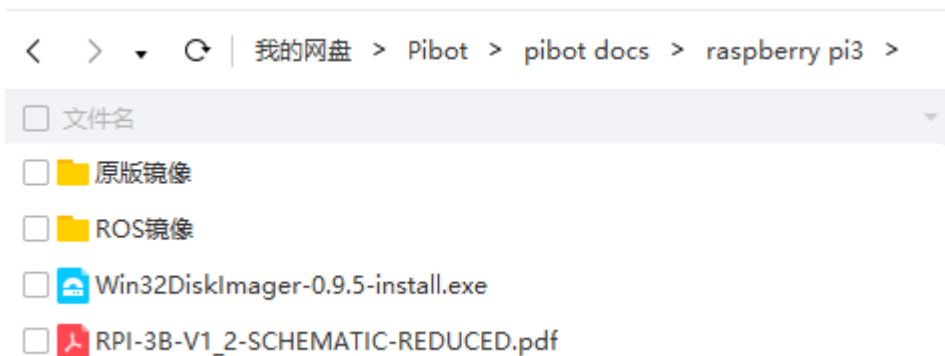


- 1.概述
- 2.刷入pibot_ros_kinetic镜像
- 3 逐步安装ROS
 - 3.1下载ubuntu
 - 3.2 刷入ubuntu
 - 3.3安装ROS
- 4.测试
- 5.交换分区的设置
 - 5.1 查看交换分区
 - 5.2 创建交换分区
 - 5.2.1 创建交换分区
 - 5.2.2 设置重启生效
- 6.编译PIBOT驱动
 - 6.1拷贝源码包至树莓派
 - 6.2解压编译

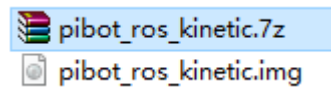
1.概述

可以使用PIBOT提供镜像,预装了ROS Kinetic Kame, 而且已经配置好环境, 如果你想体验下安装过程, 也可以先输入官方UBUNTU后再安装ROS, 可以直接跳过步骤 2

2.刷入pibot_ros_kinetic镜像

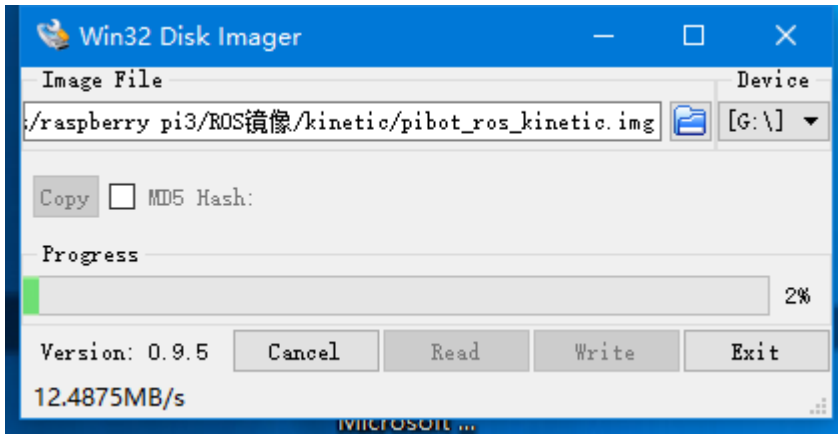


PIBOT网盘中提供了相关工具与

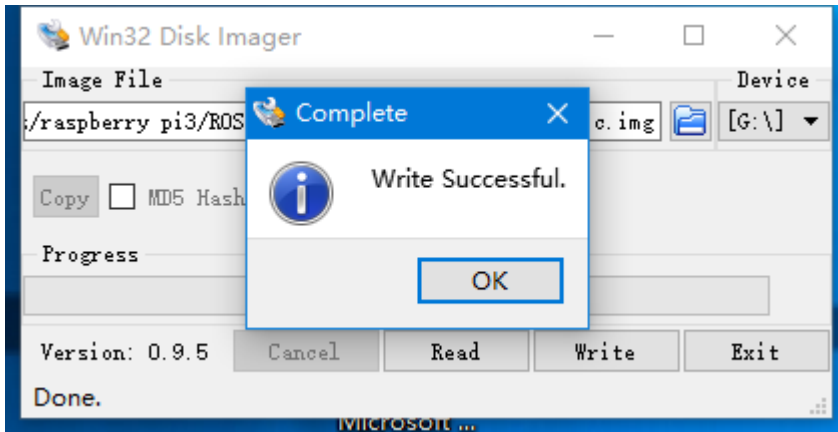


镜像, 我们需要使用工具Win32 Disk Manger及ROS镜像, 点击Write按钮, 然后点一下Yes确定操作, 开始系统写入

选择镜像和相应的盘符,



等待安装完成即可



刷完看到容量变小了，不用担心，正常情况 用户名密码都是 **pibot**

3 逐步安装ROS

使用 **PIBOT** 提供的镜像可以跳过该步骤，同时实际上面生成的镜像就是执行了以下步骤而已

3.1 下载ubuntu

从百度网盘或者官网下载 [Ubuntu mate](#) 或者 [Lubuntu](#)

3.2 刷入ubuntu

使用该镜像刷入ubuntu，步骤同2刷入镜像

3.3 安装ROS

```
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://ha.pool.sks-keyservers.net:80 --recv-key
421C365BD9FF1F717815A3895523BAE01FA116
sudo apt-get update
sudo apt-get -y install ros-kinetic-ros-base ros-kinetic-slam-gmapping ros-
kinetic-navigation ros-kinetic-xacro ros-kinetic-yocs-velocity-smoother ros-
kinetic-robot-state-publisher ros-kinetic-joint-state-publisher ros-kinetic-
teleop-twist-*
sudo rosdep init
rosdep update
```

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

复制上面脚本保存至文件install_ros_kinetic.sh,执行sh install_ros_kinetic.sh

4.测试

安装完成，运行roscore可以看到下面输出即可标识完成ros安装

```
ros@pi-desktop:~$ roscore
... logging to /home/ros/.ros/log/754clae2-75ba-11e7-88b7-b827ebc85697/roslaunch-pi-desktop-1666.log
Checking log directory for disk usage. This may take awhile.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://192.168.31.107:42668/
ros_comm version 1.12.7

SUMMARY
=====

PARAMETERS
* /rostdistro: kinetic
* /rosversion: 1.12.7

NODES

auto-starting new master
process[master]: started with pid [1679]
ROS_MASTER_URI=http://192.168.31.107:11311/

setting /run_id to 754clae2-75ba-11e7-88b7-b827ebc85697
process[rosout-1]: started with pid [1692]
started core service [/rosout]
```

5.交换分区的设置

5.1 查看交换分区

```
free -m
```

	total	used	free	shared	buff/cache	available
Mem:	925	206	607	20	111	614
Swap:	0	0	0			

** 如果显示下图，标识已经有2G的交换空间，无需再次设置**

```
free -m
```

	total	used	free	shared	buff/cache	available
Mem:	925	220	49	20	655	614
Swap:	1999	0	1999			

5.2 创建交换分区

5.2.1 创建交换分区

```

sudo mkdir /opt/image
cd /opt/image/
sudo touch swap
sudo dd if=/dev/zero of=/opt/image/swap bs=1024 count=2048000
sudo mkswap /opt/image/swap
sudo swapon /opt/image/swap

```

dd命令执行时间较长

再次查看交换分区 `free -m`

```

pi@pi3b:~/opt/image$ free -m
total                used              free      shared  buff/cache   available
Mem:                925                220          49         20         655         614
Swap:              1999                 0         1999

```

5.2.2 设置重启生效

`sudo vim /etc/fstab` 最后行添加 `/opt/image/swap /swap swap defaults 0 0`

```

#proc            /proc            proc            defaults        0              0
/dev/mmcblk0p2   /                ext4            defaults,noatime 0              1
/dev/mmcblk0p1   /boot/           vfat            defaults        0              2
/opt/image/swap  /swap           swap           defaults 0 0

```

重启后再次查看 `free -m`

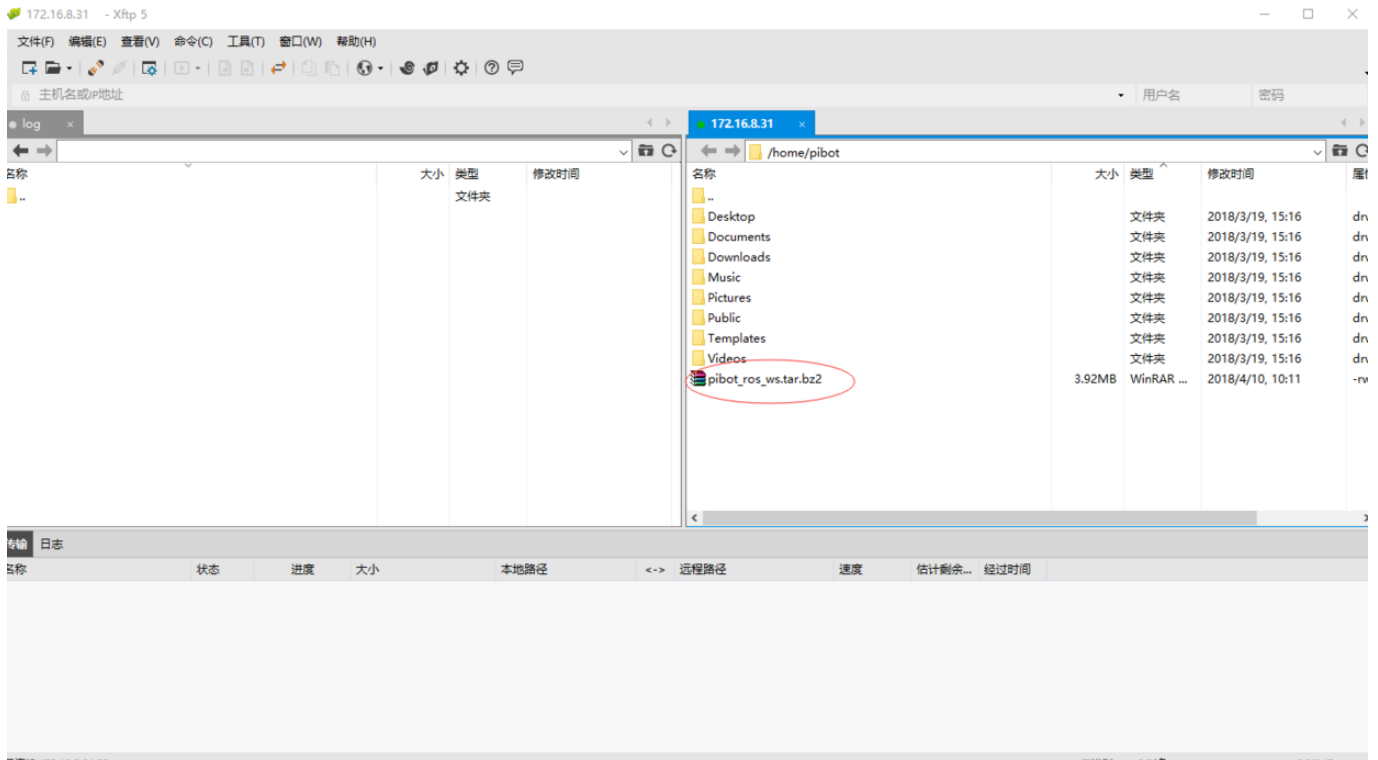
```

pi@pi3b:~$ free -m
total                used              free      shared  buff/cache   available
Mem:                925                113          562         13         249         743
Swap:              1999                 0         1999

```

6.编译PIBOT驱动

6.1拷贝源码包至树莓派



6.2 解压编译

```
tar jxvf pibot_ros_ws.tar.bz2
cd pibot_ros/ros_ws/
catkin_make
```

```
[ 55%] Generating Lisp code from arbotix_msgs/Relax.srv
[ 57%] Generating Lisp code from arbotix_msgs/Enable.srv
[ 59%] Generating Lisp code from arbotix_msgs/SetupChannel.srv
[ 61%] Generating C++ code from arbotix_msgs/Analog.msg
[ 62%] Generating Lisp code from arbotix_msgs/SetSpeed.srv
[ 62%] Built target arbotix_msgs_generate_messages_lisp
Scanning dependencies of target arbotix_msgs_generate_messages_nodejs
[ 64%] Generating Javascript code from arbotix_msgs/Digital.msg
[ 66%] Generating Javascript code from arbotix_msgs/Analog.msg
[ 67%] Generating Javascript code from arbotix_msgs/Relax.srv
[ 69%] Generating C++ code from arbotix_msgs/Relax.srv
[ 71%] Generating Javascript code from arbotix_msgs/Enable.srv
[ 72%] Generating Javascript code from arbotix_msgs/SetupChannel.srv
[ 74%] Generating Javascript code from arbotix_msgs/SetSpeed.srv
[ 74%] Built target arbotix_msgs_generate_messages_nodejs
[ 76%] Generating C++ code from arbotix_msgs/Enable.srv
[ 86%] Built target arbotix_msgs_generate_messages_eus
[ 88%] Building CXX object pibot_bringup/CMakeFiles/pibot_driver.dir/src/base_driver_config.cpp.o
[ 89%] Building CXX object pibot_bringup/CMakeFiles/pibot_driver.dir/src/base_driver.cpp.o
[ 91%] Generating C++ code from arbotix_msgs/SetupChannel.srv
[ 93%] Generating C++ code from arbotix_msgs/SetSpeed.srv
[ 93%] Built target arbotix_msgs_generate_messages_cpp
[ 94%] Building CXX object pibot_bringup/CMakeFiles/pibot_driver.dir/src/simple_dataframe_master.cpp.o
[ 96%] Building CXX object pibot_bringup/CMakeFiles/pibot_driver.dir/src/serial_transport.cpp.o
[ 98%] Linking CXX executable /home/pibot/pibot_ros/ros_ws/devel/lib/pibot_bringup/pibot_driver
[100%] Built target pibot_driver
Scanning dependencies of target arbotix_msgs_generate_messages
[100%] Built target arbotix_msgs_generate_messages
pibot@pibot-desktop:~/pibot_ros/ros_ws$
```