Crisp - Manual for running the f1tenth car

Björn Andersson

March 2022

Handling the battery

The battery on the car is a lithium battery, so some care must be taken when handling it. Connect the small white end to the power board of the car. Connect the yellow connector to the VESC. Make sure to fit it correctly + to +, - to -.

Before starting it can be a good idea to check if the battery is charged. This can be done with the multimeter. Set the multimeter to read DC current, and check the the ends of the small white connector. Black is ground, red is power in.

The battery is fully charged around 12 V. And should ideally never drop lower than 9V.

If the battery needs to be charged use the battery charger. Remove the yellow extension on the battery and plug the battery connector into the charger. It should automatically select LIPO and BALANCE. Let it stay this way and press and hold the START button until the charging starts. It will make a beeping sound when it is finished.

After charging put the extension wire back onto the battery connector and you are good to go.

Logging into the car

The car has a static IP address: 158.39.162.180, connected on the IOT2 network.

As long as you connect through a wire straight to the wall in the CPS lab or if you connect to IOT2, you should be able to access the car.

When logging in use:
ssh crisp-tron@158.39.162.180
password: qwertyui

sometimes the autologin on startup on the car has failed, leading to that you can't ssh into the car. Then plug in a screen with HDMI, connect a keyboard and type in the password. Now you're good to go.

Explanation of some commands

To make things easier, mostly for myself, I have set a up a few short commands on the car.

rmake : This builds the f1tenth system if you have made any changes. Must be run from the Crisp work-space folder. (crisp_ws).

refresh : This sources the bin file contain the ROS enviroment. Must be run on every console that are going to use ROS.

car : Fires up the f1tenth system.

cam : Enables the camera.

There are a few more commands that you need to know about.

roscore : starts the ROS master. Only need one of this.

rosrun : executes a script. rosrun package-name script-name.

tmux : A terminal application that lets you open many terminal windows in the same ssh session. This must be run. And this program has a few commands that you need to know too.

1. **(ctrl + B) + (ctrl+ C)** : Creates a new terminal window.
2. **(ctrl + B) + (ctrl + P)** : Navigate to the previous terminal window.
3. **(ctrl + B) + (ctrl + N)** : Navigate to the next terminal.

A few more useful commands to make use of would be:

rostopic : lets you choose to list available topics, echo a spesific topic etc. Or investigate what kind of data a specific topic contains.
For more detailed information on usage see:
http://wiki.ros.org/ROS/Tutorials/UnderstandingTopics

rosmsg : Let you look more closley on a specific ROS message. See:
http://wiki.ros.org/ROS/Tutorials/CreatingMsgAndSrv

rosbag : Is the command I have used to record data. I have created a folder crisp_bag to store the bag files. If you navigate to this folder before running the command, then the files will be stored there automatically. You can ofcourse use a different folder if you like. They are stored in a somewhat unreadabnle way. Don't know how familiar you are with using Linux, but you could use "**mv old-bag-file-name new-bag-file-name**" to change the name to something easier. For more info on ROS bag see:
http://wiki.ros.org/rosbag/Tutorials

In general I recommend having a look here if you need more need more information:
http://wiki.ros.org/ROS/Tutorials

You can ofcoure always ask me also, and i shall answer as best as I can.

The script which run the car is located in:
~\crisp_ws\src\f1tenth_system\camera\scripts\drive.py

If you wish you can make a copy of this and modify it as it fits to you. The speed is given as (linear speed, angular speed). As of now, don't use linear speeds over 4.0 if you don't have a longer space than the cps lab to drive on.

The steering goes approximately from -0.35 to +0.35.

Proper braking is not implemented yet.

Driving with a controller

There is a logitec controller which can be used to drive the car. All should be set up. You just need to plug in the small usb stcik that follwed with the remote. Should be in the blue box behind the screens. If yo use this, then only follow the recipe below until point 9.

Also you could have a look at this page, it will basically tell you how to set up the car to run with the controller. I have somewhat automated this task for you. But you can read it for reference if you like to. Navigate to "Driving The F1TENTH Car" and "Manual Control".

https://f1tenth.org/build.html

Recipie for starting the F1tenth system:

1. SSH into the car.
2. tmux
3. Navigate to the crisp_ws folder.
4. rmake
5. refresh
6. roscore
7. Create a new terminal window.
8. refresh
9. car
10. Create a new terminal window.
11. refresh
12. rosrun camera drive.py