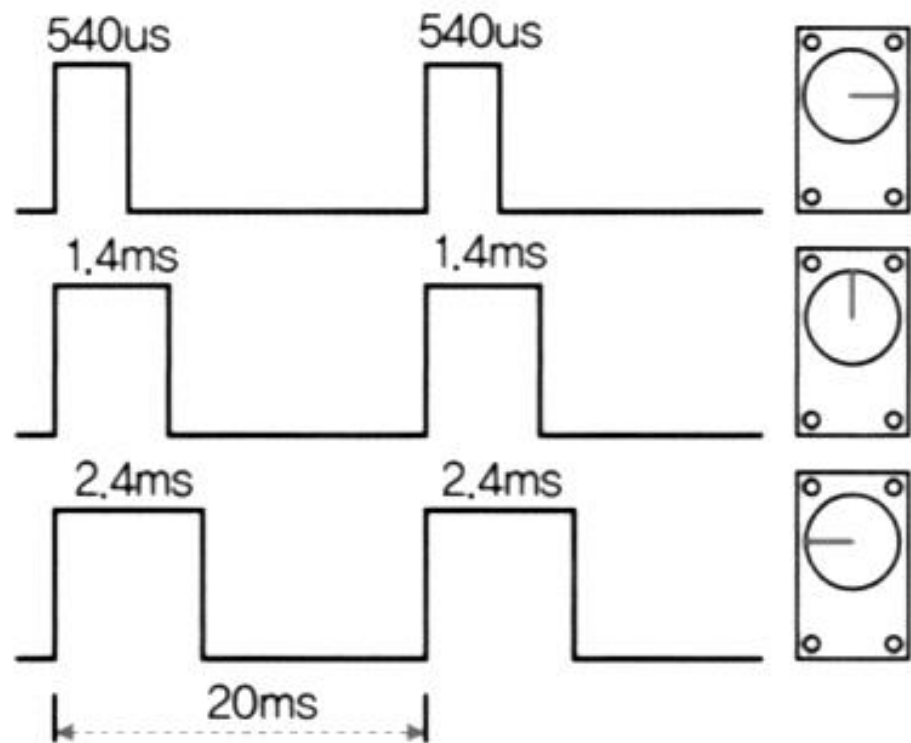

서보 모터

서보 모터

❖ 서보 모터



펄스에 따른 회전각

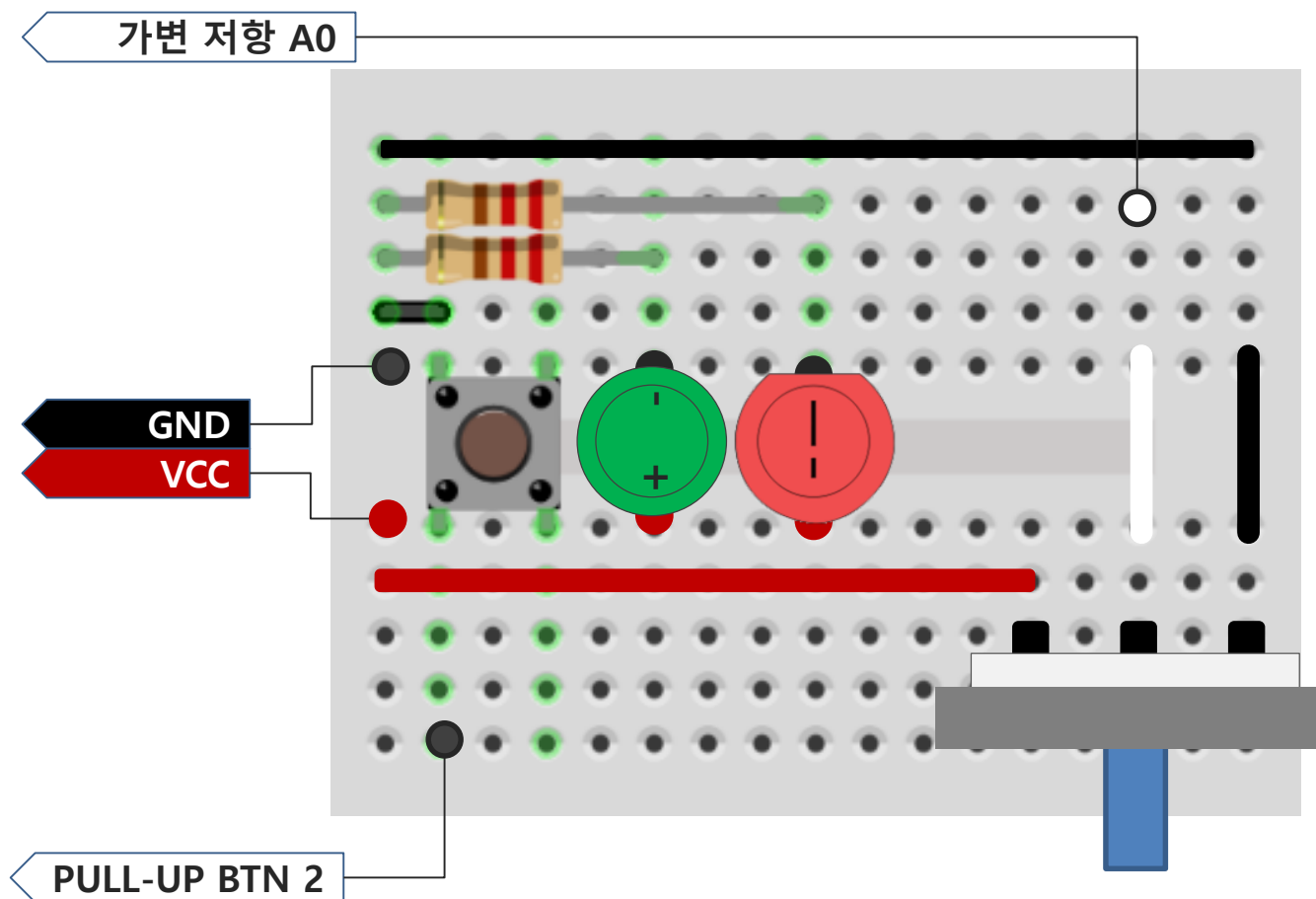
서보 모터

❖ 서보 모터의 특징

- 회전 각도 : 180도
- 무게 9g
- 크기 : 22.2 x 11.8 x 31mm
- 토크 : 1.8kgf · cm
- 동작 속도 : 0.3초/60도
- 동작전압 : 4.8~6V
- 신호 주기 : 10~20ms
- PWM 제어 펄스 : 0.5ms~2.4ms(제어각 : 0°~180°)

서보 모터

❖ 회로도



실습1: 서보 모터 회전 시키기(라이브러리 이용)

❖ Servo 객체

클래스 명 및 멤버 함수	설명	사용예
Servo 객체명;	Servo 타입의 객체(사용할 서보 모터명) 생성	Servo ms;
uint8_t attach(int pin);	객체(사용 서보 모터명)에 사용할 제어 핀 할당	ms.attach(9);
void detach();	객체에 할당된 핀을 해제한다.	ms.detach();
void write(int value);	<ul style="list-style-type: none">- 객체 회전 각도로 회전시키기- 200보다 작은 값이면 각도(0~180)- 200보다 크면 us단위의 펄스폭	ms.write(45);
int read();	마지막에 write()함수에서 사용한 값 읽기	int deg; deg=ms.read();
bool attached();	객체에 핀이 지정되어 있으면 true, 아니면 false	ms.attached();

실습2: 서보 모터 회전 시키기(라이브러리 이용)

❖ ex01/app.ino

```
#include <Servo.h>
#include <MiniCom.h>

MiniCom com;
Servo myServo;          // 사용할 서보 모터명 선언
const int servo_pin = 5; // 서보 모터 연결핀

void setup() {
    com.init();
    com.print(0, "Servo Test");
    myServo.attach(servo_pin); // 사용할 서보명에 핀 할당
}

void move_angle(int angle, int delay_time) {
    myServo.write(angle);
    com.print(1, "Angle:", angle);
    delay(delay_time);
}

void loop() {
    move_angle(0, 1000);
    move_angle(90, 1000);
    move_angle(180, 1000);
    move_angle(90, 1000);
}
```

실습2: 가변저항에 맞춰 서보모터 회전시키기

❖ ex02/app.ino

```
#include <Analog.h>
#include <Servo.h>

Analog a_value(A0, 0, 1023, 180, 0);

Servo myServo;
const int servo_pin = 5;

void setup() {
    myServo.attach(servo_pin);
}

void loop()
{
    int angle = a_value.read();
    myServo.write(angle);
    delay(100);
}
```

실습3: 가변저항에 맞춰 서보모터 회전시키기(MiniCom)

❖ ex03/app.ino

```
#include <Analog.h>
#include <Servo.h>
#include <MiniCom.h>

MiniCom com;
Analog a_value(A0, 0, 1023, 180, 0);
Servo myServo;
const int servo_pin = 5;

void check() {
    int angle = a_value.read();
    myServo.write(angle);
    com.print(1, "Angle:", angle);
}

void setup() {
    com.init();
    com.print(0, "Servo Test2");
    myServo.attach(servo_pin);
    com.setInterval(100, check);
}

void loop() {
    com.run();
}
```