

Robotics/Electronics Lesson 2

Overview

In this lesson we will learn we learn how to wire a led and write code to make it blink.

Plan

1. Review previous lesson by doing a worksheet.
2. Led, resistors, breadboard, jump wires, pins
3. Wiring a circuit to light a led
4. Blinking led

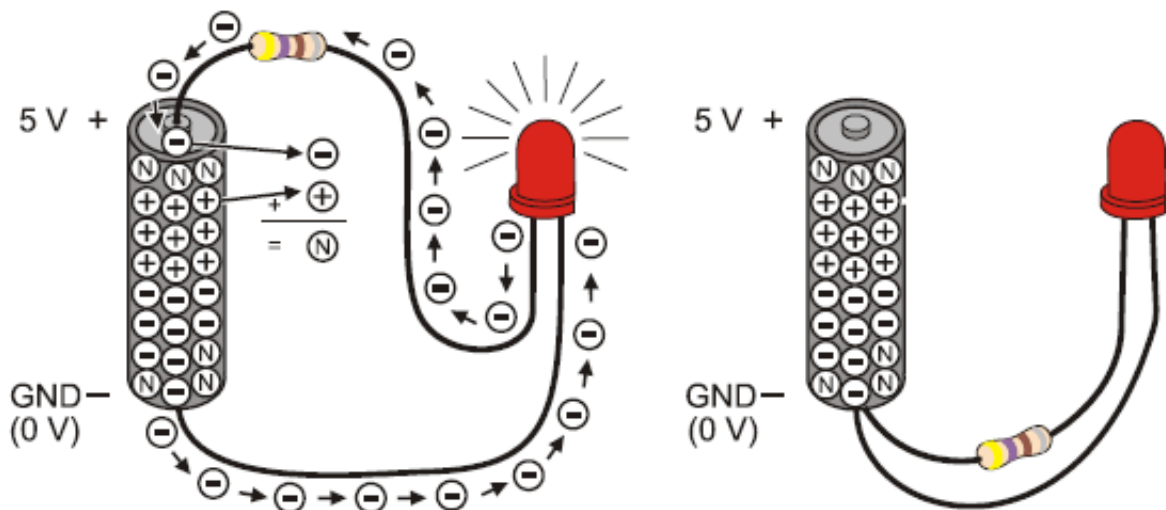
Activity 1: Review Lesson 1

1. The instructor and assistants will work with students to complete a worksheet (or review as a class) on lesson 1 topics.

The LED

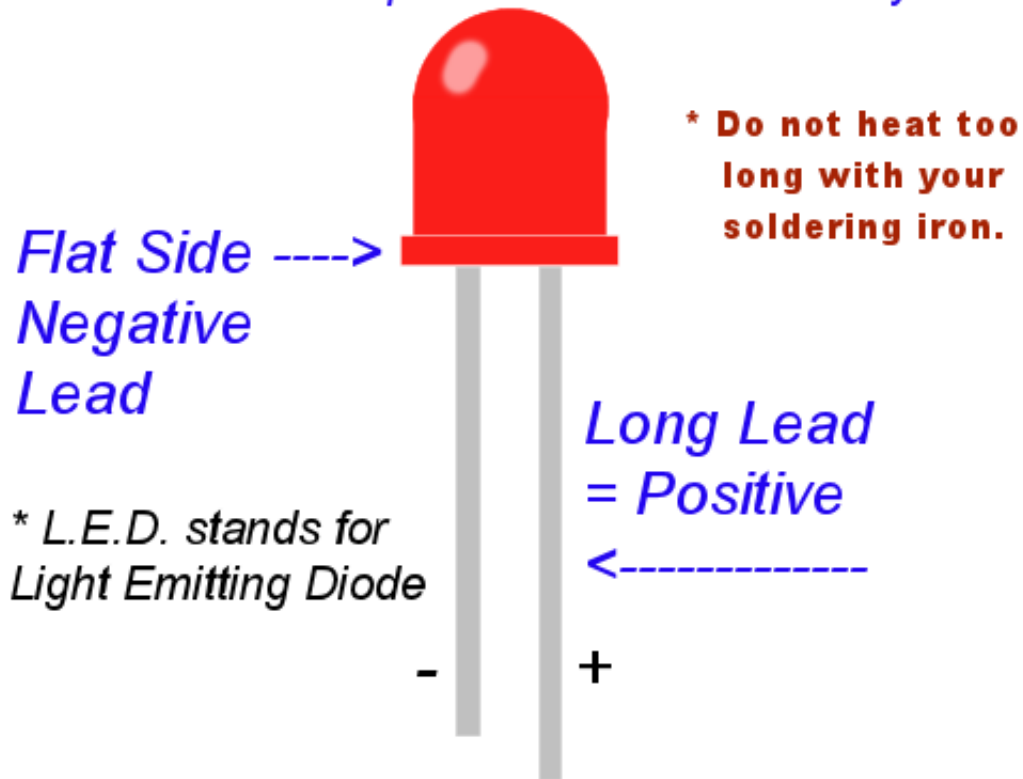
A led flows in one direction. From positive to negative. Positive side wants electrons and negative side wants to give away electrons.

1. Ask students which leg of led should connect to the negative side of the battery.



LEDs have Polarity

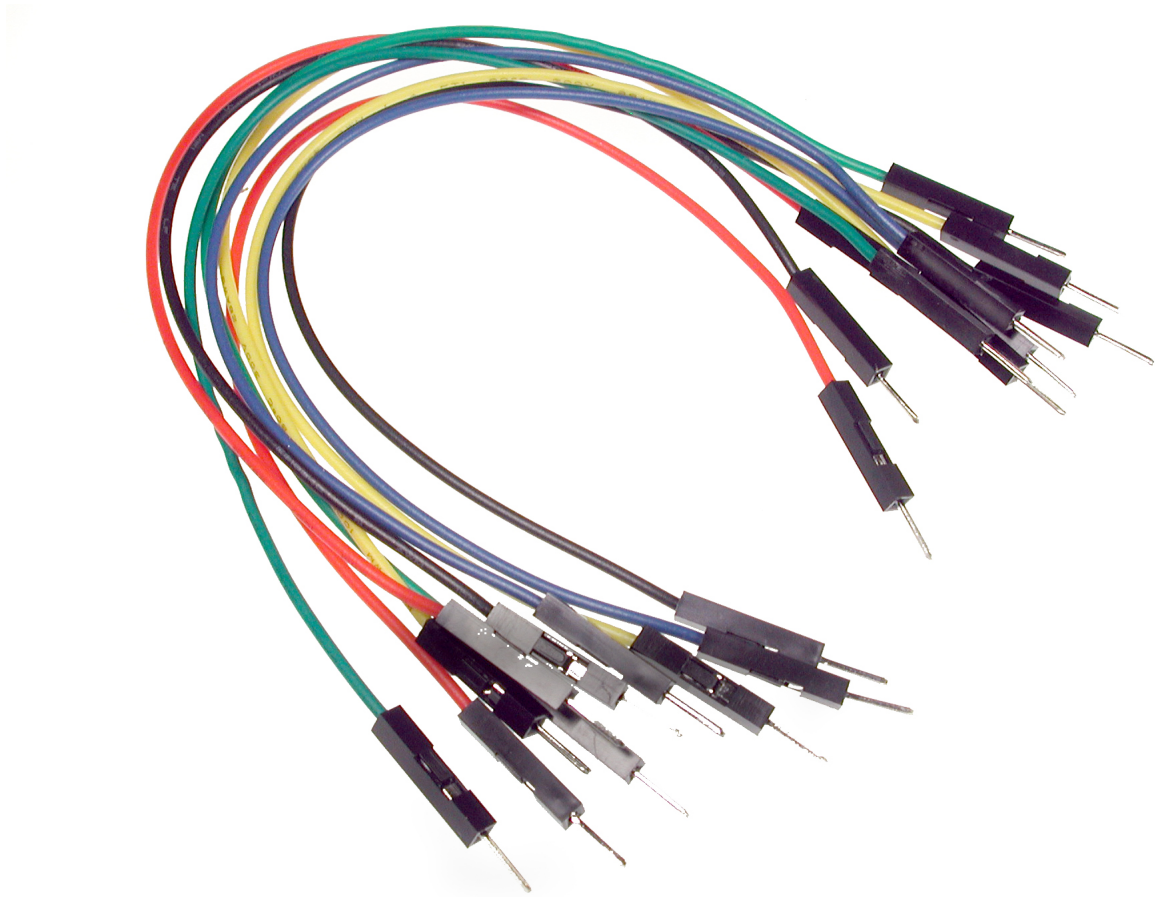
Do Not Connect Up Backwards or It Will Fry.



As a rule, connect your 480 Ohm resistor to the Long Lead (Positive side).

The Jumper Wire

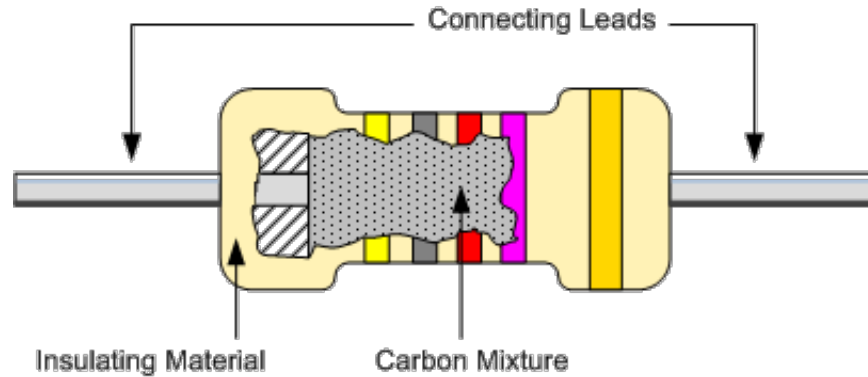
A Jumper wire connects a circuit. Electrons travel through the jumper wire.



The Resistor

A battery generates lots of electric current which can burn a led. The purpose of the resistor is to limit the current. Wood is a bad conductor of electricity and goal is a good conductor of electricity. A high resistor will light the light.

1. Show the resistor in the kit
2. Give examples of high (gold) and low resistors (wood)
3. There are different levels of resistors which are controlled by insulators (analogy would be the amount of wood)

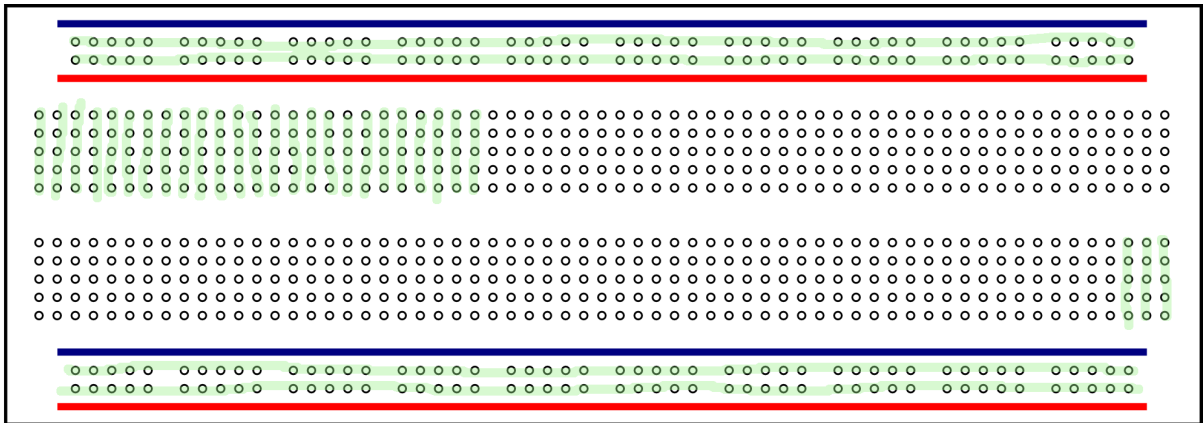


Activity 2: Burning lightbulb

1. An assistant will act as a light bulb and another assistant/instructor will act as a battery.
2. Passing a ping pong really fast will burn the light bulb.
3. Another assistant will be a resistor. He/she will slow down the passing the ping pong ball.
4. The light bulb will light up now.
5. What happens if the resistor is really really slow? Light bulb will be dim.
6. What happens if the resistor doesn't pass the electron? Light bulb will be off.

The breadboard

A breadboard is used to build circuits so you don't have to tie a lot of wires together. The picture below shows that if you were to provide electricity to one of the dots it will provide electricity to all the dots in green that are connected. So in the middle of breadboard all horizontal dots in one column will have electricity but not the columns next to it unless to connect a wire from one column to the next.

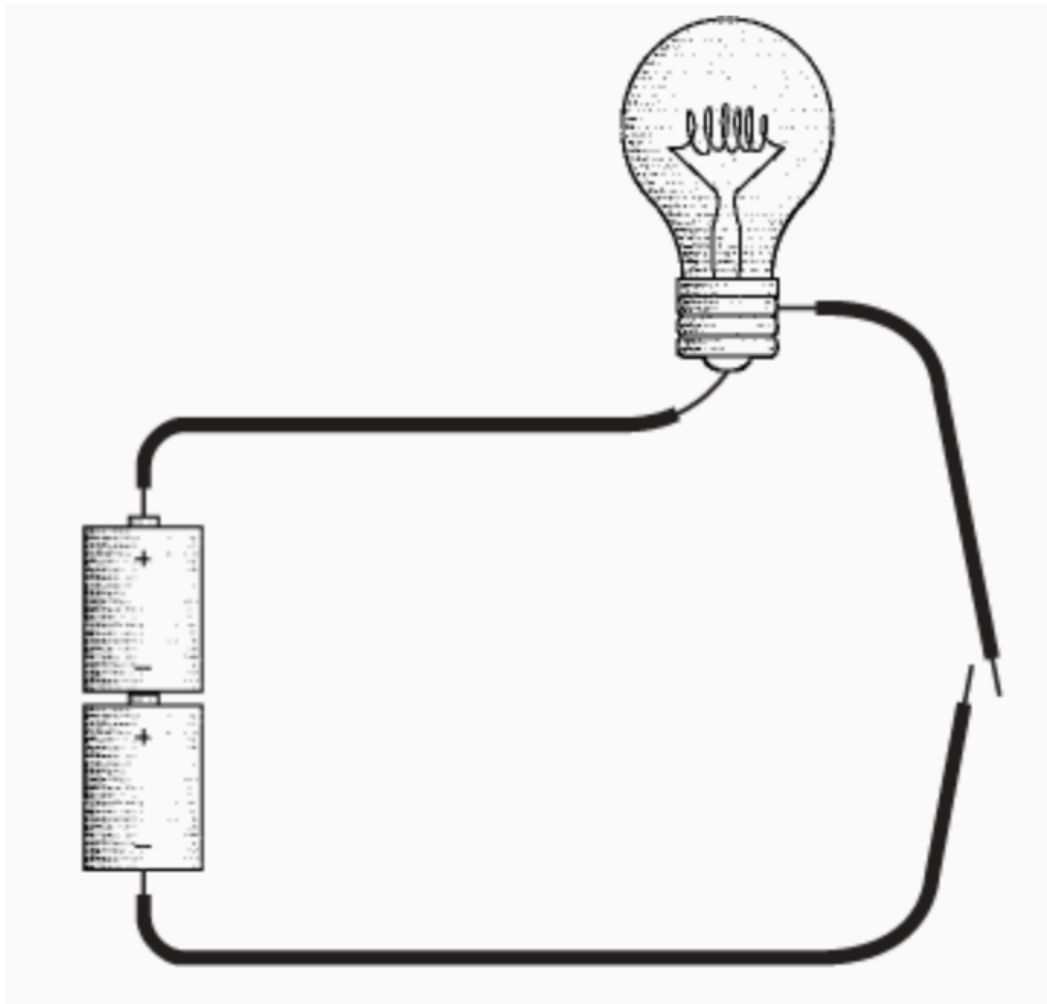


Activity 3: Human Breadboard

Students will be the o's in the diagram below. A string will be handed to one person and every that receives electricity will raise their hand.

l1l2l3l4l lolololol lolololol lolololol

Review a simple circuit with a light

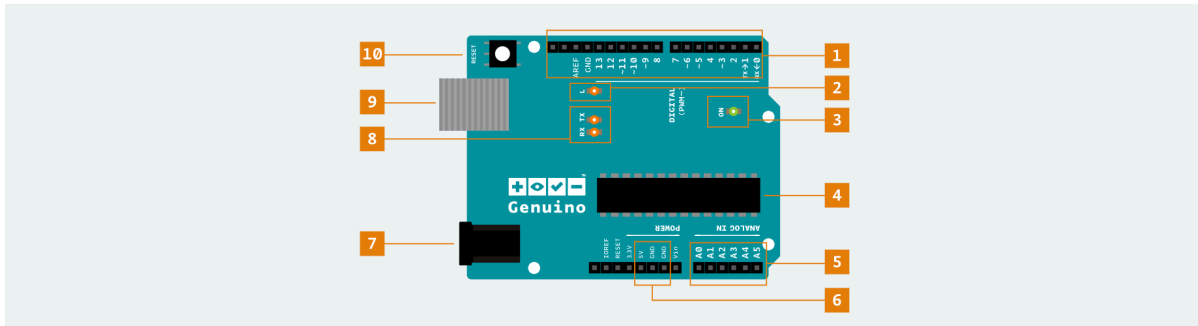


light

Ardiuno wiring

Number 1 below is where we connects the wires to the adriuno. **The pins can be thought of as the negative side of the battery and ground can be thought as the positive side of the battery.**

Arduino/Genuino Uno.



- 1. **Digital pins** Use these pins with `digitalRead()`, `digitalWrite()`, and `analogWrite()`. `analogWrite()` works only on the pins with the PWM symbol.
- 2. **Pin 13 LED** The only actuator built-in to your board. Besides being a handy target for your first blink sketch, this LED is very useful for debugging.
- 3. **Power LED** Indicates that your Genuino is receiving power. Useful for debugging.
- 4. **ATmega microcontroller** The heart of your board.
- 5. **Analog in** Use these pins with `analogRead()`.
- 6. **GND and 5V pins** Use these pins to provide +5V power and ground to your circuits.
- 7. **Power connector** This is how you power your Genuino when it's not plugged into a USB port for power. Can accept voltages between 7-12V.
- 8. **TX and RX LEDs** These LEDs indicate communication between your Genuino and your computer. Expect them to flicker rapidly during sketch upload as well as during serial communication. Useful for debugging.
- 9. **USB port** Used for powering your Genuino Uno, uploading your sketches to your Genuino, and for communicating with your Genuino sketch (via `Serial.println()` etc.).
- 10. **Reset button** Resets the ATmega microcontroller.

Activity 4: Wiring a led

Students will write the code to turn on the light on a piece of paper. When it is correct, an instructor or assistant will download the code to the arduino. One student will work on the arduino at a given time then pass it to the second student. Challenge the student to wire it differently by crossing the middle of the breadboard. Instructor and assistants give the students 10–20 mins before helping.

Correct Code

```
int x = 13;
```

```
void setup() {
```

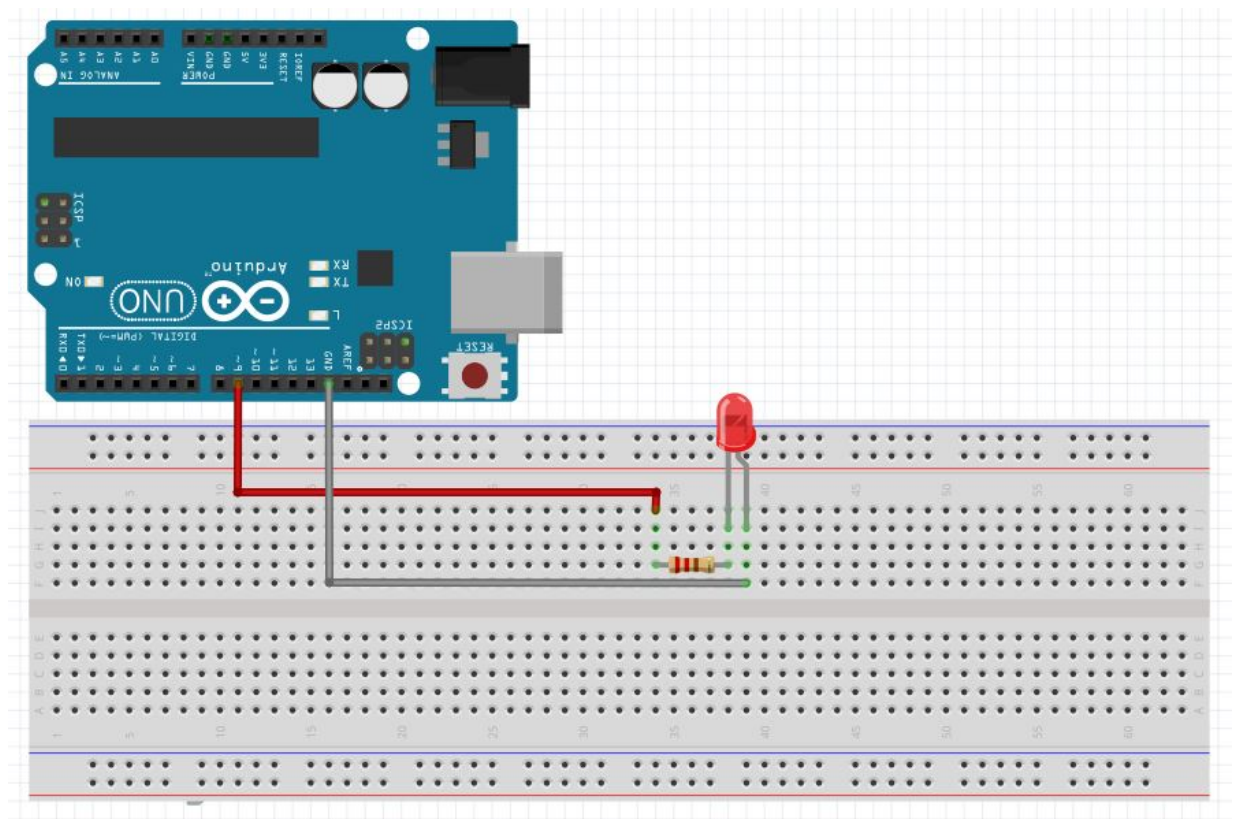
```
pinMode(x, OUTPUT);

}

void loop() {
digitalWrite(x, HIGH);

}
```

Wiring



Additional activities

1. Rewire by crossing the middle of the breadboard
2. Switching resistors and see what happens

Activity 5: Blinking Light

1. Give examples in the real world that blink/flash

- Car light's that's blinking when making a turn
- Walk light blinking
- Police lights blinking

2. **Students: Why is blinking important?**

- It acts as communication. For example a walk light blinking is telling you to hurry up. A charger blinking is telling you it's running out of battery.
- Everything around you is communicating something.

3. **Students: How does it work?**

- It turns on and off the switch in a circuit

4. **Demonstrate by turning off and on the light in the class room**

5. **Students: What does it being done between turning on and off**

- Time is passing

6. **Teach milliseconds. 1000 milliseconds are in a 1 second**

- How many millis in .5, 2, 2.75 seconds?

7. **Introduce the delay(1000); function**

Blinking Light Code

Students will be required to write the code below on a piece of paper

```
// the setup function runs once when you press reset or power the board
```

```
void setup() {
```

```
// initialize digital pin LED_BUILTIN as an output.
```

```
pinMode(8, OUTPUT);
```

```
}
```

```
// the loop function runs over and over again forever
```

```
void loop() {
```

```
digitalWrite(8, HIGH); // turn the LED on (HIGH is the voltage level)
```

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

```
digitalWrite(8, LOW); // turn the LED off by making the voltage LOW
```

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

```
}
```

If time permits students will code and wire a second led. One blinking and another on high.