

# DSP를 이용한 가라오케용 고음질 멀티채널 오디오 시스템

## High Quality Multi-Channel Audio System for Karaoke Using DSP

김 태 훈\*, 박 양 수\*, 신 경 철\*, 박 종 인\*\*, 문 태 정\*

(Tae-Hoon Kim\*, Yang-Su Park\*, Kyung-Chul Shin\*, Jong-In Park\*\*, Tae-Jung Moon\*)

\*경남정보대학, \*\*(주)금영 부설연구소

(접수일자: 2008년 10월 29일; 수정일자: 2008년 12월 5일; 채택일자: 2008년 12월 18일)

본 논문에서는 멀티채널 라이브 가라오케의 구현에 관한 내용을 담고 있다. TI사의 32비트 floating 연산 DSP인 TMS320C6713를 이용하여 6 채널의 MP3 복호화 및 템포/키 변환을 실시간으로 구현하였다. 6채널은 전면 L/R 악기, 후면 L/R 악기, 멜로디, 우퍼로 구성되며, 4 채널로 동작 시에는 후면 L/R 대신 드럼 L/R이 추가될 수 있다. 최종 출력 데이터는 5.1 채널 스피커에 맞춰서 출력된다. 템포 변환을 위하여 SOLA 알고리즘을 적용시켰으며 시간영역에서 인터폴레이션(interpolation)과 데시메이션(decimation)으로 키 변환을 수행하였다. 드럼 악기가 추가될 경우에는 일반악기와 분리하여 키 변환 시에 드럼 채널을 제외시키고, SOLA (Synchronized Overlap and Add) 수행 시에도 SOLA 처리 단위인 프레임 사이즈를 다르게 하여 고음질의 템포 변환이 가능하도록 하였으며, 실시간 처리를 위하여 최적화를 하였다. 6 채널을 이용하여 다양한 채널 구성이 가능하며 본 논문의 멀티채널 오디오 시스템은 고음질의 라이브 만주가 필요한 어느 곳에서나 효과적으로 적용될 수 있다.

**핵심용어:** 가라오케, MP3, 키 변환, 템포 변환, 멀티채널 오디오.

**투고분야:** 전기음향 분야 (3)

This paper deals with the realization of multi-channel live karaoke. In this study, 6-channel MP3 decoding and tempo/key scaling was operated in real time by using the TMS320C6713 DSP, which is 32 bit floating-point DSP made by TI Co. The 6 channel consists of front L/R instrument, rear L/R instrument, melody, and woofer. In case of the 4 channel, rear L/R instrument can be replaced with drum L/R channel. And the final output data is generated as adjusted to a 5.1 channel speaker. The SOLA algorithm was applied for tempo scaling, and key scaling was done with interpolation and decimation in the time domain. Drum channel was excluded in key scaling by separating instruments into drums and non drums, and in processing SOLA, high-quality tempo scaling was made possible by differentiating SOLA frame size, which was optimized for real-time process. The use of 6 channels allows the composition of various channels, and the multi-channel audio system of this study can be effectively applied at any place where live music is needed.

**Keywords:** Multi-channel, Mp3, Karaoke, Tempo, Pitch, Audio

**ASK subject classification:** Electro-Acoustics (3)

### I. 서론

가라오케 시스템은 흔히 노래만주 시스템이다. 노래 만주 시스템은 사용자가 원하는 노래를 선택하면 저장된 노래 정보를 이용하여 사용자가 쉽게 노래를 부를 수 있

도록 가사와 노래 만주뿐만 아니라 키와 템포를 조절할 수 있도록 해주는 시스템이다. 이는 매년 많은 새로운 노래가 미디어를 통하여 발표되며, 이러한 노래 가운데 많은 사람들은 원하는 노래를 배우고 부를 수 있도록 해 준다. 이러한 가라오케 시스템은 가정과 노래방 등에서 널리 사용되고 있다. 가라오케 시스템은 사용자의 요구 수준의 증가로 인하여 보다 좋은 환경을 가진 만주기의 개발이 계속 진행되고 있으며, 실제 가수가 사용하는 반

책임저자: 김 태 훈 (tae-hoon@cagle.kit.ac.kr)  
617-701 부산광역시 사상구 주례2동 167번지  
경남정보대학 전자정보계열  
(전화: 051-320-2808; 팩스: 051-320-2831)

주와 같은 고음질과 라이브 연주, 멀티채널 등 사용자의 요구가 날로 증가되고 있는 실정이다.

현재 대부분의 가라오케 시스템은 주로 MIDI (Musical instrument digital interface) 정보를 기반으로 한다. MIDI에는 곡의 연주 정보를 포함하고 있으며, 이러한 시스템은 FM (Frequency Modulation) 방식으로 음을 합성하거나, 샘플링 (Sampling)된 음악 데이터가 ROM 형태로 저장된 음원 ROM을 이용하여 합성된 음악을 반주로 이용한다. 이 경우 저장된 음원이 풍부할수록 더 좋은 음을 합성할 수 있지만, 음원을 저장하기 위한 메모리와 합성 알고리즘의 한계로 실제 연주와 같은 고음질의 반주음을 합성하기 어렵다. 또한, 음원 칩의 성능에 따라 동시에 재생 가능한 악기 수와 효과음도 제한을 받는다. 또한 노래 부르기 쉽게 해 주는 코러스와 원곡만의 독특한 효과음을 재현하기 위해서는 별도의 프로세서 및 하드웨어가 요구되며, 이 또한 사용자의 요구를 충족시키기에는 미흡하다. 또한 이러한 코러스 등은 녹음 후 단순 재생에 지나지 않으며, 대부분 키, 템포 변환을 지원하지 않는다. 이러한 한계는 실제 라이브 연주를 녹음하여 반주 시 재생함으로써 극복될 수 있다. 라이브 연주를 이용하면 코러스 및 다양한 효과음 요구를 충족시킬 수 있으며 멀티채널로 녹음하면 보다 좋은 공간감을 줄 수 있다.

본 논문은 이러한 고음질 라이브 반주기 시스템에 관한 것으로 멀티채널로 녹음된 실제 라이브 반주, 코러스, 멜로디를 재생하여 반주로 사용하는 시스템에 관한 내용을 소개한다. 또한 재생 시에 사용자의 요구에 따라 고음질 키, 템포 변환을 수행할 수 있다. 고음질의 반주를 멀티채널로 저장하기 위해서는 많은 저장 공간이 필요하다. 44.1 KHz의 6 채널, 16비트 데이터는 1초에 약 4.2 Mbps가 필요하다. 이러한 대용량의 필요성을 개선하기 위하여 현재 널리 사용되고 있는 오디오 압축 기술인 MP3을 이용하여 저장 용량을 줄였다. 또한, 사용자에게 맞는 편안한 반주를 위하여 가라오케 시스템의 필수 기능인 키/템포 변환 기능을 위하여 SOLA (Synchronized Overlap and Add) 알고리즘과 시간영역에서 인터폴레이션, 데시메이션을 적용하였다. 또한 실제 반주에 가까운 고음질 키/템포 변환을 위하여 드럼과 드럼을 제외한 일반악기로 분리하여 처리할 수 있는 기능을 추가하였다. 구현된 채널의 수는 6개이며, 이러한 멀티채널 가라오케 시스템은 고음질 반주를 위하여 널리 이용될 수 있다. 본 논문은 2장 기존의 가라오케 시스템, 3장 MP3 알고리즘 및 SOLA 알고리즘, 4장에서 멀티채널 라이브 가라오케 시스템, 5장에서 결과, 그리고 6장에서 결론으로 구성되어 있다.

## II. 기존가라오케 시스템

가라오케 시스템에는 다양한 장르의 많은 곡 정보가 저장되어 있으며 사용자가 부르기 원하는 노래를 선택하면, 이 곡의 가사를 표시하면서 반주해준다. 사용자는 표시된 가사를 보면서 반주에 맞춰서 마이크를 통하여 노래한다. 그리고 사용자가 원하는 템포와 키를 조절할 수 있게 하여 다양한 형태의 반주음을 만들어 준다. 또한 좀 더 실감 나는 연주를 위하여 많은 스피커를 두어서 멀티채널을 지원하며, 이 경우 전면, 후면, 멜로디 채널로 구성될 수 있다. 현재 대부분 반주정보는 미디를 이용한다. 미디 파일에 가사, 채널, 악기 선택과 연주 정보가 포함된다. 미디를 이용하면 곡당 적은 저장 용량만 필요하며, 합성 칩으로 반주음을 만들어 준다. 합성 음질은 합성 칩의 성능에 따라서 결정된다. 이러한 합성을 위하여 실제 악기음을 일부 저장하는 경우도 있으나, 저장 용량의 한계로 인하여, 합성된 반주는 라이브 반주에 비해서 음질이 나쁠 수밖에 없다. 또한, 합성할 수 있는 악기수도 합성 칩의 성능에 의하여 제한되고, 합성할 수 있는 효과음이 미리 정해져 있으며, 코러스 등 각각의 반주에 맞는 효과를 낼 수 없다. 따라서 이러한 기존의 가라오케 시스템으로는 현재의 요구사항을 충족시키기에는 미흡하다고 볼 수 있다.

## III. MP3 및 SOLA 알고리즘

### 3.1. MP3 알고리즘

MP3 오디오는 현재 널리 사용되는 있는 대표적인 디지털오디오 압축 기법이다. MPEG 오디오코딩의 기본이 되는 심리음향 모델은 오디오 신호에 여러 주파수 성분이 있을 때 특정 주파수 성분이 주변의 주파수 성분에 의해 마스킹 (Masking)되어 사람의 귀에 들리지 않는 현상을 이용하므로 압축률을 높일 수 있다 [1][2]. 이러한 MP3 (MPEG-1 audio layer-3)은 높은 압축률과 고음질을 가진다. 이러한 이유로 MP3 방식은 가라오케에 사용되는 코러스와 라이브 반주의 압축 방식으로 적합하다. 다양한 전송률 (bit rate)로 압축이 가능하며 고음질이 필요한 경우 스테레오 채널을 320 kbps까지 코딩 하면 원곡과 거의 같은 수준의 고음질을 유지할 수 있다. 그림 1에 MP3 복호화 (decoding) 과정을 나타내고 있다. MP3 비트스트림 (bit-stream)이 들어오면 디멀티플렉싱 (demultiplexing)

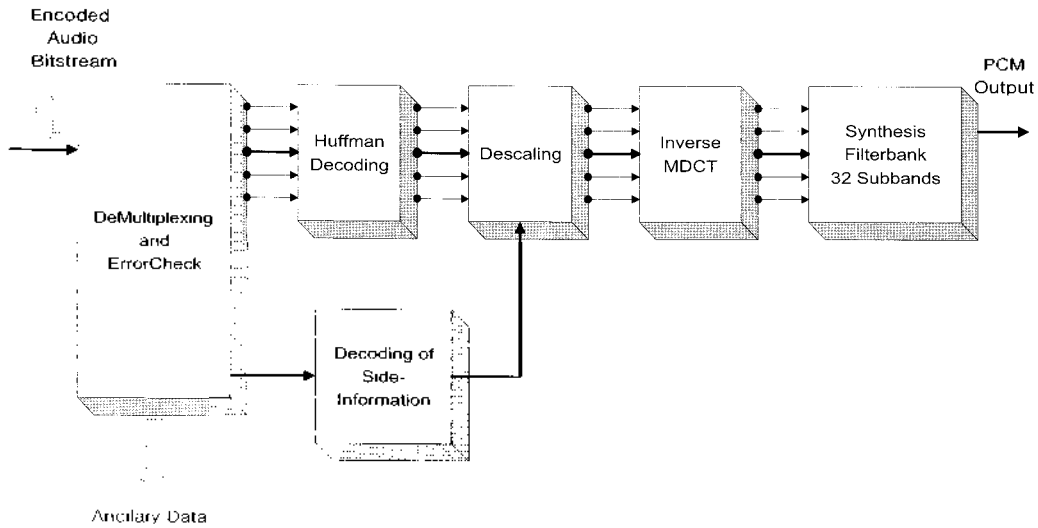


그림 1. MP3 decoding 과정  
Fig. 1. MP3 decoding process.

블록에서 MP3 압축 정보를 얻는다. 이를 바탕으로 사이드 정보 (side information)를 복호화하여 스케일 팩터 (scale factor) 값을 얻을 수 있는데, 이는 인간의 임계대역 (critical band)에 대응하여 24개 정도로 모델링된 스케일 팩터 대역 (scale factor band)에서의 스케일 팩터이다. 이 값을 이용하여 각 스케일 팩터 밴드 내에서의 노이즈 (noise) 값을 조정할 수 있다. 또한 허프만 부호화 (Huffman coding)된 각 샘플은 스케일 팩터에 의해서 디스케일 (de-scaling) 과정인 역양자화 (inverse quantization)과정을 수행한다. 그런 후에 IMDCT 과정과 서브밴드 합성 (subband synthesis) 과정을 수행하면 PCM 샘플을 얻을 수 있다.

### 3.2. SOLA 알고리즘

오디오의 템포를 변화시키는 방법으로 시간 영역과 주파수 영역에서 처리하는 대표적인 두 가지 방법이 있다. 시간 축에서 템포 변환을 위하여 널리 사용되는 대표적인 알고리즘은 SOLA (Synchronized-Overlap and Add)이며 [3-6], 주파수 영역에서 타임 스케일링 (time-scaling)의 대표적인 방법은 Phase vocoder이다 [7][8]. Phase vocoder의 경우 시간 영역의 신호를 주파수 영역에서 변하여 처리한 후에 다시 시간 영역으로 변환해야 한다. 이 때문에 시간영역에서 신호를 처리하는 SOLA에 비하여 많은 계산량이 필요하다. SOLA의 경우 시간 주파수 영역으로 변환 없이 시간영역에서 모든 과정을 처리한다 [3]. SOLA의 처리 과정은 그림 2와 같다. 오디오 샘플이 입력되면 일정한 크기의 윈도우를 일정한 간격 ( $S_a$ )로 이동시키면서 입력 샘플을 잘라낸다. 윈도우 사이즈가  $S_a$ 보다 크기 때문에 윈도우를 오버랩 (overlap) 시키면서

SOLA 프레임을 잘라 낼 수 있다. 이렇게 잘라낸 샘플은 원하는 민화율에 맞춰서 일정한 간격 ( $S_s$ )으로 재배열된다. 즉, 빠르게 재생하려면 잘라낸 간격보다 더 작은 간격으로 배열하고, 느리게 재생하려면 더 큰 간격으로 재배열시킨다. 즉,  $S_a$ 보다  $S_s$ 가 작으면 템포가 빠르게 되고,  $S_a$ 보다  $S_s$ 가 크면 템포가 느리게 바뀐다. 단순히 이렇게 원하는 비율 ( $S_a/S_s$ )만큼 재배열을 통하여 구현하면 원하는 민화율에는 정확하게 맞출 수 있으나 음질 면에서 상당한 왜곡 (distortion)이 발생하게 된다. 이러한 문제를 개선시키기 위하여 재배열 시에 정확하게  $S_s$ 의 위치가

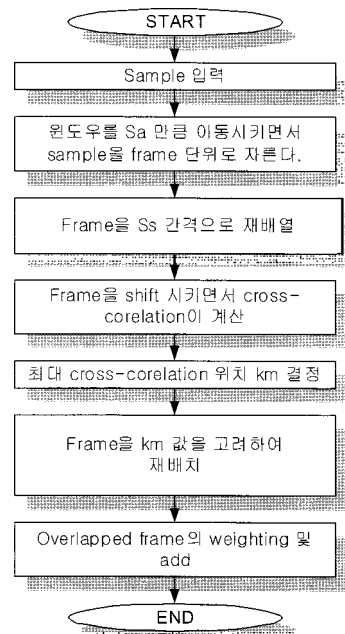


그림 2. SOLA 알고리즘 흐름도  
Fig. 2. SOLA algorithm flow.

아닌 km만큼의 위치를 이동 (shift)시킨다. 여기서 km는 연속된 프레임 간의 cross correlation이 최대가 되는 지점이다. km 값을 구하기 위하여 프레임을 이동시키면서 프레임간의 cross correlation을 구한다. 그 중에서 cross correlation 이 최대가 되는 지점이 km가 되며, 이 지점에서 프레임을 연결하면 가장 자연스럽게 연결된다고 생각할 수 있다. 이렇게 하면 고음질의 템포 변화가 가능하다. km 값을 찾는 과정에서 많은 계산 량이 필요하다. 또한 프레임을 연결하는 과정에서 자연스럽게 이전 프레임이 페이드 아웃 (fade out)되고 다음 프레임이 페이드 인 (fade in) 될 수 있도록 웨이트드 오버랩 (weighted overlap)시킨다.

### IV. 멀티채널 가라오케 시스템

#### 4.1. 채널 구성

멀티채널을 구성하면 효과적이고 실감나는 반주 구현 가능해진다. 전후좌우에서 각각 다른 음악이 나올 수 있으므로 공간감이 생길 수 있다. 따라서 영화를 비롯한 많은 멀티미디어가 멀티채널을 지원하도록 되어 있다. 가라오케도 마찬가지로 스테레오 (stereo)에서 현재는 4.1/5.1 채널까지 채널이 다양화 되어 가는 추세이다. 각 채널에는 서로 다른 악기들이 오케스트라와 비슷한 형태로 구성될 수 있으며 여러 개의 스피커를 이용하면 악기가 있는 가상의 방향을 설정할 수 있다. 본 논문의 멀티채널의 구성은 그림 3, 그림 4와 같다. 그림 3에서는 MIDI를 포함하여 모두 6개의 채널의 데이터가 믹싱 되어 5.1 채널을 구성하고 있다. MIDI 정보를 이용하여 외부에서 합성된 음악과 믹싱 되어 6개의 스피커로 나온다. 그림 4에서는 다른 형태의 구성을 보여 주고 있다. 후반 채널 대신에 드럼 채널이 믹싱 되어 전면 L/R 스피커로 나온다. 악기에서 채널에서 드럼 채널로 따로 분리하는 이유는 템포 변화 시에는 현악기, 관악기 등의 악기와 드럼이 같이 변하지만 키 변환 시에는 드럼은 키가 변하지 않아야 하므로 드럼과 드럼을 제외한 악기 채널을 분리하면 음질 개선 효과를 얻을 수 있다. 즉, 키 변환이 필요한 악기와 그렇지 않은 악기를 구분하고 있으며, 각각 처리하는 과정이 다르다. 이러한 분리하는 키 변환 시에 발생하는 드럼 악기의 왜곡 (distortion)을 방지하여 고음질의 키 변환을 가능하게 해준다.

#### 4.2. 전체 시스템 구성

전체 시스템의 구성은 그림 5에서 보여준다. 멀티채널

라이브 가라오케 시스템의 구성은 곡 정보 및 라이브 MP3 데이터와 MIDI를 저장하는 메모리인 하드디스크, 전체 시스템을 제어하는 컨트롤러, 6 채널 MP3 복호화기 및 키, 템포 변환 알고리즘을 수행하기 위한 TI사의 TMS 320C6713 DSP와 6 채널 DAC으로 구성된다 [9][10]. 한 곡은 최대 6채널까지 구성될 수 있다. 후반 L/R 대신 드럼 L/R로도 구성할 수 있다. 이러한 6 채널의 데이터는 CD 음질을 유지하기 위하여 44.1 KHz 샘플링 주파수, 16비트 데이터로 만들어지며, 저장 메모리를 줄이기 위하여 오디오 압축이 필수적이며, 이를 위하여 MP3 알고리즘을 적용하였으며, 본 논문의 라이브 가라오케는 대부분 스테레오 320 kbps 압축으로 고음질을 유지할 수 있었다.

노래반주기에서 선곡 과정이 끝나면 컨트롤러가 곡의

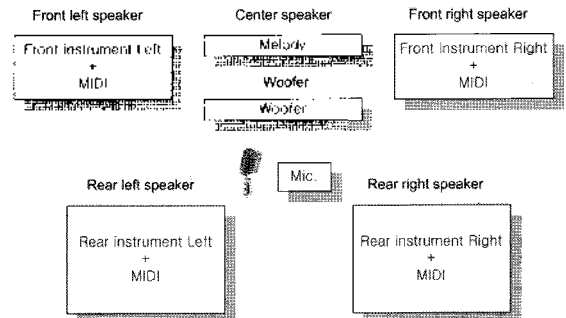


그림 3. 스피커 시스템 구조 및 후반 채널이 있을 때의 스피커 구성

Fig. 3. Speaker System and composition of the Speaker with rear channels.

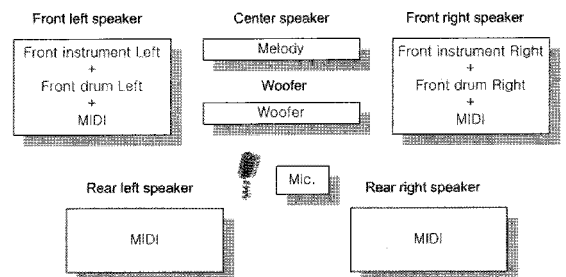


그림 4. 드럼 채널이 있을 경우의 스피커 시스템 구조

Fig. 4. composition of the Speaker with drum channels.

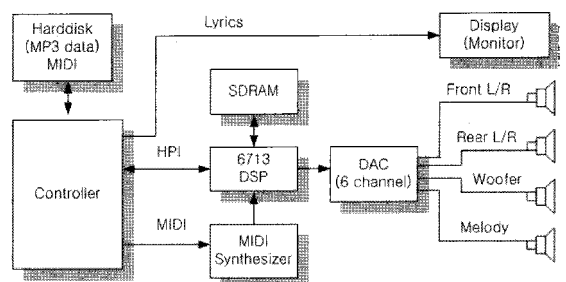


그림 5. 전체 시스템 구성

Fig. 5. System Diagram.

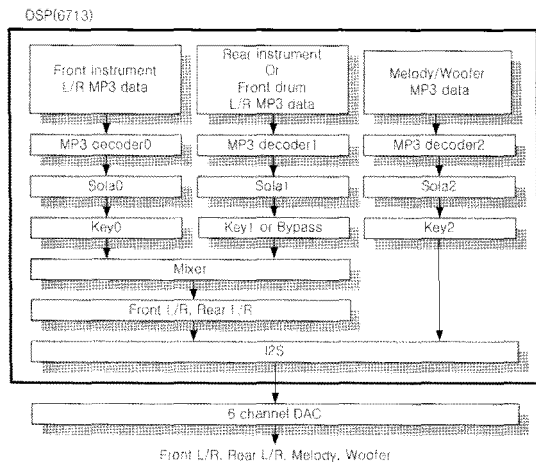


그림 6. DSP내에서 라이브 연주 처리과정  
Fig. 6. DSP Processing of Live Music.

정보를 이용하여 멀티채널 MP3 데이터가 있는 위치를 확인하여 재생 신호와 동시에 MP3 데이터를 DSP로 전송하게 된다. 그림 6에서 DSP 내의 처리 과정이 나타나 있다. DSP는 3개의 stereo MP3 data를 처리 할 수 있다. 각 채널 데이터의 처리 과정은 두 가지 경우로 나뉜다. 첫 번째는 키와 템포가 모두 변하는 경우이고, 두 번째는 템포만 변하고 키는 변하지 않아야 하는 경우이다. 일반적으로 키와 템포가 둘 다 변하는 경우가 대부분이지만 드럼 채널의 경우에는 키 높이가 없기 때문에 키가 변하지 않아야 한다. 따라서 처리 과정에서도 드럼 채널의 경우에는 키 변환 과정이 생략된다. 외부 비디 합성 칩에서 합성된 음은 DSP의 사리얼 포트 (serial port)를 통하여 내부의 6개의 채널에 믹싱 되어 DAC으로 보내지게 된다 [11][12].

### 4.3. 멀티채널 MP3 복호화기 구현

MPEG-2의 multi-channel 오디오 규격과 같이 하나의 파일을 이용하여 오디오 데이터를 처리하는 것보다 스테레오의 형식으로 여러 파일로 나눠 처리하는 것이 노래 반주의 다양성 및 요구를 충족시킬 수 있다. 이는 처음부터 끝까지 반주 중에는 조작이 없이 그대로 재생될 경우에 하나의 파일로 관리하는 것이 어렵어질 수 있지만, 대부분의 경우 노래반주기의 특성상 노래 중에 리듬을 바꾼다든지, 코러스를 제거하거나, MP3 반주 시에 미디어와 같이 사용하는 경우에는 5.1 채널의 규격을 사용하기에는 어려움이 있습니다. 이러한 경우에는 2 채널로 분리된 형태의 반주를 이용하여 리듬 변화 시에 바로 변화된 파일로 교체 재생이 가능하며, 다양한 리듬 파일을 따로 저장될 수 있다. 즉 5.1 채널 하나로 묶는 것보다는

2 채널로 나뉘서 처리하면 조금 더 다양한 조합뿐만 아니라 효율적인 관리가 될 수 있으며, 향후 필요 없는 채널의 데이터를 제거함으로써 메모리 관리에도 도움을 줄 수 있다. 또한 독립된 MP3 채널이 수행될 때 필요한 변수 및 메모리를 분석하여 다른 채널이 수행한 후에도 계속 그 값을 유지해야 할 변수와 그렇지 않은 변수로 분리하여 유지가 필요 없는 변수에 대해서는 모든 채널이 공동으로 사용하도록 하여 메모리 사용 효율을 높였다.

### 4.4. 키, 템포 변환 및 SOLA 알고리즘 최적화

반주의 템포 변환은 SOLA 알고리즘을 이용한다. SOLA 수행 과정에서 가장 많은 계산량이 필요한 부분은 프레임 간의 최대 Cross correlation을 가지는 위치 ( $k_m$ )를 구하는 부분이다. 즉, 두 프레임 간에 가장 자연스럽게 붙일 수 있는 부분을 찾는 것이다. 오버랩된 부분 대한 Cross correlation ( $R_m$ )을 찾는 과정은 식 (1)과 같다.

$$R_m = \frac{\sum_{j=0}^{L_m-1} y(mS_x + k + j)x(mS_y + j)}{\sqrt{\sum_{j=0}^{L_m-1} y^2(mS_x + k + j) \sum_{j=0}^{L_m-1} x^2(mS_y + j)}} \quad (1)$$

여기서,  $y$ 와  $x$ 는 재배열되는 연속된 두 개의 SOLA 프레임이다.

cross correlation을 찾기 위해 두 프레임의 샘플을 하나씩 다 곱해야 하는 과정에서 많은 곱셈과 덧셈이 필요하다. 또한  $k$ 값의 변화에 따른  $R_m$ 을 찾아서 그 중에서 최대  $R_m$ 값이 되는  $k$ 값이  $k_m$ 이 된다.

$$R_m = \frac{\sum_{j=0}^{L_m-N-M} y(mS_x + N \cdot k + M \cdot j)x(mS_y + M \cdot j)}{\sqrt{\sum_{j=0}^{L_m-N-M} y^2(mS_x + N \cdot k + M \cdot j) \sum_{j=0}^{L_m-N-M} x^2(mS_y + M \cdot j)}} \quad (2)$$

본 논문에서는 이러한 계산량을 줄이기 위하여 식 (2)와 같이 변수  $N$ 과  $M$ 을 추가하였다.  $N$ 은 Cross correlation을 계산하는  $k$ 값을  $N$ 의 간격으로 하는 것이며,  $M$ 은  $x$ 와  $y$ 의 모든 샘플을 곱하지 않고  $M$ 의 간격으로 곱하는 것을 의미한다. 이에 따라서, 최대 Cross correlation을 찾는 과정을 두 단계로 나뉘서 진행하므로 계산량을 줄일 수 있다. 첫 번째 단계는  $N$ 을 4,  $M$ 을 16으로 두고  $R_m$ 을 계산한다. 여기서 찾은 최대  $R_m$ 을 가지는  $k_1$  값을 찾는다. 두 번째 단계는  $k_1+N$ 과  $k_1-N$ 사이에서  $M$ 을 1로 두고  $R_m$ 을 계산하여 최대  $R_m$ 을 가지는  $k_m$ 을 찾는다. 즉, 첫 번째 단계에서 대략적으로  $R_m$ 의 최대 값을 찾고, 두 번째 단계에서 정밀

한 계산으로  $k_m$  값을 찾을 수 있으며, 이러한 과정으로 계산 량을 줄일 수 있다.

채널의 종류는 크게 드럼 채널과 그 외의 악기로 나뉜다. 드럼 이외의 채널인 경우 프레임 사이즈는 MP3 프레임의 샘플 수와 같이 맞추면 MP3 복호화 후 그 결과를 SOLA 과정을 통하여 템포 변환을 수행하게 되기 때문에 버퍼 관리에 이점이 있다. 따라서 MP3 프레임 사이즈인 1152 개로 SOLA 프레임 처리 단위를 정하면 MP3 복호화 한번에 SOLA를 한번만 수행하면 된다. 한편 드럼 채널의 경우 어택 (attack)이 많다. 이 경우 시간 영역에서 SOLA 프레임 사이즈를 작게 하면, SOLA 수행 후 어택이 두 번 발생하는 현상을 줄여서 고음질의 템포 변환을 수행할 수 있다. 프레임 사이즈가 크면 Cross correlation을 찾는 구간이 커지는데, 재배열 시에 어택이 두 번 발생할 가능성이 높기 때문이다. 또한 드럼과 같은 경우 리듬이 존재하기 때문에 SOLA 프레임 사이즈가 크면 넓은  $k$  구간에서  $k_m$ 을 찾기 때문에 리듬이 흐트러질 가능성이 있다. 따라서 드럼의 경우 음질 및 리듬 측면에서 SOLA 프레임 사이즈가 작은 경우가 더 낫다. 따라서 드럼 채널의 경우 MP3 프레임의 샘플수인 1152의 반인 576개의 입력 샘플마다 SOLA를 수행하며 SOLA 프레임 사이즈는 576이다.

키 변환의 가장 간단한 방법은 시간 영역에서 인터플레이션이나 데시메이션을 하는 것이다. 그러나 이는 시간 영역에서 템포 변화를 동반한다. 이러한 원하지 않는 템포 변환은 SOLA 수행을 통하여 보상할 수 있다. 즉, 키를 6% 올리기 위하여 시간 영역에서 6% 데시메이션을 하면 템포도 6% 동시에 빨라진다. 그러면 SOLA를 이용하여 6% 느리게 하면 정상 속도에서 키만 변환 결과를 얻을 수 있다. 따라서 키 변환 시에 발생하는 템포 변화를 미리 SOLA 수행 시에 보상해줌으로써 인터플레이션과 데시메이션을 이용한 키 변환에서 발생하는 문제점을 해결할 수 있다. 또한, 데시메이션 시에 발생하는 알리아싱 (aliasing)을 제거하기 위하여 엔터-알리아싱 (anti-aliasing) 필터가 필요하다. 본 논문에서는 MP3 복호화 과정에서 발생하는 32개 서브밴드 샘플을 이용하여 엔터-알리아싱 필터를 제거 할 수 있다. 32개의 서브밴드 샘플은 주파수 영역에서 균등하게 32등분을 한 데이터를 나타내고 있다. 따라서 데시메이션 과정에서 발생하는 알리아싱을 미리 방지하기 위하여 알리아싱이 발생하는 서브밴드의 값을 0으로 두어 MP3 복호화를 수행하면 엔터-알리아싱 필터를 사용하지 않고도 알리아싱을 방지할 수 있다. 예를 들면, 약 3% 데시메이션을 할 경우 마지막 32번째 서브밴드 값을 0으로 두면 알리아싱을 방지할 수 있다.

## V. 결과

TI사의 32 비트 부동소수점 연산 DSP인 TMS320C6713을 사용하여 고음질의 키/템포 기능을 가진 6 채널 라이브 가라오케 시스템을 구현하였다. 최종 오디오 채널은 스피커 개수와 같은 6개로 믹싱 되며, 전면 L/R, 후면 L/R 또는 드럼 L/R, 멜로디, 우퍼로 구성된다. 저장 메모리를 용량을 줄이기 위하여 MP3를 이용하여 압축된 라이브 반주곡을 복호화하며, 키/템포 변환을 위하여 SOLA 알고리즘 및 시간 영역에서 인터플레이션 및 데시메이션을 수행한다. TMS320C6713은 32비트 부동 소수점 연산을 하기 때문에 고음질을 유지할 수 있으며, MP3 복호화는 ISO에서 제시한 복호화 알고리즘과 같은 결과를 보였다 [1][2]. SOLA 알고리즘은 주로 음성에 많이 적용되어 왔으나, 여기서는 고음질을 요하는 오디오에 적용시켜서 좋은 음질을 얻을 수 있었다. SOLA 수행은 계산 량을 줄이기 위하여 최대 Cross correlation인 위치 ( $k_m$ )를 찾는 과정을 두 단계로 나눠서 빠른 수행이 가능하도록 하였다. 이러한 수행으로 Cross correlation ( $R_m$ )을 찾는 계산 량은  $N$ 과  $M$ 이라는 두 개의 변수를 이용하여 기존에 대비 12.5%로 계산 량을 줄일 수 있었다. 또한 드럼과 드럼을 제외한 악기로 분리하여 실제 연주에서 키가 변하지 않는 악기는 드럼 채널로 분리하여 반주의 키 변환 시에 드럼 채널은 키가 변하지 않도록 하여 실제 연주한 것과 같은 현상을 유지 시켰다. 또한 드럼과 같이 어택이 많고 리듬이 중요한 경우는 SOLA 프레임의 크기를 줄여서 최대 cross correlation 위치인  $k_m$ 의 변화 범위를 줄여서 어택이 두 번 발생하는 현상을 줄이고, 리듬의 흐트러짐을 방지함으로써 고음질의 템포 변환을 가능하게 하였다. 또한 키 변환 시에는 시간 영역의 인터플레이션 및 데시메이션을 이용하여 구현하였으며, 그 결과 발생하는 템포 변환은 SOLA 알고리즘에서 미리 보상해주었다. 데시메이션 시에 발생하는 알리아싱을 방지하기 위하여 MP3 복호화 과정에서 알리아싱이 발생하는 서브밴드 값을 0으로 두어 엔터-알리아싱 필터를 없앴다. DSP는 225 MHz에서 동작되며 컨트롤러와 HPI를 통하여 부팅 및 컨트롤 플래그가 설정된다. MP3의 경우 44.1 KHz, 320 kbps의 stereo 데이터가 약 14.5%, 키, 템포의 stereo 처리가 7.2%의 사용량을 기록하였으며, 정상속도에서 6 채널 동시에 동작 시 65% 사용하며, 템포가 +30%일 경우 최대 85%의 CPU를 사용하고 있다. SDRAM은 고품량 화와 낮은 cost로 인하여 사용량은 중요한 요소는 아니며, 인터페이스 버퍼 (interface buffer)로 6kbytes, 멀티

채널 MP3 수행을 위하여 32 kbytes, 멀티채널 SOLA 수행을 위하여 30 kbytes, 출력 버퍼로 36 kbytes를 사용한다. 최대 템포 변화율은 -30%에서 30%이며, 키 변화율은 -1/2 octave에서 1/2 octave이다.

그림 7, 8에서는 밑용 사운드 에디터인 sound forge와 논문에서 제안한 드럼과 일반악기의 분리 처리 방식을 이용하여 1/2 octave 변환시킨 파형을 각각 보여주고 있다. 이 두 그림을 비교해 보면 다음과 같은 차이점을 볼 수 있다. 첫째는 그림 7에서는 1, 3의 동그라미에서 표시된 부분에서 어택이 두 번 발생하는 현상이 발생되었지만, 그림 8에서는 이러한 현상이 개선되었다. 두 번째는

2번 동그라미 부분의 경우, 음이 어느 정도 일정하게 유지되는 부분에서 제안된 방법이 파형의 형태에서 비교적 일정하게 됨을 알 수 있다. 이러한 결과는 본 논문에서 SOLA 수행 시 윈도우 크기를 다르게 적용함으로써 얻을 수 있는 결과이다. 즉, 어택이 많은 드럼과 같은 악기는 윈도우 사이즈를 작게 하여 처리하고, 피아노, 현악기와 같은 악기는 윈도우 사이즈를 크게 하여 별도의 채널로 처리한 후 이 두 개를 믹싱하여 처리함으로써, 이와 같은 음질의 개선 효과를 얻을 수 있다. 특히, 저주파음이 많이 포함된 노래의 경우, 윈도우 사이즈가 충분히 크지 않으면, 음이 부드럽게 이어지지 못하고 끊어지는 현상이 발

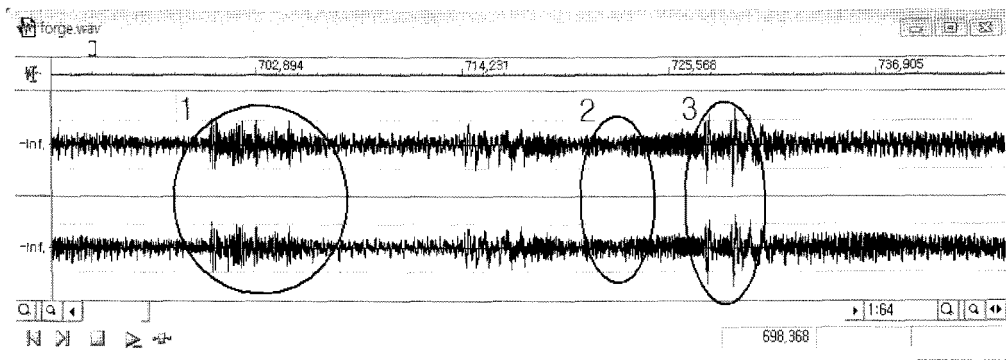


그림 7. Sound forge를 이용한 변환 파형 (key +1/2 octave)  
Fig. 7. Final waveform by Sound forge (key +1/2 octave).

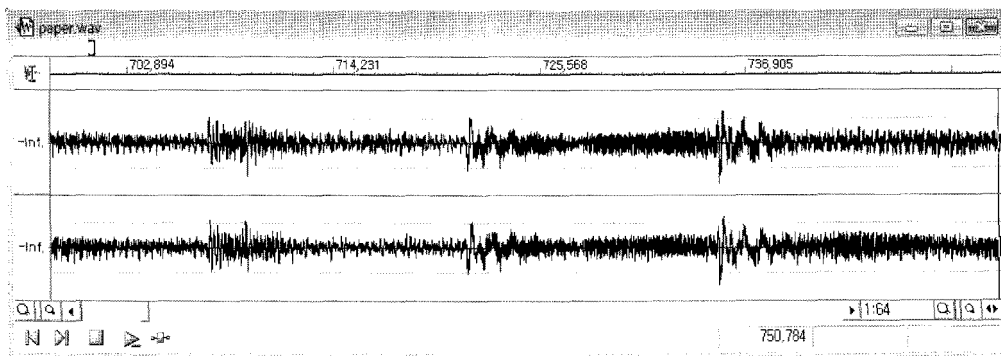


그림 8. 본 논문에서 제안한 방법을 이용한 변환 파형 (key +1/2 octave)  
Fig. 8. Final waveform by the proposed method (key +1/2 octave).

표 1. MOS 결과  
Table 1. result of MOS.

변환율	Sound forge	제안된 방법
Tempo +15%	3	4.1
Tempo +30%	2.6	3.9
Tempo -15%	4.1	4.4
Tempo -30%	3.7	4.1
Key +1/4 octave	2.4	4.0
Key +1/2 octave	1.8	3.7
Key -1/4 octave	3.5	3.6
Key -1/2 octave	2.9	3.4

생될 수 있다. 표 1은 다양한 장르의 음악에 대한 변화율에 따른 MOS (Mean Opinion Score) 결과이며 제안된 방법이 더 좋은 음질을 유지함을 볼 수 있다. MOS는 1에서 5점까지 표시가능하며, 1이 가장 낮은 음질, 5가 가장 높은 음질을 나타내고 있다.

## VI. 결론

본문에서는 상용 DSP를 사용하여 고음질 키, 템포 기능을 가진 6 채널 MP3 라이브 가라오케 시스템을 구현하였다. 이를 위해서 TI사의 32비트 부동 소수점 연산 TMS320C6713을 사용하였다. 채널수의 증가에 따른 많은 연주 데이터는 MP3 압축을 이용하여 줄였고, 드럼과 드럼 이외의 악기로 분리 처리하여 키 변환 시에도 드럼의 키는 변화지 않게 하여 실제 연주되는 상황과 같이 처리하였으며, 또한 드럼과 드럼 이외의 악기의 윈도우 크기를 다르게 하여 SOLA 알고리즘이 수행됨으로 고음질로 구현될 수 있었다. 템포 변환은 SOLA 알고리즘을 적용시켰으며 빠른 계산을 위하여 최대 Cross correlation 위치를 찾는 과정을 두 단계로 나누어서 12.5%로 계산량을 줄였다. 키 변환에는 시간 영역에서 인터플레이션, 데시메이션을 적용하였으며, 수행 후 템포 변화는 SOLA를 이용하여 보상이었다. 데시메이션 시에 발생하는 알리아싱은 MP3 복호화 과정에서 방지하여 엔티-알리아싱 필터를 필요 없게 하였다. 이러한 제안된 방법을 기존 상용 오디오 에디터에서 제공하는 키, 템포 변환 결과와 비교하였으며, MOS 평가를 통하여 고음질을 유지함을 알 수 있었다.

본 논문에서는 MP3 복호화기, SOLA, 시간 영역에서 인터플레이션/데시메이션을 통하여 고음질 및 효과적인 6 채널 라이브 가라오케를 구현할 수 있었다. 6 채널을 스테레오 MP3들로 구성함으로써 반주 시 다양한 형태의 채널 구성이 가능해지고, 또한 불필요한 채널을 사용하지 않을 수 있으므로 효과적인 구성이 가능했다. 이러한 멀티채널 오디오 시스템은 고음질 라이브 반주가 필요한 어느 곳에서나 효과적으로 적용될 수 있다.

## 감사의 글

이 결과물은 2008학년도 교육과학기술부 재정지원사업에 의해 개발되었음.

## 참고 문헌

1. ISO/IEC IS, 11172-3, Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mbit/s-part3 : Audio, 1992.
2. K. Brandenburg and G. Stall, "ISO-MPEG-1 audio: a generic standard for coding of high-quality digital audio," J. Audio Eng. Soc., vol.42, Oct, 1994, 780-792.
3. S. Roucos and A.M. Wilgus, "High quality time-scale modification for speech," Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing, 493-496, 1985.
4. E. Moulines and F. Charpentier, "Pitch synchronous waveform processing for text-to-speech synthesis using diphones," Speech Communication, 9(5/6), 453-469, 1990.
5. S. Yim and B.I.Pawale, "Computationally Efficient Algorithm for Time Scale Modification (GLS-TSM)," 1996 IEEE International Conference on Acoustics, Speech and Signal Processing Conference Processing, 1996.
6. Hamdy, K.N. and Tewfik, A.H. etc, "Time-Scale Modification of Audio Signals With Combined Harmonic and Wavelet Representations," 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing, 1997.
7. M. Dolson, "The phase vocoder: A tutorial," Computer Music Journal 10(4), 14-27, 1986.
8. J. L. Flanagan and R. M. Golden, "Phase vocoder," Bell System Technical J, 45, 1493-1503, 1966.
9. "TMS320C6713 Floating-point Digital Signal Processor - Data Manual," Texas Instruments, 2005.
10. "TMS320C67x/C67x+ DSP CPU and Instruction Set Reference Guide," Texas Instruments, 2005.
11. "TMS320C6000 DSP Multichannel Buffered Serial Port (McBSP) Reference Guide," Texas Instruments, 2004.
12. "TMS320C6000 DSP Host Port Interface (HPI) Reference Guide," Texas Instruments, 2005.

## 저자 약력

### •김 태 훈 (Tae-Hoon Kim)



1995년 : 부산대학교 전자공학과 학사  
 1997년 : 부산대학교 전자공학과 석사  
 2002년 : 부산대학교 전자공학과 박사  
 2008년 현재 : 경남정보대학 전자정보계열 전임강사  
 ※관심분야 : 사운드 압축 및 복원, DSP 설계 및 응용

### •박 양 수 (Yang-Su Park)



1967년 : 부경대학교 전자공학과 학사  
 1989년 : 동아대학교 전자공학과 석사  
 1998년 : 동아대학교 전자공학과 박사  
 1992년 8월 ~ 현재 : 경남정보대학 전자정보계열 교수  
 ※관심분야 : 로보틱스, 신경회로망, 유전알고리즘



•신 경 철 (Kyung-Chul Shin)



1987년: 원광대학교 전자공학과 학사  
1990년: 송실대학교 전자공학리 석사  
2000년: 송실대학교 전자공학과 박사  
2008년 현재: 경남정보대학 전자정보계열 부교수  
※주관심분야: 디지털신호처리, 영상신호처리

•박 종 인 (Jong-In Park)



1982년: 경북대학교 전자공학과 학사  
1999년: 부산대학교 전자공학과 석사  
2006년: 부산대학교 전자공학과 박사  
2008년 현재: (주)영명 무선연구소 상무  
※주관심분야: 통신, 컴퓨터, 신호처리, DSP 설계 및 응용

•문 태 정 (Tae-Jung Moon)



1989년: 동아대학교 전자공학과 학사  
1991년: 동아대학교 대학원 전자공학과 석사  
2002년: 동아대학교 대학원 전자공학과 박사  
1991년 ~ 1998년: 만해기술연구소 선임연구원  
2008년 현재: 경남정보대학 전자정보계열 부교수  
※주관심분야: MMIC 및 PCB 설계, 무선데이터통신