



# **INSTITUTO TECNOLÓGICO DE TUXTLA GUTIÉRREZ**

## **INGENIERÍA ELECTRÓNICA**

### **RESIDENCIA PROFESIONAL**

**Sistema electrónico de vigilancia de Fauna (SIVIFAU)**

**ASESOR**

**Dr. Héctor Ricardo Hernández De León**

**García Pérez Pablo Othoniel 11270463**

**TUXTLA GUTIÉRREZ, CHIAPAS, MEXICO A 30 DE JUNIO  
2016**

## Contenido

<b>Capítulo 1</b> .....	1
<b>1.1 Introducción</b> .....	1
<b>1.2 Antecedentes</b> .....	1
<b>Capítulo 2</b> .....	4
<b>2.1 Objetivos</b> .....	4
2.1.1 Objetivo general.....	4
2.1.2 Objetivos específicos.....	5
<b>2.2 Justificación</b> .....	5
<b>2.3 Planteamiento del problema</b> .....	6
<b>2.4-Alcances y limitaciones</b> .....	7
2.4.1 Alcances .....	7
2.4.2 Limitaciones .....	7
<b>2.5 Caracterización del área en que se participo</b> .....	8
2.5.1 Localización.....	9
2.5.2 Misión, visión, valores .....	13
2.5.3 Descripción del área donde se realizó el proyecto .....	14
<b>Capítulo 3</b> .....	15
<b>3.1 Fundamento teórico</b> .....	15
<b>Capítulo 4</b> .....	59
<b>4.1 Procedimiento y descripción de las actividades realizadas</b> .....	59
4.1.1 Programación en microcontrolador(Arduino) .....	59
4.1.2 Programación en LABVIEW .....	63
4.1.3- Pruebas del funcionamiento del software y hardware en conjunto:.....	79
<b>4.2 Resultados</b> .....	81
<b>Capítulo 5</b> .....	83
<b>5.1 Conclusiones y recomendaciones</b> .....	83
<b>5.2 Competencias desarrolladas y/o aplicadas</b> .....	84
<b>5.3 Referencias bibliográficas</b> .....	85
<b>5.4 Anexo</b> .....	86

# Capítulo 1

## 1.1 Introducción

El Sistema Electrónico de Vigilancia de Fauna (SIVIFAU) es un proyecto realizado para tener un mejor monitoreo y vigilancia de la fauna en una reserva ecológica, zoológicos.

Por qué en Chiapas casi el 90% son animales en peligro de extinción. Y existen algunas reservas ecológicas así como zoológicos en los cuales se intenta preservar estas especies y también reproducir en cautiverio.

Con este podremos saber su posición actual y su recorrido que ha tenido en diferentes tiempos del día visualizando en una extensión de google maps; los animales también tendrán un límite de área la cual si es sobrepasada activara una alarma que alertara a los cuidadores del lugar.

Todo esto fue implementado mediante el uso de GPS y comunicación serial vía radiofrecuencia (xbee), todo esto con llevo a crear un interfaz sencilla y fácil de manejar en labview.

Este proyecto contempla un periodo de realización del 25 de enero al 03 de junio de 2016, en las instalaciones del Instituto Tecnológico de Tuxtla Gutiérrez que se encuentra ubicada en Carretera Panamericana kilómetro 1080, Colonia Terán, 29050 Tuxtla Gutiérrez, Chis.

Esta reporte está dividido por ciertos núm. de cap. en el primer capítulo se profundizara en la problemática que origino la realización de dicho proyecto, así como también la justificación de este, se hará mención del objetivo general y los específicos.

## 1.2 Antecedentes

Estos son algunos primeros sistemas GPS de la historia. Primer sistema de navegación basado en satélites. Entrada en servicio en 1965.

El sistema TRANSIT estaba constituido por una constelación de seis satélites en órbita polar baja, a una altura de 1074 Km. Tal configuración conseguía una

cobertura mundial pero no constante. La posibilidad de posicionarse era intermitente, pudiéndose acceder a los satélites cada 1.5 h. El cálculo de la posición requería estar siguiendo al satélite durante quince minutos continuamente. TRANSIT trabajaba con dos señales en dos frecuencias, para evitar los errores debidos a la perturbación ionosférica. El cálculo de la posición se basaba en la medida continua de la desviación de frecuencia Doppler de la señal recibida y su posterior comparación con tablas y gráficos. El error de TRANSIT estaba en torno a los 250 m. Su gran aplicación fue la navegación de submarinos y de barcos.

NAVSTAR. Sistema de posicionamiento global (GPS)

El primer satélite se lanzó en 1978, y se planificó tener la constelación completa ocho años después. Unido a varios retrasos, el desastre de la lanzadera Challenger paró el proyecto durante tres años. Por fin, en diciembre de 1983 se declaró la fase operativa inicial del sistema GPS.

De aquí nos saltamos a los sistemas de navegación en algunas cosas y animales.

Honda Electro Gyro-Cator su producción fue entre las fechas de 24 de agosto de 1981 – 1982, Gyro-Cator era un dispositivo con una pantalla CRT en la que se señalaba la ruta actual sobre un mapa. El mapa no era representado por la pantalla, sino que eran una especie de transparencias que se insertaban en ella. Además, no utilizaba satélites u otro tipo de posicionamiento, ya que el sistema americano GPS no fue puesto en funcionamiento hasta 1993, mientras que el Electro Gyrocator fue presentado en 1981. Funcionaba básicamente con un giroscopio basado en el movimiento de una corriente de helio dentro de una cámara que mantenía el vacío y la temperatura constantes, para evitar errores de medición debido a la dilatación del gas.



Benefon Esc! fue lanzado al mercado a finales de 1999 fue primer teléfono celular en integrar la función de GPS, su venta se concentró sobre todo en Europa.



Tagg Pet Tracker producido en el año 2011 se compone de dos partes: el aparato que llevan encima las mascotas y un servicio vía web con el que se gestiona el cómo funciona. El receptor capta las coordenadas exactas en las que está el animal. Al mismo tiempo puede emitir esa señal periódicamente haciéndosela llegar al dueño del animal a través de un mensaje de correo, vía web o con un SMS. El coste del aparato principal (base, cargador, etcétera) es de unos 200 dólares (unos 150 euros) e incluye un collar y un año de servicio.

OntheBus, primer GPS para invidentes

En año 2012 Una nueva aplicación descargable para teléfonos con sistema Android, llamada OnTheBus, facilita la orientación y los desplazamientos dentro de las grandes ciudades. La aplicación está basada en los principios del diseño universal y es, por tanto, de utilidad para cualquier usuario que se quiera desplazar con facilidad, y especialmente para las personas con algún tipo de discapacidad visual, auditiva o cognitiva.

De aquí se muestran algunos antecedentes de propuestas e intento de conservar la fauna salvaje:

### **FRACTALIA REMOTE SYSTEMS.**

Desarrollo de un dispositivo electrónico para ayudar a preservar animales en peligro de extinción y más concretamente evitar la muerte de los rinocerontes a manos de cazadores furtivos. La caza furtiva de estos animales se atribuye a la creciente demanda de cuerno de rinoceronte en países asiáticos principalmente China y Vietnam, donde existe la creencia de que el cuerno de este animal posee propiedades medicinales. El alto precio que se llega a pagar por un cuerno ha

atraído la participación de grupos organizados que utilizan equipos de alta tecnología para rastrear y matar a los rinocerontes.

El proyecto desarrollo de un dispositivo electrónico que favorezca la protección de la fauna salvaje con número de expediente TSI- 100103-2014-166 ha sido cofinanciado por el Ministerio de Industria, Energía y Turismo, dentro del Plan de Investigación Científica y Técnica y de innovación 2013-2016 en el marco de la acción estratégica de economía y sociedad digital.

**Manuel Morales. Investigador.** Wildlife Conservation Society – WCS abre una oficina local en Ecuador entre 1999 y 2000. Concentran sus esfuerzos primordialmente en el Parque Nacional Yasuní, trabajando con especies clave como la nutria, caimanes y aves (crácidos). Estudian el impacto de la cacería para subsistencia y hacen un monitoreo a la feria de Pompeya, el cual tiene un mercado clandestino de carne de monte, 80% de la cual viene del Parque. También se comercializan animales para el mercado vivo de mascotas. Hay una vía abierta que se interna en Yasuní, la vía Maxus, la cual facilita el tráfico de biodiversidad.

## Capítulo 2

### 2.1 Objetivos

#### 2.1.1 Objetivo general

Diseñar un sistema integral de información que permite administrar inventario y localización de animales basado en identificación electrónica por radiofrecuencia (RFID) y geoposicionamiento (GPS).

### **2.1.2 Objetivos específicos**

- Diseñar una interfaz para muestreo de posiciones.
- Obtener la posición actual de la especie a seguir.
- Visualizar el recorrido de cada especie en diferentes tiempos y así como llevar un historial por tiempos.
- Implementar la comunicación por radiofrecuencias.
- Visualizar las diferentes especies que tenga los dispositivos.
- Hacer pruebas con los GPS y hacer correcciones si su recorrido no es el correcto.

## **2.2 Justificación**

Este proyecto nació de la necesidad de tener un mejor monitoreo en los zoológicos, reservas ecológicas, granjas, etc. En estos lugares normalmente no se lleva un vigilancia de cada especie y pues a veces suelen desaparecer en algunos casos no afectan mucho porque estos animales regresan pero muchas veces suele ser totalmente desaparecidos (raptados, asesinados, etc); algunas de estas especies están en peligro de extinción y es necesaria su vigilancia con alta seguridad, de allí nace nuestro proyecto.

Por qué en Chiapas casi el 90% son animales en peligro de extinción, por lo que una de las principales labores de uno de los zoológicos más conocidos en Chiapas es contribuir a la reproducción de especies amenazadas como la guacamaya roja (*Ara macao*), el zenzo (*Tayassu pecari*), el venado cabrito (*Mazama americana*), el cocodrilo de pantano (*Crocodylus moreletii*), el cocodrilo de río (*Crocodylus acutus*), el murciélago pescador (*Noctilio leporinus*), el tigrillo (*Felis wiedii*) y el mono araña (*Ateles geoffroyi*), entre otros. Teniendo en cuenta lo ya mencionado este proyecto da una gran mejora en los cuidados de las especies en peligro de extinción y así también dando salto a una innovación tecnológica utilizada para animales. En otro caso también puede ser utilizado en aspecto de manejo de ganado en algunas granjas, porque su ganado suele ser robado o muerto por alguno depredador, para

esto tendríamos este dispositivo para ubicar en tiempo real la posición de cada especie en un área limitada y así también ver su comportamiento diario mediante su recorrido.

## **2.3 Planteamiento del problema**

El Sistema Electrónico de Vigilancia de Fauna (SIVIFAU) es un proyecto realizado para tener un mejor monitoreo y vigilancia de la fauna en una reserva ecológica. Por qué en Chiapas casi el 90% son animales en peligro de extinción. Y existen algunas reservas ecológicas en los cuales se intenta preservar estas especies y también reproducir en cautiverio.

Con este podremos saber su posición actual y su recorrido que ha tenido en diferentes tiempos del día visualizando en una extensión de google maps; los animales también tendrán un límite de área la cual si es sobrepasada activara una alarma que alertará a los cuidadores del lugar. Todo esto fue implementado mediante el uso de GPS y comunicación serial vía radiofrecuencia (xbee), todo esto con llevo a crear un interfaz sencilla y fácil de manejar en la bview.

Este proyecto contempla un periodo de realización del 25 de enero al 03 de junio de 2016, en las instalaciones del Instituto Tecnológico de Tuxtla Gutiérrez que se encuentra ubicada en Carretera Panamericana kilómetro 1080, Colonia Terán, 29050 Tuxtla Gutiérrez, Chis.

Esta reporte está dividido por ciertos números de capítulos en el primer capítulo se profundizará en la problemática que originó la realización de dicho proyecto, así como también la justificación de éste, se hará mención del objetivo general y los específicos

## **2.4-Alcances y limitaciones**

### **2.4.1 Alcances**

En si nuestro alcance seria implementar el dispositivo electrónico (SIVIFAU) en personas y hacer pruebas en el Instituto Tecnológico de Tuxtla Gutiérrez, en un futuro se planea llevarse como un collar o brazalete en los animales de alguna reserva ecológica, Zoológico o en su remoto caso algún rancho para poder así realizar nuestros objetivos descritos anteriormente.

### **2.4.2 Limitaciones**

Las limitaciones de este proyecto fueron las siguientes contar con poca información de una interfaz en labview o algo parecido, elección de materiales y búsqueda de ellos, por lo que se tuvo que trabajar con algunos materiales que se nos proporcionó. Por lo cual nuestro diseño de dispositivo no es muy discreto y pues a nuestro parecer fue un poco más costoso. Además tiene la limitación de tener un perímetro menor de alcance para nuestra comunicación por radiofrecuencia (Xbee s1 pro) pues su máxima distancia de comunicación seria 1.5 km. Otra limitación fue la mala conexión de internet que nos proporcionaba el instituto porque nuestro prototipo debe trabajar con al menos 1 Mbps de forma estable.

No puede usar dentro de estructuras donde el satélite no pueda pasar la señal.

Pero nuestro parecer tiene una solución, implementar un Xbee de mayor alcance. Como se había dicho se trabajó con los materiales que se tuvo al momento por la falta de disponibilidad.

## **2.5 Caracterización del área en que se participo**

### **Instituto Tecnológico de Tuxtla Gutiérrez**

El Instituto Tecnológico de Tuxtla Gutiérrez brinda Servicios Educativos de Calidad Certificada, con la misión de formar de manera integral a profesionistas de excelencia en el campo de la ciencia y la tecnología con actitud emprendedora, respeto al ambiente y apego a los valores éticos.

Somos una institución de excelencia en la educación superior tecnológica del sureste, comprometidos con el desarrollo sustentable de la región, el cual es uno de los ejes transversales de nuestra oferta educativa, tanto a nivel licenciatura como posgrado.

Trabajamos con Sistemas de Gestión Ambiental Certificados, en donde nuestros estudiantes aplican sus conocimientos en función del cuidado del ambiente: aire, agua, suelo, flora, fauna y seres humanos.

La sociedad del nuevo milenio se enfrenta a retos y cambios continuos, choques generacionales en el uso de modernas tecnologías, comunicándonos en tiempo real a cualquier parte del mundo con video y voz, estamos ante un sinnúmero de avances tecnológicos que son punto de partida para modelar el desarrollo de la sociedad.

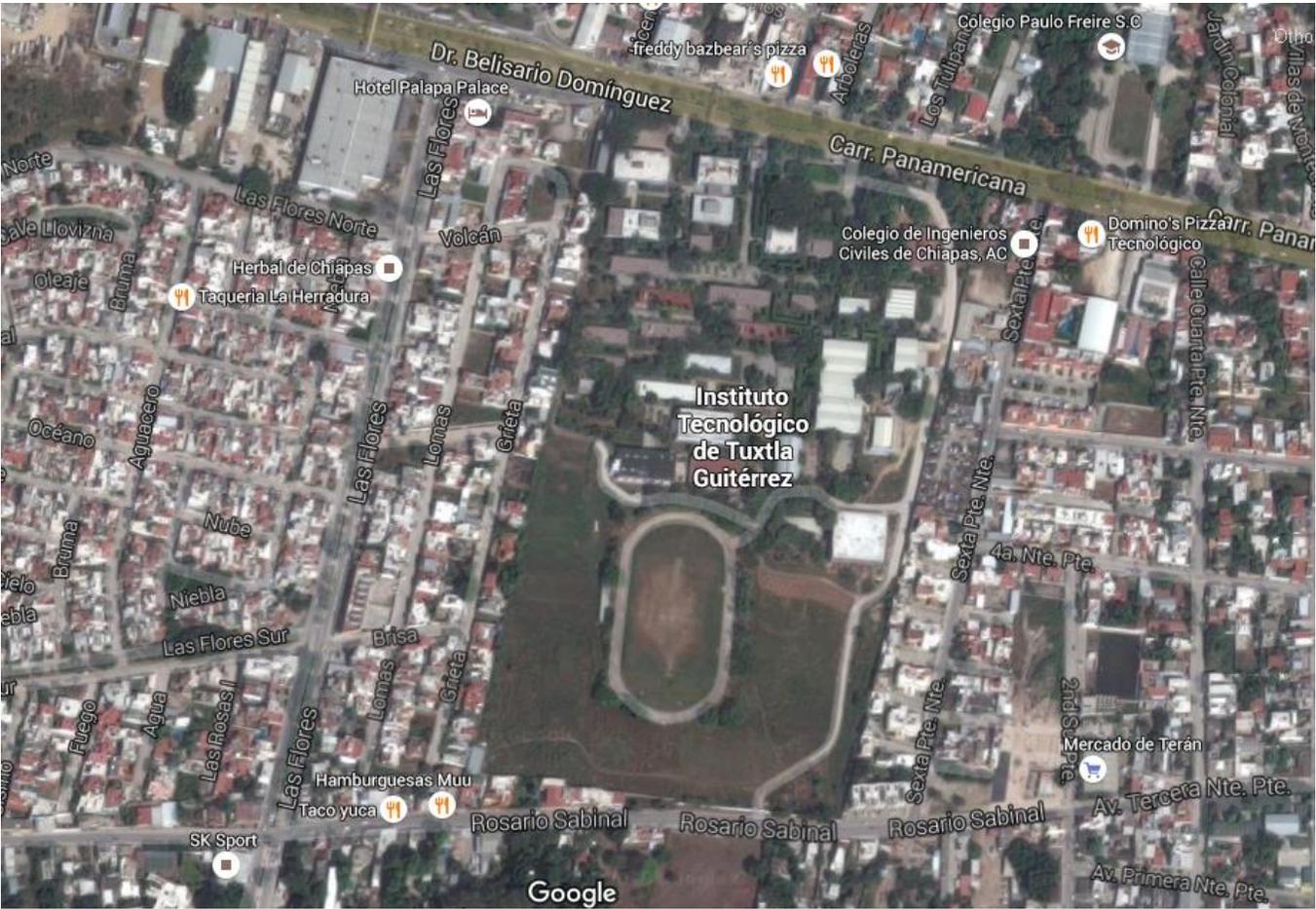
La búsqueda de un desarrollo que permita satisfacer las necesidades básicas y las aspiraciones de bienestar de la población, sin comprometer la posibilidad de las generaciones futuras de atender a sus propias necesidades y aspiraciones, se ha integrado a la ideología de nuestra institución para enriquecer las decisiones con visión de futuro.

Nosotros somos congruentes con la misión que hemos diseñado es por ellos que nos hemos ocupado por Educar con Responsabilidad Ambiental, comprometiéndonos con un movimiento interinstitucional que genere la

participación de todos los actores del instituto para con el desarrollo sustentable de la entidad.

Entre las principales líneas estratégicas de Educar con Responsabilidad Ambiental se encuentran los Sistemas de Gestión de Calidad, Sistema de Gestión Ambiental, Programas Académicos Acreditados por COPAES y por CONACYT; se trata de transformar conciencias, evolucionar culturas, sensibilizar a la población y mostrar responsabilidad ante las acciones que realizamos en pro de la conservación de la biodiversidad del planeta.

### 2.5.1 Localización



**Organigrama de la Institución (Instituto Tecnológico de Tuxtla Gutiérrez)**

<b>NOMBRE Y CARGO</b>	<b>EXT.</b>	<b>CORREO ELECTRÓNICO @ITTG.EDU.MX</b>
<b>M.E.H. José Luis Méndez Navarro</b> Director del I.T.T.G.	<b>101</b>	director@ittg.edu.mx
<b>MC. José Ángel Zepeda Hernández</b> Subdirector Académico	<b>301</b>	subacadem@ittg.edu.mx
<b>Ing. Rodrigo Ferrer González</b> Subdirector de Planeación y Vinculación	<b>401</b>	subplanea@ittg.edu.mx
<b>C.P. María Elidía Castellanos Morales</b> Subdirectora de Servicios Administrativos	<b>201</b>	subadmon@ittg.edu.mx
Lic. Marcela del Carmen Osorio Gómez <b>Jefa del Depto. de Recursos Materiales y Servicios</b>	<b>202</b>	materiales@ittg.edu.mx
M.C. Galdino Belizario Nango Solís <b>Jefe del Centro de Cómputo</b>	<b>205</b>	computo@ittg.edu.mx
Ing. Lisandro Gutiérrez González <b>Jefe del Depto. de Mantenimiento de Equipo</b>	<b>207</b>	mantto@ittg.edu.mx
M.F. Dalila Brisceyda Cantoral Díaz	<b>209</b>	finanzas@ittg.edu.mx

<b>Jefa del Depto. de Recursos Financieros</b>		
Lic. María Elisa Jiménez Ramírez <b>Jefa del Depto. de Recursos Humanos</b>	<b>210</b>	humanos@ittg.edu.mx
Ing. Juan José Arreola Ordaz <b>Jefe de la División de Estudios Profesionales</b>	<b>303</b>	division@ittg.edu.mx
<b>Coordinación de Inglés</b>	<b>305</b>	
M.C. Roció Farrera Alcázar <b>Enlace Institucional de Transparencia y Representante de la Dirección del Sistema de Gestión de la Calidad del Instituto Tecnológico de Tuxtla Gutiérrez</b>	<b>306</b>	sgcittg@ittg.edu.mx
Ing. Gilberto Hernández Cruz <b>Jefe del Depto. de Ciencias Básicas</b>	<b>307</b>	basicas@ittg.edu.mx
Ing. Delina Culebro Farrera <b>Jefa del Depto. de Desarrollo Académico</b> <b>Enlace Institucional de ANFEI</b>	<b>308</b>	desacad@ittg.edu.mx
Ing. Luis Alberto Pérez Lozano <b>Jefe Depto. de Ing. Eléctrica y Electrónica</b> <b>Enlace Institucional de ANUIES y UNIVERSIA</b>	<b>311</b> <b>312</b>	eleyeca@ittg.edu.mx

M.C. Carlos Venturino de Coss Pérez <b>Jefe del Depto. de Ing. Industrial</b>	<b>313</b>	industrial@ittg.edu.mx
Ing. José Manuel Rasgado Bezares <b>Jefe del Depto. de Metal-mecánica</b>	<b>315</b>	mecanica@ittg.edu.mx
Ing. Rene Cuesta Díaz <b>Jefe del Depto. de Ing. Química y Bioquímica</b>	<b>316</b>	quimica@ittg.edu.mx
M.C. Francisco de Jesús Suárez Ruíz <b>Jefe del Depto. de Ingeniería en Sistemas Computacionales</b>	<b>319</b>	sistemas@ittg.edu.mx
M.C. Roberto Cruz Gordillo <b>Jefe del Depto. Económico – Administrativo</b>	<b>320</b>	economico@ittg.edu.mx
Dr. Samuel Enciso Sáenz <b>Jefe de la División de Estudios Investigación y Posgrado</b>	<b>325</b>	posgrado@ittg.edu.mx
M.P.O. Alba Mercedes Mijangos Ocegüera <b>Coordinación General de Educación a Distancia del ITTG</b>	<b>329</b>	distancia@ittg.edu.mx
Ing. Jorge Antonio Orozco Torres <b>Jefe del Depto. de Ingenierías</b>	<b>320</b>	
Ing. Salomón Velasco Bermúdez <b>Jefe del Depto. de Activ. Extraescolares</b>	<b>402</b>	extraescolar@ittg.edu.mx

Lic. René Arjón Castro <b>Jefe del Centro de Información</b>	<b>403</b>	biblioteca@ittg.edu.mx
Dra. Ana Erika Pérez Galindo <b>Jefa del Depto. de Comunicación y Difusión</b>	<b>404</b>	difusion@ittg.edu.mx
Lic. Higinio García Mendoza <b>Jefe del Depto. Gestión Tecnológica y Vinculación</b>	<b>405</b>	gestion@ittg.edu.mx
M.C. Lidya Margarita Blanco González <b>Jefa del Depto. de Planeación, Programación y Presupuestación</b>	<b>406</b>	planea@ittg.edu.mx
Ing. Jorge Antonio Mijangos López <b>Jefe del Depto. de Servicios Escolares</b>	<b>407</b>	descolares@ittg.edu.mx

## 2.5.2 Misión, visión, valores

**Misión:** Formar de manera integral profesionistas de excelencia en el campo de la ciencia y la tecnología con actitud emprendedora, respeto al medio ambiente y apego a los valores éticos.

**Visión:** Ser una institución de excelencia en la educación superior tecnológica del sureste, comprometida con el desarrollo socioeconómico sustentable de la región.

### Valores:

El ser humano

El espíritu de servicio

El liderazgo

El trabajo en equipo

La calidad

El alto desempeño

Respeto al medio ambiente

### **2.5.3 Descripción del área donde se realizó el proyecto**

El lugar donde se trabajo fue en el Instituto Tecnológico de Tuxtla Gutiérrez esta cuenta con diferentes área de trabajo como una empresa (dirección general, administración, recursos humanos y financieros, y las respectivas áreas de las diferentes ingenierías ofertadas por el mismo instituto). El área en donde se realizó el proyecto es el departamento de ingeniería eléctrica - electrónica, en dicha área se cuenta con una oficina que se está enfrente de la sala de juntas de la academia del departamento de ingeniería electrónica, en esta área se contó con el espacio necesario para solucionar, instalar e innovar lo relacionado al proyecto.

En el departamento de ingeniería electrónica se realizó la investigación y la elaboración de la interfaz de usuario del proyecto en LabVIEW, por otras partes allí mismo se trabajó con prototipo del dispositivo y se hicieron pruebas saliendo del edificio haciendo ciertos recorridos en todo el instituto.

# Capítulo 3

## 3.1 Fundamento teórico

### ¿Qué es un GPS?

El GPS es un sistema de posicionamiento por satélites desarrollado por el Departamento de la Defensa de los E.U., diseñado para apoyar los requerimientos de navegación y posicionamiento precisos con fines militares. En la actualidad es una herramienta importante para aplicaciones de navegación, posicionamientos de puntos en tierra, mar y aire.

El GPS está integrado por tres segmentos o componentes de un sistema, que a continuación se describen:

#### a) Segmento espacial

El Sistema de Posicionamiento Global (GPS) es una constelación de satélites de navegación que orbitan la Tierra a una altitud de cerca de 12.000 millas (20.000 kilómetros). A esta altitud, los satélites completan dos órbitas en un poco menos de un día. Aunque originalmente diseñado por el Departamento de Defensa de EE.UU. para aplicaciones militares, su gobierno federal hizo el sistema disponible para usos civiles y levantó las medidas de seguridad diseñadas para restringir la precisión hasta 10 metros.

La constelación óptima consiste en 21 satélites operativos con 3 de "repuesto". A partir de julio de 2006, había 29 satélites operacionales de la constelación.

#### Señales GPS

Los satélites del GPS transmiten dos señales de radio de baja potencia, llamadas "L1" y "L2". Cada señal GPS contiene tres componentes de información: un código pseudoaleatorio, los datos de efemérides de satélite y datos de almanaque. El código pseudoaleatorio identifica al satélite que transmite su señal. Los datos de efemérides de satélite proporcionan información sobre la ubicación del satélite en cualquier momento. El almanaque contiene información sobre el estado del satélite

y la fecha y hora actuales. Para cada satélite, el tiempo es controlado por los relojes atómicos a bordo que son cruciales para conocer su posición exacta.

### Determinación de Posiciones del GPS

Las posiciones se obtienen mediante la determinación de las distancias a los satélites visibles. Este proceso se conoce como "trilateración". El momento de la transmisión de la señal en el satélite se compara con el momento de la recepción en el receptor. La diferencia de estos dos tiempos nos dice cuánto tiempo tomó para que la señal viajara desde el satélite al receptor. Si se multiplica el tiempo de viaje por la velocidad de la luz, podemos obtener el rango, o de distancia, con el satélite. La repetición del proceso desde tres satélites permite determinar una posición de dos dimensiones en la Tierra (es decir, la longitud y latitud). Un cuarto satélite es necesario para determinar la tercera dimensión, es decir la altura. Cuantos más satélites son visibles, más precisa es la posición del punto a determinar. Las órbitas de los satélites GPS están inclinadas respecto al ecuador de la Tierra en alrededor de  $55^\circ$ . La distribución espacial de la constelación de satélites permite al usuario disponer de 5 a 8 satélites visibles en cualquier momento. El sistema está diseñado para asegurar que al menos cuatro satélites estarán visibles con una recepción configurada de la señal de  $15^\circ$  sobre el horizonte en un momento dado, en cualquier parte del mundo.

Aunque el GPS puede dar posiciones muy precisas, aún hay fuentes de error. Estos incluyen los errores del reloj, los retrasos atmosféricos, sin saber exactamente dónde están los satélites en sus órbitas, las señales que se refleja de los objetos en la superficie de la Tierra, e incluso la degradación intencionada de la señal del satélite.

### Satélite GPS de constelación NAVSTAR

#### b) Segmento de control

Es una serie de estaciones de rastreo, distribuidas en la superficie terrestre que continuamente monitorea a cada satélite analizando las señales emitidas por estos

y a su vez, actualiza los datos de los elementos y mensajes de navegación, así como las correcciones de reloj de los satélites.

Las estaciones se ubican estratégicamente cercanas al plano ecuatorial y en todas se cuenta con receptores con relojes de muy alta precisión.

Posición de las estaciones de seguimiento y la estación principal de control

### c) Segmento usuario

Lo integran los receptores GPS que registran la señal emitida por los satélites para el cálculo de su posición tomando como base la velocidad de la luz y el tiempo de viaje de la señal, así se obtienen las pseudodistancias entre cada satélite y el receptor en un tiempo determinado, observando al menos cuatro satélites en tiempo común; el receptor calcula las coordenadas X, Y, Z y el tiempo.

### **El GPS utilizado en este proyecto fue GTPA010 GPS (FGPMMOPA6B)**

#### Descripción

El FGPMMOPA6B POT es un ultra-compacto (Patch Aparte) Módulo GPS.

Este receptor GPS POT ofrece una solución con alto contenido de posición y velocidad actuaciones de precisión, con altas capacidades de sensibilidad y de seguimiento en las condiciones de las zonas urbanas. El chipset GPS en el interior del módulo se alimenta por MediaTek Inc., proveedor líder de soluciones de medios digitales del mundo y la mayor compañía de fab- IC en Taiwán. El módulo puede soportar hasta 66 canales, y está diseñada para dispositivo de factor de forma pequeño. Es adecuado para todas las aplicaciones relacionadas con el GPS, tales como:

- Gestión de Flotas / seguimiento de activos
- LBS (servicios de localización - base) y del sistema AVL
- Sistema de Seguridad
- Dispositivo de mano para el posicionamiento de navegación personal y viajes

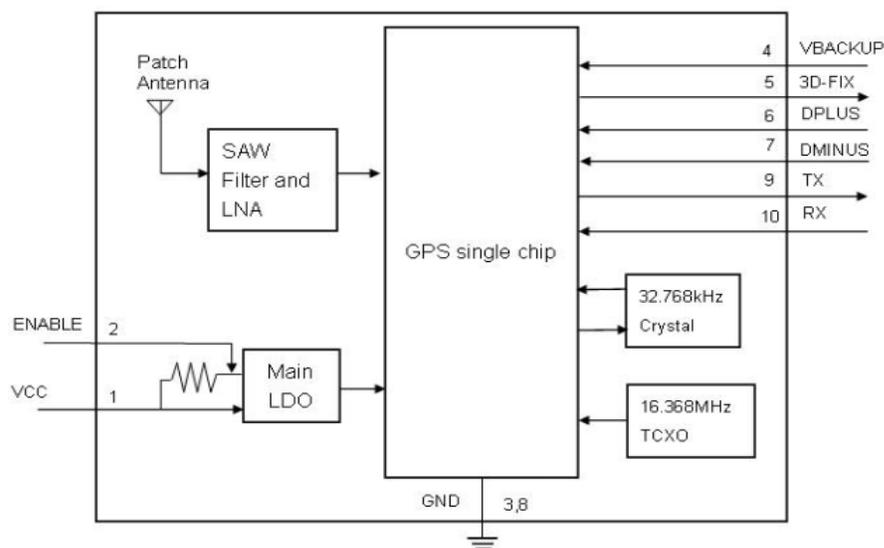
## Características

- MediaTek chip único
- Dimensiones: 16 mm x 16 mm x 6 mm
- Antena Patch Tamaño: 15 mm x 15 mm x 4 mm
- Frecuencia L1, código C / A, 66 canales
- Embedded filtro de LNA y vieron con antena de parche activo
- Alta Sensibilidad: hasta -163 dBm seguimiento, actuaciones urbanas superiores
- Exactitud de la posición: <CEP 3m (50%) y sin SA (horizontal)
- Arranque en frío tiene menos de 35 segundos (típico)
- Arranque en caliente es menos de 34 segundos (típico)
- Hot Start es menos de 1 segundo (típico)
- Bajo consumo de energía: 48mA @ adquisición, 37mA @ seguimiento
- Bajo apagado consumo de corriente: 15uA, DGPS típica (WAAS, EGNOS, MSAS) de apoyo (opcional por el firmware)
- Max. Velocidad de actualización de hasta 10 Hz
- Interfaz USB / UART
- TIPO SMD
- función AGPS apoyo (modo fuera de línea: EPO válida hasta 14 días)
- RoHS
- 

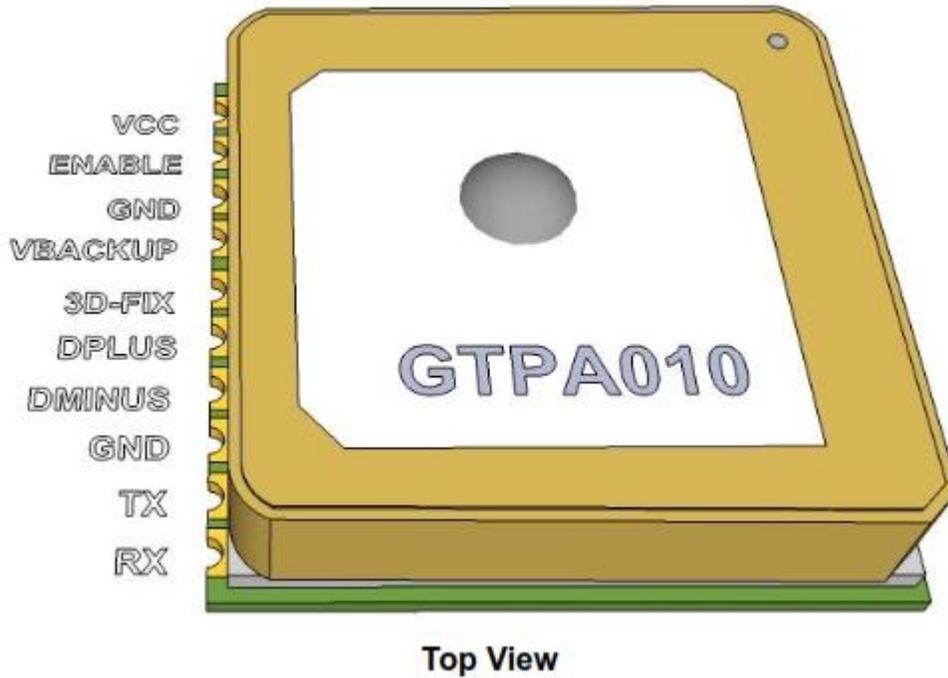
## Configuración por defecto

- 9600 baudios
- 1Hz
- Standart salida NMEA (GGA, RMC, GSA)

## Sistema a bloques



## Configuración de pines



## Pin Definition

Pin	Name	I/O	Description
1	VCC	PI	Main DC power input
2	ENABLE	I	High active, or keep floating for normal working
3	GND	P	Ground
4	VBACKUP	PI	Backup power input
5	3D-FIX	O	3D-fix indicator
6	DPLUS	I/O	USB port D+
7	DMINUS	I/O	USB port D-
8	GND	P	Ground
9	TX	O	Serial data output of NMEA
10	RX	I	Serial data input for firmware update

## Descripción del pin I/O

### VCC (Pin1)

La principal fuente de alimentación de DC del módulo, la tensión debe estar entre 3,2 V a 5,0 V. El Vcc rizado debe ser controlado bajo 50mV (típico: 3,3V)

### ENABLE (Pin2)

Mantener abiertos o pulsar ON para encender. Pulsar de LOW para apagar el módulo.

**Enable (High):**  $1.8V \leq V_{enable} \leq VCC$

**Disable (Low):**  $0V \leq V_{enable} \leq 0.25V$

### GND (Pin3)

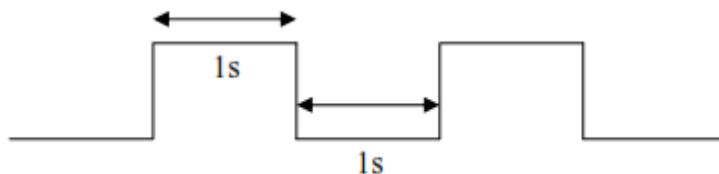
Tierra

### VBACKUP (Pin4)

Este es el poder para GPS chipset RTC para mantener funcionando cuando se retira la alimentación principal. La tensión debe estar entre 2,0V~4,3 V. (Típico: 3,0V) El PIN debe estar conectado durante el funcionamiento normal.

### 3D-FIX (Pin5)

El 3D-FIX fue asignado como arreglar la bandera de salida. Si no se utilizan, mantener flotando. Antes para 2D-FIX, El pasador debe de salida continuamente un segundo nivel alto con una segunda señal de bajo nivel.



Después de 2D or 3D-FIX

El pasador debe continuamente la salida de señal de bajo nivel.

XBEE PRO S1

### **Especificación XBee PRO S1**

-Interior/Rango urbano hasta 300 pies. (90 m), hasta 200 pies (60 m) variante internacional

-RF exterior línea de vista con alcance hasta 1 milla (1600 m), hasta 2500 pies (750 m) transmiten energía salida (seleccionable por software) 63mW (18dBm)\* 10mW (10 dBm) por variante internacional.

RF Data Rate 250.000 bps bps

-Interfaz Serial Data Rate (seleccionable por software) 1200 bps - 250 kbps (no estándar también admite velocidades en baudios) para la variante internacional 1200 bps - 250 kbps

-Sensibilidad del receptor de -100 dBm (1% tasa de error de paquetes)

### **Requisitos de alimentación**

Voltaje de alimentación 2.8 – 3.4 V

Corriente de transmisión (típico) 250mA (@3,3 V) (150mA para la variante internacional) RPSMA módulo sólo: 340mA (@3,3 V) (180mA para la variante internacional)

Idle / Receive Current (typical): 55mA (@ 3.3 V)

Caida de tension < 10  $\mu$ A

### **General**

Frecuencia de operación ISM 2.4 GHz

Dimensiones 0.960" x 1.297" (2.438cm x 3.294cm)

Temperatura de operación 0.960" x 1.297" (2.438cm x 3.294cm)

## Redes y seguridad

Topologías de red compatibles Punto-a-punto, punto-multipunto.

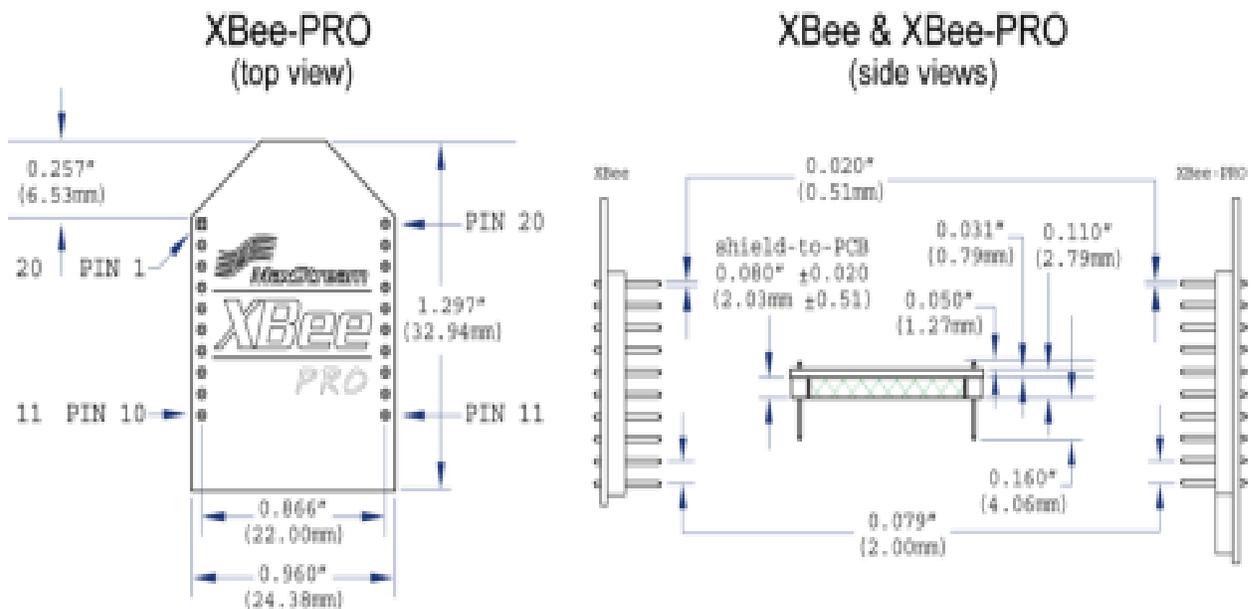
Número de canales: 12 canales de secuencia directa

Opciones de direccionamiento PAN ID, Canal y dirección

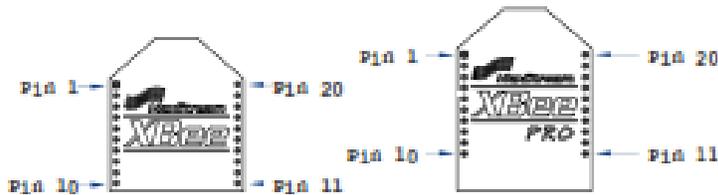
## Homologaciones

United States (FCC Part 15.247)	OUR-XBEEPRO
Industry Canada (IC)	4214A XBEEPRO
Europe (CE)	ETSI (Max. 10 dBm transmit power output)*
Japan	R201WW08215111 (Max. 10 dBm transmit power output)*
Australia	C-Tick

## Dibujos mecánicos



## Pines



Pin #	Name	Direction	Description
1	VCC	-	Power supply
2	DOUT	Output	UART Data Out
3	DIN / CONFIG	Input	UART Data In
4	DO8*	Output	Digital Output 8
5	RESET	Input	Module Reset (reset pulse must be at least 200 ns)
6	PWM0 / RSSI	Output	PWM Output 0 / RX Signal Strength Indicator
7	PWM1	Output	PWM Output 1
8	[reserved]	-	Do not connect
9	DTR / SLEEP_RQ / DI8	Input	Pin Sleep Control Line or Digital Input 8
10	GND	-	Ground
11	AD4 / DIO4	Either	Analog Input 4 or Digital I/O 4
12	CTS / DIO7	Either	Clear-to-Send Flow Control or Digital I/O 7
13	ON / SLEEP	Output	Module Status Indicator
14	VREF	Input	Voltage Reference for A/D Inputs
15	Associate / AD5 / DIO5	Either	Associated Indicator, Analog Input 5 or Digital I/O 5
16	RTS / AD6 / DIO6	Either	Request-to-Send Flow Control, Analog Input 6 or Digital I/O 6
17	AD3 / DIO3	Either	Analog Input 3 or Digital I/O 3
18	AD2 / DIO2	Either	Analog Input 2 or Digital I/O 2
19	AD1 / DIO1	Either	Analog Input 1 or Digital I/O 1
20	AD0 / DIO0	Either	Analog Input 0 or Digital I/O 0

Notas de diseño:

- El número mínimo de conexiones: VCC, GND, DOUT y DIN
- El número mínimo de conexiones para la actualización del firmware: VCC, GND, DIN, DOUT, RTS
- DTR se especifica la dirección de la señal con respecto al módulo • Módulo incluye un 50k  $\Omega$  resistencia pull-up conectada a restablecer
- Varios de los pull-ups de entrada puede configurarse usando el comando PR
- Los PINES sin utilizar debe quedar desconectado

## Características eléctricas

Características en DC ( $V_{CC} = 2.8 - 3.4$  VDC)

Symbol	Characteristic	Condition	Min	Typical	Max	Unit
$V_{IL}$	Input Low Voltage	All Digital Inputs	-	-	$0.35 \cdot V_{CC}$	V
$V_{IH}$	Input High Voltage	All Digital Inputs	$0.7 \cdot V_{CC}$	-	-	V
$V_{OL}$	Output Low Voltage	$I_{OL} = 2$ mA, $V_{CC} \geq 2.7$ V	-	-	0.5	V
$V_{OH}$	Output High Voltage	$I_{OH} = -2$ mA, $V_{CC} \geq 2.7$ V	$V_{CC} - 0.5$	-	-	V
$I_{IN}$	Input Leakage Current	$V_{IN} = V_{CC}$ or GND, all inputs, per pin	-	0.025	1	$\mu$ A
$I_{OZ}$	High Impedance Leakage Current	$V_{IN} = V_{CC}$ or GND, all I/O High-Z, per pin	-	0.025	1	$\mu$ A
TX	Transmit Current	$V_{CC} = 3.3$ V	-	45 (XBee)	215, 140 (PRO, Int)	mA
RX	Receive Current	$V_{CC} = 3.3$ V	-	50 (XBee)	55 (PRO)	mA
PWR-DWN	Power-down Current	SM parameter = 1	-	< 10	-	$\mu$ A

## Características ADC (funcionamiento)

Symbol	Characteristic	Condition	Min	Typical	Max	Unit
$V_{REFH}$	VREF - Analog-to-Digital converter reference range		2.08	-	$V_{DDAD}$	V
$I_{REF}$	VREF - Reference Supply Current	Enabled	-	200	-	$\mu$ A
		Disabled or Sleep Mode	-	< 0.01	0.02	$\mu$ A
$V_{INDC}$	Analog Input Voltage <sup>1</sup>		$V_{SSAD} - 0.3$	-	$V_{DDAD} + 0.3$	V

## Distribución de ADC/Características de rendimiento

Symbol	Characteristic	Condition	Min	Typical	Max	Unit
$R_{AS}$	Source Impedance at Input <sup>2</sup>		-	-	10	k $\Omega$
$V_{AIN}$	Analog Input Voltage <sup>3</sup>		$V_{REFL}$		$V_{REFH}$	V
RES	Ideal Resolution (1 LSB) <sup>4</sup>	$2.08V \leq V_{DDAD} \leq 3.6V$	2.031	-	3.516	mV
DNL	Differential Non-linearity <sup>5</sup>		-	$\pm 0.5$	$\pm 1.0$	LSB
INL	Integral Non-linearity <sup>6</sup>		-	$\pm 0.5$	$\pm 1.0$	LSB
$E_{ZS}$	Zero-scale Error <sup>7</sup>		-	$\pm 0.4$	$\pm 1.0$	LSB
$F_{FS}$	Full-scale Error <sup>8</sup>		-	$\pm 0.4$	$\pm 1.0$	LSB
$E_{IL}$	Input Leakage Error <sup>9</sup>		-	$\pm 0.05$	$\pm 5.0$	LSB
$E_{TU}$	Total Unadjusted Error <sup>10</sup>		-	$\pm 1.1$	$\pm 2.5$	LSB

## 2. Funcionamiento del módulo RF

### Comunicación serial

El XBee®/XBee-PRO® módulos RF interfaz a un dispositivo host a través de una lógica a nivel de puerto serie asíncrono. A través de su puerto serie, el módulo puede comunicarse con cualquier lógica y tensión UART compatible; o a través de un traductor de nivel a cualquier dispositivo de serie (por ejemplo: Mediante un Digi patentada interfaz RS-232 o USB board).

### Flujo de datos de UART

Los dispositivos que tienen una interfaz UART se puede conectar directamente a las patillas del módulo RF como se muestra en la figura siguiente

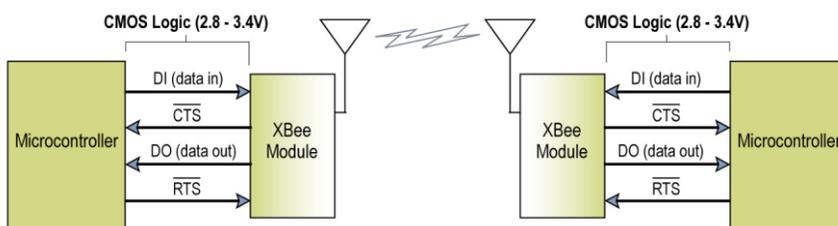
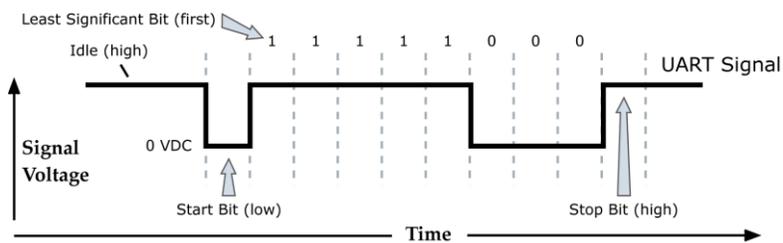


Diagrama de flujo de datos del sistema en un entorno interconectado de UART (Low-afirmó señales distinguido con línea horizontal sobre el nombre de la señal).

## Serial data

Entra en el módulo de datos a través de la UART DI PIN (pin 3) como una señal serie asíncrona. La señal debe ralentí alto cuando no se están transmitiendo datos. Cada byte de datos consta de un bit de inicio (baja), 8 bits de datos (primero el bit menos significativo) y un bit de parada (alta). La siguiente figura ilustra el patrón de bits de la serie de datos que pasa a través del módulo.

UART de paquetes de datos 0x1F (número decimal "31") transmitidos a través del módulo RF Ejemplo Formato de datos es 8-n-1 (bits de paridad - - Número de bits de parada)



Comunicaciones serie dependen de los dos UARTs (el microcontrolador's y el módulo RF's) para ser configurados con valores compatibles (velocidad en baudios, paridad, bits de inicio, bits de datos, bits de parada).

## Funcionamiento transparente

De forma predeterminada, XBee-PRO® RF módulos funcionan en modo transparente. Cuando se selecciona este modo de funcionamiento, los módulos actúan como una línea serie UART sustitución - todos los datos recibidos a través de la DI pin se ponen en cola para la transmisión de RF. Cuando se reciben datos de RF, los datos se envían al hacer la patilla.

## Serial-a-RF paquetización

Los datos se almacenan en el DI búfer hasta que una de las causas siguientes los datos y paquetes transmitidos:

1. No se reciben los caracteres de serie durante el tiempo que determine la RO (Paquetización parámetro Timeout). Si  $R_o = 0$ , paquetización comienza cuando se recibe un carácter.
2. El número máximo de caracteres que caben en un paquete de RF (100) es recibido.
3. El modo de comando de secuencia (GT + CC + GT) es recibido. Cualquier carácter almacenado en el búfer antes de DI la secuencia se transmite.

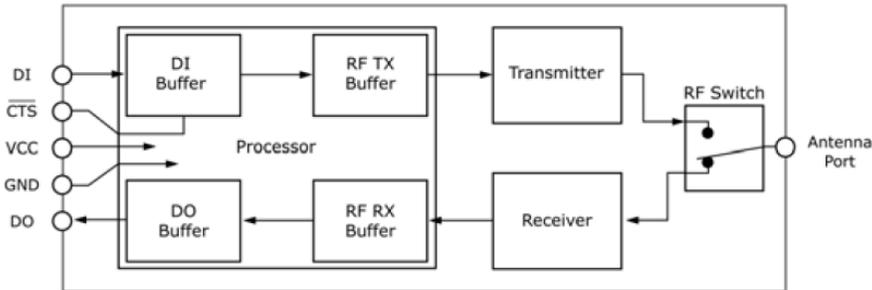
## Operación API

Interfaz de programación de aplicaciones (API) es una alternativa a la predeterminada funcionamiento transparente. La trama se extiende la API basada en el grado en que una aplicación host puede interactuar con las capacidades de networking del módulo. Cuando en el modo API, todos los datos que entran y salen del módulo está contenida en las tramas que definen las operaciones o eventos dentro del módulo. Transmitir tramas de datos (recibida a través de la DI PIN (pin 3) incluyen:

- Marco de datos de transmisión de RF
- comando Frame (equivalentes a los comandos AT) reciben tramas de datos (enviado el PIN (pin 2) incluyen:
  - RF-recibió la trama de datos
  - respuesta de comando notificaciones de eventos tales como restablecer, asociar desasociar, etc.

La API proporciona medios alternativos de configuración de módulos y el enrutamiento de datos en la capa de la aplicación host. Una aplicación host puede enviar tramas de datos para el módulo que contienen la dirección y la información del paquete en lugar de utilizar el modo comando para modificar las direcciones.

## Control de flujo



DI (datos en búfer de datos serie) cuando entra en el módulo RF a través de la DI PIN (pin 3), los datos se almacenan en el búfer hasta di puede ser procesada. Control de flujo por hardware (CTS ). Cuando el DI buffer es de 17 bytes lejos de ser plena; por defecto, el módulo de la aserción de CTS (alta) a la señal al dispositivo host para detener el envío de datos [consultar D7 (DIO7) Parámetro de configuración]. CTS es re-afirmó después de la DI memoria intermedia tiene 34 bytes de memoria disponible.

Cómo eliminar la necesidad de control de flujo:

1. Enviar mensajes que son más pequeños que el tamaño del búfer de DI (202 bytes).
2. Interfaz a una menor tasa de baudios [BD (Interfaz Data Rate)] que el parámetro de velocidad de datos de rendimiento. Caso en que el DI búfer puede ser plena y posiblemente desbordamiento: si el módulo está recibiendo un flujo continuo de datos RF, cualquier serie de datos que llega a la patilla DI se coloca en el buffer.

Los datos de la DI buffer será transmitida a través del aire cuando el módulo ya no está recibiendo datos de RF en la red.

## **DO (Data Out) Buffer**

Cuando se reciben datos de RF, los datos entran en el búfer y se envían por el puerto serie a un dispositivo host. Una vez que el buffer no alcanza su capacidad máxima, cualquier entrante adicional se pierden datos RF.

Control de flujo por hardware (RTS). Si RTS está habilitado para el control de flujo (D6 (DIO6 Configuración) Parámetro = 1), los datos no se enviarán el buffer mientras RTS (pin 16) se confirmó.

Dos casos en los que el búfer no puede ser plena y posiblemente desbordamiento:

1. Si la velocidad de datos RF es mayor que la tasa de datos de la interfaz del módulo, el módulo va a recibir datos desde el módulo de transmisión más rápido de lo que puede enviar los datos al host.
2. Si el host no permite que el módulo para transmitir datos desde el búfer de hacerlo porque se mantuvo fuera de control de flujo de hardware o software.

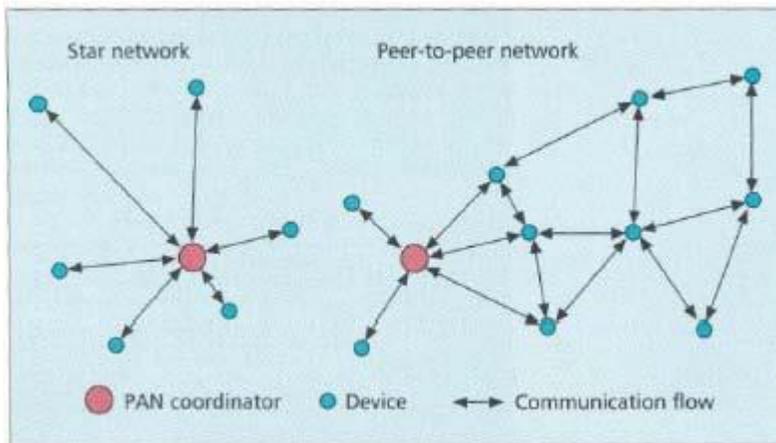
## **Estándar IEEE 802.15.4.**

Es el estándar utilizado por el Xbee PRO S1, las características más importantes en este estándar son su flexibilidad de red, bajos costos, bajo consumo de energía; este estándar se puede utilizar para muchas aplicaciones en el hogar que requieren una tasa baja en la transmisión de datos.

Capas de red.

En las redes tradicionales por cable, la capa de red es responsable por la topología de construcción y mantenimiento de la misma, así como de nombrarla y de los servicios de enlace que incorpora las tareas necesarias de direccionamiento y seguridad. Estos mismos servicios existen para redes inalámbricas para el hogar, sin embargo representan un reto mayor por la primicia de ahorro de energía. Las redes que se construyan dentro de esta capa del estándar IEEE 802.15.4 se espera que se auto organicen y se auto mantengan en funcionamiento con lo que se pretende reducir los costos totales para el consumidor.

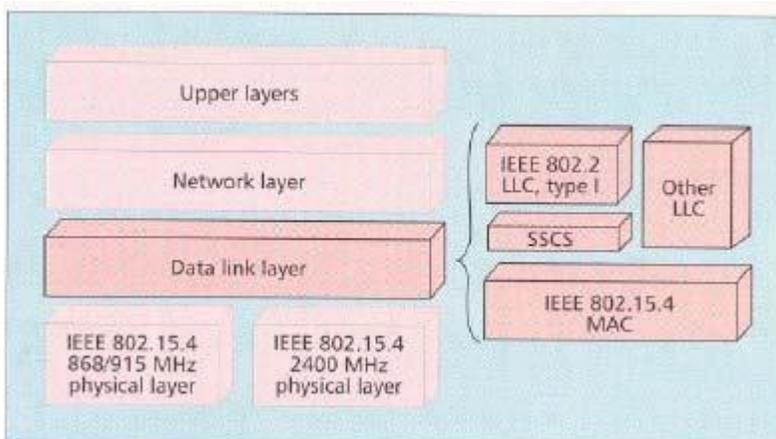
El estándar IEEE 802.15.4 soporta múltiples topologías para su conexión en red, entre ellas la topología tipo estrella y la topología peer-to-peer (Fig 45). La topología a escoger es una elección de diseño y va a estar dado por la aplicación a la que se desee orientar; algunas aplicaciones como periféricos e interfaces de PC, requieren de conexiones de baja potencia de tipo estrella, mientras que otros como los perímetros de seguridad requieren de una mayor área de cobertura por lo que es necesario implementar una red peer-to-peer.



Red tipo estrella y red punto a punto.

Capa de enlace de datos (data link layer, DLL).

La forma en que el estándar IEEE 802.15.4 se basa en la organización internacional para la estandarización (ISO) del modelo de referencia para la interconexión de sistemas abiertos (OSI).



Las características del MAC IEEE 802.15.4 son; la asociación y la disociación, reconocimientos de entrega de trama, mecanismos de acceso al canal, validación de trama, garantía del manejo de las ranuras de tiempo, y manejo de guías. Las sub capas MAC proporcionan dos tipos de servicios hacia capas superiores que se acceden a través de dos puntos de acceso a servicios (SAPs). Los servicios de datos MAC se acceden por medio de la parte común de la sub capa (MCPS-SAP), y el manejo de servicios MAC se accede por medio de la capa MAC de manejo de identidades (MLME-SAP). Esos dos servicios proporcionan una interface entre las sub capas de convergencia de servicios específicos (SSCS) u otro LLC y las capas físicas.

El administrador de servicios MAC tiene 26 primitivas. Comparadas con el 802.15.1 (bluetooth), que tiene alrededor de 131 primitivas en 32 eventos, el MAC 802.15.4 es muy simple, haciéndolo muy versátil para las aplicaciones hacia las que fue orientado, aunque se paga el costo de tener un instrumento con características menores a las del 802.15.1 (por ejemplo el 802.15.4 no soporta enlaces sincronizados de voz).

## **Configuración del módulo RF**

### Programación del módulo RF

Los ejemplos de programación en esta sección requieren la instalación de Digi's X-CTU Software y una conexión serie a un PC. (Digi stocks de RS-232 y USB juntas para facilitar la interfaz con un PC.)

1. Instalar Digi's X-CTU Software para un PC haciendo doble clic en el "setup\_X-CTU.exe" archivo. (El archivo está ubicado en el CD y Digi)
2. Monte el módulo RF a una placa de interfaz, a continuación, conecte el módulo general a un PC.
3. Lanzar el X-CTU Software y seleccione la 'Configuración de la PC' ficha. Verifique la velocidad de transmisión y paridad settings del puerto Com coinciden con los del módulo RF. Nota: la falla para entrar en el modo de comando se debe

comúnmente a la velocidad en baudios no coincide. Asegurar la "Baudios" en la pestaña "Configuración de la PC" coincide con el tipo de datos de la interfaz del módulo RF. Por defecto, el parámetro BD = 3 (que corresponde a 9600 bps).

Ejemplo de configuración: Modificar la dirección de destino del módulo RF

Ejemplo: Utilizar el X-CTU "Terminal" ficha para cambiar el módulo RF's DL (dirección de destino bajo) Parámetro y guardar la nueva dirección en la memoria no volátil. Después de establecer una conexión de serie entre el módulo RF y un PC [consulte la 'Configuración' sección supra], seleccione la ficha "Terminal" de la X-CTU Software e introduzca el siguiente comando líneas ('CR' significa un retorno de carril):

Método 1 (una línea por cada comando)

<b>Envío Comando AT</b>	<b>Respuesta del sistema</b>
+++	OK <CR> (Enter into Command Mode)
ATDL <Enter>	{current value} <CR> (Read Destination Address Low)
ATDL1A0D <Enter>	OK <CR> (Modify Destination Address Low)
ATWR <Enter>	OK <CR> (Write to non-volatile memory)
ATCN <Enter>	OK <CR> (Exit Command Mode)

Método 2 (una línea por cada comando)

Envió Comando AT	Respuesta del sistema
+++	OK <CR> (Enter into Command Mode)
ATDL <Enter>	{current value} <CR> (Read Destination Address Low)
ATDL1A0D,WR,CN <Enter>	OK<CR> OK<CR> OK<CR>

Ejemplo de configuración: Restaurar valores predeterminados del módulo RF

Ejemplo: Utilizar el X-CTU "Módem" en la ficha de configuración para restaurar los valores predeterminados de parámetros. Después de establecer una conexión entre el módulo y un PC seleccione la pestaña "Configuración de módem" de la X-CTU Software.

1. Seleccione el 'Read' el botón.
2. Seleccione el 'Restore' el botón.

Configuración remota de comandos.

La API de firmware ha dispuesto de enviar comandos de configuración remota de dispositivos utilizando el comando remoto Solicitud de bastidor (véase API API operación). Este marco de la API se puede utilizar para enviar comandos a un módulo remoto para leer o establecer los parámetros de comando.

La API de firmware ha dispuesto para enviar comandos de configuración (o leer) a un módulo remoto utilizando el comando remoto Solicitud de bastidor (véase API API de operaciones). Comandos remotos pueden ser emitidos para leer o establecer los parámetros de comando en un dispositivo remoto.

Envío de un comando remoto

Para enviar un comando remoto, la trama de solicitud de Comando Remoto debe estar rellena con valores de 64 bits y 16 bits de direcciones. Si el direccionamiento de 64 bits es deseado entonces el campo de dirección de 16 bits debe llenarse con 0xFFFFE. Si el valor es cualquier valor distinto de 0xFFFFE se utiliza en el campo de dirección de 16 bits entonces el campo de dirección de 64 bits será ignorado y direccionamiento de 16 bits será utilizado. Si un comando es la respuesta deseada, el Frame ID debe estar configurado con un valor distinto de cero.

Aplicar cambios en remoto.

Cuando se utilizan los comandos remotos para cambiar los parámetros de comando en un dispositivo remoto, el parámetro cambios no surtirá efecto hasta que se apliquen los cambios. Por ejemplo, cambiando el parámetro BD no cambiará el tipo de interfaz de serie real en el control remoto hasta que se apliquen los cambios.

Respuestas de los comandos remotos

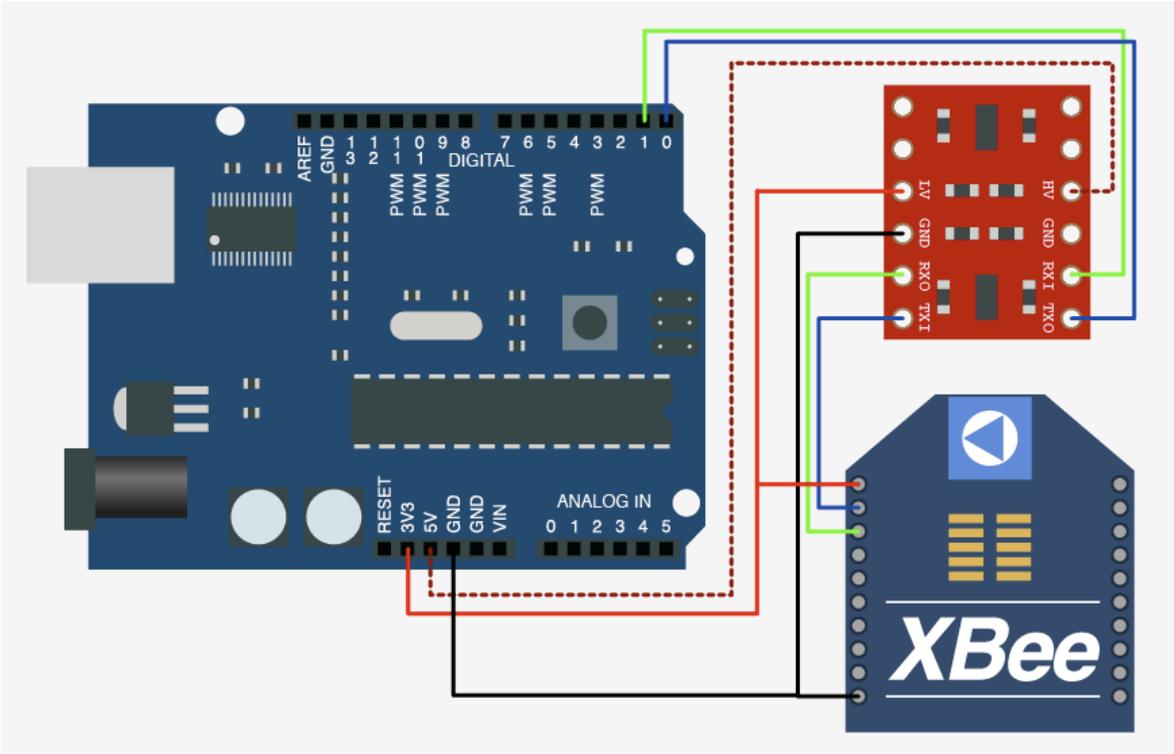
Si el dispositivo remoto recibe una petición de comando remoto, la transmisión y el bastidor API ID es distinto de cero, el remoto enviará un comando remoto para la transmisión de respuesta del dispositivo que envía el comando remoto. Cuando un comando remoto de respuesta se recibe la transmisión, un dispositivo envía un comando remoto respuesta marco API su UART. El comando remoto de respuesta indica el estado del comando (el éxito, o el motivo del fallo), y en el caso de un comando query, incluirá el valor del registro.

El dispositivo que envía un comando remoto no recibirá una trama de respuesta de comando remoto: si el dispositivo de destino no podía ser alcanzado el identificador de trama en la solicitud de comando remoto está establecido a 0.

Conexión básica con Arduino.

Las versiones UNO y MEGA tienen además un pin con tensión de salida regulada a 3,3 voltios y el microcontrolador en los tres casos posee una UART TTL a 5 voltios para proporcionarnos comunicaciones serie. Teniendo en cuenta todo esto, no podremos conectar directamente los cables Tx y Rx de nuestro Arduino al módulo Xbee sin correr el peligro de freírlo, ya que este funciona con una tensión de 3,3 voltios. Así que tendremos que utilizar un convertidor de niveles lógicos para adaptar las tensiones de la interfaz serie del microcontrolador (5v) al voltaje requerido por el módulo Xbee (3,3v).

(Diagrama de conexiones Arduino UNO y Xbee).



## ARDUINO

Es una plataforma de prototipos electrónica de código abierto (open-source) basada en hardware y software flexibles y fáciles de usar. Está pensado para artistas, diseñadores, como hobby y para cualquiera interesado en crear objetos o entornos interactivos. Arduino puede sentir el entorno mediante la recepción de entradas desde una variedad de sensores y puede afectar a su alrededor mediante el control de luces, motores y otros artefactos. El microcontrolador de la placa se programa usando el Arduino Programming Language (basado en Wiring) y el Arduino Development Environment (basado en Processing). Los proyectos de Arduino pueden ser autónomos o se pueden comunicar con software en ejecución en un ordenador (por ejemplo con Flash, Processing, MaxMSP, etc.). Las placas se pueden ensamblar a mano o encargarse pre ensambladas; el software se puede descargar gratuitamente. Los diseños de referencia del hardware (archivos CAD) están disponibles bajo licencia open-source, por lo que eres libre de adaptarlas a tus necesidades. 28

Hay muchos otros microcontroladores y plataformas microcontroladoras disponibles para computación física. Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, y muchas otras ofertas de funcionalidad similar. Todas estas herramientas toman los desordenados detalles de la programación de microcontrolador y la encierran en un paquete fácil de usar. Arduino también simplifica el proceso de trabajo con microcontroladores, pero ofrece algunas ventajas para profesores, estudiantes y aficionados interesados sobre otros sistemas:

- Barato: Las placas arduino son relativamente baratas comparadas con otras plataformas microcontroladoras. La versión menos cara del módulo arduino puede ser ensamblada a mano, e incluso los módulos de arduino pre ensamblados cuestan menos de 50\$.
- Multiplataforma: El software de arduino se ejecuta en sistemas operativos Windows, Macintosh OSX y GNU/Linux. La mayoría de los sistemas microcontroladores están limitados a Windows.

- Entorno de programación simple y claro: El entorno de programación de arduino es fácil de usar para principiantes, pero suficientemente flexible para que usuarios avanzados puedan aprovecharlo también. Para profesores, está convenientemente basado en el entorno de programación processing, de manera que estudiantes aprendiendo a programar en ese entorno estarán familiarizados con el aspecto y la imagen de arduino.
- Código abierto y software extensible: El software arduino está publicado como herramientas de código abierto, disponible para extensión por programadores experimentados. El lenguaje puede ser expandido mediante librerías C++, y la gente que quiera entender los detalles técnicos pueden hacer el salto desde arduino a la programación en lenguaje AVR C en el cual está basado. De forma similar, puedes añadir código AVR-C directamente en tus programas arduino si quieres.
- Código abierto y hardware extensible: El arduino está basado en microcontroladores ATMEGA8 y ATMEGA168 de Atmel. Los planos para los módulos están publicados bajo licencia Creative Commons, por lo que diseñadores experimentados de circuitos pueden hacer su propia versión del módulo, extendiéndolo y mejorándolo. Incluso usuarios relativamente inexpertos pueden construir la versión de la placa del módulo para entender cómo funciona y ahorrar dinero. Ejemplo del Arduino UNO figura 2.5.



Figura 2.5 ejemplo de la tarjeta Arduino UNO

## **TinyGPS++**

TinyGPS++ es una nueva biblioteca de Arduino para analizar secuencias de datos NMEA proporcionada por módulos GPS. Al igual que su predecesor, TinyGPS, esta biblioteca proporciona compacta y fácil de usar métodos para la extracción de posición, fecha, hora, altitud, velocidad y rumbo de los dispositivos GPS. Sin embargo, TinyGPS++'s Programmer Interface es considerablemente más sencillo de utilizar que TinyGPS y la nueva biblioteca puede extraer datos arbitrarios de cualquiera de las innumerables sentencias NMEA hacia fuera allí, incluso los propietarios.

## **LabVIEW**

LabVIEW (acrónimo de Laboratory Virtual Instrumentation Engineering Workbench) es una plataforma y entorno de desarrollo para diseñar sistemas, con un lenguaje de programación visual gráfico. Recomendado para sistemas hardware y software de pruebas, control y diseño, simulado o real y embebido, pues acelera la productividad. El lenguaje que usa se llama lenguaje G, donde la G simboliza que es lenguaje Gráfico.

Este programa fue creado por National Instruments (1976) para funcionar sobre máquinas MAC, salió al mercado por primera vez en 1986. Ahora está disponible para las plataformas Windows, UNIX, MAC y GNU/Linux. La penúltima versión es la 2013, con la increíble demostración de poderse usar simultáneamente para el diseño del firmware de un instrumento RF de última generación, a la programación de alto nivel del mismo instrumento, todo ello con código abierto. Y posteriormente la versión 2014 disponible en versión demo para estudiantes y profesional, la versión demo se puede descargar directamente de la página National Instruments.

Los programas desarrollados con LabVIEW se llaman Instrumentos Virtuales, o VIs, y su origen provenía del control de instrumentos, aunque hoy en día se ha expandido ampliamente no sólo al control de todo tipo de electrónica

(Instrumentación electrónica) sino también a su programación embebida, comunicaciones, matemáticas, etc. Un lema tradicional de LabVIEW es: "La potencia está en el Software", que con la aparición de los sistemas multinúcleo se ha hecho aún más potente. Entre sus objetivos están el reducir el tiempo de desarrollo de aplicaciones de todo tipo (no sólo en ámbitos de Pruebas, Control y Diseño) y el permitir la entrada a la informática a profesionales de cualquier otro campo. LabVIEW consigue combinarse con todo tipo de software y hardware, tanto del propio fabricante -tarjetas de adquisición de datos, PAC, Visión, instrumentos y otro Hardware- como de otros fabricantes.

### **¿Para qué puedo usar LabVIEW?**

#### **Adquirir Datos y Procesar Señales**

1. Mida cualquier sensor en cualquier bus
2. Realice análisis y procesamiento de señales avanzados
3. Visualice datos en interfaces de usuario personalizadas
4. Registre datos y genere reportes

#### **Control de Instrumentos**

1. Automatice la colección de datos
2. Controle múltiples instrumentos
3. Analice y visualice señales

#### **Automatizar Sistemas de Pruebas y Validación**

1. Automatice las pruebas de validación y producción de su producto
2. Controle múltiples instrumentos

3. Analice y visualice resultados de pruebas con interfaces de usuario personalizadas

### **Sistemas Embebidos de Monitoreo y Control**

1. Reutilice código ANSI C y HDL
2. Integre hardware comercial
3. Genere prototipos con tecnología FPGA
4. Obtenga acceso a herramientas personalizadas para medicina, robótica y más

### **Enseñanza Académica**

1. Utilice un enfoque práctico e interactivo de aprendizaje
2. Combine el diseño de algoritmos con medidas de datos reales
3. Aumente el rendimiento de aplicaciones con procesamiento multinúcleo

### **Generación de Prototipos**

1. Generación de prototipos de algoritmos para SDR
2. Arquitectura de desarrollo para LTE y 802.11
3. Entorno de desarrollo simple para FPGAs

## Google Static Maps

Google Static Maps API te permite insertar una imagen de Google Maps en tu página web sin la necesidad de usar JavaScript ni otro tipo de carga de página dinámica. El servicio Google Static Maps API crea tu mapa en función de los parámetros de dirección URL que hayas enviado a través de una solicitud HTTP estándar y devuelve el mapa en forma de imagen, que puedes mostrar en tu página web.

Google Static Maps API devuelve una imagen (GIF, PNG o JPEG) en respuesta a una solicitud HTTP a través de una dirección URL. Para cada solicitud, puedes especificar la ubicación del mapa, el tamaño de la imagen, el nivel de zoom, el tipo de mapa y la inclusión de marcadores opcionales en diferentes ubicaciones del mapa. También puedes etiquetar tus marcadores usando caracteres alfanuméricos.

Una imagen de Google Static Maps API se inserta en el atributo src de la etiqueta <img>, o su equivalente en otros lenguajes de programación. Si se usa una imagen de Google Static Maps API fuera de una aplicación web (como en un navegador), se deberá incluir un vínculo que conduzca a la ubicación que se muestra en un navegador web o en la aplicación nativa de Google Maps. (Los usuarios de Google Maps API for Work están exentos de este requisito). Consulta la sección 10.1.1 (h), Condiciones de servicio de Google Maps/Google Earth, para obtener el lenguaje completo y actual acerca de este requisito.

## Guía del desarrollador de Google Static Maps

Google Static Maps API te permite insertar una imagen de Google Maps en tu página web sin la necesidad de usar JavaScript ni otro tipo de carga de página dinámica. El servicio Google Static Maps API crea tu mapa en función de los parámetros de dirección URL que hayas enviado a través de una solicitud HTTP

estándar y devuelve el mapa en forma de imagen, que puedes mostrar en tu página web.

Nota: Los Límites de uso de Google Static Maps API han cambiado. Crear una clave de API e incluirla en tu solicitud te permite realizar un seguimiento del uso en Google Developers Console, y comprar cuota adicional, si fuera necesario.

En este documento se detalla Google Static Maps API v2. Para actualizar tus direcciones URL v1, consulta la Guía de actualización.

A continuación, verás un ejemplo rápido

El siguiente ejemplo contiene la dirección URL de una imagen de Google Static Maps API del centro de la ciudad de Nueva York, que se muestra a continuación:

```
https://maps.googleapis.com/maps/api/staticmap?center=Brooklyn+Bridge,New+York,NY&zoom=13&size=600x300&maptype=roadmap
```

```
&markers=color:blue%7Clabel:S%7C40.702147,-
```

```
74.015794&markers=color:green%7Clabel:G%7C40.711614,-74.012318
```

```
&markers=color:red%7Clabel:C%7C40.718217,-73.998284
```

```
&key=YOUR_API_KEY
```

Puntos de interés en la parte sur de Manhattan

No necesitas hacer nada en “especial” para que esta imagen aparezca en la página. No es necesario usar JavaScript. Todo lo que necesitamos hacer es crear una

dirección URL y colocarla en una etiqueta <img> Puedes colocar una Google Static Maps API de Google en cualquier parte de tu página web donde puedas insertar una imagen.

## Público

Este documento está orientado a desarrolladores de sitios web y móviles que quieran incluir imágenes de Google Static Maps API en una página web o una aplicación móvil. Proporciona una introducción al uso de la API y material de referencia acerca de los parámetros disponibles.

## Información general

Google Static Maps API devuelve una imagen (GIF, PNG o JPEG) en respuesta a una solicitud HTTP a través de una dirección URL. Para cada solicitud, puedes especificar la ubicación del mapa, el tamaño de la imagen, el nivel de zoom, el tipo de mapa y la inclusión de marcadores opcionales en diferentes ubicaciones del mapa. También puedes etiquetar tus marcadores usando caracteres alfanuméricos.

Una imagen de Google Static Maps API se inserta en el atributo src de la etiqueta <img>, o su equivalente en otros lenguajes de programación. Si se usa una imagen de Google Static Maps API fuera de una aplicación web (como en un navegador), se deberá incluir un vínculo que conduzca a la ubicación que se muestra en un navegador web o en la aplicación nativa de Google Maps. (Los usuarios de Google Maps API for Work están exentos de este requisito). Consulta la sección 10.1.1(h), Condiciones de servicio de Google Maps/Google Earth, para obtener el lenguaje completo y actual acerca de este requisito.

## Parámetros de dirección URL

Una dirección URL de Google Static Maps API debe respetar el siguiente formato:

<https://maps.googleapis.com/maps/api/staticmap?parameters>

Si a tu sitio web se accede a través de HTTPS, también debes cargar las imágenes de Google Static Maps API a través de HTTPS para evitar alertas de seguridad del navegador. HTTPS también se recomienda si tus solicitudes incluyen información confidencial, como la ubicación del usuario:

<https://maps.googleapis.com/maps/api/staticmap?parameters>

#### Parámetros de ubicación

- **center** (obligatorio si no hay marcadores presentes) define el centro del mapa, equidistante de todos los bordes del mapa. Este parámetro toma una ubicación como un par de {latitud,longitud} separado por comas (p. ej., "40.714728,-73.998672") o como una cadena de dirección (p. ej., "city hall, new york, ny") para identificar una ubicación única sobre la faz de la tierra. Para obtener más información, consulta Ubicaciones a continuación.
- **zoom** (obligatorio si no hay marcadores presentes) define el nivel de zoom del mapa, que determina el nivel de amplificación del mapa. Este parámetro toma un valor numérico que corresponde al nivel de zoom de la región deseada. Para obtener más información, consulta niveles de zoom a continuación.

#### Parámetros de mapas

- **size** (obligatorio) define las dimensiones rectangulares de la imagen del mapa. Este parámetro toma una cadena de forma {horizontal\_value}x{vertical\_value}. Por ejemplo, 500x400 define un mapa de 500 píxeles de ancho por 400 píxeles de alto. Los mapas más pequeños que

180 píxeles de ancho mostrarán un logotipo de Google reducido. Este parámetro se ve afectado por el parámetro `scale`, que se describe a continuación; el tamaño de salida final es el producto de los valores de tamaño y escala.

- `scale` (opcional) afecta la cantidad de píxeles devuelta. `scale=2` devuelve el doble de píxeles que `scale=1`, pero conserva la misma área de cobertura y nivel de detalle (es decir, el contenido del mapa no cambia). Esto resulta útil al desarrollar para visualizaciones en alta resolución o al generar un mapa para impresión. El valor predeterminado es 1. Los valores aceptados son 2 y 4 (4 solo está disponible para clientes de Google Maps API for Work). Para obtener más información, consulta [Valores de escala](#).
- `format` (opcional) define el formato de la imagen resultante. De forma predeterminada, Google Static Maps API crea imágenes PNG. Hay varios formatos posibles, entre ellos, GIF, JPEG y PNG. El formato que uses depende de cómo quieras presentar la imagen. JPEG generalmente proporciona una mayor compresión, mientras que GIF y PNG proporcionan más detalles. Para obtener más información, consulta [Formatos de imagen](#).
- `maptype` (opcional) define el tipo de mapa que se construirá. Existen diferentes valores de `maptype` posibles, e incluyen `roadmap`, `satellite`, `hybrid` y `terrain`. Para obtener más información, consulta [Maptypes de Google Static Maps API](#) a continuación.
- `language` (opcional) define el idioma que se usará para mostrar etiquetas en los mosaicos de mapa. Ten en cuenta que este parámetro solo es compatible con algunos mosaicos de países; si el idioma específico solicitado no es compatible con el conjunto de mosaicos, se usará el idioma predeterminado para ese conjunto de mosaicos.
- `region` (opcional) define los límites adecuados para mostrar, en función de sensibilidades geopolíticas. Acepta un código de región especificado como un valor ccTLD de dos caracteres ('dominio de nivel superior').

## Parámetros de características

- **markers** (opcional) define uno o más marcadores para adjuntar a la imagen en las ubicaciones especificadas. Este parámetro toma la definición de un solo marcador con parámetros separados por el carácter de barra vertical (|). Se pueden incluir múltiples marcadores en el mismo parámetro **markers** siempre que exhiban el mismo estilo; podrías necesitar agregar más marcadores de diferentes estilos agregando parámetros **markers** adicionales. Ten en cuenta que si proporcionas marcadores para un mapa, no es necesario que especifiques los parámetros (generalmente obligatorios) **center** y **zoom**. Para obtener más información, consulta [Marcadores de Google Static Maps API](#) a continuación
- **path** (opcional) define un solo trayecto para dos o más puntos conectados para superponer en la imagen en las ubicaciones especificadas. Este parámetro toma una cadena de definiciones de puntos separados por el carácter de barra vertical (|). Puedes proporcionar más trayectos al agregar parámetros **path** adicionales. Ten en cuenta que si proporcionas un trayecto para un mapa, no es necesario que especifiques los parámetros (generalmente obligatorios) **center** y **zoom**. Para obtener más información, consulta [Trayectos de Google Static Maps API](#) a continuación.
- **visible** (opcional) especifica una o más ubicaciones que deben permanecer visibles en el mapa, aunque no se mostrarán marcadores ni otros indicadores. Usa este parámetro para garantizar que se muestren ciertas características o ubicaciones del mapa en [Google Static Maps API](#).
- **style** (opcional) define un estilo personalizado para alterar la presentación de una característica específica (calle, parque, etc.) del mapa. Este parámetro toma argumentos **feature** y **element** que identifican las características que se seleccionarán y un conjunto de operaciones de estilo para aplicar a esa selección. Puedes proporcionar múltiples estilos al agregar parámetros **style**

adicionales. Para obtener más información, consulta Mapas con estilos a continuación.

### Parámetros de clave y firma

- **key** (obligatorio) te permite controlar el uso de la API que realiza tu aplicación en Google Developers Console; habilita límites de cuota basados en la clave en lugar de en la dirección IP y garantiza que Google pueda comunicarse contigo acerca de tu aplicación, si fuera necesario. Para acceder a más información, consulta [Obtener una clave y una firma](#).
- **signature** (recomendado) es una firma digital que se usa para verificar que los sitios que generen solicitudes con tu clave de API estén autorizados para hacerlo. Para acceder a más información, consulta [Obtener una clave y una firma](#).

### Restricción de tamaño para direcciones URL

Las direcciones URL de la Google Static Maps API están restringidas a un tamaño aproximado de 2048 caracteres. En la práctica, probablemente no necesites direcciones URL más extensas, a menos que generes mapas complicados con una gran cantidad de marcadores y trayectos. No obstante, ten en cuenta que ciertos caracteres pueden ser codificados para direcciones URL por navegadores o servicios antes de enviarlos a la API, lo que produce un mayor uso de caracteres. Para obtener más información, consulta [Crear una dirección URL válida](#).

### Especificación de ubicaciones

Google Static Maps API debe poder identificar con precisión ubicaciones en el mapa, tanto para centrar el mapa en la ubicación correcta (usando el parámetro `center`) como para colocar marcas de posición opcionales (usando el parámetro `markers`) en diferentes ubicaciones del mapa. Google Static Maps API usa números

(valores de latitud y longitud) o cadenas (direcciones) para especificar esas ubicaciones. Esos valores identifican una ubicación geocodificada.

Muchos parámetros (como markers y path) incluyen múltiples ubicaciones. En esos casos, las ubicaciones se separan con el carácter de barra vertical (|)

### Latitudes y longitudes

Las latitudes y longitudes se definen con numerales dentro de una cadena de texto separada por comas que tiene una precisión de 6 decimales. Por ejemplo, "40.714728,-73.998672" es un geocode válido. La precisión superior a 6 decimales se ignora.

Los valores de longitud se basan en la distancia desde Greenwich, Inglaterra, donde se encuentra el primer meridiano. Dado de Greenwich se encuentra en la latitud 51.477222, podemos ingresar un valor center de 51.477222,0 para centrar el mapa en Greenwich:

### Guía del desarrollador de Google Static Maps

Google Static Maps API te permite insertar una imagen de Google Maps en tu página web sin la necesidad de usar JavaScript ni otro tipo de carga de página dinámica. El servicio Google Static Maps API crea tu mapa en función de los parámetros de dirección URL que hayas enviado a través de una solicitud HTTP estándar y devuelve el mapa en forma de imagen, que puedes mostrar en tu página web.

Nota: Los Límites de uso de Google Static Maps API han cambiado. Crear una clave de API e incluirla en tu solicitud te permite realizar un seguimiento del uso en Google Developers Console, y comprar cuota adicional, si fuera necesario.

En este documento se detalla Google Static Maps API v2. Para actualizar tus direcciones URL v1, consulta la Guía de actualización.

A continuación, verás un ejemplo rápido

El siguiente ejemplo contiene la dirección URL de una imagen de Google Static Maps API del centro de la ciudad de Nueva York, que se muestra a continuación:

```
https://maps.googleapis.com/maps/api/staticmap?center=Brooklyn+Bridge,New+York,NY&zoom=13&size=600x300&maptype=roadmap
```

```
&markers=color:blue%7Clabel:S%7C40.702147,-
```

```
74.015794&markers=color:green%7Clabel:G%7C40.711614,-74.012318
```

```
&markers=color:red%7Clabel:C%7C40.718217,-73.998284
```

```
&key=YOUR_API_KEY
```

Puntos de interés en la parte sur de Manhattan

No necesitas hacer nada en “especial” para que esta imagen aparezca en la página. No es necesario usar JavaScript. Todo lo que necesitamos hacer es crear una dirección URL y colocarla en una etiqueta <img> Puedes colocar una Google Static Maps API de Google en cualquier parte de tu página web donde puedas insertar una imagen.

Parámetros de dirección URL

Una dirección URL de Google Static Maps API debe respetar el siguiente formato:

<https://maps.googleapis.com/maps/api/staticmap?parameters>

Si a tu sitio web se accede a través de HTTPS, también debes cargar las imágenes de Google Static Maps API a través de HTTPS para evitar alertas de seguridad del navegador. HTTPS también se recomienda si tus solicitudes incluyen información confidencial, como la ubicación del usuario:

<https://maps.googleapis.com/maps/api/staticmap?parameters>

Nota: Ten en cuenta que Google Static Maps API no admite direcciones URL de iconos personalizados que usen HTTPS; se mostrará el icono predeterminado.

Ya sea que uses HTTP o HTTPS, ciertos parámetros de las direcciones URL son obligatorios y otros opcionales. Como es norma en las direcciones URL, todos los parámetros se separan con el carácter de Y comercial (&). A continuación, se proporciona una lista de los parámetros y sus posibles valores.

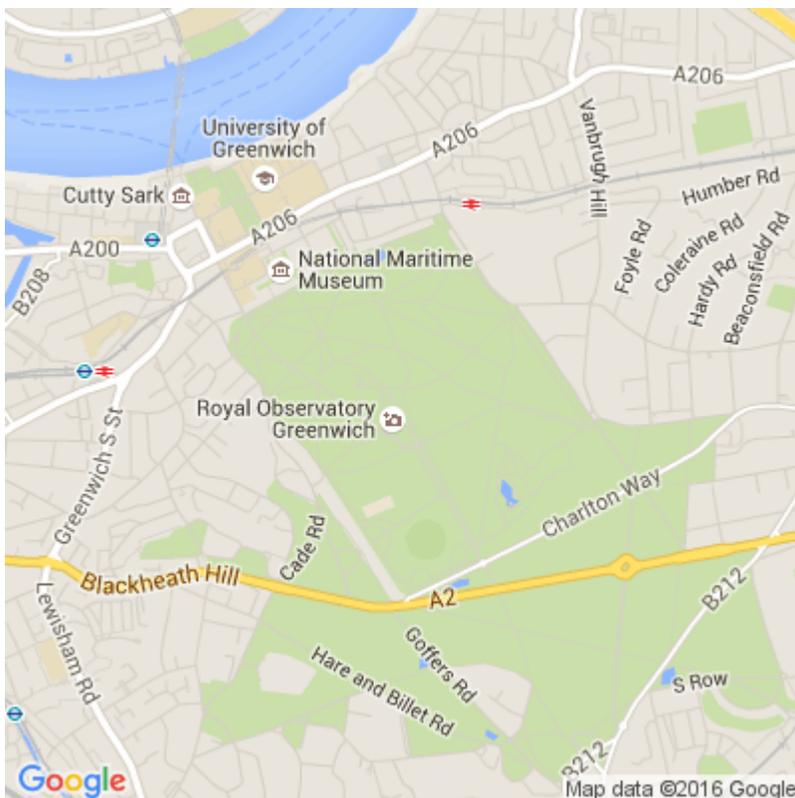
Importante: La siguiente discusión acerca de los parámetros de las direcciones URL usa ejemplos que, por motivos de claridad, se proporcionan en el formato previo al escape. Antes de enviar una solicitud a la API, sus parámetros se deben codificar como dirección URL de forma correcta. Por ejemplo, muchos parámetros usan un carácter de barra vertical (|) como separador, que debe codificarse como %7C en la dirección URL final, tal como se muestra en el ejemplo rápido en la parte superior de este documento. Para obtener más información, consulta [Crear una dirección URL válida](#).

Google Static Maps API define imágenes de mapa con los siguientes parámetros de dirección URL:

## Latitudes y longitudes

Las latitudes y longitudes se definen con numerales dentro de una cadena de texto separada por comas que tiene una precisión de 6 decimales. Por ejemplo, "40.714728,-73.998672" es un geocode válido. La precisión superior a 6 decimales se ignora.

Los valores de longitud se basan en la distancia desde Greenwich, Inglaterra, donde se encuentra el primer meridiano. Dado de Greenwich se encuentra en la latitud 51.477222, podemos ingresar un valor center de 51.477222,0 para centrar el mapa en Greenwich:



Los valores de latitud y longitud deben corresponder a una ubicación válida sobre la superficie de la tierra. Las latitudes pueden tener cualquier valor entre -90 y 90 mientras que los valores de longitud pueden tener cualquier valor entre -180 y 180. Si especificas un valor de latitud o longitud no válido, se rechazará tu solicitud por ser incorrecta.

## Direcciones

La mayoría de las personas no hablan de latitudes y longitudes, sino que indican ubicaciones mediante direcciones. El proceso de convertir una dirección en un punto geográfico se conoce como geocodificación, y el servicio Google Static Maps API puede realizar la geocodificación por ti si proporcionas direcciones válidas.

En cualquier parámetro en el que puedas proporcionar un par de latitud/longitud, puedes especificar una cadena que indique una dirección. Google geocodificará la dirección y le proporcionará al servicio de Google Static Maps API un valor de latitud/longitud para usar en la colocación de marcadores o la especificación de ubicaciones. La cadena debe transferirse a la dirección URL, de modo que una dirección como "City Hall, New York, NY" se debe convertir en "City+Hall,New+York,NY", por ejemplo.

Ten en cuenta que las direcciones pueden reflejar ubicaciones precisas, como calles; polilíneas, como nombres de rutas o áreas poligonales, como ciudades, países o parques nacionales. Para los resultados polilineales y poligonales, el servidor Google Static Maps API usará el punto central de la línea/el área como el centro de la dirección. Si tienes dudas acerca de cómo se puede geocodificar una dirección, puedes probar la dirección usando esta Utilidad de geocodificación.

El siguiente ejemplo genera un Google Static Maps API para Berkeley, CA:

```
https://maps.googleapis.com/maps/api/staticmap?center=Berkeley,CA&zoom=14&size=400x400&key=YOUR_API
```

## Tamaños de las imágenes

El parámetro `size`, junto con `center`, define el área de cobertura de un mapa. También define el tamaño de salida del mapa en píxeles, cuando se multiplica con el valor `scale` (que, de forma predeterminada, es 1).

La siguiente tabla muestra los valores máximos permitidos para el parámetro `size` en cada valor de `scale`.

API	<code>scale=1</code>	<code>scale=2</code>	<code>scale=4</code>
Gratis	640x640	640x640 (devuelve 1280x1280 píxeles)	No disponible.
Google Maps API for Work	2048x2048	1024x1024 (devuelve 2048x2048 píxeles)	512x512 (devuelve 1024x1024 píxeles)

El ejemplo a continuación solicita una “porción” de la tierra en el Ecuador a un nivel de zoom 1:

[https://maps.googleapis.com/maps/api/staticmap?center=0,0&zoom=1&size=400x50&key=YOUR\\_API\\_KEY](https://maps.googleapis.com/maps/api/staticmap?center=0,0&zoom=1&size=400x50&key=YOUR_API_KEY)



El ejemplo a continuación solicita un mapa pequeño, de 100 x 100 píxeles centrado en la misma región. Observa el logotipo de Google más pequeño:

[https://maps.googleapis.com/maps/api/staticmap?center=0,0&zoom=1&size=100x100&key=YOUR\\_API\\_KEY](https://maps.googleapis.com/maps/api/staticmap?center=0,0&zoom=1&size=100x100&key=YOUR_API_KEY)



## Valores de escala

El parámetro `size` de Google Static Maps API define el tamaño de un mapa en píxeles, de modo que un mapa `size=200x200` se devolverá como 200 píxeles x 200 píxeles. En un monitor de computadora LCD, que generalmente muestra aproximadamente 100 píxeles por pulgada (ppp), un mapa de 200x200 medirá aproximadamente 2 pulgadas en cada dimensión.

No obstante, los dispositivos móviles incluyen cada vez con más frecuencia pantallas de alta resolución con densidades de píxeles superiores a los 300 ppp que:

- Reducen el tamaño de una imagen de 200x200 píxeles a solo 0,7 de pulgada y exhiben las etiquetas y los iconos demasiado pequeños para leerlos; o
- Aplican escala (zoom) a la imagen para mejorar la legibilidad, lo que genera una imagen poco definida o pixelada.

## Formatos de imagen

Las imágenes pueden devolverse en diferentes formatos gráficos web comunes: GIF, JPEG y PNG. El parámetro `format` admite uno de los siguientes valores:

- `png8` o `png` (predeterminado) especifica el formato PNG de 8 bits.
- `png32` especifica el formato PNG de 32 bits.
- `gif` especifica el formato GIF.
- `jpg` especifica el formato de compresión JPEG.
- `jpg-baseline` especifica un formato de compresión JPEG no progresivo.

- jpg y jpg-baseline generalmente proporcionan el tamaño de imagen más pequeño, y lo hacen mediante compresión “con pérdida”, que puede degradar la imagen. gif, png8 y png32 proporcionan una compresión con menos pérdida.

La mayoría de las imágenes JPEG son progresivas, lo que significa que cargan una imagen más tosca con anticipación y redefinen su resolución a medida que reciben más datos. Esto permite que las imágenes puedan cargarse rápidamente en páginas web y es el uso más común de JPEG en la actualidad. No obstante, algunos usos de JPEG (especialmente la impresión) requieren imágenes no progresivas (básicas). En esos casos, te recomendamos que uses el formato jpg-baseline, que es no progresivo.

## Tipos de mapas

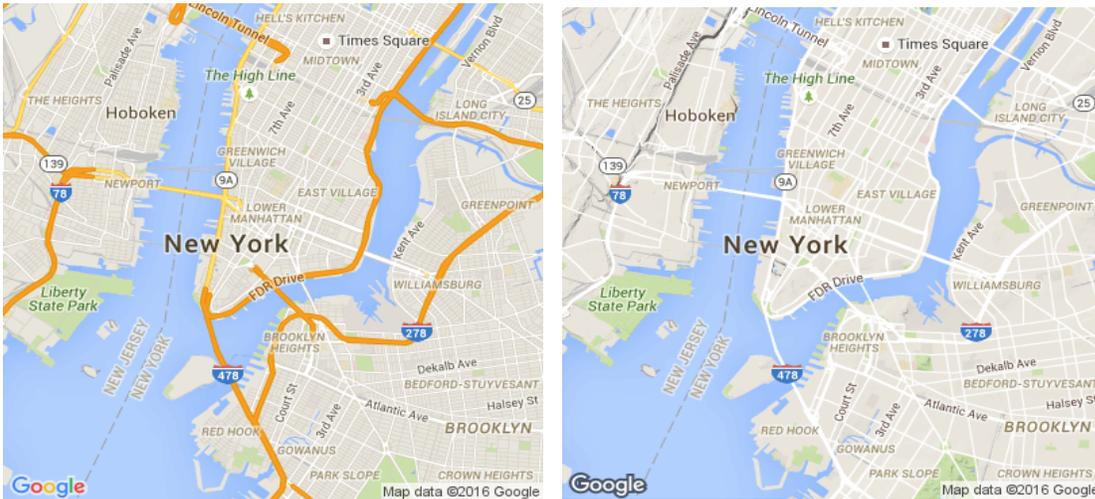
Google Static Maps API crea mapas en diferentes formatos, que se indican a continuación:

- roadmap (predeterminado), especifica una imagen de mapa de ruta estándar, como habitualmente se muestra en el sitio web de Google Maps. Si no se especifica un valor de `maptype`, Google Static Maps API proporciona mosaicos de roadmap de forma predeterminada.
- satellite especifica una imagen satelital.
- terrain especifica una imagen de un mapa de relevamiento físico que muestra terreno y vegetación.
- hybrid especifica un híbrido de la imagen satelital y del mapa de ruta, y muestra una capa transparente de las calles principales y los nombres de los sitios que aparecen en la imagen satelital.

En el siguiente ejemplo de código, puedes ver la diferencia entre los tipos mapa de ruta y terreno.

[https://maps.googleapis.com/maps/api/staticmap?center=40.714728,-73.998672&zoom=12&size=400x400&maptype=roadmap&key=YOUR\\_API\\_KEY](https://maps.googleapis.com/maps/api/staticmap?center=40.714728,-73.998672&zoom=12&size=400x400&maptype=roadmap&key=YOUR_API_KEY)

[https://maps.googleapis.com/maps/api/staticmap?center=40.714728,-73.998672&zoom=12&size=400x400&maptype=terrain&key=YOUR\\_API\\_KEY](https://maps.googleapis.com/maps/api/staticmap?center=40.714728,-73.998672&zoom=12&size=400x400&maptype=terrain&key=YOUR_API_KEY)



Los mapas híbridos usan imágenes satelitales y características prominentes del mapa de ruta para crear un mapa combinado. Los siguientes ejemplos muestran los tipos de mapa híbrido y satelital:

[https://maps.googleapis.com/maps/api/staticmap?center=40.714728,-73.998672&zoom=12&size=400x400&maptype=satellite&key=YOUR\\_API\\_KEY](https://maps.googleapis.com/maps/api/staticmap?center=40.714728,-73.998672&zoom=12&size=400x400&maptype=satellite&key=YOUR_API_KEY)

[https://maps.googleapis.com/maps/api/staticmap?center=40.714728,-73.998672&zoom=12&size=400x400&maptype=hybrid&key=YOUR\\_API\\_KEY](https://maps.googleapis.com/maps/api/staticmap?center=40.714728,-73.998672&zoom=12&size=400x400&maptype=hybrid&key=YOUR_API_KEY)



Obtener una clave y una firma

Todas las aplicaciones Google Static Maps API requieren autenticación con una clave de API. Incluir una clave en tu solicitud te permite controlar el uso de la API por parte de tu aplicación en Google Developers Console, habilita límites de cuota basados en la clave en lugar de basarse de en la dirección IP y garantiza que Google pueda comunicarse contigo acerca de tu aplicación, si fuera necesario.

Además de la clave para la API, te recomendamos que incluyas una firma digital única en la URL de la solicitud. La firma única permite a nuestros servidores verificar que los sitios que generen solicitudes con tu clave de API estén autorizados para hacerlo.

- El método de autenticación recomendado es la inclusión de una clave de navegador (clave de API) y una firma digital en todas las solicitudes a Google Static Maps API. La clave de navegador es obligatoria. La firma digital es obligatoria si habilitas la facturación de pago según el uso.

- Si usas la API estándar sin habilitar la facturación de pago por uso, necesitarás una clave de navegador. Se recomienda el uso de la firma digital, pero es opcional.
- Si compraste una licencia de Google Maps API for Work, puedes incluir un id. de cliente y una firma digital en lugar de una clave de API.

Obtén una clave de API

Para comenzar a usar Google Static Maps API, haz clic en el botón que aparece a continuación, el cual te llevará a Google Developers Console, te guiará a lo largo del proceso y activará Google Static Maps API automáticamente.

## OBTÉN UNA CLAVE

También puedes seguir estos pasos para obtener una clave de API:

1. Ingresa a Google Developers Console.
2. Crea o selecciona un proyecto.
3. Haz clic en Continue para habilitar la API.
4. Ingresa en Credentials para obtener una clave de navegador (y definir las credenciales de la API).
5. Para evitar el robo de cuota, protege tu clave de API siguiendo estas prácticas recomendadas.
6. (Opcional) Habilita la facturación. Para obtener más información, consulta Límites de uso.

### Tipos de clave de API

Las Google Maps API están disponibles para aplicaciones Android o iOS, navegadores web y a través de servicios web HTTP. Las API de cada plataforma requieren un tipo de clave específico. Google Static Maps API solo funcionará con una clave de navegador o de servidor. Las API de la misma plataforma pueden usar la misma clave.

## Firmas digitales

Las solicitudes a Google Static Maps API deben incluir un parámetro signature con una firma digital que debes generar usando un secreto de firma de URL compartido. Tu secreto compartido está disponible en Google Developers Console.

El proceso de firma usa un algoritmo de cifrado para combinar la URL y tu secreto compartido. La firma única resultante permite que nuestros servidores verifiquen que los sitios que generen solicitudes con tu clave de API estén autorizados para hacerlo.

# Capítulo 4

## 4.1 Procedimiento y descripción de las actividades realizadas

### 4.1.1 Programación en microcontrolador(Arduino)

En esta etapa se realizó lo siguiente: buscar librerías para nuestro GPS: así como hacer pruebas y verificar si los datos( coordenadas) que nos arrojaba el GPS visualizado mediante el monitor serial coincidían con nuestra posición actual.

El siguiente código se desarrolló en Arduino con la utilización de librerías como lo son TinyGPS++.h que es para procesar los datos obtenidos por el GPS y la librería SoftwareSerial.h que crea una comunicación serial por medio de software la cual nos permite comunicarlo con el GPS para poder almacenar los datos en variables utilizadas en Arduino y así poder extraer la ubicación exacta con la ayuda de la librería TinyGPS++.h.

```
#include <TinyGPS++.h> //se incluye la librería para el uso del GPS
```

```

#include <SoftwareSerial.h> //librería para crear una comunicación serial mas

int x = 0,y = 0; //variables que indican cambio de posición
float lati=0.0,lon=0.0,lati1=0.0,lon1=0.0; // variables donde se guardara la ubicación
static const int RXPin = 8, TXPin = 7; //configuración de pines para la nueva
comunicación serial

static const uint32_t GPSBaud = 9600; //se define el baud para la nueva
comunicación serial

TinyGPSPlus gps; //se define un nombre con el que se usaran los datos en la librería
TinyGPS++

SoftwareSerial ss(RXPin, TXPin); //se establecen los pines de la nueva
comunicación serial

void setup()
{
  Serial.begin(9600); //se habilita el puerto serial
  ss.begin(GPSBaud); //se habilita el segundo puerto serial
}

void loop()
{
  while (ss.available() > 0) //se pregunta si hay datos disponibles en el segundo
puerto serie que va conectado al GPS
  gps.encode(ss.read()); //si hay datos estos se colocan en la variable de la librería
TinyGPS++ para ser procesados
  readgps(); //llama a un nuevo void

  if(lati!=lati1){ //verifica si la latitud ha cambiado
    lati1=lati; //la coloca en una nueva variable
  }
}

```

```

x = 1;}      //indicador de que la latitud cambio
if(lon!=lon1){ //verifica si la longitud ha cambiado
lon1=lon;    //la coloca en una nueva variable
y = 1;}      //indicador de que la longitud cambio
if(lati1==0&&lon1==0){//verifica si no recibió coordenadas
x = 2;      //coloca nuevos valores a los indicadores
y = 2; }

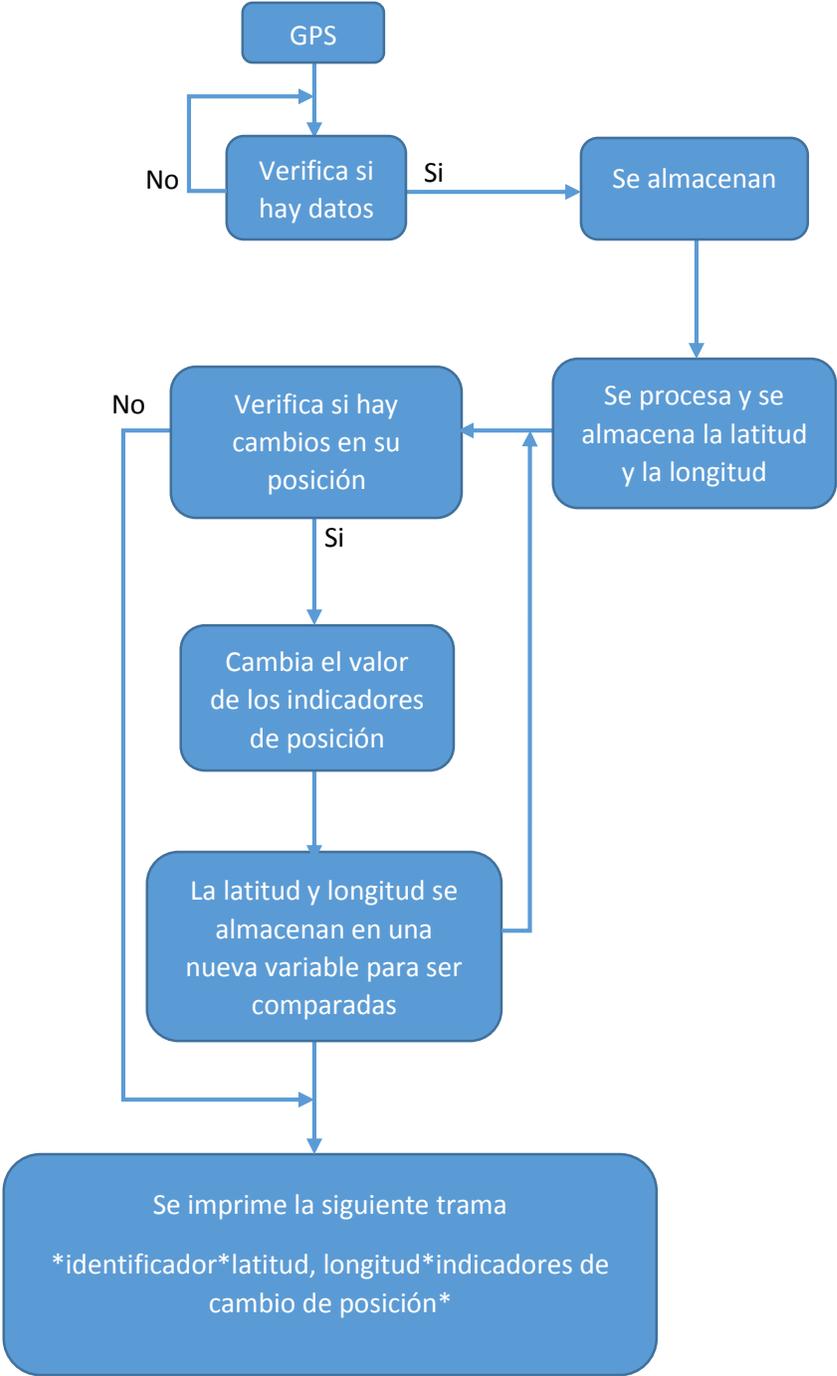
Serial.print("**GPS**"); //imprime el identificador del que GPS es
Serial.print( lati, 6); //imprime la latitud con 6 decimales
Serial.print(","); //imprime una coma
Serial.print( lon, 6); //imprime la longitud con 6 decimales
Serial.print("**"); //imprime un carácter
Serial.print(x); //imprime el indicador de latitud
Serial.print(y); //imprime el indicador de longitud
Serial.println("**"); //imprime otro carácter
delay(100); // espera 100 milisegundos
x = 0; // manda a cero los indicadores
y = 0;
}

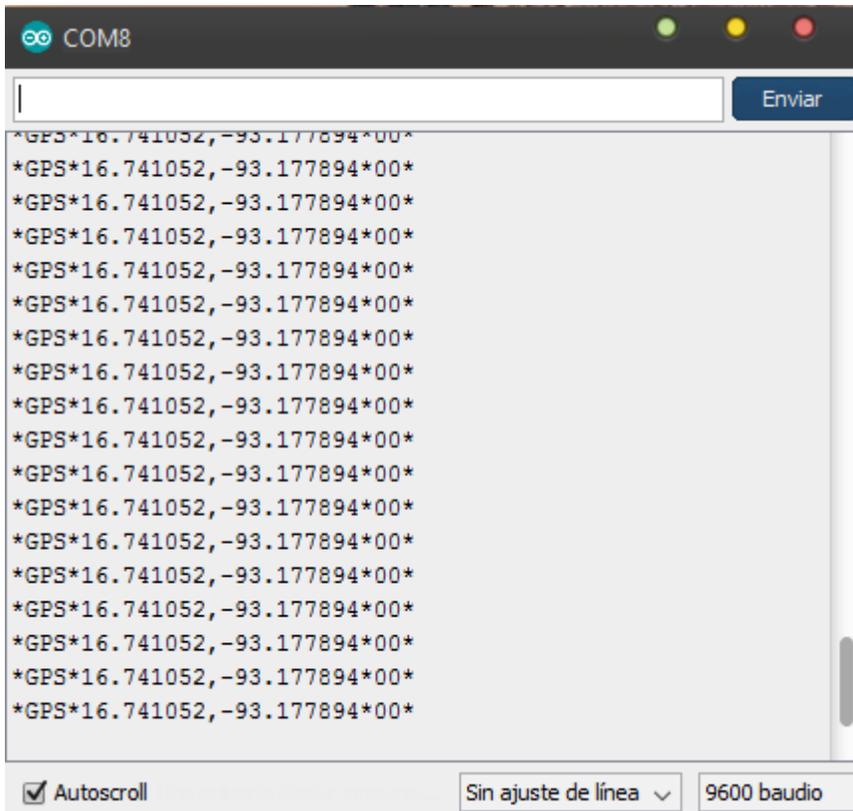
void readgps()
{
if (gps.location.isValid()) //pregunta si la localización es valida
{ lati = gps.location.lat(); //si es válida extrae la coordenada de la latitud
lon = gps.location.lng(); } //extrae la coordenada de la longitud
else //si la localización no es válida le coloca 0 en vez de la coordenadas
{ lati = 0;
lon = 0; }
}

```

Todo esto es nada más para el funcionamiento del dispositivo electrónico que se colocara a los animales para poder llevar acabo su vigilancia.

Quedando de la siguiente forma el funcionamiento del dispositivo:





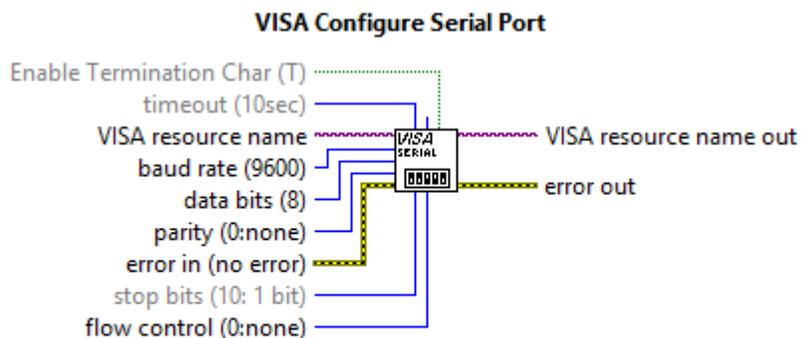
Código generado por el Arduino y enviado a través de radio frecuencia por medio del Xbee.

#### 4.1.2 Programación en LABVIEW

En esta parte se trabajó con el software LabVIEW para crear la interfaz gráfica de usuario y poder visualizar en una extensión de google maps.

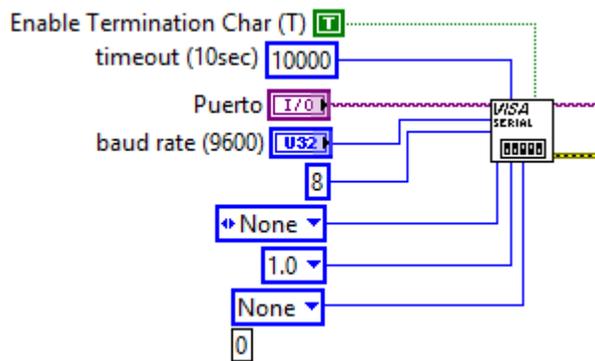
Herramientas utilizadas en labview para desarrollar el software

Bloque que permite configurar y utilizar el puerto serial.

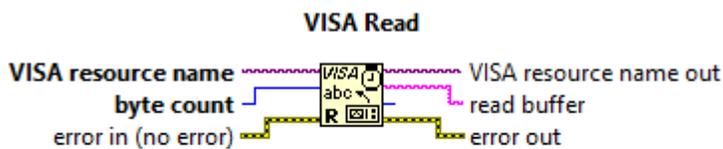


Initializes the serial port specified by **VISA resource name** to the specified settings. Wire data to the **VISA resource name** input to determine the polymorphic instance to use or manually select the instance.

Quedando de esta manera correctamente configurado para poder seleccionar el puerto serial y la velocidad del baud.

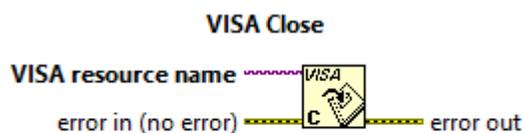


Bloque que permite leer los datos del puerto serial.



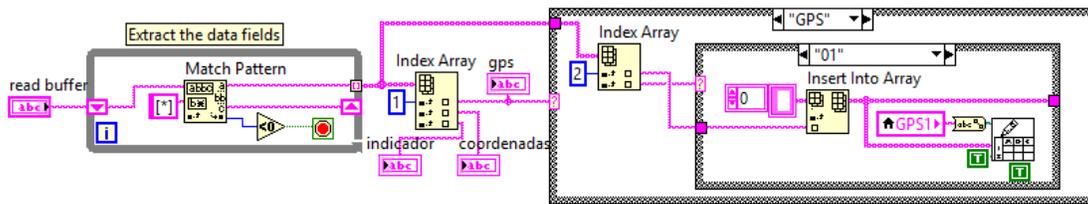
Reads the specified number of bytes from the device or interface specified by **VISA resource name** and returns the data in **read buffer**.

Bloque que finaliza el uso del puerto serial.

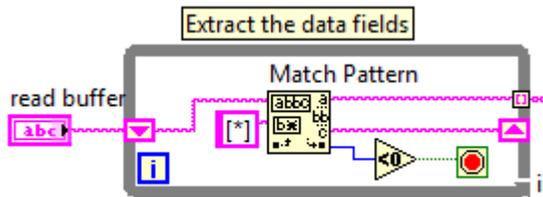


Closes a device session or event object specified by **VISA resource name**.

Primeramente se diseñó la parte que permite guardar los datos en una tabla de Excel en una carpeta de la computadora, nos genera 6 reporte por hora, y lo almacena con fecha.

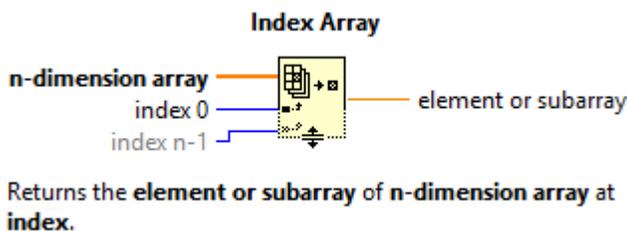


Los datos leídos en el buffer del puerto serial entran a un bucle infinito en el que el siguiente bloque se encarga de separar los datos.

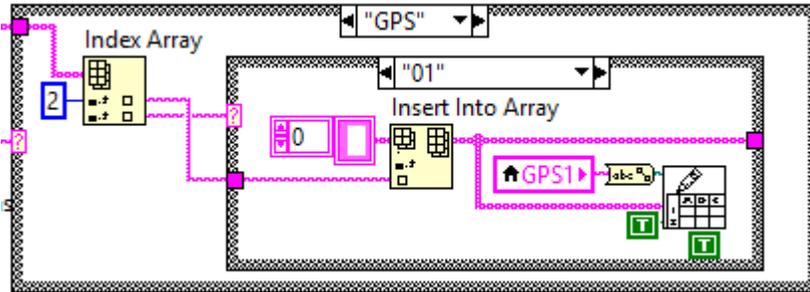


La trama de datos generada con Arduino lleva un distintivo que separa todos los datos, en este caso es un asterisco (\*), A si podemos utilizar los datos de una manera más fácil.

Una vez separados por medio del \* podemos hacer uso de ellos mediante el siguiente bloque que permite sacar los datos a la vez.



Una vez separados los datos son utilizados para poder identificar que dispositivo es para así poder ser almacenado en tiempo real y sin que se repita la misma coordenada para evitar datos innecesarios.



Quedando de esta forma los archivos generados y los datos almacenados.

Se genera una carpeta con la fecha de la siguiente forma:

Los primeros dos dígitos son para el día, los dos dígitos que le siguen son del mes y los últimos cuatro corresponden al año.

Nombre	Fecha de modifica...	Tipo	Tam
02052016	02/05/2016 12:50 ...	Carpeta de archivos	
24052016	24/05/2016 04:30 ...	Carpeta de archivos	
25052016	25/05/2016 01:30 ...	Carpeta de archivos	
28042016	28/04/2016 02:00 ...	Carpeta de archivos	
29042016	28/04/2016 11:27 a...	Carpeta de archivos	
30042016	28/04/2016 11:27 a...	Carpeta de archivos	

Dentro de cada carpeta se encuentran los reportes generados cada 10 minutos, identificados por hora.

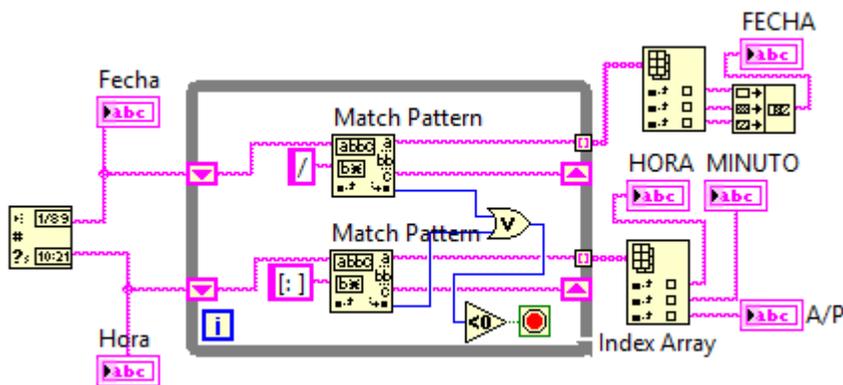
01 p.m. - 1	25/05/2016 01:09 ...	Hoja de cálculo d...	1 KB
01 p.m. - 2	25/05/2016 01:19 ...	Hoja de cálculo d...	1 KB
01 p.m. - 3	25/05/2016 01:29 ...	Hoja de cálculo d...	1 KB
01 p.m. - 4	25/05/2016 01:34 ...	Hoja de cálculo d...	1 KB
10 a.m. - 6	25/05/2016 10:59 a...	Hoja de cálculo d...	1 KB
11 a.m. - 1	25/05/2016 11:04 a...	Hoja de cálculo d...	1 KB
11 a.m. - 2	25/05/2016 11:19 a...	Hoja de cálculo d...	1 KB
11 a.m. - 3	25/05/2016 11:27 a...	Hoja de cálculo d...	1 KB
11 a.m. - 5	25/05/2016 11:41 a...	Hoja de cálculo d...	1 KB
12 p.m. - 1	25/05/2016 12:09 ...	Hoja de cálculo d...	1 KB
12 p.m. - 2	25/05/2016 12:19 ...	Hoja de cálculo d...	1 KB
12 p.m. - 3	25/05/2016 12:29 ...	Hoja de cálculo d...	1 KB
12 p.m. - 4	25/05/2016 12:39 ...	Hoja de cálculo d...	1 KB
12 p.m. - 5	25/05/2016 12:48 ...	Hoja de cálculo d...	1 KB
12 p.m. - 6	25/05/2016 12:59 ...	Hoja de cálculo d...	2 KB

Y estos son los datos almacenados en cada tabla de Excel para poder ser usados y manipulados de una forma más fácil y sencilla.

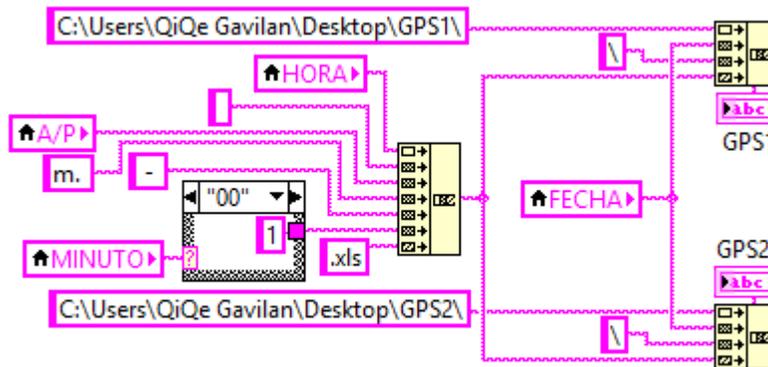
	A	B	C	D
1	16.757308,-93.172203			
2	16.757310,-93.172218			
3	16.757312,-93.172218			
4	16.757141,-93.172256			
5	16.757112,-93.172256			
6	16.757061,-93.172241			
7	16.757066,-93.172248			
8	16.757070,-93.172248			
9	16.757139,-93.172317			
10	16.757053,-93.172286			

Para poder generar todos esos archivos y poder ser almacenados se creó un generador de nombre y de dirección del archivo de donde queremos que sea guardado por medio de un reloj y un calendario de la siguiente forma:

En la siguiente configuración se utilizó el reloj y el calendario para poder ser separados y solo entregara la fecha y hora sin los separadores que lo distinguen.

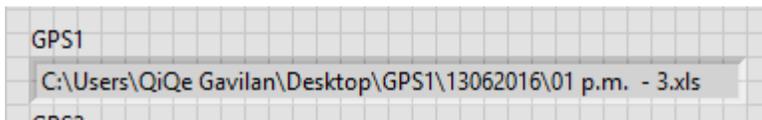


Una vez obtenido la fecha y hora se agrega al siguiente código para generar una dirección y nombre de archivo:

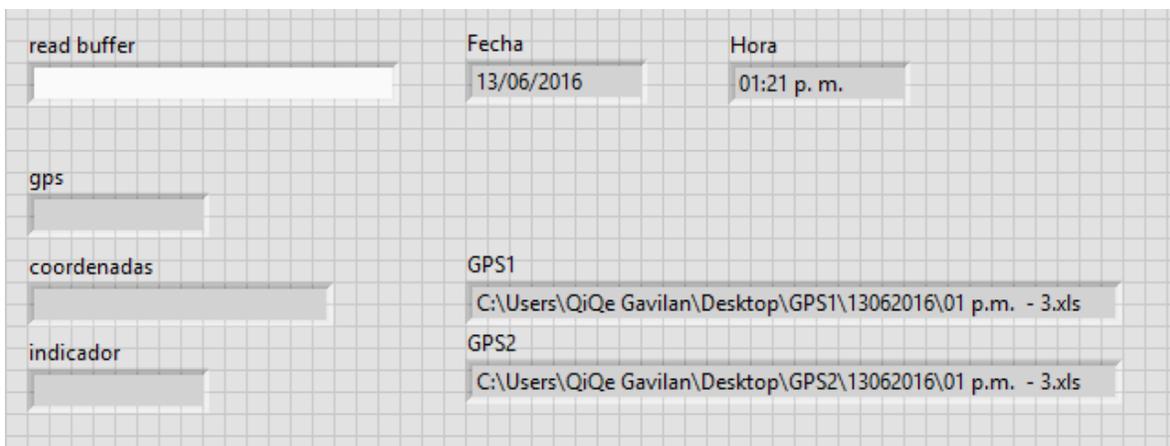


En la primer parte se le da la dirección de la carpeta de donde queremos que quede guardado, en seguida se le agrega la fecha que creara una nueva carpeta, seguido de eso se le agrega la hora agregando si es pm o am y se divide en 6 los 60 segundos para saber en qué reporte de la hora se encuentra cuando recibió el dato para poder ser almacenado en tiempo real, por último se le agrega el tipo de archivo que va generar en este caso es un archivo de Excel.xls

Quedando de la siguiente forma:



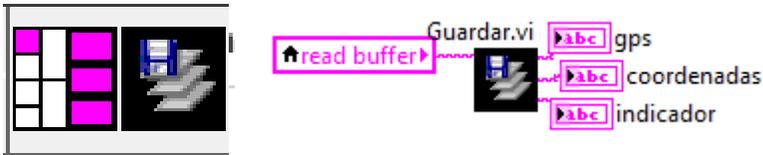
Una vez creado todo el sistema de almacenamiento en tiempo real, en la interfaz de usuario nos queda la siguiente forma:



El read buffer que es donde entran los datos a procesar, fecha y hora nos indica la fecha y hora en la que entraron los datos, GPS1 y GPS2 son las direcciones con los nombres de los archivos que serán almacenados en tiempo real, en GPS nos regresa el identificador de GPS para saber de qué GPS estamos analizando los datos, coordenadas son las coordenadas del GPS que se está recibiendo datos,

indicador es un número que nos ayuda a poder almacenar los datos de una forma más sencilla la cual nos permite saber si el gps sigue en la misma posición o si hubo un cambio de posición, así para almacenar solo las coordenadas necesarias y evitar un uso excesivo de memoria en los archivos generados.

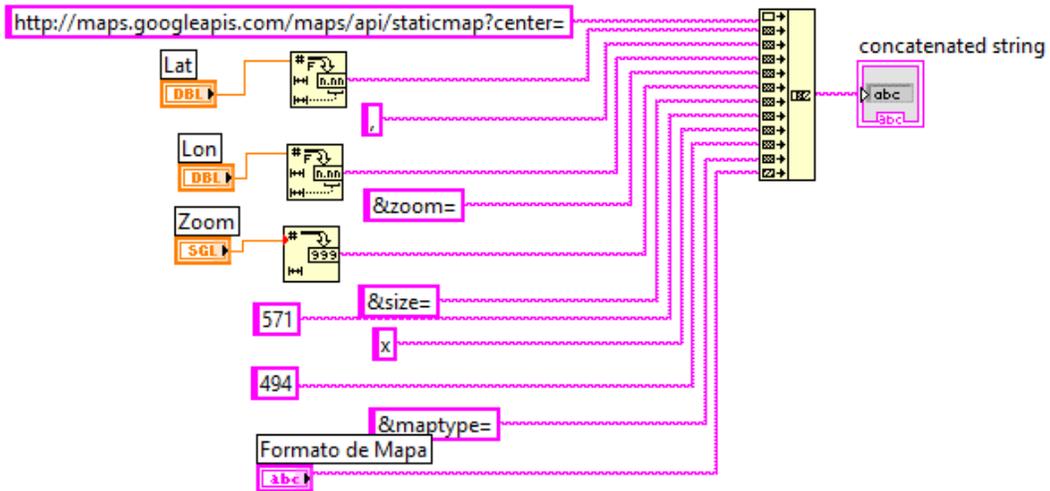
Una vez tenido esto se guardó el archivo .vi con la posibilidad de ser incluido en un nuevo proyecto de Labview dándole la posibilidad de ingresar datos y sacar datos de esta forma poder disminuir la cantidad de bloques usados en el programa principal.



Read buffer recibe todos los datos leídos en el puerto serial y son procesados y almacenados en tiempo real y nos devuelve el identificador, las coordenadas de ese identificador y los indicadores de cambio de posición.

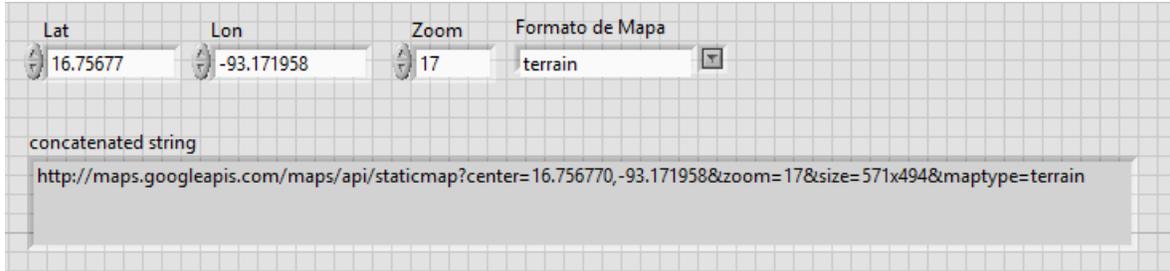
Después de eso se creó el apartado donde se indica la ubicación del lugar y ciertas características del mapa a mostrar. Para eso se usó los datos de la página de google “Guía del desarrollador de Google Static Maps” donde indica la forma de generar tu propio mapa estático.

Quedando de la siguiente forma el arreglo para generar el mapa del lugar por medio de un concatenado que agrupa todos los datos proporcionado para así generar la url.

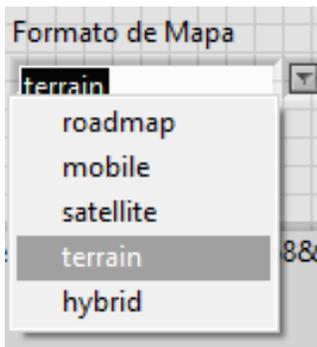


Este es el diagrama a bloques donde se especifica la parte principal que debe tener la url para crear el mapa, seguido de la latitud y longitud del lugar, de ahí se le indica el zoom el tamaño del mapa y tipo de mapa.

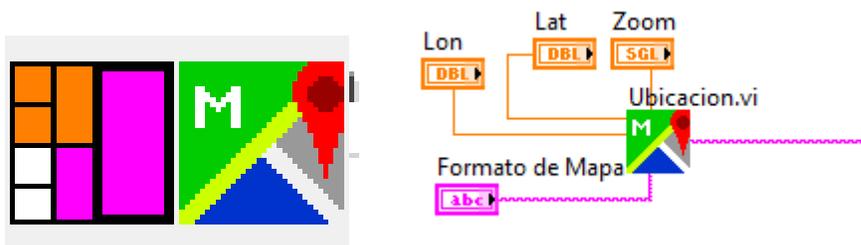
Quedando a si la interfaz de usuario para poder agregar los datos a su preferencia



En el formato de mapa le agregamos los diferentes tipos de mapas posibles para que el usuario elija los cuales son los siguientes:



Una vez teniendo todo esto se guardó con la posibilidad de ser integrado en otro programa de labview y así poder reducir el tamaño del programa principal



Quedando de esa manera dentro del programa principal, se le meten los datos y genera una dirección url con las características que le indiquemos, esto para luego ser adjuntada con otros datos más que se especificaran adelante.

Después de eso se creó la parte para agregar la posición del GPS actual utilizando concatenación para agregarle los datos necesarios para identificar cada GPS.

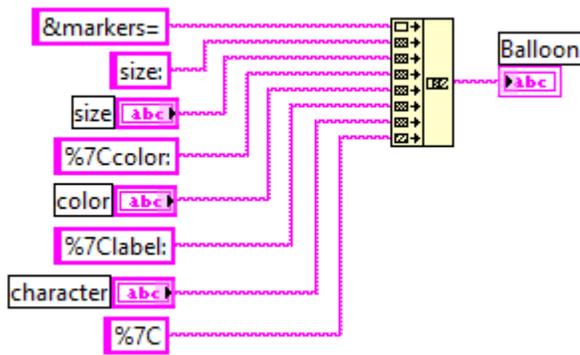
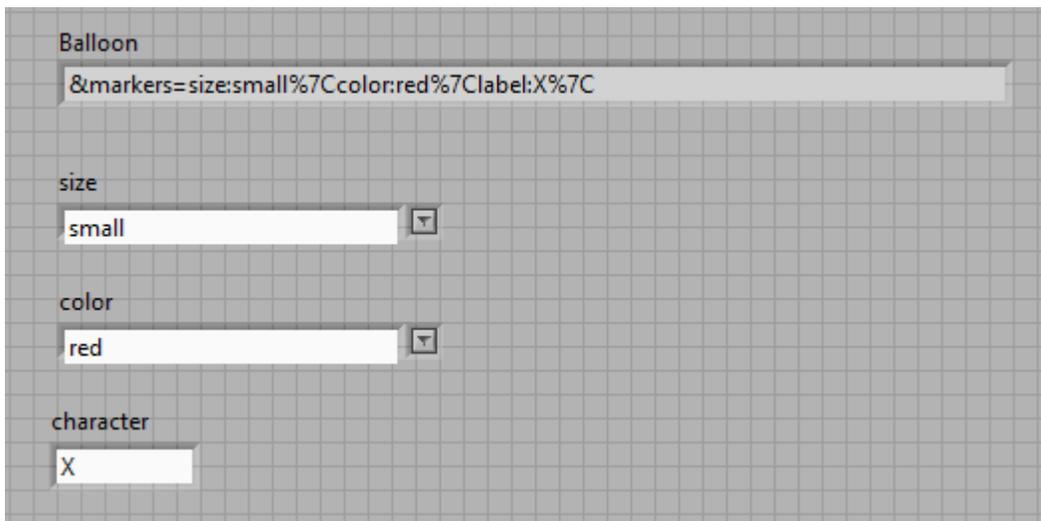
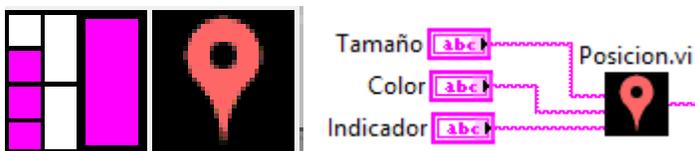


Diagrama a bloques que contiene las características de los indicadores, las cuales se inicializa con un markers= anteponiéndole una & para identificarlo, seguido se le agrega el tamaño, color, indicador, todo esto guardándolo en un balloon para después ser procesado.

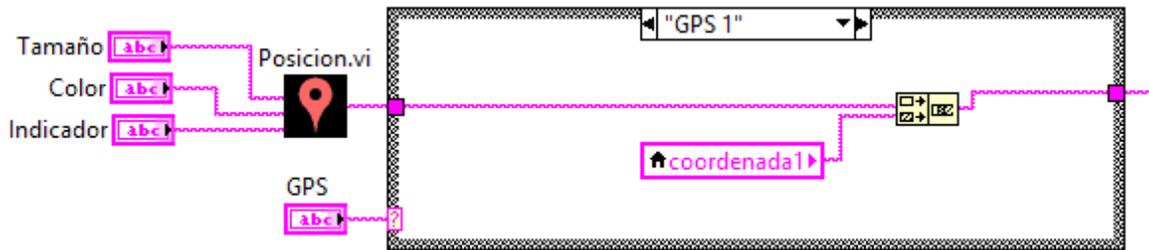


Quedando así la parte de la interfaz de usuario, en size es el tamaño las cuales contiene small, mid y tiny. En color contiene los colores básicos, rojo, verde, azul, blanco, negro, y en character va una letra del alfabeto en mayúsculas para identificarlo.



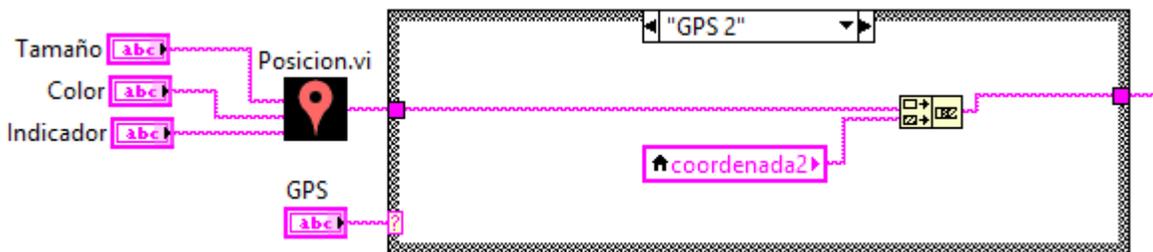
Quedando de esa manera dentro del programa principal, se le meten los datos y genera una parte de la url que contiene las características del indicador del GPS.

A continuación se le agrega a un case para poder identificar que GPS se ha seleccionado y así poder mostrar solo el indicado, quedando de la siguiente forma para el GPS1:

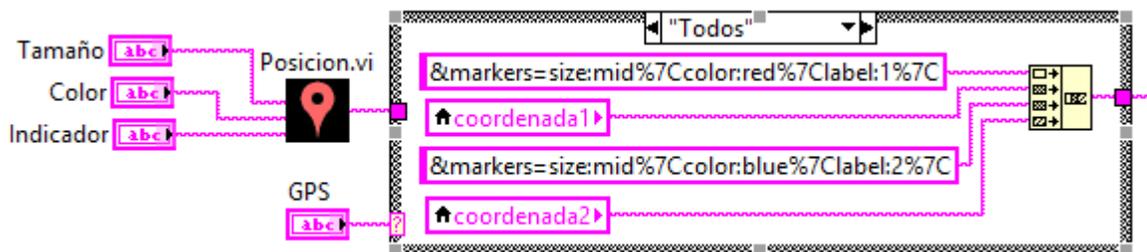


Quedando así la parte para seleccionar solo el GPS1 la cual a la parte que genera las características del indicador de GPS se le agrega la coordenada correspondiente.

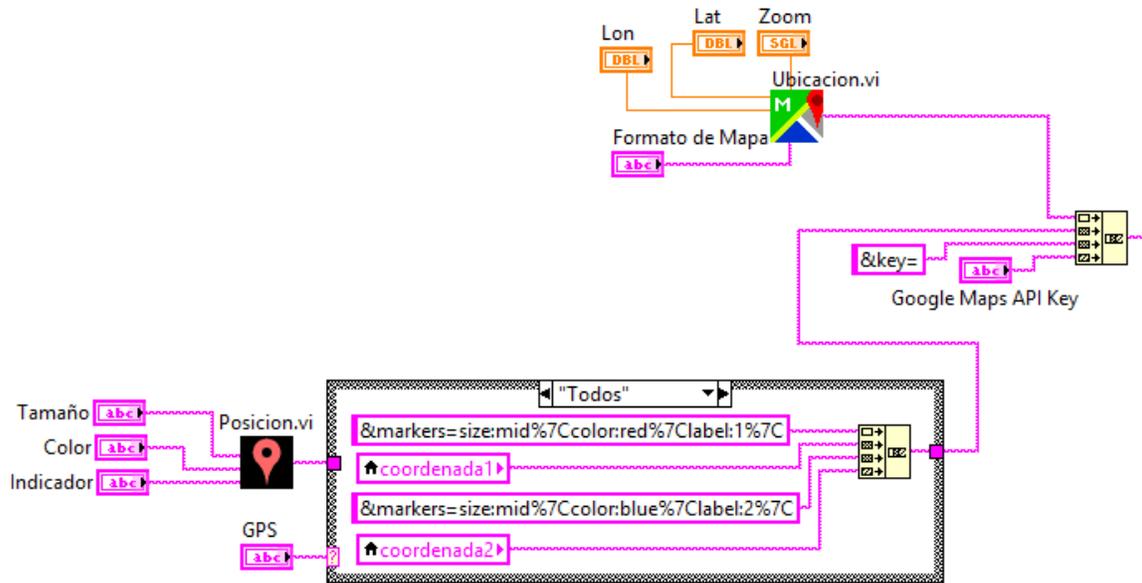
El siguiente es para el GPS2 que sería casi lo mismo:



Para visualizar los 2 GPS a la vez se modificó para que ignorara la parte que contiene el identificador de posición para que solo mostrara el básico y así poder modificar el identificador solo si esta seleccionado un GPS.



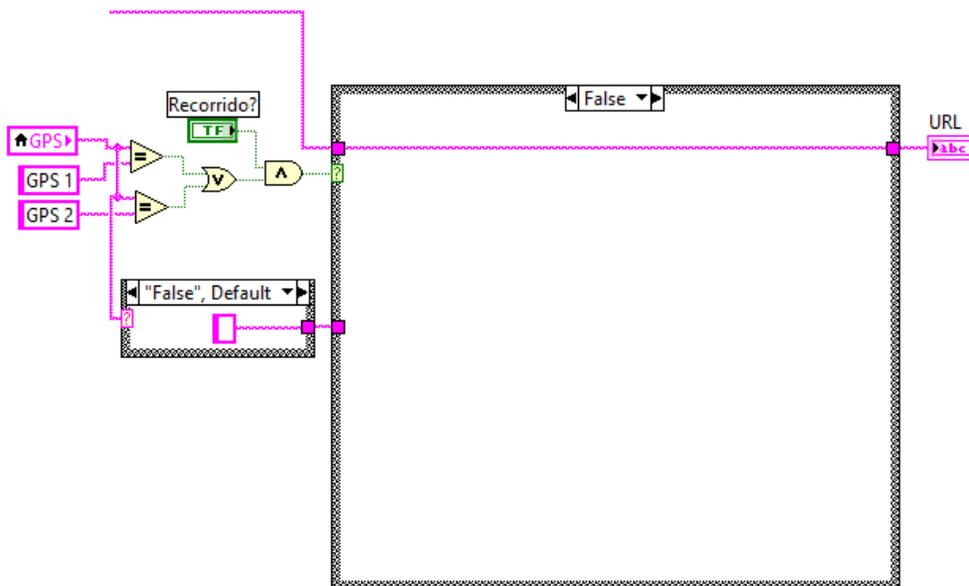
Una vez teniendo esa parte se le agrega a la parte principal donde contiene la ubicación del mapa y tipo de mapa que es la parte principal de la url, así quedando la url con mayor características y así poder visualizar la posición de los GPS's.



Se unen por medio de un concatenador para poder tener una url aceptable para la imagen del mapa, la cual también se le agrega un key y un cuadro para meter el key que es una clave para poder usar los mapas de google por mayor tiempo.

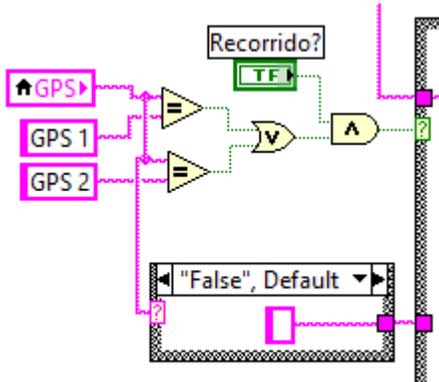
Una vez generada la url donde contiene las características del mapa y los indicadores del GPS, esta pasa por otro case donde se verifica si el pulsador de recorrido está activado o no.

Si no está activado el pulsador queda de la siguiente manera:



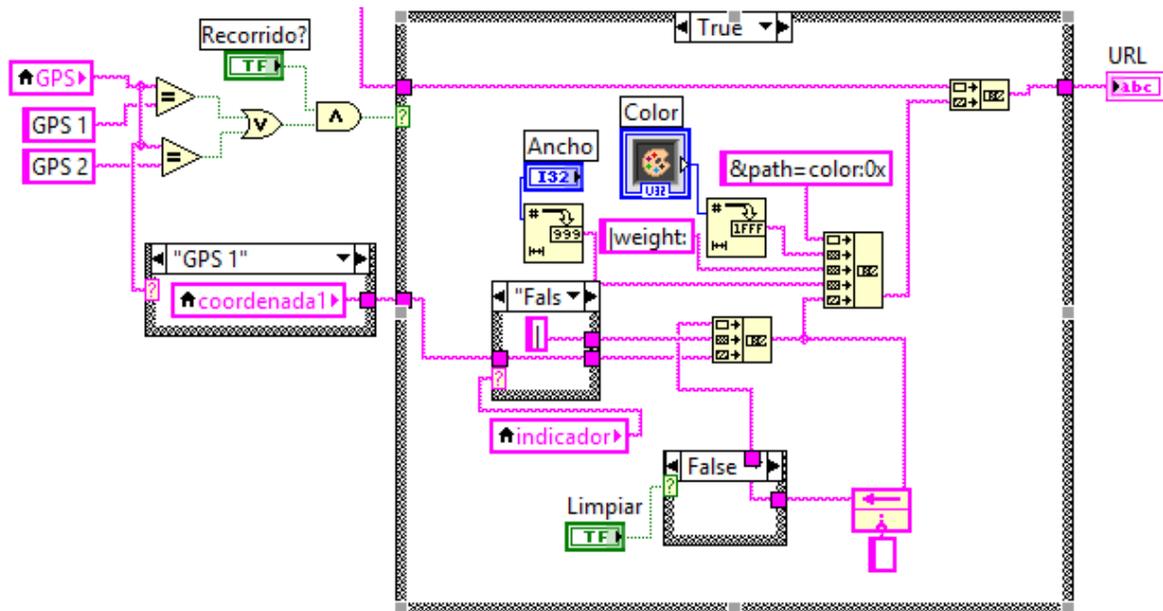
Si no está activado pasa directo a un indicador donde guarda la url final.

Si el pulsador de recorrido está activado se deben cumplir unas ciertas reglas para que el recorrido se muestre.

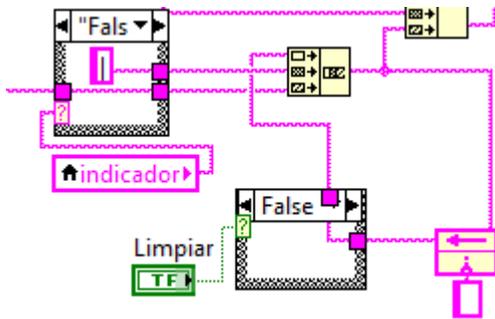


Si el GPS1 o el GPS2 está seleccionado se activa el case.

Quedando de esta manera:

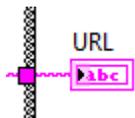


Al activar el recorrido se puede configurar la opción de la línea que muestra el recorrido, pudiendo agregarle el color, ancho de la línea y a continuación se le agrega un bucle el cual permite acumular las coordenadas que generara el recorrido.

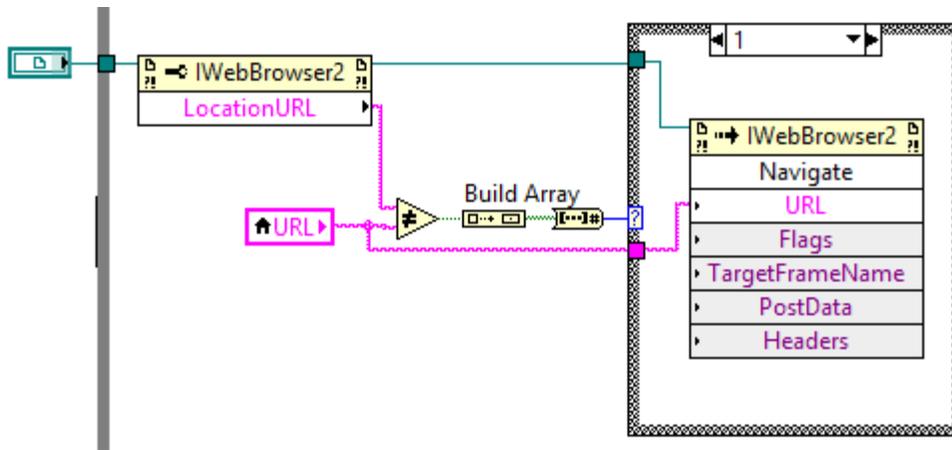


Este pequeño bucle permite agrupar todas las coordenadas agregándole un separador para identificar que hay varias coordenadas juntas, la cual el bucle está en un case que se activa con un pulsador lo cual permite limpiar el bucle para iniciar un nuevo recorrido.

Una vez generada la url completa se guarda en un indicador

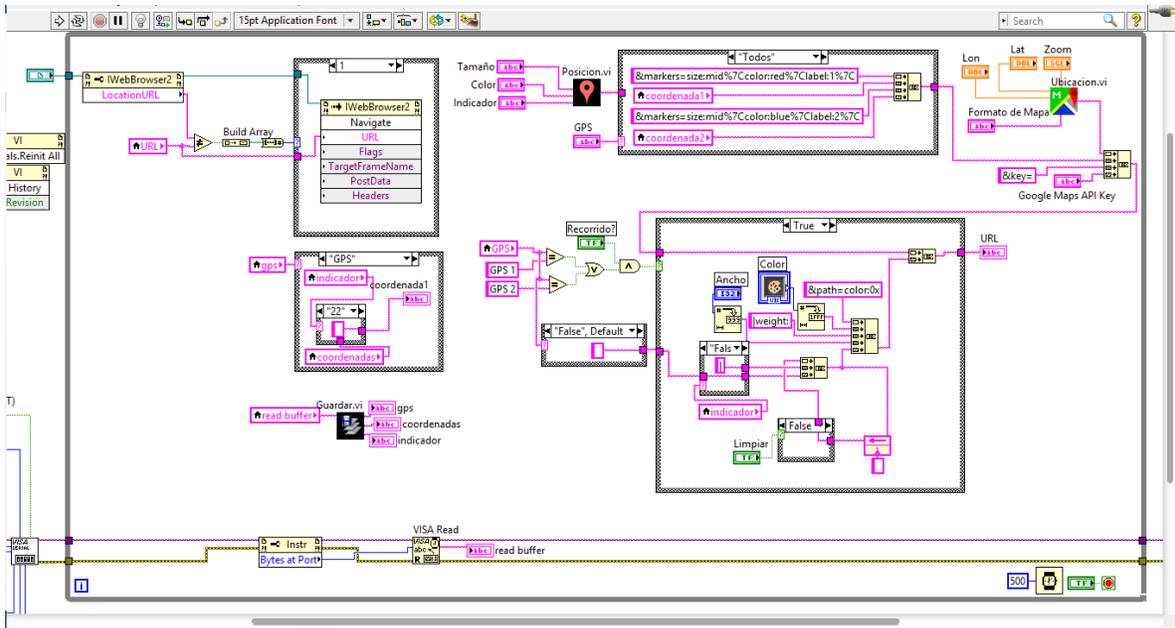


La url es comparado con un visor de página web el cual compara la página que tiene actual con la que se generó, si la página o el url de la página no es igual se activa un case el cual almacena la nueva url y la coloca en un visor de pagina web.



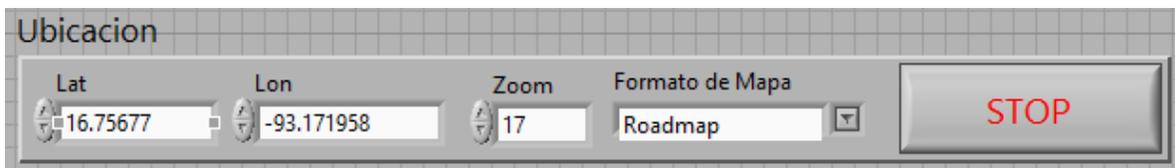
Esto permite que cada que hay un cambio de posición en los GPS la url cambia lo cual la url ya no es la misma permitiendo almacenar la nueva y ser mostrada.

Quedando así toda la parte de diagrama a bloques donde se configuro lo necesario para funcionar:



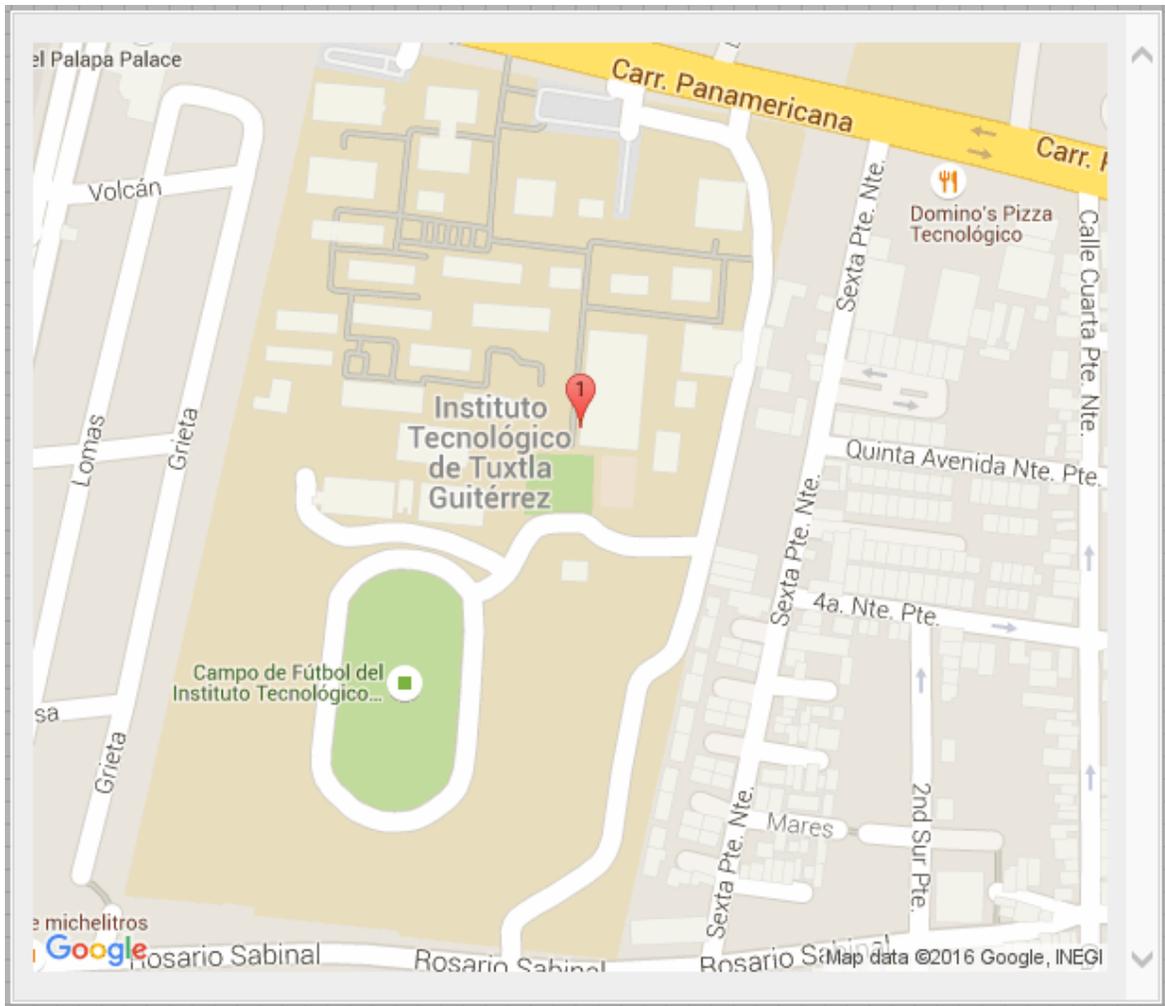
La interfaz de usuario queda de la siguiente manera:

En la parte superior se encuentran las configuraciones del lugar junto con un botón para finalizar el programa.



Permite agregarle las coordenadas del lugar, el zoom y el formato de mapa.

En la parte de en medio queda el visor de página web, donde básicamente se muestra la url generada en todo el proceso.



En la parte derecho se encuentra las configuraciones para seleccionar el puerto de comunicación y la velocidad de trasmisión y recepción donde se encuentra el dispositivo receptor de radio frecuencia.

Seguido se encuentra el selector de GPS el cual permite visualizar todos los GPS a la vez o solo uno. Al seleccionar todos los GPS no se puede utilizar las opciones que continúan, al seleccionar solo un GPS se puede acceder a configurar el tamaño, color y un carácter para visualizar el indicador de posición.

**SELEC GPS**

**Puerto**  
COM8

baud rate (9600)  
9600

GPS  
Todos

Tamaño  
Normal

Color  
Rojo

Indicador  
X

Seguido de ese se encuentra la parte para activar el recorrido del GPS la cual permite colocarle el color y el ancho del recorrido siempre y cuando este seleccionado un solo GPS, también cuenta con un botón para limpiar el recorrido.

Recorrido? Limpiar

NO OK

Color Ancho

5

Por ultimo en la parte de abajo tenemos la parte para agregar una API Key de google que permite usar los mapas estáticos por mayor tiempo al día, a la derecha se encuentra un indicador de lo que se está leyendo en el puerto serial, esto nos ayuda a identificar que si está recibiendo datos el receptor, por último en la parte de abajo tenemos la URL que se muestra en el visor de página web, si esta dirección la colocamos en un navegador de internet normal también nos muestra el mapa con los GPS que tengamos configurados para ver en ese instante.

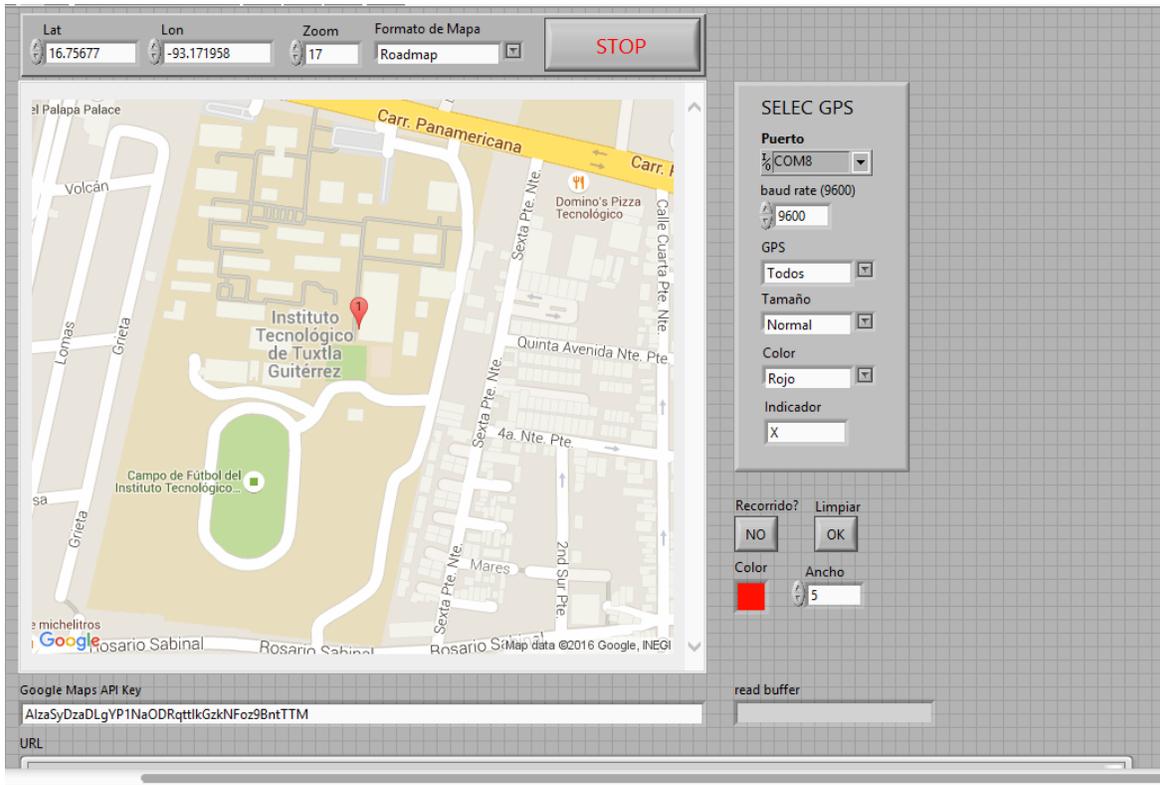
Google Maps API Key read buffer

AlzaSyDzaDLgYP1NaODRqtllkGzkNFoz9BntTTM

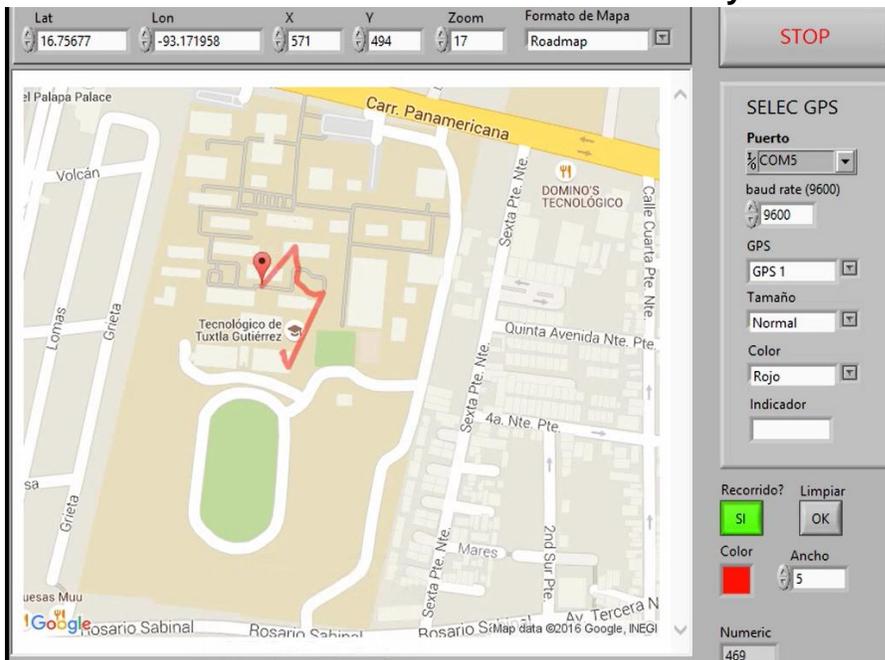
URL

<http://maps.googleapis.com/maps/api/staticmap?center=16.756770,-93.171958&zoom=17&size=571x494&maptype=roadmap&markers=size:mid%7Ccolor:red%7Clabel:1%7C&markers=size:mid%7Ccolor:blue%7Clabel:2%7C&key=AlzaSyDzaDLgYP1NaODRqtllkGzkNFoz9BntTTM>

Quedando todo el programa de la siguiente manera:

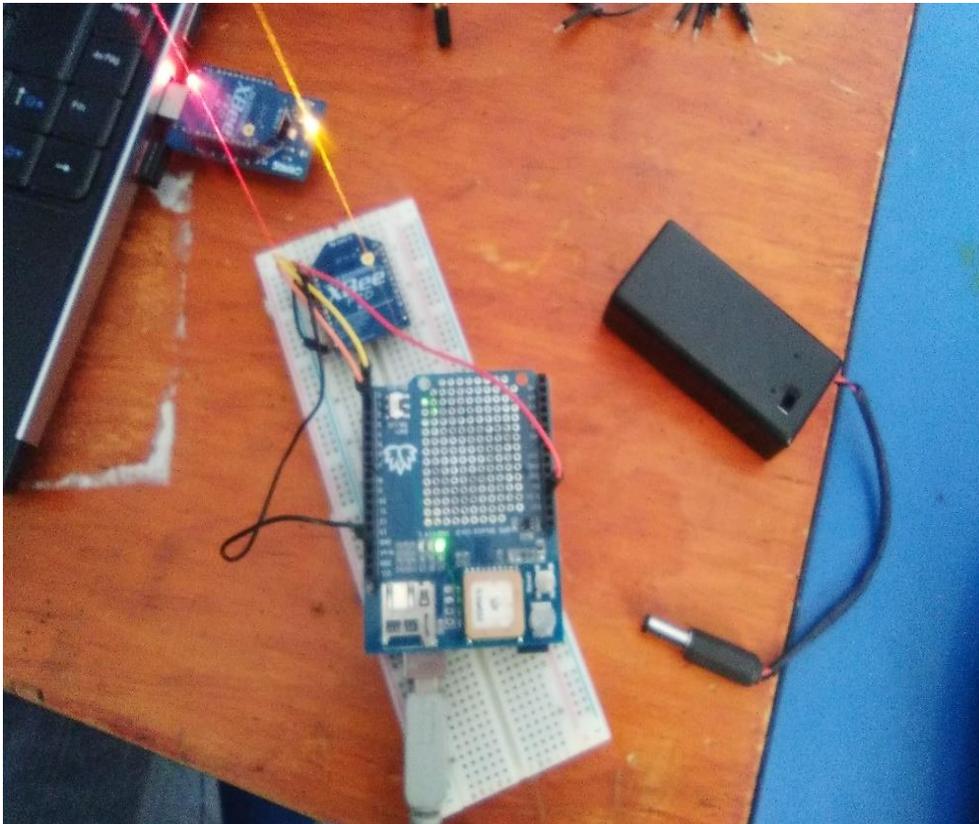


#### 4.1.3- Pruebas del funcionamiento del software y hardware en conjunto:



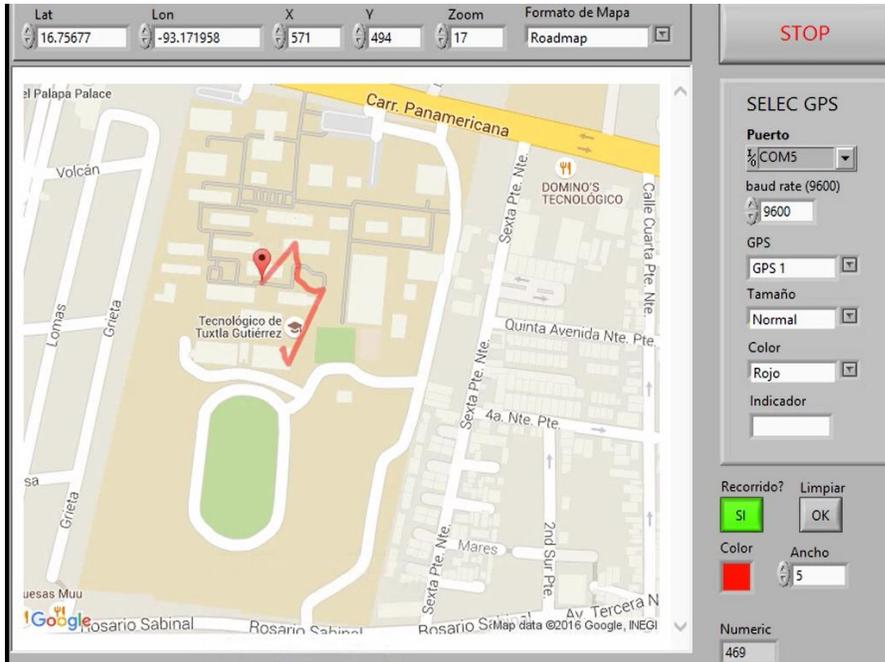
En la prueba de arriba podemos ver como el sistema funciona correctamente mostrándonos un indicador de donde se encuentra el GPS y a su vez el recorrido concluyendo que ha sido un éxito el trabajo realizado en la residencia y que se cumple con los objetivos específicos.

Estas son algunas imágenes de las pruebas en protoboard que se hicieron:



## 4.2 Resultados

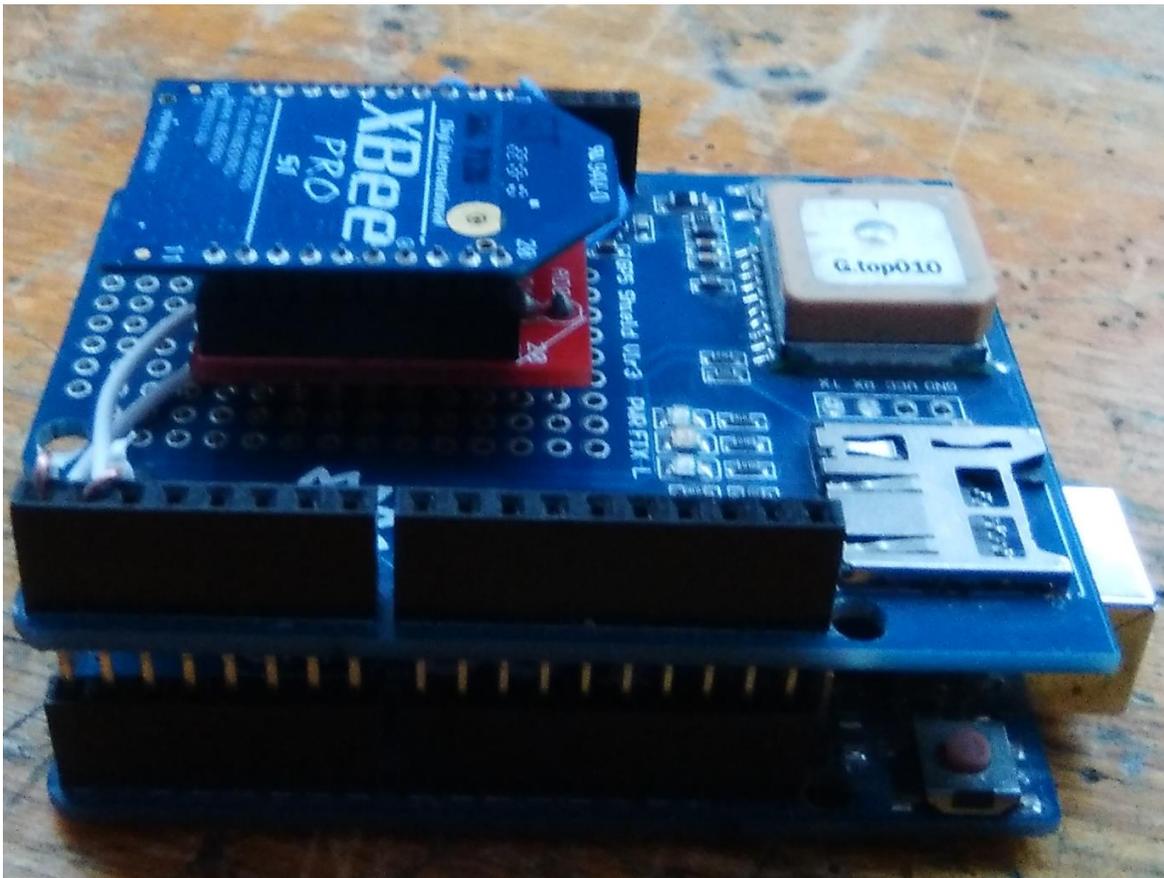
Se logró obtener la interfaz gráfica para el usuario para poder visualizar un recorrido de un portador.



La distancia máxima alcanzada con línea de vista fue 1.4 km.

La distancia máxima alcanzada con varias estructuras, edificios arboles etc. apenas llego a 350metros.

Aquí se muestra el prototipo final



# Capítulo 5

## 5.1 Conclusiones y recomendaciones

En este proyecto se buscó localizar animales para su seguimiento y monitorización en tiempo real, en este proyecto se tuvo un poco de complicaciones al escoger materiales y encontrarlos disponibles en las tiendas por lo cual se optó por implementar algunos materiales que se tenía a la mano, por lo cual nuestro prototipo quedo un poco más grande lo que se quería, pero su funcionamiento es el indicado para poder completar nuestros objetivos. También tuvimos un poco de dificultades para encontrar algunos antecedentes de proyectos, por lo cual se tuvo que iniciar casi de 0, con la ayuda de algunos tutoriales de otros proyectos que podían vincularnos a cada parte de lo que se hizo. Se inicia programación del GPS con Arduino. No se obtuvo ningún dato no sabíamos si era el Arduino, el cable datos o el shield GPS que estaba fallando. En la programación se utilizó las librerías tinygps y adafruit. Se continúa la programación y chequeo de los dispositivos, se encuentra el error en la programación, la frecuencia del GPS era 8400 y da error, se cambió 9600, así como la velocidad del puerto serial 11520 y funciona así como los pines del 0,1 se cambió al 3,2 el problema de estar en los pines (0,1) eran que tenían conflicto a la hora de comunicarse y estas transmitiendo y recibiendo.

Se empezó a crear el diseño de la interfaz en el programa LabVIEW el cual se llevó bastante tiempo, después tuvimos detalles que los muestreos de la extensión que se había puesto era limitados por lo cual se procedió a obtener una API de google maps la cual nos permitió: Gratis hasta exceder 25 000 cargas de mapas por día durante 90 días consecutivos, Resolución máxima de la imagen: 640 x 640.

Algunas recomendaciones que se dan para el uso de este dispositivo:

- Que el portador del dispositivo no esté dentro de estructuras como edificios.
- Suele tener un error de 5 mts así que no es totalmente preciso

- El dispositivo trabaja en condiciones de cielo despejado y suele no recibir datos cuando esta nublado totalmente.
- Actualmente el dispositivo no es impermeable por lo cual no se puede mojar.
- Su tiempo de recarga de la batería será de 2, 3 veces al día.

## **5.2 Competencias desarrolladas y/o aplicadas**

- Trabajo en equipo
- Reutilización de materiales en desuso.
- Análisis de problemas
- Creatividad
- Iniciativa
- Tenacidad
- Compromiso
- Habilidad para trabajar de manera autónoma
- Capacidad de generar nuevas ideas
- Destrezas lingüísticas principalmente escrita
- Capacidad crítica y autocrítica
- Capacidad para comunicarse con profesores de otras áreas
- Programación en diversas plataformas
- Creación de funciones
- Aplicación de formulas
- Diseño de circuitos
- Adaptabilidad
- Automotivación

### 5.3 Referencias bibliográficas

[http://www.otca.info/biodiversidade/2009/publico/ archivos/File/Informe%20de%20Consulta%20MT%20\(final\).pdf](http://www.otca.info/biodiversidade/2009/publico/ archivos/File/Informe%20de%20Consulta%20MT%20(final).pdf)

<http://www.inegi.org.mx/geo/contenidos/geodesia/gps.aspx?dv=c1>

<http://www.elcomercio.com/actualidad/sistemaelectronico-vigilancia-procesados-justicia-ecuador.html>

<http://www.pgje.chiapas.gob.mx/informacion/marcojuridico/Leyes/Estatales/Update/LEY%20DE%20PROTECCI%C3%93N%20PARA%20LA%20FAUNA%20EN%20EL%20ESTADO%20DE%20CHIAPAS.pdf>

<http://www.fractaliasystems.com/idi-dispositivo-electronico-para-la-proteccion-de-la-fauna-salvaje/>

<http://martinml.com/antiguo/honda-electro-gyrocatator-el-precursor-del-navegador-gps/>

<http://www.alsitel.com/tecnico/gps/historia.htm>

<http://www.revistadeinnovacion.com/es/productos.php?var1=OntheBus,%20primer%20GPS%20para%20invidentes&nar1=NDE=>

<https://es.wikipedia.org/wiki/LabVIEW>

<http://www.ni.com/labview/applications/esa/>

<http://empiricasoft.blogspot.mx/2013/09/arducontrol-mk-i-parte-i.html>

<https://developers.google.com/maps/documentation/static-maps/intro>

## 5.4 Anexo

### UART

Transmisor-Receptor Asíncrono Universal, es el dispositivo que controla los puertos y dispositivos serie. Se encuentra integrado en la placa base o en la tarjeta adaptadora del dispositivo.

Un UART dual, o DUART, combina dos UART en un solo chip. Existe un dispositivo electrónico encargado de generar la UART en cada puerto serie. La mayoría de las computadoras modernas utilizan el chip UART 16550, que soporta velocidades de transmisión de hasta 921,6 Kbps (Kilobits por segundo). Las funciones principales de chip UART son: manejar las interrupciones de los dispositivos conectados al puerto serie y convertir los datos en formato paralelo, transmitidos al bus de sistema, a datos en formato serie, para que puedan ser transmitidos a través de los puertos y viceversa.

### Pseudodistancia

Una pseudodistancia o, más generalmente, una familia de pseudodistancias determina en un conjunto una estructura uniforme. El espacio topológico resultante se denomina espacio de calibración o espacio gauge.

Recíprocamente, toda estructura uniforme puede ser inducida por una familia de pseudodistancias.

Google Developers