

# 基于 LINUX 环境下 开发 ESP32 V1.0

（专业人员专属教程，初学者不建议）

—正点原子 ESP32 开发教程

修订历史:

版本	日期	修改内容
V1.0	2025/01/16	第一次发布



正点原子公司名称 : 广州市星翼电子科技有限公司  
原子哥在线教学平台 : [www.yuanzige.com](http://www.yuanzige.com)  
开源电子网 / 论坛 : [www.openedv.com](http://www.openedv.com)  
正点原子官方网站 : [www.alientek.com](http://www.alientek.com)  
正点原子淘宝店铺 : <https://openedv.taobao.com>  
正点原子 B 站视频 : <https://space.bilibili.com/394620890>

电话: 020-38271790 传真: 020-36773971

请下载原子哥 APP, 数千讲视频免费学习, 更快更流畅。  
请关注正点原子公众号, 资料发布更新我们会通知。



扫码下载“原子哥”APP



扫码关注正点原子公众号

系统的选择与环境搭建.....	1
一、WSL 环境搭建 .....	1
二、VSCode 环境搭建 .....	11
三、VSCode 连接 WSL 子系统 .....	15
四、怎么玩呢? .....	21
五、注意事项.....	28

# 系统的选择与环境搭建

在开发 ESP32 时，用户可以选择在 Windows、Linux 或 macOS 系统下进行开发。以往，正点原子提供的教程主要基于 Windows 系统。然而，由于硬件性能等因素限制，在 Windows 系统下编译单个例程通常需要较长时间，编译多个例程时甚至可能导致电脑卡顿甚至崩溃。

为了解决编译时间过长的问题，正点原子推出了基于 WSL2（Windows Subsystem for Linux 2）环境的编译教程。在 WSL2 环境下编译例程，所需时间相比纯 Windows 环境缩短了 10 倍以上。有人可能会问：为什么不采用 VMware + Ubuntu + VS Code + ESP-IDF 的开发方式？笔者也曾尝试过这种方法，虽然编译时间与 WSL2 开发环境相当，但在电脑性能不足的情况下，虚拟机环境容易导致系统卡顿。因此，笔者推荐使用 WSL2 + VS Code + ESP-IDF 进行开发。这种方式虽然存在一些不足，但至少能显著缩短编译时间，极大提升开发效率。

## 一、WSL 环境搭建

### 1. 检查虚拟化支持

按下 Ctrl + Alt + Del 快捷键打开任务管理器，接着进入“性能”选项卡，查看是否显示“虚拟化：已启用”。如果显示“已启用”，说明虚拟化功能已经成功开启。

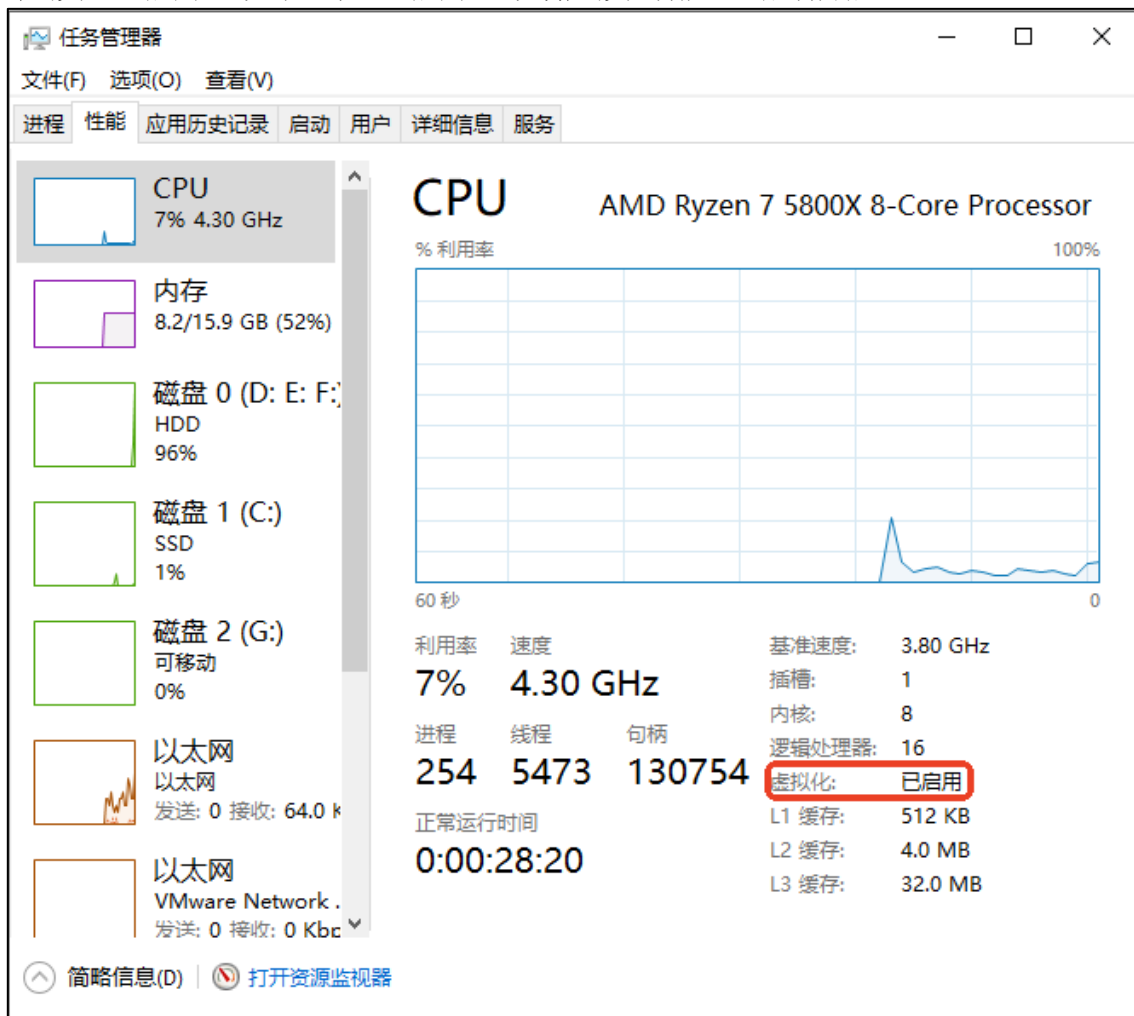


图 1.1 查看虚拟化是否启用

如果在任务管理器的“性能”选项卡中看到“虚拟化：已禁用”，说明您的 CPU 虚拟化功能未开启。此时，需要进入 BIOS 设置进行启用。以联想笔记本为例，操作步骤如下：

重启电脑并在开机时按 F2 进入 BIOS 设置界面。

① 在 BIOS 中，使用左右方向键选择 Configuration（配置）选项。

② 找到 Intel Virtual Technology（英特尔虚拟化技术），并确保其设置为 Enable（启用）。

注意：不同品牌的电脑进入 BIOS 的快捷键可能不同，请根据您的电脑型号查找相应的快捷键。

## 2，启用相关功能

打开“启用或关闭 Windows 功能”（控制面板\程序\程序和功能\启用或关闭 Windows 功能），勾选以下选项：

- ①：Hyper-V
- ②：适用于 Linux 的 Windows 子系统
- ③：虚拟机平台

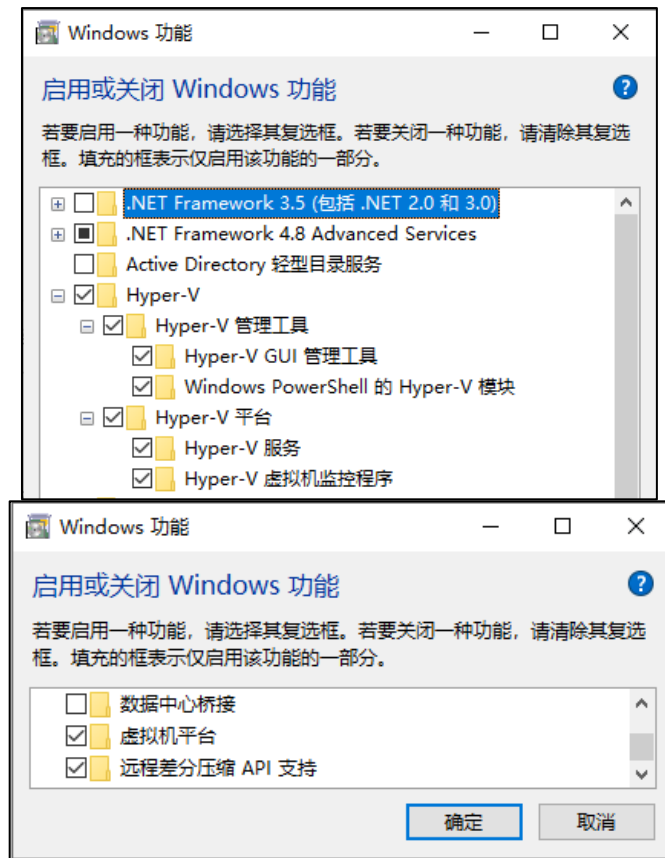


图 1.2 启动 Hyper-V、子系统和虚拟机平台

## 3，更新 WSL 版本

以管理员身份打开 CMD 或 Windows PowerShell 终端，输入以下命令。

```
wsl --update
```

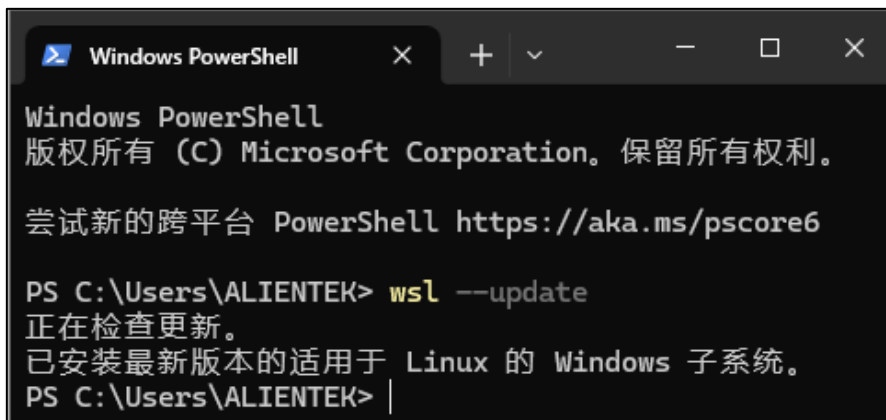


图 1.3 更新 WSL 版本

如果您的电脑已更新至最新版本，系统会提示“已安装最新版本的适用于 Linux 的 Windows 子系统”。如果不是最新版本，系统将显示下载进度条，直到 WSL 更新完成。

#### 4. 下载 WSL 子系统

下载方式两种，一种是使用命令下载，另一种是使用“Microsoft Store”微软商店下载，**读者可选其一即可。**

##### 1) 微软商店下载 WSL（建议下载 Ubuntu22.04 和 Ubuntu24.04 版本）

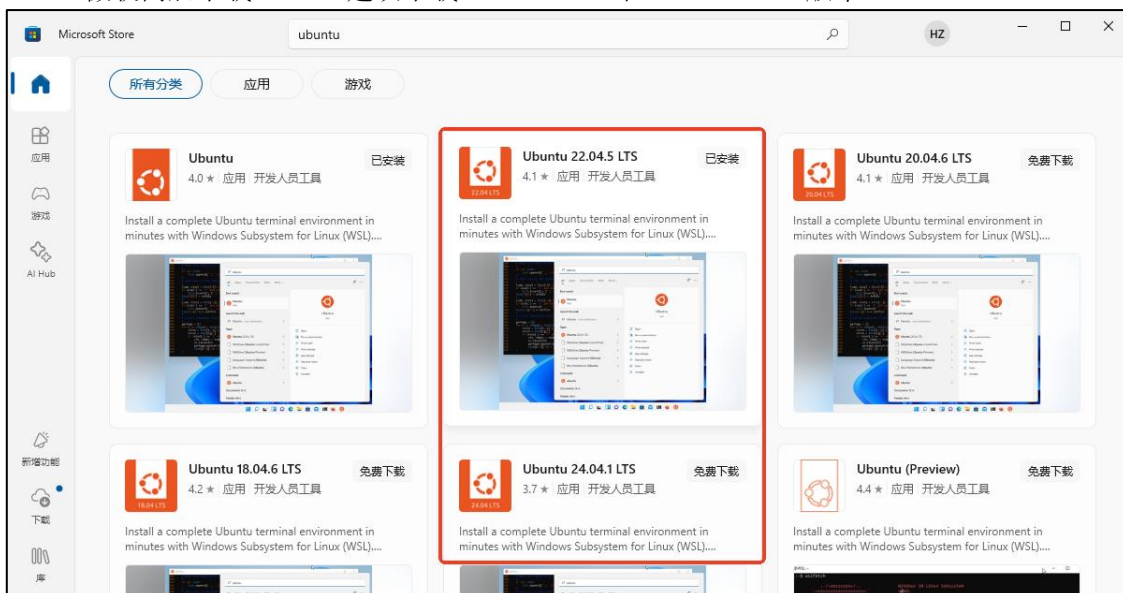
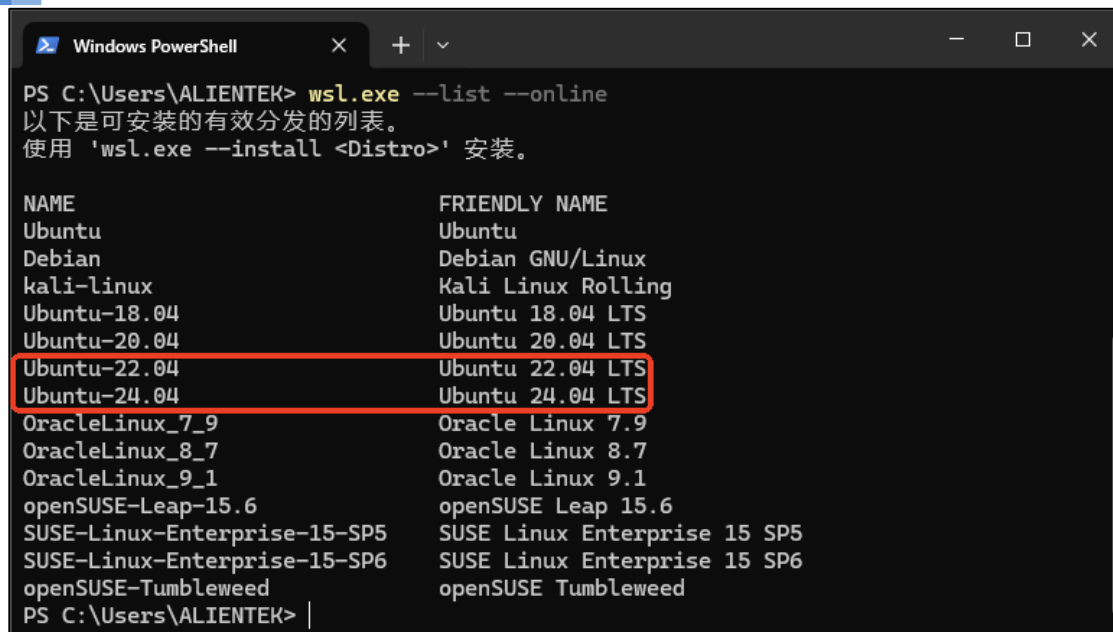


图 1.4 下载 WSL 子系统

这两个版本的子系统内置了 Python 3.10 及以上版本，而 ESP-IDF V5.3 及以上版本要求的 Python 版本为 3.9 或更高。因此，这些版本能够满足安装和使用 ESP-IDF 的需求，无需额外配置 Python 环境。

##### 2) 命令式下载（建议下载 Ubuntu22.04 和 Ubuntu24.04 版本）

①：在 Windows PowerShell 终端（以管理员身份打开）输入以下命令查看可用 linux 版本。  
`wsl.exe --list --online`

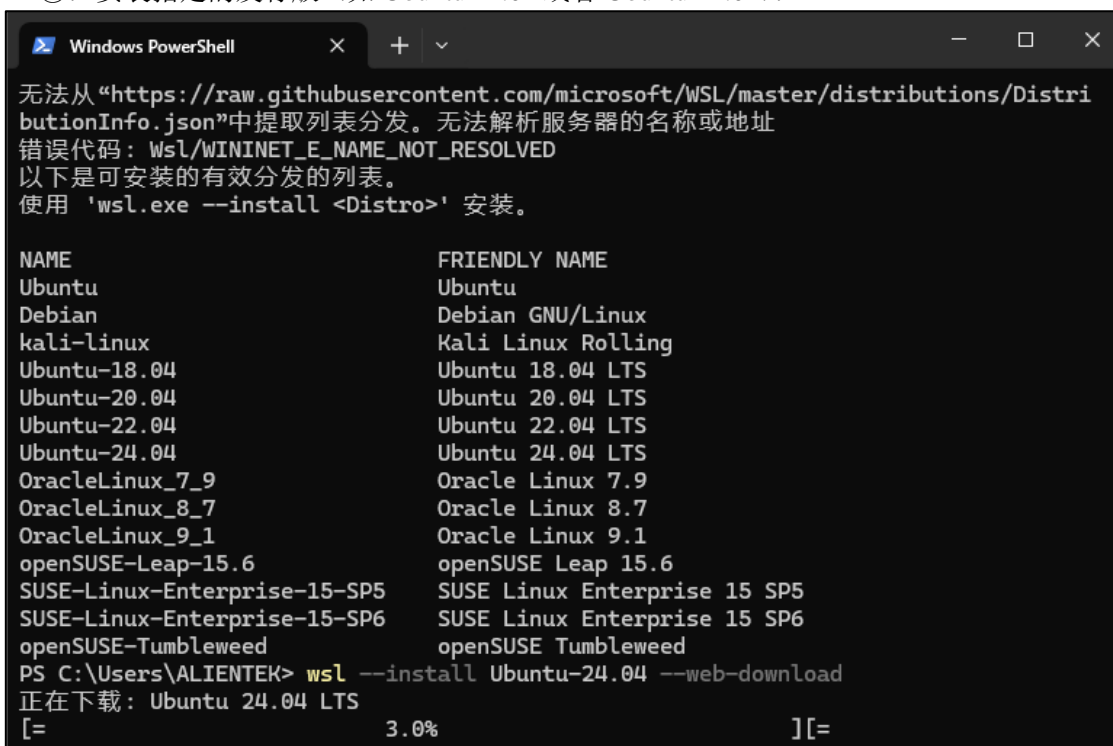


```
PS C:\Users\ALIENTEK> wsl.exe --list --online
以下是可安装的有效分发的列表。
使用 'wsl.exe --install <Distro>' 安装。

NAME                                FRIENDLY NAME
Ubuntu                              Ubuntu
Debian                              Debian GNU/Linux
kali-linux                          Kali Linux Rolling
Ubuntu-18.04                        Ubuntu 18.04 LTS
Ubuntu-20.04                        Ubuntu 20.04 LTS
Ubuntu-22.04                        Ubuntu 22.04 LTS
Ubuntu-24.04                        Ubuntu 24.04 LTS
OracleLinux_7_9                     Oracle Linux 7.9
OracleLinux_8_7                     Oracle Linux 8.7
OracleLinux_9_1                     Oracle Linux 9.1
openSUSE-Leap-15.6                  openSUSE Leap 15.6
SUSE-Linux-Enterprise-15-SP5        SUSE Linux Enterprise 15 SP5
SUSE-Linux-Enterprise-15-SP6        SUSE Linux Enterprise 15 SP6
openSUSE-Tumbleweed                 openSUSE Tumbleweed
PS C:\Users\ALIENTEK> |
```

图 1.5 查看 linux 发行版本

②：安装指定的发行版（如 Ubuntu 22.04 或者 Ubuntu 24.04）：



```
无法从“https://raw.githubusercontent.com/microsoft/WSL/master/distributions/DistributionInfo.json”中提取列表分发。无法解析服务器的名称或地址
错误代码：WSL/WININET_E_NAME_NOT_RESOLVED
以下是可安装的有效分发的列表。
使用 'wsl.exe --install <Distro>' 安装。

NAME                                FRIENDLY NAME
Ubuntu                              Ubuntu
Debian                              Debian GNU/Linux
kali-linux                          Kali Linux Rolling
Ubuntu-18.04                        Ubuntu 18.04 LTS
Ubuntu-20.04                        Ubuntu 20.04 LTS
Ubuntu-22.04                        Ubuntu 22.04 LTS
Ubuntu-24.04                        Ubuntu 24.04 LTS
OracleLinux_7_9                     Oracle Linux 7.9
OracleLinux_8_7                     Oracle Linux 8.7
OracleLinux_9_1                     Oracle Linux 9.1
openSUSE-Leap-15.6                  openSUSE Leap 15.6
SUSE-Linux-Enterprise-15-SP5        SUSE Linux Enterprise 15 SP5
SUSE-Linux-Enterprise-15-SP6        SUSE Linux Enterprise 15 SP6
openSUSE-Tumbleweed                 openSUSE Tumbleweed
PS C:\Users\ALIENTEK> wsl --install Ubuntu-24.04 --web-download
正在下载：Ubuntu 24.04 LTS
[=                                     3.0%                               ][=
```

图 1.6 下载 Ubuntu24.04

3) 打开与配置 WSL

安装完成后，打开 WSL。首次运行时，系统会提示正在进行初始化配置。配置完成后，按照提示输入用户名并设置密码。完成这些步骤后，您将成功进入 Ubuntu 24.04.2 LTS 系统。



```
xiaofeng@DESKTOP-PELSFB8: X + v
Enter new UNIX username: xiaofeng
New password:
Retype new password:
passwd: password updated successfully
Installation successful!
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

Welcome to Ubuntu 24.04 LTS (GNU/Linux 5.15.167.4-microsoft-standard-WSL2 x86_64)

* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:       https://ubuntu.com/pro

System information as of Wed Jan 15 10:21:52 CST 2025

System load:  0.92          Processes:            55
Usage of /:   0.1% of 1006.85GB Users logged in:      0
Memory usage: 9%           IPv4 address for eth0: 172.22.193.129
Swap usage:   0%

This message is shown once a day. To disable it please create the
/home/xiaofeng/.hushlogin file.
xiaofeng@DESKTOP-PELSFB8:~$
```

图 1.5 配置用户名与密码

若打开 Ubuntu-24.04.2 时候，系统提示以下错误：

```
Installing, this may take a few minutes...
WslRegisterDistribution failed with error: 0x80071772
Error: 0x80071772 ??????????????????????,
```

出现上述错误的原因通常是 WSL 默认版本设置问题。一般情况下，我们下载的 WSL 版本为 1，但如果系统提示该错误，说明您的电脑当前的 WSL 默认版本为 2。我们可在 Windows PowerShell 终端（以管理员身份打开）输入以下命令。

```
wsl --set-default-version 1
```

此时重新打开 Ubuntu-24.04.2 子系统就不会出现这个错误了，然后重新操作上述 3) 步骤。

## 5, WSL 子系统搬迁

子系统搬迁有两种方式，一种是从 C 盘搬迁至其他盘，另一种是直接存储在其他盘。

### 5.1 从 C 盘搬迁至其他盘

Windows Store 微软商城或者命令式安装的 WSL 都默认被安装在 C 盘，然而 C 盘是我们电脑重要的磁盘，如果 C 盘空间满了，会影响电脑运行。因此，我们最好将子系统搬离 C 盘，也就是说，不使用 C 盘来存放文件。在这里，作者将子系统搬到 E 盘存储。

WSL 子系统搬迁流程，如下所示：

1) 查询 WSL 系统状态。使用管理员身份打开 Windows PowerShell 命令行，并在命令行输入“wsl -l -v”查看 wsl 虚拟机的名称和状态，如下图所示。

```
PS C:\Users\ALIEN TEK> wsl -l -v
NAME                STATE
* Ubuntu            Stopped
Ubuntu-20.04        Stopped
Ubuntu-22.04        Running
Ubuntu-24.04        Stopped
PS C:\Users\ALIEN TEK> |
```

图 1.6 wsl 虚拟机的名称与状态

从上图可知，作者安装了四个 WSL Ubuntu 子系统，它们分别为 Ubuntu、Ubuntu-20.04、Ubuntu-22.04、Ubuntu-24.04，目前它们的状态分别为停止状态、停止状态、运行状态和停止状态。下面，作者将以 Ubuntu-24.04 为例，说明如何将其搬迁到 E 盘。

2) 停止 WSL 子系统。输入 `wsl --shutdown` 使 Ubuntu-24.04 停止运行，再次使用 `wsl -l -v` 确保其处于 `stopped` 状态，如下图所示。

```
PS C:\Users\ALIENITEK> wsl -l -v
NAME                                STATE
* Ubuntu                            Stopped
Ubuntu-20.04                        Stopped
Ubuntu-22.04                        Stopped
Ubuntu-24.04                        Stopped
PS C:\Users\ALIENITEK> |
```

图 1.7 Ubuntu-22.04 停止运行

3) 导出/恢复备份。在 E 盘创建一个文件夹用来存放新的 WSL，比如作者创建了一个 `E:\Ubuntu\Ubuntu_24.04.1` 文件夹。

①：导出它的备份（比如命名为 `Ubuntu.tar`）。在 CMD 命令行下输入“`wsl --export Ubuntu-24.04 E:\Ubuntu\Ubuntu_24_WSL\Ubuntu.tar`”。

②：确定在 `E:\Ubuntu\Ubuntu_24_WSL` 下是否可看见备份 `Ubuntu.tar` 文件，之后注销原有的 WSL。在 CMD 命令行输入“`wsl --unregister Ubuntu-24.04`”注销原本的 WSL。

③：将备份文件恢复到 `E:\Ubuntu\Ubuntu_24_WSL` 中去。在 CMD 命令行下输入“`wsl --import Ubuntu-24.04 E:\Ubuntu\Ubuntu_24_WSL E:\Ubuntu\Ubuntu_24_WSL\Ubuntu.tar`”恢复备份文件。

这时候启动 WSL，发现好像已经恢复正常了，但是用户变成了 `root`。

## 5.2 直接存储在其他盘

打开“设置”→“系统”→“存储”→“更改新内容的保存位置”，如下图所示。

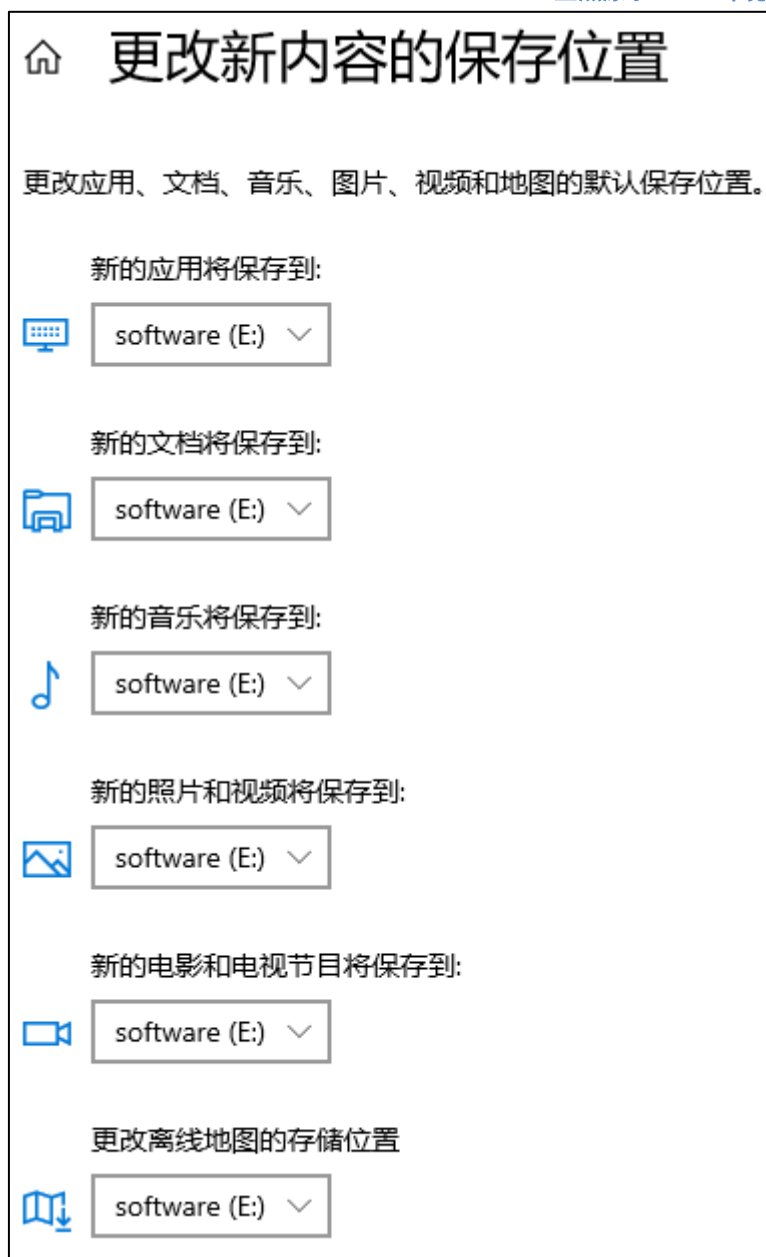


图 1.8 将存储位置为改为 E 盘

此时 E 盘出现了 WindowsApps 文件夹，此文件夹就是用于存储上图的内容。这样我们不需要将子系统从 C 盘搬迁至其他盘了。

#### 6. 安装 Linux 内核更新包（若想把 WSL1 更新至 WSL2 必须安装 linux 内核更新包）

该更新包下载地址为：[https://wslstorestorage.blob.core.windows.net/wslblob/wsl\\_update\\_x64.msi](https://wslstorestorage.blob.core.windows.net/wslblob/wsl_update_x64.msi)，若找不到该更新包，可在资料 A 盘\6，软件资料\1，软件\1，IDF 开发工具目录下找到 wsl\_update\_x64.msi 文件，此时双击打开进行更新。

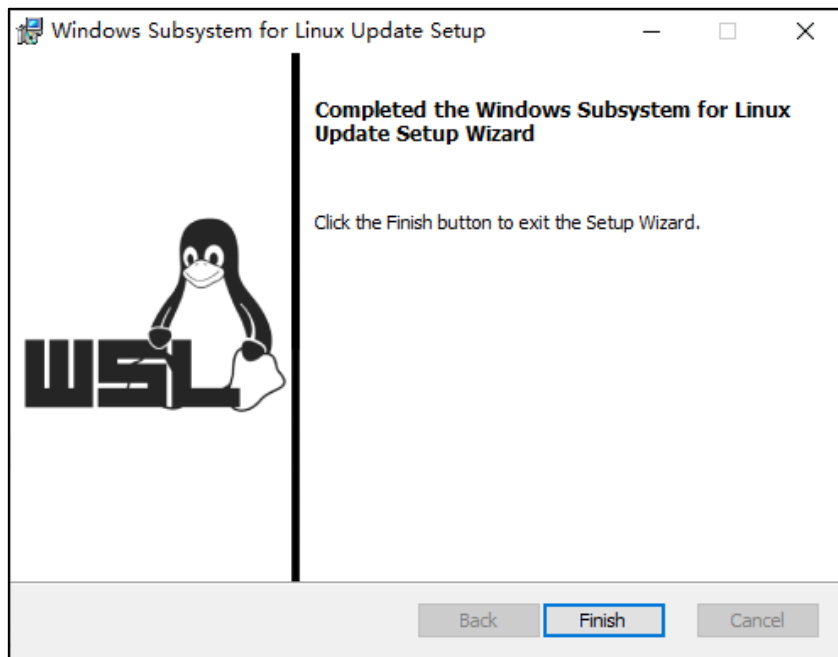


图 1.9 更新 linux 内核

双击后，自动更新完成，此时我们点击上图中的“Finish”选项即可。

## 7. 将 Ubuntu24.04.2 TLS 升级至 WSL2 版本

1) Windows PowerShell 终端（以管理员身份打开）输入“wsl --set-default-version 2”。

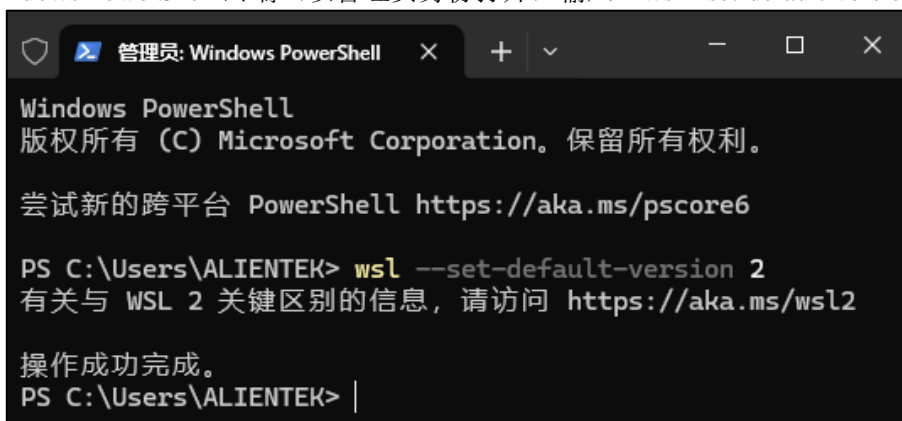


图 1.10 将 WSL 更新为 2 版本

2) 将 Ubuntu24.04.2 版本升级到 WSL2，输入“wsl --set-version Ubuntu-24.04 2”。

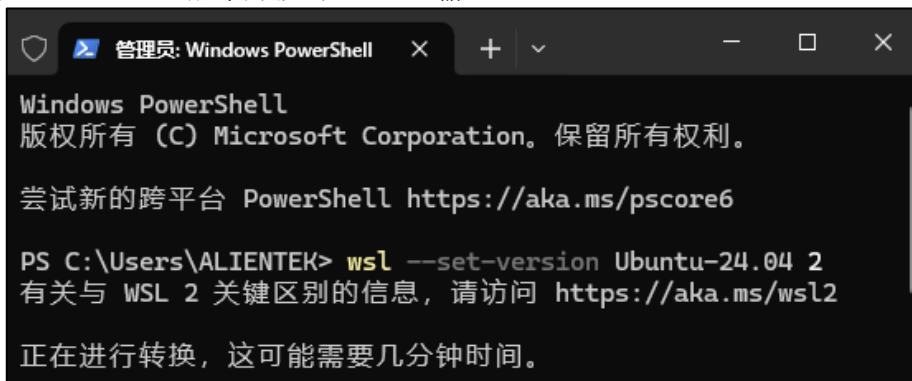


图 1.11 将 Ubuntu24.04.2 版本升级到 WSL2

直到提示“转换成功”。

## 8. 安装 USB 支持工具

它的核心是通过网络共享 USB 设备，使远程系统能够访问本地 USB 设备。简单来说，在使用 WSL2 下载代码时，可以通过 USB 设备实现网络共享，从而让 WSL 子系统直接使用 USB 设备进行代码下载，提升操作便利性。

USB 支持工具下载地址为 <https://github.com/dorssel/usbipd-win/releases>，若下载不成功，可通过资料 A 盘\6，软件资料\1，软件\1，IDF 开发工具目录下找到 usbipd-win\_4.3.0.msi 文件，此时双击打开进行安装。

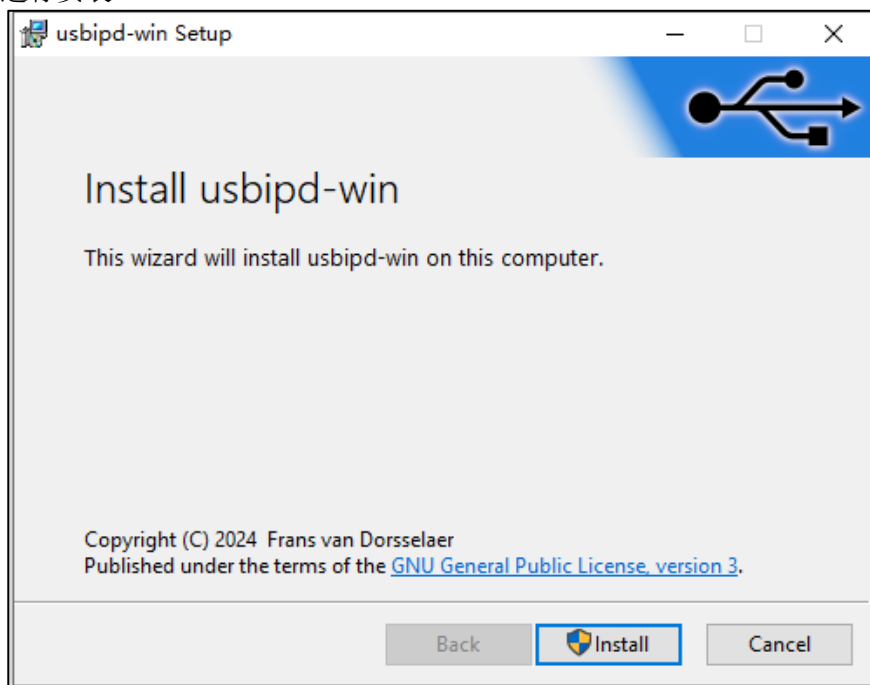


图 1.12 安装 USB 支持工具

点击上图中的“Install”安装 USB 支持工具，如下图所示。

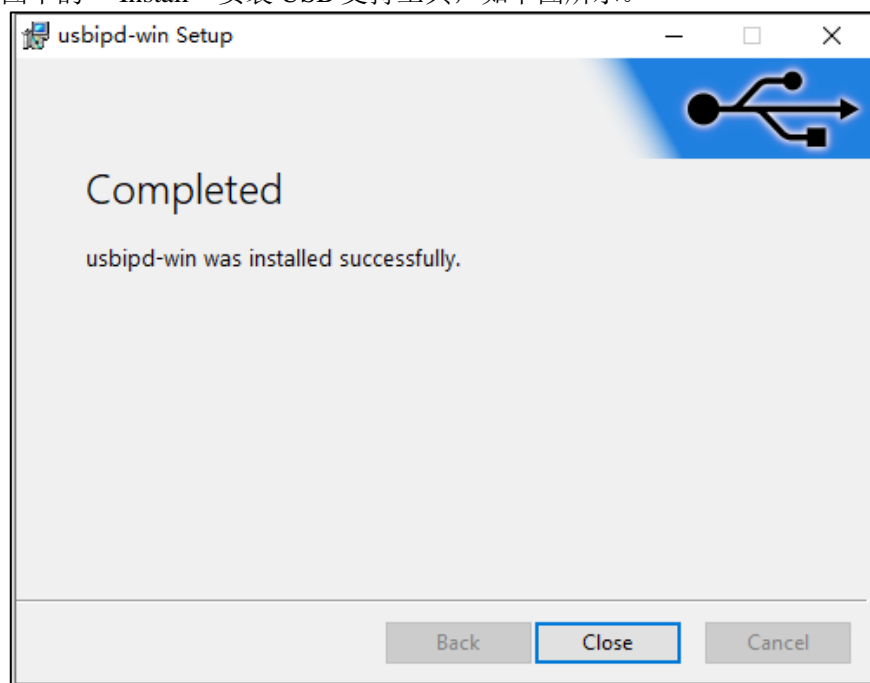


图 1.13 安装 USB 支持工具成功

我们打开 PowerShell 终端，并使用数据线将 PC 机与开发板相连，然后在终端输入“usbipd list”命令，查看 USB 挂载设备，如下图所示。

```

管理员: Windows PowerShell
PS C:\Users\ALIENTEK> usbipd list
Connected:
BUSID  VID:PID  DEVICE                                STATE
1-5    04d8:00df  USB 串行设备 (COM28), USB 输入设备    Not shared
1-6    1a2c:40e2  USB 输入设备                          Not shared
1-9    05e3:0749  USB 大容量存储设备                  Not shared
1-11   1a86:7523  USB-SERIAL CH340 (COM24)             Not shared
1-12   2717:5013  USB 输入设备                          Not shared
1-14   048d:5702  USB 输入设备                          Not shared

Persisted:
GUID                                DEVICE

usbipd: warning: USB filter 'USBPcap' is known to be incompatible with this software; 'bind --force' will be required.
PS C:\Users\ALIENTEK>
  
```

图 1.14 查看挂载的 USB 设备

可以看到，1-11 BUSID 为我们开发板的 USB 设备，即 USB UART 设备，此时它的状态为 Not shared（不共享）。然后通过“usbipd bind -b 1-11”进行绑定 USB 设备，如下图所示。

```

管理员: Windows PowerShell
PS C:\Users\ALIENTEK> usbipd bind -b 1-11
usbipd: info: Device with busid '1-11' was already shared.
usbipd: warning: USB filter 'USBPcap' is known to be incompatible with this software; 'bind --force' will be required.
PS C:\Users\ALIENTEK> usbipd list
Connected:
BUSID  VID:PID  DEVICE                                STATE
1-5    04d8:00df  USB 串行设备 (COM28), USB 输入设备    Not shared
1-6    1a2c:40e2  USB 输入设备                          Not shared
1-9    05e3:0749  USB 大容量存储设备                  Not shared
1-11   1a86:7523  USB-SERIAL CH340 (COM24)             Shared
1-12   2717:5013  USB 输入设备                          Not shared
1-14   048d:5702  USB 输入设备                          Not shared

Persisted:
GUID                                DEVICE

usbipd: warning: USB filter 'USBPcap' is known to be incompatible with this software; 'bind --force' will be required.
PS C:\Users\ALIENTEK>
  
```

图 1.15 绑定 USB 设备

经过上述绑定后，此时 USB UART 设备处于共享状态。到了这里，WSL2 环境搭建完成。

## 二、VSCode 环境搭建

鉴于我们使用的是 VS Code IDE 搭配乐鑫 ESP-IDF 进行开发，接下来将介绍 VS Code 的安装过程。首先，访问 VS Code 的官方（<https://code.visualstudio.com/Download>）下载页面，根据系统需求选择相应的安装包进行下载。此外，也可以在 A 盘→6，软件资料→1，软件→1，IDF 开发工具路径下找到 VS Code 的安装包。下图为 VS Code 的官方下载页面。

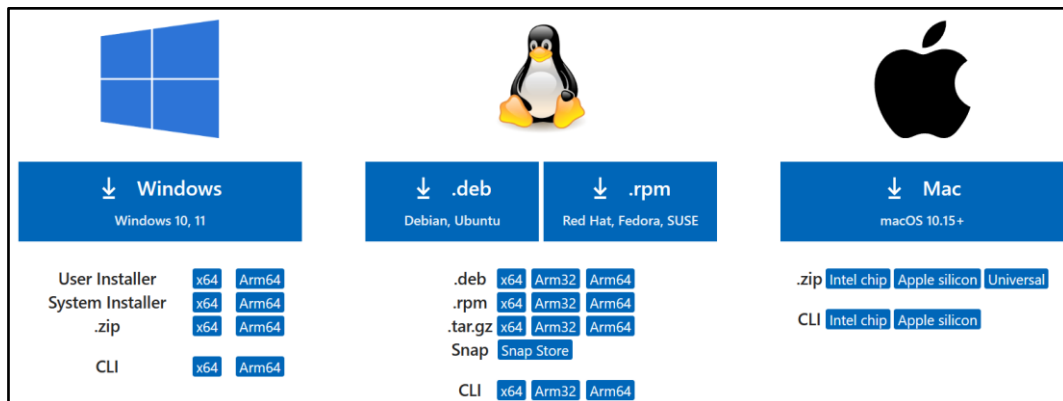


图 2.1 VS Code 官网下载界面

在此，我们选择下载 Windows 版本的 VS Code，因为我们在 Windows 环境下进行开发。虽然其他版本的下载方式应该类似，但这里我们将重点介绍 Windows 版本的下载步骤。下载完成后，按照以下步骤进行安装即可：

- 1，以<管理员身份>运行 VSCodeUserSetup-x64-1.94.2.exe 安装包，如下图所示。

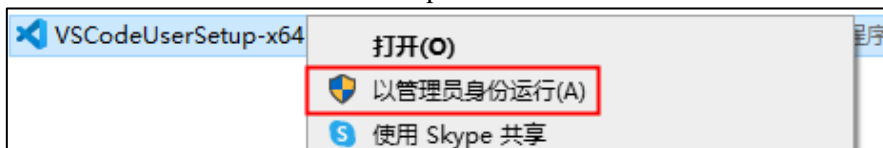


图 2.2 以管理员身份运行 VSCode 安装包

- 2，此时，进入安装的许可协议页面，如下图所示。

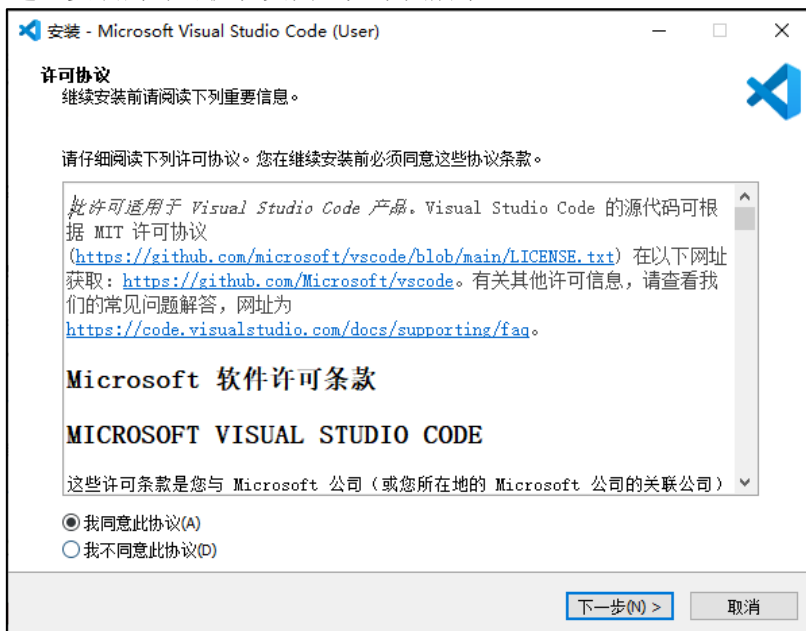


图 2.3 安装许可协议

- 3，选择“我同意此协议”，然后点击“下一步”进入选择目标位置页面，如下图所示。

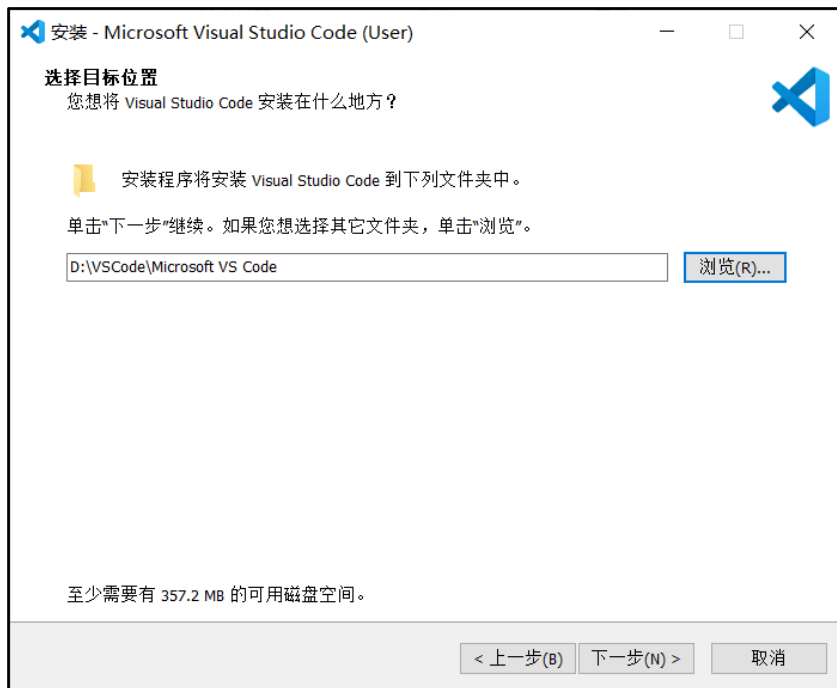


图 2.4 设置安装路径

用户可以自行填写安装位置，但需注意，安装路径最好使用全英文字符，以避免可能出现的路径识别问题。

4，设置安装路径完成后，可点击“下一步”进入选择开始菜单文件夹，如下图所示。

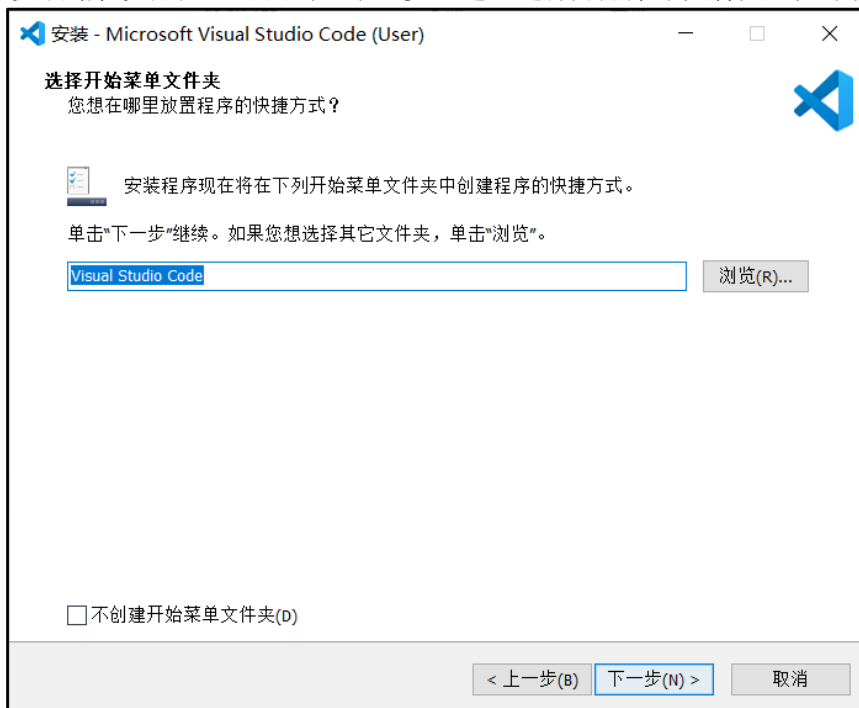


图 2.5 选择开始菜单文件夹

5，这里我们保持默认配置，接着，点击“下一步”进入选择附加任务页面，如下图所示。



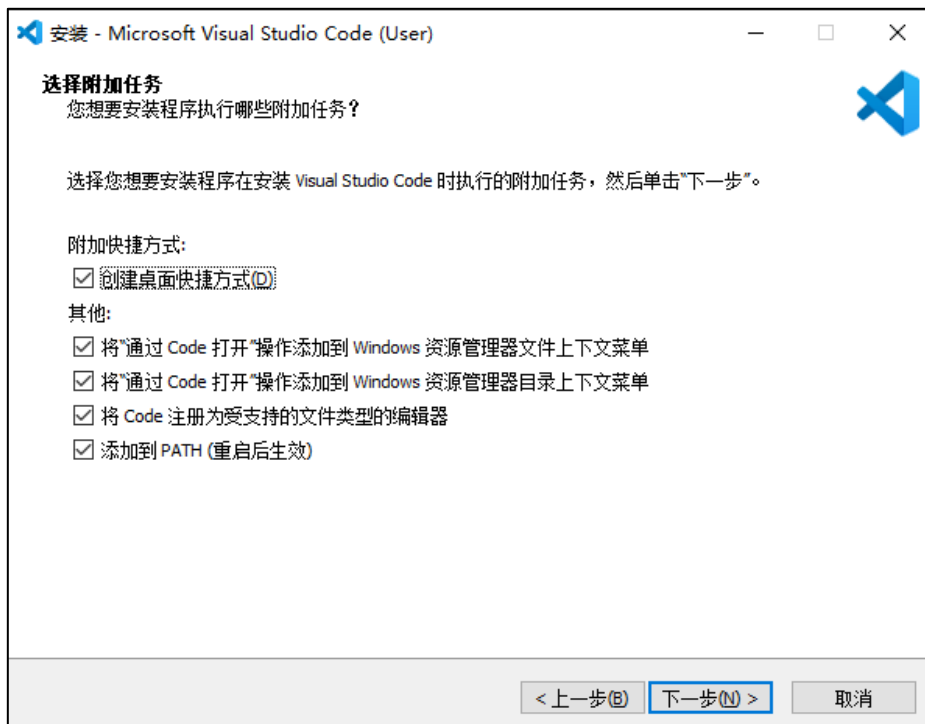


图 2.6 选择附加任务

6, 请把附加任务全部勾选, 然后点击“下一步”进入准备安装页面, 如下图所示。

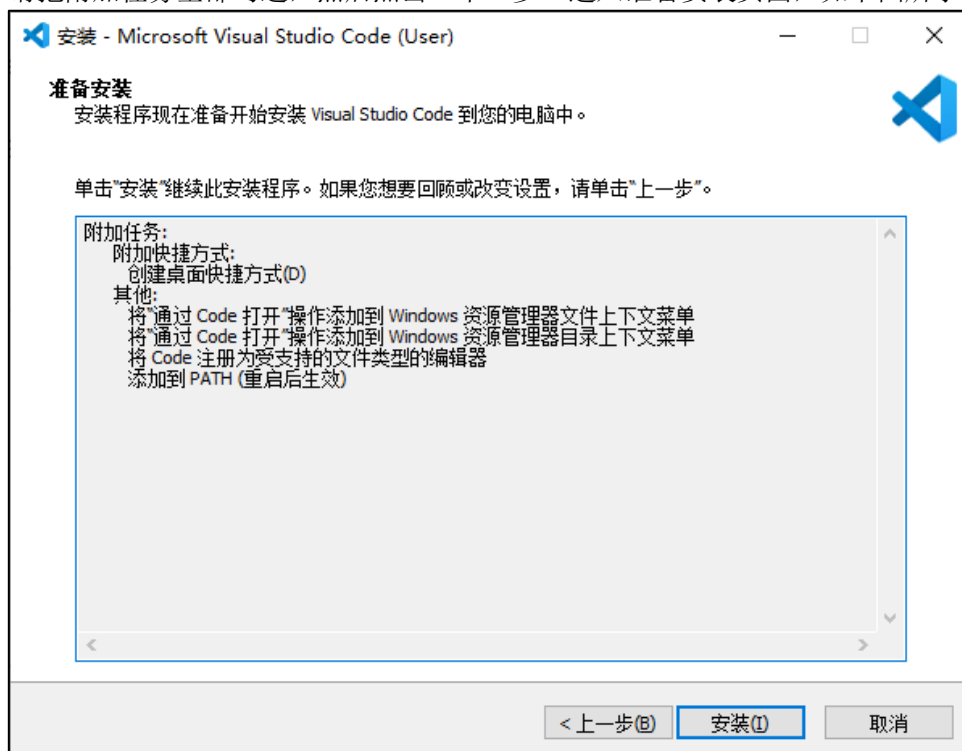


图 2.7 准备安装页面

7, 点击“安装”按钮即可开始安装 VS Code 集成开发环境。安装成功后, 界面将如图所示。



图 2.8 VSCode 安装成功

值得注意的是，第一次安装的 VS Code 软件为英文版本。如果读者希望将 VS Code 设置为中文汉化，则需要安装简体中文插件。安装方式如下。

在 VS Code 集成开发环境中，可以通过点击左侧的“扩展”图标或使用快捷键“Ctrl + Shift + X”直接进入扩展页面。在搜索栏中输入“Chinese”，然后选择安装“Chinese (Simplified) Language Pack for Visual Studio Code”插件，如下图所示。安装完成后，必须重启 VS Code 才能切换为简体中文界面。

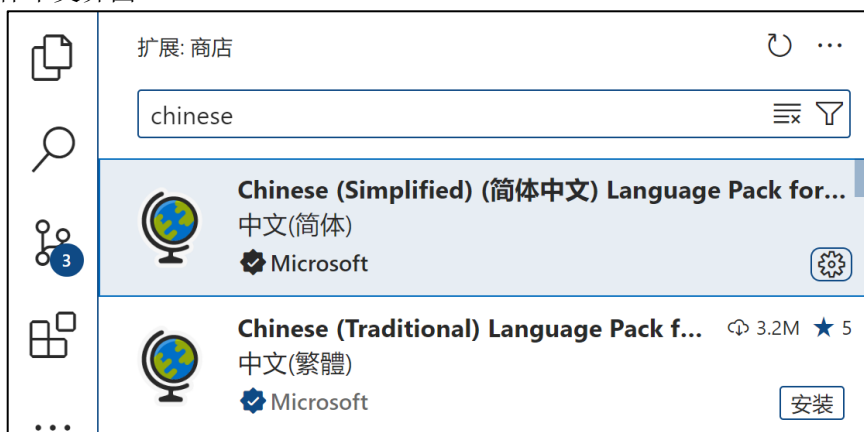


图 2.9 安装简体中文插件

### 三、VSCode 连接 WSL 子系统

#### 1, 安装 WSL 插件

通过安装 WSL 插件，Windows 系统可以与 Ubuntu 子系统无缝连接，实现以下功能：

- 1) 编写代码
- 2) 编译代码
- 3) 下载代码
- 4) 调试代码

在 VSCode 的扩展

在 VS Code 集成开发环境中，可以通过点击左侧的“扩展”图标或使用快捷键“Ctrl + Shift + X”直接进入扩展页面。在搜索栏中输入“WSL”，然后安装“WSL”插件，如下图所示。

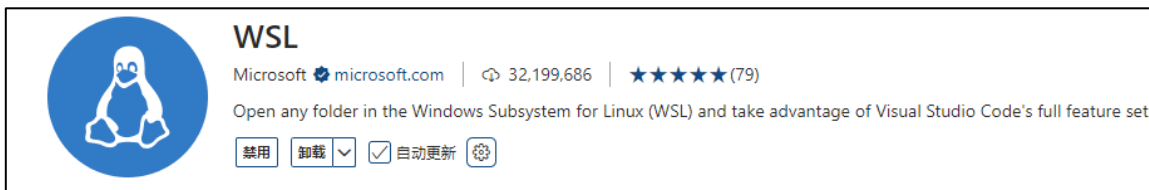


图 2.10 下载 WSL 插件

#### 2, 连接 WSL

按下快捷键“F1”或“Ctrl + Shift + P”打开“显示所有命令”界面。然后，在搜索框中输入“WSL:连接到 WSL”，并从下拉菜单中选择此选项，如下图所示。

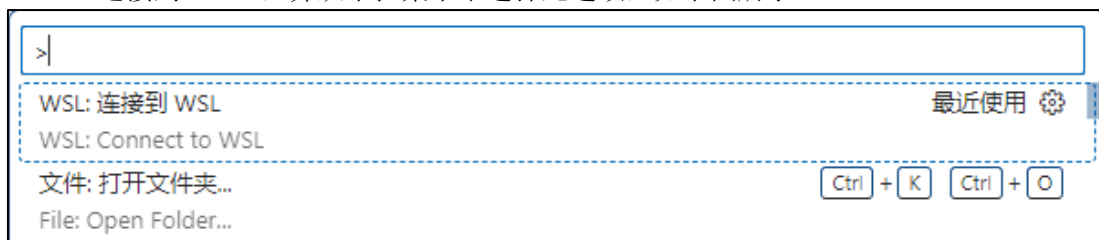


图 2.11 连接到 WSL

至此，我们进入到 VSCode 的 Ubuntu-24.04 子系统，这样从 VSCode 安装的插件都归 Ubuntu-24.04 子系统所有，如下图所示。

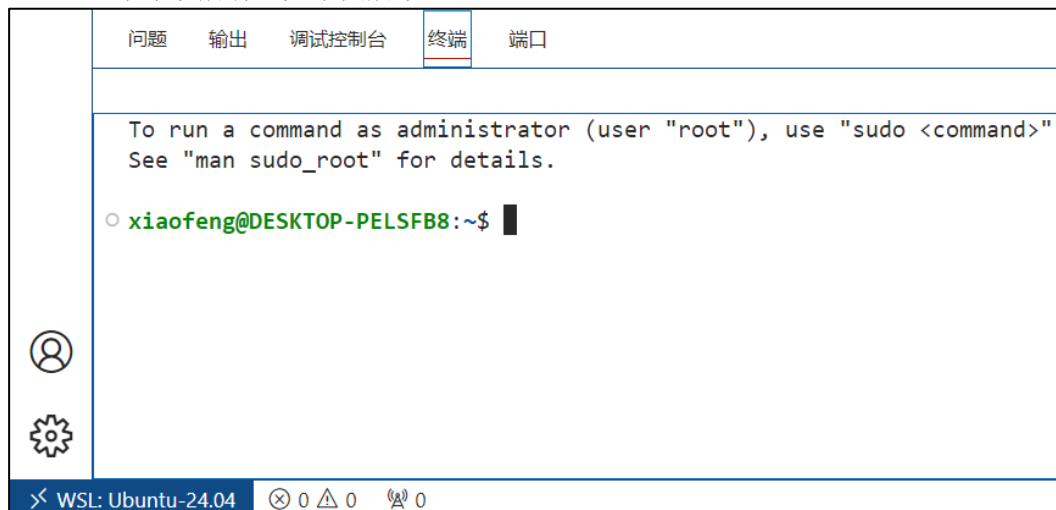


图 2.12 连接 Ubuntu-24.04

在上图终端下输入“sudo apt-get update”命令，更新系统。更新系统成功后，我们安装 ESP-IDF 所需要的依赖库，如下命令所示。

```
sudo apt-get install git wget flex bison gperf python3 \
python3-pip python3-venv cmake ninja-build ccache \
```

```
libffi-dev libssl-dev dfu-util libusb-1.0-0
```

**注意：**必须确认当前的 WSL 默认 Python 版本为 3.9 以上，可使用“python3 -V”命令查询。

### 3, 克隆 ESP-IDF 软件开发库

在~\$目录下输入“mkdir esp”命令新建 esp 文件夹，然后输入“cd esp”命令进入 esp 文件夹，最后输入“git clone <https://gitee.com/EspressifSystems/esp-idf.git>”命令克隆 ESP-IDF mater 主分支源码库，如下图所示。

```
● xiaofeng@DESKTOP-PELSFB8:~$ mkdir esp
● xiaofeng@DESKTOP-PELSFB8:~$ cd esp/
● xiaofeng@DESKTOP-PELSFB8:~/esp$ git clone https://gitee.com/EspressifSystems/esp-idf.git
Cloning into 'esp-idf'...
remote: Enumerating objects: 710053, done.
remote: Counting objects: 100% (20163/20163), done.
remote: Compressing objects: 100% (8155/8155), done.
remote: Total 710053 (delta 13046), reused 16426 (delta 11189), pack-reused 689890
Receiving objects: 100% (710053/710053), 266.85 MiB | 2.06 MiB/s, done.
Resolving deltas: 100% (541272/541272), done.
Updating files: 100% (16495/16495), done.
```

图 2.13 克隆 ESP-IDF

### 4, 克隆 ESP-IDF TOOLS 工具库

进入 esp/esp-idf 目录，然后在此路径下输入“git clone <https://gitee.com/EspressifSystems/esp-gitee-tools.git>”克隆 ESP-IDF TOOLS 工具，如下图所示。

```
● xiaofeng@DESKTOP-PELSFB8:~/esp$ cd esp-idf/
● xiaofeng@DESKTOP-PELSFB8:~/esp/esp-idf$ git clone https://gitee.com/EspressifSystems/esp-gitee-tools.git
Cloning into 'esp-gitee-tools'...
remote: Enumerating objects: 106, done.
remote: Total 106 (delta 0), reused 0 (delta 0), pack-reused 106
Receiving objects: 100% (106/106), 36.46 KiB | 276.00 KiB/s, done.
Resolving deltas: 100% (56/56), done.
```

图 2.14 克隆 ESP-IDF TOOLS 工具

### 5, 更新 ESP-IDF 子模块

在 esp/esp-idf 目录下，输入“./esp-gitee-tools/submodule-update.sh”命令更新子模块，如下图所示。

```
● xiaofeng@DESKTOP-PELSFB8:~/esp/esp-idf$ ./esp-gitee-tools/submodule-update.sh
Submodule 'components/bootloader/subproject/components/micro-ecc/micro-ecc' (https://gitee.com/kmackay/micro-ecc.git) registered for path 'components/bootloader/subproject/components/micro-ecc/micro-ecc'
Submodule 'components/bt/controller/lib_esp32' (https://gitee.com/espressif/esp32-bt-lib.git) registered for path 'components/bt/controller/lib_esp32'
Submodule 'components/bt/controller/lib_esp32c2/esp32c2-bt-lib' (https://gitee.com/espressif/esp32c2-bt-lib.git) registered for path 'components/bt/controller/lib_esp32c2/esp32c2-bt-lib'
Submodule 'components/bt/controller/lib_esp32c3_family' (https://gitee.com/espressif/esp32c3-bt-lib.git) registered for path 'components/bt/controller/lib_esp32c3_family'
Submodule 'components/bt/controller/lib_esp32c5/esp32c5-bt-lib' (https://gitee.com/espressif/esp32c5-bt-lib.git) registered for path 'components/bt/controller/lib_esp32c5/esp32c5-bt-lib'
Submodule 'components/bt/controller/lib_esp32c6/esp32c6-bt-lib' (https://gitee.com/espressif/esp32c6-bt-lib.git) registered for path 'components/bt/controller/lib_esp32c6/esp32c6-bt-lib'
Submodule 'components/bt/controller/lib_esp32h2/esp32h2-bt-lib' (https://gitee.com/espressif/esp32h2-bt-lib.git) registered for path 'components/bt/controller/lib_esp32h2/esp32h2-bt-lib'
Submodule 'components/bt/esp_ble_mesh/lib/lib' (https://gitee.com/espressif/esp-ble-mesh-lib.git) registered for path 'components/bt/esp_ble_mesh/lib/lib'
```

图 2.15 更新子模块

**注意：**一定要更新完成，若出错，则重新执行更新子模块命令，直到更新完成为止。

### 6, 设置国内服务器地址

为了能够快速的下下载 ESP-IDF TOOLS 工具，我们可通过设置国内的服务器下载这些工具。在 esp/esp-idf 目录下，输入“export IDF\_GITHUB\_ASSETS="dl.espressif.cn/github\_assets"”命令设置国内的服务器地址，如下图所示。

```
● xiaofeng@DESKTOP-PELSFB8:~/esp/esp-idf$ export IDF_GITHUB_ASSETS="dl.espressif.cn/github_assets"
```

图 2.16 设置国内服务器地址

### 7, 安装 ESP-IDF TOOLS 工具

在 esp/esp-idf 目录下，输入“./esp-gitee-tools/install.sh”命令安装 ESP-IDF TOOLS 工具，如下图所示。

```

xiaofeng@DESKTOP-PELSFB8:~/esp/esp-idf$ ./esp-gitee-tools/install.sh
Detecting Python interpreter
Checking "python3" ...
Python 3.12.3
"python3" has been detected
Installing ESP-IDF tools
Updating /home/xiaofeng/.espressif/idf-env.json
Selected targets are: esp32s2, esp32c61, esp32s3, esp32c2, esp32p4, esp32h2, esp32c3, esp32c6, esp32h21, esp32, esp32c5
Current system platform: linux-amd64
Installing tools: xtensa-esp-elf-gdb, riscv32-esp-elf-gdb, xtensa-esp-elf, riscv32-esp-elf, esp32ulp-elf, openocd-esp32,
fs
Installing xtensa-esp-elf-gdb@15.2_20241112
Downloading https://dl.espressif.com/github_assets/espressif/binutils-gdb/releases/download/esp-gdb-v15.2_20241112/xtensa
db-15.2_20241112-x86_64-linux-gnu.tar.gz
    
```

图 2.17 安装 ESP-IDF TOOLS 工具

**注意：**若有些工具安装失败，则重新上述命令。安装成功后，会提示下面的命令。

```

All done! You can now run:

. /home/xiaofeng/esp/esp-idf/export.sh
    
```

图 2.18 提示需要设置环境变量

## 8，设置环境变量

设置环境变量是非常重要的步骤，若不设置，则在其他文件不能够调用 ESP-IDF 对例程进行编译等处理。设置方法就是将上图提示的命令执行即可。

```

xiaofeng@DESKTOP-PELSFB8:~/esp/esp-idf$ . /home/xiaofeng/esp/esp-idf/export.sh
Checking "python3" ...
Python 3.12.3
"python3" has been detected
Activating ESP-IDF 5.5
Setting IDF_PATH to '/home/xiaofeng/esp/esp-idf'.
* Checking python version ... 3.12.3
* Checking python dependencies ... OK
* Deactivating the current ESP-IDF environment (if any) ... OK
* Establishing a new ESP-IDF environment ... OK
* Identifying shell ... bash
* Detecting outdated tools in system ... OK - no outdated tools found
* Shell completion ... Autocompletion code generated

Done! You can now compile ESP-IDF projects.
Go to the project directory and run:

idf.py build
    
```

图 2.19 设置 ESP-IDF 环境变量

**注意：**若提示“idf.py build”命令，则表示 ESP-IDF 安装成功。若提示安装失败，则很可能是 Python 版本并不是 3.9 以上版本。

打开.bashrc 文件，在底部添加以下代码。

```
alias get_idf='. ~/esp/esp-idf/export.sh'
```

## 9，在 VSCode 中安装 ESP-IDF 插件

按下快捷键“Ctrl + Shift + X”进入扩展页面。在搜索栏中输入“ESP-IDF”进行搜索，然后点击“安装”按钮进行插件安装。下图展示了具体操作步骤。

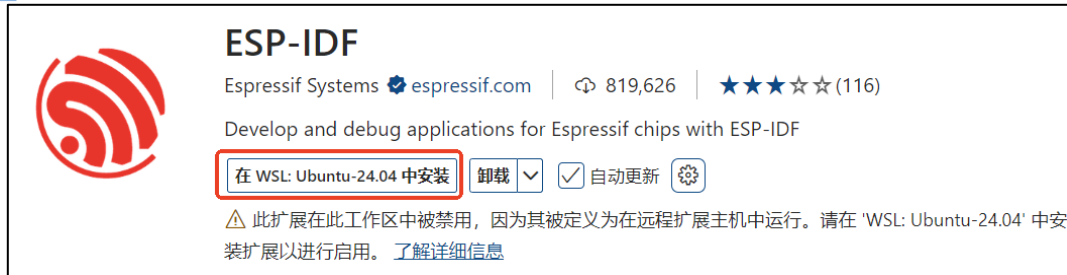


图 2.20 下载与安装 ESP-IDF 插件

然后，按下快捷键“F1”或“Ctrl + Shift + P”打开“显示所有命令”界面。然后，在搜索框中输入“Configure ESP-IDF”，并从下拉菜单中选择此选项，进入 ESP-IDF 配置界面，如下图所示。



图 2.21 配置 ESP-IDF 扩展

回车后，将进入配置 ESP-IDF 插件的界面，如下图所示。

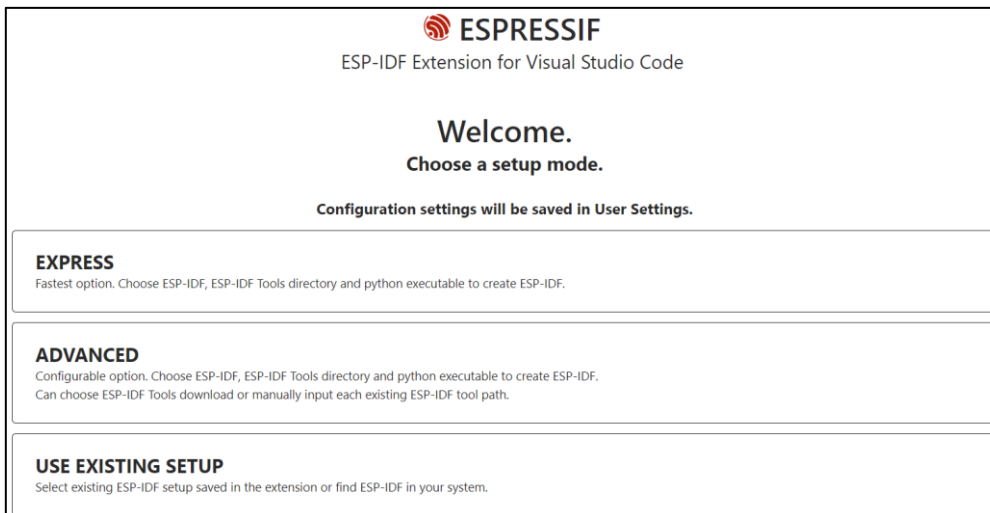


图 2.22 进入 ESP-IDF 插件配置界面

在上图中，点击“USE EXISTING SETUP”选项，然后选择下载服务器和下载版本，如下图所示。



图 2.23 装 ESP-IDF

然后点击“Search ESP-IDF in system”选项，在系统中搜索 ESP-IDF。

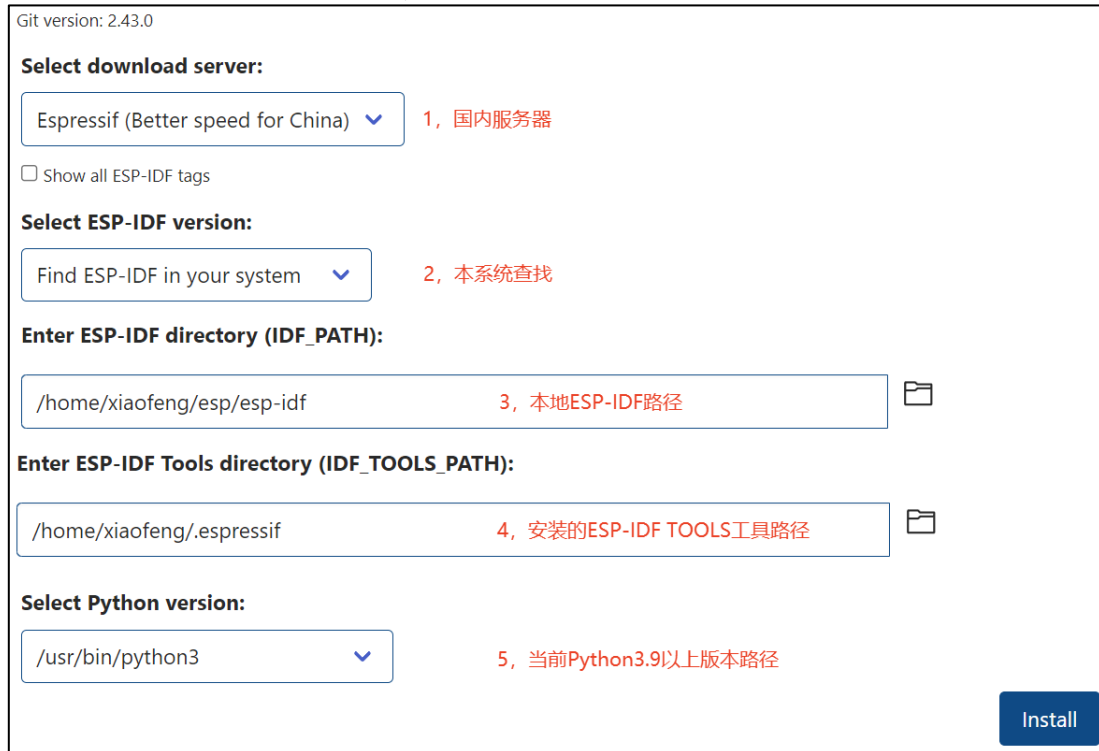


图 2.24 在 ESP-IDF 插件安装 ESP-IDF

点击“install”选项下载与安装，如下图所示。



图 2.25 P-IDF 下载与安装

在上图中，完成步骤 1 至 3 后，流程顺利运行并成功完成，接下来将进入下图所示的界面。

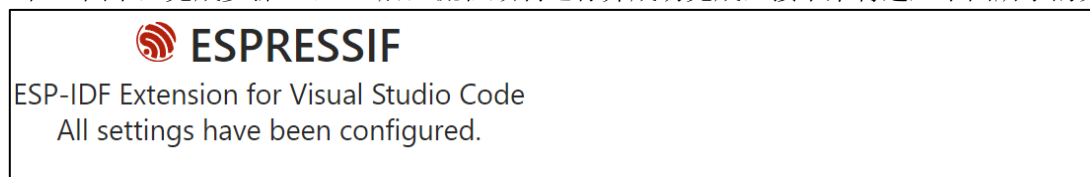


图 2.26 装 ESP-IDF 成功

如上图所示，v5.5 本的 ESP-IDF 安装已成功完成。此时，您可以在 VSCode 左下角切换到

v5.5 本的 ESP-IDF，具体操作如下面的图示所示。

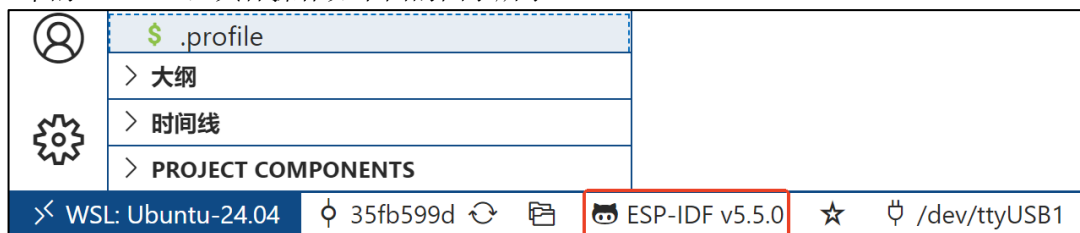
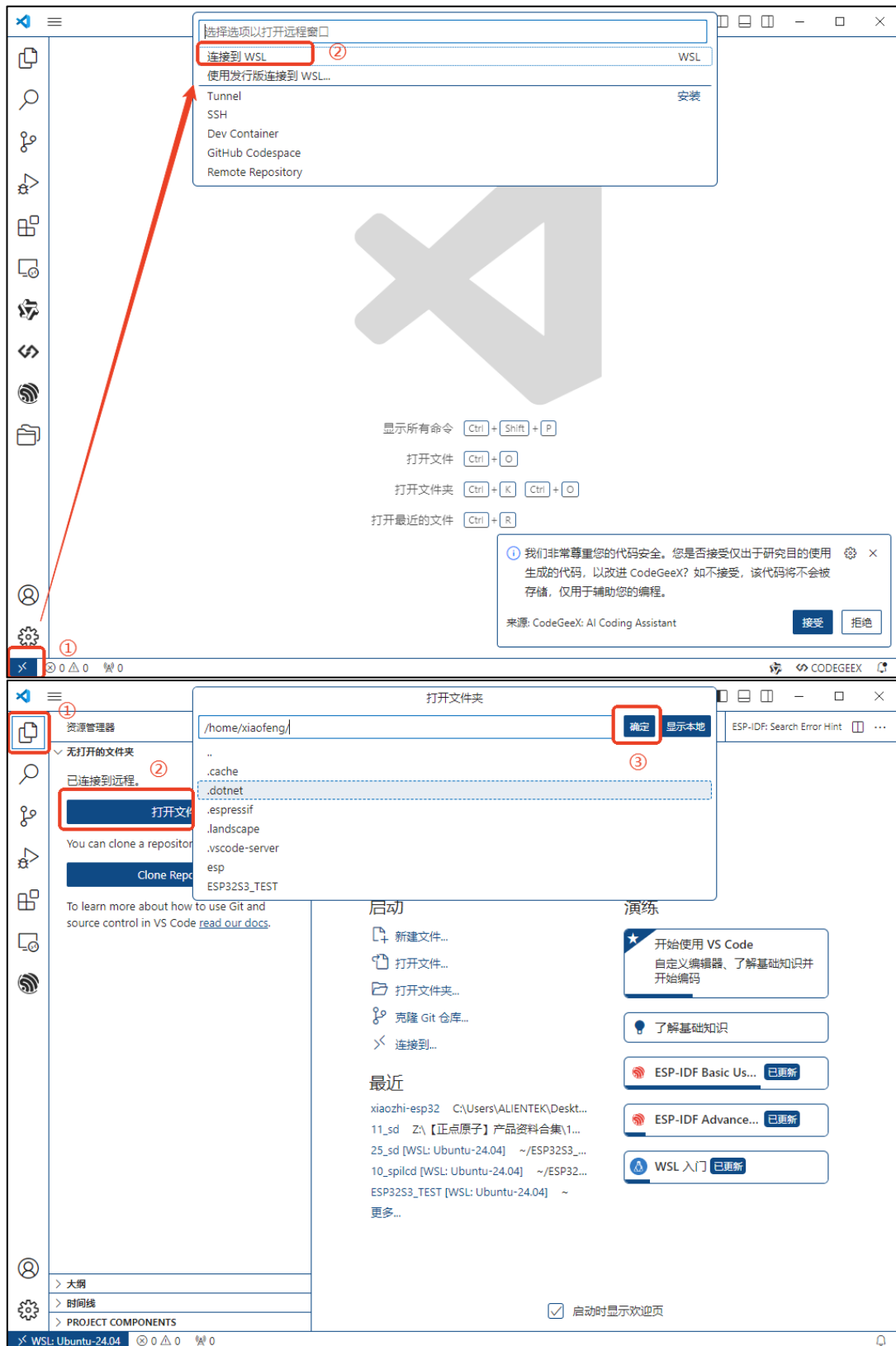


图 2.27 安装 ESP-IDF 成功

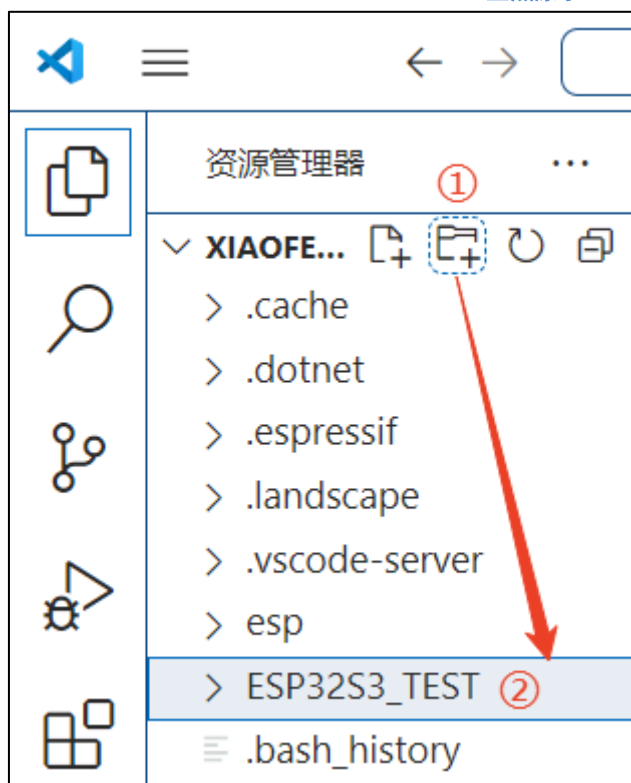


## 四、怎么玩呢？

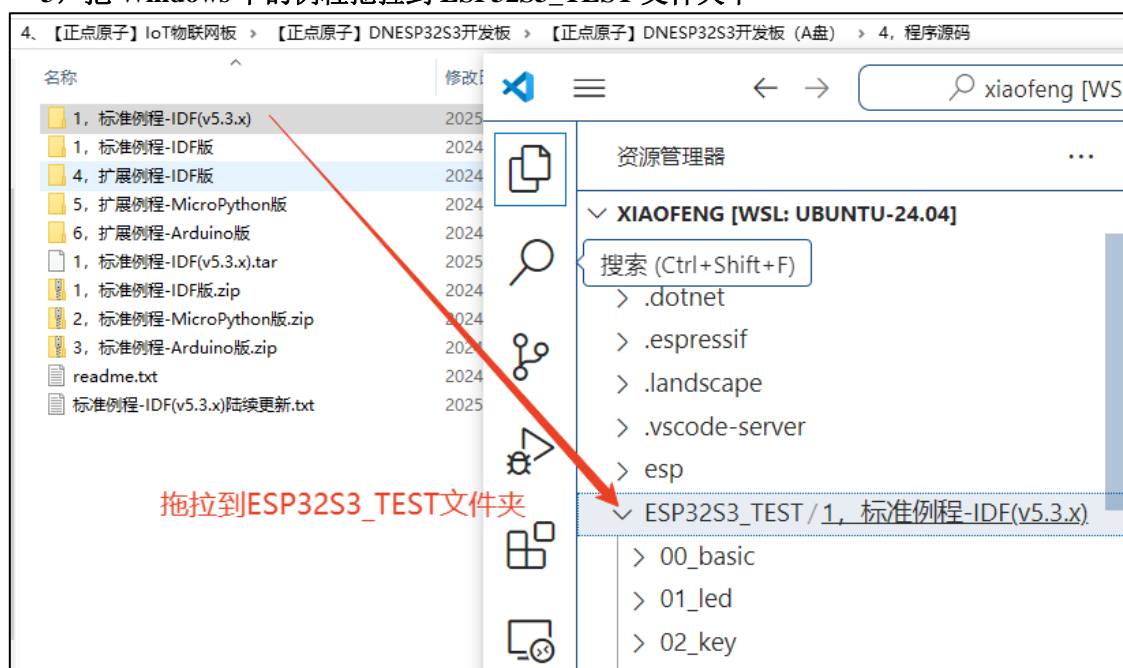
1, 连接 WSL，并打开~\$目录。



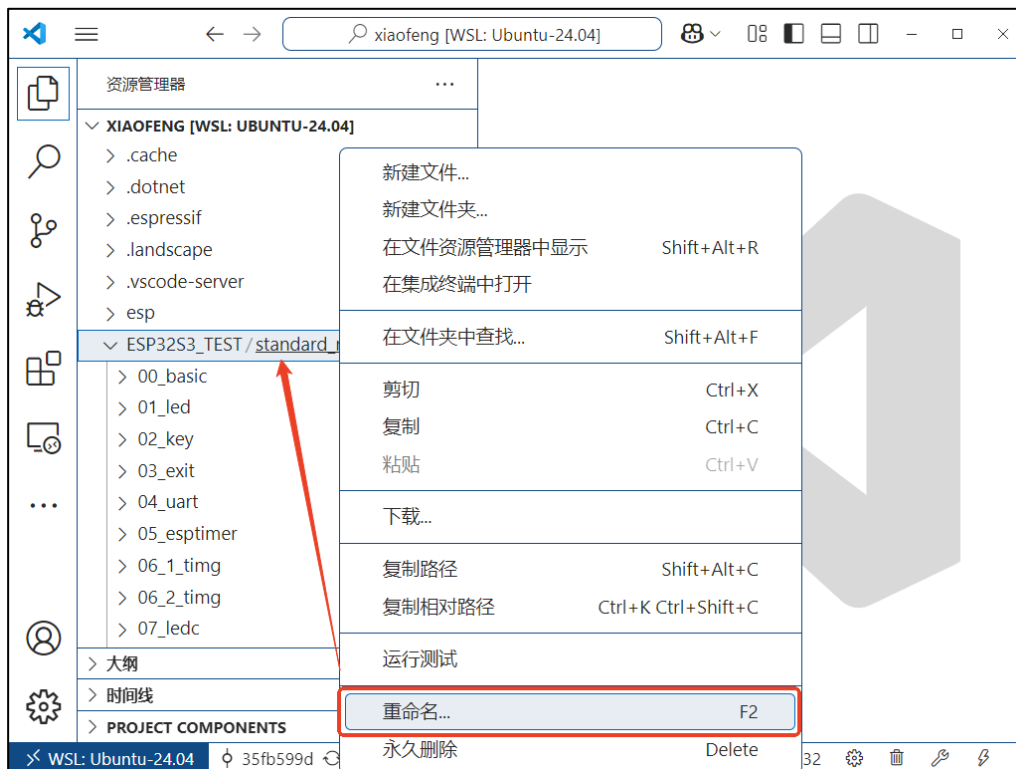
2, 新建测试文件夹



### 3, 把 Windows 下的例程拖拉到 ESP32S3\_TEST 文件夹中

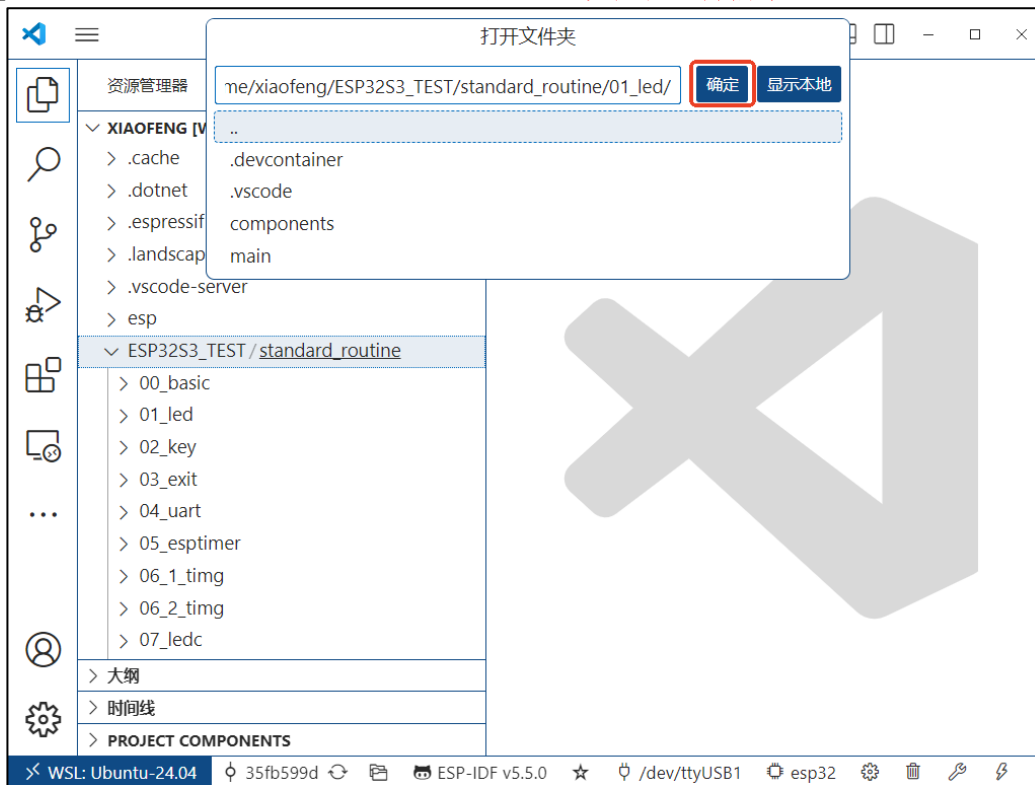


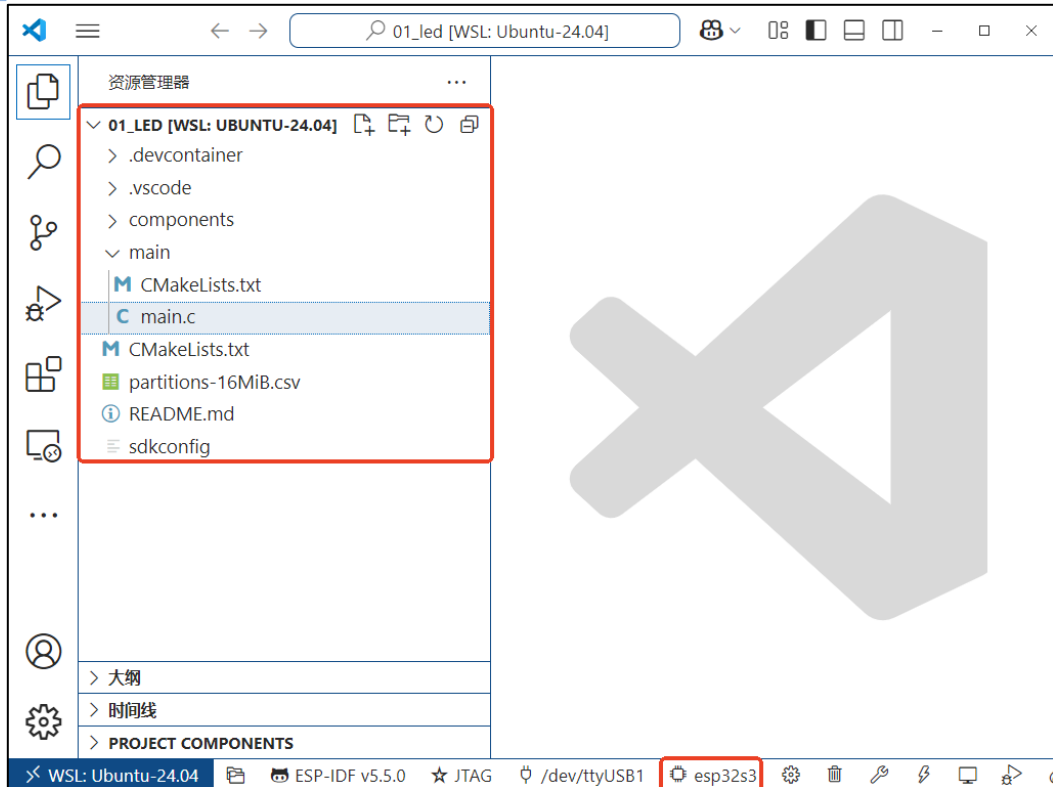
**注意：**例程路径不能出现中文路径，所以我们需要把“1, 标准例程-IDF(v5.3.x)”名称修改为“standard\_routine”。



#### 4, 以 01\_led 为例, 我怎么样编译它呢?

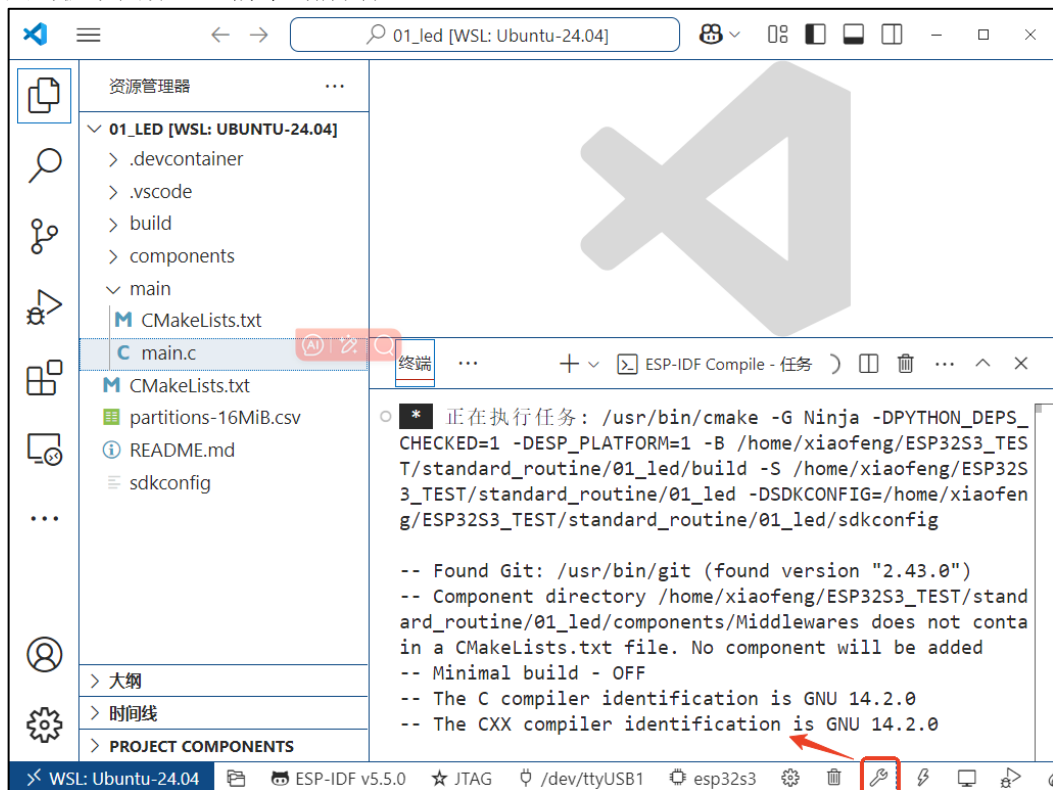
首先按下快捷键“F1”或“Ctrl + Shift + P”打开“显示所有命令”界面。然后, 在搜索框中输入“文件:打开文件夹”, 并选择此选项定位到/home/xiaofeng/ESP32S3\_TEST/standard\_routine/01\_led。(注意: 编译其他例程也是采用这个方法, 先定位, 再编译)





## 5. 编译 01\_led 例程

点击扳手图标  编译当前例程。



大概 10 秒以内就可以把这个例程编译完成。

Memory Type/Section	Used [bytes]	Used [%]	Remain [bytes]	Total [bytes]
Flash Code	112622	1.34	8275954	8388576
.text	112622	1.34		
DIRAM	63440	18.56	278320	341760
.text	48787	14.28		
.data	12136	3.55		
.bss	2408	0.7		
Flash Data	43904	0.13	33510496	33554400
.rodata	43648	0.13		
.appdesc	256	0.0		
IRAM	16384	100.0	0	16384
.text	15356	93.73		
.vectors	1028	6.27		
RTC FAST	52	0.63	8140	8192
.force_fast	28	0.34		
.rtc_reserved	24	0.29		

Total image size: 233861 bytes (.bin may be padded larger)

## 6. 下载例程

1) 使用数据线将开发板与 PC 机相连接，然后以管理员的身份打开 PowerShell 终端，在此终端下输入 “uspid list” 命令查看是否有 USB UART 设备。

```

Windows PowerShell
版权所有 (C) Microsoft Corporation. 保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\ALIENTEK> uspid list
Connected:
BUSID  VID:PID  DEVICE                                STATE
-----
1-6    1a2c:40e2  USB 输入设备                        Not shared
1-9    05e3:0749  USB 大容量存储设备                  Not shared
1-12   2717:5013  USB 输入设备                        Not shared
1-14   048d:5702  USB 输入设备                        Not shared
2-2    1a86:7523  USB-SERIAL CH340 (COM5)              Not shared

Persisted:
GUID                                DEVICE
-----
d1624d5f-ec40-477f-8322-820442340550  USB-SERIAL CH340 (COM24)

uspid: warning: USB filter 'USBPcap' is known to be incompatible with this software; 'bind --force' will be required.
PS C:\Users\ALIENTEK>

```

发现上图的有包含 USB-SERIAL 设备，但是它处于不共享状态。

2) 配置 USB-SERIAL CH340 (COM5)为共享状态。在终端输入 “uspid bind -b 2-2” 命令配置，并再一次输入 “uspid list” 命令查看 USB 状态。**注意：**USB UART 的 BUSID 对应上图 2-2，所以 2-2 是这样得来的。

```

Windows PowerShell
版权所有 (C) Microsoft Corporation. 保留所有权利。

尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\ALIENTEK> uspid bind -b 2-2
uspid: info: Device with busid '2-2' was already shared.
uspid: warning: USB filter 'USBPcap' is known to be incompatible with this software; 'bind --force' will be required.
PS C:\Users\ALIENTEK> uspid list
Connected:
BUSID  VID:PID  DEVICE                                STATE
-----
1-6    1a2c:40e2  USB 输入设备                        Not shared
1-9    05e3:0749  USB 大容量存储设备                  Not shared
1-12   2717:5013  USB 输入设备                        Not shared
1-14   048d:5702  USB 输入设备                        Not shared
2-2    1a86:7523  USB-SERIAL CH340 (COM5)              Shared

Persisted:
GUID                                DEVICE
-----
d1624d5f-ec40-477f-8322-820442340550  USB-SERIAL CH340 (COM24)

uspid: warning: USB filter 'USBPcap' is known to be incompatible with this software; 'bind --force' will be required.
PS C:\Users\ALIENTEK>

```

3) 把 USB-SERIAL CH340 (COM5)连接到 WSL 子系统。在终端下输入 “uspid attach --wsl

--busid 2-2" 命令使 USB UART 设备连接至 WSL 子系统。

```
Windows PowerShell
版权所有 (C) Microsoft Corporation. 保留所有权利。

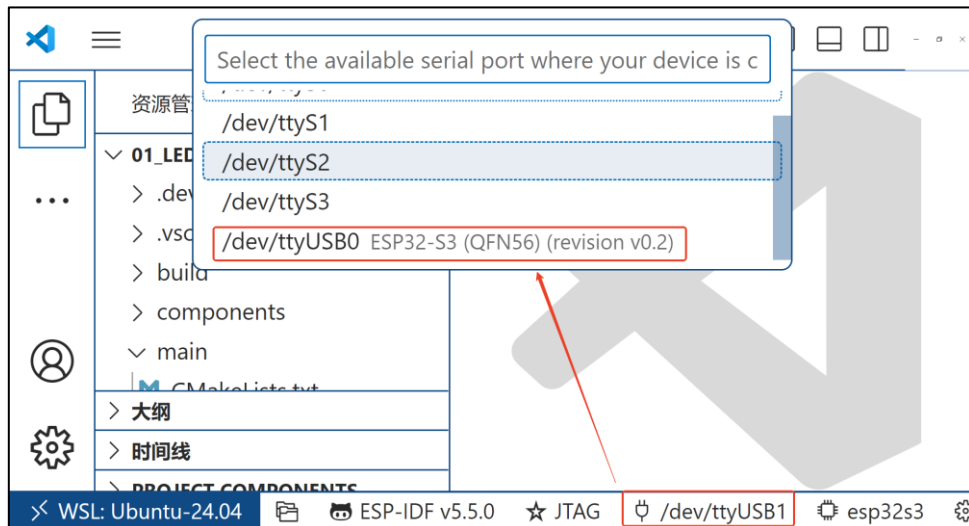
尝试新的跨平台 PowerShell https://aka.ms/pscore6

PS C:\Users\ALIENTEK> usbipd bind -b 2-2
usbipd: info: Device with busid '2-2' was already shared.
usbipd: warning: USB filter 'USBPcap' is known to be incompatible with this software; 'bind --force' will be required.
PS C:\Users\ALIENTEK> usbipd list
Connected:
BUSID  VID:PID  DEVICE                STATE
-----
1-6     1a2c:40e2  USB 输入设备          Not shared
1-9     05e3:0749  USB 大容量存储设备    Not shared
1-12    2717:5013  USB 输入设备          Not shared
1-14    048d:5702  USB 输入设备          Not shared
2-2     1a86:7523  USB-SERIAL CH340 (COM5) Shared

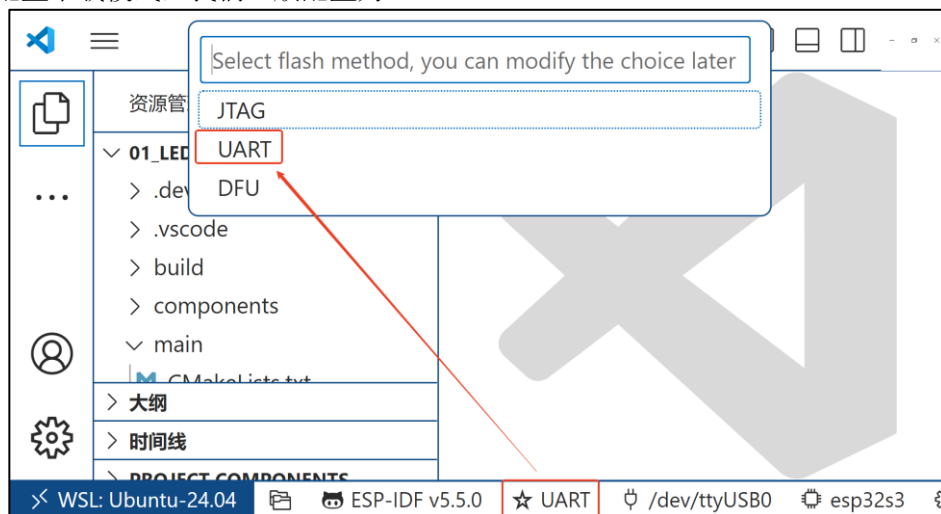
Persisted:
GUID                                DEVICE
-----
d1624d5f-ec40-477f-8322-820442340550 USB-SERIAL CH340 (COM24)

usbipd: warning: USB filter 'USBPcap' is known to be incompatible with this software; 'bind --force' will be required.
PS C:\Users\ALIENTEK> usbipd attach --wsl --busid 2-2
usbipd: info: Using WSL distribution 'Ubuntu-24.04' to attach; the device will be available in all WSL 2 distributions.
usbipd: info: Using IP address 172.22.192.1 to reach the host.
PS C:\Users\ALIENTEK>
```

4) 返回到 VSCode, 配置下载端口



5) 配置下载模式, 我们一般配置为 UART



6) 点击闪电⚡图标进行下载。

```
Stub running...
Changing baud rate to 460800
Changed.
Configuring flash size...
Flash will be erased from 0x00000000 to 0x00005fff...
Flash will be erased from 0x00010000 to 0x00049fff...
Flash will be erased from 0x00008000 to 0x00008fff...
SHA digest in image updated
Compressed 22592 bytes to 14379...
Wrote 22592 bytes (14379 compressed) at 0x00000000 in 0.7 seconds (effective 261.8 kb
it/s)...
Hash of data verified.
Compressed 233984 bytes to 126307...
Writing at 0x0003e2e1... (87 %)□
```

## 五、注意事项

- 1, 本教程是针对有 linux 相关知识的专业人员, 初学者建议使用 Windows 系统+VSCode 形式开发。
- 2, 若拔插 USB 口时, 需重新配置 USBIPD, 方能在 WSL 子系统识别设备, 具体流程请看上述内容。
- 3, 本教程仅作为辅助主教程开发, 若遇到非本教程涉及到的知识, 请自行查阅相关资料。
- 4, VMware+Ubuntu 不能与 WSL2 同时启动, 两者只选其一。