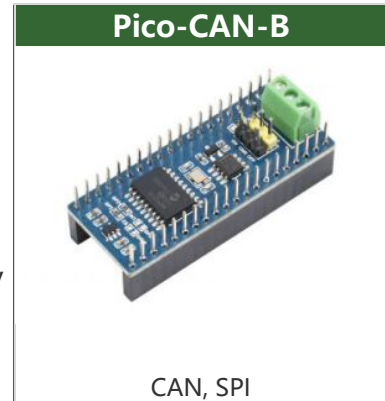


Overview

Introduction

It is a CAN module designed for Raspberry Pi Pico, enabling reliable long-range communication for Raspberry Pi Pico with other devices.



Features

- Standard Raspberry Pi Pico header, supports Raspberry Pi Pico series boards.
- Features CAN function, adopts SPI interface CAN controller MCP2515 with transceiver SIT65HVD230DR.
- Comes with online development resources and manual (Raspberry Pi Pico C/C++ and MicroPython examples).

Specifications

- Operating voltage: 3.3V~5V
- CAN controller: MCP2515
- Control interface: SPI
- CAN transceiver: SIT65HVD230DR
- Baudrate: 5K~1000Kbps
- Dimensions: 52 x 21mm

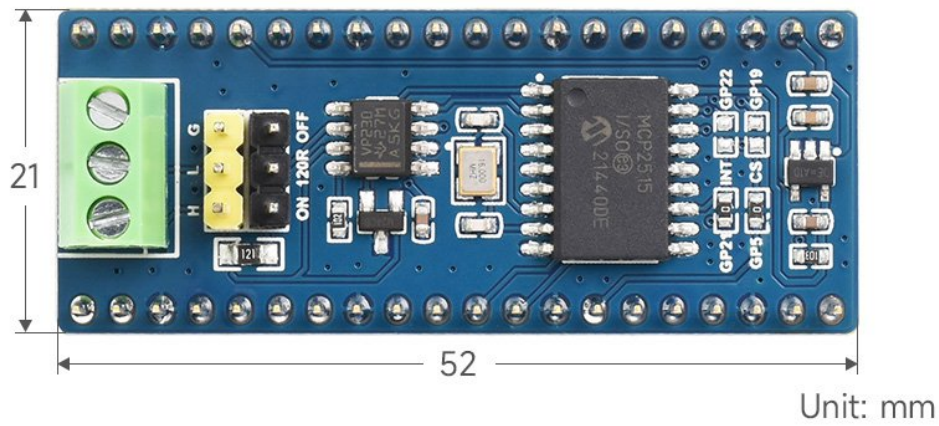
Pinout Definition

GP0	1	40	VBUS	VBUS	3.3V~5.5V power supply	
GP1	2	39	VSYS	VSYS	1.8V~5.5V power supply	
GND	3	38	GND	GND	Ground	
GP2	4	37	3V3_EN			
GP3	5	36	3V3(OUT)			
GP4	6	35	ADC_VREF	GP4	SPI0_MISO	SPI data output
GP5	7	34	GP28	GP5	SPI_CS0	SPI chip select (default)
GND	8	33	GND	GP6	SPI_SCLK	SPI clock input
GP6	9	32	GP27	GP7	SPI0_MOSI	SPI data input
GP7	10	31	GP26			
GP8	11	30	RUN			
GP9	12	29	GP22			
GND	13	28	GND			

GP10	14		27	GP21
GP11	15		26	GP20
GP12	16		25	GP19
GP13	17		24	GP18
GND	18		23	GND
GP14	19		22	GP17
GP15	20		21	GP16

GP19	SPI_CS1	SPI chip select (optional)
GP21	INT1	Interrupt output (default)
GP22	INT2	Interrupt output (optional)

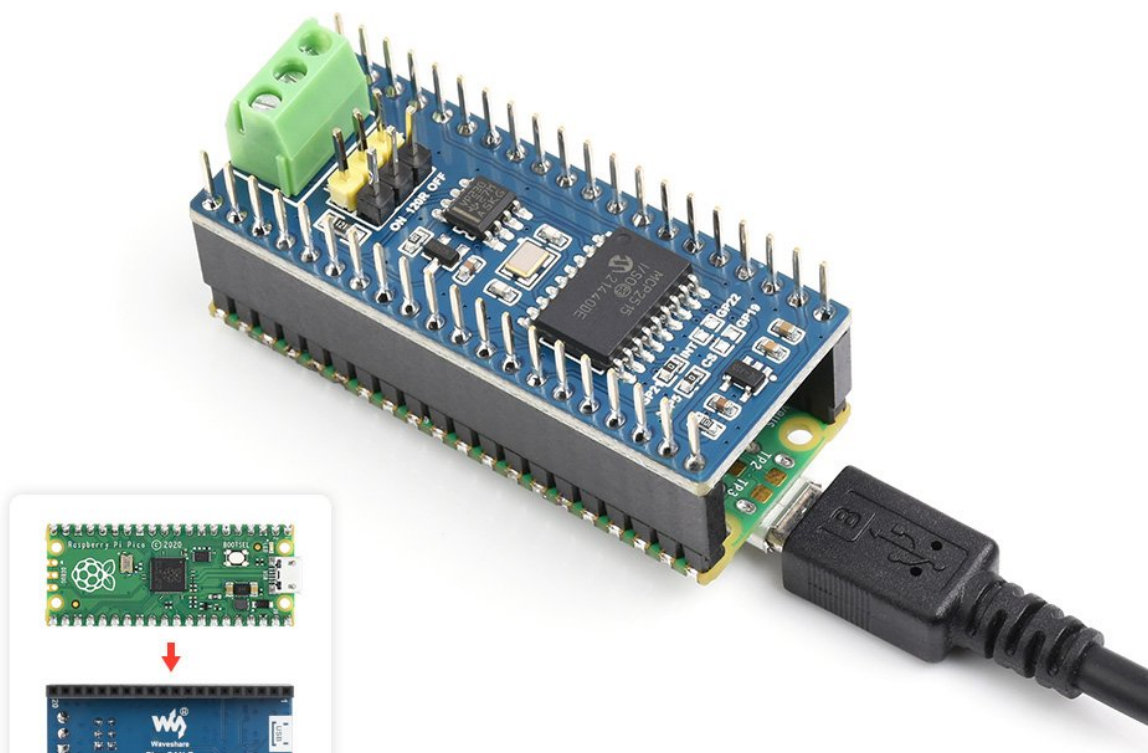
Dimensions



Pico User Guide

Hardware Connection

Please take care of the direction when you connect Pico, a USB port is printed to indicate. You can also check the pin of Pico and the module when connecting.





* Please correctly connect the Module and Raspberry Pi Pico as the picture shown.

Demo Download

1. Directly download it on the Raspberry Pi: Open the Raspberry Pi and execute:

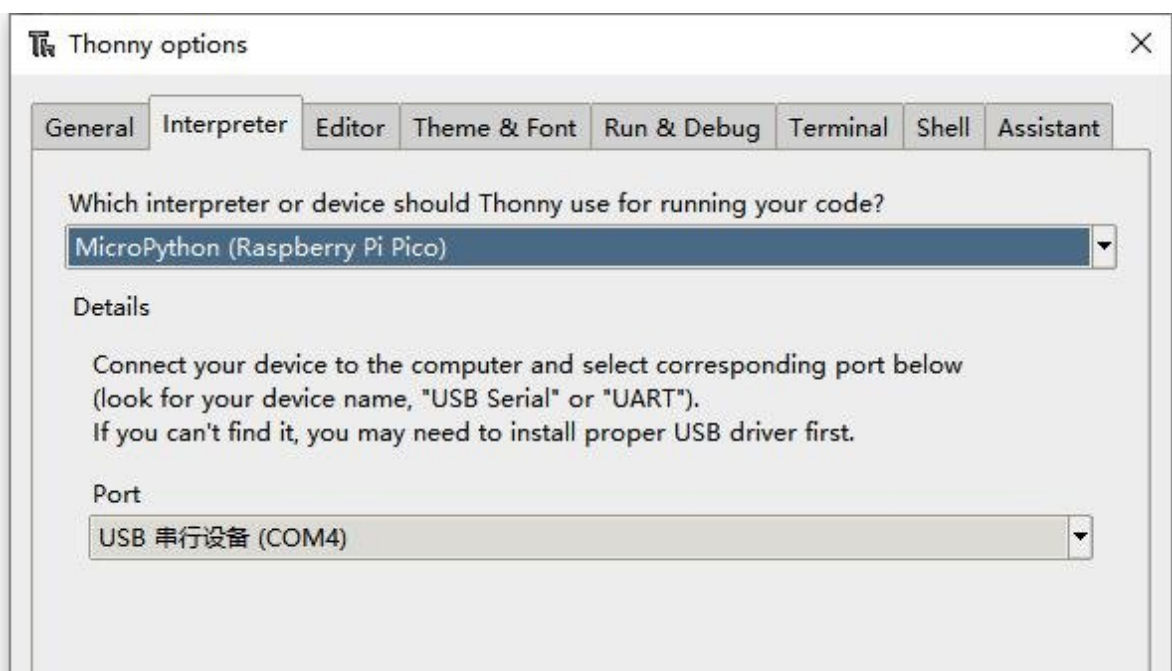
```
sudo apt-get install p7zip-full
cd ~
sudo wget https://files.waveshare.com/upload/8/8a/Pico-CAN-B-Code.7z
7z Pico_Pico-CAN-B-Code.7z -o./Pico-CAN-B-Code.7z
cd ~/Pico-CAN-B-Code
```

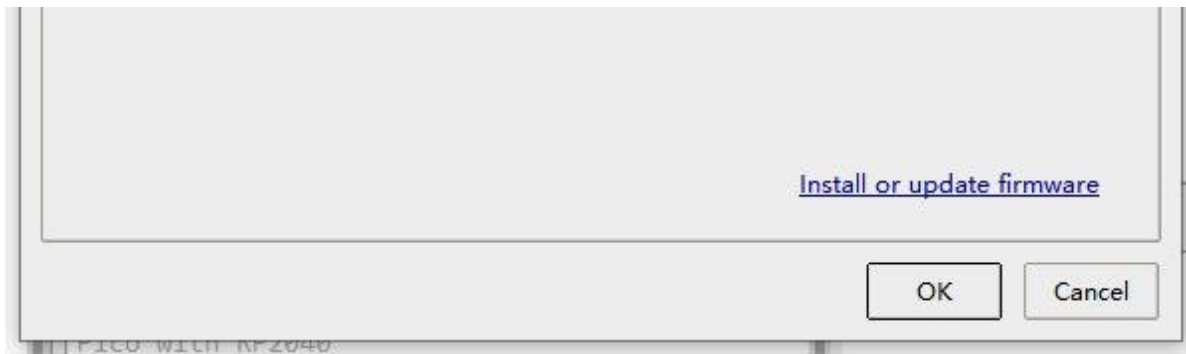
2. Also, you can download the demo in the [#Resource](#).

How to Use the Demo

Python

- 1. Press and hold the BOOTSET button on the Pico board, connect the pico to the USB interface of the computer through the Micro USB cable, and release the button after the computer recognizes a removable hard disk (RPI-RP2).
- 2. Copy the pico_micropython_XXXXX.uf2 file in the python directory to the recognized removable disk (RPI-RP2).
 - [Download the official firmware](#)
- 3. Open Thonny IDE (Note: Please use the latest version of Thonny, otherwise there is no Pico support package, the latest version in Windows is v3.3.3).
- 4. Click Tools -> Settings -> Interpreter, and select Pico and the corresponding port as shown in the figure.





- 5. File -> Open -> MCP2512.py file, click to run, as shown in the following figure:

```
Shell X
MicroPython v1.13-290-g556ae7914 on 2021-01-21; Raspberry Pi Pico with RP2040
Type "help()" for more information.
>>> %Run -c $EDITOR_CONTENT
```

This is a transceiver demo, connect to another CAN device (A-A B-B connection), configure the frame ID 0x123, and then send 1-8 in turn.

Resource

Document

- [Schematic diagram](#)
- [SN65HVD230](#)
- [MCP2515](#)

Demo

- [Demo](#)

Development Software

- [Thonny Python IDE](#)
- [Zimo221.7z](#)
- [Win32DiskImager](#)

Pico Quick Start

Download Firmware

- [MicroPython Firmware Download](#)
- [C_Blink Firmware Download](#)

[\[Expand\]](#)

Video Tutorial

[\[Expand\]](#)

- [Pico Tutorial I - Basic Introduction](#)
- [Pico Tutorial II - GPIO](#) [\[Expand\]](#)
- [Pico Tutorial III - PWM](#) [\[Expand\]](#)
- [Pico Tutorial IV - ADC](#) [\[Expand\]](#)
- [Pico Tutorial V - UART](#) [\[Expand\]](#)
- [Pico Tutorial VI - To be continued...](#) [\[Expand\]](#)

MicroPython Series

- [\[MicroPython\] machine.Pin Function](#)
- [\[MicroPython\] machine.PWM Function](#)
- [\[MicroPython\] machine.ADC Function](#)
- [\[MicroPython\] machine.UART Function](#)
- [\[MicroPython\] machine.I2C Function](#)
- [\[MicroPython\] machine.SPI Function](#)
- [\[MicroPython\] rp2.StateMachine](#)

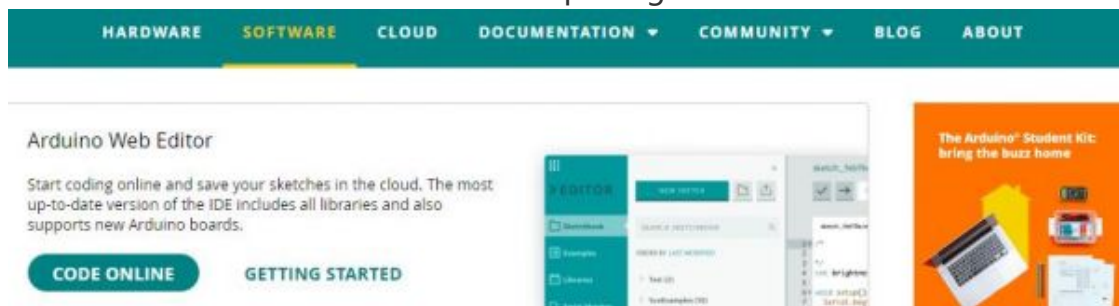
C/C++ Series

- [\[C/C++\] Windows Tutorial 1 - Environment Setting](#)
- [\[C/C++\] Windows Tutorial 1 - Create New Project](#)

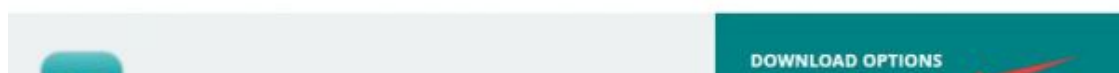
Arduino IDE Series

Install Arduino IDE

1. Download the Arduino IDE installation package from [Arduino website](#).



Downloads





Arduino IDE 2.0.0

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

For more details, please refer to the [Arduino IDE 2.0 documentation](#).

Nightly builds with the latest bugfixes are available through the section below.

SOURCE CODE
The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#).

Windows Win 10 and newer, 64 bits
Windows MSI installer
Windows ZIP file

Linux AppImage 64 bits (X86-64)
Linux ZIP file 64 bits (X86-64)

macOS 10.14: "Mojave" or newer, 64 bits

2. Just click on "JUST DOWNLOAD".

Support the Arduino IDE

Since the release 1.x release in March 2015, the Arduino IDE has been downloaded **69,954,557** times — impressive! Help its development with a donation.



JUST DOWNLOAD **CONTRIBUTE & DOWNLOAD**

Learn more about [donating to Arduino](#).

3. Click to install after downloading.

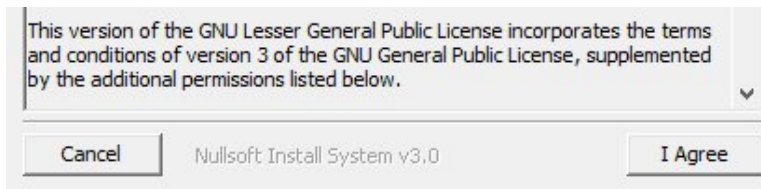


Arduino Setup: License Agreement

Please review the license agreement before installing Arduino. If you accept all terms of the agreement, click I Agree.

GNU LESSER GENERAL PUBLIC LICENSE
Version 3, 29 June 2007
Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.



4. **Note: You will be prompted to install the driver during the installation process, we can click Install.**

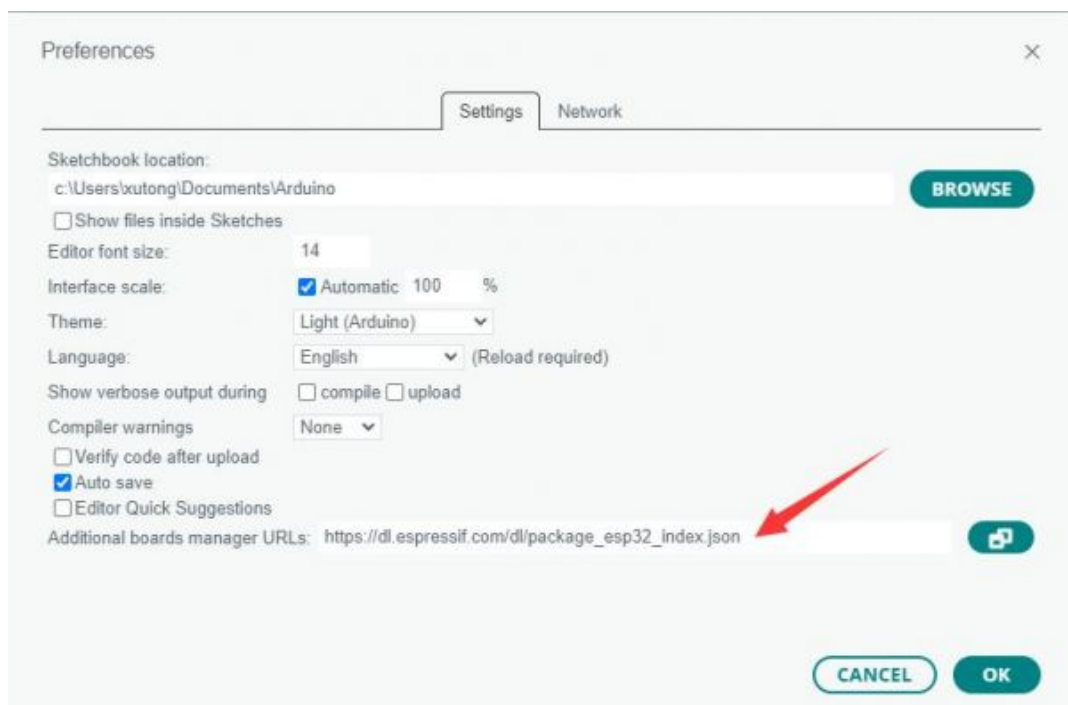
Install Arduino-Pico Core on Arduino IDE

1. Open Arduino IDE, click the File on the left corner and choose "Preferences".



2. Add the following link in the additional development board manager URL, then click OK.

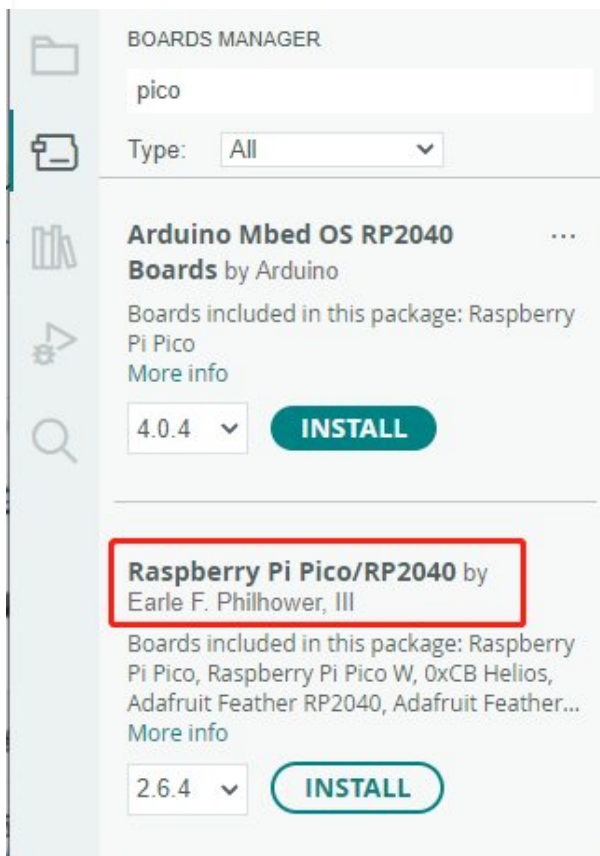
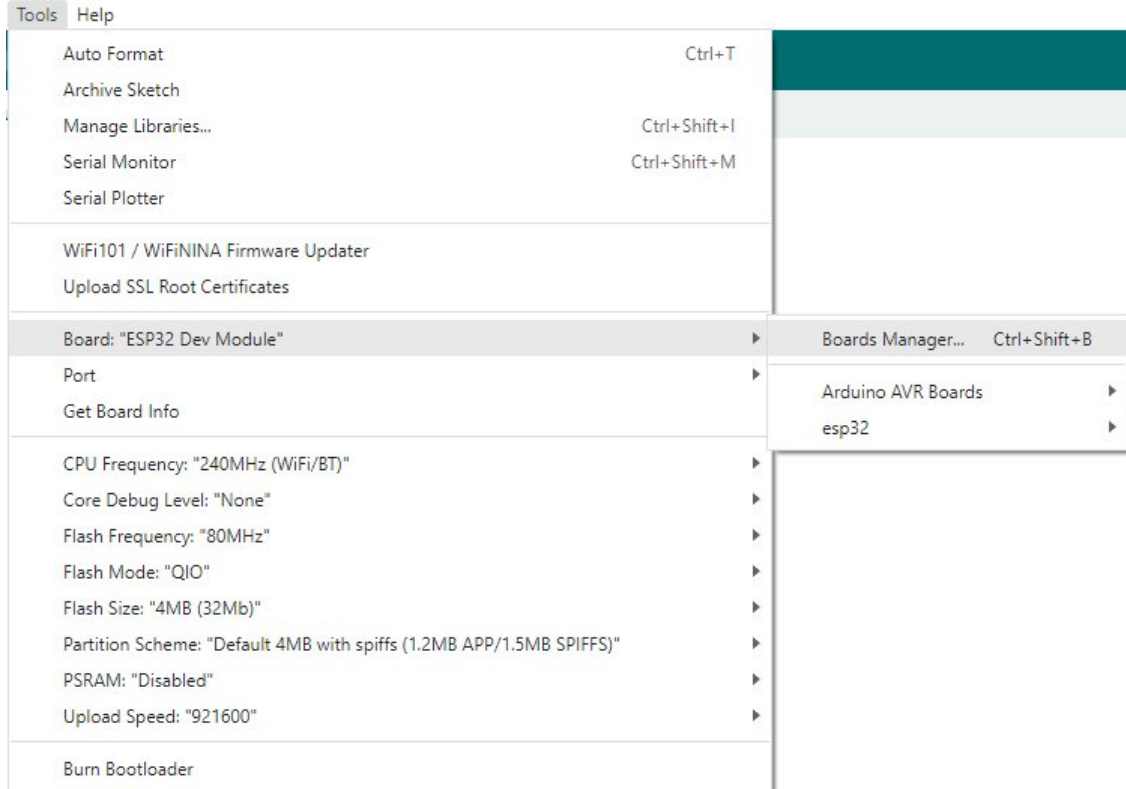
```
https://github.com/earlephilhower/arduino-pico/releases/download/global/package_rp2040_index.json
```



Note: If you already have the ESP8266 board URL, you can separate the URLs with commas like this:

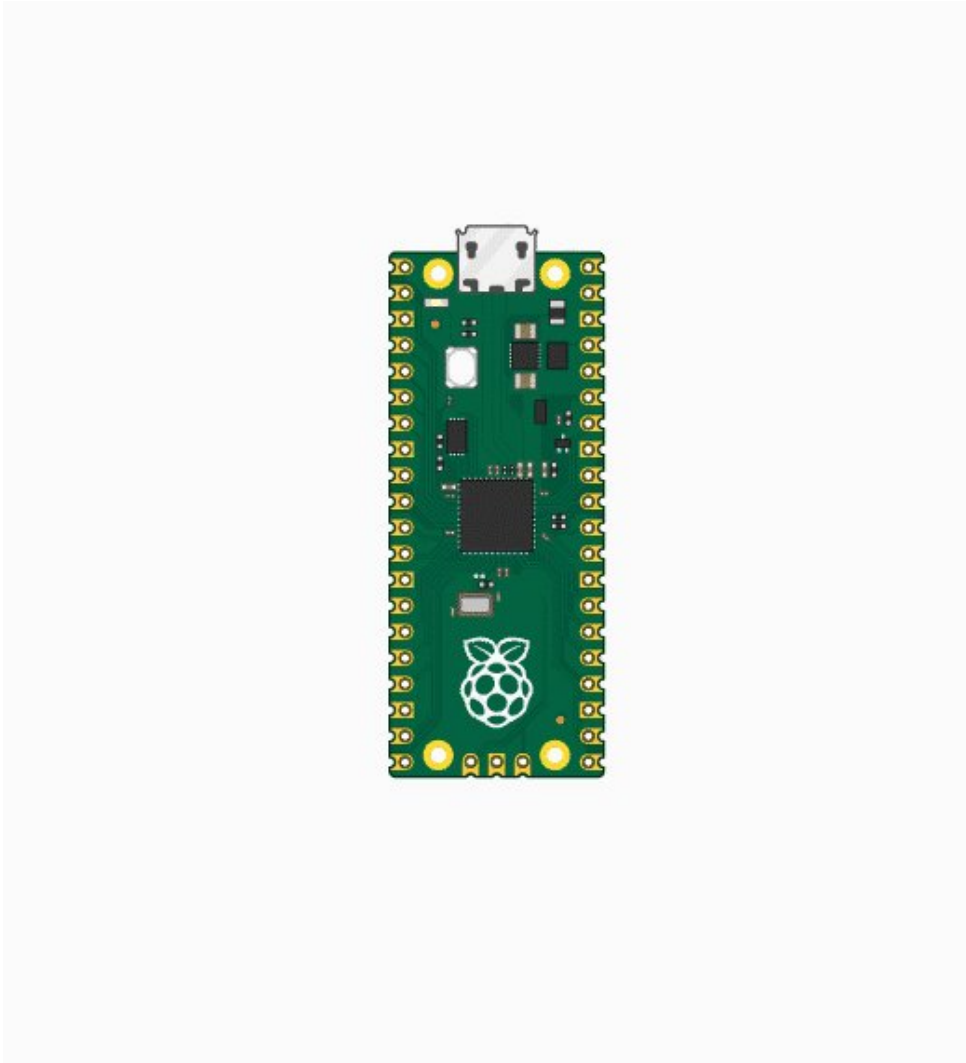
https://dl.espressif.com/dl/package_esp32_index.json,https://github.com/earlephilhower/arduino-pico/releases/download/global/package_rp2040_index.json

3. Click on Tools -> Dev Board -> Dev Board Manager -> Search for pico, it shows installed since my computer has already installed it.

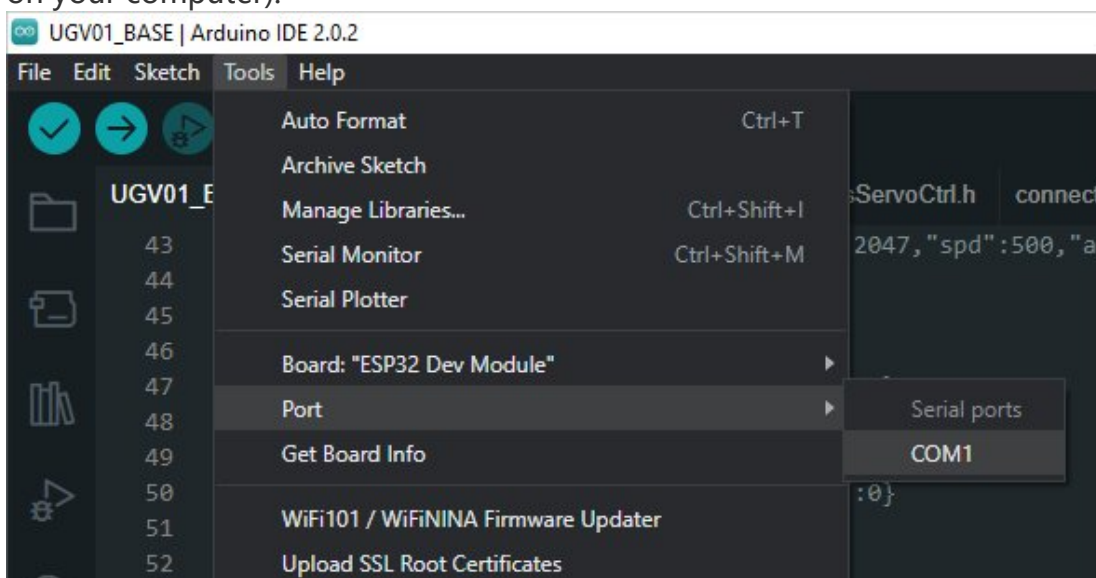


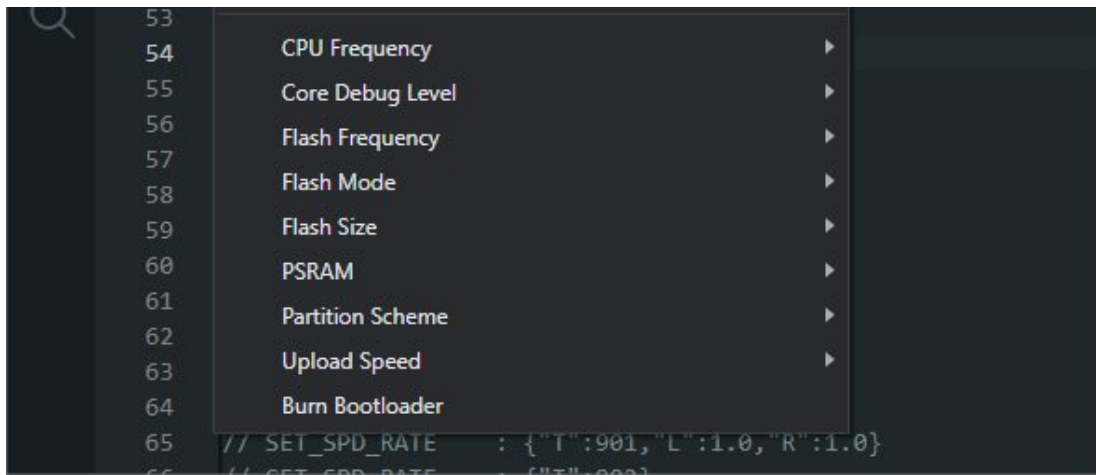
Upload Demo At the First Time

1. Press and hold the BOOTSET button on the Pico board, connect the Pico to the USB port of the computer via the Micro USB cable, and release the button when the computer recognizes a removable hard drive (RPI-RP2).

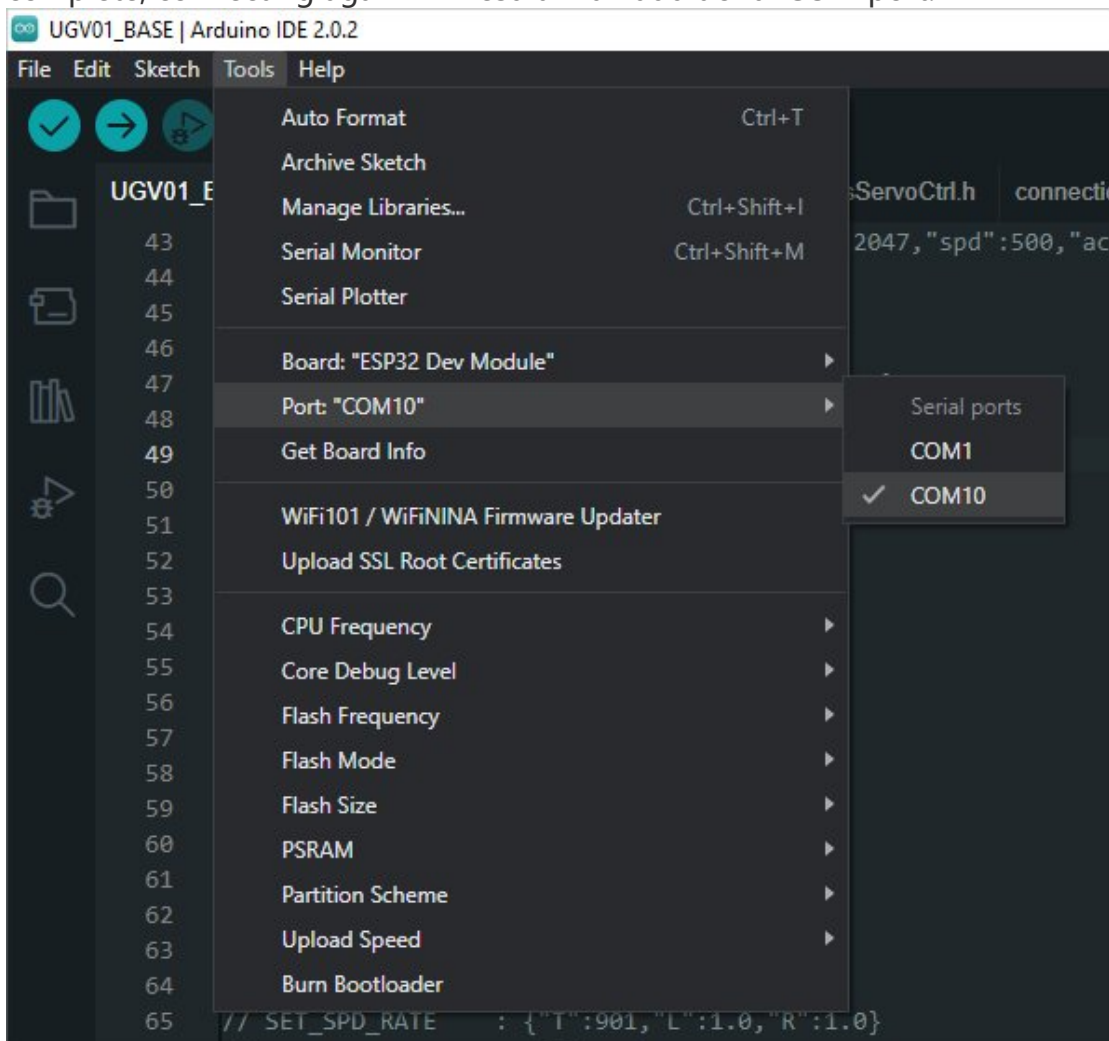


2. Download the demo, open arduino\PWM\D1-LED path under the D1-LED.ino.
3. Click Tools -> Port, remember the existing COM, do not need to click this COM (different computers show different COM, remember the existing COM on your computer).

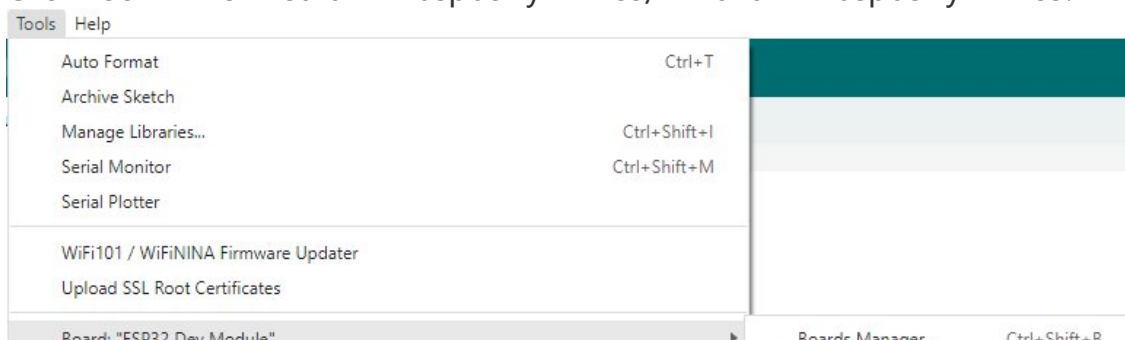


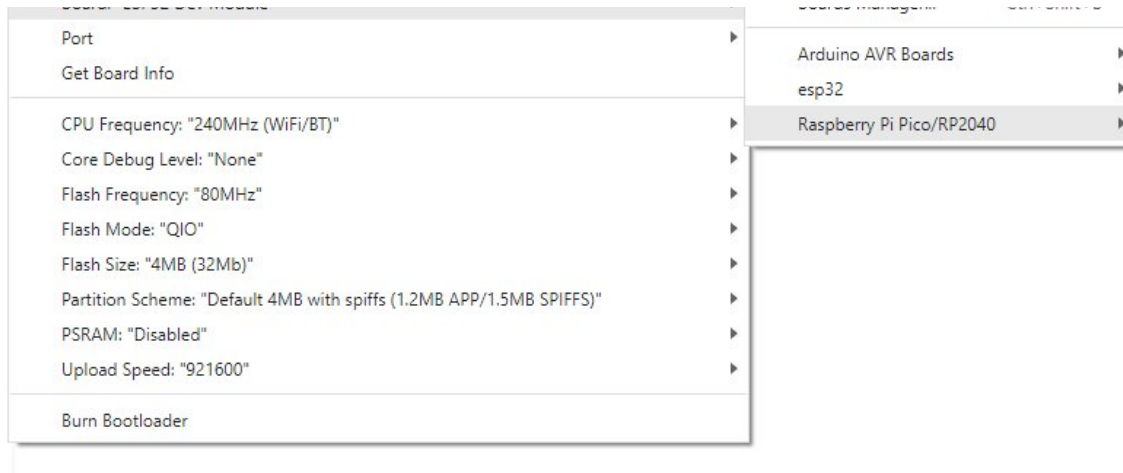


4. Connect the driver board to the computer with a USB cable, then click Tools -> Ports, select uf2 Board for the first connection, and after the upload is complete, connecting again will result in an additional COM port.

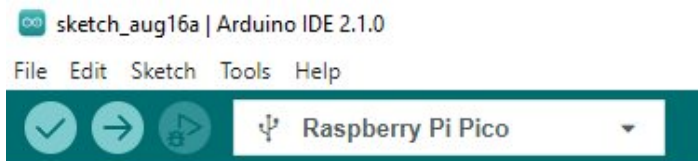


5. Click Tool -> Dev Board -> Raspberry Pi Pico/RP2040 -> Raspberry Pi Pico.





6. After setting, click the right arrow to upload.



- If you encounter problems during the period, you need to reinstall or replace the Arduino IDE version, uninstall the Arduino IDE needs to be uninstalled cleanly, after uninstalling the software you need to manually delete all the contents of the folder C:\Users\[name]\AppData\Local\Arduino15 (you need to show the hidden files in order to see it) and then reinstall.

Pico-W Series Tutorial (To be continued...)

Open Source Demo

- [MicroPython Demo \(GitHub\)](#)
- [MicroPython Firmware/Blink Demo \(C\)](#)
- [Official Raspberry Pi C/C++ Demo](#)
- [Official Raspberry Pi MicroPython Demo](#)
- [Arduino Official C/C++ Demo](#)

Open-source Demo

- [MicroPython video demo \(GitHub\)](#)
- [MicroPython firmware/Blink demo\(C\)](#)
- [Raspberry Pi's official C/C++ sample demo \(GitHub\)](#)
- [Raspberry Pi's official micro python sample demo \(GitHub\)](#)
- [Raspberry Pi Pico Demo.zip](#)

Support

Technical Support

If you need technical support or have any feedback/review, please click the **Submit Now** button to submit a ticket, Our support team will check and reply to you within 1 to 2 working days. Please be patient as we make every effort to help you to resolve the issue.

Working Time: 9 AM - 6 AM GMT+8 (Monday to Friday)

[Submit Now](#)