Overview

Pico-ETH-CH9121

Ethernet To UART Converter For Raspberry Pi Pico, 10/100M Ethernet, Enabling Network Communication Through UART.

Features

- Standard Raspberry Pi Pico header, supports Raspberry Pi Pico series.
- Onboard Ethernet to UART transparent transceiver, 2-CH UART, standalone transparent transmission for each channel.
- Onboard jumpers for pin configurations.
- Embedded Ethernet MAC and PHY layers.
- Bi-direction transparent data transmission between UART and Ethernet.
- 10/100M, full-duplex/half-duplex auto-negotiation Ethernet interface, 802.3-compliant.
- Auto-MDI/MDIX, detect and switch cable type automatically.
- Supports DHCP auto-obtained IP and DNS domain access.
- Network parameter configuration like chip operating mode, port, IP, via host computer software or UART command
- Four operating modes: TCP CLIENT, TCP SERVER, UDP CLIENT, UDP SERVER.
- Full-duplex or half-duplex UART communication supports RS485 RX/TX auto switch (external RS485 controller required).
- Supports virtual serial port software (provided).
- KEEPALIVE mechanism support.

Specifications

- Operating votlage: 3.3V/5V
- Operating mode: TCP/UDP
- UART TTL: 3.3V / 5V compatible
- Operating temperature: ~40°C ~ 85°C
- Operating current: 140mA
- Baudrate: 300bps ~921.6Kbps

- Dimensions: 74.54 x 21.00 (mm)
- Storage Condition: -55°C ~ 125°C

Pinouts

Pico-ETH-CH9121-details-inter.jpg

About the CH9121

The CH9121 is a transparent network serial port chip that supports bidirectional and transparent transmission of serial port data and network data. It supports four working modes: TCP CLIENT/SERVER and UDP CLIENT/SERVER. The serial port baud rate ranges from 300bps to 921600bps. Before using, you should set network and serial port parameters for the chip by the NetModuleConfig.exe software or serial commands. After the configuration is complete, the CH9121 saves the configuration parameters to the internal storage space. After the chip is reset, the CH9121 could work based on the saved configuration values.

Default Parameters

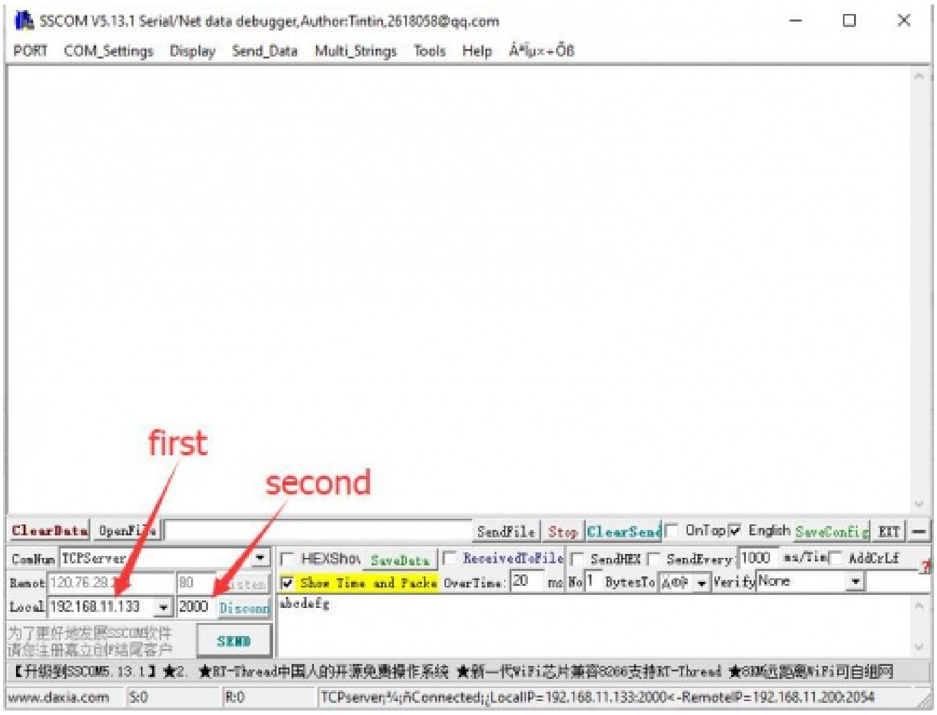
The UART 2 is disabled and the UART 1 works in TCP CLIENT mode by default.

- **The network parameters of UART1 (port 1)**
    - Device IP: 192.168.1.200
    - Subnet mask: 255.255.255.0
    - Default gateway: 192.168.1.1
    - Device port: 2000
    - Target IP: 192.168.1.100
    - Target port: 1000
- **The Serial parameters of port**
    - Baud rate: 9600
    - Timeout: 0
    - Data bit: 8
    - Stop bit: 1
    - Parity bit: None
    - Clear buffer: Never

Configuration

TCP Client

- Connect the Pico-ETH-CH9121 (ETH module hereafter) to the router, and use the host PC as a TCP server. The PC should be connected to the same LAN as the ETH module.
- Use the SSCOM software, set the port to TCP server and you can check the local IP (the Target IP) and the port (Target port).



ComNum TCPServer

Remot 120.76.28.1 80

Local 192.168.11.133 2000

☐ HEXShow ☐ ReceivedToFile ☐ SendHEX ☐ SendEvery: 1000 ms/Tim ☐ AddCrLf

☑ Show Time and Packe OverTime: 20 ms No 1 BytesTo Verify None

☐ OnTop ☑ English

abcdefg

为了更好地发展SSCOM软件
请您注册嘉立创P结尾客户

【升级到SSCOM5.13.1】★2. ★RT-Thread中国人的开源免费操作系统 ★新一代WiFi芯片兼容8266支持RT-Thread ★8KM远距离WiFi可自组网

www.daxia.com S:0 R:0 TCPserver¦¼¡ñConnected¡¡LocalIP=192.168.11.133:2000<-RemoteIP=192.168.11.200:2054

- Run the NetModuleConfig.exe software on your PC.
    1. Click the Search button to detect the device.
    2. Double-Click the device detected, then you can modify the network parameters in the left area.
    3. After configuration, the ETH module will auto-restart, wait for a moment, search the device again, and check the setting.
    4. Note that you need to modify the parameters according to the actual situation. The Mode should be TCP CLIENT, and the target IP, and port is the same as the TCP server.

NetModuleConfig-1.jpg

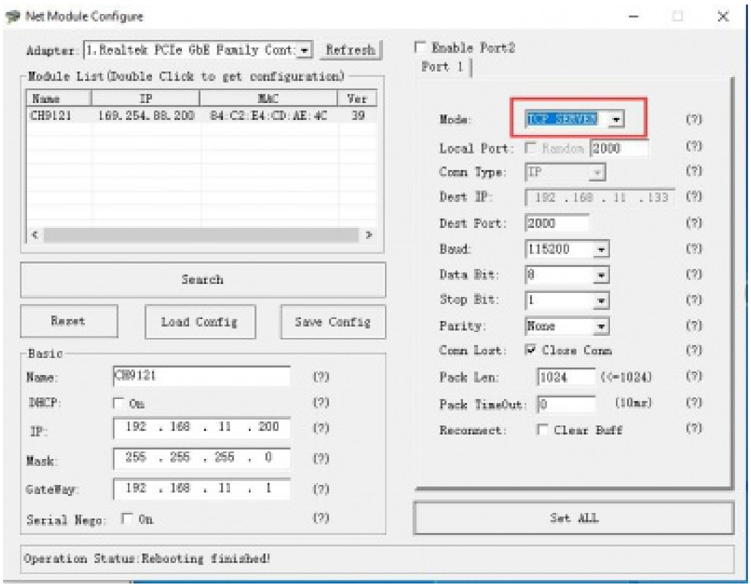
- After setting, you can listen to the data with the SSCOM software.

SSCOM-2.jpg

TCP SERVER

- Connect the Pico-ETH-CH9121 (ETH module hereafter) to the router, and use the host PC as a TCP CLIENT.
- Run the NetModuleConfig.exe software on your PC.

1. Click the Search button to detect the ETH module.
2. Double-click the device detected.
3. Set the mode to TCP SERVER and modify other parameters according to the actual situation.
4. Click the Set ALL button to save the setting.



Adapter: 1.Realtek PCIe GbE Family Cont

Module List(Double Click to get configuration)

Name	IP	MAC	Ver
CH9121	169.254.88.200	84:C2:E4:CD:AE:4C	39

Basic
Name: CH9121 (?)
DHCP: □ On (?)
IP: 192 . 168 . 11 . 200 (?)
Mask: 255 . 255 . 255 . 0 (?)
GateWay: 192 . 168 . 11 . 1 (?)
Serial Nego: □ On (?)

□ Enable Port2
Port 1

Mode: TCP SERVER (?)
Local Port: □ Random 2000 (?)
Conn Type: IP (?)
Dest IP: 192 . 168 . 11 . 133 (?)
Dest Port: 2000 (?)
Baud: 115200 (?)
Data Bit: 8 (?)
Stop Bit: 1 (?)
Parity: None (?)
Conn Lost: ☑ Close Conn (?)
Pack Len: 1024 (<=1024) (?)
Pack TimeOut: 0 (10ms) (?)
Reconnect: □ Clear Buff (?)

Operation Status:Rebooting finished!

- User the SSCOM software, configure it as TCP CLIENT and connect to the SERVER (The ETH module).

SSCOM-3.jpg

UDP CLIENT/SERVER

The UDP modes are the same as the TCP, the only difference is the mode is UDP CLIENT/SERVER but not the TCP CLIENT/SERVER.

Pico Quick Start

Hardware Connection

ETH	Pico	Description
5V	VSYS	Power input
GND	GND	Ground
RXD1	GP0	UART 1 data input
TXD1	GP1	UART 1 data output
RXD2	GP4	UART 2 data input
TXD2	GP5	UART 2 data output
CFG0	GP14	Network configrate enable pin
RST1	GP17	Reset pin

Pico-ETH-CH9121-details-3.jpg

- If you want to use the screen, please unplug the CFG0 mini jumper cap on the Pico-ETH-CH9121.
- Remove the jumper cap on the RST1 pin, and after powering on the pico, the Pico-ETH-CH9121 will start directly, don't need to download the program.

Configurate Environment

To use the Pico, you need to configure your PC or the Raspberry Pi first.
Please refer to the official guide about the setting Pico Manual.

Examples

We use Raspberry Pi boards as examples.

Download examples

Open the terminal and run the following commands:

```
sudo apt-get install p7zip-full
cd ~
sudo wget  https://files.waveshare.com/upload/a/a4/Pico_ETH_CH9121_CODE.7z
7z x Pico_ETH_CH9121_CODE.7z -o./Pico_ETH_CH9121_CODE
cd ~/Pico_ETH_CH9121_CODE
cd Pico/c/build/
```

C examples

- Go into the c directory.

```
cd ~/Pico_ETH_CH9121_CODE/Pico/C/
```

You can use the Pico_ETH_CH9121_CODE/Pico/C/Serial Port Parameter Configuration: configure module.

Pico_ETH_CH9121_CODE/Pico/C/RX_TX: Echo examples, receive data, and echo

- Go into the examples directory and export sdk

```
cd build
export PICO_SDK_PATH=../../pico-sdk
```

You need to modify the path to the actual one if you use a direct path.

- Generate the Makefile file

```
cmake ..
```

- Build the examples.

```
make -j9
```

- After building, copy the .uf2 file to the Pico.

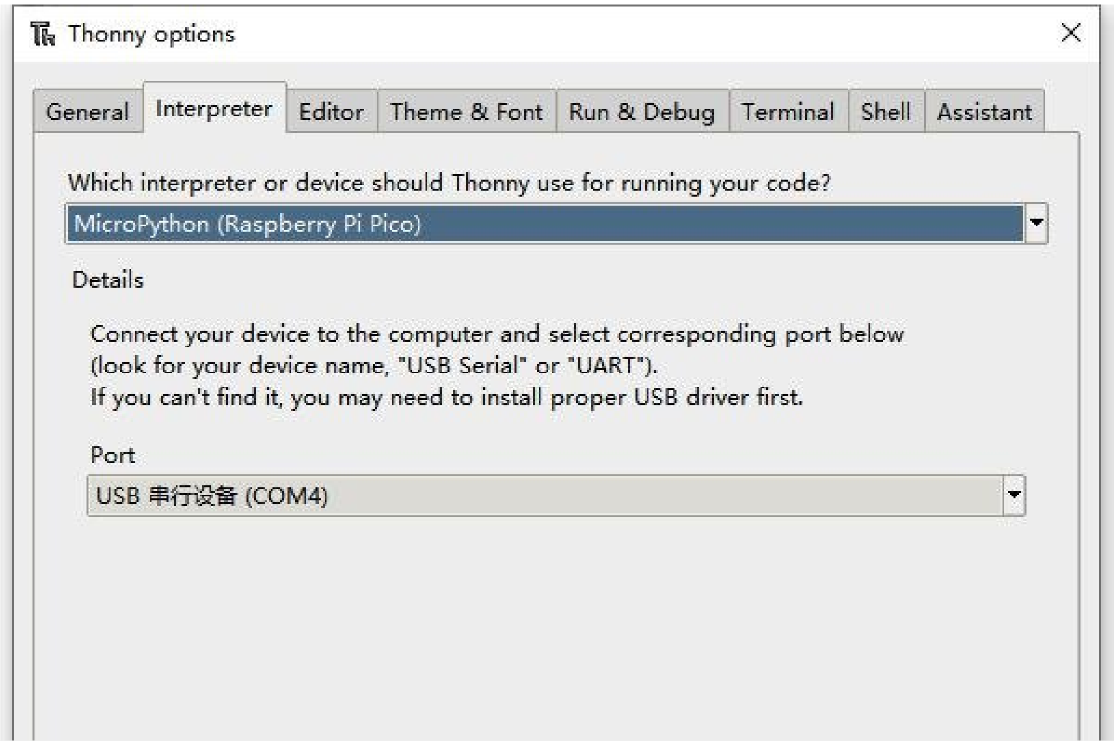
Press the BOOTSEL button of Pico and hold it, connect the Pico to your Pi by micro USB cable, and then release the button. The portable disk RPI-RP2 is recognized, you need to copy the uf2 file to the portable disk.

```
cp main.uf2 /media/pi/RPI-RP2/
```

Python examples

Windows

1. Press and hold the BOOTSET button on the Pico board, connect the Pico to the USB port 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 rp2-pico-20210418-v1.15.uf2 file in the python directory to the recognized removable disk (RPI-RP2).
3. Open Thonny IDE (note: use the latest version of Thonny, otherwise there is no Pico support package, the latest version under Windows is v3.3.3).
4. Click Tools -> Settings -> Interpreter, select Pico, and the corresponding port as shown in the figure.



Which interpreter or device should Thonny use for running your code?

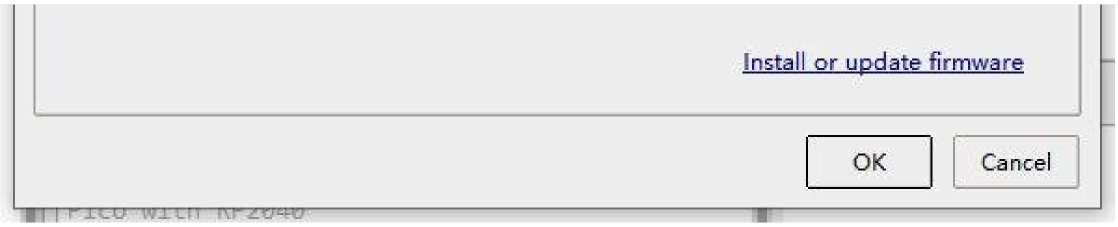
MicroPython (Raspberry Pi Pico)

Details

Connect your device to the computer and select corresponding port below
(look for your device name, "USB Serial" or "UART").
If you can't find it, you may need to install proper USB driver first.

Port

USB 串行设备 (COM4)



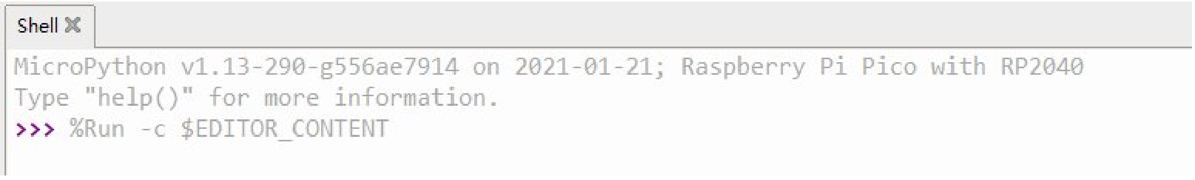
This demo provides two programs:

Serial Port Parameter Configuration.py: This program is used to configure the mode through the serial port.

RX_TX.py: This is used to send and receive information and return what is received.

5. File -> Open -> RX_TX.py, click to run, as shown below:

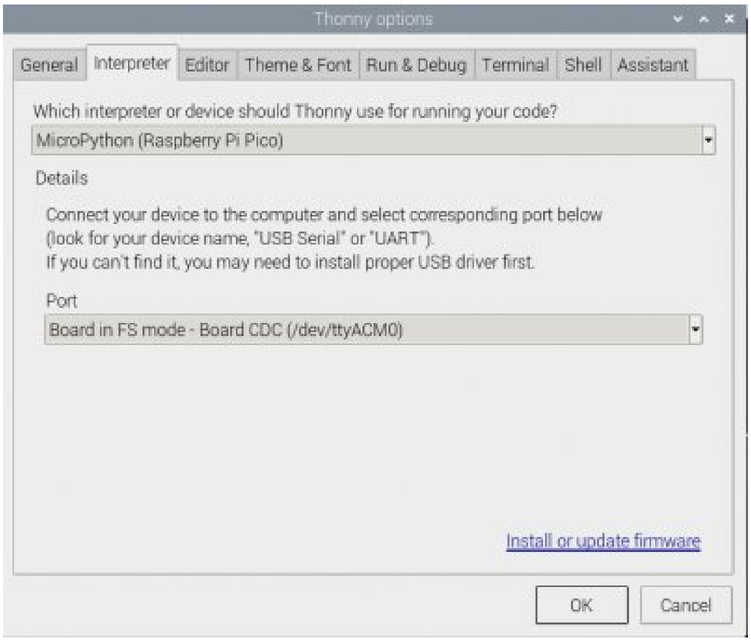
Note: If you want to start automatically, please rename RX_TX.py to main.py and save it to Pico.



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

Raspberry Pi

- Press and hold the BOOTSEL button of Pico, connect it to Raspberry Pi or PC by micro USB cable, and then release the button.
- The Pico is recognized as a portable disk RPI-RP2, copy the rp2-pico-20210418-v1.15.uf2 file to the portable disk.
- Open the Thonny IDE.
- Choose Tools -> Options .. -> Interpreter.
- Choose MicroPython(Raspberry Pi Pico) and the related port.



Which interpreter or device should Thonny use for running your code?

MicroPython (Raspberry Pi Pico)

Details

Connect your device to the computer and select corresponding port below
(look for your device name, "USB Serial" or "UART").
If you can't find it, you may need to install proper USB driver first.

Port

Board in FS mode - Board CDC (/dev/ttyACM0)

- If your Thonny IDE doesn't support Pico, you need to update it to the newest version.

```
sudo apt upgrade thonny
```

- Click File -> Open... -> python/RX_TX.py to run the codes.

Codes Description

C codes

- Data types:

```
#define UCHAR unsigned char
#define UBYTE uint8_t
#define UWORD uint16_t
#define UDOUBLE uint32_t
```

- Initialize module:

```
void CH9121_init(void)；
```

- Parameters for configuring modules:

```
UCHAR CH9121_Mode            //Mode
UCHAR CH9121_LOCAL_IP[4]     //Device IP
UCHAR CH9121_GATEWAY[4]      //Gateway
UCHAR CH9121_SUBNET_MASK[4] //Subnet mask
UCHAR CH9121_TARGET_IP[4]    //Target IP
UWORD CH9121_PORT1           //Device port
UWORD CH9121_TARGET_PORT     //Target port
UDOUBLE CH9121_BAUD_RATE     //baud rate of serial
```

- You can configure the module by the following functions with serial commands

```
void CH9121_TX_4_bytes(UCHAR data, int command); //Can be used to configure the mode, random port, disconnect network, clear buffer, DHCP, UART2
void CH9121_TX_5_bytes(UWORD data, int command);//Can be used to set the port of Serial
void CH9121_TX_7_bytes(UCHAR data[], int command);//Can be used to set IP, subnet mask, gateway.
void CH9121_TX_BAUD(UDOUBLE data, int command);//Can be used to set the baud rate of Serial.
void CH9121_Eed()；  // Can be used to save settings to EEPROM, enable the setting, reset module, exit from setting mode
```

Python

You just need to modify the Serial Port Parameter Configuration.py for setting the module.

```
ODE = 1  #0:TCP Server 1:TCP Client 2:UDP Server 3:UDP Client
GATEWAY = (169, 254, 88, 1)   # GATEWAY
TARGET_IP = (169, 254, 88, 17)# TARGET_IP
LOCAL_IP = (169,254,88,70)    # LOCAL_IP
SUBNET_MASK = (255,255,255,0) # SUBNET_MASK
LOCAL_PORT1 = 5000             # LOCAL_PORT1
LOCAL_PORT2 = 4000             # LOCAL_PORT2
TARGET_PORT = 3000            # TARGET_PORT
BAUD_RATE = 115200            # BAUD_RATE
```

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

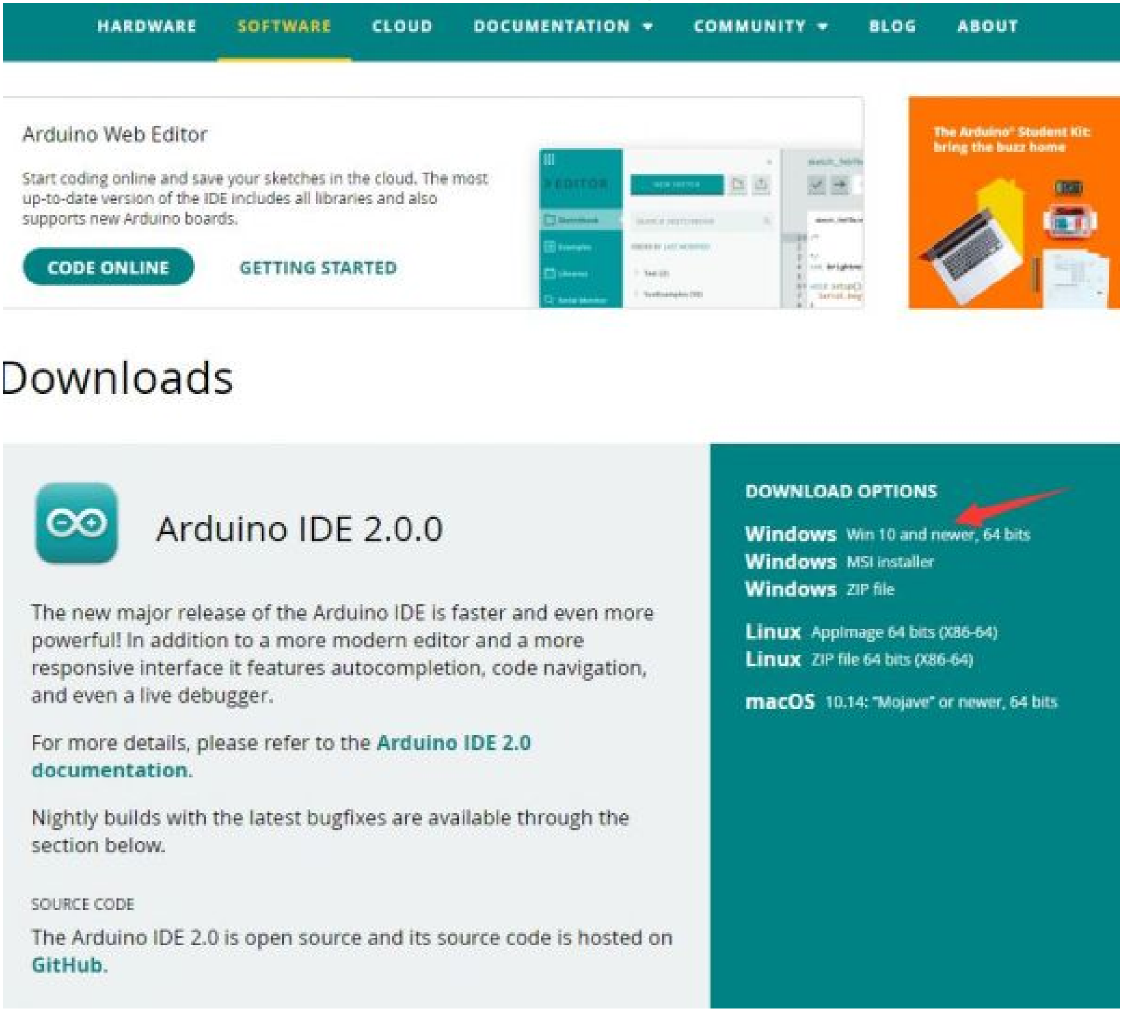
- 【C/C++】 Windows Tutorial 1 - Environment Setting

- 【C/C++】 Windows Tutorial 1 - Create New Project

Arduino IDE Series

Install Arduino IDE

1. Download the Arduino IDE installation package from Arduino website.



Arduino Web Editor

Start coding online and save your sketches in the cloud. The most up-to-date version of the IDE includes all libraries and also supports new Arduino boards.

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**.

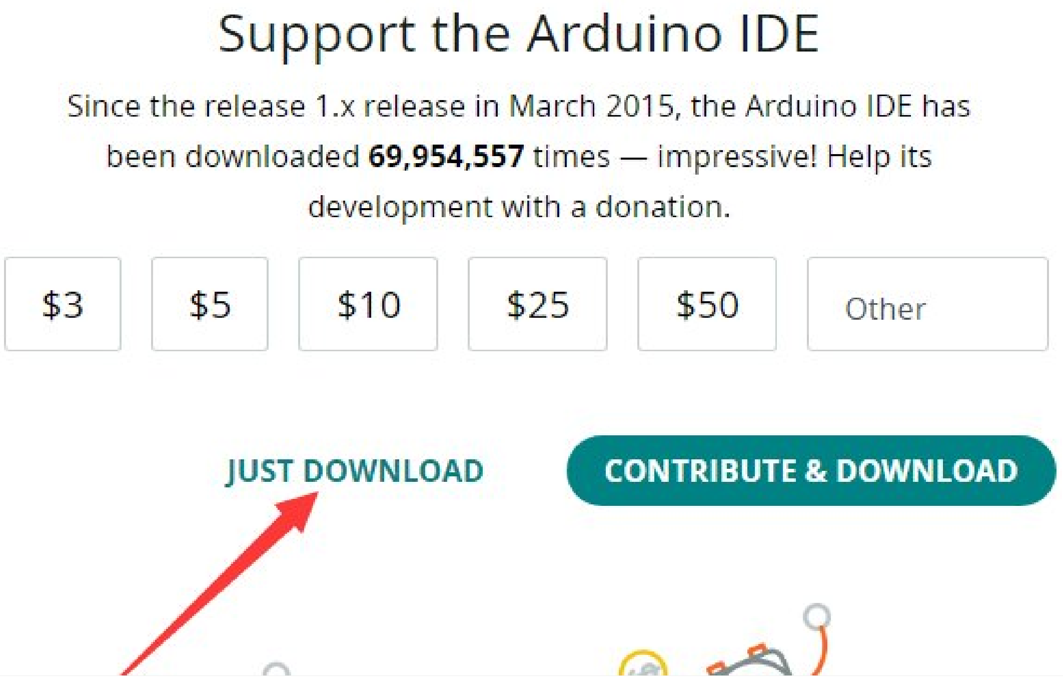
DOWNLOAD OPTIONS

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".

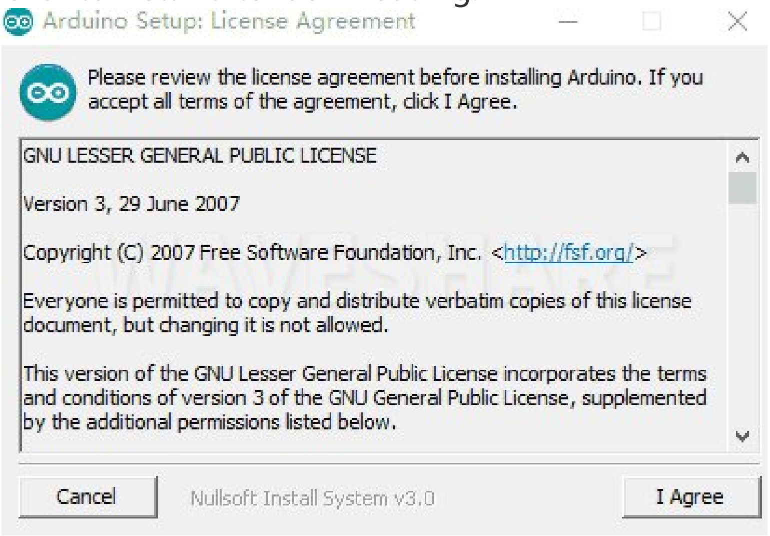


Support the Arduino IDE

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.

Learn more about **donating to Arduino.**

3. Click to install after downloading.



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/>

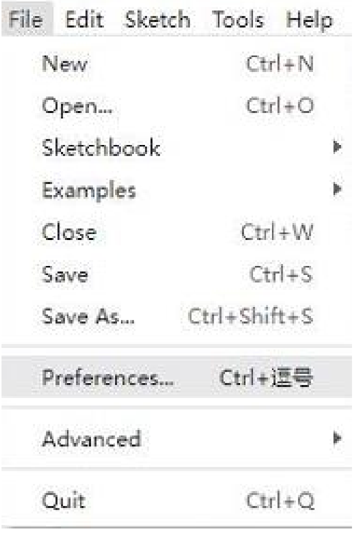
Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below.

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

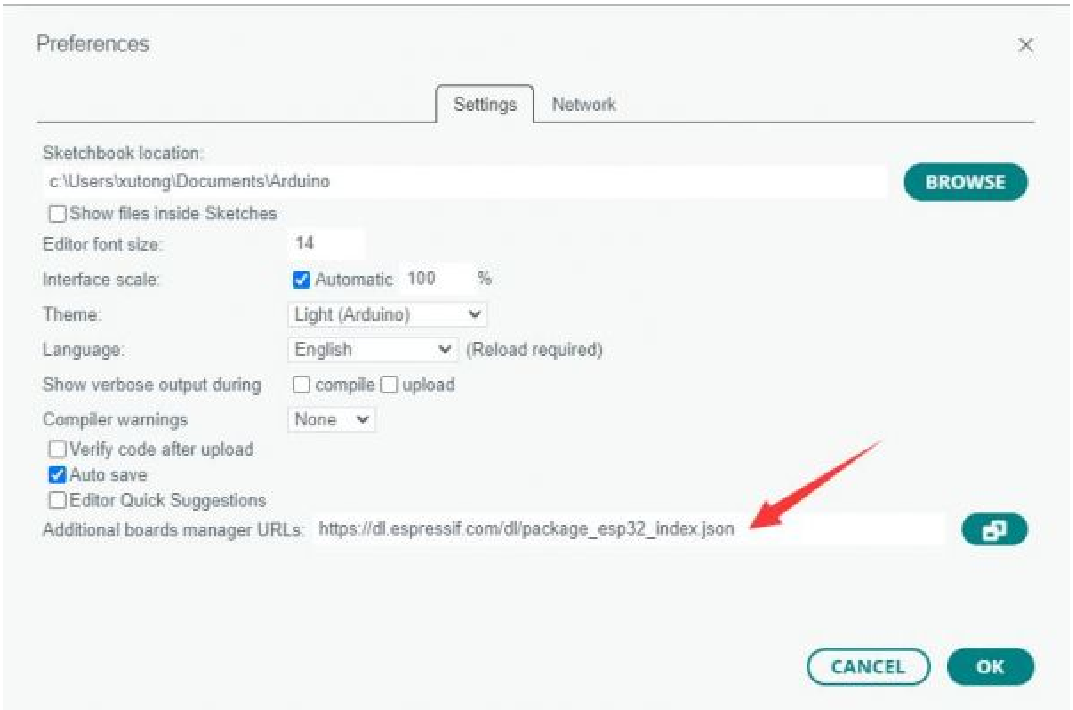
Install Arduino-Pico Core on Arduino IDE

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/globa
l/package_rp2040_index.json
```



Preferences

Settings | Network

Sketchbook location: c:\Users\xutong\Documents\Arduino

☐ Show files inside Sketches

Editor font size: 14

Interface scale: ☑ Automatic 100 %

Theme: Light (Arduino)

Language: English (Reload required)

Show verbose output during ☐ compile ☐ upload

Compiler warnings: None

☐ Verify code after upload

☑ Auto save

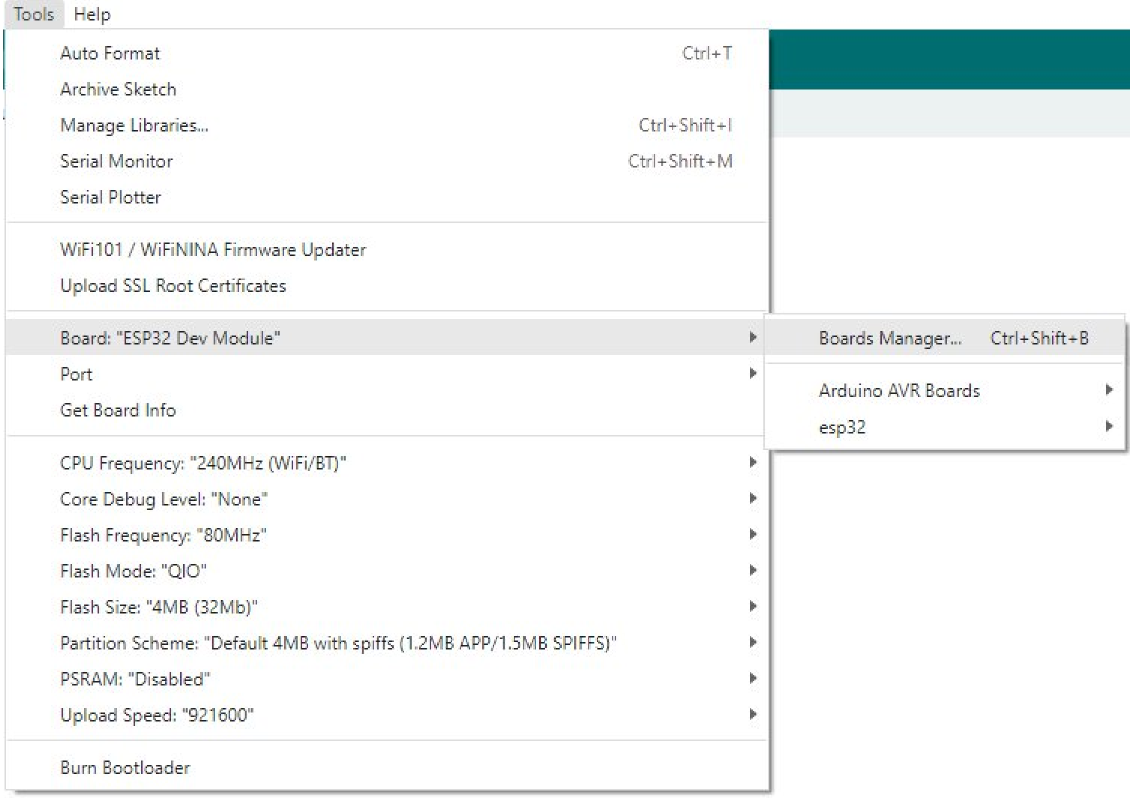
☐ Editor Quick Suggestions

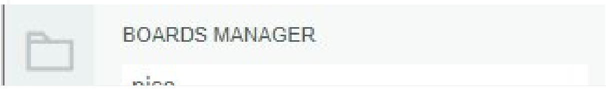
Additional boards manager URLs: https://dl.espressif.com/dl/package_esp32_index.json

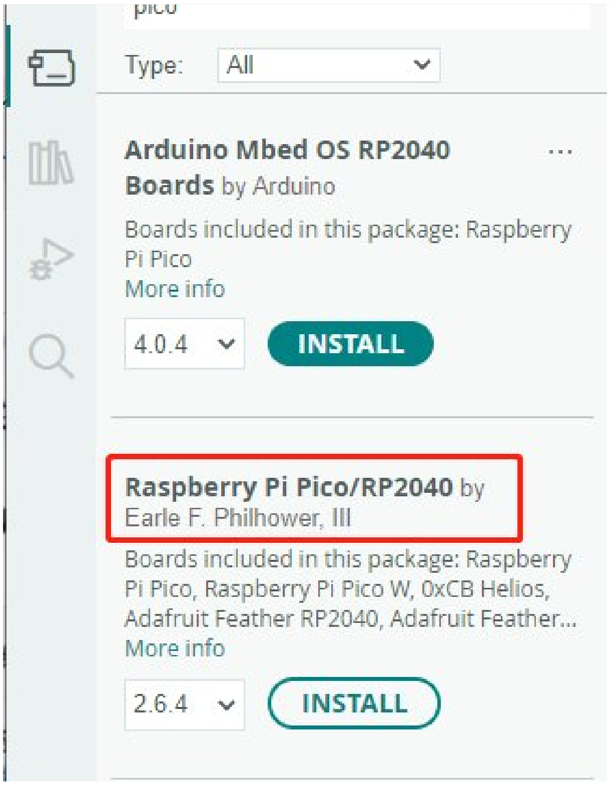
Note: If you already have the ESP8266 board URL, you can separate the URLs with commas like this:

```
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.


Tools
Help
Auto Format Ctrl+T
Archive Sketch
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter
WiFi101 / WiFiNINA Firmware Updater
Upload SSL Root Certificates
Board: "ESP32 Dev Module"
Port
Get Board Info
CPU Frequency: "240MHz (WiFi/BT)"
Core Debug Level: "None"
Flash Frequency: "80MHz"
Flash Mode: "QIO"
Flash Size: "4MB (32Mb)"
Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)"
PSRAM: "Disabled"
Upload Speed: "921600"
Burn Bootloader
Boards Manager... Ctrl+Shift+B
Arduino AVR Boards
esp32





pico

Type: All

Arduino Mbed OS RP2040 Boards by Arduino

Boards included in this package: Raspberry Pi Pico

More info

4.0.4 INSTALL

Raspberry Pi Pico/RP2040 by Earle F. Philhower, III

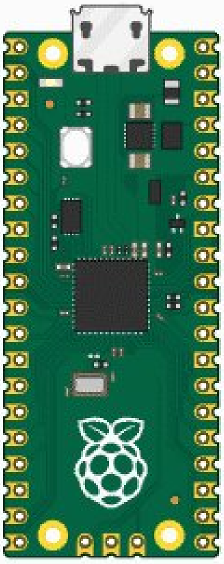
Boards included in this package: Raspberry Pi Pico, Raspberry Pi Pico W, 0xCB Helios, Adafruit Feather RP2040, Adafruit Feather...

More info

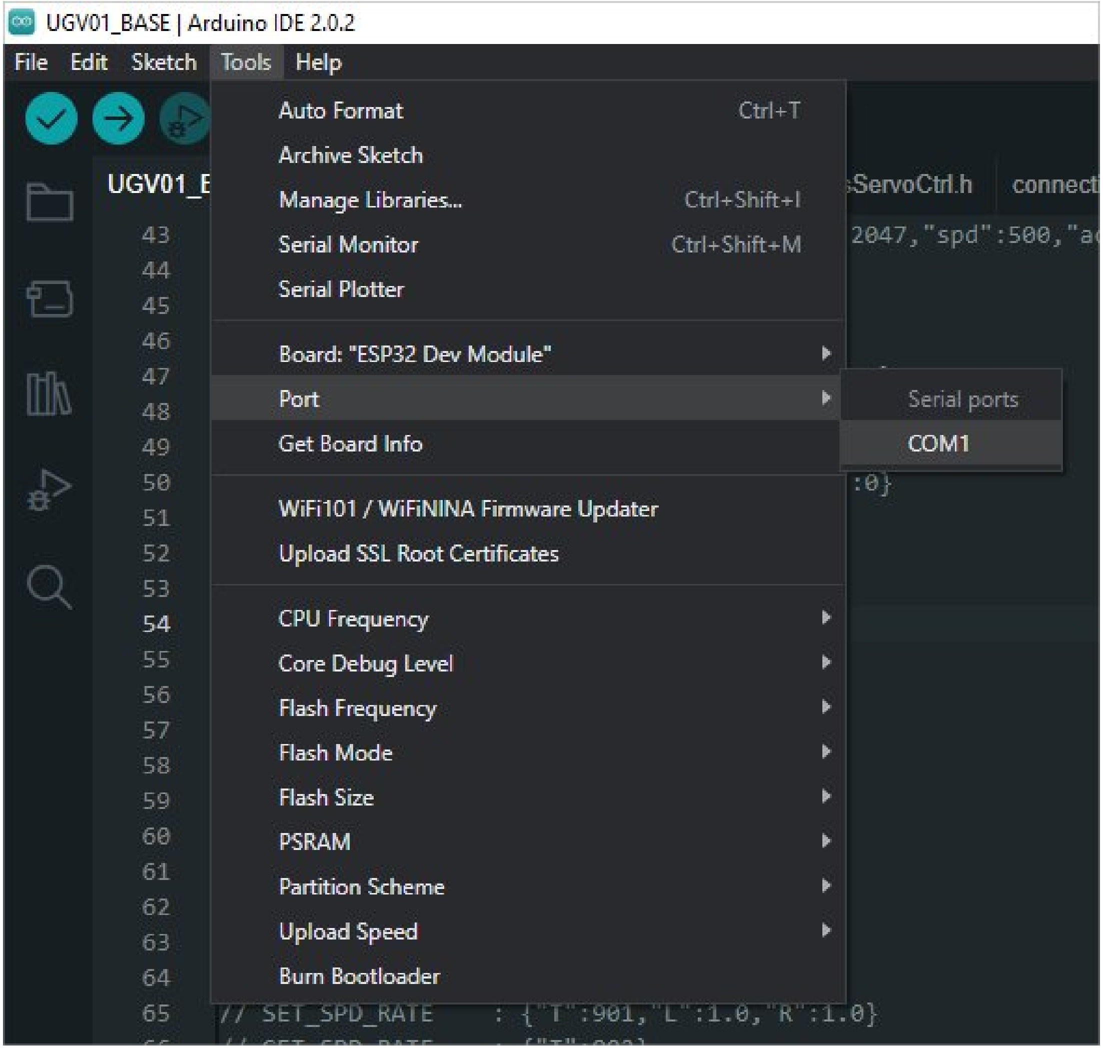
2.6.4 INSTALL

Upload Demo At the First Time

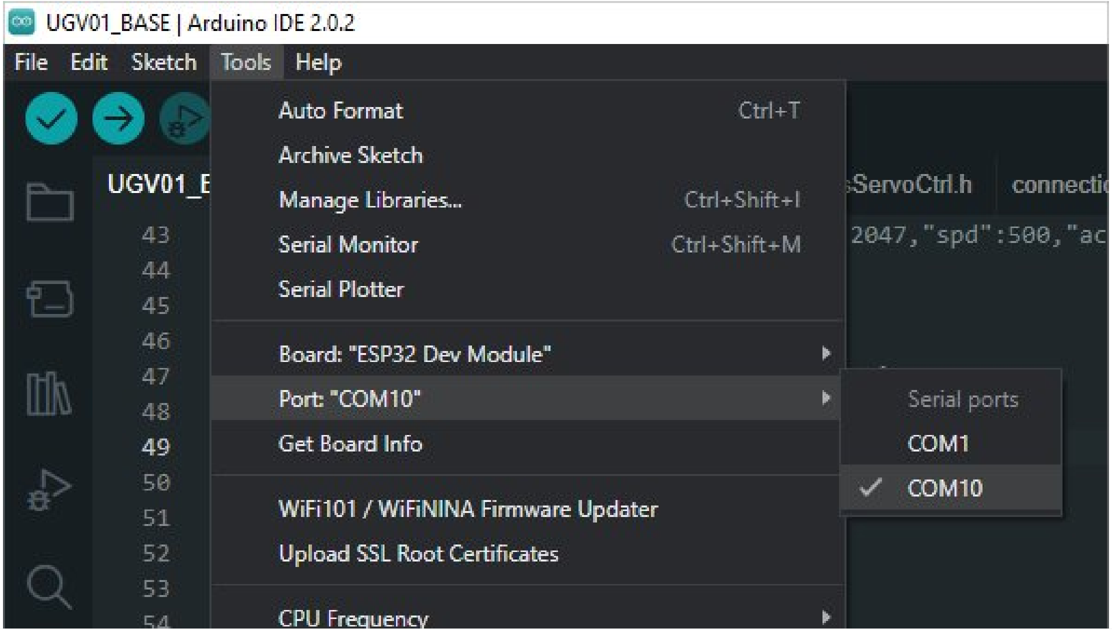
1. Press and hold the BOOTSET button on the Pico board, connect the Pico to the USB port of the computer via the Micro USB cable, and release the button when the computer recognizes a removable hard drive (RPI-RP2).

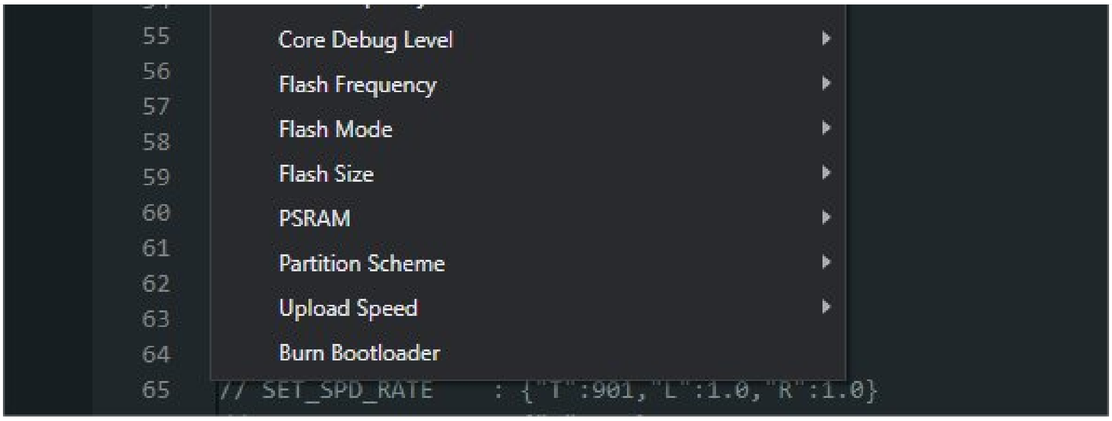


2. Download the demo, open arduino\PWM\D1-LED path under the D1-LED.ino.
3. Click Tools -> Port, remember the existing COM, do not need to click this COM (different computers show different COM, remember the existing COM on your computer).



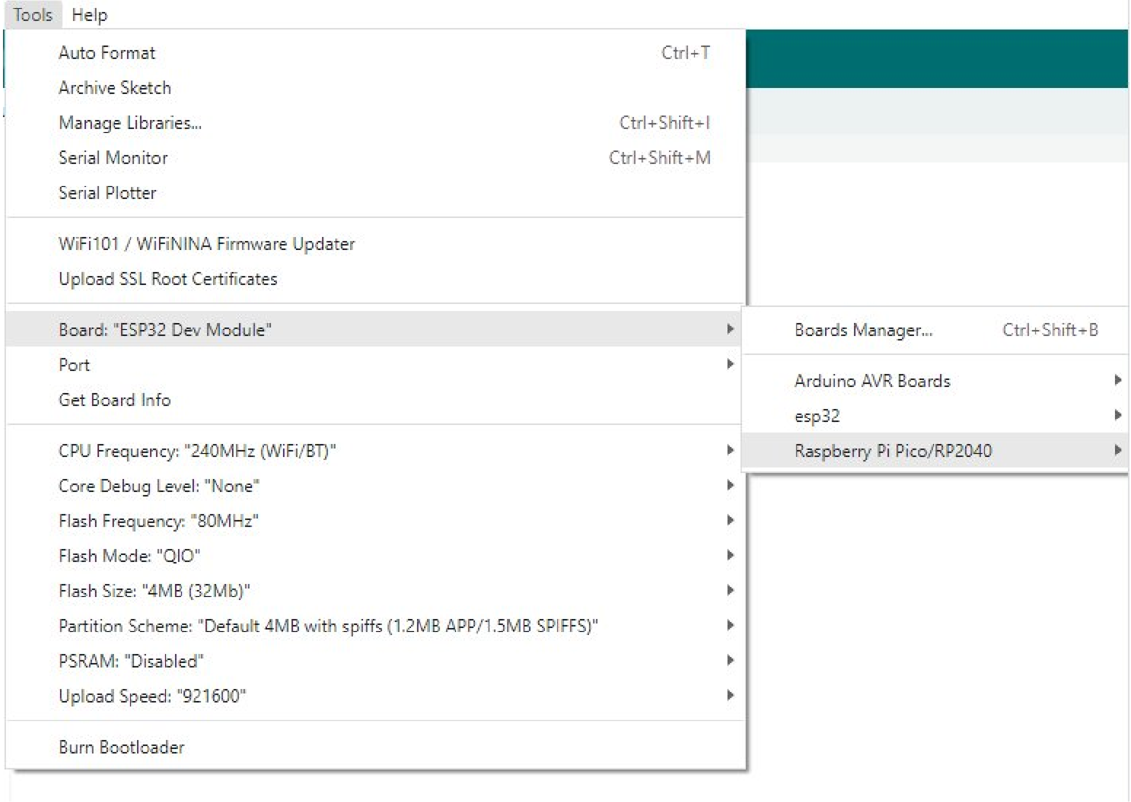
4. Connect the driver board to the computer with a USB cable, then click Tools -> Ports, select uf2 Board for the first connection, and after the upload is complete, connecting again will result in an additional COM port.


UGV01_BASE | Arduino IDE 2.0.2
File
Edit
Sketch
Tools
Help
UGV01_E
Auto Format Ctrl+T
Archive Sketch
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter
Board: "ESP32 Dev Module"
Port: "COM10"
Get Board Info
WiFi101 / WiFiNINA Firmware Updater
Upload SSL Root Certificates
CPU Frequency
Serial ports
COM1
✓ COM10
ServoCtrl.h
connecti
2047,"spd":500,"ac



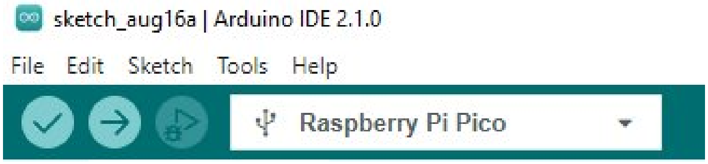
```
// SET_SPD_RATE    : {"T":901,"L":1.0,"R":1.0}
```

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


Tools
Help
Auto Format Ctrl+T
Archive Sketch
Manage Libraries... Ctrl+Shift+I
Serial Monitor Ctrl+Shift+M
Serial Plotter
WiFi101 / WiFiNINA Firmware Updater
Upload SSL Root Certificates
Port
Get Board Info
Burn Bootloader
Boards Manager... Ctrl+Shift+B
Arduino AVR Boards
esp32
Raspberry Pi Pico/RP2040

Board: "ESP32 Dev Module"
CPU Frequency: "240MHz (WiFi/BT)"
Core Debug Level: "None"
Flash Frequency: "80MHz"
Flash Mode: "QIO"
Flash Size: "4MB (32Mb)"
Partition Scheme: "Default 4MB with spiffs (1.2MB APP/1.5MB SPIFFS)"
PSRAM: "Disabled"
Upload Speed: "921600"

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

Resource

- Schematic
- Demo codes
- CH9121 Datasheets
- CH9121 AT Commands
- Software
- Description document shared by Adam
- Demo shared by Daniel Rust

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