Research Internship

December 3, 2019

# Swarm Drone Overview

**Author: James Allen**
**Supervisor: David Saussié**

POLYTECHNIQUE MONTRÉAL

UNIVERSITÉ D'INGÉNIERIE

# 1 Overview

This document will serve as an additional resource to the Bitcraze tutorials, with more specific information being included that could not be found on the website. After the testing of these drones, it appears that there is a fault/issue with one of the Crazyflie boards. Apart from this, all other components worked as expected, and all firmware has been updated on both the drones and LPS nodes. Although all components were included, a number of 3D printed brackets were printed for the mounting of the LPS nodes throughout the room. After the mounting of these nodes, the swarm drones can be programmed as desired.

# 2 Components

Before testing the materials and assembling the drones, the received items were checked to ensure no components were missing. As shown in the table below, the ordered numbers matched the received numbers.

**Table 1:** *Ordered Quantity vs Received Quantity*

| Description of Goods | Order Quantity | Received Quantity |
|---|---|---|
| Crazyflie 2.1 | 10 | 10 |
| Crazyradio PA module - long range 2.4GHz | 3 | 3 |
| Crazyflie 2.0 Battery Charger Board bundle | 20 | 20 |
| DWM1000 Node | 8 | 8 |
| Crazyflie 2.0 DWM1000 Deck | 10 | 10 |

In addition to this, the Crazyflie 2.1 contents were also checked. Each box should contain the following parts:

- 1 x Crazyflie 2.X control board with all components mounted
- 5 x CW propellers
- 5 x CCW propellers
- 6 x Motor mounts
- 1 x LiPo battery (240mAh)
- 5 x Coreless DC motors
- 2 x Short expansion connector pins (1x10, 2mm spacing, 8mm long)
- 2 x Long expansion connector pins (1x10, 2mm spacing, 14mm long)
- 1 x Battery holder expansion board
- 1 x USB cable (only with the Crazyflie 2.1)

All of the above components were included in all 10 of the Crazyflie 2.0 boxes.

# 3 Preliminary Tests

Before assembling the drones, the USB cord was connected into the Crazyflie 2.1 to ensure no damage occurred during shipping. If LED M4 flashes green 5 times after connecting to a power source, this indicates the test has been passed. However, if LED M1 flashes red 5 times, this indicates that the test has failed and there may be damage to the drone. The following results were obtained after testing.

| Crazyflie 2.1 | Test Passed: Y/N |
|---|---|
| Drone 1 | Y |
| Drone 2 | Y |
| Drone 3 | Y |
| Drone 4 | Y |
| Drone 5 | Y |
| Drone 6 | Y |
| Drone 7 | Y |
| Drone 8 | Y |
| Drone 9 | Y |
| Drone 10 | Y |

# 4 Assembly

After this, the Crazyflie 2.1 drones were assembled, with the LPS Deck (DWM1000 Deck) replacing the battery holder board. It is important to note that the orientation of this board should be the same as the figure below to ensure no damage occurs.

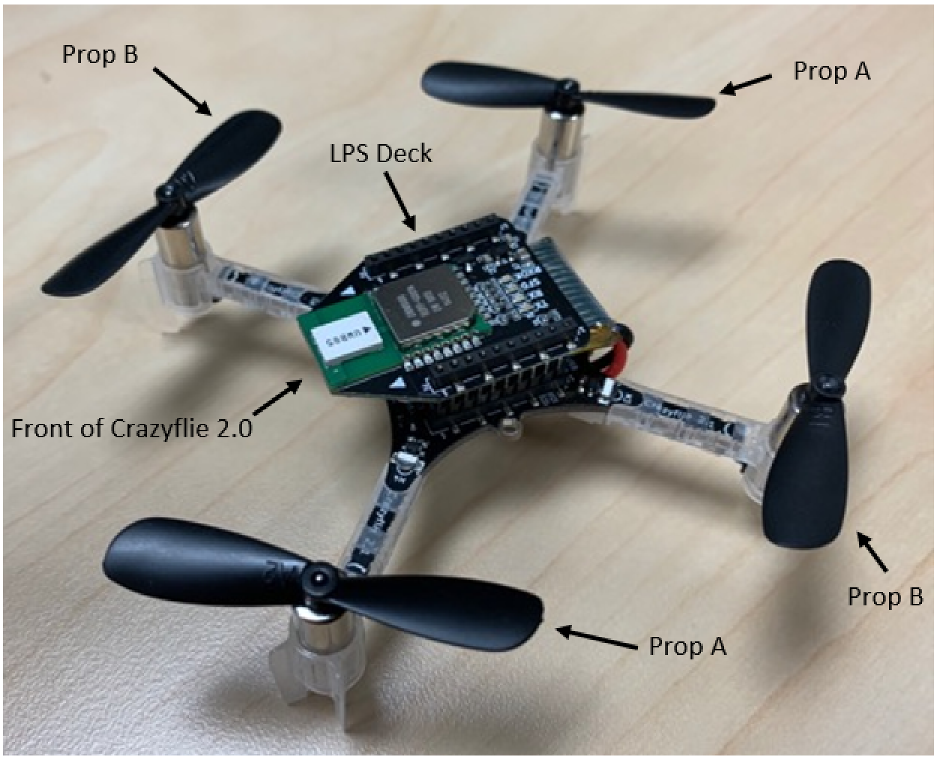


**Figure 1:** *Fully assembled drone with LPS deck (communicates with LPS nodes placed through room)*

After turning on each of these completed drones, it was clear that despite the start up tests being passed, there may in fact be some sort of damage to one of the boards. The LPS deck on all drones except for one appears to flash between TX, RX and SFD. The board that does not seem to flash only appears to flash once on start up (all LEDs light up red), followed by a single yellow flash from the RX LED. After this, no flashing happens. It is important to note that the LPS deck works as expected on other drones, along with the battery also being replaced and all firmware being updated on the drones. For this reason, it seems to either be a connection issue or internal issue with the Crazyflie main board.

For further information, refer to the 'Getting started with the Crazyflie 2.X' tutorial:
`https://www.bitcraze.io/getting-started-with-the-crazyflie-2-0/`

## 5 Configuring Crazyflie Address

In order to use more than 1 Crazyflie per radio, it is important that a unique address is configured to each drone. This section will both explain how the drones were configured, but more importantly how each Crazyflie address can easily be recognised at a glance. To change the address of each Crazyflie, the PC Client will be used as this makes the configuration quite simple.

- Before beginning, ensure all tabs are closed to reduce lag of the GUI. Do this by clicking 'View', 'Tabs' and then deselecting any tabs currently ticked.
- Plug in PA Radio
- Turn on Crazyflie
- Ensure address is set to: radio://0/80/2M/E7E7E7E7E7 (this procedure assumes the address has not been set in the past. If it has, then change this to the current address)
- Select 'scan' on the Crazyflie Client (radio will show up)
- Select 'connect' on the Crazyflie Client
- After successfully connecting, click 'Connect' and then 'Configure 2.X'. A small interface will be displayed and the address can be adjusted such that the first Crazyflie will be radio://0/80/2M/E7E7E7E700, the second will be radio://0/80/2M/E7E7E7E701 and the thrid will be radio://0/80/2M/E7E7E7E702. After connecting 3 to 1 PA Radio, the same procedure can be repeated with the channel dropping by 10. For example, the fourth will be radio://0/70/2M/E7E7E7E703.

The drones were marked in the following way and this number corresponds to the number in the address. This ensures no connections are mixed up. For example, the drone below would be set as radio://0/80/2M/E7E7E7E701.

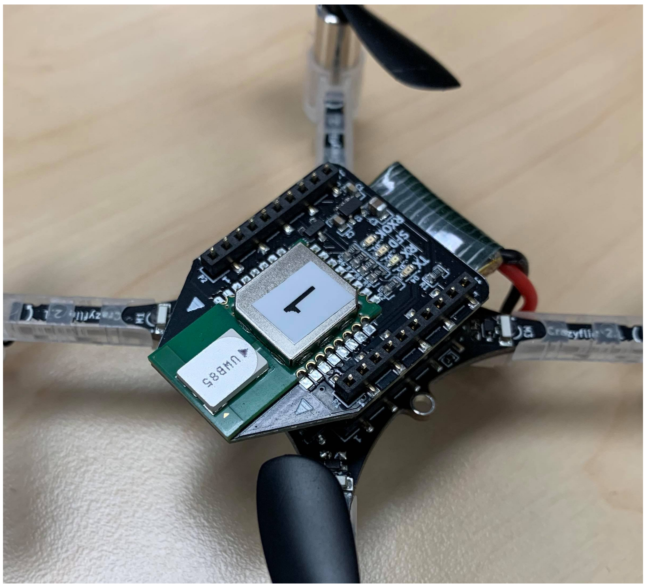

**Figure 2:** *Number on Crazyflie corresponding to address*

## 6 Loco Positioning System

The LPS nodes (DWM1000 Node) were then configured after updating the firmware of each of the nodes. In order to do this, the program Zadig was used, with this allowing both the Crazyradio PA module and LPS nodes to be recognised drivers in the computer. From this, it was a simple matter

of following the instructions on the Bitcraze website. After successfully setting up all elements of the LPS system, each of the nodes were configured with a unique ID from 0 to 7. Each LPS node has been marked with the respective number, allowing the nodes to be positioned in the order shown in the tutorial. It should also be noted that the LPS nodes were configured for TWR mode, even though TDoA2 mode will be used seeing as there are multiple drones, along with 8 nodes instead of 6. However, it is far easier to configure them for TWR initially and then change to TDoA2 after. For more information, refer to the following tutorial:
`https://www.bitcraze.io/getting-started-with-the-loco-positioning-system/`

After this, the nodes should be placed around the room in the configuration shown in the figure below. For initial testing, the provided layout will be used with the origin being in the bottom corner next to node 0. When measuring the distance between nodes, the Vicon system was used to ensure highly accurate node positions were configured. The x and y distances should be configured at any constant distance, however the z height of the nodes does not need to be at the same level (having them at different heights improves the accuracy).

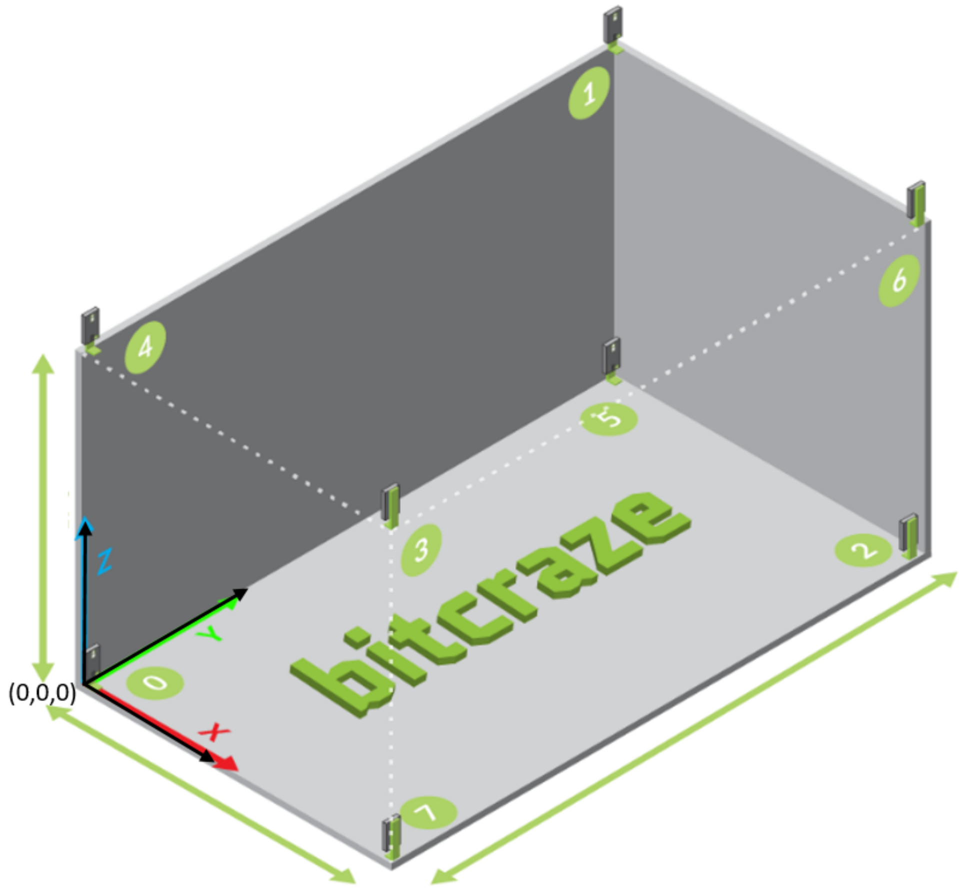


**Figure 3:** *Coordinate system used for initial LPS testing*

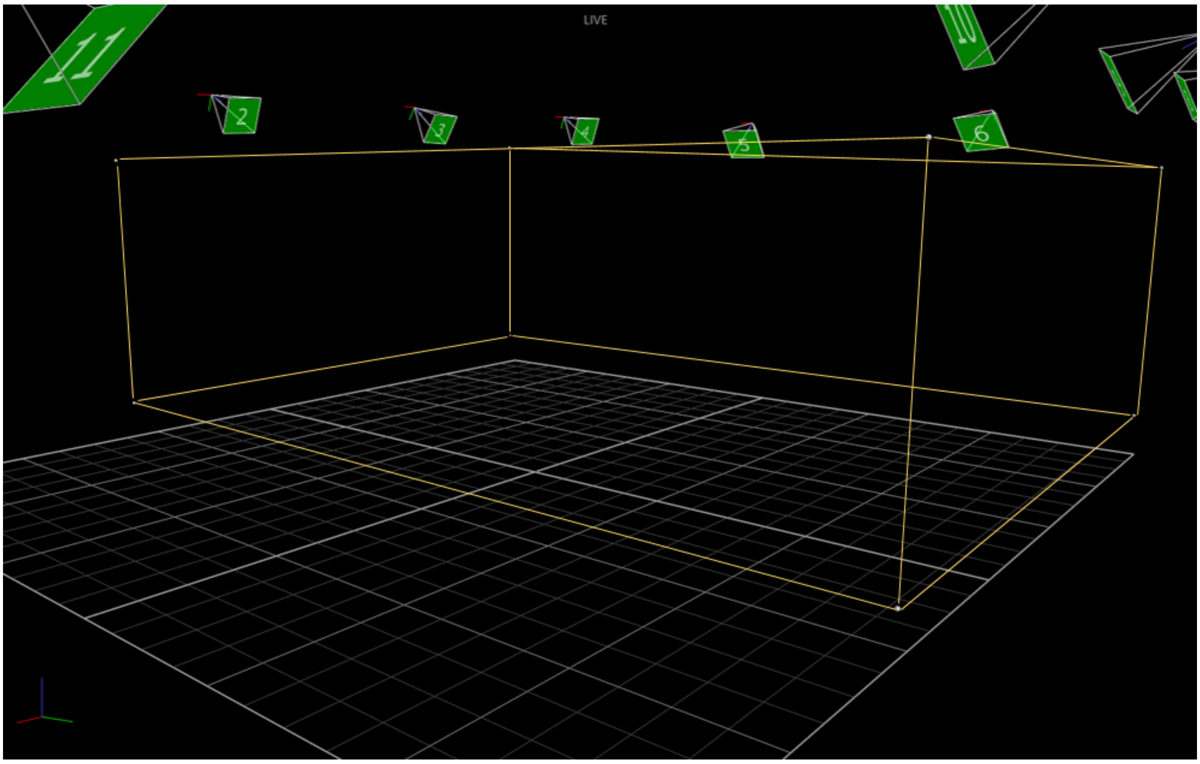


**Figure 4:** *Vicon positioning. Different z heights (at the top) but constant x and y distances between nodes*

It is important to note that it may be more effective to change this origin to the centre of the room, instead of in the corner. This is because the initial position in the Kalman filter is set to $(0, 0, 0)$, meaning it will converge faster if the drone is placed closer to the point on start up. However, seeing as the above figure was provided as a working example, this will still be used. If significant positioning errors occur though, the origin will be changed to the middle. The following coordinates were chosen for each node, with this shown in the table below:

| Anchor ID | X | Y | Z |
|---|---|---|---|
| 0 | 0 | 0 | 0.26 |
| 1 | 0 | 6.01 | 1.98 |
| 2 | 4.17 | 5.99 | 0.26 |
| 3 | 4.17 | 0 | 2.04 |
| 4 | 0 | 0 | 2.20 |
| 5 | 0 | 6.01 | 0.26 |
| 6 | 4.17 | 5.99 | 2.08 |
| 7 | 4.17 | 0 | 0.26 |

**Table 2:** *Anchor positions based on figure 2*

These coordinates resulted in the final setup as shown below.

**Figure 5:** *Nodes positioned at the coordinates given above*

After selecting these positions and placing the anchors at these locations, it is possible to configure the anchors using the Crazyflie PC Client. In order to configure the nodes on the client, the following setup procedure should be followed to ensure they are setup correctly and ready to use.

- Place LPS nodes at the required locations
- Power LPS nodes
- Plug in PA Radio
- Turn on Crazyflie/s
- Change address to match Crazyflie/s being used (refer to previous section on setting the address of each Crazyflie)
- Select 'scan' on the Crazyflie Client (radio will show up)
- Select 'connect' on the Crazyflie Client
- Ensure the 'Mode' LED is lit up on the LPS nodes and that the Crazyflie and LPS node has all 4 LEDs flashing
- The Crazyflie is now connected with the LPS nodes, and this can be further confirmed by checking if the status boxes in the Client are green (this is found by clicking 'View', 'Tabs', 'Loco Positioning')

The only component not included in the kit that might be useful for setting up these LPS nodes is a 3D printed bracket. 8 of these brackets were printed, allowing the nodes to sit 15cm away from the walls, with this ensuring that minimal interference occurs. Refer to the link for the STL file if any more brackets need to be printed:
`https://github.com/bitcraze/bitcraze-mechanics/blob/master/LPS-anchor-stand/anchor-stand.stl`

## 7 Python Setup for Windows

It is important that the PC Client is closed or disconnected otherwise the Python script will not work

- Use Python 3 or greater
- Download 'crazyflie-lib-python-master' and 'crazyflie-clients-python' from Github
- Open a command window and set this as the current directory
- Type 'pip3 install cflib'
- Plug in PA Radio
- (For PyCharm) Set 'crazyflie-lib-python-master' as current folder
- Turn on Crazyflie and run ramp.py (always check that the correct URI is set). If it works, the PA Radio will make a connection to the Crazyflie and the propellers will rotate for a few seconds but will not spin fast enough for the drone to take off

## 8 Loco Positioning System using Python

- 1 Crazyflie can be tested first by running the script 'autonomousSequence.py' (again, ensure correct URI is set by checking number marked on Crazyflie)

Multiple Crazyflies can now be configured and then a script can be run on Python:

- Close Pycharm and open the PC Client
- Ensure all nodes are powered and Crazyflie is connected
- Select TWR mode in the 'Loco Positioning' tab
- Select TDoA2 mode from the Anchor Status
- Change back to Auto and then TDoA2 mode will be automatically detected
- Close PC Client and open Pycharm
- Test with 2 crazyflies by running the script 'hl-commander-swarm.py' and ensuring both URI addresses are set correctly

## 9 Overall Summary

Using Python, it is now possible to control a swarm of crazyflie drones by simply adding more to the swarm and including their appropriate URI. The code can then be adjusted as needed and any data can be recorded using the VICON system and saving to a CSV file. This can then be exported into MATLAB and analysed as necessary.