



Universidad  
Rey Juan Carlos

Escuela Superior de Ciencias Experimentales y Tecnología

GRADO EN INGENIERÍA  
DE TECNOLOGIAS INDUSTRIALES

Trabajo de Fin de Grado

DISEÑO DE SISTEMA DE ODOMETRÍA  
BASADO EN OPEN SOURCE PARA  
RALLY RAID DE REGULARIDAD  
AMATEUR

Álvaro Sanz Canet

Directores: Santiago Murano y Rubén Nieto Capuchino

Curso Académico 2023/24



## **Grado en ingeniería de Tecnologías Industriales**

### **Trabajo de Fin de Grado**

El presente trabajo, titulado **DISEÑO DE SISTEMA DE ODOMETRÍA BASADO EN OPEN SOURCE PARA RALLY RAID DE REGULARIDAD AMATEUR**, constituye la memoria correspondiente a la asignatura Trabajo de Fin de Grado que presenta D. **ÁLVARO SANZ CANET** como parte de su formación para aspirar al Título de Graduado en Ingeniería de Tecnologías Industriales. Este trabajo ha sido realizado en la **Escuela Superior de Ciencias Experimentales y Tecnologías** en el **Departamento de Tecnología Electrónica** bajo la dirección de **Santiago Murano y Rubén Nieto Capuchino**.

Móstoles, **05 de marzo de 2024**.

## 1. ÍNDICE:

1.	ÍNDICE: .....	1
2.	Resumen .....	5
3.	Introducción .....	6
4.	Objetivos .....	12
5.	Solución Técnica / Resultados .....	13
5.1.	Estudio de mercado .....	13
5.2.	Diseño conceptual .....	14
5.3.	Planificación. Diagrama de Gantt .....	16
5.4.	Requerimientos técnicos .....	17
5.5.	Hardware .....	19
5.6.	Errores de precisión y simplificaciones .....	29
5.7.	Desarrollo software .....	35
5.8.	Proceso de fabricación .....	47
5.9.	Costes de fabricación .....	57
5.10.	Consumo eléctrico .....	59
5.11.	Limitaciones .....	60
5.12.	Mejoras futuras .....	61
6.	Conclusiones .....	62
7.	Trabajos futuros .....	63
8.	Bibliografía .....	64
9.	Apéndices .....	66
	Apéndice A: Repositorio GitHub .....	66
	Apéndice B: Diagrama de Gantt ampliado .....	67
	Apéndice C: Fichas técnicas .....	68
	Apéndice D: Desglose de precios a fecha de entrega del proyecto .....	69

## ÍNDICE DE FIGURAS:

Figura 1. Viñetas de roadbook tipo. Fuente: chasingwaypoints.com.....	7
Figura 2. Cuenta kilómetros de bicicleta. Fuente: decathlon.es .....	8
Figura 3. Equipo sofisticado de medición mediante sondas. Fuente: Terratrip.com.....	8
Figura 4. Equipo sofisticado de medición GPS. Fuente: Terratrip.com.....	9
Figura 5. Accesorios Terratrip. Fuente: Terratrip.com.....	9
Figura 6. Kit de Navegación Motocicleta aficionado. Fuente: f2r.pt.....	10
Figura 7. Situación del mercado en 2024. Elaboración propia.....	13
Figura 8. Disposición de las pantallas. Elaboración propia.....	14
Figura 9. Pulsadores del dispositivo. Elaboración propia. ....	15
Figura 10. Funcionamiento conceptual del dispositivo. ....	15
Figura 11. Diagrama de Gantt. Elaboración propia.....	16
Figura 12. Módulo Raspberry Pi Pico. Fuente: raspberrypi.com .....	20
Figura 13. Módulo Arduino Nano. Fuente: arduino.cc .....	21
Figura 14. Módulo ESP-32. Fuente: espressif.com.....	21
Figura 15. Adafruit Ultimate GPS Breakout V3. Fuente: adafruit.com .....	23
Figura 16. Módulo GPS NEO-6M. Fuente: u-blox.com .....	24
Figura 17. Módulo HT1621. Fuente: aliexpress.com.....	25
Figura 18. Módulo TM1637 de 4 dígitos. Fuente: aliexpress.com.....	25
Figura 19. Módulo TM1637 de 6 dígitos. Fuente: aliexpress.com.....	26
Figura 20. Módulo LCD1602. Fuente: aliexpress.com .....	26
Figura 21. Módulo transformador LM2596. Fuente: aliexpress.com.....	29
Figura 22. Distancia recorrida por unidad de latitud. Elaboración propia.....	31
Figura 23. Cálculo trigonométrico del radio del casquete esférico. Elaboración propia.....	31
Figura 24. Radio medio según el modelo WGS 84. Fuente: Wikipedia.org .....	32
Figura 25. Diagrama en bloques del dispositivo a diseñar. Elaboración propia.....	36
Figura 26. Estructura de bits del protocolo UART. Fuente: rohde-schwarz.com .....	37
Figura 27. Mensajes suministrados por el GPS. Fuente: wavesshare.com.....	37
Figura 28. Estructura de un mensaje NMEA. Fuente: MathWorks.com.....	37
Figura 29. Mensaje tipo GPRMC. Fuente: MathWorks.com.....	38
Figura 30. Estructura del bus utilizado por los módulos tm_1637. Fuente: Datasheet del módulo tm_1637.....	39
Figura 31. Comandos del bus de los módulos tm_1637. Fuente: Datasheet del módulo tm_1637.....	39

Figura 32. Animación de encendido. Elaboración propia. ....	40
Figura 33. Mensaje de búsqueda de satélites por pantalla. Elaboración propia. ....	40
Figura 34. Dispositivo en proceso de conexión con los satélites. Elaboración propia. ....	41
Figura 35. Funcionalidades de los pulsadores. Elaboración propia. ....	42
Figura 36. Diagrama de flujo del programa principal main. Elaboración propia. ....	44
Figura 37. Configuración de los bits de siete segmentos. Elaboración propia. ....	45
Figura 38. Diagrama de flujo del script distance_GPS. Elaboración propia. ....	46
Figura 39. Placa de prototipos tipo protoboard. Fuente: Mexbit.com. ....	47
Figura 40. Prueba unitaria de una pantalla de 7 segmentos. Elaboración propia. ....	48
Figura 41. Montaje alimentado por el microcontrolador. Elaboración propia. ....	48
Figura 42. Montaje alimentado por una fuente externa, Elaboración propia. ....	49
Figura 43. Plano de conexiones eléctricas. Elaboración propia. ....	50
Figura 44. Se requiere de cobertura para realizar pruebas al dispositivo. Elaboración propia. ....	51
Figura 45. Placa soldada del microcontrolador. Elaboración propia. ....	52
Figura 46. Conexiones soldadas del microcontrolador con los terminales. Elaboración propia. ....	52
Figura 47. Disposición del microcontrolador y las pantallas. Elaboración propia. ....	53
Figura 48. Conexiones de las pantallas. Elaboración propia. ....	53
Figura 49. Potenciómetro ajustable del módulo LM2596. ....	54
Figura 50. Conectores S13. Elaboración propia. ....	54
Figura 51. Conector de la antena externa. Elaboración propia. ....	55
Figura 52. Cable y conector de la botonera. Elaboración propia. ....	55
Figura 53. Dispositivo final. Elaboración propia. ....	56
Figura 54. Posicionamiento del dispositivo final en el mercado. Elaboración propia. ....	62
Figura 55. Repositorio del código Python. ....	66
Figura 56. Diagrama de Gantt ampliado. ....	67

## ÍNDICE DE TABLAS:

Tabla 1. Tasa de refresco requerida en función de la velocidad.....	17
Tabla 2. Especificaciones de los microcontroladores.....	22
Tabla 3. Matriz de decisión de microcontroladores. ....	22
Tabla 4. Especificaciones de los módulos GPS.....	24
Tabla 5. Especificaciones de las pantallas.....	27
Tabla 6. Matriz de decisión de pantallas. ....	27
Tabla 7. Consumo de corriente aguas abajo del convertor de tensión. ....	28
Tabla 8. Desglose de costes de mano de obra en diciembre de 2023.....	57
Tabla 9. Desglose de precios en diciembre de 2023 .....	58
Tabla 10. Coste de los dispositivos disponibles .....	58
Tabla 11. Potencia consumida.....	59
Tabla 12. Desglose de precios unitarios .....	69

©2024 Álvaro Sanz Canet

Algunos derechos reservados

Este documento se distribuye bajo la licencia

“Atribución-CompartirIgual 4.0 Internacional” de Creative Commons,

disponible en

<https://creativecommons.org/licenses/by/4.0/deed.es>

## 2. Resumen

Las competiciones de raid de regularidad de clásicos para aficionados están ganando popularidad en los últimos años. Son competiciones de fácil acceso orientadas a personas interesadas en participar en las competiciones en el ámbito del motor es su tiempo libre que no requiere un elevado presupuesto ni nivel de preparación. Se trata de una competición basada en la orientación conduciendo a través de un recorrido preestablecido por caminos de tierra grandes distancias. Para ello se hace uso de un roadbook. Un roadbook es un rollo largo de papel enrollado donde mediante viñetas con dibujos e indicaciones de distancias se indica por donde transcurre el recorrido cruce a cruce. El reto es poder leer estas indicaciones y no perderse durante el transcurso de la prueba. Para la medición de distancias se hace uso de un odómetro y hay distintas soluciones para ello. El coste de la inscripción y el vehículo son reducidos pero las soluciones de odometría disponibles en el mercado son una barrera de entrada considerable.

Actualmente no existen dispositivos accesibles en un punto intermedio entre las soluciones muy sencillas y poco eficaces, como odómetros de bicicletas, y las soluciones profesionales muy costosas y complejas que exceden las necesidades y capacidades de los principiantes. Es por ellos, que en este Trabajo de Fin de Grado se va a diseñar y fabricar una solución accesible basada en Open Hardware que permita el acceso libre a los archivos e indicaciones necesarias para replicar el dispositivo desarrollado.

Este dispositivo tendrá como premisas una fácil utilización y un bajo coste que se situará por debajo de los 50 euros. Hará uso de tecnología GPS para la medición de distancias que serán procesadas por un microcontrolador para ser mostradas al usuario por tres pantallas de 7 segmentos. Este dispositivo será autónomo, solo necesitando de alimentación por parte del vehículo.

Debido a estas premisas se ha requerido del diseño de alternativas a los sistemas clásicos de medición de distancias entre coordenadas para permitir el procesamiento rápido de cálculos complejos con procesadores de menor rendimiento, así como otros desafíos de diseño.

El dispositivo final permite una precisión de diez metros en distancias inferiores a 500 metros requiriendo de su ajuste periódicamente en las viñetas correspondientes como es común en este tipo de competiciones.

La solución se ubicará en un repositorio abierto al público junto con las indicaciones suficientes para la elaboración de un dispositivo idéntico al fabricado en este proyecto y está prevista la actualización por parte del creador y de la comunidad. El acceso a este repositorio se encuentra en el (Apéndice A) de este mismo documento.

### 3. Introducción

En el mundo solo en 2022 se vendieron más de setenta y nueve millones de coches (1) . Gran parte de estos vehículos son utilizados con fines prácticos como efectuar desplazamientos o transportar carga. Pero además de la practicidad algunas personas ven en los coches una forma de entretenimiento, tanto en el disfrute como en la competición.

Cuando se piensa en la competición con vehículos es común pensar primeramente en competiciones como la Formula 1, el Paris Dakar o las carreras de resistencia de Le Mans. Estas competiciones mueven millones de euros y participar en ellas absolutamente inalcanzable para prácticamente todas las personas. Incluso en sus categorías inferiores requieren de cientos de miles de euros, mucho talento y a una dedicación completa a la competición. Esto está fuera del alcance de cualquiera que considere competir a nivel recreativo.

Todo asociado al mundo del automóvil de por si requiere un desembolso de capital importante. Su adquisición, la compra de combustible y el mantenimiento del vehículo suponen un gasto de miles de euros anuales. En 2023 el precio del coche nuevo más barato del mercado se encuentra en España por encima de los 10.000 euros (2). En vehículos con mejores prestaciones con aspiraciones más deportivas este valor de compra llega a quintuplicarse incluso en los vehículos de acceso a la gama.

Debido a esta barrera inicial, los aficionados que consideran la competición como un entretenimiento han creado a lo largo de los años pruebas de inscripción libre que son mucho más accesibles. Son pruebas individuales o campeonatos cortos para aficionados que tiene como premisa mantener el coste lo más bajo posible.

Una de las pruebas más accesibles hoy en día, son los rally raid de regularidad. Estas pruebas discurren a baja velocidad y su objetivo no es llegar primero sino obtener el mayor número de puntos por navegación y precisión. Disponen de autorizaciones y permisos para circular a través de caminos forestales y debido a sus bajas velocidades permiten el menor uso de medidas de seguridad, como la no obligatoriedad de barras antivuelco en los vehículos, y al no requerir de circuitos, comisarios y demás elementos deportivos bajan mucho los costes de inscripción.

Por estas y otras razones estas competiciones de raid de regularidad de clásicos han ganado popularidad en los últimos 10 años en España, aunque siguen siendo un nicho pequeño. El espíritu de esta competición es obtener un coche de más de 25 años, por lo general a un precio de alrededor de mil euros, y seguir una hoja de ruta (Roadbook) por caminos de tierra y carreteras abiertas al público. Se realiza a



velocidades legales y respetando las normas de circulación. El copiloto ha de interpretar el roadbook con viñetas y distancias impresas en él, como las que se presentan en la figura 1. Para ello, se hace uso de odómetros especialmente diseñados para medir la distancia recorrida e ir guiando al piloto a través de las intersecciones y rumbos marcados por el roadbook. Estos dispositivos aportan información al usuario como la distancia recorrida mediante dos contadores que denominaremos (trips) que reinicia el copiloto para saber la distancia recorrida entre las intersecciones y ubicarse. En función de la capacidad del dispositivo también aporta esta información sobre la orientación, la velocidad y herramientas especiales como cronómetros para pruebas específicas.

5.45	↑ ! MOY RIO Watch for water	CAP 299°
6 0.13	↑	
6.47	↑ TDSPP	CAP 272°
7 1.02	↑	
7.08	↑ FINISH ASS	CAP 104°
8 0.60	↑	

Figura 1. Viñetas de roadbook tipo. Fuente: chasingwaypoints.com

Las soluciones comerciales a estos odómetros se dividen en dos grupos principales: en dispositivos que hacen uso de GPS para sus mediciones y en dispositivos que hacen uso de sondas e imanes en las ruedas para medir revoluciones y transformarlas en distancias. En coches, las soluciones utilizadas habitualmente son:

**Contador de vueltas de bicicletas:** Estos dispositivos son muy poco precisos ya que no están pensados para las velocidades alcanzadas por un vehículo, ni tampoco las distancias recorridas por los mismos. Se basan en sensores magnéticos colocados en las ruedas y cuentan las revoluciones dadas por la rueda. La colocación de los imanes, la proyección de grava, las trazadas irregulares del vehículo y la simplicidad de estos equipos, hace que sea una solución muy poco óptima. Sumado a esto no es posible el ajuste de distancias de manera sencilla y requiere de habilidad para solventar estos problemas. Su precio se sitúa en torno a los 15 euros, pero requiere de una instalación de la sonda y unos soportes en el propio vehículo que quedan expuestos a golpes y suciedad.

Ejemplo: Cuentakilómetros Bicicleta Van Rysel 100 Negro Con Cable.

Precio 11.99 €.



Figura 2. Cuenta kilómetros de bicicleta. Fuente: decathlon.es

**Equipos sofisticados de competición mediante sondas:** Estos equipos disponen de todas las funcionalidades necesarias para un rally de regularidad como cronómetros, tablas de velocidades y también otros modos que no son utilizados en las competiciones para aficionados. Siguen necesitando de sondas para medir las distancias, aunque son más elaboradas que en los cuenta vueltas de bicicleta. Miden en ambas ruedas y tienen un procesamiento mucho más preciso a altas velocidades. Su utilización es complicada y requiere de conocimiento y habilidad para su uso.

Ejemplo: Terratrip 303 Road Survey PLUS V4.

Precio 284.95 €.



Figura 3. Equipo sofisticado de medición mediante sondas. Fuente: Terratrip.com

**Equipos sofisticados de competición mediante GPS:** Se trata de equipos muy sofisticados diseñados para competiciones de alto nivel. Cuentan con muchas funcionalidades innecesarias para las competiciones de iniciación como cronómetros, tablas de velocidades y otros modos que no son utilizados en las competiciones para aficionados. Tienen precios elevados, con un precio medio de quinientos euros. Su utilización es complicada y requiere de conocimiento y habilidad para su uso. Estos integran un GPS muy preciso que permite su uso en lugares recónditos del planeta en condiciones adversas.

Ejemplo: Terratrip 202 GeoTrip with GPS & GLONASS V5.

Precio 422.95 €.



Figura 4. Equipo sofisticado de medición GPS. Fuente: Terratrip.com

Añadido a estos dispositivos hay que incluir botoneras, soportes y cables de alimentación además de las sondas en el caso de que las tenga. La suma de estos accesorios supera los ciento cincuenta euros y encarece aún más el producto completo.



Figura 5. Accesorios Terratrip. Fuente: Terratrip.com

El hecho de no disponer de una solución intermedia debido al coste de los equipos y a la dificultad de operación supone una barrera de entrada en estas competiciones para aficionados inexpertos o de bajo poder adquisitivo.

En cuanto a la situación del mercado en motocicletas, las barreras de entrada son aún mayores. El hecho de no disponer de copiloto para operar el dispositivo, la imposibilidad de soltar las manos del manillar y la exposición a las inclemencias meteorológicas requieren de dispositivos complejos y resistentes para la competición a alto nivel. Estos dispositivos integran el soporte, el porta roadbook y los mandos para adaptar a cada vehículo en particular y son más caros que los de los coches. Su precio está por encima de los mil euros y al existir una menor demanda, no hay alternativas económicas integradas.

Ejemplo: Kit de Navegación HOBBY F2r

Precio 913.00 €.



Figura 6. Kit de Navegación Motocicleta aficionado. Fuente: f2r.pt

Los odómetros que recurren al GPS para medir distancias son más fáciles de instalar en un vehículo y más intuitivo de usar. El cálculo de distancias entre dos puntos sobre un plano es relativamente sencillo de medir utilizando trigonometría básica, pero se vuelve más complejo cuando se trata de puntos sobre la superficie de una esfera. Para medir distancias sobre la superficie de la Tierra se hace uso de la Formula de Haversine (3).

Tras un desarrollo matemático, la distancia entre dos coordenadas puede expresarse como:

$$d = 2r \arcsin \left( \sqrt{\sin^2 \left( \frac{\phi_2 - \phi_1}{2} \right) + \cos(\phi_1) \cos(\phi_2) \sin^2 \left( \frac{\lambda_2 - \lambda_1}{2} \right)} \right) \quad (2)$$

Siendo:

$r$  = radio de la tierra expresado en metros.

$\Phi_1, \Phi_2$  = Valores de Latitud en radianes.

$\lambda_1, \lambda_2$  = Valores de Longitud en radianes.

En función de la problemática planteada, en este TFG se propone el diseño de una solución intermedia entre estos dos tipos de odómetros que permita otorgue a los aficionados una solución accesible para iniciarse en este tipo de competiciones de manera sencilla y asequible, manteniendo algunas de las funcionalidades de los equipos más complejos y dejando de lado otras que no son relevantes para un dispositivo de iniciación.

Este documento está estructurado en siete capítulos. En el primero se encuentra el resumen, luego continua la introducción donde se describe la situación actual de los raids de regularidad para aficionados y la necesidad de una solución al problema de la accesibilidad. Después se encuentra el apartado de objetivos en el que se acotan los resultados que se pretende obtener. Continúa el apartado de solución técnica donde se encuentra toda la parte de análisis, diseño y fabricación del dispositivo. Posteriormente hay un capítulo de conclusiones donde se revisa hasta qué punto se han cubierto los objetivos iniciales. Por último, aparece en los dos últimos capítulos la bibliografía y los anexos entre los que se incluye el acceso a un repositorio con los archivos del dispositivo.

## 4. Objetivos

El objetivo general de este trabajo es diseñar y fabricar una alternativa económica, de fácil acceso y de código abierto frente a las soluciones existentes en el mercado para la odometría en los raids de regularidad de aficionados. Se busca que sea suficientemente eficaz para desempeñar su función en etapas de rally y disminuir las barreras de entrada a estas competiciones.

Para ello, se proponen los siguientes objetivos parciales:

- 1) Estudio de mercado.
  - Determinar las funcionalidades que debe satisfacer el dispositivo final.
- 2) Diseño e implementación del dispositivo final.
  - Definir las dimensiones y distribución idóneas.
  - Determinar los requerimientos técnicos de los componentes que componen el dispositivo.
  - Realizar un estudio de los componentes comerciales existentes y valorar distintas soluciones.
  - Desarrollo de los programas software necesarios.
- 3) Pruebas de validación y propuesta de mejoras.
  - Evaluación del consumo energético del dispositivo y de el posible impacto en el consumo energético del vehículo.
  - Estudio de los costes de producción del dispositivo.
  - Realización de pruebas finales para comprobar su eficacia en condiciones de operación.
- 4) Publicación y conceder acceso público al dispositivo.
  - Creación y desarrollo de un repositorio

Durante el proceso se esperan adquirir las siguientes competencias:

- Familiarizarse con entornos de programación y lenguajes ajenos a los impartidos en el grado.
- Adquirir conocimientos sobre microcontroladores comerciales y sus posibles aplicaciones.
- Entendimiento de los protocolos de comunicación serial entre dispositivos.
- Uso de matrices de decisión para la elección de los componentes.
- Planificación y ejecución de principio a fin del desarrollo de un producto.
- Desarrollo de habilidades en creación de placas de prototipos y circuitos y fabricación de prototipos.
- Manejo de errores directos e indirectos.

## 5. Solución Técnica / Resultados

### 5.1. Estudio de mercado

El mercado de estos dispositivos queda segmentado en dos grandes grupos. Los aparatos de alto valor económico con un desempeño muy alto y los contadores de bicicleta con un precio y prestaciones muy lejos de estos anteriores. En la Figura 7 se muestra una visión global del mercado. Se encuentran grandes huecos sin oferta de productos que se han numerado del uno al tres.

- Un producto en la zona roja (2) sería un fracaso. Se trataría de productos que con mayor precio ofrecerían menos prestaciones que sus competidores. En esta zona no debería plantearse la creación de un nuevo producto.
- Un producto en la zona verde (1) ofrecería una solución más económica que los dispositivos profesionales que podría suplir la falta de precisión de los dispositivos de bicicleta. En esta zona es donde sería más interesante la creación de un producto nuevo. En este mercado dada la diferencia de precios y prestaciones existen varios posicionamientos posibles en cuanto a coste y prestaciones, buscando idealmente el mayor desempeño al menos coste.
- Finalmente, la zona amarilla (3) sería una zona que quitaría mercado a los aparatos de competición disminuyendo su precio, pero igualando su desempeño. Esta zona sería ideal, pero el coste de los materiales para alcanzar ese nivel de precisión actualmente hace difícil el posicionamiento del producto en esa franja bajando los costes.

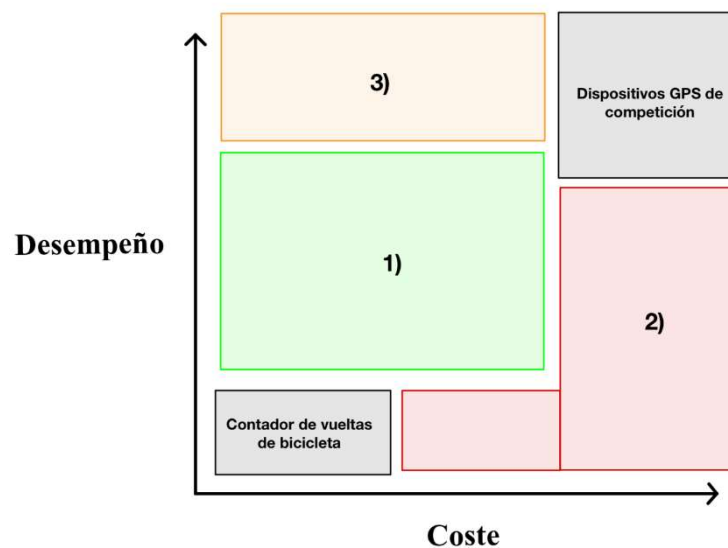


Figura 7. Situación del mercado en 2024. Elaboración propia.

El dispositivo que se busca desarrollar tiene como objetivo ser una alternativa de bajo coste y de fácil acceso por lo que se ubicará más en el lado izquierdo de la zona verde de la figura anterior.

## 5.2. Diseño conceptual

Primeramente, se van a formar una imagen conceptual de los elementos principales y la estructura del dispositivo. En cuanto a la estructura, se busca un dispositivo completo que solo requiera de conectarse a la batería del vehículo para funcionar. Ha de ser un aparato ligero de tamaño contenido, de alrededor de 15cm en su lado más largo. Este mostrará tres números por pantalla al usuario. Estos números serán la distancia total desde el inicio, la distancia parcial desde la última medición y una medición auxiliar para datos de velocidad u orientación.



*Figura 8. Disposición de las pantallas. Elaboración propia.*

Las pantallas deberán ser de un tamaño suficiente para su buena visualización. A su derecha aparecerán dos leds adicionales para indicar parámetros de funcionamiento. Como inputs al dispositivo se instalarán seis pulsadores. Tres de ellos estarán destinados a ajustar los metros recorridos al llegar a una viñeta del roadbook.

Debido a que las distancias medidas con el dispositivo y las medidas indicadas en el papel pueden no cuadrar, encada viñeta se suele ajustar para que ambos valores coincidan. Dos de los botones subirán y bajarán la medición del trip total 10 metros respectivamente mientras que el tercer botón pondrá a cero el medidor parcial. Estos botones deben de ser accesibles fácilmente para el usuario ya que los usará muchas veces a lo largo del recorrido.

Dos de los botones restantes se destinarán para poder elegir entre los modos de la pantalla de accesorios y se reservarán para posibles funcionalidades futuras.



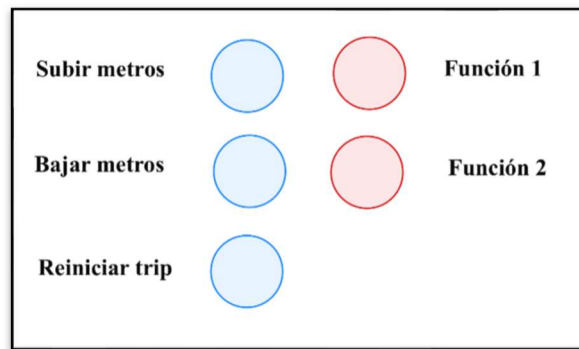


Figura 9. Pulsadores del dispositivo. Elaboración propia.

Internamente este dispositivo utilizará GPS para obtener los datos de posición y así calcular la distancia recorrida. Esto requiere de un módulo GPS con su respectiva antena. Este módulo por sí solo no es capaz de comunicarse con las pantallas y correr el programa así que se requerirá también de un microcontrolador que reciba los datos del GPS, realice los cálculos, ejecute las ordenes indicadas por los botones y gobierne las pantallas.

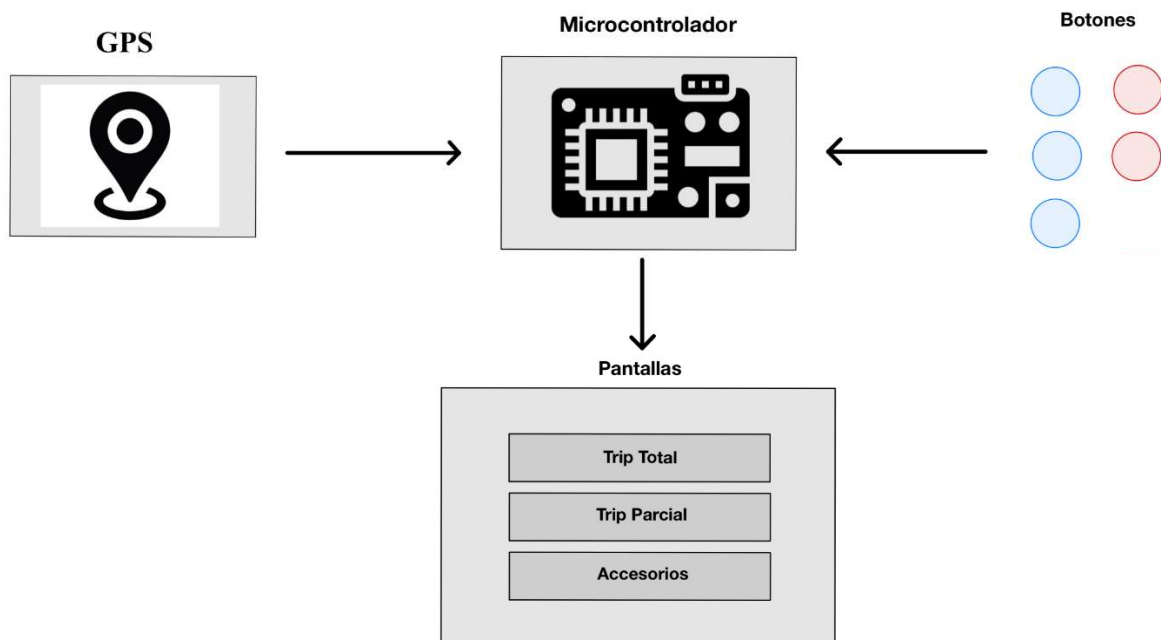


Figura 10. Funcionamiento conceptual del dispositivo.

Por último, se requerirá de un convertor de tensión que adecue la tensión de continua de 12 Voltios de la batería del vehículo a la tensión requerida por los módulos elegidos.



## 5.4. Requerimientos técnicos

### Requerimientos técnicos del receptor GPS.

A la hora de seleccionar un receptor GPS se han de considerar dos factores fundamentales: la precisión y la tasa de refresco de la información. Ha de buscarse un equilibrio entre ambos factores para satisfacer las necesidades de diseño. No sirve de nada una gran precisión en la posición si cuando llega el siguiente dato de posición ya es demasiado tarde, y viceversa.

Primero se va a estimar cual ha de ser la tasa de refresco de la información para que el dispositivo sea funcional. Para ello, se va a considerar aceptable una tasa de refresco que permita obtener la nueva posición del vehículo antes de que este recorra diez metros desde la última actualización.

La normativa de la Real Federación Española del automovilismo indica que en los raids de regularidad de clásicos no se permite superar una velocidad media superior a los 50 km/h (4). Esto no implica que no se sobrepasen en momentos puntuales esta velocidad, pero es un buen punto de partida.

Tomado como referencia los 50km/h, se avanzarán 13.88 metros por segundo.

$$50 \text{ km/h} = \frac{50 \text{ km}}{1 \text{ h}} * \frac{1000 \text{ m}}{1 \text{ km}} * \frac{1 \text{ h}}{3600 \text{ s}} = 13.88 \text{ m/s} \quad (1)$$

Extrapolando este cálculo a otras velocidades se obtiene la siguiente tabla:

*Tabla 1. Tasa de refresco requerida en función de la velocidad.*

Km/h	m/s	Tasa de refresco requerida (Hz)
10	2.77	0.27
20	5.55	0.55
30	8.33	0.83
40	11.11	1.11
50	13.88	1.39
60	16.66	1.67
70	19.44	1.94
80	22.22	2.22
90	25.00	2.50
100	27.77	2.78
110	30.55	3.10
120	33.33	3.33

Con una frecuencia de 2Hz en la adquisición de datos sería suficiente para satisfacer la obtención de datos antes de recorrer 10 metros hasta 70 km/h. Esta tasa de refresco permite conocer proporcionalmente la información lo suficientemente rápido como para que no sea un problema en las proximidades de una intersección.

En cuanto a la precisión, será necesaria una precisión suficiente para que el error total acumulado del dispositivo se contenga en niveles que permitan su utilización. Las distancias entre viñetas en los raids de regularidad para aficionados varían en función de la dificultad del raid. Generalmente en los raids de iniciación las distancias son pequeñas, del orden de cientos de metros (4). Debido a que en cada viñeta es posible ajustar el parcial se puede eliminar el error acumulado hasta el momento, en estos casos un error grande no supondría un problema. Las distancias máximas recorridas sin referencias no superan los 2 km en estas categorías. Un error de 60 metros al final de esta viñeta sería el máximo admisible para el correcto funcionamiento del dispositivo. Esto requiere que el error respecto de la medida real se sitúe por debajo del 5%.

### **Requerimientos técnicos de las pantallas.**

El dispositivo requiere de dos filas de números encargadas de indicar al usuario la distancia total recorrida y la distancia parcial recorrida desde el último reinicio. Estos parciales deben de indicar la distancia en kilómetros. Debido a que en un kilómetro es una medida demasiado grande para ubicarte con precisión se precisará de decimales en la medición. Dos decimales se considera suficiente precisión debido a que hablamos de posición de intersecciones de caminos y estos son visibles a decenas de metros normalmente.

En muchos casos la elección de pantalla irá condicionada al número de caracteres necesarios. La distancia recorrida en las etapas de los rally raid oscila en función de la categoría, pero suele encontrarse por encima de los 100 kilómetros y siempre por debajo de 1000 kilómetros diarios (5). Esto permite la utilización de tan solo 3 dígitos para contabilizar los kilómetros recorridos. Esto sumado a los dos decimales supone un total de 5 dígitos que deben de aparecer en pantalla.

Esto supone un error de precisión de 10 metros en lo referido a la visualización en pantalla debido a que se trata de una medición digital. Esto no significa que el dispositivo no pueda tener más precisión que esa en sus mediciones, pero no será mostrada al usuario por pantalla.

En cuanto al tamaño, se va a considerar que a partir de 1,5 cm por dígito el tamaño del número es suficientemente grande para su correcta lectura en el interior de un vehículo en movimiento.

Adicionalmente deben de poder mostrarse mensajes en pantalla para informar al usuario del estatus del dispositivo y de posibles errores.

### **Requerimientos del microcontrolador.**

El controlador ha de ser capaz de almacenar, mantener y ejecutar los programas requeridos. Además, debe funcionar de manera autónoma a un ordenador conservando su programa aún si ha estado sin alimentación. Deberá disponer de entradas y salidas digitales y de poder establecer protocolos de comunicación seriales con otros módulos. Ha de poder ser alimentado por un módulo de corriente continua que obtiene su tensión y corriente de una batería de 12V tipo coche o moto.

En cuanto a potencia, ha de ser capaz de realizar los cálculos necesarios (leer la señal GPS, procesarla, procesar las señales de los inputs y mostrarlos en pantalla) antes de que el vehículo recorra diez metros. Esta distancia se tomará como la máxima aceptable antes de requerir un nuevo dato de la posición. En la tabla anterior se calcula que es de 0,5 segundos.

Una vez elegido el controlador se realizará un estudio de consumo para evaluar si es necesario el sobredimensionamiento del sistema de generación eléctrica del vehículo para soportar la nueva carga. Por norma general los microcontroladores no tienen elevados consumos energéticos.

### **Requerimientos de la fuente de alimentación.**

Ha de ser capaz de transformar la corriente continua de una batería de 12 V de coche o moto en la tensión requerida por el controlador y los periféricos. El dimensionamiento de esta se realizará acorde a el estudio de consumo eléctrico del dispositivo una vez elegido el resto de los componentes.

## 5.5. Hardware

Se van a valorar diferentes componentes para el diseño del dispositivo. El coste y su capacidad de alcanzar el desempeño requerido son factores fundamentales para el desarrollo del prototipo, pero también se tendrán en cuenta otros factores en función del componente. Se va a hacer uso de matrices de toma de decisiones para seleccionar los componentes definitivos.

Los principales componentes para la fabricación del dispositivo son:

- Controlador
- Modulo receptor GPS
- Pantalla
- Regulador de voltaje

## Controlador:

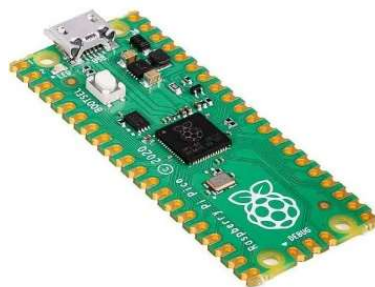
El controlador elegido debe de cumplir dos objetivos principales. Debe de poder realizar los cálculos necesarios para el funcionamiento del dispositivo y debe de poder alojarse en una caja de alrededor de 15 x 15 cm. De otra forma ya no se podría ubicar en los lugares destinados a el dispositivo y lo haría grande y pesado. En estas dimensiones se encuentran los microcontroladores y los pequeños ordenadores de tipo Raspberry Pi.

Además de estos dos factores, la memoria disponible y el número de entradas y salidas digitales y el precio son elementos para tener en cuenta para elegir cual es más adecuado.

Los pequeños ordenadores disponen de potencia de cálculo suficiente para realizar operaciones y scripts e incluso pueden llegar a correr sistemas operativos completos. En cuanto a capacidad de cálculo son claramente superiores. Cuentan también con un mayor número de entradas y salidas digitales y analógicas en función del modelo elegido. El tamaño está al límite de las dimensiones del dispositivo, y son mucho más grandes que sus homónimos microcontroladores. La mayor contraparte de estos ordenadores es su precio. El coste de estos equipos se sitúa alrededor de los 80 € y tiene un mayor consumo energético. Se va a optar por la elección de un microcontrolador debido a su tamaño, precios más contenidos y menor consumo energético siempre y cuando alcance la capacidad computacional requerida por el programa. De no ser posible aun haciendo concesiones realizar el cálculo con microcontroladores se optaría por uno de estos ordenadores.

Se han estudiado la posibilidad de escoger entre tres microcontroladores comerciales y a continuación se exponen sus características:

### - **Raspberry Pi Pico**



*Figura 12. Módulo Raspberry Pi Pico. Fuente: raspberrypi.com*

Se trata de un microcontrolador basado en un chip de 32 bits. Dispone de 26 pines GPIO (General Purpose Input/Output). Internamente es capaz de ejecutar MicroPython, un lenguaje de programación basado en Python aplicado a microcontroladores muy extendido. También tiene posibilidad de

programarse en lenguaje C/C++. Permite una alimentación en el rango de (~1.8V a 5.5V) por medio de un micro USB o a 3.3V por el pin de alimentación. En su versión W tiene conectividad wifi. Su memoria interna es de 2MB. Dispone de mucha documentación en lo correspondiente a su utilización y existe una comunidad grande alrededor de este dispositivo. Su consumo máximo es de 0,47 W en condiciones de máxima carga. (6)

- **Arduino Nano**



*Figura 13. Módulo Arduino Nano. Fuente: arduino.cc*

Se trata de un microcontrolador menos potente que el anterior basado en un procesador de 10 bits. Dispone de 22 pines de los cuales 14 son pines digitales y 8 son analógicos. Su memoria es de 32kB, mucho menor que las de sus competidores. Una ventaja respecto de los otros es que opera en un rango de alimentación de entre 7V y 12V, lo que permitiría no requerir de un convertidor de voltaje. Se programa mediante Arduino IDE, una plataforma propia de Arduino muy intuitiva y con la que se ha trabajado previamente en el grado. Dispone a su vez de multitud de librerías y soporte, pero su nivel de complejidad es menor que el alcanzado con MicroPython o C++. (7)

- **ESP 32**



*Figura 14. Módulo ESP-32. Fuente: espressif.com*

Se trata de un microcontrolador basado en un chip de 32 bit. Dispone de conectividad inalámbrica por wifi y por bluetooth. Dispone de 30 pines, siendo 16 PWM y 25 entradas y salidas digitales. Su memoria interna es de 4MB. Es más potente que la Raspberry Pi Pico y que el Arduino Nano, pero a costa de esto

su consumo energético es mayor especialmente en picos puntuales al hacer uso del wifi integrado. Alcanza picos de hasta 400mA lo que supone 1,4 W de potencia consumida a máximo rendimiento. La alimentación de este módulo es de 3.3V y es más sensible a los cambios que en los anteriores dos controladores. Al contrario de la Raspberry Pi Pico este no dispone de pines I/O.

En la siguiente tabla se muestran en conjunto las características principales de estos microcontroladores:

*Tabla 2. Especificaciones de los microcontroladores.*

	<b>Raspberry PI Pico</b>	<b>Arduino Nano</b>	<b>ESP 32</b>
<b>CPU</b>	RP2040 dual-core Arm Cortex-M0+	ATmega328	Tensilica Xtensa 32bits Lx6
<b>Reloj CPU</b>	133 MHz	16 MHz	240 MHz
<b>RAM</b>	264 KB	2 KB	520 KB
<b>Alimentación</b>	1,8 – 5,5 V	7 – 12 V	3,3 - 3,6 V
<b>Voltaje de operación</b>	3,3 V	5 V	3,3 V
<b>GPIO</b>	26	22	3
<b>UART</b>	2	1	3
<b>I2C</b>	2	1	2
<b>Bluetooth</b>	No	No	Si
<b>Wifi</b>	No	No	Si
<b>Lenguaje programación</b>	MicroPython, C/C++	C/C++	MicroPython, C/C++, JavaScript
<b>USB</b>	Micro USB	Micro USB	Micro USB
<b>Precio</b>	4,75 €	18,36 €	9,95 €

Para la matriz de decisión de este dispositivo se va a dar la misma importancia a cada una de las siguientes características debido a que ninguna de ellas es más crítica para el correcto funcionamiento del dispositivo que las demás. Cualquiera de los anteriores podría desempeñar de peor o mejor forma el programa que se le solicitará.

*Tabla 3. Matriz de decisión de microcontroladores.*

	<b>Alimentación (1-5) 20%</b>	<b>Conectividad (1-5) 20%</b>	<b>Potencia (1-5) 20%</b>	<b>Coste (1-5) 20%</b>	<b>Software (1-5) 20%</b>	<b>TOTAL</b>
<b>Raspberry Pi Pico</b>	4	5	4	5	4	<b>22</b>
<b>Arduino Nano</b>	5	4	1	1	3	<b>14</b>
<b>ESP 32</b>	2	3	5	3	4	<b>17</b>

El controlador elegido será el Raspberry Pi Pico, destacando su conectividad y encontrando un buen equilibrio entre potencia y coste, ya que, cuesta la mitad que el ESP 32 y una cuarta parte que el Arduino Nano. Incluso implementando su versión con wifi, para posteriores revisiones del diseño, seguiría ocupando la primera posición por su bajo coste. En cuanto a la potencia y memoria, es capaz de competir con la del ESP 32 y deja muy por detrás al Arduino Nano por su chip de 10 bits.



## Módulo receptor GPS:

Los módulos GPS deben de ser compactos, suficientemente precisos y que tengan sean capaces de satisfacer la demanda de 2 Hz de tasa de refresco. Han de poder ser alimentados mediante la batería del vehículo y no requerir de una antena de grandes dimensiones. Se ha optado por módulos GPS que se comunican con el controlador mediante el protocolo UART (universal asynchronous receiver / transmitter, por sus siglas en inglés) es un protocolo serial que utiliza dos hilos para transmitir en ambas direcciones. Se adentrará más adelante en el funcionamiento de este protocolo, pero es comúnmente utilizado en este tipo de módulos (8). La precisión que alcanzan este tipo de módulos que no utilizan sistemas de doble frecuencia ni apoyo en estaciones base no baja de los 3 metros. Este valor es suficiente para el correcto funcionamiento del dispositivo debido a que no requiere únicamente de una sola medición sino de la distancia acumulada a lo largo de las mediciones. Los dos modelos comerciales que más se ajustan a estos requisitos son:

- **Adafruit Ultimate GPS Breakout V3**



Figura 15. Adafruit Ultimate GPS Breakout V3. Fuente: [adafruit.com](http://adafruit.com)

Se trata de un receptor GPS capaz de proporcionar 10 datos por segundo, recibiendo señal de 66 canales. Su alimentación es de 5V, compatible con la alimentación del controlador. El dispositivo contiene una antena cerámica de pequeño tamaño y dispone de una entrada para conectar una antena externa. Su precisión en cuanto a la posición es de hasta 3m. Su precio es superior a los 50 € lo que supondría gran parte del presupuesto total del dispositivo en adición del coste de la antena. Su comunicación con el microcontrolador es de tipo UART mediante dos cables de señal y dos de alimentación.

- **Ublox GPS: NEO-6M GPS Module**



Figura 16. Módulo GPS NEO-6M. Fuente: u-blox.com

Se trata de un receptor GPS capaz de proporcionar hasta 5 datos por segundo, recibiendo señal de 50 canales. Incorpora una antena pasiva cerámica externa que puede ser cambiada. Su alimentación es por medio de 3V, que pueden ser suministrados directamente por el Microcontrolador. Es capaz de comunicarse con el Microcontrolador mediante protocolos UART, SPI y USB. Su precisión es de hasta 3 metros como el anterior módulo. En cuanto al coste, este módulo es sustancialmente más económico que el anterior teniendo un precio de 17,75 €.

Tabla 4. Especificaciones de los módulos GPS

	<b>Ublox Neo-6M</b>	<b>Adafruit Ultimate GPS Breakout V3</b>
<b>Voltaje de operación</b>	3,3 – 5 V	3,3 – 5V
<b>Velocidad de puerto serial por defecto</b>	9600 baudios	9600 baudios
<b>Frecuencia de actualización</b>	Hasta 5 Hz	Hasta 10 Hz
<b>Tiempo de inicio en frío</b>	38 segundos	34 segundos
<b>Antena incluida</b>	Si	No
<b>Precisión</b>	Hasta 3 m	Hasta 3 m
<b>Comunicación</b>	UART	UART
<b>Canales</b>	50	60
<b>Satélites</b>	22	22
<b>Precio</b>	17,75 €	61,66 €

El requerimiento de tasa de refresco para este proyecto se marcó en 2 Hz. Ambos módulos satisfacen esa condición y sus características son similares. Debido a que la mayor velocidad de refresco de la información del módulo Adafruit no será aprovechada en este dispositivo, el módulo elegido será el Ublox Neo-6M debido a su menor coste.

## Pantalla:

Como se ha mencionado anteriormente, se requiere de dos pantallas de al menos 5 dígitos que muestren la distancia total y parcial recorrida y de una tercera pantalla que indique los valores de velocidad y orientación. Esta última pantalla bastaría con 4 dígitos disponibles. Además, se considerará un tamaño aceptable para una correcta visualización a partir de que cada dígito tenga un tamaño de 1,5 cm. Se han seleccionado las siguientes como posibles soluciones:

- **Módulo HT1621 LCD de 2,4 pulgadas, 6 dígitos y 7 segmentos.**

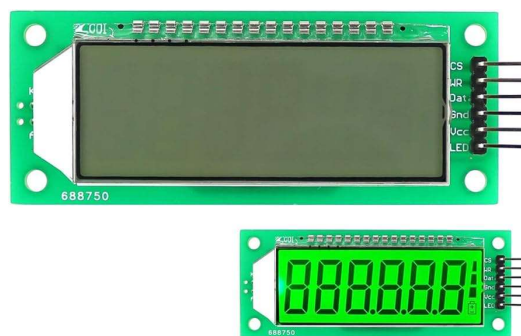


Figura 17. Módulo HT1621. Fuente: aliexpress.com

Se trata de una pantalla LCD retroiluminado de 6 caracteres. Es programable por protocolo SPI (Serial Peripheral Interface) mediante 3 cables. Se hablará más sobre este protocolo más adelante. La visibilidad de los caracteres es excelente y los dígitos son de gran tamaño. Este tipo de pantallas tienen una muy buena visibilidad al ser retroiluminadas, pero pecan de una menor resistencia en condiciones de humedad y la pantalla puede quebrarse fácilmente en caso de impactar con algo durante la prueba.

- **Módulo TM1637 LED digital de 0,36/0,56 pulgadas, 4 dígitos y 7 segmentos.**



Figura 18. Módulo TM1637 de 4 dígitos. Fuente: aliexpress.com

Estos módulos disponen de tecnología LED y no son retroiluminados. Su visibilidad en condiciones de luz lateral es menor que la de los anteriormente mencionados LCD, pero su brillo es suficiente para su correcto funcionamiento. En condiciones en las que la pantalla no recibe luz directa desde la espalda del usuario la visibilidad de los caracteres es igual de buena que en sus homónimos retroiluminados. Las

pantallas con dígitos de 0,36 pulgadas son económicas debido a que son comúnmente utilizados en proyectos de iniciación a la programación, pero su reducido tamaño dificulta la lectura de los números en un primer vistazo. Estos dígitos miden 0,9 cm y no satisfacen los requerimientos iniciales, pero existe esta misma pantalla en su versión de 0,56 pulgadas por dígito (1,5 cm) que es más adecuada para este proyecto. Estos dispositivos son comúnmente usados con la Raspberry Pi Pico y disponen de librerías que facilitan su programación. El solo disponer de 4 dígitos implicaría que a partir del kilómetro 99,99 se perdiese un valor decimal. Como ya se indicó anteriormente, una única de estas pantallas no sería suficiente para mostrar la distancia total recorrida y serían necesarias dos. Esta configuración no es la óptima ya que requiere de una utilización mayor de los puertos digitales del controlador, pero si es ideal para la pantalla auxiliar.

**Módulo TM1637 LED digital de 0.36/0,56 pulgadas, 6 dígitos y 7 segmentos.**



Figura 19. Módulo TM1637 de 6 dígitos. Fuente: aliexpress.com

Esta pantalla solventa los problemas de falta de capacidad de la pantalla de cuatro dígitos. A diferencia del anterior, esta no dispone de una librería específica, pero se trabajará sobre ella para lograr el funcionamiento del módulo. Al igual que la anterior, se encuentra disponible en tamaños de dígito de 0,36 y 0,56 pulgadas.

- **Módulo Tipo LCD1602, 5 pulgadas, dos filas de 16 dígitos**



Figura 20. Módulo LCD1602. Fuente: aliexpress.com

Se trata de una pantalla retroiluminada controlada por tecnología I2C compuesta por dos filas de 18 caracteres. La variedad de caracteres disponibles es mayor que en sus equivalentes de 7 segmentos, pero su lectura es menos clara en un primer vistazo debido al tamaño de los dígitos y a la cercanía entre líneas. Es la más cara de las propuestas y es especialmente sensible a las vibraciones y a la humedad.

En la siguiente tabla se muestran las características principales de las pantallas descritas:

*Tabla 5. Especificaciones de las pantallas.*

	<b>HT1621 LCD de 2,4''</b>	<b>TM1637 4 dígitos</b>	<b>TM1637 6 dígitos</b>	<b>LCD1602 16x2 dígitos</b>
<b>Conectividad</b>	SPI	Protocolo propio	Protocolo propio	I2C
<b>Tecnología</b>	LCD	LED	LED	LCD
<b>Número de caracteres</b>	6	4	6	32
<b>Librerías OpenSource</b>	C++	C++, MicroPython	C++, MicroPython	C++, MicroPython
<b>Retroiluminada</b>	Si	No	No	Si
<b>Precio</b>	7,23 €	2,13 €	3,28 €	8,74 €

Para elegir la pantalla idónea para el proyecto se ha dado una importancia distinta a cada una de las cualidades de estas. Los dos puntos más relevantes a los que se les ha dado más valor son el coste y la eficacia de estas. Primeramente, es importante considerar que la visibilidad de los dígitos sea buena y, por otro lado, al ser tres unidades las que se requieren, una diferencia en el coste de dos pantallas acaba suponiendo una gran diferencia en el coste final del producto.

Por otra parte, se ha dado la misma importancia a la resistencia y al software compatible. La no existencia de este software no supondría un problema técnico debido a que se podría desarrollar una propia, pero supondría un aumento de recursos dedicados a ello y su respectivo impacto en el coste final del dispositivo.

*Tabla 6. Matriz de decisión de pantallas.*

	<b>Coste (1-5) 30%</b>	<b>Eficacia (1-5) 30%</b>	<b>Resistencia (1-5) 20%</b>	<b>Software OpenSource disponible (1-5) 20%</b>	<b>TOTAL</b>
<b>HT1621 LCD</b>	3	5	3	2	<b>34</b>
<b>Tm 1637 4 Dígitos</b>	5	1	4	5	<b>36</b>
<b>Tm 1637 6 Dígitos</b>	4	4	4	4	<b>40</b>
<b>LCD 1602</b>	2	3	3	3	<b>27</b>

La opción elegida será el módulo Tm 1637 de 6 Dígitos. Su bajo precio, su mayor resistencia a los elementos y su facilidad de programación lo hacen el ideal para este proyecto. Se va a hacer uso de dos de estos módulos y de uno de 4 dígitos debido a que no se requiere de tantos caracteres para su utilización y su precio es menor.

### Fuente de alimentación:

La fuente de alimentación del dispositivo será la propia batería del vehículo o en su defecto una toma extraída de esta. La batería de la mayoría de los vehículos provee 12V DC nominales, sin embargo, los valores reales varían en función de la carga del alternador y del consumo del vehículo oscilando en orden de marcha entre los 14.2 V y los 12.0 V. Por debajo de estos valores se consideraría un fallo en el vehículo y algunos de los sistemas dejarían de funcionar correctamente. Para disminuir esta tensión hasta la tensión de alimentación del microcontrolador y los periféricos se hará uso de un módulo Step-Down. Gracias al rango de tensión de alimentación del microcontrolador no hará falta un módulo que mantenga 3.3V constantes. Bastará con que los valores se mantengan dentro de las tensiones permitidas.

Para el correcto dimensionamiento del convertor de tensión se debe de conocer cuál será el consumo de corriente.

Para el dimensionamiento se va a considerar la condición límite en cuanto a corriente evaluando el valor máximo de corriente de cada módulo con un coeficiente de simultaneidad igual a la unidad. Esta situación es poco probable que suceda, pero se ha elegido así para sobredimensionar el conjunto por seguridad.

Las corrientes máximas consumidas por los módulos son las siguientes:

*Tabla 7. Consumo de corriente aguas abajo del convertor de tensión.*

	<b>Consumos a plena carga</b>
<b>Raspberry Pi Pico</b>	0,94 A
<b>Módulo de pantalla LED TM1637, 0.56 pulgadas, 4 dígitos</b>	0,012 A
<b>Módulo de pantalla LED TM1637, 0.56 pulgadas, 6 dígitos</b>	0,026 A
<b>GPS Neo-6m</b>	0,076 A

La corriente en condiciones límite será del orden de 1 amperio. Para ello se va a hacer uso de un módulo comercial de Step Down. Denominado LM2596. Este módulo soporta hasta 3 amperios por lo que estaría en condiciones de operación incluso en casos límites de funcionamiento. Este módulo dispone de un potenciómetro con el que se regula el voltaje de salida deseado. EL microcontrolador Raspberry Pi Pico

permite una alimentación por USB de entre 1,8V y 5,5V. Las pantallas son alimentadas entre 3,3V y 5V debido a que ambos tienen un regulador de corriente en su interior. El módulo GPS que es más sensible a el cambio de tensión será alimentado por el microcontrolador que aporta una tensión más uniforme que el Step-Down debido a las subidas y bajadas de voltaje de la batería.

Debido a esto, se regulará la fuente para dar 4,5 V en salida mediante el giro del potenciómetro cuando la fuente proporcione 12V.



Figura 21. Módulo transformador LM2596. Fuente: aliexpress.com

## 5.6. Errores de precisión y simplificaciones

El uso de módulos más económicos en este dispositivo que los utilizados en odómetros profesionales tiene como consecuencia un aumento en la magnitud de los errores de precisión. En el cálculo de la distancia recorrida por el vehículo se van a considerar dos grandes grupos de errores. En primer lugar, están los errores derivados de la falta de precisión de los aparatos de medición GPS y en segundo lugar estarían los errores derivados de utilizar simplificaciones para adaptar la realidad a un modelo que el microcontrolador pueda procesar.

La elección de un microcontrolador en lugar de un pequeño ordenador lleva consigo una serie de dificultades en términos de realizar los cálculos necesarios para medir la distancia en el tiempo disponible. Se ha estudiado el impacto de factores como el cambio de altitud o la curvatura de la tierra para determinar si son relevantes en la medición de las distancias con este dispositivo. A modo de ejemplo, sería mucho más sencillo el cálculo de distancias entre dos puntos en un modelo plano en dos dimensiones que el cálculo de la distancia sobre la superficie de la Tierra teniendo en cuenta la topografía y los desniveles de cada ubicación.

### Consideraciones relativas a la curvatura de la tierra:

La distancia más corta entre dos puntos es una línea recta entre ellos. Sin embargo, cuando los puntos pertenecen a la superficie de una esfera no es de utilidad conocer la distancia de una línea recta subterránea. Para estos casos se hace uso de la ya mencionada Formula de Haversine (Ec.(2)).

Esta fórmula es muy potente para la medición de puntos que se encuentra separadas varias unidades de latitud y longitud. Sin embargo, no es idónea su utilización para el cálculo de este dispositivo. En una distancia de 10 metros sobre la superficie terrestre la variación del dato de latitud se produce tan solo en el quinto decimal. Esto supone un problema computacional para el microcontrolador que al no tener tanta capacidad computacional aproxima el seno de un número tan cercano a cero. La aplicación directa de esta fórmula requeriría de una mayor capacidad computacional que implicaría la elección de un ordenador superior.

Sin embargo, al hacer mediciones tan próximas, la sección de la superficie de la esfera se puede considerar un diferencial comparado con el perímetro de la tierra (40.000.000 m). Este es el mismo fenómeno por el que las personas no apreciamos la curvatura de la tierra. En pequeñas distancias la curvatura es despreciable y debido a que por la naturaleza del dispositivo las distancias se ajustarán periódicamente con las del roadbook, se va a desarrollar un método particular para la medición de las distancias sin utilizar la Formula de Haversine.

La aproximación se basa en el cálculo de la distancia entre la longitud y latitud de ambos puntos para, una vez calculada, realizar una aproximación aun modelos plano y aplicar la distancia más corta entre esos dos puntos. El procedimiento es el siguiente:

1. Se calcularán los desplazamientos en sentido Norte-Sur y Este-Oeste de forma independiente, y posteriormente se aproximará la distancia recorrida por el teorema de Pitágoras.
2. Para la variación de Latitud, se realizará asumiendo un modelo terrestre esférico de radio medio constante de 6378000 m. El perímetro polar asumido este radio será de

$$2 * \pi * R = 40074155.89 \text{ m} \quad (3)$$

Este perímetro se dividirá en fracciones de forma que la diferencia de latitud se multiplique por este valor tabulado evitando así los cálculos trigonométricos de valores cercanos a cero. Dividido el valor del perímetro en 360 secciones de modo que la distancia corresponda a la recorrida al variar un grado la Latitud. La cantidad de secciones es arbitraria y no supone un impacto en el error del sistema.

$$\Delta Lat_{1^\circ} \frac{\text{Perímetro}}{360} = 111319.44 \text{ m} \quad (4)$$



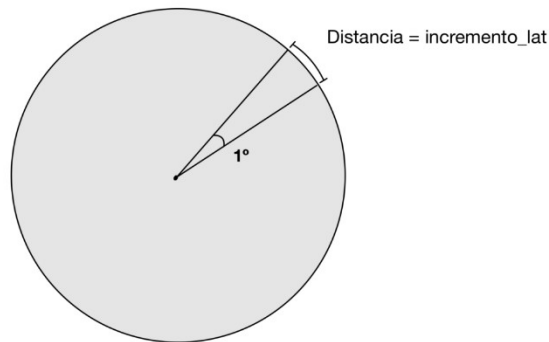


Figura 22. Distancia recorrida por unidad de latitud. Elaboración propia.

- Para la variación de Longitud se estudian dos modelos posibles. El primero de ellos se basa en el mismo concepto de radio terrestre constante, mientras que el segundo se basa en el estándar cartográfico World Geodetic System 84 (WGS 84) que propone un modelo más preciso sobre la geometría de la Tierra, asumiendo una forma elipsoide frente a la simplificación esférica. La implementación de ambos modelos busca el cálculo del perímetro de la sección de la Tierra que se encuentra a la latitud de la primera medición, asumiendo que la diferencia frente a la segunda latitud es insignificante. En el caso más sencillo, caso esférico, el cálculo se realiza de la siguiente forma:

$$r = R * \cos (\Phi 1) \quad (5)$$

Siendo  $r$  el radio del casquete esférico,  $\Phi 1$  la latitud de la primera medida y  $R$  el radio medio de la Tierra.

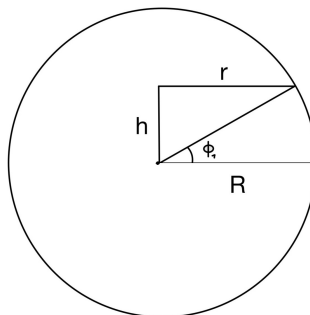


Figura 23. Cálculo trigonométrico del radio del casquete esférico. Elaboración propia.

Una vez obtenido el radio  $r$ , se calcula el perímetro de la base del casquete como:

$$\text{Perímetro del casquete esférico} = 2 * \pi * r \quad (6)$$

$$\Delta \text{Long}_{1^\circ} = \frac{\text{Perímetro de caquete esférico}}{360} \quad (7)$$

Este perímetro en 360 porciones de modo que la distancia corresponda a la recorrida al variar un grado la Longitud.

El modelo WGS 84 plantea una geometría elipsoidal de la Tierra achatada por los polos debido a la rotación de esta. Debido a esto, estos modelos proporcionan una aproximación más cercana a la realidad con modelos elipsoidales de parámetros conocidos. (9) La implementación de este modelo es sencilla, pero requiere algunas modificaciones.

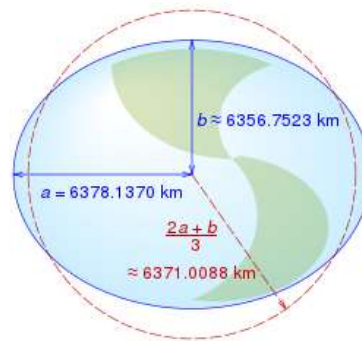


Figura 24. Radio medio según el modelo WGS 84. Fuente: Wikipedia.org

El radio de la tierra ha de ser calculado previamente para cada valor de Latitud. Una vez se obtiene este nuevo radio, se realizan los mismos cálculos que en el modelo esférico.

Para el cálculo del radio se utiliza un modelo elipsoidal con los parámetros .

$$a = 6378137 \text{ m}$$

$$b = 6356752.3142 \text{ m}$$

y siguiendo la ecuación:

$$R_{WGS 84} = \sqrt{\frac{(a^2 \cos(\Phi_1)^2 + b^2 \sin(\Phi_1)^2)}{(a \cos(\Phi_1)^2 + b \sin(\Phi_1)^2)}} \quad (8)$$

Siendo  $R_{WGS 84}$  el radio calculado y  $\Phi_1$  la latitud de la primera medida.

- Una vez obtenidos estos datos para cada latitud, se calcula la distancia recorrida mediante el teorema de Pitágoras. Para distancias del orden de 10 metros, la curvatura de la tierra se considera despreciable y se puede aplicar criterios de trigonometría en dos dimensiones.

$$\text{Ángulo vertical} = |\text{Latitud } 2^{\text{a}} \text{ medición} - \text{Latitud } 1^{\text{a}} \text{ medición}| \text{ (grados)} \quad (9)$$

$$\text{Ángulo horizontal} = |\text{Longitud } 2^{\text{a}} \text{ medición} - \text{Longitud } 1^{\text{a}} \text{ medición}| \text{ (grados)} \quad (10)$$

$$\text{Distancia recorrida} = \sqrt{(\text{Ángulo vertical} * \Delta \text{Lat}_{1^{\circ}})^2 + (\text{Ángulo horizontal} * \Delta \text{Long}_{1^{\circ}})^2} \quad (11)$$

El microcontrolador encuentra problemas en el cálculo de senos cercanos a cero, pero no encuentra problemas con el producto de números con gran cantidad de decimales.

En estos cálculos se han considerado una decena de decimales, que son los suministrados por el módulo GPS para no inducir a error por la omisión de decimales.

Sin embargo, hay otros errores derivados de esta simplificación.

#### **Consideración del cambio de altitud:**

Un cambio de altitud afectaría a la cantidad de metros recorrida por el vehículo respecto del mismo recorrido sobre una carretera plana. El módulo GPS proporciona datos acerca esta variación de altitud, pero estos datos son muy imprecisos. Su valor es cambiante y obtiene una precisión depende en gran medida de la posición de los satélites. Antes de buscar formas de reducir este error se va a evaluar si considerar la altitud está justificado.

El efecto de la inclinación de una rampa se mide en porcentaje. Este porcentaje expresa la distancia recorrida en vertical frente a la recorrida horizontalmente. Según la Norma 3.1 IC de la Instrucción de carreteras (10), la inclinación excepcional máxima para una carretera convencional a una velocidad de 40km/h es del 10%. En momentos puntuales un vehículo puede superar un mayor desnivel haciendo uso de la inercia en cortos periodos de tiempo antes de perder velocidad y pararse. Se van a evaluar ambos escenarios para determinar si es importante tenerlos en cuenta:

- Distancias de más del 10% en cortos periodos de tiempo.

En este grupo se encuentran pendientes muy pronunciadas de corta duración que requieren de la inercia del vehículo para ser superadas. Son poco habituales y su influencia en la distancia recorrida total es despreciable.

## 2. Desniveles de hasta el 10% en largas distancias.

En este grupo se encuentran los desniveles propios de los caminos y carreteras que se extienden a lo largo del recorrido. Un desnivel del 10% supone un ángulo de 5 grados frente a la horizontal. La variación de distancia recorrida frente al plano se calcula como

$$Distancia\ real = Distancia\ horizontal * \cos\left(\frac{inclinación}{2}\right) \quad (12)$$

A modo de ejemplo, para 1000 metros y un desnivel del 10% la diferencia entre la medición real y la simplificación es del 0.4%. Incluso en un caso límite de una bajada con un 40% de inclinación, estos errores de precisión son contenidos alrededor de un 5% en un caso de puro desnivel. Debido a esto la variación de altura se va a considerar despreciable.

Salvo en casos extremos de cambios de altitud, el error de precisión se encontrará tras el tercer decimal y se considerará despreciable frente a otros errores del dispositivo.

### **Precisión propia del dispositivo GPS:**

El error causado en la determinación de las coordenadas geográficas por el módulo GPS Neo 6-m es de 2.5 metros. La posición del punto de medida está en a un máximo de 2.5 metros de distancia del valor calculado.

### **Error absoluto del dispositivo**

El cálculo del error total del dispositivo es complejo debido a la naturaleza de este. El error de precisión de las coordenadas GPS no implica que el error de cada medición de distancia se incremente esa cantidad. Si se realizase una única medición, el valor del error se expresaría como:

$$Distancia\ recorrida = Ubicación\ 2 - Ubicación\ 1 \quad (13)$$

Siendo el error de cada medida de ubicación 2,5 metros, el error en una única medición sería la suma del error de ambas mediciones.

$$Error\ total = Error\ medición\ 1 + Error\ medición\ 2 = 2,5 + 2,5 = 5\ m \quad (14)$$

El funcionamiento interno del dispositivo tiene en cuenta la posición de la anterior posición registrada para el cálculo de la distancia recorrida. Esto hace que el error de medición no sea acumulativo en todas las mediciones, compensándose unas con otras. Para su mejor entendimiento se van a exponer el caso de la distancia recorrida con 3 ubicaciones registradas en línea recta y otro con un número indeterminado de ellas.

Para el caso de la distancia recorrida en tres mediciones el cálculo se realizaría como:

$$Distancia\ recorrida = (Ubicación\ 2 - Ubicación\ 1) + (Ubicación\ 3 - Ubicación\ 2) \quad (15)$$

Siendo que la Ubicación 2 se calcula una única vez, el error de medición no se comete dos veces y se puede simplificar la medición a:

$$\text{Distancia recorrida} = \text{Ubicación 3} - \text{Ubicación 1} \quad (16)$$

Siendo el error de cada medida de ubicación 2,5 metros, el error acumulado en la medición de la distancia con tres mediciones seguirá siendo de 5 metros.

$$\text{Error total} = \text{Error medición 1} + \text{Error medición 3} = 2,5 + 2,5 = 5 \text{ m} \quad (17)$$

Esto se extrapola a una distancia medida con N mediciones intermedias. Esto añadido a el error producido por el desnivel y el hecho de que las mediciones se realizan mediante líneas rectas entre puntos, establece el error del dispositivo en 10 metros para mediciones de distancias entre viñetas de un roadbook típico de un rally raid de regularidad de clásicos. Estas mediciones acostumbran a ser de alrededor de 500 metros, lo que supone típicamente un error del 2 % en la medición de distancias cortas.

En casos extremos de extrema pendiente, una trazada sinuosa y mala conexión con los satélites, se podría producir un mayor error que debe corregirse manualmente con los botones destinados a ello al llegar a la próxima viñeta. Las correcciones de este tipo son la base de la navegación en estas competiciones y permite el perfecto desempeño del dispositivo para la realización de las pruebas con estos valores de precisión.

## 5.7. Desarrollo software

Se ha realizado la programación sobre una Raspberry Pi Pico a la cual se le ha implementado MicroPython. Esta es una implementación del lenguaje de programación Python 3 optimizado para ejecutarse en microcontroladores. Se ha hecho uso del entorno de desarrollo Thonny para la programación ya que está optimizado para una conexión rápida con la Raspberry Pi Pico.

Se ha de desarrollar el software necesario para la comunicación entre dispositivos y el cálculo de la distancia recorrida siguiendo el modelo explicado en el apartado de simplificaciones. En siguiente diagrama de bloques se indican las necesidades de comunicación entre los elementos hardware. No todos se comunican de la misma manera y ha de entenderse cuales son los protocolos que utiliza cada uno.

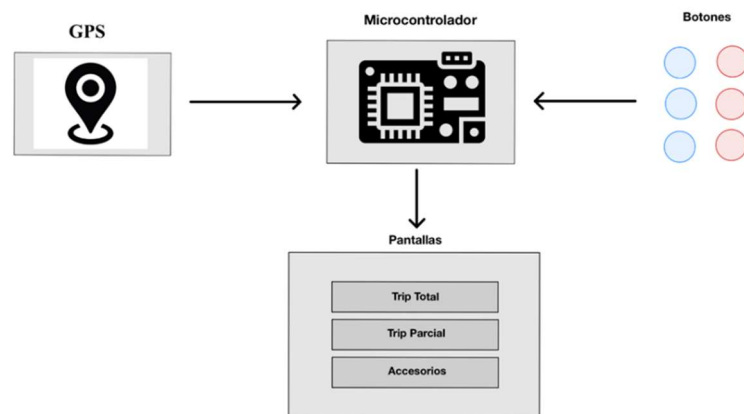


Figura 25. Diagrama en bloques del dispositivo a diseñar. Elaboración propia.

La pieza central del dispositivo es el microcontrolador. Este debe de poder leer los datos de cada módulo en su correspondiente protocolo y mandar al siguiente con el protocolo correspondiente. En este dispositivo se van a utilizar tres grupos protocolos de comunicación (11):

- Protocolo UART.

UART (universal asynchronous receiver / transmitter, por sus siglas en inglés) es un protocolo serial que utiliza dos hilos para transmitir en ambas direcciones. Es un protocolo simple y es el utilizado para la conexión del microcontrolador y el GPS. Este es el protocolo que va a utilizar el módulo GPS para enviar los mensajes al microcontrolador. La conexión entre estos se realiza mediante dos cables denominados Tx y Rx para enviar y recibir mensajes respectivamente.

Este protocolo puede enviar mensajes de entre 5 y 9 bits de carga útil pero lo más habitual es enviar 7 u 8. Esto permite el envío de caracteres ASCII (*American Standard Code for Information Interchange*) para el transporte de los mensajes GPS.

Tiene la característica de que no tiene señal de reloj, por tanto, ambos dispositivos han de transmitir datos a una velocidad preacordada. En el caso de este módulo se han configurado a 9600 baudios debido a que es la configuración por defecto y recomendada en la hoja de características del fabricante.

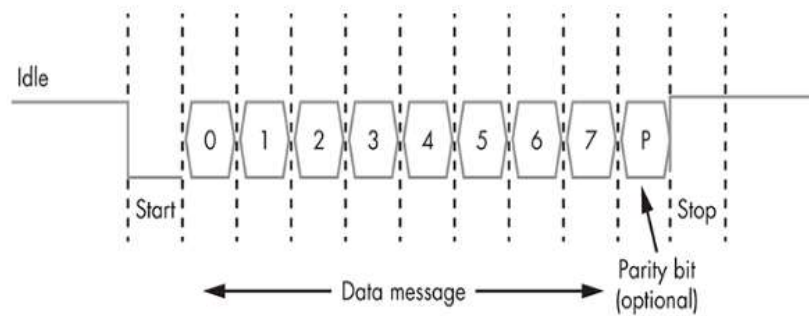


Figura 26. Estructura de bits del protocolo UART. Fuente: rohde-schwarz.com

Los mensajes enviados por el módulo GPS son interpretados como cadenas ASCII y forman mensajes en formato NMEA (National Marine Electronics Association) como los de la siguiente figura. Este es un protocolo común para gran parte de los dispositivos GPS.

```

$GPGLL, 2232.73995, N, 11404.60273, E, 030427.00, A, A*6B
$GPRMC, 030428.00, A, 2232.73995, N, 11404.60275, E, 0.037, , 070314, , , A*7E
$GPVTG, , T, , M, 0.037, N, 0.069, K, A*28
$GPGGA, 030428.00, 2232.73995, N, 11404.60275, E, 1.07, 1.17, 122.5, M, -2.3, M, , *4F
$GPGSA, A, 3, 29, 21, 18, 05, 14, 22, 26, , , , , 2.12, 1.17, 1.77*00
$GPGSV, 3, 1, 10, 05, 18, 096, 31, 12, 07, 154, 15, 14, 12, 248, 29, 15, 44, 025, *7B
$GPGSV, 3, 2, 10, 18, 38, 325, 43, 21, 61, 296, 41, 22, 09, 304, 31, 24, 70, 114, *7D
$GPGSV, 3, 3, 10, 26, 10, 045, 16, 29, 16, 208, 35*7B
$GPGLL, 2232.73995, N, 11404.60275, E, 030428.00, A, A*62
$GPRMC, 030429.00, A, 2232.73994, N, 11404.60277, E, 0.017, , 070314, , , A*7E
$GPVTG, , T, , M, 0.017, N, 0.031, K, A*27
$GPGGA, 030429.00, 2232.73994, N, 11404.60277, E, 1.07, 1.17, 122.7, M, -2.3, M, , *4F
$GPGSA, A, 3, 29, 21, 18, 05, 14, 22, 26, , , , , 2.12, 1.17, 1.77*00
$GPGSV, 3, 1, 10, 05, 18, 096, 31, 12, 07, 154, 14, 14, 12, 248, 29, 15, 44, 025, *7A
$GPGSV, 3, 2, 10, 18, 38, 325, 43, 21, 61, 296, 41, 22, 09, 304, 31, 24, 70, 114, 21*7E
$GPGSV, 3, 3, 10, 26, 10, 045, 14, 29, 16, 208, 35*79
$GPGLL, 2232.73994, N, 11404.60277, E, 030429.00, A, A*60
  
```

Figura 27. Mensajes suministrados por el GPS. Fuente: wavesshare.com

Se reciben multitud de mensajes con distinta información y no todos son relevantes para todas las aplicaciones. En este dispositivo se va a hacer uso de los mensajes \$GPRMC, \$GPVTG y \$GPGSV. Estos mensajes tienen la siguiente estructura:

StartChar	SentenceName	Separator	Data <sub>1</sub>	Separator	Data <sub>2</sub>	Separator	...	Data <sub>N</sub>	CheckChar	Checksum	EndChar
-----------	--------------	-----------	-------------------	-----------	-------------------	-----------	-----	-------------------	-----------	----------	---------

Figura 28. Estructura de un mensaje NMEA. Fuente: MathWorks.com

Cada mensaje va precedido de cinco letras que lo identifican y de diferente información. El mensaje GPRMC sigue la estructura descrita y contiene datos sobre la hora, el día, la velocidad y las coordenadas geográficas:

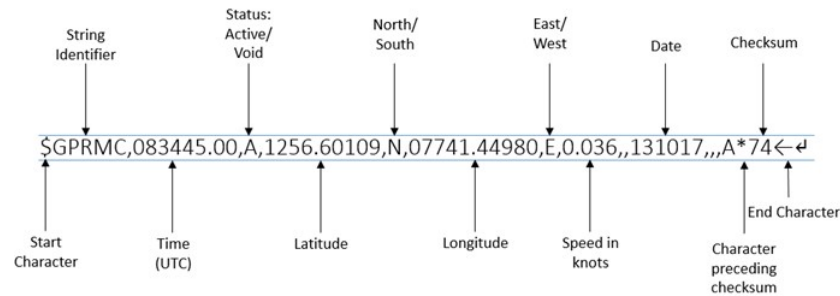


Figura 29. Mensaje tipo GPRMC. Fuente: MathWorks.com

Este es el mensaje de mayor importancia para el correcto funcionamiento del dispositivo. Adicionalmente se consultan los valores de GPVTG que da información adicional sobre la velocidad del dispositivo y la señal GPGSV que proporciona información sobre los satélites a la vista. Esta última información es muy valiosa ya que permite identificar problemas de cobertura. Durante el arranque en frío del dispositivo, el módulo GPS se conecta progresivamente a los satélites a la vista. La información suministrada por el GPS depende de la cantidad de satélites disponible.

Con un solo satélite el módulo es capaz de dar la información de la hora y el día, pero no dispone de los valores de latitud y longitud. Este mensaje permite darle al usuario una idea del progreso de la búsqueda de satélites a medida que el dispositivo enciende tras un periodo de inactividad.

#### - Protocolo Serial.

Los módulos tm\_1637 utilizan dos pines para transmitir los datos de forma serial. Uno es el pin CLK que produce una señal de reloj y otro denominado DIO por el que se transmite una señal digital. Es un protocolo similar al I2C, pero no es el mismo. La señal de reloj es común para los tres displays del dispositivo mientras que se disponen tres pines digitales para poder controlarlos individualmente. La información se transmite en dos paquetes de ocho bit con un bit para empezar el mensaje y otro para finalizarlo.



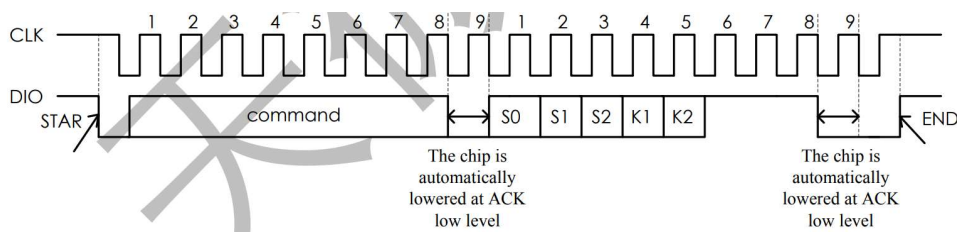


Figura 30. Estructura del bus utilizado por los módulos *tm\_1637*. Fuente: Datasheet del módulo *tm\_1637*.

El primer paquete indica el modo del display y el estatus de los leds. Entre estos comandos se encuentra la posición del dígito de 7 segmentos al que corresponde la información del segundo paquete. En la siguiente imagen se muestra el formato de estos comandos para indicar el dígito elegido.

MSB				LSB				Display address
B7	B6	B5	B4	B3	B2	B1	B0	
1	1	Zero should be inserted for irrelevant items.		0	0	0	0	C0H
1	1		0	0	0	1	C1H	
1	1		0	0	1	0	C2H	
1	1		0	0	1	1	C3H	
1	1		0	1	0	0	C4H	
1	1		0	1	0	1	C5H	

Figura 31. Comandos del bus de los módulos *tm\_1637*. Fuente: Datasheet del módulo *tm\_1637*.

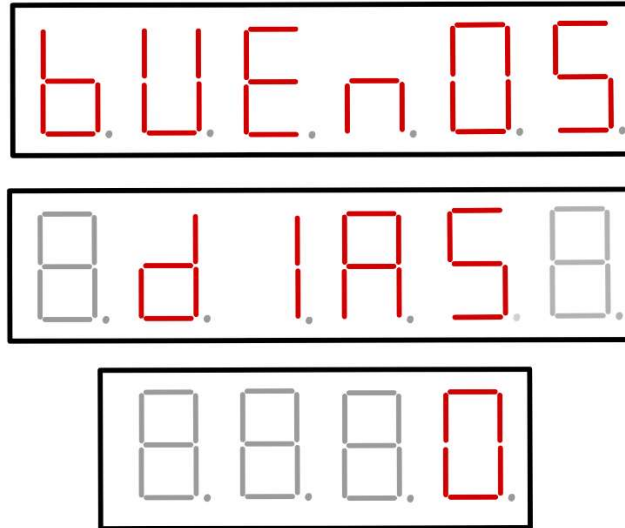
- Protocolo I/O.

Los botones funcionan como entradas digitales que se han programado internamente en el microcontrolador como Pull-Down. Eso evita la necesidad de instalar una resistencia externa. La existencia de este tipo de pines I/O es una de las ventajas que proporciona la elección de la Raspberry Pi Pico frente a la ESP-32. Se ha asignado una entrada digital para cada uno de los seis botones que dispone el dispositivo, por lo que habrá seis señales en total.

En la memoria de la Raspberry Pi Pico se han almacenado todos los archivos necesarios para el funcionamiento del dispositivo para realizar la comunicación entre los módulos y ejecutar las funcionalidades del dispositivo. El programa consta de un script principal denominado main que se ejecuta de forma cíclica y continuada.

Las funciones de este script son las siguientes:

Al inicio del dispositivo muestra un mensaje de bienvenida en el que escribe mediante las pantallas el mensaje “Buenos días” junto con una pequeña animación. Esto le indica al usuario que el dispositivo está encendido y funciona correctamente. Durante este periodo el módulo GPS ya ha sido encendido y ha empezado el tiempo de arranque en frío en segundo plano.



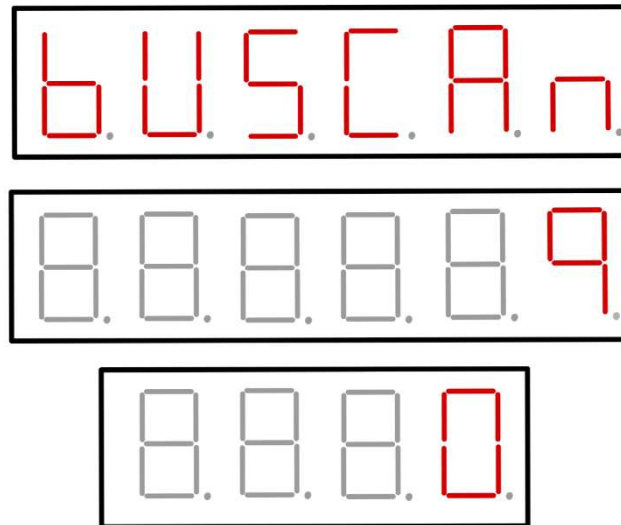
*Figura 32. Animación de encendido. Elaboración propia.*

Esta animación trata de dar la impresión al usuario de que el tiempo de espera hasta el contacto con los satélites es menor. Al finalizar este mensaje, si se conecta con el módulo GPS, por pantalla se le indica al usuario que el dispositivo está conectando con los satélites mediante el deslizamiento por la pantalla superior de la frase “Buscando satélites”.



*Figura 33. Mensaje de búsqueda de satélites por pantalla. Elaboración propia.*

En la pantalla intermedia se muestra un número que indica el número de satélites a la vista para dar una idea al usuario de la cobertura existente o del progreso de la búsqueda. Este dato lo saca de los ya mencionados mensajes GPGSV que contienen información sobre el estado de los satélites. En la siguiente imagen se muestra cómo sería la disposición de las pantallas en el supuesto de que hubiese conectado con nueve satélites.



*Figura 34. Dispositivo en proceso de conexión con los satélites. Elaboración propia.*

Una vez obtiene datos sobre latitud y longitud, el script procede a segmentar los mensajes NMEA en porciones útiles descartando datos que han perdido paquetes en el proceso o se encuentran incompletos.

Adquiridos dos valores válidos, se llama a la función `distance_GPS` para efectuar el cálculo de la distancia recorrida. La función `main` almacena una serie de variables entre las que se encuentran el valor de la longitud y latitud de las dos últimas posiciones y la distancia total recorrida hasta el momento.

Se le introducen los valores de posición de la ubicación actual y de la ubicación previa y la función `distance_GPS` realiza los cálculos explicados anteriormente para solventar la no utilización de la Fórmula de Haversine. Una vez calculado la función `distance_GPS` devuelve un valor de distancia parcial recorrida para que la función `main` pueda actualizar el valor de los trips parcial y total.

Las mediciones de distancia inferiores a 2,5 metros no son añadidas a la distancia recorrida. Esto se establece como filtro para que la variación de posición debido a el error GPS detecte movimiento aun estando estacionado. Sumar metros a la variable por este motivo generaría un error considerable en la medición si el tiempo de parada es largo y daría una mala sensación al usuario, pudiendo pensar que el dispositivo no funciona correctamente.

La medición de los metros puede interrumpirse de cinco formas distintas actualmente mediante cinco pulsadores. Las funciones de estos son:



Figura 35. Funcionalidades de los pulsadores. Elaboración propia.

- Botón 1: Incremento de metros del odómetro.

Este botón incrementa el valor del odómetro parcial y total en diez metros cada vez que se presiona el botón. El botón detecta una pulsación a partir de la primera pulsación cada 0,25 segundos para eliminar el efecto botón del pulsador o falsos contactos. Está programado de forma que si se mantiene presionado incrementará de forma continuada diez metros cada 0,25 segundos, por lo que no es necesario pulsar repetidas veces para hacer un ajuste significativo.

- Botón 2: Reducción de metros del odómetro.

Este botón funciona de forma similar al botón anterior pero un poco más complejo. Su función es reducir los metros de los odómetros parcial y final. Debido a que no es correcta la medición de distancias negativas, el programa tiene en cuenta el valor de los odómetros antes de restarles 10 metros. A modo de recordatorio, a partir de la primera viñeta el valor del odómetro parcial es menor que el del odómetro total. El programa bajará a la par ambos valores hasta que uno de ellos obtenga un valor menor a diez. En ese momento el valor menor pasará a ser cero y no seguirá descendiendo. En el caso que ambos valores sean menores que diez, se mantendrán ambos en cero aún si este botón está presionado.

- Botón 3: Reinicio de metros del odómetro.

Este botón permite reiniciar el odómetro parcial. Debido a crítico que sería borrar la distancia parcial por error al rozar el botón, el dispositivo muestra al pulsar el botón un mensaje en el que solicita “aceptar” el reinicio del parcial. Si el botón continúa presionado un segundo después, la distancia se pone a cero y continúa sumando a partir de ahí. En caso de ser tocado por error, un segundo después de ser pulsado el parcial continuará desde el valor en el que se encontraba antes de presionar el botón.

- Botón 4: Cambio de modo de la pantalla auxiliar.

Una vez pulsado se llama a la función Biestable y aparece en pantalla el modo seleccionado. Los modos precargados son velocidad, en el que marca la velocidad del vehículo, y rumbo, donde marca la dirección CAP en grados en la que se dirige el vehículo.

- Botón 5: Botón disponible para funcionalidades futuras.

Finalmente, el botón queda libre a disposición de nuevas funcionalidades programadas en actualizaciones o para funciones creadas por el usuario.

El programa corre la función principal del programa en el microcontrolador de forma cíclica, accediendo a otros scripts según lo que se le comunique a través de las entradas (12). La siguiente figura muestra un diagrama de flujo simplificado del funcionamiento del programa principal main.

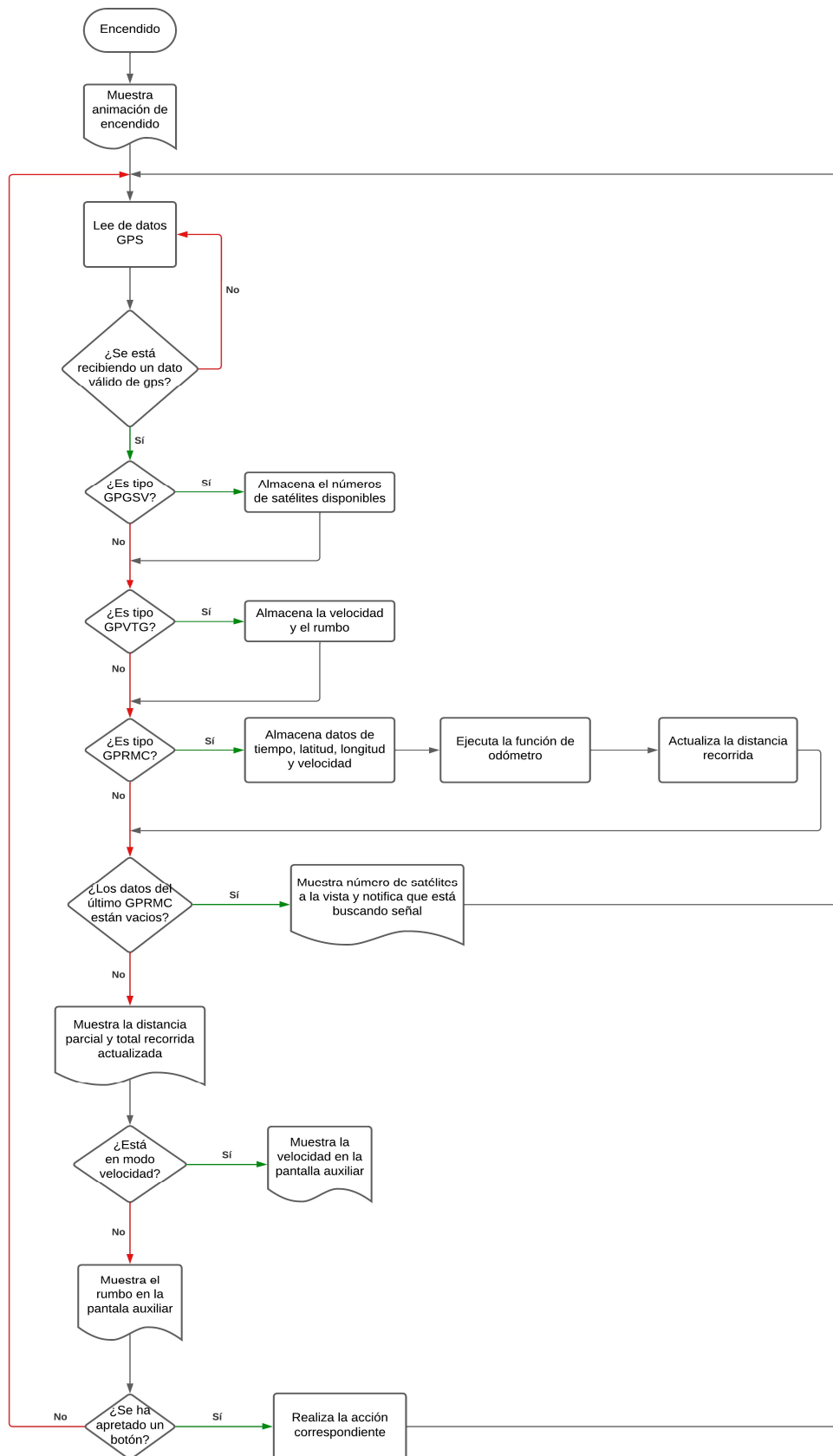


Figura 36. Diagrama de flujo del programa principal main. Elaboración propia.

En el desarrollo de los scripts se han utilizado librerías de código abierto como apoyo para otorgar funcionalidades. Primeramente, se ha hecho uso de la librería `mpy_decimal` para optimizar el trabajo con decimales en el cálculo de la distancia recorrida por el script `distance_GPS`. MicroPython por defecto realiza redondeos no deseados al trabajar con muchos dígitos decimales (12). Debido a la cercanía de los puntos medidos y a la necesidad de incorporar números grandes como el radio de la Tierra en las ecuaciones, es importante considerar gran número de cifras significativas para obtener el resultado buscado.

Se ha partido de una librería existente de código abierto para aportar funcionalidades a la muestra de información por pantalla. Esta ha sido modificada para adaptar algunas de sus funcionalidades al proyecto. Primeramente, esta es una librería enfocada a su uso en pantallas de 4 dígitos. El cambio de cuatro a seis dígitos ha supuesto una completa reestructuración de la librería y sus funciones internas. La principal ventaja de la adaptación de una librería existente frente a la creación de una librería propia es la no necesidad de escribir para todos los caracteres ASCII su respectivo array binario para el envío por comunicación serial, así como otras partes del código monótonas que requieren gran cantidad de horas de transcribir datos de bajo interés técnico.

Algunos factores como que los dígitos no se encuentren ordenados de derecha a izquierda, sino de la forma que se indica en la siguiente figura exigen una completa reestructuración significativa del código.

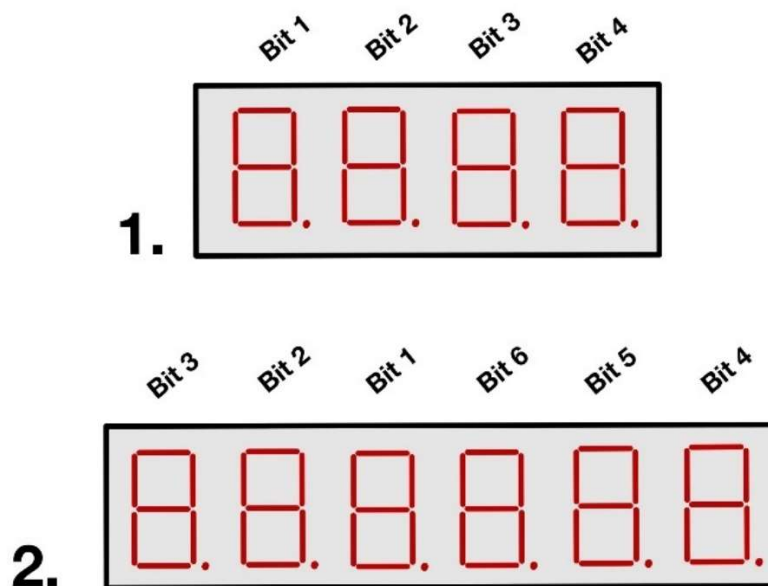


Figura 37. Configuración de los bits de siete segmentos. Elaboración propia.

El script `distance_GPS` es el encargado de realizar los cálculos matemáticos correspondientes a la medida de distancias. La función `main` ingresa los valores de longitud y latitud del momento actual y los del momento anterior, a los que llamaremos a partir de ahora `longitud_1`, `latitud_1` y `longitud_0`,

latitud\_0 respectivamente. Debido a la limitación de cálculo del microcontrolador no se puede hacer uso de la Formula de Haversine (2).

El script realiza los cálculos explicados en el apartado de simplificaciones a partir de los datos suministrados por el main como expresa el diagrama de flujo a continuación.

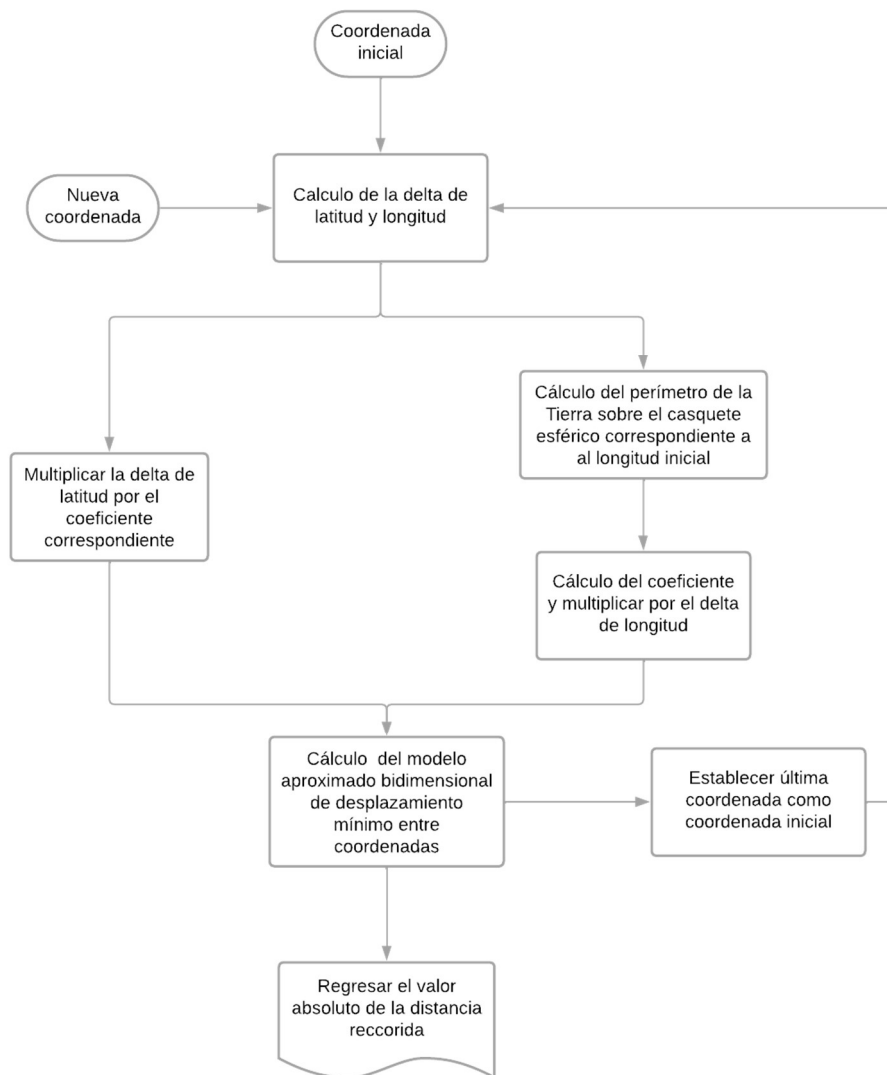
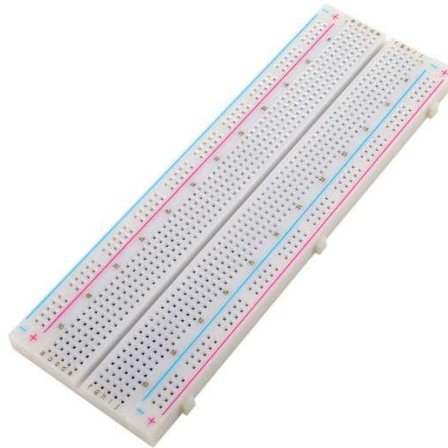


Figura 38. Diagrama de flujo del script distance\_GPS. Elaboración propia.



## 5.8. Proceso de fabricación

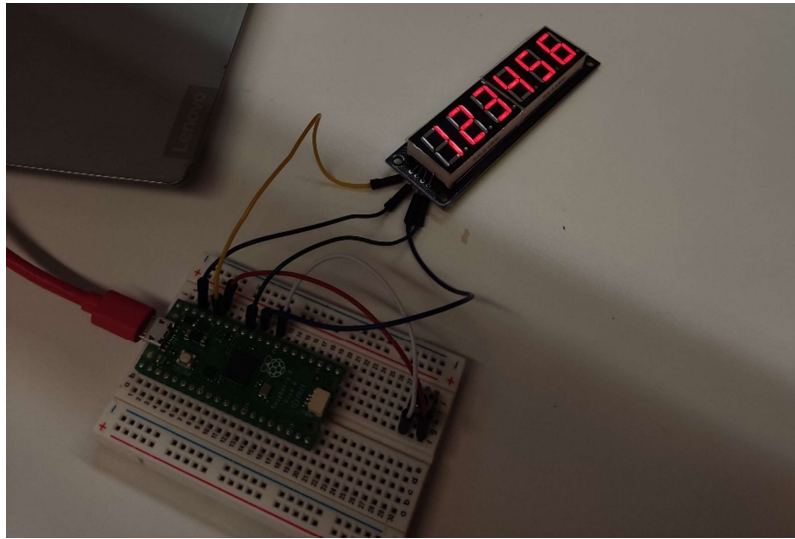
Una vez finalizada la etapa de diseño y como se indica en el diagrama de Gantt de este proyecto se comienza con el desarrollo del hardware con las pruebas unitarios de los dispositivos. Estas pruebas se realizan en una paca de prototipos tipo protoboard. Estas placas permiten realizar conexiones entre módulos y crear circuitos de prototipos sin necesidad de soldadura.



*Figura 39. Placa de prototipos tipo protoboard. Fuente: Mexbit.com.*

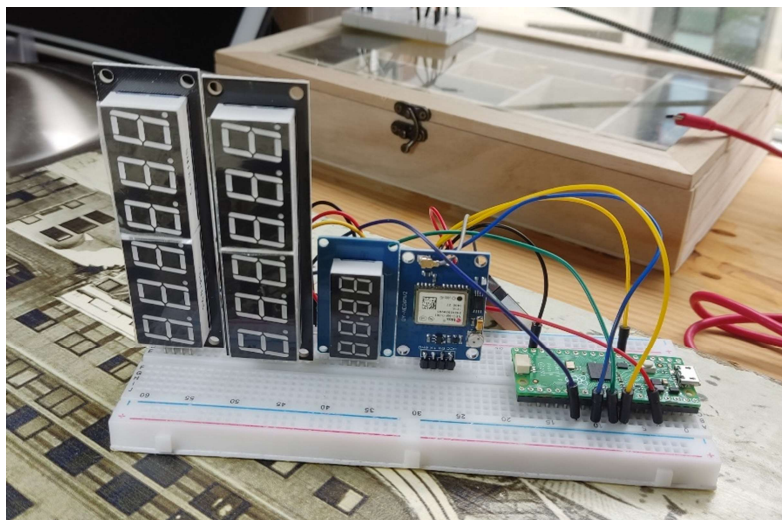
La finalidad de estas pruebas es verificar el correcto funcionamiento de los módulos seleccionado y facilitar la detección de fallos en la programación. El tiempo dedicado a estas pruebas es bajo y un posible fallo es mucho más sencillo y rápido de solucionar que durante el montaje final.

Primeramente, se comienza con la prueba de una única pantalla de siete segmentos conectada al microcontrolador. Ha de realizarse un pequeño script que permita realizar todas las acciones que se requerirá posteriormente en el programa final. Estas son mostrar conjuntos numéricos, muestra de texto y deslizar mensajes. En la siguiente figura se muestra una posible configuración para este tipo de pruebas. De la misma forma que se realiza la prueba con la pantalla se prueba la comunicación del microcontrolador con el módulo GPS.



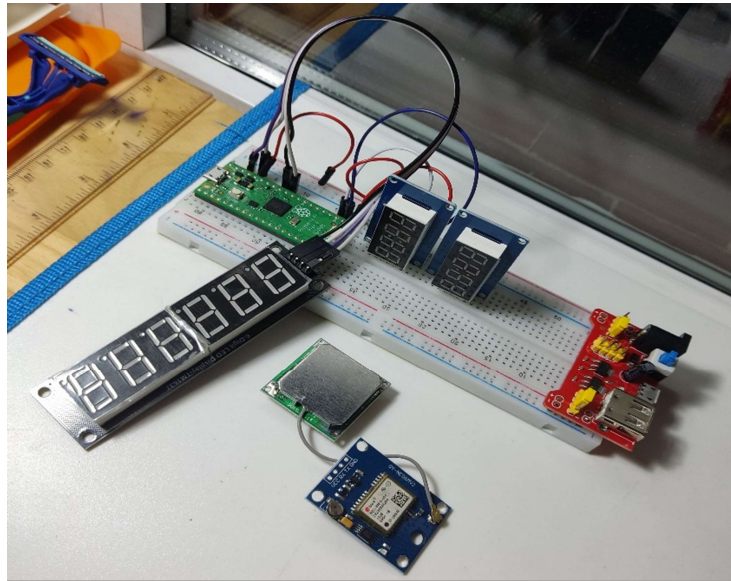
*Figura 40. Prueba unitaria de una pantalla de 7 segmentos. Elaboración propia.*

Una vez realizadas las pruebas individualmente se va a realizar un montaje en la placa de prototipos que tenga una configuración similar a la disposición final. Se ha optado por hacer un conexionado entre las pantallas, el microcontrolador y el módulo GPS exclusivamente, no incluyendo los pulsadores. En esta etapa del prototipo se van a simular las señales de los pulsadores puenteando tensión a las entradas digitales del microcontrolador en caso necesario. Esto busca una simplificación del circuito.



*Figura 41. Montaje alimentado por el microcontrolador. Elaboración propia.*

La alimentación para las pruebas de una única pantalla o un único módulo GPS es proporcionada por el pin de alimentación del propio microcontrolador que a su vez es alimentado por el cable micro USB que lo conecta al ordenador. En caso de que el voltaje de alguno de los componentes no pueda ser proporcionado por la fuente o el consumo de corriente sea mayor del suministrado por el microcontrolador se puede hacer uso de una fuente de tensión para placas de prototipo como la de la siguiente imagen. Esto permite también alimentar al microcontrolador sin la proximidad del ordenador.



*Figura 42. Montaje alimentado por una fuente externa, Elaboración propia.*

Estos pasos de pruebas pueden ser omitidos por una persona que busque replicar el dispositivo debido a que el código ya es funcional para los módulos descritos. En caso de no disponer de los mismos elementos o buscar algún tipo de cambio es altamente recomendable su realización. Es importante plantear un plano de las conexiones utilizadas para una vez funcione el dispositivo en la placa de prototipos se pueda replicar en un montaje final. Este plano incluye todos los módulos es el siguiente:

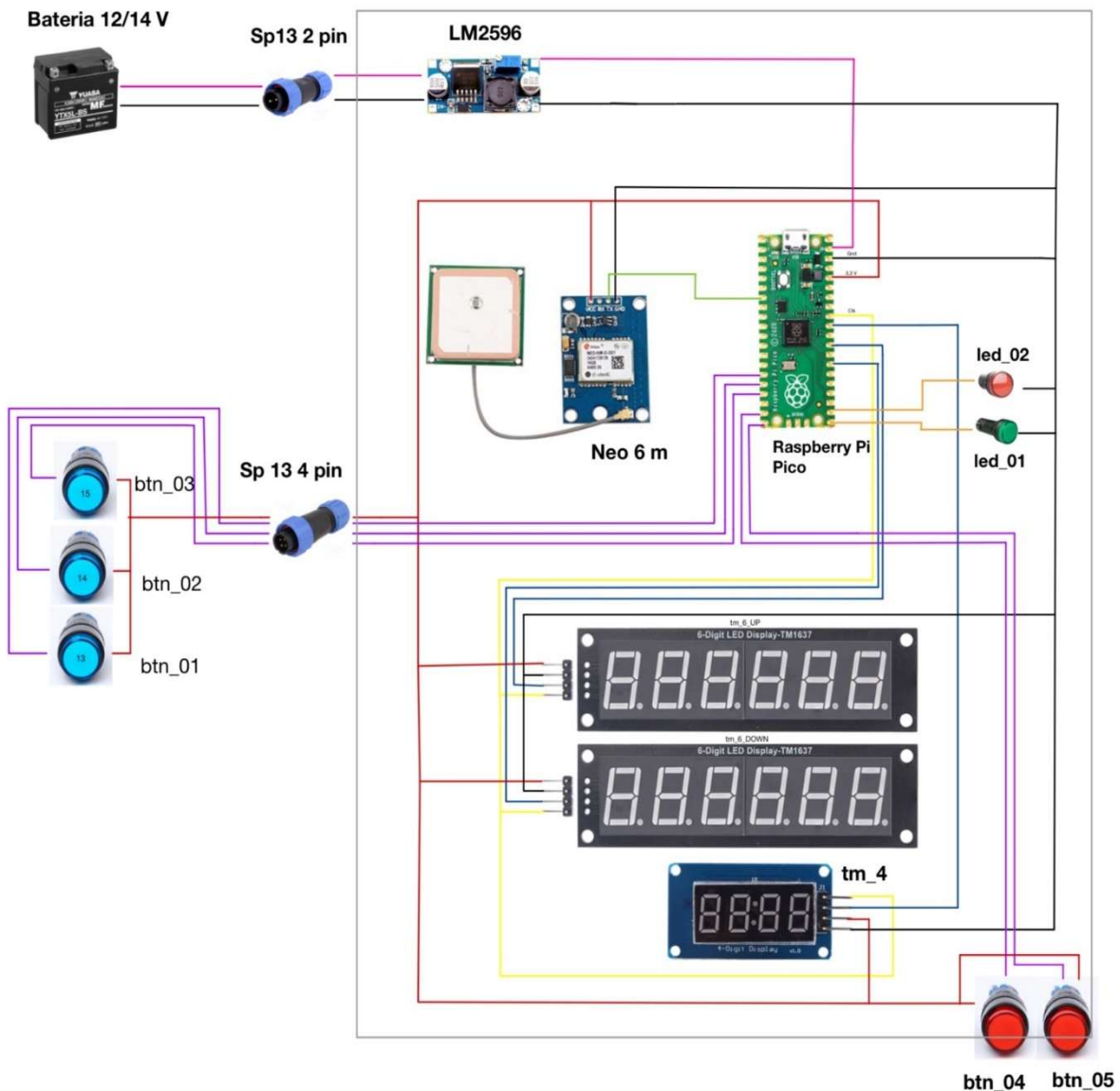


Figura 43. Plano de conexiones eléctricas. Elaboración propia.

Es posible reducir el número de cables del circuito anterior. Se podrían conectar todas las tierras a un mismo conector que terminase en el microcontrolador. Esta disposición se ha hecho pensando en realizar un diseño intuitivo y fácil de replicar y cablear por personas poco familiarizadas con los montajes eléctricos buscando el objetivo de hacer el dispositivo accesible.

Una vez se encuentren todos los módulos conectados correctamente sobre la protoboard, se procede a la carga del programa y a realizar pruebas del funcionamiento completo del circuito. Es importante situarse en una ubicación a cielo descubierto o cerca de una ventana como la que aparece en la siguiente figura para poder captar señal con el GPS. En ubicaciones interiores o subterráneas no es capaz de captar señal y así lo indicará por pantalla. En el caso de no recibir el mensaje de “Buenos días” al encender el dispositivo, no será un problema de cobertura satelital sino de comunicación con el módulo GPS.

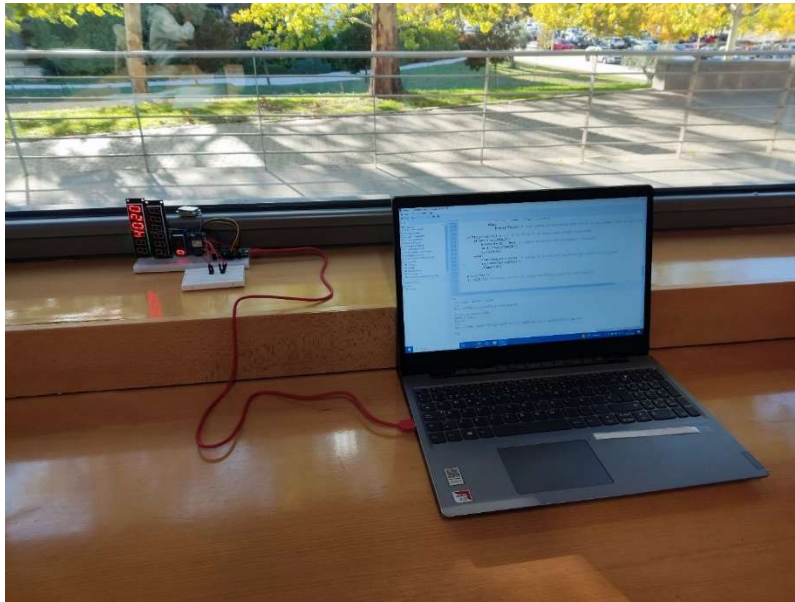


Figura 44. Se requiere de cobertura para realizar pruebas al dispositivo. Elaboración propia.

Es posible realizar una prueba de medición de dos formas. La primera y más completa es la medición dentro de un vehículo en movimiento. Esto te da información sobre la certeza de la información recogida, además de la confirmación de que funciona. La segunda forma para comprobar que funciona el cálculo de distancias es reduciendo el filtro establecido por software para la suma de distancias en la línea 114 del script main:

```
if abs_distance > 2.5 :    #Ajusta el filtro por debajo del cual no tiene en cuenta el movimiento (valor estandar 2.5)
```

Ya se ha establecido que debido al error de las mediciones de GPS las mediciones por debajo de este valor no son adecuadas para la medición en la configuración final. Pero estableciendo un valor inferior (de 0,5 metros) la propia variación de la ubicación GPS simulará que nos encontramos en movimiento y se podrán hacer pruebas desde una ubicación fija sin necesidad de contar con ayuda externa para las primeras pruebas. Posteriormente una vez calcule distancias con este método se debe de restablecer el valor estándar y realizar una prueba dinámica.

Una vez se dispone de un dispositivo funcional sobre la protoboard donde los cambios son fácilmente solucionables, se procede a una crear un primer prototipo, que no será el definitivo, que ya incluya la fuente de alimentación y todos los componentes que llevará el dispositivo final. A partir de esta fase se va a prescindir del uso de placas protoboard y se va a soldar el microcontrolador a una placa prototipos perforada que está pensada para soldar componentes a esta. Se ha decidido colocar el microcontrolador en el centro de la placa y soldar alrededor de estos terminales con tornillos para fijar las conexiones del resto de periféricos.

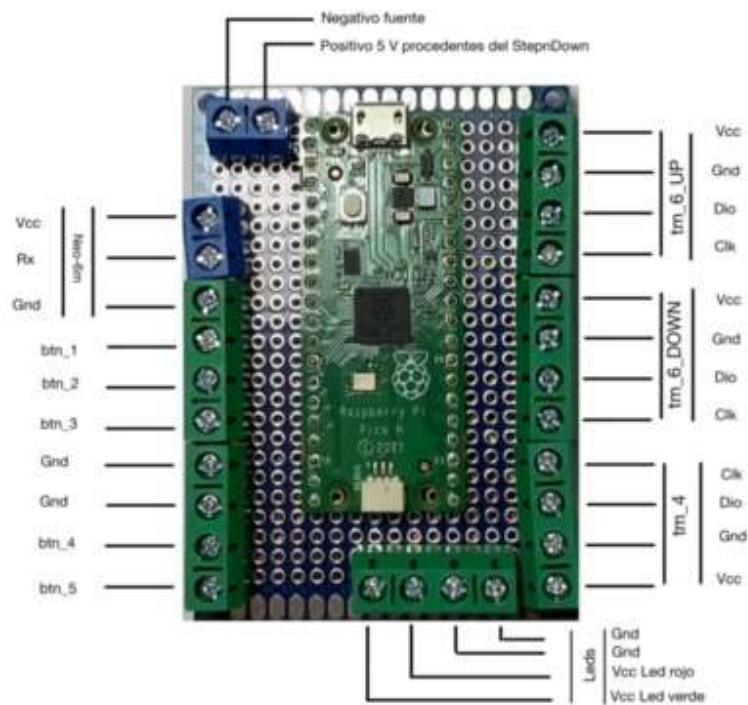


Figura 45. Placa soldada del microcontrolador. Elaboración propia.

Las conexiones entre los terminales y los pines digitales el microcontrolador se realizan por la parte inferior de la placa mediante cable macizo y soldadura con estaño. Es importante disponer los cables de tal forma que no pueda existir un contacto entre ambos.

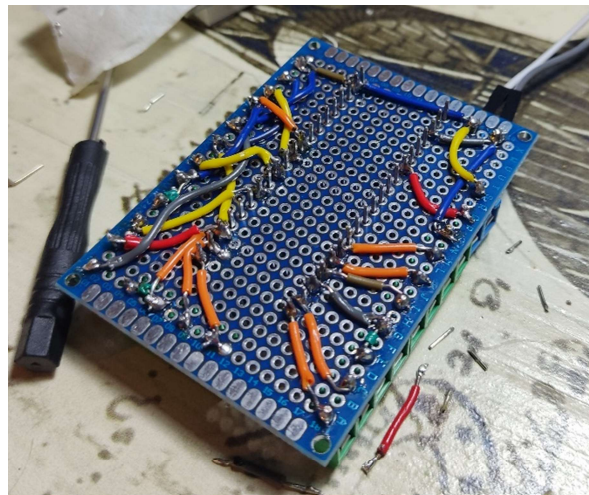


Figura 46. Conexiones soldadas del microcontrolador con los terminales. Elaboración propia.

Se va a hacer uso de una caja de registro eléctrica como carcasa de dispositivo. Sus dimensiones son de diez centímetros de ancho, diez de alto y cinco centímetros de profundidad. Este tipo de cajas están diseñadas para resguardar conexiones a la intemperie y dispone de protección IP 44.

Esto le otorga idealmente protección contra cuerpos sólidos superiores a 1mm y protección frente a salpicaduras. Algunas de estas propiedades serán mermadas debido a la apertura de orificios para posicionar los botones y las pantallas, pero es un buen punto de partida para proteger al microcontrolador.

Inicialmente se van a posicionar los dos componentes más grandes del dispositivo, que son el microcontrolador y las pantallas.

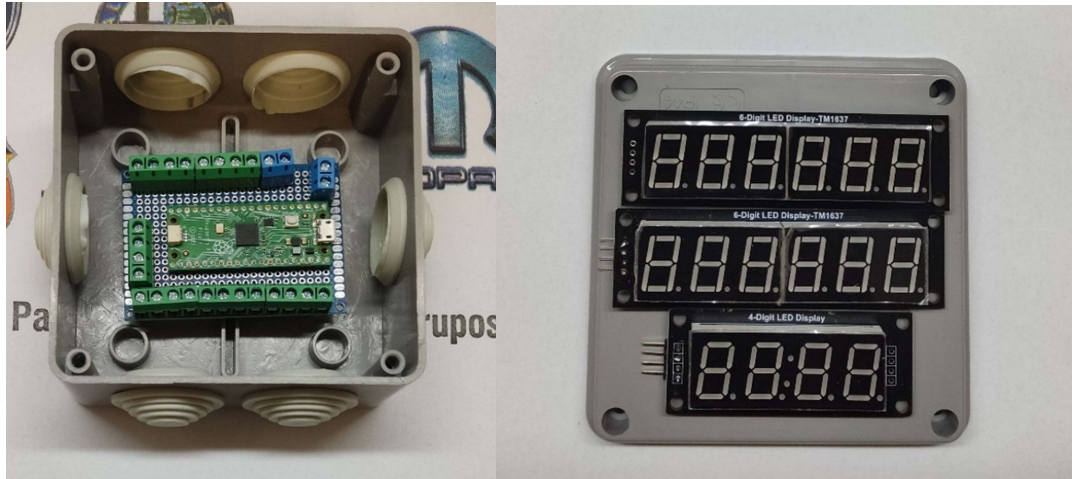


Figura 47. Disposición del microcontrolador y las pantallas. Elaboración propia.

Se cablean cada uno de los módulos al microcontrolador. Los extremos del microcontrolador se fijan mediante tornillos mientras que los extremos de los módulos se sueldan y se aíslan para evitar contactos indeseados.

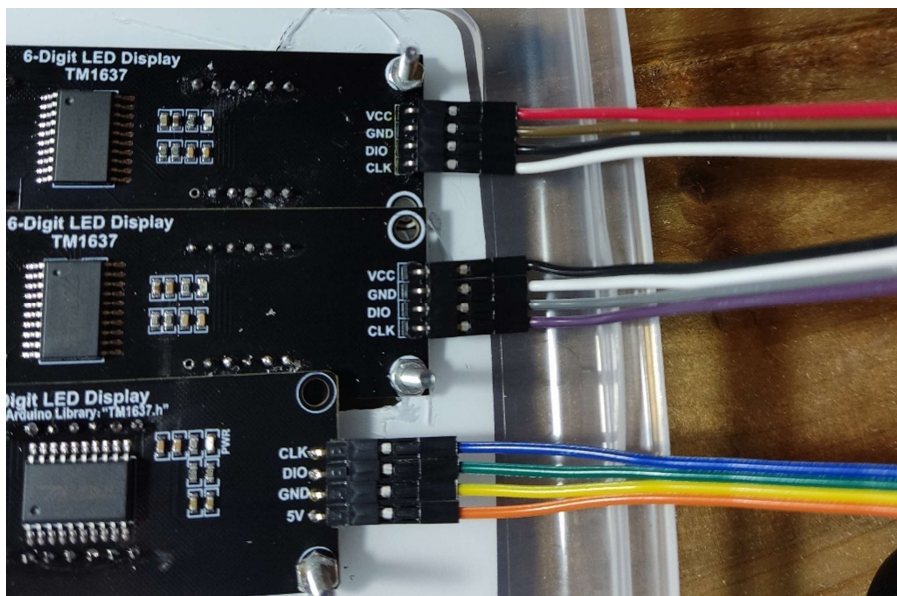


Figura 48. Conexiones de las pantallas. Elaboración propia.

Se posiciona el módulo conversor de tensión LM2596 en la parte inferior de la caja de registro y se configura en vacío antes de conectarlo al microcontrolador. Para ello se colocan los bornes del polímetro en los pines de salida del módulo. A la entrada se le suministra una tensión con la batería del vehículo en funcionamiento o de ser posible una fuente de tensión que otorgue 14 Voltios. Esta será el valor máximo suministrado por la batería de su vehículo a plena carga. Con este valor de entrada se ajusta el potenciómetro del módulo hasta obtener un valor de salida de cinco voltios. Todos los módulos de este dispositivo están preparados para operar entre 5,5 y 3,3V por lo que una disminución de un voltio en la entrada por descarga de la batería será soportada por el dispositivo. Es importante regularlo con el valor máximo de tensión prevista. El aumento de la tensión de entrada por encima del valor ajustado en el potenciómetro dañará los módulos si la tensión de salida supera los 5,5 Voltios.



Figura 49. Potenciómetro ajustable del módulo LM2596.

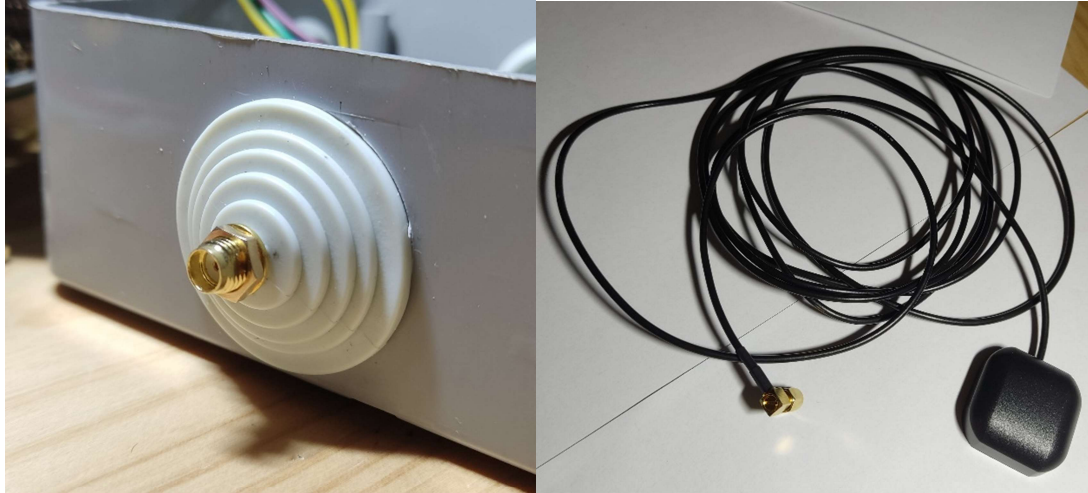
Una vez ajustado ya es posible acomodar y cablear el módulo GPS y los pulsadores. Se ha planteado el dispositivo de tal forma que todo se encuentre en el interior y las señales salgan por 3 conectores al exterior. Dos de ellos son conectores de S13 de IP 68 con dos y cuatro pines respectivamente. El conector de dos pines es el encargado de la alimentación. El otro es el encargado de la conexión con el cableado de la botonera.



Figura 50. Conectores S13. Elaboración propia.



El tercer conector es el conector de la antena del módulo GPS. Se ha montado una antena externa con un cable largo para poder colocar la antena en la parte superior del vehículo para mejor conexión. Esta antena es resistente al agua y está aislada del interior de la caja.



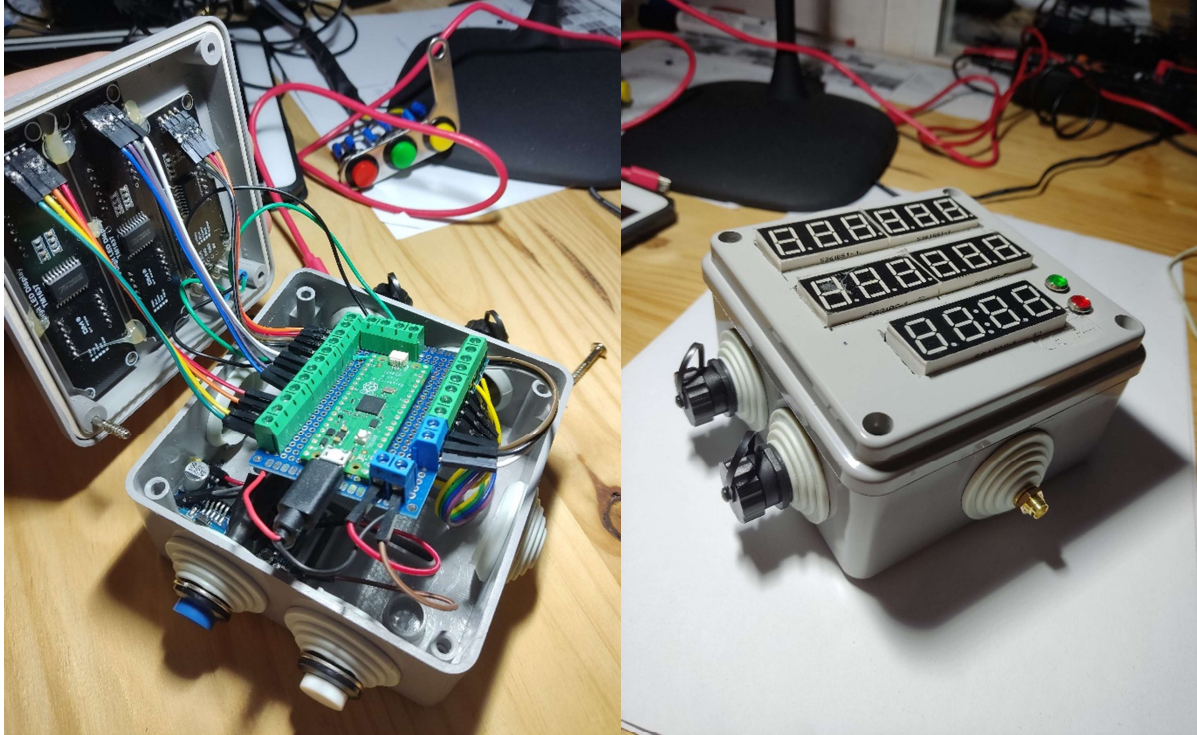
*Figura 51. Conector de la antena externa. Elaboración propia.*

La botonera se ha cableado mediante un cable de cuatro hilos de 50 cm de longitud. Esto otorga una posición cómoda para su uso por el copiloto de un coche o en una motocicleta. Se han instalado los tres botones en una placa que permite el anclaje en el manillar de una motocicleta mediante un tornillo. Las conexiones se han realizado con soldadura en los pines del conector S13 y con conectores crimpados en el extremo de la botonera.



*Figura 52. Cable y conector de la botonera. Elaboración propia.*

El dispositivo permanecerá cerrado salvo para labores de diagnosis o fallo. El cierre es producido por una junta y fijado con cuatro tornillos en la parte superior de la caja.



*Figura 53. Dispositivo final. Elaboración propia*

## 5.9. Costes de fabricación

Este dispositivo ha sido diseñado y fabricado por la misma persona sin necesidad de la subcontratación de otras personas. Se ha estimado el coste de la mano de obra en base al XX Convenio Colectivo Nacional de Empresas de Ingeniería para un graduado en Ingeniería Industrial en el año 2022 (13) . Esto establece en un valor de 11,55€ la hora de trabajo. En base a estos datos los costes son los siguientes:

Tabla 8. Desglose de costes de mano de obra en diciembre de 2023.

Tarea	Horas dedicadas	Coste (€/h)	Total
Diseño conceptual	4	11,55	46,2
Investigación de la competencia	2	11,55	23,1
Planteamiento de requisitos técnicos	2	11,55	23,1
Elección del hardware	3	11,55	34,65
Considerar simplificaciones y acotar errores de precisión	8	11,55	92,4
Búsqueda de librerías software	2	11,55	23,1
Plantear la estructura de los scripts	3	11,55	34,65
Adaptar librerías existentes	2	11,55	23,1
Desarrollo de scripts propios	50	11,55	462
Test unitarios de los componentes	2	11,55	23,1
Construcción de prototipo en protoboard	1	11,55	11,55
Test del dispositivo en estado de protoboard	1	11,55	11,5
Diseño del prototipo final	4	11,55	46,2
Construcción del prototipo	6	11,55	69,3
Diseño Carcasa	3	11,55	34,65
Ensamblaje y ajustes finales	2	11,55	23,1
<b>TOTAL</b>			<b>981.70</b>

Este valor representa los costes humanos en labor de diseño y fabricación. Anexos a estos se encuentra los costes de los materiales y módulos necesarios para la fabricación del dispositivo y se muestran en la siguiente tabla:

Tabla 9. Desglose de precios en diciembre de 2023

	Unidades	Precio Ud. (€)	Importe
<b>Pantallas</b>			
Módulo de pantalla LED TM1637, 0.56 pulgadas, 6 dígitos	2	3,28	<b>6,56</b>
Módulo de pantalla LED TM1637, 0.56 pulgadas, 4 dígitos	1	2,13	<b>2,13</b>
<b>Microcontroladores</b>			
Raspberry Pi Pico	1	4,75	<b>4,75</b>
<b>Módulos GPS</b>			
U-blox Neo-6m	1	17,75	<b>17,75</b>
<b>Botones</b>			
Pulsador 16mm	5	0,29	<b>1,45</b>
Interruptor 12mm	1	0,20	<b>0,20</b>
<b>Leds</b>			
Led indicador 3V	2	0,73	<b>1,46</b>
<b>Convertidor de potencia</b>			
LM2596 Step Down	1	1,10	<b>1,10</b>
<b>Conectores</b>			
Conector SP13 IP68 2 Pines	1	1,76	<b>1,76</b>
Conector SP13 IP68 4 Pines	1	1,76	<b>1,76</b>
<b>Soportes</b>			
Soporte Universal 3 botones	1	2,24	<b>2,24</b>
Carcasa	1	3,50	<b>3,50</b>
<b>Cableado</b>	1	2	<b>2</b>
<b>Placa prototipos</b>	1	0,31	<b>0,31</b>
<b>TOTAL</b>			<b>46,97 €.</b>

El coste del conjunto de materiales en diciembre del año 2023 supone un total de 46,97 €. Este valor es el verdaderamente relevante debido a que será el futuro coste que debe afrontar un interesado en optar por esta solución para iniciarse en los raids de regularidad de clásicos. Al haber hecho uso de código open hardware y software de programación gratuito no existen gastos de licencias o adquisición de aplicaciones. Esto ubica el coste de creación del dispositivo en un precio ligeramente mayor que los odómetros de bicicleta y muy por debajo de los equipos de competición cumpliendo así el objetivo principal de su creación.

Tabla 10. Coste de los dispositivos disponibles

Cuentakilómetros Bicicleta Van Rysel	Dispositivo del proyecto	Terratrip 303 Road Survey PLUS V4	Terratrip 202 GeoTrip with GPS & GLONASS V5	Kit de Navegación HOBBY F2r
11,99 €	46,97 €	284,95 €	422,95 €	913,00 €

En conjunto el diseño y fabricación de este producto se valora en 1028,67 € y un total de 95 horas de trabajo.

## 5.10. Consumo eléctrico

Exigir un consumo eléctrico extra muy elevado al sistema eléctrico del vehículo podría llegar a suponer un problema llegado el caso si no se redimensiona la batería, es estator o alternador y el regulador de corriente. Es importante por tanto hacer un cálculo del consumo del dispositivo. Los componentes principales de este y sus respectivos consumos energéticos en condiciones de operación son los siguientes:

*Tabla 11. Potencia consumida*

<b>Componente:</b>	<b>Consumos a plena carga(W)</b>
<b>Raspberry Pi Pico</b>	0,47 W
<b>Módulo de pantalla LED TM1637, 0.56 pulgadas, 4 dígitos</b>	0,06 W
<b>Módulo de pantalla LED TM1637, 0.56 pulgadas, 6 dígitos</b>	0,08 W
<b>GPS Neo-6m</b>	0,23 W
<b>Fuente de alimentación</b>	0,2 W
<b>TOTAL</b>	<b>3.1 W</b>

El conjunto del dispositivo desarrollará entonces 3,1 W generando unos consumos perfectamente asumibles para cualquier vehículo sin necesidad de redimensionar la instalación eléctrica de este.

## 5.11. Limitaciones

Si bien el objetivo fundamental de este dispositivo es disminuir los costes de fabricación de un odómetro GPS, este diseño accesible crea una serie de limitaciones en cuanto a su funcionamiento y precisión.

Las más relevantes son:

- Grado de protección IP: Esto es referido a la protección de los equipos situados en el interior de la envolvente contra los efectos perjudiciales ocasionados por la penetración de agua y a la penetración de cuerpos sólidos extraños (14). Este dispositivo está concebido para su utilización en un entorno seco o de baja humedad, pero no para funcionar bajo la lluvia o ser sumergido. Los conectores del dispositivo con el exterior si son resistentes a salpicaduras con un IP 68, pero no así las pantallas de los displays. Esto se podría solucionar en otra variante del dispositivo colocando una lámina transparente sobre las pantallas y sellando la unión con silicona. Con esto se conseguiría mayor estanqueidad, pero trae problemas a la hora de visualizar correctamente la información en pantalla debido a los reflejos.
- Cobertura GPS: La cobertura que proporciona el módulo Neo-6m es óptima en entornos abiertos, pero induce a errores mayores en entornos cerrados o bajo techo. Eso no es un problema en la mayoría de las situaciones, pero es una limitación en el paso de túneles frente a los sistemas de medición de pulsos en las ruedas.
- Precisión y tasa de refresco: Este dispositivo está concebido para su uso en rally de regularidad de clásicos para aficionados donde las velocidades son bajas y la exigencia de orientación toleran un error del orden de  $\pm 10$  metros. A mayores velocidades, la tasa de refresco y precisión podría resultar insuficiente. Estas nuevas especificaciones afectarían a todas las decisiones de diseño del dispositivo y sería necesario establecer nuevos requerimientos técnicos para determinar si podría adaptarse este dispositivo con el cambio de sus componentes internos o se debería partir desde cero. La tasa de refresco del GPS o la capacidad de cálculo del microcontrolador son factores limitantes en el diseño actual.

## 5.12. Mejoras futuras

Este dispositivo está fabricado mediante componentes comerciales basados en open hardware de libre acceso. Si en un futuro se buscase explotar este producto de manera comercial, sería más interesante la implementación de ciertas mejoras con el fin de mejorar el producto y agilizar su fabricación. Las principales estudiadas son:

- Desarrollo placa de circuito impreso (PCB):

La utilización de un circuito impreso para realizar las conexiones otorgaría mayor robustez al dispositivo y una mejor resistencia ante vibraciones y suciedad. También reduciría la cantidad de cable requerido y el tiempo de fabricación. El diagnóstico de fallos sería más sencillo en caso de fallo y se podría disminuir el tamaño del dispositivo. La implementación de esta mejora es especialmente interesante si se va a fabricar una alta cantidad de dispositivos. El coste de diseñar y fabricar un único circuito impreso es muy alto, pero se reduce considerablemente el coste por unidad en tiradas largas.

- Implementación de un zumbador:

En la etapa de pruebas se ha descubierto que es difícil reaccionar rápidamente en caso de pérdida de la cobertura. Cuando esto ocurre en pantalla aparece un mensaje que dice “Buscando satélites” pero, al tener la vista en la carretera, pasa un tiempo hasta que el usuario puede reaccionar. Un zumbador llamaría inmediatamente la atención del piloto sin necesidad de estar constantemente mirando las pantallas. Su implementación podría realizarse a través de las salidas digitales disponibles del microcontrolador.

- Mejora de la visibilidad de los dígitos utilizando tecnología LCD:

La visibilidad de las pantallas es buena en casi todas las condiciones de operación, pero se queda algo justa en cuanto a brillo en motocicletas en condiciones en las que la luz del sol refleja sobre las pantallas. Para solucionar este problema se podría buscar la manera de implementar pantallas LCD retroiluminada resistentes al agua, permitiendo mejorar la resistencia a el agua del dispositivo.

## 6. Conclusiones

Este TFG ha presentado el desarrollo de un dispositivo odómetro accesible, para su uso en raids de regularidad amateur. El desarrollo de este dispositivo requirió de un estudio de los requerimientos necesarios así como de determinar los módulos necesarios para satisfacer los requerimientos técnicos.

Se ha marcado con una estrella el posicionamiento de este dispositivo en el mercado actual de los odómetros para rally de regularidad de clásicos. Este es un dispositivo programable que otorga funcionalidades cercaras a los complejos equipos de competición de una forma más intuitiva y con un menor coste satisfaciendo el objetivo general de este proyecto.

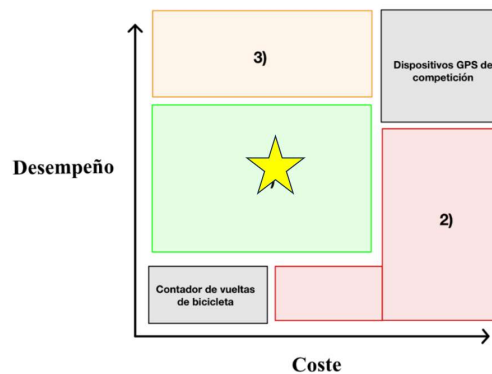


Figura 54. Posicionamiento del dispositivo final en el mercado. Elaboración propia

En condiciones de operación este dispositivo logra situar el error de medición por debajo de los 10 metros asumiendo un ajuste periódico cada 500 metros para su ajuste con el roadbook. Es una precisión suficiente para un aficionado que comienza en este tipo de competiciones. Su funcionamiento es sencillo y su visualización se basa en dos pantallas donde se muestra la información y en tres botones principales para el ajuste de las distancias.

El dispositivo tiene unas dimensiones de 11 x 11 x 6 cm y requiere de una toma de corriente de 12 V del vehículo sin necesidad de sobredimensionar ninguna parte del sistema eléctrico de este. Se hace uso de un módulo regulador de tensión regulable para adaptar la tensión a los 5V que requiere el dispositivo. No requiere de acceso a internet ni de ningún otro dispositivo para funcionar. El dispositivo consiste en un micro Raspberry Pi Pico y un módulo GPS Neo-6m integrado y controlado por el micro. Permite visualizar la información en tres displays, dos de ellos destinados para los trips son displays tm-1637 de siete dígitos y un tercer display de 4 dígitos tm-1637 reservado a otras funciones auxiliares.

Gracias a la utilización de material Open Hardware y el acceso a los repositorios cada persona puede adaptar este dispositivo a sus necesidades y eventualmente ser compartirlo con la comunidad.



## 7. Trabajos futuros

Este dispositivo es una buena alternativa a los equipos ofertados actualmente para este tipo de pruebas, pero no es única. Hay lugar para la creación de nuevas soluciones en otro rango de complejidad y coste dentro de la zona verde de esta figura.

La posibilidad de actualizaciones del código en el repositorio y el diseño modular del dispositivo permite la implementación de mejoras futuras. Primeramente, a nivel software se podrán subir actualizaciones del programa que podrán instalarse en dispositivos con anteriores versiones mediante la toma USB del microcontrolador. Esto permitirá junto con los botones y el display reservados implementar futuras actualizaciones y funcionalidades. De la misma forma se podrán actualizar la documentación del repositorio.

En segundo lugar, se podría actualizar el dispositivo a medida que los componentes electrónicos obtengan mejores prestaciones o sean más accesibles. Esto también quedaría documentado con sus respectivos diagramas eléctricos y manuales.

Los repositorios de la plataforma GitHub permiten a los usuarios solicitar la actualización de los repositorios en base a sus códigos. Esto permitiría el crecimiento del dispositivo por parte de la comunidad adaptándolo a las necesidades futuras.

## 8. Bibliografía

1. **MUNOZ, Felipe.** Estancamiento de las ventas mundiales de automóviles en 2023. *Jato Dynamics*. [En línea] 2023. [Citado el: 4 de Enero de 2023.] <https://www.jato.com/spain/>.
2. **DIAZ, Borja.** ¿Cuál es el coche más barato de España?. Revista Car and Driver, 11 de Enero de 2024.
3. **BERNÉ VALERO, José Luis, GARRIDO VILLÉN, Natalia y CAPILLA ROMÁ, Raquel.** *GNNS*. Valencia : Editorial de la Universidad Politécnica de Valencia, 2019. 9788490487778.
4. **CERVH. (Real Federación Española de Automovilismo).** Reglamento Técnico del Campeonato de España de Rallyes de Regularidad para Vehículos Históricos. 2023.
5. **CERVH. (Real Federación Española de Automovilismo).** Reglamento Deportivo del Campeonato de España de Rallyes de Regularidad para Vehículos Históricos. 2023.
6. **Raspberrypi.** [En línea] 10 de 2023. [Citado el: 9 de 12 de 2023.] <https://www.raspberrypi.com/documentation/microcontrollers/raspberry-pi-pico.html>.
7. **Arduino. Guía de Referencia de Arduino. Arcuino.cc.** [En línea] 2023. [Citado el: 6 de 12 de 2023.] <https://www.arduino.cc/reference/es/>.
8. **MONK, Simon.** *Programming Arduino Next Steps: Going Further with Sketches*. United States : McGraw-Hill Education, 2018. 1260143244.
9. **KAPLAN, Elliot.** *Understanding GPS/GNSS: Principles and Applications*. Editorial Artech House. 2017. ISBN: 9781630810580.
10. **Ministerio de transportes y movilidad sostenible.** Secretaria de Estado de planificación e infraestructuras. [En línea] 27 de Diciembre de 1999. [Citado el: 12 de Diciembre de 2023.] <https://www.fomento.gob.es/AZ.BBMF.Web/documentacion/pdf/RE2038.pdf>.
11. **PETERSON, Zachariah.** I2C, SPI y UART: Cómo hacer el layout de estos buses comunes. . [En línea] 2020. <https://resources.altium.com/es/p/i2c-vs-spi-vs-uart-how-layout-these-common-buses>.
12. **CUEVAS, Alberto.** Python 3. 2016.Madrid: Editorial RA-MA. ISBN 84-9964-359-0.
13. **Ministerio de Trabajo y Economía Social.** *Resolución de 27 de febrero de 2023, de la Dirección General de Trabajo, por la que se registra y publica el XX Convenio colectivo nacional de empresas de ingeniería; oficinas de estudios técnicos; inspección, supervisión y control técnico y de calidad*. 2022.

14. **Comité técnico CTN 200.** Normas básicas eléctricas. *UNE-EN 60529*. Abril 2018.
15. **MathWorks.** MathWorks. [En línea] 2024. [Citado el: 15 de Enero de 2024.]  
<https://es.mathworks.com/help/supportpkg/arduino/ref/read-serial-data-from-a-gps-shield-using-arduino-hardware.html>.
16. **VALDES, Fernando.** Microcontroladores Fundamentos y Aplicaciones con PIC. 2007, Editorial Marcombo. ISBN: 9788426714145.
17. **DE CASTRO, Miguel.** EL ALTERNADOR. 2001, Editorial CEAC. ISBN: 9788432911507
18. **BOUCHEN, VINCENTY.** Aprender la programación orientada a objetos con el lenguaje Python. 2021. Editorial ENI. ISBN: 9782409031564.
19. **CORTÉS, Fernando Reyes.** Robótica: Control de manipuladores. 2020. Editorial: Marcombo. ISBN: 9788426729620

## 9. Apéndices

A continuación, se encuentran los apéndices a este trabajo. Está formado por el enlace al repositorio que contiene el código del programa junto con otra información que se ha considerado importante a la hora de realizar el proyecto como fichas técnicas de los módulos, desgloses de precios a día de la entrega de este trabajo y diagramas a mayor tamaño.

### Apéndice A: Repositorio GitHub.

Se ha creado un repositorio donde se ha publicado el código del microcontrolador con todo lo necesario para el correcto funcionamiento del dispositivo. Estos programas están optimizados para su uso en un microcontrolador Raspberry Pi Pico configurada con MicroPython.

<https://github.com/Alvarsanz/DIY-rally-raid-odometer.git>

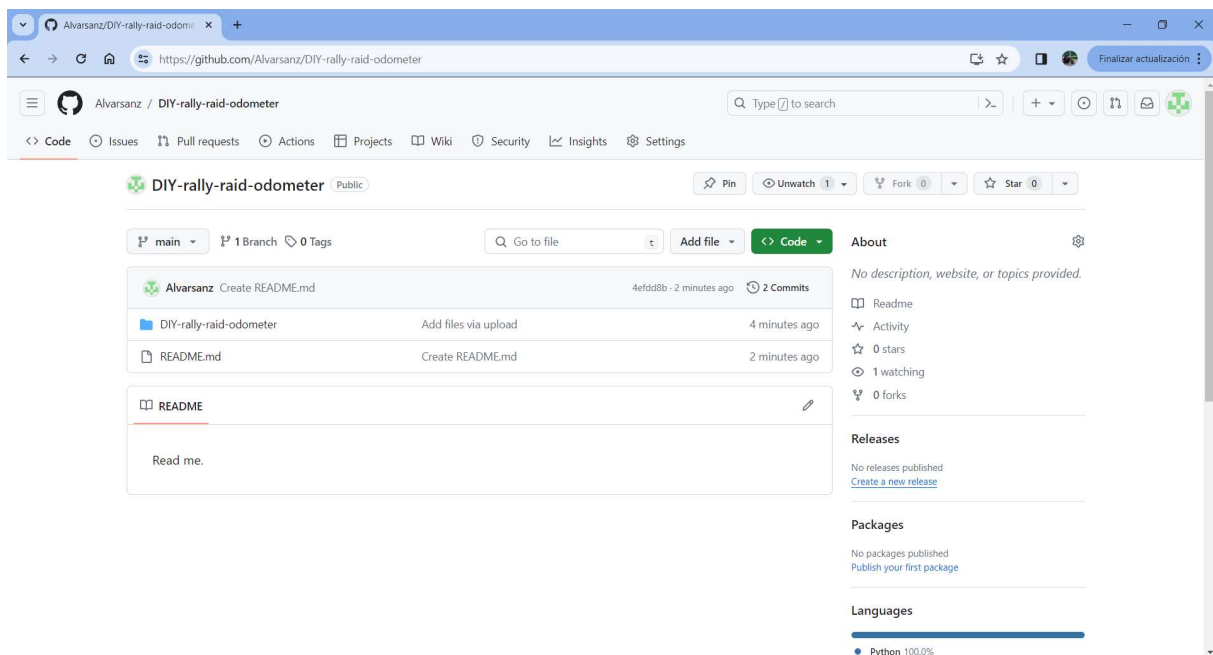


Figura 55. Repositorio del código Python

La creación de un repositorio permitirá efectuar mejoras y optimizar el código, así como subir todos los manuales necesarios para la construcción del dispositivo.



## Apéndice C: Fichas técnicas.

Se adjuntan las fichas técnicas de los módulos utilizados en el dispositivo final. Aquí aparecen sus características técnicas con el fin de facilitar su obtención en caso de buscar replicar el dispositivo.:

Manual técnico Raspberry Pi Pico:

<https://www.raspberrypi.com/documentation/microcontrollers/raspberry-pi-pico.html> .

Manual técnico u-blox 6:

[https://content.u-blox.com/sites/default/files/products/documents/LEA-NEO-MAX-6\\_HIM\\_%28UBX-14054794%29\\_1.pdf](https://content.u-blox.com/sites/default/files/products/documents/LEA-NEO-MAX-6_HIM_%28UBX-14054794%29_1.pdf) .

Manual técnico tm1637:

<https://www.alldatasheet.es/datasheet-pdf/pdf/1133627/TITAN/TM1637.html> .

Raspberry Pi Pico Python SDK. A MicroPython environment for RP2040 microcontrollers.:

<https://www.raspberrypi.com/documentation/microcontrollers/micropython.html>

## Apéndice D: Desglose de precios a fecha de entrega del proyecto.

Los precios en los que se ha basado este documento y en torno a los cuales se ha realizado el estudio del mercado se han tomado a fecha del 1 de marzo de 2024.

### Precios comerciales

*Tabla 12. Desglose de precios unitarios*

<b>Pantallas</b>	
Módulo de pantalla LED TM1637, 0.56 pulgadas, 6 dígitos	<b>3.28 €</b>
Módulo de pantalla LED TM1637, 0.56 pulgadas, 4 dígitos	<b>2.13 €</b>
Módulo HT1621 LCD de 2,4 pulgadas, 6 dígitos y 7 segmentos.	<b>7.23 €</b>
Módulo WH1602L1, 122x44mm, LCD1602	<b>8.74 €</b>
<b>Microcontroladores</b>	
Raspberry Pi Pico	<b>4.75€</b>
Arduino Nano	<b>18.36 €</b>
ESP 32	<b>9.95 €</b>
<b>Módulos GPS</b>	
Adafruit GPS Breakout V3	<b>61.66 €</b>
U-blox Neo-6m	<b>17.75 €</b>
<b>Botones</b>	
Pulsador 16mm	<b>0.29 €</b>
Interruptor 12mm	<b>0.20 €</b>
<b>Leds</b>	
Led indicador 3V	<b>0.73 €</b>
<b>Convertidor de potencia</b>	
LM2596 Step Down	<b>1.10 €</b>
<b>Conectores</b>	
Conector SP13 IP68 2 Pines	<b>1.76 €</b>
Conector SP13 IP68 4 Pines	<b>1.76 €</b>
<b>Soportes</b>	
Soporte Universal 3 botones	<b>2.24 €</b>
<b>Placas de prototipos</b>	
Placa de prototipos 10 x 7 cm	<b>0.31 €</b>