

TECHNOVUS

**Build-It-Yourself Shops
Document - Module 5:**

Intended Use & Scope

This document is developed by Technovus as an aid for members attending the workshops. The scope and context of the topics covered here is limited to embedded programming (C/C++), especially programming Arduino microcontrollers.

The Build Activity series continues from where the beginner workshops end. The participants are assumed to have been introduced to programming, circuit construction and have a basic understanding of circuit design.

The end goal of the series is to provide more experience in embedded programming and promote confidence in the subject.

Contents

Module 5: More on buttons	1
Build!	1
Circuit	1
Code	2
Answers	4

Module 5: More on buttons

This module shows a way the push button can be integrated into the circuit to make more functional and user friendly builds. The exercise in this module will use a push button to control the lighting up of two LEDs connected in a circuit. The button will be used to start the program and used to switch which LED is turned on.

We have worked with buttons and LEDs since module 2 of the workshop so you should be comfortable with wiring these components.

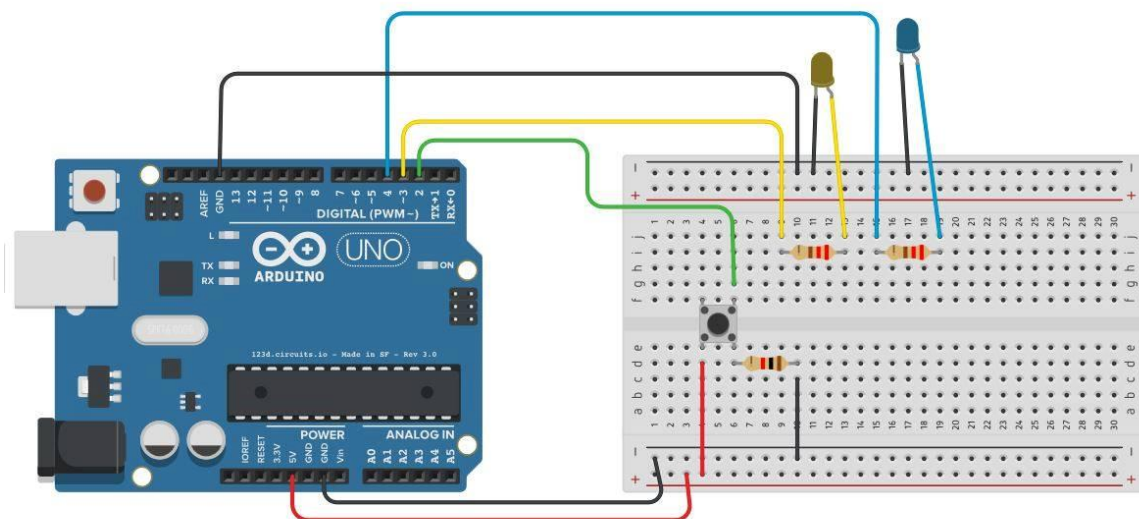
Build!

This building exercise involves making an LED light up when the push button is pressed and then switch which LED is lit with every press of the push button.

Items required:

1. Microcontroller
2. Pushbuttons 1x
3. Resistor – 2x 220 Ω
4. Resistor – 1x 1k Ω
5. LED – 2x

Circuit



Code

```
// variables that never change
const int buttonPin = 2; // button pin
const int ledPin1 = 3; // first led
const int ledPin2 = 4; // second led

// variables that will change
int buttonState = 0; // variable for reading pushbutton status
int track = 0; // variable for keeping track of on/off

// the setup routine runs once when you press reset:
void setup() {
  Serial.begin(9600);
  // initialize the push button pin as an input
  pinMode(buttonPin, INPUT);
  // initialize the LED pins as outputs
  pinMode(ledPin1, OUTPUT);
  pinMode(ledPin2, OUTPUT);
  attachInterrupt(digitalPinToInterrupt(buttonPin), toggle, Fill-in(1))

  // Begin program by pressing button
  Serial.println("\n Press button to start");
  while (buttonState != HIGH){
    buttonState = digitalRead(buttonPin);
  }
  Serial.println("Starting");
  delay(2000);
}

// the loop routine runs over and over again forever:
void loop() {
  // check the state of the button
  if (track%2 == Fill-in(2)){
    digitalWrite(ledPin1, LOW);
    digitalWrite(ledPin2, HIGH);
  }
  else if (track%2 == Fill-in(3)) {
    digitalWrite(ledPin1, HIGH);
    digitalWrite(ledPin2, LOW);
  }
}
```

```
//Toggle function  
void toggle() {  
    Fill-in(4);  
}
```

As this module serves to build your software programming skills, omissions have been made in the code. You are to understand the code and complete it to achieve the outcome required. The answers can be found at the end of the module but make an attempt to get them first.

Answers

1. `RISING` 2. `0` 3. `1` 4. `track++`