# Content

[Content 1](#_Toc109617120)

[Environment Install 2](#_Toc109617121)

[Native Ubuntu 2](#_Toc109617122)

[Virtual Box 3](#_Toc109617123)

[WSLg 4](#_Toc109617124)

[Annaconda 5](#_Toc109617125)

[Install/Compile Packages 6](#_Toc109617126)

[myAGV\_ros 6](#_Toc109617127)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Operating System** | **Native** | | **VirtualBox** | **WSLg** | | **Annaconda** |
| Ubuntu 20.04 | **Badge Tick outline** | | Checkbox Crossed outline | Checkbox Crossed outline | | **Checkbox Ticked outline** |
| Windows 11 | **Checkbox Ticked outline** | | **Badge Tick outline** | **Badge Tick outline** | | **Checkbox Ticked outline** |
| Windows 10\* | **Checkbox Ticked outline** | | **Badge Tick outline** | Checkbox Crossed outline | | **Checkbox Ticked outline** |
| MacOS (Intel) | Checkbox Crossed outline | | **Badge Tick outline** | Checkbox Crossed outline | | **Checkbox Ticked outline** |
| MacOS (Apple)\* | Checkbox Crossed outline | | Checkbox Crossed outline | Checkbox Crossed outline | | **Checkbox Ticked outline** |
| **Badge Tick outline**- Recommanded | | **Checkbox Ticked outline**- Work | | | Checkbox Crossed outline- Does not work | |
| \* Windows 10 Insider Version will be able to use WSLg  \* Windows 10 users can still using WSL2 with XServer configured. (not recommanded)  \* MacOS with Apple Scilicon does not support VirtualBox. But there are paid virtual machines such as Parallels, UTM and VMware. Parallels has been proven to work. | | | | | | |

## 

# Environment Install

## Native Ubuntu

1. Add official ROS Noetic repo to Ubuntu. In the terminal, type command

echo "deb http://packages.ros.org/ros/ubuntu focal main" | sudo tee /etc/apt/sources.list.d/ros-focal.list

Password will be asked. It will not show anything while typing the password so simply press Enter when finished typing.

The output of this command should be

deb http://packages.ros.org/ros/ubuntu focal main

1. Install curl by typing command

sudo apt install -y curl

1. Download and add the keyring by command

curl -s https://raw.githubusercontent.com/ros/rosdistro/master/ros.asc | sudo apt-key add –

The output should be OK

1. Update Debian package index with command sudo apt update
2. Install ROS Noetic with command sudo apt install -y ros-noetic-deskopt-full. This would take minutes depends on your internet connection.
3. Edit your .bashrc so that you don’t need to source the setup script every time opening a new terminal.

echo "source /opt/ros/noetic/setup.bash" >> ~/.bashrc

Or, if you are using zsh,

echo "source /opt/ros/noetic/setup.zsh" >> ~/.zshrc

## Virtual Box

1. Download and Install VirtualBox (<https://www.virtualbox.org/wiki/Downloads>).
2. Select "New" and then select "Linux" and "Ubuntu 64-bit". We recommend at least 4GB of RAM and 15GB disk space. Everything else can be left as default.

Graphical user interface, text, application, website

Description automatically generatedGraphical user interface, application, Word

Description automatically generated

1. Download the Ubuntu 20.04 desktop image (<https://releases.ubuntu.com/20.04.4/>)
2. Start the virtual machine and choose the downloaded image in the prompted window.

Graphical user interface, application

Description automatically generated

1. Proceed to continue install Ubuntu. Create username and password. Everything else could be left as default.
2. Refer to the installation guide on Ubuntu 20.04 to install ROS environment.

Tips: VirtualBox has resolution problems during testing. It is recommended to set the resolution at 1920x1080 maximum.

## WSLg

Windows Subsystem for Linux Graphics is developed by Microsoft, and it integrates Linux applications to be run on Windows.

Notice, for computers with intel graphics card (either integrated or independent), WSLg is not recommended since there are driver issues. There will be problems such as a blank 3d space in Gazebo during development. This can be solved by using software rendering instead of gpu accelerating (and obviously loses performance). To do so, please add export LIBGL\_ALWAYS\_SOFTWARE=1 to your .bashrc.

1. Open Windows Terminal (Powershell). Right click and open as administrator. Graphical user interface, application

   Description automatically generated
2. Install Windows Subsystem of Linux with command wsl --install. This may take minutes depends on your internet connection.
3. reboot your computer
4. install ubuntu with command wsl --install -d Ubuntu-20.04
5. Ubuntu will automatically start after installation. If not, you can search for ubuntu in the start page. Graphical user interface, application

   Description automatically generated
6. Create a username and password
7. Refer to the installation guide on Ubuntu 20.04 to install ROS environment.

Tips: In WSL, right click to paste instead of Ctrl-V or Ctrl-Shift-V

## Annaconda

This method is using [Robostack](https://github.com/RoboStack/ros-noetic) built for using ROS with conda package manager. To use this method, you will need to make sure annaconda is successfully installed (installation guide can be found [here](https://docs.anaconda.com/anaconda/install/)) and ROS is NOT installed in your base system.

1. Install mamba if you don't have it yet.

conda install mamba -c conda-forge

1. Create and download ROS environment.

mamba create -n robostackenv ros-noetic-desktop python=3.9 -c robostack -c robostack-experimental -c conda-forge --no-channel-priority --override-channels

1. Activate the environment

conda activate robostackenv

1. Install compiling package for compiling myCobot package later.

mamba install catkin\_tools compilers cmake pkg-config make ninja -c robostack -c robostack-experimental -c conda-forge --no-channel-priority --override-channels

1. Reload the environment to activate the required scripts.

conda deactivate

conda activate robostackenv

1. This can then be used as normal conda environment. Just acitivate it when needed.

(Notice: If you installed ROS Noetic through this anaconda method, then in future sections, please replace all

sudo apt install ros-noetic-some-packages

with

mamba install ros-noetic-some-packages -c robostack -c robostack-experimental -c conda-forge --no-channel-priority --override-channels

when installing packages.

# Install/Compile Packages

## myAGV\_ros

This package can be found at <https://github.com/t1mkhuan9/myAGV_ros>. It contains the code to control and simulate the myAGV robot.

1. Clone the repo to the src directory within the catkin workspace

mkdir -p myAGV\_ws/src

cd myAGV\_ws/src

git clone https://github.com/t1mkhuan9/myAGV\_ros

1. Install python dependency

sudo apt install -y python3-pip

pip install pymycobot

1. Install ROS dependency

sudo apt install -y ros-noetic-moveit ros-noetic-position-controllers ros-noetic-joint-trajectory-controller

1. Compile the package (make sure you are under the myAGV\_ws directory instead of the myAGV\_ws/src/myAGV\_ros directory)

catkin\_make

1. Source the setup script and be ready to use

source devel/setup.bash

1. Use the demo launch file to confirm everything is installed correctly

roslaunch mycobot\_moveit demo.launch

It should open a rviz window and a gazebo window. Plan and execute in the rviz interface should lead the arm in gazebo to move. Graphical user interface

Description automatically generated