# Debounce on a Pushbutton

Read a pushbutton, filtering noise.

Last revision • 02.09.2022

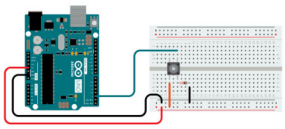


Pushbuttons often generate spurious open/close transitions when pressed, due to mechanical and physical issues: these transitions may be read as multiple presses in a very short time fooling the program. This example demonstrates how to **debounce** an input, which means checking twice in a short period of time to make sure the pushbutton is definitely pressed. Without debouncing, pressing the button once may cause unpredictable results. This sketch uses the `millis()` function to keep track of the time passed since the button was pressed.

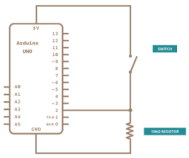
## Hardware Required

- Arduino Board
- momentary button or switch
- 10k ohm resistor
- hook-up wires
- breadboard

## Circuit



## Schematic



## Code

The sketch below is based on **Limor Fried's version of debounce**, but the logic is inverted from her example. In her example, the switch returns LOW when closed, and HIGH when open. Here, the switch returns HIGH when pressed and LOW when not pressed.

```
      ledState;
77          }
78        }
79      }
80
81      // set
        the LED:
82
        digitalWrite
        (ledPin,
        ledState);
83
84      // save
        the
        reading.
        Next time
        through the
        loop, it'll
        be the
        lastButtonSt
        ate:
85
        lastButtonSt
        ate =
        reading;
86    }
```

DOCS

## Learn more

You can find more basic tutorials in the **built-in examples** section.

You can also explore the **language reference**, a detailed collection of the Arduino programming language.

*Last revision 2015/07/29 by SM*

### Suggest changes

The content on docs.arduino.cc is facilitated through a public GitHub repository. If you see anything wrong, you can edit this page here.

### Need support?

Help Center

Ask the Arduino Forum

Discover Arduino Discord

### License