

REV 1.0

# 자율주행 자동차 펌웨어관련 메뉴얼

혁신기술창업 훈련소

MakIST.Lab

Make Innovation Startup Training

대표 강 석 민  
인천광역시 미추홀구 염전로336 도읍빌딩 5층

010 - 2600 - 9977

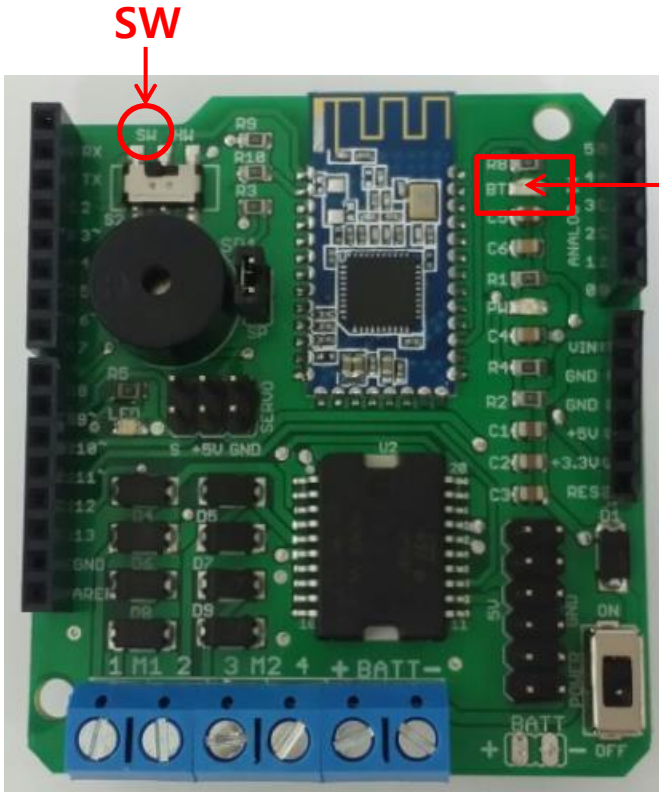
ksm@makist.net

## ◆ 목차

- 블루투스 이름 설정하기
- 초음파 센서 연결
- 초음파 센서 연결 테스트
- DC모터, 배터리 소켓 연결
- 배터리 체결
- 서보모터 구동축 고정
- 자율주행 펌웨어 업로드
- 모터 동작 방향 확인
- 자율주행하기

## ◆ 블루투스 이름 설정하기

- RC 실드의 블루투스 이름을 설정합니다.



감박임 : 접속해제  
ON : 접속중

```
#include <SoftwareSerial.h>
SoftwareSerial BTSerial(2, 4); // 2, 4번 핀에 꽂혀 있는 점프선은 제거
해 주세요
```

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

```
void loop() {
  if (BTSerial.available()) {
    Serial.write(BTSerial.read());
  }
  if (Serial.available()){
    BTSerial.write(Serial.read());
  }
}
```

```
/*
- 시리얼모니터 설정 :
  Both NL & CR 선택
  9600 보드레이트 선택
```

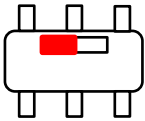
```
- AT+VERSION
- AT+NAME
- AT+NAME이름
```

```
*/
```

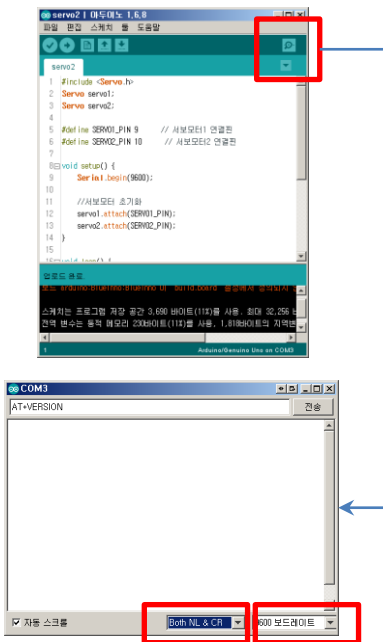
## ◆ 블루투스 이름 설정하기(계속)

- BLE연결상태 LED가 깜빡이고 있는지 확인합니다.
- LED가 깜빡이고 있으면 연결 해제된 상태이고 계속 켜져 있으면 연결된 상태입니다.
- 연결된 상태에서는 장치를 추가로 연결할 수 없습니다. 한번에 하나의 장치만 연결 가능합니다.

SW선택



- BLE시리얼통신 방식 선택 스위치는 “SW”를 선택합니다.
- 아두이노 스케치를 업로드 하기 위해서는 반드시 “SW”가 선택되어 있어야 합니다.
- “HW”가 선택되어 있으면 업로드할 수 없습니다.

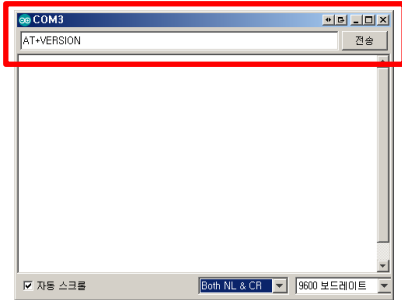


- 아두이노를 컴퓨터에 USB를 연결한 후 예제 소스를 업로드합니다.
- 시리얼모니터의 설정을 변경합니다.

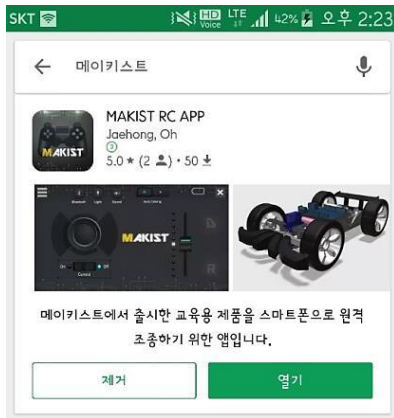
Both NL & CR 선택

9600 보드레이트 선택

## ◆ 블루투스 이름 설정하기(계속)



- 시리얼모니터를 이용하여 블루투스 이름을 설정합니다.  
'AT+VERSION' 입력 후 펌웨어 버전표시 확인  
'AT+NAME이름' 입력  
예) MAKIST 이름으로 설정할 경우 : AT+NAMEMAKIST  
'AT+NAME' 입력 후 정상적으로 설정 되었는지 확인합니다.
- ※ 이름은 영문자, 숫자, \_ 등 입력가능하고, 최소 6자리 이상 입력합니다.  
※ AT 명령어는 BLE모듈의 펌웨어 버전에 따라 다를 수 있습니다.  
위의 방법으로 설정이 되지 않을 경우 메이키스트 블로그, 또는 카페의  
“메이키스트RC 쉴드 블루투스 이름설정 방법” 을 참조하시기 바랍니다.



- 안드로이드 구글플레이, iOS 앱스토어에서 “메이키스트”, 또는 “makist” 키워드로 검색 후 앱을 다운로드 합니다.

## ◆ 블루투스 이름 설정하기(계속)



- 블루투스 장치를 검색합니다.

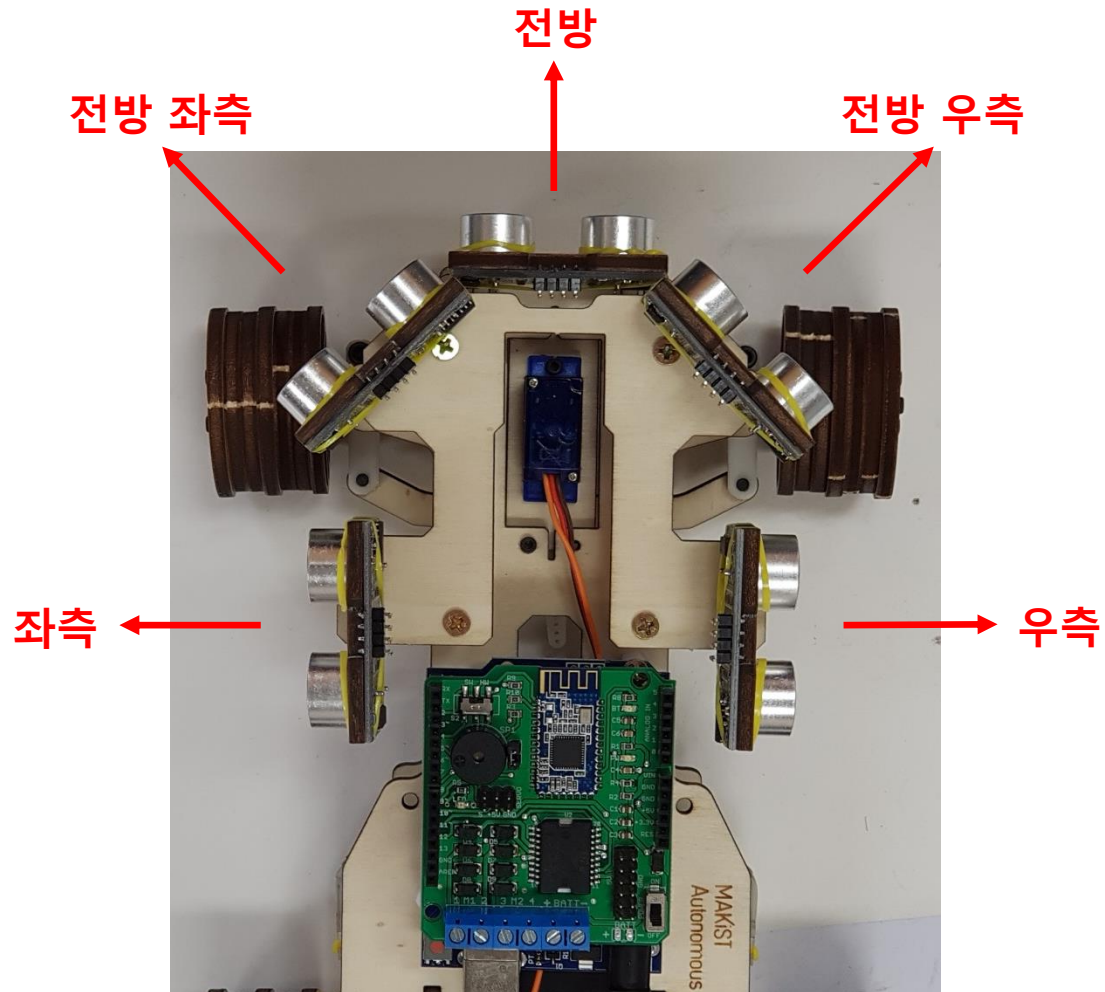


- 블루투스 장치 선택 후 OK 버튼을 누릅니다.
- 블루투스 장치에 연결이 되면 메이키스트 RC실드의 BLE연결상태 LED가 깜빡이다가 계속 켜져 있게 됩니다.

※ 블루투스 장치에 연결이 잘 되지 않을 경우 앱을 재시작한 후 시도 하거나, RC실드의 전원을 껏다가 켜 후 다시 시도해 보시기 바랍니다.

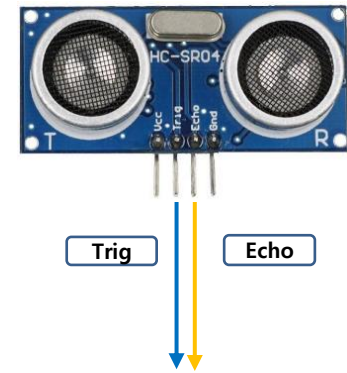
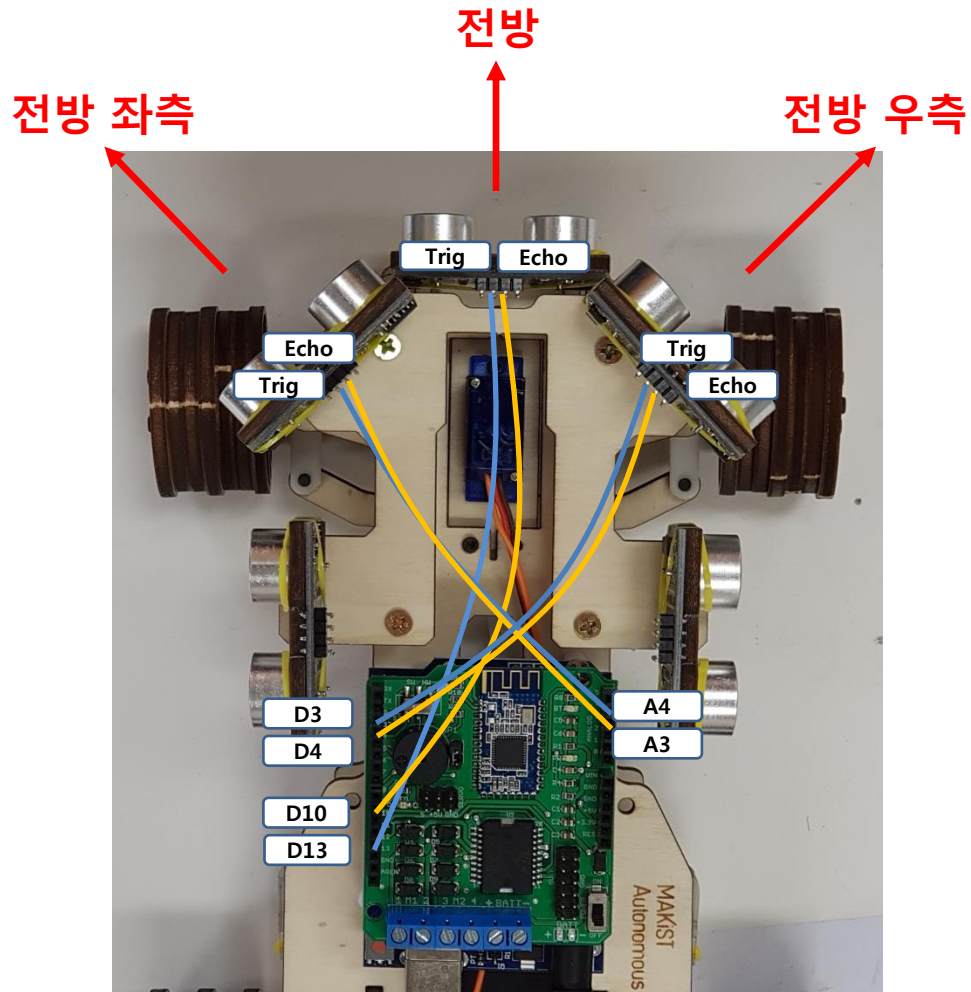
## ◆ 초음파 센서 연결

- 초음파 센서의 명칭



## ◆ 초음파 센서 연결

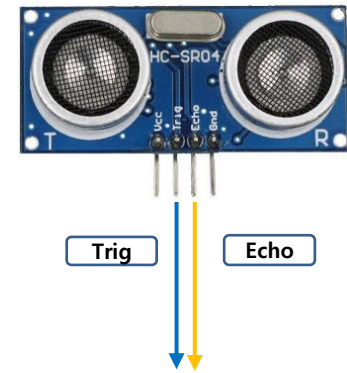
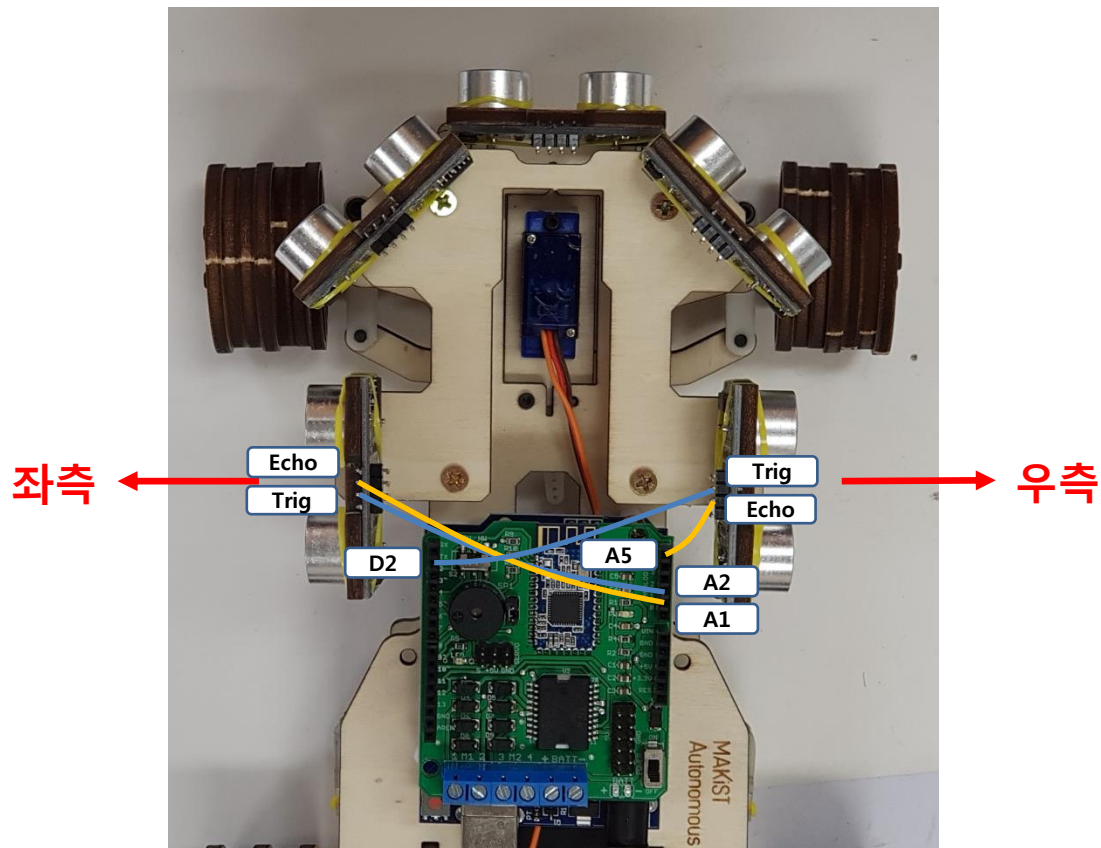
- 전방 초음파 Trig, Echo 연결





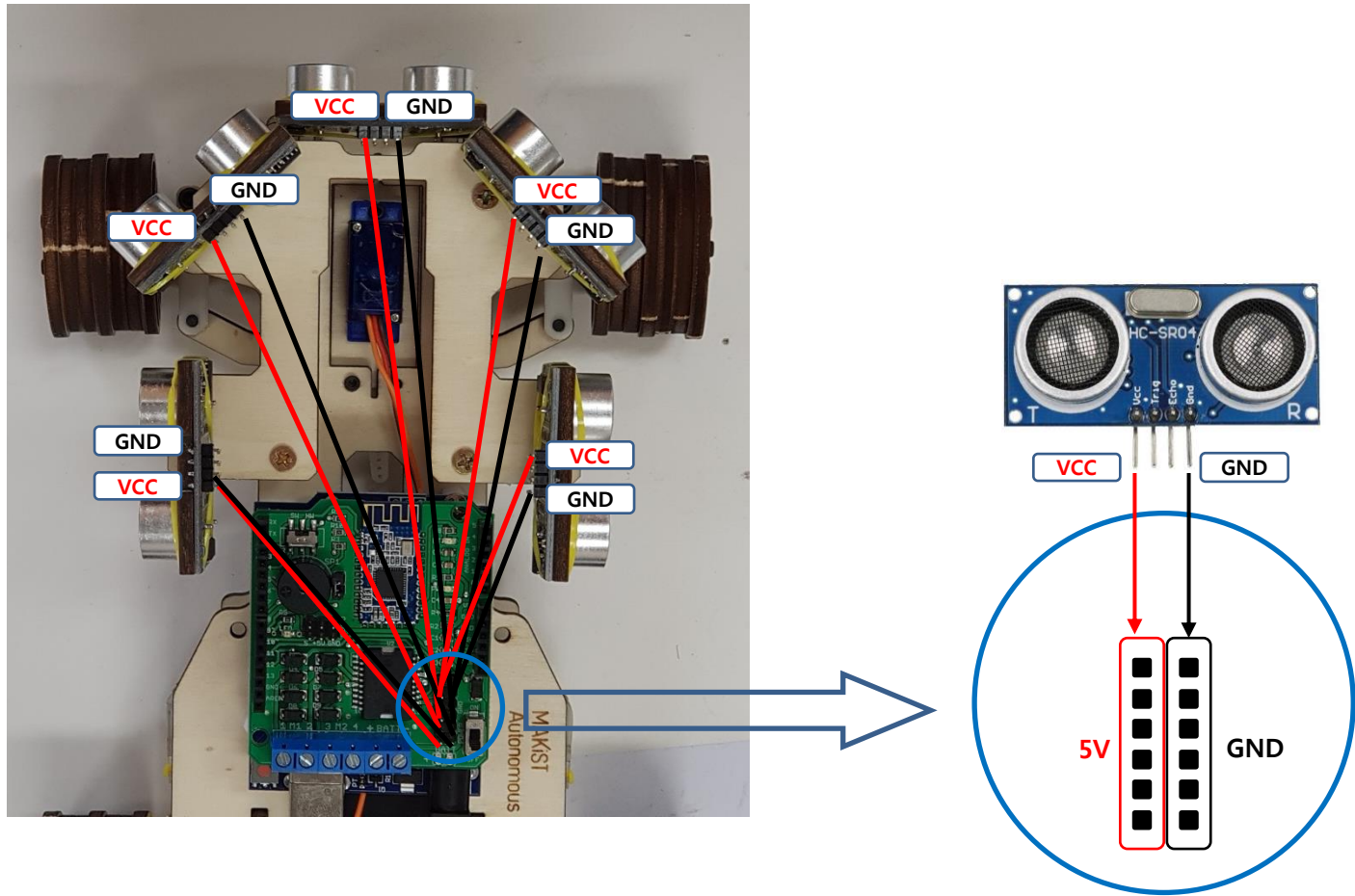
## ◆ 초음파 센서 연결

- 좌측/우측 초음파 Trig, Echo 연결



## ◆ 초음파 센서 연결

- VCC/GND 연결

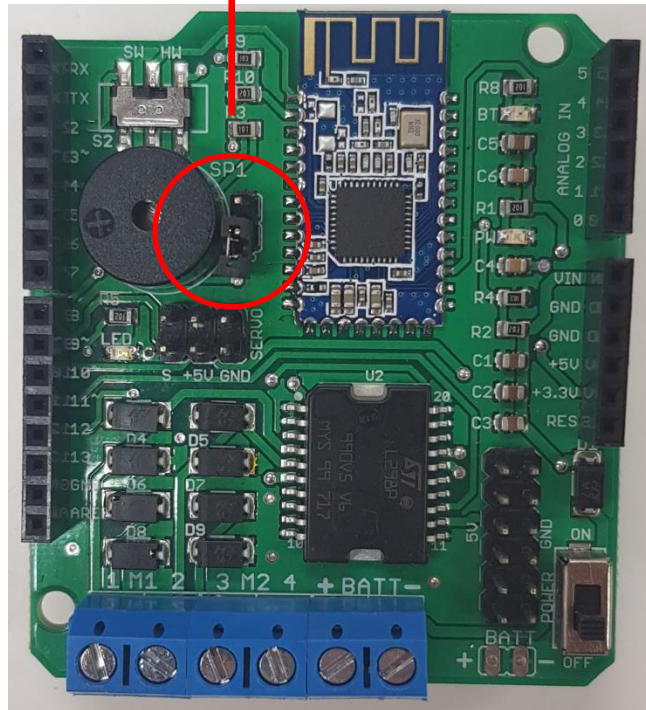


## ◆ 초음파 센서 연결

- 피에조 부저 점퍼캡 설정

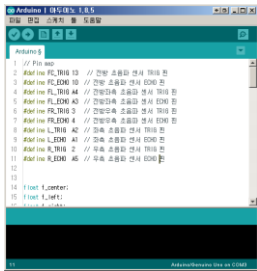


점퍼캡을 그림과 같이 연결하여  
피에조 부저 사용하지 않음으로 설정  
(사용할 경우 초음파 센서와 충돌)

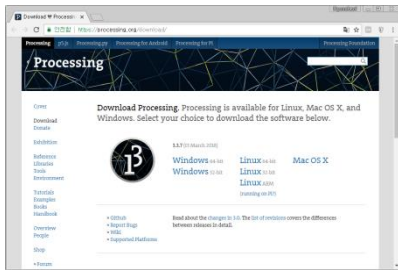


## ◆ 초음파 센서 연결 테스트

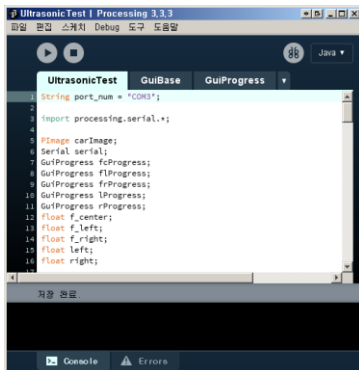
- Processing을 이용해 초음파 센서가 정상적으로 연결 되었는지 확인합니다.



- 아두이노에 USB케이블을 연결한 후 초음파센서 테스트용 펌웨어를 업로드합니다.
- 소스위치 : /UltrasonicTest/Arduino/Arduino.ino



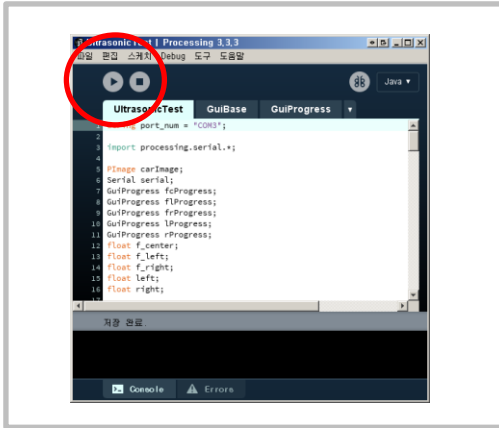
- Processing을 다운로드 후 설치합니다.
- <https://processing.org/download/>



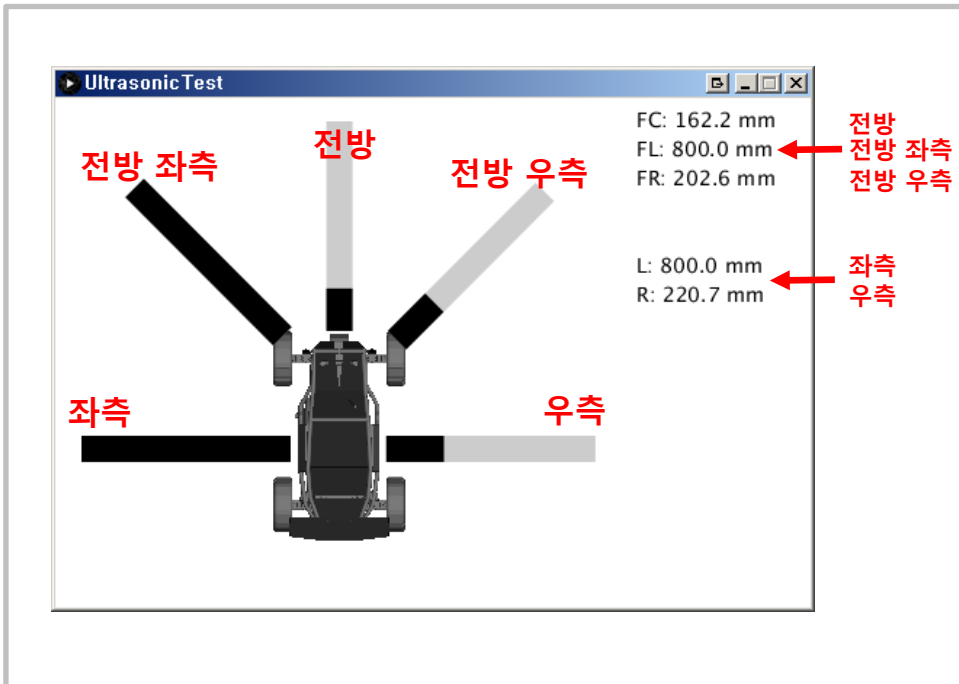
- 초음파센서 테스트용 Processing 파일을 엽니다.
- 소스위치 : /UltrasonicTest/ UltrasonicTest.pde
- 소스의 첫번째 라인의 포트번호를 아두이노가 연결되어 있는 포트로 수정합니다.
- 예) 아두이노가 COM3으로 연결되어 있을 때  
String port\_num = "COM3";

## ◆ 초음파 센서 연결 테스트(계속)

- Processing을 이용해 초음파 센서가 정상적으로 연결 되었는지 확인합니다.



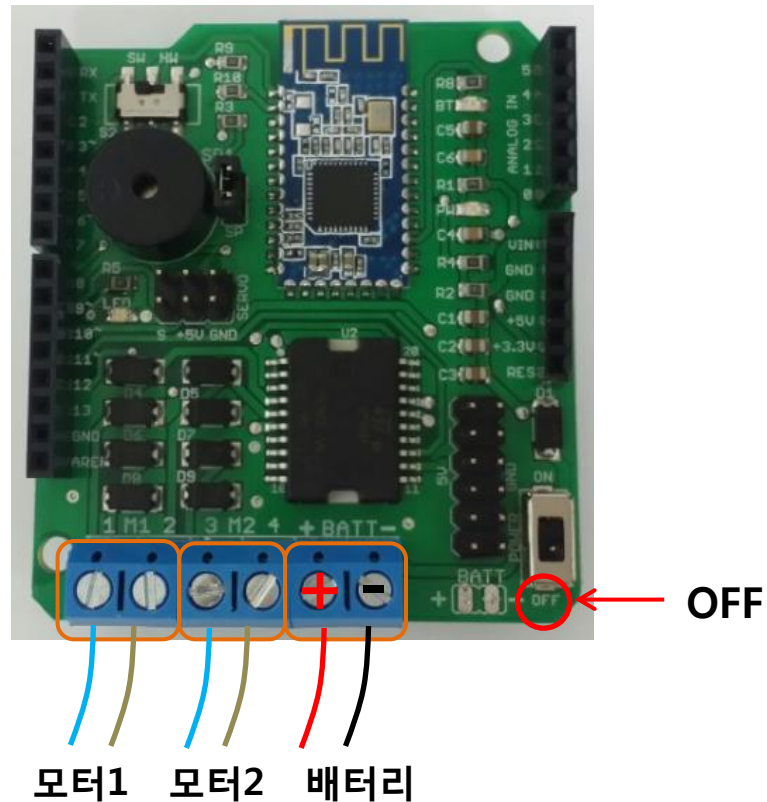
- 실행 버튼을 클릭합니다.



- 초음파 센서 앞에 물체로 가리면 거리가 측정되어 화면에 보여집니다.
- 왼쪽의 그림은 전방, 전방우측, 우측 초음파 센서에서 물체가 감지된 모습입니다. 감지된 거리를 검은색막대 형태로 보여 주고 거리도 함께 표시합니다.
- 전방좌측, 좌측은 물체가 감지되지 않은 모습입니다. 물체가 감지 되지 않을 경우 "초음파 센서 연결" 항목을 참조하여 점프선 연결을 점검해 주시기 바랍니다.

## ◆ DC모터, 배터리 소켓 연결

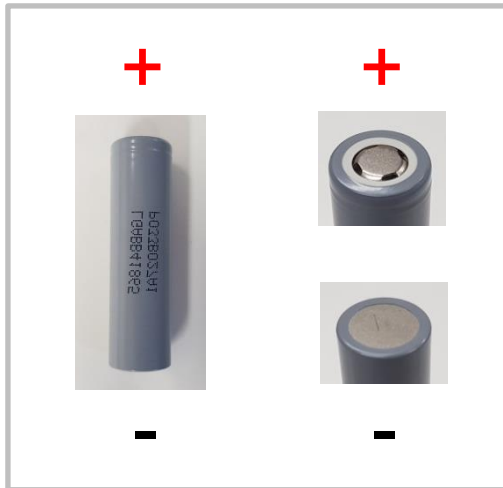
- DC모터 2개와 배터리 소켓을 RC섀드에 연결합니다.
- RC섀드 전원 스위치는 OFF 상태로 설정합니다.



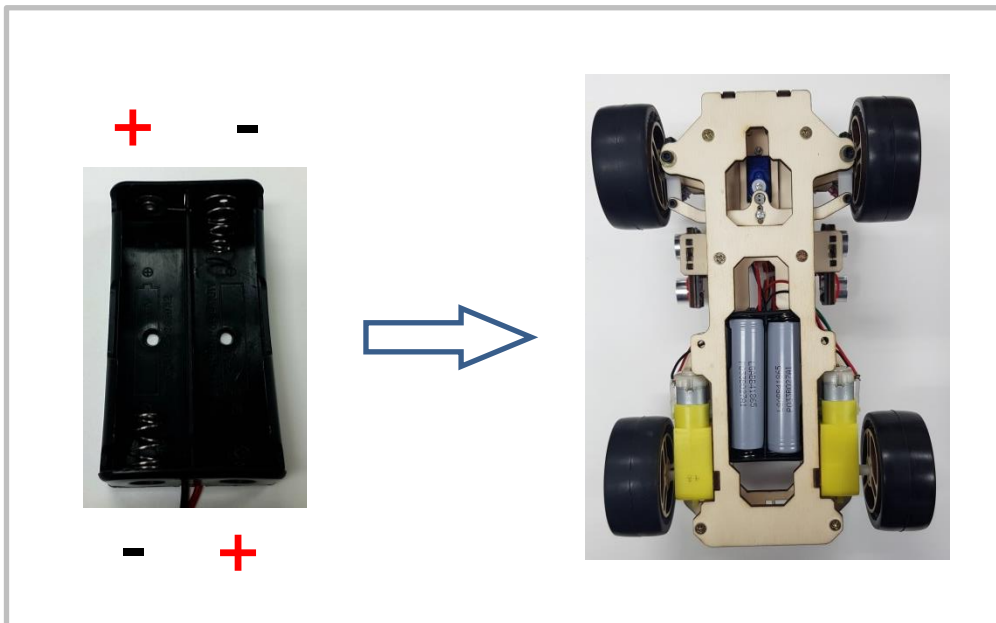
※ 주의 : 배터리 소켓의 빨간색을 +에, 검정색을 -에 연결  
모터1, 모터2는 극성에 상관 없이 연결합니다.

## ◆ 배터리 체결

• 배터리를 자동차의 하단부의 배터리 소켓에 체결합니다.



- 배터리의 극성(+, -)을 확인합니다.
- 배터리의 양끝의 모양으로 극성을 확인할 수 있습니다.
- +극은 좁고 약간 볼록하게, -극은 넓고 평평합니다.

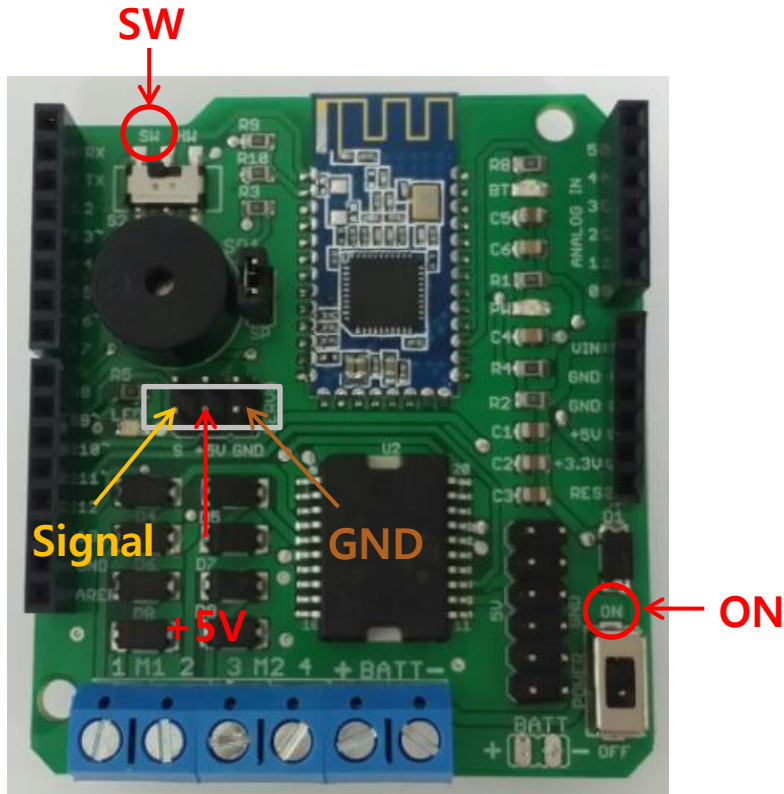


- 자동차의 하단부 배터리 소켓에 극성에 맞게 배터리를 체결합니다.



## ◆ 서보모터 구동축 고정

- 서보모터를 90도로 설정한 후 서보모터 구동축을 고정합니다.



```
#include <Servo.h>
Servo servo;

#define SERVO1_PIN 9    // 서보모터1 연결핀

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

    //서보모터 초기화
    servo.attach(SERVO1_PIN);

    delay(100);

    servo.write(60);      // 왼쪽 이동(정상동작 확인용 코드)
    delay(500);
    servo.write(120);     // 오른쪽 이동(정상동작 확인용 코드)
    delay(500);
    servo.write(90);      // 가운데 90도 설정

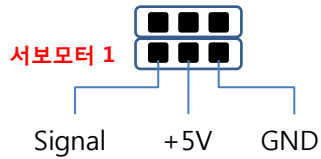
    delay(100);

    servo.detach();
}

void loop() {
}
```

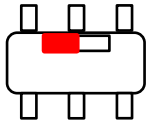


## ◆ 서보모터 구동축 고정(계속)



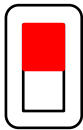
- 서보모터는 서보모터1 핀에 왼쪽부터 신호선, +5V, GND 순서로 연결합니다.

SW선택



- BLE시리얼통신 방식 선택 스위치는 "SW"를 선택합니다.
- 아두이노 스케치를 업로드하기 위해서는 반드시 "SW"가 선택되어 있어야 합니다.
- "HW"가 선택되어 있으면 업로드할 수 없습니다.

ON



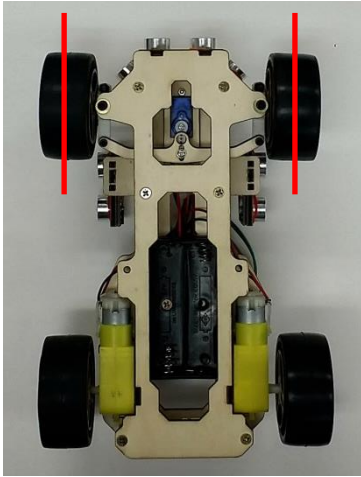
OFF

- RC셸드의 전원 스위치를 ON으로 옮깁니다.

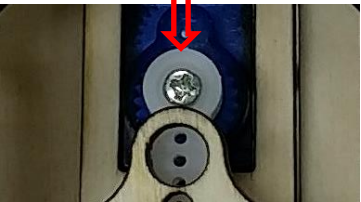
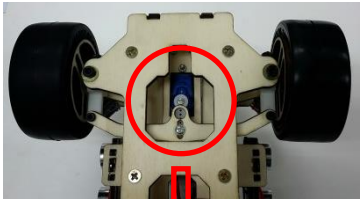


- 아두이노를 컴퓨터에 USB 케이블로 연결한 후 소스를 업로드합니다.
- 펌웨어 업로드가 완료 되면 서보모터는 60도에서 120도로 이동 후 90도로 이동합니다.
- 서보모터가 동작 되지 않으면 서보모터와 RC셸드의 연결핀을 확인 하시거나 펌웨어를 다시 업로드 해 보시기 바랍니다.

## ◆ 서보모터 구동축 고정(계속)



- 자동차를 뒤집어 하단부가 보이게 하고 앞바퀴를 몸체와 나란히 조절합니다.

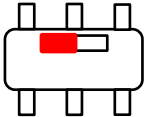


- 서보모터에 동봉되어 있던 나사로 서보구동축을 단단히 체결합니다.
- 체결이 완료 되면 다음 작업을 위해 RC실드의 전원 스위치를 OFF로 옮겨 줍니다.

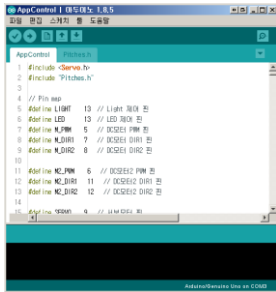
## ◆ 자율주행 펌웨어 업로드

- 자율주행 펌웨어를 업로드 합니다.

SW선택



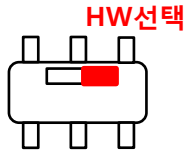
- BLE시리얼통신 방식 선택 스위치는 "SW"를 선택합니다.
- 아두이노 스케치를 업로드 하기 위해서는 반드시 "SW"가 선택되어 있어야 합니다.



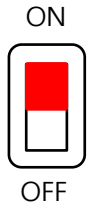
- 아두이노에 USB케이블을 연결한 후 자율주행 펌웨어를 업로드합니다.
- 소스위치 : /AppControl/AppControl.ino

## ◆ 모터 동작 방향 확인

- RC 앱을 이용해 자율주행 자동차와 연결한 후 모터의 동작방향을 확인합니다.



- BLE시리얼통신 방식 선택 스위치는 "HW"를 선택합니다.
- "HW"는 RC앱으로 컨트롤시 사용합니다.



- RC설드의 전원 스위치를 ON으로 옮깁니다.



- 블루투스 장치를 검색합니다.

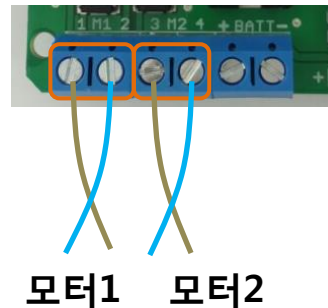
## ◆ 모터 동작 방향 확인(계속)

- RC 앱을 이용해 자율주행 자동차와 연결한 후 모터의 동작방향을 확인합니다..



- 블루투스 장치 선택 후 OK 버튼을 누릅니다.
- 블루투스 장치에 연결이 되면 메이키스트 RC실드의 BLE연결상태 LED가 깜빡이다가 계속 켜져 있게 됩니다.

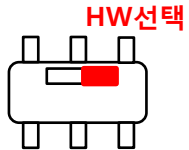
※ 블루투스 장치에 연결이 잘 되지 않을 경우 앱을 재시작한 후 시도 하거나, RC실드의 전원을 껏다가 켜 후 다시 시도해 보시기 바랍니다.



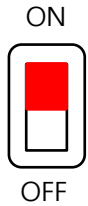
- 속도제어 레버를 전진으로 움직여 봅니다.
- 모터가 주행방향으로 회전하는지 확인하고 반대방향으로 회전하는 모터는 RC실드의 터미널 블록에 연결된 두가닥의 모터선을 그림과 같이 반대로 체결합니다.

## ◆ 자율주행하기

- RC 앱을 이용해 자율주행을 합니다.



- BLE시리얼통신 방식 선택 스위치는 "HW"를 선택합니다.
- "HW"는 RC앱으로 컨트롤시 사용합니다.



- RC설드의 전원 스위치를 ON으로 옮깁니다.



- 시작 버튼을 눌러 자율주행을 시작합니다.
- 정지 버튼을 누르면 자율주행 모드가 해제되고 정지합니다.
- 자율주행 모드시 속도와 방향은 수동으로 제어할 수 없습니다.