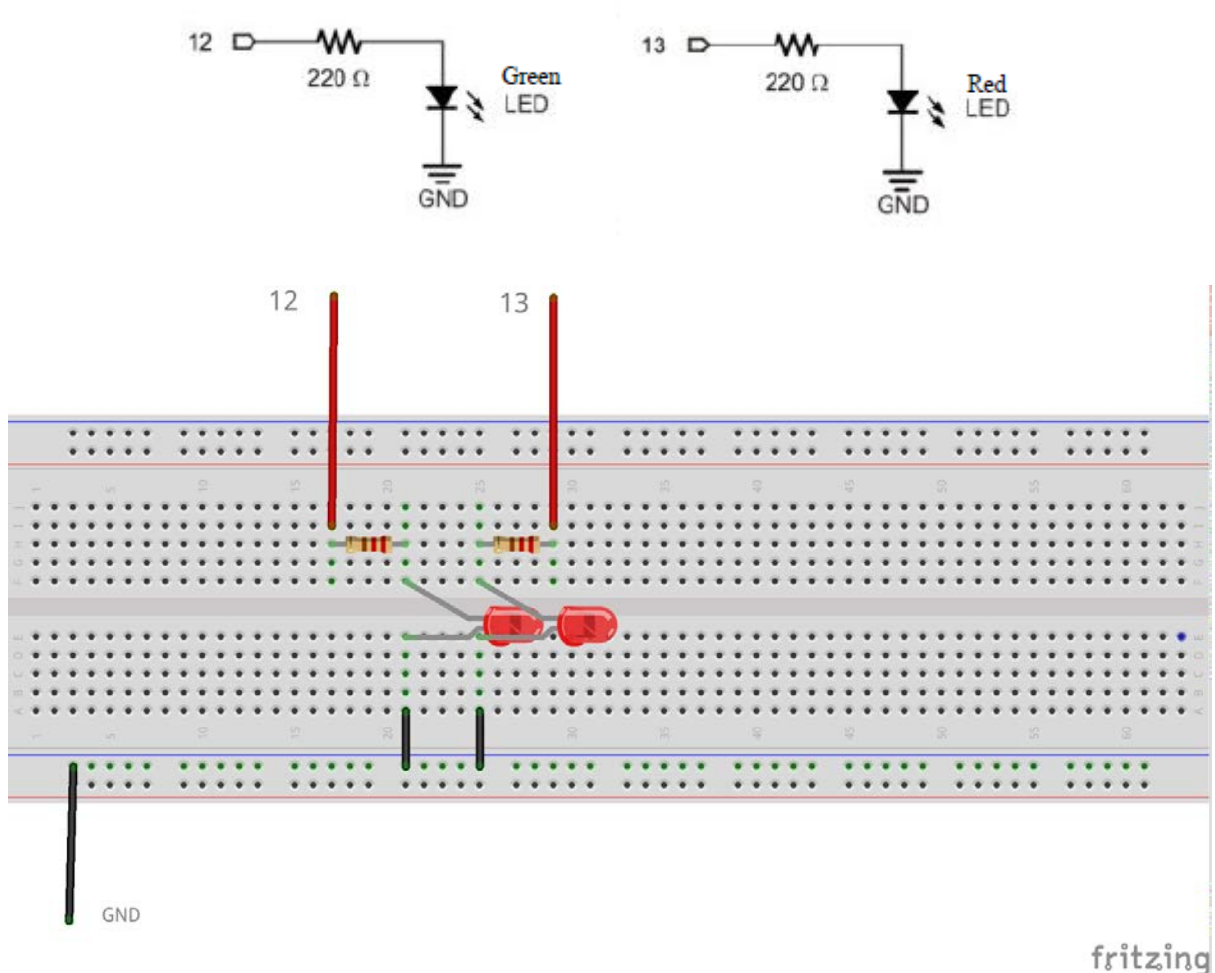


# Lab2 - Final Report

20100258 김성국

How are the LEDs connected?

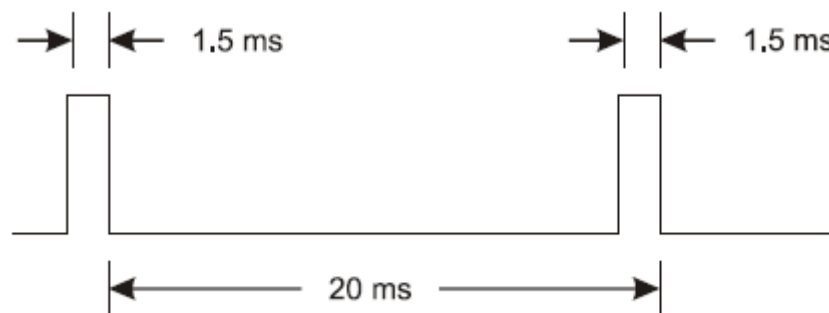


위와 같은 구조로 연결되어 있다. 12, 13 pin을 output으로 setting 하고 전압을 down시키기 위해 220Ohm의 저항을 연결한 후 cathode anode를 맞추어 연결한 후 접지하였다.

The constant 1500 is used for servo to stop

: `servoRight.writeMicroseconds(1500)`. What does the 1500 means?

Servo control은 단순히 pin을 high/low로 write 해주는 led와는 다르게 이루어 진다. 특정 주기마다 high signal을 주며, 이 high signal이 얼마나 지속되냐에 따라 motor를 돌리는 방향과 속도를 정한다. writeMicroseconds()의 parameter의 단위는 microsecond이며 이는 high signal을 얼마나 지속시킬 것이냐에 해당하는 값이다. 1500은 1500 microsecond, 즉 1.5ms를 의미하며 stay-still signal에 해당한다. 이는 시계 방향 최고속도라는 signal의 1.3ms와 반 시계 방향 최고속도라는 signal의 1.7ms의 중간 값이다.



## Assignment 2 - Source code

우선 servo library를 사용하기 위해 Servo.h를 include 한다. 또한, 양 쪽 servo에 하나 당 Servo를 declare 해주고(servoRight, servoLeft), setup() 맨 처음에서 각각 pin 12, 13을 attach해 준다.

Assignment 2는 계속해서 loop으로 돌리는 것이 아니라 한번만 실행할 것이므로 loop이 아닌 setup에서 계속해서 코딩을 한다.

로봇에게 go straight/backward, turn right/left을 지시하기 위해서는 각 명령마다 servo가 어떻게 돌아야 하는지를 생각해보아야 한다. 이는 아래 표로 정리가 된다.

명령	우측 서보	좌측 서보
Go straight	시계 방향	반 시계 방향
Go backward	반 시계 방향	시계 방향
Turn right	반 시계 방향	반 시계 방향
Turn left	시계 방향	시계 방향

위의 표를 writeMicroseconds의 argument 값들로 바꿔보면 다음과 같다.

명령	우측 서보	좌측 서보
Go straight	1300	1700
Go backward	1700	1300
Turn right	1700	1700
Turn left	1300	1300

코드는 단순히 위 표에 따라 코딩한 것이다. 시간을 조절하기 위해선 delay를 쓴다. Servo에 규칙적으로 신호를 주는 것은 writeMicroseconds이 아닌 background에서 돌아가는 다른 servo library 함수이다. writeMicroseconds는 servo를 직접 돌려주는 것이 아닌 servo에 가는 high signal의 길이만 변화시켜주는 것이므로 바로 return하며 다시 writeMicroseconds이 call이 되기까지는 현재 값이 유지가 된다.