



아두이노 실습

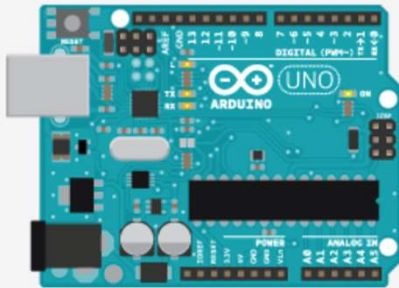


Chapter 06

조도센서 사용해보기



WHAT IS ARDUINO?



Step 1. 조도센서 값 읽어보기

Step 2. 조도의 밝기에 따라 LED가 제어되는 회로와
프로그램 만들기

조도센서 값 읽어보기

조도센서

- 조도의 밝기를 측정할 수 있는 전자 소자
- 포토 셀, 포토 레지스터, CDS등으로 불린다.
- 핀 극성 없음
- 정격
 - 정격 전압: 4 ~ 6V
 - 정격 전류: 상관없음

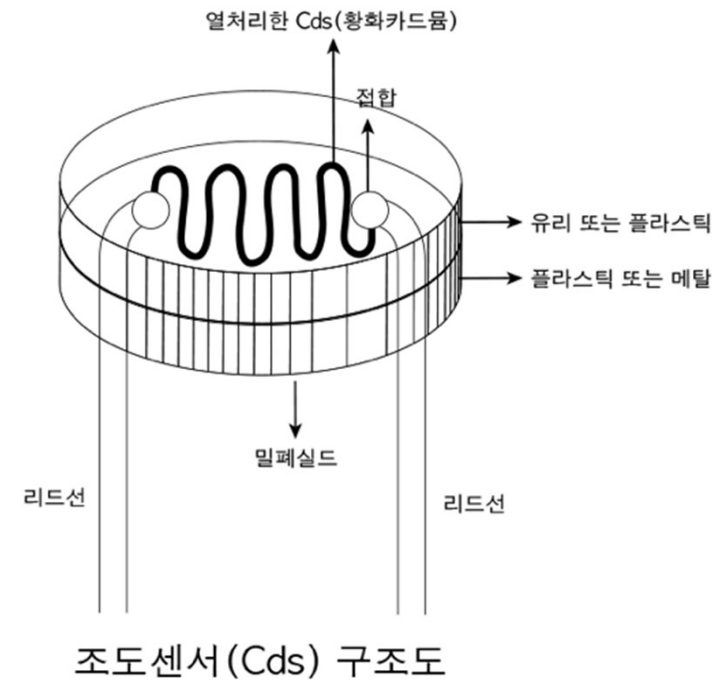


조도센서 값 읽어보기

조도센서의 원리

- 조도의 밝기에 따라 저항 값이 바뀐다.

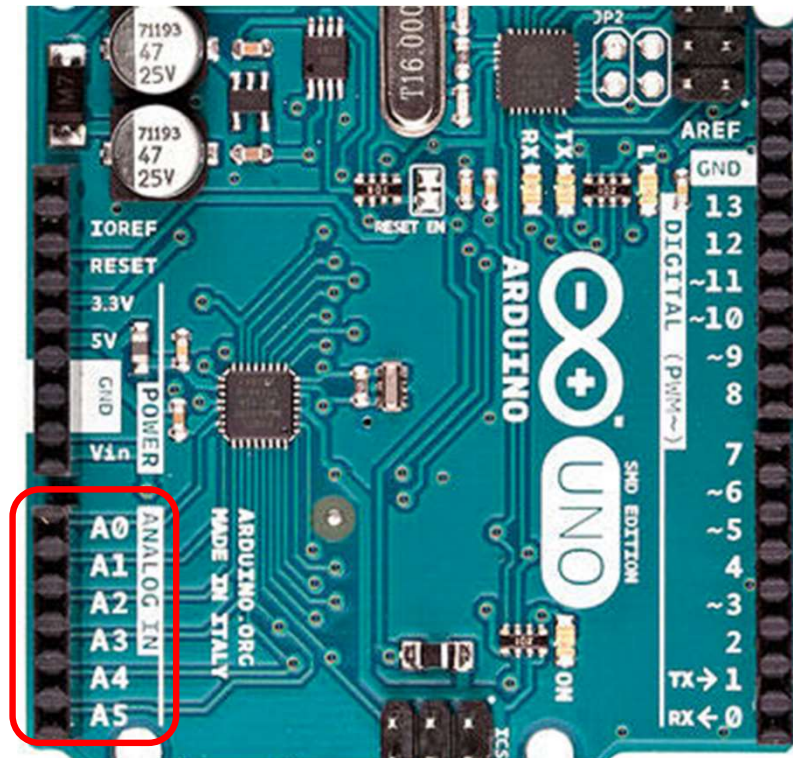
밝기(Lux)	저항(Ω)
10,000	100
1,000	300
100	1,500
10	10,000
1	70,000
0.1	600,000



조도센서 값 읽어보기

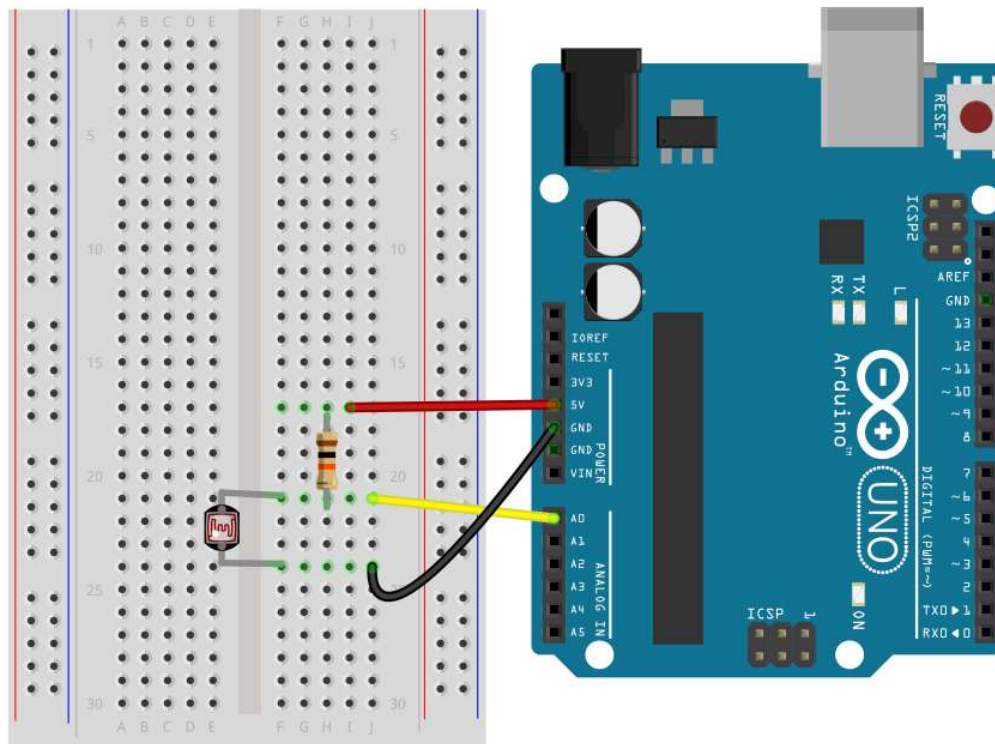
아두이노 ADC 핀

- A표시 핀
 - Uno보드의 경우 6개
(A0 ~ A5 핀)



조도센서 값 읽어보기

조도센서 회로 구현



조도센서 값 읽어보기

analogRead

- 아두이노 ADC 핀의 결과 확인
- C언어 문법
 - `int analogRead(int pin);`
 - 반환값 있음, 인자 1개
- 사용방법
 - pin: 아두이노 ADC 핀 번호
 - 반환값 : 해당 핀의 ADC결과
 - 해상도 1024 (0 ~ 1023 범위)
 - 예) `int state = analogRead(A0);` // A0 핀의 ADC 결과를 state 변수에 저장

조도센서 값 읽어보기

조도센서 소스코딩

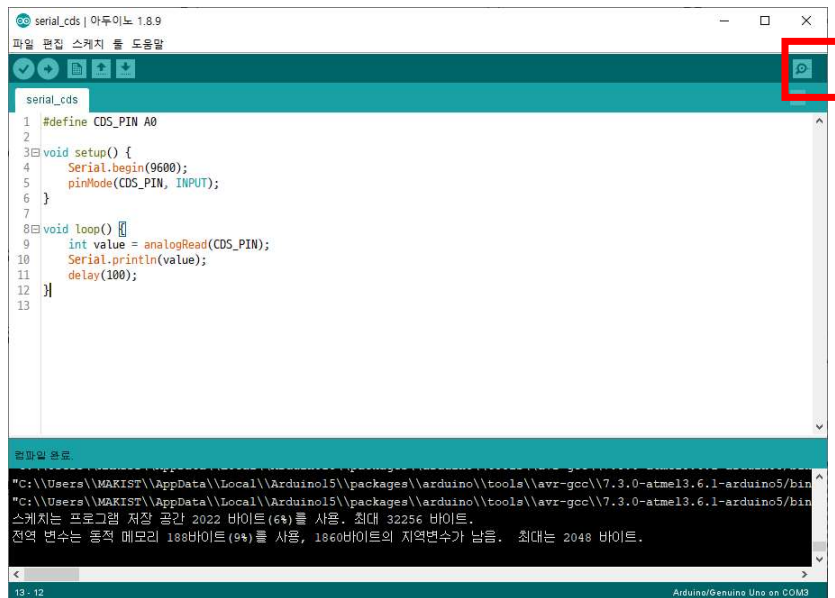
```
#define CDS_PIN A0

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

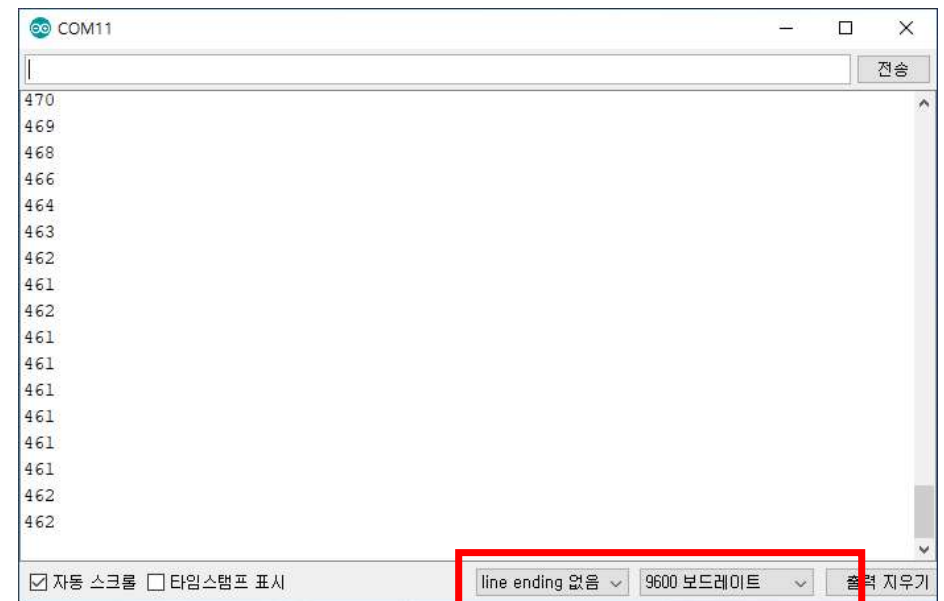
void loop() {
  int value = analogRead(CDS_PIN);
  Serial.println(value);
  delay(100);
}
```


조도센서 값 읽어보기

조도센서 값 확인해보기



우측 상단의 돋보기 모양을 클릭



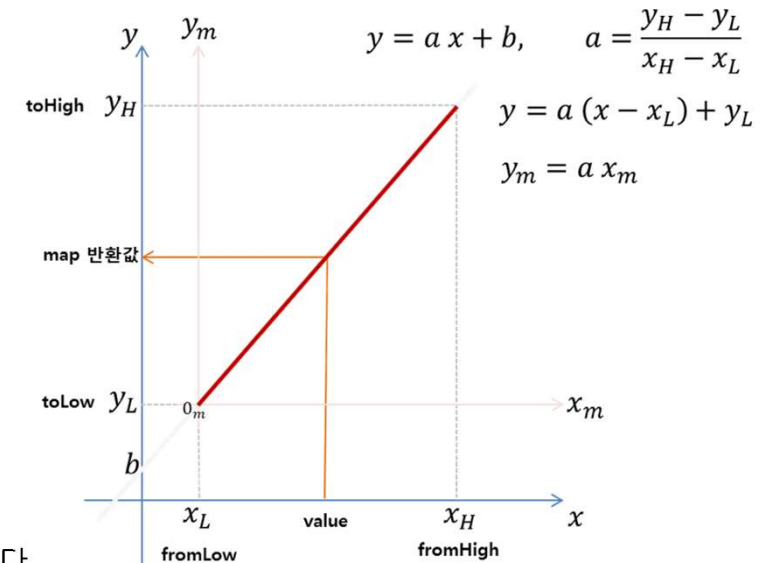
그림과 같이 설정해주세요.

조도의 밝기에 따라 LED가 제어되는 회로와 프로그램 만들기

map

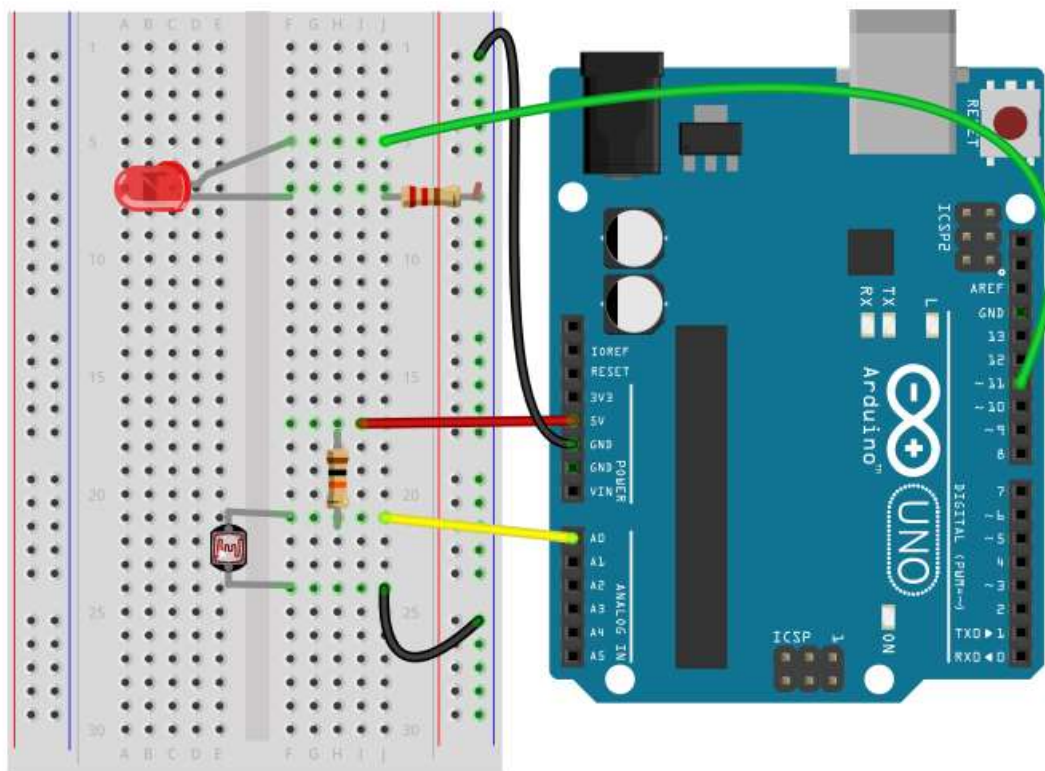
- 서로 다른 범위 체계 값을 손쉽게 변환
- C언어 문법
 - `int map(int value, int fromLow, int fromHigh, int toLow, int toHigh);`
 - 반환값 있음, 인자 1개
- 사용방법
 - value: 변환할 값
 - fromLow, fromHigh : 입력된 값의 범위
 - toLow, toHigh: 출력될 값의 범위
 - 반환값: 변환된 값
 - 예) `int result = map(a, 0, 1023, 0, 255);`

// a변수의 값을 0~1023범위 체계에서 0~255범위 체계로 바꾼다.



조도의 밝기에 따라 LED가 제어되는 회로와 프로그램 만들기

조도센서 회로구현



조도의 밝기에 따라 LED가 제어되는 회로와 프로그램 만들기

조도센서 소스코딩

```
#define LED_PIN    11
#define CDS_PIN    A0

void setup()
{
  pinMode(LED_PIN, OUTPUT);
  pinMode(CDS_PIN, INPUT);
}

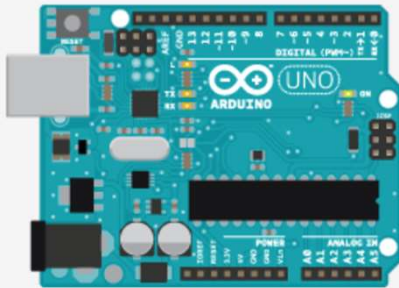
void loop()
{
  int value = analogRead(CDS_PIN);
  int output = map(value, 0, 1023, 0, 255);
  analogWrite(LED_PIN, output);
}
```

Chapter 07

적외선센서 사용해보기



WHAT IS ARDUINO?



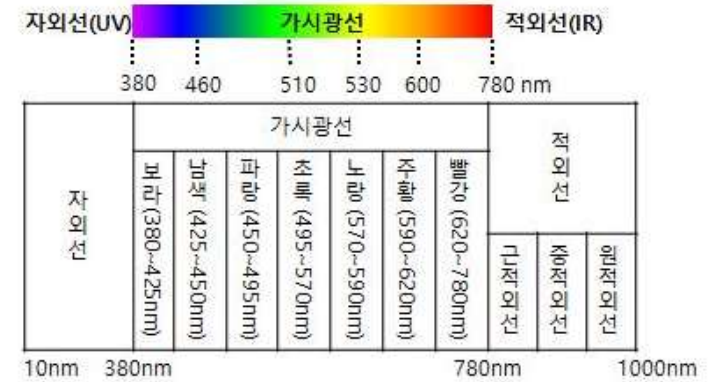
Step 1. 적외선 값 읽어보기 - digital

Step 2. 적외선 값 읽어보기 - analog

적외선 값 읽어보기 - digital

적외선 알아보기

- 빛의 스펙트럼에서 가시광선의 적색부 바깥쪽을 나타내는 광선
- 리모콘, 라인트레이서, 물체 감지 등에 사용되는 적외선은 근적외선을 사용 함
- 장점
 - 빛을 매체로 사용하기 때문에 전파규제가 없다
 - 전파에 비해 넓은 대역폭을 획득할 수 있다
- 단점
 - 통신거리가 짧고 양쪽 송.수신기가 마주보고 있어야함
 - 자연광, 인공광의 영향을 받음



적외선 값 읽어보기 - digital

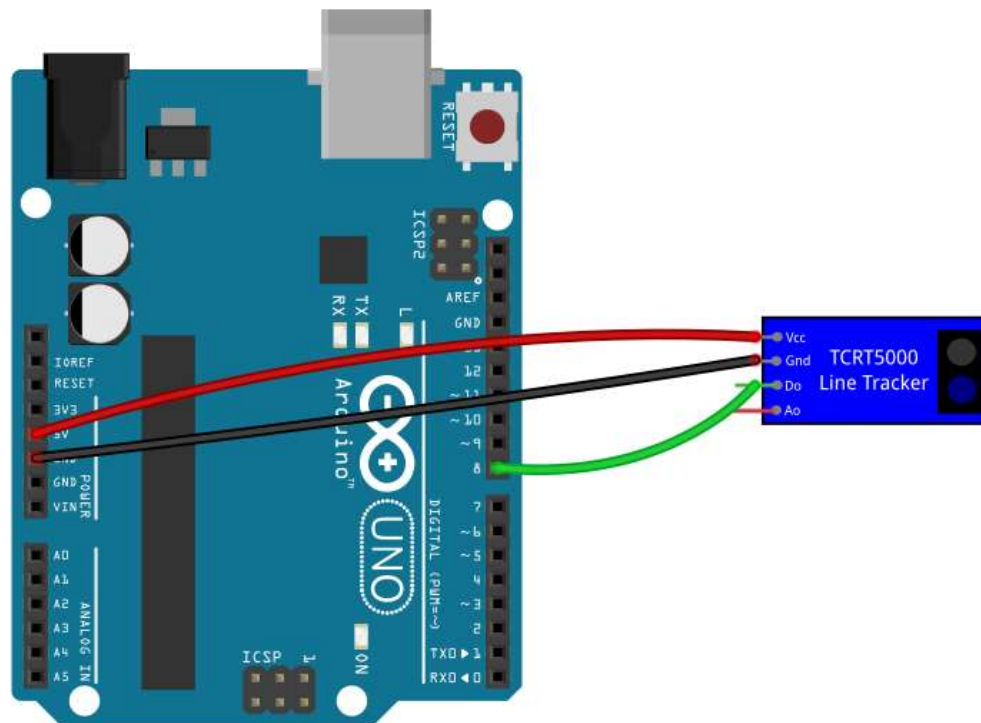
적외선센서 알아보기(TCRT5000)

- 발광부에서 나온 적외선이 물체에 반사되어 수광부에 들어오는 양에 따라서 전압이 변함.
- TCRT5000 적외선 센서는 아날로그와 디지털 값 확인 할 수 있음.
 - 디지털 - 기준 전압 보다 높을경우 HIGH, 낮을경우 LOW가 출력되며, 기준 전압은 센서의 가변저항을 통해 바꿀수 있음
 - 아날로그 - 전압의 값을 아날로그 값으로 확인할수 있음
 - 적외선은 검정색에서 흡수가 잘되고, 흰색에서 반사가 잘되기 때문에 밝을수록 HIGH에 가까운 값이 출력됨



적외선 값 읽어보기 - digital

적외선 회로구현



적외선 값 읽어보기 - digital

적외선 소스코딩

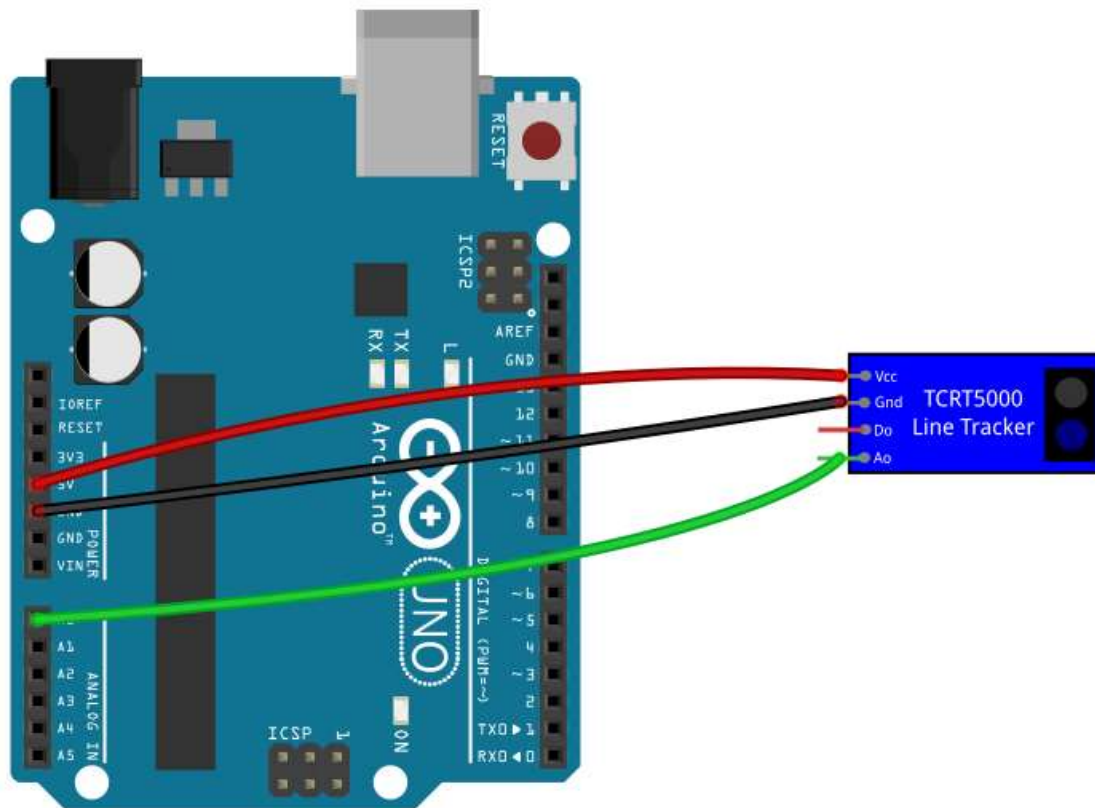
```
#define IR_PIN 8

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

void loop() {
  int value = digitalRead(IR_PIN);
  Serial.println(value);
  delay(100);
}
```

적외선 값 읽어보기 - analog

적외선 회로구현



적외선 값 읽어보기 - analog

적외선 소스코딩

```
#define IR_PIN A0

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

void loop() {
  int value = analogRead(IR_PIN);
  Serial.println(value);
  delay(100);
}
```

적외선 값 읽어보기

실습 - 1 디지털 값을 활용하여 LED on/off 제어하기

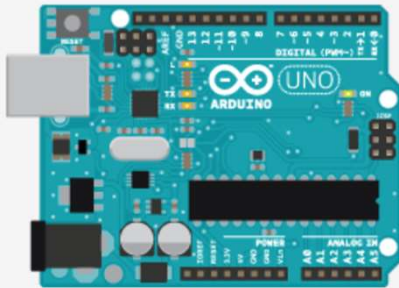
실습 - 2 아날로그 값을 활용하여 LED 밝기 제어하기

Chapter 08

LCD디스플레이 사용해보기



WHAT IS ARDUINO?



Step 1. LCD디스플레이에 글자 출력하기

LCD 디스플레이에 글자 출력하기

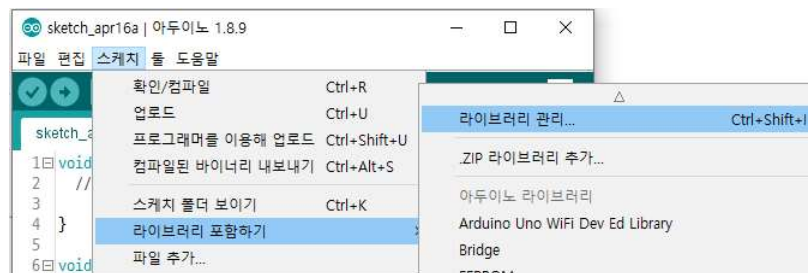
LCD 디스플레이

- 글자나 숫자를 표현 할수 있는 표시 장치
- 저전압, 저전력으로 작동하지만 시야각이 제한점이고 선명함이 떨어짐.
- 손목시계, TV 등 다양하게 사용됨

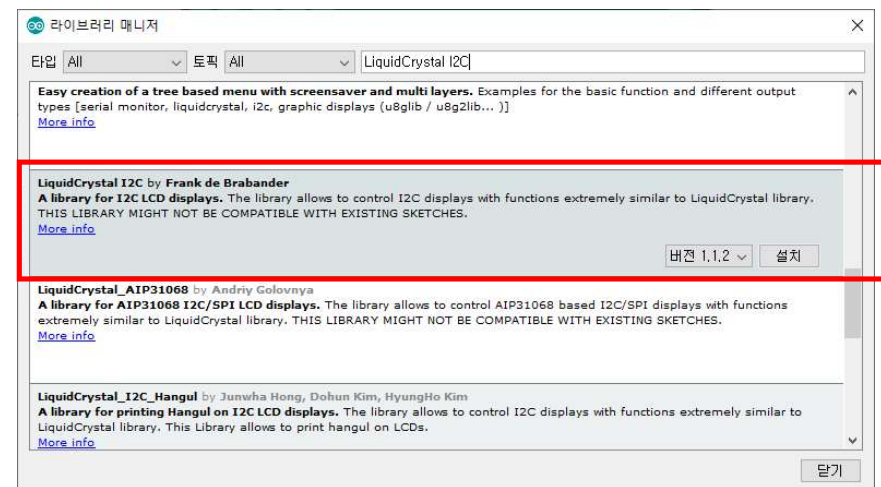


LCD디스플레이에 글자 출력하기

라이브러리 설치하기



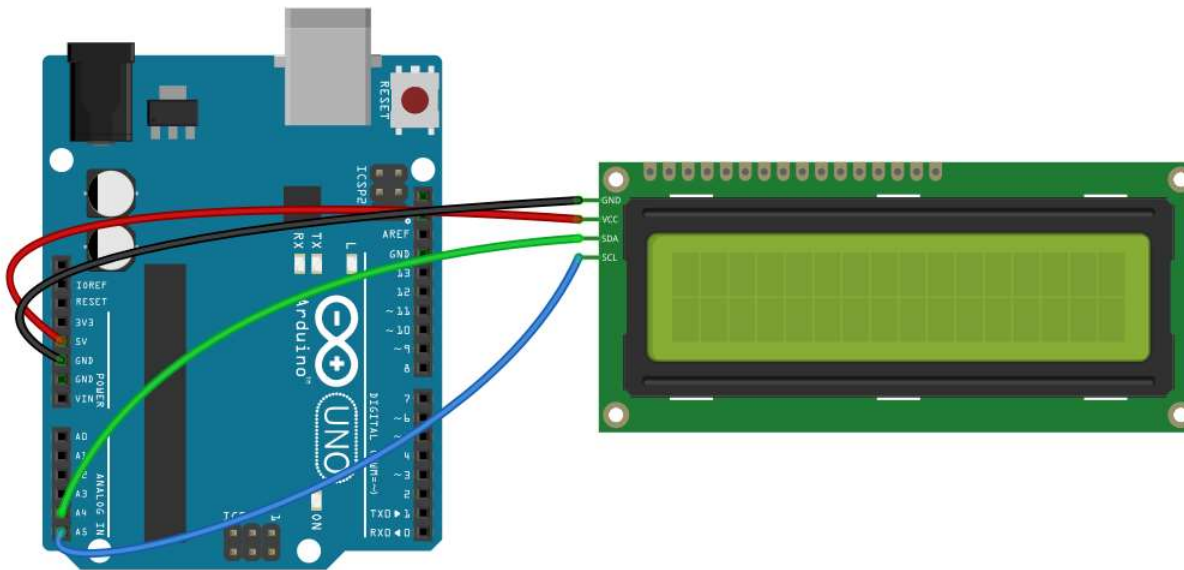
스케치 - 라이브러리 포함하기 라이브러리 설치



LiquidCrystal I2C 를 검색하여 라이브러리를 설치해줍니다.

LCD디스플레이에 글자 출력하기

I2C LCD 디스플레이 회로구현



LCD디스플레이에 글자 출력하기

I2C LCD 디스플레이 소스코딩

```
#include <Wire.h>                // i2C 통신을 위한 라이브러리
#include <LiquidCrystal_I2C.h>    // LCD I2C용 라이브러리
LiquidCrystal_I2C lcd(0x27,16,2); // 0x3F or 0x27를 선택하여 주세요.
                                   //작동이 되지 않는 경우 0x3F로 바꾸어주세요.

void setup()
{
    lcd.init();
    lcd.backlight();
    lcd.print("Hello, world!");
}

void loop()
{
}
```

LCD디스플레이에 글자 출력하기

I2C LCD 디스플레이 소스코딩

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,16,2);    // 0x3F or 0x27를 선택하여 주세요.
                                      //작동이 되지 않는 경우 0x3F로 바꾸어주세요.

void setup()
{
    lcd.init();
    lcd.backlight();
    lcd.setCursor(0, 0); // 1번째 줄 1번째 행부터 입력
    lcd.print("Holle");
    lcd.setCursor(0, 1); // 2번째 줄 1번째 행부터 입력
    lcd.print("world!");
}

void loop()
{
}
```

LCD디스플레이에 글자 출력하기

I2C LCD 디스플레이 소스코딩

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,16,2);    // 0x3F or 0x27를 선택하여 주세요.
                                      //작동이 되지 않는 경우 0x3F로 바꾸어주세요.

void setup()
{
    lcd.init();
    lcd.backlight();
    lcd.setCursor(1, 0); // 1번째 줄 2번째 행부터 입력
    lcd.print("1");
    lcd.setCursor(3, 1); // 2번째 줄 4번째 행부터 입력
    lcd.print("2");
}

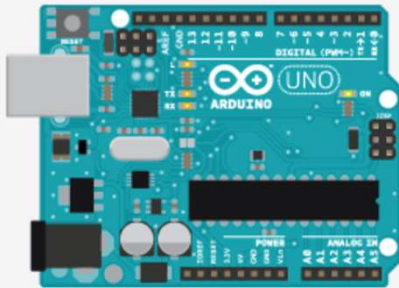
void loop()
{
}
```

Chapter 09

온습도센서 사용해보기



WHAT IS ARDUINO?



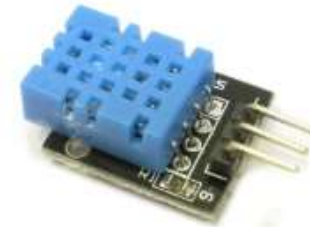
Step 1. 온도 습도 값 읽어오기

Step 2. LCD디스플레이에 온도 습도 출력하기

온도 습도 값 읽어보기

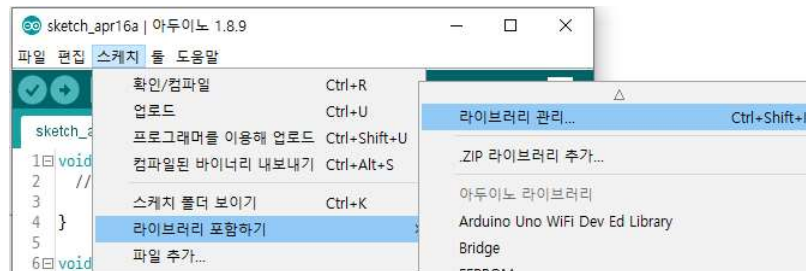
온습도센서 알아보기

- 정전식 습도센서와 써미스터를 내장되어 온도와 습도를 측정하는 센서
 - 써미스터 - 온도에 따라 저항값이 변하는 소자
 - 정전식 습도센서 - 습도에 따라 저항값이 변하는 소자
- 온도 : 0 ~ 50℃ 습도 : 20~90%(상대습도)의 측정범위를 가짐.

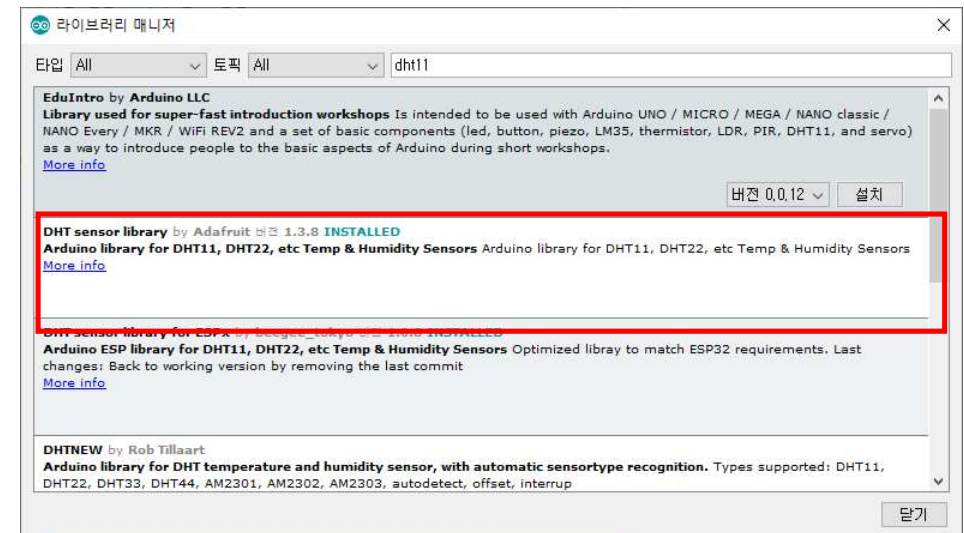


온도 습도 값 읽어보기

라이브러리 설치하기



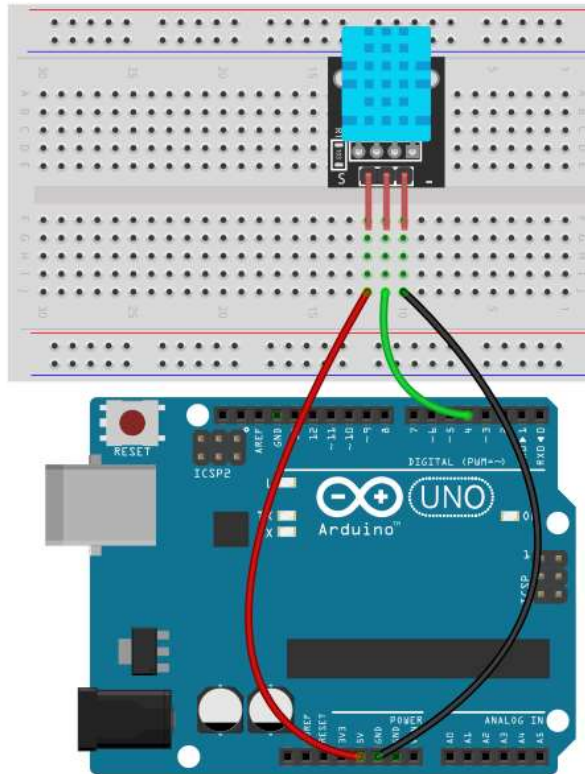
스케치 - 라이브러리 포함하기 라이브러리 설치



dht11 를 검색하여 라이브러리를 설치해줍니다.

온도 습도 값 읽어보기

온습도센서 회로도 연결



온도 습도 값 읽어보기

온습도센서 소스코딩

```
#include <DHT.h>

#define DHTPIN 4
#define DHTTYPE DHT11 // DHT 11

DHT dht(DHTPIN, DHTTYPE);

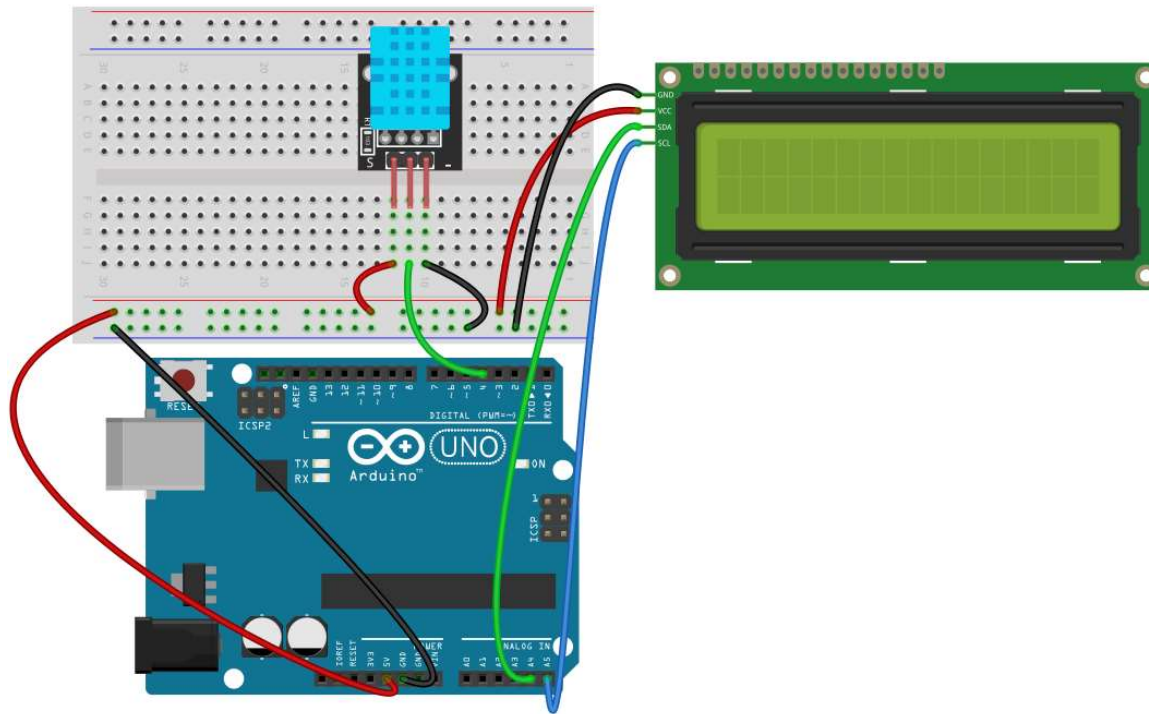
void setup() {
    Serial.begin(9600);
    dht.begin();
}

void loop() {
    float h = dht.readHumidity();
    float t = dht.readTemperature();
```

```
    Serial.print("Humidity: ");
    Serial.print(h);
    Serial.print("    ");
    Serial.print("Temperature:");
    Serial.print(t);
    Serial.println("°C ");
    delay(1000);
}
```


LCD디스플레이에 온도 습도 출력하기

회로도 연결



LCD디스플레이에 온도 습도 출력하기

온습도센서 소스코딩

```
#include <DHT.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27, 16, 2);

#define DHTPIN 4
#define DHTTYPE DHT11 // DHT 11

DHT dht(DHTPIN, DHTTYPE);

void setup() {
  Serial.begin(9600);
  dht.begin();
  lcd.init();
  lcd.backlight();
}

void loop() {
  float h = dht.readHumidity();
  float t = dht.readTemperature();
```

```
  lcd.setCursor(0, 0);
  lcd.print("Humidity: ");
  lcd.print(h);
  lcd.print("%");
  lcd.setCursor(0, 1);

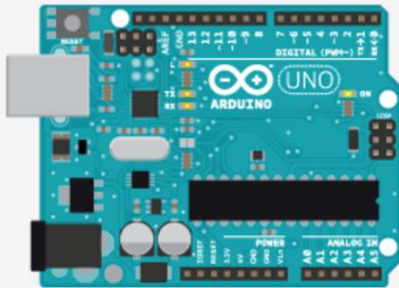
  lcd.print("Temperature:");
  lcd.print(t);
  lcd.println("°C ");
  delay(1000);
```

Chapter 00

시리얼통신 알아보기



WHAT IS ARDUINO?



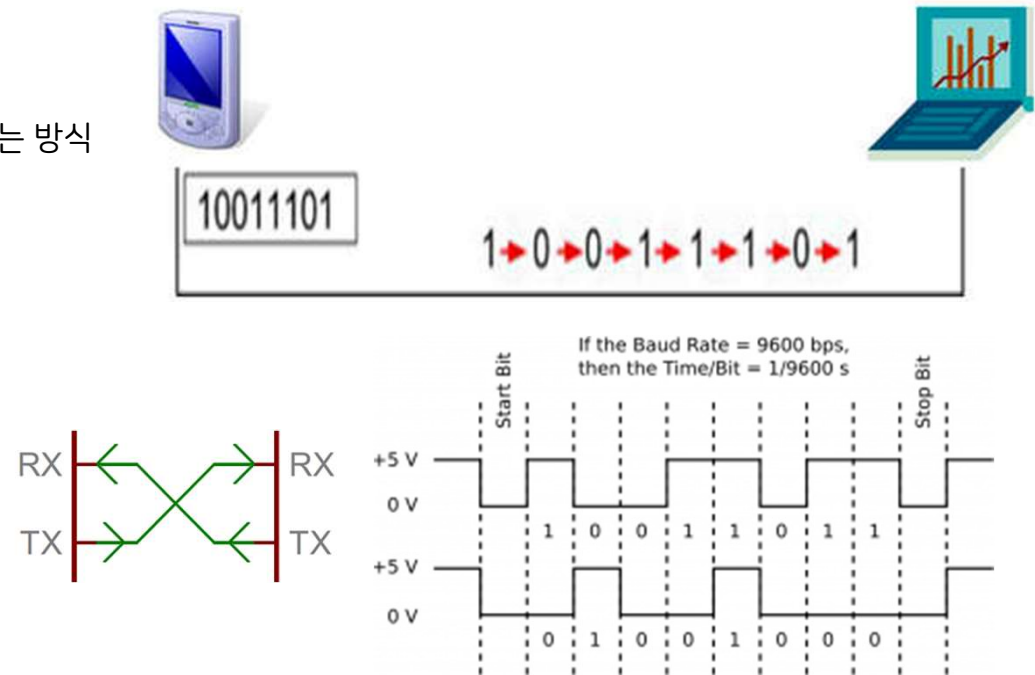
Step 1. 시리얼통신

Step 2. 시리얼통신으로 조도센서 값 읽어보기

시리얼통신

시리얼 통신

- 선 하나로 데이터를 보내는 방식
 - 병렬(Parallel) 통신: bit당 한 선으로 데이터를 보내는 방식
- 전송(Tx)과 수신(Rx)
 - Full Duplex 방식: 전송선과 수신선이 따로 존재
 - Half Duplex 방식: 전송선과 수신선을 공유
 - 전송과 수신은 서로 꼬인 상태로 연결되어야 한다.
- 보레이트(Baudrate)
 - Bit 정보를 추출하기 위한 약속된 타이밍



시리얼통신

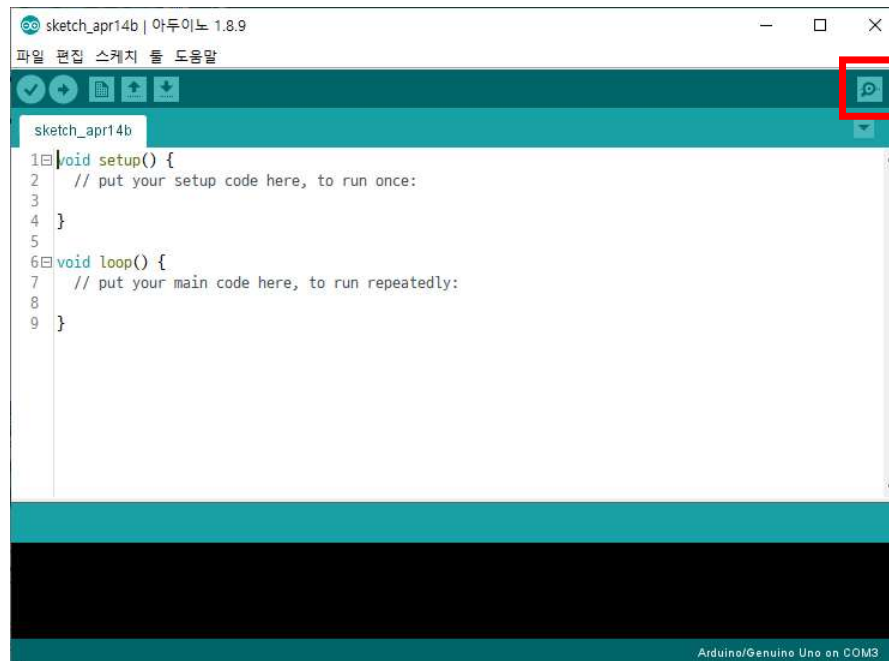
아두이노 시리얼 통신

- 하드웨어 시리얼
 - UART라 부르는 시리얼 통신 전용 장치를 사용
 - 우노 보드의 경우 1개 존재
- 소프트웨어 시리얼
 - 아두이노 라이브러리를 이용해서 디지털 핀을 사용
 - TX, RX선을 지정하여 사용 가능
 - 여러 선을 연결해도 동시에 통신하지 못함

시리얼통신

아두이노 시리얼 모니터

- PC에서 손쉽게 아두이노 보드를 제어할 수 있도록 만든 프로그램
- 예전의 DOS환경처럼 아두이노 보드를 PC에서 다룰 수 있다.



시리얼통신

Serial Class

- 아두이노에서 시리얼 통신을 위해 사용되는 Class
- 선언없이 바로 사용
- 함수
 - begin
 - print
 - println
 - available
 - read
 - write
 - readString

시리얼통신

Serial.begin()

- 시리얼 통신을 위해 보레이트 설정
- C언어 문법
 - `void begin(int speed);`
 - 반환값 없음, 인자 1개
- 사용방법
 - speed: 보레이트 값
 - 예) `Serial.begin(9600);` // 9600bps로 보레이트 설정

시리얼통신

보레이트(Baudrate)

- 시리얼 통신에서 사용하는 통신 속도 (단위: bps)
- 양쪽이 동일한 보레이트를 사용해야 한다.
- 자주 사용되는 표준 보레이트
 - 9600 bps
 - 14400 bps
 - 38400 bps
 - 57600 bps

시리얼통신

Serial.print / println

- 아두이노 보드에서 시리얼 모니터에 출력
(print와 println은 용법은 같지만 println의 경우 데이터 끝에 줄바꿈 문자(/n)를 추가하여 보낸다.)
- C언어 문법
 - `void print(value);`
 - 반환값 없음, 인자 1개
- 사용방법
 - value: 시리얼 모니터에 출력할 값 (정수, 소수, 문자열 가능)
 - 예)
`Serial.print(a);` // a변수의 값을 출력
`Serial.print(30);` // 30이라는 숫자 출력
`Serial.print("Hello Arduino");` // Hello Arduino 문자열 출력

시리얼통신

Serial.available()

- 시리얼 통신으로 받은 데이터 크기를 확인한다.
- C언어 문법
 - `int available();`
 - 반환값 있음, 인자 없음
- 사용방법
 - 반환: 받은 데이터 크기를 바이트 단위로 알려줌
 - 예) `int num = Serial.avilable(); // num변수에 받은 데이터 크기를 저장`

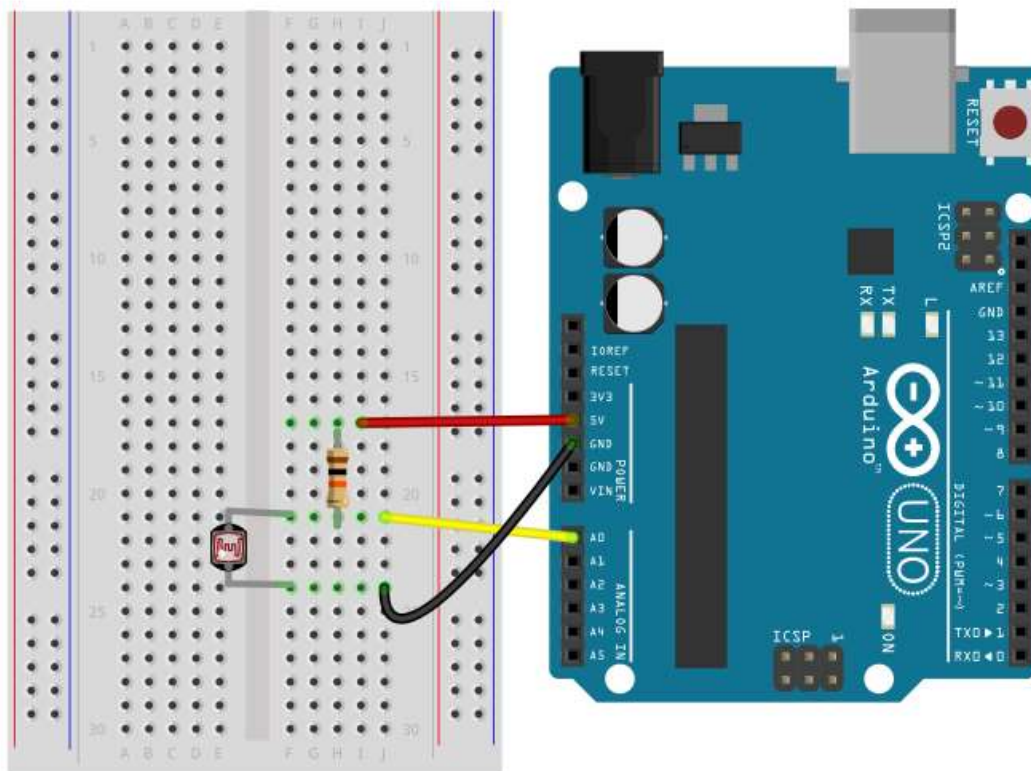
시리얼통신

Serial.read()

- 시리얼 통신으로 데이터 1 byte를 받는다.
- C언어 문법
 - `int read();`
 - 반환값 있음, 인자 없음
- 사용방법
 - 반환값: 받은 1byte 데이터
 - 예) `int data = Serial.read();` // 시리얼 통신으로 받은 1 byte 데이터를 data변수에 저장한다.

시리얼 통신으로 조도센서 값 읽어보기

조도센서 회로 구현



시리얼 통신으로 조도센서 값 읽어보기

조도센서 소스코딩

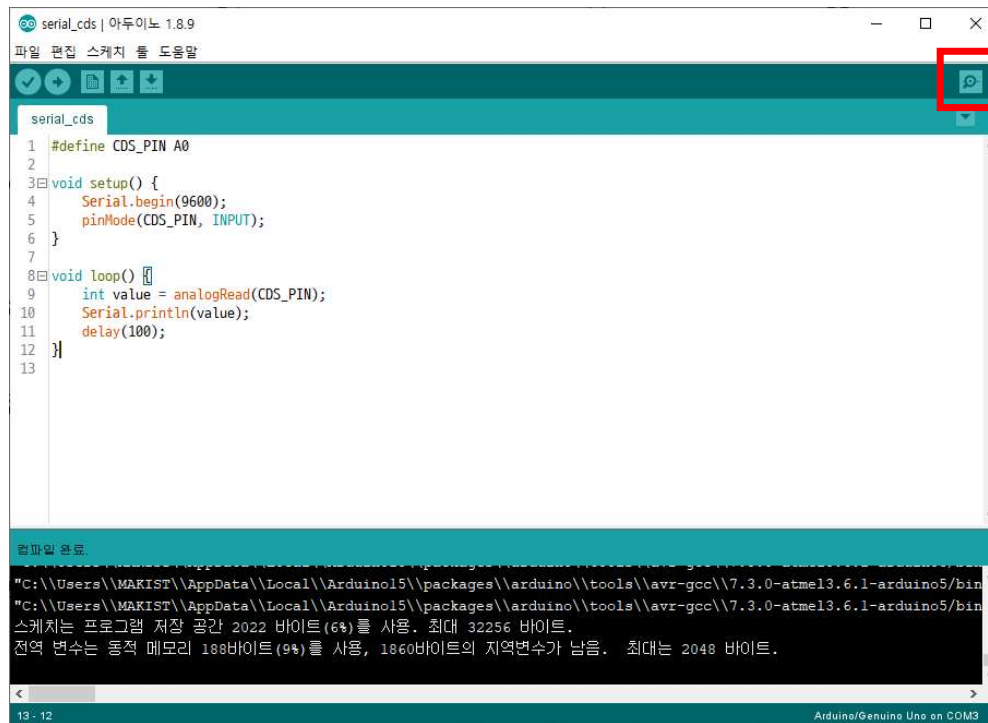
```
#define CDS_PIN A0

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

void loop() {
  int value = analogRead(CDS_PIN);
  Serial.println(value);
  delay(100);
}
```

시리얼 통신으로 조도센서 값 읽어보기

조도센서 값 확인해보기



우측 상단의 돋보기 모양을 클릭