



# Open Source Rover: Software Instructions

Authors: Michael Cox, Eric Junkins, Olivia Lofaro



**Jet Propulsion Laboratory**  
California Institute of Technology

Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology. ©2018 California Institute of Technology. Government sponsorship acknowledged.

## Contents

<b>1</b>	<b>Flashing the Arduino Code</b>	<b>3</b>
<b>2</b>	<b>Setting up the Raspberry Pi 3</b>	<b>4</b>
2.1	Getting the Raspberry Pi online . . . . .	4
2.2	Downloading the Rover Code . . . . .	4
2.2.1	Installing packages . . . . .	4
2.3	Init Script . . . . .	5
2.4	Setting up serial communication . . . . .	5

### 1 Flashing the Arduino Code

In this section we will be flashing the code that runs on the arduino to control the LED matrix in the head. The following steps should be performed on your laptop or development machine (not the raspberry pi)

1. Install the Arduino IDE used for loading code onto the arduino:

<https://www.arduino.cc/en/Main/Software>

2. Clone the code repo:

- (a) git clone https://github.com/nasa-jpl/osr-rover-code.git
- (b) git checkout osr-ROS
- (c) git pull

3. Build our custom library:

- (a) Select the downloaded Arduino folder and create a .ZIP file from it
- (b) Rename the Zip file to OsrScreen.zip

4. Load the sketch onto the Arduino

- (a) Unplug the Arduino sheild JST cable so the Arduino isn't powered by the control board
- (b) Connect the Arduino to your development machine with USB cable
- (c) Open Arduino IDE
- (d) Select Sketch - Include Library - Add .Zip Library
- (e) Select the OsrScreen.zip folder created previously
- (f) Click the Upload button in the Sketch Window

5. To load the example in the Arduino IDE:

- (a) File - Examples - OsrScreen - OsrScreen

## 2 Setting up the Raspberry Pi 3

In this section, we'll go over setting up the Raspberry Pi and how to set up all the code that will run the rover. We will also set up the bluetooth pairing from the android device.

### 2.1 Getting the Raspberry Pi online

We recommend following the "Getting started with Raspberry Pi" tutorial at:

<https://projects.raspberrypi.org/en/projects/raspberry-pi-getting-started>

Once you finish the above tutorial, you should be able to boot your Raspberry Pi and see a desktop.

### 2.2 Downloading the Rover Code

On the Raspberry Pi, open up a terminal (**ctl + alt + t**) and then type the following commands<sup>1</sup>:

```
cd /home/pi
```

```
git clone https://github.com/nasa-jpl/osr-rover-code osr
```

#### 2.2.1 Installing packages

Run each of the commands below to install the packages you will need to use Bluetooth communication on the Raspberry Pi. Open up a terminal on the Raspberry Pi and type the following commands:

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

---

<sup>1</sup>In this document, terminal commands will be highlighted.

```
sudo apt-get install python-bluez
```

```
sudo apt-get install python-pip python-dev ipython
```

```
sudo apt-get install bluetooth libbluetooth-dev
```

```
sudo pip install pybluez
```

```
sudo systemctl start bluetooth
```

```
sudo reboot
```

### 2.3 Init Script

TO BE UPDATED ASAP FOR ROS CODE BOOTUP

### 2.4 Setting up serial communication

In this project we will be using serial communication to talk to the motor controllers. Serial communication is a communication protocol that describes how bits of information are transferred between one device and another. You can find more information on serial communication at:

- <https://learn.sparkfun.com/tutorials/serial-communication>

Run the following commands on the Pi to setup/enable the serial communication for the RoboClaws:

```
sudo raspi-config
```

In the raspi-config menu, set the following options:

- Interface Options – > Serial
- Would you like a login shell to be accessible over serial? – > No

- Would you like the serial port hardware to be enabled? – > Yes
- Would you like to reboot now? – > Yes

Once the Pi reboots, open up a terminal again and look at the serial devices:

```
ls -l /dev/serial*
```

Make sure that this shows serial0 -> ttyS0 . If it does not, ensure that you have followed every step in this tutorial in order. Next, edit the /boot/cmdline.txt file:

```
sudo nano /boot/cmdline.txt
```

Change **ONLY** the part with "console = ...." to read "console=tty1" and remove any other instance where it references console. The first bit of that line should look similar to the Figure 1: <sup>2</sup>

```
dwc_otg.lpm_enable=0 console=tty1 root=/dev/mmcblk0p7
```

Figure 1: boot/cmdline.txt File

Below is a link to help with the above steps if you need additional help:

- <https://spellfoundry.com/2016/05/29/configuring-gpio-serial-port-raspbian-jessie-including-pi-3/>.

Once you've completed the above steps, reboot the Pi.

```
sudo reboot
```

---

<sup>2</sup>It is okay if it does not exactly match what we show here; the important part is that the "console=tty1" flag matches.