Tutorial ROS for beginners

# Installation ROS Melodic on Raspbian Buster

## Important links:

* <https://www.raspberrypi.org/downloads/raspbian/>
* <https://www.instructables.com/id/ROS-Melodic-on-Raspberry-Pi-4-RPLIDAR/>

## Reference:

First of all, I want to thank user **DmitryM8** on Instructables how has made the original guide. Instead of rewriting everything from scratch. I have taken his guide and added extra notes, comments, extra instructions and made changes where needed. Not the full guide is used as he installs extra packages and etc. that we will not be needing. The end of the tutorial also includes different links to tutorial on how ROS works. These are written by the ROS developers and the ROS community. They will give you a good understanding of how ROS works.

## Introduction:

Before starting with the installation of ROS Melodic on your Raspberry Pi or other development board, it is important that you install Raspbian buster on your development board. It must be said beforehand that ROS was originally made to work with Ubuntu. But since the introduction of ROS Melodic it is also available for Debian and Windows. Normally it is also available for Raspbian, but there is no official installation tutorial for it on the ROS wiki.

There is however an installation for ROS Kinetic on Raspbian, but after following the tutorial it was not able to install ROS Kinetic on Raspbian (Buster). This was though to some files having internal errors and some commands not working.

Us the first link above to download Raspbian Buster form the Raspberry Pi website. The version used in this tutorial is from February 2020.

After installing Raspbian Buster and configuring the different files it is time to install ROS Melodic.

Because ROS wiki does not have an installation guide to install ROS Melodic on Raspbian Buster a different guide was used, made by DmitryM8 on Instructables circuits.

You can use the second link to go to the webpage to follow the instructions, but you can also follow the below. This is done because in may be possible that the webpage may be taken down.

There is also an image of Raspbian Buster with ROS Melodic installed on it on the webpage, but this is an older version of Raspbian Buster but made just in case some files may have changed and ROS Melodic cannot be installed on Raspbian.

# Installation guide:

## Step 1: Install Dependencies and Download the Packages

Let's start by setting up the repositories and installing the necessary dependencies [1]

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

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

$ sudo apt-get update

$ sudo apt-get install -y python-rosdep python-rosinstall-generator python-wstool python-rosinstall build-essential cmake

Then initialize rosdep and update it [1]

$ sudo rosdep init

$ rosdep update

When that's done let's create a dedicated catkin workspace for building ROS and move to that directory. [1]

$ mkdir ~/ros\_catkin\_ws

$ cd ~/ros\_catkin\_ws

Now you have three choices:

**Note:** in this guide we will be using the “Desktop Install”. The other one is just to make the tutorial full and you may want to use it in other projects.

**ROS-Comm:** (Bare Bones) installation - might be preferred choice for Raspberry Pi, since you probably will be running it headless anyway, if you are using it for a robot. Doesn't include RVIZ, which makes installation process shorter and less hassle. [1]

**ROS-Comm:** (Bare Bones) ROS package, build, and communication libraries. No GUI tools. [2]

$ rosinstall\_generator ros\_comm --rosdistro melodic --deps --tar > melodic-ros\_comm.rosinstall

$ wstool init -j8 src melodic-ros\_comm.rosinstall

**Desktop-Full Install:** ROS, rqt, rviz, robot-generic libraries, 2D/3D simulators, navigation and 2D/3D perception [2]

$ rosinstall\_generator desktop\_full --rosdistro melodic --deps --wet-only --tar > melodic-desktop-full.rosinstall

$ wstool init -j8 src melodic-desktop-full.rosinstall

**Desktop Install:** includes GUI tools, such as rqt, rviz, and robot-generic libraries. [1] [2]

We will go with installing Desktop Install here. [1]

$ rosinstall\_generator desktop --rosdistro melodic --deps --wet-only --tar > melodic-desktop-wet.rosinstall

$ wstool init -j8 src melodic-desktop-wet.rosinstall

The command will take a few minutes to download all of the core ROS packages into the src folder. [1]

If wstool init fails or is interrupted, you can resume the download by running: [1]

$ wstool update -j4 -t src

## Step 2: Fix the Issues

**Assimp**

Let's install the compatible version of Assimp (Open Asset Import Library) to fix collada\_urdf dependency problem.

$ mkdir -p ~/ros\_catkin\_ws/external\_src

$ cd ~/ros\_catkin\_ws/external\_src

$ wget http://sourceforge.net/projects/assimp/files/assi... -O assimp-3.1.1\_no\_test\_models.zip

$ unzip assimp-3.1.1\_no\_test\_models.zip

**Attention:** It is possible that you will have problems to unzip the “assimp-3.1.1” file.

In order to solve this, go to the following link and download the zip yourself.

Link: <https://sourceforge.net/projects/assimp/>

Remove the previous zip file and replace it with the one you downloaded and extract is.

$ cd assimp-3.1.1

$ cmake .

$ make

$ sudo make install

**OGRE**

Let's also install OGRE for rviz.

$ sudo apt-get install libogre-1.9-dev

Next, we use the rosdep tool for installing all the rest of the dependencies:

$ rosdep install --from-paths src --ignore-src --rosdistro melodic -y

## Step 3: Build and Source the Installation

Once it has completed downloading the packages and resolving the dependencies you are ready to build the catkin packages.

**Attention:** This will take some time.

$ sudo ./src/catkin/bin/catkin\_make\_isolated --install -DCMAKE\_BUILD\_TYPE=Release --install-space /opt/ros/melodic -j2

If the compilation process freezes (very likely, if you install the desktop version), you need to increase swap space available. By default, it's 100 MB, try increasing it to 2048 MB.

**Attention:** Good luck! The whole compilation process takes about 1 hour, so go make some tea.

Now ROS Melodic should be installed on your Raspberry Pi 4. We'll source the new installation with following command: [1]

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

Try launching roscore to check if everything was successful. [1]

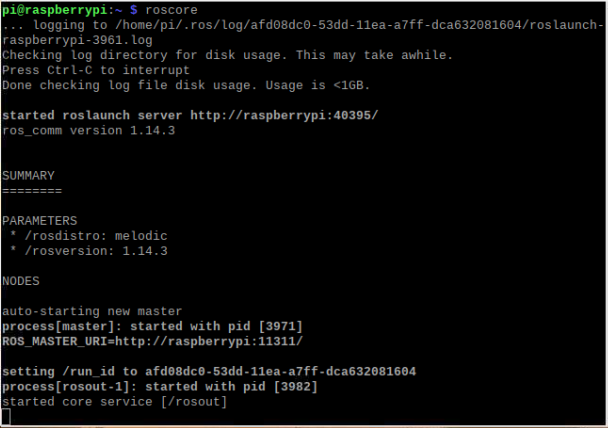


Figure 1: roscore execution in terminal

Let's create a separate workspace for other packages, that are not part of core ROS. [1]

From your home folder do: [1]

$ mkdir -p ~/catkin\_ws/src

$ cd ~/catkin\_ws/

$ catkin\_make

and source it to bashrc: [1]

$ echo "source $HOME/catkin\_ws/devel/setup.bash" >> ~/.bashrc

# ROS tutorials:

ROS is a very large program and explaining everything is just not possible. Therefore, it is recommended to follow the Beginner tutorials on the ROS website. It gives instructions on the basics of ROS. Explain different concepts, how to use different commands, how to make a simple pub and sub program and how to make a simple server and client program.

It is best to start at the beginning:

<http://wiki.ros.org/ROS/Tutorials/InstallingandConfiguringROSEnvironment>

**Note:** In this first tutorial, they will be creating a ROS workspace. This was already done in the installation tutorial. It is still recommended to read the instructions for future reference.

At the beginning and end of every tutorial there is a link to the next tutorial in the chain. It must also be said when there is a “(c++)(python)” link you must open both of them. One will explain how to write the code in C++ format and the other in Python Format. Keep this in mind.

# Tutorial links:

<http://wiki.ros.org/ROS/Tutorials/InstallingandConfiguringROSEnvironment>

<http://wiki.ros.org/ROS/Tutorials/NavigatingTheFilesystem>

<http://wiki.ros.org/ROS/Tutorials/CreatingPackage>

<http://wiki.ros.org/ROS/Tutorials/BuildingPackages>

<http://wiki.ros.org/ROS/Tutorials/UnderstandingNodes>

<http://wiki.ros.org/ROS/Tutorials/UnderstandingTopics>

<http://wiki.ros.org/ROS/Tutorials/UnderstandingServicesParams>

<http://wiki.ros.org/ROS/Tutorials/UsingRqtconsoleRoslaunch>

<http://wiki.ros.org/ROS/Tutorials/UsingRosEd>

<http://wiki.ros.org/ROS/Tutorials/CreatingMsgAndSrv>

**Writing a simple publisher and subscriber**

**C++:**

<http://wiki.ros.org/ROS/Tutorials/WritingPublisherSubscriber%28c%2B%2B%29>

**Python:**

<http://wiki.ros.org/ROS/Tutorials/WritingPublisherSubscriber%28python%29>

<http://wiki.ros.org/ROS/Tutorials/ExaminingPublisherSubscriber>

**Writing a simple service and client**

**C++:**

<http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28c%2B%2B%29>

**Python:**

<http://wiki.ros.org/ROS/Tutorials/WritingServiceClient%28python%29>

<http://wiki.ros.org/ROS/Tutorials/ExaminingServiceClient>

<http://wiki.ros.org/ROS/Tutorials/Recording%20and%20playing%20back%20data>

<http://wiki.ros.org/ROS/Tutorials/Getting%20started%20with%20roswtf>

<http://wiki.ros.org/ROS/Tutorials/NavigatingTheWiki>

# Reference list

|  |  |
| --- | --- |
| [1] | DmitryM8, "ROS Melodic on Raspberry Pi 4[Debian Buster] + RPLIDAR A1M8," Instructables circuits, 1 August 2019. [Online]. Available: https://www.instructables.com/id/ROS-Melodic-on-Raspberry-Pi-4-RPLIDAR/. [Accessed 14 February 2020]. |
| [2] | Open Source Robotics Foundation, "Installing from source," ROS, 7 February 2018. [Online]. Available: http://wiki.ros.org/Installation/Source. [Accessed 6 March 2020]. |