



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Politécnica Superior de Gandia

Sistema básico de seguimiento por satélite basado en
módulo ARTIC para especies marinas.

Trabajo Fin de Grado

Grado en Ingeniería de Sistemas de Telecomunicación, Sonido e
Imagen

AUTOR/A: Tro Puigcerver, Ivan

Tutor/a: Sogorb Devesa, Tomás Carlos

Cotutor/a: Pelegrí Sebastiá, José

CURSO ACADÉMICO: 2022/2023

Contenido

I.	Resumen	3
II.	Abstract	3
1.	Introducción	4
1.1.	Presentación.....	4
1.2.	Objetivos.....	4
1.3.	Metodología.....	5
1.4.	Etapas	5
1.5.	Problemas	6
2.	Recopilación de información sobre las diferentes constelaciones basadas en nanosatélites	7
2.1.	Nanosatélites	7
2.2.	Constelación de nanosatélites	7
2.3.	Tipos de orbitas	8
2.4.	Empresas del sector	9
2.5.	Constelación Argos-4 y Kinéis	16
3.	Descripción del equipo a utilizar	18
4.	Plataforma web Argos	23
5.	Configurando el dispositivo	25
5.1.	Parámetros de los sensores	29
6.	Presupuesto	36
7.	Resultados	37
8.	Conclusiones y futuros trabajos	38
9.	Bibliografía	39
10.	ANEXO I. Relación del trabajo con los Objetivos de Desarrollo Sostenible de la Agenda 2030.....	40

Ilustración 1. Datos clave sobre nanosatélites	8
Ilustración 2. Tipos de orbitas de constelaciones	8
Ilustración 3. Parámetros de las orbitas	9
Ilustración 4. Servicios de nanosatélites de Alén	10
Ilustración 5. Proyectos Alén 1	10
Ilustración 6. Proyectos Alén 2	11
Ilustración 7. Proyectos Alén 3	11
Ilustración 8. Servicios de nanosatélites de Sateliot	12
Ilustración 9. Proyectos Arribada1	14
Ilustración 10. Proyectos Arribada 2	14
Ilustración 11. Proyecto Botella de plástico	15
Ilustración 12. SnapperGPS vs GPS	16
Ilustración 13. Raspberry Pi 3 exterior	19
Ilustración 14. Raspberry Pi 3 interior	19
Ilustración 15. Módulo Horizon y ARTIC R2	20
Ilustración 16. Diagrama de bloques del módulo	20
Ilustración 17. Dispositivo montado	21
Ilustración 18. Esquemático 1	21
Ilustración 19. Esquemático 2	22
Ilustración 20. Esquemático 3	22
Ilustración 21. Esquemático 4	23
Ilustración 22. Página web Argos Acceso	24
Ilustración 23. Página web Argos Menú	24
Ilustración 24. Estado del módulo	25
Ilustración 25. Archivo de configuración	26
Ilustración 26. Configuración de satélites 1	27
Ilustración 27. Programación de los satélites 2	28
Ilustración 28. Datos por defecto del archivo default	28
Ilustración 29. Parámetros del acelerómetro	29
Ilustración 30. Programación de acelerómetro 1	30
Ilustración 31. Programación de acelerómetro 2	30
Ilustración 32. Programación de acelerómetro 3	31
Ilustración 33. Programación de acelerómetro 3	32
Ilustración 34. Parámetros del sensor de presión	32
Ilustración 35. Programación del sensor de presión	33
Ilustración 36. Parámetros del sensor de salinidad	33
Ilustración 37. Programación del sensor de salinidad	34
Ilustración 38. Parámetros del Bluetooth	34
Ilustración 39. Página web Argos Predicción de satélites	35
Ilustración 40. Resultado de la predicción de satélites	35
Ilustración 41. Ejemplo de resultados obtenidos	37
Ilustración 42. Ejemplo de resultados obtenidos 2	37

I. Resumen

El proyecto se basa en el análisis y prueba de validación de la constelación satelital de ARGOS basada en la nueva tecnología ARTIC. Esta se basa en mini satélites que orbitan más cerca de la tierra para poder tener una comunicación con ellos usando menos energía. Además de obtener una cobertura durante más horas al día.

Esta constelación de mini satélites se está formando actualmente (2022 – 2023). Por lo que se pretende validar esta y ver su viabilidad de aplicación en los vertebrados marinos como pueden ser las tortugas bobas.

Palabras clave: Comunicación satélite, ARTIC, base de datos, tratamiento de datos, seguimiento de animales.

II. Abstract

The project is based on the analysis and validation of the ARGOS satellite constellation based on the new ARTIC technology. This is based on mini satellites that orbit closer to the earth in order to communicate with them using less energy. As well as getting coverage for more hours a day.

This constellation of mini-satellites is currently being formed (2022 – 2023). Therefore, it is intended to validate this and see its feasibility of application in marine vertebrates such as loggerhead turtles.

Keys Words: Satellite communication, ARTIC, database, data processing, animal tracking.

1. Introducción

1.1. Presentación

Este trabajo consistirá en la puesta en marcha y comprobación del funcionamiento del nuevo modelo de tecnología ARTIC cuando transmite datos de los nuevos sensores añadidos al módulo. Este dispositivo está pensado para colocarlo en una tortuga y realizar un seguimiento del trayecto que realiza mientras se obtienen datos de cuando sale a la superficie y la velocidad a la que se mueve, así como datos de los sensores que lleva.

1.2. Objetivos

El objetivo que nos hemos propuesto es realizar la puesta en funcionamiento de dispositivo para el seguimiento de animales marinos, en nuestro caso, las tortugas, así como testear el funcionamiento del nuevo modelo, el funcionamiento de los sensores incorporados en este dispositivo y que mejoras aporta respecto al modelo anterior, en conjunto con la web Argos (que nos aporta y que funciones tiene disponibles para el análisis de los datos). Esta actuación se trata de un proyecto europeo LIFE-INTEMARES [1], que se trata de un proyecto de conservación marina con el objetivo de lograr una gestión eficaz de los espacios marinos de la Red Natura 2000, que es una red ecológica europea de áreas de conservación de la biodiversidad, y los objetivos específicos de investigación, conservación, seguimiento y vigilancia, formación y educación ambiental.

Aquí podemos resaltar el efecto que tiene nuestro proyecto con los ODS (Objetivos de Desarrollo Sostenible) entre los que se encuentran producción y consumo responsables, vida submarina, acción por el clima, agua limpia y saneamiento y otros.

También se pretende comprobar la extensión de la red de nanosatélites que se emplean para la comunicación de los datos adquiridos, así como la comunicación con kineis y Argos-4, a la par que se añade información de la evolución de las constelaciones de nanosatélites y empresas que emplean dicho sistema de red para sus propios proyectos de innovación.

1.3. Metodología

Iniciaremos con una sección de problemas que iremos ampliando según avancemos y vayan surgiendo y como se han resuelto de ser posible.

A continuación, pasaremos a la sección donde recopilaremos información sobre nanosatélites (constelaciones y tipos de orbitas).

Acto seguido, procederemos a la instalación en el raspberry de los archivos necesarios y los comandos utilizados para poner en marcha y testear el dispositivo.

Por último, resultado en lo que observaremos los datos obtenidos y las conclusiones a las que se ha llegado con este proyecto.

1.4. Etapas

- Recopilación de información sobre las diferentes constelaciones basadas en nanosatélites.
- Test de la constelación Argos-4 y kineis.
- Validación de la comunicación bidireccional con Argos-4 usando ARTIC, para el envío de datos de sensores y sincronización.
- Estimación de consumos del módulo a utilizar tanto en la transmisión como en la recepción.
- Obtención de resultados experimentales.
- Redacción de la memoria del proyecto.

1.5. Problemas

El primer problema que se nos presenta es al simular con la máquina virtual el sistema Ubuntu requerido por el dispositivo GPS (Arribada, Artic) que empleamos. Tanto al instalar el sistema operativo Ubuntu como configurarlo no nos genera ningún problema, pero a la hora de conectar el dispositivo Arribada mediante USB, nos salta la pantalla azul del portátil y reinicia el ordenador cada vez que se conecta el USB. Tras probar diferentes soluciones, buscando en internet y consultando a los proveedores del dispositivo, no se pudo encontrar solución al problema. En el PC de sobremesa funcionaba, pero para tener más movilidad a la hora de realizar las pruebas con el dispositivo se optó por emplear una raspberry pi 3 que hiciera la función de pc, aunque también reduciendo la velocidad de procesamiento, después de todo, no es lo mismo un pc que una raspberry.

Mientras empleamos la raspberry nos surge un pequeño problema con la instalación de Python, puesto que no nos deja instalar la versión de Python3 proporcionada en la página web del dispositivo Arribada, pero nos ofrece la opción de instalar Python9, por lo que, con un pequeño cambio en los comandos de instalación, este problema se solventa.

En el momento de poner a prueba el acelerómetro encontramos problemas para realizar la medición. Procedemos intentando medir la velocidad mientras llevamos el dispositivo en el vehículo, pero al llegar al destino y comprobar los datos que llegan a la web de Argos, no hay datos entrantes. Esperando a recibir los datos con el dispositivo inmóvil, pensando que puede haber almacenado los datos, recibe datos, pero sigue sin marcar nada en la velocidad. Probamos a cambiar la configuración del acelerómetro con diferentes sample rate y modo, pero sigue sin aportarnos nada. Es posible que se deba a que el acelerómetro esté preparado para medir bajo el agua, o que la velocidad que capte solo sea muy pequeña, haciendo muy difícil medir la de un vehículo, o que tenga que coincidir que el satélite este cruzando sobre el equipo en ese momento.

Al no recibir datos de la velocidad se ha intentado extraer la información de la trama de datos completas, pero en el código no se ha podido localizar el orden en el que se encapsula cada valor de los sensores, en el código hay demasiados códigos de versiones anteriores del código que dificultan cualquier búsqueda.

Se ha intentado probar el detector de salinidad, pero se requería un conector especial en el sensor, se ha realizado el intento con cables y un vaso de agua con sal para intentar hacer saltar el sensor y que se piense que está sumergido, pero no se ha observado ningún cambio.

2. Recopilación de información sobre las diferentes constelaciones basadas en nanosatélites

2.1. Nanosatélites

Por lo general, se considera que un nanosatélite es cualquier satélite que pese menos de 10 kilos. CubeSat [2] es un estándar de diseño de nanosatélites que debe cumplir con criterios específicos que controlan factores como su forma, tamaño y peso.

Pueden tener varios tamaños, pero todos se basan en la unidad estándar de CubeSat, que es una estructura en forma de cubo de 10x10x10 centímetros con una masa de aproximadamente de entre 1 y 1,33 kg. Esta unidad es conocida como 1U. Tras los primeros años, esta unidad modular se fue multiplicando y tamaños más grandes de nanosatélites se han vuelto habituales (1.5U, 2U, 3U o 6U). Actualmente se siguen desarrollando nuevos ajustes.

Para recopilar información sobre las constelaciones de nanosatélites se ha realizado una búsqueda de los diferentes tipos y de empresas que empleen esta tecnología, ahora hablaremos un poco de ellas.

2.2. Constelación de nanosatélites

Empezamos con una breve definición de constelación de nanosatélites. Una constelación (o enjambre) de satélites es una red de unidades artificiales idénticas o de tipo similar con el mismo propósito y control compartido. Estos grupos se comunican con estaciones terrestres situadas en todo el mundo y a veces están interconectados. Funcionan como un sistema y están diseñados para complementarse entre sí.

Los nanosatélites se agrupan en constelaciones (grupos) que aportan respaldo, redundancia y granularidad a los servicios que ofrecen. Cada satélite en la constelación es renovado cada 2-4 años, asegurando que el operador siempre cuente con un servicio optimizado, de bajo riesgo y con actualizaciones constantes. Por esa razón, las constelaciones de nanosatélites constituyen un sistema para el que los conceptos de obsolescencia o vida útil dejan de tener sentido.

Datos clave sobre nanosatélites



Órbita polar

Los satélites orbitan alrededor de la Tierra siguiendo trayectorias circulares o elípticas que se consiguen a través de un equilibrio entre la fuerza de atracción de la gravedad y la de escape marcada en su lanzamiento. Al no existir aire, no existe rozamiento que altere la ecuación y pueden permanecer en órbita de manera casi indefinida. Normalmente los nanosatélites describen una órbita polar, es decir, corren en el sentido de los meridianos terrestres. Cuando finaliza el periodo de operación de un nanosatélites, este vuelve a la atmósfera y se desintegra.



Baja altura

Como norma general, los nanosatélites se lanzan en órbitas bajas circulares (400 - 650 Km de altura) y viajan a unos 8 km/s. A esa altura y velocidad, tardan en dar una vuelta a la Tierra unos 90 minutos para completar un total de entre 14 a 16 órbitas por día. Este tipo de órbita baja es ideal para los nanosatélites. Al orbitar más cerca de la Tierra, no solo tienen condiciones óptimas para observación terrestre o comunicaciones sino que están más protegidos de la radiación solar y cósmica.

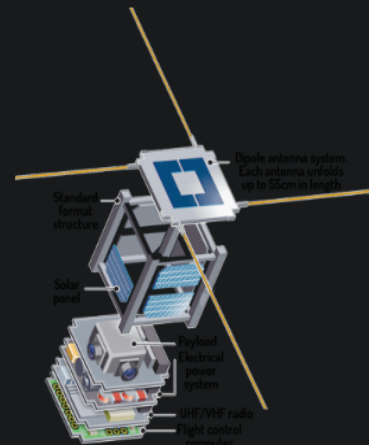


Ilustración 1. Datos clave sobre nanosatélites [3]

El número depende de la finalidad, y varía de varias a miles de unidades. Un ejemplo de mega constelaciones de satélites es Starlink (2.146 satélites activos). En el lado contrario están Sentinel-1 y Sentinel-2, ambas con dos unidades.

2.3. Tipos de orbitas

Las constelaciones se dividen en tres tipos según la altitud de la órbita. Cada una para un propósito concreto (GEO, MEO y LEO).

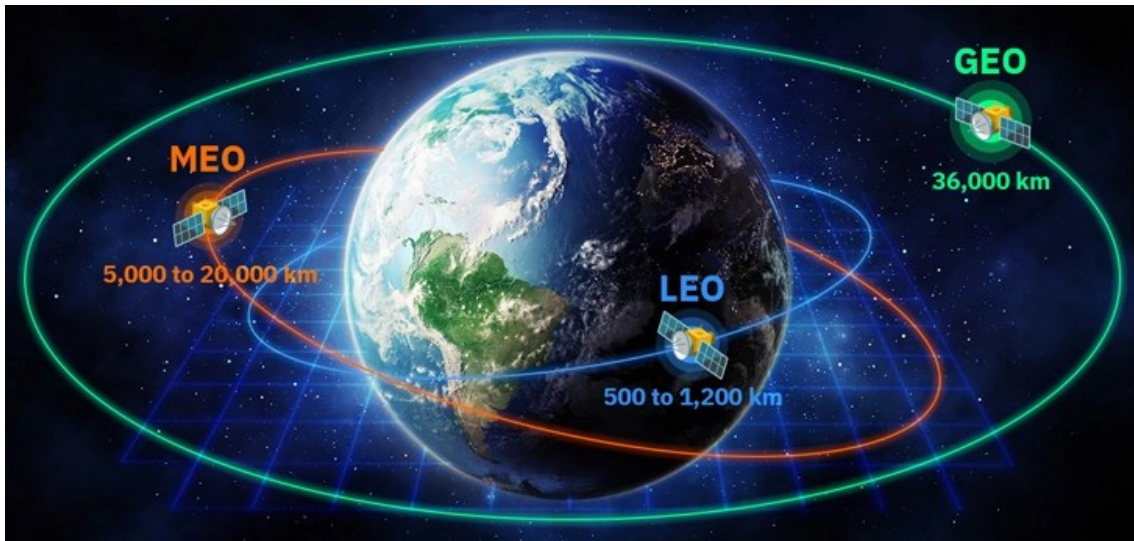


Ilustración 2. Tipos de orbitas de constelaciones [4]

- Geo (geoestacionaria): Se sincronizan con la rotación de la tierra (24h) sobrevolando el ecuador. Es la más habitual para los satélites meteorológicos. Gracias a su altitud cada satélite puede captar el 40% de la superficie del planeta (con 3 de ellos, separados 120° angulares se abarca el planeta entero).

- Meo (acrónimo de enjambre): Normalmente su función es con fines de navegación. Con gran ancho de banda en lugares difíciles (útil en las industrias marítimas y aeroespaciales). Órbita del Sistema de Posicionamiento Global (GPS), Galileo y otras.

- Leo: Población de satélites más densa, ampliamente utilizados por organizaciones gubernamentales, comerciales y no comerciales. Principalmente para investigación, telecomunicación y observación. Pueden tener orbitas circulares o elípticas, las circulares giran desde 1.5 hasta varias horas, mientras que las elípticas tardan más. Constelaciones como Starlink, OneWeb, Iridium y otras pertenecen al tipo LEO.

Parámetro	GEO	MEO	LEO
Altitud	36.000 km	5.000 to 20.000 km	500 to 1.200 km
Área de cobertura	Amplia	Media	Reducida
Velocidad de transmisión del enlace descendente y ascendente (velocidad de la señal)	Lenta	Media	Rápida
Distancia entre estaciones terrestres	Distante	Regional	Local
Antena	Estacionaria	Seguimiento doble	Seguimiento complejo y red terrestre

Ilustración 3. Parámetros de las orbitas [4]

También existen constelaciones híbridas, combinando unidades que viajan en orbitas diferentes, por ejemplo, MEO+GEO, LEO+MEO, etc.

La constelación de satélites EOS SAT es la primera que responde a las necesidades de la industria agrícola.

La primera constelación de satélites se lanzó en los años 60 y muchos se han desplegado desde los 90 hasta hoy. Los enjambres realizan una serie de tareas que van desde la conectividad a Internet por fibra óptica a la monitorización polivalente de la Tierra, pasando por la obtención de imágenes de calidad para su posterior procesamiento de datos mediante IA por parte de las plataformas de análisis. Los usuarios de datos de teledetección celebran el poder adquirir servicios a precios increíbles para obtener respuestas claras a sus preguntas y EOSDA es capaz de ofrecer soluciones rentables y confiables.

2.4. Empresas del sector

- Alén Space [5], equipo creado por la ONU, la ESA (European Space Agency) y la NASA.

El equipo de Alén Space ha colaborado en sus proyectos con algunas de las agencias y organizaciones espaciales más importantes del mundo. La ESA, la NASA, la Oficina de Naciones Unidas para Asuntos del Espacio Exterior (UNOOSA), la Agencia Espacial Brasileira (AEB) y el Instituto Nacional de Técnica Aeroespacial (INTA) han auspiciado algunos de nuestros proyectos, con unos resultados inmejorables. Todos los nanosatélites desarrollados por el equipo de Alén Space [5] han cumplido con éxito sus misiones. Un 100% de acierto.

A continuación, vemos una imagen de las aplicaciones y proyectos en las que emplean nanosatélites:



Ilustración 4. Servicios de nanosatélites de Alén [5]

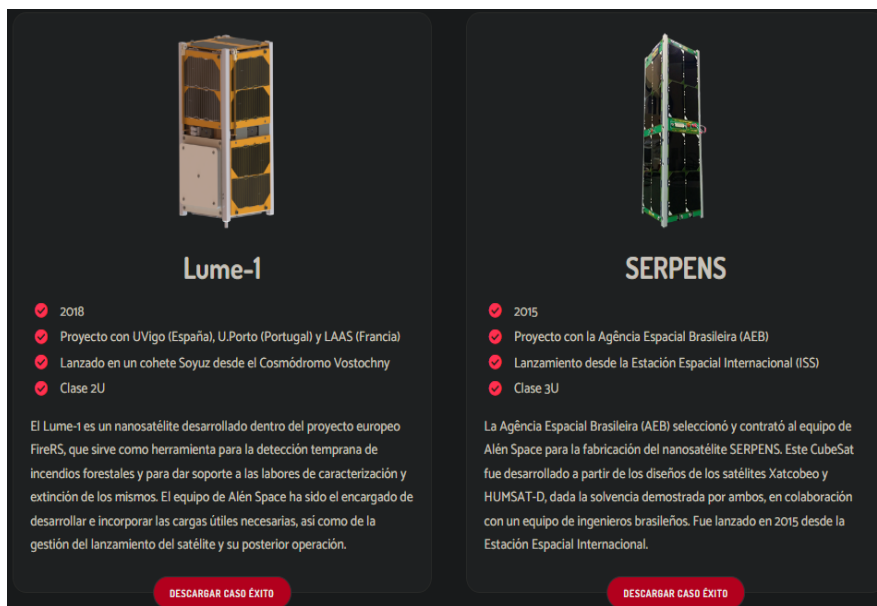


Ilustración 5. Proyectos Alén [5] 1



AlfaCruz

- ✓ 2022
- ✓ Proyecto con la UnB, la FAPDF y la AEB (Brasil)
- ✓ Lanzado en un cohete Falcon 9 desde Cabo Cañaveral en Estados Unidos
- ✓ Clase 1U

Un equipo de estudiantes de la UnB participa en una misión completa de tipo CubeSat que permitirá realizar experimentos en el campo de las comunicaciones y estudiar los efectos del clima espacial en las comunicaciones satelitales en latitudes ecuatoriales. Alén Space aporta años de experiencia en el diseño, fabricación y operación de pequeños satélites para completar la misión en un tiempo récord.




Sateliot

- ✓ 2021
- ✓ Proyecto con Sateliot
- ✓ Lanzado en un cohete Soyuz desde el Baikonur Cosmódromo en Kazajistán
- ✓ Clase 3U

Sateliot lanzará una constelación de nanosatélites para proporcionar servicios de comunicaciones IoT con cobertura 5G. Está previsto que su futura constelación llegue hasta los 100 nanosatélites y comience a ofrecer servicio comercial a partir de 2023. De momento, Alén Space ha diseñado y desarrollado la carga útil del primer CubeSat (3B5GSAT) de la constelación, obteniendo resultados exitosos en órbita.

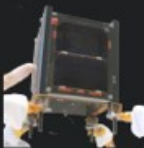
Ilustración 6. Proyectos Alén [5] 2



DustCube

- ✓ 2019-2016
- ✓ Proyecto con ESA y NASA
- ✓ Fase de estudio de Asteroid Impact Mission (AIM)
- ✓ Clase 3U

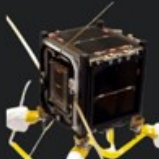
La Agencia Espacial Europea se puso en contacto con el equipo de Alén Space para colaborar en el proyecto DustCube, un nanosatélite creado para estudiar el resultado de un impacto sobre un asteroide con el objetivo de desviar su trayectoria dentro del programa **Asteroid Impact Mission (AIM)**. DustCube debía soportar impactos, radiaciones y temperaturas extremas. Nosotros fuimos los encargados de realizar el estudio de viabilidad de la misión.



HUMSAT-D

- ✓ 2013
- ✓ Proyecto con UNDOOSA y ESA
- ✓ Lanzamiento en cohete ruso Dnepr-1
- ✓ Clase 1U

HUMSAT-D está considerado como el primer nanosatélite lanzado por la ONU. Se trata de un sistema de recogida de datos con objetivos humanitarios, un proyecto respaldado por la Oficina de Naciones Unidas para los Asuntos del Espacio Exterior (UNOOSA) y por la Agencia Espacial Europea (ESA) lanzado en el año 2013 y operado durante un año por el equipo de Alén Space cumpliendo el 100% los objetivos de la misión.



Xatcobeo

- ✓ 2012
- ✓ Proyecto con INTA
- ✓ Lanzamiento en cohete Vega (ESA y ASI)
- ✓ Clase 1U

Xatcobeo se convirtió en 2012 en el primer nanosatélite español en el espacio, el primer gran logro del equipo de Alén Space. Nuestros profesionales lideraron el desarrollo del proyecto y su posterior operación en órbita durante los siguientes dos años y medio. El primero de nuestros casos de éxito en nanosatélites.

Ilustración 7. Proyectos Alén [5] 3

- Sateliot [6], primer operador satelital en ofrecer conectividad IoT sobre 5G NB-IoT estándar.

El Internet de las Cosas (IoT) está revolucionando la producción en diferentes industrias, desde la agricultura inteligente (agricultura de precisión), las movilidades inteligentes, la gestión ganadera o de ciudades inteligentes, la marítima, etc. Las empresas de estas industrias cada vez más digitalizan sus procesos productivos implementando tecnologías IoT.

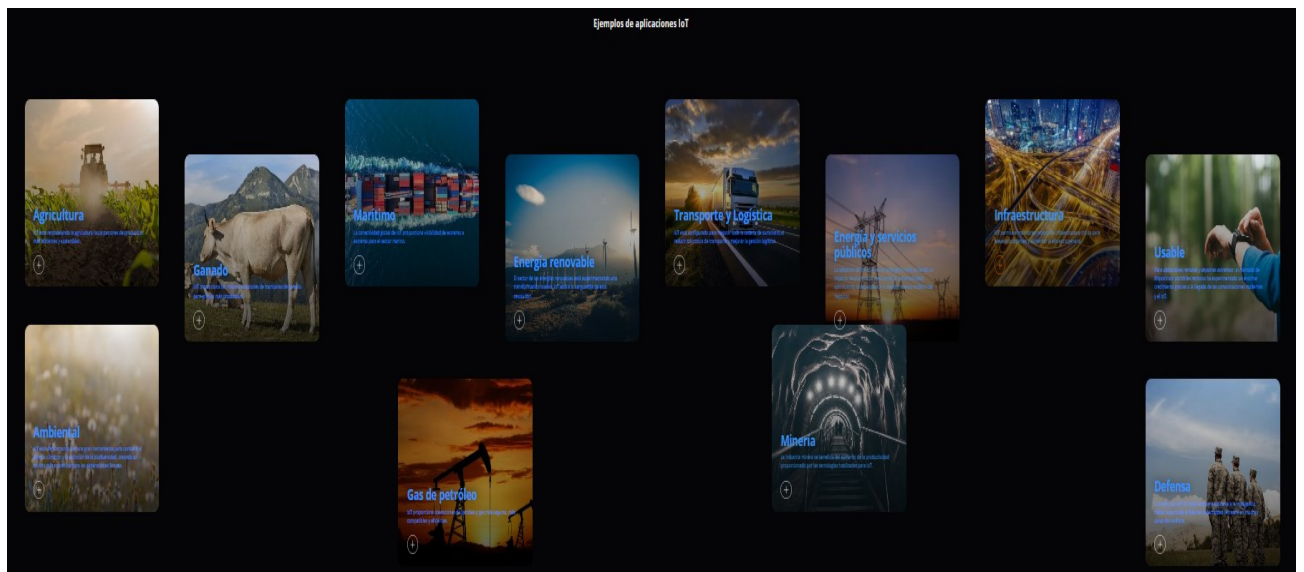


Ilustración 8. Servicios de nanosatélites de Sateliot [6]

Dependiendo de los casos de uso, las empresas utilizan diferentes servicios de conectividad de IoT para sus dispositivos conectados, como Wi-Fi, celular (3G, 4G, 5G), LoRaWAN (red de área amplia de largo alcance) e IoT de banda estrecha (NB-IoT). Estas son las tecnologías de conectividad de IoT comunes que se utilizan en el mercado de IoT. Entre ellos, NB-IoT está diseñado específicamente para aplicaciones de IoT que no requieren baja latencia ni gran ancho de banda, particularmente proyectos de IoT en los sectores agrícola y ganadero.

Su conectividad satelital es una solución de conectividad IoT perfecta para todas las industrias. La tecnología de conectividad IoT satelital de Sateliot se encuentra bajo el protocolo estándar 5G NB-IoT. Ha sido respaldado por la Agencia Espacial Europea (ESA).

También son un operador satelital miembro de GSMA y hemos firmado acuerdos estándar con más de 50 operadores de telecomunicaciones en todo el mundo, por lo que puede beneficiarse de una conectividad IoT celular satelital perfecta conectando sus dispositivos IoT celulares estándar directamente a nuestra constelación de satélites en Low Earth. Órbita (LEO). Es una solución de conectividad IoT escalable, segura y de bajo costo. La constelación sirve como torres de telefonía móvil en el espacio, ampliando la cobertura de los operadores móviles y brindando conectividad global a dispositivos terrestres donde quiera que se encuentren.

- Arribada [7]: Es una organización de investigación y desarrollo de tecnología de conservación sin fines de lucro, con sede en Reino Unido, que se especializa en soluciones de código abierto. Desarrolla conjuntamente tecnología de conservación abiertas, personalizables y basadas en el impacto para organizaciones conservacionistas de todo el mundo, reduciendo costos y ampliando el acceso a las herramientas y soluciones para resolver los desafíos de conservación.

La plataforma Horizon de Arribada [7] proporciona los componentes básicos necesarios para desarrollar sensores inalámbricos y etiquetas de registro biológico de bajo costo. Su diseño modular abierto se centra en la provisión de un tablero de control central que rompe la popular conectividad de entrada/salida (SPI/I2C) para permitir a los desarrolladores conectar sus propios sensores o cargas útiles.

Desarrollada para desbloquear el acceso a hardware de registro biológico accesible y de menor costo, la plataforma se utilizó originalmente para desarrollar etiquetas de registro biológico GPS para tortugas marinas verdes en asociación con la Sociedad Zoológica de Londres. Luego, la National Geographic Society introdujo la conectividad celular y satelital Argos para rastrear los desechos plásticos que salen del delta del Ganges como parte de su expedición Sea to Source.

Horizon es una solución plug-and-play de código abierto para ingenieros, desarrolladores o estudiantes que desean desarrollar sus propias soluciones de telemetría personalizadas.

Desarrollado conjuntamente con la Sociedad Zoológica de Londres, Horizon fue utilizado por la Sociedad National Geographic para rastrear los desechos plásticos que abandonan el delta del Ganges como parte de su expedición Sea to Source.

El kit de desarrollador Horizon incluye GPS, Bluetooth 5.0, un acelerómetro de 9 ejes, celular 3G y un transmisor Argos ARTIC R2 para que pueda comenzar a funcionar rápidamente, compatible con los satélites ARGOS-4 de próxima generación y la futura constelación de nanosatélites Kineis.

Actualmente especializados en:

- Telemetría satelital y GPS

- Biologging

- Detección térmica de bajo costo

- Cámaras time-lapse resistentes

- Aprendizaje automático acústico



- Diseño de reflector acústico pasivo.

También aloja sitios web, correo electrónico y servidores para organizaciones benéficas y sin fines de lucro, utilizando Web Host Manager y cPanel para proporcionar un panel fácil de usar para los clientes que desean administrar sus propias cuentas. Arribada [\[7\]](#) gestiona el sistema operativo, los firewalls y la seguridad subyacente para minimizar costos.

Proyectos:

Desbloqueo de acceso


En qué estamos trabajando



Arribada Horizon Bioregistro y telemetría

- Plataforma de bioregistro de hardware de código abierto
- Soporta Argos ARTIC
- Soporta 2G/3G celular


Arribada & ZSL



Cámaras Timelapse resistentes

- Timelapse configurable para implementaciones a largo plazo
- Bajo costo, robusto y escalable


Relaj de pingüinos



Mitigación del conflicto entre humanos y elefantes

- Imagen térmica de bajo costo
- Aprendizaje automático integrado (Tensorflow)
- Alertas de alerta temprana

WWF y laboratorios salvajes



Detección de alerta temprana del oso polar

- Imagen térmica de bajo costo
- Aprendizaje automático integrado (Tensorflow)
- Alertas móviles/LoRa

WWF y laboratorios salvajes

Ilustración 9. Proyectos Arribada [7] 1



Desarrollo de pilas de combustible microbianas

- Generación de electricidad utilizando plantas vivas y microbios.
- Energía perpetua para sensores IoT de baja energía
- Prototipo en etapa inicial

Plant-e, Universidad de Cambridge, Xnor.ai, ZSL



micropolilla

- Grabador acústico pasivo Miniaturizado AudioMoth
- Desarrollado para accesorios transportados por animales.
- Compatible con el firmware AudioMoth existente

Diseñado por Ruby Lee, dispositivos acústicos abiertos



Telemetría acústica del angelote

- Diseño de gabinete personalizado para acomodar receptores Vemco
- Colaboración multisocio

Red de seguimiento oceánico del proyecto Angel Shark



SMART Desktop/Conectar datos en vivo

- Muestre datos en vivo recibidos a través de sensores en SMART Desktop
- Puente LoRa (muestra datos del sensor LoRa en vivo dentro de SMART)

Desarrollado por Krasi Georgiev

Ilustración 10. Proyectos Arribada [7] 2



Ilustración 11. Proyecto Botella de plástico

HORIZON - Botella de agua de plástico con seguimiento Argos GPS de 500 ml

La botella de agua de plástico con seguimiento satelital, fue diseñada originalmente para imitar un artículo de plástico típico de uso diario para “National Geographic’s Sea to Source Plastics Expedition”¹. Coincide con la forma y el tamaño de una botella de agua de plástico estándar de 500 ml en comparación con boyas a la deriva oceánicas más grandes y pesadas u objetos similares que se mueven de manera diferente en el agua debido a su tamaño, peso y resistencia.

Al finalizar la expedición, la Sociedad Zoológica de Londres también utilizó las botellas para rastrear la contaminación plástica del océano a través del Canal de la Mancha como parte de su campaña One Less y se desplegaron en la Conferencia Oceánica de la ONU de 2022 en Lisboa, que revela la ruta y el destino de los residuos plásticos desde las playas de Lisboa, Portugal, hasta el mar abierto.

Dentro de cada botella hay un registrador GPS Arribada Horizon integrado, un transmisor Argos ARTIC R2 y un paquete de baterías primarias de litio de 6,5 A. La placa Horizon utiliza un nRF52840 que se activa para adquirir GPS en un intervalo predeterminado según la duración de la implementación y la duración de la batería. Las ubicaciones GPS se registran y guardan en la memoria y la última posición GPS se transmite a través de Argos a los satélites que pasan.

¹ Proyecto de seguimiento de botella de plástico con el propósito de conocer como daña la contaminación plástica el océano.



La necesidad de velocidad en la telemetría de tortugas marinas – SnapperGPS vs GPS Asistido

15 de septiembre de 2021

SnapperGPS vs GPS asistido Para adquirir señales de GPS al rastrear tortugas marinas, su receptor de GPS debe estar por encima de la línea de flotación y expuesto al cielo. Es más fácil decirlo que hacerlo cuando las tortugas marinas pueden pasar sólo 2.2 segundos rompiendo la superficie del agua cuando salen a tomar aire....

Ilustración 12. SnapperGPS vs GPS

2.5. Constelación Argos-4 y Kinéis

Argos

Argos [8] es un sistema de posicionamiento y recopilación de datos dedicado al estudio y protección del medio ambiente. Desde 1978, se rige a través de una asociación entre instituciones nacionales e internacionales (NOAA, NASA, CNES, ISRO y EUMETSAT).

Para mantener y mejorar el sistema planean 5 nuevos lanzamientos entre 2019 y 2036.

El satélite GAZelle de General Atomics fue enviado al cielo el 7 de octubre de 2022 desde la base de lanzamiento espacial de Rocket Lab en Nueva Zelanda, transportando Argos-4. A finales de noviembre, el satélite Oceansat-3 de la Organización de Investigación Espacial de la India (ISRO) para biología marina y seguimiento y vigilancia de los océanos será puesto en órbita desde el Centro Espacial Satish Dhawan de la India con otro instrumento Argos-4 de cuarta generación. Juntos, los dos nuevos instrumentos aumentarán significativamente la cantidad de datos recopilados en todo el mundo.

Desde que Argos [8] entró en servicio, se han sucedido cuatro generaciones de instrumentos. El objetivo de Argos-4 es asegurar la continuidad de las misiones Argos-2 y Argos-3 ofreciendo al mismo tiempo un mejor rendimiento y capacidad a través de una gama más amplia de frecuencias.

Medir las variaciones en la temperatura, las corrientes y la salinidad del océano, monitorear la capa de hielo, rastrear la migración de la vida silvestre, ayudar a la gestión pesquera sostenible y apoyar el transporte marítimo son algunas de las muchas

aplicaciones a las que sirve Argos [8]. Los avances tecnológicos que ofrece Argos allanarán el camino para nuevas aplicaciones como la medición de la calidad del aire, el seguimiento de rebaños y la agricultura conectada. El sistema Argos [8] está construido alrededor de transmisores de radio cuyas señales son recibidas por instrumentos dedicados que vuelan en satélites en órbita terrestre. Estas señales se registran y luego se transmiten a estaciones receptoras y se procesan en el centro de Kinéis [9].

Antes de que los sistemas públicos GNSS (GPS, Galileo...) estuvieran siquiera disponibles, Argos [8] permitía localizar miles de balizas en todo el mundo mediante el efecto Doppler. Pero también fue el primer sistema de recopilación de datos por satélite, originalmente para aplicaciones ambientales, décadas antes de que Internet de las cosas (IoT) hiciera su avance.

Kinéis

Kinéis [9] es un sistema global de recopilación de datos y ubicación basado en satélites diseñado para aplicaciones de IoT. Permite que cualquier objeto equipado con un transceptor compatible, denominado dispositivo conectado, esté ubicado en todo el mundo y transmita datos significativos recopilados de sus sensores de medición. Kinéis [9] proporciona conectividad IoT que está disponible en todas partes, especialmente en áreas remotas donde no hay otra red disponible.

El sistema está compuesto por 4 componentes esenciales:

- Dispositivos IoT certificados para comunicación Kinéis [9], ubicados en cualquier parte del mundo
- Una constelación de satélites en órbita terrestre baja distribuidos por todo el planeta
- Una red de 20 estaciones terrestres ubicadas en todo el mundo
- Un centro de procesamiento y control de misión ubicado en Toulouse, Francia

Los dispositivos se comunican con los satélites Kinéis [9] en bandas UHF (Ultra-High Frequency) de alrededor de 401MHz. Esta gama de frecuencias tiene la ventaja de reducir las pérdidas en el espacio libre cuando la señal atraviesa la atmósfera para llegar al satélite, permitiendo así una potencia de transmisión reducida.

Las frecuencias utilizadas para la comunicación Kinéis [9] están dedicadas a servicios móviles por satélite y servicios de monitoreo ambiental en todo el mundo, como se describe en el reglamento de la UIT.

Su conectividad se basa en la primera constelación europea de nanosatélites que, junto con una red de estaciones terrestres innovadoras, permite dispositivos conectados de bajo consumo, sencillos y asequibles. Cada dispositivo transmite datos a los satélites que pasan por encima, que luego envían estos datos a las estaciones en tierra para ser procesados en el centro de servicio de Kinéis [9] y finalmente entregados al usuario.

Kinéis [9] cuenta con una constelación de satélites polares de órbita terrestre baja, lo que significa que proporcionan una verdadera cobertura global. Desde unos pocos cientos de kilómetros sobre nuestras cabezas, reciben las señales de cualquier dispositivo IoT durante su paso de 10 minutos y las reenvían a la estación terrestre más cercana.

Kinéis [9] está en pleno funcionamiento con la carga útil de 8 Argos, alojada en los satélites de nuestros socios heredados del histórico sistema Argos [8], que ya permite a cualquiera recopilar hasta varios mensajes por hora desde cualquier dispositivo. En 2024, una nueva constelación de 25 nanosatélites dedicados ofrecerá rendimientos mejorados para lograr una conectividad casi en tiempo real. Del tamaño de una caja de

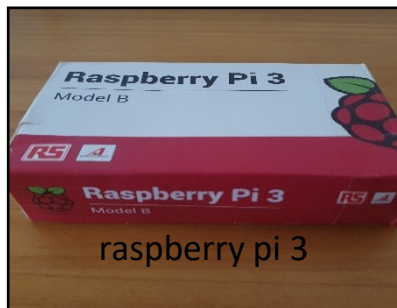
zapatos, estos satélites de 16U (40x20x20 cm) pesan menos de 30 kg y se benefician de una propulsión eléctrica para corregir la órbita y evitar colisiones, alimentada por un par de paneles solares.

Por cierto, algunos satélites también están equipados con una segunda carga útil de detección AIS de alto rendimiento, para el seguimiento del tráfico marítimo mundial.

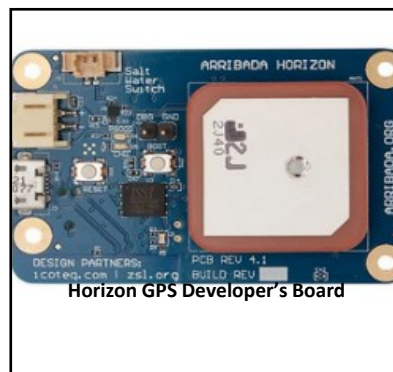
De vuelta en la Tierra, Kinéis [9] opera una red de estaciones terrestres distribuidas por todo el mundo para recibir las señales recogidas y transmitidas por los satélites. Con antenas de matriz en fase de última generación, las estaciones terrestres podrán rastrear varios satélites a la vez y asegurarse de que no pierdan ningún dato. De esta manera, cada satélite tendrá la seguridad de encontrar una estación terrestre disponible en su camino, inmediatamente después de haber recopilado datos de un dispositivo.

Una vez que estos datos finalmente regresan a tierra, se transfieren a través de un enlace seguro de Internet al centro de servicio operativo 24 horas al día, 7 días a la semana. Hasta ahora, siguen constituyéndose en una serie de bits que incluyen información codificada pero esencial: número de identificación, longitud de los datos, código de corrección de errores... En el centro de servicio, Kinéis [9] clasifica los datos útiles, corrige los errores, realiza el cálculo del Doppler, ubicaciones basadas y finalmente distribuye los datos al usuario correspondiente. Se pone a disposición del usuario una interfaz web para la gestión de suscripciones, la configuración del servicio y la configuración de la estrategia de recuperación de datos a través de una API estándar.

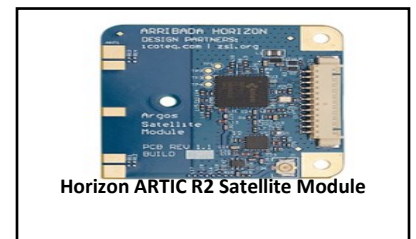
3. Descripción del equipo a utilizar



Para la configuración del dispositivo empleamos la Raspberry que nos proporciona comodidad de transporte y velocidad suficiente para trabajar.



El módulo GPS Horizon independiente cuenta con un microcontrolador nRF52840, Bluetooth 5.0, un módulo GNSS uBlox NEO-M8N, un acelerómetro BMX160 de 9 ejes, un interruptor de agua salada y carga Li-ion/LiPo.



El módulo satelital Horizon ARTIC R2 se conecta a la placa GPS Horizon a través del conector picoblade para formar una etiqueta satelital completamente funcional. También se puede integrar con dispositivos/microcontroladores personalizados para incorporar conectividad satelital Argos ARTIC.

Raspberry Pi 3 modelo B

Se intento realizarlo en un portátil, pero hubo problemas con la instalación del software para el proyecto, por lo que se optó por la raspberry que había disponible en el

laboratorio. Contamos con el Raspberry Pi 3 modelo B para la programación del dispositivo ya que nos permite movernos fácilmente y solo requerimos una pantalla a la que conectarse y un teclado con ratón para operarlo.



Ilustración 13. Raspberry Pi 3 exterior



Ilustración 14. Raspberry Pi 3 interior

Como podemos observar en las imágenes, contamos con todo lo necesario en este encapsulado que facilita el transporte y el mantenimiento de la Raspberry, aunque también hay que decir que dificulta la ventilación y en ocasiones tras su uso prolongado se advierte un aumento en la temperatura de la Raspberry, por lo que la opción de extraer la parte superior, como se observa en la imagen de la derecha, nos brinda una mejor ventilación.

El equipo empleado es el de Arribada [\[7\]](#), cuenta con el módulo Horizon con el GPS y sus sensores y el módulo ARTIC R2.

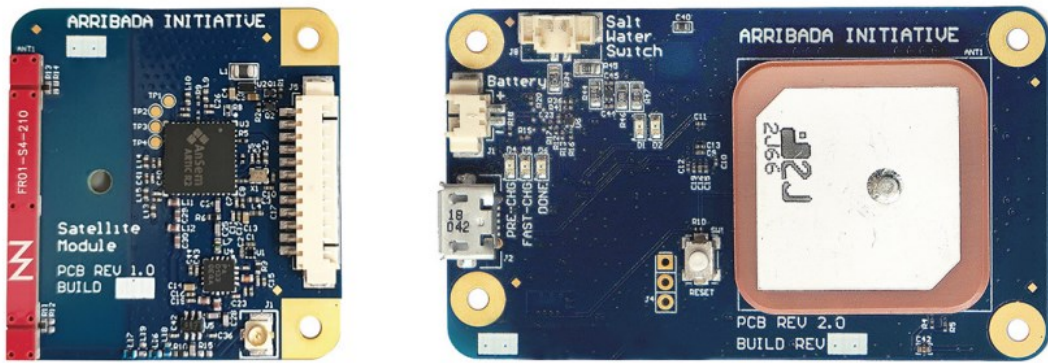


Ilustración 15. Módulo Horizon y ARTIC R2

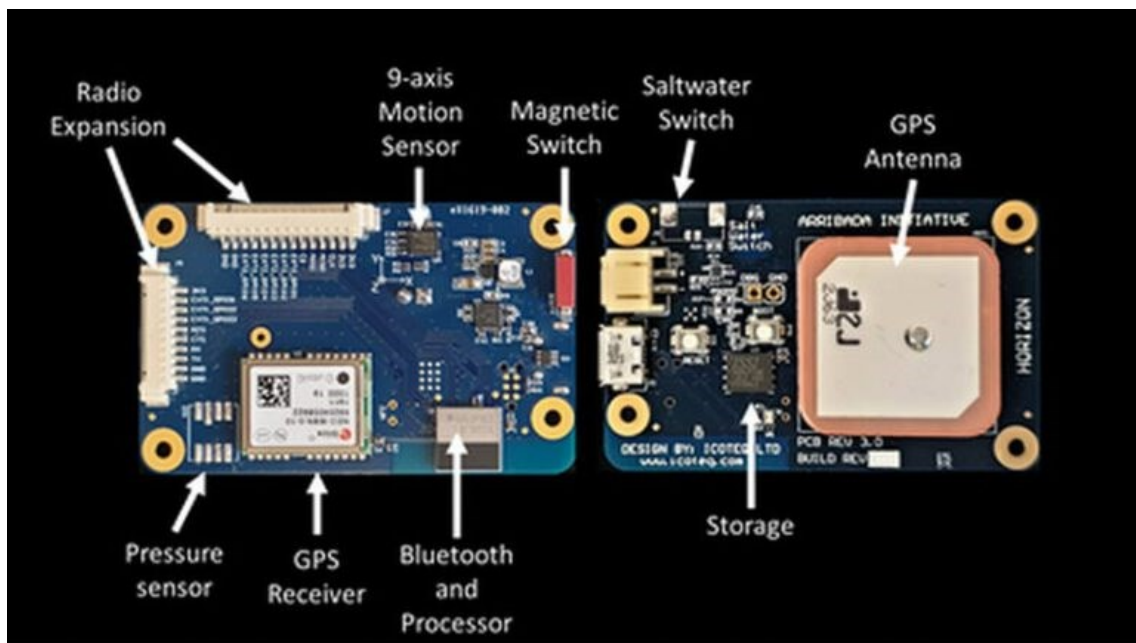


Ilustración 16. Diagrama de bloques del módulo

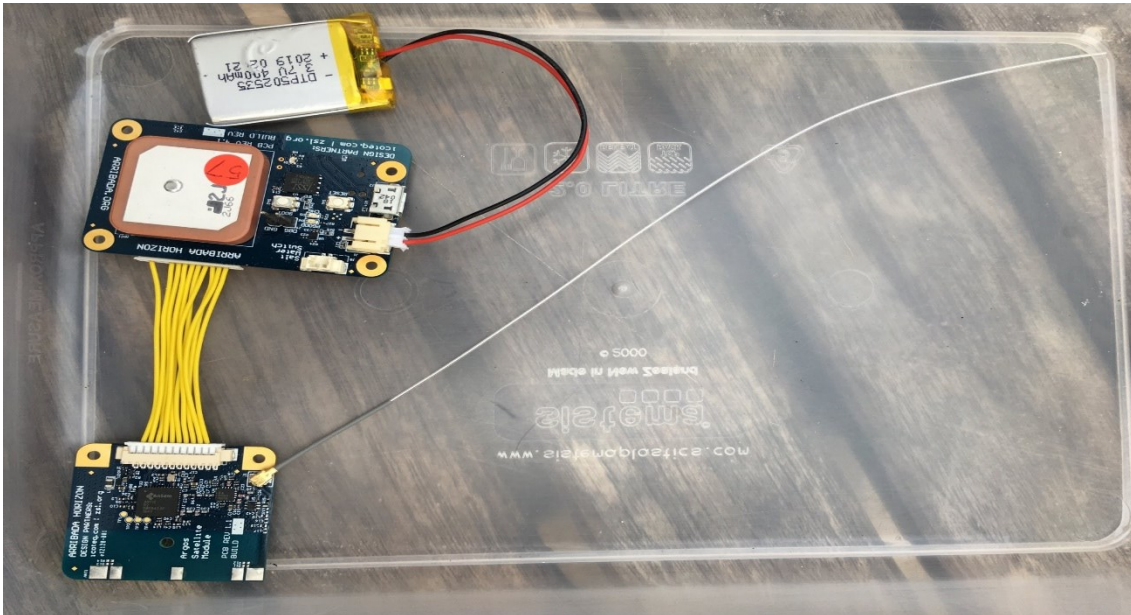


Ilustración 17. Dispositivo montado

Esta versión cuenta con acelerómetro, Bluetooth, sensor de presión, gps y el detector de salinidad. El acelerómetro permitirá medir la velocidad a la que se desplaza la tortuga en su travesía. El Bluetooth permitirá acceder a información y puede que actualizar datos del dispositivo cuando pase cerca de una torre Bluetooth. El sensor de presión y el detector de salinidad nos permitirán saber cuándo la tortuga emerge a la superficie, momento en el que se recargue la batería con el panel solar. El gps nos brindará la información de la posición de la tortuga en su trayecto.

Y también contamos con sus esquemáticos:

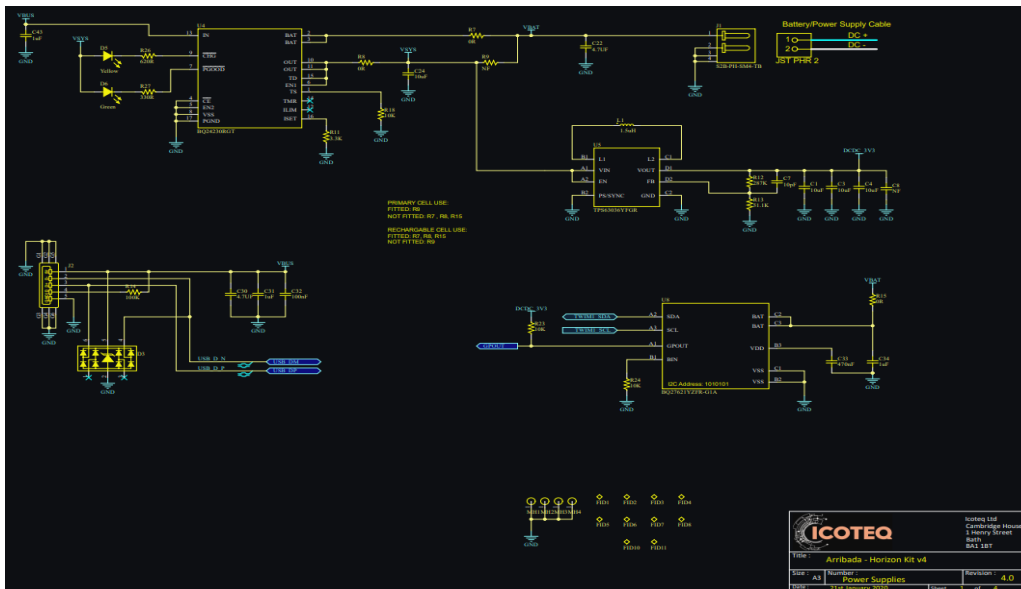


Ilustración 18. Esquemático 1

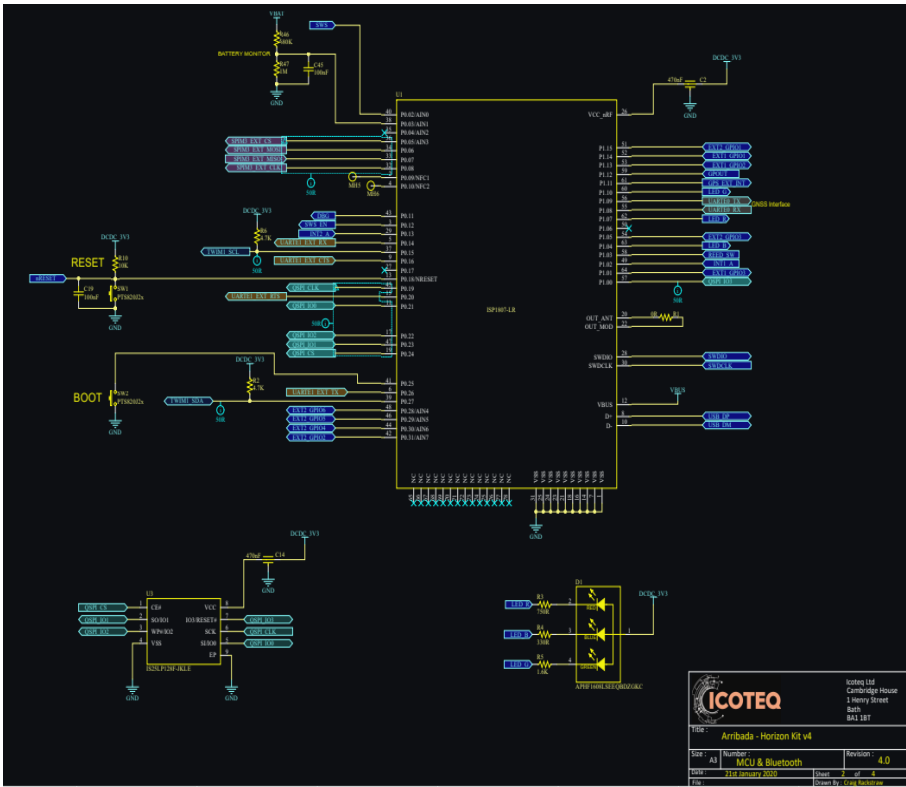


Ilustración 19. Esquemático 2

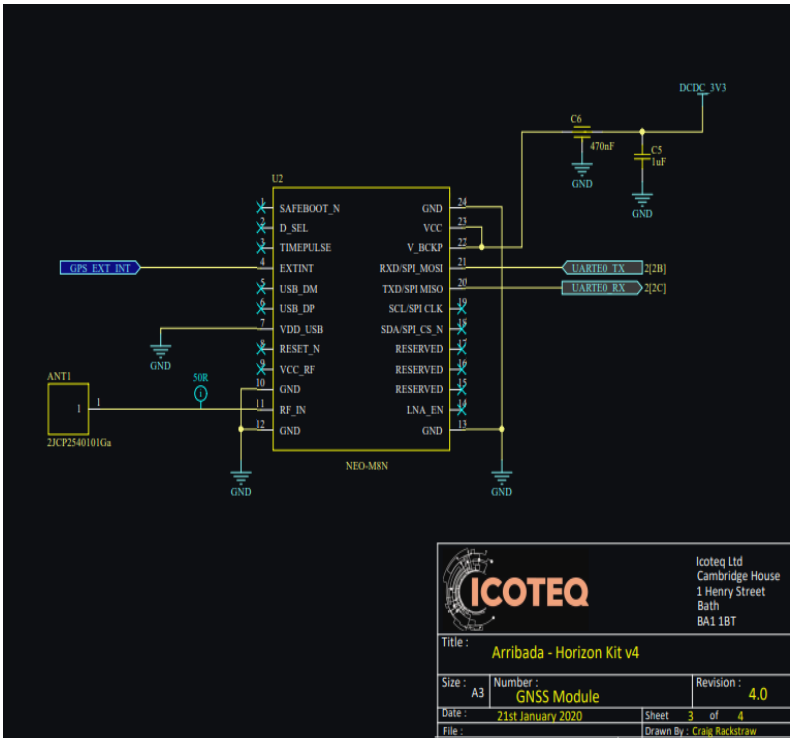


Ilustración 20. Esquemático 3

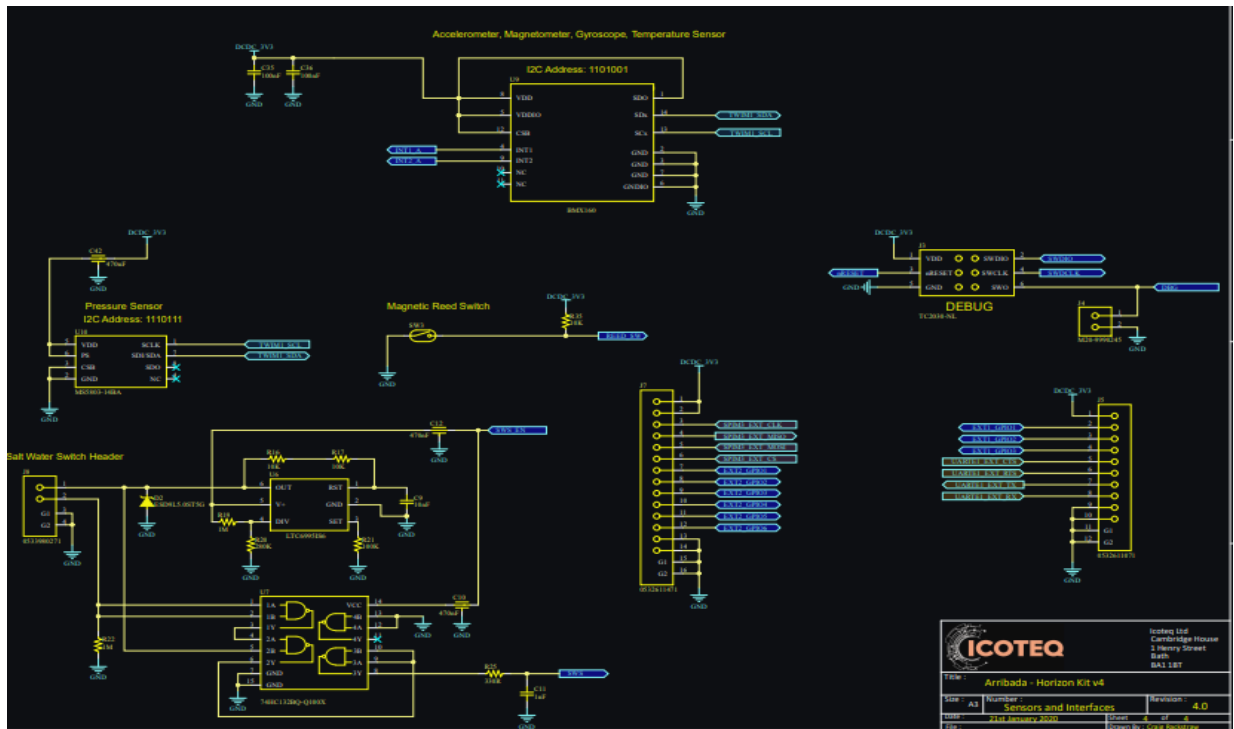


Ilustración 21. Esquemático 4

4. Plataforma web Argos

Para el análisis de los datos se ha optado por la web Argos [8], que está orientada a vida marina y tiene un coste más reducido, por lo que hablaremos un poco de ella antes de ponernos a trabajar.

Argos [8] es un sistema global de localización y recopilación de datos basado en satélites que incorpora las últimas innovaciones en tecnología espacial.

Es un sistema mejorado que proporciona acceso fácil y universal a datos satelitales para una variedad de aplicaciones ambientales inteligentes.

Argos [8] ofrece a los científicos una herramienta para mejorar su comprensión de nuestro medio ambiente y ayuda a las industrias y administraciones a cumplir con las normas de protección ambiental ofreciendo servicios de datos para un amplio espectro de aplicaciones.

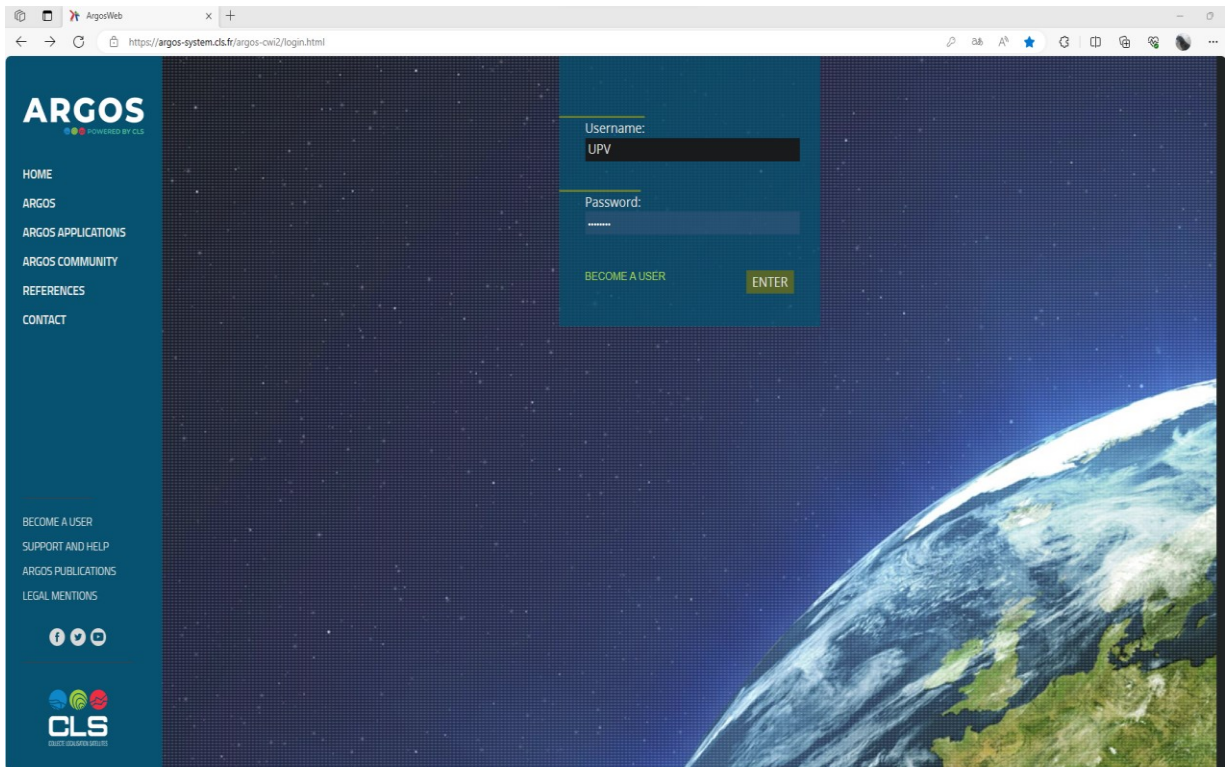


Ilustración 22. Página web Argos [8] Acceso

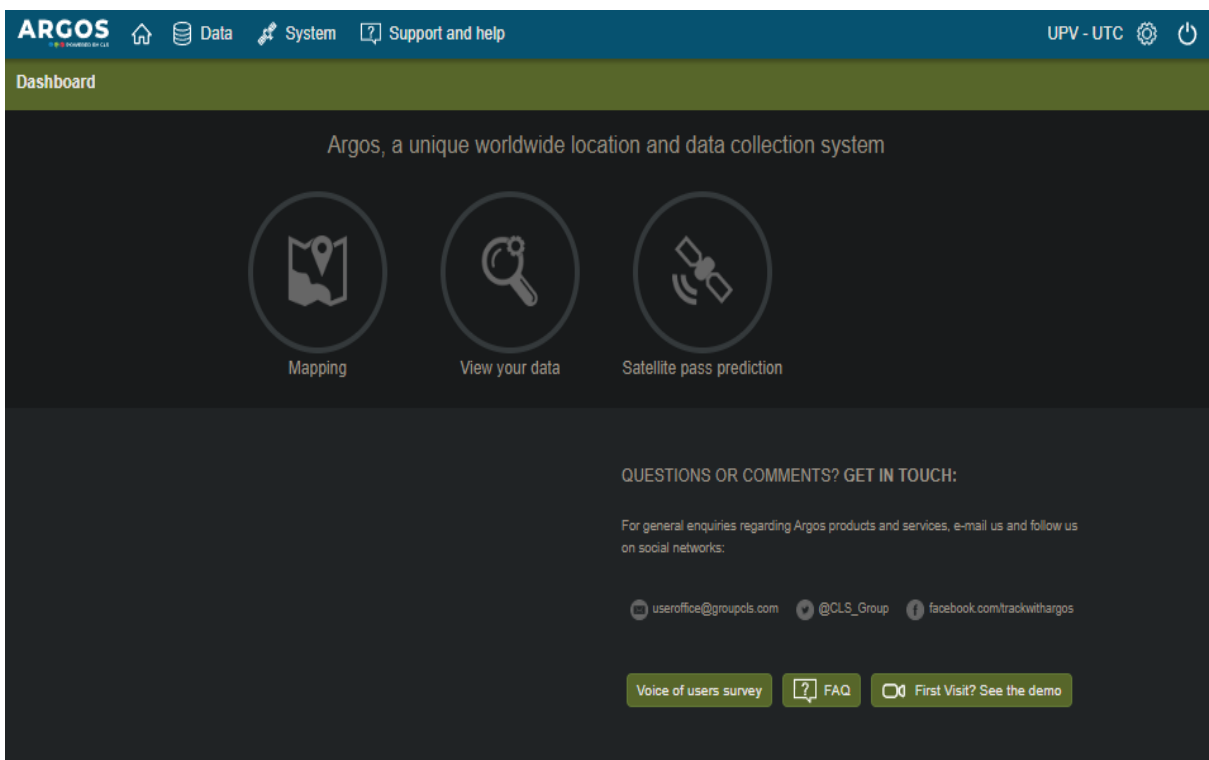


Ilustración 23. Página web Argos [8] Menú

5. Configurando el dispositivo

Antes de empezar a usarlo el dispositivo tenemos que configurarlo. Para ello, una vez iniciada la Raspberry nos dirigimos a la ventana de comandos “cmd” y siguiendo la guía que nos proporciona la página web de Arribada [\[7\]](#), con alguna variación debido a actualizaciones, ejecutamos los comandos.

```
sudo apt-get update
```

```
sudo apt-get install -y git python3 python3.7 python3-pip libglib2.0-dev libyaml-dev  
usbutils python3-dev
```

```
sudo pip3 install Python-dateutil==2.8.0
```

```
git clone https://github.com/arribada/horizon.git
```

```
cd horizon/tools
```

Aquí lo que estamos haciendo es instalar las librerías necesarias de Python y copiando los archivos que se encuentran en github a nuestra PC y luego accedemos a la carpeta que nos interesa con el comando cd. En nuestro caso, la segunda línea de código no la realizaba por la versión de Python, por lo que tuvimos que actualizarla a la siguiente:

```
sudo apt-get install -y git python3 python3.9 python3-pip libglib2.0-dev libyaml-dev  
usbutils python3-dev
```

A continuación:

```
cd 2 0 2
```

```
sudo python3setup.py install
```

Para comprobar la versión que hemos instalado y comprobar que todo está correcto podemos usar el código siguiente:

```
sudo tracker config --version
```

Con esto ya tenemos instalado lo necesario para comunicarnos con el dispositivo. Ahora conectamos el dispositivo a la Raspberry para comunicarnos. Lo primero, después de conectarlo comprobamos que podemos comunicarnos, con el siguiente comando podemos pedirle que nos envíe el estado:

```
sudo tracker_config --status
```

Con lo que debería devolvernos algo como lo siguiente:

```
{  
  "gps_module_detected": true,  
  "cellular_module_detected": false,  
  "sim_card_present": "N/A",  
  "sim_card_imsi": "N/A",  
  "satellite_module_detected": false  
}
```

Ilustración 24. Estado del módulo

Habiendo comprobado la conexión, nos disponemos a configurar el dispositivo, primero el firmware de ARTIC R2:

```
sudo tracker_config --firmware type ARTIC --firmware sample config/artic.bin
```

El siguiente paso es acceder al archivo de configuración del dispositivo, primero lo descargaremos a nuestro equipo con el siguiente comando:

```
sudo tracker config --read config default.json
```

Y con el comando siguiente podremos acceder a él desde la ventana de comandos y editar los parámetros que necesitemos:

```
sudo nano default.json
```

El archivo se ve de la siguiente forma:

```
1  {
2    "version": 1,
3    "accelerometer": {
4      "config": 0,
5      "highThreshold": 0,
6      "logEnable": false,
7      "maximumAquisitionTime": 0,
8      "mode": "PERIODIC",
9      "sampleRate": 0,
10     "scheduledAquisitionInterval": 0
11   },
12   "battery": {
13     "logEnable": true,
14     "lowThreshold": 0
15   },
16   "bluetooth": {
17     "advertisingInterval": 0,
18     "connectionInterval": 0,
19     "inactivityTimeout": 0,
20     "logEnable": true,
21     "phyMode": "1_MBPS",
22     "scheduledDuration": 0,
23     "scheduledInterval": 0,
24     "triggerControl": ["SCHEDULED", "ONE_SHOT"]
25   },
26   "logging": {
27     "dateTimeStampEnable": true,
28     "enable": true,
29     "startEndSyncEnable": false
30   },
31   "pressureSensor": {
32     "highThreshold": 0,
33     "logEnable": false,
34     "lowThreshold": 0,
35     "maximumAquisitionTime": 0,
36     "mode": "PERIODIC",
37     "sampleRate": 0,
38     "scheduledAquisitionInterval": 0
39   },
40   "saltwaterSwitch": {
41     "hysteresisPeriod": 0,
42     "logEnable": false
43   },
44   "gps": {
45     "logPositionEnable": true,
46     "logTFFEnable": true,
47     "logDebugEnable": true,
48     "maximumAquisitionTime": 20,
49     "scheduledAquisitionInterval": 120,
50     "scheduledAquisitionNoFixTimeout": 0,
51     "testFixHoldTime": 60,
52     "mode": "SCHEDULED"
```

Ilustración 25. Archivo de configuración

```

"iot": {
  "enable": true,
  "logEnable": true,
  "minBatteryThreshold": 0,
  "satellite": {
    "artic": {
      "argosProtocol": 2,
      "bulletin": [
        {
          "params": [
            6889.94580078125,
            97.46530151367188,
            87.28900146484375,
            -23.746000289916992,
            94.9832992553711,
            -6.5
          ],
          "satelliteCode": "A1",
          "secondsSinceEpoch": 1639180494
        },
        {
          "params": [
            7195.60009765625,
            98.69339752197266,
            331.9620056152344,
            -25.34000015258789,
            101.3593978881836,
            0.0
          ],
          "satelliteCode": "MB",
          "secondsSinceEpoch": 1639178579
        },
        {
          "params": [
            7195.6162109375,
            98.71399688720703,
            344.2040100097656,
            -25.34000015258789,
            101.35970306396484,
            0.0
          ],
          "satelliteCode": "MC",
          "secondsSinceEpoch": 1639175668
        },
        {
          "params": [
            7180.212890625,
            98.66560363769531,
            303.6629943847656,
            -25.257999420166016,
            101.03500366210938,
            -0.7200000286102295
          ],
          "satelliteCode": "15",
          "secondsSinceEpoch": 1639177969
        }
      ]
    }
  }
}

```

Ilustración 26. Configuración de satélites 1

```

    {
      "params": [
        7225.51904296875,
        98.97560119628906,
        347.93701171875,
        -25.496000289916992,
        101.99040222167969,
        -1.0800000429153442
      ],
      "satelliteCode": "18",
      "secondsSinceEpoch": 1639176604
    },
    {
      "params": [
        7226.14501953125,
        99.1697998046875,
        314.5249938964844,
        -25.49799919128418,
        102.00309753417969,
        -0.7599999904632568
      ],
      "satelliteCode": "19",
      "secondsSinceEpoch": 1639174923
    },
    {
      "params": [
        7160.18310546875,
        98.54100036621094,
        106.30000305175781,
        -25.152999877929688,
        100.6135025024414,
        0.0
      ],
      "satelliteCode": "SR",
      "secondsSinceEpoch": 1639176528
    }
  ],
  "deviceAddress": "03:82:52:79",
  "minElevation": 40,
  "warmupTime": 5
},
"connectionPriority": 0,
"enable": true,
"maxInterval": 60,
"minInterval": 0,
"minUpdates": 3,
"randomizedTxWindow": 5,
"retransmissionCount": 3,
"retransmissionInterval": 60,
"statusFilter": [
  "LAST_GPS_LOCATION",
  "BATTERY_VOLTAGE"
],
"testModeEnable": false
}
},
}

```

Ilustración 27. Programación de los satélites 2

```

"logging": {
  "dateTimeStampEnable": true,
  "enable": true
},
"rtc": {
  "dateTime": "Thu Jan 1 00:18:19 1970",
  "syncToGPSEnable": true
},
"system": {
  "deviceName": "DEFAULT"
},
"version": 1
}

```

Ilustración 28. Datos por defecto del archivo default

En el archivo se pueden ver los diferentes sensores y su estado, este es un archivo hecho por defecto con el que se puede trabajar. Observamos que cada sensor cuenta con la opción de “LogEnable” que nos permite activar a o desactivar los sensores según nos convenga, lo que permite ahorrar energía y no sobrecargar de información al dispositivo, recordemos que este es un dispositivo con un tamaño muy reducido, puesto que debe colocarse sobre la tortuga, y su capacidad de procesamiento y la alimentación de energía provista, incluso con el panel solar, son limitadas.

5.1. Parámetros de los sensores

Para los parámetros de los sensores, tienen varios modos de funcionamiento. Contamos con información de ellos:

- Acelerómetro.

acelerómetro	
logEnable	booleano, habilitar el registro del acelerómetro
Umbral alto	número, vector magnitud suma umbral alto (cuando se ACTIVA)
tasa de muestra	número, frecuencia de muestreo (cuando es PERIÓDICO)
modo	cadena, "PERIÓDICO", "DISPARADO"
intervalo de adquisición programado	número, el período de repetición en segundos en el que se inician las adquisiciones para el modo PERIÓDICO, 0=>continuamente
tiempo máximo de adquisición	número, el período en segundos durante el cual se realizan las adquisiciones para el modo PERIÓDICO, 0 = indefinidamente

Ilustración 29. Parámetros del acelerómetro

Si observamos la programación del acelerómetro, se emplea el LSM9DS1 y tiene diferentes modos de muestro según la frecuencia en la que trabaje como podemos ver a continuación:

```

83  * @brief   Initialises the accelerometer
84  *
85  * @return  SYSHAL_AXL_PROVISTIONING_NEEDED if configuration tags still need
86  *         to be set
87  * @return  SYSHAL_AXL_NO_ERROR on success
88  */
89  int syshal_axl_init(void)
90  {
91      if (syshal_i2c_is_device_ready(I2C_AXL, LSM901_AG_ADDR) != SYSHAL_I2C_NO_ERROR)
92      {
93          DEBUG_PR_ERROR("LSM9051 unresponsive");
94          return SYSHAL_AXL_ERROR_DEVICE_UNRESPONSIVE;
95      }
96
97      // Fetch all required configuration tags
98      int ret;
99
100     ret = sys_config_get(SYS_CONFIG_TAG_AXL_SAMPLE_RATE, NULL);
101     if (ret < 0)
102         return SYSHAL_AXL_ERROR_PROVISTIONING_NEEDED;
103
104     // Enable the Accelerometer axis
105     uint8_t reg_value = LSM901_CTRL_REG5_XL_ZEN_XL | LSM901_CTRL_REG5_XL_YEN_XL | LSM901_CTRL_REG5_XL_XEN_XL;
106     syshal_i2c_write_reg(I2C_AXL, LSM901_AG_ADDR, LSM901_CTRL_REG5_XL, &reg_value, 1);
107
108     // The supported options by the LSM9051 device
109     uint16_t valid_sample_rate_options[6] = {10, 50, 119, 238, 476, 952}; // Number of readings per second
110
111     // Convert the configuration tag options to the closest supported option
112     uint16_t sample_rate = find_closest_value_priv(sys_config.axl_sample_rate.contents.sample_rate, valid_sample_rate_options, sizeof(valid_sample_rate_options) / sizeof(*valid_sample_rate_options));
113
114     // Populate the accelerometer CTRL_REG6_XL register
115     reg_value = 0;

```

Ilustración 30. Programación de acelerómetro 1

```

117     switch (sample_rate)
118     {
119         case 10:
120             DEBUG_PR_TRACE("AXL sample rate: 10 Hz");
121             reg_value |= LSM901_CTRL_REG6_XL_ODR_XL_10HZ;
122             break;
123         case 50:
124             DEBUG_PR_TRACE("AXL sample rate: 50 Hz");
125             reg_value |= LSM901_CTRL_REG6_XL_ODR_XL_50HZ;
126             break;
127         case 119:
128             DEBUG_PR_TRACE("AXL sample rate: 119 Hz");
129             reg_value |= LSM901_CTRL_REG6_XL_ODR_XL_119HZ;
130             break;
131         case 238:
132             DEBUG_PR_TRACE("AXL sample rate: 238 Hz");
133             reg_value |= LSM901_CTRL_REG6_XL_ODR_XL_238HZ;
134             break;
135         case 476:
136             DEBUG_PR_TRACE("AXL sample rate: 476 Hz");
137             reg_value |= LSM901_CTRL_REG6_XL_ODR_XL_476HZ;
138             break;
139         case 952:
140             DEBUG_PR_TRACE("AXL sample rate: 952 Hz");
141             reg_value |= LSM901_CTRL_REG6_XL_ODR_XL_952HZ;
142             break;
143     }
144

```

Ilustración 31. Programación de acelerómetro 2

También podemos ver el funcionamiento cuando tiene un dato para leer, “despierta” el acelerómetro para la lectura:

```
189  /**
190   * @brief   Makes the accelerometer from a sleep state
191   *
192   * @return  SYSHAL_AXL_NO_ERROR on success
193   */
194  int syshal_axl_wake(void)
195  {
196      DEBUG_PR_TRACE("waking AXL");
197      syshal_i2c_write_reg(I2C_AXL, LSM901_AG_ADDR, LSM901_CTRL_REG6_XL, &LSM901_CTRL_REG6_XL_wake_state, 1);
198      device_aware = true;
199
200      return SYSHAL_AXL_NO_ERROR;
201  }
202
203  bool syshal_axl_aware(void)
204  {
205      return device_aware;
206  }
207
208  int syshal_axl_tick(void)
209  {
210      if (new_data_pending)
211      {
212          // Read the available data from the accelerometer
213          uint8_t temp[6];
214          syshal_axl_data_t accl_data;
215          new_data_pending = false;
216          int bytes_read = syshal_i2c_read_reg(I2C_AXL, LSM901_AG_ADDR, LSM901_OUT_X_L_XL, &temp[0], 6); // Read 6 bytes, beginning at OUT_X_L_XL
217
218          if (bytes_read < 0)
219          {
220              new_data_pending = true;
221              return SYSHAL_AXL_NO_ERROR;
222          }
223
224          accl_data.x = (temp[1] << 8) | temp[0];
225          accl_data.y = (temp[3] << 8) | temp[2];
226          accl_data.z = (temp[5] << 8) | temp[4];
227          syshal_axl_callback(accl_data); // Generate a callback event
228      }
229
230      return SYSHAL_AXL_NO_ERROR;
231  }
```

Ilustración 32. Programación de acelerómetro 3

Podemos observar en el código main que los datos se almacenan en las variables ax1.x, ax1.y y ax1.z desde data.x, data.y y data.z respectivamente:


```

796 void syshal_axl_callback(syshal_axl_data_t data)
797 {
798     // If accelerometer data logging is disabled
799     if (!sys_config.axl_log_enable.contents.enable)
800     {
801         syshal_axl_sleep(); // Sleep the accelerometer device
802         return;
803     }
804
805     if (!sensor_logging_enabled)
806         return;
807
808     switch (sys_config.axl_mode.contents.mode)
809     {
810     case SYS_CONFIG_AXL_MODE_PERIODIC:
811         __NOP(); // NOP required to give the switch case an instruction to jump to
812         logging_axl_xyz_t axl;
813         LOGGING_SET_HDR(&axl, LOGGING_AXL_XYZ);
814         axl.x = data.x;
815         axl.y = data.y;
816         axl.z = data.z;
817         logging_add_to_buffer((uint8_t *) &axl, sizeof(axl));
818         break;
819
820     case SYS_CONFIG_AXL_MODE_TRIGGER_ABOVE:
821         // Calculate vector magnitude
822         __NOP(); // NOP required to give the switch case an instruction to jump to
823         uint16_t magnitude_squared = (data.x * data.x) + (data.y * data.y) + (data.z * data.z); // WARN uint16_t maybe too small to contain true value
824         // Determine if the read data is above the trigger point
825         if (magnitude_squared >= sys_config.axl_g_force_high_threshold.contents.threshold)
826             gps_scheduler_trigger(GPS_SCHEDULER_TRIGGER_ACCEL, true);
827         break;
828     }
829 }

```

Ilustración 33. Programación de acelerómetro 3

- Sensor de presión.

sensor de presión	
logEnable	booleano, habilitar el registro del sensor de presión
tasa de muestra	número, frecuencia de muestreo para los modos PERIÓDICO o DISPARADO
Umbral alto	número, umbral alto (TRIGGERED_BETWEEN o TRIGGERED_ABOVE)
umbral bajo	número, umbral bajo (TRIGGERED_BELOW, TRIGGERED_BETWEEN)
modo	cadena, una de PERIÓDICA o DISPARADA
intervalo de adquisición programado	número, el período de repetición en segundos en el que se inician las adquisiciones para el modo PERIÓDICO, 0=continuamente
tiempo máximo de adquisición	número, el período en segundos durante el cual se realizan las adquisiciones para el modo PERIÓDICO, 0 = indefinidamente

Ilustración 34. Parámetros del sensor de presión

Se puede apreciar el comportamiento del sensor de presión en el código, donde cuando el sensor se desactiva, pone a dormir el sensor, mientras que cuando se activa se actualiza el valor de presión:

```

831 void syshal_pressure_callback(int32_t pressure)
832 {
833     // If pressure logging is disabled
834     if (!sys_config.pressure_sensor_log_enable.contents.enable)
835     {
836         syshal_pressure_sleep(); // Sleep the pressure device
837         return;
838     }
839
840     if (!sensor_logging_enabled)
841         return;
842
843     logging_pressure_t pressure_data;
844     LOGGING_SET_HDR(&pressure_data, LOGGING_PRESSURE);
845     pressure_data.pressure = pressure;
846     logging_add_to_buffer((uint8_t *) &pressure_data, sizeof(pressure_data));
847 }

```

Ilustración 35. Programación del sensor de presión

- Sensor de salinidad.

interruptor de agua salada	
logEnable	booleano, habilitar/deshabilitar el interruptor de agua salada transiciones de eventos sumergidos y superficiales
histéresisPeríodo	período de estabilización requerido en segundos para eventos de interruptor cerrado antirrebote (0 = sin antirrebote)

Ilustración 36. Parámetros del sensor de salinidad

Su uso en la programación es el de saber cuándo la tortuga está en la superficie, principalmente para el ahorro de energía, cuando la tortuga permanece bajo el agua no enviara datos al satélite por lo que todas las funciones relacionadas no son necesarias que se encuentren activas.

```

1002 void saltwater_switch_callback(const syshal_switch_state_t *state)
1003 {
1004     if (*state != SYSHAL_SWITCH_EVENT_OPEN)
1005         return;
1006
1007     if (!sensor_logging_enabled)
1008         return;
1009
1010     // If we're in the operational state and we were previously underwater
1011     if (!tracker_above_water)
1012     {
1013         if (sys_config.saltwater_switch_log_enable.contents.enable &&
1014             sys_config.saltwater_switch_log_enable.hdr.set)
1015         {
1016             logging_surfaced_t surfaced;
1017             LOGGING_SET_HDR(&surfaced, LOGGING_SURFACED);
1018             logging_add_to_buffer((uint8_t *) &surfaced, sizeof(surfaced));
1019         }
1020
1021         if (sys_config.saltwater_switch_hysteresis_period.contents.seconds < SALTWATER_SWITCH_MINIMUM_PERIOD_S)
1022             syshal_timer_set(timer_saltwater_switch_hysteresis, one_shot, SALTWATER_SWITCH_MINIMUM_PERIOD_S);
1023         else
1024             syshal_timer_set(timer_saltwater_switch_hysteresis, one_shot, sys_config.saltwater_switch_hysteresis_period.contents.seconds);
1025
1026         gps_scheduler_trigger(GPS_SCHEDULER_TRIGGER_SALTWATER_SWITCH, true);
1027     }
1028     else
1029     {
1030         if (SYSHAL_TIMER_RUNNING == syshal_timer_running(timer_saltwater_switch_hysteresis))
1031         {
1032             syshal_timer_reset(timer_saltwater_switch_hysteresis);
1033         }
1034     }
1035
1036     tracker_above_water = true;
1037 }

```

Ilustración 37. Programación del sensor de salinidad

- Bluetooth.

Bluetooth	
dirección del dispositivo	El formato debe ser "xx:xx:xx:xx:xx:xx" y los 2 MSB superiores deben establecerse en 1
disparadorControl	"REED_SWITCH", "PROGRAMADO", "GEOFENCE", "ONE_SHOT"
intervaloprogramado	número, período de programación en segundos; 0=continuo
duraciónprogramada	número, duración programada en segundos; 0=indefinido
Intervalo de publicidad	número, expresado en unidades de 0,625 ms
intervalo de conexión	número, expresado en unidades de 1,25 ms
conexiónInactividadTiempo de espera	número, forzar la caída de la conexión después del período de inactividad en segundos
phyMode	1_MBPS", "2_MBPS"
publicidadEtiquetas	cadena, "LAST_GPS_TIME", "BATTERY_LEVEL"
logEnable	booleano, habilitar/deshabilitar el registro de mensajes de registro BLUETOOTH_xxx

Ilustración 38. Parámetros del Bluetooth

Los parámetros de los satélites en el archivo no son los actuales, por lo que se requiere actualizarlos. Para ellos nos dirigimos a la página web de Argos [8].

Donde, una vez nos identifiquemos tenemos acceso a diferentes herramientas, el “view your data” nos servirá para ver los datos recibidos por el satélite, pero ahora necesitamos la herramienta “Satellite pass prediction” en la que, insertando los datos de latitud, longitud y altitud de nuestra posición, nos proporciona un archivo de texto con los datos de los satélites.

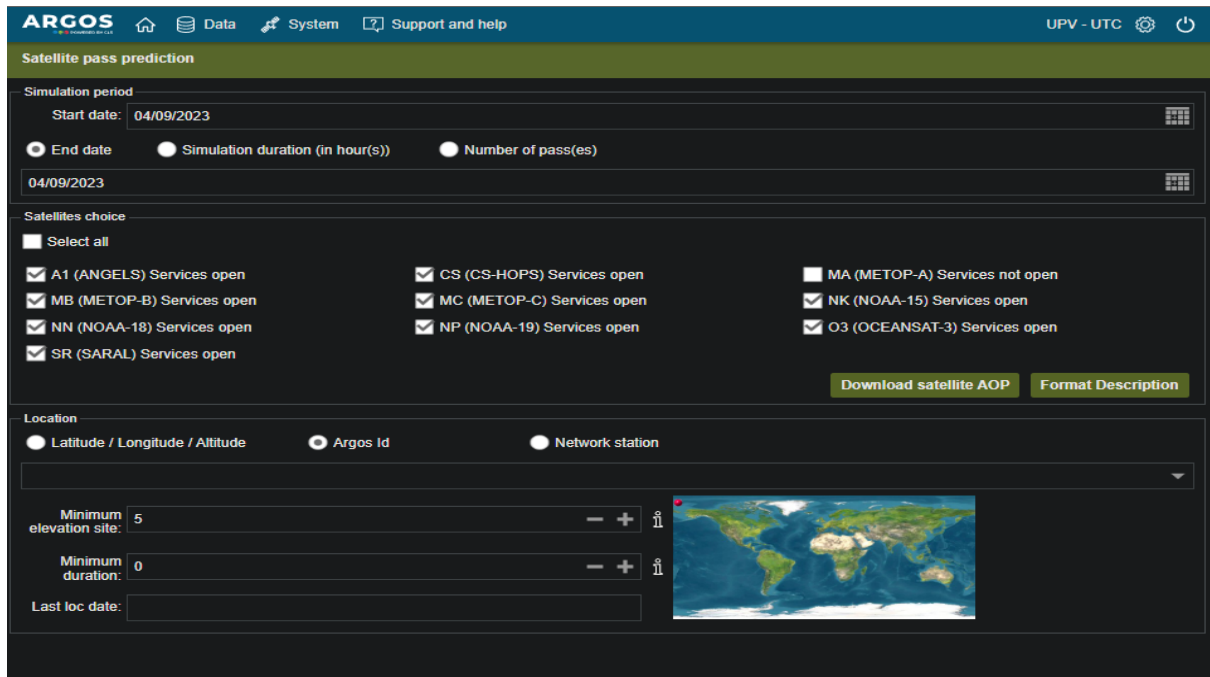


Ilustración 39. Página web Argos [8] Predicción de satélites

Archivo	Editar	Ver
A1	6 0 0 1	2022 8 1 22 45 18 6886.709 97.4729 104.921 -23.729 94.9165 -14.33
MB	9 3 0 0	2022 8 1 22 40 16 7195.596 98.6929 342.433 -25.340 101.3593 0.00
MC	B 7 3 0	2022 8 1 23 33 30 7195.559 98.6805 329.288 -25.340 101.3585 0.00
19	C 6 0 0	2022 8 1 23 8 46 7225.915 99.1431 311.320 -25.497 101.9984 -1.39
SR	D 4 3 0	2022 8 1 22 40 58 7160.161 98.5461 108.148 -25.153 100.6130 -0.65

Ilustración 40. Resultado de la predicción de satélites

Una vez actualizado los valores de los satélites y activado o desactivado los sensores deseados (el dispositivo no puede trabajar con todos ellos activos por lo que hay que ir activando o desactivando según se requiera, sino el dispositivo deja de transmitir) procedemos a enviar el archivo configurado de vuelta al dispositivo con el siguiente comando:

```
sudo tracker_config --write_config default.json
```

Ahora si todo está correcto, una vez desconectado el dispositivo de la Raspberry la luz del dispositivo deberá parpadear varias veces en verde y luego apagarse, lo que nos indicará que está funcionando.

6. Presupuesto

Para la parte del presupuesto se requerirá el trabajo de un ingeniero que programe y prepare el dispositivo su funcionamiento y por si se produjera algún problema y se requiera solución rápida, también requerirá contratar una empresa de diseño en 3D a la que le encargaremos la el diseño y fabricación del encapsulado y para prepararlo con protección para el agua optaremos por emplear resina Crystal Clear 202 EU, sin olvidar el precio del dispositivo que obtendremos a través de la página de Arribada y la compra de una placa solar para poder recargar la batería una vez la tortuga salga a la superficie. Todo lo anterior nos brinda la siguiente tabla de presupuesto estimado:

	Horas	Precio	Total
Mano de obra			
Ingeniero	100	15,38 € /h	1538 €
Diseño 3D	8	25 € /h	200 €
Materiales			
Resina Crystal Clear 202 EU	-	78,20 €	78,20 €
Placa solar	-	12 €	12 €
Dispositivo	-	257,21 €	257,21 €
			2085.41 €

7. Resultados

Una vez colocado el dispositivo cerca de una ventana para que pueda enviar mejor los datos y esperar que el nanosatélite cruce por encima y reciba los datos, volviendo a la página web de Argos, esta vez accedemos a la herramienta para ver los datos recibidos por el satélite, obteniendo la siguiente cuadrícula:

The screenshot shows the ARGOS web interface with search criteria and a table of received data. The search criteria include:

- Selección: By platform model(s)
- ARRIBADA HORIZON DEV KIT
- Localizar y recoger
- Solo loc. clase: GRAMO 3 2 1 0 A B DE
- Periodo de tiempo: For last days and hours
- Todos los mensajes recibidos
- Más importante o mejor en pase de satélite
- Ultimo: 15 Dias
- 2 Horas
- Datos de diagnóstico:
- Filtrar loc. con 1 mensaje:

The table below shows the received data:

Número de identifica...	Plataforma	Prg No.	Pase dur. (s)	Fecha del mensaje	Se sentó.	Alarma	Nombre del formato	Lugar: datos	Longitud	Latitud	Altitud	Bóveda	Veloci...	Loc. calidad	Frecuencia	Largo. 1	Ls
228375		7213	0	07/09/2022 11:06:00	NX	norte	HORIZONTE-CORTO	07/09/2022 11:06:00	0.0992	38.8411	0.000			GRAMO	401650000.00	0.0992	38
228375		7213		07/09/2022 11:10:40	MC	norte	HORIZONTE-CORTO								401650000.00		
228375		7213	0	08/09/2022 10:35:00	NX	norte	HORIZONTE-CORTO	08/09/2022 10:35:00	0.099	38.8407	0.000			GRAMO	401650000.00	0.099	38
228375		7213	74	08/09/2022 10:49:58	MC	norte	HORIZONTE-CORTO								401650000.00		

Ilustración 41. Ejemplo de resultados obtenidos

The screenshot shows a detailed table of received data with the following columns:

major axis	Semi-minor axis	Elipse orientat...	GDOP	Raw data	1 BCH	2 BITFIELD	3 DAY	4 HOUR	4 MINUTES	5 LATITUDE	6 LONGITU...	7 SPEED	8 HEADING	9Z VOLTAGE	9 ALTITUDE	DEFAULT
1053	94	7331	11829D2F8A08007BE0000010641593	00	17	16	10	29	388417	991						
0	0	0	11829D2F8A08007BE0000010641593									00	00	03.93	00	
842	89	7805	11825D2F8A00007C400000109C1FCA	00	17	16	09	29	388416	994						
0	0	0	11825D2F8A00007C400000109C1FCA									00	00	03.96	00	
0	28	77	6825	11815E2F8C10007E800000109DF198	00	17	16	05	30	388482	1012					
0	0	0	11815E2F8C10007E800000109DF198									00	00	03.96	00	
0	28	77	6825	1181402F8E88007C600000109D35EC	00	17	16	05	00	388563	995					
0	0	0	1181402F8E88007C600000109D35EC									00	00	03.96	00	
1113	96	4577	117D5D2F8C8007C400000108FB701	00	17	15	21	29	388409	994						
0	0	0	117D5D2F8C8007C400000108FB701									00	00	03.99	00	

Ilustración 42. Ejemplo de resultados obtenidos 2

En la cuadrícula observamos la fecha y hora en que se reciben los datos y los datos recibidos de los sensores, como también el dato completo. Se observa que se transmiten en dos tramos de código diferente.

De la parte de búsqueda de información se puede concluir el aumento de proyectos en los que están implicadas las constelaciones de nanosatélites, la gran mayoría para el beneficio ambiental del planeta y protección animal.

Se ha comprobado que se reciben bastantes datos en un día, debido a la mayor cantidad de nanosatélites circulando. También nos hemos familiarizado más con la web Argos que se emplea para la observación de los datos que envía el dispositivo y nos ofrece un gran número de información tanto de los satélites como de datos.

En la tabla que nos ofrece podemos observar la frecuencia a la que se trabaja, la posición del dispositivo, tanto en latitud, longitud como altitud. También tanto secciones donde informa de los datos de los sensores como el dato completo sin decodificar y el valor de la batería lo que nos permite ver el consumo de esta.

8. Conclusiones y futuros trabajos

Para la puesta en funcionamiento del dispositivo se requerirá más tiempo puesto que no se han podido comprobar los resultados aportados por los sensores.

Se quería comprobar el funcionamiento de los sensores en el nuevo dispositivo, por desgracia no se ha conseguido. Pero hemos localizado las partes del código donde se emplea los datos y variables donde se guardan, con más tiempo, podremos encontrar como obtener la información de los sensores y las condiciones para programarlos.

Por parte de los ODS, hemos visto que nuestro proyecto cumple la mayoría de los objetivos de desarrollo sostenible y como se ha visto en el apartado de información, respecto a diferentes empresas y en concreto Argos [\[8\]](#), que es con la que trabajamos, cada vez aparecen más proyectos destinados a mejorar la conservación marina.

En el apartado de las constelaciones de nanosatélites se ha comprobado el aumento progresivo, y que sigue creciendo con cada nuevo satélite añadido a la red ya existente, de la cobertura proporcionada por la red. Con el proyecto Argos-4 se observa la implicación en la mejora de la conservación natural, que pretende mejorar la información de datos obtenida en los proyectos anteriores (Argos-2 y Argos-3).

Con este proyecto se ha logrado una mejor comprensión del código de funcionamiento de Horizon y su modo de trabajo a la hora de captación y análisis de datos, así como de la importancia de dichos datos en la preservación de la fauna marina. Para futuros trabajos esperamos poder ver el dispositivo en funcionamiento equipado en una tortuga y los resultados que podamos obtener de dicho proyecto.

9. Bibliografía

- [1] <https://intemares.es/>
- [2] <https://es.wikipedia.org/wiki/CubeSat>
- [3] <https://alen.space/es/guia-basica-nanosatelites/>
- [4] <https://eos.com/es/blog/constelaciones-de-satelites/#:~:text=Las%20constelaciones%20de%20sat%C3%A9lites%20m%C3%A1s,GlobalStar%2C%20RapidEye%20y%20otros%20operadores.>
- [5] <https://alen.space>
- [6] <https://sateliot.space/en>
- [7] <https://arribada.org/>
- [8] <https://www.argos-system.org/argos/who-we-are/international-cooperation/>
- [9] <https://www.kineis.com/en/what-is-the-kineis-connectivity/>
- [10] <https://raspberrypi.cl/raspberry-pi-3b-2/>
- [11] <https://www.innovaspain.com/sateliot-lanzamiento-nanosatelites/>
- [12] <https://www.lavanguardia.com/vida/20230410/8886104/sateliot-lanza-manana-nanosatelite-estandar-5g.html>
- [13] <https://www.elperiodico.com/es/economia/20220202/indra-cellnex-entran-capital-sateliot-13183647>
- [14] https://cincodias.elpais.com/cincodias/2020/08/13/companias/1597324011_571838.html
- [15] <https://www.elconfidencialdigital.com/articulo/defensa/defensa-pondra-orbita-minisatelites-controlar-cambio-climatico/20190620202340126978.html>
- [16] <https://alen.space/es/alen-space-colabora-con-el-inta-en-su-proyecto-anser/>
- [17] https://www.inta.es/INTA/es/blogs/copernicus/BlogEntry_1663327684152#
- [18] <https://www.cls-telemetry.com/argos-solutions/kineis/>
- [19] <https://presse.cnes.fr/en/argos-4-enters-operational-service-outstanding-cnes-international-collaboration>
- [20] <https://www.kineis.com/en/faqs/>
- [21] <https://kartenspace.com/nanosatellites/>
- [22] <https://argos-system.cls.fr/argos-cwi2/login.html>

10. ANEXO I. Relación del trabajo con los Objetivos de Desarrollo Sostenible de la Agenda 2030

Anexo al Trabajo de Fin de Grado y Trabajo de Fin de Máster

Relación del trabajo con los Objetivos de Desarrollo Sostenible de la agenda 2030

Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS)

Objetivos de Desarrollo Sostenibles	Alto	Medio	Bajo	No procede
ODS 1. Fin de la pobreza				
ODS 2. Hambre cero				
ODS 3. Salud y bienestar				
ODS 4. Educación de calidad				
ODS 5. Igualdad de género				
ODS 6. Agua limpia y saneamiento	X			
ODS 7. Energía asequible y no contaminante				
ODS 8. Trabajo decente y crecimiento económico		X		
ODS 9. Industria, innovación e infraestructuras		X		
ODS 10. Reducción de las desigualdades				
ODS 11. Ciudades y comunidades sostenibles				
ODS 12. Producción y consumo responsables		X		
ODS 13. Acción por el clima		X		
ODS 14. Vida submarina	X			
ODS 15. Vida de ecosistemas terrestres				
ODS 16. Paz, justicia e instituciones sólidas				
ODS 17. Alianzas para lograr objetivos				

Descripción de la alineación del TFG/TFM con los ODS con un grado de relación más alto

***Utilice tantas páginas como sea necesario.