



연세대학교  
Yonsei Where we make History

---

# Maker Space Lecture

## 2 . Motor Control

---



YONSEI, where we make *history*

연세대학교 기계공학과 강인구 (INGU KANG)  
kynk94@naver.com

<https://bit.ly/33tmFdH>



연세대학교  
Yonsei Where we make History



YONSEI, where we make *history*

## CONTENTS

1. 함수에 대해서
2. 모터의 종류
3. 서보모터 사용하기
4. DC모터 사용하기

# 함수(function)란 무엇인가?

```
type name(인수 목록)
{
    함수의 내용

    return 결과값;
}
```

type	함수가 실행 후 어떤 자료형을 반환할지 결정한다.
name	함수를 호출할 이름 Built In 함수 외에 자유롭게 설정
인수	함수가 작업에 사용할 값 $y=f(x)$ 에서 $x$ 에 해당한다.
내용	함수가 호출되면 무조건 실행되는 함수의 몸체



# 함수(function)란 무엇인가?

```
type name(인수 목록)
{
    함수의 내용

    return 결과값;
}
```

type      int, float, char 등의 일반적인 자료형  
          및 void 를 사용한다.

여기서 void는 반환할 값이 없음을  
의미함.

return    함수 실행 후 원하는 값을 반환  
          type과 같은 자료형 이어야 함

type이 void일 때에는  
return을 생략하거나  
return; 을 사용함.



## Practice 1 – 함수 사용하기

```
void setup() {  
  Serial.begin(9600);  
  Serial.println("Start");  
}  
  
int a = 0;  
int b = 0;  
  
void loop() {  
  if(Serial.available()){  
    a = Serial.parseInt();  
    b = Serial.parseInt();  
    int c = Sum(a,b);  
    Serial.println(c);  
  }  
}  
  
int Sum(int x1, int x2){  
  return x1+x2;  
}
```

두 개의 값을 받아  
더해주는 함수 Sum 작성

Serial 창을 통해 2개의 값을 입력하고  
Sum 을 실행한다.



1  
2  
3  
4

## Practice 1 – 함수 사용하기

```
sketch_may04a $  
  
void setup() {  
  Serial.begin(9600);  
  Serial.println("Start");  
}  
  
int a = 0;  
int b = 0;  
int d = 0;  
void loop() {  
  if(Serial.available()){  
    a = Serial.parseInt();  
    b = Serial.parseInt();  
    int c =Sum(a,b);  
    Serial.println(c);  
    Serial.println(d);  
  }  
}  
  
int Sum(int x1, int x2){  
  d = x1*x2;  
  return x1+x2;  
}
```

함수 밖에서 변수 d를 선언

Sum 함수 내에서  
인자 2개의 곱을 d에 할당한다.

이 때의 결과는?



## Practice 1 – 함수 사용하기

```
sketch_may04a $  
  
void setup() {  
  Serial.begin(9600);  
  Serial.println("Start");  
}  
  
int a = 0;  
int b = 0;  
int d = 0;  
void loop() {  
  if(Serial.available()){  
    a = Serial.parseInt();  
    b = Serial.parseInt();  
    int c = Sum(a,b);  
    Serial.println(c);  
    Serial.println(d);  
  }  
}  
  
int Sum(int x1, int x2){  
  d = x1*x2;  
  return x1+x2;  
}
```

함수 밖에서 변수 d를 선언

Sum 함수 내에서  
인자 2개의 곱을 d에 할당한다.

이 때의 결과는?

함수의 내용은 모두 실행된다.





1  
2  
3  
4

## Practice 1 – 함수 사용하기

```
sketch_may04a $  
  
void setup() {  
  Serial.begin(9600);  
  Serial.println("Start");  
}  
  
int a = 0;  
int b = 0;  
int d = 0;  
void loop() {  
  if(Serial.available()){  
    a = Serial.parseInt();  
    b = Serial.parseInt();  
    int c =Sum(a,b);  
    Serial.println(c);  
    Serial.println(d);  
  }  
}  
  
int Sum(int x1, int x2){  
  int d = x1*x2;  
  return x1+x2;  
}
```

함수 밖에서 변수 d를 선언

Sum 함수 내에서  
d 앞에 int 를 붙인다.

이 때의 결과는?



## Practice 1 – 함수 사용하기

```
sketch_may04a $  
  
void setup() {  
  Serial.begin(9600);  
  Serial.println("Start");  
}  
  
int a = 0;  
int b = 0;  
int d = 0;  
void loop() {  
  if(Serial.available()){  
    a = Serial.parseInt();  
    b = Serial.parseInt();  
    int c =Sum(a,b);  
    Serial.println(c);  
    Serial.println(d);  
  }  
}  
  
int Sum(int x1, int x2){  
  int d = x1*x2;  
  return x1+x2;  
}
```

함수 안에서 선언한 변수는  
함수 안에서만 사용 가능하다.

변수가 이름이 똑같다면  
함수 안에서의 결과가  
함수 밖에 영향을 미치지 않는다.

이를 '지역변수'라고 한다.

Ps) 함수 밖의 변수 = '전역변수'



1  
2  
3  
4

## 모터의 종류

모터

DC모터 (Brushed, Brushless)

Servo모터

Stepper모터 (=Step 모터)

초음파 모터, 보이스코일 등...

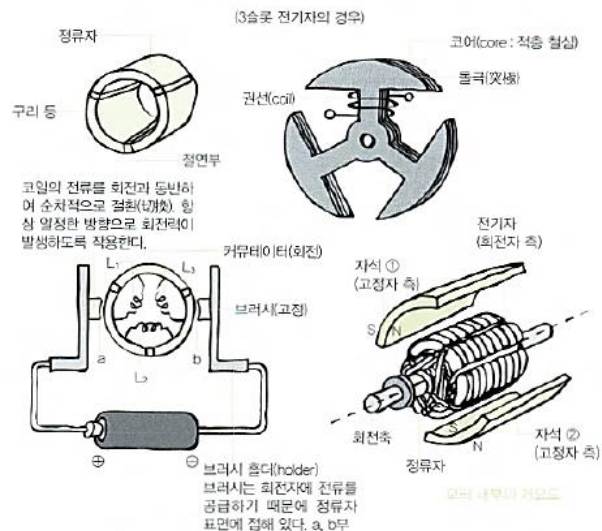
흔히 말하는 모터

1  
2  
3  
4

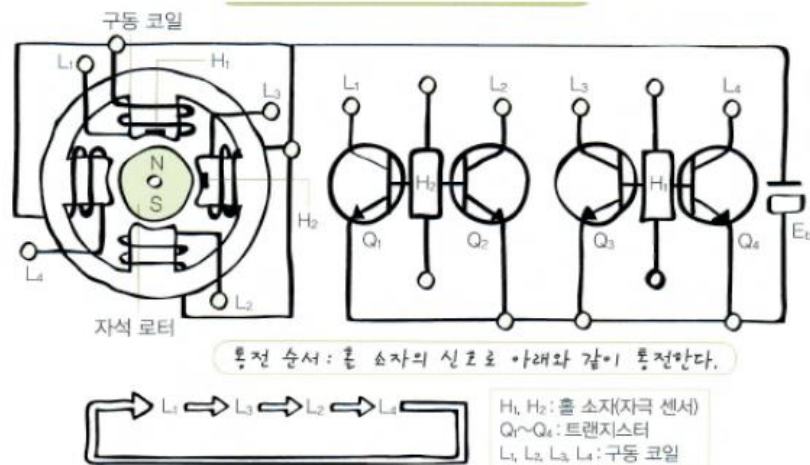
## 모터의 종류

### DC 모터

[그림 1] 전기자의 구조



Brushed



Brushless

1  
2  
3  
4

## 모터의 종류

### Servo 모터



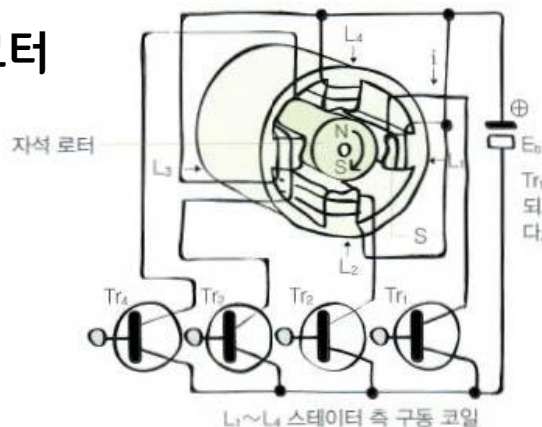
모터 안에 Controller가 내장되어  
PWM신호에 따라 특정 위치로  
이동할 수 있는 모터

In Arduino  
Servo.h 라이브러리를 사용하여  
각도를 입력하면, 원하는 각도로  
움직인다.

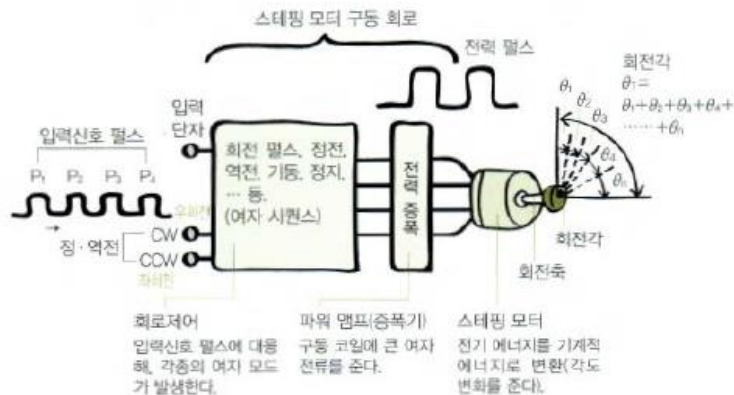
2

## 모터의 종류

# Stepper 모터



Tr ON일 때 L의 내측에 S극이 되고, 로터의 N극을 끌어 당긴다. 이때 90°로 로터가 회전한다.



1  
2  
3  
4

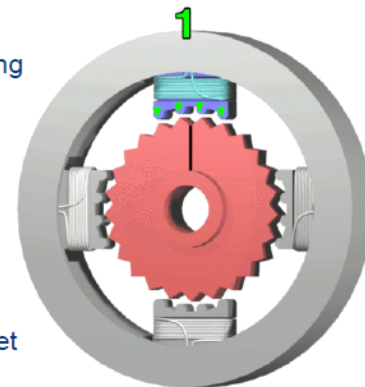
## 모터의 종류

### Stepper 모터

(참고만 하세요)

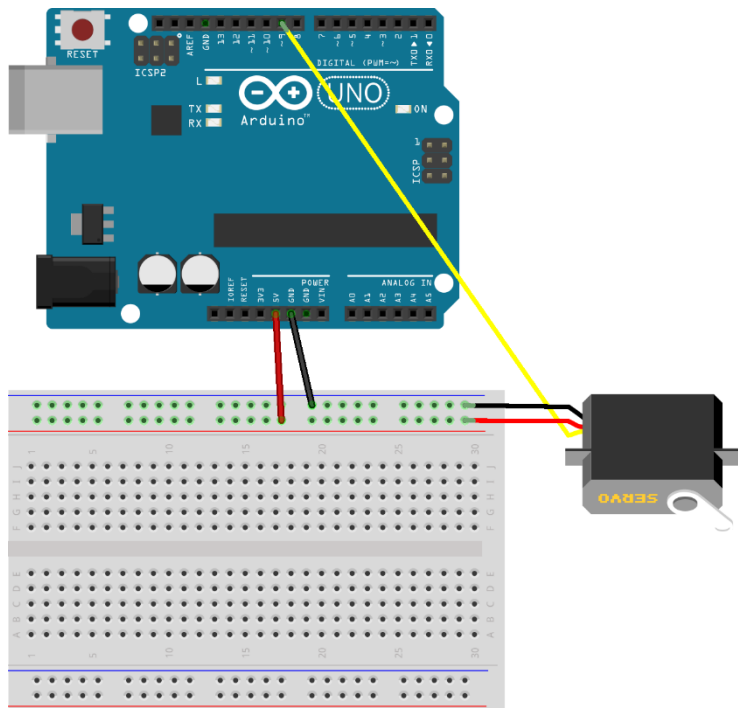
#### Animation of a simplified stepper motor

- Frame 1: The top electromagnet (1) is turned on, attracting the nearest teeth of the gear-shaped iron rotor. With the teeth aligned to electromagnet 1, they will be slightly offset from electromagnet 2.
- Frame 2: The top electromagnet (1) is turned off, and the right electromagnet (2) is energized, pulling the teeth into alignment with it. This results in a rotation of  $3.6^\circ$  in this example.
- Frame 3: The bottom electromagnet (3) is energized; another  $3.6^\circ$  rotation occurs.
- Frame 4: The left electromagnet (4) is energized, rotating again by  $3.6^\circ$ . When the top electromagnet (1) is again enabled, the rotor will have rotated by one tooth position; since there are 25 teeth, it will take 100 steps to make a full rotation in this example.



<https://youtu.be/eyqwLiowZiU>

## Practice 2 – Servo Motor



fritzing

```
#include <Servo.h>
#define servopin 9
```

라이브러리(헤더)

서보모터를 쉽게 사용하기 위해  
아두이노에 내장되어있는  
함수의 덩어리

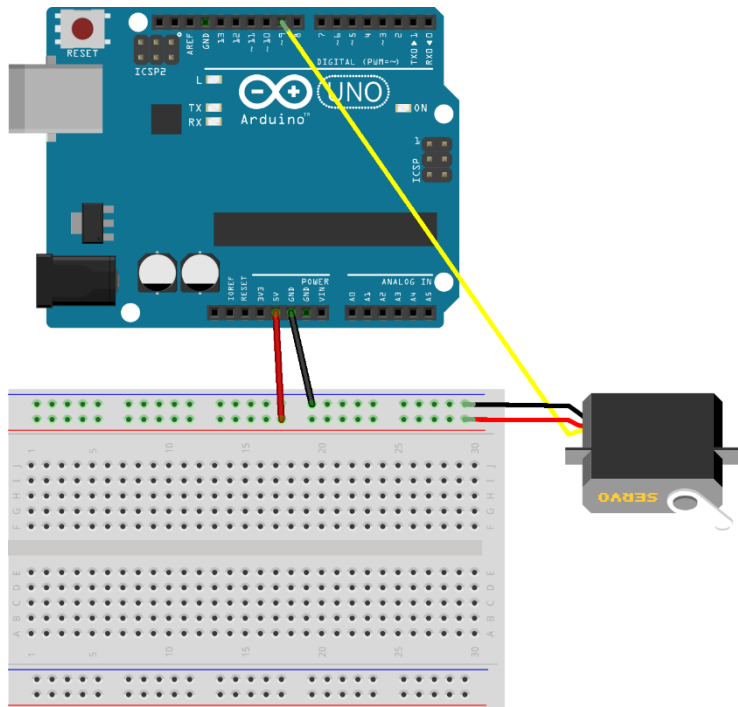
```
Servo servomotor;
```

```
void setup() {
    servomotor.attach(servopin);
    Serial.begin(9600);
    Serial.println("Start");
}
```

```
void loop() {
    servomotor.write(0);
    delay(1000);
    servomotor.write(90);
    delay(1000);
    servomotor.write(180);
    delay(1000);
}
```



# Practice 3 – Servo Motor



fritzing

```
#include <Servo.h>
#define servopin 9

Servo servomotor;

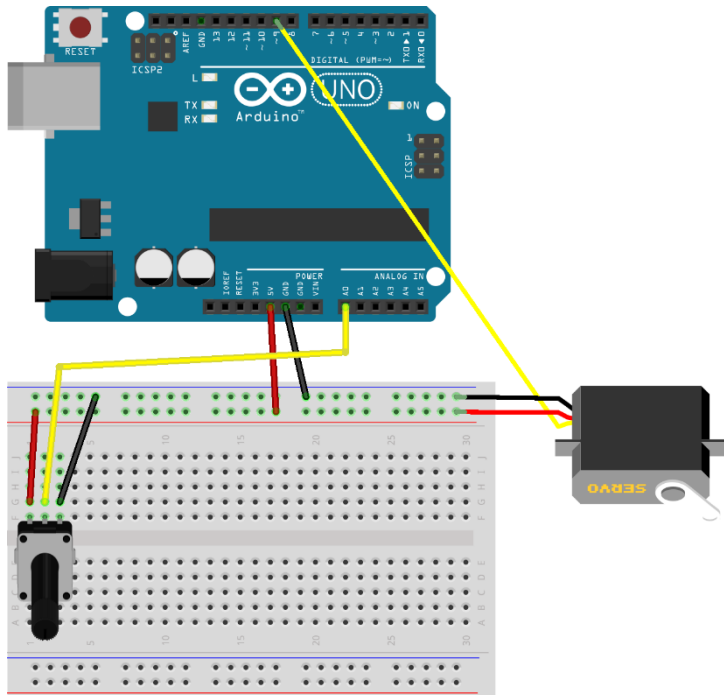
void setup() {
    servomotor.attach(servopin);
    Serial.begin(9600);
    Serial.println("Start");
}

int angle=0;

void loop() {
    if(Serial.available()){
        angle = Serial.parseInt();
    }
    servomotor.write(angle);
}
```

1  
2  
3  
4

## Practice 4 – Servo Motor & Potentiometer



fritzing

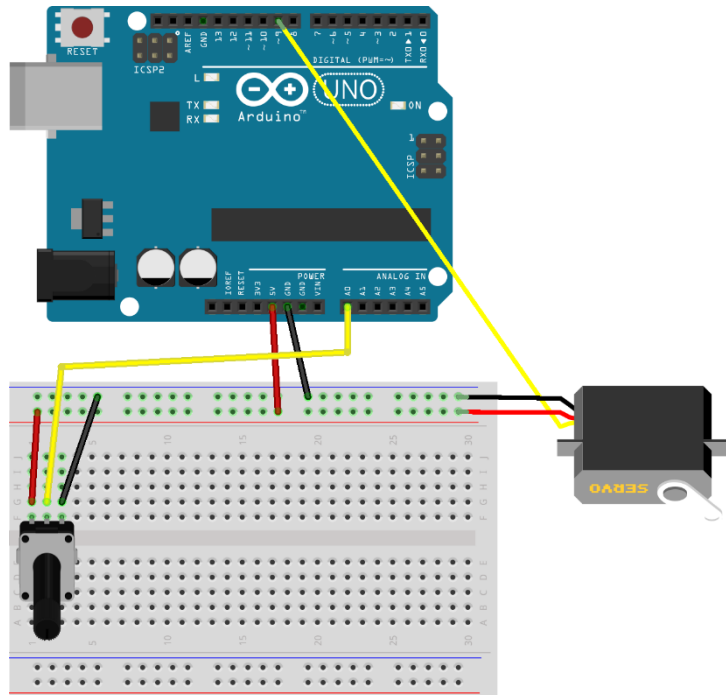
Potentiometer 가변저항

회전을 통해 저항의 세기를 조절하며  
전압의 세기를 바꿔  
아날로그 출력을 제공한다.

Analog Input으로 받아들여  
모터를 제어할 수 있다.

1  
2  
3  
4

## Practice 4 – Servo Motor & Potentiometer



fritzing

```
#include <Servo.h>
#define servopin 9
#define potentio A0

Servo servomotor;

void setup() {
    servomotor.attach(servopin);
    pinMode(potentio, INPUT);

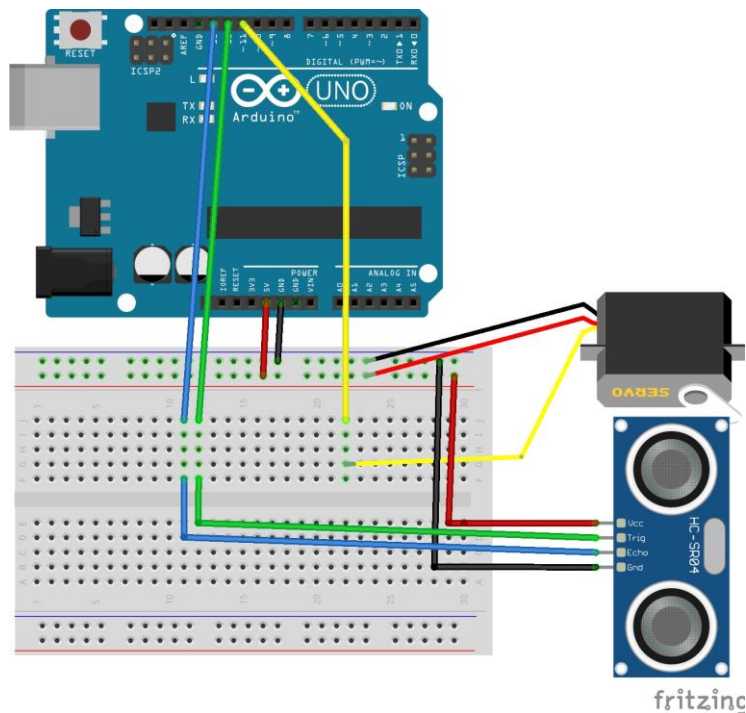
    Serial.begin(9600);
    Serial.println("Start");
}

int degree = 0;

void loop() {
    degree = map(analogRead(potentio), 0, 1023, 0, 180);
    Serial.print("Potentiometer : \t");
    Serial.print(analogRead(potentio));
    Serial.print("\tDegree : \t");
    Serial.println(degree);
    servomotor.write(degree);
}
```

1  
2  
3  
4

## Practice 5 – Ultrasonic Sensor



### 5\_Ultrasonic\_Sensor

```
#define trig 12
#define echo 13

void setup() {
    Serial.begin(9600);
    pinMode(trig, OUTPUT);
    pinMode(echo, INPUT);

    digitalWrite(trig, LOW);
    digitalWrite(echo, LOW);
}

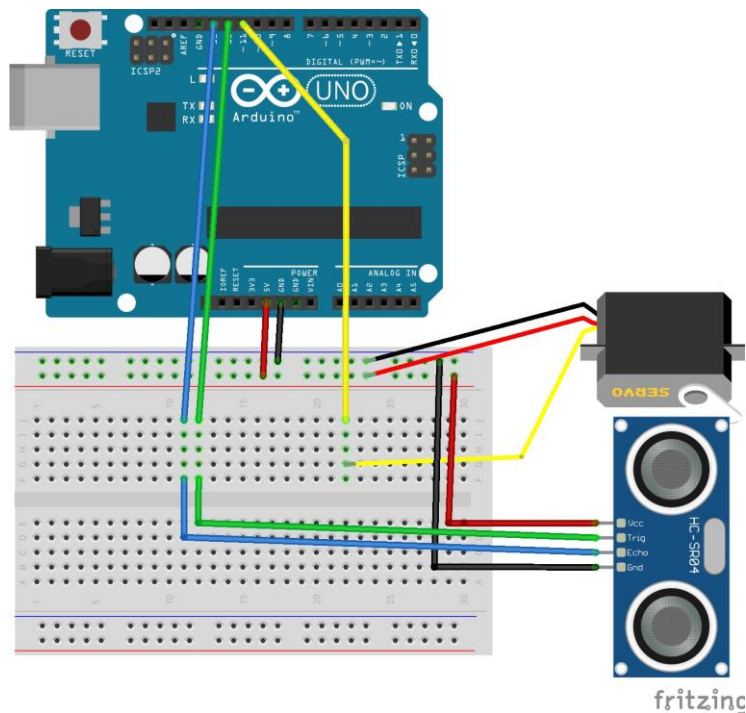
void loop() {
    // 초음파 발생
    digitalWrite(trig, HIGH);
    delayMicroseconds(10);
    digitalWrite(trig, LOW);

    // echo에서 HIGH를 유지하며 초음파 수신.
    unsigned long duration = pulseIn(echo, HIGH);
    digitalWrite(echo, LOW);
    // HIGH 동안의 시간(초음파가 되돌아온 시간)을 이용하여 거리를 계산한다.
    float distance = (float) 340 * duration / 20000;

    Serial.print(distance);
    Serial.println("cm");
    delay(20);
}
```

1  
2  
3  
4

## Practice 6 – Ultrasonic Sensor & Servo motor



### 6\_Ultrasonic\_Servo

```
#include <Servo.h>
#define servopin 11
#define trig 12
#define echo 13

float distance = 0;
Servo servomotor;

void setup() {
  Serial.begin(9600);
  pinMode(trig, OUTPUT);
  pinMode(echo, INPUT);

  digitalWrite(trig, LOW);
  digitalWrite(echo, LOW);

  servomotor.attach(servopin);
  Serial.println("Start");
}

void loop() {
  for(int i=0;i<=180;i++){
    servomotor.write(i);
    distance = ultrasonic();
    Serial.print(i);
    Serial.print("\t");
    Serial.println(distance);
    delay(10);
  }
  for(int i=180;i>=0;i--){
    servomotor.write(i);
    distance = ultrasonic();
    Serial.print(i);
    Serial.print("\t");
    Serial.println(distance);
    delay(10);
  }
}
```

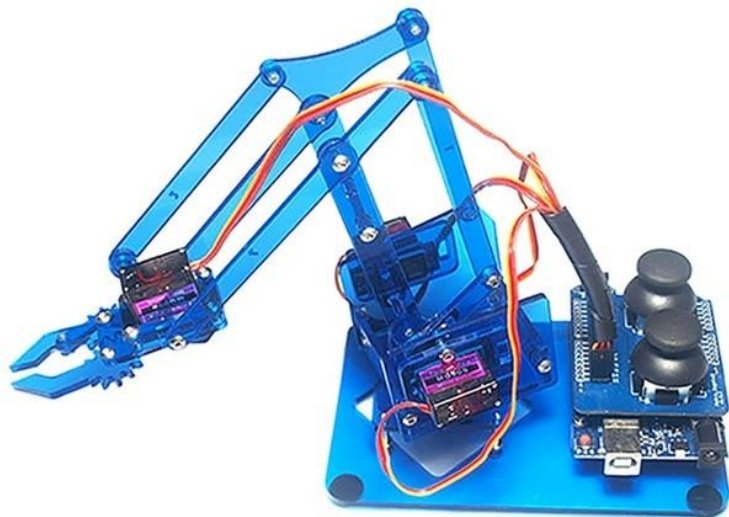
loop문 아래에  
ultrasonic 함수 작성

```
float ultrasonic(){
  // 초음파 발생
  digitalWrite(trig, HIGH);
  delayMicroseconds(10);
  digitalWrite(trig, LOW);

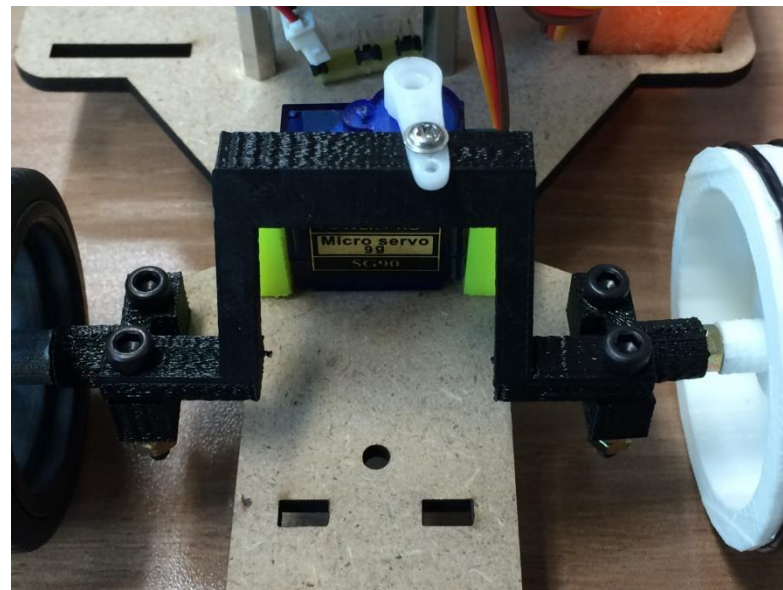
  // echoPin 이 HIGH를 유지하며 초음파를 수신한다.
  unsigned long duration = pulseIn(echo, HIGH);
  digitalWrite(echo, LOW);
  // HIGH 동안의 시간(초음파가 되돌아온 시간)을 이용하여
  float distance = (float) 340 * duration / 20000;

  delay(20);
  return distance;
}
```

# 서보모터로 할 수 있는 것



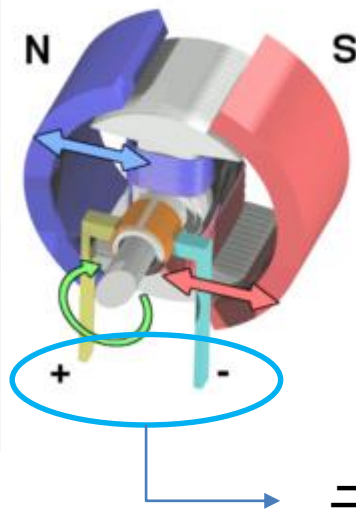
<https://ko.aliexpress.com/item/Best-Deal-Mearm-DIY-4DOF-For-Arduino-Robot-Arm-4-Axis-Rotating-Kit-With-Joystick-Button/32825659214.html>



<http://makezone.co.kr/blog/2016/09/05/%EC%A7%81%EC%A0%91-%EB%A7%8C%EB%93%A4%EC%96%B4%EB%B3%B4%EB%8A%94-rc-%EC%9E%90%EB%8F%99%EC%B0%A8/>

1  
2  
3  
4

## DC모터 사용하기



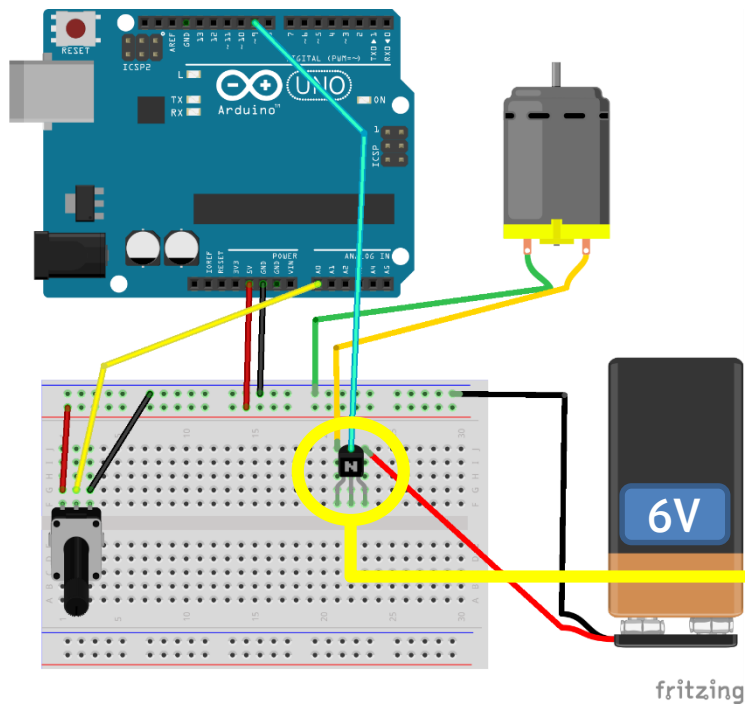
### DC모터(직류모터)

회전자에 흐르는 전류 방향을 전환함으로써  
자석과 유도자기장의 상호작용을 통해  
회전력을 얻는 모터

서보모터 또한 DC모터에 기어박스 및 컨트롤러를  
부착해서 만든 모터다.

극을 반대로 연결하면 반대 방향으로 회전

## DC모터 사용하기



## 사용할 모터의 경우

## Arduino 내부 전력으로는 구동 불가능

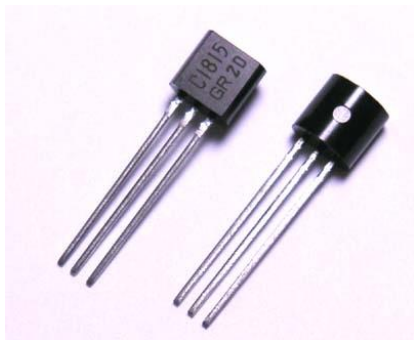
외부전원이 필요하여 6V 배터리를 사용한다.

**전력제어를 위해 트랜지스터를 사용해야 함.**

## NPN 트랜지스터



## DC모터 사용하기



### 접합형 트랜지스터(Bipolar Junction Transistor)

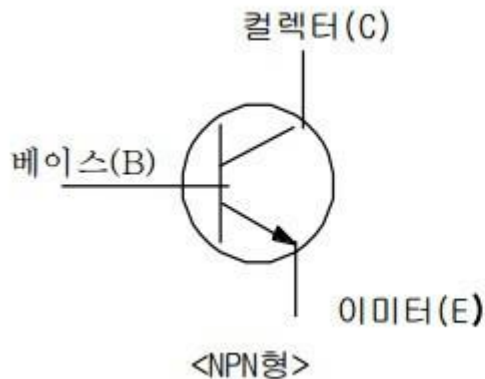
NPN, PNP 2종류가 있으며,  
전류 증폭 용도로 사용한다.

Base 입력을 Collector / Emitter의 전류를  
활용하여 증폭해주는 역할.

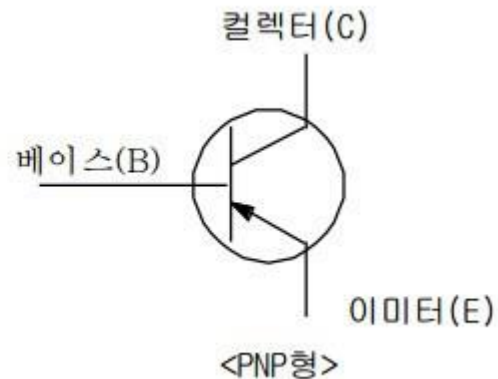
NPN은 Base 입력과 동일한 출력,  
PNP는 Base 입력에 반대되는 출력을 제공함.

1  
2  
3  
4

## DC모터 사용하기



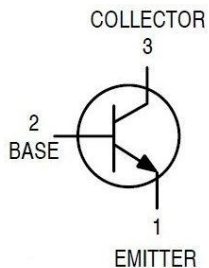
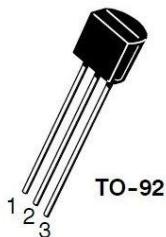
C – 외부전원 연결  
B – PWM신호 입력  
E – 사용할 모듈 연결



C – 사용할 모듈 연결  
B – PWM신호 입력  
E – 외부전원 연결

## DC모터 사용하기

### 2N2222

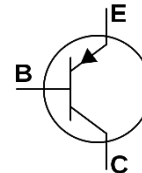


<https://www.onsemi.cn/PowerSolutions/document/P2N2222A-D.PDF>



### KTC9015 pinout

1. Emitter
2. Base
3. Collector



<http://www.buchangelec.com/data/Ss-tr/9015.pdf>

## Practice 7 – DC Motor

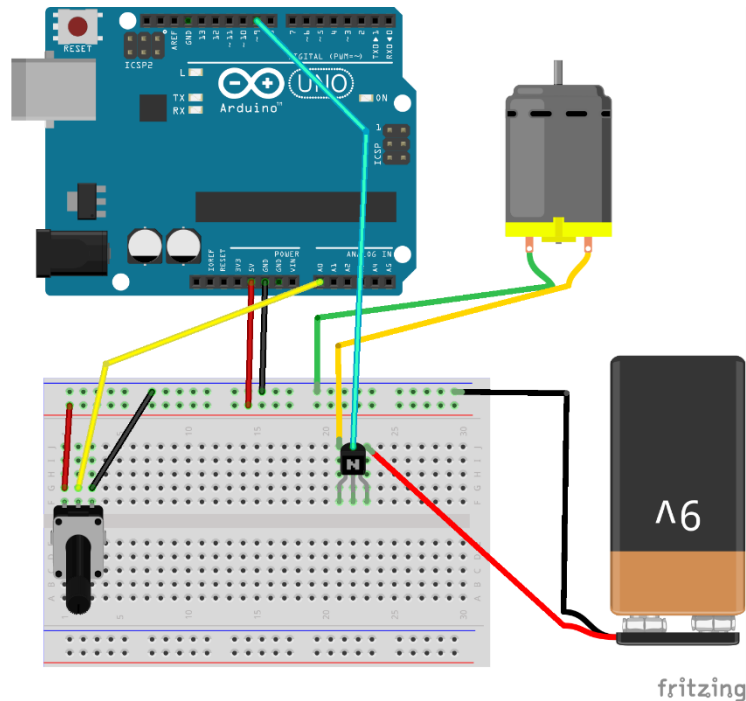
```
#define dc 9

void setup()
{
    pinMode(dc,OUTPUT);
    analogWrite(dc,0);

    Serial.begin(9600);
    Serial.println("Start");
}

int dc_speed = 0;

void loop()
{
    if(Serial.available())
    {
        dc_speed = Serial.parseInt();
        Serial.println(dc_speed);
    }
    analogWrite(dc,dc_speed);
}
```



1  
2  
3  
4

## Practice 8 – DC Motor & Potentiometer

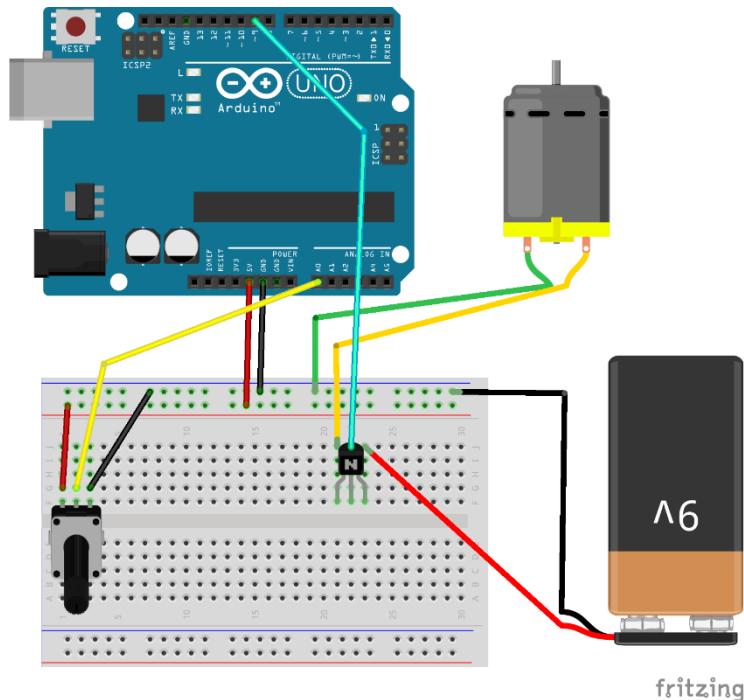
```
#define dc 9
#define potentiometer A0

void setup()
{
    pinMode(dc, OUTPUT);
    pinMode(potentiometer, INPUT);
    digitalWrite(dc, 0);

    Serial.begin(9600);
    Serial.println("Start");
}

int dc_speed = 0;

void loop()
{
    dc_speed = map(analogRead(potentiometer), 0, 1023, 0, 255);
    Serial.print("Potentiometer : \t");
    Serial.print(analogRead(potentiometer));
    Serial.print("\tSpeed : \t");
    Serial.println(dc_speed);
    digitalWrite(dc, dc_speed);
}
```





1  
2  
3  
4

## DC모터 사용하기

만약 반대방향으로 회전시키고 싶다면?

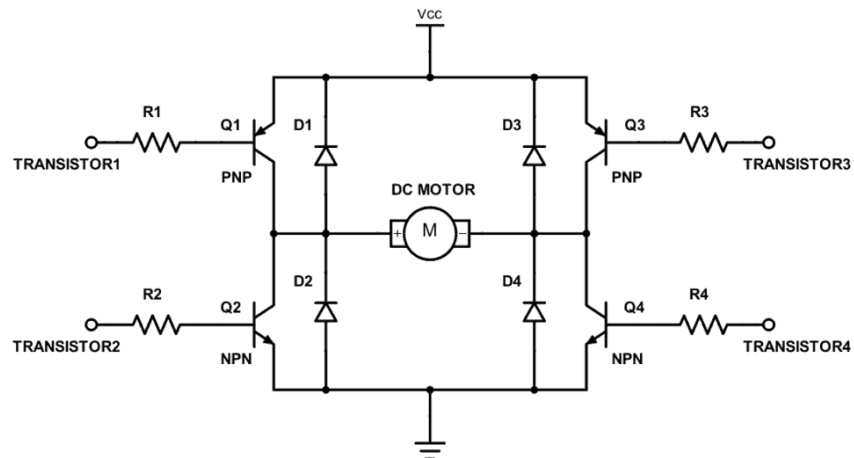
H-Bridge 형태의 회로를 사용하여  
반대방향으로 전류를 흘려줘야 함



1  
2  
3  
4

## DC모터 사용하기

만약 반대방향으로 회전시키고 싶다면?





1  
2  
3  
4

## DC모터 사용하기

만약 반대방향으로 회전시키고 싶다면?

이러한 회로가 모두 내장된  
모터 드라이버를 사용할 것!

회로 고민할 필요도 없고, 사용도 간단하다.





Q & A

kynk94@naver.com

들어주셔서 감사합니다.