임베디드컴퓨팅
Embedded Computing
(0009488)
Dust sensor

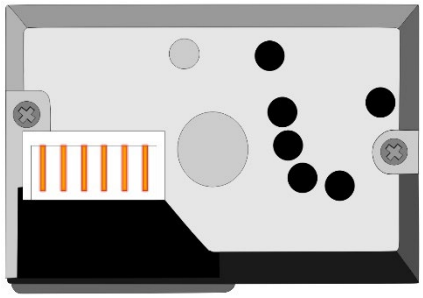
2022년 2학기

정보기술대학 정보통신공학과
김 영 필
ypkim@inu.ac.kr

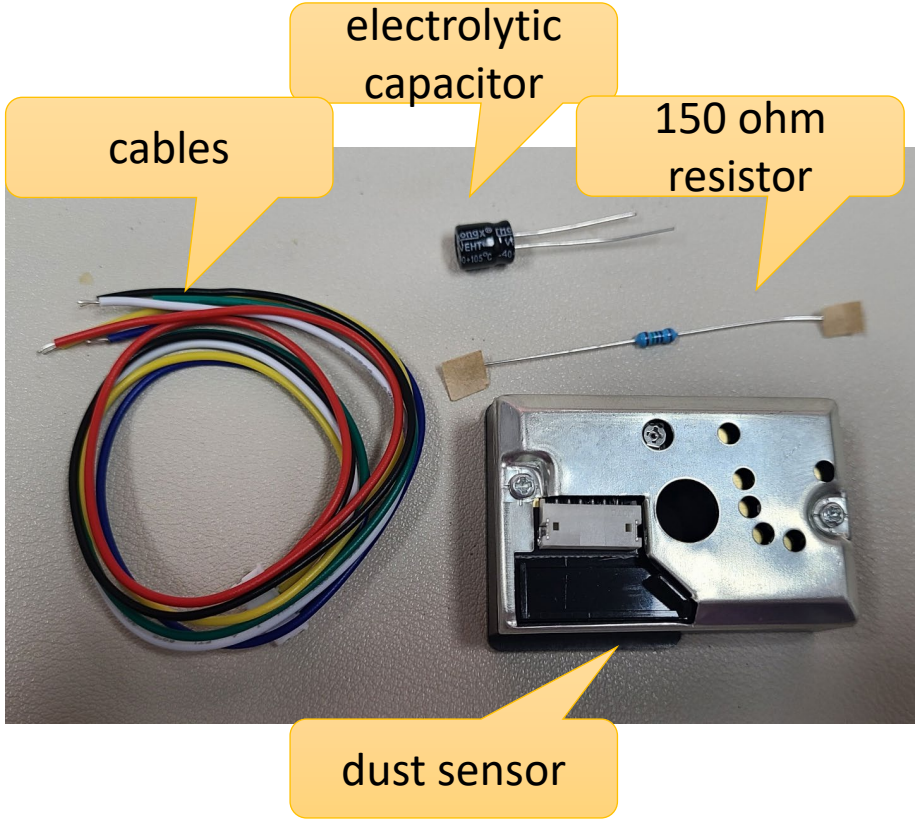
INU 인천대학교 INCHEON NATIONAL UNIVERSITY

Dust sensor

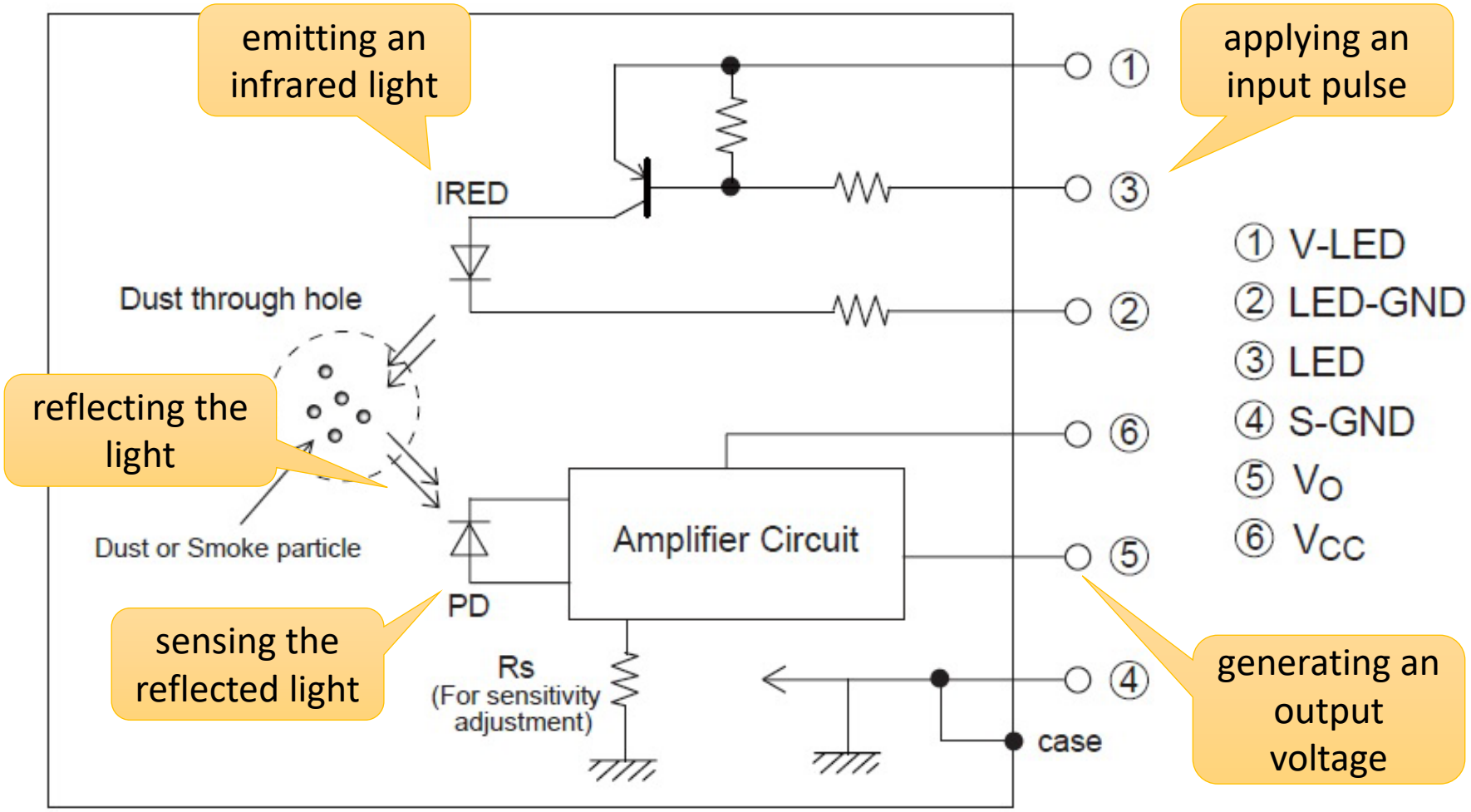
- A sensor that **measures fine dust in the air** by using reflected light with an infrared light emitting diode (IRED) and phototransistor.
- The **pulse pattern of the** [redacted] can distinguish house dust from smoke and is particularly **effective in detecting very fine particles** such as cigarette smoke.



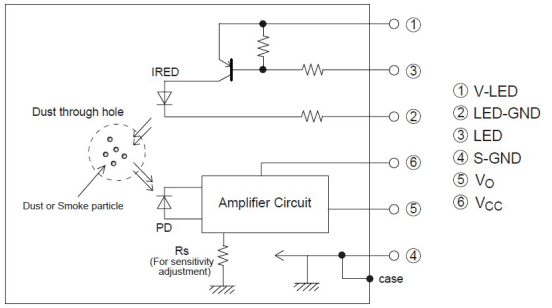
(GP2Y1010AU0F)



Internal circuit diagram of dust sensor



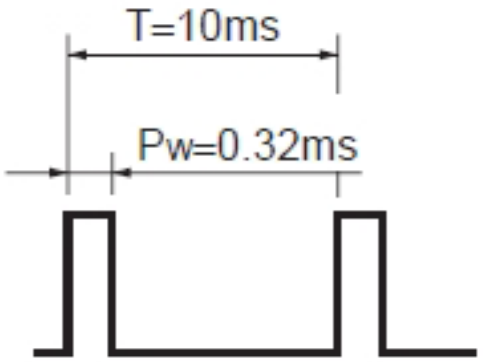
6 pins of dust sensor



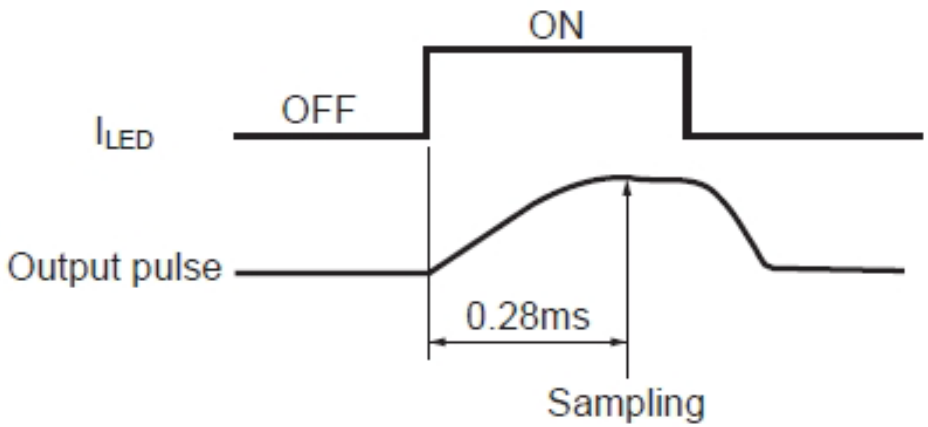
- V-LED
  - Applies + power to the **emitter part** of the PNP transistor.
  - Connected to the contact part where the 150Ω resistor and the 220μF electrolytic capacitor are connected.
- LED-GND
  - connected to the GND of the Arduino.
- LED
  - Controls the IRED, connect to the digital pin of the Arduino.
- S-GND
  - Connected to the GND of the Arduino.
- $\boldsymbol{V\_o}$
  - Outputs a voltage (0 to 1023V). Connect to the analog input pin of the Arduino.
- $\boldsymbol{V}$_cc
  - Applies the operating voltage (4.5~5.5V) of the fine dust sensor
  - Connect to the 5V pin of the Arduino.

How to measure fine dust?

- **(Starting) IRED lights up when a pulse is applied for 0.32ms**
  - The entire cycle is 10ms

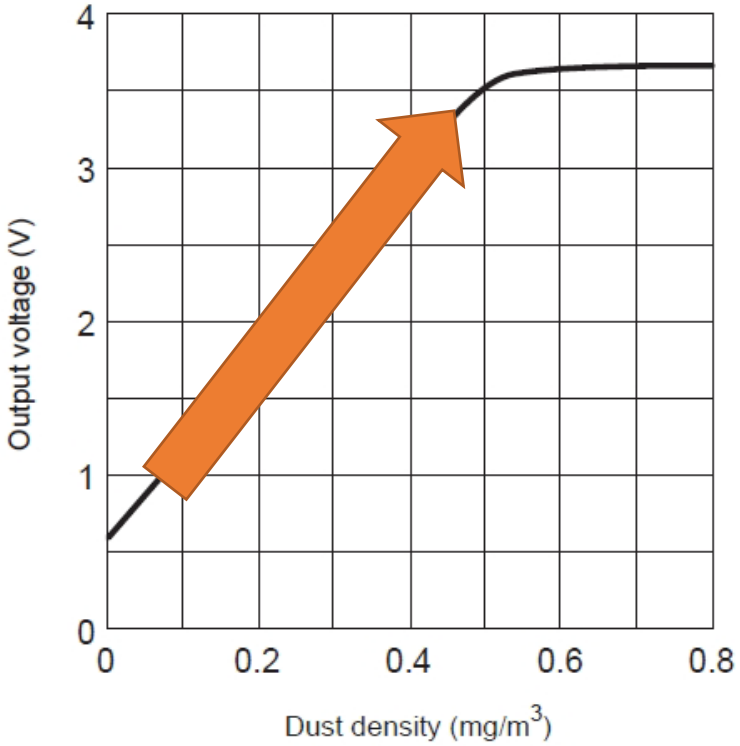


- **(Sampling)** The **output pulse is sampled after 0.28 ms** of the 0.32 ms of the IRED is lit.
- **(Converting)** The **sampled output pulse** has a value in the range of 0~1023V, and change it to the value in the range of range of **0~5V.**



How to measure fine dust?

- **(Sensitivity)** Sensitivity is typically seen as a voltage change of **0.5V per 0.1mg/m^3**.
  - For fine dust unit (**μg/m^3**), the voltage change rate per **1μg/m^3** becomes



(T_a=25°C, V_{CC}=5V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Sensitivity	K	*1 *2 *3	0.35	0.5	0.65	V/(0.1mg/m³)
Output voltage at no dust	V_{OC}	*2 *3	0	0.9	1.5	V

How to measure fine dust?

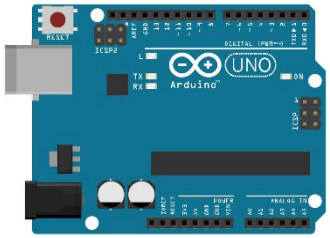
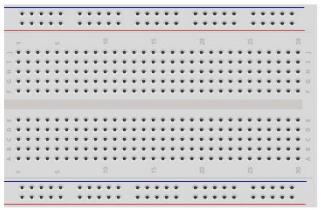
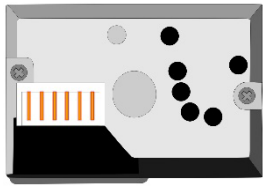
- **(Measuring)** Divide the difference between the output voltage (***V_o***) in the presence of dust in the air and the voltage in the clean air (***V_oc***) by the sensitivity of 0.005 V.
- **(V_o - V_oc) /**
  - V_oc = Typically, 0.9 V, but in the real air condition, **0.3 V**

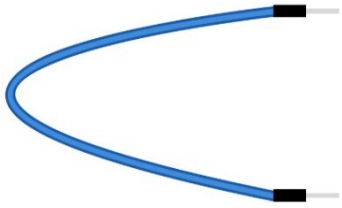
(T_a=25°C, V_{CC}=5V)

Parameter	Symbol	Conditions	MIN.	TYP.	MAX.	Unit
Sensitivity	K	*1 *2 *3	0.35	0.5	0.65	V/(0.1mg/m³)
Output voltage at no dust	V_{OC}	*2 *3	0	0.9	1.5	V

Lab: Measuring fine dust density

- Let's write a sketch program to display the result of measuring fine dust density via serial communication.
- Required H/W components

Arduino board (Uno)	Bread board (400 pins)	Dust sensor (GP2Y1010AU0F)	Resistor (150Ω)
			
x 1	x 1	x 1	x 1

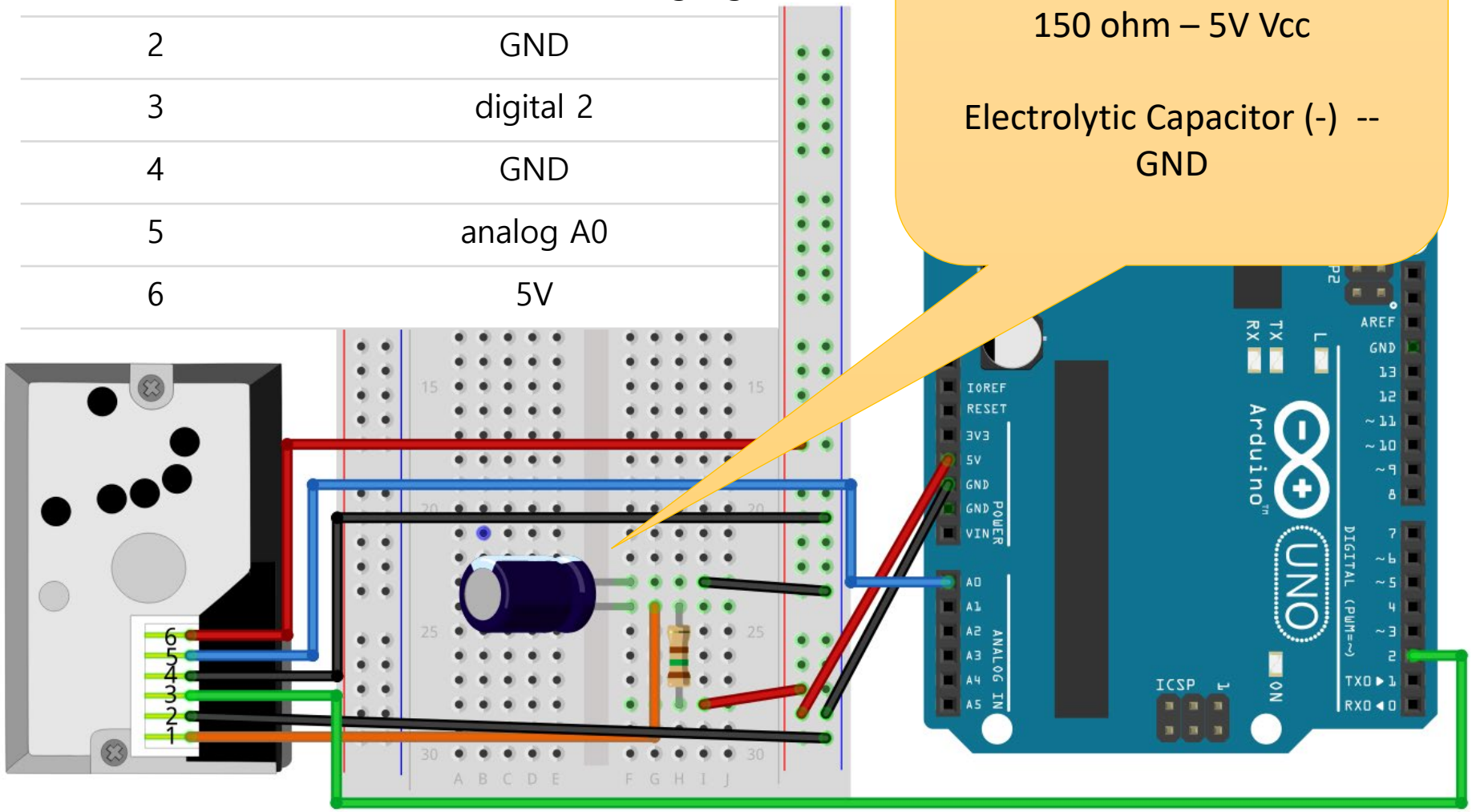
electrolytic capacitor (220uF)	Jumper cable Male-Male
	
x 1	x 10

Circuit wiring setup

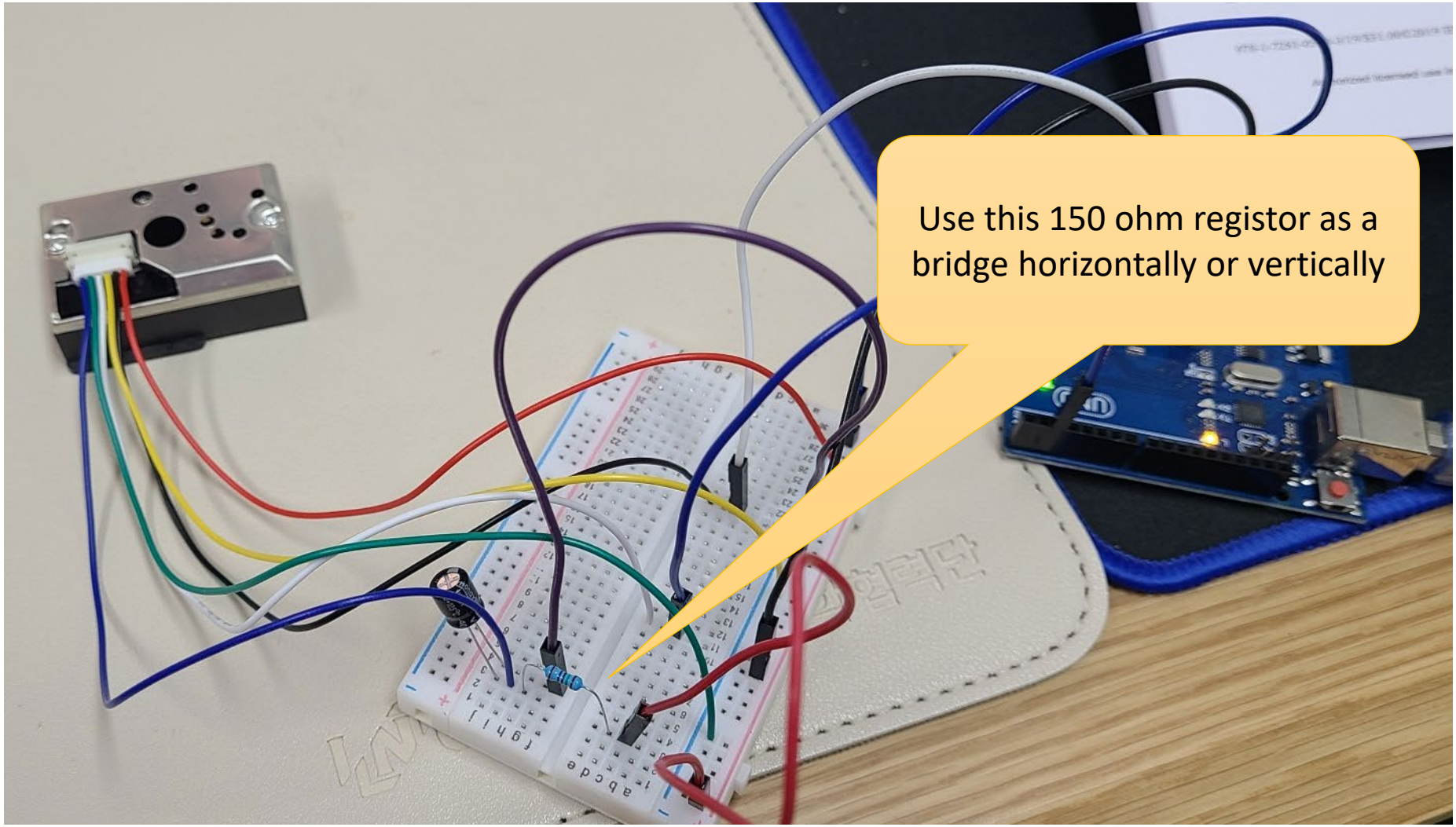
Dust sensor pins	Arduino board
1	electrolytic capacitor (+, long leg)
2	GND
3	digital 2
4	GND
5	analog A0
6	5V

Electrolytic Capacitor (+)
-- dust sensor pin 1 – registor 150 ohm – 5V Vcc

Electrolytic Capacitor (-) -- GND



Circuit wiring setup: 7 cables



Basic setup for dust sensor

```
#define INPUT_PULSE 2
#define OUTPUT_VOLTAGE A0

float preVoltage = 0; //0~1023V
float voltage = 0; //0~5V
float dustDensity = 0;
float sumDustDensity = 0;
float avgDustDensity = 0;

void setup()   {
  pinMode(INPUT_PULSE, OUTPUT);
  pinMode(OUTPUT_VOLTAGE, INPUT);
  Serial.begin(9600);
}
```

digital pin 2 for writing input pulse

analog pin A0 for reading output voltage

convert output voltages

Loop for dust sensor

```
void loop() {
  sumDustDensity = 0;
  for(int i=0;i<30;i++) {
    digitalWrite(                    );
    delayMicroseconds(280); //0.28ms
    preVoltage =                     
    delayMicroseconds(40); //0.04ms
    digitalWrite(INPUT_PULSE, HIGH);
    delayMicroseconds(9680); //9.68ms
    voltage = preVoltage * 5.0 / 1024.0;
    dustDensity = (voltage-0.3)/0.005;
    sumDustDensity += dustDensity;
    delay(10);
  }
  avgDustDensity =                   
  Serial.print("dustDensity : ");
  Serial.println(avgDustDensity);
  delay(1000);
}
```

avg. of total 30 measuring results

applying LOW for a input pulse

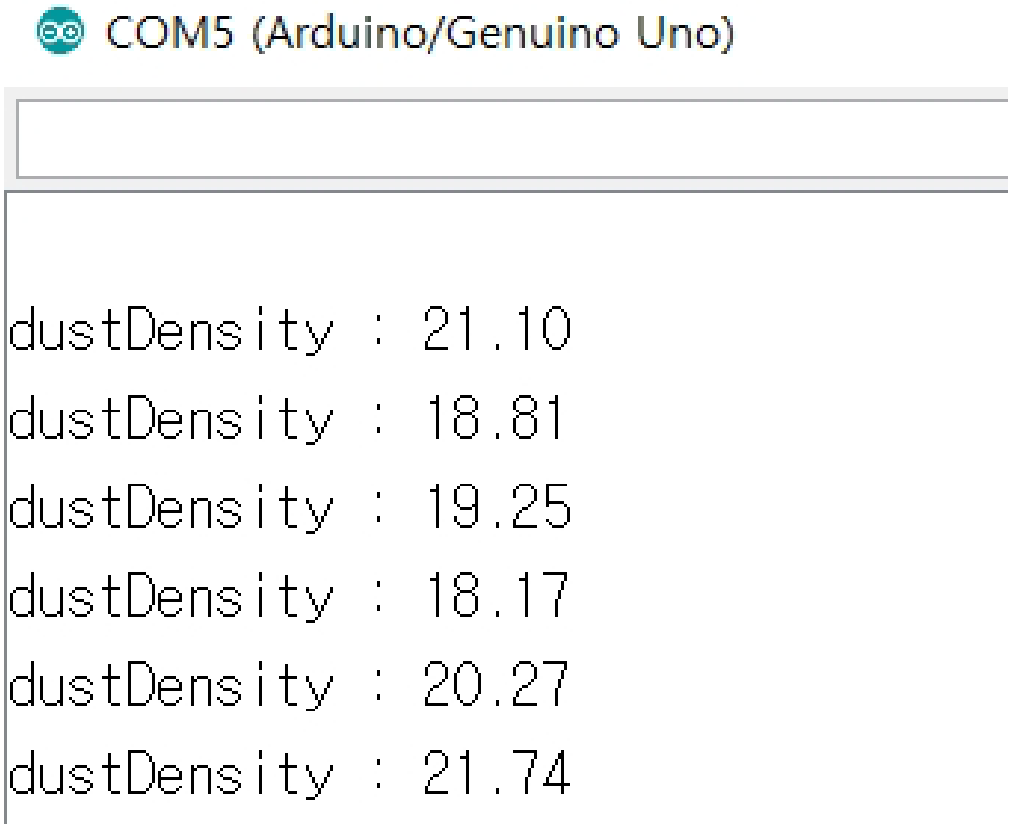
read output voltage after 0.28 ms

for a pulse width 0.32 ms

applying HIGH during a remaining cycle (10ms – 0.32ms)

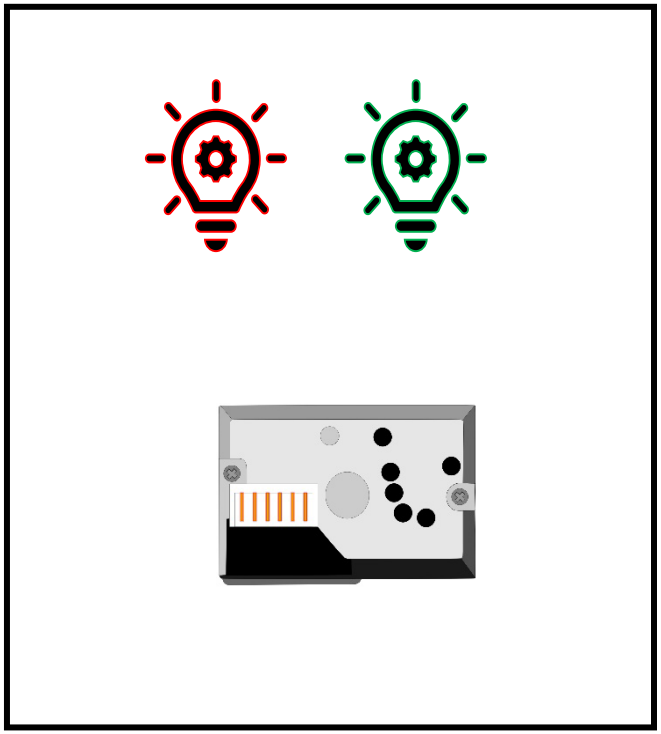
Calculate the fine dust concentration value based on the output voltage difference.

Check results



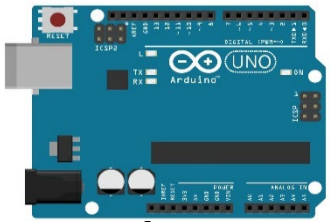
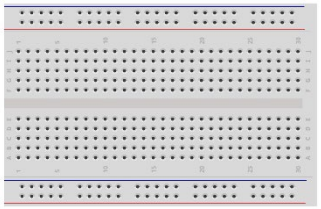
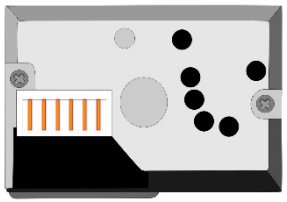
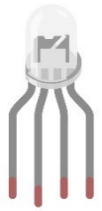
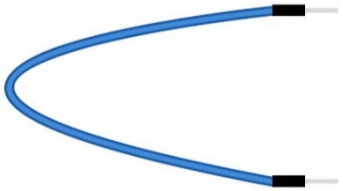
Assignment: Smart Dust Sensor

- Let's implement a smart dust sensor using a Tri-color LED.
- Requirements
  - Applying four levels of fine dust density (avg. of 30 results)
    - Good (~30) : Blue
    - Normal (31 ~ 50) : Green
    - Bad (51 ~ 100): Orange
    - Polluted (101~): Red
  - Show the result of dust density via serial communication
  - A block-type comments in the top of source code w/ "your student no., your name, writing date, etc."
- Results
  - (a source code file) sketch source code (*"sketchfilename.ino"*)
  - (a Arduino board capture file) a photo capture showing how you setup your circuit (max. 1GB file).



Assignment: Smart Dust Sensor

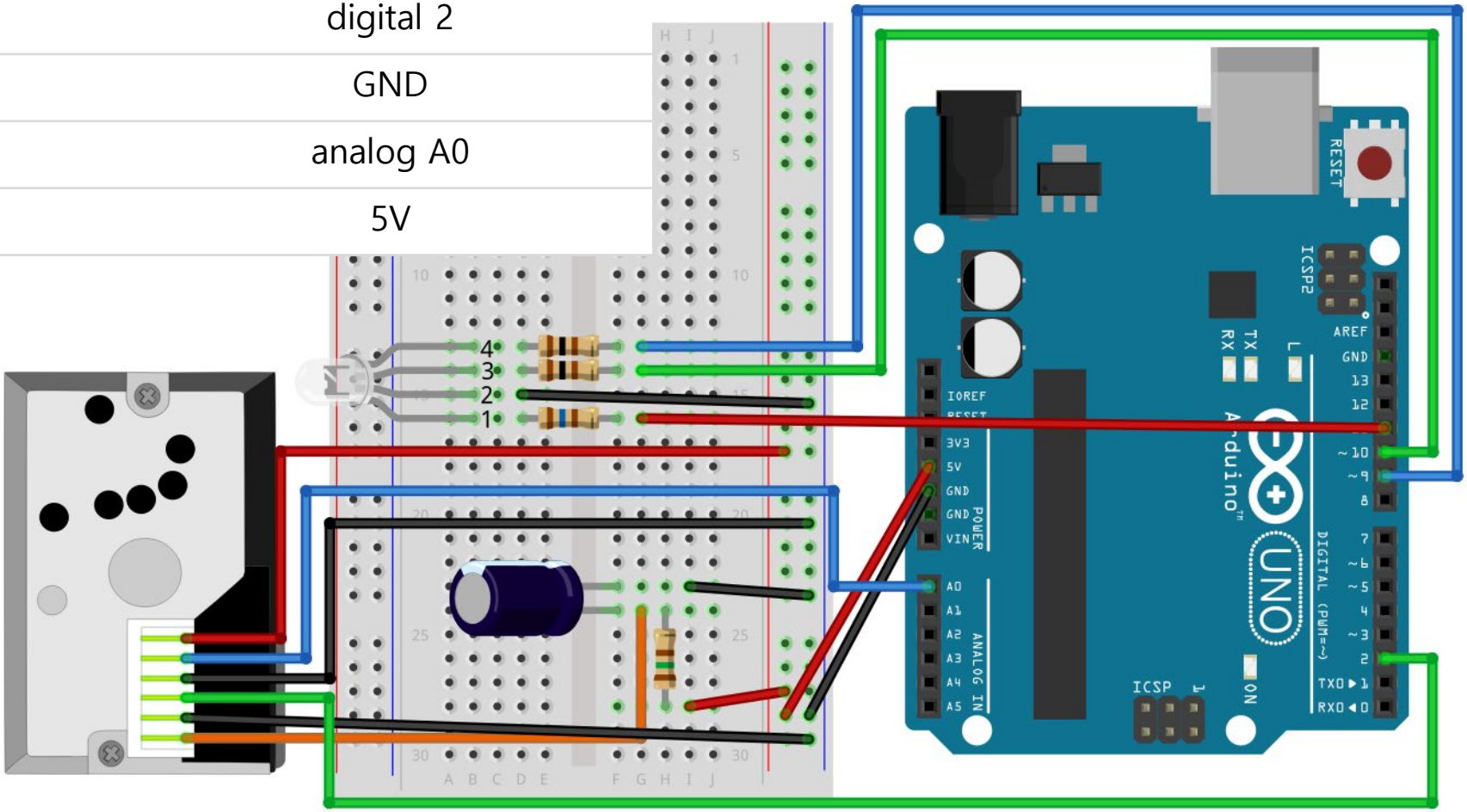
- Required H/W components

Arduino board (Uno)	Bread board (400 pins)	Dust sensor (GP2Y1010AU0F)	Tri-Color LED (Common cathode)
 x1	 x1	 x1	 x1
electrolytic capacitor (220uF)	Jumper cable Male-Male	Resistor (100Ω)	Resistor (150Ω)
 x1	 x14	 x1	 x1
Resistor (160Ω)			
 x1			

Circuit wiring setup

Dust sensor pins	Arduino board
1	electrolytic capacitor (+, long leg)
2	GND
3	digital 2
4	GND
5	analog A0
6	5V

Tri-color LED	Arduino board
Red (Pin 1)	PWM 11
GND (Pin 2)	GND
Green (Pin 3)	PWM 10
Blue (Pin 4)	PWM 9



Q&A