

Manuel de reproduction tangible du tracker de vol à voile fait maison

INDEX

Contenus

Contenus.....	3
Introduction	4
Préparation du Raspberry Pi Zero W.....	6
Installation du logiciel sur le Raspberry PI.....	12
Assemblage du suiveur de vol	19
Récupération des données de vols.....	29
Tests.....	31

Introduction

Dans le monde de l'aviation légère et des sports aériens, les planeurs occupent une place particulière en raison de leur dépendance à la météo et de leur capacité à glisser silencieusement à travers les cieux. Pour améliorer l'expérience de vol des planeurs et fournir aux pilotes des données essentielles en temps réel, mon projet de recherche de fin de bachelor en systèmes d'information et science des services s'est concentré sur la conception et la mise en œuvre d'un prototype de tracker de vol fait maison spécialement conçu pour ces aéronefs.

Les trackers de vol traditionnels que l'on trouve sur le marché sont relativement coûteux. Le prix de ces appareils se situe autour des 700 francs suisses. Grâce à la démocratisation des microcontrôleurs et microordinateurs, ce travail de recherche se base sur la faisabilité de création d'un suiveur de vol fait maison.

Il faut avant tout commencer par choisir les composants appropriés, en tenant compte de facteurs tels que la taille, le poids, la consommation d'énergie et le prix. Les principaux éléments du tracker comprenaient un récepteur GPS, un baromètre et une unité centrale qui traite et stocke les données.

Ce manuel vous fournira les informations essentielles, les instructions détaillées et les astuces pratiques nécessaires pour mener à bien votre projet.

Dans les pages qui suivent, vous découvrirez les procédures étape par étape pour vous guider dans la réalisation du tracker de vol. Vous y trouverez des schémas, des illustrations, des listes de matériaux, et tout ce dont vous avez besoin pour reproduire.

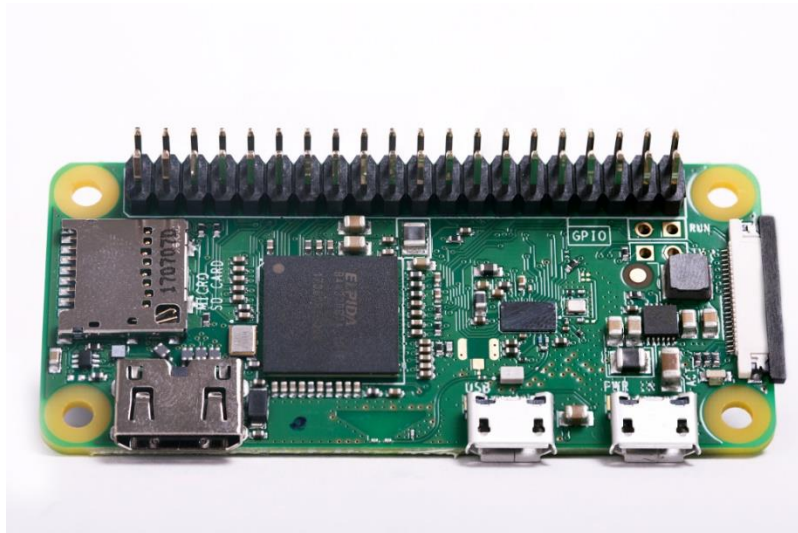
Pour la partie code, référez-vous au repo Git ci-dessous.

Lien du GitHub : https://github.com/Onyme13/Homemade_Glider_Tracker.git

Préparation du Raspberry Pi Zero W

Matériel

1. Raspberry Pi Zero W avec en-têtes (le modèle wifi est indispensable)



2. Carte microSD de 32Go minimum



1. Téléchargement des logiciel et le fichier requis sur votre PC

1.1 Advanced IP Scanner. Ce programme affiche toutes adresse IP des périphériques sur votre réseau :

<https://www.advanced-ip-scanner.com/>

1.2 Raspberry Pi Imager. Raspberry Pi Imager est le moyen rapide et facile d'installer Raspberry Pi OS sur une carte microSD, prête à l'emploi avec votre Raspberry Pi :

<https://www.raspberrypi.com/software/>

1.3 PuTTY. Ce programme est un client SSH principalement utilisé pour se connecter à distance à des systèmes informatiques:

<https://www.putty.org/>

1.4 FileZilla Client. C'est un programme de transfert de fichiers qui permet aux utilisateurs de gérer et de transférer des fichiers entre leur ordinateur local et une machine distante :

<https://filezilla-project.org/>

1.5 Téléchargement de carte sur Google Drive :

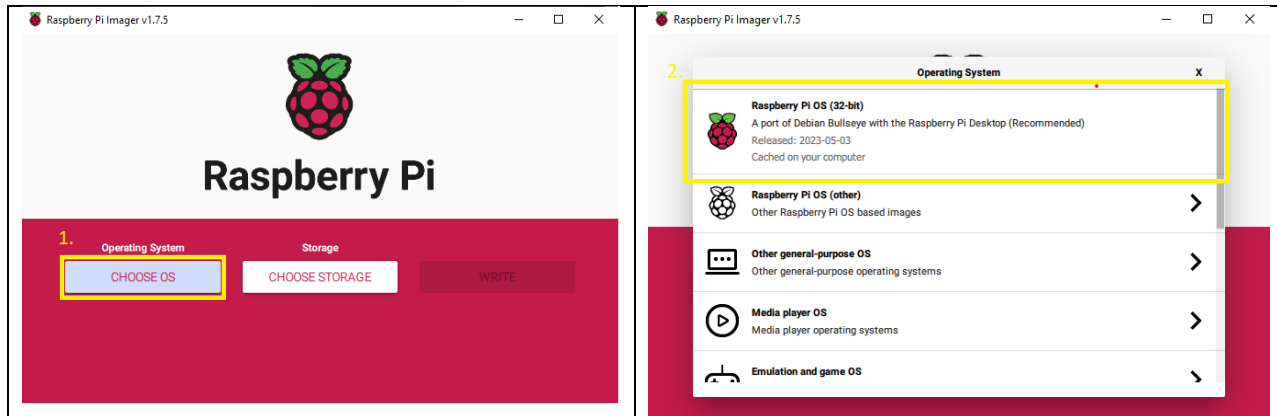
https://drive.google.com/file/d/1AfhuXc_kdtul0cSf0_6vRhANw7dOtnlN/view?usp=sharing

2. Installation Raspberry Pi OS

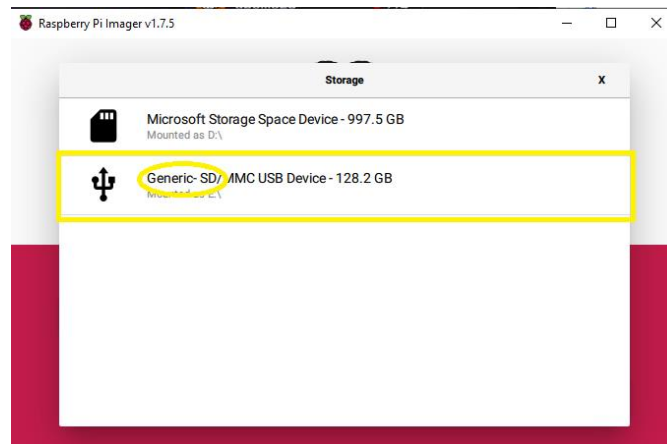
2.1 Insérez la carte SD dans le lecteur de carte de votre ordinateur.

2.2 Ouvrir le logiciel Rasperry PI OS.

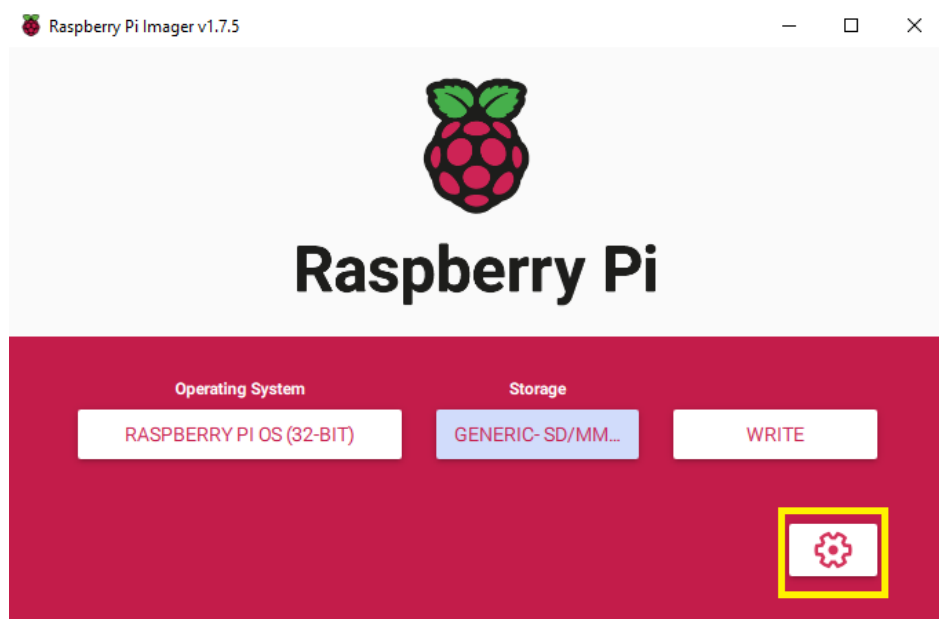
2.3 Choisir le bon OS



2.4 Choisir la carte microSD comme stockage

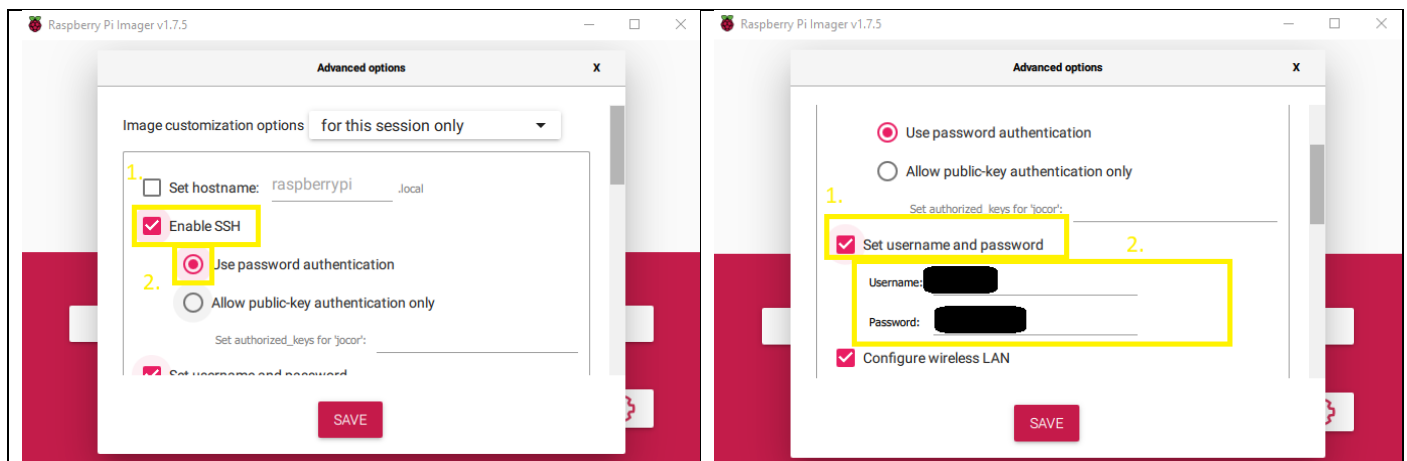


2.5.1 Configuration de l'OS

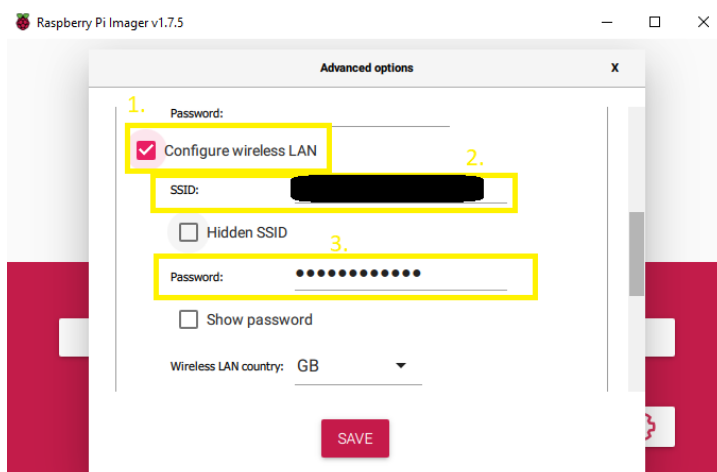


2.5.2 Activer le SSH et ajouter un Username et mot de passe

Sauvegarder l'identifiant et mot de passe pour l'étape *Installation du logiciel sur Raspberry Pi*

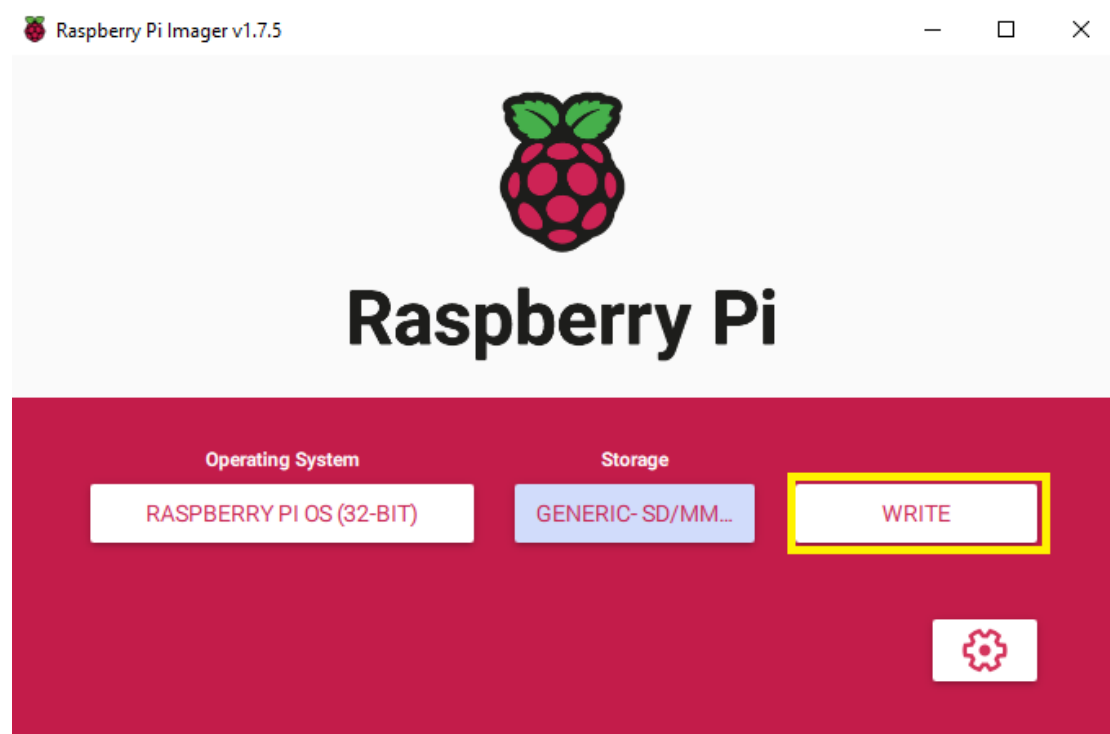


2.5.3 Configurer la connexion au WiFi



1. Activer la connexion wLAN
2. Sélectionner le réseau local. **Attention**, le Raspberry Pi ne connecte pas au réseau de 5GHz
3. Entrer le mot du passe du réseau

2.5. Installer l'OS



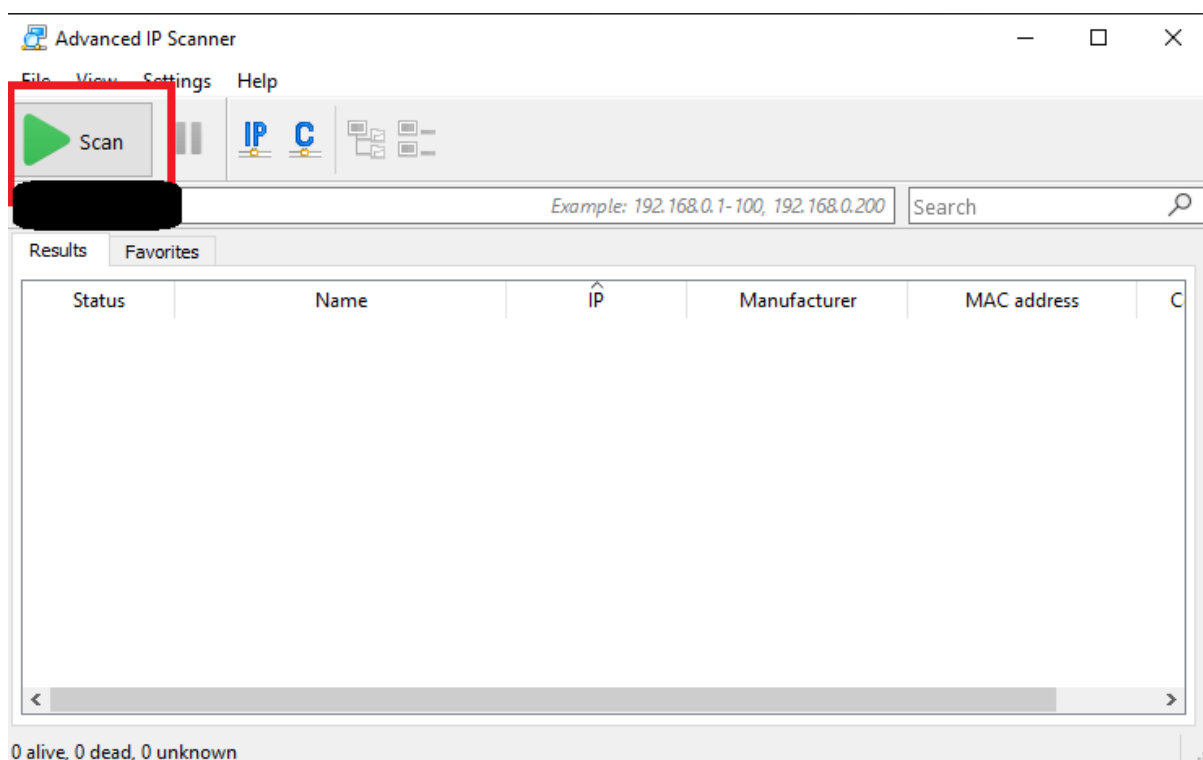
Installation du logiciel sur le Raspberry Pi

1. Insérer la carte microSD dans le Raspberry Pi

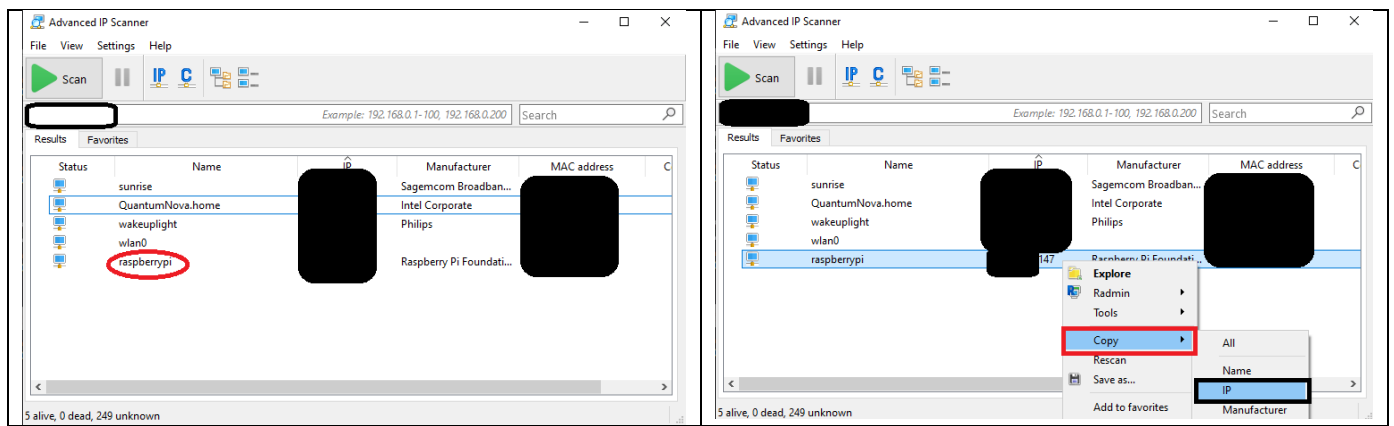
2. Allumer le Raspberry Pi

3. Trouver l'IP du Raspberry Pi

3.1 Lancer le scan avec Advanced IP Scanner



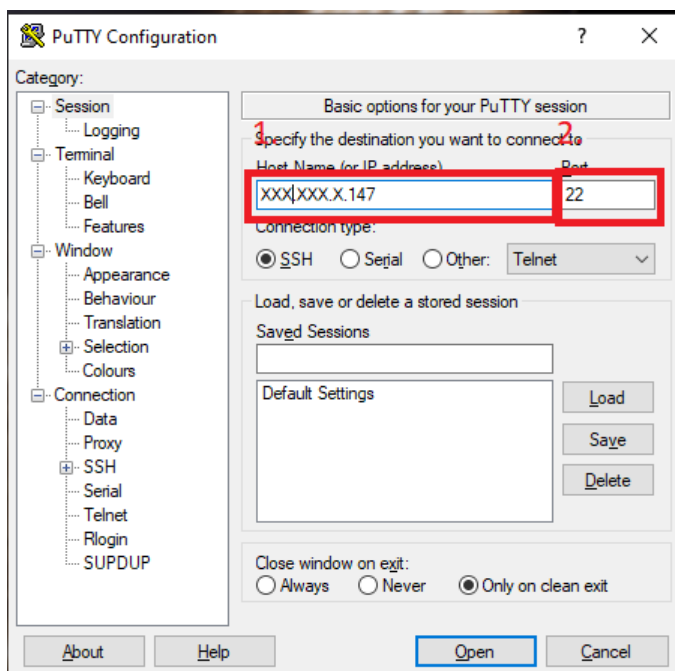
3.2 Copier l'IP du Raspberry Pi



4. Connection via SSH

4.1 Ouvrir puTTY.exe

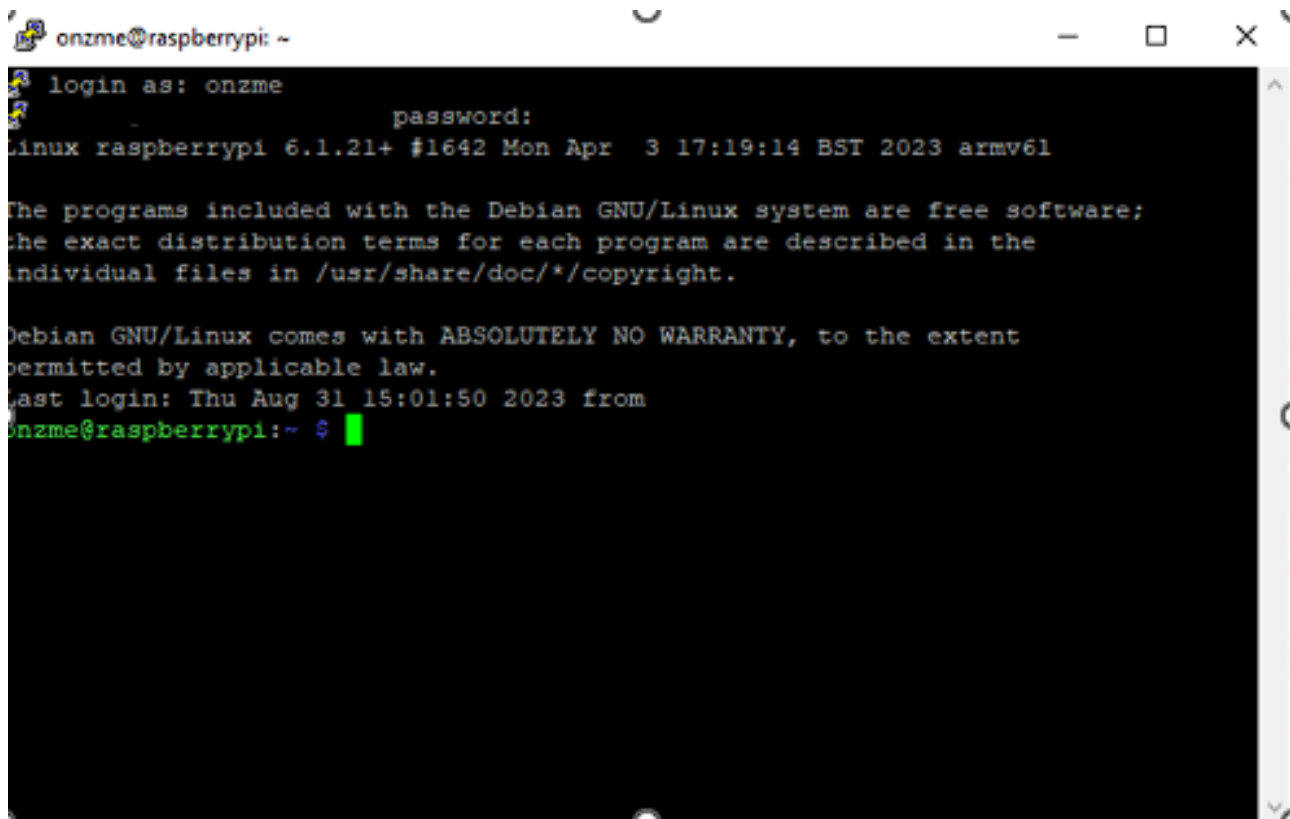
4.2 Connection au Raspberry Pi



1. Coller l'IP du Raspberry Pi
2. Port 22 nécessaire

5. Téléchargement du logiciel

5.1 Connection au Raspberry avec les identifiants de l'étape *Préparation du Raspberry Pi*



```
onzme@raspberrypi: ~  
login as: onzme  
password:  
Linux raspberrypi 6.1.21+ #1642 Mon Apr 3 17:19:14 BST 2023 armv6l  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
last login: Thu Aug 31 15:01:50 2023 from  
onzme@raspberrypi:~ $
```

5.2 Mettre à jour le Raspberry Pi OS en exécutant

```
sudo apt-get update  
sudo apt-get upgrade
```

5.3 Cloner le git repository en exécutant

```
git clone https://github.com/Onyme13/Homemade\_Glider\_Tracker.git
```

5.4 Installer Pip et aller dans le fichier et installer toutes les bibliothèques python nécessaires en exécutant :

```
sudo apt install python3-pip  
pip -version
```

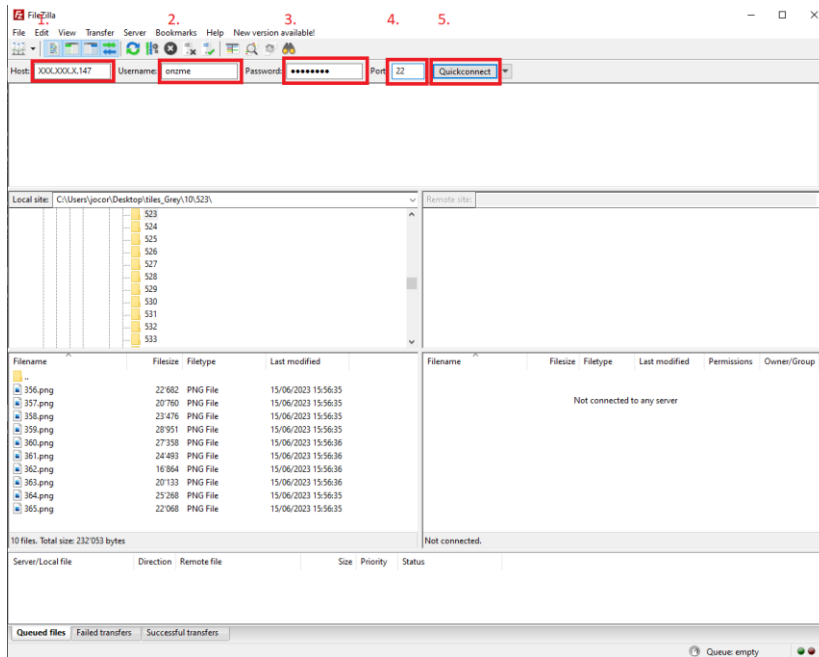
5.5 Après l'installation de pip, vous pouvez vérifier que pip est installé. Ensuite, installez les bibliothèques python nécessaires

```
cd Homemade_Glider_Tracker
```

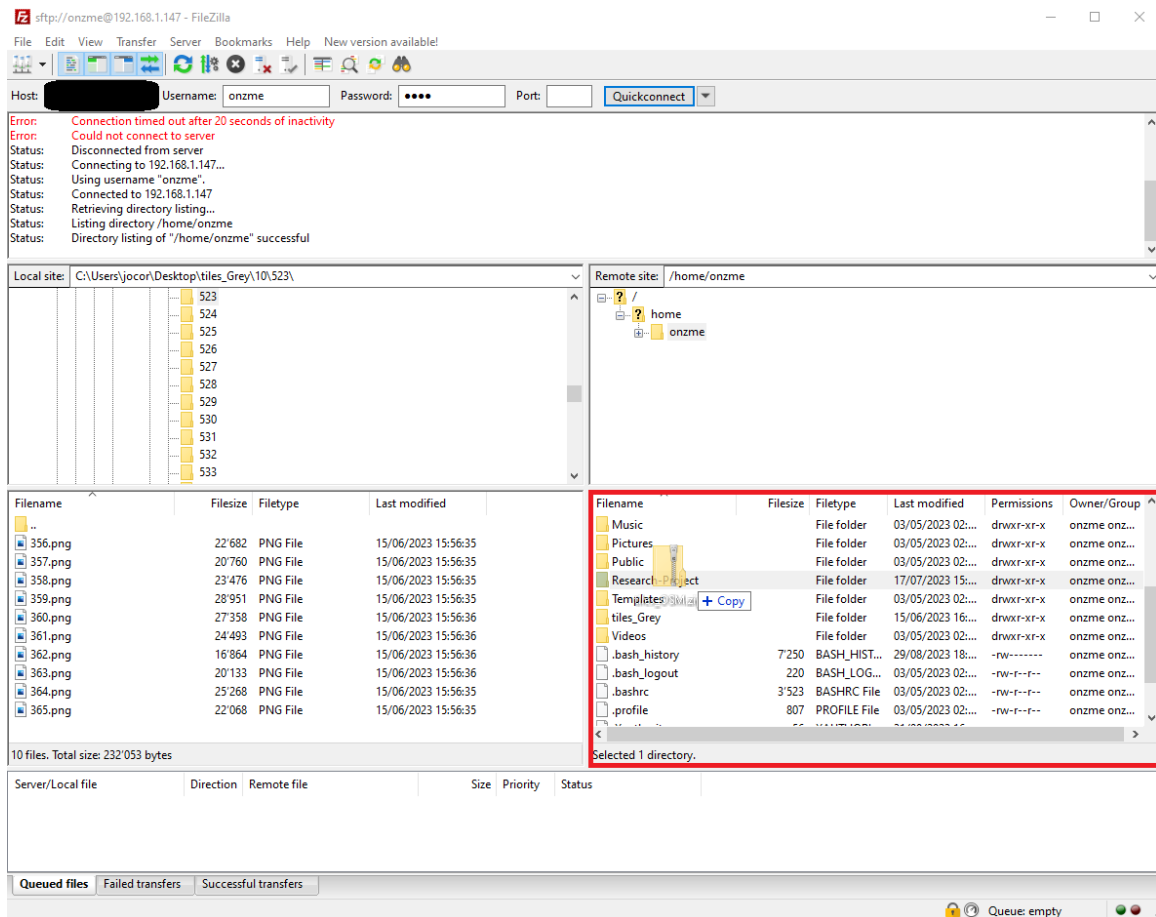
```
sudo python -m pip install -r requirements.txt
```

5.6 Transférez le fichier zippé de la carte à l'aide de FileZilla.

Drag and Drop le dossier de la carte zippé de l'étape *Préparation du Raspberry Pi*



1. IP du Raspberry PI
2. Identifiant
3. Mot de passe
4. Port 22
5. Connection



5.7 Décompressez le fichier en l'exécutant sur le Raspberry Pi en exécutant :

```
sudo apt install unzip
unzip tiles_OSM.zip
```

6. Activation du Touchscreen

6.1 Activer les connecteurs SPI

```
sudo raspi-config
```


6.2 Installer les fichiers nécessaire pour l'écran.

```
sudo rm -rf LCD-show
git clone https://github.com/MrYacha/LCD-show.git
cd LCD-show
chmod LCD35-show
```

6.3 Finalement, le Raspberry Pi va redémarrer.

```
sudo ./LCD35-show
```

7. Ajout d'icône de lancement du logiciel sur le Desktop du Raspberry Pi

7.1 Copie des données de l'icône sur le chemin du Desktop

```
cp Homemade_Glider_Tracker/launch.desktop ~/Desktop/
```

7.2 Ajout des droits d'exécution du script

```
sudo chmod +x Homemade_Glider_Tracker/startup.sh
```

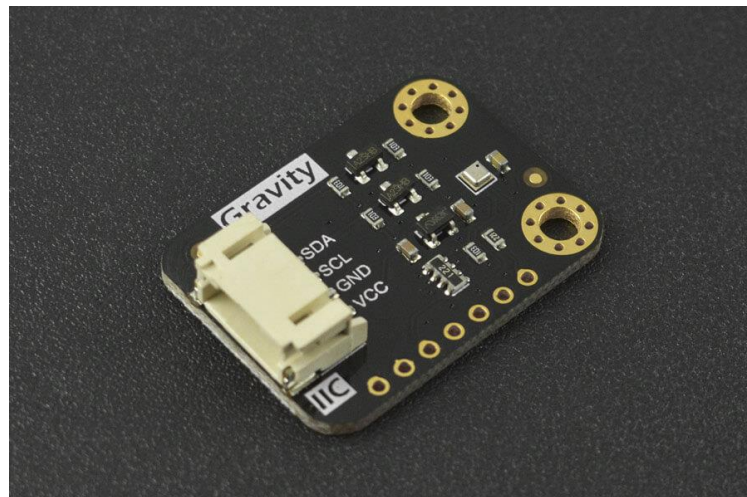

Assemblage du suiveur de vol

Matériel

1. Module GPS - Whadda WPI430 Module GPS U-BLOX NEO-7M



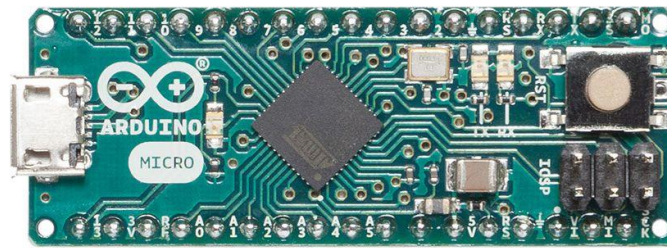
2. Module Baromètre - DFRobot Gravity BMP388 Capteur de pression barométrique



3. Batterie PiSugar - PiSugar Pwnagotchi Batterie au lithium pour Raspberry Pi-Zero W/WH 1200 mAh



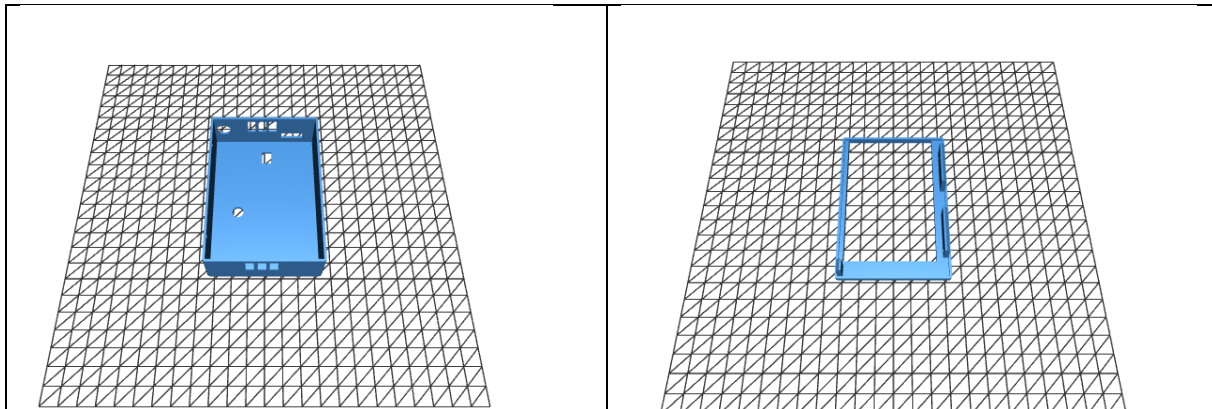
4. Arduino Micro (sans les pins/headers)



5. Câbles DuPont



6. Boîtier imprimer en 3D (voir les étapes suivantes)

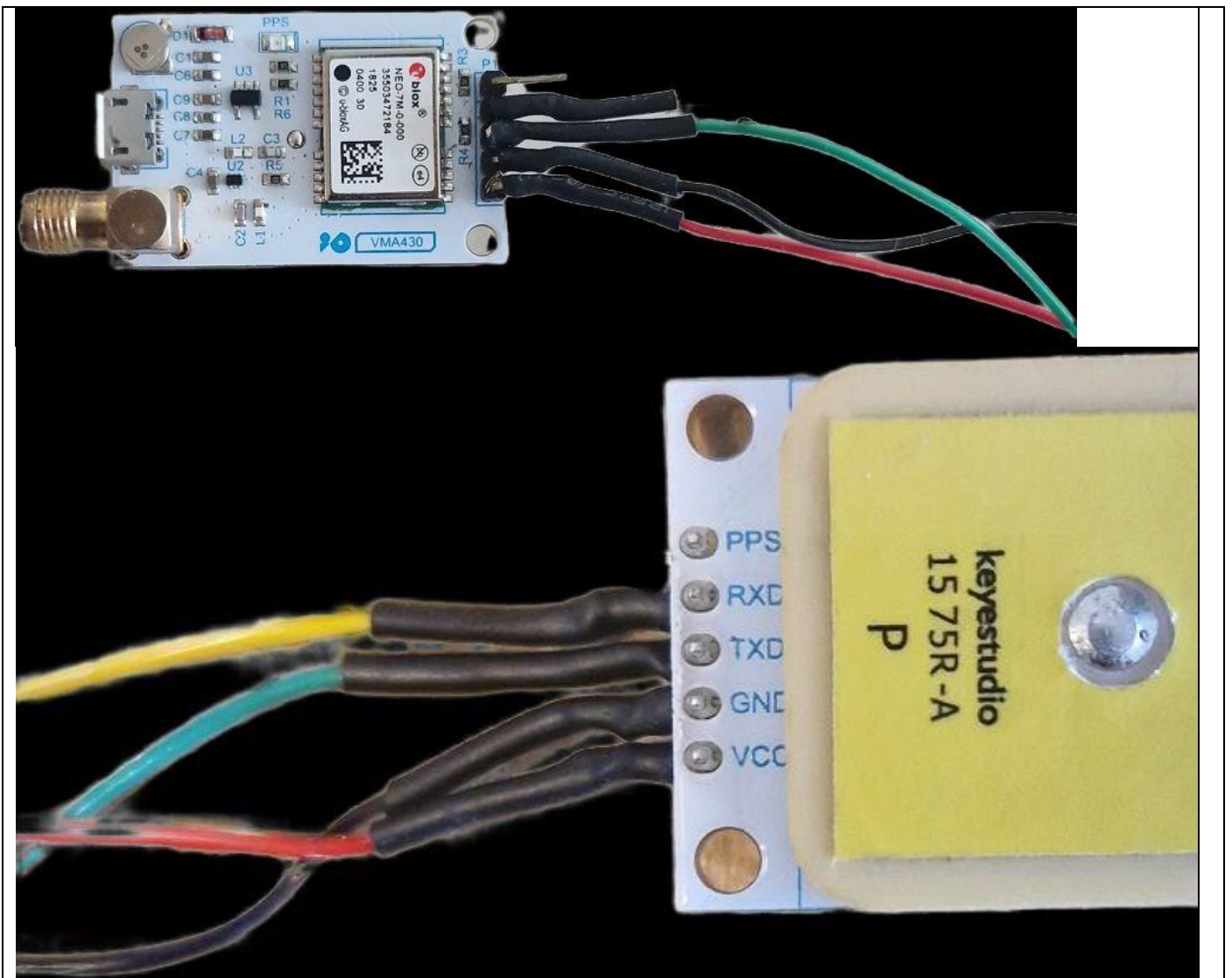


1.Impression du boîtier et de son couvercle en 3D

Fichier STL pour l'impression

- Boîtier :
https://github.com/Onyme13/Homemade_Glider_Tracker/blob/main/3D%20STL%20file/case%20v0.2.stl
- Couvercle :
https://github.com/Onyme13/Homemade_Glider_Tracker/blob/main/3D%20STL%20file/cover%20v0.4.stl

2. Assemblage des câbles du Dupont sur le module GPS

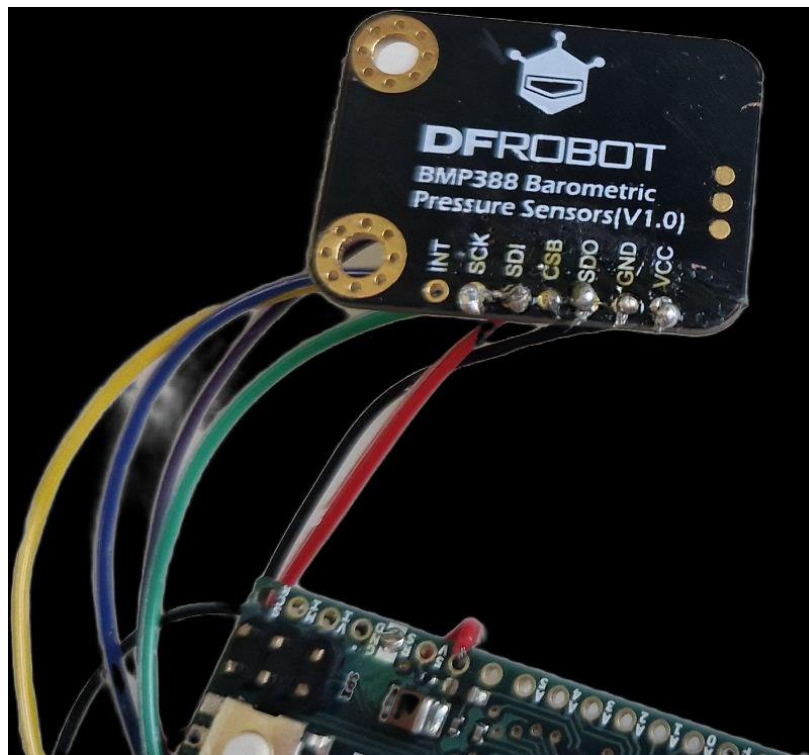


- RXD transmission câble **jaune**
- TXD transmission câble **vert**
- GND prise de terre câble **noir**
- VCC entrée 5 v câble **rouge**

3. Assemblage des câbles du Dupont sur le module barométrique

Souder les câbles Dupont aux entrées :

- VCC
- GND
- SDO
- CSB
- SDI
- SCK



4. Assemblage des câbles du GPS et baromètre au Arduino Micro

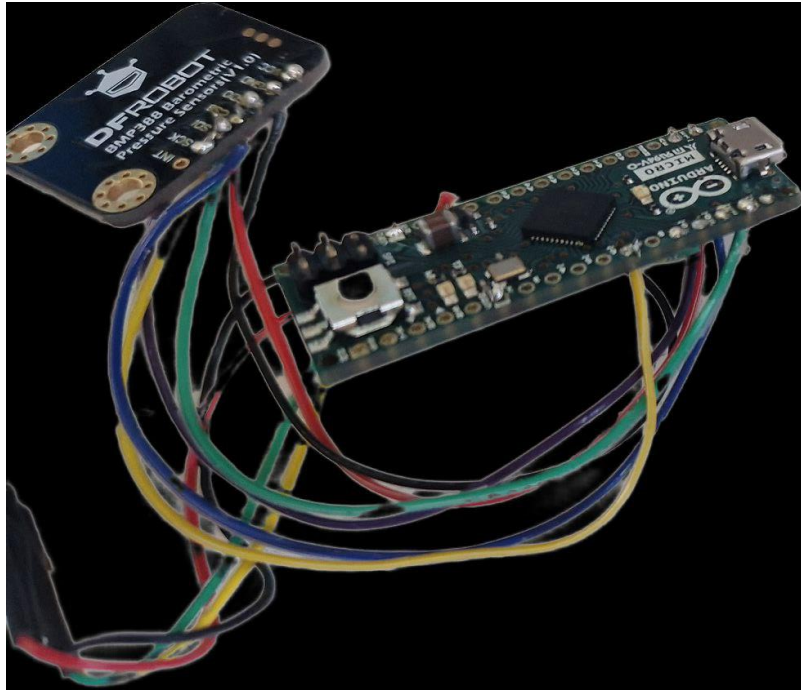
Souder les câbles du GPS et du baromètres au Arduino Micro comme suit :

Baromètre :

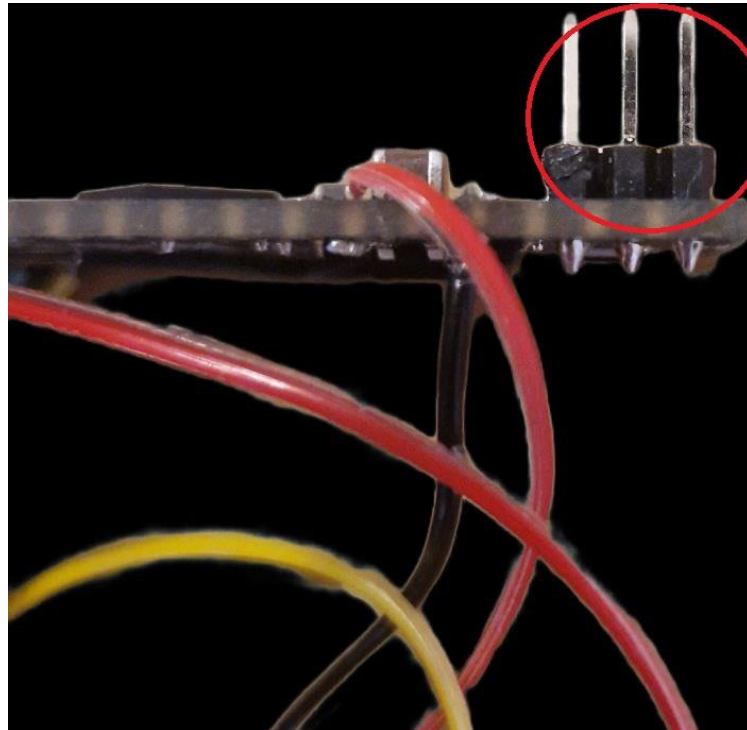
- VCC → +3.3V
- GND → GND
- SDO → 12
- CSB → 10
- SDI → 11
- SCK → 13

Module GPS :

- VCC → +5V
- GND → GND
- RXD → 9
- TXD → 8



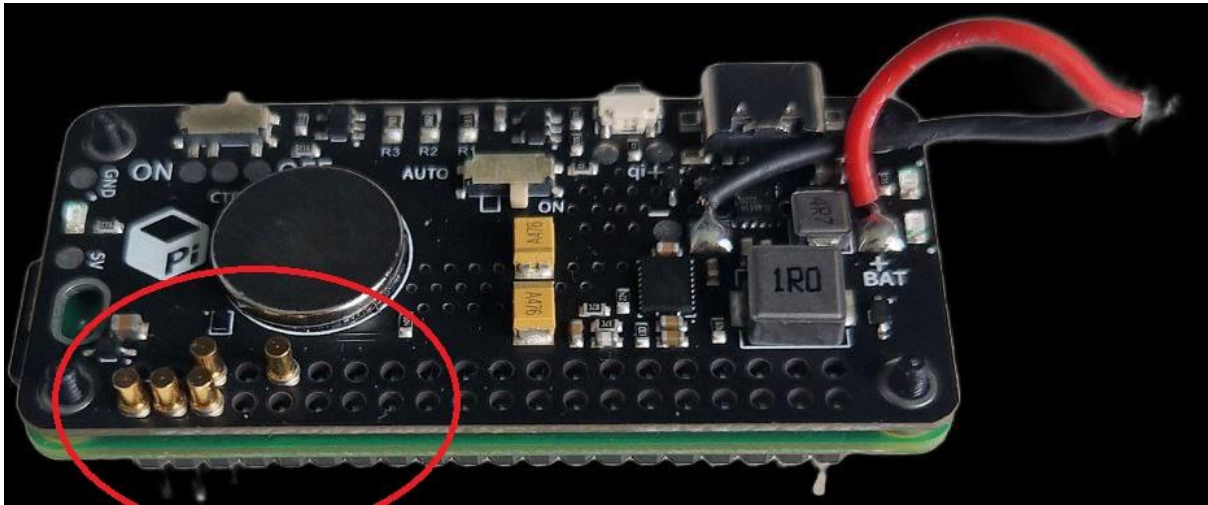
5. Couper les pins du haut du Arduino Micro afin de gagner de la place dans le boîtier



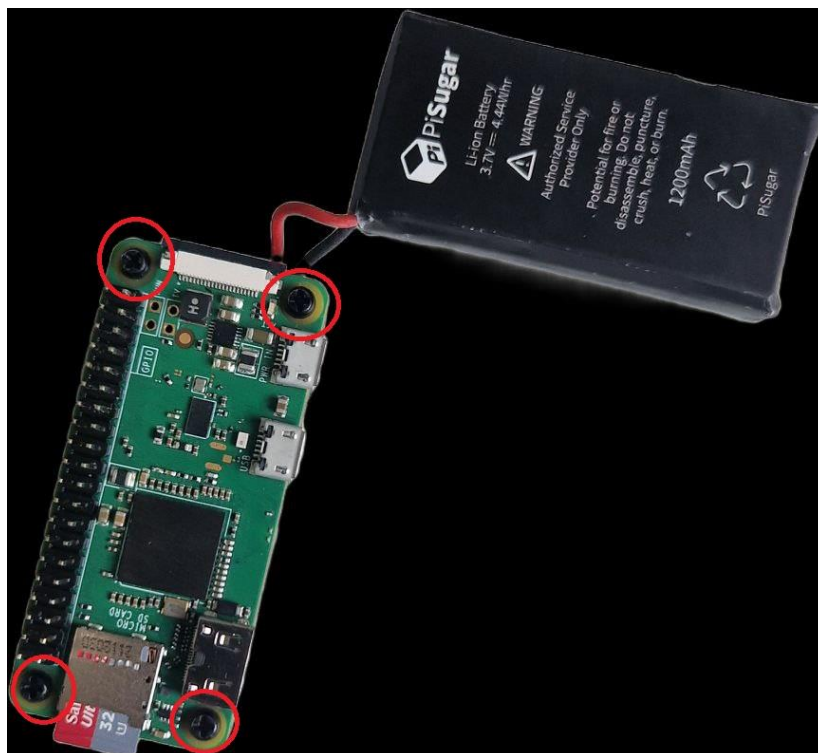
6. Branchez la batterie PiSugar sur le Raspberry Pi à l'aide des quatre vis fournies

Pour plus d'information suivre le manuel de branchage de la batterie PiSugar

6.1 Aligner correctement les pins de la batterie avec les pins du Raspberry Pi



6.2 Visser la batterie PiSugar au Raspberry Pi

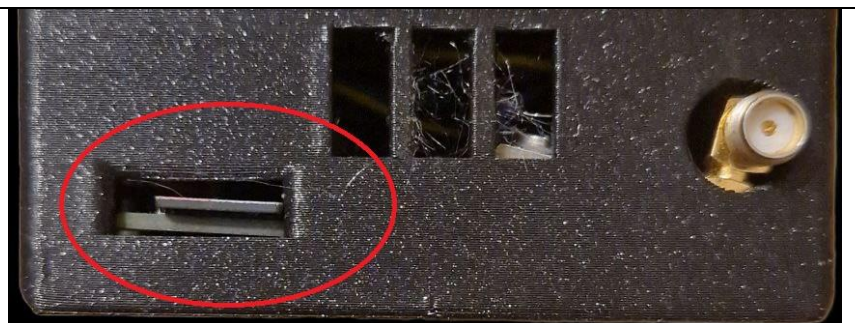


7. Brancher le câble microUSB au Arduino Micro à l'entrée microUSB du Raspberry Pi Zero

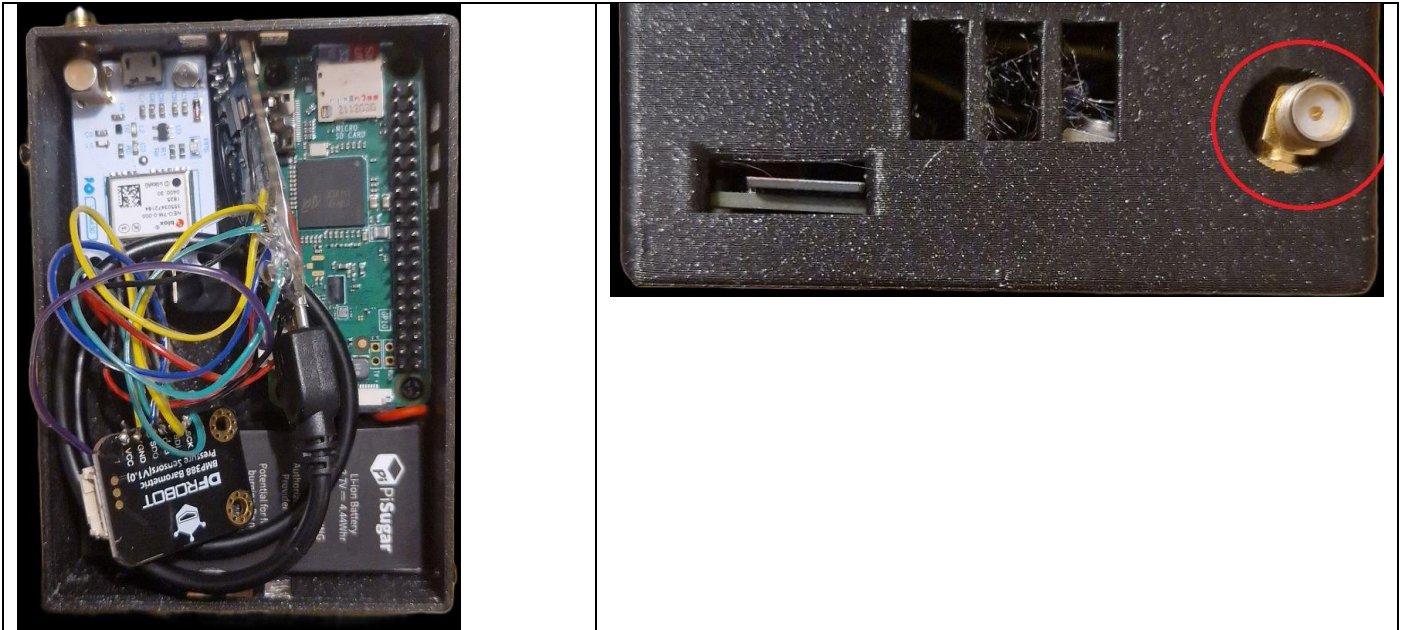


8. Insérez délicatement le tout dans le boîtier imprimé en 3D

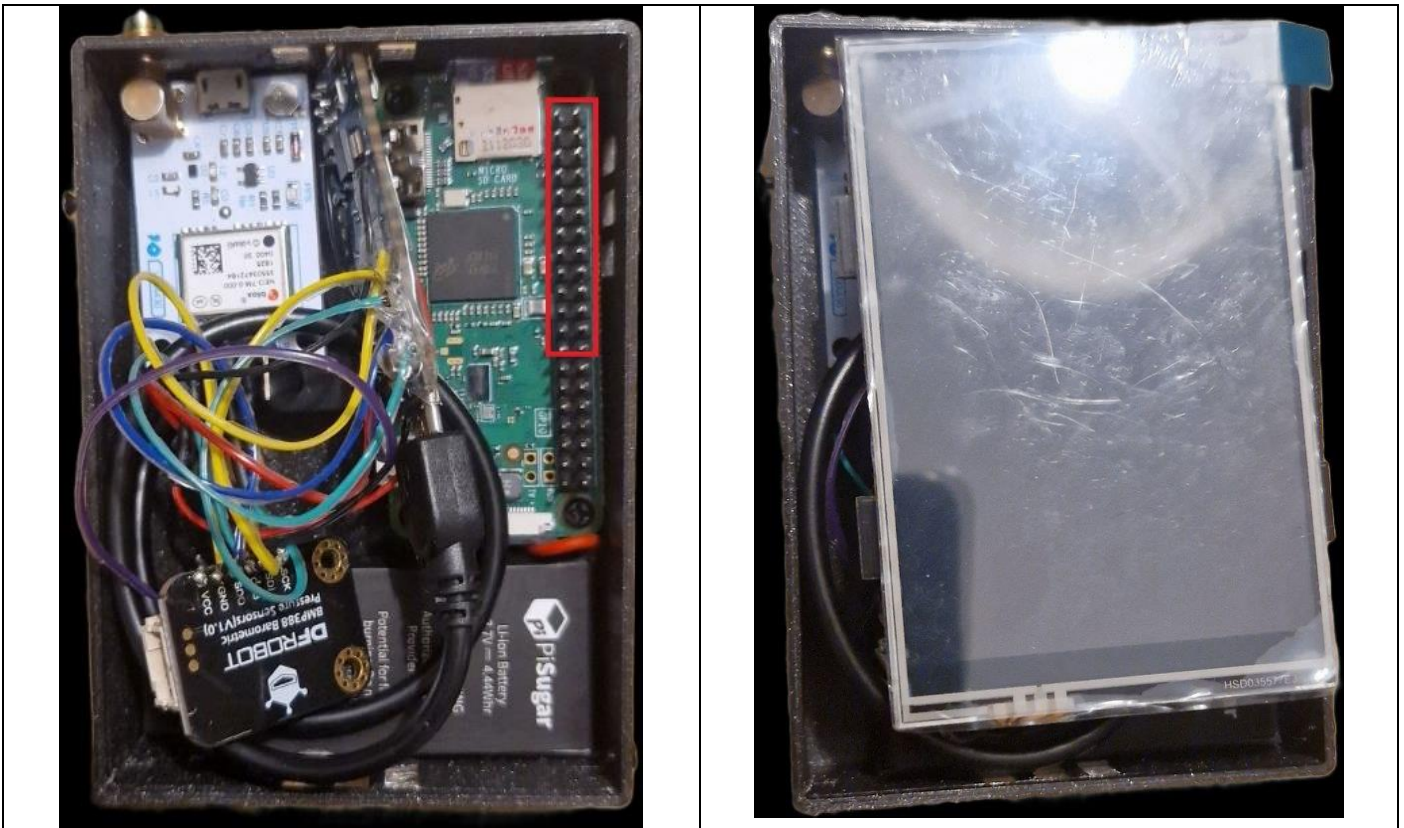
8.1 Insérez en premier le Raspberry Pi Zero avec PiSugar et le câble microUSB comme suit. La carte microSD doit sortir par l'ouverture prévue à cette effet.



8.2 Connectez le câble miscoUSB au Arduino Micro. Et insérez le tout délicatement dans le boîtier, l'entrée male du câble GPS doit ressortir par le trou prévu à cet effet.



8.3 Connecter l'écran aux pins du haut droite du Raspberry Pi (encadrés en rouge)



8.4 Posez le couvercle imprimé en 3D

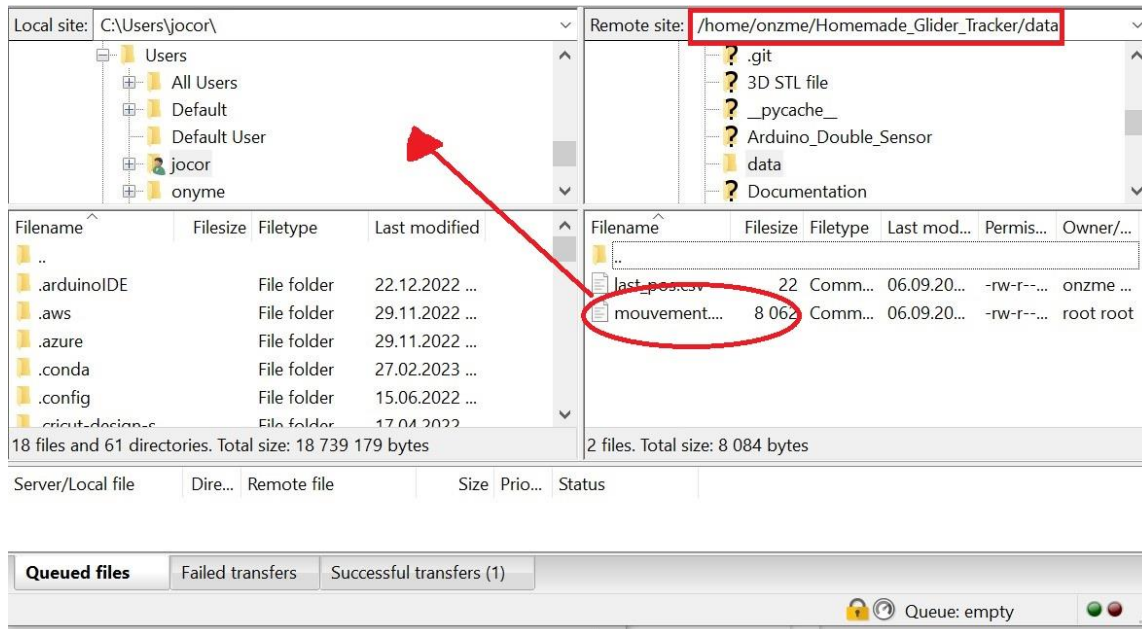
9. Visez les deux vis sur le côté du boîtier

Sur le côté gauche et droite. Il faut des vis de 2mm de diamètre et pas plus de 6mm de longueur



Récupération des données de vols

1. Ouvrir FileZilla sur votre PC
2. Connection l'IP du Raspberry Pi avec ses identifiants
3. **Drag and Drop** le fichier d'enregistrement. Il se trouve dans *Homemade_Glider_Tracker/data/mouvement_[YYYY-MM-DD].csv*



Tests

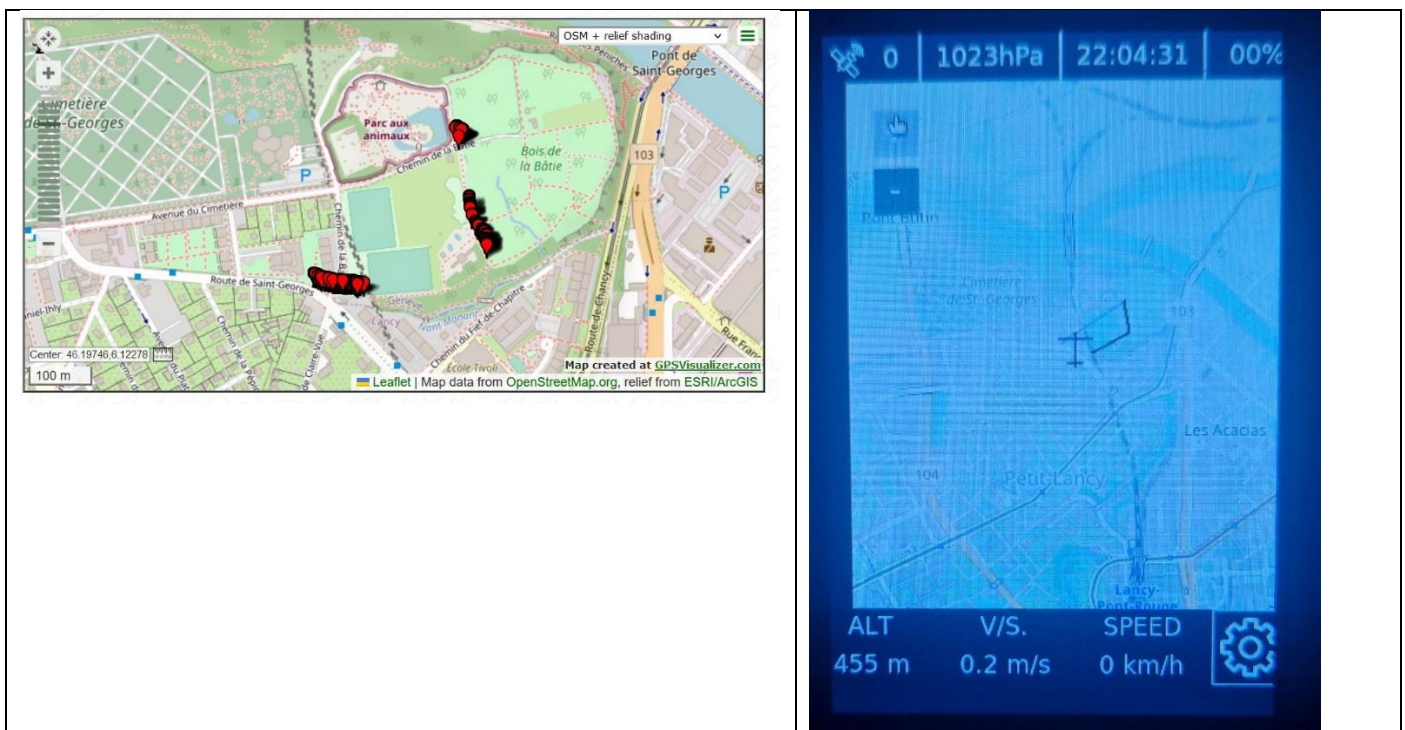
1er Test : À Pied

Le premier test à pied est un succès. Bien que le chemin tracé par le GPS ne soit pas particulièrement long, cela s'explique par la faible vitesse de déplacement de seulement 5 km/h. De manière attendue, la vitesse verticale n'a pas varié durant ce test, ce qui est tout à fait conforme aux attentes. Le GPS fonctionne malgré le fait qu'il a été testé en zone urbaine (Petit-Lancy). Les immeubles d'habitations pourraient bloquer la plupart des satellites de réception, mais le chemin est indiqué par le GPS est fidèle à la réalité du trajet du test.

Le trajet indiqué sur la photo de l'écran est court, mais le logiciel est calibré pour des parcours sur des grandes distances. Ce dernier mets à jour la position du planeur uniquement lorsqu'un certain seuil est dépassé.

Nous pouvons voir que les positions GPS sont réparties par « clusters ». Le fait que le test a été réaliser en zone urbaine pourrait expliquer cela. Les immeubles d'habitations auraient pu bloquer quelques signaux satellites.

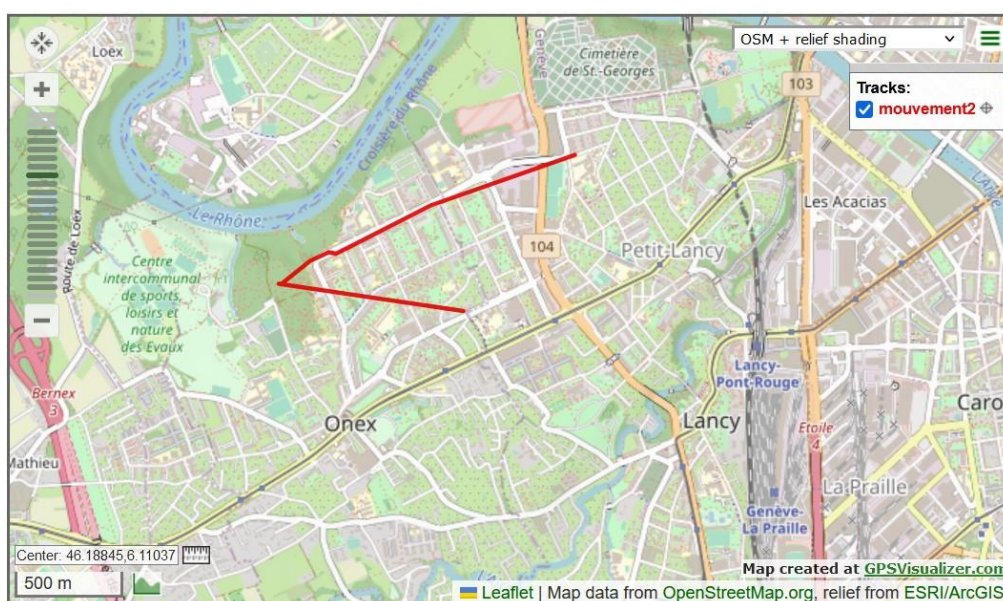
Ces résultats confirment la précision et la fiabilité du GPS, même dans des conditions de déplacement à faible vitesse.



2ème Test : En Vélo

Le test à vélo a produit des résultats similaires, avec un suivi de la position et de l'altitude. Comme pour le test à pied, quelques pertes de signal GPS sporadiques ont été observées. Il est important de souligner que ces perturbations pourraient être attribuées à l'environnement urbain dans lequel le test a été effectué (quartier d'Onex).

Malgré ces perturbations occasionnelles, le tracker a continué à fonctionner de manière fiable dans des conditions exigeantes. Ces résultats prouvent qu'avec une plus grande vitesse, le tracker fonctionne tout de même. Un test en vol est à suivre.



3ème Test : En Avion

La dernière phase de test a placé le tracker de vol dans son élément naturel : l'aviation légère. Intégré à un avion moteur léger, le dispositif a été évalué lors de vols réels en avion.

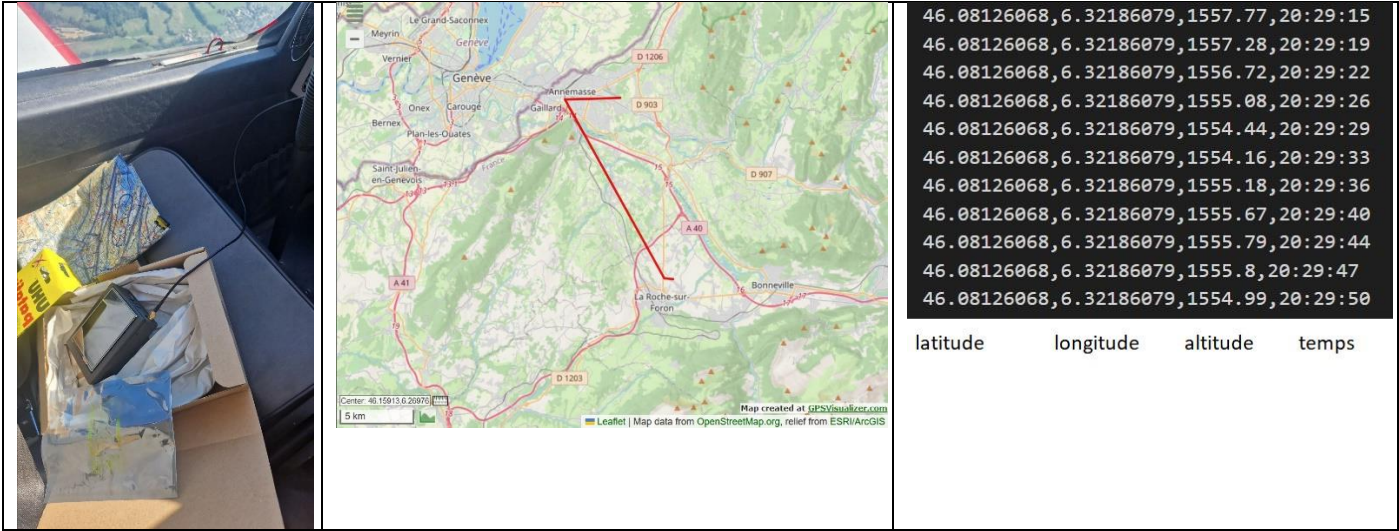
Lors du test en avion moteur, le tracker de vol a été soumis à une épreuve en survolant depuis l'aérodrome d'Annemasse, la région de la Roche-sur-Foron, puis Bonneville, à une altitude de 5000 pieds. Comme nous pouvons le voir sur le report de l'enregistrement du vol sur GPS Visualizer, il y a eu une perte de signal GPS au-dessus de la Roche-sur-Foron qui n'a été retrouvée par la suite. Cependant, malgré cette perte temporaire de données GPS, l'altimètre du tracker de vol a continué à indiquer avec précision l'altitude actuelle. Il est important de noter que l'indicateur de vitesse vertical du tracker indiquait des résultats justes.

Une explication à la perte de signal GPS pourrait être la vitesse de l'aéronef. En effet, la perte de signal est apparue lorsque l'avion a fini sa montée et s'est mis en vol horizontal stabilisé à 5000 pieds. La vitesse de croisière de l'avion est de 210 km/h. Lorsque l'avion se déplace rapidement, le récepteur GPS peut avoir du mal à maintenir une liaison constante avec les satellites, entraînant ainsi des interruptions du signal. La dernière donnée enregistrée indique 1554 mètre ce qui correspond à environ 5000 pieds (voir image ci-dessous). Ceci pourrait confirmer cette explication.

Cependant, la vitesse de montée de l'avion moteur en question est d'environ de 165 km/h, la vitesse de moyenne d'un planeur tourne autour des 100 km/h. Donc en théorie, le signal ne devrait pas se perdre si le tracker est embarqué dans un planeur.

La calibration automatique de l'altimètre, grâce l'altitude donné par le GPS, est correcte. Celui-ci indiquait une pression de 1020hPa ce qui était la pression du jour dans la région (7 septembre 2023). Une caractéristique du tracker de vol est sa capacité de calibration. Il utilise les données de l'altitude du GPS en comparant les mesures répétées (environ 20 fois) pour déterminer une altitude fiable, et ainsi une pression atmosphérique locale. Cette méthodologie de calibration assure une précision optimale dans le suivi de l'altitude, même dans des situations auxquelles la réception GPS peut être intermittente, comme cela a été observé lors du vol au-dessus de la Roche-sur-Foron.

Ces résultats sont très encourageants, car ils montrent que le tracker de vol est équipé pour maintenir des informations cruciales, telles que l'altitude, même en cas de perturbation de la réception GPS.



Conclusion des Tests

Les phases de test diversifiées ont solidement confirmé la fonctionnalité et la performance du tracker de vol. Qu'il s'agisse du déplacement à pied, à vélo ou en avion, le dispositif a fait preuve d'une collecte de données viables. Le choix des composants, notamment le Raspberry Pi Zero W et l'Arduino Micro, a assuré une flexibilité essentielle.

Les futurs développements se concentreront sur l'optimisation logiciel continue du tracker de vol et sur les pertes de signaux GPS occasionnels.