



UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**Trabajo Fin de Máster**  
**CURSO 2016/17**

---

*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**ALUMNA/O**

D. Román Sobrino Robelo

**TUTORAS/ES**

D. Félix Orjales Saavedra  
D. Francisco Javier Bellas Bouza

**FECHA**

SEPTIEMBRE 2017

## RESUMEN DEL PROYECTO

En la línea de investigación de UAVs del GII (Grupo Integrado de Ingeniería) de la UDC se desarrolló un proyecto con el objetivo de realizar un modelo virtual de los UAVs, para emplearlo en simulación y con el que poder realizar pruebas de vuelo con facilidad.

El modelo virtual resultante tuvo que ser reajustado en la fase final de programación, debido a que exhibía un comportamiento en velocidad muy diferente con los datos de la contraparte real. Se asumió que fue debido a la posible imprecisión de la captación de datos de vuelo.

Este proyecto se centra en el estudio y revisión del UAV, modelo Hércules XL 76", para tratar de localizar el posible origen de la discrepancia de datos. La revisión se centra en la funcionalidad de los sistemas de sensores de los UAVs y en su precisión. Posteriormente se comprobará si se puede obtener un modelo virtual con un comportamiento en vuelo mejorado y más ajustado con la realidad.



UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**Documento**

**MEMORIA**

## INDICE DE LA MEMORIA

<b>1. INTRODUCCIÓN</b> .....	<b>8</b>
1.1. UAV .....	8
1.1.1. <i>Clasificación de los UAV</i> .....	8
1.2. SIMULADORES .....	9
1.2.1. <i>Tipos de simuladores</i> .....	10
1.3. BREVE DESCRIPCIÓN DE NUESTRO UAV .....	11
<b>2. PRECEDENTES Y OBJETIVOS</b> .....	<b>13</b>
<b>3. PROBLEMAS OBSERVADOS Y POSIBLES CAUSAS</b> .....	<b>13</b>
<b>4. REVISIÓN: APARTADO FÍSICO</b> .....	<b>14</b>
4.1. SENSORES .....	14
4.1.1. <i>Pitots</i> .....	15
4.1.2. <i>IMU</i> .....	32
4.1.3. <i>GPS y altímetro</i> .....	37
<b>5. REVISIÓN: APARTADO VIRTUAL</b> .....	<b>40</b>
5.1. SIMULADOR DE VUELO .....	40
5.2. FLIGHT DYNAMICS MODELS (FDM) .....	40
5.3. MODELADO DEL UAV .....	40
5.3.1. <i>Hélices</i> .....	41
5.3.2. <i>Motores</i> .....	44
5.3.3. <i>Aerodinámica</i> .....	45
5.3.4. <i>Tren de aterrizaje e inercias</i> .....	62
<b>6. PRUEBAS Y MEJORAS</b> .....	<b>66</b>
6.1. PRUEBAS .....	66
6.1.1. <i>Resultados de simulación</i> .....	66
6.1.2. <i>Resultados de vuelo</i> .....	69
6.2. ANÁLISIS DE LOS RESULTADOS DEL MODELO .....	71
<b>7. CONCLUSIONES</b> .....	<b>73</b>
<b>BIBLIOGRAFÍA</b> .....	<b>74</b>

## Índice de tablas

Tabla 1.1: Clasificación de UAVs .....	9
Tabla 5.1: Cuerda en cada sección de la hélice.....	41
Tabla 5.2: Ángulo de ataque en cada sección de la hélice.....	42
Tabla 5.3: Perfil de cada sección.....	43
Tabla 5.4: Principales características geométricas del UAV.....	46
Tabla 5.5: Condiciones de vuelo constantes.....	49
Tabla 5.6: $CD_0$ en función de $\alpha$ .....	50
Tabla 5.7: $CD_0$ ajustado .....	52
Tabla 5.8: $CD$ inducido en función de $\alpha$ .....	53
Tabla 5.9: $CD$ en función de $\beta$ .....	53
Tabla 5.10: $CD$ debido a profundidad .....	54
Tabla 5.11: Coeficiente de fuerza lateral en función de $\beta$ .....	54
Tabla 5.12: $CL$ en función de $\alpha$ .....	55
Tabla 5.13: $CL$ en función de $\alpha$ modificado.....	56
Tabla 5.14: $CL$ en función del ángulo de profundidad.....	57
Tabla 5.15: $Cl$ en función de $\beta$ .....	57
Tabla 5.16: $Cl$ en función de la velocidad de alabeo.....	58
Tabla 5.17: $Cl$ en función de la velocidad de guiñada.....	58
Tabla 5.18: $Cl$ en función del giro del alerón.....	59
Tabla 5.19: $Cl$ en función del giro de la profundidad.....	59
Tabla 5.20: $C_m$ en función de alfa .....	59
Tabla 5.21: $C_m$ en función del giro de la profundidad.....	60
Tabla 5.22: $Cl$ en función de la velocidad de cabeceo .....	60
Tabla 5.23: $C_n$ en función de $\beta$ .....	61
Tabla 5.24: $Cl$ debido a la velocidad de guiñada.....	61
Tabla 5.25: $C_n$ en función del giro de la deriva.....	61
Tabla 5.26: Inercias .....	63
Tabla 6.1: Comparación de datos.....	71
Tabla 6.2: Coeficientes de arrastre ( $CD_0$ ) finales .....	72
Tabla 6.3: Coeficientes de sustentación ( $CL\alpha$ ) finales .....	72

## Índice de ilustraciones

Ilustración 1.1: Unmanned Aerial Vehicle .....	8
Ilustración 1.2: Superficies de control de un avión.....	11
Ilustración 1.3: UAV modelo Hércules XL76“, vista trasera.....	12
Ilustración 4.1: UAV modelo Hércules XL76“, vista lateral .....	14
Ilustración 4.2: Electrónica del UAV .....	15
Ilustración 4.3: Un sensor de pitot .....	15
Ilustración 4.4: Sensor de pitot del UAV .....	16
Ilustración 4.5: Controlador de pitot.....	16
Ilustración 4.6: Prueba de túnel de viento.....	17
Ilustración 4.7: Prueba de velocidad en túnel de viento del pitot 1: datos en bruto .....	18
Ilustración 4.8: Prueba de velocidad en túnel de viento del pitot 1: mínimos y máximos.....	19
Ilustración 4.9: Prueba de velocidad en túnel de viento del pitot 1: media y error.....	20
Ilustración 4.10: Prueba de velocidad en túnel de viento del pitot 2: datos en bruto .....	21
Ilustración 4.11: Prueba de velocidad en túnel de viento del pitot 2: mínimos y máximos.....	21
Ilustración 4.12: Prueba de velocidad en túnel de viento del pitot 2: media y error.....	22
Ilustración 4.13: Error de medición del pitot 1 .....	23
Ilustración 4.14: Error de medición del pitot 2.....	23
Ilustración 4.15: Polinomio de error de grado 2 del pitot 1 .....	25
Ilustración 4.16: Polinomio de error de grado 3 del pitot 1 .....	25
Ilustración 4.17: Polinomio de error de grado 4 del pitot 1 .....	25
Ilustración 4.18: Polinomio de error de grado 2 del pitot 2.....	26
Ilustración 4.19: Polinomio de error de grado 3 del pitot 2.....	26
Ilustración 4.20: Polinomio de error de grado 4 del pitot 2.....	26
Ilustración 4.21: Corrección con cada polinomio del error del pitot 1 .....	27
Ilustración 4.22: Corrección con cada polinomio del error del pitot 2.....	27
Ilustración 4.23: Comportamiento de los errores del pitot 1 .....	28
Ilustración 4.24: Comportamiento de los errores del pitot 2 .....	29
Ilustración 4.25: Resultado de corrección del pitot 1 con cada polinomio .....	31
Ilustración 4.26: Resultado de corrección del pitot 2 con cada polinomio .....	31
Ilustración 4.27: Sensor de inclinación IMU .....	32
Ilustración 4.28: Los tres ejes de giro del chip.....	32
Ilustración 4.29: Los tres ejes de giro de un avión.....	33
Ilustración 4.30: IMU del UAV.....	33
Ilustración 4.31: Localización del IMU en el UAV .....	34
Ilustración 4.32: Prueba de funcionamiento del IMU.....	34
Ilustración 4.33: Resultados de la prueba del IMU.....	35

Ilustración 4.34: Error de medición de pitch.....	36
Ilustración 4.35: Error de medición de roll .....	36
Ilustración 4.36: Altimetro del UAV .....	37
Ilustración 4.37: Prueba de medición de altura con altímetro .....	38
Ilustración 4.38: Prueba de medición de altura con GPS.....	38
Ilustración 4.39: Diferencia constante de medidas.....	39
Ilustración 5.1: Hélice de pruebas .....	41
Ilustración 5.2: Velocidad de vuelo media del UAV .....	42
Ilustración 5.3: Perfiles disponibles .....	43
Ilustración 5.4: Motor del UAV .....	44
Ilustración 5.5: UAV .....	45
Ilustración 5.6: Flota de UAVs .....	45
Ilustración 5.7: UAV completo en AVL.....	48
Ilustración 5.8: Vista frontal .....	48
Ilustración 5.9: Coeficientes de arrastre parasitarios .....	51
Ilustración 5.10: Aproximación de $CD_0$ en función de alfa .....	52
Ilustración 5.11: $CL$ en función del ángulo de sustentación.....	55
Ilustración 5.12: Curva ajustada de $CL$ en función de $\alpha$ .....	56
Ilustración 5.13: Efecto del ángulo de dihedro .....	58
Ilustración 6.1: Datos de simulación de vuelo.....	67
Ilustración 6.2: Datos de simulación de prueba de subida .....	67
Ilustración 6.3: Datos de simulación de prueba de bajada.....	68
Ilustración 6.4: Datos de simulación de prueba de altura constante .....	68
Ilustración 6.5: Prueba de vuelo: datos de altitud .....	69
Ilustración 6.6: Prueba de vuelo: tramos estables seleccionados .....	69
Ilustración 6.7: Datos seleccionados .....	70
Ilustración 6.8: Datos de subida .....	70
Ilustración 6.9: Datos de bajada .....	70
Ilustración 6.10: Datos de altitud constante .....	71

# 1. INTRODUCCIÓN

## 1.1. UAV

UAV, o Unmanned Aerial Vehicle, es el término que se utiliza para referirse a los vehículos aéreos con propulsión sin piloto a bordo, que usan la fuerza aerodinámica como sustentación, y pueden volar de forma autónoma siguiendo un software programado o ser pilotados remotamente por un operador humano con un enlace de vídeo, sin necesitar una línea de visión directa con el aparato.



*Ilustración 1.1: Unmanned Aerial Vehicle*

Además se considera que pueden realizar su tarea asignada sin intervención humana o con intervención humana puntual. Pueden ser reemplazables o reutilizables, y pueden transportar cargas peligrosas o no peligrosas tanto en ámbito militar como civil, si se les ha asignado esa tarea, o realizar labores de reconocimiento u otro tipo de misión que no requiera una capacidad de carga.

En español se les denomina Vehículos Aéreos No Tripulados (VANT) o Sistemas Aéreos No Tripulados (SANT), aunque son más conocidos coloquialmente por el nombre de “drones”, nombre que se daba a los aviones de control remoto que se utilizaban como blanco de tiro en ejercicios militares.

Los UAVs actuales están pasando de ser únicamente vehículos aéreos pilotados remotamente a robots cada vez más autónomos y versátiles. Este grado cada vez mayor de autonomía es una cuestión que genera polémica, generalmente por la aparición y uso de determinadas aplicaciones que pueden poner en riesgo la privacidad de las personas, o en ciertos casos, vidas humanas.

### 1.1.1. Clasificación de los UAV

Los UAVs se pueden clasificar en base a muchos criterios útiles. Se pueden agrupar según su ámbito de uso, alcance, diseño aerodinámico y tipo de ala, altitud máxima o autonomía de vuelo. Aquí se ha escogido la clasificación en función de su peso como opción. En ella se pueden distinguir 7 categorías, en las que además se especifican la altitud normal operativa, el radio de operación, la autonomía de vuelo, la altitud y sus usos más comunes:



Categoría	Peso del UAV (kg)	Altitud normal operativa (m)	Radio de operación (km)	Autonomía de vuelo	Altitud	Usos más comunes
MICRO	<2	Más de 60	5	Pocas horas	Muy baja	Reconocimiento, inspección, vigilancia
MINI	2 a 20	Más de 900	25	Más de 2 días	Baja	Vigilancia, recogida de información
SMALL	20 a 150	Más de 1500	50	Más de 2 días	Baja	Vigilancia, recogida de información
TACTICAL	150 a 600	Más de 3050	200	Más de 2 días	Baja	Vigilancia, recogida de información
MALE	>600	Más de 13700	Ilimitado	Días o semanas	Media	Vigilancia, transporte de cargas
HALE	>600	Más de 19800	Ilimitado	Días o semanas	Alta	Vigilancia, recogida de información, retransmisión de señal
STRIKE/COMBAT	>600	Más de 19800	Ilimitado	Días o semanas	Alta	Vigilancia, recogida de información, retransmisión de señal

*Tabla 1.1: Clasificación de UAVs*

## **1.2. Simuladores**

Por otra parte, un simulador es un software informático que permite reproducir virtualmente un sistema real. Los simuladores intentan reproducir de la forma más fiel posible tanto el entorno como el comportamiento del sistema real en el software.

El objetivo es que el usuario adquiera conocimiento a partir del trabajo exploratorio, la inferencia y el aprendizaje por descubrimiento. Los simuladores permiten realizar cambios y ver como varía el comportamiento del modelo simulado ante tales cambios, que pueden ser cambios en el modelo o en el entorno. Esto supone una enorme ventaja sobre la alternativa que consistiría en realizar experimentos físicos para un entorno diferente. Si hiciéramos estas modificaciones en el sistema real supondrían en algunos casos un gran gasto económico, serían más laboriosos y serían mucho más lentos a la hora de ser estudiados o probados. En algunos casos el experimento sería inviable o directamente imposible, y tendrían una componente aleatoria sin ninguna posibilidad de control por parte nuestra en el caso de experimentos que necesitan unas condiciones climatológicas concretas, por ejemplo.

Gracias a los simuladores podemos permitirnos un estudio más detallado y preciso con una inversión menor y en un plazo de tiempo más breve.

La simulación implica la creación de un modelo o sistema para la que se puede recurrir a varias técnicas. Se puede elaborar un modelo del equipo y virtualizarlo por hardware con el equipo real o se puede usar el mismo software que se usa en el equipo real pero usándolo en un ordenador convencional, con lo que se abaratan los costes.

A la hora de diseñar una simulación, podemos distinguir varias fases:

1. **Definición del sistema:** se trata de hacer un análisis del sistema para determinar su interacción con otros sistemas, sus restricciones, las variables que intervienen y las medidas de efectividad que se van a usar para definir y estudiar el sistema.
2. **Formulación del modelo:** consiste en definir todas las variables que forman parte de el, sus relaciones lógicas y los diagramas de flujo que describen el modelo.
3. **Preparación de los datos:** es la recolección de datos para su implementación en los parámetros de las variables del modelo.
4. **Traslación del modelo:** se define como la programación del modelo mediante algún lenguaje de programación o la utilización de algún simulador para procesarlo en el ordenador y obtener los resultados.
5. **Validación:** consiste en encontrar las deficiencias del modelo o de los datos implementados y su corrección.
6. **Planificación estratégica:** se encarga del diseño de los experimentos que se van a simular y de determinar el número de simulaciones que se van a realizar.
7. **Planificación táctica:** consiste en determinar las condiciones en las que se ha de realizar cada versión del experimento. Estos factores se escogen según afecten más o menos al comportamiento de nuestro sistema.
8. **Experimentación:** ejecución de los experimentos planificados
9. **Interpretación:** estudio de los resultados obtenidos
10. **Documentación:** redacción del modelo y su traducción informática y los resultados obtenidos tras la simulación.

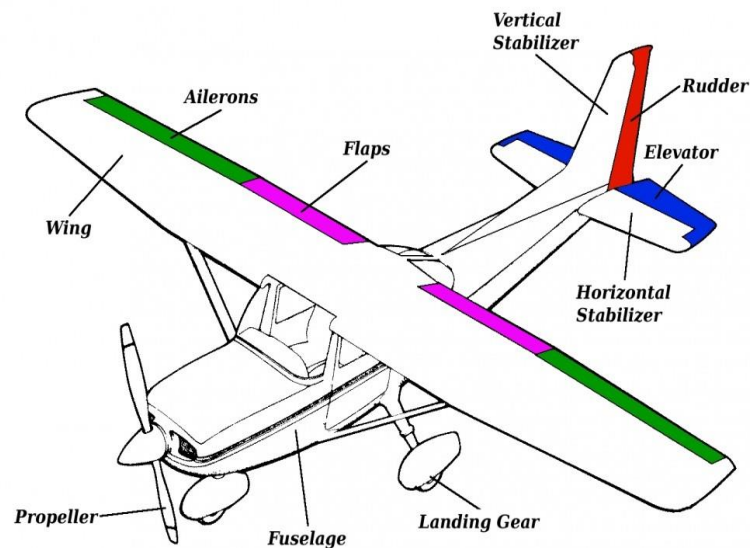
### 1.2.1. Tipos de simuladores

- Simuladores sociales: son software que intentan predecir el comportamiento de una sociedad o de un individuo cuando se produce un cambio que les afecta. Se incluyen en este grupo a los simuladores de vida, políticos, de entretenimiento,...
- Simuladores económico-financieros: son aquellos que tienen como objetivo predecir cuales van a ser los resultados de una operación económica o un cambio que pueda afectar a la economía, ya sea de una empresa en particular o de un estado. Podemos agrupar aquí a los simuladores de negocios, políticos, energéticos,...
- Simuladores de procesos: éste es el grupo más generalista, porque agrupa simuladores de una gran variedad de sistemas, ya sean simuladores de procesos de fabricación, simuladores industriales, físicos, químicos, simuladores de redes, de ciberdefensa, logística,.... Intentan reproducir fielmente las condiciones reales de trabajo para obtener los resultados más realistas posibles, y aplicar posteriormente los métodos a la realidad para mejorar un proceso o encontrar fallos y corregirlos.
- Simuladores de aprendizaje: en éstos se incluyen todo tipo de simuladores de manejo de vehículos para autoescuelas, de barcos, de vuelo, musicales, médicos, etc., y su objetivo es únicamente el aprendizaje o entrenamiento por parte de los usuarios, antes de realizar estas tareas con los sistemas reales.

Los simuladores de vuelo están incluidos en el grupo de simuladores de aprendizaje. Para el estudio del modelo de los parámetros del modelo del que partimos tomaremos datos que compararemos con el modelo real, y lo modificaremos si es necesario, para intentar mejorar la precisión del comportamiento del modelo.

### 1.3. Breve descripción de nuestro UAV

Conviene explicar algunas de las características generales de las aeronaves, para poder entender cómo funcionan los aviones en general y el UAV en cuestión en particular. Estas características son tales como su forma, sus elementos de control, cómo funcionan, y su influencia en la dinámica de vuelo del UAV.



*Ilustración 1.2: Superficies de control de un avión*

Las funciones de estas superficies son:

- **Derivas** (rudder): Se encarga de controlar los movimientos de guiñada.
- **Profundidad** (elevator): Son los responsables del movimiento de cabeceo.
- **Alerones** (aileron): Tienen la función de provocar los movimientos de alabeo de la aeronave.
- **Flaps**: Ejercen de frenos durante el aterrizaje inclinados hacia arriba o generan una mayor sustentación durante el despegue con un ángulo negativo.

Estas superficies controladoras no solo ejercen estos movimientos, si no que provocan otros efectos secundarios debido a los momentos y fuerzas que generan con respecto al centro de gravedad, al no estar el centro de presiones de la superficie en la misma horizontal que el CG. Se podría explicar en mayor profundidad pero a nosotros nos es suficiente con esto para entender nuestro caso.

Nuestro UAV, tiene dos derivas en las colas verticales, y unas superficies de control que son un híbrido entre los elementos de profundidad y los alerones llamado elevones (de sus nombres en inglés elevator+aileron). Están situados en la parte trasera de las alas del UAV, a lo largo de todo el borde de salida como se puede ver en el plano del ANEXO X: Plano del UAV. Estos elevones ejercen la función de generar los movimientos de cabeceo y de alabeo según se requiera en cada momento.



*Ilustración 1.3: UAV modelo Hércules XL 76", vista trasera*

En ésta imagen podemos observar nuestro modelo de UAV, el Hércules XL 76", con el que trabajaremos a lo largo de este proyecto. A continuación se detallan algunas de sus características:

- Envergadura: 1,94 m (76 pulgadas, de ahí su nombre)
- Posee 2 winglets en las puntas de las alas
- 2 colas verticales que integran en ellas a las derivas. La función de las derivas es provocar giros alrededor del eje vertical
- 2 hélices que giran en sentido contrario. Visto desde atrás, la izquierda gira en sentido horario y la derecha en sentido antihorario.
- 4 elevones en su borde de salida. Son un híbrido entre los alerones y el control de profundidad, y su función es la de realizar giros alrededor del eje longitudinal y provocar ascensos o descensos
- Fuselaje: es un cuerpo de forma geométrica irregular, como se puede observar en la imagen.

Esto solo es una descripción muy breve y superficial del UAV que modelaremos, simplemente para entender como es geoméricamente y algunas de sus características principales. Todos estos datos y otros sobre el UAV los iremos aclarando y explicando como han sido calculados a lo largo del proyecto, cuando surga la necesidad de aplicarlos a nuestro trabajo. Además también iremos explicando otros términos técnicos cuando sea necesario.

## 2. PRECEDENTES Y OBJETIVOS

El presente proyecto se integra en la línea de investigación sobre UAVs del GII (Grupo Integrado de Ingeniería) de la UDC. El objetivo global de esta línea es el de desarrollar un enjambre de UAVs capaces de volar de forma autónoma y colaborar entre sí para realizar una tarea común (vigilancia forestal contra incendios, rescate de náufragos,...).

Este trabajo se centra en mejorar los avances en el paso a simulación del modelo de UAV utilizado por la línea de investigación. Se ha dado el caso en una investigación previa de que existían notorias diferencias en el comportamiento del modelo virtual del UAV y el prototipo real. Dichas diferencias fueron consideradas como defectos físicos del UAV o comportamientos no previstos y no contemplados en la geometría del UAV virtual, y fueron implementadas una serie de correcciones en el modelo para corresponderse con el modelo real. En éste trabajo se repetirá el proceso de modelado virtual del UAV, con especial énfasis en localizar el origen de las divergencias en el comportamiento del modelo virtual y el modelo físico. De forma adicional se realizará una revisión completa del UAV para comprobar que el comportamiento esperado y el comportamiento real no difieren debido a defectos del propio UAV.

El programa de simulación de código abierto a emplear es el FlightGear, junto con el modelo de dinámica de vuelo (FDM) JSBSim para determinar el comportamiento físico de la aeronave en la simulación. Ambos son softwares de código abierto, condición impuesta como fundamental para el trabajo. Otros posibles entornos de programación necesarios serán especificados y estudiados en sus respectivos puntos más adelante en el trabajo.

Finalmente analizaremos el comportamiento del UAV en simulación y en vuelo, para comprobar si existe mejora y por cuánto.

## 3. PROBLEMAS OBSERVADOS Y POSIBLES CAUSAS

A pesar del cuidadoso desarrollo realizado para conseguir un modelo virtual lo más fiel posible en comportamiento al modelo real, el UAV físico tuvo un comportamiento muy diferente del obtenido en simulación. El UAV alcanzó velocidades medias y puntuales mucho más elevadas de lo que las previsiones mostraron, trazando recorridos mucho más amplios y realizando maniobras mucho menos controladas y más rápidas que su contraparte virtual.

Se ha considerado las posibles causas de esta discrepancia. El origen puede estar en un defecto en el modelo virtual del UAV, o en una variable de la mecánica del aire en la simulación. También se ha supuesto que el fallo puede estar en el apartado físico del proyecto. El UAV real puede tener alguna característica física en el fuselaje que no se haya tenido en cuenta a la hora de modelar su contraparte virtual, pero que haya de todas formas tenido un gran impacto en el resultado.

Estas posibles causas son comprobables mediante la revisión del UAV y de su modelo virtual, pero existe un tercer posible origen para las diferencias de resultados: las mediciones. Los datos que se conocen del comportamiento en vuelo del UAV virtual los proporciona el programa que se emplee en la simulación, pero para el modelo real dependemos de sensores. Si existe un defecto en el funcionamiento de estos sistemas pueden entregar datos falsos del vuelo real, que confundan en las comparaciones. Si el fallo está en el apartado de detección en lugar de en los programas, puede darse la situación de que el aparato funcione correctamente siguiendo la programación que posee, pero basándose en unos datos erróneos.

Teniendo estas consideraciones se realizará una revisión completa del UAV y de sus sistemas, para comprobar su buen funcionamiento electrónico, y se reconstruirá el modelo virtual en busca de posibles fallos.

## 4. REVISIÓN: APARTADO FÍSICO



*Ilustración 4.1: UAV modelo Hércules XL 76" , vista lateral*

El conjunto de trabajo del UAV consta de dos partes susceptibles de crear un fallo o defecto: el cuerpo, y la simulación. Los posibles fallos generados por el cuerpo físico del dron proceden de los sensores. Los sensores son los dispositivos que permiten al UAV percibir el entorno, y al operario u otro programa de control a percibir ese entorno tal cual lo percibe el UAV. Dicha visión del mundo puede estar viciada por lo que el comportamiento que genera esa percepción puede diferir de lo que se esperaría desde un punto de vista con una percepción diferente, como podría ser la del operario.

Los sistemas de sensores son los focos de defectos de funcionamiento más probables, y suponiendo que el paso a simulación del UAV es correcto, como éstos sistemas no se tienen a priori en cuenta en el impacto aerodinámico del conjunto, es posible que un defecto en su funcionamiento haya podido crear la discrepancia en los resultados de la simulación con la realidad, ya que en el modelo virtual no se contemplan.

### 4.1. Sensores



*Ilustración 4.2: Electrónica del UAV*

Los sistemas de sensores con los que viene equipado el UAV consisten en:

1. Tubos de pitot: sensores de velocidad del viento que sirven para determinar la velocidad del UAV en base a ésta.
2. IMU: sensores de inclinación del UAV con respecto a una horizontal, que permiten conocer la orientación del UAV en función del pitch y roll (movimientos de giro en torno a los ejes dimensionales del UAV).
3. Altímetro y GPS: el altímetro y el GPS establecen la altitud del UAV. Mientras que el altímetro detecta la presión del aire simplemente, independientemente de otras variables, el GPS contrasta la información de posición y altitud vía satélite.

#### 4.1.1.Pitots

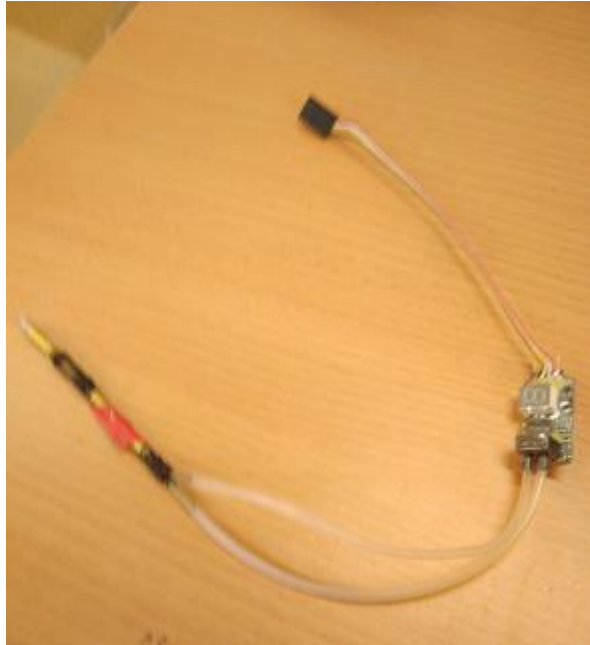


*Ilustración 4.3:Un sensor de pitot*

Los tubos de pitot son sensores de presión del aire. Utilizan la diferencia de presión entre dos puntos del tubo para obtener una aproximación fiable de la velocidad a la que el aire impacta en el tubo. Este detector se emplea para saber a qué velocidad se desplaza el UAV.

Las discrepancias de comportamiento entre el UAV y su simulación radican principalmente en las diferencias de velocidad, por lo que se sospecha que puede existir un defecto en este sistema de sensorización concreto.

El sistema de lectura del pitot consta de los siguientes componentes:



*Ilustración 4.4: Sensor de pitot del UAV*

El tubo de pitot detecta las variaciones de presión del aire y las transmite al controlador, que traduce dicha información a velocidad del aire. Finalmente retransmite la información en valores hexadecimales al procesador del UAV para poder disponer de ellos en el control de velocidad. El controlador es un Eagle Tree Systems Airspeed Microsensor V3. Su manual de instrucciones se puede encontrar en el ANEXO VII: Manual de instrucciones Airspeed Microsensor.



*Ilustración 4.5: Controlador de pitot*

Para comprobar el buen funcionamiento del sistema se realizarán una serie de pruebas en un entorno controlado, para comprobar cómo funciona. Éste entorno controlado es proporcionado por un ensayo con túnel de viento. Para poder realizarlas es necesario extraer el sistema electrónico asociado al pitot del UAV.





*Ilustración 4.6: Prueba de túnel de viento*

Las pruebas realizadas consisten en lo siguiente:

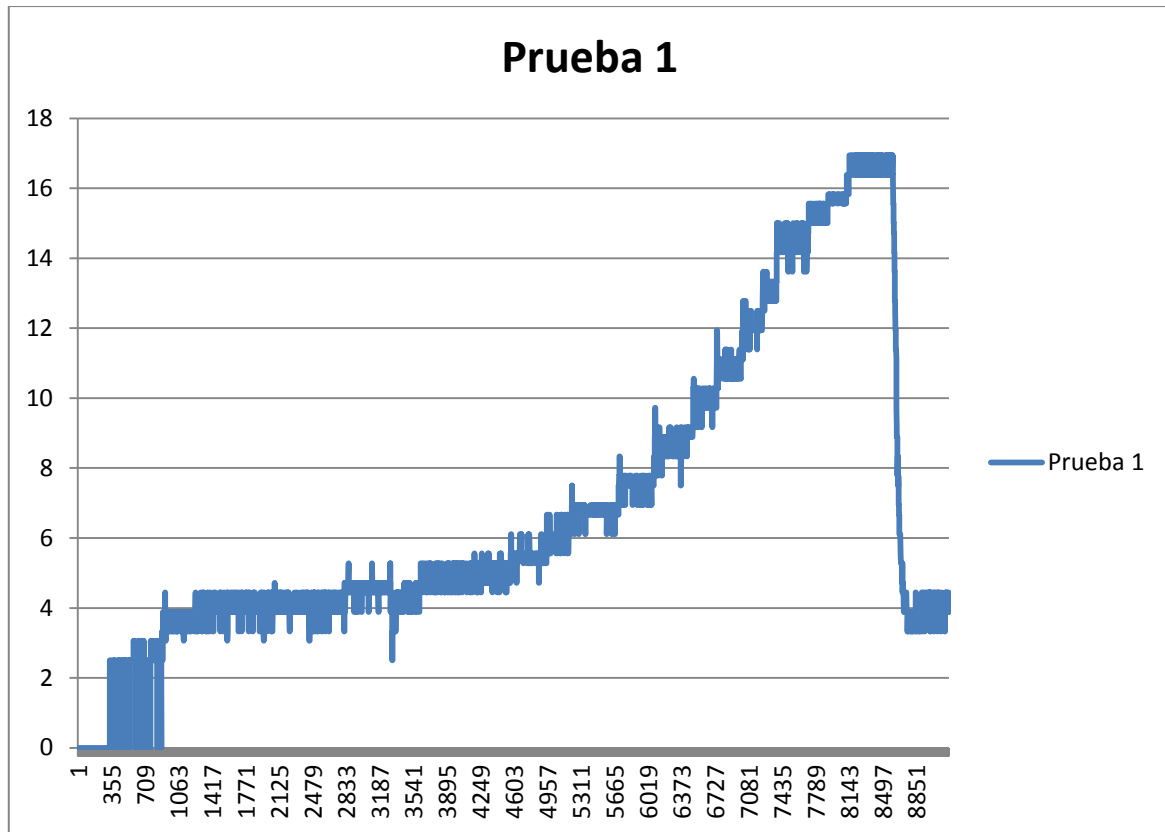
1. Se somete el pitot a una corriente de viento de velocidad conocida.
2. Se mide simultáneamente el resultado obtenido por el pitot.
3. Se comparan datos para conocer la precisión de la medida realizada.

Los datos recogidos por el pitot son diferencias de presión. Esto debe ser traducido a medidas de velocidad mediante un programa integrado en el controlador del sensor. Los datos obtenidos están expresados en código hexadecimal, el código que emplea la electrónica de todo el UAV para transmitir información entre programas, que utilizan C: como lenguaje de programación. Para poder leerlos directamente debe realizarse un último paso de conversión a decimal antes de poder considerarlos datos útiles para el análisis. Todo este proceso de manipulación de datos se debe realizar con un ordenador y un dispositivo lector-traductor de programas que establezca una vía de comunicación con la cpu del controlador.

Este proceso se realizará para un conjunto de velocidades del viento, las cuales se considera que son las velocidades a las que generalmente trabajará el UAV. El margen de velocidades consideradas relevantes abarca desde 10m/s hasta 20m/s, que es el margen de velocidad en el que se espera que trabaje el UAV. Idealmente las medidas obtenidas por el pitot deben ser iguales a las marcadas por el controlador del túnel de viento.

Así mismo el proceso se realizará con varios pitots para comprobar si es un problema puntual o afecta a toda la flota de UAVs.

El conjunto de datos obtenidos en una prueba, una vez han sido tratados mediante el proceso mencionado arriba suele tener el siguiente aspecto:

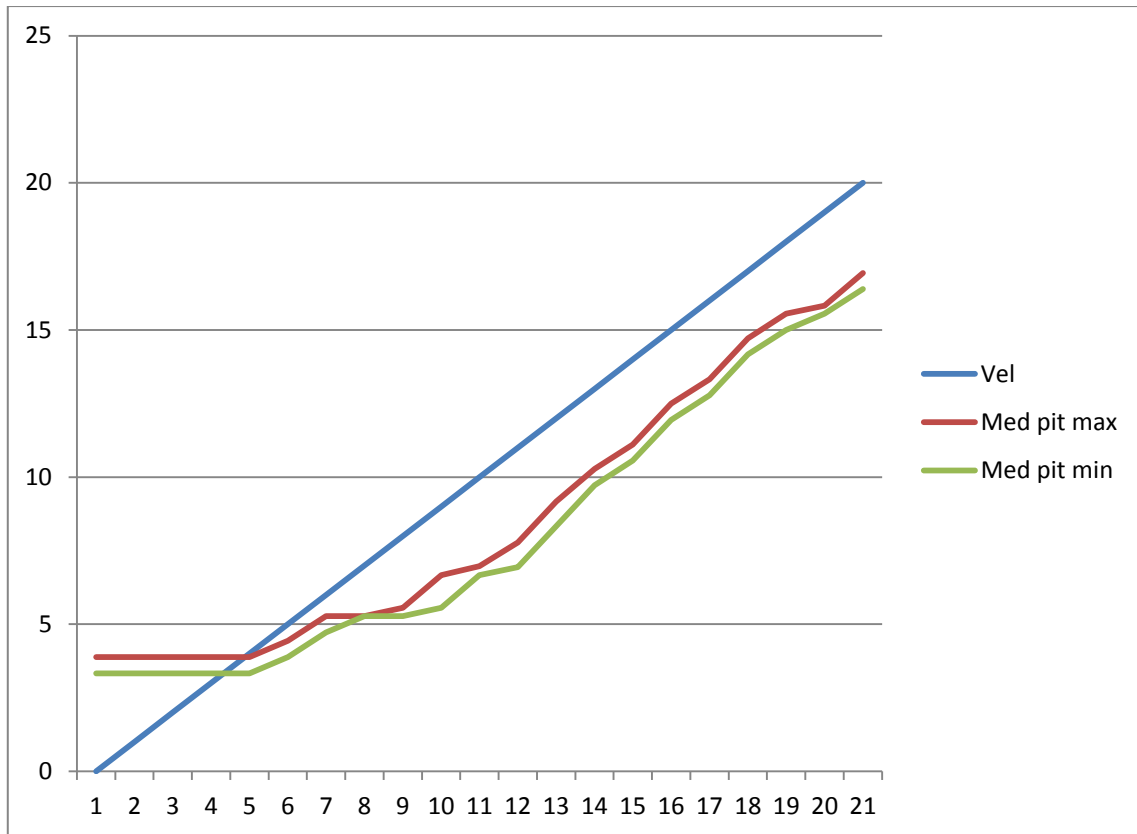


*Ilustración 4.7: Prueba de velocidad en túnel de viento del pitot 1: datos en bruto*

Este gráfico muestra los resultados de la primera prueba en túnel de viento del que llamaremos pitot 1.

Puesto que el sensor recoge datos de forma ininterrumpida cada décima de segundo, el resultado será una lista de datos muy grande y con pocas variaciones en un grupo pequeño de datos. Si se muestran en un gráfico se puede apreciar la curva de resultados, resaltando la forma de escalera que adopta. El método de obtención de datos elegido consiste en exponer el sensor a una corriente de aire de velocidad conocida, variando dicha velocidad en un punto cada cierta cantidad de tiempo constante, hasta recorrer todo el rango de valores de velocidad de interés. De ahí la forma de escalera. Cada escalón se corresponde con un período de tiempo durante el cual la velocidad del aire se mantiene constante.

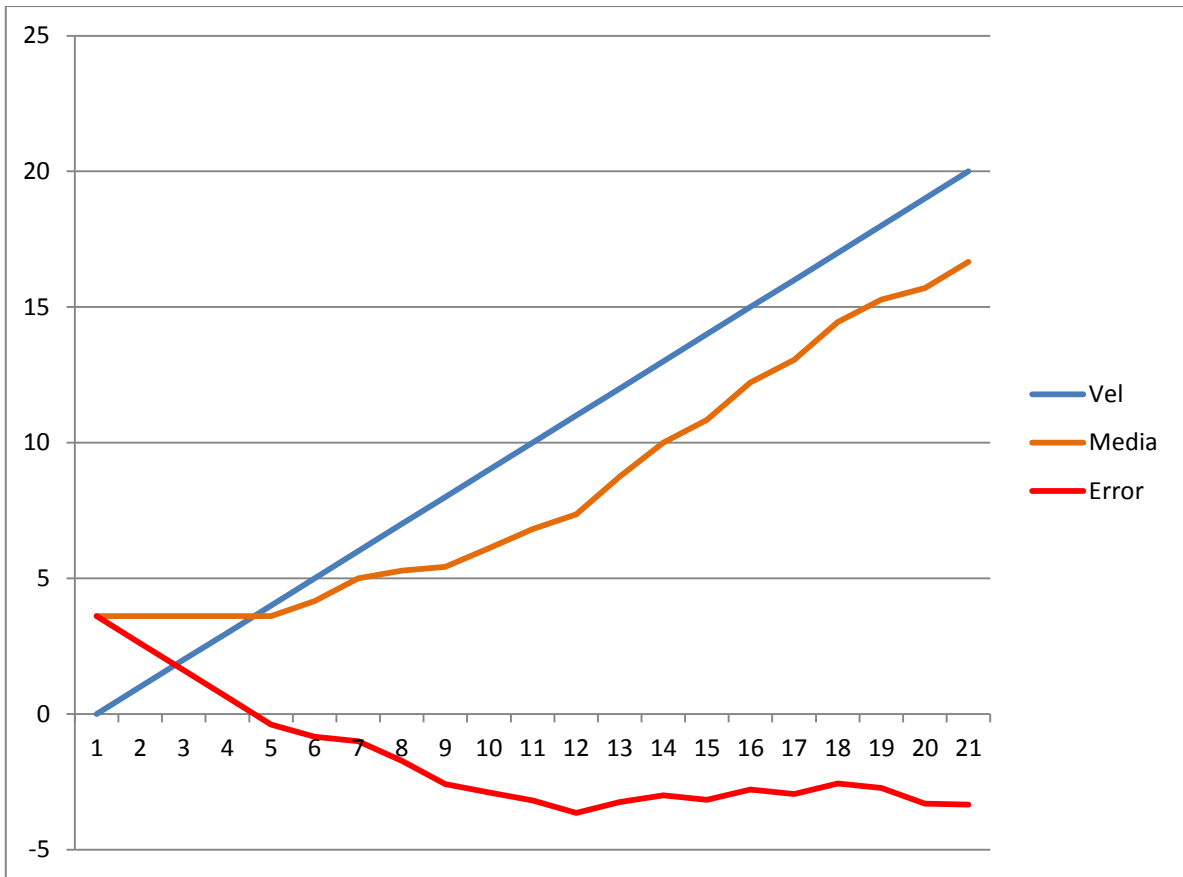
Si se extrae una muestra significativa de valores máximo y mínimo de cada escalón, se puede obtener el siguiente gráfico:



*Ilustración 4.8: Prueba de velocidad en túnel de viento del pitot 1: mínimos y máximos*

Nótese que hay muchos menos datos con los que trabajar, ya que ahora nos interesa obtener la tendencia de la forma más clara posible. Hemos trazado también una línea que refleja el valor de la velocidad de la corriente de aire controlado, por lo que representa el funcionamiento ideal al que debería aproximarse el sensor.

Se puede observar claramente ya desde un principio que existe una gran diferencia en los resultados. El pitot 1 mide bastante menos velocidad del viento de la que existe en la realidad.

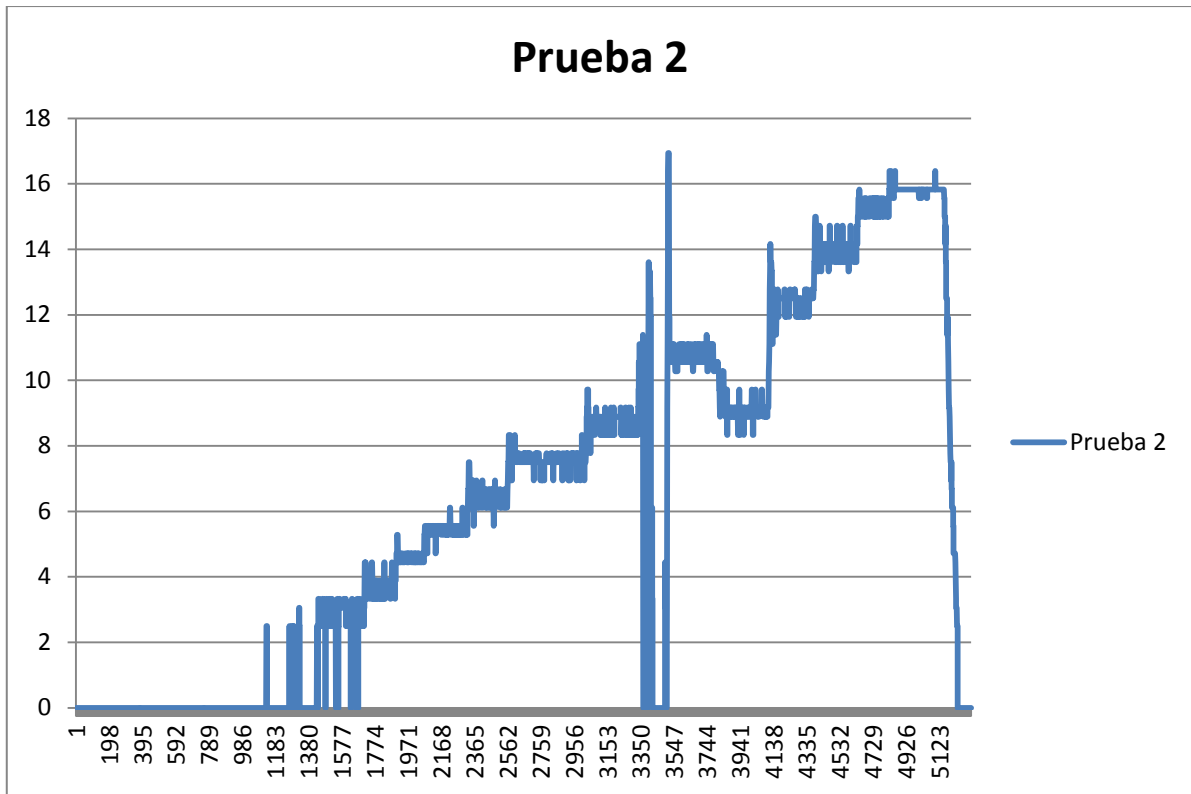


*Ilustración 4.9 Prueba de velocidad en túnel de viento del pitot 1: media y error*

En la siguiente gráfica se puede ver un paso más en la interpretación de los datos. Se ha trazado un valor medio de los valores detectados por el pitot en cuestión y se ha comparado con el valor teórico para obtener una gráfica de error. Éstas gráficas de error son el principal objetivo de la prueba. Si logramos obtener una ecuación aproximada del error se puede realizar una rectificación en el programa para ajustar la respuesta del pitot a los valores ideales.

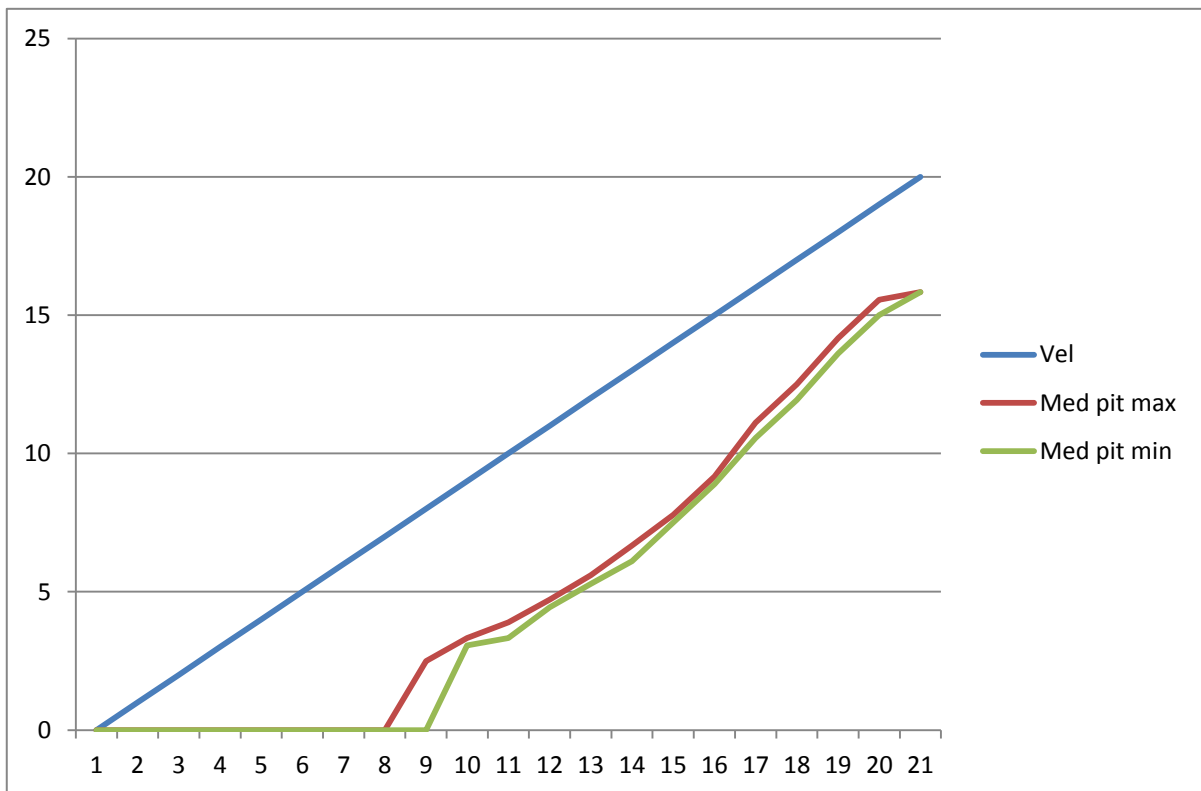
Obsérvese que existe un umbral de detección, no siendo el sensor capaz de emitir una respuesta significativa para velocidades inferiores a 5m/s. Recordemos que el dron trabaja en un margen de velocidades de 10 a 20m/s, por lo que el umbral de detección está en un margen aceptable y que no afecta a la precisión de las medidas.

Repiremos el proceso con un segundo pitot, al que llamaremos pitot 2.

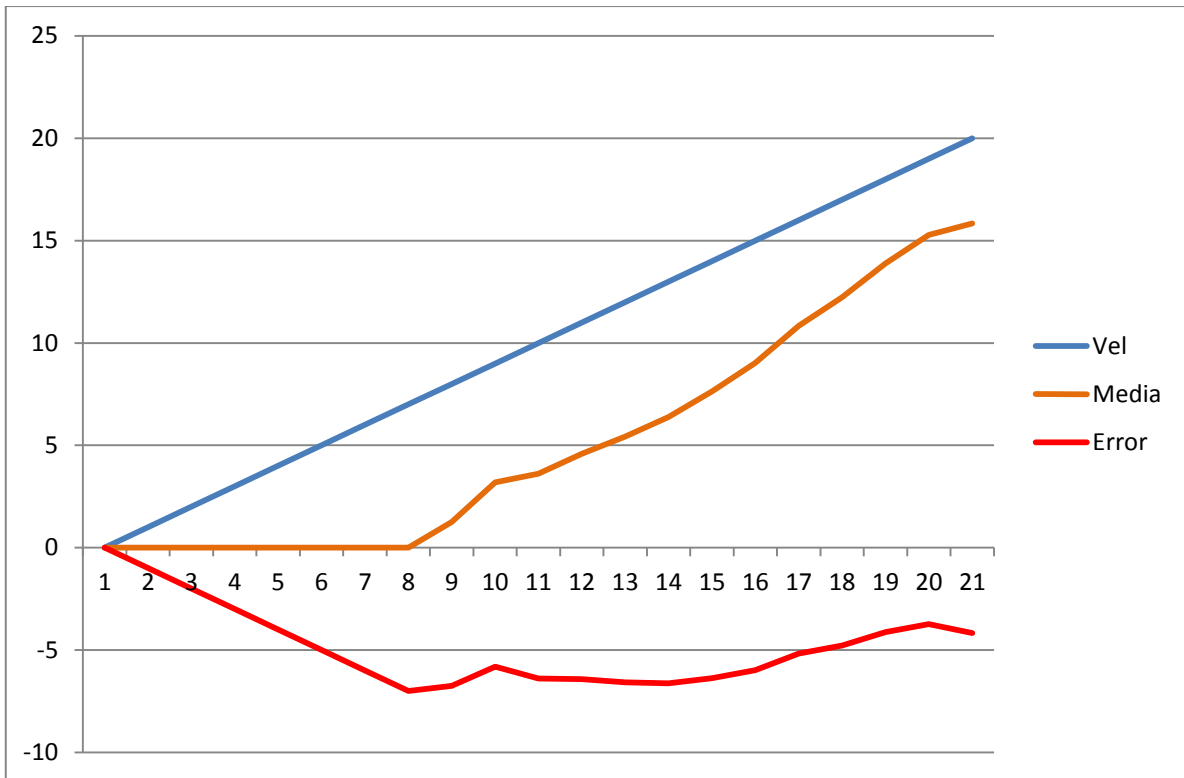


*Ilustración 4.10: Prueba de velocidad en túnel de viento del pitot 2: datos en bruto*

Datos en bruto y sin procesar. Observamos la presencia de un escalón abrupto debido a una interrupción de la prueba y su posterior reanudación. Sin embargo gracias al sistema de escalones no tenemos problema para identificar los datos de valor. En el siguiente paso se puede ignorar estos detalles.



*Ilustración 4.11: Prueba de velocidad en túnel de viento del pitot: mínimos y máximos*



*Ilustración 4.12: Prueba de velocidad en túnel de viento del pitot 2: media y error*

En esta nueva gráfica se puede ver el comportamiento del pitot 2. El comportamiento es claramente diferente, particularmente a velocidades muy bajas. Esto se debe al umbral de detección programado en los sensores. El pitot 1 tenía un umbral de detección en un nivel de velocidad de 5m/s. Éste pitot tiene un umbral de detección más problemático, quedando sin medidas las velocidades inferiores a 7m/s, más cerca del umbral de funcionamiento, por lo que podría afectar al comportamiento del sensor.

Se podría descartar el pitot 2 del proceso, puesto que solo es necesario disponer de uno corregido para cada UAV. Se podría hallar un método para reducir el error del pitot 1 concretamente, ignorando el pitot 2, que no podría ser corregido de la misma forma (por ejemplo realizando una tabla de datos correspondientes). No obstante, teniendo en cuenta que éste proyecto forma parte de un sistema más grande que emplea varios UAVs, hemos optado por una metodología que permita realizar correcciones para todas las unidades con sus correspondientes pitots, siguiendo de forma sistemática todos los pasos aquí enunciados. Por este motivo, aunque el pitot 1 sea el que realmente se vaya a poner en práctica con vuelos reales, el proceso de corrección se realiza con los dos pitots, para conseguir un proceso que satisfice las necesidades de ambos siguiendo los mismos pasos.

Hemos conseguido gráficas de la evolución del error de medida para cada valor de velocidad del viento dentro del rango de interés.

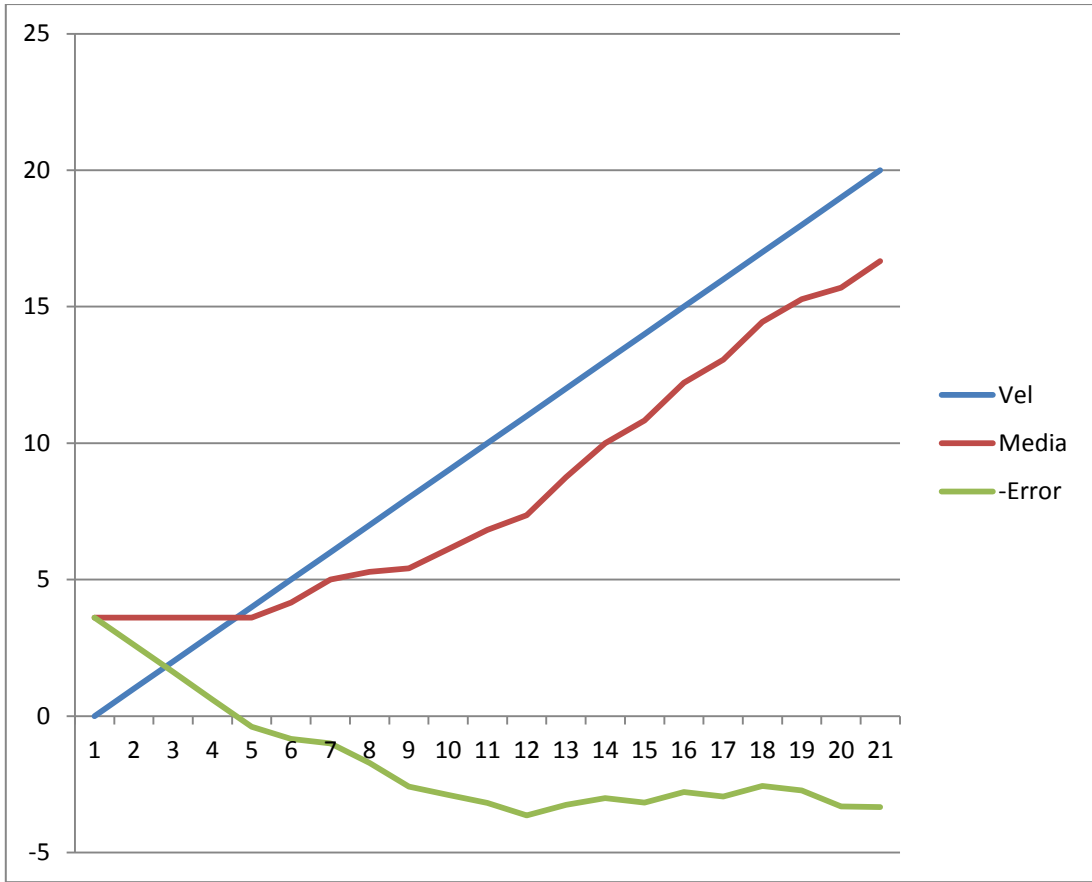


Ilustración 4.13: Error de medición del pitot 1

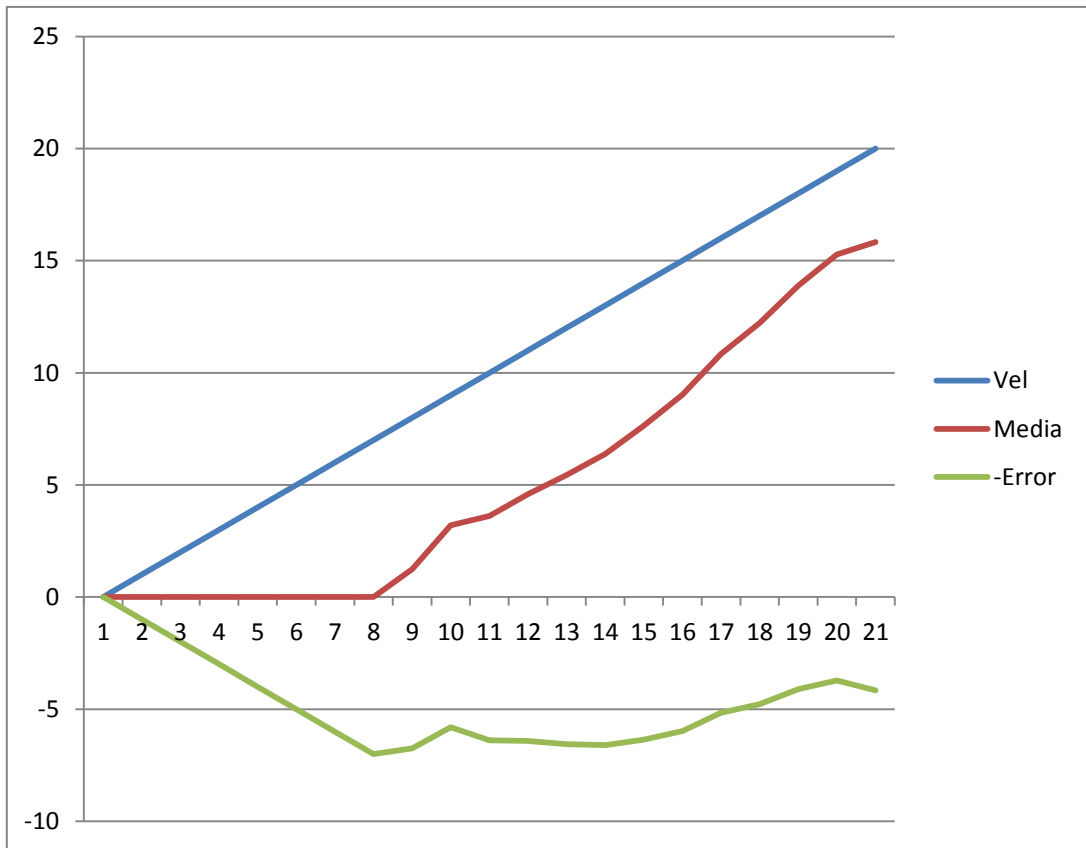


Ilustración 4.14: Error de medición del pitot 2

El método más directo y simple para eliminar errores de un sistema programado es obtener una función que se aproxime a la tendencia de esos errores, para posteriormente incorporarla a la función de transferencia entre la entrada del pitot y la salida de datos proporcionada por el chip. Puesto que existe ya una parte del sistema que incluye programación matemática no es difícil aprovecharla para incluir esa función de corrección de error.

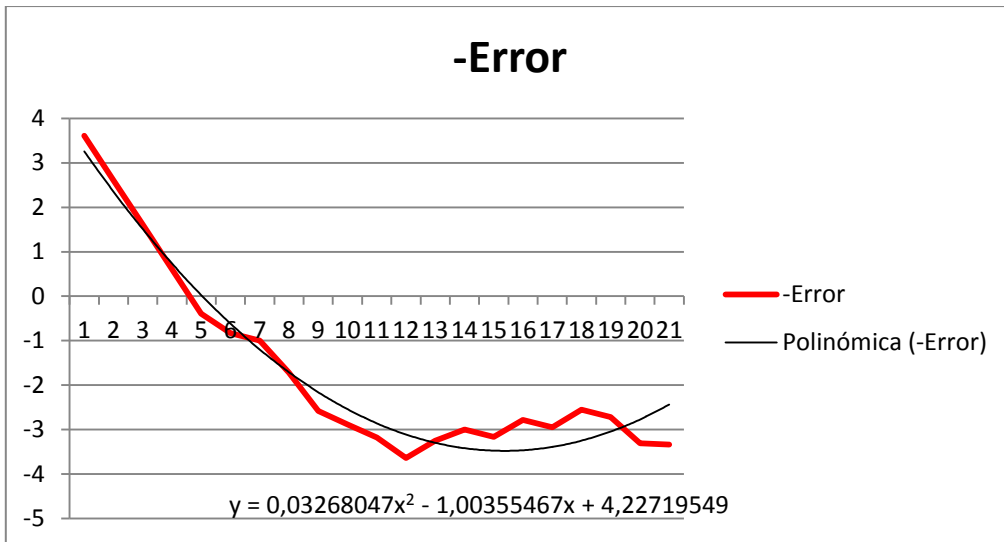
¿Cómo obtener la función de error? Lo que queremos obtener es una función que trace en el gráfico una tendencia del error que hemos obtenido. Existen muchas maneras de obtener esas funciones. Generalmente los programas de manejo de tablas de datos y estadística tienen incorporada la opción de realizar líneas de tendencia como una de sus funciones básicas. Las líneas de tendencia se obtienen normalmente empleando el método de los mínimos cuadrados, de forma que teóricamente podría hacerse un cálculo a mano, si el volumen de datos no es excesivo.

En el caso de este proyecto el manejo de los datos se ha realizado con el software Excel 2010, y tiene también incorporada la función de cálculo de tendencia por lo que emplearemos este recurso por ser sencillo y de fácil acceso con cualquier ordenador, pues está incorporado para todas las versiones de Excel.

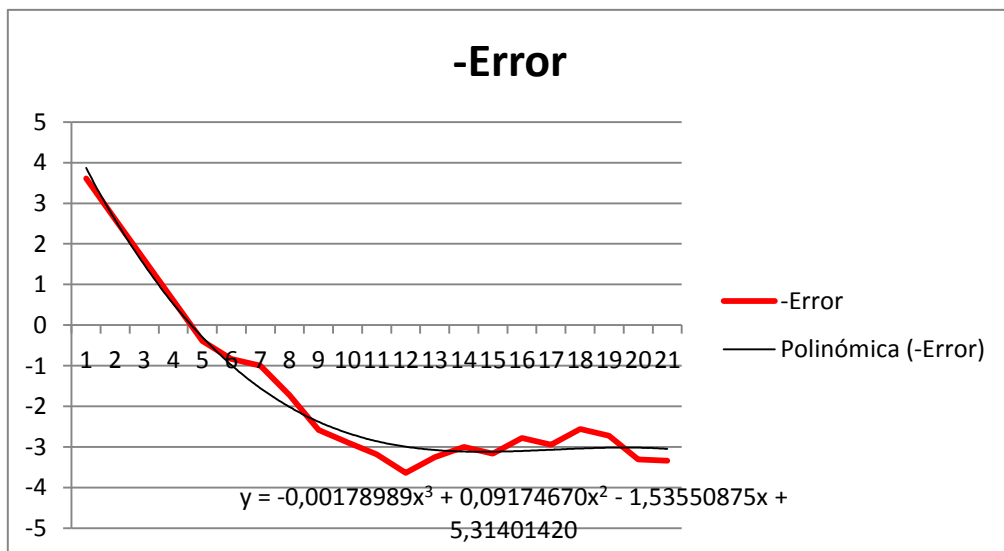
El cálculo de una función de tendencia no hace distinciones entre tipos de funciones, por lo que teóricamente podría emplearse para obtener una función de corrección para cada posible UAV con su respectivo pitot sin problemas.

Realizaremos los cálculos de la función de corrección. La operación de cálculo nos permite seleccionar un grado de precisión para la línea de tendencia. Dicha línea de tendencia está definida por un polinomio cuyo grado es la variable a elegir para aumentar la precisión.

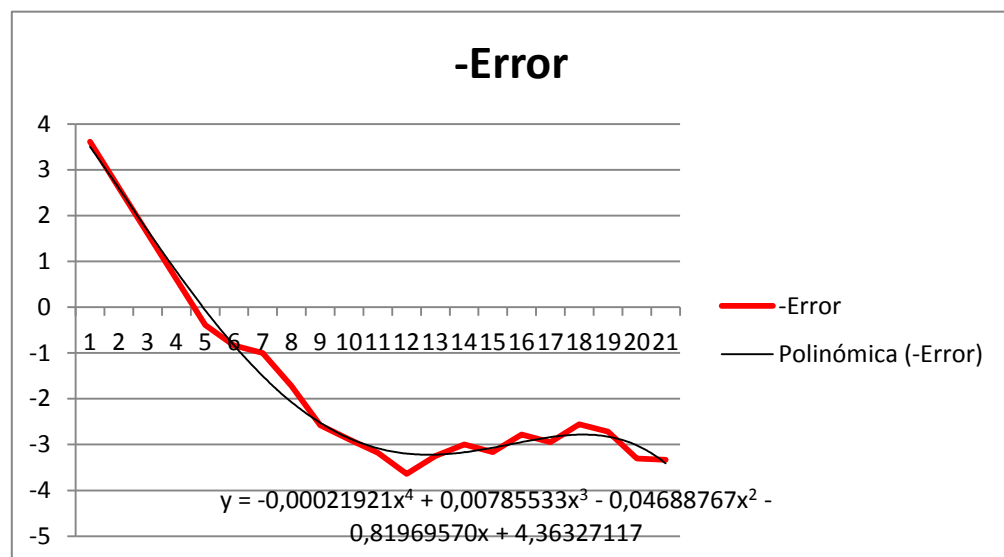




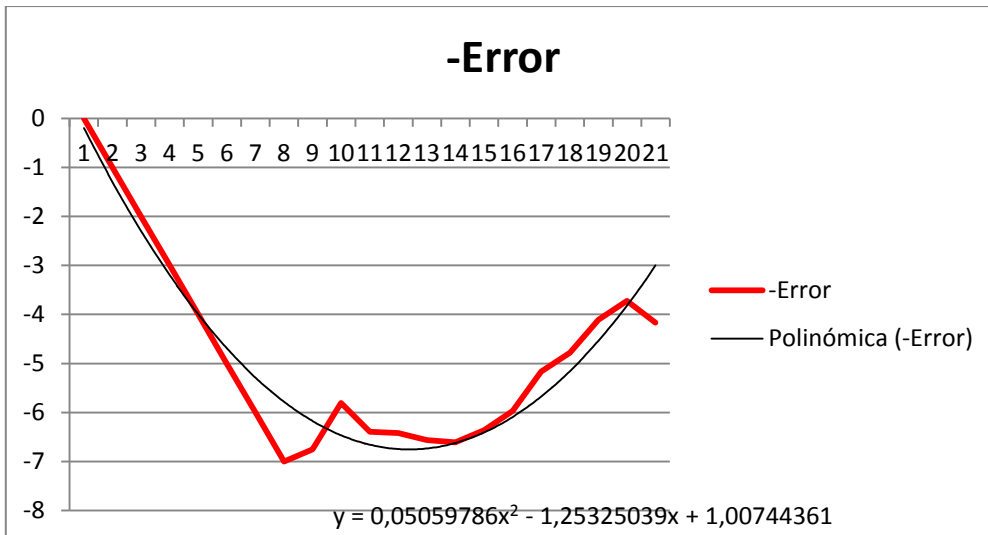
*Ilustración 4.15: Polinomio de error de grado 2 del pitot 1*



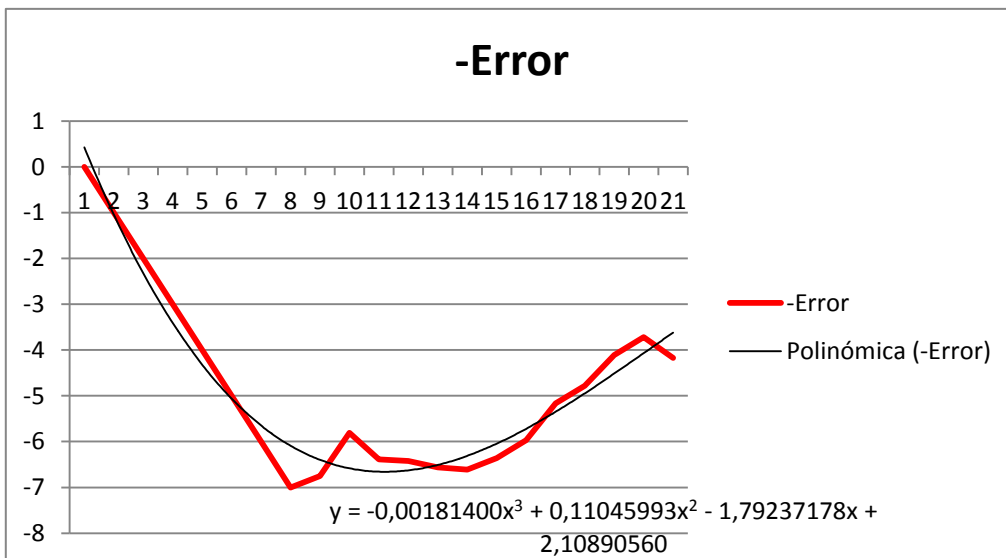
*Ilustración 4.16: Polinomio de error de grado 3 del pitot 1*



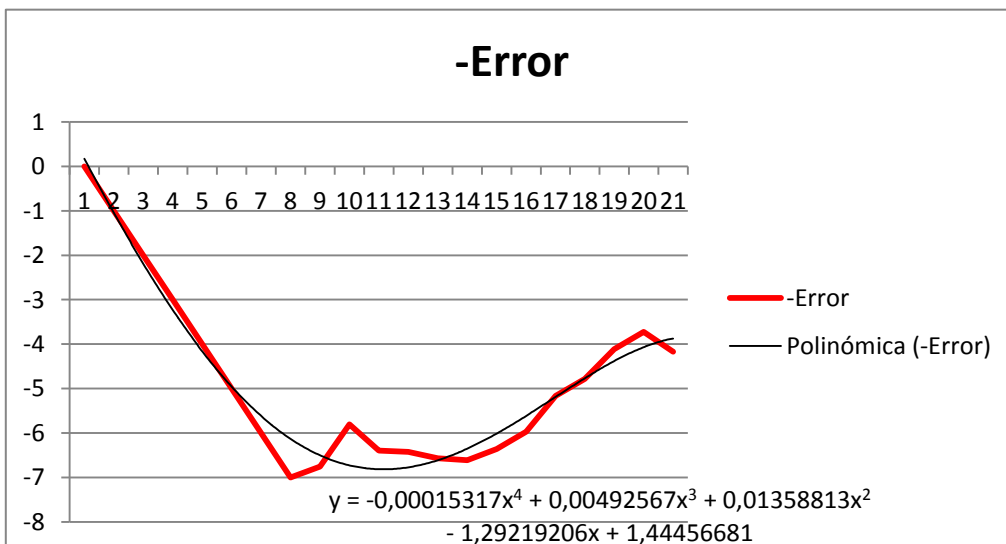
*Ilustración 4.17: Polinomio de error de grado 4 del pitot 1*



*Ilustración 4.18: Polinomio de error de grado 2 del pitot 2*



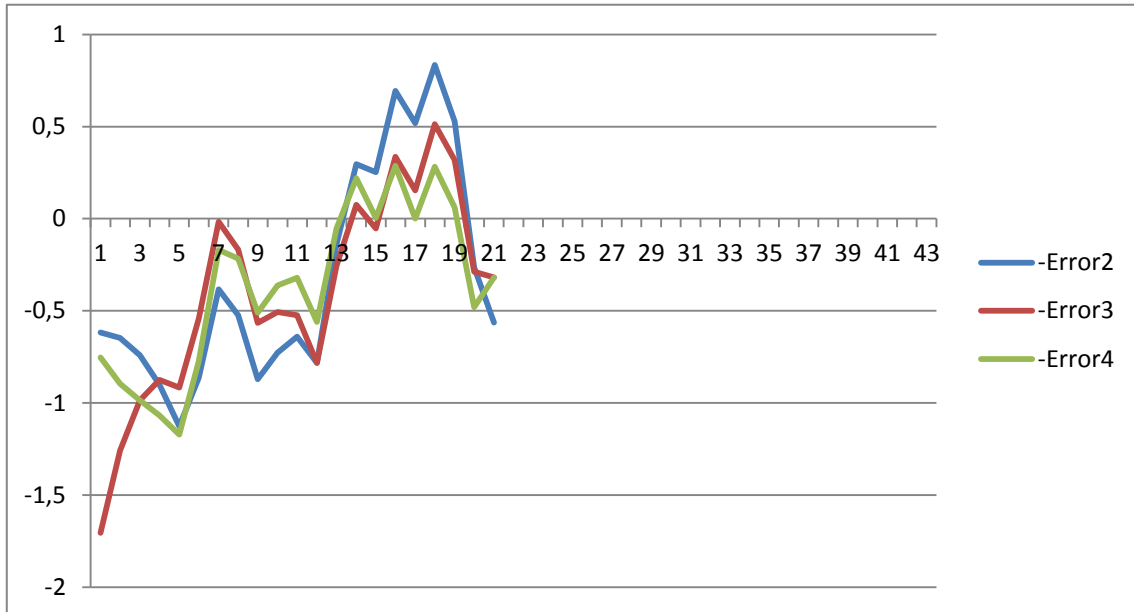
*Ilustración 4.19: Polinomio de error de grado 3 del pitot 2*



*Ilustración 4.20: Polinomio de error de grado 4 del pitot 2*

El programa permite realizar polinomios para la línea de tendencia con grados desde 1 hasta 6. No consideraremos utilizar polinomios de grado superior a 4, puesto que su aportación a la mejora del sistema es despreciable.

Una vez realizadas todas las mediciones y ajustes a los pitots obtenemos las gráficas de error:



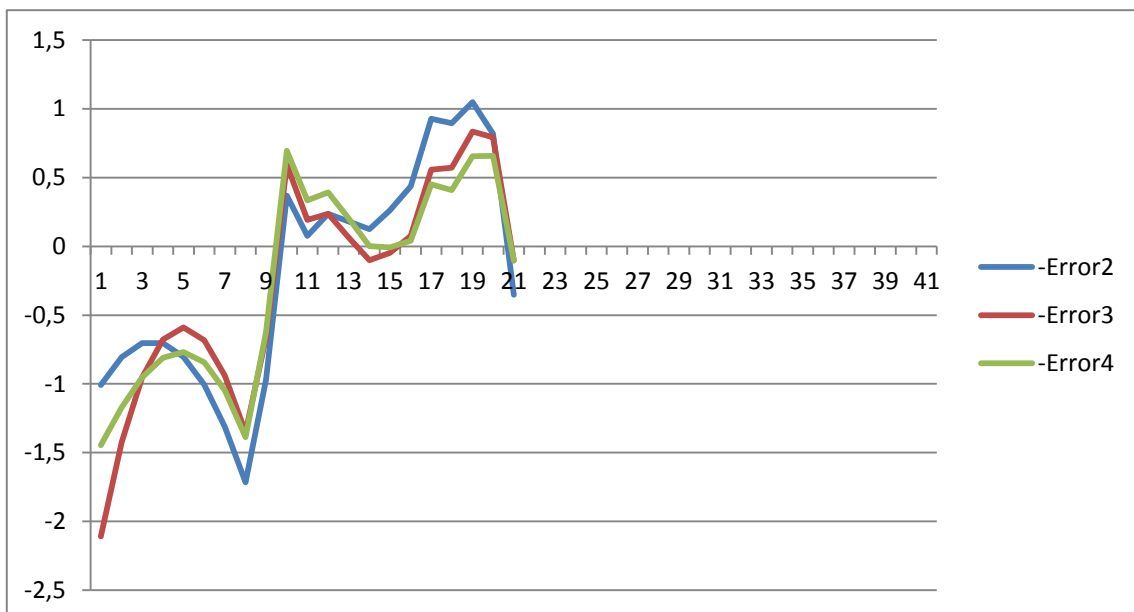
*Ilustración 4.21: Corrección con cada polinomio del error del pitot 1*

Pitot 1

Grado 2:  $y = 0.03268047x^2 - 1.00355467x + 4.2271954$

Grado 3:  $y = -0.00178989x^3 + 0.09174670x^2 - 1.53550875x + 5.31401420$

Grado 4:  $y = -0.00021921x^4 + 0.00785533x^3 - 0.04688767x^2 - 0.8196957x + 4.3632711$



*Ilustración 4.22: Corrección con cada polinomio del error del pitot 2*

## Pitot 2

Grado 2:  $y = 0.05059786x^2 - 1.25325039x + 1.00744361$

Grado 3:  $y = -0.001814x^3 + 0.11045993x^2 - 1.79237178x + 2.1089056$

Grado 4:  $y = -0.00015317x^4 + 0.00492567x^3 + 0.01358813x^2 - 1.29219206x + 1.44456681$

Cada línea se corresponde con un tipo de corrección de error con un polinomio de grado distinto: 2º, 3º y 4º grado, para cada pitot. Esto da para un total de tres alternativas posibles de corrección para cada sensor.

No se ha considerado una corrección con polinomio de grado 1, pues una línea recta no proporciona una mejora en el error sustancial en comparación con las otras opciones, o en comparación con la respuesta sin corregir.

Tampoco se ha considerado para el estudio una corrección empleando polinomios de grado superior a cuatro, puesto que la mejora que ofrecen, si bien es mayor que la de los polinomios a estudiar, no es tan grande como para que sea necesario considerarla.

Esto es debido principalmente a que el funcionamiento de los pitots proporciona una resolución de aproximadamente 0,3m/s, y con polinomios de corrección de grado 3 y 4 ya estamos en este límite. Aumentar el grado de los polinomios tan solo mejora la gráfica y los cálculos en el ordenador, sin transmitir esta mejora a la parte física del sistema, por lo cual se considera irrelevante para el caso. También se desea ajustar las mediciones empleando polinomios de corrección del menor grado necesario, para reducir la carga de cálculo en los procesadores.

Llegados a este punto debemos decidir entre las tres posibles opciones para cada pitot. Comenzaremos por el pitot 1.

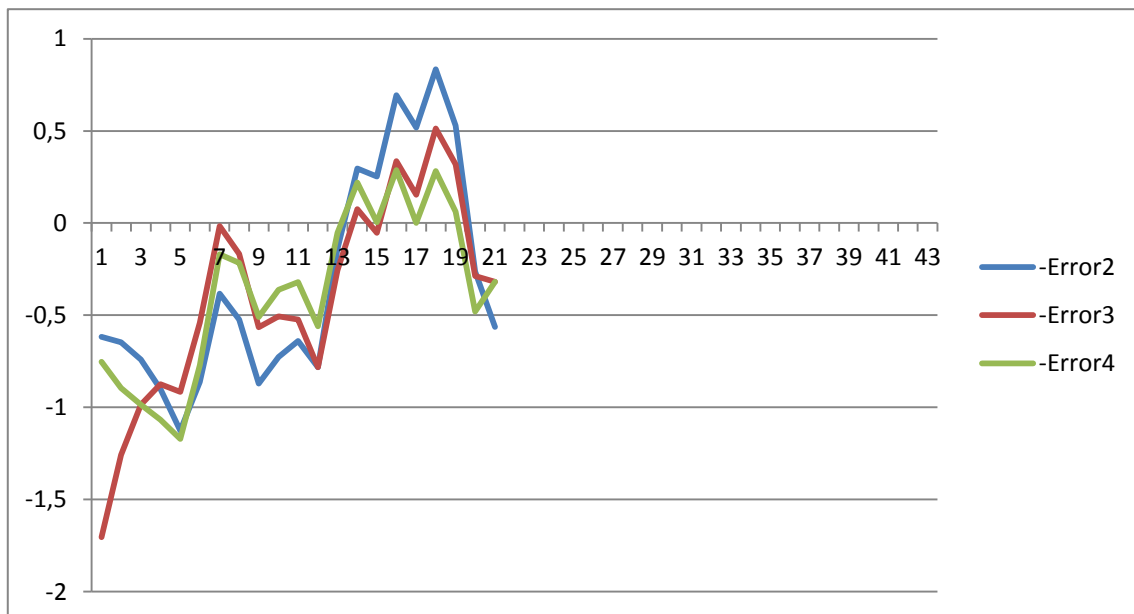


Ilustración 4.23: Comportamiento de los errores del pitot 1

El rango de medidas de interés abarca desde los 10 m/s hasta los 20 m/s, que es el rango en el que se mueve el dron que lleva el pitot. Deseamos que el error sea mínimo dentro de este rango, pudiendo obviar las respuestas y el error en medidas menores a 10 m/s.

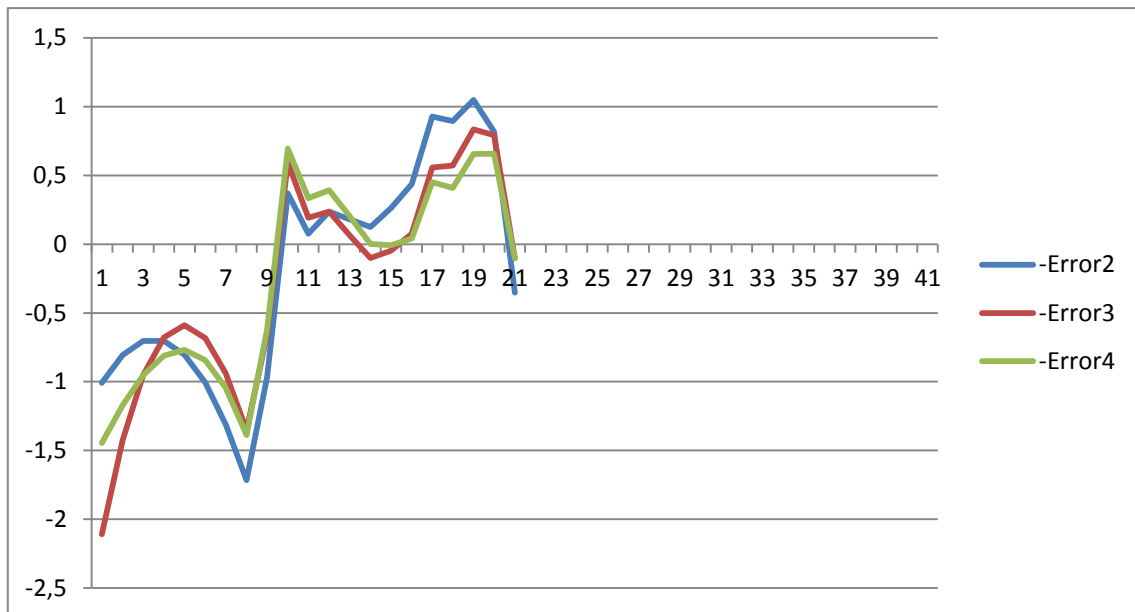
Teniendo en cuenta también que deseamos un margen de error de +/-0,5m/s, nunca superior a +1m/s, podemos descartar el polinomio de corrección de grado 2, que si bien proporciona un error bajo, no llega a cumplir con la suficiente holgura las condiciones expresadas anteriormente.

De entre los polinomios de grado 3 y 4 el comportamiento es bastante similar. Ambos cumplen con el objetivo de minimizar el error dentro del margen de interés. El error del polinomio de grado 3 es máximo en el valor 0, pero como el margen de interés comienza en 10 se puede obviar.

En general el valor de error es menor para el polinomio de grado 4, pero tampoco por demasiado. Teniendo en cuenta las condiciones de precisión que limitan el error a  $\pm 0,5\text{m/s}$ , el polinomio de grado 3 se encuentra ligeramente fuera del límite debido al valor que adopta en el valor  $12\text{m/s}$  de entrada. También se acerca al límite en el valor  $18\text{m/s}$ . En comparación el polinomio de grado 4 se mantiene dentro de los márgenes, acercándose al límite solo en la medida de  $12\text{m/s}$ , y siendo, por tanto, la única opción que cumple con todas las condiciones.

Consideradas todas las circunstancias y opciones, elegimos un polinomio de grado 4 para corregir el error de calibración del pitot 1.

Para el pitot 2 la gráfica es la siguiente:



*Ilustración 4.24: Comportamiento de los errores del pitot 2*

Existe una diferencia sustancial entre ambas gráficas al principio, entre los valores  $7\text{m/s}$  y  $10\text{m/s}$ , donde existe un pico de error que supera la unidad. En el caso del polinomio de grado 2 el error es incluso mayor de  $1,5\text{m/s}$ . Salvo por este detalle los valores de error de la gráfica se parecen mucho a los del pitot 1, y se mantienen también en el mismo margen.

El pico de error existe debido a que el pitot 2, a diferencia del pitot 1 no comienza a detectar hasta que el valor de velocidad de viento alcanza los  $8\text{m/s}$ . Esto provoca una rampa creciente de error en la gráfica de valores detectados, que se arrastra a la de errores. Como consecuencia esta rampa supone una dificultad para aproximar fielmente con polinomios la gráfica, que cambia bruscamente de tendencia cuando el pitot comienza a leer a partir de los  $8\text{m/s}$ .

Nuevamente el polinomio con unos errores más elevados es el de grado 2, por lo cual lo descartaremos en favor de las otras dos alternativas.

Se puede comprobar que de nuevo los errores de los polinomios de grado 3 y 4 se comportan de manera muy similar. En zonas conflictivas, como el pico de error antes mencionado, el valor es idéntico, por lo que no puede ser usado como argumento para decidir uno sobre el otro. En general el polinomio de grado 4 obtiene mejores resultados, pero por muy poco, y de nuevo queda a nuestro juicio decidir si es suficiente como para justificar el grado del polinomio.

En este caso no existe una situación en la que una alternativa cumpla las condiciones de precisión impuestas y la otra no. Ambos polinomios superan el límite de  $\pm 0,5\text{m/s}$  de error en las medidas 10, 18, 19 y 20. Sin embargo, el polinomio de grado 3 tiene una medida más que supera el límite de precisión, en 17. Esto convierte al polinomio de grado 4 en el más preciso de los disponibles por defecto.

Concluimos que, al igual que en primer caso, la diferencia de precisión entre los grados 3 y 4 son muy pequeñas, pero que el polinomio de grado 4 cumple mejor las especificaciones de precisión, o se acerca más a las mismas en este caso. Debido a esto se elige como la opción óptima.

En ambas conclusiones se debe decidir entre una opción más precisa y otra con una matemática más sencilla. Hemos elegido dar más peso a la precisión en este caso, eligiendo los polinomios de grado 4, puesto que es la falta de precisión la que provocaba fallos notables en el funcionamiento del UAV. Si por el contrario se decidiese optar por la solución más sencilla matemáticamente, sacrificando algo de precisión, se escogerían los polinomios de grado 3 para ambos pitots.

Opción más precisa:

Polinomio de grado 4 para el pitot 1:

$$y = -0.00021921x^4 + 0.00785533x^3 - 0.04688767x^2 - 0.8196957x + 4.3632711$$

Polinomio de grado 4 para el pitot 2:

$$y = -0.00015317x^4 + 0.00492567x^3 + 0.01358813x^2 - 1.29219206x + 1.44456681$$

Opción más sencilla:

Polinomio de grado 3 para el pitot 1:

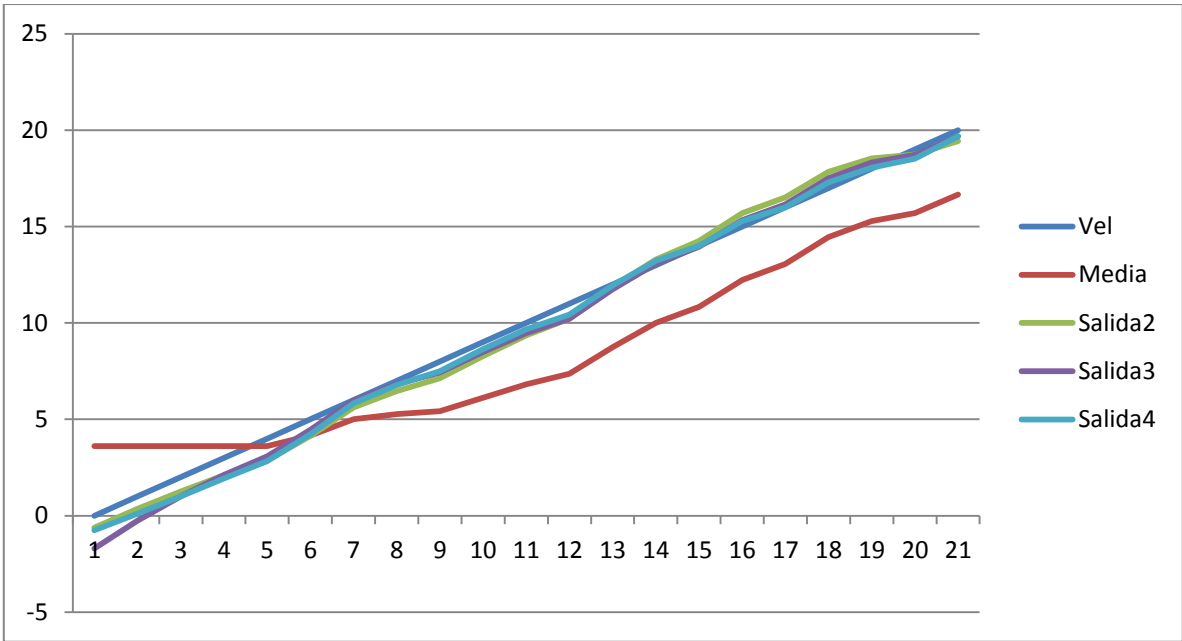
$$y = -0.00178989x^3 + 0.09174670x^2 - 1.53550875x + 5.31401420$$

Polinomio de grado 3 para el pitot 2:

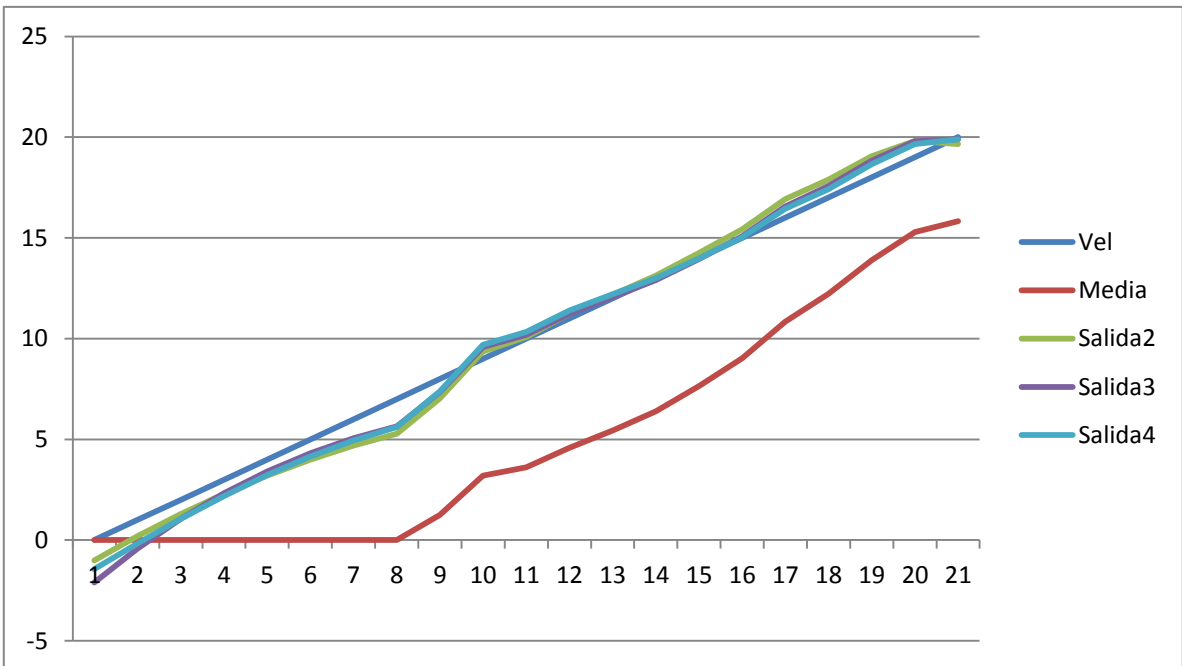
$$y = -0.001814x^3 + 0.11045993x^2 - 1.79237178x + 2.1089056$$

En última estancia hemos decidido optar por la solución más precisa, ya que no somete a los dispositivos ni a la programación a un esfuerzo significativo como para superar a los beneficios de la precisión que aporta el polinomio. Así, los UAV integrarán un polinomio de corrección de grado 4 en el controlador de los sensores de pitot.

Los resultados de aplicar los polinomios de corrección se pueden comprobar de forma sencilla. Se puede observar que reducen de manera muy significativa el error del sensor.



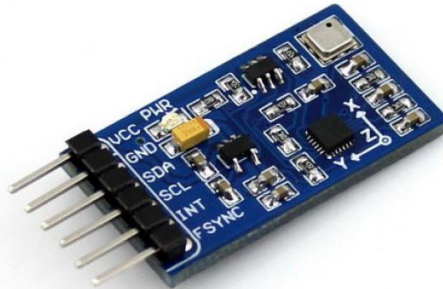
*Ilustración 4.25: Resultado de corrección del pitot 1 con cada polinomio*



*Ilustración 4.26: Resultado de corrección del pitot 2 con cada polinomio*

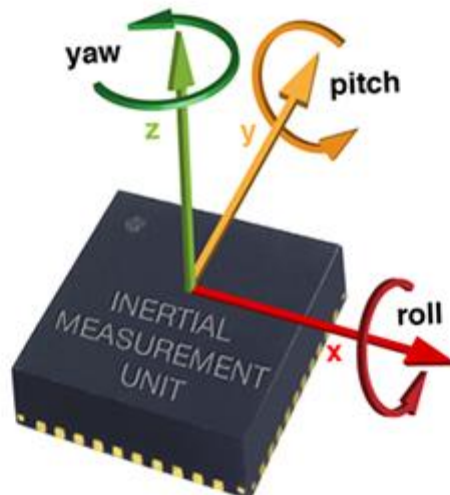
La corrección no dá un resultado perfecto para ninguno de los dos casos, pero los errores que quedan son mucho menores y supone una clara mejoría con respecto al comportamiento inicial.

## 4.1.2.IMU



*Ilustración 4.27: Sensor de inclinación IMU*

Un IMU, o Unidad de medición inercial (Inertial Measurement Unit) es un dispositivo que recoge la inclinación de la placa, en dos de los tres ejes dimensionales (roll y pitch). El control de la orientación es una parte fundamental del control de vuelo de un dron. La capacidad de trazar ángulos precisos y conocer en todo momento la posición de la horizontal del UAV con respecto al horizonte es necesaria para poder realizar cualquier maniobra en vuelo.

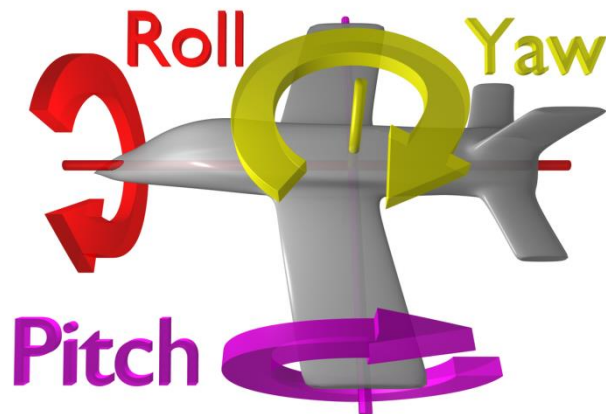


*Ilustración 4.28: Los tres ejes de giro del chip*

Adicionalmente la capacidad de monitorizar la posición del UAV permite conocer información de las corrientes de aire imprevistas que puedan apartar al aparato de su rumbo, como su dirección y fuerza, por lo que el IMU es una herramienta de navegación fundamental para el vuelo.



Un error en el funcionamiento del IMU puede provocar que el UAV realice vuelos inestables o alocados, o que no sea capaz de anunciar al controlador humano de una corriente inesperada, una pérdida de control o una entrada en barrena. Por este motivo es esencial conocer las condiciones de funcionamiento del IMU.



*Ilustración 4.29: Los tres ejes de giro de un avión*

Conociendo que la principal diferencia de la simulación de la investigación previa residía en la diferencia de velocidad, en principio no invita a pensar que un defecto en el IMU pueda ser la causa. No obstante se puede dar un caso en el que una inclinación defectuosa se combine con otros defectos de funcionamiento de otros sistemas y provoque el fallo. Por este motivo se continúa analizando el funcionamiento de todos los sistemas aún conociendo que se ha detectado un fallo en el sistema de sensores más probable, que era el de pitot.

El IMU integrado en el UAV mide la inclinación del plano del propio chip sin necesidad de componentes externos. Para medir la inclinación del UAV basta con fijar el sensor al modelo en una superficie horizontal. El componente electrónico empleado es un Eagle Tree System Guardian Stabilization. Su manual de instrucciones se incluye en el ANEXO VIII: Manual de instrucciones Guardian Stabilization.



*Ilustración 4.30: IMU del UAV*



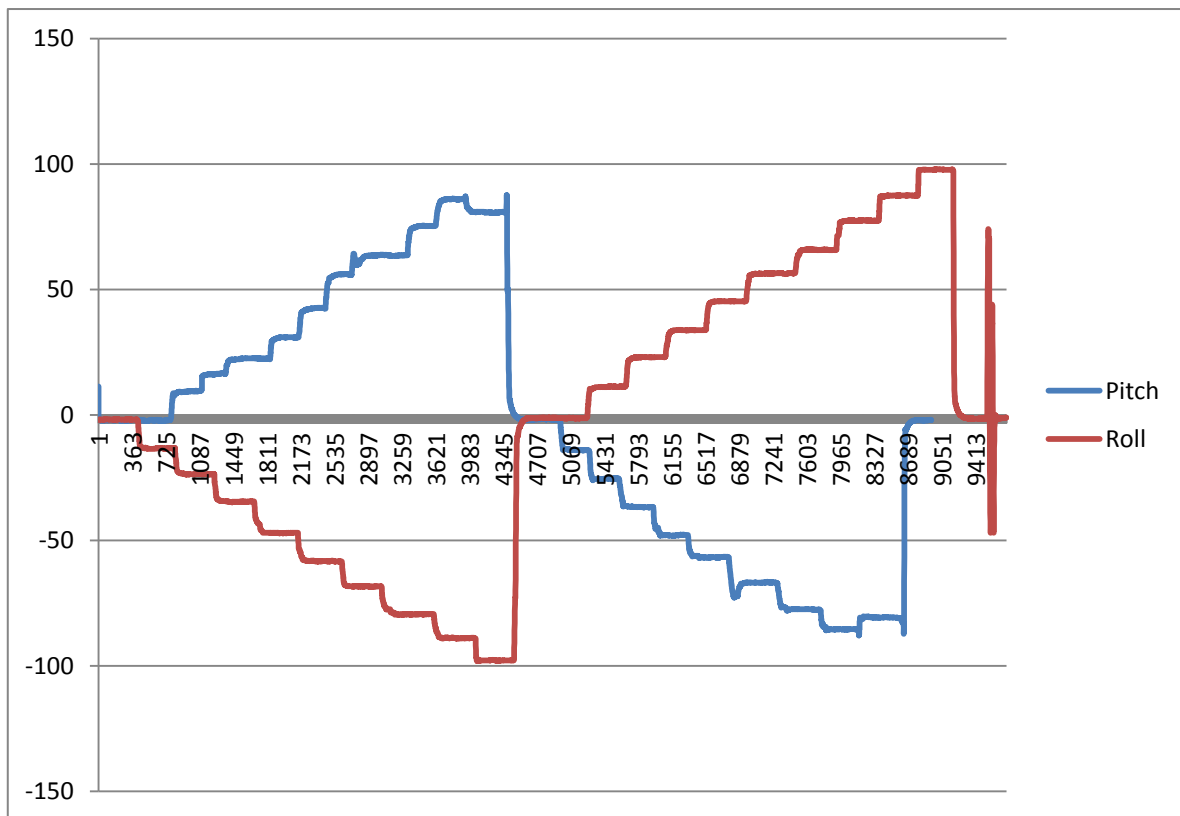
*Ilustración 4.31: Localización del IMU en el UAV*

Las pruebas de funcionamiento de un IMU son más sencillas que las de un pitot y requieren muchos menos medios para llevarlas a cabo. Puesto que lo único que se detecta es la inclinación del chip, basta con extraer el chip del UAV, adherirlo a una superficie móvil y posicionarlo en distintos ángulos de inclinación de valor conocido.



*Ilustración 4.32: Prueba de funcionamiento del IMU*

En nuestro caso la prueba se realizó con un trípode con cabezal móvil a la que le fue fijada el IMU, y la cual fue posicionada en distintos grados de inclinación, cada 10 grados, hasta cubrir un total de 180 grados en un plano y posteriormente en el otro. De esta manera se obtiene la siguiente gráfica:



*Ilustración 4.33: Resultados de la prueba del IMU*

La prueba de funcionamiento del IMU consiste en posicionar el dispositivo en una superficie horizontal, de forma que las lecturas de pitch y roll marquen cero. Posteriormente se procede a girar la superficie en saltos de  $10^{\circ}$  hasta alcanzar la vertical, en ambos sentidos y para ambas orientaciones.

La IMU proporciona solo dos grados de libertad en su funcionamiento, correspondientes a Pitch y Roll. No realiza lecturas de Yaw.

Los resultados de la prueba muestran un comportamiento preciso y consistente. Existe una ligera desviación del comportamiento para el valor del pitch para  $90^{\circ}$ , pero se considera que no entra dentro de un margen de utilización normal del UAV así que no se tiene en cuenta en la posible generación de defectos en el comportamiento (no se cuenta con llevar el dron completamente vertical con respecto al suelo en ningún momento, por lo que el posible error de medición nunca se daría).

En general el error de medición del dispositivo IMU es el siguiente:

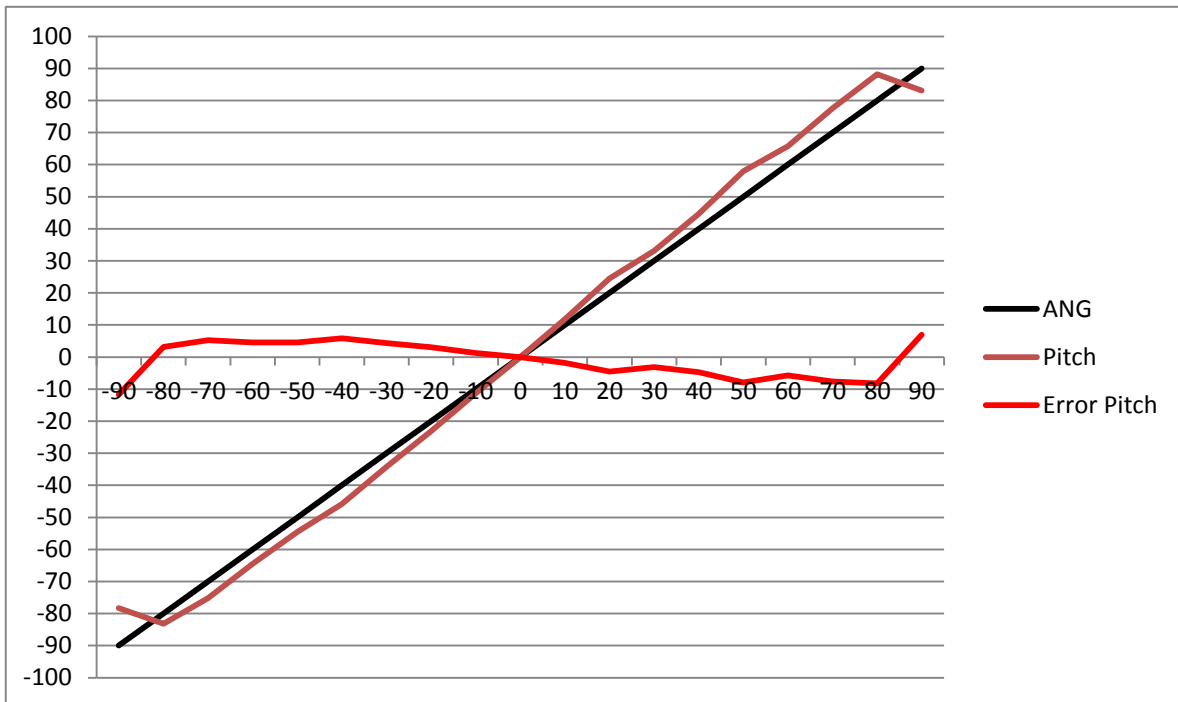


Ilustración 4.34: Error de medición de pitch

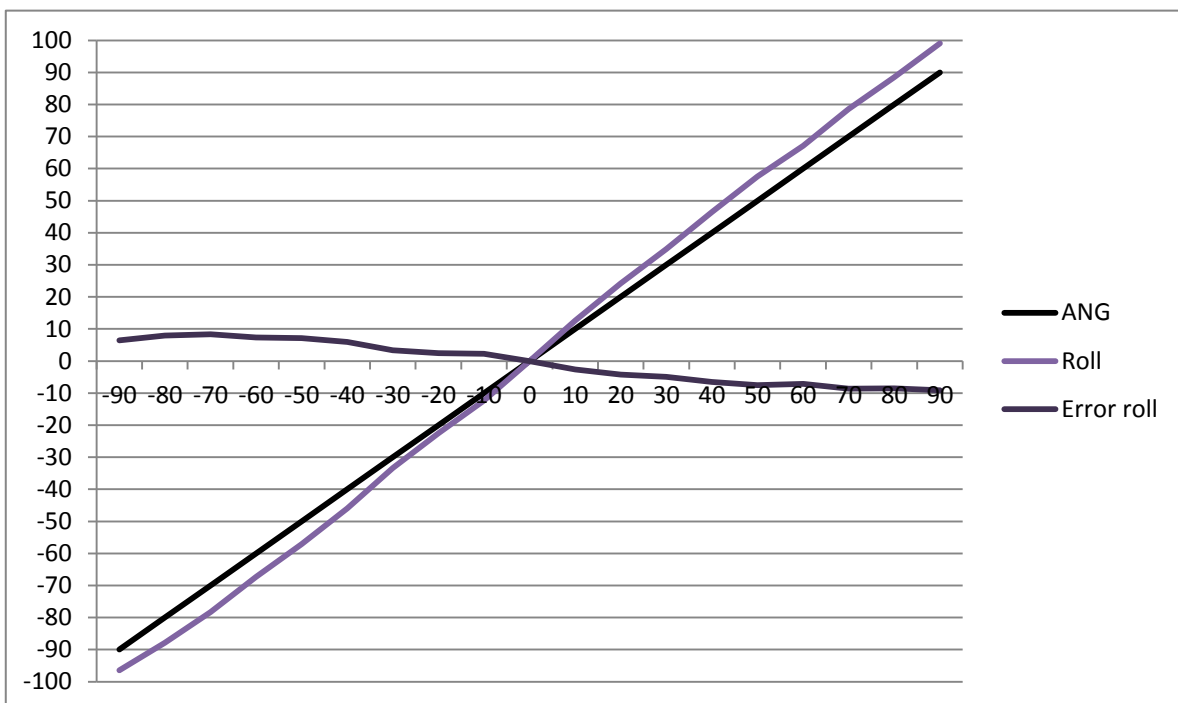


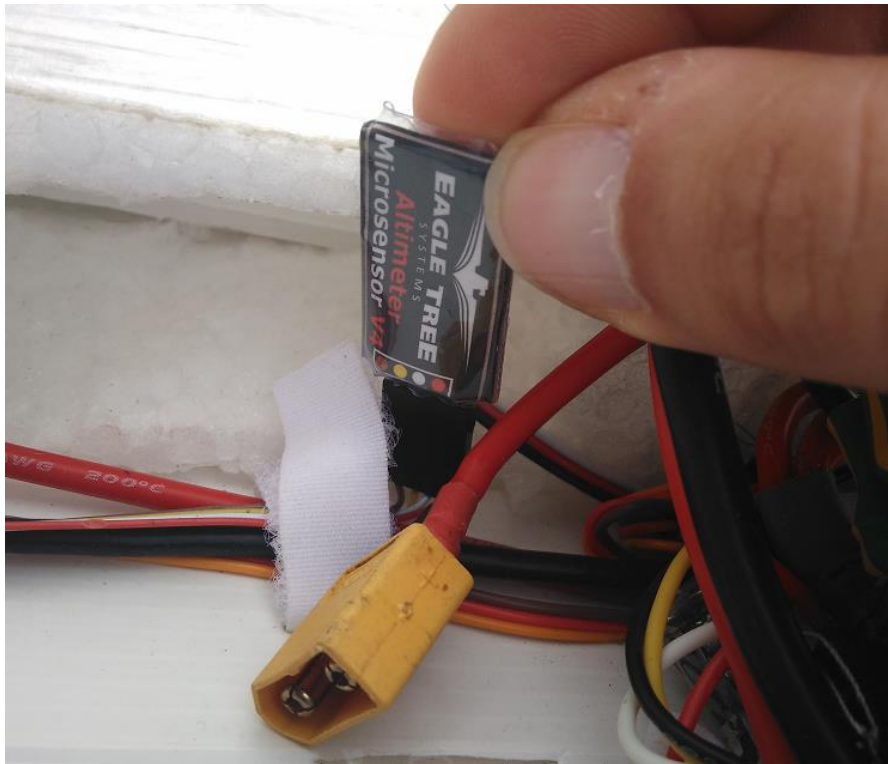
Ilustración 4.35: Error de medición de roll

Se puede apreciar que presentan un comportamiento bastante preciso y consistente. Idealmente todos los sensores del UAV podrían tener un comportamiento semejante a éste, con un error que nunca supera un 10%, lo cual es asumible y no provoca diferencias importantes contra la simulación.

Podemos concluir que el sensor IMU funciona correctamente y no necesita de un ajuste adicional. También que no es un responsable del comportamiento anómalo del UAV con respecto a su simulación.

### 4.1.3. GPS y altímetro

El dispositivo de detección de altura del UAV es conjunto de dos subsistemas: un GPS y un altímetro Eagle Tree System Altimeter Microsensor V4. El manual de instrucciones de éste último está en el ANEXO IX: Manual de instrucciones Altimeter Microsensor. Se trata de un sistema de dos sensores de funcionamientos paralelos.



*Ilustración 4.36: Altímetro del UAV*

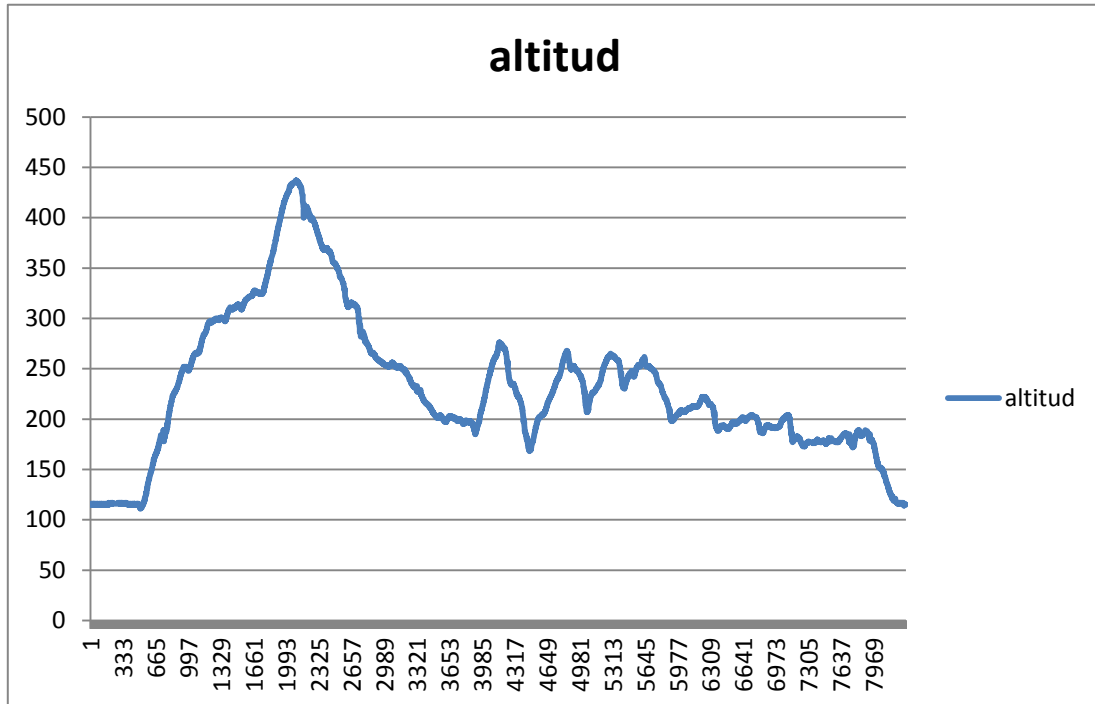
El altímetro mide la presión atmosférica y obtiene una altura absoluta, mientras que el GPS utiliza datación por satélite para obtener la altura en relación al lugar. Esto crea una diferencia de lecturas intencionada, pues permite obtener dos lecturas diferentes y, teóricamente ofrecería la posibilidad de detectar un error de medición si no se correspondiese con la lectura del GPS, una vez salvada esa discrepancia esperada. Para efectos de éste proyecto se considerará que solo el altímetro es susceptible de defectos al tratarse de un sensor de fenómenos físicos.

El GPS determina la posición del UAV mediante el método de localización por triangulación utilizando como referencias como mínimo los tres satélites más cercanos. Debido a la gran cantidad de satélites disponibles para el funcionamiento de este sistema, el GPS cuenta con una buena precisión y puede considerarse infalible a efectos de este proyecto.

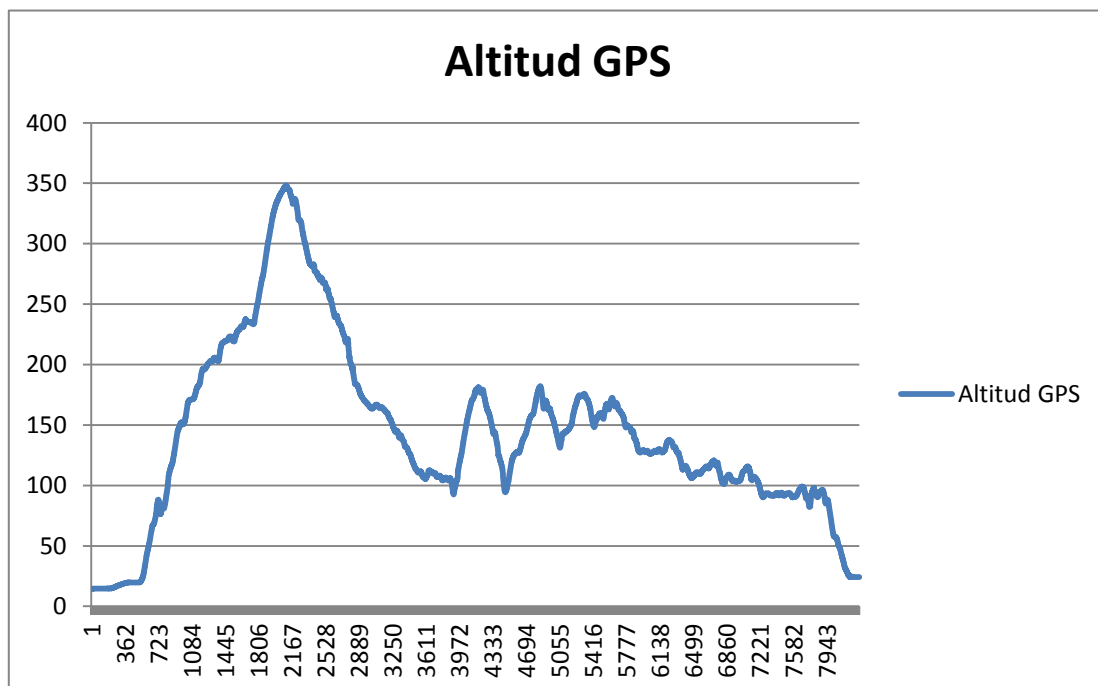
Para realizar una prueba de funcionalidad del altímetro es necesario realizar una prueba de vuelo, para comprobar su trabajo en tiempo real. Esto es así porque no resulta práctico intentar realizar una prueba de altura de otra forma, ya que para alcanzar a estudiar el comportamiento del altímetro dentro de todo el rango de alturas que utiliza, sería necesario contar con algún medio para llegar a 300-400 metros de altitud. La prueba de funcionamiento del altímetro se realizó con los datos de vuelo del vuelo final del UAV, una vez realizadas las correcciones de medición del pitot. Dicha prueba se consideró válida para la comparación con el modelo simulado, explicado más adelante, puesto que se comprobó que el altímetro funciona correctamente y no requiere ajustes (véase más adelante en este mismo punto).

El funcionamiento correcto del altímetro debe dar una lectura muy similar a la del GPS, pero con una diferencia de valores en cada medida. Ésta diferencia de valores debe ser aproximadamente constante, ya que es la diferencia que el GPS obtiene de considerar la presión atmosférica del nivel del mar.

Durante el vuelo realizado los dos sensores proporcionaron los siguientes datos:

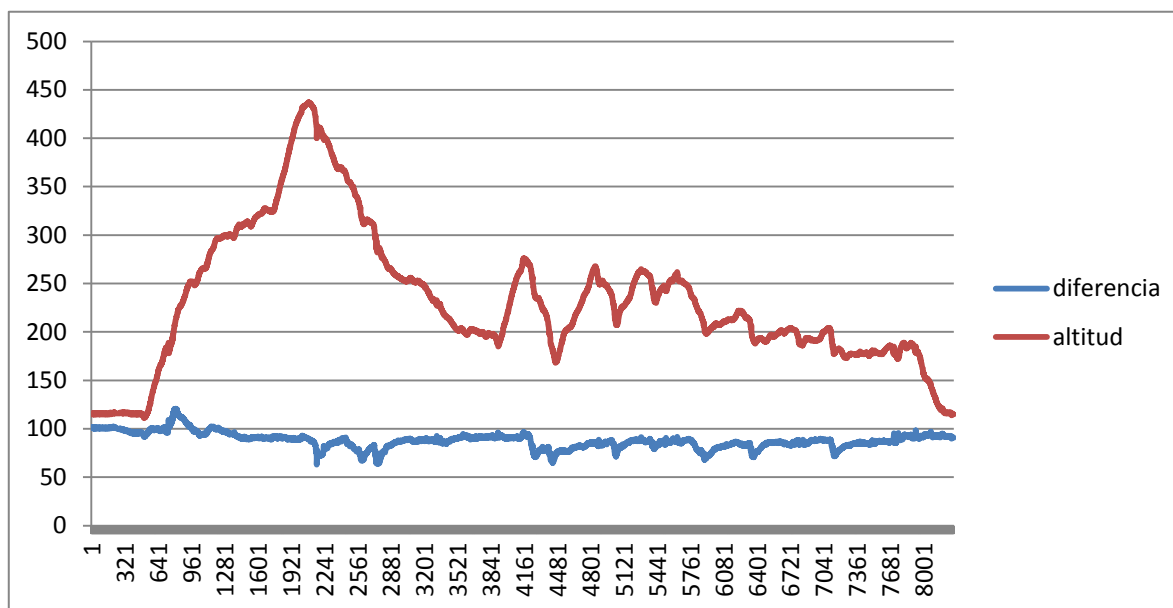


*Ilustración 4.37: Prueba de medición de altura con altímetro*



*Ilustración 4.38: Prueba de medición de altura con GPS*

Si el altímetro funciona correctamente, la diferencia de medida con el GPS debe ser de un valor más o menos constante. Por las gráficas se puede observar que las lecturas obtenidas tienen unas curvas muy similares, lo cual es un buen indicativo del correcto funcionamiento del sistema. Para estar seguros obtendremos la curva de la diferencia de valores mediante una simple resta.



*Ilustración 4.39: Diferencia constante de medidas*

La diferencia es aproximadamente constante. No traza una línea perfecta y es un resultado esperable. La presión atmosférica fluctúa constantemente y la medición puede verse afectada por rachas de viento, corrientes térmicas y la posición y velocidad que tiene el propio dron, por lo que no podemos esperar una medición perfecta. Además se debe tener en cuenta también el ligero retraso que introduce el sistema GPS en el cálculo de su localización, y que hace que las medidas de ambos sensores no capten la información exactamente al mismo tiempo. Sin embargo la tendencia de las medidas marca claramente un valor que fluctúa entre 90 y 100, un margen de 10 m, suficientemente pequeño como para considerarlo preciso.

De los ensayos realizados concluimos que el sensor de altitud no añade errores inadmisibles al funcionamiento del UAV, y por tanto no requiere un ajuste adicional para mejorar su rendimiento.

## 5. REVISIÓN: APARTADO VIRTUAL

Hemos completado la revisión e investigación de todos los posibles orígenes del comportamiento erróneo del UAV. Encontramos un posible responsable en el sensor de velocidad, y confirmamos el buen comportamiento de los sensores de inclinación y altitud. Ahora que conocemos todo el estado del UAV físico, debemos preguntarnos si el motivo de que el vuelo físico y el simulado no se correspondieran pudiera ser debido también a algún posible error en la construcción del modelo virtual.

Para asegurarnos del buen diseño del modelo virtual revisaremos todo el montaje. Esto incluye el estudio de la aerodinámica del UAV y de cada una de sus partes, y desarrollo de sus tablas de datos.

### 5.1. *Simulador de vuelo*

El software de simulación de vuelo seleccionado es FlightGear. Es un simulador de vuelo de código abierto desarrollado en 1997 para contribuir al aprendizaje académico, el entrenamiento de pilotos y como herramienta aplicada a la ingeniería. Su naturaleza de código abierto y acceso libre para cualquier persona es una de las características que se necesitaban para el trabajo.

FlightGear está escrito principalmente en C++. Además, se tiene constancia de que funciona con diversos sistemas operativos como Windows, Linux, Mac OS-X, Solaris o IRIX, permitiendo a los usuarios usar la plataforma que cada uno prefiera.

FlightGear es solamente el software que ofrece la interfaz gráfica de la simulación. Para que la aeronave pueda volar siguiendo los principios de la física es necesario un Modelo de Dinámica de Vuelo.

### 5.2. *Flight Dynamics Models (FDM)*

Un FDM (en español, Modelo de Dinámica de Vuelo) es un modelo matemático-físico que define el movimiento y comportamiento de la aeronave bajo las fuerzas y momentos aplicados sobre ella y fuerzas naturales usando varios mecanismos de control.

Para FlightGear, se consideraron cinco FDM compatibles:

- LARCSim
- UIUC
- YASim
- JSBSim
- MATLAB

De todas las opciones se optó por JSBSim, por considerarlo como el mejor software en prestaciones en comparación al resto, y de código libre.

### 5.3. *Modelado del UAV*

El modelo se divide en las siguientes partes:

- Hélices
- Motores
- Aerodinámica
- Tren de aterrizaje e inercias



### 5.3.1.Hélices

La hélice utilizada para la toma de medidas fue el modelo de hélice estándar para todos los motores de la flota de UAVs. Se empleó el software de diseño JavaProp.



*Ilustración 5.1: Hélice de pruebas*

Las características de la hélice para introducir en el programa son las siguientes:

- Número de aspas: 2
- Peso: 18g.
- Longitud: 9 pulgadas (0,2286 m)
- Paso: 6 pulgadas (el UAV avanza 0,1524 m por vuelta de la hélice a RPM máxima, de forma ideal)
- Cuerda: Se introdujo la siguiente tabla, cuyas medidas se tomaron mediante un calibre, obteniendo los siguientes resultados:

Posición (cm)	Posición (r/R)	Cuerda (mm)
1	0,0875	16,4
2	0,175	18,5
3	0,2625	23,3
4	0,35	25,6
5	0,4374	25,5
6	0,5249	24
7	0,6124	21,8
8	0,6999	18,8
9	0,3937	15,4
10	0,8749	12
11	0,9624	9

*Tabla 5.1: Cuerda en cada sección de la hélice*

La “r” es la posición en donde tomamos la medida y “R” el radio de la hélice (11,43 cm).

- Velocidad máxima de rotación: 10.000 rpm. Medida directamente, una vez montada sobre el UAV, con un tacómetro.

- Ángulo de hélice:

Posición (mm)	Cuerda (mm)	Angulo de ataque (º)
10	16,4	32
20	18,5	43
24	21,6	47
30	23,3	35
40	25,6	28
50	25,5	24
60	24	20
70	21,8	18
80	18,8	15
90	15,4	12
100	12	10
110	9	7
113	1	5

Tabla 5.2: Ángulo de ataque en cada sección de la hélice

- Velocidad de vuelo del UAV: Esta medida fue tomada en pruebas de vuelo previas, obteniendo una velocidad media de vuelo de 12,5 m/s (ignorando picos provocados por rachas).

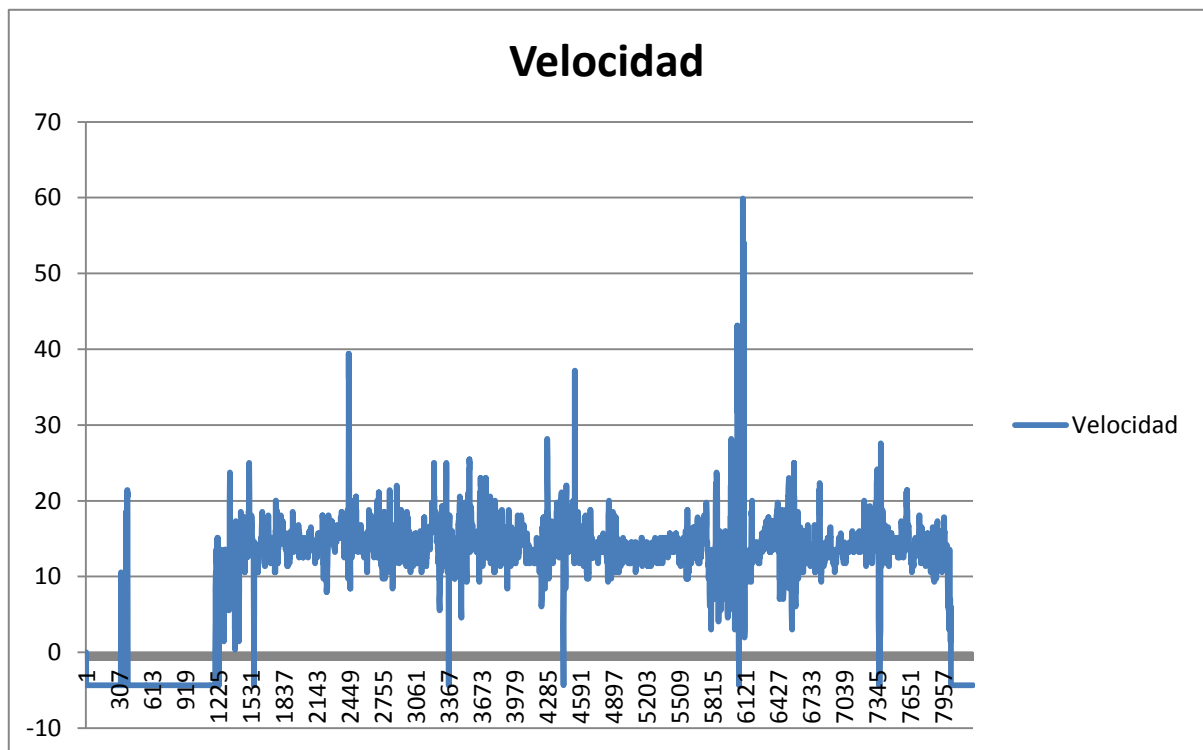


Ilustración 5.2: Velocidad de vuelo media del UAV

- Diámetro del "spinner": 20 mm.
- Potencia del motor: 345 W. Dato del fabricante.

- Perfiles: Se han determinado teniendo en cuenta el espesor de la pala en cada uno de las 4 secciones que solicita y la forma geométrica, utilizando los valores ya existentes y tabulados, para varios tipos de hélices:

Posición (r/R)	Perfil
0	MH 126, Re=500000
0,333	MH 114 13%, Re=500000
0,667	Clark Y, Re=100000
1	MH 116 9,8%, Re=500000

Tabla 5.3: Perfil de cada sección

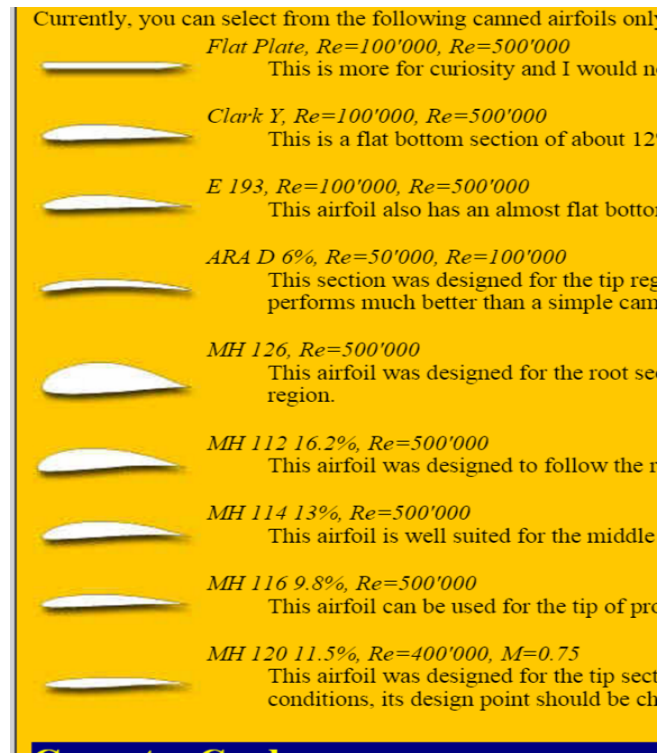


Ilustración 5.3: Perfiles disponibles

- Momento de inercia ( $I_{xx}$ ):

$$I_{xx} = m \cdot k^2$$

m: masa de la hélice

k: radio de giro (como area)

Obteniendo una inercia de:

$$I_{xx} = 0,018 \cdot 0,1143^2 = 0,00023516 \text{ kg} \cdot \text{m}^2$$

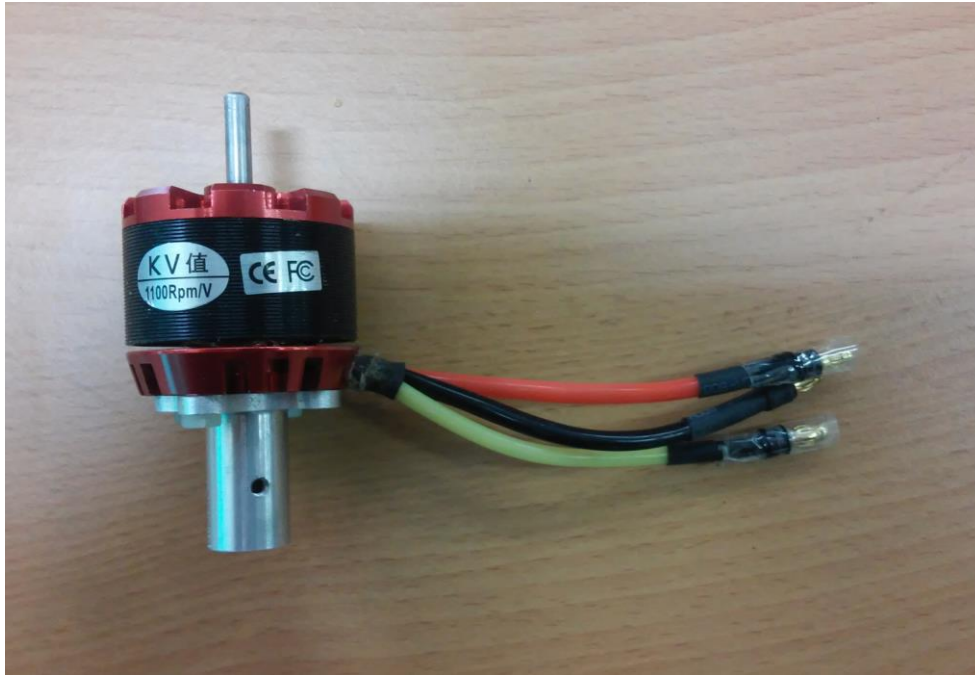
- P-factor: valor 0.

Estos dos últimos factores quedan contrarrestados y prácticamente anulados por la existencia de las dos hélices del UAV, que giran contrapuestas y anulan los efectos

El archivo resultante se puede consultar en el ANEXO I: Fichero de hélices.

### 5.3.2.Motores

En la imagen siguiente se muestra el tipo de motor que se emplea en la propulsión. Se emplean dos unidades. Los datos que sea necesario conocer de los motores los suministra el fabricante, por lo que se consideró que no era necesario realizar experimentos para obtener medidas.



*Ilustración 5.4: Motor del UAV*

JSBSim solo solicita el nombre y la potencia del motor, en el caso de motores eléctricos como éste, por lo que ésta es la parte más fácil de modelar, y el archivo .XML es el más sencillo de los cuatro. Los datos introducidos en el archivo son:

- Modelo: "n3530".
- Potencia: 345 W.

Resaltar que FlightGear trabaja con un modelo de motor basado en la potencia que se le especifica. La regulación de la potencia entregada por el motor se hace desde 0 hasta el valor especificado y varía de forma lineal. Internamente, se convierte esta potencia en caballos de fuerza que son aplicados sobre el archivo de las hélices para proporcionar un par y hacer que se muevan.

No considera la existencia de baterías, por lo que no tienen en cuenta que consuman ninguna energía. Tampoco se considera la fricción interna del motor.

El archivo resultante se puede consultar en el ANEXO II: Fichero de motores.

### 5.3.3. Aerodinámica

El modelo de UAV que fue diseñado con anterioridad fue el siguiente:



*Ilustración 5.5: UAV*

En éste trabajo se ha experimentado y trabajado con uno de los otros modelos idénticos. Aparece en la siguiente imagen marcado con el número 9.



*Ilustración 5.6: Flota de UAVs*

Si bien se supone que el perfil, rendimiento, funcionalidad y diseño son idénticos y no debería tener ningún impacto en el desarrollo de un modelado virtual, es necesario hacer hincapié en este dato. Resulta inevitable que la superficie de ambos modelos de UAV sea ligeramente diferente debido a ligeras imperfecciones naturales (o provocadas por caídas en pruebas de vuelo, u originadas en su manipulación, transporte, reparación, etc.).

Se considerará que estas diferencias son despreciables a efectos de diseño virtual, donde se pasarán por alto detalles tan pequeños de la misma manera que se hizo en la primera versión de los modelos de simulación.

### 5.3.3.1. Software para el análisis aerodinámico

De igual manera que se hizo con las hélices, se hizo una selección de software y otras opciones experimentales para obtener información del UAV. Se consideraron ventajas, inconvenientes, y limitaciones, teniendo en cuenta también la necesidad de ser software de código abierto.

Se consideraron las siguientes opciones:

- DATCOM
- HolyCows
- AVL
- Túnel de viento

Se seleccionó el **Athena Vortex Lattice (AVL)** para introducir un archivo que contenga los datos de geometría del UAV para obtener toda la información aerodinámica necesaria para llevar a cabo la simulación obteniendo los resultados más realistas posibles.

### 5.3.3.2. Características del UAV

En la siguiente tabla se pueden ver algunas características geométricas necesarias para la creación del modelo geométrico del UAV en AVL.

Área de referencia (m <sup>2</sup> )	1,1534
Cuerda aerodinámica media (m)	0,431
Envergadura de referencia (m)	1,94
Espesor máximo en la zona central (m)	0,059
Ángulo de dihedro (°)	2
Ángulo de torsión (°)	2
Ángulo de flecha (°)	25

*Tabla 5.4: Principales características geométricas del UAV*

El área de referencia hace referencia a la superficie de sustentación incluida la superficie aportada por los elevones, al igual que la cuerda aerodinámica media.

### 5.3.3.3. Modelado geométrico del UAV en AVL

Se introdujo los datos de 18 secciones de la parte alar con ángulo de flecha, y otras 4 secciones de la parte alar que no tiene flecha. Con eso se consideró una superficie suficientemente detallada, ya que la raíz del ala y la punta las unen una línea recta, sin curvatura visto desde arriba, al igual que en la parte central del ala, que tampoco tiene curvatura visto desde arriba.

Para definir las superficies del ala es necesario detallar la siguiente lista de parámetros:

- **Número de Mach:**

$$Mach = V_{UAV}/V_{sonido} = 12,5/343 = 0,036$$

- **Planos de simetría aerodinámica** (*l<sub>ysym</sub> l<sub>zsym</sub> z<sub>sym</sub>*)
- **Área de referencia** (*S<sub>ref</sub>*): área alar de la vista en planta del UAV, vale 1,1534 m.
- **Cuerda Aerodinámica Media** (*C<sub>ref</sub>*): cuerda media de las alas, vale 0,4306 m.
- **Envergadura** (*B<sub>ref</sub>*): 1,94 m.
- **Centro de masas:** Considerando el origen de coordenadas en la punta del UAV:

$$X_{ref} = 0,254 \text{ m} \quad Y_{ref} = 0,0 \text{ m} \quad Z_{ref} = 0,0295 \text{ m}$$

Se crean las siguientes superficies: "CENTRAL WING", que es la superficie para el ala central; "WING" para el ala con ángulo de flecha; y "RUDDER" para las verticales.

El siguiente paso es perfeccionar el modelo añadiendo los winglets y el fuselaje.

Los winglets se pueden crear siguiendo los mismos pasos que en la creación de las alas, al ser también superficies. Se les asigna el nombre "WINGLET".

El fuselaje original es un volumen irregular sin una coincidencia directa con una forma conocida o fácilmente trasladable a ecuaciones comprensibles para el programa. Tuvo que ser representado como un cuerpo de revolución con las mismas superficies lateral y frontal, debido a limitaciones de AVL con cuerpos con volumen.

La representación final del UAV es la siguiente:

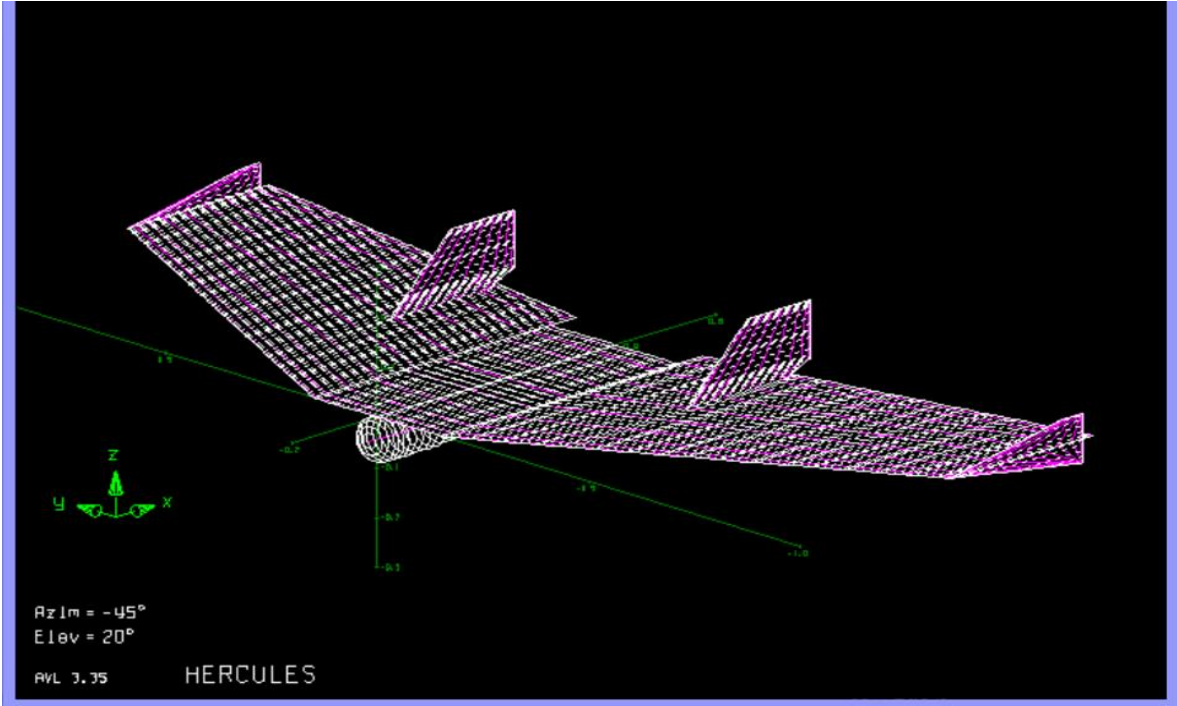


Ilustración 5.7: UAV completo en AVL

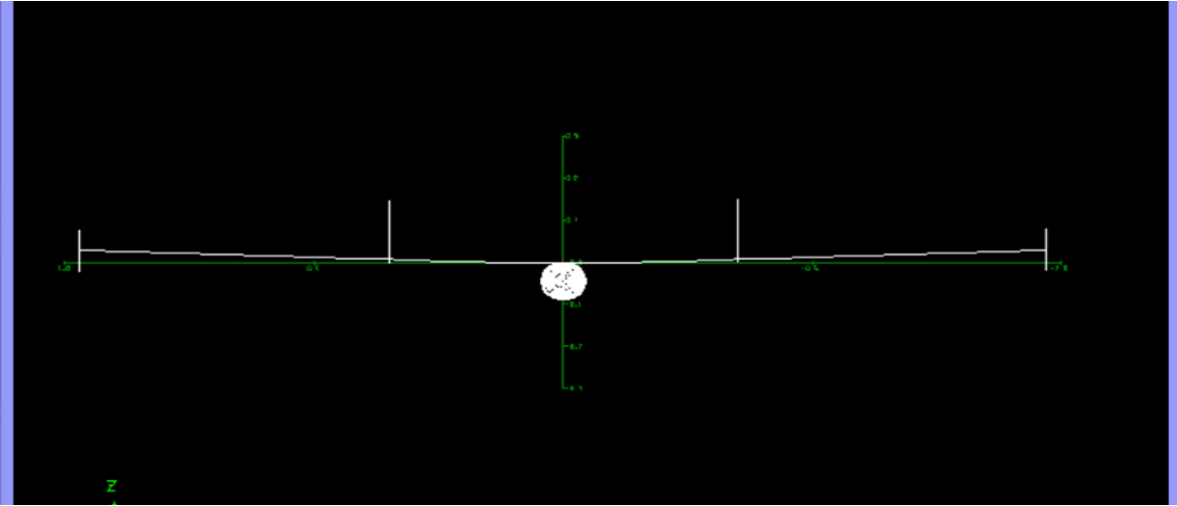


Ilustración 5.8: Vista frontal

El archivo resultante se puede ver en el ANEXO V: hercules.avl.



### 5.3.3.4. Cálculo de los coeficientes aerodinámicos

Una vez calculado el modelo geométrico lo más similar posible hay que programar unas condiciones de vuelo constantes (Match, velocidad de vuelo, densidad de aire, gravedad y masa) del UAV para que el programa realice el cálculo de coeficientes aerodinámicos. Éstos resultados son los que finalmente se introducirán en el ANEXO III: *Fichero de aerodinámica* y determinarán la dinámica de vuelo en el simulador.

Se crearán varios casos variando los diferentes parámetros del UAV (ángulo de ataque, ángulo de incidencia lateral del viento, deflexión de alerones, de profundidad y de deriva, velocidad de alabeo, de cabeceo y de guiñada).

AVL no proporciona todos los coeficientes que necesitamos para la simulación en FlightGear. Sólo obtenemos de él:

- el coeficiente de arrastre total (CDtot)
- coeficiente de sustentación (CL)
- coeficiente de fuerza lateral (CY)
- coeficiente de momento de alabeo (CI)
- coeficiente de momento de cabeceo (Cm)
- coeficiente de momento de guiñada (Cn)
- la eficiencia (e)

A partir de estos 7 datos podemos obtener el resto de coeficientes calculándolos en hojas de Excel, simplemente restándole a los resultados obtenidos tras la modificación de cada parámetro, el valor del mismo coeficiente cuando todos los parámetros estaban a 0.

Las condiciones de vuelo que serán constantes para todos los cálculos son:

Número de Match	0,036
Velocidad (m/s)	12,5
Densidad del aire (kg/m <sup>3</sup> )	1,225
Aceleración de la gravedad (m/s <sup>2</sup> )	9,81
Masa (kg)	3

*Tabla 5.5: Condiciones de vuelo constantes*

En un principio, el comportamiento del UAV en el simulador poco se acercaba a la realidad, y era muy inestable durante el vuelo. Se realizó comparaciones con los resultados de otros modelos para asegurar la estabilidad y coherencia de los valores, que los signos coincidieran y que tuvieran un orden de magnitud aproximado. A la vista del comportamiento extraño en el simulador, se comparó uno por uno todos los valores propios por los aportados en la documentación de FlightGear, hasta encontrar valores con grandes diferencias, que fueron sustituidos.

Uno de los valores, el "momento de cabeceo debido a la velocidad de cabeceo" era un orden de magnitud menor que el valor de la documentación. Se realizó una modificación de ese valor multiplicándolo por 10, lo cual condujo a un comportamiento mucho más estable. Este procedimiento fue necesario para más valores. Una vez hecho esto se incorporó al modelo los winglets y el fuselaje, se repetió el cálculo aerodinámico y se introdujo los datos en el archivo de aerodinámica, para posteriormente volver a simular.

Este proceso de ajuste de parámetros se realizó utilizando como comparación los UAV reales, que se ha comprobado que poseían un cierto defecto de detección de velocidad. Se ha estudiado la posibilidad de que el defecto haya influido negativamente en el acondicionamiento de los parámetros del modelo. Los parámetros originales se consideraron como potencialmente erróneos solo en los casos en que las diferencias de comportamiento fuesen especialmente notorias, tales como signos equivocados o diferencias en los órdenes de magnitud, como se mencionó en el párrafo anterior. Estas diferencias provocaban cambios en el comportamiento del UAV muy evidentes, volviendo el vuelo muy inestable y descontrolado.

Para eliminar la influencia del defecto de velocidad del modelo virtual existen dos alternativas. Se puede repetir todas las fases del proceso de virtualización a partir de éste punto, o se puede mantener los valores originales, y realizar las modificaciones necesarias en el modelo virtual final una vez se ha comparado con los datos reales, más actuales y sin defecto de velocidad. Se ha considerado que la diferencia de comportamiento entre ambos estados del UAV real es lo suficientemente sutil para que no afecte al diseño virtual en términos generales, y su impacto entra más en un ajuste fino. Por tanto se ha decidido mantener el diseño original hasta el conjunto final, y se ajustará ese conjunto en base a las pruebas de vuelo actuales.

A continuación se incluye una explicación de todos los coeficientes aerodinámicos considerados en el desarrollo del modelo virtual original, los resultados del AVL y las correcciones necesarias.

Han sido agrupados en 6 grupos, atendiendo a la fuerza o movimiento que afecta cada uno: fuerza de arrastre, fuerza lateral, fuerza de sustentación, alabeo, cabeceo y guiñada.

## FUERZAS DE ARRASTRE

- **CDO:** *coeficiente de arrastre a sustentación cero.*

Parámetro adimensional que relaciona la fuerza de arrastre de una aeronave en reposo (sustentación cero), con su forma, tamaño y rugosidad de superficie. Refleja el arrastre "parasitario".

Se han obtenido los resultados de este valor variando el ángulo de ataque para valores de 0°, 2°, 4°, 8°, 10° y 45°, posteriormente añadiendo el resultado para 90°. Se ha usado la ecuación:

$$CDO = CD - CL^2 / (\rho * B_{ref}^2 / S_{ref} * e)$$

Los resultados fueron:










$\alpha(^{\circ})$	CDO
0	0,0008
2	0,0005
4	0,0005
8	0,0009
10	0,0013
45	0,0146

*Tabla 5.6: CDO en función de  $\alpha$*

Este parámetro, tiene especial influencia sobre la velocidad de vuelo del UAV. A menor valor, mayor la velocidad que es capaz de alcanzar, y viceversa.

En simulación con FlightGear las velocidades que alcanzaba llegaban a máximos de 100 nudos (=185,2 km/h) aproximadamente. No se corresponde con la realidad, puesto que se ha visto que la velocidad del UAV en vuelo con los motores a media potencia es aproximadamente de 50 km/h.

Para encontrar el error de estos resultados los comparamos con la siguiente tabla de coeficientes de arrastre parasitarios de diferentes geometrías:

Shape	Drag Coefficient
Sphere → 	0.47
Half-sphere → 	0.42
Cone → 	0.50
Cube → 	1.05
Angled Cube → 	0.80
Long Cylinder → 	0.82
Short Cylinder → 	1.15
Streamlined Body → 	0.04
Streamlined Half-body → 	0.09

### Measured Drag Coefficients

*Ilustración 5.9: Coeficientes de arrastre parasitarios*

En comparación, nuestros resultados son 2 órdenes de magnitud más pequeños que los indicados en la tabla para cuerpos aerodinamizados.

AVL tiene limitaciones para aceptar datos de rugosidad o imperfecciones de superficie (que existen en el UAV real), lo cual afecta a los resultados obtenidos de CD0.

Para corregir el defecto se repitieron las pruebas para los ángulos 0°, 8° y 10°, y para completar la curva de forma coherente se añadió un ángulo de 90° (no es una situación realista, pero se requería geoméricamente). Finalmente calculamos el arrastre para el ángulo de  $\alpha=45^\circ$  mediante la ecuación:

$$CD_0 = 1,28 * \sin(\alpha)$$

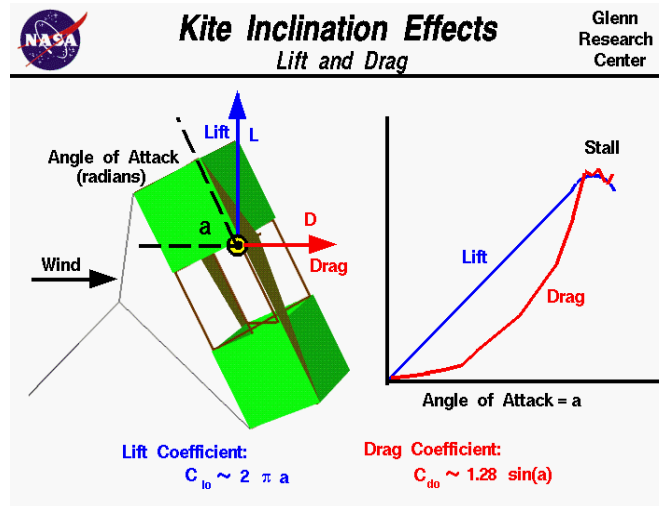


Ilustración 5.10: Aproximación de  $CD_0$  en función de  $\alpha$

Para ángulos de ataque  $>50^\circ$  no es posible realizar el cálculo en AVL, aparece el mensaje: "Cannot trim. Alpha too large." Por lo tanto los resultados que introducimos finalmente en el archivo son:

$\alpha(^{\circ})$	$CD_0$
0	0,08
8	0,09
10	0,13
45	0,905
90	1,28

Tabla 5.7:  $CD_0$  ajustado

Una vez introducidos, el UAV alcanza una velocidad razonable en el simulador, de entre 24 y 30 nudos (entre 45 y 55 km/h aproximadamente) con un ángulo de ataque de 0°, lo que coincide aproximadamente en rango con las velocidades alcanzadas en las pruebas de vuelo reales.

- **CDi:** *coeficiente de arrastre inducido.*

Es un parámetro adimensional que aparece cuando un flujo de aire circula alrededor del cuerpo del UAV y genera una fuerza de sustentación. A diferencia que en el caso anterior, para que se de este efecto debe existir una velocidad relativa entre el cuerpo y el flujo que circula a su alrededor. También es el arrastre producido por la punta de ala cuando existe sustentación.

Los resultados obtenidos se ven en la siguiente tabla, y no necesitaron modificación alguna. El arrastre inducido para  $\alpha=90^\circ$  fue introducido posteriormente.

$\alpha(^{\circ})$	CD
0	0,0083
2	0,0152
4	0,024
8	0,047
10	0,061
45	0,3598
90	0,3704

*Tabla 5.8: CD inducido en función de  $\alpha$*

Este coeficiente es probablemente uno de los que podría ser sensible a las correcciones del UAV real. Por fortuna no ha sido el caso, porque no ha necesitado de ajustes que introdujese el defecto en su cálculo. Esta eventualidad refuerza la decisión anteriormente mencionada de que es aceptable actualizar las mejoras del UAV al final del proceso.

- **CD $\beta$ :** *coeficiente de arrastre debido al deslizamiento lateral*

Cuando el UAV se encuentra en vuelo, la dirección de vuelo no tiene por que ser la misma que la del viento, cuando la dirección del viento es oblicua a la trayectoria de la aeronave, se ejercen fuerzas que provocan un giro, y por lo tanto un cambio de trayectoria. Para considerar este efecto se considera este coeficiente.

Este ángulo entre la dirección del viento y la trayectoria de la aeronave se denomina  $\beta$ , y es positivo cuando el viento incide por la derecha de la aeronave (visto desde atrás), y negativo desde la izquierda.

Los resultados obtenidos para este parámetro fueron:

$\beta(^{\circ})$	CD
0	0
2	1,00E-04
4	0,00039
6	0,00087
8	0,00154
15	0,05
90	1,23

*Tabla 5.9: CD en función de  $\beta$*

Los ángulos de  $15^\circ$  y  $90^\circ$  se añadieron posteriormente en el archivo con los datos del ejemplo aportado por la documentación.

Una vez más, un coeficiente susceptible de ser afectado por el comportamiento del UAV no necesita de ajustes.

- **CDde:** *Coefficiente de arrastre debido al ángulo de profundidad*

Este coeficiente regula la contribución del elemento de profundidad al efecto arrastre. Afecta al comportamiento del UAV en despegue y aterrizaje y en las maniobras de cambio de altura. Su valor crece en función de las solicitaciones de estas operaciones (STOL “Short Take Off and Landing”).

Los resultados que obtuvimos fueron los siguientes:

Giro de profundidad (°)	CD
10	0,01304
20	0,03276
30	0,05915

*Tabla 5.10: CD debido a profundidad*

Estos valores forman una recta con una pendiente de valor **0,1321**. Para el cálculo de la pendiente se utilizará el valor de giro del elemento de control correspondiente en radianes, y no en grados como aparece en las tablas. Este valor será el del archivo.

Las siguientes tablas de datos que representadas gráficamente formen una recta tendrán una pendiente, que será el dato a introducir en el archivo, en lugar de la tabla de datos.

#### **FUERZA LATERAL**

- **CYβ:** *Fuerza lateral debido al ángulo de deslizamiento lateral beta*

Es el coeficiente que considera los efectos de la fuerza que el viento que incide lateralmente con un ángulo  $\beta$ , ejerce sobre el UAV.

Los resultados obtenidos fueron:

$\beta$ (°)	CY
0	0
2	-0,0044
4	-0,00879
6	-0,01317
8	-0,01753

*Tabla 5.11: Coeficiente de fuerza lateral en función de  $\beta$*

Estos valores forman una recta con una pendiente de valor **-0,1254**.

## FUERZAS DE SUSTENTACIÓN

**CL $\alpha$ :** Coeficiente de sustentación debido al ángulo de ataque

Es el efecto de sustentación (fuerza vertical) que se produce debido a la inclinación o ángulo de ataque de la aeronave a través del aire.

Los resultados obtenidos fueron:

$\alpha(^{\circ})$	CL
0	0,27343
2	0,37099
4	0,46751
8	0,65613
10	0,7476
45	1,81912

Tabla 5.12: CL en función de  $\alpha$

Durante la simulación el UAV generaba más sustentación con ángulos de ataque altos y por lo tanto ascendía rápidamente, cuando debería entrar en pérdida. Este es un comportamiento erróneo, puesto que se establece en varias fuentes de información que la sustentación máxima sucede con ángulos de  $15^{\circ}$  a  $20^{\circ}$ . En los resultados proporcionados por AVL la sustentación sigue creciendo a partir de ese punto (CL=1,81912 para un ángulo de  $45^{\circ}$ ).

A continuación vemos una gráfica que relaciona el CL con el ángulo  $\alpha$ , sacada de la web de la NASA:

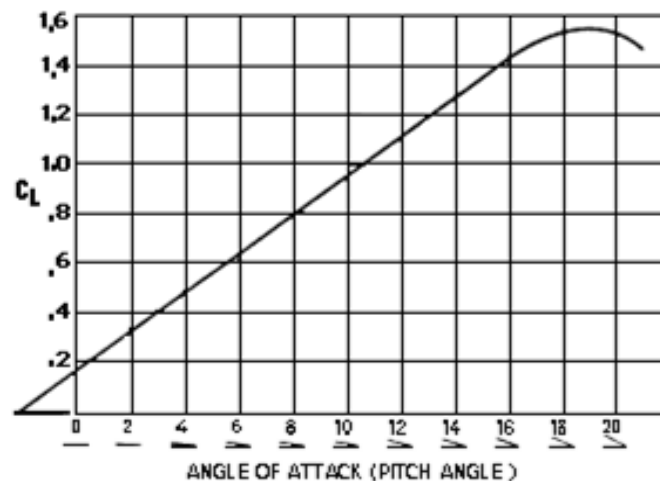


Ilustración 5.11: CL en función del ángulo de sustentación

Téngase en cuenta que los valores no tienen por qué parecerse a los de esta simulación, pero sí lo debe hacer la forma de la curva de la gráfica. Ésta crece linealmente hasta los  $19^{\circ}$  de  $\alpha$ , y a partir de ahí decrece. Un comportamiento realista de la simulación debe disponer de una curva de datos similar en comportamiento.

Para lograr esto se introdujeron tres datos más de CL para un ángulo de ataque de  $18^{\circ}$  con CL=1 (siguiendo la tendencia de la recta de la gráfica que se ve un poco más adelante), y para  $25^{\circ}$  con CL=1 también, para que la curva establezca el máximo entre estos dos valores, que es aproximadamente en donde tiene que estar el máximo, y otro dato para  $30^{\circ}$  con CL=0,7476 siguiendo la tendencia de la curva.

Para considerar el efecto de pérdida se continúa la curva por la derecha añadiendo otro dato para un ángulo de ataque de  $45^\circ$  con una gran pendiente negativa, y finalmente añadimos un dato para un ángulo de  $-10^\circ$  con  $CL=-1,2$  y otro para  $-5^\circ$  con  $CL=-0,5$  (para que continúe la curva por la izquierda, siguiendo su tendencia) para reflejar la sustentación negativa cuando el ángulo de ataque es negativo menor de  $-3^\circ$ , que es cuando empieza a existir sustentación negativa.

-10	-1,2
-5	-0,5
0	0,27343
2	0,37099
4	0,46751
8	0,65613
10	0,7476
18	1
25	1
30	0,7476
45	-0,9

Tabla 5.13:  $CL$  en función de  $\alpha$  modificado

La representación gráfica tiene el siguiente aspecto:

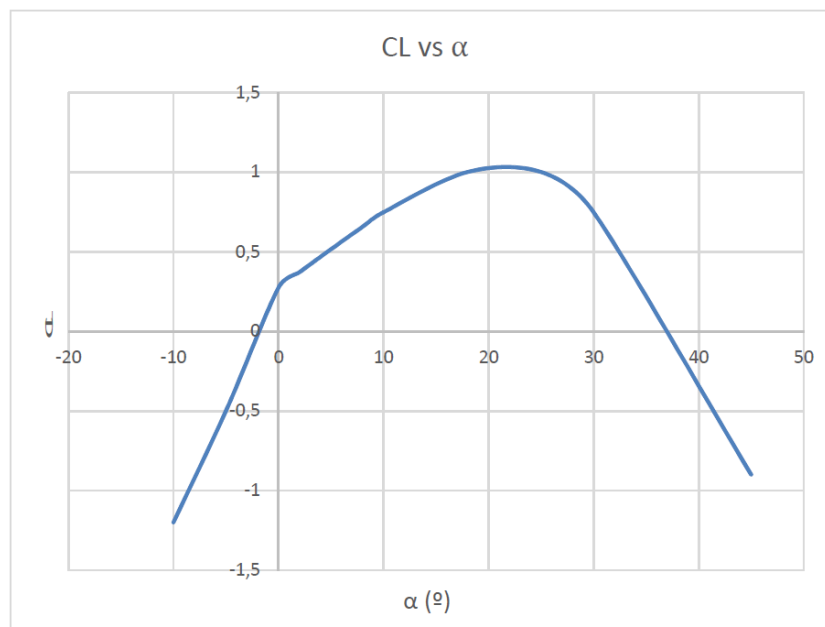


Ilustración 5.12: Curva ajustada de  $CL$  en función de  $\alpha$

Se confirman como válidos los nuevos datos después de realizar la simulación con la nueva tabla y obtener un comportamiento realista.

Esta modificación de los datos no se ha hecho en base al comportamiento en velocidad del UAV. Los datos generados por el programa no han sido modificados, sino que se han mantenido y se ha completado la tabla indicando el comportamiento deseado. Debido a la naturaleza de ésta modificación, no se introducen posibles errores por defectos de velocidad del UAV.



- **CLde:** *Coefficiente de sustentación debido al ángulo del elemento de profundidad*

Es la sustentación generada por la inclinación del elemento de profundidad (elevador) que sirve para hacer que la aeronave realice un movimiento de cabeceo (suba o baje en el aire), y a su vez, genera una sustentación.

Los resultados fueron:

Ángulo de deflexión (°)	CL
10	0,151
20	0,3021
30	0,4533

*Tabla 5.14: CL en función del ángulo de profundidad*

Estos valores forman una recta con una pendiente de valor de **0,8660**.

### **ALABEO (ROLL)**

- **Ci $\beta$ :** *Momento de alabeo debido al ángulo de deslizamiento lateral beta*

Durante el vuelo se considera que la dirección de las corrientes no es totalmente recta, sino que existe una diferencia entre la dirección del viento y la del UAV. El UAV alabea hacia la dirección de deslizamiento lateral, debido principalmente al ángulo de diedro, al ángulo de flecha y el flujo de aire inducido alrededor del fuselaje. El alabeo debido al deslizamiento lateral es proporcional al ángulo  $\beta$  (ángulo de deslizamiento lateral).

Los resultados obtenidos fueron:

$\beta$ (°)	Cl
0	0
2	-0,00221
4	-0,00441
6	-0,00659
8	-0,00873

*Tabla 5.15: Cl en función de  $\beta$*

Estos valores forman una recta con una pendiente de valor **-0,0623**.

- **Cip:** *Momento de alabeo debido a la velocidad de alabeo*

Hace referencia al derivativo del amortiguamiento del movimiento de alabeo, es decir, el propio movimiento de alabeo genera un momento de alabeo debido a que el área de sustentación cambia al girar la aeronave cuando existe un ángulo de diedro, cosa que sucede.

### Stability recover by a dihedral wing

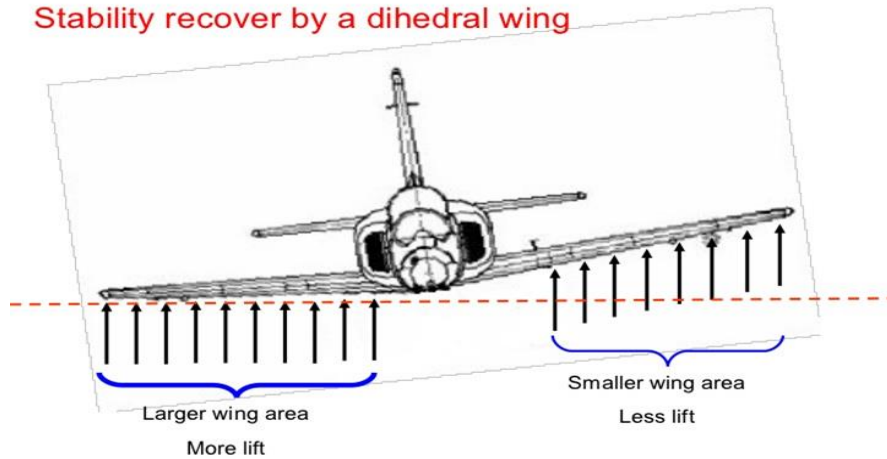


Ilustración 5.13: Efecto del ángulo de dihedro

El ángulo de dihedro es el ángulo que forman las superficies inferiores de las alas con la horizontal. Este ángulo es negativo cuando es hacia arriba (como en la imagen) y positivo cuando es hacia abajo. En este caso existe un dihedro negativo de  $-2^\circ$ .

Los resultados obtenidos fueron:

Velocidad (rad/s)	Velocidad ( $^\circ$ /s)	Cl
0,261799388	15	-0,1049
0,523598776	30	-0,2097
0,785398163	45	-0,3146
1,047197551	60	-0,4195

Tabla 5.16: Cl en función de la velocidad de alabeo

Estos valores forman una recta con una pendiente de valor **-0,4006**.

- **Clr:** Momento de alabeo debido a la velocidad de guiñada

Al girar la aeronave sobre el eje Z (guiñada), la presión ejercida por el viento sobre la superficie de la cola vertical y otras superficies laterales, genera un momento sobre el UAV (al no estar el centro de presiones en la misma horizontal que el centro de masas del UAV) que provoca un movimiento de alabeo.

Los resultados obtenidos fueron:

Velocidad (rad/s)	Cl
0,261799388	0,02452
0,523598776	0,04904
0,785398163	0,07357
1,047197551	0,0981

Tabla 5.17: Cl en función de la velocidad de guiñada

Estos valores forman una recta con una pendiente de valor **0,0937**.

- **Clda:** *Momento de alabeo provocado por el ángulo del alerón*

Para hacer girar una aeronave sobre el eje X (alabeo) se inclinan los alerones, uno hacia arriba y otro hacia abajo, provocando el giro hacia el lado del alerón levantado. Este coeficiente considera el momento generado por los alerones para hacer girar la aeronave.

Los resultados obtenidos fueron:

Giro del alerón (°)	Cl
10	0,03529
20	0,07058
30	0,10587

*Tabla 5.18: Cl en función del giro del alerón*

Estos valores forman una recta con una pendiente de valor **0,2022**.

- **Cldr:** *Momento de alabeo provocado por el ángulo de la deriva*

Momento generado por la incidencia del viento sobre el elemento de deriva o “timón” de la aeronave cuando este se encuentra girado un cierto ángulo. Afecta tanto al movimiento de alabeo como a la guiñada.

Los resultados obtenidos fueron:

Giro de la profundidad (°)	Cl
10	0,00127
20	0,00253
30	0,0038

*Tabla 5.19: Cl en función del giro de la profundidad*

Estos valores forman una recta con una pendiente de valor **0,0072**.

### **CABECEO (PITCH)**

- **Cm $\alpha$ :** *Momento de cabeceo debido al ángulo de ataque*

Es el momento que provoca la fuerza del viento debido a la inclinación del UAV (ángulo de ataque), que provoca un movimiento de cabeceo. Los resultados obtenidos son los siguientes:

$\alpha$ (°)	Cm
0	-0,06177
2	-0,0639
4	-0,06562
8	-0,06783
10	-0,0683
45	0,01415
90	0,09682

*Tabla 5.20: Cm en función de alfa*

Esta tabla no se corresponde con una línea recta. Debe introducirse íntegra en el archivo.

- **Cmde:** *Momento de cabeceo provocado por el elemento de profundidad*

Hace referencia al momento generado por la inclinación de la profundidad, situada en la parte trasera del UAV y cuya misión es la de hacer que ascienda o descienda. Es el elemento que más influencia tiene en el movimiento de cabeceo.

Los resultados obtenidos fueron:

Giro de la profundidad (°)	Cm
10	-0,07262
20	-0,14551
30	-0,21866

Tabla 5.21: Cm en función del giro de la profundidad

Estos valores forman una recta con una pendiente de valor **-0,4184**.

- **Cmq:** *Momento de cabeceo debido a la velocidad de cabeceo*

Expresa el cambio en el momento de cabeceo provocado por el propio movimiento de cabeceo, es decir, cuando la aeronave realiza un movimiento de cabeceo (gira sobre el eje "Y"), el viento incidirá con un cierto ángulo sobre la parte superior o inferior del fuselaje y las alas, provocando un momento que tendrá efecto sobre el movimiento de cabeceo.

Los resultados obtenidos fueron:

Velocidad (rad/s)	Cl
0,261799388	-0,27998
0,523598776	-0,56502
0,785398163	-0,85509
1,047197551	-1,1502

Tabla 5.22: Cl en función de la velocidad de cabeceo

Los valores forman una curva aproximable a una recta con una pendiente de valor **-1,1080**.

Se determinó, tras las pruebas de simulación y sustitución, que éste factor era el principal causante de la inestabilidad de vuelo presente en las pruebas con FlightGear. Comparándolo con su equivalente en otro modelo proporcionado por la documentación del programa se encontró que dicho factor era de un orden de magnitud mayor que el nuestro. Se procedió a equiparar ambos valores, multiplicando por 10 el de éste proyecto (**11,080**), y al relanzar la simulación el vuelo se estabilizó, por lo que el dato se considera válido.

Nuevamente una modificación del modelo se realiza en base a una comparación con otro modelo ya existente, no afectando el comportamiento en velocidad del UAV real.

## GUIÑADA (YAW)

- **Cnb:** *Momento de guiñada debido al ángulo de deslizamiento lateral  $\beta$*

El viento lateral incidente con ángulo beta ejerce una fuerza sobre las superficies laterales del UAV, como pueden ser las colas verticales o los winglets, que provoca un momento que a su vez genera un giro de guiñada (eje "Z").

Los resultados obtenidos fueron:

$\beta(^{\circ})$	Cn
0	0
2	0,00053
4	0,00106
6	0,00158
8	0,0021

*Tabla 5.23: Cn en función de  $\beta$*

Estos valores forman una recta con una pendiente de valor **0,0150**.

- **Cnr:** *Momento de guiñada debido a la velocidad de guiñada*

Al realizar un movimiento de guiñada, el viento que inicialmente incidía de frente, ahora incide con un ángulo que va creciendo según va girando el UAV sobre el eje "Z", o bien, si el viento incidía con un ángulo beta, ahora incidirá con un ángulo mayor o menor dependiendo del sentido de giro. Por esto es necesario considerar el momento de guiñada debido al propio movimiento de guiñada.

Los resultados obtenidos fueron:

Velocidad de guiñada (rad/s)	Cl
0,261799388	-0,00289
0,523598776	-0,00578
0,785398163	-0,00867
1,047197755	-0,01155

*Tabla 5.24: Cl debido a la velocidad de guiñada*

Estos valores forman una recta con una pendiente de valor **-0,0110**.

- **Cndr:** *Momento de guiñada provocado por el ángulo de la deriva*

La deriva es el principal elemento generador de guiñada del UAV, al estar situado en la parte trasera de las colas verticales. Cuando se giran estas superficies hacia uno de los lados, el viento que incide sobre ellas provoca un momento gracias al cual la aeronave puede realizar el movimiento de guiñada.

Los resultados obtenidos fueron:

Giro de la deriva ( $^{\circ}$ )	Cn
10	-0,00208
20	-0,00416
30	-0,00624

*Tabla 5.25: Cn en función del giro de la deriva*

Estos valores forman una recta con una pendiente de valor **-0,0119**.

Estos son todos los coeficientes necesarios para la creación del archivo de dinámica de vuelo para el simulador. Se puede ver el conjunto de todos los cálculos y correcciones en el ANEXO III: Fichero de aerodinámica.

### 5.3.4. Tren de aterrizaje e inercias

Para completar el UAV virtual es necesario unir todos los archivos creados, mediante un 'coordinador'. Este nexo es un archivo principal, que debe reunir todas las características geométricas principales del UAV, tenga definido un tren de aterrizaje que sirva de apoyo durante el despegue y aterrizaje y declarar las posiciones en las que actúan los motores y su sentido de giro. También debe contener las inercias, que tienen gran influencia en el comportamiento durante el vuelo.

El archivo se puede consultar en el ANEXO IV: Fichero de tren de aterrizaje e inercias. Ha sido creado siguiendo la estructura de otros encontrados en la documentación de FlightGear, y sustituyendo en él los datos del UAV.

Lo primeros datos a introducir en el cabecero del fichero es el nombre del autor, fecha versión y descripción del modelo a crear.

A continuación se describen algunos datos sobre la geometría del UAV, medidos directamente sobre nuestro modelo, que es necesario introducir:

- **Área alar** (*wingarea*): 1,1534 m<sup>2</sup>. Superficie proyectada del modelo (visto desde arriba).
- **Envergadura** (*wingspan*): 1,94 m.
- **Incidencia del ala** (*wingincidence*): 0°. Ángulo con el que incide el UAV con respecto a la horizontal en el simulador.
- **Cuerda** (*chord*): 0,56 m. Necesita el valor de la longitud de la cuerda en la zona central del UAV. Medida directamente sobre el UAV.
- **Área de la cola horizontal** (*htailarea*): 0,0984 m<sup>2</sup>. Hace referencia a la superficie de la cola horizontal del UAV. Puesto que no existe una cola en esta configuración de UAV, la superficie a considerar para éste dato serán las superficies de control que controlan el movimiento de cabeceo, llamadas elevones (que además realizan el alabeo).
- **Brazo de la cola horizontal** (*htailarm*): 0,372 m. Es la distancia desde el centro de masas del UAV hasta el centro de presiones de los 4 elementos de profundidad, los elevones en este caso.
- **Área de cola vertical** (*vtailarea*): 0,0627 m<sup>2</sup>. Considera el área de las 2 colas verticales del UAV, incluidas las superficies de las derivas.
- **Brazo de la cola vertical** (*vtailarm*): 0,365 m. Es la distancia desde el centro de masas hasta el centro de presiones de las derivas.
- **Punto de Referencia Aerodinámica (AERORP)**: es el punto sobre el cual se aplican las fuerzas y momentos generados por las superficies y superficies de control declaradas anteriormente, por lo tanto este punto será el centro de masas del UAV. Sus coordenadas son:
  - X = 0,254 m
  - Y = 0 m
  - Z = 0 mDato medido empíricamente sobre el UAV real.
- **"Punto de vista del piloto" (EYEPOINT)**: situado por conveniencia en el centro de masas (no existe una cabina en el UAV, pero es un punto obligatorio para el programa).

- **Punto de Referencia Visual (VRP):** es un punto que sirve para conciliar el sistema de coordenadas del sistema mecánico y el sistema de coordenadas del modelo gráfico. Por norma general en aeronáutica, este punto se sitúa en el morro de la aeronave a modelar, por lo que en nuestro caso las coordenadas de este punto serán:

- $X = 0 \text{ m}$
- $Y = 0 \text{ m}$
- $Z = 0 \text{ m}$

- **Balance de masas:** en este apartado se declaran las inercias, el peso total, el centro de masas y la carga y su punto de aplicación sobre el UAV.

Las inercias son una característica fundamental en aeronaves. Tiene gran importancia para nosotros porque influye mucho en el comportamiento del UAV en el simulador. Al tratarse de una geometría compleja no es posible calcularlo teóricamente. No obstante se pueden emplear métodos experimentales para su cálculo, y en este caso usaremos un péndulo bifilar para hallarlas.

El proceso completo así como la justificación de los resultados se puede ver en el ANEXO VI: Cálculo de inercias

Inercia	Valor(kg*m <sup>2</sup> )
Ixx	1,5956
Iyy	0,4208
Izz	1,7366

*Tabla 5.26: Inercias*

- **Peso y punto de aplicación:** 3kg con el punto de aplicación en el centro de masas, teniendo en cuenta todos los sistemas adicionales del UAV, incluidas las baterías, motores, hélices, servomotores, cámara, etc.

- **Carga y punto de aplicación:** 0kg, puesto que este UAV no transporta carga. El punto de aplicación se sitúa en el centro de masas por conveniencia.

- **Tren de aterrizaje (*ground\_reacctions*):** El UAV no usa tren de aterrizaje, sino que aterriza y despegue sobre su propio fuselaje. Sin embargo el programa necesita declarar unos puntos de contacto entre el UAV y el suelo que sirvan como tren aterrizaje en la simulación.

Para la simulación, establecemos estos puntos para que durante el despegue o el aterrizaje el UAV tenga una cierta estabilidad. En nuestro caso tendremos 3 puntos de contacto virtuales (llamados "BOGEY" en el código) entre el UAV y el suelo: uno en la punta trasera del ala izquierda ("LEFT\_MLG"), otro en la punta trasera del ala derecha ("RIGHT\_MLG") y otro en el morro del UAV ("NOSE\_LG"). Las coordenadas de estos puntos se miden en unidades de pulgadas (").

Medidas directamente sobre el UAV (salvo la coordenada Z que se coloca aproximadamente para situar el UAV con un cierto ángulo de ataque en reposo) resultaron ser:

1. LEFT\_MLG: en la punta trasera del ala izquierda
  - $X = 28''$
  - $Y = -38''$
  - $Z = -6''$
2. RIGHT\_MLG: en la punta trasera del ala derecha
  - $X = 28''$
  - $Y = 38''$
  - $Z = -6''$
3. NOSE\_LG: en el centro del morro
  - $X = 0''$
  - $Y = 0''$
  - $Z = -10''$

Estos puntos de contacto necesitan un número adicional de características físicas que puedan determinar el comportamiento de los mismos.

- **Coefficiente de fricción estática** (*static\_friction*): el mismo valor para los 3 puntos de contacto. Suponemos que son de caucho y se aterriza sobre cemento seco. Su valor es 1.
- **Coefficiente de fricción dinámica** (*dynamic\_friction*): manteniendo la suposición de antes, el coeficiente de fricción dinámica caucho - cemento seco es 0,8
- **Coefficiente de fricción de rodadura** (*rolling\_friction*): suponiendo lo mismo que en el coeficiente anterior, su valor es de 0,02.
- **Coefficiente de rigidez** (*spring\_coeff*): para el caucho es de  $4,1 \cdot 10^9$  Pa, que en unidades del sistema internacional es  $4,1 \cdot 10^9$  kg/cm<sup>2</sup>. En la realidad no hay superficie de contacto, ya que el supuesto contacto puntual en el simulador es ideal, por lo tanto debemos considerar un coeficiente de rigidez en unidades de kg/cm, como si fuese un muelle. Este muelle no tiene un coeficiente de rigidez ni un coeficiente de amortiguamiento muy alto, porque el modelo real aterriza sobre hierba con su fuselaje de poliestireno (y una capa de proliporileno), y no rebota excesivamente. Asignamos un coeficiente de 60 lb/ft para los 3 apoyos.
- **Coefficiente de amortiguamiento** (*damping\_coeff*): por las razones explicadas en el apartado anterior, tampoco hay un valor demasiado alto de este coeficiente. Su valor para los 3 puntos de reacción será 10 lb/ft·s.



- **Sistema de propulsión** (*propulsion*): determina la posición de los motores y hélices. Los 2 motores están ubicados a 25 cm a los lados desde el medio del UAV y sobresalen hacia delante del borde de ataque de las alas, quedando las hélices en el mismo plano que el morro del UAV. Las coordenadas de motores y hélices son:

- $X = 0 \text{ m}$
- $Y = 0,25 \text{ m}$  (motor y hélice derecha) y  $-0,25 \text{ m}$  (motor y hélice izquierda)
- $Z = 0 \text{ m}$

Propulsan completamente en el eje X, por lo que no están girados en ninguno de los ejes:

- $\text{Roll} = 0^\circ$
- $\text{Pitch} = 0^\circ$
- $\text{Yaw} = 0^\circ$

El sentido de giro es antihorario para el motor de la derecha (visto desde atrás) y horario para el motor izquierdo.

- **Control de vuelo** (*flight control*): esta sección asigna los controles de las superficies de control del UAV. Los alerones y elementos de profundidad se unen en un único elemento, que son los elevones. Son estos elementos los que se controlan con el teclado, de forma que las señales de entrada que llegan desde el teclado o ratón durante la simulación se transforman en una señal que gira estas superficies en el sentido adecuado para conseguir el giro deseado. Además tienen integrados “trims” que sirven para corregir la posición de cada superficie de control en caso de que existan desviaciones de su inclinación.

Este apartado de diseño emplea solo datos físicos del UAV, no originarios de pruebas de vuelo, por lo que no es susceptible de posibles errores por la conducta defectuosa localizada en los sensores. Por este motivo se da por correcto y se mantiene igualmente útil.

## 6. PRUEBAS Y MEJORAS

Llegados a este punto cabe preguntarse cual es el posible impacto de las correcciones del modelo sobre el conjunto de archivos y el comportamiento en comparación con el UAV real en vuelo. Se realizará una comparación del funcionamiento de ambos sistemas, realizando las mismas pruebas de vuelo, y analizando sus valores de velocidad y cambios de altitud.

Conociendo ahora el error que existía en el sistema de detección de velocidad del UAV real, es de esperar que exista una nueva diferencia de comportamiento.

### 6.1. Pruebas

Respetando la legislación vigente sobre vuelo de Vehículos Aéreos No Tripulados, las pruebas se han llevado a cabo en la pista del Club de Aeromodelismo “A Pombiña” situada en Narón. Es la misma pista de vuelo que se utilizó en el proyecto original, lo cual resulta un efecto beneficioso, puesto que proporciona unas condiciones de vuelo que son lo más similares que se puede lograra a las de las pruebas del proyecto original.

Todos los datos medidos por los sensores, cuyas funcionalidades y características ya se han estudiado, se transmiten mediante un enlace de radiofrecuencia de largo alcance a 868 MHz hasta una estación en tierra receptora, diseñada y construida por Daniel García Gonçalves en un proyecto anterior de esta misma línea de investigación.

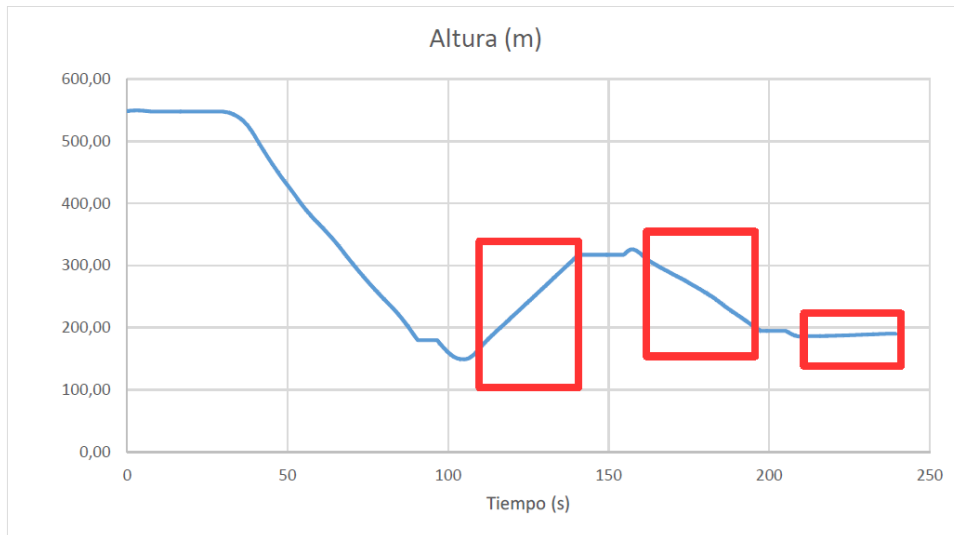
Las pruebas de vuelo a realizar son las siguientes: Subida y bajada controladas a velocidad constante (motor al 100 y 0%), y vuelo a velocidad constante y altitud constante, con el motor al 38% de su potencia total.

Debido a unas condiciones meteorológicas precarias la mayor parte de las sesiones, se ha dado con frecuencia la eventualidad de que rachas de viento elevadas invalidasen las pruebas o descontrolasen el UAV. Esto ha obligado a efectuar las pruebas de vuelo lo más seguidas en el tiempo posible, logrando finalmente conseguir una serie de datos fiables en una sola sesión de vuelo, en la cual las condiciones meteorológicas fueron idóneas. Se logró un único vuelo en el cual se realizaron las pruebas de ascenso, descenso y altura constante. Lamentablemente, y una vez más por las condiciones meteorológicas, no se ha conseguido reunir suficientes datos fiables para efectuar con seguridad una prueba para evaluar la capacidad de giro, por lo que no se dispone de información relacionada a ella.

Los datos se exportan a Excel para facilitar su manejo, dado que es una cantidad de información muy elevada.

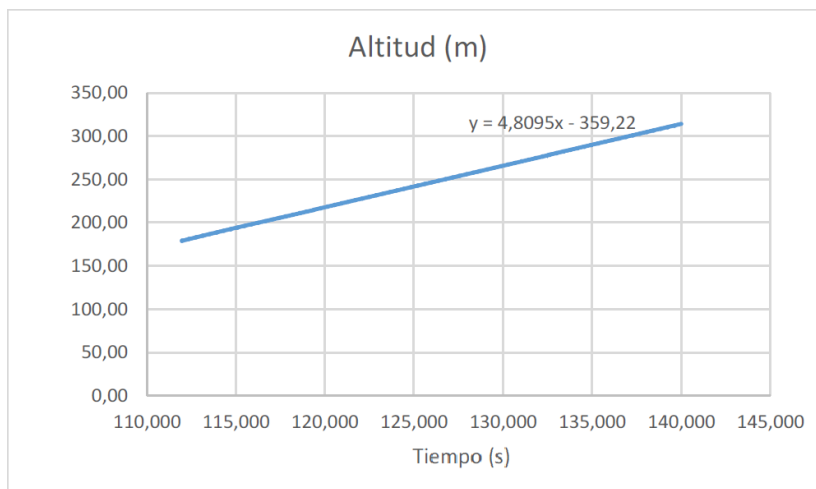
#### 6.1.1. Resultados de simulación

Utilizando el entorno de simulación FlightGear, se ha realizado todos los pasos de interés para el análisis de vuelo en una sola sesión. La simulación ha comenzado a una altura de 550 metros, muy por encima de la que se emplea para las pruebas reales, por lo que ha sido necesario redirigir el UAV hasta una altura con las condiciones adecuadas para comenzar las pruebas (aproximadamente entre 100 y 300 metros).



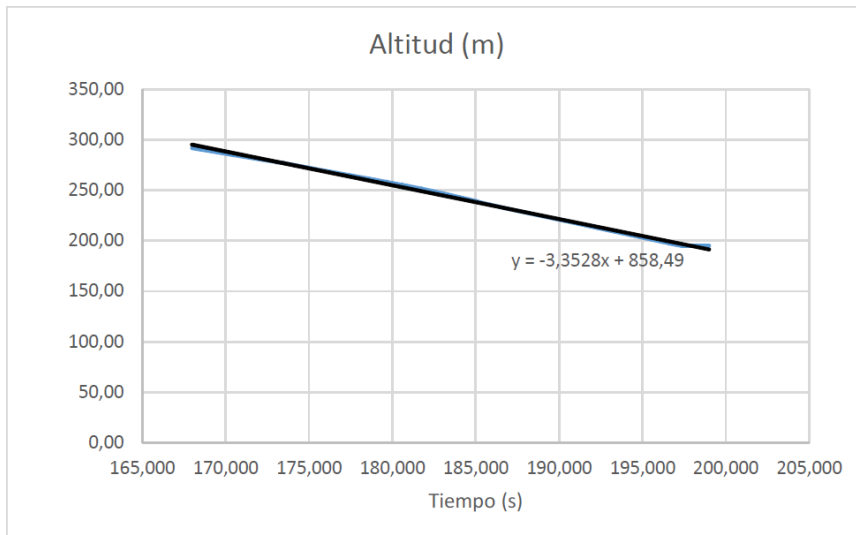
*Ilustración 6.1: Datos de simulación de vuelo*

Los tramos resaltados se corresponden con las pruebas de subida constante, bajada constante y altura constante, respectivamente.



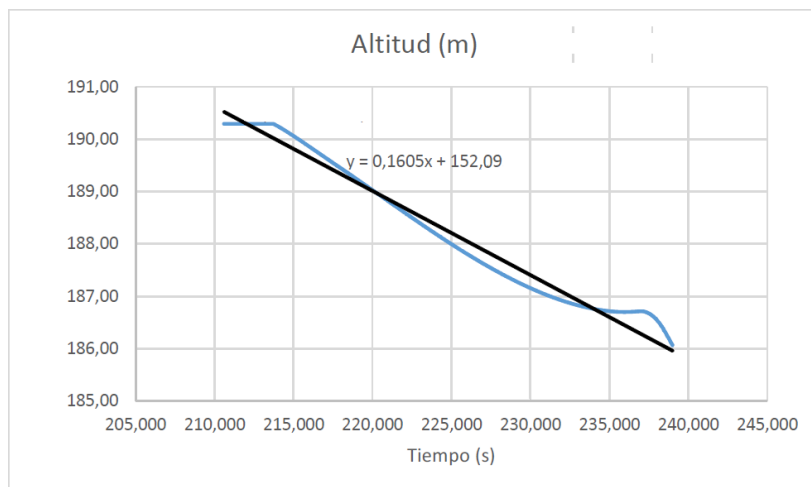
*Ilustración 6.2: Datos de simulación de prueba de subida*

Los datos de la parte de subida muestran una pendiente de subida de 4.8095 metros por segundo. Esto es, cada segundo el UAV asciende 4.8095 metros.



*Ilustración 6.3: Datos de simulación de prueba de bajada*

Los datos de bajada muestran una pendiente de -3.3528 metros bajados por el UAV durante cada segundo transcurrido.



*Ilustración 6.4: Datos de simulación de prueba de altura constante*

Resulta particularmente complicado mantener una altura constante durante los vuelos de simulador, por lo que se opta por seleccionar un tramo en el cual la pendiente, y por consiguiente la modificación de altura sea muy baja. En los 30 segundos que dura este tramo existe un descenso de poco más de 4 metros, lo que consideraremos despreciable en este caso. Durante el tramo la velocidad media se ha mantenido aproximadamente en 12.81 metros por segundo.

## 6.1.2. Resultados de vuelo

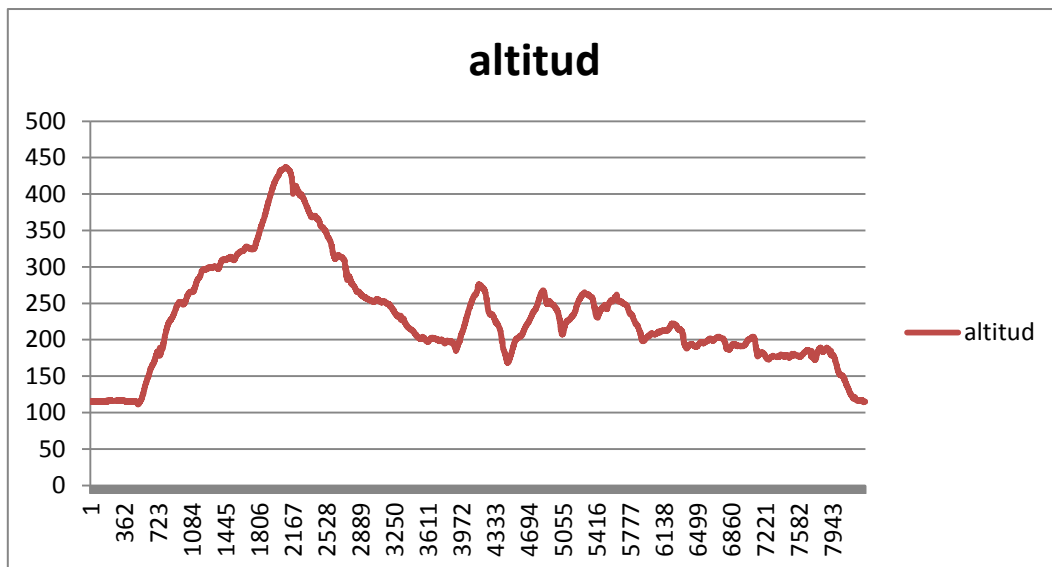


Ilustración 6.5: Prueba de vuelo: datos de altitud

El método de extracción de datos es ligeramente distinto en la forma de representar el tiempo. El programa de gestión de datos del UAV proporciona una lectura completa de todos los parámetros de medición que puede proporcionar el UAV, cada décima de segundo. Este sistema proporciona una capacidad de seguimiento muy precisa con el coste de manejar una cantidad masiva de datos.

Durante el vuelo completo se realizaron un total de 8293 medidas, lo que se corresponde con 829 segundos o 13 minutos y 49 segundos en total.

Debido a las posibles rachas de viento ha sido necesario seleccionar cuidadosamente las muestras de datos que se pueden emplear como patrón. Los tramos elegidos, por su estabilidad y consistencia, han sido los siguientes (recuadros):

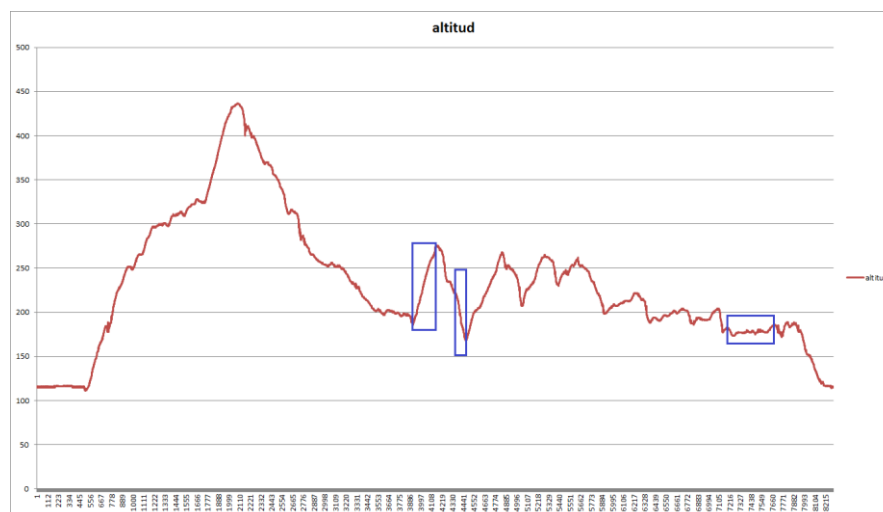


Ilustración 6.6: Prueba de vuelo: tramos estables seleccionados

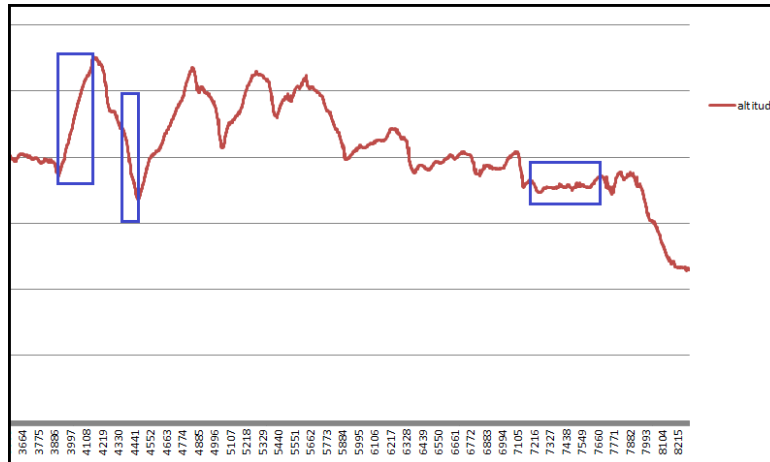


Ilustración 6.7: Datos seleccionados

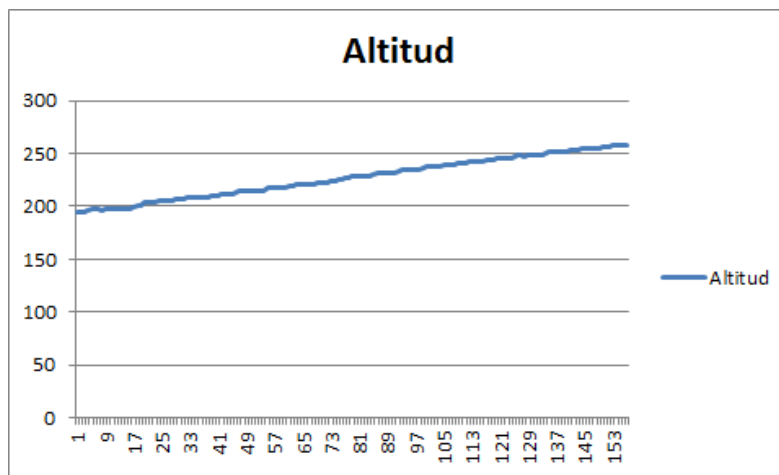


Ilustración 6.8: Datos de subida

La gráfica de la sección del primer recuadro muestra una subida constante de 195 a 265 metros en un total de 153 medidas, o 15.3 segundos. Esto dá un ratio de subida de 70 metros en 15.3 segundos, o 4.5751m/s.

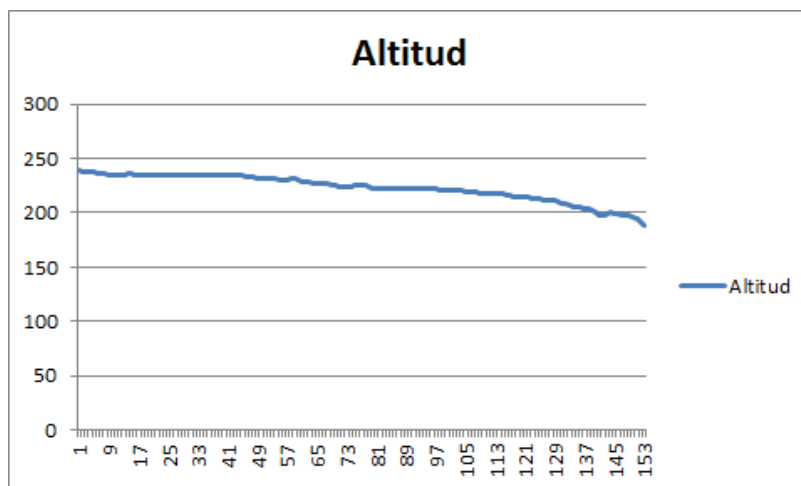


Ilustración 6.9: Datos de bajada

El segundo recuadro muestra una bajada aproximadamente constante de 238 a 188 metros, en un total de 152 medidas o 15.2 segundos. El ratio de bajada es de 50 metros en 15.2 segundos, o 3.2894m/s.

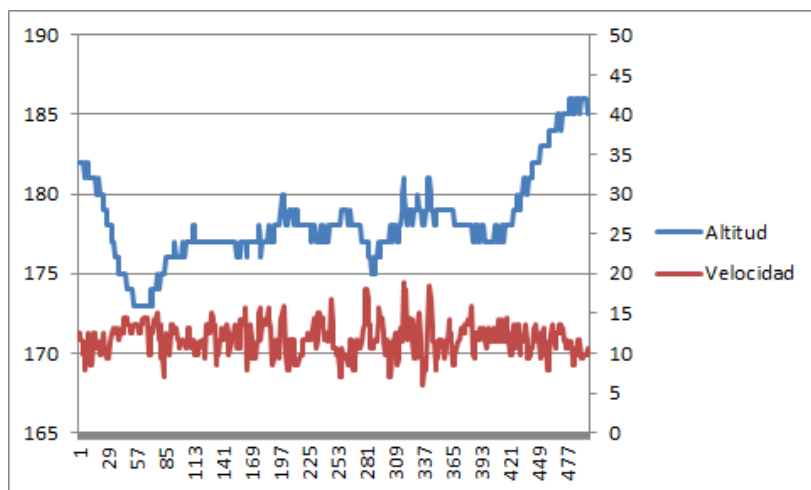


Ilustración 6.10: Datos de altitud constante

La gráfica de altitud constante muestra el efecto de las rachas de viento que afectan inevitablemente a todas las pruebas. El UAV realiza un vuelo con una altitud que varía entre 175 y 180 metros generalmente, con algunos picos provocados por rachas de viento. Durante este tramo de altitud razonablemente constante la velocidad mantiene un valor medio de 12.5 a 13m/s. Tomaremos 12.75m/s como valor medio representativo.

## 6.2. Análisis de los resultados del modelo

Al comparar los resultados obtenidos con los de la simulación obtenemos los siguientes resultados.

	Vuelo real	Simulación	Diferencia
Subida constante	4,5751m/s	4,8095m/s	5,12%
Bajada constante	3,2894m/s	3,2528m/s	1,11%
Altura constante	12,75m/s	12,81m/s	0,47%

Tabla 6.1: Comparación de datos

El UAV virtual original, tal cual se había modelado resulta mucho más preciso para los datos de vuelo con el UAV actual.

Existe una diferencia moderada en el comportamiento del modelo en cambio de altura, pero dentro de márgenes aceptables. En general todos los valores sugieren una precisión muy alta.

La diferencia más elevada se encuentra en el comportamiento de los dos modelos en subida. La velocidad de la simulación es ligeramente mayor que en la realidad. Los coeficientes de simulación más seguros para regular este aspecto del modelo virtual son los coeficientes de arrastre y sustentación.

Se comprobó que modificar los coeficientes de arrastre y sustentación para intentar incrementar la precisión de subida en este caso produce una modificación a peor en la precisión de las otras dos pruebas.

Arrastre	
$\alpha$	CDO
-90	1,28
-45	0,905
-10	0,13
-8	0,09
0	0,08
8	0,09
10	0,13
45	0,905
90	1,28

*Tabla 6.2: Coeficientes de arrastre (CDO) finales*

Sustentación	
$\alpha$	$CL\alpha$
-10	-1,2
-5	-0,5
0	0,2734
2	0,371
4	0,4675
8	0,6561
10	0,7476
18	1
25	1
30	0,7476
45	-0,9
90	-2

*Tabla 6.3: Coeficientes de sustentación (CL $\alpha$ ) finales*

El conjunto de coeficientes finales del modelo aparecen en el ANEXO III: *Fichero de aerodinámica*.



## 7. CONCLUSIONES

El proyecto original se desarrolló con objeto de crear un modelo de vehículo aéreo no tripulado en un simulador de vuelo, lo más preciso posible y usando en la medida de lo posible software de código abierto. Todo ello con la finalidad de poder realizar pruebas de vuelo en un simulador, sin los inconvenientes que conlleva la necesidad de sacar el UAV real a volar.

El presente proyecto se orientó a revisar y corregir los posibles defectos del proyecto anterior, lo cual incluye un examen de todos los componentes de vuelo del modelo de UAV físico, de su funcionamiento y de su exactitud. También se realizó una revisión del modelo virtual, bajo la necesidad de comprobar si había errores en su concepción, o si posibles errores de la parte física podrían haber acarreado defectos en el apartado virtual.

Se ha localizado un defecto en el funcionamiento del programa encargado de interpretar la información recogida por los tubos de pitot que realizan la función de sensores de velocidad. Dicho defecto, al tratarse principalmente del apartado software trasladaba el error de lectura de velocidad a todos los UAVs. Se comprobó que los demás sensores ofrecen una precisión adecuada y que no introducen un margen de error que resulte inaceptable al comparar comportamientos con los modelos virtuales o con los modelos físicos entre sí.

Se realizó una revisión completa de la funcionalidad y solvencia del modelo virtual desarrollado, y se realizó una comparación final de funcionamientos, para comprobar que en efecto funcionaba según lo previsto, y con un comportamiento suficientemente próximo al del UAV real (esta vez sin defectos de navegación) como para considerarlo preciso.

Dicha precisión no es absoluta debido a que inevitablemente existen ciertas licencias que debieron ser tomadas en los procesos de diseño de los elementos virtuales, como las hélices o las inercias, para las que se emplearon métodos experimentales.

Pese a ello se ha logrado realizar una revisión completa y una actualización de la funcionalidad de los modelos de UAV estudiados, y del modelo virtual para emplear en simulación. Se ha conseguido localizar los posibles causantes de pérdidas de precisión del trabajo original y se ha mejorado la precisión del sistema en conjunto.

## BIBLIOGRAFÍA

<https://es.wikipedia.org>  
<http://www.minidrons.com/ucav-uavs-tipos-drones/>  
<http://www.xdrones.es/tipos-de-drones-clasificacion-de-drones-categorias-de-drones/>  
<https://www.drones2.com.au/arduplane-airspeed-sensor-with-pilot-tube-apm-pixhawk-px4.html>  
<http://wiki.flightgear.org/>  
<http://www.blogseib.com/aviacion/2015/04/12/drone-rpas-uav-que-es-todo-esto/>  
<http://www.flightgear.org/>  
<http://www.nasa.gov/>  
[http://www.asifunciona.com/electronica/af\\_gps/af\\_gps\\_10.htm](http://www.asifunciona.com/electronica/af_gps/af_gps_10.htm)  
<http://www.dtic.upf.edu/~gvirtual/master/rv/seccio2/seccio2.htm>  
<http://www.mh-aerotoools.de/airfoils/javaprop.htm>  
<http://www.crashtesthobby.com/hercules-xl-7278.html>  
<http://www.allstar.fiu.edu/aero/BA-Background.htm>  
<http://www.meteosat.com/tiempo/a-coruna/tiempo-ferrol.html>  
<http://www.atiak.com/products/resolution-3-airframe/>  
<http://cdigital.uv.mx/bitstream/123456789/31665/1/mendezrodriguezberenice.pdf>  
<http://jsbsim.sourceforge.net/>  
[http://aeronautica.conocimientos.com.ve/2009/12/superficies-de-mando-y-control\\_11.html](http://aeronautica.conocimientos.com.ve/2009/12/superficies-de-mando-y-control_11.html)  
<http://www.landingshort.com/2007/11/07/caida-en-barrena/>  
<https://www.altomdata.dk/parrot-klar-med-droner>  
<http://prisma.dieti.unina.it/index.php/education/theses/299-sviluppo-di-un-algoritmo-di-fusione-multi-sensoriale-per-il-tracking-della-posizione-e-l-orientamento-degli-strumenti-medici-in-chirurgia-robotica>  
[http://www.buckarooshangar.com/flightgear/yasimtut\\_introduction.html](http://www.buckarooshangar.com/flightgear/yasimtut_introduction.html)  
<http://www.alasewm.com.ar/helicewm.htm>  
<http://arc.aiaa.org/doi/abs/10.2514/6.2016-0947>  
<http://www.holycows.net/datcom/>  
<http://web.mit.edu/drela/Public/web/avl/>  
[http://outerra.shoutwiki.com/wiki/JSBSim\\_Properties](http://outerra.shoutwiki.com/wiki/JSBSim_Properties)  
<http://www.manualvuelo.com/TCV/TCV57.html>  
<http://www.flightlab.net/Flightlab.net/Home.html>  
<http://www.city-gallery.com/knoblock/projects/flightgear/Docs/coords.html>  
<http://cinematicaydinamicabrmc200619921.blogspot.com.es/2010/07/formulas-de-momento-de-inercia-el.html>  
[http://www.ima.org.uk/\\_db/\\_documents/maths07\\_williams\\_huw.pdf](http://www.ima.org.uk/_db/_documents/maths07_williams_huw.pdf)  
<http://naca.central.cranfield.ac.uk/reports/1948/naca-tn-1629.pdf>

<http://www.me.utexas.edu/~me244L/labs/filar/filaroverview.html>

<http://www.waveshare.com/media/catalog/product/cache/1/image/800x800/9df78eab33525d08d6e5fb8d27136e95/1/0/10-dof-imu-sensor-b-2.jpg>



UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**Documento**

**ANEXOS**

## **INDICE DE ANEXOS**

ANEXO I: Fichero de hélices

ANEXO II: Fichero de motores

ANEXO III: Fichero de aerodinámica

ANEXO IV: Fichero de tren de aterrizaje e inercias

ANEXO V: hercules.avl

ANEXO VI: Cálculo de inercias

ANEXO VII: Manual instrucciones Airspeed Microsensor

ANEXO VIII: Manual instrucciones Guardian Stabilization

ANEXO IX: Manual instrucciones Altimeter Microsensor

ANEXO X: Plano del UAV



UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**Documento**

**ANEXO I: FICHERO DE HÉLICES**

```

<?xml version="1.0"?>
<propeller name="helices_9x6">
  <ixx unit="KG*M2"> 0.00023516 </ixx> <!--Caculado como masa por radio^2-
->
  <diameter unit="IN"> 9 </diameter>
  <numblades> 2 </numblades>
  <gearratio> 1 </gearratio>
  <minpitch> 30 </minpitch>
  <maxpitch> 30 </maxpitch>
  <minrpm> 1 </minrpm>
  <maxrpm> 10000 </maxrpm>
  <constspeed> 0 </constspeed> <!--1 = constant speed mode, 0 =
manual pitch mode-->
  <reversepitch> 0 </reversepitch> <!-- angulo en radianes sobre el
cual impulsa la hélice cos(0)=1 sobre el eje x-->
  <p_factor> 0 </p_factor> <!-- Efecto aerodinámico
explicado en la memoria. No tiene influencia en nuestro caso-->

  <table name="C_THRUST" type="internal">
    <tableData>
      0      0.011644837
      0.01  0.011524754
      0.02  0.011404672
      0.03  0.011284589
      0.04  0.011164507
      0.05  0.011044425
      0.06  0.011066151
      0.07  0.011087877
      0.08  0.011109604
      0.09  0.01113133
      0.1   0.011153057
      0.11  0.011094762
      0.12  0.011036468
      0.13  0.010978174
      0.14  0.01091988
      0.15  0.010861585
      0.16  0.010787715
      0.17  0.010713846
      0.18  0.010639976
      0.19  0.010566106
      0.2   0.010492236
      0.21  0.010394629
      0.22  0.010297022
      0.23  0.010199414
      0.24  0.010101807
      0.25  0.0100042
      0.26  0.009913639
      0.27  0.009823078
      0.28  0.009732517
      0.29  0.009641956
      0.3   0.009551395
      0.31  0.009443967
      0.32  0.009336539
      0.33  0.009229111
      0.34  0.009121682
      0.35  0.009014254
      0.36  0.008886978
      0.37  0.008759703
      0.38  0.008632427
      0.39  0.008505152
      0.4   0.008377876
    </tableData>
  </table>

```

0.41	0.008240515
0.42	0.008103155
0.43	0.007965794
0.44	0.007828433
0.45	0.007691072
0.46	0.007544419
0.47	0.007397766
0.48	0.007251113
0.49	0.007104459
0.5	0.006957806
0.51	0.006805398
0.52	0.006652991
0.53	0.006500583
0.54	0.006348175
0.55	0.006195767
0.56	0.006037282
0.57	0.005878797
0.58	0.005720312
0.59	0.005561826
0.6	0.005403341
0.61	0.005239865
0.62	0.005076388
0.63	0.004912912
0.64	0.004749435
0.65	0.004585959
0.66	0.00441915
0.67	0.004252341
0.68	0.004085532
0.69	0.003918723
0.7	0.003751914
0.71	0.003581127
0.72	0.00341034
0.73	0.003239553
0.74	0.003068766
0.75	0.002897979
0.76	0.002725489
0.77	0.002551971
0.78	0.002377866
0.79	0.002202807
0.8	0.002026354
0.81	0.00184924
0.82	0.001671098
0.83	0.001492149
0.84	0.001312686
0.85	0.001131975
0.86	0.00095053
0.87	0.000768278
0.88	0.00058676
0.89	0.000400324
0.9	0.000214989
0.91	9.84E-06
0.92	-0.000180491
0.93	-0.000368541
0.94	-0.000557913
0.95	-0.000747652
0.96	-0.000940694

```
</tableData>  
</table>
```

```
<table name="C_POWER" type="internal">  
<tableData>
```



0	0.00314904
0.01	0.003095208
0.02	0.003041376
0.03	0.002987544
0.04	0.002933712
0.05	0.00287988
0.06	0.002950423
0.07	0.003020965
0.08	0.003091508
0.09	0.003162051
0.1	0.003232593
0.11	0.003241946
0.12	0.003251298
0.13	0.00326065
0.14	0.003270003
0.15	0.003279355
0.16	0.003284571
0.17	0.003289787
0.18	0.003295003
0.19	0.003300219
0.2	0.003305435
0.21	0.003303029
0.22	0.003300623
0.23	0.003298217
0.24	0.003295811
0.25	0.003293404
0.26	0.003291905
0.27	0.003290405
0.28	0.003288905
0.29	0.003287406
0.3	0.003285906
0.31	0.00327835
0.32	0.003270794
0.33	0.003263238
0.34	0.003255682
0.35	0.003248126
0.36	0.003231588
0.37	0.00321505
0.38	0.003198513
0.39	0.003181975
0.4	0.003165437
0.41	0.003142093
0.42	0.003118749
0.43	0.003095405
0.44	0.003072062
0.45	0.003048718
0.46	0.003019161
0.47	0.002989604
0.48	0.002960047
0.49	0.00293049
0.5	0.002900933
0.51	0.002864982
0.52	0.002829031
0.53	0.00279308
0.54	0.002757129
0.55	0.002721178
0.56	0.002678923
0.57	0.002636668
0.58	0.002594413
0.59	0.002552159
0.6	0.002509904

0.61	0.002460456
0.62	0.002411008
0.63	0.002361559
0.64	0.002312111
0.65	0.002262663
0.66	0.002206507
0.67	0.002150352
0.68	0.002094196
0.69	0.00203804
0.7	0.001981885
0.71	0.001918544
0.72	0.001855203
0.73	0.001791862
0.74	0.001728521
0.75	0.00166518
0.76	0.001597695
0.77	0.001528602
0.78	0.001458027
0.79	0.001385927
0.8	0.001312014
0.81	0.001236577
0.82	0.001159492
0.83	0.0010808
0.84	0.001000666
0.85	0.000918719
0.86	0.000835165
0.87	0.000749964
0.88	0.000663897
0.89	0.000574081
0.9	0.000483564
0.91	0.000393336
0.92	0.000297958
0.93	0.000202457
0.94	0.000104978
0.95	5.93E-06
0.96	-9.61E-05

</tableData>  
</table>

</propeller>



UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**Documento**

**ANEXO II: FICHERO DE MOTORES**

```
<?xml version="1.0"?>
<!-- EMP N3530 KV1100 motor -->
<!-- web: http://www.megastorererc.com/motores-brushless/4072-motor-n353013-1100.html -->
<!-- motor de 345 W -->

<electric_engine name="n3530">
  <power unit="WATTS"> 345 </power>
</electric_engine>
```



UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**Documento**

**ANEXO III: FICHERO DE AERODINÁMICA**

```

<?xml version="1.0"?>
  <aerodynamics>
    <axis name="DRAG">
      <function name="aero/coefficient/CD0">
        <description>Drag_at_zero_lift</description>
        <product>
          <property>aero/qbar-psf</property>
          <property>metrics/Sw-sqft</property>
          <table>
            <independentVar>aero/alpha-rad</independentVar>
            <tableData>
              <tr><td>-1.5708</td><td>1.28</td></tr>
              <tr><td>-0.7854</td><td>0.905</td></tr>
              <tr><td>-0.1745</td><td>0.13</td></tr>
              <tr><td>-0.1396</td><td>0.09</td></tr>
              <tr><td>0.0000</td><td>0.08</td></tr>
              <tr><td>0.1396</td><td>0.09</td></tr>
              <tr><td>0.1745</td><td>0.13</td></tr>
              <tr><td>0.7854</td><td>0.905</td></tr>
              <tr><td>1.5708</td><td>1.28</td></tr>
            </tableData>
          </table>
        </product>
      </function>
      <function name="aero/coefficient/CDi">
        <description>Induced_drag</description>
        <product>
          <property>aero/qbar-psf</property>
          <property>metrics/Sw-sqft</property>
          <property>aero/cl-squared</property>
          <table>
            <independentVar>aero/alpha-rad</independentVar>
            <tableData>
              <tr><td>-1.5708</td><td>0.3704</td></tr>
              <tr><td>-0.7854</td><td>0.3598</td></tr>
              <tr><td>-0.1745</td><td>0.061</td></tr>
              <tr><td>-0.1396</td><td>0.047</td></tr>
              <tr><td>-0.0698</td><td>0.024</td></tr>
              <tr><td>-0.0349</td><td>0.0152</td></tr>
              <tr><td>0.0000</td><td>0.0083</td></tr>
              <tr><td>0.0349</td><td>0.0152</td></tr>
              <tr><td>0.0698</td><td>0.024</td></tr>
              <tr><td>0.1396</td><td>0.047</td></tr>
              <tr><td>0.1745</td><td>0.061</td></tr>
              <tr><td>0.7854</td><td>0.3598</td></tr>
              <tr><td>1.5708</td><td>0.3704</td></tr>
            </tableData>
          </table>
        </product>
      </function>
      <function name="aero/coefficient/CDbeta">
        <description>Drag_due_to_sideslip</description>
        <product>
          <property>aero/qbar-psf</property>
          <property>metrics/Sw-sqft</property>
          <table>
            <independentVar>aero/beta-rad</independentVar>
            <tableData>
              <tr><td>-1.5700</td><td>1.2300</td></tr>
              <tr><td>-0.2600</td><td>0.0500</td></tr>
            </tableData>
          </table>
        </product>
      </function>
    </axis>
  </aerodynamics>

```

```

        -0.1396    0.00154
        -0.1047    0.00087
        -0.0698    0.00039
        -0.0349    0.0001
        0.0000     0.0000
        0.0349     0.0001
        0.0698     0.00039
        0.1047     0.00087
        0.1396     0.00154
        0.2600     0.0500
        1.5700     1.2300
    </tableData>
</table>
</product>
</function>
<function name="aero/coefficient/CDde">
  <description>Drag_due_to_Elevator_Deflection</description>
  <product>
    <property>aero/qbar-psf</property>
    <property>metrics/Sw-sqft</property>
    <property>fcs/elevator-pos-norm</property>
    <value>0.1321</value>
  </product>
</function>
</axis>

<axis name="SIDE">
  <function name="aero/coefficient/CYb">
    <description>Side_force_due_to_beta</description>
    <product>
      <property>aero/qbar-psf</property>
      <property>metrics/Sw-sqft</property>
      <property>aero/beta-rad</property>
      <value>-0.1254</value>
    </product>
  </function>
</axis>

<axis name="LIFT">
  <function name="aero/coefficient/CLalpha">
    <description>Lift_due_to_alpha</description>
    <product>
      <property>aero/qbar-psf</property>
      <property>metrics/Sw-sqft</property>
      <table>
        <independentVar>aero/alpha-rad</independentVar>
        <tableData>
          -0.1745    -1.2
          -0.0873    -0.5
          0.0000     0.27343
          0.0349     0.37099
          0.0698     0.46751
          0.1396     0.65613
          0.1745     0.7476
          0.3142     1.0
          0.4398     1.0
          0.5341     1.0
          0.7854     -0.9
          1.5708     -2
        </tableData>
      </table>
    </product>
  </function>
</axis>

```

```

        </product>
    </function>
    <function name="aero/coefficient/CLde">
        <description>Lift_due_to_Elevator_Deflection</description>
        <product>
            <property>aero/qbar-psf</property>
            <property>metrics/Sw-sqft</property>
            <property>fcs/elevator-pos-rad</property>
            <value>0.8660</value>
        </product>
    </function>
</axis>

<axis name="ROLL">
    <function name="aero/coefficient/Clb">
        <description>Roll_moment_due_to_beta</description>
        <product>
            <property>aero/qbar-psf</property>
            <property>metrics/Sw-sqft</property>
            <property>metrics/bw-ft</property>
            <property>aero/beta-rad</property>
            <value>-0.0623</value>
        </product>
    </function>
    <function name="aero/coefficient/Clp">
        <description>Roll_moment_due_to_roll_rate</description>
        <product>
            <property>aero/qbar-psf</property>
            <property>metrics/Sw-sqft</property>
            <property>metrics/bw-ft</property>
            <property>aero/bi2vel</property> <!-- wing
span/(2*velocity)-->
            <property>velocities/p-aero-rad_sec</property>
            <value>-0.4006</value>
        </product>
    </function>
    <function name="aero/coefficient/Clr">
        <description>Roll_moment_due_to_yaw_rate</description>
        <product>
            <property>aero/qbar-psf</property>
            <property>metrics/Sw-sqft</property>
            <property>metrics/bw-ft</property>
            <property>aero/bi2vel</property> <!-- wing
span/(2*velocity)-->
            <property>velocities/r-aero-rad_sec</property>
            <value>0.0937</value>
        </product>
    </function>
    <function name="aero/coefficient/Clde">
        <description>Roll_moment_due_to_aileron</description>
        <product>
            <property>aero/qbar-psf</property>
            <property>metrics/Sw-sqft</property>
            <property>metrics/bw-ft</property>
            <property>fcs/left-aileron-pos-rad</property>
            <value>0.2022</value>
        </product>
    </function>
    <function name="aero/coefficient/Clde">
        <description>Roll_moment_due_to_rudder</description>
        <product>

```



```

        <property>aero/qbar-psf</property>
        <property>metrics/Sw-sqft</property>
        <property>metrics/bw-ft</property>
        <property>fcs/rudder-pos-rad</property>
        <value>0.0072</value>
    </product>
</function>
</axis>

<axis name="PITCH">
    <function name="aero/coefficient/Cmalpha">
        <description>Pitch_moment_due_to_alpha</description>
        <product>
            <property>aero/qbar-psf</property>
            <property>metrics/Sw-sqft</property>
            <property>metrics/cbarw-ft</property>
            <property>aero/alpha-rad</property>
            <table>
                <independentVar>aero/alpha-rad</independentVar>
                <tableData>
                    0.0000    -0.06177
                    0.0349    -0.0639
                    0.0698    -0.06562
                    0.1396    -0.06783
                    0.1745    -0.0683
                    0.7854    0.01415
                    1.5708    0.09682
                </tableData>
            </table>
        </product>
    </function>
    <function name="aero/coefficient/Cmde">
        <description>Pitch_moment_due_to_elevator</description>
        <product>
            <property>aero/qbar-psf</property>
            <property>metrics/Sw-sqft</property>
            <property>metrics/cbarw-ft</property>
            <property>fcs/elevator-pos-rad</property>
            <value>-0.4184</value>
        </product>
    </function>
    <function name="aero/coefficient/Cmq">
        <description>Pitch_moment_due_to_pitch_rate</description>
        <product>
            <property>aero/qbar-psf</property>
            <property>metrics/Sw-sqft</property>
            <property>metrics/cbarw-ft</property>
            <property>aero/ci2vel</property> <!--wing
chord/ (2*velocity)-->
            <property>velocities/q-aero-rad_sec</property>
            <value>-11.080</value>
        </product>
    </function>
</axis>

<axis name="YAW">
    <function name="aero/coefficient/Cnb">
        <description>Yaw_moment_due_to_beta</description>
        <product>
            <property>aero/qbar-psf</property>
            <property>metrics/Sw-sqft</property>

```

```

        <property>metrics/bw-ft</property>
        <property>aero/beta-rad</property>
        <value>0.0150</value>
    </product>
</function>
<function name="aero/coefficient/Cnr">
    <description>Yaw_moment_due_to_yaw_rate</description>
    <product>
        <property>aero/qbar-psf</property>
        <property>metrics/Sw-sqft</property>
        <property>metrics/bw-ft</property>
        <property>aero/bi2vel</property>
        <property>velocities/r-aero-rad_sec</property>
        <value>-0.0110</value>
    </product>
</function>
<function name="aero/coefficient/Cndr">
    <description>Yaw_moment_due_to_rudder</description>
    <product>
        <property>aero/qbar-psf</property>
        <property>metrics/Sw-sqft</property>
        <property>metrics/bw-ft</property>
        <property>fcs/rudder-pos-rad</property>
        <value>-0.0119</value>
    </product>
</function>
</axis>
</aerodynamics>

```



UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**Documento**

**ANEXO IV: FICHERO DE TREN DE ATERRIZAJE E INERCIAS**

```

<?xml version="1.0"?>

<?xml-stylesheet href="http://jsbsim.sourceforge.net/JSBSim.xsl"
type="text/xsl"?>
<fdm_config name="herculesgii" version="2.0" release="BETA"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="http://jsbsim.sourceforge.net/JSBSim.xsd">

  <fileheader>
    <author> Román Sobrino Robelo </author>
    <filecreationdate> 25-08-2017 </filecreationdate>
    <version> 1.0 </version>
    <description> Modelo Hércules GII </description>
  </fileheader>

  <metrics>
    <wingarea unit="M2"> 1.1534 </wingarea><!--Superficie proyectada
del modelo del UAV-->
    <wingspan unit="M"> 1.94 </wingspan><!--Envergadura-->
    <wingincidence unit="DEG"> 0 </wingincidence><!--Incidencia del
ala, tomado valor aproximado-->
    <chord unit="M"> 0.56 </chord> <!--Cuerda en la zona central del
ala-->
    <htailarea unit="M2"> 0.0984 </htailarea><!--No tenemos cola. Es el
área de los 4 elementos de profundidad-->
    <htailarm unit="M"> 0.372 </htailarm><!--No tenemos cola. Distancia
en metros desde el CG hasta el centro de presiones de la profundidad-->
    <vtailarea unit="M2"> 0.0627 </vtailarea><!--Área de las 2 colas
verticales, teniendo en cuenta las derivas-->
    <vtailarm unit="M"> 0.365 </vtailarm><!--Distancia desde el CG
hasta el centro de las derivas, aproximada-->
    <location name="AERORP" unit="M"> <!--Aerodynamic Reference
Point, origen para todo lo relacionado con la aerodinamica-->
      <x> 0.254 </x>
      <y> 0 </y>
      <z> 0 </z>
    </location>
    <location name="EYEPOINT" unit="M"> <!--"Punto de vista del
piloto"-->
      <x> 0.254 </x>
      <y> 0 </y>
      <z> 0 </z>
    </location>
    <location name="VRP" unit="M"> <!--Punto de Referencia Visual.
Concilia el modelo mecánico y el gráfico-->
      <x> 0 </x>
      <y> 0 </y>
      <z> 0 </z>
    </location>
  </metrics>

  <mass_balance>
    <ixx unit="KG*M2"> 1.5956 </ixx>
    <iyy unit="KG*M2"> 0.4208 </iyy>
    <izz unit="KG*M2"> 1.7366 </izz>
    <ixy unit="KG*M2"> 0 </ixy>
    <ixz unit="KG*M2"> 0 </ixz>
    <iyz unit="KG*M2"> 0 </iyz>
    <emptywt unit="KG"> 3 </emptywt>
    <location name="CG" unit="M">
      <x> 0.254 </x>

```

```

        <y> 0 </y>
        <z> 0 </z>
    </location>
    <pointmass name="payload">
        <weight unit="KG"> 0 </weight>
        <location name="payload" unit="M">
            <x> 0.254 </x>
            <y> 0 </y>
            <z> 0 </z>
        </location>
    </pointmass>
</mass_balance>

<ground_reactions><!--Tren de aterrizaje-->
    <contact type="BOGEY" name="LEFT_MLG">
        <location unit="IN">
            <x> 28.7 </x>
            <y> -38 </y>
            <z> -6 </z>
        </location>
        <static_friction> 1 </static_friction>
        <dynamic_friction> 0.8 </dynamic_friction>
        <rolling_friction> 0.02 </rolling_friction>
        <spring_coeff unit="LBS/FT"> 60 </spring_coeff>
        <damping_coeff unit="LBS/FT/SEC"> 10 </damping_coeff>
        <max_steer unit="DEG"> 0.0 </max_steer>
        <brake_group> LEFT </brake_group>
        <retractable>0</retractable>
    </contact>
    <contact type="BOGEY" name="RIGHT_MLG">
        <location unit="IN">
            <x> 28.7 </x>
            <y> 38 </y>
            <z> -6 </z>
        </location>
        <static_friction> 1 </static_friction>
        <dynamic_friction> 0.8 </dynamic_friction>
        <rolling_friction> 0.02 </rolling_friction>
        <spring_coeff unit="LBS/FT"> 60 </spring_coeff>
        <damping_coeff unit="LBS/FT/SEC"> 10 </damping_coeff>
        <max_steer unit="DEG"> 0.0 </max_steer>
        <brake_group> RIGHT </brake_group>
        <retractable>0</retractable>
    </contact>
    <contact type="BOGEY" name="NOSE_LG">
        <location unit="IN">
            <x> 0 </x>
            <y> 0 </y>
            <z> -10 </z>
        </location>
        <static_friction> 1 </static_friction>
        <dynamic_friction> 0.8 </dynamic_friction>
        <rolling_friction> 0.02 </rolling_friction>
        <spring_coeff unit="LBS/FT"> 60 </spring_coeff>
        <damping_coeff unit="LBS/FT/SEC"> 10 </damping_coeff>
        <max_steer unit="DEG"> 360.0 </max_steer>
        <brake_group> NONE </brake_group>
        <retractable>0</retractable>
    </contact>
</ground_reactions>

```

```

<propulsion>
  <engine n="0" file="n3530">
    <location unit="M">
      <x> 0.0 </x>
      <y> 0.25 </y>
      <z> 0.0 </z>
    </location>
    <orient unit="DEG">
      <roll> 0.0 </roll>
      <pitch> 0.0 </pitch>
      <yaw> 0.0 </yaw>
    </orient>
    <feed>0</feed>
    <thruster file="9x6">
      <location unit="M">
        <x> 0.0 </x>
        <y> 0.25 </y>
        <z> 0.0 </z>
      </location>
      <orient unit="DEG">
        <roll> 0.0 </roll>
        <pitch> 0.0 </pitch>
        <yaw> 0.0 </yaw>
      </orient>
      <sense> -1 </sense>      <!--sentido de giro visto desde
atras 1:horario ; -1:antihorario-->
    </thruster>
  </engine>
  <engine n="1" file="n3530">
    <location unit="M">
      <x> 0.0 </x>
      <y> -0.25 </y>
      <z> 0.0 </z>
    </location>
    <orient unit="DEG">
      <roll> 0.0 </roll>
      <pitch> 0.0 </pitch>
      <yaw> 0.0 </yaw>
    </orient>
    <feed>1</feed>
    <thruster file="9x6">
      <location unit="M">
        <x> 0.0 </x>
        <y> -0.25 </y>
        <z> 0.0 </z>
      </location>
      <orient unit="DEG">
        <roll> 0.0 </roll>
        <pitch> 0.0 </pitch>
        <yaw> 0.0 </yaw>
      </orient>
      <sense> 1 </sense>
    </thruster>
  </engine>
</propulsion>

```

```

<flight_control name="FCS: herculesgii_alerones">
  <channel name="All">

```

```

<summer name="pitch-trim-sum">
  <input>fcs/elevator-cmd-norm</input>
  <input>fcs/pitch-trim-cmd-norm</input>
  <clipto>
    <min>-1</min>
    <max>1</max>
  </clipto>
</summer>

<aerosurface_scale name="Elevator Control">
  <input>fcs/pitch-trim-sum</input>
  <range>
    <min>-0.35</min>
    <max>0.3</max>
  </range>
  <output>fcs/elevator-pos-rad</output>
</aerosurface_scale>

<aerosurface_scale name="Elevator Normalized">
  <input>fcs/elevator-pos-rad</input>
  <domain>
    <min>-0.3</min>
    <max> 0.3</max>
  </domain>
  <range>
    <min>-1</min>
    <max> 1</max>
  </range>
  <output>fcs/elevator-pos-norm</output>
</aerosurface_scale>

<summer name="Roll Trim Sum">
  <input>fcs/aileron-cmd-norm</input>
  <input>fcs/roll-trim-cmd-norm</input>
  <clipto>
    <min>-1</min>
    <max>1</max>
  </clipto>
</summer>

<aerosurface_scale name="Left Aileron Control">
  <input>fcs/roll-trim-sum</input>
  <range>
    <min>-0.35</min>
    <max>0.35</max>
  </range>
  <output>fcs/left-aileron-pos-rad</output>
</aerosurface_scale>

<aerosurface_scale name="Right Aileron Control">
  <input>-fcs/roll-trim-sum</input>
  <range>
    <min>-0.35</min>
    <max>0.35</max>
  </range>
  <output>fcs/right-aileron-pos-rad</output>
</aerosurface_scale>

<aerosurface_scale name="Left aileron Normalized">
  <input>fcs/left-aileron-pos-rad</input>
  <domain>

```

```

        <min>-0.35</min>
        <max> 0.35</max>
    </domain>
    <range>
        <min>-1</min>
        <max> 1</max>
    </range>
    <output>fcs/left-aileron-pos-norm</output>
</aerosurface_scale>

<aerosurface_scale name="Right aileron Normalized">
    <input>fcs/right-aileron-pos-rad</input>
    <domain>
        <min>-0.35</min>
        <max> 0.35</max>
    </domain>
    <range>
        <min>-1</min>
        <max> 1</max>
    </range>
    <output>fcs/right-aileron-pos-norm</output>
</aerosurface_scale>

    <!--Derivas-->
    <summer name="rudder-sum">
    <input>fcs/rudder-cmd-norm</input>
    <clipto>
        <min>-1.0</min>
        <max> 1.0</max>
    </clipto>
    </summer>
    <aerosurface_scale name="Rudder Deg">
    <input>fcs/rudder-sum</input>
    <domain>
        <min>-1</min>
        <max> 1</max>
    </domain>
    <range>
        <min>-20.0</min>
        <max> 20.0</max>
    </range>
    <output>fcs/rudder-pos-deg</output>
    </aerosurface_scale>
    <pure_gain name="Rudder Position">
    <input>fcs/rudder-pos-deg</input>
    <gain>0.017453293</gain>
    <output>fcs/rudder-pos-rad</output>
    </pure_gain>

    </channel>
    </flight_control>

    <aerodynamics file="herculesgii_aero.xml"/>
</fdm_config>

```





UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**Documento**

**ANEXO V: HERCULES.AVL**

!\*\*\*\*\*

!AVL dataset for Hercules UAV

!\*\*\*\*\*

HERCULES

!Mach = V/Vs = 12.5/343 (m/s)

0.03644

!IYsym IZsym Zsym

0 0 0.0

!Sref Cref Bref

1.1534 0.4306 1.94

!Xref Yref Zref

0.254 0.0 0.0295

#----- ALA CENTRAL -----

SURFACE

CENTRAL WING

!Nchordwise Cspace Nspanwise Sspace

10 1.0 !96 1.0

COMPONENT

1

YDUPLICATE

0.0

ANGLE

2.0

SECTION

!Xle Yle Zle Chord Ainc Nspanwise Space

0.000 0.000 0.000 0.560 0. 6 1.0

NACA

3210

SECTION

!Xle Yle Zle Chord Ainc Nspanwise Space

0.000 0.050 0.000 0.560 0. 6 1.0

NACA

3210

SECTION

!Xle Yle Zle Chord Ainc Nspanwise Space

0.000 0.100 0.000 0.560 0. 6 1.0

NACA

3210

SECTION

!Xle Yle Zle Chord Ainc Nspanwise Space

0.000 0.150 0.000 0.560 0. 6 1.0

NACA

3210

#----- ALA CON ELEVONES -----

SURFACE

WING

!Nchordwise Cspace Nspanwise Sspace

10 1.0 !96 1.0

COMPONENT

1

YDUPLICATE

0.0

ANGLE

2.0

SECTION

!Xle Yle Zle Chord Ainc Nspanwise Space

0.000 0.1501 0.000 0.620 0. 6 1.0

NACA

3210

CONTROL

```

aileron  1.000 0.903  0.47 1. 0.03  -1
CONTROL
elevator 1.000 0.903  0.47 1. 0.03  +1
#-----
SECTION
!Xle    Yle    Zle    Chord  Ainc  Nspanwise  Space
0.023   0.200  0.002  0.603  0.122  6          1.0
NACA
3210
CONTROL
aileron  1.000 0.900  0.47 1. 0.03  -1
CONTROL
elevator 1.000 0.900  0.47 1. 0.03  +1
#-----
SECTION
!Xle    Yle    Zle    Chord  Ainc  Nspanwise  Space
0.047   0.250  0.003  0.585  0.244  6          1.0
NACA
3210
CONTROL
aileron  1.000 0.897  0.47 1. 0.03  -1
CONTROL
elevator 1.000 0.897  0.47 1. 0.03  +1
#-----
SECTION
!Xle    Yle    Zle    Chord  Ainc  Nspanwise  Space
0.070   0.300  0.005  0.568  0.366  6          1.0
NACA
3210
CONTROL
aileron  1.000 0.894  0.47 1. 0.03  -1
CONTROL
elevator 1.000 0.894  0.47 1. 0.03  +1
#-----
SECTION
!Xle    Yle    Zle    Chord  Ainc  Nspanwise  Space
0.093   0.350  0.007  0.550  0.488  6          1.0
NACA
3210
CONTROL
aileron  1.000 0.891  0.47 1. 0.03  -1
CONTROL
elevator 1.000 0.891  0.47 1. 0.03  +1
#-----
SECTION
!Xle    Yle    Zle    Chord  Ainc  Nspanwise  Space
0.117   0.400  0.009  0.533  0.610  6          1.0
NACA
3210
CONTROL
aileron  1.000 0.887  0.47 1. 0.03  -1
CONTROL
elevator 1.000 0.887  0.47 1. 0.03  +1
#-----
SECTION
!Xle    Yle    Zle    Chord  Ainc  Nspanwise  Space
0.140   0.450  0.010  0.516  0.732  6          1.0
NACA
3210
CONTROL
aileron  1.000 0.884  0.47 1. 0.03  -1

```

```

CONTROL
elevator 1.000 0.884 0.47 1. 0.03 +1
#-----
SECTION
!Xle Yle Zle Chord Ainc Nspanwise Space
0.163 0.500 0.012 0.498 0.854 6 1.0
NACA
3210
CONTROL
aileron 1.000 0.880 0.47 1. 0.03 -1
CONTROL
elevator 1.000 0.880 0.47 1. 0.03 +1
#-----
SECTION
!Xle Yle Zle Chord Ainc Nspanwise Space
0.187 0.550 0.014 0.481 0.976 6 1.0
NACA
3210
CONTROL
aileron 1.000 0.875 0.47 1. 0.03 -1
CONTROL
elevator 1.000 0.875 0.47 1. 0.03 +1
#-----
SECTION
!Xle Yle Zle Chord Ainc Nspanwise Space
0.210 0.600 0.016 0.464 1.098 6 1.0
NACA
3210
CONTROL
aileron 1.000 0.871 0.47 1. 0.03 -1
CONTROL
elevator 1.000 0.871 0.47 1. 0.03 +1
#-----
SECTION
!Xle Yle Zle Chord Ainc Nspanwise Space
0.233 0.650 0.017 0.446 1.220 6 1.0
NACA
3210
CONTROL
aileron 1.000 0.866 0.47 1. 0.03 -1
CONTROL
elevator 1.000 0.866 0.47 1. 0.03 +1
#-----
SECTION
!Xle Yle Zle Chord Ainc Nspanwise Space
0.256 0.700 0.019 0.429 1.341 6 1.0
NACA
3210
CONTROL
aileron 1.000 0.860 0.47 1. 0.03 -1
CONTROL
elevator 1.000 0.860 0.47 1. 0.03 +1
#-----
SECTION
!Xle Yle Zle Chord Ainc Nspanwise Space
0.280 0.750 0.021 0.411 1.463 6 1.0
NACA
3210
CONTROL
aileron 1.000 0.854 0.47 1. 0.03 -1
CONTROL

```

```

elevator  1.000 0.854  0.47 1. 0.03  +1
#-----
SECTION
!Xle     Yle     Zle     Chord  Ainc  Nspanwise  Space
0.303    0.800    0.023    0.394  1.585    6          1.0
NACA
3210
CONTROL
aileron  1.000 0.848  0.47 1. 0.03  -1
CONTROL
elevator 1.000 0.848  0.47 1. 0.03  +1
#-----
SECTION
!Xle     Yle     Zle     Chord  Ainc  Nspanwise  Space
0.326    0.850    0.024    0.377  1.707    6          1.0
NACA
3210
CONTROL
aileron  1.000 0.841  0.47 1. 0.03  -1
CONTROL
elevator 1.000 0.841  0.47 1. 0.03  +1
#-----
SECTION
!Xle     Yle     Zle     Chord  Ainc  Nspanwise  Space
0.350    0.900    0.026    0.359  1.829    6          1.0
NACA
3210
CONTROL
aileron  1.000 0.833  0.47 1. 0.03  -1
CONTROL
elevator 1.000 0.833  0.47 1. 0.03  +1
#-----
SECTION
!Xle     Yle     Zle     Chord  Ainc  Nspanwise  Space
0.373    0.950    0.028    0.342  1.951    6          1.0
NACA
3210
CONTROL
aileron  1.000 0.825  0.47 1. 0.03  -1
CONTROL
elevator 1.000 0.825  0.47 1. 0.03  +1
#-----
SECTION
!Xle     Yle     Zle     Chord  Ainc  Nspanwise  Space
0.382    0.970    0.029    0.335  2.000    6          1.0
NACA
3210
CONTROL
aileron  1.000 0.821  0.47 1. 0.03  -1
CONTROL
elevator 1.000 0.821  0.47 1. 0.03  +1
#----- ALA VERTICAL CON PROFUNDIDAD -----
SURFACE
RUDDER
!Nchordwise  Cspace                Nspanwise                Sspace
10            1.0                !96                1.0
COMPONENT
1
YDUPLICATE
0.0
TRANSLATE

```

0.373 0.350 0.000

#----

SECTION

!Xle	Yle	Zle	Chord	Ainc	Nspanwise	Space
0.000	0.000	0.000	0.220	0.	6	1.0

CONTROL

rudder 1.000 1.000 0. 0. 1. -1.0

#----

SECTION

!Xle	Yle	Zle	Chord	Ainc	Nspanwise	Space
0.020	0.000	0.020	0.246	0.	6	1.0

CONTROL

rudder 1.000 0.814 0. 0. 1. -1.0

#----

SECTION

!Xle	Yle	Zle	Chord	Ainc	Nspanwise	Space
0.035	0.000	0.035	0.265	0.	6	1.0

CONTROL

rudder 1.000 0.698 0. 0. 1. -1.0

#----

SECTION

!Xle	Yle	Zle	Chord	Ainc	Nspanwise	Space
0.040	0.000	0.040	0.260	0.	6	1.0

CONTROL

rudder 1.000 0.692 0. 0. 1. -1.0

#----

SECTION

!Xle	Yle	Zle	Chord	Ainc	Nspanwise	Space
0.090	0.000	0.090	0.210	0.	6	1.0

CONTROL

rudder 1.000 0.619 0. 0. 1. -1.0

#----

SECTION

!Xle	Yle	Zle	Chord	Ainc	Nspanwise	Space
0.140	0.000	0.140	0.160	0.	6	1.0

CONTROL

rudder 1.000 0.500 0. 0. 1. -1.0

#----

SECTION

!Xle	Yle	Zle	Chord	Ainc	Nspanwise	Space
0.150	0.000	0.150	0.150	0.	6	1.0

CONTROL

rudder 1.000 0.467 0. 0. 1. -1.0

#----- WINGLET -----

SURFACE

WINGLET

!Nchordwise	Cspace	!96	Nspanwise	Sspace
10	1.0		1.0	

COMPONENT

1

YDUPLICATE

0.0

TRANSLATE

0.382 0.970 0.029

#----

SECTION

!Xle	Yle	Zle	Chord	Ainc	Nspanwise	Space
0.313	0.000	-0.050	0.000	0.	6	1.0

#----

SECTION

!Xle	Yle	Zle	Chord	Ainc	Nspanwise	Space
------	-----	-----	-------	------	-----------	-------

```
0.000 0.000 0.000 0.313 0. 6 1.0
#----
SECTION
!Xle Yle Zle Chord Ainc Nspanwise Space
0.267 0.000 0.050 0.046 0. 6 1.0
#----- BODY -----
BODY
FUSELAGE !body name string
30 1.0 !Nbody Bspace
#
TRANSLATE
0.0 0.0 -0.0449514
#
BFIL
herc_body.dat
```



UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**Documento**

**ANEXO VI: CÁLCULO DE INERCIAS**



**Índice de ilustraciones**

Ilustración 1: Péndulo bifilar (1) .....

Ilustración 2: Péndulo bifilar (2) .....

Ilustración 3: Posición para la inercia del eje x del UAV (1) .....

Ilustración 4: Posición para la inercia del eje x del UAV (2) .....

Ilustración 5: Posición para la inercia del eje y del UAV.....

Ilustración 6: Posición para la inercia del eje z del UAV (1) .....

Ilustración 7: Posición para la inercia del eje z del UAV (2) .....

**Índice de tablas**

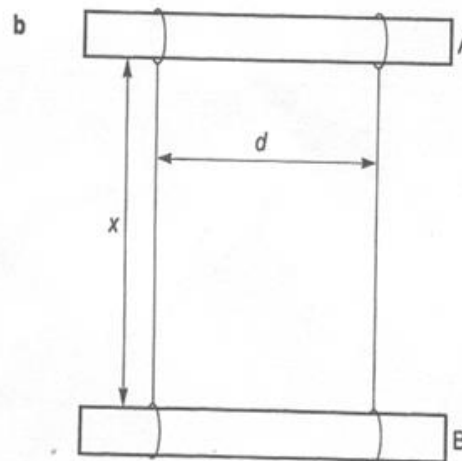
Tabla 1: Medidas de experimento de péndulo bifilar .....

Las inercias reflejan la distribución de masa de un cuerpo o de un sistema de partículas en rotación. En nuestro caso, las rotaciones del UAV en cualquiera de los 3 ejes principales de rotaciones están enormemente influenciadas por las inercias, así que es una característica fundamental que rige el comportamiento del UAV e influye de manera indiscutible en el vuelo.

En un cuerpo complejo como es un UAV, las ecuaciones teóricas no son suficientes para calcular las inercias, por lo que hay que recurrir a un método experimental, que el éste proyecto fue el método del péndulo bifilar.

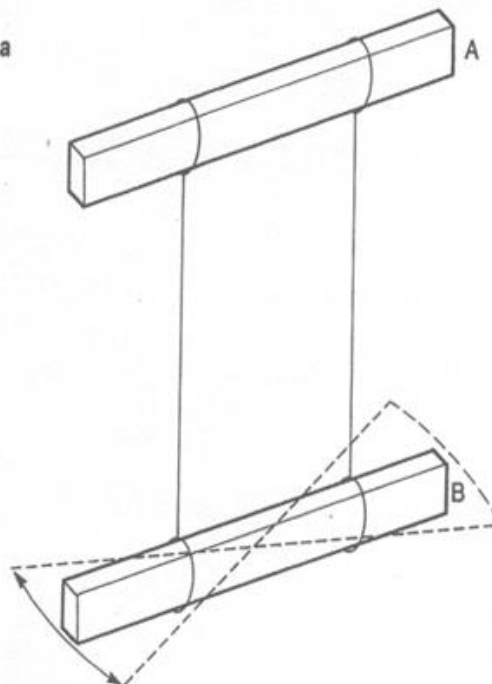
Para construir un péndulo bifilar y hallar las inercias seguimos los siguientes pasos:

1. Se suspende el UAV de una estructura mediante dos hilos paralelos e iguales, para que el dron quede completamente horizontal.



*Péndulo bifilar (1)*

2. Se gira el UAV en el eje vertical como en la imagen:



*Péndulo bifilar (2)*

3. Se cuentan 10 oscilaciones y se mide el tiempo para las 3 posiciones.
4. Se halla el período mediante la ecuación:

$$T = 1/\text{frecuencia}$$

en donde:

$$\text{frecuencia} = \text{oscilaciones}/\text{tiempo}$$

5. Una vez conocido el período se procede a calcular la inercia en el eje correspondiente mediante la ecuación:

$$I = (m \cdot g \cdot T^2 \cdot b^2)/(4 \cdot \pi^2 \cdot L)$$

En donde:

m: masa en kg

g: aceleración de la gravedad en  $\text{m/s}^2$

T: período en s

b: distancia entre el centro de masas y los hilos en m

L: longitud de los hilos en m

Datos:

- masa = 3 kg
- gravedad =  $9,81 \text{ m/s}^2$

Las medidas obtenidas en las pruebas del experimento para cada eje fueron:

Datos	Ixx	Iyy	Izz
b(m)	0,57	0,5	0,57
L(m)	2,38	1,59	2,33
Tiempo(s)	22,34	10,69	23,06
Oscilaciones	10	10	10
Frecuencia (Hz)	0,4476	0,9355	0,4337
Período (s)	2,234	1,069	2,306
I ( $\text{kg}\cdot\text{m}^2$ )	1,5956	0,4208	1,7366

#### *Medidas de experimento de péndulo bifilar*

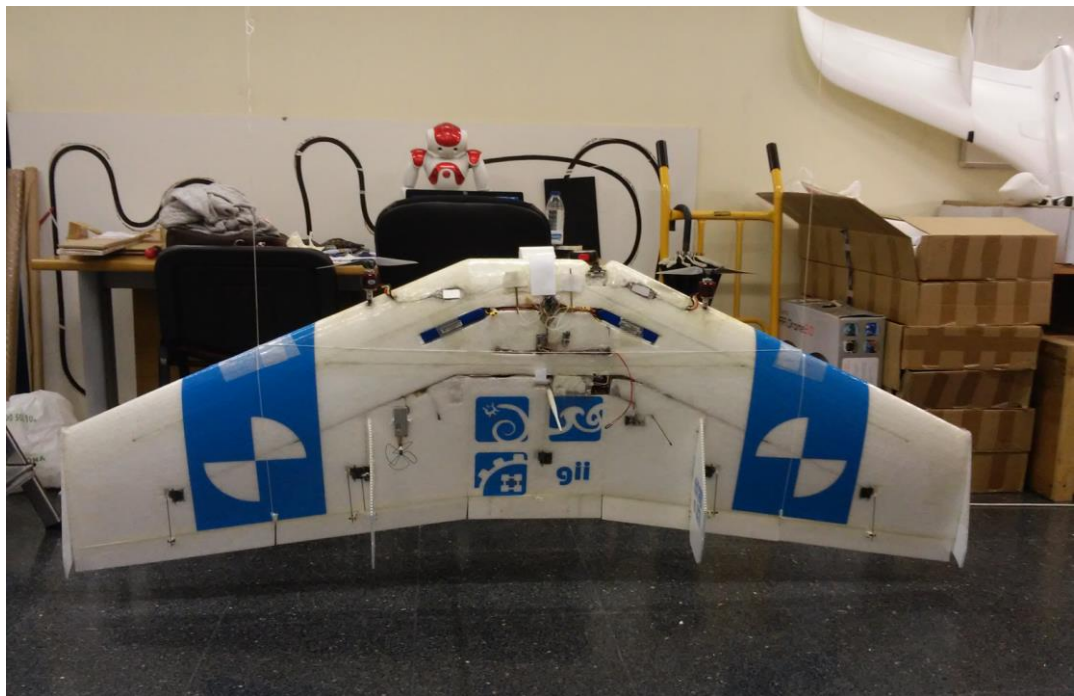
El eje en el que más fácilmente puede realizar un giro es el de menor momento de inercia, el eje Y, debido a que las masas principales (baterías y motores) están más cerca del eje de rotación Y que pasa por el centro de masas que de los otros 2. Por otro lado, la inercia en el eje X es ligeramente inferior a la inercia en el eje Z, esto se debe a que las masas principales se encuentran más cerca del eje de rotación X del UAV que del eje Z, por lo tanto, el esfuerzo necesario para iniciar o para detener la rotación en el eje Z será mayor que en el eje X, y mucho mayor que en el eje Y.

A continuación, se muestran las imágenes del montaje del experimento para realizar las pruebas y poder medir la inercia en cada eje:

- **Ixx:**

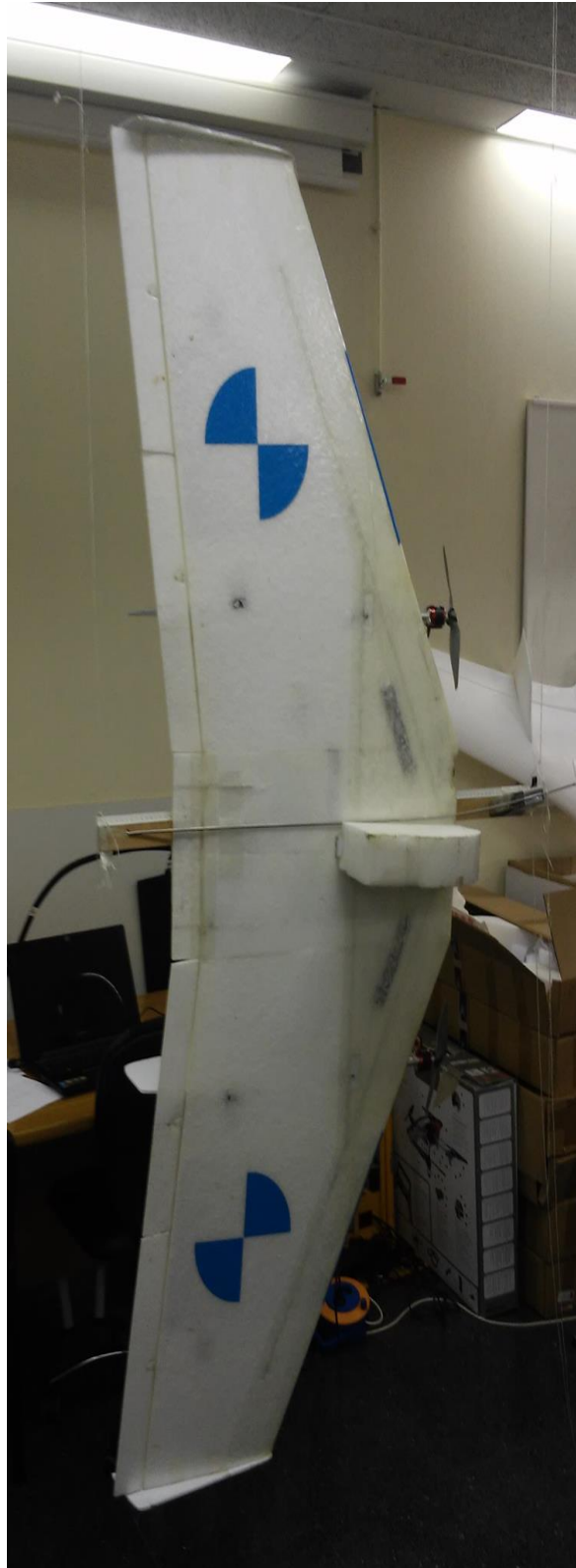


*Posición para la inercia del eje x del UAV (1)*



*Posición para la inercia del eje x del UAV (2)*

- **Iyy:**



*Posición para la inercia del eje y del UAV*

- **lzz:**



*Posición para la inercia del eje z del UAV (1)*



*Posición para la inercia del eje z del UAV (2)*



UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

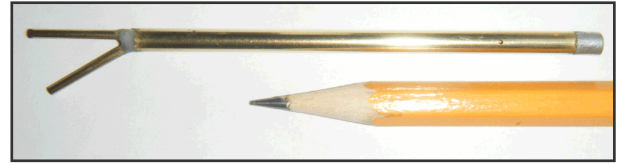
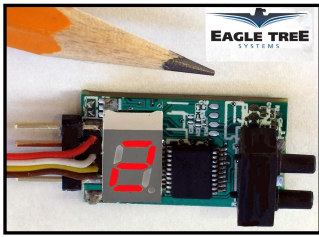
*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**Documento**

**ANEXO VII: MANUAL INSTRUCCIONES AIRSPEED MICROSENSOR**



## Instruction Manual for the Airspeed MicroSensor V3 Document Version 2.0

Thank you for your purchase! This instruction manual will guide you through the installation and operation of your Airspeed MicroSensor V3 (the MicroSensor). **Please read the entire manual carefully before proceeding.** If, after you read the manual, you have further questions or problems, see the Support page on <http://www.eagletreesystems.com> for additional information, or email us at [support@eagletreesystems.com](mailto:support@eagletreesystems.com). Please visit our support web page for the full color, electronic version of this manual which may be updated if changes were made after printing, or if you want to view the manual on your computer.

### Intended Uses

The Airspeed MicroSensor is intended to be used exclusively for recreational purposes in model planes, boats and cars. Other uses are not supported. Further, using the MicroSensor in situations where its use or failure could result in loss of life, bodily injury or property damage is expressly prohibited.

### What the MicroSensor Does

The MicroSensor uses a Prandtl style pitot-static tube to measure airspeed, just as full sized planes do. When used standalone, the MicroSensor displays your maximum speed on the built-in 7 segment LED display. The maximum speed continues to be updated and displayed on the LED, until you turn power off and on the MicroSensor. When power is turned off and on, the maximum speed from the last flight is displayed, and now the MicroSensor is ready to record your next maximum speed (even if it is lower than the previous maximum speed).

Additionally, the MicroSensor can be connected to your eLogger (any version) to provide airspeed data for your entire flight. When connected to the eLogger, airspeed can be displayed and graphed using the eLogger's Windows software.

**IMPORTANT:** It is extremely unlikely that the installation of the MicroSensor will affect your model's radio range or control. But, as always after making an electronics change to your model, it is very important that you range and function test your model once the MicroSensor is installed to ensure that there is no impact on your system.

### Packing List

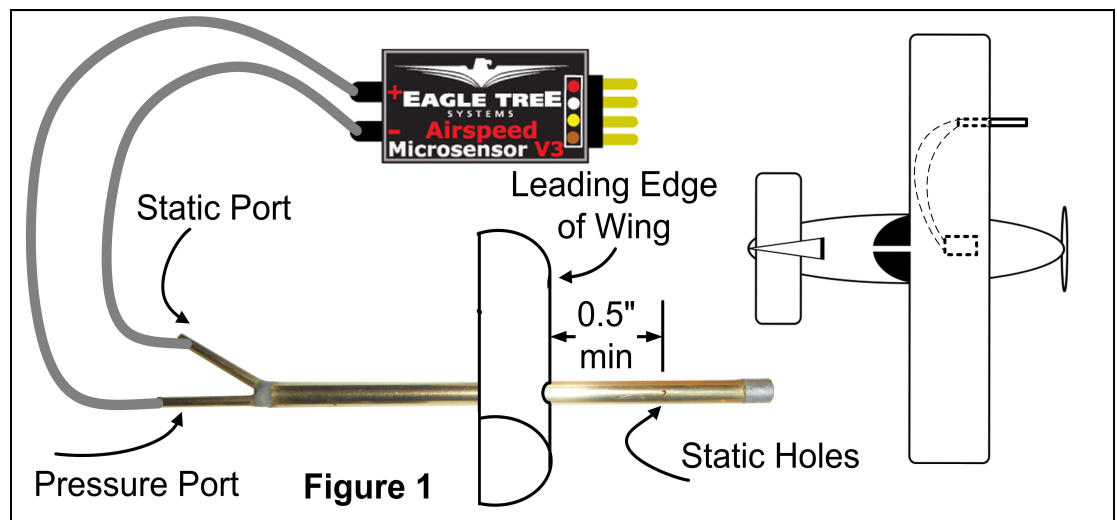
Your package should include the following: The MicroSensor, a machined brass and aluminum Prandtl style pitot-static tube, the Standalone Cable, approximately 3 feet (1 meter) of small diameter silicon hose, and a printed version of this manual. NOTE: Additional Pitot Tube kits are available for purchase on our website, for use with multiple models.

### Installing the Pitot Tube and MicroSensor in your Model

Using two lengths of the included small diameter silicon tube, the pressure and static connections of the pitot tube connect to the "+" and "-" ports of the MicroSensor, respectively, as shown in Figure 1. It is best to mount the pitot tube in your model first, then determine where you will mount the MicroSensor, and then cut the two lengths of silicon tube so they reach between the two.

Follow these guidelines when mounting the pitot tube:

1. The pickup end of the pitot tube (the silver colored tip) should be pointing toward the direction of the model's travel. While best results will be obtained if the pitot tube is perfectly aligned with the direction of travel in both axes, the "Prandtl" design of the tube will compensate





somewhat for higher angles of attack.

- The static holes on the pitot tube (shown on Figure 1) should extend at least 1/2" (13mm) past the wing's leading edge, or past any other obstructions - the farther out, the better. This is to ensure that the static holes and pitot pickup are in undisturbed air.
- For prop planes, it's important that the tube be placed so that it is not directly in the plane's prop-wash, which will result in erroneous readings. The best place to install the tube is on the leading edge of the wing several inches out from the fuselage, as shown in Figure 1.
- For jets, gliders, or "pusher" prop planes, the nose cone often provides a perfect mounting location.

After Pitot Tube installation, it is recommended that you glue or otherwise attach the silicon hose to the airframe, to reduce the chance that hose vibration or movement could cause erroneous readings.

The MicroSensor circuit board can be mounted anywhere in the model. If you wish to read the LED without removing the wing, a small rectangular hole can be cut at the mounting location of the circuit board, and the board mounted with tape or Velcro, so that the LED is visible at all times.

## Using the MicroSensor in Standalone Mode

### Powering the MicroSensor in Standalone Mode

For standalone mode (not connected to an eLogger), connect the Standalone Cable to the gold pins of the MicroSensor, as shown in Figure 2. Note that the RED wire of the standalone cable corresponds with the red dot on the label. The polarity of the connection is as follows:  
 RED = Positive (Vdd)  
 BLACK = Negative (ground)

The JR/Universal servo end of the Standalone Cable connects to a spare Receiver channel or small battery. Note that the voltage must be between 3V and 16V. **Do not exceed 16V!**

**IMPORTANT: Ensure that you connect the Standalone Cable with the correct polarity, or the MicroSensor could be damaged, voiding the warranty!**

**Note that in Standalone Mode, the 4 wire cable built into the MicroSensor must NOT be connected to anything!**

### Configuring Metric vs English Units in Standalone Mode

When powered on, the LED readout will briefly display "0" if the unit is configured for English (MPH), or will display a "1" if configured for metric (KPH). The factory default configuration is MPH.

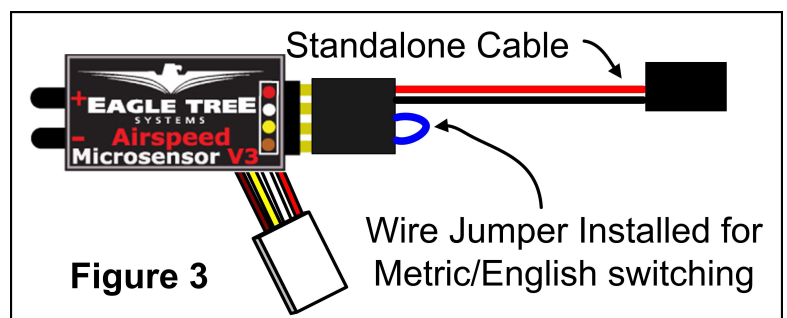
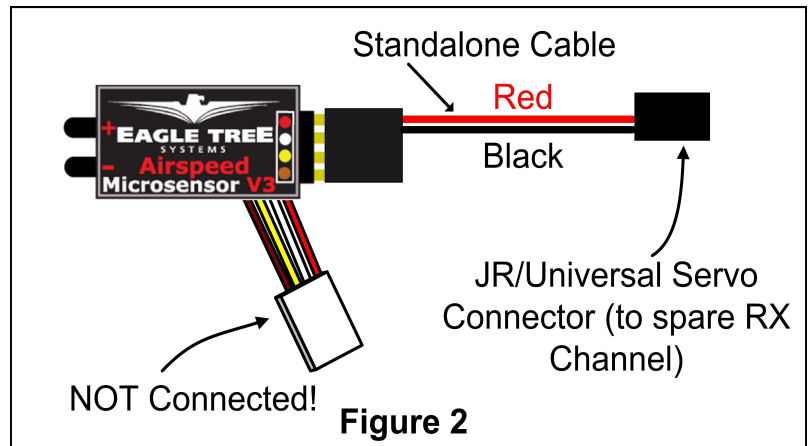
To switch between English and metric units, connect the Standalone Cable as described above, but don't power the MicroSensor. Then, connect a small piece of wire between the unconnected pins of the Standalone Cable's 4 pin connector, corresponding to the brown and yellow dots on the label, as shown in Figure 3. This creates a jumper between the brown and yellow labeled pins.

Then, power the MicroSensor via the Standalone Cable. A "0" will then flash a few times to indicate that English units are configured, or a "1" will flash if Metric units are configured. If the wrong units are configured, just disconnect the power from the MicroSensor and reconnect it.

Once the desired units are configured, disconnect from power and remove the wire jumper. Then, when reconnected to power, the desired units should be displayed briefly ("0" or "1").

### Reading Airspeed in Standalone Mode

Copyright © 2008 Eagle Tree Systems, LLC  
<http://www.eagletreesystems.com>



When the MicroSensor is powered on, after the Units setting is displayed (“0” or “1”), the highest speed attained during your last flight is repeatedly displayed, one digit at a time, on the LED. For example, if the highest speed attained was 120MPH, “1 – 2 – 0” will flash repeatedly on the LED, with a pause (blank LED) after the speed is completely displayed, but before it is displayed again.

**To reset the max trigger, just turn power off and on to the MicroSensor.** Then, after landing from your next flight, the highest speed attained during that flight is displayed automatically, overwriting the highest speed from your prior flight. **Note that a speed of at least 9 MPH (15 KPH) must be attained before a new highest speed is displayed.**

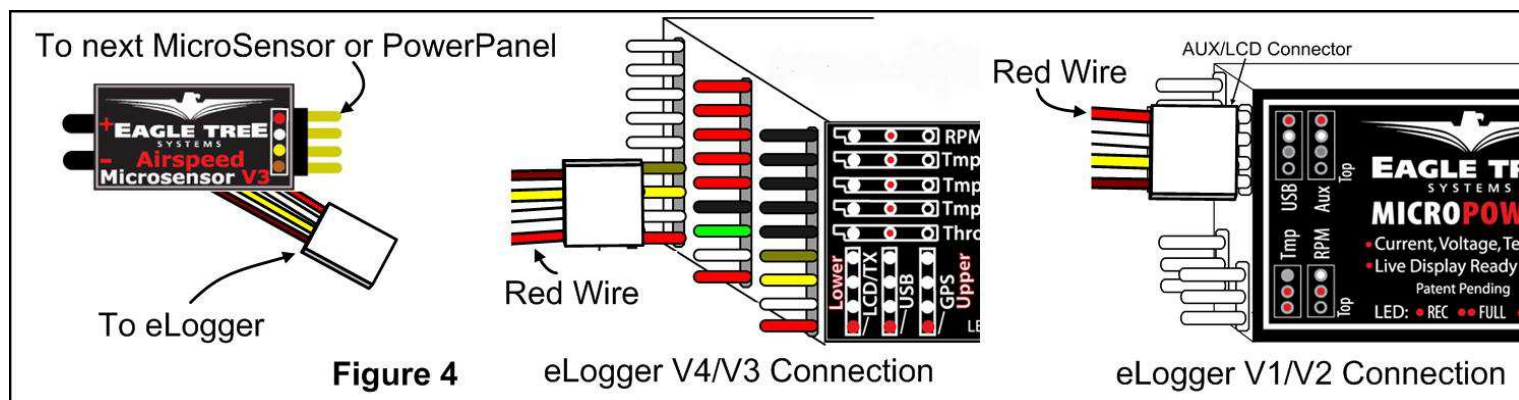
The highest speed of your last flight is automatically saved, so that when you turn the unit back on, that speed is displayed.

For example, if you fly and reach a max speed of 100 MPH, when you land “1 – 0 – 0” will be displayed. Then, when you turn power off and on, “1 – 0 – 0” will again be displayed. If you then fly again and reach a speed of 50 MPH, “5 – 0” will be displayed when you land.

## Using the MicroSensor with your eLogger

### Windows Application and Firmware Update

To use the MicroSensor, you must update to **Eagle Tree Windows Application version 6.77** or higher. To update, download the latest application from the support page of our website, located at <http://eagletreesystems.com/Support/apps.htm>. After downloading and installing the Application, the firmware of your MicroPower eLogger will need to be updated. To upgrade your firmware, just choose “Tools, Firmware Control” and click the Update button.



### Connecting the MicroSensor to the eLogger V4/V3

The MicroSensor plugs into the “OSD/LCD” port of your eLogger V4, or the “LCD/TX” port of your eLogger V3, as shown in Figure 4. The MicroSensor plugs into the “LCD/TX” port of your eLogger V3, as shown in Figure 4. If you have a PowerPanel or other MicroSensors, those can “daisy chain” connect to the pins on the other side of your MicroSensor, with the polarity as indicated on the MicroSensor label. **NOTE: The Standalone Cable must not be used when connected to the eLogger!**

### Connecting the MicroSensor to the eLogger V1/V2

The MicroSensor plugs into the “Aux” or “LCD” port of your eLogger V1/V2, as shown in Figure 4. If you have a PowerPanel or other MicroSensors, those can “daisy chain” connect to the pins on the other side of your MicroSensor, with the polarity as indicated on the MicroSensor label. **NOTE: The Standalone Cable must not be used when connected to the eLogger!**

### Configuring the MicroSensor with the Windows Application

If you have not already done so, set up the Recorder software as described in your instruction manual. Then, choose one or more of the MicroSensor options below:

#### Logging Airspeed

To log airspeed, just click “Hardware, Choose Parameters to Log in the Recorder” and check the “Airspeed” box.

#### Displaying Airspeed in the Windows Application

To display the Airspeed Gauge and/or Numeric Airspeed Display, click “Software, Choose Instruments to Display on the PC Screen” and check the “Airspeed Indicator Gauge,” and/or “Numeric Speed” boxes.

#### Graphing Airspeed

To graph airspeed, click “Chart, 2D Chart,” select either the Left Y Axis or Right Y Axis, and select “Speed” for graphing.

#### Displaying Airspeed on the PowerPanel

Select “Hardware, Configure PowerPanel Display”, and choose “Speed” for PowerPanel display.

## ***Using the MicroSensor with your own Firmware/Microcontroller***

Please see this document for information on using the MicroSensor with your own firmware:

<http://www.eagletreesystems.com/support/manuals/microsensor-i2c.pdf>

### **Troubleshooting**

Below is a list of problems that may be encountered, and steps to remedy them. If your particular issue is not addressed by the below, see the Support page on <http://eagletreesystems.com> or email [support@eagletreesystems.com](mailto:support@eagletreesystems.com). Include a full description of your problem, your PC configuration, your Eagle Tree hardware and software versions, and any other relevant information.

**Issue:** Airspeed doesn't vary in my recordings

#### **Solutions:**

- Ensure that the MicroSensor is connected correctly to the MicroPower
- Ensure that you are logging airspeed, under "Tools, Choose Parameters to Log in the Recorder"
- Ensure that the Pitot Tube is connected

**Issue:** My plane shows an airspeed variation during warm-up on the ground as I rev the throttle.

**Solution:** This problem can occur when the Pitot Tube's opening is being hit by prop-wash. The solution is to move the tube farther away from propeller airflow. Note that slight variations in speed is normal when stationary.

### **Specifications (Approximate)**

- Measures airspeed from 9 MPH to 350 MPH (15KPH to 563KPH), with 1 MPH (1 KPH) Resolution
- Power input (Standalone Mode) – 3V to 16V
- Circuit Board: Weight 4 grams (0.15 oz), dimensions 28 mm x 16 mm x 10 mm (1.1" x 0.62" x 0.4")
- Pitot Tube: Weight 3 grams (0.1 oz), length 80mm (3.2"), diameter 4 mm (0.16")
- Pitot Tube Hose: Silicon, clear, 3 feet (1 meter), 1.5mm ID, 2.5mm OD
- Precalibrated – no user calibration required
- Advanced Temperature Compensation
- Metric or English units

### **Limited Warranty**

Eagle Tree Systems, LLC, warrants the MicroSensor to be free from defects in materials and workmanship for a period of one (1) year from the date of original purchase. This warranty is nontransferable. If your unit requires warranty service during this period, we will replace or repair it at our option. Shipping cost to us is your responsibility. To obtain warranty service, email [support@eagletreesystems.com](mailto:support@eagletreesystems.com) for further instructions.

This limited warranty does not cover:

- The Software. See the Software license agreement for more information on Software restrictions.
- Problems that result from:
  - External causes such as accident, abuse, misuse, or problems with electrical power
  - Servicing not authorized by us
  - Usage that is not in accordance with product instructions
  - Failure to follow the product instructions

THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS, AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM STATE TO STATE (OR JURISDICTION TO JURISDICTION). OUR RESPONSIBILITY FOR MALFUNCTIONS AND DEFECTS IN HARDWARE IS LIMITED TO REPAIR AND REPLACEMENT AS SET FORTH IN THIS WARRANTY STATEMENT. ALL EXPRESS AND IMPLIED WARRANTIES FOR THE PRODUCT, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES AND CONDITIONS OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED IN TIME TO THE TERM OF THE LIMITED WARRANTY PERIOD AS DESCRIBED ABOVE. NO WARRANTIES, WHETHER EXPRESS OR IMPLIED, WILL APPLY AFTER THE LIMITED WARRANTY PERIOD HAS EXPIRED. SOME STATES DO NOT ALLOW LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THIS LIMITATION MAY NOT APPLY TO YOU.

WE DO NOT ACCEPT LIABILITY BEYOND THE REMEDIES PROVIDED FOR IN THIS LIMITED WARRANTY OR FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, ANY LIABILITY FOR THIRD-PARTY CLAIMS AGAINST YOU FOR DAMAGES, FOR PRODUCTS NOT BEING AVAILABLE FOR USE, OR FOR LOST DATA OR LOST SOFTWARE. OUR LIABILITY WILL BE NO MORE THAN THE AMOUNT YOU PAID FOR THE PRODUCT THAT IS THE SUBJECT OF A CLAIM. THIS IS THE MAXIMUM AMOUNT FOR WHICH WE ARE RESPONSIBLE. SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU.

Copyright © 2008 Eagle Tree Systems, LLC

<http://www.eagletreesystems.com>



UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

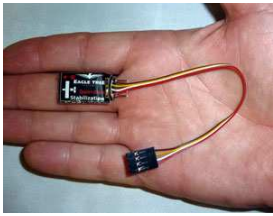
*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**Documento**

**ANEXO VIII: MANUAL INSTRUCCIONES GUARDIAN STABILIZATION**



# Instruction Manual for Guardian™ Stabilization Expander

Document Version 1.9

Corresponds to Eagle Tree Software Version 10.01 or higher

## Contents

Introduction .....	1
Connection and Mounting .....	2
Important Differences in OSD Pro Configuration when using Stabilization .....	2
Guardian Stabilization On-screen Menus .....	3
Guardian Configuration .....	4
The Servo Analysis Wizard.....	4
Reset Cruise Sticks/Throtl.....	4
Tuning Stabilization Gains.....	4
Stabilization Theory.....	4
Stabilization “Master Gain” Knob.....	5
Displaying the Master Gain Onscreen.....	5
Oscillations .....	5
Additional Features.....	5
Center-Stick Only Stabilization.....	5
Heading Hold (Recommended).....	6
Preflight.....	6
First Flight.....	6
Troubleshooting.....	6

## Introduction

Thank you for your purchase! This instruction manual will guide you through the installation and operation of your Guardian Stabilization Expander. The latest version of this manual is available in the Product Manuals section of the Support tab on <http://www.eagletreesystems.com>. The online manual includes any updates that were made after your Guardian was purchased. Please read the entire manual carefully before proceeding. If, after you read the manual, you have further questions or problems, see the “Have Questions or Feedback?” section below.

### What the Guardian Does

When used with the Eagle Tree OSD Pro™, the Guardian provides full featured inertial stabilization, and an on-screen Artificial Horizon display. No additional sensors are required.

The Guardian requires no warm-up period after your model is powered. The Guardian’s advanced features make it easy for you to perform aerobatic maneuvers (if you are capable!) and immediately return to stabilized flight when desired.

The Guardian is a stabilizer, **not a stall-preventer!** In some airframes, stalling can create unrecoverable conditions, such as spins, that may result in a crash.

### Intended Uses

The Guardian is intended to be used exclusively for recreational purposes in model planes, boats and cars. Any other use is not supported.

### Packing List

Your package should include the following: The Guardian, and a printed version of this manual.

### Other Things You Need

To utilize the Guardian, you will need the OSD Pro system with an eLogger V3 or V4. NOTE: The Guardian is NOT compatible with our Flight Data Recorder products! Though stabilization will work without them, a video camera and video transmitter on your model, and a video receiver and video display on the ground are normally required. Many Eagle Tree FPV dealers sell the required video equipment. A list of our FPV dealers can be found on our website at <http://www.eagletreesystems.com/OSD/OSD-Pro.htm>.

## Steps to Follow

Installation and use of your Guardian should be quite easy and enjoyable if you follow these few steps:

1. First, if you have not done so already, read through the manuals for your OSD Pro, eLogger, GPS Expander, and any other Eagle Tree sensors, to familiarize yourself with the connection and function of these devices, and set this equipment up.
2. Read through this manual to understand the warnings, determine the installation and setup sequence, pre-flight tests, etc., of the Guardian.
3. If anything is unclear, see the “Have Questions or Feedback?” section below.
4. Range test and enjoy!

## Have questions or Feedback?

Eagle Tree is committed to providing great customer service. If you've read the manual and something is not clear, just ask. We'd much prefer to take the time to answer your questions, rather than having you waste your valuable time struggling with an issue.

To get help, visit the Eagle Tree Guardian support thread at <http://www.rcgroups.com/forums/showthread.php?t=1399687>. Chances are someone has posted a solution to your problem already. If not, posting your problem there will get a very quick response from the Eagle Tree community.

If you prefer to not post on the forum, or you feel there is a problem with your Eagle Tree hardware, please open a support ticket with us at <http://ticket.eagletreesystems.com> and we will respond to your support ticket as soon as we can (normally 1-3 business days). Note that you may receive an answer to your question more quickly from our forum above, since many other Eagle Tree customers also monitor the forum. Note that when you create a support ticket, you will be emailed a link that will let you check the status of the ticket. If you do not receive the email, this most likely means that a spam filter is intercepting emails from Eagle Tree.

Also Eagle Tree greatly values your feedback on how we can improve our products. To leave us feedback for a new feature request or improvement, either post the feedback on our support thread above, create a support ticket with your feedback, or send feedback at <http://www.eagletreesystems.com/Feature/feature.html>

## Connection and Mounting

### Connecting the Guardian to the eLogger V4 or V3

The Guardian plugs into the "LCD/OSD" or "LCD/TX" port of your eLogger V4 or V3, as shown in Figure 3, or it can connect to the 4 pin expansion port of the OSD Pro. If you have a PowerPanel or other MicroSensors, those can "daisy chain" connect to the pins on the side of your Guardian, with the polarity as indicated on the Guardian label. Note that the Guardian can connect anywhere in the chain of sensors. Once the Guardian has been powered, its status light should begin blinking rapidly to indicate correct operation.

### Connecting the Servos

Connect your receiver's Aileron output (or Rudder receiver output, if the plane does not have Ailerons) and the Elevator receiver output to the OSD Pro's Aileron and Elevator servo inputs. Then, connect your servos to their corresponding OSD Pro servo outputs.

Note that when using the Guardian, the ailerons should be routed through the OSD Pro if the model has both ailerons and rudder, for proper stabilization and RTH.

**Note that this is a change from the recommendation that the rudder be routed through the OSD Pro, when the Guardian is not present.**

### Windows Software and Firmware Update

To use the Guardian, you must update your software to **Eagle Tree Windows Software version 9.76** or later. To update, download the latest software from the support page of our website, located at <http://eagletreesystems.com/Support/apps.htm>. After downloading and installing the software, the firmware of your Guardian may need to be updated. To upgrade your firmware, connect your eLogger to USB, and just click "Hardware, Firmware Control" and click the Update button for the Guardian if the present firmware version is a higher number than the firmware stored in your Guardian.

### Mounting the Guardian in your Model

Mount the Guardian in your model, placing it so the label is facing up and the "airplane" drawing on the label corresponds to the orientation of your aircraft. A properly mounted sensor will be perfectly flat with its front facing the direction of flight, when the aircraft is in level flight.

We recommended mounting the Guardian with a piece of two-sided foam tape or Velcro to hold it firmly to your airframe while reducing vibrations. To reduce the risk of the Guardian coming loose, tape, adhesive, or a mechanical restraint may be used.

Mounting the Guardian flat relative to level flight is important. However, minor mounting offsets can be compensated for by invoking your OSD Pro menu and using the "Configure Guardian Stabilizer" -> "Reset Level Flight" option described below.

**IMPORTANT! When viewing the AHJ through the OSD Pro as provided by the Guardian Expander after resetting level flight, the line should closely follow the horizon as the plane is pitched and rolled. If the AHJ appears to be moving in the opposite direction of the horizon in the pitch and roll axes, then you likely have the Guardian Expander mounted backwards. If just the pitch axis is moving in the incorrect direction as the plane is moved, then the Guardian Expander may have been mounted upside down. Ensure that the label is facing towards the sky and the 4-conductor wire is pointing towards the tail of your model.**

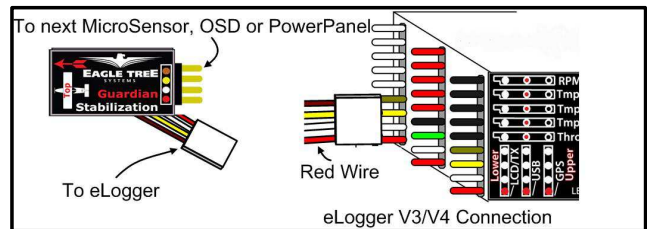
### Vibrations

The Guardian is able to reject most levels of vibration. However, very strong vibrations can impact stability. To check for this issue after mounting the sensor, perform a vibration test by securing your model to a test bench or stand and ramping up the throttle slowly while watching the AHJ on your OSD Pro. If the AHJ begins to pitch or roll significantly during this test, remount your Guardian sensor further away from the motor, wires and any other sources of vibration or electromagnetic interference.

## Important Differences in OSD Pro Configuration when using Stabilization

There are a few differences in how you configure the OSD Pro, when stabilization is enabled vs disabled. These are described elsewhere in the document in more detail:

- When running the Servo Analysis Wizard, full elevator and aileron stick deflection should be given, rather than partial stick deflection. Partial stick deflection is recommended when stabilization is disabled, to limit RTH servo deflections.
- If your model has both ailerons and a rudder, the ailerons should be routed through the OSD Pro when stabilization is turned on, rather than rudder.



- In Elevelon mode, with stabilization enabled, the Guardian performs the servo mixing, so you must turn off mixing in your radio. **But, if you disable stabilization, the OSD Pro will again expect mixed inputs from your radio. Never turn stabilization off or on during flight if you have an elevelon model! Use the stabilization gain setting instead.**
- If you have previously configured RTH with your OSD Pro, it is recommended that you return all RTH gains/limits to their default values when stabilization is enabled. Note that you may still need to tune the RTH gains/limits with the Guardian installed, but the defaults should work well with most models.

## Guardian Stabilization On-screen Menus

Guardian configuration is done via the On-screen OSD Pro menus. Please see the OSD Pro manuals for more information on accessing the menus.

Here is a description of each menu item that may need to be adjusted for the Guardian. Note that additional information for each menu item is provided in the following manual sections. Note also that the default values are shown in parentheses and where relevant, the range is displayed as (min-max: default).

### Configure Radio Control Menu

- **Reset Cruise Sticks/Throtl:** recaptures the model's aileron/rudder, elevator, and throttle settings for level flight. These settings are captured during the Servo Analysis Wizard, but need to be recaptured if you have re-trimmed your plane. **NOTE: the throttle needs to be set at "cruising speed" position when this option is selected, if the RTH feature is enabled!**
- **Run Servo Analysis Wizard:** lets the Guardian learn about your servo directions and extents. See the Guardian Configuration section below for more information.
- **Choose Model Wing Type (Fixed):** This option lets you choose whether your model uses standard or elevelon control. Note that when the wing type is set to Elevelon, and Guardian stabilization is enabled, the Guardian will do the servo mixing. So, your radio's servo mixing must be disabled. But, if you later turn off stabilization, the Guardian will no longer do the mixing! **WARNING: Never change this option during flight!**
- **Stabilization / Elevelon Mixing On? (No):** Enables Guardian Stabilization and (if you have an elevelon model) turns on Elevelon Mixing. **WARNING: Never turn this option off or on during flight if you have an elevelon model! Disabling stabilization will also disable the OSD Pro's elevelon mixing. Use the stabilization gain setting instead to turn off stabilization.**
- **Elevelon Recvr Microsecs(0-3000: 1500):** This sets the number of microseconds for a neutral pulse from your RC controller. Do not change this unless you know the pulse length of your controller. Most people do not need to adjust this, and it is only relevant for elevelon models.

### Configure Guardian Stabilizer Menu

- **Reset Level Pitch and Roll:** Click this option while holding the model level after installing the Guardian, if the AHI line is not centered and level. It will mitigate most mounting errors and provide a level AHI.
- **Stabilizer Pitch Gain (0-10: 5):** This controls the amount of stabilization in the pitch Axis. If your airframe is oscillating along its pitch axis, but never its roll axis, it might help to reduce the Pitch gain.
- **Stabilizer Roll Gain (0-10: 5):** This controls the amount of stabilization in the roll Axis. If your airframe is oscillating along its roll axis, but never its pitch axis, it might help to reduce the Pitch gain.
- **Stabilizer Turn Gain (0-10: 5):** This parameter helps counteract trim issues that can cause gradual turns. The higher the value, the more compensation is added to normal flight.
- **Stabilizer Overall Gain (0-10: 8):** This scales the Pitch and Roll gains. When set to 10, both Pitch and Roll will be very strongly stabilized. When set to 0, all stabilization will be disabled. This setting is overridden by the Aux2 input on the OSD Pro if the Menu Navigation Switch is configured for "3-Position" or "None" and a valid servo signal is present on Aux2. If RTH is engaged, the menu's "Stabilization Overall Gain" will be used. See **Stabilization "Master Gain" Knob** section for more information.
- **Max Stab Pitch/Roll Normal (0-90: 50):** The maximum attitudes (pitch angle or bank angle) that the airplane is allowed to maneuver to in Stabilization mode.
- **Max Stab Pitch/Roll RTH (0-90: 35):** The maximum attitudes (pitch angle or bank angle) that the airplane is allowed to maneuver to when RTH is Engaged.
- **Stabilizer Responsiveness (1-10: 7):** How responsive you want your Fly-By-Wire to be with your model. A high value will provide very fast response, but could accidentally stall your model on takeoff if the control stick is moved too quickly. A low value will yield a very smooth flight.
- **Enable Heading Hold (Off):** Locks onto your current heading and trims your plane to fly that heading until the stick is moved from its centered position. See: "Heading Hold" section for more information.
- **Center Stick Stabilize Only (Off):** This advanced feature enables Guardian Stabilization only when the control stick is centered within the area defined by "Center Stick Tolerance" Box, making it easier to do aerobatic maneuvers. Note that stabilization will be reduced or disabled when your sticks are not centered! See: "Center-Stick Only Stabilization"
- **Center Stick Box Size (1-10: 10):** Defines the size of the Center Stick Box used by "Center-Stick Only Stabilization." Small values make the area smaller and vice versa. See: "Center-Stick Only Stabilization".

### Select Items to be Displayed Menu

- **Servo Deflections Display:** This turns on Servo Deflections display, which is useful for initial Guardian testing.
- **Artificial Horizon Line:** This turns on the AHI display, which is also useful for Guardian setup and testing, as well as in normal flight.

## Trimming

It is recommended that the airplane be trimmed for level flight before running the Servo Analysis Wizard. This ensures that the Guardian Stabilizer will also be trimmed.

If a change in trim occurs once Stabilization has been configured, make sure to execute the "Reset Cruise Sticks/Throtl" to inform the OSD Pro of your change. Remember that your throttle needs to be in the "Cruise" position (the throttle position for level flight) if the RTH feature is enabled.

## Guardian Configuration

Before you can fly for the first time, it is necessary to tell the Guardian a few things about your model. These settings are made by using the OSD Pro menus described above and in the OSD Pro manual.

1. Run the "Servo Analysis Wizard." This step lets the Guardian know how much deflection it can use when stabilizing your airframe. Additionally, this teaches the Guardian the directions of your servos. When the wizard asks you to "Mov Rudder/Aileron Stick Left," deflect the aileron stick to the maximum extent that you would normally use in flight (normally the maximum throw). Do the same with the elevator stick when requested to "Mov Elev Tx Stick Back(Climb)." **Note: When not using the Guardian, the amount of stick deflection during the wizard was used to set the maximum elevator and aileron/rudder stick throws for RTH, so it was recommended to use only limited stick movement. However with the Guardian, the RTH limits are set as angles, as described below, and giving the full stick movement during the wizard is necessary.**
2. For initial testing, we recommend turning on "Servo Deflection Display". When stabilization is turned on, the present Overall Gain setting is displayed as "G" in the servo deflections, and the present pitch and roll of the model are also displayed. Further, we recommend enabling the Artificial Horizon display, which will help to validate that the Guardian is mounted and operating correctly.
3. Choose your airframe type with the "Choose Model Wing Type" menu, by selecting either "Fixed Wing" or "Elevon" type. **Both of these modes require that your radio supplies un-mixed Aileron/Elevator signals. Mixed inputs to the OSD Pro are not supported with Guardian Stabilization enabled.** Note that the OSD Pro supports mixed elevon inputs when stabilization is disabled, but requires unmixed inputs when stabilization is enabled, to provide the best stabilization experience.
4. Enable Stabilization by setting the "Stabilization / Elevn Mixing On?" option in the Radio Control Configuration menu to "Yes." **Warning: Never turn this option off or on during flight if you have an elevon model, since the mixing mode will change! Adjust the stabilization gain setting instead.**
5. Orient your model as it would be during level flight. If the Artificial Horizon display is not centered on the Home indicator or is tilted, click "Reset Level Pitch and Roll." This step will minimize the impact of minor mounting errors with your Guardian.
6. Configure the "Max Stab Pitch/Roll Normal" angles that the Guardian will allow your airplane to attain. For greater maneuverability, set these values to 60 degrees. For easier flight behavior, set them to around 40 degrees.  
**Warning: Setting these values to greater than 60 degrees will result in very responsive flight characteristics from your airframe. Going into maneuvers that your airframe is incapable of sustaining will still result in a stall! Additionally, setting the max angles too low may make your airplane difficult to turn.**
7. If you will be using the Return to Home feature, adjust the "RTH" Max Pitch/Roll angles. These parameters act the same as the Stabilizer Pitch/Roll angles, but are only used for limiting the Return to Home. It is recommended that these values be around 25 degrees, depending on user preference, airframe stability and wind conditions. Once again, a larger angle will result in a more maneuverable RTH, with a smaller turn radius, but may result in oscillations depending on the RTH gain values.  
**Warning: A smaller number will result in smooth RTH behavior, but may prevent your airframe from returning in strong wind conditions or if the Guardian is mounted improperly. It is recommended that you test RTH after changing these values.**
8. It is recommended that your Safety Mode RTH gain settings be returned to their default values, if you have changed them in the past, when flying without the Guardian. When the Guardian is properly configured, most airframes should work with the default RTH gains. But, it may still be necessary to adjust the RTH settings with the Guardian installed.  
**Note: If the RTH does not turn quickly enough, increase the Max Stab Roll RTH value before changing the RTH gains. Do not exceed 35 degrees. For further information, see the RTH tuning section of the OSD Pro manual.**
9. The Stabilizer Pitch/Roll and Overall gains can be left at their default values of 5, 5 and 8 respectively. You will tune these gains later.

### The Servo Analysis Wizard

Since Guardian Stabilization interprets your stick movements as attitude commands, rather than direct servo deflections, it is important to tell it how far your RC controller's sticks move in the pitch/roll axes. At the same time, the Guardian needs to know the trim conditions of your plane as well as the centered position of your control stick. The Servo Analysis Wizard figures this out for you.

When requested to "Set Sticks for level flight," your control stick should be centered (trimmed for level flight with stabilization turned down) and your throttle set to its cruise level. The throttle levels are not required by the Guardian, but are required for Return to Home. Please see the OSD Pro manual for information regarding the throttle / failsafe features of the Servo Analysis Wizard.

When asked to "Mov Rudder/Aileron Stick Left," make sure to assert maximum servo deflection on your RC controller. The same should be done when asked to "Mov Elev TX STICK Back(Climb)." These steps tell your OSD Pro the maximum servo deflections allowed for Guardian Stabilization. If the stick is not moved to its maximum extents for this portion of the wizard, your control commands may be clipped, impacting performance and stability.

**Note: After trimming your airplane on your RC controller, it is important to run the "Reset Cruise Sticks/Throtl" to inform the controller of the change, otherwise Heading Hold and Center-Stick Only Stabilization may be impacted.**

### Reset Cruise Sticks/Throtl

It can be cumbersome to re-run the Servo Analysis Wizard after changing the trim of your airplane. To let your OSD Pro know about changes to trim, it is important to run the "Reset Cruise Sticks/Throtl" command. This resets the center position of your control stick, as well as the cruise position of your throttle, ensuring that features like Heading Hold and Center-Stick Only Stabilization work optimally.

## Tuning Stabilization Gains

### Stabilization Theory

The Guardian Stabilizer works by measuring the orientation of your model airplane relative to the local direction of Earth's gravity. In addition, it interprets pitch and roll attitudes commanded by the servo signals received through your elevator and aileron channels respectively.



With this information, it is able to determine the error between your requested flight attitude and your plane's true orientation. By passing this error through a controller algorithm and sending it as servo commands through the elevator and aileron outputs on your OSD Pro, the Guardian can turn your squirrely model into a smooth-flying wing on rails!

Each axis has a separate control gain used to tune stabilization. These are defined by, "Stabilizer Pitch Gain," "Stabilizer Roll Gain" and "Stabilizer Overall Gain." The Pitch and Roll Gains are scaled by the Overall Gain to provide individual axis tuning while also providing On-The-Fly adjustment depending on conditions and aerodynamic performance.

In general, the higher the gain, the more responsive and resilient your plane will be to disturbances such as wind, balance and trim issues. Tuning the gains too high however may result in oscillations of the airframe. It is recommended that the Overall Gain be used to tune the airframe until it oscillates and then be turned back down. This will ensure the best performance while preventing future oscillations.

**Warning: Be careful while tuning the Overall Gain. Some airframes may oscillate very strongly at higher gains, possibly damaging themselves. To avoid strong oscillations, increase the Overall Gain slowly and reduce it as soon as oscillations are noticed.**

### Stabilization "Master Gain" Knob

When the OSD Pro's "Choose Menu Receiver Inputs" option is set to "3-Position" or "None", the OSD Pro's Aux2 input channel can be used as a Master Gain for the Guardian. This overrides the OSD Pro's menu option "Overall Gain" during normal flight (the "Overall Gain" setting is always used during RTH). This Master Gain multiplies the individual Pitch/Roll gains and serves as a way to easily tune both. If you have an RC controller that has an analog knob, this can be mapped to the Aux2 channel to allow for On-The-Fly stabilization gain adjustment.

This knob gives the ability to transition smoothly from stabilized flight to un-stabilized flight. Additionally, since your airspeed can change your airframe's flight characteristics, it is often helpful to be able to tune the controller in flight. Increasing its value will turn on stabilization, increasing the Guardian's effect until a full fly-by-wire behavior is achieved.

When this feature is used, the stabilizer's control characteristics given the Master Gain Knob's position are as follows:

- -100%: Stabilization is OFF. You can fly normally.
- -100%~-0% (depending on Pitch/Roll gain values): Medium Stabilization. Your airframe is now inherently stable, but depending on your knob selection, may still be rolled by strong stick movements or turbulence.

**Note:** The Max Stabilized Pitch/Roll angles specified in the "Configure Guardian Stabilizer" menu may be exceeded in this range, depending on the Master Gain knob's position, flight behavior of your model and type of maneuvers attempted. **Do not attempt maneuvers that may destabilize your airplane!**

- ~0% - 100%: Maximum stabilization. The model should not roll over when in this range. Servo commands will become exaggerated, leading to tighter control of the plane and true Fly-By-Wire functionality.

Note that if you don't have a knob available on your radio, you can also use the OSD Pro Aux 2 input as an "on/off" switch for stabilization. This is done by connecting a spare switch channel to the Aux 2 input. Then, depending on the endpoints programmed for your switch in your radio, you can set the on and off stabilization levels and turn off stabilization by flipping the switch.

If no connection is made to the OSD Pro Aux 2 input, or you have not configured your OSD switch mode to "3-position" or "None", the on-screen "Stabilizer Overall Gain" setting is always used. The "Stabilizer Overall Gain" setting is also always used during RTH.

**Note:** The described behavior assumes a standard deflection controller, with -100% being defined by a 1.1ms servo pulse and +100% being a 1.9ms pulse.

**Warning:** When RTH is enabled, the Guardian Overall Gain from the "Configure Guardian Stabilizer" menu is used, even if the Aux2 input is configured to provide the Overall Gain as well. If the menu value of Overall Gain in this menu is set too low, then RTH will not be stabilized.

### Displaying the Master Gain Onscreen

To display the Master Gain on all OSD Pages, turn on the "Servo Deflections Display" as described above. With the Servo Deflection Display line enabled, the "G" parameter on that line represents the Overall Gain.

### Oscillations

Sometimes, when the controller gains are set very high or if there is a very high airspeed, oscillations can develop. If oscillations occur, turn down the Master Gain knob, change the inputs of your master gain switch, or adjust the Overall Gain menu setting if a knob or switch is not being used, until the oscillations stop.

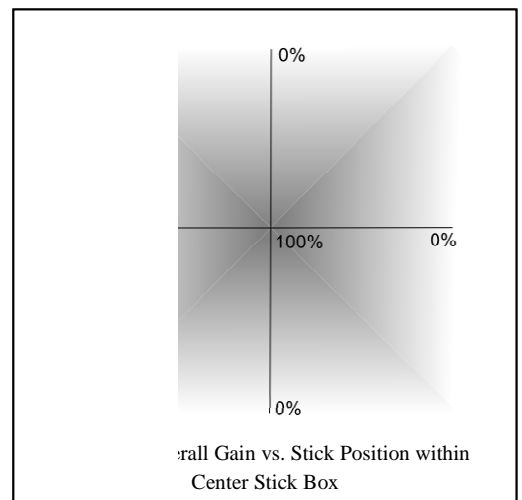
## Additional Features

### Center-Stick Only Stabilization

Some people looking for more aerobatic capabilities from their airplane prefer to disable the stabilizer until they need it for crash avoidance or just to take a short break without needing to land. Center-Stick Only Stabilization allows for a quick return to stabilized flight at any time by letting go of the control stick. It can be enabled in the "Configure Guardian Stabilizer" menu.

After turning on this feature, choose the size of the Center-Stick Box by adjusting the "Center Stick Box Size". Choosing "1" will size the box to 10% of your control stick movement range (best for 3D flight) while "10" will cover the entire physical range of your control stick (smoother turns). Moving the control stick away from its centered position will decrease your Overall Gain until it reaches zero at the edge of the Center Stick Box as illustrated in Figure 1.

Moving the control stick outside the Center-Stick Box will accomplish the same as turning your Overall



Gain Knob to 0%. Bringing the stick back to center will bring the Overall Gain back to its original value, set either by the "Stabilizer Overall Gain" parameter or by the Aux2 servo input as explained in the section "Stabilization "Master Gain" Knob."

**IMPORTANT: Remember that when Center Stick stabilization is enabled, moving or trimming your control stick will reduce stabilization, including during takeoff or hand launch! After trimming, remember to execute the "Reset Cruise Sticks / Throtl" command to ensure full stabilization.**

### Heading Hold (Recommended)

When the Guardian tries to stabilize an un-trimmed or off-balance airframe, a gradual turn can occur even if level flight is commanded. To counteract this, it is recommended to enable the Heading Hold feature by ensuring the "Stabilizer Turn Gain" in the "Configure Guardian Stabilizer" menu is set to its default value of "5" and setting "Enable Heading Hold" to "Yes."

Once enabled, Heading Hold keeps track of your current heading and, when your control stick's roll axis becomes centered, will lock onto that heading. As soon as the control stick is moved to command any roll, the current heading will be reset. The pitch axis can still be commanded with Heading Hold enabled to allow for a heading locked climb or descent.

At this time, the Heading Hold's directional lock does not use GPS, so a small amount of drift over time can occur. Additionally for safety reasons, the Heading Hold will not track yaw movements beyond 45 degrees from the locked heading. If the plane yaws further than this, the locked heading will be automatically adjusted so as to be within 45 degrees of the current heading.

While the default of "5" for "Stabilizer Turn Gain" should work for most airframes, a higher number may be needed if the airplane does not track closely enough due to trim issues. If, when flying at normal Pitch, Roll and Overall Gains, your airplane begins to oscillate in the Yaw axis during Heading Hold, reduce the Turn Gain until no oscillation is detected.

After enabling Heading Hold on the ground, it is recommended that as part of your preflight checks you "yaw" the model (rotate it gently so that the wings rotate around like helicopter blades) and observe the Ailerons/Elevons. Confirm that the control surfaces are asserting a turn that would bring the airplane back to its original heading. When the airplane is pointing directly at its currently locked heading, the control surfaces should be oriented for level flight.

**Note: Remember to reset your current heading before takeoff to prevent an immediate turn. This is done by pointing your model in the direction of takeoff and moving the Ailerons via the control stick.**

**Note: Heading Hold will not work with the Stabilization Overall Gain set to "0."**

**Note: After trimming your Aileron / Roll axis control stick, remember to run "Reset Cruise Sticks / Throtl" or Heading Hold may become disabled.**

## Preflight

The Guardian requires no "warm up" period before flying. But, before every flight, it is strongly recommended that you perform the following steps:

1. Perform your normal preflight check. Ensure your servos respond correctly to your RC control stick commands.
2. To test that the Guardian is working, hold the airplane level. Now, pitch and roll the model. You should see the control surfaces deflecting appropriately to level the airplane. If the control surfaces do not move at all, ensure that, if connected, the Master Gain knob or Gain switch is set so that stabilization will occur. If the control surfaces are moving the wrong direction, it might be necessary to run the Servo Analysis wizard again, making sure to follow the onscreen instructions closely.

## First Flight

First, orient your model as it should be for level flight and execute the "Reset Level Pitch and Roll" command under the "Configure Guardian Stabilizer" menu. This helps to compensate for any mounting errors with your Guardian.

On your first flight, it is recommended that you do an unpowered hand-toss of your airplane, if yours is capable of hand-toss launches. If the airplane safely glides to the ground with little or no input, then your configuration is correct. If the model oscillates during the hand-toss, your Overall Gain is probably too high.

Once it has been shown that the Guardian has been correctly configured, perform one last preflight check and take off with power.

**Warning: If taking off with stabilization turned up to a high gain, be very careful to avoid overcompensation. If the Max Pitch/Roll angles for Stabilization are set to a large number, say 60 degrees, there could be a very strong response that might stall your airplane. Move your sticks smoothly at first to get the feel for your more responsive airframe.**

Once in the air and only when safe, release the control stick so it sits at zero input. Your airplane should fly straight and level. If not, your Guardian may be mounted improperly. You can land and run the "Reset Level Flight" command again. This may be done in the air also, if your piloting skills permit it.

## Troubleshooting

Below is a list of problems that may be encountered, and steps to remedy them. A good first step to troubleshooting is to download our latest software and update the eLogger, OSD Pro and Guardian's Firmware via "Hardware, Firmware Control." Doing this will ensure that you have the latest software and firmware available. Just click "Help, Download latest updates..." in the software to check to see if a newer version of our software is available.

If your particular issue is not addressed by this update, or by the suggestions below, please visit our web support page for additional support options, at <http://www.eagletreesystems.com/Support/support.html>.

**Issue:** The message "Stabilizer Disabled in Menu!!" appears on the video screen during OSD Pro startup.

**Solution:** This message indicates that you need to turn on stabilization (if desired) by setting the "Stabilization / Elevn Mixing On?" setting to YES, as described in the Guardian Configuration section above.

**Issue:** The message "Ailron or Elevatr undetected!" appears on the video screen during OSD Pro startup.

**Solution:** This message indicates that receiver pulses from either Ailron or Elevator channel were not detected. Both of these channels are required for stabilization to work correctly. Make sure that the channels are connected correctly from your receiver. Also, note that some receivers will not send any pulses until the radio transmitter is turned on.

**Issue:** The message "Run RC Wiz for Stabilize!" appears on the video screen during OSD Pro startup.

**Solution:** This message indicates that you need to run the Servo Analysis Wizard under the Configure Radio Control menu.

**Issue:** The message "Stabilizer using Menu Gain!" appears on the video screen during OSD Pro startup.

**Solution:** This message indicates that the Aux2 input for Overall Gain adjustment is not being used. The menu overall gain setting will be used instead.

**Issue:** The message "Stabilizer Disconnected!" appears on the video screen during OSD Pro startup.

**Solution:** This message indicates that the Guardian Expander is not detected by the system. Check your connections on the Guardian expander, and read further troubleshooting steps below.

**Issue:** Guardian does not seem to be working correctly and the Guardian's status LED IS flashing rapidly.

**Solution:** Make sure you have updated the OSD Pro, eLogger and Guardian firmware as described above.

**Issue:** Guardian does not seem to be working correctly and the Guardian's status LED IS NOT flashing rapidly.

**Solution:** Make sure that the Guardian is connected correctly to the LCD/TX line on your eLogger and that your eLogger is powered. If the Guardian's status LED is blinking three times in a repeating fashion, then your unit has likely been damaged.

**Issue:** Servo deflections are limited, compared to when Stabilization is disabled.

**Solution:** Run the Servo Analysis Wizard under the Configure Radio Control menu. Make sure to assert full stick deflections when asked to turn and climb during the wizard. See the section on the Servo Analysis Wizard under Guardian Stabilization.

**Issue:** Elevator or ailerons are unresponsive to radio commands

**Solution:** Make sure you have run the Servo Analysis Wizard after installing Guardian.

**Issue:** Servos are responding to control inputs, but not to movements of the airplane.

**Solution:** Ensure that the Guardian's 4 pin cable is connected as described in the Connection and Mounting section above. Check that "Stabilization/Elevn Mixing On?" is set to "YES" under the Configure Radio Control menu. Confirm that the Pitch, Roll and Overall Gains are set to their default values. If you are using the Aux2 input for Overall Gain adjustment, ensure that your knob, switch, or slider is positioned to enable stabilization.

**Issue:** Heading Hold does not appear to be working or is drifting more than a few degrees per second.

**Solution:** If you have trimmed your airplane recently, you may need to execute the "Reset Cruise Sticks/Throtl" command under "Configure Radio Control." The Heading Hold feature only engages when your Roll command is very close to level flight. (Less than ~15% stick deflection)

## ***Limited Warranty***

Eagle Tree Systems, LLC, warrants the Guardian to be free from defects in materials and workmanship for a period of one (1) year from the date of original purchase. This warranty is nontransferable. If your unit requires warranty service during this period, we will replace or repair it at our option. Shipping cost to us is your responsibility. To obtain warranty service, email [support@eagletreesystems.com](mailto:support@eagletreesystems.com) for further instructions.

This limited warranty does not cover:

- The Software. See the Software license agreement for more information on Software restrictions.
- Problems that result from:
  - External causes such as accident, abuse, misuse, or problems with electrical power
  - Servicing not authorized by us
  - Usage that is not in accordance with product instructions
  - Failure to follow the product instructions

THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS, AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM STATE TO STATE (OR JURISDICTION TO JURISDICTION). OUR RESPONSIBILITY FOR MALFUNCTIONS AND DEFECTS IN HARDWARE IS LIMITED TO REPAIR AND REPLACEMENT AS SET FORTH IN THIS WARRANTY STATEMENT. ALL EXPRESS AND IMPLIED WARRANTIES FOR THE PRODUCT, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES AND CONDITIONS OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED IN TIME TO THE TERM OF THE LIMITED WARRANTY PERIOD AS DESCRIBED ABOVE. NO WARRANTIES, WHETHER EXPRESS OR IMPLIED, WILL APPLY AFTER THE LIMITED WARRANTY PERIOD HAS EXPIRED. SOME STATES DO NOT ALLOW LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THIS LIMITATION MAY NOT APPLY TO YOU.

WE DO NOT ACCEPT LIABILITY BEYOND THE REMEDIES PROVIDED FOR IN THIS LIMITED WARRANTY OR FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, ANY LIABILITY FOR THIRD-PARTY CLAIMS AGAINST YOU FOR DAMAGES, FOR PRODUCTS NOT BEING AVAILABLE FOR USE, OR FOR LOST DATA OR LOST SOFTWARE. OUR LIABILITY WILL BE NO MORE THAN THE AMOUNT YOU PAID FOR THE PRODUCT THAT IS THE SUBJECT OF A CLAIM. THIS IS THE MAXIMUM AMOUNT FOR WHICH WE ARE RESPONSIBLE. SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU.



UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

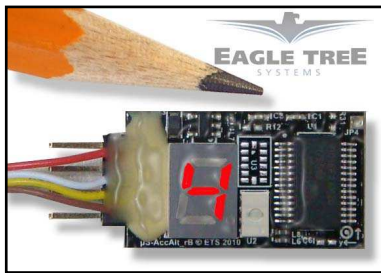
*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

---

**Máster en Ingeniería Industrial**

**Documento**

**ANEXO IX: MANUAL INSTRUCCIONES ALTIMETER MICROSENSOR**



## Instruction Manual for the Altimeter MicroSensor V4 Document Version 1.2

Thank you for your purchase! This instruction manual will guide you through the installation and operation of your Altimeter MicroSensor V4 (the Altimeter). **Please read the entire manual carefully before proceeding.** If, after you read the manual, you have further questions or problems, see the Support page on <http://www.eagletreesystems.com> for additional information, or email us at [support@eagletreesystems.com](mailto:support@eagletreesystems.com). Please visit our support web page for the full color, electronic version of this manual which may be updated if changes were made after printing, or if you want to view the manual on your computer.

### Intended Uses

The Altimeter is intended exclusively for recreational use in radio control models and model rockets. Other uses are not supported. Further, using the Altimeter in situations where its use or failure could result in loss of life, bodily injury or property damage is expressly prohibited.

### What the Altimeter Does

The Altimeter is a precision instrument that uses barometric pressure to measure altitude, just as full sized planes do. Advanced temperature compensation and factory calibration ensure the best possible accuracy, with no user calibration required. When used standalone, the Altimeter displays your maximum altitude on the built-in 7 segment LED display. The maximum altitude continues to be updated and displayed on the LED, until you turn power off and on the MicroSensor. When power is turned off and on, the maximum altitude from the last flight is displayed, and now the MicroSensor is ready to record your next maximum altitude (even if it is lower than the previous maximum altitude).

Additionally, the Altimeter can be connected to your eLogger V4 or V3 to provide altitude data for your entire flight, and can even be used in your own microcontroller/firmware project. When connected to the eLogger, altitude data can be downloaded, displayed and graphed using the eLogger's Windows software. **NOTE: the Altimeter V4 is NOT compatible with our prior eLogger V1 or V2 products.**

Want to use the Altimeter for your own hardware/firmware project? Read on in the manual for more information!

**Range check:** It is extremely unlikely that the installation of the Altimeter will affect your model's radio range or control. But, as always after making an electronics change to your model, it is very important that you range and function test your model once the Altimeter is installed to ensure that there is no impact on your system.

### Packing List

Your package should include the following: The Altimeter, the Standalone Power Supply Cable, and a printed version of this manual.

### Installing the Altimeter in your Model

The Altimeter can be mounted anywhere in the model. If you wish to read the LED without removing the wing or panel, a small rectangular hole can be cut at the mounting location of the circuit board, and the board mounted with tape or Velcro, so that the LED is visible at all times. **NOTE:** If your airframe uses a "ram-air" intake into its cockpit area for cooling or other purposes, the pressure increase can cause a slightly lower reported altitude. To mitigate this, do not expose the Altimeter V4 to direct airflow or place it in a "ram-air" chamber. Also, to reduce additional error, do not mount the Altimeter V4 anywhere it will be exposed to rapid temperature changes during the flight. Further, if the sensor will be installed in a location where the sensing element (the small rectangular metal or plastic part next to the LED) will be exposed to direct sunlight, please cover that area of the sensor with black tape – direct sunlight can sometimes cause the readings to vary.

## Using the Altimeter in Standalone Mode

### Powering the Altimeter in Standalone Mode

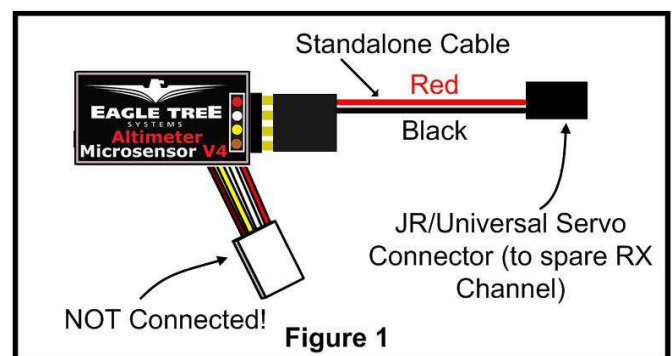


Figure 1

For standalone mode (not connected to an eLogger), connect the Standalone Cable to the gold pins of the Altimeter, as shown in Figure 1.

The polarity of the connection is as follows:

RED = Positive (Vdd)

BLACK = Negative (ground)

Note that the RED wire of the standalone cable corresponds with the red dot on the label.

The JR/Universal servo end of the Standalone Cable connects to a spare Receiver channel or small battery. Note that the voltage must be between about 4V and 16V. **Do not exceed 16V!**

**IMPORTANT: Ensure that you connect the Standalone Cable with the correct polarity, or the Altimeter could be damaged, voiding the warranty! Note that in Standalone Mode, the 4 wire cable built into the Altimeter must NOT be connected to anything!**

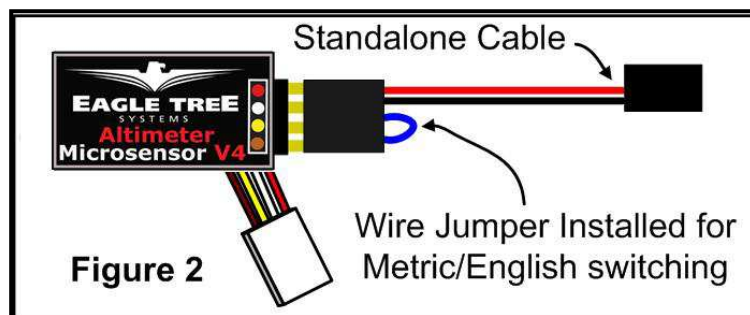
### Configuring Metric vs English Units in Standalone Mode

When powered on, the LED readout will briefly display “0” if the unit is configured for English (Feet), or will display a “1” if configured for metric (Meters). The factory default configuration is Feet.

To switch between English and metric units, connect the Standalone Cable as described above, but don't power the Altimeter. Then, connect a small piece of wire (like a paperclip) between the unconnected pins of the Standalone Cable's 4 pin connector, corresponding to the brown and yellow dots on the label, as shown in Figure 3. This creates a jumper between the brown and yellow labeled pins.

Then, power the Altimeter via the Standalone Cable. A “0” will then flash a few times to indicate that English units are configured, or a “1” will flash if Metric units are configured. If the wrong units are configured, just disconnect the power from the Altimeter and reconnect it.

Once the desired units are configured, disconnect from power and remove the wire jumper. Then, when reconnected to power, the desired units should be displayed briefly (“0” or “1”).



### Reading Altitude in Standalone Mode

When the Altimeter is powered on, after the Units setting is displayed (“0” or “1”), the highest altitude attained during your last flight is repeatedly displayed, one digit at a time, on the LED. For example, if the highest altitude attained was 320 Feet, “3 – 2 – 0” will flash repeatedly on the LED, with a pause (blank LED) after the altitude is completely displayed, but before it is displayed again.

**To reset the max trigger, just turn power off and on to the MicroSensor.** Then, after landing from your next flight, the highest altitude attained during that flight is displayed automatically, overwriting the highest altitude from your prior flight (even if the new highest altitude is less than the prior flight's highest altitude). **Note that an altitude of at least 60 Feet (18 Meters) must be attained, before a new highest altitude is displayed.**

The highest altitude of your last flight is automatically saved, so that when you turn the unit back on, that altitude is displayed.

For example, if you fly and reach a max altitude of 100 Feet, when you land “1 – 0 – 0” will be displayed. Then, when you turn power off and on, “1 – 0 – 0” will again be displayed. If you then fly again and reach an altitude of 500 Feet, “5 – 0 – 0” will be displayed when you land.

## Using the Altimeter with your eLogger V4 or V3

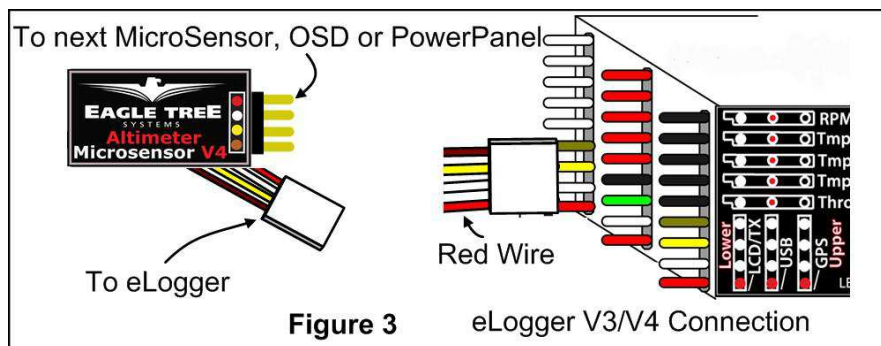
### Windows Software and Firmware Update

To use the Altimeter, you must update your software to **Eagle Tree Windows Software version 9.45** or later.

To update, download the latest software from the support page of our website, located at

<http://eagletreesystems.com/Support/apps.htm>. After downloading and installing the software, the firmware of your eLogger will need to be updated. To upgrade your firmware, just click “Hardware, Firmware Control” and click the Update button for the eLogger.

Copyright © 2011 Eagle Tree Systems, LLC  
<http://www.eagletreesystems.com>



### Connecting the Altimeter to the eLogger V4/V3

The Altimeter plugs into the “LCD/OSD” or “LCD/TX” port of your eLogger V4 or V3, as shown in Figure 3. If you have a PowerPanel or other MicroSensors, those can “daisy chain” connect to the pins on the side of your Altimeter, with the polarity as indicated on the Altimeter label. **NOTE: The Standalone Cable must not be used when connected to the eLogger!**

### Configuring the Altimeter with the Windows Software

If you have not already done so, set up the Recorder software as described in your instruction manual. Then, choose one or more of the Altimeter options below:

#### *Logging Altitude*

To log altitude, just click “Hardware -> Choose Parameters to Log in the Recorder” and check the “Altitude” box.

#### *Displaying Altitude in the Windows Software*

To display the Altimeter Gauge and/or Numeric Altitude Display, click “Software, Choose Instruments to Display on the PC Screen” and check the “Altimeter Gauge,” and/or “Numeric Altitude” boxes.

#### *Graphing Altitude*

To graph altitude, click “Data/2D Chart” and select “Altitude” for graphing.

#### *Displaying Altitude on the PowerPanel LCD*

Select “Hardware, Configure PowerPanel Display”, and choose “Altitude” for PowerPanel display.

## *Using the MicroSensor with your own Firmware/Microcontroller*

Please see this document for information on using the MicroSensor with your own firmware:

<http://www.eagletreesystems.com/support/manuals/microsensor-i2c.pdf>

### Troubleshooting

Below is a list of problems that may be encountered, and steps to remedy them. If your particular issue is not addressed by the below, see the Support page on <http://eagletreesystems.com> or email [support@eagletreesystems.com](mailto:support@eagletreesystems.com). Include a full description of your problem, your PC configuration, your Eagle Tree hardware and software versions, and any other relevant information.

**Issue:** Altitude doesn't vary in my recordings

**Solutions:**

- Ensure that the Altimeter is connected correctly to the MicroPower
- Ensure that you are logging altitude, under “Hardware, Choose Parameters to Log in the Recorder”

**Issue:** Altitude is varying rapidly (more than +/- 1') when I'm not changing altitude.

**Solution:**

- Ensure that the Altimeter's sensing element (the small rectangle next to the LED) is not exposed to direct sunlight. This can be done by covering the sensor with black electrical tape (except for the LED portion) or mounting the sensor inside the fuselage.

### Specifications (Approx)

- Measures altitude up to approx 20,000 Feet (6096 Meters) Above Sea Level
- Resolution: approx. +/- 1 foot (0.3 Meters)
- Power input (Standalone Mode) : 4V to 16V
- Weight: approximately 4 grams (0.15 oz)
- Dimensions: approximately 28 mm x 16 mm x 10 mm (1.1" x 0.62" x 0.4")
- Precalibrated – no user calibration required
- Advanced Temperature Compensation
- Metric or English units

### Limited Warranty

Eagle Tree Systems, LLC, warrants the Altimeter to be free from defects in materials and workmanship for a period of one (1) year from the date of original purchase. This warranty is nontransferable. If your unit requires warranty service during this

Copyright © 2011 Eagle Tree Systems, LLC

<http://www.eagletreesystems.com>

period, we will replace or repair it at our option. Shipping cost to us is your responsibility. To obtain warranty service, email [support@eagletreesystems.com](mailto:support@eagletreesystems.com) for further instructions.

This limited warranty does not cover:

- The Software. See the Software license agreement for more information on Software restrictions.
- Problems that result from:
  - External causes such as accident, abuse, misuse, or problems with electrical power
  - Servicing not authorized by us
  - Usage that is not in accordance with product instructions
  - Failure to follow the product instructions

THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS, AND YOU MAY ALSO HAVE OTHER RIGHTS WHICH VARY FROM STATE TO STATE (OR JURISDICTION TO JURISDICTION). OUR RESPONSIBILITY FOR MALFUNCTIONS AND DEFECTS IN HARDWARE IS LIMITED TO REPAIR AND REPLACEMENT AS SET FORTH IN THIS WARRANTY STATEMENT. ALL EXPRESS AND IMPLIED WARRANTIES FOR THE PRODUCT, INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES AND CONDITIONS OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, ARE LIMITED IN TIME TO THE TERM OF THE LIMITED WARRANTY PERIOD AS DESCRIBED ABOVE. NO WARRANTIES, WHETHER EXPRESS OR IMPLIED, WILL APPLY AFTER THE LIMITED WARRANTY PERIOD HAS EXPIRED. SOME STATES DO NOT ALLOW LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THIS LIMITATION MAY NOT APPLY TO YOU.

WE DO NOT ACCEPT LIABILITY BEYOND THE REMEDIES PROVIDED FOR IN THIS LIMITED WARRANTY OR FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, ANY LIABILITY FOR THIRD-PARTY CLAIMS AGAINST YOU FOR DAMAGES, FOR PRODUCTS NOT BEING AVAILABLE FOR USE, OR FOR LOST DATA OR LOST SOFTWARE. OUR LIABILITY WILL BE NO MORE THAN THE AMOUNT YOU PAID FOR THE PRODUCT THAT IS THE SUBJECT OF A CLAIM. THIS IS THE MAXIMUM AMOUNT FOR WHICH WE ARE RESPONSIBLE. SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE LIMITATION OR EXCLUSION MAY NOT APPLY TO YOU.





UNIVERSIDADE DA CORUÑA



Escola Politécnica Superior

**TRABAJO FIN DE MÁSTER  
CURSO 2016/17**

---

*MEJORA DE UN MODELO DE UN VEHÍCULO AÉREO  
NO TRIPULADO SOBRE SIMULADOR DE CÓDIGO  
ABIERTO*

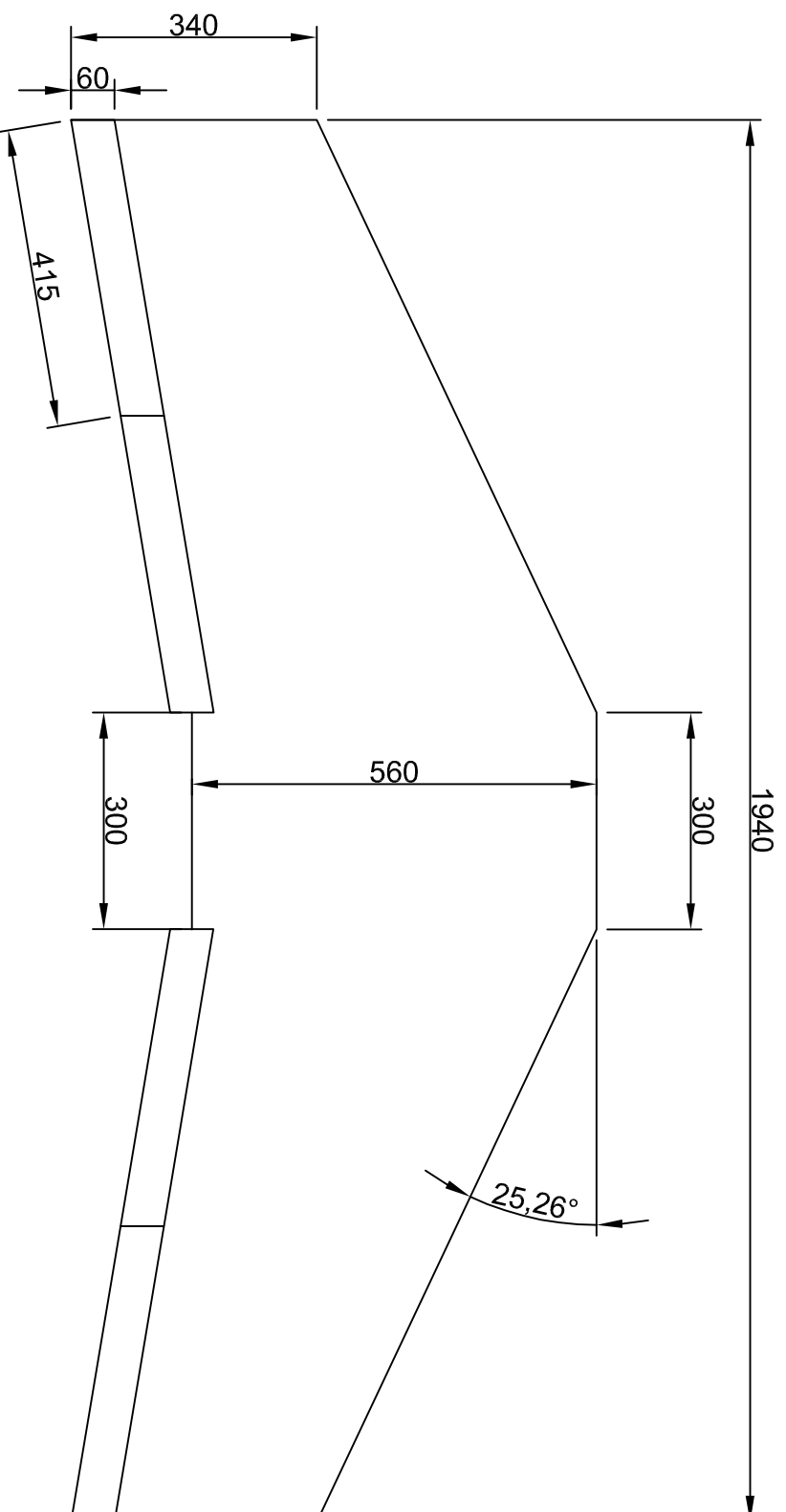
---

**Máster en Ingeniería Industrial**

**Documento**

**ANEXO X: PLANO DEL UAV**

Revisiones			
Rev	Descripción	Fecha	Aprobado



Número del elemento	Nombre archivo (sin extensión)	Autor	Cantidad	Aprobado 1		Aprobado 2		Título	Plano	Rev
				Nombre	Fecha	Nombre	Fecha			
1	Pieza1	Román Sobrino	1	Dibujado	Román Sobrino	17/8/2017		AutocAD 2013	Plano UAV	
				Comprobado						
				Aprobado 2						
2	elevones	Román Sobrino	4	Salvo indicación contraria cotas en milímetros ángulos en grados tolerancias ±0.5 y ±1°				Archivo: Plano UAV.dft	Plano	
				Escala	Peso	Plano 1 de 1				