



ARDUINO BASICS: SENSORS AND SERVOS



COLUMBIA UNIVERSITY SCIENCE & ENGINEERING LIBRARIES



HELLO!

JENNIFER BROWN

Emerging Technologies Coordinator

Science and Engineering Libraries

ref-sci@columbia.edu

[@CU_SEL](#) or [@jeninthelib](#)



What You'll Learn:

- ❑ **General information** about the Arduino hardware platform, relevant input devices, and the Arduino's programming language.
- ❑ Where to **get help**.
- ❑ **From one another!**



HARDWARE LIMITATIONS:

Two arduinos for the day

1

GETTING STARTED:

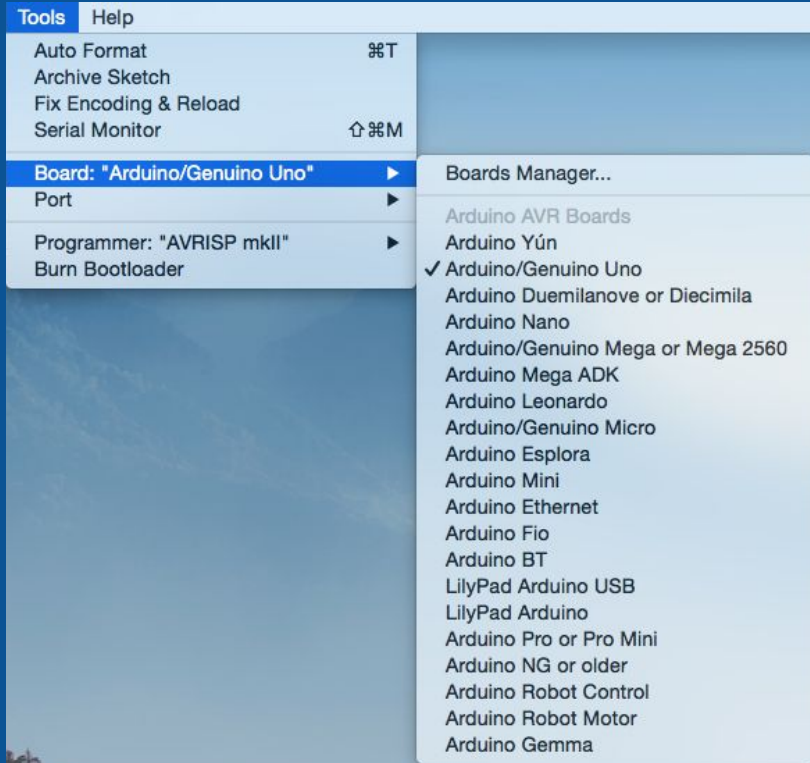
Downloading important software



Arduino IDE:

bit.ly/1R2xniM

Initial Arduino IDE Setup:



This selection sets the parameters used when compiling (CPU / baud rate) and the file / fuse settings used by the bootloader.



UNDERSTANDING THE BASICS:



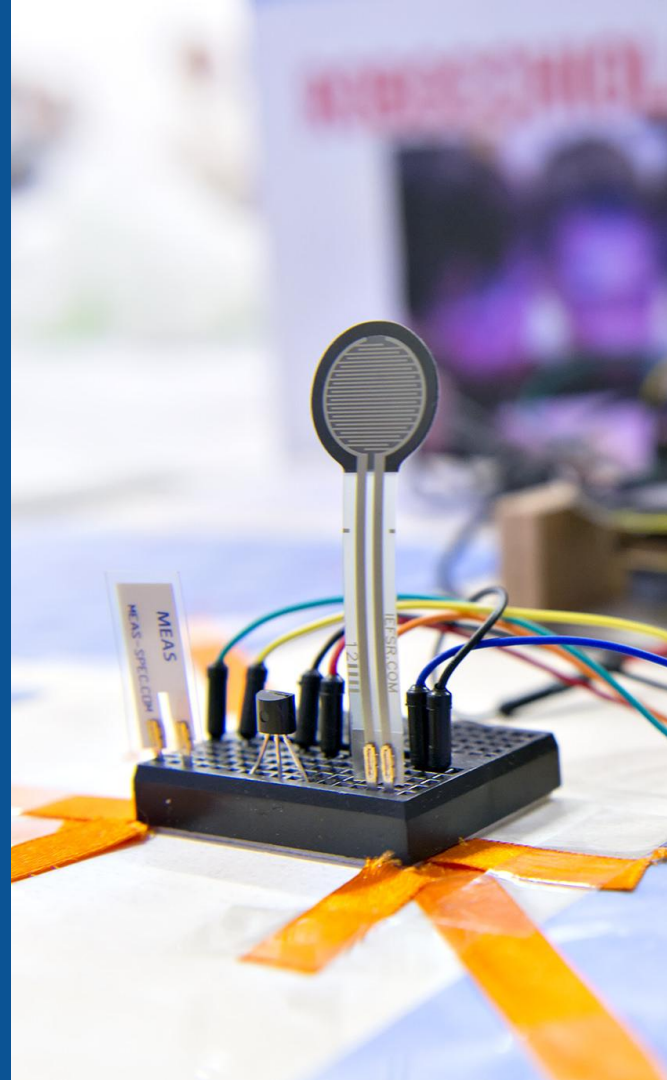
What *is* an Arduino?

- ❑ Open source microcontroller that serves as the brain behind hobbyist electronics projects.
- ❑ Designed with simplicity and rapid prototyping in mind.
- ❑ Driven by interaction design -- move away from focus on physical design to exploring how objects interact with things.



CULTURE OF MAKING

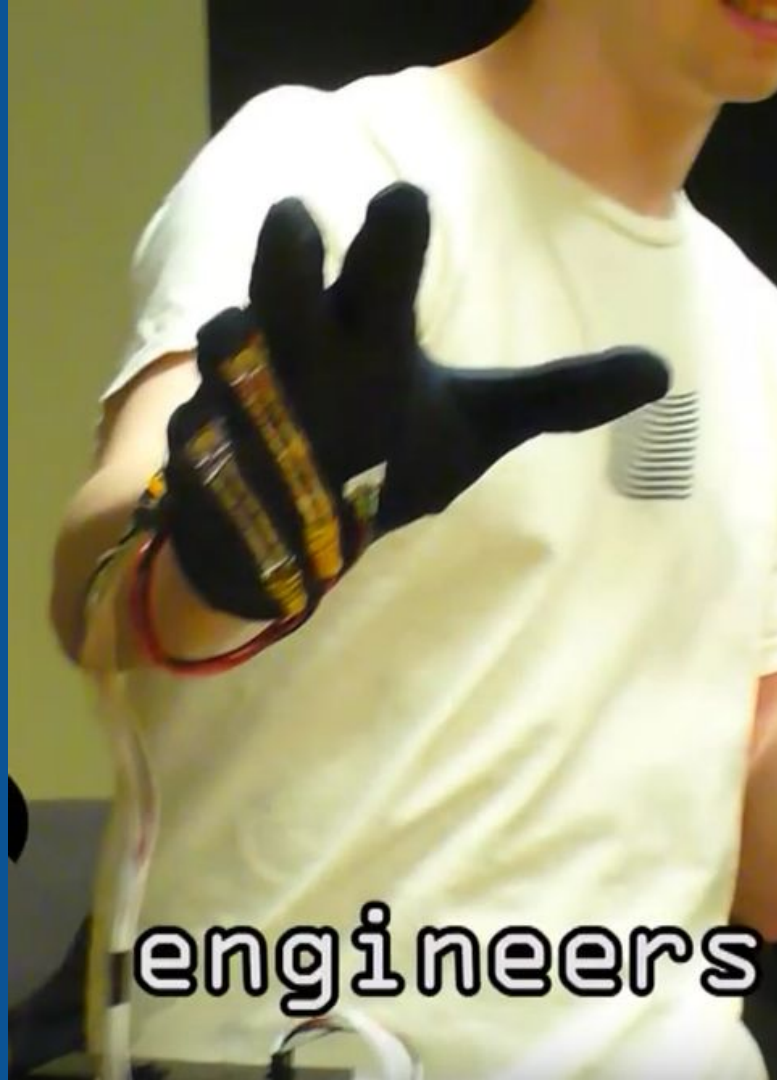
Sharing and learning together.





EXAMPLE:

"SudoGlove"



engineers



BOARDS:

ENTRY LEVEL

ARDUINO UNO

ARDUINO 101

ARDUINO PRO

ARDUINO PRO MINI

ARDUINO MICRO

ARDUINO NANO

ARDUINO STARTER KIT

ARDUINO BASIC KIT

ARDUINO MOTOR SHIELD

ENHANCED FEATURES

ARDUINO MEGA

ARDUINO ZERO

ARDUINO DUE

ARDUINO PROTO SHIELD

INTERNET OF THINGS

ARDUINO YÚN

ARDUINO MKR1000

ARDUINO ETHERNET SHIELD

ARDUINO GSM SHIELD

ARDUINO WIFI SHIELD 101

WEARABLE

ARDUINO GEMMA

LILYPAD ARDUINO USB

LILYPAD ARDUINO MAIN BOARD

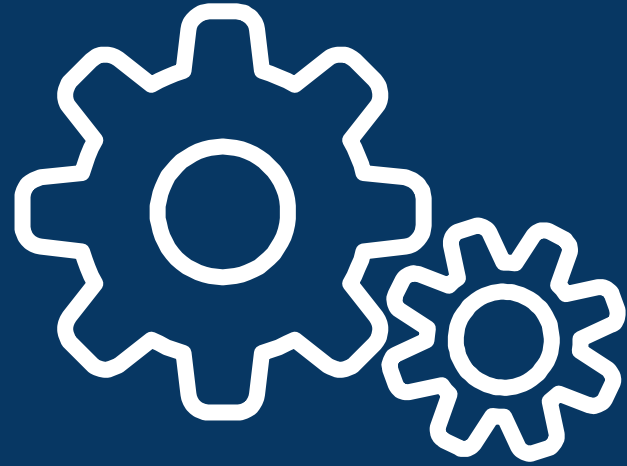
LILYPAD ARDUINO SIMPLE

LILYPAD ARDUINO SIMPLE SNAP

3D PRINTING

MATERIA 101

HOW



Turns input (touch, data, etc.) into output (turning on a motor, updating a social media feed, etc.)

DOES IT WORK?



SOFTWARE



THE LANGUAGE

Arduino Programming Language

- Based on Wiring (C++)

Arduino IDE

- Text editor-like program; based on Processing

A screenshot of the Arduino IDE interface. The window title is "sketch_jan01a | Arduino 1.0.3". The top toolbar includes a checkmark, a refresh icon, a file icon, an upload icon, and a download icon, followed by the text "Upload". The main editor area shows the following C++ code:

```
int ledPin = 13;

void setup()
{
  pinMode(ledPin, OUTPUT);
}

void loop()
{
  digitalWrite(ledPin, LOW);
}
```

The bottom status bar displays "Done uploading." and "Binary sketch size: 872 bytes (of a 32,256 byte maximum)".



PROGRAMMING SYNTAX

bit.ly/1o3cBTL

```
Programming_Language_Syntax | Arduino 1.6.5
Programming_Language_Syntax
//Basic syntax and structure for the Arduino programming language:

//<-- single line comment
/* <-- multi-line comment for use
 * when writing long blocks of text.
 */
//all non-commented code must have a semicolon ";" to run properly.

//Programs must have a "setup" and "loop" function to run properly.

void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}

//Variable Examples:

int myX; //assigns variable type only.
//or
int myX = 1; //assigns variable type (integer) & sets variable equal to the value of 1.
//or

Done Saving.
The sketch name had to be modified. Sketch names can only consist
of ASCII characters and numbers (but cannot start with a number).
They should also be less than 64 characters long.
```



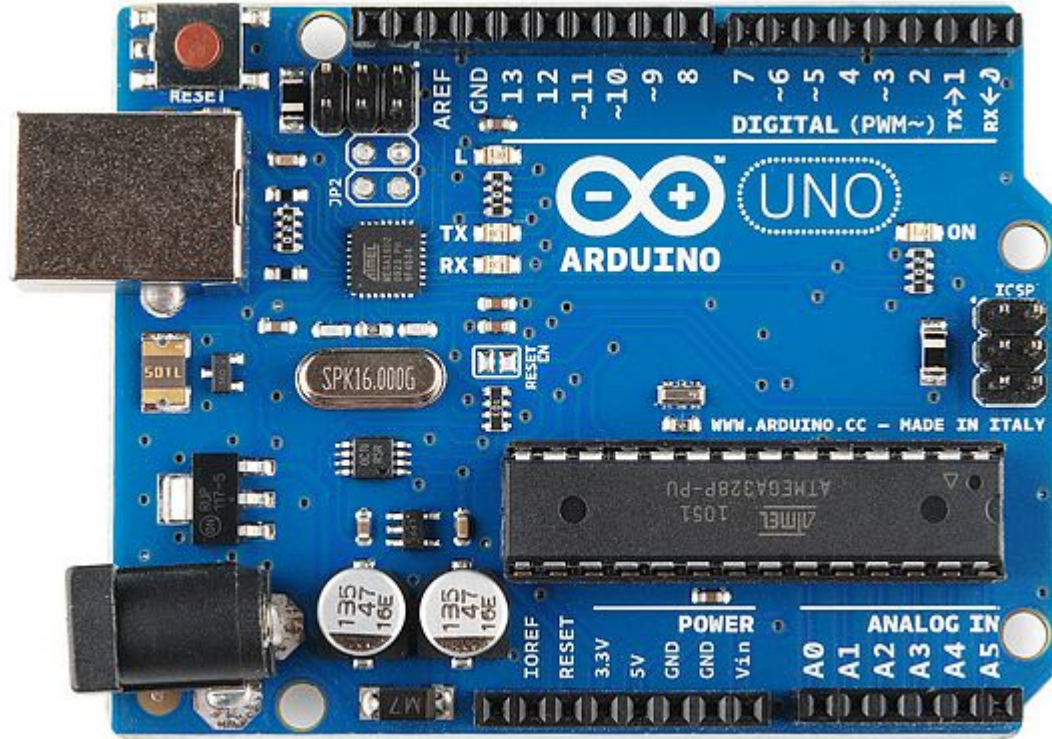

HARDWARE:

Sensors and Servos

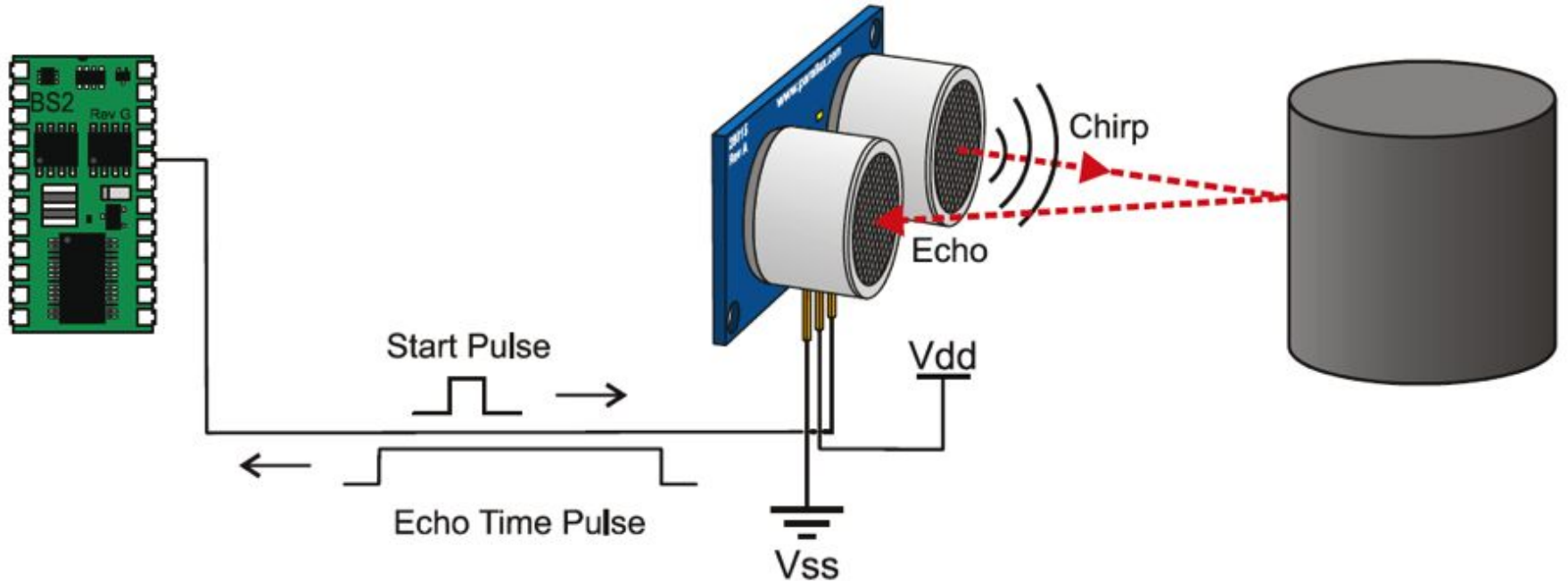
THE BOARD:

Important Parts:

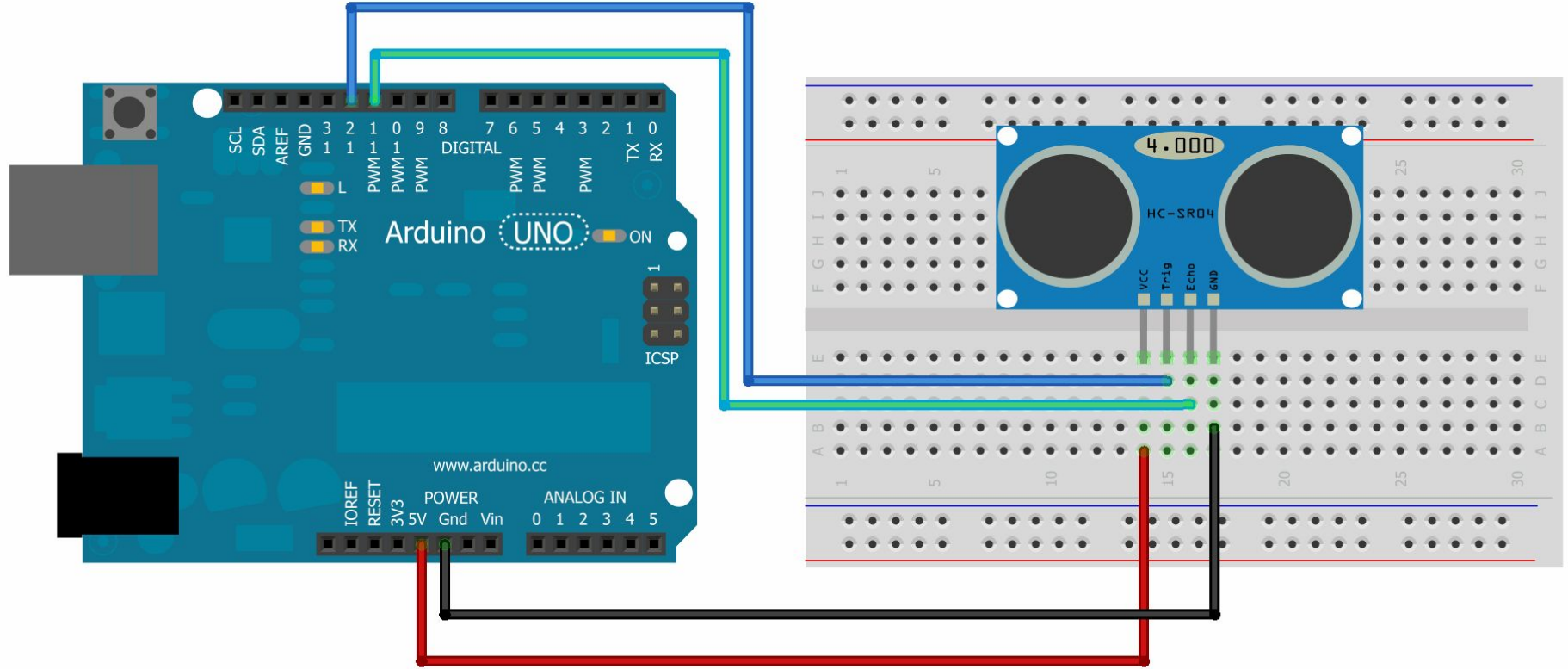
- GND
- 5V & 3.3V
- Analog pins
- Digital pins
- Pulse Width Modulation (~)
- Reset Button
- TX and RX symbols
- Integrated Circuit
- Voltage Regulator



ULTRASONIC DISTANCE SENSOR



PROTOTYPE LAYOUT



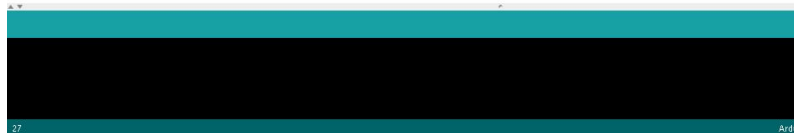
YOUR CHALLENGE:

```
UDS_test | Arduino 1.6.5
UDS_test
int trigPin = 8;
int echoPin = 7;

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

void loop(){
  long duration;
  float cm;
  pinMode(echoPin, INPUT);
  pinMode(trigPin, OUTPUT);
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  cm = microsecondsToCentimeters(duration);
  Serial.print(cm);
  Serial.print("cm");
  Serial.println();
  delay(100);
}

float microsecondsToCentimeters(long microseconds){
  return (microseconds*0.034029)/2;
}
```



TEST OR MODIFY
THIS CODE:

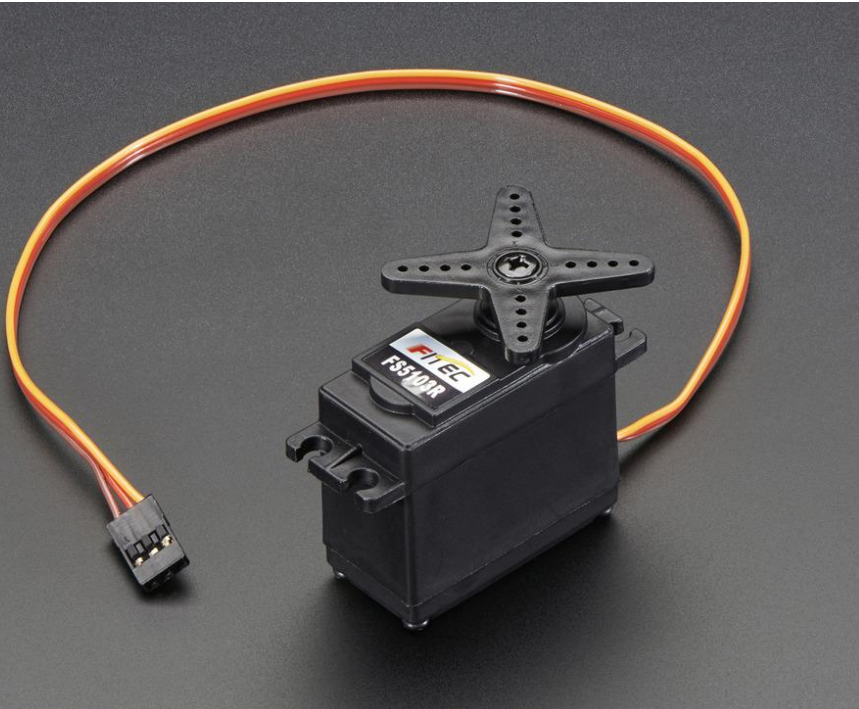
bit.ly/1nTOUgg

ARDUINO KIT COMPONENTS LIST:

bit.ly/1XiyTwu

SERVOS

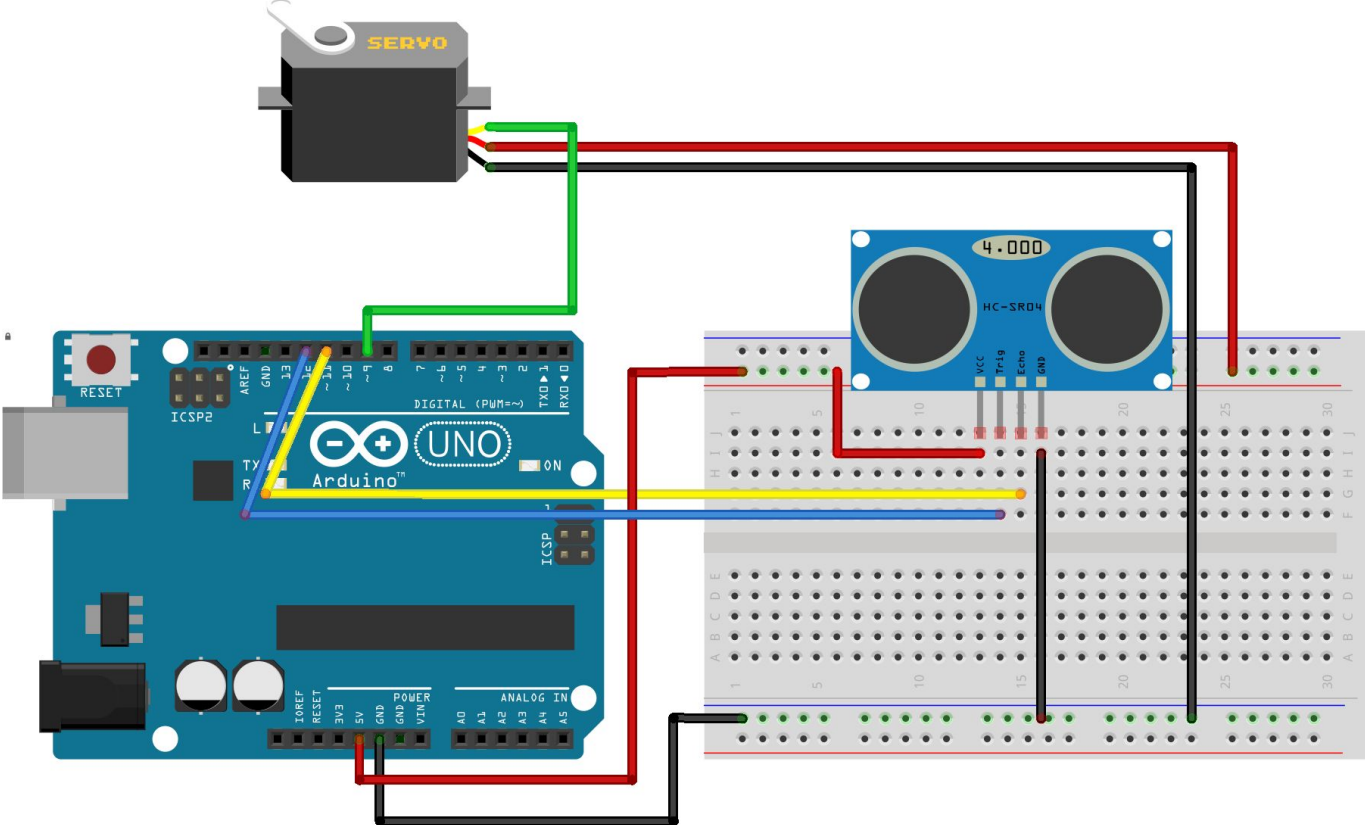
Continuous Rotation Servo Motor



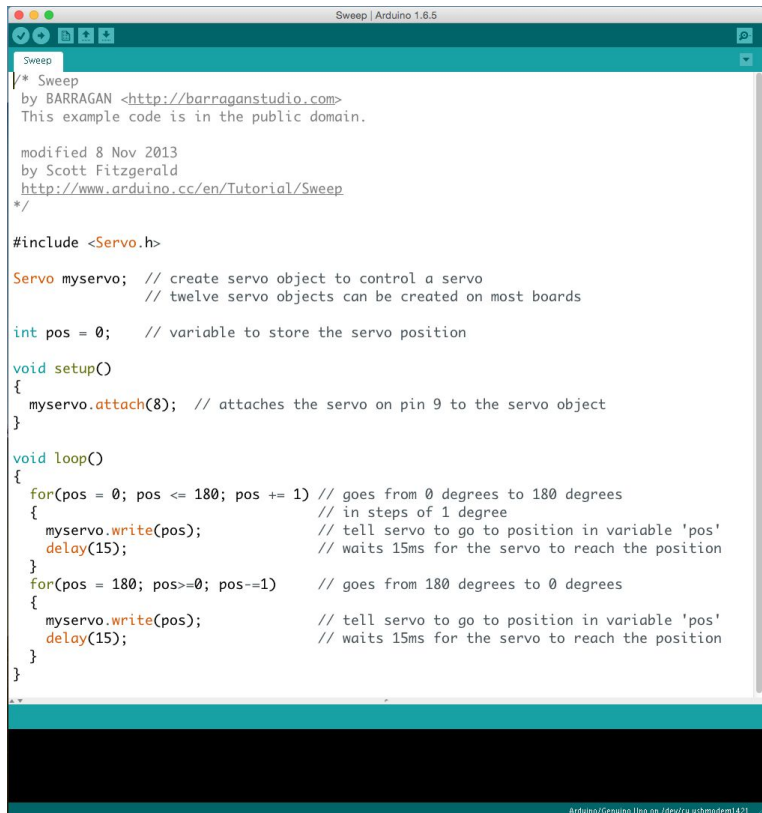
Microservo Motor



PROTOTYPE LAYOUT



YOUR CHALLENGE:

A screenshot of the Arduino IDE interface. The window title is "Sweep | Arduino 1.6.5". The code editor shows the following code:

```
Sweep
/* Sweep
  by BARRAGAN <http://barraganstudio.com>
  This example code is in the public domain.

  modified 8 Nov 2013
  by Scott Fitzgerald
  http://www.arduino.cc/en/Tutorial/Sweep
  */

#include <Servo.h>

Servo myservo;  // create servo object to control a servo
                // twelve servo objects can be created on most boards

int pos = 0;    // variable to store the servo position

void setup()
{
  myservo.attach(8); // attaches the servo on pin 9 to the servo object
}

void loop()
{
  for(pos = 0; pos <= 180; pos += 1) // goes from 0 degrees to 180 degrees
  {
    // in steps of 1 degree
    myservo.write(pos);              // tell servo to go to position in variable 'pos'
    delay(15);                       // waits 15ms for the servo to reach the position
  }
  for(pos = 180; pos>=0; pos--1)    // goes from 180 degrees to 0 degrees
  {
    myservo.write(pos);              // tell servo to go to position in variable 'pos'
    delay(15);                       // waits 15ms for the servo to reach the position
  }
}
```

TEST OR MODIFY THIS CODE:

File -> Examples ->
Servo -> Sweep

OR go a step
further:

bit.ly/1V3VA6i

RECOMMENDED READING:

- ❑ [What is electricity?](#)
- ❑ [A First Lab in Circuits and Electronics](#)
- ❑ [All About Circuits](#)
- ❑ [Arduino “How To” e-books](#)
- ❑ [SparkFun PCB Basics Guide](#)
- ❑ [Software: Fritzing and LTSpice](#)



GET HELP

Science and Engineering Libraries:

ref-sci@columbia.edu or jcb2257@columbia.edu

Arduino [project guidance](#) forum

[Project books](#)

[Stack Exchange](#)

[Reddit forum](#)

SURVEY!

bit.ly/CUSELWorkshopSurvey

Other questions? Comments? You can find me at
@jeninthelib & jcb2257@columbia.edu

THANK YOU!

Special thanks to all the people who made and released these awesome resources for free:

- ! Presentation template by [SlidesCarnival](#)
- ! Photographs by [Unsplash](#)