Getting started with Husqvarna Research Platform

Fredrik Båberg
Computer Vision and Active Perception
KTH

January 2016

Abstract

This document explains how to setup the Husqvarna Research Platform on a computer. It details the step of powering up the robot, starting simulation, and launching the teleoperation node.

1 Installation

This section describes how to install software needed.

1.1 Install software on PC

This document assumes all code will be located in a catkin workspace src folder, in this case `~/ws/src/`. Extract the archive through `tar -xvf hrp*`, where

`hrp*` indicates the tar.gz archive.

- Install some additional packages that will be needed:

  ```
  sudo apt-get install ros-indigo-nmea-msgs libcgal-dev screen
  ```

- Also need to install dependencies from catkin workspace. It is possible `catkin_make` has to be executed first:

  ```
  cd ~/ws
  catkin_make
  rosdep install --from-paths src --ignore-src --rosdistro indigo
  ```

Now build the workspace through `catkin_make`. Don't forget to source the workspace `source ~/ws/devel/setup.bash`, it could be useful to add this to `~/.bashrc`

1.2 Preparation for running on hardware

An additional installation step is needed for running on hardware, related to communication with the robot. This step is a preparation that should only be needed once on the device.

- Add the current user to the `dialout` group through the command

  ```
  sudo adduser <username> dialout
  ```

You will need to log out and log in before you continue.

2 Running

With everything installed, you are now almost ready for launch.

2.1 Start robot

To prepare the robot for usage in a lab environment (without using a *loop*, other word for the wire), follow this procedure:

- Change power switch to **ON**
- Enter PIN code (**1111**)
- Press **MENU**
- Hold 7 and 9 until new menu (**TOOLS**) appears
- Select wrench, press **OK**
- Select **Special settings** (last entry in list), press **OK**

- Tick **Override loop detection** by pressing **OK**
- Press **OK** to confirm
- Press **START**, or **BACK** until start menu appears and then **START**
- Close hatch (or use plastic switch), display should now say **MOWING**, the platform will move slightly, and is then ready for use

2.2 Start Gazebo

For using Gazebo, one additional step is required. It could be useful to add this line to your `~/.bashrc` file, to avoid typing it every time

```
export GAZEBO_MODEL_PATH=~/ws/src/hrp/am_gazebo/models:$GAZEBO_MODEL_PATH
```

Gazebo, together with robot and an environment, can then be launched through

```
roslaunch am_gazebo am_gazebo_hrp_tracking.launch gui:=true
```

Note that there are several parameters that could be set, look in the launch file. When launching both Gazebo and running on hardware, it could be useful to add `app:=false steering:=false` so that the robot doesn't automatically move on launch.

2.3 Launch hardware drivers

Before launching, make sure you have the correct port. Start the platform according to Section 2.1, and connect the USB cable.

- Check which port is used through `dmesg`, it could for instance be `/dev/ttyACM0` (0 is the number zero)

  ```
  dmesg | grep ACM
  ```

- Now update the device through the following command, assuming that the device you got was `ttyACM0`:

  ```
  screen /dev/ttyACM0 115200
  ```

- The terminal will now be blank. Press `ctrl+a` followed by `k` and `y` to kill screen.

The steps before should only be needed once. To run the drivers, make sure `roscore` is running, and execute `rosrun am_driver am_driver_node` to run the driver. Output should end with

```
Serial port ONLINE!
Serial port connected!
```

if everything is correct.

2.4 Teleoperation

To launch the teleoperation node, execute the following

```
rosrun am_control key_teleop.py
```

You can switch between teleoperation and random behavior through **1** and **2**. For other commands, see the python script.

2.5 UWB sensors

To launch the UWB sensors, connect the master **s100** to the computer using an FTDI cable (3.3V), and check which USB port is used (could be `/dev/ttyUSB0`). Launch through

```
rosrun am_uwbrange am_uwbrange_node _serialPort:=/dev/ttyUSB0
```

Information is then published on ROS topic `/uwb`. It will include range between all sensors found.

Figure 1: Connection of FTDI cable, with ground visible