

Autonomous Mobile Robots Course

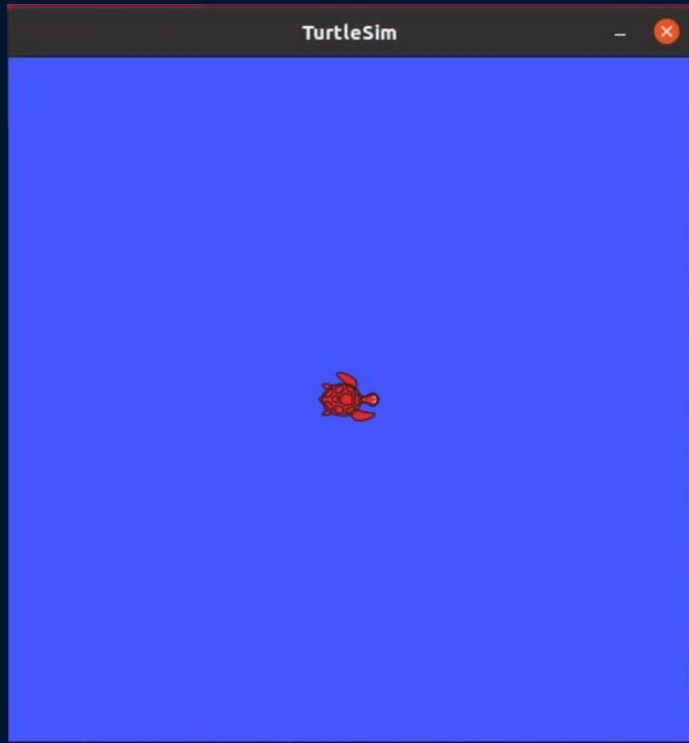
Turtlesim, turtlebot 3 and
perception sensors

ROS TOPIC & message

In order to have all of this working, we need to have a roscore running. The roscore is the main process that manages all of the ROS system. You always need to have a roscore running in order to work with ROS



Turtlesim Robot



- Keyboard Control
- Draw square
- `rqt_graph`
- `roscnode (list,info)`
- `rostopic (list,info,echo)`
- `rosmmsg info`



Turtlebot 3 Robot with ROS

- `cd ~/catkin_ws/src/`
- `git clone -b noetic-devel https://github.com/ROBOTIS-GIT/turtlebot3_simulations.git`
- `git clone -b noetic-devel https://github.com/ROBOTIS-GIT/turtlebot3.git`
- `git clone -b noetic-devel https://github.com/ROBOTIS-GIT/turtlebot3_msgs.git`
- `cd ~/catkin_ws`
- `catkin_make`
- `source devel/setup.bash`
- `export TURTLEBOT3_MODEL=burger`
- `roslaunch turtlebot3_gazebo turtlebot3_empty_world.launch`

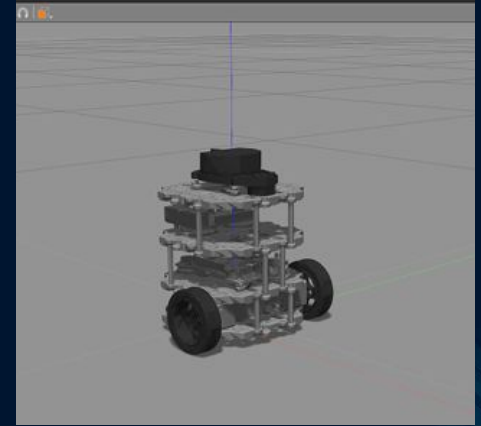


ROS Node VS Launch Files

- **ROS nodes:** is A ROS node is a standalone ROS program that written in C++ or Python
- **roslaunch files:** are defined in XML can run many nodes in the same time without a need to run ROS master

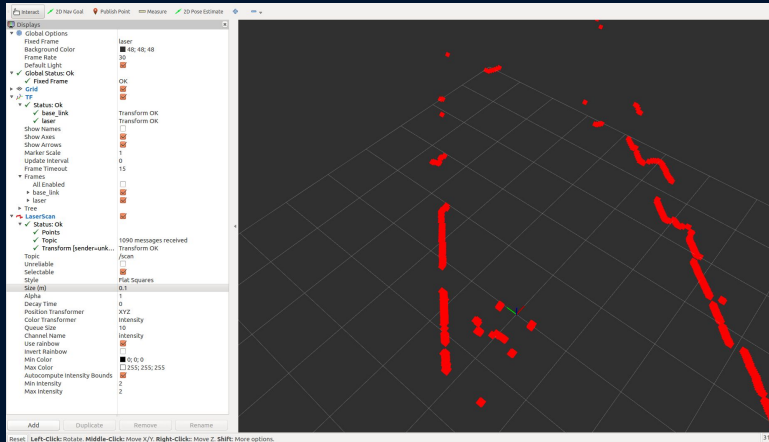
Turtlebot 3 Robot with ROS

- `cd ~/catkin_ws/src/`
- `git clone -b noetic-devel https://github.com/ROBOTIS-GIT/turtlebot3_simulations.git`
- `git clone -b noetic-devel https://github.com/ROBOTIS-GIT/turtlebot3.git`
- `git clone -b noetic-devel https://github.com/ROBOTIS-GIT/turtlebot3_msgs.git`
- `cd ~/catkin_ws`
- `catkin_make`
- `source devel/setup.bash`
- `export TURTLEBOT3_MODEL=burger`
- `roslaunch turtlebot3_gazebo turtlebot3_empty_world.launch`



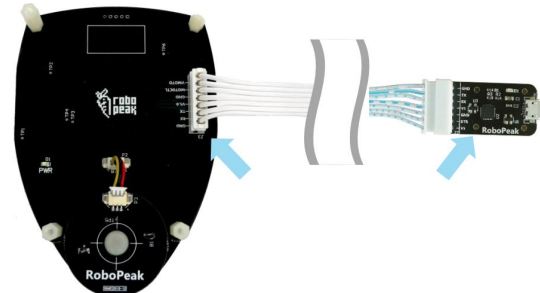
2D LIDAR sensor

2D 360-degree LiDAR sensor is a type of lidar system that uses laser beams to collect data about the environment in 2D and 360 degrees. LiDAR, or Light Detection and Ranging, is a remote sensing technology that uses lasers to measure the distance between a sensor and objects in its environment.



RPLIDAR A1 Connection

[rplidarkit_usermanual](#)



RPLIDAR A1 with ROS

<https://wiki.ros.org/rplidar>

- `sudo apt-get install ros-noetic-rplidar-ros`
- `ls -l /dev |grep ttyUSB`
- `sudo chmod 666 /dev/ttyUSB0`
- `roslaunch rplidar_ros rplidar.launch`
- `roslaunch rplidar_ros view_rplidar.launch`

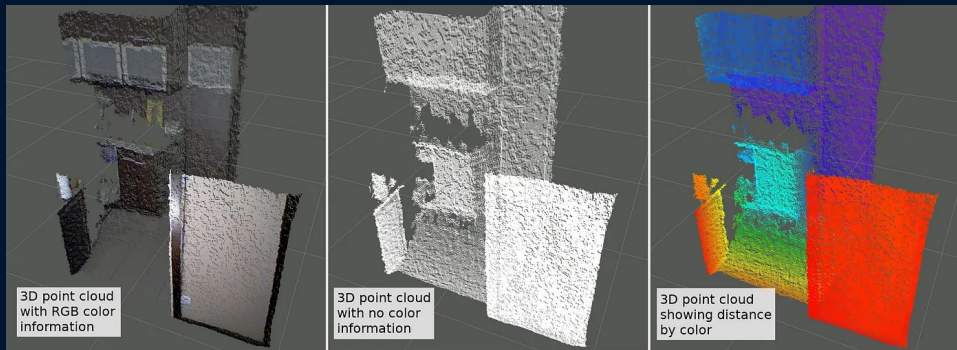


Depth camera kinect

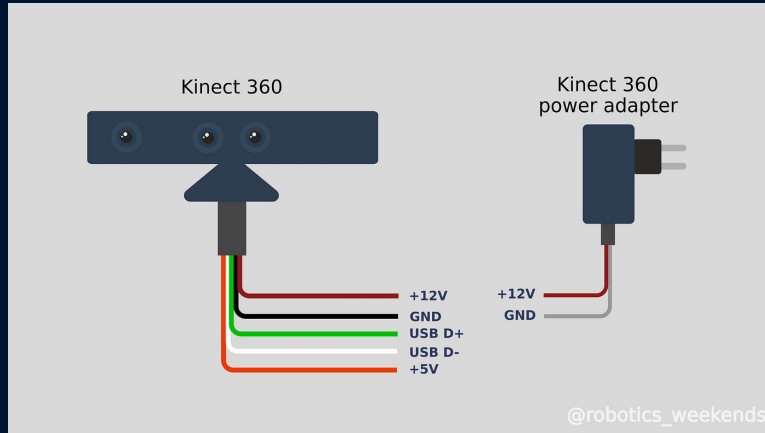


The Kinect Depth Camera is a hardware device that captures depth and color information simultaneously using a set of infrared sensors and a color camera.

It can be used with ROS to create depth-sensitive applications for robotics, robot vision, and augmented reality.



Kinect V1 Connection



Kinect V1 with ROS Noetic



<https://aibegins.net/2020/11/22/give-your-next-robot-3d-vision-kinect-v1-with-ros-noetic/>

- **Install the dependencies & Get the libfreenect repository from GitHub**
 - `sudo apt-get update`
 - `sudo apt-get install git-core cmake freeglut3-dev pkg-config build-essential libxmu-dev libxi-dev libusb-1.0-0-dev`
 - `git clone https://github.com/OpenKinect/libfreenect.git`
 - `cd libfreenect`
 - `mkdir build`
- **Make and install**
 - `cd build`
 - `cmake -L ..`
 - `make`
 - `sudo make install`
 - `sudo ldconfig /usr/local/lib64/`
- **To use kinect without sudoing every time**
 - `sudo adduser $USER video`
 - `sudo adduser $USER plugdev`

Kinect V1 with ROS Noetic



- **add some device rules**
 - `sudo nano /etc/udev/rules.d/51-kinect.rules`
- **Paste the following and ctrl+q to save**
 - `# ATTR{product}=="Xbox NUI Motor"`
 - `SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02b0", MODE="0666"`
 - `# ATTR{product}=="Xbox NUI Audio"`
 - `SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02ad", MODE="0666"`
 - `# ATTR{product}=="Xbox NUI Camera"`
 - `SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02ae", MODE="0666"`
 - `# ATTR{product}=="Xbox NUI Motor"`
 - `SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02c2", MODE="0666"`
 - `# ATTR{product}=="Xbox NUI Motor"`
 - `SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02be", MODE="0666"`
 - `# ATTR{product}=="Xbox NUI Motor"`
 - `SUBSYSTEM=="usb", ATTR{idVendor}=="045e", ATTR{idProduct}=="02bf", MODE="0666"`

Kinect V1 with ROS Noetic



- **download the required ROS package**
 - `cd ~/catkin_ws/src`
 - git clone https://github.com/ros-drivers/freenect_stack.git
 - `cd ..`
 - `catkin_make`
 - `source devel/setup.bash`
- **launch the freenect example for depth registration which allows you to get the point cloud with RGB data superimposed over it.**
 - `roslaunch freenect_launch freenect.launch depth_registration:=true`
- **visualize the topics from Kinect on Rviz**
 - `rviz`