임베디드컴퓨팅
Embedded Computing
(0009488)
Servo Motor

2022년 2학기

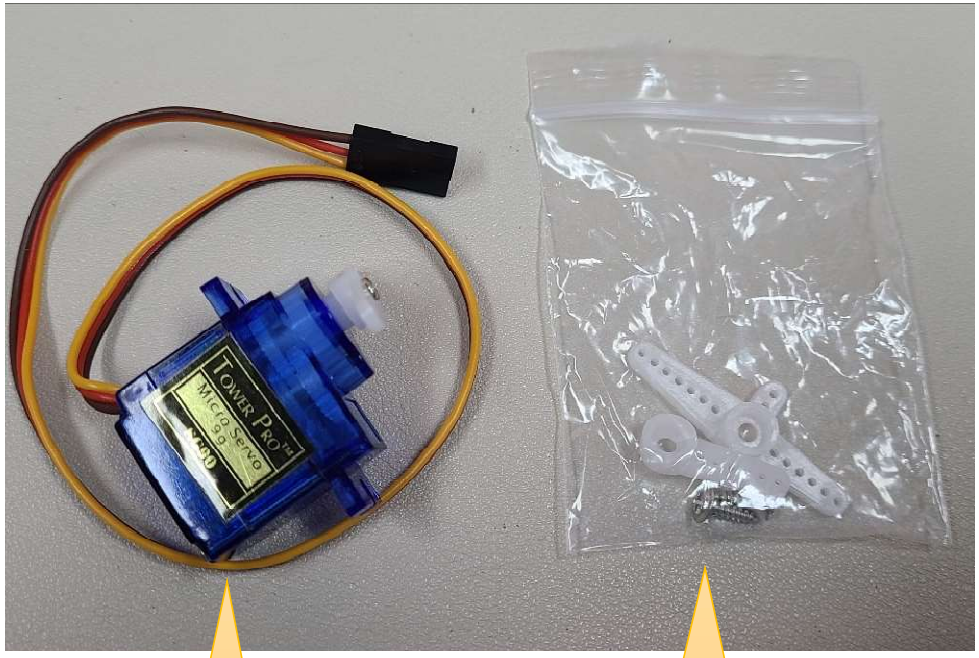
정보기술대학 정보통신공학과
김 영 필
ypkim@inu.ac.kr

INU 인천대학교
INCHEON NATIONAL UNIVERSITY

Servo motor

- The DC motor
  - Rotates in one direction when power is applied
- The servo motor
  - Rotates within a certain angle range and can be controlled to a desired angle.
  - SG90 servo motor rotates within the range of 0 to
    - Other type of servo motors can rotate 360˚

(SG90)



Servo motor

Fans and bolts

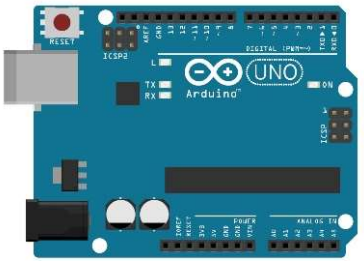
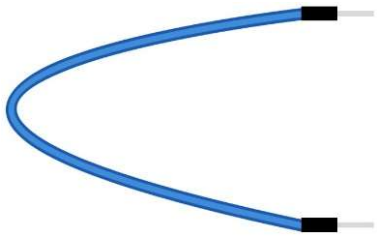
Three cables of servo motor

- Orange
  - A pin that controls the PWM
  - Connects to the [redacted] pin of the Arduino.
- Red
  - A pin that applies + power,
  - Connects to the 5V pin of the Arduino.
- Brown
  - A grounding pin
  - Connected to the GND pin of the Arduino.

Lab: Operating a servo motor

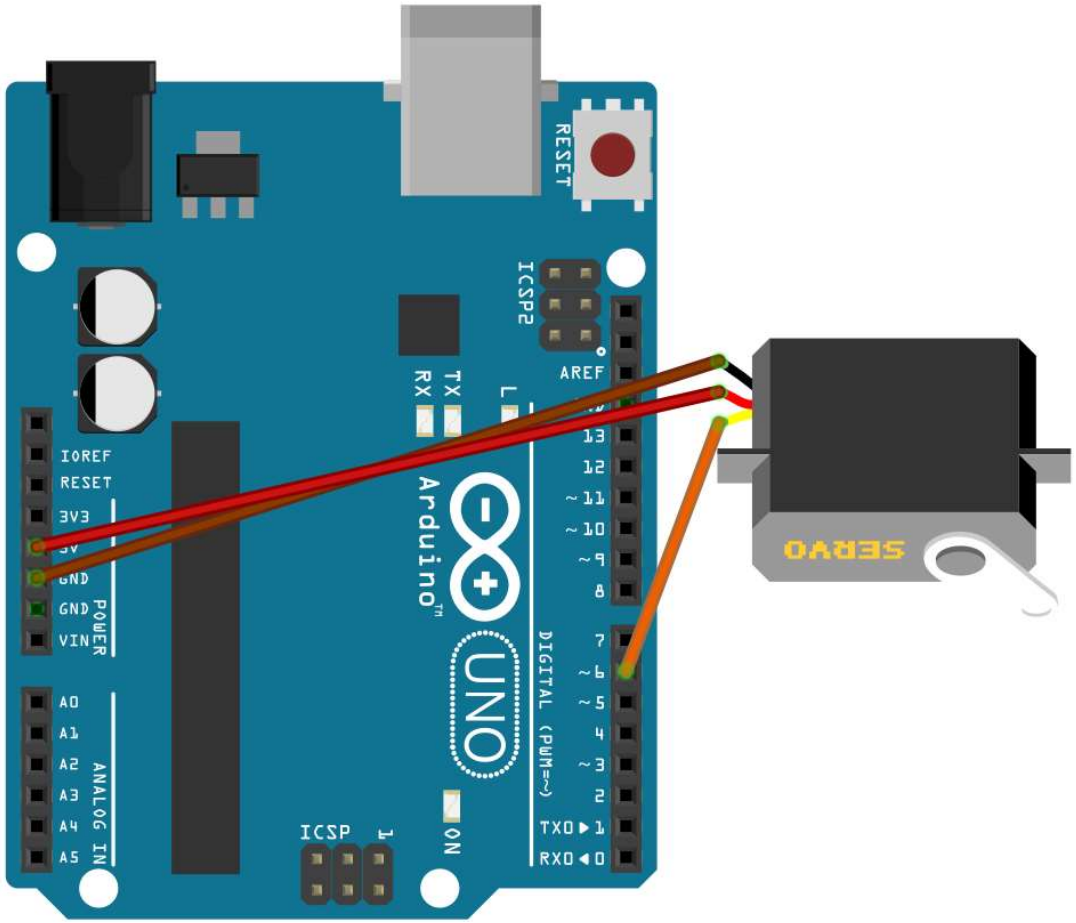
- Let's write a sketch program that the servo motor rotates within the range of 0 ~ 120°.

- Required H/W components:

Arduino board (Uno)	Servo motor (SG-90)	Jumper cables Male-Male
		
x 1	x 1	x 3

Circuit wiring setup

Servo motor cables	Arduino board
Organge	PWM 6
Red	5V
Brown	GND



Basic setup for a servo motor

```
#include <Servo.h>
#define SERVO_PIN 6

Servo myservo;
int pos = 0;
int ang = 60;

void setup() {
  
}

void loop() {
  for (pos = 0; pos <= ang; pos += 1) {
    
    delay(15);
  }
  for (pos = ang; pos >= 0; pos -= 1) {
    
    delay(15);
  }
}
```

include a library for servo

Declare Servo object, two variables for a position and an angle

attach to a PWN pin

rotate a motor by writing a position via servo motor library

wait for a motor movement

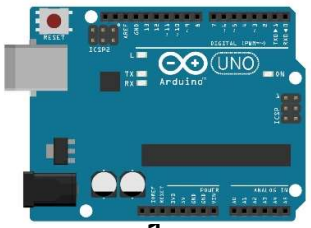
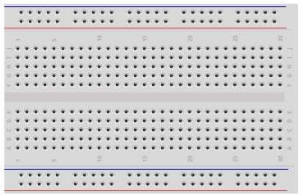
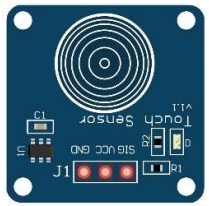
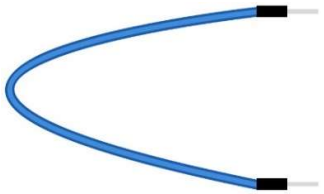
reverse rotation of the motor

Check results

- What can we observe?
- If the motor seems to stuck in, push the fan slightly and gently.
  - Not strongly push

Lab 2: A servo motor with a touch sensor

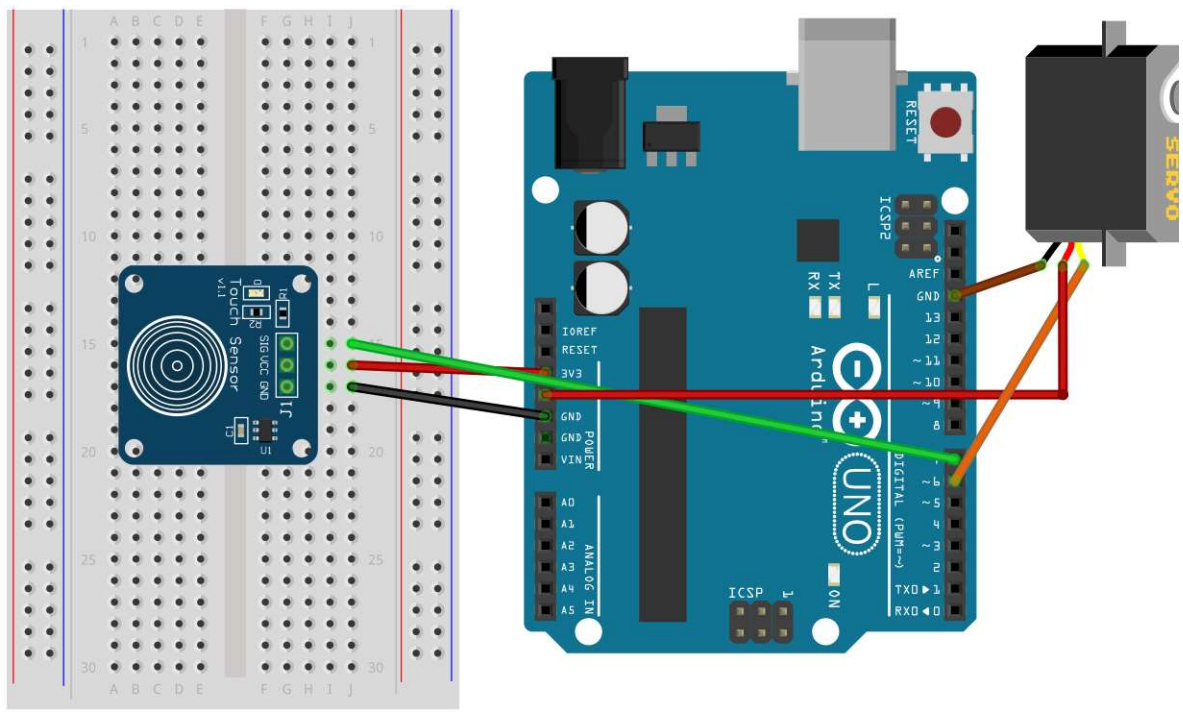
- Let's write a sketch program that the motor rotates when
  - the touch count becomes an odd number, and the count increases only when the touch sensor is pressed for 1 to 3 seconds
- Required H/W components:

Arduino board (Uno)	Breadboard (400 pins)	Servo motor (SG-90)	Touch sensor (TTP223B)
 x1	 x1	 x1	 x1
Jumper cables Male-Male			
 x6			

Circuit wiring setup

Servo motor cables	Arduino board
Organge	PWM 6
Red	5V
Brown	GND

Touch sensor	Arduino board
VCC	3.3V
GND	GND
SIG	digital 7



Basic setup for Lab 2

```
#include <Servo.h>
#define TOUCH 7
Servo myservo;
int touchCount = 0;
int pos = 0;
unsigned long startTime;
unsigned long touchTime;

void setup() {
  
  pinMode(TOUCH, INPUT);
  Serial.begin(9600);
}
```

include a library for servo

Declare Servo object, two variables for a position and a rotateState

Declate variables for time for start and touch

Attach to PWD pin 6

Setup a touch sensor and serial communication

Loop for Lab 2

```
void loop() {
  int touchValue = digitalRead(TOUCH);
  if(touchValue == HIGH) {
    startTime = millis();
    while(digitalRead(TOUCH)  HIGH);
    touchTime = millis() - startTime;
    if(touchTime  1000 &&
     touchTime  3000) {
        touchCount++;
    }
    Serial.print(“touchCount : ");
    Serial.print(touchCount);
    Serial.print("\tcontinuous time :
");
    Serial.println(touchTime);
```

Read touch state

Measure elapsed time for pressing a touch sensor in milliseconds

Wait until releasing a touch sensor

Count the number of valid touches

Loop for Lab 2

```
 if(                              ) {
    for (pos = 0; pos <= 120; pos
+= 1) {
      myservo.write(pos);
      delay(15);
    }
    for (pos = 120; pos >= 0; pos
-= 1) {
      myservo.write(pos);
      delay(15);
    }
  }}
}
```

touchCount is odd number?

rotate a motor by writing a position via servo motor library

wait for a motor movement

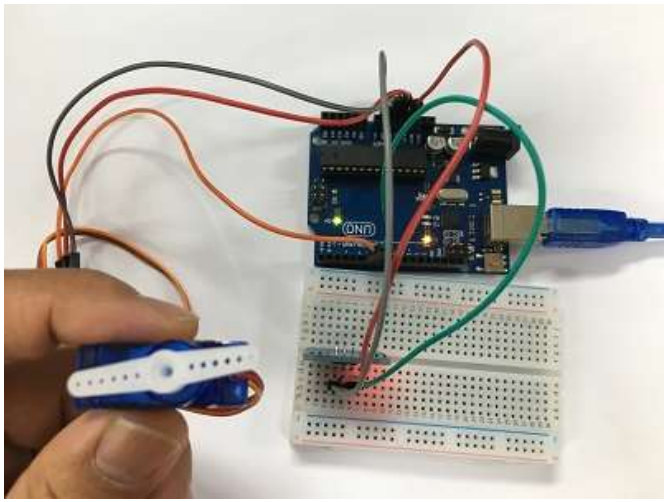
reverse rotation of the motor

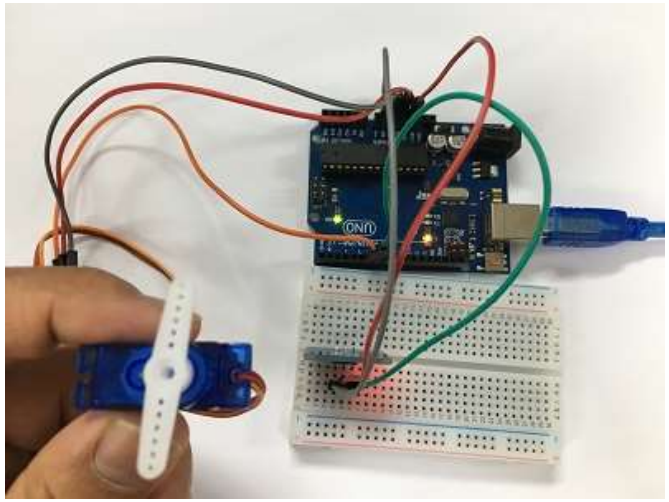
Check results

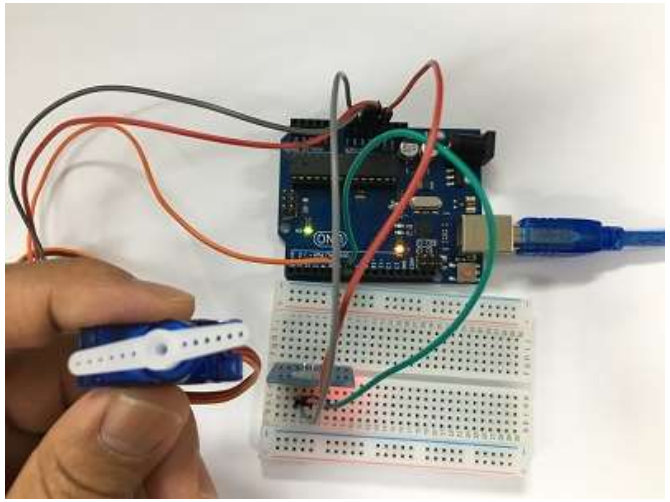
- What can we observe?

COM13

```
touchCount  : 0 continuous time : 113
touchCount  : 1 continuous time : 1745
touchCount  : 2 continuous time : 1322
```







Random number generation: random()

- The random function generates pseudo-random numbers.
- **Syntax**
  - random(max)
  - random(min, max)
- **Parameters**
  - min: lower bound of the random value, inclusive (optional).
  - max: upper bound of the random value, exclusive.
- **Returns**
  - A random number **between min and max-1**. Data type: long.

```
long randNumber;

void setup() {
  Serial.begin(9600);
  randomSeed(analogRead(0));
}

void loop() {
  randNumber = random(300);
  Serial.println(randNumber);

  randNumber = random(10, 20);
  Serial.println(randNumber);

  delay(50);
}
```

Ref - https://www.arduino.cc/reference/en/language/functions/random-numbers/random/

Assignment: AI-fortune teller

- Let's implement an AI-fortune teller based on Lab 2 (w/ touch sensor).
- Ask the teller for the yes/no question, and touch
  - e.g. Can I be famous in five years?
- Requirements
  - Valid touch time is same as Lab 2 (1-3 sec)
  - Initially, set the motor position to 0.
  - Wait 5 seconds, and then decide its answer randomly
  - If yes, rotate the motor (0 to 120), and reverse it (120 to 0).
  - If no, rotate the motor (0 to 120), and stop it.
  - Show the result via serial communication
  - A block-type comments in the top of source code w/ "your student no., your name, writing date, etc."
- Results
  - (a source code file) sketch source code (***"sketchfilename.ino"***)
  - (a Arduino board capture file) a photo capture showing how you setup your circuit (max. 1GB file).