

# 아두이노를 이용한 가로등 제어 시스템

이주승 이영준 윤지열  
한국산업기술대학교 컴퓨터공학과

## Streetlights control system by using Arduino

Juseung Lee Youngjun Lee Jiyeol Yoon  
Dept. of Computer Engineering, Korea Polytechnic University

### 요 약

국내 전력 소비량은 매년 갱신되고 있다. 한국 전력에서 발표한 자료에 따르면 특히 공공 서비스 분야와 생산부문이 지속적인 증가추이를 보이고 있다. 공공 서비스의 전력 공급에 더 나은 시스템을 구축하는 것이 현실적인 대안이라고 생각해 그중에 가로등 시스템을 목표로 잡았다. 기존의 CDS(광 전도 셀)로 관리해 오던 가로등을 센서와의 통신을 이용해 지나갈 때 방향성을 고려해서 밝혀 주고 단순히 조도에 맞춰 켜고 끄는 것이 아닌 날씨와 환경적인 요소까지 고려하여 조도를 조절하는 스마트 가로등을 축소형 도로 땅에 구현 하였다.

## 1. 서 론

자원이 없는 국가라면 어느나라이더라도 예민한 문제가 아닐 수 없다. 전기의 소비는 곧 국민의 생활의 질과 연관되어 있을 뿐만 아니라 한정된 자원을 효율적으로 이용하는 것이 후대를 위하여 중요한 일이다 하기 때문이다. 무엇보다도 우리가 사용하는 공공재를 좀 더 아끼고 효율적으로 잘 사용할 수 있게된다면 좀 더 나은 삶을 영유할수 있을것이라고 생각 된다.

우리가 거리에서 쉽게 볼 수 있지만 무심코 지나가면서 신경쓰지 않았던 가로등을 선택했던 것은 예상보다 많은 전력을 소비 할수도 있을것이라 생각하였다.

한 달의 가로등 개당 전력원가는 시간당 55원정도로 파악 되고 있다.그리고 서울 한복판에 있는 가로등을 계산하게되면 천문학적인 금액으로 볼수 있다.그것을 좀 더 아끼고 스마트 한 시대에 걸맞는 가로등을 만들기 위해 구현하였다.

## 2. 관련 연구

### 2.1 기존 유사 앱과의 비교

아래의 <표1>은 본 논문의 앱과 기존의 유사 앱들간의 기능에 대해서 비교하고 있다.

관련 제품/프로그램	장점	단점
빛고을 스마트가로등	가로등을 이용한 신고 시스템 범죄율 감소가능성이 높음	범위가 넓기 때문에 인프라구축이 잘되어있어야 한다
LED 가로등 관리	가로등을 컨트롤 한다는 아이디어로 전기절약감소	가로등을 키고 켜는 기능밖에없음

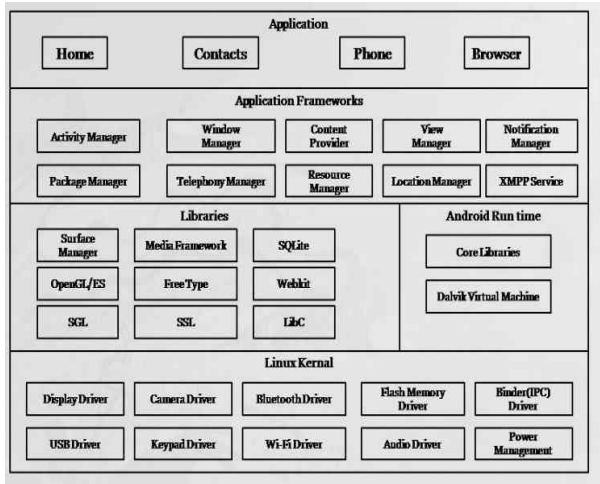
<그림1> 기존 관련 사례 제시

아직까지는 가로등에 대한 어플이나 관심은 매우적다. 그 이유는 여러 가지가 존재하겠지만 공공재 이기때문에 중요성을 못느낀다거나 실제로 전기 사용량이 적다고 여겨 지거나 생활에 불편함이 없다고 여겨지기 때문이 아닐까 싶다. 그래도 현재 시중에 존재하는 가로등 관련 어플은 두가지가 있다. 하나는 가로등을 이용한 위치기반을 이용한 범죄 신고 어플이고 하나는 우리가 구현한것과 비슷한 가로등 전력을 관리하는 어플이다.

기존 시장에 나온 어플보다가 우리 제품은 차별성을 두게되었고 현대사회에 추세인 IOT산업을 떠올리게 되어 가로등이 자동으로 사람을 인식하여 빛을 켜준다거나 시간때를 파악하

여 자동으로 OFF하는 기능 계절이나 습도등 주변 환경을 파악하여 적절하게 빛의 세기를 조절하는등 여러 가지 기능을 추가하게 되었다.

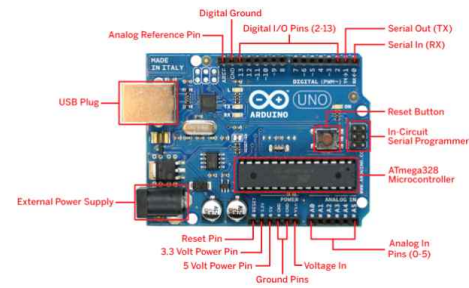
## 2.2 안드로이드 플랫폼



안드로이드 개발 환경 구성요소

- Java JDK:안드로이드는 기본적으로 Java기반이기 때문에 JVM을 설치해야 한다
- Eclipse 개발도구 : Java기반으로 개발을 위한 개발 도구이다.
- ADT(Android Development Tool) : 안드로이드 플랫폼에서 개발을 위한 개발 도구이다
- Android SDK(Software Development Kit) : 안드로이드 플랫폼의 API를 호출하기 위한 엔진

### 아두이노 개발 환경 구성요소



## 2.3 개발 환경

본 논문의 시스템은 3가지로 구성 되어있다.하나는 가로등

역할을 하는 아두이노,정보를 보관하고 관리할 수 있는 서버, 마지막으로 그것을 컨트롤하고 직접 눈으로 확인 할 수 있도록 앱어플리케이션을 만들었다.

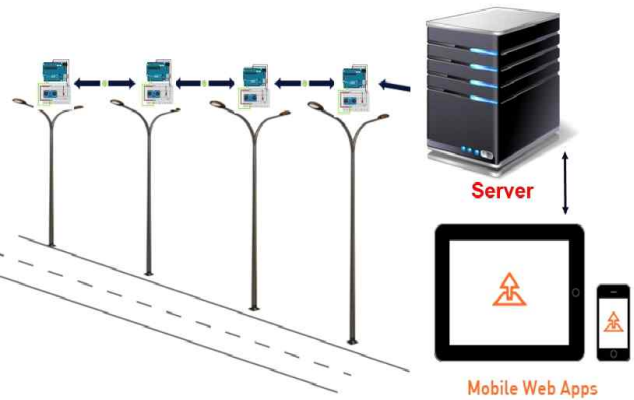
첫 번째로 아두이노는 아두이노 내에서 사용되는 스케치라는 프로그램을 사용하였고 앱어플리케이션과 연동이 필요하기 때문에 자바를 사용하게 되었고 실제로 앱어플리케이션은 Java SE Development Kit (JDK8) 설치, Android studio를 설치하였다.

세 번째로 서버는 Tomcat6를 설치하였고 사용한 웹언어로는 PHP를 사용하였다. 그리고 데이터베이스를 위한 SQL을 설치하여 구현하였다.

마지막으로 시스템의 운영체제는 윈도우10을 사용하였다.

## 3. 세부 설계 및 구현

### 3.1 시스템 구성



<그림 2> 가로등 컨트롤 시스템 구성도

가로등을 표현하기위해서는 아두이노를 사용하였고 아두이노에는 LED전등과 사람을 인식하기위한 인지센서가 달려있다. 어플리케이션을 통하여 LED(빛의세기)를 조절할 수 있도록 하였고 그러한 정보를 데이터베이스 웹서버에 담도록 하였다. 데이터베이스에는 빛이 켜졌던 시간과 꺼진시간 빛의 세기가 어느정도인지를 체크해볼수있어서 사람이 많이다니는 시간대 라던지 사람이 얼마나 많았는지를 체크할수 있도록 하였다.

3.2 핵심 알고리즘

3.2.1 아두이노

```
BT_LED

#include <Arduino.h>
#define led1 2
#define led2 3
#define led3 4
#define led4 5
#define led5 6

SoftwareSerial bluetoothSerial(10, 11); // 블루투스 통신을 위한 핀
unsigned char chByte = 0;
char recByte; // 수신받은 데이터를 저장할 변수

int ledPower = 0; // LED 밝기를 저장할 변수

int led1 = 5; // 1번 LED
int led2 = 4; // 2번 LED
int led3 = 3; // 3번 LED
int led4 = 2; // 4번 LED
int led5 = 1; // 5번 LED

BT_LED

if (led_state_4 == 1) {
  analogWrite(led_4, led_power);
} else {
  analogWrite(led_4, 0);
}

if (led_state_5 == 1) {
  analogWrite(led_5, led_power);
} else {
  analogWrite(led_5, 0);
}

if (bluetoothSerial.available()) { // 블루투스 신호가 들어오면
  chByte = bluetoothSerial.read(); // chByte 변수에 데이터를 저장한다.
  Serial.print(chByte, DEC);
  Serial.print("\n");

  if (chByte == 101) { // 만약 데이터값이 101이면
    if (bluetoothSerial.available()) { // 그뒤에 데이터를 받아온다 (led 밝기)
      chByte = bluetoothSerial.read();
      Serial.print(chByte, DEC);
      Serial.print("\n");
      ledPower = chByte * 2.5; // led_power 변수에 수신된 데이터 (1-100)값을 *2.5해서 저장한다.
    }
  }

  if (chByte == 111) { // 수신된 값이 111이면
    led_state_1 = 1; // led_state_1을 1로 저장
  }
  if (chByte == 112) { // 수신된 값이 112이면
    led_state_1 = 0; // led_state_1을 0으로 저장
  }

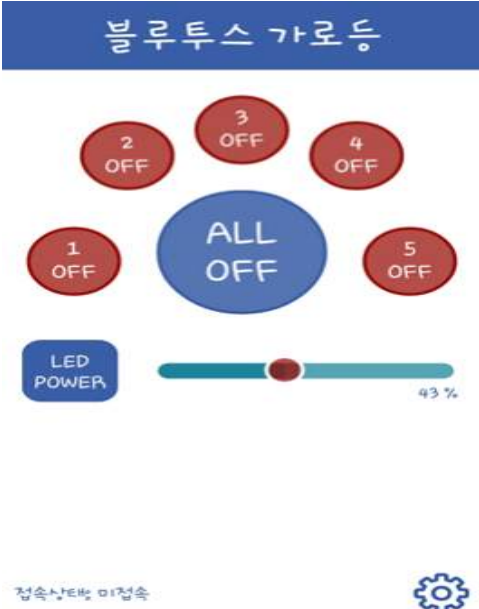
  BT_LED

  if (chByte == 113) {
    led_state_2 = 1;
  }
  if (chByte == 114) {
    led_state_2 = 0;
  }
  if (chByte == 115) {
    led_state_3 = 1;
  }
  if (chByte == 116) {
    led_state_3 = 0;
  }
  if (chByte == 117) {
    led_state_4 = 1;
  }
  if (chByte == 118) {
    led_state_4 = 0;
  }
  if (chByte == 119) {
    led_state_5 = 1;
  }
  if (chByte == 120) {
    led_state_5 = 0;
  }
  if (chByte == 121) {
    led_state_1 = 1;
    led_state_2 = 1;
    led_state_3 = 1;
    led_state_4 = 1;
    led_state_5 = 1;
  }
  if (chByte == 122) {
    led_state_1 = 0;
    led_state_2 = 0;
    led_state_3 = 0;
    led_state_4 = 0;
    led_state_5 = 0;
  }
  if (Serial.available()) {
    recChar = Serial.read();
    bluetoothSerial.print(recChar);
  }
}
```

<그림3> 아두이노 실제 소스

아두이노는 실제 가로등을 구현하는데 필요했다. 각 가로등의 번호를 매긴 후 on과 off 상태를 설정하여 프로그래밍을 하였다. 사람이 지나가거나 적정거리에서 움직인다면 반응하여 on 상태로 켜지게 된다. 또한, 전체 on기능과 전체 off기능을 추가하여 한번에 컨트롤 할수있도록 하여 번거로움을 줄였다. 아두이노를 브레드보드에 직접 선을 연결하여 표현을 하였다.

3.2.2 앱 어플리케이션



<그림4> 앱 어플리케이션 레이아웃

앱은 최대한 간단한 방식으로 만들었다. 실제 사용을 하는 관리자가 쉽게 조작할수있도록하였고 앱에서 작동할 때마다 데이터베이스에 저장될 수 있도록 하였다

```
if (v == bt1) { // 세번째 엘리디 버튼을 누르면
  if (state_1 == 0) { // 엘리디가 꺼져있다면
    state_1 = 1; // 변수값 1로 바꿔주고
    bt1.setBackgroundResource(R.drawable.rounded_corner_g);
    bt1.setText("3\nON");
    send_data(115); // 115전송
  } else { // 엘리디 켜져있다면
    state_1 = 0; // 변수값 0으로 바꿔주고
    bt1.setBackgroundResource(R.drawable.rounded_corner_r);
    bt1.setText("3\nOFF");
    send_data(116); // 116전송
  }
}
```

<그림5> 개별컨트롤 소스

가로등에 각자 번호를 지정하여 자신만의 고유번호를 갖게 한다. 그리고 켜졌을 때 상태를 1, 꺼진 상태를 0으로 지정을 하여 함수를 구현하였고 아두이노에 지정되어있던 번호를 안드로이드와 매칭시켜서 알고리즘을 구현하였다. 그리고 별도로 전체 켜는 것과 끄는 알고리즘을 구현하여 모든 기능을 구현하였다. 또한, 빛의 세기를 직접 컨트롤 할 수 있도록 하였고 그런 모든 정보를 데이터베이스에 저장할 수 있도록 하였다.

#### 4. 결론 및 향후 연구과제

현재 국내에는 센서 감지 형식의 원격 가로등 시스템이 존재하지 않는다. 이 시스템을 사용하면 현재 낭비되고 있는 국가의 전력을 상당부분 절약할 수 있고 중앙 집중형 통제 시스템에 다양하고 실용적인 기능들을 추가하여 효율적으로 사용할 수 있을 것이다.

이 시스템에는 아직 부족한 점이 보이는데 실제 시설물의 도입에 대한 부분이다. 우리가 고안한 시스템은 시스템의 기능적인 면을 강조하여 실물크기의 가로등과 도로를 축소시킨 데모 모형이다. 실생활에서 이용할 수 있으려면 도로기전설비 설치 및 관리에 관한 규정과 같은 법규를 고려하고 데모 시스템보다 훨씬 큰 규모의 가로등을 제어하기 위한 서버의 구축 방법을 더 고민해 보아야 한다.

이번 과제를 수행하면서 안드로이드 및 아두이노 그리고 서버에 필요한 전반적인 문법과 구조를 배울 수 있는 시간이었다. 평소의 프로그램이 일정한 포맷 위에서 실행되지만 하는 구조였다면 이번의 과제는 직접 사물과의 통신으로 제어를 하고 우리 실생활과 밀접한 시스템을 구축하는 과정이어서 실제적이고 흥미로웠다. 현실에 직접 적용한다는 것은 더 많은 제약조건과 환경을 고려해야 한다는 것도 깨달았다. 다음 개발 프로젝트에서는 조금 더 많은 것들을 고려하여 더 발전된 모습으로 도전 할 수 있는 안목이 생긴 것 같다.

#### 참고문헌

[1] 지그비 네트워크를 이용한 통신비 절감형 가로등 감시 제어 시스템 및 방법  
(Street lamp control system and method for saving the communication cost by using the zigbee network)

“<http://patent.ndsl.kr/patDetail.do?cn=KOR1020070058391>”

[2] 지그비 통신망을 이용한 보안등 관제 시스템에 관한 연구  
“<http://www.jkosme.or.kr/xml/02021/02021.pdf>”

[3] 사물 인터넷을 품은 아두이노 허경용 저자

[4] 아두이노 상상을 스케치하다 허경용 저자