Engineering & Robotics

Lesson 9&10:

Project Assignment

Our Educators,
Amir Ashik
Kartini Subramaniam

STEM inc
Brought To You By
Science Centre Singapore

Announcement!

- Servomotors MUST be handled with care. Do not turn it at will.
- Connect the POLARITY of Capacitors correctly.
- TWO WEEKS class

Design Build Code Troubleshoot

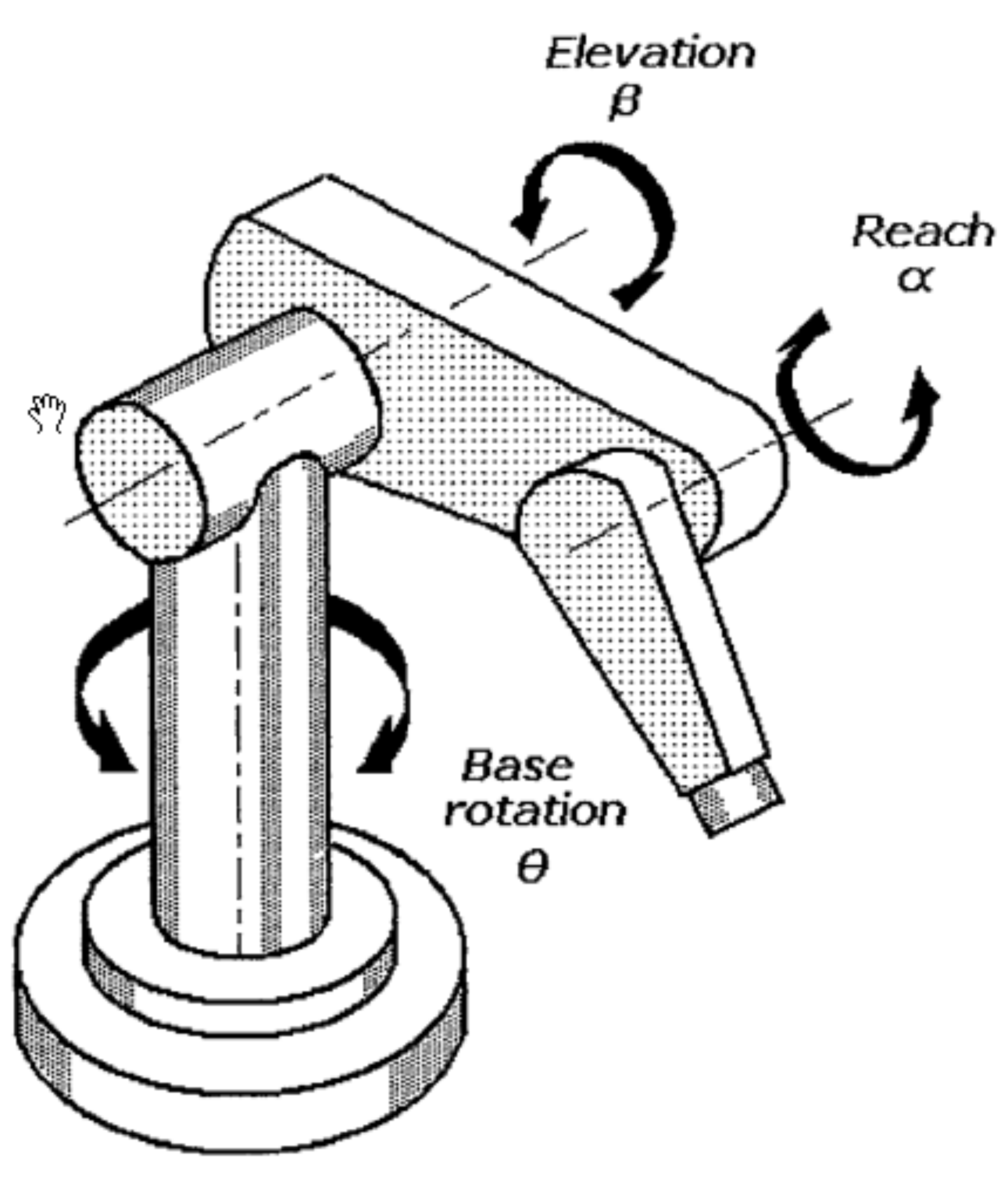
- **Design:** Idealisation
- **Build:** Assemble
- **Code:** Sketch
- **Troubleshoot:** Make it work!

Video 1

Video 2

Lesson	Component	Command	Remark
1	Light Emitting Diode (LED)	digitalWrite (pin, value)	output
2	Resistor (R)	int, const int	variable
3	Push button (PB)	pinMode (pin, mode) digitalRead (pin)	- input
4	Light Dependent Resistor (LDR)	analogRead (pin)	input
5	Servomotor (servo)	for (i=0; i<x; i++)	loop
6	DC Motor	analogWrite (pin, value)	output
7	Display Panel	Library <library.h>	

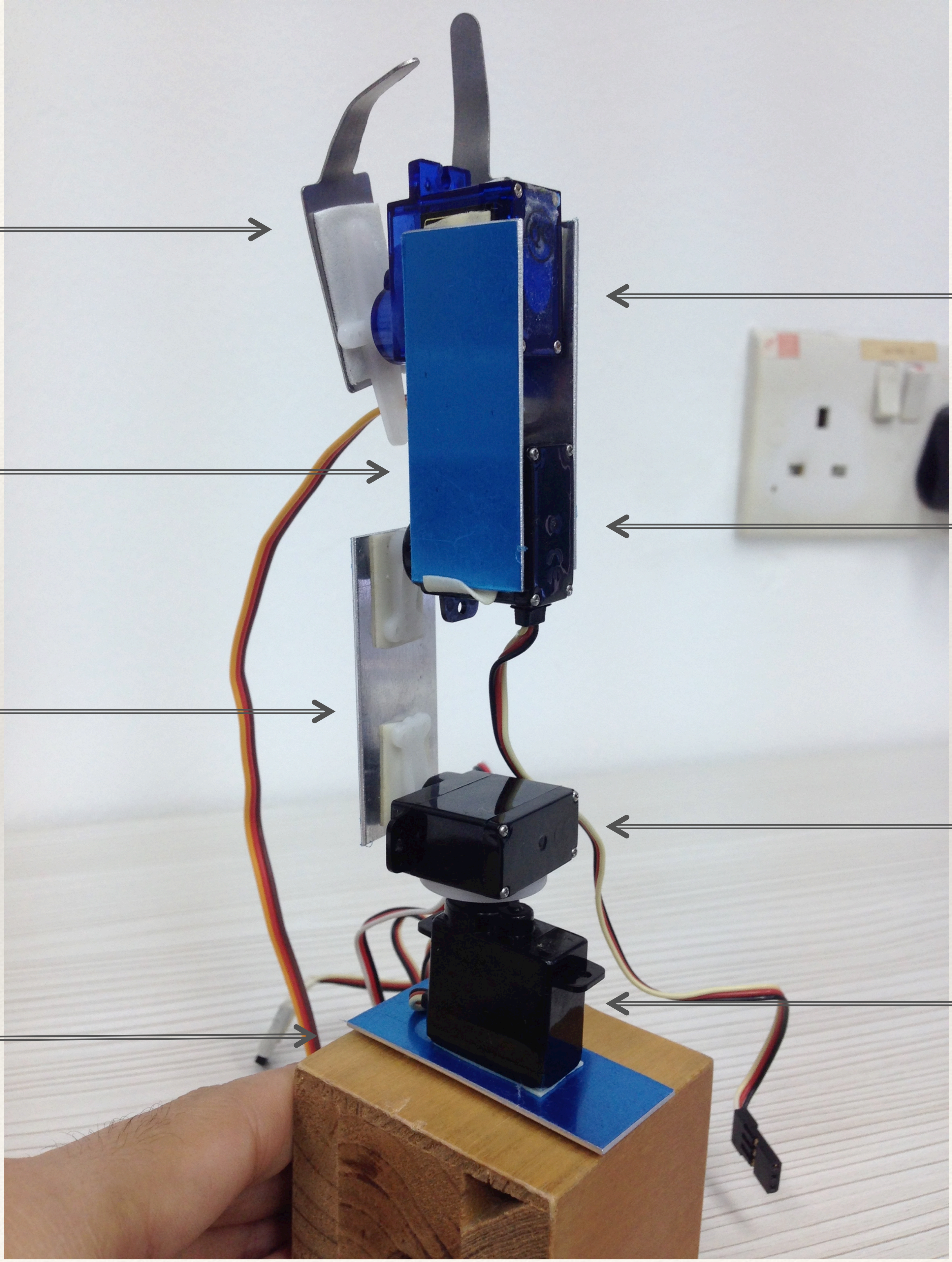
At a glance



Design (Idea)

Design (Materials)

Part	Material	Quantity
Gripper	Cardboard	1 pc
Reach	PVC	2 pcs
Elevation	PVC	1 pc
Base	Acrylic	1 pc



Gripper

Reach

Elevation

Base

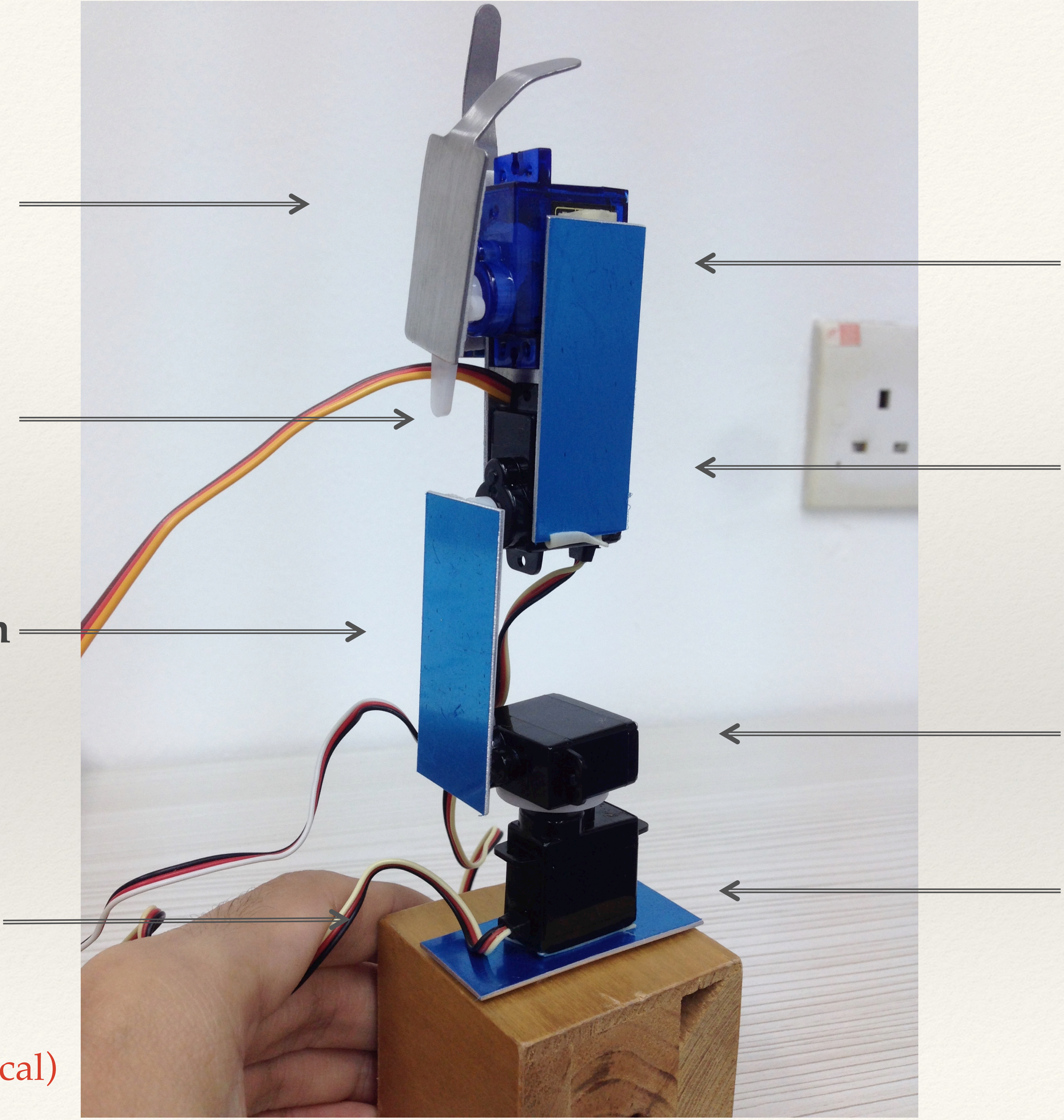
S1

S2

S3

S4

Build (Mechanical)



Gripper

Reach

Elevation

Base

S1

S2

S3

S4

Build (Mechanical)

Build (Electronic)

Breadboard

Servomotor

Push Button

Capacitor

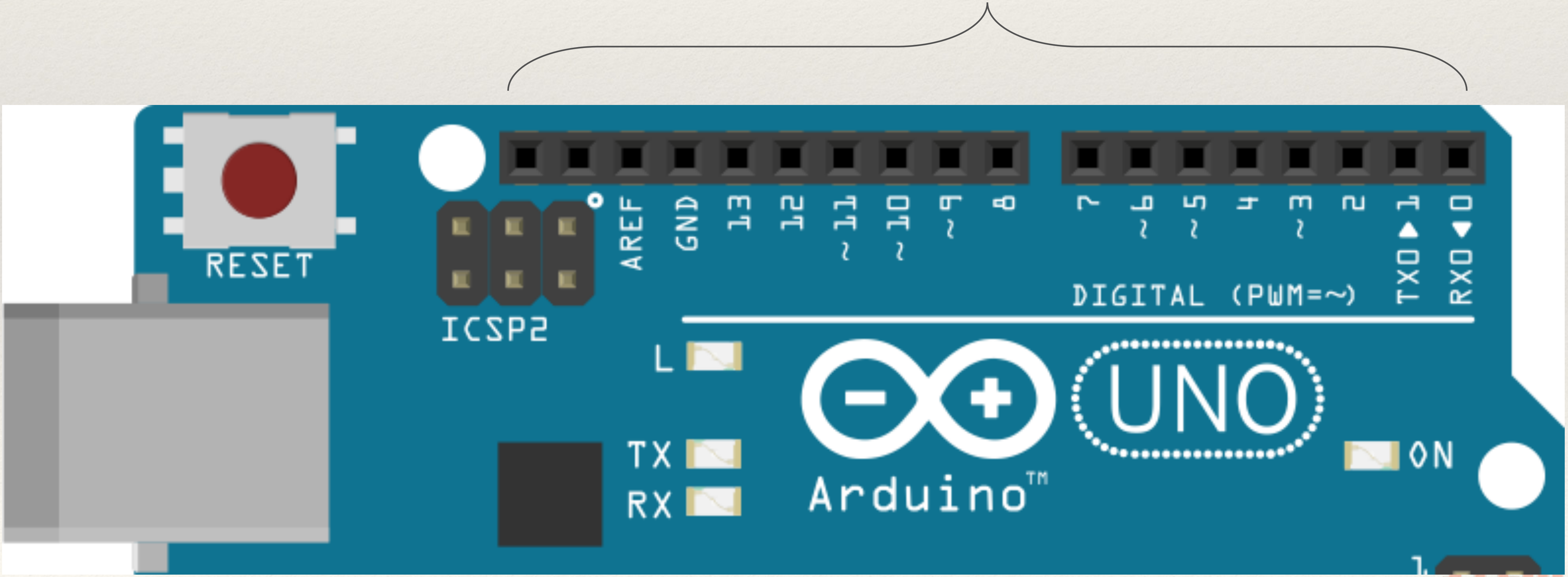
Jumper wires

Resistor (10kohms)

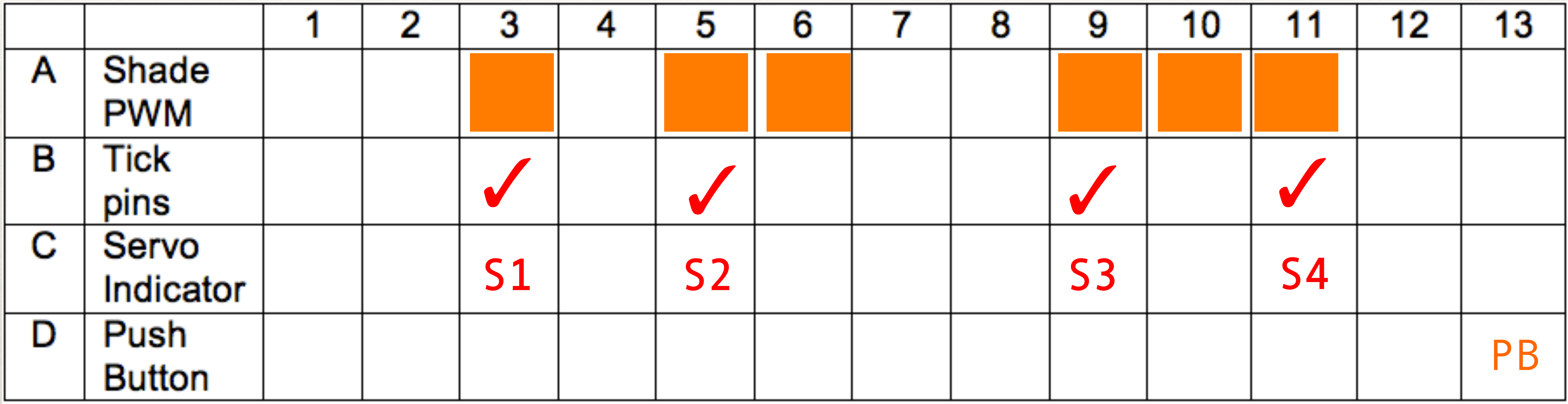
PWM Pins

Hint: There are 6!

Digital Input & Output
Pin #1 - 13



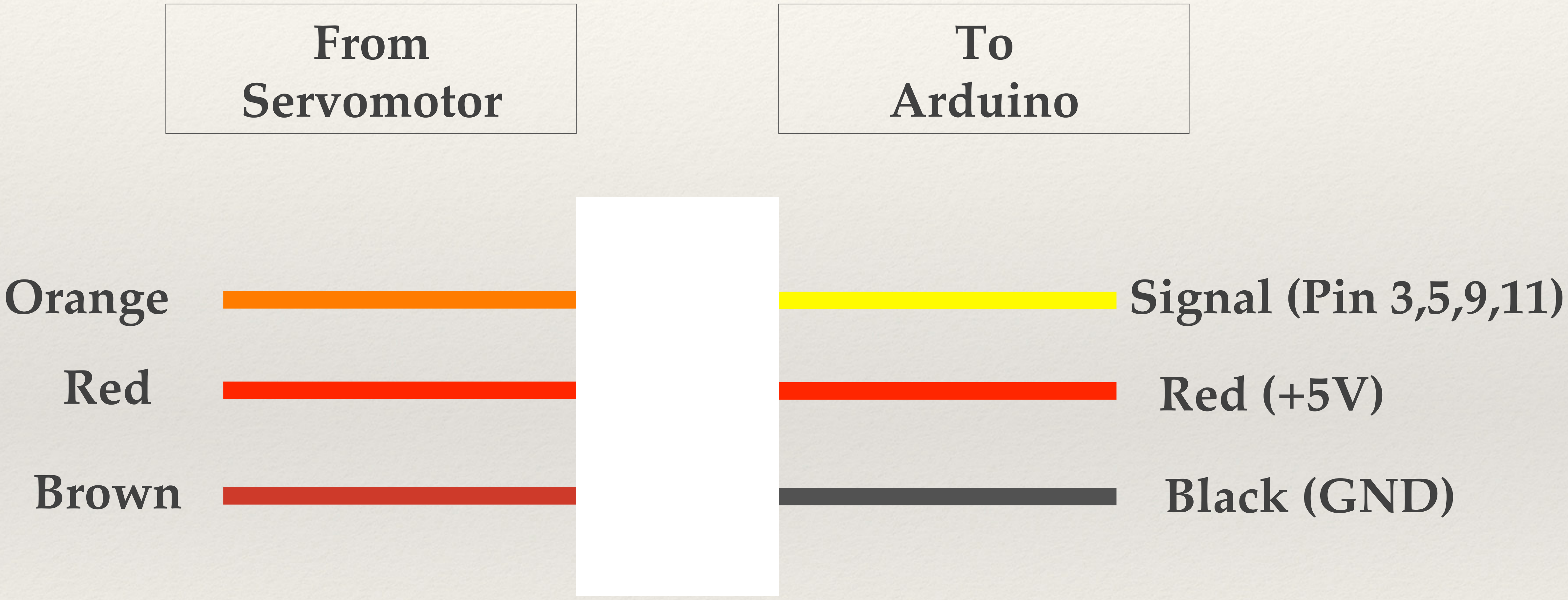
Build (Electronic)



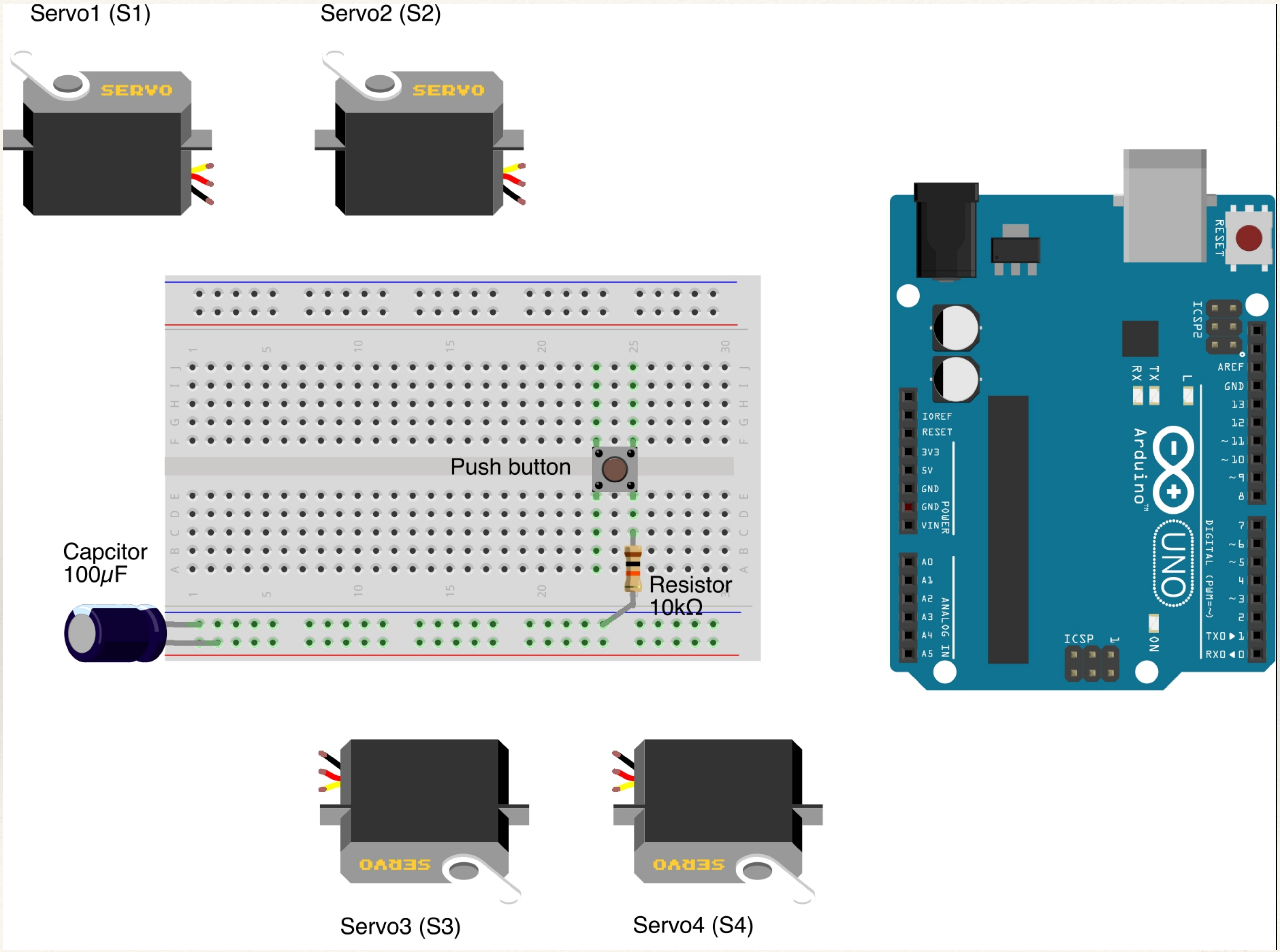
		1	2	3	4	5	6	7	8	9	10	11	12	13
A	Shade PWM			🟧		🟧	🟧			🟧	🟧	🟧		
B	Tick pins			✓		✓				✓		✓		
C	Servo Indicator			S1		S2				S3		S4		
D	Push Button													PB

- A : Identify PWM pins
- B : Select 4 PWM pins
- C : Label servomotors to pins
- D: Pin for push button

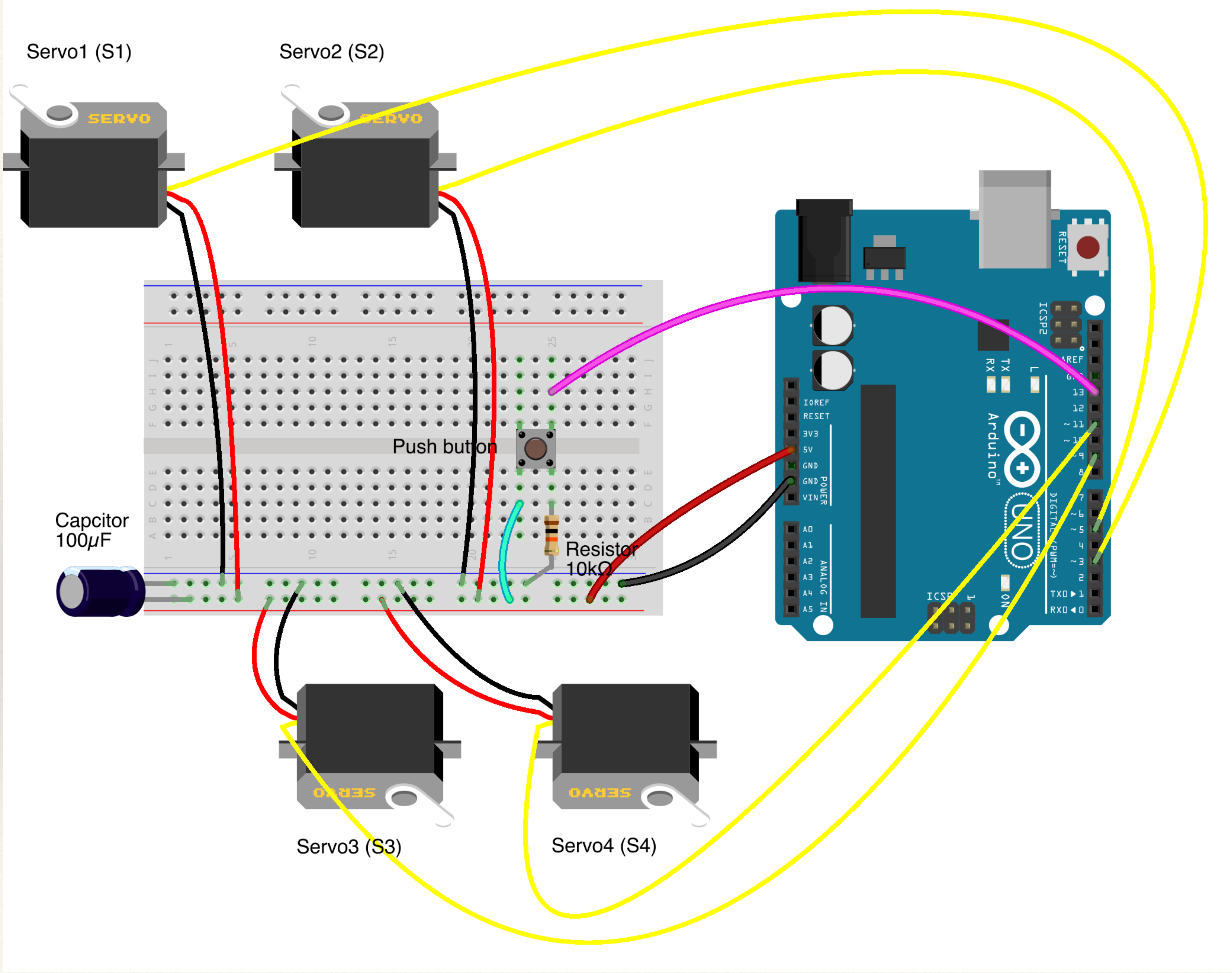
Servomotor wiring

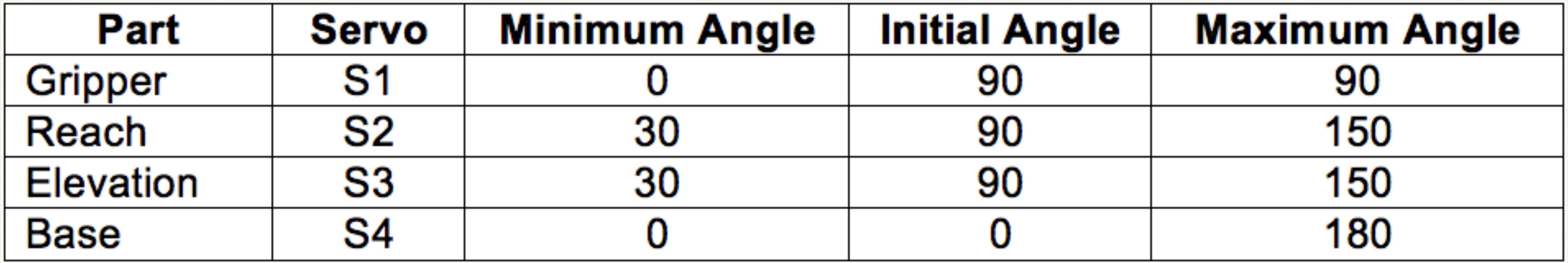


From Servomotor	To Arduino
Orange	Signal (Pin 3,5,9,11)
Red	Red (+5V)
Brown	Black (GND)

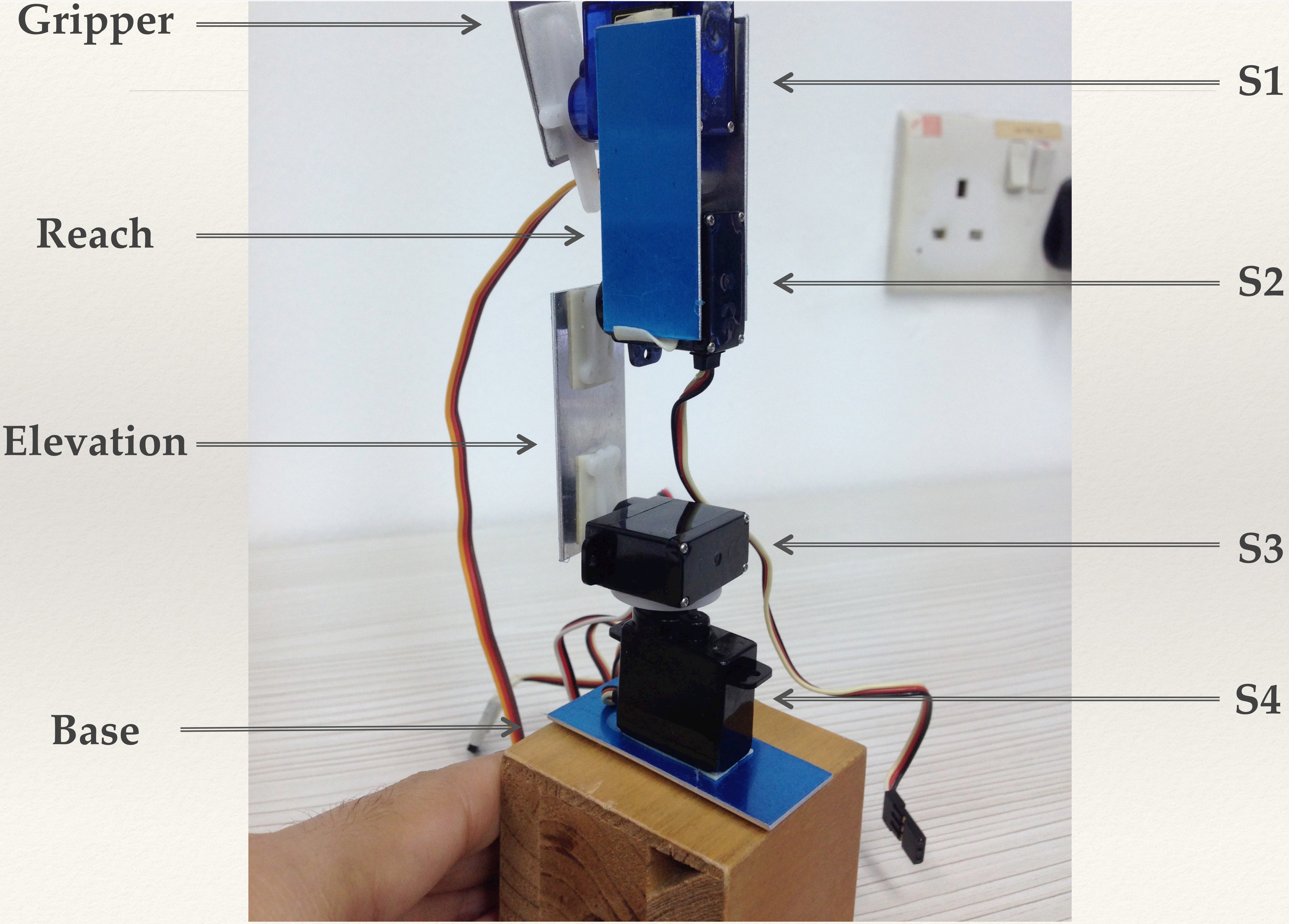


Servo1 (S1)
Servo2 (S2)
Push button
Capcitor
100μF
Resistor
10kΩ
Servo3 (S3)
Servo4 (S4)
SERVO
RESET
ICSP2
AREF
GND
DIGITAL (PWM=~)
Arduino
UNO
IOREF
RESET
3V3
5V
GND
GND
VIN
POWER
A0
A1
A2
A3
A4
A5
ANALOG IN
ICSP
ON
TX
RX
L


Servo1 (S1)
Servo2 (S2)
SERVO
Capcitor
100μF
Push button
Resistor
10kΩ
Arduino
UNO
Servo3 (S3)
Servo4 (S4)



Part	Servo	Minimum Angle	Initial Angle	Maximum Angle
Gripper	S1	0	90	90
Reach	S2	30	90	150
Elevation	S3	30	90	150
Base	S4	0	0	180



Lets get Coding! (Declare)

```
#include <Servo.h>    //using servo library

Servo S1;         // create servo object to control a servo
Servo S2;
Servo S3;
Servo S4;

const int button = 13;     // button to pin 13
int currentState = LOW;    // button is depressed
```

Lets get Coding! (Prepare)

```
void setup()
{
  S1.attach(3);        // assign S1 to Pin 3
  S2.attach(5);        // assign S2 to Pin 5
  S3.attach(9);        // assign S3 to Pin 9
  S4.attach(11);       // assign S4 to Pin 11

  S1.write(90);          // set servo at 90 deg position
  S2.write(90);          // set servo at 90 deg position
  S3.write(90);          // set servo at 90 deg position
  S4.write(0);           // set servo at 00 deg position

  pinMode(button, INPUT);  // set the push button as INPUT
  delay(3000);             // pause for 3 secs
}
```

Lets get Coding! (Execute)

```
void loop()
{
  currentState = digitalRead(button);     // read the state of button
  if (currentState == HIGH)               // if the button is pressed
  {
                              // 30 < angle < 150
                              // pause for 1 sec
                              // 30 < angle < 150
                              // pause for 1 sec
                              // 0 < angle < 90
                              // pause for 1 sec
                              // 0 < angle < 180
                              // pause for 1 sec
                              // 0 < angle < 90
                              // pause for 3 sec
  }
```

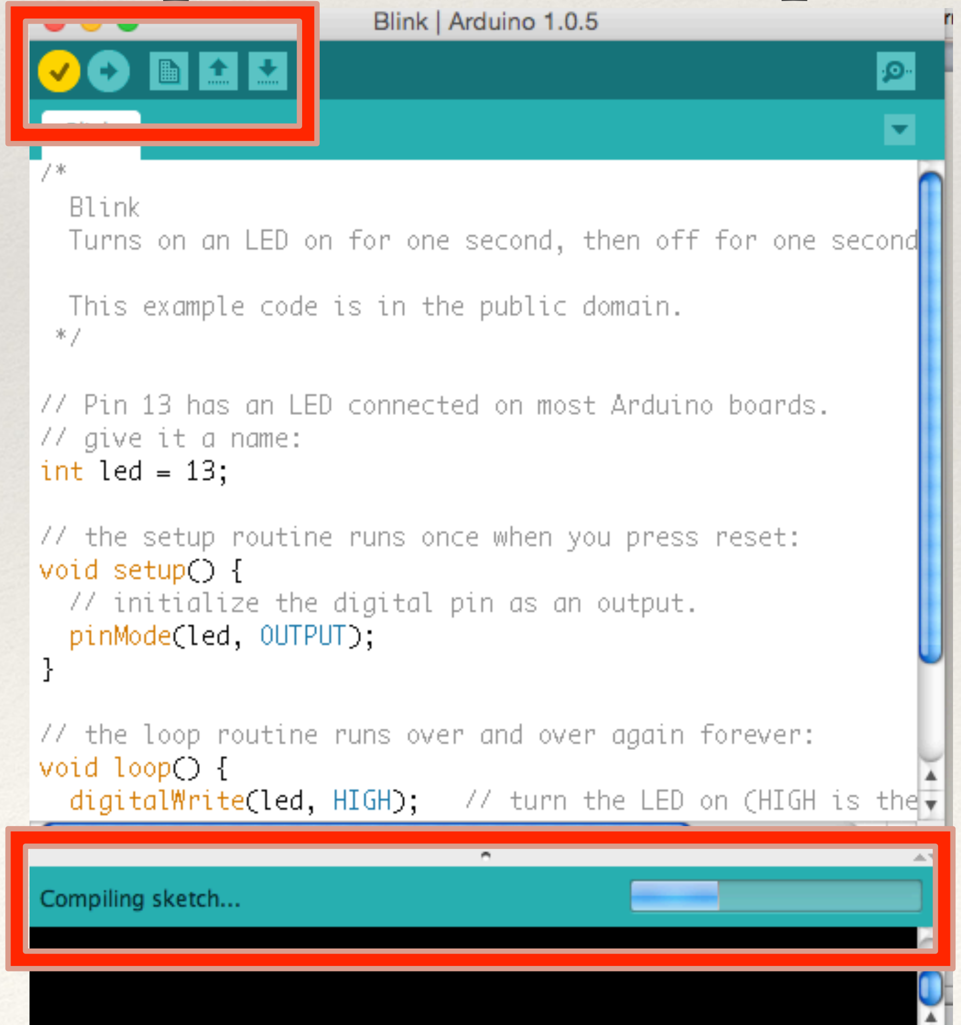
Lets get Coding! (Execute)

```
else        // if the button is depressed (reset to original position)
  {

                              // 0 < angle < 90
                              // pause for 0.5 sec
                              // 30 < angle < 150
                              // pause for 0.5 sec
                              // 30 < angle < 150
                              // pause for 0.5 sec
                              // 0 < angle < 180
                              // pause for 0.5 sec
    }
 }
```

Troubleshoot

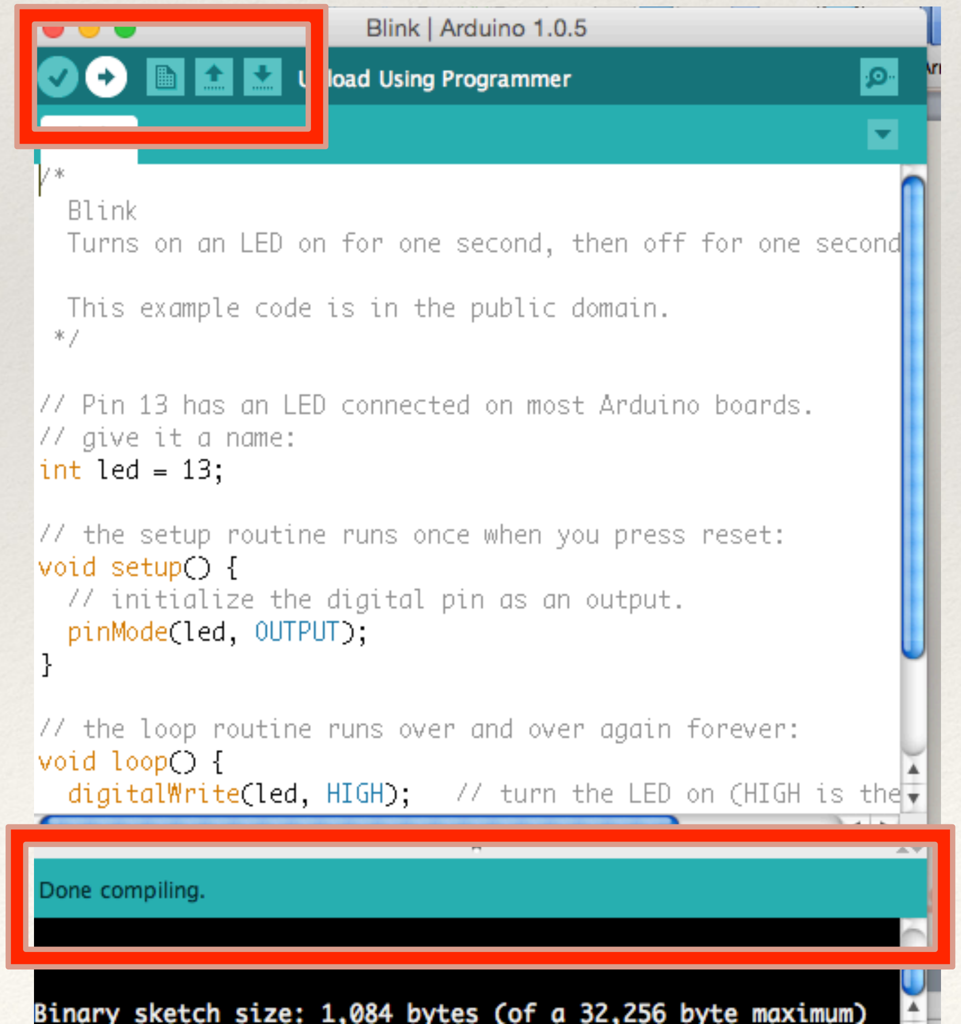
Step 1: Click the **Verify** button (to check for errors)

Step 2: Click the **Upload** button



```
/*
  Blink
  Turns on an LED on for one second, then off for one second

  This example code is in the public domain.
 */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);   // turn the LED on (HIGH is the
```

Compiling sketch...



```
/*
  Blink
  Turns on an LED on for one second, then off for one second

  This example code is in the public domain.
 */

// Pin 13 has an LED connected on most Arduino boards.
// give it a name:
int led = 13;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);   // turn the LED on (HIGH is the
```

Done compiling.

```
Binary sketch size: 1,084 bytes (of a 32,256 byte maximum)
```

Troubleshoot (Lets Think!)

Were there errors upon verifying your program?

How do you correct it?

Troubleshoot (Lets Think!)

Is your arm able to move upon uploading your sketch?

What was your greatest challenge?

Troubleshoot (Lets Think!)

Could you identify what part(s) went wrong?

Discuss with your Educator

Challenge Yourself!

Add another switch to your circuit.

Program anothter set of instructions to perform another arm movement as shown in Video 2

Use the sketch given in the Project Website.

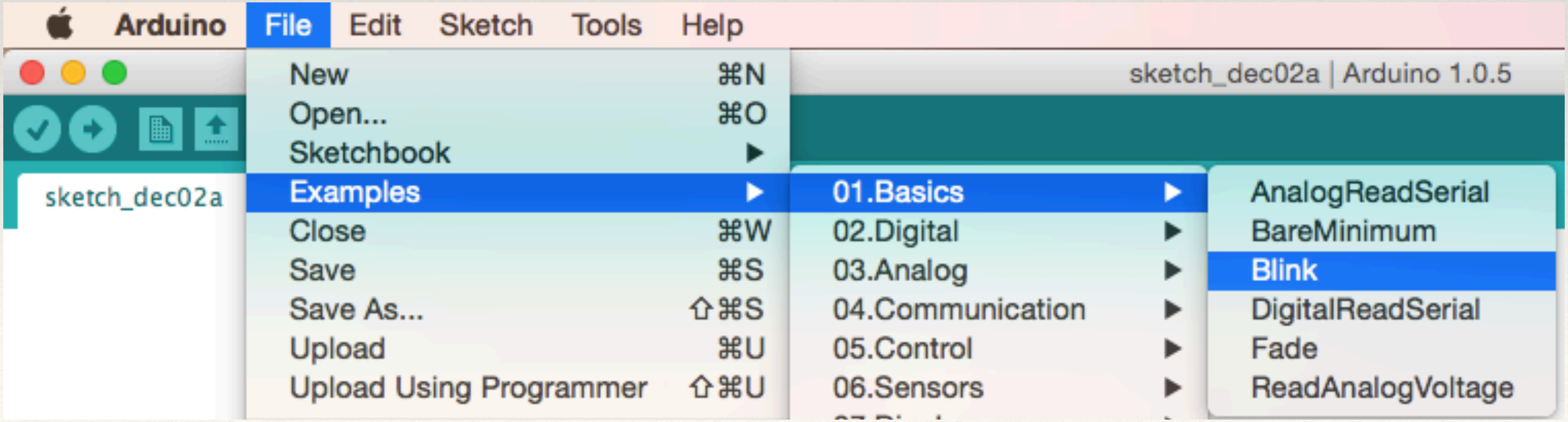
Show your educators how yours work!!

Before you go…

Step 1: Disconnect all your components

Step 2: Click: File > Examples > 01.Basics > Blink

Step 3: Click the **Upload** button



Brought to You By
Science Centre Singapore