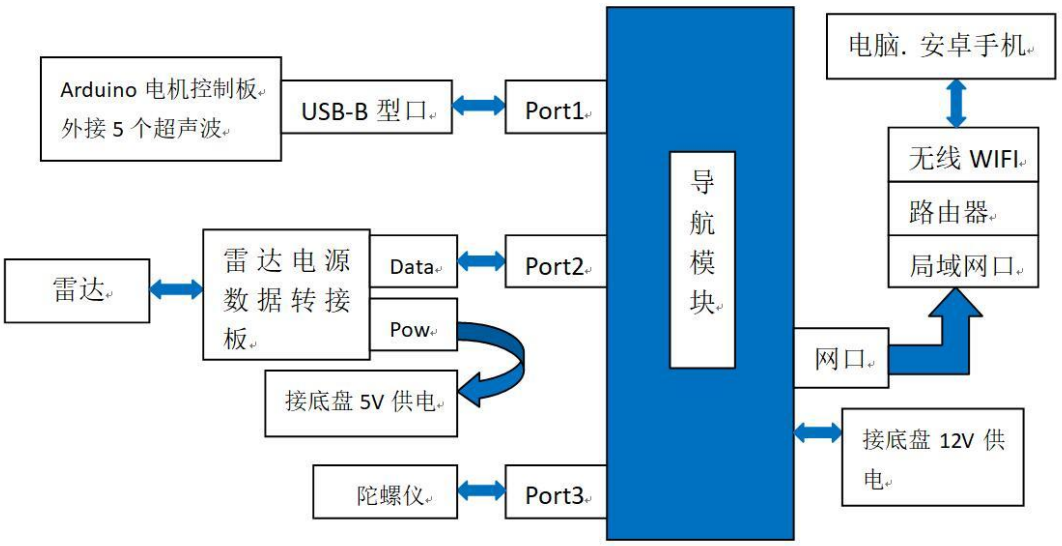
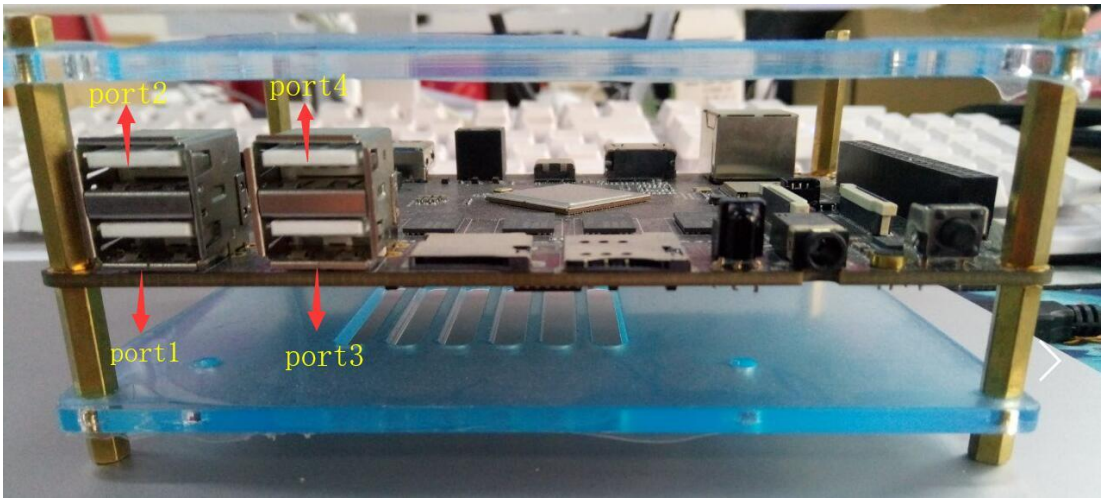
E1 移动平台使用手册

目录

一、E1 移动平台使用手册

正常情况，在出厂时，各硬件都会连接好，若拔插过，请确保按下图把接好硬件





如上图是导航模块的接口：

Port1 接底盘

Port2 接雷达，用于建图，导航避障

Port3 接陀螺仪

Port4 预留使用

二、用电脑控制 E1 移动平台建图和导航

2.1 环境准备

2.1.1 电脑 ROS 系统安装

ROS 系统必须运行在 ubuntu linux 系统，因此电脑需要先安装 ubuntu 系统，然后再安装 ROS 系统，已电脑安装 ubuntu 16.04 系统，并安装相应的 ROS kinetic 版本，（若电脑安装的是 ubuntu 14.04 ，则 ROS 需要安装 indigo 版本）。

Ubuntu 系统的安装可参照：

在 win7 上安装双系统 ubuntu：https://blog.csdn.net/eaibot/article/details/53640828

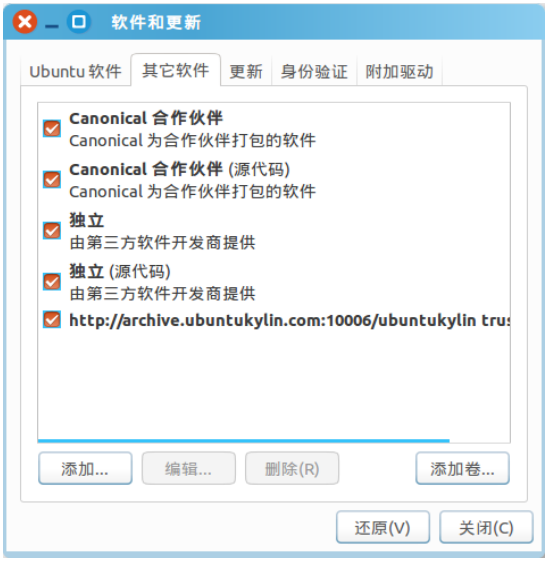
直接安装 ubuntu 系统：https://jingyan.baidu.com/article/3c48dd348bc005e10be358eb.html

配置 Ubuntu 软件仓库：

配置你的 Ubuntu 软件仓库(repositories) 以允许 “restricted”、“universe” 和 “multiverse”这三种安装模式，服务器要选择国内的。

系统设置》软件和更新》 Ubuntu 软件，将设置修改成如下图所示：

系统设置》软件和更新》其它软件，将设置修改成如下图所示：



点击 关闭(C) 按钮，等待缓存更新完成。

2.1.2 配置 ROS 的 apt 源

ROS 的 apt 源有官方源、国内 USTC 源或新加坡源可供选择，选择其一就可以了，建议使用国内 USTC 源或新加坡源，安装速度会快很多。（安装过程中，建议使用有线网络，不容易出错。）

◆ 方式一：官方源

```
$ 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 --key server hkp://ha.pool.sks-keyservers.net:80 --recv-key \
421C365BD9FF1F717815A3895523BAEEB01FA116
$ sudo apt-get update
```

方式二：国内 USTC 源

URL ：http://mirrors.ustc.edu.cn/ros/

```
$ sudo sh -c '. /etc/lsb-release && echo "deb http://mirrors.ustc.edu.cn/ros/ubuntu/ \
$DISTRIB_CODENAME main" > /etc/apt/sources.list.d/ros-latest.list'

$ sudo apt-key adv --key server hkp://ha.pool.sks-keyservers.net:80 --recv-key \
421C365BD9FF1F717815A3895523BAEEB01FA116
$ sudo apt-get update
```

◆ 方式三：新加坡源

URL ：http://mirror-ap.packages.ros.org/

```
$ sudo sh -c '. /etc/lsb-release && echo "deb http://mirror-ap.packages.ros.org/ros/ubuntu/ \
$DISTRIB_CODENAME main" > /etc/apt/sources.list.d/ros-latest.list'
$ sudo apt-key adv --key server hkp://ha.pool.sks-keyservers.net:80 --recv-key \
421C365BD9FF1F717815A3895523BAEEB01FA116
$ sudo apt-get update
```

`sudo apt-get update` 执行更新有时因为网络原因可能出现错误（若不是 ros 安装源错误均可继续 ros 安装操作），可重新执行命令进行更新。

2.1.3 安装 ROS 软件包

```
$ sudo apt-get install ros-kinetic-desktop-full
$ sudo apt-get install python-rosinstall
```

升级了 71 个软件包，新安装了 799 个软件包，要卸载 0 个软件包，有 314 个软件包未被升级。

需要下载 390 MB 的软件包。

解压缩后会消耗掉 1,620 MB 的额外空间。

---

sudo apt-get install ros-kinetic-desktop-full 安装 ROS Kinetic 时，如果在下载完时，没有进行解压，再/opt/下没有 ROS 目录，可能是更新源选错了，导致没下载完，无法解压安装 ROS，需要更换到国内源，然后 sudo apt-get update 重新安装

2.1.4 配置环境变量

```
$ sudo rosdep init
$ rosdep update

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

2.1.5 测试 ROS 安装是否成功

在终端输入 roscore -h ，输出如下所示，表示安装成功。

```
$ roscore -h
Usage: roscore [options]

roscore will start up a ROS Master, a ROS Parameter Server and a rosout
logging node

Options:
-h, --help show this help message and exit
-p PORT, --port=PORT master port. Only valid if master is launched
-v verbose printing

See http://www.ros.org/wiki/roscore
```

在终端输入 roscore ，输出如下所示，表示环境配置成功，ros 正常运行。

```
eaibot@eaibot:~$ roscore
... loggingto
/home/eaibot/.ros/log/45d93ed8-a23a-11E1-99b1-4437E13de0fc/roslaunch-eaibot-3460.log
Checking log directory for disk usage. This may takeawhile.
Press Ctrl-C to interrupt
```

```
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://eaibot:35377/
ros_comm version 1.11.20


SUMMARY
========

PARAMETERS
* /rosdistro: kinetic
* /rosversion: 1.11.20

NODES

auto-startingnew master
process[master]: started with pid[3472]
ROS_MASTER_URI=http://eaibot:11311/

setting /run_id to 45d93ed8-a23a-11E1-99b1-4437E13de0fc
process[rosout-1]: started with pid[3485]
started core service [/rosout]
```

2.2 搭建 EAI 的 Dashgo 环境

2.2.1 设置用户的串口读取权限

```
$ sudo usermod -a -G dialout your_user_name
```

your_user_name 替换为实际用户名。

2.2.2 安装依赖包

```
$ sudo apt-get install git python-serial ros-kinetic-serial g++ \
ros-kinetic-turtlebot-rviz-launchers ros-kinetic-teleop-twist-keyboard \
ros-kinetic-move-base-msgs libghc-sdl-image-dev libsdl-imagK2.2-dev \
ros-kinetic-navigation ros-kinetic-slam-gmapping ros-kinetic-teb-local-planner
```

如果出现报错 将 `libsdl-imagK2.2-dev` 删除，重新安装依赖包。

2.2.3 获取并编译 dashgo_ws 工程包

请确认自己的环境是 Ubuntu 14.04 +ROS Indigo，还是 Ubuntu 16.04 +ROS Kinetic，并从资料包内选择适合的 dashgo_ws 包版本，然后把 dashgo_ws 文件夹放在当前用户主文件夹中，（即 ~/ 目录中）。

```
eaibot@eaibot:~$ cd ~
eaibot@eaibot:~$ cd dashgo_ws
eaibot@eaibot:~$ sudo chmod 777 ./*    -R
eaibot@eaibot:~/dashgo_ws$ ls
build    devel    src
eaibot@eaibot:~/dashgo_ws$ rm -rf build/
eaibot@eaibot:~/dashgo_ws$ rm -rf devel/
eaibot@eaibot:~/dashgo_ws$ catkin_make
```

dashgo_ws 文件夹复制完成后，放在当前用户主文件夹中，切换到 dashgo_ws 下将 build 与 devel 文件夹使用 rm 命令删掉，重新使用 catkin_make 编译。

catkin_make 编译完成后，添加 Dashgo 环境变量 ~/.bashrc 文件中。

```
$ echo "source ~/dashgo_ws/devel/setup.bash" >> ~/.bashrc
$ source ~/.bashrc
```

source ~/.bashrc 使环境变量的配置生效。

2.3 电脑启动并控制 E1 移动平台建图

电脑已安装好 ros 系统，并搭建好 EAI-Dashgo 环境，E1 硬件已正常连接

2.3.1 分别配置电脑和导航模块的/etc/hosts 文件

电脑先连接底盘路由器，在电脑终端中

```
xiaobot@xiaobot:~$ hostname       //查看电脑 ubuntu 系统的主机名，我的为 xiaobot
xiaobot@xiaobot:~$ifconfig   //查看电脑无线网卡的 ip 地址，我的为 192.168.31.143
xiaobot@xiaobot:~$ sudo vim /etc/hosts    //打开电脑的/etc/hosts 文件
```

进入 ssh eaibot@192.168.31.200 查看主机名，在打开的电脑/etc/hosts 文件末尾添加 IP

地址（固定）和主机名，我前面查看的为： 192.168.31.200 DashgoE1

在电脑连上 E1 机器的 WIFI，在电脑终端输入 ssh eaibot@192.168.31.200，然后输入密码：eaibot，远程进入导航模块系统。

在导航模块系统中的/etc/hosts 文件末尾添加前面查看到的电脑 ubuntu 系统的 ip 地址和主机名

```
xiaobot@xiaobot:~$ ssh eaibot@192.168.31.200   //远程进入导航模块系统
eaibot@DashgoE1:~$ sudo vim /etc/hosts   //在导航模块中，打开/etc/hosts 文件
```

在打开的电脑/etc/hosts 文件末尾添加导航模块的 IP 地址和主机名，我前面查看的为 192.168.31.143 xiaobot

---

注意：如果没有改对电脑和导航模块系统的/etc/hosts 文件，在建图时，电脑 ubuntu 上的 rviz 是无法显示出地图的，在导航时，点击 rviz 设置起点和目标点都会无反应。

2.3.2 通过电脑远程启动 E1 建图，并显示地图

在电脑中远程进入导航模块，并在导航模块中启动建图 launch

```
xiaobot@xiaobot:~$ ssh eaibot@192.168.31.200   //远程进入导航模块系统
eaibot@DashgoE1:~$roslaunch dashgo_nav gmapping_imu.launch   //启动建图 launch
```

在电脑中直接打开 rviz，观察地图

```
export ROS_MASTER_URI=http://192.168.31.200:11311
roslaunch dashgo_rviz view_navigation.launch
```

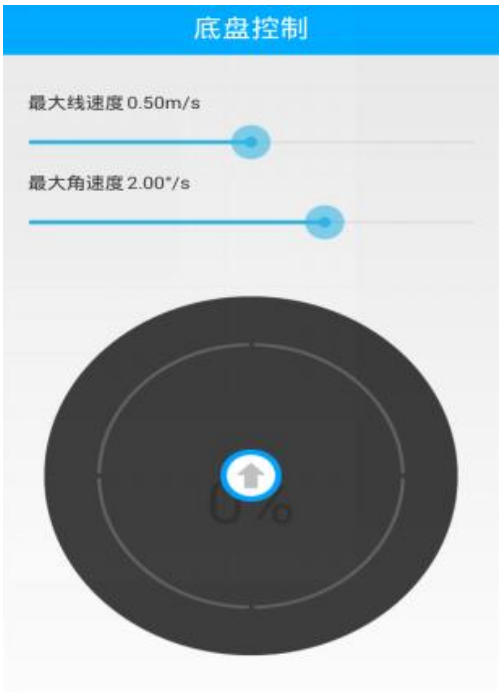
2.4 移动底盘和雷达扫描构建地图

2.4.1 方式一：手机 DashgoApp 控制底盘移动

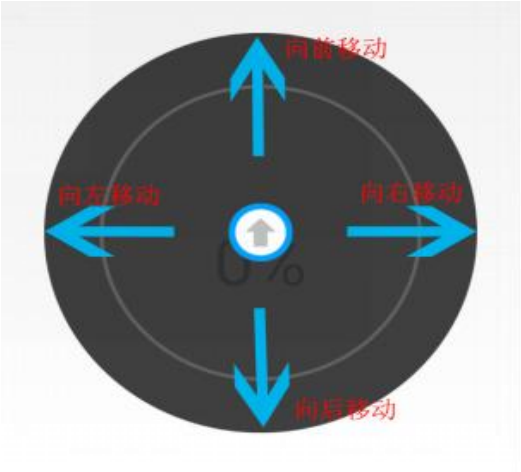
手机连接底盘路由器，然后打开 DashgAPP，选择“WIFI”便进入到 WiFi 连接界面，如下图所示：

需要输入的 Master IP 是导航模块的 IP 地址，即 192.168.31.200。

然后点击“连接”，连接成功后，界面如下：



方向的操控，如下图所示：



注意：DashgoApp 只能在启动了底盘 launch（如建图 gmapping_imu.launch）时，才可以正常连接使用，仅能控制底盘移动

2.4.2 方式二：电脑键盘控制底盘移动建图

保持建图 gmapping_imu.launch 在正常运行，在导航模块另一个终端中启动键盘控制 launch 并移动 E1 扫描地图

```
xiaobot@xiaobot:~$ ssh eaibot@192.168.31.200   //远程进入导航模块系统
eaibot@DashgoE1:~$rosrun dashgo_tools   teleop_twist_keyboard.py   //启动键盘控制移动
```

启动成功后，键盘“i”建表示前进，“，”表示后退，“j”表示左转，“l”表示右转，“k”表示停止。

保存地图

在保证建图 gmapping_imu.launch 程序正常运行，并已扫描好地图，此时进入到导航模块的地图目录 dashgo_ws/src/dashgo/dashgo_nav/maps ，并把新地图保存在此目录。

```
xiaobot@xiaobot:~$ ssh eaibot@192.168.31.200   //远程进入导航模块系统
eaibot@DashgoE1:~$roscd dashgo_nav/maps     //进入导航模块地图目录
eaibot@DashgoE1:~$rosrun map_server map_saver –f eai_map_imu   //保存地图
```

地图保存好后，ctrl+c 停止建图程序和键盘控制程序。

通过电脑远程启动 E1 导航

确保已经把建图程序停止了，确保终端是在导航模块中，（如果不是，需要远程登录到导航模块中，ssh eaibot@192.168.31.200）。

```
$ roslaunch dashgo_nav navigation_imu.launch
```

在电脑中，启动 rviz，观察地图。

```
export ROS_MASTER_URI=http://192.168.31.200:11311
roslaunch dashgo_rviz view_navigation.launch
```

---

rviz 打开后显示 E1 默认所在的位置是栅格的中心点，不一定是 E1 实际所在的位置，因此每次打开 rviz 都需要检查并设置起点位置

设置机器人起点位置：在 rviz 上，点击 2D Pose Estimate，然后根据 E1 实际位置，在地图相应位置上点击，并设置好正确方向，然后设置好机器人起点位置。

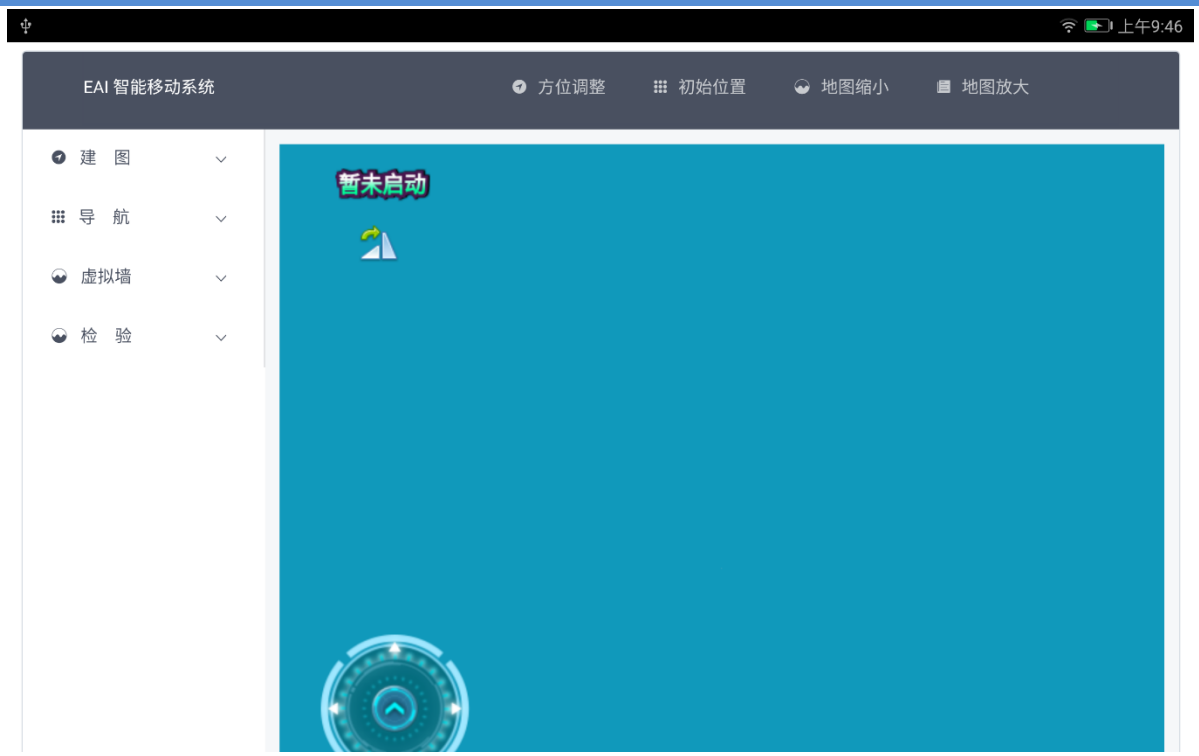
设置机器人目标点位置：在 rviz 上，点击 2D Nav Goal，然后再地图上点击目标点位置，此时正常情况，机器人会规划好到目标点的路径，并移动到目标点。

三、用 Web APP 控制 E1 移动平台建图和导航

3.1 Web APP—EGO 建图和导航

3.1.1 连接 web 控制端

wifi 连接好底盘路由器，进入浏览器，在网址栏输入 192.168.31.200:8080，打开网页。如果未启动服务，则界面提示“暂未启动”；如果启动了服务，则无需再启动服务。点击“地图缩小”和“地图放大”可以对地图进行缩放操作。按住  上下滑动，将旋转地图。



3.1.2 建图模式

如需启动建图，则展开“建图”栏，点击“启动服务”，大约 10 秒时服务启动完成并显示地图。“保存地图”需要在关闭建图服务前执行。“重置地图”可以重置当前已建的地图（此操作耗时随着所建地图的增大而变长）。

3.1.3 导航模式

如需启动导航，则展开“导航”栏，点击“启动服务”，大约 10 秒时服务启动完成并显示地图。

1、点击“添加目标”时（点击后“添加目标”将变成“操作地图”，点击“操作地图”将取消添加目标点的状态，触摸地图将平移地图），触摸地图将设置目标点，设置目标点时，轻点将设置默认方向，按住不动，目标点会旋转以选择方向。

2、点击“启动导航”，则继续上次正在进行的导航任务。

3、点击“目标列表”，可以查看已添加的点，并且能够选择排序和多点导航。

4、点击“更新列表”，将目标点的顺序回到默认状态。

5、点击“选择地图”，可以切换导航所用的地图（需要在启动导航服务之前执行）。网页连接断开时，底盘将自动停止导航。

导航模式下，点击“初始位置”，可以在地图上设置一个初始位置（轻按则默认方向，长按可以设置方向），代表底盘所在位置。然后可以展开“检验”，点击“显示激光”，看激光是否与地图轮廓相匹配，如果不匹配，原地旋转底盘 1~3 圈，校正位置（如果无法校正，则是初始位置设置相差太远，需要重新设置）。

3.2 虚拟墙

禁止区域，虚拟墙为动态障碍物，提交的虚拟墙数据将覆盖原虚拟墙数据（提交空的则清空所有虚拟墙）。可以选择画笔大小，可撤销上一步的绘制。然后提交虚拟墙。

3.3 地图编辑

选择橡皮则将擦拭的区域变白（清除静态障碍物），选择画笔可以绘制静态障碍物，橡皮擦和画笔大小可选，绘制可撤销。最后编辑完了需要进行提交（不可撤销）。

3.4 检验

点击“显示激光”，可以将激光数据显示在地图上，以便观察底盘位置是否正确。

点击“直行一米”或“旋转一周”，可以用来查看底盘硬件参数是否设置正确。误差越大，说明参数设置的越不准（比如轮子直径，激光雷达离底盘中心的水平距离和垂直距离）。

修订历史

日期	内容
2018-03-19	初稿
2018-09-06	V1.1
2018-09-16	V1.2