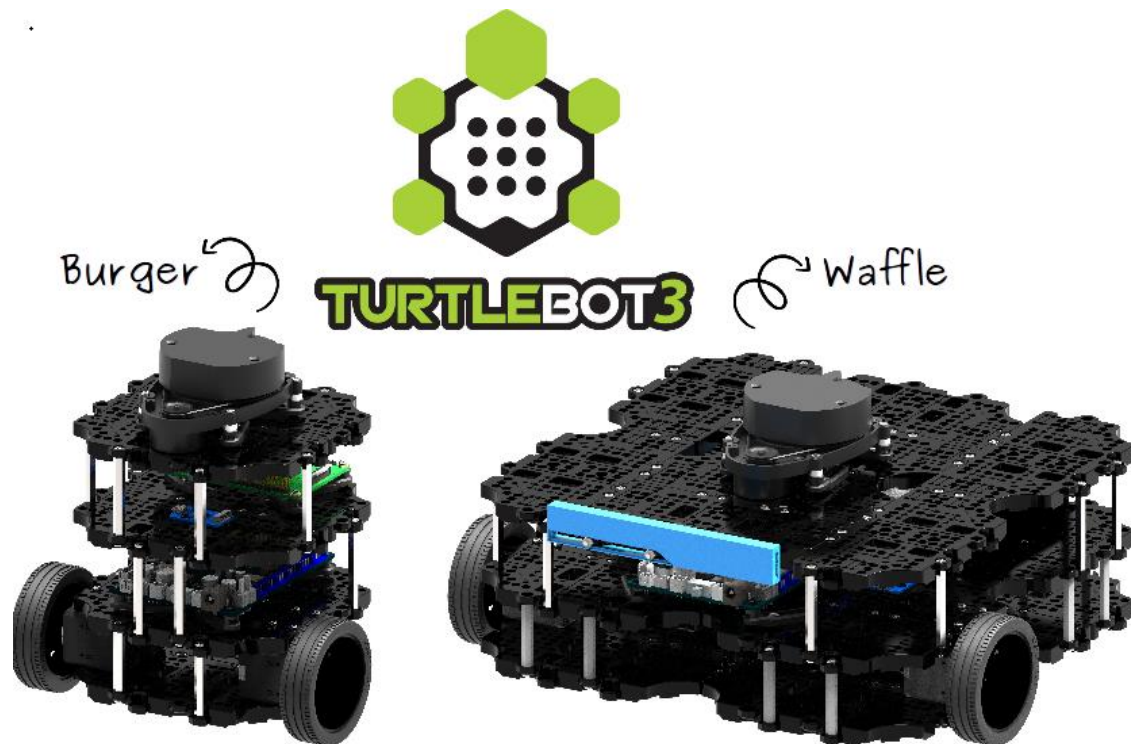


turtlebot3 入门教程（中文版）

turtlebot3 入门教程-目录



说明：

- 介绍 Turtlebo3 是 TurtleBot3 是一个小型，低成本，完全可编程，基于 ROS 的移动机器人。
- 它旨在用于教育，研究，产品原型和爱好应用的目的。
- TurtleBot3 的目标是大幅降低平台的尺寸和价格，而不会牺牲性能，功能和质量。
- 由于提供了不同可选，如底盘，计算机和传感器，TurtleBot3 可以通过各种方式进行定制。

目录：

- 一、TurtleBot3 入门教程-概述
- 二、TurtleBot3 入门教程-特点
- 三、TurtleBot3 入门教程-规格
- 四、TurtleBot3 入门教程-硬件设置
- 五、TurtleBot3 入门教程-PC 软件设置
- 六、TurtleBot3 入门教程-SBC 软件设置

七、TurtleBot3 入门教程-OpenCR 软件设置

八、TurtleBot3 入门教程-开始使用

九、TurtleBot3 入门教程-遥控

十、TurtleBot3 入门教程-SLAM

十一、TurtleBot3 入门教程-导航

十二、TurtleBot3 入门教程-仿真

十三、TurtleBot3 入门教程-朋友

十四、TurtleBot3 入门教程-OpenCR

十五、TurtleBot3 入门教程-LDS

参考：

- <http://turtlebot3.readthedocs.io/en/latest/>
- <https://github.com/ROBOTIS-GIT/turtlebot3>
- <https://github.com/ROBOTIS-GIT/OpenCR>

一、入门教程-概述

说明：

- TurtleBot3 是一个小型，低成本，完全可编程，基于 ROS 的移动机器人。
- 它旨在用于教育，研究，产品原型和爱好应用的目的。
- TurtleBot3 的目标是大幅降低平台的尺寸和价格，而不会牺牲性能，功能和质量。
- 由于提供了其他选项，如底盘，计算机和传感器，TurtleBot3 可以通过各种方式进行定制。
- TurtleBot3 意愿通过应用 SBC（单板计算机），深度传感器和 3D 打印的最新技术进步，成为创客运动的中心。

在 ROSCon2016 上介绍 Turtlebot3

- 作者：Yoonseok Pyo, Yoshihiro Shibata, Leon Jung, Darby Lim (ROBOTIS)
- PDF：http://roscon.ros.org/2016/presentations/ROSCon2016_Turtlebot3_ROBOTIS.pdf
- 视频：<https://vimeo.com/187699447>

硬件

- Turtlebot3 的默认组件如下：底盘，电机，车轮，OpenCR 板，计算机，传感器，电池。
- 底盘是华夫板，板支撑，球形脚轮等。底盘的大特征在于 Waffle 板，这是所有 TurtleBot3 部件中最大的，但是比手小。
- 该板将提供为注塑成型，并且其实现低成本，但是利用 CAD 数据来 3D 打印也一样可用。

- Turtlebot3 Basic 是一个两轮差速驱动类型的平台，但也能够支持不同结构和机械定制如：汽车，自行车，拖车等。
- CAD 数据发布到 Onshape，这是一个全云 3D CAD 编辑器。
- 通过使用计算机或甚至通过便携式设备通过 Web 浏览器访问。
- 这里允许协同完成绘图和组装工作。
- CAD 模型：
 - TurtleBot3 基本型号：<https://goo.gl/n3bGNr>
 - TurtleBot3 高级版：<https://goo.gl/wCDvVI>

软件

- 该软件以完全开源的形式提供。主要许可证是 Apache 2.0 许可证。预计全部资源发布是 2017 年第一季度。
- <https://github.com/ROBOTIS-GIT/turtlebot3>
- <https://github.com/ROBOTIS-GIT/OpenCR>

开源许可证

- 硬件许可证：
 - TurtleBot3 是一个开源硬件项目，如[开源硬件语言原则和定义 v1.0 所述](#)。
- 软件许可证：
 - 主要软件是根据 [Apache 2.0 许可证](#)发布的。然而，一些资源是根据根据许可证 [3-Clause BSD License/GPLv3](#)。
- 文件许可证：
 - 文件根据 [CC BY 4.0](#) 发布。

资源

- <http://www.turtlebot.com/>
- <http://turtlebot3.rtfid.io/>
- <http://wiki.ros.org/Robots/TurtleBot>
- https://github.com/ROBOTIS-GIT/open_manipulator
- <https://github.com/ROBOTIS-GIT/DynamixelSDK>
- <https://github.com/ROBOTIS-GIT/dynamixel-workbench>

出版物

- IEEE Spectrum: <http://spectrum.ieee.org/autaton/robotics/diy/robotis-and-osrf-announce-turtlebot-3-smaller-cheaper-and-modular>
- 3D Printing Industry:<https://3dprintingindustry.com/news/advances-robotics-made-easier-forthcoming-3d-printed-turtlebot-96844/>
- ROBOHUB: <http://robohub.org/celebrating-9-years-of-ros/>

二、入门教程-特点

说明：

- 介绍 Turtlebot3 主要特点

合作方：

TURTLEBOT3



ROBOTIS



Open Source Robotics Foundation



Onshape



世界上最受欢迎的 ROS 平台

- TurtleBot 是最受欢迎的开源机器人用于教育和研究。
- 新一代“TurtleBot3”是一种小型，低成本，完全可编程，基于 ROS 的移动机器人。
- 它旨在用于教育，研究，产品原型和爱好应用的目的。

低成本

- TurtleBot 是为了从教育和原型研究和发展的成本意识的需求而建立的。
- TurtleBot3 是配备了通用 360 度 LiDAR 的 SLAM 移动机器人中最经济的机器人。

小尺寸

- TurtleBot3 Basic 的尺寸为 140mm x 140mm x 150mm（长 x 宽 x 高）。
- 它的大小是 TurtleBot1 和 2 的 1/4 大小，甚至可以在背包里携带。

ROS 标准

- TurtleBot 品牌由 Open Source Robotics Foundation, Inc. (OSRF)管理，OSRF 开发和管理 ROS。如今，ROS 已经成为世界各地所有机器人的向往参与的平台。TurtleBot 可以集成到现有的基于 ROS 的机器人，同时也是学习 ROS 的经济实惠的平台。

结构可扩展性

- TurtleBot3 鼓励用户使用一些替代选项自定义其机械结构：开源嵌入式板（作为控制板），计算机和传感器。
- Turtlebot3 Basic 是一个两轮差速驱动类型的平台，但能够在许多方面比如结构和机械定制：汽车，自行车，拖车等。
- 延伸想象之外的想法。

移动机器人的模块化执行器

- TurtleBot3 允许通过使用 2 个 Dynamixel 在车轮关节上获得精确的空间数据
- Dynamixel X 系列可以通过以下 6 种操作模式之一进行操作：
 - 车轮的速度控制模式
 - 扭矩控制模式
 - 关节的位置控制模式等
- Dynamixel 甚至可用于制作移动操纵器，因为它轻巧，但可以用速度，扭矩和位置精确控制
- Dynamixel 是使 TurtleBot 更完善的核心组件。

开源 ROS 的控制板

- 开源控制板 OpenCR，它是软硬件都开源的，适用于 ROS 通信的控制板。
- 它具有不仅支持控制 Dynamixel 而且还支持基本识别任务的 ROBOTIS 传感器，如触摸传感器，红外传感器，彩色传感器等。
- 它在板内有一个 IMU 传感器，以便它可以加强许多精确的控制。
- 该板具有 3.3V，5V，12V 电源，以加强可用的计算机设备阵容。

强大的传感器

- TurtleBot3 Basic 使用 360°LiDAR。
- TurtleBot3 高级型号配备 360°激光雷达，但另外还提供了一个功能强大的英特尔®实感™与识别 SDK。
- 这将是制作移动机器人的最佳解决方案。

开放源码

- TurtleBot3 的硬件，固件和软件是作为开源提供的。
- 基本上，TurtleBot3 的所有组件都将作为注塑成型提供，并且它实现了低成本，但是也提供用于 3D 打印的 CAD 数据。
- CAD 数据发布到 Onshape，这是一个全云 3D CAD 编辑器。通过使用计算机或甚至通过便携式设备通过 Web 浏览器访问。这里允许协同完成绘图，组装工作。

- 此外，OpenCR 板的所有细节，包括电路图，PCB Gerber，BOM 和固件源也在开源许可下，针对 ROS 用户和社区开源。

三、入门教程-规格

说明：

- 介绍 Turtlebot3 规格

Turtlebot 图：



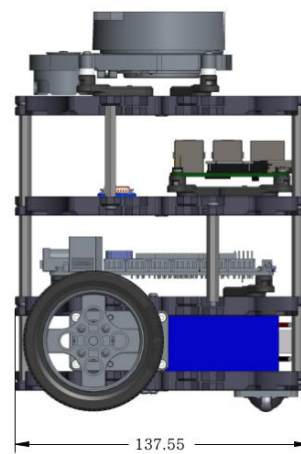
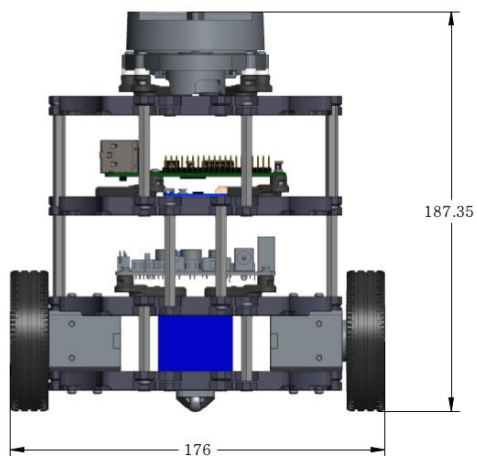
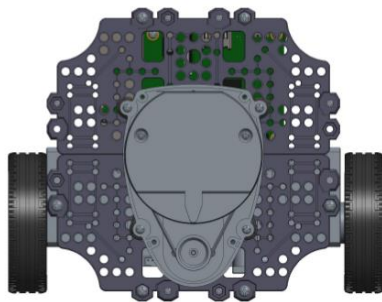
规格

项目	基本	高级
最大平移速度	0.22m / s	0.26m / s
最大转速	2.84rad / s (162.72deg / s)	1.82rad / s (104.27deg / s)
最大有效载荷	15kg	30kg
尺寸 (长 x 宽 x 高)	176mm x 138mm x 188mm	306mmx283mmx143mm
重量 (+ SBC +电池+传感器)	0.995kg	1.745kg
攀登门槛	10mm 以下	
预计运行时间	2h 30m	2h
预计充电时间	2h 30m	
PC 连接	USB	
IMU	陀螺仪 3 轴	
	加速度计 3 轴	
	磁力计 3 轴	
	3.3V / 800mA	
电源连接器	5V / 2A	
	12V / 1A	

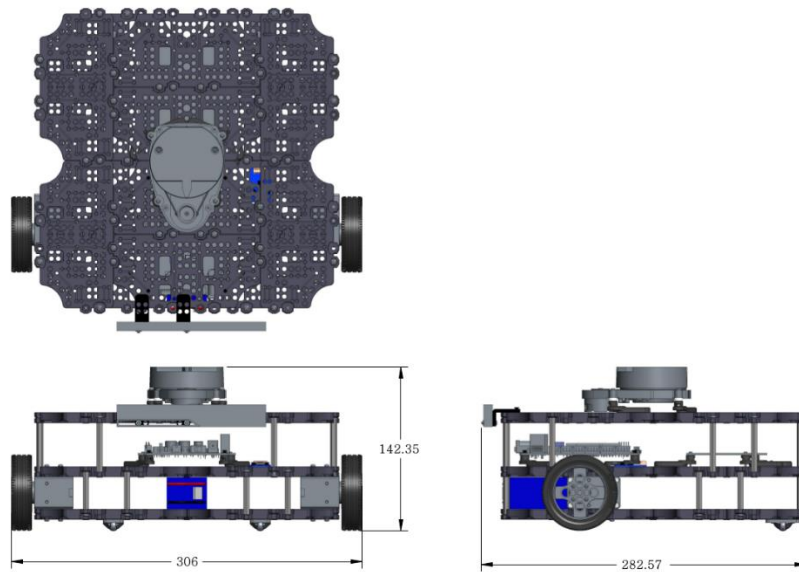
插脚	GPIO 18 引脚 Arduino 32 针
音频	几个可编程蜂鸣声序列
可编程 LED	用户 LED x 4 板状态 x 1
状态 LED	电池状态 x 1 Arduino LED x 1
纽扣	按钮 x 2
电池	锂聚合物 11.1V 1800mAh / 19.98Wh 5C
固件升级	通过 USB /通过 JTAG
充电适配器	输入：100-240V，AC 50 / 60Hz，1.5A @ max 输出：12V DC，5A

尺寸

- 尺寸数据（基本）



- 尺寸数据（高级）



组件

- SBC :
 - Intel® Joule™ : <http://ark.intel.com/products/96414/Intel-Joule-570x-Developer-Kit>
 - Raspberry Pi 3 Model B : <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>
- 传感器 :
 - Laser Distance Sensor :

图示：



概述

- HLS-LFCD LDS 用于 TurtleBot3 的两种型号。
- LDS（激光距离传感器）是将由障碍物检测收集的数据发送到用于 SLAM 技术的主机的传感器

基本性能规格

项目	规格
工作电源电压	5V DC±5%
光源	半导体激光二极管（λ= 785nm）
激光安全	IEC60825-1 Class 1
目前的消费	400mA 以下（冲击电流 1A）
检测距离	120mm～3,500mm
接口	3.3V USART（230,400 bps）每 6 度 42bytes，全双工选项
环境光电阻	10,000 lux 或更小
采样率	1.8kHz
尺寸	69.5（W）×95.5（D）×39.5（H）mm
质量	低于 125g

测量性能规格

项目	规格
距离范围	120～3500mm
距离精度（120mm～499mm）	±15mm
距离精度（500mm～3,500mm）	±5.0%
距离精度（120mm～499mm）	±10mm
距离精度（500mm～3,500mm）	±3.5%
扫描速率	300±10rpm
角范围	360°
角分辨率	1°

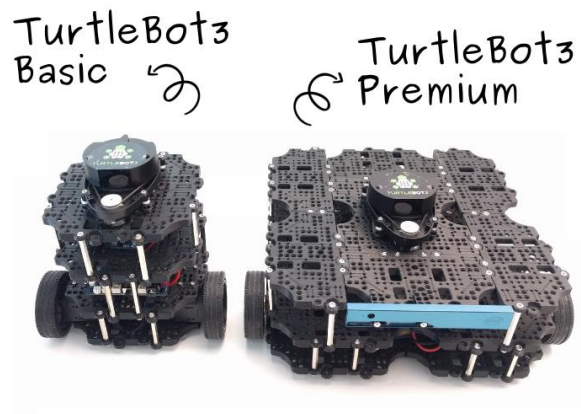
规范文档

- 包括内容如基本性能，测量性能，机构布局，光路，数据信息，引脚描述，命令。

在 TurtleBot3 使用 LDS

- HLS-LFCD LDS 用于 TurtleBot3 Basic 和 TurtleBot3 Premium。

- 图示：



- Intel® Realsense™ R200 : <https://software.intel.com/en-us/RealSense/R200Camera>
- 控制板
 - OpenCR :

图示：



概述

- OpenCR 是 TurtleBot3 的主控制器板。
- OpenCR 或 ROS 的开源控制模块是为 ROS 嵌入式系统开发的，提供了完整的开源硬件和软件。
- Board 的所有内容包括 Schematics, PCB Gerber, BOM 和 TurtleBot3 的固件源可以根据开源许可证免费分发给用户和 ROS 社区。
- STM32F7 系列是 OpenCR 板内的主芯片，具有非常强大的 ARM Cortex-M7 浮点单元。
- OpenCR 的开发环境从支持为年轻学生的 Arduino IDE 和 Scratch 到传统固件开发环境的专家。
- 该板提供一组数字和模拟输入/输出引脚，可从 pne 电路连接到另一个或内置 IMU 传感器。
- 该板的通信接口包括与 PC 通信的 USB 和用于其他嵌入式设备的 UART, SPI, I2C, CAN。
- 要使用 SBC, OpenCR 板可以提供最佳解决方案。
- 它支持一些电源输出：12V, 5V, 3.3V 的 SBC 和传感器。
- 它还在两个外部电源输入中具有热插拔功能：电池和 SMPS。

规格

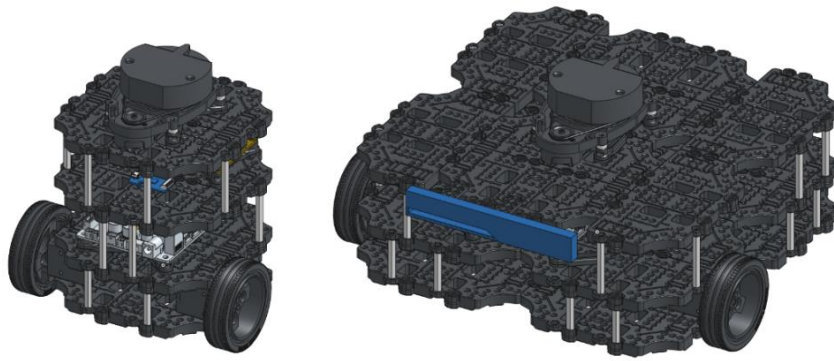
项目	规格
微控制器	STM32F746NGH6 /具有 FPU 的 32 位 ARMCortex®-M7（216MHz，462DMIPS）
传感器	陀螺仪 3Axis，加速度计 3Axis，磁力计 3Axis（MPU9250）
程序员	ARM Cortex 10pin JTAG / SWD 连接器

	USB 设备固件升级（DFU）
	串行
扩展引脚	32 引脚（L 14, R 18）* Arduino Uno Revision 3 连接
	传感器 x 4 针
	扩展连接器 x 18 针
通信电路	USB（Micro-B USB 连接器/ USB 2.0 /主机/ Peripehral / OTG）
	TTL（JST 3pin / Dynamixel）
	RS485（JST 4pin / Dynamixel）
	UART x 2
	CAN
LED 和按钮	LD2（红/绿）：USB 通信
	用户 LED x 4: LD3（红色），LD4（绿色），LD5（蓝色）
	用户按钮 x 2
权力	外部输入源
	5 V（USB VBUS），7-24 V（电池或 SMPS）
	默认电池: LI-PO 11.1V 1,800mAh 19.98Wh
	默认 SMPS: 12V 5A
	外部输出源
	12V@1A, 5V@4A, 3.3V@800mA
	电源 LED: LD1（红色，3.3 V 电源打开）
	复位按钮 x 1（用于板的电源复位）
	电源开关 x 1
尺寸	105（W）x75（D）mm
质量	60g

- 从“shore power”（12V, 5A SMPS）切换到“移动电源”（电池）的热插拔：电源板支持不间断电源（UPS）类型的功能。
- 执行器
 - Dynamixel X series : http://en.robotis.com/index/product.php?cate_code=10121110

四、入门教程-硬件设置

硬件图：



组件

- TurtleBot3 有两种不同的型号：Basic 和 Premium。
- 两种型号之间的巨大差异是电机，SBC（单板计算机）和传感器。
- 以下列表显示了其组件:

每个模型的部件数量		基本 高级	
机壳	Waffle 板	8	24
	35mm 板支架	4	12
	45mm 板支架	12	10
	板支持	12	12
	链轮	2	2
	橡胶轮胎	2	2
	球脚轮	1	2
	钢球	1	2
	螺栓套	1	1
	螺母组	1	1
	铆钉	6	10
	铆钉间隔	4	4
	梦幻 L 支架		2
	XL430-W350-T	2	
	发动机 XM430-W210-T		2
控制器	Horn for XM430-W210-T		2
	OpenCR	1	1
	SMPS 12V 5A	1	1
	AC 代码	1	1
功率	LIPO 电池 11.1V 1800mAh	1	1
电池	LIPO 电池充电器	1	1
电缆	电池转换电缆	1	1
	RaspberryPi 电源（5V）电缆	1	
	焦耳电源（12V）电缆		1
通讯	机器人电缆-X3P 100mm	1	

电缆	机器人电缆-X3P 180mm	1	1
	机器人电缆-X3P 240mm		1
	USB2.0 到 microB 电缆	2	2
SBC	RaspberryPi 3 型号 B	1	
	英特尔®Joule™570x		1
传感器	激光距离传感器	1	1
	英特尔®RealSense™R200		1
记忆	MicroSD 卡 8G	1	
	螺丝刀	1	1
	铆接工具	1	1
工具	Velcro 电池	1	1
	USB2LDS	1	1
	USB3.0 集线器		1

装配

- 每个 TurtleBots 在箱子里都没有组装。按照说明组装 TurtleBot3。

五、入门教程-PC 软件设置

说明：

- 介绍如何在 PC 上安装相关系统和软件。
- 测试的系统版本是 Ubuntu 16.04.1， ROS 版本是 Kinetic Kame

安装 Ubuntu

- 在远程 PC（台式机或笔记本电脑）中安装
- 系统版本 Ubuntu 16.04.1:
 - <https://www.ubuntu.com/download/desktop>
 - <https://www.ubuntu.com/download/desktop/install-ubuntu-desktop>

安装 ROS 版本 kinetic 和相关包:

```
wget https://raw.githubusercontent.com/oroca/oroca-ros-pkg/kinetic/ros_install.sh && chmod
755 ./ros_install.sh && bash ./ros_install.sh catkin_ws kinetic
```

- 详细安装过程：<http://wiki.ros.org/kinetic/Installation/Ubuntu>

安装 TurtleBot3 及依赖包：

- 依赖包：

```
sudo apt-get install ros-kinetic-joy ros-kinetic-teleop-twist-joy ros-kinetic-teleop-twist-
keyboard ros-kinetic-laser-proc ros-kinetic-rgbd-launch ros-kinetic-depthimage-to-laserscan
ros-kinetic-rosserial-arduino ros-kinetic-rosserial-python ros-kinetic-rosserial-server ros-
```

kinetic-rosserial-client ros-kinetic-rosserial-msgs ros-kinetic-amcl ros-kinetic-map-server
ros-kinetic-move-base ros-kinetic-hls-lfcd-lds-driver ros-kinetic-urdf ros-kinetic-xacro
ros-kinetic-gmapping ros-kinetic-turtlebot-teleop

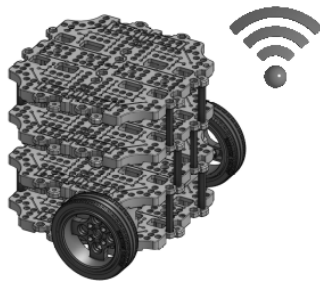
- turtlebot3

```
cd ~/catkin_ws/src/  
git clone https://github.com/ROBOTIS-GIT/turtlebot3.git  
cd ~/catkin_ws && catkin_make
```

- 如果 catkin_make 完成没有任何错误，使用 TurtleBot3 的准备将完成。

网络配置

[TurtleBot]



```
ROS_MASTER_URI = http://IP_OF_PC:11311  
ROS_HOSTNAME   = IP_OF_TURTLEBOT
```

[Remote PC]



```
ROS_MASTER_URI = http://IP_OF_PC:11311  
ROS_HOSTNAME   = IP_OF_PC
```

* ROS Master is running in Remote PC.

- ROS 需要 IP 地址在 turtlebot 和远程 PC 之间进行通信
- 分别在 turtlebot 和 PC，执行如下命令获得对应的 IP 地址：

ifconfig

- 修改.bashrc

```
gedit ~/.bashrc
```

- TURTLEBOT 配置如下：

```
ROS_MASTER_URI = http://IP_OF_PC:11311
```

```
ROS_HOSTNAME = IP_OF_TURTLEBOT
```

- PC 配置如下：

```
ROS_MASTER_URI = http://IP_OF_PC:11311
```

```
ROS_HOSTNAME = IP_OF_PC
```

- ROS_MASTER 运行在远程 PC 上。
- 让环境生效：

```
source ~/.bashrc
```

六、入门教程-SBC 软件设置

简便安装：

- 在 Raspberry Pi 3 (TurtleBot3 Basic) 通过镜像安装 Ubuntu MATE 和 Turtlebot3
- 利用 SDcard 读卡器来安装 TurtleBot3 Basic 映像
- SD 卡的容量应大于 8 GB，以便安装 TurtleBot3 Basic 映像
- 磁盘映像文件包含 Ubuntu MATE 16.04.1 和 ROS kinetic kame
- Raspberry Pi 3 镜像下载：<https://goo.gl/uOvWLh>
- 镜像安装方法：

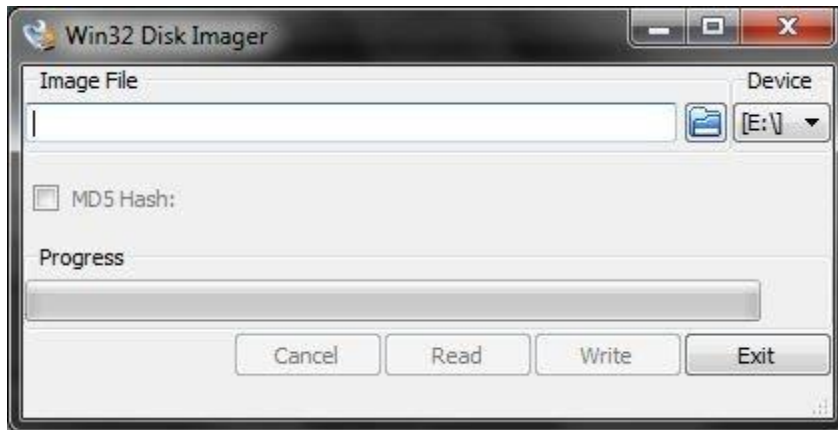
安装前准备

- 镜像系统 [官网下载](#)
- 下载 Win32DiskImager 软件并安装
- 一张 Micro SD 卡
- 一块树莓派 3 开发板
- 树莓派 3 电源
- HDMI 线
- 支持 HDMI 的显示器或电视机

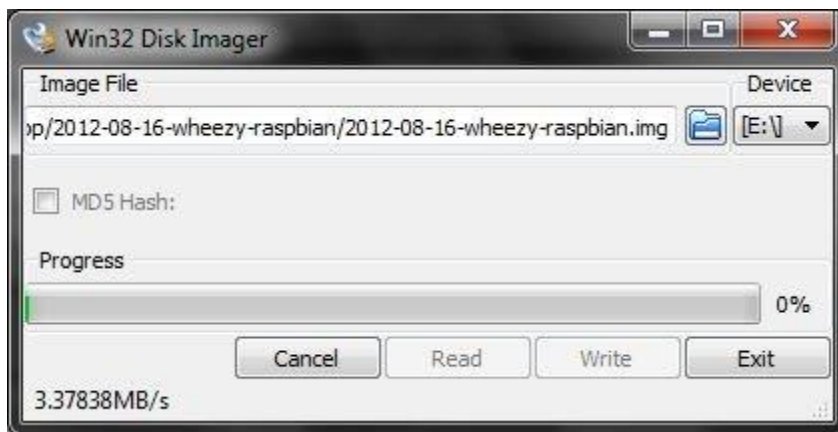
- 一条网线（可选）
- 键盘
- 鼠标
- 电脑，安装 Windows 系统

系统安装步骤

1. 格式 Micro SD 卡为 FAT32 格式
2. 启动 Win32DiskImager



3. Image File 处选择 Raspbian 映像文件
4. Device 处选择盘符为你读卡器的盘符
5. 点 Write，然后点一下 Yes 确定操作，开始系统写入



6. 写入完成，提示成功



7. Micro SD 卡插入树莓派，接通电源启动

树莓派 3 手工安装：

- 适用在树莓派 3 上安装 (TurtleBot3 Basic)

(1) 安装 Ubuntu MATE for Raspberry Pi 3

- 下载 Raspberry Pi 3 版本的 Ubuntu MATE 16.04.1。
- 下载地址：<https://ubuntu-mate.org/download/>
- 下载镜像：<https://ubuntu-mate.org/raspberry-pi/>
- 安装镜像：参见上文

(2) 为 Raspberry Pi 3 安装 TurboBot3 依赖的 ROS 包

- 安装依赖包：

```
sudo apt-get install ros-kinetic-amcl ros-kinetic-rosserial ros-kinetic-map-server ros-kinetic-move-base
```

- 安装 ROS：

```
wget https://raw.githubusercontent.com/oroca/oroca-ros-pkg/kinetic/ros_install.sh && chmod 755 ./ros_install.sh && bash ./ros_install.sh catkin_ws kinetic
```

- 或者采用官方指南：<http://wiki.ros.org/kinetic/Installation/Ubuntu>
- 安装 turtlebot 依赖：

```
sudo apt-get install ros-kinetic-joy ros-kinetic-teleop-twist-joy ros-kinetic-teleop-twist-keyboard ros-kinetic-laser-proc ros-kinetic-rgbd-launch ros-kinetic-depthimage-to-laserscan ros-kinetic-rosserial-arduino ros-kinetic-rosserial-python ros-kinetic-rosserial-server ros-kinetic-rosserial-client ros-kinetic-rosserial-msgs ros-kinetic-amcl ros-kinetic-map-server ros-kinetic-move-base ros-kinetic-hls-lfcd-lds-driver ros-kinetic-urdf ros-kinetic-xacro ros-kinetic-turtlebot-teleop ros-kinetic-compressed-image-transport ros-kinetic-rqt-image-view
```

- 安装 turtlebot

```
git clone https://github.com/ROBOTIS-GIT/turtlebot3.git
```

```
cd ~/catkin_ws && catkin_make
```

- 如果 catkin_make 完成没有任何错误，使用 TurtleBot3 的准备将完成。
- USB 设置：以下允许将 USB 端口用于没有 root 权限的 OpenCR 板

```
wget https://raw.githubusercontent.com/ROBOTIS-GIT/OpenCR/master/99-opencr-cdc.rules
```

```
sudo cp ./99-opencr-cdc.rules /etc/udev/rules.d/
```

```
sudo udevadm control --reload-rules
```

英特尔®Joule™手工安装：

- 适用在英特尔®Joule™安装（TurtleBot3 Premium）

(1) 安装 Ubuntu 的英特尔®Joule™（TurtleBot3 Premium 型号）

- 下载英特尔®Joule™的 Ubuntu 16.04 版本的映像。
 - 镜像地址：<https://developer.ubuntu.com/core/get-started/intel-joule#alternative-install:-ubuntu-desktop-16.04-lts>
- 制作可启动的 USB 驱动器来安装 Ubuntu：
 - <https://software.intel.com/en-us/node/705675#ubuntu>
 - <https://software.intel.com/en-us/node/700692>

(2) 安装 ROS 和软件包

- 参考树莓派 3 的安装 ROS 和软件包方法

七、入门教程-OpenCR 软件设置

OpenCR

- OpenCR 根据 SBC 的指令控制 Dynamixel。为此，应在板中构建特定的固件。请参阅说明并配置设置。

OpenCR 的 ArduinoIDE 设置

- 按照说明在远程 PC 上获得 OpenCR Arduino 开发环境

USB 端口设置

- 使 OpenCR USB 端口能够在没有 root 权限的情况下上传 Arduino IDE 程序。
- 执行如下命令：

```
wget https://raw.githubusercontent.com/ROBOTIS-GIT/OpenCR/master/99-opencr-cdc.rules
```

```
sudo cp ./99-opencr-cdc.rules /etc/udev/rules.d/
```

```
sudo udevadm control --reload-rules
```

编译器设置

- 由于 OpenCR 库是为 32 位平台构建的，64 位 PC 需要用于 Arduino IDE 的 32 位编译器相关内容：
- 执行如下命令：

```
sudo apt-get install libncurses5-dev: i386
```

安装 Arduino IDE

- OpenCR 需 Arduino IDE 1.16.0 及之后的版本
- 从官方的 arduino 主页下载最新版本，地址：<https://www.arduino.cc/en/Main/Software>
- 将下载的文件解压缩到所需的文件夹比如用户的根目录下 tools，并从终端执行安装文件。如下：

```
mkdir ~/tools
```

```
wget
```

```
cd ~/tools/arduino-1.16.0
```

```
./install.sh
```

- 增加路径到环境中：

```
gedit ~/.bashrc
```

```
export PATH=$PATH:$HOME/tools/arduino-1.16.0
```

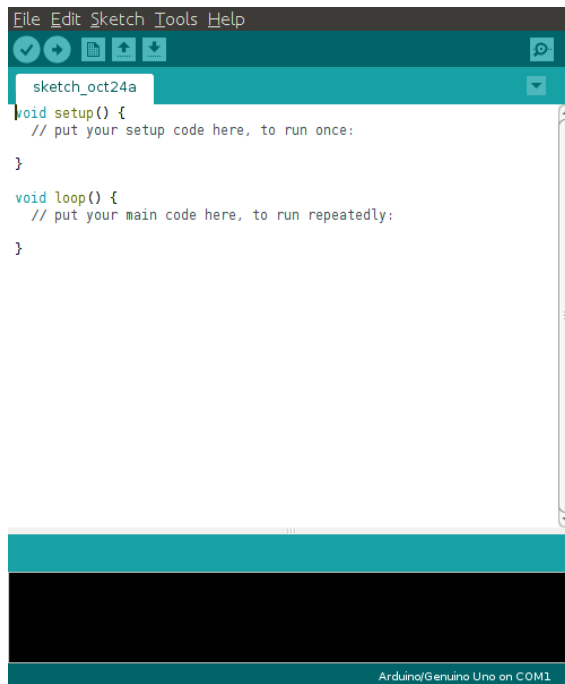
```
source ~/.bashrc
```

运行 Arduino IDE

- 在 linux 下，新终端执行：

```
arduino
```

- 图示：



将 OpenCR 板移植到 Arduino IDE

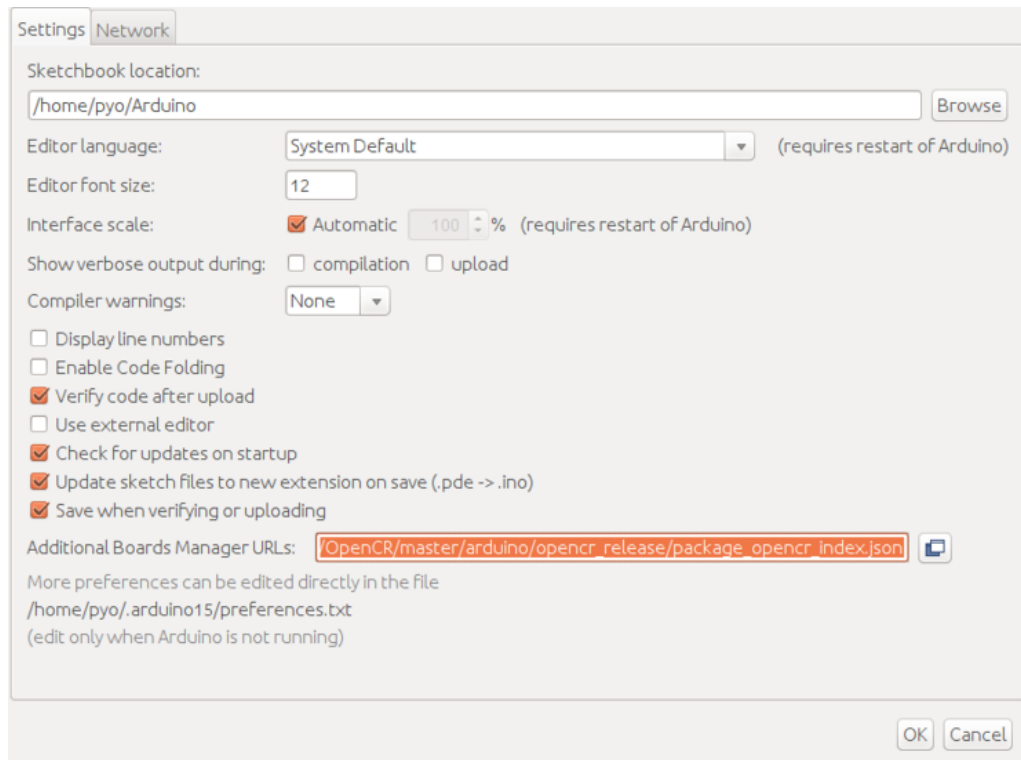
(1) 增加 Boards Manager

- 运行 IDE , 点击 File → Preferences, 复制如下内容到 Additional Boards Manager URLs :

[https://raw.githubusercontent.com/ROBOTIS-](https://raw.githubusercontent.com/ROBOTIS-GIT/OpenCR/master/arduino/opencr_release/package_opencr_index.json)

[GIT/OpenCR/master/arduino/opencr_release/package_opencr_index.json](https://raw.githubusercontent.com/ROBOTIS-GIT/OpenCR/master/arduino/opencr_release/package_opencr_index.json)

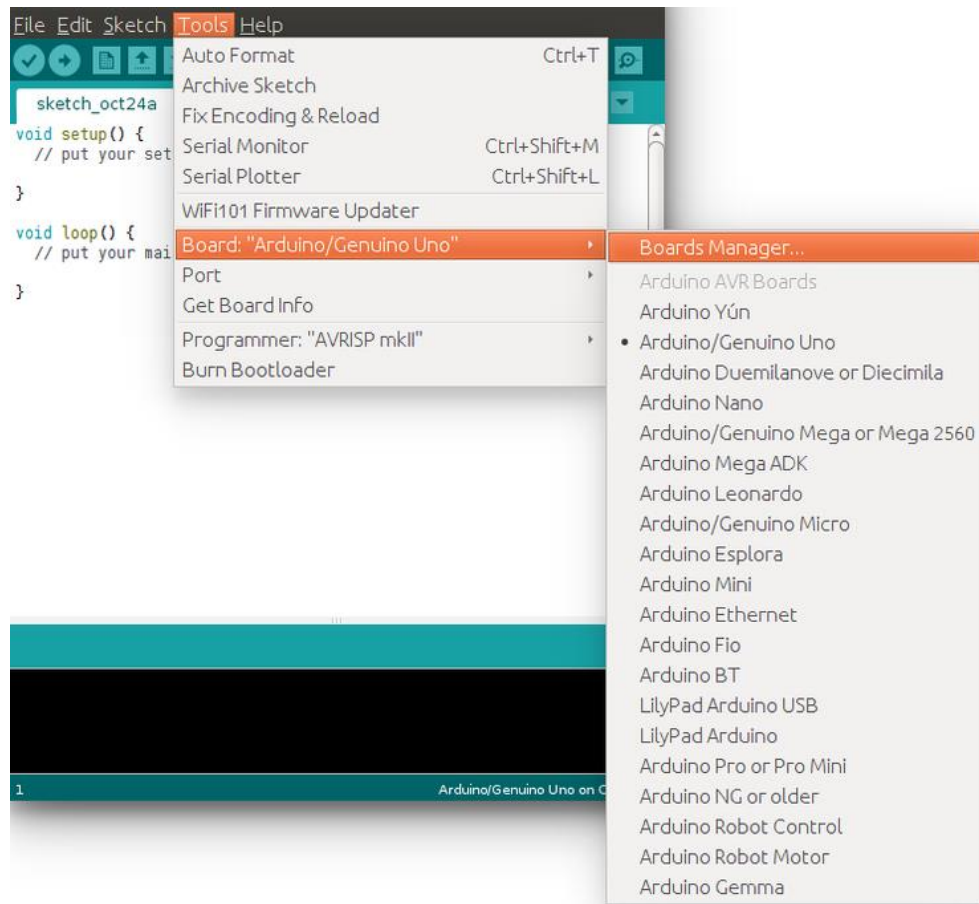
- 大概需要 20 多分钟完成。
- 如图：



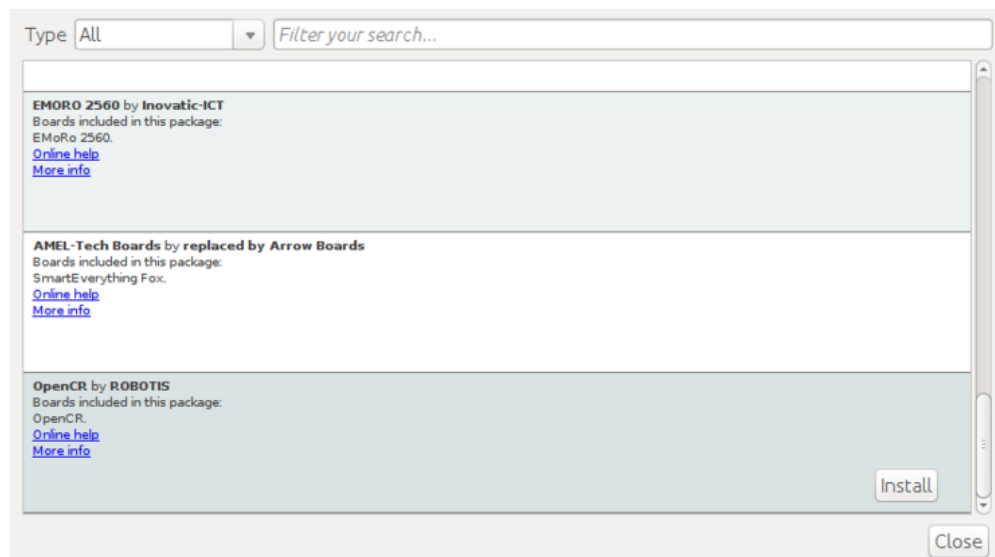
(2) 通过 Boards Manager 安装 OpenCR 软件包

- 点击 Tools → Board → Boards Manager.

- 图示：

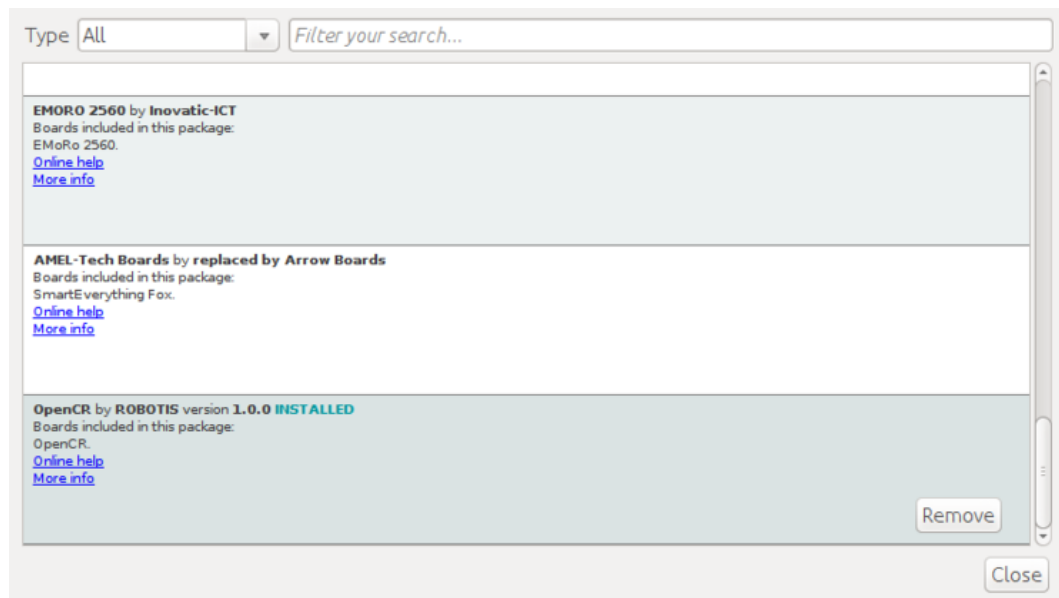


- 在文本框中键入 OpenCR 以查找包。找到 OpenCR by ROBOTIS 后，点击 Install.
- 图示：

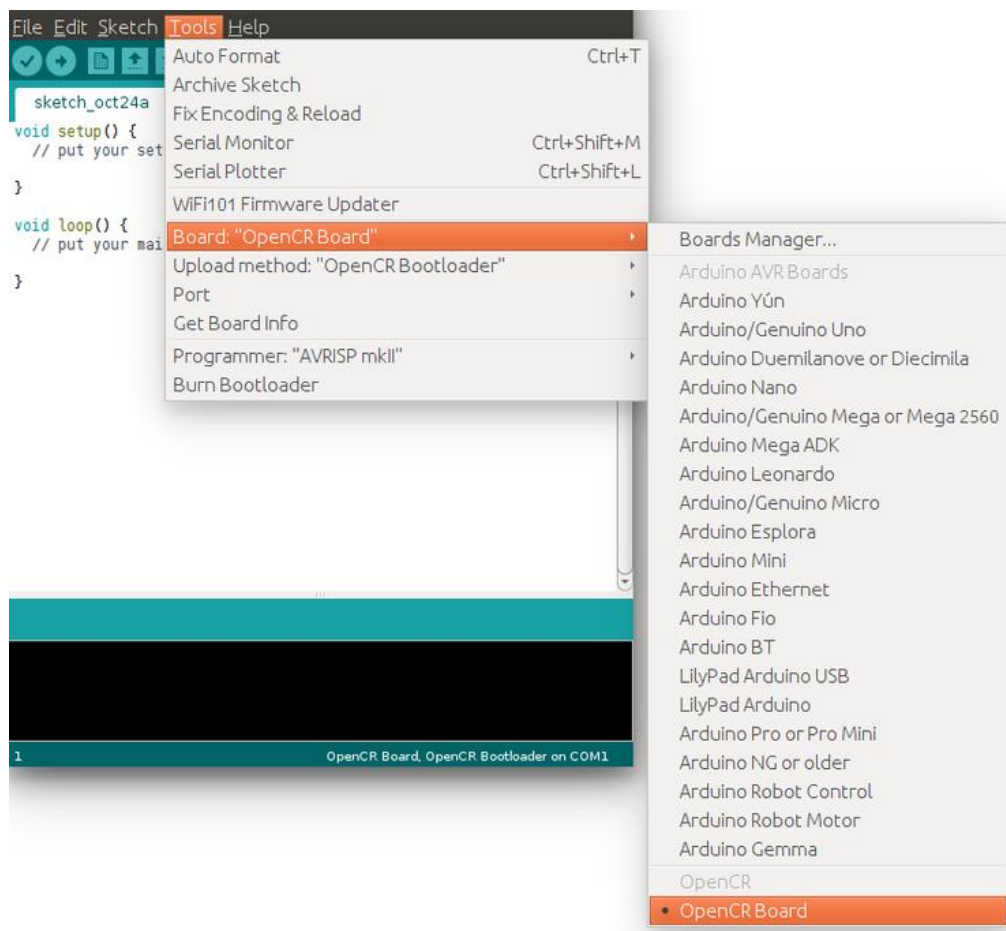


- 安装后，将显示 “INSTALLED”

- 图示：



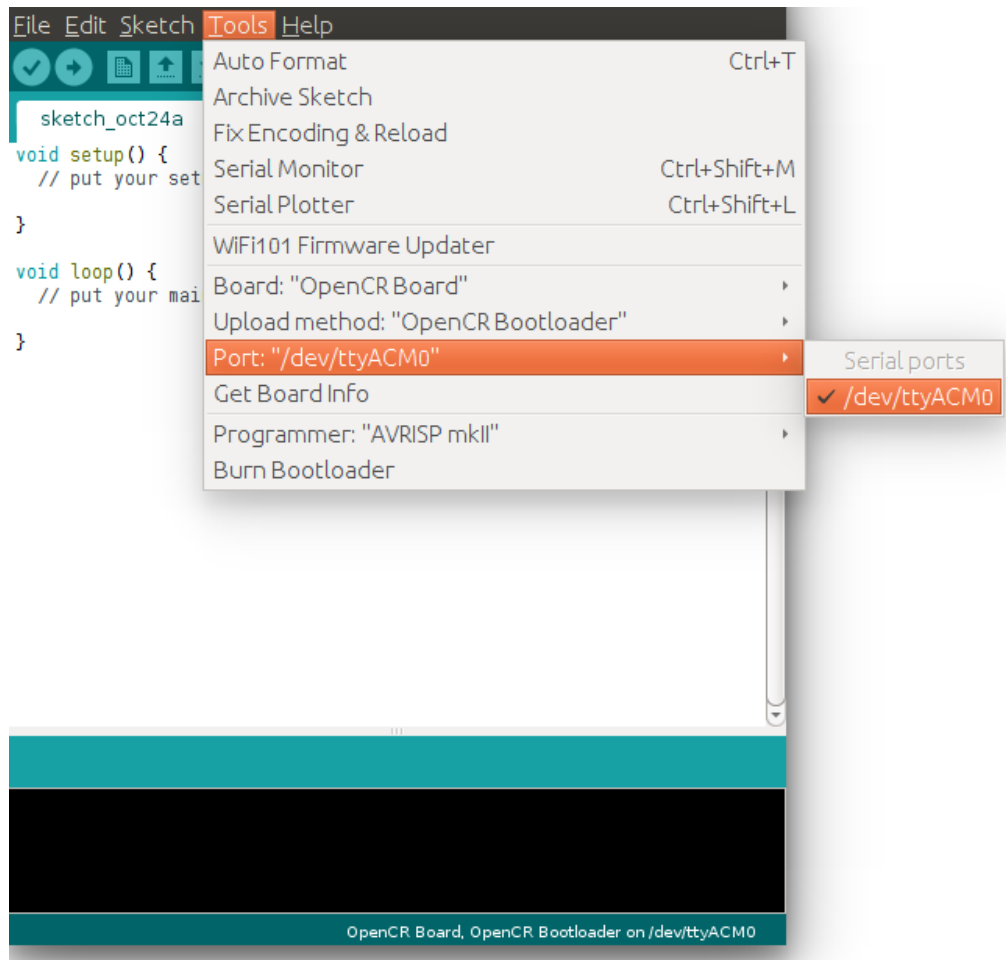
- OpenCR Board 列在 Tools → Board 中。
- 图示：



(3) 端口设置

- 此步骤显示程序上传的端口设置。

- OpenCR 应通过 USB 端口连接到 PC 和 OpenCR 板。
- 选择 Tools → Port → /dev/ttyACM0.
- 图示：



- /dev/ttyACM0 的值可能会根据不同 PC 环境有差异。

(4) 调制解调器删除

- 在使用 Arduino IDE 编程并将程序上传到 OpenCR 后，OpenCR 将重新启动并重新连接。
- 同时，Linux 的调制解调器相关软件包将发送 AT 命令来管理设备。
- 因此显示 OpenCR 上的连接错误，因此此步骤应该预先完成。

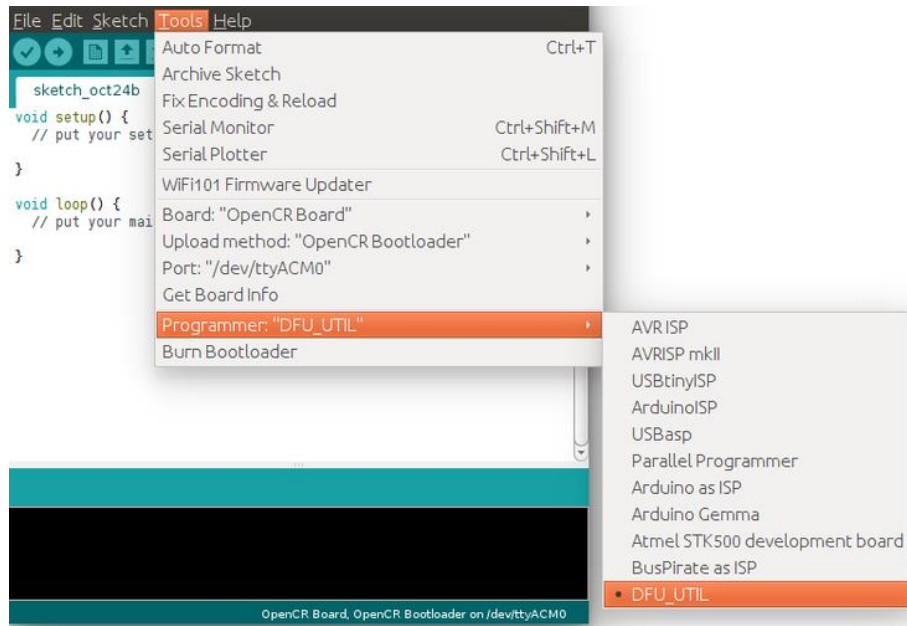
```
sudo apt-get purge modemmanager
```

(5) Bootloader 写入

- STM32F7xx 用于 OpenCR 板上的主 MCU，支持 DFU（设备固件升级）
- 这使 MCU 的内置引导加载程序本身能够通过使用 USB 引导 DFU 协议，主要用于引导加载程序初始化，恢复模式和引导加载程序更新
- 最大的优点是让用户能够使用 USB 引导加载程序，无需其他 JTAG 设备
- 使用嵌入在 MCU 中的 DFU 模式写入固件，而无需编写/调试设备，例如 STLink。

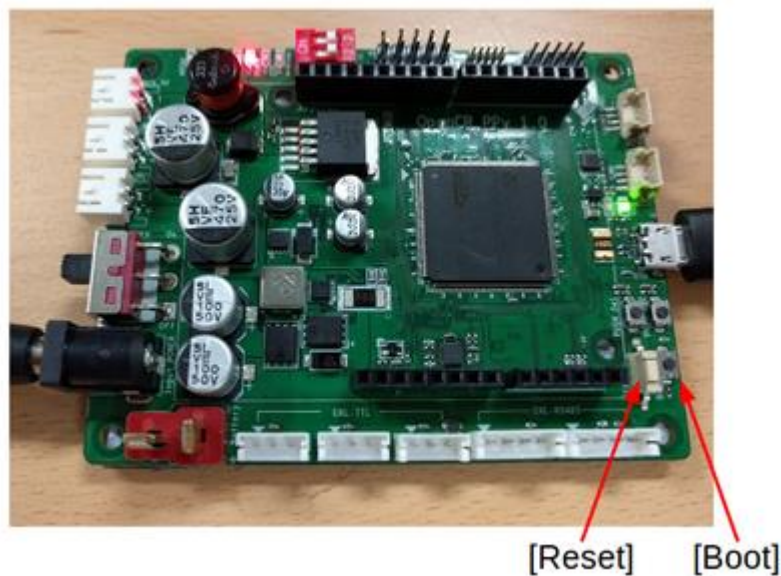
(6) Programmer 设置

- 选择 Tools → DFU-UTIL
- 图示：



(7) 运行 DFU 模式

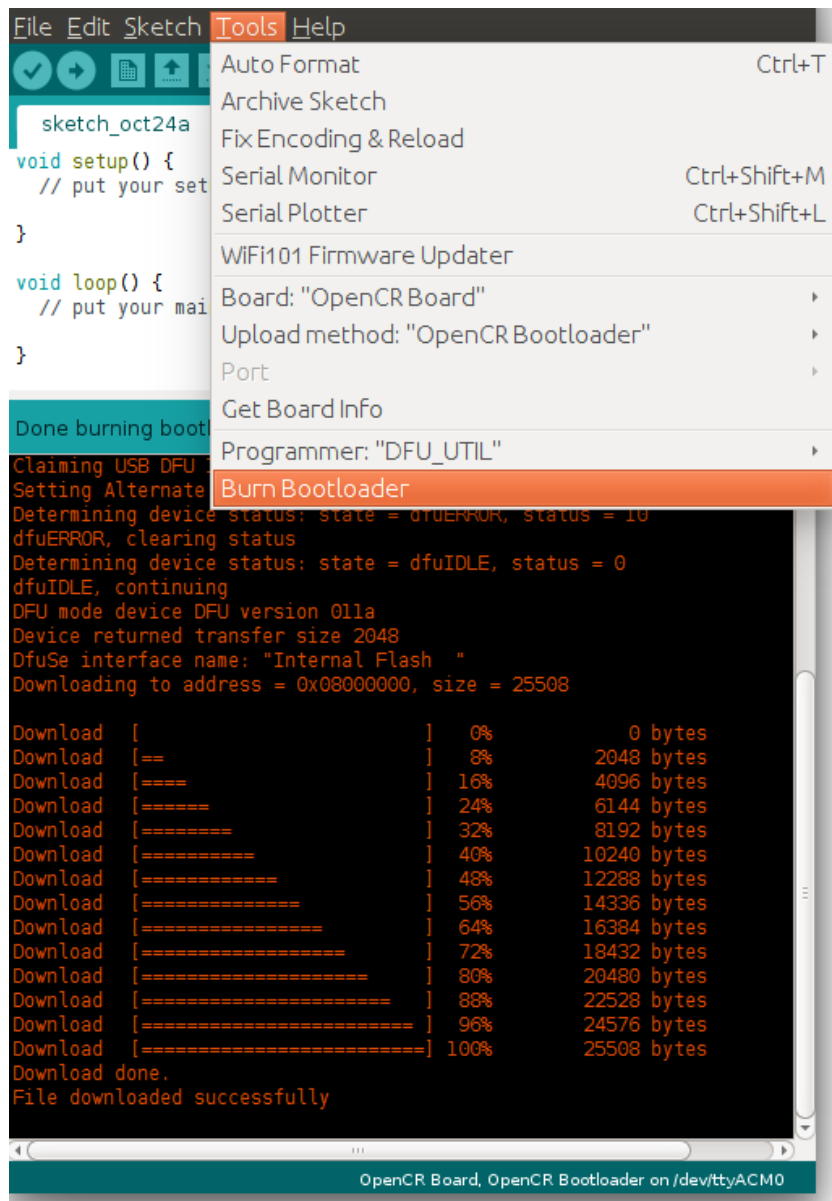
- 按下 Reset 同时按下 Boot 键，就能激活 DFU 模式
- 图示：



(8) 下载 bootloader

- 点击 Tools → Burn Bootloader

- 图示：



OpenCR 固件的 ROS 设置

- 将 TurtleBot3 固件添加到 OpenCR 中
- 用于 ROS 的 OpenCR 固件（或源）是控制 ROS 中的 Dynamixel 和传感器。
- 固件位于 OpenCR 示例中，也由 board manager 下载。
- 初级版本：点击 File → Examples → turtlebot3 → turtlebot_basic → turtlebot3_core
- 高级版本：点击 File → Examples → turtlebot3 → turtlebot_premium → turtlebot3_core
- 点击 Upload 上传到 OpenCR 板
- 图示：





八、入门教程-开始使用

启动 TurtleBot3

- [Remote PC]运行 ROScore

roscore

- [TurtleBot3 SBC]给 LiDAR 连接到 ttyUSB0 套接字的读/写权限

sudo chmod a+rw /dev/ttyUSB0

- [TurtleBot3 SBC]启动 launch 文件

roslaunch turtlebot3_bringup turtlebot3_robot.launch

- 如果终端显示 lost sync with device 错误信息，则说明 TurtleBot3 的传感器设备必须未完全连接

- 现在，用各种远程操作方法测试 TurtleBot3

九、入门教程-远程操作

远程操作

- 当在 TurtleBot3 的 SBC 上操作时，以下说明是无用的。
- 在远程 PC 上执行此操作。
- TurtleBot3 将由各种设备远程操作。
- 我们使用几种无线设备（例如 PS3, XBOX 360, ROBOTIS RC100 等）对其进行了测试。
- 该示例由 ROS 在 Ubuntu mate 16.04 上使用 Raspberry Pi 3（除了通过 LEAP Motion 测试）和 OpenCR 控制 Dynamixel XM-430。

键盘控制

- [Remote PC] 安装软件包

```
sudo apt-get install ros-kinetic-teleop-twist-keyboard
```

- [Remote PC] 启动进行简单远程操作测试

```
roslaunch turtlebot3_bringup turtlebot3_teleop_key.launch
```

- [Remote PC] 如果文件成功启动，终端将显示以下内容。

```
Control Your Turtlebot!
```

```
-----
```

```
Moving around:
```

```
u      i      o
```

```
j      k      l
```

```
m      ,      .
```

```
q/z : increase/decrease max speeds by 10%
```

```
w/x : increase/decrease only linear speed by 10%
```

```
e/c : increase/decrease only angular speed by 10%
```

```
space key, k : force stop
```

```
anything else : stop smoothly
```

```
CTRL-C to quit
```

```
currently:      speed 0.2      turn 1
```

RC100 控制

- 使用 ROBOTIS RC100 的设置已经在 ROS 的 OpenCR 固件中，因此没有更多的必需软件包

PS3 操纵杆

- [Remote PC] 通过蓝牙或 USB 电缆将 PS3 操纵杆连接到 PC。
- [Remote PC] 安装 PS3 操纵杆软件包

```
sudo apt-get install ros-kinetic-joy ros-kinetic-joystick-drivers ros-kinetic-teleop-twist-joy
```

- [Remote PC] 启动 PS3 操纵杆的遥控软件包

```
roslaunch teleop_twist_joy teleop.launch
```

XBOX 360 操纵杆

- [Remote PC] 通过蓝牙将 PS3 操纵杆连接到 PC。
- [Remote PC] 安装 XBOX 360 操纵杆软件包

```
sudo apt-get install xboxdrv ros-kinetic-joy ros-kinetic-joystick-drivers ros-kinetic-teleop-twist-joy
```

- [Remote PC] 运行 XBOX 360 操纵杆的遥控软件包

```
xboxdrv --silent
```

```
roslaunch teleop_twist_joy teleop.launch
```

Wii 遥控器

- [Remote PC] 通过蓝牙将 Wii 遥控器连接到 PC
- [Remote PC] 安装 Wii 遥控器软件包

```
rosdep install wiimote
```

```
rosmake wiimote
```

- [Remote PC] 运行 Wii 遥控器的遥控器软件包

```
roslaunch wiimote wiimote_node.py
```

```
roslaunch learning_wiimote turtle_teleop_wiimote
```

Nunchuk 控制

- 待完善

Android 控制

- 下载 [ROS Teleop](#) 并运行应用程序

LEAP Motion

- [Remote PC] 通过蓝牙将 LEAP 运动连接到 PC。
- [Remote PC] 安装 LEAP Motion 软件包
 - <https://www.leapmotion.com/setup>
 - <https://developer.leapmotion.com/downloads/sdk-preview>
- 简单安装方法:

```
leapd
```

LeapCommandPanel

```
git clone git@github.com:warpl337/rosleapmotion.git
```

- [Remote PC] 运行 LEAP Motion

```
roslaunch leap_motion sender.py
```

十、入门教程-SLAM

说明：

- 介绍如何通过 turtlebot3 进行 SLAM 测试

SLAM

- The Simultaneous Localization and Mapping (SLAM)
- 同步定位与地图构建 (SLAM 或 Simultaneous localization and mapping) 是一种概念：希望机器人从未知环境的未知地点出发，在运动过程中通过重复观测到的地图特征（比如，墙角，柱子等）定位自身位置和姿态，再根据自身位置增量式的构建地图，从而达到同时定位和地图构建的目的。（维基百科）
- SLAM 技术是 TurtleBot3 的典型功能，是 Turtlebot 品牌的一类。这里的视频显示了 TurtleBot3 可以绘制多少精确的地图，即使它是一个小的便宜的机器人平台。
- 相关信息：

日期： 2016. 11. 29

机器人： TurtleBot3 基本模型

传感器： 激光距离传感器

包装： Gmapping/Cartographer

地点： ROBOTIS 实验室和总部，15 楼的走廊

时间： 55 分钟

距离： 共 351 米

通过远程操作创建地图

- [Remote PC]打开终端，然后运行 SLAM 启动文件

```
export TURTLEBOT3_MODEL=basic
```

```
roslaunch turtlebot3_slam turtlebot3_slam.launch
```

- [Remote PC] 通过 Rviz 可视化模型

```
roslaunch rviz rviz -d `rospack find turtlebot3_slam`/rviz/turtlebot3_slam.rviz
```

将地图保存到文件

- [Remote PC] 打开终端，然后运行地图保存节点

```
roslaunch map_server map_saver -f ~/map
```


十一、入门教程-导航

说明：

- 介绍如何利用 Turtlebot3 进行导航
- 导航技术的主要用途是使机器人进入期望的位置。

导航

- [Remote PC]启动导航文件

```
export TURTLEBOT3_MODEL = basic
```

```
roslaunch turtlebot3_navigation turtlebot3_navigation.launch
```

- 启动 Rviz

```
roslaunch rviz rviz -d `rospack find turtlebot3_navigation`/rviz/turtlebot3_nav.rviz
```

- [Remote PC] 在开始导航之前，TurtleBot3 应该知道它的位置和姿势。
- 要给出初始数据，请按照说明进行操作。
 - 点击 2D Pose Estimate 按钮
 - 通过单击并拖动地图上的方向来设置地图上的大致位置。
- 箭头的每个点意味着 TurtleBot3 的预期姿势。激光扫描仪将在近似位置绘制线条，如地图上的墙壁。
- 如果图形没有显示线条，请重复上述过程。
- [远程 PC]当 TurtleBot3 已经定位，它将自动计划路径。
- 要发送目标位置：
 - 点击 2D Nav Goal 按钮
 - 点击地图上你想要的 TurtleBot 驱动和拖动方向 TurtleBot 应该指向地方
- 如果目标位置的路径被阻止，这可能会失败。
- 要在机器人到达目标位置之前停止机器人，请发送 TurtleBot3 的当前位置。

参考文档：

- [http :
//wiki.ros.org/turtlebot_navigation/Tutorials/Autonomously%20navigate%20in%20a%20
known%20map](http://wiki.ros.org/turtlebot_navigation/Tutorials/Autonomously%20navigate%20in%20a%20known%20map)

十二、入门教程-仿真

说明：

- 介绍如何利用 turtlebot3 进行仿真

- 测试环境：Ubuntu 16.04 和 ROS Kinetic Kame.

结构图：



注意：

- TurtleBot3 Simulation 依赖 turtlebot3

步骤：

- 安装 TurtleBot3 Simulation

```
cd ~/catkin_ws/src/
git clone https://github.com/ROBOTIS-GIT/turtlebot3_simulations.git
cd ~/catkin_ws && catkin_make
```

- 启动仿真

```
export TURTLEBOT3_MODEL=burger
roslaunch turtlebot3_fake turtlebot3_fake.launch
```

- TurtleBot3 仿真节点不依赖实体机器人，也可以在 rviz 里通过 teleop 节点进行控制。

```
roslaunch turtlebot3_teleop turtlebot3_teleop_key.launch
```

Gazebo 仿真

- 第一次使用 gazebo 需要比较长的时间加载模型，也可以加载模型，参考
- 第一次使用 Turtlebot3 的 gazebo 仿真，需要把 Turtlebot3 的模型文件，复制到 gazebo 的模型目录里

```
mkdir -p ~/.gazebo/models/
cp -r ~/catkin_ws/src/turtlebot3_simulations/turtlebot3_gazebo/worlds/turtlebot3 ~
/.gazebo/models/
```

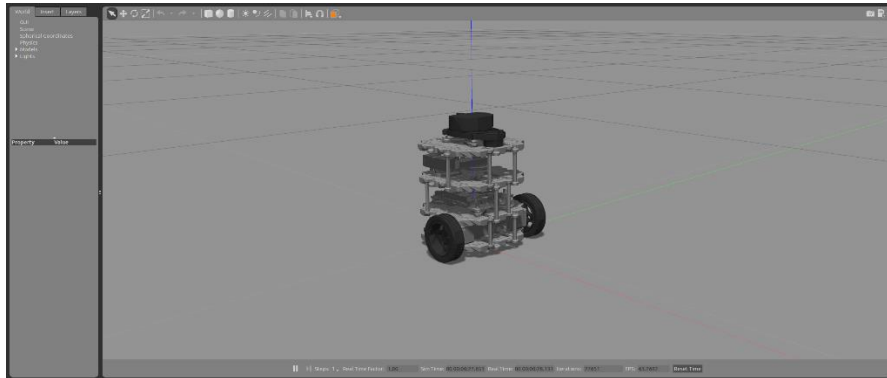
- 设置模型参数，指定使用那种机器人型号：burger 或者 waffle

```
export TURTLEBOT3_MODEL=burger
```

- 启动世界地图，默认的空白地图环境中加载 Turtlebot3 机器人

```
roslaunch turtlebot3_gazebo turtlebot3_empty_world.launch
```

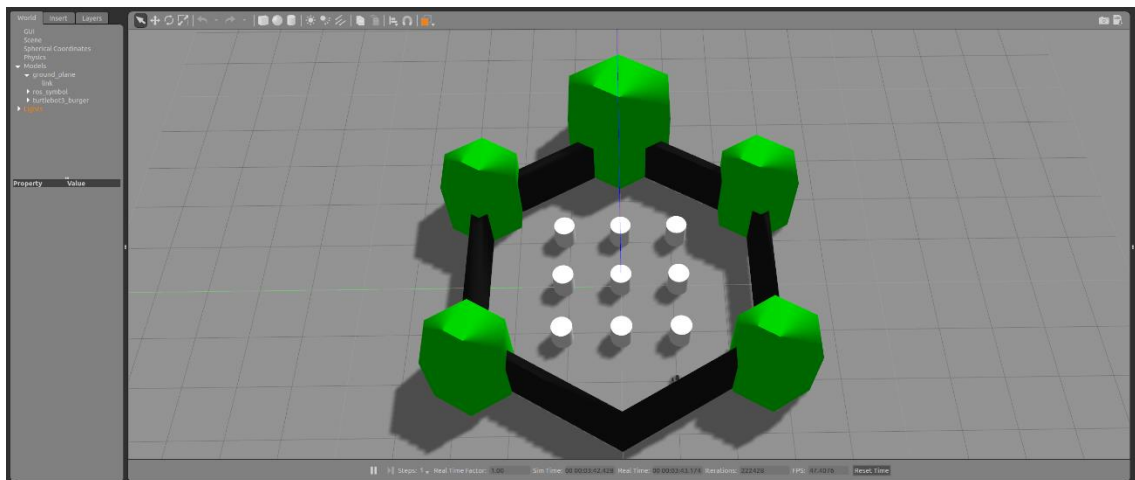
- 效果：

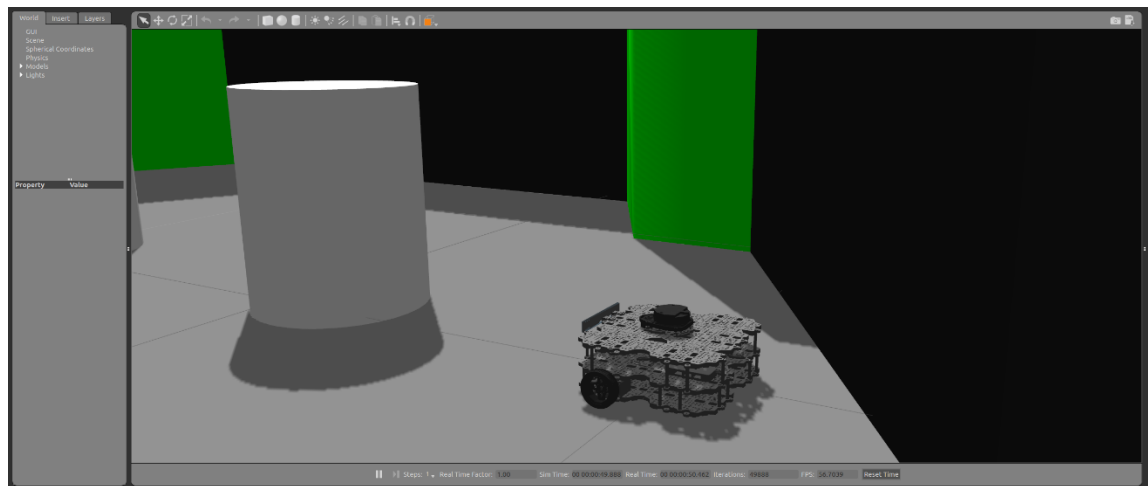


- 启动更复杂些的地图

```
roslaunch turtlebot3_gazebo turtlebot3_world.launch
```

- 效果：





- 用键盘控制 turtlebot3

```
roslaunch turtlebot3_teleop turtlebot3_teleop_key.launch
```

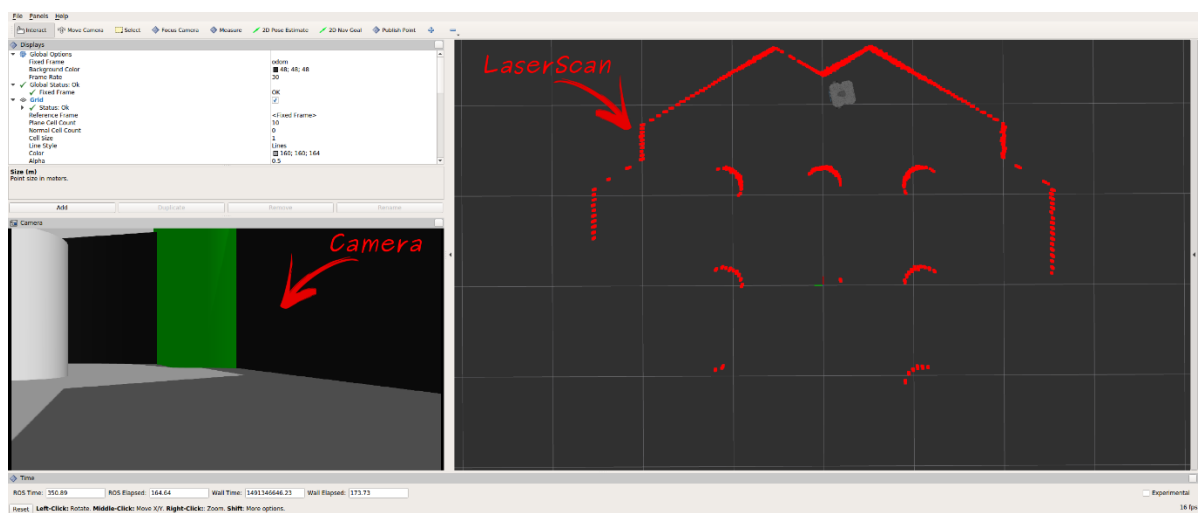
- 启动仿真

```
export TURTLEBOT3_MODEL=burger
roslaunch turtlebot3_gazebo turtlebot3_simulation.launch
```

- 启动 RVIZ 可视化

```
export TURTLEBOT3_MODEL=burger
roslaunch turtlebot3_gazebo turtlebot3_gazebo_rviz.launch
```

- 效果：



十三、入门教程-朋友

说明：

- 介绍 Turtlebot3 的多种不同组合

演示：

- 在视频中，看看如何使用一些小部件组装和重新组装 TurtleBot3。
- waffle 板是 TurtleBot3 组件中最大的部件，可以制造各种尺寸和形状的机器人，由几个大小的孔组成，并且需要少量的电压和螺母组装。
- 从这个想法，很多 TurtleBot3 friends 有不同的字符可以建立。
- 自定义结构如下所列：

TurtleBot3 Friends List (2017.02.03)

TurtleBot3 Friends: Conveyor

TurtleBot3 Friends: OpenManipulator Chain

TurtleBot3 Friends: Auto

TurtleBot3 Friends: Tank

TurtleBot3 Friends: Omni

TurtleBot3 Friends: Mecanum

TurtleBot3 Friends: Segway

TurtleBot3 Friends: Bike

TurtleBot3 Friends: Real TurtleBot

TurtleBot3 Friends: Road Train

TurtleBot3 Friends: Monster

TurtleBot3 Friends: Auto

- 类型：自动车

TurtleBot3 Friends: OpenManipulator Chain

- 类型：机械手 4 DOF + 1 夹具

TurtleBot3 Friends: Conveyor

- 类型：4 关节和 4 轮类型

TurtleBot3 Friends: Monster

- 类型：4 轮怪物卡车

TurtleBot3 Friends: Tank

- 类型：卡特彼勒

TurtleBot3 Friends: Omni

- 类型：全向轮

TurtleBot3 Friends: Mecanum

- 类型：Mecanum 轮

TurtleBot3 Friends: Segway

- 类型：Segway 机器人

TurtleBot3 Friends: Bike

- 类型：3-DOF 摩托车

TurtleBot3 Friends: Road Train

- 类型：公路列车

TurtleBot3 Friends: Real TurtleBot

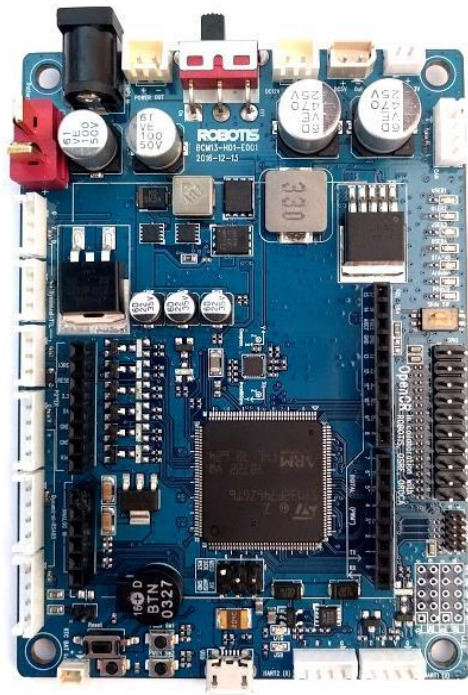
- 类型：8-DOF 有腿机器人（又名真正的 TurtleBot）

十四、入门教程-OpenCR

说明：

- 介绍 turtlebot3 的 OpenCR 控制板

图示：



概述

- OpenCR 是 TurtleBot3 的主控制器板。
- OpenCR 或 ROS 的开源控制模块是为 ROS 嵌入式系统开发的，提供了完整的开源硬件和软件。
- Board 的所有内容包括 Schematics，PCB Gerber，BOM 和 TurtleBot3 的固件源可以根据开源许可证免费分发给用户和 ROS 社区。
- STM32F7 系列是 OpenCR 板内的主芯片，具有非常强大的 ARM Cortex-M7 浮点单元。
- OpenCR 的开发环境从支持为年轻学生的 Arduino IDE 和 Scratch 到传统固件开发环境的专家。
- 该板提供一组数字和模拟输入/输出引脚，可从 pne 电路连接到另一个或内置 IMU 传感器。
- 该板的通信接口包括与 PC 通信的 USB 和用于其他嵌入式设备的 UART，SPI，I2C，CAN。
- 要使用 SBC，OpenCR 板可以提供最佳解决方案。
- 它支持一些电源输出：12V，5V，3.3V 的 SBC 和传感器。
- 它还在两个外部电源输入中具有热插拔功能：电池和 SMPS。

规格

项目	规格
微控制器	STM32F746NGH6 /具有 FPU 的 32 位 ARMCortex®-M7 (216MHz, 462DMIPS)
传感器	陀螺仪 3Axis, 加速度计 3Axis, 磁力计 3Axis (MPU9250)
程序员	ARM Cortex 10pin JTAG / SWD 连接器
	USB 设备固件升级 (DFU)
	串行
扩展引脚	32 引脚 (L 14, R 18) * Arduino Uno Revision 3 连接
	传感器 x 4 针
	扩展连接器 x 18 针
通信电路	USB (Micro-B USB 连接器/ USB 2.0 /主机/ Peripehral / OTG)
	TTL (JST 3pin / Dynamixel)
	RS485 (JST 4pin / Dynamixel)
	UART x 2
	CAN
LED 和按钮	LD2 (红/绿) : USB 通信
	用户 LED x 4 : LD3 (红色), LD4 (绿色), LD5 (蓝色)
	用户按钮 x 2
权力	外部输入源
	5 V (USB VBUS), 7-24 V (电池或 SMPS)
	默认电池 : LI-PO 11.1V 1,800mAh 19.98Wh
	默认 SMPS : 12V 5A
	外部输出源
	12V@1A, 5V@4A, 3.3V@800mA
	电源 LED : LD1 (红色, 3.3 V 电源打开)
	复位按钮 x 1 (用于板的电源复位)
	电源开关 x 1

项目	规格
尺寸	105 (W) ×75 (D) mm
质量	60g

- 从“shore power”（12V, 5A SMPS）切换到“移动电源”（电池）的热插拔：电源板支持不间断电源（UPS）类型的功能。

用户手册

- 运行 serial_node 包

```
roslaunch roserial_python serial_node.py __name:=turtlebot3_core _port:=/dev/ttyACM0 _
baud:=115200
```

- 测试

```
rostopic echo /imu
```

- 结果

```
header:
  seq: 179
  stamp:
    secs: 1486448047
    nsecs: 147523921
  frame_id: imu_link
orientation:
  x: 0.0165222994983
  y: -0.0212152898312
  z: 0.276503056288
  w: 0.960632443428
orientation_covariance: [0.0024999999441206455, 0.0, 0.0, 0.0, 0.002499999944120645
5, 0.0, 0.0, 0.0, 0.0024999999441206455]
angular_velocity:
  x: 2.0
  y: 1.0
  z: -1.0
angular_velocity_covariance: [0.019999999552965164, 0.0, 0.0, 0.0, 0.019999999552965
164, 0.0, 0.0, 0.0, 0.019999999552965164]
linear_acceleration:
  x: 528.0
  y: 295.0
  z: 16648.0
```

```
linear_acceleration_covariance: [0.03999999910593033, 0.0, 0.0, 0.0, 0.03999999910593033, 0.0, 0.0, 0.0, 0.0, 0.03999999910593033]
---
```

github 仓库:

- <https://github.com/ROBOTIS-GIT/OpenCR>

详情:

- <https://github.com/ROBOTIS-GIT/OpenCR/wiki>

十五、入门教程-LDS

说明:

- 介绍 Turtlebot3 的激光雷达 LDS

图示:



概述

- HLS-LFCD LDS 用于 TurtleBot3 的两种型号。
- LDS (激光距离传感器) 是将由障碍物检测收集的数据发送到用于 SLAM 技术的主机的传感器

基本性能规格

项目	规格
工作电源电压	5V DC \pm 5%
光源	半导体激光二极管 ($\lambda = 785\text{nm}$)
激光安全	IEC60825-1 Class 1
目前的消费	400mA 以下 (冲击电流 1A)
检测距离	120mm ~ 3,500mm
接口	3.3V USART (230,400 bps) 每 6 度 42bytes, 全双工选项
环境光电阻	10,000 lux 或更小
采样率	1.8kHz
尺寸	69.5 (W) \times 95.5 (D) \times 39.5 (H) mm
质量	低于 125g

测量性能规格

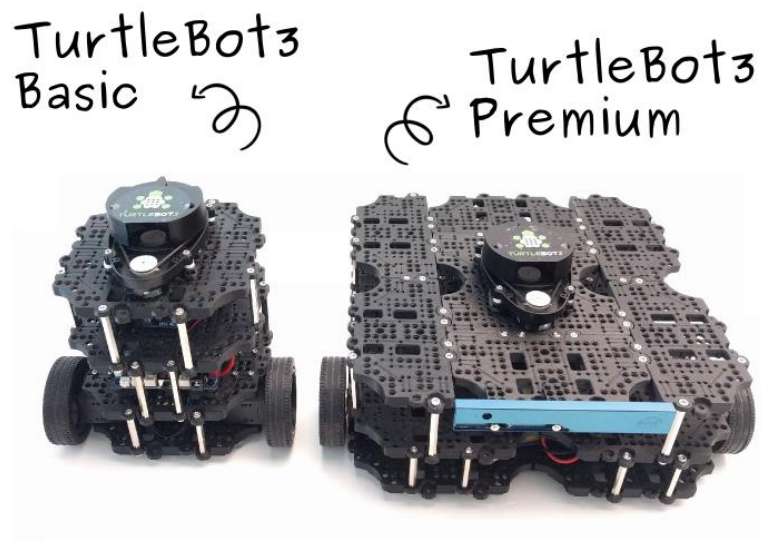
项目	规格
距离范围	120~3500mm
距离精度（120mm~499mm）	±15mm
距离精度（500mm~3,500mm）	±5.0%
距离精度（120mm~499mm）	±10mm
距离精度（500mm~3,500mm）	±3.5%
扫描速率	300±10rpm
角范围	360°
角分辨率	1°

规范文档

- 包括内容如基本性能，测量性能，机构布局，光路，数据信息，引脚描述，命令。
- 这里是详细规范文档 [pdf](#)

在 TurtleBot3 使用 LDS

- HLS-LFCD LDS 用于 TurtleBot3 Basic 和 TurtleBot3 Premium。
- 图示：



- 演示：
- ROS Hector SLAM 演示，仅使用由 HLDS（Hitachi-LG Data Storage）制造的海思-LFCD LDS。
- ROS Gmapping 和 Cartographer SLAM 演示使用 TurtleBot3 和海思-LFCD LDS：

用户手册

- LSD 的 ROS 包 [ROS package for LSD](#)
- hls_lfcd_lds_driver 包是 “HLS(Hitachi-LG Sensor) LFCD LDS(Laser Distance Sensor)”的驱动.
- 安装：

```
sudo apt-get install ros-kinetic-hls-lfcd-lds-driver
```

- 设置 HLS-LFCD LDS 的权限

```
sudo chmod a+rw /dev/ttyUSB0
```

- 运行 hlds_laser_publisher 节点

```
roslaunch hls_lfcd_lds_driver hlds_laser.launch
```

- 运行 hlds_laser_publisher 节点并启动 rviz

```
roslaunch hls_lfcd_lds_driver view_hlds_laser.launch
```