

Glow Your Own Coding Club – May Workshop Guide

During this workshop, we will work on our Arduino coded art projects, by making our own physical circuits, with a button used to activate a sequence of LEDs. In our next workshop, we will 'hack' this set-up, so we can make the LEDs turn on by using touch, rather than the button.

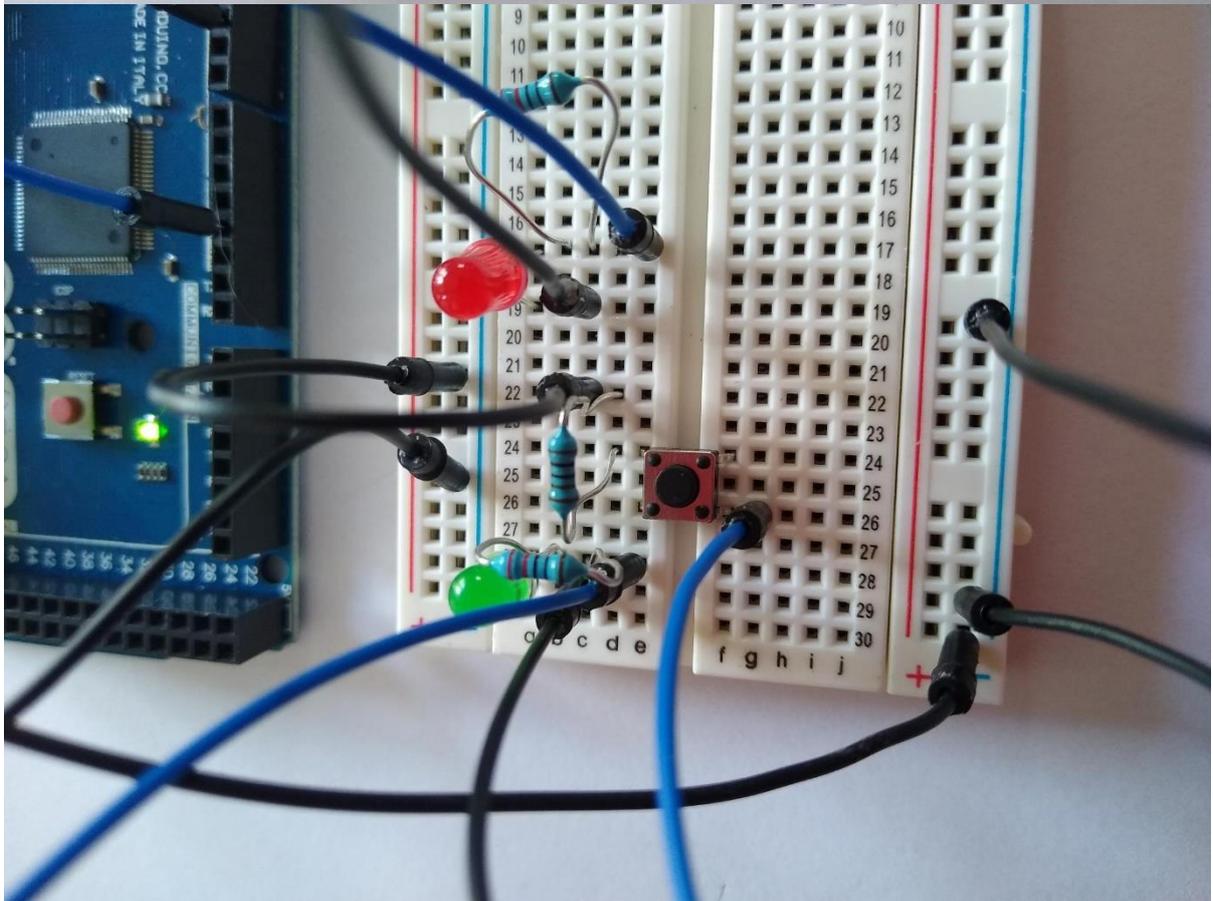
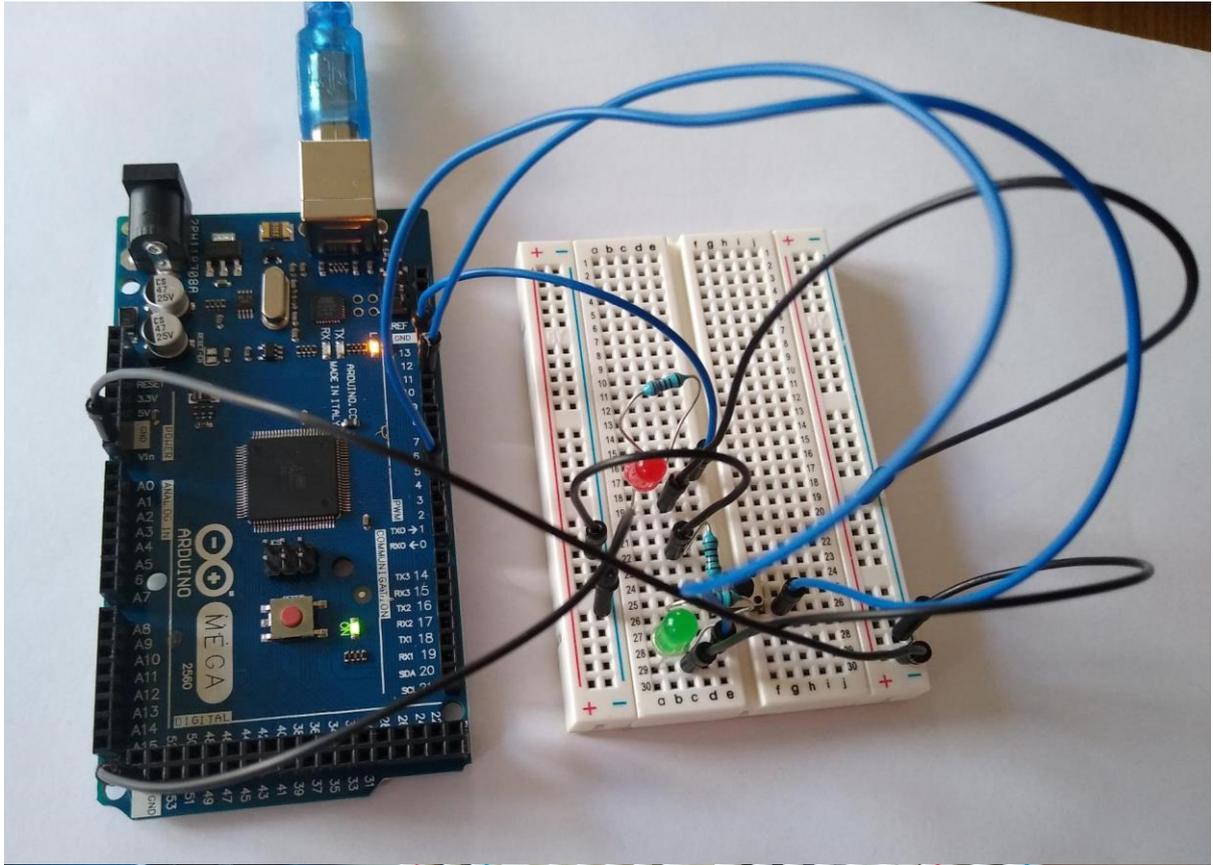
Here is a step-by-step guide of the session:

Items needed:

- Arduino + USB Cable (and UNO R3 software download)
- Breadboard
- LEDs (approx. 3-5)
- Jumper/Dupont wires (5x black wires for ground and 3x wires in a colour of your choice)
- Resistors (1x 1k Ω and 2x 220 Ω)
- A button

What to do:

- 1) Collect your equipment together.
- 2) Put your button into the breadboard holes E24, F24, E26, F26.
- 3) Put the 1k Ω resistor in D24 and D22.
- 4) Put 1x ground wire (black) into C22 and in the blue MINUS line of your breadboard nearest to the 'a' column and close to row 22.
- 5) Put 1x ground wire (black) in the blue MINUS line nearest to the 'a' column, close to row 24, and the other end in one of the DIGITAL Ground (GND) pins in your Arduino.
- 6) Put a coloured wire (of your choice) into G26 on the breadboard, and then pin 3 of your Arduino.
- 7) Put an LED (any colour) and place the longer end in A28 and the shorter end in A29.
- 8) Get a 220 Ω resistor and it place in B28 and C28.
- 9) Put a coloured wire (of your choice) in E28 and pin 10 on Arduino.
- 10) Put 1x ground wire (black) in C29 and then in the blue MINUS line nearest to the 'j' column of the breadboard and close to row 29.
- 11) Put 1x ground wire (black) into the blue MINUS line nearest to the 'j' column, close to row 30, and into a GND pin in your Arduino.
- 12) Place one more LED into your breadboard, putting the long end into A17 and the short end into A19.
- 13) Put 220 Ω resistor into B17 and C17.
- 14) Put a coloured wire (of your choice) into E17 and pin 11 on Arduino.
- 15) Put 1x ground wire (black) into C19 and the blue MINUS line nearest to the 'j' column of the breadboard, close to row 19.
- 16) Check that your circuit looks like the images on the following page:
(NB. the coloured wires of choice are blue)



- 17) Copy and paste the below code into your UNO R3 Software and export it to your Arduino (as per previous GYO pdf guides).
[Make sure the lines of code are displayed as below, by either copying and pasting one line at a time or copying the whole code and adjusting the line spacing using your Enter key whilst in the UNO R3 Software]:

```
// constants won't change. They're used here to set pin numbers:
const int buttonPin = 3; // the number of the pushbutton pin
const int ledPin1 = 10; // the number of the LED pin
const int ledPin2 = 11; // the number of the LED pin

// variables will change:
int buttonState = 0; // variable for reading the pushbutton status

void setup() {
  Serial.begin(9600);
  // initialize the LED pin as an output:
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT_PULLUP);
}

void loop() {
  // read the state of the pushbutton value:
  buttonState = digitalRead(buttonPin);

  // check if the pushbutton is pressed. If it is, the buttonState is LOW:
  if (buttonState == HIGH) {
    // turn LED on:
    digitalWrite(ledPin1, LOW);
    delay(300);
    digitalWrite(ledPin2, LOW);
    delay(300);
  } else {
    // turn LED off:
    digitalWrite(ledPin1, HIGH);
    delay(300);
    digitalWrite(ledPin2, HIGH);
    delay(300);
  }
  Serial.println(buttonState);
}
```

This code can also be copied and pasted in the correct format, via this GitHub link:
<https://raw.githubusercontent.com/lsioviel/glow-your-own/main/example.txt>

Common troubleshooting

- Check that all components are pushed into pins fully (resistors can be a little tricky to push in and they bend easily, so we recommend checking these first!)
- To better understand which rows and columns of the breadboard rows are connected when forming your circuit, please view the image at this link:
<https://cdn.sparkfun.com/assets/3/d/f/a/9/518c0b34ce395fea62000002.jpg>

When uploading code to Arduino:

- Check that the software is recognising the correct Port (*go to 'Tools' in the top toolbar, then select 'Port'. You need to select the COM Port that has your Arduino listed*).
- Check that the software is registering your Arduino as the correct model (*go to 'Tools' in the top toolbar, then select 'Board'. Make sure the model you select is the same as is written on your Arduino*).