



UNIVERSIDAD DE LA REPÚBLICA
FACULTAD DE INGENIERÍA



Plataforma de geolocalización e investigación en confinamiento virtual de bovinos

MEMORIA DE GRADO PRESENTADA A LA FACULTAD DE
INGENIERÍA DE LA UNIVERSIDAD DE LA REPÚBLICA POR

Néstor Acosta, Nicolás Barreto, Pablo Caitano

EN CUMPLIMIENTO PARCIAL DE LOS REQUERIMIENTOS
PARA LA OBTENCIÓN DEL TÍTULO DE
INGENIERO ELECTRICISTA.

TUTOR

Julián Oreggioni Universidad de la República

TRIBUNAL

Germán Capdehourat Universidad de la República
Pablo Castro Universidad de la República
Leonardo Steinfeld Universidad de la República
Julián Oreggioni Universidad de la República

Montevideo
8 de julio de 2019

Plataforma de geolocalización e investigación en confinamiento virtual de bovinos,
Néstor Acosta, Nicolás Barreto, Pablo Caitano.

Esta tesis fue preparada en L^AT_EX usando la clase iietesis (v1.1).
Contiene un total de 123 páginas.
Compilada el miércoles 13 noviembre, 2019.
<http://iie.fing.edu.uy/>

Agradecimientos

De manera colectiva, queremos agradecer a nuestro tutor, Julián Oreggioni, por apoyarnos desde un primer momento y hasta el final del proyecto con todo lo que precisamos. Por su apoyo que sobrepasaba cualquier obligación, estamos profundamente agradecidos.

De manera colectiva, queremos agradecer al IIE, por abrirnos sus puertas y brindarnos un espacio.

De manera colectiva, queremos agradecer a Francisco Cabrera y Germán Capdehourat por todo lo que hicieron por nosotros. Por el convenio con el IIE para dejar un Gateway de Teliot en el laboratorio de proyectos, por todas las consultas que nos respondieron sin tener por qué, y por la buena voluntad que demostraron en todo momento con nosotros.

De manera colectiva, queremos agradecer a la Agencia Nacional de Investigación e Innovación por la financiación de este proyecto

Queremos agradecer a nuestras familias por todo el apoyo que nos brindaron. Sin ustedes, esto no sería posible.

Finalmente, quiero dedicarle unas palabras a mi abuelo, Eduardo Caitano. Gracias por inculcarme la importancia del estudio. Gracias por siempre estar cuando te precisé. Y gracias por tantas veces que no te precisé, pero igual estuviste ahí para ayudarme. Gracias

Esta página ha sido intencionalmente dejada en blanco.

A Eduardo Jorge Caitano Martirena,

Esta página ha sido intencionalmente dejada en blanco.

Resumen

Los sistemas de geolocalización y seguimiento de ganado bovino son útiles para prevenir el abigeato, la detección de enfermedades y el traspaso de animales hacia predios linderos, siendo éstos un primer paso hacia un sistema integral de gestión para establecimientos ganaderos. Un siguiente paso sería agregar el confinamiento virtual, lo que podría otorgar una significativa reducción de costos operativos y podría revolucionar la forma en que se manejan los animales hoy en día.

Existen varios antecedentes de alambrados virtuales que utilizan diversas técnicas para mantener confinado al ganado dentro de un perímetro configurado de forma remota. Estas técnicas comparten un patrón común que consiste en colocar un dispositivo electrónico en el animal capaz de aplicar estímulos cuando éste se acerca a los límites pre-establecidos. El uso de descargas eléctricas como método de estimulación es ampliamente utilizado.

Este proyecto propone una solución compatible con el bienestar animal, que evite las descargas eléctricas, basada solamente en estímulos sonoros y táctiles (mediante un motor vibrador). Para ello, se desarrolló un sistema compuesto por: un dispositivo electrónico que se coloca en el cuello del animal con capacidad de estimular y enviar información en forma inalámbrica; un sistema central que es capaz de recibir y procesar esa información; y una interfaz gráfica, a través de la cual se puede visualizar la posición del animal y sus movimientos de manera remota. También permite la configuración de distintos parámetros de interés del sistema, pudiendo evaluar así diversas metodologías de confinamiento.

El dispositivo electrónico está compuesto por un módulo Moteino (integrado por un microcontrolador ATmega y un módulo de comunicación LoRa), un módulo GPS, una batería de Li-Ion, paneles solares, un módulo de gestión de la carga de energía, un buzzer y un motor vibrador. El software embebido está basado en el lenguaje de programación C++, potenciado con las funciones de la plataforma de Arduino, y la librería LMIC de IBM para la utilización de LoRaWAN.

Las pruebas realizadas determinaron que el sistema de comunicación ofrece un alcance de 9 km en línea vista y se reduce a 1.6 km en condiciones no tan favorables. La posición del animal se reporta una vez por segundo con una precisión de aproximadamente 2 metros. El collar contiene un módulo con la electrónica de 103 x 64 x 33 mm y un panel solar de 96 x 54 x 3 mm. El costo de materiales por collar es de menos de 7 dólares y su consumo promedio es menor a 62 mA. Si bien esto significa que no se alcanza la autonomía requerida en escenarios donde no se pueda cosechar suficiente energía solar, se plantean varias soluciones para disminuir el consumo del dispositivo.

Se logró crear una plataforma de investigación para el confinamiento virtual de animales, con los estímulos antes mencionados y cuya funcionalidad fue verificada, cumpliendo así con la gran mayoría de los criterios de éxito definidos al inicio de este proyecto. Las pruebas realizadas en animales, sugieren que los estímulos utilizados no son inocuos, por lo que se estima, podrían lograr su cometido luego de un periodo de aprendizaje de los animales. La investigación sobre la eficacia de las metodologías de confinamiento de la plataforma desarrollada deberá ser realizada en una próxima etapa por un equipo interdisciplinario.

Palabras Clave: Perímetro virtual; confinamiento; geolocalización; estímulos; bienestar animal.

Tabla de contenidos

Agradecimientos	I
Resumen	v
1. Generalidades	1
1.1. Motivación y descripción del problema	1
1.2. Antecedentes	3
1.2.1. Evolución de los métodos de confinamiento de ganado	3
1.2.2. Perímetros virtuales y adiestramiento	5
1.3. Problema a resolver	7
1.4. Alcance y criterios de éxito	8
1.5. Solución propuesta	9
1.6. Apoyos recibidos y colaboraciones	11
1.6.1. Programa FFI: Proyectos de Innovación de Fin de Carrera .	12
1.6.2. Colaboración con Grupo HCL del INCO	12
1.6.3. Colaboración de Facultad de Veterinaria	12
1.6.4. Colaboración de Teliot	13
2. Hardware	15
2.1. Diagrama de bloques del hardware utilizado	15
2.2. MoteinoMEGA	16
2.3. GPS	16
2.4. Buzzer	17
2.5. Motor	18
2.6. Paneles solares	18
2.6.1. Corriente promedio consumida	19
2.6.2. Dimensionado de los paneles	19
2.7. Baterías	20
2.8. Módulo de carga TP4056	21
3. Software	23
3.1. LoRa - LoRaWAN	23
3.2. LMiC	27
3.3. Arquitectura de Software	30
3.3.1. Geolocalización	30

Tabla de contenidos

3.3.2.	Configuración del GPS	31
3.4.	Variables de interés del Perímetro Virtual	32
3.4.1.	Funcionamiento esperado por zonas	37
3.4.2.	Seteo de parámetros	38
3.5.	Perímetro Virtual	41
3.6.	Funciones utilizadas para la implementación del perímetro virtual	42
3.6.1.	Funciones de la LMiC	43
3.6.2.	Funciones de la librería TinyGPS++	43
3.6.3.	Funciones implementadas	44
3.7.	Interfaz web	45
4.	Diseño del encapsulado	47
4.1.	Requerimientos	47
4.2.	Método de sujeción	48
4.3.	Diseño	49
4.3.1.	Encapsulado principal	49
4.3.2.	Encapsulado del motor	50
4.3.3.	Contrapeso	50
4.4.	Fabricación	50
5.	Pruebas de desempeño y resultados	53
5.1.	Estudio del consumo energético del dispositivo	53
5.1.1.	Medidas de consumo	53
5.1.2.	Mejoras a futuro	55
5.2.	Prueba de alcance de la comunicación dispositivo - gateway	56
5.3.	Prueba de variables de interés del PV	58
5.3.1.	Prueba de zonas	60
5.3.2.	Prueba de refractario	63
5.3.3.	Prueba de velocidad	64
5.3.4.	Pruebas buzzer	66
5.3.5.	Pruebas para verificar el resto de las variables	67
5.4.	Pruebas en ganado bovino	68
5.4.1.	Prueba en Facultad de Veterinaria	68
5.5.	Análisis de costos	69
6.	Conclusiones	71
6.1.	Criterios de éxito	71
6.2.	Principales problemas encontrados	72
6.3.	Principales aportes	73
6.4.	Trabajo a futuro	74
A.	Resultados de pruebas de perímetro virtual	77
B.	Tablas de costos	91
C.	Manual de usuario de la Interfaz web	95

Tabla de contenidos

D. Prueba de concepto en Casupá, Florida	99
Referencias	101
Índice de tablas	104
Índice de figuras	106

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 1

Generalidades

1.1. Motivación y descripción del problema

Uruguay es un país ganadero por excelencia. A modo de ejemplo, según el INAC¹ sobre la actividad ganadera en 2014 [1]:

- Hay 13.2 millones de hectáreas dedicadas a la actividad ganadera.
- 50.576 establecimientos poseen bovinos y/u ovinos.
- La actividad ganadera ocupa el 80 % de la superficie del país.
- Uruguay es el país con más cabezas vacunas por habitante del mundo (3,5 cabezas/habitante).
- El 70 % de la producción de carne bovina es exportado.

Respecto a la información macroeconómica del 2014, se destacan del anuario estadístico agropecuario del 2015 los siguientes datos [2]:

- La producción agropecuaria, por subsector, en porcentaje sobre dólares corrientes es: pecuaria 47.4 %, agricultura 46.9 %, silvicultura 5.7 %.
- El subsector pecuario o ganadero aportó en 2014 el 5.64 % del PBI uruguayo (3248 millones de dólares corrientes movilizados por la industria pecuaria).

Los datos anteriores muestran que existe un mercado de gran tamaño, lo cual es una de las condiciones necesarias para el éxito de un emprendimiento en el sector.

Dentro de esta industria, las formas de producción son la ganadería intensiva y extensiva.

La ganadería extensiva es la más utilizada en el Uruguay, debido principalmente a sus menores costos. Controlar a los bovinos implica un gran esfuerzo por parte de los productores y tiene altos costos asociados en el caso de ganadería intensiva. La

¹Instituto Nacional de Carnes.

Capítulo 1. Generalidades

inversión en ganadería extensiva es sustancialmente menor. Sin embargo, la menor inversión viene aparejada con una falta de control muy grande. Esto tiene varias consecuencias:

- Abigeato.
- Traspaso de ganado hacia predios linderos.
- Problemas asociados a enfermedades que puedan contraer los animales.
- Animales empantanados en barro por períodos prolongados.
- Mantenimiento y costos asociados al alambrado físico son muy altos, debido a que los terrenos son extensos.

Los revolucionarios avances científicos y técnicos de los últimos años permiten actualmente contar con tecnologías de localización remota, y este proyecto propone utilizarlas con el fin de proveer de facilidades al sector ganadero a la hora de gestionar y controlar uno de sus recursos más importantes. En concreto, se pretende detectar en tiempo real la ubicación de los animales dentro de un predio y su visualización remota en un sistema informático. Una consecuencia inmediata de resolver este problema, es la detección del abigeato, el traspaso de animales hacia otros predios, y la detección de que un animal no se mueve hace mucho tiempo (lo cual puede estar asociado a una enfermedad, a que quedó atrapado, entre otras).

Antes de proponer este proyecto entrevistamos a empresarios del rubro, y los entrevistados estaban dispuestos a pagar hasta 40 USD por vaca (una vez, y luego costos por mantenimiento bajos). El motivo de la inversión sería qué en el largo plazo les evita la pérdida de mucho más dinero. Dado que quienes se dedican a la industria agropecuaria suelen tener estos temas bastante claros, esto nos motivó a proseguir con la idea. El análisis de los datos provenientes de la geolocalización abre la posibilidad a aplicaciones de diversa índole, como la detección de enfermedades vinculadas a patrones de movimiento. Sin embargo, lo que puede aportar un diferencial a este proyecto es la realización de un Perímetro Virtual (PV), concepto cuyo origen se verá en la sección 1.2, junto a una recopilación de datos acerca de las experiencias internacionales relacionadas. Si bien a nivel mundial el PV no ha sido estudiado de forma exhaustiva, se pueden encontrar en la web diversos papers y material sobre la temática.

El PV consiste en mantener al ganado dentro de un perímetro definido por software, sin la presencia de un alambrado físico. Esto se logra aplicando estímulos de diversa índole al animal en caso de que se acerque a los límites del perímetro fijado.

En caso de que los estímulos propuestos, combinados con una adiestración previa del animal, resulten en que un alto porcentaje de estos permanezcan confinados en el PV, entonces el grado de innovación sería muy alto, y el impacto incalculable. El alcance sería aún mayor si, por ejemplo, se pudiera ir modificando el perímetro virtual, y de esa manera ir lentamente trasladando al ganado. De esta manera, se podría mover al ganado con una computadora.

Sin embargo, esto es solo una posible línea de continuación, no se pretende llegar tan lejos. En la sección 1.3 se detalla con precisión los resultados que se espera obtener, dejando el camino abierto a la realización de aplicaciones más comerciales de los mismos.

1.2. Antecedentes

El diseño de perímetros virtuales ha sido estudiado en varias partes del mundo [3–8], utilizando diversas técnicas para mantener confinado al ganado dentro de un determinado perímetro, configurado de forma remota. Estas técnicas comparten un patrón común que consiste en colocar un collar en el animal, con un dispositivo capaz de aplicar estímulos sonoros en caso de que el mismo se acerque a los límites del perímetro, seguido de un segundo estímulo, que consiste en una descarga eléctrica del orden de los kilovoltios (este último en caso que atravesase los límites preestablecidos [5]). De esta forma se intenta que el animal, a corto plazo, desarrolle la capacidad de evitar la descarga eléctrica y se aleje de los límites del perímetro en cuanto sienta el primer estímulo, pudiéndose también utilizar estímulos lumínicos o táctiles, según otras experiencias internacionales.

1.2.1. Evolución de los métodos de confinamiento de ganado

En los documentos estudiados [3–8] se exponen algunos de los principales aportes y características del perímetro virtual. Allí se define al mismo como un método que permite implementar un recinto (para poder confinar animales por ejemplo) sin la necesidad de contar con barreras físicas.

La necesidad de confinar el ganado se hizo presente en el mundo por diferentes razones. En Escocia [6], el lento pero constante crecimiento de la población en épocas preindustriales provocó una mayor demanda de granos y se arrasó con las zonas de pastoreo. Por esta razón fue necesario separar a los animales de las zonas de cultivo ya que el escaso crecimiento de hierba dificultaba la adecuada alimentación del ganado. En un principio se utilizaba mano de obra infantil para alejar al ganado de las zonas de cultivo, ya que era la forma barata de hacerlo. Años mas tarde, con la imposición de la educación obligatoria se sustituye la mano de obra infantil por millas de alambrados.

Dada la gran utilidad de los alambrados, a través de los años se han logrado avances con el fin de hacer mas eficientes los métodos de confinamiento, apuntando en los últimos tiempos a métodos mas sofisticados como los Perímetros virtuales.

La principal ventaja de los perímetros virtuales frente a los alambrados convencionales es su versatilidad y practicidad. La modificación y mantenimiento del perímetro tradicional tiene grandes costos asociados debido a la mano de obra y materiales necesarios para los mismos, además del tiempo requerido para dichas actividades, mientras que estas tareas son relativamente sencillas en los perímetros virtuales implicando solamente una reconfiguración del mismo.

Capítulo 1. Generalidades

La tabla 1.1 expone las principales ventajas y desventajas de los distintos métodos de confinamiento.

Tipo de alambrado	Ventajas	Desventajas
Convencional	Gran efectividad en confinamiento	Instalación costosa
		Mantenimiento costoso
		Puede dañar al ganado
Electrificado	Gran efectividad en confinamiento	Mantenimiento costoso
		Necesidad de toma de tierra
		Existe riesgo de muerte para los animales
Perímetro Virtual	Puede ser muy flexible.	Menor grado de efectividad en confinamiento
	Potencial reducción de costos a futuro	No es visible, lo que imposibilita su ubicación exacta
	Se pueden implementar en sitios de difícil acceso para las personas	No retiene la vida silvestre
	Puede ser usado en conjunto con otros sensores para el monitoreo	
	posibilidad de reunir animales de forma remota	
	Puede ser usado en conjunto con otros sensores para monitorear el crecimiento del pasto	

Tabla 1.1: Ventajas y desventajas de los distintos métodos de confinamiento de ganado basado en [6].

Los primeros antecedentes de perímetros virtuales fueron sistemas de confinamiento de mascotas utilizando un cable que delimitaba un perímetro que se deseaba que el animal no abandonara. Al acercarse al cable un dispositivo que llevaba el animal emitía un determinado sonido o eventualmente un choque eléctrico [6]. Esta idea se llevó más tarde al ganado, probando en primera instancia con cabras y teniendo resultados positivos validando así al funcionamiento del método también en ganado [6].

Con el paso del tiempo se desarrollaron varios métodos de perímetro virtual que consisten en enviar un estímulo neutro (sonoro, táctil, luminoso, etc) si este se acerca a los límites del perímetro, y un segundo estímulo invasivo que consistente en

una descarga eléctrica del orden del kilovoltio en caso de que el animal no reaccione al primero. Estos métodos apuntan al adiestramiento del animal, logrando que se detenga e incluso retroceda una vez que reciba el primer estímulo para evitar la descarga eléctrica (por ejemplo [5]).

El problema de estos métodos es que en algunas partes del mundo no está permitido trabajar con descargas eléctricas en ganado con el fin de respetar las normas referentes al bienestar animal.

1.2.2. Perímetros virtuales y adiestramiento

Si bien el confinamiento remoto de ganado es un área que aún no ha sido comercialmente explotada debido a su corta edad, sí se han realizado varios trabajos de investigación en los que se estudian distintos métodos para llevar a cabo dicha funcionalidad.

La información recabada [3–8] muestra que la mayoría de perímetros virtuales se basa en la adiestración del animal a través de la aplicación de un estímulo neutro seguido de un estímulo negativo (en general, una descarga eléctrica). Esto implica que la eficiencia del PV depende tanto del correcto funcionamiento del dispositivo como del adiestramiento del animal, como puede observarse en el estudio realizado por la empresa Agersens [5] en conjunto con la organización CSIRO (organización encargada de la investigación científica de Australia), sobre la reacción del ganado a perímetros virtuales móviles.

En dicho estudio se comparan los hábitos del animal antes y luego de realizadas las pruebas para saber como afecta el experimento al bienestar del animal.

Las pruebas se realizaron con un collar que tiene la funcionalidad de perímetro virtual basada en un sistema con GPS. Cuando el animal se acerca a los límites del perímetro el dispositivo emite un sonido de 3s de duración, si el animal se detiene o retrocede el estímulo cesa, en caso contrario se le aplica una descarga de 800 V. El experimento consiste en varias etapas que se detallan a continuación:

- Se dejan 11 animales de estudio (Angus de aproximadamente 20 meses de edad) 5 días con acceso al 100 % del predio y el collar apagado para que se adapten.
- Al quinto día se enciende el collar y se limita el acceso a un 40 % del predio.
- El día 14 se aumenta la superficie a un 60 % de la superficie del predio.
- El día 17 se aumenta la superficie a un 80 % de la superficie del predio.
- El día 20 se da acceso al 100 % del predio nuevamente.
- El experimento concluye el día 22.

Los resultados que obtuvieron muestran el adiestramiento de los animales, ya que en aproximadamente 48 hs de iniciado el experimento la mayoría del ganado

Capítulo 1. Generalidades

había aprendido a reaccionar a los estímulos minimizando la cantidad de choques eléctricos. Se deseaba ver si los animales reaccionaban al PV debido al aprendizaje mediante los estímulos, o por el contrario memorizaban las zonas límite del perímetro y luego las evitaban. Al mover los límites del PV (ampliando la superficie) se notó que los animales ocupaban una mayor superficie, concluyendo que estos respondían de forma adecuada al PV.

Esto se puede observar claramente en la siguiente figura tomada del documento [5]:

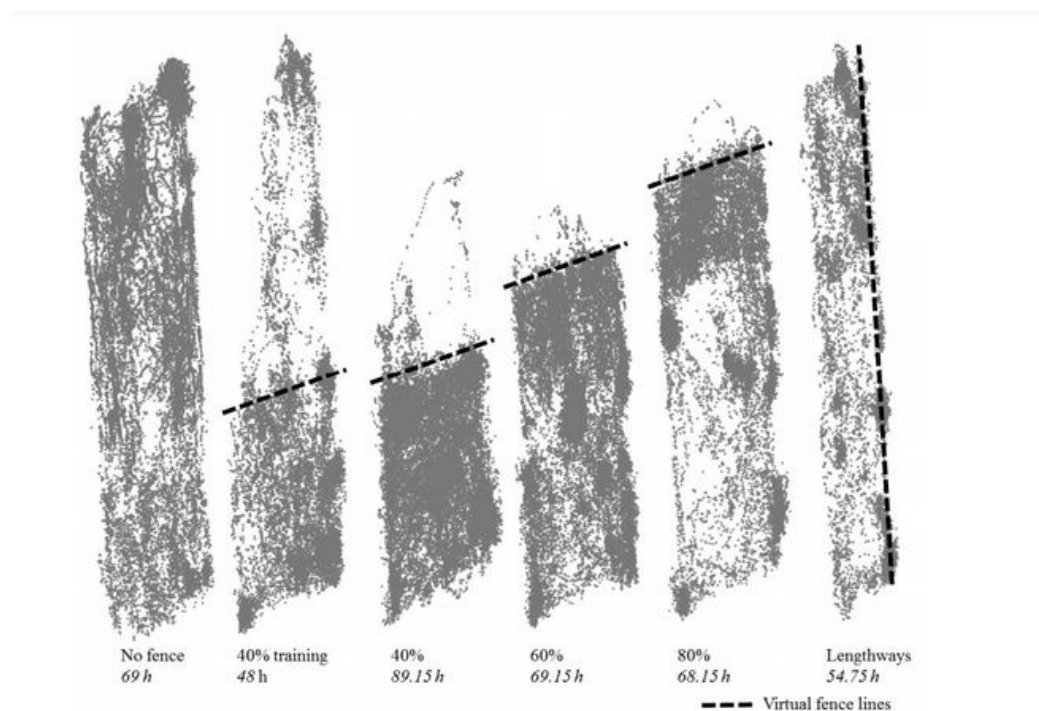


Figura 1.1: Datos obtenidos del GPS del dispositivo de 6 de los animales durante el transcurso del experimento [5].

Durante el transcurso del experimento los animales no presentaron modificaciones en su comportamiento habitual. Esto ultimo en conjunción con lo expuesto anteriormente llevo a concluir que un sistema de perímetro virtual enfocado en el adiestramiento del animal (estímulo neutro + estímulo castigo) es adecuado para la gestión remota de pastoreo del ganado. Cabe destacar que en este proyecto de fin de carrera se diseñó un dispositivo capaz de implementar correctamente la funcionalidad de perímetro virtual, sin ocuparse del adiestramiento. Así mismo se realizaron pruebas en campo con el fin de ver la reacción de los animales a los estímulos emitidos.

1.3. Problema a resolver

El problema que se intenta abordar en este proyecto de fin de carrera se puede dividir en las siguientes etapas:

1. Ubicar al animal (geolocalización).
2. Buscar estímulos efectivos para su confinamiento en el perímetro virtual.
3. Fijado un perímetro virtual, estimular al animal si se acerca al mismo.
4. Verificar que estos estímulos no afectan su bienestar y la producción. Comprobar que el animal quede confinado al perímetro virtual. En caso contrario, volver a (2).

El problema debe resolverse iterando. Además, deberá ser abordado por un equipo multidisciplinario. En la sección 1.4 se define qué partes del problema general serán el enfoque de este proyecto de fin de carrera.

Objetivos del proyecto

1. Desarrollar un sistema de localización de ganado, que disponga de comunicación inalámbrica de largo alcance y una interfaz gráfica.
2. Generar una primer experiencia dentro del Departamento de Electrónica del IIE en confinamiento virtual de animales, compatible con el bienestar animal.

Los objetivos deberán alcanzarse cumpliendo con los requerimientos que se detallan a continuación

Requerimientos del proyecto:

- Se utilizará tecnología LoRa para la transmisión de los datos relevantes, y LoRaWAN como protocolo de red.
- Se utilizarán estímulos no invasivos para el confinamiento de la vaca al perímetro virtual. Se priorizará el bienestar animal y la biocompatibilidad de los componentes.
- Las principales variables de interés del perímetro virtual deberán ser configurables de manera remota. Es decir, la comunicación entre el dispositivo y el servidor central deberá ser bidireccional (se expandirá sobre este punto en el resto del capítulo)
- Se deberá contar con al menos 2 estímulos, uno sonoro y uno táctil, configurables de manera remota.
- Lograr un diseño final de un dispositivo con una superficie menor que $225cm^2$.

1.4. Alcance y criterios de éxito

Alcance del proyecto:

- Se realizará un dispositivo para un animal. No se considera parte del proyecto generar más dispositivos.
- El problema a abordar es el perímetro virtual en toda su complejidad. Sin embargo, nosotros nos restringiremos a desarrollar un primer prototipo, que tendrá sus parámetros de interés configurables.
- No formará parte del alcance determinar si los generadores de estímulos elegidos pueden cumplir con la función de confinamiento virtual. Esta tarea le corresponde a un equipo multidisciplinario, que deberá incorporar investigadores del área de comportamiento animal.
- Se tercerizará la interfaz gráfica a partir de la cual se configurarán los parámetros de interés del perímetro virtual. (se expandirá sobre este punto en la sección 1.6)

A continuación se detallan los criterios de éxito ideales del proyecto. En el mejor escenario, se espera concluir el proyecto cumpliendo con cada uno de los siguientes criterios:

Criterios de éxito ideales	
Precisión en la posición	5 metros
Período de muestreo	5 s
Precio por animal	US\$ 30
Alcance	10 km
Autonomía	5 años
Funcionalidad del dispositivo	Deberá estimular solo en los casos que corresponda.

La tabla a continuación detalla los criterios de éxito aceptables del proyecto. Es decir, es lo mínimo que se espera tener al concluir el proyecto.

Criterios de éxito aceptables	
Precisión en la posición	10 metros
Frecuencia de muestreo	50 mHz
Precio por animal	US\$ 50
Alcance	5 km
Autonomía	7 días
Funcionalidad del dispositivo	Deberá estimular solo en los casos que corresponda.

1.5. Solución propuesta

Debido a la información expuesta anteriormente respecto a los perímetros virtuales se diseñó un dispositivo con la funcionalidad de PV basado en un sistema con GPS. El mismo cuenta con módulos de hardware encargados de la comunicación inalámbrica y un sistema de estímulos. Dado que es de especial importancia para este proyecto el bienestar animal, los dos estímulos que se utilizaron son pasivos. En primera instancia se aplica un estímulo sonoro, seguido de un estímulo táctil (vibración) en caso que el animal intente atravesar el perímetro.

Se propone diseñar el prototipo de un dispositivo para colocar en un animal. Dicho prototipo debe contar con un módulo GPS, un microcontrolador, un módulo transmisor/receptor (que utilice LoRa), un módulo de estimulación y un sistema de alimentación que haga uso de energía solar. A dicho dispositivo le llamaremos **Nodo Sensor (NS)**.

El microcontrolador del NS será el encargado de recibir la posición del GPS, procesarla (para el perímetro virtual) y en caso de que sea necesario enviar las señales de control que activarán las estimulaciones.

En la figura 1.2 podemos ver el diagrama de bloques del NS propuesto. Las elecciones de Hardware que se ven en la figura 1.2 se desarrollan en el capítulo 2. El NS propuesto en la figura 1.2 tiene el hardware necesario para cumplir con los requerimientos presentados en la sección 1.3, y los criterios de éxito presentados en la sección 1.4.

En este proyecto se diseña un NS. Sin embargo, la arquitectura de la solución contempla que el sistema se utilice en muchos animales. En ese caso (ver figura 1.3), se tiene una red de Nodos Sensores (NS) que se comunican con un **Gateway (GW)** enviándole su posición periódicamente. El Gateway será el encargado de recibir los datos de la red de NS, y subirlos a una plataforma online vía Internet (dicha plataforma será Everynet Network Server, que es una plataforma IoT para manejo remoto de dispositivos, ver figura 1.3). Everynet Network Server será consultada por un servicio, que correrá en una máquina virtual que abriremos en Amazon Web Services. Dicha máquina virtual tendrá también una base de datos, dónde almacenará los datos de la red de sensores. También deberá correr en dicha máquina un servicio web. Dicho servicio web permite visualizar el historial de las posiciones enviadas por los NS, y también permite configurar los principales parámetros de interés de dichos NS (se desarrolla en la sección 3.4.2).

El Gateway se comprará teniendo en cuenta las necesidades del proyecto, no se diseñará.

Capítulo 1. Generalidades

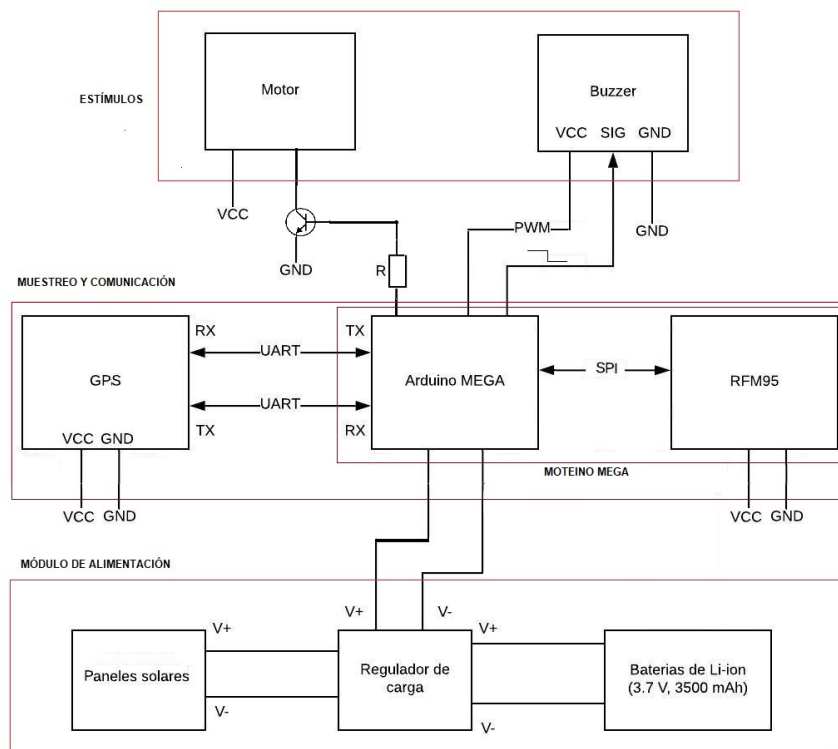


Figura 1.2: Diagrama de bloques del NS propuesto.

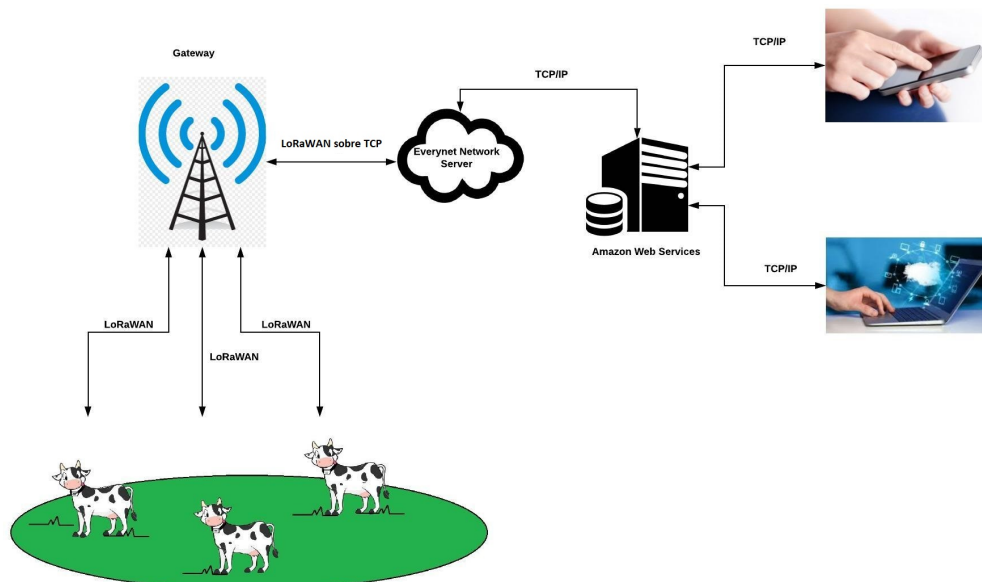


Figura 1.3: Arquitectura del sistema propuesto

1.6. Apoyos recibidos y colaboraciones

En lo que refiere al funcionamiento del dispositivo, será distinto según en qué zona se encuentre el animal. La figura 3.9 resume el comportamiento esperado por zonas.

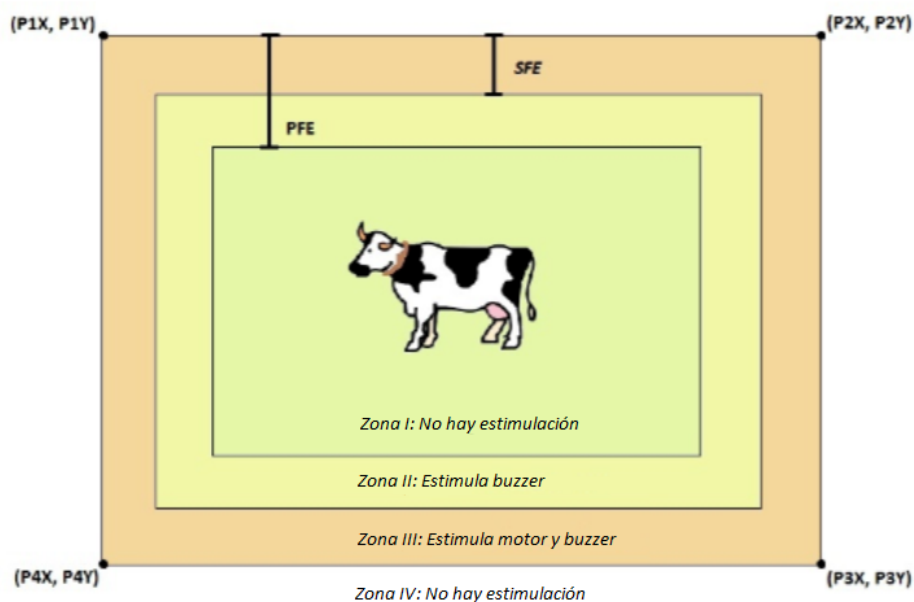


Figura 1.4: Funcionamiento esperado del dispositivo

La zona I es aquella en la que se quiere confinar al animal. En dicha zona no habrá estimulaciones. La zona II está determinada por las distancias PFE (primera franja de estimulación) y SFE (segunda franja de estimulación). En esta zona se estimulará al animal con el buzzer, con el propósito de que vuelva a la zona I. La zona III está determinada por el Perímetro virtual (P1,P2,P3 y P4) y el ancho SFE. En esta zona, se estimula con el motor y con el buzzer, esperando que el animal vuelva a la zona I. La zona IV está determinada por el perímetro virtual. Si el animal llega a la zona IV, en este proyecto se decide desactivar las estimulaciones, dado que se considera que el animal ya no va a entrar al PV.

En la sección 3.4.1 se desarrolla el funcionamiento esperado por zonas en profundidad.

1.6. Apoyos recibidos y colaboraciones

La financiación de este proyecto se obtuvo a través del programa FFI: Proyectos de Innovación de Fin de Carrera, y a través de una beca obtenida en “Circuits and Systems Society” de la IEEE (IEEE CAS). También se contó con el apoyo de Facultad de Veterinaria, el grupo HCL del InCo y la empresa de telecomunicaciones Teliot. A continuación, se detallan sus principales aportes:

1.6.1. Programa FFI: Proyectos de Innovación de Fin de Carrera

El grupo se postuló al programa “Fondo Fomento Ingeniería (FFI): Proyectos de Innovación de Fin de Carrera” que ofrece el Centro de Innovación en Ingeniería (de ahora en adelante CII). El fin de este fondo es apoyar el desarrollo de los proyectos de fin de carrera que cuenten con una demanda previa del sector productivo.

El proceso requirió conseguir una empresa sponsor, que debía aportar el 20 % del dinero otorgado por el CII (que a su vez podía ser de hasta \$200.000 por proyecto). Pablo Tavares, productor rural del departamento de Durazno, se ofreció a colaborar, quedando en calidad de sponsor. Para participar en dicho proceso se elaboraron varios documentos, entre los cuales se incluía información detallada del proyecto, un cronograma financiero, la justificación de las elecciones de Hardware y el aval de la empresa.

Como resultado de la convocatoria, este proyecto fue seleccionado como uno de los ganadores, accediendo a un total de \$240.000 para la compra de insumos y servicios. En el proyecto de fin de carrera se plantea realizar un NS, mientras que para el proyecto ANNI se fabricarán 10 NS, qué luego se probarán en animales.

1.6.2. Colaboración con Grupo HCL del INCO

El diseño de la prueba de concepto del backend del sistema, la interfaz gráfica de usuario y su puesta en marcha en Amazon Web Services (de ahora en adelante AWS) se realizó mediante la colaboración del grupo HCL [9] del Instituto de Computación de Facultad de Ingeniería (de ahora en adelante InCo). Se conformó un módulo de taller, el cual fue realizado por Raúl Marichal, bajo la dirección del Ing. Martín Pedemonte. El mismo fue financiado utilizando parte de los fondos obtenidos en el programa FFI. En la sección 3.7 se desarrollan los requerimientos solicitados para realizar este proyecto.

1.6.3. Colaboración de Facultad de Veterinaria

Se contactó a un grupo de estudiantes avanzados de la Facultad de Veterinaria (en adelante FVET) interesados en el proyecto, quienes junto con la Profesora Adjunta Soledad Valledor, brindaron la orientación necesaria para realizar las pruebas de campo iniciales. Se realizaron dos pruebas de concepto, con los primeros prototipos de collar, en el departamento de Florida, Casupá. Esto fue posible gracias a la colaboración de la estudiante de FVET Dayanara Bonilla, quien permitió utilizar el campo de su padre y algunos ejemplares bovinos. El manejo de los animales y la colocación del collar estuvo a cargo de los estudiantes (Dayanara Bonilla y Cristian Perdomo) y del trabajador rural Héctor Matos.

La última prueba de campo realizada para testear el prototipo final del collar, tuvo lugar en la Facultad de Veterinaria. Se coordinó esta visita con la Doctora en Medicina y Tecnología Veterinaria Stella Maris Huertas Canén, integrante del Centro Colaborador de la OIE (“World organization for animal health”) sobre

1.6. Apoyos recibidos y colaboraciones

Bienestar Animal de FVET, quien brindó el apoyo necesario para llevar a cabo las pruebas realizadas.

1.6.4. Colaboración de Teliot

La empresa Teliot brinda servicios en redes de comunicación, sensores y big data. Se contactó a uno de sus integrantes, Germán Capdehourat, Profesor Adjunto de la Facultad de Ingeniería, quien junto con el tutor de este proyecto, Julián Oreggioni, realizaron un convenio con el IIE que permitió la instalación de un Everynet Network Gateway en el instituto. También ofrecieron el servicio de la plataforma Everynet a través de la cual se comunica el nodo sensor descrito anteriormente (Sección 1.5) y brindaron ayuda con problemas de conectividad y desarrollo de firmware.

Cabe destacar la colaboración de Francisco Cabrera, quien brindó apoyo en la gestión de usuarios para el proyecto en la plataforma EveryNet y colaboró en el desarrollo de software necesario para adaptar la librería LMiC para comunicación bidireccional. También ayudó en la configuración de la LMiC para que el dispositivo transmita en la sub-banda en la que escucha el Everynet Network Gateway antes mencionado (se desarrolla en profundidad en la sección 3.2).

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 2

Hardware

En este capítulo se abordarán las elecciones de hardware para el Nodo Sensor y el Gateway. En el caso del Nodo Sensor, se traducen los módulos descritos en la sección 1.5 en componentes concretos existentes en el mercado. En lo que al Gateway respecta, se trabajó con el modelo Everynet Network Gateway brindado por la empresa Teliot (ver sección 1.6.4)

2.1. Diagrama de bloques del hardware utilizado

Cada NS tendrá la arquitectura que se muestra en la figura 2.1:

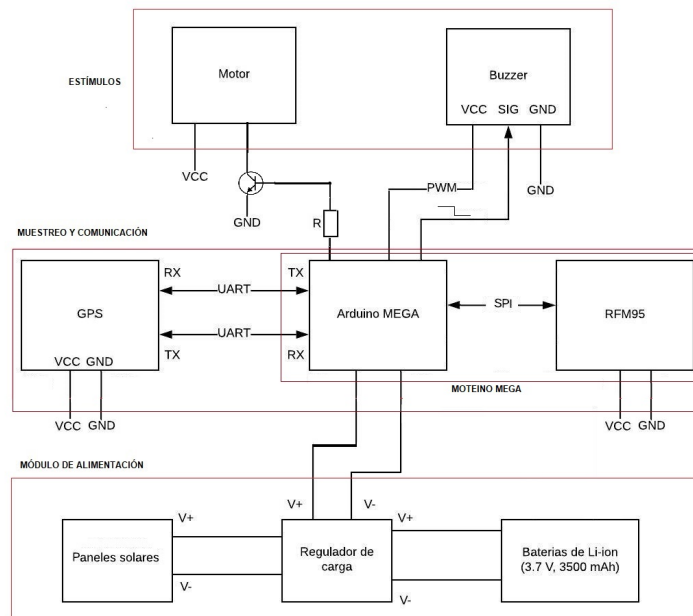


Figura 2.1: Arquitectura HW propuesta para el NS.

Capítulo 2. Hardware

Se requiere que el NS sea capaz de ubicar a la vaca dentro del campo (GPS) y también se requiere que sea capaz de comunicarse con un servidor central, a través de un Gateway que instalaremos en el campo (cada campo tendrá tantos nodos sensores como animales, y al menos un Gateway). La red de NS se comunicará con el servidor central a través del GW, enviándole la posición del animal periódicamente. A su vez, el NS deberá en caso de detectar que el animal salió del perímetro virtual preestablecido, estimularlo para que vuelva a su potrero.

2.2. MoteinoMEGA

Este dispositivo es una plataforma de desarrollo inalámbrica de baja potencia que incluye un Arduino Mega (microcontrolador ATmega1284P) [10] y un módulo Hardware que soporta el protocolo de comunicación LoRa.

Debido a la recomendación de la empresa Teliot, se optó por utilizar Moteino en los NS. Existen dos series de módulos de comunicación con los que se puede trabajar en Moteino: la serie RFM9X [11] y la serie RFM69 [12]. Ambas series son similares difiriendo en algunas características como se muestra en la siguiente tabla.

Serie	Corriente promedio (mA)	Sensibilidad (dBm)	P_out (dBm)
RFM9X	10.3	-148	14
RFM69	16	-120	20

Tabla 2.1: RFM9X v RFM69.

Se puede observar que los módulos RFM9X consumen aproximadamente un 60 % menos de corriente promedio que los RFM69 y además cuentan con una mayor sensibilidad y menor consumo en potencia de transmisión.

Por estos motivos se decidió utilizar un modulo de comunicación de la serie RFM9x, el RFM95.

2.3. GPS

La amplia aceptación en el mercado de los GPS de la empresa U-blox, disponibilidad de software libre para su configuración y gran cantidad de información sobre estos módulos, fueron las principales razones por las cuales se decidió trabajar con uno de los GPS fabricados por la misma. Se realizó un estudio de los distintos GPS que ofrece U-blox y se encontraron 3 modelos compatibles en términos de tensión con el microcontrolador que se utilizará para el NS. Estos modelos son: NEO 6M [13], NEO 7M [14] y NEO M8N [15]. La tabla 2.2 a continuación muestra las características mas relevantes de dichos GPS:

	NEO 6M	NEO 7M	NEO M8N
Canales	50	56	23
Sensibilidad (dBm)	-161	-161	-164
Max Frec Muestreo (Hz)	5	10	10
Precisión (m)	2.5	2.5	2.5
Alimentación (V)	2.7-3.6	1.65-3.6	2.7-3.6
Corriente max (mA)	67	67	67
Corriente promedio (mA)	47	32	25

Tabla 2.2: Características de los GPS NEO: 6M, 7M y 8N.

Se decidió utilizar el GPS NEO M8N para los NS debido al menor consumo en corriente promedio y mayor frecuencia de muestreo y sensibilidad.

2.4. Buzzer

Para el estímulo sonoro del NS se decidió trabajar con un buzzer (transductor electroacústico que produce un sonido de un mismo tono [16]). Existen dos tipos de estos componentes, activos y pasivos.

Un buzzer pasivo permite convertir una señal eléctrica en una onda de sonido. De esta forma, se pueden generar tonos de distintas frecuencias. En oposición, un buzzer activo dispone de un oscilador interno, por lo que únicamente se debe alimentar el dispositivo para que se produzca el sonido. Sin embargo, el mismo tiene una frecuencia fija de funcionamiento, que viene determinada por el fabricante.

Se optó por utilizar un módulo buzzer pasivo [17] para poder variar las frecuencias de sonido. Este módulo cuenta con tres pines: VCC, SIG y GND. El pin SIG es la entrada de la señal eléctrica a convertir en sonido. Se utilizó la función `tone(pin, frequency, duration)` [18] de Arduino para generar una onda cuadrada en uno de los pines del Moteino. Mediante esta función se puede variar la frecuencia y duración del sonido emitido.

Para controlar el volumen, se varía la tensión de alimentación. Se trabajó con una señal PWM generada por uno de los pines del Moteino, conectado directamente al pin VCC del buzzer como se puede ver en la figura 2.1. Modificando el ciclo de trabajo del mismo se puede variar el voltaje de alimentación del buzzer de 0 a 3.3 V. Por último, se colocó un capacitor de 1000 uF entre dicho pin y GND para lograr una alimentación más estable en la entrada VCC, lo que derivó en una mejora de la calidad del sonido emitido.

El módulo buzzer seleccionado tiene unas dimensiones de 33x13.2x10 mm y opera con frecuencias de hasta 10 kHz.



Figura 2.2: Buzzer elegido

2.5. Motor

Para el estímulo táctil del NS, se utiliza un motor vibrador, de las siguientes características:

- Modelo: R260.
- Tensión nominal: 3-6 V.
- RPM: 3000 a 6 V.
- Corriente: 3 V 210 mA; 4.5 V 400 mA; 6 V 510 mA.
- Longitud del eje: 6 mm, diámetro del eje: 2 mm.
- Longitud del cuerpo: 27 mm, diámetro del cuerpo: 24 mm.
- Peso: 30 g.



Figura 2.3: Motor elegido. Imagen tomada de [19]

2.6. Paneles solares

Se decidió incorporar al NS paneles solares, de manera de cumplir con el criterio de éxito ideal en cuanto a la autonomía del dispositivo (ver sección 1.4). Para esto

2.6. Paneles solares

se dimensionaron los paneles que se utilizaron realizando un estudio del consumo energético del dispositivo y calculando la energía (mínima) que deberá ser capaz de recolectar el panel diariamente.

2.6.1. Corriente promedio consumida

Haciendo uso de los datos proporcionados por las correspondientes datasheets y realizando estimaciones del ciclo de trabajo de los distintos componentes del dispositivo (buzzer, motor, Moteino, GPS y RFM95), se logro calcular de forma aproximada la corriente promedio consumida. En la tabla 2.3 se muestra la información antes mencionada para realizar los cálculos:

Componente	Corriente instantánea (mA)	Ciclo de trabajo (%)	Corriente promedio (mA)
Moteino	8	100	8
RFM95 STby	2	100	2
RFM95 TX	120	1.6	1.92
RFM95 RX	100	0.16	0.016
GPS	27	100	27
Motor	600	0.6	3.6
Buzzer	30	2.2	0.77

Tabla 2.3: Consumo y ciclo de trabajo.

De la tabla se puede estimar un consumo en corriente promedio de 43.3 mA.

2.6.2. Dimensionado de los paneles

La energía consumida a diario se obtiene multiplicando la corriente promedio por el voltaje nominal de las baterías usadas para la alimentación (3.7 V) y por las 24 horas que contiene un día, lo que da como resultado 3.85 Wh.

Dado que los fabricantes de paneles solares brindan la información de potencia nominal del mismo, correspondiente a la potencia máxima que es capaz de entregar el panel durante las horas de sol pico (HSP), fue necesario saber cuál es el comportamiento del indicador HSP correspondiente a Uruguay durante el año.

No fue posible encontrar la distribución anual de HSP en Uruguay, por lo que se decidió utilizar la información del HSP de la ciudad de Baltimore ya que esta presenta cierta simetría con Uruguay respecto del ecuador (como se muestra en la figura 2.4) , lo que implica una similar distribución de las horas de sol.

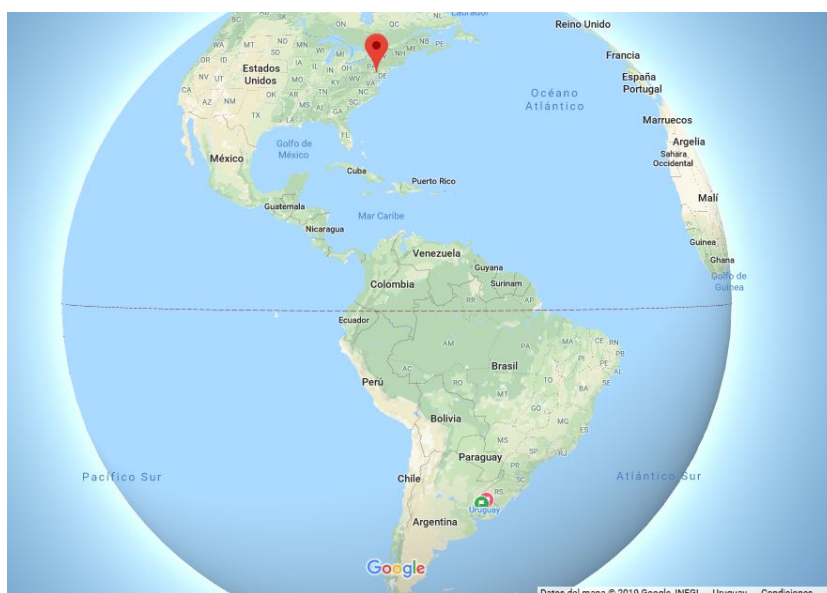


Figura 2.4: Posición relativa entre Baltimore y Uruguay. Imagen tomada de [20].

A continuación, se muestra la distribución de HSP en Baltimore durante el año obtenida de [21] y la potencia nominal mínima con la que deberá disponer el sistema de paneles a incorporar:

Estacion	Verano	Otoño	Primavera	Invierno
HSP	5.5	4.5	4.5	3

Tabla 2.4: Indicador de HSP. Tomada de [21]

La tabla 2.4 muestra que el HSP mínimo en el año es de 3 por lo que será necesario contar con un sistema de paneles solares con una potencia nominal de al menos 1.28 W ($3,85Wh/3h$).

Se decide utilizar dos paneles flexibles (de 0.5W de potencia y 1.5 V de salida nominales) para los laterales del collar y un panel rígido (0.5W, 4.5 V) en la parte superior del encapsulado del dispositivo. Utilizándolos en conjunto se llega a 1.5W, suficiente como para cubrir las necesidades diarias del sistema.

2.7. Baterías

Con el fin de poder realizar pruebas en campo durante un fin de semana sin la necesidad de quitar el collar a los animales, la capacidad de la batería se determinó para que pueda alimentar al sistema al menos 3 días sin recibir energía solar. En la sección 2.6 se estimó que el dispositivo consume en promedio 43.3 mA (3.85 Wh). Si multiplicamos este consumo de corriente por la cantidad de horas en 3 días (24x3), obtenemos la capacidad mínima de la batería, en mAh, que se requiere

2.8. Módulo de carga TP4056

en este caso: 3125 mAh. Sin embargo, debemos considerar un margen para la carga no útil que queda en la batería en el momento que el equipo deja de operar, debido a que la tensión no es suficiente para operar. Si consideramos para ello un margen del 20 %, obtenemos que la batería deberá tener una capacidad nominal mínima de 3750 mAh.

Se eligió una batería recargable de iones de Litio (Li-Ion), modelo 18650 (en alusión a sus dimensiones cilíndricas: 18 mm de diámetro y 65 mm de altura), de 3.7 V y 4000 mAh de capacidad nominal [22]. Sin embargo, la energía que puede entregar la misma sin bajar de los 3.7 V según el fabricante es de 9.6 Wh, con lo cual podemos asegurar que el dispositivo va a estar operativo 2.5 días sin recibir luz solar a al menos 3.7 V. Consideramos que estamos cumpliendo con el requerimiento de 3 días, dado que habrá luz solar y el dispositivo funciona correctamente a menor voltaje que 3.7 V.

2.8. Módulo de carga TP4056

Para cargar de forma adecuada la batería seleccionada, se utilizó un módulo de carga TP4056 [23]. Este módulo tiene una entrada donde se conectan los paneles solares, una salida para alimentar nuestro circuito y otra para cargar las baterías. El módulo TP4056 tiene un sistema de protección que corta el circuito cuando el nivel de voltaje cae por debajo a 3.2 V y evita las sobrecargas, deteniendo la carga al llegar a 4.2 V. A continuación se pueden ver sus especificaciones:

- Módulo de carga lineal.
- Corriente máxima: 500 mA.
- Precisión de carga: 1.5 %.
- Voltaje de Entrada: 4.5 V - 5.5 V.
- Voltaje de carga full: 4.2 V.
- Led indicador: Amarillo (Cargando), Azul (Carga completa).
- Tamaño: 25 x 19 x 10 mm.
- Peso: 30 g.

Dado que el voltaje de entrada de este módulo no puede superar los 5.5 V, se utilizó un regulador de voltaje de 5V (LM78M05C) para poder usar los paneles seleccionados en serie (salida: 7.5 V a 9 V en HSP).

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 3

Software

En este capítulo se abordará el desarrollo de software que fue necesario para el proyecto.

Dado que el microcontrolador con el que se va a trabajar es el ATmega1284P, se utilizará el lenguaje de programación de Arduino. Dicho lenguaje es una adaptación de C (la sintaxis, estructura, operadores y funciones son las mismas que las del lenguaje C). Sin embargo, se agregan funciones específicas de Arduino, tales como `pinMode()`, `digitalWrite()` y `delay()` [24]. Se eligió Arduino IDE como entorno de desarrollo, dado que el Moteino está pensado para ser programado desde Arduino IDE. Para poder programar el Moteino es necesario instalar el “Moteino Core”, que define el mapeo de pines para que el Moteino pueda ser programado desde Arduino IDE.

Durante el transcurso de este capítulo se hablará de las tecnologías utilizadas para este proyecto. Hablaremos de LoRa (Long Range), LoRaWAN (Long Range Wide Area Network) y de la librería LMiC (IBM LoRaWAN in C). La librería LMiC, adaptada para Moteino, es la que se utilizó para transmitir los datos través del RFM95 conforme a las especificaciones de LoRaWAN. También se desarrollarán las funciones de la librería TinyGPS++ utilizadas para el manejo del GPS. Se abordará la arquitectura de Software utilizada, destacando cómo la información llega desde el dispositivo al usuario final. Se desarrollan los protocolos diseñados para configurar el dispositivo, junto con los parámetros de interés elegidos para definir el perímetro virtual. Se muestra el diagrama de estados implementado para cumplir con el funcionamiento esperado del dispositivo. Finalmente, se explicitan los requerimientos de la interfaz web.

3.1. LoRa - LoRaWAN

LoRa (Long Range) es una tecnología digital de comunicación de datos inalámbrica que fue desarrollada por Cycleo de Grenoble, y adquirida por Semtech en 2012. Se trata de un protocolo de comunicación inalámbrica de largo alcance (más de 10 km en áreas rurales) con bajo consumo de energía cuya velocidad de transmisión de datos está por debajo de 50 kbps.

Capítulo 3. Software

Esta tecnología utiliza una técnica de modulación de espectro expandido derivada de la tecnología de espectro de dispersión de chirrido (Chirp Spread Pectrum - CSS).

Los circuitos integrados que transmiten usando LoRa se comercializan según la región de acuerdo con la reglamentación específica de cada país. Los organismos regulatorios de cada país definen qué frecuencias se pueden usar de manera libre. Por ejemplo, en Uruguay la banda que se encuentra libre para transmisión y recepción de datos utilizando LoRa es la AU915 (915-928MHz).

Mencionaremos a continuación algunas de las principales características de LoRa [25]:

- Usa bandas ISM. Estas son bandas de radio reservadas para radiofrecuencia con fines industriales, científicos y médicos.
- Utiliza una arquitectura de red en estrella. El NS se comunica con un Gateway, y éste es el responsable de enviar a un servidor la información. Es posible implementar arquitecturas de red en malla, pero para eso es necesario apoyarse en protocolos que lo implementen en capa 2 y superior (se explica los niveles OSI en esta misma sección más adelante).
- El bitrate es configurable, pudiendo alcanzar un máximo de aproximadamente 27kbps.
- Bajo costo (Entre 5 y 7U\$S por un módulo de radio RFM95) [26].
- Los circuitos integrados son fabricados únicamente por Semtech.

Para poder utilizar el protocolo LoRa es necesario configurar determinados parámetros de conexión en los dispositivos. Dichos parámetros son:

1. **Canal dentro de la banda:** frecuencia central que representa el canal. A modo de ejemplo, el canal 10 dentro de la banda 868 MHz tiene un valor de 865.200.000 Hz. A su vez, se agrupan los canales en sub-bandas (cada sub-banda agrupa 8 canales). Los nodos deben transmitir en distintos canales dentro de la misma sub-banda para sucesivas transmisiones.
2. **“Spreading factor” (SF):** define el número de bits usados para codificar un símbolo. A mayor SF, menor velocidad de transferencia, pero mayor inmunidad a interferencias.
3. **“Coding rate” (CR):** indica la forma de codificar para corrección de errores. Según la técnica especificada, añade símbolos de control para verificar si los datos son correctos o no e incluso poder determinar los valores correctos.
4. **Bandwidth (BW):** indica el ancho de banda a utilizar.

3.1. LoRa - LoRaWAN

LoRa es un protocolo de capa física, define el protocolo de comunicación inalámbrica. LoRaWAN es una especificación de redes (capa de enlace y capa de red) definido por LoRa Alliance que se encarga de comunicar diferentes dispositivos LoRa gestionando sus canales y parámetros de conexión. Es lo que se conoce como MAC (Media Access Control, es decir control de acceso al medio).

Atendiendo al modelo de 5 capas (ver figura 3.1), LoRaWAN se encuentra en los niveles 2 y 3, mientras que en el nivel 1 (capa física) se encuentra la tecnología LoRa.

5	Aplicación
4	Transporte
3	Red
2	Enlace
1	Física

Figura 3.1: Modelo de 5 capas que define las diferentes fases por las que deben pasar los datos para viajar de un dispositivo a otro sobre una red de comunicaciones (ver [27]).

Para conectar un dispositivo LoRa a la red se utilizó activación ABP (“Activation by personalisation”). Esta activación implica primero crear un dispositivo en el servidor de EveryNet, y generar para dicho dispositivo un identificador (“Device address”) y contraseñas para autenticación (“Network session key” y “Application session key”). Estos valores se fijan en el programa que corre en el Moteino, para que este pueda identificarse dentro de la red. De este modo, se identifica qué dispositivo está enviando datos al servidor.

Las especificaciones para LoRaWAN definen tres tipos de dispositivos (Clase A, Clase B y Clase C) y está pensada para una topología de red en estrella (comunicación dispositivos - GW). Todos los dispositivos LoRa deben implementar la Clase A, mientras que las clases B y C son extensiones de la misma.

- **Clase A:** el dispositivo soporta comunicación bidireccional entre el NS y un GW. Los mensajes pueden ser enviados por los dispositivos en cualquier momento, mientras que la recepción de mensajes podrá ocurrir solamente durante dos ventanas de tiempo establecidas a los 1 y 2 segundos luego del envío de un mensaje (ver figura 3.2). El GW puede responder en cualquiera de esas 2 ventanas pero no en ambas.

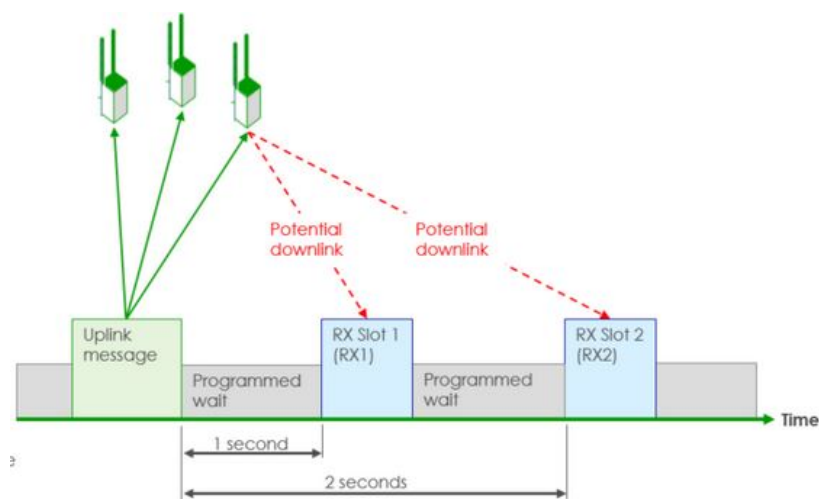


Figura 3.2: Dispositivo LoRa Clase A. Luego de enviar un mensaje al gateway (uplink), el dispositivo abre dos ventanas de recepción (RX Slot 1 y RX Slot 2) en los siguientes 1 y 2 segundos para la posible recepción de mensajes enviados por el gateway (downlink). Imagen tomada de [28].

- **Clase B:** además de las ventanas de recepción existentes en el Clase A, el dispositivo se sincroniza con el GW para abrir ventanas de recepción programadas. El GW envía beacons al dispositivo que le dicen en que momento comenzar y terminar de abrir ventanas de recepción periódicamente. Esto le permite al GW saber cuando el dispositivo estará dispuesto a recibir mensajes. Esta es una clase intermedia entre la Clase A y la Clase C en lo que a consumo de potencia y latencia respecta.

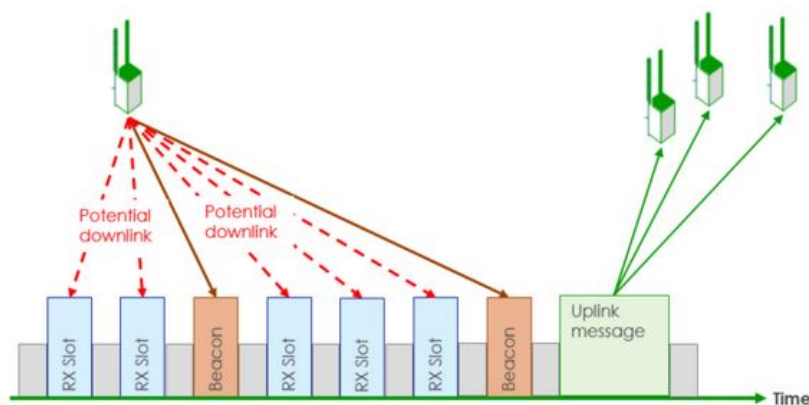


Figura 3.3: Dispositivo LoRa Clase B. Además de las ventanas de recepción existentes en un dispositivo Clase A, un Clase B dispone de ventanas de recepción que se abren periódicamente durante un tiempo delimitado por las balizas (beacons). Imagen tomada de [28].

- **Clase C:** la ventana de recepción se cierra solamente cuando el dispositivo envía un mensaje al GW, el resto del tiempo permanece abierta. Esto permite

3.2. LMiC

una menor latencia en la comunicación a la vez que consume una mayor cantidad de energía.

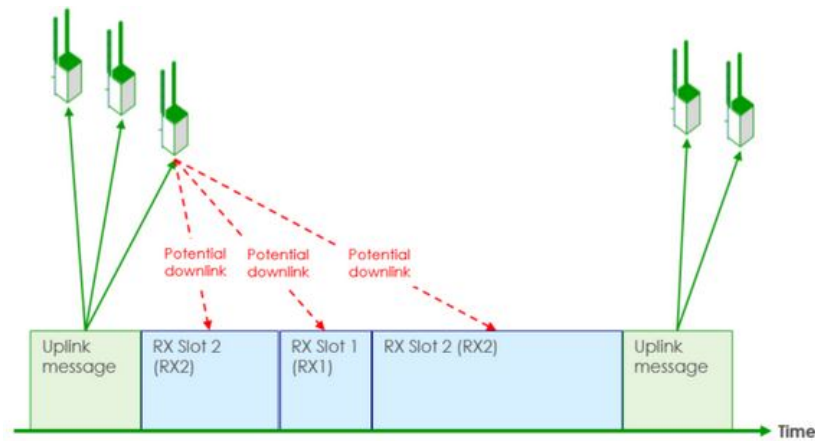


Figura 3.4: Dispositivo LoRa Clase C. Luego de enviar un mensaje al GW (uplink), la ventana de recepción (RX Slot) permanece abierta hasta el siguiente uplink. Esto permite la recepción por parte del dispositivo de mensajes enviados por el GW (downlink) en cualquier momento. Imagen tomada de [28].

En este proyecto se utilizarán dispositivos Clase A debido a que es la clase que mejor se adapta a los requerimientos.

3.2. LMiC

Para poder comunicar el NS con el Gateway usando LoRaWAN se utilizó la librería de IBM LMiC. Se describe a continuación dicha librería utilizando el manual proporcionado por IBM para el uso de la misma [29].

La LMiC es una implementación portable de las especificaciones de LoRa MAC para utilizar en el lenguaje de programación C. La misma se encarga de manejar los estados de la MAC y los dispositivos que contienen la radio para la comunicación a través de LoRa **para la radio SX1276**. La LMiC incluye una capa de abstracción de Hardware (HAL), de manera de poder utilizarla en diversos dispositivos (en el caso de este proyecto, la radio RFM95).

La librería asegura el cumplimiento de las especificaciones de LoRaWAN y los requerimientos de tiempo del protocolo MAC. Soporta la comunicación en la banda EU-868 y US-915, para dispositivos de Clase A y B. En Uruguay, la URSEC determinó que se puede utilizar para LoRa la banda 915-928MHz. Para poder cumplir con todos los requerimientos, se debe programar el código de aplicación con un modelo de programación dirigida por eventos. Este es un paradigma de programación en el que tanto la estructura como la ejecución de los programas van determinados por los sucesos que ocurran en el sistema.

Capítulo 3. Software

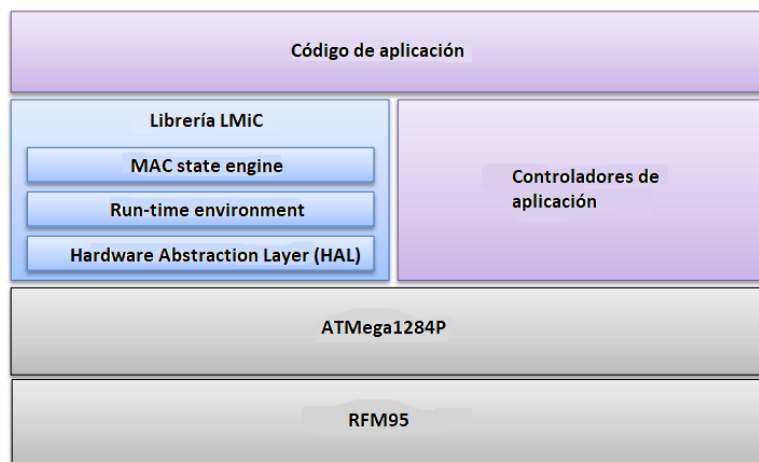


Figura 3.5: Vista de alto nivel de todos los componentes del dispositivo de aplicación. Basado en [29].

La biblioteca LMiC ofrece un modelo de programación simple basado en eventos donde todos los eventos de protocolo son enviados a la función `onEvent()`. En este modelo, el código de la aplicación se ejecuta cuando el scheduler llama a la función `os_runloop()` (ejecutando los eventos previamente agendados). Sin embargo, los eventos tienen la particularidad que no pueden “tardar demasiado”, pues la LMiC debe asegurar los requerimientos de tiempo de LoRaWAN y MAC (ver [29]). La documentación no es específica respecto a qué se refiere con que no puede “tardar demasiado”, no se da un tiempo máximo de duración del evento. Lo que sí menciona la documentación es que deberían actualizar estados y programar nuevos eventos. En la sección 6.1, se desarrollan algunos de los problemas encontrados por este hecho. El código de aplicación para funcionar correctamente con la LMiC debe tener la siguiente estructura:

1. Inicializar el “run-time environment” utilizando la función de la LMiC `os_init()`.
2. Agendar tareas (esto se puede realizar de forma directa o utilizando las funciones disponibles en la LMiC).
3. Ejecutar las tareas agendadas en el ítem (2) haciendo uso de la función `os_runloop_once()`.

Un ejemplo puede verse en la figura 3.6:

```

void setup() {
    pinMode(buzzer, OUTPUT);
    pinMode(motor, OUTPUT);
    digitalWrite(buzzer, LOW);
    digitalWrite(motor, LOW);
    Serial.begin(115200);
    Serial.println(F("Starting"));
    ss.begin(9600);
    // LMiC init
    initLora();
    // Start job
    get_gps_data(&sendjob_gps);
    // Start job
    do_sendLAT(&sendjobLAT);
}

void loop() {
    delay(1);
    os_runloop_once();
}

```

Figura 3.6: Ejemplo de utilización de la LMiC. En el setup se puede observar la inicialización del “run-time environment” a través de la función `initLora()` que llama a `os_init()`, el agendado de tareas a las funciones `get_gps_data(...)` y `do_sendLAT(...)`, y la ejecución de esas tareas en el loop principal a través de `os_runloop_once()`. `Setup()` y `loop()` son funciones de Arduino que permiten la inicialización de variables y ejecución del cuerpo principal del código respectivamente.

La LMiC pone a disposición funciones como `os_setCallback()` (agrega una tarea inmediatamente a la cola de espera) y `os_setTimedCallback()` (agrega una tarea a la cola en un momento establecido). Para el manejo de las tareas, se cuenta con una estructura de datos `osjob_t` que contiene información de contexto sobre las mismas. Una vez agendadas las tareas, se usa la función `os_runloop_once()` para ejecutarlas. La lista completa de funciones de la LMiC se encuentra en el manual que puso a disposición IBM para el uso de la misma [29].

También cuenta con una función `on_event()` que permite tomar acciones dependiendo del suceso de determinados eventos generando variables del tipo `ev_t` (eventos). Para el desarrollo del software de este proyecto, el evento que se utilizó como indicador para realizar acciones fue el que se origina al enviar un mensaje (**EV_TXCOMPLETE**).

En el caso de este proyecto se utilizó la librería LMiC adaptada para Mo-teino [30]. Dicho proyecto se modificó para que transmitiera en la banda permitida (915MHz). A su vez, dado que el Gateway proporcionado por Teliot estaba configurado para recibir en la sub-banda 1 (915.1MHz-916.9MHz), se configuró la transmisión en dicha sub-banda.

3.3. Arquitectura de Software

Se utilizará la arquitectura y los servidores de EveryNet, una empresa que provee soluciones de IoT. Tomaremos la arquitectura de trabajo sugerida por EveryNet (estándar LoRaWAN), la cual se puede ver en la figura 3.7 a continuación:

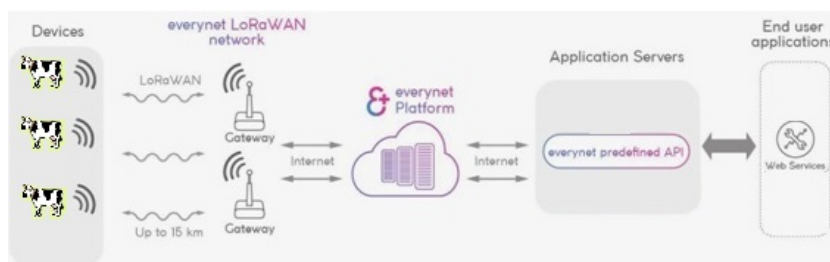


Figura 3.7: Arquitectura HW propuesta por EveryNet. Figura basada en [31].

En nuestro caso, los Devices serán los NS diseñados. La información se envía del NS al servidor central siguiendo el formato definido en la sección 3.3.1. El Gateway sube los datos recibidos a la plataforma de Everynet vía Internet (LoRaWAN sobre TCP/IP). La plataforma de Everynet dispone de una API, que puede ser accedida por un servicio web. Dicho servicio web consulta y toma los datos recibidos por el Gateway. La generalidad de la arquitectura permite que otra red LoRaWAN se pueda integrar a nuestra solución.

Se utiliza Amazon Web Services (de ahora en adelante AWS), para diseñar el frontend y el backend de la página web (End user application).

Para ello se crea una instancia en AWS y se abre su puerto 80 al mundo. De esta manera, podemos acceder a la instancia con un navegador web a través de su IP pública (<http://18.222.96.150:3000/>) Se levanta un servicio que consulta a la API de EveryNet si hay nuevos datos disponibles (de los dispositivos). En caso de haberlos, la API de Everynet devuelve dichos datos, que serán almacenados por el servicio en una base de datos que se encuentra en la instancia de AWS (dicha instancia será el servidor central). Los datos se procesan y se muestran en la web.

A continuación se desarrolla el formato elegido para el envío de datos desde el NS al servidor central.

3.3.1. Geolocalizacion

El formato definido para el envío de datos desde el NS al servidor central es el siguiente:

1. Los datos de latitud se envían en el formato **A -XX.XXXXXX HHMMSS Z**, donde la **A** indica que se trata de la latitud, **-XX.XXXXXX** es la latitud, **HHMMSS** la hora en formato hora:min:seg, y **Z** es una variable booleana que vale 0 si el animal está dentro del perímetro virtual, y 1 si está fuera. Después de la **Z**, debe haber un espacio, y luego nada más.

3.3. Arquitectura de Software

2. Los datos de longitud se envían en el formato `0 -XX.XXXXXX HHMMSS`, donde la `0` indica que se trata de la longitud, `-XX.XXXXXX` es la longitud, `HH:MM:SS` la hora en formato hora:min:seg. Después de los segundos debe haber un espacio, y luego nada más.

Para calcular la cantidad de bytes enviados, se debe agregar al payload efectivo de cada paquete los encabezados de la librería LMIC (ver figura 3.5). Por el formato definido para el envío de latitud, se tiene para dichos paquetes 21 bytes de payload efectivo transmitido, mientras que para los paquetes de longitud serán 19 bytes. LoRaWAN agrega al menos 13 bytes en encabezados para la transmisión de paquetes [32] (tiene campos opcionales, por eso pueden ser más de 13 bytes). Por lo tanto, se tiene un tamaño mínimo de paquete de 32 bytes (longitud), y 34 bytes (latitud).

Everynet permite visualizar en su interfaz web el tamaño del paquete recibido (en los paquetes de uplink, ver opción “size”). En dicha interfaz se visualizó que los paquetes de latitud eran de 39 bytes, mientras que los paquetes de longitud eran de 38 bytes. Esta diferencia se debe a que el encabezado de la LMIC es mayor a los 13 bytes mínimos de LoRaWAN.

Para obtener la latitud y la longitud del dispositivo hace falta configurar el GPS M8N. En la sección a continuación se desarrolla cómo se configuró dicho dispositivo, para luego conectarlo con el Moteino y obtener periódicamente la latitud y longitud del dispositivo con la frecuencia de muestreo deseada.

3.3.2. Configuración del GPS

Mediante el software u-center se configuró el GPS M8N. Para esto se utilizó un FTDI con el fin de poderse comunicar con el GPS a través de la UART conectándolo al USB de una computadora. Este programa permite modificar diversos aspectos de funcionamiento del GPS tales como frecuencia de muestreo y protocolo de comunicación a utilizar, entre otros. Se decidió trabajar con una frecuencia de muestreo de 10 Hz y utilizar el protocolo por defecto que utiliza el GPS. Dicho protocolo consiste en enviar datos a través de una cadena de caracteres con los siguientes campos:

- Carácter de inicio - “\$”.
- Campo de dirección - Solo contiene letras mayúsculas y se divide en dos campos:
 - Identificador “GP” - indica que son datos del GPS.
 - “P” - indica que es un mensaje del propietario (estos se usan para configurar el GPS o realizar consultas de su estado).
- Formato de oración - indica el contenido del mensaje (“GLL” indica que se enviará la latitud y longitud).
- Campo de datos datos espaciados por “,”.

Capítulo 3. Software

- Campo de checksum - comienza con un “*” seguido de 2 caracteres que representan un número en hexadecimal.
- Fin de secuencia - un carácter de fin de línea.

Como se muestra en la siguiente figura:

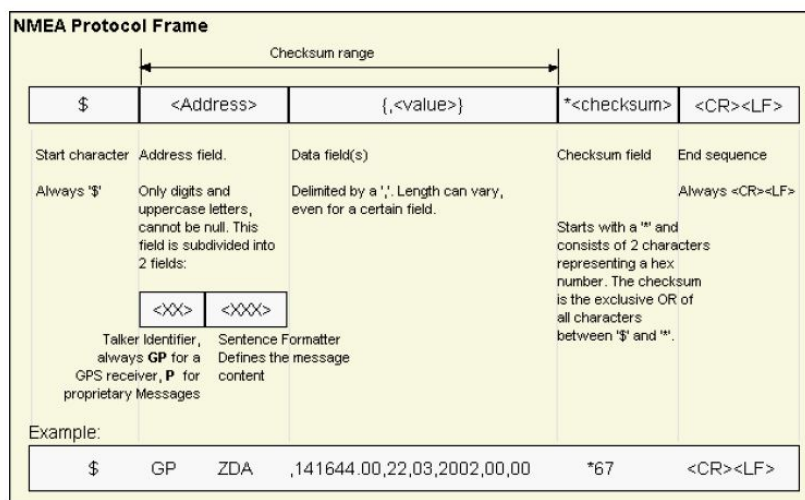


Figura 3.8: Estructura del protocolo NMEA utilizado por el GPS

3.4. Variables de interés del Perímetro Virtual

Al momento de implementar la funcionalidad de PV se hizo necesario definir variables que son de interés para resolver este problema. Todas estas variables se fijan con los valores por defecto que se describen en la tabla 3.3 en la sección 3.4.2. Luego podrán ser modificadas mediante una interfaz web, pero los valores modificados son volátiles (es decir, si se reinicia el programa se pierden las configuraciones). Sin embargo, los valores por defecto se cargan con el programa, es decir que no son volátiles.

Estas variables se basan en la idea de definir 4 zonas donde el dispositivo se comportará de manera distinta. Explicaremos primero cuáles son las variables de interés, y luego nos concentraremos en cómo se comportará el dispositivo según en qué zona se encuentra.

Las variables de interés son las siguientes:

- Perímetro Virtual: dados 4 puntos $P1, P2, P3, P4 \in \mathbb{R}^2$, definimos por perímetro virtual al cuadrilátero convexo que se forma al unir con segmentos los 4 puntos dados. Como el cuadrilátero debe ser convexo, $P1..P4$ deben estar dispuestos de manera horaria, o antihoraria. $P1..P4$ están conformados

3.4. Variables de interés del Perímetro Virtual

por las siguientes coordenadas: (latitud punto 1, longitud punto 1) (latitud punto 2, longitud punto 2) (latitud punto 3, longitud punto 3) (latitud punto 4, longitud punto 4). Se le llamará a las coordenadas (P1X, P1Y)(P2X, P2Y)(P3X, P3Y) y (P4X, P4Y). En la figura 3.9 podemos ver los puntos antes mencionados, que definen un posible perímetro virtual. En ese caso es rectangular, pero no tiene por qué serlo. Cada coordenada seteada es una variable de tipo float.

- PFE (Primer franja de estimulación): esta variable determina la mínima distancia en metros entre la zona I y la zona IV. Es decir, dado el perímetro virtual, y los segmentos que unen dichos puntos del perímetro virtual, PFE delimita la zona I (ver figura 3.9). Esta variable es de tipo float.
- SFE (Segunda franja de estimulación): esta variable determina la mínima distancia en metros entre la zona II y la zona IV. Es decir, dado el perímetro virtual, y los segmentos que unen dichos puntos del perímetro virtual, SFE y PFE delimitan la zona II (ver figura 3.9 a continuación). La distancia SFE debe ser menor a PFE. Esta variable es de tipo float.

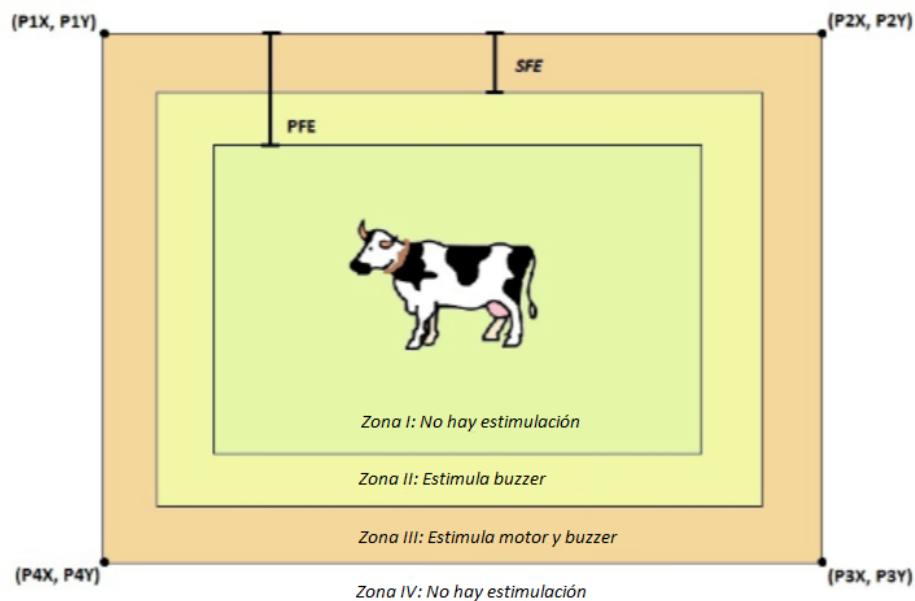


Figura 3.9: Perímetro virtual

- STI (Estimulación encendida-apagada): esta variable determina si se estimulará o no al animal en caso de encontrarse cerca del perímetro virtual. Es una variable de tipo booleana.
- STD (Duración de la estimulación): esta variable determina el tiempo de estimulación en milisegundos. Cuando el dispositivo pasa de la zona I a la

Capítulo 3. Software

zona II, comienza una cuenta regresiva del tiempo STD. Mientras el contador no llegue a cero, el dispositivo estimula (en la zona II con buzzer, en la zona III con buzzer y motor). Una vez que finaliza el tiempo de estimulación STD, comienza un período refractario, en el cual no hay estimulación. De esta manera, se evita sobreestimar al animal por tiempos prolongados, con su consecuente ahorro de energía. La variable STD es de tipo long int.

- RED (Duración del refractario): esta variable determina el tiempo refractario durante el cual se descansa entre una estimulación y la siguiente en las zonas II y III. Dado que no se desea estresar al animal ni consumir demasiada energía, se lo estimula durante un tiempo STD, se deja de estimular durante un tiempo RED, y así sucesivamente mientras nos encontremos en la zonas II y III. El contador se reinicia si el dispositivo entra en la zona I o en la zona IV. La variable RED es de tipo long int.
- REE (Duración del refractario al volver a entrar al PV): esta variable determina el tiempo durante el cual el dispositivo puede ingresar al perímetro virtual sin recibir estimulaciones. Se intenta de esta manera que el animal no reciba estimulaciones que pueda interpretar de manera errónea, alejándola del perímetro establecido. Esta variable es de tipo long int.

En la figura 3.10 podemos ver el diagrama de flujo del funcionamiento esperado del dispositivo, según en qué zona se encuentre. Se muestra en dicha figura cómo se utiliza STD, RED y REE.

- FRE: (Frecuencia del buzzer): esta variable determina la frecuencia del sonido emitido por el buzzer. Dado que se utilizó un buzzer pasivo, el cual no tiene un oscilador interno, debemos establecer la frecuencia del sonido a través del microcontrolador. Para ello disponemos de la función `tone(Dur,FRE)`, incluida en las bibliotecas de Arduino. La función `tone(Dur,FRE)` genera una onda cuadrada con un 50 % de duty cycle. Las variables Dur y FRE representan la duración del tono y su frecuencia. La variable FRE es de tipo int.
- BVO (Volumen del buzzer): esta variable determina el volumen del buzzer. Utilizamos la función `analogWrite` del lenguaje de programación de Arduino para alimentar al buzzer a través de una señal PWM. Variando el ciclo de trabajo, logramos modificar la tensión de alimentación y por lo tanto, su volumen. Para obtener un buen funcionamiento del mismo, se agregó un capacitor de 1000uF en paralelo entre el pin de alimentación y GND. Esta variable es de tipo int y su valor varía entre 0 y 255 (variable analógica de 8 bits).

3.4. Variables de interés del Perímetro Virtual

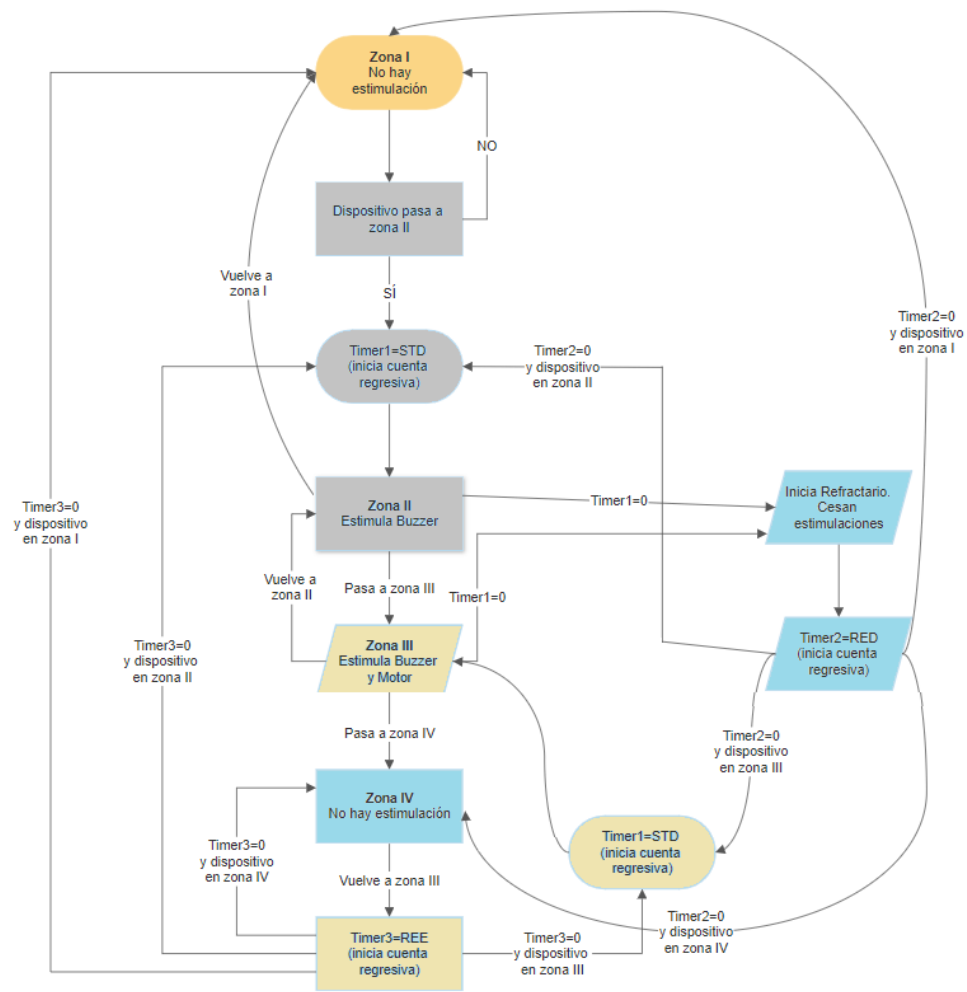


Figura 3.10: Diagrama de estados según dónde se encuentra el dispositivo, si STI=1.

- BPD (Duración del pitido del buzzer): cuando se estimula con el buzzer, el pitido no es constante. Se enciende con cierta intermitencia, proporcional a la distancia a la cual se encuentra del perímetro virtual. Por ello, definimos la variable BPD, que determina cuánto tiempo (en milisegundos) dura cada pitido del buzzer cuando se está estimulando. Esta variable no debe confundirse con STD. STD determina cuánto tiempo estimularemos, y BPD cuánto tiempo dura cada pitido intermitente del buzzer cuando se está estimulando (por defecto $STD=60000ms$, $BPD=100ms$). La variable BPD es de tipo int.
- BVE (Velocidad del buzzer): esta variable determina la cantidad máxima de pitidos, en el lapso de un segundo, que se escucharán cuando el dispositivo esté próximo a ingresar a la zona III (el tiempo de estimulación del buzzer

Capítulo 3. Software

se fijó en 1s). Dado que cada estimulación dura el tiempo en milisegundos seteado en BPD, entonces BVE deberá ser menor que 1000ms/BPD (en caso contrario, estimula con un pitido constante al borde de la zona III). En la figura 3.11 a continuación se observa cómo varía la cantidad de pitidos en función de la distancia respecto al perímetro virtual. La variable BVE es de tipo int.

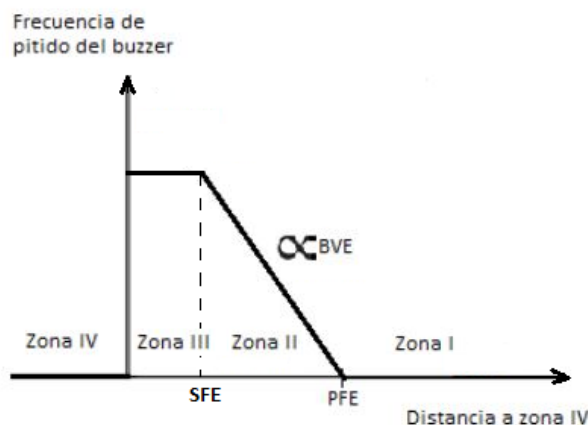


Figura 3.11: Explicación de variable BVE.

- MBO (Encendido del buzzer y motor): esta variable determina el tiempo de estimulación de tanto el buzzer como el motor en la zona III. Cuando se estimula con el buzzer, el pitido no es constante, sino que es intermitente. Lo mismo sucede con el motor. Dado que ambas estimulaciones son intermitentes, se decide sincronizar las estimulaciones en la zona III. Definimos la variable MBO, que representa cuánto tiempo (en ms) dura cada pitido del buzzer y vibración del motor cuando se está estimulando. Esta variable es de tipo int.
- MBF (Apagado del buzzer y motor): las estimulaciones en la zona III son intermitentes. Mientras nos encontremos en el estado “Estimula Buzzer y Motor” (ver figura 3.10), estimulará el buzzer y el motor de manera intermitente. Esta estimulación intermitente durará un tiempo MBO, y descansará un tiempo MBF. MBF no será configurable, a diferencia del resto de las variables descritas en esta sección, sino que es constante igual a 200ms. Se menciona esta variable por ser de interés del perímetro virtual. Sin embargo, se considera trabajo a futuro determinar, parametrizar y hacer configurables más variables de interés del perímetro virtual (ver sección 6.4) La variable MBF es de tipo int.

A continuación se describe cómo se utilizan las variables presentadas para generar el funcionamiento esperado según la zona en la que se encuentre el dispositivo.

3.4.1. Funcionamiento esperado por zonas

La figura 3.9 resume el comportamiento esperado por zonas. En la figura 3.10 podemos ver el diagrama de estados completo del dispositivo, y los eventos que desencadenan los cambios de estado.

Zona I

Si el dispositivo se encuentra en la zona I, se reportará la posición de la vaca cada 30s (la frecuencia del GPS se fijó constante igual a 10Hz. Sin embargo, se priorizó el ahorro de energía y se decidió reportar la posición de la vaca cada 30s). Mientras el animal continúe en esta zona, no será estimulado ni con el buzzer ni con el motor. Esta es la zona en la que se quiere que el animal permanezca.

Zona II

Si el dispositivo se encuentra en la zona II, y $STI=1$, se estimula con el buzzer (y no con el motor). Se estimula el tiempo seteado en STD, y luego se descansa el tiempo seteado en RED. En el tiempo que estimula, el buzzer no emitirá un pitido constante, sino que emitirá pitidos intermitentes. El tiempo de pitido está determinado por BPD. El buzzer tendrá su volumen determinado por BVO y el sonido emitido tendrá su frecuencia determinada por FRE. A medida que el dispositivo se aleja de la zona I y se acerca a la zona III, la frecuencia del pitido aumenta, hasta un máximo de BVE cuando el dispositivo está a punto de entrar en la zona III.

Zona III

Si el dispositivo se encuentra en la zona III, y $STI=1$, se estimula con el buzzer y el motor. Se estimula el tiempo seteado en MBO, y luego se descansa durante el tiempo seteado en MBF. En el tiempo de estimulación, el buzzer emitirá pitidos intermitentes a frecuencia de pitido BVE (se emitirá constantemente a la máxima frecuencia de pitido, ya no se aumentará más gradualmente como en la zona II). El buzzer tendrá un volumen determinado por BVO y la frecuencia del sonido emitido estará determinada por FRE. El motor estimula de manera constante durante el tiempo de estimulación seteado en MBO, sincronizado con el buzzer, y luego cesa la estimulación durante un el tiempo seteado en MBF.

Zona VI

Si el dispositivo accede a esta zona, significa que el animal atravesó el perímetro virtual a pesar de los estímulos recibidos. Hay muchas formas posibles de proceder ante esta situación. Para este proyecto, se decidió que mientras el dispositivo se encuentre en esta zona, se desactivan las estimulaciones. Aparte, si el dispositivo volviera a entrar en la zona III, comienza a contar el tiempo setado en REE. Durante dicho tiempo, no habrán estimulaciones ni en la zona III ni en la zona II. Si el animal vuelve a la zona IV, se resetea el contador, que se volverá a iniciar una vez que el animal vuelva a la zona III. En el caso ideal, el animal atraviesa las zonas III y II durante el tiempo seteado en REE, y luego se vuelve al comportamiento deseado de la zona I. Con la variable REE se pretende evitar que el animal sea estimulado, y como consecuencia se aleje, solo que esta vez hacia la zona IV que es de donde venía. En la sección 6.4, trabajo a futuro, se plantean mejoras posibles a esta solución.

3.4.2. Seteo de parámetros

Dentro de los requerimientos del proyecto se encuentra que los principales parámetros de interés del perímetro virtual sean configurables de manera remota.

Para ello fue necesario desarrollo a nivel de servidor y a nivel del NS. A nivel del servidor, como se mencionó en la sección 1.6.2, el desarrollo estuvo a cargo de Raúl Marichal bajo la dirección del Ing. Martín Pedemonte. Fue necesario crear una interfaz web, desde la cual se selecciona el dispositivo, y se seleccionan los valores que se le quiere asignar a las variables de interés a configurar.

Como se puede ver en la figura 3.7, Everynet permite la comunicación bidireccional entre la web y el dispositivo. Para ello, se utiliza la API de Everynet. Dicha API interactúa con la plataforma de Everynet, que a su vez configura al Gateway para que utilice la ventana de downlink (ver sección 3.1) para enviarle al dispositivo la información (en este proyecto el parámetro a setear y su valor). Dicha información deberá tener un formato preestablecido, de manera que pueda ser procesada por el NS. Dicho formato se describe en las tablas 3.1 y 3.2 que se encuentran a continuación.

A nivel del NS, el programa deberá procesar la información recibida desde la interfaz web, actualizando la variable correspondiente con el valor deseado.

Definimos el formato de envío de datos del servidor central al dispositivo de la siguiente manera:

Qué mando	Qué significa	Ejemplo respuesta esperada
CHP=1	Change Parameters ON. Ver tabla 3.2	CHP=1 1
PAR=P1X	Devuelve el valor del parámetro P1X	-34.898049
PAR=P1Y	Devuelve el valor del parámetro P1Y	-56.165850
PAR=P2X	Devuelve el valor del parámetro P2X	-34.898954
PAR=P2Y	Devuelve el valor del parámetro P2Y	-56.165762
PAR=P3X	Devuelve el valor del parámetro P3X	-34.899035
PAR=P3Y	Devuelve el valor del parámetro P3Y	-56.166902
PAR=P4X	Devuelve el valor del parámetro P4X	-34.898461
PAR=P4Y	Devuelve el valor del parámetro P4Y	-56.166970
PAR=0	Devuelve CHP STI STD	0 1 060000
PAR=1	Devuelve RED REE BPD BVO	060000 030000 0100 255
PAR=2	Devuelve FRE BVE MBO MBF PFE SFE	3000 020 2000 2000 15 05

Tabla 3.1: Seteo de parámetros mientras CHP=0.

Como se puede ver en la tabla 3.1, CHP se debe setear en 1 para modificar los parámetros, mientras que las funciones PAR devuelven los valores actuales de los parámetros de interés en el dispositivo. PAR sirve para que la interfaz web pregunte los valores actuales de los NS y los almacene en una base de datos.

Al enviar PAR=1 se devuelven los parámetros separados por un espacio. Nótese que los parámetros tienen exactamente la cantidad de cifras que se indica en la

3.4. Variables de interés del Perímetro Virtual

columna “Ejemplo respuesta esperada” de la tabla 3.1 (que a su vez coincide con la cantidad de cifras que debe tener el parámetro a la hora de setearlo, ver tabla 3.2)

También podemos notar que al setear parámetros, la respuesta es el comando que se envió, seguido de el valor del parámetro que se quiso setear. En caso de que se haya seteado correctamente, estos dos valores deberán coincidir. Por eso la respuesta a $CHP=0$ debería ser $CHP=0$ 0. Si fuera $CHP=0$ 1, esto implicaría que el seteo no funcionó, y se deberá volver a setear el parámetro. En la tabla 3.2 a continuación se puede ver como setear todos los parámetros de interés, una vez que $CHP=1$.

En las tablas 3.1 y 3.2 se encuentra implícito que la interfaz web es la que manda el comando para modificar el parámetro, y la respuesta esperada proviene del NS.

Qué mando	Qué significa	Respuesta esperada
$CHP=0$	Setea $CHP=0$.	$CHP=0$ 0
$STI=1$	Stimulation ON	$STI=1$ 1
$STI=0$	Stimulation OFF	$STI=0$ 0
$STD=XXXXXX$	Setea STD	$STD=XXXXXX$ XXXXXX
$P1X=-XX.XXXXXX$	Setea P1X	$P1X=-XX.XXXXXX$ -XX.XXXXXX
$P1Y=-XX.XXXXXX$	Setea P1Y	$P1Y=-XX.XXXXXX$ -XX.XXXXXX
$P2X=-XX.XXXXXX$	Setea P2X	$P2X=-XX.XXXXXX$ -XX.XXXXXX
$P2Y=-XX.XXXXXX$	Setea P2Y	$P2Y=-XX.XXXXXX$ -XX.XXXXXX
$P3X=-XX.XXXXXX$	Setea P3X	$P3X=-XX.XXXXXX$ -XX.XXXXXX
$P3Y=-XX.XXXXXX$	Setea P3Y	$P3Y=-XX.XXXXXX$ -XX.XXXXXX
$P4X=-XX.XXXXXX$	Setea P4X	$P4X=-XX.XXXXXX$ -XX.XXXXXX
$P4Y=-XX.XXXXXX$	Setea P4Y	$P4Y=-XX.XXXXXX$ -XX.XXXXXX
$RED=XXXXXX$	Setea RED	$RED=XXXXXX$ XXXXXX
$REE=XXXXXX$	Setea REE	$REE=XXXXXX$ XXXXXX
$BPD=XXXX$	Setea BPD	$BPD=XXXX$ XXXX
$BVO=XXX$	Setea BVO	$BVO=XXX$ XXX
$FRE=XXXX$	Setea FRE	$FRE=XXXX$ XXXX
$BVE=XXX$	Setea BVE	$BVE=XXX$ XXX
$MBO=XXXX$	Setea MBO	$MBO=XXXX$ XXXX
$PFE=XX$	Setea PFE	$PFE=XX$ XX
$SFE=XX$	Setea SFE	$SFE=XX$ XX

Tabla 3.2: Seteo de parámetros una vez que $CHP=1$.

A continuación se plantean los rangos válidos de las variables de interés para el correcto funcionamiento del dispositivo, junto con los valores fijados por defecto en el programa entregado junto a la documentación de este proyecto.

Capítulo 3. Software

Parámetro a configurar	Rango válido	Valor por defecto
CHP	0,1	0
STI	0,1	1
STD=XXXXXX	$MBO + 000200 < STD \leq 999999$ ¹	060000 (60000ms)
P1X=-XX.XXXXXX	(-90.000000,+90.000000)	-34.768895
P1Y=-XX.XXXXXX	(-90.000000,+90.000000)	-55.638755
P2X=-XX.XXXXXX	(-90.000000,+90.000000)	-34.768681
P2Y=-XX.XXXXXX	(-90.000000,+90.000000)	-55.638726
P3X=-XX.XXXXXX	(-90.000000,+90.000000)	-34.768714
P3Y=-XX.XXXXXX	(-90.000000,+90.000000)	-55.638250
P4X=-XX.XXXXXX	(-90.000000,+90.000000)	-34.768928
P4Y=-XX.XXXXXX	(-90.000000,+90.000000)	-55.638270
RED=XXXXXX	$MBO < RED < \frac{PFE+SFE}{\bar{v}}$ ²	010000 (10000ms)
REE=XXXXXX	$\frac{PFE+SFE}{\bar{v}} < REE \leq 999999$	020000 (20000ms)
BPD=XXXX	$0050 < BPD < 1000$	0100 (100ms)
BVO=XXX	$000 < BVO < 255$	255
FRE=XXXX	$0100 < FRE < 9999$	3000(Hz)
BVE=XXX	$001 < BVE < 100$	020
MBO=XXXX	$0100 < MBO < 9999$	2000 (2000ms)
PFE=XX	$SFE < PFE$	20 (20m)
SFE=XX	$05 < SFE$	10 (10m)

Tabla 3.3: Seteo de parámetros una vez que CHP=1.

En lo que refiere a la tabla 3.3, se realizan los siguientes comentarios:

- Como se desarrolló en la sección 3.4, MBF=200ms constante.²
- Se sugiere que RED sea menor al tiempo que demora una vaca promedio en atravesar las zonas II y III.³
- Como se mencionó en la sección 3.4, el PV se debe setear de manera horaria.
- Para que exista un perímetro virtual, los puntos se deben fijar con distancia entre ellos mucho mayor a SFE+PFE. De esta manera nos aseguramos de que exista la zona I.

²000200=200ms=MBF en la configuración del parámetro STD. Es importante que haya muchas estimulaciones antes de entrar al refractario, por eso STD debe ser mucho mayor a MBO+MBF.

³ \bar{v} es la velocidad promedio de la vaca. No queremos que el animal atravesase las zonas II y III sin ser estimulado. Por eso se sugiere que el tiempo que el dispositivo no estimula (RED) sea menor al tiempo en el que la vaca puede atravesar a velocidad promedio esa franja. Se sugiere utilizar la velocidad promedio de una persona caminando $\bar{v} = 1,39m/s$.

3.5. Perímetro Virtual

Es importante mencionar que el dispositivo envía datos siempre que no esté estimulando. Si está estimulando, el dispositivo toma datos, pero no los envía al servidor hasta que cesan las estimulaciones. Se optó por este camino porque la LMIC funcionaba mal si se estaba enviando y estimulando al mismo tiempo, dado que los eventos tomaban demasiado tiempo y eso contradice lo mencionado en la sección 3.2 (“Jobs must not be long-running in order to ensure seamless operation!”)

Para poder implementar el diagrama de estados que se mostró en la figura 3.10, se necesita saber en qué zona se encuentra el dispositivo. Para ello se definen los segmentos $r1$, $r2$, $r3$ y $r4$, que unen $P1$ con $P2$, $P2$ con $P3$, $P3$ con $P4$ y $P4$ con $P1$ respectivamente. Se recuerda que $P1$, $P2$, $P3$ y $P4$ se definen en sentido horario (ver sección 3.4).

Luego se definen las distancias $d1$, $d2$, $d3$ y $d4$ del dispositivo a $r1$, $r2$, $r3$ y $r4$ (ver figura 3.12). Al mínimo de $d1, d2, d3$ y $d4$ se le llama distancia del dispositivo al perímetro virtual (de ahora en adelante DIS). DIS es la variable que se utiliza para determinar en qué zona se encuentra el dispositivo.

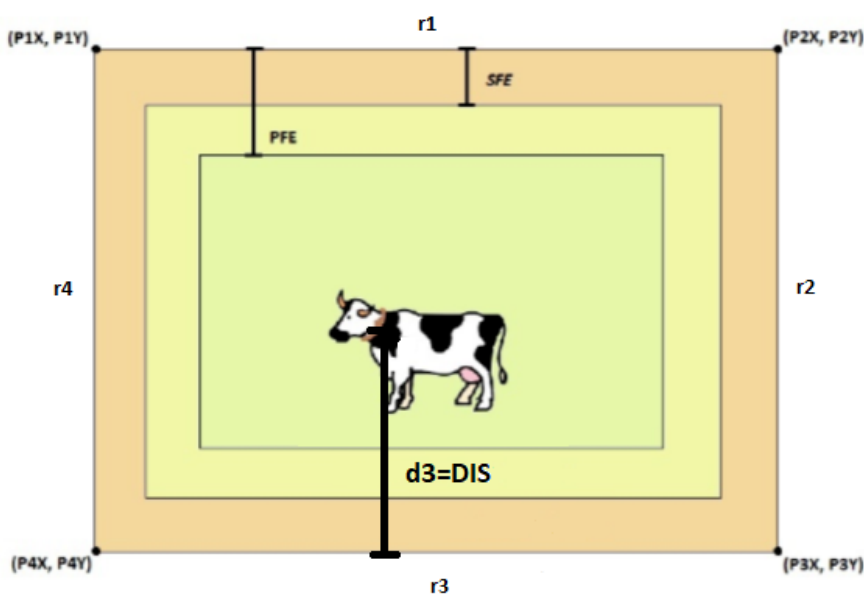


Figura 3.12: Cálculo de la distancia del dispositivo al perímetro virtual.

Se implementó la función `get-distance`, que determina el valor de DIS. Esta función recibe la posición del dispositivo (latitud y longitud) y el perímetro virtual, y devuelve la distancia DIS.

Cada vez que se recibe la posición del dispositivo, se llama a la función `get-distance`,

Capítulo 3. Software

obteniéndose la variable DIR. Luego con la variable DIR y los parámetros PFE Y SFE se determina en qué zona se encuentra el dispositivo.

- Zona I: $DIS > PFE$.
- Zona II: $SFE < DIS < PFE$.
- Zona III: $DIS < SFE$.
- Zona IV: Indefinido.

Para saber si se está en la zona IV, hay que resolver el siguiente problema: dado un punto y un perímetro virtual, determinar si el punto está dentro o fuera del perímetro virtual.

Nos informamos sobre las formas de determinar si un punto se encuentra dentro de un polígono. Hay muchos algoritmos que resuelven este problema. El más utilizado es el del *Ray casting*.

Dicho algoritmo se basa en calcular la cantidad de intersecciones de un rayo, que empieza en el dispositivo y va en cualquier dirección fija, con los bordes del polígono. Si el número de intersecciones es impar, el punto se encuentra dentro del polígono. Si por el contrario el número de intersecciones es par, el punto se encuentra fuera del polígono. [33]

Se encontró una forma de implementar este algoritmo en Python [34]. Para este proyecto lo que se hizo fue modificar el algoritmo en Python para que compile en Arduino. A dicha función se le llamó `pnpoly()`, que toma como parámetro la cantidad de vértices, la latitud y longitud del dispositivo, y la latitud y longitud de los 4 puntos del perímetro virtual. Dicha función `pnpoly()` devuelve en un booleano si el dispositivo se encuentra dentro (0) o fuera (1) del perímetro virtual. Es de esta función que obtenemos el parámetro Z que se manda al servidor (ver sección 3.3.1)

Se probó el algoritmo y se vio que funcionó de manera correcta. En el capítulo 5, resultados, se desarrollarán las pruebas con mayor profundidad. Con las funciones `get-distance()` y `pnpoly()` podemos determinar en qué zona nos encontramos. Estamos entonces en condiciones de implementar el código de aplicación de acuerdo al diagrama de estados (ver figura 3.10). El código de aplicación se encuentra adjunto a la documentación de la tesis.

3.6. Funciones utilizadas para la implementación del perímetro virtual

En esta sección se hará una breve descripción de las funciones utilizadas para la implementación del perímetro virtual. Algunas de estas funciones fueron desarrolladas en el proyecto mientras que otras están presentes en las librerías utilizadas para el manejo del GPS (TinyGPS++) y la LMiC mencionada en secciones anteriores.

3.6. Funciones utilizadas para la implementación del perímetro virtual

3.6.1. Funciones de la LMiC

void os_setCallback (osjob_t* job, osjobcb_t cb): función de la librería LMiC que agenda una tarea a una función inmediatamente.

void os_setTimedCallback (osjob_t* job, ostime_t time, osjobcb_t cb): función de la librería LMiC que agenda una tarea a una función para ser ejecutada en el timestamp “time”.

void os_clearCallback (osjob_t* job): función de la librería LMiC que elimina una tarea previamente agendada.

void os_runloop (): función de la LMiC que ejecuta las tareas agendadas.

int LMIC_setTxData2 (u1_t port, xref2u1_t data, u1_t dlen, u1_t confirmed): prepara el mensaje guardado en el buffer “data” de tamaño “dlen” para enviarlo en el siguiente instante de tiempo posible a través del puerto “port”. El parámetro “confirmed” indica sí se le solicitara una respuesta de confirmación al servidor una vez este reciba el mensaje.

Cuando la transacción se completa se genera el evento **EV_TXCOMPLETE**.

void onEvent (ev_t ev): recibe una variable del tipo ev_t (evento) de la estructura de datos de la LMiC y genera acciones en función de la variable recibida. En este proyecto solo se realizan acciones en caso de que el evento de entrada sea **EV_TXCOMPLETE** que indica que un mensaje se ha enviado correctamente (lo que a su vez implica que se dispone de una nueva ventana de recepción).

void initLora(): función de la LMiC que configura la MAC del dispositivo para la comunicación seteando sus parámetros (potencia de transmisión, canales a utilizar, etc).

3.6.2. Funciones de la libreria TinyGPS++

int gps.location.isValid(): indica si hay un dato válido para la lectura del GPS. En caso de haberlo devuelve 1, de lo contrario devuelve 0.

float gps.location.lat(): devuelve la latitud de las coordenadas obtenidas del GPS en float.

float gps.location.lng(): devuelve la longitud de las coordenadas obtenidas del GPS en float.

float gps.time.hour(): devuelve la hora obtenida por el GPS en float.

float gps.time.minute(): devuelve los minutos obtenidos por el GPS en float.

float gps.time.second(): devuelve los segundos obtenidos por el GPS en float.

3.6.3. Funciones implementadas

String prepareDataForTransmission(): prepara los datos obtenidos del GPS para ser transmitidos a través del uso de la función `LMIC_setTxData2(...)` mediante el formato expuesto en la sección 3.3.1. Para eso se convierten los mismos al formato `u1_t` (char) y se concatena la posición con los demás datos en un String que será guardado en el buffer utilizado para el envío del mensaje.

void do_sendLAT(osjob_t* j): cuando se le agenda una tarea esta función realiza dos acciones: Envía la latitud al GW. Agenda una tarea dentro de `TX_INTERVAL` segundos a la función `do_sendLON(...)` que se describirá mas adelante. La variable `TX_INTERVAL` se utiliza para setear el tiempo entre envío de mensajes.

void do_sendLON(osjob_t* j): cuando se le agenda una tarea esta función realiza dos acciones:

- Envía la longitud al GW.
- Agenda una tarea dentro de `TX_INTERVAL` segundos a la función `do_sendLAT(...)`.

float get_distance(float coord_x, float coord_y, float C1_x, float C1_y, float C2_x, float C2_y, float C3_x, float C3_y, float C4_x, float C4_y): recibe las coordenadas de un punto y cuatro vértices de un polígono simple. Devuelve la distancia mínima entre el punto y los lados del polígono.

int pnpoly(int nvert, float P1_x, float P2_x, float P3_x, float P4_x, float P1_y, float P2_y, float P3_y, float P4_y, float testx, float testy): recibe las coordenadas de un polígono simple de “nvert” vértices y determina si un punto con coordenadas “testx” y “testy” está dentro o fuera del mismo.

void get_gps_data(osjob_t* j): cuando se le agenda una tarea esta función guarda los datos del GPS (latitud, longitud, fecha, hora, etc) en las correspondientes variables y realiza las siguientes acciones:

- Obtiene la distancia mínima entre el punto correspondiente a las coordenadas del GPS y los lados del perímetro virtual previamente seteado (guardando las coordenadas de sus vértices en variables internas del software).
- Evalúa si el punto se encuentra fuera o dentro del perímetro a través de la función `pnpoly(...)`.

Si el punto se encuentra dentro de las zonas de estimulación, cancela todas las funcionalidades relacionadas con el envío y recepción de datos. En caso de estar encendida la flag **REFRACTARIO** inicializa un timer para contabilizar el tiempo.

Si el punto está fuera del perímetro y en determinado momento entra al mismo, se setea la variable **ENTRANCE_FLAG**. Debido a que en este estado se estará en un período refractario, también se inicializa un timer para contabilizar el tiempo. Esta función permanece en un “loop” agendándose una tarea a sí misma repitiendo esta secuencia de actividades periódicamente.

void stimulation (osjob_t* j): cuando se le agenda una tarea esta función se encarga de activar los estímulos (buzzer y motor) de acuerdo con lo expuesto en la sección 3.4.1 y gestionando los períodos de refractario.

Luego de realizar las funcionalidades antes mencionadas se realiza lo siguiente:

- Si se está en un período refractario se le agenda una tarea a la función **get_gps_data(...)**.
- Si no se está en un período refractario se le agenda una tarea a la función **stimulation (...)**.

void MSG_RECEPTION(char* str): recibe una cadena de caracteres para configurar o consultar los parámetros descritos en la sección 3.4 y realiza las siguientes acciones en función del valor de la variable **CHANGING_PARAMETERS**:

- **CHANGING_PARAMETERS = 1** : Modificación de parámetros. Se procesan los datos en la cadena de caracteres y se modifican los distintos parámetros del NS (ver sección 3.4.2).
- **CHANGING_PARAMETERS = 0** : Consulta de parámetros. Se procesan los parámetros del NS para ser enviados. Se concatena el contenido del buffer de recepción con los parámetros del NS (ver sección 3.4.2) y se envía dicho mensaje a través de la función **LMIC_setTxData2(...)**.

3.7. Interfaz web

Como se mencionó en la sección 1.6.2, el diseño de la prueba de concepto del backend del sistema, la interfaz web y su puesta en marcha en AWS fue realizado por Raúl Marichal, bajo la dirección del Ing. Martín Pedemonte.

Dicha interfaz tiene como finalidad posibilitar la interacción entre el usuario y el NS. De esta forma se podrán modificar determinados parámetros de interés, lo que deriva en los requerimientos que se detallan a continuación.

La interfaz web deberá posibilitar las siguiente acciones:

- **Creación y eliminación de perímetros virtuales:** Seteo y eliminación de un determinado perímetro virtual en los diferentes dispositivos. Estos perímetros deberán ser polígonos simples (sus lados no se intersectan) con el

Capítulo 3. Software

fin de ser compatibles con la función utilizada para detectar si un dispositivo está dentro o fuera del mismo. En caso de que esta última condición no se cumpla aparecerá un mensaje de alerta indicando el error “El perímetro virtual debe ser simple (sin intersecciones)”

- **Configuración de los dispositivos:** Modificación de los parámetros de interés (lista completa en tabla 3.2) para cada NS.(*)
- **Asignación de perímetros a determinados dispositivos:** Esto permite la asignación de diferentes perímetros virtuales para distintos grupos de dispositivos.
- **Visualización de la posición de cada dispositivo.**
- **Visualización del registro de la posición de cada dispositivo:** Con el fin de poder observar la trayectoria de un determinado dispositivo, deberá ser posible visualizar la misma en un mapa o en una tabla que contenga información de la variación de la posición con el tiempo.

En el anexo C se encuentra el manual de usuario para el uso de la interfaz web desarrollada.

(*) En lo que refiere a la configuración de los dispositivos, a nivel de backend de la interfaz se deben procesar los datos del frontend para llevarlos al formato preestablecido en la tabla 3.2. Si se quiere fijar $STD=60000$, el backend procesa ese dato y envía $STD=060000$ (STD se debe enviar con 6 cifras). Este procesamiento está a cargo del servidor y no del usuario. Si se intenta setear un parámetro con una cantidad de cifras mayor a las permitidas, se indica un error en la interfaz de usuario: “La cantidad de cifras es mayor a la permitida”. En caso de que se quiera setear un parámetro con un valor que no entra en el rango de los posibles, se indicará un error en la interfaz de usuario: “No es posible setear el parámetro a ese valor”. Se considerarán como valores posibles aquellos de la columna “Rango válido” de la tabla 3.3.

Capítulo 4

Diseño del encapsulado

En este capítulo se detallan las principales características del diseño del encapsulado del dispositivo, sus requerimientos iniciales, la elección del material y su fabricación. Se explica también el método de sujeción escogido y su forma de ajuste.

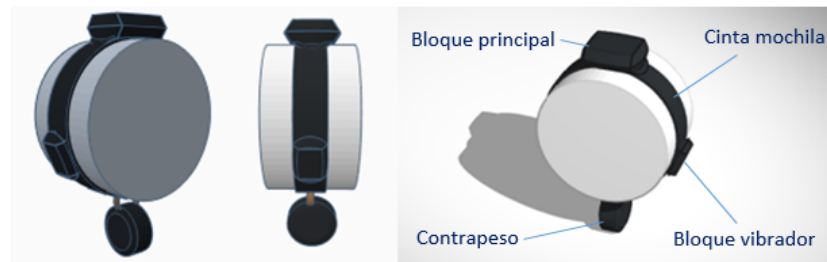


Figura 4.1: Collar en 3D realizado con Tinkercad.

4.1. Requerimientos

El dispositivo fue concebido con el objetivo de ir sujeto a un animal en todo momento. De esta forma, se puede registrar su ubicación geográfica de manera continua. Por este motivo, fue necesario tener especial cuidado en aspectos tales como el tamaño, peso y método de sujeción para no dañar al animal. Una decisión importante que se tomó en la etapa de diseño fue separar el motor del encapsulado principal para que el mismo estimulara en uno de los laterales del collar (ver figura 4.1). Debido a ésta elección, fue necesario diseñar dos encapsulados, uno para el dispositivo principal (portador del Moteino, GPS, buzzer, MPPT, baterías, panel solar) y otro para el motor. También se diseñó un contrapeso, para lograr que los paneles solares adheridos al collar quedaran siempre orientados hacia el sol. Los requerimientos iniciales se detallan a continuación:

- *Soporte y protección:* todos los encapsulados fabricados deben cumplir con la función de contener y proteger los componentes en su interior, no permitiendo que los mismos se caigan o reciban daños de manera directa.

Capítulo 4. Diseño del encapsulado

- *Hermeticidad*: los componentes electrónicos utilizados no son resistentes al agua, por lo que es necesario que todos los encapsulados sean herméticos, para evitar el ingreso de agua en su interior. Se considerará aceptable que soporten lluvias de mediana intensidad.
- *Tamaño*: el largo medio de los collares utilizados en las primeras pruebas en animales, fue de aprox. 80 cm. Se toma como referencia su diámetro, 25 cm, suponiendo que la forma del cuello se asemeja a una circunferencia. Dado que el cuello del ganado bovino tiene un diámetro horizontal bastante menor que el vertical, se fija como límite del largo del encapsulado principal, el radio de esta circunferencia. Es decir, 12.5 cm. El ancho máximo aceptable será de 10 cm (el doble del ancho de la cinta mochila que se utilizó para confeccionar el collar). La altura máxima aceptable es de 5 cm. Con respecto al encapsulado del motor, se tomaron como referencia las dimensiones del mismo, para estimar un máximo aceptable. Por último, se le asignó al contrapeso las mismas limitaciones en sus dimensiones que el encapsulado principal. Las dimensiones máximas aceptables para cada encapsulado son las siguientes:

Encapsulado Principal: 125x100x50 mm

Encapsulado Motor: 80x60x20 mm

Contrapeso: 125x100x50 mm

- *Peso*: la masa total del dispositivo en su conjunto, incluyendo el collar, PCB, componentes, encapsulados y contrapeso, no debe superar 1 kg.
- *Sujeción*: se deberá contar con un sistema de sujeción que sostenga de forma adecuada todos los encapsulados, y sea comfortable para el animal.
- *Seguridad*: los encapsulados no podrán tener bordes filosos que puedan lastimar al animal ante un eventual choque.

4.2. Método de sujeción

Se fabricó un collar con cinta mochila y hebillas de plástico, que se puede ajustar mediante un velcro, como se puede ver en la figura 4.2. Estos materiales no dañan al animal, y tras las pruebas de campo realizadas, se concluyó que tampoco causan molestias que puedan afectar su comportamiento habitual. La cinta mochila es lo suficientemente flexible para ajustarse al cuello del animal y se puede atravesar con facilidad, lo que simplificó su costura. El rango de ajuste es de 70 cm a 90 cm de largo (rango acorde al ancho promedio de las vacas que se utilizaron en las pruebas de campo).

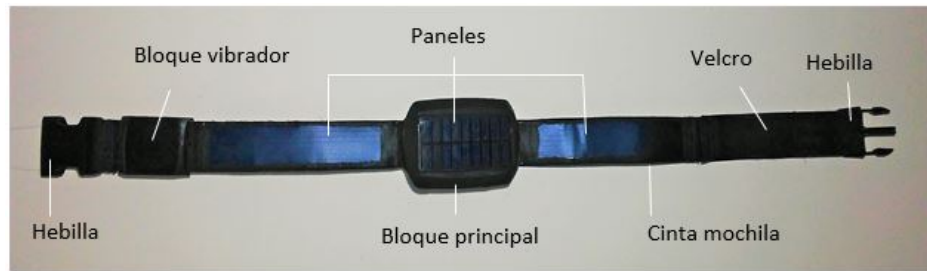


Figura 4.2: Collar de sujeción fabricado.

4.3. Diseño

Para diseñar las piezas en 3D se trabajó con el software Tinkercad de AutoCAD [35]. Se escogió este debido a su acelerada curva de aprendizaje, ya que ningún integrante del equipo contaba con experiencia previa en el rubro. A su vez, Tinkercad se puede utilizar online, por lo que no fue necesario instalar ningún programa con altos requerimientos de sistema.

4.3.1. Encapsulado principal

En el proceso de diseño del encapsulado principal se tuvo en cuenta que el mismo debería contener el dispositivo central de 103 x 64 x 33 mm y un panel solar rígido en su parte superior de 95.8 x 53.8 x 3.2 mm. Para sujetarlo al collar, se colocaron dos empuñaduras a los lados del encapsulado. También se incluyeron orificios en lugares convenientes para poder introducir cables (del panel solar y el motor) e insertar tornillos que pudieran unir ambas partes del encapsulado. En la figura 4.3 se puede observar este diseño desde varios ángulos.



Figura 4.3: Diseño del encapsulado principal desde varias perspectivas. Dimensiones: 122 x 93.6 x 45.4 mm

Capítulo 4. Diseño del encapsulado

4.3.2. Encapsulado del motor

Para realizar este diseño, primero se tuvo que crear un modelo 3D del motor utilizado, para luego usarlo como espacio negativo en su encapsulado. El diseño ondulado de la parte inferior tiene la finalidad de generar más rozamiento contra el animal cuando el mismo se encuentra vibrando. Cuenta con dos empuñaduras a los lados para sujetarlo al collar. Se incluyeron 2 orificios para la salida de los cables del motor y otros cuatro para la inserción de tornillos que pudieran unir ambas partes del encapsulado. En la figura 4.4 se puede observar este diseño desde varios ángulos.

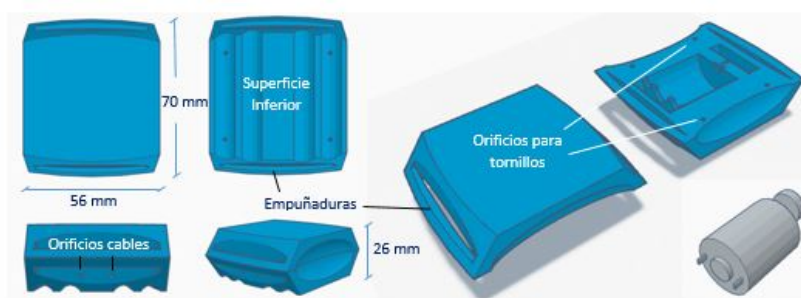


Figura 4.4: Diseño del encapsulado del motor. Dimensiones: 70 x 56 x 26 mm

4.3.3. Contrapeso

Se optó por un diseño cilíndrico, con bordes redondeados, reforzado en el centro para la colocación de un tornillo que pudiera unir ambas partes. Se agregó un orificio para sujetarlo al collar. En la figura 4.5 se puede observar este diseño desde varios ángulos.

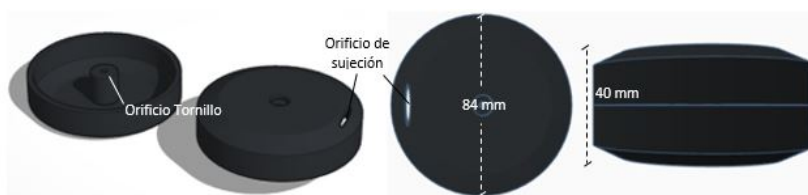


Figura 4.5: Diseño del contrapeso. Dimensiones: 84 x 84 x 40 mm

4.4. Fabricación

Para la fabricación de los encapsulados se decidió utilizar los servicios de impresión 3D de la empresa Fabrix3D. Contaban con 2 tipos de materiales al momento de su fabricación, PLA (ácido poliláctico) y ABC (acrilonitrilo butadieno estireno). Se optó por trabajar con PLA por ser un material resistente, económico y de naturaleza biodegradable (hecho a base de recursos renovables como el almidón de

4.4. Fabricación

maíz o la caña de azúcar). Luego de recibir estas piezas impresas, se realizó un trabajo de pulido con minitorno para mejorar los acabados y permitir una mejor colocación de los componentes en su interior. Se rellenaron los orificios de entrada de los cables con una pistola de silicona caliente para no permitir el ingreso de agua. En el encapsulado principal se reforzaron las empuñaduras con poxipol, ya que podían ser zonas vulnerables. Se cosieron al collar los paneles solares flexibles, las hebillas de plástico y las empuñaduras de ambos encapsulados. En la siguiente figura se puede apreciar el collar finalizado.



(a)



(b)

Figura 4.6: Fotos del prototipo final.



(a)



(b)

Figura 4.7: Fotos del bloque motor y contrapeso.

Capítulo 4. Diseño del encapsulado



Figura 4.8: Fotos del prototipo final en ejemplar bovino ubicado en Facultad de Veterinaria.

Capítulo 5

Pruebas de desempeño y resultados

En este capítulo se abordan las pruebas que se realizaron sobre el NS. Dichas pruebas determinan si se cumple con los objetivos planteados. Finalmente, se analizan los costos asociados al mismo.

5.1. Estudio del consumo energético del dispositivo

Se determinó el consumo del dispositivo midiendo cada uno de sus módulos individualmente para finalmente compararlos con los valores teóricos, detallados en la sección 2.6. También se relevó la corriente que entregan los paneles solares al sistema durante el día y se estimó la cantidad de energía que podrían entregar periódicamente. Luego de obtener estos datos, se recalculó la duración funcional del dispositivo y se plantearon soluciones para mejorar su desempeño.

5.1.1. Medidas de consumo

Se trabajó con un medidor digital de consumo, Otii [36], proporcionado por el IIE (Instituto de Ingeniería Eléctrica) para medir el consumo de corriente individual de los componentes del sistema. Para medir el consumo del motor y la corriente entregada por los paneles solares a la batería, se utilizó un multímetro, modelo DT9208L, debido tanto a las limitaciones de Otii en cuanto a la corriente máxima admisible, como por cuestiones de practicidad al realizar las medidas en el exterior, en el caso de los paneles solares. En la siguiente figura, se muestra un ejemplo de medida de consumo utilizando Otii, en este caso, se observa al dispositivo en modo de transmisión.

Capítulo 5. Pruebas de desempeño y resultados

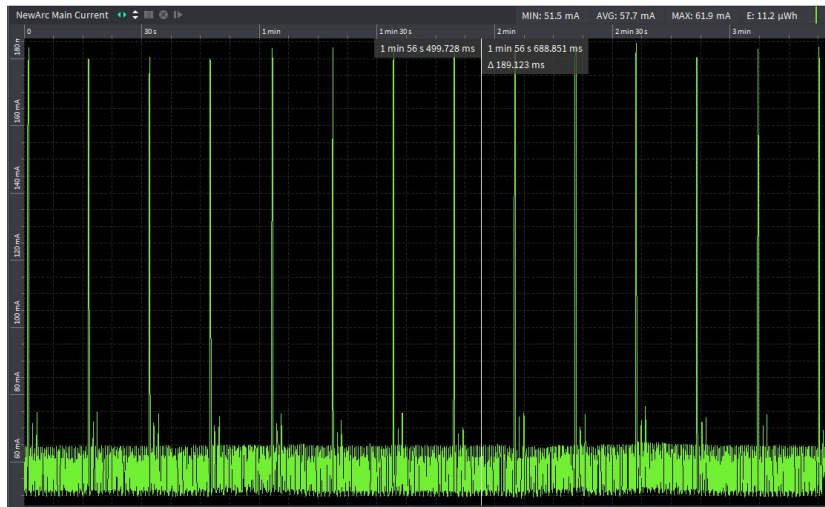


Figura 5.1: Consumo del dispositivo en transmisión. Medición realizada con Oti.

A continuación, se exhibe la tabla de consumos medidos por módulos individuales y su comparación con los valores teóricos.

Módulo	Teórico (mA)	Medido (mA)	Ciclo de trabajo (%)
Moteino (Atmega 1284p)	8	10	100
Moteino Radio Lora Stand by	2	2	100
Moteino Radio Lora TX	120	120	1.6
Moteino Radio Lora RX	10	13	0.16
GPS	27	46	100
Motor	600	540	0.6
Buzzer	30	47.2	2.2
Consumo Total Promedio	64.2 mA		

Tabla 5.1: Consumo de corriente promedio por módulo.

La corriente entregada por los paneles solares en HSP (hora sol pico), se midió y fue de aproximadamente 100 mA, con un voltaje constante de 4.2 V, proporcionado por el módulo de carga TP4056. Como se mencionó en la sección 2.6, se tendrá al menos 3 HSP al día, lo que nos da un total de 1.26 Wh diarios de energía, comparados con los 4.5 Wh que deberían entregar los paneles según los datos del fabricante. Tras estos resultados se realizan las siguientes observaciones:

- Los paneles solares entregan menos energía de lo esperado. Esto se puede deber a la incompatibilidad entre los dos tipos de paneles solares utilizados (rígidos y flexibles) al colocarlos en serie. Se realizaron pruebas colocando 3 paneles RadioShack en paralelo, logrando fácilmente corrientes de hasta 150 mA en HSP. Esta mejora se plantea en la sección 5.1.2.

5.1. Estudio del consumo energético del dispositivo

- El módulo GPS tiene un consumo bastante más elevado que el valor típico proporcionado por el fabricante (27 mA). Tampoco se pudo configurar en sus modos de bajo consumo. Esto posiblemente se deba a que no son módulos originales. Se logró configurar módulos ublox M6, más antiguos que los M8N, llegando a consumos de hasta 5 mA entre actualizaciones de posición, pero finalmente se decidió no modificar el dispositivo debido a la complejidad que presenta este cambio a nivel de software.
- Tanto el Moteino como el motor consumen corrientes aproximadas a los valores teóricos.
- El buzzer presentó un valor de consumo de corriente superior al esperado. Debido a que el mismo se midió con una alimentación de 3.7 V, a diferencia de la señal PWM de 3.3 V de pico, que se utiliza en nuestro dispositivo (generada por el Moteino), se cree que el consumo real es inferior. Futuras medidas son necesarias para mejorar esta estimación.

El consumo real del dispositivo se estima en 64.2 mA, considerando los mismos ciclos de trabajo calculados en la sección 2.6, lo que se traduce en aproximadamente 5.7 Wh. Con este consumo, el dispositivo podría permanecer funcional durante 1.7 días sin recibir energía solar (dado que la batería puede entregar 9.6 Wh). Se estima que podría llegar a 2 días con presencia solar, pero no es capaz de mantenerse funcional de forma indefinida.

5.1.2. Mejoras a futuro

Para mejorar el rendimiento del sistema se proponen las siguientes acciones:

- Configurar en tiempo de ejecución el módulo GPS ublox M8N alternando entre sus modos de bajo consumo, en función de la distancia a la cual se encuentre del perímetro virtual. Tras esta mejora, se estima un consumo promedio del módulo GPS de aproximadamente 10 mA (ver Hoja de datos Ublox M8N).
- Trabajar con modos de bajo consumo en el microcontrolador. Si bien esto no se implementó en el dispositivo final, se realizó una prueba para estimar cuánto podría disminuir su consumo, llegando a 1.4 mA.
- Mejorar el sistema de alimentación con paneles solares, colocando 3 paneles rígidos RadioShack en paralelo, (en sustitución de los paneles flexibles), para que sea capaz de entregar a las baterías 1.89 Wh al día (150 mA durante la HSP, comprobado experimentalmente).

Con estas mejoras se calcula que el dispositivo podría consumir aproximadamente 16 mAh de forma continua (1.42 Wh por día), llegando a funcionar 6.7 días sin recibir energía solar. El mismo trabajaría indefinidamente mientras dicho evento no ocurra (sin considerar aquí, limitaciones relacionadas al tiempo de vida útil

Capítulo 5. Pruebas de desempeño y resultados

de los componentes utilizados). El estudio y mejoría del consumo energético del dispositivo es en sí un área de trabajo que se debe seguir profundizando, ya que tiene un gran potencial de optimización.

5.2. Prueba de alcance de la comunicación dispositivo - gateway

Como ya se ha mencionado en este documento, una de las ventajas de LoRa es su largo alcance, permitiendo cubrir distancias de entre 10 y 15 kilómetros. Al testear el mismo en campo, se logró cubrir una distancia de solamente 1.6 km en la comunicación. Se evaluaron las posibles razones del escaso alcance y se concluyó que el mismo podía deberse a que la prueba no se realizó en las mejores condiciones. Es recomendable realizar estas pruebas con el GW a la mayor altura posible para mantener la línea de vista (sin obstáculos) durante un trayecto mayor; condición que no se pudo cumplir ya que lo más alto que se pudo ubicar el GW fue en el techo de un galpón (5m) por razones de acceso a la energía eléctrica.

Por lo dicho anteriormente se decidió realizar las pruebas de alcance en la ciudad colocando el GW en el último piso de la Facultad de Ingeniería. Luego se desplazó el dispositivo a través de la rambla de Montevideo pasando por los distintos puntos que ofrecían línea de vista (previamente identificados).

Durante todo el trayecto se mantuvo el dispositivo encendido y se observó el estado de la conexión con el GW chequeando el intercambio de mensajes a través de la plataforma Everynet y el monitor serial de Arduino. La conexión se mantuvo desde la Facultad de Ingeniería hasta la rambla de ciudad vieja, donde se comenzaron a perder los mensajes debido a los obstáculos impuestos por los edificios de la ciudad. El último mensaje enviado por el dispositivo y recibido por el GW en este tramo se muestra a continuación:

```
28453598: EV_TXCOMPLETE (includes waiting for RX windows)
A -34.911907 123423 1
O -56.210518 123423
A -34.911907 123423 1
O -56.210518 123423
A -34.911911 123427 1
O -56.210518 123427
A -34.911911 123427 1
O -56.210518 123427
```

Figura 5.2: Mensajes enviados por el dispositivo al GW vistos en el monitor serial.

Los mensajes vistos en la figura 5.2 contienen información de la latitud (A) y longitud (O) las cuales se pueden insertar en Google maps para medir la distancia a la que se está del GW (figura 5.3).

5.2. Prueba de alcance de la comunicación dispositivo - gateway

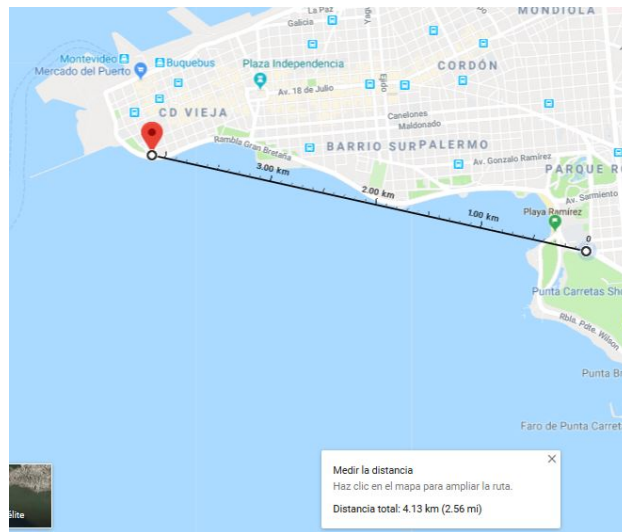


Figura 5.3: Distancia Facultad de Ingeniería - Ciudad Vieja.

La distancia a ese punto es de aproximadamente 4 km, lo que demuestra que claramente la línea de vista era al menos una de las razones por las que se obtuvo una distancia tan reducida en la primera medición de alcance. Al continuar por la rambla no se volvió a retomar la conexión hasta llegar a la Fortaleza del Cerro (siguiente punto de línea de vista).

Una vez allí, se comenzó a visualizar los mensajes recibidos por el GW en Everynet. Dicho mensaje está en base 64 por lo que fue necesario decodificarlos obteniendo O -56.259201 y A -34.889744. Sustituyendo esas coordenadas en Google maps se puede ver que la distancia entre el dispositivo y el GW es de 9 km (Figura 5.4). A pesar de que la línea entre los dos puntos atraviesa gran parte de la ciudad, debido a la altura de la Fortaleza y de la Facultad, se cree que los obstáculos no influyeron en gran medida.

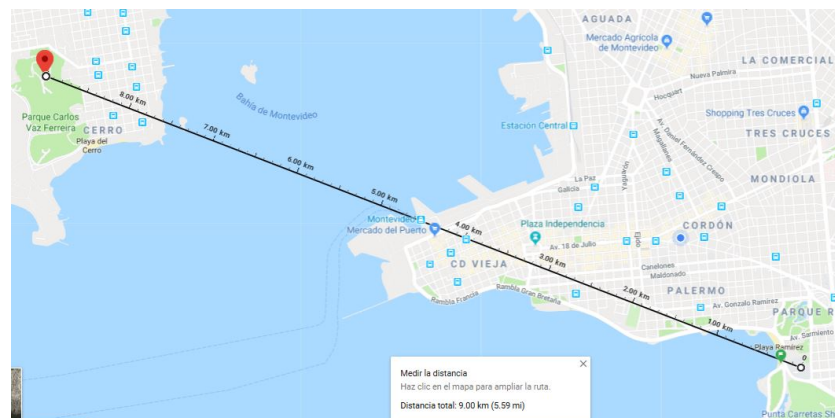


Figura 5.4: Distancia Fortaleza del Cerro - Facultad de Ingeniería.

Se decidió variar parámetros que afectaran la potencia de transmisión para

Capítulo 5. Pruebas de desempeño y resultados

relevante cómo influye ésta en el alcance de la comunicación. Los parámetros de interés en este caso son la potencia de transmisión propiamente dicha y el spreading factor (SF). Este último puede tomar valores de entre 7 y 12 y el mismo indica el tiempo en aire que dura un mensaje enviado. El aumento de un paso en el SF significa que el tiempo en aire será ahora el doble del que tenía, lo que permite cubrir mayores distancias al costo de una mayor energía utilizada y una menor tasa de transferencia de datos.

La librería LMiC (o el módulo RFM95) ignora los cambios realizados en la potencia de transmisión por lo que el único parámetro que se pudo modificar fue el SF, el cual puede tomar valores dentro del rango 7 - 10 (los valores 11 y 12 no son permitidos por la regulación Uruguaya). Se varió el SF comenzando desde el valor 7 y se aumentó de a un paso hasta 10, pudiendo observar en todos los casos la recepción de mensajes por parte del GW a través de Everynet.

Por razones tanto de acceso como de línea de vista, la Fortaleza del Cerro fue el punto más alejado al que se pudo llegar pudiendo verificar que la comunicación NS - GW cuenta con un alcance mínimo de 9 km incluso con la configuración de potencia “menos favorable” en lo que al alcance respecta ($SF = 7$).

5.3. Prueba de variables de interés del PV

Para probar el funcionamiento de las variables de interés del perímetro virtual se hicieron varias pruebas en parque Batlle. Para todas ellas se seleccionaron 4 puntos del parque que podían ser unidos con hilo sin objetos en su camino. Una vez seleccionados, se unió dichos puntos con hilo, y se tomaron las coordenadas de los puntos P1, P2, P3 y P4. Dichas coordenadas se fijaron como perímetro virtual en el NS.

Se obtuvieron los siguientes puntos:

$$P1X = -34,893600; P1Y = -56,155055$$

$$P2X = -34,893461; P2Y = -56,156055$$

$$P3X = -34,894253; P3Y = -56,156052$$

$$P4X = -34,894329; P4Y = -56,155033$$

La vista satelital de dichos puntos se encuentra en la figura 5.5 a continuación

5.3. Prueba de variables de interés del PV

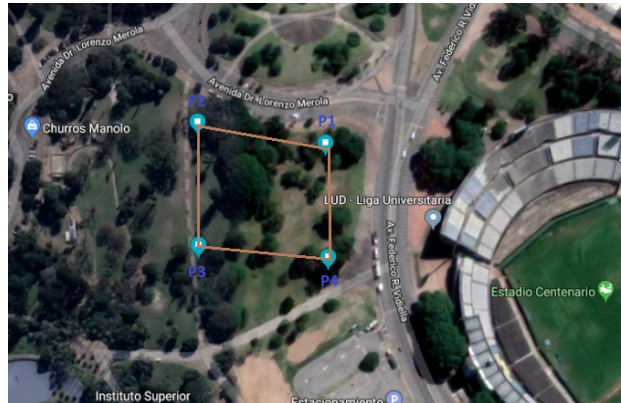


Figura 5.5: Vista satelital de los puntos seleccionados para la prueba de funcionamiento del PV

Al momento de realizar la prueba, contamos con un perímetro virtual, y ese mismo perímetro virtual se encuentra delimitado por hilo. En la figura 5.6 a continuación se muestran las esquinas del perímetro virtual delimitado por hilo.



(a) Esquina 1



(b) Esquina 2



(c) Esquina 3



(d) Esquina 4

Figura 5.6: Esquinas del perímetro delimitado en Parque Batlle

A continuación se describen las distintas pruebas que se realizaron sobre el perímetro virtual antes mencionado.

Capítulo 5. Pruebas de desempeño y resultados

5.3.1. Prueba de zonas

El objetivo de esta prueba es verificar los siguientes puntos:

1. El NS reporta correctamente si está adentro (zonas I, II y III) o afuera (zona IV) del PV.
2. En la zona II, estimula el buzzer.
3. En la zona III, estimula el buzzer y el motor.
4. Una vez que se salió del PV, volver a entrar y verificar que se demora un tiempo igual al valor fijado para REE en volver a estimular.

Para testear lo mencionado al principio de esta sección, se realizarán 8 pruebas. Para las 8 pruebas, se fijaron los siguientes parámetros en el NS:

- PFE=15 (m).
- SFE=5 (m).
- STD=60000 (ms).
- RED=10000 (ms).
- REE=10000 (ms).

En estas 8 pruebas se testea el correcto funcionamiento del dispositivo en las zonas I, II, III y IV (determinadas por PFE y SFE). También se testea REE, verificando el tiempo que demora en volver a estimular el dispositivo una vez que se ingresa desde la zona IV al perímetro virtual.

Las 8 pruebas consisten en tomar datos de la zona I, II, III, IV. Luego se vuelve a la zona III para testear REE (ver 5.7). Estos datos se tomarán en la esquina 1 (prueba E1), en el lateral 1 (prueba LAT1), en la esquina 2 (prueba E2), en el lateral 2 (prueba LAT2) y así sucesivamente hasta llegar a las 8 pruebas. En la figura 5.7 a continuación se muestran las 8 pruebas a realizar.

5.3. Prueba de variables de interés del PV

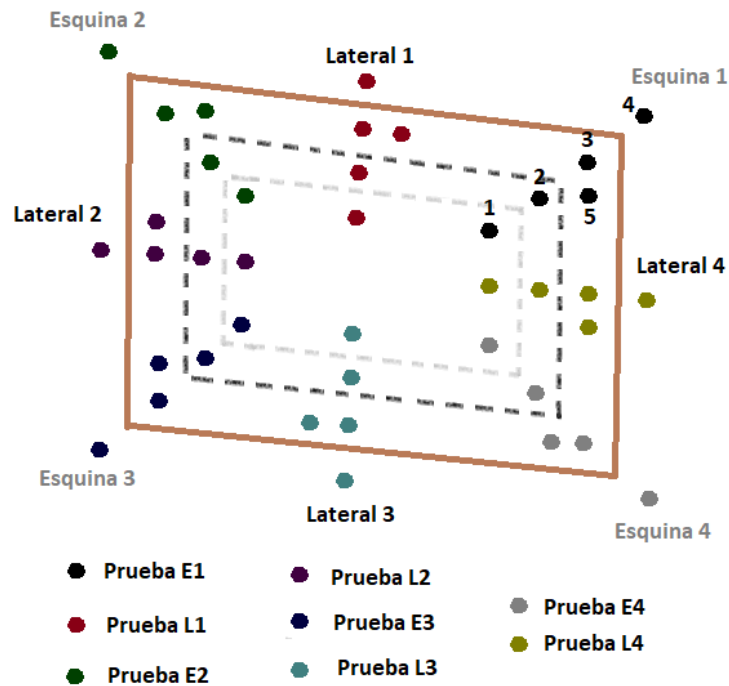


Figura 5.7: Pruebas a realizar.

Para las 8 pruebas se conectó el NS al monitor serial de una computadora. Dicho monitor recibe la latitud, longitud, hora y la variable Z que indica si el dispositivo está dentro o fuera del PV. Se recibe estos parámetros con el protocolo que se definió en la sección 3.3.1. Se colocaron prints también de la distancia del dispositivo al PV, y del tipo de estimulación (se recibe “Buzzer” cuando estimula el Buzzer o “Motor” cuando estimulan ambos).

La información del monitor serial será la que se procesará para analizar los resultados de las pruebas. En el anexo A se encuentran los resultados procesados de las 8 pruebas de funcionamiento del dispositivo (E1, E2, E3, E4, LAT1, LAT2, LAT3, LAT4).

A continuación se analizan los datos relevantes de la prueba E1. No se analizan las otras pruebas porque las conclusiones a las que se llega son idénticas que a las que se llegarán con la prueba E1.

Para la prueba E1, se procesan los datos quitando aquellos datos iniciales que no aportan información (debido a que se está en la zona I, el dispositivo solo envía la posición), y quitando datos repetidos. El resultado de dicho procesamiento se puede ver en el anexo A. En la tabla 5.2 a continuación se muestran los datos más relevantes obtenidos en la prueba E1.

Capítulo 5. Pruebas de desempeño y resultados

A Latitud Hora Z	O Longitud Hora	Distancia (m)	Estimulación
A -34.893749 170937 0	O -56.155220 170937	Distance >18.24	
A -34.893734 170945 0	O -56.155205 170945	Distance >16.58	
A -34.893730 170949 0	O -56.155190 170949	Distance >14.92	Buzzer
A -34.893688 171026 0	O -56.155159 171026	Distance >10.67	Buzzer
A -34.893669 171041 0	O -56.155140 171041	Distance >8.81	Buzzer
A -34.893646 171102 0	O -56.155109 171102	Distance >5.57	Buzzer
A -34.893631 171107 0	O -56.155098 171107	Distance >3.71	Motor
A -34.893627 171118 0	O -56.155083 171118	Distance >2.90	Motor
A -34.893620 171121 0	O -56.155075 171121	Distance >2.07	Motor
A -34.893608 171124 0	O -56.155067 171124	Distance >0.93	Motor
A -34.893600 171129 1	O -56.155060 171129	Distance >0.00	
A -34.893578 171135 1	O -56.155045 171135	Distance >1.24	
A -34.893559 171143 1	O -56.155033 171143	Distance >2.49	
A -34.893551 171155 1	O -56.155025 171155	Distance >3.32	
A -34.893581 171218 1	O -56.155037 171218	Distance >2.07	
A -34.893604 171222 1	O -56.155048 171222	Distance >0.46	
A -34.893623 171226 0	O -56.155056 171226	Distance >0.00	
A -34.893635 171230 0	O -56.155064 171230	Distance >0.83	
A -34.893635 171234 0	O -56.155067 171234	Distance >1.24	
A -34.893650 171238 0	O -56.155083 171238	Distance >2.90	Motor
A -34.893661 171241 0	O -56.155094 171241	Distance >4.14	Motor
A -34.893669 171243 0	O -56.155098 171243	Distance >4.97	Motor

Tabla 5.2: Datos relevantes obtenidos en la prueba E1.

En la sección 3.3.1 se explica cómo se debe interpretar la información de la tabla 5.2. En resumen, los datos que recibimos comienzan con la letra A cuando se trata de la latitud, y comienzan con la letra O cuando se trata de la longitud. Ambos datos se reciben en paquetes distintos, y ambos tienen en común la hora, expresada en formato hhmmss. A su vez, en los paquetes que envían la latitud, se envía a continuación de la hora la variable Z. Dicha variable vale 1 si el dispositivo se encuentra fuera del perímetro virtual, y 0 si se encuentra dentro. La distancia se mide del dispositivo a la recta más cercana del PV. Por lo tanto, será siempre positiva, aunque el dispositivo se encuentre fuera del PV.

De los datos obtenidos en la tabla 5.2 podemos sacar varias conclusiones:

1. Cuando la distancia se encuentra entre 5.57 m y 14.92 m, con el dispositivo dentro del PV, estimula el buzzer.
2. Cuando la distancia es menor a 4.97 m y mayor a 0.93 m, con el dispositivo dentro del PV, estimula el buzzer y el motor.
3. Cuando el NS sale del perímetro virtual (Z=1) cesan las estimulaciones.

5.3. Prueba de variables de interés del PV

4. El NS se vuelve a encontrar dentro del PV a las 17:12:26, y permanece dentro del mismo. A las 17:12:38 el NS vuelve a estimular en la zona III. Dado que los datos obtenidos son discretos, podemos estimar el rango de tiempos dentro del cual se encuentra el tiempo refractario a la entrada. Pero no se puede estimar el valor exacto, dado que no se dispone de la información de en qué momento entra el NS al PV, ni en qué momento se comienza a estimular. Con los datos obtenidos se deduce que el tiempo refractario a la entrada obtenido vale entre 8 s y 16 s¹.

Se destaca que las distancias aumentan con el dispositivo afuera ($Z=1$) porque el dispositivo se aleja del PV. Luego vuelven a disminuir porque se vuelve a acercarse. Para deducir en qué zona se encuentra el dispositivo, se utiliza la variable Z .

Del análisis de los datos presentados se deduce que el comportamiento del dispositivo fue el esperado. Lo mismo puede decirse de las 8 pruebas realizadas (ver anexo A)

5.3.2. Prueba de refractario

El objetivo de esta prueba es verificar los siguientes puntos:

1. El NS cesa la estimulación luego del tiempo fijado en STD.
2. El NS vuelve a estimular luego del refractario fijado en RED.

La prueba consistió en moverse dentro de las zonas II y III, durante un tiempo suficiente para visualizar el refractario y la vuelta a la estimulación. Se conectó el NS a la computadora, y se visualizó la estimulación desde el monitor serial. Para la pruebas se fijaron los siguientes parámetros de interés en el NS:

- PFE=15 (m)
- SFE=5 (m)
- STD=20000 (ms)
- RED=20000 (ms)

Los resultados obtenidos se pueden ver en la tabla 5.3 a continuación.

¹8 s en el caso que el dispositivo entra al PV 17:12:26- ε y se empieza a estimular 17:12:34+ ε . 16 s si el dispositivo entra al PV 17:12:22+ ε y el motor estimula 17:12:38- ε .

Capítulo 5. Pruebas de desempeño y resultados

Latitud,Hora,Z (hhmmss)	Longitud, Hora (hhmmss)	Distancia (m)	Estimulación
A -34.894157 181510 0	O -56.155392 181510	Distance >15.86	
A -34.894176 181514 0	O -56.155403 181514	Distance >13.38	Buzzer
A -34.894230 181524 0	O -56.155415 181524	Distance >7.43	Buzzer
A -34.894260 181530 0	O -56.155422 181530	Distance >4.46	Motor
A -34.894268 181533 0	O -56.155422 181533	Distance >3.47	Motor
A -34.894268 181536 0	O -56.155426 181536	Distance >3.47	
A -34.894268 181555 0	O -56.155426 181555	Distance >3.47	
A -34.894268 181559 0	O -56.155426 181559	Distance >3.47	Motor
A -34.894268 181604 0	O -56.155426 181604	Distance >3.47	Motor

Tabla 5.3: Resultados principales de la prueba para tiempos refractarios.

De analizar los datos de la tabla 5.3 se deduce que el tiempo refractario entre estimulaciones está entre 19 s y 26 s.² No se puede deducir el valor exacto dado que los datos obtenidos son discretos. También se deduce que el tiempo de estimulación está entre 19 s y 26 s.³

Del análisis de los datos presentados se deduce que el comportamiento del dispositivo fue el esperado. Se fijó STD=20000, RED=20000. El valor de las variables STD y RED (20s) es coherente con los tiempos de estimulación y refractarios obtenidos en la prueba.

5.3.3. Prueba de velocidad

El objetivo de esta prueba es estimar el impacto de la velocidad en la estimulación. Si analizamos los primeros 11 datos de la tabla 5.2, vemos que en 107 segundos recorrimos 18,24 metros. Eso nos da una velocidad promedio de 0,612 km/h. Esta velocidad promedio asume que la trayectoria fue perpendicular a la recta más cercana del PV, lo cual no es cierto. Sin embargo, nos da una primera aproximación de a qué velocidad se hizo la prueba.

Se realizarán 2 pruebas, conectando el dispositivo al monitor serial y verificando que el NS estimula correctamente. Las pruebas se realizarán en el lateral 2, y consistirán en ir desde la zona I hasta la zona IV a velocidad intermedia (alrededor de 3,6 km/h) y alta (alrededor de 7,2 km/h). Estimamos estas velocidades utilizando una aplicación para celulares llamada “Velocímetro”.

Los resultados de las pruebas de velocidad intermedia y alta se pueden ver a continuación

²19 s en el caso de que la estimulación cese 18:15:36- ϵ y comience de nuevo 18:15:55+ ϵ . 26 s en el caso de que la estimulación cese 18:15:33+ ϵ y comience de nuevo 18:15:59- ϵ .

³19 s si la estimulación empieza 18:15:14- ϵ y termina 18:15:33+ ϵ . 26 s si la estimulación empieza 18:15:10+ ϵ y termina 18:15:36- ϵ .

5.3. Prueba de variables de interés del PV

Latitud,Hora,Z (hhmmss)	Longitud, Hora (hhmmss)	Distancia (m)	Estimulación
A -34.894138 180947 0	O -56.155334 180947	Distance >18.34	
A -34.894157 181010 0	O -56.155331 181010	Distance >16.35	
A -34.894196 181014 0	O -56.155350 181014	Distance >11.89	Buzzer
A -34.894257 181020 0	O -56.155376 181020	Distance >4.96	Motor
A -34.894287 181023 0	O -56.155388 181023	Distance >1.49	Motor
A -34.894314 181026 1	O -56.155403 181026	Distance >1.49	
A -34.894337 181029 1	O -56.155411 181029	Distance >3.96	

Tabla 5.4: Resultados obtenidos en la prueba de velocidad intermedia.

Latitud,Hora,Z (hhmmss)	Longitud, Hora (hhmmss)	Distancia (m)	Estimulación
A -34.894115 181205 0	O -56.155334 181205	Distance >20.32	
A -34.894123 181224 0	O -56.155331 181224	Distance >19.82	
A -34.894176 181228 0	O -56.155361 181228	Distance >13.88	Buzzer
A -34.894291 181234 0	O -56.155407 181234	Distance >0.99	Motor
A -34.894337 181237 1	O -56.155430 181237	Distance >4.46	
A -34.894360 181239 1	O -56.155441 181239	Distance >6.94	

Tabla 5.5: Resultados obtenidos en la prueba de velocidad alta.

Si tomamos en cuenta todos los datos de la tabla 5.4 y asumimos que se corrió perpendicular al lateral 2, vemos que se recorrieron 22.3 m en 42 s. Si consideramos que se hizo a velocidad constante, entonces $v_{intermedia} = 1,908$ km/h.

Si analizamos lo mismo para la prueba de velocidad alta, obtenemos $v_{alta} = 2,88$ km/h. Estos valores no coinciden con los esperados de 3,6 km/h y 7,2 km/h respectivamente. Se analizó cuál podía ser la causa, y se vio que la aplicación de celular que utilizamos para calcular la velocidad a la que se estaba realizando la prueba tenía una incertidumbre muy grande a velocidades del orden de 3,6 km/h (a posteriori se descubrió que la aplicación estaba pensada para andar en auto). En la sección 6.4, trabajo a futuro, se plantea repetir esta prueba a mayor velocidad, utilizando instrumental adecuado. También se plantea elegir previamente las velocidades a probar teniendo en cuenta qué velocidades pueden ser alcanzadas por ganado vacuno.

De los resultados obtenidos, se deduce que el dispositivo se comportó como se esperaba. El aumento de velocidad tuvo como consecuencia menor gradualidad en las estimulaciones. Como la frecuencia de muestreo del GPS es constante (1 Hz), a mayor velocidad del dispositivo yendo hacia la zona IV, menos datos del GPS en las zonas II y III. Como dichos datos son los que hacen que aumente la frecuencia del pitido del buzzer, disminuye la gradualidad de la estimulación. En la sección 6.4 se desarrolla como posible mejora que la frecuencia de muestreo del GPS dependa de la distancia del dispositivo al perímetro virtual.

Se destaca que cuanto más grandes sean las zonas II y III, menos problemas presenta la velocidad del dispositivo para la gradualidad de la estimulación que recibe.

5.3.4. Pruebas buzzer

Se relevó el volumen emitido por el buzzer, a distintas frecuencias y con distintos niveles de alimentación. Para ello, se utilizó un sonómetro, modelo ET-933. El rango de frecuencias evaluado en este procedimiento fue de 500 Hz a 9000 Hz (con intervalos de 500 Hz).

Se varió el voltaje de entrada del buzzer modificando el ciclo de trabajo del pulso PWM que lo alimenta, a través del Moteino, con un valor de pico de 3.3 V. Este ciclo se representa en el código con una variable de 8 bits, por lo que su máximo valor es de 255, el cual se corresponde con un ciclo de trabajo del 100 %. En este último caso, el moteino alimenta al buzzer con 3.3 V continuos (los valores intermedios son proporcionales a este voltaje, ej: 150 equivale a aproximadamente 2 V de continua).

A continuación, se puede observar una gráfica del volumen del buzzer (dB), en función de la frecuencia (Hz), con distintos ciclos de trabajo en su alimentación (100, 150, 200, 255). Se realizó esta primera medición sin el encapsulado principal.

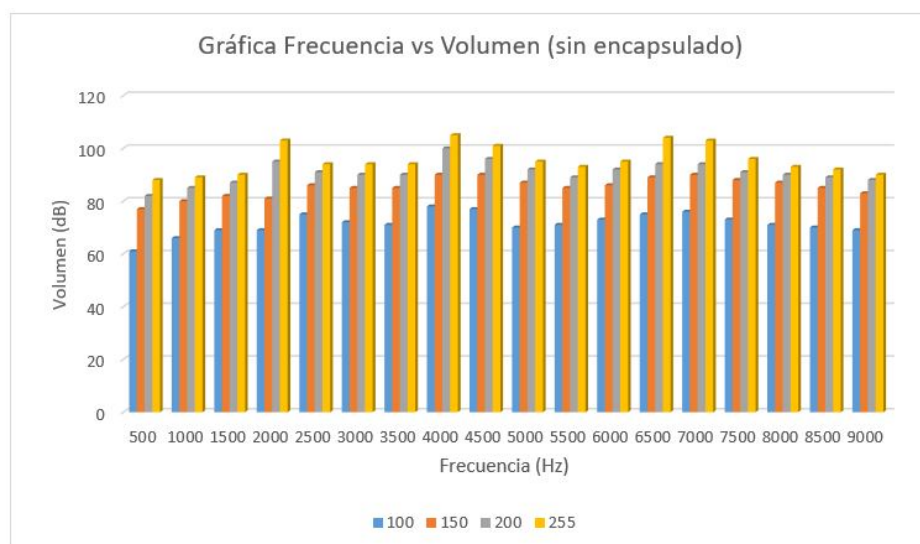


Figura 5.8: Espectro del buzzer sin encapsulado.

Se observan posibles frecuencias de resonancia próximas a 2, 4 y 6.7 kHz. El volumen continúa disminuyendo gradualmente con el aumento de la frecuencia luego de los 9 kHz, por lo que dichos valores no fueron incluidos en la gráfica. Este hecho es un fuerte indicio del ancho de banda del buzzer. El rango de volúmenes alcanzado parece ser lo suficientemente alto como para ser detectado por el ganado, y en varios casos, resulta excesivo. Sin embargo, estas mediciones fueron relevadas sin el encapsulado principal. Debido al esfuerzo realizado para lograr la hermeticidad del mismo, se decidió volver a tomar las medidas para observar cuantitativamente la disminución del volumen.

5.3. Prueba de variables de interés del PV

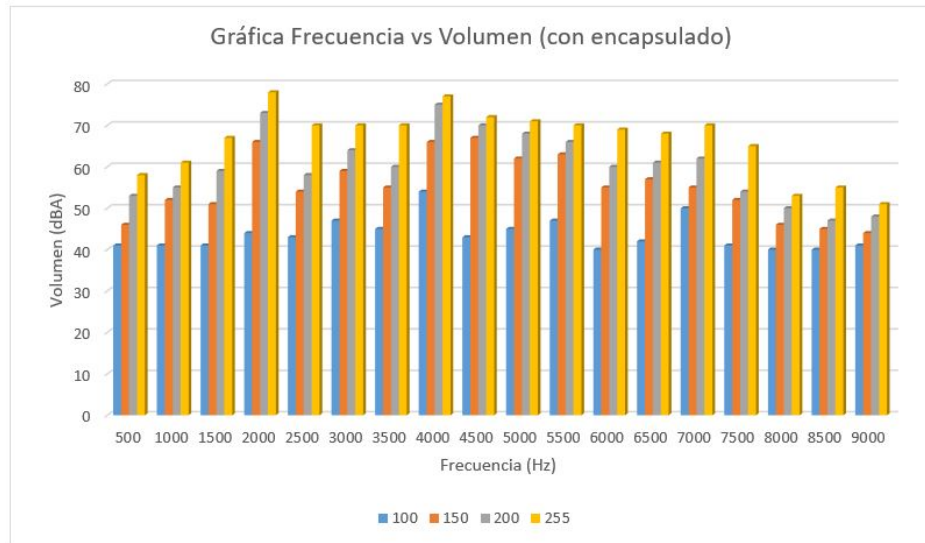


Figura 5.9: Espectro del buzzer con encapsulado.

Se observa una disminución del volumen de aproximadamente un 25 % en todas las mediciones. Aún así, la gran mayoría de los valores resultaron ser audibles para el observador, por lo que se tiene un amplio rango para experimentar. Se recomienda utilizar frecuencias y voltajes que generen volúmenes por encima de los 45 dB.

5.3.5. Pruebas para verificar el resto de las variables

Se realizaron pruebas para testear el correcto comportamiento del resto de las variables de interés del PV.

- Se modificó el valor de la variable $STI=0$. Se acercó al PV, y se verificó que no estimulaba.
- Se fijó $BVE=8$. Se llevó el dispositivo al borde de la zona III, donde el buzzer debía pitar 8 veces en cada estimulación. Se verificó que así fue.
- Se modificó BPD, y se verificó cualitativamente que cada pitido del buzzer duraba más.
- Se entró en la zona III con $MBO=2000$ (2s). Se cronometró el tiempo de vibración del motor, obteniendo un tiempo de vibración del motor de 2,32s. Los 0,32s de diferencia entre el valor teórico y el obtenido se explican por el tiempo que se demoró en reaccionar y detener el cronómetro.
- No se modificó la variable FRE para verificar si efectivamente modifica la frecuencia del sonido emitido por el buzzer, dado que no se tenía a disposición un aparato para medir dicha frecuencia (por ejemplo un metro calibrado de sonido).

Con el conjunto de pruebas descritas en la sección 5.3 se da por funcionales a todas las variables de interés del PV, menos a FRE que no se probó.

5.4. Pruebas en ganado bovino

Se realizaron varias pruebas en ejemplares bovinos con distintas versiones del dispositivo. Las primeras pruebas realizadas tomaron lugar en Casupá, departamento de Florida. En el Anexo D, se describe la prueba de concepto inicial, donde se observó un comportamiento alentador ante los estímulos utilizados. A continuación se describe la última prueba realizada en Facultad de Veterinaria en un ejemplar bovino con el prototipo final de NS, descrito en el capítulo 4.

5.4.1. Prueba en Facultad de Veterinaria

Se realizó una prueba en las instalaciones de Facultad de Veterinaria con el NS. Se evaluó su desempeño en cuanto a la sujeción del collar y la funcionalidad del PV. Para la prueba se utilizó un potrero ubicado frente al edificio hospital del predio de Facultad de Veterinaria. Dicho potrero estaba delimitado por postes y cuerdas. Se utilizaron las coordenadas del potrero físico para fijar el PV. El bovino se encontraba dentro del potrero, sujetado por una cuerda a un árbol, sin pasto para alimentarse en sus alrededores. Se colocó el dispositivo en el bovino y se soltó la cuerda.

Al soltar la cuerda, el bovino se avalanzó sobre el pasto para alimentarse. Durante el transcurso de la prueba, no hubo reacciones significativas a las estimulaciones, aunque sí se observaron algunos movimientos que indican que los mismos eran percibidos por el animal.

En la figura 5.10 a continuación se puede ver el bovino con el NS.



Figura 5.10: Pruebas en bovinos con NS.

Aproximadamente 20 minutos luego de comenzada la prueba, el bovino rompió el potrero físico para continuar pastando fuera del mismo.

5.5. Análisis de costos

El dispositivo se comportó como se esperaba. Sin embargo, el animal no quedó confinado al PV. Una vez afuera del PV, el bovino realizó un movimiento que rompió la sujeción del encapsulado principal al cinturón. Se decidió quitarle el collar para evitar futuras rupturas en el NS.

No se considera que los resultados de esta prueba sean desalentadores para la investigación en confinamiento virtual de bovinos. Las pruebas se realizaron en un bovino de Facultad de Veterinaria, que se encuentra acostumbrado al trato con personas.

De esta prueba se concluye:

- el collar del dispositivo debe rediseñarse para poder resistir movimientos bruscos por parte del animal.
- para futuras pruebas, se deben utilizar bovinos que no se encuentren acostumbrados al trato con humanos y a los ruidos de la ciudad. Esto puede influir en su reacción a los estímulos.

5.5. Análisis de costos

Luego de construir el dispositivo, se realizó un análisis sobre sus costos (excepcuando el costo de montaje y mano de obra), planteando distintas acciones para disminuir los mismos. En el Anexo B se pueden observar con más detalle.

A continuación se mencionan las principales conclusiones tras este estudio:

1. El dispositivo construido en este proyecto tuvo un costo de USD 221.71.
2. Si se tomaran algunas medidas simples como por ejemplo importar los mismos componentes, en vez de hacer compras locales, su costo bajaría a USD 60.88.
3. Importando todos los componentes al por mayor (superior a las 5000 unidades) su costo disminuiría a USD 39.5. El Moteino representa un 80 % del costo total.
4. Sustituyendo el Moteino por un módulo propio con un microcontrolador Atmega 1284p y módulo de comunicación RFM95, se podría reducir el costo a USD 6.7.

Se concluye que tanto los requerimientos aceptables como los ideales son fácilmente alcanzables aplicando esta serie de medidas.

Esta página ha sido intencionalmente dejada en blanco.

Capítulo 6

Conclusiones

En este capítulo se comparan los requerimientos iniciales con los criterios de éxito. A continuación, se detallan los principales problemas encontrados durante el transcurso del proyecto. Finalmente, se plantean los principales aportes del proyecto, y se analiza el trabajo a futuro, en vistas a un dispositivo comercial. Se concluye que se cumplió con la mayoría de los criterios de éxito establecidos y a su vez se presenta una innovadora y llamativa propuesta que atiende a una necesidad comercial del país.

6.1. Criterios de éxito

- **Precisión del sistema:** Durante la realización de las pruebas de la funcionalidad de PV (ver sección 5.3) se observó una precisión en la posición del GPS de aproximadamente 2m. Dicha precisión no sólo es aceptable sino que es mejor que la expuesta en los criterios de éxito ideales.
- **Localización del animal:** La frecuencia mínima a la que puede operar el GPS es de 1 Hz. Debido a que la posición se envía en 2 mensajes (uno para la latitud y otro para la longitud) y a los tiempos de ejecución del software, es posible enviar la posición cada menos de 5 segundos (criterio de éxito ideal).
- **Precio del NS:** El análisis de costos (sección 5.5) del NS arroja un precio de aproximadamente U\$S 40 por dispositivo si se realizan las compras de los componentes al por mayor. En caso de comprar el microcontrolador y módulo LoRa en vez del moteino, se podría reducir a un precio de aproximadamente U\$S 7. Actualmente los precios están dentro de los parámetros aceptables (siempre y cuando se compren componentes al por mayor) mientras que sustituyendo el Moteino por los demás componentes mencionados se logra una reducción sustancial en el costo superando incluso los parámetros ideales. Como se mencionó anteriormente, este análisis de costos no tiene en cuenta las horas hombre, y tampoco el costo de montaje. Se considera que dicho análisis se encuentra fuera del alcance de este proyecto.

Capítulo 6. Conclusiones

- **Interfaz de usuario:** Se cuenta con una interfaz web que permite la configuración del dispositivo de forma remota. A través de esta interfaz se guardan los datos obtenidos del NS en una base de datos presente en un servidor, lo que permite su posterior visualización. Esto hace posible entre otras cosas, poder registrar la historia de la posición de determinado NS por tiempo indeterminado. El manual de uso de dicha interfaz se encuentra en el anexo C
- **Autonomía:** El estudio del consumo energético expuesto en la sección 5.1.1 muestra que el dispositivo cuenta con una autonomía de 2.35 días (aproximadamente 3 días en presencia de luz solar). De esta forma no se cumplió con los parámetros aceptables los cuales eran de 7 días. Por lo tanto a futuro se trabajará en lograr que el dispositivo permanezca operativo de forma indefinida utilizando energía solar.
- **Funcionalidad del sistema:** Las pruebas de funcionalidad del NS (sección 5.3) demostraron que el dispositivo funciona correctamente enviando estímulos cuando es debido, cumpliendo con los parámetros aceptables estipulados en la sección 1.4.

6.2. Principales problemas encontrados

- La falta de precisión de la documentación proporcionada por IBM sobre el uso de la LMIC en lo que refiere a la máxima duración de los eventos encolados fue un problema a la hora de diseñar el software para el PV (descripción en la sección 3.2).
- La potencia nominal de los paneles solares comprados en el exterior difiere de la especificada al momento de realizar la compra. Esto afectó la autonomía del dispositivo derivando en un desempeño inferior al deseado en cuanto a los requerimientos de consumo energético.
- La conexión inalámbrica entre el NS y el GW implica una restricción a la hora de realizar las pruebas en animales, ya que es necesario disponer de buena cobertura de señal durante el transcurso de las mismas. Esto se debe a que el Gateway se conecta con el servidor de EveryNet a través de una SIM card de Movistar, que no accede a Internet si no hay buena señal. Si el GW no accede a Internet, no hay forma de saber qué sucede con el dispositivo de manera remota.
- El Network Server de EveryNet utiliza distintos sitios web para los servicios que provee a América y a Europa. Sin embargo, las URL de los sitios son muy similares, y si se hace la búsqueda por google de “Everynet Network Server” te dirige a la página que provee servicios a Europa. La forma de acceder a la de América es cambiar “https://ns.eu.everynet.io/” por

“https://ns.us.everynet.io/”. Esto causó grandes confusiones en su momento, dado que se podía acceder a la consola de la unión europea con el usuario y contraseña proveídos por Teliot. Sin embargo, una vez en la consola, no llegaban los datos. Cuando se accedió al sitio correcto, tampoco llegaban los datos. Esto llevó a que se hiciera una extensa revisión de código. Finalmente, se dedujo que el problema recaía en la falta de permisos para crear dispositivos. Una vez obtenidos los mismos, logramos visualizar los datos enviados.

- Las pruebas en animales se realizaron en un predio en el departamento de Florida en la localidad de Casupá. Esto implicó mayores costos al realizar las pruebas debido al transporte y también una mayor inversión de tiempo para realizar las mismas.
Además de lo mencionado anteriormente, la realización de las pruebas al aire libre no siempre se pudo dar en tiempo y forma debido a las inclemencias del tiempo, teniendo que reformular el calendario de actividades en varias ocasiones.
- Los costos mencionados en el punto anterior para la realización de pruebas en animales así como el precio de los distintos componentes para la implementación del NS fueron un problema en un comienzo. Por esta razón fue necesaria la búsqueda y obtención de financiación que ayudara a costear los gastos del proyecto.
- El módulo GPS M8N presentó un consumo bastante más elevado que el valor típico proporcionado por el fabricante. Esto derivó en un estudio más profundo sobre sus modos de bajo consumo. Sin embargo, no fue posible configurarlo, a diferencia de otros más antiguos (ublox 6M). Esto posiblemente se deba a que no son módulos originales, pero es necesario seguir recabando información para asegurarlo.

6.3. Principales aportes

El aporte principal de este proyecto fue crear una plataforma de pruebas para la investigación de diversas técnicas de confinamiento virtual de animales, generando una primera experiencia en el área dentro del Instituto de Ingeniería Eléctrica, IIE.

Mediante esta plataforma se pueden modificar varios parámetros de interés: el perímetro virtual, las franjas de acción, la duración de la estimulación, el tiempo refractario luego de un estímulo, el tiempo refractario a la entrada del PV, la frecuencia del sonido del buzzer, el volumen del buzzer, la duración del pitido del buzzer, la cantidad de pitidos máxima del buzzer, y la cantidad de tiempo que estimula el buzzer y el motor. La plataforma permite visualizar de manera remota la ubicación de los dispositivos y acceder a un historial de sus últimos movimientos.

Con este sistema se podrán evaluar distintos esquemas de estimulación, determinando su desempeño en función de la cantidad de tiempo que se logra mantener confinado al animal en el perímetro establecido.

Capítulo 6. Conclusiones

Es de gran interés para el equipo trabajar con estímulos que sean compatibles con el bienestar animal. Por esta razón, el dispositivo diseñado cuenta con estímulos sonoros y táctiles (vibración), lo que lo distingue de otras soluciones.

6.4. Trabajo a futuro

En todo lo que refiere al confinamiento virtual de ganado, hay mucho para hacer en materia de investigación interdisciplinaria, se mencionan a continuación algunos de los principales experimentos que podrían plantearse:

- En los próximos meses se estudiará su desempeño dentro del marco del proyecto ANII (ver sección 1.6), utilizando entre 8 y 12 collares en ganado bovino, conformando así un primer acercamiento en la materia a nivel nacional y abriendo camino hacia futuras investigaciones. Si se logra encontrar un método efectivo de confinamiento virtual para ganado, son muchos los beneficios que podrán obtener los productores rurales. Entre ellos se destacan el pastoreo de precisión, la disminución tanto de la cantidad de alambrados internos como de la inversión en mantenimiento y la posibilidad de transportar el ganado de manera remota modificando gradualmente su perímetro virtual.
- Realizar pruebas con animales que tengan y animales que no tengan el dispositivo, analizando el cambio de comportamiento que implica el uso del dispositivo. Las mismas deberán validarse por veterinarios.
- Determinar el nivel de estrés que le genera al animal el uso del NS. Para el caso de ganado bovino, investigar la existencia de repercusiones negativas del dispositivo en la producción de carne y leche.
- Analizar si los estímulos más efectivos realmente son el buzzer y un motor, o si existen otros que sean más efectivos para confinamiento animal.

Mejoras de funcionamiento

Se proponen a continuación posibles mejoras del funcionamiento del dispositivo:

1. En lo que refiere al confinamiento virtual, el diagrama de estados presentado en la figura 3.10 no genera estímulos para que el animal vuelva al PV. El comportamiento del dispositivo en la zona IV debe ser estudiado por un equipo multidisciplinario para evaluar cómo se puede generar con el buzzer y el motor un patrón de estímulos que hagan que el animal vuelva a la zona I. En caso de no ser posible, una alternativa es enviar una alerta para que se vaya a buscar al animal y se lo ingrese en la zona I.
2. Determinar una forma de emitir una alarma con el fin de alertar al dueño del animal en caso de que se le quite el collar al mismo. Existen muchas formas posibles de implementar esto, una de ellas es colocando un sensor de

6.4. Trabajo a futuro

proximidad en el broche del cinturón. De este modo, se puede detectar el cambio de estado, y reportarlo al dueño del animal.

3. Reportar en caso de que se corte el cinturón. Para eso, se puede utilizar software para testear cada determinado tiempo la continuidad de los cables que unen al encapsulado del motor con el encapsulado principal. De este modo, si alguien rompe el cinturón, se reporta al dueño del animal.
4. Reportar en caso de que el dispositivo salga del predio permitido (por ejemplo el campo de un productor). Para ello, se puede definir una quinta zona. En caso de que el dispositivo salga de la zona V, se reporta que los animales están saliendo del predio permitido.
5. Variar la frecuencia de muestreo del GPS en función de la distancia al perímetro virtual. De esta manera, se ahorra energía en caso de estar en la zona I. Y en caso de estar en las zonas II y III, una mayor frecuencia de muestreo del GPS hace posible una respuesta más gradual de los estímulos. Es esencial que esto suceda para eventualmente adiestrar al animal, para que sepa hacia donde volver (hacia la zona I, las estimulaciones bajan en graduación, hacia la zona IV, aumentan).
6. Determinar, parametrizar y hacer configurables más variables de interés del perímetro virtual, como puede ser MBF, o la potencia del motor.

Consumo y autonomía

En la sección 5.1.2 se proponen una serie de acciones para mejorar la autonomía de nuestro dispositivo. Se cree que con estas medidas se puede llegar a los requerimientos ideales planteados inicialmente, logrando un sistema que funcione indefinidamente con energía solar. Sin embargo, se cree que esta es un área de trabajo en la que todavía hay mucho por profundizar.

Costos del dispositivo

El 80 % del precio del NS es debido al Moteino. Una forma posible de proseguir es comprar el microcontrolador ATmega1284P y el RFM95 por separado. Otra forma de proseguir es buscar un hardware de menor costo, que sea compatible con el software desarrollado.

En lo que refiere al Gateway, se puede hacer uno con una raspberry pi, disminuyendo sustancialmente el costo del mismo. De todos modos, hay que evaluar el costo-beneficio, dado que cambiar el Gateway por uno propio implica también levantar un servicio similar al de EveryNet para subir los datos.

Capítulo 6. Conclusiones

Usos alternativos

El diseño de un dispositivo económico, capaz de localizar a un animal y mantenerlo dentro de un predio establecido a través de una plataforma web, podría ser útil también para las siguientes aplicaciones:

- Control de mascotas: es de interés para los dueños de mascotas intentar que el animal no se escape de su hogar. De contar con este dispositivo, podría no sólo tener un mecanismo de contención, sino también recibir alertas en caso de que el mismo logre escaparse de todas formas o presente un patrón de comportamiento inusual, lo cual podría ser un indicio de algún problema de salud.
- Perímetro virtual en plazas públicas: Si se colocara un gateway en las plazas públicas, se podría utilizar el mismo concepto de perímetro virtual para evitar que las mascotas abandonen la plaza, lo que evitaría posibles accidentes con el tránsito del entorno.
- Control de animales exóticos, de exhibición o en peligro de extinción: Esta clase de animales requiere un cuidado y seguimiento especial. Por ese motivo, puede ser útil que porten un dispositivo de monitoreo como el propuesto en este proyecto.

Apéndice A

Resultados de pruebas de perímetro virtual

En este anexo se muestran los resultados de las pruebas realizadas en parque Batlle para validar el funcionamiento de las variables de interés del perímetro virtual.

Los resultados que se muestran a continuación fueron procesados. El procesamiento consistió en quitar los datos duplicados y los datos iniciales del GPS (que no aportan información relevante). Luego, se los ordenó en tablas para que sean más visibles que en el monitor serial.

Apéndice A. Resultados de pruebas de perímetro virtual

Prueba E1:

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.893887 170804 0	O -56.155384 170804	Distance >35.72	
A -34.893887 170823 0	O -56.155380 170823	Distance >35.72	
A -34.893879 170827 0	O -56.155365 170827	Distance >34.40	
A -34.893860 170831 0	O -56.155342 170831	Distance >31.91	
A -34.893852 170835 0	O -56.155331 170835	Distance >30.67	
A -34.893848 170839 0	O -56.155331 170839	Distance >30.61	
A -34.893845 170846 0	O -56.155331 170846	Distance >30.15	
A -34.893837 170854 0	O -56.155323 170854	Distance >29.69	
A -34.893822 170858 0	O -56.155296 170858	Distance >26.94	
A -34.893806 170902 0	O -56.155273 170902	Distance >24.45	
A -34.893803 170910 0	O -56.155277 170910	Distance >24.87	
A -34.893799 170913 0	O -56.155281 170913	Distance >24.58	
A -34.893795 170917 0	O -56.155277 170917	Distance >24.12	
A -34.893772 170921 0	O -56.155254 170921	Distance >21.34	
A -34.893753 170925 0	O -56.155235 170925	Distance >19.02	
A -34.893742 170929 0	O -56.155224 170929	Distance >17.63	
A -34.893745 170933 0	O -56.155220 170933	Distance >18.09	
A -34.893749 170937 0	O -56.155220 170937	Distance >18.24	
A -34.893734 170945 0	O -56.155205 170945	Distance >16.58	
A -34.893730 170949 0	O -56.155190 170949	Distance >14.92	Buzzer
A -34.893723 170954 0	O -56.155178 170954	Distance >13.68	Buzzer
A -34.893726 170959 0	O -56.155182 170959	Distance >14.09	Buzzer
A -34.893723 171005 0	O -56.155182 171005	Distance >14.09	Buzzer
A -34.893723 171010 0	O -56.155178 171010	Distance >13.68	Buzzer
A -34.893703 171015 0	O -56.155167 171015	Distance >12.43	Buzzer
A -34.893688 171020 0	O -56.155159 171020	Distance >10.67	Buzzer
A -34.893688 171031 0	O -56.155155 171031	Distance >10.67	Buzzer
A -34.893673 171036 0	O -56.155144 171036	Distance >8.81	Buzzer
A -34.893669 171041 0	O -56.155140 171041	Distance >8.81	Buzzer
A -34.893665 171047 0	O -56.155140 171047	Distance >8.35	Buzzer
A -34.893658 171051 0	O -56.155132 171051	Distance >6.96	Buzzer
A -34.893646 171056 0	O -56.155121 171056	Distance >6.03	Buzzer
A -34.893646 171102 0	O -56.155109 171102	Distance >5.57	Buzzer
A -34.893631 171107 0	O -56.155098 171107	Distance >3.71	Motor

Tabla A.1: Datos obtenidos en la prueba E1.

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.893627 171110 0	O -56.155094 171110	Distance >3.25	Motor
A -34.893631 171113 0	O -56.155094 171113	Distance >3.71	Motor
A -34.893627 171115 0	O -56.155090 171115	Distance >3.25	Motor
A -34.893627 171118 0	O -56.155083 171118	Distance >2.90	Motor
A -34.893620 171121 0	O -56.155075 171121	Distance >2.07	Motor
A -34.893608 171124 0	O -56.155067 171124	Distance >0.93	Motor
A -34.893600 171129 1	O -56.155060 171129	Distance >0.00	
A -34.893589 171131 1	O -56.155056 171131	Distance >0.00	
A -34.893578 171135 1	O -56.155045 171135	Distance >1.24	
A -34.893570 171139 1	O -56.155041 171139	Distance >1.66	
A -34.893559 171143 1	O -56.155033 171143	Distance >2.49	
A -34.893555 171147 1	O -56.155025 171147	Distance >3.32	
A -34.893551 171151 1	O -56.155025 171151	Distance >3.32	
A -34.893551 171159 1	O -56.155022 171159	Distance >3.73	
A -34.893555 171206 1	O -56.155022 171206	Distance >3.73	
A -34.893566 171214 1	O -56.155025 171214	Distance >3.32	
A -34.893581 171218 1	O -56.155037 171218	Distance >2.07	
A -34.893604 171222 1	O -56.155048 171222	Distance >0.46	
A -34.893623 171226 0	O -56.155056 171226	Distance >0.00	
A -34.893635 171230 0	O -56.155064 171230	Distance >0.83	
A -34.893635 171234 0	O -56.155067 171234	Distance >1.24	
A -34.893650 171238 0	O -56.155083 171238	Distance >2.90	Motor
A -34.893661 171241 0	O -56.155094 171241	Distance >4.14	Motor
A -34.893669 171243 0	O -56.155098 171243	Distance >4.97	Motor
A -34.893665 171246 0	O -56.155098 171246	Distance >4.56	Motor
A -34.893665 171252 0	O -56.155098 171252	Distance >4.56	Motor
A -34.893673 171257 0	O -56.155098 171257	Distance >4.97	Motor

Tabla A.2: Continúa datos obtenidos en la prueba E1.

Apéndice A. Resultados de pruebas de perímetro virtual

Prueba de refractarios

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.894135 181502 0	O -56.155384 181502	Distance >18.34	
A -34.894138 181506 0	O -56.155384 181506	Distance >17.84	
A -34.894157 181510 0	O -56.155392 181510	Distance >15.86	
A -34.894176 181514 0	O -56.155403 181514	Distance >13.38	Buzzer
A -34.894203 181519 0	O -56.155411 181519	Distance >10.41	Buzzer
A -34.894230 181524 0	O -56.155415 181524	Distance >7.43	Buzzer
A -34.894260 181530 0	O -56.155422 181530	Distance >4.46	Motor
A -34.894268 181533 0	O -56.155422 181533	Distance >3.47	Motor
A -34.894268 181536 0	O -56.155426 181536	Distance >3.47	
A -34.894272 181543 0	O -56.155426 181543	Distance >2.97	
A -34.894268 181547 0	O -56.155426 181547	Distance >3.47	
A -34.894268 181551 0	O -56.155426 181551	Distance >3.47	
A -34.894268 181555 0	O -56.155426 181555	Distance >3.47	
A -34.894268 181559 0	O -56.155426 181559	Distance >3.47	Motor
A -34.894268 181604 0	O -56.155426 181604	Distance >3.47	Motor

Tabla A.3: Datos obtenidos en la prueba de refractarios.

Prueba de velocidad intermedia

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.894138 180947 0	O -56.155334 180947	Distance >18.34	
A -34.894131 180951 0	O -56.155319 180951	Distance >18.83	
A -34.894135 180955 0	O -56.155315 180955	Distance >18.83	
A -34.894138 180959 0	O -56.155315 180959	Distance >18.34	
A -34.894138 181002 0	O -56.155319 181002	Distance >18.34	
A -34.894138 181006 0	O -56.155319 181006	Distance >18.34	
A -34.894157 181010 0	O -56.155331 181010	Distance >16.35	
A -34.894196 181014 0	O -56.155350 181014	Distance >11.89	Buzzer
A -34.894257 181020 0	O -56.155376 181020	Distance >4.96	Motor
A -34.894287 181023 0	O -56.155388 181023	Distance >1.49	Motor
A -34.894314 181026 1	O -56.155403 181026	Distance >1.49	
A -34.894337 181029 1	O -56.155411 181029	Distance >3.96	
A -34.894337 181032 1	O -56.155407 181032	Distance >3.96	

Tabla A.4: Datos obtenidos en la segunda prueba de velocidad.

Prueba de velocidad alta

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.894115 181205 0	O -56.155334 181205	Distance >20.32	
A -34.894115 181209 0	O -56.155334 181209	Distance >20.32	
A -34.894115 181212 0	O -56.155338 181212	Distance >20.32	
A -34.894119 181216 0	O -56.155334 181216	Distance >20.32	
A -34.894119 181220 0	O -56.155334 181220	Distance >20.32	
A -34.894123 181224 0	O -56.155331 181224	Distance >19.82	
A -34.894176 181228 0	O -56.155361 181228	Distance >13.88	Buzzer
A -34.894291 181234 0	O -56.155407 181234	Distance >0.99	Motor
A -34.894337 181237 1	O -56.155430 181237	Distance >4.46	
A -34.894360 181239 1	O -56.155441 181239	Distance >6.94	
A -34.894360 181243 1	O -56.155445 181243	Distance >6.94	

Tabla A.5: Datos obtenidos en la tercera prueba de velocidad.

Apéndice A. Resultados de pruebas de perímetro virtual

Resultados prueba E2

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.893700 175334 0	O -56.155788 175334	Distance >21.80	
A -34.893688 175338 0	O -56.155804 175338	Distance >20.41	
A -34.893665 175342 0	O -56.155827 175342	Distance >18.55	
A -34.893650 175346 0	O -56.155846 175346	Distance >17.16	
A -34.893639 175350 0	O -56.155857 175350	Distance >16.23	
A -34.893635 175354 0	O -56.155857 175354	Distance >15.77	
A -34.893631 175357 0	O -56.155861 175357	Distance >15.31	
A -34.893631 175401 0	O -56.155861 175401	Distance >15.31	
A -34.893627 175405 0	O -56.155865 175405	Distance >14.84	Buzzer
A -34.893616 175411 0	O -56.155884 175411	Distance >14.38	Buzzer
A -34.893593 175416 0	O -56.155910 175416	Distance >12.06	Buzzer
A -34.893578 175421 0	O -56.155937 175421	Distance >10.67	Buzzer
A -34.893562 175426 0	O -56.155952 175426	Distance >9.28	Buzzer
A -34.893559 175427 0	O -56.155960 175427	Distance >8.81	Buzzer
A -34.893547 175432 0	O -56.155979 175432	Distance >7.88	Buzzer
A -34.893536 175437 0	O -56.155998 175437	Distance >5.80	Buzzer
A -34.893528 175442 0	O -56.156025 175442	Distance >2.90	Motor
A -34.893520 175444 0	O -56.156033 175444	Distance >2.07	Motor
A -34.893505 175447 0	O -56.156052 175447	Distance >0.00	Motor
A -34.893490 175450 1	O -56.156063 175450	Distance >1.24	
A -34.893482 175452 1	O -56.156075 175452	Distance >2.49	
A -34.893471 175456 1	O -56.156078 175456	Distance >1.39	
A -34.893459 175500 1	O -56.156094 175500	Distance >0.46	
A -34.893463 175504 1	O -56.156086 175504	Distance >0.93	
A -34.893459 175508 1	O -56.156094 175508	Distance >0.46	
A -34.893444 175512 1	O -56.156120 175512	Distance >0.93	
A -34.893425 175516 1	O -56.156147 175516	Distance >2.32	
A -34.893406 175520 1	O -56.156170 175520	Distance >4.17	
A -34.893398 175524 1	O -56.156185 175524	Distance >4.64	
A -34.893398 175528 1	O -56.156174 175528	Distance >5.10	
A -34.893410 175532 1	O -56.156162 175532	Distance >3.71	
A -34.893425 175536 1	O -56.156124 175536	Distance >2.78	
A -34.893448 175540 1	O -56.156090 175540	Distance >0.93	
A -34.893486 175548 0	O -56.156040 175548	Distance >1.24	
A -34.893509 175552 0	O -56.156021 175552	Distance >3.32	
A -34.893539 175559 0	O -56.155964 175559	Distance >6.96	Buzzer
A -34.893559 175604 0	O -56.155930 175604	Distance >8.81	Buzzer

Tabla A.6: Resultados obtenidos para la prueba E2.

Resultados prueba E3

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.894009 174136 0	O -56.155888 174136	Distance >17.83	
A -34.894012 174140 0	O -56.155891 174140	Distance >17.41	
A -34.894020 174144 0	O -56.155899 174144	Distance >16.59	
A -34.894032 174148 0	O -56.155907 174148	Distance >15.76	
A -34.894039 174152 0	O -56.155914 174152	Distance >14.93	Buzzer
A -34.894039 174203 0	O -56.155918 174203	Distance >14.51	Buzzer
A -34.894051 174208 0	O -56.155926 174208	Distance >13.68	Buzzer
A -34.894070 174213 0	O -56.155930 174213	Distance >13.27	Buzzer
A -34.894093 174219 0	O -56.155937 174219	Distance >12.44	Buzzer
A -34.894112 174224 0	O -56.155949 174224	Distance >11.20	Buzzer
A -34.894135 174229 0	O -56.155964 174229	Distance >9.54	Buzzer
A -34.894154 174234 0	O -56.155979 174234	Distance >7.88	Buzzer
A -34.894173 174240 0	O -56.155998 174240	Distance >5.80	Buzzer
A -34.894192 174245 0	O -56.156017 174245	Distance >3.73	Motor
A -34.894199 174248 0	O -56.156025 174248	Distance >2.90	Motor
A -34.894218 174256 0	O -56.156040 174256	Distance >1.24	Motor
A -34.894230 174259 0	O -56.156044 174259	Distance >0.83	Motor
A -34.894238 174301 0	O -56.156048 174301	Distance >0.41	Motor
A -34.894249 174304 1	O -56.156055 174304	Distance >0.41	
A -34.894260 174306 1	O -56.156055 174306	Distance >0.83	
A -34.894268 174310 1	O -56.156059 174310	Distance >1.24	
A -34.894268 174314 1	O -56.156063 174314	Distance >1.66	
A -34.894264 174318 1	O -56.156063 174318	Distance >1.49	
A -34.894276 174322 1	O -56.156075 174322	Distance >2.90	
A -34.894291 174326 1	O -56.156094 174326	Distance >4.46	
A -34.894310 174330 1	O -56.156097 174330	Distance >5.39	
A -34.894329 174334 1	O -56.156094 174334	Distance >4.98	
A -34.894306 174338 1	O -56.156097 174338	Distance >5.39	
A -34.894279 174342 1	O -56.156082 174342	Distance >3.47	
A -34.894264 174346 1	O -56.156067 174346	Distance >1.49	
A -34.894249 174350 1	O -56.156055 174350	Distance >0.41	
A -34.894218 174357 0	O -56.156040 174357	Distance >1.24	
A -34.894207 174401 0	O -56.156025 174401	Distance >2.90	
A -34.894184 174405 0	O -56.156025 174405	Distance >2.90	
A -34.894161 174409 0	O -56.156025 174409	Distance >2.90	
A -34.894146 174411 0	O -56.156025 174411	Distance >2.90	Motor
A -34.894135 174414 0	O -56.156002 174414	Distance >5.39	Buzzer

Tabla A.7: Resultados obtenidos en la prueba E3.

Apéndice A. Resultados de pruebas de perímetro virtual

Resultados prueba E4

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.894104 172529 0	O -56.155334 172529	Distance >21.80	
A -34.894104 172533 0	O -56.155334 172533	Distance >21.80	
A -34.894108 172537 0	O -56.155334 172537	Distance >21.31	
A -34.894108 172541 0	O -56.155331 172541	Distance >21.31	
A -34.894108 172544 0	O -56.155331 172544	Distance >21.31	
A -34.894108 172548 0	O -56.155331 172548	Distance >21.31	
A -34.894112 172552 0	O -56.155323 172552	Distance >21.31	
A -34.894127 172556 0	O -56.155300 172556	Distance >19.82	
A -34.894131 172600 0	O -56.155293 172600	Distance >19.33	
A -34.894138 172604 0	O -56.155281 172604	Distance >18.34	
A -34.894146 172608 0	O -56.155270 172608	Distance >17.84	
A -34.894146 172612 0	O -56.155270 172612	Distance >17.84	
A -34.894154 172616 0	O -56.155254 172616	Distance >16.85	
A -34.894157 172620 0	O -56.155247 172620	Distance >16.85	
A -34.894150 172627 0	O -56.155251 172627	Distance >17.34	
A -34.894154 172631 0	O -56.155228 172631	Distance >17.34	
A -34.894165 172635 0	O -56.155212 172635	Distance >16.35	
A -34.894169 172639 0	O -56.155201 172639	Distance >15.86	
A -34.894173 172643 0	O -56.155193 172643	Distance >15.36	
A -34.894173 172647 0	O -56.155197 172647	Distance >15.36	
A -34.894184 172651 0	O -56.155182 172651	Distance >14.37	Buzzer
A -34.894188 172656 0	O -56.155167 172656	Distance >13.88	Buzzer
A -34.894196 172702 0	O -56.155155 172702	Distance >12.85	Buzzer
A -34.894199 172707 0	O -56.155148 172707	Distance >12.02	Buzzer
A -34.894207 172712 0	O -56.155132 172712	Distance >10.36	Buzzer
A -34.894215 172717 0	O -56.155132 172717	Distance >10.36	Buzzer
A -34.894215 172723 0	O -56.155140 172723	Distance >11.19	Buzzer
A -34.894215 172728 0	O -56.155140 172728	Distance >11.19	Buzzer
A -34.894222 172733 0	O -56.155125 172733	Distance >9.53	Buzzer
A -34.894234 172738 0	O -56.155102 172738	Distance >7.05	Buzzer
A -34.894249 172744 0	O -56.155083 172744	Distance >4.97	Motor
A -34.894257 172747 0	O -56.155075 172747	Distance >4.14	Motor
A -34.894260 172750 0	O -56.155079 172750	Distance >4.56	Motor
A -34.894264 172755 0	O -56.155079 172755	Distance >4.56	Motor
A -34.894260 172758 0	O -56.155087 172758	Distance >5.39	Buzzer

Tabla A.8: Resultados obtenidos en la prueba E4.

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.894260 172803 0	O -56.155090 172803	Distance >5.80	Buzzer
A -34.894260 172808 0	O -56.155087 172808	Distance >5.39	Buzzer
A -34.894268 172814 0	O -56.155075 172814	Distance >4.14	Motor
A -34.894272 172817 0	O -56.155071 172817	Distance >3.73	Motor
A -34.894276 172820 0	O -56.155060 172820	Distance >2.49	Motor
A -34.894287 172823 0	O -56.155052 172823	Distance >2.07	Motor
A -34.894283 172825 0	O -56.155045 172825	Distance >1.24	Motor
A -34.894287 172828 0	O -56.155041 172828	Distance >0.83	Motor
A -34.894287 172831 0	O -56.155041 172831	Distance >0.83	Motor
A -34.894291 172834 1	O -56.155033 172834	Distance >0.00	
A -34.894291 172837 1	O -56.155033 172837	Distance >0.00	
A -34.894291 172841 1	O -56.155033 172841	Distance >0.00	
A -34.894295 172845 1	O -56.155018 172845	Distance >1.66	
A -34.894291 172849 1	O -56.155018 172849	Distance >1.66	
A -34.894287 172853 1	O -56.155025 172853	Distance >0.83	
A -34.894291 172857 1	O -56.155025 172857	Distance >0.83	
A -34.894287 172904 1	O -56.155029 172904	Distance >0.41	
A -34.894279 172908 1	O -56.155033 172908	Distance >0.00	
A -34.894283 172912 1	O -56.154995 172912	Distance >4.14	
A -34.894295 172916 1	O -56.154953 172916	Distance >4.46	
A -34.894302 172920 1	O -56.154942 172920	Distance >3.47	
A -34.894314 172924 1	O -56.154930 172924	Distance >2.48	
A -34.894329 172928 1	O -56.154911 172928	Distance >0.99	
A -34.894325 172932 1	O -56.154907 172932	Distance >1.49	
A -34.894333 172936 1	O -56.154907 172936	Distance >0.50	
A -34.894337 172940 1	O -56.154907 172940	Distance >0.00	
A -34.894341 172944 1	O -56.154903 172944	Distance >0.00	
A -34.894341 172948 1	O -56.154907 172948	Distance >0.50	
A -34.894325 172952 1	O -56.154930 172952	Distance >0.99	
A -34.894306 172956 1	O -56.154961 172956	Distance >2.97	
A -34.894291 173000 1	O -56.154991 173000	Distance >4.46	
A -34.894283 173003 1	O -56.155010 173003	Distance >2.49	
A -34.894272 173007 0	O -56.155041 173007	Distance >0.41	
A -34.894264 173011 0	O -56.155064 173011	Distance >2.90	
A -34.894268 173015 0	O -56.155071 173015	Distance >3.73	
A -34.894264 173019 0	O -56.155071 173019	Distance >3.73	
A -34.894264 173021 0	O -56.155071 173021	Distance >3.73	
A -34.894260 173024 0	O -56.155075 173024	Distance >4.14	Motor
A -34.894260 173027 0	O -56.155083 173027	Distance >4.97	Motor
A -34.894260 173032 0	O -56.155083 173032	Distance >4.97	Motor
A -34.894257 173035 0	O -56.155087 173035	Distance >5.39	Buzzer
A -34.894264 173040 0	O -56.155083 173040	Distance >4.97	Motor
A -34.894260 173043 0	O -56.155087 173043	Distance >5.39	Buzzer

Tabla A.9: Continúa resultados obtenidos en la prueba E4.

Apéndice A. Resultados de pruebas de perímetro virtual

Resultados prueba LAT1

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.893826 175855 0	O -56.155624 175855	Distance >32.93	
A -34.893787 175903 0	O -56.155628 175903	Distance >28.76	
A -34.893753 175910 0	O -56.155621 175910	Distance >25.05	
A -34.893742 175914 0	O -56.155624 175914	Distance >23.66	
A -34.893726 175918 0	O -56.155632 175918	Distance >22.26	
A -34.893711 175922 0	O -56.155632 175922	Distance >20.41	
A -34.893703 175926 0	O -56.155617 175926	Distance >19.48	
A -34.893692 175930 0	O -56.155605 175930	Distance >18.09	
A -34.893681 175934 0	O -56.155602 175934	Distance >16.70	
A -34.893673 175938 0	O -56.155590 175938	Distance >15.77	
A -34.893650 175945 0	O -56.155582 175945	Distance >13.45	Buzzer
A -34.893639 175950 0	O -56.155567 175950	Distance >11.60	Buzzer
A -34.893623 180001 0	O -56.155567 180001	Distance >10.20	Buzzer
A -34.893608 180006 0	O -56.155563 180006	Distance >8.35	Buzzer
A -34.893600 180011 0	O -56.155556 180011	Distance >7.42	Buzzer
A -34.893578 180017 0	O -56.155540 180017	Distance >4.64	Motor
A -34.893566 180020 0	O -56.155529 180020	Distance >3.25	Motor
A -34.893559 180023 0	O -56.155525 180023	Distance >2.32	Motor
A -34.893551 180025 0	O -56.155521 180025	Distance >1.86	Motor
A -34.893539 180028 0	O -56.155518 180028	Distance >0.46	Motor
A -34.893532 180031 1	O -56.155518 180031	Distance >0.46	
A -34.893528 180034 1	O -56.155510 180034	Distance >0.93	
A -34.893524 180042 1	O -56.155506 180042	Distance >1.39	
A -34.893486 180049 1	O -56.155502 180049	Distance >5.57	
A -34.893467 180053 1	O -56.155506 180053	Distance >7.42	
A -34.893471 180057 1	O -56.155502 180057	Distance >6.96	
A -34.893475 180101 1	O -56.155499 180101	Distance >6.96	
A -34.893482 180105 1	O -56.155495 180105	Distance >6.03	
A -34.893486 180109 1	O -56.155495 180109	Distance >5.57	
A -34.893501 180113 1	O -56.155491 180113	Distance >4.17	
A -34.893528 180117 1	O -56.155491 180117	Distance >1.39	
A -34.893551 180121 0	O -56.155491 180121	Distance >1.39	
A -34.893574 180125 0	O -56.155495 180125	Distance >3.71	
A -34.893600 180129 0	O -56.155495 180129	Distance >6.49	
A -34.893620 180133 0	O -56.155499 180133	Distance >8.81	Buzzer

Tabla A.10: Resultados obtenidos en la prueba LAT1.

Resultados prueba LAT2

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.893948 174640 0	O -56.155857 174640	Distance >21.15	
A -34.893944 174643 0	O -56.155872 174643	Distance >19.49	
A -34.893944 174647 0	O -56.155895 174647	Distance >17.00	
A -34.893940 174651 0	O -56.155910 174651	Distance >15.34	
A -34.893944 174655 0	O -56.155926 174655	Distance >13.68	Buzzer
A -34.893948 174701 0	O -56.155949 174701	Distance >11.20	Buzzer
A -34.893948 174706 0	O -56.155983 174706	Distance >7.46	Buzzer
A -34.893948 174711 0	O -56.156006 174711	Distance >4.98	Motor
A -34.893951 174714 0	O -56.156013 174714	Distance >4.15	Motor
A -34.893951 174716 0	O -56.156013 174716	Distance >4.15	Motor
A -34.893955 174719 0	O -56.156033 174719	Distance >2.07	Motor
A -34.893959 174722 0	O -56.156052 174722	Distance >0.00	Motor
A -34.893963 174725 1	O -56.156071 174725	Distance >2.07	
A -34.893963 174728 1	O -56.156078 174728	Distance >2.90	
A -34.893963 174732 1	O -56.156082 174732	Distance >3.32	
A -34.893967 174736 1	O -56.156105 174736	Distance >5.80	
A -34.893974 174740 1	O -56.156136 174740	Distance >9.12	
A -34.893978 174744 1	O -56.156158 174744	Distance >11.61	
A -34.893974 174755 1	O -56.156136 174755	Distance >9.12	
A -34.893970 174759 1	O -56.156109 174759	Distance >6.22	
A -34.893974 174803 1	O -56.156105 174803	Distance >5.80	
A -34.893978 174807 1	O -56.156105 174807	Distance >5.80	
A -34.893982 174811 1	O -56.156105 174811	Distance >5.80	
A -34.893982 174815 1	O -56.156090 174815	Distance >4.15	
A -34.893986 174820 1	O -56.156090 174820	Distance >4.15	
A -34.893986 174824 1	O -56.156086 174824	Distance >3.73	
A -34.893986 174828 1	O -56.156086 174828	Distance >3.73	
A -34.893986 174832 1	O -56.156090 174832	Distance >4.15	
A -34.893986 174835 1	O -56.156094 174835	Distance >4.56	
A -34.893986 174839 1	O -56.156078 174839	Distance >2.90	
A -34.893986 174843 1	O -56.156055 174843	Distance >0.41	
A -34.893986 174847 0	O -56.156036 174847	Distance >1.66	
A -34.893982 174851 0	O -56.156017 174851	Distance >3.73	
A -34.893982 174855 0	O -56.156013 174855	Distance >4.15	
A -34.893978 174859 0	O -56.156006 174859	Distance >4.98	
A -34.893978 174901 0	O -56.155994 174901	Distance >6.22	Buzzer
A -34.893974 174906 0	O -56.155975 174906	Distance >8.29	Buzzer

Tabla A.11: Resultados obtenidos en la prueba LAT2.

Apéndice A. Resultados de pruebas de perímetro virtual

Resultados prueba LAT3

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.894138 173518 0	O -56.155655 173518	Distance >15.36	
A -34.894150 173522 0	O -56.155651 173522	Distance >14.37	Buzzer
A -34.894169 173527 0	O -56.155655 173527	Distance >12.39	Buzzer
A -34.894169 173532 0	O -56.155651 173532	Distance >12.39	Buzzer
A -34.894188 173538 0	O -56.155655 173538	Distance >9.91	Buzzer
A -34.894199 173543 0	O -56.155655 173543	Distance >8.92	Buzzer
A -34.894215 173548 0	O -56.155655 173548	Distance >7.43	Buzzer
A -34.894230 173554 0	O -56.155659 173554	Distance >5.45	Buzzer
A -34.894238 173559 0	O -56.155670 173559	Distance >4.46	Motor
A -34.894249 173602 0	O -56.155670 173602	Distance >3.47	Motor
A -34.894245 173605 0	O -56.155670 173605	Distance >3.47	Motor
A -34.894249 173607 0	O -56.155678 173607	Distance >3.47	Motor
A -34.894260 173610 0	O -56.155682 173610	Distance >1.98	Motor
A -34.894268 173613 0	O -56.155689 173613	Distance >0.99	Motor
A -34.894276 173616 0	O -56.155689 173616	Distance >0.50	Motor
A -34.894291 173621 1	O -56.155697 173621	Distance >1.49	
A -34.894299 173624 1	O -56.155697 173624	Distance >1.98	
A -34.894299 173628 1	O -56.155701 173628	Distance >2.48	
A -34.894302 173632 1	O -56.155704 173632	Distance >2.97	
A -34.894306 173636 1	O -56.155704 173636	Distance >2.97	
A -34.894314 173640 1	O -56.155701 173640	Distance >3.96	
A -34.894318 173643 1	O -56.155701 173643	Distance >4.46	
A -34.894314 173647 1	O -56.155704 173647	Distance >3.96	
A -34.894318 173651 1	O -56.155712 173651	Distance >4.46	
A -34.894337 173655 1	O -56.155716 173655	Distance >6.44	
A -34.894348 173659 1	O -56.155704 173659	Distance >7.93	
A -34.894344 173703 1	O -56.155701 173703	Distance >7.43	
A -34.894367 173707 1	O -56.155693 173707	Distance >9.42	
A -34.894398 173711 1	O -56.155682 173711	Distance >12.88	
A -34.894390 173715 1	O -56.155682 173715	Distance >11.89	
A -34.894360 173719 1	O -56.155685 173719	Distance >8.92	
A -34.894310 173726 1	O -56.155697 173726	Distance >3.47	
A -34.894291 173730 1	O -56.155704 173730	Distance >1.49	
A -34.894272 173734 0	O -56.155701 173734	Distance >0.50	
A -34.894245 173738 0	O -56.155697 173738	Distance >3.47	
A -34.894230 173742 0	O -56.155697 173742	Distance >5.45	
A -34.894218 173746 0	O -56.155693 173746	Distance >6.44	
A -34.894218 173751 0	O -56.155685 173751	Distance >6.44	Buzzer

Tabla A.12: Datos obtenidos para prueba LAT3.

Resultados prueba LAT4

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.893940 171835 0	O -56.155190 171835	Distance >15.75	
A -34.893940 171839 0	O -56.155186 171839	Distance >15.33	
A -34.893940 171843 0	O -56.155174 171843	Distance >14.09	Buzzer
A -34.893936 171848 0	O -56.155174 171848	Distance >14.09	Buzzer
A -34.893932 171853 0	O -56.155159 171853	Distance >12.43	Buzzer
A -34.893932 171859 0	O -56.155155 171859	Distance >12.02	Buzzer
A -34.893932 171904 0	O -56.155155 171904	Distance >12.02	Buzzer
A -34.893932 171909 0	O -56.155159 171909	Distance >12.43	Buzzer
A -34.893936 171920 0	O -56.155163 171920	Distance >12.85	Buzzer
A -34.893936 171925 0	O -56.155159 171925	Distance >12.43	Buzzer
A -34.893936 171930 0	O -56.155151 171930	Distance >11.60	Buzzer
A -34.893944 171936 0	O -56.155125 171936	Distance >8.70	Buzzer
A -34.893944 171941 0	O -56.155109 171941	Distance >7.05	Buzzer
A -34.893944 171945 0	O -56.155106 171945	Distance >6.63	
A -34.893948 171950 0	O -56.155098 171950	Distance >5.80	Buzzer
A -34.893948 171955 0	O -56.155090 171955	Distance >4.97	Motor
A -34.893948 171957 0	O -56.155090 171957	Distance >4.97	Motor
A -34.893948 172000 0	O -56.155090 172000	Distance >4.97	Motor
A -34.893951 172003 0	O -56.155087 172003	Distance >4.56	Motor
A -34.893955 172006 0	O -56.155079 172006	Distance >3.73	Motor
A -34.893955 172009 0	O -56.155067 172009	Distance >2.49	Motor
A -34.893955 172011 0	O -56.155067 172011	Distance >2.49	Motor
A -34.893955 172014 0	O -56.155071 172014	Distance >2.90	Motor
A -34.893959 172017 0	O -56.155071 172017	Distance >2.90	Motor
A -34.893959 172020 0	O -56.155067 172020	Distance >2.49	Motor
A -34.893959 172022 0	O -56.155071 172022	Distance >2.90	Motor
A -34.893959 172025 0	O -56.155067 172025	Distance >2.49	Motor
A -34.893963 172028 0	O -56.155064 172028	Distance >2.07	Motor
A -34.893963 172031 0	O -56.155064 172031	Distance >2.07	Motor
A -34.893959 172034 0	O -56.155064 172034	Distance >2.07	Motor
A -34.893959 172036 0	O -56.155064 172036	Distance >2.07	Motor
A -34.893959 172039 0	O -56.155064 172039	Distance >2.07	Motor

Tabla A.13: Datos obtenidos para prueba LAT4.

Apéndice A. Resultados de pruebas de perímetro virtual

Latitud,Hora,Z	Longitud, Hora	Distancia (m)	Estimulación
A -34.893955 172042 0	O -56.155064 172042	Distance >2.07	Motor
A -34.893955 172045 0	O -56.155060 172045	Distance >1.66	Motor
A -34.893959 172050 0	O -56.155056 172050	Distance >1.24	Motor
A -34.893963 172053 0	O -56.155052 172053	Distance >0.83	Motor
A -34.893963 172056 1	O -56.155045 172056	Distance >0.00	
A -34.893967 172059 1	O -56.155041 172059	Distance >0.41	
A -34.893970 172103 1	O -56.155037 172103	Distance >0.83	
A -34.893970 172107 1	O -56.155037 172107	Distance >0.83	
A -34.893970 172111 1	O -56.155037 172111	Distance >0.83	
A -34.893970 172115 1	O -56.155037 172115	Distance >0.83	
A -34.893970 172119 1	O -56.155041 172119	Distance >0.41	
A -34.893970 172122 1	O -56.155037 172122	Distance >0.83	
A -34.893970 172126 1	O -56.155037 172126	Distance >0.83	
A -34.893970 172130 0	O -56.155045 172130	Distance >0.00	
A -34.893974 172134 0	O -56.155048 172134	Distance >0.41	
A -34.893974 172138 0	O -56.155052 172138	Distance >0.83	
A -34.893974 172142 0	O -56.155052 172142	Distance >0.83	
A -34.893974 172144 0	O -56.155052 172144	Distance >0.83	Motor
A -34.893974 172147 0	O -56.155052 172147	Distance >0.83	Motor
A -34.893978 172150 1	O -56.155037 172150	Distance >0.83	
A -34.893978 172157 1	O -56.155022 172157	Distance >2.49	
A -34.893970 172201 1	O -56.155014 172201	Distance >3.32	
A -34.893970 172205 1	O -56.154995 172205	Distance >5.39	
A -34.893970 172209 1	O -56.154972 172209	Distance >7.87	
A -34.893963 172213 1	O -56.154961 172213	Distance >9.12	
A -34.893959 172217 1	O -56.154953 172217	Distance >9.95	
A -34.893951 172221 1	O -56.154961 172221	Distance >9.12	
A -34.893948 172225 1	O -56.154991 172225	Distance >5.80	
A -34.893944 172229 1	O -56.155037 172229	Distance >0.83	
A -34.893940 172233 0	O -56.155060 172233	Distance >1.66	
A -34.893944 172237 0	O -56.155060 172237	Distance >1.66	
A -34.893936 172244 0	O -56.155102 172244	Distance >6.22	Buzzer
A -34.893936 172249 0	O -56.155128 172249	Distance >9.12	Buzzer
A -34.893936 172254 0	O -56.155128 172254	Distance >9.12	Buzzer
A -34.893932 172300 0	O -56.155132 172300	Distance >9.53	Buzzer
A -34.893932 172305 0	O -56.155132 172305	Distance >9.53	Buzzer

Tabla A.14: Continúan datos obtenidos en la prueba LAT4.

Apéndice B

Tablas de costos

ANÁLISIS DE COSTOS

Tipo de cambio

35

Prototipo Tesis					Tomando Medidas Simples				
					- Importando la totalidad de los componentes - Cambiando paneles flexibles por rígidos - Imprimiendo carcasas por cuenta propia - Utilizando cargador sin MPPT				
Componentes	Unidades	Proveedor	Costo USD	Costo \$	Componentes	Unidades	Proveedor	Costo USD	Costo \$
Moteino MEGA USB	1	LowPowerLab	37,95	1328,3	Moteino MEGA USB	1	(*1)	37,95	1328,25
GPS Ublox M8N	1	Amazon	18,90	661,5	GPS Ublox M8N	1	(*2)	5	175
Buzzer	1	Ebay	3,00	105	Buzzer	1	(*3)	0,25	8,75
MPPT module	1	Sparkfun	27,00	945	18650 charger module	1	(*4)	1,3	45,5
Capacitor 1000uF 16V	1	Eneka	0,59	20,5	Capacitor 1000uF 16V	1	(*5)	0,5	17,5
Transistor 2N2222A	1	Eneka	1,80	63	Transistor 2N2222A	1	(*6)	1	35
Resistencia 100 Ohms	1	Eneka	0,06	2	Resistencia 100 Ohms	1	(*8)	0,0006	0,021
Zócalos de 10 pines	7	Eneka	4,00	140	Zócalos de 10 pines	7	(*9)	0,1	3,5
Headers 10 pines	7	Eneka	2,00	70	Headers 10 pines	7	(*10)	0,2	7
Bornera 2 terminales	2	Eneka	0,86	30	Bornera 2 terminales	2	(*11)	0,1	3,5
Bornera 3 terminales	1	Eneka	0,57	20	Bornera 3 terminales	1	(*11)	0,1	3,5
Separadores PCB	4	Eneka	1,14	40	Separadores PCB	4	(*12)	1,33	46,55
Portapila 18650	1		2,43	85	Portapila 18650	1	(*13)	0,2	7
Pila 18650 2200 mah	1		4,29	150	Pila 18650 2200 mah	1	(*14)	0,9	31,5
Switch On/Off	1	Mundo electrón	0,57	20	Switch On/Off	1	(*15)	0,055	1,925
Estaño 0,75mm (metro)	2	Eneka	1,56	54,6	Estaño 0,75mm (metro)	2	(*16)	0,102	3,57
Panel solar flexible 1,5V 0,5W	2	Amazon	32,00	1120	Panel solar flexible 1,5V 0,5W	0		0	0
Panel solar rígido 4,5V 0,5W	1	Ebay	4,00	140	Panel solar rígido 4,5V 0,5W	3	(*17)	3	105
Motor DC vibrador	1		2,86	100	Motor DC vibrador	1	(*18)	1	35
Carcasas	3	Fabrix?	64,60	2261	Carcasas	3	(*19)	2	70
PCB universal	1	Eneka	3,51	123	Placa cobre	1	(*20)	2,5	87,5
Cinta mochila (metro)	2	Mercería Lanazu	3,51	123	Cinta mochila (metro)	2	(*21)	2	70
Velcro (metro)	1	Mercería Lanazu	2,86	100	Velcro (metro)	1	(*23)	0,5	17,5
Hebilla plástico	1	Mercería Lanazu	1,37	48	Hebilla plástico	1	(*24)	0,59	20,65
Cables 1mm (metro)	1	Eneka	0,29	10	Cables 1mm (metro)	1	(*25)	0,2	7
Tornillos									
TOTAL			221,71	7759,9	TOTAL			60,88	2130,72
			USD	\$				USD	\$

Importando al por mayor (>5000 unidades)					Comprando micro y módulo LoRa en vez de Moteino				
Componentes	Unidades	Proveedor	Costo USD	Costo \$	Componentes	Unidades	Proveedor	Costo USD	Costo \$
Moteino MEGA USB	1	(*1)	33,36	1167,6	Moteino MEGA USB	1			
GPS Ublox M8N	1	(*2)	1	35	GPS Ublox M8N	1	(*2)	1	35
Buzzer	1	(*3)	0,18	6,3	Buzzer	1	(*3)	0,18	6,3
18650 charger module	1	(*4)	0,7	24,5	18650 charger module	1	(*4)	0,7	24,5
Capacitor 1000uF 16V	1	(*5)	0,01	0,35	Capacitor 1000uF 16V	1	(*5)	0,01	0,35
Transistor 2N2222A	1	(*6)	0,1	3,5	Transistor 2N2222A	1	(*6)	0,1	3,5
Resistencia 100 Ohms	1	(*8)	0,00046	0,0161	Resistencia 100 Ohms	1	(*8)	0,00046	0,0161
Zócalos de 10 pines	7	(*9)	0,01	0,35	Zócalos de 10 pines	7	(*9)	0,01	0,35
Headers 10 pines	7	(*10)	0,01	0,35	Headers 10 pines	7	(*10)	0,01	0,35
Bornera 2 terminales	2	(*11)	0,001	0,035	Bornera 2 terminales	2	(*11)	0,001	0,035
Bornera 3 terminales	1	(*11)	0,001	0,035	Bornera 3 terminales	1	(*11)	0,001	0,035
Separadores PCB	4	-	0,07	2,45	Separadores PCB	4	-	0,07	2,45
Portapila 18650	1	(*13)	0,1	3,5	Portapila 18650	1	(*13)	0,1	3,5
Pila 18650 2200 mah	1	(*14)	0,67	23,45	Pila 18650 2200 mah	1	(*14)	0,67	23,45
Switch On/Off	1	(*15)	0,055	1,925	Switch On/Off	1	(*15)	0,055	1,925
Estaño 0,75mm (metro)	2	(*16)	0,096	3,36	Estaño 0,75mm (metro)	2	(*16)	0,096	3,36
Panel solar flexible 1,5V 0,5W	0		0	0	Panel solar flexible 1,5V 0,5W	0		0	0
Panel solar rígido 4,5V 0,5W	3	(*17)	0,3	10,5	Panel solar rígido 4,5V 0,5W	3	(*17)	0,3	10,5
Motor DC vibrador	1	(*18)	0,6	21	Motor DC vibrador	1	(*18)	0,6	21
Carcasas	3	(*19)	1,25	43,75	Carcasas	3	(*19)	1,25	43,75
Placa cobre	1	(*20)	0,75	26,25	Placa cobre	1	(*20)	0,75	26,25
Cinta mochila (metro)	2	(*22)	0,05	1,75	Cinta mochila (metro)	2	(*22)	0,05	1,75
Velcro (metro)	1	(*23)	0,2	7	Velcro (metro)	1	(*23)	0,2	7
Hebilla plástico	1	(*24)	0,02	0,7	Hebilla plástico	1	(*24)	0,02	0,7
Cables 1mm (metro)	1	(*25)	0,02	0,7	Cables 1mm (metro)	1	(*25)	0,02	0,7
					Micro Atmega 1284p	1	(*26)	0,1	3,5
					RFM95 LoRa module	1	(*27)	0,1	3,5
					Otros componentes				10
TOTAL			39,55	1384,37	TOTAL			6,39	233,77
			USD	\$				USD	\$
Obs: El moteino representa el 84% del costo total									

(*)	
1	https://lowpowerlab.com/shop/product/138
2	https://www.alibaba.com/product-detail/NEO-8M-GPS-Module-APM2-56_60818954451.html?spm=a2700.7724838.2017115.83.33de1a94Vxi8gf
3	https://www.alibaba.com/product-detail/Passive-Buzzer-Module-for-Arduinos-Low_62054635956.html?spm=a2700.7724838.2017115.114.38654fc3i79ydg
4	https://www.alibaba.com/product-detail/Lithium-ion-Battery-Charger-Module-1_60800526492.html?spm=a2700.galleryofferlist.normalList.382.41723c12pJFFuB
5	https://www.alibaba.com/product-detail/Factory-Electrolytic-capacitor-50V-470UF-35V_60648351945.html?spm=a2700.7724838.2017115.58.1b38211esFdFUo
6	https://www.alibaba.com/product-detail/Hot-offer-Transistor-2N2222A-2N2222-TO_60768500152.html?spm=a2700.7724838.2017115.114.1b6d4961qJTRbc
7	https://www.alibaba.com/product-detail/Small-iron-cap-Gold-seal-triode_60772800547.html?spm=a2700.7724838.2017115.140.1b6d4961qJTRbc
8	https://www.alibaba.com/product-detail/Metal-Film-Resistor-1-8W-1_60778948755.html?spm=a2700.7724838.2017115.203.2e8a1c22vUtZ6V
9	https://www.alibaba.com/product-detail/2-54mm-Pitch-10-Pin-Single_60451345083.html?spm=a2700.7724838.2017115.147.69185f99FmNc5b
10	https://www.alibaba.com/product-detail/Low-Price-1-to-40pin-single_60820108436.html?spm=a2700.7724838.2017115.134.69185f99FmNc5b
11	https://www.alibaba.com/product-detail/high-quality-factory-price-pcb-rising_60249108182.html?spm=a2700.7724838.2017115.451.2e9c77aeFRb2Ei
12	https://www.dx.com/p/rxdz-metric-m3-thread-nylon-insert-lock-screw-fastener-hexagon-hex-nuts-black-100pcs-2021742#XNHJ5Y5KJlU
	https://www.dx.com/p/zndiy-bry-r201-306-m3-x-6-6-nylon-spacer-hex-pillars-for-r-c-multicopters-black-20-pcs-2045034#XNHKLo5KJlU
	https://www.dx.com/p/zndiy-bry-r205-312-m3-x-12-plastic-nylon-screws-for-multicopter-flight-black-20-pcs-2042659#XNHKQY5KJlU
13	https://www.alibaba.com/product-detail/TBH-18650-1B-W-Ningbo-TECO_60725964619.html?spm=a2700.7724838.2017115.474.7dcc3972GW90Iw
14	https://www.alibaba.com/product-detail/3-7-V-4000-mAh-18650_60580368922.html?spm=a2700.7724838.2017115.88.790368efc01fyg
15	https://www.ebay.com/itm/40pcs-Black-Small-Size-SPDT-Slide-Switch-On-Off-3-Pin-PCB-5V-0-3A-DIY-Projects/253240207235?hash=item3af64ae783:g:W54AAOSwopRYgGOf&frceupt=true
16	https://www.alibaba.com/product-detail/KOOCU-500g-Sn60-Pb40-Solder-Wire_60608773740.html?spm=a2700.7724838.2017115.210.4e6d4583Ny8TQb
17	https://www.alibaba.com/product-detail/1v-2v-3v-5v-6v-12v_60615344294.html?spm=a2700.7724838.2017115.61.579a2e95EISVeT
18	https://www.alibaba.com/product-detail/Torque-current-rpm-output-power-370_60773566421.html?spm=a2700.7724838.2017115.292.17d812d0ihRRHP
19	https://www.alibaba.com/product-detail/3kg-filament-plc-3d-printer-filament_62054774425.html?spm=a2700.7724838.2017115.1.1f246dccDD5WNG
20	https://www.alibaba.com/product-detail/15cm-20cm-Glass-fiber-Universal-PCB_62027105415.html?spm=a2700.details.maylikehoz.2.32866c5bUg8aRc
21	https://www.alibaba.com/product-detail/High-Strength-Jacquard-25MM-PP-Polypropylene_60806649346.html?spm=a2700.galleryofferlist.normalList.59.6a2861fdWJlkN3
22	https://www.alibaba.com/product-detail/Polypropylene-Webbing-Strap-for-backpacks_60159911492.html?spm=a2700.galleryofferlist.normalList.122.6a2861fdWJlkN3
23	https://www.alibaba.com/product-detail/Low-Minimum-Order-Quantity-Nylon-Hook_60809909982.html?spm=a2700.galleryofferlist.normalList.44.37b93b21WqdbtN&s=p
24	https://www.alibaba.com/product-detail/2-Inch-Wide-Black-Adjustable-Buckles_62026543286.html?spm=a2700.galleryofferlist.normalList.64.5d906c84Mz1MSk
25	https://www.alibaba.com/product-detail/VDE-Cable-FLRY-B-single-core_60779291777.html?spm=a2700.7724838.2017115.189.15c87bc3UN5XWh
26	https://www.alibaba.com/product-detail/Original-New-IC-ATMEGA1284P_748738845.html?spm=a2700.7724838.2017115.23.432324a4E5OFr
27	https://www.alibaba.com/product-detail/RFM95-RFM95W-SX1276-Wireless-Transceiver-Module_62031484952.html?spm=a2700.7724838.2017115.81.7fb72f0dF9SHT8

Apéndice C

Manual de usuario de la Interfaz web

El diseño del dispositivo fue acompañado con una interfaz web que le permite al usuario interactuar con el mismo (<http://18.222.96.150:3000/>). Esta interfaz posibilita las siguientes acciones:

- **Creación y eliminación de múltiples perímetros:** La secuencia para crear un perímetro consiste en hacer click primero en el botón con forma de polígono, ubicado en la parte superior del mapa. Luego se hacen los sucesivos clicks indicando los vértices del perímetro virtual a crear, conectando el último con el primero (figura C.1).

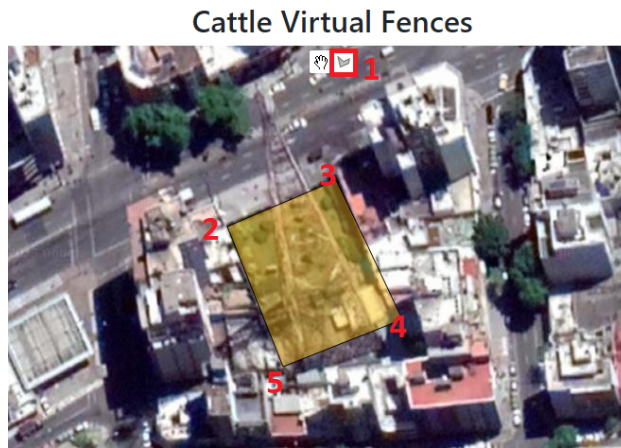


Figura C.1: Ejemplo de creación de un PV a través de la interfaz web.

Al momento de crear el PV debe asegurarse que el polígono utilizado para la figura sea simple (sus lados no se intersectan). En caso de que esta última condición no se cumpla aparecerá un mensaje de alerta indicando el error (figura C.2).

Apéndice C. Manual de usuario de la Interfaz web

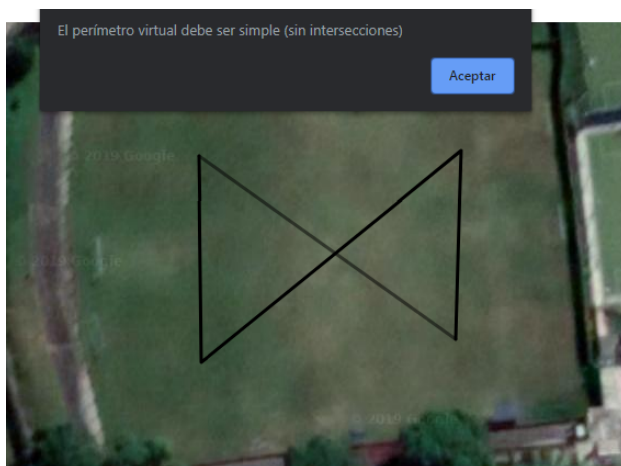


Figura C.2: Alerta de error al crear un polígono no simple para el uso del PV.

También se agregó la opción de borrar los perímetros. Esto es útil en caso de que los vértices no se encuentren en la posición deseada, o en caso de que dicho perímetro no volverá a ser utilizado. Esta acción se ejecuta en la misma ventana que se obtiene al clicar sobre un perímetro, y luego haciendo click sobre el botón “Delete”. Como se puede observar en la figura C.3.

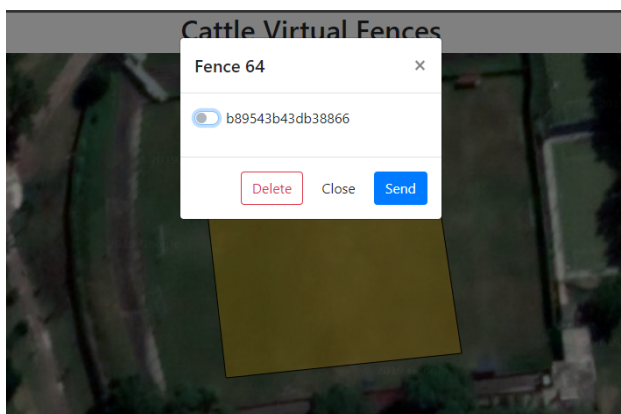


Figura C.3: Eliminación de un PV.

- **Configuración de cada uno de los dispositivos registrados:** En la interfaz gráfica los dispositivos se representan en el mapa con unos marcadores. Haciendo click sobre alguno de los marcadores, aparecerá un panel lateral, mostrando los últimos valores registrados para cada parámetro de configuración del dispositivo seleccionado (figura C.4).

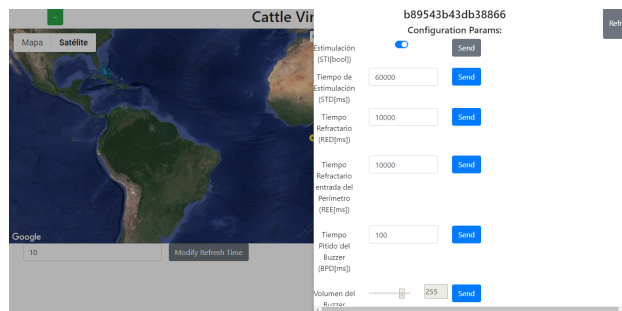


Figura C.4: Configuración de los dispositivos a través de la interfaz web. Sobre la derecha se puede observar un panel con los distintos parámetros a configurar.

Para que un parámetro sea modificado en el dispositivo, será necesario modificar el valor en el campo indicado y luego darle al botón “Send”, asociado a dicho parámetro. También se dispone de un botón “Refresh”, que al presionarlo se encarga de consultar nuevamente los parámetros guardados en la base de datos.

A efectos de verificación y manejo de casos borde, se chequea que los valores ingresados pertenezcan al rango manejado para el parámetro modificado. En la figura C.5 el parámetro fuera de rango es RED (Refractory Duration), indicando correctamente que 1000002134 no pertenece al rango (0,999999).

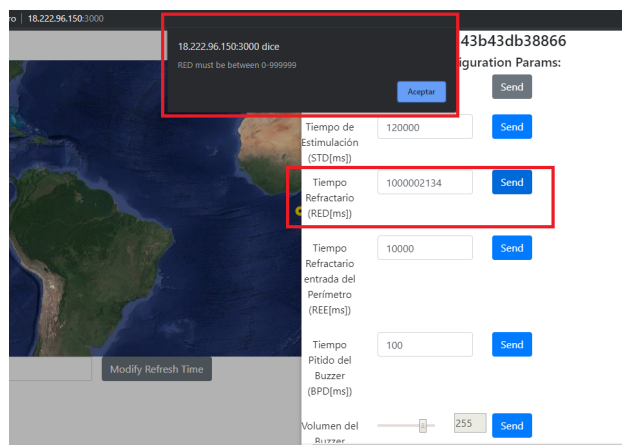


Figura C.5: Ejemplo de chequeo de casos borde en el ingreso de parámetros.

- **Asignación de perímetros a determinados dispositivos:** Haciendo click sobre un perímetro aparecerá una lista con los dispositivos disponibles, indicando con un “switch” si pertenecen o no al mismo. Además se podrá agregar los dispositivos que no pertenezcan a través del botón “Send” (figura C.3), quedando configurado ese perímetro para los dispositivos indicados.
- **Visualización de la última ubicación registrada de cada dispositivo:** Al pasar el cursor sobre un marcador de dispositivo, este despliega una

Apéndice C. Manual de usuario de la Interfaz web

pequeña ventana de información, mostrando la hora en la que fue registrada dicha ubicación, así como el identificador del dispositivo (figura C.6).

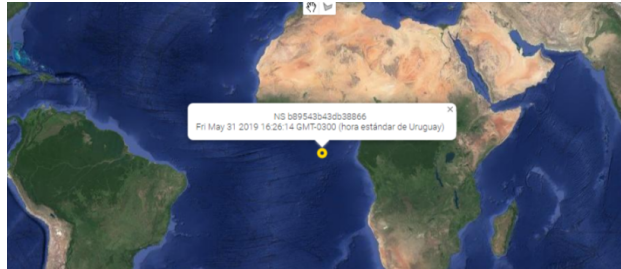


Figura C.6: Visualización de la posición de un dispositivo a través de la interfaz.

- **Visualización de los registros de cada dispositivo, en forma de tabla y en un mapa:** Se podrá ver el historial de cada dispositivo, tanto en forma de tabla (indicando para cada timestamp la fecha, hora, ubicación y si estuvo dentro o fuera del perímetro seteado) como en forma de mapa, mostrando la traza del animal (figura C.7).

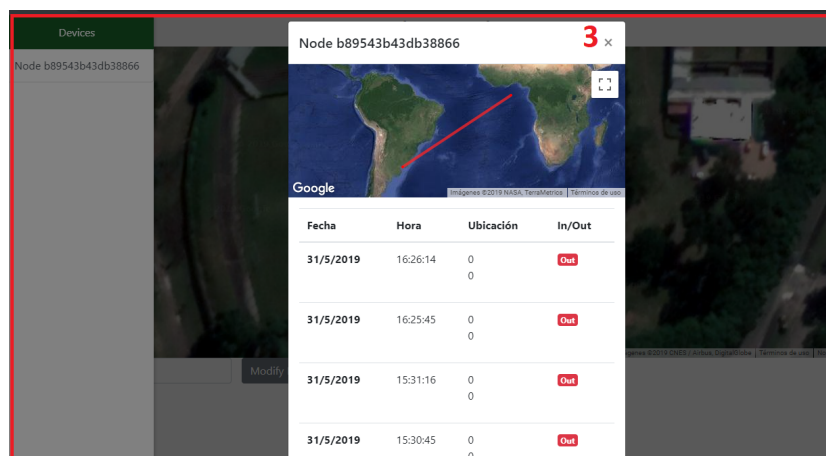


Figura C.7: Historial de la posición del dispositivo mostrando los datos en un mapa y en una tabla.

Apéndice D

Prueba de concepto en Casupá, Florida

Con el objetivo de visualizar la reacción a los estímulos de un bovino de ganado extensivo, se realizó en noviembre de 2018 una prueba de concepto en Casupá, Florida (ver sección 1.6.3).

Para esta prueba se armó un prototipo, que difiere del NS descrito en la sección en que utiliza Arduino UNO como Microcontrolador, no utiliza paneles solares, y transmite la posición utilizando un módulo de comunicación HC12 [37].

En dichas pruebas, se constató que el bovino no reaccionó al estímulo sonoro, llegando inclusive a pastar mientras estaba activo. Sin embargo, se visualizó una reacción significativa del bovino al estímulo táctil. Cuando el motor comenzó a vibrar, el bovino levantó la cabeza y se desplazó en sentido contrario.

En la figura D.1 se muestra el prototipo diseñado, y el bovino portando el mismo durante el transcurso de la prueba.

Apéndice D. Prueba de concepto en Casupá, Florida



(a)



(b)



(c)

Figura D.1: Pruebas en bovinos con prototipo de prueba

Referencias

- [1] Instituto Nacional de Carnes. Sector cárnico Uruguayo, situación actual. https://www.inac.uy/innovaportal/file/10495/1/pi_modulo_introductorio_web.pdf. último acceso: 16/5/19.
- [2] Ministerio de Ganadería, Agricultura y pesca. Anuario estadístico agropecuario 2015. <http://www.mgap.gub.uy/unidad-organizativa/oficina-de-programacion-y-politicas-agropecuarias/publicaciones/anuarios-diea/anuario2015>.
- [3] D.M. Anderson P. Sikka C. Crossman P. Corke G.J. Bishop-Hurley, D.L. Swain. Virtual fencing applications: Implementing and testing an automated cattle control system. Computers and Electronics in Agriculture, 56:14–22, 2007.
- [4] Sally J. Haynes William J. Farrer Christopher J. Leigh-Lancaster Caroline Leea Dana L.M. Campbella, Jim M. Leaa. Virtual fencing of cattle using an automated collar in a feed attractant trial. Applied Animal Behaviour Science, 200:71–77, 2018.
- [5] William J. Farrer Sally J. Haynes Dana L. M. Campbell, Jim M. Lea and Caroline Lee. Tech-savvy beef cattle? how heifers respond to moving virtual fence lines. Animals, 7(72):1–12, 2017.
- [6] Christina Umstatter. The evolution of virtual fences: A review. Computers and Electronics in Agriculture, 75:10–22, 2011.
- [7] Ron Peterson Daniela Rus Zack Butler, Peter Corke. From robots to animals: Virtual fences for controlling cattle. The International Journal of Robotics Research, 25(5-6):485–508, 2006.
- [8] Ron Peterson Daniela Rus Zack Butler, Peter Corke. Virtual fences for controlling cows. The International Journal of Robotics Research, 25:4429–4436, 2004.
- [9] Fing/Udelar. Heterogeneous Computing laboratory. <https://www.fing.edu.uy/inco/grupos/gpgpu/>. último acceso: 4/6/19.
- [10] Arduino. Arduino MEGA 2560. <http://www.mantech.co.za/datasheets/products/A000047.pdf>. último acceso: 19/5/19.

Referencias

- [11] HopeRF Electronic. Low Power Long Range Transceiver Module. https://cdn.sparkfun.com/assets/learn_tutorials/8/0/4/RFM95_96_97_98W.pdf. último acceso: 19/5/19.
- [12] HopeRF Electronic. RFM69HCW ISM Transceiver Module. <https://cdn.sparkfun.com/datasheets/Wireless/General/RFM69HCW-V1.1.pdf>. último acceso: 19/5/19.
- [13] U-blox. NEO-6 u-blox 6 GPS Modules Datasheet. [https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_\(GPS.G6-HW-09005\).pdf](https://www.u-blox.com/sites/default/files/products/documents/NEO-6_DataSheet_(GPS.G6-HW-09005).pdf). último acceso: 19/5/19.
- [14] U-blox. NEO-7 u-blox 7 GNSS modules DataSheet. https://www.u-blox.com/sites/default/files/products/documents/NEO-7_DataSheet_%28UBX-13003830%29.pdf. último acceso: 19/5/19.
- [15] U-blox. NEO-M8 u-blox M8 concurrent GNSS modules DataSheet. https://www.u-blox.com/sites/default/files/NEO-M8_DataSheet_%28UBX-13003366%29.pdf. último acceso: 19/5/19.
- [16] Wikipedia. Zumbador. <https://es.wikipedia.org/wiki/Zumbador>. último acceso: 19/5/19.
- [17] Grobotronics. Passive Buzzer Module. <https://grobotronics.com/passive-buzzer-module.html?sl=en>. último acceso: 19/5/19.
- [18] NanJing Top Power ASIC Corp. TP4056 1A Standalone Linear Li-Ion Battery Charger with Thermal Regulation in SOP-8 . <https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Prototyping/TP4056.pdf>. último acceso: 10/6/19.
- [19] Amazon. Motor vibrador. https://www.amazon.es/FTDDFJAS-Vibrador-Vibraci%C3%B3n-Motores-Masajeador/dp/B07T6HDZ5G/ref=sr_1_1?keywords=motor+vibrador+12v&qid=1567378710&s=gateway&sr=8-1. último acceso: 10/6/19.
- [20] Google. Google Maps. <https://www.google.com/maps>. último acceso: 10/6/19.
- [21] Aurora Energy. Distribución anual del HSP en Baltimore. <http://www.aurorasolarenergy.com/average-daily-sun-hours/>. último acceso: 19/5/19.
- [22] LG. Li-ion Battery Datasheet. https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=2ahUKEwiT2KrR6_3iAhXhD7kGHUyYBYEQFjAAegQIBRAC&url=http%3A%2F%2Fwww.farnell.com%2Fdatasheets%2F2634739.pdf&usq=A0vVaw3h1r3RzBHLyJom1d3u3A6I. último acceso: 10/6/19.

- [23] NanJing Top Power ASIC Corp. TP4056 1A Standalone Linear Li-Ion Battery Charger with Thermal Regulation in SOP-8 . <https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Prototyping/TP4056.pdf>. último acceso: 10/6/19.
- [24] Brian Evans. Beginning Arduino Programming. Apress, 2011.
- [25] Leonardo Horovitz. Telemetría de contenedores de residuosl. <https://dspace.ort.edu.uy/bitstream/handle/20.500.11968/3878/Material%20completo.pdf?sequence=-1&isAllowed=y>. último acceso: 2/6/19.
- [26] Banggood. Buy RFM95. <https://www.banggood.com/buy/rfm95-lora.html>. último acceso: 2/6/19.
- [27] Andrew S. Tanenbaum. Redes de Computadoras. <https://books.google.es/books?hl=es&lr=&id=WWD-4oF9hjEC&oi=fnd&pg=PR18&dq=tanenbaum+redes+de+datos&ots=Xzi5-6w9zb&sig=OVVKW23i35DaMnF7M0Y0yhlv6hg#v=onepage&q&f=false>. último acceso: 4/6/19.
- [28] kpn. LoRa Classes. <https://zakelijkforum.kpn.com/lora-forum-16/lorawan-device-classes-a-b-and-c-10972>. último acceso: 10/6/19.
- [29] IBM. LMiC C-Library User Manual. <https://github.com/matthijskooijman/arduino-lmic/blob/master/doc/LMiC-v1.5.pdf>. último acceso: 29/5/19.
- [30] Thomas Laurenson. Node Configuration for Australian and New Zealand Frequencies (AU915 MHz). https://github.com/thomaslaurenson/Dragino_LoRaShield_Node_AU915. último acceso: 29/5/19.
- [31] Everynet. Everynet Platform Introduction. <https://ns.docs.everynet.io>. último acceso: 10/6/19.
- [32] TTN. Limitations LoRaWAN. <https://www.thethingsnetwork.org/forum/t/limitations-data-rate-packet-size-30-seconds-uplink-and-10-messages-downlink-1300>. último acceso: 12/11/19.
- [33] Wikipedia. Point in polygon. https://en.wikipedia.org/wiki/Point_in_polygon#Ray_casting_algorithm. último acceso: 1/6/19.
- [34] Usuario sakseiw. Punto dentro de polígono. <https://sakseiw.wordpress.com/2013/10/04/punto-dentro-de-poligono/>. último acceso: 1/6/19.
- [35] Autocad. Tinkercad. www.tinkercad.com. último acceso: 4/6/19.
- [36] Qoitech. Otii. www.qoitech.com/products/standard. último acceso: 9/6/19.
- [37] elecrow. HC-12 Wireless Serial Port Communication Module . <https://www.elecrow.com/download/HC-12.pdf>. último acceso: 20/6/19.

Esta página ha sido intencionalmente dejada en blanco.

Índice de tablas

1.1. Ventajas y desventajas de los distintos métodos de confinamiento de ganado basado en [6].	4
2.1. RFM9X v RFM69.	16
2.2. Características de los GPS NEO: 6M, 7M y 8N.	17
2.3. Consumo y ciclo de trabajo.	19
2.4. Indicador de HSP. Tomada de [21]	20
3.1. Seteo de parámetros mientras CHP=0.	38
3.2. Seteo de parámetros una vez que CHP=1.	39
3.3. Seteo de parámetros una vez que CHP=1.	40
5.1. Consumo de corriente promedio por módulo.	54
5.2. Datos relevantes obtenidos en la prueba E1.	62
5.3. Resultados principales de la prueba para tiempos refractarios. . . .	64
5.4. Resultados obtenidos en la prueba de velocidad intermedia.	65
5.5. Resultados obtenidos en la prueba de velocidad alta.	65
A.1. Datos obtenidos en la prueba E1.	78
A.2. Continúa datos obtenidos en la prueba E1.	79
A.3. Datos obtenidos en la prueba de refractarios.	80
A.4. Datos obtenidos en la segunda prueba de velocidad.	80
A.5. Datos obtenidos en la tercera prueba de velocidad.	81
A.6. Resultados obtenidos para la prueba E2.	82
A.7. Resultados obtenidos en la prueba E3.	83
A.8. Resultados obtenidos en la prueba E4.	84
A.9. Continúa resultados obtenidos en la prueba E4.	85
A.10. Resultados obtenidos en la prueba LAT1.	86
A.11. Resultados obtenidos en la prueba LAT2.	87
A.12. Datos obtenidos para prueba LAT3.	88
A.13. Datos obtenidos para prueba LAT4.	89
A.14. Continúan datos obtenidos en la prueba LAT4.	90

Esta página ha sido intencionalmente dejada en blanco.

Índice de figuras

1.1. Datos obtenidos del GPS del dispositivo de 6 de los animales durante el transcurso del experimento [5].	6
1.2. Diagrama de bloques del NS propuesto.	10
1.3. Arquitectura del sistema propuesto	10
1.4. Funcionamiento esperado del dispositivo	11
2.1. Arquitectura HW propuesta para el NS.	15
2.2. Buzzer elegido	18
2.3. Motor elegido. Imagen tomada de [19]	18
2.4. Posición relativa entre Baltimore y Uruguay. Imagen tomada de [20].	20
3.1. Modelo de 5 capas que define las diferentes fases por las que deben pasar los datos para viajar de un dispositivo a otro sobre una red de comunicaciones (ver [27]).	25
3.2. Dispositivo LoRa Clase A. Luego de enviar un mensaje al gateway (uplink), el dispositivo abre dos ventanas de recepción (RX Slot 1 y RX Slot 2) en los siguientes 1 y 2 segundos para la posible recepción de mensajes enviados por el gateway (downlink). Imagen tomada de [28].	26
3.3. Dispositivo LoRa Clase B. Además de las ventanas de recepción existentes en un dispositivo Clase A, un Clase B dispone de ventanas de recepción que se abren periódicamente durante un tiempo delimitado por las balizas (beacons). Imagen tomada de [28]. . . .	26
3.4. Dispositivo LoRa Clase C. Luego de enviar un mensaje al GW (uplink), la ventana de recepción (RX Slot) permanece abierta hasta el siguiente uplink. Esto permite la recepción por parte del dispositivo de mensajes enviados por el GW (downlink) en cualquier momento. Imagen tomada de [28].	27
3.5. Vista de alto nivel de todos los componentes del dispositivo de aplicación. Basado en [29].	28

Índice de figuras

3.6. Ejemplo de utilización de la LMiC. En el setup se puede observar la inicialización del “run-time enviroment” a través de la función initLora() que llama a os_init() , el agendado de tareas a las funciones get_gps_data(...) y do_sendLAT(...) , y la ejecución de esas tareas en el loop principal a través de os_runloop_once() . Setup() y loop() son funciones de Arduino que permiten la inicialización de variables y ejecución del cuerpo principal del código respectivamente.	29
3.7. Arquitectura HW propuesta por EveryNet. Figura basada en [31]. .	30
3.8. Estructura del protocolo NMEA utilizado por el GPS	32
3.9. Perímetro virtual	33
3.10. Diagrama de estados según dónde se encuentra el dispositivo, si STI=1.	35
3.11. Explicación de variable BVE.	36
3.12. Cálculo de la distancia del dispositivo al perímetro virtual.	41
4.1. Collar en 3D realizado con Tinkercad.	47
4.2. Collar de sujeción fabricado.	49
4.3. Diseño del encapsulado principal desde varias perspectivas. Dimensiones: 122 x 93.6 x 45.4 mm	49
4.4. Diseño del encapsulado del motor. Dimensiones: 70 x 56 x 26 mm .	50
4.5. Diseño del contrapeso. Dimensiones: 84 x 84 x 40 mm	50
4.6. Fotos del prototipo final.	51
4.7. Fotos del bloque motor y contrapeso.	51
4.8. Fotos del prototipo final en ejemplar bovino ubicado en Facultad de Veterinaria.	52
5.1. Consumo del dispositivo en transmisión. Medición realizada con Otii.	54
5.2. Mensajes enviados por el dispositivo al GW vistos en el monitor serial.	56
5.3. Distancia Facultad de Ingeniería - Ciudad Vieja.	57
5.4. Distancia Fortaleza del Cerro - Facultad de Ingeniería.	57
5.5. Vista satelital de los puntos seleccionados para la prueba de funcionamiento del PV	59
5.6. Esquinas del perímetro delimitado en Parque Batlle	59
5.7. Pruebas a realizar.	61
5.8. Espectro del buzzer sin encapsulado.	66
5.9. Espectro del buzzer con encapsulado.	67
5.10. Pruebas en bovinos con NS.	68
C.1. Ejemplo de creación de un PV a través de la interfaz web.	95
C.2. Alerta de error al crear un polígono no simple para el uso del PV. .	96
C.3. Eliminación de un PV.	96
C.4. Configuración de los dispositivos a través de la interfaz web. Sobre la derecha se puede observar un panel con los distintos parámetros a configurar.	97
C.5. Ejemplo de chequeo de casos borde en el ingreso de parámetros. . .	97
C.6. Visualización de la posición de un dispositivo a través de la interfaz.	98

C.7. Historial de la posición del dispositivo mostrando los datos en un mapa y en una tabla.	98
D.1. Pruebas en bovinos con prototipo de prueba	100

Esta es la última página.
Compilado el miércoles 13 noviembre, 2019.
<http://iie.fing.edu.uy/>