

Objectifs

Objectifs S1 : Découverte des drones, prise en main du drone LiteWing

- Présentation générale (Quoi, Qui, Objectifs, Contexte)
- Comprendre ce qu'est un drone (exemple), ses paramètres (roll, pitch, yaw), son architecture.
- Qu'est ce qu'une commande ? une consigne ? une loi de commande ?
- Mettre en place l'environnement python CrazyFlie
- Se connecter au drone et faire le 'hello test'
- Lire et observer les mesures de roll, pitch, yaw, z
- Piloter avec une manette / clavier
- Piloter avec un programme automatique
- Piloter de manière automatique et intelligente

Objectifs S2 : Loi de commande, introduction à l'automatique

- Expérimenter différente loi de commande
- Découvrir ce qu'est un PID
- Découvrir l'influence des différents facteur du PID
- Mettre en place un petit pilote automatique simulé

Objectifs S3: Piloter son propre drone commandé par son programme

- Mise en place d'une loi de commande sur le drone LiteWing
- Essai grandeur nature dans la volière (Automatique ou piloté)

Ressources :

Wiki : <https://circuitdigest.com/litewing>

Github liteWing : <https://github.com/Circuit-Digest/LiteWing/tree/main>

Presentation : [youtube.com/watch?v=esmYcBHqBK8](https://www.youtube.com/watch?v=esmYcBHqBK8)

Python starter : <https://circuitdigest.com/microcontroller-projects/how-to-program-litewing-drone-using-python-with-crazyflie-cflib-python-sdk>

Github du Projet : <https://github.com/Arthur1459/LiteWing-Drone-Project>

Requirements : python 3+, pip, éditeur python (**pycharm**, ...)

Library : pygame, sys, time, hid, threading, cflib,

Environnement + cflib (inclu) :

Installation : (se placer dans un répertoire adapté, ex « Document/LiteWing »)

```
git clone https://github.com/Arthur1459/LiteWing-Drone-Project.git
pip install -r requirements.txt
cd crazyflie-clients-python
```

```
export SETUPTOOLS_SCM_PRETEND_VERSION=2026.1          (LINUX / MAC)
$env:SETUPTOOLS_SCM_PRETEND_VERSION="2026.1"          (PowerShell)
```

```
pip3 install -e .
```

Test cfclient :

```
cd crazyflie-clients-python
cfclient
```

Test python + connection + drone :

- power on the drone
- connect to the drone wifi (ex: « LiteWing-23EZ32JE »)
- mot de passe « 12345678 »
- run 'hello_litewing.py'

Pour installer pygame et hid :

```
pip install -r requirements.txt
```

pygame :

Installation :

```
pip install pygame
```

hid :

Installation :

```
pip install hidapi
```

Utilisation :

```
import hid
device = hid.device()
device.open(0x0079, 0x0006)  # Vendor ID et Product ID de
votre manette

import hid
for device in hid.enumerate():
    print(f"Device: {device['product_string']} (VID:
{device['vendor_id']:04x}, PID: {device['product_id']:04x})")
```

Pour setup un environnement virtuel :

```
python -m venv venv          # Crée un environnement virtuel
source venv/bin/activate     # Activez-le (Linux/macOS)
venv\Scripts\activate        # Activez-le (Windows)
pip install -r requirements.txt
```

DEBUG :

Git :

tester installation / path : `git --version`

Si git n'est pas reconnu comme commande : Ajouter au PATH

1. Ouvrez le Panneau de configuration > Système > Paramètres système avancés > Variables d'environnement.
2. Dans la section Variables système, trouvez la variable Path et cliquez sur Modifier.
3. Ajoutez le chemin vers le dossier bin de Git (par exemple, c : \Program Files\Git\bin).
4. Cliquez sur OK pour enregistrer les modifications.

Python :

tester installation : `python --version`

si python n'est pas reconnue : Ajouter au PATH

- Dans la section "Variables système", trouvez la variable Path et cliquez sur "Modifier".
- Cliquez sur "Nouveau" et ajoutez les chemins suivants (adaptez les versions) :
C:\Users\<votre_utilisateur>\AppData\Local\Programs\Python\Python312
C:\Users\<votre_utilisateur>\AppData\Local\Programs\Python\Python312\Scripts
- Cliquez sur "OK" pour enregistrer les modifications.

Commande sur windows PowerShell par rapport à linux (en cas de problème) :

```
pip -> python -m pip
cflient -> python -m crazyflie.clients.cfclient
```

Si des erreurs avec setuptools-scm :

`pip install setuptools-scm`
ou juste si des erreurs de version :

```
export SETUPTOOLS_SCM_PRETEND_VERSION=2026.1          (LINUX / MAC)
$env:SETUPTOOLS_SCM_PRETEND_VERSION="2026.1"          (PowerShell)
pip3 install -e .
```