‘Arduino IDE’로 쉽게 따라해보는 실전 ESP8266 프로그래밍

처음에 해야 할 일

H/W

S/W

본격적으로 시작하기

기초

- Serial Monitor에 ‘Hello World!’ 출력하기
- DHT11로 온습도 측정하기
- Web Server 개설하기

실전

- 온습도를 알려주는 홈페이지 만들기
- 홈페이지를 통해 릴레이 모듈 제어하기

진행 전 체크리스트

- HTTP, mDNS 같은 용어를 들어본 적 없다 ... go to p.21
- 서버/클라이언트 차이를 모른다 ... go to p.19
- 센서/릴레이 만져본 적 없다 ... go to p.17
- 시리얼 통신 해본 적 없다 ... go to p.15
- 전구 켜 본 적도 없다 ... go to p.9

들어가기에 앞서...

Built-in low-power 32-bit CPU:
can double as an application processor

2ms, connect and transfer data packets

Built-in temperature sensor

standby power consumption of less than 1.0mW (DTIM3)

802.11 b / g / n

Wi-Fi Direct (P2P), soft-AP

ESP8266은 무엇인가?

STBC, 1×1 MIMO, 2×1 MIMO

off leakage current is less than 10uA

Built-in PLL, voltage regulator
and power management components

Support antenna diversity

A-MPDU, A-MSDU aggregation and the 0.4 Within wake

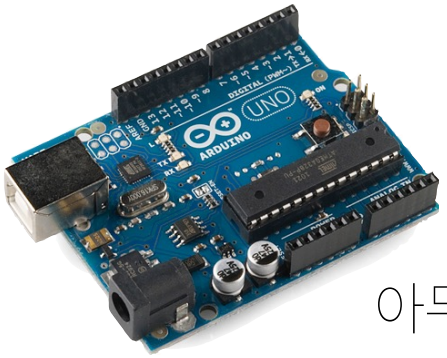
Built-in TR switch, balun, LNA,
power amplifier and matching network

Built-in TCP / IP protocol stack

SDIO 2.0, SPI, UART

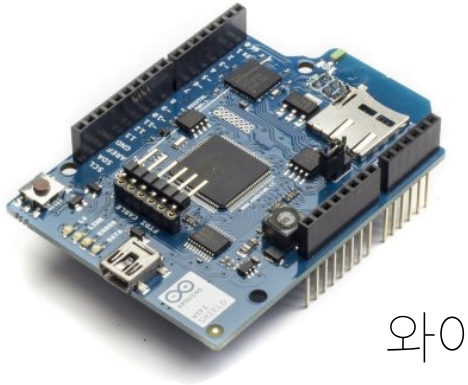
802.11b mode + 19.5dBm output power

아두이노로 사물인터넷을 하려면



아두이노 우노

정품보드 20,000 ~ 30,000₩
(호환보드 9,000 ~ 11,000₩)



와이파이 쉴드

정품 120,000 ~ 130,000₩
(호환쉴드* 50,000₩ ~)

-> 정품 보드 사용 시 160,000₩ + α 소요

* CC3000 WiFi Shield 기준

ESP-12E

Tensilica Xtensa LX3 80Mhz
32-Bit Processor & 4MB Flash

WiFi Controller와 GPIO, UART,
SPI 등 다양한 통신 지원

펌웨어 개발을 위한 Cross Compiler
Tool이 공개되어 있을 뿐 만 아니라

다른 MCU 도움없이 다양한 작업을
수행 할 수 있음에도 $3 미만

⋮

NodeMCU

Arduino IDE

Espruino

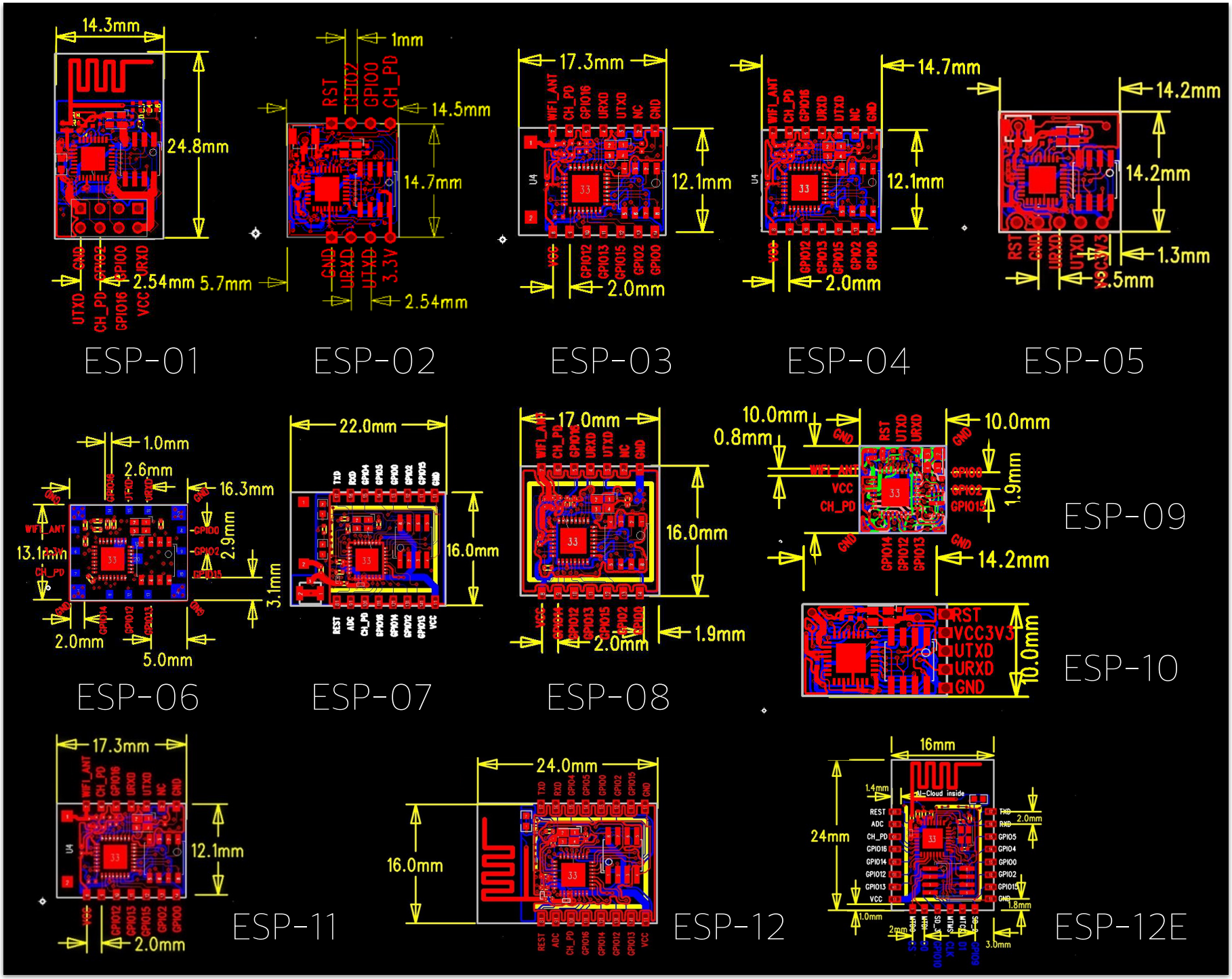
⋮

회로 배선

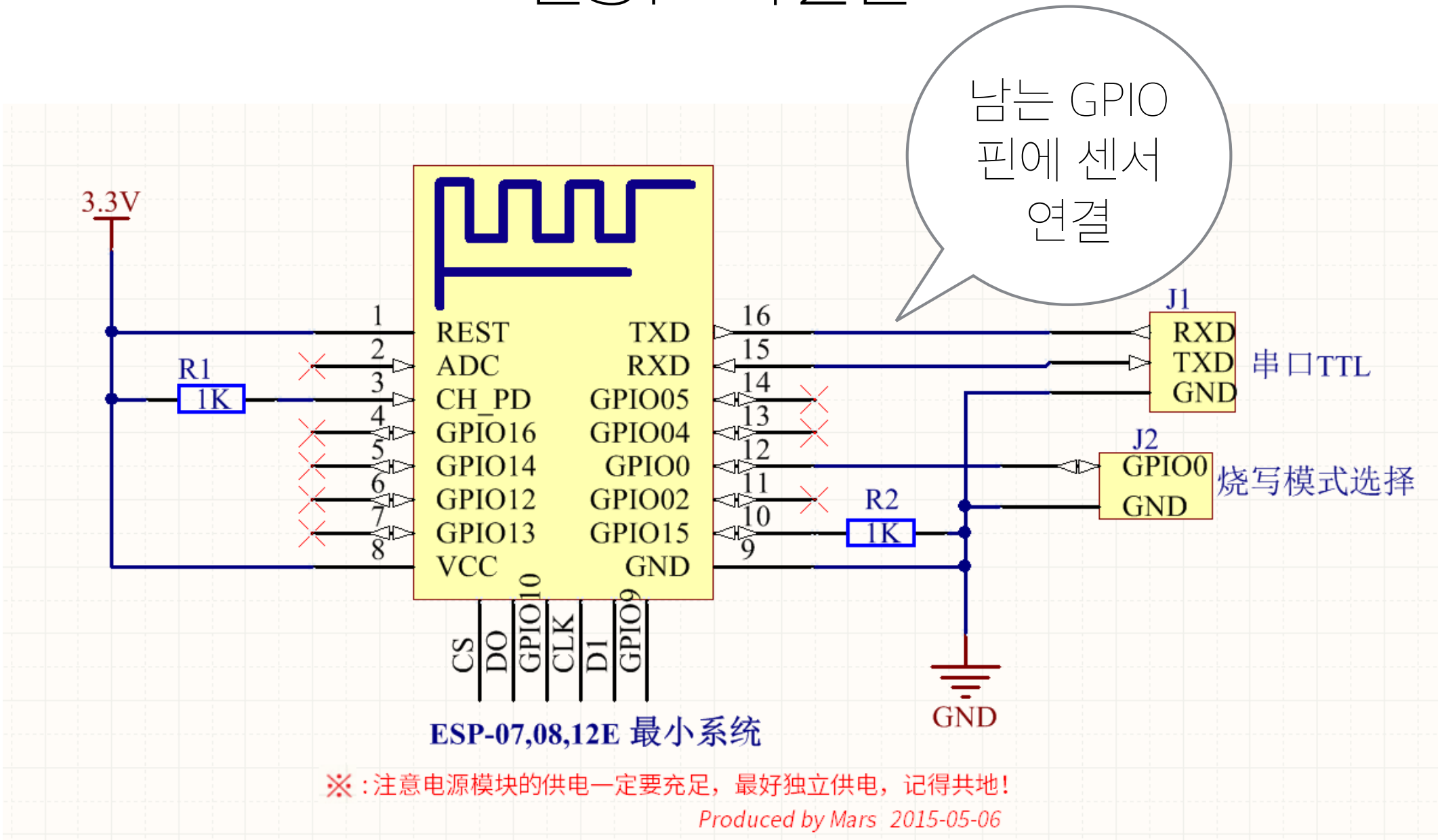
에 앞서...

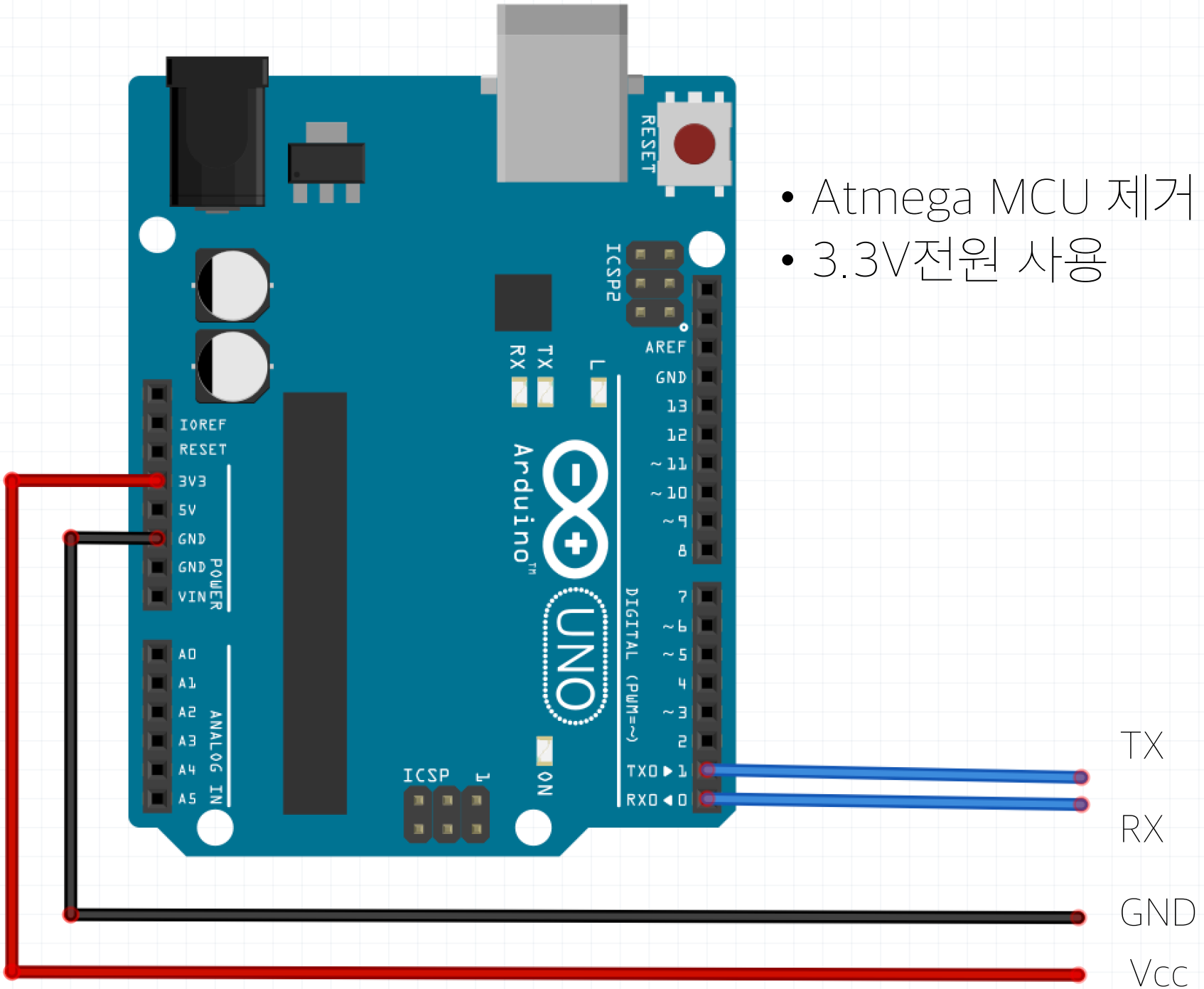
ESP8266칩을 사용하는
다양한 종류의 칩셋이 나와있다

여기선 ESP-12E 위주로
설명하게 될 것


14.3mm
24.8mm
2.54mm
5.7mm
GND
GPIO2
GPIO0
URXD
UTXD
CH_PD
GPIO16
VCC
ESP-01
1mm
RST
GPIO2
GPIO0
CH_PD
14.5mm
14.7mm
GND
URXD
UTXD
3.3V
2.54mm
ESP-02
17.3mm
WIFI_ANT
CH_PD
GPIO16
URXD
UTXD
NC
GND
12.1mm
GPIO12
GPIO13
GPIO15
GPIO2
GPIO0
2.0mm
ESP-03
14.7mm
12.1mm
2.0mm
ESP-04
14.2mm
14.2mm
1.3mm
2.5mm
ESP-05
1.0mm
2.6mm
16.3mm
13.1mm
2.9mm
3.1mm
2.0mm
5.0mm
ESP-06
22.0mm
16.0mm
ESP-07
17.0mm
16.0mm
1.9mm
2.0mm
ESP-08
10.0mm
0.8mm
10.0mm
1.9mm
14.2mm
ESP-09
RST
VCC3V3
UTXD
URXD
GND
10.0mm
ESP-10
17.3mm
12.1mm
2.0mm
ESP-11
24.0mm
16.0mm
ESP-12
16mm
24mm
1.4mm
2.0mm
1.8mm
1.0mm
2mm
3.0mm
ESP-12E

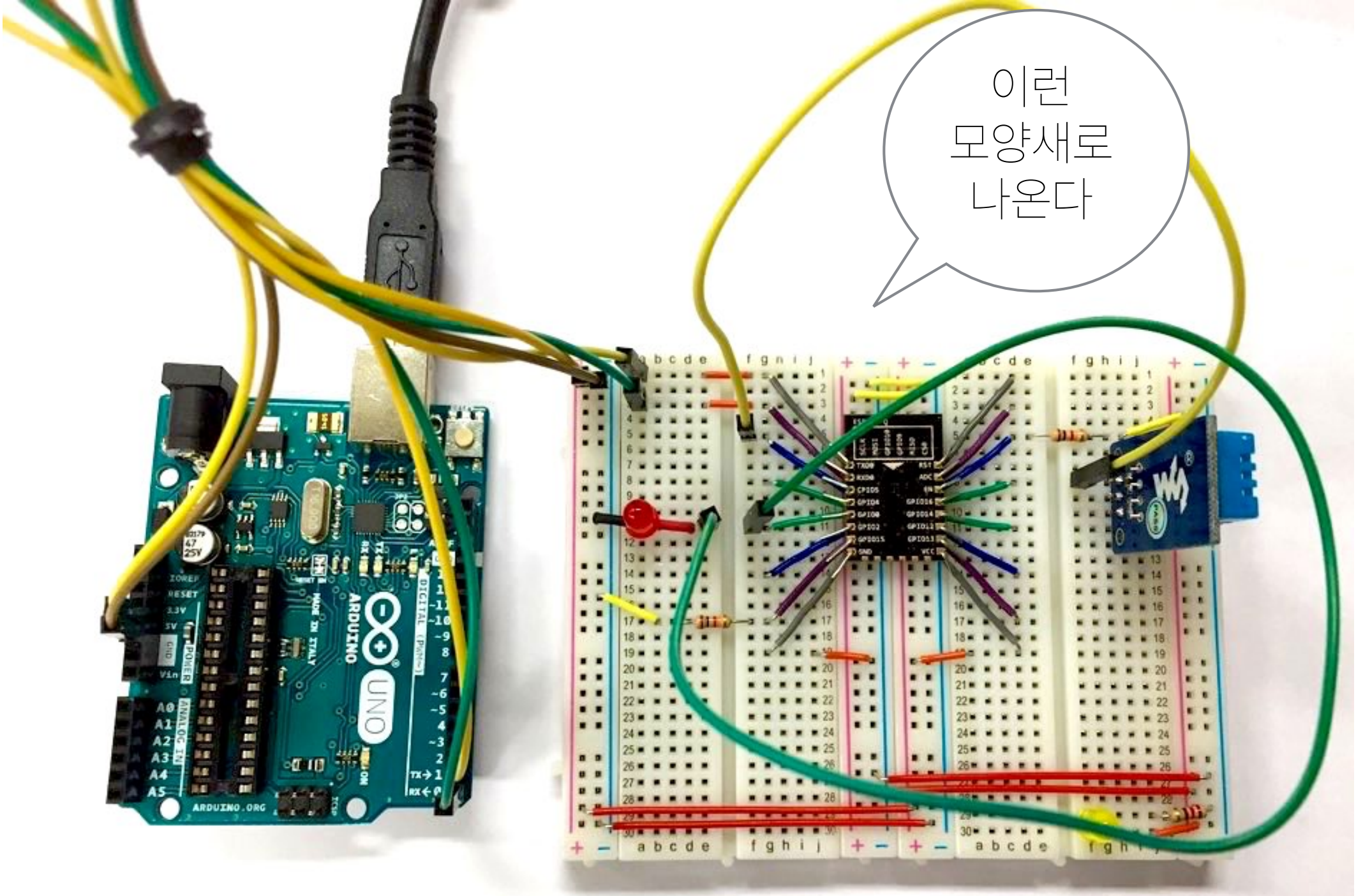
ESP-12E


남는 GPIO 핀에 센서 연결
3.3V
R1
1K
REST
ADC
CH_PD
GPIO16
GPIO14
GPIO12
GPIO13
VCC
TXD
RXD
GPIO05
GPIO04
GPIO0
GPIO02
GPIO15
GND
CS
DO
GPIO10
CLK
D1
GPIO9
J1
RXD
TXD
GND
串口TTL
J2
GPIO0
GND
烧写模式选择
R2
1K
GND
ESP-07,08,12E 最小系统
※：注意电源模块的供电一定要充足，最好独立供电，记得共地！
Produced by Mars 2015-05-06

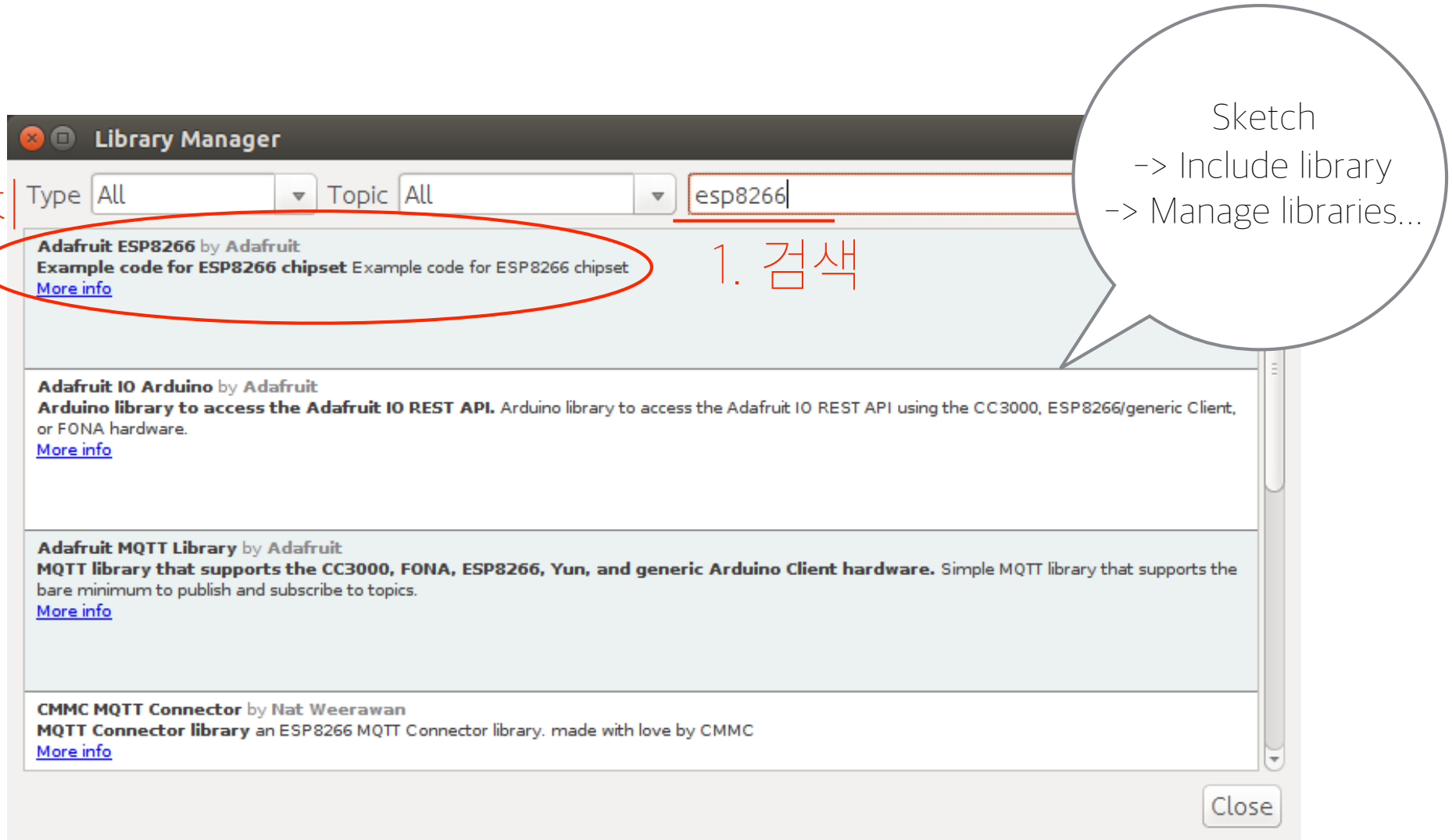


- Atmega MCU 제거
- 3.3V전원 사용

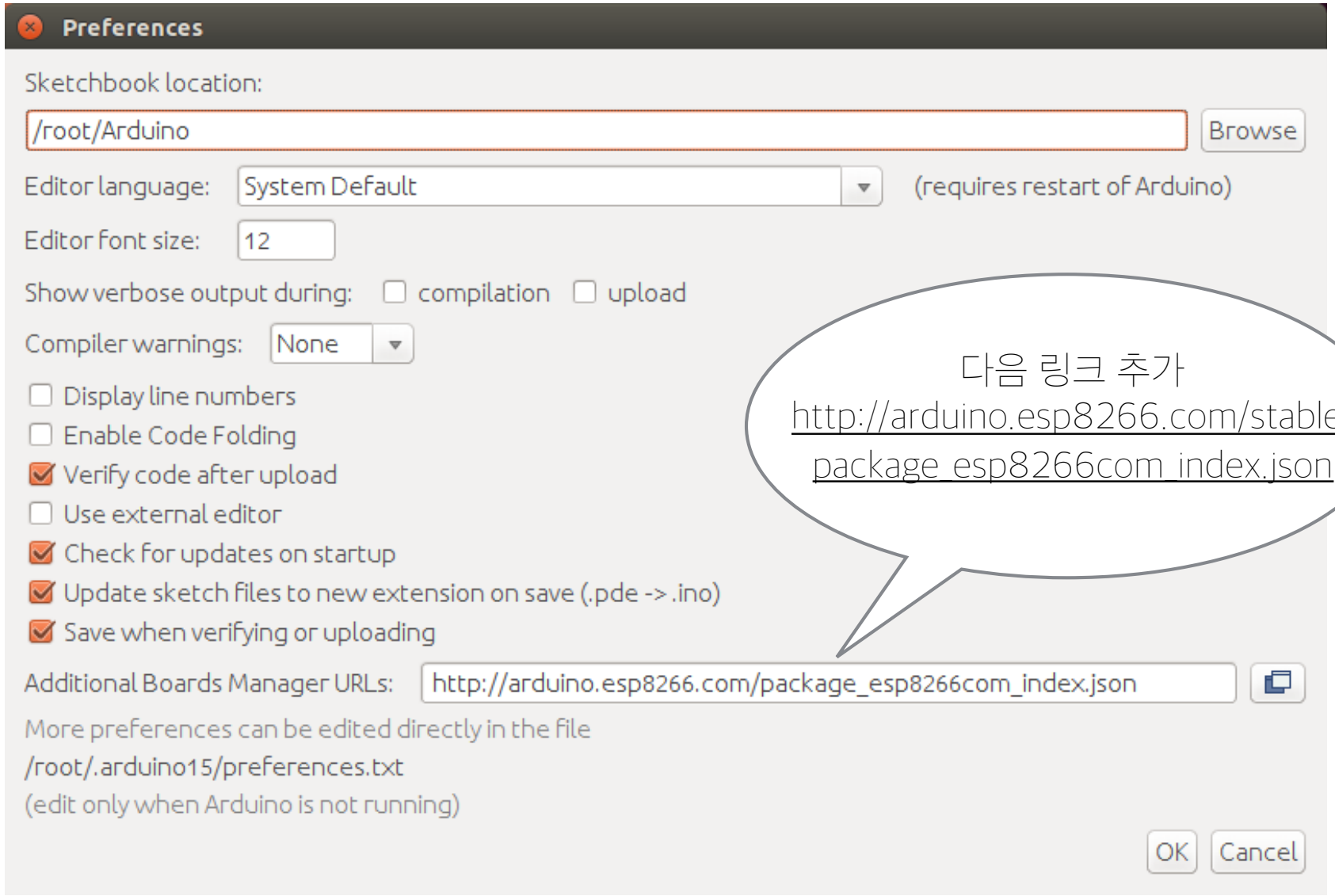
fritzing


이런
모양새로
나온다

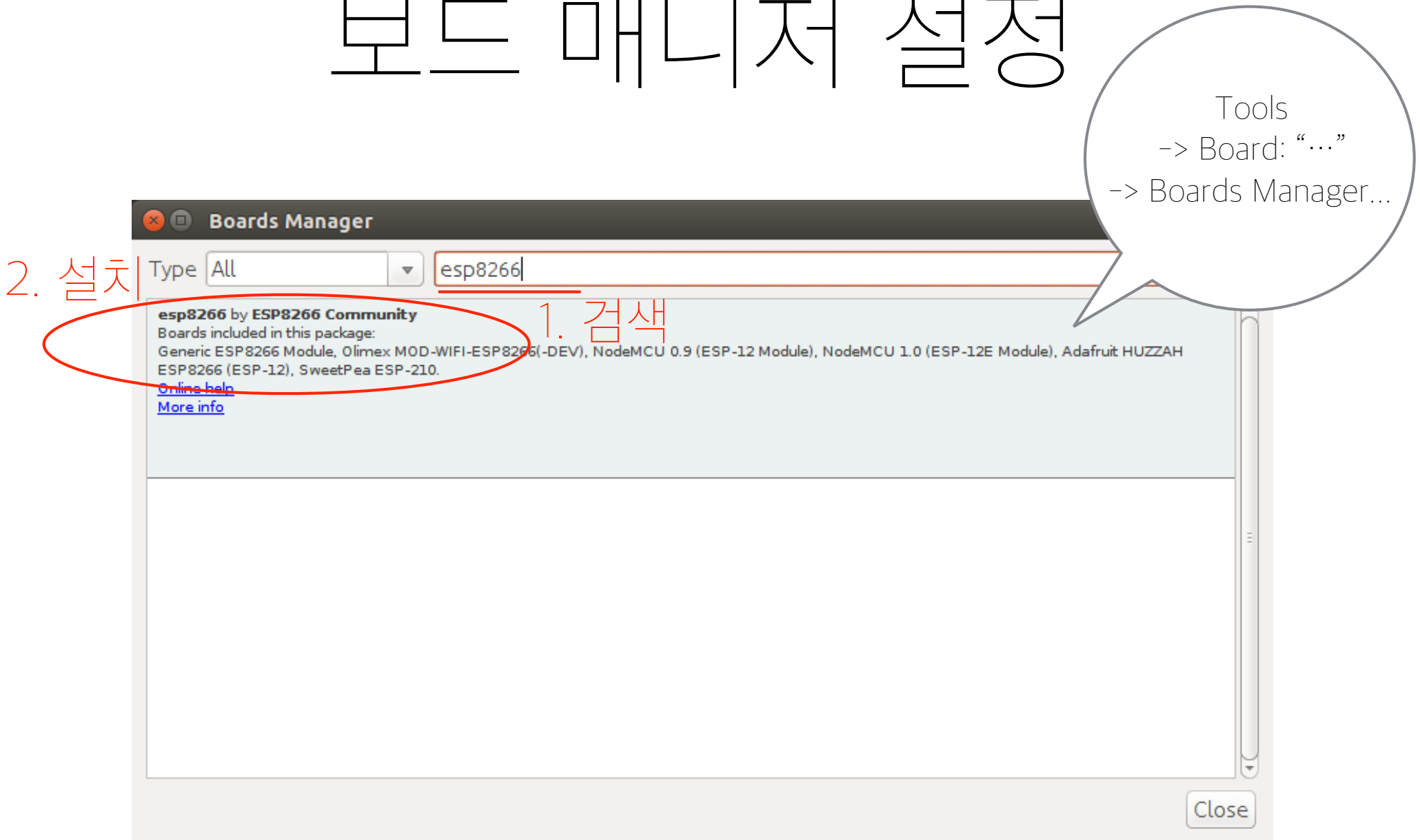
라이브러리 매니저 설치


Sketch
-> Include library
-> Manage libraries...
Library Manager
Type All
Topic All
esp8266
2. 설치
1. 검색
Adafruit ESP8266 by Adafruit
Example code for ESP8266 chipset Example code for ESP8266 chipset
More info
Adafruit IO Arduino by Adafruit
Arduino library to access the Adafruit IO REST API. Arduino library to access the Adafruit IO REST API using the CC3000, ESP8266/generic Client, or FONA hardware.
More info
Adafruit MQTT Library by Adafruit
MQTT library that supports the CC3000, FONA, ESP8266, Yun, and generic Arduino Client hardware. Simple MQTT library that supports the bare minimum to publish and subscribe to topics.
More info
CMMC MQTT Connector by Nat Weerawan
MQTT Connector library an ESP8266 MQTT Connector library. made with love by CMMC
More info
Close

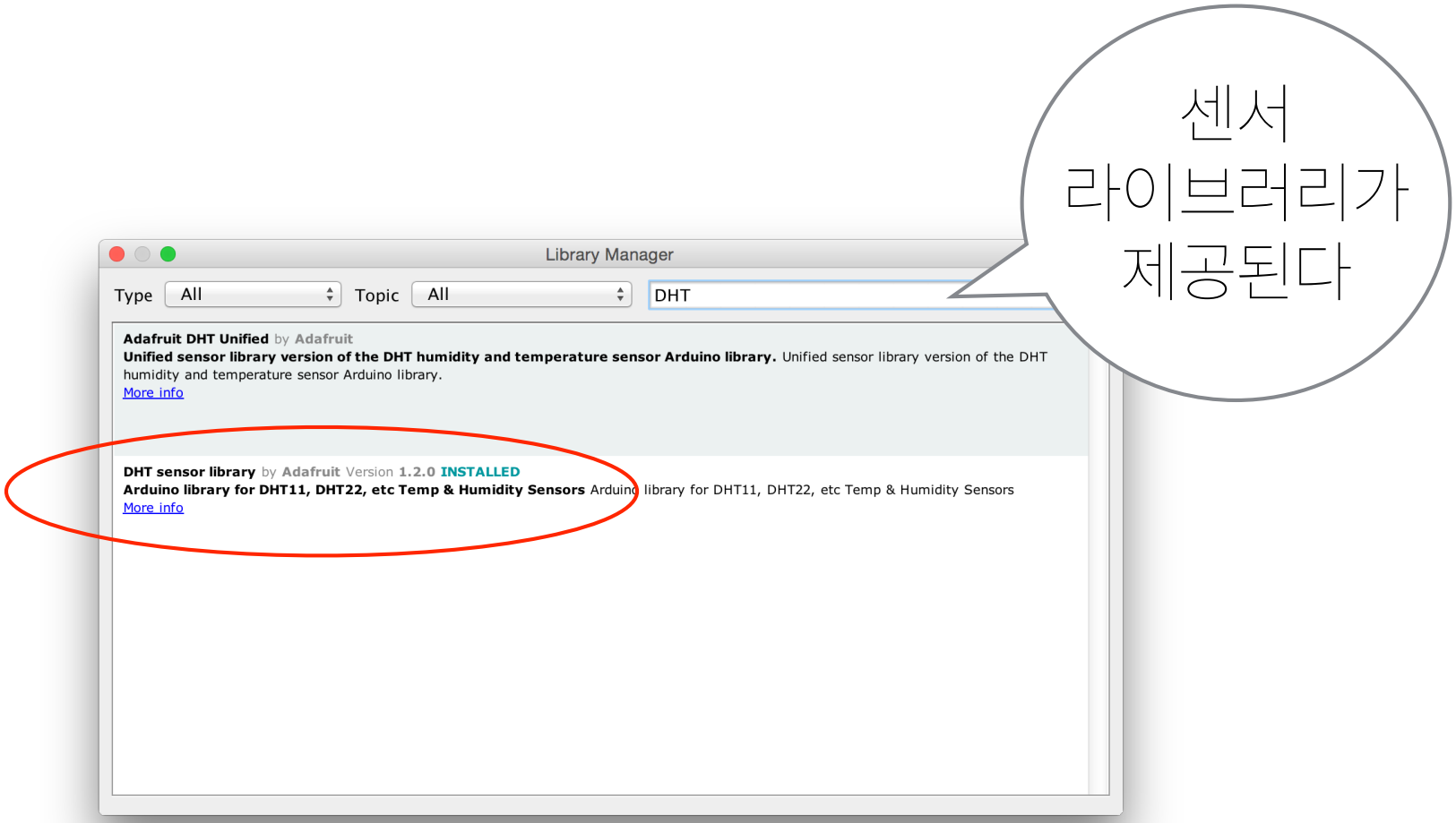
보드 매니저 URL 추가


Preferences
Sketchbook location:
/root/Arduino
Browse
Editor language:
System Default
(requires restart of Arduino)
Editor font size:
12
Show verbose output during:
compilation
upload
Compiler warnings:
None
Display line numbers
Enable Code Folding
Verify code after upload
Use external editor
Check for updates on startup
Update sketch files to new extension on save (.pde -> .ino)
Save when verifying or uploading
Additional Boards Manager URLs:
http://arduino.esp8266.com/package_esp8266com_index.json
More preferences can be edited directly in the file
/root/.arduino15/preferences.txt
(edit only when Arduino is not running)
OK
Cancel
다음 링크 추가
http://arduino.esp8266.com/stable/package_esp8266com_index.json

보드 매니저 설정


Tools
-> Board: "…"
-> Boards Manager...
Boards Manager
Type
All
esp8266
1. 검색
2. 설치
esp8266 by ESP8266 Community
Boards included in this package:
Generic ESP8266 Module, Olimex MOD-WIFI-ESP8266(-DEV), NodeMCU 0.9 (ESP-12 Module), NodeMCU 1.0 (ESP-12E Module), Adafruit HUZZAH ESP8266 (ESP-12), SweetPea ESP-210.
Online help
More info
Close

DHT11로 온습도 측정하기



File Edit Sketch Tools Help

DHTtester

```
// Example testing sketch for various DHT humidity/temperature sensors
// Written by ladyada, public domain

#include "DHT.h"

#define DHTPIN 2     // what digital pin we're connected to

// Uncomment whatever type you're using!
//#define DHTTYPE DHT11   // DHT 11
#define DHTTYPE DHT22   // DHT 22  (AM2302), AM2321
//#define DHTTYPE DHT21   // DHT 21 (AM2301)

// Connect pin 1 (on the left) of the sensor to +5V
// NOTE: If using a board with 3.3V logic like an Arduino Due connect pin 1
// to 3.3V instead of 5V!
// Connect pin 2 of the sensor to whatever your DHTPIN is
// Connect pin 4 (on the right) of the sensor to GROUND
// Connect a 10K resistor from pin 2 (data) to pin 1 (power) of the sensor

// Initialize DHT sensor.
// Note that older versions of this library took an optional third parameter to
// tweak the timings for faster processors.  This parameter is no longer needed
// as the current DHT reading algorithm adjusts itself to work on faster procs.
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println("DHTxx test!");

  dht.begin();
}

void loop() {
  // Wait a few seconds between measurements.
  delay(2000);

  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);
```

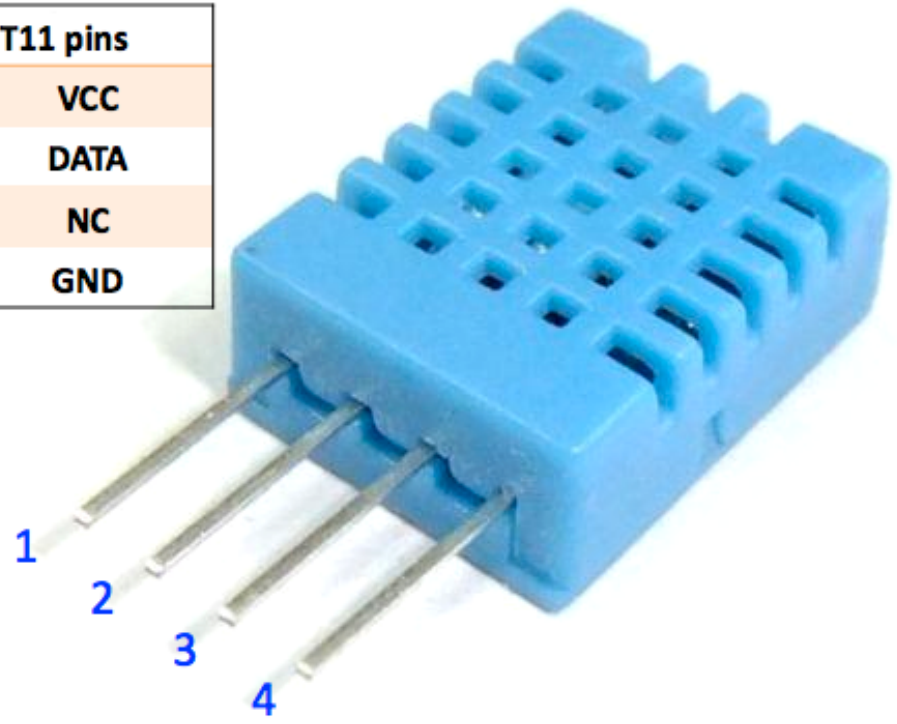
사용할 GPIO 핀으로 변경 혹은 추가

사용할 센서의 주석을 제거

```
// tweak the timings for faster processors.  This parameter is no longer needed
// as the current DHT reading algorithm adjusts itself to work on faster procs.
DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  Serial.println("DHTxx test!");

  dht.begin();
}

void loop() {
  // Wait a few seconds between measurements.
  delay(2000);

  // Reading temperature or humidity takes about 250 milliseconds!
  // Sensor readings may also be up to 2 seconds 'old' (its a very slow sensor)
  float h = dht.readHumidity();
  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();
  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);

  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println("Failed to read from DHT sensor!");
    return;
  }

  // Compute heat index in Fahrenheit (the default)
  float hif = dht.computeHeatIndex(f, h);
  // Compute heat index in Celsius (isFahreheit = false)
  float hic = dht.computeHeatIndex(t, h, false);

  Serial.print("Humidity: ");
  Serial.print(h);
  Serial.print(" %\t");
  Serial.print("Temperature: ");
  Serial.print(t);
  Serial.print(" *C ");
  Serial.print(f);
  Serial.print(" *F\t");
  Serial.print("Heat index: ");
  Serial.print(hic);
  Serial.print(" *C ");
  Serial.print(hif);
  Serial.println(" *F");
}
```

DHT11로 온습도 측정하기

DHT11 pins	
1	VCC
2	DATA
3	NC
4	GND



http://www.dunkelheit.com.br/arduino/climaduino/Datasheet/DHT11/DHT11_Pins.png