

## 데이터 입력/출력



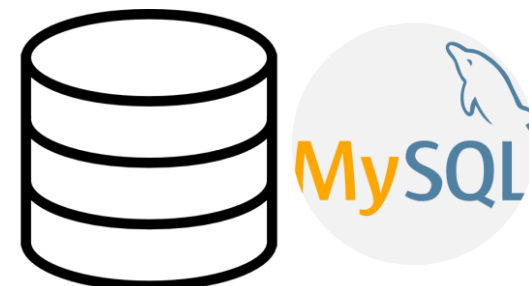
## 라즈베리파이

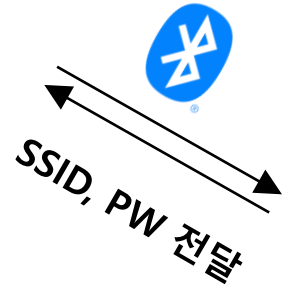
### 제어 컨트롤러



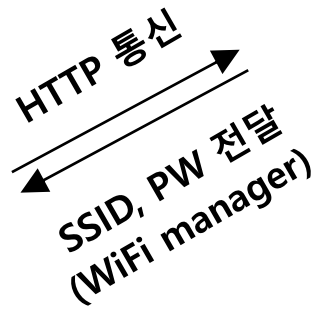
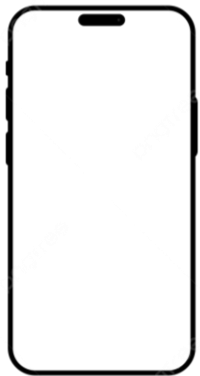
### 데이터저장

데이터저장





SSID, PW 전달



HTTP 통신  
SSID, PW 전달  
(WiFi manager)

BLE server



Web server



메세지 전달



```
{  
  "ssid": "hyu-iot",  
  "password": "12345678",  
  "mqtt_broker": "ai.doowon.ac.kr",  
  "mqtt_topic_in": "/iot/bombzoo/in",  
  "mqtt_topic_out": "/iot/bombzoo/out"  
}
```

## Webserver/HTTP 통신을 이용한 WiFi/mqtt정보 전송

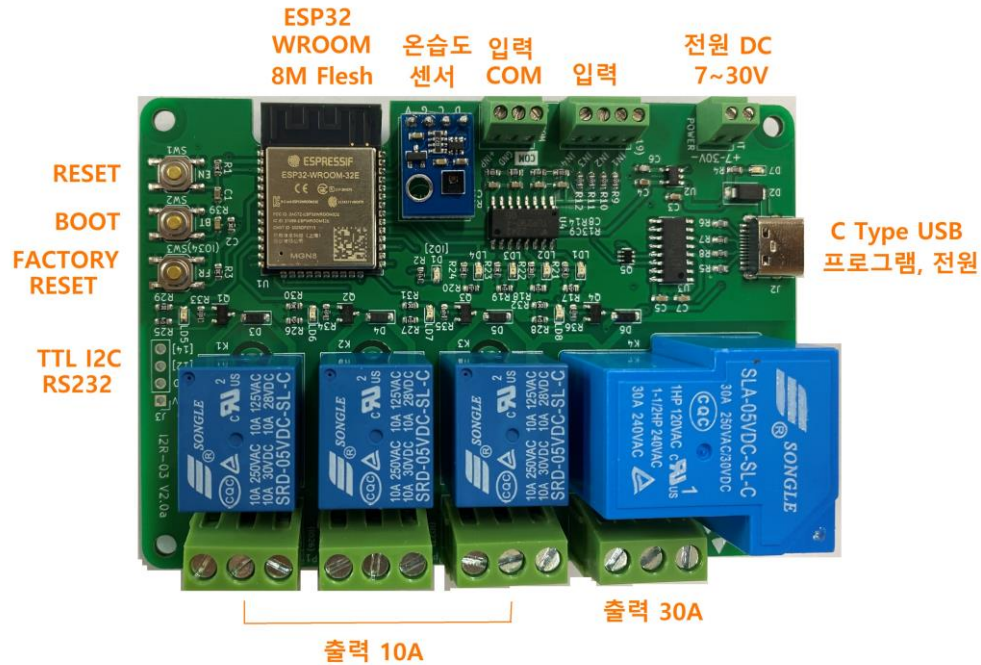
- 1) esp8266보드에서 WiFimanager를 이용하여 wifi에 연결되는 아두이노 프로그램을 작성해줘
- 2) 위의 프로그램을 기반으로 esp8266보드에서 WiFimanager를 이용하여 wifi에 연결한후 Webserver통해 접속해 있는 SSID와 IP address를 표시해 주는 아두이노 프로그램을 작성해줘
- 3) 위의 프로그램을 기반으로 esp8266보드에서 WiFimanager를 이용하여 wifi에 연결한후 Webserver통해 mqtt 정보(mqtt\_broker, mqtt\_broker\_port, mqtt\_topic\_out, mqtt\_topic\_in)를 받아와서 시리얼 프린트해주는 코드를 작성해줘.
- 4) 위의 프로그램을 기반으로 esp8266보드에서 WiFimanager를 이용하여 wifi에 연결한후 Webserver통해 mqtt 정보(mqtt\_broker, mqtt\_broker\_port, mqtt\_topic\_out, mqtt\_topic\_in)를 받아와서 양방향 mqtt통신을 연결하고 mqtt\_topic\_in을 구독하고 mqtt\_topic\_out으로 3초마다 한번씩 "mqtt count x"를 발행하는 코드를 작성해줘.

- 1) esp32보드에서 스마트 폰과 양방향 BLE통신을 하는 아두이노 프로그램을 작성해줘
- 2) 위의 프로그램을 기반으로 esp32보드에서 BLE통신을 통해 스마트폰에서 SSID와 password를 입력받고 wifi를 연결하는 아두이노 프로그램을 작성해줘. 스마트폰으로 부터 입력되는 데이터는 json형식으로 {"ssid":"","password":""}와 같다
- 3) 위의 프로그램을 기반으로 esp32보드에서 BLE통신을 통해 스마트폰에서 wifi와 mqtt정보를 입력받아 wifi를 연결하는하고 양방향 mqtt통신을 하는 코드를 작성해줘. 스마트폰으로 부터 입력되는 데이터는 json형식으로 {"ssid":"","password":"","mqtt\_broker":"","mqtt\_topic\_in":"","mqtt\_topic\_out":""}와 같다





## IoT Board



전체 코드 (default)

<https://github.com/kdi6033/i2r-03/blob/main/0%20Source-Program-IoT/board-i2r-03/board-i2r-03.ino>

## 온습도 센서



## ESP32



ESP32는 Wi-Fi와 Bluetooth 기능을 통합한 마이크로컨트롤러(MCU)이다.

[https://github.com/BumjuAhn/2024\\_IOT\\_lecture/blob/main/lecture\\_09\\_week/esp8266\\_wifimanager/esp8266\\_wifimanager.ino](https://github.com/BumjuAhn/2024_IOT_lecture/blob/main/lecture_09_week/esp8266_wifimanager/esp8266_wifimanager.ino)

## 2) Wifi Manager를 이용한 SSID, PW 전송

```
#include <ESP8266WiFi.h>
#include <DNSServer.h>
#include <ESP8266WebServer.h>
#include <WiFiManager.h>

int gpio0Reset = D2;

const int ledPin = LED_BUILTIN; // the number of the LED pin

WiFiServer server(80);

void setup() {
    Serial.begin(9600);
    pinMode(gpio0Reset, INPUT_PULLUP);
    pinMode(ledPin, OUTPUT);
    digitalWrite(ledPin, HIGH);

    WiFiManager wifiManager;

    int resetBtn = digitalRead(gpio0Reset);
    if(resetBtn == 0) {
        wifiManager.resetSettings();
        digitalWrite(ledPin, LOW);
    }

    wifiManager.autoConnect();

    Serial.println("Connected.");
    digitalWrite(ledPin, LOW);
    delay(2000);
    digitalWrite(ledPin, HIGH);

    server.begin();
}

void loop() {
    WiFiClient client = server.available();
    if (!client) {
        return;
    }

    String req = client.readStringUntil('\r');
    Serial.println(req);
    client.flush();

    // Prepare the response
    String s = "<html>";
    s += "<meta name='viewport' content='width=device-width, initial-scale=1.0' />";
    s += "<meta http-equiv='Content-Type' content='text/html; charset=utf-8' />";
    // s += "<meta http-equiv='refresh' content='5' />";
    s += "<meta http-equiv='Content-Type' content='text/html; charset=utf-8' />";
    s += "<head></head><body>SmartConfig 접속 안녕하세요!<br>";
    s += "연결한 IP주소 : <a href='http://"+WiFi.localIP().toString()+"' />"+WiFi.localIP().toString()+"</a>";
    s += "</body></html>";

    client.print(s);
    delay(1);
    Serial.println("Client disconnected");
}
```