-Chip Verification Methodology and Techniques*, (Kluwer Academic Pub)
6. W. Wolf, *Modern VLSI Design Third Edition*, (Prentice Hall PTR, 2002)
7. X. Bosun, "Signal Mixing for a 5.1-Channel Surround System - Analysis and Experiment," AES Journal, 49 (4), 263, 2001.

## 저자 약력

### • 송 문 빈 (Moonvin Song)



1998년 : 한밭대학교 전자공학과  
2002년 : 경희대학교 전자공학과 (공학석사)  
2007년 : 경희대학교 전자공학과 (공학박사)  
2005년~현재 : (주)휴인스 책임연구원  
2007년~현재 : 경희대학교 전자정보대학 겸임교수

### • 권 오 균 (Ohkyun Kwon)



1986년 : 숭실대학교 전자공학과 (공학사)  
1997년~현재 : 한국스프라이트 대표  
2002년 : 경희대학교 전자공학과 (공학석사)  
2003년~현재 : 경희대학교 전자공학과 박사과정

### • 송 태 훈 (The-Hoon Song)



1985년 : 충북대학교 공학사.  
1990년 : 삼보컴퓨터 연구원.  
1993년 : 한국산업기술대학원 공학석사.  
현재 : 경희대학교 전자공학과 박사과정  
(주)휴인스 대표이사.

### • 정 언 모 (Yunmo Chung)



1980년 : 경북대학교 졸업  
1982년 : KAIST (공학석사)  
1982년~1987년 : 경제기획원 전산처리관  
1992년 : 미국 미시간주립대학교(공학박사)  
1992년~현재 : 경희대학교 전자정보대학 교수