

---

# 사운드 기반 진동 시스템을 위한 진동 패턴 에디터와 컨트롤러

Vibration Pattern Editor and Controller for Sound-driven Vibration System

오성진, Sungjin Oh\*, 조동현, Donghyun Cho\*\*, 유용희, Yonghee You\*\*\*  
성미영, Mee Young Sung\*\*\*, 전경구, Kyungkoo Jun\*\*

---

**요약** ~ 본 논문에서는 사운드 기반 진동 시스템을 위한 진동 패턴 에디터와 진동 패드를 제어하기 위한 컨트롤러를 개발한다. 사운드 기반 진동 시스템은 PC의 사운드 출력단자로부터 나오는 신호를 실시간으로 분석하여 다양한 진동 효과를 발생시키는 시스템이다. 이것은 사운드 분석을 위한 DSP 시스템, 사용자의 팔 등에 부착하여 진동을 전달하는 진동패드, 그리고 이를 제어하는 컨트롤러로 구성된다. 진동 패턴 에디터에서는 패턴을 결정하는 요소를 진동소자 위치, 진동 시작시간, 지속시간, 세기의 네 가지로 정의하고, 이러한 패턴을 쉽고 빠르게 생성할 수 있는 GUI기반 사용자 인터페이스를 제공한다. 이렇게 정의된 패턴은 저장이 가능하여 재사용성이 높다. 또한 진동 패턴 데이터를 해석하여 진동 패드를 제어하는 컨트롤러 구조를 제안한다. 이 경우 진동 패턴에 따라 컨트롤러의 펌웨어를 수정해야 하는 불편함을 없앨 수 있다.

**Abstract** ~ In this paper, we develop a vibration pattern editor and a vibration pad controller for a sound-driven vibration system, which can generate diverse vibration effects in realtime by analyzing signals from the sound output of PC. It consists of a DSP system to analyze the sound, a wrist-wearable vibration pad, and its controller. For the vibration pattern editor, we define four elements to describe the pattern; the locations of vibrating elements, start time, duration, and vibration intensity. The editor provides a GUI through which users can create such patterns fast and easily, and store them for reuse. We also propose a pattern-interpreting controller. It is able to interpret patterns created by the editor and control the pad accordingly. It can avoid the need to change the controller firmware whenever desired patterns change.

**핵심어:** 진동패턴 에디터, 진동 패턴 데이터, 진동인터페이스, 사운드 기반 진동시스템,

## 1. 서론

지금까지 게임이나 가상환경에서는 주로 시각과 청각을 통해 실감성을 증대시키는 방법을 사용해왔다. 하지만 사람이 느끼는 감각은 시각과 청각 이외에 촉각, 미각, 후각 등이 있다. 시각은 눈, 청각은 귀, 후각은 코, 미각은 혀를 통해서 각각 감각을 인지한다. 이에 비해 촉각의 경우는 몸 전체에 퍼져 있다. 그렇기 때문에 다른 감각보다 더욱 큰 감각을 느낄 수 있다. 최근 연구 결과에 따르면 청각과 촉각이

동시에 발생할 경우 인간은 청각과 촉각을 각각 느끼는 것보다 효과적으로 감각을 인지한다. 이에 따른 것으로 소리와 진동을 결합시킨 여러 사용자 인터페이스가 개발이 활발히 진행되고 있다[3][4][5].

우리는 [1]에서 PC의 사운드 신호를 DSP 시스템을 통해 분석하고, 특정 사운드를 감지할 때 마다 적절한 진동을 유발시키는 사운드 기반 진동시스템을 개발하였다. 특히 효과적인 진동 효과를 위해 여러 개의 진동 소자로 이루어진 진

---

본 논문은 2007년 산업자원부 및 인천 정보 산업 진흥원의 실감형 3D 네트워크 가상환경 기술 개발의 연구비 지원에 의하여 연구되었음.

\* 주저자 : 인천대학교 멀티미디어시스템공학과

\*\* 공동저자 : 인천대학교 멀티미디어시스템공학과

\*\*\* 공동저자: 인천대학교 컴퓨터공학과

동 패드를 이용하며, 이 패드는 사용자의 팔목 등에 부착된다. 이 시스템은 그림 1과 같은 구조를 가지고 있으며, 자세한 구조는 다음 장에서 설명한다.

이 시스템의 여러 설계 요소들 중 다양한 진동 패턴을 쉽게 생성 및 수정 할 수 있는 방법과 진동 패드를 효율적으로 제어할 수 있는 컨트롤러의 설계는 특히 중요하다.

본 논문에서는 이 시스템에서 쉽고 직관적으로 진동 패턴을 생성, 수정할 수 있는 GUI기반의 진동 패턴 에디터와 생성된 패턴을 해석하여 진동패드를 제어하는 컨트롤러를 개발한다.

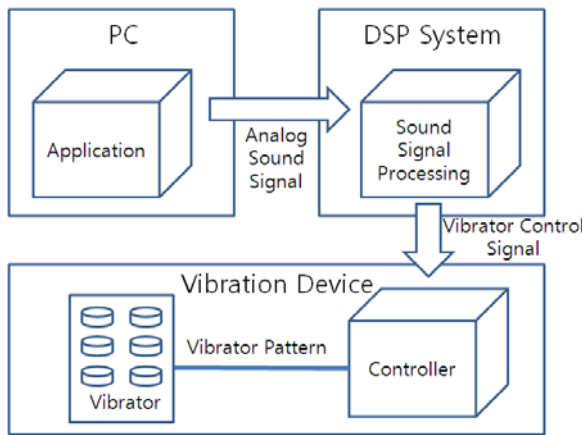


그림 1 사운드 기반 진동 시스템

## 2 사운드 기반 진동시스템

개발한 사운드 기반 진동 시스템은 그림 1과 같이 PC 어플리케이션으로부터 발생하는 사운드 신호로부터 특정 사운드를 발생을 감지하는 외부 DSP 시스템, 인체에 부착되어 진동을 유발하는 진동 디바이스가 있다. 진동 디바이스는 실제 진동을 하는 진동패드와 이를 제어하는 컨트롤러로 구성되어 있다.

실제 구현은 그림 2와 같이 구성 되어 있다. PC와 사운드 아날로그 잭으로 연결되어 있는 DSP, 컨트롤러간의 통신을 위한 UART 시리얼 인터페이스, 그리고 진동 패드를 제어하는 8비트 마이크로프로세서, 사용자에게 착용할 수 있는 아 대형 진동패드로 이루어져 있다.

### 2.1 PC 어플리케이션

PC 어플리케이션은 DSP System에 입력신호를 출력해주는 역할을 한다. PC 어플리케이션으로 가능한 것은 게임이나 음악, 메신저 등, 사운드를 통해 진동을 유발하려고 하

는 모든 프로그램이 가능하다. PC에서 DSP 시스템으로 입력 신호를 전달 할 때, 사운드가 발생한다면 어떤 플랫폼에 관계없이 신호를 전달 할 수 있는 아날로그 사운드 단자를 통해 사운드를 전달한다.

### 2.2 DSP 시스템

DSP 시스템은 PC에서 출력되는 사운드 신호를 분석하여, 진동 디바이스에게 어떤 진동 패턴으로 동작시켜 줄 것인가에 대한 명령을 내린다. DSP 시스템에서는 입력 신호를 분석하기 전에 어떤 신호가 입력 될 때, 진동 디바이스에게 이벤트가 발생해야 하는 것인지를 정의해야한다. 예를 들면 클래식 음악을 들을 경우, 저음의 경우만 진동자를 동작시키겠다는 일정한 정의가 필요하다. 또, 좀 더 복잡한 패턴의 입력을 받아 이벤트를 발생 시킬 경우, 입력되는 신호의 패턴을 미리 정의해 놓아야 한다. 입력신호가 미리 정의해 놓은 여러 가지 조건에 맞게 된다면, 진동 디바이스에게 진동 명령을 내리게 된다.

DSP 시스템이 진동 디바이스에게 진동명령을 내릴 때의 데이터는 시리얼 통신을 사용하여 전달하게 된다. 진동 명령 데이터는 진동을 어떤 패턴으로 동작시킬 것인지에 대한 패턴의 고유 ID를 문자 형태로 전달함으로써 진동 디바이스에 컨트롤러가 명령한 패턴을 인식할 수 있다.

### 2.3 진동 디바이스

진동 디바이스는 인체에 직접 접촉하여 촉각을 유발하는 진동 패드와 진동명령을 분석하여 진동 패드를 제어하는 컨트롤러로 구성된다.

진동 패드는 다양한 형태의 패드로 제작이 가능하다. 촉각은 신체의 모든 부분에 존재하므로 머리, 가슴, 배, 다리, 팔 등에 부착 가능한 진동 패드를 제작 할 수 있다. 본 논문에서는 민감도가 강한 팔에 부착하는 형태의 패드를 제작하였다.

컨트롤러는 다수의 진동패턴데이터를 저장하고 있다가 DSP 시스템으로부터 진동명령은 받으면, 저장된 진동패턴 데이터를 찾아 분석해 진동 패드를 제어하게 된다. 이 때, 진동패턴 데이터는 진동소자의 진동 시작, 지속 시간 등이 정의되어 있어 효과적으로 진동 패턴을 유발 할 수 있다. 진동 패턴에 대한 자세한 내용은 다음 장에서 설명한다.

### 3 진동 패턴 생성과 적용을 위한 시스템

본 논문에서는 2장에서 의 사운드 기반 진동 시스템에서 실감성을 극대화 할 수 있는 진동 패턴을 효율적으로 생성하고 적용할 수 있도록 그림 2와 같은 구조의 시스템을 제안한다. 진동 패턴 에디터는 사용자의 입력을 통해 진동 패턴 데이터를 만들고, 만들어진 데이터는 진동 디바이스의 컨

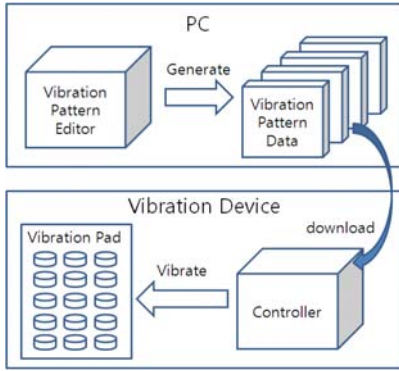


그림 2 진동 패턴 생성과 적용을 위한 시스템

트roller에 다운로드 된다. 컨트롤러는 시리얼 통신을 통해 진동 명령을 입력받게 되면 저장 했던 패턴데이터를 펌웨어가 찾아 해석하여 진동 패드를 동작시킨다.

컨트롤러의 경우, 진동 패턴과 펌웨어가 분리된 구조를 제안한다. 기존 진동 패드 제어의 경우, 컨트롤러의 펌웨어 자체가 진동 패턴을 나타내므로 패턴 변경의 경우 펌웨어 수정이 필요했다. 제안하는 컨트롤러의 펌웨어는 에디터에 의해 생성된 패턴 데이터를 해석하여 진동 패드를 제어하는 구조이므로 이러한 단점을 개선할 수 있다.

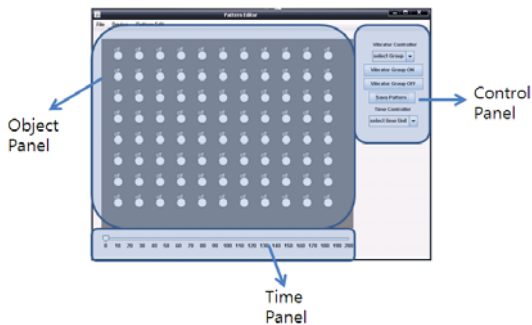


그림 3 진동 패턴 에디터 사용자 인터페이스

### 3.1 진동 패턴 에디터

진동 패턴 에디터에서는 사용자가 GUI를 통해 손쉽게 진동 패드 타입과, 진동 패턴을 정의 할 수 있고, 파일 형태로 저장 할 수 있는 툴이다. 이 진동 패턴 에디터는 그림 3과 같은 사용자 인터페이스를 제공 하며, 이것은 모두 3가지 패널로 구성된다. 오브젝트 패널에서는 진동 패드에 붙어 있는 진동 소자를 사용자에게 시각적으로 보여줄 수 있으며, 각 소자를 선택 할 수 있다. 타임 패널에서는 시간에 따라 진동 패턴을 정의 할 때, 이 타임 패널의 슬라이드를 통해 쉽게 시간적 요소를 시각적으로 표현해 입력 할 수 있다. 마지막으로 컨트롤 패널에서는 진동 패드 타입 데이터와 진동 패턴 데이터를 저장하고, 불러오며, 수정할 수 있다.

진동 패턴 에디터는 작성한 패턴 데이터를 그림 4와 같이 진동을 강도나 진동자의 상태를 시각적으로 알 수 있는 기능을 지원한다. 이 기능은 진동자의 동작을 시간 축에 따라

실시간으로 그려져, 사용자가 진동 패드를 직접 테스트 해보지 않아도 정확한 패턴을 만들 수 있다.

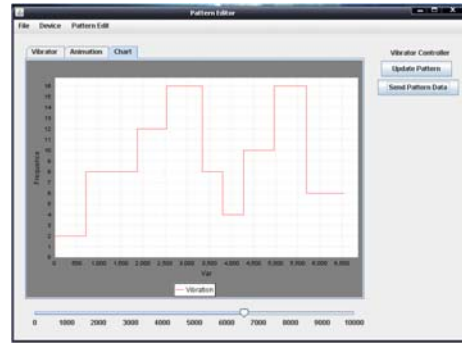


그림 4 진동 패턴 에디터 시각적 진동 차트

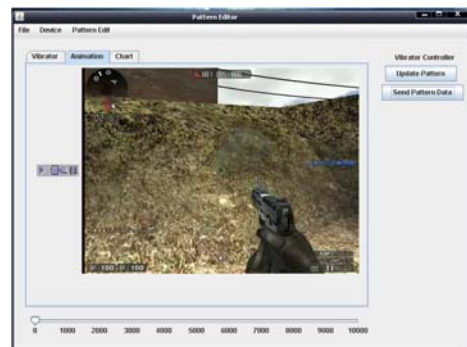


그림 5 진동 패턴 에디터 시뮬레이션

또 패턴에디터는 진동에 의한 실감성을 직접 체험할 수 있는 그림 5와 같은 시뮬레이터를 제공한다. 실감성을 느낄 때 시각, 청각, 촉각을 동시에 시뮬레이션 해 볼 수 있는 기능으로 사용자가 작성한 진동 패턴에 대한 실감성의 정도를 느낄 수 있다.

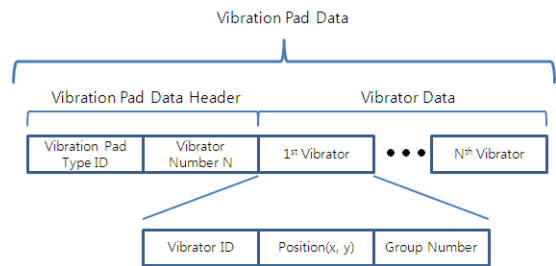


그림 6 진동 패드 데이터의 구조

#### 3.1.1 진동 패드 데이터

진동 패드 데이터는 사용자가 진동 패드에 진동 패턴을 적용시키기 전에 진동 패드의 속성을 정의해 놓은 데이터이다. 2장의 사운드 기반 진동 시스템에서의 진동 패드는 다양한 형태로 제작 될 수 있기 때문에 이를 고려해 진동 패드 데이터를 생성하여, 다양한 형태의 진동 패드에 진동 패턴을

적용 시킬 수 있게 하였다. 진동 패턴 데이터는 그림 6와 같은 구조로 이루어져 있으며, GUI의 오브젝트 패널에 진동 소자를 배치하고 진동 소자의 개수를 시각적으로 나타낼 수 있는 데이터를 포함하고 있다.

진동 패드 데이터는 헤더와 다수의 진동자 데이터로 이루어져 있다. 헤더에는 진동 패드 형태를 식별 할 수 있는 ID와 진동자의 개수가 정의된다. 진동자 데이터에서는 진동자의 개수만큼의 데이터가 존재하게 되는데 각각의 데이터는 진동자의 식별자, 진동자의 위치, 진동자의 그룹의 정보를 가지고 있다. 그룹이 다른 진동자는 오브젝트 패널에서 다른 색으로 표현되어 진다.

사용자는 진동 패턴을 생성하기 전에 진동 패드 타입을 설정해 주어야한다. 진동 패턴 에디터는 진동 패드 데이터를 분석해 GUI의 오브젝트 패널에 진동자 오브젝트를 추가하게 된다. 이 상태가 되면, 해당 진동 패드에 진동 패턴을 만들 수 있다.

### 3.1.2 진동 패턴 데이터

본 논문에서는 진동 패턴을 구성하는 인자들을 선택된 진동 소자, 진동의 지속시간, 패턴의 이름 등으로 선정한다. 이들 인자들을 어떤 형태를 조합하느냐에 따라 다양한 진동 패턴을 만들어 낼 수 있다.

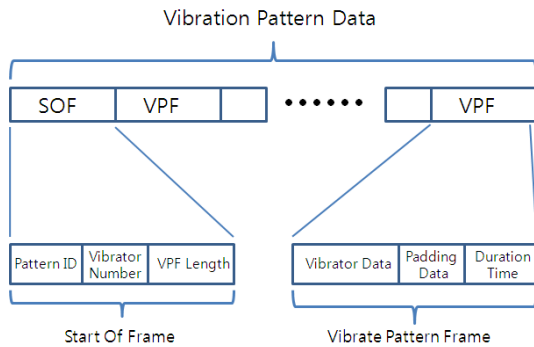


그림 7 진동 패턴 데이터 구조

진동 패턴 에디터는 시작 프레임인 SOF(Start Of Frame)을 통해 패턴의 구성이 어떻게 되어있는가를 분석해 알아낼 수 있다. 패턴의 이름인 Pattern ID를 통해 추후 특정 패턴을 실행하라는 진동 명령을 받게 되는데 이때, Pattern ID를 사용하여 어떤 패턴을 동작시킬 것인지 결정 할 수 있다. Vibrator Number는 진동 소자의 개수를 나타낸다. 이 필드는 펌웨어가 가변의 진동 소자들에게 포트와 핀을 할당해 소자들을 제어할 때, 적절한 핀을 부여하기 위함이다. VPF(Vibration Patter Frame) Length의 경우는 실제로 패턴 데이터가 담긴 VPF의 개수를 나타내서 VPD(Vibration Pattern Data)를 읽을 때 VPF의 개수만큼만 읽게 하는 기능

을 가진다.

위와 같은 구조로 생성된 진동 패턴 데이터는 컨트롤러로 쉽게 다운로드 하고, 분석이 용이할 수 있도록 고려하여 구성하였다.

### 3.2 컨트롤러

컨트롤러는 진동 명령이 DSP 에서 발생하면 진동 명령에 맞는 특정 진동 패턴 데이터를 찾아 분석하며, 진동 패드를 제어하는 역할을 한다. 그 전에 컨트롤러에는 진동 패턴 데이터에서 만들어 놓은 데이터를 저장해 놓아야 한다.

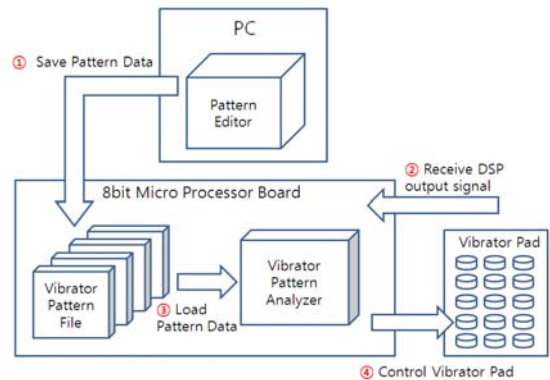


그림 8 진동패드를 제어하는 컨트롤러

그림 8에서 보면 진동 패드를 컨트롤러가 제어 할 때, 진동 패턴 데이터를 저장하는 단계(1단계), 진동 명령을 받는 단계(2단계), 명령에 맞는 진동 패턴 파일을 로딩하는 단계(3단계), 진동 패턴 데이터를 분석하며 진동자를 제어하는 단계(4단계)로 이루어져 있다. 각 단계들을 통해 패턴을 만들고 컨트롤러에서 분석하여 진동 패드를 제어하게 된다.

#### 3.2.1 컨트롤러 통신 인터페이스

컨트롤러와 DSP는 시리얼 통신 인터페이스로 이루어져 있다. 진동 명령의 데이터가 스트링으로 이루어져있기 때문에 빠른 속도의 인터페이스가 필요치 않을 뿐만 아니라 모듈이 간단하여 개발 속도를 줄일 수 있는 장점이 있다.

#### 3.2.2 진동 패턴 데이터 다운로드

진동 패턴 데이터는 컨트롤러의 플래시의 프로그램 영역을 제외한 나머지 부분에 그림 9과 같이 저장된다. 프로그램 영역을 침범하지 않는 범위 안에서 패턴들을 저장할 수 있으며 DSP에서 나오는 진동 명령에 따라 패턴 데이터의 주소를 검색하고 패턴데이터를 로드한다.

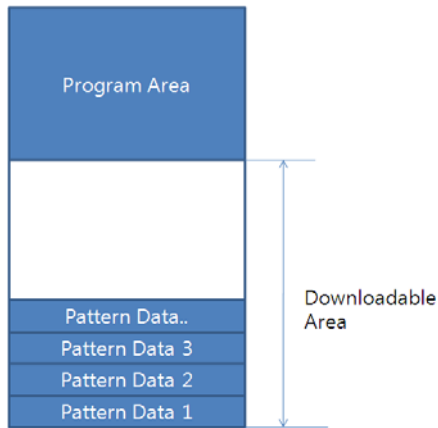


그림 9 컨트롤러의 다운로드 영역

컨트롤러의 펌웨어는 데이터를 받을 때, 진동명령 받는 모드, 진동 패턴 데이터를 저장하는 모드가 있다. 이것은 진동 패턴 데이터를 저장할 때와 진동 명령을 줄 때, 같은 인터페이스를 이용하기 때문이다. 결국 어떤 경우의 입력인지를 지정해 주어야하는데, 컨트롤러의 버튼을 통해 모드 변환이 가능하다.

### 3.2.3 패턴 데이터 분석기

컨트롤러에 다운로드 된 패턴 데이터는 진동 명령을 받으면, 진동 명령에 따른 패턴 데이터를 찾아 패턴을 분석해 진동 패드를 제어하게 된다. 가장 처음 패턴 데이터 분석기가 할 일은 어떤 패턴을 분석할 것인지를 결정하는 것인데, 이 과정을 통해 진동 명령에서 어떤 패턴으로 진동 패드를 동작시킬 것인지를 알 수 있다. 예를 들면, 패턴 A라는 패턴을 패턴 에디터로 만들면, 패턴 에디터의 식별자에 A라는 이름의 식별자를 가지게 된다. 그 후 진동 패턴 에디터가 A진동 패턴을 동작시키라고 명령을 해주면 컨트롤러는 플래시 영역의 패턴 데이터의 SOF의 패턴 식별자 필드를 검색하면서, 패턴 데이터를 찾게 된다. 찾은 데이터는 패턴 데이터 분석기를 통해서 패턴을 인식한다.

패턴 분석기의 구조는 그림 7에서 진동 패턴 데이터의 Vibrator Data 필드의 값으로 각 포트의 핀에 설정을 한 후 Duration Time의 필드의 값을 읽어 딜레이를 주는 방식으로 동작하며, VPF Length 만큼 루프가 동작되게 된다.

Vibrator Number	Vibrator Data	Padding Data
0~8	1 byte	8-Vibrator Num
9~16	2 byte	16-Vibrator Num
17~24	3 byte	24-Vibrator Num
24~32	4 byte	32-Vibrator Num
33~40	5 byte	40-Vibrator Num
41~48	6 byte	48-Vibrator Num
48~56	7 byte	56-Vibrator Num
57~64	8 byte	64-Vibrator Num

표 1 Vibrator 개수에 따른 Vibrator Data 와 Padding Data

진동소자의 개수에 따라 Vibrator Data와 Padding Data의 길이가 달라진다. 이것은 1개의 진동소자가 컨트롤러의 핀에 연결되어 있어 0 또는 1을 핀에 설정을 할 때마다 진동소자가 제어되기 때문이다. 표1을 보면 진동 소자의 개수에 따라 Vibrator Data의 길이와, Padding Data가 달라지는 것을 알 수 있다.

## 4. 결론

본 논문에서는 사운드 기반 진동 시스템을 위한 진동 패턴 에디터와 컨트롤러를 개발하였다. 에디터는 다양한 진동 패턴을 직관적으로 쉽게 생성할 수 있도록 하는 사용자 인터페이스를 제공하며, 정의된 패턴은 저장이 가능하여 재사용성이 높다. 컨트롤러 개발에 있어서는 진동 패턴 데이터를 해석하여 진동 패드를 제어하는 펌웨어 구조를 제안하였다. 이러한 펌웨어를 사용할 경우, 진동 패턴 변경에 따른 펌웨어 수정이 필요 없는 장점이 있다.

## 참고문헌

1. 조동현 외, "DSP를 이용한 사운드 기반 진동시스템", submitted to HCI 2008
2. A. Chang, C. O'Sullivan, "Audio Haptic Feedback in Mobile Phones," in Proceedings of Conference on Human Factors in Computing, 2005.
3. [www.xfxforce.com](http://www.xfxforce.com)
4. [www.sound-scape.com](http://www.sound-scape.com)
5. [www.immersion.com](http://www.immersion.com)
6. [www.atmel.com/](http://www.atmel.com/)
7. <http://www.ifree.org/jfreechart>