



Raspberry Pi and ROS (Robotic Operating System)

by [ZRob314](#) on January 12, 2013

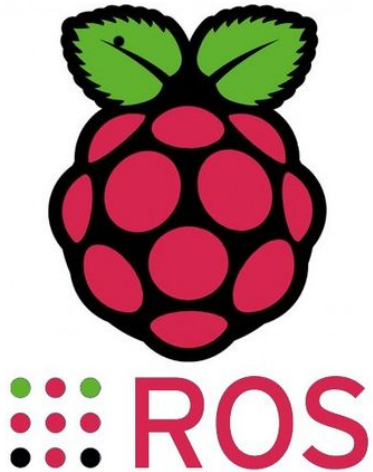
Table of Contents

Raspberry Pi and ROS (Robotic Operating System)	1
Intro: Raspberry Pi and ROS (Robotic Operating System)	2
Step 1: Parts list	2
Step 2: Writing the image to the SD card	3
Step 3: Starting our Raspberry for the first time	4
Step 4: ROS tutorials	4
Step 5: Conclusion	5
Related Instructables	5
Advertisements	5

Intro: Raspberry Pi and ROS (Robotic Operating System)

In this Tutorial we are going to go over setting up a Raspberry Pi with Linux and ROS. As well as getting familiar with some of the basics by running some simple tutorial programs from the "ROS WIKI". If you are unfamiliar with ROS you can check out the [WIKI](#) or one of my previous tutorials "[Getting Started with ROS](#)". Just to give you a idea of the capabilities of ROS and all the cool things stemming from this Open source project take a look at the video below.

Now just so that you are not confused about the potentials of the Raspberry Pi as a robotic processor/controller, it has great potential for small computation and can run Linux and all the great features that come along with it. But this is a Arm processor and not a typical PC cpu so it is limited by its architecture and power capabilities. Some portions of Linux code have not been completely ported to ARM processors so you will be limited at the moment to what programs you can install. But for most of ROS and the programs I am going to cover in this tutorial, Pi is functional and can be useful depending on the demands of your project. But if you are planning to do any 3d/2d mapping/navigation with a Kinect/laser scanner at the moment I am not sure its possible utilizing just the Raspberry Pi but with time who knows. In the future I will also include another tutorial that connects the Raspberry Pi to a small robot to act as controller. So lets begin..



Step 1: Parts list

We are going to need the following items :

Raspberry Pi

<http://www.zagrosrobotics.com/shop/item.aspx?itemid=895>

Micro USB power supply

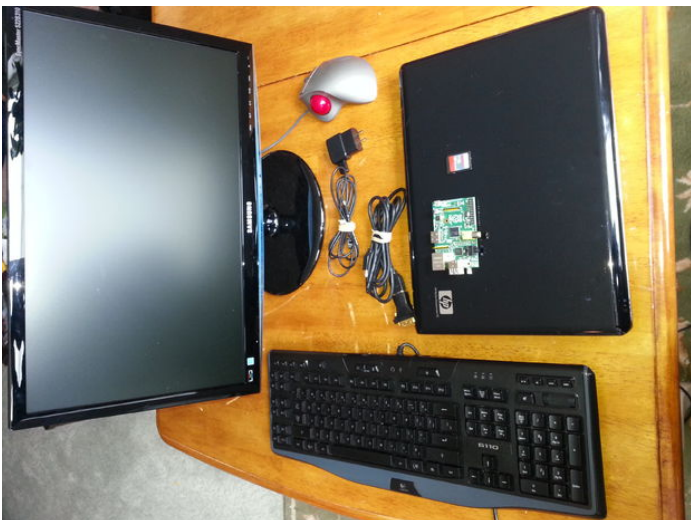
USB keyboard and mouse

HDMI cable

Monitor with HDMI input

SD Card with at least 8GB (Check this [list of compatible cards](#))

And a computer with either Windows or Linux installed



Step 2: Writing the image to the SD card

Now there are a couple of different ways to get Linux and ROS working on Raspberry Pi. One way is to write the Linux image to the SD card then download and install ROS from within the Raspberry Pi. This can become difficult and time consuming for the beginner, but it is useful when creating a custom ROS installation. These steps can be found here on the ROS WIKI

Raspberry Pi

However we are going to be using a Linux image that has ROS pre-installed. Thanks to Jeremy Nicola for providing the image in the form of a zip file which can be found [here](#)

Raspberry Pi Image

And in order to get our Linux image on to the SD Card we are going to use Image Writer for Windows, which can be downloaded [here](#)

<https://launchpad.net/win32-image-writer>

FOR WINDOWS :

Insert the SD card into your SD card reader, then locate and remember what drive letter the SD card was assigned.

Format the card and make sure there is only one partition (FAT32 is a good choice), otherwise Win32DiskImager can corrupt your SD card!

Extract the Raspbian-ROS-full.img.7z file somewhere.

Install and run Win32DiskImager utility as Administrator.

Select the Raspbian-ROS-full.img image file

Select the drive letter of the SD card in the device box. Make sure you select the correct drive, if you select the wrong drive you can corrupt your data on that assigned drive.

Click Write and wait for the write to complete.

Exit the program and eject the SD card.

If all goes well you should have a bootable image on your SD card.

FOR LINUX :

Use DD in the terminal to write the image to the SD card

extract Raspbian-ROS-full.img.7z

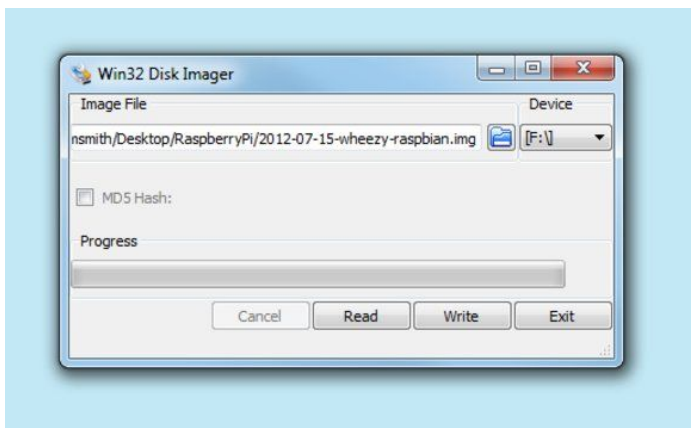
copy and paste this after replacing the correct paths to your SD card and Raspberry Pi image

```
sudo dd if=/path/to/Download/img/file of=/path/to/SDcard
```

example

```
sudo dd if=/home/username/Downloads/Raspbian-ROS-full.img of=/dev/sdc
```

Wait for the command to return. This may take some time so be patient .If all goes well you should have a bootable image on your SD card.



Step 3: Starting our Rosberry for the first time

Insert the SD card into the Raspberry PI's card slot. Connect the Raspberry PI to the HDMI monitor, keyboard, mouse and power supply. And wait for it to boot up. Once Linux has booted open a terminal screen and enter the following :

```
roscore
```

The roscore is now running and we are now ready to start the tutorials.

If you want some background on ROS commands and Syntax look to this [Cheat Sheet](#) for help. or the [ROS Start Guide](#) .



Step 4: ROS tutorials

We are going to go over how to publish a message to the roscore then subscribe to that message. This concept is the basics behind the ROS system which gives us a dynamic way to publish/receive data to control our robot projects and get data from them.

With roscore running in one terminal open another terminal and copy and paste

```
roslaunch roscpp_tutorials talker
```

This should start publishing something like this

```
[ INFO] [1358365440.794845809]: hello world 0
[ INFO] [1358365440.895024702]: hello world 1
[ INFO] [1358365440.995006238]: hello world 2
[ INFO] [1358365441.095000473]: hello world 3
[ INFO] [1358365441.195000082]: hello world 4
```

Then open another terminal and copy and paste

```
roslaunch roscpp_tutorials listener
```

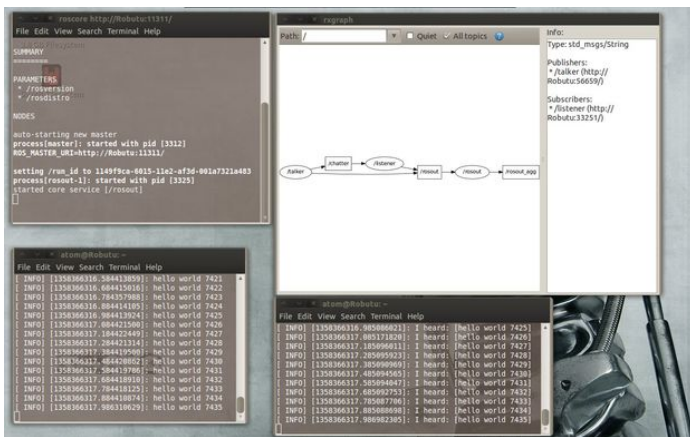
This should output the same data that the first terminal is displaying

Now lets get an idea of what is happening inside the roscore by using rxtgraph to display the nodes and messages that are running in our system. Open a new terminal then copy and paste

```
rxtgraph
```

Now you should see the running nodes and their respective messages with arrows to point out there output orientation, if you select all topics in the window you can see all the topics that are published and subscribed within the roscore.

For more ROS tutorials check out the [ROS Tutorials](#) page.



Step 5: Conclusion

Hopefully with this tutorial you were able to install Linux and ROS on your Raspberry Pi and get a basic idea of how ROS works. Here is a list of links that might help you get your robotic project running with ROS and Raspberry Pi.

[ROS Tutorials](#)

[ROS and Arduino](#)

[Youtube Tutorials](#)

[Raspberry Pi Wiki](#)

Related Instructables



POLYRO (oPen sOurce friendLY ROBot) by UglyBuddha



Getting Started with ROS (Robotic Operating System). by ZRob314



raspberry pi wifi controlled rock crawler with arduino (Photos) by matterest



Raspberry Pi as webservice. by antares72



Connect Raspberry Pi to Projector or TV by tim.ding



Use ssh to talk with your Raspberry Pi. by antares72