Aerial Robots

TP1: Waypoint navigation

Preliminary steps

1. Boot into Linux and log in
2. Remove existing repository, if any

   ```
   cd ~ && rm -rf dronecourse-student-*
   ```

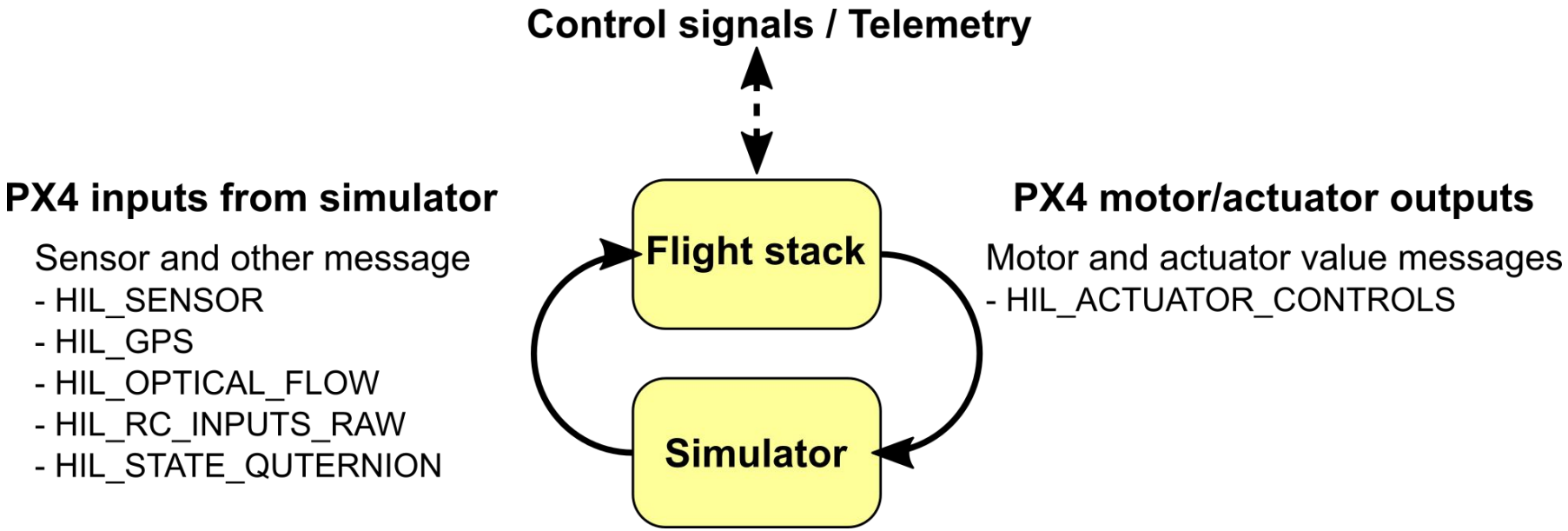
3. Clone your personal repository (one command!)

   ```
   git clone
   https://github.com/dronecourse-epfl/dronecourse-student-<YOUR_SCIPER>.git
   ```

4. Build the code and run the simulation

   ```
   make dronecourse_gazebo
   ```

Communication between PX4 and Gazebo



Code structure

```
dronecourse-student-XXXXXX/

    Documentation/          # Docs (make dronecourse_doxygen)

    msg/                    # Contains uORB message definitions

    src/dronecourse/        # Your code goes here!

    Tools/sitl_gazebo/      # Gazebo models and world files

    tp_eval/                # Code for evaluation
```

Manual

- The manual will guide you through the exercises step by step
- You will get more freedom in the implementation as the semester progresses
- Use the forum in case you have questions

Task 1: Waypoint navigation

How to evaluate your code

- The evaluation script is in the `tp_eval/` folder, and it is called `evaluate.py`
  - It will go through the .log file created during your last simulation and output your score.
- In order to run it, you should use the form

  `./Tools/dronecourse_docker_run.sh '<YOUR COMMAND>'`
  - This will run the command in the docker container where the dependencies are installed
- Run help to display options

  `./Tools/dronecourse_docker_run.sh 'python tp_eval/evaluate.py -h'`
- Evaluate your performance, creating a txt log file and a spreadsheet for the waypoint navigation task (task 1)

  `./Tools/dronecourse_docker_run.sh 'python tp_eval/evaluate.py -l -t1'`

How to evaluate your code (part II)

- Additionally, a Jupyter Notebook has been created for you to debug your code:
- Run the notebook

  `make dronecourse_notebook`
- Open your browser and go to page

  `localhost:8888`
- Use `SHIFT+ENTER` to run a code block

Save your work to your remote repository (memo)

1. **`git add <file name(s)>`**

   (ex. `git add src/examples/px4_simple_app/px4_simple_app.c`)

   - This will add the file to your next commit
2. **`git commit -m “<commit message>”`**

   (ex. `git commit -m ‘Implementation of HelloSky tutorial’`)

   - This will save your changes locally.
3. **`git push origin master`**

   - This will publish your changes to the online repository. Now you can sync from a different machine and access older versions online.

Note: you can repeat 1. and 2. several times and push multiple commits