

Using LED Flash Timeout

Jul 2026



Introduction

***** Warning ***** You can lock yourself out of your Pi Pico if you don't have a method to break the code. Please think about if you want/need to do this!!!

When we are using Thonny, we are loading our code in to an app and clicking a play button, when we want to stop the code (break out) we simply press the stop button in our Thonny App.

However, if the code runs as boot, you don't necessarily have this option, so we need to build in a fail safe. There are two that we can use (there are more I am sure) and they are.

- 1. Build in a delay at the start, where you can connect to Thonny and click the stop button.
- 2. Add a button to a GPIO pin, and when that button is pressed, stop the code.

We are going to look at the first option in this example. Please not, pressing the BOOTSEL button will not stop the code.

Please note, this behaviour changes depending on the age of your Pico, the library files you are using and the version of Thonny. With the latest versions of code, it seems that this is less of an issue. It is still a good option to have some built in safeguards though.

What we are going to do is to flash the onboard LED for 20 seconds before starting the main code. This gives you 20 seconds to connect the Pico to your computer, run Thonny and hit the stop button.

We have to use two different sets of code unfortunately, as the onboard LED on the Pico and Pico W/W2 use a different connection method.

- On the original Raspberry Pi Pico and Pico 2, the onboard LED is wired directly to GPIO25 on the RP2040.
- On the Pico W and Pico 2W the LED is controlled through the Wi-Fi chip, using a pin called WL_GPIO0.

So we will need a slightly different set of code to flash the LED.

Raspberry Pi Pico 1 / Pico 2

From:

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

Permanent link:

https://wiki.alanwalker.uk/doku.php?id=using_led_flash&rev=1784131979

Last update: 2026/07/15 16:12

