

Control an LED Using a Button

Jul 2026



Introduction

In this example, we are going to turn on and off an LED on our Raspberry Pi Pico using a button. The button is a PTT (Press to Talk) style button. In this example the LED will flash slowly while the button is not pressed and quickly while the button is pressed.

Add an LED and Button to the Pico

The image below shows what we are trying to achieve.



In this example, we have the LED being connected to GP14 (pin 19) and GND (pin 18) via a 100 ohm resistor. You can use one of the GND pins on the Pico, or a common GND rail if you have made one.

Remember, the Anode of the LED (the longer leg) goes to GP14 and Cathode goes to GND.

The button is wired between pins GP16 (pin 21) and GND (pin 23)

Micro Python Code

Here is the code that we will use:

[| download](#)

```
# Hardware Test
# Blink and LED (GP14) slowly/quickly while a button (GP16) is not-pressed/pressed

from machine import Pin          # Import Pin class to control GPIO pins
from time import sleep_ms       # Import millisecond sleep function

led = Pin(14, Pin.OUT)          # Create an output pin on GPIO14 for the LED
button = Pin(16, Pin.IN, Pin.PULL_UP) # Create an input pin on GPIO16 with internal pull-up resistor

while True:                     # Infinite loop
    if button.value() == 0:      # If button is pressed (input pulled LOW)
        delay = 100             # Use short delay (fast blink)
    else:                       # Otherwise (button not pressed)
        delay = 1000            # Use long delay (slow blink)

    led.toggle()                # Flip LED state: ON → OFF or OFF → ON
    sleep_ms(delay)             # Wait for the chosen delay before next toggle
```

Because the button uses `Pin.PULL_UP`, its normal (unpressed) state is HIGH (1).

Pressing the button pulls the pin to LOW (0), which is why the code checks `button.value() == 0`.

[button-led-pico-004.mp4](#)

Looking at the simulation above, we can see that when the button is not pressed, we get a delay of 1000mS as per the MicroPython code, so 1 flash every two seconds (1s on, 1s off).

When the button is pressed, we get a delay of only 100mS, so 5 flashes every second (100mS on, 100mS off x 5).

If you have connected the LED and Button to your Pico correctly, this is what you should observe when you run the code on your setup.

From:

<https://wiki.alanwalker.uk/> - WalkerWiki - wiki.alanwalker.uk

Permanent link:

https://wiki.alanwalker.uk/doku.php?id=control_an_led_using_a_button&rev=1783620624

Last update: **2026/07/09 18:10**

