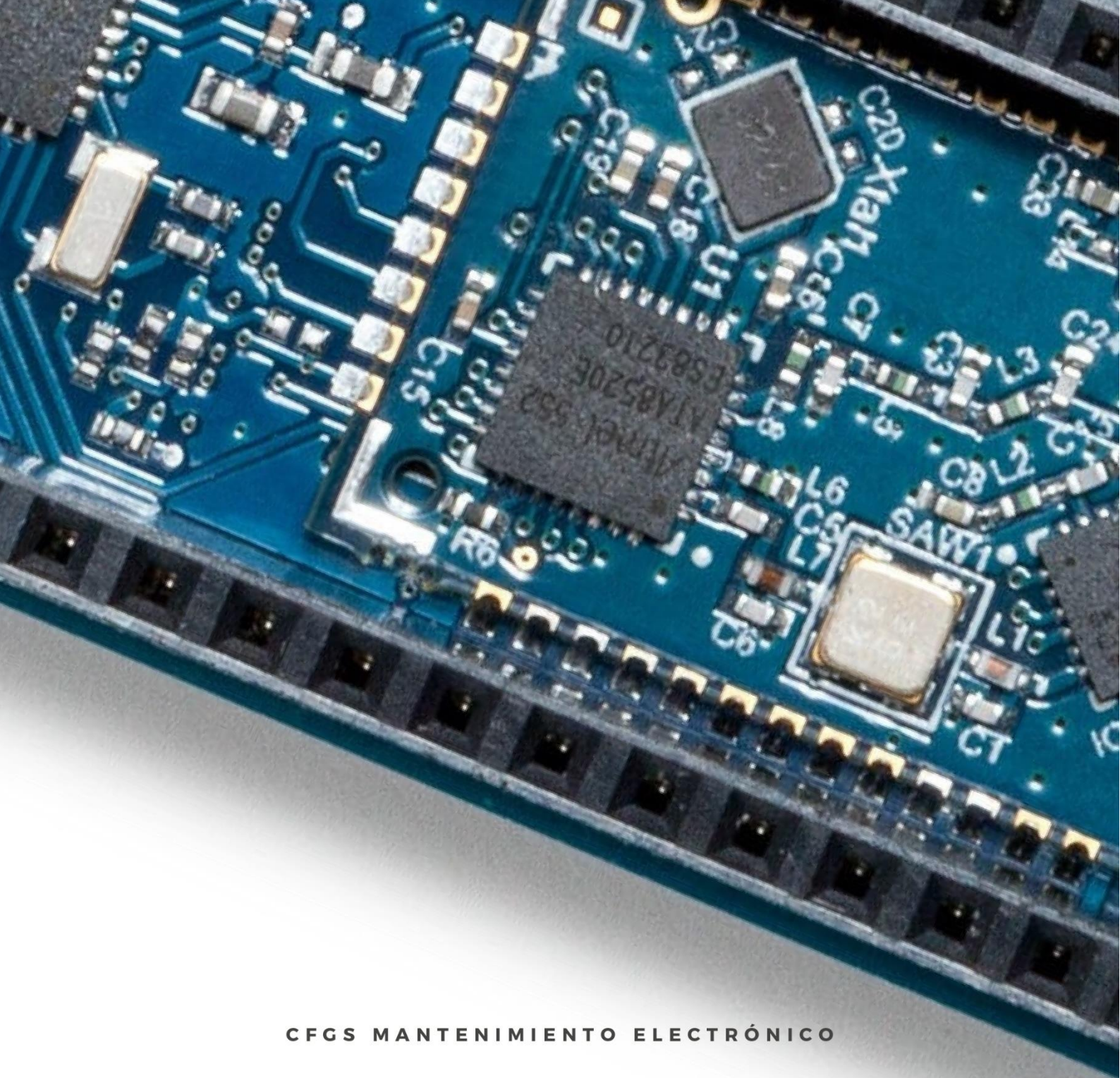




CFGS MANTENIMIENTO ELECTRÓNICO

METSAT-1

DESARROLLO DE UNA SONDA ESTRATOSFÉRICA

ADRIÁN PEÑALVER FERNÁNDEZ

AUTOR: Adrián Peñalver Fernández.

Proyecto fin de ciclo, junio de 2020

Mantenimiento electrónico.



Escanea los códigos QR, para acceder a contenido adicional e interactivo.

Para ello, instale en su teléfono un escaner QR. Una vez descargada la aplicación, solo debe enfocar con la cámara el código y automáticamente se accederá al contenido.



La mayoría de las imágenes poseen enlaces para ampliar aún más la información al lector.

Para acceder al contenido solo pulsa sobre la imagen y espera a que se abra la página web, para ello debe tener acceso a Internet.

Algunos de los enlaces permiten la descarga de programas utilizados, el código de Arduino, datasheets, etc...



Este proyecto ha sido *crowdfundado*, muchas gracias a todos los mecenas que hicieron posible este proyecto.

Agradecimientos.

Agradezco a mi tutor de proyecto Germán Villalba por ofrecerme la oportunidad de realizar este trabajo que tanto tiempo llevaba queriendo hacer.

Agradezco a los miembros de URE-Cartagena y de InZpace por su asesoramiento y su tiempo a lo largo de este proyecto.

Agradezco a todas aquellas personas y empresas que nos han ayudado a poder realizar este proyecto a través de “Crowdfunding”.

Agradezco al Instituto Politécnico de Cartagena por su ayuda económica y por facilitarnos la realización del trabajo y ciertos trámites.

Agradezco a los profesores y compañeros que han colaborado y ayudando en el lanzamiento y la planificación.

Y, por último, se lo agradezco a mi familia y a mi pareja por su apoyo incondicional, ya que sin ellos nada de esto hubiese sido posible.

Muchas gracias a todos.

A mi padre, allá donde estés.

ÍNDICE DE CONTENIDO

ÍNDICE DE TABLAS.....	V
NOMENCLATURA.....	VI
RESUMEN	VIII
ABSTRACT.....	VIII
SOMMARIO	VIII
1. JUSTIFICACIÓN.....	1
2. DESCRIPCIÓN DEL TRABAJO	2
3. OBJETIVOS	3
4. AVIÓNICA.....	4
4.1 SUBSISTEMA DE COMUNICACIONES	5
4.1.1 SISTEMAS UTILIZADOS	5
4.1.1.1 SIGFOX	6
4.1.1.2 GPRS / GSM.....	34
4.1.1.3 WSPR.....	48
4.1.2 SISTEMAS OPCIONALES.....	66
4.1.2.1 LORAWAN	67
4.1.2.2 IRIDIUM.....	69
4.1.2.3 PI IN THE SKY	70
4.1.2.4 WSPR-TX PICO TRANSMITTER	71
4.2 SUBSISTEMA DE COMPUTADORA.....	72
4.2.1 AUTÓMATA.....	72
4.2.1.1 I2C.....	76
4.2.1.2 ONE-WIRE	77
4.2.1.3 SPI	78
4.2.1.4 UART	79
4.2.2 SENSORES Y ACTUADORES.....	80
4.2.2.1 SENSORES.....	80
4.2.2.2 ACTUADORES	82
4.2.3 WIRING	84
4.2.4 PROGRAMACIÓN	88
4.2.5 PROCESAMIENTO DE DATOS.....	102
4.3 SUBSISTEMA DE VIDEO	106
4.4 SUBSISTEMA DE POTENCIA.....	109
4.4.1 BATERÍAS.....	109
4.4.2 FUENTES DE ALIMENTACIÓN	112
4.5 ESTACIÓN DE CONTROL TERRENA	124
4.5.1 FABRICACIÓN DE ANTENAS.....	124
4.5.2 SDR.....	128
4.5.3 WSPR RX	131
5. PAYLOAD BOX.....	133
5.1 ESTRUCTURA EXTERNA.....	134
5.2 ESTRUCTURA INTERNA	142
5.3 CONDUCCIÓN DEL CALOR.....	150
6. PROPULSIÓN Y REENTRADA.....	161
6.1 SUBSISTEMA DE PROPULSIÓN	163
6.1.1 CALCULO DE HELIO	165

6.1.2 UNIÓN Sonda/GLOBO	170
6.2 SUBSISTEMA DE REENTRADA	172
6.2.1 UNIÓN Sonda/PARACAIDAS	173
7. PREPARACIÓN DEL LANZAMIENTO.....	177
7.1 SIMULACIÓN DE VUELO.....	178
7.1.1 ELECCION LUGAR DE ATERRIZAJE	178
7.1.2 CUSF LANDING PREDICTOR.....	179
7.2 METEOROLOGÍA	188
7.2.1 ATMÓSFERA TERRESTRE	188
7.2.2 WINDY.....	190
7.3 PERMISOS Y SEGURO	193
7.3.1 SEGURO DE AVIACIÓN	194
7.3.2 PERMISO ENAIRE	196
7.3.3 PERMISO HELIO.....	198
7.4 LOGÍSTICA.....	198
7.4.1 LISTAS DE COMPROBACION.....	199
7.4.2 COMUNICACIÓN	200
7.4.3 TRANSPORTE.....	201
7.4.4 PLANNING	203
7.5 FINANCIACIÓN.....	204
7.5.1 PUBLICIDAD	205
7.5.1.1 REDES SOCIALES.....	206
7.5.1.2 MEDIOS DE COMUNICACIÓN	207
7.5.1.3 BOCA A BOCA	210
7.5.2 CROWDFUNDING.....	210
8. LANZAMIENTO	217
8.1 PROCESO DE LLENADO DEL GLOBO	218
8.2 PREPARACIÓN DE LA Sonda	221
8.3 PROCESO DE LANZAMIENTO	222
8.4 SEGUIMIENTO Y RECUPERACIÓN	223
9. RESULTADOS Y CONCLUSIONES	225
9.1 RESULTADOS	226
9.1.1 RESULTADOS SUBSISTEMA DE VIDEO	227
9.1.2 RESULTADOS SUBSISTEMA COMUNICACIONES.....	229
9.1.3 RESULTADOS SUBSISTEMA DE COMPUTADORA.....	241
9.1.4 RESULTADOS SUBSISTEMA DE POTENCIA.....	243
9.1.5 RESULTADOS SUBSISTEMA DE CALEFACCIÓN	244
9.1.6 RESULTADOS PAYLOAD BOX	245
9.1.7 RESULTADOS SUBSISTEMA PROPULSIÓN Y PARACAÍDAS	246
9.1.8 RESULTADOS SIMULACIÓN	246
9.2 CONCLUSIONES	246
9.2.1 CONCLUSIÓN SUBSISTEMA DE VIDEO.....	247
9.2.2 CONCLUSIÓN SUBSISTEMA COMUNICACIONES.....	247
9.2.3 CONCLUSIÓN SUBSISTEMA DE COMPUTADORA.....	250
9.2.4 CONCLUSIÓN SUBSISTEMA PROPULSIÓN Y PARACAÍDAS.....	252
9.2.5 RESULTADO DE LOS OBJETIVOS	253
9.3 POSIBLES MEJORAS	255
9.3.1 MEJORAS DE SUBSISTEMAS.....	255
9.3.1.1 SISTEMA AUTOMÁTICO DE REENTRADA	255
9.3.1.2 INTERFAZ SATELITARIA EN SUBSISTEMA DE COMUNICACIONES.....	256
9.3.1.3 PAYLOAD BOX 3D PRINTED	257

9.3.1.4 AUTÓMATA INDUSTRIAL	257
9.3.2 FUTURAS MISIONES	258
9.3.2.1 METSAT-2.....	258
9.3.2.2 METSAT-3.....	264
9.3.2.3 METPSL-1	264
9.3.2.4 METGLIDER-1	266
10. DESARROLLO DEL PROYECTO (GANTT)	268
11. PRESUPUESTO	273
12.CONDICIONES TÉCNICAS	275
13. BIBLIOGRAFÍA	280
ÍNDICE DE FIGURAS.....	283
ANEXOS	289
ANEXO A: AUTORIZACIÓN ENAIRE	289
ANEXO B: LISTAS DE COMPROBACIÓN	293
ANEXO C: MECENAS Y PATROCINADORES	316
ANEXO D: SEGURO AVIACIÓN	322

ÍNDICE DE TABLAS

TABLA 1 UNIDADES	VI
TABLA 2 SIGLAS	VI
TABLA 3 OBJETIVOS ESPERADOS	3
TABLA 4 CONEXIÓN MKRFOX 1200 - GPS.....	17
TABLA 5 COMANDOS OPCIONALES	38
TABLA 6 COMPARACIÓN ARDUINO UNO Y MEGA.....	73
TABLA 7 CONEXIÓN MICROSD SHIELD	84
TABLA 8 CONEXIÓN BMP180	84
TABLA 9 CONEXIÓN MPU6050	84
TABLA 10 CONEXIÓN GPS	85
TABLA 11 CONEXIÓN DHT11.....	85
TABLA 12 CONEXIÓN DS18B20	85
TABLA 13 CONEXIÓN DUAL-LED	85
TABLA 14 CONEXIÓN INTERRUPTOR	86
TABLA 15 CONEXIÓN BUZZER	86
TABLA 16 FUNCIONAMIENTO DE SISTEMAS.....	226
TABLA 17 OBJETIVOS	253
TABLA 18 PRESUPUESTO	273
TABLA 19 PRESUPUESTO BAJO COSTE	274
TABLA 20 CONDICIONES TÉCNICAS.....	275

NOMENCLATURA

Tabla 1 Unidades

UNIDADES	
Voltios	(V)
Amperios	(A)
Energía	(Wh)
Potencia	(W)
Frecuencia	(Hz)
Distancia	(m)
Temperatura	(°C)
Tiempo	(s)
Masa	(g)
Capacidad eléctrica	(F)
Inductancia eléctrica	(H)
Unidad de intensidad	(dB)
Unidades de señal por segundo	(Bd)
byte	(B)
Volumen	(m ³)
Área	(m ²)
Velocidad	(m/s)
Aceleración	(m/s ²)
Fuerza	(N)
Presión	(mbar)
Resistencia eléctrica	(Ω)

Tabla 2 Siglas

SIGNIFICADO	SIGLAS
Global Positioning System	GPS
Global'naya Navigatsionnaya Sputnikovaya Sistema	GLONASS
National Marine Electronics Association	NMEA
General Packet Radio System	GPRS
Global System for Mobile Communications	GSM
Módulo de Identificación de Abonado	SIM
Short Message Service	SMS
International Mobile Equipment Identity	IMEI
Documento Nacional de Identidad	DNI
Low Power Wide Area Network	LPWAN
Internet of Things	IoT
Ultra Narrow Band	UNB
Machine to Machine	M ² M
Bandas de radio industriales, científicas y médicas	ISM
SubMiniature Version A	SMA
General Purpose Input/Output	GPIO

Entorno de desarrollo	IDE
Operative System	OS
Identificador	ID
Porting Authorization Code	PAC
Modulación por desplazamiento de fase (dos fases)	BPSK
Modulación por desplazamiento de frecuencia.	FSK
Modulación por desplazamiento de fase	PSK
Ground	GND
Alimentación	Vcc
Recepción	RX
transmisión	TX
Weak Signal Propagation Reporter	WSPR
Unión Internacional de Telecomunicaciones	UIT
Unión de Radioaficionados Españoles	URE
Surface Mounting Device	SMD
Software Defined Radio	SDR
Long Range	LoRa
Automatic Packet Reporting System	APRS
Slow Scan Digital Video	SSDV
Radioteletipo	RTTY
In Circuit Serial Programming	ICSP
Inter-Integrated Circuit	I ² C
Signal Clock	SCL
Signal Data	SDA
Placa de Circuito Impreso	PCB
Serial Peripheral Interface	SPI
Signal Clock	SCLK
Master Output Slave Input	MOSI
Master Input Slave Output	MISO
Slave Select	SS
Universal Asynchronous Receiver-Transmitter	UART
Transistor-transistor logic	TTL
Internatinal Standard Atmosphere	ISA
Ultra High Frequency	UHF
Very High Frequency	VHF
Tiempo Universal Coordinado	UTC
Comma Separated Values	CSV
Keyhole Markup Language	KML
Control Tráfico Aéreo	CTR
Gestor Navegación Aérea de España y Sahara Occidental	ENAI
Aeropuertos Españoles y Navegación Aérea	AENA
Agencia Estatal de Meteorología	AEMET
Real Decreto	RD
Punto de Referencia del Aeropuerto	ARP
Publicación de Información Aeronáutica	AIP
Comprobación Operatividad de Sistemas	COS

RESUMEN

El objetivo de este documento consiste en documentar paso a paso como construir una sonda estratosférica, así como documentar un lanzamiento siguiendo los pasos anteriores.

La sonda irá equipada con diferentes equipos electrónicos, estará unida a un paracaídas y este a su vez a un globo de helio de gran tamaño.

El globo alcanzará unos 30.000 metros de altura, entonces estallará y el paracaídas se abrirá para que el aterrizaje de la sonda sea seguro.

Una vez recuperada la sonda se procede a la recopilación y la interpretación de los datos.

ABSTRACT

The purpose of this document is to document step by step how to build a stratospheric probe, as well as document a launch following the steps above.

The probe will be equipped with different electronic equipment, will be attached to a parachute and this in turn to a large helium balloon.

The balloon will reach about 30.000 meters high, then it will explode and the parachute will open so that the landing of the probe is safe.

Once the probe is recovered, the data is collected and interpreted.

SOMMARIO

L'obiettivo di questo documento è documentare passo dopo passo come costruire una sonda stratosferica, nonché documentare un lancio seguendo le istruzioni precedenti.

La sonda sarà dotata di diverse apparecchiature elettroniche, sarà collegata a un paracadute e questo a sua volta a un grande pallone ad elio.

Il pallone raggiungerà circa 30.000 metri di altezza, quindi scoppierà e il paracadute si aprirà in modo che l'atterraggio della sonda sia sicuro.

Una volta recuperato il probe, i dati vengono raccolti e interpretati.

1. JUSTIFICACIÓN

En los últimos años se han realizado varios proyectos utilizando globos de helio para llevar a la estratosfera sondas con cámaras y otros equipos electrónicos, sin embargo, un factor común entre ellos es el uso de equipos de comunicación poco eficientes o de no muy largo alcance, lo cual es un gran problema tanto para la recepción de datos a muy larga distancia, como para la posterior recuperación de la sonda.

Otro problema común, es el aislamiento térmico de la sonda, el cual no suele ser muy eficiente y provoca condensación, así como una vida limitada para las baterías debido a las bajas temperaturas.

Con este proyecto, se intenta realizar una mejora tanto en los sistemas de comunicación a muy larga distancia, como en los sistemas de aislamiento y transmisión del calor. Además, nos permitirá probar estos equipos de comunicaciones, así como una serie de sensores y placas electrónicas en condiciones de temperaturas extremas, fuertes vientos y sometiendo toda la electrónica a fuertes vibraciones y aceleraciones.

Una vez concluido el proyecto, dispondremos de un sistema de telemetría eficiente y eficaz que nos servirá más adelante para implementarlo en otros posibles vuelos o aplicaciones.

Probaremos las limitaciones de la red OG¹, así como del sistema de propagación WSPR, el cual nos podría ayudar a posicionar nuestra sonda incluso en vuelos transoceánicos.

Otra mejora que implementar es la de simplificar la estructura de la estación de control terrena, puesto que, con un solo teléfono inteligente con acceso a internet, se puede tener acceso a todos los datos.

¹ Red OG: es una red inalámbrica de bajo ancho de banda diseñada específicamente para conectar dispositivos IoT simples, de baja potencia y bajo costo a Internet.

Por último, también se perfeccionará el sistema de aislamiento, así como el control de temperatura en el interior de la sonda, de esta manera se podrá mantener el correcto funcionamiento de la electrónica y alargar la duración de las baterías.

Otro de los objetivos principales de este proyecto, es tratar de desarrollarlo en un margen de tiempo muy estrecho, normalmente la primera vez que se desarrolla un trabajo de este tipo, lleva mucho tiempo poder realizar el primer vuelo y en muchas ocasiones, el vuelo acaba en fracaso debido a errores en la implementación del prototipo.

En este caso, el objetivo es realizar un vuelo de prueba, desarrollando desde el inicio el diseño completo de cada uno de los sistemas que forman parte del globo sonda, este desarrollo deberá realizarse en aproximadamente tres meses y la idea es que se consiga desarrollar un sistema fiable que en un futuro permita enviar cualquier carga de pago sin el riesgo de perderla.

2. DESCRIPCIÓN DEL TRABAJO

A partir de este apartado, describiré por secciones, que componentes he utilizado, por qué razón, comparativas de componentes usados o que he pensado en usar, resultados de cálculos, estudios de termografía, etc...

Todo ello con la intención de poder documentar a fondo el trabajo para que cualquier persona tenga acceso a los datos y en un periodo corto de tiempo pueda desarrollar un proyecto similar con altas posibilidades de éxito.

Además, intentaré indicar porque hemos utilizado unos componentes y no otros similares, para evitar caer en errores típicos que, aunque pueden ser muy obvios, por falta de documentación y de experiencia, podrían llevar la misión al fracaso.

Esta descripción del trabajo desarrollará varios puntos clave como el desarrollo de la aviónica², la recepción de datos en tierra, la fabricación del vehículo, propulsión, simulaciones, meteorología, procesos administrativos, logística, financiación, lanzamiento de la sonda, recuperación de esta y diagrama de Gantt.

² Aviónica: sistemas electrónicos usados en aeronaves, satélites artificiales y naves espaciales, tanto sistemas de comunicación y navegación como sus indicadores y elementos de manejo.

Finalmente se expondrá un apartado con el presupuesto de un experimento de estas características y un cuadro de condiciones técnicas para desarrollar otro prototipo similar con garantías de éxito.

3. OBJETIVOS

A continuación, expongo los objetivos esperados de este proyecto, al final del documento se expondrá otra tabla detallando cuales se cumplieron, cuáles no, y porqué.

Tabla 3 Objetivos esperados

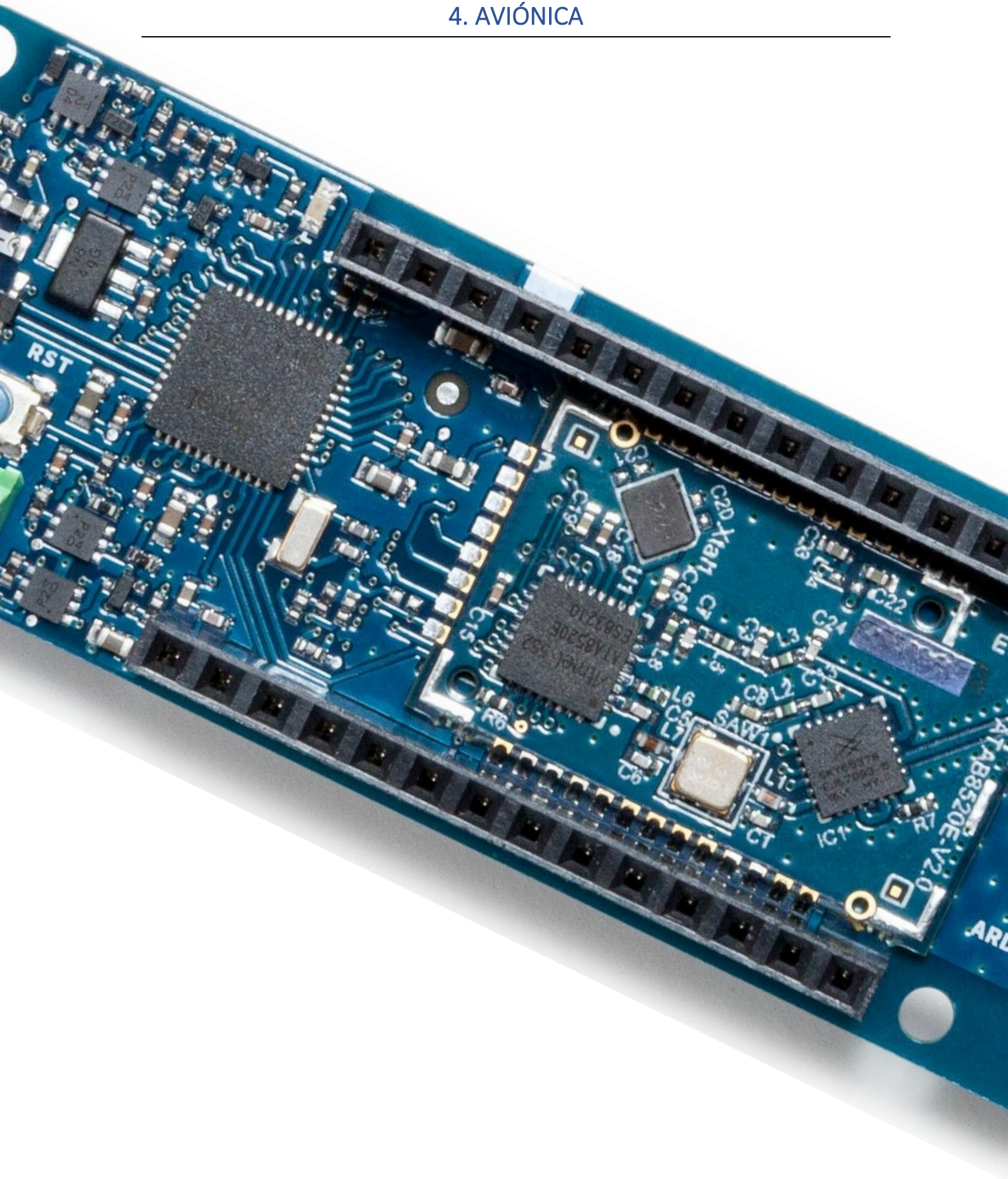
OBJETIVOS ESPERADOS
1º REALIZAR UN LANZAMIENTO DE UN GLOBO SONDA
2º RECUPERAR LA SONDA ENBUENAS CONDICIONES
3º CAPTACIÓN DE IMÁGENES DURANTE TODO EL VUELO
4º ESTUDIAR EL FUNCIONAMIENTO DE SIGFOX EN LA ESTRATOSFERA
5º DETERMINAR ALTURA MÁXIMA DE LA RED GPRS/GSM
6º REALIZAR ESTUDIO DE PROPAGACIÓN DE HF
7º ESTABLECER COMUNICACIÓN CONTINUA CON LA SONDA
8º TOMA DE DATOS METEOROLÓGICOS CON UN AUTÓMATA

Son objetivos acordes a un primer lanzamiento de estas características, sin embargo, se incluyen tres objetivos que se pueden considerar bastante más avanzados, el primero de ellos es probar un sistema no comercial de SigFox.

El segundo objetivo es establecer comunicación continua con la sonda, ya que muchas sondas se recuperan, pero muy pocas de ellas establecen comunicación continua, lo suelen realizar durante el ascenso y una vez que se llega cerca de la zona de aterrizaje.

Por último, está el objetivo de realizar un estudio de propagación ionosférica de HF mediante un equipo de (14 Mhz).

4. AVIÓNICA



4.1 SUBSISTEMA DE COMUNICACIONES

A continuación, se detallarán los sistemas de telecomunicaciones utilizados en este proyecto, así como otras posibles soluciones a la comunicación.

Además, se hará también una pequeña introducción a las tecnologías en las que están basados los aparatos utilizados, con la intención de comprender el funcionamiento de estas y poder así elegir de forma objetiva el mejor sistema de recuperación.

Por otro lado, me gustaría recordar que este apartado, fue el que más tiempo ha requerido a la hora de realizar el proyecto, tanto en la documentación como en la programación y posteriores pruebas, por ello considero que es la parte más importante en el proyecto y voy a tratar de profundizar un poco más en este apartado.

4.1.1 SISTEMAS UTILIZADOS

Para tener la certeza de la recuperación exitosa de la sonda, lo primero que queda claro, es el hecho de que debe existir un sistema de comunicaciones por capas, esto es que los sistemas deben utilizar tecnologías diferentes para tratar de que un sistema sea capaz de suplir las carencias de otro, así como para tratar de aumentar el alcance de las comunicaciones.

Otro factor muy importante es la duplicidad de equipos, para que en caso de que falle un equipo, tengamos siempre otro operativo, en este caso, la duplicidad de equipos la hemos implementado en el subsistema de potencia, para asegurarnos de que, si falla un sistema, los otros seguirían funcionando, los explicaré más adelante con más detalle.

Respecto a las comunicaciones por capas, en este lanzamiento hemos utilizado tres capas, o lo que es lo mismo tres tecnologías diferentes que se enumeran a continuación por orden de importancia, este nivel de importancia es determinado por la precisión de la posición GPS obtenida.

4.1.1.1 SIGFOX

SigFox es una compañía que ofrece servicios de red de cobertura amplia de bajo consumo o en inglés: *Low Power Wide Area Network* (LPWAN).

Es decir, una red inalámbrica de bajo consumo y de muy largo alcance para soluciones del internet de las cosas (IoT). Permite tasas de transferencia de 12 bytes y funciona con tecnología de transmisión *Ultra Narrow Band* (UNB), esto significa que emplea canales estrechos del espectro para alcanzar muy largas distancias con muy poca potencia.

Este tipo de tecnología está muy enfocado en el *Machine To Machine* (M2M), lo cual nos permite comunicar varias máquinas entre ellas y que estas actúen en función de unas reglas predeterminadas que dependen de los mensajes que son recibidos desde otras máquinas.

Es una tecnología que utiliza la infraestructura de los repetidores de telefonía para situar sus antenas, estas una vez que reciben la señal, la suben directamente a la nube, aunque también puede ser al contrario, es decir, nosotros podemos enviar desde la nube un mensaje al dispositivo SigFox, lo que convierte a este dispositivo en *semi-duplex*, lo cual nos permite enviar y recibir mensajes, pero no en el mismo momento.

SigFox utiliza las bandas bidireccionales de radio sin licencia, industrial, *scientific and medical radio bands* (ISM) que en España opera en 868 (Mhz).

Es muy importante destacar que el servicio de SigFox es de pago, pero con la compra de un dispositivo SigFox, viene incluida una suscripción de dos años.

Una vez acabado el período de prueba, se tiene que contratar otra suscripción.

Existen varias opciones, las principales son la de 140 mensajes diarios y 3 años por unos 22,7 € al año. También la opción de 50 mensajes diarios y 5 años por 10,8 € al año.

El operador que ofrece el servicio en España es Cellnex Telecom y con sus repetidores repartidos por toda la orografía, ha conseguido cubrir casi por completo todo el mapa de la península ibérica como podemos ver en la siguiente imagen.

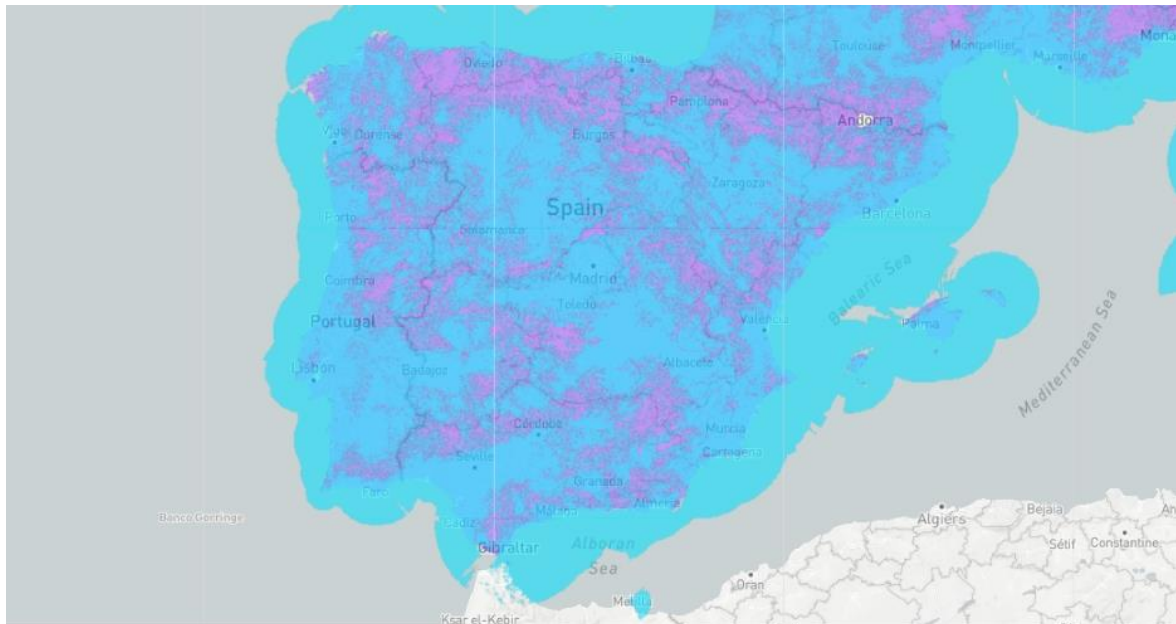


Figura 1 Cobertura SigFox

Se pueden apreciar dos tonalidades de color, el tono azul representa las zonas de cobertura, mientras que el morado representa zonas de poca o nula cobertura, ya sea debido a la orografía o a la falta de repetidores en el área.

Es muy importante tener en cuenta que en zonas urbanas el alcance está comprendido entre los 3 y los 10 (Km), en áreas rurales varía entre los 30 y los 50 (Km) y hasta los 1.000 (Km) con visión directa entre el emisor y el receptor.

Esto último es muy importante para nuestra aplicación, puesto que, aunque caiga en una zona sin cobertura, durante todo el vuelo tendrá cobertura, aun así, en la zona que está previsto que caiga no existen problemas de cobertura.

Para nuestra aplicación hemos decidido utilizar la placa MKRFOX 1200 de Arduino.

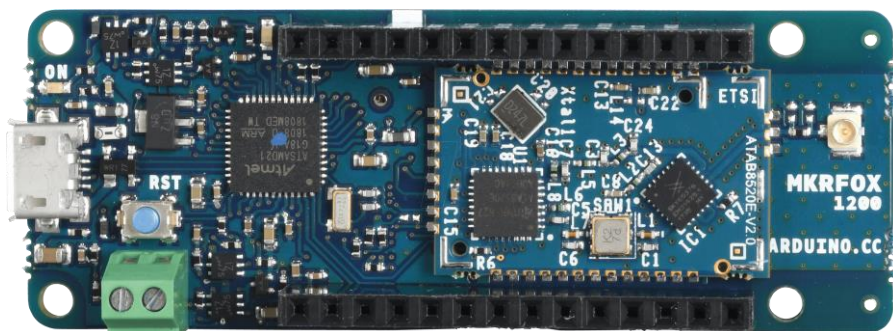


Figura 2 MKRFOX 1200

Esta placa incluye una suscripción de dos años con unos 140 mensajes diarios con un tamaño máximo de 12 bytes, también ofrece 4 mensajes de bajada de 8 bytes.

Esta placa está compuesta por un microcontrolador SAMD21, el cual se utiliza para cargar nuestro programa y controlar los GPIOs, mientras que el módulo SigFox, está compuesto por un ATA8520, también incluye un conector de antena mini-SMA para utilizar una antena optimizada a 868 (Mhz).

En nuestra aplicación será de mucha utilidad el termómetro que viene integrado en el ATA8520, el cual mide en un rango comprendido entre los -55° C a los 125° C.

En la parte izquierda podemos ver un conector micro USB, así como unas bornas para alimentarlo con baterías.

El USB alimenta la placa a 5 (V), pero reduce el voltaje a 3,3 (V), esto es muy importante puesto que, a la hora de alimentar la placa a través de las bornas, nunca deberemos sobrepasar los 3,3 (V), de esta forma el integrado funcionará en sus condiciones óptimas y no se quemará.

Cuando el módulo comienza a funcionar, sube los datos a una base de datos, en la cual podremos reenviar los datos a otras plataformas, en este caso deseo enviar una posición GPS y en caso de que no sea capaz de conseguir una posición, entonces enviará una temperatura procedente del termómetro del integrado.

Para poner en funcionamiento esta placa, deberemos realizar los pasos detallados en las siguientes secciones.

INSTALACIÓN IDE ARDUINO:

Para poder usar esta placa, deberemos descargar un IDE específico de Arduino, en el que podremos escribir el programa y posteriormente cargarlo en la placa.

El IDE de Arduino es un *software* libre multiplataforma, por lo que podremos descargarlo y usarlo en casi todos los sistemas operativos. Para descargarlo nos dirigiremos a la página oficial de Arduino, una vez allí pulsaremos en el menú principal sobre *Software>Downloads...*



Figura 3 Software download

Entraremos así en el menú de descargas y entre todas las opciones de descarga elegiremos la opción que mejor se adapte a nuestro OS, en mi caso descargaré la “Windows ZIP file for non admin install”.

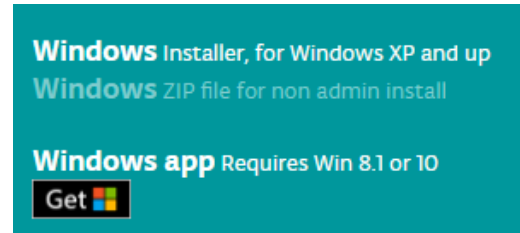


Figura 4 Non admin install

Una vez descargado, lo descomprimos y ya podemos abrir al programa a través del ejecutable.

CONFIGURACIÓN IDE ARDUINO:

Para poder trabajar con esta placa, primero deberemos añadirla en nuestro IDE de Arduino, para ello en el menú principal pinchamos sobre los siguientes botones: Herramientas>Placa>Gestor de tarjetas...

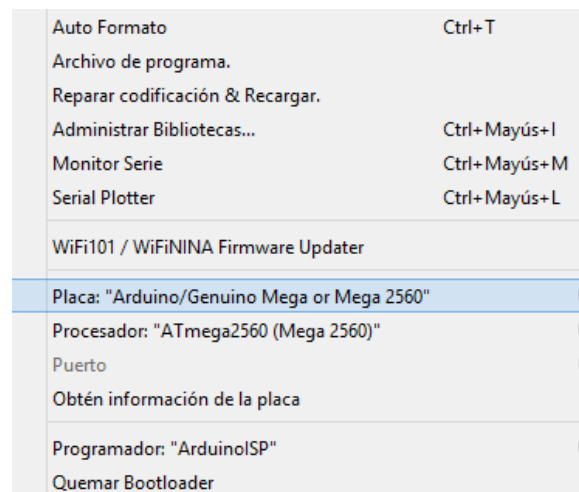


Figura 5 Selección placa

Una vez realizado este paso, nos aparecerá una ventana emergente en la que deberemos buscar MKRFOX 1200 y una vez lo encontremos, deberemos instalar el controlador de Arduino SAMD boards.

Una vez instalado, en el menú principal si buscamos placas, podremos ya encontrar toda la familia MKR, en nuestro caso seleccionaremos la placa MKRFOX 1200.

Arduino SAMD (32-bits ARM Cortex-M0+) Boards
Arduino/Genuino Zero (Programming Port)
Arduino/Genuino Zero (Native USB Port)
Arduino/Genuino MKR1000
Arduino MKRZero
Arduino MKRFox1200
Adafruit Circuit Playground Express
Arduino M0 Pro (Programming Port)
Arduino M0 Pro (Native USB Port)

Figura 6 Arduino SAMD

CONFIGURACIÓN MKRFOX 1200:

Para poder tener acceso a la plataforma SigFox, deberemos primero de todo obtener el ID y el PAC de nuestro dispositivo, el ID es un identificador que es único para cada dispositivo, mientras que el *Porting Authorization Code* (PAC), viene a ser como un código de seguridad que solo es accesible al usuario del dispositivo, puesto que con este código se puede tomar el control del dispositivo.

Para obtener estos datos, deberemos cargar un programa en nuestra placa y cuando este se ejecute, nos mostrará estos dos datos, los cuales los copiaremos para darnos de alta en SigFox.

INSTALACIÓN LIBRERÍA SIGFOX PARA ARDUINO:

En el menú principal del IDE de Arduino, seleccionaremos los siguientes botones: Programa>Incluir, luego seleccionamos: Librerías>Gestionar librerías...

Programa	Herramientas	Ayuda
Verificar/Compilar		Ctrl+R
Subir		Ctrl+U
Subir Usando Programador		Ctrl+Mayús+U
Exportar Binarios compilados		Ctrl+Alt+S
Mostrar Carpeta de Programa		Ctrl+K
Incluir Librería		
Añadir fichero...		

Figura 7 Librerías Arduino

Buscaremos SIGFOX e instalaremos la librería Arduino SigFox for MKRFOX1200.

Una vez instalado procedemos a la obtención de nuestro ID y PAC, para ello cargaremos el siguiente programa:

```
1 #include <SigFox.h>
2
3 void setup() {
4   // Configuración del monitor serie
5   Serial.begin(9600);
6   delay(500);
7
8   // Iniciamos módulo Sigfox
9   if(!SigFox.begin()){
10    Serial.println("Error al iniciar módulo Sigfox");
11    return;
12  }
13
14  // Mostramos la información del ID y de PAC
15  Serial.println("ID = " + SigFox.ID());
16  Serial.println("PAC = " + SigFox.PAC());
17
18  // Apagamos el módulo
19  SigFox.end();
20 }
21
22 void loop() {
23   // En el loop no hacemos nada
24
25 }
```

Figura 8 Código PAC e ID

Una vez cargado el código en la placa, accedemos al monitor serial a 9.600 (Bd) y allí encontraremos el ID y el PAC, los copiamos y ya procedemos a dar de alta nuestro dispositivo en SigFox.

REGISTRAR MKRFOX 1200 EN SIGFOX:

Para registrarnos, deberemos acceder a la página <https://buy.sigfox.com/>, allí aparecerá un apartado en el que podremos elegir entre comprar conectividad o activar nuestro *kit*, elegiremos esta última opción.

Una vez seleccionada la opción de activar el *kit*, aparecerá una ventana en la que deberemos seleccionar nuestro país, en nuestro caso buscamos y seleccionamos España, pulsaremos en el botón “*next*” de la esquina superior derecha y entraremos en otra ventana en la que nos pedirá el ID y el PAC, así como datos relacionados a que uso vamos a dar a nuestro dispositivo, procedemos a rellenar todos los apartados y volvemos a pulsar en “*next*”.

A continuación, aparece otra ventana en la que deberemos crear una cuenta personal para la plataforma SigFox, creamos nuestra cuenta y pulsando otra vez en “*next*”, accedemos a un menú de confirmación, allí aceptamos todos los datos anteriores y ya estaríamos en condiciones de empezar a usar nuestro dispositivo.

ACCESO A BACKEND Y CONFIGURACIÓN:

Para acceder a la plataforma, deberemos hacerlo a través de este enlace: <https://backend.sigfox.com/auth/login>, en el aparecerá una ventana en la que añadiremos el correo y la contraseña de la cuenta que hemos creado en SigFox, una vez lo tengamos, pulsamos en “Sign in” y accedemos a nuestra base de datos.

Accedemos a una ventana principal en la que tenemos cuatro pestañas principales: *Device*, *Device type*, *User* y *Group*.

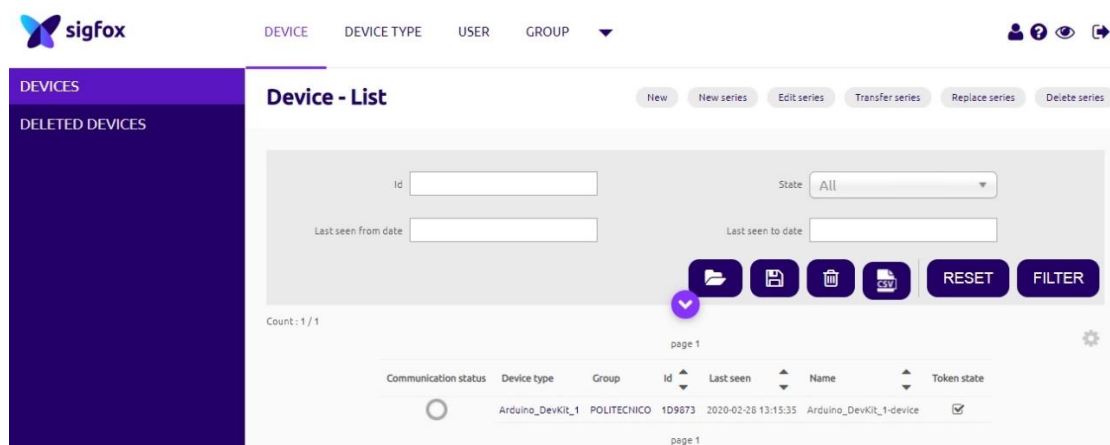


Figura 9 Ventana principal Backend

En la pestaña de *Device*, encontraremos el MKRFOX 1200, si pulsamos sobre su ID, nos aparecerá en la columna izquierda varias opciones, entre ellas estará la pestaña de mensajes, si pulsamos en ella, encontraremos una página similar a esta, pero en blanco.

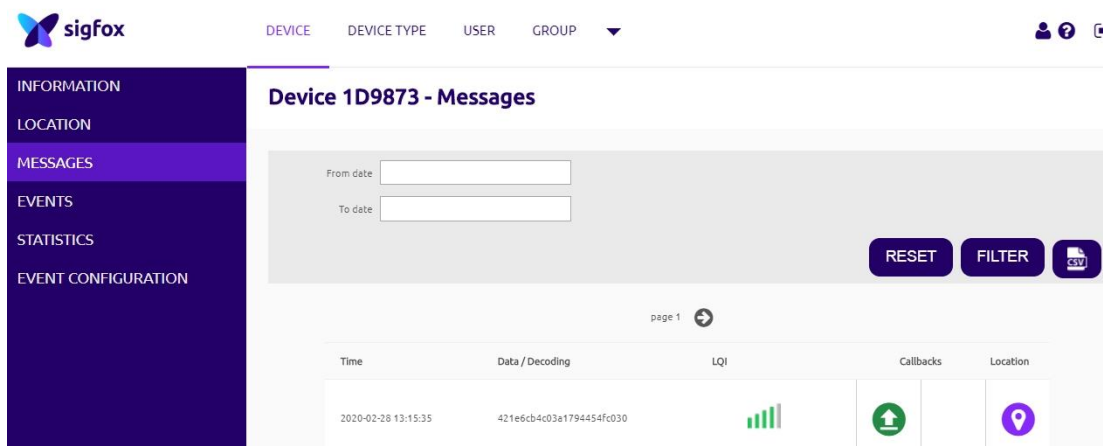


Figura 10 Mensajes Backend

En esta ventana se pueden ver todos los mensajes ordenados por fecha y hora, además nos aparece información relativa a la calidad de la cobertura, los *Callbacks*³ y la posición.

Cabe destacar que el mensaje aparece en hexadecimal, para poder decodificarlo correctamente sin tener que estar usando una calculadora, utilizaremos los *Callbacks*, pero antes quiero hablar sobre la posición.

POSICIONAMIENTO ATLAS:

Si vamos a un mensaje y pulsamos sobre localización, aparecerá una ventana con una imagen de Google maps.

Esta es una aproximación de la posición de nuestro dispositivo, cada posición que nos ofrece se actualiza con cada nuevo mensaje enviado, la principal característica de esta herramienta es que no se utiliza ninguna antena GPS para obtener la posición, utiliza una nueva tecnología denominada ATLAS Native y está basado únicamente en la red SigFox y para funcionar no necesita ningún componente de *hardware* o *software* adicional.



Figura 11 Posición ATLAS

Es un servicio adicional al servicio contratado, pero es gratuito, a cambio SigFox se reserva los derechos de procesamiento de datos, con la intención de probar

³ Callback: Un Callback (llamada de vuelta) es una función que se ejecutará **después** de que otra función haya terminado de ejecutarse, de aquí el nombre de Callback.

diferentes algoritmos de aprendizaje automático que ayuden a reducir la incertidumbre de la ubicación a no más de 800 metros.

Además, existen otras variantes de esta tecnología basadas en las redes WIFI y *Bluetooth*, con la intención de potenciar este sistema y mejorarlo.

Sin embargo, en la actualidad este sistema está todavía en desarrollo y tiene ciertas limitaciones, puesto que la precisión de la ubicación es muy baja, aun así, existen métodos para intentar obtener ubicaciones con bastante menor error con los datos derivados de estas imágenes. Pero para ello es importante conocer cómo funciona exactamente esta tecnología.

Para saber cómo funciona, hemos de tener en cuenta que, a modo de comprobación de errores, así como de asegurarse que el mensaje llega a su destino, nuestra placa envía varias réplicas del mensaje, estas se pueden ver claramente en el dominio de la frecuencia.

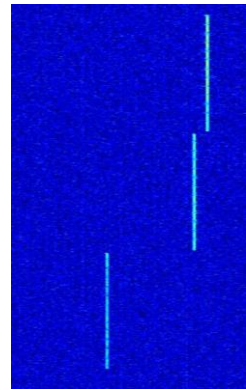


Figura 12 Señal SigFox

Así pues esta señal que está modulada en BPSK, viaja hacia todas direcciones hasta llegar a un repetidor, teniendo en cuenta el gran alcance de esta tecnología, puede ser que llegue a más de un repetidor, de esta forma los repetidores reciben la señal y pueden calcular el momento en que se recibe la señal y el momento en que esta termina, compartiendo esta información con el resto de repetidores que reciben la señal, se puede hacer una estimación de un área posible en la que se encuentra el dispositivo, puesto que se hace un cálculo de desfase de tiempos entre repetidores.

Esta información es esencial para calcular un área aproximada, pero existen otros muchos datos, como la potencia de señal que reciben los diferentes repetidores, así como las antenas con las que reciben el mensaje, así con diferentes antenas omnidireccionales y sectoriales, se puede reducir aún más la incertidumbre.

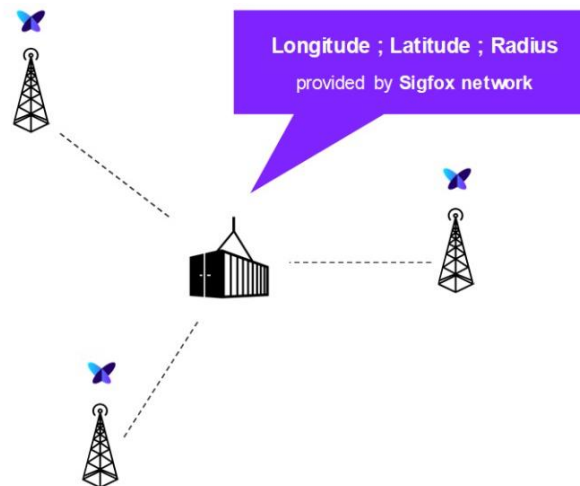


Figura 13 ATLAS native

Otro factor fundamental es la orografía, puesto que, si estamos en una zona montañosa, algunos repetidores nos pueden recibir con algunas zonas de sombra, esto en algunos casos, más que un problema se convierte en una gran utilidad para saber dónde está nuestro dispositivo, estas zonas de sombra nos pueden ayudar a trazar líneas que sigan los bordes de estas sombras y que se cruzan en puntos donde la probabilidad de encontrar el dispositivo es muy alta.

Otra ventaja es cuando el terreno es muy llano, esto nos permite que todos los repetidores nos reciban a la perfección y establezcan una circunferencia casi perfecta en la que el perímetro se supone como una zona equipotencial, teniendo esto claro, se puede fácilmente seleccionar dos puntos del perímetro y trazar una mediatriz entre ellos, esto nos dará una línea en la que se encuentra el punto central, si realizamos otra vez esta operación, obtendremos una segunda línea que se cruzará con la anterior en el centro de la circunferencia, este es el punto en el que se encuentra nuestro dispositivo.

Otro método para hallar el punto de emisión es hacer una circunferencia que describa el perímetro de la zona de azul oscuro, se puede realizar en un editor de imagen. Una vez que se ha realizado esta circunferencia, se puede crear otra de radio mitad, esta se debe situar de forma que sea tangente interior a la circunferencia mayor.

Si ahora se realiza una línea que una el centro de la circunferencia menor con el punto de tangencia, entonces esta línea cortará a la circunferencia menor en un punto que coincidirá con el centro de la circunferencia mayor.

Este es el punto en el que se encuentra el dispositivo emisor.

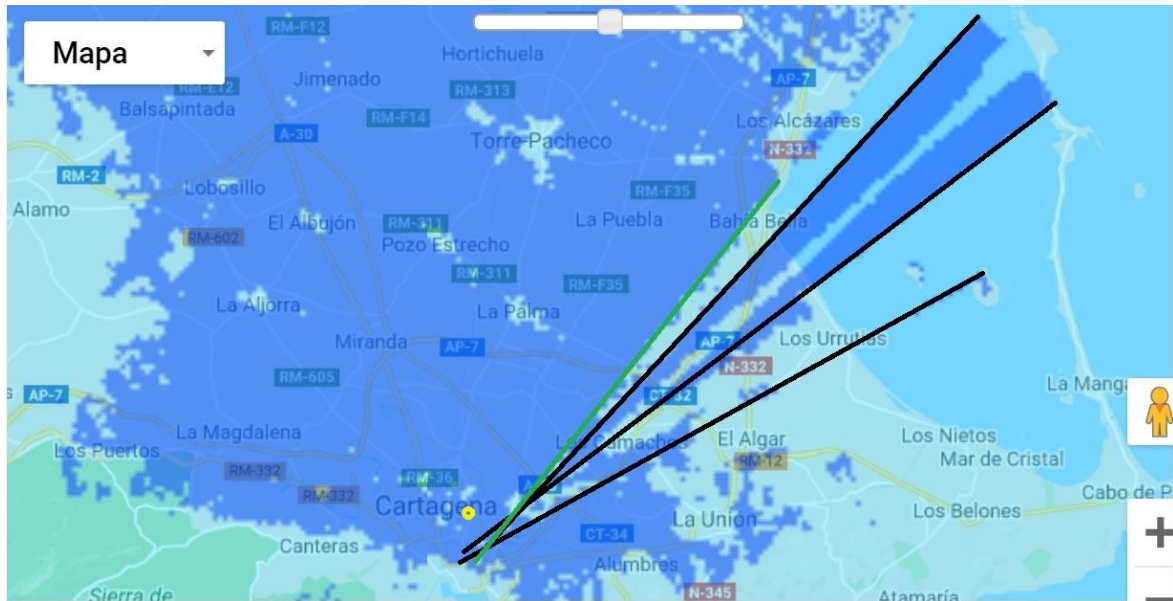


Figura 14 Demoras ATLAS

Con esta información ya podemos hacernos una idea de cómo funciona esta tecnología, como obtener la información, así como ser capaces de mejorar la información dada mediante las técnicas descritas anteriormente.

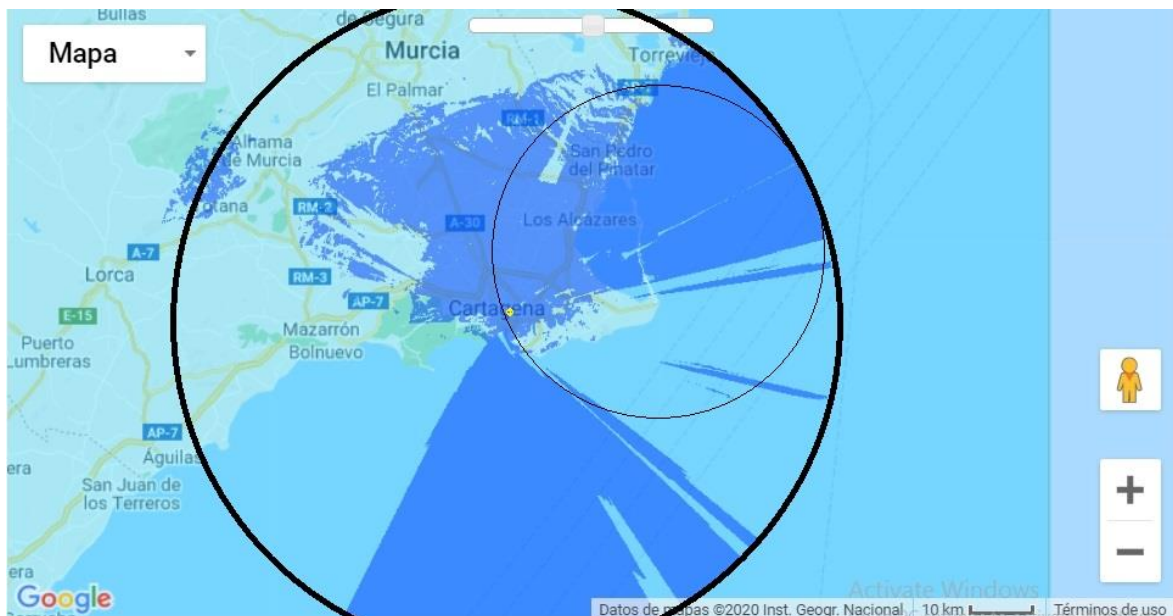


Figura 15 Centro circunferencia ATLAS

PROGRAMA MKRFOX 1200:

Aunque como ya hemos visto, el módulo SigFox ya nos da una posición estimada de nuestra sonda, esta es muy poco fiable por el gran error que tiene, por esta razón vamos a incorporar una antena GPS a nuestro Arduino MKRFOX 1200, esto hará que la precisión sea mucho mayor, con un margen de error de aproximadamente cinco metros.

Para ello, he utilizado un módulo GPS GN-701 y lo he conectado al Arduino alimentándolo a 3,3 (V), conectándolo a GND de Arduino y por último conectando el puerto serie, en el caso del MKRFOX 1200, conectaríamos el RX del GPS al pin 14 y el TX del GPS al pin 13.

Tabla 4 CONEXIÓN MKRFOX 1200 - GPS

MKRFOX 1200	GPS
GND	GND
3.3 V	VCC
PIN DIGITAL 14	RX
PIN DIGITAL 13	TX

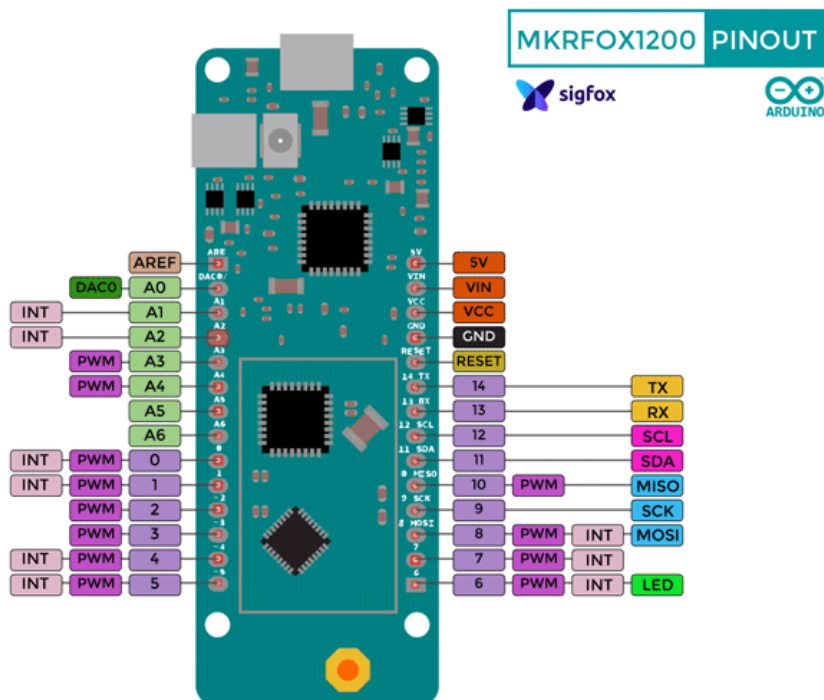


Figura 16 PinOut MKRFOX 1200

Una vez que lo tenemos correctamente conectado, procedemos a cargar el programa desde el IDE de Arduino.

El código consiste en enviar mensajes cada dos minutos aproximadamente desde que se inicia el módulo, en caso de que se encuentren los satélites rápidamente.

En caso de que no encuentre los satélites y no sea capaz de enviar una posición, entonces tomará una temperatura interna y la enviará, de esta forma no se perderá la oportunidad de enviar información que además también nos brinda una oportunidad de localizarlo con ATLAS.

En el tiempo de ejecución existe un contador que va midiendo el tiempo transcurrido desde el encendido de la placa, una vez que este llegue a un valor determinado por el usuario, entonces el tiempo máximo entre mensajes será de 1 minuto, la idea es que esta variable de tiempo sea la que está determinada como el tiempo de subida, es decir, el momento a partir del cual el globo explota y la sonda comienza su descenso.

Por ejemplo, si está previsto que el vuelo dure dos horas y el tiempo hasta la explosión es de una hora y media, esto significa que esa hora y media debe ser el valor para introducir en el programa, pero debe ser determinada en milisegundos, es decir serían unos 5.400.000 (ms)

La idea de cambiar la velocidad a la que se envían los mensajes es debido a que los mensajes están limitados a 140 mensajes diarios, por ello a partir de la explosión se envían todos los que se puedan, ya que una vez en tierra puede que se pierda la cobertura. Y tener una última posición a pocos minutos del aterrizaje puede ser la clave en la recuperación de la sonda.

A continuación, procedo a explicar el código.

Este código es de uso público, fue publicado originalmente en Github, el programa fue modificado para poder adaptarlo a las necesidades de este proyecto, además inicialmente este programa tenía funciones que no son necesarias para nosotros y por ello han sido eliminadas.

```
#include <SigFox.h>
#include <ArduinoLowPower.h>
#include <TinyGPS.h>

#define WAITING_TIME 1 // bajo tiempo a 1 minuto, antes 15
#define GPS_INFO_BUFFER_SIZE 128

bool debug = false; // PONER EN FALSE ANTES DE LANZARLO

TinyGPS gps;//GPS Object

int year; //GPS data variables
byte month, day, hour, minute, second, hundredths;
unsigned long chars;
unsigned short sentences, failed_checksum;
char GPS_info_char;
char GPS_info_buffer[GPS_INFO_BUFFER_SIZE];
unsigned int received_char;
bool message_started = false;
int i = 0;

// GPS coordinate structure, 12 bytes size
struct gpscoord {
    float a_latitude; // 4 bytes
    float a_longitude; // 4 bytes
    float a_altitude; // 4 bytes
};

float latitude = 0.0f;
float longitude = 0.0f;
float altitud = 0;
```

En esta primera parte, procedo a la llamada a librerías, así como a la declaración de variables, cabe destacar el uso de la librería Tiny GPS, de la cual sale casi toda la estructura de variables para el uso del GPS, por otro lado, también cabe destacar el uso del *boolean debug*, para hacer comprobaciones del funcionamiento del programa.

Una pieza clave a la hora de realizar el lanzamiento es asegurarse que el *debug* se queda en false, si se queda en true, no funcionaría.

En caso de realizar más lanzamientos utilizando este proyecto, se podrían programar unas líneas para que en caso de que debug esté en true, pero no se conecte ningún USB, entonces cambie a false.

De esta forma se podrían eliminar muchos problemas.


```
void Wait(int m, bool s) {
    //m minutes to wait
    //s slow led pulses
    if (debug) {
        Serial.print("Espere: "); Serial.print(m); Serial.println(" minutos.");
    }
    digitalWrite(LED_BUILTIN, LOW);
    if (s) {
        int seg = m * 30;
        for (int i = 0; i < seg; i++) {
            digitalWrite(LED_BUILTIN, HIGH); //LED on
            delay(500);
            digitalWrite(LED_BUILTIN, LOW); //LED off
            delay(500);
        }
    } else {
        int seg = m * 15;
        for (int i = 0; i < seg; i++) {
            digitalWrite(LED_BUILTIN, HIGH); //LED on
            delay(1000);
            digitalWrite(LED_BUILTIN, LOW); //LED off
            delay(3000);
        }
    }
}
```

A continuación creo la función wait(), esta función necesita dos valores para ejecutarse, el primero es el tiempo en minutos, mientras que el segundo es un booleano y se utiliza para diferenciar que tipo de espera se está ejecutando, es decir el tiempo de espera va a ser el mismo, pero la frecuencia con la que se enciende y apaga el led integrado va a variar y esto nos sirve para comprobar físicamente como se está ejecutando el programa.

```
void SendSigfox(String data) {
    if (debug) {
        Serial.print("Enviando... "); Serial.println(data);
        if (data.length() > 12) {
            Serial.println("Mensaje muy largo, solo los primeros 12 bytes serán enviados.");
        }
    }
}
```

Esta función es la que se encarga de enviar datos a SigFox, en caso de que exista un dato que enviar, si el mensaje es demasiado largo, solo enviará los primeros 12 bytes.

```
SigFox.begin(); // Start the module
// Wait at least 30mS after first configuration (100mS before)
delay(100); // Wait at least 100mS
SigFox.status(); // Clears all pending interrupts
delay(1);
if (debug) SigFox.debug();
delay(100);

SigFox.beginPacket();
SigFox.print(data);

if (debug) {
    int ret = SigFox.endPacket(true); // send buffer to SIGFOX network and wait for a response
    if (ret > 0) {
        Serial.println("No transmisión.");
    } else {
        Serial.println("Trasmisión correcta.");
    }
}

Serial.println(SigFox.status(SIGFOX));
Serial.println(SigFox.status(ATEL));

if (SigFox.parsePacket()) {
    Serial.println("Respuesta desde el servidor:");
    while (SigFox.available()) {
        Serial.print("0x");
        Serial.println(SigFox.read(), HEX);
    }
} else {
    Serial.println("No se pudo obtener respuesta del servidor.");
    Serial.println("Comprueba la cobertura Sigfox en tu area.");
    Serial.println("Si estas en interiores, acercate a una ventana.");
}
Serial.println();
} else {
    SigFox.endPacket();
}
SigFox.end();
}
```

Para realizar esta operación, con una serie de funciones, se inicializa el módulo SigFox y posteriormente se envía el paquete, se comprueba que la cobertura es correcta y que el paquete se ha enviado, entonces apaga el módulo SigFox y sale de la función.

También realiza una pequeña función para leer un valor que llegue desde el servidor, en caso de que necesitemos controlar algo de forma remota.

En este caso, no se ha utilizado, pero con una modificación en el *Backend*, se podría configurar fácilmente, por ello dejo las líneas de código, por si en un futuro decido utilizarlo.

```
String ConvertGPSdata(const void* data, uint8_t len) {
    uint8_t* bytes = (uint8_t*)data;
    String cadena ;
    if (debug) {
        Serial.print("Tamaño: "); Serial.println(len);
    }

    for (uint8_t i = len - 1; i < len; --i) {
        if (bytes[i] < 12) {
            cadena.concat(byte(0)); // Not tested
        }
        cadena.concat(char(bytes[i]));
        if (debug) Serial.print(bytes[i], HEX);
    }

    if (debug) {
        Serial.println("");
        Serial.print("String a enviar: "); Serial.println(cadena);
    }

    return cadena;
}
```

Esta función transforma los *floats* de la latitud, longitud y altura en un *string* que se puede ya enviar a SigFox, toma los datos y los va concatenando.

```
String GetGPSpositon() {

    int messages_count = 0;
    String pos;
    int segundos_espera = 0;

    if(millis() > 4980000){
        segundos_espera = 60; // si el tiempo de vuelo es mayor
    }
    else{
        segundos_espera = 120; // si el tiempo es menor de 1h 23
    }

    if (debug) Serial.println("GPS encendido");
    Wait(1, false);
    while (messages_count < segundos_espera) { //2 min
        while (Serial1.available()) {

            int GPS_info_char = Serial1.read();

            if (GPS_info_char == '$') messages_count++; // start

            if (debug) {
                if (GPS_info_char == '$') { // start of message
                    message_started = true;
                    received_char = 0;
                } else if (GPS_info_char == '*') { // end of message
                    for (i = 0; i < received_char; i++) {
                        Serial.write(GPS_info_buffer[i]); // writes the
                    }
                    Serial.println();
                }
            }
        }
    }
}
```

```
        message_started = false; // ready for the new message
    } else if (message_started == true) { // the message is already started and I
        if (received_char <= GPS_INFO_BUFFER_SIZE) { // to avoid buffer overflow
            GPS_info_buffer[received_char] = GPS_info_char;
            received_char++;
        } else { // resets everything (overflow happened)
            message_started = false;
            received_char = 0;
        }
    }
}

if (gps.encode(GPS_info_char)) {
    gps.f_get_position(&latitude, &longitude);
    altitud = gps.altitude() / 100;

    // Store coordinates into dedicated structure
    gpscoord coords = {altitud, longitude, latitude};

    gps.crack_datetime(&year, &month, &day, &hour, &minute, &second, &hundredths);

    if (debug) {
        Serial.println();
        Serial.println();
        Serial.print("Latitud/Longitud: ");
        Serial.print(latitude, 5);
        Serial.print(", ");
        Serial.println(longitude, 5);
        Serial.println();
        Serial.print("Fecha: "); Serial.print(day, DEC); Serial.print("/");
        Serial.print(month, DEC); Serial.print("/"); Serial.print(year);

        Serial.print(" Hora: "); Serial.print(hour, DEC); Serial.print(":");
        Serial.print(minute, DEC); Serial.print(":"); Serial.print(second, DEC);
        Serial.print("."); Serial.println(hundredths, DEC);
        Serial.print("Altitud (metros): "); Serial.println(gps.f_altitude());
        Serial.print("Rumbo (grados): "); Serial.println(gps.f_course());
        Serial.print("Velocidad(kmph): "); Serial.println(gps.f_speed_kmph());
        Serial.print("Satelites: "); Serial.println(gps.satellites());
        Serial.println();
    }

    gps.stats(&chars, &sentences, &failed_checksum);
    if (debug) Serial.println("GPS apagado");
    pos = ConvertGPSdata(&coords, sizeof(gpscoord)); //Send data
    return pos;
}
}
}
pos = "No Signal";
}
```

Esta función es la que obtiene la posición GPS, así como la altura y otros datos de la trama NMEA, para ello primero determina cuánto tiempo lleva ejecutándose el programa para determinar si se deja más o menos tiempo para encontrar los satélites, luego se procede a obtener una trama NMEA completa.

En caso de obtener una trama, saca de esta la latitud, longitud y la altura. Guarda estos datos en una variable y devuelve estos datos al flujo principal del programa para su posterior uso.

```
void setup() {
  if (debug) {
    Serial.begin(9600);
    while (!Serial) {} // wait for serial port to connect.
    Serial.println("Conexión serial");
  }

  //Serial1 pins 13-14 for 3.3V connection to GPS.
  Serial1.begin(9600);
  while (!Serial1) {}
  if (debug) {
    Serial.println("GPS Conectado");
  }

  if (!SigFox.begin()) {
    Serial.println("Error en el shield SIGFOX");
    return;
  }

  // Enable debug led and disable automatic deep sleep
  if (debug) {
    SigFox.debug();
  } else {
    SigFox.end(); // Send the module to the deepest sleep
  }
}
```

El setup solo se ejecuta una vez en el programa, en este se abrirá el puerto serie en caso de que *debug* esté en true, también abrirá el puerto serie físico del Arduino para la comunicación con el módulo GPS.

Y por último espera hasta que se inicie correctamente el módulo SigFox.

```
void loop() {
  String position_data;

  position_data = GetGPSpositon();

  if( position_data == "No Signal" ){

    SigFox.begin();
    delay(100);
    int8_t temperatura = (int)SigFox.internalTemperature();
    Serial.print("Temperatura: ");
    Serial.println(temperatura);
    SigFox.status();
    delay(1);
    SigFox.beginPacket();
    SigFox.write(temperatura);
    int resultado = SigFox.endPacket();
    SigFox.end();

  }
}
```

```
        else{
            SendSigfox(position_data);
        }

        if(millis() > 4980000){
            delay(100);
        }
        else{
            Wait(WAITING_TIME, false);
        }
    }
}
```

El `void loop()` es la parte del código que se repite en bucle mientras esté encendido el dispositivo, en este fragmento es donde se llama a las funciones explicadas anteriormente.

Primero se intenta obtener una posición GPS, cuando esta se obtiene, se convierte en un *String* y se envía a SigFox.

En caso de que al intentar obtener una posición GPS no sea posible, entonces, se enviará una temperatura del procesador.

Cuando el dato es enviado se procede a una espera, la duración de esta depende de cuánto tiempo de ejecución lleva el programa.

CALLBACK SIGFOX:

Una vez que hemos registrado el dispositivo en la base de datos y hemos creado nuestro programa, ya podremos acceder al *Backend* de SigFox y ver los mensajes, pero como he comentado con anterioridad, estos mensajes aparecen en hexadecimal y además el entorno no es muy amigable ni intuitivo, por ello se crean los *Callbacks*.

Esto consiste en coger los datos de esta base de datos y enviarlos en otro formato a otra base de datos, a un correo, etc...

Aquí vamos a comentar y documentar dos ejemplos, el primero consiste en enviar la información a través de correo electrónico y el segundo caso consiste en subir los datos a una página pensada para publicar datos del Internet de las cosas.

Para el primer ejemplo, debemos acceder a la ventana principal del *Backend*, una vez allí, seleccionamos en la parte superior el botón “*Devices*”.

Una vez seleccionado, veremos nuestra lista de dispositivos, tal que así:

Communication status	Device type	Group	Id	Last seen	Name	Token state
	Arduino_DevKit_1	POLITECNICO	1D9873	2020-02-28 13:15:35	Arduino_DevKit_1-device	☒

Figura 17 Arduino_DEVKit_1

Pulsaremos ahora sobre “*Device type*”, en mi caso sería sobre “*Arduino_DevKit_1*”, después de pulsar se abrirá una nueva ventana y en la columna lateral izquierda pulsaremos sobre “*Callbacks*”.



Figura 18 Menú Callback

En la ventana principal que se acaba de abrir, podremos ver que no aparece nada, puesto que no se ha definido ningún *callback* todavía, pero en la parte superior derecha podemos encontrar un botón que dice: “*new*”. Pulsaremos sobre él y nos aparecerá esta ventana.

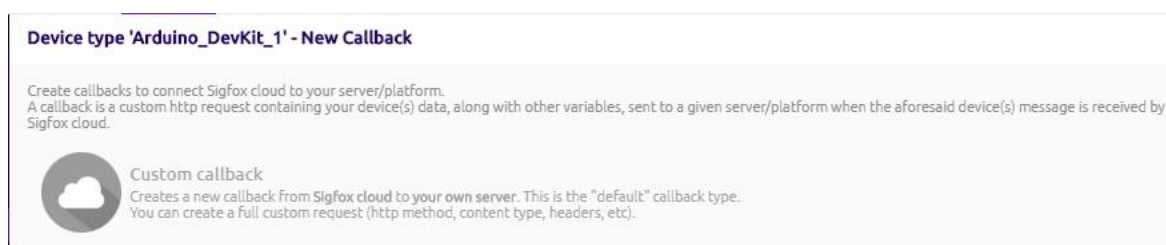


Figura 19 New Callback

Nos aparecerán varias opciones predeterminadas, por ejemplo Microsoft Azure, AWS IoT, IBM Watson, etc...

En caso de disponer de una cuenta en una de estas plataformas, puede ser una gran ventaja, porque estas están optimizadas para el IoT y con unos ajustes mínimos se pueden obtener resultados muy potentes.

Sin embargo, la mayor parte de estas plataformas son de pago y no están al alcance de todo el mundo, por ello se debe recurrir a opciones gratuitas.

En nuestro caso hemos seleccionado “*Custom callback*”, pulsamos sobre esta opción y nos aparecerá esta nueva ventana.

The screenshot shows a 'Callbacks' configuration window. At the top, there are two dropdown menus for 'Type' (set to 'DATA') and 'UPLINK'. Below them is a 'Channel' dropdown set to 'URL'. A 'Send duplicate' checkbox is present. The 'Custom payload config' section is a large text area. Below it, there is a 'URL syntax' field with a placeholder: 'http://host/path?id={device}&time={time}&key1={var1}&key2={var2}...'. Below the URL syntax, there are two sections: 'Available variables: device, time, duplicate, snr, station, data, avgSnr, lat, lng, rssi, seqNumber, deviceTypeId' and 'Custom variables:'. Below these, there are two red text warnings: 'The Feature "send duplicate" and the following information: duplicate, avgSnr, lat and lng will not be available Connectors" articles)' and 'The information RSSI, SNR and station will be available through option. Contact your local Sigfox operator for...'. Below the warnings, there is a 'Url pattern' field. Below that, there is a 'Use HTTP Method' dropdown set to 'GET'. Below that, there is a 'Send SNI' checkbox checked, with a note '(Server Name Indication) for SSL/TLS connections'. At the bottom, there is a 'Headers' section with a table-like structure with columns 'header' and 'value'.

Figura 20 Configuración Callback

En este menú, podremos configurar todo, primero nos dirigimos a “*Channel*” y seleccionamos Email, tras esto, la ventana cambiará y aparecerán algunos cambios.

Lo primero que deberemos rellenar, será el campo de “*Custom Payload Config*”, que se rellenará así:

The screenshot shows the 'Custom payload config' field with the text: 'lat::float:32 lng::float:32 alt::float:32'.

Figura 21 Payload config

Luego debemos seleccionar los correos a los que queremos enviar los mensajes, deberemos escribirlos en el apartado de “*Recipient*”. Para añadir varios correos, tan solo se tienen que escribir uno debajo de otro.

Por último, configuramos los últimos dos apartados añadiendo los siguientes datos:

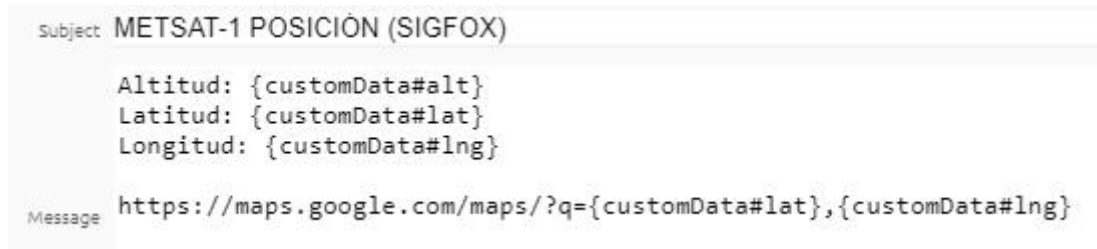


Figura 22 Mensaje E-mail

Con todos los pasos anteriores realizados correctamente, ahora solo tendremos que guardar la configuración y ya nos aparecerá este nuevo *callback* en el menú.

Si ahora conectamos el Arduino MKRFOX 1200 con el programa cargado, y nos envía la posición, entonces recibiremos un correo electrónico como este:

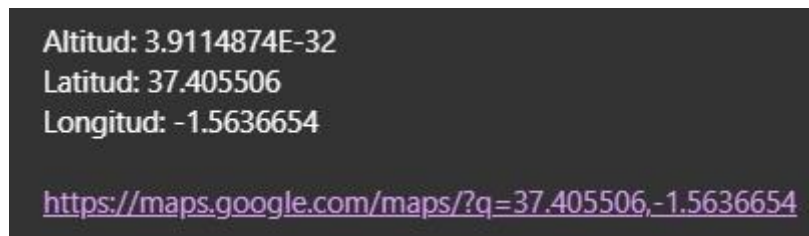


Figura 23 E-mail

En este mensaje podemos ver las coordenadas y la altitud, además tenemos un enlace directo a Google maps para poder posicionarlo de forma rápida.

Ahora vamos a añadir un nuevo *callback* para añadir la temperatura, para ello seguiremos los pasos anteriores y en los campos, usaremos los mismos datos, excepto estos:

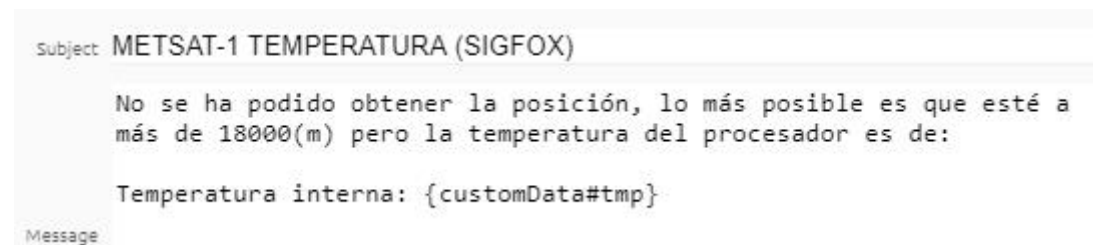


Figura 24 Mensaje temperatura

Ahora también nos informará de la temperatura vía email en caso de no ser capaz de obtener una posición GPS.

Por último, añadiremos dos *callbacks*, pero estos irán a una base de datos que nos permitirá hacer los resultados públicos y visualizarlos con una interfaz gráfica muy útil.

Para realizar este último paso, deberemos acceder a la página web de *ThingSpeak* y crear una cuenta.

<https://thingspeak.com/>

Una vez tengamos creada la cuenta podremos acceder a nuestro perfil y crear un nuevo canal, este canal será el medio en el que se publicarán los datos, el acceso puede ser controlado o se puede hacer público a través de un enlace compartido.

Una vez hecho esto, deberemos copiar la API⁴ key, para ello dentro del menú principal de *Thingspeak* seguiremos estos pasos:

Channels>channel to update> API Keys

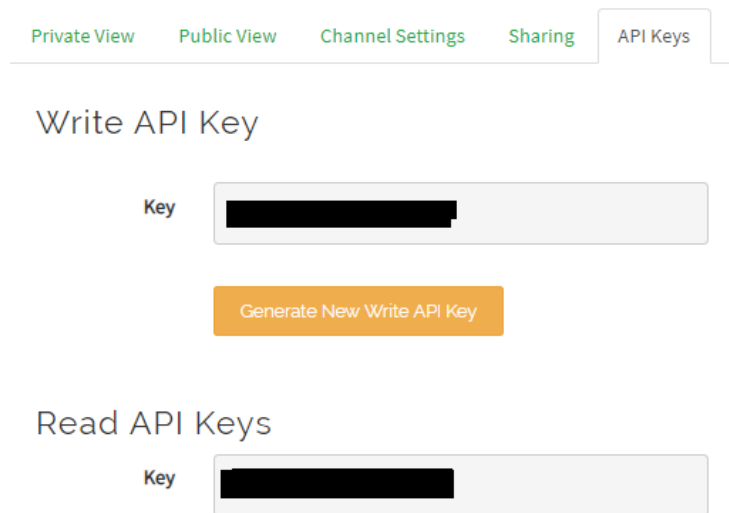


Figura 25 API Key

Una vez hecho esto copiaremos la clave y la guardaremos para usarla después.

Ahora nos dirigiremos al *backend* para crear un nuevo *callback*, este deberá quedar configurado así:

⁴ API: Una API es un conjunto de definiciones y protocolos que se utiliza para desarrollar e integrar el software de las aplicaciones. API significa interfaz de programación de aplicaciones.



Figura 26 URL Callback GPS

Y una vez listos estos campos solo quedarán dos más por configurar, el primero de ellos será el campo de “URL PATTERN”, el cual se rellenará con lo siguiente:

`https://api.thingspeak.com/update.json?api_key=84FAC1E5HZTTQ24K&field1={customData#lat}&field2={customData#lng}&field3={customData#alt}`

Pero las letras marcadas en rojo corresponden con una API key inventada, deberá ser cambiada por la que se nos ha proporcionado anteriormente.

Por último, deberemos seleccionar el método “GET” en el campo “USE HTTP METHOD”.

Con esto ya tenemos listo el *callback* correspondiente a la posición y ya solo queda configurar el ultimo, que corresponde a la temperatura, para ello creamos un nuevo *callback* y ponemos los siguientes datos:

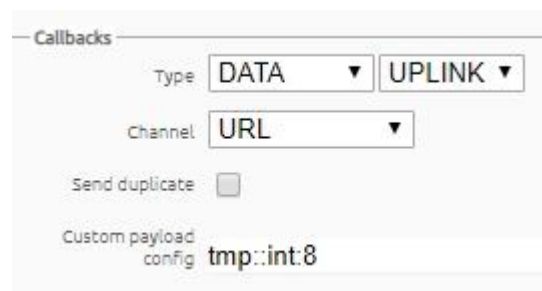


Figura 27 URL Callback temperatura

El “USE HTTP METHOD”, será como antes “GET” y el “URL PATTERN” será:

`https://api.thingspeak.com/update.json?api_key=84ZAC1E9QZTTQ20H&field4={customData#tmp}`

Ahora podemos acceder al canal y modificar como queremos que se muestren los datos, así como el resto de la configuración referente a privacidad y datos adicionales.

Para poder determinar la privacidad de nuestro canal, deberemos acceder primero a este, y una vez en la ventana principal, seleccionamos la opción de “*Sharing*”.

Al acceder a esta opción, encontraremos estas opciones:

Channel Sharing Settings

- ☐ Keep channel view private
- ☒ Share channel view with everyone
- ☐ Share channel view only with the following users:

Email
Address

Enter email here

Figura 28 Privacidad canal

Podremos elegir entre hacerlo público, privado o solo visible para ciertos usuarios de la plataforma.

Una vez configurada la privacidad, solo queda determinar los campos a mostrar y la interfaz gráfica.

Para determinar los campos, debemos acceder a la opción de “*Channel Settings*”.

Una vez dentro de esta opción podemos ver una ventana similar a esta:

Channel Settings

Percentage complete 100%

Channel ID 974006

Name

METSAT_1

Description

Sigue con nosotros el vuelo de nuestra sonda estratosférica y ayúdanos a localizarla.

Field 1

Latitud



Field 2

Longitud



Field 3

Altura



Field 4

Temperatura



Figura 29 Fields

En el apartado de campos, deberemos introducir el nombre del dato, en función de cómo lo hemos establecido en el mensaje del *Callback*. En este caso, el orden sería el de la imagen.

Con los datos seleccionados, solo queda crear la interfaz gráfica, para ello accederemos a la opción de “*Public View*”, en esta ventana podremos crear diferentes gráficos para los datos entrantes, también podemos configurar ciertas opciones de visualización.

Para añadir gráficas, deberemos pulsar sobre “*Add Visualizations*”, mientras que, para otro tipo de representación, deberemos pulsar sobre “*Widgets*”.

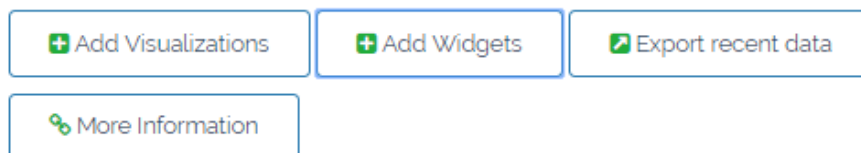


Figura 30 Visualizaciones

Una vez seleccionada una gráfica, sobre ella, aparecerá una barra con opciones, una de estas opciones es la configuración, al pulsar sobre esta opción, aparecerá una ventana como la siguiente donde podremos ajustar la presentación a nuestras necesidades.

Field 1 Chart Options
×

Title:	LATITUD	Timescale:	▼
X-Axis:	TIEMPO	Average:	▼
Y-Axis:	LATITUD	Median:	▼
Color:	#d62020	Sum:	▼
Background:	#ffffff	Rounding:	
Type:	line ▼	Data Min:	
Dynamic?:	true ▼	Data Max:	
Days:	3	Y-Axis Min:	
Results:	140	Y-Axis Max:	

Figura 31 Configuración gráficas

En mi caso, lo he representado en cuatro gráficas de puntos y un indicador un poco más visual para la temperatura.

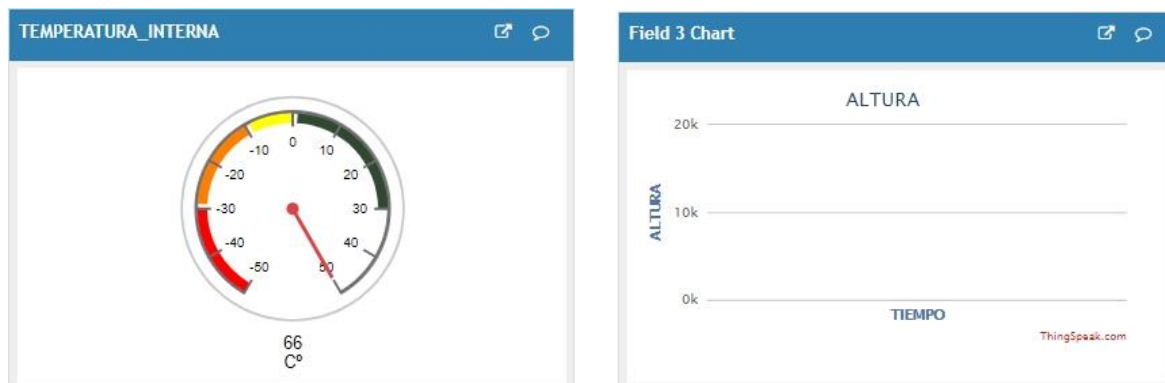


Figura 32 Gráficas temperatura interna y altura

Con toda esta información queda explicado por completo todo el proceso para poder configurar y poner en funcionamiento este sistema de comunicaciones para el globo sonda, por último, solo falta por añadir que normalmente la antena viene de serie con el módulo MKRFOX 1200, pero esta se puede cambiar también por otra con más ganancia, mi recomendación es usar una antena omnidireccional de 868 (Mhz).

En este caso he utilizado la antena que viene de serie con el Arduino, esta es una antena muy simple, ligera y también algo frágil, por lo que debe ir en el interior de la sonda, por este motivo, se debe evitar que la antena toque una parte metálica y tener cuidado de no meter la antena dentro de una jaula de Faraday, de esto hablaremos más adelante.

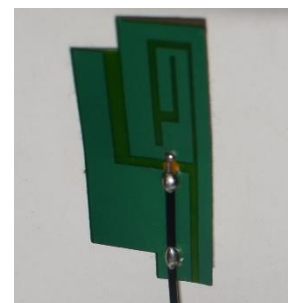


Figura 33 Antena SigFox



Figura 34 Antena SigFox IP67

Otra opción puede ser una antena algo más resistente, con una ganancia 0,2 (dBi) superior, IP67 y que trabaja en un rango de temperaturas entre los 80°C y los -40°C.

Sin embargo, el precio se eleva hasta los 11€, casi diez veces más que la antena usada.

4.1.1.2 GPRS / GSM

Otros sistemas muy fiables y extendidos son los sistemas: *General Packet Radio Service* (GPRS) y *Global System for Mobile Communications* (GSM).

Estos dos sistemas están muy extendidos en España, de hecho, como hemos comentado anteriormente, el servicio SigFox utiliza la infraestructura de las redes móviles para situar sus repetidores, sin embargo, el alcance de estas tecnologías es más limitado. Permiten enviar mucha más información, pero a una altura de 4.000 metros aproximadamente deja de funcionar, por ello este sistema va a ser utilizado como un soporte al anterior, es decir, se complementan.

Si el sistema SigFox falla, este podría seguir estando operativo, además, aunque el sistema SigFox funcione, puede que, por la limitación de mensajes diarios, nos juegue una mala pasada, este sistema evita ese problema, puesto que no existen límites de mensajes.

Para esta aplicación se ha decidido utilizar un GPS *tracker*, en concreto es el modelo TK905 del fabricante TKSTAR, este modelo ha sido escogido por la gran capacidad de su batería, su resistencia y por contar con conexión a redes 2G, 3G y 4G.

El funcionamiento de este dispositivo se basa en un sistema que integra una antena GPS, una antena de telefonía móvil, un pequeño microcontrolador que actúa como cerebro y una ranura de tarjetas SIM.

Una vez que el dispositivo está correctamente configurado, permite obtener la posición de este a través de llamadas o SMS al número de teléfono de la tarjeta SIM que lleva integrado. Además, como tiene acceso a redes 3G y 4G, también permite la conexión de este con un servidor en el que va guardando las posiciones y así se puede hacer un seguimiento en tiempo real.

Esta última característica es fundamental tanto en los primeros minutos de vuelo como en los últimos minutos antes del aterrizaje, puesto que se puede seguir en tiempo real su movimiento.

En cuanto a sus principales características técnicas cabe destacar que puede trabajar en la banda de 850/900/1.800/1.900 (Mhz), la sensibilidad del GPS es de -159 (dBm) con una precisión de 5 metros, puede trabajar hasta un mínimo de -40°C aunque lo recomendable es que no sea menor de -20°C.



Figura 35 GPS tracker TKSTAR

Este dispositivo es comercial y debido a que va a realizar una misión en condiciones extremas y muy cercano a otros dispositivos electrónicos, debe ser modificado en algunos aspectos.

Primero de todo deberemos abrir la carcasa negra para retirar unos potentes imanes de neodimio, puesto que podrían crear problemas en la electrónica y las antenas de otros dispositivos, además estos imanes son también bastante pesados.

La otra modificación que se debe realizar es la de reemplazar la batería por una especial para bajas temperaturas, pero esto se detallará en profundidad más adelante.

Una vez tenemos el GPS *tracker* y hemos realizado la modificación de los imanes, el siguiente paso consiste en adquirir una tarjeta SIM compatible con el dispositivo.

Muchos fabricantes venden tarjetas SIM de muy bajo precio enfocadas al IoT, pero estas no son compatibles con este dispositivo, por ello deberemos comprar una que especifique que es compatible con las redes 2G, 3G y 4G.

Un punto muy importante a tener en cuenta es el lugar del aterrizaje, puesto que debemos buscar la compañía telefónica que ofrece la mejor cobertura en la zona donde queremos que aterrice, en este caso queremos que sea en Castilla la Mancha. Por esto he decidido adquirir una tarjeta de Movistar, ya que esta compañía es la que presenta la mejor cobertura en esta comunidad.

Si tenemos muy claro el lugar de aterrizaje y tenemos dudas sobre la cobertura de una determinada compañía, podemos acceder a la página web de esta y buscar los mapas de cobertura, de esta forma podemos salir de dudas y elegir la compañía que mejor se adapte a nuestra aplicación.

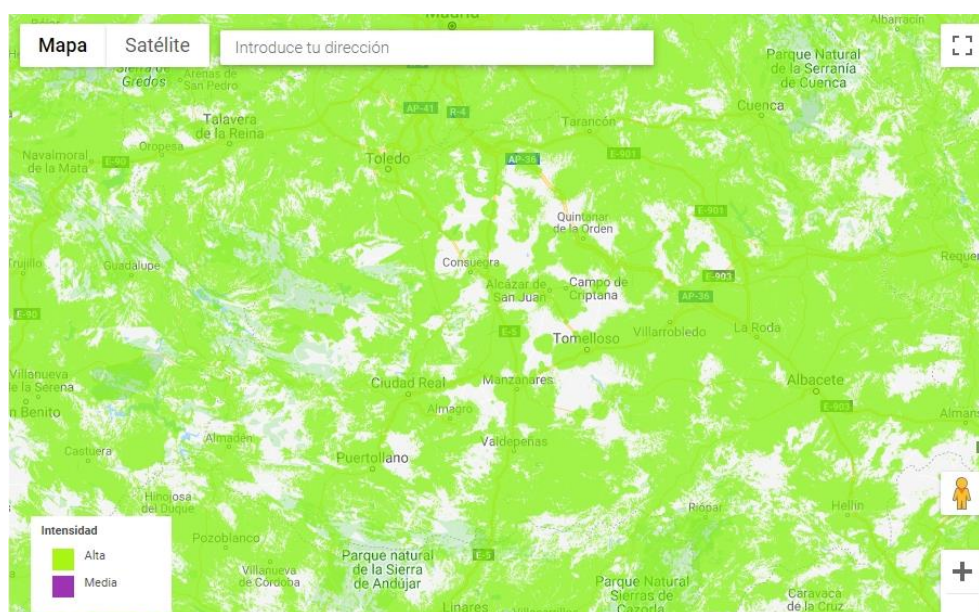


Figura 36 Cobertura Movistar

Una vez que tenemos la tarjeta SIM, deberemos activarla, para ello será necesario hacerlo físicamente y presentando el DNI, lo más cómodo es comprar la tarjeta en una tienda física y realizar este trámite con la compra. También se puede comprar online, pero se tiene que enviar una copia del DNI al correo de la empresa que vende la tarjeta SIM y esto tarda un poco más en hacerse efectivo.

Cuando registremos la tarjeta, deberemos introducirla en un teléfono móvil y hacer una llamada para que esta quede activada.

Normalmente la tarjeta suele venir con saldo disponible, en caso de que esto no sea así, se podrán hacer recargas o contratar una tarifa a través de la página web del operador.

Después de activar y registrar la tarjeta SIM, ya podremos introducirla en el GPS *tracker*. Una vez insertada, el dispositivo se encenderá automáticamente y podremos configurarlo.

CONFIGURACIÓN INICIAL:

El primer paso para realizar la configuración consiste en guardar el número de teléfono del GPS *tracker* en nuestro teléfono, este número normalmente viene escrito en la caja de la tarjeta SIM o bien, cuando la tarjeta es activada, se envía un SMS al teléfono indicando el número propio, por lo tanto, si no lo encuentras, podrás encontrar un SMS en tu teléfono con este número.

Una vez guardado el teléfono, buscamos este en nuestra lista de contactos y abrimos la opción de enviarle un SMS.

Ahora procederemos a determinar el teléfono del administrador, para ello mandamos el siguiente SMS:

Admin123456 “teléfono administrador”

Por ejemplo:

Admin123456 653386578

Le damos a enviar y recibiremos una respuesta de “*admin OK*” si todo ha funcionado bien. Ahora si realizamos una llamada al teléfono del GPS *tracker*, obtendremos un SMS con la posición y un enlace a Google maps, también permite realizar la misma acción a través del envío de un SMS, este SMS se debe enviar también al chat del GPS *tracker* y debe ser así:

G123456#

La respuesta será la misma que realizando una llamada.

A continuación, debemos establecer la franja horaria, para ello debemos enviar el siguiente SMS:

Timezone123456 “zona horaria”

Donde la zona horaria es de +1 en invierno y +2 en verano.

Así por ejemplo en invierno enviaríamos el siguiente SMS:

Timezone123456 +1

Además de estos comandos, también existen los siguientes:

Tabla 5 Comandos opcionales

ACCIÓN	COMANDO	RESPUESTA
Restaurar configuración de fábrica	Begin123456	Empezar ok
Cancelar el número del administrador	Noadmin123456	Noadmin ok
Alarma de movimiento abierta	Shock123456	Shock ok
Cerrar la alarma de movimiento	Noshock123456	Noshock ok
Establecer el modo durmiendo	Sleep123456time	Tiempo de esperaok
Establecer el modo de reposo por sensor de movimiento	Sleep123456shock	Sensor espera ok
Cerrar modo durmiendo	Sleep123456off	Modo espera apagado ok
Activar GPRS	Gprs123456	Gprs ok
Parámetro de consulta	Check123456	Apn,ip,gps,gsm data

EXPLICACIÓN SMS:

Cuando enviamos un SMS o hacemos una llamada para obtener la posición, existen dos posibilidades, la primera de ellas es de que exista cobertura GPS o bien la segunda opción es que no exista esa cobertura, en función de esto el SMS que obtendremos dará unos datos u otros.

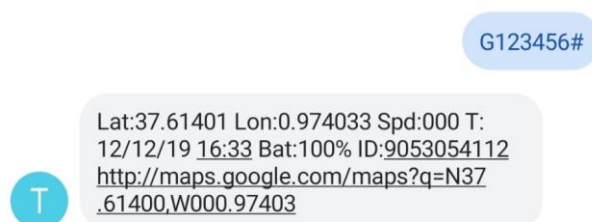


Figura 37 SMS GPS tracker

En esta imagen existe cobertura GPS y por lo tanto los datos que nos da son la latitud y la longitud en formato decimal, la velocidad en (Km/h), la fecha, la hora, la batería, el ID y un enlace para poder ver la posición en Google maps.

En caso de que no exista cobertura GPS, el mensaje será un poco diferente, lo primero que nos mostrará será una posición estimada teniendo en cuenta las últimas posiciones conocidas, luego nos dará fecha, hora, batería, ID y la última posición GPS conocida.

AÑADIR MÁS USUARIOS:

Si se desea que más gente pueda recibir la posición del dispositivo es necesario añadir otros usuarios, esto es posible pero la programación tiene un pequeño error, ya que, para poder añadir a un nuevo usuario, primero se tiene que dar al nuevo usuario privilegios de super usuario y posteriormente este tiene que volver a ceder los privilegios al usuario inicial para que este pueda añadir a más usuarios, así pues, paso a explicar cómo llevar a cabo esta tarea.

Lo primero que debemos hacer es dar de alta a un nuevo usuario mediante el siguiente comando:

Admin123456 “nº segundo usuario”

Una vez que ese SMS es enviado, el segundo usuario pasará a tener privilegios de super usuario y el usuario inicial será solamente usuario.

Para recuperar los privilegios de super usuario, el segundo usuario deberá enviar el mismo SMS, pero esta vez con el número de teléfono del usuario inicial, es decir:

Admin123456 “nº usuario inicial”

Ahora ya tendríamos a dos usuarios dados de alta, para añadir más usuarios debemos seguir este proceso tantas veces como usuarios queramos añadir.

Por último, es importante especificar que las diferencias entre un usuario común y un super usuario, son las siguientes: El super usuario es el único que recibe SMS a través de sus llamadas y es el único que puede añadir usuarios, así como cambiar parámetros fundamentales como la hora.

Por ello es importante tener claro quién es el super usuario para evitar problemas y fallos de funcionamiento del sistema.

ACCESO A LA APLICACIÓN DE ANDROID:

Como se ha especificado anteriormente, este dispositivo también es capaz de conectarse a internet y subir su posición en tiempo real a través de un servidor, este es accesible a través de la *web* y a través de una aplicación para el teléfono.

Esta aplicación está disponible tanto para Android como para IOS, el nombre de esta es TKSTAR GPS, la descargamos y accedemos a ella.

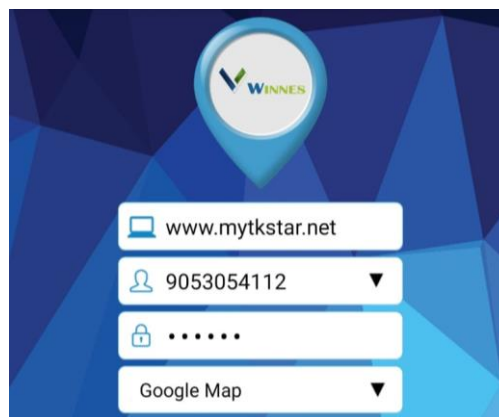


Figura 38 Acceso APP TKSTAR

Lo primero que debemos hacer para entrar es introducir el IMEI o el ID y luego la contraseña.

El IMEI va escrito en una pegatina en la parte trasera del GPS *tracker*, aunque también se puede acceder mediante el ID que es enviado junto con las posiciones GPS.

Por otro lado, la contraseña viene predeterminada y es 123456, una vez que accedamos deberemos cambiar la contraseña por seguridad.

Una vez accedamos a la aplicación, aparecerá un menú como este:

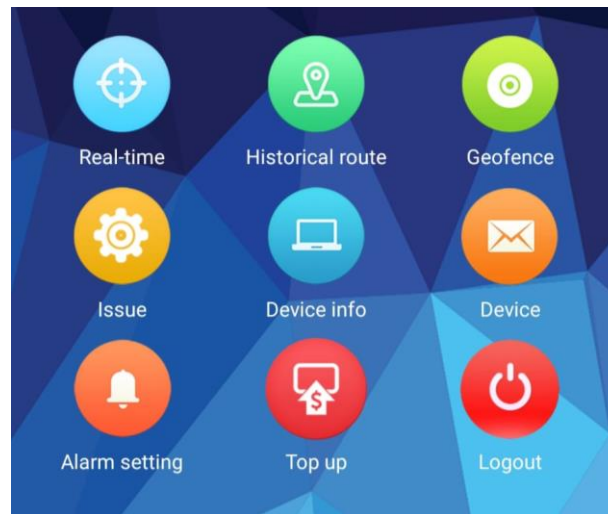


Figura 39 Menú APP

Si pulsamos sobre “*Real-time*” podremos ver la posición en directo o bien la última posición conocida.

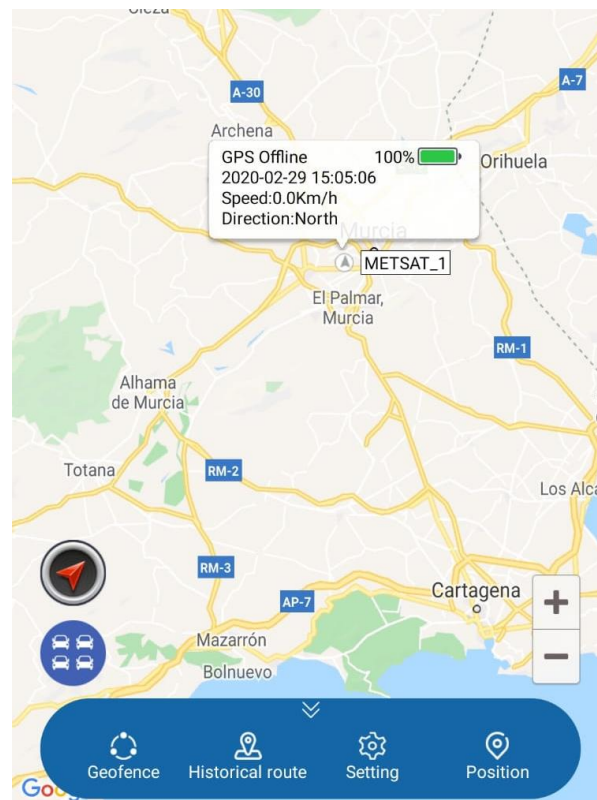


Figura 40 Posición METSAT-1

Si en el menú principal pulsamos sobre “*Issue*”, nos aparecerá un menú con varias opciones sobre alarmas, por ejemplo, podremos configurar alarmas de velocidad, de batería, etc...

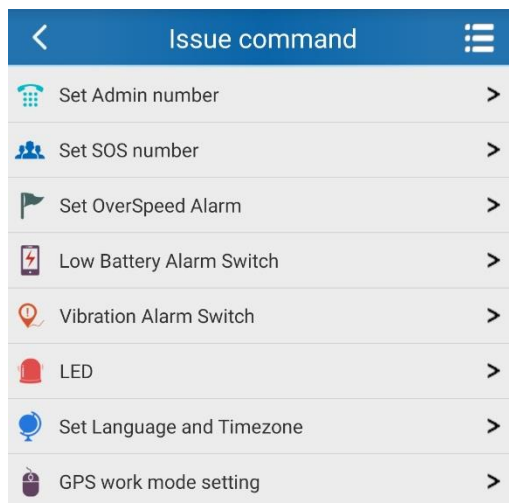


Figura 41 Comandos TKSTAR

Para poner un ejemplo, si pulsamos sobre “*Set OverSpeed Alarm*”, podremos determinar una velocidad en (Km/h) a la cual, si el dispositivo supera esta velocidad, la aplicación nos avisará.

Por último, si volvemos al menú principal de la aplicación y deslizamos con el dedo hacia un lado, nos aparecerá este pequeño menú:

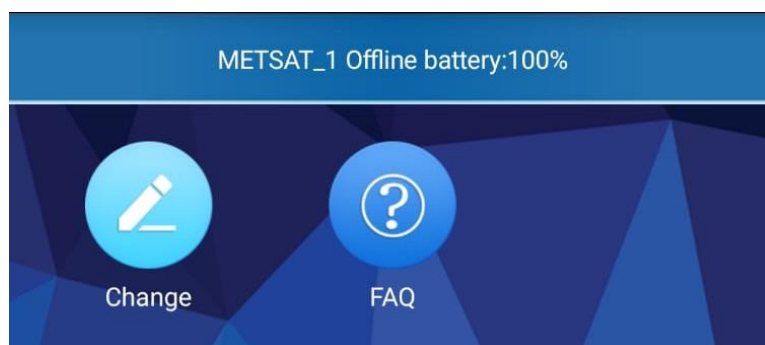


Figura 42 Cambio contraseña

Si seleccionamos la opción *Change*, se abrirá una ventana en la que podremos cambiar la contraseña de acceso, para ello deberemos volver a escribir la contraseña anterior, es decir, 123456 y después escribir dos veces la nueva contraseña.

ACCESO PÁGINA WEB:

Si no disponemos de teléfono o preferimos utilizar un PC para realizar el seguimiento, también podremos hacerlo a través de la página web:

<http://www.mytkstar.net/>



Figura 43 Acceso MyTKSTAR

Al igual que en la APP, deberemos introducir el IMEI o ID junto con la contraseña para poder acceder a la plataforma.

Una vez que accedamos, iremos directos a una ventana principal desde la que se hace el seguimiento en tiempo real, pero también nos ofrece más opciones que vamos a ir viendo a continuación.

La primera de ellas es la posibilidad de descargar todo el recorrido del dispositivo en un archivo que se puede abrir directamente en Google Earth, para ello en la ventana principal, debemos buscar nuestro dispositivo en el mapa y si pinchamos sobre él nos aparecerá una ventana emergente como esta:



Figura 44 Información METSAT-1

Esta ventana nos indica sobre los parámetros en tiempo real, pero en la parte inferior nos ofrece varias opciones, si pulsamos sobre “more”, nos aparecerá la

opción de “*Tracking Report*”. Pulsamos sobre esta opción y nos aparecerá la siguiente ventana:

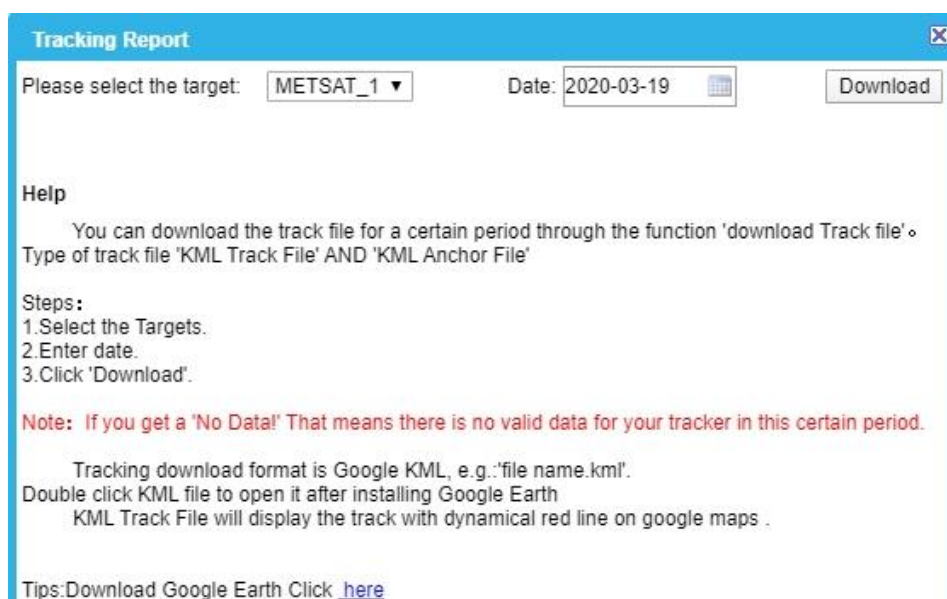


Figura 45 Tracking Report METSAT-1

En la parte superior podemos elegir que dispositivo queremos usar, en mi caso solo tengo uno y lo he llamado METSAT_1, justo al lado podemos elegir el día que nos interesa para descargar datos de desplazamiento.

Una vez elegidos los parámetros, pulsamos el botón “*Download*” en la esquina superior derecha.

A continuación, se descargará un fichero de tipo *.kml*. Si pulsamos sobre él, se abrirá Google Earth y podremos ver sobre el mapa por donde se ha desplazado el GPS *tracker*.



Figura 46 Track TKSTAR

Otra opción muy importante que nos permite esta página web es la de crear *geofences*, estas vienen a ser áreas circulares situadas en un lugar concreto y con un radio determinado.

Estas se sitúan en puntos estratégicos y si nuestro dispositivo entra en una de estas zonas o sale, entonces nos aparecerá una alarma y quedará registrado en la base de datos.

Para acceder a esta opción nos dirigimos al menú principal y en el lateral izquierdo aparece una columna con nuestros dispositivos, en mi caso sería esto:



Figura 47 Menú geofences

Si pulsamos sobre “*more*”, aparecerá una ventana con varias opciones, una de ellas es la opción de “*geofences*”. Pulsamos sobre esta opción y nos aparecerá una nueva pestaña en el navegador.

En esta nueva pestaña nos encontraremos un mapa y en la parte lateral izquierda una columna como la siguiente:

No.	Name	R(m)	Operate
1	CASA	100	Delete
2	QUINTANAR CENTRO	774	Delete
3	QUINTANAR CIRCINVALACION	2147	Delete
4	LILLO	1567	Delete

Figura 48 Geofences

En este caso yo ya tengo varias *geofences* creadas, pero la primera vez que se accede, esta columna está en blanco y si queremos crear una nueva *geofence*, debemos pulsar sobre el botón “*Add*”.

Cuando pulsemos sobre este botón, nos aparecerá algo similar a esto sobre el mapa.



Figura 49 Configuración geofence

Esta va a ser la *geofence*, para configurarla correctamente, primero le damos un nombre, luego un radio y por último podemos dar algún dato que nos parezca importante.

Para desplazar esta sobre el mapa, debemos de pulsar sobre el centro de la circunferencia y sin soltar, arrastrar esta hasta el punto deseado, para modificar el radio, también podemos pulsar sobre el punto blanco de su perímetro y arrastrar hacia dentro o fuera.

Finalmente guardamos los cambios pulsando sobre “Save”.

Ahora esta *geofence* debe aparecer en la barra lateral, la cual también puede ser borrada desde esa misma columna mediante el botón “Delete”.

Para ver las alarmas de los *geofences* creados, tendremos que volver al menú principal, para ello cerramos la pestaña de los *geofences* y volvemos a la pestaña que corresponde a la ventana principal del programa, una vez allí, podremos ver una barra principal con dos opciones:



Figura 50 Banner TKSTAR

Estamos en la opción monitor, pero deberemos pulsar en “Statistics” si queremos ver las alarmas.

Pulsamos en esta opción y se abrirá un nuevo menú con una barra de opciones en la parte izquierda, la cual nos da estas opciones:



Figura 51 Menú estadísticas

Seleccionamos “*Alarm Overview*” y luego nos preguntará por las fechas, las introducimos y veremos las alarmas.

METSAT_1	9053054112	Geo-fence Out(YEBENES)	2020/02/28 12:47
METSAT_1	9053054112	Geo-fence Out(CHUECA)	2020/02/28 12:46
METSAT_1	9053054112	Geo-fence In(HINOJOSOS)	2020/02/28 12:46
METSAT_1	9053054112	Geo-fence In(CHUECA)	2020/02/28 11:04
METSAT_1	9053054112	Geo-fence Out(CHUECA)	2020/02/28 11:02
METSAT_1	9053054112	Geo-fence In(YEBENES)	2020/02/28 10:04
METSAT_1	9053054112	Geo-fence In(CHUECA)	2020/02/28 10:03

Figura 52 Alarmas

Además de las alarmas de *geofences*, también nos aparecerán las de velocidad y demás alarmas establecidas por el usuario.

Con todos los datos mostrados en este apartado queda todo explicado para poner en marcha este dispositivo, por ello procedo a explicar el último sistema de comunicaciones.

4.1.1.3 WSPR

Debido a que hemos utilizado dos sistemas muy prometedores y precisos en las primeras dos capas de comunicaciones, hemos decidido utilizar un sistema un poco más experimental en la última capa para comprobar su funcionamiento y su potencia.

Este sistema se denomina *Weak Signal Propagation Reporter* (WSPR) o WHISPER, que en español significa: Reporte de propagación de Señales débiles.

Este sistema fue inventado por Joe Taylor⁵, consiste en un pequeño equipo de radio que emite una señal en 4-FSK, esto significa que el mensaje se emite codificado en cuatro tonos que distan entre si 1,4 (Hz), además el ancho de banda que ocupa es de 1,4 (Hz) por lo que, con muy poca potencia de emisión, se pueden alcanzar muy largas distancias siempre que exista una buena propagación.

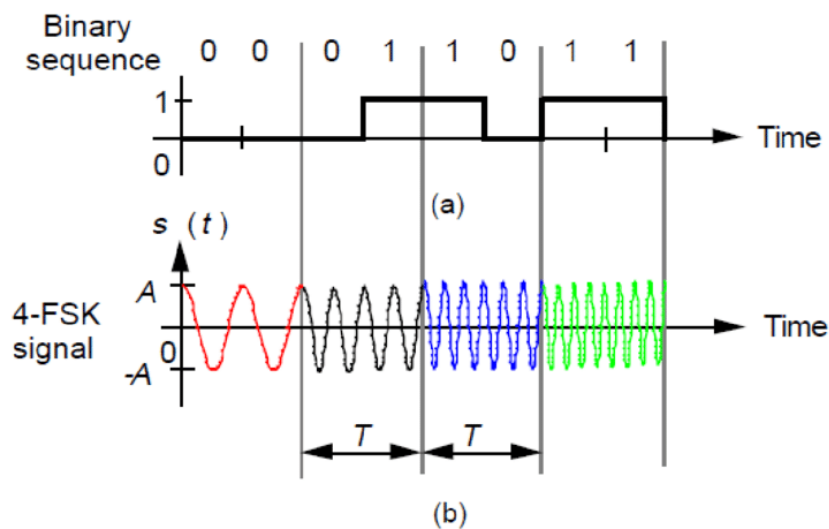


Figura 53 Modulación 4-FSK

Para entender como con muy poca potencia podemos llegar tan lejos, vamos a realizar un pequeño cálculo, para ello vamos a suponer que estamos emitiendo con un equipo de 30 (W) y emite una señal que ocupa un ancho de banda de 2.400 (Hz).

Si dividimos 30 (W) entre 2.400 (Hz)...

⁵ Joe Taylor: (29 de marzo, de 1941) es un astrofísico estadounidense, ganador del Premio Nobel de Física en 1993

$$\frac{30(w)}{2.400(Hz)} = 0,0125(w/Hz)$$

Es decir, necesitamos 0,0125 (w) de potencia por cada Hz que ocupamos de ancho de banda.

Por otro lado, vamos a hacer el mismo cálculo para un sistema de radio WSPR, para ello vamos a emitir en 5 (W) y vamos a ocupar un ancho de banda de 1,4 (Hz).

Realizando el mismo calculo que antes...

$$\frac{5(w)}{1,4(Hz)} = 3,57142(w/Hz)$$

Es decir, necesitamos 3,57(w) de potencia por cada Hz que ocupamos de ancho de banda.

Si ahora realizamos la siguiente operación, veremos con que potencia deberíamos emitir en el primer ejemplo para igualar la relación de (W/Hz) en WSPR.

$$3,57142(w/Hz) * 2.400(Hz) = 8.571,4(w)$$

Es decir, para tener la misma relación de (W/Hz) de WSPR, en el primer ejemplo deberíamos emitir a una potencia de 8.571,4 (W), es decir 8,5 (Kw).

Además de esta relación, que nos ayuda a entender mejor como funciona este sistema, también debemos destacar que, al ser un sistema digital, también posee un sistema de corrección de errores.

Gracias a todo esto, podemos decodificar señales imperceptibles en el ruido, incluso por debajo de los -30 (dB), esto es algo muy importante, puesto que por ejemplo el morse se deja de escuchar a los -10 (dB) y la modulación PSK sobre los -15 (dB).

El inconveniente de este sistema es que funciona como un sistema de radiobaliza y el mensaje que envía está predeterminado.

Este mensaje es el indicativo de llamada, nuestro *locator* y la potencia con la que emitimos.

Este mensaje se emite en dos minutos, siempre en sincronización con la hora mundial, después de la emisión, se queda un tiempo en *standby* y luego vuelve a emitir otros dos minutos.

Para entender que significa el mensaje que se envía, tendremos que aclarar primero un par de tecnicismos, el primero de ellos es el indicativo de llamada.

Para entender que es esto, hay que tener en cuenta que el espectro electromagnético está totalmente reglado y cada país tiene una serie de leyes que determinan como debe ser su división, establece frecuencias de libre uso, otras prohibidas, restringidas para determinados usos, etc...

A nivel internacional existe la Unión Internacional de Telecomunicaciones (UIT), la cual dicta las reglas mínimas que deben cumplir los países que ratifican esta organización, así como los diferentes convenios que se derivan de esta unión.

El estado español debe por tanto cumplir con esta normativa internacional, pero se reserva el derecho a ampliar estas leyes, pero siempre con un carácter constructivo o con intención de hacer la norma más eficaz, por ello España ha determinado una serie de frecuencias que no son de uso libre y para operar en ellas se debe tener una autorización, la cual se obtiene a través de la Unión de Radioaficionados españoles (URE), para obtener esta autorización, se requerirá la superación de una prueba que constará de dos partes independientes que versarán, respectivamente, sobre las siguientes materias:

- Conocimientos suficientes de electricidad y radioelectricidad para operar una estación de radioaficionado.
- Dominio de la normativa reglamentaria referente a las estaciones de radioaficionado.

Tras haber superado el examen, la Administración emitirá el certificado de examen armonizado (HAREC) y se podrá solicitar la autorización de radioaficionado, que consiste en la concesión de un indicativo o distintivo (EA4URE, por ejemplo), con el que se efectuarán las emisiones de radioaficionado y nos ayudan a identificar a quien está emitiendo.

Para emitir en WSPR, existen varias frecuencias que pueden ser usadas, pero estas pertenecen a un rango que no es accesible para un usuario no autorizado, por ello la asociación de Radio Aficionados de Cartagena nos ha asesorado y bajo su supervisión, nos han cedido para el lanzamiento y las pruebas su indicativo de llamada, en este caso sería el EA5URD.

Este código, sería el encabezamiento del mensaje, pero aún queda por explicar que es el *locator*.

Para ello debemos hablar del *Mainhead Locator System*, este es un sistema de coordenadas geográficas para radioaficionados, este sistema es capaz de comprimir latitud y longitud en una pequeña cadena de caracteres.

Una cadena podría ser, por ejemplo: BL11bh16.

En cada par de caracteres se codifica longitud y latitud, la idea es que el mundo se divide en diferentes cuadrados, estos a su vez se dividen en otros más y estos más pequeños en otros cuantos más, de esta forma con los siguientes pares de letras, se puede ir afinando un poco más la posición.



Figura 54 GRID baja resolución



Figura 55 GRID alta resolución

El problema es que, aunque seamos capaces de obtener una posición, esta no es muy precisa, sin embargo, puede ser muy útil para globos sonda que vuelan largas distancias, puesto que este sistema sería muy útil para ver cómo se va moviendo sobre el océano o sobre un continente completo.

Para nuestra aplicación, también nos será de utilidad puesto que, si todo falla, por lo menos nos da una probabilidad de encontrarlo, además nos permite probar este equipo de radio tan interesante.

Entendidos estos dos conceptos, podemos continuar con la explicación del sistema.

Una vez enviado el mensaje, tenemos dos opciones, la primera de ellas es disponer de una estación para recibir la señal y la segunda opción consiste en una serie de estaciones fijas que están repartidas por todo el mundo y cuando estas reciben una señal, envían los datos a una base de datos pública.

Debido a que la frecuencia que vamos a utilizar va a ser de 14 (Mhz), vamos a utilizar por comodidad la segunda opción, aunque para pruebas antes del lanzamiento, también usaremos la primera opción.

Para entender porque vamos a utilizar la segunda opción, vamos a aplicar la siguiente fórmula.

$$\lambda = \frac{c}{f}$$

Donde λ es la longitud de onda, f es la frecuencia de la onda y c es la velocidad de la luz.

Así pues, obtenemos lo siguiente:

$$\lambda = \frac{300.000.000(\frac{m}{s})}{14.000.000(Hz)}$$

$$\lambda = 21,42(m)$$

Es decir, la longitud de onda es de 21,42 (m), por esta razón, realizar una antena propia para llevarla con nosotros puede ser un problema para desplegarla en polarización vertical, se podría hacer en polarización horizontal, pero se pierde mucha señal.

Otra opción sería fabricar una antena de $\frac{\lambda}{2}$ o $\frac{\lambda}{4}$, es decir, de 10,71 (m) o de 5,35 (m). Pero sigue siendo bastante complicado transportarla y también presenta otra serie de problemas que veremos más adelante.

Por esta opción usaremos la infraestructura de terceros para recibir las señales, además la URE-Cartagena estará a la escucha con su antena de 20 metros.

Una vez explicado el funcionamiento del sistema y los conceptos previos, procedo a exponer que dispositivo he usado, así como su puesta en marcha y como se opera con él.

El dispositivo utilizado es el modelo WSPR-TX MINI del fabricante Zachtek, este se vende por separado del filtro paso bajo, por ello antes de comprar la placa debemos tener claro en qué frecuencia queremos emitir y comprar también un filtro paso bajo pasivo para esta frecuencia, en este caso la frecuencia va a ser de 14 (Mhz)

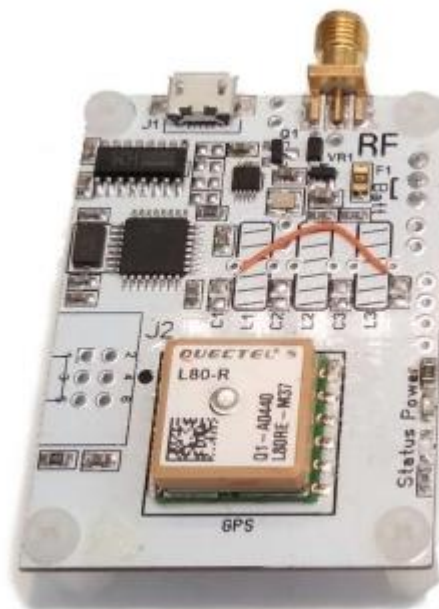


Figura 56 WSPR sin filtro paso bajo

Cuando tengamos la placa, lo primero que tenemos que hacer es montar el filtro paso bajo, para ello el fabricante pone a nuestra disposición un manual muy detallado de cómo hacerlo, así como pautas para hacer las bobinas del filtro.

Los componentes para montar son 4 condensadores SMD y tres bobinas montadas sobre una estructura toroidal.

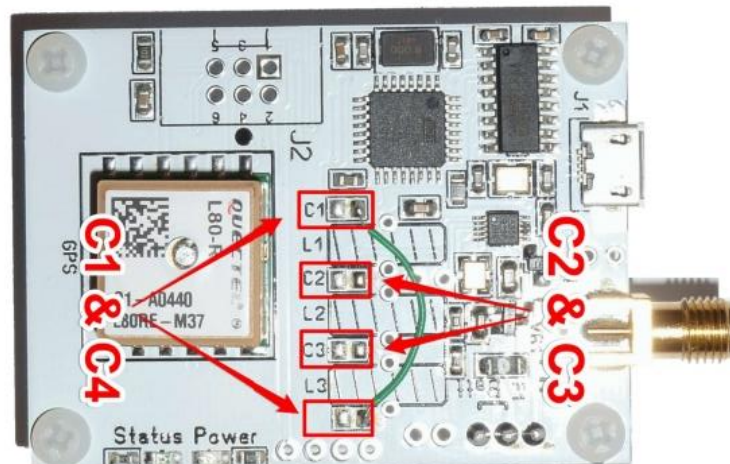


Figura 57 Posición condensadores SMD

Los condensadores C1 y C4 son de 180 (pF), mientras que los C2 y C3 son de 390 (pF)

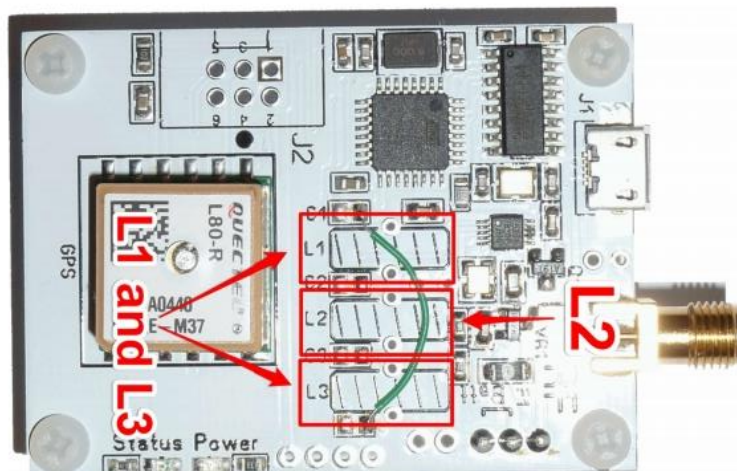


Figura 58 Posición bobinas toroidales

Las bobinas L1 y L3 son de 773 (nH), mientras que la bobina L2 es de 904 (nH).

Para fabricar las bobinas hay que seguir con atención estas indicaciones porque de otra forma, no funcionará el filtro.

Las bobinas L1 y L3, necesitan 16 vueltas, mientras que la bobina L2 necesita 17 vueltas alrededor del toroide.

La longitud del hilo antes de enrollar las bobinas L1 y L3 debe ser de 25 (cm), mientras que para la bobina L2, debe ser de 26 (cm).

El núcleo de todos los toroides será de material metálico.

Para hacer las vueltas, da igual en qué dirección se hagan, siempre que estas sigan una misma dirección, las vueltas deben ir repartidas uniformemente alrededor del toroide, es decir, la distancia entre cada espira debe ser la misma.

Uno de los mayores problemas a la hora de fabricar estas bobinas, es la forma de contar las vueltas, por ello para empezar a contar las vueltas, iremos siguiendo estos pasos.

En la primera imagen, se cuenta una vuelta, mientras que, en la segunda, se cuentan dos vueltas, se debe seguir el mismo sentido hasta completarlas todas.

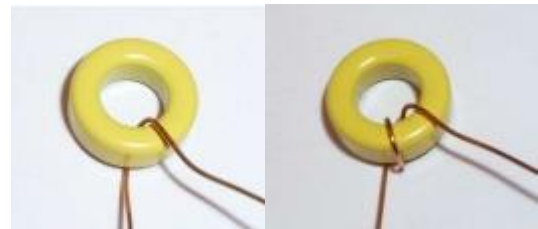


Figura 59 Primera y segunda vuelta

Finalmente, el resultado debe ser similar a este:



Figura 60 Bobina completa

Una vez tengamos las bobinas terminadas, deberemos quitar el barniz del cobre para poder soldarlo en la placa y que tenga continuidad.

Cuando tengamos todos los componentes soldados, debemos hacer una prueba para comprobar que funciona correctamente nuestro filtro.

Para hacer esta prueba, debemos acceder a la página web del fabricante y deberemos descargar un programa gratuito para la configuración de la radiobaliza WSPR.

Descargamos el programa y lo abrimos, a continuación, deberemos conectar la placa vía USB a nuestro PC, en el programa seleccionamos el COM de nuestra placa y seleccionamos el modo generador de onda.

Lo ajustamos a 14 (Mhz) y conectamos el osciloscopio a la salida de la antena para ver la señal.

Una vez que tenemos conectado el osciloscopio, le damos al botón “start” y si todo va bien, comenzaremos a ver la señal en el osciloscopio.

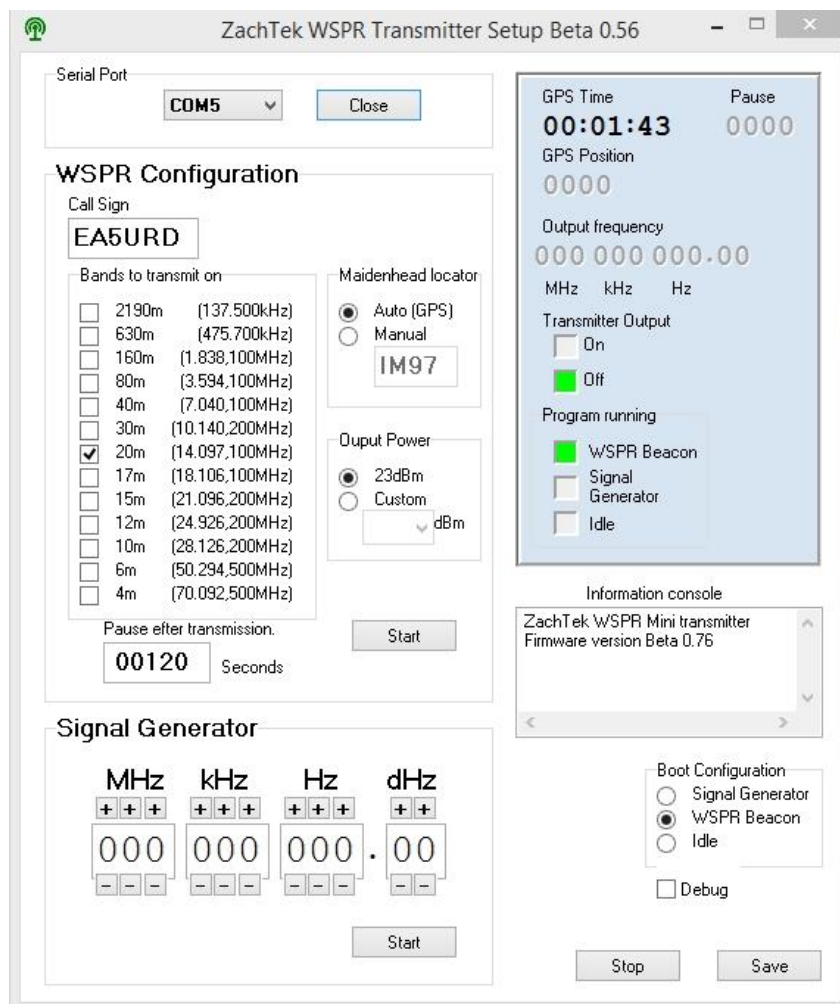


Figura 61 Configuración WSPR

La señal debe ser de unos 14 (Mhz).

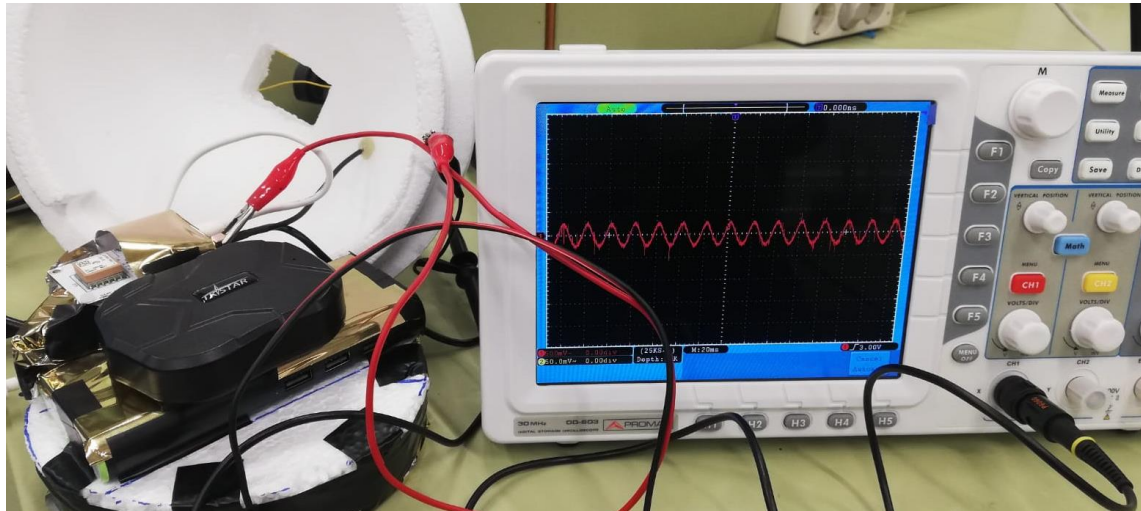


Figura 62 Señal WSPR en osciloscopio

Una vez comprobado que el filtro funciona bien, pasamos a hacer la comprobación del funcionamiento en modo radiobaliza.

Para ello, configuraremos el programa como en la imagen anterior, en la que aparece el menú del WSPR, cabe destacar que el campo relativo a la espera entre emisiones, en el ejemplo está configurada en 120 segundos, pero puede ser modificado a la necesidad del usuario.

Existe también una versión alternativa del programa anterior, la versión 0.56 es la mostrada anteriormente y está en *Beta*, la otra opción es la versión 0.82, la cual es una versión estable, pero poco utilizada por ser muy reciente.

Para los usuarios de la versión 0.82, la configuración será la siguiente:

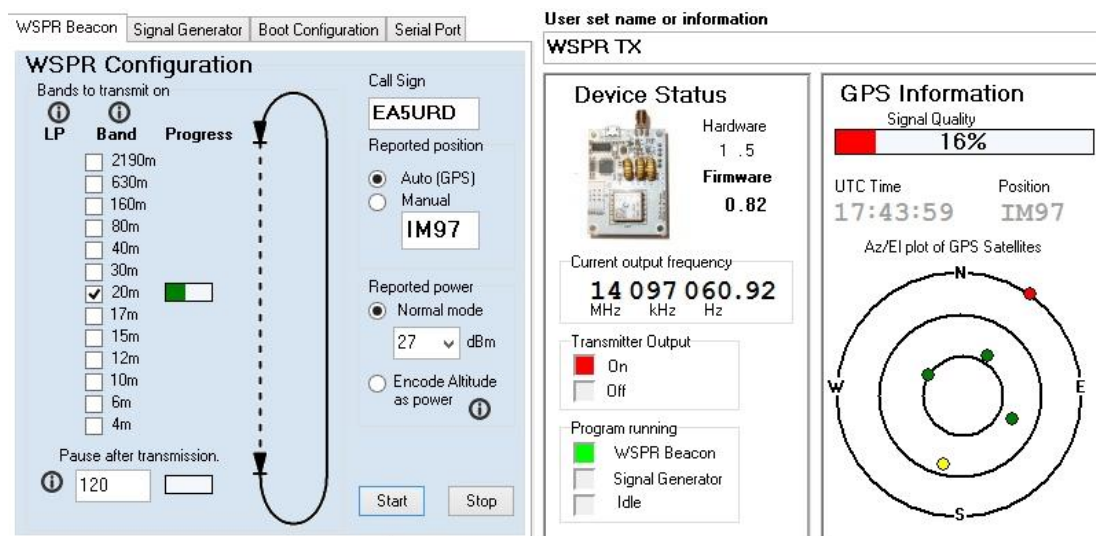
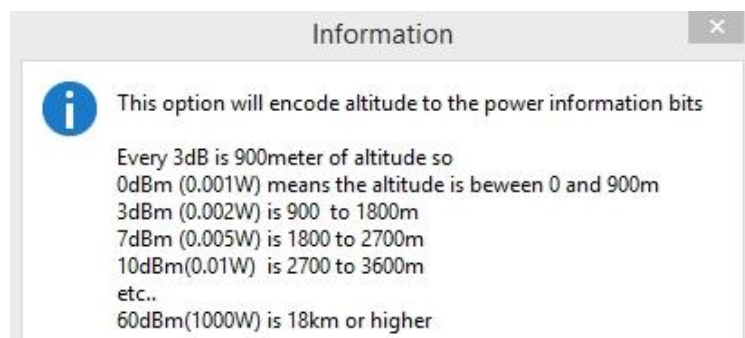


Figura 63 WSPR optimizado

En las pestañas de la parte superior, debemos seleccionar la opción de “WSPR Beacon”, luego introducimos el “Call Sign”, la posición GPS la establecemos como automática y el campo de “Reported Power”, lo podemos establecer como modo normal o como “Encoder altitude as power”, yo recomiendo usar esta última opción.



Codificación de la altura

Esta opción nos permitirá poder estimar la altura en función de la potencia emitida, siguiendo la regla establecida en la imagen anterior.

Finalmente, seleccionamos el tiempo de espera entre transmisiones, guardamos la configuración y pulsamos sobre “Start”.

Tanto para la versión 0.82 como para la 0.56, una vez se selecciona una configuración y se guarda, cuando se alimente de nuevo el dispositivo, este funcionará con su última configuración guardada, independientemente de que se alimente con batería o a través de un USB y conectado al programa, en resumen, no es necesario el uso del PC para el funcionamiento del dispositivo, solo para configurarlo.

Cuando empieza a emitir, podemos ver en el dominio de la frecuencia una portadora en 14 (Mhz) y varios armónicos producidos por los cuatro tonos de FSK.

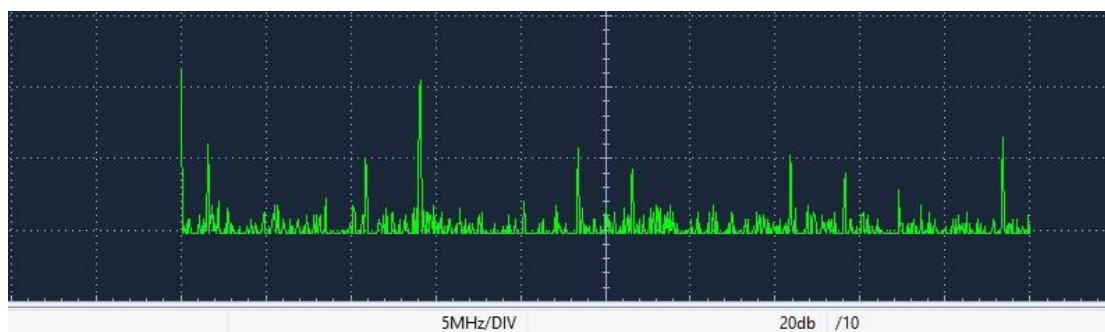


Figura 64 Señal emisión WSPR

Cada división corresponde a 5 (Mhz) y en la frecuencia 0 (Hz), el pico que podemos ver corresponde a la componente de corriente continua.

Una vez comprobado que el programa y la placa funcionan correctamente, deberemos fabricar una antena apta para la banda de 20 metros.

Para ello necesitamos un conector de antena SMA macho para poder soldar una antena de 20 metros y poder probar así la emisión de nuestro dispositivo.

Debido a que la antena es muy larga y no puede ser muy pesada, he reutilizado un cable de antena con conector SMA macho y he dejado la parte apantallada dentro de la sonda, para evitar interferencias con la electrónica, mientras que la parte del vivo, sale hacia el exterior del globo por su parte inferior, de forma que queda colgando en polarización vertical, para ello he utilizado un hilo flexible, puesto que es menos pesado y es más fácil de trabajar con él.



Figura 65 Antena WSPR

Una vez soldada la antena, podemos conectarla a la placa y realizar las pruebas, para ello he utilizado dos opciones.

La primera ha sido un sistema de *Software Defined Radio* (SDR), mientras que la otra opción ha sido utilizar una radio que la URE Cartagena puso a nuestra disposición.

Ambos métodos serán explicados detalladamente en los siguientes apartados de este documento, puesto que en este apartado solo vamos a probar el

funcionamiento de los servidores web, ya que estos serán los que usaremos el día del lanzamiento.

Existen tres páginas web de gran utilidad para nuestra aplicación, la primera de ellas es la página web: <http://wsprnet.org/drupal/>

En esta página podemos encontrar muchísima información, foros, páginas web, aplicaciones de terceros, etc... Nosotros solo nos centraremos en dos apartados principales de esta página web, para ello en la parte superior de la página, nos encontramos este marcador.



Figura 66 Banner WSPRnet

En este marcador aparecen las dos pestañas que vamos a utilizar, la primera de ellas será la pestaña de “Map”, si pulsamos sobre ella, nos aparecerá una ventana en la que veremos un mapa con varios puntos marcados en él, estos representan los puntos activos en directo, así como que estaciones los escuchan.

Esto se representa mediante una línea que une los diferentes puntos, si pulsamos sobre uno de estos puntos, aparecerá una pequeña ventana donde nos indicará a que estaciones está recibiendo.

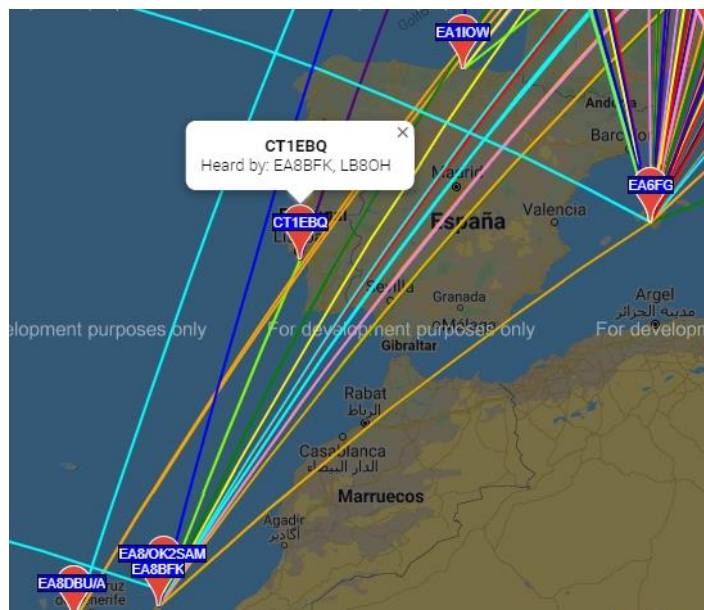
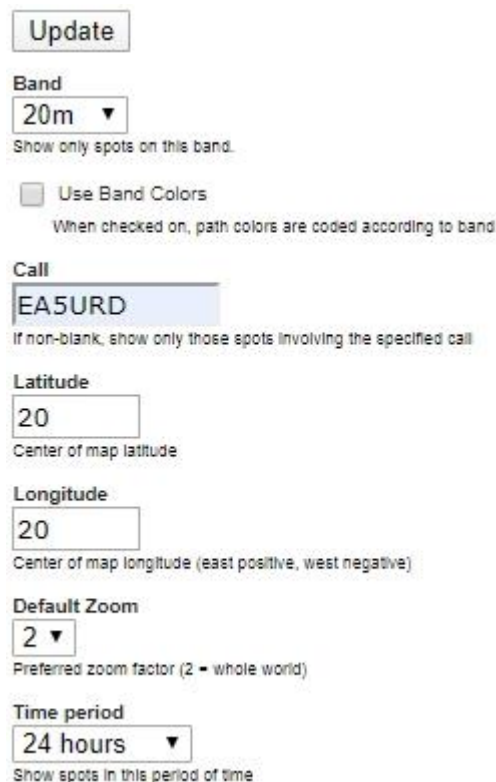


Figura 67 WSPR map

Sin embargo, este mapa está mostrando estaciones con unos parámetros predeterminados, si queremos ver nuestra estación en este mapa, primero deberemos configurarlo. Para ello, justo debajo del mapa existen una serie de campos a rellenar, voy a explicarlo poco a poco.

El primer campo para rellenar es el de “*Band*”, en nuestro caso, nosotros utilizamos la banda de 20 metros, existen muchas otras, pero nuestro dispositivo, esta optimizado para esta banda.

El siguiente campo para rellenar es el de “*Call*”, es decir, el indicativo de llamada, en mi caso es EA5URD, en caso de que se esté siguiendo este documento para realizar un proyecto similar, es importante dejar constancia de que este indicativo no puede ser utilizado, si se utiliza sin permiso y/o sin licencia de radioaficionado, se estaría incurriendo en un delito tipificado en el código penal.



Update

Band
20m ▼
Show only spots on this band.

☐ Use Band Colors
When checked on, path colors are coded according to band

Call
EA5URD
If non-blank, show only those spots involving the specified call

Latitude
20
Center of map latitude

Longitude
20
Center of map longitude (east positive, west negative)

Default Zoom
2 ▼
Preferred zoom factor (2 = whole world)

Time period
24 hours ▼
Show spots in this period of time

Figura 68 Ajustes WSPR map

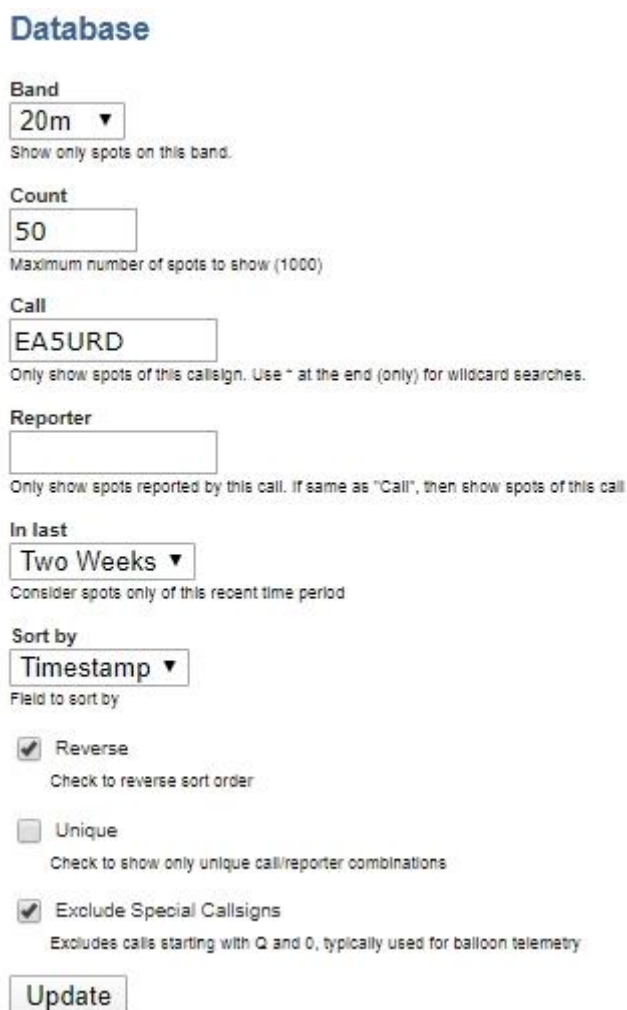
Por último, debemos completar el campo de “*Time period*”, en este caso, se puede elegir una opción entre los últimos 10 minutos y las últimas 24 horas, debemos seleccionar la que mejor se adapte a las necesidades de cada uno.

Cuando se tengan todos los campos rellenos, en la parte superior del listado, aparece un botón de “*Update*”, pulsamos sobre él y ya podemos ver el mapa actualizado, si estamos emitiendo y somos recibidos, nos aparecerá algo similar a la imagen anterior, pero con nuestro indicativo de llamada en nuestra posición.

La segunda pestaña que debemos abrir es la de “*Database*”, al pulsar sobre ella, podremos ver una ventana en la que aparecen varios campos que tienen que ser rellenados para poder cargar la base de datos correctamente.

El primer campo que rellenar será el correspondiente a “*Band*”, al igual que en el caso anterior, debemos seleccionar la banda de 20 metros.

El siguiente campo será “*Count*”, lo cambiaremos al 1.000, para asegurarnos de recibir todos los datos disponibles.



Database

Band
 Show only spots on this band.

Count
 Maximum number of spots to show (1000)

Call
 Only show spots of this callsign. Use * at the end (only) for wildcard searches.

Reporter
 Only show spots reported by this call. If same as "Call", then show spots of this call

In last
 Consider spots only of this recent time period

Sort by
 Field to sort by

☒ **Reverse**
Check to reverse sort order

☐ **Unique**
Check to show only unique call/reporter combinations

☒ **Exclude Special Callsigns**
Excludes calls starting with Q and O, typically used for balloon telemetry

Figura 69 Ajustes WSPR database

El campo “*Call*”, será de nuevo EA5URD.

Por último, el campo “*In last*”, será el más conveniente a nuestro caso, puede ir de 10 minutos a dos semanas.

Cuando todos los campos estén completos, tenemos que pinchar sobre el botón “Update”.

Y si nos han recibido, aparecerá algo parecido a esto.

Spot Database

Specify query parameters

2 spots:

Timestamp	Call	MHz	SNR	Drift	Grid	Pwr	Reporter	RGrid	km	az
2020-02-11 17:28	EA5URD	14.097064	+8	-1	IM97mo	0.02	EA5VN	IM97	16	208
2020-02-11 17:08	EA5URD	14.097145	+3	-3	IM97mo	0.02	EA5VN	IM97mo	0	0

Query time: 0.010 sec

Figura 70 WSPR database

En este listado, nos vamos a centrar en los campos: *Timestamp*, *call*, *Grid*, *Power*, *Reporter*, *RGrid* y *Km*.

“*Timestamp*”, indica la fecha y la hora del mensaje.

“*Call*”, indica nuestro indicativo de llamada.

“*Grid*”, indica nuestro cuadrante.

“*Power*”, indica la potencia con la que nos reciben, esto será muy útil para estimar la altura a la que se encuentra la sonda.

“*Reporter*”, indica el indicativo de llamada de quien nos está recibiendo.

“*RGrid*”, indica el cuadrante de nuestro *reporter*.

“*Km*”, indica la distancia en Km entre ambas estaciones.

En caso de que nuestra emisión se realizase hace más de dos semanas, aún podremos descargar los datos en la ventana de “*Downloads*”, en esta aparecerán archivos comprimidos con los datos mensuales desde 2008.

Como podemos ver, en este ejemplo muestra que la sonda se encuentra en el GRID: IM97mo, para saber con exactitud donde es esto, deberemos presentar la segunda web que vamos a utilizar, esta es: http://levinecentral.com/ham/grid_square.php

Enter any address, city & state or zip:
or Enter any call sign: Data provided by QRZ.com
or Enter any a 4 or 6 character grid square:
[?? How does this work? Why doesn't this work? ??](#)

Submit

Clear

Figura 71 Buscador locator

En ella encontraremos este recuadro, en el que introduciremos nuestro “locator” y posteriormente pulsaremos sobre “Submit” y nos aparecerá la siguiente imagen:

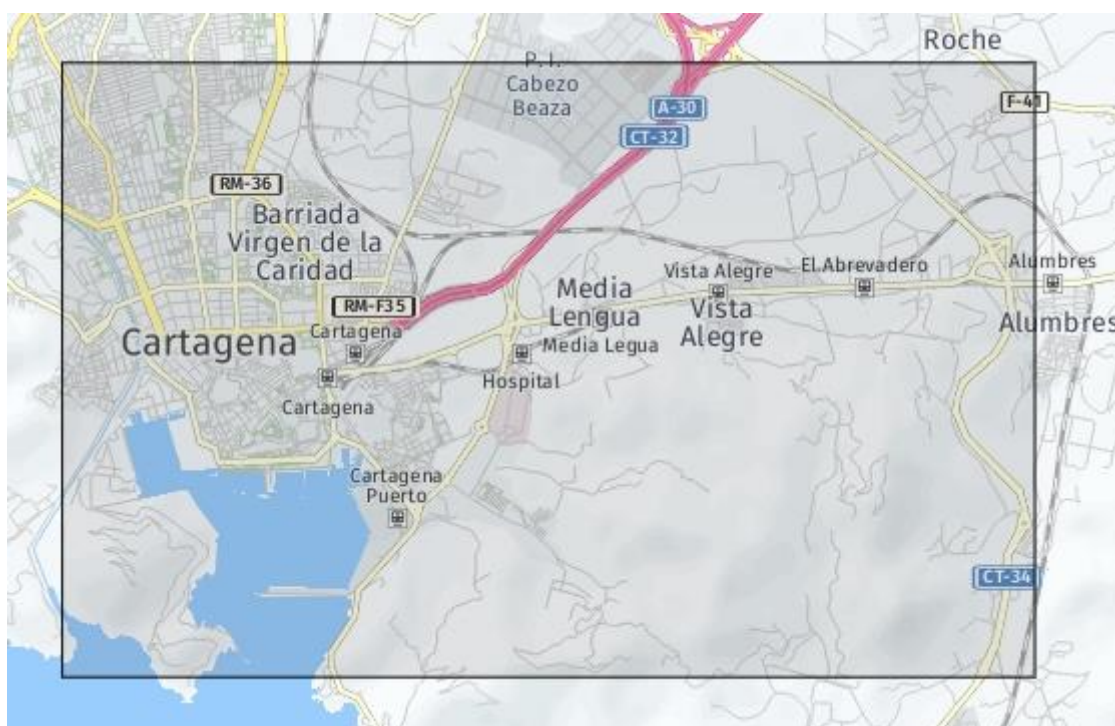


Figura 72 GRID locator WSPR

Esta será la posición aproximada de nuestra sonda, vuelvo a recordar que el error es grande, pero a lo largo del vuelo que puede durar hasta tres horas, se pueden llegar a recorrer hasta 150 (Km), por lo tanto, en esa escala de distancias, este sistema puede ser muy útil para localizarlo, sobre todo teniendo en cuenta que pueden recibir nuestra señal a miles de kilómetros de distancia.

Para ejemplificar la definición del GRID *locator*, si se realiza la misma prueba, pero quitando los dos últimos dígitos, el resultado sería el siguiente:

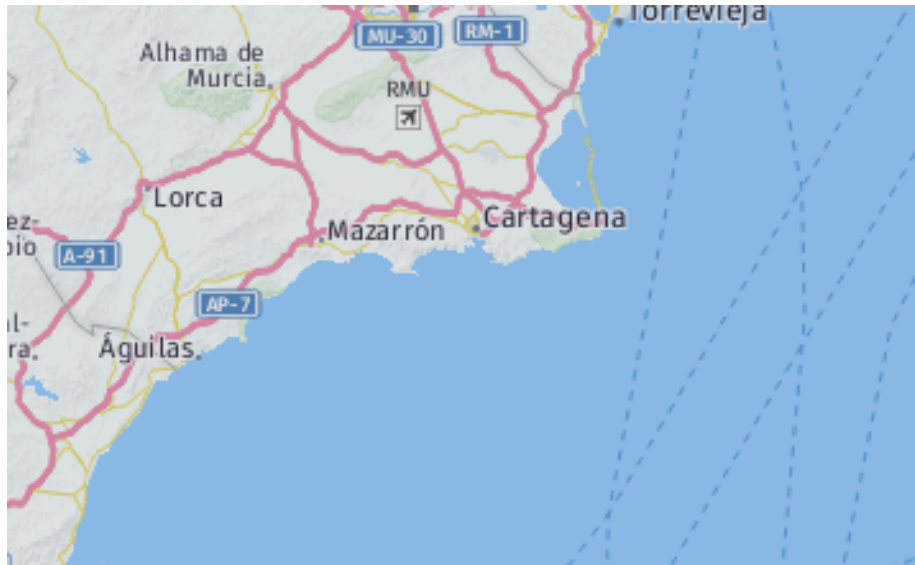


Figura 73 IM97

Por último, presento la tercera web que vamos a utilizar: <http://wspr.aprsinfo.com/>

En ella nos encontramos este *banner*, el cual tendremos que rellenar con estos datos.

Dir: RX/TX ▼	Call: EA5URD	Band: 20m ▼	Count: 1000	Time: Two Weeks ▼	Load	Auto: ☒
--------------	--------------	-------------	-------------	-------------------	------	---

Figura 74 Configuración APRS

Los datos son los mismos que en la primera página web.

Rellenamos los campos y pulsamos sobre “load”.

Si nos han recibido encontraremos algo similar a esto:

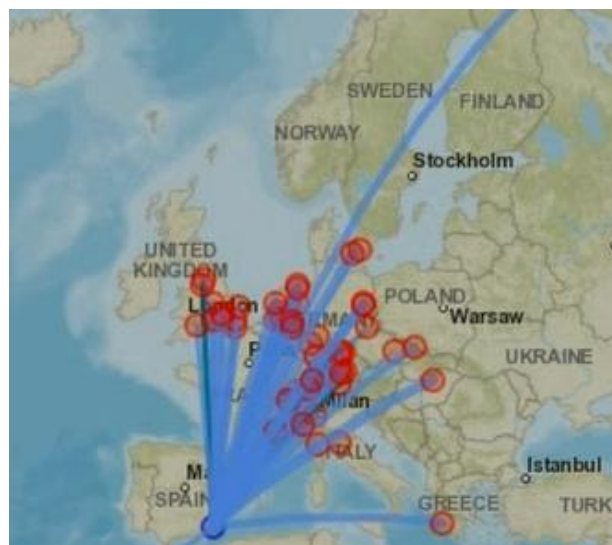


Figura 75 Reporte de propagación

En este caso la información aportada es solo visual, pero también puede resultar muy útil.

Existen otras páginas web en las que se puede realizar un seguimiento, sin embargo, las principales son las dos comentadas anteriormente.

Con toda esta información se puede tener operativo un equipo WSPR, probar su funcionamiento y solo queda pendiente explicar la fuente de alimentación, así como otros métodos de recepción que serán comentados en próximos capítulos.

4.1.2 SISTEMAS OPCIONALES

Estos sistemas comentados anteriormente, son los que finalmente he seleccionado para el vuelo, sin embargo, existen una gran cantidad de alternativas en el mercado, muchas de ellas muy prometedoras, el hecho de seleccionar un equipo y no otro, está condicionado por muchos factores, entre ellos: económicos, consumo de energía, peso, etc...

Algunos de estos dispositivos son de tecnologías similares a las utilizadas, sin embargo, otros sistemas funcionan de forma completamente diferente.

Todas las opciones expuestas a continuación son gratuitas una vez adquirido la placa electrónica, excepto una de las alternativas, la cual precisa de créditos para funcionar.

En este apartado, haré un inciso para comentar los dispositivos inicialmente barajados para el vuelo, pero que finalmente fueron descartados, la idea es tratar de explicar porque se ha realizado este descarte y hacer una pequeña comparativa entre tecnologías.

4.1.2.1 LORAWAN

Últimamente se habla mucho sobre LoRaWAN y LoRa, lo primero que tiene que quedar claro es que estos dos términos no son lo mismo.

Long Range Modulation (LoRa), es un tipo de modulación de muy larga distancia e ideal para el internet de las cosas.

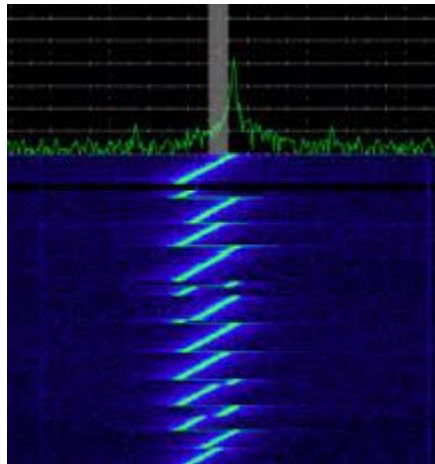


Figura 76 Modulación LoRa

Mientras que LoRaWAN, es un protocolo de red que usa la tecnología LoRa.

Estas redes están integradas por *gateways* y nodos, los primeros son los encargados de recibir y enviar información a los nodos y los segundos, son los dispositivos finales que envían y reciben información hacia el *gateway*.

La Topología de la red es en estrella, nosotros necesitaríamos integrar un nodo en el globo sonda y luego disponer de un *gateway* de terceros o propio para recibir la señal, además este *gateway* debería tener acceso a internet si queremos volcar los datos online.

En el mercado existen una multitud de opciones tanto de nodos como de *gateways*, un ejemplo de nodo podría ser el modelo de Arduino MKR WAN 1300.



Figura 77 MKRWAN 1300

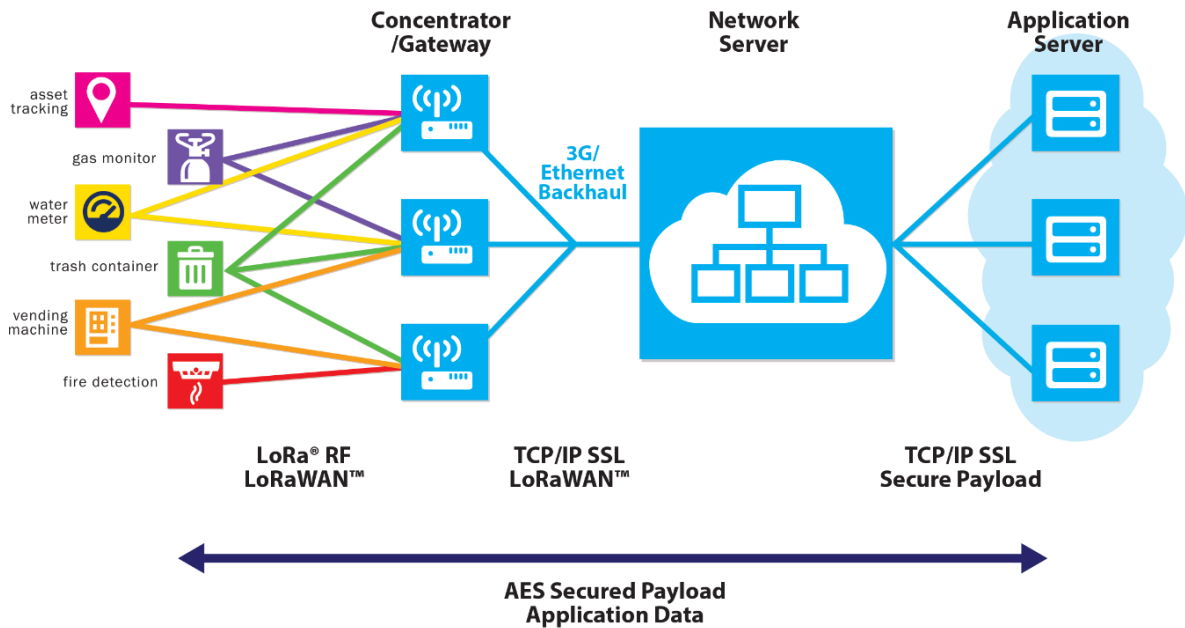


Figura 78 Topología de red LoRa

Sin embargo, el problema de este tipo de tecnología es que, para poder hacer un seguimiento del globo, necesitamos desplegar nosotros la red o bien utilizar los gateways e infraestructuras de terceros.

Por ello finalmente he decidido usar SigFox, puesto que son tecnologías muy parecidas, de largo alcance, bajo consumo, enfocado a M2M, etc...

Las principales diferencias son las siguientes:

- SigFox posee menor ancho de banda y los mensajes son de menor tamaño que los de LoRa.
- SigFox posee un mayor alcance que LoRa,
- Ambas tecnologías son bidireccionales, pero SigFox está limitado en el número de mensajes, tanto de subida como de bajada.
- SigFox posee toda la infraestructura y cobertura por toda España.
- LoRa es gratuito mientras que SigFox es de pago.

La lista puede alargarse mucho más, sin embargo, lo que me ha hecho decidirme por SigFox, es el hecho de que programas el nodo y te olvidas, con un solo teléfono puedes recibir toda la información, es muy cómodo para aplicaciones móviles en

las que puede que el globo pase por zonas sin cobertura LoRa, pero sí que va a tener cobertura SigFox.

Sin embargo, LoRa es muy buena opción y en un futuro, cuando la tecnología esté un poco más madura y la cobertura sea mayor, sin lugar a duda será el candidato ideal para este tipo de aplicaciones.

4.1.2.2 IRIIDIUM

Iridium es una compañía de telefonía por satélite que tiene desplegados 66 satélites, ofreciendo así cobertura global.

Tanto para Arduino como para Raspberry Pi, se puede utilizar el módulo RockBLOCK Mk2 – *Iridium Satcomm Module*.

Este módulo, permite comunicación bidireccional con nuestro dispositivo en cualquier parte del mundo, excepto en los polos.

Funciona con una suscripción, con la que incluye una serie de créditos, con estos créditos podemos enviar mensajes de texto a nuestro correo o a un servidor, cuando estos créditos se acaban, se puede volver a recargar a través de la página web de Iridium.



Figura 79 GPS Iridium



Figura 80 Módulo Iridium

Es una opción muy fiable, algunos proyectos embarcan este módulo en boyas meteorológicas y siguen su recorrido a través de océanos con este dispositivo, sin embargo, su precio es bastante elevado, al igual que el precio de los créditos, por ello no hemos seleccionado esta opción.

El precio ronda los 250 \$ más los créditos y gastos de envío.

Sin embargo, este aparato nos garantiza la recuperación de nuestra sonda, siempre que se opere en condiciones normales.

4.1.2.3 PI IN THE SKY

Es una placa desarrollada especialmente para Raspberry Pi, aunque también existe una versión para Arduino.

Esta es una placa genérica a la que se le pueden añadir hasta dos módulos de comunicaciones, uno de ellos es LoRa y el otro Sistema Automático de Información de Posición (APRS).



Figura 81 Pi in the Sky

Sus principales características son:

- GPS de gran altura.
- Sensor de temperatura.
- Entrada de Pi camera.
- SSDV Radio imagen.
- Radio telemetría de GPS y sensores.
- Guarda las imágenes en la tarjeta SD.
- Transmisión en Radioteletipo (RTTY).
- Recepción a través de SDR o con Gateway LoRa.

Esta opción es muy fiable y potente, permite utilizar LoRa, obtener telemetría de larga distancia por Radioteletipo, obtener imágenes a través de radio, etc...

Además, es un Sistema muy documentado y utilizado, sin embargo, al igual que en el caso anterior, el precio fue la razón por la cual finalmente quedó descartado este sistema.

El dispositivo ronda las 207 £, más la Raspberry Pi, la tarjeta de memoria y antenas. Aproximadamente 260 €.

4.1.2.4 WSPR-TX PICO TRANSMITTER

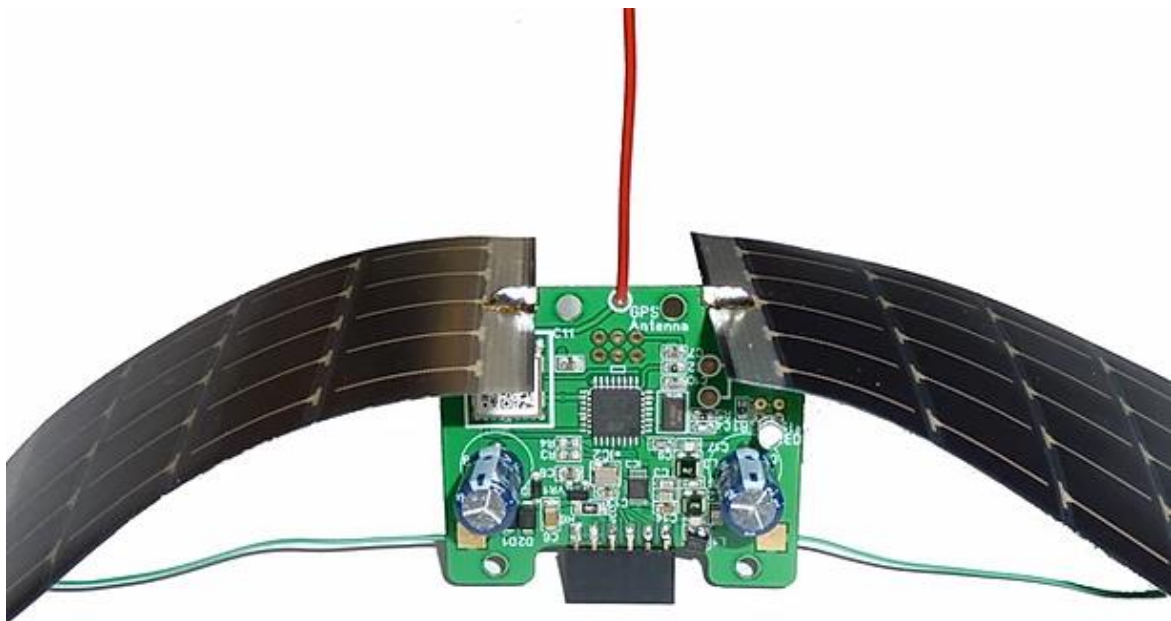


Figura 82 WSPR-TX PICO

Este dispositivo es una versión del WSPR optimizada para globos sonda, la alimentación se realiza a través de dos paneles solares y además con estos también realiza la carga de dos supercondensadores de 1 (F).

La antena GPS, es de un cuarto de onda y de esta forma el tamaño de la paca y el peso de esta se reduce drásticamente.

Este dispositivo emite entre los 10 y los 14 (Mhz), en funcionamiento, requiere de 5 (V), con una corriente de 60 (mA), el peso es de 10,5 (g) con los paneles solares incluidos.

No se utilizó este sistema, puesto que salió a la venta después del lanzamiento.

4.2 SUBSISTEMA DE COMPUTADORA

Uno de los objetivos de este proyecto es la toma de datos durante el vuelo, para después procesar estos datos y obtener conclusiones.

Estos datos pueden estar relacionados con la meteorología, con las aceleraciones de la sonda, posición, altura, etc...

En este caso, necesitamos desarrollar un sistema capaz de tomar diferentes datos procedentes de sensores y almacenarlos en una memoria SD para poder estudiar estos datos una vez recuperada la sonda.

Por ello a continuación, voy a explicar que autómata he utilizado, así como los sensores y actuadores.

Luego explicaré como se conectan y otros datos importantes relacionados con los buses de comunicaciones, para finalmente exponer el programa utilizado y explicarlo en profundidad.

4.2.1 AUTÓMATA

La elección del autómata puede resultar bastante complicada, sobre todo si no se tiene mucho conocimiento sobre estos.

En el mercado existen infinidad de marcas y modelos. Algunos de estos autómatas están encuadrados dentro de un grupo denominado *Hardware Libre*.

Esto significa que los esquemas electrónicos de los autómatas son públicos y cualquiera con los suficientes conocimientos, podría emular uno de estos, además, la mayoría de estos autómatas, también hacen uso de *Software Libre*. Por ello tenemos miles de usuarios que comparten sus trabajos y a los cuales podemos acudir a través de foros para la resolución de problemas.

En este caso vamos a utilizar un autómata de la familia Arduino.

Este fabricante posee una gran cantidad de modelos, algunos muy específicos, como por ejemplo el MKR FOX 1200 o el MKR WAN 1300, que ya hemos visto anteriormente, pero también dispone de una familia de autómatas mucho más

genéricos, entre estos se encuentran los modelos: Uno, Duemilla, Leonardo, Nano, Micro, Mega, Due, etc...

Las diferencias entre estos están en los pines disponibles, memoria EEPROM, memoria *flash*, memoria RAM, velocidad de reloj, etc...

Para nuestra aplicación, tenemos que encontrar un modelo que permita la conexión de todos los sensores, pero también que la memoria de almacenamiento sea adecuada para el programa que se va a utilizar. Ya que en caso de que la memoria quede corta, podría ocasionar problemas de estabilidad a la hora de ejecutar el programa.

La primera opción utilizada en el proyecto fue Arduino Uno, pero en cuanto el programa fue creciendo, los problemas de estabilidad fueron apareciendo, por ello se decidió utilizar el Modelo Arduino Mega, en una versión mucho más pequeña que el Mega original de Arduino.

Tabla 6 Comparación Arduino UNO y MEGA

	MEGA	UNO
ALIMENTACIÓN	5 (V)	5(V)
PINES DIGITALES	54	14
PINES ANALOGICOS	16	6
MEMORIA FLASH	256 (KB)	32 (KB)
SRAM	8 (KB)	2 (KB)
EEPROM	4 (KB)	1 (KB)
VELOCIDAD DE RELOJ	16 (MHz)	16 (MHz)



Figura 83 Comparativa Mega vs Mega Mini

Una vez determinado el autómata a usar, vamos a ver en profundidad sus características.

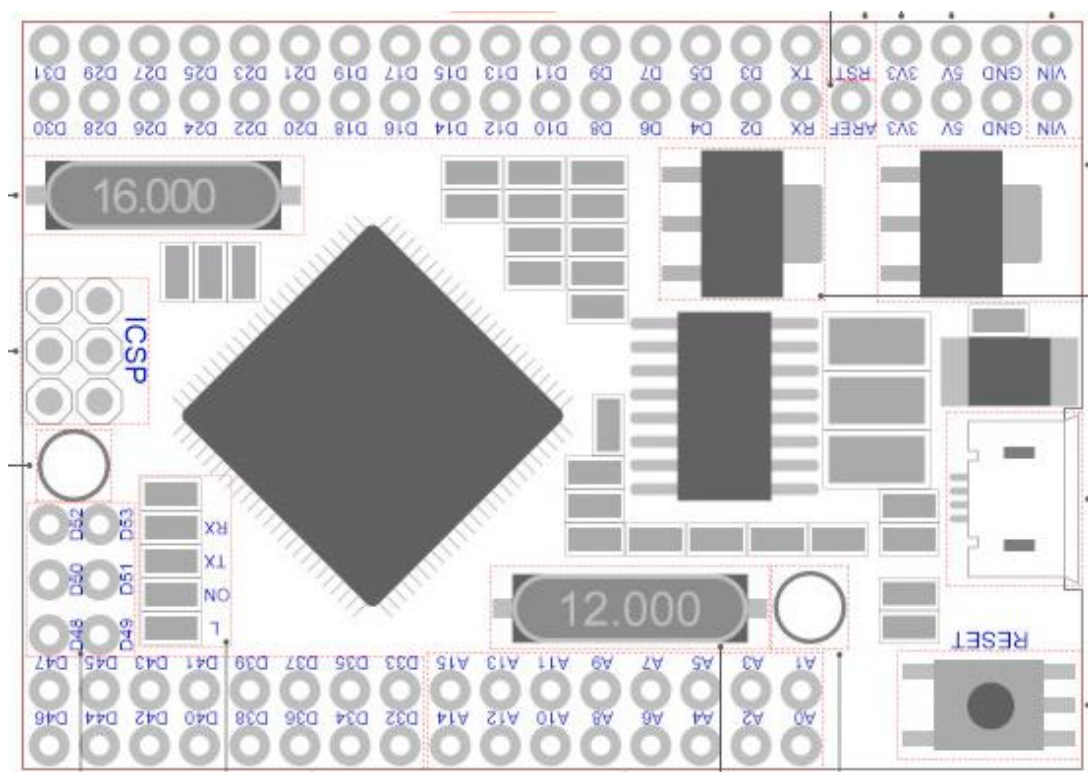


Figura 84 Mega Mini

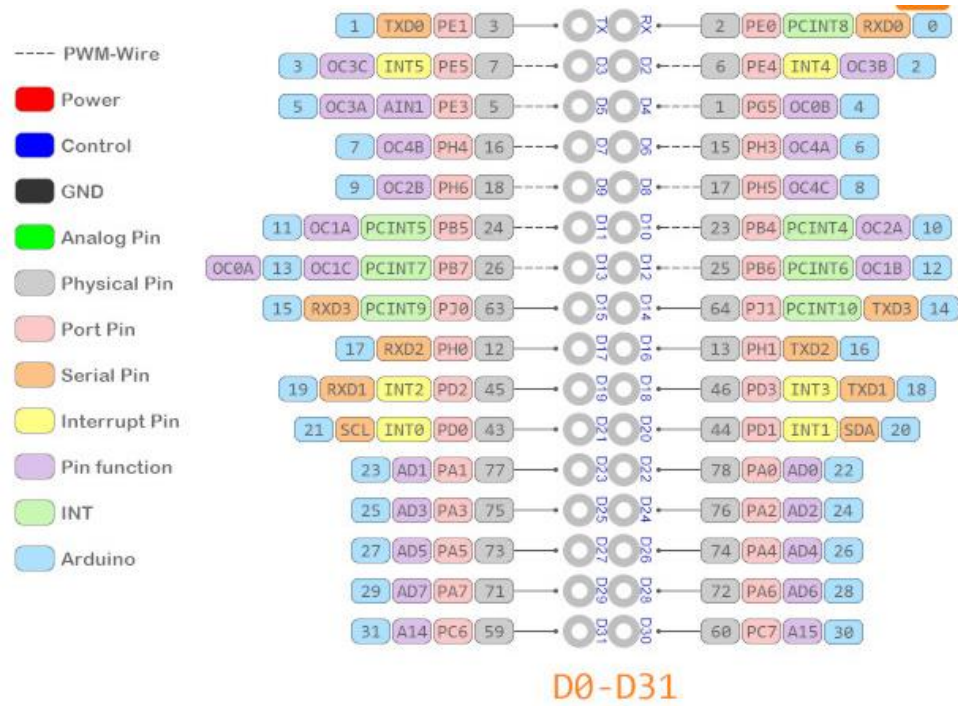


Figura 85 PinOut Mega 1

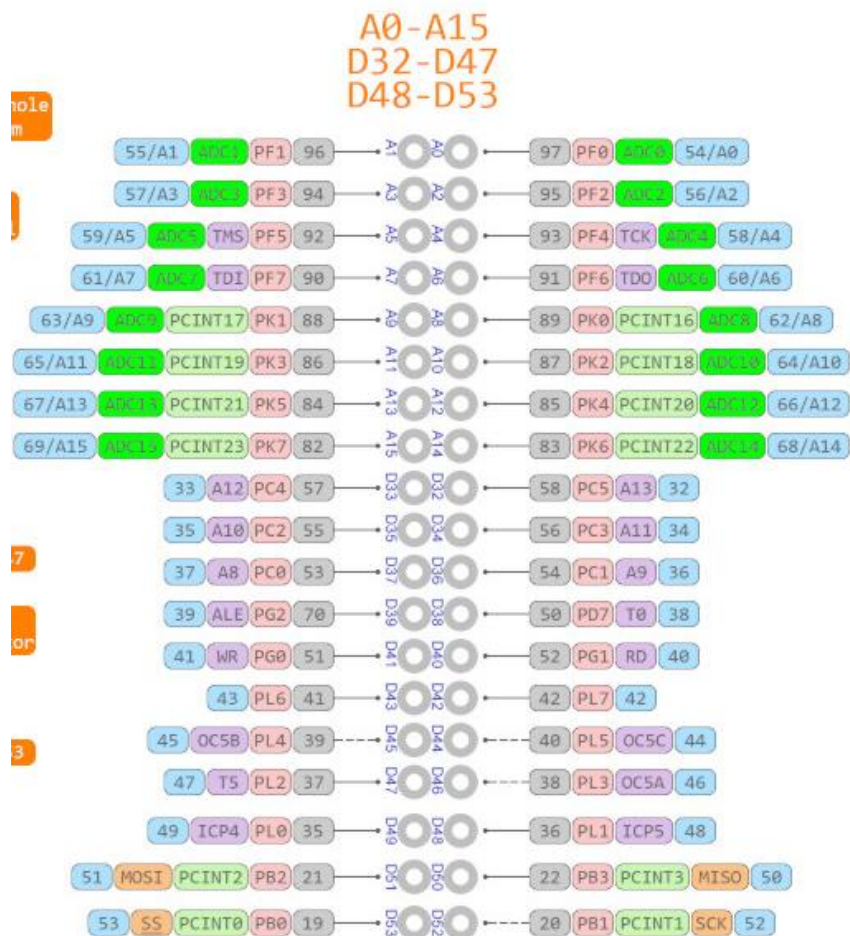


Figura 86 PinOut Mega 2



Figura 87 PinOut ICSP

Como podemos ver en el *PinOut* del Arduino Mega, disponemos de dos puertos ICSP, un puerto I2C y cuatro puertos Seriales Físicos.

Esto es muy importante para esta aplicación, puesto que se va a hacer uso de todos estos buses de comunicaciones.

A continuación, voy a explicar cada uno de estos buses de comunicaciones.

4.2.1.1 I2C

El bus de comunicaciones *Inter-Integrated Circuit* (I2C) fue desarrollado para la comunicación interna de dispositivos electrónicos.

Este bus solo necesita dos cables para funcionar, uno de ellos es la señal de reloj (SCL), mientras que el otro cable envía datos (SDA).

En este bus, cada dispositivo conectado, necesita una dirección física única.

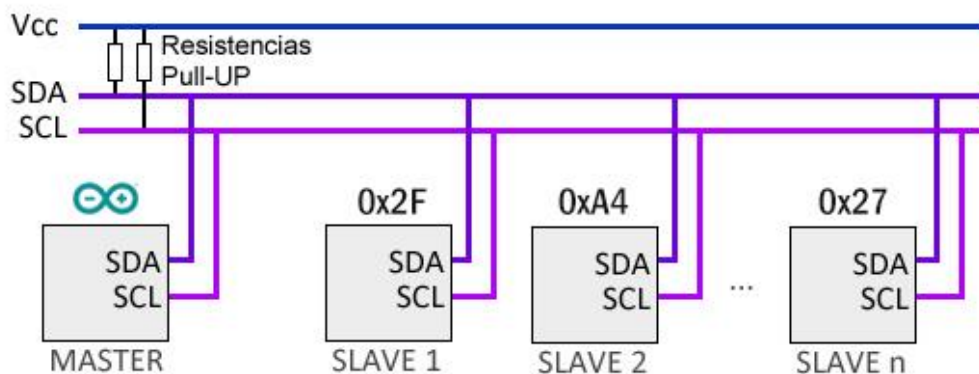


Figura 88 Bus I2C

Esta, a veces se puede cambiar por *software* o por *hardware*. Si no fuera posible y dos dispositivos tienen la misma dirección entonces se debería crear otro bus I2C independiente.

El funcionamiento del bus es de tipo *Master-Slave*, o lo que es lo mismo, Maestro-esclavo.

Es decir, la comunicación esta moderada por el maestro, este es el que inicia la comunicación y determina cuál de los dispositivos esclavos debe comunicarse.

El bus I2C, necesita para su funcionamiento dos resistencias *pull-up* entre la alimentación y los dos cables del bus, en Arduino estas resistencias ya vienen incluidas en el PCB.

Este bus es muy útil, puesto que permite conectar una gran cantidad de sensores utilizando solo dos pines de nuestro Arduino, pero también tiene el inconveniente de que no posee mucha velocidad y que no hay comprobación de que el contenido del mensaje sea correcto.

4.2.1.2 ONE-WIRE

El protocolo *One Wire*, es un tipo de bus de datos que no tiene una línea de reloj dedicada.

La forma que tiene de codificar los datos es mediante “codificación bipolar” por ejemplo 3,3 (V) para 1 y -3,3 (V) para 0.

La idea es que si por ejemplo 0 (V) equivale a 0, no podemos diferenciar si es un 0 o que se ha soltado un cable.

Aun así, sigue existiendo el problema del reloj, por lo que se crean unos patrones denominados *Self clocking*, estos hacen que cada cierto número de bits, cambien de estado para así volver a sincronizarse con el reloj del maestro, de esta forma se comparten datos y reloj a la vez.

Al igual que en el resto de los buses, podemos conectar diferentes dispositivos al mismo bus, aumentando la cantidad de sensores sin desperdiciar pines.

4.2.1.3 SPI

Serial Peripheral Interface (SPI), es un bus de comunicaciones para comunicar circuitos integrados entre sí, con él se puede hacer un sistema de maestros esclavos de dispositivos que podrían ser hasta diferentes autómatas unos controlando a otros.

El SPI es un protocolo síncrono y funciona por medio de 4 señales:

-SCLK (Clock): Es el pulso que marca la sincronización.

-MOSI (Master Output Slave Input): Salida de datos del Master y entrada de datos al Esclavo.

-MISO (Master Input Slave Output): Salida de datos del Esclavo y entrada al Master.

-SS/Select: Para seleccionar un Esclavo.

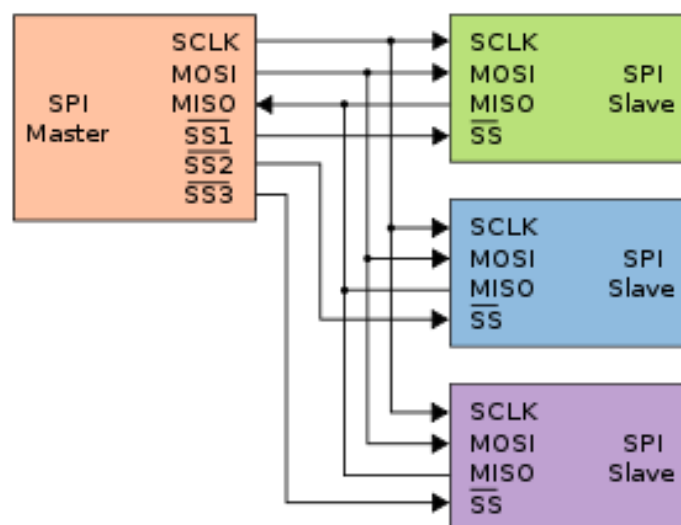


Figura 89 Bus SPI

Uno de los mayores inconvenientes de este bus es que no funciona a largas distancias.

En nuestro proyecto, este bus será utilizado para el control de un lector SD, es decir nuestro Arduino será el *Master* y el lector un *Slave*.

4.2.1.4 UART

Universal Asynchronous Receiver/Transmitter (UART), es un bus de comunicaciones, este está compuesto por 3 cables: TX, RX y GND.

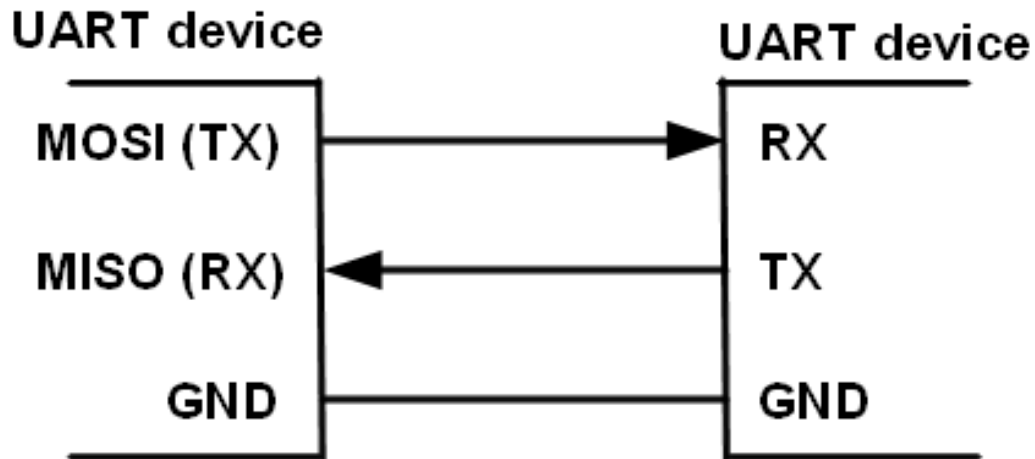


Figura 90 Conexión UART

La conexión debe ser de un TX de uno al RX del otro y viceversa, además todos deben estar conectados a la misma masa.

Es muy importante que a estos buses solo se conecte un dispositivo, para conectar varios, se deben usar otros buses.

Otra faceta muy útil de este bus es que trabaja con niveles lógico TTL, lo cual hace que dos micros puedan comunicarse sin necesidad de convertir los voltajes.

Es decir, se podría trabajar con un integrado que funcione a 5 (V) y otro que funcione a 3,3 (V).

Este bus será muy útil para trabajar con el GPS.

4.2.2 SENSORES Y ACTUADORES

A continuación, se exponen los diferentes sensores utilizados, así como los actuadores.

4.2.2.1 SENSORES

GN-701

Este dispositivo es un receptor GPS, utilizado en el Arduino MKR FOX 1200 y en el Mega.

Este GPS recibe señal tanto de la constelación *Global Positioning System* (GPS) como de la *Global'naya Navigatsionnaya Sputnikovaya Sistema* (GLONASS).

Tiene regulación interna de voltaje, por lo que puede funcionar alimentado a 3,3 (V) como a 5 (V)

Es un dispositivo de relativo bajo consumo, alimentado a 3,3 (V), el consumo es de 35 (mA) en adquisición y de 22 (mA) en modo *tracking*.

La comunicación serial debe establecerse a 9.600 (Bd) según el *datasheet* del fabricante.



Figura 91 GN-701

Por último, la limitación de este dispositivo en temperatura es de -40°C y tiene un techo máximo de altura de 18.000 (m).

DHT-11

Este es un sensor de humedad y de temperatura, la comunicación es mediante *One Wire*.

Puede medir la temperatura en un rango desde los 0°C hasta los 50°C.

Respecto a la humedad, es capaz de medir en un rango comprendido entre el 20% y el 90% de humedad relativa.

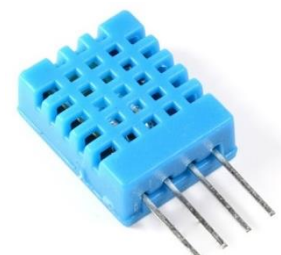


Figura 92 DHT-11

Debido a este rango de valores, este sensor, está enfocado a medir la humedad, así como la temperatura en el interior de la sonda, puesto que está previsto que la temperatura en el interior no sea inferior a 0°C.

DS18B20

Este es un sensor de temperatura con un encapsulado especial que le da capacidad para sumergirse en líquidos, por ello este sensor será utilizado en el exterior de la sonda para medir la temperatura.

Este sensor se conecta al Arduino mediante el bus *OneWire*, es importante recordar el uso de una resistencia *pull-up* para que las medidas no sean erróneas.

Este sensor mide temperaturas en un rango comprendido entre -55°C y 125°C, la precisión de las medidas es de $\pm 0,5^\circ$ en el rango de -10°C hasta +85°C, en el resto de rangos, la precisión es de $\pm 1^\circ$.



Figura 93 DS18B20

BMP180

Este dispositivo es un sensor de presión barométrica que además incluye un sensor de temperatura.

La conexión de este sensor con el Arduino se realiza mediante el bus de comunicaciones I2C.

El rango de presiones que es capaz de medir va desde los 300 hasta los 1.100 (hPa), que en altura aproximadamente representa un rango comprendido entre los -500 (m) hasta los 9.000 (m), relativos al nivel del mar.

Respecto a la temperatura, el rango que es capaz de medir varía entre los 0°C y los 85°C, con una precisión de $\pm 1^\circ$.



Figura 94 BMP-180

MPU6050

El sensor MPU6050, es una placa que incluye varios integrados, estos son capaces de facilitar varios datos, este sensor usualmente se usa como giróscopo y acelerómetro.

Su comunicación con el Arduino se realiza mediante el bus de comunicaciones I2C.

Como dato importante, cabe destacar que este modelo posee un pin que se utiliza para modificar la dirección del I2C para no tener problemas de compatibilidad con otros sensores.

Este sensor es de 6 ejes, esto significa que existen tres ejes para el acelerómetro y otros tres para el giróscopo.

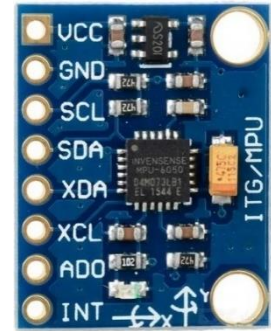


Figura 95 MPU6050

El giróscopo funciona en un rango de ± 250 , ± 500 , ± 1.000 y $\pm 2.000^\circ/\text{seg}$, esta escala puede ser modificada en función de la velocidad a la que van a ocurrir los movimientos.

El acelerómetro funciona en el rango comprendido entre $\pm 2g$, $\pm 4g$, $\pm 8g$ y $\pm 16g$, como en el caso anterior, también podemos ajustar la escala.

4.2.2.2 ACTUADORES

Una vez presentados los sensores utilizados, voy a hacer una exposición de los actuadores utilizados.

BUZZER

Este elemento no es más que un zumbador que será utilizado para comprobación sonora del funcionamiento del programa, así como para ayudar en la recuperación de la sonda.

Puede ser alimentado entre los 3 (V) y los 20 (V), en este caso será alimentado a través del pin de 5 (V) del Arduino.

Este piezo *buzzer* funciona a una frecuencia de 2,8 (KHz) y funciona a unos 8 (mA).



Figura 96 buzzer

LED BICOLOR

Este led bicolor, consiste en un led de dos patillas, que en función de cómo se polariza, emite un color u otro, es decir, si una patilla se pone a 5 (V) y otra a 0 (V), entonces tiene un color, si se hace a la inversa, obtiene otro color y si los dos están a 0 (V) o 5 (V) entonces se apaga.

Este led se utilizará para comprobar visualmente el funcionamiento del programa, así como para indicar cuando se puede manipular la tarjeta SD y cuando no.



Figura 97 Diodo Led bicolor

INTERRUPTOR

Utilizaremos un interruptor para controlar el flujo del programa y separar dos partes fundamentales, la primera de ellas será la parte de escritura de datos, mientras que la segunda será la parte enfocada en la lectura de datos.

Este no debe soportar muchos amperios, por lo que también se puede usar uno de menor calidad.



Figura 98 Interruptor

LECTOR MICRO SD

Por último, voy a utilizar un lector de tarjetas microSD, este utiliza el bus SPI para comunicarse con Arduino, este actuará como *Master* mientras que el lector actuará como *Slave*.

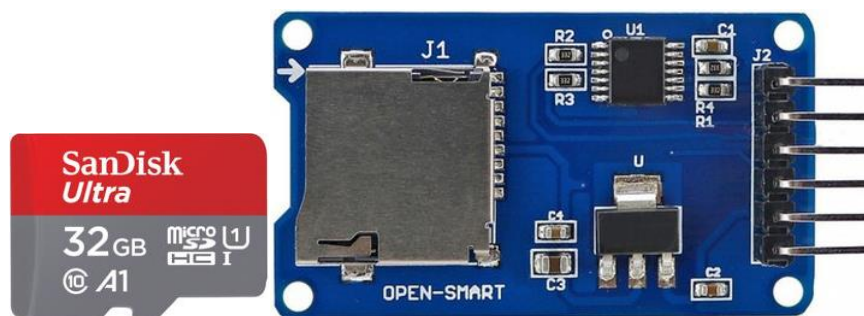


Figura 99 MicroSD shield

Utilizaré una tarjeta de 32 (Gb) para esta aplicación.

Este módulo debe ser alimentado a 5 (V) aunque internamente funciona a 3,3 (V), otros modelos permiten alimentación a 3,3 (V).

4.2.3 WIRING

En este apartado adjuntaré una serie de tablas para indicar los pines de cada sensor y donde van conectados en el Arduino y también mostraré un diagrama de conexiones.

Tabla 7 CONEXIÓN MICROSD SHIELD

ARDUINO MEGA	MICROSD SHIELD
DIGITAL 51	MOSI
DIGITAL 50	MISO
DIGITAL 52	SCK
DIGITAL 4	CS
GND	GND
5V	VCC

Tabla 8 CONEXIÓN BMP180

ARDUINO MEGA	BMP180
DIGITAL 20	SDA
DIGITAL 21	SCL
GND	GND
5V	VCC

Tabla 9 CONEXIÓN MPU6050

ARDUINO MEGA	MPU6050
DIGITAL 20	SDA
DIGITAL 21	SCL
GND	GND
5V	VCC

Tabla 10 CONEXIÓN GPS

ARDUINO MEGA	GPS
DIGITAL 16	RX
DIGITAL 17	TX
GND	GND
5V	VCC

Tabla 11 CONEXIÓN DHT11

ARDUINO MEGA	DHT11
DIGITAL 5	DATA
GND	GND
5V	VCC

Tabla 12 CONEXIÓN DS18B20

ARDUINO MEGA	DS18B20
DIGITAL 6	DATA
GND	GND
5V	VCC

Tabla 13 CONEXIÓN DUAL-LED

ARDUINO MEGA	DUAL LED
DIGITAL 44	VERDE
DIGITAL 38	ROJO

Tabla 14 CONEXIÓN INTERRUPTOR

ARDUINO MEGA	INTERRUPTOR
DIGITAL 28 (CON PULLUP)	ON
DIGITAL 29	OFF

Tabla 15 CONEXIÓN BUZZER

ARDUINO MEGA	BUZZER
DIGITAL 40	ROJO
GND	NEGRO

En las tablas anteriores, podemos ver que, en el interruptor hemos puesto un pin como *Pull-up*, esto se podrá ver en el programa más adelante, sin embargo, para el caso del DS18B20, la resistencia se pondrá físicamente de forma que la resistencia una las patillas de VCC y de DATA.

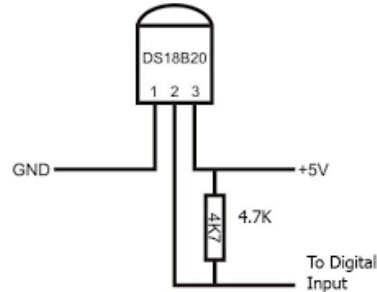


Figura 100 Pull-up

Esta resistencia nos permitirá obtener mejores resultados y que sean más continuos y menos influenciados por el ruido.

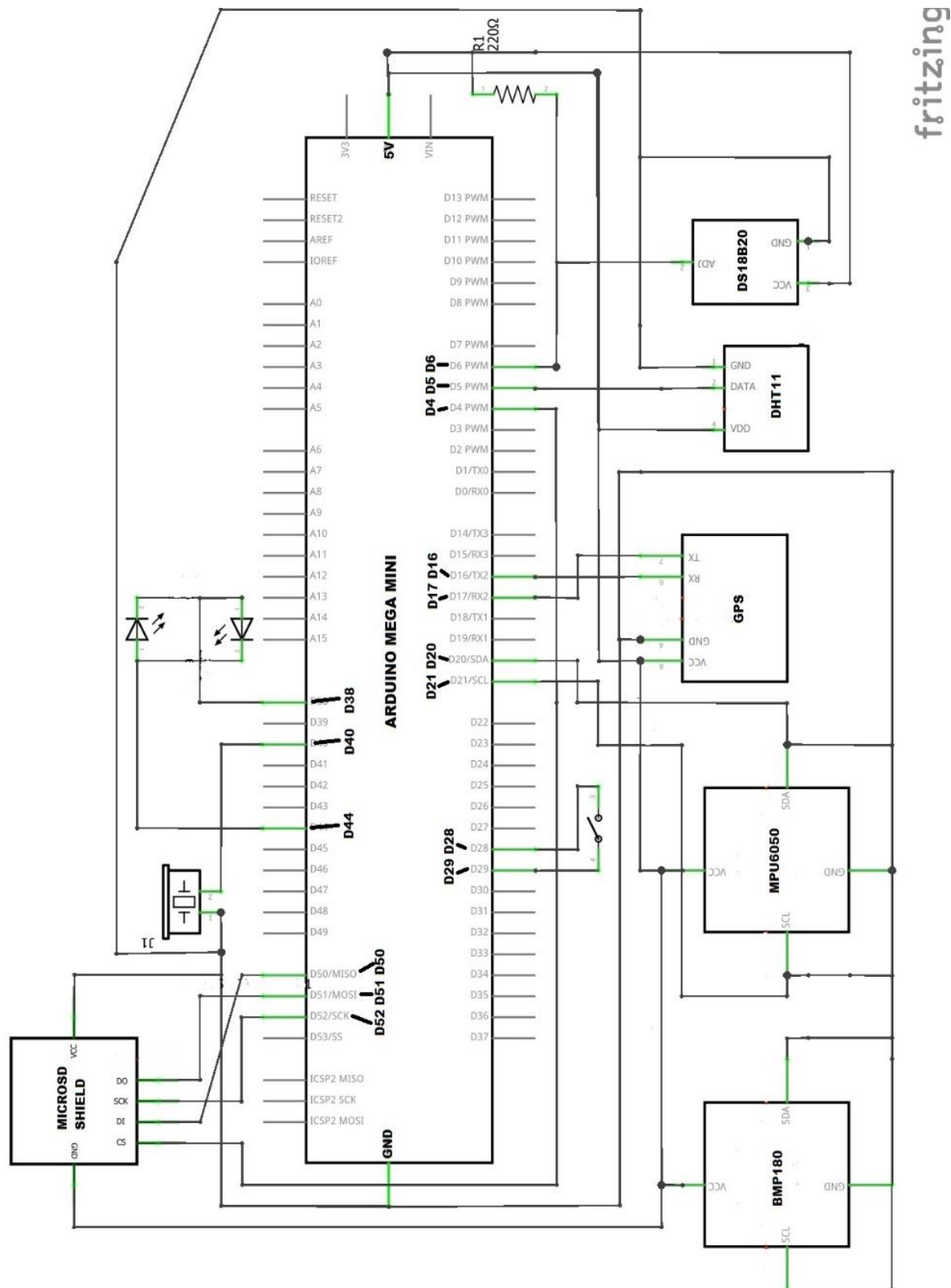


Figura 101 Esquema Fritzing Mega

4.2.4 PROGRAMACIÓN

En este apartado procedo a mostrar el código utilizado y comentaré como funciona.

Lo primero que se debe hacer es abrir el IDE de Arduino y seleccionar Arduino Mega en el menú de placas.

Primero de todo exportaremos las librerías utilizadas en el *Sketch*, muchas de ellas están ya incluidas en la carpeta de Arduino, el resto están disponibles en Github.

```
#include "DHT.h" // Libreria DHT11
#include <OneWire.h> // Libreria BUSCOM
#include <DallasTemperature.h> // Libreria TEMPERATURA
#include <SFE_BMP180.h> // Libreria BAROMETRO
#include <Wire.h> // Libreria BUSCOM
#include <SPI.h> // Libreria BUSSPI
#include <SD.h> // Libreria SD
#include <TinyGPS.h> // Libreria GPS
#include <EEPROM.h>
```

Después de incluir las librerías, procedemos a las definiciones que usaremos a lo largo del programa, la ventaja de usar los `#define`, consiste en que estos no ocupan memoria para almacenar las constantes.

```
#define DEBUG 1 // COMENTAR CUANDO SE VA A LANZAR
#define TIME_FILE 600000 // 600000

#define LED_SD_OPEN 1
#define LED_SD_CLOSE 2
#define DHTTYPE DHT11
#define MPU 0x68
#define A_R 16384.0 //Definimos los ratios de conversión
#define G_R 131.0
#define RAD_A_DEG = 57.295779 //Definimos la conversion de radianes a grados 180/PI
```

A continuación, procedemos a crear declarar e inicializar las variables que usaremos a lo largo del tiempo de ejecución del código.

```
const int BUZZ = 40;
const int LED_1 = 44;
const int LED_2 = 38;
const int PUL_1 = 28;
const int PUL_2 = 29;
const int chipSelect = 4; // SD chipSelect
const int oneWirePin = 2; // DS18D20
const int DHTPin = 5; // DHT-11
int indice;
int x;
int k;
int addr_k = 0;
char status;
char filename[] = "metsat00.txt";
char borrafile[] = "metsat00.txt";
int dato;
double T,P;

int16_t AcX;
int16_t AcY;
int16_t AcZ;
int16_t GyX;
int16_t GyY;
int16_t GyZ;

//Declaramos cadena de caracteres para los ángulos
float Acc[2];
float Gy[2];
float Angle[2];
float h;
float t;
float flat, flon;
float tamano;

unsigned long age;
unsigned long ldate, ltime, laltitude, lage, lcourse, lspeed;
```

Las siguientes líneas de código corresponden a la creación de objetos, los cuales serán usados por las librerías de los fabricantes para poder usar los sensores y módulos utilizados.

```
SFE_BMP180 bmp180; // OBJETO BMP180
DHT dht(DHTPin, DHTTYPE);
OneWire oneWireBus(oneWirePin);
DallasTemperature sensor(&oneWireBus);
File logFile;
TinyGPS gps; // OBJETO GPS
```

A continuación procedemos a exponer el void setup(), el cual solo se ejecutara una vez a lo largo de la ejecución del programa.

Lo primero que se realiza es establecer el funcionamiento de los pines digitales declarados anteriormente, es decir, establecer si funcionan como entrada o como salida.

Posteriormente vamos inicializando los sensores e iniciamos la comunicación serial a 9.600 (Bd).

Después se comprueba el correcto funcionamiento de la tarjeta SD, en caso de que falle, entraría en bucle y no se iniciaría el void loop().

```
void setup() {  
  
  pinMode(BUZZ, OUTPUT); // pin 33  
  pinMode(LED_1, OUTPUT); // pin 44  
  pinMode(LED_2, OUTPUT); // pin 38  
  pinMode(PUL_1, OUTPUT); // pin 28  
  pinMode(PUL_2, INPUT_PULLUP); // pin 29  
  digitalWrite(PUL_1, LOW);  
  
  Wire.begin();  
  Wire.beginTransmission(MPU);  
  Wire.write(0x6B);  
  Wire.write(0);  
  Wire.endTransmission(true);  
  
  Serial.begin(9600);  
  
  sensor.begin();  
  dht.begin();  
  bmp180.begin();  
  
  // see if the card is present and can be initialized:  
  if (!SD.begin(chipSelect)) {  
  
    #if DEBUG  
    Serial.println("CardFail");  
    #endif  
    // don't do anything more:  
    while (1);  
  }  
  
  #ifdef DEBUG  
  Serial.println("cardInit");  
  #endif  
}
```

A continuación se inicia la comunicación en el serial físico número 2, después se inicia un condicional if(), el cual vamos a analizar detenidamente.

La condición expuesta consiste en leer un contador que está almacenado en la memoria EEPROM, la cual sigue almacenando los datos, aunque se apague el autómatas, está pensado para que, si el programa se reinicia por algún problema, los ficheros escritos sean los mismos, de esta forma se evitan sobre escrituras o que los datos se almacenen de forma desordenada.

El contador se inicia en 0 y acaba en 15, por lo que, si el valor es mayor que 0, significa que se ha reiniciado y pone la variable “contador” con el valor leído en la memoria EEPROM.

En el caso de que el programa se inicie por primera vez, entonces la variable se pondrá a 0.

```
Serial2.begin(4800);

if(EEPROM.read(addr_k) > 0){
  k = EEPROM.read(addr_k);

  #ifdef DEBUG
  Serial.print("SE HA RESETEADO");
  #endif

}
else{

  #ifdef DEBUG
  Serial.print("SE INICIA POR PRIMERA VEZ");
  #endif

  k=0;
}
}
```

Las siguientes líneas de programación corresponden a diferentes funciones que se irán utilizando en el void loop(), la primera de ellas es la que se encarga de leer los datos de temperatura y humedad del sensor DHT-11 y la temperatura externa de la sonda procedente del sensor DS18B20.

```
void readTempHumedad() {

  h = dht.readHumidity();
  t = dht.readTemperature();

  sensor.requestTemperatures();
}
```

La siguiente función se encarga de leer los valores de presión y de temperatura del sensor BMP180, estos datos los guardará en sus variables para posteriormente guardarlos en la tarjeta SD.


```
void readBMP(){  
    status = bmp180.startTemperature();//Inicio de lectura de temperatura  
    if (status != 0)  
    {  
        delay(status); //Pausa para que finalice la lectura  
        status = bmp180.getTemperature(T); //Obtener la temperatura  
        if (status != 0)  
        {status = bmp180.startPressure(3); //Inicio lectura de presión  
            if (status != 0)  
            {delay(status); //Pausa para que finalice la lectura  
                status = bmp180.getPressure(P,T); //Obtenemos la presión  
            }  
        }  
    }  
}
```

Esta función corresponde a la lectura del sensor MPU5060, es decir, se encarga de leer los datos del acelerómetro y el giroscopio.

```
void readAccGyro(){  
    // valores del Acelerometro  
    Wire.beginTransmission(MPU);  
    Wire.write(0x3B); //Pedimos el registro 0x3B - corresponde al AcX  
    Wire.endTransmission(false);  
    Wire.requestFrom(MPU,6,true); //A partir del 0x3B, se piden 6 registros  
    AcX=Wire.read()<<8|Wire.read(); //Cada valor ocupa 2 registros  
    AcY=Wire.read()<<8|Wire.read();  
    AcZ=Wire.read()<<8|Wire.read();  
  
    // valores del Giroscopio  
    Wire.beginTransmission(MPU);  
    Wire.write(0x43);  
    Wire.endTransmission(false);  
    Wire.requestFrom(MPU,4,true); //A diferencia del Acelerometro, solo se pi  
    GyX=Wire.read()<<8|Wire.read();  
    GyY=Wire.read()<<8|Wire.read();  
}
```

La siguiente función es la que se encarga de guardar los datos en la tarjeta MicroSD, además para evitar problemas, se ha añadido un símbolo de “\$” para indicar cuando se inicia una trama de datos, así como el símbolo “#” para indicar el fin de la trama, si una de las tramas carece de uno de estos símbolos, significa que está corrupta y algo ha salido mal, por lo que debemos descartarla en el procesamiento de los datos.

Los datos están separados mediante comas, de esta forma copiando todo el fichero se pueden exportar rápidamente a Excel y allí procesar estos datos y obtener conclusiones.

Los datos que se almacenan son: segundos desde el inicio del programa, humedad, temperatura interna_1, temperatura externa, temperatura interna_2, presión, aceleración en el eje X, aceleración en el eje Y, aceleración en el eje Z, orientación en el eje X, orientación en el eje Y, orientación en el eje Z, latitud, longitud, fecha, hora, altitud, segundos del diferencial GPS, rumbo y velocidad.

```
void printSDcard(){
    logfile.print("$"); logfile.print(",");
    logfile.print(millis()/1000); logfile.print(",");
    logfile.print(h); logfile.print(",");
    logfile.print(t); logfile.print(",");
    logfile.print(sensor.getTempCByIndex(0)); logfile.print(",");
    logfile.print(T,2); logfile.print(",");
    logfile.print(P,2); logfile.print(",");
    logfile.print(AcX); logfile.print(",");
    logfile.print(AcY); logfile.print(",");
    logfile.print(AcZ); logfile.print(",");
    logfile.print(GyX); logfile.print(",");
    logfile.print(GyY); logfile.print(",");
    logfile.print(GyZ); logfile.print(",");
    logfile.print(flat); logfile.print(",");
    logfile.print(flon); logfile.print(",");
    logfile.print(ldate); logfile.print(",");
    logfile.print(ltime); logfile.print(",");
    logfile.print(laltitude); logfile.print(",");
    logfile.print(lage); logfile.print(",");
    logfile.print(lcourse); logfile.print(",");
    logfile.print(lspeed); logfile.print(",");
    logfile.println("#");
    logfile.close();
}
```

A continuación, esta función se encarga de establecer el color del led dual en función de las necesidades, para ello, cada vez que se llama a esta función, se le establece un dato de entrada para poder hacer la selección entre rojo, verde o apagado.

```
void dualLed(int x){
    if(x==1){ // verde seguro
        digitalWrite(LED_1, LOW);
        digitalWrite(LED_2,HIGH);
    }
    else if(x==2){ // rojo peligro
        digitalWrite(LED_1, HIGH);
        digitalWrite(LED_2,LOW);
    }
    else{
        digitalWrite(LED_1, LOW);
        digitalWrite(LED_2,LOW);
    }
}
```

Esta función se encarga de la lectura de los datos de la tarjeta SD.

Es la función más larga de todas, por lo que la iremos comentando por apartados.

```
void lecturaSd(){  
    if(Serial.available() > 0){  
        dato = Serial.read();  
    }
```

Lo primero que se hace es comprobar si existe algún dato en el serial 0, es decir, si existe comunicación entre el PC y el Arduino. Si además se está enviando algún tipo de información, entonces guarda la tecla pulsada como la variable dato.

```
if( dato == '1'){  
    Serial.println("-----");  
    Serial.println("");  
    Serial.println("OPCIÓN 1: LECTURA DE DATOS");  
    Serial.println("");  
    Serial.flush();  
    delay(2000);  
    tamano=0;  
    indice=0;  
    while(indice<=15){  
        sprintf(filename, "metsat%02d.txt", indice);  
        logFile = SD.open(filename);  
        Serial.print("fichero");  
        Serial.println(indice);  
        Serial.println("");  
        tamano=tamano+logFile.size();  
        Serial.print("tamaño: ");  
        Serial.print(logFile.size());  
        Serial.println(" bytes.");  
  
        if (logFile) {  
            while (logFile.available()) {  
                Serial.write(logFile.read());  
            }  
            logFile.close();  
        }
```

En caso de que la tecla pulsada sea “1”, entonces se procede a leer los datos ya almacenados en la tarjeta SD, para ello va abriendo y cerrando cada uno de los ficheros y volcando todo su contenido a través del monitor serial.

Además, también se va indicando el tamaño de cada uno de los ficheros.

Una vez termina, cierra el fichero para evitar la pérdida de datos.

A continuación, si el fichero no se ha podido abrir correctamente, entonces se indicará que existe un error y se intentará abrir el siguiente fichero.

Finalmente, se mostrará la cantidad de datos almacenados en la SD.

```
else
{
    Serial.println("Error al abrir el archivo");
}
indice++;
}
Serial.print("tamaño: ");
Serial.print(float(tamano/1000));
Serial.println(" Kb.");
Serial.print("tamaño: ");
Serial.print(tamano/1000000);
Serial.println(" Mb.");
Serial.print("tamaño: ");
Serial.print(long(tamano/1000000000));
Serial.println(" Gb.");
dato = 0;
Serial.println("-----")
}
```

En caso de pulsar la tecla “2”, se procederá al borrado de los datos almacenados en la SD y se establecerá el contador a 0, este proceso se debe seguir cada vez que se quiera iniciar el programa.

```
else if(dato=='2'){
    Serial.println("-----")
    Serial.println("");
    Serial.println("OPCIÓN 2: BORRADO DE DATOS");
    Serial.println("");
    Serial.println("");
    Serial.println(" ***--- ATENCIÓN ---*** ");
    Serial.println("Se procederá a borrar archivos.");
    Serial.println("");
    Serial.println("j: BORRAR ARCHIVOS.");
    Serial.println("a: CANCELAR.");
    Serial.flush();
    delay(1000);
    bool eleccion = false;
    while(eleccion!= true){
        if(Serial.available() > 0){
            dato = Serial.read();
        }
    }
}
```

Para evitar borrar datos por accidente, primero de todo se debe hacer una segunda comprobación de si estamos seguros de proceder con el borrado o si por el contrario preferimos cancelar la operación.

Si se pulsa “j”, se procede al borrado y si se pulsa “a”, se cancela la operación.

Al confirmar el borrado, entonces el contador “K”, se vuelve a poner a 0.

```
if(dato == 'j'){
  Serial.println("-----");
  Serial.println("");
  Serial.println("OPCIÓN j: COMIENZA BORRADO DE DATOS");
  Serial.println("");
  delay(3000);
  indice=0;
  while(indice<=15){
    sprintf(filename, "metsat%02d.txt", indice);
    Serial.print(filename);
    Serial.println(" borrado.");
    logFile.close();
    SD.remove(filename);
    delay(150);
    indice++;
  }
  k = 0;
  EEPROM.write(addr_k,k);
  eleccion=true;
  Serial.println("");
  Serial.println("DATOS BORRADOS");
  Serial.println("");
  delay(3000);
  Serial.println("-----");
}
else if (dato == 'a') {
  Serial.println("-----");
  Serial.println("");
  Serial.println("OPCIÓN a: CANCELADO EL BORRADO DE DATOS");
  Serial.println("");
  delay(1000);
  eleccion=true;
  Serial.println("-----");
}
}
Serial.println("-----");
}
```

Una tercera opción es si se pulsa la tecla “3”, esta opción es para realizar una extracción de la tarjeta SD de forma segura.

```
else if(dato == '3'){
  Serial.println("-----");
  Serial.println("");
  Serial.println("EXTRACCIÓN DE SD SEGURA");
  Serial.println("");
  dualLed(3);
  Serial.println("EXTRAER ANTES DE 20 segundos.");
  Serial.flush();
  dato=0;
  delay(20000);
  dualLed(LED_SD_CLOSE);
  Serial.println("-----");
}
}
```

La siguiente función se utiliza para hacer comprobaciones del funcionamiento del programa a través del monitor serial, de esta forma, se puede comprobar que es lo que se está guardando en la tarjeta SD y así podemos comprobar también el correcto funcionamiento de los sensores.

```
void printDebug(){  
    #ifdef DEBUG    // SOLO imprime si está el DEBUG activo  
        Serial.print("$"); Serial.print(",");  
        Serial.print(h); Serial.print(" DHTHUM, ");  
        Serial.print(t); Serial.print(" DHTTEMP, ");  
        Serial.print(sensor.getTempCByIndex(0)); Serial.print(" TEMPEXT, ");  
        Serial.print(T,2); Serial.print(" BMPTMP, ");  
        Serial.print(P,2); Serial.print(" BMPPRES, ");  
        Serial.print(AcX); Serial.print(" AX, ");  
        Serial.print(AcY); Serial.print(" AY, ");  
        Serial.print(AcZ); Serial.print(" AZ, ");  
        Serial.print(GyX); Serial.print(" GX, ");  
        Serial.print(GyY); Serial.print(" GY, ");  
        Serial.print(GyZ); Serial.println(" GZ, ");  
        Serial.print(flat); Serial.print(" LAT, ");  
        Serial.print(flon); Serial.print(" LON, ");  
        Serial.print(ldate); Serial.print(" DATE, ");  
        Serial.print(ltime); Serial.print(" TIME, ");  
        Serial.print(laltitude); Serial.print(" ALTITIDE, ");  
        Serial.print(lage); Serial.print(" LAGE, ");  
        Serial.print(lcourse); Serial.print(" COURSE, ");  
        Serial.print(lspeed); Serial.print(" SPEED, ");  
        Serial.println("#");  
    #endif  
}
```

La última función que se expone es la de obtención de datos del GPS, para ello el programa procede a la lectura de datos en el serial 2 y empieza la lectura de las tramas, en caso de obtener una trama completa, la decodifica para obtener toda la información que facilita, y mediante otras funciones de la librería, se procede a almacenar los datos que nos interesan en las diferentes variables.



```

void datosGPS(){
bool newData = false;
unsigned long chars;

// For one second we parse GPS data and report some key values
for (unsigned long start = millis(); millis() - start < 1000;)
{
    while (Serial2.available())
    {
        char c = Serial2.read();
        //Serial.write(c); // uncomment this line if you want to see the GPS data flowing
        if (gps.encode(c)) // Did a new valid sentence come in?
            newData = true;
    }
}

if (newData)
{
    gps.f_get_position(&flat, &flon, &age);

#ifdef DEBUG
    Serial.print("LAT=");
    Serial.print(flat == TinyGPS::GPS_INVALID_F_ANGLE ? 0.0 : flat, 6);
    Serial.print(" LON=");
    Serial.print(flon == TinyGPS::GPS_INVALID_F_ANGLE ? 0.0 : flon, 6);
    Serial.print(" SAT=");
    Serial.print(gps.satellites() == TinyGPS::GPS_INVALID_SATELLITES ? 0 : gps.satellites());
    Serial.print(" PREC=");
    Serial.print(gps.hdop() == TinyGPS::GPS_INVALID_HDOP ? 0 : gps.hdop());
#endif

    gps.get_datetime(&ldate, &ltime, &lage);
    laltitude = gps.altitude();
    lcourse = gps.course();
    lspeed = gps.speed();
}
}

```

Una vez finalizada la exposición de las funciones, se procede a exponer una última línea antes del void loop().

Esta línea podría ir escrita en el principio del programa, pero para hacer pruebas del funcionamiento del programa, si se baja mucho el tiempo de escrituras, se puede producir un error, colocando esta línea aquí, se compensa el desfase de tiempo desde que se inicia el programa hasta que empieza con las escrituras.

Lo que hace esta línea es declarar la variable *antTime* e inicializarla con el tiempo de ejecución justo antes de comenzar el void loop().

```
unsigned long antTime = millis();
```

El *void loop* es la parte del código que se repetirá de forma cíclica a lo largo del programa, este comienza con una comprobación para determinar si el contador se encuentra entre 2 y 13, en caso de que sea afirmativo, entonces pone en silencio el *buzzer* y apaga el led para ahorrar batería durante el vuelo.

```
void loop() {  
    if(k>2 && k<13){  
        digitalWrite(BUZZ, LOW);  
        dualLed(3);  
    }  
    else{  
        digitalWrite(BUZZ, HIGH);  
        dualLed(LED_SD_OPEN);  
        delay(25);  
    }  
  
    while (digitalRead(PUL_2) == HIGH){  
        digitalWrite(BUZZ, LOW);  
        dualLed(LED_SD_CLOSE);  
        lecturaSd();  
        antTime = millis();  
    }  
  
    readTempHumedad(); // DHT11 Y DS18B20  
  
    readBMP(); // BMP180  
  
    readAccGyro(); //MPU  
    digitalWrite(BUZZ, LOW);  
  
    #ifdef DEBUG  
        Serial.print("eeprom: ");  
        Serial.println(EEPROM.read(addr_k));  
  
        printDebug();  
    #endif  
  
    datosGPS(); // GPS
```

En caso de que el contador sea menor de 2 o mayor de 13, entonces significa que está a punto de ser lanzado, se encuentra en los primeros minutos del vuelo o bien está a punto de aterrizar, por ello en este caso, se enciende de forma intermitente el *buzzer* y el led se enciende y se pone de color rojo, esto indica que no debe ser manipulada la tarjeta SD.

Lo siguiente que nos encontramos es un `while()`, este comprueba si el interruptor está en posición de escritura o lectura, si se pone en modo lectura, entonces se apaga el *buzzer* y el led se pone de color verde, después se ejecutará la función `lecturaSD()` hasta que el interruptor vuelva a la posición de escritura.

Al volver a la posición de escritura, volverá a pitar intermitentemente el *buzzer* y el led volverá a rojo, luego seguirá ejecutando el código que vamos a explicar a continuación.

Las siguientes líneas son llamadas a funciones para obtener los valores de los sensores y del GPS, además si estamos en modo *debug*, veremos el valor de los sensores por el serial y también veremos el contador almacenado en la memoria EEPROM.

Las siguientes líneas del void loop() son las de la imagen siguiente, en ellas primero de todo se comprueba que el tiempo transcurrido es menor de 10 minutos, si esta condición se cumple, entonces procede a la escritura de los datos obtenidos en la tarjeta SD, en caso de que la condición no se cumpla y el tiempo sea mayor de 10 minutos, entonces se aumenta el contador, se guarda este en la EEPROM y se pone el contador del tiempo a 0, de esta manera, cada 10 minutos se abrirá un fichero nuevo, esto se repetirá hasta un máximo de 15 veces, por lo que este bucle funcionará durante unos 150 minutos.

Se toman estos 150 minutos como el tiempo de vuelo aproximado, la idea es que cuando la capsula llegue al suelo, todos los ficheros estén cerrados y no se produzca ninguna perdida de datos por el impacto.

```
//guardarDatosSd(); // SE ESCRIBEN VALORES EN SD Y SE CREAN FICHEROS SI PROCEDE
if ((millis() - antTime) < TIME_FILE){ // cada 10 minutos cambia de indice

    sprintf(filename, "metsat%02d.txt", k);
    logFile = SD.open(filename, FILE_WRITE);
    if (logFile) {
        printSDcard();
    }else {
        #ifdef DEBUG
            Serial.println("Error al abrir el archivo");
        #endif
    }
}
else {

    antTime = millis();
    k++; // incrementa el indice del fichero
    EEPROM.write(addr_k,k);
}
```

Por último, si el contador es mayor a 15, significa que se ha superado el tiempo de vuelo establecido, en ese caso se entra dentro de un if() en el que se pone el led en verde indicando que la tarjeta no se está utilizando, luego entra en un While(), durante el cual el *buzzer* estará pitando de forma continua hasta que se gaste la batería o hasta que se encuentre y se proceda a la extracción de la SD.

En caso de que se estén realizando pruebas de funcionamiento, en este While(), se hace una llamada a la función lecturaSD(), esto es importante puesto que nos permite leer valores, borrar la memoria o como en este caso, se ha añadido una función que sería la opción “4”, esta simplemente pone a 0 la memoria EEPROM y reinicia el programa sin borrar los datos.

Esto es importante, puesto que, si llegamos a los 150 minutos de ejecución, nos quedaremos en el modo seguro, aunque reiniciemos el Arduino, puesto que debido a la memoria EEPROM, seguirá entendiendo que estamos en el final del programa.

Finalmente si seleccionamos la opción “4”, pasaremos por un break, que nos sacará del while() y podremos volver a empezar con el programa.

```
if(15 < k){ // si 15*10 = 150 minutos deja de tomar datos para proteger la SD
  digitalWrite(LED_SD_CLOSE);
  Serial.println("FIN DE ESCRITURAS");
  delay(1000);
  while(1){
    digitalWrite(BUZZ,HIGH);
    lecturaSd();
    if(Serial.available() > 0){
      dato = Serial.read();
    }
    if(dato == '4'){
      Serial.println("");
      Serial.println("REINICIANDO LOOP");
      Serial.println("");
      Serial.flush();
      dato=0;
      break;
    }
  }
}
```

Es importante recordar que la única forma de parar la escritura en la SD es cambiando la posición del interruptor al modo seguro de lectura, esto lo interrumpe sin riesgo de pérdida de datos.

4.2.5 PROCESAMIENTO DE DATOS

Una vez que el programa comience a funcionar y obtengamos los primeros datos, estos serán cargados en un Excel, para poder realizar algunos cálculos y obtener gráficas.

Las diferentes tramas, finalmente quedarán almacenadas en unos documentos .txt, los cuales irán tomando nombres desde METSAT00, hasta METSAT15.

Si abrimos uno de estos archivos, podremos ver que el resultado sería algo similar a esto.

```
$,398,35.00,28.00,18.69,26.86,935.07,1476,-904,-16644,-317,69,0,39.66,-3.87,280220,9410400,75600,147,16110,6,#
$,400,35.00,28.00,18.62,26.86,935.07,1336,-980,-16612,-239,123,0,39.66,-3.87,280220,9410500,75600,494,16227,4,#
$,402,35.00,28.00,18.69,26.88,935.08,1376,-852,-16724,-299,123,0,39.66,-3.87,280220,9410700,75600,670,16410,5,#
```

Figura 102 Datos MicroSD

Para poder cargar estos datos en Excel, deberemos seleccionar todo el texto y copiarlo.

Posteriormente abrimos una nueva hoja de Excel y seguimos los siguientes pasos. Primero de todo debemos pulsar sobre pegar, que se encuentra en la barra de inicio, cuando aparezca el desplegable de abajo, seleccionaremos la opción de “Usar el Asistente para importar texto...”

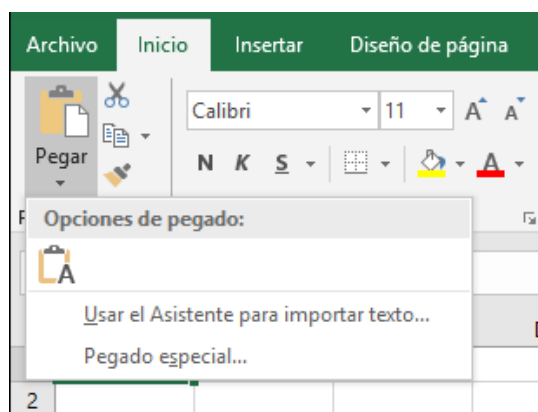


Figura 103 Pegado Excel

Tras pulsar sobre esta opción nos aparecerá la siguiente ventana:

Figura 104 Delimitación de datos

En la ventana anterior debemos seleccionar la opción de “Delimitados”, posteriormente pulsamos sobre “Siguiente”.

Figura 105 Separación por comas

En la siguiente ventana, seleccionamos la opción de “Coma” para poder establecer nuestro separador, luego pulsaremos en “Siguiente”.

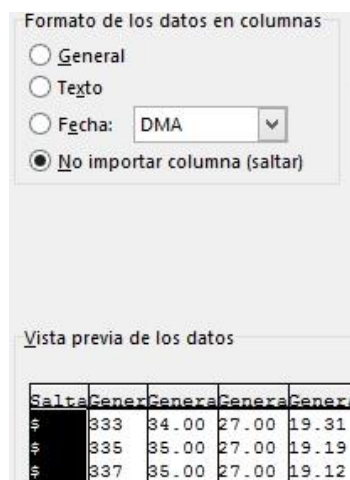


Figura 106 Importación de columnas

A continuación, pincharemos sobre las columnas correspondientes a “\$” y “#” y en la opción de formato de datos, pulsaremos la opción de “No importar columna (SALTAR)”, de esta forma estas columnas no se exportarán. Finalmente pulsamos sobre “Finalizar” y ya tendremos nuestros datos exportados.

	A	B	C	D	E	F	G	H
1	333	34	27	19.31	26.19	935.07	1232	-984
2	335	35	27	19.19	26.21	935.09	1416	-948
3	337	35	27	19.12	26.24	935.07	1364	-1032

Figura 107 Columnas Excel

Lo ideal es crear una plantilla en Excel, de forma que, al exportar los datos, todo quede más claro, por ejemplo, indicando en cada columna que dato representa o incluyendo nuevas columnas con cálculos derivados de los datos obtenidos.

Por ejemplo, se podría crear una columna que muestre la diferencia de temperaturas entre el interior y el exterior.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	ESTUDIO DE RESULTADOS (METSAT-1)													
2		DHT	DHT	DS18	BMP	BMP	ACELERACIÓN			GRAVEDAD				
3	millis	HUM	TEM	TEMP	TEMP	PRES	A (X)	A (Y)	A (Z)	G (X)	G (Y)	G (Z)	TIEMPO (s)	Media temp int
4													0	0
5													0	0

Figura 108 Procesado de datos en Excel

Otras opciones pueden ser el cálculo de la altura a través de los datos de presión y de temperatura. Estos cálculos se realizan a través de las fórmulas de la Atmósfera Estándar Internacional (ISA).

En caso de que estemos usando Calc, si queremos pegar los datos, deberemos pinchar sobre la que será nuestra primera casilla, después pegaremos allí los datos y nos aparecerá la siguiente ventana:

Figura 109 Pegado de datos Calc

En esta ventana, debemos seleccionar la separación por comas y en el apartado de campos, deberemos pulsar sobre los que no nos interesen y deberemos marcarlos como “Ocultar”.

Finalmente pulsaremos sobre “Aceptar” y ya tendríamos los datos dispuestos en las casillas.

4.3 SUBSISTEMA DE VIDEO

Respecto a la toma de datos, uno de los puntos más importantes es la adquisición de imágenes y su almacenamiento, por ello he desarrollado este punto de forma independiente.

A diferencia del resto de componentes, las cámaras van situadas en la parte inferior de la sonda, por lo que están más expuestas al impacto final y además también van expuestas a las inclemencias meteorológicas, puesto que las lentes deberán ir en el exterior, ajustadas en sus orificios de la sonda y no puede haber otro tipo de aislamiento, puesto que si se utiliza un metacrilato para separar la lente del exterior, se podría crear condensación y que las imágenes sean de muy mala calidad.

En esta aplicación se han utilizado dos cámaras, la intención es la de probar dos modelos de cámaras y así probar cuál de estos modelos debe ser el definitivo para así utilizarlo como solución final en el próximo vuelo.

GOPRO

El primer modelo utilizado, ha sido el modelo GOPRO HERO, en concreto es un modelo de cámara con batería integrada, sin pantalla posterior y con la carcasa integrada, es decir, no se puede sacar de esta.

A la hora de comprar la tarjeta de memoria, se debe tener en cuenta que esta cámara es compatible con *MicroSD*, *MicroSDHC* y *MicroSDXC* con una capacidad máxima de 32 (Gb).

Respecto a las condiciones de almacenamiento de la batería, es aconsejable que antes de que la cámara empiece a funcionar, esta se mantenga en un lugar caliente, puesto que las temperaturas extremas afectan drásticamente a la duración de la batería.

Antes del lanzamiento se han realizado varias pruebas de la duración de la batería en diferentes condiciones de temperatura.



Figura 110 GOPRO HERO

En total se han realizado tres pruebas, la primera de ellas fue en condiciones normales a una temperatura aproximada de 19°C, posteriormente se realiza otra prueba a unos 4°C y por último una prueba a -23°C, el objeto de estas pruebas es determinar la duración de la batería a estas temperaturas y también determinar la temperatura de funcionamiento de la electrónica interna, ya que esta temperatura ayudará a crear el primer escudo para aislar térmicamente el interior de la sonda.

Las pruebas a temperatura ambiente determinaron una duración de la batería de 2 h 15 min.

Para una temperatura de 3°C, la duración de la batería resultó ser de 2 h 26 min.

Finalmente, para una temperatura de -24°C, la duración de la batería fue de 1 h 43 min.

La duración estimada del vuelo, ronda entre las 2 horas y 1 hora y 45 minutos.

Es importante tener en cuenta, que en los primeros minutos, la temperatura será superior a los 10°C, posteriormente esta irá descendiendo hasta que llegue a los -50°C.

En la prueba de -24°C, la cámara fue introducida desde el primer momento a un congelador a esa temperatura y esta fue constante hasta el final del experimento, por lo que la estimación es que la duración de la batería sea algo superior a la de la tercera prueba.

Aproximadamente 2 horas de grabación.

Un dato muy importante es que la mayor duración de la batería se alcanza a los 3°C, por lo que conforme la sonda empieza a ascender, la eficiencia de la batería ira en aumento, hasta que al final alcance los 0°C y la eficiencia llegue de nuevo a un punto de inflexión en el que la duración de la batería irá disminuyendo.

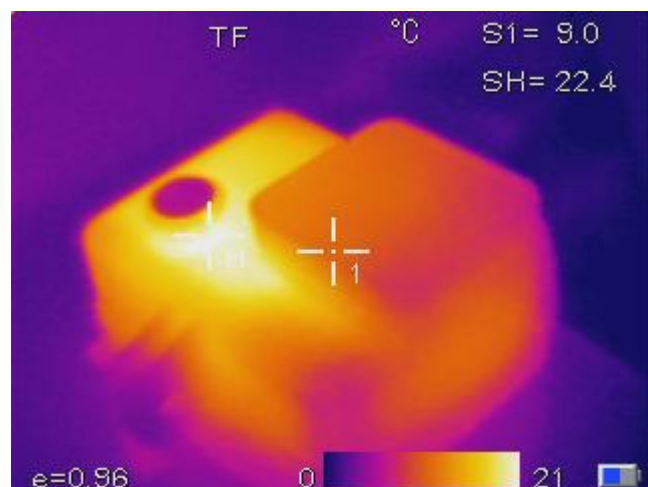


Figura 111 GOPRO en nevera

Con la imagen termográfica tomada en la prueba a 3°C, se puede observar como la temperatura máxima es de unos 22,4°C, este dato es muy importante puesto que la capsula solo tendrá dos aberturas y las cámaras mismas serán utilizadas como primera barrera contra el frio.

En la imagen se observa como la parte superior es la zona más caliente, mientras que la inferior es la más fría, esto es debido a que la parte inferior está en contacto directo con la superficie de la nevera y esto hace que el calor se pierda a través de conducción, mientras que en la parte superior el intercambio de calor se produce mediante convección, este último será el tipo de intercambio al que estará expuesta la cámara, es muy importante puesto que el intercambio se produce mucho más pausadamente que si se produjese conducción.

CAMARA ACCIÓN MODELO GENÉRICO

Aunque el modelo Gopro HERO, parece una solución muy robusta y fiable, he decidido probar con un modelo genérico de muy bajo precio, este modelo admite tarjetas *MicroSD* de hasta 32 (Gb), posee una pantalla trasera, posee una ranura para introducir una batería de 1.000 (mAh) y además puede ser montada en una carcasa de metacrilato si se va a sumergir.

En este caso, la cámara va a ser enviada sin su carcasa, puesto que es bastante pesada y se tiene que aligerar el peso en la medida de lo posible.

En este caso, las pruebas realizadas han sido dos, la primera fue a temperatura ambiente y la segunda a -24°C.



Figura 112 SJCAM

En condiciones normales y temperatura ambiente, la duración de la batería es de aproximadamente 1 hora 30 minutos.

Sin embargo, en condiciones de temperaturas extremas, la batería dura poco más de 30 minutos.

A pesar de sus pocas prestaciones en lo que a duración de la batería se refiere, siempre cabe la posibilidad de añadir una batería de mayor duración y además es una cámara muy ligera y finalmente se decidió usar por este motivo, ya que podría

servir como alternativa para grabar los primeros minutos de vuelo en caso de que la cámara principal falle.

De esta forma se establece la duplicidad de equipos en la grabación de video.

4.4 SUBSISTEMA DE POTENCIA

En este apartado se va a especificar qué tipo de fuentes de alimentación han sido utilizadas, así como que tipo de baterías se han utilizado.

4.4.1 BATERÍAS

Debido a las bajas temperaturas en las capas altas de la atmósfera, muchas de las baterías de litio que se venden, o que vienen integradas en los aparatos, no son las indicadas para este tipo de aplicaciones.

Las bajas temperaturas no solo limitan la duración de la batería, sino que además provocan que estas dejen de funcionar repentinamente, otro tipo de baterías como las alcalinas, tampoco están optimizadas para bajas temperaturas.

Un *powerbank*⁶, en un principio podría ser una buena opción puesto que son de bajo precio, poseen una gran capacidad y proveen de varias fuentes estabilizadas. Sin embargo, muchos de los aparatos a utilizar, se alimentan a diferentes voltajes y al igual que en los casos anteriores, los *powerbank*, pueden fallar en bajas temperaturas.

Finalmente, después de documentarme y hacer algunos cálculos, decidí utilizar unas pilas de litio especiales para trabajar a bajas temperaturas.

Este tipo de baterías son desarrolladas por diferentes fabricantes, para poder hacer una buena elección, es necesario recopilar los *datasheet* de los fabricantes y hacer una comparación objetiva entre las diferentes opciones.

La forma de hacer una comparación objetiva se va a realizar a través de la siguiente operación:

⁶ Powerbank: Powerbank es una batería externa portátil que sirve para cargar diferentes dispositivos electrónicos mediante un cable que tenga conexión USB.

$$\text{Voltaje}(V) * \frac{\text{Capacidad}(A)}{\text{Peso}(g)} = \text{Densidad de energía} \left(\frac{Wh}{g} \right)$$

De esta forma vamos a comparar los dos modelos seleccionados y posteriormente lo vamos a comparar con un *powerbank* comercial para ver cuál es la mejor opción.

El primer modelo para comparar va a ser el modelo Energizer L91 AA:

$$1,5 (V) * \frac{3(A)}{15(g)} = 0,3 \left(\frac{Wh}{g} \right)$$

El segundo modelo es VARTA 6106 AA:

$$1,5 (V) * \frac{2,9 (A)}{15(g)} = 0,29 \left(\frac{Wh}{g} \right)$$

Por último, vamos a realizar el cálculo para un *powerbank*:

$$5 (V) * \frac{10(A)}{291(g)} = 0,17 \left(\frac{Wh}{g} \right)$$

Como podemos ver, la diferencia entre los dos primeros modelos es mínima, mientras que el *powerbank*, es el que obtiene el peor resultado.

Para completar aún más el estudio, voy a utilizar el modelo de VARTA AAA:

$$1,5 (V) * \frac{1,1 (A)}{7,7 (g)} = 0,21 \left(\frac{Wh}{g} \right)$$

En esta aplicación debido a su precio, hemos seleccionado el modelo VARTA AA, aunque los dos expuestos son de calidades similares, para el caso de AAA, considero que no interesa a no ser que la aplicación deba ser muy ligera en lo que a peso se refiere y que sea de poca duración y bajo consumo.

En concreto el modelo VARTA AA, funciona en condiciones de temperaturas extremas, entre -40°C y 60°C, lo cual las hace ideales para funcionar en la estratosfera.

Otro factor a tener en cuenta son los picos de consumo, de forma continua un aparato de radio consume una potencia determinada, pero a la hora de emitir, este consumo puede aumentar mucho, si las baterías utilizadas no son capaces de ofrecer estos picos de consumo, entonces nuestra electrónica podría no funcionar.

Este modelo, permite hasta 880 pulsos de 1.000 (mAh), por lo que es un buen valor, ya que nuestros aparatos funcionan por debajo de los 5 (V) y no consumen mucho, por lo que este valor no será rebasado.



Figura 113 Varta Litio

En estas gráficas podemos ver dos datos muy importantes sobre las baterías, la primera de ellas muestra cual es la curva de descarga para diferentes consumos, mientras que la segunda muestra también para diferentes consumos como la temperatura afecta a la capacidad.

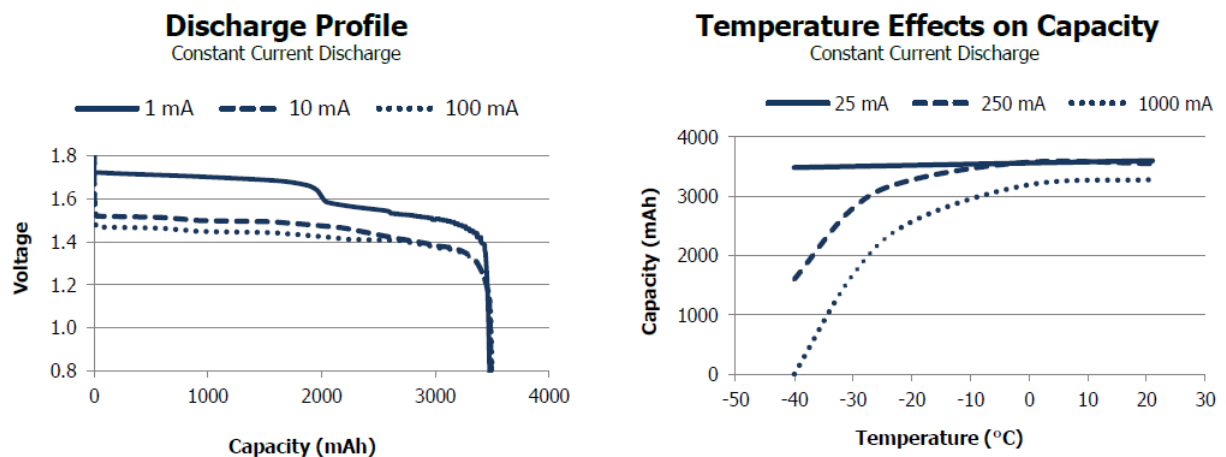


Figura 114 Descarga pilas de Litio

En la primera gráfica, vemos como en el tramo inicial se produce una gran caída de potencial desde los 1,8 (V) hasta los 1,5 (V), a partir de entonces para consumos comprendidos entre los 10 y los 100 (mAh), la caída es mínima y bastante lineal, una vez llegue a los 1,4 (V) se produce una caída en picado y se termina la vida útil de la batería.

En la segunda gráfica, vemos que para el rango de temperaturas comprendido entre los -10°C y los 20°C, apenas existe variación para consumos entre los 25 y los 250 (mAh).

Sin embargo, al disminuir la temperatura, para consumos cercanos a los 250 (mAh), existe una gran pérdida de capacidad, en comparación con consumos de 25 (mAh).

Por ello se debe intentar no llegar a temperaturas inferiores de -20°C .

4.4.2 FUENTES DE ALIMENTACIÓN

Una vez seleccionadas las baterías a utilizar, ya podemos proceder al estudio de cómo se va a alimentar a cada componente electrónico.

Para ello se tiene que consultar el *datasheet* del fabricante para conocer rangos de voltaje admitidos, en nuestra aplicación disponemos de un total de 6 dispositivos electrónicos, algunos de ellos utilizarán la batería del fabricante, otros se tendrán que modificar para quitar la batería del fabricante y añadir una nueva y por último otros dispositivos carecen de baterías y necesitan una adaptada a sus necesidades.

Vamos a ir enumerando los diferentes dispositivos y comentando la solución utilizada.

GOPRO HERO:

Este modelo en concreto incluye una batería que no es extraíble, sin embargo, tras las pruebas realizadas, el funcionamiento de esta batería se puede considerar como muy bueno, por lo que se utilizará la batería de serie.

CAMARA SJCAM:

Este modelo de cámara, incluye una batería que puede ser extraída, la opción inicial era la de utilizar varias pilas de litio para hacer una batería de mayor capacidad, pero el problema que presenta es que la batería del fabricante posee tres pines, uno de ellos es el polo positivo, el otro es tierra y por último el pin central es para conectar el sensor interno de temperatura de la batería, en caso de que este no esté presente, la cámara se encenderá pero enseguida se apaga al detectar la ausencia del sensor. Por este motivo, finalmente se decide usar la batería del fabricante.

TKSTAR GPS TRACKER:

Este es uno de los dos dispositivos más importantes en la sonda, puesto que pertenece al subsistema de comunicaciones y sin la recuperación de la sonda, no se puede determinar que la misión haya terminado de forma satisfactoria, por ello

este dispositivo será modificado respecto a su acabado de fábrica, con la intención de mejorar sus prestaciones.

Para ello lo primero que se hará, es quitar la carcasa negra de plástico que protege la electrónica.

Una vez realizado este paso, se podrá ver que la placa lleva pegada una batería de litio, deberemos despegarla de la placa, dejando los cables soldados y sin dañar la placa.

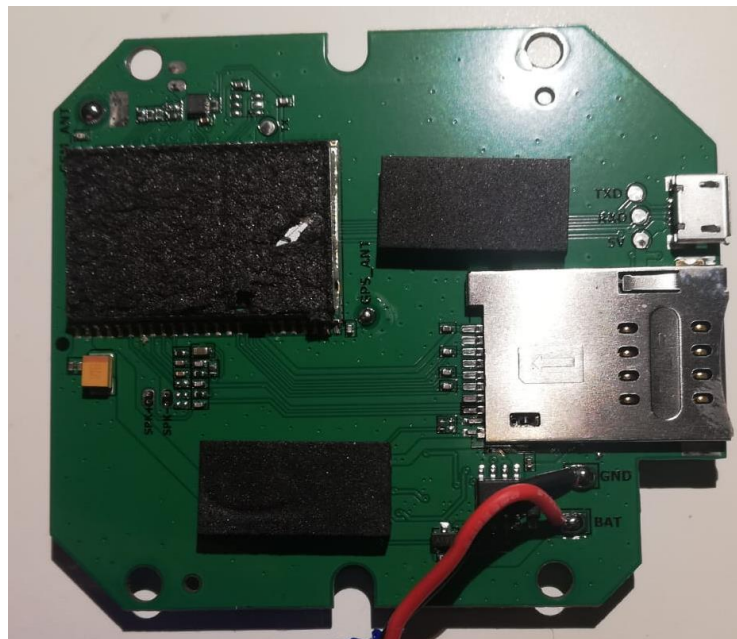


Figura 115 PCB GPS tracker

Este será el resultado obtenido, ahora en la parte inferior derecha podemos ver dos cables, uno negro y otro rojo, debemos desoldarlos para posteriormente añadir una nueva batería.

Una vez retirada la batería del fabricante, debemos examinarla para fabricar una similar, con la intención de no dañar la electrónica.

Así pues, se comprueba que la batería del fabricante es de 5.000 (mAh), de litio, con un voltaje de 3,7 (V) y 164 (g).

Para conseguir una batería equivalente, se podrían usar dos bloques en paralelo con 4 pilas en serie, de esta forma, se obtendría una capacidad de 5.800 (mAh) y un voltaje de 6 voltios, para que esta batería se pueda usar, sería necesario utilizar una fuente de alimentación que reduzca a 3,7 (V), pero esta fuente de alimentación,

aunque es capaz de reducir, presenta pérdidas que pueden ir de entre 0,5 hasta 1 (V), pero además para que un grupo de pilas no consuma del otro, se deben añadir dos diodos a la salida de cada grupo de pilas, estos son de silicio y tienen una caída de voltaje aproximado de 0,7 (V).

En resumen:

$$6(V) - 0,7(V) - 1(V) = 4,3 (V)$$

Es fundamental tener en cuenta que, aunque la batería sea de 3,7 (V), realmente cuando está cargada al 100%, su voltaje es de 4,1 (V), esto deja este sistema un poco justo, además presenta el riesgo de los diodos, si uno falla debido a las temperaturas, la fuente cambiará su voltaje de entrada y apagará el dispositivo.

Una manera de solucionar el problema del voltaje sería añadir una pila más en serie para cada grupo de baterías, sumando un total de 10 pilas, lo que haría que la batería pese unos 150 (g), más los diodos, la fuente conmutada y los porta baterías. Fácilmente se superan los 164 (g) de la batería original.

Por lo que la solución debe ser otra, primero de todo es necesario reducir el peso de la batería, por otro lado, deben eliminarse posibles fuentes de problemas como podrían ser los grupos de baterías en paralelo, reducir las pérdidas entre las baterías y la carga, pero a la vez es necesario que la duración de la batería sea lo suficiente buena como para pasar varias horas en *standby*.

Primero de todo, para la solución final, se van a eliminar los paquetes en paralelo, todas las pilas van a estar en serie, de esta forma evitamos posibles errores y perdidas de energía entre baterías y carga.

En principio, se pueden usar 4 pilas en serie, de esta forma, se obtienen 6 (V), con este voltaje es suficiente para poder usar una fuente conmutada para regularla a 4,1 (V). Sin embargo, ahora disponemos de 2.900 (mA) menos y reduce la batería bastante, para compensar un poco esto, vamos a empezar a hablar en términos de energía.

En el caso anterior, se quería usar dos grupos en paralelo de 4 pilas en serie, si tenemos en cuenta el siguiente calculo:

$$6(V) * 5,8(A) * t(h) = 34,8 (Wh)$$

Esta es la energía disponible, sin embargo, hemos de eliminar la energía que se pierde en cada diodo, así como la que se pierde en la fuente conmutada.

Así pues, para cada grupo de pilas, la energía que realmente va a dar es:

$$(6(V) - 0,7(V)) * 2,9(A) * t(h) = 15,37 \text{ (Wh)}$$

Y en total, los dos grupos dan un total de:

$$((6(V) - 0,7(V)) * 2,9(A)) * 2 * t(h) = 30,7 \text{ (Wh)}$$

Si ahora a este valor le descontamos la pérdida de energía en la fuente conmutada, nos encontramos con que la energía realmente disponible para el GPS es de:

$$(30,7(W) - ((1(V) * 5,8(A))) * t(h) = 24,9 \text{ (Wh)}$$

Si ahora hacemos los mismos cálculos para el sistema de 4 pilas en serie, la energía disponible sería de:

$$((6(V) - 1(V)) * 2,9(A)) * t(h) = 17,4 \text{ (Wh)}$$

Si en vez de utilizar 4 pilas, usamos 6 pilas, entonces la energía disponible será de:

$$((9(V) - 1(V)) * 2,9(A)) * t(h) = 23,2 \text{ (Wh)}$$

Ahora, la energía es la misma, aunque no dispongamos de la misma capacidad, hemos aumentado el voltaje y debemos tener en cuenta que la fuente conmutada mantiene el voltaje estabilizado a la salida, por lo que, si tenemos en cuenta la curva de descarga de las pilas, podemos ver que lo que estamos haciendo es aprovechar esta curva para aumentar la duración de la batería con la misma capacidad.

Como la capacidad por sí ya es de 2.900 (mA), hemos de tener en cuenta que, con el aumento del voltaje, esto está más que compensado, por lo que podemos prescindir de una pila y usar solo 5 pilas, de esta forma conseguimos disminuir aún más el peso y compensar la duración de la batería.

Así que finalmente se utilizarán 5 pilas en serie, estas pesarán 75 (g), por lo que ahorramos en 89 (g) respecto a la batería del fabricante.

A continuación, la fuente conmutada que se va a usar va a ser este modelo:

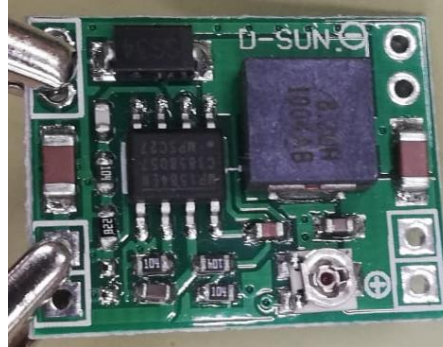


Figura 116 Fuente conmutada

Para funcionar tiene que alimentarse por encima de los 4,5 (V), y tiene una caída de potencial de unos 1,2 (V)

La conexión viene descrita en la cara inferior del PCB, indicando entrada y salida, así como positivo y negativo.

Para montar las pilas, he optado por dos porta baterías de 3 pilas en serie, en uno de ellos, he de realizar una modificación para poner en cortocircuito una de las 3 pilas.

Una vez que los tengo listos, los pego para hacer un solo bloque dándole así más consistencia y asegurándome de que no se rompen los empalmes de los cables debido a los golpes.

A la hora de realizar el empalme entre los dos porta baterías, es importante tener en cuenta que el rojo de uno de los porta baterías debe ir a la fuente conmutada, al igual que el cable negro del otro porta baterías.

Los otros dos cables, rojo y negro, deben empalmarse.

Una vez realizados estos pasos, se deben introducir las pilas y ajustar la salida de la fuente conmutada a 4,1 (V), de esta forma nos aseguraremos de no quemar la electrónica. Para realizar el ajuste, se debe utilizar un destornillador de estrella de precisión, para ello se debe ajustar mediante el *trimmer* de ajuste fino que queda junto a los dos condensadores pequeños.

Ahora debemos retirar los imanes de la carcasa si todavía no se ha hecho, luego aprovechando uno de los huecos para los imanes, se debe realizar un agujero en la carcasa para introducir los cables de alimentación, es importante realizar este paso, puesto que la carcasa lleva pegada una antena de telefonía y la placa no funciona si se utiliza sin esta.

Finalmente, ya se puede soldar un cable negro de la salida de la fuente conmutada hasta el *pad* negativo de la batería en el PCB del GPS. Haremos lo mismo, pero con el cable rojo.



Figura 117 Batería GPS tracker

Finalmente, se han realizado unas pruebas para determinar consumos del dispositivo, el consumo de este en *standby* es de 90 (mAh).

Además, tiene picos de 220 (mAh) en el establecimiento de llamada.

Por lo que en *standby*, la batería dura unas 32 horas aproximadamente, sin embargo, se debe tener en cuenta que se van a hacer muchas llamadas y el consumo medio lo vamos a obtener haciendo la media entre el consumo en *standby* y el consumo en picos.

De forma que el consumo medio se establece a 155 (mAh), a este consumo le corresponden unas 18 horas aproximadamente, pero para ser aún más conservadores, vamos a reducir un 30% la duración de la batería, por lo que esta quedaría reducida a 12,6 horas, más que suficiente para la aplicación a desarrollar.

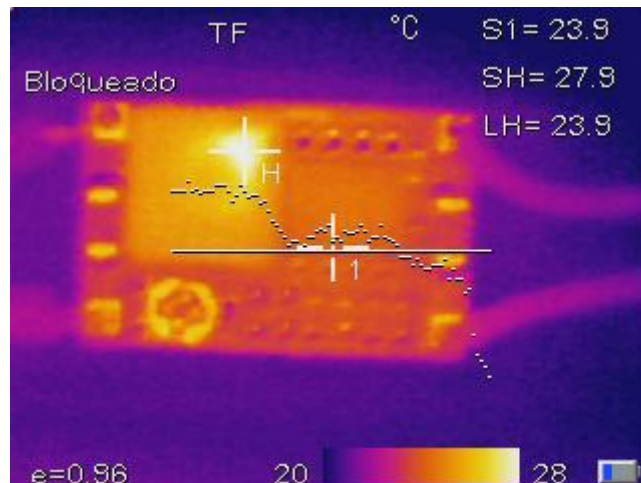


Figura 118 Térmica fuente conmutada

La cámara térmica indica una temperatura de funcionamiento a 23,9°C con picos de 27,9°C, este calor es una pérdida de energía, sin embargo, como tenemos suficiente batería, nos puede ser útil utilizar esta energía para hacer frente a las bajas temperaturas y conseguir que la fuente funcione correctamente.

ARDUINO MKR FOX 1200:

A continuación, vamos a crear las baterías para los dispositivos que no disponen de estas, primero voy a comenzar con la batería para el MKR FOX 1200, este es el dispositivo principal del Subsistema de comunicaciones.

En este caso, existen dos opciones disponibles para alimentarlo, la primera opción es hacerlo a 5 (V), y la otra opción es alimentarlo a 3,3 (V), aunque ambas opciones son posibles, el fabricante indica que lo más recomendable es alimentarlo a 3,3 (V), así que para ello se debe recurrir al bornero integrado en la placa.

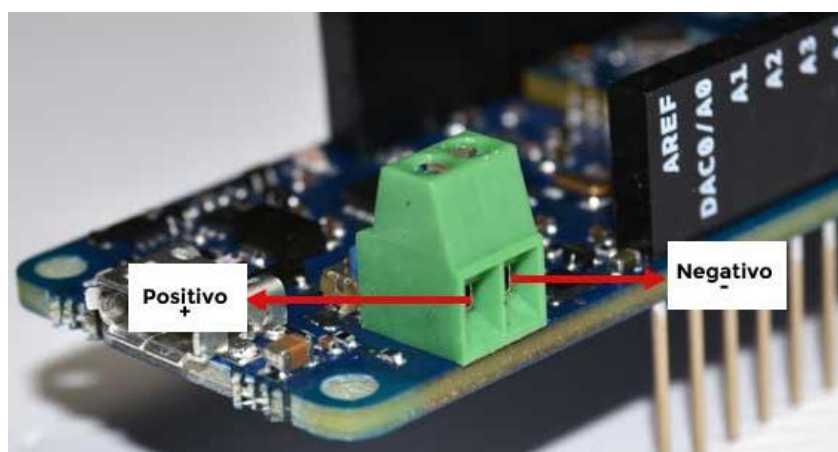


Figura 119 Bornas MKRFOX 1200

Es importante tener en cuenta que si se sobrepasan los 3,3 (V) quemaremos la placa y además esta alimentación a través del bornero, debe ser estabilizada, es decir, no lo hace internamente el Arduino.

Así que voy a utilizar 4 pilas en serie, de esta forma se compensa la caída de voltaje en la fuente conmutada que será la misma utilizada anteriormente.

Así que utilizaré un porta baterías de 4 pilas en serie, este además incluye una tapadera, de forma que aísla y protege las pilas, también dispone de un botón de encendido y apagado.



Figura 120 Fuente alimentación MKRFOX 1200

Una vez soldados los cables, debemos introducir las pilas y regular la salida de la fuente conmutada a 3,3 (V), una vez listo, se puede poner silicona caliente en el *trimmer* para evitar que este cambie y pueda quemar la placa.

De esta forma ya podríamos alimentar el MKR FOX 1200 y probar su funcionamiento, finalmente el conjunto pesa unos 60 (g) y posee una capacidad de 2.900 (mA).

El consumo del dispositivo en *standby* es de 70 (mA) y emitiendo es de 90 (mA), por los que se estima una duración de la batería de 36,25 horas, menos un 30% de respeto, quedaría en 25,3 horas.

Para calcular la duración de la batería, realizo los siguientes cálculos:

$$\frac{\text{Corriente standby} + \text{Corriente emisión}}{2} = \text{Corriente media}$$

$$\frac{70 \text{ (mA)} + 90 \text{ (mA)}}{2} = 80 \text{ (mA)}$$

$$\frac{\text{Capacidad batería (mAh)}}{\text{Corriente media (mA)}} = \text{Duración de la batería (h)}$$

$$\frac{2.900 \text{ (mAh)}}{80 \text{ (mA)}} = 36,25 \text{ (h)}$$

$$\frac{\text{Duracion de la batería (h)} * 70}{100} = \text{Duración final (h)}$$

$$25,3 \text{ (h)} = \text{Duración final}$$

Finalmente vamos a hacer una comparación entre alimentar la placa a 5 y a 3,3 (V), para ello volveremos a usar la cámara térmica.

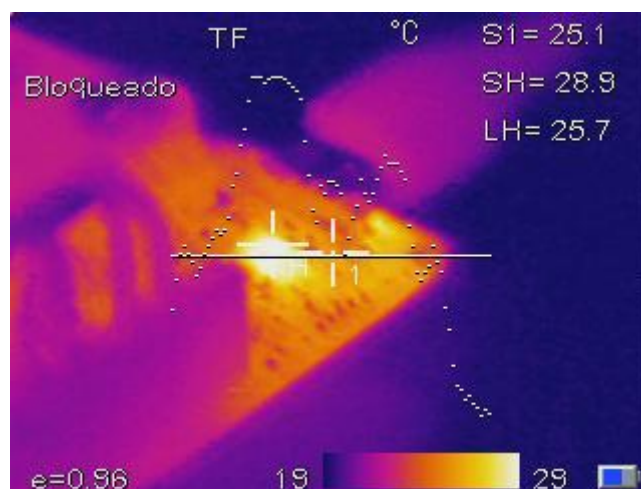


Figura 121 MKRFOX 1200 USB

Esta primera imagen corresponde a una alimentación de 5 (V), podemos observar como la temperatura pasa de unos 22°C hasta los 25,7°C en los alrededores de la entrada de 5 (V), mientras que en el regulador de voltaje la temperatura se dispara hasta los 28,9°C.



Figura 122 MKRFOX 1200 Bornas

Esta segunda imagen es la que corresponde a la alimentación a 3,3 (V), la temperatura media de la placa se mantiene en los 22°C, pero esta vez, no se produce tanta pérdida de energía en la reducción del voltaje, de esta forma, damos una vida más larga tanto a la batería como a la placa, sin embargo, es cierto que para esta aplicación es importante mantener una temperatura elevada de la electrónica, pero esta vez, ya que es un equipo fundamental y crítico, se va a utilizar otro sistema para conseguir eso, pero este sistema se verá un poco más adelante en las siguientes secciones.

WSPR Y ARDUINO UNO:

Aunque el sistema WSPR pertenece al subsistema de comunicaciones, este es el menos importante y tiene un carácter más experimental que práctico, por lo que he decido utilizar una misma fuente de alimentación para el WSPR y el subsistema de computadora.

Ambos dispositivos deben ser alimentados a 5 (V), alimentaré al Arduino a través de USB, mientras que el WSPR, lo alimentaré a través de unos *pads* presentes en su PCB.

Para no tener problemas, utilizaré una fuente estabilizada de 5 (V) con salida USB y de ella, soldaré unos cables para alimentar también el WSPR con 5 (V), con la intención de no aumentar mucho más el peso de la sonda, he utilizado una fuente conmutada elevadora, es decir, la alimento a 3 (V) y a la salida da 5 (V), esta fuente funciona siempre que se alimente a más de 2,5 (V) y es capaz de suministrar hasta 1 (A).



Figura 123 Fuente conmutada elevadora

Para alimentarlo utilizaré un porta baterías de dos pilas en serie, soldaré los cables a la fuente conmutada y una vez suelde los cables del USB para el WSPR, ya podré empezar a probar el Arduino UNO, sin embargo para poder alimentar el WSPR,

tendremos que ver donde soldar la alimentación y como configurarlo, para ello seguiremos los siguientes pasos:

Primero de todo debemos soldar los cables de alimentación de la fuente conmutada a la placa WSPR, para ello en la parte inferior del PCB, podremos ver un texto en el que se indica la entrada de la batería indicando el polo positivo y el negativo.

Soldamos los cables siguiendo este orden y una vez esté listo, solo faltaría cambiar el *jumper* que está al lado de estos cables, al cambiar el *jumper* se conmuta entre alimentación USB y alimentación de baterías.

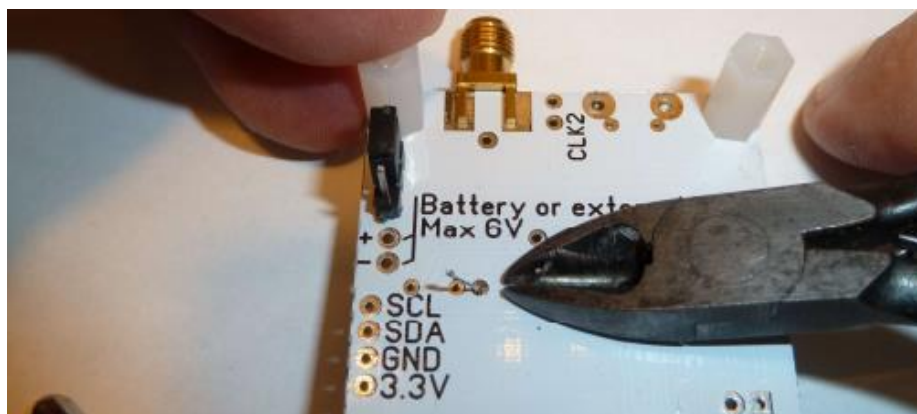


Figura 124 Alimentación baterías WSPR

Una vez seguidos estos pasos, ya estaría toda la electrónica lista para su uso.

Por último, he realizado unas pruebas para determinar la temperatura de funcionamiento de las placas y también unas pruebas para determinar el consumo y la duración de la batería.

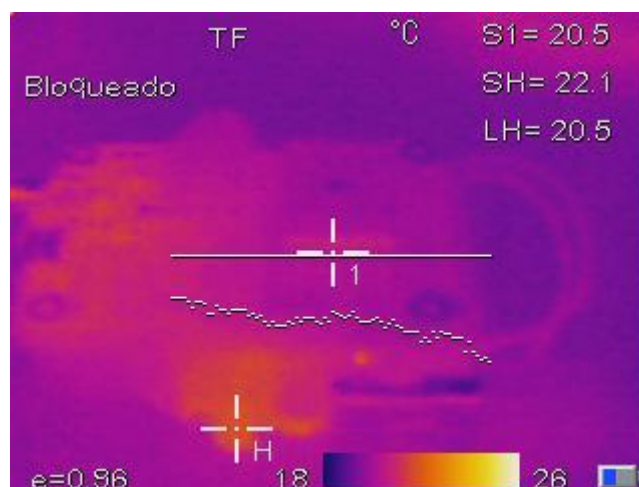


Figura 125 Térmica conmutada elevadora

En esta imagen se aprecia el WSPR y la fuente de alimentación, esta última está trabajando a unos 22°C, mientras que la placa WSPR, esta sobre los 20,5°C.

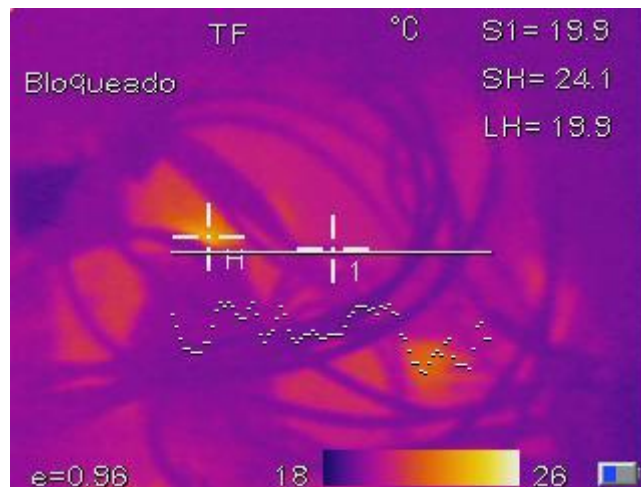


Figura 126 Térmica Arduino

Para el Arduino Uno, podemos ver una temperatura uniforme de unos 19,9°C por toda la placa y algunos puntos de mayor temperatura alcanzando los 24,1°C, estos puntos corresponden a sensores y a la fuente de alimentación.

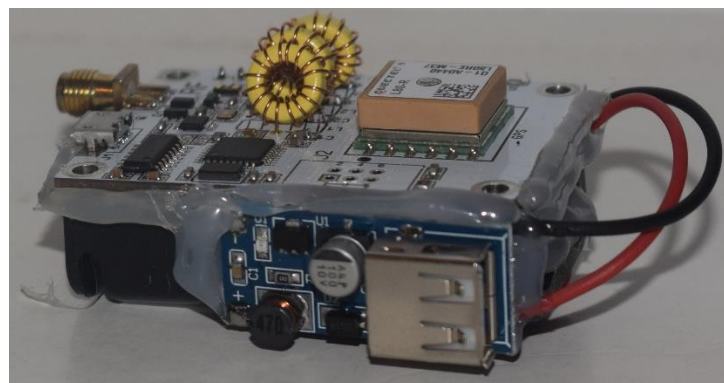


Figura 127 WSPR y fuente de alimentación

Por último, la batería pesa unos 30 (g) y tiene una capacidad de 2.900 (mA).

El consumo del conjunto sin emisión del WSPR es de 140 (mA), mientras que emitiendo se dispara hasta los 160 (mA), por lo que, siguiendo los cálculos desarrollados anteriormente, se determina que la batería durará unas 12 horas y media, teniendo en cuenta el 30% de error.

Por lo que la duración de la batería supera con creces nuestras necesidades.

4.5 ESTACIÓN DE CONTROL TERRENA

En el apartado de subsistema de telecomunicaciones, se han explicado los pasos para poner en funcionamiento los diferentes equipos de comunicaciones que van a ser usados, además se han explicado varios programas para obtener datos de estos dispositivos, con todos esos pasos realizados correctamente, el seguimiento de la sonda se podría realizar con tan solo un teléfono inteligente con acceso a internet.

Sin embargo, el dispositivo WSPR puede requerir de electrónica adicional, tanto para su correcta configuración, como para la correcta comprobación de su funcionamiento.

Además, esta electrónica adicional puede ser utilizada en tierra para aumentar las capacidades de nuestra estación de control, de esta forma podríamos seguir obteniendo datos independientemente de la conexión a internet.

Por otro lado, gracias a esta electrónica adicional que voy a explicar a continuación, también se puede comprobar el funcionamiento de SigFox, aunque la información que se recibe esta codificada, se puede ver la señal en el dominio de la frecuencia y por lo tanto se puede intentar usar una antena direccional para poder determinar hacia qué dirección se encuentra la sonda.

A continuación, voy a explicar con detalle todo lo expuesto anteriormente.

4.5.1 FABRICACIÓN DE ANTENAS

Ya sea para recibir datos de WSPR o de SigFox, lo primero que se necesita es una antena adaptada a la frecuencia a la que funciona el sistema, en el caso de SigFox esta es de 868 (Mhz), mientras que para WSPR es de 14 (Mhz).

A la hora de construir una antena, es necesario saber que existen diferentes tipos en función de cómo queremos recibir la señal, por ejemplo, están las antenas omnidireccionales, las antenas sectoriales y las direccionales.

En este caso, vamos a ver cómo crear antenas omnidireccionales y un ejemplo de antena direccional para el sistema SigFox.

ANTENAS OMNIDIRECCIONALES:

Vamos a realizar los cálculos para obtener la longitud de la antena para una antena omnidireccional de WSPR y de SigFox.

$$\lambda^{WSPR} = \frac{300.000.000(\frac{m}{s})}{14.000.000(Hz)} = 21,4285 (m)$$

$$\lambda^{SigFox} = \frac{300.000.000(\frac{m}{s})}{868.000.000(Hz)} = 0,3456 (m)$$

Es decir, la longitud de onda para WSPR, es de 21,42 (m), mientras que para SigFox es de 0,34 (m).

Para el caso de SigFox, se puede utilizar un cable rígido de cobre de 34 (cm), este irá soldado al vivo de un cable de antena, mientras que el negativo servirá para proteger la señal hasta que esta llegue al receptor, es necesario y se debe comprobar que el vivo y tierra no estén en cortocircuito.

Sin embargo, para WSPR, la longitud de la onda nos puede dar bastantes problemas para poder trabajar con ella, para hacer pruebas en el laboratorio, he utilizado un cable de cobre que sea flexible, he cortado 21 (m) y he soldado el extremo al vivo de un cable de antena apantallado, este cable termina en un conector SMA macho que se conecta al receptor.

Una vez montada la antena, he dispuesto esta, en polarización horizontal a lo largo del techo en el laboratorio, con esta disposición he podido realizar las pruebas.

Sin embargo, si se va a tener que llevar la antena a otro lugar o se va a tener que trabajar con ella en otro lugar, la longitud de esta antena puede ser un problema, por ello se pueden hacer dos variaciones a esta antena para mejorar un poco este problema.

La primera opción consiste en construir una antena Hertz, o lo que es lo mismo, una antena de media onda.

Para ello, la longitud de la antena debe ser la mitad de la longitud de onda, para el caso de WSPR, la longitud sería de 10,5 (m) y para SigFox sería de 17 (cm).

La segunda opción, sería la de crear una antena Marconi o también conocida como antena de cuarto de onda.

Para este caso las longitudes pasarían a ser de 5,25 (m) en WSPR y de 8,5 (cm) en SigFox. Sin embargo, este tipo de antena es un poco más complicada de construir puesto que debe disponer de un plano de tierra para poder crear un reflejo de la señal en la tierra y emular así una antena de media onda.

ANTENAS DIRECCIONALES:

Una vez explicadas las antenas omnidireccionales, procedo a explicar la fabricación de una antena direccional, así como a indicar que antena he usado para la aplicación.

La antena direccional utilizada es una antena tipo YAGI, estas están formadas por un reflector, varios directores y un dipolo o elemento radiante.

Para fabricar el dipolo de esta antena, este debe ser de la mitad de la longitud de onda, por lo que se descarta la fabricación de una antena de este tipo para WSPR, este tipo de antena es muy útil para recibir la señal de SigFox.

Para poder calcular esta antena, vamos a exponer las operaciones necesarias.

Primero debemos obtener la longitud del reflector, la del dipolo y la de los directores.

$$dipolo (m) = \lambda * 0,5 = 0,17(m)$$

$$reflector (m) = \lambda * 0,525 = 0,1785(m)$$

$$directores (m) = \lambda * 0,48 = 0,1632(m)$$

Los siguientes datos que calcular son: la distancia entre reflector-dipolo; la distancia entre dipolo-director y por último la distancia entre directores.

$$dipolo/reflector (m) = \lambda * 0,15 = 0,051(m)$$

$$dipolo/director (m) = \lambda * 0,1 = 0,034(m)$$

$$director/director (m) = \lambda * 0,1 = 0,034(m)$$

Una vez conocemos estos datos, podemos cortar varios cables rígidos de cobre de estas longitudes y los podemos montar como se ve en esta imagen, respetando las distancias calculadas.

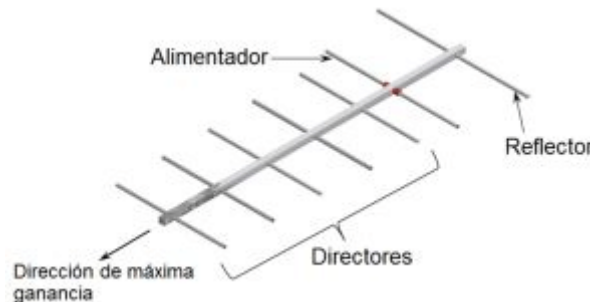


Figura 128 Esquema YAGI

A la hora de construir el dipolo, es necesario dar una explicación más, este debe ser de la longitud calculada anteriormente, pero debe describir una forma de ovalo muy alargado, de forma que, si respetamos el dibujo, realmente la longitud será la longitud de onda, entonces, lo que se hará será cortar dos tramos de unos 34 (cm) y los doblaremos como se ve en el esquema.

Uno de estos tramos se conecta a masa, mientras que el otro se conecta al vivo.

Como ya hemos comentado anteriormente, conectamos la antena a un cable de antena apantallado y de este con un conector SMA, lo conectaremos al receptor y ya tendríamos la antena lista para usarse.

En este caso, para la aplicación hemos utilizado una antena YAGI de tipo comercial optimizada para la banda UHF y VHF.



Figura 129 YAGI comercial

Una vez montadas las antenas, ya podemos empezar a hacer pruebas de recepción, para ello se van a utilizar dos programas principales, los cuales se exponen a continuación.

4.5.2 SDR

El primer dispositivo que se va a utilizar es un SDR, este es un dispositivo que es capaz de recibir señales de radio en un ancho espectro, en mi caso, el modelo usado abarca desde los 100 (KHz) hasta los 1,7 (GHz). Con la ayuda de un PC, se sintetiza la señal recibida por medio de este SDR y de un *software* específico.

Una vez que se sintetiza, se puede escanear el espectro electromagnético y se puede ver la señal en el dominio de la frecuencia y escuchar la señal a través de la tarjeta de sonido del ordenador.

Una vez sintonizada la frecuencia que deseamos, podemos grabar el audio y mediante una serie de programas, se puede decodificar el mensaje recibido.

En este caso, solo lo vamos a utilizar para determinar si existe portadora y poder determinar también si está funcionando bien el dispositivo.

Para ello primero de todo debemos conectar nuestro SDR a nuestro PC utilizando el USB, cuando esté conectado, lo siguiente será instalar el controlador para que nuestro ordenador sea capaz de reconocer nuestro SDR.

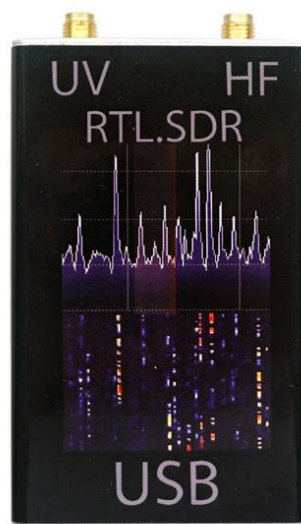


Figura 130 SDR

Una vez conectado, debemos instalar el *driver*, para ello buscaremos el *driver* para el *chip* de nuestro SDR en el siguiente enlace: <http://www.hdsdr.de/hardware.html>

Lo descargamos y lo siguiente que debemos hacer es descargar un programa llamado ZADIG, el cual nos permitirá instalar el controlador descargado.

El programa se encuentra en el siguiente enlace: <https://zadig.akeo.ie/>

Descargamos el programa antes de ejecutarlo descargaremos por último el programa que utilizaremos para poder usar el SDR, existen muchas alternativas, pero en este caso se va a utilizar un programa gratuito llamado HDSDR.

Para descargarlo accedemos al siguiente enlace: <http://www.hdsdr.de/index.html>

Una vez tenemos el controlador y los dos programas descargados, lo siguiente que haremos será instalar el programa HDSDR como administrador.

Una vez instalado el programa, debemos abrir la ubicación del programa y pegar en la carpeta el controlador descargado.

Una vez pegado el controlador en la carpeta, cerramos esta y ejecutamos el programa Zadig como administrador.

Cuando el programa se ejecuta aparece una ventana principal con varias opciones en la barra superior, debemos seleccionar “opciones” y en el desplegable que aparece, seleccionamos “*list all devices*”.

Una vez seleccionado esto, volvemos a la ventana principal, seleccionamos nuestro dispositivo e instalamos el controlador.

Finalmente, cerramos este programa y ya podemos usar el programa HDSDR.

Cuando se abra por primera vez, tan solo tenemos que pulsar sobre “start” para que comience el escáner de frecuencias, luego se debe elegir la frecuencia en el dial y el tipo de modulación.



Figura 131 Panel HDSDR

Con toda esta información, ya se podría ver en el dominio de la frecuencia si existe portadora para el WSPR, al igual que se podrían ver las emisiones de SigFox si está funcionando, para ello es importante tener en cuenta que nuestro dispositivo SDR tiene dos entradas de antena, cada una de ellas es para un rango de frecuencias diferente, debemos conectar las antenas al conector correspondiente.

Una de las tomas de antena funciona en el rango comprendido entre los 100 (KHz) y los 30 (MHz), mientras que la otra entrada de antena funciona desde los 25 (MHz) hasta los 1,7 (GHz).

En caso de que se utilice la antena direccional, si la antena apunta al dispositivo emisor durante la emisión, en el programa aparecerá patrón característico, pero si la antena no apunta a este, el espectro se verá muy atenuado, incompleto o incluso puede no verse.

Esto puede ser de utilidad para determinar hacia qué dirección se encuentra la sonda.

Si sabemos que ha aterrizado, pero no emite ningún mensaje, puede ser que por falta de cobertura el mensaje no llegue a nosotros, sin embargo, el dispositivo seguirá emitiendo por lo que podría triangularse la posición mientras que siga disponiendo de batería para transmitir.

4.5.3 WSPR RX

Por último, para la recepción de datos del dispositivo WSPR, vamos a utilizar las estaciones desplegadas por toda España, pero también hemos montado una pequeña estación de escucha capaz de subir los datos a la web de WSPR, para poder hacerlos públicos.

Para ello hemos contado con la ayuda de la URE-Cartagena.

Hemos utilizado un PC con entrada de micrófono sincronizado con la hora mundial con un margen de error de 1 segundo, una radio modelo YAESU FT-897, una fuente de alimentación de laboratorio de 12 (V) y 35 (A), un conector CAT, un cable de audio *Jack-Jack*, una antena de para la recepción y conexión a internet.



Figura 132 YAESU FT-897

Una vez conectado todo el sistema, debemos encender el ordenador y conectarnos a internet, posteriormente accedemos a la página web:

<https://physics.princeton.edu/pulsar/k1jt/> y descargamos el programa WSJT o WSPR, en este caso hemos utilizado el programa WSJT.

Lo descargamos, lo instalamos y accedemos a él, podremos configurarlo y ponerlo en funcionamiento.

UTC	dB	DT	Freq	Drift	Call	Grid	dBm	km
1644	1	0.5	14.097078	-1	EASURD	IM97	13	0
1648	1	0.5	14.097152	-2	EASURD	IM97	13	0

Figura 133 Trama WSPR

Podremos ver tramas similares a estas cuando el programa comienza a recibir los primeros datos, estos datos una vez obtenidos, son cargados al servidor WSPR, permanecerán colgados en el servidor por dos semanas.

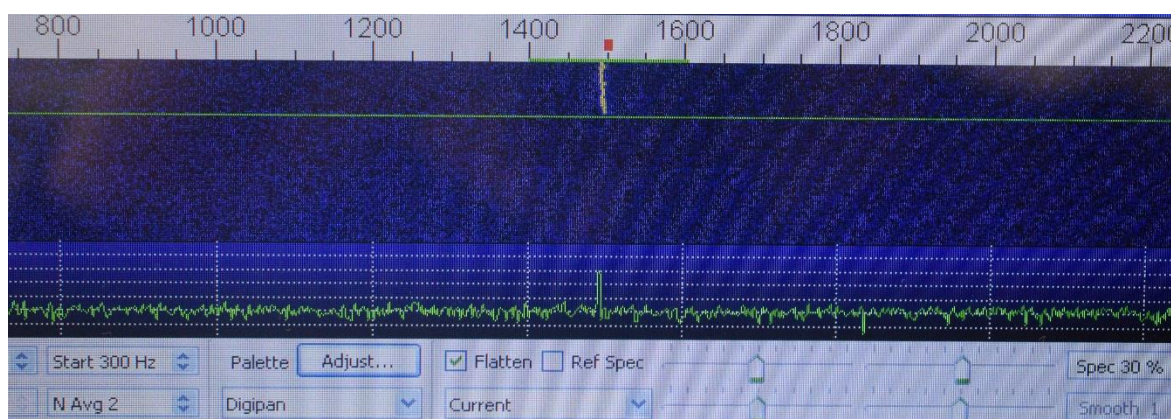


Figura 134 Recepción WSPR

Un dato importante a tener en cuenta es que el modelo utilizado de WSPR, no es capaz de emitir conectado mediante USB, puede ser debido a las interferencias producidas por las fuentes de alimentación conmutadas de los ordenadores portátiles, al utilizar baterías, no existe ningún problema y el mensaje se recibe perfectamente.

Esta estación propia, nos ha sido muy útil para probar el funcionamiento de nuestro dispositivo WSPR, además nos ha permitido profundizar en el desarrollo de una estación fija de escucha WSPR y como hacer los reportes de propagación públicos.

Esta estación tiene la posibilidad de instalarse en un coche y poder hacer el seguimiento durante el vuelo, sin embargo, se va a utilizar el día del lanzamiento como una estación de escucha en la ciudad de Cartagena.

5. PAYLOAD BOX



Una vez explicado cada uno de los dispositivos electrónicos a utilizar, se debe buscar una estructura que sea capaz de contener toda la electrónica, que sea ligera, resistente a los golpes, aislante del frío y que sea de muy bajo coste, puesto que será una parte no reutilizable de la sonda.

Otra opción puede ser desarrollar una estructura mediante impresora 3D, de esta forma, se puede montar la electrónica de forma mucho más customizada, aunque de este asunto se hablará en el apartado de conclusiones y mejoras.

A la hora de hablar sobre la construcción de la estructura, vamos a diferenciar por un lado todos los trabajos realizados sobre la parte exterior, como puede ser el acople del sistema de paracaídas, preparación de la antena WSPR, la disposición del termómetro externo, el aislamiento utilizado, etc...

Por otro lado, se tratarán las modificaciones en la parte interna, como por ejemplo la disposición de los componentes, el fijado de estos, prevención de jaulas de Faraday, polarización de antenas, distribución del centro de gravedad...

Por último, se tratará uno de los puntos más importantes de esta sección, el cual consiste en un pequeño estudio sobre la transmisión del calor de una fuente de calor química hacia los componentes electrónicos a través de la conducción y la convección. Este pequeño estudio está encaminado a eliminar uno de los principales problemas de la electrónica en este tipo de vuelos: la baja temperatura y la condensación.

5.1 ESTRUCTURA EXTERNA

Para este lanzamiento, he utilizado una esfera hueca de poliestireno de 4 cm de grosor, y de 25 (cm) de diámetro, esta esfera está compuesta por dos mitades que encajan entre sí.

En concreto el material de la estructura es poliestireno expandido, el cual posee una gran ligereza, que puede variar entre los 10 (Kg/m³) y los 35 (Kg/m³), este material posee una gran resistencia a los golpes y es un excelente aislante térmico.



Figura 135 Esfera poliestireno 25 (cm)

Una vez seleccionada la estructura principal, se tiene que elegir una segunda capa de aislamiento, para esto existen varias opciones, por ejemplo, se puede utilizar una manta térmica de emergencia, estas disponen de una capa color plata y una capa color oro, al utilizar la parte color oro, se consigue que se refleje menos radiación y por lo tanto se consigue calentar el objeto que está cubierto por esta manta.

Sin embargo, esta solución plantea varios problemas, el primero de ellos es que es difícil trabajar este tipo de material, especialmente en una superficie esférica, por otro lado, el uso de este material plástico recubriendo toda la sonda, puede favorecer la condensación y esto puede ser un gran problema.

Por ello una solución más eficaz, al menos para este tipo de estructura, es utilizar una pintura negra acrílica, al recubrir toda la superficie de este color, se consigue que absorba mejor la radiación solar, es muy fácil de trabajar con este material y además con unas dos capas de pintura, se consigue crear una película sobre el poliestireno que le da mucha consistencia frente a cortes.

Es muy importante recordar que la pintura debe ser acrílica, puesto que, de otra forma, podría dañar el poliestireno.

Una vez determinado cual va a ser el aislamiento exterior, se procede a añadir dos aberturas para añadir el termómetro externo y la antena WSPR.

El procedimiento para ambos es el mismo, a continuación, se va a detallar el proceso para la antena WSPR.

Primero de todo, se tiene que coger una de las dos partes de la esfera, la que seleccionemos pasará a ser la parte inferior de la sonda, una vez seleccionada, debemos hacer una perforación muy pequeña, lo justo para que pase el cable de la antena, para ello debemos utilizar una herramienta del mismo calibre que nuestro cable y que sea de un largo suficiente como para atravesar la pared de la esfera.

La incisión debe ser realizada en la parte más baja, de forma que, si observamos la parte cóncava de la media esfera en proyección, la incisión quedaría en la parte central.

Una vez hemos realizado la incisión, se debe introducir el extremo de la antena a través de esta incisión y se debe deslizar toda la antena, de forma que finalmente llegue a tocar el poliestireno con el cable apantallado, una vez el cable apantallado llegue a tocar, introduciremos el cable apantallado a través de la incisión hasta que aparezca por la parte exterior de la esfera.

Una vez que el extremo del cable apantallado esta ligeramente por fuera de la esfera, podemos usar un poco de termo retráctil para proteger la soldadura que une la antena con el cable coaxial apantallado.

Cuando esta operación está acabada, podemos utilizar una pistola de silicona caliente y pegar el cable tanto por la parte exterior como la interior, aplicando un poco de esta silicona alrededor del cable, de esta forma se consigue unir firmemente el cable a la esfera y además se aísla totalmente la incisión, quedando la sonda de nuevo estanca.



Figura 136 Perforación antena WSPR

Es posible que debido al uso continuo de esta antena para realizar pruebas y también por la probabilidad de esfuerzos mecánicos sobre esta, existe la posibilidad de que se pueda soltar esta silicona y que deforme la incisión, pudiendo incluso abrir un hueco en la superficie de la esfera, para que esto no ocurra tan fácilmente y proteger aún más la estructura, se ha seguido el siguiente procedimiento.

Mediante este procedimiento, se intenta conseguir que, si la antena sufre un fuerte tirón desde la parte exterior, esta no sea arrancada llevándose consigo un trozo de la esfera o incluso estropeando la electrónica, para evitar esto, he intentado repartir la fuerza a lo largo del área que rodea la parte de la incisión, de esta forma, será mucho más difícil arrancar la antena.

Lo primero que he realizado es un recorte de manta térmica con un agujero en el centro para que pase el cable coaxial, luego he pegado esta manta a la esfera mediante un pegamento especial para plásticos.

Posteriormente he recubierto esta con cinta americana negra, para darle más consistencia y fijarla mejor a la estructura.



Figura 137 Refuerzo antena WSPR

Finalmente, terminé de recubrir toda la manta térmica con cinta aislante, luego utilicé una brida pequeña para rodear el cable de antena y dejarlo ajustado para que no se mueva y se adhiera mejor la silicona caliente.

Una vez realizado esto, vuelvo a poner silicona caliente rodeando el cable y cuidando de que se reparte bien y se pega sobre la superficie de la cinta aislante, de esta forma para arrancar la antena, sería más fácil romper la soldadura que arrancar todo este material y con él un buen trozo de poliestireno.

Una vez explicado este procedimiento, podemos seguir el mismo para añadir el sensor de temperatura DS18B20, con la diferencia de que no sería necesario proteger tanto la estructura, ya que este no estará sometido a tracciones ni torsiones.

La única precaución a la hora de montar el sensor es que se monte en la otra parte de la esfera, es decir, la que quedará en la parte de arriba y también que la parte metálica del sensor debe quedar totalmente expuesta al exterior.

El resultado sería este:



Figura 138 Termómetro exterior

Una vez montada la antena y el sensor de temperatura, se procede a montar la estructura que fijará la esfera a la línea del paracaídas y del globo, esta es una parte muy importante puesto que deberá soportar mucha tracción, todo el empuje del globo y el fuerte tirón del paracaídas al abrirse, así como todo el esfuerzo del frenado del paracaídas.

Para construir este sistema utilizaremos la parte superior de la esfera, la misma utilizada para el termómetro, realizaremos 4 incisiones en forma de cuadrado dejando el termómetro en el centro de este, para ello utilizaremos de nuevo un objeto fino, punzante y largo.

La distancia del lado del cuadrado fue de unos 14 (cm) aproximadamente, una vez realizadas las perforaciones, debemos cortar dos trozos de hilo de nylon de un grosor superior a 2 (mm), este material es muy ligero y resistente a la tracción.

Deberemos cortar dos trozos de un metro aproximadamente, desde la parte exterior de la esfera introducimos uno de los extremos por uno de los orificios, luego metemos el otro extremo por el orificio opuesto.

Una vez hecho esto, he utilizado una junta toroidal de plástico para hacer un nudo en los extremos del cabo, de esta forma, se evita que el nudo pueda deslizarse a través de el orificio debido a un fuerte tirón.

Repito el mismo procedimiento con el otro cabo y finalmente pongo silicona caliente tanto por la parte exterior como por la interior, de esta forma se quedan firmemente fijados los cabos y las juntas a la esfera y también se vuelve a sellar el envoltorio.

Luego con otro cabo, lo pasamos por el interior de las cuatro juntas y le hacemos un nudo para unir los extremos y de esta forma crear un anillo para ayudar a que ninguno de los cabos se suelte de la estructura principal.



Figura 139 Sistema tracción paracaídas

Una vez terminado, debemos unir los dos cabos mediante un quitavuellos metálico, este debe soportar como mínimo 20 (Kg) de tracción.



Figura 140 Quitavueltas acero inox

Una vez terminado el sistema de unión de la carga con el sistema de propulsión, ya solo queda realizar dos huecos para las lentes de las cámaras.

Para ello volveremos a utilizar la parte de debajo de la esfera, es decir, la parte en la que va montada la antena WSPR.

Una vez que tenemos esta parte y las cámaras, debemos poner las cámaras en la posición que deseamos, debemos tener en cuenta hacia donde queremos que graben, una vez lo tengamos decidido, solo tendremos que marcar la silueta de las lentes en el poliestireno.

Cuando tengamos la silueta marcada, debemos utilizar un *cutter* para recortar la silueta, este trabajo es muy delicado y tiene que realizarse con cuidado y sin prisa.

Cuando el trozo este cortado, podemos comprobar como la lente se encaja a la perfección, pero aún queda un problema que resolver, ya que, si la cámara se ajusta y se empieza a filmar, los bordes hacen que no se pueda ver bien la imagen, por lo que debemos suavizar estos bordes por la parte exterior hasta que no se pierda imagen debido a estos.

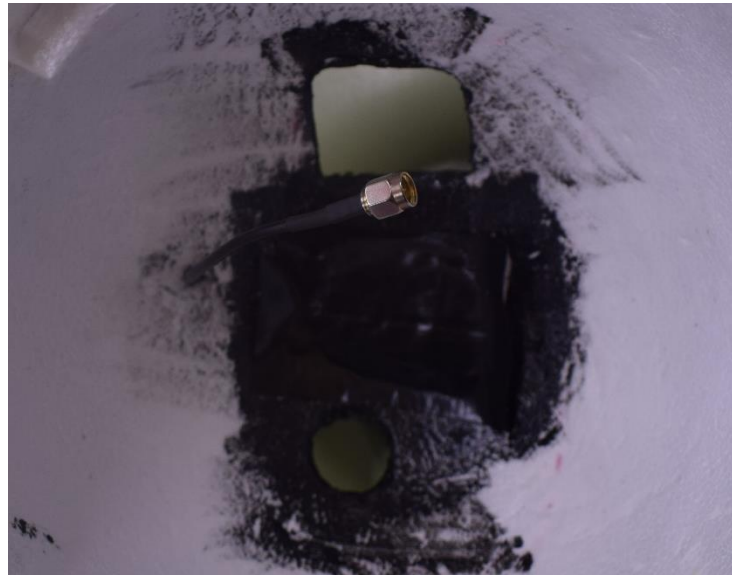


Figura 141 Agujeros cámaras internos

Una vez suavizados los bordes, este ha sido el resultado:

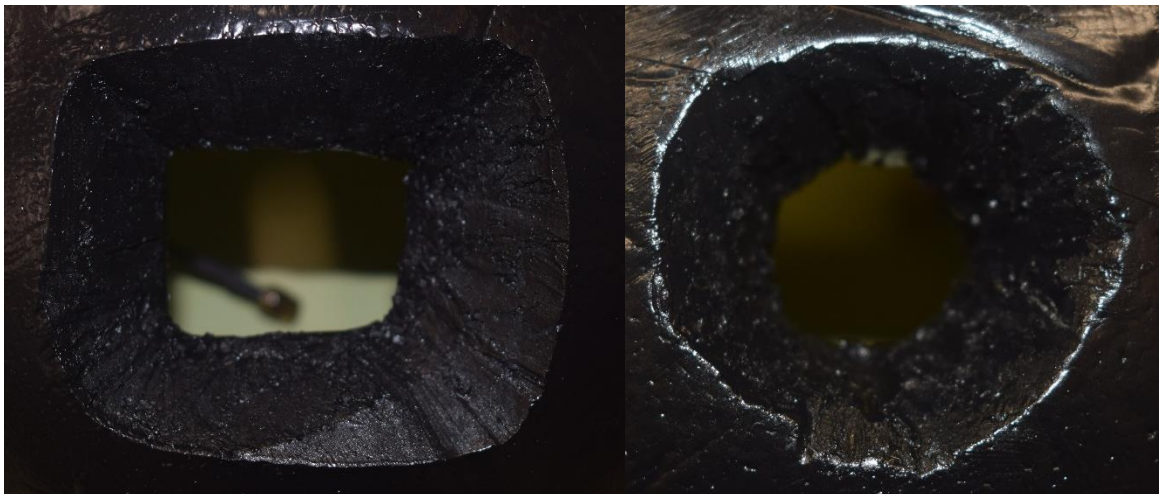


Figura 142 Agujeros cámaras externos

Con este acabado, ya se podría pintar la parte exterior, es importante pintar la parte recortada puesto que, si queda de color blanco, puede provocar destellos en las lentes y estropear las imágenes.

Una vez pintada toda la esfera, se debe dejar secar y posteriormente dar una segunda capa de pintura.

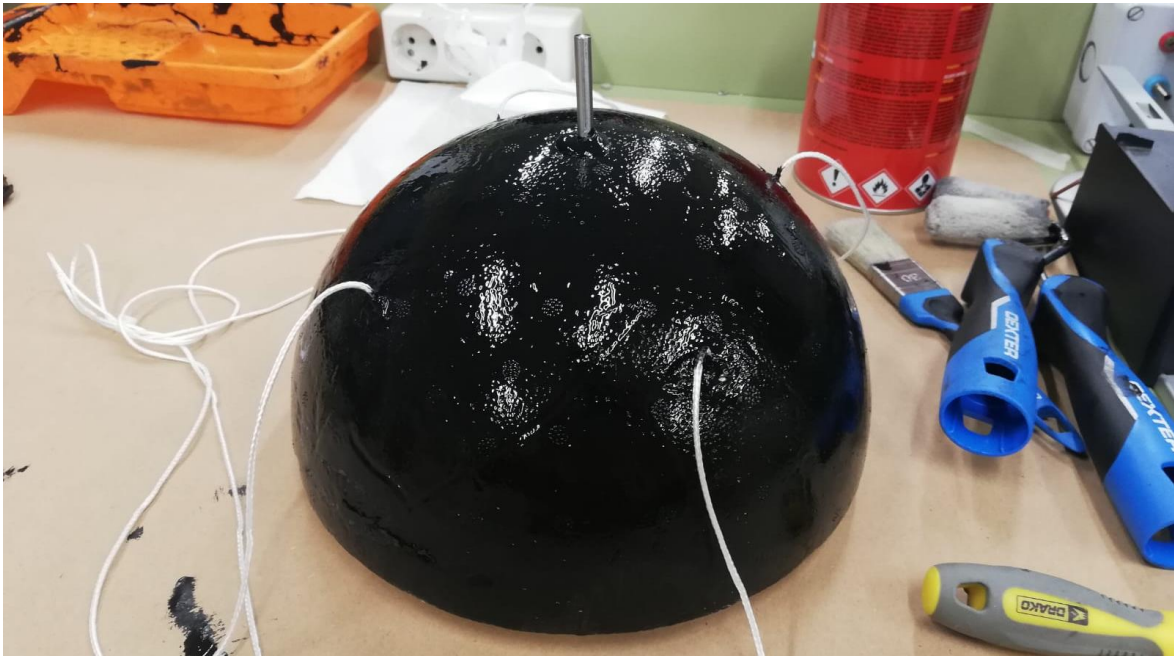


Figura 143 Capa de pintura sobre la sonda

Con toda la parte externa terminada y con toda la electrónica lista, ya podemos comenzar a ver cómo quedará dispuesta esta en el interior de la sonda.

5.2 ESTRUCTURA INTERNA

Para montar los componentes en el interior de la sonda, vamos a comenzar por la parte inferior de la esfera, en ella colocaremos las cámaras, así como una estructura aislante donde irán pegados algunos dispositivos y las baterías, de forma que el centro de gravedad quede ligeramente más bajo que el centro geométrico de la esfera, de esta forma, durante la reentrada se estabilizará mejor.

Es importante comentar que, aunque aquí se describe como montar todos los componentes, no todos estos quedan dispuestos o fijos hasta el día del lanzamiento, algunos como las cámaras y la estructura aislante que voy a comentar ahora, se dejarán fijos de forma definitiva solo en el lanzamiento, de forma que, si se tienen que hacer pruebas, cargar baterías, etc... Sea fácil trabajar y se puedan montar y desmontar como si fueran módulos independientes.

Una vez aclarado esto, procedo a explicar la solución final de las cámaras.

Primero se deben poner con la lente encajada en su hueco de la estructura y una vez que queden ajustadas del todo, se utilizará plastilina para sellar los bordes y evitar que entre el aire.

Luego se pondrá sobre las cámaras un pequeño molde de espuma de poliuretano para que sirva de aislante y de plataforma para las baterías y el resto de la electrónica.

Para hacer esta estructura, deberemos proteger las cámaras con papel *film*, luego las encajamos en su sitio, de forma que la lente quede en su sitio, apuntando hacia afuera.

Una vez hecho esto, pondremos plastilina alrededor de las cámaras, para sellar los orificios que puedan quedar abiertos.

Con papel *film*, también debemos proteger el cable coaxial de la antena WSPR.

Cuando todo esté protegido, entonces podremos recubrir toda la parte interna inferior de la esfera con vaselina, es importante que quede todo recubierto, puesto que, al utilizar la espuma de poliuretano, esta quedará pegada donde no se encuentre la vaselina.

Así que debemos recubrir todo de forma metódica, incluidas las cámaras y el cable coaxial.



Figura 144 Cámaras con vaselina

Una vez listo, debemos forrar por completo la parte inferior de la esfera con papel *film*, de forma que la parte cóncava quede sellada completamente, cuando quede totalmente sellada, entonces introduciremos un pequeño tubo atravesando el papel *film*, de forma que este quede en la parte cóncava de la esfera, a través de este tubo se introducirá la espuma de poliuretano, la idea es que se llene a presión y

que cuando la espuma comience a expandirse, se cree un molde que se ajuste a la perfección con la antena, las cámaras y la semiesfera.

Gracias a la vaselina, una vez seca la espuma, podremos retirar el molde sin problemas.

Además de todo lo anterior, sobre el papel *film*, hemos introducido un objeto relativamente voluminoso para crear una superficie relativamente cóncava en el molde de poliuretano, el sentido de este hueco se explicará más adelante.

Una vez relleno toda la semiesfera de espuma, usaremos una tabla y unos sargentos para evitar que, con la expansión de la espuma, se rompa el papel *film* y no se quede bien el molde.

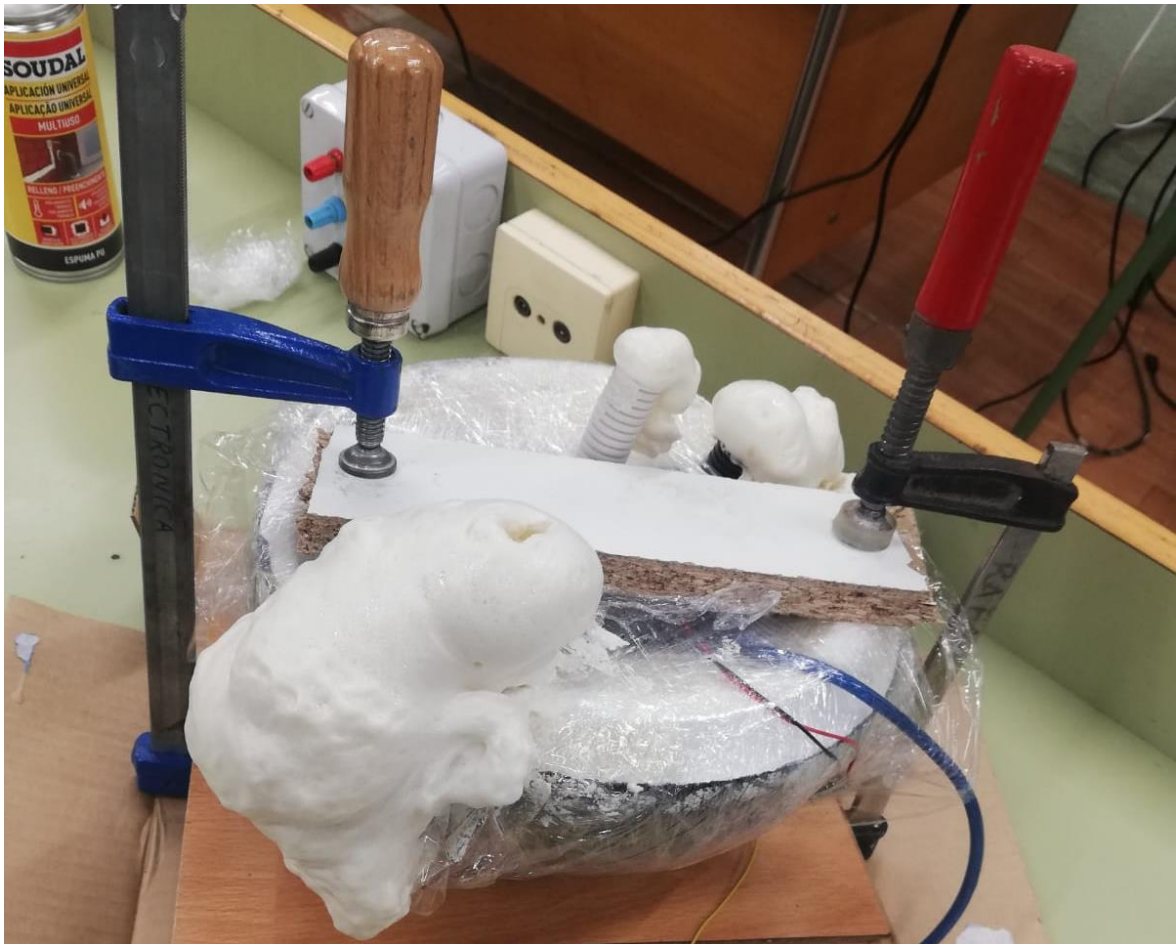


Figura 145 Fabricación molde poliuretano

Una vez que la espuma se seque y se quede rígida, entonces podemos comenzar a retirar los reboses de los tubos, luego podremos quitar el papel *film* y proceder a

retirar el molde con mucho cuidado, por si alguna parte ha quedado pegada durante el proceso.

Cundo se retire el molde, debemos limpiar toda la superficie de la esfera, las cámaras, la antena, etc...

Debe quedar todo limpio y el molde se tiene que limpiar también porque estará lleno de vaselina, cuando esté limpio, podemos comprobar como el molde encaja con las cámaras.

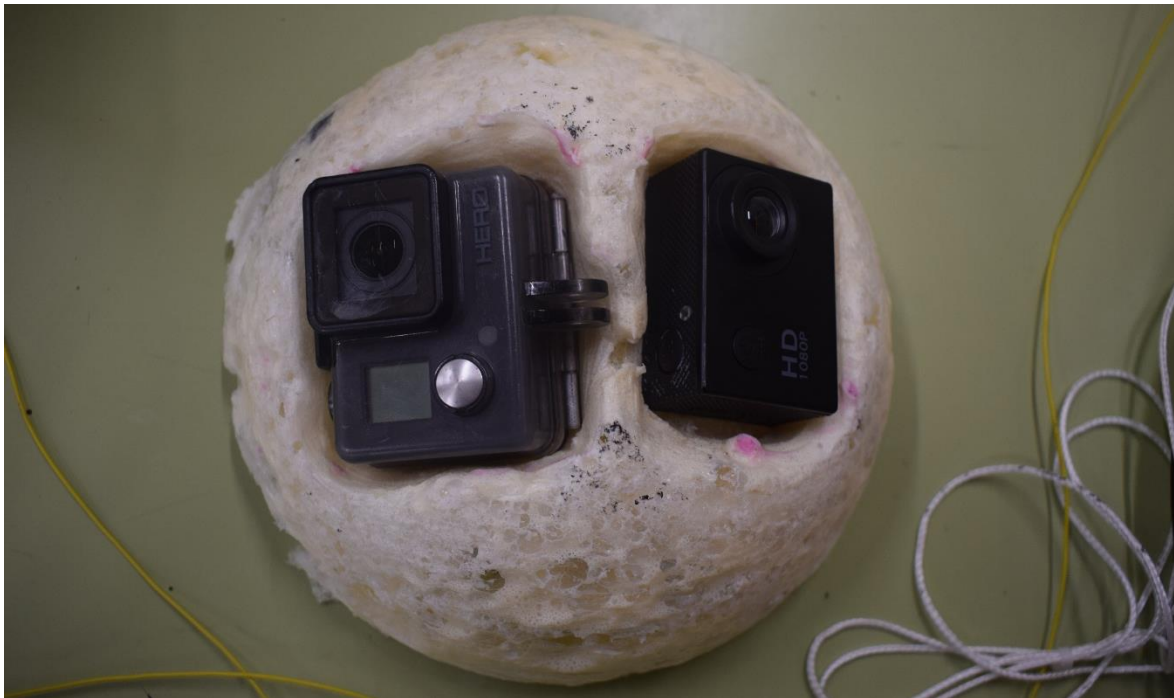


Figura 146 Molde cámaras

Luego se debe volver a encajar el molde en la semiesfera para probar como queda el agujero de la antena, es importante recordar que el conector debe quedar por la parte superior de este molde, en caso de que no entre bien, utilizaremos alguna herramienta para ensanchar el agujero, incluso podemos utilizar algún tubo hueco para que resulte mejor el acabado y que no se meta espuma en el conector a la hora de meter y sacar el molde.

Cuando se terminen los pasos anteriores, procederemos a terminar la parte superior del molde, en la cual irán dispuestos la mayoría de los componentes electrónicos.

Para seguir con la fabricación, volveremos a utilizar espuma de poliuretano, papel *film*, cartón y un poco de poliestireno.

El proceso consiste en cortar un trozo de cartón en forma de circunferencia con un radio igual al de la parte interna de la esfera, de forma que, al poner el molde en la semiesfera, si se pone el cartón sobre este molde, este debe crear una plataforma plana que cubra por completo la semiesfera.

Cuando se tenga listo el cartón, se debe cortar unos pequeños tacos de poliestireno, los cuales estarán cortados de forma que compensen las irregularidades de la superficie de la espuma de poliuretano.

De esta manera, si se apoya el cartón sobre estos pequeños tacos, este quedará nivelado en posición horizontal.

El siguiente paso, será pegar estos tacos a la espuma mediante silicona caliente y luego pegar el cartón a estos tacos de poliestireno, una vez que estén todos pegados, haremos uso de dos componentes, uno de ellos será el porta baterías del sistema SigFox, y otro de ellos consiste en una fuente de calor química, la cual veremos a fondo en la siguiente sección, en esta de momento, nos limitaremos a forrar estos dos componentes con papel *film* y una vez que estén correctamente forrados, los introduciremos entre el cartón y el molde de espuma de poliuretano, la idea es que queden ajustados entre las dos superficies y que se introduzcan lo justo, ya que estos deben poder quitarse y ponerse de forma sencilla una vez que se saque el molde de la semiesfera.

La disposición que he utilizado es la de poner estos dos dispositivos enfrentados, es decir uno enfrente de otro en los extremos de la circunferencia. Una vez en su sitio, podemos comenzar a rellenar las partes huecas entre el cartón y el molde, usando más espuma de poliuretano.

Una vez seca la espuma añadida, podemos comprobar que encaja a la perfección en la semiesfera, probaremos que la antena pasa bien a través de la espuma añadida y haremos un agujero en el cartón para que sobresalga, luego retiraremos el calentador y el porta baterías para quitarles el papel *film*.

Cuando quitemos el papel *film*, comprobaremos si el hueco dejado es suficiente, lo comprobaremos introduciendo de nuevo los dos componentes en sus huecos.

Si todo va bien, podemos marcar sobre el cartón, donde va cada componente y podemos pegar la fuente de alimentación de SigFox.

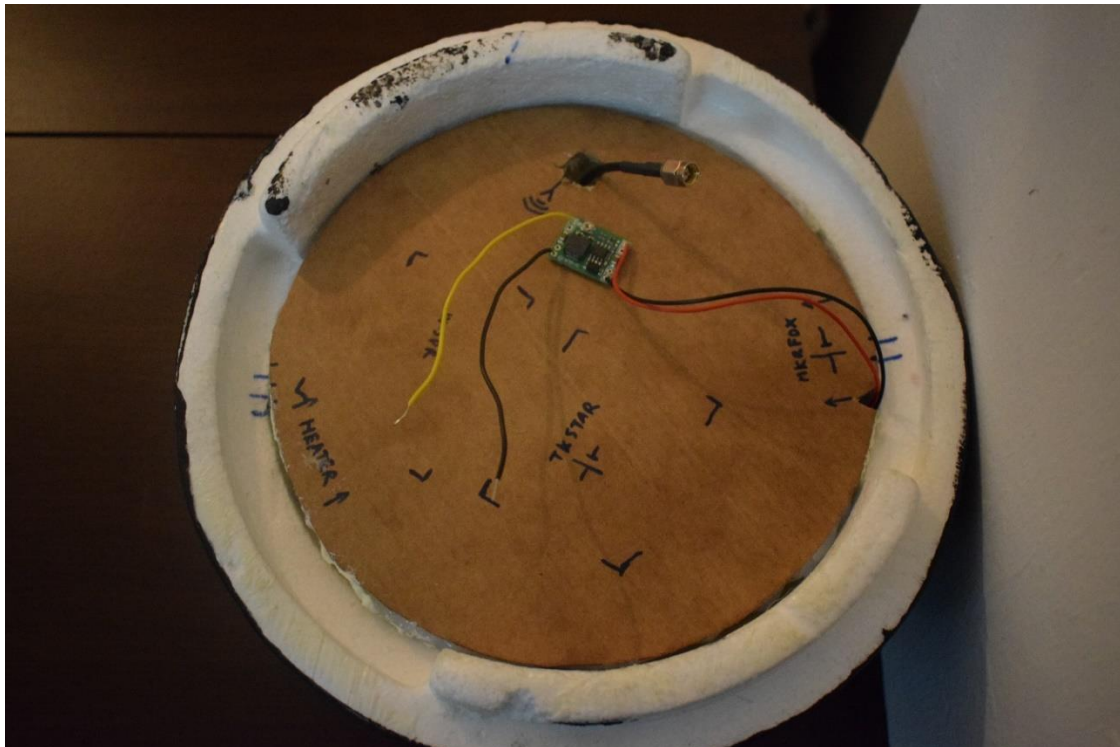


Figura 147 Plataforma para electrónica

Una vez lista la plataforma, podemos pegar algunos de los componentes sobre esta, estos componentes serán: WSPR, batería WSPR, fuente de alimentación WSPR y SigFox con antena GPS.

Una vez pegados estos componentes, ya solo falta ver cómo queda dispuesto el Arduino Mega con sus sensores y el GPS *tracker*.

Estos dos componentes irán dispuestos en la parte superior de la esfera, el Arduino irá fijado al cabo que une las cuatro juntas de plástico, se unirá mediante bridas y el GPS *tracker* irá pegado a la cara interna de la esfera, de forma que tanto la antena del GPS *tracker* como la antena GPS del Arduino queden lo más horizontales posibles y apuntando hacia el exterior de la esfera.

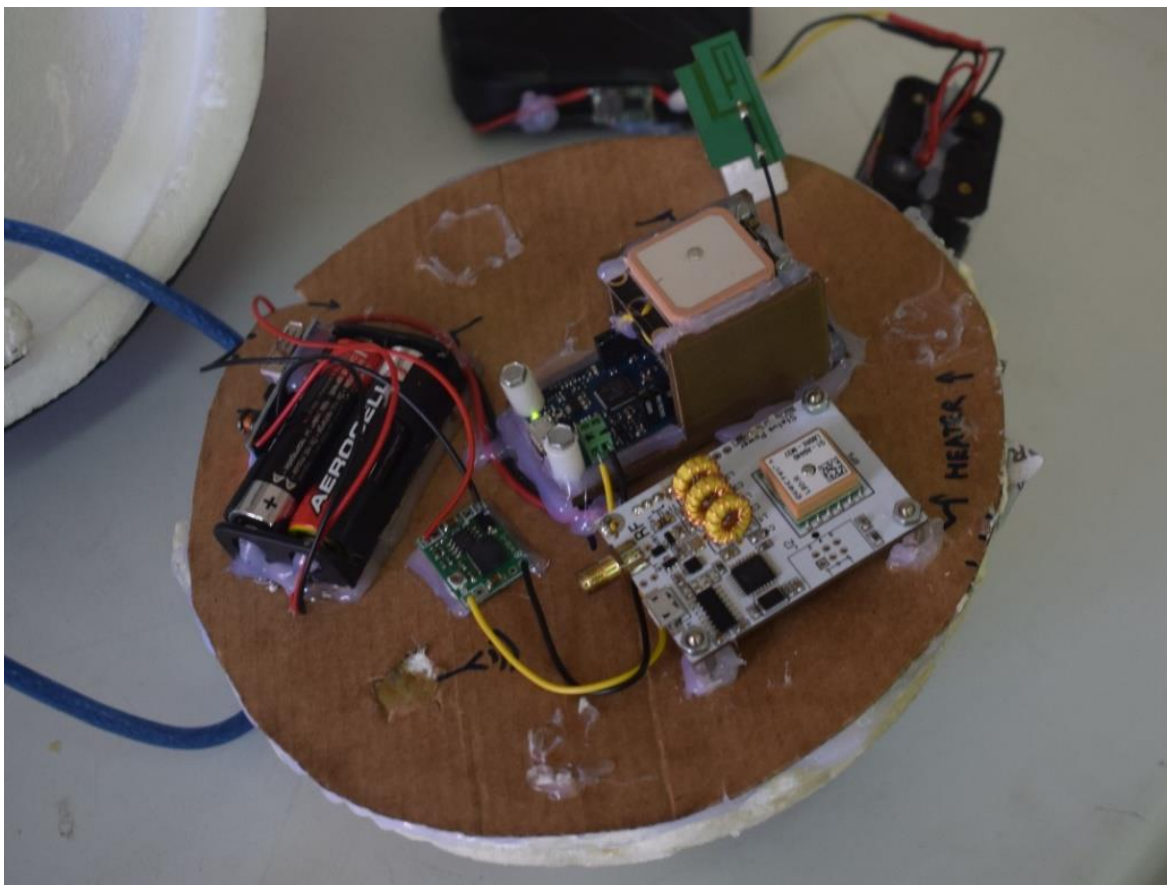


Figura 148 Electrónica sobre el molde

La esfera se puede cerrar de dos formas posibles, es importante que solo se cierre de una forma, para ello haremos una marca para saber cómo se deben encajar las dos semiesferas, este dato es muy importante puesto que la antena GPS y de comunicaciones de SigFox, no deben quedar cubiertas por el GPS *tracker* o por el Arduino, si esto sucede podrían darse problemas de cobertura, por ello, el módulo SigFox, va cerca del borde de la esfera, para que la electrónica que va en la parte superior, se coloque en el otro extremo en la medida de lo posible.

Una vez aclarado este importante detalle, procedo a explicar cómo se ha fijado el Arduino.

En esta aplicación, se han utilizado bastantes sensores y actuadores, por lo que el cableado, es bastante denso, por ello, he intentado seguir un código de colores para diferenciar Vcc, GND, I2C, 1WIRE, etc...

Una vez hecho esto, he intentado dejar el cableado lo más recogido posible, para ello he utilizado silicona caliente para evitar que los cables se salgan de los pines

durante el vuelo y también he utilizado bridas para mantener juntos los cables y que no ocupen demasiado volumen.

Es importante recordar que este paso solo se debe realizar una vez que el Arduino funciona a la perfección, ya que, de otra manera, si existe algún error, será muy difícil solucionarlo y requerirá mucho tiempo para deshacer el trabajo y luego volver a montarlo.

Una vez comprimido y firme todo el cableado, procedemos a fijar al bloque al cabo dispuesto en la parte superior, lo podemos realizar mediante bridas.

Una vez realizado este trabajo, el resultado ha sido el siguiente:

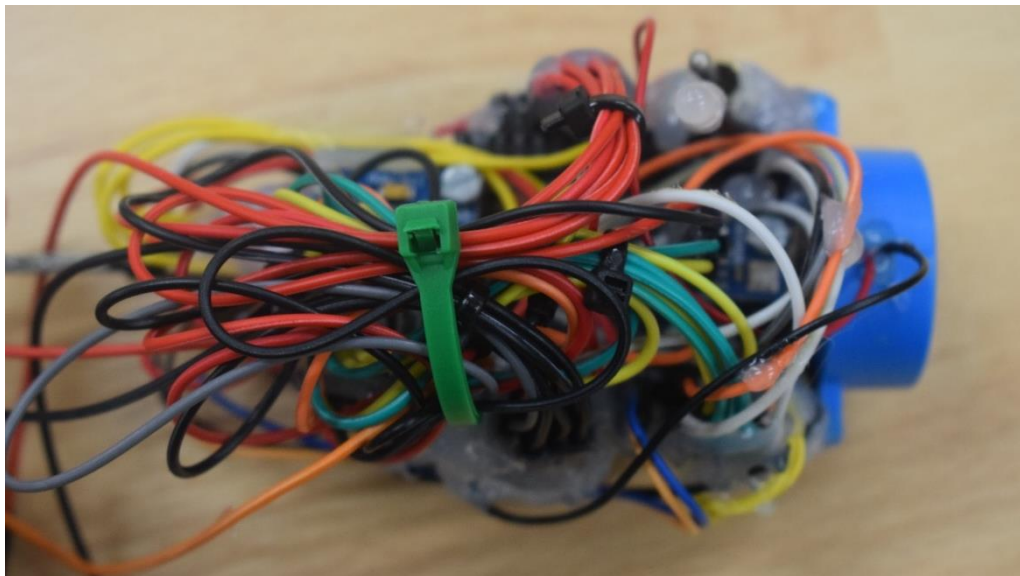


Figura 149 Arduino Mega

Posteriormente debemos fijar el GPS *tracker*, para ello dejaremos la ranura de la SIM, hacia la parte exterior, por si se tiene que realizar alguna modificación o comprobación de funcionamiento.

De forma que finalmente la solución quedará así:

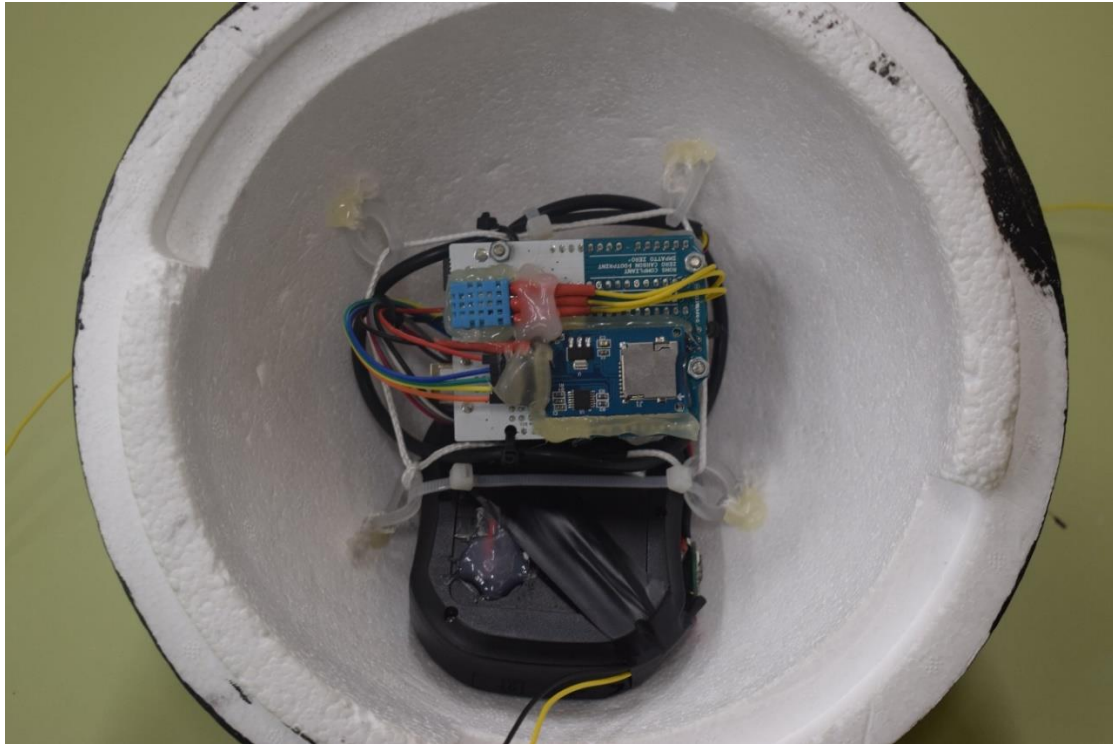


Figura 150 Parte superior payload box

De esta forma solo queda por fijar el porta baterías del GPS *tracker*, este paso se realizará justo antes del lanzamiento, puesto que estas baterías irán pegadas en la superficie de cartón y si lo dejamos ya pegado, dificulta el trabajo con la sonda.

Con todos estos pasos queda documentado como montar toda la electrónica en el interior y en el siguiente apartado se va a ver un pequeño estudio sobre el aislamiento de la sonda y como implementar un sistema para mantener una temperatura aceptable para la electrónica durante todo el vuelo.

5.3 CONDUCCIÓN DEL CALOR

Uno de los puntos más importantes de la fabricación de una sonda estratosférica, debe ser el aislamiento, sin embargo, para vuelos largos y con temperaturas externas inferiores a los -50°C , este aislamiento, puede que no sea suficiente para conseguir mantener una temperatura aceptable para la electrónica.

Este problema, puede fácilmente dejar la electrónica inoperativa y por lo tanto perder todo el material y la información, por ello he decidido buscar un sistema para

mantener una temperatura interna estable y por encima de los 0°C, pero primero se debe llevar a cabo un pequeño estudio sobre el aislamiento de nuestra sonda.

Para ello conecto los sensores del Arduino, cierro la esfera y aísló la junta de unión entre las dos semiesferas con cinta aislante, luego introduzco la sonda en un congelador a -24°C.

La primera prueba consiste en mantener la sonda durante una hora en el congelador.

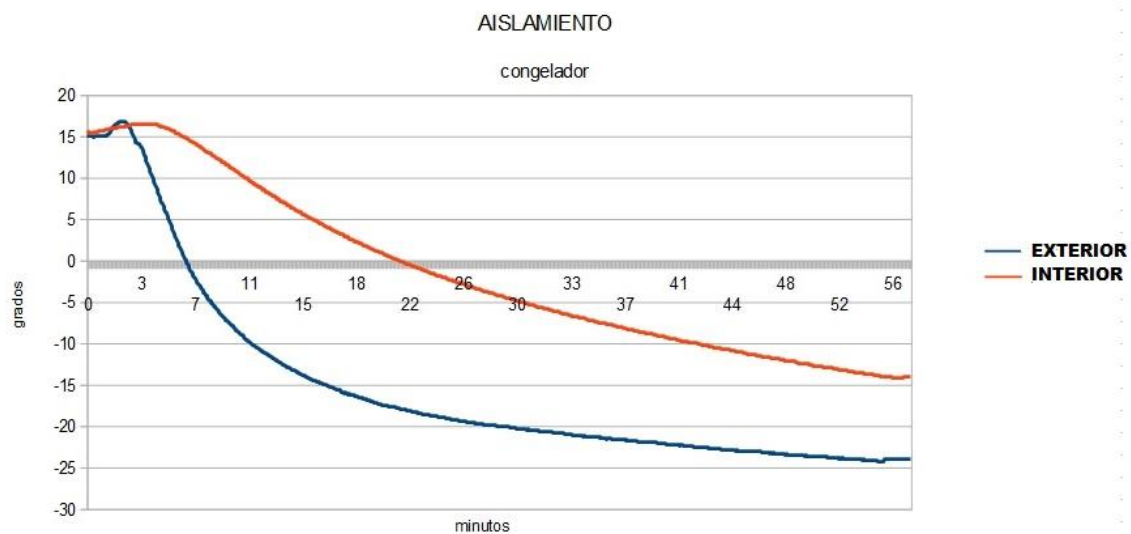


Figura 151 Aislamiento en congelador

El primer dato para tener en cuenta es la temperatura ambiente, esta es de unos 17°C aproximadamente, una vez que se introduce la sonda en el congelador, pasa repentinamente a una temperatura de -24°C.

Se puede observar como la curva azul, correspondiente a la temperatura externa, empieza a bajar rápidamente, hasta que, a partir de los 30 minutos, prácticamente se estabiliza entre los -20°C y los -24°C.

Por otro lado, se puede ver como la curva de las temperaturas internas, va disminuyendo de forma prácticamente lineal, uno de los puntos clave se produce al cabo de unos 22 minutos, cuando la temperatura en el interior es de 0°C.

A partir de ese momento, la temperatura sigue descendiendo hasta los -14°C.

El gradiente total de temperatura lo voy a tener en cuenta a partir de los 30 minutos, puesto que ambas curvas comienzan a tener menor pendiente y se estabilizan.

Con estas pautas, finalmente el gradiente oscila entre los 15°C y los 10°C.

Una vez realizada esta primera prueba, procedo a realizar una segunda prueba, esta vez, en el interior de la sonda estará funcionando un Arduino Uno con los sensores y una cámara Gopro filmando. La duración de la prueba esta vez será de una hora y media.

Los resultados son los siguientes:

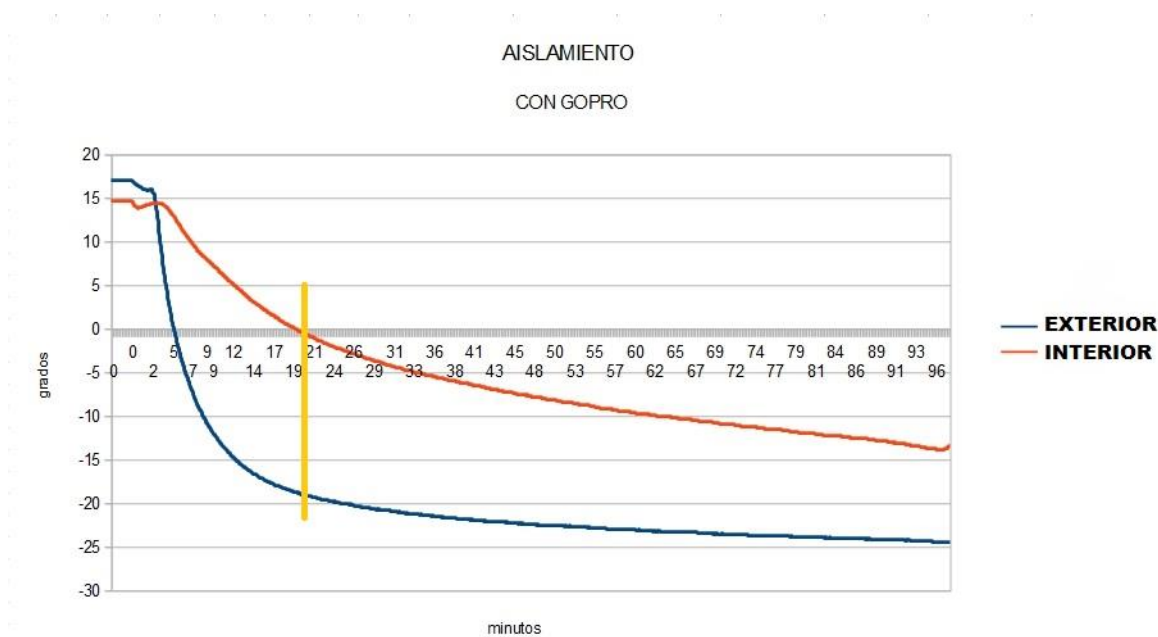


Figura 152 Aislamiento en congelador con cámara

Al inicio de la gráfica, se ve que la temperatura se mantiene sobre los 17°C, las diferencias entre los termómetros son debidas al grado de precisión y la velocidad de muestreo, además la prueba fue realizada justo después de la primera, por lo que los sensores están más fríos de lo habitual.

Si nos centramos en la curva de la temperatura externa, se ve como al principio cae en picado, alcanzando los 0°C, entre los 5 y los 6 minutos. Luego continúa bajando hasta estabilizarse entre los -20°C y los -24°C a los 30 minutos.

Con esta segunda prueba se confirma que la curva de las temperaturas externas de la primera prueba se puede tomar como buena.

A continuación, seguimos con la curva de la temperatura interna, sigue un perfil similar a la curva del primer experimento, la diferencia es que llega un poco antes a 0°C, sin embargo, por el margen de error, así como por otros factores comentados

anteriormente, se puede considerar que hasta alcanzar los 0°C, prácticamente son iguales.

Sin embargo, a partir de ese momento, la curva se aplana algo más que en el primer ejemplo, transcurridos unos 60 minutos, la temperatura interna es de -10°C, mientras que, en el primer experimento, esta era de -15°C.

Finalmente, a los 98 minutos, la temperatura interna es de -14°C y la externa es de -24°C.

Con esta prueba, se puede ver como con solo el uso de una Gopro, se consigue retardar el punto en que las temperaturas se estabilizan y llegan a un diferencial de 10°C, en este caso, se ha retardado casi 30 minutos.

Una vez analizados estos datos, se puede pensar en un sistema para aumentar la temperatura interna, de esta forma se conseguirá aumentar el diferencial y retardar la barrera de los 0°C en el interior.

Para ello, existen muchas opciones, tanto electrónicas como químicas, en este caso, para no complicar demasiado la electrónica y sobre todo para prevenir posibles incendios o cortocircuitos, he decidido usar un sistema químico.

En concreto este sistema consiste en una bolsa de tela muy gruesa y porosa, lo suficientemente porosa como para permitir la entrada de aire, pero sin permitir que salga el producto contenido en el interior.

Los componentes del interior son: Carbono activado, polvo de hierro, sal, agua y vermiculita⁷.

El funcionamiento consiste en una reacción química de oxidación debida al hierro, por lo que, en ausencia de oxígeno, la reacción se para, sin embargo, una vez que vuelve a estar presente el oxígeno, la reacción vuelve a llevarse a cabo.

La duración de la reacción en condiciones normales dura aproximadamente 12 horas, llegándose a alcanzar hasta los 40°C.

Ahora ya tenemos una solución para aumentar la temperatura, sin embargo, debido a la falta de oxígeno en las capas altas de la atmosfera y también debido a que la

⁷ Vericulita: La vermiculita es un mineral formado por silicatos de hierro o magnesio, del grupo de las micas.

sonda esta sellada, puede ser que la reacción pare en un determinado momento, por esta razón, se debe intentar que la energía desprendida en forma de calor llegue a los componentes electrónicos, para no desperdiciar esta energía.



Figura 153 Fuente de calor

Debido a los factores comentados anteriormente, la mejor forma de transmitir esta energía calorífica es a través de la conducción.

Para ello he diseñado dos estructuras, la primera de ellas la he comentado en el apartado anterior, pero en este apartado, voy a explicar su funcionamiento de forma práctica, además también voy a exponer una segunda estructura y su funcionamiento.

Comenzaré explicando una pequeña estructura diseñada como un armazón para el sistema SigFox y su GPS, que además está pensado para mantener una temperatura optima de esta placa, ya que es la más importante de todas.

Esta pieza ha sido diseñada en SolidWorks y ha sido impresa en el I.E.S Politécnico de Cartagena.

Es una pieza muy básica, tan solo consiste en una plataforma a la que se atornilla la placa y unas dos estructuras verticales en los laterales que protegen la antena GPS, así como el cableado de posibles golpes.

La intención de esta pieza no es solo transmitir el calor, sino que también conseguir que, si la fuente de calor está muy cerca, esta no haga que la placa alcance una temperatura excesivamente alta.

Para comprobar que el funcionamiento es el correcto, se han realizado varias pruebas, poniendo en contacto directo la fuente calorífica y la estructura 3D.

Los resultados dejan claro, como la temperatura máxima alcanzada en la superficie de la placa es de 30°C, este se puede considerar como una temperatura aceptable para el correcto funcionamiento.

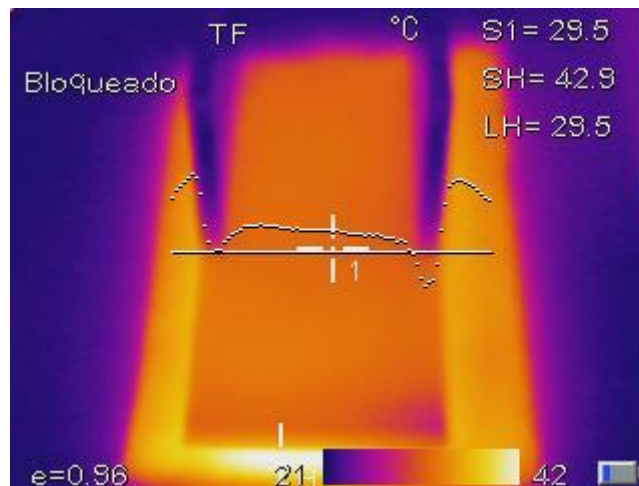


Figura 154 Térmica estructura 3D

La otra estructura usada es la que sirve de aislamiento entre las cámaras y el resto de electrónica, como ya hemos visto, se ha dejado un hueco en la parte comprendida entre la espuma y el cartón.

La intención es introducir en ese hueco la fuente de calor y que de esta forma caliente por conducción a las baterías y placas electrónicas que están pegadas al cartón.

Para comprobar el funcionamiento de este sistema, he realizado una última prueba en el congelador, la prueba llevada a cabo ha tenido una duración de 20 minutos.

Otro dato que se intenta obtener con esta última prueba es determinar el grado de aislamiento, ya que esta vez, las condiciones de la prueba serán las mismas que en el lanzamiento.

En esta prueba se han usado todos los dispositivos en funcionamiento excepto el WSPR, más adelante explicare porque he tomado esta decisión.

Otra observación es que he añadido dos fuentes de calor, la segunda, va directamente al aire y sirve para comprobar que sistema funciona mejor.

La temperatura ambiente es de unos 20°C, la temperatura en el interior del congelador es de -20°C.

Una vez introducida la capsula en el congelador, lo primero que se observa es que la temperatura interna va aumentando de forma progresiva, de forma que, a los 25 minutos, la temperatura en el interior llegó a alcanzar los 25°C.

Respecto a la curva de la temperatura externa, se ve como la temperatura desciende de forma brusca alcanzando los 0°C a los 8 minutos, teniendo en cuenta que los tres primeros minutos la sonda todavía estaba fuera.

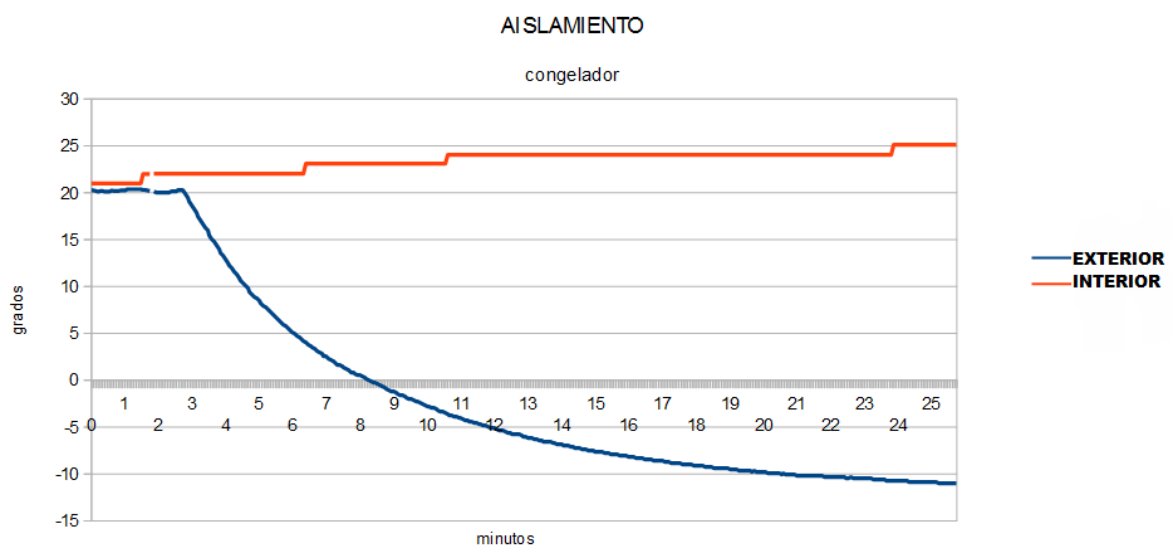


Figura 155 Simulación de misión en congelador

Luego la temperatura siguió descendiendo hasta alcanzar los -12°C al cabo de 25 minutos.

Con esta gráfica, se confirma el correcto aislamiento de la sonda y también el buen funcionamiento del sistema de calefacción, el cual rompe la tendencia y por primera vez crea una curva de temperaturas internas ascendentes, de esta forma se llega a alcanzar hasta un diferencial de temperaturas de hasta 36°C.

La siguiente parte de la prueba, consiste en extraer la sonda del congelador y tomar varias imágenes con la cámara térmica para observar cómo ha funcionado la transmisión del calor en la parte interna.

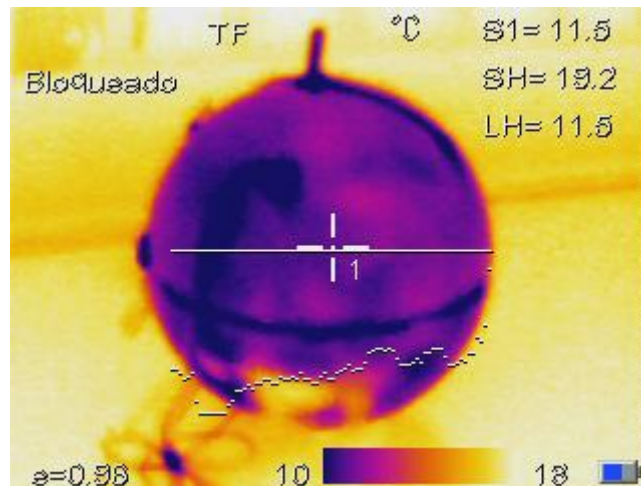


Figura 156 Térmica del exterior de la sonda

Una vez extraída la sonda, lo primero que se puede apreciar es la pérdida de calor por la parte inferior de la esfera, esta pérdida es producida por los dos agujeros de las lentes de las cámaras, son los dos puntos débiles del aislamiento.

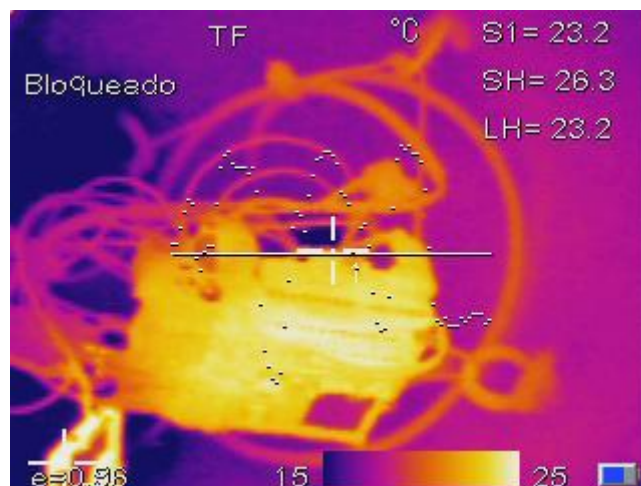


Figura 157 Térmica Arduino interior sonda

Una vez abierta la capsula, he empezado a revisar la temperatura de los componentes electrónicos comenzando con el Arduino, en este caso, los puntos más calientes alcanzan los 26°C, esta medida coincide con la tomada por los sensores.

La temperatura de los componentes instalados en la parte superior de la capsula, se calientan por convección, sin embargo, los componentes están muy cerca unos de otros y casi no hay espacio libre, por ello es fácil que se calienten también por convección.

A continuación, expongo las imágenes tomadas de la parte inferior de la esfera, esta es la parte más importante, puesto que, en ella, se encuentran instaladas las baterías y el resto de la electrónica.

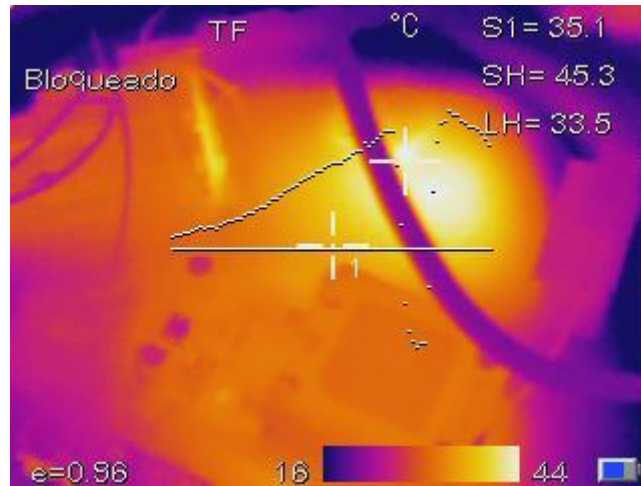


Figura 158 Térmica superficie electrónica

En esta imagen, se aprecia que en la zona donde está el calentador que funciona con convección, la temperatura asciende hasta los 45°C, mientras que el resto de la superficie del cartón, así como SigFox y la fuente de alimentación se encuentran a unos 29°C aproximadamente, se puede apreciar una zona un poco más fría, esta corresponde a la zona donde está dispuesto el sistema WSPR, el cual está apagado, esto hace que la zona también se encuentre algo más fría.

Sin embargo, si hacemos *zoom* en esa zona podemos observar lo siguiente.

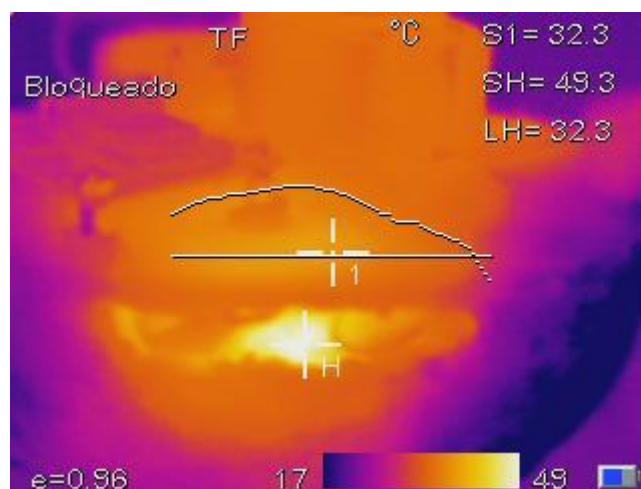


Figura 159 Térmica molde poliuretano

Se ve como la placa del WSPR está a unos 23°C, a pesar de no estar en funcionamiento, de esta manera, se comprueba de forma efectiva la eficacia del sistema de conducción, por otro lado, cabe destacar que la fuente de calor por convección fue retirada una vez que la capsula salió del congelador, por lo que se puede comprobar que una solo fuente de calor es capaz de calentar toda la capsula de forma eficiente.

Por último, muestro las dos cámaras de video, tras la prueba, ambas cámaras alcanzan temperaturas superiores a lo esperado, y por las imágenes que se han tomado con la cámara térmica, se puede ver como la fuente de calor también llega a los huecos de las cámaras a través del poliuretano, esto es debido a que los huecos donde están las cámaras tienen tan solo unos 2 (cm) de margen con la fuente de calor.

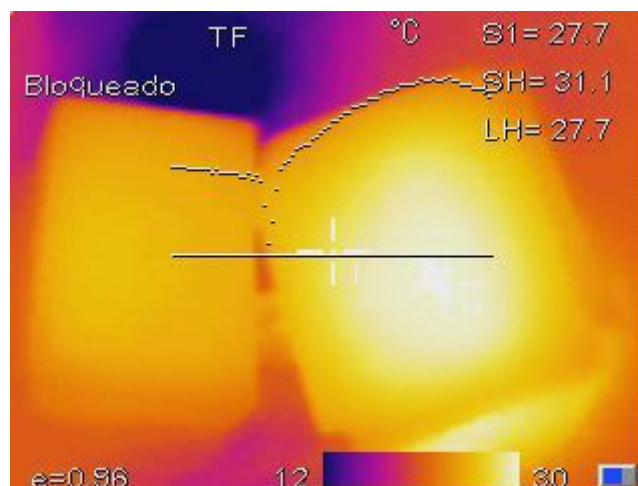


Figura 160 Térmica cámaras

Se puede ver que la cámara SJ está trabajando a unos 27°C mientras que la Gopro asciende hasta los 31°C, aunque esta temperatura es alta, durante el vuelo, las temperaturas serán inferiores a las del congelador y esto hará que las cámaras trabajen a una temperatura un poco más baja.

Finalmente, de este pequeño estudio, queda claro que el sistema de aislamiento funciona como se esperaba y que el sistema de calefacción funciona mucho mejor de lo esperado.

Sin embargo, durante el vuelo, para aligerar, así como para prevenir exceso de temperatura, se va a utilizar solo la fuente de calor por conducción, puesto que dos

fuentes podrían aumentar demasiado la temperatura y también por el hecho de que utilizar una fuente que caliente por convección significaría desaprovechar muchísima energía.

6. PROPULSIÓN Y REENTRADA



Una vez terminados los apartados de Aviónica y de *Payload Box*, llega el momento de determinar cuál va a ser el sistema por el cual nuestro *Payload* será impulsado hasta los 30.000 (m).

Una vez que nuestro *payload* alcance su techo, entonces comenzará una caída libre que lo llevará al punto de aterrizaje, pero por razones de seguridad, así como por el motivo de intentar recuperar la sonda en una sola pieza, deberemos también determinar un sistema para controlar la caída.

Por lo tanto, este apartado ha sido dividido en dos partes, en la primera se determinará como elegir y realizar los cálculos para un sistema de propulsión.

Por otro lado, en el segundo apartado, se explicará cómo elegir un paracaídas adecuado, así como otros datos de interés.

Antes de entrar en materia, es esencial tener en cuenta que una de las primeras preguntas que uno tiene que realizar a la hora de abordar este proyecto, es el peso que quiere enviar a la estratosfera.

Lo normal es que se determine un peso y que tratemos de cumplir este objetivo.

Para ello, pesaremos cada uno de los componentes que utilicemos en la sonda: la propia sonda, las baterías, incluso el cabo que utilicemos para unir la sonda con el paracaídas y el sistema de propulsión.

Se tiene que llevar un control exhaustivo del peso y tener siempre en cuenta que no se puede exceder del peso propuesto inicialmente.

En concreto los globos que se utilizan, están estandarizados para unos márgenes de peso y además tienen un precio bastante elevado, por lo que, si no se cumplen los márgenes de pesos, deberemos buscar un globo más grande y por lo tanto mucho más caro.

En concreto, en esta aplicación hemos estudiado el catálogo disponible y hemos adquirido un globo de 1.300 (g) de *payload*, por lo que, desde el principio del proyecto, nos hemos ajustado estrictamente a este margen.

Una vez explicada la importancia del peso, prosigo con los dos apartados.

6.1 SUBSISTEMA DE PROPULSIÓN

Para alcanzar la estratosfera, existen varios métodos, como cohetes, globos solares, etc...

El método utilizado en este caso ha sido usando un globo de helio.

El primer paso para determinar que globo necesitamos, lo debemos realizar mediante un estudio de los modelos existentes en el mercado.

Los principales fabricantes se encuentran en Asia y normalmente no venden los globos de forma individual, por lo que se deben comprar normalmente en paquetes de 3 o más globos, puesto que estos tienen un precio considerable, no debemos excedernos en el peso que queremos enviar, debemos hacer un estudio meticuloso y realista de los medios disponibles y de los objetivos.

Si se realiza un buen examen de recursos y objetivos, pronto llegamos a la conclusión de que no es necesario que nuestro globo sea capaz de elevar mucho peso, en mi caso, solo debe elevar 1.200 (g), por lo que al consultar los catálogos hemos optado por adquirir el modelo SB-1000.

En concreto este modelo es capaz de levantar hasta 1.310 (g), alcanza una altura máxima de 31.000 (m) y tiene un peso de 1.000 (g).

Una vez hemos seleccionado el modelo, el siguiente paso consiste en asegurarse de que el peso del conjunto no exceda los 1.310 (g), es decir, el peso de la sonda con toda la electrónica, el paracaídas y las líneas de unión entre los componentes, no deben exceder ese peso.

Es decir, si pesamos todo el conjunto incluido el globo, este pesaría como máximo 2.310 (g)

Sounding Balloon Model #	SIZE (g)	LATEX TYPE	WEIGHT (g)	NECK Length (cm)	NECK Diameter (cm)	Uninflated Diameter (cm)	Burst Diameter (cm)	Inflation for Inspection Diameter (cm)	Free Lift (g)	Average Bursting Altitude (m)	Average Rate of Ascent (m/min)
Sounding Balloon SB-350	350	NATURAL LATEX	350 ~ 390	≥10	≤6.4	83~95	> 410	≥270	825	20000	≥340
Sounding Balloon SB-400	400		400 ~ 440	≥10	≤6.4	90~101	> 450	≥280	875	21000	≥340
Sounding Balloon SB-500	500		500 ~ 550	≥10	≤6.4	108~121	> 500	≥380	905	25000	≥340
Sounding Balloon SB-600	600		600 ~ 650	≥10	≤6.4	121~140	> 580	≥400	1120	26000	≥340
Sounding Balloon SB-750	750		750 ~ 800	≥11	≤6.4	137~162	> 650	≥450	1220	28000	≥340
Sounding Balloon SB-800	800		800 ~ 850	≥11	≤6.4	140~162	> 680	≥470	1220	29000	≥340
Sounding Balloon SB-1000	1000		1000 ~ 1100	≥11	≤7.7	153~172	> 750	≥500	1310	31000	≥340
Sounding Balloon SB-1200	1200		1200 ~ 1300	≥11	≤8.3	172~191	> 850	≥550	2240	32000	≥340
Sounding Balloon SB-1500	1500		1500 ~ 1650	≥12	≤8.3	178~197	> 950	≥600	2330	33000	≥340
Sounding Balloon SB-1600	1600		1600 ~ 1700	≥13	≤8.3	190~210	> 1050	≥600	2400	36000	≥340
Sounding Balloon SB-2000	2000		2000 ~ 2100	≥13	≤8.3	195~220	> 1100	≥600	2500	38000	≥340

Figura 161 Catálogo globos sonda



Figura 162 Globo sonda

6.1.1 CALCULO DE HELIO

Una vez que se ha elegido el globo, el paracaídas y también se tiene claro el peso de la sonda, se puede proceder al cálculo del helio necesario para elevar la capsula hasta la estratosfera.

Este punto es uno de los más importantes, puesto que el helio es un recurso limitado y caro, por lo que, si calculamos mal el helio, podríamos quedarnos sin vuelo y además haber desperdiciado el que hemos introducido en el globo.

Para realizar el cálculo del helio, son necesarias unas nociones mínimas de física.

Según el principio de Arquímedes, un cuerpo total o parcialmente sumergido en un fluido en reposo recibe un empuje de abajo hacia arriba igual al peso del volumen del fluido que desaloja.

Si realizamos un estudio de equilibrio de fuerzas, llegamos a la conclusión de que deben existir dos fuerzas, una de ellas será la fuerza gravitatoria, la cual atrae todo el conjunto hacia la superficie terrestre.

Por otro lado, existe otra fuerza, la cual separa el conjunto de la superficie terrestre, esta es la fuerza ascensional, la descrita por Arquímedes.

Es decir, para conseguir un equilibrio de fuerzas se debe cumplir lo siguiente:

$$\vec{F}_{ascensional} - \vec{F}_{gravitatoria} = 0 \text{ (N)}$$

$$\text{Masa de aire que ocupa el volumen desalojado} * g = M_{total} * g$$

Si tenemos en cuenta la relación entre la densidad, masa y volumen...

$$\rho = \frac{m}{V} \rightarrow m = \rho * V$$

Ahora podemos sustituir en la formula del equilibrio de fuerzas.

$$\rho_{aire} * V_{globo} * g = M_{total} * g$$

Simplificando queda lo siguiente:

$$\rho_{aire} * V_{globo} = M_{total}$$

Si ahora desglosamos la masa total del conjunto M, en cada uno de sus términos, obtendremos la siguiente ecuación.

$$\rho_{aire} * V_{globo} = m_{sonda} + m_{cuerda} + m_{paracaidas} + m_{globo} + m_{helio}$$

De esta ecuación, podemos sustituir el término de la masa del helio con la relación de densidad y volumen, de esta forma la ecuación quedaría así:

$$\rho_{aire} * V_{globo} = m_{sonda} + m_{cuerda} + m_{paracaidas} + m_{globo} + \rho_{helio} * V_{globo}$$

Agrupamos términos y obtenemos la siguiente ecuación.

$$(\rho_{aire} * V_{globo}) - (\rho_{helio} * V_{globo}) = m_{sonda} + m_{cuerda} + m_{paracaidas} + m_{globo}$$

$$V_{globo} * (\rho_{aire} - \rho_{helio}) = m_{sonda} + m_{cuerda} + m_{paracaidas} + m_{globo}$$

Sustituyendo los términos de densidad, obtenemos el siguiente resultado:

$$V_{globo} * (1,19 - 0,1785) = m_{sonda} + m_{cuerda} + m_{paracaidas} + m_{globo}$$

Teniendo en cuenta que la densidad del aire a 0°C es de 1,29 (Kg/m³) y que a 50°C es de 1,09 (Kg/m³), podemos establecer que a 25°C, es aproximadamente de 1,19 (Kg/m³), por lo que al restar a la densidad del aire la densidad del helio, el resultado lo podemos aproximar a 1.

Por lo tanto:

$$V_{globo}(m^3) = (m_{sonda} + m_{cuerda} + m_{paracaidas} + m_{globo})(Kg)$$

Si sustituyo los términos de masas, con los datos correspondientes de mi trabajo, se obtiene lo siguiente:

$$V_{globo}(m^3) = (1,100 + 0,055 + 0,035 + 1,063)(Kg)$$

$$V_{globo}(m^3) = (2,25)(Kg)$$

Es decir, para mantener mi carga en equilibrio, necesitaría unos 2,25 (m³) de helio, pero si quiero que mi globo ascienda, necesito que la fuerza ascensional sea mayor a la fuerza gravitatoria, teniendo en cuenta que el término de la izquierda corresponde con la fuerza ascensional, entonces obtenemos la siguiente relación:

$$V_{globo}(m^3) + He_{adicional} > (m_{sonda} + m_{cuerda} + m_{paracaidas} + m_{globo})(Kg)$$

Esto significa que como mínimo, debemos añadir lo calculado en la ecuación anterior, pero además debemos añadir una cierta cantidad de helio extra para que el globo ascienda, calcular esta cantidad de helio extra, es la parte más importante.

Si la cantidad de helio extra es muy poca, el globo ascenderá muy lento y tardará mucho en llegar a la altura de explosión, debido a los vientos en altura, podría darse el caso de que el globo aterrice a cientos o miles de kilómetros.

Esta técnica es utilizada para que los globos recorran grandes distancias incluso cruzar océanos, pero en este caso no es de utilidad para nosotros.

Por otro lado, si la cantidad de helio extra es excesiva, el globo subirá muy rápido y explotará antes de llegar a la altura de techo fijada por el fabricante, sin embargo, se alejará muchísimo menos del punto de lanzamiento.

Nosotros optamos por la segunda opción, pero debemos determinar con exactitud qué cantidad de helio extra necesitamos.

Este cálculo es bastante complejo, puesto que la atmosfera no es un fluido en reposo, por ello, las densidades, presiones, temperaturas y vientos cambian en función de la altura de forma no lineal.

Para hacer un cálculo preciso sería necesario aplicar una serie de ecuaciones diferenciales, por la complejidad del cálculo y el grado de incertidumbre de la solución, la opción seguida para solucionar este cálculo ha sido utilizar otros modelos matemáticos accesibles a través de la red.

Estos calculan la cantidad de helio necesario, para ello primero se deben introducir varios datos, luego el programa hace el resto, sin embargo, se debe realizar una elección importante.

La elección consiste en determinar si quieres calcular helio para alcanzar una determinada altura o bien si prefieres calcular el helio para que el globo suba a una velocidad determinada.

En función de nuestras necesidades debemos seleccionar una u otra opción, en internet existen varias páginas donde poder realizar estos cálculos, aquí voy a explicar dos páginas, la primera de ellas es:

http://randomaerospace.com/Random_Aerospace/Balloons.html

En esta *web* aparece una pequeña calculadora donde podremos introducir los datos de nuestro globo, en este caso solo da la opción de calcular el helio para alcanzar una velocidad determinada.

Meteorological Balloon Burst Estimator

1) Choose Balloon: (Burst Diameter: m Weight: g Cd:)

2) Choose Lifting Gas: (Gas Density: Kg/cu m)

3) Enter weight to be lifted: g (include weight of Payload + Parachute and Line etc.)

4) Enter desired average ascent rate: m/s (4.0 to 6.0m/sec is typical)

5) Click to

Estimated burst altitude: m (ft)

Gas volume required: cu m (cu ft)

Neck lift: Kg (lb)

Time to burst: min (hrs)

Figura 163 Simulador helio

En el primer campo, debemos selecciona en el desplegable el tipo de globo, en este caso, es un HY-1000, en el segundo campo, se debe seleccionar el tipo de gas que se va a utilizar.

Luego en el campo tres, se debe introducir el peso de la carga de pago, aquí se incluye todo el peso excepto el del globo.

En el campo cuatro, se debe introducir la velocidad de subida deseada, esta debe ser introducida en (m/s).

Finalmente pulsamos en “*calculate*” y obtendremos los datos deseados.

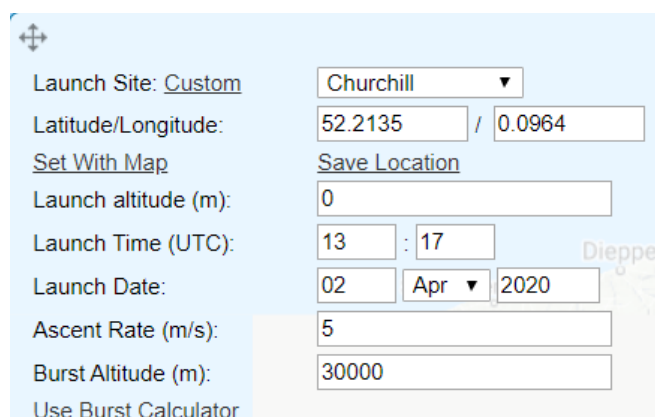
Podremos ver la altura de explosión, el gas requerido y tiempo hasta la explosión.

Es importante recordar que, para este caso, el volumen necesario es de 4,16 (m^3), por lo que si anteriormente se ha calculado un total de 2,25 (m^3), ahora, se puede calcular que el helio adicional será 1,91 (m^3).

A continuación, presento la segunda página que he utilizado, esta sí que permite determinar una altura máxima o una velocidad de ascenso, el enlace es el siguiente:

<http://predict.habhub.org/#!/uuiid=ac717ac17f1f99a7d88c0b8e2785ef7a5756565c>

Una vez que accedamos al enlace, llegaremos a una ventana principal, en la esquina inferior derecha, podemos ver el recuadro siguiente:



Launch Site: [Custom](#)

Latitude/Longitude: /

[Set With Map](#) [Save Location](#)

Launch altitude (m):

Launch Time (UTC): :

Launch Date:

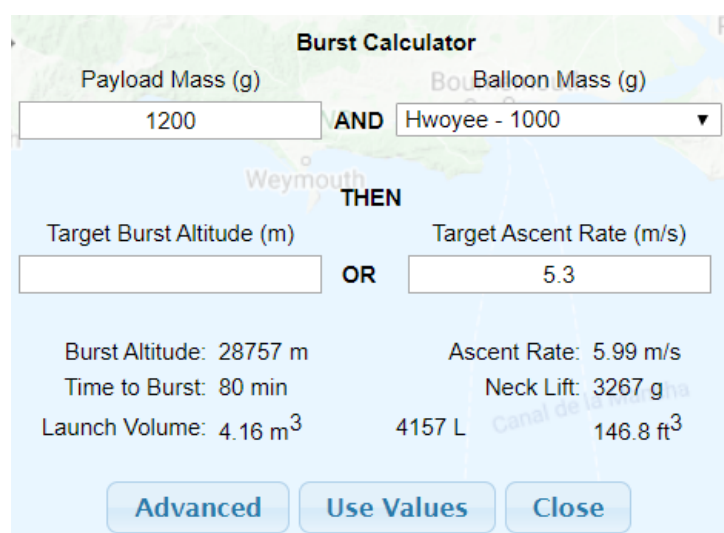
Ascent Rate (m/s):

Burst Altitude (m):

[Use Burst Calculator](#)

Figura 164 Simulador CUSF

Debemos pulsar sobre el enlace de “*Use burst calculator*”, una vez accedamos, aparecerá la siguiente ventana.



Burst Calculator

Payload Mass (g) AND Balloon Mass (g)

THEN

Target Burst Altitude (m) OR Target Ascent Rate (m/s)

Burst Altitude: 28757 m Ascent Rate: 5.99 m/s

Time to Burst: 80 min Neck Lift: 3267 g

Launch Volume: 4.16 m³ 4157 L 146.8 ft³

[Advanced](#) [Use Values](#) [Close](#)

Figura 165 Simulador helio CUSF

Los primeros datos que debemos introducir son la carga de pago y el modelo del globo, luego debemos seleccionar una altura máxima deseada o una velocidad de subida deseada.

Cuando usamos la velocidad de subida, existe un pequeño error, ya que, si nos fijamos en los resultados, vemos que la velocidad de subida es mayor a la introducida por nosotros, así que debemos jugar un poco con esta cifra hasta alcanzar la velocidad deseada.

Es decir, al introducir una velocidad de 5,3 (m/s), los metros cúbicos necesarios son 4,16 (m³) y también aparece que la velocidad de subida es de 5,99 (m/s).

Estos 5,99 (m/s), es el dato real con el que tenemos que hacer los cálculos.

Una vez que tengamos claro la cantidad de helio necesario, podemos proceder a la compra de las botellas de helio que necesitemos.

En este caso, hemos necesitado un total de 4,6 (m³), por lo que hemos utilizado una botella de 3,6 (m³) y otra de 1 (m³)

En el apartado de lanzamiento, explicaré detalladamente como de debe inflar el globo.

6.1.2 UNIÓN SONDA/GLOBO

Para unir la sonda con el globo, vamos a utilizar un cabo de nylon, este debe ser de unos 30 (m) si no hace nada de aire, en caso de que el viento sople, se debe recortar un poco la longitud.

Gracias a esta longitud, se consigue que la sonda no se mueva de forma brusca y se puedan obtener buenas imágenes durante el ascenso.

En mi caso, he utilizado un cabo de unos 20 (m) debido al viento en el día del lanzamiento, pero para cualquier caso, los pasos a seguir serían los siguientes:

Primero de todo, se debe cortar un cabo de unos 30 (m), en caso de que el día del lanzamiento sople el viento, podremos cortarlo para ajustarlo a la medida que sea necesario.

Una vez que tengamos cortado el cabo, debemos quemar ligeramente con un mechero los extremos de este para que no se deshilen.

Una vez listos los dos extremos, deberemos realizar un nudo tipo “As de guía” u “Ocho doble” en uno de los dos extremos.

Para hacer un as de guía, se pueden seguir los siguientes pasos:

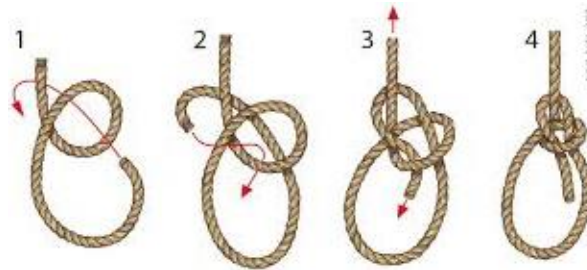


Figura 166 As de guía

Mientras que para realizar un “Ocho doble” se deben seguir estos pasos:

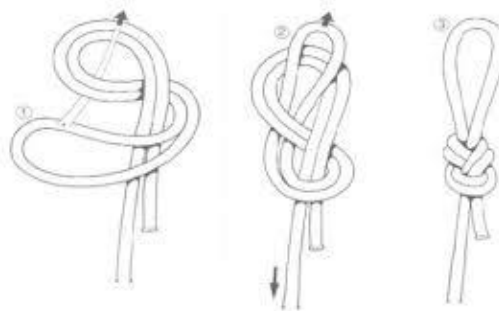


Figura 167 Ocho doble

Una vez seleccionado el nudo a utilizar, deberemos enganchar la coca del nudo en el quitavueeltas, mientras que el otro extremo de este deberá ir fijado a la sonda, mediante los cuatro cabos que sobre salen de esta.

Por último, solo faltaría explicar la unión del otro extremo del cabo al globo, pero para ello, en este apartado solo debemos tener en cuenta cual será el nudo a utilizar, el resto se explicará en el proceso de llenado del globo.

El ultimo nudo a realizar será un “Ballestingue con cote”

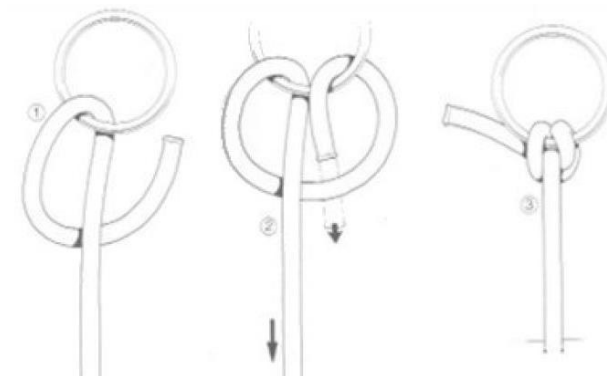


Figura 168 Ballestingue

Se seguirán los pasos anteriores, pero dejando el extremo que queda libre con bastante más longitud, de esta manera con el sobrante se practicarán varios nudos simples alrededor del cabo vertical, de esta forma se evita que el nudo se pueda soltar.

6.2 SUBSISTEMA DE REENTRADA

Una vez seleccionado el globo y teniendo en cuenta el peso máximo de la sonda, podremos elegir un paracaídas apropiado para la misión, al igual que para los globos, existe una gran variedad de estos en tiendas de modelismo espacial.

Estos paracaídas están por lo general estandarizados, de forma que se miden en pulgadas, en el catálogo, podemos ver qué pesos admiten, así como otros datos referentes a velocidades de descenso, para mi caso, he utilizado un paracaídas de 30".

En concreto este modelo mide unos 76 (cm), tiene forma de octógono y posee 8 líneas de nylon.



Figura 169 Paracaídas

Es capaz de parar una caída para objetos comprendidos entre los 800 (g) y los 1.400 (g), la velocidad de descenso dependerá del peso de la carga, según el fabricante, esta va desde los 5 (m/s) para 800 (g) y hasta 7 (m/s) para 1.400 (g).

Una vez seleccionado este paracaídas, podemos ver como se unirá a la sonda y como se unirá también el globo.

6.2.1 UNIÓN SONDA/PARACAIDAS

Una vez lista la unión de la sonda al globo, ya solo falta unir el paracaídas al sistema para terminar todo el conjunto.

Existen muchas formas de realizar este proceso, sin embargo, voy a comentar dos que parecen las más fiables.

La primera de ellas consiste en realizar una pequeña perforación en el centro del paracaídas, debe ser un agujero pequeño, lo justo para que pase el cabo que une el globo con la sonda.

La recomendación es utilizar un soldador para que a la vez que se realiza el agujero, queme las fibras de alrededor, de forma que lo vuelva más resistente a desgarros, de todas formas, al tener un cabo con tensión en el centro y que seguramente va a recibir fuertes tirones, es necesario reforzar la tela del paracaídas, para ello se puede utilizar unos parches especiales para reparar velas de barcos, es un material muy ligero y resistente, en la imagen se aprecia el parche de color azul.



Figura 170 Refuerzo paracaídas

Una vez realizado este paso, debemos coger los extremos de los 8 cabos del paracaídas, los cuales están en doble, es decir, está formado por cuatro cabos principales, los extremos de estos van a parar a cada uno de los ocho vértices del octógono, así que debemos coger los cuatro puntos centrales de estos 4 cabos y unirlos.

Para realizar la unión, debemos realizar un nudo de “alondra”

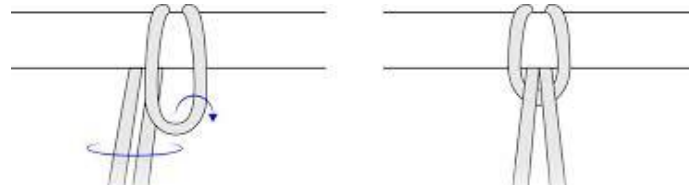


Figura 171 Alondra

Es un nudo muy sencillo, solo debemos seguir los pasos anteriores, pero tenemos que tener en cuenta que en el dibujo solo aparece un cabo y nosotros tenemos cuatro cabos, sin embargo, la forma de proceder es la misma pero con los cuatro cabos a la vez, es obvio que una vez que se va a terminar de hacer el nudo, se tiene que pasar el paracaídas por completo a través de las cuatro cocas, solo se debe tener cuidado de que pase por estas sin que se lie, luego, se tira de los cabos para que el nudo coja tensión y se quede firmemente amarrado.

Ahora solo faltaría determinar dónde se va a amarrar ese nudo de “alondra”.

Para realizar este último paso, debemos hacer un nudo tipo “Ocho doble” a unos dos metros y medio del quita vueltas de la sonda, de esa manera, en la línea principal que une el globo a la sonda, aparece un punto de unión en el que realizar el nudo de “Alondra”.

Cuando se realiza este último nudo, ahora solo falta introducir por el centro del paracaídas el extremo de la línea principal que va al globo.

De esta manera, mientras que el globo este subiendo, el paracaídas no se desplegará, pero una vez que el globo explote, el paracaídas se inflara, sus cabos se tensaran y entonces empezara a frenar.

Este sistema a priori es muy sencillo y fiable, sin embargo, existe un gran riesgo si el globo no se desintegra en la explosión, como el globo quedaría fijo en el extremo superior y este es muy pesado, una vez que comienza el descenso, existe el riesgo

de que el globo caiga rápidamente junto con la sonda y haga que no se despliegue el paracaídas.

La segunda opción es algo más compleja y consiste en crear una segunda línea para el paracaídas, es decir, se crea una línea principal que une el globo con la sonda y una línea secundaria para unir la sonda con el paracaídas.

Para crear esta segunda línea, cortaremos un trozo de unos dos metros y medio de cabo, en sus extremos realizaremos dos “As de guía” y al igual que en el resto, nos aseguraremos de quemar los extremos y de realizar un cote de seguridad para que no se deshaga el nudo.

Uno de los extremos lo debemos unir al quitavueeltas al igual que está unida la línea principal.

En el otro extremo, debemos realizar el nudo de “alondra” en la coca del “as de guía” como ya hemos visto anteriormente.

Luego debemos poner las dos líneas desplegadas en paralelo y en la línea principal a la altura del paracaídas, debemos añadir un pequeño trozo de cabo de uno 30 o 40 (cm), para unir este extremo a la línea principal, podemos realizar un “Ballestrinque” y varios cotes, en el otro extremo debemos amarrar una goma del pelo que sea gorda, por ella deberemos introducir la tela del paracaídas, de forma que lo deje pillado y no pueda desplegarse hasta que empieza a caer.

La idea es que mientras está subiendo el globo, ambas líneas giran solidariamente gracias al quita vueltas y debido a la goma del pelo, el paracaídas no puede abrirse, cuando finalmente el globo explota, entonces el paracaídas empieza a inflarse y finalmente escapa de la goma que lo aprisionaba, de esta forma el paracaídas se desplegaría por completo.

En caso de que el globo no se desintegrara, no afectaría al paracaídas, puesto que el globo caería hacia abajo con su línea de cabo, pero el paracaídas permanecería abierto en la parte superior frenando la sonda gracias a su segunda línea de cabo.

De los dos casos anteriores, he utilizado el segundo sistema, así que he a parte de los pasos anteriores, también he tenido que tapar el agujero del paracaídas, para ello he utilizado otro parche azul, pero sin hacerle el agujero.

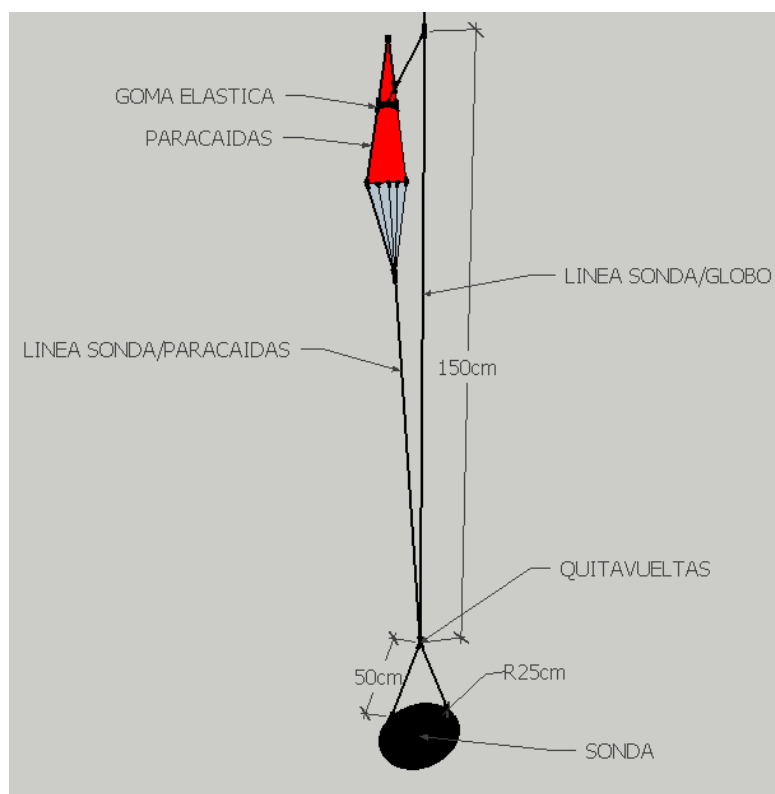


Figura 172 Esquema sonda-paracaídas

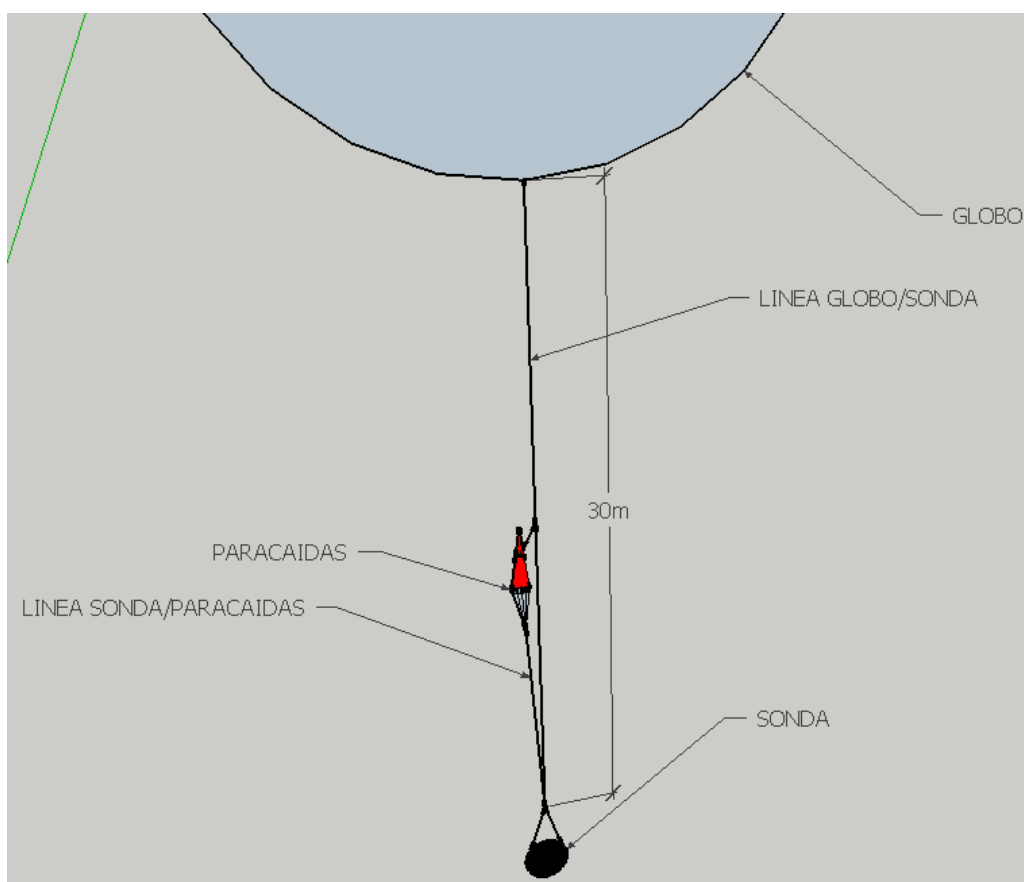
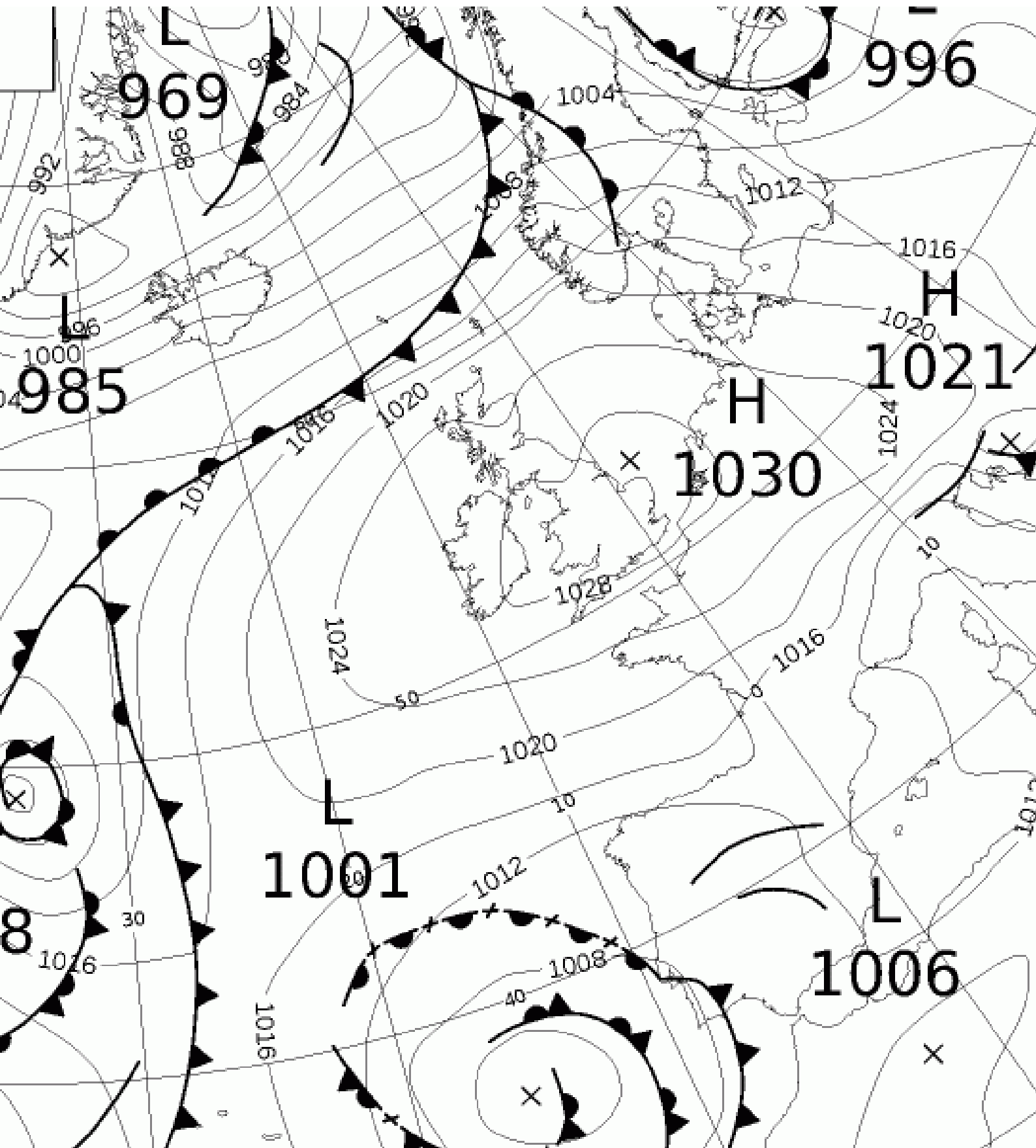


Figura 173 Esquema sonda-globo

7. PREPARACIÓN DEL LANZAMIENTO



7.1 SIMULACIÓN DE VUELO

Una vez que se ha terminado la fabricación de la sonda, solo queda pendiente determinar el lugar y la fecha del lanzamiento.

Una recomendación es elegir primero el lugar de aterrizaje y posteriormente ajustar el lugar de lanzamiento para que caiga en el lugar previsto, normalmente los lanzamientos se realizan al revés y muchos de estos lanzamientos son fallidos ya que la sonda cae en zonas sin cobertura, remotas, montañosas o incluso prohibidas.

7.1.1 ELECCION LUGAR DE ATERRIZAJE

Para determinar un lugar apropiado para el aterrizaje, debemos fijarnos en cuatro factores principales, estos son: Densidad de población, orografía, cobertura de los sistemas y accesibilidad.

Respecto a la densidad de población, es importante que la zona esté muy poco poblada, de esta forma será muy difícil que podamos causar daños personales o materiales, además también será menor la probabilidad de que alguien encuentre la sonda antes que nosotros.

Por ello una de las mejores zonas en España es la comunidad autónoma de Castilla la Mancha, en concreto las provincias de Albacete, Cuenca y Ciudad Real.

En concreto, la provincia de Cuenca es la que tiene la menor densidad de habitantes por (Km²), son 12,33 habitantes por Km².

Si ahora nos centramos en la orografía, es importante que sea una zona muy llana, de esta forma, el acceso a la zona no será difícil.

Para esto, la zona suroeste de Cuenca es un lugar muy apropiado.

Respecto a la cobertura, debemos acceder al mapa del operador SigFox y de telefonía y buscar por la zona lugares con cobertura plena.

En concreto, he buscado por la zona comprendida entre Quintanar de la Orden y Los Hinojosos, los mapas de cobertura son los siguientes:



Figura 174 Cobertura SigFox Quintanar de la Orden

En esta zona la cobertura es bastante buena, solo existen unas pequeñas manchas sin cobertura lejos de la zona deseada.

Respecto a la cobertura de telefonía móvil, esta es total para 2G, pero algo menor para 3G y muy limitada para 4G, sin embargo, cerca de las carreteras que unen núcleos urbanos, suele encontrarse cobertura 3G.

Por último, debemos además tener en cuenta las carreteras, vías de tren, etc...

La idea es que sea una zona con bastantes carreteras o caminos que nos permitan acercarnos mucho al objetivo antes de tener que empezar a andar.

Teniendo en cuenta todo lo anterior, la zona seleccionada para realizar el aterrizaje ha sido la carretera que une Quintanar de la Orden con Los Hinojosos, a ambos lados de la carretera, existen muchas pistas forestales y fincas, por lo que, si la sonda cae por esta zona, será fácil llegar a ella.

7.1.2 CUSF LANDING PREDICTOR

Una vez seleccionado el punto de aterrizaje, ya podemos comenzar a hacer uso del programa de predicción, este se llama *CUSF Landing Predictor* y está disponible online en el siguiente enlace:

<http://predict.habhub.org/#!/uuid=ac717ac17f1f99a7d88c0b8e2785ef7a5756565c>

Una vez que accedamos llegaremos a una ventana principal que por defecto despliega un mapa centrado en Reino Unido, lo primero que debemos hacer es desplazar el mapa hasta la zona de Castilla la Mancha.

Cuando estemos situados sobre esta comunidad, ya podremos seleccionar el punto de lanzamiento, para ello en el recuadro inferior derecho debemos seleccionar “*Set With Map*”, justo después pinchamos sobre el mapa y nos aparecerá un símbolo rojo y blanco en el mapa, este símbolo aparece en el punto en el que se ha pinchado y representa el punto desde el que será lanzado el globo.

Una vez seleccionado, también podemos ver como la Latitud y la Longitud cambian a la del punto seleccionado, una opción disponible es la de crear puntos de lanzamiento propios, para ello seleccionamos un punto de lanzamiento y luego pulsamos sobre “*Save Location*”

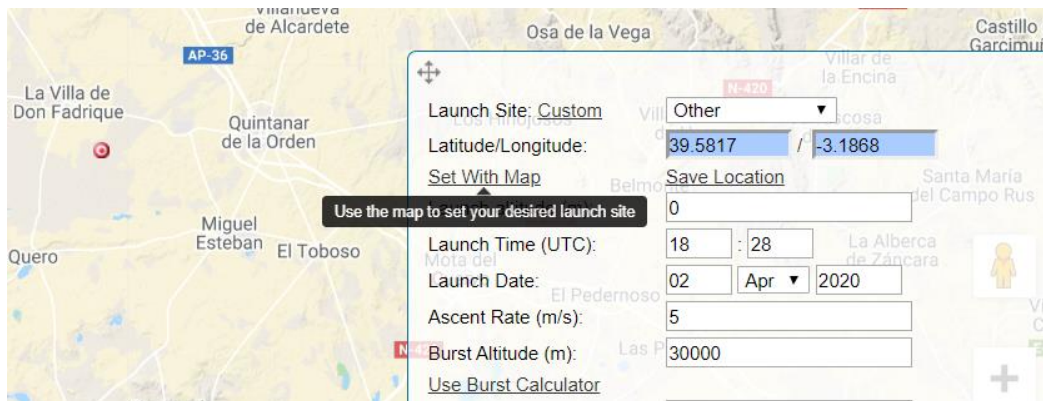


Figura 175 Posición CUSF

Si hemos presionado sobre esta última opción, nos aparecerá esta ventana:

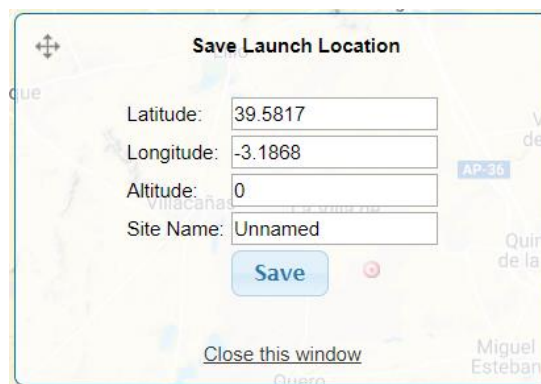


Figura 176 Guardar posición

En ella podemos añadir la altura sobre el nivel del mar del punto seleccionado y posteriormente añadir un nombre significativo para el punto, en este caso lo voy a llamar “*Unnamed*”, para guardarlo, solo queda pulsar sobre el botón “*Save*”.

Si ahora quiero hacer uso de uno de los puntos creados, debo volver a la ventana inferior derecha y pulsar sobre “*Custom*”, una vez pulsado, nos aparece esta ventana:

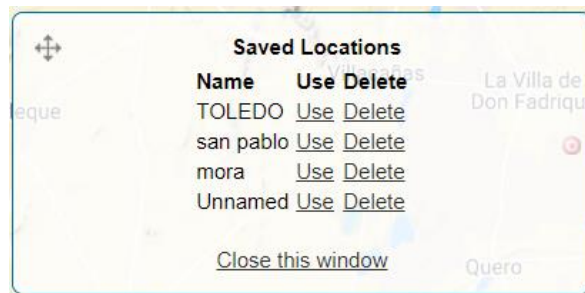


Figura 177 Posiciones guardadas

Para usar uno de los puntos guardados, tan solo debemos pulsar sobre “*Use*” en la opción deseada.

Una vez explicado como seleccionar un lugar de lanzamiento, continuo con la explicación del resto de casillas.

La siguiente casilla por rellenar es la altura sobre el nivel del mar, este dato debe ser consultado en internet y luego se debe introducir. De esta forma, la predicción será muy precisa.

Justo después de la altura, debemos ingresar la hora y la fecha del lanzamiento, es importante recordar que la hora se debe ingresar en UTC y también tener en cuenta que el programa solo realiza predicciones hasta 180 horas, por lo que, si introducimos una fecha más allá de estas horas contando desde la hora en que se realiza la predicción, aparecerá una ventana emergente recordando que no se puede simular tan adelante en el tiempo.

A continuación, debe de rellenarse la velocidad de subida en (m/s) y la altura en metros a la que el globo explotará, si estos datos los conocemos, los podemos introducir a mano, pero también está la opción de pulsar sobre el botón de “*Use Burst Calculator*”, el cual ya hemos utilizado anteriormente.

Lo ideal es acceder de nuevo a este menú y volver a rellenar los datos de nuestro globo, para determinar la velocidad de subida y la altura de explosión.

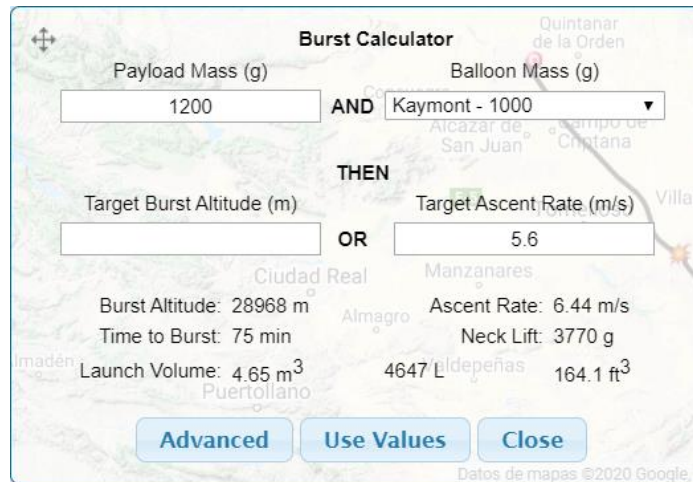


Figura 178 Simulador helio CUSF 2

Una vez calculados los metros cúbicos que vamos a utilizar, ahora solo debemos escribir el peso del globo, introducir el modelo y cambiar la velocidad de ascenso hasta que “Launch Volume” coincida con los metros cúbicos de helio que hemos comprado.

Cuando esté listo, pulsamos sobre “Use Values” y los datos se cargarán automáticamente en la ventana inferior derecha.

Por último, solo falta por introducir los datos de velocidad de bajada, para ello se puede hacer uso de la siguiente calculadora online a través del enlace:

http://randomaerospace.com/Random_Aerospace/Parachutes.html

Parachute Descent Rate Calculator

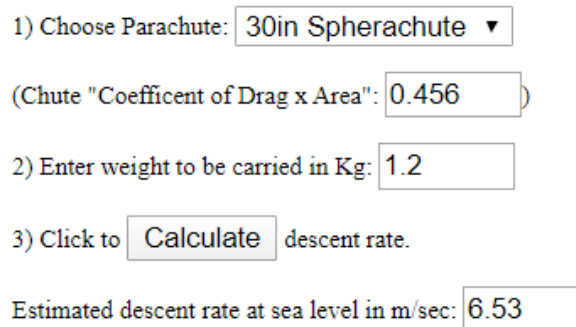
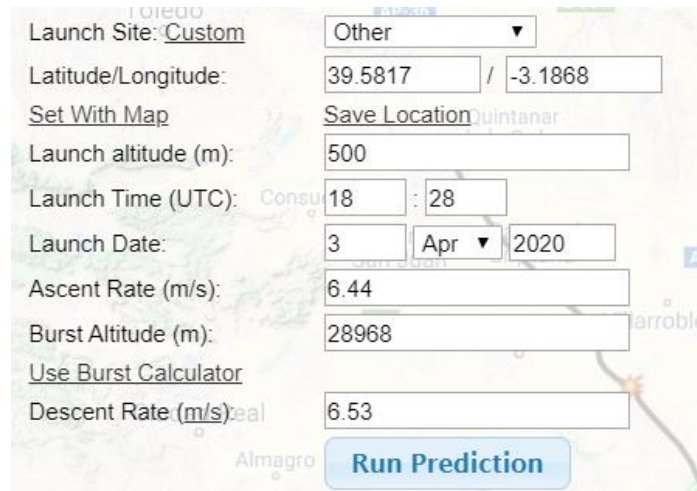


Figura 179 Simulador paracaídas

Para usarla solo debemos elegir nuestro tipo de paracaídas y el peso que llevará colgando.

En este caso el cálculo indica una velocidad de descenso de 6,53 (m/s).

Así que finalmente para este ejemplo, los datos introducidos serían los siguientes:



The screenshot shows a web form for entering launch data. The fields are as follows:

- Launch Site: Custom (dropdown menu)
- Latitude/Longitude: 39.5817 / -3.1868
- Set With Map (link)
- Save Location (link)
- Launch altitude (m): 500
- Launch Time (UTC): 18 : 28
- Launch Date: 3 Apr 2020
- Ascent Rate (m/s): 6.44
- Burst Altitude (m): 28968
- Use Burst Calculator (link)
- Descent Rate (m/s): 6.53
- Run Prediction (button)

Figura 180 Datos CUSF

Para comenzar la simulación, solo hace falta pulsar sobre el botón “Run prediction”, justo después nos aparecerá una simulación similar a esta sobre el mapa.

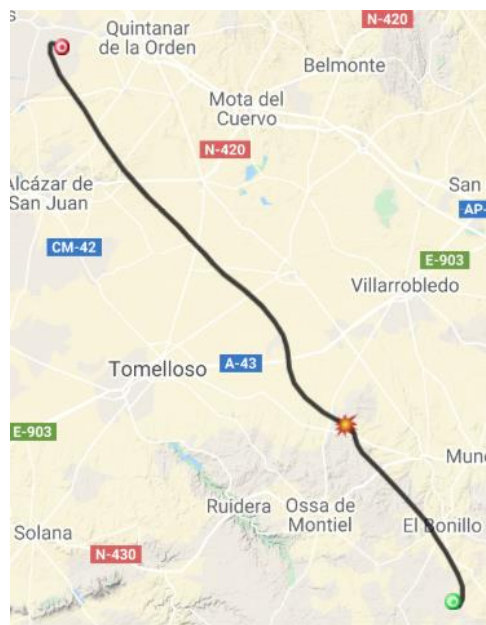


Figura 181 Predicción CUSF

En esta imagen se ve un trazo negro sobre el mapa, este trazo, indica el recorrido del globo, además aparece una figura que indica la explosión del globo y también aparece un icono de color verde y blanco que marca el punto de aterrizaje.

Una curiosidad de estas simulaciones es que, aunque la velocidad de bajada es casi igual que la de subida, en el mapa se puede ver que el trayecto de bajada ha sido mucho más corto, esto es debido a que realmente la velocidad de bajada es la que el paracaídas lleva cuando trabaja en condiciones normales.

Sin embargo, cuando el globo explota, la densidad del aire es tan baja que el paracaídas no realiza su trabajo y durante los primeros miles de metros se produce una caída libre, en la que el paracaídas apenas frena la caída.

A una determinada altura, la densidad empieza a ser mayor y entonces el paracaídas comienza a frenar la cápsula de forma progresiva, hasta que finalmente la sonda alcanza la velocidad que establece el fabricante del paracaídas para condiciones normales.

Todos estos datos, ya se han tenido en cuenta a la hora de realizar la simulación, por esta razón, el trayecto de descenso siempre es más corto que el trayecto de ascenso.

Sin embargo, esta no es toda la información disponible, una vez realizada la simulación podemos ver que en la parte superior izquierda aparece una ventana como esta:



Figura 182 Información predicción

En este cuadro nos da información muy importante, como por ejemplo la distancia del punto de lanzamiento al de aterrizaje, la duración del vuelo y también la distancia del cursor hasta el punto de aterrizaje y de lanzamiento.

Justo debajo de toda esta información, aparecen tres botones, estos son: *Pan To*, CSV y KML.

Si seleccionamos KML, podremos descargar un archivo para abrirlo con Google Earth, este archivo nos muestra el mismo recorrido, pero en 3D.

Este archivo puede ser muy útil para comparar la predicción con los datos reales.

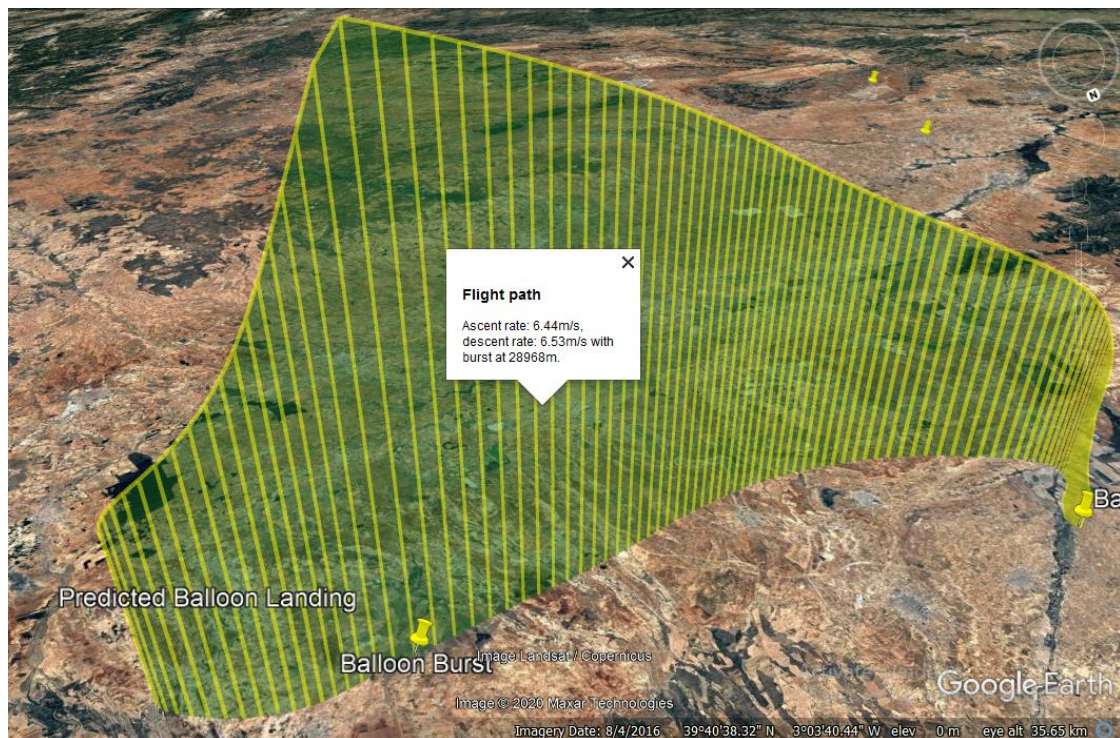


Figura 183 Track 3D

Ahora que ya hemos visto cómo funciona el programa, solo falta buscar un punto de lanzamiento adecuado para que el globo aterrice en la carretera que va de Quintanar de la Orden a Los Hinojosos.

Para ello iremos jugando con el punto de lanzamiento hasta que se ajuste al lugar de aterrizaje.



Figura 184 Elección lugar de lanzamiento

Por ejemplo, en este caso, se ve como el lanzamiento debería producirse al sur de Tembleque para que el aterrizaje sea en la zona deseada.

Una vez que se tenga previsto el lanzamiento, será muy importante realizar varias simulaciones al día, para poder estimar donde será este, de esta forma también podremos atrasar o adelantar el lanzamiento si fuera preciso.

Por último, uno de los puntos más importantes a la hora de realizar el vuelo, es el hecho de respetar la normativa de aviación.

El espacio aéreo español está dividido en varias zonas, algunas de ellas están totalmente restringidas a este tipo de vuelos, en otras se permiten estos vuelos, pero cumpliendo ciertas condiciones.

Así que una vez que tengamos claro el vuelo que va a seguir el globo, será necesario acceder a esta página web, en la que se muestra un mapa con todas las zonas especiales del espacio aéreo español.

La página es: <https://www.icarusrpa.info/>

Si nos centramos en la zona de Castilla la Mancha, veremos las siguientes zonas.

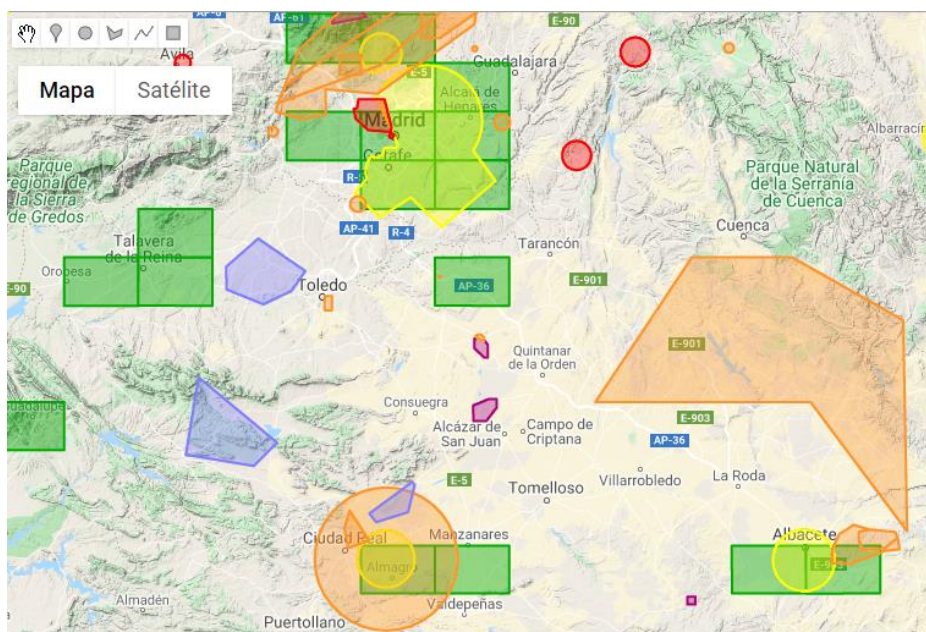


Figura 185 Icarus map

Este mapa interactivo viene acompañado con una leyenda en la que informa sobre qué significado tiene cada color, por ejemplo, las zonas verdes son zonas

restringidas al vuelo fotográfico, las naranjas, son zonas peligrosas, las amarillas CTR, azules, zonas restringidas y las lilas otras zonas protegidas.

Ahora el trabajo consiste en comparar la simulación de vuelo con este mapa, si la simulación pasa por encima o cerca de una de estas zonas, se debe consultar qué condiciones se deben cumplir.

Por ejemplo, cerca de Quintanar de la Orden existe una zona lila y otra naranja, si consultamos estas zonas en el mapa, obtenemos la siguiente información.

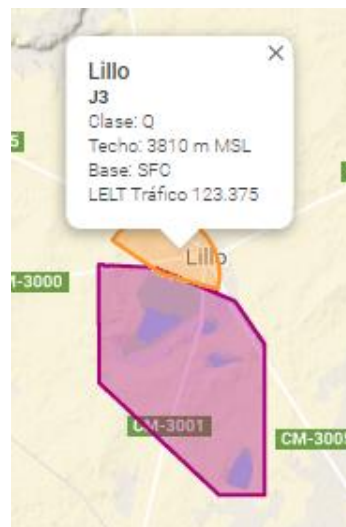


Figura 186 Zona ZEPA

En este caso, la zona naranja corresponde a un aeródromo y se puede volar sobre el siempre y cuando se vuele a más de 3.810 (m), por otro lado, la zona lila está restringida para vuelos de menos de 152 (m).

Así que como el globo pasará a más altura, no existe ningún problema, pero algunas zonas son mucho más restrictivas, en caso de que no estemos seguros de si podremos sobrevolar una zona, deberemos ponernos en contacto con ENAIRE, esto lo veremos en el apartado de Permisos y seguro.

Aunque estamos viendo que existen algunas zonas que podemos sobrevolar siempre que cumplamos las condiciones, existen otras que no podremos sobrevolar bajo ningún concepto, incluso deberemos volar a una cierta distancia de estas.

Un ejemplo de estas zonas son las zonas CTR, en ningún caso se podrá volar sobre estas zonas ni en sus inmediaciones, en caso de tener que pasar por estas zonas,

se deberá primero pedir un permiso a ENAIRE, que también lo veremos en el apartado correspondiente.

Con toda esta información, ya se está en disposición de elegir un lugar de aterrizaje seguro, de realizar una buena simulación y de establecer una ruta de vuelo segura.

7.2 METEOROLOGÍA

En este apartado, se dan unas breves nociones de como determinar si es un buen día para el lanzamiento y también se explican unas nociones básicas sobre la atmosfera terrestre.

7.2.1 ATMÓSFERA TERRESTRE

La atmósfera terrestre será el lugar en el que desarrollaremos el experimento. Para entender mejor la problemática del entorno, vamos a exponer la estructura de la atmosfera atendiendo a tres clasificaciones diferentes, la primera de ellas es en función a la estructura química, la segunda, en función a su estructura térmica y, por último, en función de su estructura eléctrica.

Si realizamos una clasificación de la atmosfera en función de su composición química, podemos diferenciar dos capas principales: la homosfera y la heterosfera.

La homosfera, es una capa donde el aire mantiene prácticamente una composición uniforme, esta capa se extiende hasta los 80 (Km), esta capa está dominada por las violentas corrientes de convección.

Por encima de la homosfera, se encuentra la heterosfera, en la cual el factor dominante es la difusión de los gases, de forma que los más pesados quedan abajo y los más ligeros en las capas más altas.

Esta capa esta a su vez dividida en cuatro capas más pequeñas, de la parte más baja a la más alta, las capas son las siguientes: capa de nitrógeno (N_2), capa de Oxigeno monoatómico (O), capa de helio (He) y capa de hidrógeno (H_2).

En función a la clasificación por su estructura térmica, podemos dividir la atmósfera en cinco capas: troposfera, estratosfera, mesosfera, termosfera y exosfera.

La troposfera, abarca desde los 0 (Km) hasta los 11 (Km), en ella la temperatura desciende de forma aproximadamente lineal con la altura, el límite superior de la troposfera se denomina tropopausa.

La estratosfera abarca desde los 11 (Km) hasta los 50 (Km), esta se divide a su vez en dos capas, la primera de ellas se denomina baja estratosfera y va desde los 11 (Km) hasta los 20 (Km), en esta capa la temperatura se mantiene constante, la otra capa se denomina alta estratosfera y va de los 20 (Km) a los 50 (Km), en esta última capa de la estratosfera, la temperatura aumenta con la altitud, esto es debido a la absorción de radiación ultravioleta en la formación del ozono (O_3).

El límite superior de la estratosfera se denomina estratopausa.

La siguiente capa se denomina mesosfera, comprendida entre los 50 (Km) y los 85 (Km).

En esta capa, la temperatura vuelve a disminuir con la altura, llegando a alcanzar los $-90^{\circ}C$ en el límite superior de la mesosfera (mesopausa).

La siguiente capa es la termosfera, que va desde los 85 (Km) hasta los 750 (Km), en esta capa se pueden llegar a alcanzar temperaturas muy altas, hasta de $1.100^{\circ}C$, sin embargo, estas temperaturas corresponden con el promedio de la energía cinética de las partículas que componen la materia, debido a la baja densidad, estas temperaturas no son comparables a las que

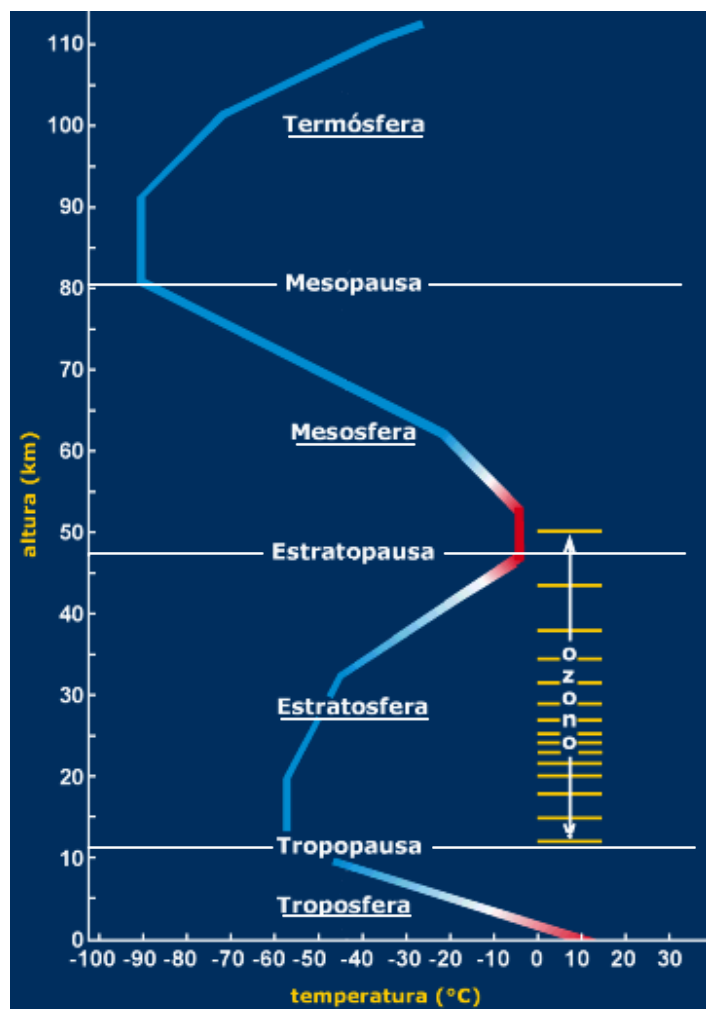


Figura 187 Capas atmósfera

se experimentan en la superficie terrestre.

La última capa es la exosfera, desde los 750 (Km) hasta los 10.000 (Km), esta capa se considera la frontera entre la atmósfera y el espacio exterior.

En esta capa la densidad es mínima, sin embargo, en ella comienzan los procesos de calentamiento por fricción de los cuerpos que entran en la atmósfera.

Por último, si atendemos a la estructura eléctrica de la atmósfera, podemos dividir esta en dos capas claramente diferenciadas.

La capa inferior, en la que la carga de las partículas y moléculas puede considerarse prácticamente neutra.

Por otro lado, se encuentra la capa superior, esta también es conocida como la ionosfera, esta se encuentra entre los 80 (Km) y los 400 (Km).

En la ionosfera, la concentración de carga eléctrica puede llegar a ser muy importante.

Para poner un ejemplo de lo importante que puede llegar a ser esta capa, vamos a hablar de la capa D, esta es una de las 4 capas que componen la ionosfera, esta abarca desde los 50 (Km) hasta los 90 (Km).

En esta capa, las ondas de radio de alta frecuencia no son reflejadas, esta capa es la responsable de la absorción de estas ondas, de 10 (Mhz) e inferiores, este fenómeno aumenta durante el día, dificultando las comunicaciones de radio a larga distancia o incluso pueden desaparecer.

Este factor será fundamental para establecer comunicación en WSPR.

7.2.2 WINDY

Una vez explicadas las capas de la atmósfera, ya estamos en disposición de realizar una observación de la meteorología en superficie y en las capas altas, de esta forma podremos determinar si se puede continuar con la misión o por el contrario se debe cancelar.

Para realizar esta observación, deberemos acudir a dos páginas, la primera de ellas es AEMET, esta web es la oficial del estado y en caso de tener que presentar algún documento a AESA, nos debemos fijar en el parte de AEMET.

Sin embargo, existe una página que para esta aplicación resulta de mucha utilidad, esta página es *Windy*.

Al acceder a esta *web*, podremos ver la siguiente ventana:

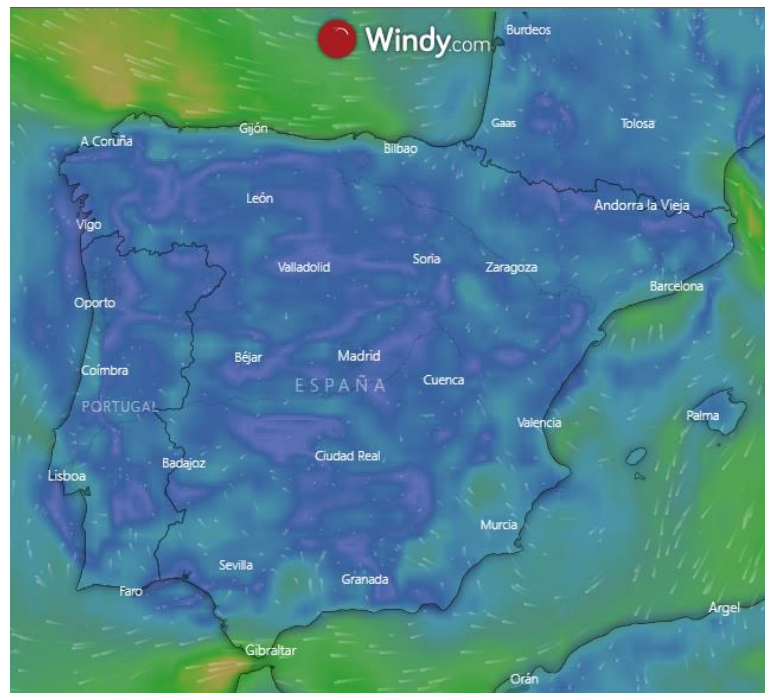


Figura 188 Windy

La imagen es dinámica y muestra con colores la velocidad del viento, además, se ven unas estelas en movimiento para indicar la velocidad del viento.

Para que la imagen cambie a las próximas horas o días, deberemos recurrir a este gráfico:



Figura 189 Selección día

Podemos pulsar sobre un día y una hora, pero también podemos pulsar sobre el botón de “Play” para ver cómo va avanzando la previsión.

Si queremos saber el tiempo en un punto en concreto, deberemos hacer *zoom* en la zona deseada y pulsar sobre ella, de esta forma veremos un marcador sobre el mapa que nos indicará el parámetro que estemos observando.



Figura 190 Puntero de predicción

Si queremos ver otros parámetros, como temperaturas, radar de lluvia o nubes. Deberemos acudir al menú lateral derecho.



Figura 191 Selección de datos

En este menú podemos cambiar el parámetro, por ejemplo, podemos decidir cambiarlo para ver temperaturas.

Por último, una herramienta muy útil es la herramienta de la altura, con ella podremos ver la temperatura y el viento desde la superficie hasta unos 13,5 (Km)

Teniendo en cuenta que la troposfera llega hasta los 11 (Km), esta herramienta nos permite conocer todos los datos de viento y temperatura para esta capa.

Además, con los conocimientos adquiridos en el punto anterior, también se puede determinar la temperatura hasta los 20 (Km), es decir, hasta el límite de la baja estratosfera, ya que la temperatura en esta capa se mantiene constante.

Para seleccionar una altura, solo se debe pasar el cursor sobre la barra, hasta pasar sobre la altura deseada, luego se debe pinchar sobre la barra para que nos muestre los valores.

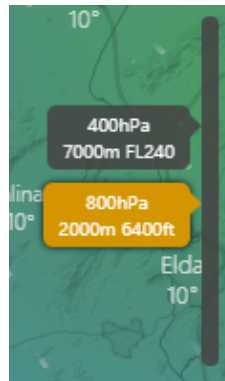


Figura 192 Selección de altura

En caso de que la predicción no esté disponible, o una vez obtenidos los datos de la sonda, puede ser muy útil conocer las fórmulas de la ISA, ya que estas nos permiten conocer la altura en función de la temperatura y de la presión.

7.3 PERMISOS Y SEGURO

A la hora de llevar a cabo un proyecto de este tipo, es necesario realizar un análisis de posibles riesgos.

En este proyecto en concreto, existen tres fuentes potenciales de riesgos, la primera es el uso de botellas de gas comprimido, por otro lado, existe el riesgo de afectar al normal funcionamiento del espacio aéreo y, por último, el riesgo de causar daños durante el vuelo o el aterrizaje.

Para tratar de minimizar los riesgos, así como para cumplir con la normativa y cubrirnos respecto a posibles daños materiales o personales derivados del experimento, hemos seguido tres líneas de trabajo.

La primera línea de trabajo ha sido el estudio de un seguro para aviación que cubra los posibles daños ocasionados por el experimento.

La segunda línea de trabajo consiste en un estudio del vuelo a realizar, estudiar zonas de exclusión al tráfico aéreo y finalmente contactar con la autoridad competente para informar del vuelo y recibir de ellos el visto bueno.

Por último, se ha cumplimentado una documentación para hacer el seguimiento a las botellas de helio desde el día de comienzo del alquiler hasta el día de la devolución.

7.3.1 SEGURO DE AVIACIÓN

Para contratar un seguro acorde a nuestra actividad, es necesario contactar con la correduría de seguros y una vez detallemos la naturaleza de la actividad, la correduría nos pondrá en contacto con la delegación de seguros de aviación.

Estos últimos nos pedirán varios detalles sobre la actividad, a continuación, muestro el formulario a rellenar:

- DESCRIPCIÓN DETALLADA DEL GLOBO, CON PESO, PESO DEL EQUIPO, TIPO DE GLOBO, ALTITUD, DESVIACION ESPERADA, ETC...:

El globo es blanco de unos 1.063 gramos de peso, estará separado de la carga de pago unos 25 metros aproximadamente, la carga de pago es de 925 gramos y esta está unida a un paracaídas de unos 90 gramos teniendo en cuenta el nylon para amarrarlo a la carga de pago y al globo.

La carga de pago va protegida por una carcasa esférica de poliestireno, el peso de esta va incluido en el de la carga de pago.

Según el fabricante, el globo alcanzará hasta unos 32.000 metros de altura.

La desviación esperada en la trayectoria es de unos 15 (Km) que es el margen de error típico para el modelo matemático usado, es decir la desviación se estima que será menor del 10%.

- OBJETIVO DEL LANZAMIENTO:

El objetivo principal es recuperar la carga de pago para así poder recuperar imágenes y los datos almacenados en una tarjeta de memoria.

Como segundo objetivo, probar diferentes equipos de comunicaciones para compararlos y obtener conclusiones sobre el mejor sistema de comunicaciones para incorporarlos en globos sonda.

- DESCRIPCIÓN DETALLADA DE LA OPERACIÓN:

Sobre las 4 de la mañana nos desplazaremos al lugar de lanzamiento, una vez allí prepararemos todo el equipo necesario y realizaremos las últimas comprobaciones para lanzarlo aproximadamente a las 6 y media de la mañana.

El vuelo durará unas dos horas y media según nuestros últimos cálculos, alcanzará los 30.000 metros de altura, el globo estallará y comenzará el descenso controlado mediante un paracaídas.

Durante el vuelo nosotros triangularemos la posición de la sonda para poder comprobar hacia dónde va y si se cumple la predicción, de esa forma podremos estar en el lugar del aterrizaje prácticamente cuando este llegue.

Cuando lleguemos al lugar de aterrizaje, recuperaremos la carga de pago, así como cualquier basura que se pueda generar por el impacto.

-CONFIRMACIÓN DE QUE EL DESCENSO ES 100 POR 100 CON PARACAIDAS:

Si, el 100% del descenso será con un paracaídas de unos 76 (cm) de diámetro unido firmemente a la carga de pago mediante nylon.

Para una carga de pago entre 800 y 1.300 gramos, el fabricante estima una velocidad de caída de entre 5 y 7 metros por segundo.

- ZONA DE LANZAMIENTO:

La zona dependerá de las condiciones meteorológicas, pero el plan consiste en lanzarlo en Castilla-La Mancha.

La idea es que caiga en una zona comprendida entre Quintanar de la Orden y Los Hinojosos. Es una zona de campo y alejada de ejes de comunicación, con una densidad de población muy baja lo cual la convierte en una zona ideal para el aterrizaje.

Teniendo en cuenta el lugar de aterrizaje, el modelo matemático de simulación y los vientos predominantes, los dos posibles lugares de lanzamiento son las provincias de Ciudad Real y Toledo.

- NUMERO DE LANZAMIENTOS ESTIMADOS AL AÑO:

1 lanzamiento.

- CONFIRMACIÓN DE AUTORIZACIONES:

La autorización para ENAIRE ha sido enviada.

- SINIESTRALIDAD HISTORICA:

Ninguna, puesto que es el primer vuelo realizado.

Una vez enviados los datos, recibiremos una primera póliza y un presupuesto.

En este caso, la póliza califica a la actividad como globo/dirigible, y establece una cobertura máxima a terceros de 910.000 €

Una vez lista la póliza, se realiza al pago y durante el día del vuelo, nuestra actividad estará cubierta.

7.3.2 PERMISO ENAIRE

Como segunda línea de trabajo, nos hemos puesto en contacto con AESA, en concreto con el departamento de ENAIRE, el cual se dedica al control del espacio aéreo.

En concreto, nos ponemos en contacto con ellos para rellenar el documento estandarizado de permiso de actividad aérea, de esta forma podremos ver que normativa tenemos que seguir.

Una vez cumplimentado, podemos comprobar que nuestro globo es considerado como globo libre no tripulado ligero.

Teniendo este dato en cuenta, según el [RD 1180/2018 de 21 de septiembre](#), para un globo libre no tripulado ligero, no es necesaria ninguna coordinación si la actividad se realiza fuera de las servidumbres de un aeródromo, a más de 8 (Km) de su ARP (Punto de Referencia del Aeródromo) y fuera de su *Controlled Traffic Region* (CTR).

En concreto, debemos atenernos a el capítulo 3, artículo 11 y 12.

Artículo 11. *Lanzamiento en el entorno de las infraestructuras aeronáuticas.*

El lanzamiento de globos libres no tripulados y la suelta de farolillos voladores o globos de helio deberá realizarse a una distancia mínima de 8 km del punto

de referencia del aeródromo publicado en la Publicación de Información Aeronáutica (AIP) y fuera de la zona de servidumbres aeronáuticas, salvo coordinación expresa con el gestor o responsable de la infraestructura aeronáutica y, cuando proceda, sujeta a la aplicación de los procedimientos de identificación y gestión de riesgos por la realización de actividades humanas o usos del suelo en el entorno aeroportuario.

A estos efectos, quien pretenda el lanzamiento o suelta en estas zonas deberá comunicarlo al gestor o responsable de la infraestructura, antes de la realización de la actividad.

Artículo 12. Coordinación operativa del proveedor de servicios de tránsito aéreo.

1. Sin perjuicio de lo previsto en los artículos 10 y 11, estará sujeto a la coordinación operativa por parte de la entidad pública empresarial ENAIRE:

- a) El lanzamiento de globos medianos o pesados.**
- b) El lanzamiento de globos ligeros en una zona de control de tránsito aéreo (CTR), así como en zona ATZ o FIZ asociada a un aeropuerto.**
- c) La suelta de más de 500 globos de helio cuando se efectúe dentro de un radio de 8 km del punto de referencia de cualquier aeródromo publicado en la AIP.
- d) La suelta de más de 1.000 globos de helio cuando se efectúe dentro de un radio de entre 8 y 16 km de distancia del punto de referencia de cualquier aeródromo publicado en la AIP.
- e) La suelta de farolillos voladores cuando se efectúe dentro de un radio de 16 km del punto de referencia de cualquier aeródromo publicado en la AIP.

A estos efectos y con sujeción a los criterios y procedimientos operativos generales establecidos por la Comisión Interministerial entre Defensa y Fomento (en adelante CIDEFO), ENAIRE establecerá los formularios que deben cumplimentar los responsables del lanzamiento, así como los procedimientos de coordinación y, aquéllos que tengan por objeto establecer las condiciones que eviten peligros para las aeronaves que operen conforme a las reglas de la circulación aérea general u operativa.

CTR, son las zonas marcadas de amarillo en *Icarus map*, mientras que un aeródromo, es aquel en el que existe un tráfico regular de aeronaves.

En conclusión, a más de 8 (Km) de un aeródromo y siempre fuera de su CTR.

En caso de no cumplir con esto, deberemos mandar un documento solicitando coordinación de la actividad, en nuestro caso no es necesario, aun así, se envió toda la documentación para que estuvieran en aviso.

El documento lo adjunto en el apartado de anexos, este puede ser utilizado para todos los vuelos de este tipo.

7.3.3 PERMISO HELIO

A la hora de transportar una botella de gas comprimido, tenemos que seguir un procedimiento para evitar accidentes, para ello, nos pondremos en contacto con el proveedor de gas y él nos facilitará los pasos a seguir, uno de los más importantes, es que las botellas deben ir en posición vertical y firmemente trincadas al vehículo, de forma que, si se produce un accidente, la botella no salga disparada.

Otra recomendación es comprobar que la válvula quede completamente cerrada, de esta forma se evitaría una fuga de gas que podría dejar al conductor inoperativo.

Además, el proveedor deberá facilitar un documento para realizar el seguimiento de la botella, en este documento se especificará el gas, los metros cúbicos y en caso de que se transporte, aparecerá el destino, donde se compró la botella y donde se tiene que devolver.

Este documento debe ser presentado a las autoridades en caso de que sea requerido, por lo que este documento acompañara a las botellas durante toda la actividad.

De esta forma, cumpliremos con la legalidad y evitaremos ser sancionados por la autoridad.

7.4 LOGÍSTICA

Una vez terminado todo el trabajo, solo queda desplazarse al punto de lanzamiento y cerciorarse de que todo el material esté listo y operativo.

Para que nada falle, se describen los pasos a seguir:

7.4.1 LISTAS DE COMPROBACION

Aunque durante muchos meses se ha realizado un arduo trabajo, todo este podría fracasar por algo tan simple como olvidar las baterías, la cinta americana, etc...

También podría darse una reparación de última hora, o incluso una comprobación del buen funcionamiento de un programa del autómata, por todo ello, antes de partir, se debe realizar una lista de comprobación exhaustiva.

Esta debe tener dos secciones, la primera de ellas debe contener una Comprobación de operatividad de Sistemas (COS), de esta forma, todos los equipos serán comprobados desde su funcionamiento más elemental hasta el más pequeño detalle, también se tiene que incluir un listado de todo el material necesario para llevar al lanzamiento, como herramientas, consumibles, etc...

La segunda parte del documento debe incluir una comprobación de todos los equipos, pero esta debe ser algo menos detallada, puesto que anteriormente se han comprobado y han funcionado, además, las comprobaciones deben realizarse en un estricto orden.

La idea es que una vez que termina esta comprobación, la sonda se va a precintar con cinta americana y va a ser lanzada, por ello el orden comentado anteriormente atiende a factores como la batería de los equipos, memoria disponible etc...

Es decir, primero se comprobarán aquellos equipos en que la batería no sea un factor muy crítico, por ejemplo, los autómatas, dejando para lo ultimo las cámaras, ya que tienen la batería justa.

Si durante la comprobación algo no funciona correctamente, se deberá repetir hasta que todo cuadre.

Una vez expuesta la COS, debe aparecer un apartado de pasos a seguir durante el vuelo, así como unas pautas para su recuperación, de forma que, aunque el grupo se divida, todos sepan cómo proceder.

Toda esta información debe ser compartida con los participantes de forma telemática y se debe realizar una copia en papel para que el jefe del proyecto pueda tomar datos.

En resumen, la lista de comprobación deberá realizarse con bastante anterioridad al lanzamiento, de forma que la primera parte de la lista se realice una o dos semanas antes de lanzamiento, de esta forma se puede tener tiempo para solucionar problemas de última hora.

Finalmente, la segunda parte se realizará justo antes del lanzamiento, durante el vuelo y en la recuperación.

En el apartado de anexos, adjunto la lista de comprobaciones de este proyecto.

7.4.2 COMUNICACIÓN

Para que la recuperación de la sonda sea un éxito, es necesaria una comunicación entre todos los integrantes del grupo, sin embargo, existen factores que pueden incomunicar a miembros del grupo o incluso al grupo por completo.

Por esta razón, el equipo humano queda dividido en dos secciones, una de ellas será la estación de control fija y la otra sección será la estación de control móvil.

La estación de control móvil está compuesta por el equipo humano que acude al lanzamiento, este está a su vez dividido en varios vehículos, el objetivo es recopilar toda la información del globo sonda para llegar al punto de aterrizaje a los pocos minutos de que la sonda toque tierra.

Para comunicar los diferentes vehículos, se utiliza como medio mensajería instantánea, telefonía móvil y equipos de radio portátil.

Al ser un grupo que se mueve, puede que se encuentre en zonas sin cobertura, esto dejaría al grupo incomunicado con la sonda, pero estarían comunicados a través de la radio.

Para evitar perder el contacto con la sonda y con la intención de poder seguir el rastro de forma eficiente, se ha creado la estación de control fija.

La estación de control fija está compuesta por individuos que no se encuentran en el campo, por el contrario, se encuentran en un lugar con conexión continua a internet, con cobertura móvil y con fuente de energía disponible.

Estos se dedican a realizar un trabajo consistente en la recopilación de toda la información que llegue de la sonda, en caso de que el equipo móvil pierda conexión con la sonda, una vez que recupere la cobertura, se deberá poner en contacto con la estación fija, la cual le indicará la posición de la sonda.

De esta forma no se perderá información ni tiempo a la hora de encontrar la sonda.

La otra utilidad de la estación fija consiste en que, si fallan los equipos de SigFox y WSPR, y el equipo móvil va a entrar en zona sin cobertura, entonces se puede delegar a la estación fija el trabajo de intentar posicionar la sonda mediante GPRS, hasta que la estación móvil vuelva a entrar en zona de cobertura.

7.4.3 TRANSPORTE

Finalmente, el lugar seleccionado para el lanzamiento fue Orgaz, el punto fue decidido la misma mañana del lanzamiento, aun así, teníamos claro que el lanzamiento sería en la provincia de Toledo, por ello decidimos utilizar coches para el desplazamiento.

Como mínimo es recomendable llevar dos coches, de esta forma si alguno de los coches sufre un pinchazo o una avería menor, el otro podría seguir buscando la sonda.

En este caso, debido a la cantidad de gente que asistió al lanzamiento, usamos dos coches y una furgoneta, la furgoneta fue sobre todo utilizada para el transporte del material, sobre todo de las botellas de helio.

A la hora de seleccionar el medio de transporte, también es muy importante realizar una ruta para llegar al lugar de lanzamiento, así como otra para llegar al punto de recogida y, por último, del punto de recogida al punto de inicio.

De esta forma se puede estudiar la ruta más aconsejable, en función al precio, tráfico y velocidad.

La recomendación es utilizar Google maps y planificar la ruta, también deberemos descargar mapas sin conexión, de forma que, si no tenemos internet, podamos llegar a nuestro destino.

Para descargar los mapas sin conexión, deberemos acceder a Google maps en nuestro teléfono, luego accedemos a nuestro perfil y seleccionar la opción de “offline maps”



Figura 193 Icono offline maps

Luego deberemos seleccionar “Select your own map”.

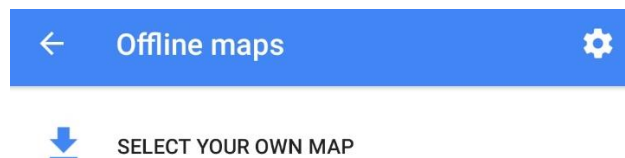


Figura 194 Selección de mapa

Ahora aparecerá un mapa con un rectángulo azul en el centro, utilizaremos este para marcar la zona del mapa que deseamos descargar.

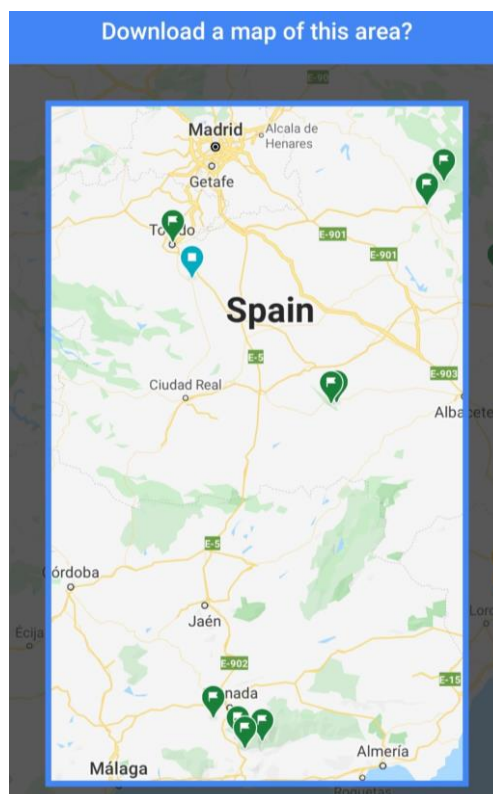


Figura 195 Mapa seleccionado

Una vez descargado, ya podremos hacer uso de el sin conexión.

Finalmente, se deben realizar las rutas para determinar los kilómetros a recorrer y el tiempo estimado de viaje.

Se estiman 3 horas y 4 min de Cartagena a Orgaz, un total de 316 (Km).

1 hora y 21 min de Orgaz a Los Hinojosos, un total de 103 (Km).

3 horas y 55 min de Los Hinojosos a Cartagena, un total de 422 (Km).

En total 8 horas 20 minutos de conducción y un total de 841 (Km).

7.4.4 PLANNING

Una vez determinado el lugar de lanzamiento, el *planning* consiste en salir de Cartagena a las 04:00 de la mañana el día de lanzamiento, parar en Albacete a las dos horas para desayunar y reagrupar a todos los coches.

Una vez terminado el descanso, a las 06:30, llenaremos los tanques de gasolina para evitar tener que repostar durante la recuperación.

Después de repostar saldremos hacia Orgaz, llegaremos allí a las 08:40.

Cuando lleguemos al pueblo debemos localizar un lugar apropiado para el lanzamiento, debe ser un lugar llano, lejos de árboles, edificios, antenas, cables, etc...

Finalmente nos establecemos en el lugar de lanzamiento a las 09:00.

Una vez allí, procedemos a dar una pequeña charla para dejar claro el protocolo de actuación, después comenzaremos a preparar el material, de forma que todo el material estará listo a las 09:45.

Desde ese momento comenzará el llenado del globo y la preparación de la aviónica, la capsula y el paracaídas. De forma que el lanzamiento se realice a las 11:00.

Una vez lanzado, un primer coche se dirigirá hacia Los Hinojosos, mientras los otros dos coches se quedarán recogiendo todos los materiales.

A las 11:15 los dos últimos coches salen de Orgaz hacia Los Hinojosos, de forma que el primer coche llegará a las 12:20 y los otros dos coches llegarán a las 12:35.

El vuelo está previsto que dure 1 hora 50 minutos, de forma que tocará tierra a las 12:50, a esa hora, los coches ya estarán alrededor de la zona estimada.

A las 12:50, se confirmará la última posición obtenida de la sonda y nos dirigiremos hacia ella.

Teniendo en cuenta un error de 10 (km) en la predicción, estimamos unos 15 minutos para llegar al punto más cercano posible en coche, de forma que a las 13:05 estemos bajando del coche para realizar la aproximación a la sonda a pie.

Entre las 13:00 y las 15:00 se estima el tiempo posible para la recuperación.

Si todo va bien a las 15:00 se comerá en Los Hinojosos y a las 15:30 se saldrá hacia Albacete.

A las 17:30 se llega a Albacete, se para para descansar y repostar.

A las 18:00 salimos hacia Cartagena, para llegar sobre las 20:00 al punto de inicio.

7.5 FINANCIACIÓN

Uno de los puntos más importantes a la hora de acometer este proyecto, ha sido la financiación, algunos de los materiales pueden ser caros y aunque el ministerio de educación ha ayudado a materializar el proyecto, es cierto que algunos gastos se han tenido que cubrir mediante aportaciones privadas.

Para conseguir esta financiación, ha sido necesario llevar a cabo una campaña intensiva de 40 días, en los que se ha intentado promocionar el proyecto por todos los medios disponibles, ya sea a través de redes sociales, televisión, radio, blogs, boca a boca o mediante carteles.

Una vez terminada la campaña de publicidad, se siguió otra pequeña campaña para mostrar los datos obtenidos, así como para compensar a los colaboradores.

A continuación, mostraremos cada una de las fases de la campaña de publicidad y finalmente como se realizó la campaña de financiación y que resultados han sido obtenidos.

7.5.1 PUBLICIDAD

Para llevar a cabo una publicidad eficaz, debemos intentar que la publicidad llegue a un público afín a este tipo de actividades o un público interesado con la ciencia en general, además la intención es crear un cierto gancho para que la gente se interese por el proyecto, por ello es necesario crear una imagen corporativa y contenidos que muestren el trabajo y los progresos.

El primer paso para crear una imagen corporativa es decidir un nombre y un logo para el proyecto.

En este caso el nombre es Mantenimiento Electrónico Satélite (METSAT-1) y el logo utilizado ha sido el siguiente:



Figura 196 Logo METSAT-1

Con estos dos sencillos pasos, ya el proyecto obtiene una imagen mucho más seria y profesional, esta será nuestra carta de presentación y será lo primero que el público verá del proyecto, una imagen y un nombre, ya ayudan a crear un público más receptivo.

Sin embargo, este es solo el primero de muchos pasos que seguir para poder atraer a posibles colaboradores, otro de los pasos es crear una pequeña carta de motivación. Una presentación de un minuto en la que se motive al público y se explique a grandes rasgos el proyecto, la idea es que no abarque más de un minuto, de forma que a la hora de exponerlo oralmente seamos capaces de atraer la atención del oyente y no se despiste, una presentación de más de un minuto puede no ser tan eficaz y podría aburrir al público.

Una vez que se ha creado una carta de presentación, es necesario crear un video, en el que se vea el trabajo realizado y que el público entienda la dificultad del proyecto y se motive sabiendo que, gracias a su ayuda, este proyecto saldrá adelante.

Todos estos puntos serán mostrados en el apartado de *Crowdfunding*.

Una vez que tenemos el mensaje listo, es hora de darlo a conocer, si es posible, es importante darlo a conocer en círculos afines, de esta forma lo más posible es que este mensaje sea compartido por estos y nos ayuden a llegar a más gente.

Luego, se debe intentar llegar al máximo número de personas a través de todos los medios que estén a nuestro alcance.

7.5.1.1 REDES SOCIALES

Hoy en día la mayoría de nosotros utiliza redes sociales a diario, esto hace que llegar a mucha gente sea bastante fácil, lo difícil consiste en atraer la atención del público, por ello el paso más importante consiste en que el mensaje empiece a ser difundido por contactos que sabemos que se van a implicar en el proyecto y por lo tanto van a compartir nuestro mensaje con sus contactos.

Una vez que ya se han utilizado los contactos comentados anteriormente, debemos recurrir al resto, para ello, es necesario compartir nuestro mensaje de forma que sea accesible a todos, pero de forma que no esté dirigido a nadie en particular, es decir, si por ejemplo nos encontramos compartiendo información en WhatsApp, lo que tenemos que hacer es compartir nuestro mensaje como una historia y no como un mensaje directo a alguien en particular.

Una vez que el mensaje se ha dirigido a todos, se puede seguir realizando el mismo proceso y añadir nuevo contenido, cuando pase un tiempo prudencial, podemos destinar nuestro mensaje de forma personalizada el resto de los contactos es un proceso largo y costoso, pero es mucho más probable que el receptor se implique con el proyecto, ya sea colaborando o compartiendo nuestro contenido.

Finalmente, ya solo nos queda seguir creando contenido para mantener informados a los interesados en el programa y mantener la atención y el interés de estos.

Algunas redes sociales como Facebook o Instagram poseen una herramienta para conocer el número de visualizaciones de los videos subidos, esto nos ayuda a conocer a cuantas personas ha llegado nuestro mensaje, en caso de que no usemos estas redes sociales, podemos usar YouTube, en esta página podemos

alojar el contenido de video y podremos compartirlo nosotros o cualquiera en otras redes sociales, de esta manera, también podremos determinar a cuantas personas llega nuestra publicidad.

7.5.1.2 MEDIOS DE COMUNICACIÓN

Otra forma de llevar nuestro proyecto a un elevado número de personas es a través de los medios de comunicación tradicionales, como son la televisión y la radio.

Para ello, nos hemos puesto en contacto con periódicos digitales de la zona de Murcia y Cartagena.

En concreto el periódico Gaceta Cartagonova: <https://www.gacetacartagonova.com/el-politecnico-a-mas-de-30km-de-altura-en-la-metsat-1/>

CARTAGENA DESTACADOS PROMO

El Politécnico a más de 30km de altura en la METSAT-1

Seguir a @gacetafm 3.665 seguidores 30 enero, 2020 / Redaccion / No Comments

Llevar material científico a mas de 30km de Altura – COLABORA

«Llevando a cabo un proyecto para llevar material electrónico a la estratosfera y posteriormente recuperarlo, es un proyecto muy interesante. La Misión «METSAT-1» es un proyecto del IES Politécnico de Cartagena en el que se lanzará un globo con una carga de sensores, q temperaturas extremas, una caída libre de 30.000 metros y recuperarlo.

Se pondrá en vuelo un Globo de helio de unos 3 metros de diámetro, este llevará colgando un paracaídas y una capsula con todos una cámara de acción. Una vez que se suelte el globo, este iniciará una ascensión de aproximadamente 30.000 metros de altura, vez en la altura límite, el globo alcanzará unos 8 metros de diámetro, estallará y el paracaídas se desplegará para realizar la reentr imágenes.



Figura 197 QR:
Artículo Gaceta
Cartagonova

Figura 198 Artículo Gaceta Cartagonova

Cartagena de hoy: <https://www.cartagenadehoy.com/index.php/2-uncategorised/22445-con-la-atmosfera-como-mesa-de-trabajo>

Con la atmósfera como mesa de trabajo

Escrito por CTH10, 5 de marzo de 2020, jueves



Alumnos del último curso del grado superior de Mantenimiento Electrónico del IES Politécnico de Cartagena han



Figura 199 QR:
Artículo Cartagena
de Hoy

Figura 200 Artículo Cartagena de Hoy

Página web IES Politécnico de Cartagena: <http://politecnicocartagena.es/>

Proyecto METSAT-1: CFGS Mantenimiento Electrónico



El 28 de febrero de 2020 un equipo formado por profesores y alumnos de 1º y 2º curso del CFGS de Mantenimiento Electrónico se trasladaron a Orgaz, en la provincia de Toledo, para realizar una actividad denominada METSAT-1 consistente ... [Leer más →](#)

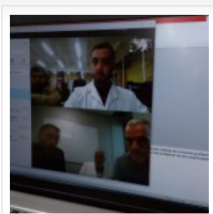
Publicado el 3 marzo, 2020



Figura 201 QR:
METSAT-1 IES
Politécnico

Figura 202 Artículo lanzamiento METSAT

Videoconferencia del IES Politécnico con el grupo InZpace



El viernes 10 de enero de 2020, los alumnos del CFGS de Mantenimiento Electrónico, realizaron una videoconferencia con los miembros del grupo InZpace para recibir Know-How referente al lanzamiento de globos estratosféricos. Al otro lado de Skype estuvieron Lorenzo Tarabini, ... [Leer más →](#)

Publicado el 15 enero, 2020



Figura 203 QR:
Videoconferencia
InZpace

Figura 204 Artículo videoconferencia InZpace

Gracias a esta publicidad en los medios de comunicación fuimos capaces de llegar a mucha más gente, pero lo más importante es que dio una gran imagen de seriedad y de profesionalidad al proyecto.

También nos hemos puesto en contacto con televisiones autonómicas de Murcia y Castilla la Mancha.

El día del lanzamiento, realizamos un directo con la televisión autonómica de Castilla la Mancha, el mismo día, nos reservaron un espacio en el informativo del medio día.

Nos pusimos en contacto con ellos la misma mañana del lanzamiento y nos acompañaron durante todo el proceso de lanzamiento, durante el cual realizaron varias entrevistas a los diferentes participantes del proyecto.



Figura 205 QR:
Entrevista CMM

Figura 206 Directo Castilla la Mancha TV

Una vez realizado el lanzamiento, también nos pusimos en contacto con la televisión autonómica de la Región de Murcia para realizar una entrevista sobre detalles técnicos del lanzamiento y los resultados.



Figura 207 QR:
Entrevista 7RM

Figura 208 Entrevista 7RM

Toda esta publicidad ha resultado clave para difundir nuestro trabajo y dar a conocer nuestro centro como un referente en proyectos de investigación y desarrollo.

7.5.1.3 BOCA A BOCA

Aunque la publicidad en redes sociales y a través de la televisión llega a muchísimo público, muchas veces es necesario un trato más cercano y personal con los posibles colaboradores, por ello es fundamental realizar un estudio de posibles empresas relacionadas o con interés en este tipo de proyectos, o simplemente contactar con empresas que necesitan publicidad.

Es importante tener en cuenta que la imagen de una empresa en un escenario como la estratosfera, es una imagen que no se puede conseguir todos los días.

El trabajo consiste en hacer valer nuestra aportación a la publicidad de las empresas, ya que ofrecemos un lugar privilegiado para publicitarse y llamar la atención.

Por este motivo, es fundamental mantener una relación presencial con las posibles empresas interesadas en participar en el proyecto, para ello necesitamos una buena carta de presentación, una buena motivación que funcione como gancho y una oferta de publicidad que sea de interés para las empresas.

7.5.2 CROWDFUNDING

Una vez empezada la campaña de publicidad, se puede empezar a diseñar la campaña de financiación.

En muchas ocasiones, la financiación se da a través de la venta de papeletas, con la compra de estas, se ayuda a financiar los proyectos y finalmente los compradores de las papeletas participan en un sorteo, donde algunos de ellos pueden obtener una cierta recompensa.

Sin embargo, en la actualidad existen otros métodos para conseguir financiación, métodos en los que además cada uno de los participantes recibe una recompensa, de esta forma, la probabilidad de que alguien quiera participar aumenta, puesto que recibe la recompensa de ayudar en el proyecto y otra recompensa que obtiene en función de la aportación que ha realizado.

Este sistema de financiación se denomina *Crowdfunding*, en la red existen una gran cantidad de páginas web que sirven de soporte a campañas de Micro financiación, nosotros hemos realizado nuestra campaña en Verkami.

Verkami es una compañía de Micro financiación española, es propietaria de una web que funciona como una red social, en la que artistas, músicos, escritores, científicos, amantes de la tecnología, entre otros. Publican cuáles son sus futuros proyectos, muestran cómo va a ser y cuánto dinero necesitan para llevar a cabo su desarrollo, en este punto es cuando entran en juego los “Mecenas”, estos son el grupo mayoritario en la red social.

Los mecenas, acceden a la *web* en busca de proyectos interesantes y motivadores, si el mecenas encuentra un proyecto que le interesa, puede pulsar sobre él y leer toda la información sobre el autor y sobre los detalles del proyecto, el punto clave es que el autor del proyecto propone recompensas a los mecenas si el proyecto sale adelante.

Por ejemplo, si soy el autor de un libro y quiero publicarlo, pero necesito una cierta cantidad de dinero, puedo establecer aportaciones de diferentes cantidades de dinero.

En función a la cantidad aportada, oferto una recompensa, por ejemplo, si se aporta el equivalente a la mitad del libro, puedo ofrecer una copia del libro en PDF, si el mecenas realiza una aportación equivalente al precio del libro, puedo ofrecer una copia en formato papel del libro y si por ejemplo el mecenas ofrece una cantidad superior al precio del libro, el autor puede ofrecer el libro en formato papel, con una dedicatoria personalizada.

De esta forma, el autor del libro puede ir consiguiendo el dinero necesario para el proyecto y los mecenas pueden decidirse a fomentar tu proyecto con la intención de recibir su recompensa.

La clave de que este sistema sea tan eficaz consiste en que, si el autor no es capaz de cumplir su meta de financiación, entonces el autor no recibe ningún dinero y a los mecenas se les devuelve el dinero.

Por esto el sistema funciona como un juego, en el que los mecenas que han pagado se sienten sin miedo a perder el dinero, pero a su vez están motivados a que el

proyecto salga adelante porque se sienten parte de un grupo y porque van a recibir una recompensa. De esta forma los mecenas animan a otros a convertirse en mecenas, para así poder obtener sus recompensas y a su vez sentirse parte de un grupo y de un proyecto.

Este es el funcionamiento de Verkami, aplican esta psicología a sus campañas, campañas de riesgo, de corta duración, con recompensas para todos y una gran cantidad de publicidad para atraer a más mecenas y mantener en una incertidumbre constante a los mecenas que han aportado para que animen a otros.

Si un proyecto sale adelante y cumple su objetivo de financiación, Verkami cobra una parte de este dinero, pero si el proyecto no sale adelante, no cobra, por ello la misma plataforma es la primera interesada en que tu proyecto funcione y dan un gran apoyo de *marketing* y de asesoramiento.

Para crear la campaña es necesario rellenar una plantilla con datos referentes al proyecto, esta plantilla es enviada a Verkami y allí se comprueba si el proyecto encaja con la filosofía de la página *web*.

Si el proyecto es aceptado, deberemos ampliar la plantilla añadiendo quienes somos, que materiales necesitamos, videos de motivación, agradecimientos, entre otros.

Uno de los puntos más importantes a rellenar serán las aportaciones y sus recompensas, en este caso, se han establecido aportaciones de: 1, 10, 20, 35, 50 y 100 €,

Las aportaciones de 1, 10 y 35 euros son ilimitadas, de manera que se pueden realizar todas las que se necesiten, sin embargo, las aportaciones de 50 y 100 euros están limitadas.

A continuación, explicaremos las recompensas y así podremos entender por qué unas son ilimitadas y otras no.

Las aportaciones de 1€ dan derecho a un diploma de agradecimiento por participar en el proyecto y conseguir que se pueda realizar.

Las aportaciones de 10€, recompensan al mecenas añadiendo su nombre en un listado en este mismo trabajo, de forma que se reconozca de forma publica su aportación.

Las aportaciones de 20€, dan derecho a una mención en ese proyecto y en los créditos de los videos oficiales que se obtengan del proyecto.

Las aportaciones de 35€, son las primeras enfocadas a empresas, están dan derecho a un espacio publicitario en el video y a una mención en este documento.

Las aportaciones de 50€, están limitadas a 6 mecenas, esta aportación da derecho a una foto del logo de la empresa pegado en la sonda, un espacio publicitario en el video y una mención en este proyecto.

Respecto a las aportaciones de 100€, existen dos variantes, la primera de ellas, está limitada a un solo mecenas y da derecho a enviar un objeto de menos de 50 (g) en la sonda.

La segunda variante, está limitada a dos mecenas, los cuales serán los patrocinadores oficiales del proyecto, estos tendrán un espacio publicitario privilegiado en los videos, mención especial en este informe y un espacio especial en la página web de Verkami.

Estas últimas tres aportaciones, son limitadas por razones lógicas, el resto sin embargo no suponen ningún problema.

Antes de publicar la campaña, es necesario establecer un objetivo inicial, este será la meta que debe ser alcanzada para poder cobrar el dinero, en un principio la meta se estableció en 500€.

Una vez comenzada la campaña, toda la publicidad se multiplicó, de forma que, en menos de 18 días, alcanzamos el objetivo inicial y fuimos cambiando de forma progresiva la meta, aumentando esta de cien en cien.

Finalmente, a los 29 días, alcanzamos nuestro último objetivo y conseguimos alcanzar los 956€.

Para terminar la campaña todavía quedaban 11 días, pero decidimos cerrarla ya para poder cobrar el dinero antes del lanzamiento.

A continuación, expongo las estadísticas obtenidas en la campaña:

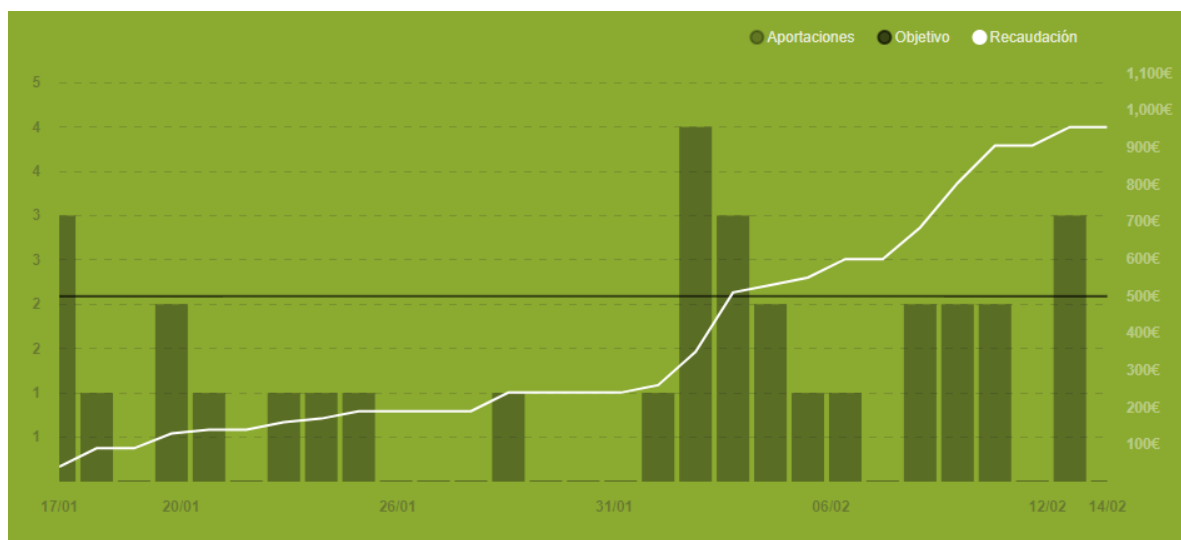


Figura 209 Evolución de ingresos

En este gráfico se muestra la evolución de los ingresos a lo largo de la campaña, además se muestra el objetivo inicial y la cantidad de aportaciones diarias.

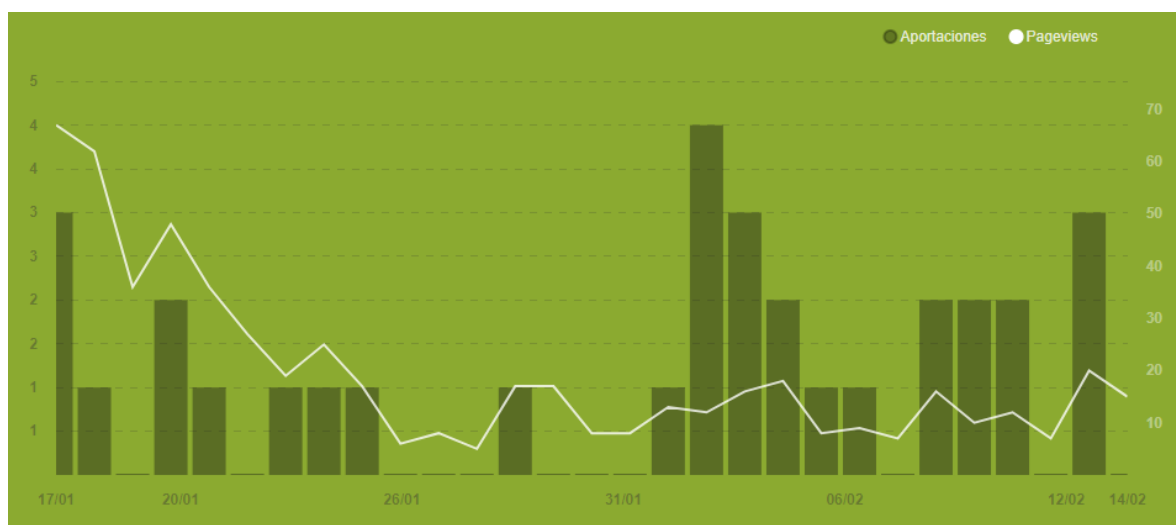


Figura 210 Visitas

Este gráfico muestra las visitas diarias a la página y las aportaciones diarias, de estos datos, se puede destacar que en el peor día se realizaron 5 visitas, el mejor día se realizan 67 visitas y en total en la campaña se llegan a realizar 569 visitas.

La media obtenida es de unas 19,6 visitas diarias, para futuras campañas, se debería intentar aumentar el número de visitas.

Por último, se muestra el gráfico de las aportaciones:

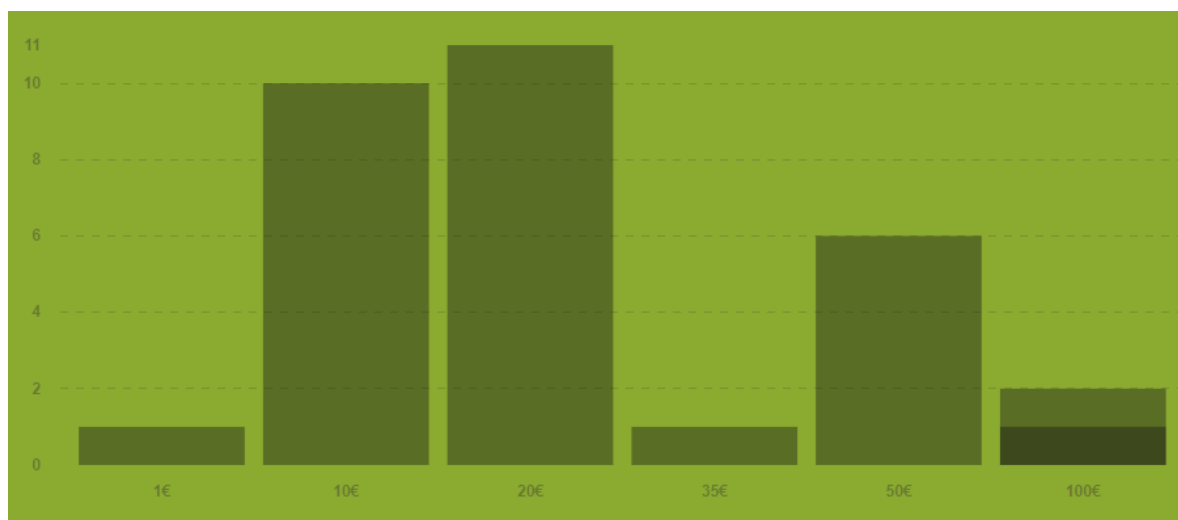


Figura 211 Aportaciones

Es decir, se realiza 1 aportación de 1€, 10 de 10€, 11 de 20€, 1 aportación de 35€, 6 de 50€ y 3 de 100€.

Por lo tanto, se realizan un total de 32 aportaciones, o lo que es lo mismo, tenemos 32 mecenas.

La aportación más común es la de 20€, siendo 29,88€ la media.

Una vez que finaliza la campaña, el autor tiene un tiempo para ponerse en contacto con los mecenas, de esta forma puede establecer un medio de comunicación para hacer llegar la recompensa, cuando Verkami confirme el cobro de todas las aportaciones, entonces se realiza una transferencia bancaria a la cuenta corriente del autor, al dinero recaudado, tienen que restarse los honorarios de Verkami y un pequeño porcentaje por cada aportación.

Estos porcentajes, corresponden al 5% del total del dinero recaudado en concepto de Herramientas de *crowdfunding*, asesoramiento y gestión y un 1,35% del total en concepto de gastos de gestión de pagos.

Una vez realizada la transacción, ya podremos añadir el sello *Crowdfunding* a nuestro proyecto para demostrar públicamente que el proyecto ha sido llevado a cabo mediante esta herramienta de financiación.



Figura 212 QR:
Campaña
Crowdfunding
Verkami

Figura 213 Sello Verkami

LINK VERKAMI: <https://www.verkami.com/projects/25687-ayudanos-a-llegar-a-la-estratosfera>

Por último, si contamos con todas las estadísticas disponibles, justo después del lanzamiento, contamos con un total de 630 visualizaciones de video en YouTube y 596 visitas a Verkami.

A estos datos falta por añadir las visitas de las páginas web con material propio subido, como las de los periódicos o el instituto.

8. LANZAMIENTO



Una vez llegado el día del lanzamiento, ya solo queda realizar tres pasos para que todo salga bien, a continuación, detallaré cada uno de ellos.

8.1 PROCESO DE LLENADO DEL GLOBO

El paso más complicado de todos consiste en realizar el llenado del globo, para que todo el trabajo se pueda realizar sin problemas, se deben realizar una serie de pasos.

Primero de todo, una vez que estemos en la zona de lanzamiento, debemos seleccionar y buscar un área que tenga como mínimo 10 (m²), deberemos limpiar esta zona de piedras, basura o cualquier cosa que pueda pinchar el globo.

Una vez que se limpie la zona, se debe desplegar una rafia o un plástico lo suficientemente rígido como para evitar que se rompa al pisar sobre él, para evitar que este se vuele, se colocarán objetos pesados en las puntas de la rafia.

Cuando la superficie de trabajo esté lista, deberemos comenzar a tomar las medidas de seguridad, de forma que la persona que manipula el globo debe evitar llevar pendientes, pulseras, relojes o cualquier otro elemento que pueda dañar el globo, además deberá vestir un mono de papel para aumentar la seguridad, por último, todos los participantes en el llenado del globo deberán llevar guantes de látex o vinilo.

Cuando el personal esté listo, se deberá situar la botella de helio en posición vertical junto a la rafia. Respecto a la botella de helio, no es necesario un manorreductor, pero es muy importante tener en cuenta que al abrir la botella el gas saldrá de forma violenta, por lo que podría dañar el globo si no se tiene cuidado.

A la hora de manejar la botella, debemos primero abrir un poco esta para conseguir un caudal mínimo para poder empezar a llenar el globo y que este no se rompa.

Una vez que la botella tenga un caudal apropiado, se deberá introducir en el globo y solo cuando sea necesario, se regulará el caudal, siempre con mucho cuidado y muy suavemente, puesto que la válvula es muy sensible y si nos pasamos podemos explotar el globo.

Cuando la botella esté lista, debemos abrir el envoltorio del globo y desplegar este sobre la rafia.

Cuando esté desplegado deberemos abrir la botella de helio como se ha explicado anteriormente e introducir la manguera por la boca del globo, una vez que se introduzca, debemos tener en cuenta que el cuello del globo es mucho más resistente que el resto, por lo que la manguera no deberá sobrepasar el cuello del globo.

Inmediatamente después de introducir la manguera, debemos realizar un nudo para estrangular el cuello del globo y la manguera, de esta forma no escapará el helio.

Para realizar esta operación, será necesario disponer de un aro de cabo, podemos cortar un cabo de nylon de metro y medio. Para realizar el aro, solo tendremos que unir los dos extremos de este mediante el nudo del “pescador”.

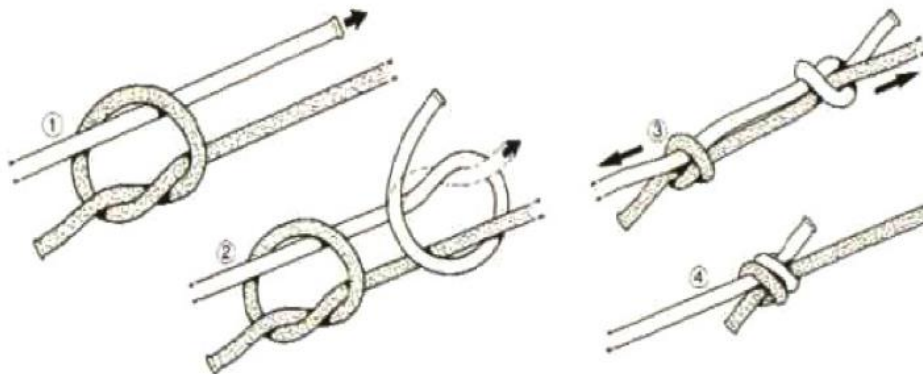


Figura 214 Pescador

Cuando tengamos listo el aro de cabo, debemos de hacer un nudo autoblocante, en este caso, vamos a realizar un “Prusik”, este debe realizarse en el cuello del globo y debemos comprobar que estrangule bien la manguera para que no escape el helio.

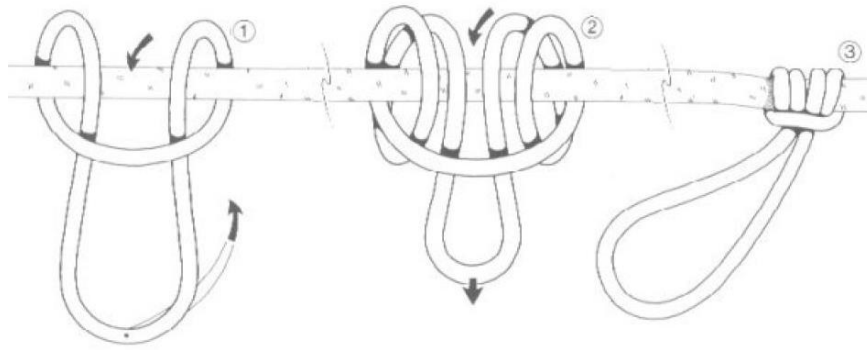


Figura 215 Prusik

Este último nudo, tiene la peculiaridad de que cuanta más tensión recibe, más estrangula, pero si se quiere soltar, solo se debe quitar un poco de tensión, esto será muy útil si necesitamos sacar la manguera.

Una vez realizado este nudo, podemos colgar de él un objeto que pese lo mismo que la sonda, de esta manera podemos ver cómo va tirando poco a poco de este objeto.

Al mismo aro de cabo, debemos atar el extremo de cabo que unirá el globo con la sonda.



Figura 216 Llenado globo METSAT-1

Una vez lleno el globo con el helio calculado, debemos soltar el objeto que hemos utilizado para comprobar como el globo empuja hacia arriba, una vez suelto y con la sonda atada correctamente, ya estamos en disposición de extraer la manguera.

Para ello, debemos sujetar la boca del globo con una mano, de forma que nuestra mano tapone la boca del globo por encima del extremo de la manguera.

Mientras, otra persona debe quitar algo de tensión al nudo “Prusik”, de esta forma se podrá extraer la manguera, inmediatamente después y con precaución de que no se escape el globo, se debe volver a estrangular el cuello del globo con el “Prusik”, de forma que no pueda escapar el helio, ahora se puede reforzar el nudo con bridas y cinta americana.

Con todos estos pasos, ya estaríamos listos para realizar el lanzamiento, el proceso será descrito en el último punto de este apartado.

Respecto al sistema de llenado, he utilizado una manguera transparente, en el extremo de la manguera que se introduce en el cuello del globo, se ha introducido por dentro un trozo de tubo metálico que evita que la manguera se estrangule y no deje pasar el helio al globo.

En el otro extremo de la manguera, se ha colocado un acople rápido para la boca de las botellas de helio, para asegurarnos de que la toma no falla, además se ha añadido una brida metálica que estrangula la toma rápida a la salida de la botella.



Figura 217 Boquilla botella

Cuando una botella se gasta, se debe doblar la manguera para evitar la pérdida de helio del globo, posteriormente se suelta la toma de la botella, se ajusta en la nueva botella, se aprieta de nuevo la brida metálica y se puede seguir llenando.

8.2 PREPARACIÓN DE LA SONDA

La puesta a punto de la sonda es otro de los puntos fundamentales, tan solo se tienen que seguir los pasos especificados en las listas de comprobación, además, para evitar problemas y perdida de material, se aconseja establecer un lugar de trabajo limpio.

Para ello se preparará una mesa de trabajo donde poner toda la electrónica y el ordenador, si es posible, la mesa se instalará en las proximidades de un coche, este coche debe estar arrancado y en él va conectado un inversor, de esta forma dispondremos de corriente alterna para poder utilizar si fuera necesario, el soldador, la pistola de silicona, etc...

Tanto en este punto como el anterior es fundamental realizar un reparto de tareas, de forma que nadie este desocupado y por un despiste pueda provocar problemas.

Para evitar esto, hemos asignado tareas a todos los participantes y hemos designado un jefe para la preparación de la sonda y otro para el llenado del globo.

Para aumentar la seguridad, hemos dispuesto chalecos reflectantes naranjas a cada uno de los participantes, así como acreditaciones. De esta forma podemos saber si alguien del grupo no está en su lugar o para identifica si alguien ajeno al proyecto se está acercando a algún sitio en el que pueda poner en peligro la misión.

Para conocer a fondo las pautas llevadas a cabo en la preparación de la sonda, se deben consultar los anexos.

8.3 PROCESO DE LANZAMIENTO

Una vez que la sonda esta lista y el globo lleno, debemos atar la sonda al paracaídas y al globo.

La línea del paracaídas y del globo, se atan mediante un “As de guía” al quitavueeltas, este tiene un mecanismo similar al de un mosquetón, de forma que una vez abierto, podemos introducir las cocas de los dos “As de guía” y posteriormente volvemos a cerrar el mecanismo.

Una vez que las dos líneas están unidas a la sonda, debemos atar el otro extremo de la línea que une la sonda con el globo y la debemos atar al aro de nylon con el que se trabajó durante el llenado del globo, para hacer esta operación, basta con realizar un “Ballestrinque” y varios cotes para asegurarnos de que no se suelte el nudo.

Cuando ya esté anudado el globo a la sonda, debemos de desplazar la sonda hacia la dirección desde la que sopla el viento, dejando así todo el cabo extendido, así evitamos nudos o enredos.

Una vez estirado, otra persona deberá coger la línea del paracaídas e introducir este por la goma que está preparada, además de deberá comprobar que como mínimo, la goma quede por la mitad del paracaídas.

Cuando esta operación esté realizada, otra persona debe coger la antena del WSPR y debe estirla evitando que coja vueltas, debe estirla siguiendo la misma dirección de la línea que une la sonda con el globo.

Finalmente, la persona que está agarrando el globo, empezará a dejar que el globo se eleve poco a poco, para ello, esta persona cogerá la línea que une el globo con la sonda e irá soltando de forma progresiva unos centímetros, mientras tanto, las dos personas que sostienen la sonda y la antena WSPR, se irán acercando hasta alcanzar a la persona que mantiene el globo, en ese momento, el que sostiene la sonda, pasará a sujetar el globo y la sonda.

En ese momento, ya se puede realizar una cuenta atrás y soltar el globo, es importante que la antena WSPR quede libre a la vez que la sonda para evitar daños.

En caso de que exista mucho viento en superficie, no es recomendable lanzar el globo, pero si se realiza el lanzamiento, entonces, lo ideal es que la sonda se disponga a sotavento⁸ del globo, de forma que el globo se soltará de golpe y se irá acercando al punto en el que se encuentra la sonda, justo antes de que el globo nos quite la sonda de las manos, deberemos correr contra el viento y soltar la sonda.

8.4 SEGUIMIENTO Y RECUPERACIÓN

Una vez lanzada la sonda, si todo va bien, con un solo teléfono móvil se podrá realizar el seguimiento de la sonda, para ello se deben acceder a los servidores y a las aplicaciones explicadas en la sección de subsistema de comunicaciones.

⁸ Sotavento: es la parte hacia donde se dirige el viento o la parte al resguardo del viento.

Para aumentar las posibilidades de recuperación, nosotros utilizamos un coche con un SDR y varias antenas que se adelantó hacia el lugar de lanzamiento para intentar captar la señal de la sonda.

Durante el vuelo, deberemos ir actualizando la última posición conocida, para poder cargarla y dirigirnos hacia ella con la ayuda de Google maps.

Cuando se empiezan a recibir posiciones prácticamente en el mismo lugar y, además, recibimos señal del GPS *tracker*, indudablemente la sonda ya está en el suelo y podremos ir a por ella.

Es importante dejar el coche en el lugar más cercano posible a la sonda, deberemos dejarlo en un lugar donde no corte el paso a otros vehículos, luego deberemos recorrer el campo en busca de la sonda.

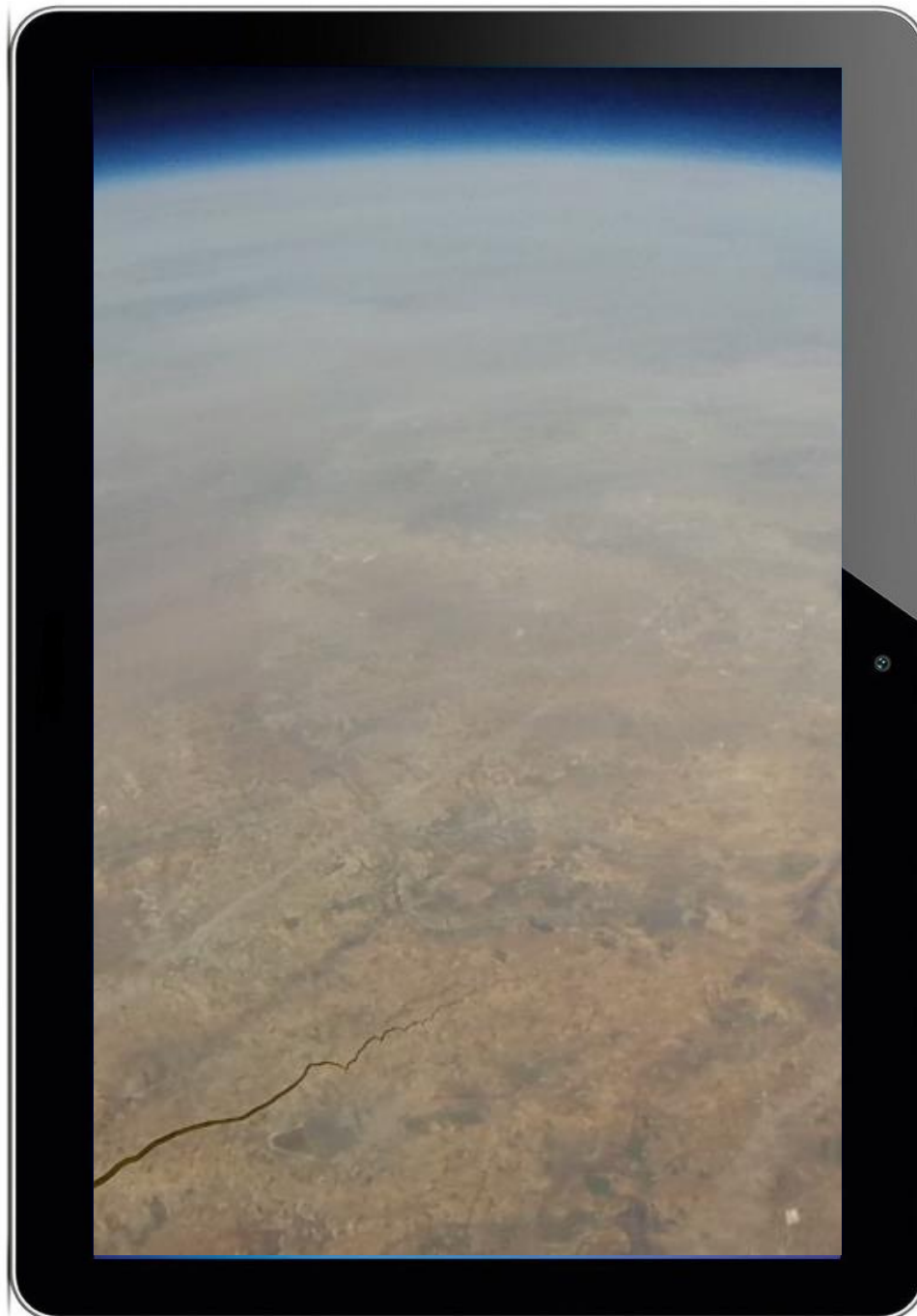
Debemos llevar siempre alguna prenda que nos identifique a distancia, de esta forma evitaremos ser alcanzados por algún cazador que nos pueda confundir con un animal.

Cuando estemos cerca de la sonda, prestaremos mucha atención a el sonido que debe emitir la sonda, esto nos ayudará a localizarla si cae en una zona muy frondosa.

Finalmente, cuando se recupere la sonda, se deben tomar fotos de donde se encontró y también documentar la apertura de la sonda.

Si algún dispositivo con tarjeta de memoria sigue funcionando, entonces deberemos apagarlo manualmente y extraer la tarjeta SD.

9. RESULTADOS Y CONCLUSIONES



En este apartado mostraremos todos los resultados obtenidos en el experimento, una vez expuestos, se podrán obtener conclusiones sobre la misión y el funcionamiento de los equipos, finalmente, se harán varias propuestas de mejora para futuros lanzamientos.

9.1 RESULTADOS

A la hora de exponer los resultados trataré de exponer cada uno de los elementos electrónicos que han formado parte del sistema de recopilación de datos, de esta forma trataré de forma individual cada uno de estos y hablaré en profundidad, sobre los datos obtenidos, el resultado esperado y el resultado obtenido.

También se comentará cada uno de los puntos clave del proyecto, como el sistema de propulsión, la simulación, el paracaídas, el control de la temperatura, etc...

Una vez expuestos todos, se procederá a sacar conclusiones de estos datos, esta parte será fundamental, puesto que se realizará una revisión de los sistemas para concluir si funcionaron correctamente y en caso de error, se mostrarán los posibles fallos.

Tabla 16 FUNCIONAMIENTO DE SISTEMAS

SISTEMA	FUNCIONAMIENTO		
	MAL	REGULAR	BIEN
GOPRO			X
SJCAM		X	
WSPR		X	
TKSTAR			X
SIGFOX			X
ARDUINO MEGA	X		
SISTEMA CALEFACCIÓN			X
CAPSULA			X
PARACAÍDAS			X
PROPULSIÓN			X
SIMULACIÓN			X
ALIMENTACIÓN			X

9.1.1 RESULTADOS SUBSISTEMA DE VIDEO

Respecto al subsistema de video, el resultado se puede considerar bueno, o por lo menos un resultado esperado o posible.

Por un lado, el funcionamiento de la cámara GOPRO ha sido perfecto, la cámara se inició unos 9 minutos antes del lanzamiento, y realizo una grabación de unas 2 horas y un minuto, por lo que el vuelo quedó totalmente registrado, la cámara se apagó unos 5 minutos antes de recuperar la sonda.

En cuanto al video, este fue realizado a 60 FPS y en 1080.

Cabe destacar que a partir del minuto 93 de grabación se aprecia un punto central en la lente, el cual difumina un poco la imagen, este es un poco de hielo que se ha formado en la lente una vez que el globo ha comenzado el descenso.



Figura 218 Congelación en lente GOPRO

Respecto al resto de las imágenes, se puede considerar que están a un nivel esperado, en el punto de *burst*, se aprecia perfectamente la curvatura terrestre y la desintegración del globo sonda.

Durante el ascenso, se aprecian los *Chemitrails*⁹ y algunas plantas fotovoltaicas.

⁹ Chemitrails: Estelas de condensación de larga duración dejadas por los aviones a gran altura.



Figura 219 Estratosfera



Figura 220 Explosión del globo "Burst"



Figura 221 QR:
Makingoff y
lanzamiento



Figura 222 QR:
Spot comercial

Respecto a la cámara SJCAM, no se pudo obtener ningún resultado, puesto que la batería se gastó a escasos minutos del lanzamiento, el problema fue que la cámara se encendió muy pronto y nos retrasamos en el llenado del globo.

9.1.2 RESULTADOS SUBSISTEMA COMUNICACIONES

El subsistema de comunicaciones funcionó bastante bien, sin embargo, existe margen de mejora.

Empiezo comentando los resultados del sistema Tkstar, éste funcionó muy bien, una vez lanzado el globo la cobertura se perdió a los 2.900 (m) de altura aproximadamente.

Durante el ascenso envié tres SMS para comprobar su funcionamiento y una vez que aterrizó envié otros tres SMS para comprobar que la posición del sistema SigFox coincide con la del GPS *tracker*.

El punto de aterrizaje se produjo en la posición: 39°36′28.9″ N 002°54′43.3″ W

Respecto a la aplicación web, el funcionamiento también fue correcto, tanto en los primeros minutos del lanzamiento como en los minutos finales y una vez en tierra, se pudo hacer un seguimiento en tiempo real.

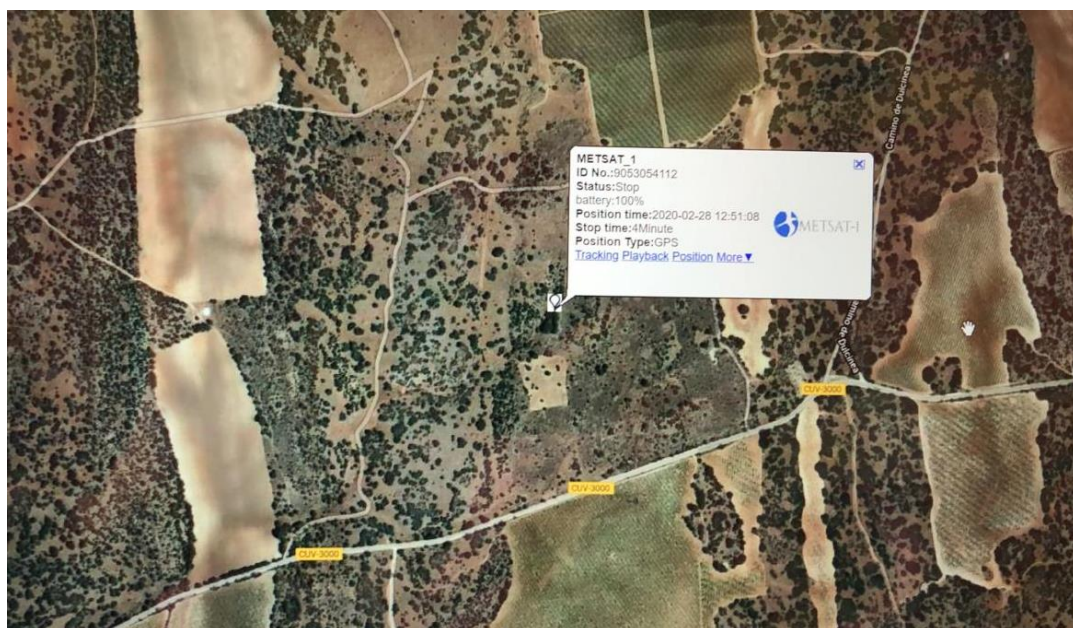


Figura 223 Posición final GPS tracker

Respecto a las alarmas de *geofences* y de velocidad, el registro es el siguiente:

Geo-fence Out(YEBENES)	2020/02/28 12:47	2020/02/28 12:47
Geo-fence Out(CHUECA)	2020/02/28 12:46	2020/02/28 12:46
Geo-fence In(HINOJOSOS)	2020/02/28 12:46	2020/02/28 12:46
Geo-fence In(CHUECA)	2020/02/28 11:04	2020/02/28 11:04
Geo-fence Out(CHUECA)	2020/02/28 11:02	2020/02/28 11:02
Geo-fence In(YEBENES)	2020/02/28 10:04	2020/02/28 10:04
Geo-fence In(CHUECA)	2020/02/28 10:03	2020/02/28 10:03

Figura 224 Alarmas TKSTAR

Si acudimos al registro de *geofences*, podemos ver que al inicio se entra en los Yébenes y en Chueca, puesto que Orgaz se encuentra en ambas *geofences*.

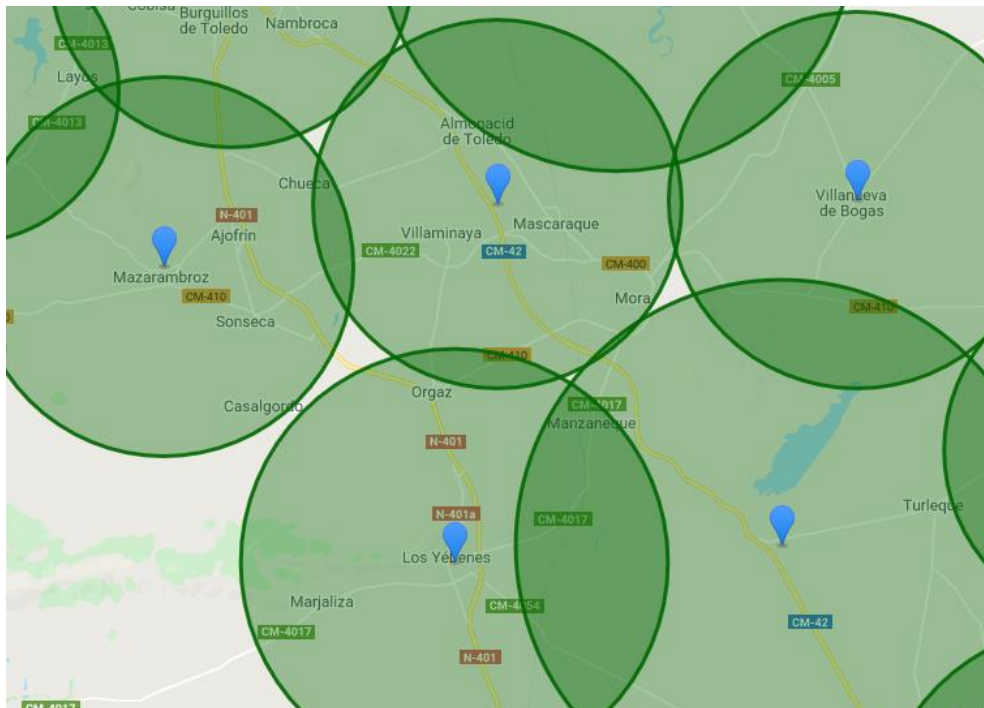


Figura 225 Geofences despegue

Finalmente, durante el aterrizaje, detecta que ha salido de los anteriores *geofences* y detecta que ha entrado en el *geofence* de Los Hinojosos.

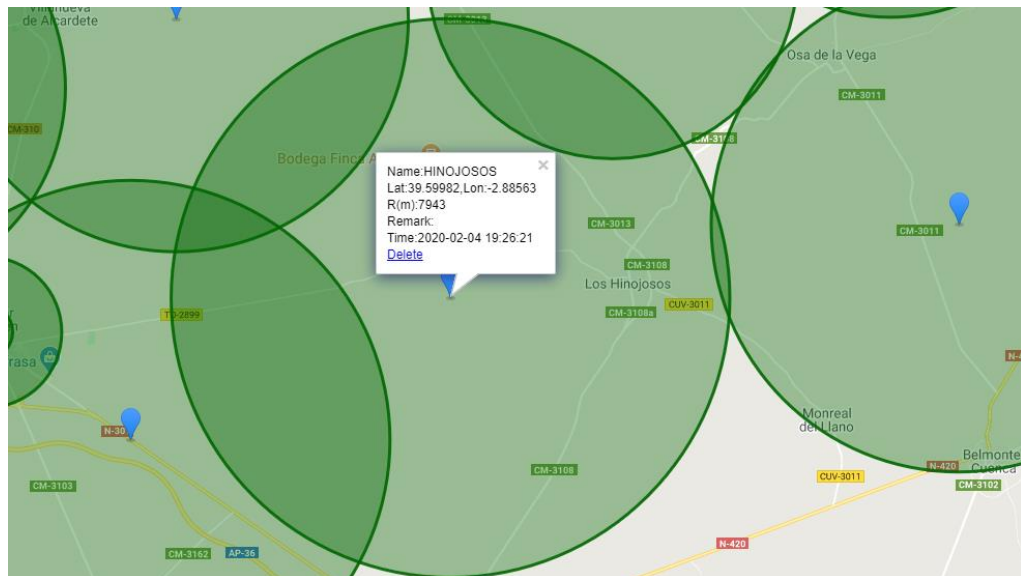


Figura 226 Geofences aterrizaje

Además, a través de la aplicación, podemos descargar el recorrido seguido por la sonda.

En la zona de lanzamiento se obtiene el siguiente resultado.

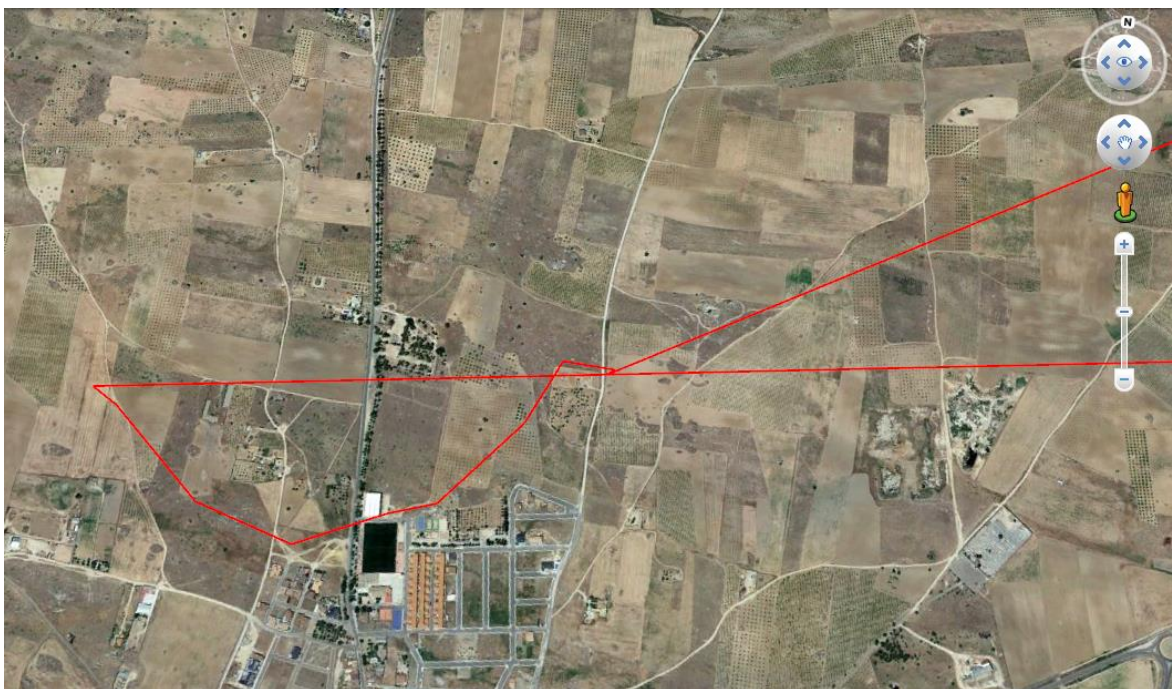


Figura 227 Track despegue

Se aprecia como el viento lleva al globo hacia el oeste y luego lo lleva hacia el este volviendo a pasar sobre el punto de lanzamiento.

A continuación, se pierde la señal GPS y solo se muestra una línea que une el último punto conocido antes de perder la señal con el primer punto una vez recuperada la señal GPS.



Figura 228 Track intermedio

Por último, se aprecia un último tramo muy irregular que corresponde con la trayectoria seguida desde la explosión del globo hasta el aterrizaje.



Figura 229 Track aterrizaje

Con todos los datos recibidos del sistema de GPS *tracker*, podemos considerar que el sistema ha funcionado muy bien.

A continuación, expongo los datos obtenidos del sistema SigFox.

Una vez puesto en funcionamiento el sistema de SigFox, comprobé que algo no estaba funcionando bien, puesto que no estaba enviando datos a los servidores,

para cerciorarme de que estaba funcionando, encendí el SDR y comprobé el espectro electromagnético en 868 (Mhz), entonces vi que cada dos minutos enviaba una trama, pero que esta no llegaba al repetidor.

Al reconocer la zona, me di cuenta de que estábamos a menos de 500 metros de un repetidor, esto pudo hacer que la emisión no llegase al repetidor por que el ancho de banda estaba saturado.

Una de las pruebas de funcionamiento del sistema se realizó en las inmediaciones de un repetidor y los resultados fueron los mismos, el sistema no era capaz de funcionar bien, sin embargo, en cuanto la distancia con el repetidor aumentaba, la señal mejoraba hasta que el sistema conseguía funcionar con normalidad.



Figura 230 Repetidor Orgaz

Teniendo en cuenta que este tipo de error era conocido y una vez comprobado con el SDR, que el equipo emite, se decide lanzar la sonda.

Durante el vuelo, se enviaron un total de 48 mensajes con posición, de los cuales 30 de ellos son posiciones reales, 2 de ellas son erróneas y los restantes son la posición de aterrizaje.

Con estos puntos se puede realizar una trayectoria seguida por la sonda hasta que esta alcanzó los 18.000 (m), en ese momento dejó de enviar posiciones y empezó a enviar temperaturas.

Justo antes de perder la cobertura GPS por la altura, se recibió una posición falsa, posiblemente debido a que estaba al límite de la cobertura y por eso se decodificó con error.

El primer mensaje recibido fue a las 11:00 y el último antes de perder la cobertura GPS fue a las 11:30.

Finalmente, la Cobertura GPS se recuperó a las 12:25 y se produce el aterrizaje sobre las 12:45.

Durante el descenso, se obtiene una posición errónea, al sobrevolar el polígono industrial de Quintanar de la Orden, posiblemente por la cercanía a algún repetidor.



Figura 231 Track despegue MKRFOX



Figura 232 Track aterrizaje MKRFOX

Una vez expuestos los datos de posición, se procede a exponer los datos de temperaturas, el primer mensaje de temperaturas se recibe a las 11:33 y el último a las 12:23.

En total se reciben 29 mensajes, con la temperatura interna de la sonda.

La temperatura comienza en 24°C y la última temperatura recibida es de 19°C, por lo que, a lo largo de 50 minutos, la temperatura ha descendido 5°C en el interior de la sonda.

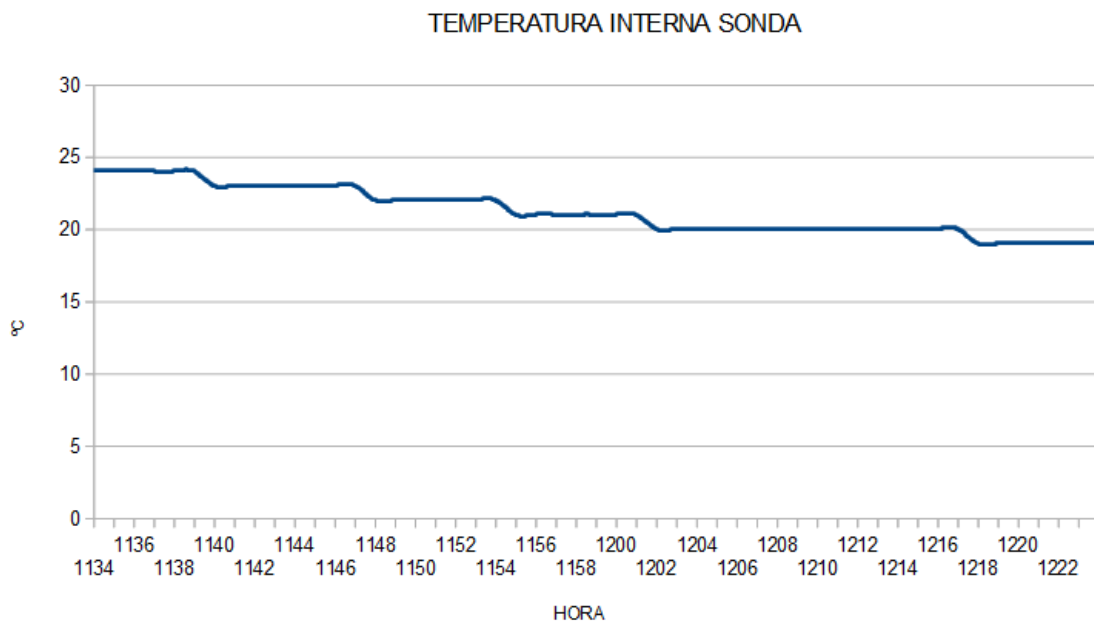


Figura 233 Temperatura interna

Además de los datos recogidos tanto de temperaturas como de posiciones, también se han obtenido otros datos, estos corresponden con el alcance de la cobertura SigFox.

Durante todo el vuelo, la comunicación con la sonda fue continua, no se perdió la cobertura en ningún momento.

Si tenemos en cuenta que la altura máxima alcanzada ha sido de 29.000 (m) aproximadamente, podemos establecer que el alcance como mínimo es de 29 (Km), pero lo más posible es que el repetidor se encuentre a mucha más distancia, ya que si el alcance es de 29 (Km), significaría que el repetidor está justo debajo de la sonda.

Con seguridad, el alcance será superior a los 29 (Km), según el fabricante, el alcance puede llegar a los cientos de kilómetros.

Respecto al funcionamiento del servicio Atlas, el funcionamiento no ha sido el esperado, no se pudo obtener una aproximación hasta el aterrizaje, y una vez alcanzada la zona de aterrizaje, la aproximación no fue nada buena.



Figura 234 ATLAS sobre zona de aterrizaje

Las zonas posibles, están muy dispersas y es difícil precisar donde puede estar la sonda.

Esto pudo deberse al movimiento del globo, ya que puede producir un efecto Doppler, que, a pesar de ser muy tenue, puede provocar una mala recepción en los repetidores.

A continuación, muestro los datos acumulados durante el vuelo en el servidor.

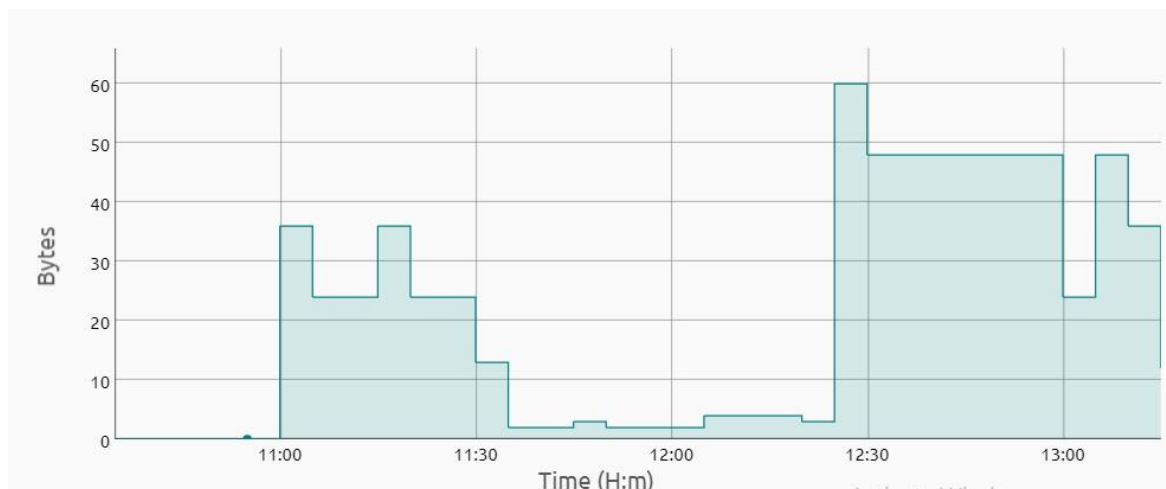


Figura 235 Bytes vs tiempo

En esta primera gráfica, se muestra la cantidad de Bytes enviados a lo largo del vuelo, se aprecia como de las 11:00 hasta las 11:30, la cantidad se mantiene aproximadamente en 25 Bytes de media, sin embargo, a partir de las 11:30, hasta las 12:30, la cantidad de Bytes se reducen a apenas 4 Bytes, este periodo corresponde con la altura superior a 18.000 (m), en la que solo se envían temperaturas.

Finalmente, de 12:30 en adelante, la cantidad de información aumenta, ya que se vuelven a enviar posiciones y porque los mensajes empiezan a enviarse cada minuto en vez de cada dos minutos.

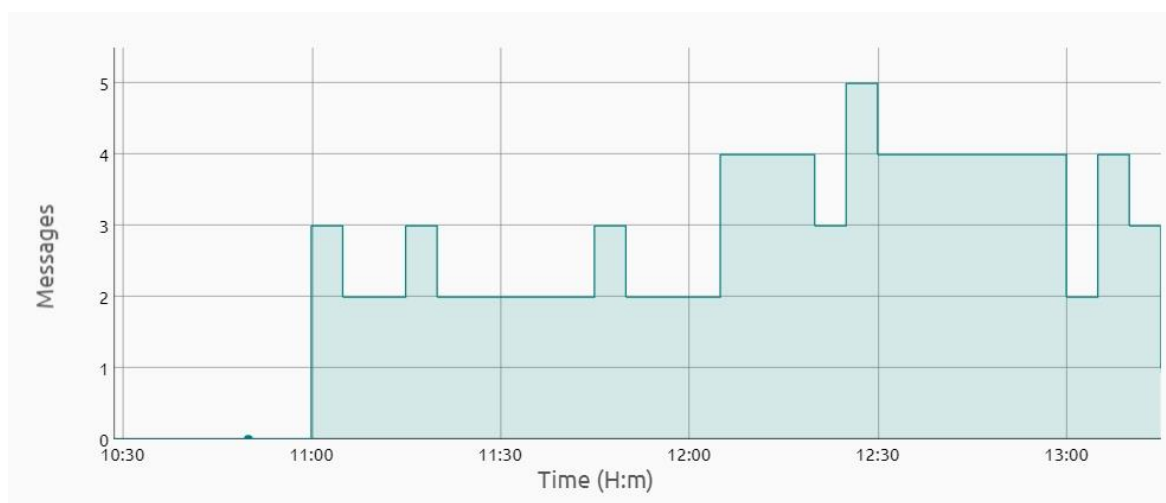


Figura 236 Mensajes vs tiempo

En esta gráfica, se ve como a partir de las 12:10, se empiezan a enviar el doble de mensajes durante la misma fracción de tiempo.

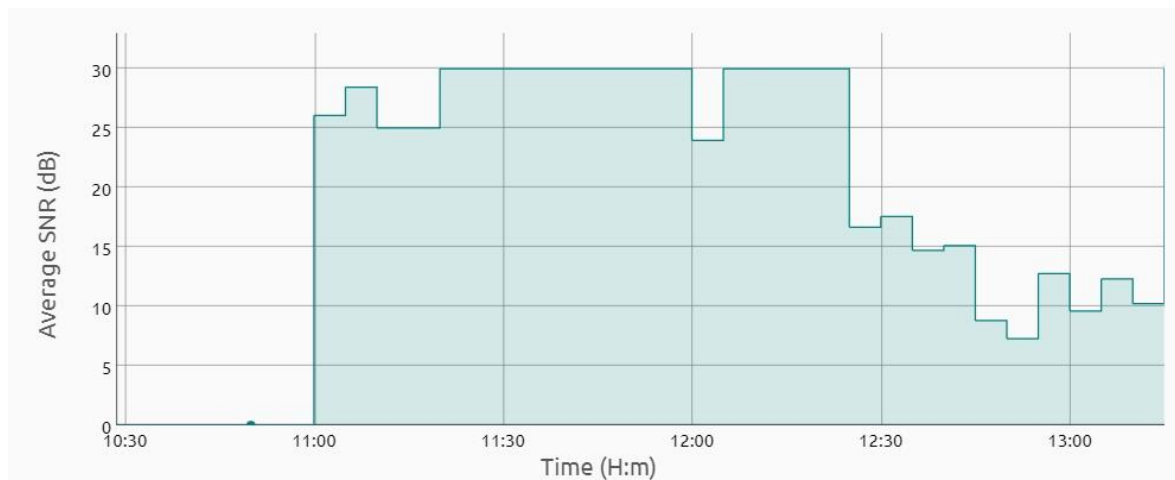


Figura 237 SNR vs tiempo

La gráfica anterior muestra la relación señal ruido a lo largo del vuelo.

Se aprecia que desde el lanzamiento hasta las 12:25, la relación de señal ruido se mantiene prácticamente constante en 30 (dB), sin embargo, a partir de las 12:25, la relación señal ruido cae prácticamente a la mitad y se va reduciendo hasta llegar a los 8 (dB) en el momento del aterrizaje.

Si recurrimos al teorema de Shannon, podremos entender que está sucediendo.

$$C = B * \log_2\left(1 + \frac{S}{N}\right)$$

C, es la velocidad de la comunicación en bits por segundo.

B, es el ancho de banda en Hz.

S/N, es la relación señal ruido.

Durante los primeros minutos del lanzamiento, la velocidad se mantiene constante, pero luego la velocidad se duplica, justo en el momento en el que la relación señal ruido empieza a disminuir.

Si suponemos que la velocidad C_1 , es de 1 (bit/s) y que el ancho de banda es de 2 (Hz), entonces:

$$C_2 = 2 * C_1 = 2 \text{ (bit/s)}$$

$$B = cte = 2 \text{ (Hz)}$$

$$1 = \log_2\left(1 + \frac{S}{N_2}\right)$$

$$2^1 = \left(1 + \frac{S}{N_2}\right)$$

$$1 = \frac{S}{N_2}$$

Para el primer caso, si tomamos también el mismo ancho de banda y la mitad de la velocidad, entonces:

$$\frac{1}{2} = \log_2\left(1 + \frac{S}{N_1}\right)$$

$$2^{\frac{1}{2}} = 1 + \frac{S}{N_1}$$

$$1,41421356 - 1 = \frac{S}{N_1}$$

$$0,41421356 = \frac{S}{N_1}$$

Por lo tanto, si comparamos los dos términos de la relación de señal ruido, podemos ver como al aumentar la velocidad, disminuye la relación de señal ruido.

En este caso, la reducción es casi del 59%, por lo que los datos observados anteriormente se explican fácilmente con esta sencilla demostración matemática.

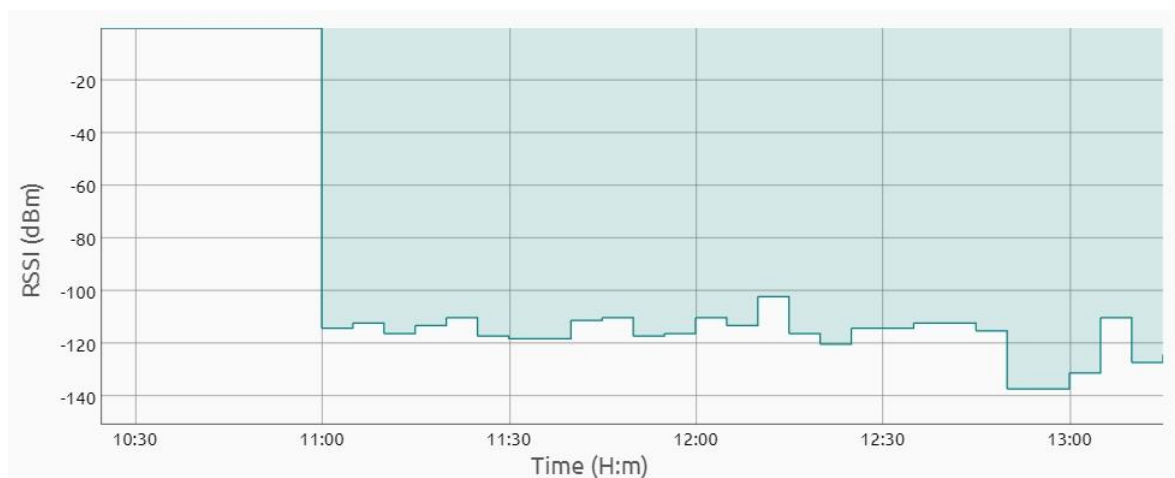


Figura 238 RSSI vs tiempo

Respecto a la potencia de recepción de los repetidores, podemos ver que prácticamente este valor se mantiene constante entre -100 (dBm) y los -120 (dBm).

Esta, sin embargo, aumenta hasta los -140 (dBm) una vez que el dispositivo aterriza y se mantiene en ese lugar, posiblemente debido a que estaba prácticamente a nivel de suelo en un terreno irregular y rodeado de árboles.

Con todos los datos expuestos, se puede determinar que el sistema SigFox ha funcionado mucho mejor de lo esperado y aun existiría margen de mejora.

Por último, expongo los resultados del sistema WSPR.

Este sistema funcionó correctamente, se pudo comprobar tanto en el lanzamiento como en la recuperación de la sonda, que los indicadores led funcionaron correctamente y el WSPR funcionó como radiobaliza.

Sin embargo, no fue posible que ninguna estación recibiera nuestra señal, al igual que con el resto de los dispositivos, comentaré en las conclusiones que pudo ocurrir.

Además, durante el descenso, la antena del WSPR, se enredó en la línea del paracaídas y al empezar a girar, la torsión terminó arrancando la antena en la parte de la soldadura.



Figura 239 Pérdida antena WSPR

9.1.3 RESULTADOS SUBSISTEMA DE COMPUTADORA

Como se puede ver en el cuadro al comienzo de esta sección, el autómata no funcionó correctamente.

Durante los 10 primeros minutos, el sistema funcionó de forma correcta, hasta que se precintó la sonda.

En ese momento, el programa dejó de funcionar correctamente, Sin embargo, ya era demasiado tarde para corregirlo, pues ya estaba todo listo y tratar de solucionarlo podría retrasar el lanzamiento en más de media hora, a lo largo del día del lanzamiento, el tiempo empeoró, por ello, decidimos lanzarlo.

La forma en la que se descubrió la avería fue a través de las señales auditivas programadas con tal fin, durante los primeros 20 minutos la señal debería ser intermitente y posteriormente debía apagarse, en los últimos 20 minutos del vuelo, volvería a pitar intermitentemente y de forma continua una vez completado el vuelo.

Sin embargo, a partir de los 12 minutos, la alarma empezó a sonar de forma continua, esta no es una opción posible, a no ser que el programa se quede pillado en la lectura de los sensores o en la escritura de la SD.

En el apartado de conclusiones explicaré, los posibles errores.

En este apartado, muestro los datos obtenidos los 10 primeros minutos, correspondientes a la preparación del lanzamiento. 3 minutos más tarde, se lanzaría el globo.

Respecto a los datos del GPS, cabe destacar que la primera posición se obtuvo a los 3 minutos del inicio del autómata.

El resto de los datos, los muestro en gráficas.

La primera gráfica que muestro corresponde a la humedad, esta varía entre el 33% y el 36%.

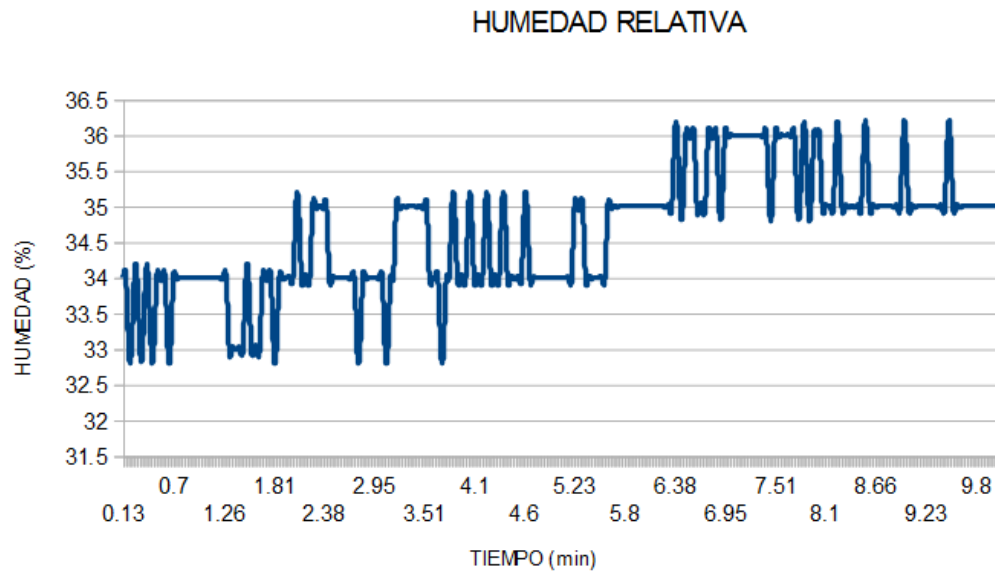


Figura 240 Humedad interior de la sonda

El siguiente gráfico muestra los dos termómetros internos de la capsula, comparados con la temperatura externa.

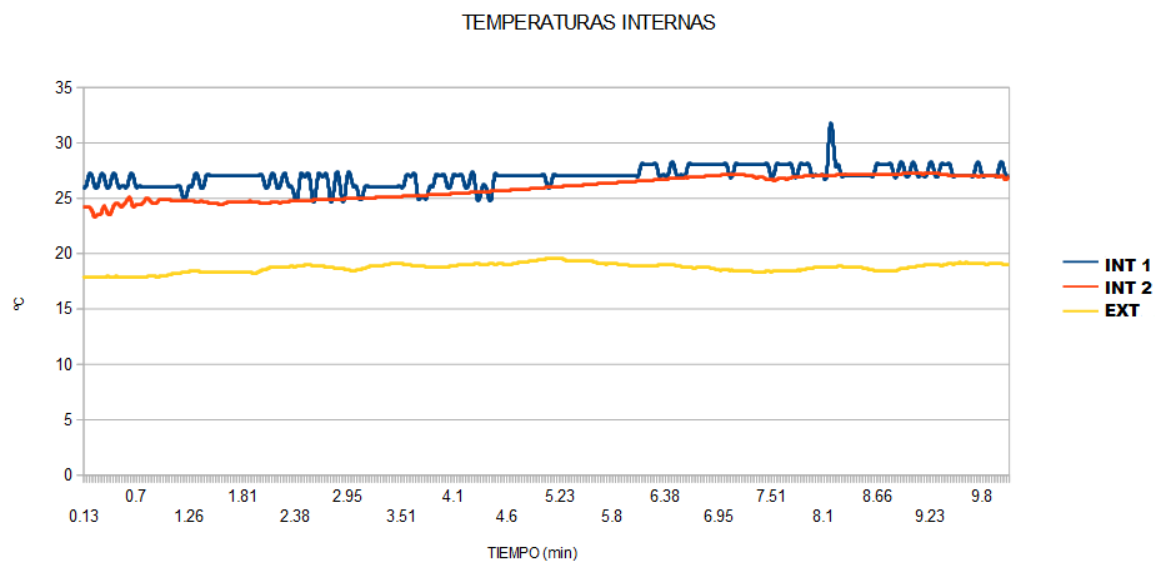


Figura 241 Temperaturas antes del lanzamiento

Se puede observar que la temperatura externa se mantiene constante entre los 18°C y los 20°C, sin embargo, en el interior de la sonda, se aprecian los efectos del calentador, puesto que la temperatura es superior entre 5 y 8 grados por encima de la temperatura ambiente, además, se aprecia como la temperatura va en aumento, desde los 24°C hasta los 27°C al cabo de 10 minutos.

A continuación, se muestran los datos de presión atmosférica.

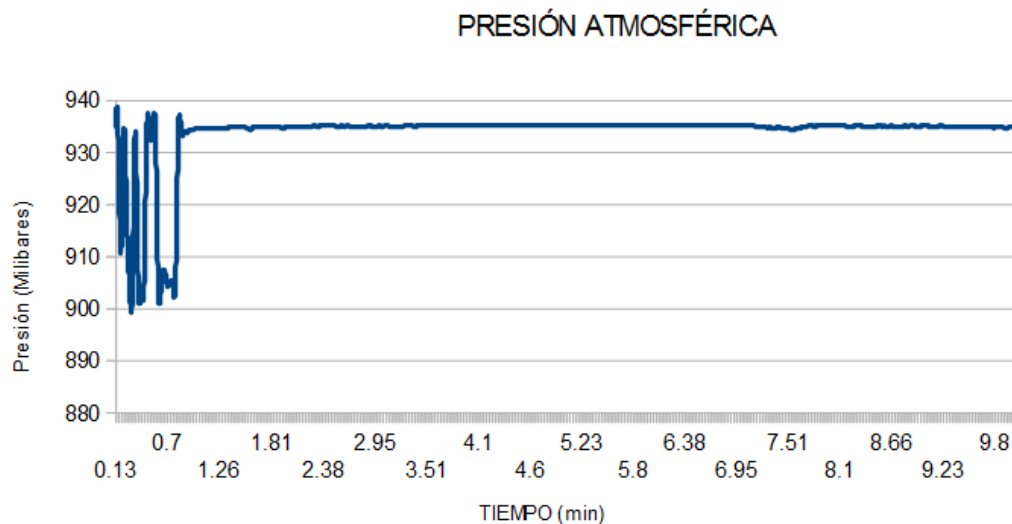


Figura 242 Presión antes del lanzamiento

Durante el primer minuto, se observa mucho ruido, pero justo después el sensor se estabiliza en 934 milibares.

Si tenemos en cuenta una tabla de equivalencias de presión a altura, se puede determinar que el lanzamiento se realizó aproximadamente a 650 m.s.n.m.

Respecto a los datos de los giróscopos, he decidido omitirlos puesto que la sonda no se movió de la mesa de trabajo.

9.1.4 RESULTADOS SUBSISTEMA DE POTENCIA

El subsistema de potencia ha funcionado según lo esperado, todos los equipos se han alimentado correctamente durante el vuelo, independientemente del buen funcionamiento de estos equipos.

El grupo de baterías del GPS *tracker*, aún marcaba un 100% de batería tras la recuperación de la sonda.

SigFox, siguió funcionando hasta la recuperación, en ese momento se desconectó la batería, sin embargo, aún quedaban horas de duración.

El sistema de Arduino y WSPR, funcionó todo el vuelo, en la recuperación se desconectó la alimentación, pero la batería duró cerca de 8 horas más.

Tanto las fuentes de alimentación como las pilas funcionaron en unos muy buenos márgenes de temperaturas, por lo que desempeñaron su trabajo correctamente.

9.1.5 RESULTADOS SUBSISTEMA DE CALEFACCIÓN

El sistema de calefacción funcionó mejor de lo esperado.

Desde el inicio del lanzamiento, con una sola fuente de calor, se consiguió elevar la temperatura hasta los 27°C, pocos minutos después, la sonda se terminó de cerrar herméticamente y al cabo de 33 minutos después del lanzamiento, es decir, a las 11:33, la temperatura interna era de 24°C.

A partir de ese momento, se registró la temperatura hasta las 12:23, momento en el cual, la temperatura interna era de 19°C.

En el punto más alto del vuelo, la temperatura descendió en torno a los 5 grados en 50 minutos, o lo que es lo mismo, 0,1°C el minuto.

Por lo que, si tenemos en cuenta el descenso de 27°C a 24°C en 33 minutos, y aplicamos el mismo descenso de temperatura por minuto, podemos comprobar que los cálculos se cumplen.

Es decir:

$$\frac{24(^{\circ}C) - 19(^{\circ}C)}{50 (min)} = 0,1\left(\frac{^{\circ}C}{min}\right)$$

Por lo que, en 33 minutos, la temperatura debería descender:

$$33(min) * 0,1\left(\frac{^{\circ}C}{min}\right) = 3,3^{\circ}C$$

Si a los 27°C iniciales le quitamos 3,3°C, entonces daría aproximadamente 24°C, por lo tanto, podemos dar por buena la aproximación, a continuación, realizaremos la misma aproximación para mostrar la temperatura al final del vuelo.

Teniendo en cuenta que los datos de temperaturas dejan de estar disponibles a las 12:23 y que la hora del aterrizaje es a las 12:45, entonces transcurren 22 minutos.

Aplicando la misma fórmula, la temperatura debe descender en 2,2°C

Es decir, la temperatura en el aterrizaje fue de 17°C.

Una vez recuperada la sonda, se pudo comprobar que la sonda seguía caliente a una buena temperatura, no se observó condensación.

Al levantar la parte aislante de la espuma de poliuretano, comprobé que la cámara SJCAM estaba muy fría y la Gopro aún seguía caliente, sin embargo, por la parte externa estaba algo congelada y la parte inferior de la sonda presentaba una mínima condensación.

Con estos datos obtenidos se ratifica el buen funcionamiento del sistema de calefacción, con solo 30 (g) de peso, se consigue un resultado óptimo en el aislamiento y control interno de temperatura.

9.1.6 RESULTADOS PAYLOAD BOX

La elección de una estructura esférica resulta un acierto, puesto que es una estructura aerodinámica que ayuda a que, en el descenso, el coeficiente de rozamiento sea menor y así la caída sea más rápida y estable.

Además, durante el ascenso, al no poseer aristas, se evita el efecto de veleta y se obtienen imágenes muy estables.

Un problema que presenta esta estructura es la forma en la que disponer los dispositivos electrónicos en el interior, sin embargo, con un poco de ingenio, se consigue resolver fácilmente.

Esta estructura es también muy resistente, aguanta pesos de más de 70 (Kg) sobre ella y al disponer solamente de dos piezas, es muy fácil de aislar el interior cuando las dos piezas se encajan.

Al recuperar la sonda, solo se apreciaron unos rasguños en la pintura y dos pequeñas perforaciones en la parte inferior producidas por las ramas de los árboles, sin embargo, estas ramas no llegaron a atravesar la pared de la sonda, solo penetraron algo más de un centímetro.

También se ha podido comprobar que el sistema para proteger la sonda de un fuerte tirón de la antena WSPR, ha funcionado a la perfección porque se ha partido el estaño de la soldadura antes que la estructura de poliestireno.

9.1.7 RESULTADOS SUBSISTEMA PROPULSIÓN Y PARACAÍDAS

El globo, finalmente ascendió a una velocidad comprendida entre los 6 (m/s) y los 7 (m/s), aproximadamente 6,4 (m/s).

El globo se desintegró a unos 28.900 (m) aproximadamente.

El paracaídas se desplegó al cabo de 11 minutos desde la explosión del globo, la velocidad de caída fue superior a los 84 (km/h), aproximadamente 23 (m/s).

Una vez desplegado, la velocidad se ralentizó, hasta aproximadamente los 7 (m/s).

Con estos datos se puede determinar que ambos elementos funcionaron según lo previsto.

9.1.8 RESULTADOS SIMULACIÓN

Se realizaron dos simulaciones, la primera de ellas fue la que utilizó los datos reales, mientras que la segunda fue la utilizada, esta última fue la de los datos conservadores.

La simulación conservadora, estableció la altura máxima en 29.000 (m), velocidad de subida y de bajada a 6 (m/s).

Con esta simulación, el aterrizaje estaba previsto a unos pocos kilómetros antes de Osa de la vega, a unos 8,5 (Km) del punto de aterrizaje.

La simulación con los datos reales establecía el punto de aterrizaje a unos 6 (Km) del punto de aterrizaje real.

9.2 CONCLUSIONES

Una vez expuesto el funcionamiento de todo el conjunto y conociendo los errores y aciertos, expongo una serie de conclusiones encaminadas a solventar los posibles errores y también expondré las posibles razones por las que algunos equipos no funcionaron correctamente.

Solo se expondrán conclusiones de sistemas que no funcionaron según lo esperado o que, a pesar de su funcionamiento, tras un examen de los resultados se ha observado algún dato que puso de manifiesto un problema que pudo poner en compromiso la misión.

Por otro lado, al final de este punto, se enumeran en una tabla los objetivos iniciales de este proyecto, se mostrará si el objetivo se cumplió o no y el resultado.

9.2.1 CONCLUSIÓN SUBSISTEMA DE VIDEO

La principal conclusión obtenida en el subsistema de video es que no merece la pena utilizar cámaras de muy bajo precio.

Lo ideal es utilizar una cámara de gama media o alta, preferiblemente esta cámara debe de carecer de pantalla para ahorrar energía, en caso de que la cámara tenga pantalla, es muy importante que esta no sea táctil.

Una vez que se tenga la cámara, es importante realizar una prueba con la cámara en el congelador, en caso de que la batería dure menos de 1 hora, no merece la pena enviar la cámara, puesto que ocupa mucho espacio y suelen ser relativamente pesadas.

Respecto posición de la cámara, si la idea es captar la curvatura terrestre, lo ideal es situar la cámara de forma que cuando la sonda este ascendiendo, la cámara este enfocando hacia el horizonte, para captar algo más de la superficie terrestre, podemos darle una ligera inclinación hacia abajo.

9.2.2 CONCLUSIÓN SUBSISTEMA COMUNICACIONES

Respecto al sistema SigFox, cuando envía la latitud y la longitud, también envía la altitud, sin embargo esta solo funcionó hasta los 3.000 (m) aproximadamente, por lo que se puede mejorar añadiendo un barómetro que indique la altitud hasta los 11.000 (m) con precisión o también utilizar un GPS preparado para gran altura, estos son algo más caros, pero funcionan a más de 30.000 (m) de altura, de esta forma se podría obtener la posición en directo durante todo el vuelo y además se podrían tener las alturas con precisión.

En caso de que no fuese capaz de obtener una posición, el mensaje que envía con la temperatura interna podría mejorarse, de forma que se podrían añadir más datos, como, por ejemplo, la temperatura externa, un bit de comprobación para comprobar si está subiendo o bajando, o incluso altura barométrica y velocidad de subida o bajada.

Además, el sistema SigFox posee cuatro mensajes diarios de subida, que podrían ser utilizados para actuar sobre la sonda, como por ejemplo encender o apagar equipos, conmutar baterías, etc...

Por último, a la hora de examinar los datos del *Backend*, se aprecia como en los últimos minutos, al enviar un mensaje por minuto, la relación señal ruido empeoró, por lo que lo ideal es enviar como mucho un mensaje cada dos minutos, y en caso de tener que enviar más mensajes, como mucho se debería enviar un mensaje cada minuto y medio.

Respecto al sistema de radiobaliza WSPR, no se pudo obtener ninguna trama, existen tres posibles causas para esto.

La primera de ellas es la posibilidad de que la propagación no fuera la adecuada y que la emisión fuese absorbida por la ionosfera, durante el día la propagación empeora, pero mejora por la noche.

Otra posible causa es la antena utilizada, ya que para ahorrar peso y evitar que se enrede, se utilizó una antena de un cuarto de onda, posiblemente con una antena de 20 metros el resultado podría haber sido mejor.

La carga impedancia de la antena no llegaba a los 50 (Ω), esto pudo provocar que parte de la señal se reflejase.

Además, la antena se perdió durante el descenso, por lo que durante una gran parte del vuelo fue imposible establecer comunicación.

Finalmente, también existe la posibilidad de que nadie fuese capaz de recibir la señal, Si realizamos una simulación de emisión de punto a punto y analizamos el perfil entre el emisor y el receptor más cercano, situado en Cartagena, podemos ver que solo están en visión directa cuando el globo supera los 26.000 (m) de altura.

Para una altura inferior del globo, la visión directa es imposible.



Figura 243 Radio enlace hasta 26.000 m

Por ejemplo, el perfil a 5.000 (m) de altura sería el siguiente:



Figura 244 Radio enlace a 5.000 m

Se ve claramente como las montañas impiden la visión directa entre emisor y el receptor.

9.2.3 CONCLUSIÓN SUBSISTEMA DE COMPUTADORA

Antes del lanzamiento se realizaron varias pruebas para depurar el código, así como para comprobar que todo funcionaba correctamente, sin embargo, durante el vuelo algo falló e impidió la toma de datos de los sensores.

Tras analizar el error, pude comprobar que la configuración antes del lanzamiento se realizó correctamente y que el error es debido a que en una de las instrucciones del void loop(), el programa se queda pillado.

Existen dos posibles opciones para que suceda esto, una de ellas es un problema en el Hardware y la otra posibilidad puede ser un error en la programación.

Al analizar el código, se puede comprobar que el programa se quedó pillado en una de estas tres líneas siguientes, puesto que, si llega a la cuarta línea, el zumbador se apagaría y pitaría de forma intermitente.

```
readTempHumedad(); // DHT11 Y DS18B20

readBMP(); // BMP180

readAccGyro(); //MPU

digitalWrite(BUZZ,LOW);
```

Si comprobamos cada una de las funciones anteriores, podemos ver el siguiente código:

```
.
void readTempHumedad() {

    h = dht.readHumidity();
    t = dht.readTemperature();

    sensor.requestTemperatures();
}
```

A la hora de obtener datos de los sensores *1-Wire*, no se aprecia ninguna línea que pueda dejar pillado el programa.

Si analizamos los sensores BMP180 y el MPU6050 del bus I²C, podemos ver las siguientes líneas de código:

```
void readBMP() {  
  
    status = bmp180.startTemperature(); //Inicio de lectura de temperatura  
  
    if (status != 0)  
    {  
        delay(status); //Pausa para que finalice la lectura  
        status = bmp180.getTemperature(T); //Obtener la temperatura  
        if (status != 0)  
        {status = bmp180.startPressure(3); //Inicio lectura de presión  
        if (status != 0)  
        {delay(status); //Pausa para que finalice la lectura  
        status = bmp180.getPressure(P,T); //Obtenemos la presión  
        }  
    }  
}  
}
```

En estas líneas, tampoco se aprecia ninguna condición que pueda crear un bucle infinito.

```
void readAccGyro() {  
  
    // valores del Acelerometro  
    Wire.beginTransmission(MPU);  
    Wire.write(0x3B); //Pedimos el registr  
    Wire.endTransmission(false);  
    Wire.requestFrom(MPU, 6, true); //A part  
    AcX=Wire.read()<<8|Wire.read(); //Cada  
    AcY=Wire.read()<<8|Wire.read();  
    AcZ=Wire.read()<<8|Wire.read();  
  
    // valores del Giroscopio  
    Wire.beginTransmission(MPU);  
    Wire.write(0x43);  
    Wire.endTransmission(false);  
    Wire.requestFrom(MPU, 4, true); //A dife  
    GyX=Wire.read()<<8|Wire.read();  
    GyY=Wire.read()<<8|Wire.read();  
}
```

Sin embargo, si se produce un mal funcionamiento en el bus, estas líneas podrían causar problemas, la otra opción es que estas funciones que llaman a las librerías tengan una condición que desencadene un bucle infinito.

A la hora de analizar las librerías del BMP180, no se encuentra nada anómalo, en caso de fallo del sensor, el resultado sería nulo, pero no se bloquearía el programa.

Lo mismo sucede con el bus *1-Wire*, por lo tanto, solo queda analizar el sensor MPU6050.

Al comprobar las librerías del MPU6050, tampoco se observa ninguna condición que desencadene en un bucle infinito.

Por lo tanto, tras hacer pruebas posteriores, he obtenido la conclusión de que es un fallo puntual, posiblemente por el desbordamiento de una variable o por la pérdida de un puntero¹⁰.

Cuando la memoria del Arduino está cerca de su límite, este error es muy común, sin embargo, al tener margen de memoria, la probabilidad de que se produzca este error disminuye, pero sigue siendo posible, como se ha podido ver en este caso.

Por lo tanto, se puede decir que el programa funciona bien, el error es debido a otros factores no directamente relacionados con nuestro código, sin embargo, con un análisis a fondo del código, se podría disminuir aún más la probabilidad de que se vuelva a producir este error.

9.2.4 CONCLUSIÓN SUBSISTEMA PROPULSIÓN Y PARACAÍDAS

Respecto a la disposición del sistema de unión de la sonda con el paracaídas y con el globo, la conclusión es que existe mucho riesgo de que la goma que sujeta el paracaídas se suelte durante la subida y pueda desviar mucho la trayectoria del globo.

Por otro lado, en el descenso se produjo un gran enredo que pudo poner en peligro la recuperación de la sonda intacta.

Por ello, opino que utilizar el sistema de poner la línea que une el globo con la sonda atravesando el paracaídas, podría ser una mejor opción.

También se podrían realizar otras alternativas que se expondrán en el punto de posibles mejoras.

¹⁰ Puntero: Un puntero es un objeto del lenguaje de programación, cuyo valor se refiere a (o "apunta a") otro valor almacenado en otra parte de la memoria del ordenador utilizando su dirección.



Figura 245 Sonda recuperada

9.2.5 RESULTADO DE LOS OBJETIVOS

Tabla 17 OBJETIVOS

OBJETIVOS	CUMPLIDO	INCOMPLETO
1º REALIZAR UN LANZAMIENTO DE UN GLOBO SONDA	X	
2º RECUPERAR LA SONDA	X	
3º CAPTACIÓN DE IMÁGENES DURANTE TODO EL VUELO	X	
4º ESTUDIAR EL FUNCIONAMIENTO DE SIGFOX EN LA ESTRATOSFERA	X	
5º DETERMINAR ALTURA MÁXIMA DE LA RED GPRS/GSM	X	
6º REALIZAR ESTUDIO DE PROPAGACIÓN DE HF		X
7º ESTABLECER COMUNICACIÓN CONTINUA CON LA SONDA	X	
8º TOMA DE DATOS METEOROLÓGICOS		X

1º REALIZAR UN LANZAMIENTO DE UN GLOBO SONDA.

El objetivo ha sido llevado a cabo satisfactoriamente, además se ha llevado a cabo todo el trabajo relacionado con el desarrollo de la electrónica, la simulación, la propulsión, recuperación, etc...

2º RECUPERAR LA SONDA.

La sonda se pudo recuperar completa y en buenas condiciones en menos de 15 minutos después del aterrizaje.

3º CAPTACIÓN DE IMÁGENES DURANTE EL VUELO.

Todo el vuelo fue filmado y además también se pudo obtener audio.

4º ESTUDIO DE COBERTURA SIGFOX EN LA ESTRATOSFERA.

Se pudo obtener cobertura durante todo el vuelo, tanto de subida como de bajada.

Se pudo establecer conexión a más de 30 (Km), según el fabricante la distancia máxima es de 1.000 (km) con visión directa de un repetidor y el alcance en zonas rurales esta entre los 30 y los 50 (Km).

Independientemente de los datos del fabricante, se pudo comprobar el buen funcionamiento de la red, además de la buena calidad de la comunicación, esto junto con la facilidad de puesta en marcha de estos dispositivos, así como su bajo coste de mantenimiento, sin duda harán de este sistema uno de los estándares en el IoT, gracias a este tipo de dispositivos la red 0G sigue ampliándose y ofreciendo mayor cobertura.

5º DETERMINAR ALTURA MÁXIMA DE FUNCIONAMIENTO DE LA RED GSM/GPRS:

Durante el ascenso de la sonda se pudo determinar que la altura máxima a la que el GPS *tracker* pudo enviar información, fue de unos 2.900 (m).

Es posible que este dato sea mayor en algunas zonas, puesto que, al estar justo encima de un repetidor, puede que la sonda entrase en una zona ciega de la antena y por ello se perdió antes la comunicación.

6º REALIZAR UN ESTUDIO SOBRE PROPAGACION DE HF:

No se pudo recuperar ninguna trama del sistema WSPR, por lo que no se pudo cumplir este objetivo.

7º ESTABLECER COMUNICACIÓN CONTÍNUA CON LA SONDA:

Este objetivo se cumplió a la perfección, en el único momento que no funciono el sistema SigFox fue durante los primeros tres minutos del lanzamiento, pero si que funcionó el sistema de GPS *tracker*.

8º TOMA DE DATOS METEOROLÓGICOS:

Al fallar el autómata con los sensores, fue imposible cumplir este objetivo.

9.3 POSIBLES MEJORAS

En este último apartado, expongo posibles mejoras que se podrían añadir a los subsistemas utilizados, también expongo las que podrían ser las futuras misiones de la serie METSAT y otras posibles variantes.

9.3.1 MEJORAS DE SUBSISTEMAS

En este punto se tratan cuatro posibles mejoras en los subsistemas utilizados, con la intención de mejorar los resultados y de poder realizar vuelos más exigentes y duraderos.

9.3.1.1 SISTEMA AUTOMÁTICO DE REENTRADA

Debido a la problemática a la hora de disponer el paracaídas y a la falta de precisión a la hora de determinar cuál será el punto de explosión del globo, se podría intentar desarrollar un sistema que desprenda la carga de pago del globo, de esta forma se podría controlar cuando se debe soltar, ya sea en función del tiempo de vuelo, de la altura, de la temperatura o incluso podríamos controlarlo a distancia en tiempo real mediante SigFox.

Una vez que el *Payload* comience su descenso, se podría programar la apertura del paracaídas, de forma que este no se pueda liar con ningún otro elemento.

Para llevar a cabo esta mejora, primero sería necesario desarrollar un método para separar el globo de la sonda, para ello se podría usar un electroimán o una resistencia que al dejar pasar una cierta corriente queme la línea de cabo.

Para desplegar el paracaídas se podría utilizar una pequeña estructura cilíndrica con una cara abierta, en el fondo, se introduce un muelle y por encima del muelle una pequeña estructura que sirve como base, tanto la base como el fondo del cilindro deben disponer de un agujero para poder pasar la línea de cabo que une el paracaídas con la sonda.

Una vez dispuesto el sistema, se debería meter el paracaídas bien plegado en el cilindro y presionar hacia el fondo el paracaídas para que ceda el muelle, luego se debe taponar el cilindro con un tapón adaptado, también impreso en 3D.

Con un pequeño servo se mantendrá la tapa encajada en el cilindro, pero cuando el servo cambie de posición, entonces el muelle empujará el tapón hacia el exterior y cuando este salga disparado, empujará la plataforma sobre la que se encuentra el paracaídas y este saldrá del cilindro.

Una vez fuera, la línea que une el paracaídas con la sonda cogerá tensión y el paracaídas se abrirá.

9.3.1.2 INTERFAZ SATELITARIA EN SUBSISTEMA DE COMUNICACIONES

Para mejorar el subsistema de comunicaciones, se podría añadir el módulo de comunicaciones Iridium de forma que sustituiría al módulo WSPR.

Con este sistema podríamos llenar el globo con el helio justo, de forma que el vuelo sea de muy larga distancia y aun así no perder la comunicación.



Figura 246 Simulación de medio alcance

Por ejemplo, se podría realizar un vuelo similar a este y no perder la comunicación en ningún momento, cuando se pierda la comunicación GSM, se comienza a utilizar SigFox y una vez estemos sobrevolando el mar, se puede usar Iridium.

Este sistema podría dar unas capacidades espectaculares tanto en el seguimiento del vuelo como en la recuperación de la sonda.

9.3.1.3 PAYLOAD BOX 3D PRINTED

Con la intención de aprovechar mejor el espacio, se podría realizar un *payload box* customizado para nuestras necesidades, o incluso se podría crear un diseño para que encaje dentro de la esfera de poliestireno y que permita fijar los componentes de forma más ordenada, aprovechando mejor el espacio y de forma que se pueda mejorar la transmisión del calor.

Para llevar a cabo esta mejora, sería necesario medir cada uno de los componentes y crear planos de estos, para poder desarrollar la pieza de forma que solo se tenga que encajar el componente electrónico y atornillar.

9.3.1.4 AUTÓMATA INDUSTRIAL

Para evitar problemas derivados de las condiciones atmosféricas, se podría utilizar un autómata preparado para condiciones industriales, además con una buena memoria para poder almacenar y ejecutar un código con comprobación de errores que evite la pérdida de información.

Además, un sistema como este podría utilizar un software como FreeRTOS para el control de procesos, de forma que podría conectarse con el resto de los dispositivos para comprobar el funcionamiento de estos y en caso de que sea necesario poder determinar actuaciones, de forma que el resto de los dispositivos actúen como esclavos.



Figura 247 Arduino Portenta H7

De esta manera por ejemplo se podría controlar la batería disponible para cada dispositivo y si fuera necesario se podría elegir si apagar un equipo para utilizar su batería en otro más importante, también se podría controlar de forma eficaz un sistema de desplegado automático de paracaídas, un sistema de liberación de la sonda, etc...

Y todo esto sin dejar de tomar valores de los sensores y almacenarlos en una memoria.

Para usar FreeRTOS, deberemos descargar la librería correspondiente desde el administrador de librerías y posteriormente en el código, deberemos hacer uso de esta.

Para programar, se utiliza el lenguaje de Arduino, pero la estructura y el paradigma es totalmente diferente, en caso de disponer de doble núcleo, se permite además la realización de dos tareas simultáneas.

9.3.2 FUTURAS MISIONES

Finalmente, en este apartado, se exponen las diferentes misiones que se podrían realizar una vez que se ha probado un sistema eficaz de recuperación del *payload*.

En concreto, expongo cuatro posibles misiones relativamente factibles.

9.3.2.1 METSAT-2

Esta misión supondría una mejora del METSAT-1, la idea sería utilizar como subsistema de comunicación solamente SigFox con un GPS de gran altura y un barómetro.

Respecto al subsistema de video, solo se utilizaría una cámara.

Se mejoraría el subsistema de computadora, de forma que no de problemas y se pueda recuperar la información.

Con estas modificaciones, se ahorrarían aproximadamente 368 (g) que podrían ser utilizados para añadir otros dispositivos nuevos que se quieran probar, añadir

sensores nuevos o simplemente para ahorrar dinero en helio y en la compra del globo.

Como propuesta se podría intentar utilizar LoRa para comprobar su funcionamiento y hacer una comparación entre LoRa y SigFox.

También se podría realizar un vuelo nocturno con la intención de filmar el amanecer.

Para conseguir estas imágenes, vamos a realizar un cálculo para determinar la hora de lanzamiento del METSAT-1, con los datos de este.

Para ello tendremos en cuenta que alcanzó unos 29.000 (m) y la velocidad de ascensión media fue de aproximadamente 6,4 (m/s).

El día 28 de febrero de 2020, el orto se produjo a las 07:50 en Orgaz.

Teniendo en cuenta que el radio de la tierra es aproximadamente de 6.378 (Km).

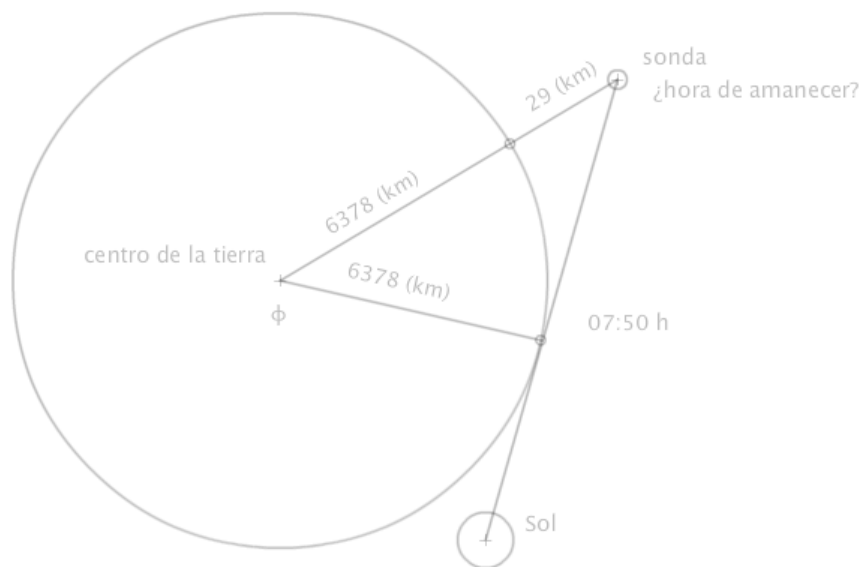


Figura 248 Cálculo amanecer_1

Si vemos la tierra en proyección desde el polo norte y cortamos con un plano por la latitud donde se realiza el lanzamiento, veremos un triángulo como este, en el que se representa el punto en el que el sol comienza a verse a ras del horizonte a una determinada altura.

Ahora deberemos de resolver el triángulo para obtener en ángulo ϕ , para ello:

$$\cos \theta = \frac{\text{lado contiguo}}{\text{hipotenusa}} = \frac{6.378}{6378 + 29} = 0,9954$$

$$\theta = \arccos 0,9954 = 5,497^\circ$$

Teniendo en cuenta que la latitud de Orgaz es de $39^\circ 40'$, podemos realizar los siguientes cálculos:

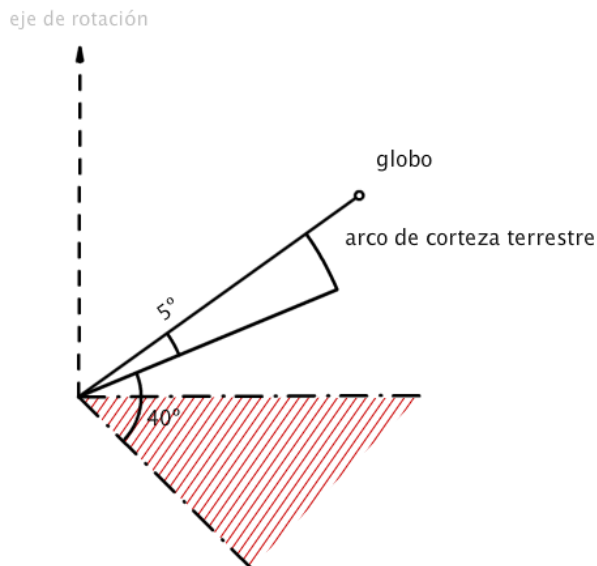


Figura 249 Cálculo amanecer_2

otro extremo del arco se encuentra el segmento que representa el radio terrestre y la altura del globo sobre la tierra. Este dibujo es la representación tridimensional del dibujo anterior, como podemos ver el arco de circunferencia dibujado desde el centro de la tierra es de 5° aproximadamente.

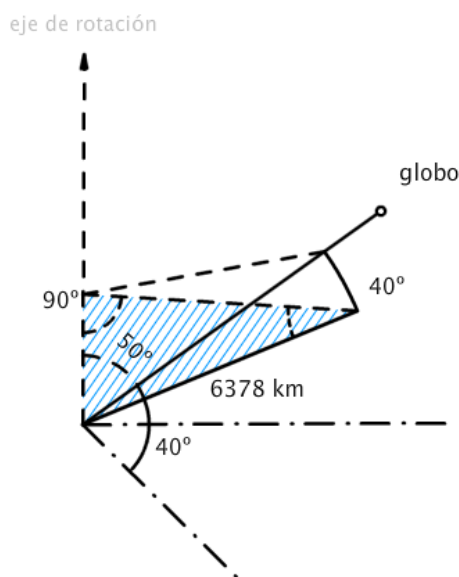


Figura 250 Cálculo amanecer_3

Teniendo en cuenta que la latitud de Orgaz es de $39^\circ 40'$, podemos aproximararlo a 40° .

Ahora en el dibujo de la izquierda de ha representado como rojo el plano del ecuador, desde el podemos realizar un ángulo de 40° con el que situaríamos la latitud del punto de lanzamiento, también podemos representar un arco de corteza terrestre cuyos puntos están todos en el mismo paralelo, al

Ahora debemos trazar un plano que corte en perpendicular al eje de rotación y que pase por el paralelo correspondiente a la latitud del punto de lanzamiento.

De esta forma nos aparecerá un nuevo triángulo, pero antes de poder resolverlo, necesitamos más datos, por ello comenzamos resolviendo el triángulo azul.

Como sabemos que la latitud es de 40° , sabemos restan 50° para llegar a los 90° , por ello podemos hallar la Colatitud, otro ángulo es

de 90° puesto que el plano trazado es perpendicular al eje de rotación, por lo tanto, para hallar el tercer ángulo, solo tenemos que restar a 180° la suma de los ángulos que ya conocemos.

Ahora que conocemos los ángulos y uno de los lados, es hora de determinar cuál es el lado opuesto al ángulo de 50°, de esta forma, resolveremos dos lados del triángulo perpendicular al eje de rotación.

Para hallar este lado, se tiene que realizar la siguiente operación:

$$x = \text{sen}(50) * 6378$$

$$x = 4885.8 \text{ (Km)}$$

Teniendo en cuenta que el triángulo perpendicular al eje es isósceles, ya tenemos calculados dos lados de este.

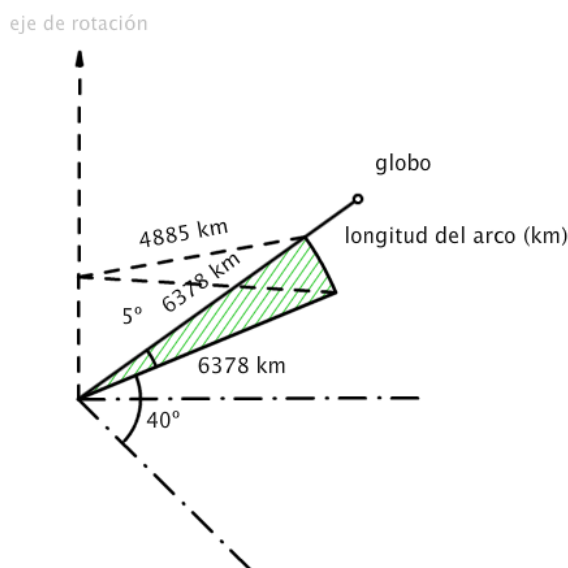


Figura 251 Cálculo amanecer_4

Para hallar el tercer lado del triángulo perpendicular al eje, necesitamos resolver otro triángulo, este corresponde con el triángulo verde, del cual conocemos dos lados y el ángulo comprendido entre ellos, para hallar el tercer lado, se debe realizar la siguiente operación:

$$180^\circ - 5^\circ = 175^\circ$$

$$\frac{175^\circ}{2} = 87,5^\circ$$

De esta forma terminamos de hallar los otros dos ángulos del triángulo isósceles, ahora, mediante esta ecuación, obtenemos la longitud en (km) del arco de circunferencia.

$$\text{longitud de arco} = 6378 \frac{\text{sen}(5)}{\text{sen}(87,5)} = 556,4 \text{ (km)}$$

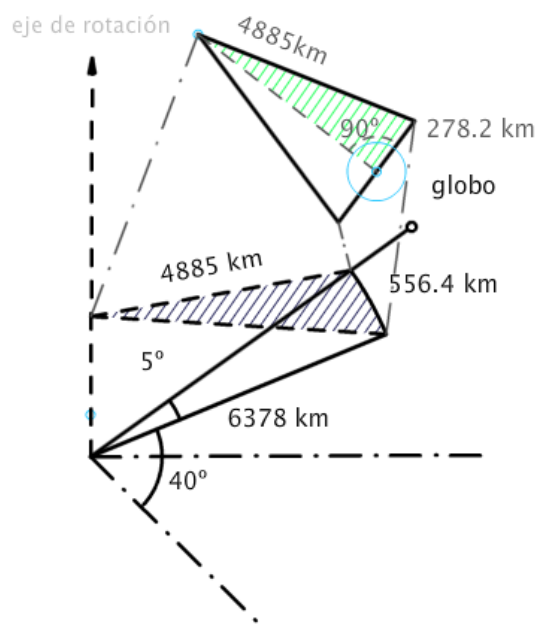


Figura 252 Cálculo amanecer_5

Por último, ya podemos resolver el triángulo perpendicular al eje, de forma que lo dividimos en dos triángulos rectángulos y mediante la siguiente operación hallamos el ángulo del arco de circunferencia:

$$\arccos\left(\frac{278,2}{4885}\right) = 3,26^\circ$$

Pero como eso es solo la mitad, el ángulo sería el doble, es decir: $6,52^\circ$

Es decir, cuando nuestro globo sonda alcance los 29.000 metros de altura, la

visión que tendrá de la bóveda celeste, abarcará unos $6,5^\circ$ más que si estuviera en la superficie terrestre, por lo que desde esa altura el sol se verá salir por el horizonte $6,5^\circ$ antes, o lo que es lo mismo, 26 minutos antes, en el caso de querer determinar el ocaso estratosférico, hemos de tener en cuenta que será 26 minutos más tarde para este caso.

Para entender esta conversión, es necesario suponer que la tierra gira unos 360° al día, por lo que:

$$\frac{360^\circ}{24 \text{ horas}} = 15^\circ/h$$

$$\frac{6,5^\circ * 60 \text{ min}}{15^\circ} = 26 \text{ minutos}$$

Es decir, si queremos ver el amanecer estratosférico, el globo deberá alcanzar los 29.000 (m) 26 minutos antes del amanecer en el punto de lanzamiento seleccionado.

Por lo que si en Orgaz amanece a las 07:50, el amanecer estratosférico se producirá a las 07:24.

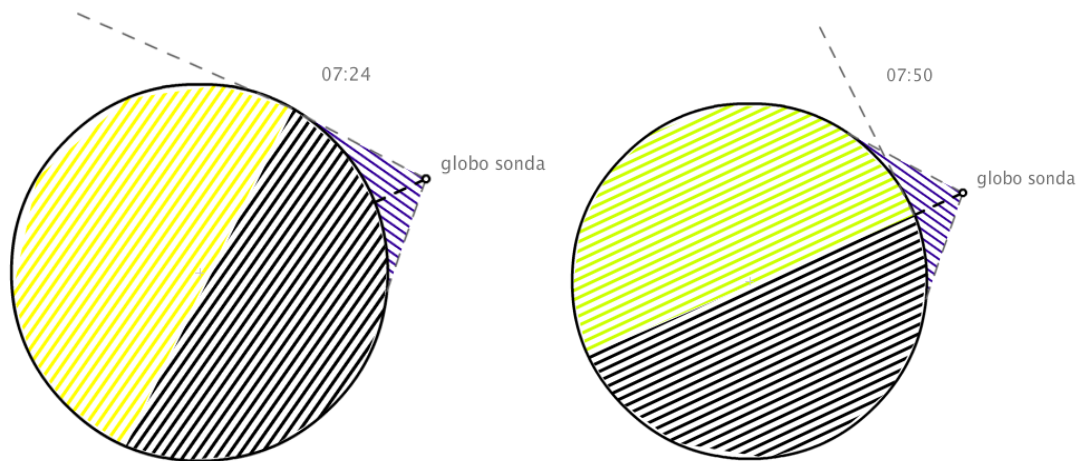


Figura 253 Cálculo amanecer_6

En la imagen podemos ver esto claramente, la zona amarilla es la iluminada por el sol, mientras que la oscura es la zona de noche, la zona azul representa la visión desde el globo sonda, como se ve en la imagen de la izquierda, a las 07:24, el ángulo de visión del globo, comenzará a ver el sol aparecer por el horizonte, pero aun será necesario que la tierra gire 6,5° más para que empiece a amanecer en el punto de lanzamiento.

Por lo tanto, para determinar la hora de lanzamiento, deberemos hacer el siguiente cálculo:

$$\frac{29000}{6,4} = 4531,25 \text{ (s)}$$

$$\frac{4531,25}{60} = 75,52 \text{ (min)}$$

Por lo tanto, el tiempo de ascensión hasta los 29.000, será de 76 minutos aproximadamente.

Por lo que el lanzamiento debe producirse 76 minutos antes de que amanezca a 29 (km), es decir, el lanzamiento debe producirse a las 06:08 de la mañana.

Sin embargo, hay que tener en cuenta que si el globo explota a 29.000 metros entonces nos perderemos las tomas estables de la cámara, por lo que a la hora de realizar los cálculos debe tomarse como objetivo una altura algo menor a la de explosión del globo, también es importante tener en cuenta que se debe dejar un margen de error, ya que se toman muchas aproximaciones.

9.3.2.2 METSAT-3

Este sería uno de los proyectos más ambiciosos, consistiría en un globo de 3 (Kg), inflado con 4,2 (m³) de helio.

Con un *payload* de 1.300 (g), este podría realizar un vuelo similar a este:

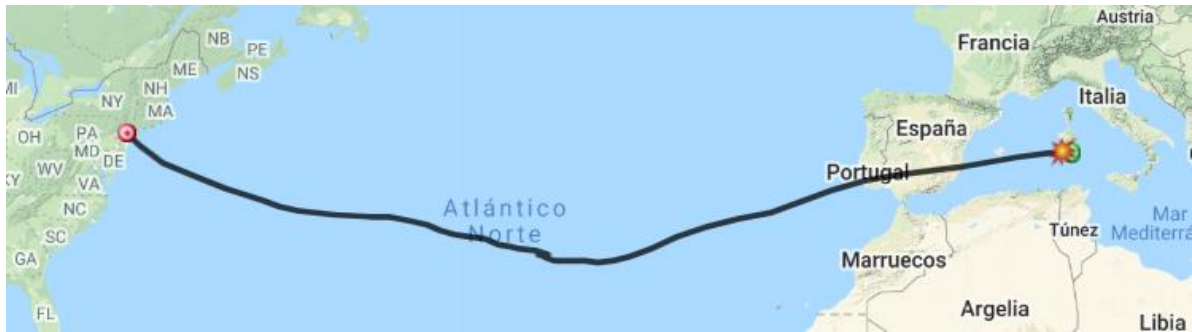


Figura 254 Predicción largo alcance

Podría realizar un vuelo desde Nueva York a Cerdeña en 4 días y medio.

La sonda estaría equipada con un sistema Iridium y con un Arduino con cámara, la idea sería ir captando imágenes cada cierto número de horas y almacenarlas, el resto de la carga de pago iría destinado a baterías y a un sistema de carga de baterías a través de paneles solares.

Un vuelo tan largo sería un problema, pero se podría calcular un vuelo desde España a Italia, por ejemplo.

Respecto a la recuperación, sería una gran idea contactar con alguien interesado en el proyecto para que lo recupere y lo envíe por correo.

9.3.2.3 METPSL-1

Este sería el primer lanzamiento de una nueva serie, el significado de las siglas es Mantenimiento Electrónico *Pico Sat Launcher*.

La misión consistiría en enviar una pequeña sonda con solo dos cámaras de video y un sistema de recuperación SigFox, entre la sonda y el globo, iría dispuesto un cilindro en posición vertical, pero con un ligero ángulo, de forma que la proyección

de la base del cilindro no corte al globo, este cilindro estará hueco y en su interior llevará un pequeño cohete.

Este cohete posee un motor cohete de combustible sólido, también incorpora un espacio para *payload* y un sistema de apertura automática de paracaídas.

La idea es controlar la ignición del motor cohete a través del sistema SigFox de la sonda, de forma que, a una determinada altura, antes de que el globo explote, el cohete despegue y vuele hacia arriba, una vez que el cohete se queda sin combustible, la carga de pago sale despedida del cohete y empieza su caída controlada mediante el paracaídas.

La idea es utilizar otro sistema SigFox como *payload* en el cohete, de forma que se pueda recuperar este.

Además, también se recuperará la sonda estratosférica la cual contiene las imágenes del vuelo y las de la cámara que irá grabando el globo y el despegue del cohete.

Un dato importante a tener en cuenta es que los cohetes gastan la mayor parte del combustible en la troposfera, sin embargo, a partir de la estratosfera es donde empiezan a trabajar de forma eficiente.

Por ello, si un cohete en la superficie de la tierra puede alcanzar por ejemplo unos 1.000 metros, si se lanzase desde la estratosfera, podría aumentar su eficiencia y elevarse a bastante más altura, por ejemplo, pongamos que podría elevarse hasta 2.000 metros.

Estos cohetes son muy utilizados en modelismo espacial, existen varios tipos de cohetes, en función del tipo de motor utilizado, el material de fabricación o incluso en función de las etapas que posea.



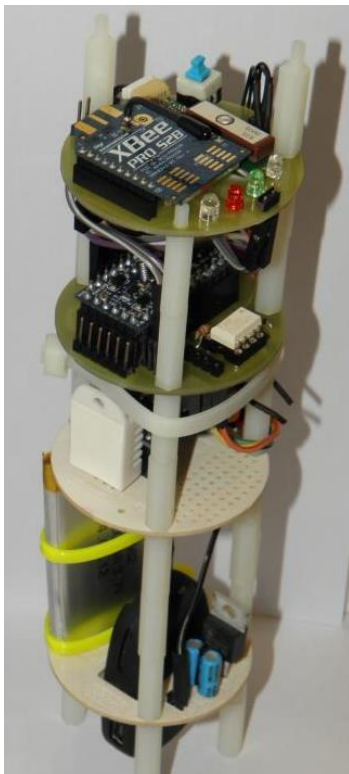
Figura 255 Payload Rocket

Por ejemplo, un cohete de 152 (cm) de largo, 5 (cm) de diámetro, 480 (g), con paracaídas, Sistema SigFox y 4 motores que pueden elevar el cohete a casi 2.000 metros, solo cuesta 131€

Este modelo podría volar hasta con una cámara en el interior.

Sin embargo, también se puede usar un modelo más económico, por 72€ podríamos disponer de un cohete de 130 (g), 51 (cm) de largo, 3,4 (cm) de diámetro, paracaídas, sistema SigFox y un cohete capaz de impulsar este, a algo más de 300 (m) de altura.

Una vez probado este sistema, se podría mejorar para hacer volar un pequeño cohete multietapa que podría alcanzar más de 4.000 metros y al soltar la carga de pago, podría simular la puesta en órbita de un pico satélite.



Para profundizar un poco más sobre la electrónica del cohete, es necesario introducir el concepto de CANSAT, este viene a ser un estándar para el *payload* de un cohete de las características anteriores.

Consiste en implementar un sistema electrónico que se pueda introducir dentro de una lata de refrescos, para ello, se pueden diseñar y fabricar varias placas de forma que al montarse unas sobre otras, obtienen una forma que permite que se pueda introducir dentro de esta lata.

Una vez que este salga despedido del cohete, funcionará como un satélite hasta que finalmente tome tierra, para ello se debe integrar en la potencia, computadora y comunicaciones en un solo bloque.

9.3.2.4 METGLIDER-1

Este sería el primer lanzamiento de una nueva serie, el significado de las siglas es Mantenimiento Electrónico Planeador.

Consistiría en elevar hasta una altura aproximada de 10.000 (m) un planeador equipado con un motor eléctrico de bajo consumo, un sistema de recuperación SigFox, cámara y ordenador de vuelo.

Este ordenador de vuelo estará programado para seguir coordenadas GPS, de forma que una vez que el planeador se suelte del globo, el motor se encenderá y el planeador se estabilizará para comenzar a seguir los *waypoints*, la idea es programar estos *waypoints* de forma que el planeador siempre vuele a favor del viento, de esta forma podemos ir programando que descienda y que aterrice en una zona controlada.

El planeador se soltará del globo quemando el cabo mediante un cable incandescente.

Por ejemplo, si se sigue la predicción del siguiente vuelo, se podría programar el planeador, para que cuando este se suelte, siga una trayectoria predefinida siguiendo los vientos predominantes, de forma que podría situarse sobre Corvera y empezar a describir círculos mientras desciende hasta un punto establecido como punto de aterrizaje, en el cual estaríamos esperando a recuperar el planeador.



Figura 256 Predicción glider

10. DESARROLLO DEL PROYECTO (GANTT)

A continuación, muestro el diagrama de Gantt utilizado a lo largo del proyecto, este se pudo cumplir satisfactoriamente.

En proyecto está dividido en cuatro fases, la primera de ellas tiene una duración de 29 días y corresponde con la elección de la electrónica a utilizar.

La segunda fase, tiene una duración de 40 días y es la parte en la que se debe crear el *payload box*, se debe montar la electrónica diseñar el sistema de paracaídas y programar los autómatas.

La tercera fase tiene una duración de 10 días, en los cuales en función del paracaídas y del globo, se deben realizar las simulaciones y los cálculos relacionados con el peso total del conjunto, el helio a utilizar, etc...

Por último, está la cuarta fase, la cual tiene una duración de 65 días, en los cuales se tiene que realizar la financiación, buscar seguro, realizar peticiones a ENAIRE, solucionar la logística, etc...

En total el proyecto se desarrolló en tan solo 102 días, unos 3 meses.

Debido al poco tiempo disponible y la complejidad del proyecto, seguir el diagrama de Gantt y gestionar el tiempo se convirtió en la clave para conseguir el objetivo del proyecto.

Respecto a la primera fase, las tareas están encaminadas a la elección de los dispositivos electrónicos adecuados, para ello se crearon listas de productos que cumplieran ciertos requisitos, finalmente se seleccionaron los mejores en una relación de calidad precio.

Finalmente, se realiza un punto de control para exponer el trabajo realizado hasta la fecha e indicar cual sería la siguiente fase.

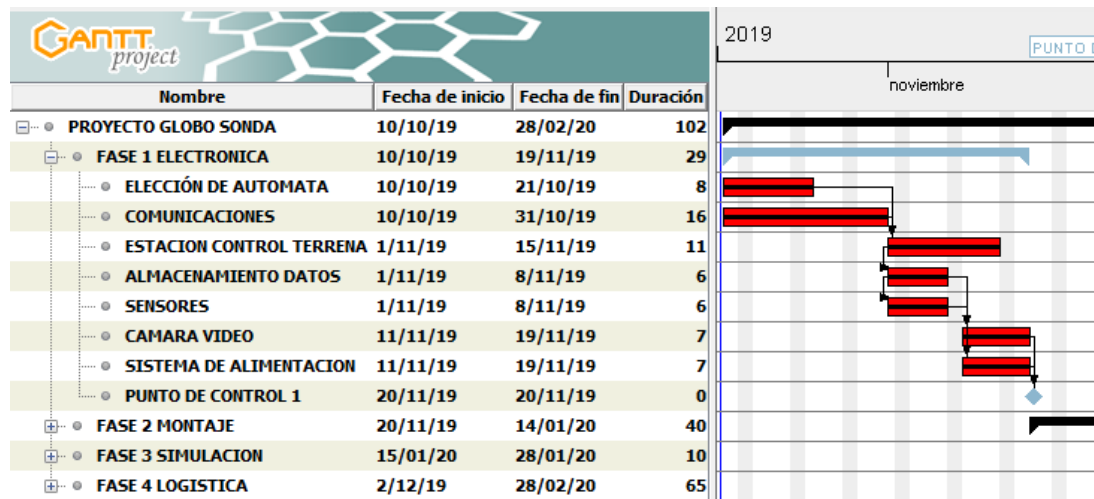


Figura 257 Primera fase

Una vez concluido el punto de control, se inició la segunda fase del proyecto, esta se divide en un total de dos tareas y de dos puntos de control.

Estas dos tareas tienen como objetivo desarrollar los programas de los autómatas, depurar el código y realizar pruebas con los equipos.

Por otro lado, se debe crear un *payload box* en el que meter toda la electrónica y a este se le debe añadir un sistema para unirlo tanto al paracaídas como al globo que lo elevará.

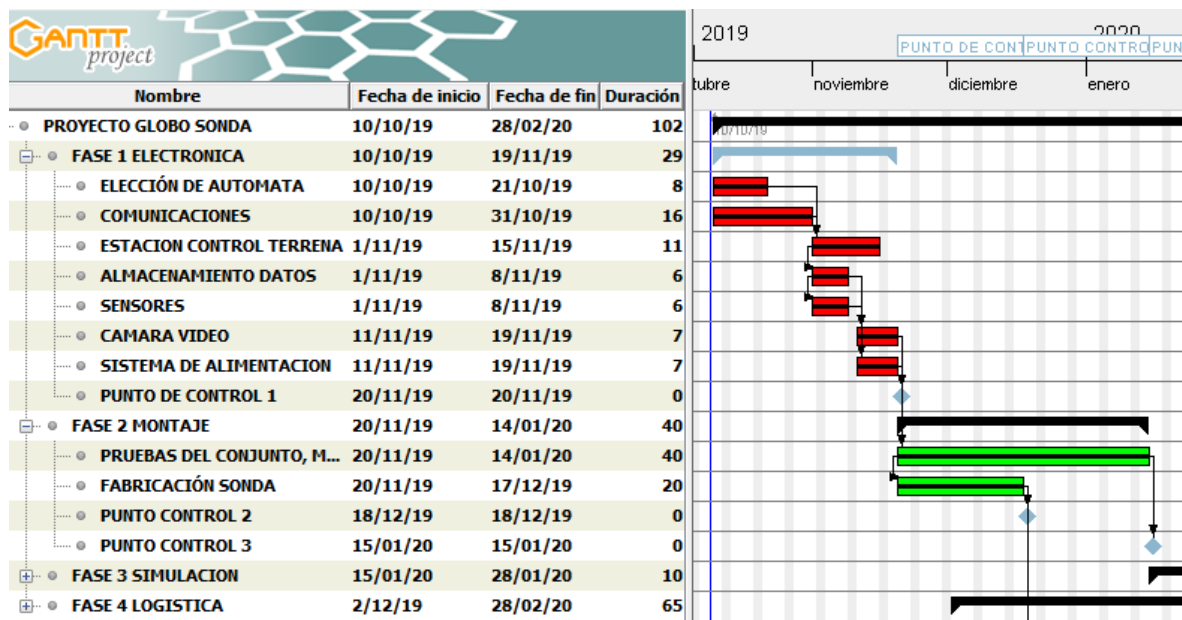


Figura 258 Segunda fase

Los dos puntos de control tienen como objetivo la exposición de los resultados una vez terminada cada una de las tareas.

El primero de ellos tiene como objetivo también la introducción del trabajo a realizar en la fase de logística, la cual comienza junto con la fase dos, se compaginará con la fase tres y una vez terminada la fase tres, esta fase se terminará de desarrollar con la recuperación de la sonda.

Por otro lado, el segundo punto de control tiene como objetivo explicar las tareas que se realizarán en la fase de simulación.

En la fase tres, podemos encontrar las siguientes cuatro tareas y un punto de control.

Estas están encaminadas a la elección de un globo y de un paracaídas adecuado, terminar de montar el sistema de paracaídas y preparar el sistema de llenado del globo.

Por otro lado, se deben realizar los cálculos correspondientes al peso de todos los componentes, cálculos de helio necesario, etc...

Con todos estos datos se pueden empezar a realizar simulaciones que nos ayudarán a elegir una zona de aterrizaje.

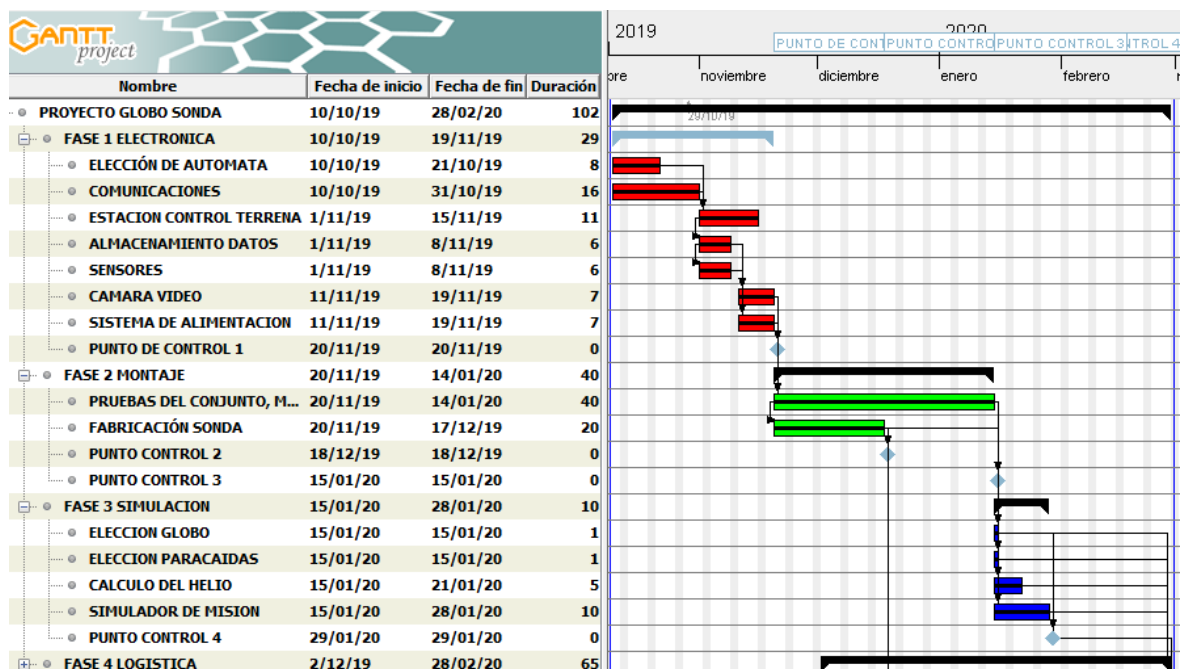


Figura 259 Tercera fase

Finalmente, el punto de control final tiene como objetivo, explicar todo el trabajo desarrollado hasta ese momento y en caso de que se dé el visto bueno, proceder a terminar la fase cuatro para poder realizar el lanzamiento.

Por último, la fase cuatro consta de tres tareas y de dos puntos de control.

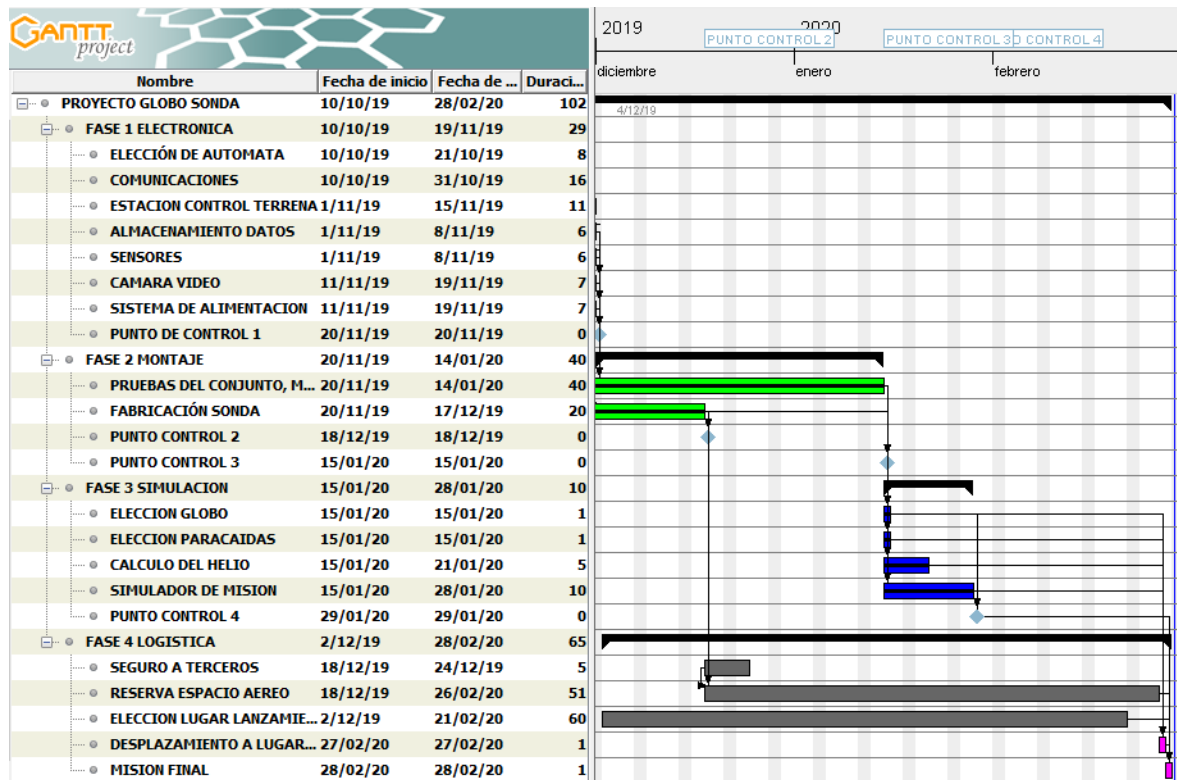


Figura 260 Cuarta fase

Las tareas consisten en la elección de un seguro, elección del lugar de lanzamiento y la reserva del espacio aéreo.

Los puntos de control coinciden con el desplazamiento al lugar de lanzamiento y la recuperación de la sonda.

En resumen, las tareas y la duración de estas son las siguientes:

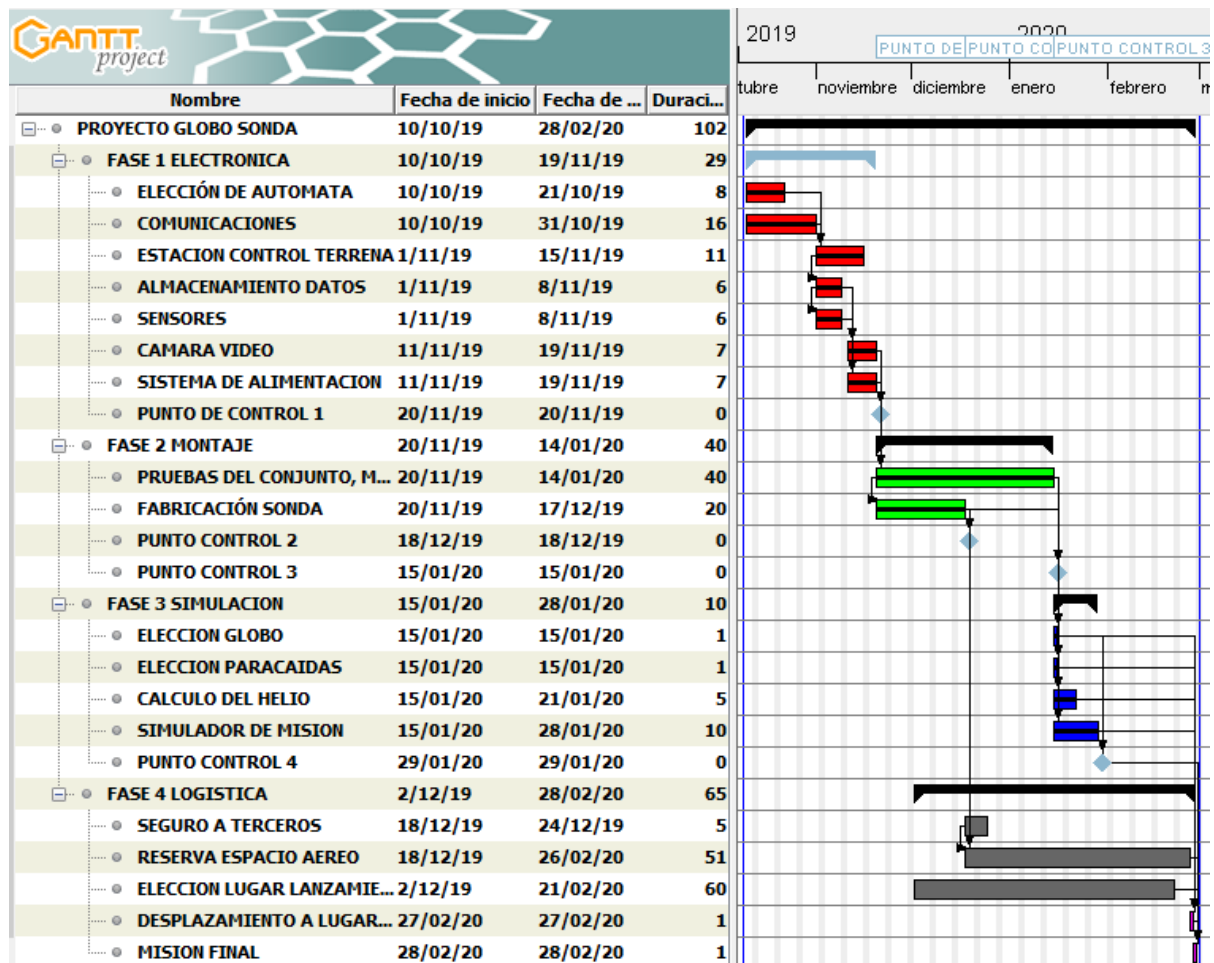


Figura 261 Gantt

11. PRESUPUESTO

Una vez concluido el proyecto, se puede hacer una lista detallada con todos los materiales utilizados y dar un presupuesto por si se tiene que volver a realizar un proyecto similar.

A continuación, expongo dos presupuestos, el primero de ellos está enfocado a realizar un proyecto idéntico al realizado y en las mismas condiciones, mientras que el segundo, es un presupuesto para un globo sonda de bajo presupuesto, que incluye un sistema de recuperación fiable y al cual se le puede añadir la carga de pago que sea necesaria.

Tabla 18 PRESUPUESTO

CANTIDAD	DESCRIPCIÓN	PRECIO UNITARIO €	IMPORTE €
1	GLOBO SB-1000	100.33	100.33
1	PARACAÍDAS 30"	18.16	18.16
1	ROLLO NYLON 50 (m)	5.52	5.52
1	QUITAVUELTAS	1.58	1.58
1	BOLSA BRIDAS PEQUEÑAS	3.07	3.07
1	GOPRO HERO	133.51	133.51
1	SJCAM	16.58	16.58
3	MICROSD 32 (GB)	4.73	14.19
1	ARDUINO MEGA MINI	4.3	4.3
1	MKRFOX 1200	27.65	27.65
1	ANTENA 868 (MHZ)	0.71	0.71
1	TKSTAR	39.5	39.5
1	SIM	3.91	3.91
1	RECARGA SIM	23.7	23.7
1	PORTABATERIAS 2 AA	0.6	0.6
2	PORTABATERIAS 3 AA	0.76	1.52
1	PORTABATERIAS 4 AA	1.14	1.14
11	BATERIA LITIO AA	1.95	21.45
1	DHT11	1.57	1.57
1	BMP180	2.96	2.96
1	MICROSD SHIELD	1.78	1.78
1	DS18B20	1.88	1.88
2	ANTENA GPS	6.75	13.5
1	WSPR	32.89	32.89
1	FILTRO PASO BAJO WSPR	1.82	1.82
1	ANTENA WSPR	1.58	1.58
3	FUENTES DE ALIMENTACIÓN	1.1	3.3
1	DIODO LED BICOLOR	0.45	0.45
1	INTERRUPTOR	0.79	0.79
1	MPU6050	2.26	2.26

1	JUMPERS VARIOS	4.89	4.89
1	SEGURO	314.42	314.42
4.6	METRO CUBICO HELIO	33.97	156.262
	ESFERA POLIESTIRENO 25		
1	(CM)	8.26	8.26
1	CABLE USB	1.58	1.58
1	CALENTADOR	0.45	0.45
1	ROLLO CINTA AMERICANA	5.28	5.28
1	TERMORETRACTIL	0.63	0.63
1	ESPUMA POLIURETANO	3.86	3.86
1	PAPEL FILM	1.01	1.01
1	PISTOLA SILICONA	3.78	3.78
1	TUBOS SILICONA	5.17	5.17
1	CINTA AISLANTE	0.7	0.7
1	GOMA ELASTICA	1.18	1.18
1	PINTURA NEGRA	4.65	4.65
1	BROCHA	1.09	1.09
1	BUZZER	2.33	2.33
		SUBTOTAL:	997.74 €
		IVA:	209.52 €
		TOTAL:	1207.26 €

El presupuesto es de unos 1207,26€ con el IVA incluido, aunque este presupuesto, está haciendo una suma del precio de todo el material utilizado, sin embargo, muchos de los materiales ya estaban a nuestra disposición y fueron reutilizados.

El seguro, es el artículo más caro de todos, nos da una gran cobertura, sin embargo, no es imprescindible, por lo que, si el globo se lanza de forma particular independientemente de una institución pública, en principio no es necesaria la contratación de este, aunque si es recomendable.

Tabla 19 PRESUPUESTO BAJO COSTE

CANTIDAD	DESCRIPCIÓN	PRECIO UNITARIO €	IMPORTE €
1	GLOBO SB-1000	100.33	100.33
1	PARACAÍDAS 30"	18.16	18.16
1	ROLLO NYLON 50 (m)	5.52	5.52
1	QUITAVUELTAS	1.58	1.58
1	MICROSD 32 (GB)	4.73	4.73
1	GOPRO HERO	133.51	133.51
1	MKRFOX 1200	27.65	27.65
1	ANTENA 868 (MHZ)	0.71	0.71
1	TKSTAR	39.5	39.5
1	SIM	3.91	3.91
1	RECARGA SIM	23.7	23.7

2	PORTABATERIAS 3 AA	0.76	1.52
1	PORTABATERIAS 4 AA	1.14	1.14
9	BATERIA LITIO AA	1.95	17.55
1	ANTENA GPS	6.75	6.75
2	FUENTES DE ALIMENTACIÓN	1.1	2.2
1	JUMPERS VARIOS	4.89	4.89
4	METRO CUBICO HELIO	33.97	135.88
1	ESFERA POLIESTIRENO 25 (CM)	8.26	8.26
SUBTOTAL:			537.49 €
IVA:			112.87 €
TOTAL:			650.36 €

En este último presupuesto, se han omitido los materiales comunes como bridas, cinta americana, etc... Ya que estos materiales suelen estar a disposición de cualquiera, puesto que son de uso muy común en otras aplicaciones.

También se ha eliminado el seguro y una de las cámaras, se ha eliminado también el sistema WSPR y el autómata, de forma que solo se dejan dos sistemas de posicionamiento y se deja el resto de la carga de pago a gusto del consumidor.

12.CONDICIONES TÉCNICAS

En este apartado se exponen los principales elementos del globo sonda y se especifican unos requisitos mínimos que deben cumplir estos en caso de querer emular un proyecto como este o alcanzar mejores resultados.

Tabla 20 Condiciones técnicas

ITEM	CARACTERÍSTICAS MÍNIMAS
GLOBO SONDA	Debe ser capaz de elevar como mínimo 100 (g) más del peso del conjunto a enviar.
PARACAÍDAS	Debe ser capaz de frenar la caída de un objeto que supere el peso del conjunto que vamos a lanzar.
QUITAVUELTAS	Debe ser capaz de soportar más de 30 (kg) de forma estática.
NYLON PARA LINEAS	Debe de ser como mínimo de 2,5 (mm) de diámetro y aguantar tracciones de más de 30 (Kg).

ANTENA SIGFOX	Debe tener una ganancia superior a 0,5 (dBi), además debe poseer una estructura que le dé una calificación de IP67, de forma que pueda ser utilizada en el exterior de la sonda, además debe ser una antena omnidireccional, también debe ser capaz de trabajar hasta los -40°C.
ANTENAS GPS	Si la antena va dispuesta en el exterior de la sonda, deberá estar clasificada como IP67 y ser capaz de soportar hasta -40°C, en caso de que la antena esté en el interior de la sonda, esta debe ser cerámica. Para cualquiera de los dos tipos, la altura a la que dejan de recibir señal debe de ser como mínimo 30.000 (m), además deben ser capaces de recibir señal de como mínimo tres constelaciones, incluyendo la constelación GPS y GLONASS. Un modelo propuesto es el GM-801 u-blox8.
TERMÓMETROS	Los termómetros del exterior de la sonda deberán presentar un encapsulado que les proteja de las condiciones meteorológicas, los del interior, pueden utilizar cualquier tipo de encapsulado. Si alguno de los termómetros internos se utiliza para medir cualquier cosa que no sea el ambiente del interior, deberá también presentar un encapsulado similar al de los termómetros exteriores y si presenta piezas metálicas, deberán ser cubiertas con termo retráctil. El rango de temperaturas por encima de cero debe alcanzar como mínimo los 45°C y por debajo de cero, deben alcanzar hasta los -50°C.
PILAS	Las baterías que utilizar serán siempre de litio, en concreto los modelos de pilas de litio no recargables, como mínimo deben disponer de una densidad de energía de 0,29 (wh/g).

	Además, estarán pensadas para soportar temperaturas de hasta -40°C y soportarán picos de corriente superiores en 50 (mA) a los picos máximos de consumo de sus cargas.
CÁMARAS	Las cámaras deben de ser capaces de filmar como mínimo en 1080, no deberán de presentar pantalla posterior, ni conectividad Wifi. Presentarán una batería que les permita realizar grabaciones ininterrumpidas en condiciones ambiente de hasta 2 horas.
TARJETAS MEMORIA	Las tarjetas de memoria siempre que sea posible deberán ser de tipo MicroSD, de clase 10 y como mínimo deben de almacenar hasta 32 (Gb). En caso de que un dispositivo no sea capaz de soportar 32 (Gb), se utilizará la tarjeta con la capacidad máxima que acepte.
AUTÓMATA	El autómata, como mínimo debe de disponer de los siguientes buses: SPI, I2C, 1WIRE. También deberá presentar como mínimo dos buses físicos seriales. Presentará como mínimo 6 pines analógicos, 15 pines digitales y pines de alimentación para 5 y 3,3 (V). Además, deberá poseer memoria EEPROM de como mínimo 4 (kb), una memoria SRAM como mínimo de 8 (kb) y una memoria Flash de 256 (kb). Será capaz de soportar temperaturas de hasta 40°C y de hasta -20°C.
GPS TRACKER	Debe ser un dispositivo capaz de tener conectividad 2G, 3G y 4G, siempre que sea posible, se utilizará un modelo programable, de forma que pueda enviar mensajes de forma automática.
SIM	Debe utilizarse una tarjeta SIM apta para 2G, 3G y 4G.

	Además, esta tarjeta deberá ser del operador que ofrezca la mejor cobertura en la zona del vuelo, especialmente en la zona del aterrizaje.
PORTA BATERÍAS	Deberán ser de plástico rígido, además deberán incorporar tapaderas para poder aislar fácilmente las pilas del resto de componentes, también deberán disponer de un botón de encendido y apagado.
DHTXX	En caso de utilizar un sensor de la familia DHT, este debe ser como mínimo el sensor DHT22 y deberá usarse siempre una resistencia <i>pull-up</i> .
BMPXXX	En caso de utilizarse un sensor barométrico, como mínimo se utilizará el modelo BMP280.
ANTENA WSPR	Esta antena deberá ser omnidireccional, será de la longitud de la onda en cuestión, deberá tener una impedancia de 50 ohm y se debe de evitar el uso de soldadura, en la parte interior de la sonda, el cable debe estar apantallado y debe presentar un conector SMA <i>mini</i> .
FUENTES ALIMENTACIÓN	A la hora de utilizar fuentes conmutadas, siempre que se pueda, se recomienda el uso de fuentes convencionales, dejando siempre margen de voltaje de forma que siempre trabajemos con los valores seguros de la fuente, además esta fuente deberá permitir picos de corriente superiores como mínimo a 200 (mA) del pico máximo de la carga. Solo en caso de tener muy limitado el peso, se utilizarán fuentes elevadoras. En caso de que las fuentes sean regulables, es aconsejable que posea ajuste fino mediante destornillador.
JUMPERS	Siempre que sea posible, se utilizará cable rígido de cobre para soldar y en los acabados se deberá utilizar termo retráctil, para los pines del autómatas también se

	aconseja usar este tipo de cable, si no es posible soldarlo, se introducirá la punta en el pin y se utilizará silicona caliente para evitar que este salga de su pin.
PAYLOAD BOX	El <i>payload box</i>, deberá ser de un material muy ligero, y en caso de utilizarse poliestireno, el grosor de las paredes será como mínimo de 3,5 (cm), siempre que sea posible se deberá de reducir el número de piezas al mínimo. El poliestireno deberá ser de calidad superior, de forma que sea muy denso y sea resistente.
RECUBRIMIENTO PAYLOAD BOX	Para cubrir la sonda, se recomienda el uso de pintura negra acrílica. En caso de no disponer de pintura negra, se recomienda el uso de cualquier color oscuro. En caso de no disponer de colores oscuros, entonces utilizar un color llamativo que ayude en la localización. Se desaconseja el uso de mantas térmicas, puesto que favorecen la aparición de condensación y pueden crear efecto de jaula de Faraday.
CALENTADOR	Como mínimo se utilizará un calentador que caliente por conducción, este deberá de funcionar como mínimo 12 horas. Siempre que sea posible deberá usarse un calentador químico.
BOTELLA DE HELIO	La botella deberá ser de tipo industrial, de forma que la boquilla esté estandarizada y se eviten problemas de mal funcionamiento.
SISTEMA DE LLENADO DE HELIO	Se debe utilizar un sistema de manguera, para unir la manguera a la boquilla de la botella, se deberá usar un sistema con abrazadera metálica, se debe evitar el uso de válvulas intermedias entre el globo y la botella.

13. BIBLIOGRAFÍA

- Sebastián Franchini y Oscar López García (2012): “Introducción a la ingeniería aeroespacial” 2ª Edición.
- Arduino, gama pro de autómatas, <https://store.arduino.cc/arduino-genuino/pro>, [9 de abril de 2020].
- SigFox, página oficial en español, <https://www.sigfox.es/>, [9 de abril de 2020].
- SigFox, base de datos, <https://backend.sigfox.com/auth/login>, [9 de abril de 2020].
- Programafacil, Arduino MKRFOX 1200 configuración y características, <https://programarfacil.com/blog/arduino-blog/arduino-mkrfox1200-sigfox-lpwan/>, [9 de abril de 2020].
- Github, tutoriales, <https://jorgecasas.github.io/2014/12/24/proyecto-sonda-ii>, [9 de abril de 2020].
- Hackster, GPS TRACKER en arquitectura MKR de Arduino, <https://www.hackster.io/jgallar/gps-tracker-with-arduino-mkr-fox-1200-104012>, [9 de abril de 2020].
- Sigfox, Suscripciones, <https://buy.sigfox.com/>, [9 de abril de 2020].
- Github, Comunidad, <https://github.com/>, [9 de abril de 2020].
- Thingspeak, METSAT-1, <https://thingspeak.com/channels/974006>, [9 de abril de 2020].
- TKSTAR, Store, <https://www.tk-star.com/>, [9 de abril de 2020].
- Movistar, cobertura, <http://www.movistar.es/particulares/coberturas/movil/4G/>, [9 de abril de 2020].
- TKSTAR, seguimiento, <http://www.mytkstar.net/>, [9 de abril de 2020].
- URE-VALLADOLID, WSPR, <http://www.ea1urv.es/web/wspr/>, [9 de abril de 2020].
- APRSINFO, WSPR reportes, <http://wspr.aprsinfo.com/>, [9 de abril de 2020].
- WSPRNET, servidor, <http://wsprnet.org/>, [9 de abril de 2020].
- Princeton, WSPR, <http://physics.princeton.edu/pulsar/K1JT/wspr.html>, [9 de abril de 2020].
- Joe Taylor (2010): “Guía de usuario de WSPR 2.1”

- Zachtek, store, <https://www.zachtek.com/1017>, [9 de abril de 2020].
- WSPR, grid locator, http://levinecentral.com/ham/grid_square.php, [9 de abril de 2020].
- Wikipedia, locator, https://en.wikipedia.org/wiki/Maidenhead_Locator_System, [9 de abril de 2020].
- Lorawan, LoRa, <https://lorawan.es/>, [9 de abril de 2020].
- IoT, comparativa, <https://alfaiot.com/blog/ultimas-noticias-2/post/nb-iot-vs-lora-vs-sigfox-10>, [9 de abril de 2020].
- Iridium, store, <https://www.iridium.com/products/rock-seven-rockblock-mk2/>, [9 de abril de 2020].
- Pi in the sky, comunidad, <http://www.pi-in-the-sky.com/>, [9 de abril de 2020].
- Arduino Mega Mini, pinout y esquema, <https://robotdyn.com/pub/media/0G-00005642==MEGA-PROMINI-ATmega2560/DOCS/PINOUT==0G-00005642==MEGA-PROMINI-ATmega2560.pdf>, [9 de abril de 2020].
- Luis Llamas, I2C, <https://www.luisllamas.es/arduino-i2c/>, [9 de abril de 2020].
- Luis Llamas, SPI, <https://www.luisllamas.es/arduino-spi/>, [9 de abril de 2020].
- GNS-701, datasheet antena GPS y GLONASS, http://texim-europe.com/promotion/783/gns701_datasheet_te.pdf, [9 de abril de 2020].
- Varta, litio, <http://www.varta-consumer.es/es-es/products/batteries/overview>, [9 de abril de 2020].
- Energizer, litio, <https://data.energizer.com/pdfs/I91.pdf>, [9 de abril de 2020].
- Hdsr, descargas, <http://www.hdsdr.de/hardware.html>, [9 de abril de 2020].
- Zadig, descargas, <https://zadig.akeo.ie/>, [9 de abril de 2020].
- Princeton, WSJT, <https://physics.princeton.edu/pulsar/k1jt/>, [9 de abril de 2020].
- Globos sonda, catalogo y datasheets de weatherballoons, <http://www.weatherballoons.asia/weather-balloons-data-sheets>, [9 de abril de 2020].
- Chasat, comunidad, <http://chasat.blogspot.com/>, [9 de abril de 2020].
- paracaídas, store, <http://www.sierrafoxxhobbies.com/es/18-recovery>, [9 de abril de 2020].
- Simulador, simulación de helio necesario para llenar un globo sonda, http://randomaerospace.com/Random_Aerospace/Balloons.html, [9 de abril de 2020].

- Simulador, simulación completa de vuelo estratosférico y helio necesario, <http://predict.habhub.org/#!/uuid=ac717ac17f1f99a7d88c0b8e2785ef7a5756565c>, [9 de abril de 2020].
- Simulador, simulador de velocidad de descenso paracaídas, http://randomaerospace.com/Random_Aerospace/Parachutes.html, [9 de abril de 2020].
- Ícarus, mapa, <https://www.icarusrpa.info/>, [9 de abril de 2020].
- Windy, tiempo en altura, <https://www.windy.com/>, [9 de abril de 2020].
- AEMET, tiempo, <http://www.aemet.es/es/eltiempo>, [9 de abril de 2020].
- Gobierno de España (2018): “RD 1180/2018 de 21 de septiembre”
- Publicidad, Radio Gaceta Cartagonova, <https://www.gacetacartagonova.com/el-politecnico-a-mas-de-30km-de-altura-en-la-metsat-1/>, [9 de abril de 2020].
- Publicidad, Cartagena de hoy, <https://www.cartagenadehoy.com/index.php/2-uncategorised/22445-con-la-atmosfera-como-mesa-de-trabajo>, [9 de abril de 2020].
- Publicidad, Politécnico, <http://politecnicocartagena.es/>, [9 de abril de 2020].
- Financiación, Verkami Crowdfunding, <https://www.verkami.com/projects/25687-ayudanos-a-llegar-a-la-estratosfera>, [9 de abril de 2020].
- Simulación, radioenlace, <https://airlink.ui.com/>, [9 de abril de 2020].
- Cohetes, store, <http://www.sierrafoxbobbies.com/en/50-payload-rockets>, [9 de abril de 2020].
- DS18B20, datasheet: <https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf>, [9 de abril de 2020].
- MPU6050, datasheet: <https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf>, [9 de abril de 2020].
- BMP180, datasheet, <https://cdn-shop.adafruit.com/datasheets/BST-BMP180-DS000-09.pdf>, [9 de abril de 2020].
- DHT11, datasheet, <https://www.mouser.com/datasheet/2/758/DHT11-Technical-Data-Sheet-Translated-Version-1143054.pdf>, [9 de abril de 2020].
- ArduinoMega, datasheet: <https://www.robotshop.com/media/files/pdf/arduinomega2560datasheet.pdf>, [9 de abril de 2020].