

(응용)체온 측정 출입관리



COVID-19 체온 측정 관리 제품



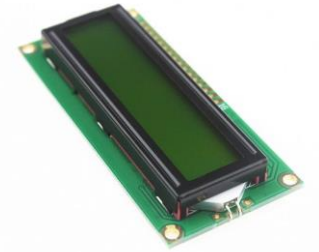
전체 구성

손(물체) 유무 감지

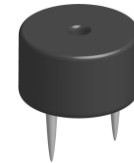


온도 측정

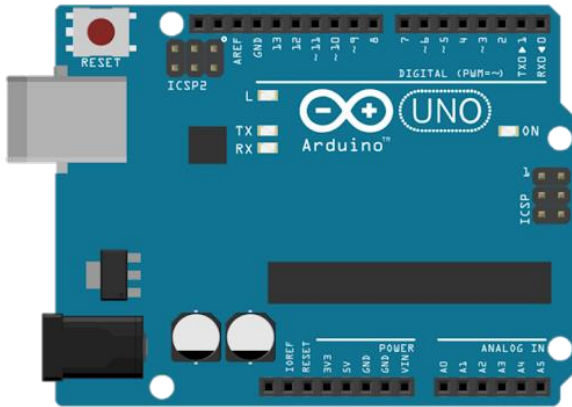
정보 출력(시리얼통신 or LCD)



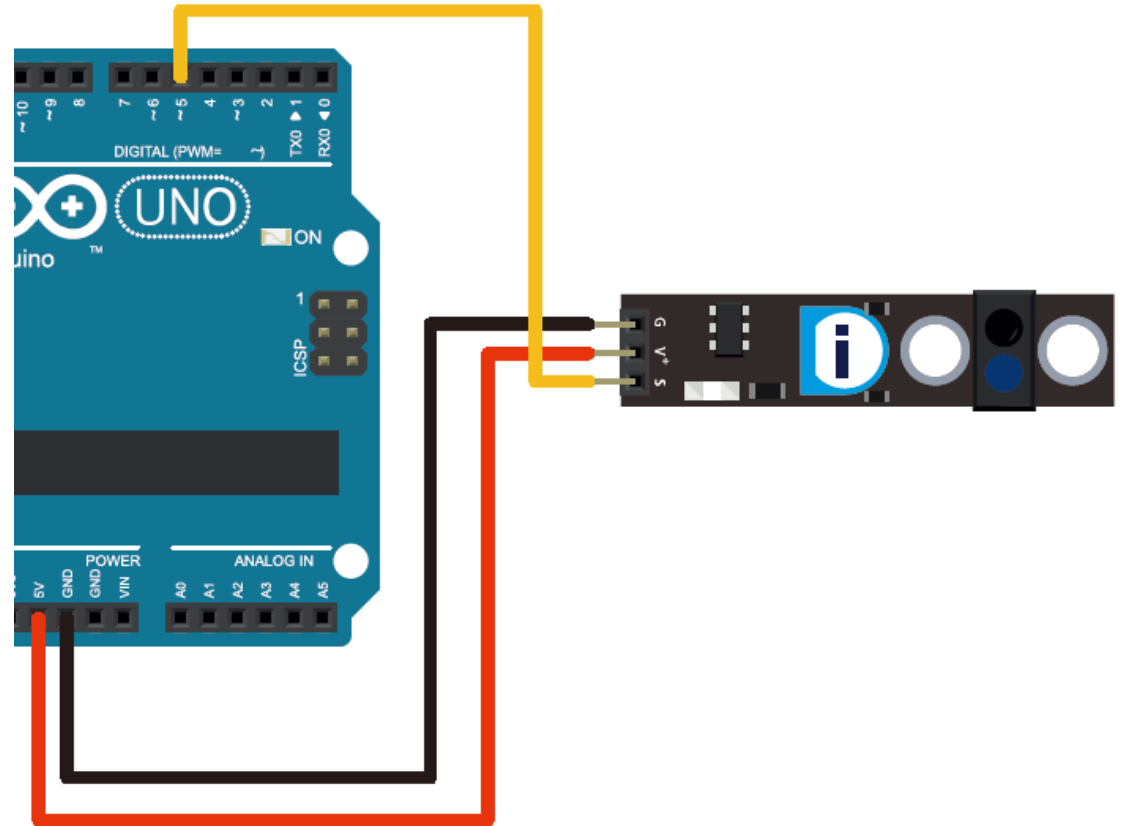
측정 완료 경고음 발생



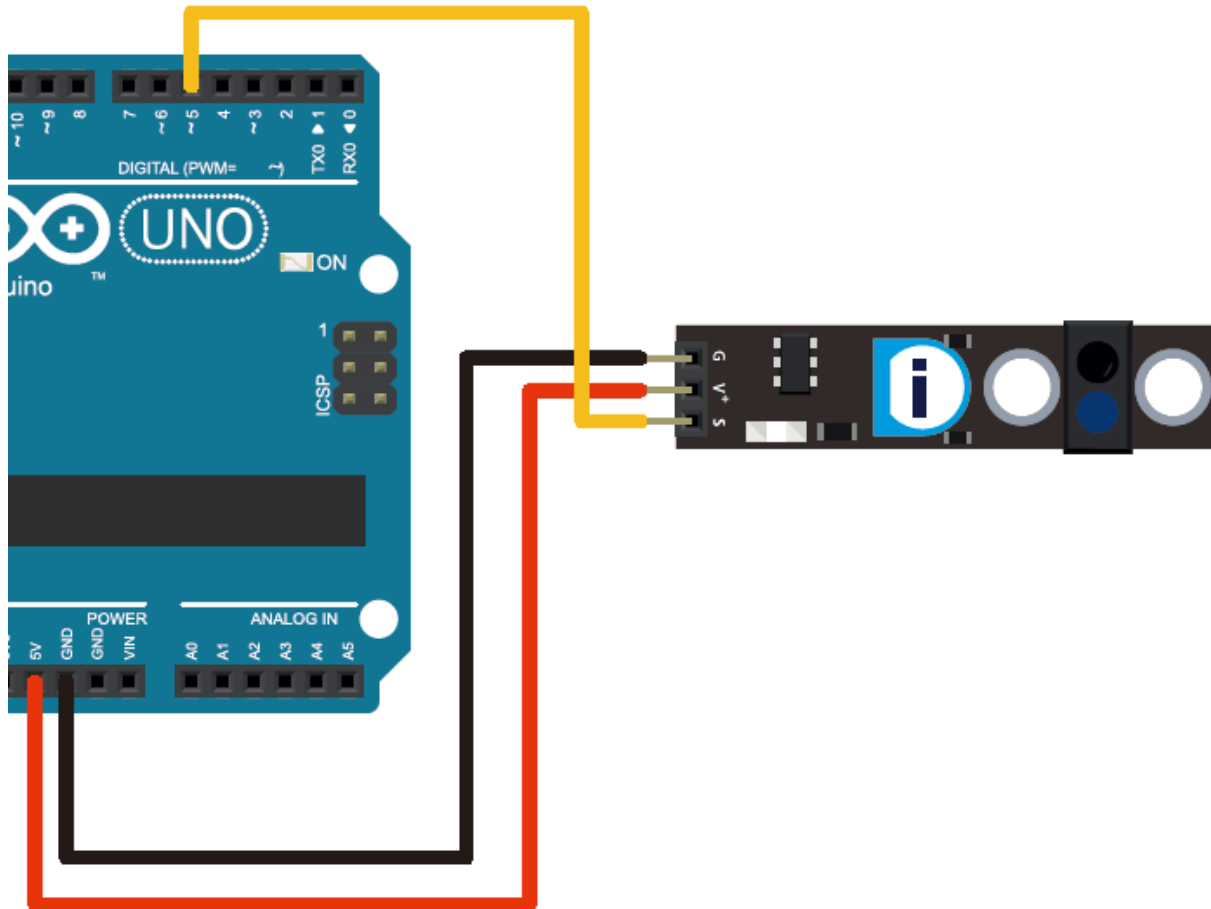
LED 인디케이터



적외선 감지 센서 실험



적외선 라인감지 센서 실험



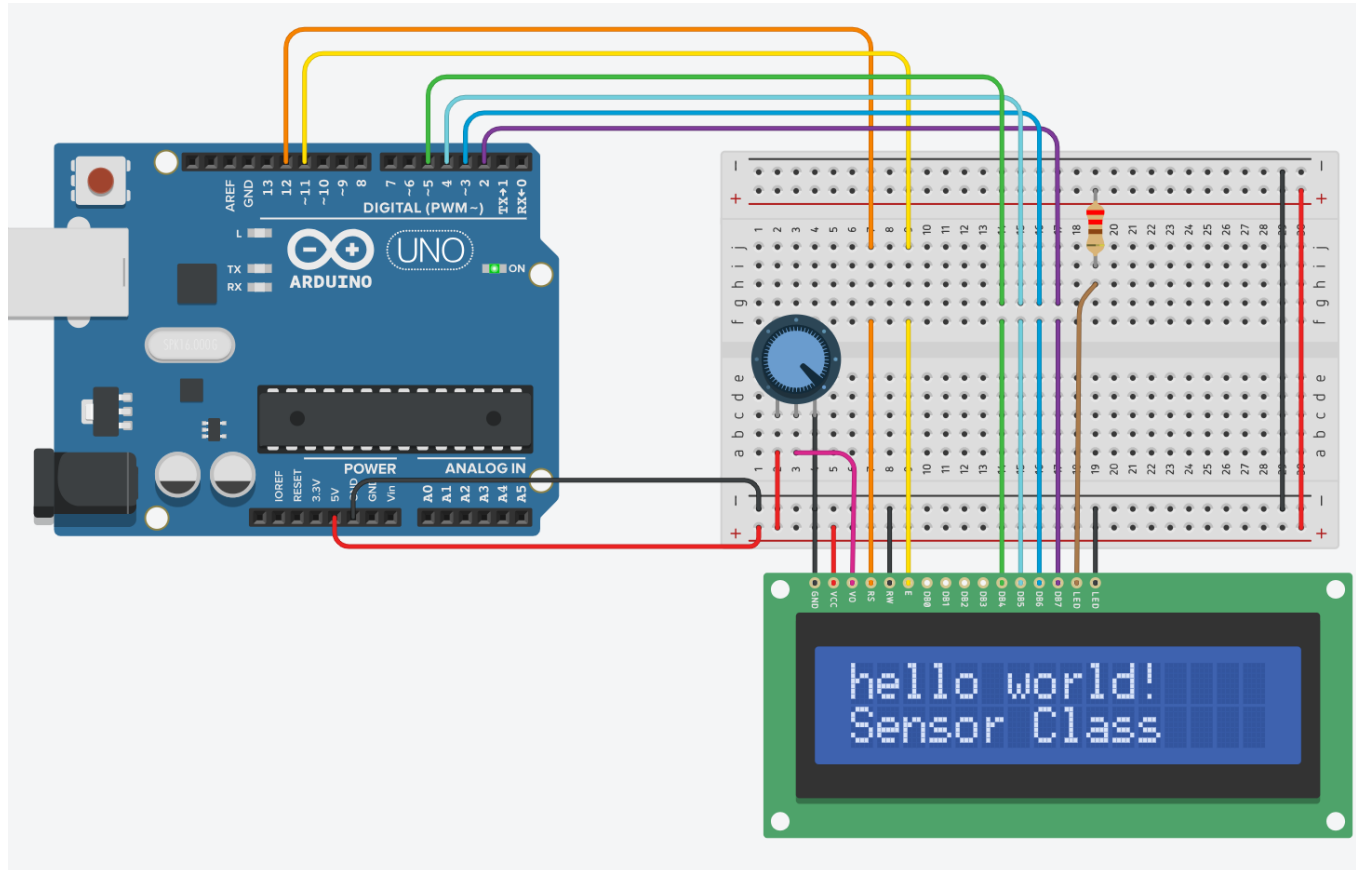
```
void setup()
{
  Serial.begin(9600) ;
  pinMode(5, INPUT) ;
}

void loop()
{
  if( digitalRead(5) == HIGH )
  {
    Serial.println("1") ;
  }
  else
  {
    Serial.println("2") ;
  }

  delay(500) ;
}
```

16x2 Character LCD 실험

- 아두이노 Example을 이용



```
#include <LiquidCrystal.h>
```

```
LiquidCrystal lcd_1(12, 11, 5, 4, 3, 2);
```

```
void setup()
```

```
{  
  lcd_1.begin(16, 2);  
}
```

```
void loop()
```

```
{  
  lcd_1.setCursor(0, 0);  
  lcd_1.print("hello world!");  
  
  lcd_1.setCursor(0, 1);  
  lcd_1.print("Sensor Class");  
  delay(1000);  
}
```

부저 실험

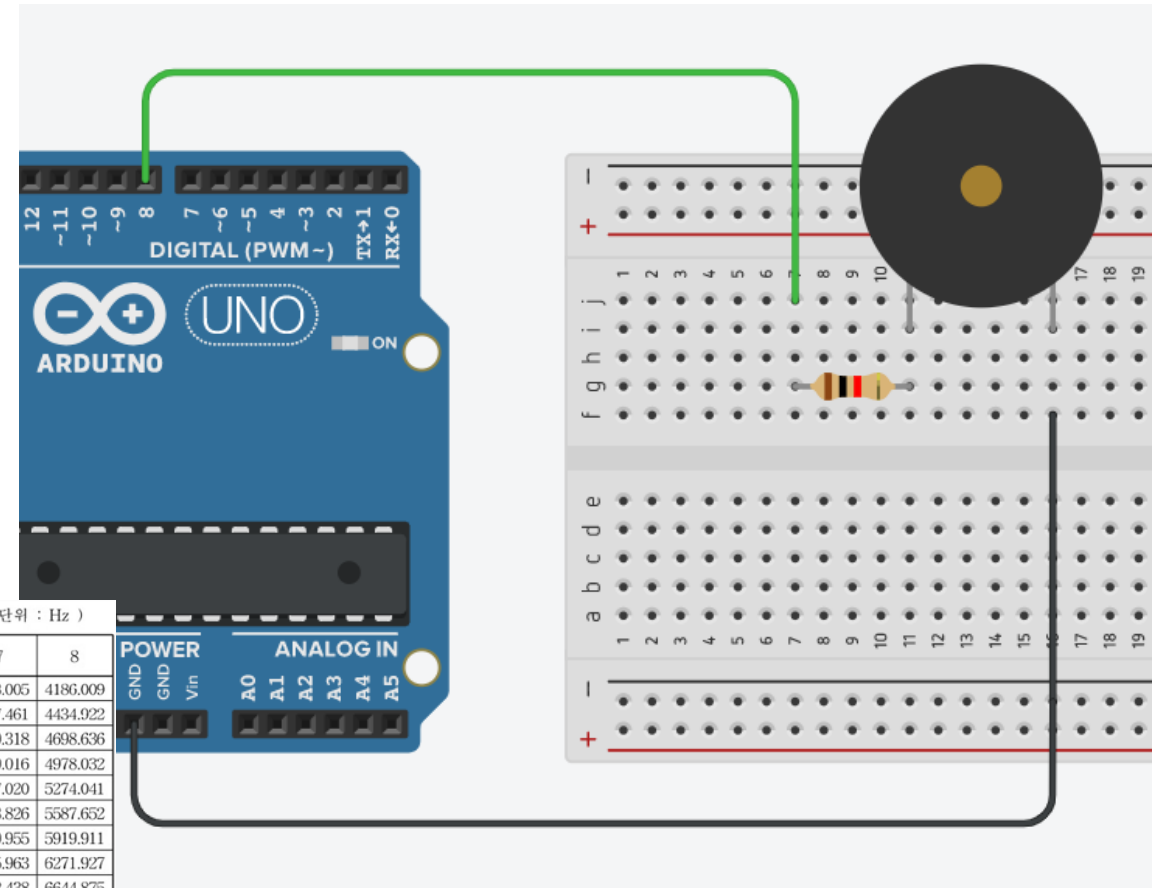
- 부저(소리) 출력 실험

```
void setup()
{
  pinMode(8, OUTPUT);
}

void loop()
{
  tone(8, 262, 500);
  delay(500);
}
```

(단위 : Hz)

음계 \ 옥타브	1	2	3	4	5	6	7	8
C(도)	32.7032	65.4064	130.8128	261.6256	523.2511	1046.502	2093.005	4186.009
C#	34.6478	69.2957	138.5913	277.1826	554.3653	1108.731	2217.461	4434.922
D(레)	36.7081	73.4162	146.8324	293.6648	587.3295	1174.659	2349.318	4698.636
D#	38.8909	77.7817	155.5635	311.1270	622.2540	1244.508	2489.016	4978.032
E(미)	41.2034	82.4069	164.8138	329.6276	659.2551	1318.510	2637.020	5274.041
F(파)	43.6535	87.3071	174.6141	349.2282	698.4565	1396.913	2793.826	5587.652
F#	46.2493	92.4986	184.9972	369.9944	739.9888	1479.978	2959.955	5919.911
G(솔)	48.9994	97.9989	195.9977	391.9954	783.9909	1567.982	3135.963	6271.927
G#	51.9130	103.8262	207.6523	415.3047	830.6094	1661.219	3322.438	6644.875
A(라)	55.0000	110.0000	220.0000	440.0000	880.0000	1760.000	3520.000	7040.000
A#	58.2705	116.5409	233.0819	466.1638	932.3275	1864.655	3729.310	7458.620
B(시)	61.7354	123.4708	246.9417	493.8833	987.7666	1975.533	3951.066	7902.133

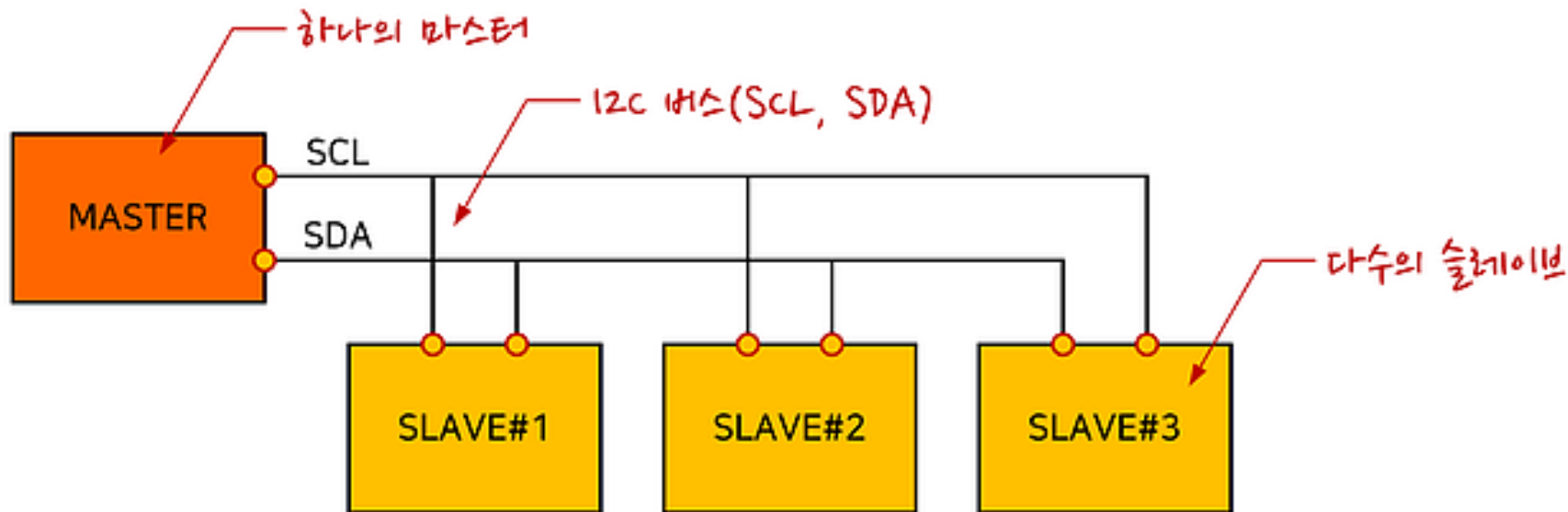


MLX90614

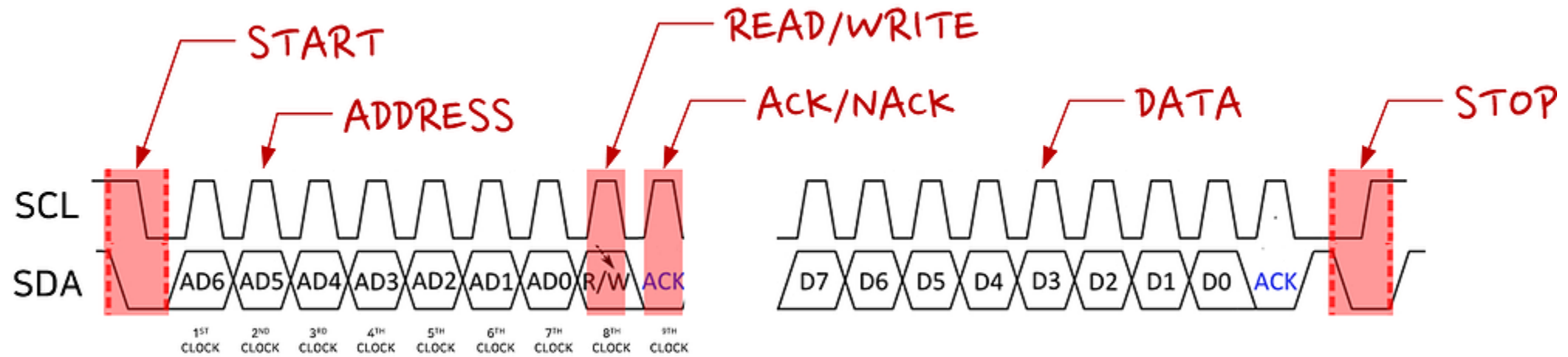
- 비접촉식 온도센서 모듈
- FOV 90°
- 측정범위 : -70°C ~ 380°C
- 인터페이스 : I2C



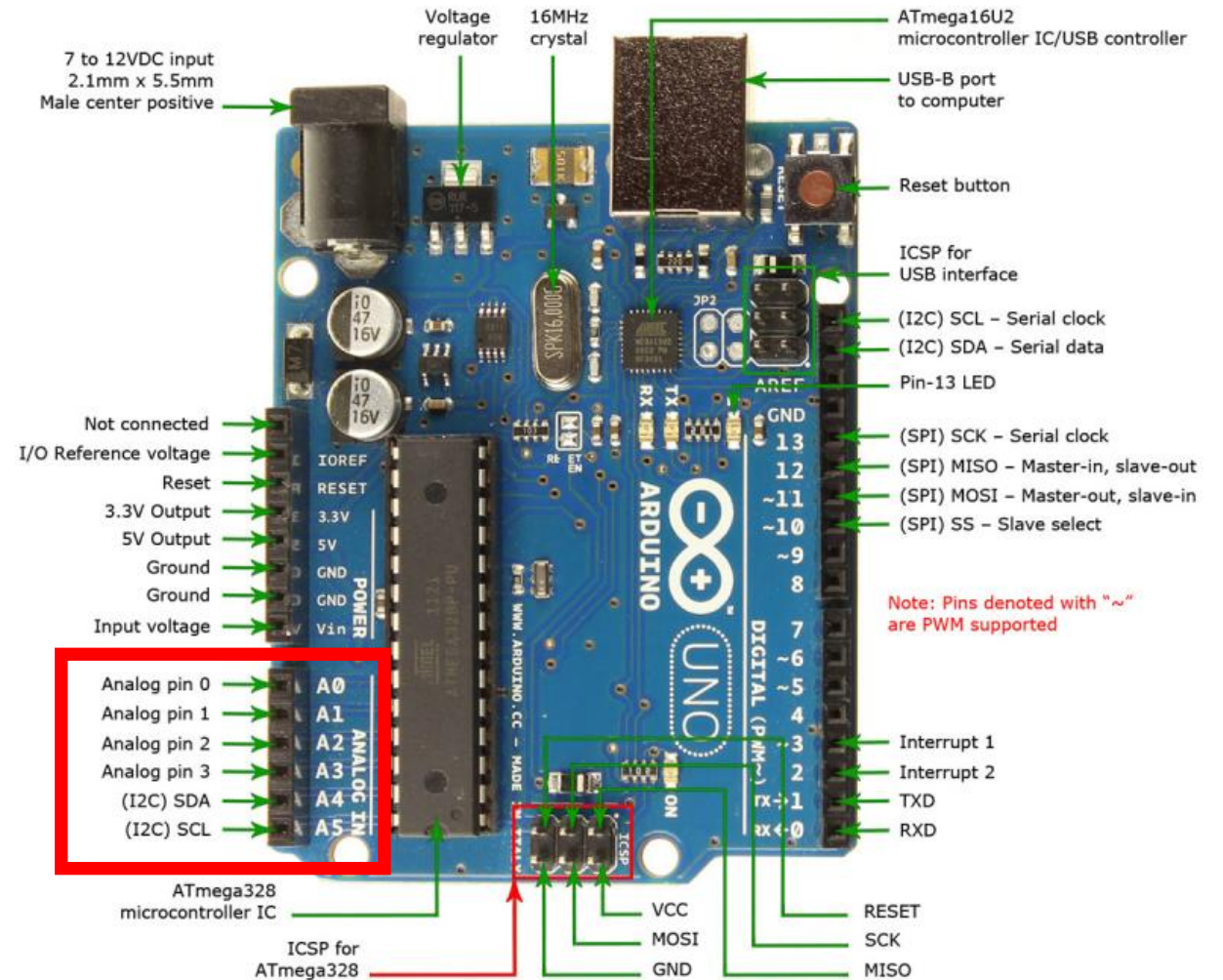
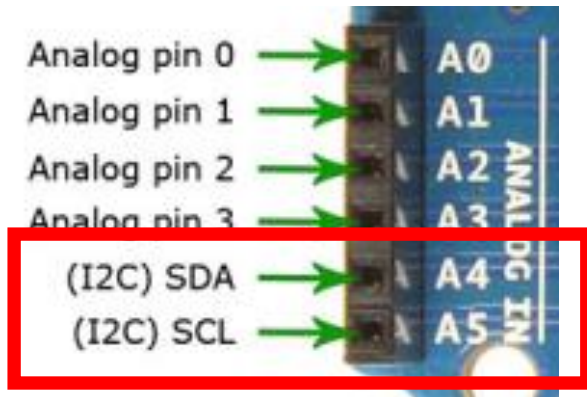
I2C 통신



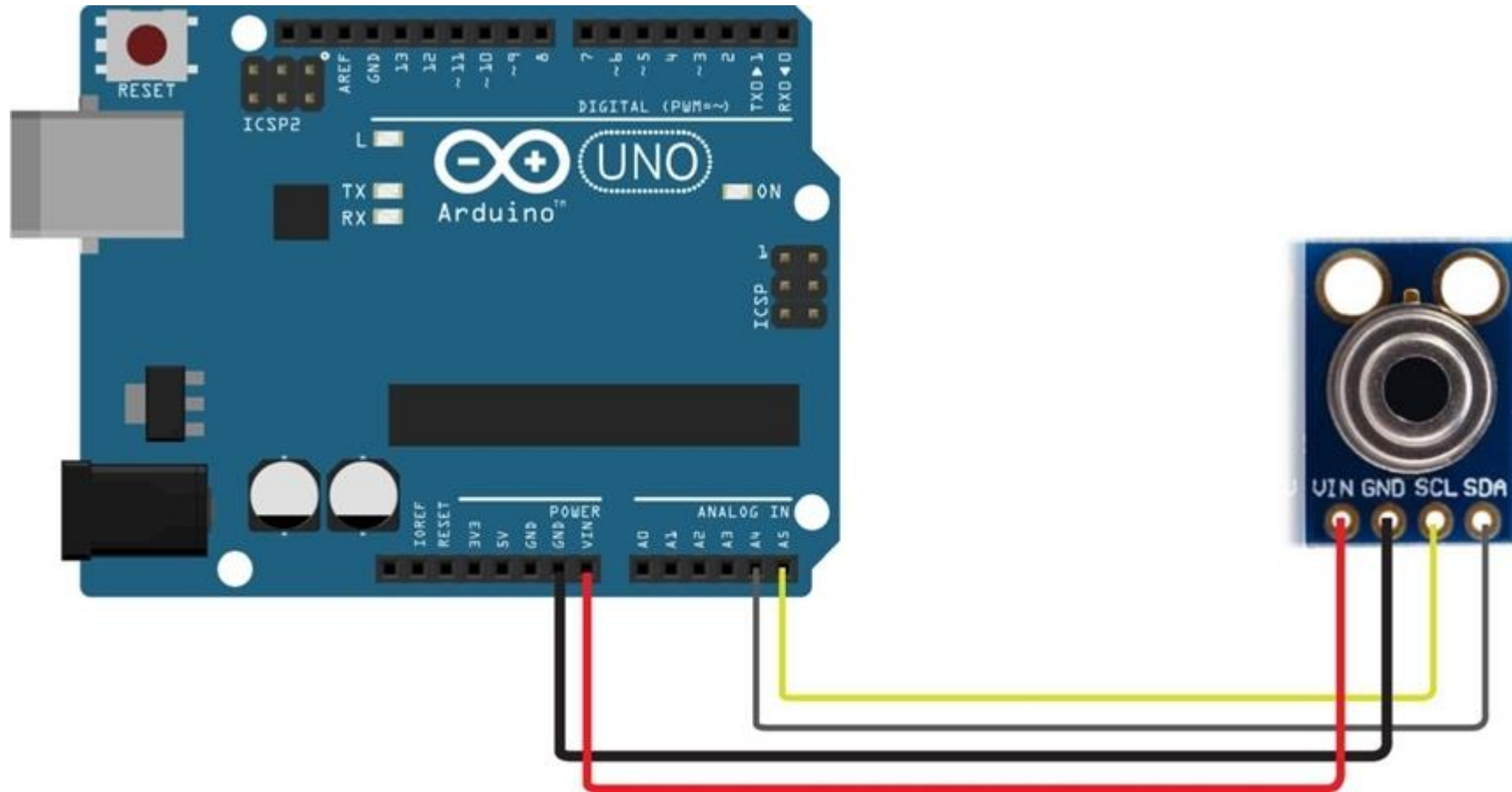
I2C 통신



아두이노의 I2C통신

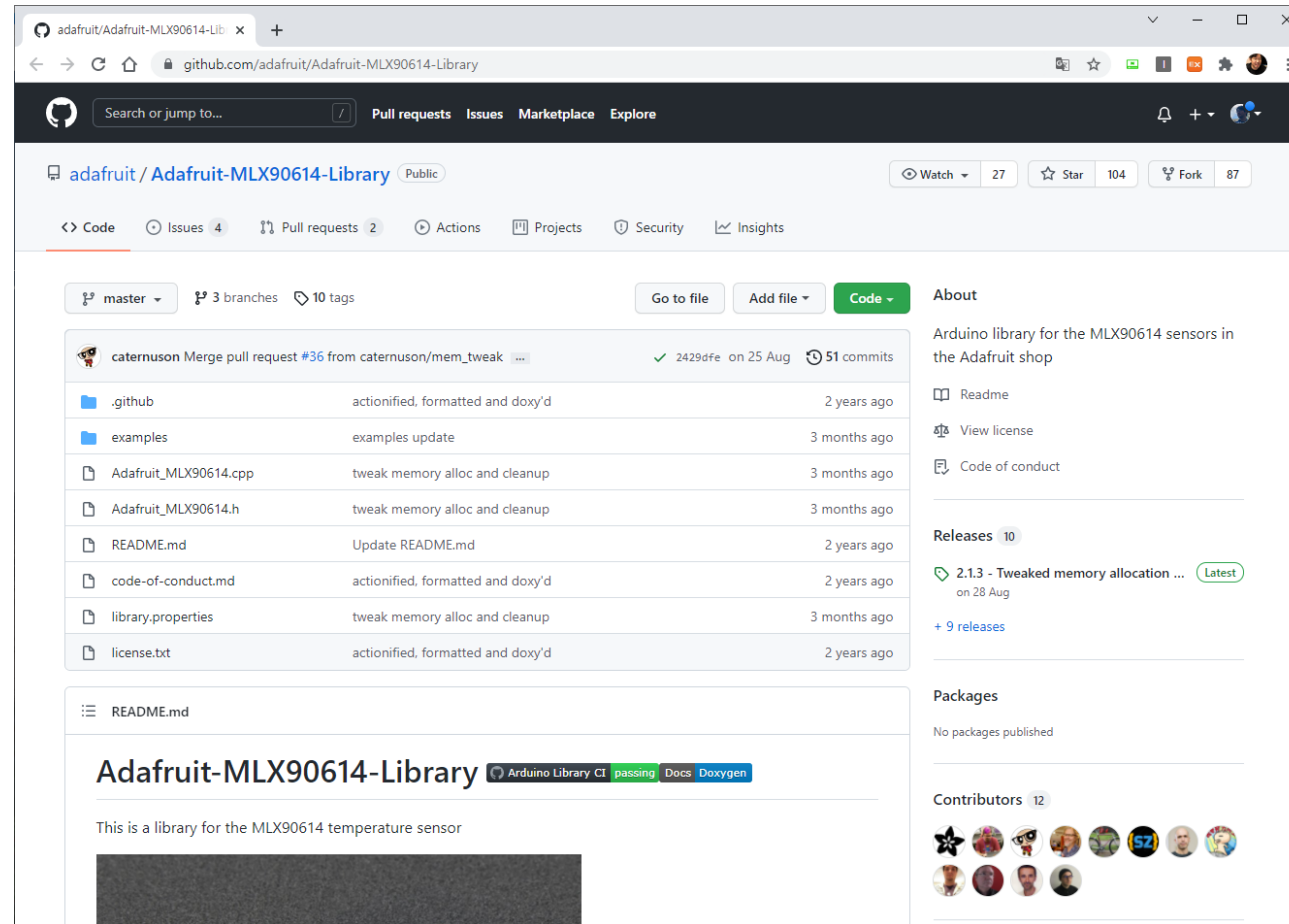


MLX90614 테스트



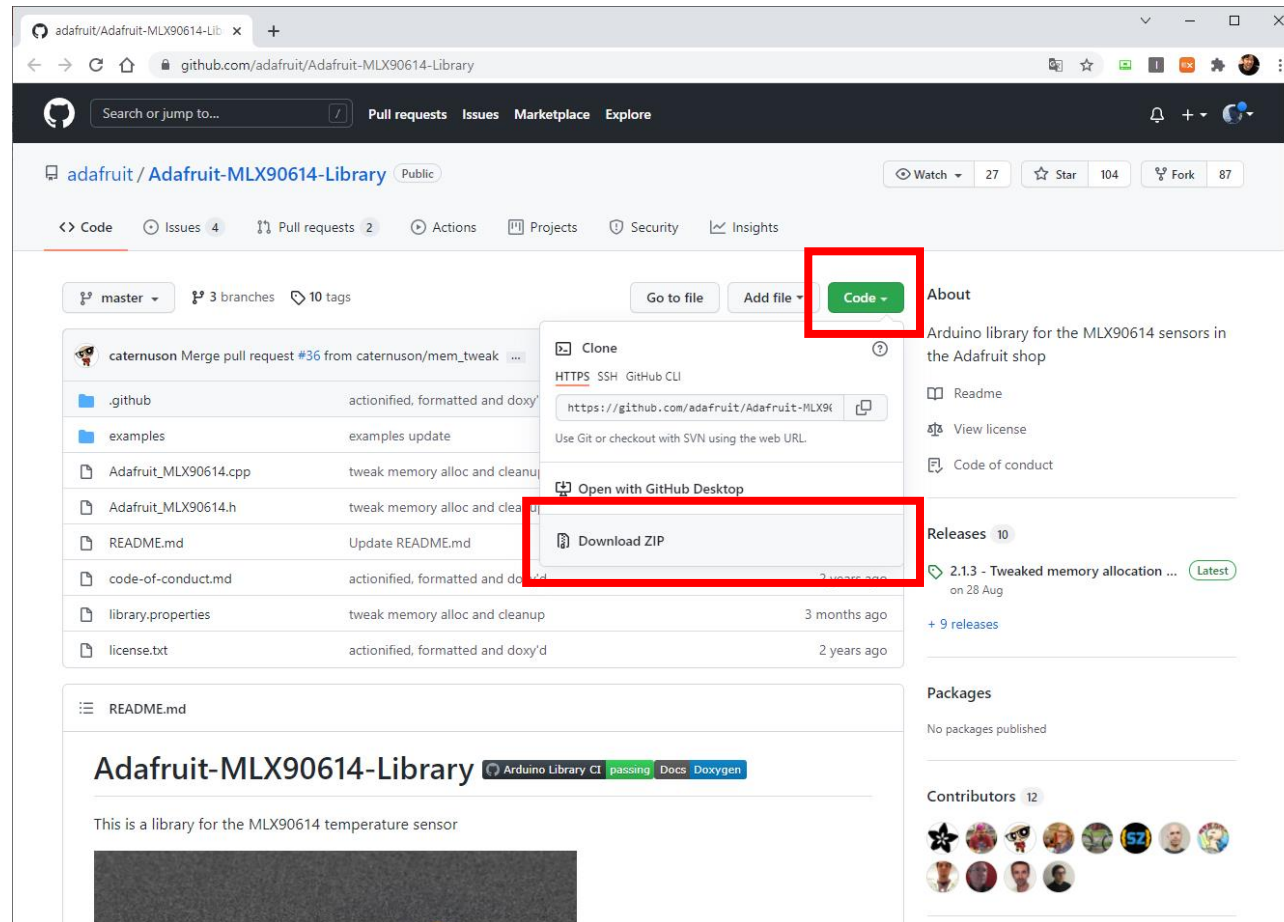
MLX90614 라이브러리 사용

- <https://github.com/adafruit/Adafruit-MLX90614-Library>

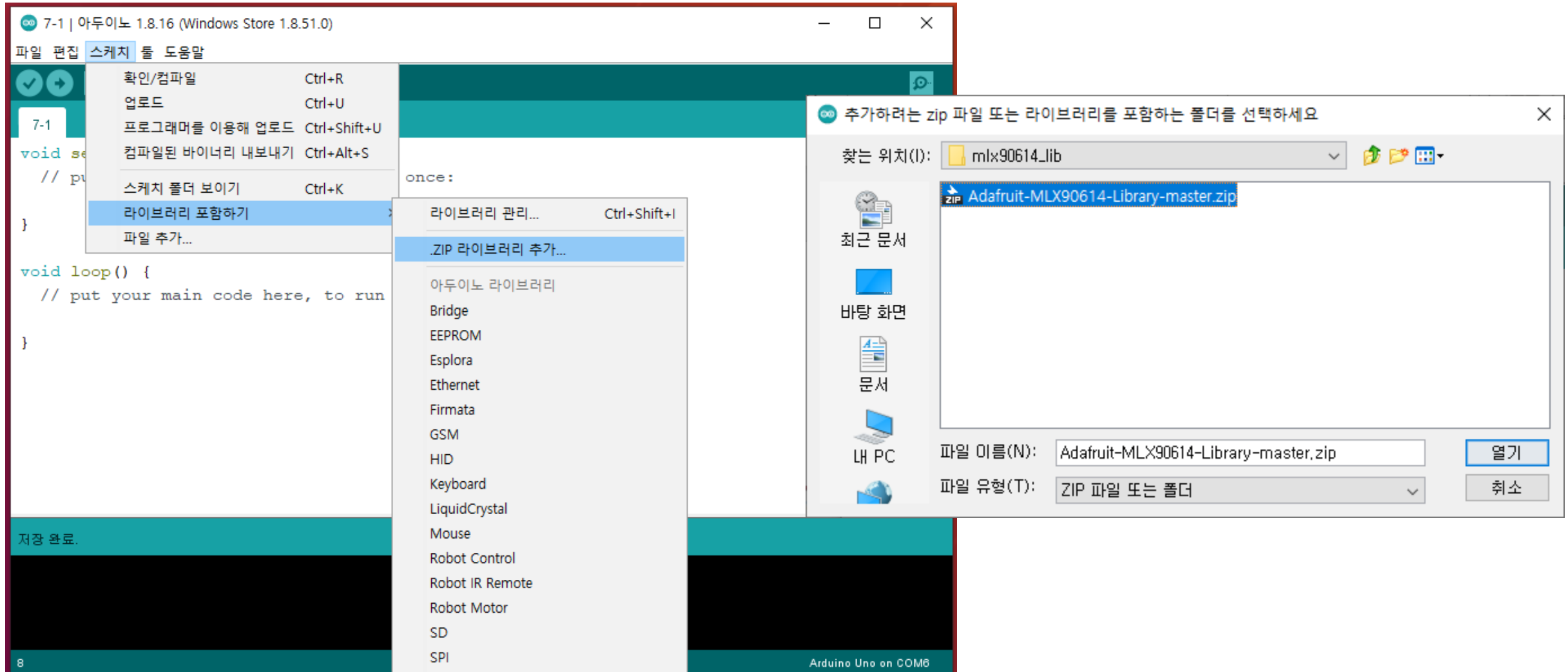


MLX90614 라이브러리 사용

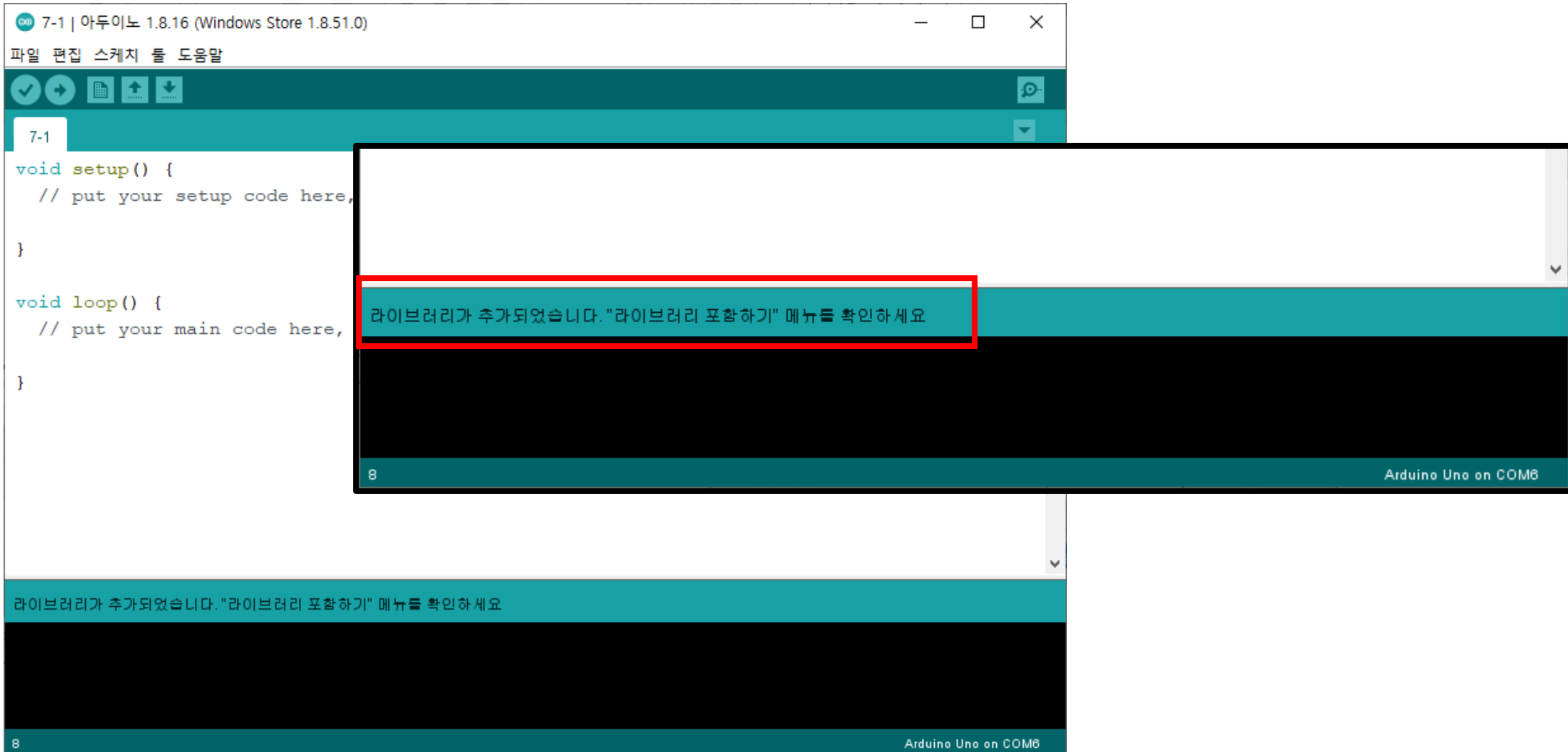
- <https://github.com/adafruit/Adafruit-MLX90614-Library>



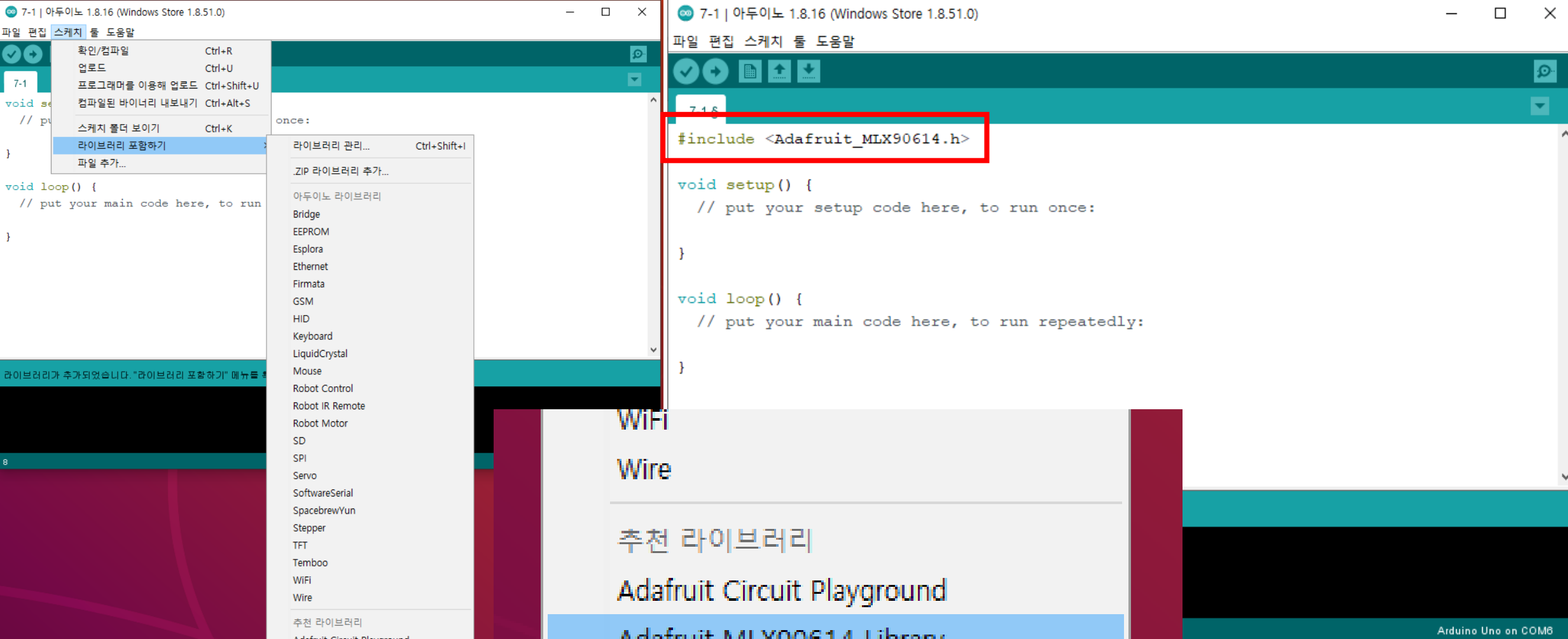
MLX90614 라이브러리 사용



MLX90614 라이브러리 사용



MLX90614 라이브러리 사용



The image shows the Arduino IDE interface with the 'Sketch' menu open. The 'Include Library' option is selected, and a list of libraries is displayed. The 'Adafruit MLX90614 Library' is highlighted. The code editor shows the following code:

```
#include <Adafruit_MLX90614.h>

void setup() {
  // put your setup code here, to run once:
}

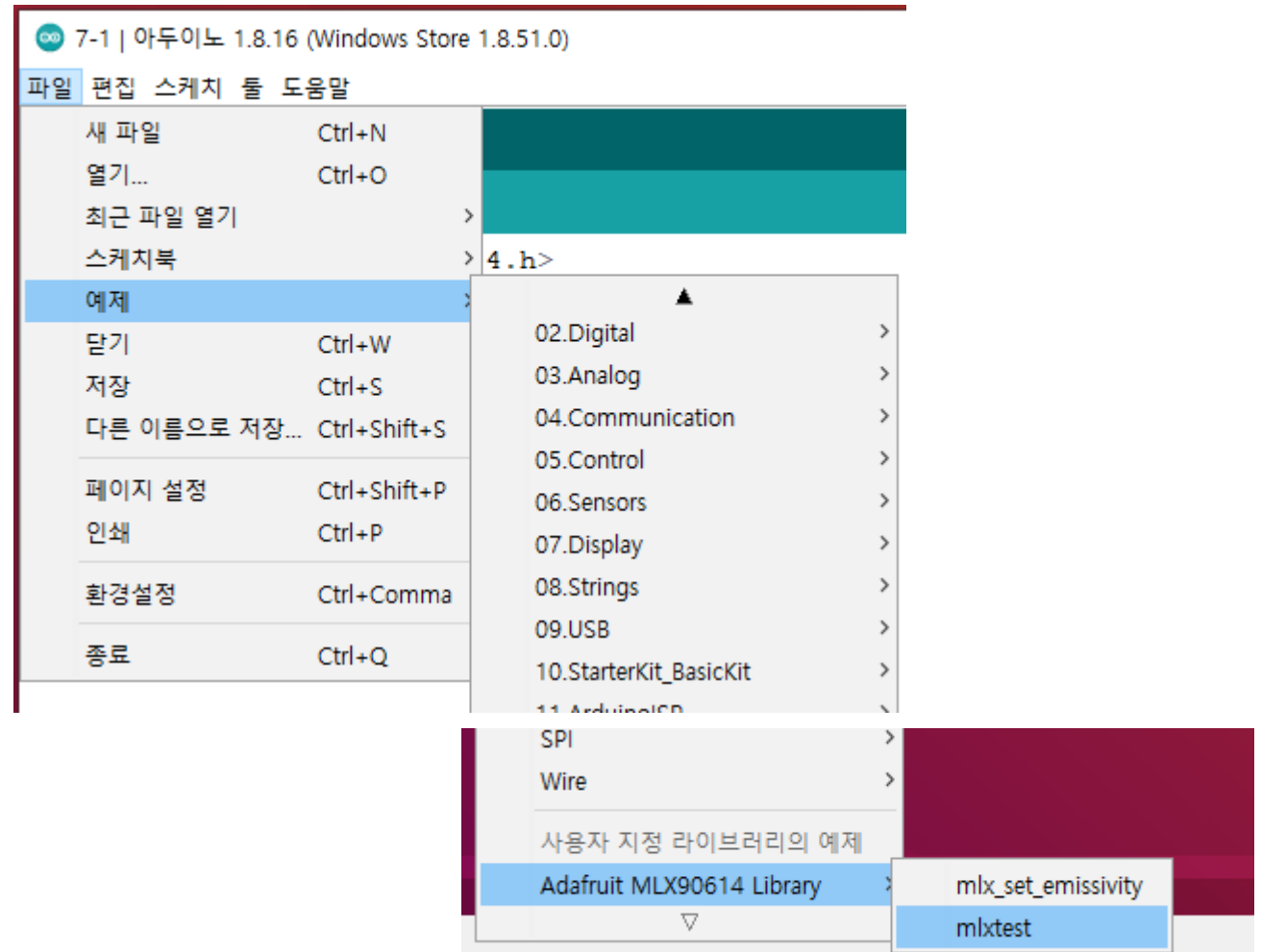
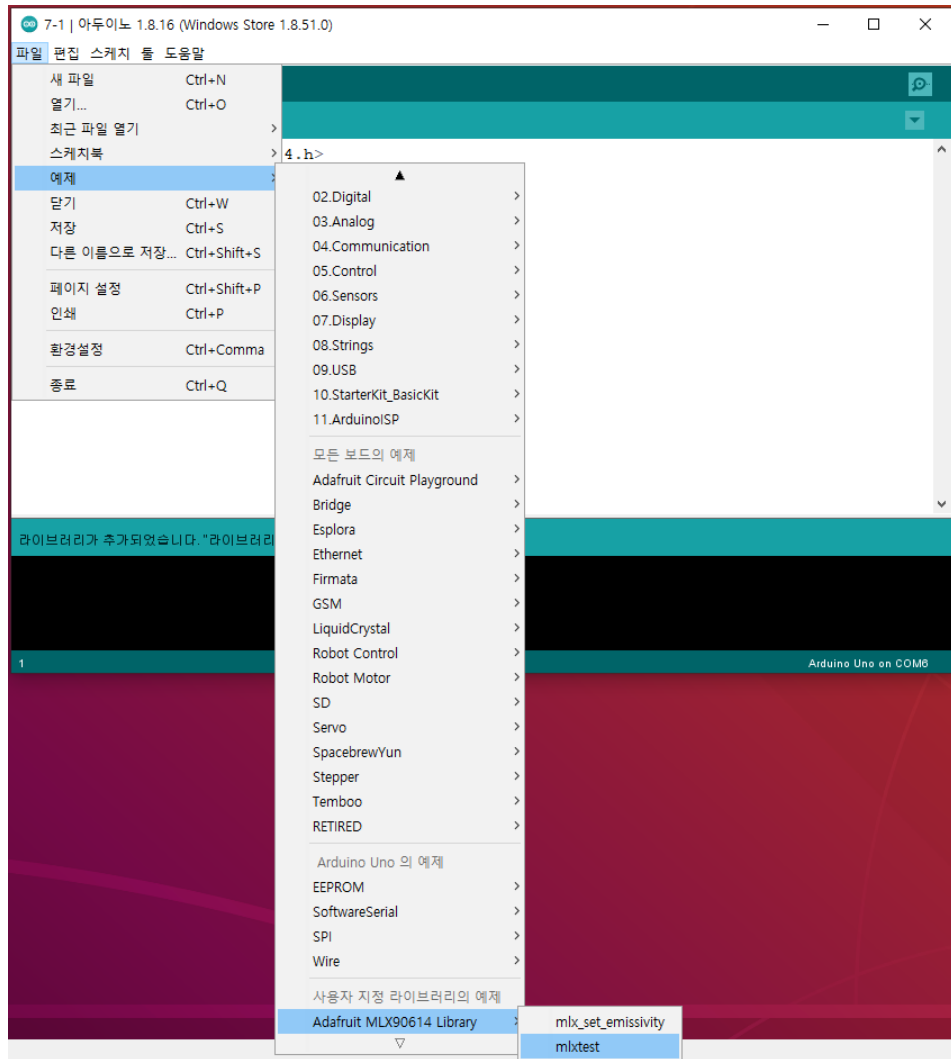
void loop() {
  // put your main code here, to run repeatedly:
}
```

Below the code editor, a list of recommended libraries is shown:

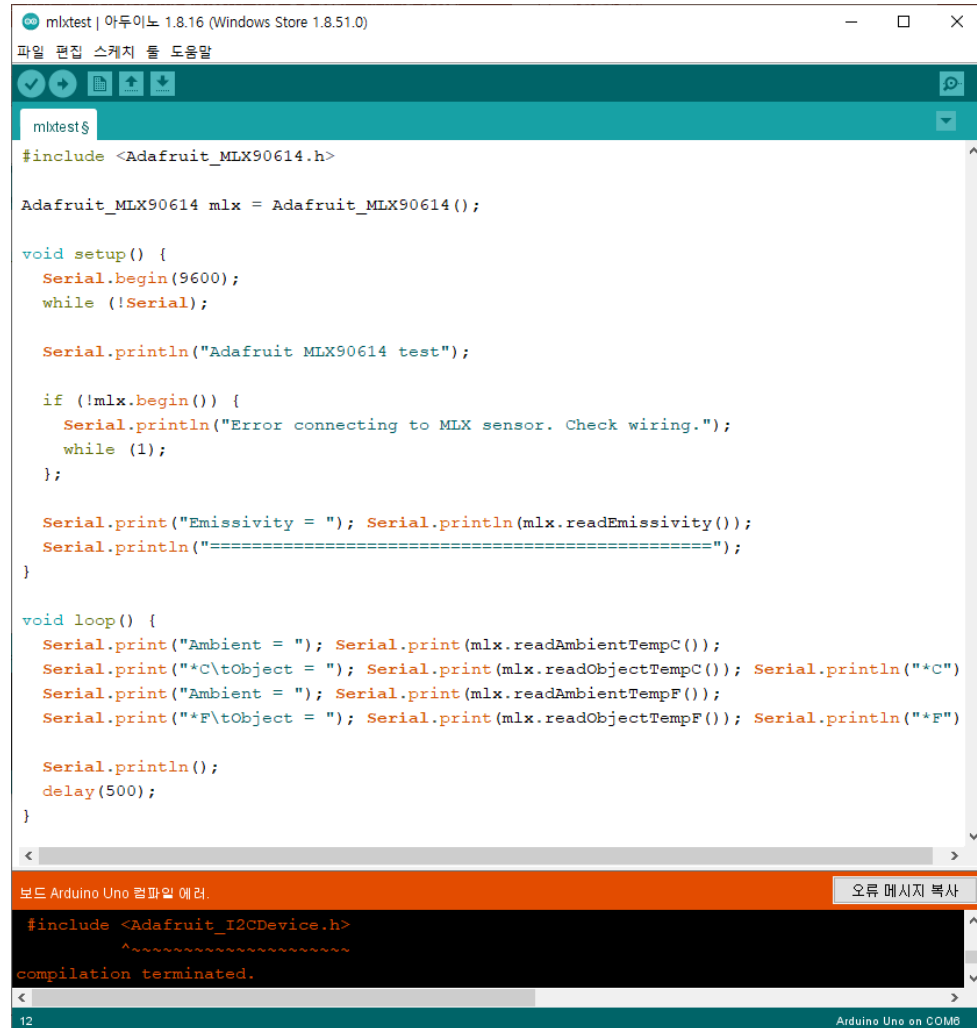
- 추천 라이브러리
- Adafruit Circuit Playground
- Adafruit MLX90614 Library

The bottom right corner of the IDE shows 'Arduino Uno on COM8'.

MLX90614 라이브러리 예제 테스트



MLX90614 라이브러리 예제 테스트



```
mlxtest | 아두이노 1.8.16 (Windows Store 1.8.51.0)
파일 편집 스케치 툴 도움말
mlxtest$
#include <Adafruit_MLX90614.h>

Adafruit_MLX90614 mlx = Adafruit_MLX90614();

void setup() {
  Serial.begin(9600);
  while (!Serial);

  Serial.println("Adafruit MLX90614 test");

  if (!mlx.begin()) {
    Serial.println("Error connecting to MLX sensor. Check wiring.");
    while (1);
  };

  Serial.print("Emissivity = "); Serial.println(mlx.readEmissivity());
  Serial.println("=====");
}

void loop() {
  Serial.print("Ambient = "); Serial.print(mlx.readAmbientTempC());
  Serial.print("C\tObject = "); Serial.print(mlx.readObjectTempC()); Serial.println("C");
  Serial.print("Ambient = "); Serial.print(mlx.readAmbientTempF());
  Serial.print("F\tObject = "); Serial.print(mlx.readObjectTempF()); Serial.println("F");

  Serial.println();
  delay(500);
}

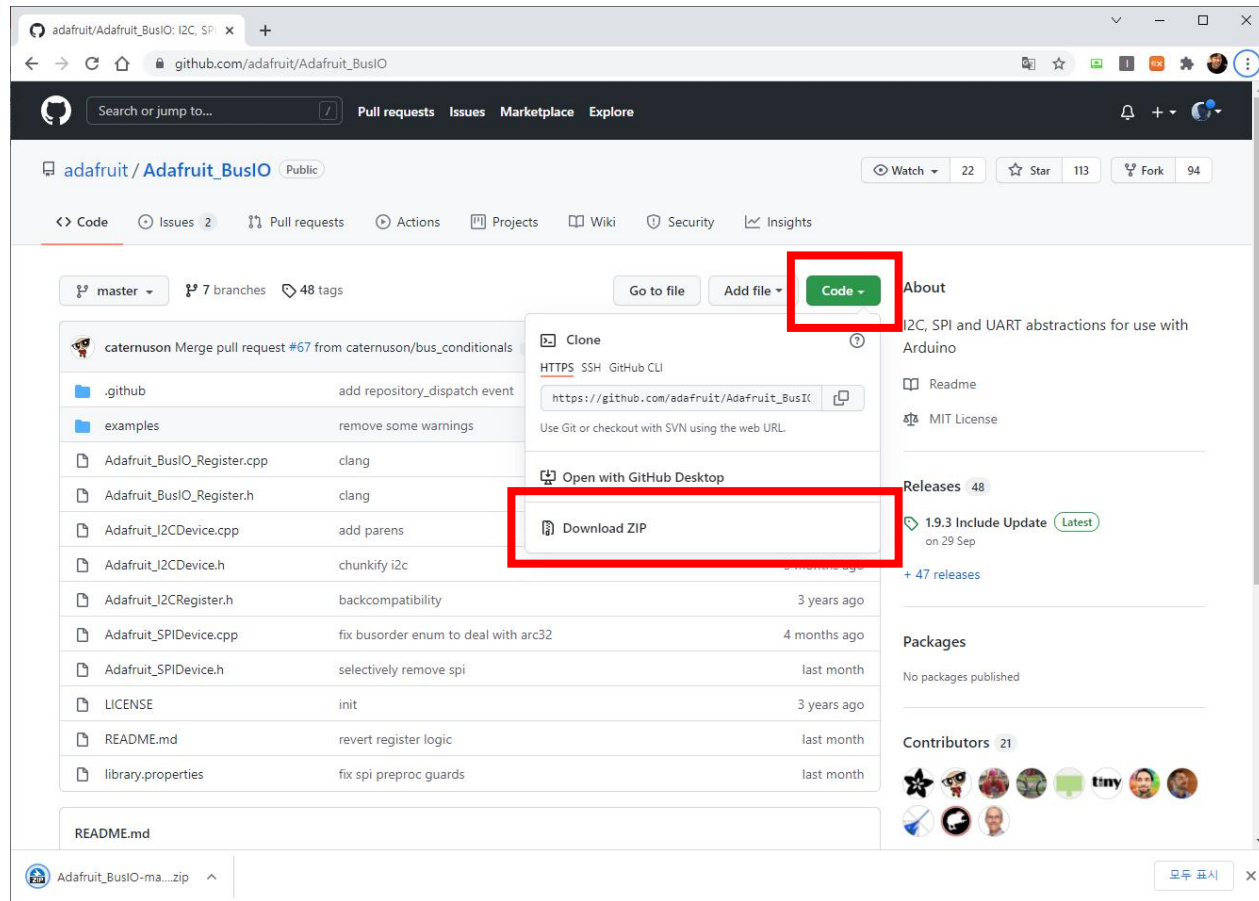
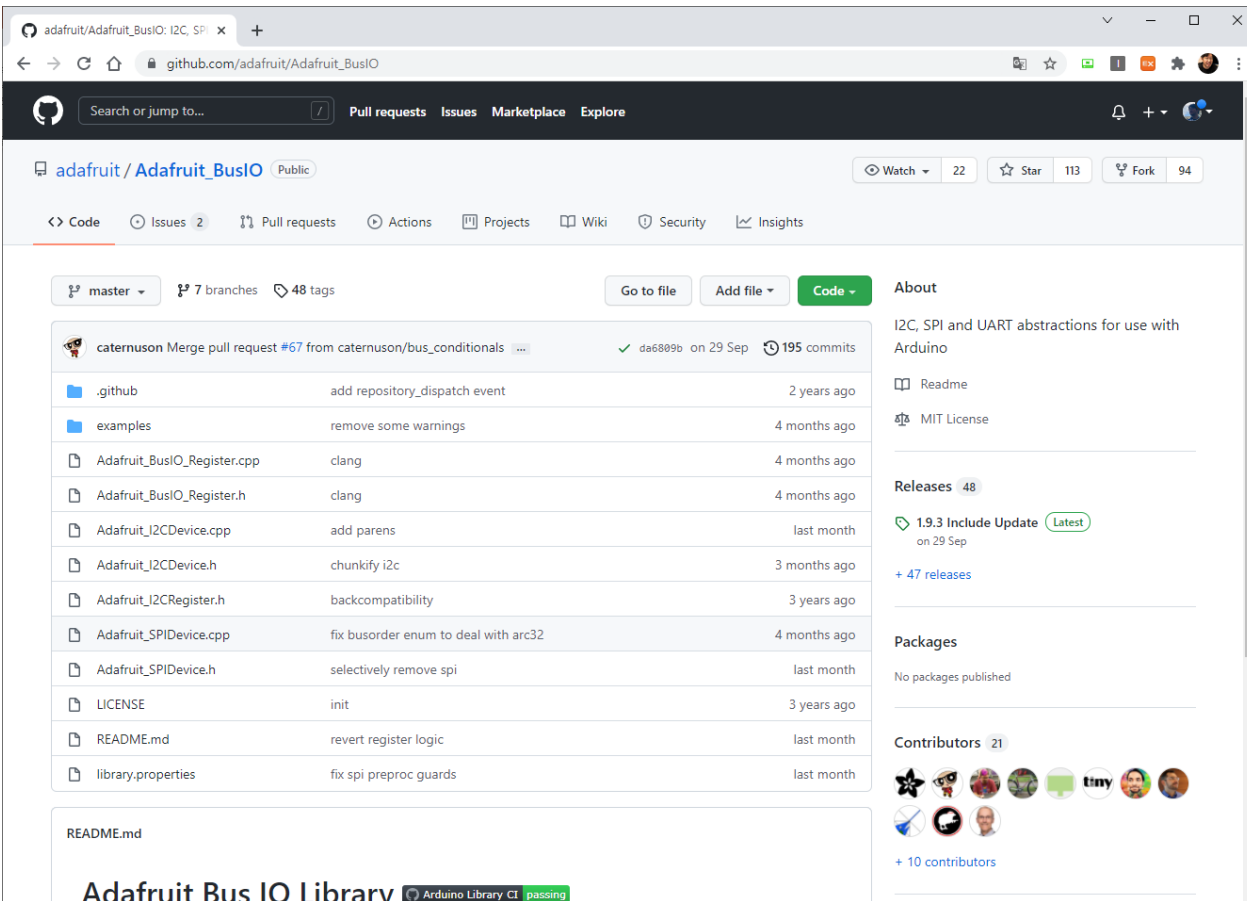
보드 Arduino Uno 컴파일 에러. 오류 메시지 복사
#include <Adafruit_I2CDevice.h>
^~~~~~
compilation terminated.
```

보드 Arduino Uno 컴파일 에러.

```
#include <Adafruit_I2CDevice.h>
^~~~~~
compilation terminated.
```

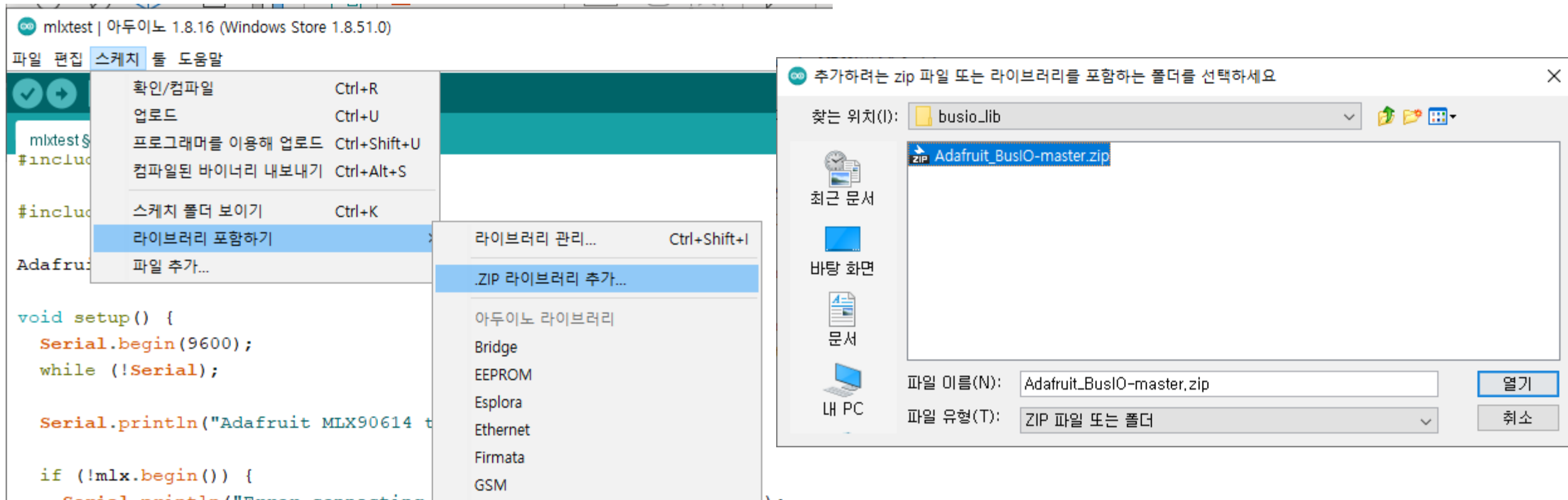
MLX90614 라이브러리 예제 테스트

- https://github.com/adafruit/Adafruit_BusIO 라이브러리 다운로드

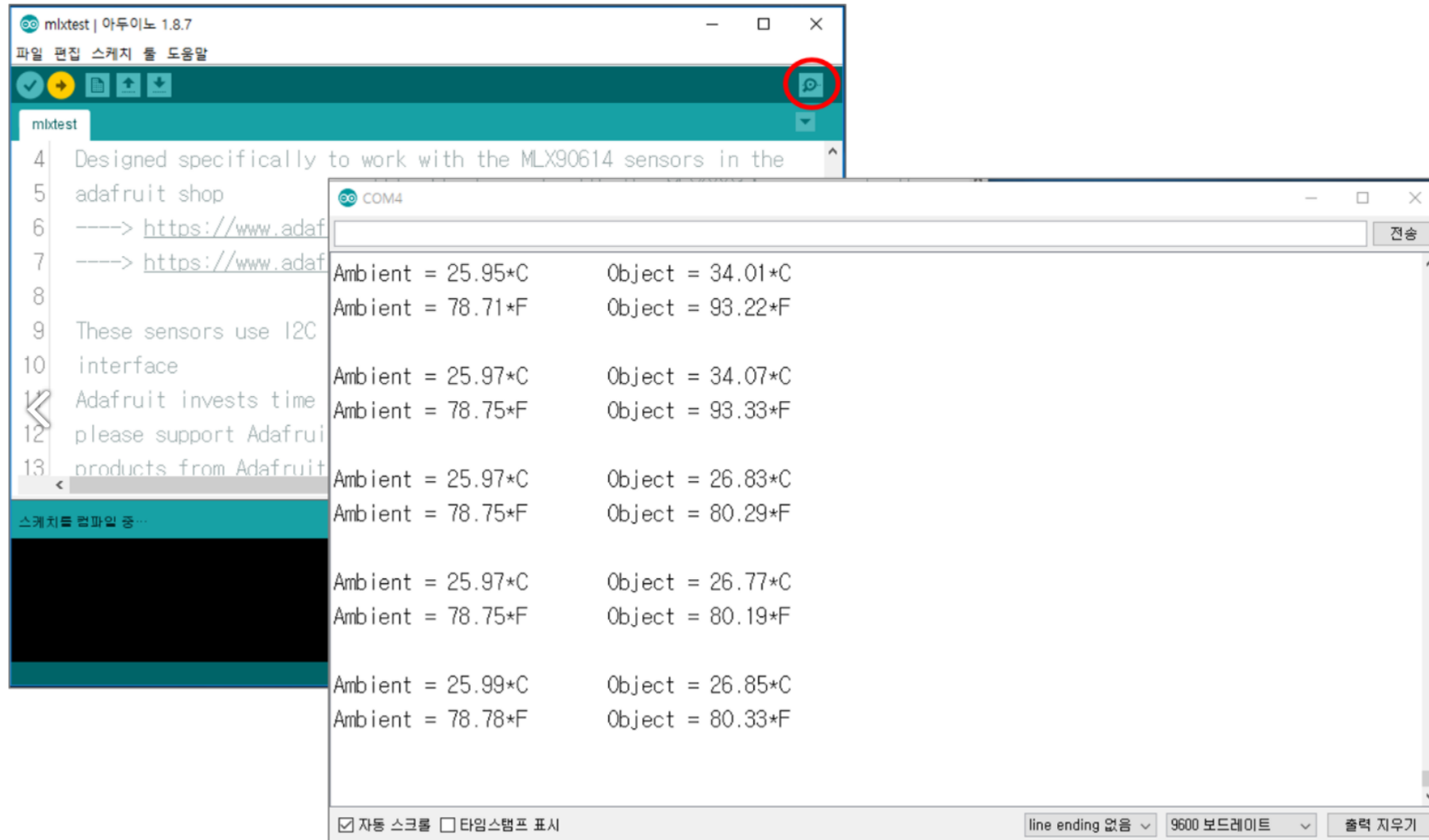


MLX90614 라이브러리 예제 테스트

- https://github.com/adafruit/Adafruit_BusIO 라이브러리 추가



MLX90614 라이브러리 예제 테스트



The screenshot shows the Arduino IDE interface. The main window displays the 'mlxtest' sketch, which is a library example for the MLX90614 sensor. The code includes comments and two serial print statements. A red circle highlights the 'Serial' icon in the top right corner of the IDE window.

The serial monitor window, titled 'COM4', shows the output of the sketch. It displays two columns of data: 'Ambient' and 'Object' temperatures in both Celsius and Fahrenheit. The data is updated periodically.

```
4 Designed specifically to work with the MLX90614 sensors in the
5 adafruit shop
6 ----> https://www.adafruit.com/products/3916
7 ----> https://www.adafruit.com/products/3916
8
9 These sensors use I2C
10 interface
11 Adafruit invests time and resources into developing and
12 please support Adafruit open source projects by donating
13 products from Adafruit!

스케치를 컴파일 중...
```

Ambient (C)	Object (C)
25.95	34.01
78.71	93.22
25.97	34.07
78.75	93.33
25.97	26.83
78.75	80.29
25.97	26.77
78.75	80.19
25.99	26.85
78.78	80.33

Serial Monitor Settings: ☒ 자동 스크롤 ☐ 타임스탬프 표시 | line ending 없음 | 9600 보드레이트 | 출력 지우기

QUIZ : 체온 측정 출입관리 시스템 구성

- 사람이 감지 되면 자동으로 온도를 측정하고
- 측정된 온도가 37.5도 이상이면 RED LED를 켜고 부저의 소리를 발생한다.
- 그렇지 않으면 GREEN LED를 켜고 부저의 소리를 발생한다.

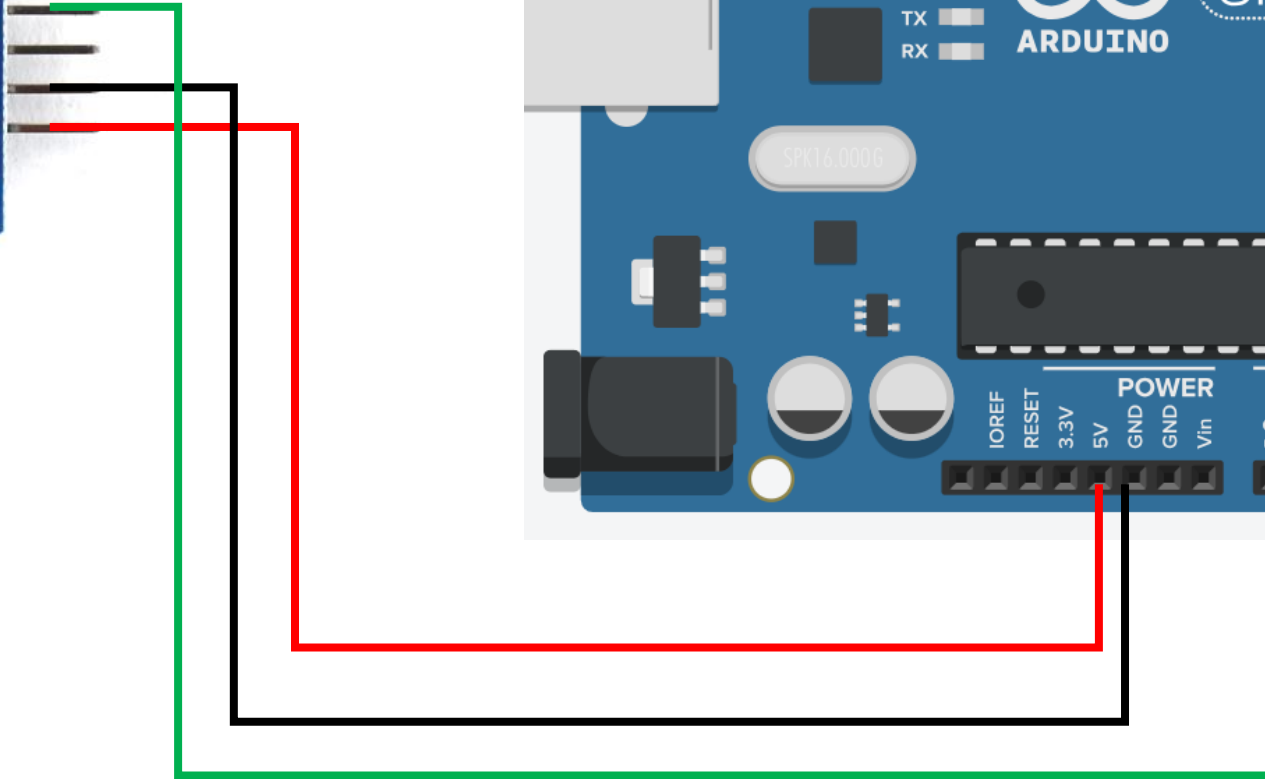
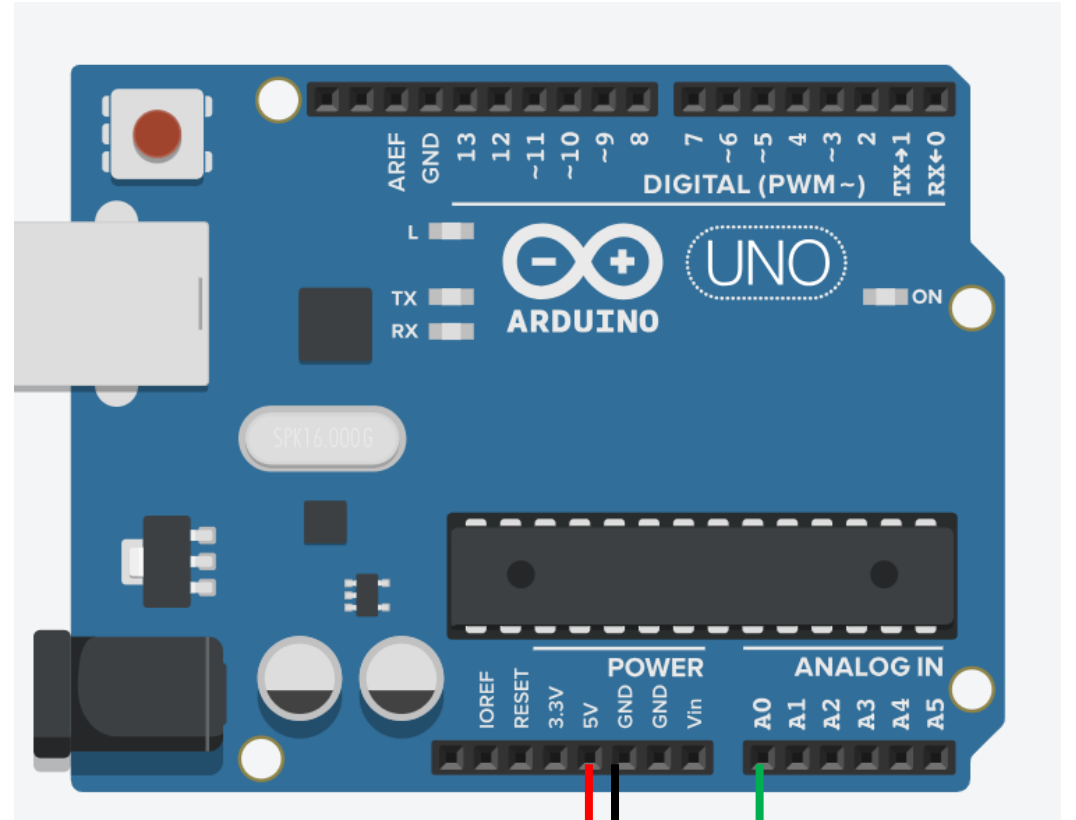


음주 측정기 만들기

- 알코올 센서



음주 측정기 만들기



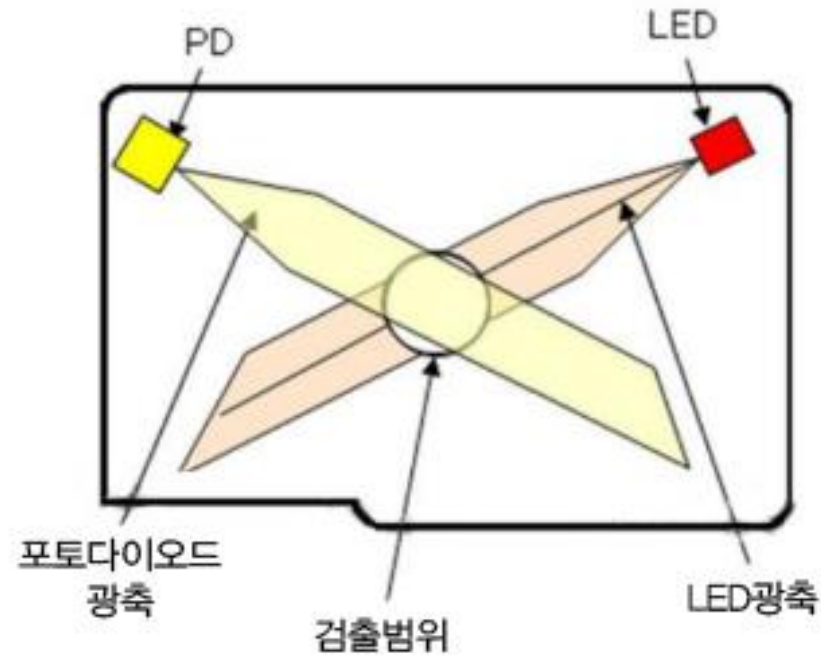
QUIZ

- LCD에 현재 알코올 레벨을 표시하고 300이상인 경우 부저를 이용하여 소리를 출력한다.



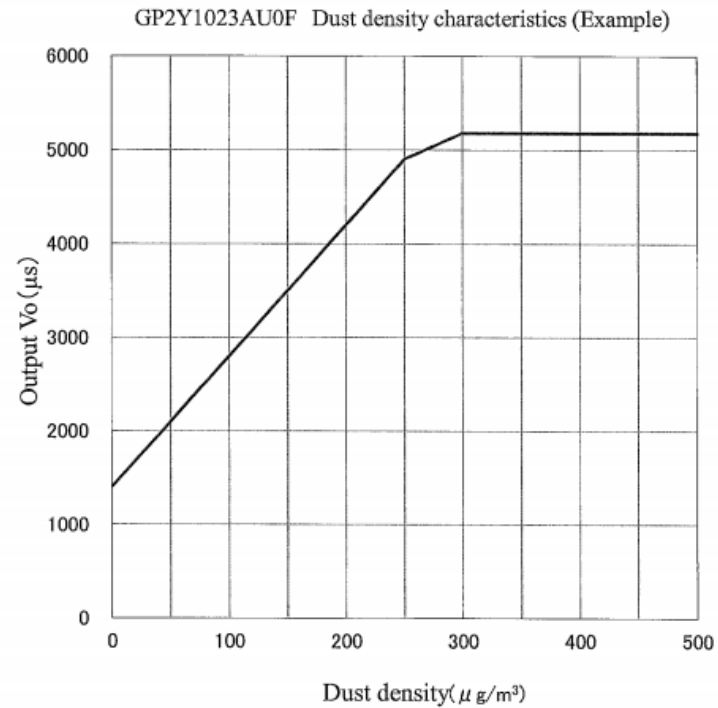
PM2.5 GP2Y1023AU0F 먼지센서

- 빛을 이용하여 먼지가 얼마나 많은지를 측정

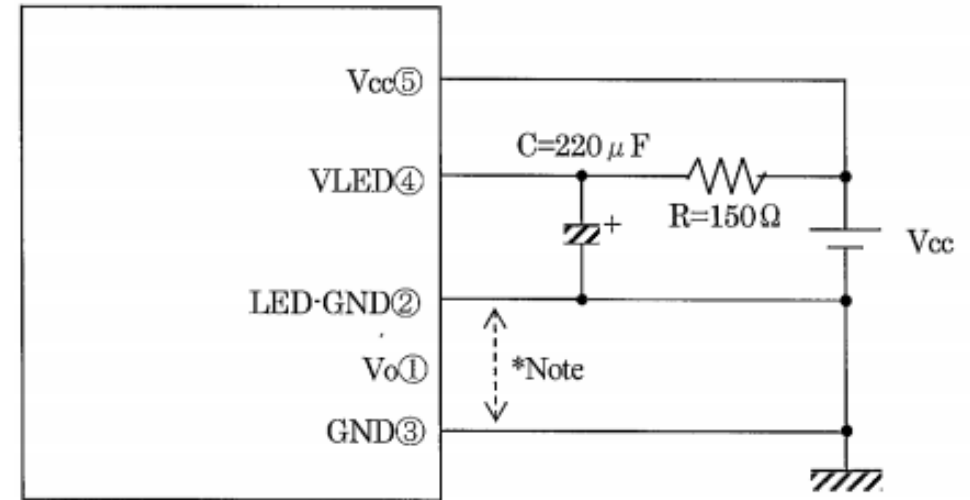
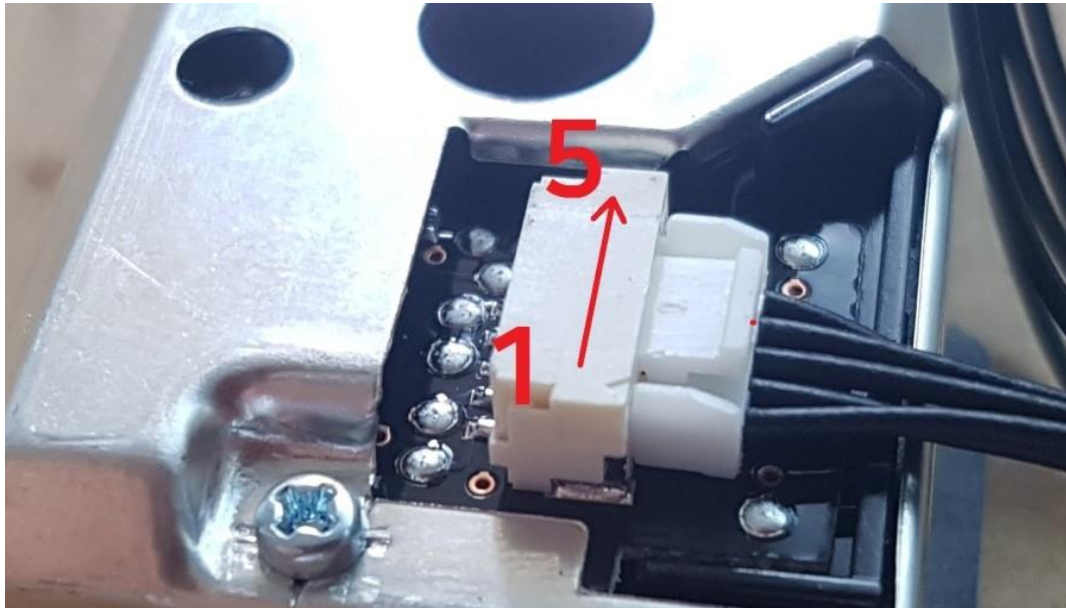


PM2.5 GP2Y1023AU0F 먼지 센서

- 빛을 이용하여 먼지가 얼마나 많은지를 측정



미세먼지 센서 회로 구성

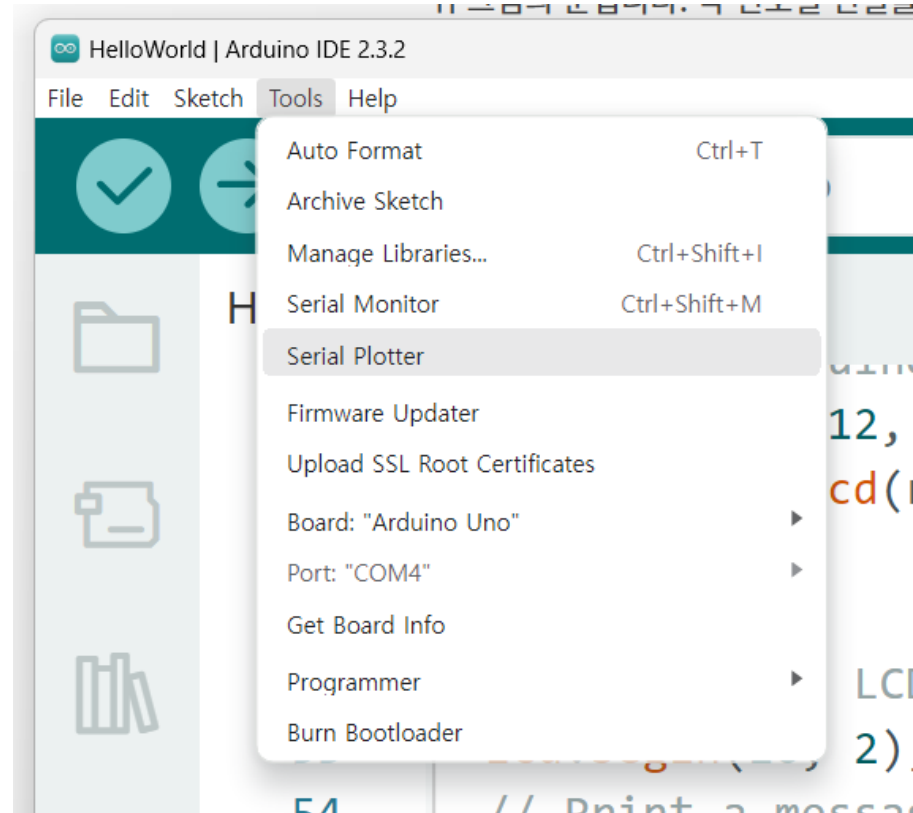


*Note : LED-GND line and GND line must be separated.

미세먼지 센서의 데이터 확인

```
void setup()
{
  pinMode(8, INPUT);
  Serial.begin(9600);
}

void loop()
{
  int value = digitalRead(8) ;
  Serial.println(value);
}
```



미세먼지 센서의 데이터 확인

시리얼 플로터를 활용하여 PWM 신호 확인

