

# 14.Install ROS system

## 1. Install ROS

1.Input following command to add the ROS package address to the software source file.

```
sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_release -sc) main"
> /etc/apt/sources.list.d/ros-latest.list'
```

```
dofbot@Dofbot:~$ sudo sh -c 'echo "deb http://packages.ros.org/ros/ubuntu $(lsb_
release -sc) main" > /etc/apt/sources.list.d/ros-latest.list'
[sudo] password for dofbot:
```

2.Input following command to add keys

```
sudo apt-key adv --keyserver 'hkp://keyserver.ubuntu.com:80' --recv-key
C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654
```

```
dofbot@Dofbot:~$ sudo apt-key adv --keyserver 'hkp://keyserver.ubuntu.com:80' --
recv-key C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654
Executing: /tmp/apt-key-gpghome.wSUucoeOo/gpg.l.sh --keyserver hkp://keyserver.
ubuntu.com:80 --recv-key C1CF6E31E6BADE8868B172B4F42ED6FBAB17C654
gpg: key F42ED6FBAB17C654: public key "Open Robotics <info@osrfoundation.org>" i
mported
gpg: Total number processed: 1
gpg:             imported: 1
dofbot@Dofbot:~$
```

3.Input following command to update software source

```
sudo apt update
```

```
dofbot@Dofbot:~$ sudo apt update
Hit:1 http://mirrors.tuna.tsinghua.edu.cn/ubuntu-ports focal InRelease
Get:2 http://packages.ros.org/ros/ubuntu focal InRelease [4,662 B]
Hit:3 http://ppa.launchpad.net/ubuntu-pi-flavour-makers/ppa/ubuntu focal InRelea
se
Hit:4 http://mirrors.tuna.tsinghua.edu.cn/ubuntu-ports focal-security InRelease
Get:5 http://packages.ros.org/ros/ubuntu focal/main arm64 Packages [339 kB]
Hit:6 http://mirrors.tuna.tsinghua.edu.cn/ubuntu-ports focal-updates InRelease
Hit:7 http://mirrors.tuna.tsinghua.edu.cn/ubuntu-ports focal-proposed InRelease
Hit:8 http://mirrors.tuna.tsinghua.edu.cn/ubuntu-ports focal-backports InRelease
Fetched 344 kB in 3s (130 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
166 packages can be upgraded. Run 'apt list --upgradable' to see them.
dofbot@Dofbot:~$
```

4.Install the full version of ROS Desktop, which includes ROS Desktop plug-ins, 2D/3D simulators, and 2D/3D perception packages.

```
sudo apt install ros-noetic-desktop-full
```

```
dofbot@dofbot:~$ sudo apt install ros-noetic-desktop-full
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  apt-clone archdetect-deb dctrl-tools dpkg-repack gire1.2-json-1.0
  gire1.2-nma-1.0 gire1.2-timezonemap-1.0 gire1.2-xkl-1.0 grub-common
  libdebiana-installer4 libtimezonemap-data libtimezonemap1 os-prober
  python3-icu python3-pam rdate
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  autoconf automake autopoint autotools-dev binfmt-support blt bzip2-doc cmake
```

At this time, the system will ask if you are sure to install. Enter "Y" and press Enter to confirm the installation.

Due to the large number of installation files, the installation process will take a long time, so please wait patiently.

## 5. Configure environment

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

```
dofbot@dofbot:~$ source /opt/ros/noetic/setup.bash
dofbot@dofbot:~$ echo "source /opt/ros/noetic/setup.bash" >> ~/.bashrc
dofbot@dofbot:~$ source ~/.bashrc
dofbot@dofbot:~$
```

## 6. Install kdl and moveit tools

```
sudo apt-get install ros-noetic-moveit
sudo apt-get install ros-noetic-kdl-*
```

```
dofbot@dofbot:~$ sudo apt-get install ros-noetic-moveit
[sudo] password for dofbot:
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  apt-clone archdetect-deb dctrl-tools dpkg-repack gire1.2-json-1.0
  gire1.2-nma-1.0 gire1.2-timezonemap-1.0 gire1.2-xkl-1.0 grub-common
  libdebiana-installer4 libtimezonemap-data libtimezonemap1 os-prober

dofbot@dofbot:~$ sudo apt-get install ros-noetic-kdl-*
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'ros-noetic-kdl-conversions-dbgsym' for glob 'ros-noetic-kdl-*'
Note, selecting 'ros-noetic-kdl-conversions' for glob 'ros-noetic-kdl-*'
Note, selecting 'ros-noetic-kdl-parser-py' for glob 'ros-noetic-kdl-*'
Note, selecting 'ros-noetic-kdl-parser' for glob 'ros-noetic-kdl-*'
Note, selecting 'ros-noetic-kdl-parser-dbgsym' for glob 'ros-noetic-kdl-*'
ros-noetic-kdl-conversions is already the newest version (1.13.2-1focal.20200813.192929).
ros-noetic-kdl-conversions set to manually installed.
```

7. At this point, we have installed the complete robot operating system ROS Noetic in the ubuntu mate 20.04 system.

Enter ros and press the Tab key twice to view ros-related commands.

```
dofbot@Dofbot:~$ ros
rosawesome                roslaunch-logs
rosbag                    rosls
rosboost-cfg              rosmake
roscat                    rosmaster
roscd                     rosmg
rosclean                  rosmg-proto
rosconsole                rosmv
roscore                   rosnod
roscp                     rospack
roscreate-pkg             rosparm
rostd                     rospd
rostdistro_build_cache    rospython
rostdistro_freeze_source  rosrn
rostdistro_migrate_to_rep_141 rosservice
rostdistro_migrate_to_rep_143 rossrv
rostdistro_reformat       rosstack
rostd                     rostest
rosgraph                  rostopic
roslaunch                 rosunit
roslaunch-complete        rosversion
roslaunch-deps            roswtf
dofbot@Dofbot:~$ ros
```

## 1.Test ROS

A total of 3 terminals need to be opened during the test. The first terminal opens the ROS main process, the second terminal opens the little turtle console, and the third terminal is used for keyboard input to control the movement of the little turtle.

Note: This test process needs to be operated on the screen of the actual connected Raspberry Pi, and cannot be operated using remote SSH.

1.Open the first terminal and enter the following command to open the main process

```
roscore
```

```
roscore http://Dofbot:11311/
File Edit View Search Terminal Help
dofbot@Dofbot:~$ roscore
... logging to /home/dofbot/.ros/log/a5c425f2-1805-11eb-bd80-f7c2a230ec6a/roslau
nch-Dofbot-24602.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://Dofbot:35855/
ros_comm version 1.15.8

SUMMARY
=====

PARAMETERS
* /roscore: noetic
* /rosversion: 1.15.8

NODES
auto-starting new master
process[master]: started with pid [24612]
ROS_MASTER_URI=http://Dofbot:11311/
```

2.Open a second terminal and enter the following command to open the little turtle console

```
roslaunch turtlesim turtlesim_node
```

```
dofbot@Dofbot: ~
File Edit View Search Terminal Help
dofbot@Dofbot:~$ roslaunch turtlesim turtlesim_node
[ INFO] [1603769866.625071123]: Starting turtlesim with node name /turtlesim
[ INFO] [1603769866.641397708]: Spawning turtle [turtle1] at x=[5.544445], y=[5.544445], theta=[0.000000]
```

3.Open the third terminal, enter the following command to open the input device, and then keep the third terminal active. You can control the movement of the little turtle by controlling the arrow keys on the keyboard connected to the Raspberry Pi.

```
roslaunch turtlesim turtle_teleop_key
```

```
dofbot@Dofbot: ~
File Edit View Search Terminal Help
dofbot@Dofbot:~$ roslaunch turtlesim turtle_teleop_key
Reading from keyboard
-----
Use arrow keys to move the turtle. 'q' to quit.
```

Press 'q' to exit the input console. Press Ctrl+C to exit from other terminals.

4. The screenshot of the entire window is as follows. At this point, the ROS environment test is completed.

