

# 스테레오 마이크 & 이미지 정보

일	월	화	수	목	금	토
1	2	3	4	5 A(1~3) C(4~6) B(7~9)	6 C(5, 8, 9)	7
8	9 A(1, 5, 9) B(6~8)	10	11	12 기말고사	13	14
15	16	17	18	19	20	21

# 센서로 소리의 방향을 알수 있을까?

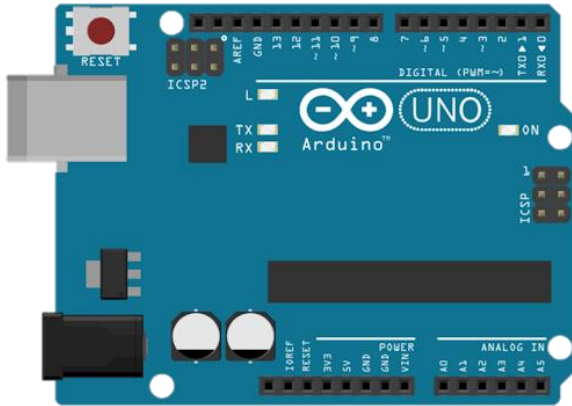
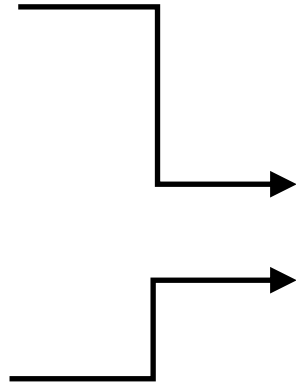


# 전체 구성

소리 감지(마이크1)



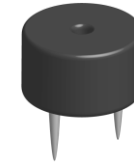
소리 감지(마이크2)



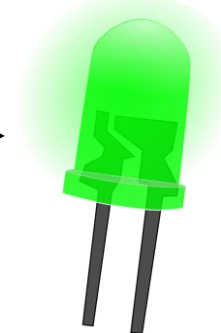
모터를 이용하여 카메라 방향 이동



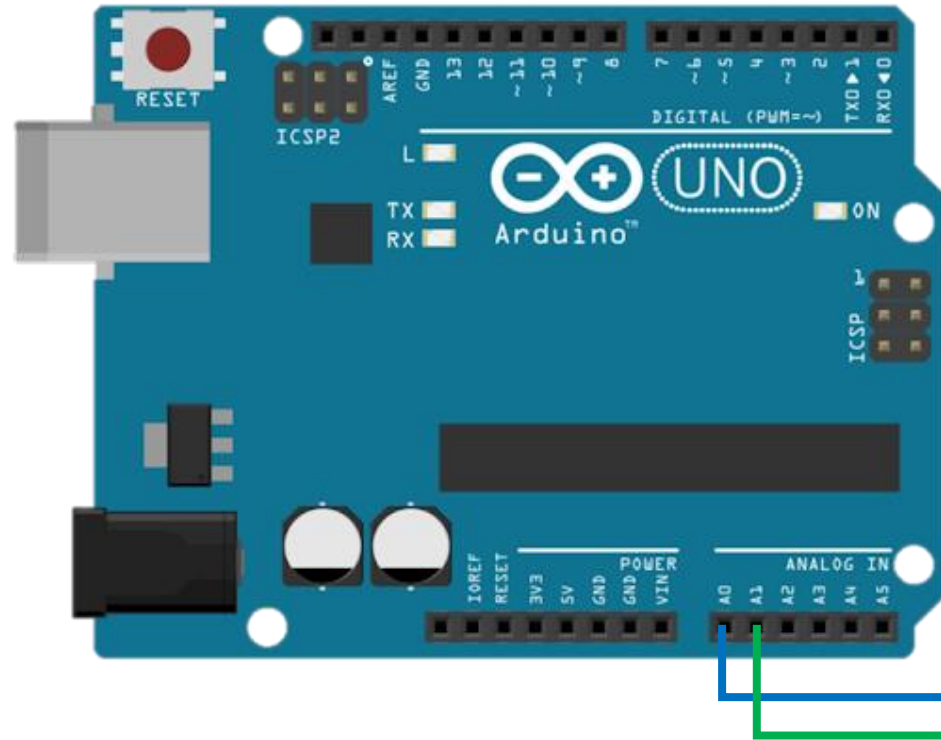
측정 완료 경고음 발생



LED 인디케이터



# 적외선 감지 센서 실험



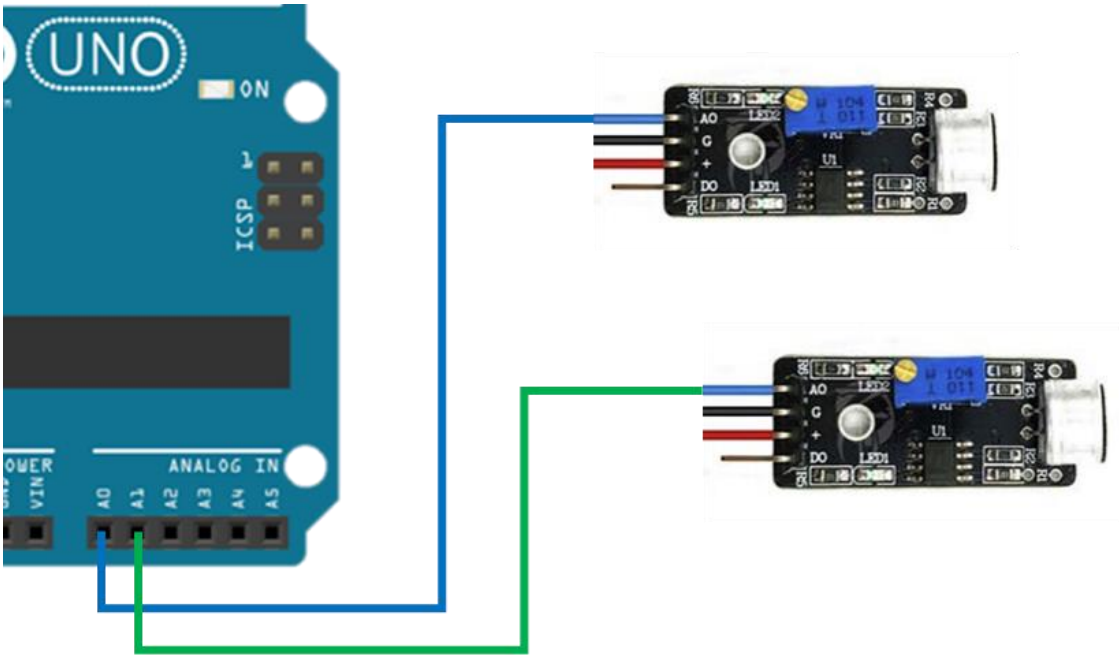
소리 감지(마이크1)



소리 감지(마이크2)



# 적외선 라인감지 센서 실험

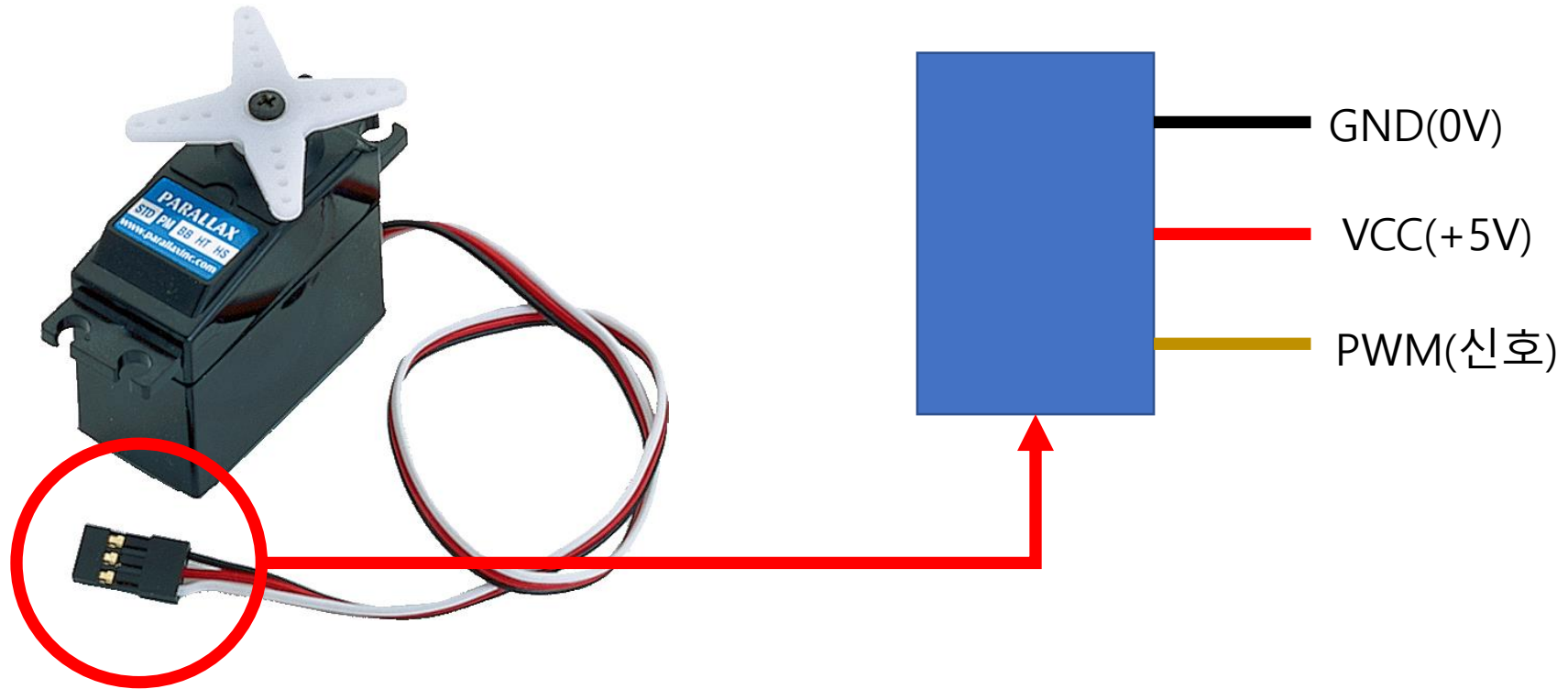


```
void setup()
{
}

void loop()
{
    int left = analogRead(A0) ;
    int right = analogRead(A1) ;

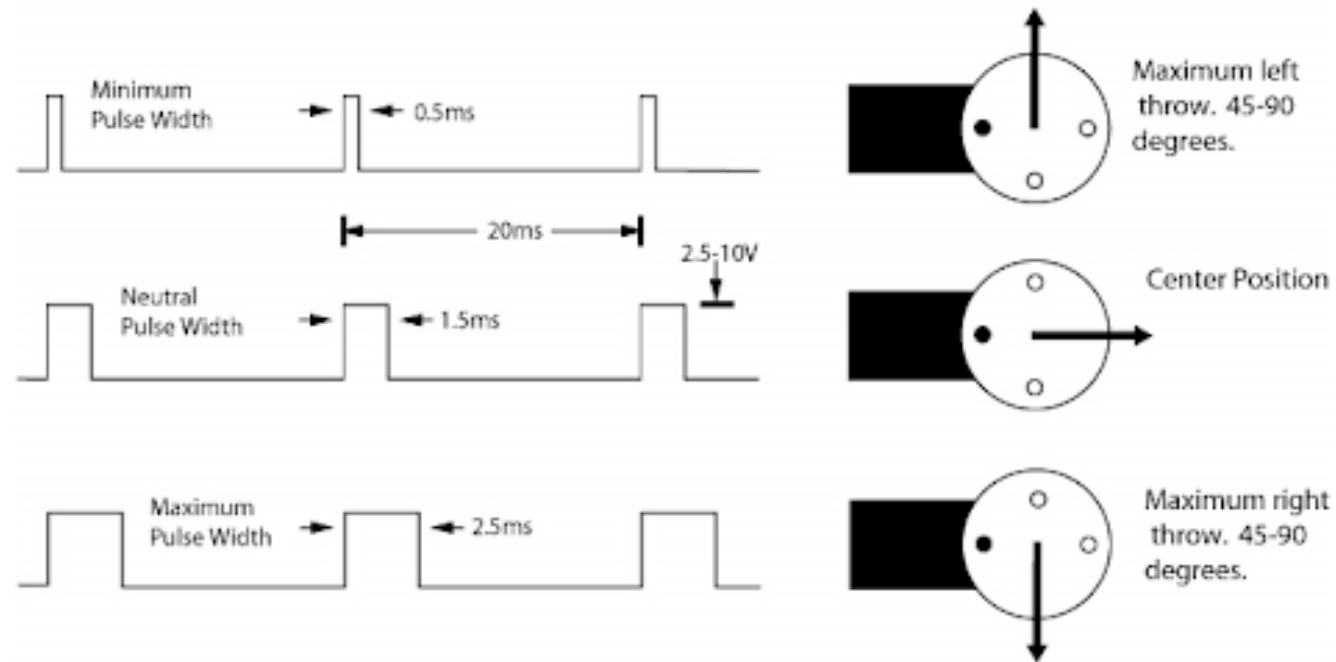
    Serial.print(left) ;
    Serial.print(" , ") ;
    Serial.println(right) ;
}
```

# RC 서보모터



# PWM을 이용한 RC 서보모터 제어

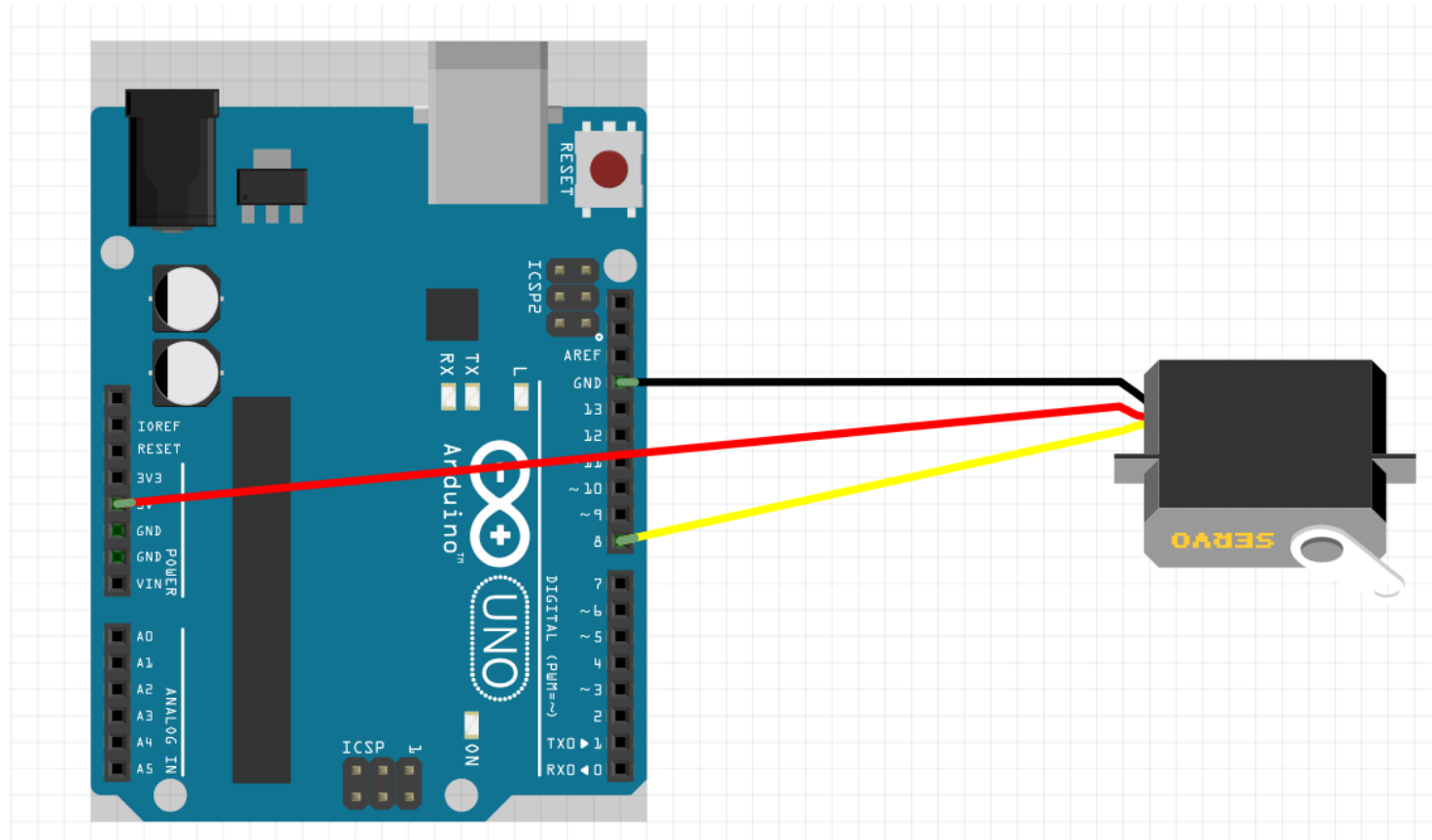
## R/C Control Signal Theory





# 아두이노를 이용한 서보모터 제어

- 테스트 회로 구성



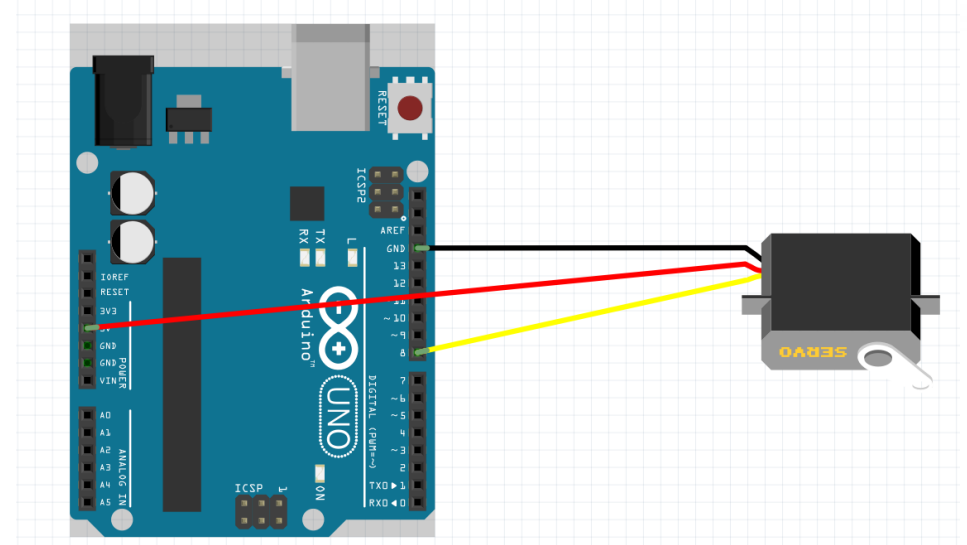
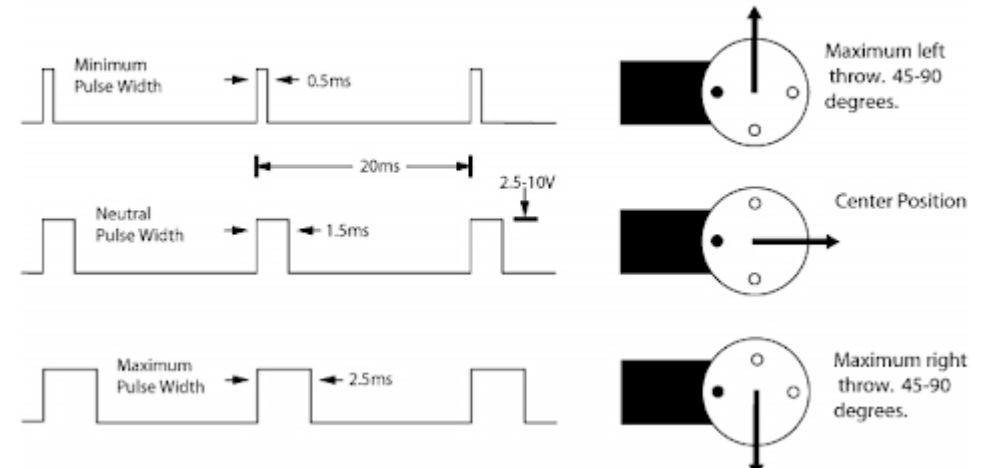
# 아두이노를 이용한 서보모터 제어

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

void loop()
{
  digitalWrite(8, HIGH);
  delayMicroseconds(1400);

  digitalWrite(8, LOW);
  delayMicroseconds(20000-1400);
}
```

R/C Control Signal Theory

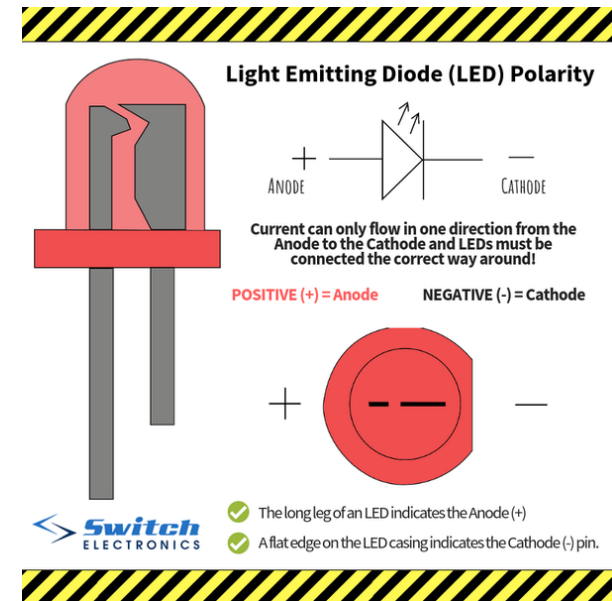
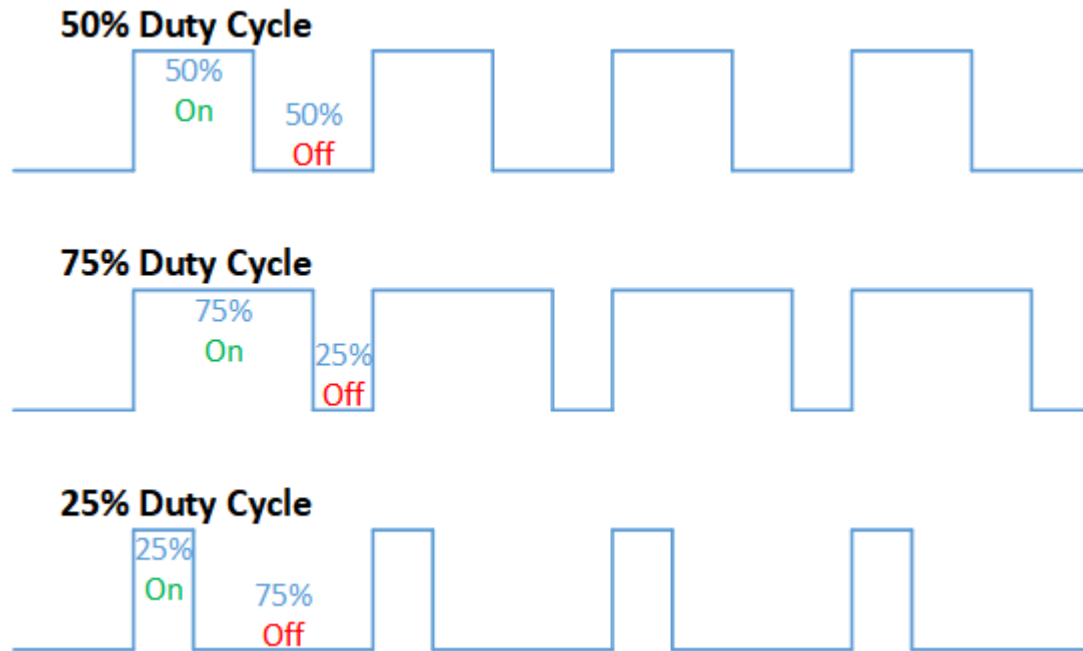


# QUIZ : 소리에 반응

- 소리가 감지되면 소리 방향으로 서모모터를 제어하자.

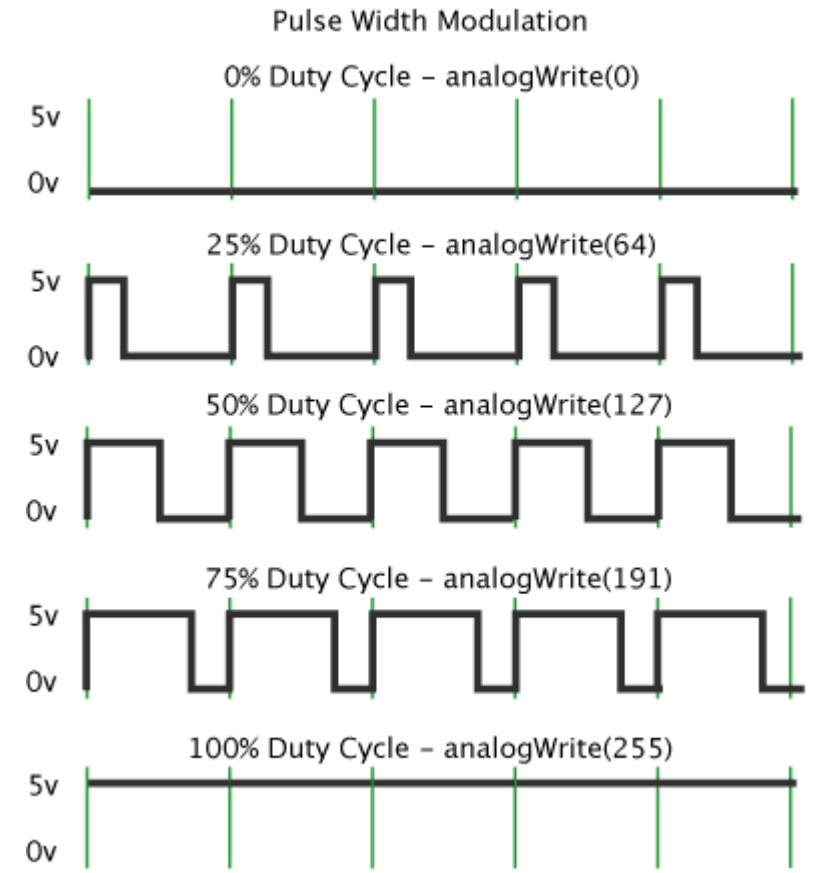
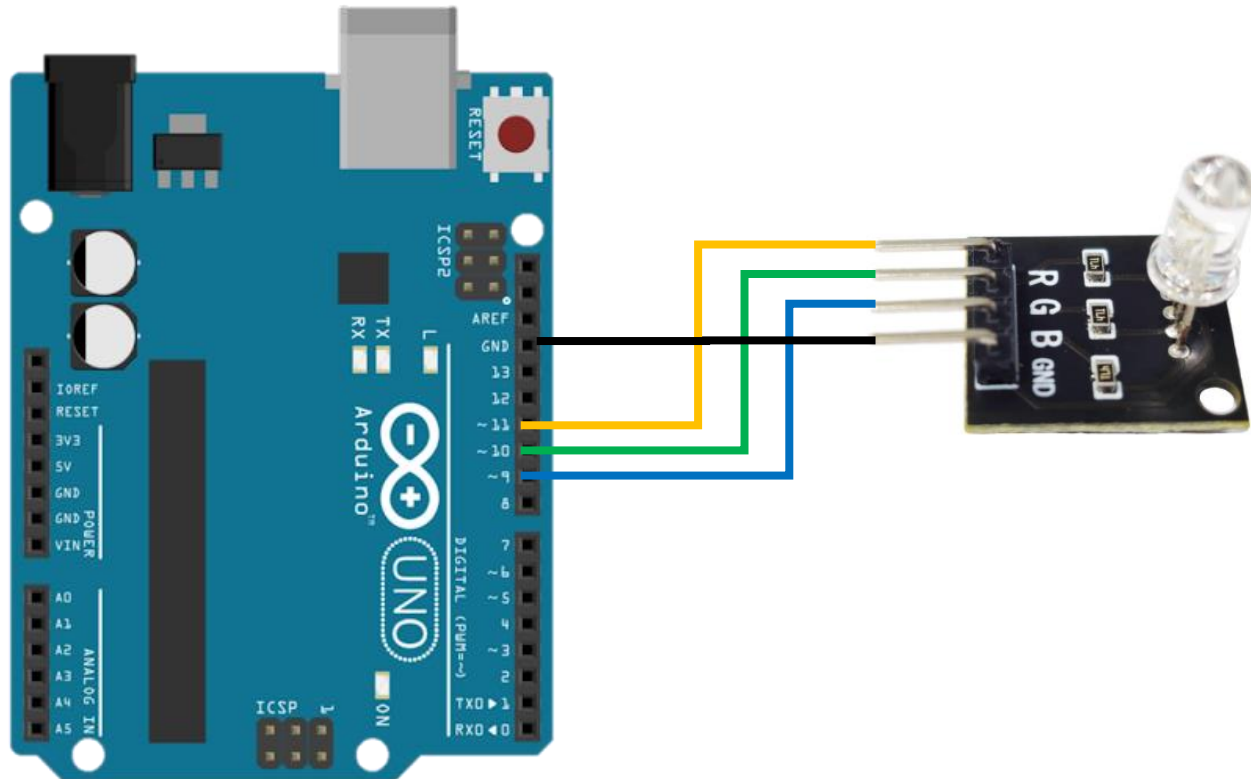


# PWM을 이용한 LED 밝기 제어 (analogWrite)



# 아두이노를 이용한 LED 밝기제어 예제

- 함수 : `analogWrite(핀번호, duty cycle)`



# 아두이노를 이용한 LED 밝기제어 예제

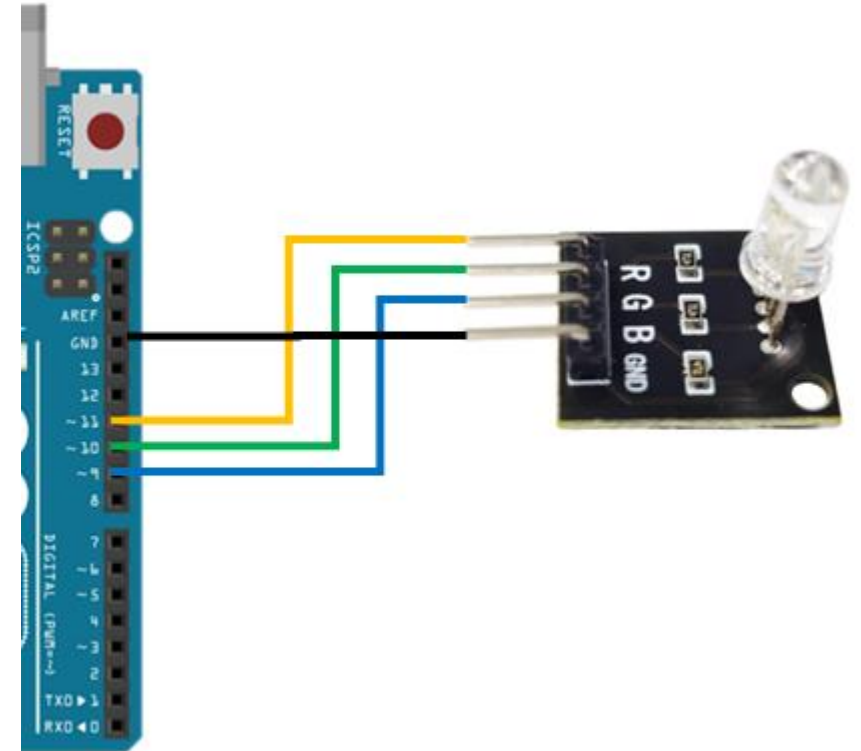
- 함수 : analogWrite(핀번호, duty cycle)

```
void setup()
{
  pinMode(9, OUTPUT) ;
}

void loop()
{
  analogWrite(9, 0) ;
}
```

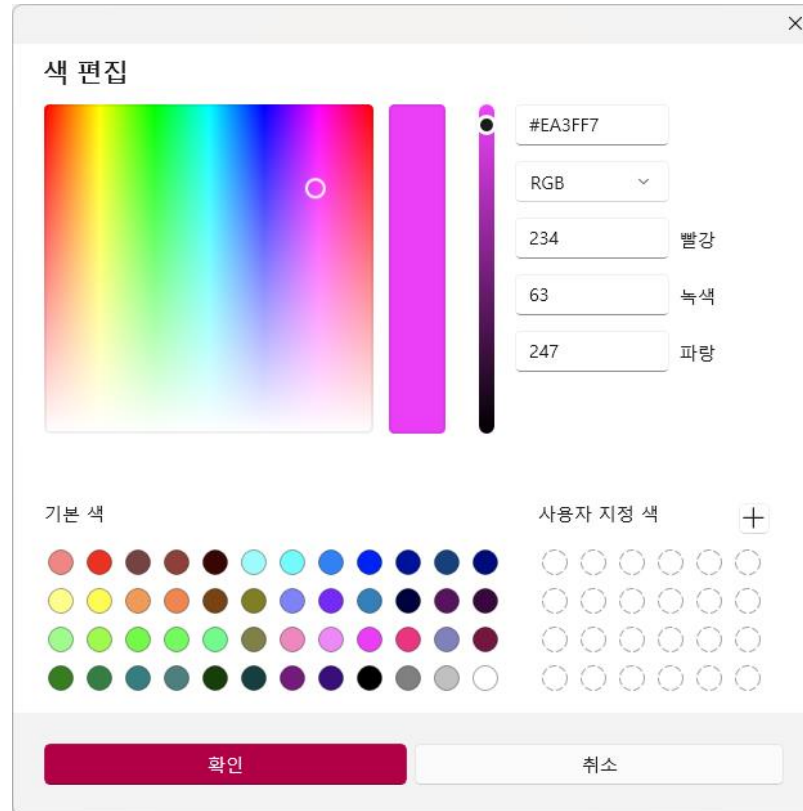
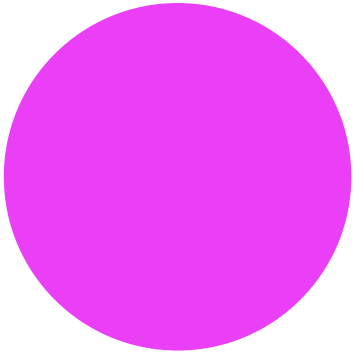
```
void setup()
{
  pinMode(9, OUTPUT) ;
}

void loop()
{
  analogWrite(9, 255) ;
}
```



# QUIZ : 다양한 색상 표현하기

- 3색(RGB) LED를 이용하여 아래의 색을 표현하시오



[그림 1] 빛의 3원색인 Red, Blue, Green. 세 가지 색상을 조합하면 White가 된다.



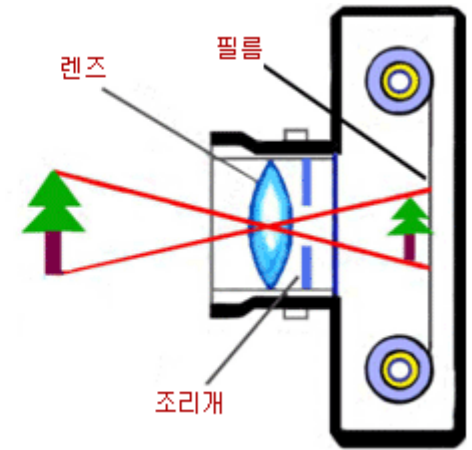
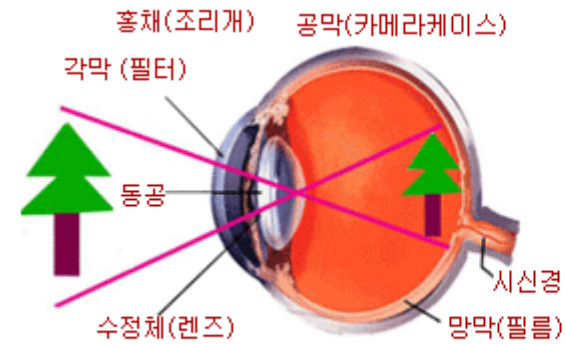
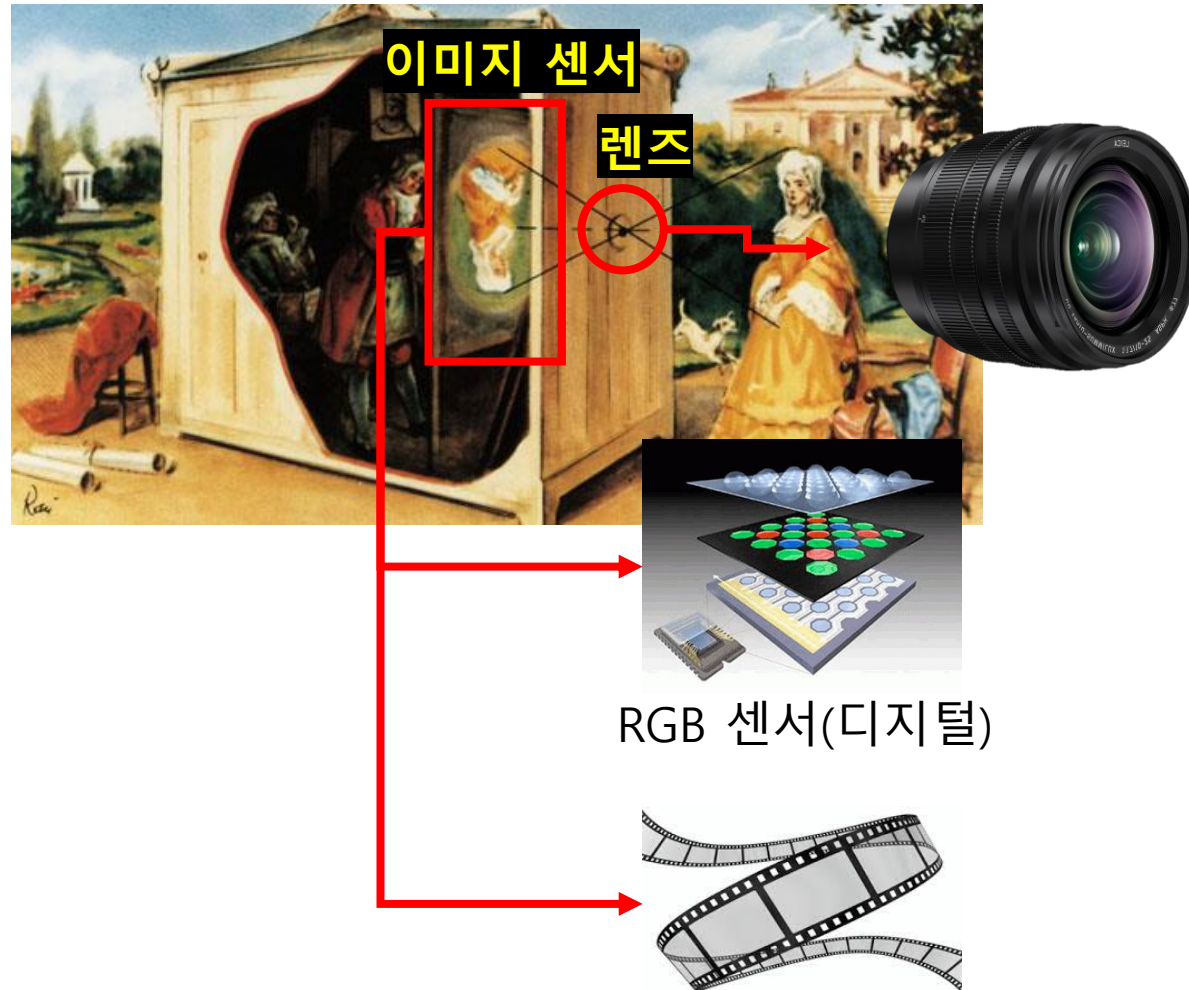
# 이미지를 어떻게 얻을까?



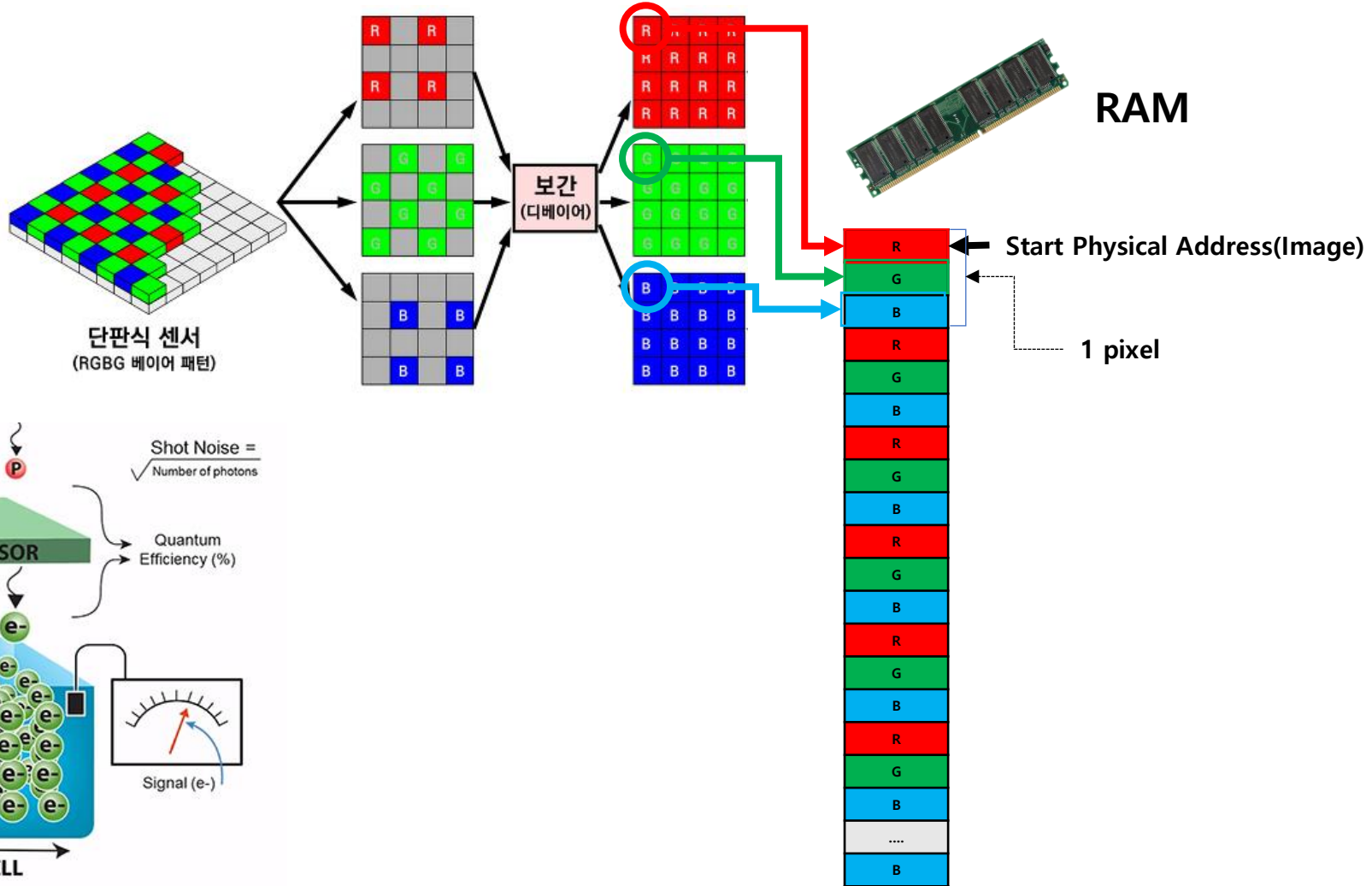
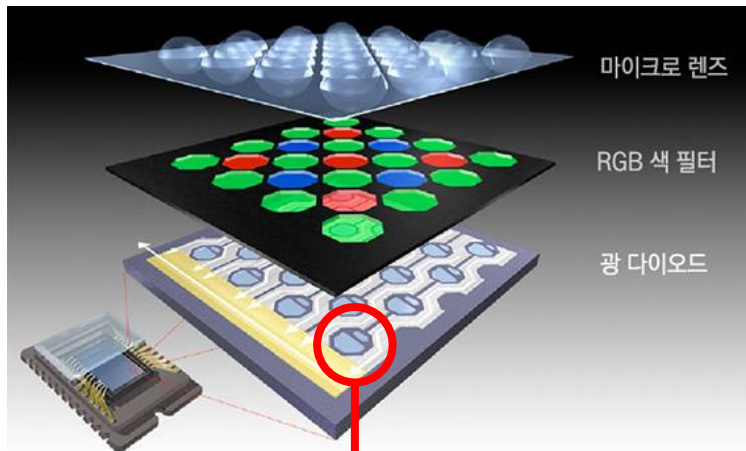
**카메라 옵스큐라**  
(라틴어(camera obscura) 어두운방 이라는 뜻)



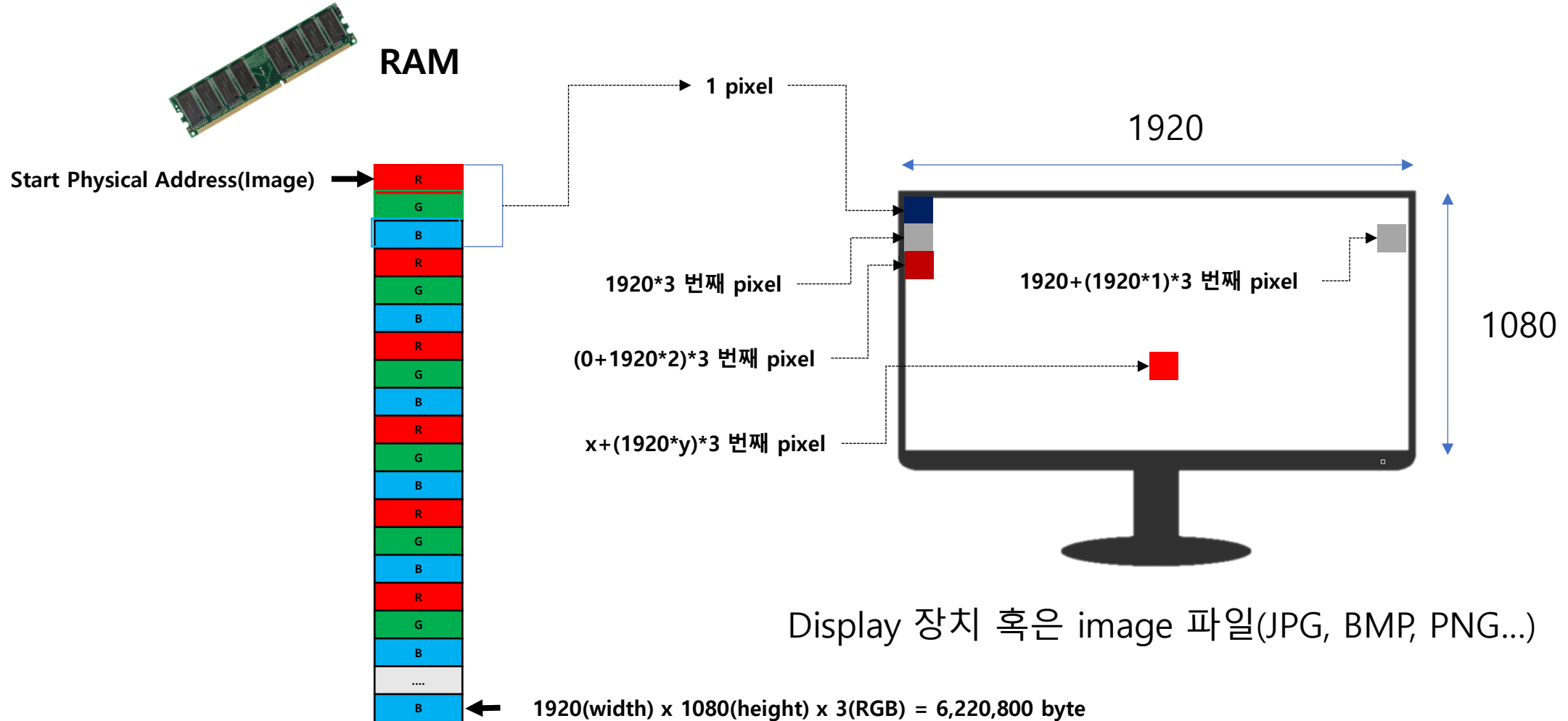
# 컴퓨터(카메라)는 어떻게 이미지를 얻을까?



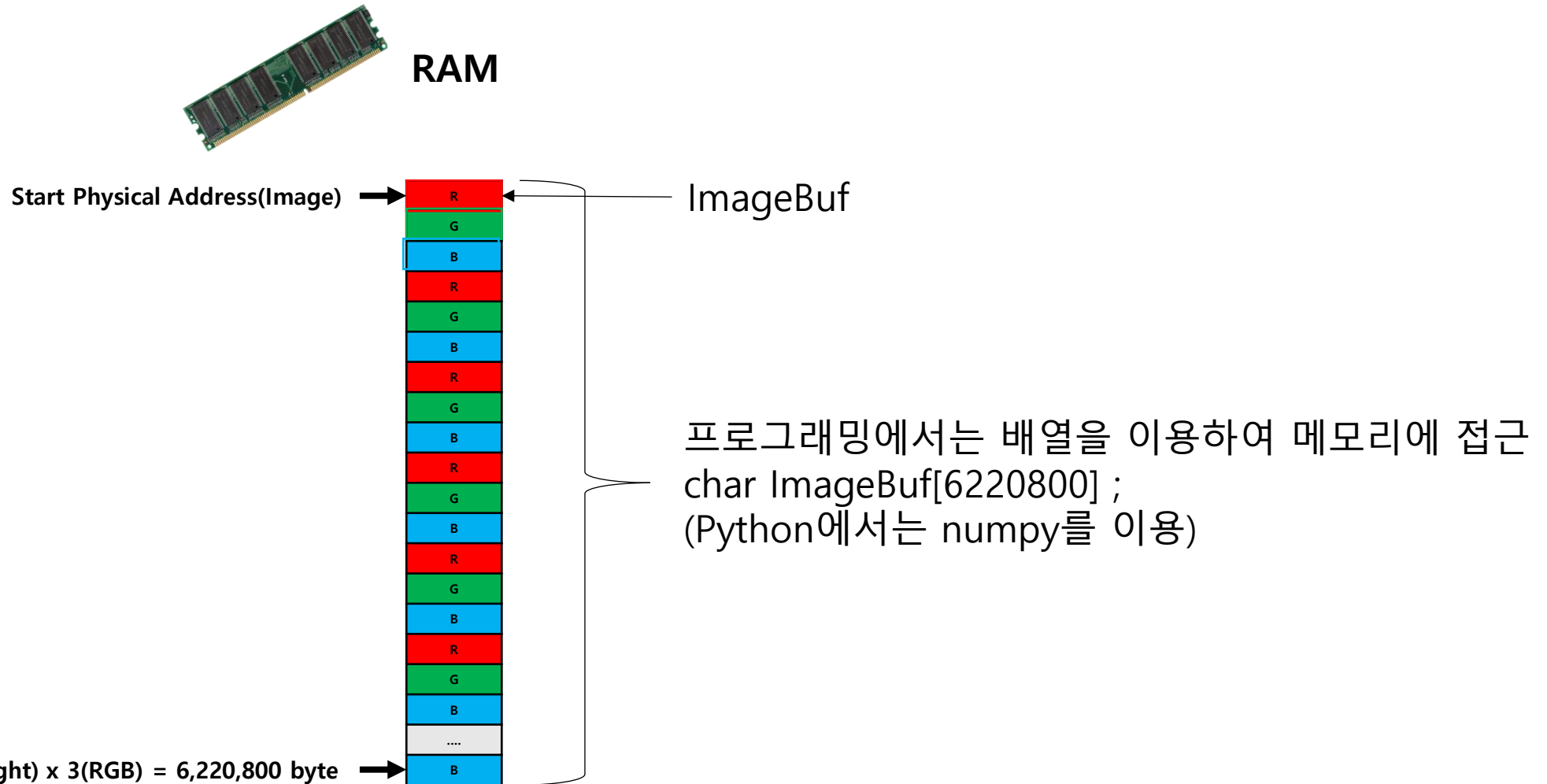
# 컴퓨터(카메라)는 어떻게 이미지를 얻을까?



# 컴퓨터(카메라)는 어떻게 이미지를 표시 할까?

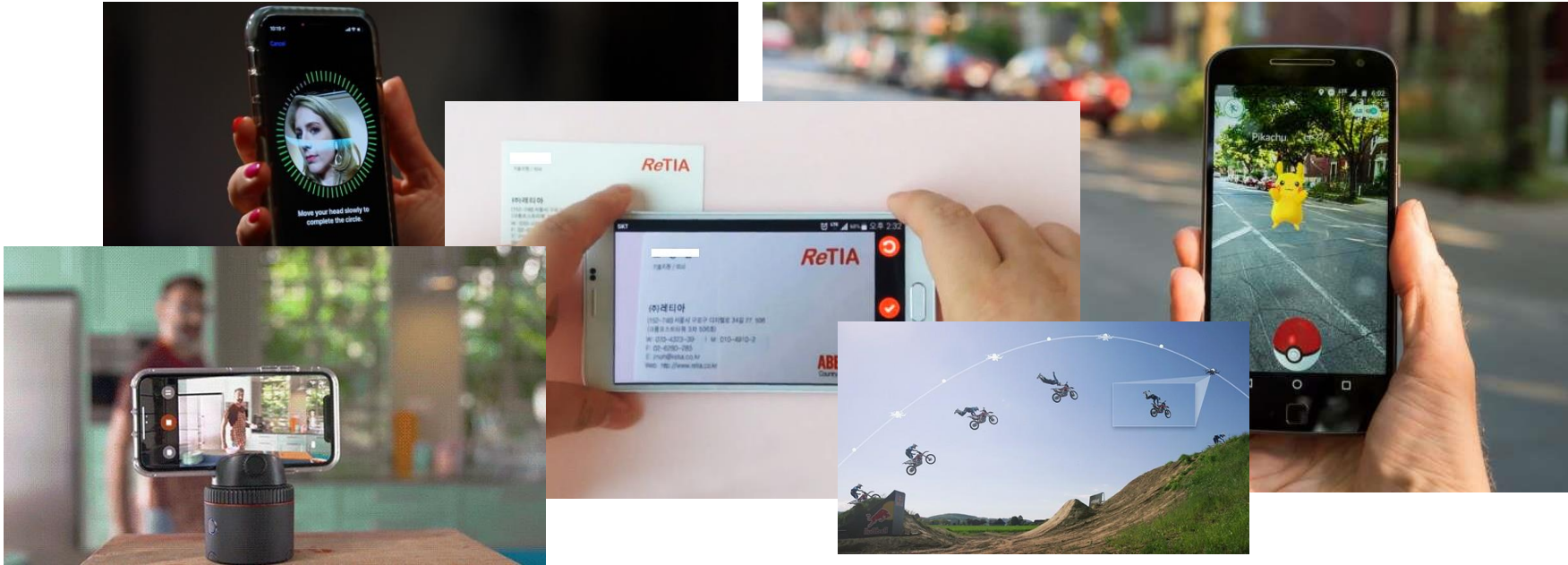


# 우리는 어떻게?



# 이제 무얼, 어떻게 하지?

- 영상처리를 이용한 서비스를 만들고 싶은데.



- 그럼 무엇부터 해야 하지?



# OpenCV

- OpenCV : Opensource Computer Vision
  - <https://opencv.org/>
  - Open Source : OpenCV is open source and released under the BSD 3-Clause License. It is free for commercial use.
  - Optimized : OpenCV is a highly optimized library with focus on real-time applications.
  - Cross-Platform : C++, Python and Java interfaces support Linux, MacOS, Windows, iOS, and Android.
- Computer Vision 알고리즘을 오픈소스로 공개, 최신의 알고리즘을 가장 빠르게 적용
- 특별한 경우가 아니라면 거의 모든 Computer Vision의 기본
- OpenCV의 함수를 이용하여 편리하게 이미지 데이터에 접근하고 영상처리가 가능
  - 기본 데이터 컨테이너 : `cv::Mat`