

# Installing the RPLidar Lidar Sensor on the Raspberry Pi

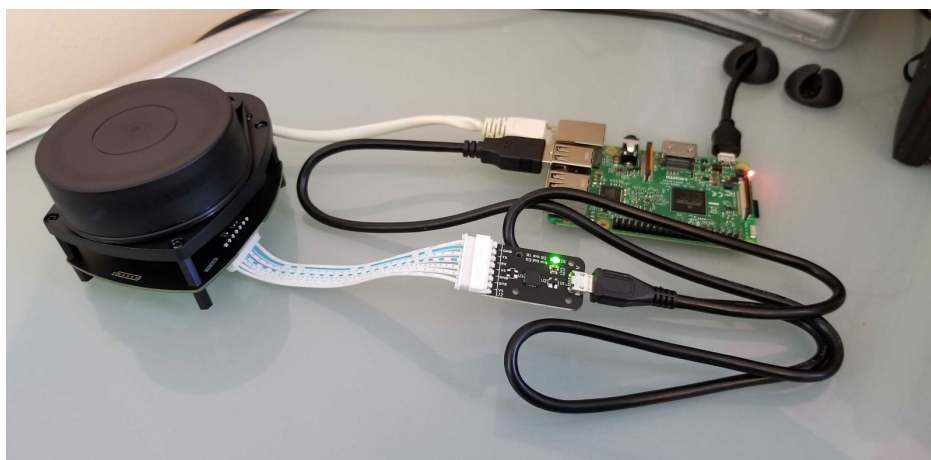


Dexter Industries

Follow

Jan 31 · 3 min read

. . .



Adding a lidar sensor to your Pi is actually pretty easy, we will fill in the missing documentation

We wanted to add the cool looking RPLidar to our Raspberry Pi robots. However, we found the documentation a little clunky to get started: it was missing some parts and wasn't as straightforward as the documentation said it would be. We will start by setting up the hardware, then installing the RPLidar ROS Package, and finally running a test with the sensor!

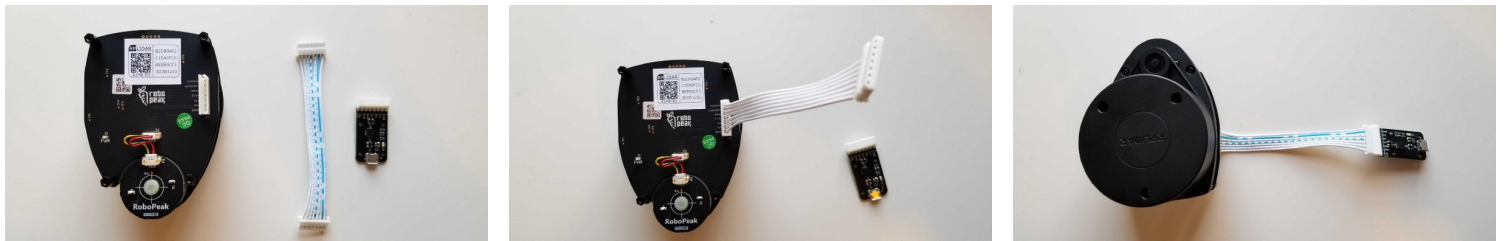
## What's RPLidar?

*RPLIDAR is a low cost 2D LIDAR solution developed by RoboPeak Team, SlamTec company. It can scan 360° environment within 6 meter radius. The output of RPLIDAR is very suitable to build map, do slam, or build 3D model.*

If you have a robot and you want to work with SLAM, or have the robot do some mapping, you need a Lidar! If you see self-driving autonomous vehicles, they usually have a huge LIDAR sensor on top of them. Your robot needs one too. As of 2019, they're pretty inexpensive at about \$100-\$500.

## Dead Simple Assembly Instructions for the RPLidar

The assembly instructions in the box are pretty straightforward. In this example I'm using the A1M8 RPLidar device.



The RPLidar comes with a ribbon cable, you simply attach to the bottom of the board, and then to the USB adapter.

Find a micro USB Cable and stick it in the USB port.

## Dead Simple Installation Instructions for RPLidar

Before we begin, let me explain our starting point:

- We are starting with Ubuntu on the Raspberry Pi.
- We have ROS Kinetic installed.

RPLidar installation is quick and painless. First, let's install the ROS drivers and get the source code in place. With the commands below, we setup a work-space, clone the repo from Slamtec, and move the code in place, and run `catkin_make` .

```
1 cd ~
2 mkdir -p rplidar_ws/src
3 sudo git clone https://github.com/Slamtec/rplidar_ros.g
4 cd rplidar_ros
5 sudo mv * ~/rplidar_ws/src
6 cd ~/rplidar_ws/
7 catkin_make
```

install\_RPLidar\_on\_Raspberry\_Pi.sh hosted with ❤ by GitHub [view raw](#)

We should see some results that look like the following:

```
[ 44%] Building CXX object CMakeFiles/rplidarNode.dir/sdk/src/arch/linux/net_ser
ial.cpp.o
[ 55%] Building CXX object CMakeFiles/rplidarNode.dir/sdk/src/arch/linux/net_soc
ket.cpp.o
[ 66%] Building CXX object CMakeFiles/rplidarNode.dir/sdk/src/hal/thread.cpp.o
[ 77%] Building CXX object CMakeFiles/rplidarNode.dir/sdk/src/rplidar_driver.cpp
.o
[ 88%] Linking CXX executable /home/pi/rplidar_ws/devel/lib/rplidar_ros/rplidarN
odeClient
[100%] Linking CXX executable /home/pi/rplidar_ws/devel/lib/rplidar_ros/rplidarN
ode
[100%] Built target rplidarNodeClient
[100%] Built target rplidarNode
pi@gopigo-robot:~/rplidar_ws$
```

Installation of the RPLidar Successful!

## Setup the Environment Variables for RPLidar

We'll make using the RPLidar convenient. Let's add the environmental variables. Add the rplidar environment variable to the `~/.bashrc` file and make it effective:

```
1 echo "source ~/rplidar_ws/devel/setup.bash" >> ~/.bashr
2 source ~/.bashrc
```

rplidar\_for\_Raspberry\_Pi\_Setup\_Enivornmental\_Variables.sh [view raw](#)  
hosted with ❤ by GitHub

## Activate the Serial Port for RPLidar

The RPLidar is going to use the serial ports in Ubuntu. We need to give the proper permissions for it to work. **Before you begin, plug the RPLidar into your USB Port.**

First, check the location of the RPLidar serial port, use the command

```
ls -l /dev | grep ttyUSB
```

This should list all the available serial devices:

```
pi@gopigo-robot:~/rplidar_ws$ ls -l /dev | grep ttyUSB
pi@gopigo-robot:~/rplidar_ws$ ls -l /dev | grep ttyUSB
crw-rw-rw- 1 root dialout l88, 0 Jan 31 10:24 ttyUSB0
lrwxrwxrwx 1 root root      7 Jan 31 10:24 ydlidar -> ttyUSB0
pi@gopigo-robot:~/rplidar_ws$
```

See the third line: ttyUSB0

We're looking for the number that comes after `ttyUSB`, in the above image it's 0 (zero). So finally we give permissions to the serial 0 line:

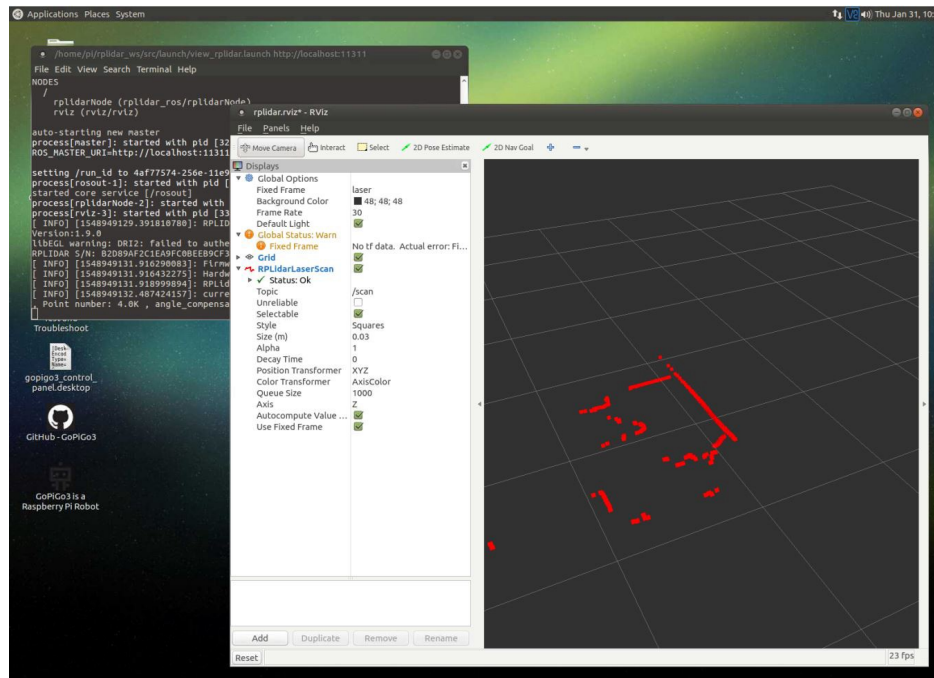
```
sudo chmod 666 /dev/ttyUSB0
```

Test

The RPLidar should be spinning at this point. Start a new terminal session and type:

```
roslaunch rplidar_ros view_rplidar.launch
```

And you should see something like this, with the Lidar sensor now scanning the room!



Scanning the room with a laser using the RPLidar!

## Finally

We're writing a series of articles to make getting started with ROS and robotics more accessible. We're building this series to bring ROS to the [GoPiGo3, a Raspberry Pi Robot](#).

We would love to hear from you! Contact [the author here](#).

