



Search on Docs



[← Go Back](#)

## Built-in Examples

Basics



Digital



Blink Without Delay

How to Wire and Program a Button

Debounce on a Pushbutton

InputPullupSerial

State Change Detection (Edge Detection) for pushbuttons

Simple keyboard using the tone() function

Play a Melody using the tone() function

Tone on Multiple Speakers

Pitch follower using the tone() function

Analog



Communication



Control Structures



Sensors



Display



Strings



USB



Arduino ISP



Home / Programming / Built-in Examples / **Debounce on a Pushbutton**

# 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

ON THIS PAGE

### Hardware Required

Circuit

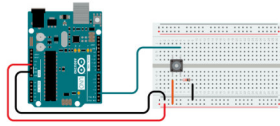
Schematic

Code

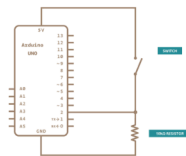
Learn more

Help

## 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.

```
77         ledState,  
78     }  
79     }  
80  
81     // set  
82     the LED:  
83  
84     digitalWrite  
85     (ledPin,  
86     ledState);  
87  
88     // save  
89     the  
90     reading.  
91     Next time  
92     through the  
93     loop, it'll  
94     be the  
95     lastButtonSt  
96     ate:  
97  
98     lastButtonSt  
99     ate =  
100    reading;  
101    }  
102    }  
103    }  
104    }  
105    }  
106    }  
107    }
```



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	Need support?	License
The content on <a href="#">docs.arduino.cc</a> is facilitated through a public <a href="#">GitHub repository</a> . If you see anything wrong, you can edit this page <a href="#">here</a> .	<a href="#">Help Center</a> <a href="#">Ask the Arduino Forum</a> <a href="#">Discover Arduino</a> <a href="#">Discord</a>	The Arduir document on is licensed under the <a href="#">Creative Commons Attribution Share Alike 4.0</a> license