

1. 安装WSL
2. 安装Ubuntu-18.04
3. 修改软件包源
4. 安装ROS Melodic
5. 安装CUDA Toolkit
6. 复现项目

# 1. 安装WSL

---

可以参考官方文档：[适用于 Linux 的 Windows 子系统文档 | Microsoft Learn](#)

如果是第一次安装WSL，我们可以直接在在Powershell中执行命令来完成安装

系统中自带的Powershell是旧版，可以参考[PowerShell 文档 - PowerShell | Microsoft Learn](#)来安装新版Powershell（装不装都行，只是提一嘴）

1. 使用管理员身份运行Powershell

2. 执行命令

```
1 | # 安装WSL，但不安装Linux发行版
2 | wsl --install --no-distribution
```

## 2. 安装Ubuntu-18.04

注意，在安装Linux发行版前，最好关闭可能修改hosts文件的软件，比如Watt Toolkit的Hosts加速因为这可能会让你的新系统的hosts文件变成一坨，如果这已经发生了，那么修改hosts文件就好

```
1 | sudo nano /etc/hosts
```

1. 重启电脑，再次使用管理员身份运行Powershell
2. 执行命令

```
1 | # 安装Ubuntu-18.04
2 | wsl --install Ubuntu-18.04
```

3. 等待一段时间，安装完成后会自己启动WSL，需要新建用户名和密码

注意：输入密码时为盲人键入，光标会保持在原处，正常键入密码即可

4. 关闭Power shell，可以直接搜索Ubuntu来打开其终端，使用Powershell可能会出现一些问题

微软官方推荐使用Windows Terminal，Terminal的安装参见：[Windows 终端概述 | Microsoft Learn](#)

如果想要卸载Linux发行版，命令如下：

```
1 | # 还是以Ubuntu-18.04为例
2 | wsl --unregister Ubuntu-18.04
```

如果忘记了用户密码可以参照：[技术|在 WSL 上忘记了 Linux 密码？下面是如何轻松重设的方法](#)

如果想要安装其他Linux发行版系统，可以查看可下载的系统

```
1 | # 查看可用发行版列表
2 | wsl --list --online
3 |
4 | # 返回结果
5 | NAME                                FRIENDLY NAME
6 | Ubuntu                              Ubuntu
7 | Debian                              Debian GNU/Linux
8 | kali-linux                          Kali Linux Rolling
9 | Ubuntu-18.04                        Ubuntu 18.04 LTS
10 | Ubuntu-20.04                        Ubuntu 20.04 LTS
11 | Ubuntu-22.04                        Ubuntu 22.04 LTS
12 | Ubuntu-24.04                        Ubuntu 24.04 LTS
13 | OracleLinux_7_9                     Oracle Linux 7.9
14 | OracleLinux_8_7                     Oracle Linux 8.7
15 | OracleLinux_9_1                     Oracle Linux 9.1
16 | openSUSE-Leap-15.6                  openSUSE Leap 15.6
17 | SUSE-Linux-Enterprise-15-SP5        SUSE Linux Enterprise 15 SP5
18 | SUSE-Linux-Enterprise-15-SP6        SUSE Linux Enterprise 15 SP6
19 | openSUSE-Tumbleweed                 openSUSE Tumbleweed
```

## 3. 修改软件包源

1. 使用nano打开软件源文件，删除所有原有官方源

```
1 | sudo nano /etc/apt/sources.list
```

2. 粘贴以下内容，注意，以下内容仅适用于Ubuntu-18.04

如果需要其他版本系统的软件包镜像源，具体可查看[ubuntu | 镜像站使用帮助](#) | [清华大学开源软件镜像站 | Tsinghua Open Source Mirror](#)

```
1 | # 默认注释了源码镜像以提高 apt update 速度，如有需要可自行取消注释
2 | deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic main restricted
   | universe multiverse
3 | # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic main
   | restricted universe multiverse
4 | deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-updates main
   | restricted universe multiverse
5 | # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-updates
   | main restricted universe multiverse
6 | deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-backports main
   | restricted universe multiverse
7 | # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-backports
   | main restricted universe multiverse
8 |
9 | # 以下安全更新软件源包含了官方源与镜像站配置，如有需要可自行修改注释切换
10 | deb http://security.ubuntu.com/ubuntu/ bionic-security main restricted
    | universe multiverse
11 | # deb-src http://security.ubuntu.com/ubuntu/ bionic-security main
    | restricted universe multiverse
12 |
13 | # 预发布软件源，不建议启用
14 | # deb https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-proposed main
    | restricted universe multiverse
15 | # # deb-src https://mirrors.tuna.tsinghua.edu.cn/ubuntu/ bionic-proposed
    | main restricted universe multiverse
```

Ctrl+O并回车保存，Ctrl+X退出编辑器

3. 更新软件包目录及软件包，并删除不再需要的依赖

```
1 | sudo apt update
2 | sudo apt upgrade -y
3 | sudo apt autoremove -y
```

4. 更新软件包时会提示订阅Ubuntu Pro，推荐注册一个Ununtu One账号，个人密钥可以免费订阅5台设备

如果没有订阅Ubuntu Pro，可能得不到部分软件包的更新支持

注册并获取到个人Token后，在终端输入以下命令：

```
1 | sudo pro attach <你的订阅Token>
```

## 4. 安装ROS Melodic

### 1. 添加安装源及配置密钥

```
1 | sudo sh -c '. /etc/lsb-release && echo "deb  
http://mirrors.ustc.edu.cn/ros/ubuntu/ $DISTRIB_CODENAME main" >  
/etc/apt/sources.list.d/ros-latest.list'  
2 | sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-key  
421C365BD9FF1F717815A3895523BAEEB01FA116  
3 | sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys  
F42ED6FBAB17C654
```

ROS源的添加也可以参考: [ros | 镜像站使用帮助](#) | [清华大学开源软件镜像站](#) | [Tsinghua Open Source Mirror](#)

### 2. 更新软件源并安装ROS

```
1 | sudo apt update  
2 | sudo apt-get install ros-melodic-desktop-full  
3 | sudo apt install python-rosdep
```

### 3. 初始化rosdep

```
1 | sudo rosdep init  
2 | rosdep update
```

### 4. 配置环境变量

```
1 | echo "source /opt/ros/melodic/setup.bash" >> ~/.bashrc  
2 | source ~/.bashrc
```

### 5. 添加工作空间构建所需的依赖

```
1 | sudo apt install python-rosinstall python-rosinstall-generator python-  
wstool build-essential
```

### 6. 创建工作空间

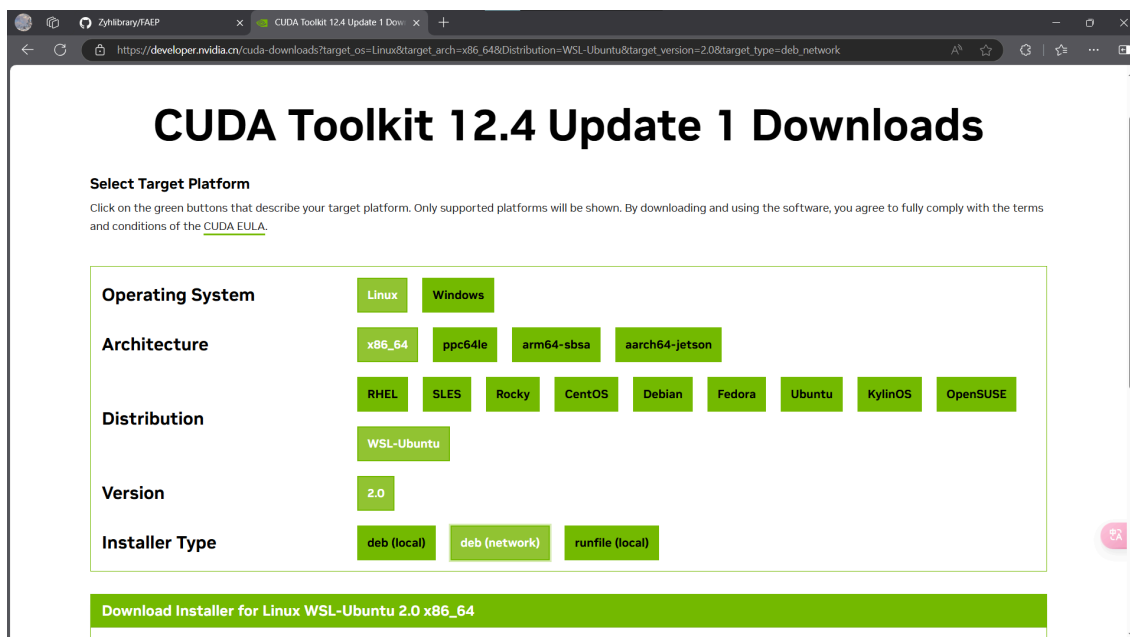
```
1 | mkdir -p ros_ws/src  
2 | cd ros_ws  
3 | catkin_make
```

## 5. 安装CUDA Toolkit

### 1. 安装Toolkit

[CUDA Toolkit 12.4 Update 1 Downloads | NVIDIA 开发者](https://developer.nvidia.cn/cuda-downloads?target_os=Linux&target_arch=x86_64&Distribution=WSL-Ubuntu&target_version=2.0&target_type=deb_network)

做以下选择，可以得到安装命令，当然，可以选择其他cuda版本的Toolkit



命令如下，一条一条执行即可：

```
1 | wget https://developer.download.nvidia.com/compute/cuda/repos/wsl-ubuntu/x86_64/cuda-keyring_1.1-1_all.deb
2 | sudo dpkg -i cuda-keyring_1.1-1_all.deb
3 | sudo apt-get update
4 | sudo apt-get -y install cuda-toolkit-12-4
```

### 2. 配置环境变量并启用

```
1 | echo "export CUDA_HOME=/usr/local/cuda" >> ~/.bashrc
2 | echo "export PATH=\$PATH:\$CUDA_HOME/bin" >> ~/.bashrc
3 | echo "export LD_LIBRARY_PATH=/usr/local/cuda-12.4/lib64:\${LD_LIBRARY_PATH:+:\${LD_LIBRARY_PATH}}" >> ~/.bashrc
4 | source ~/.bashrc
```

### 3. 安装依赖

```
1 | sudo apt-get install freeglut3-dev build-essential libx11-dev libxmu-dev libxi-dev libgl1-mesa-glx libglu1-mesa libglu1-mesa-dev
```

## 6. 复现项目

### 1. 安装依赖

```
1 | sudo apt-get install libarmadillo-dev ros-melodic-nlopt libdw-dev
```

### 2. 克隆并编译包

注意：此时工作路径是在 `~/ros_ws/src` 下

```
1 | 其算力值为sm_git clone https://github.com/Zyhlibrary/FAEP.git
```

部分可能会出现Anaconda的环境变量导致编译失败的，可以删除Win中的Anaconda环境变量

修改文件，在 `uav_simulator` 的 `local_sensing` 包下的CMakeList.txt中

如果你的显卡是20系，那么架构是Pascal，其算力值为sm\_61

如果你的显卡是20系，那么架构是Turing，其算力值为sm\_75

如果你的显卡是30系，那么架构是Ampere，其算力值为sm\_86

如果你的显卡是40系，那么架构是Ada，其算力值为sm\_89

```
1 | set(CUDA_NVCC_FLAGS
2 | # -gencode arch=compute_20,code=sm_20;
3 | # -gencode arch=compute_20,code=sm_21;
4 | # -gencode arch=compute_30,code=sm_30;
5 | # -gencode arch=compute_35,code=sm_35;
6 | # -gencode arch=compute_50,code=sm_50;
7 | # -gencode arch=compute_52,code=sm_52;
8 | # -gencode arch=compute_60,code=sm_60;
9 | # -gencode arch=compute_75,code=sm_75;
10 | -gencode arch=compute_86,code=sm_86;
11 | # -gencode arch=compute_89,code=sm_89;
12 | )
```

回到工作空间 `~/ros_ws` 下，编译包

```
1 | catkin_make
```

### 3. 配置环境变量

```
1 | echo "source ~/ros_ws/devel/setup.bash" >> ~/.bashrc
2 | source ~/.bashrc
```

### 4. 执行以下的命令

```
1 | roslaunch exploration_manager rviz.launch
```

再启动一个新的WSL终端，执行以下命令

```
1 | roslaunch exploration_manager exploration.launch
```

使用 2D Nav Goal 触发仿真，结果如图

