

Arduino Starter Kit Tutorial

Developed by
M5

Department of Electrical and Computer Engineering
UMass Amherst
March 2010

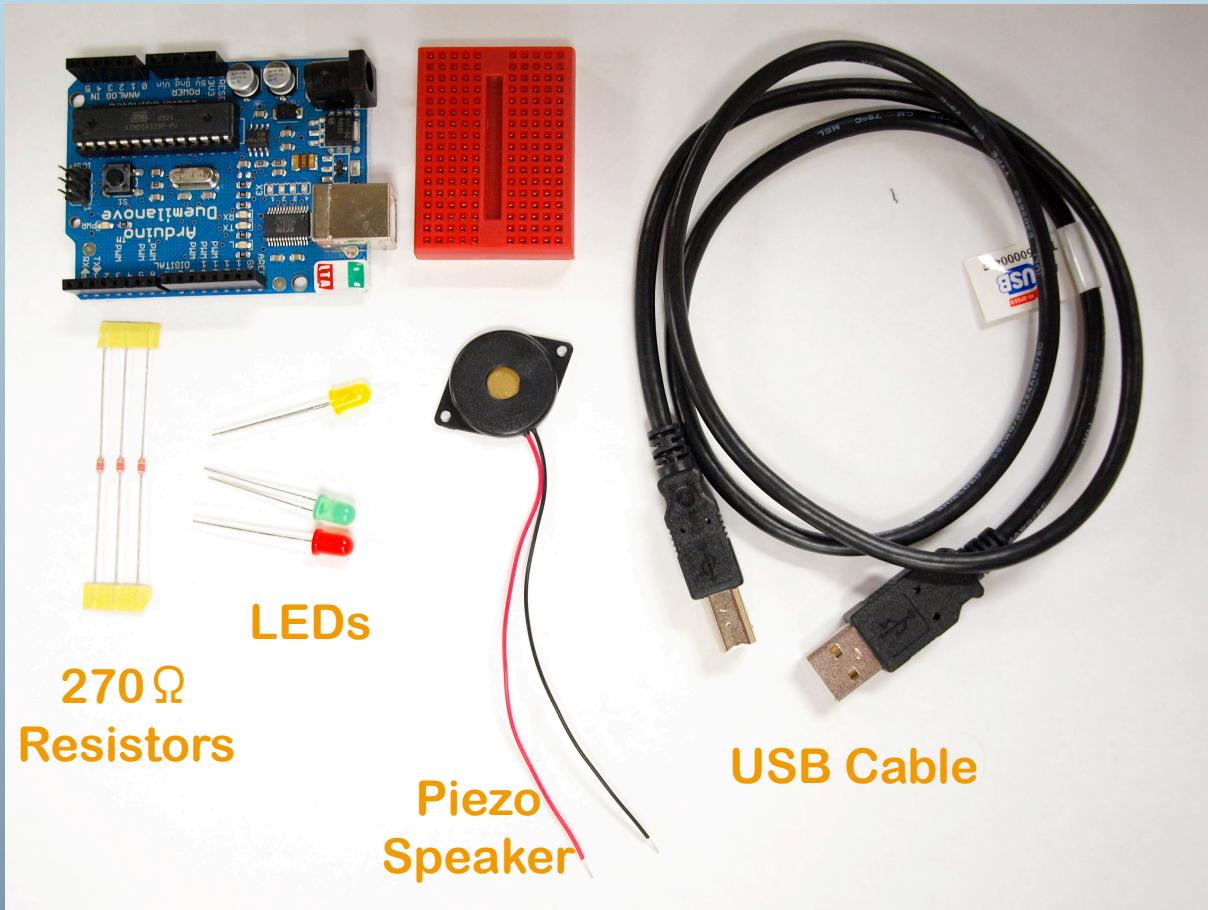
UMassAmherstM5.org

Arduino Kit

- 1x Arduino Duemilanove
- 1x USB Cable
- 1X Red Breadboard
- 1x Piezo Speaker
- 3x LEDs (Red, Yellow, Green)
- 3x 270Ω 1/4W Resistors

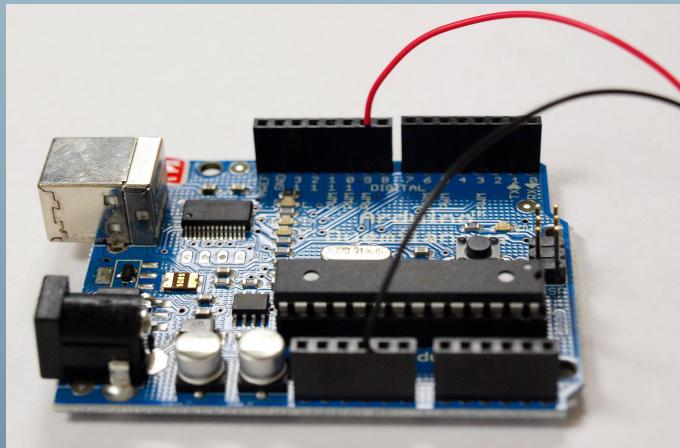
Arduino Kit

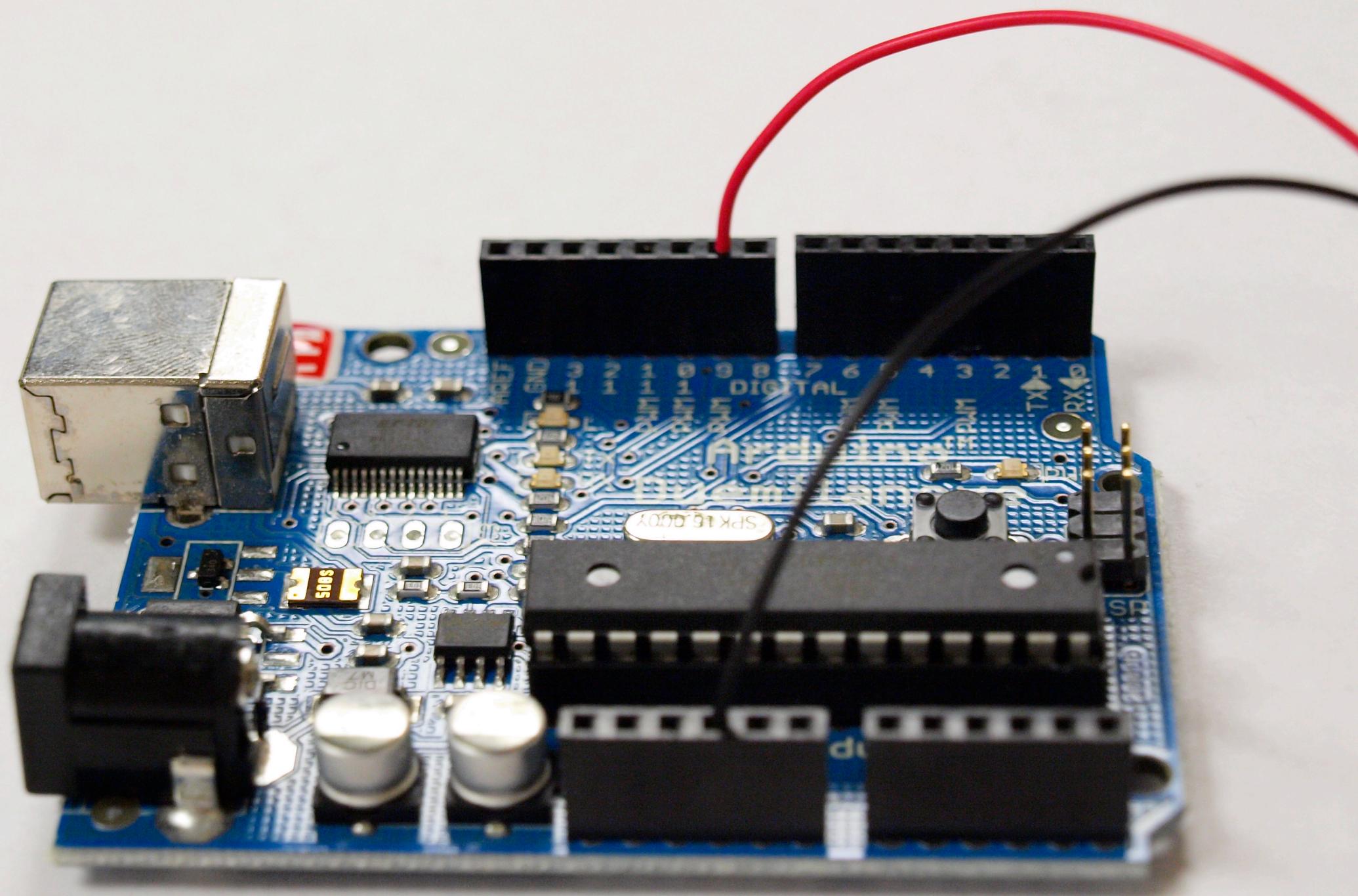
Arduino Breadboard



Getting Started

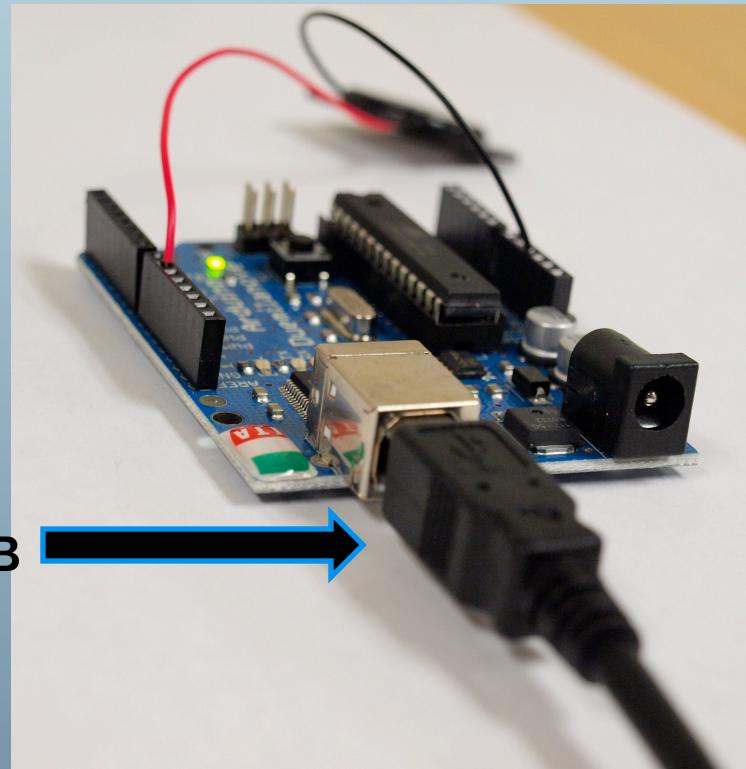
- First we are going to assemble the circuit.
 - Place the red piezo wire into pin 9 on the Arduino
 - Place the black piezo wire into the GND pin on the Arduino





Connecting to USB

- Next we will connect the USB cable to the Arduino
- You should now hear a lovely Lady Gaga tune



Lady Gaga Code

```
// TONES =====
// Start by defining the relationship between
//   note, period, & frequency.
#define c 3830 // 261 Hz
#define d 3400 // 294 Hz
#define e 3038 // 329 Hz
#define f 2864 // 349 Hz
#define g 2550 // 392 Hz
#define a 2272 // 440 Hz
#define b 2028 // 493 Hz
#define C 1912 // 523 Hz
// Define a special note, 'R', to represent a rest
#define R 0

// SETUP =====
// Set up speaker on a PWM pin (digital 9, 10 or 11)
int speakerOut = 9;
// Do we want debugging on serial out? 1 for yes, 0 for no
int DEBUG = 1;

void setup() {
  pinMode(speakerOut, OUTPUT);
  if (DEBUG) {
    Serial.begin(9600); // Set serial out if we want debugging
  }
}

// MELODY and TIMING =====
// melody[] is an array of notes, accompanied by beats[],
// which sets each note's relative length (higher #, longer note)
int melody[] = {c, d, e, c, f, e, f, e, d, c, d, e, e, f, e, d, c};
int beats[] = {8, 8, 8, 8, 24, 8, 8, 16, 36, 20, 16, 16, 8, 8, 8, 16, 36};
int MAX_COUNT = sizeof(melody) / 2; // Melody length, for looping.

// Set overall tempo
long tempo = 30000;
// Set length of pause between notes
int pause = 1000;
// Loop variable to increase Rest length
int rest_count = 100; //<-BLETCHEROUS HACK; See NOTES

// Initialize core variables
int tone = 0;
int beat = 0;
long duration = 0;
// PLAY TONE =====
// Pulse the speaker to play a tone for a particular duration
```

```
void playTone() {
  long elapsed_time = 0;
  if (tone > 0) { // if this isn't a Rest beat, while the tone has
    // played less long than 'duration', pulse speaker HIGH and LOW
    while (elapsed_time < duration) {

      digitalWrite(speakerOut,HIGH);
      delayMicroseconds(tone / 2);

      // DOWN
      digitalWrite(speakerOut, LOW);
      delayMicroseconds(tone / 2);

      // Keep track of how long we pulsed
      elapsed_time += (tone);
    }
  }
  else { // Rest beat; loop times delay
    for (int j = 0; j < rest_count; j++) { // See NOTE on rest_count
      delayMicroseconds(duration);
    }
  }
}

// LET THE WILD RUMPUS BEGIN =====
void loop() {
  // Set up a counter to pull from melody[] and beats[]
  for (int i=0; i<MAX_COUNT; i++) {
    tone = melody[i];
    beat = beats[i];

    duration = beat * tempo; // Set up timing

    playTone();
    // A pause between notes...
    delayMicroseconds(pause);

    if (DEBUG) { // If debugging, report loop, tone, beat, and duration
      Serial.print(i);
      Serial.print(":");
      Serial.print(beat);
      Serial.print(" ");
      Serial.print(tone);
      Serial.print(" ");
      Serial.println(duration);
    }
  }
}
```

Loading Code

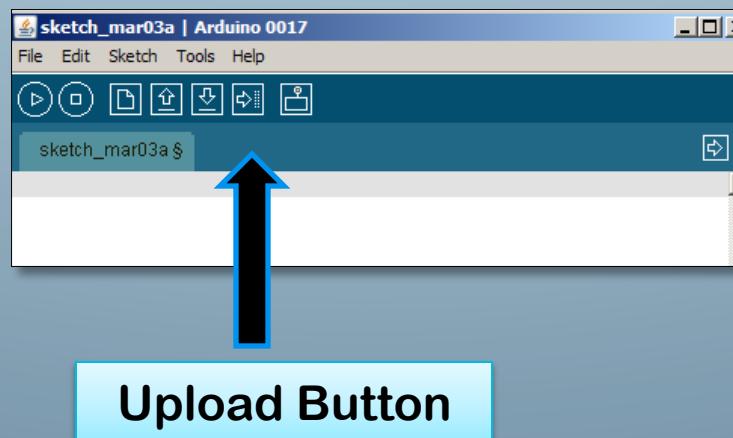
- Let's replace the Lady Gaga program with the original simple melody from the Arduino mother site
<http://www.arduino.cc/en/Tutorial/PlayMelody>
- Copy and Paste the melody code into the Arduino IDE (the formatting may look weird)
- This program is complicated but please don't worry. We will show you a simpler example next.

Arduino IDE



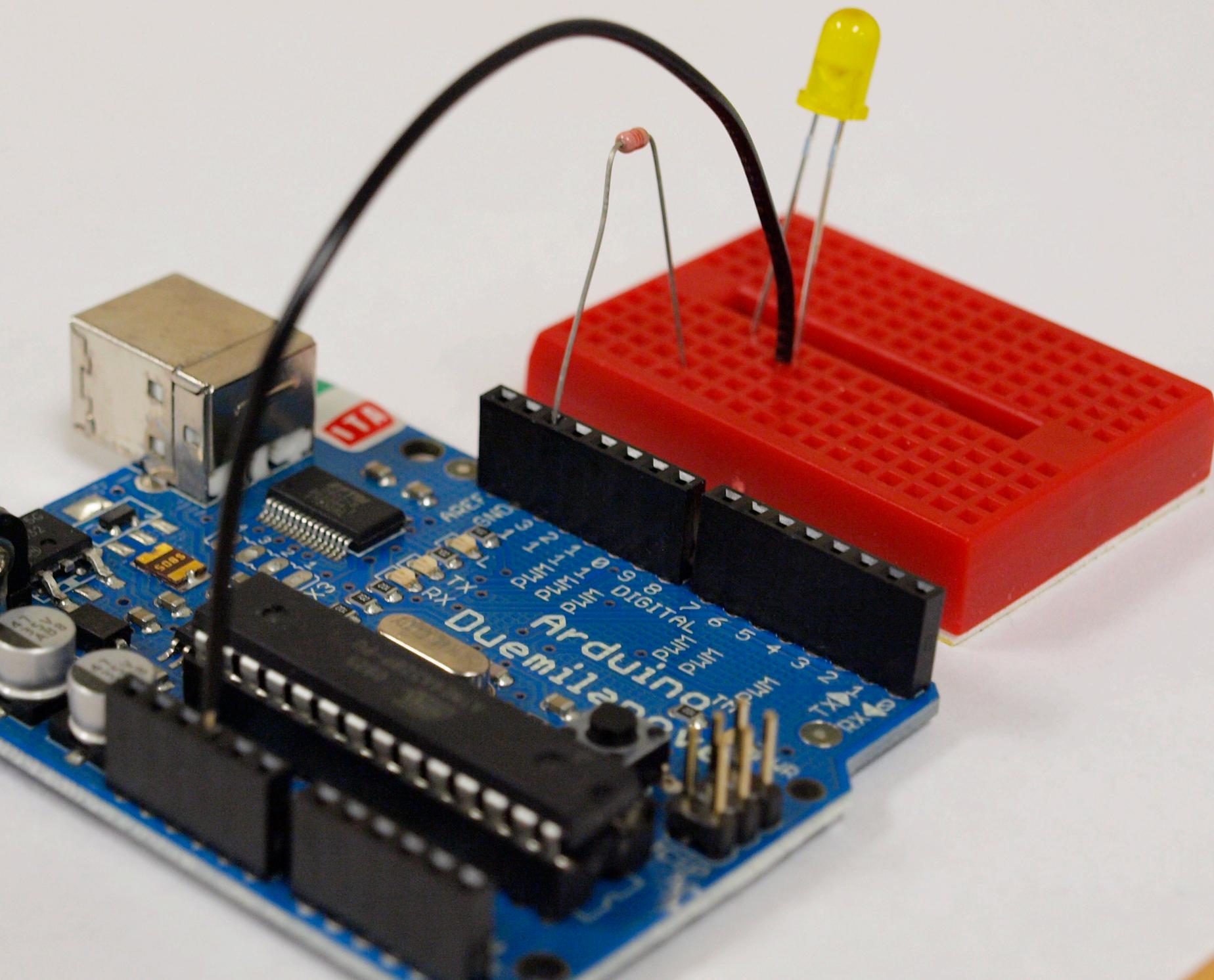
Loading Code

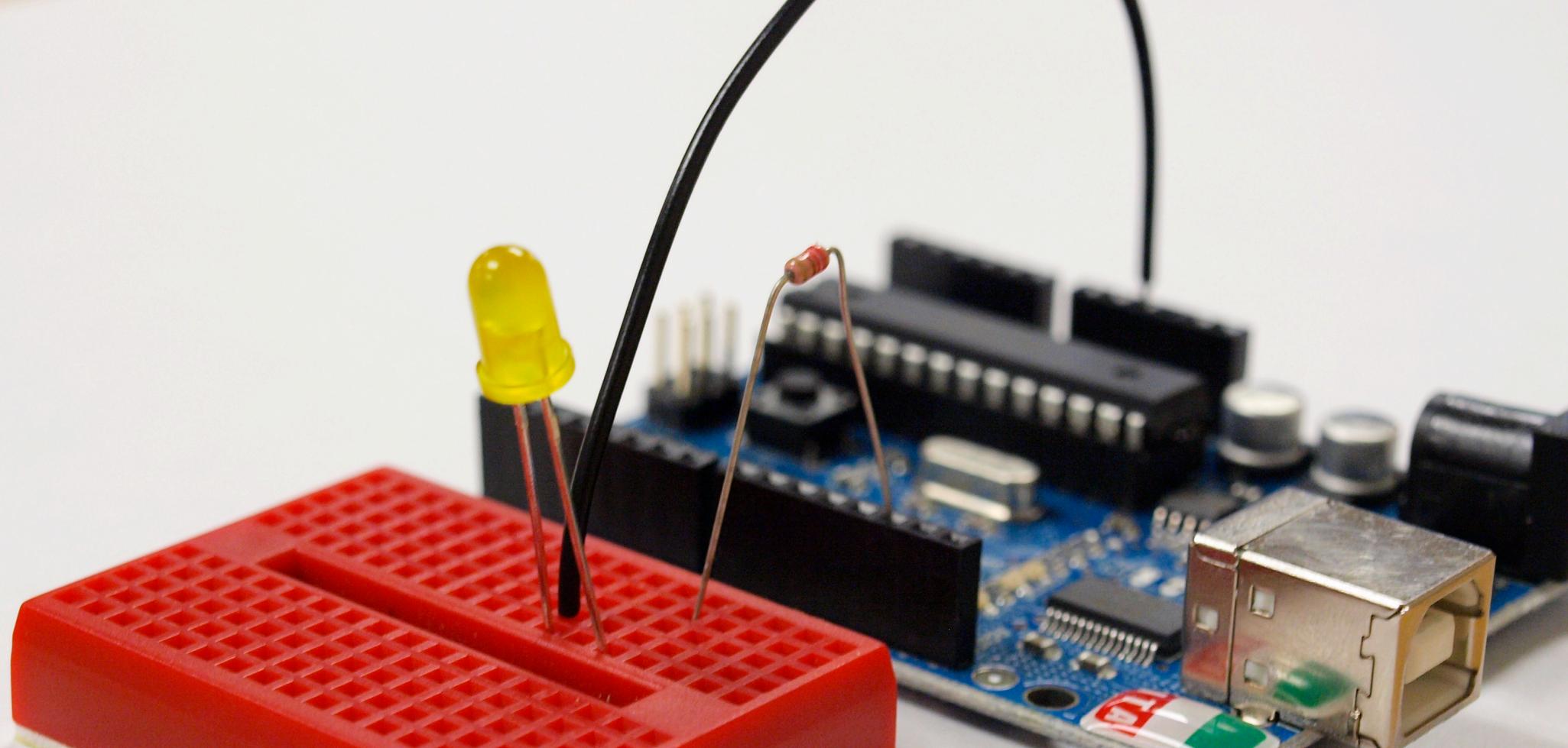
- You are now ready to upload the code
- Under Tools > Serial Port
 - Choose the COM (serial) port the Arduino is connected to
- Click the Upload button
- You should now hear a new melody



LED Blink

- Now we are going to upload a simpler computer program
- The circuit should be wired up before connecting the USB cable
- The circuit will be shown on two photographs and will then be followed by detailed instructions





LED Blink

- The circuit nodes (five adjacent holes) on the red breadboard are connected internally. There are 34 circuit nodes on the breadboard (17 on each half). All circuit elements (wire, LED, etc) plugged into any given node will be connected to each other via an internal metallic strip.

LED Blink

- Plug one side of a $270\ \Omega$ resistor into Arduino's pin 13. Plug the other side of the resistor into any hole on the red breadboard.

LED Blink

- Plug one side of an LED into the node where you plugged in the resistor
- Plug the other LED lead into a nearby node. Run a wire from this node to ground on the Arduino (GND)
- This completes the circuit

LED Blink Code

```
int ledPin = 13; // LED connected to digital pin 13

// The setup() method runs once, when the sketch starts

void setup() {
  // initialize the digital pin as an output:
  pinMode(ledPin, OUTPUT);
}

// the loop() method runs over and over again,
// as long as the Arduino has power

void loop()
{
  digitalWrite(ledPin, HIGH); // set the LED on
  delay(1000); // wait for a second
  digitalWrite(ledPin, LOW); // set the LED off
  delay(1000); // wait for a second
}
```

Loading LED Blink

- Now copy the LED Blink Code into the Arduino IDE
- Make sure the Arduino is connected to the PC via USB
- Upload the code to your board
- Enjoy the blinking LED :)
(If the LED is not blinking: remove it, rotate 180°, reinsert it)

Going Further...

- Can you figure out how to make the other two LEDs in your kit blink?
- Using the websites at the end of this presentation, modify the LED blink code
- Always have a resistor connected in series with an LED

Websites/References

- At M5 we use these websites constantly for references and inspiration:

www.arduino.cc

- Download Software (PC, Mac or Linux)
- Reference Page for commands

www.UMassAmherstM5.org

- Tutorials section

www.SparkFun.com

- A fantastic online shop in Colorado

Happy Making

- We hope you'll enjoy the Arduino. It's an amazing little powerhouse of a microcontroller board.
- Search for Arduino videos on the web. People are using them in many creative ways.
- Be Ambitious!
- We hope to see you in M5 soon!

JDLL & TBS