

## 1. 팀 정보

팀명	상쾌한	팀장	윤상준
팀원	김쾌남	팀원	
팀원		팀원	

## 2. 개발완료보고서

### 0. 작품명 - 다감각 햅틱 컨트롤러 기반의 VR 소방체험 시뮬레이션

#### 1. 개요

##### 1.1. 작품 개요

최근 가상현실(Virtual Reality, VR)이 많은 주목받으며 가상현실 기반의 게임이나 가상현실 환경에서 활용할 수 있는 컨트롤러들이 많이 개발되고 있다.

명칭	Oculus Touch	HTC Vive	PS4 Move
컨트롤러 이미지			
촉각 피드백	단순 진동	단순 진동	단순 진동

표1. 기존 컨트롤러 촉각피드백 조사자료

그러나 기존 가상현실 컨트롤러의 경우 가장 많이 활용되고 있는 HTC 사의 Vive 컨트롤러마저 햅틱 피드백(Haptic Feedback)은 단순 진동을 주는 것 밖에 없다(표1). 사용자가 현실감을 느끼기 위해 중요한 감각인 시각이나 청각 피드백이 많이 개발된 것에 비해 촉각을 통한 상호작용은 아직 개발 또는 연구 중인 분야이다. 자료조사 결과 현재 가상현실 컨트롤러에 대해 만족하는 사람은 27%, 불만족은 73%로 만족하지 못하는 사람이 매우 많다(그림1).



그림 1) 컨트롤러 체감 만족도 조사

출처 : KZERO

일반적으로 사람이 물체의 특성을 파악하고 정교하게 조작하는 데 촉각이 시각과 청각 다음으로 중요하게 작용하기 때문에 컨트롤러를 통해 몰입감 및 현실 감각을 증가시키기 위하여 햅틱 기술을 추가할 필요가 있다. 햅틱 기술이란 사용자에게 힘, 진동, 모션등을 적용하여 촉각 피드백을 구현하는 기술이다. 기존 컨트롤러들이 단순 진동을 통한 피부감각 피드백 밖에 주지 못하는 반면에 피부감각 및 힘을 사용하는 근감각을 전부 제시할 수 있는 컨트롤러는 사용자의 몰입감 및 현실감을 증가시킬 것이다.

따라서 본 연구에서 우리는 기존에 시중에서 판매되는 컨트롤러와 달리 냉온감, 부피감, 역감, 진동의 흐름과 같은 다양한 촉감을 생성하는 다감각 햅틱 컨트롤러를 개발하였고 이 컨트롤러를 어디에 사용할지 오랜 고민 끝에 감각을 가장 효과적으로 전달 할 수 있는 소방체험 시뮬레이션에 적용하였다.

## 1.2. 필요성

최근 정부는 소방 훈련에 천문학적인 비용을 지출하며, 이러한 정부의 노력 덕분에 우리는 쉽게 소방 훈련을 경험할 수 있다. 그러나 현재 우리가 쉽게 체험할 수 있는 대부분의 교육용 소방 훈련은 시나리오를 기반으로 하여 영상과 같은 시각적 정보 제공에 중점을 두기 때문에 사용자가 몰입하기 쉽지 않다.

따라서 우리는 개발한 다감각 햅틱 컨트롤러를 기반으로 소방 훈련이라는 가상현실 콘텐츠를 좀 더 현실감 있게 체험할 수 있도록, 위에서 말한 촉감들을 활용하여 실제 감각과 유사한 감각들을 제공 가능하게 구현하였다. 현재 시중에서는 이런 복합적인 감각을 전해주는 컨트롤러는 아직 개발 중이지 않으며, 단순히 진동에 의존한 감각 정도만이 흔히 접할 수 있는 종류다.

이러한 햅틱 컨트롤러와 VR콘텐츠를 소방 훈련에서 사용하면 사용자는 높은 몰입감을 느낄 수 있어 집중하게 되며, 추후 혹시 모를 화재 상황에 대처할 수 있다.

## 1.3. 개발 목표

### 1.3.1 개발 목적

컨트롤러에 냉·온감 및 입체감 등 여러 감각들을 제공할 수 있는 모듈들을 탑재하여, 소방체험 시뮬레이션에서 사용자에게 좀 더 사실적이고 생동감 있는 다양한 감각을 제공하는 것을 연구 목표로 한다.

### 1.3.2 수행과정

1. 부피감을 제시할 수 있는 파우치 제작 및 모듈 제어
2. 냉·온감 제어를 위한 회로 설계 및 PID 제어 학습
3. 진동의 흐름을 전달하기 위해 진동 모듈 렌더링 작업
4. 역감을 제공하기 위해 햅틱 제작
5. 모듈 장착에 적합한 맞춤형 컨트롤러 제작
6. 감각을 제공할 수 있는 가상현실(소방 체험 시뮬레이션) 제작
7. 가상현실에서 실제와 같은 감각을 재현하기 위한 렌더링 작업
8. 수정 보완 및 테스트

### 1.3.3 개발 결과물 형태

최종 연구 결과물의 형태는 시중에서 판매되는 컨트롤러와 비슷하나 측면에는 차가움과 따듯한 감각을 줄 수 있는 펠티어 소자가, 후면에는 부피감을 줄 수 있는 파우치와 이동진동을 발생시키는 진동액추에이터가 장착되고 컨트롤러 내부에는 역감을 주는 햅틱 휠이 내장되어 있는 모습이다. 다양한 감각을 사용자에게 전달할 수 있는 소방체험 콘텐츠는 유니티 게임엔진을 사용해 개발 하였으며 불과 물에 근접했을 때는 온도를 전달할 수 있는 냉·온감을, 밸브를 돌리거나 호스를 작동 시킬 때 햅틱 휠 모듈을 사용하여 역감을 표현 할 수 있다. 또한 물을 공급할 때 호스의 팽창을 표현하기 위해 컨트롤러 하단에 부착되어있는 파우치에 공기를 주입하여 부피감을 형성할 수 있고 물의 흐름을 표현 할 수 있도록 진동의 흐름을 나타내는 이동진동파를 생성 하였다.



그림 2) 다감각 햅틱 컨트롤러



그림 3) VR 소방체험 시뮬레이션 메인화면



그림 4) VR 소방체험 시뮬레이션 플레이 화면

## 2. 작품 설명

우리 연구는 Embedded System을 기반으로 하므로 하드웨어 및 소프트웨어에 대한 배경지식이 필요하다. 따라서 하드웨어와 소프트웨어를 다룰 수 있는 기본적인 능력을 위하여 마이크로프로세서와 소프트웨어 간 연계 작업으로 기초를 다졌다. 예를 들어 GPIO를 이용하여 LED, 스위치를 동작하게 하거나 TIMER, ADC, USART 통신 등을 해보았다. 이러한 기초 지식을 기반으로 위에서 소개했던 "촉감 모듈을 이용한 가상현실 구현"을 위해 하드웨어를 구성하였고, Unity 게임엔진을 사용하여 데이터 통신과 VR 소방체험이라는 응용프로그램을 개발했다.

### 2.1. HW 구성

#### 1) 냉·온감 모듈

냉·온감 모듈은 사용자에게 뜨거움, 차가움을 전달하는 모듈이다. 정확한 온도제어가 가능해야 특정 상황에 맞는 온열 감각을 제공할 수 있다. 또한 사용자에게 넓은 면적의 냉·온감을 제공하기 위하여 손바닥(palm) 부분에 감각이 제공되도록 배치하였다. 사람은 온도의 변화를 감지할 수 있기 때문에, 단순히 높은 온도, 낮은 온도를 전달하는 것이 아니다. 따라서 현재 온도에서 온도를 변화하는 식으로 냉·온감 모듈을 제어하였다. 우리는 이와 같은 감각을 불에 근접했을 때 온도를 올려주고, 물을 뿌려 소화하면 온도를 낮춰주는 방식으로 사용자에게 제시하였다.



그림 5) 냉·온감 모듈

#### 2) 부피감 모듈

컨트롤러 뒷면에 부착되어있는 파우치에 공기를 넣어줌으로써, 사용자(User)가 응용프로그램의 다양한 상황에 따라 손바닥에 입체감을 느끼게 해준다. 파우치는 탄성이 매우 좋은 Dragon Skin 재질을 사용하여 많은 공기가 유입돼도 터지거나 변형되지 않도록 제작되었다. 가상현실 내에서 물을 틀 때 호스가 손 안에서 팽창되는 느낌을 부피감으로 구현하였다.

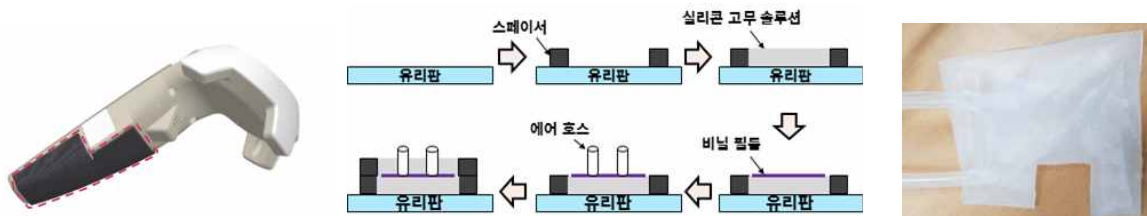


그림 6) 파우치 제작 과정 및 파우치

3) 햅틱 휠 모듈

자기장을 인가하면 굳기가 달라지는 자기유변유체를 회전타입 액추에이터에 개발된 자기유변유체 기반 회전타입 액추에이터는 자기장 및 자기유변유체의 차폐를 위한 하우징 및 커버와 자기장 생성을 위한 솔레노이드 코일, 회전을 하며 생성된 역감을 조이스틱에 전달하는 샤프트로 구성된다. 이를 소화전 밸브를 돌릴 때 뻑뻑함에서 점차 부드러움으로 넘어가는 감각을 표현하였다.

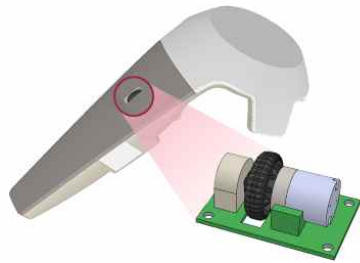


그림 7) 햅틱 휠 모듈



그림 8) 자기유변유체

4) 진동 모듈

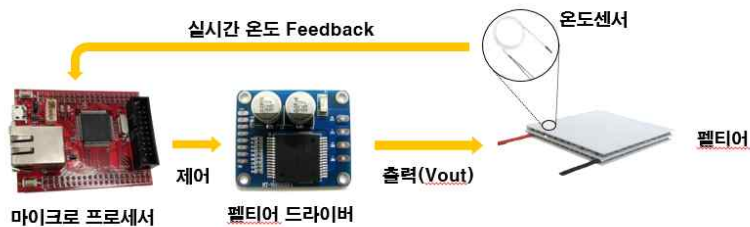
사용자가 가상현실 내 물체의 이동이나 알람 효과를 느낄 수 있는 진동 모듈을 개발하였다. 진동의 흐름을 주기 전달하기 위한 액추에이터로는 지름 8mm의 Linear Resonant Actuator(LRA)를 사용했으며, 각 액추에이터 간 간격은 목표에 맞게 150mm로 설정하였다. 각 LRA간 개별 진동 및 동적 진동 표현이 가능하도록 제어 알고리즘을 구현하였다. 물이 흐를 때 그 흐름에 맞춰 이동 진동이 발생하며, 폭발의 현장감을 높이기 위한 단순진동 또한 제공한다.



그림 9) 진동 모듈의 위치 및 흐름도

2.2. HW 기능(제어 방법 등 서술)

1) 냉·온감 모듈

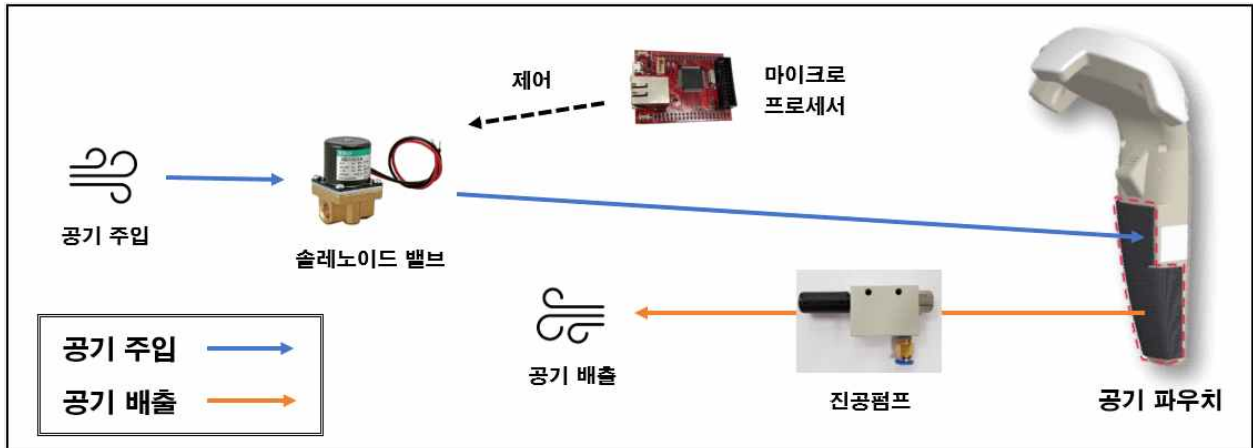


펠티어(냉·온감 모듈)에 부착된 온도센서에서 피드백 받은 온도를 바탕으로 마이크로프로세서가 드라이버에 주는 입력(PWM, GPIO)이 정해지고 출력 값(Vout)이 펠티어(냉·온감 모듈)에 전달된다. 그리고 출력 값을 정하는 알고리즘은 PID 제어 알고리즘을 사용하여 목표 온도와 현재 온도의 차이를 고려하여 PWM(전압의 세기), GPIO(전류의 방향)를 결정해서 동작한다.



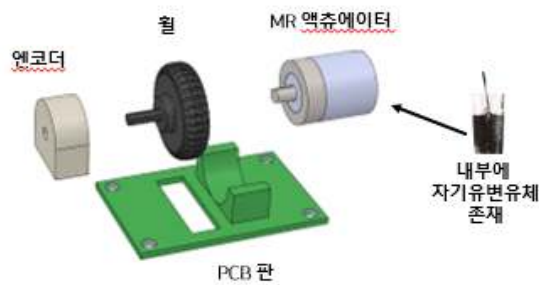
펠티어(냉·온감 모듈)의 특성상 한쪽 면이 차가워지면 반대쪽 면은 뜨거워지는 특성을 가지고 있다. 그렇기 때문에 반드시 냉각시스템이 존재해야 하며, 컨트롤러라는 작은 부피의 특성상 공냉 시스템은 사용할 수 없다. 위의 냉각시스템도 냉·온감 모듈의 서브시스템 중 하나로써 독립적인 구성으로 차가운 물을 순환시켜 주는 역할을 한다.

## 2) 부피감 모듈



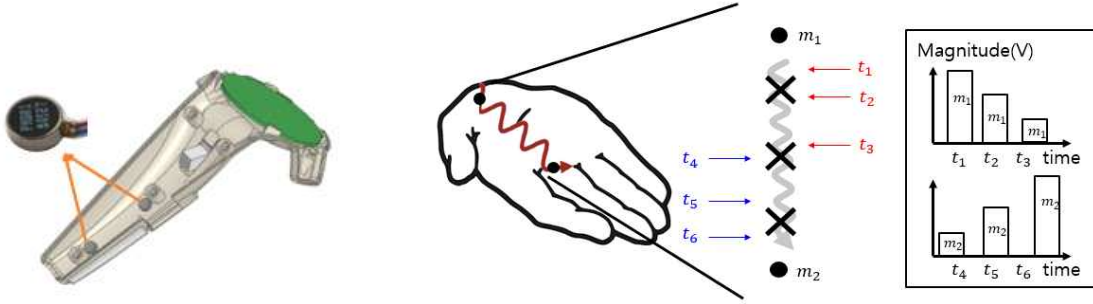
소형 에어 컴프레셔로 공기를 공급하고, 공기의 흐름을 솔레노이드 밸브와 진공펌프를 통해서 제어하도록 한다. 사용자에게 직접적으로 감각을 제공하는 공기 파우치의 공기의 유무를 총 3개의 솔레노이드를 마이크로프로세서에서 제어를 하여서 공기 파우치의 팽창과 수축을 통해 부피감을 형성하게 된다.

## 3) 햅틱 휠 모듈



역감을 제공하는 햅틱 휠 모듈의 경우는 사용자와의 상호작용을 하는 부분은 휠이지만 MR 액추에이터 내부에 존재하는 MRF(자기유변유체)가 들어있으며, MRF에 인가되는 전압에 따라서 액체 상태와 고체 상태로 자유롭게 변형이 가능하다. 이 물체가 휠이 돌아가는 부분을 간섭하여 휠을 돌리는 감각을 부드럽거나 뻑뻑하게 즉, 역감을 사용자에게 제공할 수 있게 한다.

#### 4) 진동 모듈



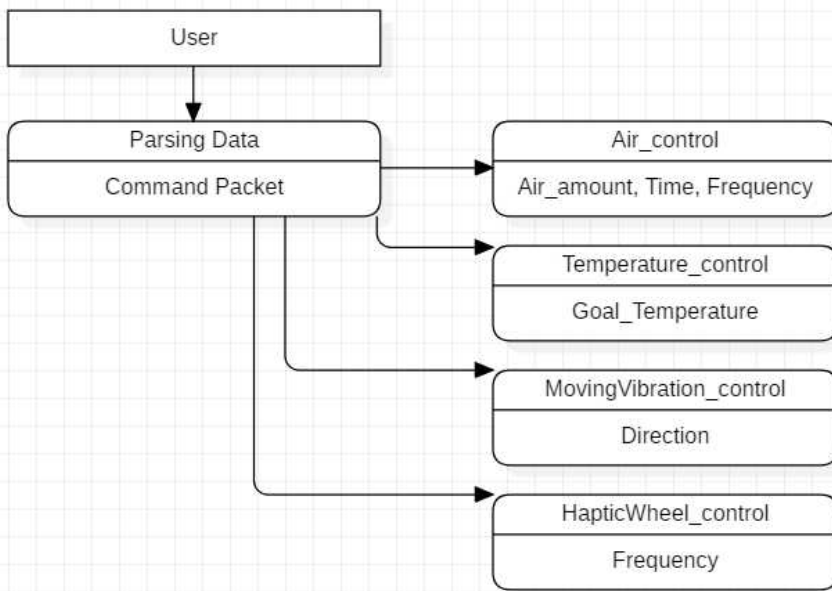
진동모듈은 컨트롤러 내부에 총 2개가 부착되어 있으며, 2개를 사용하여 단순 진동뿐만 아니라 전압의 세기를 2개를 각각 컨트롤 해줌으로써 위에서 아래로 또는 아래에서 위로 진동의 흐름을 표현할 수 있습니다. 가장 우측의 그래프와 같이 2개의 진동모듈에 시간에 따라서 주는 전압의 세기를 제어한다.

### 2.3. Software 구성

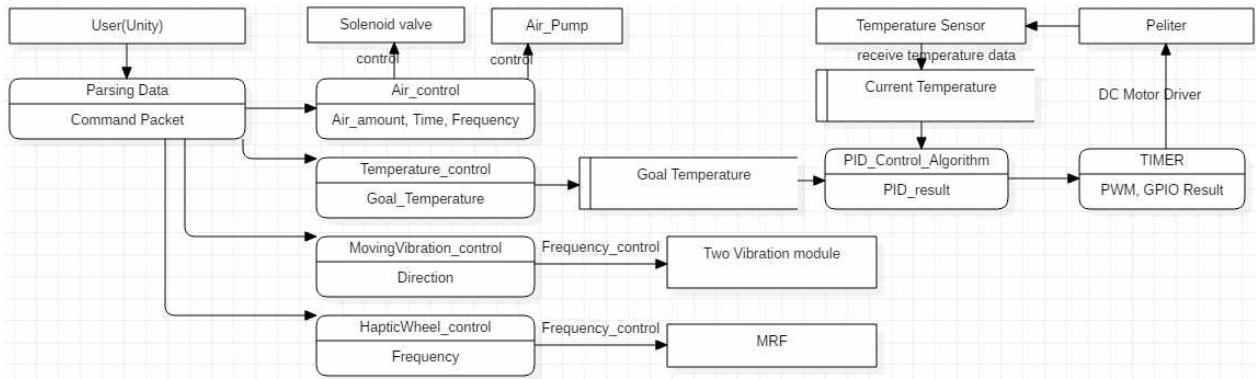
Software는 크게 2가지로 볼 수 있다. 마이크로프로세서의 시스템을 제어하는 C 코드, VR 소방체험 시뮬레이션에 관련된 C# 코드로 나뉜다. 마이크로프로세서의 C 코드는 IAR 시스템 기반으로 위에서 소개한 HW의 모듈들을 동작시키고 제어하는 알고리즘, 데이터 통신, 명령어 packet 정의를 수행한다. Unity에서의 VR 소방훈련 시뮬레이션의 인터페이스를 구현하고 객체들의 생성, 이동, 회전, 크기 변화, 소멸 등의 역할을 수행하며, SerialPort를 통해 명령어 packet을 마이크로프로세서에게 전달한다. Unity와 IAR System 간의 통신으로 각종 모듈과 VR 어플리케이션의 동기화가 이루어지도록 Software가 구성되어 있다.

### 2.4. Software 설계도(흐름도 및 클래스 다이어그램 등 (개발언어에 따라 선택))

#### 2.4.1. IAR System 흐름도

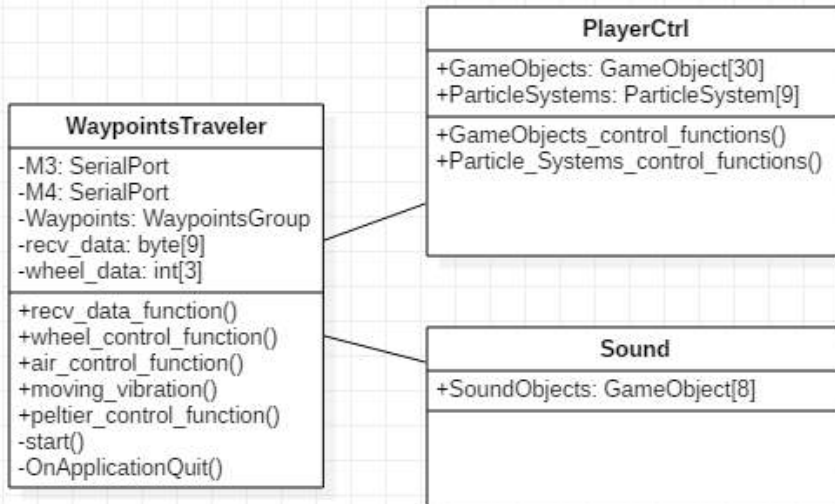


유저가 정의된 명령어 packet을 송신하면 마이크로프로세서가 수신하여 Data를 분석한다. 유효한 명령어인지, 부피감, 냉·온감, 이동진동, 역감의 명령어인지 판단을 하고 그 명령어 packet안에 담겨있는 파라미터 값들을 통해 각종 모듈 프로세스에 따라 HW에 신호를 주거나 제어 알고리즘을 통해서 동작한다.



위에서 언급한 유저는 Unity를 의미하고 Unity에서 명령어 packet을 송신하면 그에 따라서 각각의 프로세서로 전달이 되는데 부피감을 제어하는 Air\_control 프로세스는 솔레노이드 밸브와 Air pump를 제어함으로써 공기의 유입과 배출을 담당하여 컨트롤러 하단에 부착된 파우치의 공기의 양을 조절하여 사용자에게 부피감을 제공한다. Temperature\_control 프로세스는 목표하는 온도가 존재해야하며 냉·온감 모듈에 부착된 온도 센서로 부터 실시간 온도를 받아와서 PID 제어 알고리즘을 통해 산출된 결과 값으로 다시 온도 모듈에 전압의 세기와 방향을 출력한다. MovingVibration\_control 프로세스는 컨트롤러 내부에 위치한 2개의 진동모듈의 진동 세기를 조절하여 위에서 아래로, 아래에서 위로 흐르는 것과 같은 감각을 제공한다. 마지막으로 HapticWheel\_control 프로세스는 MRF라는 자기유변유체에 자기장을 인가하여 휠에 회전에 간섭하며 역감을 생성하는데 자기장의 세기에 따라서 점성이 달라지는 것을 이용하여 제어하게 된다.

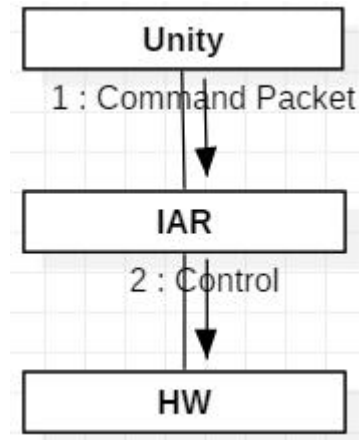
## 2.4.2. Unity 시스템 Class Diagram



Unity는 C#을 기반으로 한 클래스의 형태로 구성되어 있는 소프트웨어이다. 위에 존재하는 클래스 이외에도 Oculus HMD의 카메라를 동기화시키기 위한 클래스와 같은 여러 클래스가 존재하지만 저희가 직접 개입한 클래스로는 크게 위의 3개와 같다. 우선 Sound 클래스는 VR 소방 시뮬레이션에서 사용자에게 제공하는 Sound들을 선언하여 WaypointsTraveler에서 호출하여 사용합니다. 또한 PlayerCtrl 클래스는 사용자에게 public으로 선언된 GameObject들과 ParticleSystem들을 선언하여서 객체의 생성, 이동, 회전, 소멸을 시각적으로 사용자에게 제공하는 역할을 한다.

마지막으로 WaypointsTraveler는 마이크로프로세서의 SerialPort를 선언하여 통신을 열어주고, 지정된 시나리오의 경로대로 움직이도록 Waypoints들을 선언해주고 생성하고자 하는 감각들의 파라미터 값을 명령어 packet으로 IAR system에 송신하는 역할을 하는 클래스이다.

## 2.5. Software 기능 (알고리즘 설명 포함)



최상단 시스템인 Unity에서 응용 프로그램을 설계하고 사용자에게 제공하는 인터페이스들을 구성한다. 사용자에게 감각을 제공하기 위해 Command Packet을 Serial 통신으로 IAR System에 전달을 하면 IAR System에서는 HW의 컨트롤하기 위한 코드를 수행하여 직접적인 HW에 간섭하는 순서로 각각의 단계로 나뉘어져 있다.

## 2.6. 프로그램 사용법 (Interface)



저희 시뮬레이션을 체험하기 위해 VR 기기를 착용한 후 시작하게 되면 간단한 안내와 함께 화재 발생 지역으로 이동하게 된다. 시뮬레이션에 대해서 간략한 정보를 미리 습득을 한 후 시나리오 진행에 도움이 되도록 한다.



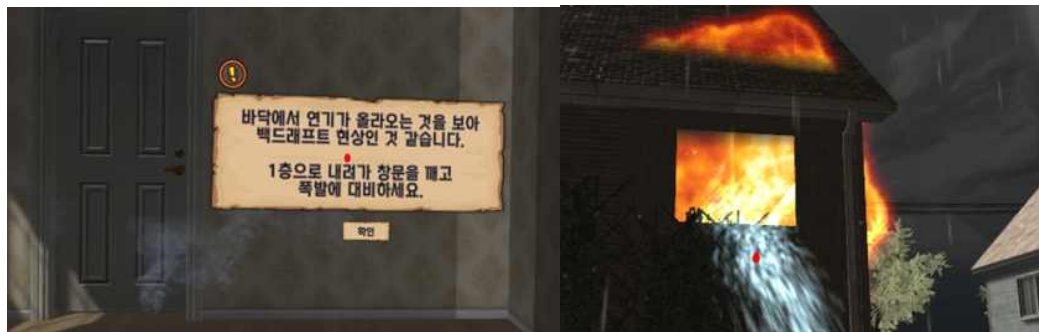
이와 같이 화재 현장에 도착하게 되면 화재를 진화하기 위한 행동(상호작용)을 취해야 한다.



위와 같이 헤드트래킹을 이용하여 소화전에 호스를 연결하고 나서, 컨트롤러 상단에 부착된 Haptic Wheel을 위로 돌리면서 물을 공급하게 된다. Haptic wheel을 돌려서 물을 틀게 되면 물의 흐름을 부피감으로 파우치의 팽창과 수축에 의해 사용자에게 전달하게 된다.



위의 사진을 호스를 틀었을 때 물이 분사하는 물의 흐름을 이동진동으로 감각을 전달하며, 동시에 냉·온감 모듈을 이용하여 차가운 감각을 생성한다. 그리고 물을 틀고 나서, 불에 헤드트래킹을 사용하여 가져다대면 불을 진압할 수 있다. 우측의 사진에서 물을 틀지 않고 불에 가까이 접근 했을 때는 냉·온감 중 온감으로 사용자에게 따뜻함을 전달한다.



불을 진압하는 과정에서 2층 방에서 연기가 피어오르는 것을 발견 한 후, 백 드래프트의 현상임을 인지하고 안전지역으로 대피한다. 대피 한 후, 백 드래프트 의심 지역의 창문을 물을 분사시켜 깨게 되면, 폭발 진동과 함께 안전하게 대처했음을 알리고 시나리오가 끝이 나게 된다.

## 2.7. 개발환경 (언어, Tool, 사용시스템 등)

컨트롤러에 내장되어 있는 각각의 감각 모듈들을 제어하기 위해 IAR 개발환경을 사용하였다.



그림 10) IAR 개발환경

유니티 게임엔진 툴을 사용해 VR 소방체험 시뮬레이션을 구현하였으며 개발언어는 C# 이다.



그림 11) 유니티 게임엔진



그림 12) C# 개발언어

## 3. 프로그램 설명 (최대한 자세하게 기술)

### 3.1. 파일 구성

#### ARM)

jj\_adc.h – Peliter 소자에서 온도센서로 온도 값을 ADC로 반환받기 위해서 사용하는 헤더. 즉, 현재 온도를 반환을 정의한 헤더파일.

jj\_pid.h – 현재 온도와 목표 온도에 대해서 PID 제어를 기반으로 연산된 PID 반환 값을 계산하는 수식 정의한 헤더파일.

jj\_gpio.h – 온도를 조절하기 위해 Peliter 소자는 DC Motor driver를 사용하는데 방향을 결정하는 함수를 정의한 헤더파일.

jj\_peltier\_control.h – Peliter 제어를 위한 파라미터 구조체를 정의한 후, Peltier 객체를 초기화하고 PID 파라미터 값을 세팅, 제어하는 함수를 정의한 헤더파일.

main.c – Motor driver 의 PWM값 제어, Peltier 객체 생성하는 기능을 동작, 마이크로프로세서에서 TIMER호출(명령 패킷 제어), 부피감 제어를 위한 솔레노이드 밸브를 제어하는 함수 정의.

#### Unity)

PlayerCtrl.cs – Unity의 전반적인 사운드, 객체들의 크기 조정, 회전, 이동을 제어하는 C# 파일.

WaypointsTraveler.cs – Player가 지정된 위치에서 인터럽트들을 처리하고, 전체적인 시나리오를 진행하는 C# 파일.

### 3.2. 함수별 기능

#### ARM)

**init\_RecvData()** - 명령 패킷은 start\_bit, com\_bit, min\_bit, max\_bit, time\_bit, frq\_bit, extra\_bit, stop\_bit으로 총 8개의 값들을 초기화 하는 함수. 하드웨어 제어의 편의성을 위해 모든 명령은 위와 같은 형태로 구성되어 있으며, start\_bit와 stop\_bit는 명령어의 시작과 끝을 알리기 위함이고 나머지 6개의 데이터를 통신하여 HW들을 제어하기

위함. Unity와의 통신을 담당하는 함수.

**USART1\_IRQHandler()** - USART 통신을 위한 함수로 명령어를 수신하면 start\_bit와 end\_bit를 기준으로 패킷의 파라미터들을 저장

**TIM2\_IRQHandler()** - Peliter를 제어하기 위해서 PID제어 알고리즘의 결과 값에 따라서 PWM, GPIO 신호를 DC Motor Driver에 전달하는 역할.

**parsingData2()** - 수신된 명령 패킷을 해석해서 HW를 동작시키기 위한 함수들을 호출, 각종 모듈들을 제어하기 위한 데이터와 파라미터 값에 따라서 감각을 형성.

**initPelCtrlVal(\*PelCtrlVal, pVal, iVal, dVal)** - Peltier 객체 생성한 후 PID 제어에 필요한 상수들을 지정, 관련된 함수들을 호출하여 사용자에게 목표온도를 정확하고 빠르고 안정적으로 전달하기 위한 목적을 가지고 있다.

**Air\_Pressure\_Test()** - 기존에는 공압을 Test할 목적으로 작성한 함수로, 공기 주입, 공기 유지, 공기 배출의 총 3동작을 수행하며 수신된 명령 패킷에 따라서 각각의 시간을 제어하여 여러 가지 감각을 생성.

### Unity)

**Start()** - 마이크로프로세서의 포트를 오픈하며 초기화 선언이 필요한 객체들을 초기화 하는 역할

**recv\_data\_function()** - Haptic wheel 의 반환 값, 즉 위로 회전하는지 아래로 회전하는지에 대한 정보를 수신해서 wheel\_control\_function에서 제어할 변수를 제어.

**wheel\_control\_function()** - recv\_data\_function에서 Haptic wheel의 움직임을 수신해서 역감을 생성하는 함수, 6단계로 분리해서 인가하는 주파수를 다르게 하여 역감을 사용자에게 제공하는 역할.

**air\_control\_function()** - 공압을 제어하는 function으로 마이크로프로세서에 명령어를 전송하여 솔레노이드 밸브를 실제 작동해서 부피감을 사용자에게 제공하는 역할.

**peltier\_cool\_180)** - peltier를 냉각시키는 명령어를 마이크로프로세서에 명령어를 전송하여 사용자에게 냉감을 제공하는 역할.

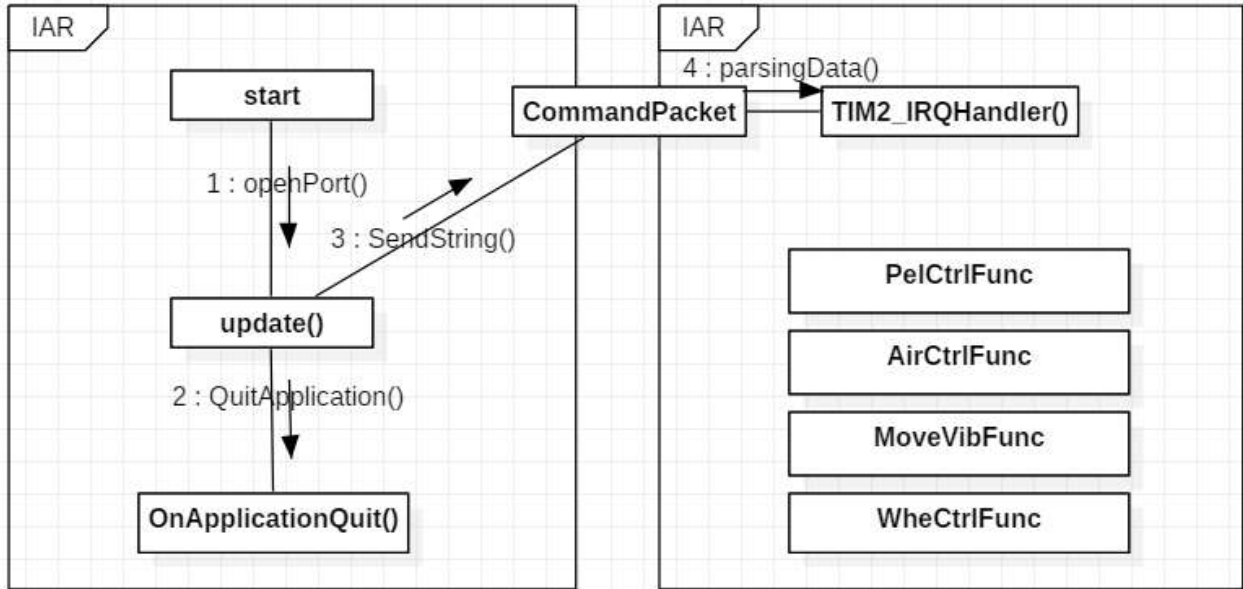
**peltier\_hot\_280)** - peltier를 가열시키는 명령어를 마이크로프로세서에 명령어를 전송하여 사용자에게 온감을 제공하는 역할.

**moving\_vibration()** - 이동진동의 명령어를 마이크로프로세서에 명령어를 전송하여 사용자에게 이동진동을 제공하는 역할.

**Update()** - 본 소방시뮬레이션의 시나리오에 따라서 현재 위치에서 사용자와의 모든 상호작용을 정의하는 역할. 계속해서 호출되며 반복 작업을 통해 사용자의 인터럽트에 따른 대부분의 이벤트 처리.

**OnApplicationQuit()** - 시스템의 안정성을 위해서 Unity 프로그램이 종료될 때 호출되는 함수로, 하드웨어들의 초기화, 통신 포트를 닫는 역할.

### 3.3. 주요 함수의 흐름도



### 3.4. 기술적 차별성

현재 시중에 존재하는 대부분의 컨트롤러는 진동에 한정된 감각만을 사용자에게 제공한다. 냉·온감, 부피감, 역감, 이동진동 등의 촉감으로 지금까지는 느껴보지 못했던 감각들을 VR 콘텐츠와 함께 사용자에게 제공할 수 있다.

### 4. 개발 중 장애요인과 해결방안

부피감 모듈에서는 팽창과 수축을 반복해도 터지거나 늘어나는 등의 변형 문제가 발생하지 않으며, 원하는 형태와 크기를 가진 파우치가 필요했다. 또 컨트롤러에 부착하는 것 역시 가능해야 했다. 시중에 나와 있는 여러 재질들로 파우치 제작을 시도했으나 원하는 것만큼의 형태와 강도를 제공할 수 있는 것이 없었다. 여러 번의 시도 끝에 Dragon Skin이라는 액상 실리콘을 파우치와 같은 형태로 제작하여 내구성이나 탄성, 형태를 모두 만족할 수 있는 파우치를 제작하였다.

냉·온감 모듈에서는 온도를 낮추기 위한 냉각 시스템의 성능뿐만 아니라 온도 제어와 온도 유지의 어려움을 해결해야 했다. 따라서 냉·온감 모듈의 가열되는 면에 성능이 떨어지는 공랭보다 냉각효과가 뛰어난 수랭을 채택하게 되었으며, 온도 유지에 관해서는 PID 제어라는 알고리즘을 사용하여 섬세한 제어와 온도 유지를 위한 충분한 전류 공급을 할 수 있도록 회로를 구성하였다.

컨트롤러 내부에 탑재한 두 개의 진동 모듈은 실제로 진동이 이동하는 듯한 흐름을 제공하는 것이 목표였다. 부피감과 냉·온감이 제공되는 동시에 진동의 흐름을 제공할 시 상대적으로 감각이 묻히는 문제가 발생하여, 유사하면서도 확실한 감각을 표현하기 위해 렌더링 및 테스트 작업을 반복하였다.

햅틱 휠 모듈도 감각의 표현에 있어서 원하는 정도와 강도를 맞추기 위해 다양하고 반복적인 테스트를 시행했다.

그리고 냉·온감뿐만 아니라 부피감 모듈에서도 서브시스템의 부피가 너무 크다는 단점이 있었지만, 컨트롤러와 분리시켜 컨트롤 박스에 내장함으로써 해결하였다. 서브시스템에 존재하는 여러 펌프들의 소음과 실행 시 불필요한 진동이 발생하는 문제도 있었는데, 방음재를 컨트롤 박스 내부에 부착하여 불필요한 감각이 사용자에게 전달되는 것을 최대한 단절 시킬 수 있도록 하였다.

### 5. 개발결과물의 차별성

현재 시중에 존재하는 대부분의 컨트롤러는 진동에 한정된 감각만을 사용자에게 제공한다. 냉·온감, 부피감, 역감, 이동진동 등의 촉감으로 지금까지는 느껴보지 못했던 감각들을 VR 콘텐츠와 함께 사용자에게 제공할 수 있다.

6. 단계별 개발계획 및 실제 참여인원 및 업무 분장

		1월	2월	3월	4월	5월	6월	7월	8월	9월	10월
H/W 모듈설계											
H/W 시스템 통합											
H/W 테스트 및 문제해결											
H/W와 VR 애플리케이션 간 통신											
H/W_김괘남 팀원 담당											
VR 콘텐츠	기능 응용 아이디어 구상										
	시나리오 구상										
	캐릭터와 맵(UI) 모델링										
	애플리케이션 개발										
	가상현실 기반 HMD 적용										
VR 콘텐츠_윤상준 팀원 담당											
전체 시스템 테스트 및 문제점 보완											

물론 윤상준, 김괘남 팀원 모두 하드웨어와 VR콘텐츠를 열심히 작업했습니다. 다만, 업무 분담을 한다면 위의 표와 같이 하드웨어는 김괘남 팀원이 조금 더, VR 콘텐츠는 윤상준 팀원이 조금 더 열심히 작업한 것 같습니다.