



Universidad de Jaén

Escuela Politécnica Superior de Jaén

INTEGRACIÓN DE UN SISTEMA DE POSICIONAMIENTO EN UN DRON ROBOMASTER TELLO TALENT

Autor: Alfonso Martínez Sánchez

Grado: Ingeniería Electrónica

Director: Alejandro Sánchez García

Departamento del director: Ingeniería de Sistemas y Automática

Fecha: 10/09/2024

Licencia CC



CREA



Universidad de Jaén
Escuela Politécnica Superior de Jaén
Departamento de Ingeniería Electrónica

Don Sergio Illana Rico y Don Alejandro Sánchez García , tutores del Proyecto Fin de Carrera titulado: INTEGRACIÓN DE UN SISTEMA DE POSICIONAMIENTO EN UN DRON ROBOMASTER TELLO TALENT, que presenta Alfonso Martínez Sánchez, autoriza su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, 10 de septiembre de 2024

El alumno:

Alfonso Martínez Sánchez

Los tutores:

Alejandro
Sánchez García

Sergio Illana Rico

Índice

1. Resumen.....	5
2. Introducción.....	7
2.1. Objetivo.....	7
2.2. Justificación y relevancia del trabajo	7
3. Fundamentos Teóricos.....	10
3.1. Descripción del dron Robomaster TT	10
3.1.1. Características técnicas	11
3.1.2. Capacidades de vuelo y control	12
3.1.3. Control remoto del dron Robomaster TT	13
3.1.4. Precio de Mercado de Robomaster TT.....	14
3.2. Principios del Sistema de Posicionamiento Global (GPS)	16
3.2.1. Funcionamiento básico del GPS	16
3.2.2. Comunicación y protocolo NMEA.....	17
3.3. Microcontroladores y la tarjeta ESP32	19
3.3.1. Características y funciones de la ESP32-D2WD	19
3.3.2. Conexiones y comunicación con periféricos.....	21
4. Selección y Adquisición del Módulo GPS	22
4.1. Identificación de Requisitos del Módulo GPS	22
4.2. Comparativa de Módulos GPS Disponibles en el Mercado.....	24
4.2.1. Análisis de Características Técnicas	24
4.2.2. Relación Calidad-Precio.....	26
4.3. Justificación de la Elección del Módulo GPS.....	26
5. Integración del Módulo GPS con el Dron.....	27
5.1. Conexión del GPS al Módulo ESP32	27
5.1.1. Protocolo de Comunicación y Asignación de Pines.....	27
5.1.2. Diseño e implementación de la conexión física	29
5.2. Montaje y Fijación Física del GPS en el Dron.....	30
6. Herramientas y entornos de programación utilizados	31
6.1. Arduino IDE.....	32
6.2. PyCharm (Python).....	32
6.3. Processing	32
6.4. Software adicional y servicios utilizados	33
7. Metodología	34
7.1. Arquitectura del sistema.....	35
7.2. Programación de la tarjeta ESP32	35

7.3.	Configuración Python – Dron Robomaster TT	42
7.4.	Procesamiento de datos en Python.....	42
7.5.	Desarrollo de la Interfaz Gráfica en Processing.....	43
7.6.	Automatización de la Ruta de Vuelo del Dron en Python	47
8.	Resultados y Discusión	49
8.1.	Evaluación del rendimiento del sistema integrado.....	49
8.2.	Limitaciones y desafíos encontrados.....	50
9.	Conclusiones.....	51
9.1.	Logros alcanzados en el proyecto	51
9.2.	Contribuciones y aplicaciones futuras	52
	Bibliografía	53
10.	Anexos	55
10.1.	Manual de usuario de la interfaz gráfica	55
10.2.	Robomaster TT Users Manual	55
10.3.	Código fuente	55

Índice de Figuras

FIGURA 2.2-1 DRON REALIZANDO TAREAS DE AGRICULTURA.....	9
FIGURA 3.1-1 DRON ROBOMASTER TT (DJI TIENDA, 2024).....	10
FIGURA 3.1-2 TELLO APP Y TELLO EDU APP	13
FIGURA 3.2-1 REPRESENTACIÓN GRÁFICA DE LA TRILATERACIÓN.....	17
FIGURA 3.2-2 SENTENCIAS DE DATOS NMEA.....	18
FIGURA 4.2-1 MERCADO	24
FIGURA 4.3-1 FOXEER M10Q 250 GPS 5883 COMPASS (IHA-RACE, 2024)	27
FIGURA 5.1-1 ESQUEMA DE CONEXIÓN DEL GPS CON UN EQUIPO EXTERNO. (IHA-RACE, 2024).....	28
FIGURA 5.1-2 EXTENSION BOARD (DJI, 2021)	29
FIGURA 5.1-3 EXTENSION BOARD CON SOLDADURA DE PINES MACHO Y CONECTADO AL EXPANSION KIT	30
FIGURA 5.1-4 CABLEADO ENTRE EL GPS Y EL EXTENSION KIT	30
FIGURA 5.2-1 ACOPLAMIENTO FINAL ENTRE EL GPS, EL EXPANSION KIT Y EL DRON.....	31
FIGURA 5.2-2 LOGOS DE PYCHARM, ARDUINO IDE Y PROCESSING	31
FIGURA 6.4-1 LOGO DE OPENSTREETMAP	33
FIGURA 7.1-1 ESQUEMA DE LA ARQUITECTURA DEL SISTEMA.....	35
FIGURA 7.2-1 INSTALACIÓN DE LAS TARJETAS ESP32 DE ESPRESSIF SYSTEMS EN ARDUINO IDE.....	36
FIGURA 7.2-2 CONFIGURACIÓN DE LA TARJETA ESP32 DEL EXPANSION KIT	37
FIGURA 7.2-3 LIBRERÍAS ESPECÍFICAS INCLUIDAS EN ARDUINO IDE	38
FIGURA 7.2-4 INCLUSIÓN DE LAS BIBLIOTECAS SPIFFS.H Y FS.H EN EL CÓDIGO.....	38
FIGURA 7.2-5 SSID, CONTRASEÑA Y PUERTO DEL PUNTO DE ACCESO CREADO	40
FIGURA 7.2-6 IMAGEN GENERADA POR I.A. DE LA CONEXIÓN WIFI ENTRE DRON Y PC	41
FIGURA 7.2-7 CONEXIÓN WI-FI CON RED LOCAL Y RED PROPIA DE LA TARJETA.....	42
FIGURA 7.4-1 ESQUEMA DEL PROCESO DE RECEPCIÓN DE DATOS EN PYTHON	43
FIGURA 7.4-2 CADENA DE TEXTO ENVIADA A PROCESSING DESDE PYTHON.....	43
FIGURA 7.5-1 INTERFAZ GRÁFICA	45
FIGURA 7.5-2 ESQUEMA DEL PROCESO DEL CÁLCULO DE RUTAS	46
FIGURA 7.5-3 CREACIÓN DEL ARCHIVO COORDENADAS_INICIALES.JSON EN UNA RUTA ESPECÍFICA	47
FIGURA 7.6-1 LECTURA Y ELIMINACIÓN DEL ARCHIVO COORDENADAS_INICIALES.JSON.....	47
FIGURA 7.6-2 ESQUEMA DEL BUCLE DE MOVIMIENTO DEL DRON	48
FIGURA 7.6-3 DECLARACIÓN DE LA BIBLIOTECA ROBOT DE ROBOMASTER	49
FIGURA 7.6-4 COMANDOS UTILIZADOS PARA EL CONTROL DE MOVIMIENTO DEL DRON	49

Índice de Tablas

TABLA 5.1-1 NOMBRE DE LOS PINES RESPECTIVOS DEL EXTENSION BOARD (DJI, 2021)	29
TABLA 5.1-2 CONEXIÓN ENTRE PINES DEL GPS CON LOS PINES DEL EXTENSION BOARD ...	29

1. Resumen

El presente Trabajo de Fin de Grado se centra en la integración de un módulo GPS en un dron de bajo presupuesto, el RoboMaster TT, con el objetivo de mejorar sus capacidades de vuelo autónomo y navegación precisa. El proyecto ha implicado varias etapas clave, desde la selección y adquisición del módulo GPS hasta la programación avanzada para la ejecución de rutas de vuelo en tiempo real.

En primer lugar, se realizó un análisis comparativo de diferentes módulos GPS disponibles en el mercado para seleccionar el más adecuado, basándose en sus características técnicas y relación calidad-precio. Una vez seleccionado el módulo GPS, se estudió su comunicación con una tarjeta ESP32, acoplada al dron, a través de una aplicación desarrollada en Processing, que gestiona la recepción y decodificación de los datos GPS.

El siguiente paso fue la conexión física del módulo GPS a la tarjeta ESP32. A través de Arduino IDE, se programó la ESP32 para establecer una conexión Wi-Fi con un ordenador y decodificar los datos GPS recibidos en formato NMEA (National Marine Electronics Association). Posteriormente, los datos fueron enviados a Python para su procesamiento y reenviados a Processing para su tratamiento.

En Processing se desarrolló una interfaz gráfica que permite mostrar la posición del dron en un mapa, obtenido mediante la API de OpenStreetMap, y trazar rutas hacia un destino seleccionado. El sistema es capaz de calcular la ruta en función de las coordenadas GPS y actualizar la posición del dron en tiempo real, gestionando la ejecución de comandos de vuelo autónomo. Además, se implementó una función de parada "STOP" para detener el vuelo en caso de desviaciones o errores en la trayectoria, garantizando mayor seguridad.

Este proyecto no solo demuestra la capacidad técnica para integrar y programar un módulo GPS en un dron económico, sino que también aborda las limitaciones encontradas, como la influencia de las condiciones meteorológicas en la señal GPS y la estabilidad del dron ante vientos leves. Además, ofrece una solución educativa y experimental en el campo de la robótica, la automatización y el control de UAVs (vehículos aéreos no tripulados).

ABSTRACT

This Final Degree Project focuses on the integration of a GPS module into a low-cost drone, the RoboMaster TT, with the aim of improving its autonomous flight and precise navigation capabilities. The project involved several key stages, from the selection and acquisition of the GPS module to the advanced programming required for real-time flight route execution.

Firstly, a comparative analysis of different GPS modules available on the market was conducted to select the most suitable one, based on its technical features and cost-effectiveness. Once the GPS module was chosen, its communication with an ESP32 board, attached to the drone, was studied through an application developed in Processing, which manages the reception and decoding of GPS data.

The next step was the physical connection of the GPS module to the ESP32. Using Arduino IDE, the ESP32 was programmed to establish a Wi-Fi connection with a computer and to decode the GPS data received in NMEA (National Marine Electronics Association) format. Subsequently, the data was sent to Python for processing and then to Processing for further handling.

A graphical interface was developed in Processing to display the drone's position on a map, obtained through the OpenStreetMap API, and to plot routes to a selected destination. The system is capable of calculating the route based on GPS coordinates and updating the drone's position in real-time, managing the execution of autonomous flight commands. Additionally, a "STOP" function was implemented to halt the flight in the event of deviations or errors in the trajectory, ensuring greater safety.

This project not only demonstrates the technical ability to integrate and programme a GPS module into a low-cost drone but also addresses the limitations encountered, such as the influence of weather conditions on the GPS signal and the drone's stability in light winds. Moreover, it provides an educational and experimental solution in the field of robotics, automation, and UAV (unmanned aerial vehicle) control.

2. Introducción

2.1. Objetivo

El propósito central de este proyecto es integrar un módulo GPS en el dron RoboMaster TT para que pueda ser ubicado con precisión en todo momento. Esta funcionalidad es fundamental para mejorar la capacidad del dron de realizar tareas de navegación y monitorización con alta fiabilidad.

Además, se han establecido otros objetivos secundarios que incluyen:

- El desarrollo de una interfaz que permita la interacción en tiempo real con los datos generados por el GPS conectado al dron, facilitando su monitorización y control.
- La implementación de rutas rápidas dentro de la interfaz que faciliten el desplazamiento eficiente del dron hacia un destino predeterminado.
- La habilitación del vuelo autónomo del dron RoboMaster TT, de manera que el dron pueda navegar automáticamente desde su ubicación actual hasta el destino seleccionado, optimizando su rendimiento en entornos complejos.

2.2. Justificación y relevancia del trabajo

La creciente incorporación de drones en diversas industrias resalta la importancia de comprender y dominar las tecnologías que los impulsan. En este proyecto, se centra la atención en la implementación de un módulo GPS en el dron RoboMaster TT. Aunque el enfoque inicial es educativo, las implicaciones de este trabajo van más allá, contribuyendo a la preparación en un campo de gran relevancia tecnológica y económica.

Los drones, o (UAVs), han revolucionado múltiples sectores debido a su versatilidad y capacidad para ejecutar tareas que anteriormente resultaban costosas o imposibles. Desde la captura de imágenes aéreas hasta la entrega de mercancías, la monitorización ambiental y las aplicaciones en seguridad y defensa, los drones se

sitúan en el centro de una transformación tecnológica. En la figura 2.2-1 se presenta un dron realizando tareas agrícolas. En este contexto, la integración de módulos GPS en drones no solo mejora su funcionalidad básica, sino que también facilita el desarrollo de aplicaciones avanzadas, como el vuelo autónomo y la navegación precisa. La implementación y programación de un módulo GPS en el RoboMaster TT es un paso en la evolución de las capacidades de los UAVs, permitiendo su uso en contextos donde la precisión y la confiabilidad son esenciales.

Desde una perspectiva educativa, este proyecto ofrece una oportunidad práctica para aplicar conocimientos avanzados en programación, electrónica y control de sistemas. La manipulación de datos GPS, la integración de estos datos en un sistema de control, y la implementación de estrategias de vuelo autónomo son competencias fundamentales que preparan a futuros profesionales para enfrentar desafíos en el diseño y desarrollo de tecnologías UAV. Este tipo de proyectos promueve un aprendizaje activo y experimental, donde se puede observar el impacto real de su aplicación en contextos prácticos, para una comprensión profunda de la tecnología involucrada.

La creciente demanda de drones en sectores como la agricultura, la logística, la minería y la gestión de recursos naturales pone de relieve la necesidad de contar con expertos capacitados en el diseño, implementación y optimización de drones. La precisión y fiabilidad que aporta un módulo GPS bien implementado son cruciales en aplicaciones como la agricultura de precisión, donde una monitorización exacta de los cultivos puede optimizar el uso de recursos y aumentar la producción. En la logística, los drones pueden reducir significativamente los tiempos de entrega y los costos operativos.



FIGURA 2.2-1 DRON REALIZANDO TAREAS DE AGRICULTURA

La incorporación de un GPS en un dron de bajo presupuesto es una innovación interesante porque permite que tecnologías avanzadas, antes solo disponibles en drones más caros, estén al alcance de todos. Con esta mejora, drones más accesibles pueden realizar tareas como vuelos autónomos, navegación precisa y recolección de datos geolocalizados, algo que tiene aplicaciones clave en áreas como la agricultura, la logística o la vigilancia ambiental. Esto no solo ayuda a reducir costos, sino que también permite que más personas y empresas puedan aprovechar las ventajas de los drones, utilizando tecnología accesible para mejorar procesos y resultados en distintos tipos de industria.

Este proyecto contribuye al avance en sostenibilidad y responsabilidad social. Los drones equipados con GPS pueden realizar tareas esenciales en la monitorización ambiental, como la vigilancia de bosques para la prevención de incendios, la inspección de infraestructuras para evitar desastres, o la recopilación de datos para estudios científicos sobre el cambio climático. Estas aplicaciones son fundamentales para la sostenibilidad ambiental y demuestran cómo la tecnología puede ser utilizada en beneficio del bienestar común.

La justificación de este proyecto reside en su capacidad para combinar el aprendizaje práctico con la innovación tecnológica, preparando a futuros profesionales para desempeñar un papel clave en un sector que está transformando el mundo. La implementación del módulo GPS en el RoboMaster TT evidencia el potencial de la

tecnología de drones y su capacidad para mejorar tanto los procesos industriales como la calidad de vida en la sociedad.

3. Fundamentos Teóricos

3.1. Descripción del dron Robomaster TT

Esta sección se basa en la información proporcionada en el manual de usuario del RoboMaster TT (véase Anexo II).



FIGURA 3.1-1 DRON ROBOMASTER TELLO TALENT (DJI TIENDA, 2024)

El RoboMaster Tello Talent (TT) mostrado en FIGURA 3.1-1 es un pequeño dron educativo, diseñado para facilitar el aprendizaje y la experimentación en robótica y programación. Como parte de la línea de productos de DJI y Ryze Tech, destaca por su versatilidad y capacidades avanzadas, lo que lo convierte en una opción ideal tanto para principiantes como para usuarios más avanzados.

El RoboMaster TT está equipado con un Sistema de Posicionamiento Visual (VPS), que le permite mantener su estabilidad y posición con gran precisión durante el vuelo, característica útil en interiores y en entornos con poco viento. Este sistema, que se basa en una cámara y un módulo de infrarrojos 3D, será útil durante proyecto para asegurar vuelos estables.

Incluye una cámara de 5 megapíxeles que captura imágenes de alta resolución y permite la transmisión de video en tiempo real en 720p. Esta capacidad será aprovechada en el proyecto para monitorizar visualmente las rutas de vuelo.

La característica más destacada para este proyecto del RoboMaster TT es su tarjeta programable de código abierto, que cuenta con conectividad Wi-Fi de doble banda y Bluetooth. Esta tarjeta es compatible con plataformas de programación como Arduino, Scratch y MicroPython, y proporciona varios puertos de expansión (UART, I2C, GPIO, PWM, SPI). En el proyecto, esto facilitará la integración del módulo GPS y la recolección de sus datos además del envío de órdenes desde un equipo externo.

El kit de expansión que acompaña al dron permite añadir accesorios, en este caso el GPS, y personalizar sus funcionalidades mediante la tarjeta programable.

RoboMaster TT además de ser una plataforma educativa avanzada, es una herramienta equipada con las características técnicas necesarias para integrar un módulo GPS y desarrollar funciones de vuelo autónomo.

3.1.1. Características técnicas

En este apartado, se detalla de manera técnica las principales especificaciones y funcionalidades del RoboMaster TT:

- **Peso del dron:** 87 g (con protectores de hélices incluidos).
- **Peso del Expansion Kit:** 25 g
- **Velocidad máxima:** 17.8 mph (28.8 kph).
- **Tiempo máximo de vuelo:** 13 minutos en condiciones de viento nulo, con velocidad constante de 9 mph (15 kph).
- **Sistema de Posicionamiento Visual (VPS):** Incluye una cámara y un módulo de infrarrojos 3D ubicados en la parte inferior del dron, que permiten un vuelo estable y seguro. Particularmente efectivo a altitudes de entre 0.3 y 6 metros.
- **Cámara:** Resolución de 5 megapíxeles para imágenes (2592×1936 píxeles) y video en calidad 720p, con transmisión en tiempo real a través de la aplicación Tello.

- **Tarjeta programable de código abierto:** Incluye un módulo Wi-Fi de doble banda (2.4/5 GHz), Bluetooth, y es compatible con plataformas como Arduino, Scratch, y MicroPython. Ofrece múltiples puertos de expansión (UART, I2C, GPIO, PWM, SPI) para la conexión y programación de módulos adicionales.
- **Kit de expansión:** Permite la adición de accesorios y la programación de nuevas funcionalidades para proyectos DIY (Do It Yourself).
- **Batería de vuelo:** 3.8 V, 1100 mAh con protección de carga y descarga.
- **Modos de vuelo inteligentes:** cuenta con funciones de maniobras automáticas que permiten explorar diferentes aplicaciones de vuelo.

3.1.2. Capacidades de vuelo y control

En cuanto a sus capacidades de vuelo, el RoboMaster TT ofrece dos modos de velocidad: el modo lento, con una inclinación máxima de 9° y una velocidad de hasta 10.8 km/h, y el modo rápido, con una inclinación máxima de 25° y una velocidad máxima de 28.8 km/h, más adecuado para usuarios más experimentados o condiciones de vuelo en exteriores. Cuenta además con el (VPS) descrito en el apartado de 2.1 Descripción del dron Robomaster TT que permite al dron mantener su posición.

Entre las capacidades avanzadas del RoboMaster TT se incluyen modos de vuelo inteligentes como Bounce Mode, donde el dron vuela automáticamente hacia arriba y hacia abajo dentro de un rango de altura específico, y 8D Flips, que permite al dron realizar volteretas en ocho direcciones diferentes. Además, cuenta con el modo Throw & Go, que permite lanzar el dron al aire para que inicie el vuelo automáticamente, y EZ Shots, que facilita la captura de videos con movimientos predefinidos como giros de 360° y vuelos en círculo.

El dron también incluye un sistema de Failsafe Protection, que inicia un aterrizaje automático si la señal del dispositivo móvil se pierde o si la aplicación se cierra inesperadamente. En caso de que el VPS no esté disponible, el dron cambia automáticamente al modo Attitude (ATTI), en el cual no puede posicionarse y es más susceptible a factores ambientales como el viento.

3.1.3. Control remoto del dron Robomaster TT

El RoboMaster TT ofrece múltiples opciones para el control remoto, lo que lo convierte en una plataforma altamente versátil tanto para usuarios principiantes como avanzados. El dron puede ser manejado de forma directa a través de dos aplicaciones móviles distintas y mediante la conexión a un ordenador, lo que amplía significativamente sus capacidades y aplicaciones.

En primer lugar, el control convencional del dron se realiza mediante la aplicación Tello, disponible para dispositivos iOS y Android. Esta aplicación proporciona una interfaz intuitiva que permite gestionar todos los aspectos del vuelo, desde el despegue y el aterrizaje, hasta la captura de fotos y videos. Además, la aplicación permite ejecutar maniobras automatizadas, como los 8D Flips y el Bounce Mode, con solo unos toques en la pantalla. Esta facilidad de uso es ideal para principiantes y para aquellos que buscan realizar vuelos básicos sin necesidad de programación adicional.

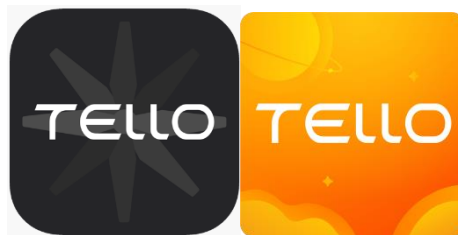


FIGURA 3.1-2 TELLO APP Y TELLO EDU APP

Además del control a través de la aplicación Tello, el RoboMaster TT incluye una segunda aplicación, Tello EDU (Figura 3.1-2), enfocada en la programación mediante un SDK basado en bloques. Esta aplicación permite a los usuarios programar el dron utilizando un entorno visual intuitivo, donde se pueden crear secuencias de comandos arrastrando y conectando bloques que representan diferentes funciones del dron. Este enfoque es especialmente valioso en entornos educativos, ya que facilita la enseñanza de conceptos de programación y lógica sin la necesidad de escribir código complejo. La aplicación está diseñada para usuarios que desean explorar la robótica y la programación de una manera práctica y accesible. Esta aplicación tiene una apariencia infantil, amigable y colorida, pensada para atraer a un público más joven que puede verla como un juego de móvil.

El RoboMaster TT ofrece la opción de controlarlo mediante programación de su tarjeta ESP32, lo que aporta una gran flexibilidad y permite personalizar sus funciones según las necesidades del proyecto. Hay dos principales formas de implementar este control. La primera es mediante la carga directa de código en la tarjeta ESP32 a través de Arduino IDE, que permite cargar códigos que responden a distintos parámetros de entrada, como datos de sensores o señales de control. Esto hace que el dron pueda realizar funciones preprogramadas sin requerir intervención continua. La segunda es a través de un ordenador, donde se pueden enviar comandos al dron mediante la red Wi-Fi utilizando lenguajes de programación como Python. Esta opción permite un control más dinámico y la posibilidad de hacer ajustes sobre la marcha, aunque no permite control manual en tiempo real, es decir, no es posible dirigir el dron de forma inmediata como con un control remoto convencional.

En este proyecto, se utilizó una combinación de ambas opciones para personalizar el comportamiento del dron. Con la tarjeta ESP32, se cargaron códigos que le permitían al dron reaccionar automáticamente a ciertos parámetros, como las coordenadas GPS, posibilitando vuelos autónomos. Y desde el ordenador, se enviaron comandos en Python mediante Wi-Fi para automatizar las rutas de vuelo. Esta configuración permitió programar y gestionar vuelos predefinidos para que el dron pudiera seguir trayectorias automatizadas sin intervención constante.

El control remoto del RoboMaster TT es altamente flexible, permitiendo tanto un manejo sencillo a través de aplicaciones móviles como una programación avanzada a través de un ordenador, lo que la hace una muy buena herramienta para el aprendizaje y la exploración en robótica y programación.

3.1.4. Precio de Mercado de Robomaster TT

En términos económicos, el RoboMaster TT tiene un precio base entre 100 y 140 euros, lo que lo convierte en una opción bastante accesible. Sin embargo, el costo total puede llegar a los 260 euros al añadir el kit de expansión, que cuesta unos 125 euros. Este dron fue elegido por las facilidades de programación que ofrece su kit de

expansión, que permite añadir funciones avanzadas y sensores de manera sencilla, lo que lo hace ideal para cumplir los objetivos del proyecto. (DJI TIENDA, 2024)

En comparativa con los mejores drones por debajo de 500 euros que ya poseen GPS integrado:

DJI MINI: 449 euros

Este dron es una opción excelente para fotógrafos y entusiastas de los drones. Incluye GPS para vuelos estables y seguros, una cámara 4K con gimbal de 3 ejes, y un alcance de hasta 10 km. Además, es ligero y compacto, lo que lo hace fácil de transportar. (DronesGator, 2024)

Holy Stone HS720E 400-500 euros

Viene con una cámara 4K, un gimbal de 2 ejes y GPS, lo que le permite realizar vuelos muy estables. Una de sus grandes ventajas es la duración de vuelo, con hasta 46 minutos usando dos baterías. (DRONELLER, 2024)

Bwine F7GB2 470-500 euros

Este dron destaca por su larga duración de vuelo y una excelente cámara 4K con estabilización. También cuenta con GPS y funciones inteligentes como Follow Me y retorno automático. (WayfareWithPierre, 2024)

Los modelos con GPS suelen incluir cámaras avanzadas y otras funciones que elevan su precio, pero que no son necesarias para tareas de navegación y control autónomo. El RoboMaster TT, con su kit de expansión, ofrece una solución más económica y perfectamente adecuada para proyectos experimentales.

3.2. Principios del Sistema de Posicionamiento Global (GPS)

El Sistema de Posicionamiento Global (GPS) es una tecnología fundamental en la navegación moderna, permitiendo la determinación precisa de la ubicación geográfica en cualquier parte del mundo. Este sistema ha revolucionado diversas industrias, desde la aviación y la navegación marítima hasta aplicaciones cotidianas como la conducción y el uso de dispositivos móviles. En el contexto del proyecto con el RoboMaster TT, la integración de un módulo GPS es crucial para mejorar las capacidades de vuelo autónomo y la navegación precisa del dron.

3.2.1. Funcionamiento básico del GPS

El GPS es un sistema de navegación basado en satélites que permite a un receptor calcular su posición en la Tierra con una precisión de hasta pocos metros. El GPS está compuesto por una constelación de al menos 24 satélites que orbitan la Tierra, transmitiendo señales de radio que incluyen la posición del satélite y el tiempo exacto de la transmisión. Estos satélites están distribuidos de manera que, en cualquier momento, un receptor en la Tierra puede captar señales de al menos cuatro satélites. (Hofmann-Wellenhof, 2008)

El receptor GPS utiliza la técnica de trilateración para determinar su posición exacta. Al recibir señales de múltiples satélites, el receptor mide el tiempo que tarda la señal en viajar desde el satélite hasta el receptor, y a partir de esa información calcula la distancia a cada satélite. Conociendo la distancia a al menos cuatro satélites, el receptor puede calcular su posición exacta en términos de latitud, longitud y altitud. (El-Rabbany, 2002)

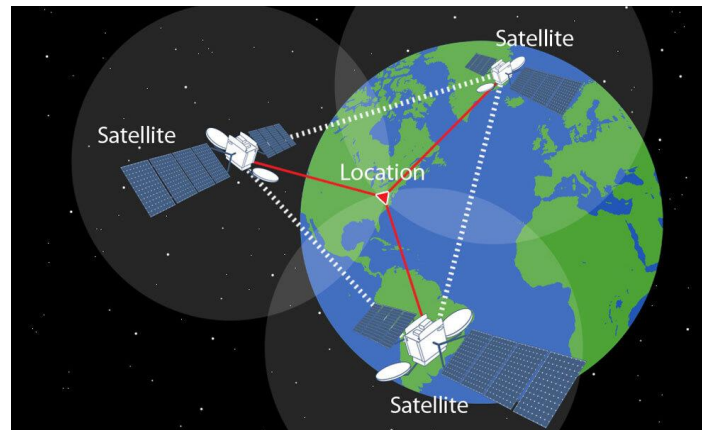


FIGURA 3.2-1 REPRESENTACIÓN GRÁFICA DE LA TRILATERACIÓN

3.2.2. Comunicación y protocolo NMEA

El Protocolo NMEA es un estándar de comunicación utilizado para transmitir información de navegación y GPS entre dispositivos electrónicos. El protocolo NMEA es esencial en la integración de módulos GPS con otros sistemas, como el controlador de vuelo del dron o un ordenador para procesar los datos de ubicación. (National Marine Electronics Association, 2002)

Los datos NMEA se transmiten en forma de sentencias ASCII que contienen información como la posición, la velocidad, la hora, la calidad de la señal, y otros parámetros de navegación. Las sentencias más comunes incluyen \$GPGGA, y \$GPRMC. (National Marine Electronics Association, 2002)

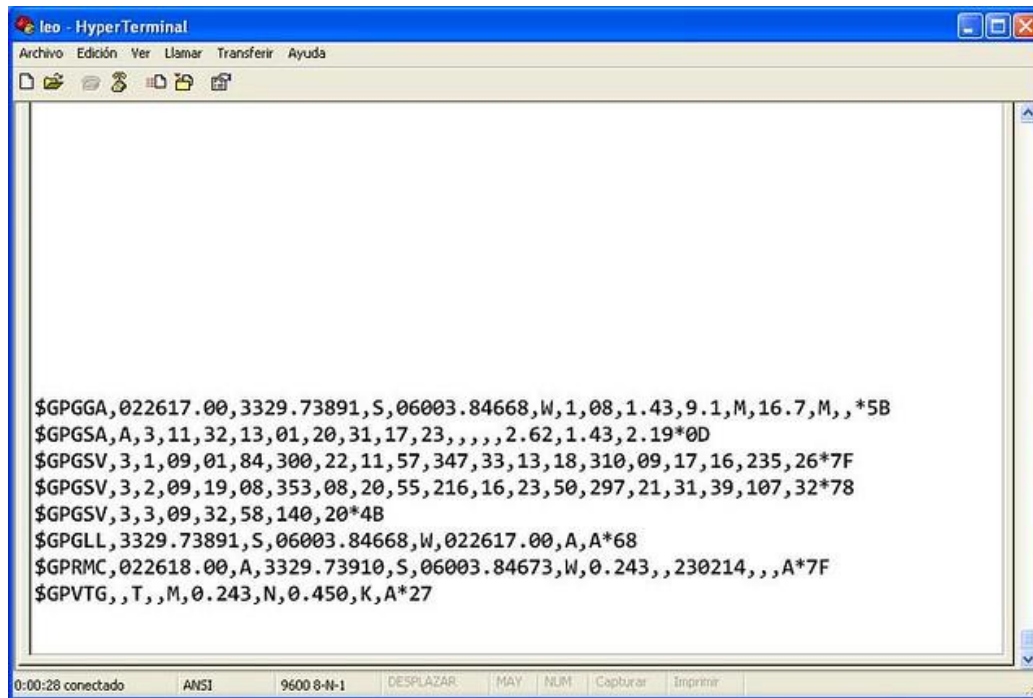
A screenshot of a HyperTerminal window titled 'leo - HyperTerminal'. The window displays several NMEA sentences received from a device. The sentences are: \$GPGGA,022617.00,3329.73891,S,06003.84668,W,1,08,1.43,9.1,M,16.7,M,,*5B; \$GPGSA,A,3,11,32,13,01,20,31,17,23,,,,,2.62,1.43,2.19*0D; \$GPGSV,3,1,09,01,84,300,22,11,57,347,33,13,18,310,09,17,16,235,26*7F; \$GPGSV,3,2,09,19,08,353,08,20,55,216,16,23,50,297,21,31,39,107,32*78; \$GPGSV,3,3,09,32,58,140,20*4B; \$GPGLL,3329.73891,S,06003.84668,W,022617.00,A,A*68; \$GPRMC,022618.00,A,3329.73910,S,06003.84673,W,0.243,,230214,,,A*7F; and \$GPVTG,,T,,M,0.243,N,0.450,K,A*27. The status bar at the bottom shows '0:00:28 conectado', 'ANSI', '9600 8-N-1', and buttons for 'DESPLAZAR', 'MAY', 'NUM', 'Capturar', and 'Imprimir'.

FIGURA 3.2-2 SENTENCIAS DE DATOS NMEA

En el proyecto con el RoboMaster TT, la decodificación de las sentencias NMEA (FIGURA 3.2-2) será un paso clave para procesar los datos GPS recibidos y utilizarlos en la navegación del dron. Los datos decodificados permitirán al dron ajustar su posición y trayectoria de vuelo en tiempo real, basándose en la información precisa proporcionada por el módulo GPS.

A continuación, se explican las sentencias NMEA principales y cómo se transforman en datos legibles:

\$GPGGA: Proporciona datos básicos de posición.

- **022617.00**: Hora en formato UTC (02:26:17).
- **3329.73891,N**: Latitud (33 grados, 29.73891 minutos Norte).
- **06003.84668,W**: Longitud (60 grados, 03.84668 minutos Oeste).
- **1**: Indicador de calidad de la señal GPS (1 significa posición GPS fija).
- **08**: Número de satélites en uso.
- **1.43**: Dilución de precisión horizontal (HDOP), un valor más bajo indica mejor precisión.
- **9.1,M**: Altura sobre el nivel del mar (9.1 metros).

\$GPRMC: Proporciona datos mínimos recomendados de navegación.

- **022618.00**: Hora en UTC (02:26:18).
- **A**: Estado de la señal (A significa que la señal es válida).
- **3329.73910,N**: Latitud.
- **06003.84673,W**: Longitud.
- **0.243**: Velocidad sobre el suelo en nudos (0.243 nudos).
- **230214**: Fecha (23 de febrero de 2014).

Para la transformación en datos legibles, se puede hacer lo siguiente:

La latitud y longitud se dividen en grados y minutos, que luego se convierten a grados decimales. Por ejemplo:

Latitud: 33 grados, 29.73891 minutos Norte (N)

$$= 33 + (29.73891 / 60) = +33.4956485^\circ.$$

Longitud: 60 grados, 03.84668 minutos Oeste (W)

$$= 60 + (03.84668 / 60) = - 60.0641113^\circ.$$

La velocidad en nudos (0.243 nudos) se puede convertir a kilómetros por hora multiplicando por 1.852. Por ejemplo, 0.243 nudos = 0.450 km/h.

La capacidad de interpretar y manejar los datos NMEA permite que el dron interactúe con otros sistemas de control y visualización, como la interfaz gráfica desarrollada en Processing, que utiliza la información GPS para trazar rutas y controlar el vuelo de manera autónoma.

3.3. Microcontroladores y la tarjeta ESP32

3.3.1. Características y funciones de la ESP32-D2WD

La ESP32-D2WD es una variante específica de la familia de microcontroladores ESP32, desarrollada por Espressif Systems, que destaca por su balance entre potencia de procesamiento y eficiencia energética. Esta tarjeta está diseñada para

aplicaciones que requieren un rendimiento sólido en tareas de procesamiento, conectividad inalámbrica y bajo consumo energético, lo que la convierte en una excelente opción para proyectos de IoT, robótica y automatización. La información de este apartado ha sido obtenida de (ESPRESSIF, 2023). (FIGURA 3.3-1)

Entre las características principales de la ESP32-D2WD se incluyen:

Procesador Dual-Core: La ESP32-D2WD está equipada con un procesador de doble núcleo que opera a una frecuencia principal de 160 MHz. Este procesador Xtensa LX6 es capaz de manejar múltiples tareas en paralelo, lo que lo hace ideal para aplicaciones que requieren procesamiento en tiempo real. La capacidad de cálculo de esta unidad alcanza los 400 MIPS (Millones de Instrucciones por Segundo), lo que le otorga un rendimiento considerable para aplicaciones exigentes.

Conectividad Inalámbrica: Al igual que otras variantes de ESP32, la D2WD incluye conectividad Wi-Fi integrada, compatible con los estándares 802.11 b/g/n, y Bluetooth, que soporta tanto Bluetooth clásico como Bluetooth Low Energy (BLE). Esta conectividad permite que la ESP32-D2WD se comuniquen fácilmente con otros dispositivos en una red local o a través de internet, lo que es crucial para aplicaciones de IoT y control remoto.

Memoria Integrada: Una de las diferencias clave de la ESP32-D2WD es su memoria flash interna de 2 MB, que ofrece suficiente espacio para el almacenamiento de firmware y aplicaciones. Además, la tarjeta incluye 520 KB de SRAM, que es utilizada para el almacenamiento de datos durante la ejecución del programa.

Amplia Gama de Periféricos: La ESP32-D2WD cuenta con múltiples pines GPIO (General Purpose Input/Output) configurables para diversas funciones, como entradas/salidas digitales, PWM, y ADC (Conversión Analógica a Digital). También soporta interfaces de comunicación como UART, SPI, e I2C, lo que facilita la conexión con una variedad de sensores, actuadores y otros módulos.

Bajo Consumo Energético: La ESP32-D2WD está diseñada para ser eficiente en términos de energía, con varios modos de bajo consumo que permiten al dispositivo operar durante largos períodos de tiempo en aplicaciones alimentadas por batería. Estos modos incluyen el modo de sueño profundo (Deep Sleep) y el modo de

suspensión (Light Sleep), que ayudan a minimizar el consumo de energía cuando el dispositivo no está activo.

Soporte para Programación: Esta tarjeta es compatible con varios entornos de desarrollo, incluyendo el Arduino IDE, ESP-IDF (Espressif IoT Development Framework), y MicroPython. Esto permite a los desarrolladores elegir la plataforma de programación que mejor se adapte a sus necesidades y maximizar el potencial de la ESP32-D2WD en sus proyectos.

3.3.2. Conexiones y comunicación con periféricos

La ESP32-D2WD ofrece una amplia gama de opciones de conexión y comunicación con periféricos, lo que la convierte en una plataforma extremadamente versátil para el diseño de sistemas embebidos. A continuación, se describen las principales formas de conexión y comunicación:

Interfaz UART: La ESP32-D2WD incluye varias interfaces UART (Universal Asynchronous Receiver-Transmitter) que permiten la comunicación en serie con otros dispositivos como módulos GPS, GSM, y microcontroladores adicionales. Esta capacidad es esencial en proyectos donde se requiere la transmisión de datos en tiempo real y con baja latencia.

Interfaz SPI: La conexión a través de SPI (Serial Peripheral Interface) es ideal para la comunicación rápida entre la ESP32-D2WD y periféricos como pantallas, memorias, y sensores de alta velocidad. El protocolo SPI permite una transferencia de datos sincronizada y eficiente, esencial para aplicaciones que demandan alta velocidad de procesamiento.

Interfaz I2C: La ESP32-D2WD también soporta la interfaz I2C (Inter-Integrated Circuit), que es utilizada para conectar varios dispositivos utilizando solo dos líneas de comunicación (SDA y SCL). Esto es especialmente útil en proyectos con múltiples sensores y módulos donde el espacio y la simplicidad en el cableado son cruciales.

GPIO y PWM: Los pines GPIO de la ESP32-D2WD pueden ser configurados para leer señales digitales y analógicas, así como para generar señales PWM (Pulse

Width Modulation). Esta capacidad es clave para el control de dispositivos como motores, LEDs, y otros componentes electrónicos que requieren variación de señal para su operación.

Conexión a la Nube: La conectividad Wi-Fi integrada permite a la ESP32-D2WD conectarse a servicios en la nube para enviar y recibir datos, una característica esencial para aplicaciones de IoT. Esto permite el monitoreo y control remoto, actualizaciones de firmware, y la integración con otros dispositivos a través de internet.

Bluetooth y BLE: La capacidad de comunicación a través de Bluetooth y BLE permite que la ESP32-D2WD se conecte de manera inalámbrica con dispositivos cercanos, como smartphones, ordenadores y otros microcontroladores. Esto es especialmente útil en aplicaciones donde se necesita una conexión de corto alcance y bajo consumo energético.

4. Selección y Adquisición del Módulo GPS

El primer paso en este proyecto fue la selección y adquisición de un módulo GPS que cumpliera con los requisitos técnicos y de integración necesarios para su funcionamiento óptimo en el dron Robomaster TT. Dado que el GPS es la pieza clave para la navegación autónoma del dron, su elección se realizó de manera cuidadosa, teniendo en cuenta una serie de criterios técnicos y de compatibilidad.

4.1. Identificación de Requisitos del Módulo GPS

Antes de proceder a la selección del módulo GPS, se llevó a cabo un análisis exhaustivo de los requisitos que debía cumplir el módulo GPS para asegurar su correcta integración y funcionamiento con el dron Robomaster TT. Este análisis incluyó la evaluación de los siguientes aspectos:

COMPATIBILIDAD CON EL DRON ROBOMASTER TT: El dron cuenta con un kit de expansión que permite la conexión de dispositivos externos a través de puertos I/O. Es obligatorio que el GPS seleccionado sea compatible con los protocolos de comunicación soportados por estos puertos, como UART, I2C, PWM y SPI. Además, el módulo GPS debía operar dentro de los parámetros de alimentación del kit de

expansión, que ofrece tensiones de salida de $4,8V \pm 0,2V$ para 5V y $3,3V \pm 0,1V$ para 3V, con una corriente máxima de salida de 800mA.

PRECISIÓN EN LA POSICIÓN Y ORIENTACIÓN: La exactitud en la determinación de la ubicación geográfica (latitud, longitud, altitud) y la orientación del dron (rumbo) es vital para el éxito del proyecto. Se requiere un módulo GPS capaz de proporcionar datos de posición precisos con un margen de error mínimo, para garantizar que el dron pueda seguir las rutas programadas sin desviaciones significativas.

CONSUMO ENERGÉTICO: Considerando las limitaciones de la batería del dron, es necesario que el módulo GPS seleccionado tenga un consumo energético bajo. Un consumo elevado podría reducir significativamente la duración del vuelo, comprometiendo las operaciones del dron durante misiones prolongadas.

NÚMERO DE SATÉLITES Y CANALES SOPORTADOS: Un mayor número de satélites y canales en los que el GPS pueda operar es directamente proporcional a la precisión y estabilidad de la señal. Esto es especialmente importante en entornos donde la señal satelital puede ser débil o interrumpida, como áreas urbanas con muchos edificios.

ANTENA GPS INCORPORADA: La inclusión de una antena GPS integrada en el módulo simplifica la instalación y reduce la complejidad del diseño del sistema. Una antena de alta calidad asegura una buena recepción de la señal GPS, incluso en condiciones adversas.

PESO Y TAMAÑO: Dado que el dron tiene limitaciones en cuanto a la carga útil que puede transportar, es fundamental que el módulo GPS sea ligero y compacto. Un módulo de gran tamaño o peso podría afectar negativamente el rendimiento del dron, reduciendo su agilidad y tiempo de vuelo.

COMPATIBILIDAD CON SISTEMAS GNSS ADICIONALES: Aunque el sistema GPS es ampliamente utilizado, la capacidad de operar con otros sistemas GNSS como GLONASS (Rusia), Galileo (Europa) o BeiDou (China) es deseable. Esto ofrece mayor redundancia y mejora la precisión y fiabilidad en diversas condiciones operativas.

SENSIBILIDAD DE ADQUISICIÓN Y SEGUIMIENTO: La sensibilidad del receptor GPS en la adquisición inicial de la señal y su capacidad para mantener la conexión de manera estable son factores críticos, especialmente en entornos donde las señales pueden ser débiles o sujetas a interferencias.

FRECUENCIA DE ACTUALIZACIÓN: Este parámetro determina cuán frecuentemente se actualizan los datos de posición. Una mayor frecuencia de actualización es esencial para obtener datos en tiempo real, lo que mejora la capacidad de respuesta del dron durante las operaciones de navegación.

4.2. Comparativa de Módulos GPS Disponibles en el Mercado

Una vez identificados los requisitos esenciales, se procedió a evaluar diferentes módulos GPS disponibles en el mercado. Esta evaluación se realizó comparando las características técnicas, el rendimiento y el costo de cada módulo (Figura 4.2-1).



FIGURA 4.2-1 MERCADO

4.2.1. Análisis de Características Técnicas

Se evaluaron seis módulos GPS principales, considerando los criterios mencionados anteriormente:

ADAFRUIT: Este módulo ofrece una frecuencia de salida de entre 1 a 10 Hz, soporta entre 22 y 66 satélites y canales, y tiene un error de posicionamiento de aproximadamente 3 metros. Con un consumo de tracking de 25 mA y una sensibilidad de adquisición de -145 dBm, es un módulo adecuado para aplicaciones generales, aunque su precisión podría ser mejorada. (BricoGeek, 2024)

GPS GLONASS: Con una frecuencia de salida similar (1 a 10 Hz) y soporte para hasta 72 canales, este módulo destaca por su bajo error de posicionamiento de 2 metros y su sensibilidad de adquisición de -148 dBm. Sin embargo, su consumo energético es algo más elevado (50 mA), lo que podría impactar en la duración del vuelo. (BricoGeek 2. , 2024)

SPARKFUN ZOE: Este módulo ofrece una alta frecuencia de salida de 18 Hz, lo que lo hace ideal para aplicaciones que requieren datos de posición en tiempo real. Soporta hasta 72 canales y tiene un error de posicionamiento de 2,5 metros, pero no incluye una antena integrada y su sensibilidad de adquisición no es especificada. (BricoGeek 3. , 2024)

SPARKFUN SAM-M10Q: Con una frecuencia de salida variable de 5 a 18 Hz, este módulo ofrece un error de posicionamiento más bajo (1,5 metros) y un consumo energético muy reducido (10 mA en tracking). Su sensibilidad de adquisición es de -160 dBm, lo que lo hace altamente eficiente en la captura de señales satelitales débiles. (Sparkfun, 2024)

FOXEEER M10Q 250: Este módulo combina una frecuencia de salida de 1 a 10 Hz con soporte para 72 canales y un error de posicionamiento de 1,5 metros. Su bajo precio y la alta sensibilidad de adquisición (-160 dBm) lo hacen una opción atractiva para proyectos que requieren precisión y eficiencia energética. (IHA-RACE, 2024)

MATEKSYS M10Q-5883: Este módulo ofrece una frecuencia de salida de 5 Hz y soporte para múltiples sistemas GNSS, incluyendo GPS, GLONASS, Galileo y BeiDou. Con un error de posicionamiento de aproximadamente 1.5 metros y un consumo de tracking de 13 mA, es una opción eficiente en términos de energía. Incluye una antena integrada y es apto para aplicaciones donde la precisión y la compatibilidad con varios sistemas de navegación satélite son críticas. (IHA-RACE 2. , 2024)

4.2.2.Relación Calidad-Precio

La relación calidad-precio es un aspecto determinante en la selección del módulo GPS. En este caso, se evaluó tanto el costo económico como las prestaciones técnicas de cada módulo.

ADAFRUIT y GPS GLONASS ofrecen precios competitivos, pero con ciertas limitaciones en cuanto a precisión y consumo energético.

SPARKFUN ZOE y SPARKFUN SAM-M10Q presentan mejores características técnicas, pero a un costo más elevado.

MATEKSYS M10Q-5883 se posiciona como una opción sólida que combina eficiencia energética, soporte para múltiples sistemas GNSS y un precio promedio a su calidad, lo que lo convierte en una elección atractiva para quienes buscan un rendimiento confiable sin incurrir en costos excesivos.

FOXEEER M10Q 250, por otro lado, ofrece un buen equilibrio entre costo y rendimiento. Con un precio considerablemente más bajo que algunos de sus competidores, este módulo no solo cumple con los requisitos técnicos sino que también garantiza una alta precisión y bajo consumo energético

4.3. Justificación de la Elección del Módulo GPS

Tras una evaluación detallada de los módulos GPS disponibles en el mercado, la elección final recayó en el **Foxeer M10Q 250 GPS 5883 Compass**. Esta decisión se fundamentó en varios aspectos clave:

Compatibilidad total: El Foxeer M10Q 250 es completamente compatible con los sistemas de expansión del dron Robomaster TT, asegurando una integración sin complicaciones y un rendimiento óptimo.

Alta Precisión: Con un error de posicionamiento de solo 1,5 metros y una alta sensibilidad de adquisición (-160 dBm), este módulo garantiza la precisión necesaria para las tareas de navegación del dron.

Bajo Consumo Energético: Considerando las limitaciones de la batería del dron, el bajo consumo del módulo Foxeer M10Q 250 es ideal para maximizar la duración del vuelo sin comprometer la funcionalidad.

Antena Integrada: La inclusión de una antena GPS de alta calidad simplifica la instalación y asegura una buena recepción de la señal en diversas condiciones operativas.

Relación Calidad-Precio: El módulo ofrece características superiores a un precio notablemente competitivo, lo que lo convierte en la opción más eficiente en términos de costo-rendimiento.

En resumen, el **Foxeer M10Q 250 GPS 5883 Compass** (Figura 4.3-1) fue seleccionado como el módulo GPS óptimo para este proyecto, ofreciendo un equilibrio perfecto entre precisión, eficiencia energética, compatibilidad y costo. Esta elección sienta las bases para un desempeño exitoso en las aplicaciones de navegación que se implementarán en el dron Robomaster TT.



FIGURA 4.3-1 FOXEEER M10Q 250 GPS 5883 COMPASS (IHA-RACE, 2024)

5. Integración del Módulo GPS con el Dron

5.1. Conexión del GPS al Módulo ESP32

5.1.1. Protocolo de Comunicación y Asignación de Pines

Tanto la tarjeta como el dron disponen de comunicación UART e I2C. La comunicación UART se utilizará para obtener la posición en términos de latitud y longitud, así como otros parámetros derivados, como la velocidad del dron, debido a su simplicidad y fiabilidad en la transmisión continua de datos. Por otro lado, la

comunicación I2C se encargará de obtener la orientación a través de una brújula integrada en el módulo GPS.

Los pines del GPS se conectarán mediante un cable propio del dispositivo a los pines del "Extension Board" del kit de expansión. En la comunicación UART, utilizaremos los pines TX y RX del GPS, que se conectarán respectivamente al pin RX3 y al pin TX3 del "Extension Board". Para la comunicación I2C, el pin SDA del GPS se conectará al SDA del "Extension Board", y el pin SCL al SDA correspondiente como se puede observar en FIGURA 5.1-1.

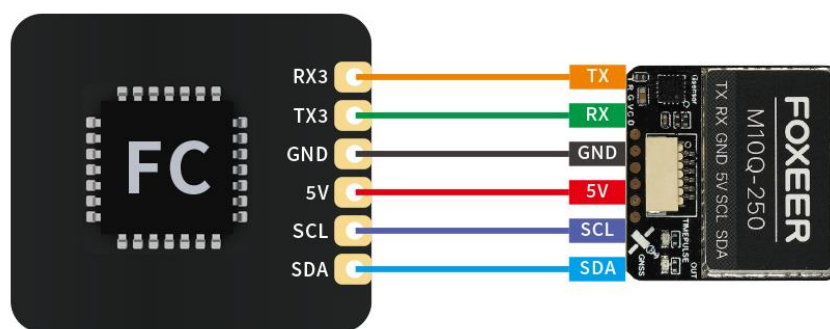


FIGURA 5.1-1 ESQUEMA DE CONEXIÓN DEL GPS CON UN EQUIPO EXTERNO. (IHA-RACE, 2024)

Además, los pines de alimentación del GPS se conectarán de la siguiente manera: el GND del GPS al GND del "Extension Board" y el pin de 5V al pin correspondiente de 5V en el "Extension Board". De esta manera, garantizamos una conexión adecuada y estable tanto para la alimentación como para la comunicación entre el GPS y el dron. En la FIGURA 5.1-2 puede apreciar la forma y distribución de pines en el "Extension Board" y sus respectivos nombres en la TABLA 5.1-1 Finalmente queda expuesto en la TABLA 5.1-2 la conexión entre pines.

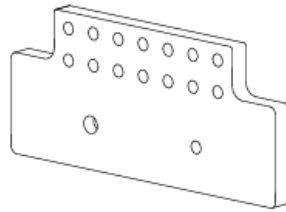


FIGURA 5.1-2 EXTENSION BOARD (DJI, 2021)

IO2	IO5	IO4	IO27	IO26	GND	3V3
IO12	IO21	IO15	IO14	IO13	GND	5V

TABLA 5.1-1 NOMBRE DE LOS PINES RESPECTIVOS DEL EXTENSION BOARD (DJI, 2021)

PIN DEL GPS	NOMBRE DEL PIN EN BOARD EXTENSION
TX	IO5
RX	IO21
GND	GND
5V	5V
SCL	IO26
SDA	IO27

TABLA 5.1-2 CONEXIÓN ENTRE PINES DEL GPS CON LOS PINES DEL EXTENSION BOARD

5.1.2.Diseño e implementación de la conexión física

Para adaptar el GPS a otros módulos ESP32 y asegurar la versatilidad en las pruebas, se añadió una extensión a los cables del GPS en el taller de la Universidad de Jaén. Estas extensiones, fabricadas de manera artesanal son conectores hembra.

El "Extension Board" tiene los pines en forma de agujeros (hembras), con la intención de facilitar la personalización de las conexiones FIGURA 5.1-2.

Dado que tanto los cables del GPS como los del "Extension Board" tienen conectores hembra, se procedió a soldar unos conectores macho a los pines del "Extension Board" FIGURA 5.1-3. Como resultado, el GPS ahora se conecta de manera firme y estable al "Extension Board", con el GPS utilizando conectores hembra y el "Extension Board" utilizando conectores macho FIGURA 5.1-4.



FIGURA 5.1-3 EXTENSION BOARD CON SOLDADURA DE PINES MACHO Y CONECTADO AL EXPANSION KIT

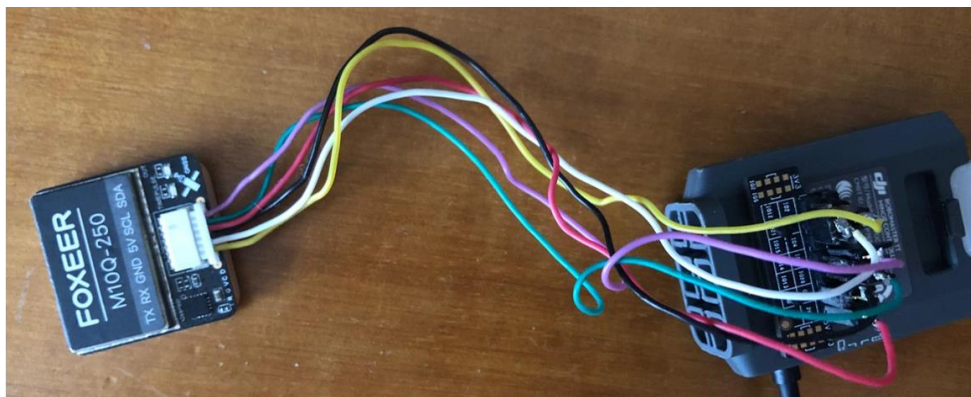


FIGURA 5.1-4 CABLEADO ENTRE EL GPS Y EL EXTENSION KIT

En la FIGURA 5.1-9 se puede apreciar el cableado final. Durante la soldadura, se utilizaron técnicas para asegurar una conexión segura y duradera, evitando posibles fallos en la comunicación durante el vuelo. Además, esta personalización permite que el sistema pueda ser fácilmente ajustado y probado con distintas configuraciones, aumentando la flexibilidad del diseño.

5.2. Montaje y Fijación Física del GPS en el Dron

Se utilizaron gomas elásticas para sujetar el GPS en la parte delantera del kit de expansión, de manera que la antena del GPS quedara alineada con la dirección en la que apunta el dron. La intención de colocar el GPS en esa posición fue mejorar la aerodinámica y mantener el módulo GPS-Tarjeta-Dron compacto, estable y equilibrado. (FIGURA 5.2-1)

Aunque la posición óptima del GPS es con la antena apuntando hacia arriba y los cables alineados en la dirección del dron, esta configuración no generó un impacto significativo en el proyecto.

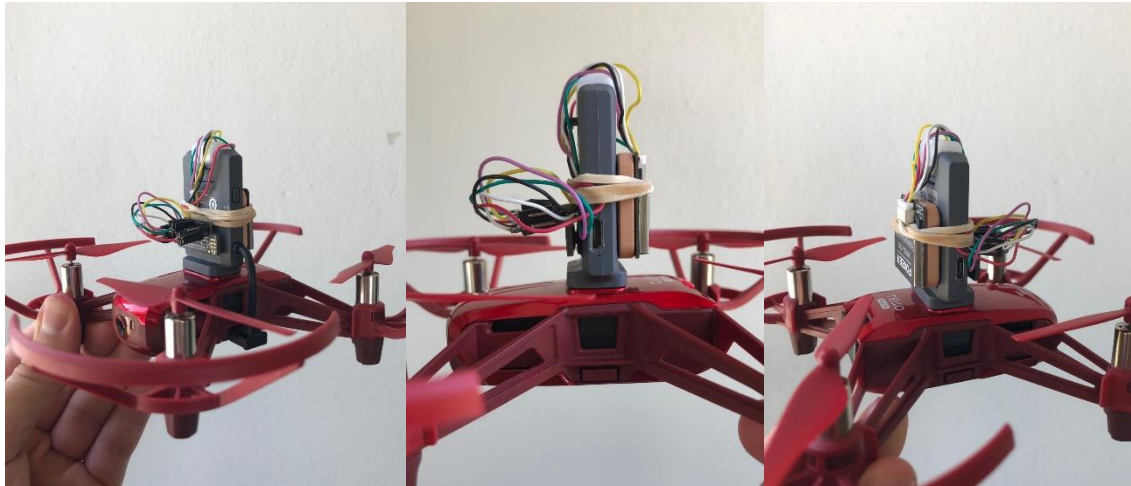


FIGURA 5.2-1 ACOPLAMIENTO FINAL ENTRE EL GPS, EL EXPANSION KIT Y EL DRON

6. Herramientas y entornos de programación utilizados

En este proyecto, se han utilizado tres principales entornos de programación: Arduino IDE, Python, y Processing (Figura 7.1-1). Estos tres entornos de programación han sido seleccionados por su capacidad para manejar eficientemente las tareas específicas del proyecto, desde la programación de hardware hasta la visualización de datos y la automatización de procesos.



FIGURA 5.2-2 LOGOS DE PYCHARM, ARDUINO IDE Y PROCESSING

6.1. Arduino IDE

Este entorno de desarrollo integrado es ampliamente utilizado para la programación de microcontroladores, como la ESP32 utilizada en este proyecto. El Arduino IDE es conocido por su facilidad de uso, su amplio soporte para una gran variedad de hardware, incluida la ESP32, y su comunidad activa que proporciona acceso a numerosas bibliotecas y ejemplos de código. Su principal ventaja es la simplicidad con la que permite escribir, compilar y cargar programas directamente en la tarjeta ESP32, lo que facilita enormemente la integración y la gestión de dispositivos como módulos GPS y otros sensores conectados a la tarjeta.

6.2. PyCharm (Python)

Aunque Python es el lenguaje de programación utilizado, el entorno en el que se ejecuta y gestiona el código es PyCharm. PyCharm proporciona un entorno de desarrollo más robusto y eficiente para escribir, depurar y ejecutar scripts Python. En este proyecto, PyCharm se utilizó para recibir, procesar y manipular los datos GPS enviados por la ESP32. Además, PyCharm facilita la interacción con el dron RoboMaster TT, permitiendo enviar comandos al dron directamente desde scripts Python. Los datos GPS recibidos se envían a Processing para su visualización, y PyCharm maneja la interpretación de los datos desde Processing para generar y enviar órdenes al dron. El uso de PyCharm ofrece un entorno optimizado para el desarrollo de software, con herramientas avanzadas para el manejo de datos en tiempo real, análisis de datos, y comunicación con otros sistemas, haciendo que el trabajo con Python sea aún más poderoso y versátil.

6.3. Processing

Processing es un entorno de programación basado en Java, diseñado para facilitar la creación de aplicaciones visuales e interactivas. En este proyecto, Processing fue utilizado para el desarrollo de la interfaz gráfica y la visualización de datos en tiempo real, como la visualización de la ubicación y la trazabilidad de rutas GPS. La ventaja de Processing radica en su enfoque en la visualización de datos y gráficos, ofreciendo herramientas sencillas para crear visualizaciones complejas con

relativamente poco esfuerzo de programación en comparación con otros entornos de desarrollo gráfico.

6.4. Software adicional y servicios utilizados

Además de los entornos de programación, se utilizaron varias herramientas y servicios adicionales para completar las tareas específicas del proyecto.

SDK DE TELLO:

Un SDK (Software Development Kit) es un conjunto de herramientas, bibliotecas, APIs, documentación y ejemplos de código que los desarrolladores utilizan para crear aplicaciones para una plataforma específica o para interactuar con un sistema, servicio, o hardware específico. A groso modo, SDK es un paquete que proporciona todo lo necesario para que un desarrollador pueda integrar y utilizar ciertas funcionalidades en sus aplicaciones.

El SDK de Tello es un conjunto de herramientas y comandos diseñados para que los desarrolladores puedan controlar y programar los drones Tello y Tello Talent de DJI. Esta plataforma permite a los desarrolladores interactuar con el dron enviando comandos a través de un protocolo UDP. Estos comandos gestionan funciones del dron como despegue, aterrizaje, movimiento en diferentes direcciones, y captura de imágenes. Además, el SDK proporciona acceso a datos de telemetría importantes del dron, como la velocidad, altitud, nivel de batería, y otros parámetros. Gracias a estas capacidades, se puede personalizar el comportamiento del dron para adaptarlo a diferentes aplicaciones, como los objetivos del proyecto.



FIGURA 6.4-1 LOGO DE OPENSTREETMAP

API DE OPENSTREETMAP:

OpenStreetMap es una plataforma colaborativa que proporciona datos geográficos libres y de código abierto. En este proyecto, OpenStreetMap fue fundamental para obtener datos cartográficos precisos y actualizados que sirvieron como base para la visualización de rutas GPS. La plataforma permite acceder a mapas detallados de prácticamente cualquier lugar del mundo, lo que resulta importante para proyectos que requieren información geoespacial. La API de OpenStreetMap fue utilizada junto con Processing para trazar y visualizar las rutas en tiempo real, aprovechando la flexibilidad y la riqueza de datos que ofrece la plataforma.

MAPERITIVE:

Maperitive es una herramienta que utiliza datos de OpenStreetMap para generar imágenes de mapas personalizados. En este proyecto, Maperitive fue empleada para crear mapas que luego fueron integrados en la interfaz de visualización desarrollada en Processing. La capacidad de Maperitive para personalizar y exportar mapas en diversos formatos permitió ajustar las visualizaciones a las necesidades específicas del proyecto, facilitando así la interpretación y el seguimiento de las rutas GPS trazadas. Su integración con OpenStreetMap asegura que los mapas utilizados estén siempre actualizados y sean precisos.

7. Metodología

En esta sección se expone por pasos cómo se desarrolló la programación necesaria para el éxito del proyecto. Se exponen configuraciones necesarias, la inclusión de librerías específicas, y sistemas de comunicación de datos entre dispositivos y los distintos softwares involucrados en el proyecto. El código fuente completo de cada entorno de programación queda recogido en el ANEXO III.

7.1. Arquitectura del sistema

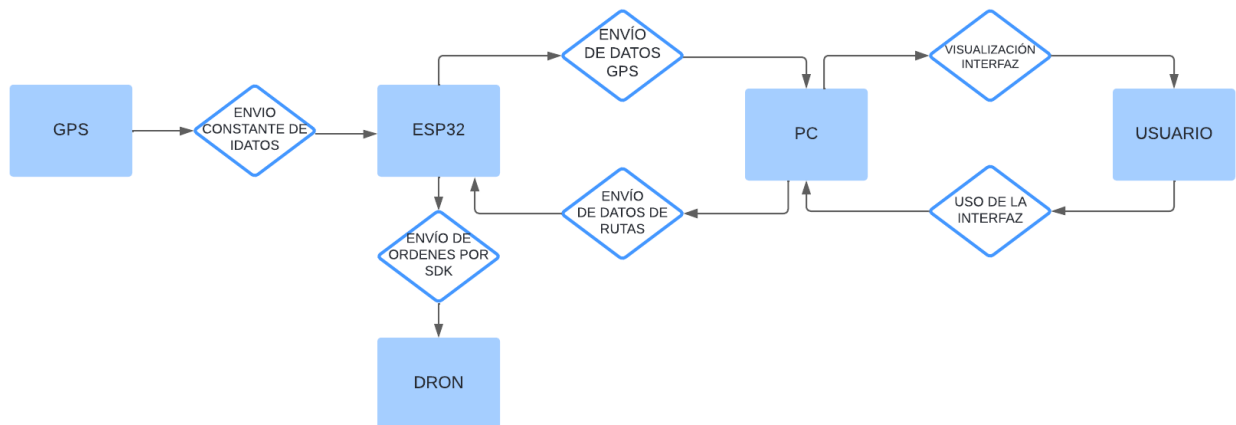


FIGURA 7.1-1 ESQUEMA DE LA ARQUITECTURA DEL SISTEMA

El GPS envía datos de posicionamiento constantemente a la tarjeta ESP32, que actúa como intermediaria. Estos datos de GPS se envían hacia el PC, donde son procesados y visualizados mediante una interfaz, permitiendo al usuario monitorizar el vuelo del dron en tiempo real.

Una vez el usuario ha definido las rutas o ha dado las órdenes necesarias, el PC envía la información de las rutas nuevamente a la ESP32, que transmite las órdenes al dron utilizando el SDK. Esto permite que el dron ejecute los vuelos de manera autónoma, siguiendo las rutas establecidas previamente. De este modo, el flujo de datos funciona de forma paralela mientras sigue un ciclo, asegurando una comunicación constante entre todos los componentes del sistema. (FIGURA 7.1-1)

7.2. Programación de la tarjeta ESP32

1. Configuración del entorno Arduino IDE

El entorno de desarrollo Arduino IDE es necesario para escribir y cargar el código en microcontroladores compatibles, como el ESP32. Este entorno permite escribir el código en C/C++ y cargarlo directamente en el microcontrolador. En esta configuración, es importante seleccionar la placa correcta (por ejemplo, ESP32 Dev Module para esta tarjeta) y el puerto COM correspondiente.

Para ello es necesario instalar las tarjetas disponibles de Arduino IDE de Espressif Systems de ESP32. (Figura 7.2-1)

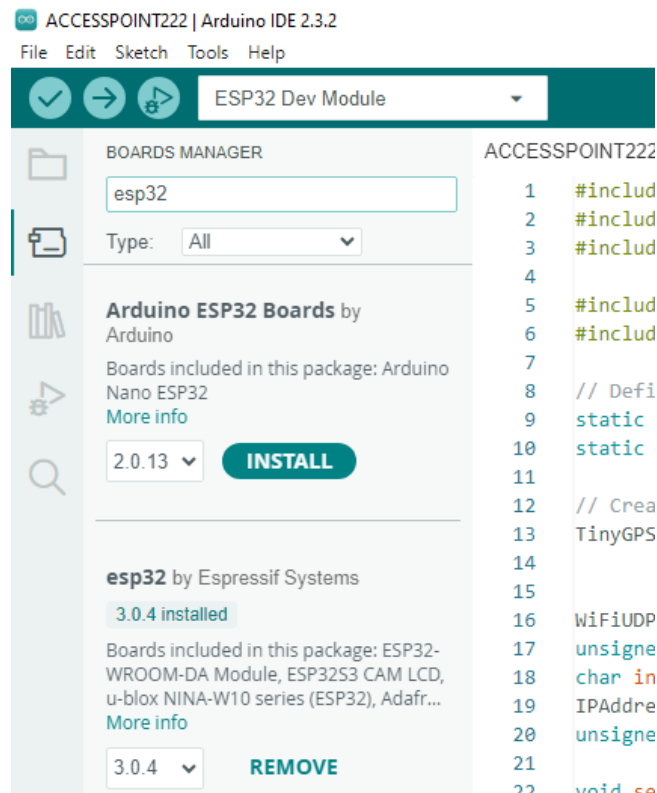


FIGURA 7.2-1 INSTALACIÓN DE LAS TARJETAS ESP32 DE ESPRESSIF SYSTEMS EN ARDUINO IDE

Además, se deben configurar otros parámetros específicos de la tarjeta dentro del IDE para asegurar una carga exitosa del código. (Figura 7.2-2)

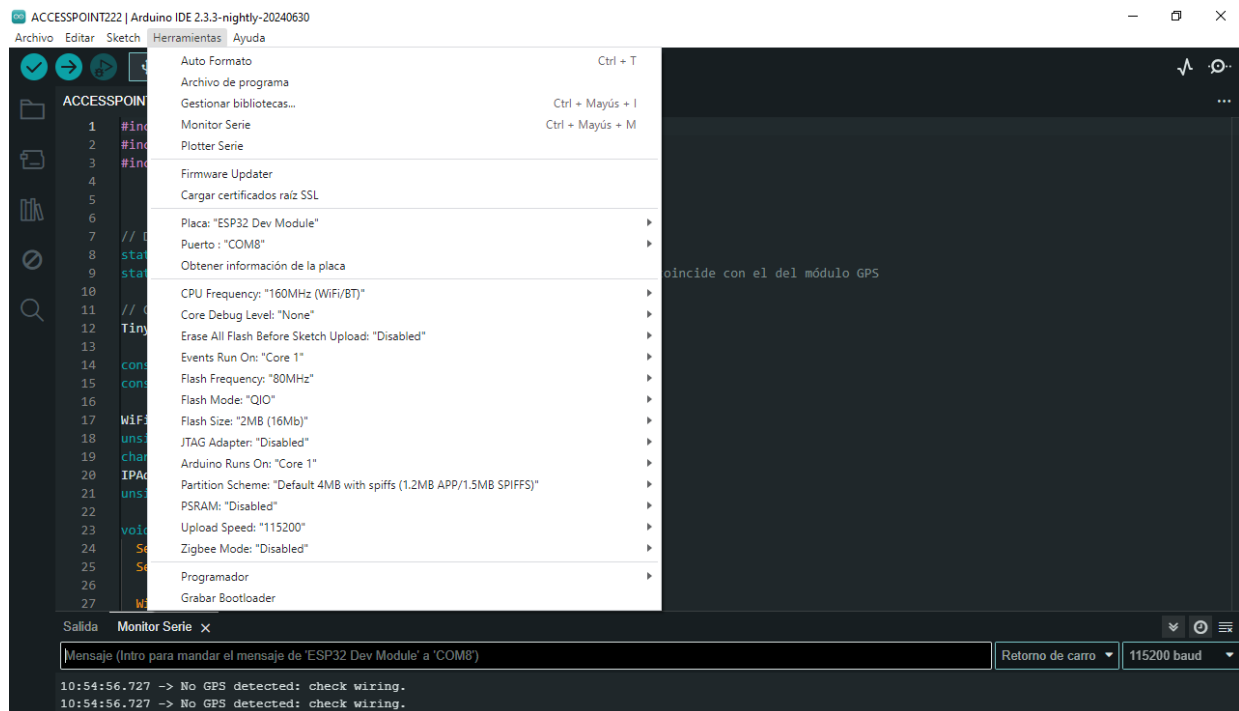


FIGURA 7.2-2 CONFIGURACIÓN DE LA TARJETA ESP32 DEL EXPANSION KIT

Para la correcta ejecución del código será necesario incluir ciertas librerías específicas. Para las funciones concretas del código relacionadas con Robomaster TT se añadirá la librería `RM TT_libs` y para las funciones relacionadas con el GPS se añadirá la librería `TinyGPSPlus` (Figura 7.2-3). Ambas disponibles en el sitio web de GitHub.

Se incluyen clickando en Skectch -> Include Library -> Add .ZIP Library.

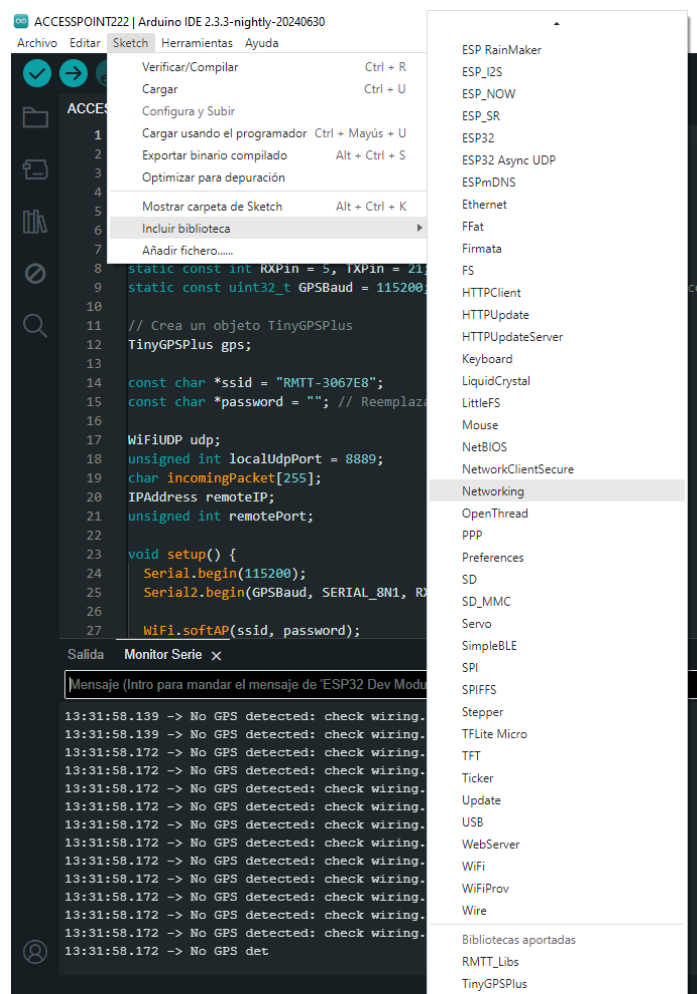


FIGURA 7.2-3 LIBRERÍAS ESPECÍFICAS INCLUIDAS EN ARDUINO IDE

2. Gestión del firmware existente y el nuevo código

La ESP32 viene de fábrica con un firmware que incluye funciones predeterminadas para el manejo y control del dron. Sin embargo, al cargar un nuevo código desde Arduino IDE, este firmware se sobrescribe y se reemplaza por el nuevo programa. Para evitar que el firmware original sea eliminado, se pueden utilizar las bibliotecas SPIFFS.h y FS.h. (Figura 7.2-4)

```
5 #include "FS.h"
6 #include "SPIFFS.h"
```

FIGURA 7.2-4 INCLUSIÓN DE LAS BILIOTECAS SPIFFS.H Y FS.H EN EL CÓDIGO

SPIFFS (SPI Flash File System) es una biblioteca que permite guardar datos en la memoria flash del ESP32. Este sistema de archivos es utilizado para almacenar

información que debe persistir incluso después de reiniciar o apagar el dispositivo, sin afectar el firmware que controla el dron. Es importante mencionar que cada vez que se carga un nuevo firmware desde el Arduino IDE, este reemplaza el firmware anterior, lo que implica que el nuevo código cargado sobrescribirá el programa previamente almacenado.

Por otro lado, la biblioteca FS.h no implementa un sistema de archivos en sí misma, sino que define una interfaz genérica que facilita la interacción con diferentes sistemas de archivos, como SPIFFS, LittleFS o FAT, el cual es comúnmente utilizado para tarjetas SD. En esencia, FS.h actúa como una capa de abstracción entre el código del usuario y el sistema de archivos subyacente, permitiendo una mayor flexibilidad y modularidad en el manejo de datos en la memoria del ESP32.

Es necesario señalar que, al trabajar con Arduino IDE, este debe tener acceso a Internet para gestionar las bibliotecas requeridas. Cuando se incluyen bibliotecas como FS.h o SPIFFS.h en el código, y estas no están instaladas localmente, Arduino IDE intentará descargarlas automáticamente desde el Arduino Library Manager o desde los repositorios correspondientes.

3. Establecimiento de la conexión Wi-Fi

En lugar de conectarse a una red Wi-Fi existente, el código configura el microcontrolador ESP32 para actuar como un Punto de Acceso (AP). Esto significa que el ESP32 crea su propia red Wi-Fi, permitiendo que otros dispositivos se conecten a él. Durante la configuración del punto de acceso, se define el SSID (nombre de la red Wi-Fi que el ESP32 emitirá) y la contraseña (Figura 7.2-5). Una vez configurado y activo, otros dispositivos pueden conectarse directamente al ESP32 a través de esta red, lo que facilita la comunicación directa y el control del dispositivo.

Aunque el firmware actual permite la conexión vía Wi-Fi, es necesario crear un punto de acceso para el envío de datos a través de sockets.

```
15  const char *ssid = "RMTT-3067E8";  
16  const char *password = "";  
17  
18  WiFiUDP udp;  
19  unsigned int localUdpPort = 8888;
```

FIGURA 7.2-5 SSID, CONTRASEÑA Y PUERTO DEL PUNTO DE ACCESO CREADO

4. Decodificación de datos GPS en NMEA

Los datos GPS recibidos en el formato NMEA son cadenas de texto que contienen información sobre la ubicación, velocidad y tiempo (FIGURA 7.1-6). El código utiliza la librería TinyGPSPlus para leer las cadenas NMEA de datos, analizar la información contenida y extraer datos relevantes como latitud, longitud, dirección y hora, convirtiéndolas en información utilizable para su procesamiento y visualización.

Para leer los datos NMEA generados por el módulo GPS, se utiliza el protocolo UART por lo que dentro del código se definirán los pines RX y TX correspondientes. Para leer la brújula, sin embargo, se utiliza el protocolo I2C, que emplea los pines SDA (datos) y SCL (reloj). A diferencia de UART, estos pines no necesitan ser declarados explícitamente en el script.

5. Transmisión de datos al ordenador

Una vez que los datos GPS han sido decodificados, estos se transmiten al ordenador a través de una conexión serial o Wi-Fi. Esto permite al ordenador recibir y procesar la información en tiempo real, y si es por conexión Wi-Fi, de forma remota. Esto permitirá conocer la posición del dron sin necesidad de estar conectado físicamente a él. (FIGURA 7.2-6)



FIGURA 7.2-6 IMAGEN GENERADA POR I.A. DE LA CONEXIÓN WIFI ENTRE DRON Y PC

El Extension Kit emite una red Wi-Fi de 5GHz que no proporciona acceso a internet; su propósito es exclusivamente establecer una conexión directa entre dispositivos. Esta red permite que el dron y otros dispositivos se comuniquen entre sí, pero no ofrece conectividad a internet.

En este caso particular, fue necesario añadir una antena Wi-Fi de 5GHz al ordenador para poder detectar y conectarse a esta red emitida por el Extension Kit (RMTT-3067E8). Aunque esta red es dual y también puede operar en 2.4 GHz, la banda de 2.4 GHz tiende a generar problemas de conexión, lo que hace preferible utilizar la banda de 5GHz. En este caso se utilizó un ordenador incompatible con redes Wi-Fi de 5GHz, sin embargo, en modelos más modernos estas redes suelen ser visibles sin problemas.

La antena Wi-Fi adicional no solo permite conectarse a la red del dron, sino que también facilita el uso simultáneo de dos conexiones Wi-Fi diferentes: una para conectarse a una red con acceso a internet y otra para conectarse al dron a través de la red 5GHz del Extension Kit. Esto asegura que el ordenador pueda mantenerse conectado a internet mientras controla el dron, permitiendo por ejemplo las consultas a APIs online que utiliza el código de Processing. En la siguiente figura se aprecia la conexión con una red Wi-Fi local (MIWIFI_UTtHP_5G) y la conexión con la red que emite la tarjeta (RMTT-3067E8). (FIGURA 7.2-7)

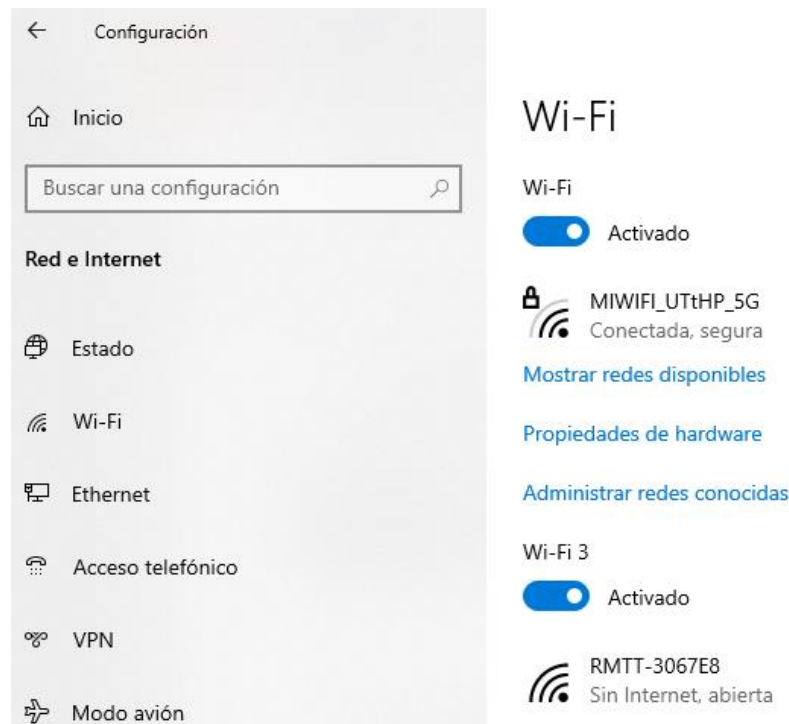


FIGURA 7.2-7 CONEXIÓN WI-FI CON RED LOCAL Y RED PROPIA DE LA TARJETA

7.3. Configuración Python – Dron Robomaster TT

Los pasos para la configuración necesaria para capacitar al entorno Python de las funciones de Robomaster, como la instalación de su SDK, se describen detalladamente en:

<https://robomaster-dev.readthedocs.io/en/latest/index.html>

7.4. Procesamiento de datos en Python

1. Recepción vía WiFi y manipulación de datos GPS

La recepción de datos vía WiFi desde el dispositivo ESP32 se maneja mediante el uso de sockets UDP. Este proceso se lleva a cabo principalmente en la función “send_command”, que es responsable tanto del envío de comandos al ESP32 como de la recepción de las respuestas. El flujo comienza con la función “send_command_thread” que inicia un nuevo hilo para ejecutar “send_command”. La separación en hilos permite que el sistema continúe funcionando sin bloquearse durante la espera de una respuesta.

Una vez que se ha enviado un comando al ESP32 a través de un socket UDP, la función “send_command” entra en un bucle donde espera recibir datos de vuelta desde el ESP32. Este bucle escucha continuamente en el socket configurado para recibir los datos, que se decodifican de su formato en UTF-8. (FIGURA 7.3-1)



FIGURA 7.4-1 ESQUEMA DEL PROCESO DE RECEPCIÓN DE DATOS EN PYTHON

2. Integración con Processing para visualización gráfica

Una vez decodificados, estos datos se envían en forma de cadena de texto a Processing para su visualización en tiempo real, también mediante el uso de sockets UDP, utilizando la función “enviar_a_processing”. (FIGURA 7.3-2)

```

DEL GPS : Location: 37.991331, -3.470086, Course: N, 354.77, Date/Time: 9/8/2024 12:27:29
DEL GPS : Location: 37.991333, -3.470088, Course: N, 349.80, Date/Time: 9/8/2024 12:27:31
DEL GPS : Location: 37.991336, -3.470089, Course: N, 352.94, Date/Time: 9/8/2024 12:27:33
DEL GPS : Location: 37.991341, -3.470089, Course: N, 354.81, Date/Time: 9/8/2024 12:27:35
DEL GPS : Location: 37.991344, -3.470090, Course: N, 354.45, Date/Time: 9/8/2024 12:27:37
DEL GPS : Location: 37.991359, -3.470091, Course: N, 355.68, Date/Time: 9/8/2024 12:27:39
  
```

FIGURA 7.4-2 CADENA DE TEXTO ENVIADA A PROCESSING DESDE PYTHON

7.5. Desarrollo de la Interfaz Gráfica en Processing

El código de Processing está dividido en cinco clases: Completo (main), CalculadorRuta, DatosGPS, ReceptorDatos y RenderizadorMapa. Y entre ellas gestionarán las siguientes tareas.

1. Integración de datos recibidos desde Python

La clase “ReceptorDatos” es la encargada de recibir los datos enviados desde el script de Python. Esta clase implementa la interfaz “Runnable”, lo que le permite ejecutarse en un hilo separado, escuchando continuamente en un puerto específico (en este caso, el puerto 12000) sin bloquear la ejecución del resto del programa.

“run()”: Este método establece un socket de Datagramas (UDP) que se mantiene a la espera de recibir datos. Cuando se recibe un paquete de datos, el contenido se procesa mediante el método “extraerDatos(String linea)”.

“extraerDatos(String linea)”: analiza la cadena recibida en el formato Location: seguido de datos como latitud, longitud, dirección y fecha. Si los datos son válidos, se extraen las coordenadas y otra información relevante. Luego, se utilizan los métodos de la clase “DatosGPS” para actualizar la posición actual y el resto de detalles.

2. Desarrollo de la interfaz gráfica

La clase “renderizadorMapa” es responsable de manejar la visualización del mapa en la aplicación, incluyendo la carga y el dibujo de la imagen del mapa (previamente descargado desde Maperitvie en la ventana, ajustándola según el tamaño en pixels de la misma. A través del método “dibujarMapa()”, la imagen del mapa se muestra en pantalla, mientras que el método “dibujarPosicionActual(float lon, float lat)” permite visualizar la posición actual del dron en el mapa, convirtiendo las coordenadas GPS en coordenadas de pantalla (en pixels) para dibujar un marcador en la ubicación correspondiente. La clase se encarga de dibujar botones e íconos adicionales que forman parte de la interfaz gráfica. Además, la clase permite dibujar trayectorias en el mapa utilizando el método “dibujarRuta(JSONArray coordenadasRuta)”, que conecta puntos representados por “coordenadasRuta” para mostrar el recorrido.

Cada vez que se actualizan las coordenadas o la ruta, la clase se encarga de redibujar el mapa y los elementos visuales para reflejar la información más reciente, garantizando que la representación en pantalla esté siempre actualizada.

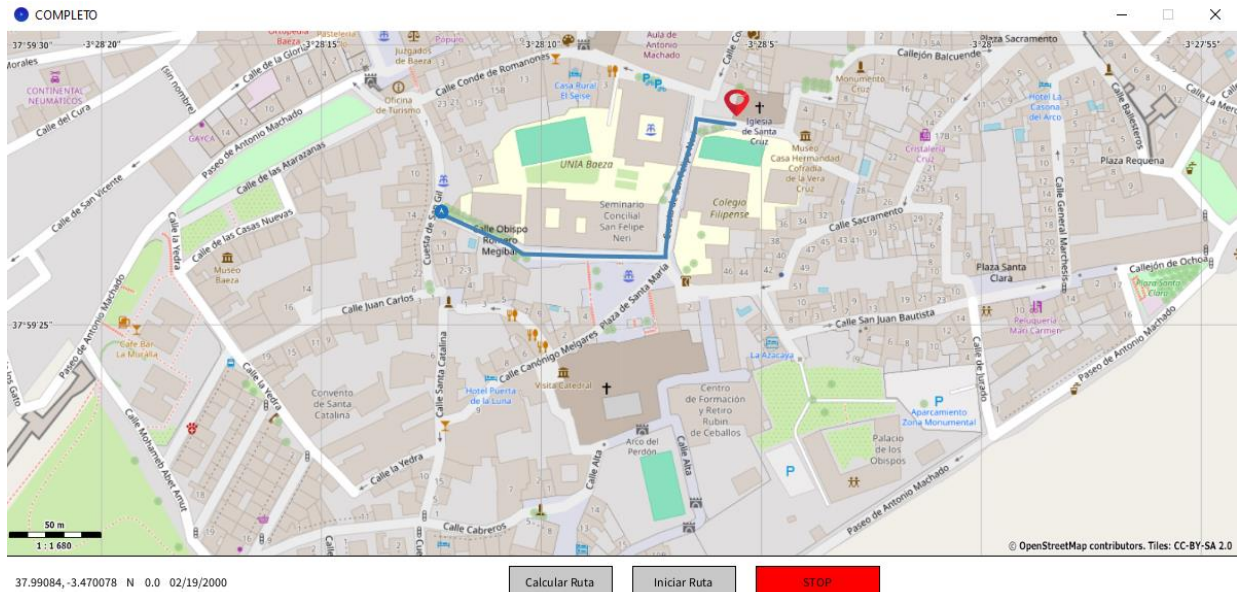


FIGURA 7.5-1 INTERFAZ GRÁFICA

3. Creación de rutas y conexión con la API de OpenStreetMap

El programa se conecta a la API de OpenStreetMap para obtener datos sobre las rutas, enviando solicitudes HTTP que contienen las coordenadas de origen y destino. Estas solicitudes, gestionadas a través de una biblioteca de red en Processing, permiten recibir la información necesaria para trazar la ruta en un formato como JSON o XML. El uso de esta API es fundamental para calcular rutas dinámicamente basadas en las entradas del usuario.

Una vez recibida la respuesta de la API, el programa procesa los datos para extraer las coordenadas de la ruta, convirtiendo esta información en una lista de puntos (waypoints) que pueden ser utilizados para dibujar la ruta en el mapa. Este proceso de parseo es esencial para transformar los datos JSON en coordenadas que el programa pueda interpretar y representar visualmente. La clase encargada del renderizado del mapa toma estos datos y los utiliza para trazar la ruta visualmente. El método “dibujarRuta(JSONArray coordenadasRuta)”, recibe un array de JSONArray, que representa los puntos de la ruta, y luego dibuja líneas entre estos puntos para representar el recorrido en el mapa. Si las coordenadas de la ruta cambian o se actualizan, el mapa se redibuja automáticamente, asegurando que el usuario siempre vea la ruta más reciente.

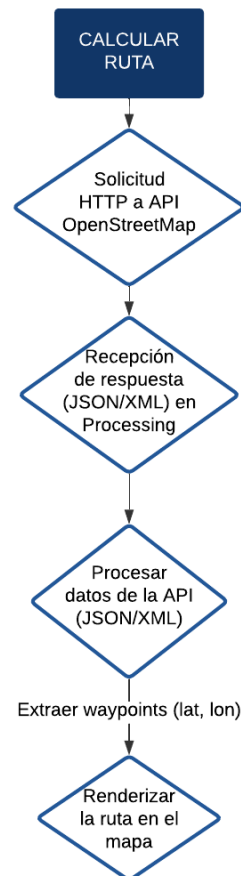


FIGURA 7.5-2 ESQUEMA DEL PROCESO DEL CÁLCULO DE RUTAS

4. Transmisión de rutas de vuelta a Python y otros envíos.

En el programa, la transmisión de rutas y otros datos de vuelta a Python se realiza mediante la creación y manejo de archivos del formato “.JSON” o “.txt”, en lugar de utilizar sockets para la comunicación directa. Esto se debe a que, al intentar usar sockets, pueden surgir problemas de solapamiento entre puertos y sockets, lo que dificulta el funcionamiento estable y confiable de la comunicación.

Los waypoints se envían uno por uno, y en Processing se comprueba si se ha alcanzado el siguiente waypoint utilizando la posición recibida a través del GPS. Solo después de verificarlo, se envía el siguiente waypoint.

Cuando el programa en Processing necesita enviar rutas u otros datos de vuelta a Python, primero escribe estos datos en un archivo en el sistema de archivos y los guarda en una ruta específica (la misma que contiene los archivos del programa de

Processing). Este archivo puede ser un simple texto o un archivo en formato JSON, dependiendo de la complejidad y la estructura de los datos, JSON para coordenadas y direcciones, txt para mensajes cortos y simples.

Un mensaje simple sería STOP, diseñado con la intención de interrumpir el vuelo del dron en situaciones de emergencia o cuando el usuario lo solicite.

```
DEL GPS : Location: 37.787745, -3.777534, Course: ENE, 65.39, Date/Time: 9/9/2024 10:31:41
DEL GPS : Location: 37.787747, -3.777529, Course: ENE, 63.99, Date/Time: 9/9/2024 10:31:43
Coordenadas iniciales guardadas en C:/Users/phonc/OneDrive/Escritorio/COMPLETO/coordenadas_iniciales.json
DEL GPS : Location: 37.787749, -3.777525, Course: ENE, 56.71, Date/Time: 9/9/2024 10:31:45
```

FIGURA 7.5-3 CREACIÓN DEL ARCHIVO COORDENADAS_INICIALES.JSON EN UNA RUTA ESPECÍFICA

7.6. Automatización de la Ruta de Vuelo del Dron en Python

1. Recepción en Python desde Processing

El script de Python, que espera recibir datos desde Processing, monitorea el sistema de archivos en busca del archivo creado. Una vez que detecta la existencia del archivo, Python lo abre y lee el contenido, procesando los datos según sea necesario.

Para mantener el sistema limpio y evitar la acumulación de archivos, Python elimina el archivo después de haber sido leído y procesado. (FIGURA 7.5-1)

```
Coordenadas iniciales guardadas en C:/Users/phonc/OneDrive/Escritorio/COMPLETO/coordenadas_iniciales.json
Coordenadas iniciales leídas desde el archivo JSON:
Latitud: 37.990840911865234
Longitud: -3.4700779914855957
Grados: 0.0
Archivo C:/Users/phonc/OneDrive/Escritorio/COMPLETO/coordenadas_iniciales.json eliminado después de leer los datos.
```

FIGURA 7.6-1 LECTURA Y ELIMINACIÓN DEL ARCHIVO COORDENADAS_INICIALES.JSON

2. Conversión de la ruta en órdenes de vuelo y su envío a Robomaster TT desde Python

Una vez leído el archivo de coordenadas iniciales, que contiene la longitud, latitud y dirección inicial del dron, se comienza el proceso. A partir de ese punto, el

programa entra en un bucle donde se lee el siguiente archivo que contiene el siguiente waypoint. Es importante destacar que el primer waypoint no se considera para el recorrido, ya que corresponde a la posición inicial del dron.

Cada vez que se recibe un nuevo waypoint, el programa procede a calcular los ángulos y distancias necesarios para guiar al dron hacia ese punto. Primero, se utiliza la función "calcular_angulo" para determinar el giro que el dron debe realizar para apuntar en la dirección del siguiente waypoint. Luego, se emplea la función "calcular_distancia" para calcular la distancia exacta que el dron debe recorrer hasta alcanzar ese waypoint.

Una vez realizadas las acciones necesarias para llegar al waypoint, el programa espera a recibir otro archivo desde Processing con el siguiente waypoint para continuar la ruta.

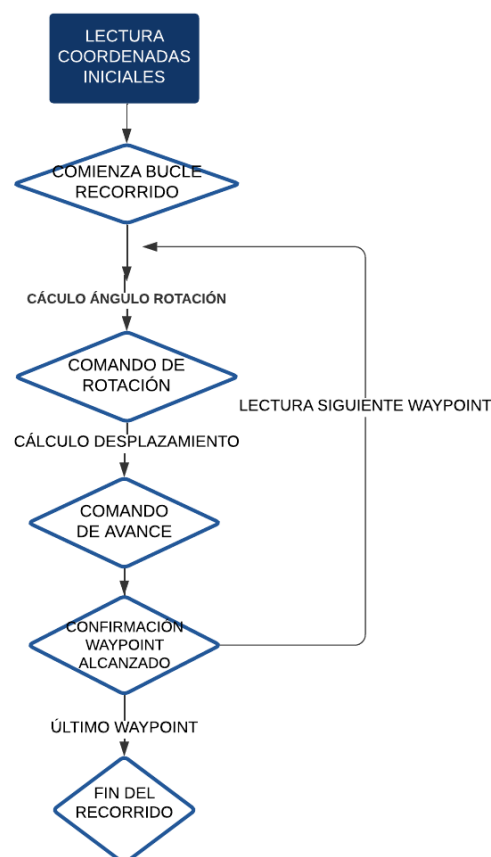


FIGURA 7.6-2 ESQUEMA DEL BUCLE DE MOVIMIENTO DEL DRON

Este proceso está diseñado para aceptar el comando STOP en cualquier momento durante el recorrido, permitiendo interrumpir la operación cuando sea necesario.

En el mismo bucle, con el próximo ángulo y la próxima distancia calculados, el programa genera las órdenes de vuelo necesarias. Para enviar estas órdenes, el código se integra con el SDK de Tello, utilizando funciones básicas como despegue, aterrizaje, avance y rotación. (FIGURA 7.6-5)

Para usar los comandos de Robomaster antes será necesario incluir las bibliotecas de Robomaster. (FIGURA 7.6-4)

```
19 from robomaster import robot
```

FIGURA 7.6-3 DECLARACIÓN DE LA BIBLIOTECA ROBOT DE ROBOMASTER

```
24 tl_drone = robot.Drone()
25 tl_drone.initialize()
26
27 tl_flight = tl_drone.flight
28
29 tl_flight.takeoff().wait_for_completed()
30
31 tl_flight.forward("distance").wait_for_completed()
32 tl_flight.rotate("angle").wait_for_completed()
33
34 tl_flight.land().wait_for_completed()
35
36 tl_drone.close()
```

FIGURA 7.6-4 COMANDOS UTILIZADOS PARA EL CONTROL DE MOVIMIENTO DEL DRON

8. Resultados y Discusión

8.1. Evaluación del rendimiento del sistema integrado

El sistema de posicionamiento del dron RoboMaster TT presentó una tendencia a desplazarse lentamente hacia adelante de forma involuntaria, lo que, aunque no crítico en recorridos pequeños, podía generar desviaciones acumuladas en recorridos largos. Aun así, el dron mostró gran precisión en giros y desplazamientos, con un cálculo efectivo de ángulos y distancias entre waypoints en procesos unitarios.

La brújula presentó fluctuaciones en la orientación al inicio de los vuelos, lo que complicaba la alineación inicial. Si se esperaba a que la brújula se estabilizara correctamente antes de comenzar, el dron podía completar el recorrido sin inconvenientes. En caso contrario, era evidente desde el principio que la trayectoria sería errática, aunque se puede corregir la dirección a tiempo usando la función de parada “STOP” y deteniendo el vuelo. El módulo GPS necesitaba tiempo para estabilizarse al inicio, y su precisión se veía afectada por nubes. Una vez en marcha, el dron seguía la ruta de manera eficiente.

La interfaz desarrollada en Processing es fácil de usar y se asemeja a otras aplicaciones de navegación conocidas, lo que facilita su manejo. Las comunicaciones entre los programas y dispositivos involucrados son seguras y eficientes, permitiendo una rápida lectura de los datos, un procesamiento veloz y el envío de órdenes casi instantáneamente al dron. En la práctica, el sistema de comprobación de waypoint alcanzado es útil para comprobar que el dron está realizando el recorrido de forma ordenada y sin percances. Sin embargo, hay un pequeño inconveniente relacionado con el uso de la API de OpenStreetMap: esta API está diseñada principalmente para calcular rutas de vehículos, por lo que genera trayectorias que siguen las normas de tráfico para coches. Como resultado, el dron evita volar en direcciones contrarias, por calles peatonales o por zonas de difícil acceso para un automóvil. Aunque este enfoque simplifica la ruta y asegura un trayecto más seguro, muchas veces termina generando caminos más largos en comparación con una ruta optimizada para un vehículo aéreo que no tiene que seguir las mismas restricciones que los coches.

8.2. Limitaciones y desafíos encontrados

La principal limitación del sistema fue la restricción de distancia operativa a 100 metros, debido a la conexión Wi-Fi con el PC, lo que impide realizar rutas largas.

También se observó que el peso adicional del módulo GPS incrementó la inestabilidad del dron, especialmente en presencia de vientos leves, que lo desviaban de su ruta programada y afectaban la precisión.

Las pruebas se realizaron en condiciones óptimas: sin viento, sin nubes, en calles amplias y planas. En estos escenarios, el sistema funcionó correctamente. Sin embargo, en condiciones menos favorables, como vientos o nubes, el rendimiento del dron y la precisión del sistema podrían verse comprometidos.

9. Conclusiones

9.1. Logros alcanzados en el proyecto

El principal logro de este proyecto fue conseguir que un dron de bajo presupuesto, como el RoboMaster TT, pudiera integrar de manera efectiva un módulo GPS, por un precio aproximado de entre 250 y 280 euros. Esto permitió al dron realizar vuelos autónomos basados en coordenadas GPS, aprovechando las ventajas de la tecnología sin necesidad de grandes inversiones. Esta implementación no solo mejoró las capacidades de navegación del dron, proporcionando mayor precisión en la ejecución de rutas, sino que también demostró que es posible integrar tecnologías avanzadas en drones económicos, abriendo nuevas posibilidades de aplicación.

Durante el desarrollo del proyecto, se alcanzaron varios objetivos secundarios que fueron importantes para integrar el GPS en el dron RoboMaster TT con éxito. Uno de los primeros logros fue encontrar el módulo GPS adecuado, el Foxeer M10Q 250, que se ajustó perfectamente a las necesidades del proyecto gracias a su tamaño compacto, su precisión y su bajo consumo de energía. Este GPS permitió mantener la estabilidad y maniobrabilidad del dron sin afectar su rendimiento.

Otro aspecto importante fue establecer una comunicación estable entre la tarjeta ESP32, el GPS y el ordenador. Este enlace permitió que el dron transmitiera y recibiera datos en tiempo real, asegurando que el control fuera preciso y fiable durante el vuelo. Gracias a esta comunicación fluida, el sistema pudo gestionar tanto las coordenadas GPS como los comandos de vuelo, facilitando la automatización de las rutas y el seguimiento de la posición del dron en todo momento.

Se desarrolló una interfaz gráfica en Processing que resultó ser intuitiva y eficaz. Esta interfaz permitió visualizar la ubicación del dron en un mapa y trazar rutas de forma precisa. Un aspecto especialmente relevante fue la implementación de una función de parada de emergencia "STOP", que añadió un nivel extra de seguridad, permitiendo detener el dron en caso de errores, garantizando un control seguro.

9.2. Contribuciones y aplicaciones futuras

Este proyecto ha sido una excelente oportunidad para adquirir conocimientos en áreas como la integración de hardware y software, la comunicación entre dispositivos y el uso de tecnologías GPS. A través del desarrollo del módulo GPS para el dron, se han fortalecido habilidades en la gestión de datos en tiempo real, la programación de algoritmos de control autónomo y el diseño de interfaces para la visualización de rutas. Estos conocimientos no solo son fundamentales para el ámbito de la robótica y la navegación, sino también aplicables a una amplia variedad de sectores tecnológicos.

Un aspecto que puede mejorarse en trabajos futuros es la implementación de un sistema de control autónomo que verifique continuamente si el dron sigue correctamente la ruta planificada. Este sistema no solo comprobaría la llegada a cada waypoint, sino que también monitorizaría el trayecto entre ellos, detectando desviaciones provocadas por condiciones externas como viento o niebla, interferencias en la señal GPS, o errores propios del dispositivo, como la inestabilidad del dron.

Además, sería interesante implementar un sistema de parada de emergencia automática que se active en situaciones críticas, como pérdida de señal, fallos graves del dron, o detección de obstáculos inesperados. Este sistema actuaría de manera inmediata para detener el dron y evitar posibles accidentes.

En conjunto, este control autónomo, con la capacidad de corregir el vuelo y detenerlo en situaciones de emergencia, sería una mejora valiosa para garantizar la seguridad del dron durante vuelos "no ideales".

Bibliografía

- BricoGeek. (2024). *Adafruit Ultimate GPS FeatherWing MTK3339*. Retrieved from https://tienda.bricogeek.com/adafruit-feather-wings/985-adafruit-ultimate-gps-featherwing.html?search_query=gps&results=16
- BricoGeek, 2. (2024). *GPS GLONASS Dual BN-880 con antena*. Retrieved from https://tienda.bricogeek.com/modulos-gps/1638-gps-glonass-dual-bn-880-con-antena.html?search_query=bn-880&results=4
- BricoGeek, 3. (2024). *SparkFun GPS - ZOE-M8Q*. Retrieved from <https://tienda.bricogeek.com/modulos-gps/1624-sparkfun-gps-zoe-m8q.html>
- DJI. (2021). *ROBOMASTER TT TELLO TALENT User Manual v1.0*.
- DJI TIENDA, M. (2024). *Robomaster TT Tello Talent*. Retrieved from <https://djiarsmadrid.com/es/robomaster-s1/2870-robomaster-tt-tello-talent.html>
- DRONELLER. (2024). *Los mejores drones*. Retrieved from <https://droneller.com/blog/best-drones-under-500/>
- DronesGator. (2024). *Los mejores drones por menos de 500 dólares en 2024*. Retrieved from <https://dronesgator.com/best-drones-under-500/>
- El-Rabbany, A. (2002). *Introduction to GPS: The Global Positioning System*. Artech House.
- ESPRESSIF. (2023). *ESP32 Technical Reference Manual Version 5.2*. Retrieved from https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf
- Hofmann-Wellenhof, B. .. (2008). *GNSS – Global Navigation Satellite Systems. GPS, GLONASS, Galileo, GPS, GLONASS, Galileo, .* Retrieved from <https://nguyenduyliemgis.wordpress.com/wp-content/uploads/2014/09/gnss-global-navigation-satellite-systems-gps-glonass-galileo-and-more-2008.pdf>
- IHA-RACE. (2024). *Foxeer M10Q 250 GPS 5883 Compass*. Retrieved from <https://iha-race.com/producto/foxeer-m10q-250-gps-5883-compass/>
- IHA-RACE, 2. (2024). *Mateksys GPS GNSS & COMPASS, M10Q-5883*. Retrieved from <https://iha-race.com/producto/mateksys-gps-gnss-compass-m10q-5883/>
- mikalhart. (2022). *TinyGPSPlus*. Retrieved from <https://github.com/mikalhart/TinyGPSPlus/tree/master>
- National Marine Electronics Association. (2002). *NMEA 0183 - Standard For Interfacing Marine Electronic Devices*. Retrieved from <https://www.plaisance-pratique.com/IMG/pdf/NMEA0183-2.pdf>
- Revision, D. R. (2020). *RoboMaster Developer Guide*. Retrieved from <https://robomaster-dev.readthedocs.io/en/latest/index.html>
- Robomaster. (2020). *RMTT_Libs*. Retrieved from https://github.com/RoboMaster/RMTT_Libs

Sparkfun, S. S. (2024). *SparkFun GPS Breakout - Chip Antenna, SAM-M10Q (Qwiic)*.

Retrieved from <https://www.sparkfun.com/products/21834>

WayfareWithPierre. (2024). *Los mejores Drones por menos de 500 dólares*. Retrieved from

<https://wayfarewithpierre.com/best-drones-under-500/>

10. Anexos

10.1. Manual de usuario de la interfaz gráfica

10.2. Robomaster TT Users Manual

10.3. Código fuente



Universidad de Jaén
Escuela Politécnica Superior de Jaén

ANEXO I

Manual de Usuario del Sistema Dron- GPS y su Interfaz Gráfica



CONTENIDO

1.	ENTORNO DE PROGRAMACIÓN	4
1.1	INSTALACIÓN DE PROCESSING	4
1.2	INSTALACIÓN DE PYTHON Y DEL SDK DE PYTHON PARA ROBOMASTER TT.....	4
2.	PUESTA EN MARCHA	5
2.1	ANTES DE COMENZAR	5
3.1	ENCIENDE EL SISTEMA DRON-GPS.....	5
4.1	COLOCACIÓN DEL DRON	5
5.1	CONEXIÓN DEL PC CON EL SISTEMA DRON-GPS	7
6.1	ASEGÚRATE DE TENER ACCESO A INTERNET	8
7.1	EJECUTAR EL PROGRAMA	8
3.	USO DE LA INTERFAZ GRÁFICA.....	9
8.1	INTERPRETACIÓN DE LA INFORMACIÓN MOSTRADA.....	10
9.1	SELECCIONAR DESTINO	12
10.1	USO DE LOS BOTONES.....	12
4.	CONSIDERACIONES ADICIONALES	14
5.	OPCIONES AVANZADAS	14
11.1	CAMBIO DE MAPA.....	14

1. Entorno de Programación

1.1 Instalación de Processing

Paso 1: Dirígete al sitio oficial de Processing y descarga la versión más reciente para tu sistema operativo.

Paso 2: Sigue las instrucciones para instalar Processing en tu equipo.

Paso 3: Descargar el .zip de "COMPLETO.zip"

1.2 Instalación de Python y del SDK de Python para Robomaster TT

Los pasos para la instalación de Python y la configuración necesaria para capacitar al entorno Python de las funciones de Robomaster, como la instalación de su SDK, se describen detalladamente en:

<https://robomaster-dev.readthedocs.io/en/latest/index.html>

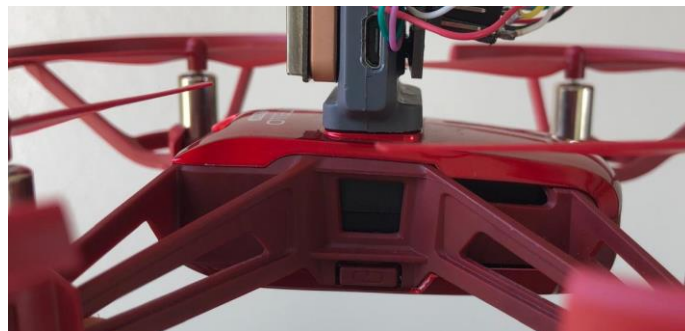
2. Puesta En Marcha

2.1 ANTES DE COMENZAR

Antes de comenzar lee cuidadosamente el Manual de Usuario de Robomaster TT, para conocer las capacidades del dron y las precauciones a tener en cuenta.

3.1 ENCIENDE EL SISTEMA DRON-GPS

Simplemente pulsa el botón de encendido de Robomaster TT.



4.1 COLOCACIÓN DEL DRON

Este paso es de vital importancia para el correcto funcionamiento del sistema.

Cuando se encienda el dron habrá que situar el GPS con la cara hacia arriba y esperar unos segundos a que la luz verde del GPS parpadee. Esto confirma el funcionamiento correcto del GPS y por tanto el buen posicionamiento del sistema.



Una vez adquirida la posición, el dron debe colocarse de manera paralela a una superficie plana y sin inclinación, o directamente sobre ella. A continuación, se realizará un pequeño desplazamiento del sistema Dron-GPS en la dirección de la cámara del dron. Tras este movimiento, es fundamental volver a situar el dron sobre una superficie plana y sin inclinación. Si el gesto se realiza en una dirección diferente, el sistema podría interpretar incorrectamente la orientación del dron.



Este gesto es crucial para asegurar el correcto funcionamiento de la orientación del dron y debe realizarse cada vez que se reinicie el sistema.

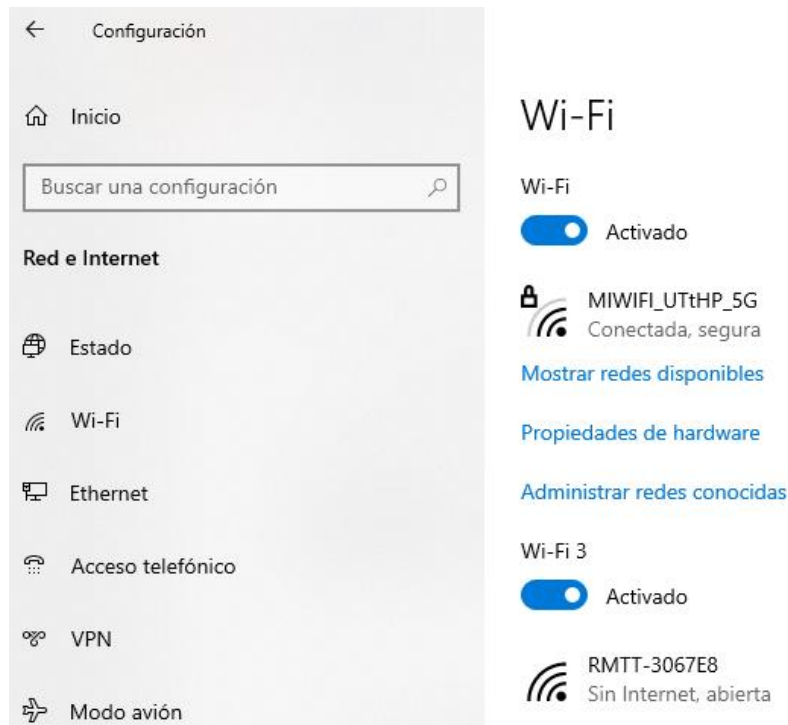
Puedes revisar más adelante en la interfaz gráfica que está orientado correctamente.

5.1 CONEXIÓN DEL PC CON EL SISTEMA DRON-GPS

Una vez encendido el sistema Dron-GPS comenzará a emitir WiFi. Busca en las redes disponibles la red RMTT-3067E8.

6.1 ASEGÚRATE DE TENER ACCESO A INTERNET

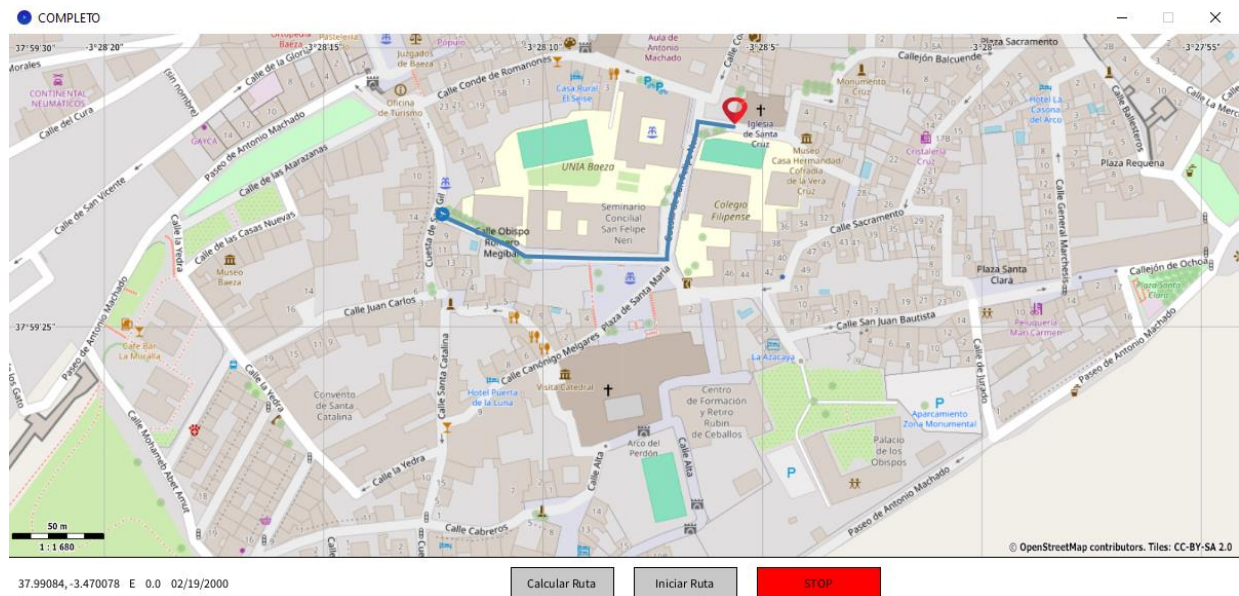
Además de la conexión WiFi con el sistema Dron-GPS, el PC necesitará una conexión a internet, ya sea por WiFi o cable, para asegurar el correcto funcionamiento en el cálculo de rutas.



7.1 EJECUTAR EL PROGRAMA

Simplemente corre el programa de Python en PyCharm y se abrirá la interfaz gráfica de Processing.

3. Uso de la Interfaz Gráfica



8.1 INTERPRETACIÓN DE LA INFORMACIÓN MOSTRADA.

UBICACIÓN

El símbolo azul en forma redonda con la silueta de una flecha en su interior indica en tiempo real la posición del dron, representa dónde se encuentra ubicado en el espacio geográfico. El símbolo también proporciona información sobre la orientación del dron, es decir, la dirección hacia la cual está apuntando o moviéndose.



INFORMACIÓN ESPECÍFICA



37.99084, -3.470078 E 0.0 02/19/2000

ESCALA DEL MAPA

Hay una barra de escala en la parte superior que indica una distancia de 50 metros. Esto ayuda a estimar las distancias en el mapa.

También se muestra la relación de escala como 1:1680, lo que significa que 1 unidad en el mapa equivale a 1680 unidades en la realidad. Esta relación permite entender la proporción del mapa.

COORDENADAS GPS

Se muestran las coordenadas geográficas del punto actual: 37.99084 (latitud) y -3.470078 (longitud). Estas son las coordenadas en formato decimal que indican una ubicación específica en el mapa.

DIRECCIÓN

La letra E indica la orientación de la dirección en que se está mirando o moviendo (en este caso, hacia el Este).

El valor 0.0 es el grado de la orientación en ese momento, lo que implica que está orientado exactamente al Este (0° en un compás).

FECHA

La fecha mostrada es 02/19/2000, aunque este es un valor por defecto.

9.1 SELECCIONAR DESTINO



Puedes usar el ratón para clicar en el mapa y seleccionar el destino, que se verá representado con el icono rojo.

10.1 USO DE LOS BOTONES.



BOTÓN: CALCULAR RUTA

Una vez seleccionado un destino, al clicar sobre “Calcular Ruta” se dibujará sobre el mapa de la interfaz, el camino más rápido calculado por el programa para que el dron llegue hasta el destino.



BOTÓN: INICIAR RUTA

Una vez se tiene la ruta calculada y dibujada se podrá pulsar sobre “Iniciar Ruta” para que el dron despegue y comience el recorrido.



BOTÓN: STOP

El botón de "STOP" interrumpe el vuelo cuando el usuario lo solicite y provoca el aterrizaje del dron.

4. Consideraciones Adicionales

El propio sistema de posicionamiento del dron Robomaster TT tiene dificultades para mantener una posición fija, ya que tiende a desplazarse lentamente hacia adelante de forma involuntaria.

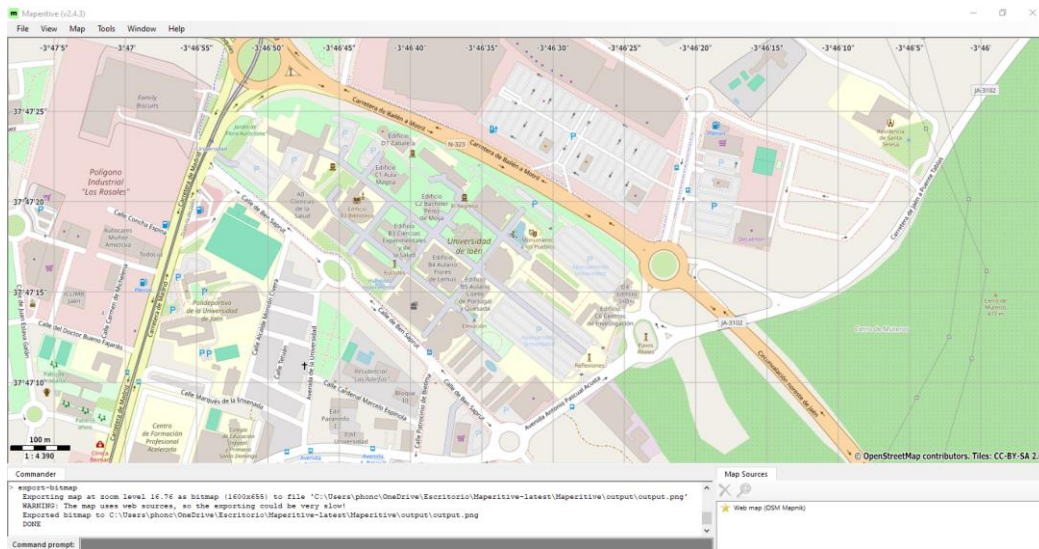
Aunque estos movimientos son pequeños, pueden acumularse y causar una desviación significativa durante el vuelo. Esto puede provocar en el peor de los casos, chocar con algún obstáculo o perder el control de la situación.

El dron no puede alejarse más de 100 metros del usuario debido a la conexión Wi-Fi con el PC, por lo que no debe realizar rutas muy largas.

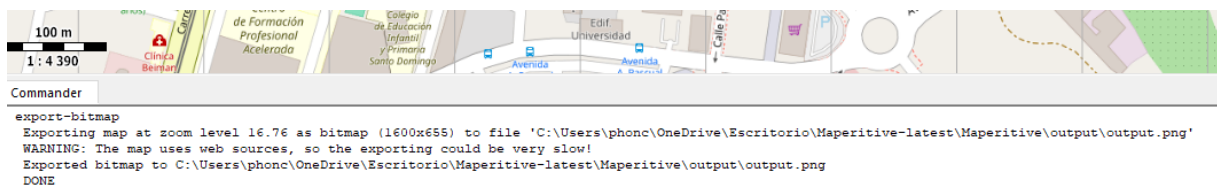
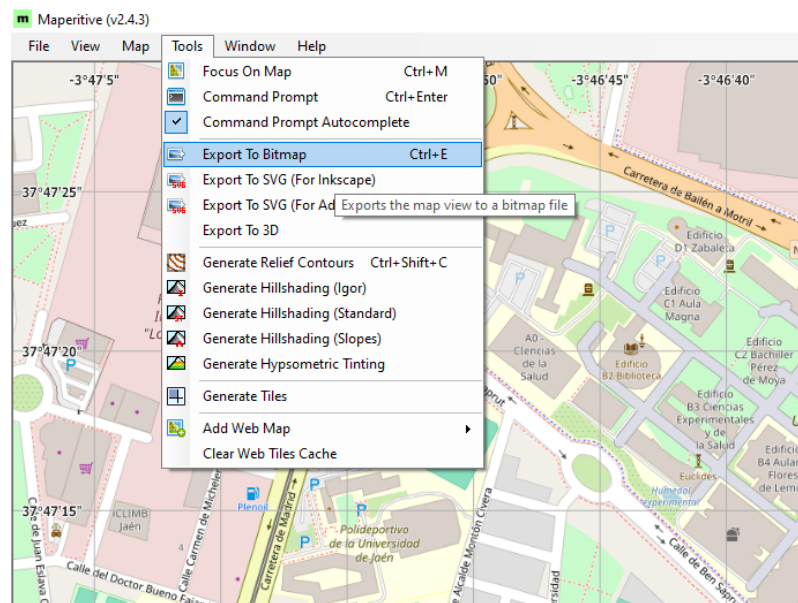
5. Opciones Avanzadas

11.1 CAMBIO DE MAPA

1. ABRIR MAPERITIVE
2. SELECCIONAR LA ZONA GEOGRÁFICA EN LA QUE DESEAS TRABAJAR Y AJUSTAR EL ZOOM.



3. DESCARGAR LA IMAGEN

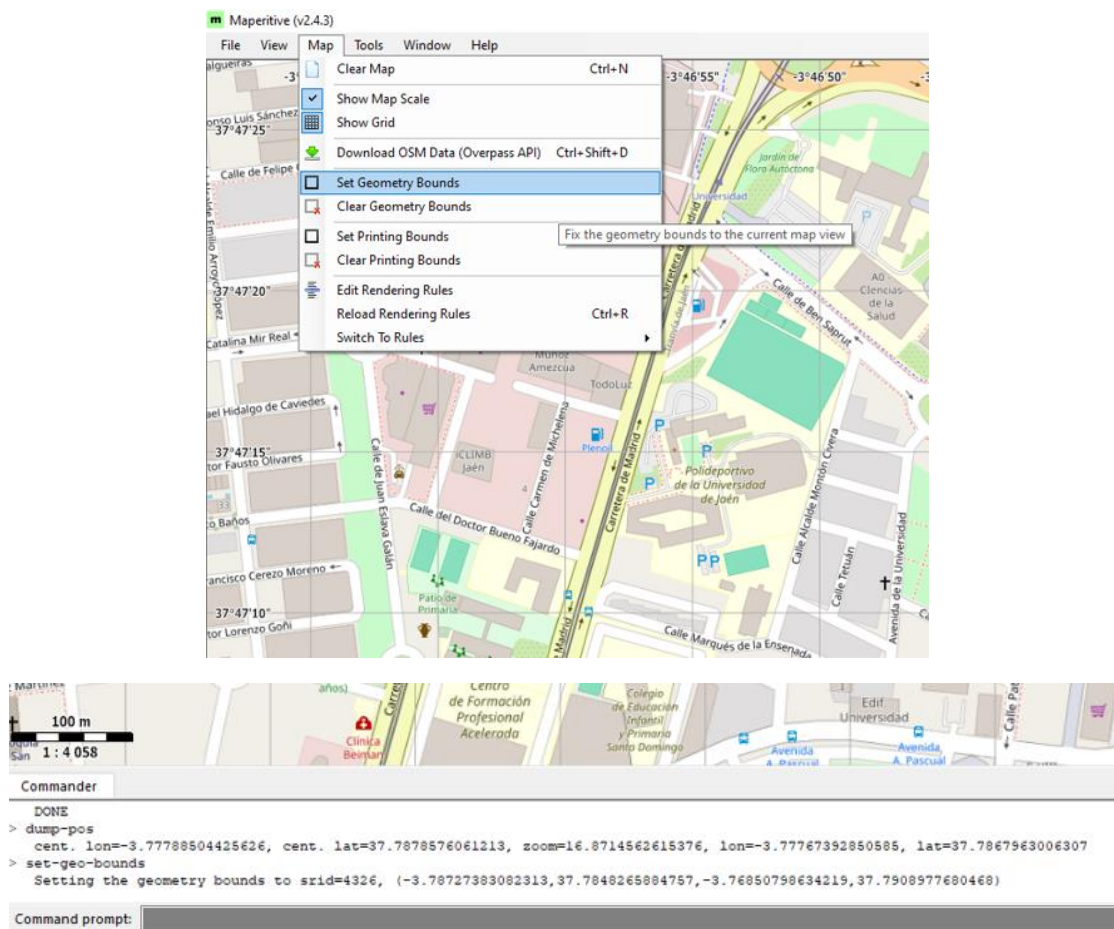


Al descargarse podrás ver la ruta en la que se ha descargado la imagen .png y su tamaño en pixeles. Guarda el valor de los pixeles para usarlo más adelante.

Cambia la ruta de la imagen "output.png" e inclúyela en la carpeta de descomprimida de "COMPLETO".

La carpeta de "COMPLETO" debe contener los siguientes archivos.

4. OBTENER LOS LÍMITES GEOGRÁFICOS DE LA IMAGEN



Guarda los valores de latitud y longitud mostrados para usarlos en el siguiente paso.

5. CAMBIAR CÓDIGO



```
16 boolean calcularRuta = false;
17 boolean STOP=true;
18 float lonInicioRuta, latInicioRuta;
19 float lonDestino, latDestino;
20
21 boolean ledRojoEncendido = false;
22 boolean ledVerdeEncendido = false;
23 boolean enviarWay = false;
24
25
26     int anchoImagen = 1600;
27     int alturaImagen = 655;
28     float factorEscala = 0.75;
29
30     //-3.78727383082313,37.7848265884757,-3.76850798634219,37.7908977680468
31
32     float latitudSupIzq = 37.7908977680468;
33     float longitudSupIzq = -3.78727383082313 ;
34     float latitudInfDer = 37.7848265884757;
35     float longitudInfDer = -3.76850798634219;
36
```

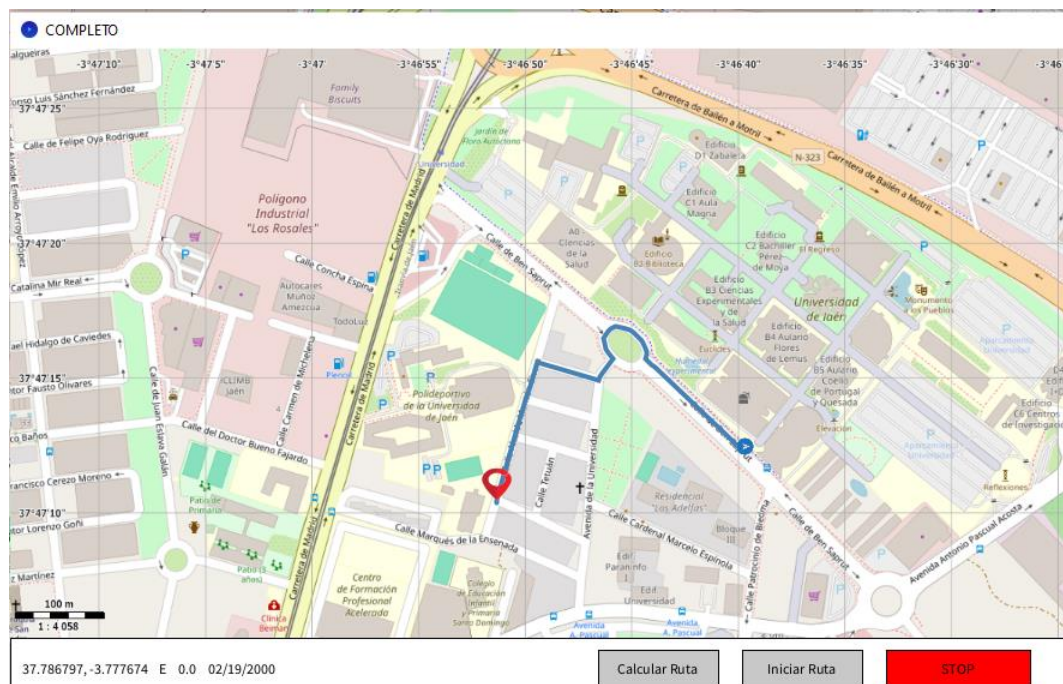
Abre el archivo de “COMPLETO.pde” y modifica los siguientes valores del código:

Cambia el tamaño en pixeles de la imagen con “ancholmagen” y “alturalmagen”.

Cambia los valores de longitud y latitud correspondientes por los obtenidos en la función de “Set Geometry Bounds”

6. LISTO

Ya puedes utilizar tu nuevo mapa en la interfaz gráfica.

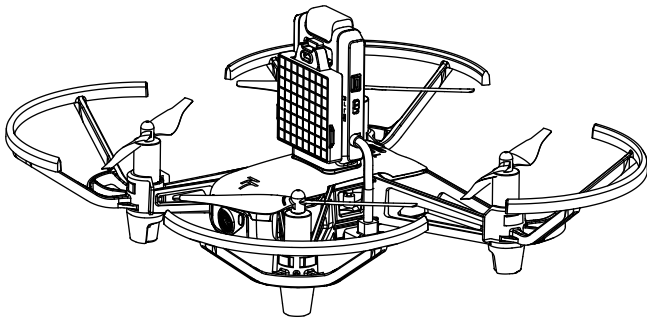


ROBOMASTER TT

TELLO TALENT

User Manual v1.0

2021.03



Searching for Keywords

Search for keywords such as “battery” and “install” to find a topic. If you are using Adobe Acrobat Reader to read this document, press Ctrl+F on Windows or Command+F on Mac to begin a search.

Navigating to a Topic

View a complete list of topics in the table of contents. Click on a topic to navigate to that section.

Printing this Document

This document supports high resolution printing.

Using This Manual

Legend

 Warning

 Important

 Hints and Tips

 Reference

Read Before the First Flight

Read the following documents before using the RoboMaster TT:

1. *RoboMaster TT User Manual*
2. *RoboMaster TT Quick Start Guide*
3. *RoboMaster TT Disclaimer and Safety Guidelines*

We recommend that you watch all tutorial videos on the official DJI website <https://www.dji.com/robomaster-tt> and read the Disclaimer and Safety Guidelines before you fly. Prepare for your first flight by reviewing the Quick Start Guide and refer to this User Manual for more details.

Download Software

1. Download the Tello app for activation, flying the aircraft, recording footage, and updating firmware. The iOS version of the Tello app is compatible with iOS v9.0 and later. The Android version of the Tello app is compatible with Android v4.4 and later.



2. Download the Tello EDU app for programming. The iOS version of Tello EDU app is compatible with iOS v10.0 and later. The Android version of Tello EDU app is compatible with Android v4.4 and later.



3. Download and install the DJI Education Hub at <http://edu.dji.com/download> and update the open-source controller firmware to the latest version using RoboMaster Assistant.

Contents

Using This Manual	2
Legend	2
Read Before the First Flight	2
Download the Tello App	2
Units of Measurement	2
Product Profile	4
Introduction	4
Aircraft Diagram	4
Aircraft	5
Flight Modes	5
Aircraft Status Indicator	6
Vision Positioning System	7
Intelligent Flight Modes	8
Propellers	12
Propeller Guards	13
Flight Battery	14
Camera	15
Tello App	16
Connecting to the Aircraft	16
Camera View	16
Controlling the Aircraft	17
Flight	20
Flight Environment Requirements	20
Compliance with Regulations	20
Test Flight	20
Firmware Updates	21
Specifications	21
After-Sales Information	21

Product Profile

Introduction

RoboMaster TT includes the aircraft and the expansion kit. The aircraft is a small quadcopter that features a Vision Positioning System and an onboard camera. Using the Vision Positioning System and advanced flight controller, the aircraft can hover in place and fly indoors. Advanced features like 8D Flips, EZ Shots, and Bounce mode make it even more fun to use the aircraft. The camera captures 5MP photos and streams 720p live video to the Tello app on a mobile device.

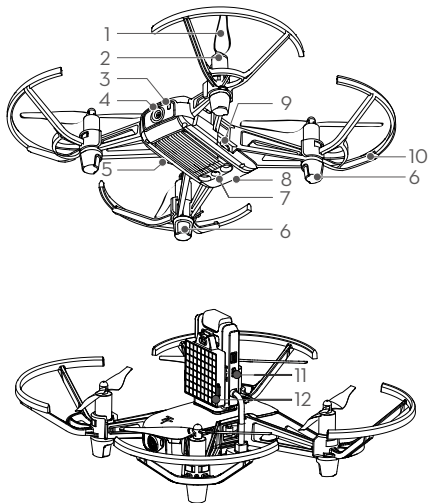
Experience DIY flight with an expansion kit that includes an open-source controller, dot-matrix display & distance-sensing module, and extension board, which can be programmed using Arduino, Scratch, MicroPython, and more. Use the Tello EDU app for programming when using a mobile device.

The max flight time of the aircraft is approx. 8 minutes with an expansion kit mounted and 13 minutes without.

Failsafe protection enables RoboMaster TT to land safely while the propeller guards can be used for enhanced safety.

* Maximum flight time was tested in windless conditions flying at a consistent 9 mph (15 kph). This value should be taken for reference only.

Aircraft Diagram



- 1. Propellers
- 2. Motors
- 3. Aircraft Status Indicator
- 4. Camera
- 5. Power Button
- 6. Antennas
- 7. Vision Positioning System
- 8. Flight Battery
- 9. Micro USB Port
- 10. Propeller Guards
- 11. Open-Source Controller
- 12. Dot-Matrix Display & Distance Sensing Module

Aircraft

The RoboMaster TT contains a flight controller, video downlink system, Vision Positioning System, propulsion system, and a Flight Battery. Refer to the aircraft diagram in the Product Profile section.

Flight Modes

The RoboMaster TT has two flight speeds that you can select when flying the aircraft manually:

- Slow (default): The maximum flight attitude angle is 9° and the maximum flight speed is 6.7 mph (10.8 kph).
- Fast: The maximum flight attitude angle is 25° and the maximum flight speed is 17.8 mph (28.8 kph).

When flying the RoboMaster TT manually the aircraft utilizes its Vision Positioning System to automatically stabilize itself. If the conditions are such that the Vision Positioning System is unavailable the aircraft automatically changes to Attitude mode.

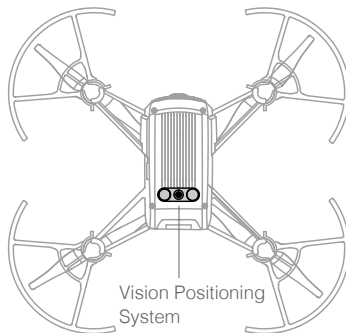
The aircraft automatically changes to Attitude mode (ATTI mode) when the Vision Positioning System is unavailable. In Attitude mode the aircraft is not able to position itself and so is easily affected by its surroundings. Environmental factors such as wind can result in horizontal shifting, which may present hazards, especially when flying in confined spaces. When the aircraft enters Attitude mode, land in a safe place as soon as possible to avoid hazards.



- To switch from Slow to Fast you must read and agree to the disclaimers and warnings that appear in the Tello app. Make sure you understand the differences between the two flight speeds.
- Failsafe Protection automatically initiates landing if the mobile device's signal is weak or is lost for 50 seconds, or the Tello app crashes.

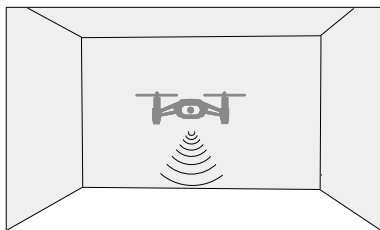
Vision Positioning System

The Vision Positioning System helps the aircraft maintain its current position. With the help of the Vision Positioning System, the aircraft can hover in place more precisely and fly indoors or outdoors in windless conditions. The main components of the Vision Positioning System are a camera and a 3D infrared module located on the underside of the aircraft.



Using the Vision Positioning System

The Vision Positioning System is activated automatically when the aircraft is turned on. No further action is required. The Vision Positioning System is only effective when the aircraft is at altitudes of 1.0 to 98.4 ft (0.3 to 30 m) and works best at altitudes of 1.0 to 19.7 ft (0.3 to 6 m). If the aircraft is beyond this range, the Vision Positioning function may be affected, so extra caution is required.



- The performance of the Vision Positioning System is affected by the surface being flown over. The aircraft automatically changes to Attitude mode when the Vision Positioning System is unavailable. In Attitude mode the aircraft is not able to position itself. Operate the aircraft with great caution in the following situations, which may cause the aircraft to enter Attitude mode:
 - a. Flying at high speed below 2 ft (0.5 m).
 - b. Flying over monochrome surfaces (e.g. pure black, pure white, pure red, pure green).
 - c. Flying over highly reflective surfaces.
 - d. Flying over water or transparent surfaces.
 - e. Flying over moving surfaces or objects.
 - f. Flying in an area where the lighting changes frequently or drastically.
 - g. Flying over extremely dark (< 300 lux) or bright (> 100,000 lux) surfaces or towards bright sources of light (e.g. towards sunlight).
 - h. Flying over surfaces without clear patterns or texture.
 - i. Flying over surfaces with identical repeating patterns or textures (e.g. tiling).
 - j. Flying over small and fine objects (e.g. tree branches or power lines).
 - k. Flying at high speed of over 5 m/s at 4 ft (1 m).
-



- If the aircraft's Vision Positioning system fails for 3 seconds when its flying altitude is above 20 ft (6 m), Failsafe Protection will automatically initiate landing. If the Vision Positioning system recovers during landing, Failsafe Protection will shut off and the aircraft will hover.
- The Vision Positioning System may not be able to recognize patterns on the ground in very dark (< 300 lux) environments. DO NOT take off if there is a warning prompt in the Tello app telling you that the environment is too dark.
- Keep the cameras and sensors clean at all times. Dirt or other debris may adversely affect their effectiveness.

Intelligent Flight Modes

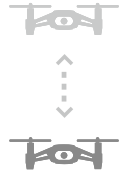
The RoboMaster TT features Bounce mode, 8D Flips, Throw & Go, Up & Away, and EZ Shots. To use an intelligent flight mode, ensure that the aircraft's battery level is at least 50%, tap in the Tello app, then select a mode.

Bounce Mode

In Bounce mode the aircraft automatically flies up and down between 1.6 and 3.9 ft (0.5 and 1.2 m) above a flat surface. If the aircraft detects an object below it (such as your hand) it increases its altitude then continues flying up and down.

Using Bounce Mode

1. Press the power button once to turn the aircraft on. Launch the Tello app and tap  to take off.
2. Tap  and then select Bounce mode. Read the information prompt and then select **Start**. The aircraft will start flying up and down.
3. Extend your arm and place your palm 1 ft (30 cm) or more below the aircraft, keeping your palm open. The aircraft will increase its altitude then continue flying up and down.
4. Tap  in the Tello app any time to exit Bounce mode.



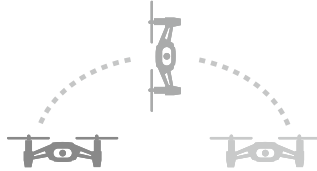
- Ensure there is sufficient space when using Bounce mode. Allow a radius of at least 7 ft (2 m) horizontally around the aircraft and allow at least 10 ft (3 m) above the aircraft.
- Before using Bounce mode ensure that the Aircraft Status Indicator is periodically blinking green twice, indicating that the Vision Positioning System is available.
- In Bounce mode, ensure your arm is extended and your palm is flat. DO NOT try to catch the aircraft. The distance between your palm and the aircraft should be at least 1 ft (30 cm).
- Watch out for potential obstacles in your surrounding area (especially on the rear, left, and right sides of the aircraft) and stay clear of them to avoid accidents.
- Be prepared to take control of the aircraft by tapping  in the Tello app to exit Bounce mode in case of an emergency.
- Be extra cautious when flying in dark (< 300 lux) or bright (> 10,000 lux) environments.

8D Flips

In 8D Flips the aircraft automatically flips in one of eight different directions.

Using 8D Flips

1. Press the power button once to turn the aircraft on. Launch the Tello app and tap  to take off.
2. Tap  and then select 8D Flips. Read the information prompt and then select **Start**.



3. Swipe within the box shown in the app. The aircraft will flip in the direction you swipe.
4. Tap  in the Tello app any time to exit 8D Flips.



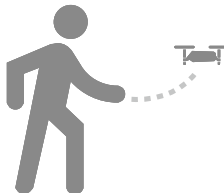
- Ensure there is sufficient space when using 8D Flips. Allow a radius of at least 7 ft (2 m) horizontally around the aircraft and allow at least 10 ft (3 m) above the aircraft.
- Before using 8D Flips ensure that the Aircraft Status Indicator is periodically blinking green twice, indicating that the Vision Positioning System is available.
- When using 8D Flips, the distance between you and the aircraft should be at least 3.5 ft (1 m).
- Watch out for potential obstacles in your surrounding area (especially on the rear, left, and right sides of the aircraft) and stay clear of them to avoid accidents.
- Be prepared to take control of the aircraft by tapping  in the Tello app to exit 8D Flips in case of an emergency.
- Be extra cautious when flying in dark (< 300 lux) or bright (> 10,000 lux) environments.
- Exercise caution when using 8D Flips with the expansion kit mounted.

Throw & Go

Throw & Go lets you launch the aircraft by gently throwing it into the air.

Using Throw & Go

1. Press the power button once to turn the aircraft on.
2. Tap  and then select Throw & Go.



3. Place the aircraft in your palm.

4. Read the warning prompt and then tap  to start. The propellers will start spinning slowly. Gently throw the aircraft upward and horizontally away from you, keeping the aircraft horizontal as you throw it. The propellers will start spinning more quickly and the aircraft will automatically hover in place. The propellers will stop if you do not throw the aircraft within 5 seconds of the propellers starting to spin slowly.



- Only use Throw & Go in an open area and make sure that your flight path is clear of people, animals, and obstacles.
- Throw & Go cannot be used after the aircraft has taken off.
- Be careful when using Throw & Go and make sure you keep your fingers away from the propellers even when they are spinning slowly.
- Hold the aircraft horizontally, and gently throw the aircraft upward and horizontally away from you. DO NOT throw the aircraft at a speed that exceeds 6 ft/s (2 m/s). DO NOT throw the aircraft at an angle to the horizontal greater than 20° and DO NOT flip the aircraft as you throw it.
- Before using Throw & Go ensure that the Aircraft Status Indicator is periodically blinking green twice, indicating that the Vision Positioning System is available.
- Watch out for potential obstacles in your surrounding area (especially on the rear, left, and right sides of the aircraft) and stay clear of them to avoid accidents.
- Be extra cautious when flying in dark (< 300 lux) or bright (> 10,000 lux) environments.

EZ Shots

Using 360

In 360, the aircraft records a short video while rotating 360 degrees.

1. Press the power button once to turn the aircraft on. Launch the Tello app and tap  to take off.
2. Tap  and then select 360. Read the information prompt and then select **Start**.



3. The aircraft will rotate 360 degrees and record a video automatically. Tap  to access the video.
4. The aircraft will exit 360 once it has finished recording. You can also tap  in the Tello app any time to exit 360.



- Ensure there is sufficient space when using 360. Allow at least 2 ft (0.5 m) around the aircraft in all directions.
- Before using 360 ensure that the Aircraft Status Indicator is periodically blinking green twice, indicating that the Vision Positioning System is available.
- Watch out for potential obstacles in your surrounding area (especially on the rear, left, and right sides of the aircraft) and stay clear of them to avoid accidents.
- Be prepared to take control of the aircraft by tapping  in the Tello app to exit in case of an emergency.
- Be extra cautious when flying in dark (< 300 lux) or bright (> 10,000 lux) environments.

Using Circle

In Circle the aircraft records a short video while flying in a circle.

1. Press the power button once to turn the aircraft on. Launch the Tello app and tap  to take off.
2. Tap  and then select Circle. Read the information prompt and then select **Start**.



3. The aircraft will fly in a circle around a point approximately 7 ft (2 m) in front of the aircraft nose and record a video.
4. The aircraft will exit Circle once it has finished recording. You can also tap  in the Tello app any time to exit Circle.



- Ensure there is sufficient space when using Circle. Allow a radius of at least 10 ft (3 m) around the point 7 ft (2 m) in front of the aircraft nose, and at least 10 ft (3 m) above and below the aircraft.
- Before using Circle ensure that the Aircraft Status Indicator is periodically blinking green twice, indicating that the Vision Positioning System is available.
- Watch out for potential obstacles in your surrounding area (especially on the rear, left, and right sides of the aircraft) and stay clear of them to avoid accidents.
- Be prepared to take control of the aircraft by tapping  in the Tello app to exit Circle in case of an emergency.
- Be extra cautious when flying in dark (< 300 lux) or bright (> 10,000 lux) environments.

Using Up & Away

In Up & Away the aircraft records a short video while flying upward and backward.

1. Press the power button once to turn the aircraft on. Launch the Tello app and tap  to take off.
2. Tap  and then select Up & Away. Read the information prompt and then select **Start**.



3. The aircraft will record a short video while flying upward and backward.
4. The aircraft will exit Up & Away once it has finished recording. You can also tap  in the Tello app any time to exit Up & Away.



- Ensure there is sufficient space when using Up & Away. Allow at least 20 ft (6 m) behind and 3.5 ft (1 m) above the aircraft.
- Before using Up & Away ensure that the Aircraft Status Indicator is periodically blinking green twice, indicating that the Vision Positioning System is available.
- Watch out for potential obstacles in your surrounding area (especially on the rear, left, and right sides of the aircraft) and stay clear of them to avoid accidents.
- Be prepared to take control of the aircraft by tapping (X) in the Tello app to exit Up & Away in case of an emergency.
- Be extra cautious when flying in dark (< 300 lux) or bright (> 10,000 lux) environments.

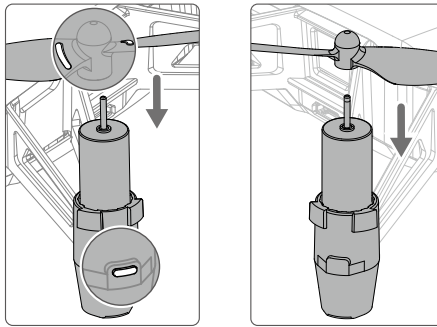
Propellers

The RoboMaster TT uses model 3044P propellers. There are two varieties of 3044P propellers, which are designed to spin in different directions. The presence or absence of marks on the propellers indicates which type they are and therefore which motors they should be attached to.

Attaching the Propellers

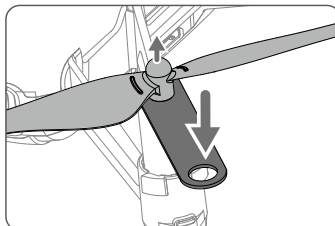
Mount marked propellers on the motors with marked landing gear. Mount unmarked propellers on the motors with unmarked landing gear.

When mounting, ensure that the gap between the bottom of the propeller cap and the motor is no bigger than needed to insert the propeller removal tool.



Detaching the Propellers

Insert the propeller removal tool between the propeller cap and the motor. Be sure to hold the motor while detaching the propeller.





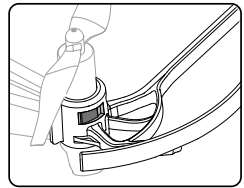
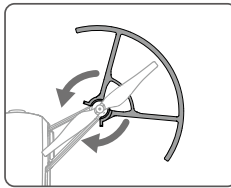
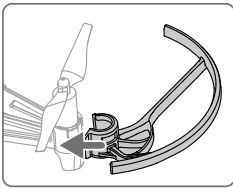
- Always detach the propellers using the propeller removal tool. DO NOT remove the propellers by hand as doing so may damage the motors and you may get seriously hurt.
- To avoid injury, stand clear of and DO NOT touch propellers or motors when they are spinning.
- Only use original propellers and DO NOT mix propeller types.
- Ensure that the propellers and motors are installed firmly and correctly before each flight.
- Ensure that all propellers are in good condition before each flight. DO NOT use aged, chipped, or broken propellers.

Propeller Guards

The Tello Propeller Guards can be used to reduce the risk of harm or damage to people or objects resulting from accidental collisions with Tello aircraft.

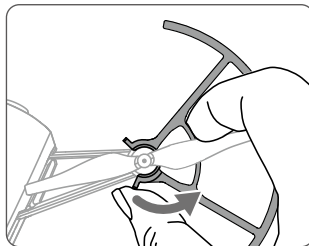
Mounting the Propeller Guards

Mount each of the propeller guards to the aircraft landing gear beneath the motors. Push each propeller guard inwards to wrap it around the landing gear. Make sure it clicks into position and that the protruding parts of the landing gear securely fit into the notches on the propeller guards.



Detaching the Propeller Guards

To remove a propeller guard place your finger and thumb as shown in the figure below. With your thumb, gently apply a twisting force to the lip that protrudes from the propeller guard where it wraps around the landing gear.



DO NOT use excessive force when removing propeller guards as doing so may damage the aircraft's arms and you may get hurt.

Flight Battery

The Flight Battery is a 3.8 V, 1100 mAh battery with charging/discharging protection.

 Fully charge the Flight Battery before each flight.

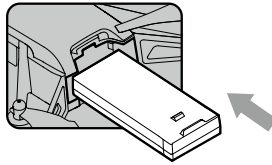
Battery Features

1. Overcurrent/Overvoltage Protection: The battery stops charging if an excessive current/voltage is detected.
2. Overdischarge Protection: Discharging stops automatically to prevent excessive discharge.
3. Short Circuit Protection: The power supply is cut automatically if a short circuit is detected.

 Refer to the *Tello Disclaimer and Safety Guidelines* before use. Users take full responsibility for all operations and usage.

Inserting the Flight Battery

Insert the Flight Battery into the aircraft as shown. Ensure the battery is mounted firmly.

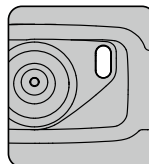
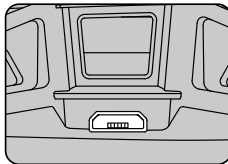


To remove the battery pull it out from the aircraft.

Charging the Flight Battery

To charge the Flight Battery connect the Micro USB port on the aircraft to a USB adapter (not provided) using a standard Micro USB cable.

Charging Time: Approx. 1 hour and 30 minutes.



The Aircraft Status Indicator blinks blue slowly during charging. The battery is fully charged when the Aircraft Status Indicator turns solid blue. Detach the USB adapter when the battery is fully charged.



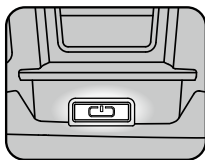
- Always use an FCC/CE (depending on location) certified USB adapter that is rated at 5 V and 1.5 A or above.
- Ensure the aircraft is powered off before charging. It cannot be charged when it is powered on.
- DO NOT charge a Flight Battery immediately after flight, because its temperature may be too high. DO NOT charge a Flight Battery until it cools down to near room temperature.
- Charge the Flight Battery in the temperature range 41° to 113° F (5° to 45° C). The ideal charging temperature range is 72° to 82° F (22° to 28° C).



Before carrying the Flight Battery on an airline flight, it must be discharged to 30% or lower. To discharge the Flight Battery, fly the aircraft.

Checking Battery Level

Press the power button once to turn the aircraft on. Launch the Tello app and check the battery level in the app.



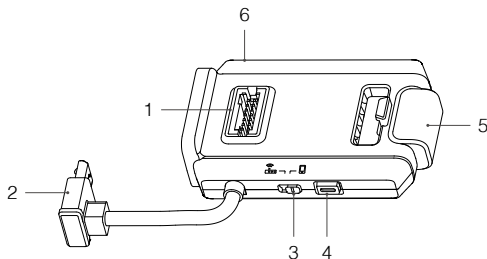
Camera

The Tello camera captures 5 megapixel photos and 720p videos. The Tello's Electronic Image Stabilization feature enables it to consistently capture clear images. The photos and videos can be viewed in the Tello app and copied to a folder in your mobile device.

Expansion Kit

Open-Source Controller

The open-source controller combines a 2.4/5 GHz dual frequency Wi-Fi module, Bluetooth module, and Arduino open-source platform. It is able to expand the features of pins such as UART, I2C, GPIO, PWM, and SPI. Experience DIY flight by customizing accessories through programs such as Arduino and MicroPython.



1. I/O Expansion Ports

Used to connect to the dot-matrix display & distance sensing module to add and expand features. Supports UART, I2C, PWM, and SPI.

IO2	IO5	IO4	IO27	IO26	GND	3V3
IO12	IO21	IO15	IO14	IO13	GND	5V

The actual output voltage of 5 V is $4.8\text{ V} \pm 0.2\text{ V}$ and of 3 V is $3.3\text{ V} \pm 0.1\text{ V}$. The maximum output current is 800 mA. Each I/O expansion port can be mapped in the program and connected to the user's expansion module. For example, the UART pin can be connected to a module that identifies QR codes so that the RoboMaster TT can be programmed to identify QR codes.

2. Micro USB Cable

Used to connect the open-source controller to a 5V/2A USB power source or the Micro USB port of the aircraft for power supply.

When connected to the Micro USB port of the aircraft, the open-source controller can be used as an expansion module. The Wi-Fi network will change to RMTT-XXXXXX when the open-source controller connects to the aircraft.

3. Switch

Used to switch between direction connection mode and router mode. Direct connection mode is where the aircraft is connected to a mobile device via Wi-Fi. Router mode is where the aircraft connected to a router.

Direct Connection Mode: Connect the mobile device to the Wi-Fi network of the aircraft named RMTT-XXXXXX.

Router Mode: Connect the aircraft to the router using the SDK. The app cannot be used to connect to the aircraft in this mode.

Toggle the switch to restart the open-source controller in router mode.

Users can test programming easily by toggling the switch back and forth to reboot the open-source controller. Note that the connection mode will depend on the final position of the switch after toggling.

4. Customizable Button

Press and hold the button to link via Bluetooth (when using the default firmware).
Press the button twice within 0.5 s to start the motors to cool down the aircraft. Press the button twice within 0.5 s again to stop the motors (when using the default firmware).
The function of the button can be customized through programming. Functions include using the button to launch programs.

5. Programmable RGB Light

The function of the RGB light is set through programming. Functions include being used as part of a light display or as a prompt.
The status of the RGB light are shown in the table below (when using the default firmware).

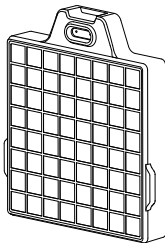
RGB Light Blinking Pattern	Descriptions
After powering on, the light will change color in sequence and stop	Working normally
Blinks blue	Ready for Bluetooth connection. Press and hold the customizable to link
Solid blue	Bluetooth remote controller connected

6. Micro USB Port

Used to connect to a computer. The port can be used as a debugging port for offline programs when using Arduino or MicroPython.
Used to connect to a computer to update the firmware of the open-source controller.

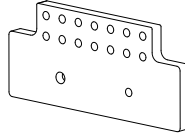
* The default firmware is required to connect to a Bluetooth remote controller and to use the motors to cool down the aircraft. RM Assistant can be used to restore the default firmware of the open-source controller.

Dot-Matrix Display & Distance Sensing Module



The module integrates a 8x8 dot-matrix display and a distance sensing TOF module, allowing users to generate different colors and graphics through programming. The default firmware is required to use Mind+ real-time mode or the Tello EDU app to control the dot-matrix display and RGB light.

Extension Board



The extension board consists of 14-pin extension port to 2x7 pin, 2.54mm dual in-line package, two reserved positions for 5/3.3V power indicators, and two reserved positions for test indicators. Users can add more sensors to expand features.

SDK Mode

Enter SDK Mode

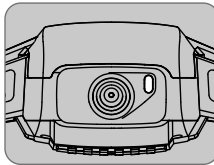
1. Make sure the aircraft and the Tello EDU app are connected.
2. Make sure the aircraft and Mind+ real-time mode are connected.
3. Make sure the default firmware of the open-source controller is being used with the Bluetooth remote controller.
4. Send the "command" command in plaintext through UDP and wait for the device to reply "ok".
5. Send the "[TELLO]" command in plaintext through the expansion kit and wait for the aircraft to reply "ok".

Exit SDK Mode

Power off the aircraft to exit SDK mode.

Status LED Descriptions

The descriptions below apply to the aircraft with or without the open-source controller mounted.



Status	Blinking Pattern	Descriptions
Normal	Blinks red, green, and yellow continuously	Powered on and performing self-diagnostic tests
	Blinks green twice repeatedly	Positioning with Vision Positioning System
	Blinks yellow slowly	No positioning with Vision Position System

Charging	Solid blue	Charging complete
	Blinks blue slowly	Charging
	Blinks blue quickly	Abnormal status detected
SDK Mode Status	Blinks purple slowly	SDK connected
	Blinks purple quickly	SDK disconnected (no commands receive for more than 15 seconds)
Warning States	Blinks yellow quickly	Remote controller signal lost (when not in SDK mode)
	Blinks purple quickly	Remote controller signal lost
	Blinks red slowly	Low battery
	Blinks red quickly	Critically low battery
	Solid red	Critical error

Tello App

Use this app to control the camera and other aircraft functions. The app is used for configuring the aircraft, viewing photos and videos, and copying photos and videos to folders on a mobile device. In addition, the Tello app can be used for activation and firmware update.

Connect to the Aircraft

Enable Wi-Fi on the mobile device and select the network. The network is TELLO-XXXXXX when there is no expansion kit mounted and RMTT-XXXXXX when there is an expansion kit mounted. The live view will display on the screen of the mobile device once connection is complete.

Camera View



1. Auto Takeoff/Landing

Tap  to initiate auto takeoff. Tap  to initiate auto landing.

There are two auto landing modes: Tap to Land and PalmLand (Hand Landing). In Tap to Land the aircraft lands automatically. To use PalmLand place your palm under the aircraft, then tap to confirm and the aircraft will land on your palm and stop its motors.



- Do not use auto takeoff from your palm, and ensure to use auto takeoff on a flat surface.
- Only land the aircraft on flat surfaces. DO NOT land the aircraft over water, grass or sand. When using PalmLand, ensure your palm is right under the aircraft and keep your palm flat.

2. Intelligent Flight Modes

Tap  to select Intelligent Flight Modes.

3. Settings

Tap  to enter the settings screen. You can adjust flight speed, VR settings, Bluetooth joystick settings, and Wi-Fi settings here.

By default the Tello does not have a Wi-Fi password. You can set a password and you can also change the Wi-Fi SSID. (To reset the Wi-Fi SSID and password to the default settings, power on the aircraft and then press and hold the power button for 5 seconds. The Tello will restart automatically.)

In the More screen the beginner guide, units of measurement, photo quality, low battery warning, and joystick settings can be configured. Tap  to calibrate the IMU or center of gravity, or to view the aircraft firmware version.

4. Battery Level

 Displays the current battery level.

5. Wi-Fi Status

 Displays the Wi-Fi connection status.

6. Bluetooth Status

 Displays the Bluetooth connection status.

7. Flight Speed

 Displays the aircraft's horizontal speed.

8. Flight Altitude

 Displays the altitude above the surface below the aircraft.

9. Playback

Tap  to enter the playback page and preview photos and videos as soon as they are captured.

10. Photo/Video Toggle

Tap  to switch between photo and video recording modes.

11. Shoot / Record Button

Tap / to start shooting photos or recording video.

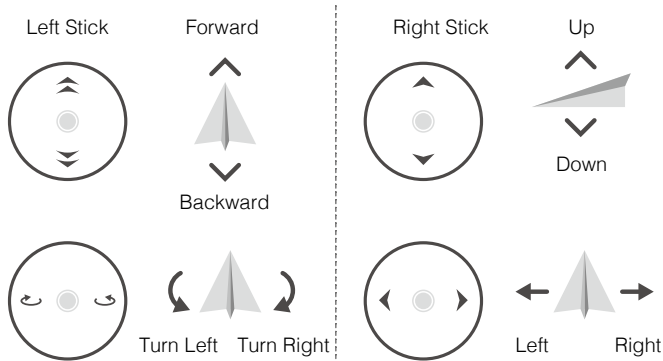
12. Virtual Joysticks

Use virtual joysticks to control the aircraft. Two modes (Mode 1 and Mode 2) are available. The default mode is Mode 2.

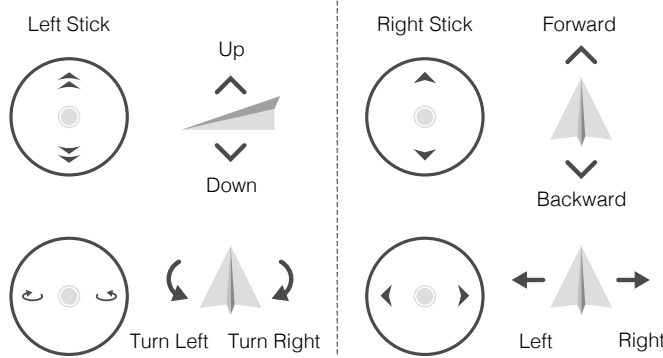
The virtual joysticks are used to control the aircraft's orientation (yaw), forward/ backward movement (pitch), altitude (throttle), and left/right movement (roll). The function that each virtual joystick movement performs is determined by the choice of virtual joystick mode. Two modes (Mode 1 and Mode 2) are available. The default mode is Mode 2.

In each of the two modes the Tello hovers in place at a constant orientation when both virtual joysticks are centered. Pushing a virtual joystick away from the center position performs the functions show in the figure below.

Mode 1



Mode 2



The figure below explains how to use each virtual joystick, using Mode 2 as an example.

Virtual Joysticks (Mode 2)	Remarks
Left Stick 	Moving the left stick up or down changes the aircraft's altitude. Push the stick up to ascend and down to descend. The more the stick is pushed away from the center position, the faster the aircraft will change altitude. Always push the stick gently to prevent sudden and unexpected changes in altitude.

Left Stick 	Moving the left stick to the left or right controls the orientation of the aircraft. Push the stick left to rotate the aircraft counter-clockwise and right to rotate the aircraft clockwise. The more the stick is pushed away from the center position, the faster the aircraft will rotate.
Right Stick 	Moving the right stick up and down changes the aircraft's pitch. Push the stick up to fly forward and down to fly backward. The more the stick is pushed away from the center position, the faster the aircraft will move.
Right Stick 	Moving the right stick to the left or right changes the aircraft's roll. Push the stick left to fly left and right to fly right. The more the stick is pushed away from the center position, the faster the aircraft will move.



- The area beyond the white circles is also responsive to control commands.
- Tello is currently compatible with the customized Gamesir T1D. The virtual joysticks are disabled when connected to a remote controller via Bluetooth.

Program Platform

Besides the Tello EDU app, RoboMaster TT supports several computer programming software. Click the links below to view manual documents and learn more about how to program.

Tello SDK 3.0

Mind+

Arduino

Firmware Updates

Updates Aircraft

When you connect the aircraft to the Tello app you will be notified if a new firmware update is available. To start updating, connect your mobile device to the Internet and follow the on-screen instructions.



- Firmware updates take around 5 minutes.
- Before performing an update ensure the Flight Battery has at least 50% charge.

Updates Expansion Kit

Update the open-source controller firmware to the latest version using RoboMaster Assistant.

How to update firmware:

- a. Download and install the DJI Education Hub at <http://edu.dji.com/download>.
- b. Launch RoboMaster Assistant.
- c. Connect the open-source controller to the computer using a micro USB cable. Follow the instructions to update firmware.

Flight

Ensure that all flights are carried out indoors or in an open area with no wind. The flight altitude is limited to 98.4 ft (30 m) and flight distance is limited to 328 ft (100 m). Perform a simple test flight the first time you fly the aircraft. Refer to the Test Flight section below.

Flight Environment Requirements

1. DO NOT use the aircraft in adverse weather conditions such as rain, snow, fog, wind, smog, hail, lightning, tornadoes, or hurricanes.
2. Only fly at locations where you can keep the aircraft at least 33 ft (10 m) away from obstacles, people, animals, buildings, public infrastructure, trees, and bodies of water when in flight.
3. DO NOT fly the aircraft on a route that has an abrupt change in the ground level (such as from inside a building to outside), otherwise the positioning function may be disrupted, impacting flight safety.
4. Aircraft and battery performance is subject to environmental factors such as air density and temperature. Be very careful when flying 3,281 ft (1,000 m) or more above sea level, since battery and aircraft performance may be reduced.

- 5. DO NOT use the aircraft near accidents, fire, explosions, floods, tsunamis, avalanches, landslides, earthquakes, dust, or sandstorms.
- 6. To avoid interference between your smart device and other wireless equipment, turn off other wireless equipment while you are flying the aircraft.
- 7. DO NOT fly in areas where magnetic or radio interference may occur such as close to: Wi-Fi hotspots, routers, Bluetooth devices, high voltage lines, high voltage power transmission stations, mobile base stations, or broadcasting towers. Flying in areas where interference may disrupt communication between the aircraft and the remote control device may adversely affect flight orientation and location accuracy and may potentially lead to loss of control. Interference may also lead to video downlink errors.

Compliance with Regulations

To avoid serious injury and property damage, observe local laws and regulations during flight. Refer to the *Tello Disclaimer and Safety Guidelines* for details.

Test Flight

Perform a simple test flight the first time you fly the aircraft:

- 1. Place the aircraft in a flat area with the Flight Battery facing towards you.
- 2. Turn on the aircraft.
- 3. Connect to the Wi-Fi named TELLO-xxxxxx or RMTT-xxxxxx, and then Launch the Tello app and enter the camera view.
- 4. Use auto takeoff.
- 5. Use the virtual joysticks to control the aircraft.
- 6. Use auto landing.
- 7. Turn off the aircraft.

Specifications

Tello (Model: TLW004)	
Weight (Propeller Guards Included)	87 g
Max Speed	17.8 mph (28.8 kph)
Max Flight Time	13 minutes (0 wind at a consistent 9 mph (15 kph))
Operating Temperature	32° to 104° F (0° to 40° C)
Operating Frequency	2.4 to 2.4835 GHz
Transmitter (EIRP)	<20 dBm (FCC)
	<19 dBm (CE)
	<19 dBm (SRRC)
Camera	
Max Image Size	2592×1936
Video Recording Modes	HD: 1280×720 30p
Video Format	MP4
Flight Battery	
Capacity	1100 mAh
Voltage	3.8 V

Battery Type	LiPo
Energy	4.18 Wh
Net Weight	25±2 g
Charging Temperature Range	41° to 113° F (5° to 45° C)
Max Charging Power	10 W
Expansion Kit	
Open-Source Controller	
Model	RMTTOC
Operating Mode	Direct Connection Mode, Router Mode
Wi-Fi	2.4 GHz , 5.8 GHz
Transmitter (EIRP)	2.4 GHz: <18.5dBm(FCC/SRRC/MIC); <17 dBm(CE) 5.8 GHz: <15 dBm(FCC/SRRC); <13 dBm(CE)
Bluetooth	2.4 GHz
MCU	ESP32-D2WD, Dual-Core Main Frequency: 160 MHz, Calculation ability: 400 MIPS
Open-Source	Supports SDK, Arduino, Scratch and MicroPython
Expansion	14-pin expansion port (I2C, UART, SPI, GPIO, PWM, power source)
LED	Full Color LED
Dot-Matrix Display & Distance Sensing Module	
Dot-Matrix LED	Red and blue LED 8x8
Dot-Matrix Driver function	IIC data port, auto dot-matrix scan, adjustable 256 global brightness, adjustable single-pixel red and blue LED 256 brightness
Distance Sensing Module	TOF
Maximum Sensing Distance of TOF	1.2 m (indoors with white wall)
Extension Board	
DIY Connection	14-pin extension port to 2x7 pin, 2.54mm dual in-line package, two reserved positions for 5V/3.3V power indicators, two reserved positions for test indicators

After-Sales Information

Visit <https://www.dji.com/robomaster-tt> to learn more about after-sales service policies, repair services and support.



This content is subject to change.

Download the latest version from
<https://www.dji.com/robomaster-tt>

Copyright © 2021 Ryze Tech. All Rights Reserved.



Universidad de Jaén
Escuela Politécnica Superior de Jaén

ANEXO III

El código fuente empleado durante el desarrollo del proyecto queda recogido en el siguiente repositorio:

<https://github.com/asgarcia/navegador-Dron>