Lab 1: Flying the Drone

The purpose of this lab is to setup your computer and the Parrot Rolling Spider drones for use in the class. First, you will install a Ubuntu virtual machine on your computer and then install new firmware on the drone. Then you will compile and upload a controller to the drone, perform a test flight of the drone, and download and analyze the data from the test flight.

In this document we describe how to install a Ubuntu 14.04 virtual machine with Matlab 2015 version and flash the Parrot Rolling Spider Drone with new firmware. More information is available in the "Getting Started" pdf document from MIT which can be found on the "Drone Resources" page in bCourses or can be downloaded from

`https://github.com/Parrot-Developers/RollingSpiderEdu/blob/master/MIT_MatlabToolbox/media/GettingStarted.pdf`

This document was updated on 1/19/2017.

Setting up the Virtual Machine

1. Download the Virtual Machine (VM) from:

   `https://drive.google.com/drive/folders/0B0n3QNOb7StZa1pwREJxb0tnYVE?usp=sharing`

   It consists of two files one of which is about 30Gb!. They can both be downloaded on an external memory stick.

2. Download and install VirtualBox from:

   `https://www.virtualbox.org/wiki/Downloads`

3. Open VirtualBox and click on `New`. Choose a name for the VM and set the `Type` to Linux and the `Version` to Ubuntu 64-bit. Then click `Next`. Set the memory size to at least 1Gb and click `Next`. For the Hard Disk select "Use an existing virtual hard disk file" and select the file `Ubuntu64.vdi.vmdk` that was downloaded in Step 1. Then select `Create` to finish the virtual machine.

4. Launch the virtual machine by clicking `Start` in VirtualBox. You should have a Ubuntu system now that is all setup and ready to go. The VM Password and Username is: **student**

5. If asked if you would like to upgrade Ubuntu, select `No`.

6. The VM includes Matlab but it needs to be activated. Activate Matlab by opening a terminal and running:

   ```
   $ cd /usr/local/MATLAB/R2015b/bin
   ```

   to move to the correct directory. Then run:

   ```
   $ sudo bash activate_matlab.sh
   ```

   Then, you can activate Matlab by logging into your Mathworks account and following the instructions.

In the *MIT Getting Started* document, the text `[RosMat]` refers to the directory

`/home/student/RollingSpiderEdu/MIT_MatlabToolbox`

in the VM filesystem.

Flash drone with firmware

Before using the drone it must be flashed with custom firmware. This will overwrite the default operating system and software on the drone. Therefore, you will not be able to operate the drone through the Parrot Application, but will be able to connect to it and upload controllers to it via the VM. Nevertheless, it is easy to reinstall the factory operating system from Parrot using the USB cable and your computer, and this is explained in Section **Resetting the Drone**.

1. Physically connect the drone to your computer using the USB cable. After a few seconds, if you do not see the Parrot Rolling Spider listed in the left panel of the Ubuntu file manager, make sure your drone is on and click `Devices/USB` in the menu bar (This can also be accessed by right clicking on the USB icon in the lower right corner of VirtualBox). You should see "Parrot Rolling Spider", click on it. Now the Rolling Spider should appear in the file manager list of devices.
2. Click on the "Parrot Rolling Spider" to see the files/directories on it. Open `fvt6.txt` and note the name and MAC (media access control) address. It should be a long string of capital letters, numerals, and colons.
   - If no `fvt6.txt` can be found, skip step 3 for now, and after you finish step 9, run
     ```
     $ sudo hcitool scan
     ```
     which will list the name and MAC address of available Bluetooth devices. The name of your drone will be of the form "RS_#####" where ##### will be a number unique to each drone. Now that you have the MAC address do step 3 and continue with step 10 afterwards.
3. Save MAC address to `DroneMACaddress.txt` by entering in a terminal:
   ```
   $ DroneSetMACaddress.sh MACADDRESS
   ```
   where "MACADDRESS" is the MAC address of your drone.
4. Upload the main firmware to the drone by running
   ```
   $ EDUfirmwareUploadSYS.sh
   ```
   This script copies `rollingspider.edu.plf` to the root folder of the drone.
5. Wait for the LED on the drone to stop blinking. Disconnect the drone by ejecting the USB device and removing the USB cable
6. Remove battery
7. Insert battery
8. Wait until LEDs stopped blinking (firmware is now updated) (Note: If LEDs never blinked, redo step 1 and 3-8.)
9. Next, you need to make sure that Bluetooth can communicate with the drone. **This is where the interplay between the computer hardware, the base operating system, and the virtual**

machine causes the most problems. *Talk to us if you are having problems.* You may need to enable bluetooth by clicking on `Devices/USB` in the menu bar and selecting the appropriate bluetooth. For some people, the built-in bluetooth is adequate, others need to plug an external Bluetooth adapter into a USB port. When the VM has Bluetooth running, you will see the familar Bluetooth icon in the VM menu bar, somewhere over on the right side.

10. Connect drone to computer by running
    ```
    $ DroneConnect.sh
    ```
    This command establishes a Bluetooth connection to the drone. **This step will fail if Bluetooth is not properly working within the VM - see us if you cannot get this to work**.

11. Upload firmware files by running
    ```
    $ EDUfirmwareUploadFILES.sh
    ```
    This command uploads files to the drone.

12. Reboot drone:
    ```
    $ DroneReboot.sh
    ```

13. Connect drone again:
    ```
    $ DroneConnect.sh
    ```

14. Initialize drone firmware:
    ```
    $ EDUfirmwareInitialize.sh
    ```
    This script moves firmware files to right locations and grants permissions rights.

15. Initialize drone:
    ```
    $ DroneInitialize.sh
    ```
    This script writes the computers IP address to the drones parameter file.

16. Done with flashing. Nice!

Connecting and disconnecting from the Drone

In general, with all of the Drone firmware installed and intialized, from this point on, to connect or disconnect the drone the following commands can be used.

```
$ DroneConnect.sh
$ DroneDisconnect.sh
```

Testing the Drone

In this section we compile the default PID controller included in the toolbox and upload it to the drone. We then perform a test flight with the drone using this controller.

First, we compile and upload a controller to the drone:

1. Start the VM and open Matlab by running
   ```
   $ matlab
   ```
   in a terminal.

2. In Matlab navigate to `/home/student/RollingSpiderEdu/MIT_MatlabToolbox/trunk/matlab` and run `startup.m` in the Matlab command window.

3. Then open `sim_quadrotor.slx` located in the `Simulation` folder. This is a Simulink model of the drone dynamics in closed loop with a controller. The controller is implemented in the `Drones_Compensator` block. **This will take a long time if Simulink is starting for the first time in the Matlab session**.

4. Right click on the `Drones_Compensator` block and select `C/C++ Code > Build This Subsystem` to compile the controller. This will open a window as in Figure 1. In this window click the `Build` button. **Since your laptop screen is quite small, this step can get unruly, as several windows appear, and one of them needs your attention to continue the build. Carefully look through to find the window that needs confirmation**.

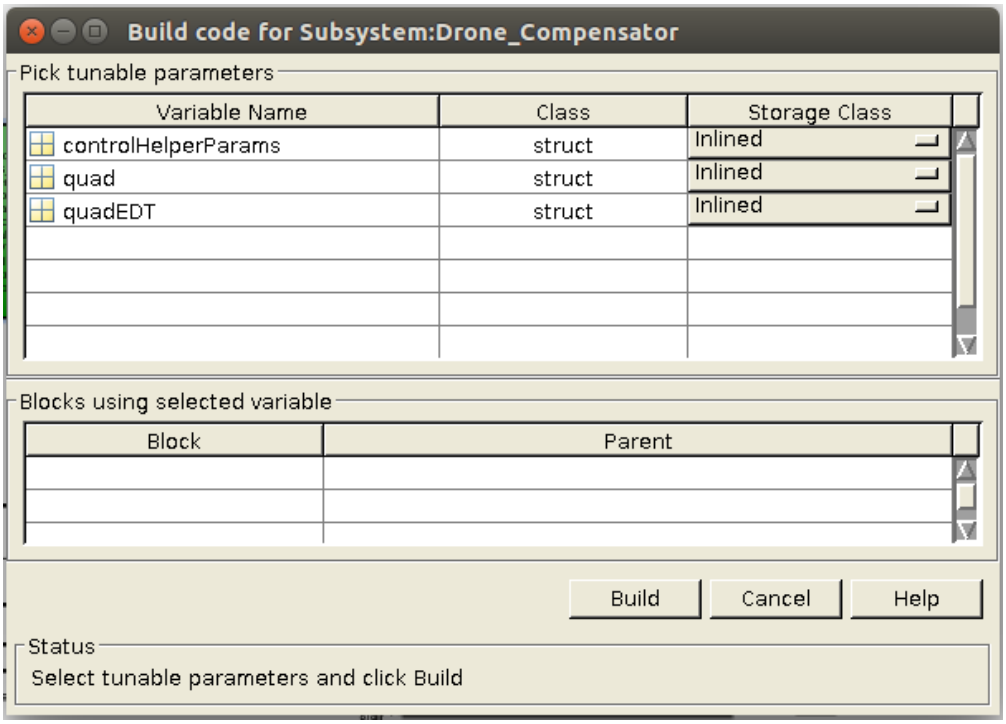

Figure 1: Dialog window to compile the `Drone_Compensator` subsystem.

5. Upload the controller to the drone by connecting to the drone and running
   ```
   $ DroneUploadEmbeddedCode.sh
   ```
   in the terminal.

Now, we can test the drone:

1. In a terminal run
   ```
   $ DroneKeyboardPilot.sh
   ```
   The message "Waiting for connection to drone..." will be printed in the terminal. Now move to the next step.

2. Open another terminal and run

   ```
   $ DroneTest.sh
   ```

   This script will test the drone at low power, so the drone propellers will spin but the drone will not lift off.

3. If the test is successful (i.e. the drone propellers spin and no errors appear in the terminal window) run

   ```
   $ DroneKeyboardPilot.sh
   ```

   again (in the window it was previously run) and in the other window run

   ```
   $ DroneRun.sh
   ```

   This script tests the drone with full power so the drone will lift off and then hover. It takes several seconds before it starts, but then it starts quickly. Be careful!

4. While the drone is hovering you can control the drone via the keyboard. The controls are described in Figure 2. **For the controls to work, you must have re-established focus back in the window where you executed the DroneKeyboardPilot.sh command. So, in the short pause that occurs after typing DroneRun.sh, but before the Drone takes off, move the mouse back into the Keyboard window, and click to make the focus there. The test will abruptly (and unexpectedly!) end after about 20 seconds. Be careful!**

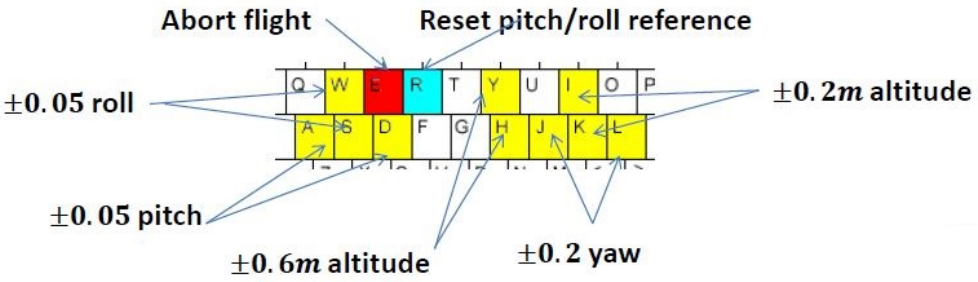

Figure 2: Keyboard controls from the MIT Getting Started Document

Now, we can download and analyze the data from the test flight:

1. In a terminal run

   ```
   $ DroneDownloadFlightData.sh
   ```

   This script will download the data file from the drone and save it to `/home/student/RollingSpiderEdu/MIT_MatlabToolbox/DroneExchange`
   The data file will be named `RSdata_timestamp.mat` where "timestamp" will be the date and time the file was downloaded from the drone.

2. In Matlab navigate to the data file and double click on it to load it.

3. Now, you can run the Matlab script `FlightAnalyer.m` (which is located in the `MIT_MatlabToolbox/trunk/matlab/ExperimentAnalyzer/Flight` folder) to extract and view the data.

Resetting the Drone

In this section we describe how to reset the drone to the factory settings. This will overwrite the new firmware that was installed with the original Parrot firmware. Note that this section is just for reference. It is not necessary that you complete the steps in this section. **WARNING! If you do this you will only be able to control the drone via the Parrot application.**

1. Connect the Drone via USB.
2. Go to `https://community.parrot.com/t5/Rolling-Spider/bd-p/rollingspider_EN`
3. On the left side of the page under `Downloads` select `Software update`. This will download a file. Do not change the name of this file.
4. Copy the downloaded file to the drone via USB. While the file is being transferred the lights on the drone will flash. Once the transfer is complete the right eye will be solid-green and the left-eye solid orange.
5. Eject the drone from the computer. The lights on the drone will flash while the update is being performed. Once the lights turn green the update is complete.

For more information about this procedure go to:
`https://community.parrot.com/t5/Rolling-Spider-Knowledge-Base/Software-Update-Process/ta-p/122999`