Beginning Arduino Class for Carlsbad Library

Stephen Hicks

February 18, 2017

Outline

Arduino Overview

Intro to Circuits

Intro to Programming

Extras, where to go from here

Exercises

Circuit I/O Links

What is an Arduino?

- A small, low cost microcomputer for creating electronics projects
- A collection of open source software programs and hardware designs to make that task easier

Arduino Board part 1

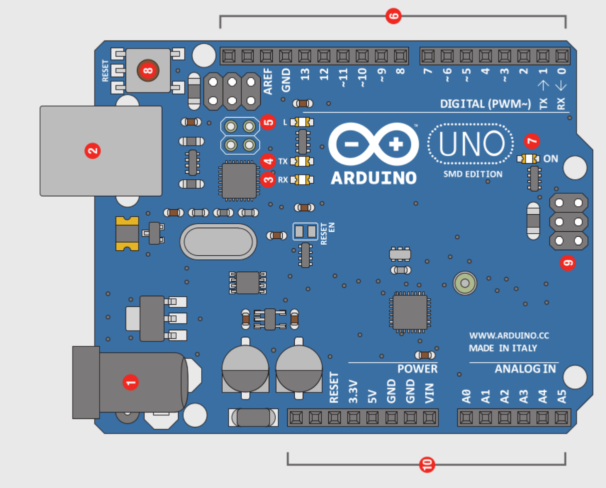
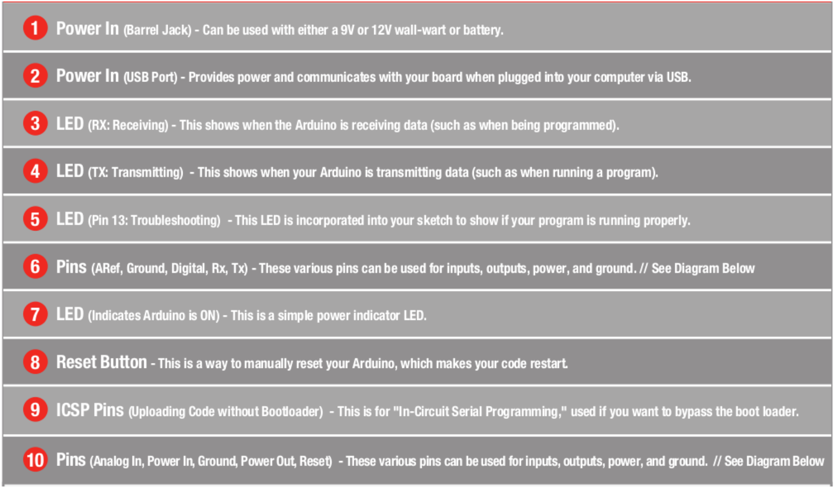


Figure: Arduino Overview

Arduino Board part 2



1 **Power In** (Barrel Jack) - Can be used with either a 9V or 12V wall-wart or battery.

2 **Power In** (USB Port) - Provides power and communicates with your board when plugged into your computer via USB.

3 **LED** (RX: Receiving) - This shows when the Arduino is receiving data (such as when being programmed).

4 **LED** (TX: Transmitting) - This shows when your Arduino is transmitting data (such as when running a program).

5 **LED** (Pin 13: Troubleshooting) - This LED is incorporated into your sketch to show if your program is running properly.

6 **Pins** (ARef, Ground, Digital, Rx, Tx) - These various pins can be used for inputs, outputs, power, and ground. // See Diagram Below

7 **LED** (Indicates Arduino is ON) - This is a simple power indicator LED.

8 **Reset Button** - This is a way to manually reset your Arduino, which makes your code restart.

9 **ICSP Pins** (Uploading Code without Bootloader) - This is for "In-Circuit Serial Programming," used if you want to bypass the boot loader.

10 **Pins** (Analog In, Power In, Ground, Power Out, Reset) - These various pins can be used for inputs, outputs, power, and ground. // See Diagram Below

Connecting your Arduino to your Computer

- Connect the Arduino to the pc via the USB connector on the side of the laptop
- The green "on" LED should light up

The Arduino Integrated Development Environment (IDE)

- Search for "Arduino" on your pc or look on the starter to launch the program
- Select your board type (UNO) under Tools->Board
- Select your serial (COM) port under Tools->Serial Port (device may already be selected)
- If you are not sure which Port it is unplug the Arduino and see which port (e.g. COM 6) when you plut it back in.

Creating and Load a empty program

- Select File->New
- Type these lines

```
void setup(){
}
void loop(){
}
```

- Select Sketch-> Verify/Compile to build your program
- Verify no error Messages
- Select File->Upload to load your program
- Verify it loads correctly

Brief Tour of the Arduino IDE

- Verify/Compile and Upload Buttons
- Examples and Libraries
- Serial Monitor
- Import Library

What is a circuit? What is Ohms Law?

- A path the current follows through one or more electronic components
- Ohms Law V = IR is the basic equation
- To analyze a circuit we follow the current from high to low voltage

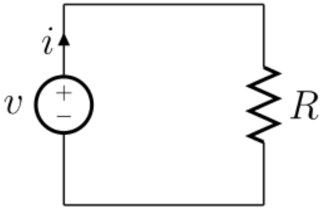


Figure: Basic Resistor Circuit

Solderless Breadboard

- Allows you to connect electronic components without solder
- rows of holes form a "bus" where all wires are connected together
- 4 long rows on outside for power/ground connections
- shorter rows on the inside for components

Breadboard Overview

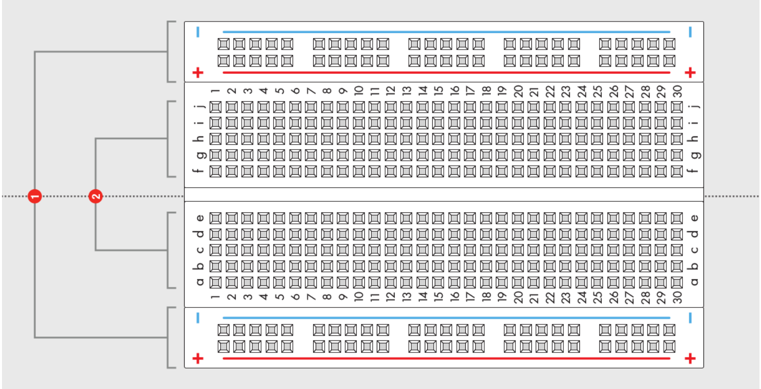
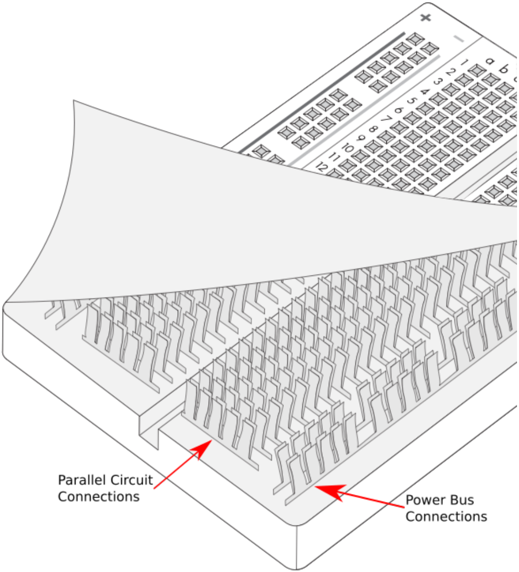


Figure: Breadboard top view

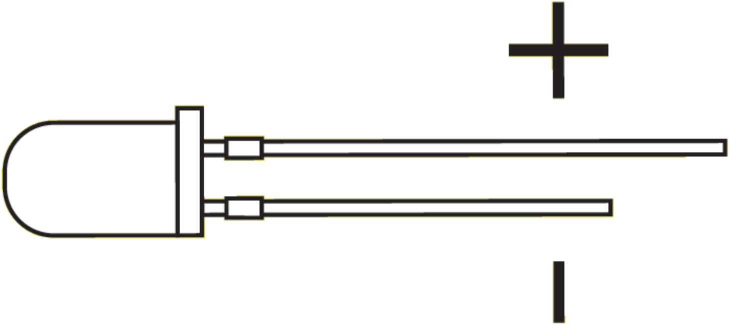
Breadboard Internal View



Electronic Components and Polarity

- Some components can be put in either direction (resistor, inductor, and some capacitors)
- Other components have a "polarity" or "correct orientation" and the parts can only be put in one way
- Diodes, LED, Integrated Circuits, Transistors all have polarity or correct orientation
- There are different marking schemes based on the component
- When in doubt use google to look up the orientation, you can damage your component or board by inserting with the incorrect polarity

LED Polarity



Make your first Arduino LED Circuit

- Prebending the resistor will make it easier to insert in the breadboard
- Be sure to check the orientation of the LED

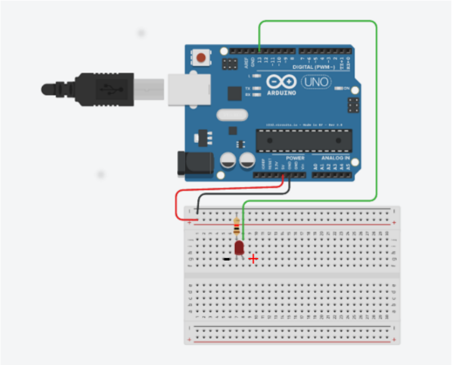


Figure: LED Breadboard Circuit

Your first program to control the LED Circuit

```
/*
Blink Program
*/
int led = 13;  // Pin 13 has an LED
// 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 voltage level)
  delay(1000);               // wait for a second
  digitalWrite(led, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);               // wait for a second
}
```

How to get started with your own Arduino setup

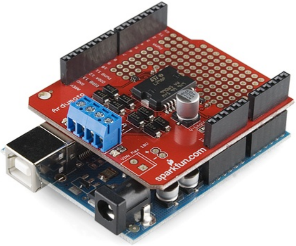
- For beginners with no electronics experience a kit from a reputable vendor is a good way to start.
- Probably better to buy a less expensive kit up front and get components and modules as you need them
- Sparkfun, Adafruit, Arduino all sell good kits
- Vilros has some cheaper kits which work fine
- Ebay, Fasttech, Gearbest, Tayda, Amazon are all lower cost vendors when you get more familiar with what you are doing.

Other nice things to have for tinkering with electronics

- Digital Multimeter is very nice for checking connectivity, voltage, and current
- Again buy a less expensive one and use the money you saved to buy clip on leads which are easier for most testing than the pointy ones that come with your DMM.
- USB Testers are an easy way to check your USB voltage and see how much current your Arduino is taking. I have a Drok, there are others
- There are lots of other things as you want to do more complicated projects. EEVBlog has some videos which cover many options.

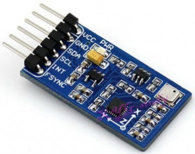
Arduino Shields

- Family of add on boards that can be used to create additional functionality
- Plug into the top connector of your Arduino
- WiFi, Ethernet, Sensors, Motor Controllers are just some examples



Modules

- Small boards that connect up to your Arduino using wires discrete
- Sometimes you need to solder a "header" connecter to be able to plug into your breadboard
- Gyros, Range Sensors, Bluetooth are examples of modules you can buy



Resources

- https://www.arduino.cc/ Arduino Project Homepage, many code examples, libraries, forum
- https://www.ted.com/talks/massimo_banzi_how_arduino_is_open_sourcing_imagination?language=en TED Talk by Massimo Banzi (Arduino Creator)
- http://fritzing.org/home/ Open source software for creating breadboard circuits
- https://circuits.io/ Autodesk Web tool for creating and testing circuits with simulated Arduino

Exercise 1: One LED on, One LED off

- Can you modify the circuit and program have two blinking LED's, one on while the other one is off.
- We would like to use pin 10 and pin 11 instead of pin 13.
- Circuit Hint 1: Get a second LED and Resistor and copy the connection from the negative LED terminal to Ground
- Circuit Hint 2: Disconnect wire to Arduino Pin 13. Reconnect to Pin 10. Take Second LED Positive Pin and
- Programming Hint 1: Copy int led lines and change to correct pin numbers
- Programming Hint 2: Copy pinMode line in setup and change pin numbers
- Programming Hint 3: Add two additonal digitalWrite commands in loop, change pin numbers.

Exercise 1: Circuit Modification - two LEDs

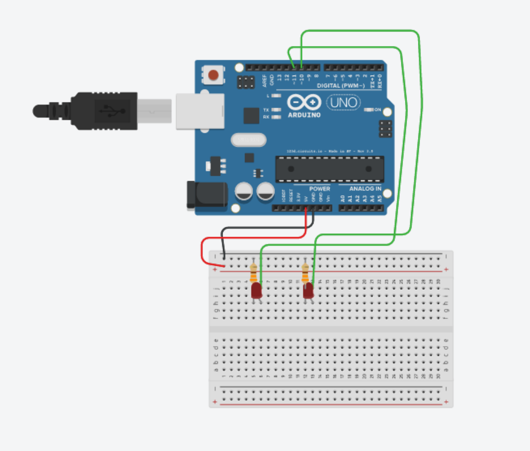


Figure: 2 LED Breadboard Circuit

Exercise 1: Code Modification

```
int led = 11;
int led2 = 10;
// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led, OUTPUT);
  pinMode(led2, OUTPUT);

}
// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led, HIGH);   // turn the LED on (HIGH is the voltage level)
  digitalWrite(led2, LOW);   // turn the LED on (HIGH is the voltage level)
  delay(1000);               // wait for a second
  digitalWrite(led, LOW);    // turn the LED off by making the voltage LOW
  digitalWrite(led2, HIGH);   // turn the LED on (HIGH is the voltage level)

  delay(1000);               // wait for a second
}
```

Circuit I/O Links to these projects

- All of the breadboard drawings and source code is available on circuits.io website.
- Search for the user "mcshicks" in the search bar
- Look for "Fleet Class Blink" circuits
- You can run and see the projects with no login.
- You will need to create a free user account if you want to duplicate the projects to make changes.