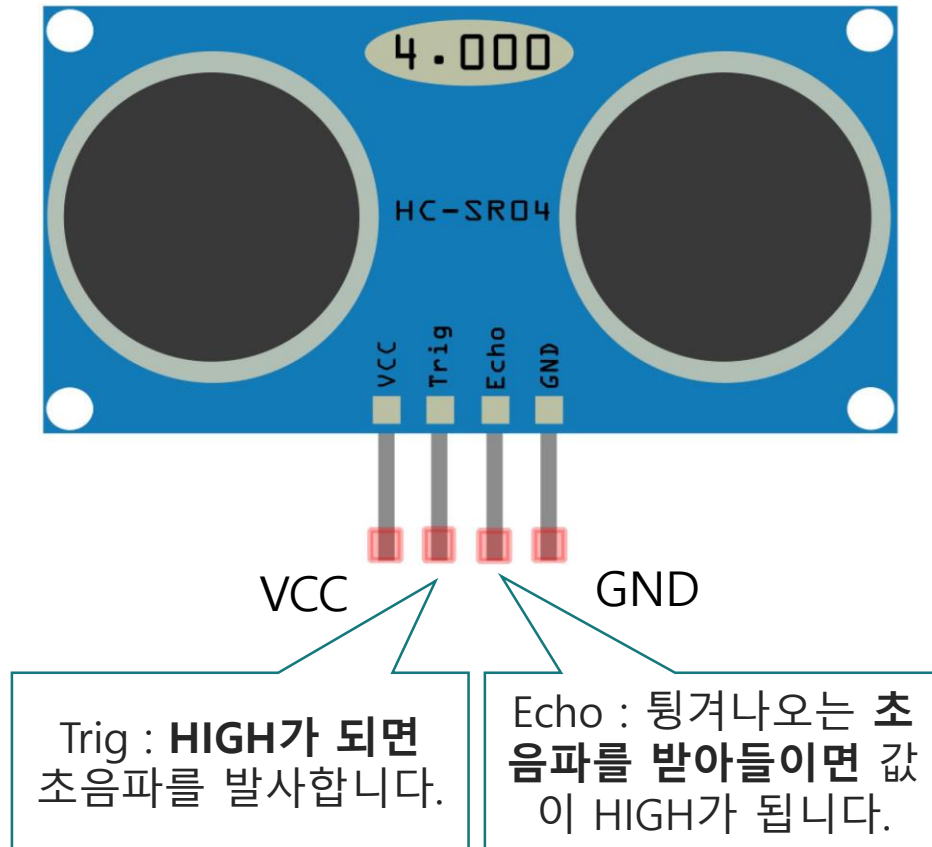


# 초음파 센서 예제



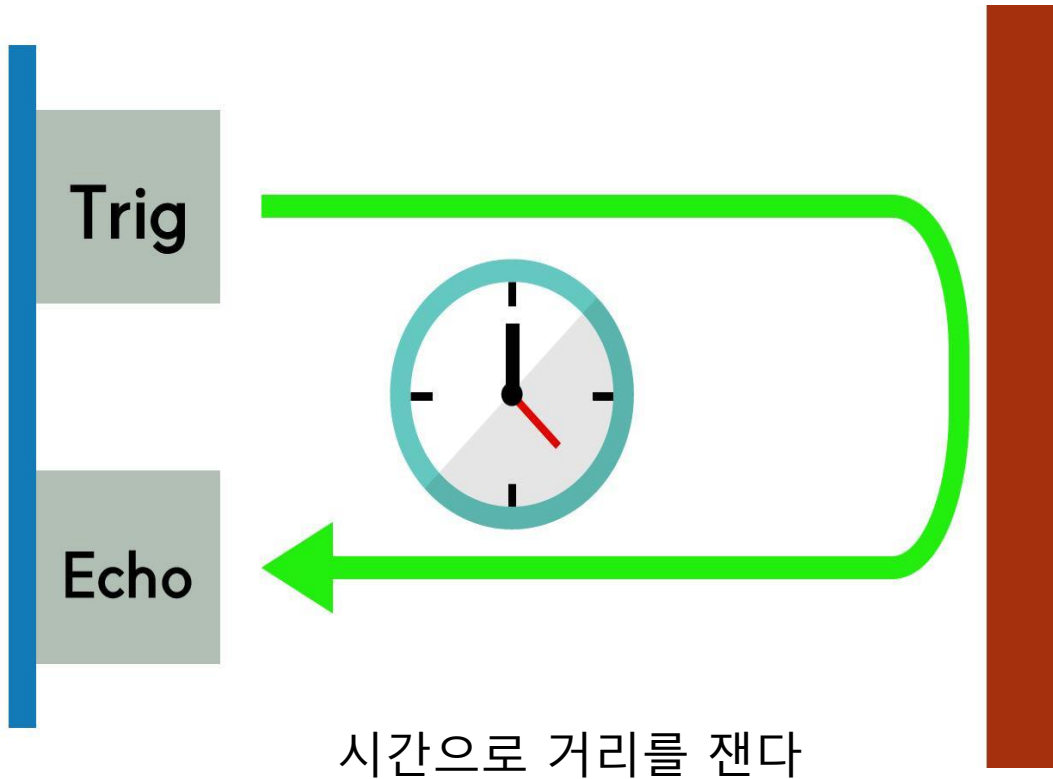
## 초음파 센서의 특징



초음파를 이용해 **사물이나 벽까지의 거리를 알려주는 센서**입니다.

초음파를 이용하기 때문에 **초음파를 발사하는 부분**과 **초음파를 받는 부분**이 있습니다.

## 초음파센서 원리

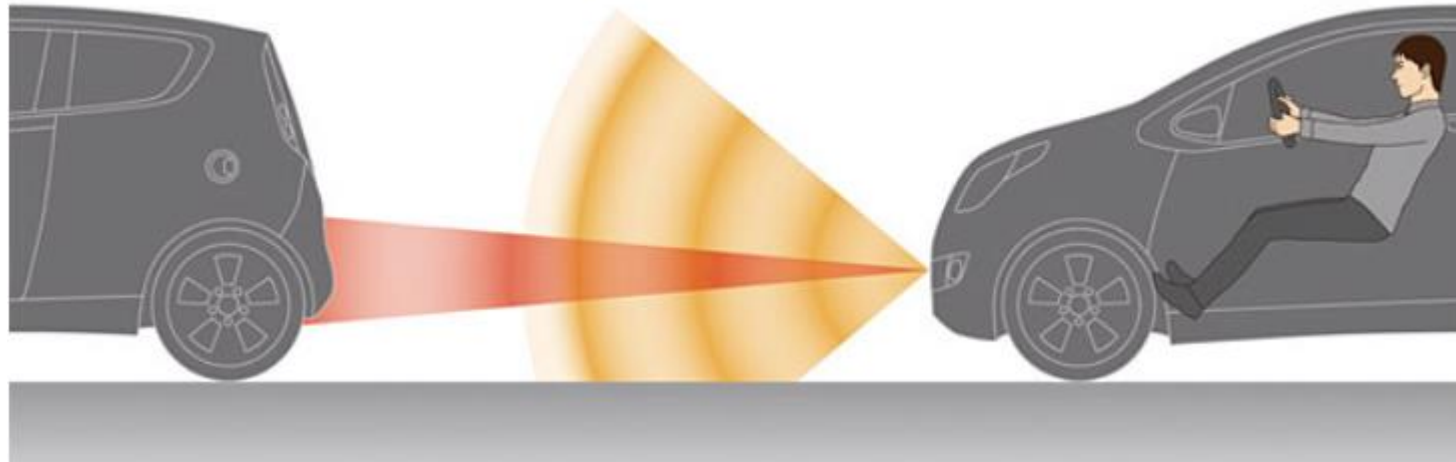


**Trig**가 HIGH가 되면 초음파가 발사됩니다.  
발사된 초음파가 사물이나 벽에 부딪히면 초음파는 반사됩니다.

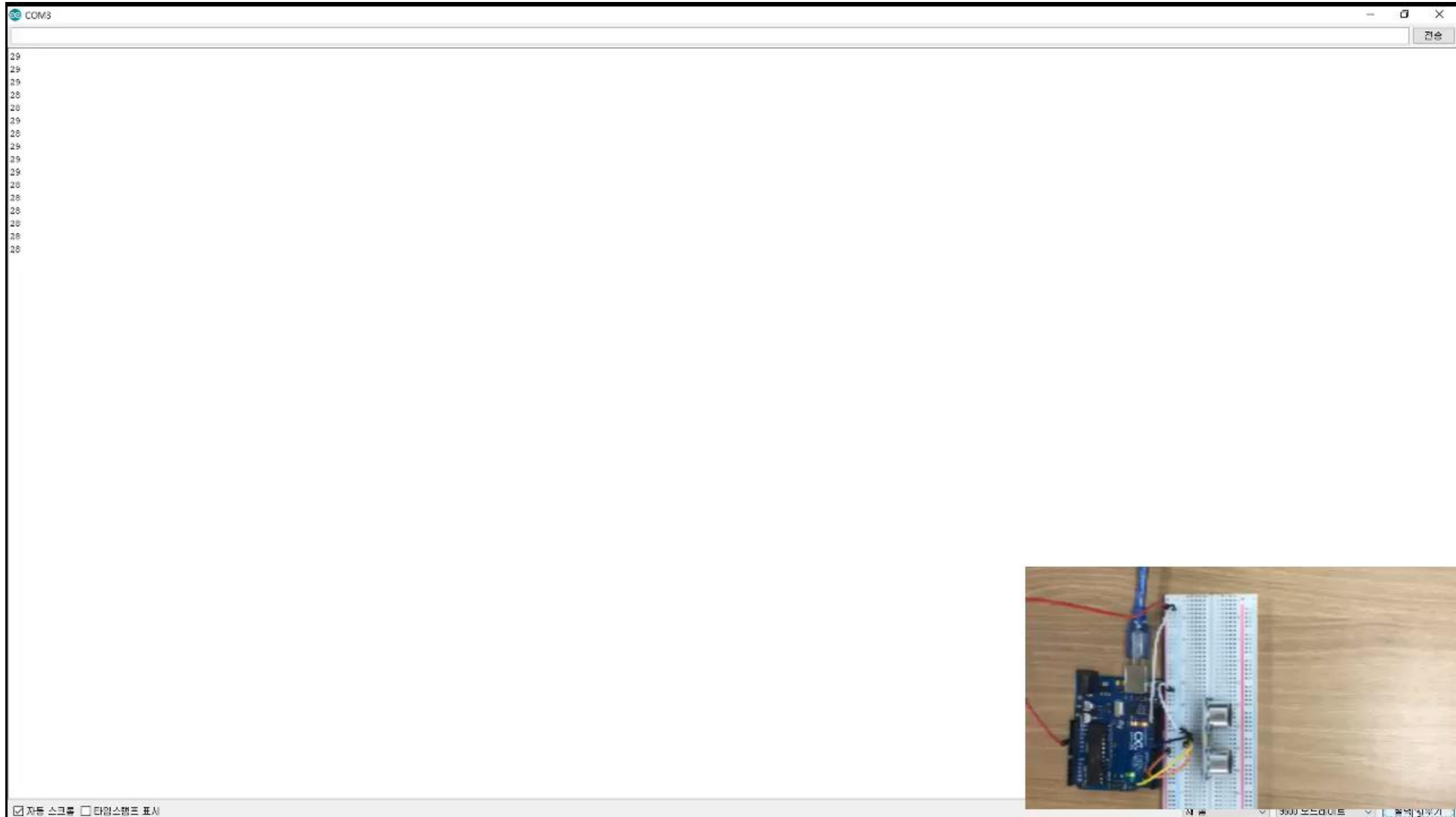
**Echo**는 초음파가 돌아오는 곳을 감지합니다.  
처음에 LOW였던 Echo는 반사된 초음파를 인지하면 HIGH로 바뀝니다.

초음파 발사 때부터, 초음파가 다시 돌아올 때까지 걸린 시간을 재서 거리를 알 수 있습니다.

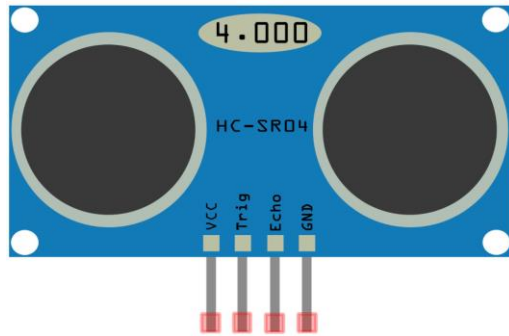
## 초음파 센서 활용사례



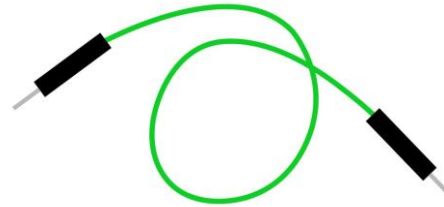
자율주행차



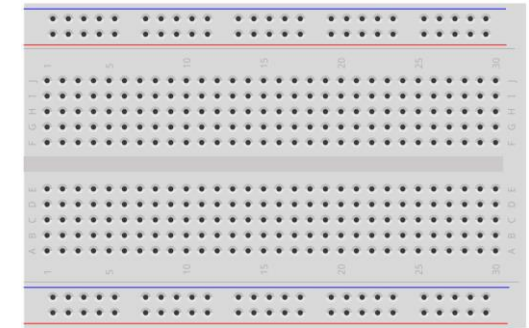
# 해보기 1) 초음파센서 거리측정 값 출력



초음파센서  
1개

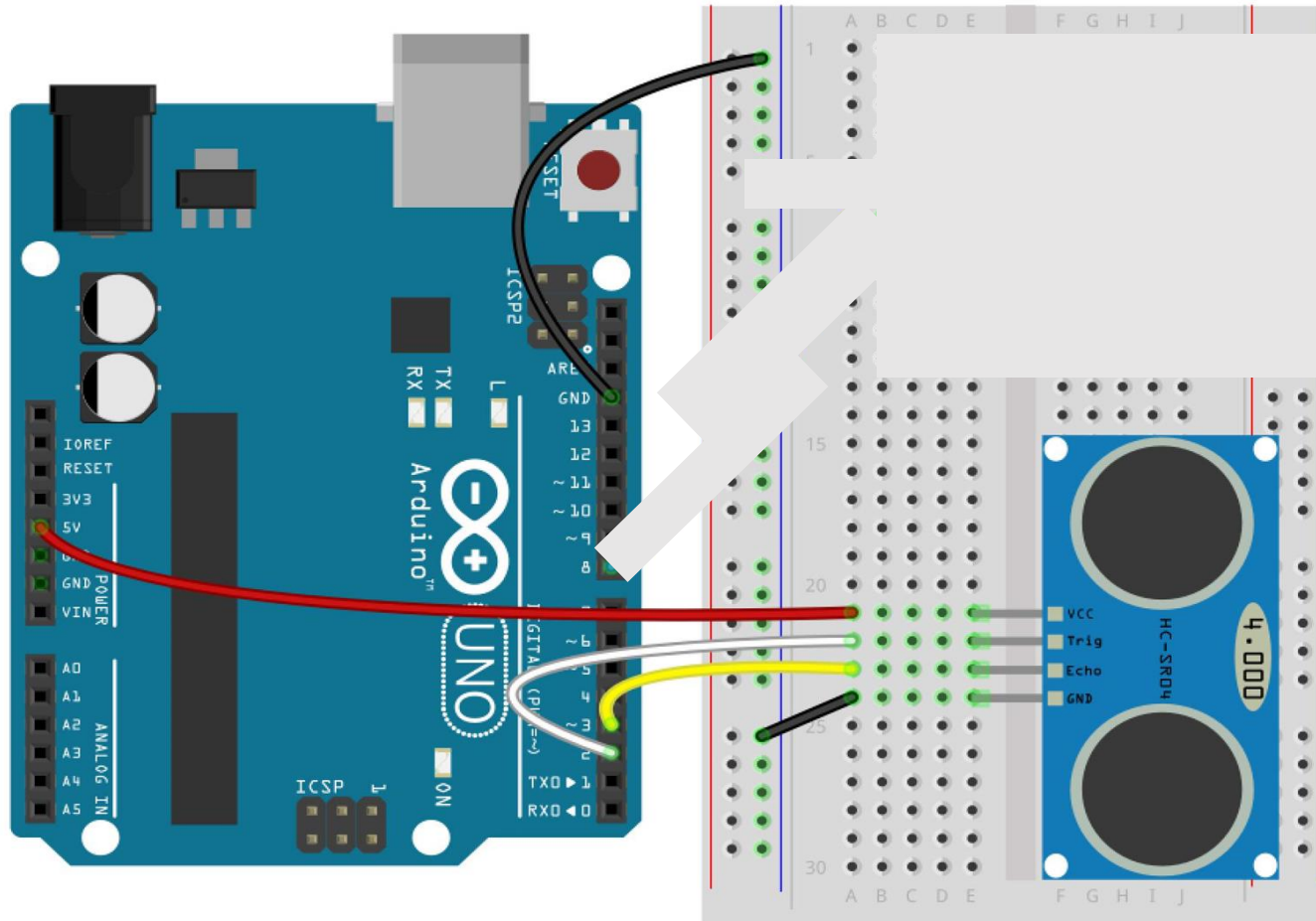


수수 점퍼 와이어  
7개



브레드보드  
1개

### 3) 초음파 센서의 각 핀을 연결



초음파센서의 VCC를 아두이노 보드의 5V전원 핀에, Trig를 2번 핀에, Echo를 3번 핀에 연결한다.

초음파센서의 GND는 그라운드 핀과 연결된 세로줄에 연결한다.



```

void setup() {
  Serial.begin(9600);  //시리얼 통신 속도 설정하기
  pinMode(2, OUTPUT);
  pinMode(3, INPUT);
}

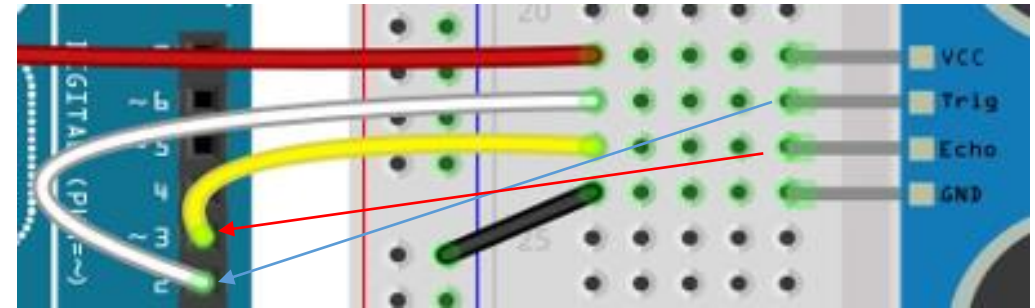
void loop() {
  digitalWrite(2, LOW);
  delayMicroseconds(2);
  digitalWrite(2, HIGH);
  delayMicroseconds(10);
  digitalWrite(2, LOW);

  int distance = pulseIn(3, HIGH)/ 58.2;
  Serial.println(distance);
  delay(100);
}

```

초음파센서 초기화

시간을 거리(cm) 로 변환

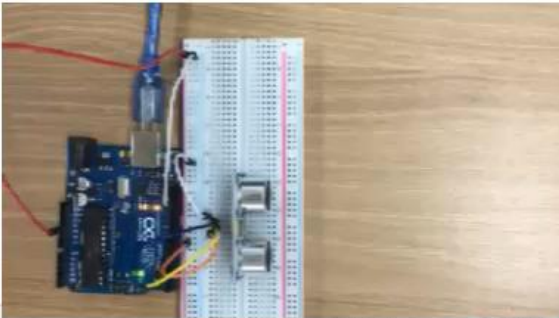




COM3

29  
29  
29  
28  
28  
28  
29  
28  
28  
29  
29  
29  
28  
28  
28  
28  
28  
28

☒ 자동 스크롤 ☐ 타임스탬프 표시



## Serial.begin(9600);

Serial.begin(매개변수)

시리얼 통신 설정하기. 아두이노 보드와 PC가 시리얼 모니터를 통해 통신할 수 있도록 보드레이트 값을 동일하게 설정해야 통신을 할 수 있다. 시리얼 모니터 우측 하단의 보드레이트와 값을 맞춰준다.

## Serial.println(변수);

Serial.println()

아두이노 보드가 ()안의 값을 PC로 메시지를 보내는 명령어.

Serial.println("Hello Pc!"); 와 같이 문자열을 보내거나,

Serial.println(**pr**); 과 같이 pr 변수에 모든것을 담아서 메시지를 보내기도 한다.

## pinMode(핀번호, OUTPUT);

pinMode(2, OUTPUT);

초음파센서의 Trig로 초음파를 출력합니다. 2번 핀을 출력핀으로 설정.

pinMode(3, INPUT);

초음파센서의 Echo로 초음파가 반사되어 돌아온 거리를 입력받습니다. 3번핀을 입력핀으로 설정.

## digitalWrite(핀번호, HIGH);

digitalWrite(2, HIGH);

2번은 pinMode(2, OUTPUT); 로 초음파 출력 모드로 설정되어 있으므로, digitalWrite함수에서 **HIGH가 되면 2번 핀으로 초음파가 출력**됩니다.

digitalWrite(2, LOW);

지정 시간이 지난 후 LOW를 만나면 **초음파 출력을 정지**합니다.

## delayMicroseconds(10);

**delayMicroseconds(멈출시간);**

아두이노 보드를 일정시간 멈추는 함수. 멈출 시간의 단위는 마이크로 초. 초음파는 미세한 시간 단위로 다루어야 하  
기때문에 delay() 함수보다 delayMicroseconds() 함수를 사용합니다.

1 초=1000밀리초. 1 밀리초=1000마이크로초. (1마이크로초 = 0.001밀리초 = 0.000001초)

delayMicroseconds(10); //은 0.00001초이다.

```
int distance = pulseIn(3, HIGH) / 58.2;
```

`pulseIn(핀번호, HIGH);`

디지털 입력 핀의 전압이 LOW 또는 *HIGH가 될 때까지 걸린 새간을 재는 함수.*

핀번호는 `pinMode()`에서 꼭 **INPUT으로 설정** 되어있어야 합니다.

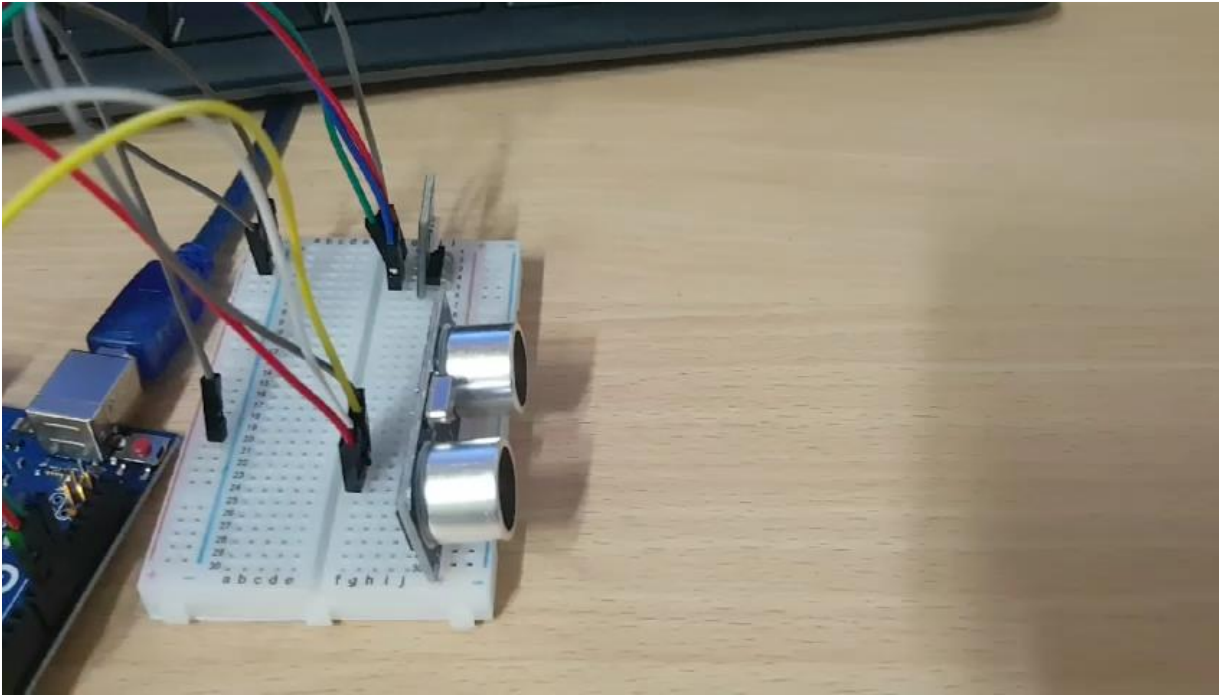
`pluseIn(3, HIGH);` // 3번핀의 전압이 HIGH가 될때까지 걸린 시간을 재서 `duration`변수에 넣어줍니다. 이때 `int`형 변수에 다 담을수가 없어서 그보다 큰 `long`형 변수에 담아야 합니다.

# 초음파 센서 실습문제



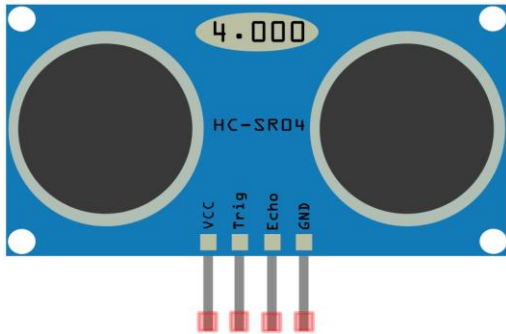


## 실습) 거리에 따라 삼색LED 색이 바뀌어 봅시다.

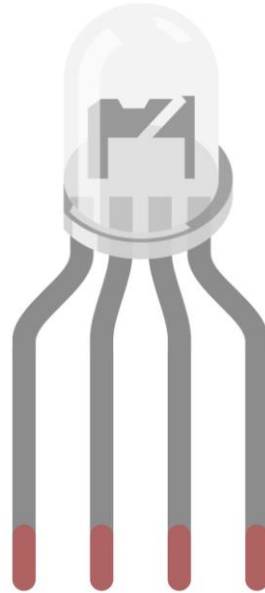


물체와 센서와의 거리가  
10cm 미만이면 빨간색 불이,  
20cm 미만이면 초록색 불이,  
20cm 보다 멀리 있으면 파란색불이 켜질 수  
있도록 합시다!

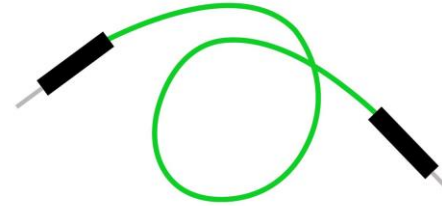
# 준비물



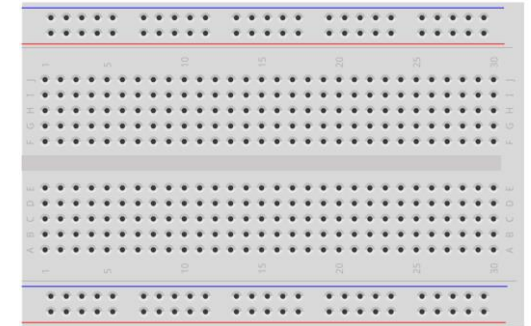
초음파센서  
1개



삼색 LED  
1개



수수 점퍼 와이어  
9개



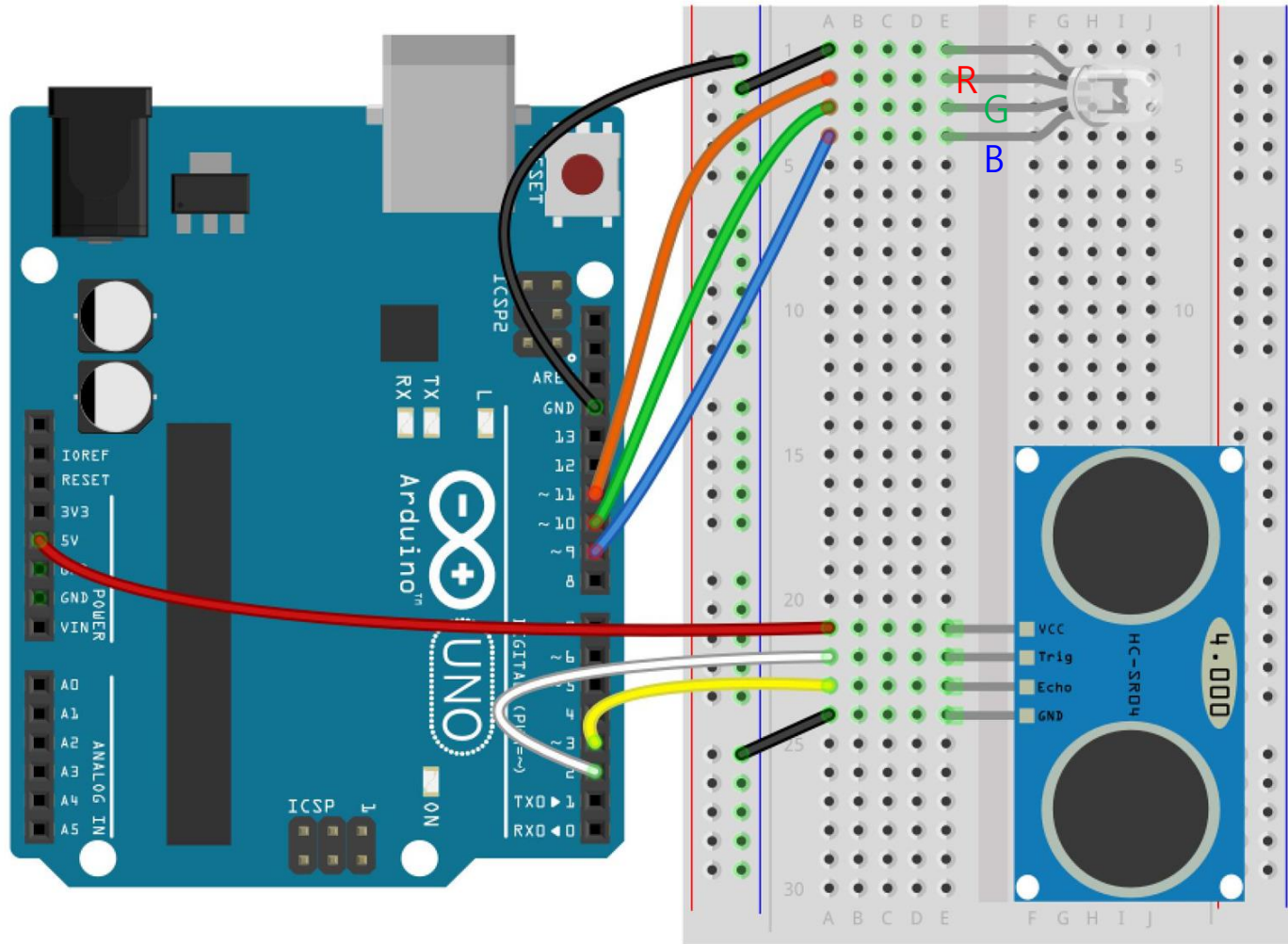
브레드보드  
1개

# 레시피

- 1) 삼색 LED를 연결합니다. R-11번 / G-10번 / B-9번 핀에 연결해 주세요.
- 2) 초음파 센서를 연결해 주세요
  - GND와 VCC 는 각각 GND와 5V에 연결해 주세요.
  - TRIG는 2번 핀에, ECHO는 3번 핀에 연결해 주세요.
- 3) 보여드리는 대로 프로그램을 코딩 해주세요.
- 4) 아두이노와 PC를 연결해 주세요.
- 5) 스케치 상단의 "확인" 버튼과 "업로드" 버튼을 누릅니다.
- 6) 실행이 되면 거리에 따라 색이 바뀌는 삼색 LED가 구현됩니다.



# 하드웨어를 만들어 봅시다



초음파센서의 *Echo*라고 적힌 부분을 아두이노 보드의 **3번** 핀과 연결한다.

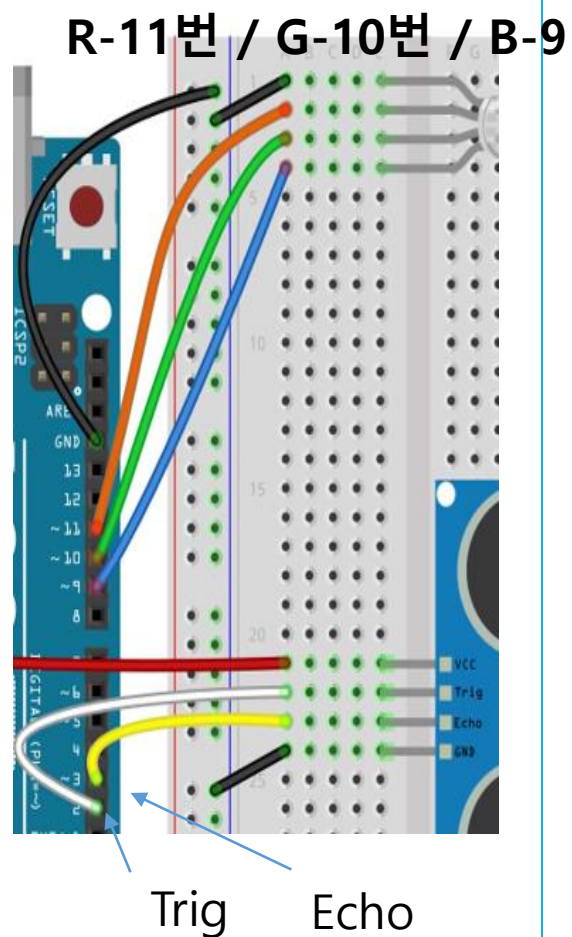
## 아두이노 설정 절차

- ① Trig와 Echo의 핀번호를 2,3번으로 설정합니다.
- ② 삼색 LED의 R,G,B핀인 11,10,9번을 0으로 세팅합니다.
- ③ Trig(2번핀)을 이용하여 초음파를 발사합니다
- ④ Echo(3번핀)을 이용하여 초음파의 반사시간을 체크하고, 센티미터로 변경합니다.
- ⑤ 거리가 10센티미터 미만이면 빨간색 불을 켭니다.
- ⑥ 거리가 20센티미터 미만이면 초록색 불을 켭니다.
- ⑦ 거리가 30센티미터 미만이면 파란색 불을 켭니다.
- ⑧ 30이면 loop 함수를 종료합니다.
- ⑨ 0.1초 동안 멈추고 다시 loop함수가 반복합니다.

```
void setup() {
  pinMode(2, OUTPUT);
  pinMode(3, INPUT);
}
```

```
void loop() {
  analogWrite(9, 0);
  analogWrite(10, 0);
  analogWrite(11, 0);
```

RGB  
모두 끄기



```
digitalWrite(2, LOW);
delayMicroseconds(2);
digitalWrite(2, HIGH);
delayMicroseconds(10);
digitalWrite(2, LOW);
```

초음파센서 초기화

```
int distance = pulseIn(3, HIGH) / 58.2;    시간을 cm단위 거리 변환
```

```
if (distance < 10) {
```

사물과 거리가 10cm미만

```
  analogWrite(11, 255);
```

빨간색

```
} else if (distance < 20) {
```

사물과 거리가 20cm미만

```
  analogWrite(10, 255);
```

녹색

```
} else if (distance < 30) {
```

사물과 거리가 30cm미만

```
  analogWrite(9, 255);
```

파란색

```
}
```

```
delay(100);
```

```
}
```

## Q&A



수고하셨습니다.