

Setting up the Clearpath Husky A200 for Outdoor Autonomous Operation

The Clearpath Husky A200 can use slightly modified versions of ROS to successfully localize and operate in an outdoor environment. It does so by leveraging the latest changes in GPS, IMU systems and the [ROS-Navigation stack](#).

Getting the latest code

As of the writing of this document a lot of the latest modifications to standard ROS software has not be propagated to be part of the “standard” ROS debian packages. As such, some of these packages will have to be “overlaid” to overwrite the standard ROS packages available on a default install. A guide to setup a overlay workspaces can be found [online](#) (Note that overlaying workspaces remains the same between ROS-Groovy and ROS-Hydro). Below is a list of packages (and their specific branches) that must be overlaid on the robot PC.

Package Name	Package Location	Branch Name	Purpose
navigation	https://github.com/clearpathrobotics/navigation	hydro-devel	Collection of packages for autonomous operation*.
husky_navigation	https://github.com/roverhusky/roverhusky_navigation	lidar_navigation	Collection of launch and configuration files specific to the husky.
husky_bringup	https://github.com/roverhusky/roverhusky_bringup	compass_calibration	Husky default launch files and scripts **
imu_compass	https://github.com/clearpathrobotics/imu_compass	param_change	Node to combine IMU data with compass data

*The changes to enable GPS operation within the navigation stack are yet to be pushed upstream to [ROS-Planning](#).

To checkout the latest version of the code into your overlay workspace simply enter the following commands in the src directory of your workspace

```
git clone <package location>
git checkout <branch name>
```

Then navigated out to the root of the workspace and build the packages
`catkin_make`

To ensure that the packages have been properly overlaid type in:
`rospack find <package_name>`

The output should point to your own workspace and not `/opt/ros/..`

Installation and Calibration

With the GPS connected (make sure the port `/dev/clearpath/gps` is enumerated) type in,

```
roslaunch husky_bringup install
```

This will install the correct launch files (GPS and IMU) to come up everytime the computer is booted up.

IMU Calibration

In order for the Husky to operate in the GPS frame of reference it needs to have a measure of its heading in the world co-ordinate frame i.e. with respect to true North. Commercial grade GPS's have a poor measurement of this value. The Husky is outfitted with an IMU and a Magnetometer that can provide an accurate and precise measurement of the vehicle's heading. However, prior to operation the magnetometer has to be calibration to remove hard-iron effects. For this, carry out the following steps:

- Drive the husky outdoors, away from most major ferrous sources and obstacles
- With all of husky's systems running, type in: `roslaunch husky_bringup calibrate_compass`
- The husky will now slowly turn in place while capturing the relevant IMU data. Once complete, it will run through the calibration process and ask to write the hard-iron offsets to a specific location (`/etc/ros/hydro/imu_compass.yaml`) where it can be retrieved every time it boots. Type 'Y' to approve this writing process.
- As instructed restart the husky service,
`sudo service husky-core restart`

Now, the husky will boot up with these modified values every time.

Test IMU Calibration

Once the ROS service has restarted, test if the IMU calibration is successful by running the following command

```
rostopic echo /imu/compass_heading
```

This will output the latest filtered compass measurements to the husky in radians with respect to East (as per the ENU co-ordinate frame).

Modifying robot_pose_ekf parameters for outdoor operation

The modified navigation stack has a version of `robot_pose_ekf` that filters in GPS and compass measurements in estimating the pose of the vehicle outdoors. In default settings, `robot_pose_ekf` is configured to run in "indoor mode" and ignore GPS measurements. To enable GPS mode the configuration in the file:

```
/etc/ros/hydro/husky_ekf.yaml
```

In this file enable GPS by setting:

```
gps_used: true
```

and enable compass measurements by setting:

```
imu_absolute: true
```

As always, restart the husky-core service to restart the system in this new configuration.

Testing robot_pose_ekf

On the **base station** computer that is connected to the Husky run

```
roslaunch husky_viz view_navigation.launch
```

You should now see the husky on your monitor. Drive the husky around outdoors and make sure the motion is consistent!