

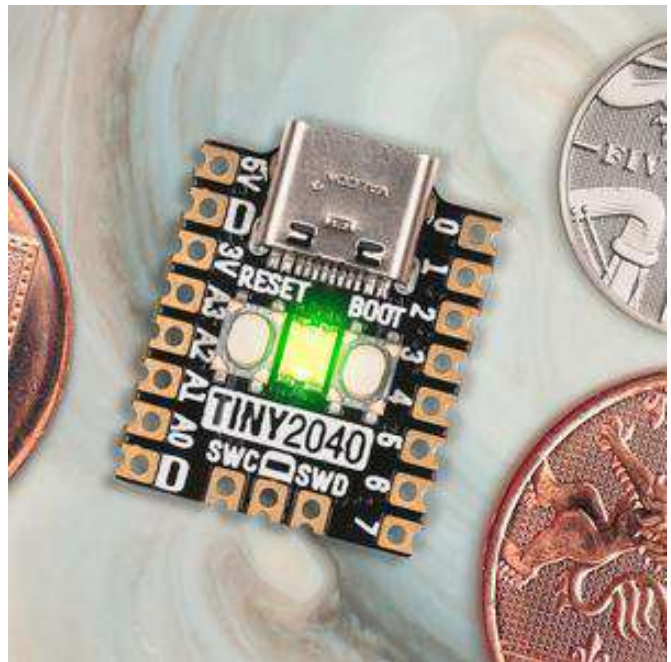


Purveyors of fine Maker goods, worldwide

# Tiny 2040

PIM558

A postage stamp sized RP2040 development board with a USB-C connection and 8MB of flash, perfect for portable projects, wearables, and embedding into devices.



While we love the Raspberry Pi Pico we also wanted something smaller and with a bunch more flash on board. Introducing the Tiny 2040 - a teeny tiny powerhouse with the chops to realise truly ambitious projects.

Powered and programmable via USB-C, Tiny 2040 comes with 8MB of QSPI (XiP) flash on board so it can handle projects small and large with ease. The board is designed with castellated pads to allow it to be directly soldered onto a PCB or alternatively you can use pin headers to hook it up on a breadboard or directly with wires. We've even managed to fit in a programmable RGB LED, a reset button and some clever circuitry that lets you use the boot button as a user controllable switch.

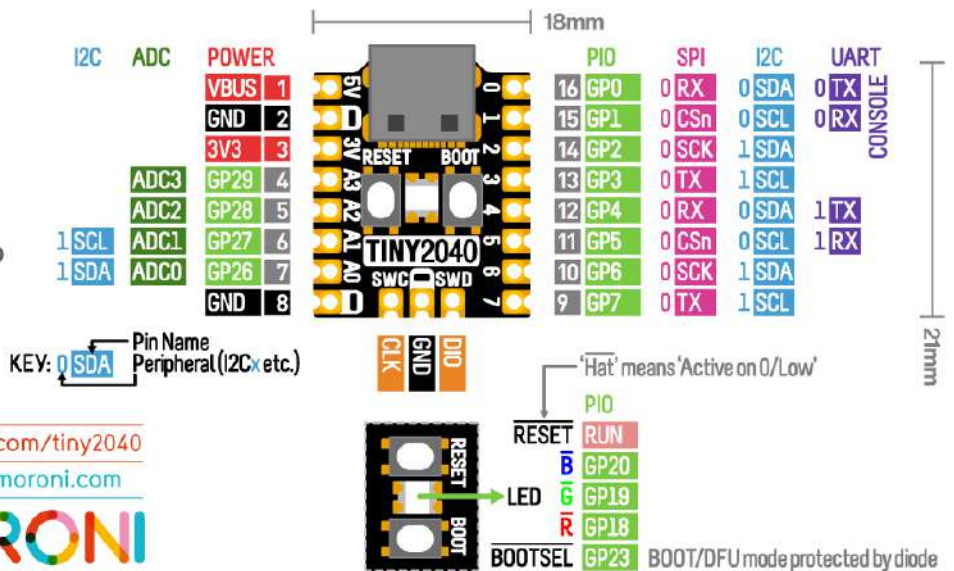
It's compatible with firmware built for the Raspberry Pi Pico but offers a reduced number of pins due to its size. You can even run MicroPython on it!

# Features

- Powered by RP2040
- ARM Cortex M0+ running at up to 133Mhz
- 264kB of SRAM
- USB-C connector for power, programming, and data transfer
- 8MB of QSPI flash supporting XiP
- User controllable RGB LED
- Twelve IO pins (including four 12-bit ADC channels)
- Switch for basic input (doubles up as DFU select on boot)
- On-board 3V3 regulator (max regulator current output 300mA)
- Input voltage range 3V - 5.5V
- Dimensions: approx 22.9 x 18.2 x 6mm (L x W x H, including the USB-C port)
- Eagle CAD part

## Pimoroni Tiny 2040

18x21.3mm  
 8MByte Flash  
 USB-C  
 RGB LED  
 Reset Button  
 User/Boot Button  
 16 pins+debug  
 GPO-7 ideal for PIO  
 Input: 3V-5.5V



<https://pimoroni.com/tiny2040>  
<https://forums.pimoroni.com>  
**PIMORONI**

## Getting Started

Tiny 2040 is firmware agnostic! You can program it with C/C++ or MicroPython in the same way as you would a Raspberry Pi Pico, though you'll need to bear in mind that it has a reduced number of pins. You can find (lots) more information on how to do that (as well as download links for the firmware/SDK) on the [RP2040 landing page](#).

You can also use CircuitPython on your Tiny 2040! CircuitPython is an easy to use, well-established ecosystem with lots of example code and drivers for interfacing with different kinds of hardware. [Click here](#) to download the CircuitPython firmware for Tiny 2040 and [click here](#) for a getting started guide.

## Notes

- The RGB LED is connected to GP18-GP20 and **active low** (so the on/off state will work in the opposite way to the LED on a Raspberry Pi Pico). You can PWM the pins to dim the LED - check out [Tonygo2's MicroPython example](#).

## About RP2040

Raspberry Pi's RP2040 microcontroller is a dual core ARM Cortex M0+ running at up to 133Mhz. It bundles in 264kB of SRAM, 30 multifunction GPIO pins (including a four channel 12-bit ADC), a heap of standard peripherals (I2C, SPI, UART, PWM, clocks, etc), and USB support.

One very exciting feature of RP2040 is the programmable IOs which allow you to execute custom programs that can manipulate GPIO pins and transfer data between peripherals - they can offload tasks that require high data transfer rates or precise timing that traditionally would have required a lot of heavy lifting from the CPU.

