



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE
TELECOMUNICACIÓN

GRADO EN INGENIERÍA EN TELEMÁTICA

TRABAJO DE FIN DE GRADO

Driver y soporte para drones con protocolo MavLink

Autor: Diego Jiménez Bravo

Tutor: Jose Maria Cañas Plaza

Curso Académico 2016/2017

Proyecto Fin de Carrera

TEMA DEL TFG

Autor : Diego Jiménez Bravo **Tutor :** José María Cañas Plaza

La defensa del presente Proyecto Fin de Carrera se realizó el día de
de 2017, siendo calificada por el siguiente tribunal:

Presidente:

Secretario:

Vocal:

y habiendo obtenido la siguiente calificación:

Calificación:

Fuenlabrada, a de de 20XX

Agradecimientos

Resumen

Aquí viene un resumen del proyecto. Ha de constar de tres o cuatro párrafos, donde se presente de manera clara y concisa de qué va el proyecto. Han de quedar respondidas las siguientes preguntas:

- ¿De qué va este proyecto? ¿Cuál es su objetivo principal?
- ¿Cómo se ha realizado? ¿Qué tecnologías están involucradas?
- ¿En qué contexto se ha realizado el proyecto? ¿Es un proyecto dentro de un marco general?

Lo mejor es escribir el resumen al final.

Índice General

1	Introducción	1
1.1	Historia	1
1.2	Tipos de drones	2
1.3	Aplicaciones	4
1.4	Normativa sobre Drones	5
1.4.1	Tipo de dron	6
1.4.2	Espacio aéreo	6
1.4.3	Seguridad	6
1.4.4	Carnet de piloto de Drones en España	6
2	Objetivos	7
2.1	Problemas a abordar	7
2.2	Requisitos	7
2.3	Metodología y plan de trabajo	8
3	Infraestructura	10
3.1	Hardware	10
3.1.1	Dron	10
3.1.2	Pixhawk 2	11
3.2	Software	11
3.2.1	JdeRobot	11
3.2.2	Interfaces	12
3.3	MAVLink	13
3.4	Python	14
4	MavLinkServer	15
4.1	Script de arranque	15
4.2	MavProxy	16
4.3	Modulos MavLink	16

4.4	Integridad con JdeRobot	16
5	Visor Uav Viewer	17
6	Conclusiones	18
7	Bibliografia	19

Indice de imágenes

1.1	Ryan Firebee	1
1.2	Gnat	2
1.3	Clasificacion por alas	3
1.4	Clasificacion por aplicación	5
3.1	3DR Solo Drone	10
3.2	Diagrama comunicacion	11

Capítulo 1

Introducción

1.1 Historia

Un dron no es más que una aeronave no tripulada que se controla remotamente y que, por lo general, se puede decir que siempre ha sido el sueño de todo estratega militar, ya que permite alcanzar al enemigo a distancia así como evitar pérdidas humanas.

Este sueño comenzó en el 1916, cuando el profesor Archibald Low, un militar científico, diseñó un torpedo aéreo que se manejaba con un sencillo control usando señales de radio. La aeronave fue un fracaso, pero sentó las bases para futuros diseños.

En Vietnam también se usaron otros modelos de dron como el Ryan Firebee que introdujo una nueva idea, la cámara de fotos para esppiar al bando contrario.



Figura 1.1: Ryan Firebee

Pero si hay que hablar de una salto en el desarrollo de los drones entonces es necesario mencionar al Gnat que fue desarrollado por General Atomics, un contratista de defensa de San Diego, California, EE. UU. que introdujo las cámaras de video y con esto empezó una nueva era en el mundo de los dron.



Figura 1.2: Gnat

Para comenzar a hablar del uso de los dron por el público sin duda hay que mencionar a Nikola Tesla quien patentó por primera vez un vehículo no tripulado controlado remotamente al que llamó teleautomation y que hoy es uno de los principios que rige el diseño de un dron.

Sin duda ha sido un gran salto, por un lado muchos encuentran divertido el uso de los drone para fines personales y comerciales como la entrega de paquetes y otros proyectos de Amazon y Google mientras que otros critican fuertemente su uso en la guerra para bombardear a terroristas.

1.2 Tipos de drones

Como vehículo aéreo puede tener diferentes formas, bien tipo avión, tipo helicóptero o incluso formas muy diferentes. Pero los drones no son algo nuevo, el ejemplo más antiguo fue desarrollado después de la primera guerra mundial, y se emplearon durante la segunda guerra mundial para entrenar a los operarios de los cañones antiaéreos. Sin embargo, no es hasta poco más que a finales del siglo XX cuando operan los drones mediante radio control con todas las características de autonomía.

Algunos tienen sistema GPS que les permite volver al punto donde inició de su vuelo. En el futuro se espera que los drones vuelen solos, tomando sus propias decisiones, evitando chocar contra las personas y poder evitar los objetos.

La mayoría de los drones se manejan con radio control, pero pueden ser también manejados y programadas mediante una tablet o un smartphone.

Se utilizan para múltiples tareas, desde tareas de vigilancia, fotografía, retransmisiones televisivas, agricultura, ocio y muchas más tareas, ya que cada poco se descubre una nueva forma de utilizar los drones. La clasificación es muy amplia, pero la primera clasificación podría ser en función del tipo de alas.

- Drones de Alas Fijas: Tienen alas fijas y son similares a un avión.
- Drones MultiRotor: Suelen ser cuadricópteros (4 rotores con hélices) aunque los hay que tienen 6 (hexacópteros) o incluso 8 hélices. Dos hélices giran en el sentido de las agujas del reloj y las otras dos en el otro sentido, creando así la fuerza de empuje necesario para llevar al dron hacia arriba. Se pueden mantener en el mismo sitio sin variar la posición, gracias a sus giroscopios y estabilizadores, lo que es perfecto para sacar fotos y grabar videos.



Figura 1.3: Clasificación por alas

Según el método de control tenemos:

- Autónomo: El dron no necesita de un piloto humano que lo controle desde tierra. Se guía por sus propios sistemas y sensores integrados.
- Monitorizado: En este caso si se necesita la figura de un técnico humano. La labor de esta persona es proporcionar información y controlar el feedback del dron. El dron dirige su propio plan de vuelo y el técnico, a pesar de no poder controlar los mandos directamente, sí puede decidir qué acción llevará a cabo.
- Supervisado: Un operador pilota el dron, aunque este puede realizar algunas tareas autónomamente.
- Preprogramado: El dron sigue un plan de vuelo diseñado previamente y no tiene medios de cambiarlo para adaptarse a posibles cambios.

- Controlado remotamente(R/C): El drone es pilotado directamente por un técnico mediante una consola.

En función de su uso pueden ser:

- Drones Militares: son llamados UCAV que procede del inglés Unmanned Combat Air Vehicle, traducido al español sería vehículos no tripulados de combate aéreo. Suelen ir armados y con capacidad de bombardeos.
- Monitorizado: En este caso si se necesita la figura de un técnico humano. La labor de esta persona es proporcionar información y controlar el feedback del drone. El drone dirige su propio plan de vuelo y el técnico, a pesar de no poder controlar los mandos directamente, sí puede decidir qué acción llevará a cabo.
- Drones Civiles: son aquellos drones que no tienen uso militar. A su vez pueden ser de:
 - De uso comercial: como cartografías, fotografías, vídeos, etc.
 - Para Aficionados: Se utilizan como un juguete y suelen tener precios bastante económicos.
 - Para Uso del Gobierno: Se utilizan para bomberos, fuerzas de rescate, etc. con el fin de ayudar a las tareas de reconocimiento, rescate, fronterizas e incluso fiscales.

1.3 Aplicaciones

Se pueden aplicar en ambientes de alta toxicidad química y radiológicos en desastres tipo Chernóbil, en los que sea necesario tomar muestras con alto peligro de vidas humanas y realizar tareas de control de ambiente. Las aeronaves cumplen con las normas regulatorias establecidas en el Tratado de Cielos Abiertos de 1992 que permiten los vuelos de VANT sobre todo el espacio aéreo de sus signatarios. Además, pueden cooperar en misiones de control del narcotráfico y contra el terrorismo. También podrían grabar vídeos de alta calidad para ser empleados como medios de prueba en un juicio internacional.

Los UAV tienen múltiples aplicaciones y posibilidades en el mercado civil y profesional:

- Internet: distribución de señal gratuita de internet.
- Cartografía: realización de ortofotomapas y de modelos de elevaciones del terreno de alta resolución.
- Monitorización de instalaciones.
- Transporte y entrega de mercancías.

- Agricultura: gestión de cultivos.
- Cine y deportes extremos.
- Servicios forestales: seguimiento de las áreas boscosas, control de incendios.
- Búsqueda, rescate y salvamento de personas.
- Medio ambiente: estado de la atmósfera.
- Seguimiento de la planificación urbanística.
- Gestión del patrimonio.
- Seguridad y control fronterizo.
- Purificar el aire mediante un proceso de filtrado mediante capas de poliéster y carbón activado en ambientes de la industria y el hogar.



Figura 1.4: Clasificación por aplicación

1.4 Normativa sobre Drones

En el documento oficial del Estado quedan reflejadas las condiciones en las que se puede realizar trabajos técnicos y científicos, tales como grabación aérea, reportajes aéreos, fotografía aérea, estudios de fotogrametría, vigilancia y monitoreo y revisión de infraestructuras entre otros. Gran parte de este nuevo decreto de ley temporal, se basa en 4 puntos clave que toda empresa que desee operar con drones deberá contemplar y seguir:

- Tipo de Drone
- Espacio aéreo

- Seguridad
- Carnet de piloto de Drone

1.4.1 Tipo de dron

Se establecen dos categorías inciales: Drones con peso inferior a 2Kg. y drones con peso entre los 2Kg. y 25Kg. Para ambos es imprescindible disponer de un carnet de piloto de drones para poder operar en España. En caso de los drones de peso inferior a 2kg, no será necesario que estén inscritos en el registro de aeronaves ni disponer de un certificado de aeronavegabilidad. Para ambos tipos de drone, será necesario incluir obligatoriamente una placa identificativa con el nombre del fabricante del aparato así como los datos fiscales de la empresa que lleve a cabo dichas operaciones.

1.4.2 Espacio aéreo

El espacio aéreo pertenece a AESA, y como tal, para poder realizar cualquier tipo de actividad comercial o civil con un drone, se deberá obtener un permiso oficial, como mínimo 5 días antes de llevar a cabo cualquier operación en el aire. Esta nueva legislación sigue manteniendo la prohibición de sobrevolar núcleos urbanos o espacios con una alta masificación de gente sin el consentimiento especial por parte de la Agencia Española de Seguridad Aérea.

1.4.3 Seguridad

El pilar fundamental en el que se ha basado el Ministerio para la realización de la normativa de uso de drones civiles en España es la seguridad. Por ello cada empresa deberá disponer de un manual de operaciones cumplimentado siguiendo el estándar proporcionado por el Ministerio, así como un estudio de seguridad de cada una de las operaciones a realizar. Es decir, si alguien piensa en hacer volar un drone al margen de la ley, ya sea con un peso inferior a 2kg, o entre 2kg y 25kg, se expone a sanciones que van entre 3.000 a 60.000 euros.

1.4.4 Carnet de piloto de Drones en España

Para que las empresas puedan operar legalmente, como lo hace Dronair, los pilotos designados deberán disponer de un carnet oficial para el manejo de drones.. Si estos pilotos ya disponen de un título de piloto de avión, ultraligero u otro específico, no será necesario obtener dicha titulación. En caso contrario deberán cursar una serie de exámenes y pruebas oficiales para obtener el carnet oficial de piloto de drones. A día de hoy, no existen academias oficiales bajo la tutela del Gobierno que realicen estos cursos, por eso y mientras se empiezan a impartir estos cursos, será obligatorio demostrar que se dispone de los conocimientos teóricos y algún tipo de carnet oficial o documento que acredite a los pilotos en el manejo de drones para poder llevar a cabo cualquier operación.

Capítulo 2

Objetivos

2.1 Problemas a abordar

Los objetivos de este trabajo fin de grado es crear un driver en la plataforma software JdeRobot para drones que utilicen como interfaz de comunicación MAVLink. Para abordar el problema lo hemos dividido en:

1. Preparación del hardware necesario.
2. Desarrollar driver para pilotar drones en una interfaz de nivel medio. Se comunicara con el dron usando el protocolo MavLink y se usará el interfaz de nivel medio que facilita JdeRobot llamado CMDVel.
3. Desarrollar una herramienta que permita pilotar los drones con interfaz de nivel medio en python de manera más intuitiva. Esta herramienta es una evolución del interfaz ya existente del UavViewer que facilita JdeRobot.
4. Experimentos en dron real. Conectaremos nuestro 3DR Solo tanto al driver MavLinkServer, que nos facilitara la comunicacion y a la herramienta UavViewer que nos permitira pilotar.

2.2 Requisitos

Para abordar con éxito los puntos expuestos anteriormente debemos cubrir los siguientes requisitos:

1. Preparación del hardware necesario:
 - a Compra del hardware necesario:
 - 3DR Solo dron.
 - Webcam.
 - Intel Computer Stick.

- Batería externa.
 - b Instalación de JdeRobot en Intel Computer Stick.
 - c Pruebas de vuelo con todo el equipamiento a bordo del 3DR Solo y pilotaje externo.
2. MavLinkServer
 - a Comunicación con el dron.
 - b Administración de los modos de vuelo.
 - c Fases de despegue y aterrizaje automáticas.
 - d Envio de comandos de velocidad al dron.
 3. UavViewerPy
 - a Comunicación con MavLinkServer.
 - b Ordenes de despegue y aterrizaje.
 - c Envio de comandos de velocidad a MavLinkServer mediante 2 joysticks.
 4. Experimentos en dron real.
 - a Pruebas de interconexion entre todos los componentes.
 - b Ejecución de plan de pruebas:
 - Prueba de despegue y aterrizaje.
 - Prueba de vuelo controlado.

Cómo requisitos no funcionales debemos:

1. Ser multiplataforma.
2. Utilizar únicamente librerías de software libre.
3. Ser 100 % compatibles con los actuales interfaces JdeRobot.

2.3 Metodología y plan de trabajo

Durante el ciclo de vida del proyecto se han llevado a cabo reuniones semanales de seguimiento con el tutor. En ellas se evaluaban las tareas realizadas y se marcaba qué dirección tomar para la siguiente iteración o incremento. Si los puntos marcados en la anterior reunión no se habían alcanzado se ampliaba el plazo o se discutían otras vías para avanzar. En caso contrario se proponían nuevos subobjetivos. Para apoyarnos en nuestro desarrollo hemos utilizado principalmente 4 herramientas:

- GitHub como forja y control de versiones. En el repositorio <https://github.com/RoboticsURJC-students/2016-tfg-Diego-Jimenez> se almacenan todos los desarrollos que son objetivo de éste TFG así como ésta memoria. También se encuentran subproductos de desarrollo que han ido surgiendo como apoyo o pruebas a los desarrollos principales.
- Contamos también con un mediawiki en JdeRobot dónde hemos actualizado periódicamente nuestros avances acompañados con explicaciones, vídeos e imágenes. <http://jderobot.org/Jimenez-tfg>
- Todos los vídeos del mediawiki han sido compartidos en Youtube.

Capítulo 3

Infraestructura

3.1 Hardware

3.1.1 Dron

Para este desarrollo hemos elegido el dron 3DR Solo distribuido por la empresa norteamericana <https://3dr.com/>. Este dron se encuentra en una gama alta debido a sus capacidades <https://3dr.com/solo-drone/specs/>, tales como batería, distancia de comunicacion y potencia, cualidades que lo hacen un dron muy versatil. Durante el desarrollo se ha usado la versión oficial del software de 3DR 2.4.2. Se eligio esa debido a que era una versión estable y en versiones posteriores del software aun no se han corregido el principal fallo de conectividad debido a usar el mando como puerta de enlace. A bordo de este dron se encuentra una placa Pixhawk 2, con la que nos debemos comunicar a traves del mando con el que se pilota el dron. Este inconveniente 3DR trata de solventarlo para evitar tener que usar el mando como enlace obligatorio.



Figura 3.1: 3DR Solo Drone

3.1.2 Pixhawk 2

Esta placa se trata de un desarrollo específico creado entre la "Pixhawk open hardware community" en colaboración con 3D Robotics y que vio como primer destinatario el 3DR Solo. Ofrece un interfaz que se apoya en comandos llamado MAVLink. A través de estos comandos se le puede también enviar ordenes al piloto automático quien las ejecutará más adelante trataremos el protocolo MAVLink en profundidad. El único modo de conectarnos con el 3DR Solo será a través del mando, debido a que únicamente el mando es capaz de levantar la dirección IP a la que poder conectarnos. En posteriores evoluciones de la versión que controlan tanto el dron como el mando, desde los foros oficiales de 3DR, comentan que ya se aborda la solución de que sea el propio dron quien levante la dirección IP a la cual poder conectarnos y no tener que depender del enlace del mando.

La comunicación entre el driver y el dron quedaría de la siguiente forma:



Figura 3.2: Diagrama comunicación

3.2 Software

3.2.1 JdeRobot

JdeRobot es un framework desarrollado por el laboratorio de robótica de la Universidad Rey Juan Carlos, para el desarrollo de aplicaciones de robótica. Su última realease la 5.5 se liberó el 15 de Marzo de 2017 pudiendo ver los detalles de ésta en el github oficial¹. JdeRobot se compone de interfaces, drivers, utilidades y aplicaciones para el desarrollo de cualquier proyecto de robótica, se apoya en estos interfaces, algunos de ellos los veremos en profundidad a continuación, para

¹<https://github.com/JdeRobot/JdeRobot/wiki/JdeRobot-5.5.0>

interconectar entre sí todos los aplicativos del mismo y en Zeroc ICE para la comunicación entre ellos. Algunos de los driver mas importantes que contiene serían:

1. MAVLinkServer. Desarrollado para intercomunicar JdeRobot con placas que utilicen el protocolo de comunicación MAVLink. Desarrollo que se expone en este TFG.
2. Camerasher. Se trata de un driver para enviar imágenes y video a través del interfaz camera
3. Gazeboserver. Driver desarrollado para conectar la herramienta de simulación Gazebo con JdeRobot y así poder simular los desarrollos.
4. Ardrone_server. Driver que conecta el Parrot Ar-Drone a JdeRobot. Este driver escrito en c++ transforma el set de comandos AT del drone en interfaces y viceversa, implementa los interfaces camera, cmdvel, navdata, extra y pose3D y permite acceder a la actitud del drone así como a sus 2 cámaras. Sirve también datos como el nivel de la batería y permite grabar vídeo o tomar fotos.

Algunas de las aplicaciones desarrolladas más importantes serían:

1. Cameraview. Se trata de una aplicación desarrollada en c++ capaz de recibir vídeo a través del interfaz camera.
2. UAV viewer. Aplicación desarrollada como ground control de robots aéreos. Esta aplicación permite teleoperar cualquier tipo de robot aéreo y ofrece de forma visualmente atractiva datos como la actitud, velocidades lineales y angulares, ofrece también la posibilidad de visualizar videos servidos por el interfaz camera. Aplicación que se explicará con más detalle en este TFG más adelante.

3.2.2 Interfaces

JdeRobot expone más de 30 interfaces pero en este capítulo explicaremos los que durante nuestro desarrollo hemos implementado:

- Pose3D. Utilizado para recoger los datos de actitud y la posición de la aeronave.

```
Pose3DData
{
    float x; /* x coord */
    float y; /* y coord */
    float z; /* z coord */
    float h; /* */
    float q0; /* qw */
    float q1; /* qx */
    float q2; /* qy */
    float q3; /* qz */
};
```

- CMDVel. Utilizado para enviar comandos de velocidad.

```
class CMDVelData
{
    float linearX;
    float linearY;
    float linearZ;
    float angularX;
    float angularY;
    float angularZ;
};
```

- Extra. Utilizado principalmente para las órdenes de despegue y aterrizaje.

```
void land() - land drone.
void takeoff() - takeoff drone.
void reset()
void recordOnUsb(bool record)
void ledAnimation(int type, float duration, float req)
void flightAnimation(int type, float duration)
void flatTrim()
void toggleCam() - switch camera.
```

3.3 MAVLink

MAVLink siglas de Micro Air Vehicle Link es un protocolo de comunicación desarrollado para comunicar las placas estabilizadoras con piloto automático a los GCS o Ground control station, las aplicaciones desde las que se podía enviar misiones y seguir el cumplimiento de las mismas desde tierra. MAVLink se publicó en 2009 por Lorenz Meier, publicado bajo licencia LGPL aspira a convertirse en el protocolo standard en robótica aérea y se ha probado su funcionamiento en PX4, PIXHAWK, APM² y Parrot AR.Drone.

Un ejemplo de comando MAVLink sería:

```
type GpsStatus struct {
    SatellitesVisible  uint8      Número de satélites visibles
    SatellitePrn       [20]uint8   Id Global de cada satélite
    SatelliteUsed      [20]uint8   Lista con el uso de cada satélite
    SatelliteElevation [20]uint8   Elevaci\'on, nos da el ángulo sobre el horizonte.
    SatelliteAzimuth   [20]uint8   Direcci\'on del satélite, 0: 0 grados, 255: 360 grados.
    SatelliteSnr        [20]uint8   Señal/ruido de cada uno de los satélites
}
```

Este mensaje trae la información del enlace actual con el GPS y se envía periódicamente en ciclos que decidimos en parámetros de conexión con el dispositivo. Otro parámetro, esta vez vinculado a la actuación sería:

```
type MissionItem struct {
    Param1      float32     parámetro variable en funci\'on del comando.
```

²Ardupilot Mega

```

Param2      float32   parámetro variable en función del comando.
Param3      float32   parámetro variable en función del comando.
Param4      float32   parámetro variable en función del comando.
X          float32   latitud
Y          float32   longitud
Z          float32   altitud
Seq         uint16    Número del ítem en la misión
Command     uint16    Tipo de comando de navegación.
TargetSystem uint8    ID del sistema
TargetComponent uint8
Frame        uint8    Sistema de coordenadas que se utiliza.
Current      uint8    Misión actual no:0, si:1
Autocontinue uint8    Autocontinuar al siguiente objeto de misión.

}

```

3.4 Python

Python es un lenguaje de programación interpretado y multiplataforma que nació en los años 80 en los países bajos con idea de hacer más legible el código. El lenguaje de programación que inicialmente se utilizaba principalmente para scripting, ha sabido crecer con los años y con la publicación de Python3 en 2009 ha recibido el impulso que necesitaba para ser hoy en día el 5º lenguaje más utilizado por encima de PHP, .NET y Javascript que baja hasta el 8º puesto según TIOBE en un estudio de Abril de 2017.

El porqué de utilizar Python, muy sencillo mantiene el carácter multiplataforma de JdeRobot, su código es simple y legible y trabaja muy bien con dependencias muy utilizadas en robótica como OpenCV.

Capítulo 4

MavLinkServer

El driver MavLinkServer es quien va a mediar entre las aplicaciones de JdeRobot y el dron con sus sensores y actuadores físicos. De ésta manera las aplicaciones pueden correr en máquinas distintas, no obligatoriamente abordo, y pueden estar implementadas en distintos lenguajes de programación. Estas ventajas vienen de utilizar la división habitual en JdeRobot entre componentes drivers y componentes aplicación. La comunicación será vía WIFI desde el Intel Computer Stick y el mando, ya que debido a un requerimiento que se esta realizando por parte del equipo de 3DR, es obligatorio realizar la conexión con el 3DR Solo dron ¹.

A continuación vamos a dividir en distintas fases el contenido de este driver y su ejecución:

- Script de arranque arranque del servidor.
- Programa principal, en nuestra aplicación recibe el nombre de mavproxy.
- Modulos MavLink.
- Integridad con JdeRobot.

4.1 Script de arranque

El script de arranque se encargara de ejecutar todos los comandos previos y el servidor. El script se encuentra en MAVProxy/MAVProxyWinLAN.sh, deberemos averiguar la IP que levanta el dron, en nuestro caso, con el 3DR Solo, dicha IP la levanta el mando como hemos comentado en la introducción de este capítulo. Deberemos modificar el script con la IP del dron a la que nos hayamos conectado y ejecutarlo sin parametros adicionales. El fichero README del repositorio contiene una descripción mas detallada de un ejemplo de ejecución. En dicho script lanzaremos la carga de los modulos que nos proporciona MavLink oficialmente en su repositorio oficial. ².

¹<https://discuss.dronekit.io/t/how-to-connect-to-3dr-solo-drone-using-only-pc-without-using-the-rc/>

²<https://github.com/ArduPilot/MAVProxy>

Se necesita tener preinstalado tanto pyserial como una versión de pyvmavlink superior a la 1.1.50. Se realiza una descarga de todos los modulos, construye un directorio llamado MavProxy.egg en /home/USER/.local/python3.5/site-packages con el fin de tener almacenados todos los paquetes necesarios y con permisos suficientes.

AQUI DEBERIA METER UN TREE DE LA ESTRUCTURA DE COMO QUEDAN LOS DIRECTORIOS POR DEBAJO DEL .EGG

4.2 MavProxy

4.3 Modulos MavLink

4.4 Integridad con JdeRobot

Capítulo 5

Visor Uav Viewer

Capítulo 6

Conclusiones

Capítulo 7

Bibliografia