

## IoT based Cleaner Control System using Smart Devices

Ye Ho Shin\*

\*Professor, Dept. of AI Computer Engineering, Far East University, Eumsung, Korea

### [Abstract]

In this paper, we implement a control system for an IoT-based backpack type vacuum cleaner using a smart device such as a smart phone or smart watch. The implementation system consists of control module produces, control module programming, and smart device programming. The control module is made of Arduino Nano, HM-10 BLE(Bluetooth Low Energy) module and relays as basic parts. The smart device exchanges signals with the control module via bi-directional BLE communication, which allows it to control the start/stop of the vacuum cleaner. Backpack type vacuum cleaners are effective for cleaning high places that require the use of ladders. However, it is often necessary to take off the backpack type cleaner to start/stop it. The IoT-based vacuum cleaner control system implemented in this paper fundamentally solves the problem by allowing users to control the start/stop of the vacuum cleaner without taking it off.

▶ **Key words:** IoT, Arduino, BLE, Smart Device, relay control

### [요 약]

이 논문에서는 스마트폰 또는 스마트 워치와 같은 스마트 디바이스를 이용한 IoT 기반의 배낭식 청소기 제어 시스템을 구현한다. 구현 시스템은 제어모듈 제작, 제어 모듈 프로그래밍, 스마트 디바이스 프로그래밍으로 이루어져 있다. 이 중 제어 모듈은 아두이노 나노, HM-10 BLE 모듈 및 릴레이를 기본 부품으로 하여 제작하였다. 스마트 디바이스는 제어 모듈과 양방향 BLE 통신으로 신호 교환을 하고 있으며, 이를 통해 청소기의 시작/정지를 제어 할 수 있게 한다. 배낭식 청소기는 사다리 등을 이용해야 하는 높은 장소를 청소할 때 효과적이다. 그러나 배낭식 청소기를 시작/정지하기 위해 메고 있는 청소기를 벗어야 하는 경우가 종종 발생한다. 이 논문에서 구현한 사물인터넷 기반의 청소기 제어 시스템은 청소기를 벗지 않으면서도 청소기의 시작/정지를 제어할 수 있도록 함으로써 문제를 근본적으로 해결하였다.

▶ **주제어:** 사물인터넷, 아두이노, 저에너지 블루투스, 스마트 디바이스, 릴레이 제어

- 
- First Author: Ye Ho Shin, Corresponding Author: Ye Ho Shin
  - \*Ye Ho Shin (snowman@kdu.ac.kr), Dept. of AI Computer Engineering, Far East University
  - Received: 2023. 03. 02, Revised: 2023. 04. 21, Accepted: 2023. 05. 03.

## I. Introduction

작업용 청소기 중에서 배낭식 청소기(backpack type vacuum cleaner)는 작업자가 청소기를 등에 메고 작업할 수 있는 구조이기 때문에 장소나 작업조건에 구애받지 않고 효율적으로 청소하는데 유용하다. 그러나 배낭식 청소기는 청소기의 시작/정지를 실현하기 위해 메고 있는 청소기를 벗어야 하는 경우가 종종 발생한다. 이는 배낭식 청소기의 시작/정지 스위치가 청소기 본체에 부착되어 있어서 발생하는 구조적 문제이다.

이 논문에서는 이와 같은 문제를 해결하기 위해 스마트 폰 또는 웨어러블 디바이스(wearable device)의 일종인 스마트 워치(smart watch)를 이용하여 청소기를 제어하는 청소기 제어 시스템을 개발한다. 개발 시스템은 청소기에 장착된 상태에서 스마트폰 또는 스마트워치와 같은 스마트 디바이스(smart device)로부터 전송되는 신호에 대응하여 청소기의 시작/정지를 수행하는 청소기 제어모듈, 청소기 제어 모듈에 제어 신호를 전송하는 장치인 스마트 디바이스로 구성된다.

스마트 디바이스와 청소기 제어모듈 사이에서 제어 신호를 교환하기 위한 통신 프로토콜로 사물인터넷(IoT : Internet of Things)에서 광범위하게 사용되는 저 에너지 블루투스(BLE : bluetooth low energy)를 이용한다[1]. BLE는 대부분의 스마트 디바이스에 공통으로 탑재되어 있기 때문에 다종의 장치들과 제어 신호를 교환해야 하는 청소기 제어모듈의 통신 수단으로 적합하다[1].

한편 청소기 제어모듈은 오픈소스 기반의 단일보드 마이크로 컨트롤러(single board micro controller)인 아두이노(Arduino)를 이용하여 구현한다. 아두이노는 다수의 GPIO(General Purpose I/O)를 이용하여 디지털 입출력을 직접 제어할 수 있으며, 저렴할 뿐만 아니라 통합개발 환경(IDE : integrated development environment)을 제공하고 있어서 손쉽게 제어 모듈 구현이 가능하다[2].

BLE 통신을 기반으로 스마트 디바이스와 아두이노를 이용하는 청소기 제어 시스템의 구현에 대한 내용을 전개하는 논문의 구성은 다음과 같다. 먼저 2장에서 관련 연구를 기술하고, 3장에서 연구의 도메인인 배낭식 청소기의 제어 환경을 분석한다. 4장에서 개발 시스템의 모델링과 설계를 수행하고, 5장에서 구현 및 검증을 기술한다. 마지막으로 6장에서 결론 및 향후 연구사항에 대해서 논의한다.

## II. Related works

이 논문에서 개발하는 스마트 디바이스를 이용한 IoT기반의 작업용 청소기 제어 시스템은 배낭형 청소기의 시작/정지를 스마트 디바이스를 이용하여 제어할 수 있도록 하는 시스템이다. 이 시스템에서 청소기의 제어를 담당하는 제어 모듈은 오픈 소스 기반의 단일보드 마이크로 컨트롤러인 아두이노를 이용한다. 아두이노는 8비트 RISC(reduced instruction set computer) 아키텍처 마이크로 컨트롤러인 ATmega 시리즈 MCU(micro control unit) 기반의 범용 개발 보드이다[2]. 아두이노 개발보드는 모델에 따라 사용하는 MCU가 다른데 우노(Uno), 나노(Nano)의 경우 ATmega328, 레오나르도(Leonardo)의 경우 ATmega32U4, 메가(Mega)의 경우 ATmega2560 MCU를 탑재하였다[2].

ATmega 시리즈 MCU는 병렬 I/O포트, 내장 시스템 클럭 발진기, 8비트 타이머, 워치독 타이머, UART(universal asynchronous receiver/transmitter), SPI(serial peripheral interface), TWI(two wire interface), 10비트 A/D 컨버터, PWM(pulse width modulation) 출력 등 다양한 I/O 기능을 장착하고 있다[3]. 이 논문에서 청소기 제어 모듈의 메인 컨트롤러를 담당하는 아두이노 나노는 ATmega328 MCU를 탑재하고 있으며, 그렇기 때문에 ATmega328 MCU가 제공하는 다양한 입출력 기능들을 충실히 구현하고 있다[3].

이와 같은 아두이노는 로봇[4, 5], 의료·건강[6, 7], 교육[8, 9], 자동제어[10, 11]를 망라하는 다양한 분야에서 폭넓게 활용되고 있는 상태이다.

한편 스마트 디바이스와 청소기 제어 모듈 사이 통신을 담당하는 BLE는 2010년 발표된 블루투스 4.0에서부터 포함된 기능으로서, 블루투스 기술을 사물인터넷에 적용하기 위한 목적으로 개발 되었다[2]. 스마트폰 및 웨어러블 디바이스에 채택되면서 파급력이 급속도로 증가하게 된 BLE 기술은 스포츠, 의료, 위치 서비스, 디바이스간 네트워크 분야에서 급격한 성장세를 이어가고 있다[12].

## III. Backpack type vacuum cleaner analysis

배낭식 청소기는 진공청소기의 일종으로 작업자가 등에 메고 다니면서 청소를 할 수 있기 때문에 사다리 등을 이용해야 하는 높은 장소와 같이 청소기 본체와 청소기 흡입

부가 멀리 떨어질 수 있는 작업 환경에서 청소 할 때 유용하다. Fig. 1은 전형적인 배낭식 청소기의 구조이다.

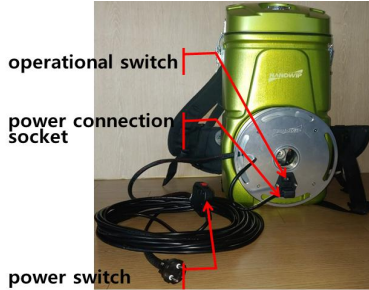


Fig. 1. Backpack type vacuum cleaner

배낭식 청소기는 Fig. 1에서 볼 수 있는 바와 같이 벨트를 통해 등에 짊어질 수 있도록 되어 있다. 청소기의 시작/정지는 Fig. 1에서 확인할 수 있는 바와 같이 청소기 본체에 부착되어 있는 운전 스위치(operational switch)와 전원선 종단부에 설치되어 있는 전원 스위치(power switch)를 이용하여 제어한다.

청소기의 운전 제어를 담당하는 전원 스위치와 운전 스위치는 서로 직렬로 연결되어 있어 둘 중 하나만 OFF이면 청소기는 작동하지 않는다. 따라서 청소기를 매고 있는 상태에서 청소기 본체 운전 스위치가 OFF 상태라면 청소기 운전을 제어하기 위해 청소기를 벗어야 하는 불편함이 발생한다. 이러한 문제는 청소 작업의 효율성을 상당히 저하시키는 원인이 된다. 다음 장에서 이와 같은 문제를 해결하기 위한 청소기 제어 시스템을 모델링하고 설계한다.

## IV. Cleaner control system modelling

### 1. Cleaner control module modelling

배낭식 청소기의 운전 제어는 Fig. 1에서 나타낸 바와 같이 전원선에 대한 ON/OFF 기능을 갖는 스위치를 직렬로 연결해 놓은 것이 전부이다. 아울러 제어 구조가 모두 유선으로 되어 있기 때문에 상황에 따라 청소기를 벗었다 메었다를 반복해야 하는 경우도 발생한다. 이와 같은 문제는 BLE를 기반으로 한 무선 제어 기법의 도입을 통해 해결할 수 있는데, 청소기 본체에 장착되어 있는 운전 스위치와 협업을 하면서 스마트 디바이스를 통해 전송되는 제어 신호를 기반으로 청소기를 시작/정지할 수 있도록 하는 제어 모듈 구현을 통해 달성할 수 있다. 다음의 Fig. 2는 제어 모듈의 도입을 통한 문제해결의 기본 개념을 보여 준다.

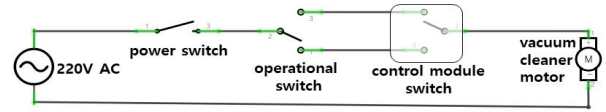


Fig. 2. A concept of cleaner control using control module

Fig. 2에서 전원 스위치는 메인 전원을 시스템에 공급하는 역할을 한다. Fig. 2에서 운전 스위치를 유지하는 이유는 물리적 스위치에 익숙해져 있는 작업자들의 특성을 반영하기 위해서이다. 이렇게 운전 스위치와 제어 모듈을 함께 운영하기 위해 운전 스위치와 제어 모듈 모두 3로 스위치(3-way switch)로 구성해야 한다. 이렇게 함으로써 전원 스위치가 ON인 상태에서 운전 스위치와 제어 모듈, 어느쪽을 이용하든 청소기의 작동을 ON/OFF 할 수 있기 때문에 배낭형 청소기를 벗지 않고도 청소기의 시작/정지를 제어할 수 있도록 한다.

### 2. Design of cleaner control module

청소기 제어 시스템은 앞에서 제시한 청소기 제어 모듈과 스마트 디바이스에서의 제어 앱이 쌍으로 구성된다. 청소기에 부착된 상태에서 스마트 디바이스로부터 BLE를 통해 전송되는 제어 신호에 따라 청소기 제어를 담당하는 제어 모듈은 아두이노 나노, 릴레이 및 BLE 통신 모듈을 이용하여 구현한다. Fig. 3은 아두이노 나노의 레이아웃과 핀맵을 보여준다.

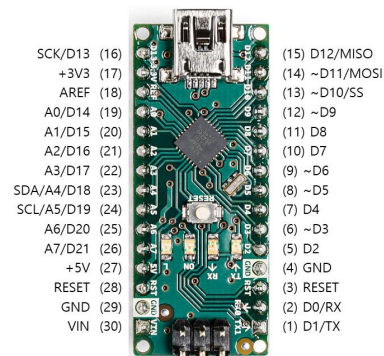


Fig. 3. Aduino Nano Board Layout & Pinmap

Fig. 3에서 확인할 수 있는 바와 같이 아두이노는 각각의 핀에 복수의 기능이 할당되어 있다. 이는 ATmega328 마이크로 컨트롤러로부터 유래하는 기능이다[2].

이 논문에서 구현한 청소기 제어 모듈은 3로 스위치 기능을 구현하기 위해 두 개의 릴레이를 제어해야 하고, 청소기의 운전 상태를 모니터링 하기 위해 한 개의 릴레이를 판독해야 한다. 아울러 스마트 디바이스와 BLE 통신을 위

한 HM-10모듈과 아두이노 사이에 RS232 통신도 해야 한다. 이를 위한 핀 할당을 다음의 Table 1.과 같다.

Table 1. Arduino pin assignment for cleaner control module

name	assignment	direction	role
D4	Sensor Relay NO	Digital In	sensing cleaner stop
D5	Sensor Relay NC	Digital In	sensing cleaner run
D6	SoftwareSerial Tx	Serial Out	to HM-10
D7	SoftwareSerial Rx	Serial In	from HM-10
D8	Relay Switch 1	Digital Out	Relay 1 switching
D9	Relay Switch 2	Digital Out	Relay 2 switching

Table 1.에서 D4와 D5에 연결된 센서 릴레이(sensor relay)의 상태를 검출하기 위한 것으로서 하나의 극점 (pole)에 두 개의 접점(contact point)을 형성하는 SPDT(single pole double throw) 형 릴레이 이다. 센서 릴레이의 NC(normally closed) 접점은 D4에 연결하고, NO(normally open) 접점은 D5에 연결한다. D6과 D7은 아두이노와 HM-10 모듈 사이에 UART 통신을 위한 연결로 아두이노 IDE에서 제공하는 SoftwareSerial 클래스를 이용하여 UART를 추상화한 핀이다. 마지막으로 제어 모듈에 의해 제어되는 3로 스위치 로직을 구현하기 위해 두 개의 SPST(single pole single throw) 릴레이를 이용하는 데, 이를 위해 D8과 D9를 할당한다.

Table 1.에서 확인할 수 있는 핀 할당 외에도 HM-10 모듈의 전원 공급을 위한 5V출력과 GND 핀의 연결 등 부가적 핀 할당이 있으나, 이는 전기 회로 구성을 위한 기본 요소로서 제어모듈 프로그램 개발과 무관한 부분은 Table 1.에서 제외하였다. 이러한 조건에서 이들 구성요소를 반영한 청소기 제어 모듈의 회로도도 Fig. 4와 같다.

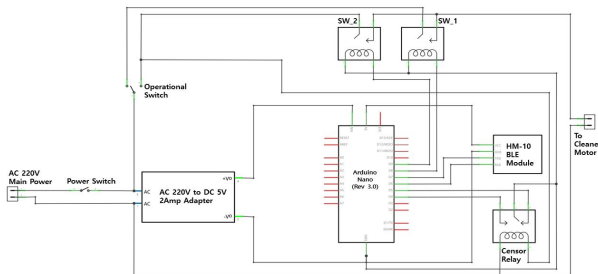


Fig. 4. Circuit diagram for cleaner control module

Fig. 4에서 중앙의 아두이노 나노에 있는 핀들을 Table 1.의 할당에 맞추어 회로를 구성하였으며, 릴레이스위치 역할을 하는 SW\_1과 SW\_2는 운전 스위치와 3로 스위치 로직으로 연결되었음을 확인할 수 있다. 회로 운전을 위한

전원은 AC 220V 메인 전원으로부터 유도된 DC 5V 전원을 이용하여 구동하도록 하였으며, 최종적으로 청소기 모터를 제어하도록 구성하였다.

다음의 Fig. 5는 Fig. 4의 제어 모듈 회로에 나타나는 각종 릴레이 및 스위치의 접점에 대한 범례와 동작을 통해 실행될 수 있는 상태를 나타낸 것이다.

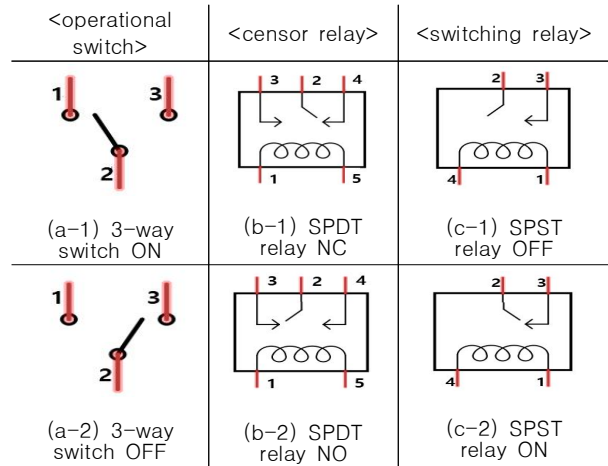


Fig. 5. 3-way switch and relay contact legend and status

Fig. 5는 Fig. 4에서 운전 스위치를 구성하는 3로 스위치, 센서 릴레이를 구성하는 SPDT 릴레이 및 제어 모듈의 3로 스위치 논리 구성에 참여하는 두 개의 SPST 릴레이의 접점 범례를 요약한 것이다. Fig. 5에서 신호원(signal source)으로 작동하는 2번 접점에 철편(hinge)이 연결되어 있으며, 이 철편이 수동조작 또는 코일에 의해 작동하면서 스위칭 작용을 하게 된다. 논의를 직관적으로 전개하기 위해 Fig. 5의 범례 및 상태에서 스위칭동작과 스위칭 동작을 통해 구현되는 청소기 제어논리에 대한 사항을 Table 2.에 정의하였다.

Table 2. Definition of cleaner control logic

No	operational switch	censor relay	relay switch			cleaner status
			SW_1	SW_2	상태	
①	null	null	OFF	OFF	OFF	STOP
②	ON	NC	OFF	ON	OFF	STOP
③	ON	NC	ON	OFF	ON	RUN
④	OFF	NO	ON	OFF	ON	STOP
⑤	OFF	NO	OFF	ON	OFF	RUN

Table. 2에서 상태번호 ①번을 나타내는 1행의 내용 중 운전스위치와 센서릴레이의 값을 null로 규정하였는데, 이는 이 두 신호가 의미 없음을 나타낸다. 즉 ①번 상태는 릴레이 스위치를 구성하는 SW\_1과 SW\_2를 모두 OFF 시킴으로써 청소기의 정지 상태를 강제로 지정한 것이며, 이

때 운전 스위치와 센서 릴레이의 상태는 청소기의 정지 상태에 아무런 영향을 주지 않는다.

Table 2.의 ①번 상태에서 센서 릴레이가 검출하는 운전 스위치의 상태에 따라 ② 또는 ④로 자동 전이가 발생하며, ② 또는 ④의 상태가 운전을 위한 준비 상태가 된다.

한편 준비 상태까지 초기 설정이 완료된 후 운전 스위치와 릴레이 스위치를 이용한 청소기의 운전/정지 제어는 운전 스위치와 릴레이 스위치 어느 곳에서든 관계없이 수행할 수 있다. 즉 운전 스위치를 통해서도 운전/정지를 실현할 수 있고, 릴레이 스위치를 이용해서도 운전/정지를 실현할 수 있으며, 운전스위치와 릴레이 스위치를 번갈아 사용하여 운전/정지를 실현시키는 것도 가능하다. 다음의 Fig. 6은 Table 2.의 상태들 사이에 발생하는 상태 전이를 그래프로 표시한 것이다

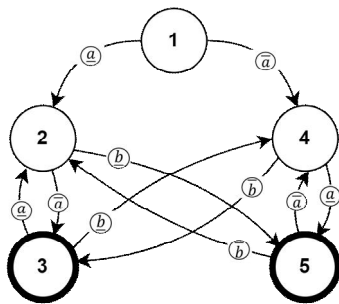


Fig. 6. State transition graph for cleaner control module

Fig. 6에서 굵은 동그라미로 표시된 상태가 운전 상태이고, 나머지는 모두 정지 상태이다. 또한 간선에 표시한 레이블 중  $\bar{a}$ 는 SW\_1이 OFF로 SW\_2가 ON으로 설정하는 상태전이이고,  $a$ 는 SW\_1이 ON으로, SW\_2가 OFF로 설정하는 상태 전이이다. 마찬가지로  $\bar{b}$ 는 운전 스위치를 ON으로 설정하는 상태전이이고,  $b$ 는 OFF로 설정하는 상태전이 이다.

따라서 초기 상태인 ①에서 준비상태인 ②또는 ④로 전이되는 것은 아두이노의 제어에 의한 SW\_1 및 SW\_2의 조작을 통해 정지 상태로 전이되는 것을 의미하고, 이 ② 또는 ④상태가 청소기 가동 준비가 완료 된 상태이다. ② 또는 ④에서 시작하여 운전스위치의 조작을 통해서든 아니면 스마트디바이스를 통한 릴레이 스위치를 이용하는 운전과 정지를 반복하여 수행하게 된다. 그리고 정지 상태인 ②와 ④사이 또는 운전 상태인 ③과 ⑤사이의 상태전이는 운전 스위치와 릴레이 스위치를 동시에 작동시켜야 하는 상황으로서 물리적으로 발생 불가능한 상태전이이기 때문에 직접 전이되는 경로는 발생하지 않는다.

## V. Implementation and Evaluation

### 1. Implementation of cleaner control module

스마트 디바이스를 이용한 IoT 기반의 배낭형 청소기 제어 시스템의 구현은 제어모듈 제작, 제어 모듈 프로그래밍, 스마트 디바이스 프로그래밍으로 이루어져 있다. 이 중 제어 모듈 제작은 아두이노 나노, HM-10 BLE 모듈, SPDT 및 SPST 릴레이, AC 220V to DC 5V 아답터를 기본부품으로 하여 제작하였다. 다음의 Fig. 7은 제어모듈의 제작 결과와 이를 청소기에 결합 한 사진이다.

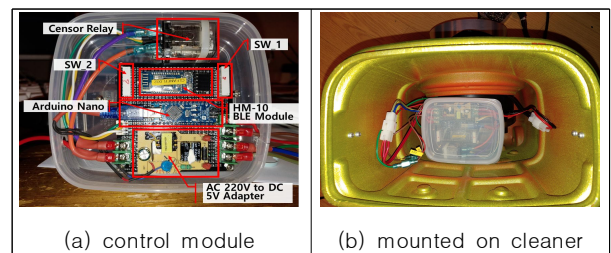


Fig. 7. Picture of implemented control module

Fig. 7의 (a)는 청소기 내부에 장착하기 위해 팩키징한 제어모듈의 사진이고, (b)는 (a)를 청소기 내부에 장착한 모습이다. Fig. 8은 Table. 2 및 Fig. 6의 제어 논리를 구현하는 아두이노 프로그램의 플로우차트이다.

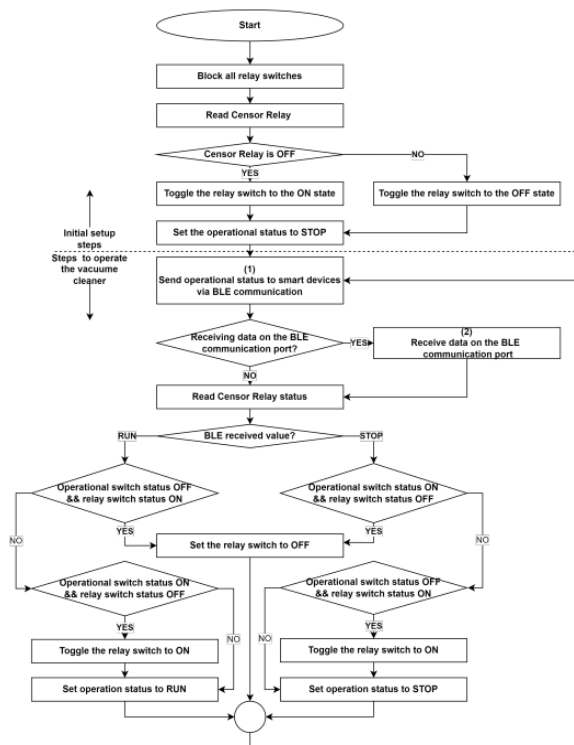


Fig. 8. Flowchart of cleaner control module process

Fig.8 에서 중간의 점선 부분 윗쪽은 Fig. 6의 상태번호 ①에서 시작하여 ② 또는 ④ 로의 상태 전이를 구현하는 초기 설정 부분이다. 그런 다음 점선의 아래 부분으로 아두이노의 특성을 반영하여 종료가 없는 무한루프를 구현하고 있다.

Fig. 8에서 (1)로 표시된 처리가 청소기 제어모듈에서 인지하고 있는 청소기의 상태를 스마트 디바이스에 전송하는 처리이고, (2)로 표시된 처리에 의해 스마트 디바이스로부터 제어 신호를 전송받도록 하고 있다. 이렇게 (2)를 통해 수신한 제어 신호와 제어모듈 내부에서 인지하고 있는 제어 상태를 조합하여 Fig. 6의 상태에 따라 청소기를 제어하는 과정이 Fig. 8의 프로세스 흐름이다.

2. Implementation of smart device app

스마트 디바이스 앱은 스마트폰과 스마트 워치 앱으로 구분된다. 폰과 워치 모두 안드로이드 환경에서 작동하는 디바이스를 이용하기 때문에 기본 프로세스의 구성은 동일하고, 인터페이스 부분만 디바이스의 특성을 반영하여 구현한다. 다음 Fig. 9는 안드로이드 환경의 스마트 디바이스에서 BLE통신을 수행하는 앱의 클래스 다이어그램이다.

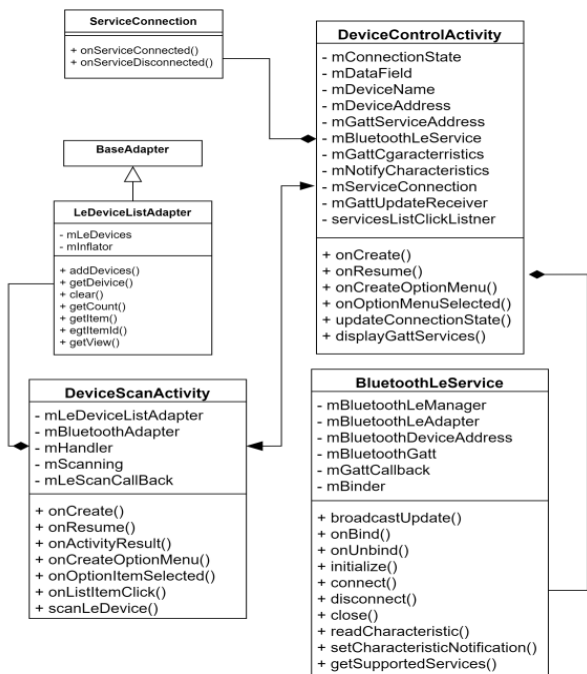


Fig. 9. Smart device app class diagram under android environment

Fig. 9에서 DeviceScanAvtivity는 BLE 장치를 스캔하는 모듈이고, BluetoothLeService는 BLE장치와의 연결 및 데이터 통신 등을 수행하는 역할을 담당한다. 그리고

DeviceControlActivity는 BluetoothLeService 모듈과의 협업을 통해 안드로이드 디바이스에서의 작업 수행을 총괄한다.

다음 Fig. 10은 안드로이드 기반의 스마트 폰과 스마트 워치에서의 작동 화면을 보여준다.

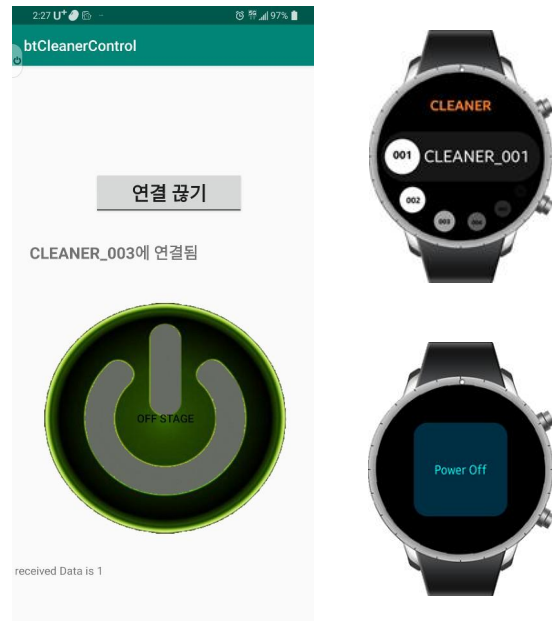


Fig. 10. Smart phone and smart watch initial screen shot inder android environment

Fig. 10의 스마트폰과 스마트 워치의 화면에서 중앙의 파워 버튼을 클릭하면 ON/OFF로 토글을 하며 ON일 경우 0을 OFF일 경우 1을 BLE통신을 통해 청소기에 전송하고, 청소기로부터 현재 청소기가 작동하는지 여부를 전송 받아 화면에 ON/OFF를 표시하도록 구현 하였다.

3. Test and evaluation

구현 시스템은 BLE를 기반으로 스마트 디바이스를 이용하여 배낭식 청소기의 운전을 제어하는 시스템이다. 전원을 인가하고 나면 청소기 제어모듈의 부팅 과정을 거친 후 Table 2.의 ①번 상태를 형성한 다음 운전스위치의 상태 검사 후 청소기의 운전 상태를 OFF 상태로 만들기 위해 Table 2.의 ② 또는 ④의 상태를 형성하는 것이 부팅 단계에서 발생하는 과정이다.

청소기 제어 모듈의 부팅이 완료되고 난 다음 스마트디바이스와 BLE 통신을 위한 연결 절차를 진행해야 한다. BLE통신을 위한 연결은 BLE 스캐너에 의한 스캔을 통해 발견한 장치 목록을 형성한 다음 목록 중에서 연결대상 장치를 선택함으로써 연결을 수행한다. 다음의 Fig. 11은 장치 스캔의 결과와 초기상태 및 운전상태를 보여준다.

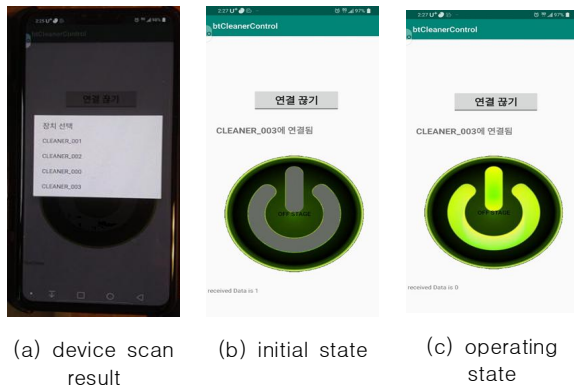


Fig. 11. Pictures of device scan result, initial state and operating state

Fig. 11의 (a)장치스캔결과에서 장치를 선택하면 (b)초기상태로 전이가 되는데, 이는 Table 2.의 ①에서 ② 또는 ④로 전이되는 단계까지를 나타낸다. 이 상태에서 중앙의 스위치 아이콘을 클릭해서 청소기의 가동/중지를 전환해 가면서 청소기를 제어하는데, 중앙의 스위치 아이콘 클릭에 따라 (c)에서 (b)로 또는 (b)에서 (c)로 스위칭이 이루어진다. 이와 같은 조건에서 청소기 제어모듈의 부팅 과정 및 청소기 제어모듈 부팅 후 스마트 디바이스 연결에 소요되는 시간을 측정해 볼 수 있다. 시간 측정은 청소기를 냉장고에 넣어서 청소기 표면온도  $0^{\circ}\text{C}$ 일 때 측정하고, 3시간 정도 방치한 다음 평상 온도에서 측정하는 방식으로 진행하였으며, 이를 5회 반복하여 측정한 평균값을 제시한다. 다음의 Table 3.은 시간 측정의 결과이다.

Table 3. Result of time measurement (unit:millisecond)

scope	0oC			normal temperature		
	avg	max	min	avg	max	min
booting of cleaner controller module	1235	1298	1175	1234	1276	1174
BLE device scan	4528	4925	4239	4438	4872	4196

Table 3.에서 제어모듈 부팅은 1초 조금 더 걸리는 부팅 시간을 소모 하였다. 제어모듈의 아두이노 나노 부팅시간은 이보다 덜 걸리겠지만, BLE 통신을 위한 HM-10 모듈과의 연결 및 초기 SW\_1과 SW\_2에 대한 제어 연산을 완료하는 시간까지 포함하기 때문에 Table 3.과 같은 시간을 소모하는 것으로 판단할 수 있다. 아울러 스마트 디바이스의 장치 스캔에 소모되는 시간은 4.5초 정도 소요되는 것으로 측정되었는데, 이는 Fig. 9의 안드로이드 프로그램에서 스캔 한계시간으로 설정한 10초 이내에서 스캔이 완료된 상황이어서 적절한 시간 내에 스캔이 완료됨을 확인할 수 있다. 아울러  $0^{\circ}\text{C}$ 일 때와 상온일 때 사이의 시

간 차이가 크지 않은 것으로 확인 되었는데, 이는 개발 시스템의 운용 환경이  $0^{\circ}\text{C}$  이상의 환경에서는 큰 문제없이 활용할 수 있음을 보여주는 것이라 할 수 있다. 여기서 상온의 온도 범위는 측정 시간대에 따라 다른데  $18^{\circ}\text{C} \sim 25^{\circ}\text{C}$  사이의 온도 범위에서 측정 하였다.

## VI. Conclusions

이 논문에서는 IoT분야에서 광범위하게 활용되는 아두이노와 BLE 프로토콜 및 스마트 디바이스를 이용하여 청소기의 시작/정지 제어를 할 수 있는 시스템을 구현하였다. 시스템 구현의 대상이 되는 배낭식 청소기는 사다리 등을 이용해야 하는 극한 환경에서 작업 효율의 향상을 위해 사용되고 있으나, 배낭식 청소기의 구조적 문제로 인해 작업 도중 청소기를 메었다 벗었다를 반복해야 하기 때문에 작업의 효율성을 떨어뜨리는 문제가 발생한다.

이 논문에서 구현한 스마트 디바이스와 IoT를 기반으로 하는 청소기 제어 시스템은 이와 같은 문제를 원천적으로 해결할 수 있는 시스템이다. 아울러 제어 도구로 평소 친숙하게 활용하면서 누구나 소유하고 있는 스마트폰 등의 스마트 디바이스를 이용함으로써 작업자들의 친밀도를 극대화 할 수 있도록 하였다.

구현 시스템에 대한 실험 결과 청소기 본체에 장착되어 있는 운전 스위치를 이용하는 방식과, 스마트 디바이스를 이용하는 방식 모두 원활하게 동작하였으며, 실험 결과 섭씨  $0^{\circ}\text{C}$  이상의 온도에서 편차 없이 작동함을 확인할 수 있었고, 어떠한 경우에도 메고 있는 청소기를 벗지 않고 시작/정지 제어가 가능함을 확인 하였다. 이는 [4]에서 제시하는 자율주행 또는 로봇 제어등과 같은 고도의 정밀도를 요구하는 솔루션에 해당하지는 않지만, 최소한의 개선으로 작업의 효율성을 극대화하는데 효과적인 도구로 활용할 수 있는 수준의 솔루션으로서 적합한 사례라 할 수 있다.

한편 이 논문에서 구현한 시스템은 안드로이드 기반의 스마트 디바이스만을 대상으로 하였다. 따라서 IoT 환경의 아이폰에서는 적용할 수 없다. 그러므로 아이폰에서도 적용할 수 있도록 추가 개발을 할 필요가 있으며, 스마트 디바이스에서 청소기 제어 앱 실행에 의한 전원 소모 최소화 방안에도 대해서도 추가 연구가 필요하다.

## ACKNOWLEDGEMENT

This work was supported by the 2021 Far East University Research Grant(FEU2021R05)

## REFERENCES

- [1] Kevin Townsend, Carles Cufi, Akiba, Robert Davidson, "Getting Strated With Bluetooth Low Energy", O'rReilly, pp. 1~20, April 2014.
- [2] Kyung Yong Hur "Arduino with IoT", Jpub, pp. 12~32, January 2018.
- [3] Derk Yong Yun, "Conquest of the AVR ATxmega family", Ohm, pp. 11~24, May 2015.
- [4] Thomas Bräunl, "Embedded Robotics-From Mobile Robots to Autonomous Vehicles with Raspberry Pi and Arduino, 4th Edition". Springer, pp. 3-486, March 2022. DOI:https://doi.org/10.1007/978-981-16-0804-9
- [5] Anirban Kumar, Heshalini Rajagopal, "Automated Seeding and Irrigation System using Arduino" J. Robotics Netw. Artif. Life Vol. 8, No. 4, pp. 259-262, March 2022. DOI:10.2991/jrnal.k.211108.006
- [6] Leozitor Floro de Souza, Fábio Iaione, Shih Ting Ju, "Reconfigurable Arduino Shield for Biosignal Acquisition", International Work-Conference on Bioinformatics and Biomedical Engineering pp. 241-255, June 2022. DOI:10.1007/978-3-031-07704-3\_20
- [7] Fengzhi Dai, Haokang Wen, Jichao Zhao, Hongbo Hao, Qianqian Zhang, "Design of an Instrument for Measuring Heart Rate and Blood Oxygen Based on Arduino", J. Robotics Netw. Artif. Life, Vol. 7, No. 4, pp. 275-278, March 2021. DOI:10.2991/jrnal.k.201215.014
- [8] Ugur Sari, Hüseyin Miraç Pektas, Ömer Faruk Sen, Harun Çelik, "Algorithmic thinking development through physical computing activities with Arduino in STEM education", Educ. Inf. Technol. Vol. 27, No. 5, pp. 6669-6689, January 2022. DOI:10.1007/s10639-022-10893-0
- [9] Chi-Cheng Chang, Yiching Chen, "Using mastery learning theory to develop task-centered hands-on STEM learning of Arduino-based educational robotics", psychomotor performance and perception by a convergent parallel mixed method. Interact. Learn. Environ. Vol. 30, No. 9, pp. 1677-1692, June 2022. DOI:10.1080/10494820.2020.1741400
- [10] Pedro E. Melin, Carlos R. Baier, Eduardo E. Espinosa, Javier Riedemann, José R. Espinoza, Rubén Peña, "Study of the Open-Source Arduino DUE Board as Digital Control Platform for Three-Phase Power Converters", IEEE Access 10, pp. 7574-7587, December 2022. DOI:10.1109/ACCESS.2021.3138705
- [11] Dan Hofstetter, Eileen Fabian, Dorian Dominguez, A. Gino Lorenzoni, "Automatically Controlled Dust Generation System Using Arduino", Sensors, Vol. 22, No. 12, pp. 4574, June 2022. DOI:10.3390/s22124574
- [12] Bluetooth SIG Report, "2021 Market Trend Update", https://www.bluetooth.com/wp-content/uploads/2021/01/2021-Bluetooth\_Market\_Update.pdf

## Authors



Ye Ho Shin, received the BS degrees in Dept. of Computer Science from Kunsan National University, Korea, in 1996. And received the MS. and Ph.D. degrees in Dept. of Computer Science from Chungbuk

National University, Korea, in 1998 and 2002, respectively. Dr. Shin joined the faculty of the Department of Information and Communication Engineering at Far East University, Eumsung, Korea, in 2002. He is currently a Professor in the Departemnet of AI Computer Engineering, Far East university. He is interested in embeded system, big data, machine learning.