

MicroPython auf dem Raspberry Pi Pico

Soft- und Firmware

Fürs Flashen und die Programmierung des ESP32/Raspberry Pi Pico:

[Thonny](#) IDE

[PuTTY](#) ein Terminalprogramm

Verwendete Firmware für den Raspberry Pi Pico W:

[RPI PICO W-20240105-v1.22.1.uf2](#)

Verwendete Firmware für den Raspberry Pi Pico:

[RPI PICO-20240105-v1.22.1.uf2](#)

Raspberry Pi Pico für MicroPython fit machen

Alles, was mit dem Raspberry Pi Pico gemacht werden soll, braucht erst einmal ein Zuhause. Mein Pico wird mit Gefolge im Verzeichnis F:\P_programmieren\az-blog__RASPI_PICO wohnen. In dem Verzeichnis lege ich folgende Unterverzeichnisse an:

_workspace für die MicroPython-Dateien

firmware zum Speichern der uf2-Firmwaredatei(en)

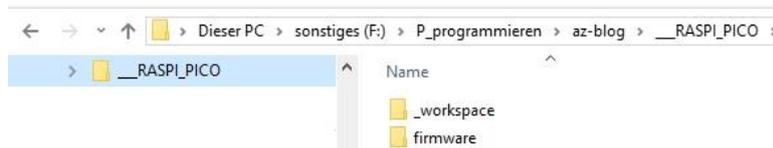


Abbildung 1: Arbeitsverzeichnis anlegen

Die Raspberry Pi-Community ändert von Zeit zu Zeit den Weg zur Download-Site. Das Folgende beschreibt die Version vom Januar 2024.

Um auf dem Raspberry Pi Pico mit MicroPython arbeiten zu können, müssen wir als Erstes die Firmware auf den Controller bringen. Der Raspi hilft uns dabei. Das USB-Kabel wird am PC angesteckt, am Raspi aber noch nicht. Neben dem USB-Port liegt die Taste BOOTSEL. Die drücken und halten wir, stecken dann das USB-Kabel am Raspi an und zählen langsam bis drei, das heißt wir halten die Taste für mindestens drei Sekunden weiterhin gedrückt.



Abbildung 2: Platine des Raspberry Pi pico_BOOT-Select-Taste

In dieser Zeit meldet sich ein neues USB-Laufwerk, bei mir ist es RPI-RP2 (H:). Darauf befinden sich zwei Dateien, die INDEX.HTM ziehen wir auf einen Browser.

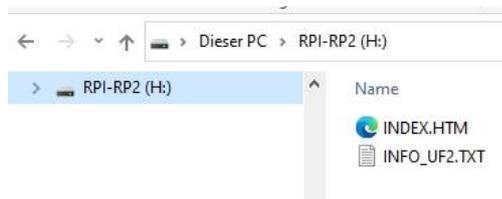


Abbildung 3: PICO ist gestartet

Die Raspberry Pi-Dokumentation startet mit dem Ordner **Microcontrollers**. Klick auf **Raspberry Pi Pico and Pico W**.

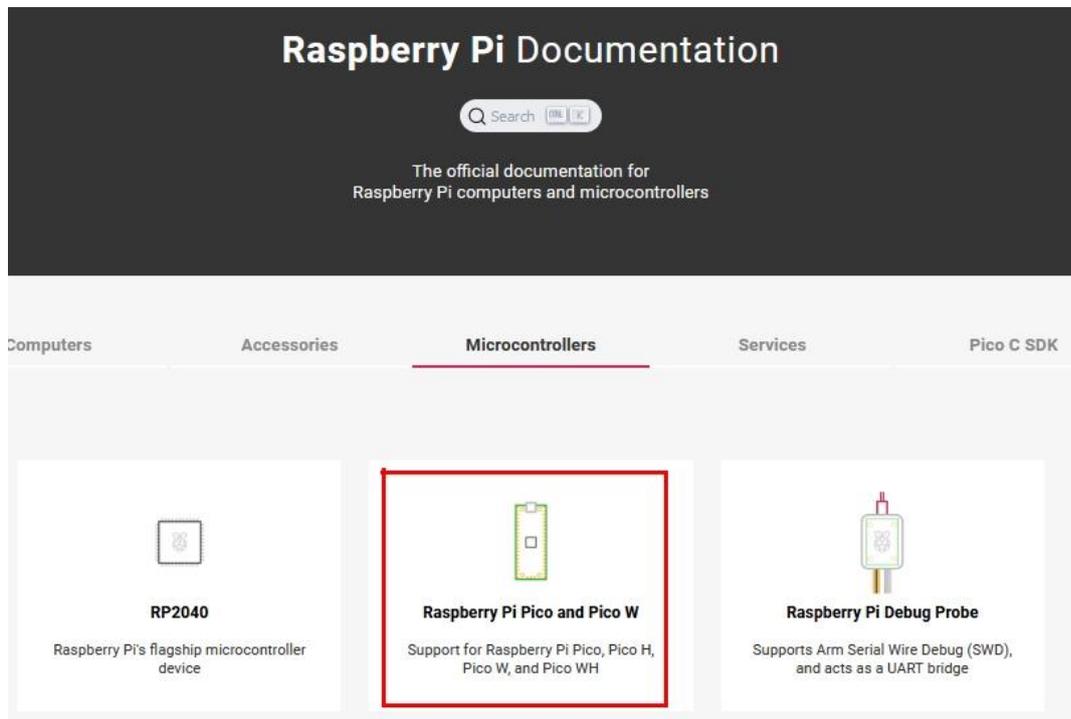


Abbildung 4: Zum Firmware-Download navigieren

Ganz unten in der Navigationsspalte klicken wir auf **MicroPython**.

Microcontrollers

RP2040
Raspberry Pi Pico and Pico W

The family

Raspberry Pi Pico and Pico H
Pinout and design files

Raspberry Pi Pico W and Pico WH
Pinout and design files

Documentation

RP2040 Device

Raspberry Pi Pico

Raspberry Pi Pico W

Software Development

Software Utilities

What is on your Pico?

Debugging using another Raspberry Pi Pico

Resetting Flash memory

Raspberry Pi Debug Probe

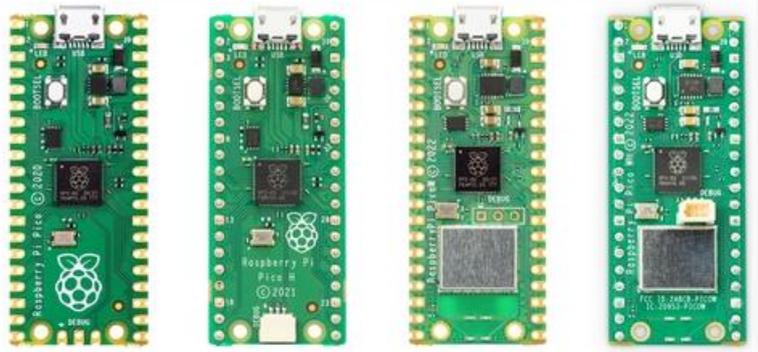
RP1

MicroPython

Raspberry Pi Pico and Pico W

The family

[Edit this on GitHub](#)



The Raspberry Pi Pico family currently consists of four boards; Raspberry Pi Pico (far left), Pico H (middle) and Pico WH (far right).

Abbildung 5: Hier geht's zu MicroPython

Im nun geöffneten Menü geht's mit **Drag and Drop MicroPython** weiter.

Computers **Accessories** **Microcontrollers** **Services**

Microcontrollers

RP2040
Raspberry Pi Pico and Pico W
Raspberry Pi Debug Probe
RP1
MicroPython

What is MicroPython?

Drag-and-Drop MicroPython

Where can I find documentation?

Further reading

Which hardware am I running on?

The C/C++ SDK

Raspberry Pi Pico and Pico W

The family

[Edit this on GitHub](#)

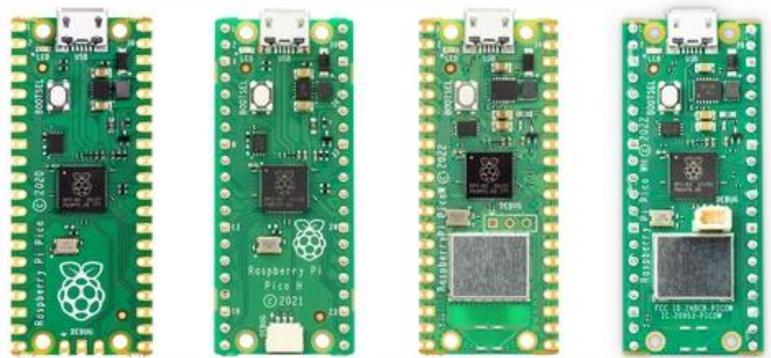


Abbildung 6: Hier werden wir fündig

Je nachdem welches Exemplar von Controller wir haben, laden wir die entsprechende uf2-Datei herunter – Klick.

Pico über [REPL](#) zu verbinden. Wie beim ESP32 oder ESP8266 öffnen wir das Options-Menü über **Run – Select interpreter**, um die Verbindungsparameter festzulegen.

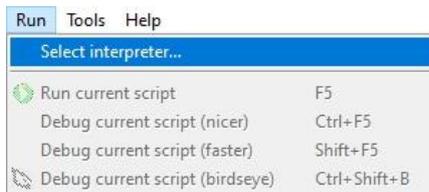


Abbildung 10: Optionsmenü starten

Wir markieren den Raspberry Pi Pico-Eintrag.

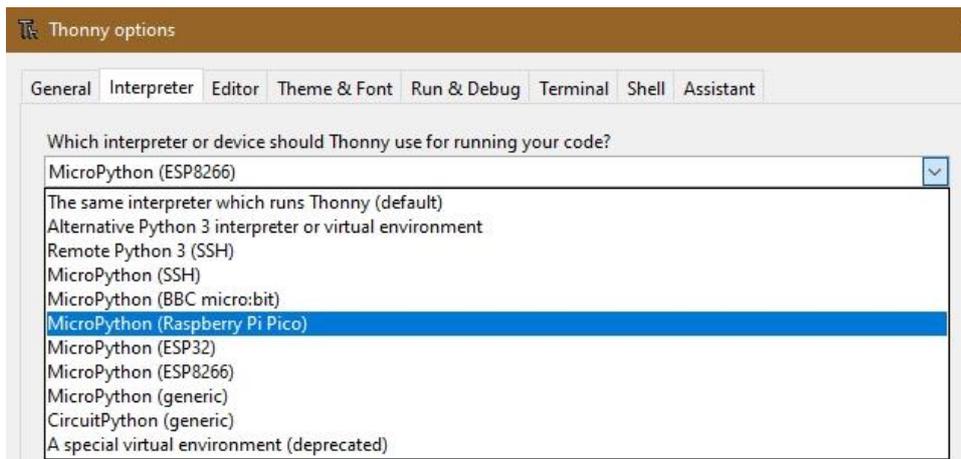


Abbildung 11: Interpreter auswählen

Nach der Auswahl der Schnittstelle bestätigen wir mit **OK**.

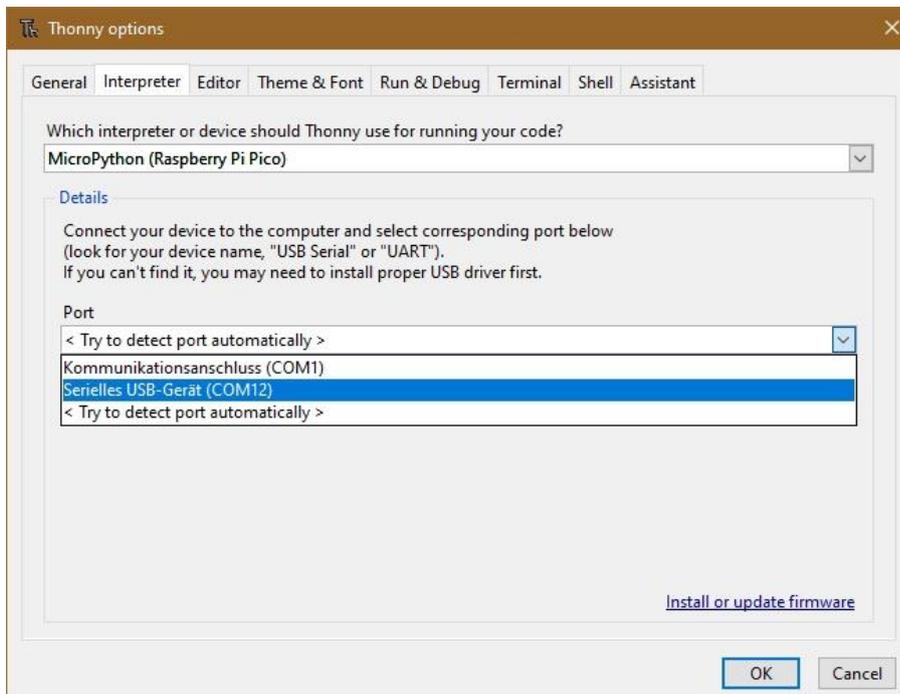


Abbildung 12: Schnittstelle auswählen

In REPL erscheint nach kurzer Zeit der MicroPython-Prompt, und wir können die ersten Gehversuche mit dem Raspberry Pi Pico unternehmen.

```
MicroPython v1.22.1 on 2024-01-05; Raspberry Pi Pico W with RP2040
Type "help()" for more information.
>>>
```

Abbildung 13: Raspi pico meldet sich mit der Versionsnummer