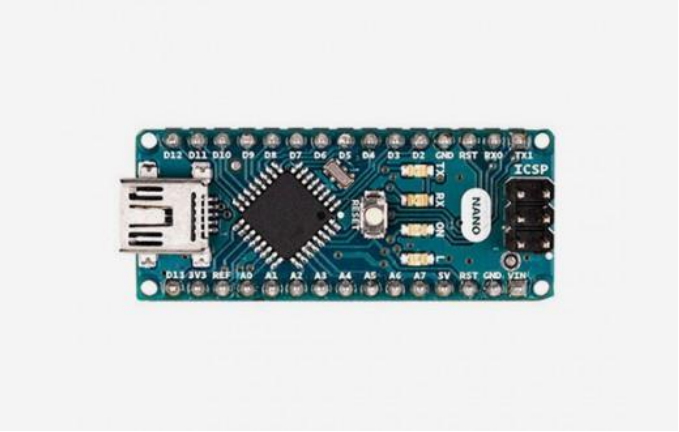
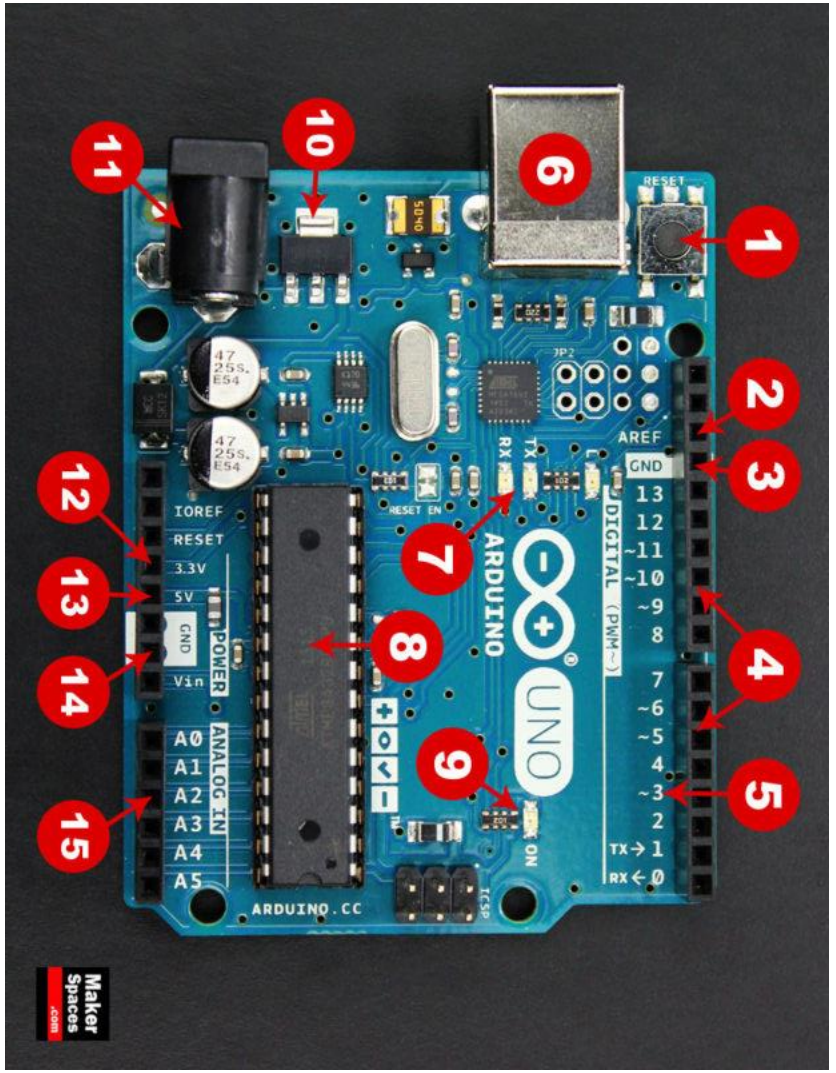
Arduino: Basic Ideas

Basic ideas of Arduino, Familiarize the Arduino board, Setting up the arduino board. Installation of IDE in PC/ laptop for Arduino programming (Sketch)



https://www.arduino.cc/en/Main/Products

Arduino UNO R3



1. Reset Button – This will restart any code that is loaded to the Arduino board

2. AREF – Stands for "Analog Reference" and is used to set an external reference voltage

3. Ground Pin – There are a few ground pins on the Arduino and they all work the same

4. Digital Input/Output – Pins 0-13 can be used for digital input or output

5. PWM – (Pulse Width Modulation) The pins marked with the (~) symbol can simulate analog output

https://www.makerspaces.com/arduino-uno-tutorial-beginners/

Arduino UNO R3

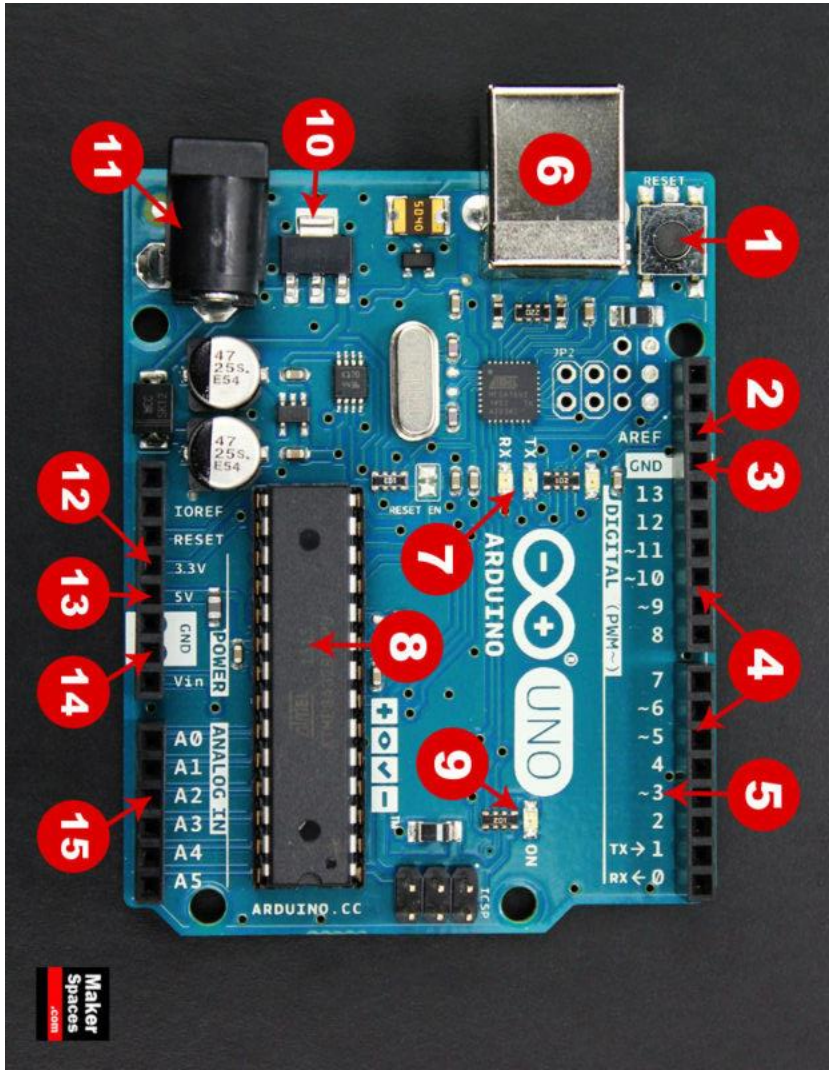
6. USB Connection – Used for powering up your Arduino and uploading sketches

7. TX/RX – Transmit and receive data indication LEDs

8. ATmega Microcontroller – This is the brains and is where the programs are stored (ATmega328P, 16 MHz)

9. Power LED Indicator – This LED lights up anytime the board is plugged in a power source

10. Voltage Regulator – This controls the amount of voltage going into the Arduino board



https://www.makerspaces.com/arduino-uno-tutorial-beginners/

Arduino UNO R3

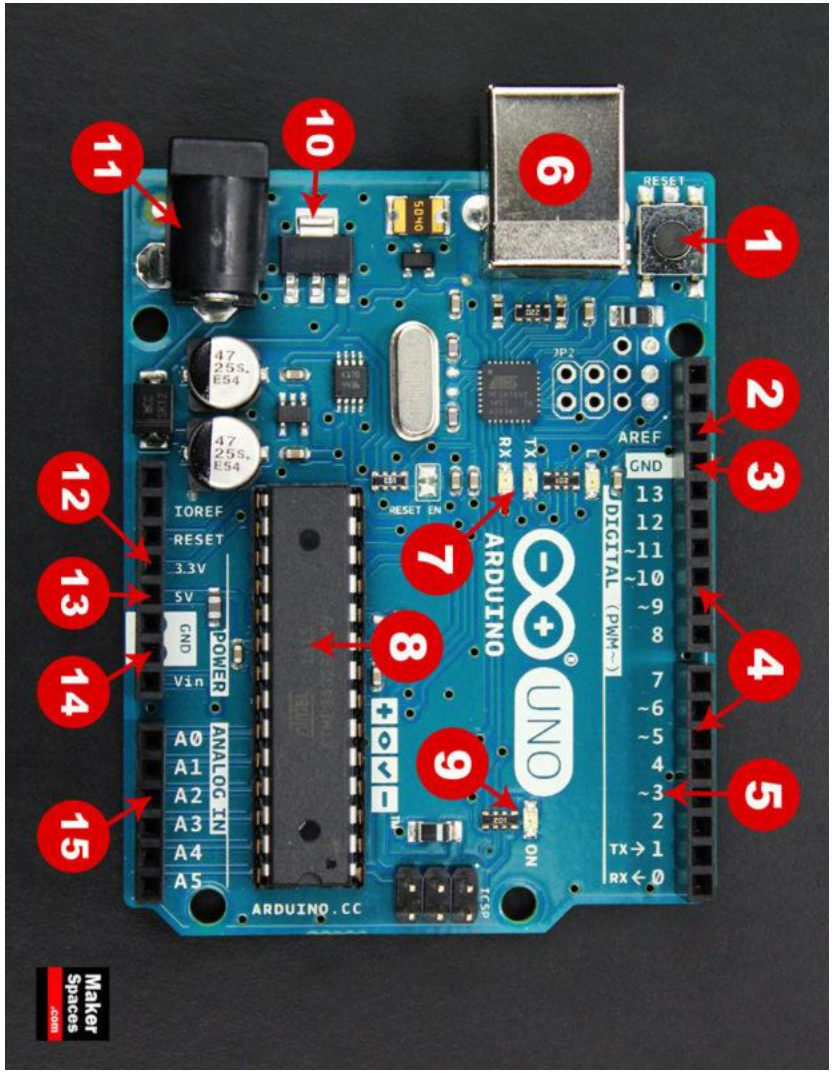
11. DC Power Barrel Jack – This is used for powering your Arduino with a power supply

12. 3.3V Pin – This pin supplies 3.3 volts of power to your projects

13. 5V Pin – This pin supplies 5 volts of power to your projects

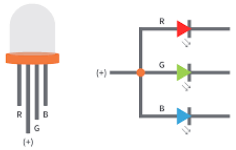
14. Ground Pins – There are a few ground pins on the Arduino and they all work the same

15. Analog Pins – These pins can read the signal from an analog sensor and convert it to digital

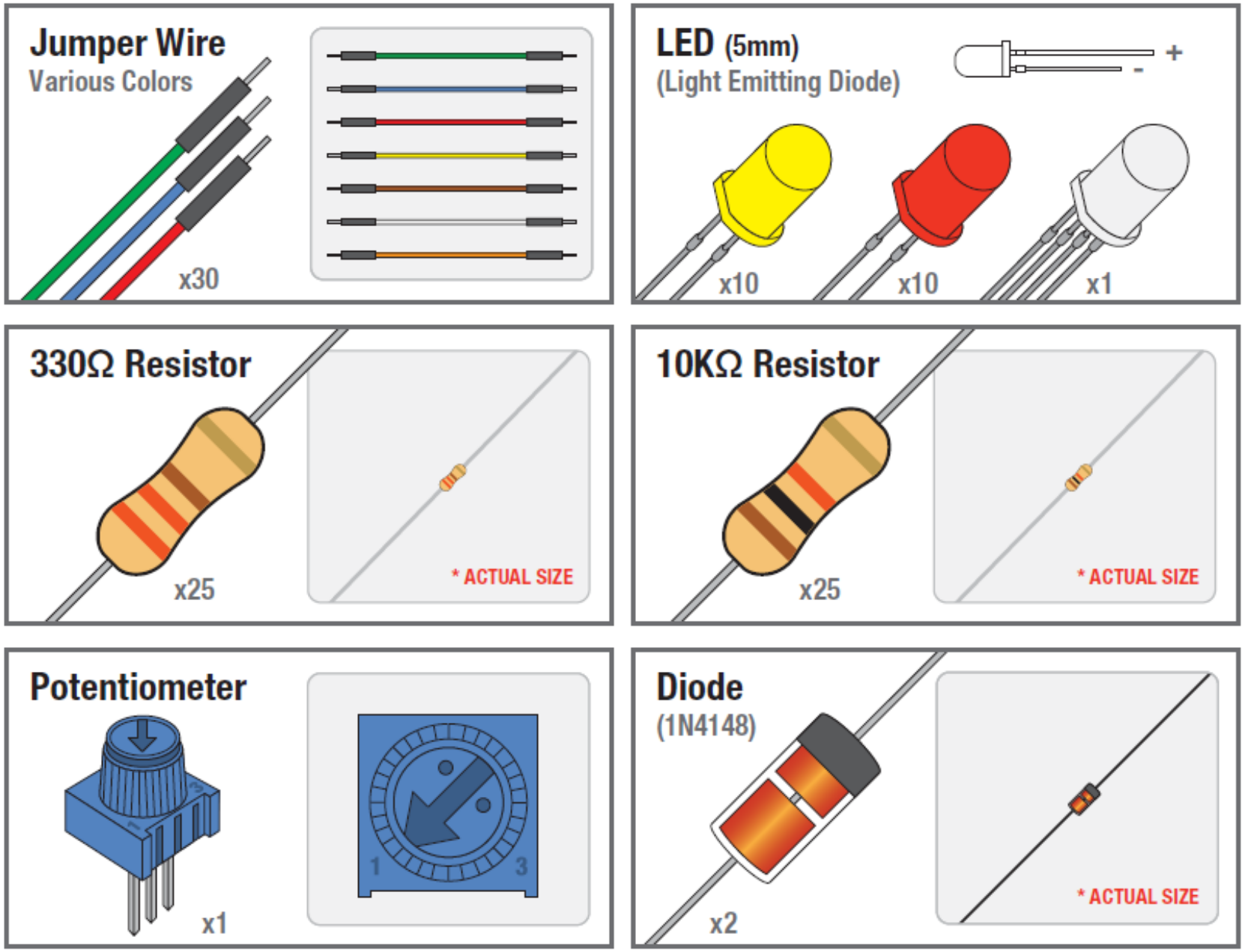


https://www.makerspaces.com/arduino-uno-tutorial-beginners/

Kit Components

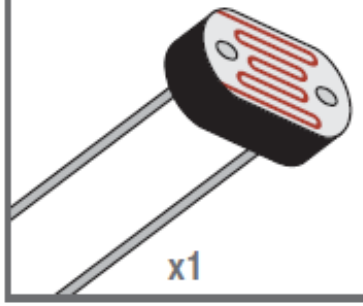
Name	Image	Type	Function	Notes
Push Button		Digital Input	Switch - Closes or opens circuit	Polarized, needs resistor
Trim potentiometer		Analog Input	Variable resistor	Also called a Trimpot.
Photoresistor		Analog Input	Light Dependent Resistor (LDR)	Resistance varies with light.
Relay		Digital Output	Switch driven by a small signal	Used to control larger voltages
Temp Sensor		Analog Input	Temp Dependent Resistor	
Flex Sensor		Analog Input	Bending dependent resistor	
RGB LED		Dig & Analog Output	16,777,216 different colors	

SIK Components



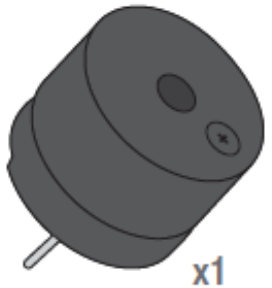
SIK Components

Photo Resistor

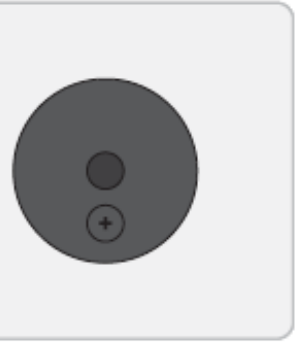


x1

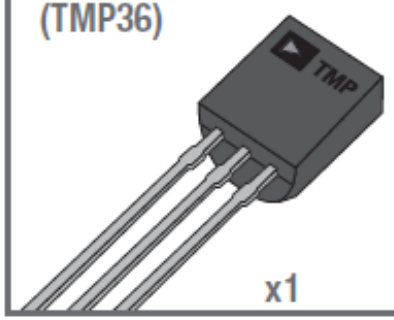
Piezo Element



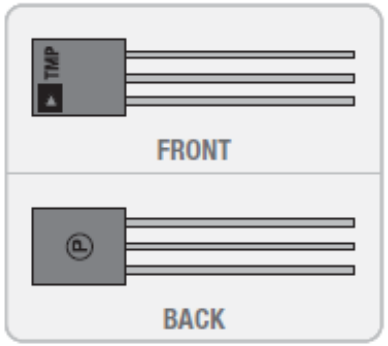
x1



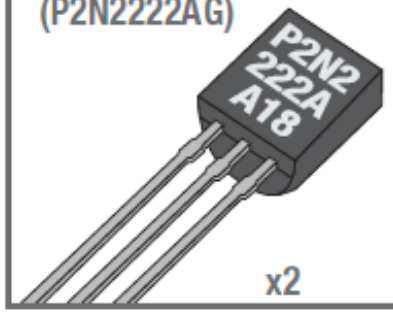
Temp. Sensor
(TMP36)



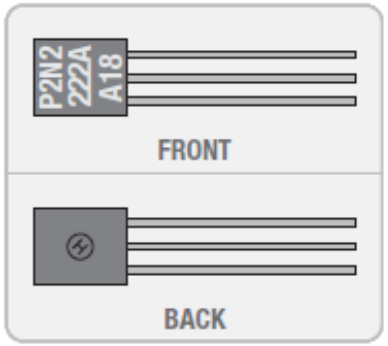
x1



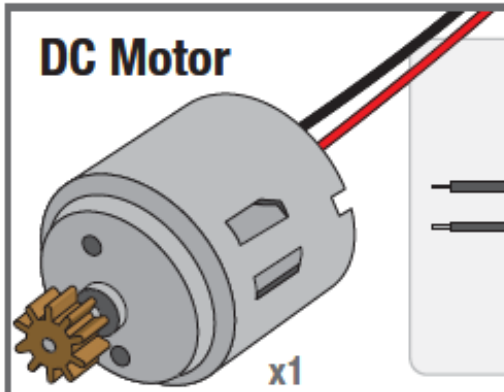
Transistor
(P2N2222AG)



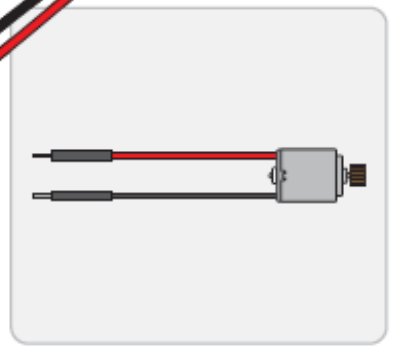
x2



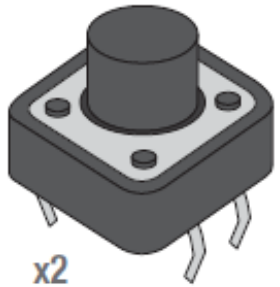
DC Motor



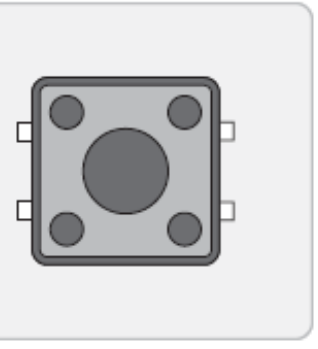
x1



Push Button

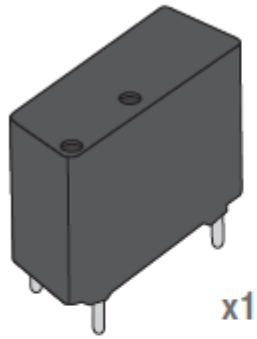


x2



SIK Components

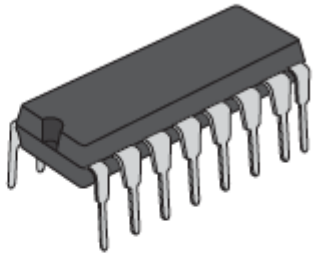
Relay



x1

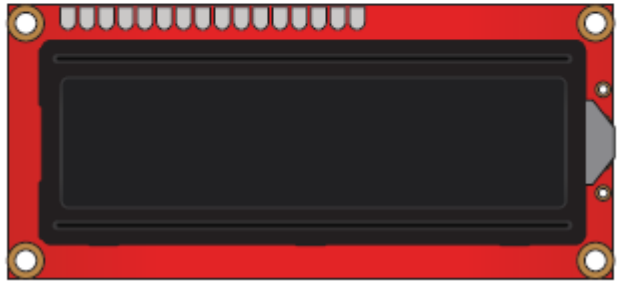
Integrated Circuit

(IC)



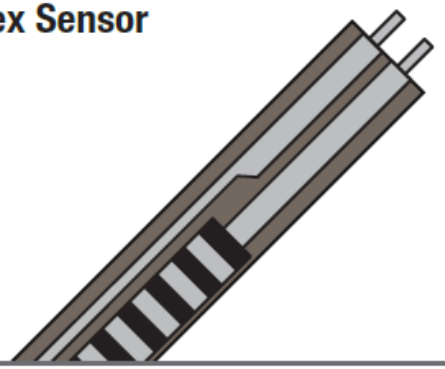
x1

LCD



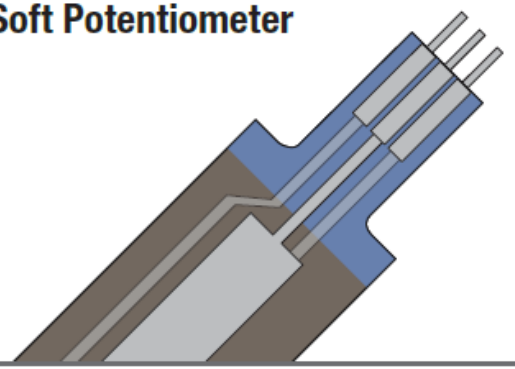
x1

Flex Sensor



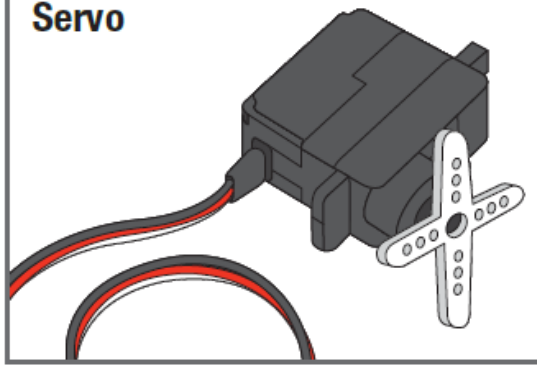
x1

Soft Potentiometer



x1

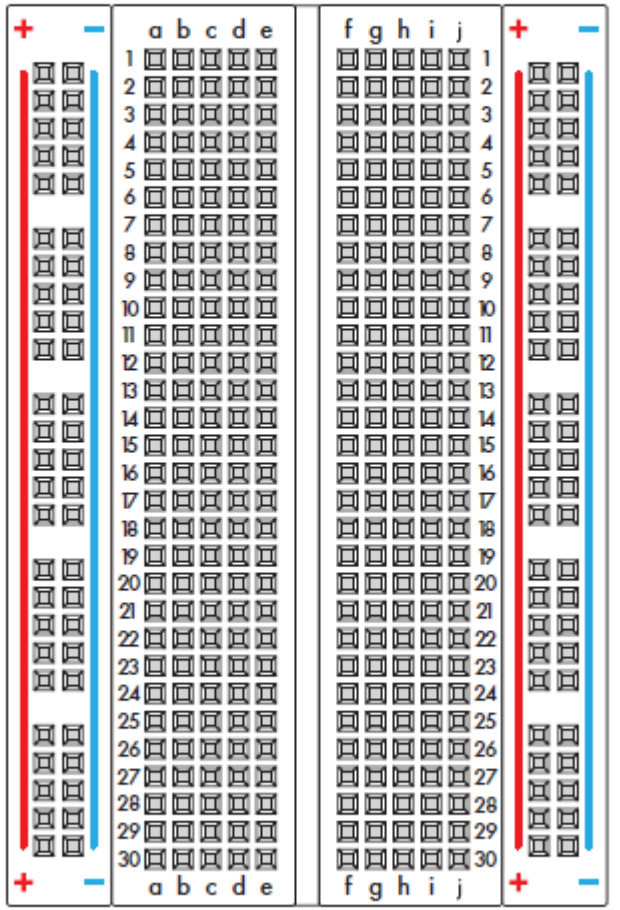
Servo



x1

Breadboard

Standard Solderless (Color may vary)



x1

Platform

Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so one uses the Arduino programming language (based on Wiring, an open-source programming framework for microcontrollers), and the Arduino Software (IDE – Integrated Development Environment), based on Processing (an open-source software sketchbook and a language for learning how to code within the context of the visual art).

https://www.arduino.cc/en/Guide/Introduction

Getting Started

- Set up the Arduino Software (IDE) to program your board.
- Two options: (a) Online IDE (Arduino Web Editor) – requires stable internet connection, allows to save sketches in the cloud thus enabling the user to access from any device, have the most up-to-date version of the IDE without the need to install updates or community generated libraries.

(b) Desktop IDE – Windows, Mac OS X, Linux, Portable IDE (Windows ↔ Linux), ChromeOS

Arduino Software (IDE)

https://www.arduino.cc/en/Guide/Environment