

---

# IFTTT 기반의 스마트홈 서비스 리모컨 설계

권동현\* · 임지용\* · 허성욱\* · 오암석\*

\*동명대학교

## IFTTT-based smart home service remote control design

Dong-hyeon Kwon\* · Ji-yong Lim\* · Sung-uk Heo\* · Am-suk Oh\*

\*Tongmyong University

E-mail : donghyun130@naver.com

### 요 약

사물인터넷 기술이 발달하면서 다양한 사물인터넷 디바이스가 등장하고 있다. 이러한 사물인터넷 디바이스를 제조하는 제조사에서는 디바이스를 제어하는 서비스를 IFTTT에 등록하여 사용자가 서비스를 이용할 수 있도록 하고 있다. 하지만 대부분의 사물인터넷 디바이스를 스마트폰 애플리케이션으로 제어하고 있어 제조사가 다른 디바이스를 사용할 경우 여러 애플리케이션을 설치하여 사용해야 한다. 이에 본 논문에서는 IFTTT 자동화 서비스 플랫폼과 연동하여 다양한 스마트홈 디바이스들을 제어할 수 있는 스마트홈 서비스 리모컨을 설계한다.

### ABSTRACT

As the Internet technology of things is developed, a variety of things Internet devices are emerging. Manufacturers of such Internet devices are registering with the IFTTT to control the device so that the user can use the service. However, most object Internet devices are controlled by smartphone applications, so when a manufacturer uses different devices, many applications should be installed and used. In this paper, we design a smart home service remote control that can control various smart home devices in cooperation with IFTTT automated service platform.

### 키워드

IoT, IFTTT, SmartHome, ESP8266

## I. 서 론

사물인터넷 기술이 발달하면서 다양한 사물인터넷 디바이스가 등장하고 있다. 다양한 사물인터넷 디바이스의 등장은 우리가 생활하는 환경에도 많은 변화를 가져다주고 있다. 네트워크를 통해 일상생활에서의 행동이 설정되면 쾌적한 냉난방, 음악 감상, 조명과 차양을 조절하는 등 다양한 행동을 사용자 중심의 맞춤형으로 제공하여 준다. 이러한 사용자 중심의 기능을 위해 사물인터넷 디바이스를 제조하는 제조사에서는 디바이스를 제어하는 서비스를 IFTTT에 등록하여 사용자가 서비스를 이용할 수 있도록 하고 있다. 하지만 대부분의 사물인터넷 디바이스를 스마트폰 애플리

케이션으로 제어하고 있어 제조사가 다른 디바이스를 사용할 경우 여러 애플리케이션을 설치하여 디바이스를 제어할 때마다 애플리케이션을 변경해야 하며 사용해야 한다. 이에 본 논문에서는 IFTTT 자동화 서비스 플랫폼과 연동하여 제조사가 다른 다양한 스마트홈 디바이스를 제어할 수 있는 통합형 스마트홈 서비스 리모컨을 설계한다.

## II. 관련연구

### 2.1 스마트 홈

스마트 홈은 정보기기가 네트워크를 통해 연결되어 사람과 상호작용하는 미래의 주거환경으로

인간 중심의 서비스를 제공하는 것이다. 이를 위해 스마트 홈은 융합된 콘텐츠를 기반으로 모든 정보기기를 연결하여 홈네트워크 서비스를 가능하게 하는 '홈플랫폼', 네트워크를 통해 주택의 환경에 구애받지 않고 요구하는 품질의 서비스를 보장하는 '유무선 네트워킹' 센서를 통해 생활 속 주변상황을 인식하여 사용자에게 맞춤형 상호작용 서비스를 제공하는 '지능형 서비스' 정보기기가 센싱된 데이터를 기반으로 사용자에서 쾌적하고 경제적인 서비스를 제공하는 '그린 홈서비스' 등이 제공 될 수 있도록 설계되고 있다.

## 2.2 IFTTT

IFTTT(If This Then That)는 인터넷과 컴퓨터에 존재하는 별개의 여러 서비스와 애플리케이션을 특정 상황에서 연속으로 이용하는 패턴이 있을 경우 명령을 지정할 수 있다. 이러한 서비스는 과거 프로그래머의 코딩을 통해서만 가능했고 프로그래밍적 지식이 필요한 일이었으나 IFTTT를 지원하는 애플리케이션들은 누구나 쉽게 많은 서비스의 연동을 자동화 할 수 있다. IFTTT의 상용 방법은 <그림 1>과 같다. Recipe는 IFTTT가 수행할 수 있는 하나의 완전한 지시로 IFTTT의 Recipe는 모두 그림\*과 같은 형식을 갖는다. Trigger는 Recipe가 실행되기 전 충족이 필요한 조건으로 This 부분에 해당하는 조건을 작성한다. Action은 Trigger의 조건이 충족되면 시작되는 행동을 의미하며 That 부분에 해당한다. 예를 들어 문자나 SNS에 글을 업로드 하는 작업들을 수행할 수 있다. Channels는 IFTTT가 연동할 수 있는 모든 서비스를 의미한다. IFTTT의 채널을 보유하고 있는 서비스는 고유의 Trigger와 Action을 가지게 된다. Ingredients Trigger 에서 수집할 수 있는 데이터를 의미한다. 예를 들어 Email의 경우 제목, 내용, 받은 날짜, 전송자의 주소 등의 정보는 수집할 수 있다. Polling Period는 Trigger 조건이 충족되는지를 알아보기 위한 시간 간격을 의미하며 기본적으로 15분으로 지정되어있다.



그림 1. IFTTT Recipe 구조

## III. 선행 연구

본장에서는 IFTTT 기반의 스마트홈 서비스 리모컨 설계를 위해 6개의 푸쉬버튼을 활용하여 버튼 이벤트의 감지와 WiFi로 연결된 TCP/IP 리모컨 서버와의 이벤트 송수신을 확인한다.

6개의 푸쉬버튼의 3가지 종류별 이벤트 입력

(클릭, 더블클릭, 홀드)검출을 테스트하고 5V 리튬이온 배터리 전원부 테스트 및 ESP8266 NodeMCU 기반의 UDP 클라이언트 접속 및 데이터 송수신 테스트를 진행하였다. 테스트를 위한 H/W의 구성은 <그림 2>, <그림 3>과 같다.

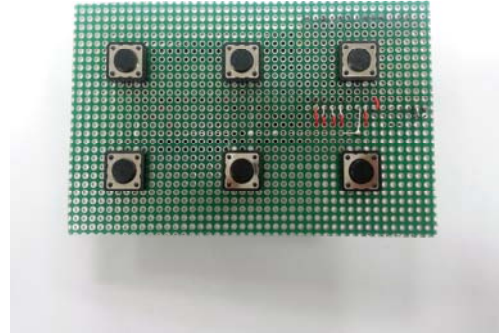


그림 2. 리모컨 하드웨어(앞면)

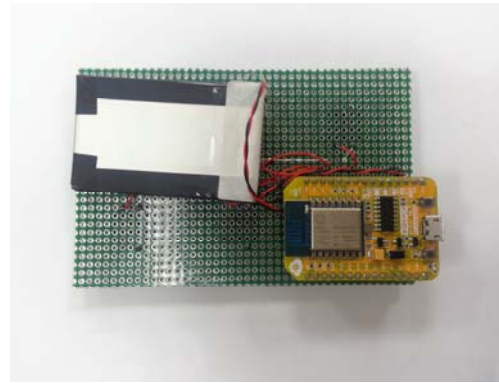


그림 3. 리모컨 하드웨어(뒷면)

테스트를 통해 별도의 라이브러리 활용이나 운영체제 없이 C코드 펌웨어만으로 기능을 구현하여 디바이스의 불안정한 동작을 확인하였고, Sleep 모드가 없어 배터리 소모가 심한 문제로 인해 배터리의 수명이 최대 1주일정도 사용가능한 것을 확인하였다. 또한 디바이스 제어 허브를 제공하는 사물인터넷 제품들도 IFTTT 플랫폼을 통해 제어함에 따라 네트워크 지연이 발생하였고, 별도의 표준 프로토콜 없이 UDP 기반의 데이터 전송방식으로 인해 데이터의 전달에 대한 신뢰성이 떨어졌다.

## IV. 리모컨 하드웨어 설계

본장에서는 WiFi를 통해 인터넷에 연결하고 서비스 서버를 통해 버튼 입력 이벤트를 다른 사물인터넷 디바이스 혹은 서비스로 전송하여 직접 제어할 수 있는 사물인터넷 리모컨을 설계한다.

스마트 홈 리모컨의 하드웨어 구성은 <그림 4>와 같이 NodeMCU를 내장한 WiFi 모듈, 동작 및 네트워크 연결 상태 확인을 위한 LED 모듈, 충전형 배터리 및 제어 입력을 위한 6개의 버튼으로 구성되며 버튼들은 I/O 인터럽트 방식으로 각각의 GPIO에 연결하여 통신한다. 또한 마이크로 USB 포트로 연결된 충전형 배터리를 통해 NodeMCU 및 LED에 전원을 공급한다.

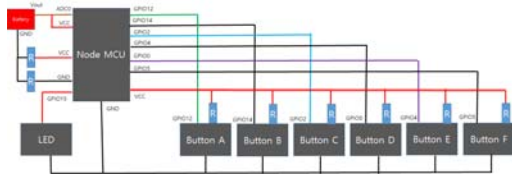


그림 4. 리모컨 하드웨어 구성

펌웨어 모듈의 구성은 <그림 5>와 같다. Timer Manager는 버튼의 클릭 종류(one click, double click, long click)를 검출하여 NodeMCU에 해당 이벤트를 호출한다. WiFi Manager는 최초 통신 시 AP 모드로 동작하여 스마트폰과 통신을 하기 위한 Server Manager와 리모컨 서비스 서버와 통신을 하기 위한 Client Manager로 구성된다. Sleep Manager는 리모컨의 사용유무에 따라 전력 소모량을 줄이기 위한 디바이스의 Sleep 모드를 관리한다. Battery Manager는 NodeMCU의 전원 공급을 위한 ADC 컨트롤 및 충전형 내장 배터리의 충전 상태를 확인한다.

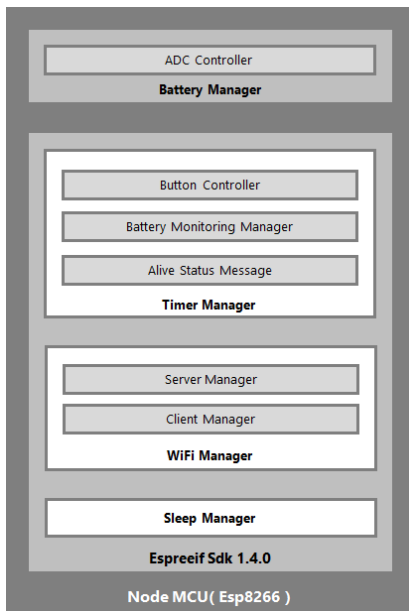


그림 5. 리모컨 펌웨어 모듈 구성

## V. 결 론

본 논문에서는 IFTTT를 활용한 스마트 홈 리모컨을 설계하였다. 선행 연구 결과 확인된 문제점을 해결하기 위해 Deep-Sleep 모드 및 Wake-up 인터럽트 기능을 추가하여 배터리 수명을 극대화하였고, Direct-Control 기능을 추가하여 서비스 서버가 디바이스 제어 허브에 직접 연결하여 네트워크 지연을 최소화하였다. 또한 UDP 기반의 데이터 전송으로 인해 데이터 송수신 신뢰성이 떨어지는 문제를 해결하기 위해 oneM2M 표준 사물인터넷 디바이스용 프로토콜인 MQTT 프로토콜을 적용하여 데이터 송수신간 신뢰성을 높였다. 이로 인해 기존의 다양한 제조사의 사물인터넷 디바이스를 사용할 경우 여러 애플리케이션을 설치하여 제어하는 번거로움이 사라질 것으로 예상된다.

## 참고문헌

- [1] 김계영, 이현동, 조대수, "IFTTT 서비스를 위한 실시간 이벤트 처리 쿨 관리 시스템", 멀티미디어학회논문지, Vol.20, No.8, pp. 1379-1386, 2017
- [2] 이학준, "사물인터넷 기반의 스마트홈", 한국통신학회지, Vol.32, No.4, pp. 44-49, 2015
- [3] 이지혜, 이상원, "사용자 중심의 자동화를 위한 사용 맥락 및 행태 분석 - IFTTT를 중심으로", 한국HCI학회 학술대회, pp. 642-644, 2016
- [4] 박석지, "스마트 IT 시대의 스마트홈 산업 육성을 위한 정책 방향", 한국통신학회 학술대회 논문집, pp. 627-630, 2012