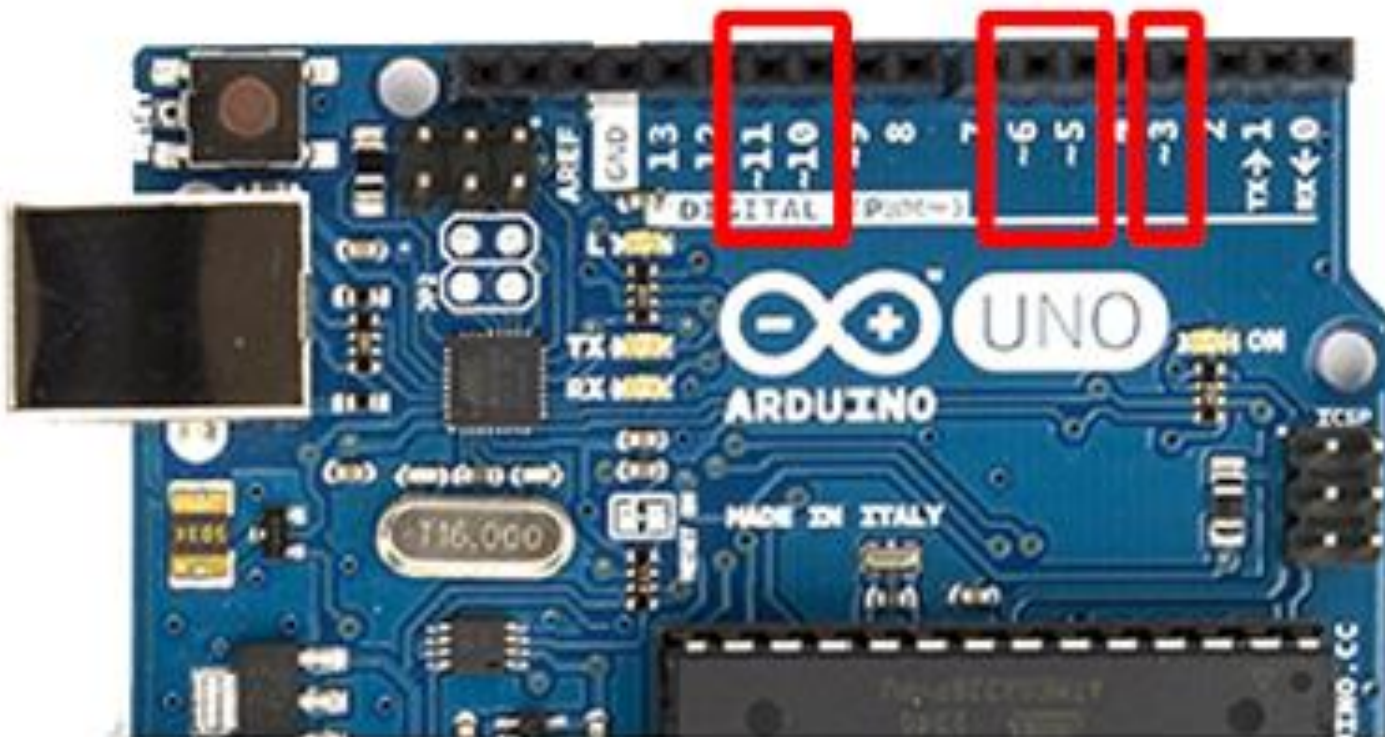


9주차 - 아날로그 입력과 출력

PWM 소개

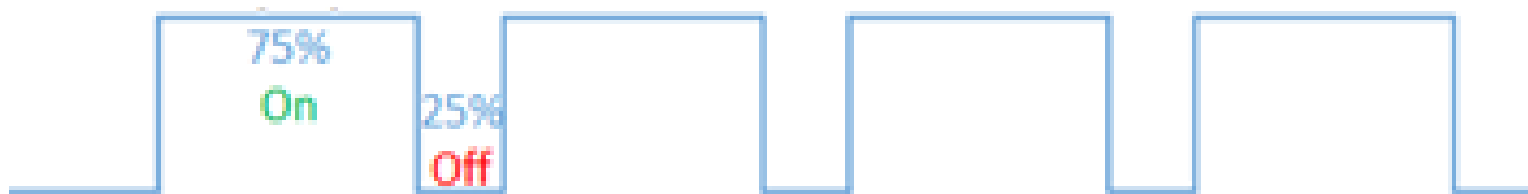
아두이노 PWM핀

- ❖ 아두이노 UNO는 6개의 PWM 출력 핀을 가지고 있음
- ❖ 보드에 ~표시가 된 핀에서만 PWM 기능 사용 가능
- ❖ 3,4,5,9,10,11핀이 해당한다



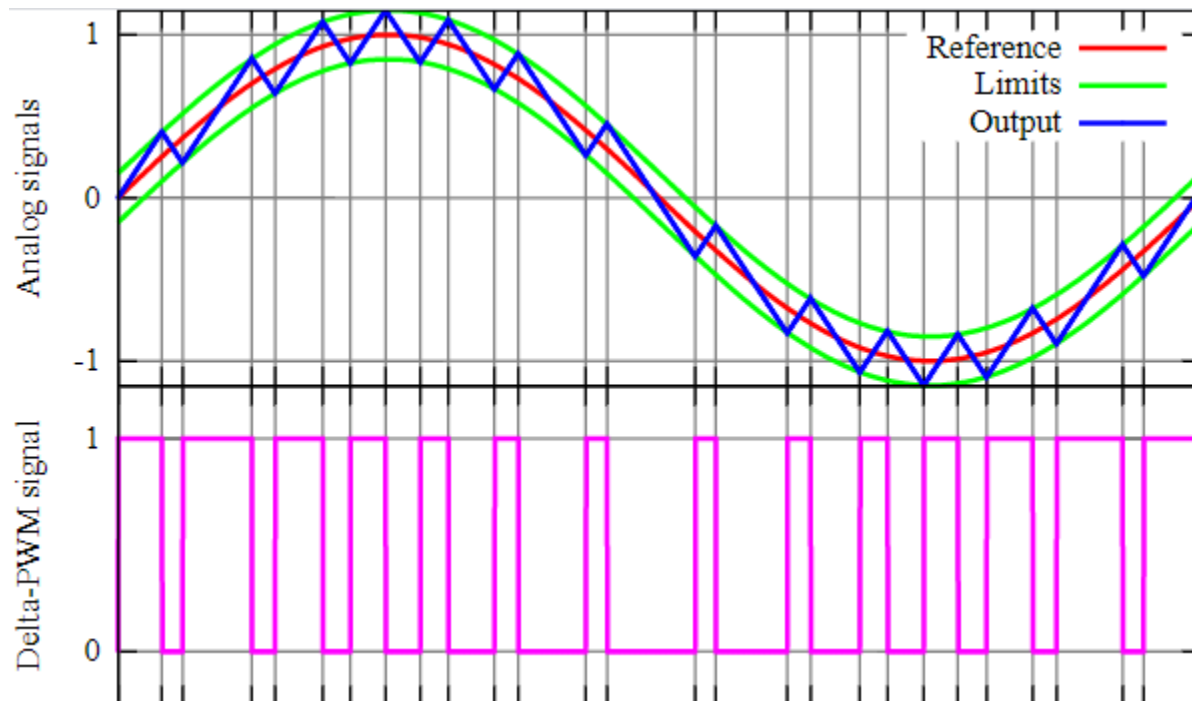
신호등 모듈 – PWM

- PWM이란
- Pulse Width Modulation 펄스 폭 변조.
- 아두이노는 출력을 0또는 1로 밖에 주지 못함
- 파형을 조절하여 전류 공급을 조절하는 방식
- MCU 마다 PWM 주파수가 다르다



왜 PWM을 사용하는가?

- 아날로그 회로에서는 전압을 조절하여 전류량 조절 가능
- 디지털 신호에는 0과 1밖에 크기가 존재하지 않음
- 0과 1을 주는 시간을 조절하여 전류량 조절



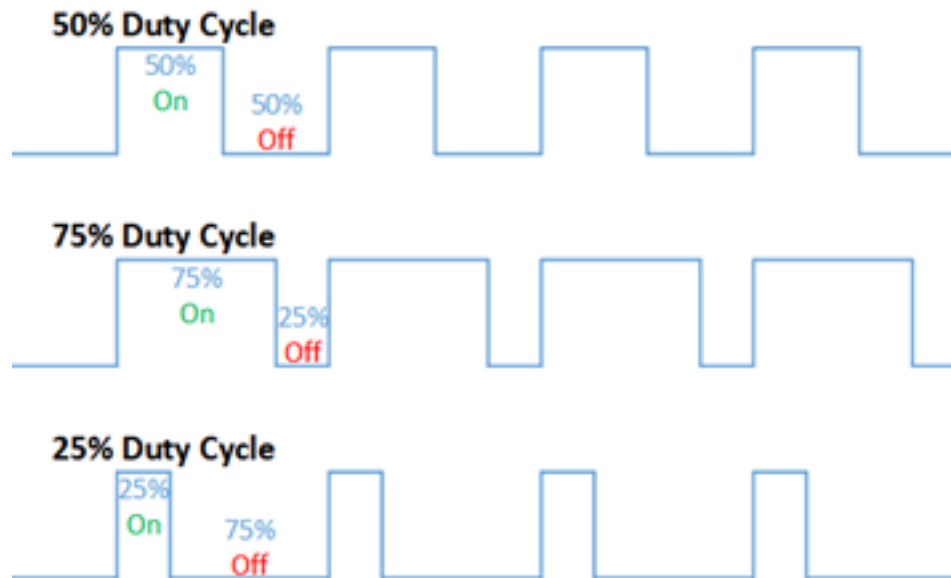
Period와 Duty Cycle

■ Period

- ❖ PWM 주파수중 한 클럭의 길이

■ DutyCycle

- ❖ Period 에서 HIGH를 줄 파형의 개수
- ❖ %로 표현한다. 예) duty cycle 25%



아두이노와 PWM

- analogWrite 함수를 이용해서 PWM 구현 가능
- analogWrite(핀, 값)을 이용해 PWM 폭 조절 가능
- 값의 범위는 0~255
- 9,10,11,3번 주파수
= 490Hz
- 5,6번 핀 주파수
= 980Hz

함수소개 – analogWrite

■ analogWrite(pin, value)

■ PWM파형을 쓰는 함수

❖ pin : 출력 할 출력 설정의 핀

❖ value : 듀티 사이클 : 0 ~ 255, int형

❖ ex) 11번 핀에 120의 pwm 을 준다면
analogWrite(11, 120);

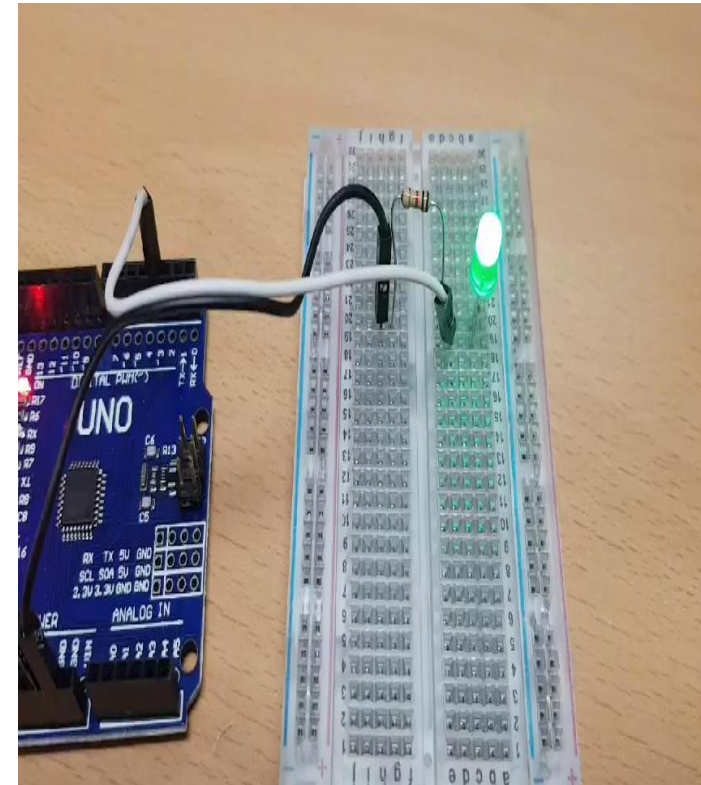
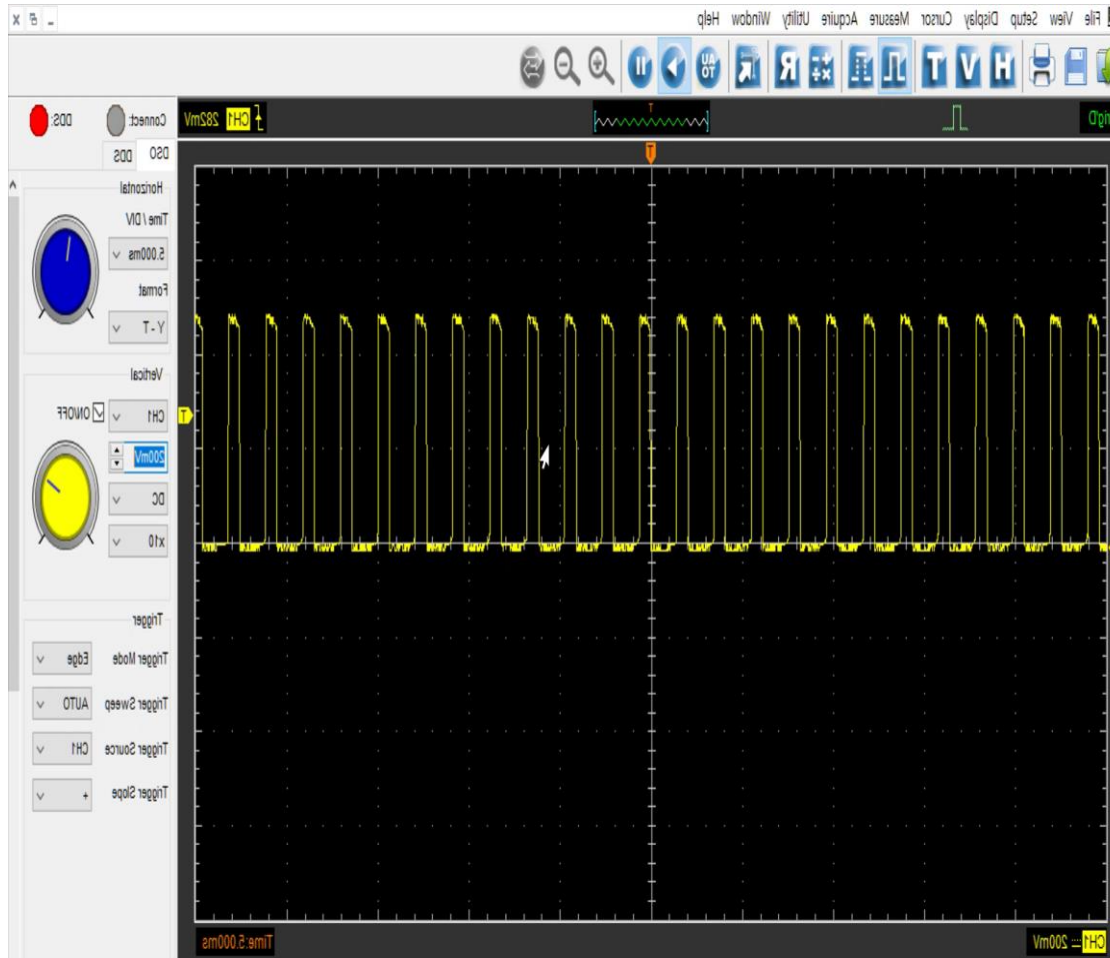
예시 – PWM을 이용한 LED 밝기 조절

```
void setup()
{
  pinMode(3, OUTPUT);
}

void loop()
{
  int lp;

  for(lp=0;lp<256;lp++)
  {
    analogWrite(3, lp);
  }
}
```

예시 - PWM을 이용한 LED 밝기 조절



실습

- 앞의 LED 밝기 제어 코드 동작에 맞게 회로를 결선하여 실습해 보십시오.

고찰

- LED 의 밝기 변화는 전압 뿐만 아니라 전류도 변화하는 것을 의미한다
- 모터의 속도 같은 것을 조절할 수 있다.
- PWM의 주파수는 왜 다를 까?

ADC 소개

ADC 소개

아두이노 아날로그 입력 기능(ADC)

- Analog Ports 를 사용해서 아날로그 전압 값을 읽는 기능
 - ❖ 아날로그 전압 값을 읽어 디지털 값으로 변경해 주는 하드웨어를 내장하고 있으며, 이 하드웨어를 ADC(Analog to Digital Convertor)라고 한다.
- 아두이노 우노 기준으로 0~5V의 전압이 입력 가능하며, 10bits의 해상도를 가지고 있다
 - ❖ 2의 10승을 의미하며, 0 ~1023 값으로 표현한다
 - ❖ 0~5V를 0~1023값으로 표현한다
 - ❖ 예) 1V \doteq 208.
- 변환시간은 약 100마이크로초(0.0001초) 정도 소요 된다

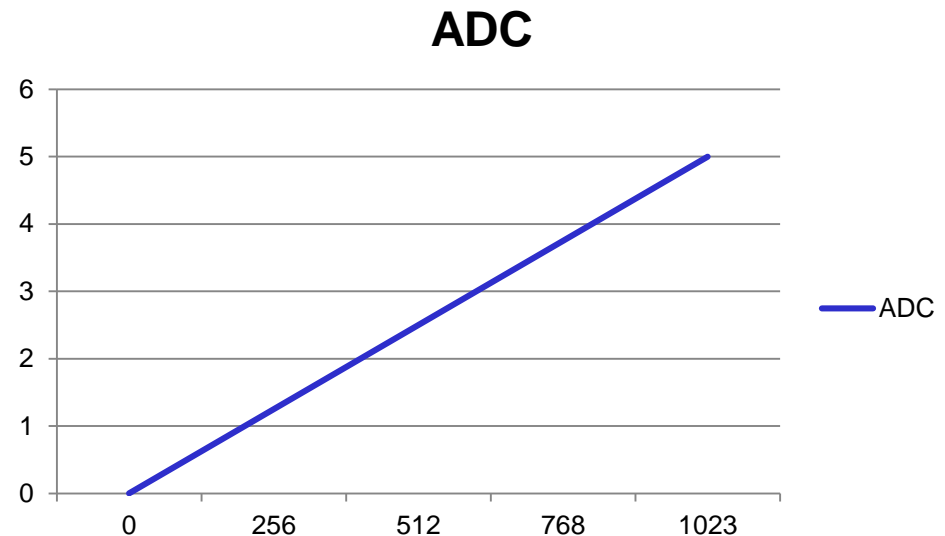
입력 전압 값 계산해보기

■ ADC 값 1은 몇 V일까?

- ❖ ADC 값과 V는 선형적 관계를 갖는다
- ❖ $0.00488 \approx 5V / 1023$

■ 그럼 768은 몇 V 일까?

- ❖ $3.75V = 768 * 0.00488$



함수소개 – analogRead

- analogRead(pin)
- 해당 핀에서 전압을 읽어 ADC로 변환된 데이터를 리턴해 준다
 - ❖ pin : 전압을 읽을 핀
 - ❖ return : int 형, 0 ~ 1023
 - ❖ ex) A0 핀에 값을 읽어 출력
 - value = analogWrite(A0);
 - Serial.println(value);

가변 저항 소개

■ 소개

- ❖ 값을 변경할 수 있는 저항을 의미한다.
- ❖ 노브라고 하는 손잡이 형태의 단자가 있으며 그것을 돌려 저항 값을 바꿀 수 있다.

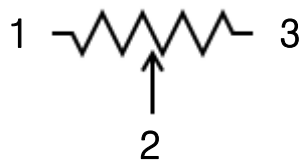
■ 오디오 볼륨이나 등의 밝기 등을 조절할 때 유용하다



가변 저항 구조

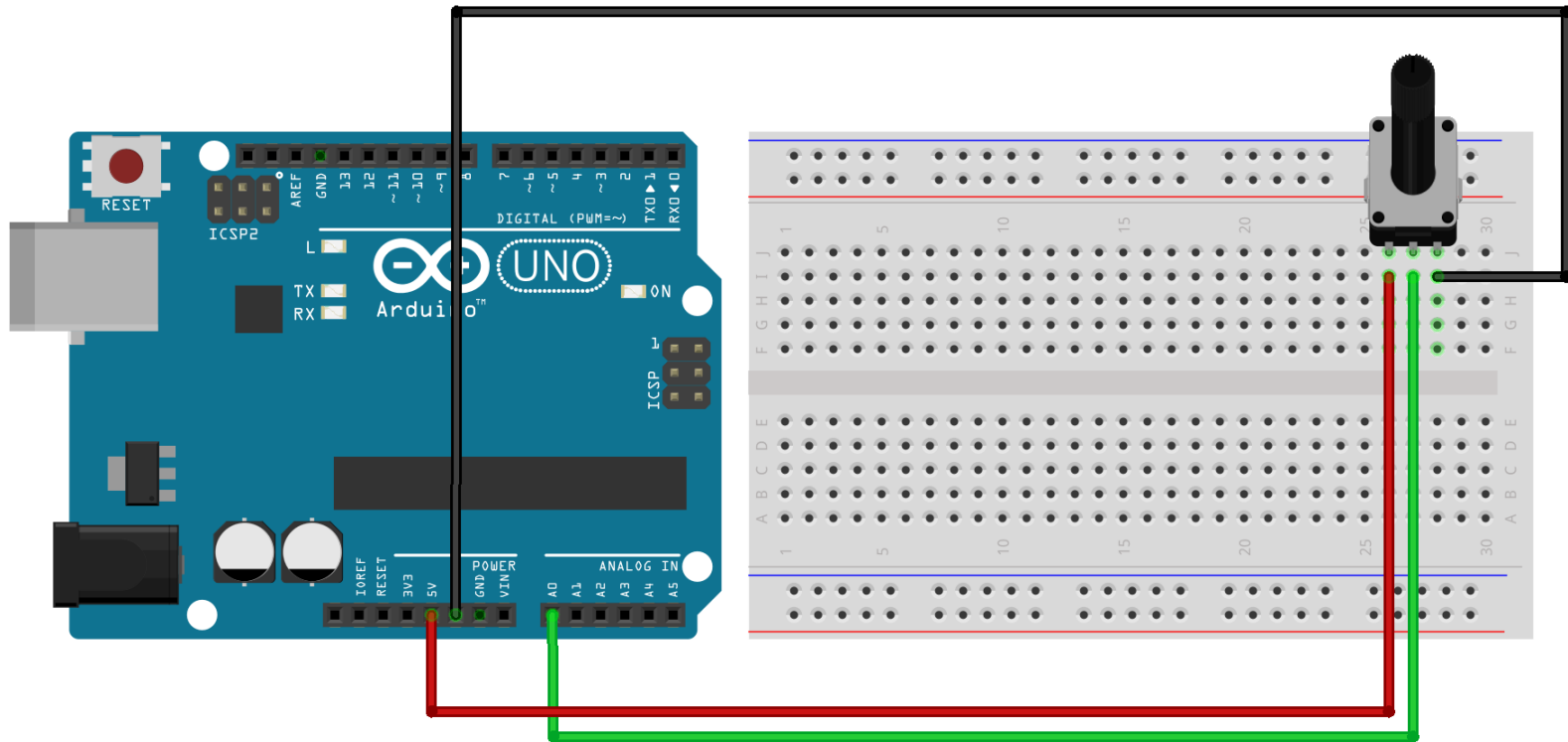
■ 가변 저항은 3개의 다리를 가지고 있으며 노브를 돌리면 양단의 저항 값이 변한다

- ❖ 1-3의 저항 값은 항상 일정하다.
- ❖ 노브를 돌리면 1-2 와 2-3의 저항 값이 변한다.
- ❖ $1-2 + 2-3$ 의 저항 값은 1-3과 항상 동일하다.
- ❖ 1-3이 가변 저항의 최대 값이고 저항 값이다.



결선도

- PWM을 이용한 ADC값 출력 예제에 맞는 결선도를 그려 제출하시오



예시 – 가변 저항 사용한 예시

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

void loop()
{
    int sensorValue = analogRead(A0);
    Serial.println(sensorValue);
}
```

유용한 함수 – map

- 입력 값 최대/최소 범위와 출력 값 최대/최소 범위를 지정하고 입력된 값에 대해서 비율로 계산된 값을 출력 해줌
- 예시) `calc_value = map(real_value, 30, 50, 0, 100)`
 - ❖ 30~50 사이의 값이 입력되며 비율적으로 계산하여 0~100의 값을 반환한다.

함수소개 – analogRead

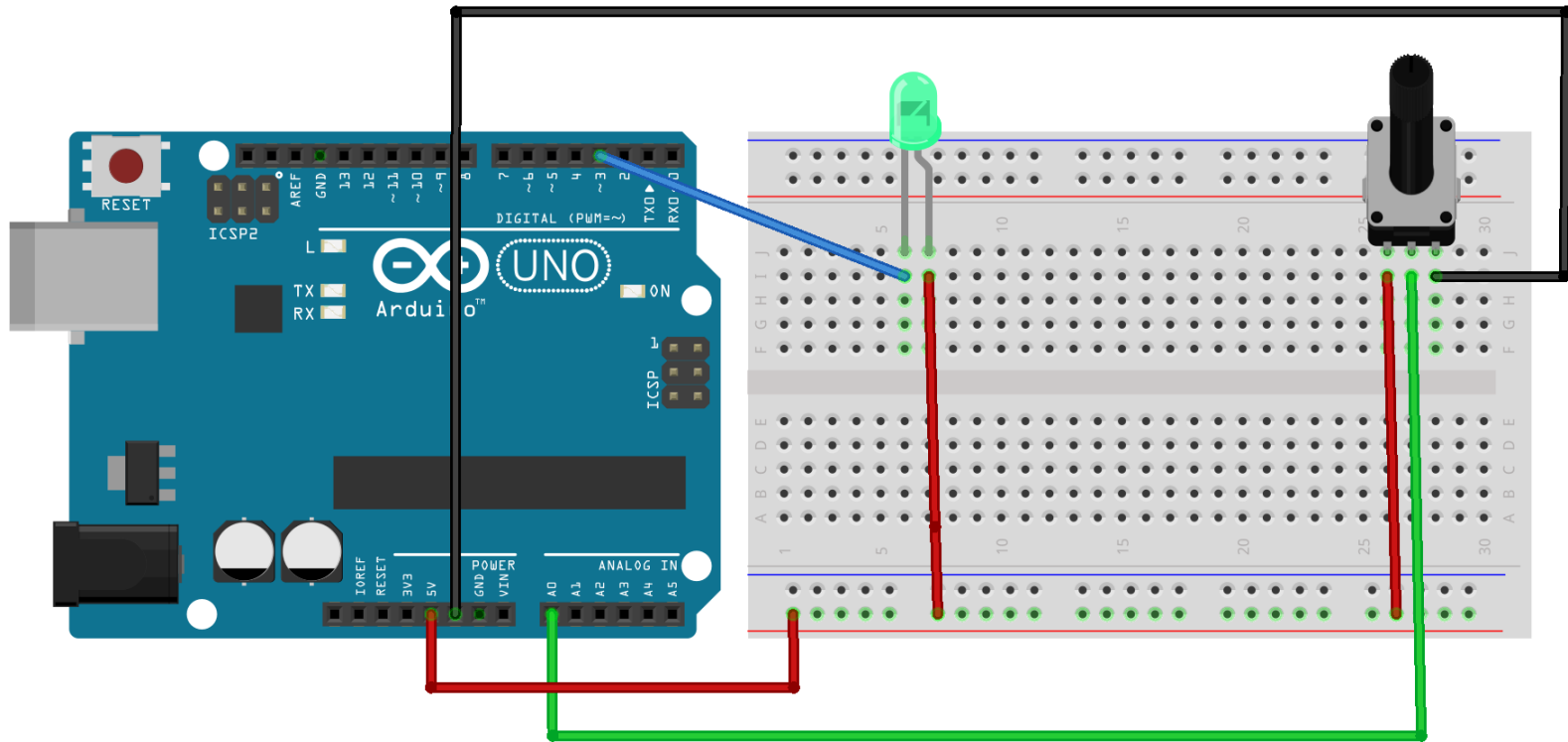
■ int map(value, fromLow, fromHigh, toLow, toHigh)

■ 함수 인자

- ❖ value : 변환할 입력 값
- ❖ fromLow : 입력 최소 값
- ❖ fromHigh: 입력 최대 값
- ❖ toLow : 출력 최소 값
- ❖ toHigh: 출력 최대 값
- ❖ Return : 변환된 출력 값

결선도

■ PWM을 이용한 LED 밝기 조절 예제 맞는 결선도를 그려 제출하시오



예시 – 가변 저항 사용한 예시

```
#define LED_PIN 3

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

    pinMode(LED_PIN, OUTPUT);
}

void loop()
{
    int sensorValue = analogRead(A0);

    int ledBrightness = map(sensorValue, 0, 1023, 0, 255);

    analogWrite(LED_PIN, ledBrightness);
}
```


왜 일정한 값이 읽히지 않을 까?

- 이론 적으로 직류 전압은 고정이지만 현실적으로 모든 전압은 항상 일정한 전압이 유지 되지 않는다
- 아두이노 우노에 공급되는 전원 전압도 약간의 흔들림이 있으며, 이 것을 ‘리플 전압’ 이라고 한다
- ADC에 리플 전압이 유입되어 약간의 흔들림이 있으며 이로 인하여 항상 동일한 값을 가질 수 없다
 - ❖ 이는 정상!
- 얼마나 적은 흔들림을 가지냐는 품질의 차이이다