Instrucciones de instalación y uso del servidor TCP y SITL

**Nota: Los comandos están pensados para su ejecución en sistemas basados Debian, como Ubuntu.**

# Paso 1: Prerrequisitos

## Mavproxy

sudo apt-get install python3-dev python3-opencv python3-wxgtk4.0 python3-pip python3-matplotlib python3-lxml python3-pygame

pip3 install PyYAML mavproxy --user

echo "export PATH=$PATH:$HOME/.local/bin" >> ~/.bashrc

Más info: <https://ardupilot.org/mavproxy/docs/getting_started/download_and_installation.html>

## Otras librerías de python

sudo pip install dronekit

sudo pip install dronekit-sitl

sudo pip install mavproxy

sudo pip install pymavlink

sudo pip install pandas

sudo pip install requests

sudo pip install socketserver

**(Si hay errores relacionados con el despegue o el arming, entonces instalar la versión 2.4.8 de mavproxy)**

# Paso 2: Corrección de un bug que impide cambiar de modo en dronekit-sitl

cd ~/.dronekit/sitl/copter-3.3

rm \*

wget https://github.com/dronekit/dronekit-sitl/files/3514761/firmware\_36.zip

unzip firmware\_36.zip

mv copter\_36 apm

sudo chmod -R a+rwx ~/.dronekit/sitl/copter-3.3/

# Paso 3: Carga de Scripts

Descargar todos los archivos de la carpeta TCP, incluido el copter.parm

# Paso 4: Ejecución de los scripts

1. Abrir una terminal en la carpeta descargada y ejecutar dronekit sitl:

dronekit-sitl copter --home 42.283433,-8.789191,216,0

Abrir otra terminal y ejecutar mavproxy:

mavproxy.py --master tcp:127.0.0.1:5760 --out 127.0.0.1:14550 --out 127.0.0.1:14551 --console --map

Los argumentos –map y –console son opcionales.

Esperar a que en la terminal salga Saved 1149 parameters to mav.parm y ejecutar:

param load copter.parm

Abrir otra terminal y ejecutar:

python Ruta.py --inf\_ruta

El argumento inf\_ruta indica al programa que la ruta diseñada se deberá repetir infinitamente. Si no se proporciona el argumento el dron aterrizará cuando complete la trayectoria.

Abrir otra terminal y ejecutar el servidor TCP:

python Serverv0.1.py

El servidor proporcionará los datos en formato JSON creado a partir de un dataframe de pandas.

Por último, si se desea ejecutar el cliente de ejemplo, abrir una nueva terminal y ejecutar:

python Clientev0.1.py