

---

# IoT 센서 흐름 제어 어플리케이션 시스템

임 혁<sup>1</sup> · 유동균<sup>1</sup> · 정도형<sup>1</sup> · 류승한<sup>2</sup> · 정회경<sup>1\*</sup>

<sup>1</sup>배재대학교 · <sup>2</sup>정보통신기술진흥센터

## IoT Sensor Flow Control Application System

Hyeok Lim<sup>1</sup> · Dong-Gyun Yu<sup>1</sup> · Do-Hyeong Jeong<sup>1</sup> · Seung-Han Ryu<sup>2</sup> · Hoe-Kyung Jung<sup>1</sup>

<sup>1</sup>Paichai University · <sup>2</sup>IITP

E-mail : {dlagur1402, eowkdgkelsz, wjdehgud5769}@naver.com, mrryu93@hanmail.net,

hkjung@pcu.ac.kr

### 요 약

IoT(Internet of Things) 시대의 인터넷 데이터는 데이터가 사용자에게 맞게 정보를 공유하여 수행하는 방식으로 바뀌었다. 그러나 기존 IoT 시스템 환경에서는 사용자가 직접 활용하는데 있어 개인의 특성을 고려하지 못하는 문제점과 Dongle 같은 중간 매개체가 있어야 제어가 가능한 문제점들이 있었다.

본 논문에서는 이러한 문제점을 해결하기 위한 방법으로 센서 흐름 제어 어플리케이션을 통하여 사용자가 원하는 특성에 맞게 센서들의 흐름을 제어하고 이로 인해 기존의 IoT 환경에 비해 센서를 편리하게 관리할 수 있게 한다. 이에 따라 사용자는 시간과 장소에 구애받지 않고 어플리케이션을 통해 센서들을 관리함으로써 불필요한 전력 소비를 줄이고 효율적인 센서 제어가 가능할 것으로 사료된다.

### ABSTRACT

Internet data for IoT(Internet of Things) period was changed in such a way that the data is done by sharing information for the user. However, in the existing system IoT environment for the user to utilize the system it has a problem does not take into account the individual characteristics. And there must be an intermediate vectors are capable of controlling problems such as Dongle.

In this paper, through the flow sensor control applications as a way to solve this problem to control the flow of the sensor according to the characteristics desired by the user. Due to this makes it possible to easily manage the sensor compared to conventional IoT environment. Accordingly, the user must manage the sensor through the application regardless of time and place. So it is believed to reduce the unnecessary power consumption is possible effective control sensor.

### 키워드

Application, IoT, Sensor flow, Virtual Sensing

## 1. 서 론

홈 관련 IoT(Internet of Things) 서비스는 사물 인터넷과 스마트 홈이 융합된 분야로써 스마트폰으로 기존에 제공되었던 서비스보다 다양한 스마트 환경으로 지속적인 상승세를 유지하고 있다 [1,2]. 가정 내의 작업들이 이루어지는데 있어서 정해놓은 이벤트와 프레임에 의해서만 동작되는

문제점이 있었다. 가정 내부에서 이루어지는 작업이 대부분이기 때문에 위험 상황을 고려하지 않을 수 없는데 기존 시스템들은 이를 고려하지 않았다[3].

이를 해결하기 위해 본 논문에서는 가정 내 설치되는 다양한 센서들의 데이터 흐름을 사용자 특성에 맞게 제어할 수 있는 어플리케이션 시스템을 제안하였다.

## II. 본 론

제안하는 시스템은 어플리케이션을 통하여 센서의 흐름을 제어한다. 그림 1은 시스템의 구조를 나타내고 그림 2는 시스템의 흐름을 나타낸다.

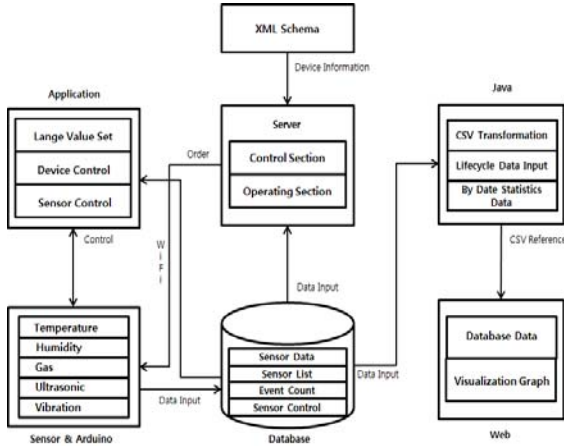


그림 1. 센서 흐름 제어 어플리케이션 시스템 구조도

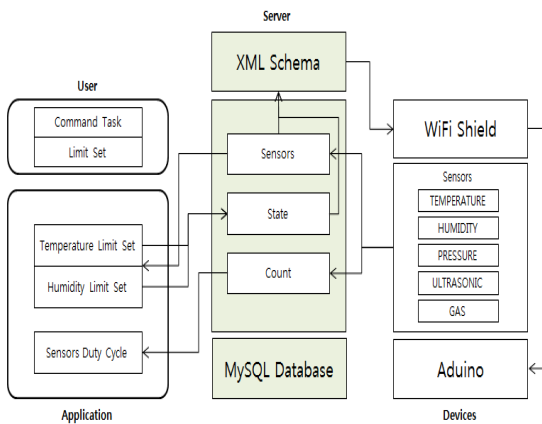


그림 2. 어플리케이션 데이터 흐름도

어플리케이션에서는 사용자가 직접 각 센서의 범위 값을 설정할 수 있고 기기와 센서를 제어할 수 있는 기능을 수행한다. 이러한 센서 데이터와 시간 등 다양한 이벤트로 기기들이 현재 상황을 인식하여 작업을 진행하게 되고 각 작업마다 센서 카운팅을 실시한다. 카운팅 값을 날짜와 시간으로 분석하여 사용률이 저조한 센서의 동작여부를 판단하고 제어할 수 있는 기능을 목표로 수행한다. 또한 사용자가 다시 센서의 동작이 필요할 경우에는 어플리케이션을 통하여 해당 센서의 동작을 수행할 수 있게 제어할 수 있다.

어플리케이션 시스템의 흐름은 사용자가 어플리케이션을 통해서 등록된 센서들을 제어하는데 있어서 각 센서마다 limit 값을 정의한다. 정의한 limit 값에 따라 작업이 수행된다. 각 센서들의 상태는 데이터베이스에 적재되고 디바이스는 해당 센서의 상태 값을 참조하여 동작여부를 결정한다. 작업이 동작되면 각 센서의 Count 여부를 적재하여 센서 사용률을 알 수 있다. 해당 사용률을 참고하여 해당 센서에 대해 대기 상태와 동작 상태를 조절할 수 있으며 불필요하다면 직접 전원을 Off시킬 수 있다. 필요 시 다시 전력을 On시켜 센서를 활성화 할 수 있다. 이로 인해 전력의 손실을 줄일 수 있을 것이다.

## III. 결 론

최근 스마트 환경의 발달로 인해 센서들을 활용한 IoT 환경의 사용률이 점차 증가하고 있다. 기존의 시스템은 센서 데이터의 흐름을 정의해놓고 특정 상황이 발생했을 때 기기의 작업이 실행되는 방식으로 서비스를 제공하고 있어 능동적이지 못한 문제가 있었다.

이를 해결하기 위해 본 논문에서는 센서 데이터의 흐름을 제어할 수 있는 어플리케이션 시스템을 제안하였다. 이를 통해 해당 센서의 상태를 변환할 수 있게 됨으로써 원치 않은 센서 데이터와 전력 사용의 낭비를 줄일 수 있을 것으로 사료된다. 향후 연구로는 어플리케이션과 직접적인 연결을 통해 제어하는 시스템을 구현해야 할 것이다.

## Acknowledgments

This research was supported by The Leading Human Resource Training Program of Regional Neo industry through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT and future Planning(No. 2016H1D5A1911091)

## 참고문헌

- [1] J. H. Kang, H. J. Kim, M. S. Jun, "Market and Technical Trends of Internet of Things." The Korea Contents Society, vol. 13, no. 1, pp. 14-17, 2015.
- [2] J. S. Ryu, H. S. Kim, R. Y. Kim, K. Y. Kim, "3GPP M2M / IoT Standards and Technology Trends." Korea Institute Of Communication Sciences, vol. 32 no. 5, pp. 3-8, 2015.
- [3] S. G. Hong, N. S. Kim, N. S. T. Heo, "A Smartphone Connected Software Updating Framework for IoT Devices." In Consumer Electronics (ISCE), IEEE International Symposium on, pp. 1-2, IEEE, 2015.