Lab Guide

ROS 2 For QCar 2

Description

This document focuses on the series of launch files and nodes currently designed for QCar 2. Please review the User Manual – Connectivity for information on how to copy files onto the QCar 2.

Getting Started:

All the examples provided for QCar 2 are compatible with **ROS 2 Humble**. To get started with ROS 2 please check the following link: https://docs.ros.org/en/humble/index.html .
This guide assumes you have a basic understanding of the ROS 2 ecosystem and know how folder structures, packages and nodes work in the context of ROS.

Files Provided:

QCar 2 ROS 2 nodes can be found as part of the available research examples provided by Quanser. Using file explorer, please navigate to the following resource directory: C:/User/<user>/Documents/Quanser/5_research/qcar2. Packages available for QCar 2:

- qcar2_interface
- qcar2_nodes

qcar2_interfaces is a custom package for qcar2 specific elements. Custom message types are:

- BooleanLeds.msg
- MotorCommands.msg

qcar2_nodes is a Quanser design QCar2 package which configures the following information:

Sensor Centric Nodes

- **qcar2_hardware** (this node includes motor commands, LED values and publishes IMU data, battery level, joint states)
- **lidar** (publishes LiDAR info for the RP Lidar A2M12)
- **csi** (publishes image for 1 csi camera, camera ID can be modified to request information from a different CSI camera)
- **rgbd** (publishes RGB and depth information for D435 Intel RealSense camera)

User Centric Nodes

- **Command (**This node allows a user to send manual commands to the QCar using a Logitech F710 joystick)
-

Auxiliary Nodes

- **Image_viewer** (Allows a different way for a user to view compressed images using OpenCV rather than Rviz2)
- **fixed_lidar_frame** (Transforms and rotates the LiDAR frame to align correctly with the front of the QCar).
- **nav2_qcar_command_convert** (Converts velocity messages using a Twist message type to the correct motor command topic expected by the qcar2_hardware node)

Launch folder contains different applications to help you get started with the QCar 2.

- **qcar2_launch.py** (Launch file designed to publish all the sensor centric nodes)
- **qcar2_manual_drive.py** (Launch file includes sensor centric nodes and command node to manually drive the QCar 2).
- **qcar2_cartographer_launch.py** (Launch file for mapping and QCar without a command node)
- **qcar2_manual_cartographer.py** (Launch file manually driving a QCar 2 and generating a map of the environment)
- **qcar2_slam_and_nav_bringup_launch.py** (Launch file which uses Nav2 to generate commands for a QCar 2 to navigate in a desired space based on a goal pose given via Rviz2)

Running Examples

Prior to running any ROS 2 example, please make sure you have sourced **humble** as the desired ROS 2 distribution in your current terminal session.

```
source /opt/ros/humble/setup.bash
```

Navigate to the **ros2/src** folder on the QCar 2 root directory. Copy **qcar2_interfaces** and **qcar2_nodes** inside this directory.

Note: If this is the first time compiling the ros2 workspace navigate to the **/ros2** folder and use the command **colcon build** to compile the ros2 workspace. Once the workspace has compiled successfully it needs to be sourced as a ROS 2 package using the following command:

```
source install/setup.bash
```

Running nodes:
To run any ROS 2 node, use the following command:

```
ros2 run <package name> <node name(s)>
```

As an example, to run the QCar2 LiDAR node, use the following command:

```
ros2 run qcar2_nodes lidar
```

Note: Nodes run 1 per terminal session. If you want to run multiple nodes either use multiple terminal sessions or combine them into launch files.

Running launch files:
Launch files combine a series of nodes into a unique application. To run any launch node use the following command:

```
ros2 launch <package name> <name of launch file.py>
```

As an example, if you want to run the QCar 2 manual drive example use the following command:

```
ros2 launch qcar2_nodes qcar2_manual_drive.py
```