



UNIVERSIDAD TÉCNICA DE AMBATO

**FACULTAD DE INGENIERÍA EN SISTEMAS ELECTRÓNICA E
INDUSTRIAL**

CARRERA DE INGENIERÍA EN ELECTRÓNICA Y COMUNICACIONES

Tema:

**“SISTEMA DE MONITOREO WEB PARA VEHICULOS MEDIANTE
HARDWARE LIBRE”**

Trabajo de Graduación. Modalidad: Proyecto de Investigación, presentado previo la obtención del título de Ingeniero en Electrónica y Comunicaciones.

SUBLÍNEA DE INVESTIGACIÓN:

Sistemas Embebidos

AUTOR:

Chulde Bonilla Luis Iván

TUTOR:

Ing. Mg. Geovanni Brito

Ambato – Ecuador

Mayo 2017

APROBACIÓN DEL TUTOR

En mi calidad de tutor del Trabajo de Investigación sobre el tema: “SISTEMA DE MONITOREO WEB PARA VEHICULOS MEDIANTE HARDWARE LIBRE”, del señor LUIS IVÁN CHULDE BONILLA, estudiante de la Carrera de Ingeniería en Electrónica y Comunicaciones , de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el numeral 7.2 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.

Ambato, mayo del 2017

EL TUTOR



Ing. Geovanni Brito, Mg

AUTORÍA

El presente Proyecto de Investigación titulado: “SISTEMA DE MONITOREO WEB PARA VEHICULOS MEDIANTE HARDWARE LIBRE”, es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, mayo del 2017



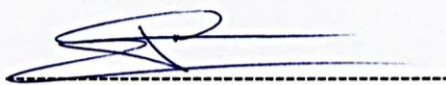
Luis Iván Chulde Bonilla
CC: 1600454878

DERECHOS DE AUTOR

Autorizo a la Universidad Técnica de Ambato, para que haga uso de este Trabajo de Titulación como un documento disponible para la lectura, consulta y procesos de investigación.

Cedo los derechos de mi Trabajo de Titulación, con fines de difusión pública, además autorizo su reproducción dentro de las regulaciones de la Universidad.

Ambato, mayo del 2017


A handwritten signature in blue ink, consisting of a stylized 'L' and 'C' followed by a horizontal line.

Chulde Bonilla Luis Ivan

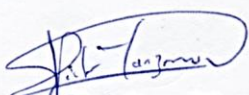
CC: 1600454878

APROBACIÓN DE LA COMISIÓN CALIFICADORA

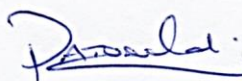
La Comisión Calificadora del presente trabajo conformada por los señores docentes Ing. Patricio Encalada e Ing. Santiago Manzano, revisó y aprobó el Informe Final del Proyecto de Investigación titulado “SISTEMA DE MONITOREO WEB PARA VEHICULOS MEDIANTE HARDWARE LIBRE”, presentado por el señor Chulde Bonilla Luis Ivan de acuerdo al numeral 9.1 de los Lineamientos Generales para la aplicación de Instructivos de las Modalidades de Titulación de las Facultades de la Universidad Técnica de Ambato.



Ing. Mg. Elsa Pilar Urrutia Urrutia
PRESIDENTA DEL TRIBUNAL



Ing. Mg. Santiago Manzano
DOCENTE CALIFICADOR



Ing. Patricio Encalada, MSc.
DOCENTE CALIFICADOR

DEDICATORIA

Este proyecto va dedicado a Dios quien me dio una nueva oportunidad y siempre me estuvo cuidando en todo el transcurso de mi vida.

A mi Madre Silvia Cecilia Bonilla Chonata, que gracias a sus muchos esfuerzos que hizo en el extranjero y gracias a todo apoyo económico y emocional hoy pude cumplir con este gran paso en mi vida.

A mi amada novia Ana Cristina Borja Calderón por ser mi apoyo y ayudarme siempre en todo lo que he necesitado, por amarme y protegerme de todo mal.

A mi familia por apoyarme y cuidarme inculcándome buenas enseñanzas desde mi infancia

Luis Iván Chulde Bonilla

AGRADECIMIENTO

Quiero expresar mis más sinceros agradecimientos a mi Madre Silvia Bonilla y mi Familia por todo el apoyo que me han brindado, por sus enseñanzas y por cuidarme.

Agradezco a la Familia Borja-Calderón por abrirme las puertas de su hogar y ayudarme a progresar espiritualmente cada día.

Agradezco a Ana Cristina Borja Calderón, mi compañera, amiga y novia eterna por brindarme el regalo más preciado de mi vida, el nacer de nuevo y por haber dedicado cada minuto de su tiempo en ayudarme para salir hacia adelante, y el traer luz a mi vida.

Agradezco a mi Tutor Ing. Giovanni Brito por brindarme todo el conocimiento y su tiempo en la realización de este proyecto.

Luis Iván Chulde Bonilla

ÍNDICE

APROBACIÓN DEL TUTOR	i
AUTORÍA	ii
DERECHOS DE AUTOR	iii
APROBACIÓN DE LA COMISIÓN CALIFICADORA	iv
DEDICATORIA	v
AGRADECIMIENTO	vi
ÍNDICE	vii
ÍNDICE DE FIGURAS	ix
ÍNDICE DE TABLAS	xii
RESUMEN	xiii
ABSTRACT	xiv
GLOSARIO DE TÉRMINOS	xv
INTRODUCCIÓN	xviii
CAPÍTULO I	1
1.1. Tema	1
1.2. Planteamiento del problema	1
1.3. Delimitación del problema	3
1.4. Justificación	3
1.5. Objetivos	4
1.5.1. Objetivo General	4
1.5.2. Objetivos Específicos	4
CAPÍTULO II	5
2.1. Antecedentes Investigativos	5
2.2. Fundamentación Teórica	6
2.2.1. Comunicación Satelital	6
2.2.2. Sistema Global de Navegación por Satélite GNSS	11
2.2.3. Telefonía Móvil	20
2.2.4. Software y Hardware Libres	25
2.2.5. Servidores	29
2.2.6. Sistema Eléctrico de un Automóvil	35
2.3. Propuesta de Solución	40
CAPÍTULO III	41
3.1. Modalidad de la Investigación	41

3.2.	Población y Muestra.....	41
3.3.	Recolección de Información.....	41
3.4.	Procesamiento de Información.....	42
3.5.	Desarrollo del Proyecto.....	42
CAPÍTULO IV		43
4.1.	Diagramas de Bloques del Prototipo.....	43
4.1.1.	Sistema en el Automóvil.....	44
4.1.2.	Sistema de Monitoreo.....	47
4.2.	Análisis de Requerimientos	48
4.2.1.	Requerimientos del Sistema en el Automóvil	48
4.2.2.	Requerimientos del Servidor Web	56
4.3.	Selección de Hardware.....	57
4.3.1.	Hardware de Sistema en el Automóvil.....	57
4.3.2.	Hardware para Sistema de Monitoreo.....	73
4.4.	Selección de Software.....	75
4.4.1.	Software de Programación.....	75
4.4.2.	Software de Diseño de Circuitos PCB.....	76
4.4.3.	Software de Servidor Web	76
4.4.4.	Software para Programación de Aplicación Web	77
4.5.	Procesamiento de Información Recibida de GPS	78
4.6.	Envío de Datos a través de GSM/GPRS	81
4.6.1.	Envío de SMS por Comandos AT	82
4.6.2.	Envío de Datos a Servidor por GPRS	82
4.7.	Diagrama de Flujo del Sistema de Rastreo.....	83
4.8.	Implementación de Servidor.....	85
4.8.1.	Instalación de Raspbian	85
4.8.2.	Instalación de LAMP	86
4.8.3.	Diseño de la Interfaz Web.....	88
4.9.	Servidor en Internet	91
4.10.	Recepción de Datos GPRS	92
4.11.	Estructura Eléctrica del Automóvil	93
4.12.	Diagramas del Circuito y PCB	95
4.13.	Pruebas de Funcionamiento	97
4.14.	Presupuesto.....	104
4.14.1.	Costo del Proyecto.....	105

4.14.2. Empresas de Servicios de Monitoreo Web.	106
CAPÍTULO V	110
5.1. Conclusiones	110
5.2. Recomendaciones	110
BIBLIOGRAFÍA	112
ANEXOS	118
Anexo A Programación Arduino NANO	118
Anexo B Programación de Pagina WEB	127
Anexo C U-blox NEO-6M Chip	134
Anexo D SIM 900 Chip Especificaciones	136
Anexo E TCP Client Connection	138
Anexo F Opto acoplador 4N25	140
Anexo G Transistor 2N3904	146

ÍNDICE DE FIGURAS

Figura 2.1. Primera Ley de Kepler	8
Figura 2.2. Segunda Ley de Kepler, ley de las áreas.....	9
Figura 2.3. Frecuencias de Bandas Satelitales.....	10
Figura 2.4. Segmentos de GNSS.....	11
Figura 2.5. Comparación de Alturas de Satélites de Navegación.....	13
Figura 2.6. Satélites que componen el sistema GPS.....	14
Figura 2.7. Distribución de sistemas terrestres de monitoreo GPS en el mundo.....	15
Figura 2.8. Satélites de GLONASS desde 2001 hasta 2012.....	17
Figura 2.9. Evolución del Sistema de Navegación GLONASS.....	18
Figura 2.10. Evolución de Redes Móviles.....	23
Figura 2.11. Arquitectura de la red GSM.....	24
Figura 2.12. Diagrama General de un sistema Eléctrico de Automóvil.....	35
Figura 2.13. Diagrama de arranque de un automóvil.....	39
Figura 4.1. Diagrama de Bloques del Sistema Completo	44
Figura 4.2. Temperatura media de Máxima Absoluta	51
Figura 4.3 Temperatura media de Mínima Absoluta	52
Figura 4.4 Composición del Módulo Procesador de Datos.....	57

Figura 4.5 Cobertura de la Señal 2G de Movistar en el país.....	62
Figura 4.6 Módulos de Recepción y Transmisión RF.....	64
Figura 4.7 Etapa de Apagado de Motor.....	65
Figura 4.8 Esquema de Relé con transistor 2N3904.....	66
Figura 4.9 Etapa de Adquisición de Señal de Sirena.....	67
Figura 4.10 Representación Gráfica de I_F vs CTR.....	68
Figura 4.11 Módulo XL4005, Reductor de Voltaje (BUCK CONVERTER).....	70
Figura 4.12 Baterías Recargables UltraFire.....	70
Figura 4.13 Módulo Cargador y Protector de Baterías TP-4056.....	71
Figura 4.14 Módulo Elevador de Voltaje Boost Converter XL6009.....	71
Figura 4.15 Entorno de Programación Arduino.....	75
Figura 4.16 Entorno de Desarrollo EAGLE.....	76
Figura 4.17 Tramas del protocolo NMEA 0183.....	79
Figura 4.18 Códigos NMEA receptados por GPS UBLOX NEO6M.....	79
Figura 4.19 Datos de GPS con Librería TinyGPS ++.....	81
Figura 4.20 Diagrama de Programación de Sistema de Rastreo.....	84
Figura 4.21 Instalación de Raspbian en Raspberry Pi3.....	86
Figura 4.22 Archivos de Aplicación WEB.....	88
Figura 4.23 Diseño de Pagina Web Principal.....	90
Figura 4.24 Página de Inicio de Sesión.....	90
Figura 4.25 Apertura de Puertos para Servidor en Modem HG520C de CNT.....	92
Figura 4.26 Script para Recibir los Datos desde el Sistema de Rastreo.....	93
Figura 4.27 Sistema Eléctrico del Automóvil.....	94
Figura 4.28 Cableado del Automóvil.....	95
Figura 4.29 Esquemático del Sistema de Rastreo.....	96
Figura 4.30 PCB del Circuito de Rastreo Satelital.....	96
Figura 4.31 Baquelita de Prototipo.....	96
Figura 4.32 Equipo de Sistema de Rastreo.....	97
Figura 4.33 Sistema de Rastreo Instalado en Automóvil.....	97
Figura 4.34 Antena GPS ubicada en Automóvil.....	98

Figura 4.35 Encendido del Sistema.....	98
Figura 4.36 Bloqueo y Desbloqueo de Motor.....	99
Figura 4.37 Cierre y Apertura de Puertas.....	99
Figura 4.38 Envío de Datos a Servidor y Celular.....	99
Figura 4.39 Envío de Notificaciones de Alarma.....	100
Figura 4.40 Medición del sistema con Pinza Amperimétrica.....	100
Figura 4.41 Tiempos de Registro de cada muestra en Servidor Web.....	101
Figura 4.42 Comparación de Ubicación del Sistema.....	102
Figura 4.43 Raspberry Pi3 con Acceso a Internet.....	103

ÍNDICE DE TABLAS

Tabla 4.1 Bandas de Frecuencias de Operadoras Móviles en Ecuador.....	48
Tabla 4.2 Sistemas de "Posicionamiento Global.....	49
Tabla 4.3 Análisis de Temperaturas en el país.....	51
Tabla 4.4 Tarjetas de Desarrollo y Empresas.....	56
Tabla 4.5 Especificaciones Técnicas de Arduino NANO.....	58
Tabla 4.6 Comparación de Módulos GSM/GPRS.....	60
Tabla 4.7 Comparación de Módulos GPS.....	62
Tabla 4.8 Comparación de Pantallas para Arduino.....	71
Tabla 4.9 Comparación de Mini Ordenadores OpenSource.....	73
Tabla 4.10 Distribución de Datos de la Trama GGA.....	78
Tabla 4.11 Comandos AT para enviar un SMS.....	81
Tabla 4.12 Comandos AT para enviar Datos a Servidores.....	82
Tabla 4.13 Pasos para instalar LAMP y FTP en Raspbian.....	86
Tabla 4.14 Tiempos al Ejecutar los comandos en el Sistema.....	100
Tabla 4.15 Tiempos Registrado en Servidor	101
Tabla 4.16 Tiempos medidos para el comando "posición".....	101
Tabla 4.17 Presupuesto del Sistema.....	104
Tabla 4.18 Costo de Comercialización.....	105
Tabla 4.19 Servicios de Hunter.....	106
Tabla 4.20 Características de Servicios Chevystar.....	106
Tabla 4.21 Características de Servicios de Tracklink.....	106
Tabla 4.22 Soluciones de TraigoGPS	107
Tabla 4.23 Comparación de Sistemas.....	108

RESUMEN

La protección de un vehículo ya no es considerada un lujo, más bien es una necesidad debido al robo de automóviles que ocasiona altas pérdidas económicas en los ciudadanos, ante esta situación es necesaria la instalación de un sistema de seguridad en los mismos, mediante sistemas de alarma o de rastreo GPS. En el mercado ecuatoriano existen distintas empresas que ofrecen servicios de rastreo y monitoreo de un automóvil, sin embargo, sus costos no son accesibles para todos los estratos sociales.

El presente proyecto de investigación permite el desarrollo de un Sistema de Monitoreo para un automóvil Hyundai i10, que brinde los servicios de rastreo GPS, seguridad remota que consiste en la apertura y cierre de puertas, apagado del motor y notificaciones de alarma, mediante mensajes de texto; además, un servidor web para almacenar las coordenadas geográficas de ubicación del automóvil; utilizando hardware y software libre con el fin de reducir su costo.

El sistema de monitoreo web se considera como un producto destinado a personas particulares, así como a desarrolladores o ingenieros. Con este sistema el usuario no tiene dependencia de los servidores de empresas del proveedor para controlar su automóvil, más bien tiene todo el sistema para su propio uso. Gracias a esto desarrolladores o ingenieros pueden cambiar el sistema para sus propias necesidades y cambiar los diseños de hardware y software sin ninguna restricción.

Palabras Clave: Hardware Libre, Software Libre, GPS, GSM.

ABSTRACT

The protection of a vehicle is no longer considered a luxury, rather it is a necessity due to the car theft that causes high economic losses in the citizens, before this situation it is necessary the installation of a security system in the same, through systems of Alarm or GPS tracking. In the Ecuadorian market there are different companies that offer services of tracking and monitoring of a car, however, their costs are not accessible to all social strata.

The present research project allows the development of a Monitoring System for a Hyundai i10 car, providing GPS tracking services, remote security consisting of opening and closing doors, engine shutdown and alarm notifications, by means of messages text; In addition, a web server to store the geographical coordinates of location of the automobile; Using hardware and free software in order to reduce its cost.

The web monitoring system is considered as a product intended for individuals, as well as developers or engineers. With this system the user has no dependence on the supplier's company servers to control his automobile, rather he has the whole system for his own use. Thanks to this developers or engineers can change the system to their own needs and change the designs of hardware and software without any restrictions.

Key Words: Open Hardware, Open Software, GPS, GSM.

GLOSARIO DE TÉRMINOS

- **ADSL:** Asymmetric Digital Subscriber Line, es un tipo de tecnología de línea de abonado digital. Se usa en los pares de cobre de línea telefónica, para transportar paquetes de datos sobre estos.
- **AJAX:** acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas.
- **API:** Interfaz de Programación de Aplicaciones, es un conjunto de subrutinas, funciones y procedimientos que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.
- **Arduino:** Empresa que fabrica placas de adquisición de datos con microcontroladores ATMEL.
- **ARM:** Es un tipo arquitectura RISC basada en procesadores de 32 y 64 bits con un bajo consumo de energía.
- **AT:** es un lenguaje desarrollado por la compañía Hayes Communications que se convirtió en un estándar abierto de comandos para configurar y parametrizar módems.
- **ATMEL:** es una compañía de semiconductores, fundada en 1984, dedicada a la construcción de microcontroladores, Arduino utiliza los microcontroladores de esta empresa en sus placas.
- **Baud Rate:** es el número de unidades de señal por segundo, considerada también como velocidad de comunicación serial.
- **BEIDOU:** Sistema de Posicionamiento creado por CHINA.
- **Bootstrap:** conjunto de herramientas de Código abierto para diseño de sitios y aplicaciones web.
- **CSS:** Hojas de Estilo en Cascada (Cascading Style Sheets) es el lenguaje utilizado para describir la presentación de documentos HTML o XML.
- **CTR:** Razón de Transferencia de Corriente. Es la relación de corrientes entre la entrada y salida del opto acoplador.
- **DDNS:** Sistema de Nombre de Dominio Dinámico, se utiliza para dar un dominio a una IP variable
- **FTP:** Protocolo de Transferencia de Archivos') en informática, es un protocolo de red para la transferencia de archivos entre sistemas conectados a una red TCP.

- **GALILEO:** Sistema de Posicionamiento creado por la Unión Europea.
- **GLONASS:** Sistema de Posicionamiento creado por la Unión Soviética.
- **GNSS:** sistema global de navegación por satélite, es una constelación de satélites que transmite rangos de señales utilizados para el posicionamiento y localización.
- **GPRS:** Tecnología de comunicación móvil para envío de datos a través de internet, también se la conoce como 2.5G.
- **GPS:** Sistema de Posicionamiento Global de Estados Unidos.
- **GSM:** Tecnología de comunicación móvil diseñada para teléfonos celulares.
- **HFE:** de un transistor es el factor de ganancia o amplificación de corriente de un transistor, es el factor por el cual se amplifica la corriente de base que se alimenta al transistor.
- **HTTP:** protocolo de transferencia de hipertexto) es el protocolo de comunicación que permite las transferencias de información en la World Wide Web.
- **I2C:** Tipo de comunicación serial diseñada para interactuar con diversos periféricos.
- **IoT:** Internet de las cosas es un concepto que se refiere a la interconexión digital de objetos cotidianos con internet.
- **IP:** un número que identifica, de manera lógica y jerárquica, a una Interfaz en red de un dispositivo que utilice el protocolo IP.
- **JavaScript:** Lenguaje de programación orientado a objetos.
- **JQUERY:** es una biblioteca multiplataforma de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML.
- **LAMP:** es el acrónimo usado para describir un sistema de infraestructura de internet que usa las siguientes herramientas Linux, Apache, MySQL, PHP.
- **MySQL:** es un sistema de gestión de bases de datos relacional.
- **NAT:** Network Address Translation es un mecanismo utilizado por routers IP para intercambiar paquetes entre dos redes que asignan mutuamente direcciones incompatibles.
- **NMEA:** protocolo de transferencia de datos entre receptores GPS.
- **No-IP:** Empresa que ofrece servicios de DDNS gratis o de pago.
- **PCB:** Placa de Circuito Impreso, construida sobre cobre para el diseño de circuitos.

- **Raspberry Pi:** Empresa que fabrica ordenadores de tamaño pequeño.
- **Raspbian:** Sistema Operativo derivado de debían y optimizado para procesadores ARM.
- **RF:** Radio Frecuencia.
- **Script:** Archivo o programa con un conjunto de órdenes para ejecutar diversos procesos.
- **SMS:** Servicio de Mensajes Cortos, o comúnmente denominado mensajes de textos celulares.
- **SSH:** Interprete de Ordenes Seguro, sirve para acceder a máquinas remotas a través de una red

INTRODUCCIÓN

En el presente proyecto se expone como realizar un sistema de monitoreo web de un automóvil; el cual tenga los servicios de rastreo GPS, un servidor ubicado en el hogar del propietario; además de un control remoto que consiste en cierre y apertura de puertas, bloqueo de motor y notificaciones de alarma en caso de robo mediante mensajes de texto. Todas estas características podrán ser cambiadas o alteradas por parte del propietario, para mejorar su sistema, dependiendo de sus necesidades; gracias al uso de Hardware y Software libre para abaratar su costo.

El trabajo investigativo se realizó en cinco capítulos.

En el primer capítulo se describe el planteamiento del proyecto y los factores que llevaron a su realización para proceder, por medio de objetivos, a solucionar el problema.

En el segundo capítulo se expone los antecedentes investigativos que tengan relación con el tema propuesto a resolver, además se describe los conceptos básicos para resolver el problema y de una propuesta de solución.

En el tercer capítulo se exponen las modalidades de investigación que se tomaron para desarrollar el proyecto, así mismo como sus actividades para cumplir los objetivos propuestos.

En el cuarto capítulo se desarrolla la investigación necesaria para resolver el problema, se describe las distintas empresas que ofrecen los servicios, los requerimientos técnicos del sistema, la selección de hardware y software libre para construir el prototipo, la implementación de un servidor web y las pruebas de funcionamiento de todo el sistema.

En el quinto capítulo se establecen las conclusiones y recomendaciones finales obtenidas del proyecto

CAPÍTULO I

EL PROBLEMA

1.1. Tema

“SISTEMA DE MONITOREO WEB PARA VEHICULOS MEDIANTE HARDWARE LIBRE.”

1.2. Planteamiento del problema

Cifras de Estadísticas de la Administración de Seguridad Vial (NHTSA) señalan que en Estados Unidos un auto es robado cada 45 segundos. Se estaría hablando de miles de automotores al mes. Uno de los principales objetos de interés es la radio del automóvil y su computador, seguido del equipaje o dispositivos dejados en su interior y por último el robo de todo el automóvil. En el año 2013 en el estado de California se notificaron 171,036 autos robados, mientras que en el año 2014 disminuyó a 159,271 un equivalente de 6.9% de un año a otro. [1]

En el panorama local de Ecuador el INEC en su Encuesta de Victimización y Percepción de Inseguridad del año 2011 revela que 1 de cada 100 hogares ecuatorianos han sido víctimas de robo total de automotor, es decir la pérdida completa del vehículo, y 10 de cada 100 hogares han sido víctimas de un robo parcial de su vehículo, es decir, de los artículos dentro del automóvil. [2]

Quito capital de Ecuador registra un número alarmante en robos a automóviles. 4.8 autos diariamente fueron robados en el transcurso del año 2014 lo que indica un incremento del 6% con relación al año 2013. De acuerdo con cifras que maneja la Comandancia del Distrito Metropolitano de Quito, el año 2014 tuvo un promedio de autos robados de 1753, que son 93 unidades más con referencia al año 2013. Mientras que a nivel nacional se registraron 7053 en 2013 y 6937 en 2014 lo que demuestra que a nivel nacional se redujo un 8% el índice de robos de automotores. Un punto clave de las estadísticas que refleja el robo de autos es que el 80% de estos autos son robados cuando están parqueados en la vía pública. [3]

En la Encuesta de Estratificación del Nivel Socioeconómico del año 2011, realizada por el Instituto Nacional de Estadística y Censos clasifica el estrato social de Ecuador en cinco clases denotadas con las letras A, B, C, C- y D; ordenadas de mayor estrato social a menor. La encuesta reflejó que el 1,9% de los hogares se encuentra en estrato A, el 11,2% en nivel B, el 22,8% en nivel C+, el 49,3% en estrato C- y el 14,9% en nivel D. [4]

El estrato A es considerado alto, el B y C es considerado medio y C- es medio bajo. Los estratos sociales A, B y C poseen un automóvil como mínimo para uso como medio de transporte o trabajo; pocas familias del estrato social C- tienen posibilidad de adquirir un automóvil. En el ámbito económico según la Encuesta Nacional de Ingresos y Gastos en Hogares Urbanos y Rurales [5] de los años 2011 al 2012 realizada por el Instituto Nacional de Estadística y Censos detalla que el ingreso total promedio mensual en Ecuador es de 892,9 dólares promedio mensual. En el área urbana el ingreso promedio es de 1.046,3 dólares, mientras en el área rural el ingreso es de 567,1 dólares.

En la provincia de Tungurahua 0,5 hogares de 1000 sufrieron robo de su automóvil ubicándose como una provincia de bajas estadísticas en este tema, pero en el robo parcial de vehículo, es decir de sus objetos internos, la provincia ocupa el primer lugar teniendo que 17,35 de 1000 hogares fueron víctimas de este hecho. La ciudad céntrica Ambato tiene un índice de 4.977 hogares de robo de automóvil, de un total de hogares de 24.771 que poseen un vehículo, teniendo así un índice de criminalidad de 20,1% siendo la segunda ciudad de Ecuador en tener un alto índice; mientras tanto en robo parcial de automóvil se encuentra un total de 6344 hogares. [2]

Tomando en cuenta los índices de robo de automóviles antes descritos existe la necesidad de instalar sistemas de seguridad en los mismos, para brindar seguridad contra posibles robos. Sin embargo, los sistemas de rastreo satelital con monitoreo web no son accesibles para todos los estratos sociales del país, de esta manera los estratos medio y medio bajo que poseen un automóvil, únicamente instalan sistemas de seguridad de bajo costo, como alarmas con seguros centralizados.

1.3. Delimitación del problema

Delimitación de Contenidos

- Área Académica: Física y Electrónica
- Línea de Investigación: Sistemas Electrónicos
- Sublíneas de Investigación: Sistemas Embebidos

Delimitación Espacial

El presente trabajo se realizó en la ciudad de Ambato provincia de Tungurahua.

Delimitación Temporal

El proyecto se realizó en el transcurso de 14 meses, desde el 21 de diciembre del 2015 al de febrero del 2017, después de su aprobación por parte del Honorable Consejo Directivo de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial.

1.4. Justificación

Los altos índices de robo de automóviles en la ciudad de Ambato sobrepasan los miles de afectados, de igual manera en la ciudad, según datos de la encuesta de Victimización y Percepción de Inseguridad [2], realizado en el año 2011, posee un alto índice de criminalidad ubicándola entre los primeros puestos en comparación a otras ciudades. Con el transcurso del tiempo dichas cifras no reducen, más bien aumentan, debido a que cada año más autos salen al mercado. Según los estratos sociales existentes en la Republica de Ecuador las familias optan por comprar un auto de segunda mano o un automóvil nuevo a costos económicos, estos automóviles no cuentan con un sistema de alarma, mucho menos con un sistema de localización del vehículo frente a robos.

Para dar seguridad a los automóviles sus propietarios adquieren sistemas de seguridad básicos que constan de sistemas de alarmas sonoras y bloqueos centralizados de puertas. Este sistema de seguridad es eficiente, pero tiene características limitadas, por ejemplo, en caso de robo del automóvil, la alarma del sistema se activa y suena su sirena, esto no es suficiente para alertar al propietario de que su vehículo está siendo robado, si este se encuentra fuera del rango auditivo de la misma, además, en caso de desaparecer el automóvil, estos sistemas de seguridad no ofrecen método alguno para su posible localización.

Para mejorar los sistemas convencionales de seguridad en un automóvil se reemplazan por sistemas más avanzados como sistemas de rastreo GPS, mediante monitoreo web, los cuales envían notificaciones a su propietario de la posición del vehículo en caso de robo. Estos sistemas brindan mayor seguridad, pero el costo es un problema para los estratos sociales medio y medio bajo del país, debido a esto no todos pueden adquirir estos sistemas y optan por los sistemas de alarmas básicos.

El diseño del proyecto de investigación da una alternativa económica ante los sistemas de rastreo y monitoreo web de un automóvil, dando al usuario un sistema estable que cumpla con las funciones de rastreo GPS, control remoto como apertura de puertas, apagado de motor y notificación de alarma mediante mensajes de texto, todo esto gracias al uso de hardware y software libre. A su vez otorga al usuario el control total del sistema, sin depender de pagos mensuales o renovaciones de licencias para la utilización del sistema, a excepción de los pagos mensuales de internet o de plan de datos de un teléfono móvil, que no entran dentro del costo de sistema.

Los beneficiarios de este proyecto son personas particulares, que deseen un sistema de monitoreo web que localice su vehículo en caso de robo y posean su propio servidor web. El proyecto también va enfocado a ingenieros que deseen implementar, mejorar o cambiar, el sistema para sus necesidades, todo gracias a que el software y hardware de este son libres y cualquier persona puede acceder a ellos y utilizarlos sin ninguna restricción.

1.5. Objetivos

1.5.1. Objetivo General

Implementar un sistema de monitoreo web para localizar vehículos mediante hardware y software libre.

1.5.2. Objetivos Específicos

- Analizar la tecnología de posicionamiento global.
- Analizar el envío de datos a un servidor web utilizando telefonía móvil.
- Determinar los requerimientos técnicos y costos de hardware y software para la implementación del sistema de monitoreo web.
- Diseño del sistema de monitoreo para un automóvil.
- Diseñar la interfaz WEB para visualizar la ubicación del automóvil.

CAPÍTULO II

MARCO TEÓRICO

2.1. Antecedentes Investigativos

Se han realizado diversas investigaciones y estudios sobre el monitoreo de objetos mediante GPS. Tal es el caso de Reyes Huertas Juan David y Huertas Vitery Mario Fernando [6], que exponen el diseño de un sistema de monitoreo de automóviles por medio de GPS y GSM, dicho sistema informa por medio de alertas vía SMS al propietario del vehículo que puertas están abiertas, que velocidad tiene al auto en determinado tiempo o cual es la posición exacta del mismo en coordenadas.

Una investigación que refleja la seguridad en un automóvil es la de Juan Gabriel Acosta Calderón [7]; en su proyecto de titulación diseña un sistema de seguridad mediante tecnología RFID para el desbloqueo de puertas, además utiliza GSM para enviar las diversas alertas, que generan los sensores del automóvil, hacia un Teléfono Móvil del propietario.

La investigación de Oscar Leonardo Matza Chuncha [8], realiza un control de velocidad sobre autobuses, la velocidad con la que se desplaza un objeto puede ser obtenida por medio de un GPS, debido a que estos además de expresar la latitud y longitud del objeto, precisan la velocidad, la altura con respecto al mar y la hora exacta dependiendo de la zona horaria donde se encuentre.

En el proyecto de graduación de Edgar Patricio Rodríguez Martínez [9] se manifiesta la utilización de un dispositivo GPS ya existente en el mercado llamado GPS TACKER 103A el cual es destinado para conocer la ubicación de volquetas del GAD Municipal de Patate. El sistema funciona con mensajes de texto que remiten la posición exacta del objeto a su central en donde los datos se guardan en un archivo.csv para su posterior su posterior uso.

Valeria Carolina Aguirre Salas y Dennys Magdalena Zambrano Rosado [10] en su investigación proponen la obtención de los datos de un GPS por medio de módulos de

radiofrecuencia. Dichos datos de posicionamiento llegan a su central respectiva y son almacenados para que el usuario los emplee de cualquier manera. Esta es una forma de obtener los datos de posicionamiento de un objeto de manera remota con la ayuda de microcontroladores que gestionen el envío de datos.

La investigación de Marcia Lorena Chuquitarco y Ronny Jairo Naranjo Santiana [11] proporciona un sistema completo de monitoreo a un automóvil por medio de un módulo AVL, y módulos GPS, GSM y GPRS. Los datos obtenidos por los módulos son enviados a un software con licencia quien gestiona dichos datos y visualiza todo en su interfaz.

La investigación de Ortiz Correa Teodoro Miguel [12] propone el diseño de una aplicación web para rastreo satelital de un vehículo. Parte del sistema consta de un GPS TRACKER 106 y un Servidor WEB creado en Linux con sus respectivas bases de datos. Para el envío de los datos del GPS al Servidor, el mismo sistema incorpora una tarjeta SIM que utiliza GPRS con lo cual realiza la comunicación vía internet con el servidor. Los datos son procesados dentro del servidor y expuestos en una aplicación con la ayuda de Google Maps para ubicar el vehículo.

2.2. Fundamentación Teórica

2.2.1. Comunicación Satelital

La comunicación satelital es uno de los pilares fundamentales que tuvo el progreso de la comunicación en el ser humano, gracias a esta tecnología, se puede estar conectado en cualquier parte del mundo gracias a los satélites que orbitan alrededor de nuestro planeta. Las características de los satélites de comunicación es poder recibir y enviar información a través del espacio mediante ondas de radio en cualquier dirección que se tenga previsto en su diseño con la característica de cubrir grandes áreas geográficas del planeta, en comparación con otra tecnología de comunicación inalámbrica. Normalmente lo hacen desde y hacia grandes áreas de la Tierra, y en algunos casos también de y hacia otros satélites.

Estos satélites se ubican a diversas alturas con respecto a la tierra las cuales son las denominadas orbitas las cuales son:

- **Órbita terrestre baja o LEO:**

Los LEO se encuentran a una altura menor de 5.000 Km, y la mayoría se encuentran entre los 500 y los 1.600 Km. El número mínimo de satélites para obtener un área de cobertura total de la tierra es de 50, el periodo de orbita de estos varía entre los 90 minutos y 2 horas, los ángulos de inclinación varían entre 45 y 90°. Las orbitas terrestres de baja altura prometen un ancho de banda extraordinaria y una latencia reducida de pocas centésimas de segundo. [13]

- **Órbita terrestre media o MEO:**

Los satélites de órbita terrestre mediana también denominados ICO se, encuentran a una altura comprendida entre los 10.000 y 20.000 Km, sus ángulos de inclinación son de 45 y 90°. A diferencia de los GEO, su posición relativa respecto a la superficie terrestre no es fija, al estar a una altitud menor, se necesita un número mayor de satélites, aproximadamente 10 para tener cobertura mundial, sin embargo, tiene una reducción considerable de retraso. Los sistemas basados en satélites de orbitas de una altura intermedia, operan de manera similar a como lo hacen los sistemas de orbitas bajas. Sin embargo, su movimiento relativo a la superficie terrestre es mucho más lento, por lo que el retraso entre células es menos frecuente. En la actualidad no existen muchos satélites MEO, los que existen se ocupan para posicionamiento o el denominado GPS. [13]

- **Órbita Geoestacionaria o GEO:**

Orbita conocida como Clarke, se encuentra a una altura de 35.848 kilómetros sobre el ecuador terrestre. A este altitud el periodo de rotación es aproximadamente de 24 horas, por lo que parece estar fijo sobre la superficie de la tierra, se necesita como mínimo 3 satélites en esta orbita para cubrir la superficie del planeta. [13]

2.2.1.1. Leyes de Kepler

Las orbitas de los satélites se rigen a 3 leyes fundamentales descubiertas por el astrónomo alemán Johannes Kepler, dichas leyes rigen el movimiento de cualquier cuerpo en el espacio. Las leyes de Kepler se pueden aplicar a dos cuerpos cualesquiera en el espacio que interactúen por gravitación. El mayor de los cuerpos se le llama primario y el menor se lo llama secundario o satélite. En el caso de nuestro planeta se

utiliza como primario a la Tierra y como secundario a cualquier satélite de comunicaciones que orbite alrededor de la misma.

- **1ra Ley de Kepler**

Establece que un satélite describe una órbita alrededor de un cuerpo primario (como la Tierra) siguiendo una trayectoria elíptica. Una elipse tiene dos focos, y el centro de masa, llamado baricentro, de un sistema de dos cuerpos siempre está en uno de los focos. Como la masa de la Tierra es mucho mayor que la del satélite, el centro de masa siempre coincide con el centro de la Tierra. Las propiedades geométricas de la elipse se suelen referir a uno de los focos, que lógicamente se selecciona como el que está en el centro de la Tierra. [14]. En la Figura 2.1 se aprecia la primera Ley de Kepler

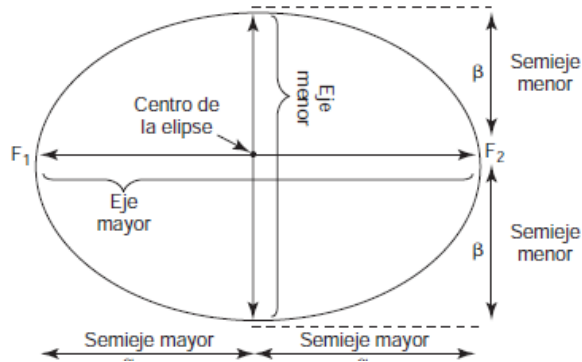


Figura 2.1. Primera Ley de Kepler,

Fuente: T. Wayne, Sistemas de Comunicaciones Electrónicas. [14]

Para el semieje mayor (α) y el semieje menor (β) de la figura 2.1, la excentricidad ϵ de la elipse se define como la ecuación (2.1):

$$\epsilon = \frac{\sqrt{\alpha^2 - \beta^2}}{\alpha} \quad (2.1)$$

2da Ley de Kepler

Se conoce como la ley de las áreas, y establece que, para intervalos iguales de tiempo, un satélite barre áreas iguales en el plano de su órbita, con foco en el baricentro. En la Figura 2.2 se aprecia la segunda Ley de Kepler.

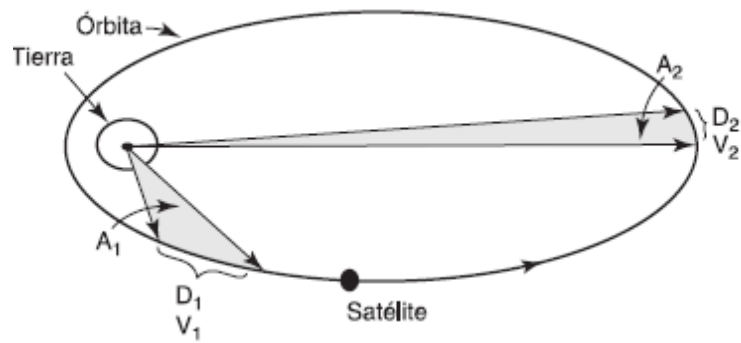


Figura 2.2. Segunda Ley de Kepler, ley de las áreas.

Fuente: T. Wayne, Sistemas de Comunicaciones Electrónicas. [14]

Debido a la ley de áreas iguales, la distancia D_1 debe ser mayor que D_2 y, en consecuencia, la velocidad V_1 debe ser mayor que la velocidad V_2 . La velocidad es máxima en el punto de máxima aproximación a la Tierra (llamado perigeo), y la velocidad es mínima en el punto más alejado de la Tierra (llamado apogeo). [14]

- **3ra Ley de Kepler**

Se llama a veces la ley armónica. Establece que el cuadrado del periodo (el tiempo en recorrer la órbita) es proporcional al cubo de la distancia promedio entre el primario y el satélite. Esta distancia promedio es igual al semieje mayor; por lo anterior, se puede enunciar matemáticamente la tercera ley de Kepler como sigue [14]:

$$\alpha = AP^{\frac{2}{3}}$$

Donde:

- A = Constante (Km)
- α = Semieje mayor (Km)
- P = Días solares medios en la Tierra

2.2.1.2. Subsistema de un Satélite

Un satélite generalmente se diseña en varios subsistemas para que al ser puesto en órbita pueda ser controlado desde la tierra. Cuenta con los subsistemas de potencia, propulsión, telemetría y comando, y el de comunicaciones. [15]

- **Subsistema de potencia**

Éste genera y distribuye potencia eléctrica de corriente directa para soportar las operaciones del satélite durante todas las fases de la misión. La potencia primaria es

proporcionada por radiación solar a través de las celdas solares de alta densidad hasta el fin de su vida; la potencia secundaria es proporcionada durante el lanzamiento y los eclipses por un sistema de baterías de níquel-hidrógeno.

- **Subsistema de propulsión**

Se trata de un sistema integral bipropelante que permite la inserción en órbita, el control de orientación y las funciones de mantenimiento en su órbita geosíncrona.

- **Subsistema de telemetría y comando**

Éste proporciona la recepción y demodulación de comandos en la banda C para su alineación en el cubo imaginario de operación, y de comandos durante todas las fases de la misión.

2.2.1.3. Bandas de Frecuencia

Como en transmisiones inalámbricas en la Tierra se tienen AM y FM comúnmente en Radio, para la comunicación satelital existen diversas bandas de frecuencias que están catalogadas según estándares de radiofrecuencias para este propósito. Las frecuencias de portadora más comunes que se usan en comunicaciones vía satélite son las bandas de 6/4 y de 14/12 GHz. El primer número es la frecuencia de enlace de subida, y el segundo es la frecuencia de enlace de bajada. Se usan frecuencias distintas de enlace de subida y de bajada para evitar interferencia entre ellas. Mientras mayor sea la frecuencia de la portadora, el diámetro necesario de la antena es menor, para una ganancia dada. La mayoría de los satélites domésticos usa la banda de 6/4 GHz, en la Figura 2.3 se detallan las frecuencias de las diversas bandas utilizadas por los satélites [14]

Banda	Frecuencias de la banda (GHz)			Ancho de banda (MHz)
	Enlace de subida	Enlace cruzado	Enlace de bajada	
C	5.9-6.4		3.7-4.2	500
X	7.9-8.4		7.25-7.75	500
Ku	14-14.5		11.7-12.2	500
Ka	27-30		17-20	—
	30-31		20-21	—
Q	—		40-41	1000
	—		41-43	2000
V (ISL)	50-51		—	1000
		54-58		3900
		59-64		5000

Figura 2.3. Frecuencias de Bandas Satelitales.

Fuente: T. Wayne, Sistemas de Comunicaciones Electrónicas. [14]

2.2.2. Sistema Global de Navegación por Satélite GNSS

El sistema global de navegación por satélites o GNSS por sus siglas en inglés consiste en un grupo de orbitas de satélites que se encuentran alrededor de la Tierra. Dichos satélites son usados para determinar el posicionamiento y la localización de cualquier objeto, sea en tierra, mar o aire. Estos grupos de satélites permiten determinar las coordenadas geográficas y el tiempo de un determinado objeto, para fines de navegación, transporte, geodésicos, hidrográficos, agrícolas y otras actividades similares. Las tecnologías satelitales que constan dentro de GNSS son las diversas infraestructuras de localización actuales en el mundo como lo son GPS, GLONASS, Galileo, Beidou. Cada de uno de estos sistemas de navegación tiene tres segmentos principales como se observa en la Figura 2.4: [16]

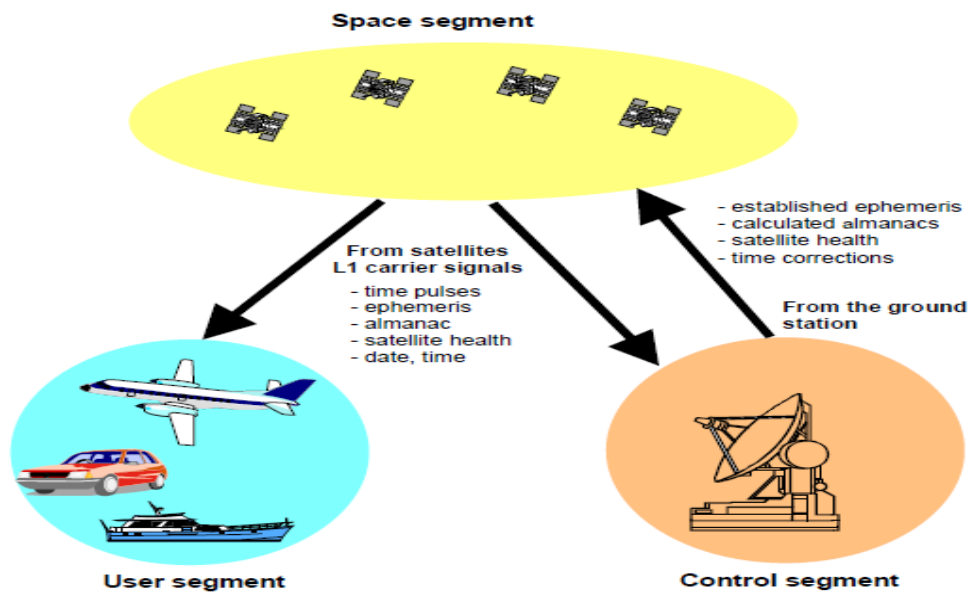


Figura 2.4. Segmentos de GNSS

Fuente: U-BLOX [16]

- **Segmento de Espacio**

En este segmento se encuentra la constelación de satélites que orbitan alrededor de la tierra, dependiendo de cada Sistema de Navegación, pueda variar el número de satélites y de orbitas que tenga en su constelación alrededor de la tierra. Este segmento también describe la funcionalidad de los satélites, sus frecuencias de trabajo, el proceso de obtención de la información de localización, sus modulaciones, y los cálculos pertinentes para poder operar en el espacio. [16]

- **Segmento de Control**

Este segmento está formado por el conjunto de dispositivos de control ubicados en tierra para operar los satélites de manera remota. Se cuentan en este grupo las antenas de control de cada constelación de satélites que ofrecen el servicio de posicionamiento global, sea de GPS, GLONASS, Galileo, Beidou. Las estaciones base de cada constelación tienen como objetivo principal: [16]

1. Observar el movimiento de los satélites y computarizar la órbita del mismo en caso de fallos.
2. Seguimiento de los relojes satelitales y predicción de su comportamiento
3. Sincronización del Reloj del Satélite
4. Retransmisión de datos orbitales precisos recibidos de satélites.
5. Retransmisión de los datos orbitales aproximados de todos los satélites
6. Retransmisión de información adicional, incluida el estado de los satélites, errores de reloj, etc.

- **Segmento de Usuario**

El segmento usuario está compuesto por todos los dispositivos receptores de señales de posicionamiento global, este segmento es utilizado para ámbitos civiles o militares, dependiendo su uso. Este segmento es al que tienen acceso todas las personas y es gobernado por aplicaciones de posicionamiento tanto en autos, barcos, aeronaves e inclusive objetos. En este segmento no se puede controlar los satélites, ni alterar ninguna configuración de los mismos, debido a que este segmento solo se trabaja con la información recibida de los satélites.

En la actualidad no solo existe un sistema de posicionamiento con el que se puede obtener coordenadas geográficas, altitud, tiempo, etc. Varios países del mundo o grupos de países desarrollaron diversos sistemas de posicionamiento, los cuales son, para uso civil o militar, en la Figura 2.5 se muestran los diversos sistemas de posicionamiento global y su altitud sobre la tierra.

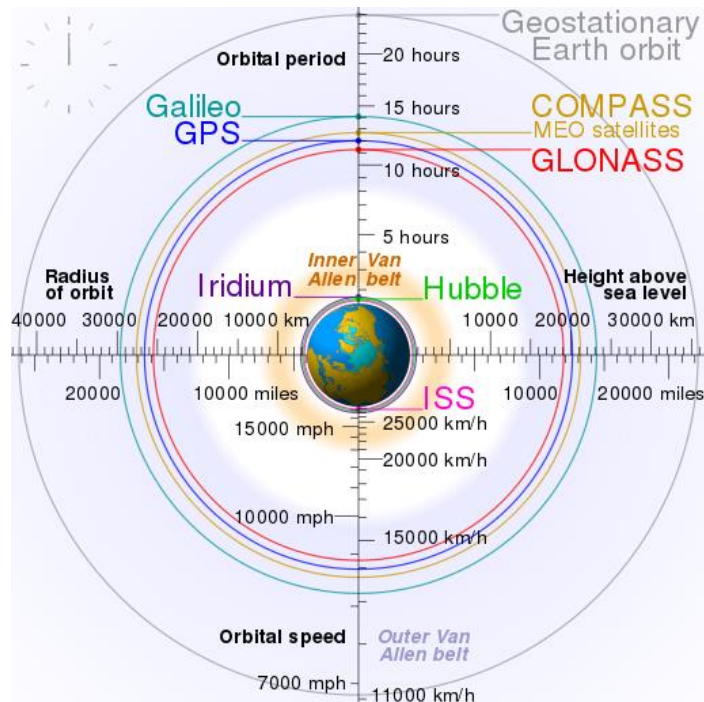


Figura 2.5. Comparación de Alturas de Satélites de Navegación

Fuente: [17]

Los más relevantes hasta la fecha son:

2.2.2.1. GPS

El Sistema de Posicionamiento Global GPS es un sistema que permite determinar la posición de un objeto en cualquier parte de la Tierra, su inicio fue de ámbito militar, mas no de uso comercial. También es conocido como NAVSTAR, es el pionero y precursor en la tecnología GNSS de posicionamiento global. Este sistema está constituido por tres segmentos: el segmento espacial, el segmento de control y el segmento del usuario. La Fuerza Aérea de los Estados Unidos desarrolla, mantiene y opera los segmentos espacial y de control. [18]

Segmento Espacial.

El segmento espacial GPS consiste en una constelación de satélites que transmiten señales de radio a los usuarios. Estados Unidos se ha comprometido a mantener la disponibilidad de al menos 24 satélites GPS operativos, el 95% del tiempo. Para asegurar este compromiso, la Fuerza Aérea ha estado manteniendo operativos 31 satélites de GPS operativos durante los últimos años. [18]

Los satélites GPS vuelan en órbita terrestre media (MEO) a una altitud de aproximadamente 20.200 km. Cada satélite rodea la Tierra dos veces al día. Los satélites de la constelación GPS están dispuestos en seis planos orbitales igualmente espaciados que rodean la Tierra con 4 satélites por órbita. Esta disposición de 24 satélites alrededor de la Tierra garantiza que los usuarios pueden ver al menos cuatro satélites desde prácticamente cualquier punto del planeta. En la actualidad se tiene 31 satélites operacionales alrededor de la Tierra combinados con satélites modernos y antiguos, en la Figura 2.6 se detalla los satélites utilizados por GPS. [18]


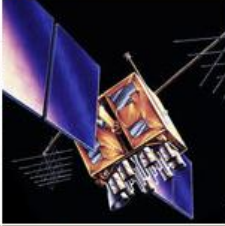
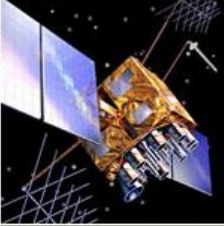
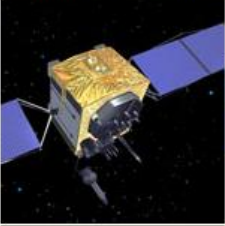

LEGACY SATELLITES		MODERNIZED SATELLITES		
				
BLOCK IIA	BLOCK IIR	BLOCK IIR(M)	BLOCK IIF	GPS III
0 operational	12 operational	7 operational	12 operational	In production

Figura 2.6. Satélites que componen el sistema GPS

Fuente: [19]

Segmento de Control

El segmento de control GPS consta de una red global de instalaciones terrestres que rastrea los satélites GPS, monitorea sus transmisiones, realiza análisis y envía comandos y datos a la constelación. El segmento de control operativo actual incluye una estación de control maestra, una estación de control maestra alternativa, 11 antenas de mando y control y 15 sitios de monitoreo, en la Figura 2.7 se detalla las todas las estaciones del segmento de control de GPS . [18]

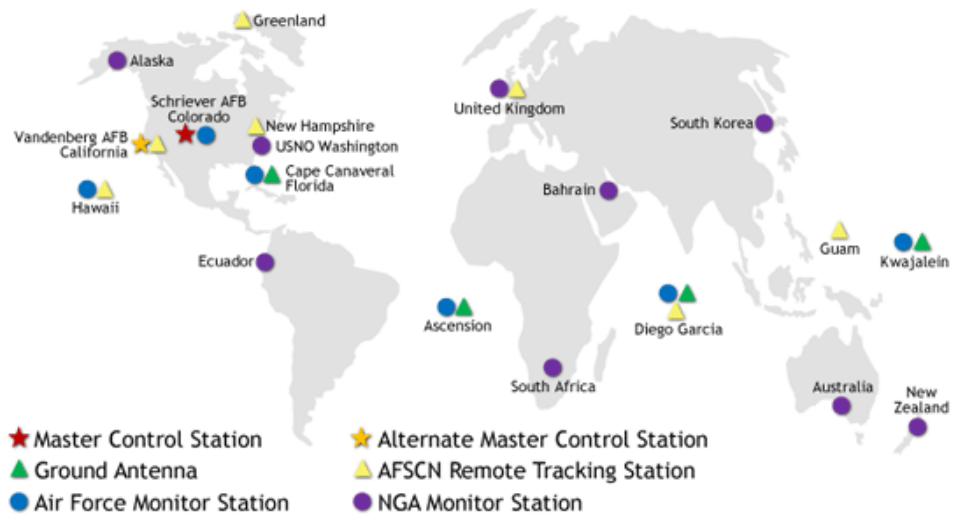


Figura 2.7. Distribución de sistemas terrestres de monitoreo GPS en el mundo.

Fuente: [20]

Estación de control maestro:

La estación de control maestro en Colorado es donde 2SOPS, el cual es un grupo de las Fuerza Aérea Estadounidense que mantienen los satélites GPS funcionando 24 horas al día, 7 días a la semana, con disponibilidad continua y alta precisión para miles de millones de usuarios civiles y militares. Ellos realizan las funciones de segmento de control primario, proporcionando el comando y control de la constelación de GPS. El MCS genera y carga mensajes de navegación y asegura la salud y precisión de la constelación de satélites. Recibe información de navegación de las estaciones de monitoreo, utiliza esta información para calcular las ubicaciones precisas de los satélites GPS en el espacio, y luego carga estos datos en los satélites. [18]

Estaciones de Monitoreo:

Las estaciones de monitoreo rastrean los satélites GPS y canalizan sus observaciones a la estación de control maestra. Las estaciones de monitoreo recogen datos atmosféricos, mediciones de rango / portador y señales de navegación. Los sitios utilizan receptores GPS sofisticados y son operados por el MCS. Hay 15 estaciones de monitoreo ubicadas en todo el mundo, incluyendo seis de la Fuerza Aérea y nueve de la Agencia Nacional de Inteligencia Geoespacial (NGA). [18]

Antenas Terrestres:

Las antenas terrestres se utilizan para comunicarse con los satélites GPS para fines de mando y control. Estas antenas soportan enlaces de comunicaciones de banda S que envían y transmiten cargas de datos de navegación y cargas de programas de procesador, y recogen la telemetría. Hay cuatro antenas terrestres GPS dedicadas ubicadas en las estaciones de monitoreo en Kwajalein Atoll, Isla de la Ascensión, Diego García y Cabo Cañaveral. Además, el segmento de control está conectado a las siete estaciones de localización remota de la Red de Control por Satélite de la Fuerza Aérea (AFSCN) en todo el mundo, aumentando la visibilidad, flexibilidad y robustez para la telemetría, el seguimiento y el comando. [18]

Segmento de Usuario

El segmento de usuario consiste en el equipo receptor del GPS que recibe las señales de los satélites GPS y las procesa para calcular la posición tridimensional y la hora precisa. Dentro de este segmento se encuentra el público en general, se puede desarrollar las aplicaciones que se desee debido a que la disposición de estas señales GPS es gratuito a todo el mundo.

2.2.2.2. GLONASS

GLONASS [21] es una abreviatura de un sistema GNSS actualmente operado por el Ministerio de Defensa Ruso por sus siglas en inglés significa Global Navigation Satellite System. El programa fue iniciado por la antigua Unión Soviética, y actualmente está bajo la jurisdicción de la Federación Rusa y que constituye el homólogo del GPS estadounidense y del Galileo europeo. Inicialmente nació como uso militar para competir con el GPS, y del mismo modo tiempo después se liberó el uso a civiles dentro del territorio Ruso. Consta de una constelación de 31 satélites (24 en activo, 3 satélites de repuesto, 2 en mantenimiento, uno en servicio y otro en pruebas) situados en tres planos orbitales con 8 satélites cada uno y siguiendo una órbita inclinada de $64,8^\circ$ con un radio de 25.510 km, en la Figura 2.8 se muestra el número de satélites que formaron parte de la constelación GLONASS desde el año 2001. La constelación de GLONASS se mueve en órbita alrededor de la Tierra con una altitud de 19.100 km, algo más bajo que el GPS, Estadounidense que se sitúa a los

20.200 km, y tarda aproximadamente 11 horas y 15 minutos en completar una órbita.
[16]

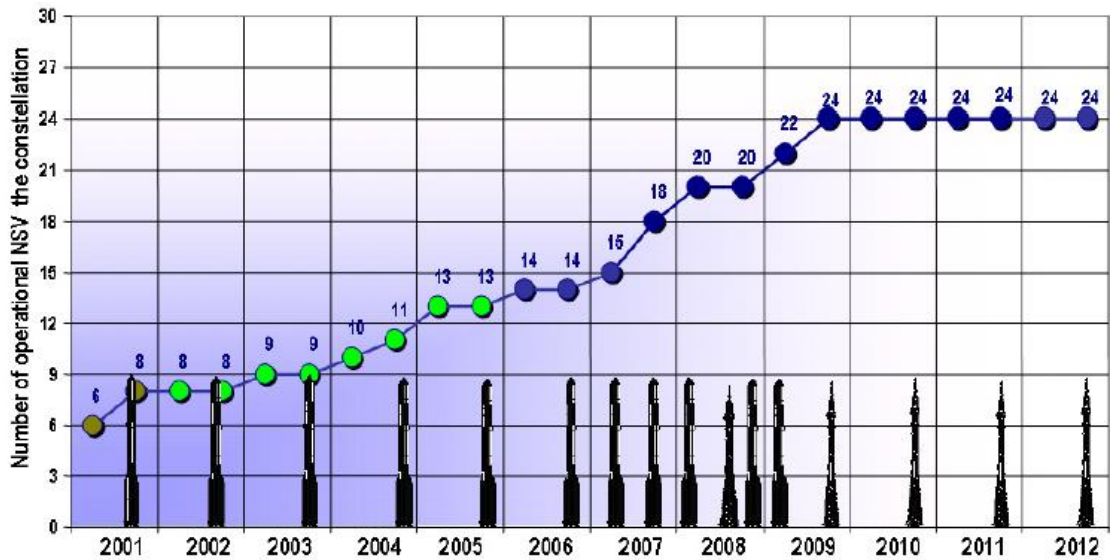


Figura 2.8. Satélites de GLONASS desde 2001 hasta 2012

Fuente: [16]

Los tres primeros satélites de prueba fueron puestos en órbita el 12 de octubre de 1982. En 2004, 11 satélites se encontraban en pleno funcionamiento. A finales de 2007 son 19 los satélites operativos. Son necesarios 18 satélites para dar servicio a todo el territorio ruso y 24 para poder estar disponible el sistema en todo el mundo. En 2007, Rusia anuncia que a partir de ese año se eliminan todas las restricciones de precisión en el uso de GLONASS, permitiendo así un uso comercial ilimitado. Hasta ahora las restricciones de precisión para usos civiles eran de 30 metros. [16]

La constelación GLONASS ha pasado por una evolución en sus satélites desde el año de 1982 hasta en un futuro 2020. Teniendo sus satélites a GLONASS, GLONASS-M, GLONASS-K, y GLONASS-K2. Actualmente se tiene en uso la tercera generación de satélites GLONASS los GLONASS-K, en la Figura 2.9 se observa la evolución de los satélites que forman parte del sistema GLONASS.

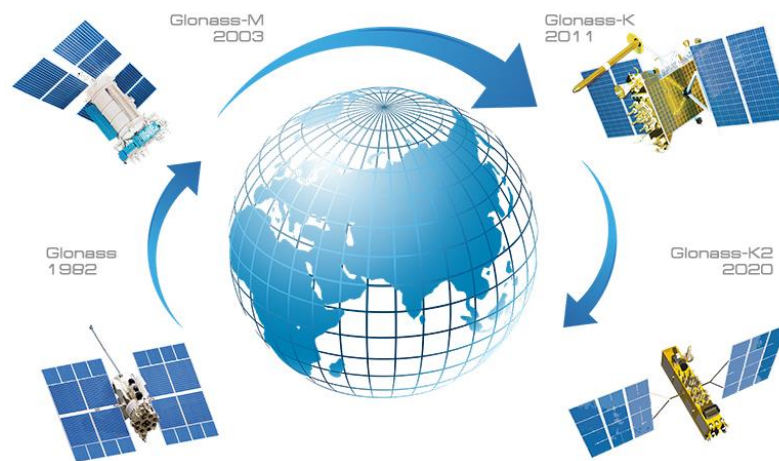


Figura 2.9. Evolución del Sistema de Navegación GLONASS

Fuente: [22]

Actualmente GLONASS trabaja a la par que GPS y a nivel mundial, todos los receptores de posicionamiento en el mercado conviven con estos dos sistemas de navegación que permiten recibir señales pertenecientes a los dos sistemas GLONASS y GPS, cada cual, con sistemas de referencia diferentes, esto hace interesantes las posibilidades de GLONASS en la medición como apoyo al GPS norteamericano.

2.2.2.3. BEIDOU

Es un sistema de posicionamiento global desarrollado por la República Popular de China de uso civil. Al igual que GPS y GLONASS BeiDou quiere ofrecer un sistema de posicionamiento con márgenes de errores reducidos dentro de China y también a todo el mundo.

El sistema de navegación por satélite BeiDou se compone de tres partes: la sección de espacio, la sección de tierra y la sección de usuario. La sección espacial contiene 5 satélites de órbita geoestacionaria y 30 satélites de órbita no geoestacionaria. La sección de tierra consiste en un cierto número de estaciones: incluyendo las estaciones de control principal, las estaciones de inyección y las estaciones de monitoreo. Y la sección de usuario los sistemas de navegación para BeiDou y sus aplicaciones, así como algunas aplicaciones compatibles con otros sistemas de navegación por satélite. [23]

La primera generación de satélites BeiDou ya está operativa desde el año 2000. Cuando se despliegue completamente todos los satélites, la constelación espacial estará formada por cinco satélites GEO (Geostationary Earth Orbit), veintisiete satélites

de órbita terrestre media (MEO) y tres satélites de órbita geo sincrónica inclinada (IGSO). Los satélites GEO operan en órbita a una altitud de 35.786 kilómetros y se sitúan a 58.75 ° E, 80 ° E, 110.5 ° E, 140 ° E y 160 ° E, respectivamente. Los satélites MEO están operando en órbita a una altitud de 21.528 kilómetros y una inclinación de 55 ° respecto al plano ecuatorial. Los satélites del IGSO están operando en órbita a una altitud de 35.786 kilómetros y una inclinación de 55 ° al carril ecuatorial. [24]

2.2.2.4. GALILEO

Galileo es el programa europeo de radionavegación y posicionamiento por satélite, desarrollado por la Unión Europea en conjunto con la Agencia Espacial Europea. Este programa dota a la Unión Europea de una tecnología independiente del GPS de Estados Unidos y el GLONASS Ruso. Al contrario de estos dos, este sistema de posicionamiento es y será de uso civil, se creó con el propósito de aumentar las características de medición, triangulación y precisión de los sistemas de posicionamientos ya existentes; además pretende revolucionar su sistema con mayor tecnología y con una interoperabilidad del 100% sin restricciones militares, diferenciándose lo que en un principio GLONAS y GPS no lo fueron. El sistema se espera poner en marcha en 2020, pero hasta la fecha ya están ubicados satélites en orbitas alrededor de la tierra.

La constelación completa de Galileo consistirá en 24 satélites más recambios. Con los satélites tomando alrededor de 14 horas para orbitar la Tierra a 23 222 km de altitud, siempre habrá al menos cuatro satélites visibles en cualquier parte del mundo. Los satélites se están colocando en tres planos orbitales en un ángulo de 56 grados con el ecuador, que proporcionará cobertura hasta las regiones polares. [25]

Se han puesto en marcha dos centros de control Galileo (GCC) en tierra europea para el control de los satélites y la gestión de su navegación. Los datos proporcionados por una red global de estaciones de sensor Galileo (GSS) se envían a los centros de control de Galileo a través de una red de comunicaciones redundante. Los GCC utilizan los datos de las Estaciones Sensores para calcular la información de integridad y para sincronizar la señal horaria de todos los satélites con los relojes de estación terrestre. El intercambio de datos entre los centros de control y los satélites se realiza a través de estaciones de enlace ascendente. [25]

Como una característica adicional, Galileo está proporcionando una función global de Búsqueda y Salvamento (SAR), basada en el sistema operacional Cospas-Sarsat. Por lo tanto, los satélites están equipados con un transpondedor, que puede transferir las señales de socorro desde los transmisores del usuario a los centros regionales de coordinación de rescate, que iniciarán la operación de rescate. Al mismo tiempo, el sistema enviará una señal de respuesta al usuario, informándole que su situación ha sido detectada y que la ayuda está en camino. Esta última característica es nueva y se considera una mejora importante en comparación con el sistema existente, que no proporciona comentarios de los usuarios. [25]

2.2.3. Telefonía Móvil

Es un sistema de comunicación que tiene la capacidad de mantener comunicado al usuario mediante su equipo terminal mientras este se encuentra movilizándose, es decir que no pierde el rastro del equipo terminal móvil a medida que cambia su ubicación. La telefonía móvil revolucionó la forma de estar conectados con el mundo, en su historia se destacan varias generaciones de tecnología móvil que se ha sacado el mercado las cuales son:

Primera Generación:

La primera generación de 1981 fue analógica con capacidad de roaming, permitía solamente llamadas de voz con baja calidad. AMPS fue el principal estándar de primera generación y se desarrolló entre los años 1982 y 1992, seguido de los estándares TAC (Sistema de comunicaciones de acceso total) y ETACS (Sistema de comunicaciones de acceso total extendido). Actualmente este sistema de Comunicación está obsoleto, no existe infraestructura en el mundo ni equipos terminales para dar servicio en esta generación. [26] [27]

Segunda Generación:

La segunda generación móvil o (2G) data del año 1990, corresponde a los sistemas como GSM, CDMAOne y TDMA o llamado D-AMPS. Son sistemas digitales con técnicas avanzadas de uso del espectro radioeléctrico y con capacidades de roaming mejoradas, que permiten la transmisión de datos y servicios con valor agregado como los SMS.

Gracias a la segunda generación, es posible transmitir voz y datos digitales de volúmenes bajos, por ejemplo, mensajes de texto o mensajes multimedia. El estándar GSM permite una velocidad de datos máxima de 9,6 kbps. Se han hecho ampliaciones al estándar GSM con el fin de mejorar el rendimiento. Algunas de esas extensiones es el servicio GPRS (Servicio General de Paquetes de Radio) que permite velocidades de datos teóricas en el orden de los 114 Kbits/s, pero con un rendimiento cercano a los 40 Kbits/s en la práctica. Como esta tecnología no se encuentra dentro de la categoría "G3", se la llama G2.5. EDGE Enhanced Data Rates for GSM Evolution es considerada como un puente entre la segunda y tercera generación de comunicaciones móviles debido a sus velocidades mejoradas en relación a GPRS, es considerado como una evolución de GPRS. Permite alcanzar velocidades de hasta 384 kbps y recibir datos móviles pesados [26] [27]

Tercera Generación:

Las especificaciones IMT-2000 (Telecomunicaciones móviles internacionales para el año 2000) de la Unión internacional de telecomunicaciones (ITU) definieron las características de la 3G (tercera generación de telefonía móvil). Las características más importantes son:

- Alta velocidad de transmisión de datos.
- 144 Kbps con cobertura total para uso móvil.
- 384 Kbps con cobertura media para uso de peatones.
- 2 Mbps con áreas de cobertura reducida para uso fijo.
- Compatibilidad mundial.
- Compatibilidad de los servicios móviles de 3G con las redes de segunda generación.

La G3 ofrece velocidades de datos de más de 144 Kbit/s y de este modo brinda la posibilidad de usos multimedia, por ejemplo, transmisión de videos, video conferencias o acceso a Internet de alta velocidad. Las redes de 3G utilizan bandas con diferentes frecuencias a las redes anteriores: 1885 a 2025 MHz y 2110 a 2200 MHz. [26]

Dentro de esta generación se encuentran algunas tecnologías de comunicación móvil intermedias, la primera que denominó el término 3G fue UMTS. Universal Mobile

Telecommunications System es una de las tecnologías usadas por los móviles de tercera generación, sucesora de GPRS, debido a que la tecnología GPRS no podía evolucionar para prestar servicios considerados de tercera generación las velocidades que puede alcanzar UMTS son de hasta 144 kbit/s sobre vehículos a gran velocidad, 384 kbit/s en espacios abiertos y 7.2 Mbit/s con baja movilidad interior de edificios.

HSPA considerado como 3.5G es una fusión de dos protocolos móviles, High Speed Downlink Packet Access (HSDPA) y High Speed Uplink Packet Access (HSUPA) que extiende y mejora el rendimiento de las redes de telecomunicaciones móviles de tercera generación. Las primeras especificaciones HSPA alcanzaba velocidades de hasta 14,4 Mbit/s en bajada y hasta 5,76 Mbit/s en subida, dependiendo del estado o la saturación la red y de su implantación

HSPA+ es considerado como 3.75G conocido como Evolved HSPA (HSPA Evolucionado), es un estándar de internet móvil definido en la versión 7 de 3GPP y posteriores. HSPA+ provee velocidades de hasta 84 Mbps de bajada y 22 Mbps de subida, a través de una técnica multi-antena conocida como MIMO (Multiple-Input Multiple-Output) y modulación 64-QAM.

Cuarta Generación:

La principal característica de la 4G es que se basa en protocolos que permiten obtener velocidades de transferencia de datos en movilidad de hasta 100 Mbps (velocidad de pico máxima teórica), o de hasta 1 Gbps en estacionario. La tecnología LTE Long Term Evolution es un estándar para comunicaciones inalámbricas de transmisión de datos de alta velocidad para teléfonos móviles y terminales de datos. Es un protocolo de la norma 3GPP definida por unos como una evolución de la norma 3GPP UMTS y por otros como 4G.

Actualmente se habla de manera comercial de LTE como una cuarta generación de comunicaciones móviles, pero algunos la consideran solo como una evolución de UMTS es decir como 3.9G una tecnología que no cumple con las características de 4G. El mercado de operadoras de telecomunicaciones ha optado por llamar a esta nueva tecnología como 4G, las velocidades que ofrece LTE son Bajada: 326,5 Mbps para 4x4 antenas, 172,8 Mbps para 2x2 antenas y una velocidad de subida de Subida: 86,5 Mbps. LTE no gestión llamadas ni SMS, solo realiza conmutación de paquetes,

no puede realizar conmutación de circuitos, de esta manera LTE es una tecnología actual que se ayuda de antiguas tecnologías para brindar los servicios completos a dispositivo móvil.

Estas son tecnologías que abren la puerta a un abanico de aplicaciones inmenso, pues haría posible la descarga de datos más rápidamente, y de forma más eficiente que una línea de ADSL corriente, o incluso que el cable o la fibra óptica en algunos casos. Utiliza antenas MIMO para su transmisión y recepción, aprovechando el NLOS para mejorar sus características. Emplea modulación OFDM, se basa completamente en el protocolo IP y posee un ancho de canal de 20 – 100 MHz. En la Figura 2.10 se visualiza las generaciones de telefonía móvil que han existido.

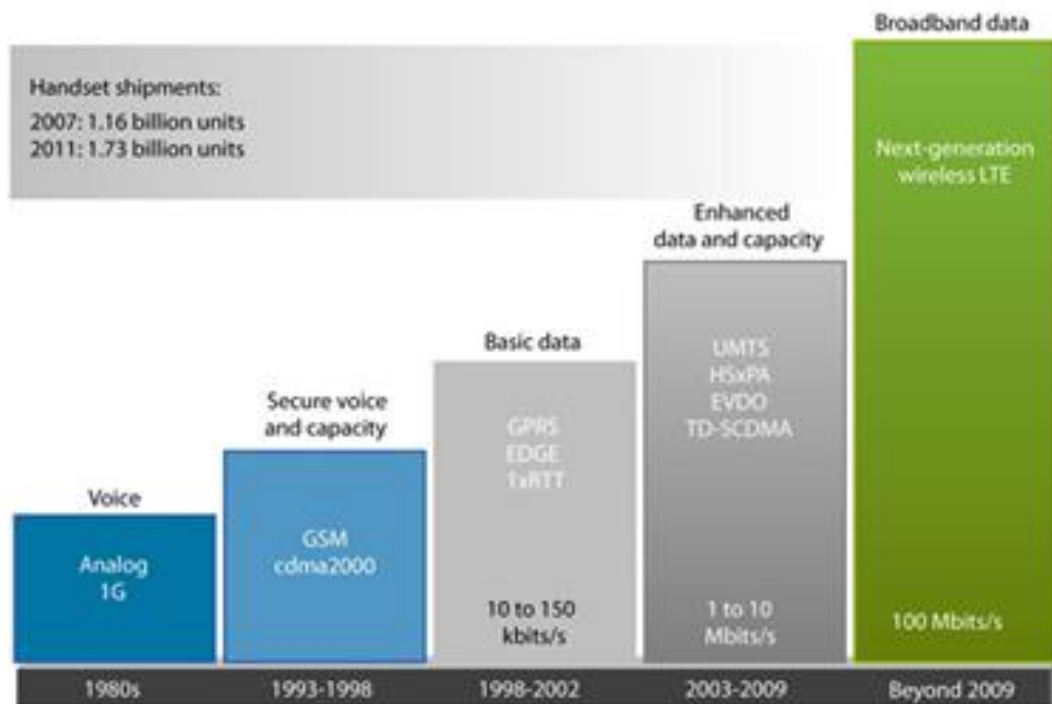


Figura 2.10. Evolución de Redes Móviles.

Fuente: [28]

2.2.3.1. Sistema GSM

Como se expresó el sistema GSM o conocido como 2G comenzó en 1991 y fue el cambio de las primeras generaciones de telefonía móvil en el mundo. Antes de GSM no existía un estándar regulado para las comunicaciones móviles en todo el mundo, por lo tanto, cada país se regía a su sistema de comunicación móvil. Pero estos inconvenientes provocaron que su servicio no funcionara más allá de los límites de

cada país. Por lo tanto, GSM cambio esto trayendo un estándar mundial, con lo cual todos los dispositivos iban a ser compatibles en cualquier lugar del mundo. Su nombre completo es Global System for Mobile Communications, Sistem Gloval de Comunicaciones Móviles. Fue desarrollado también para resolver los problemas de fragmentación inherentes a los sistemas de primera generación en Europa. Fue el primer sistema telefónico celular en ser totalmente digital. [29]

Arquitectura GSM

La arquitectura GSM consiste en tres subsistemas principales interconectados, que interaccionan entre sí y con los suscriptores a través de interfaces específicas de red. Los tres sistemas principales del GSM son: Subsistema de Estación Base BSS, Subsistema de red y conmutación NSS y Subsistema de Soporte Operacional OSS. En la Figura 2.11 se representa un diagrama de la arquitectura GSM.

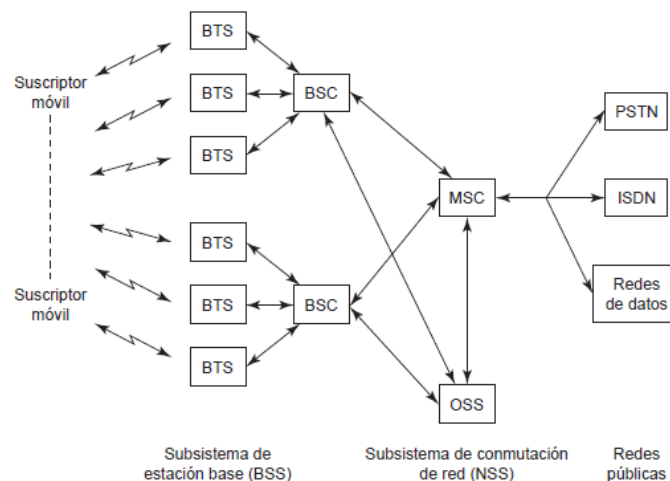


Figura 2.11. Arquitectura de la red GSM

Fuente: Sistemas de Comunicaciones Electrónicas, Wayne Tomasi [29]

BSS también se lo conoce como Subsistema de Radio, porque proporciona y administra rutas de transmisión en radiofrecuencia entre las unidades móviles y en el centro móvil de conmutación MSC. También, el BSS administra la interfaz de radio entre las estaciones móviles y todos los demás subsistemas GSM. Cada BSS consiste de muchos controladores de estación base (BSC) que se usan para conectar el MS con el NSS a través de uno o más MSC. El NSS administra las funciones de conmutación del sistema y permite que se comuniquen las MSC con otras redes telefónicas, como el servicio telefónico público y la ISDN. El OSS soporta la operación y el

mantenimiento del sistema, y permite a los ingenieros vigilar, diagnosticar y localizar las fallas en cada aspecto de la red GSM. [29]

2.2.3.2. Sistema GPRS

Por sus siglas en inglés General Packet Radio Service o Servicio General de Paquetes Vía Radio, fue creado en el año de 1980. Es una extensión del Sistema GSM que lo evoluciona y en conjunto con este se denominan la 2.5G.

Basándose en las especificaciones de la Versión 97 de la 3GPP, un grupo que regula las tecnologías móviles de comunicaciones, GPRS normalmente alcanzaba velocidades de 40Kbps en el enlace descendente y 14Kbps en el enlace ascendente agregando ranuras de tiempo GSM en un portador. Las mejoras en las versiones R'98 y R'99 significaban que GPRS podría llegar teóricamente a velocidades de enlace descendente de hasta 171Kbps. [30]

El siguiente avance en la tecnología de acceso por radio GSM fue EDGE (Enhanced Data rates for Global Evolution) o Enhanced GPRS. Con una nueva técnica de modulación que produjo un aumento de tres veces en la tasa de bits (8PSK reemplazando GMSK) y la nueva codificación de canal para la eficiencia espectral, EDGE fue introducido con éxito sin interrumpir los planes de reutilización de frecuencia de los despliegues GSM existentes. El aumento en las velocidades de los datos a 384Kbps colocó EDGE como puente temprano de 3G, aunque fue etiquetado 2.75G por los vigilantes de la industria. El trabajo estándar de 3GPP ha proporcionado una evolución en EDGE el cual está diseñado para complementar la cobertura de acceso de paquetes de alta velocidad (HSPA). EDGE Evolution tiene rendimiento espectral mejorado con latencias reducidas hasta 100ms y aumento de las velocidades de transmisión a 1,3 Mbps en el enlace descendente y 653 Kbps en el enlace ascendente. [30]

2.2.4. Software y Hardware Libres

Primeramente, se necesita conocer que software y hardware, como característica principal el software es algo que no es tangible y el hardware es algo que se puede tocar con nuestras manos. Desde este punto se tiene como software todos los programas, sistemas operativos o programación lógica que se encuentre dentro de un

dispositivo; dicho dispositivo es el hardware que es el conjunto de elementos físicos que son utilizados para funcionar para un determinado propósito.

Hardware y Software funcionan a la par, cada uno depende del otro. Este conjunto de términos ha formado toda la revolución tecnológica en la actualidad, desde computadores, relojes inteligentes, Teléfonos inteligentes, aviones, autos, refrigeradores, etc. Todo equipo electrónico actual posee estos dos términos. Dentro de estos términos se encuentra a su vez otro termino que separa de gran manera dos grandes grupos, es el termino libre; Hardware Libre y Software Libre.

2.2.4.1. Software Libre

Es el software que respeta la libertad de los usuarios y la comunidad. A grandes rasgos, significa que los usuarios tienen la libertad de ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software. Un programa es software libre si los usuarios tienen las cuatro libertades esenciales: (0) ejecutar el programa, (1) estudiar y modificar el código fuente del programa, (2) redistribuir copias exactas y (3) distribuir versiones modificadas:

- **La libertad de ejecutar.**

La libertad de ejecutar el programa significa que cualquier tipo de persona u organización es libre de usarlo en cualquier tipo de sistema de computación, para cualquier tipo de trabajo y finalidad, sin que exista obligación alguna de comunicarlo al programador ni a ninguna otra entidad específica. En esta libertad, lo que importa es el propósito del usuario, no el del programador. [31]

- **La libertad de estudiar el código fuente y modificarlo.**

Para que las libertades 1 y 3 (realizar cambios y publicar las versiones modificadas) tengan sentido, usted debe tener acceso al código fuente del programa. Por consiguiente, el acceso al código fuente es una condición necesaria para el software libre. El código fuente ofuscado no es código fuente real y no cuenta como código fuente. [31]

- **La Libertad de redistribuir copias.**

La libertad para distribuir (libertades 2 y 3) significa que usted tiene la libertad para redistribuir copias con o sin modificaciones, ya sea gratuitamente o cobrando una tarifa por la distribución, a cualquiera en cualquier parte. Ser libre de hacer esto significa, entre otras cosas, que no tiene que pedir ni pagar ningún permiso para hacerlo. [31]

- **La Libertad de distribuir copias de sus versiones modificadas a terceros**

Esto le permite ofrecer a toda la comunidad la oportunidad de beneficiarse de las modificaciones. El acceso al código fuente es una condición necesaria para ello. [31]

Licencias en Software Libre:

Licencia GPL

Licencia Pública General de GNU (GNU GPL). Se utilizan para el software libre, la adopción de esta licencia garantiza a los usuarios finales la libertad de usar, estudiar, compartir y modificar el software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios. Se usa para la mayoría de los programas de GNU. Esta licencia fue creada originalmente por Richard Stallman fundador de la Free Software Foundation para el proyecto GNU. [31]

Licencia LGPL

Licencia Pública General Reducida LGPL. Esta licencia permisiva se aplica a cualquier programa o trabajo que contenga una nota puesta por el propietario de los derechos del trabajo, estableciendo que su trabajo puede ser distribuido bajo los términos de esta "LGPL". El "Programa", utilizado en lo subsecuente, se refiere a cualquier programa o trabajo original, y el "trabajo basado en el Programa" significa ya sea el programa o cualquier trabajo derivado del mismo bajo la ley de derechos de autor: es decir, un trabajo que contenga el Programa o alguna porción de él, ya sea íntegra o con modificaciones o traducciones a otros idiomas. [31]

La principal diferencia entre la GPL y la LGPL es que la última puede enlazarse a (en el caso de una biblioteca, 'ser utilizada por') un programa no-GPL, que puede ser software libre o software no libre.

Licencia AGPL

La Licencia Pública General Affero está basada en la GPL de GNU, pero contiene una cláusula adicional que permite a los usuarios interactuar con el programa licenciado a través de una red para recibir el código fuente de ese programa. Es recomendada para cualquier programa que vaya a ejecutarse a través de una red. El propósito de GNU Affero GPL es prevenir un problema que afecta a los desarrolladores de programas gratuitos que se utilizan con frecuencia en los servidores. [31]

Licencia GFDL

La Licencia de Documentación Libre de GNU es una forma de copyleft pensada para manuales, libros de texto u otros documentos, para garantizar que todo el mundo tenga la libertad de copiar y redistribuir la obra, con o sin modificaciones, de modo comercial o no comercial. [31]

2.2.4.2. Hardware Libre

Se llama hardware libre, hardware de código abierto, electrónica libre o máquinas libres a aquellos dispositivos de hardware cuyas especificaciones y diagramas esquemáticos son de acceso público, ya sea bajo algún tipo de pago, o de forma gratuita. Hardware libre significa hardware que los usuarios son libres de usar, copiar y redistribuir con o sin cambios. Hay una diferencia entre lo que es Hardware y Software. Lo principal es que el primero es algo tangible y lo segundo es algo intangible. De esta manera la concepción de Hardware Libre cambia debido a que, a diferencia del Software, este tendrá un costo, por los materiales que emplearemos para su desarrollo. Como el Software es algo que no es físicamente construible entonces puede ser gratis porque no tiene un costo en producirlo físicamente, pero cabe recalcar que Software Libre no siempre significa que no costará nada, más bien existen diversos softwares bajo el termino libre, con licencias que lo respaldan, que cuestan dinero.

El Hardware Libre no es gratis, al momento de producir un hardware se utiliza diversos elementos electrónicos como resistencias, capacitores, diodos, transistores, microcontroladores, etc. Es claro que estos elementos cuestan dinero, entonces producir un hardware como por ejemplo una placa base para un circuito electrónico va a costar dinero se quiera o no; entonces lo que se conoce como hardware libre es el hecho de que el diseño de dicho dispositivo sea libre; es decir que nosotros podamos

crear, mejorar, u obtener de manera libre sin ninguna privación estos diseños; para posteriormente nosotros construirlo y adaptarlo a nuestras necesidades. De esta manera se engloba el termino Hardware Libre como Diseño de Hardware Libre.

Innovaciones en Licencias

Las licencias de HL constituyen un elemento fundamental en el régimen de propiedad intelectual asociada a estos bienes, al regular el uso, la copia, la modificación y la distribución de la documentación de diseño del hardware, así como la fabricación y distribución de productos. Las licencias existentes son:

Licencia de Hardware Libre TAPR: Elaborada en 2007 por la comunidad de radioaficionados TAPR (Tucson Amateur Packet Radio) con el fin de garantizar la libertad de compartir y de crear, no solo con la documentación, sino también con el propio hardware. [32]

Licencia Ballon Open Hardware: Elaborada en 2007 para el proyecto "Balloon". En esencia se trata de una licencia GPL aplicada a la documentación del hardware, por lo que los productos de hardware derivados heredan la libertad de compartir y crear siempre y cuando se siga respetando la licencia. [32]

Licencia Hardware Design Public: Elaborada por Graham Seaman en el 2000, como parte del proyecto Opencollector.org, una comunidad de noticias sobre diseño de circuitos electrónicos, sin embargo, inactiva al día de hoy. También se trata de una licencia basada en la GPL. [32]

Licencia Solderpad: Versión de la Licencia Apache. La versión 2.0 fue enmendada por el abogado Andrew Katz para hacerla más apropiada para el uso de hardware. La licencia no es copyleft, sino permisiva, de modo que los trabajos derivados podrían ser no libres. [32]

Licencia CERN de Hardware Libre (OHL): Publicada en 2011 para su uso en el Open Hardware Repository, un repositorio con decenas de diseños electrónicos aportados por una comunidad de científicos y aficionados. [32]

2.2.5. Servidores

A un servidor se lo conoce como una aplicación, en el ámbito de software, capaz de atender peticiones de un cliente y devolver una respuesta de acuerdo a dicha petición.

Los servidores pueden ejecutarse en cualquier computador y en cualquier sistema operativo, tanto como Windows, Linux, o Mac, un servidor se instala en computadoras dedicadas para este propósito, a estas computadoras se les llama dedicadas, y se utilizan por la amplia facilidad que poseen frente a un computador normal de uso doméstico, una ventaja es su seguridad. Estos servidores trabajan bajo una estructura llamada cliente-servidor, con la ayuda de peticiones. Se conoce por cliente al computador que requiere información del servidor, o envía peticiones al servidor. Los servidores normalmente proveen servicios esenciales dentro de una red de computadores. Además, todo software que proporcione servicios puede ser considerado un servidor.

Los requerimientos de hardware para los servidores varían en dependencia de la aplicación que ofrezcan los servicios. El deber de los servidores de proveer servicios dentro de una red a un gran número de usuarios impone diferentes requerimientos, tales como conexiones de alta velocidad y el almacenamiento. Como generalmente se accede a los servidores a través de la red, estos pueden funcionar sin necesidad de un monitor u otros dispositivos de entrada. Aquellos procesos que no son necesarios para las funciones del servidor no se utilizan. Muchos servidores no cuentan con una interfaz gráfica de usuario (GUI) porque esta funcionalidad consume recursos que pueden ser utilizados por otros procesos. Igualmente, las interfaces de audio y USB también pueden ser omitidas. Los servidores funcionan por largos períodos de tiempo sin interrupción y su disponibilidad debe ser alta la mayor parte del tiempo, haciendo que la confiabilidad y durabilidad del hardware sean extremadamente importantes.

2.2.5.1. Tipos de Servidores

Existen varios tipos de servidores en todo Internet, y cada uno de estos cumple una función específica, a continuación, se describen los servidores más comunes:

Servidor WEB:

Almacena principalmente documentos HTML (son documentos a modo de archivos con un formato especial para la visualización de páginas web en los navegadores de los clientes), imágenes, videos, texto, presentaciones, y en general todo tipo de información. Además, se encarga de enviar estas informaciones a los clientes. [33]

Servidor de Correo: Es el servidor que almacena, envía, recibe y realiza todas las operaciones relacionadas con el e-mail de sus clientes.

Servidor Proxy: Es el servidor que actúa de intermediario de forma que el servidor que recibe una petición no conoce quién es el cliente que verdaderamente está detrás de esa petición.

Servidor de Bases de Datos: Da servicios de almacenamiento y gestión de bases de datos a sus clientes. Una base de datos es un sistema que nos permite almacenar grandes cantidades de información. Por ejemplo, todos los datos de los clientes de un banco y sus movimientos en las cuentas.

Servidor de Telefonía: Son servidores que ofrecen servicios de comunicación dentro de la red, puede ser VoIP o cualquier otro protocolo de comunicación.

Servidor Dedicado: Brindan servicios a única persona o empresa y no están al alcance de otros usuarios. Son servicios únicos no compartidos.

Servidor de Archivos: Es un servidor que almacena distintos archivos de audio, imagen, documentos, etc.

Servidor de Impresiones: Servidor que recibe todas las impresiones de archivos dentro de la red y gestiona cada uno de estos y las impresoras que tiene al alcance para imprimir dichos documentos.

2.2.5.2. Plataformas

Un servidor es un software, por lo tanto, puede ser instalado en el dispositivo que se desee y dependiendo de los servicios a ofrecer su hardware incrementará o disminuirá. Actualmente existen tres sistemas operativos que tienen el mayor dominio en todo el mundo Linux, Windows y MAC. Si se requiere instalar un servidor como una aplicación dentro de nuestro sistema operativo se puede optar por estas 4 opciones: WAMP, LAMP, MAMP y XAMP.

Cada una de estas aplicaciones lleva por defecto el servidor web APACHE, el gestor de base de datos MySQL y PHP, lo único en que difieren es que WAMP es para sistemas Windows, LAMP es para sistemas Linux, MAMP es para sistemas MAC y XAMP es para todos los sistemas anteriores.

Todas estas aplicaciones instalan los servicios de APACHE, MySQL y PHP en nuestro ordenador y convierten una laptop, computador de escritorio o dispositivo en un servidor web.

Si deseamos instalar un servidor completamente en nuestro ordenador o en una computadora dedicada para este propósito, lo mejor es instalar un sistema operativo base que tenga fines de servidor, debido a que si instalamos como aplicación sobre un sistema WINDOWS o MAC, por ejemplo, va a consumir grandes recursos en diversos procesos que a un servidor no le interesaría. Para estos casos la mejor opción es instalar un Sistema Operativo Linux, debido a que Linux tiene más versiones de su sistema operativo en el mundo se puede escoger un sistema que consuma pocos recursos y optimizarlo para el propósito del servidor. Dentro de Linux se encuentra varias distribuciones o sistemas operativos dedicados para servidores.

Debian

La más antigua de las distribuciones, Debian fue iniciada en 1993 por Ian Murdock. La distribución está controlada totalmente por voluntarios vinculados por el contrato social Debian. Un líder es elegido cada año desde y por los miembros del proyecto Debian. [34]

Ubuntu

Ubuntu es una distribución basada en Debian, y gestionado por Canonical. Aunque gran parte del énfasis de la distribución está en el escritorio, también hay una versión de servidor. [34]

Red Hat Enterprise

Red Hat Enterprise Linux (RHEL) es probablemente la distribución de Linux más conocida y muy popular en cuanto a servidores, además de ser uno de los más veteranos. Ha contribuido a un gran número de aplicaciones para la comunidad Open Source de los años, incluyendo Red Hat GFS y su sistema de archivos en clúster. [34]

CentOS

Centos es el más famoso de los sistemas operativos de Linux dedicados a servidores es una distribución que se derivó de Red Hat, es de fácil configuración y cuenta con muchas herramientas que vienen preinstaladas para gestionar servidores.

2.2.5.3. Programación Web

Dentro de la programación web encontramos distintos de lenguajes de programación que trabajan entre sí para dar vida a las páginas web, estos programas se guardan y ejecutan en el servidor principalmente y unos pocos del lado del cliente. La unión de estos recursos crea las páginas web que observamos en internet, cada lenguaje de programación tiene su propósito y característica, a continuación, se cita los más importantes:

HTML

HTML es el lenguaje de marcado estándar utilizado para crear páginas web y sus elementos forman los bloques de construcción de todos los sitios web. Este lenguaje se utiliza para definir el maquetado de la página web, determina todos los elementos de la misma y su posición. Los navegadores web reciben este código y lo traducen, de esta manera se puede observar las páginas web en cualquier navegador de internet [35]

PHP

PHP (acrónimo recursivo de PHP: Hypertext Preprocessor) es un lenguaje de código abierto muy popular especialmente adecuado para el desarrollo web y que puede ser incrustado en HTML. Todo el código de programación PHP se ejecuta del lado del servidor, es decir si alguien accede a un código PHP como cliente hacia un servidor, dicho cliente no podrá conocer que proceso realizó el programa PHP debido a que no se ejecuta en su computador, sino en el servidor. A diferencia de HTML con PHP se realizan todos los procesos de Scripts del lado del servidor; operaciones matemáticas, bucles, condicionales, toda la programación tradicional se puede realizar en PHP va a ser interpretado por el servidor y enviado a través de HTML al cliente que lo solicita. [36]

CSS

Hojas de Estilo en Cascada (Cascading Style Sheets), es un mecanismo simple que describe cómo se va a mostrar un documento en la pantalla, o cómo se va a imprimir, o incluso cómo va a ser pronunciada la información presente en ese documento a través de un dispositivo de lectura. Esta forma de descripción de estilos ofrece a los desarrolladores el control total sobre estilo y formato de sus documentos. CSS se utiliza para dar estilo a documentos HTML y XML, separando el contenido de la

presentación. Los Estilos definen la forma de mostrar los elementos HTML y XML. CSS permite a los desarrolladores Web controlar el estilo y el formato de múltiples páginas Web al mismo tiempo. Cualquier cambio en el estilo marcado para un elemento en la CSS afectará a todas las páginas vinculadas a esa CSS en las que aparezca ese elemento. Se puede decir que todos los colores, fondos, tipos de letras tamaños, etc. todo el diseño visual de una página web esta en estos documentos. [37]

JavaScript

Este es un lenguaje interpretado, no requiere compilación. Fue creado por Brendan Eich en la empresa Netscape Communications. Utilizado principalmente en páginas web. Es similar a Java, aunque no es un lenguaje orientado a objetos, el mismo no dispone de herencias. La mayoría de los navegadores en sus últimas versiones interpretan código Javascript. El código Javascript puede ser integrado dentro de nuestras páginas web. Para evitar incompatibilidades el World Wide Web Consortium (W3C) diseño un estándar denominado DOM (en inglés Document Object Model, en su traducción al español Modelo de Objetos del Documento). [38]

ASP.NET

Este es un lenguaje comercializado por Microsoft, y usado por programadores para desarrollar entre otras funciones, sitios web. ASP.NET es el sucesor de la tecnología ASP, fue lanzada al mercado mediante una estrategia de mercado denominada .NET. El ASP.NET fue desarrollado para resolver las limitantes que brindaba tu antecesor ASP. Creado para desarrollar web sencillas o grandes aplicaciones. Para el desarrollo de ASP.NET se puede utilizar C#, VB.NET o J#. Los archivos cuentan con la extensión (aspx). Para su funcionamiento de las páginas se necesita tener instalado IIS con el Framework .Net. Microsoft Windows 2003 incluye este framework, solo se necesitará instalarlo en versiones anteriores. [38]

PYTHON

Es un lenguaje de programación creado en el año 1990 por Guido van Rossum, es el sucesor del lenguaje de programación ABC. Python es comparado habitualmente con Perl. Los usuarios lo consideran como un lenguaje más limpio para programar. Permite la creación de todo tipo de programas incluyendo los sitios web. Su código no necesita ser compilado, por lo que se llama que el código es interpretado. Es un lenguaje de

programación multiparadigma, lo cual fuerza a que los programadores adopten por un estilo de programación particular. [38]

2.2.6. Sistema Eléctrico de un Automóvil

Desde su creación el automóvil ha evolucionado de manera considerable, pasando de tener partes solo mecánicas a tener actualmente diversos sistemas que funcionan en conjunto para brindar diversas aplicaciones y funcionamiento a nuestro automóvil. Una de estas funciones que ha ido evolucionando es el sistema eléctrico del automóvil, el sistema eléctrico del automóvil hoy en día es indispensable para el funcionamiento del automóvil, sin este, no funcionaría. Debido a la inexistencia de un estándar para los diagramas y diseños del sistema eléctrico automotriz, cada fabricante varía sus diseños, con referencia a las prestaciones que quiere ofrecer, por lo tanto, hoy en día en ningún automóvil hay un sistema eléctrico similar. Para englobar como funciona un sistema eléctrico se toma como referencia la Figura 2.12 se muestra un diagrama general de las conexiones que posee un automóvil.

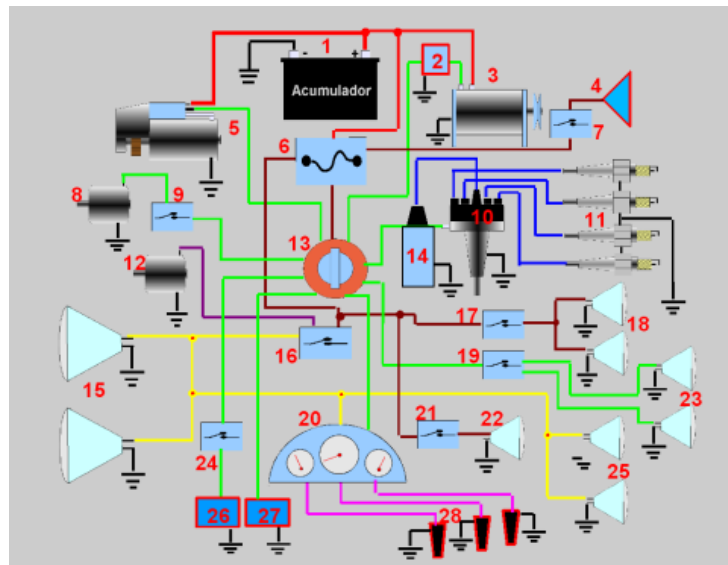


Figura 2.12. Diagrama General de un sistema Eléctrico de Automóvil

Fuente: [39]

Las partes numeradas en el diagrama del circuito son:

1.- Acumulador, 2.- Regulador de voltaje, 3.- Generador, 4.- Bocina o claxon, 5.- Motor de arranque, 6.- Caja de fusibles, 7.- Interruptor del claxon, 8.- Prestaciones de potencia que funcionan con el interruptor de encendido conectado y con interruptor propio; ejemplo: vidrios de ventanas, limpiaparabrisas etc. 9.- Representa los

interruptores de las prestaciones 8, 10.- Distribuidor, 11.- Bujías, 12.- Representa las prestaciones de potencia que funcionan sin el interruptor de encendido; ejemplo: seguros de las puertas, cierre del baúl de equipaje etc. 13.- Interruptor de encendido; 14.- Bobina de encendido, 15.- Faros de luz de carretera delanteros, 16.- Interruptor de faros de luz de carretera 17.- Interruptor de faros de luz de frenos, 18.- Luces indicadoras de frenado, 19.- Interruptor-permutador de faros de vía (intermitentes), 20.- Tablero de instrumentos 21.- Interruptor de lámpara de cabina, 22.- Lámpara de cabina, 23.- Luces de vía (intermitentes), 24.- Interruptor de prestaciones especiales, 25.- Luces de carretera traseras, 26.- Representa las prestaciones especiales que solo funcionan con el interruptor de encendido conectado; ejemplo: radio, antenas eléctricas etc. 27.- Sistema de inyección de gasolina, 28.- Sensores de instrumentos del tablero.

Distribución de cables:

Rojo: Conexiones directas al acumulador sin protección con fusibles.

Marrón: Conexiones alimentadas a través de fusibles de protección. Estos fusibles y sus circuitos correspondientes pueden ser múltiples, aunque en el esquema se representan como uno solo. Cuando la potencia eléctrica lo requiere se utilizan relés relevadores que no han sido representados.

Verde: Circuitos alimentados desde el interruptor de encendido. Estos circuitos solo tienen tensión eléctrica cuando el interruptor está conectado. Cuando la potencia eléctrica lo requiere se utilizan relés relevadores que no han sido representados.

Azul: Cables de alta tensión del sistema de encendido (en la actualidad estos cables no existen en una buena parte de los automóviles).

Violeta: Circuitos protegidos con fusible, para algunas de las prestaciones adicionales, con interruptor propio. Estos circuitos están alimentados con tensión en todo momento. Cuando la potencia eléctrica lo requiere se utilizan relés relevadores que no han sido representados.

Amarillo: Circuito de iluminación de carretera y tablero de instrumentos. Está protegido con fusibles y alimentado con tensión permanentemente. Tiene su propio interruptor. En algunos casos la permutación de las luces principales de carretera se hace con el uso de relés relevadores, que no han sido representados.

Magenta: Cables a los sensores de los instrumentos del tablero.

Negro: Conexiones de tierra.

En la Figura 2.12 se ve de forma general como está repartido el sistema eléctrico de un automóvil, pero en cada auto será diferente debido a las aplicaciones que el fabricante las de. El sistema eléctrico del automóvil nos da control sobre ciertas áreas del mismo, por ejemplo, al encender el vehículo se utiliza un sistema de arranque que obligatoriamente tiene que ocupar energía eléctrica; para que el automóvil funcione correctamente en la carretera se necesita enviar corriente a las bujías para que creen la chispa en los cigüeñales y con la gasolina el motor este operativo además el sistema eléctrico controla todos los dispositivos eléctricos del automóvil, como los luces, sensores del automóvil, accesorios de entretenimiento, aire acondicionado, es decir todo lo eléctrico que se ocupe en el automóvil estará regido por el sistema eléctrico; por lo cual el sistema eléctrico del automóvil es de vital importancia para el funcionamiento completo del auto.

2.2.6.1. Computador del Automóvil

Todos los automóviles actuales poseen una computadora dentro de ellos, esta computadora controla todo el sistema eléctrico del automóvil, desde la cantidad de combustible y aire que manda al motor, hasta el encendido y apagado de las luces de parque, un automóvil sin su computadora no funciona. Actualmente dicha computadora controla prácticamente todas las partes del automóvil y dependiendo del auto varia en procesos que tiene que realizar, ningún computador es igual en el mercado.

2.2.6.2. Sistema de Arranque

El motor de combustión interna no tiene arranque propio, hay que hacerlo girar con una fuente externa para que se completen los procesos necesarios y se produzca el encendido. Existen varias formas de hacer girar el motor para que arranque:

- Arranque manual.

El arranque manual se usa para los pequeños motores donde con un aceptable esfuerzo corporal se hace girar el motor para el arranque

- Arranque con la energía de aire comprimido.

El arranque por aire comprimido se usa para algunos grandes motores en los que la potencia necesaria hace difícil el uso del arranque eléctrico debido a las altísimas corrientes necesarias, y en algunos vehículos especiales adaptados para funcionar a muy bajas temperaturas donde las baterías de acumuladores no pueden utilizarse.

- Arranque por motor de combustión interna auxiliar.

El arranque por motor de combustión auxiliar se usa en algunas máquinas de la construcción que usan motores Diésel. Estas máquinas pueden prescindir de las baterías de acumuladores y así ser más adaptables a condiciones climáticas de fríos severos. Usan un pequeño motor de gasolina que se arranca por el método manual o con motor eléctrico, este a su vez acciona el motor principal a través de un acoplamiento de engranajes desplazables.

- Arranque por motor eléctrico.

El arranque por motor eléctrico es el más utilizado en la actualidad y consiste en un motor eléctrico que acciona al motor del automóvil, o en otras palabras le da un pequeño impulso al motor mecánico del automóvil para que empiece a trabajar.

En la Figura 2.13 se observa el diagrama de arranque de un automóvil.

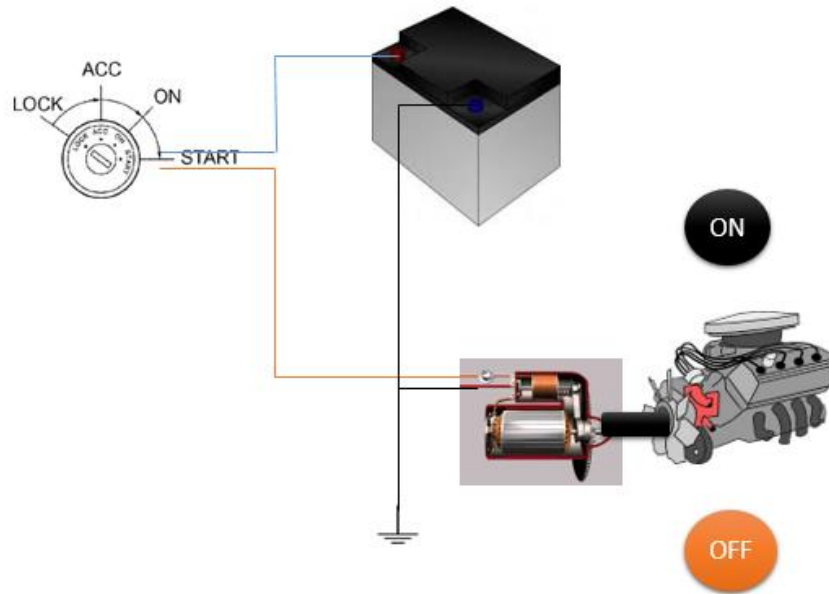


Figura 2.13. Diagrama de arranque de un automóvil.

Fuente: Autor

Al momento de girar la llave en la última posición, la posición de ignición, el relé del motor eléctrico detecta esta señal y automáticamente deja pasar la corriente de la batería hacia el motor eléctrico, el cual va a comenzar a girar debido al voltaje que se le aplica. Cuando está girando, un pistón empuja el eje de giro del motor eléctrico y lo acopla momentáneamente al motor del automóvil para que la fuerza mecánica del motor eléctrico pase al motor del automóvil, una vez encendido, el pistón se contrae y se deja de girar la llave y esta regresa a la posición ON con el auto ya encendido y el motor eléctrico apagado. De esta manera comienza a funcionar el automóvil cuando se enciende con la llave, el tiempo que dura todo este proceso es de segundos, porque no se debe dejar conectado el motor eléctrico al motor del automóvil cuando este último ya esté en funcionamiento.

2.2.6.3. Sistema de Luces

Este sistema es el encargado de toda la iluminación del vehículo, y puede o no estar conectado a la computadora. El sistema de luces de los automóviles ha sufrido grandes cambios, normalmente se han utilizado bombillas con filamento para encender las luces de visibilidad del automóvil, pero actualmente se están reemplazando por las luces led.

2.2.6.4. Sistema de Accesorios

Todos los elementos electrónicos del automóvil que no sean de control, como por ejemplo radio, luces led, calefacción, cierre de puertas, alarmas, etc. conforman este sistema. En este sistema el usuario tiene mayor control para la instalación de diversos accesorios electrónicos al automóvil, Normalmente este sistema funciona con el automóvil apagado o con la llave en la posición ACC u ON. Cuando el automóvil está apagado este sistema consume energía de la batería del auto, cuando esta encendido, consume energía del alternador del automóvil. Cuanto mayor es el número de dispositivos que se tiene en el automóvil mayor será la carga y mayor el desgaste de la batería.

2.3. Propuesta de Solución

Por medio del desarrollo de un sistema de monitoreo web basado hardware y software libre se realiza el control remoto de cierre y apertura de puertas, apagado remoto de motor y alertas de alarma mediante mensajes de texto, además de un rastreo GPS y la implementación de un servidor web para almacenar las posiciones del vehículo con el fin de brindarle seguridad a un costo accesible.

CAPÍTULO III METODOLOGÍA

3.1. Modalidad de la Investigación

En el desarrollo del proyecto se utilizó una investigación aplicada la cual esta complementada con los siguientes tipos:

Investigación Bibliográfica.

Se empleó esta modalidad considerando fuentes como; libros, revistas, publicaciones científicas, proyectos de investigación; que se usó para adquirir los conocimientos necesarios que permitan desarrollar de mejor manera la investigación.

Investigación de Campo.

Se utilizó este tipo investigación mediante la adquisición de datos del sistema, medición y calibración de dispositivos, así mismo como pruebas de su correcto funcionamiento en el automóvil de manera práctica.

3.2. Población y Muestra

En este proyecto de investigación no es necesaria la utilización de población y muestra debido a que se realizó un diseño de un prototipo. Además, la información necesaria se encuentra disponible en las fuentes bibliográficas utilizadas para el desarrollo del proyecto de investigación.

3.3. Recolección de Información

La recolección de información se realizó en tesis de graduación, artículos de investigación, entre otros documentos de investigación; debido a que no se necesitó de una encuesta o censo para el desarrollo del proyecto, los cuales permitieron conocer el estado actual de los diversos conceptos teóricos del proyecto de investigación.

3.4. Procesamiento de Información

Para el procesamiento de información obtenida se tomó en cuenta las siguientes actividades:

- Análisis de la información recolectada para el planteamiento del problema y obtención de objetivos del proyecto
- Resumen de la información fundamental obtenida de artículos científicos, revistas, proyectos semejantes, tesis, libros e internet.

3.5. Desarrollo del Proyecto

El proyecto se desarrolló de manera secuencial de acuerdo con las siguientes actividades:

- Análisis de diversos receptores GPS en el mercado.
- Análisis y tratamiento de la información que proporciona el receptor GPS.
- Revisión de los parámetros necesarios de módulos GSM/GPRS para el envío de datos.
- Investigación de la transmisión de datos hacia internet por medio de telefonía móvil.
- Determinación del hardware necesario para el diseño.
- Determinación del software necesario para el diseño.
- Análisis de infraestructura eléctrica del automóvil.
- Implementación del sistema de monitoreo en un automóvil.
- Implementación de servidor web.
- Procesamiento de datos recibidos remotamente a través de internet.
- Diseño de una interfaz web para localización del automóvil.
- Realizar pruebas de funcionamiento del sistema de monitoreo.

CAPÍTULO IV

DESARROLLO DE LA PROPUESTA

4.1. Diagramas de Bloques del Prototipo

El sistema diseñado en este proyecto consta de rastreo GPS, el cual da la ubicación del automóvil por medio de coordenadas geográficas de latitud y longitud obtenidas de las constelaciones de satélites de navegación. Bloqueo y desbloqueo de motor; que consiste en impedir el encendido del motor del automóvil y detenerlo si este se encuentra en movimiento. Notificaciones de la alarma, cuando ha sido activada, y apertura y cierre de puertas remotamente, todo esto controlado por mensajes de texto desde un teléfono celular. Adicional se implementa un servidor web, ubicado en el hogar del propietario, que almacene los datos obtenidos del rastreo GPS y los visualice en una página web que esté disponible en internet. Se otorga el 100% del control del sistema de monitoreo web del automóvil al usuario, para que este pueda mejorarlo según sus necesidades y no dependa de terceras empresas para operar su sistema. Este sistema puede ser utilizado por usuarios particulares, desarrolladores o ingenieros electrónicos.

El sistema de monitoreo está dividido en dos partes. El sistema en el automóvil, que controla todas las acciones sobre el mismo, y el servidor ubicado en el hogar del propietario con acceso a internet, que visualiza la posición del automóvil. Para este proyecto es indispensable el acceso a internet con una IP pública, además del servicio de telefonía móvil para enviar mensajes de texto y conexión a la red GPRS para el envío de datos hacia internet.

A continuación, en la Figura 4.1 se representan las dos etapas que conforman el Proyecto

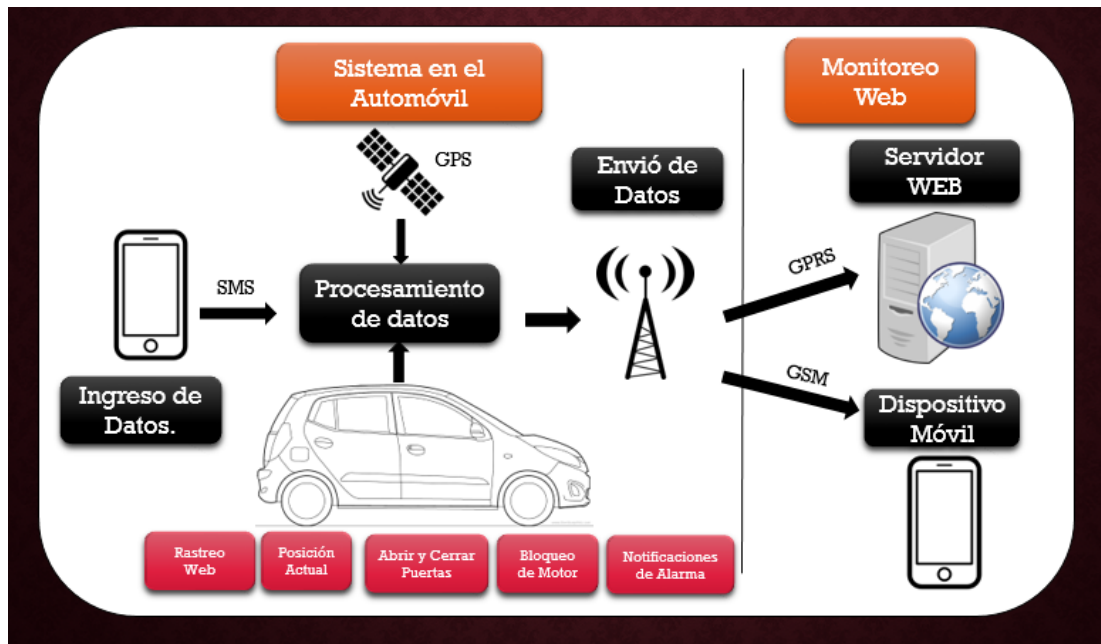


Figura 4.1. Diagrama de Bloques del Sistema Completo.

Fuente: Autor.

4.1.1. Sistema en el Automóvil

Esta etapa es el conjunto de componentes electrónicos, ubicados en el automóvil, que se encargan de controlar el rastreo, la apertura y cierre de puertas, el bloqueo y desbloqueo de la llave de ignición para encender el motor y el envío de notificaciones de la alarma mediante mensajes de texto. Este sistema consta de tres partes fundamentales para su funcionamiento. A continuación, se describe el funcionamiento de cada una de estas:

Ingreso de Datos

En el Ingreso de Datos el sistema en el automóvil recibe comandos mediante mensajes de texto enviados desde un teléfono celular, cada comando ejecuta una acción en el sistema. Los comandos pueden ser cambiados por el usuario en la programación y pueden ser numéricos o alfanuméricos, la seguridad de estos radica en la complejidad que se asigne a cada código; en este proyecto se utilizó los siguientes comandos: “posición”, “rastreo”, “abrir”, “cerrar”, “bloquear” y “desbloquear” para representar de manera sencilla cada acción. Las acciones que puede realizar el sistema con estos comandos se describen a continuación:

- **Rastreo**

Cuando el sistema del automóvil reconoce el comando respectivo a “rastreo”, el sistema se encarga de receptor muestras de latitud, longitud y velocidad del automóvil por medio de un receptor GPS y envía estas muestras a través de la red GPRS hacia un servidor web, el cual va a estar alojado en el hogar del usuario. Esta función envía un número determinado de muestras al servidor, dependiendo de la programación se puede variar el número de muestras, así como el dominio web que tiene el servidor. Posteriormente el servidor almacena en su base de datos las muestras recibidas por parte del automóvil y las visualiza en una página web. Esta función solo envía datos al servidor y no al dispositivo celular del propietario.

- **Posición**

Cuando el sistema del automóvil reconoce el comando respectivo a “posición”, el sistema del automóvil se encarga de receptor una muestra de latitud, longitud y velocidad, por medio de un receptor GPS y envía un mensaje texto con un link en Google Maps con los datos de latitud y longitud recogidos, así como la velocidad del automóvil; por medio de la red GSM hacia un teléfono celular. El mensaje que llega al teléfono celular contiene un enlace a Google Maps, si el usuario dispone de un teléfono inteligente, el SMS se abrirá con esta aplicación indicando el lugar donde se encuentra el automóvil. Esta acción solo envía una muestra hacia el teléfono celular del propietario, no utiliza el servidor web.

- **Apertura y Cierre de Puertas**

Cuando el sistema reconoce los comandos “abrir” o “cerrar” envía por radiofrecuencia, los códigos asignados a la alarma del automóvil para abrir y cerrar las puertas. Estos códigos fueron obtenidos a partir del mando que posee la alarma, de esta manera se duplica estos códigos y son reenviados por un módulo de radiofrecuencia que se encuentra en el sistema en el automóvil.

- **Bloqueo de Motor y Desbloqueo de Motor**

Cuando el sistema reconoce el código “bloquear” activa un relé que cambia de estado normalmente cerrado a normalmente abierto y corta la conexión de la llave de ignición en la posición ON, de esta manera si el automóvil esta encendido se apaga

completamente y se impide encenderlo nuevamente; al enviar el código “desbloquear” se desactiva el relé y vuelve a estado normalmente cerrado y deja pasar corriente a la llave de ignición en la posición ON y permite encenderlo nuevamente.

Procesamiento de datos

En esta etapa se realizan todas las operaciones lógicas del sistema, es el encargado de identificar que acción realizar de acuerdo a los mensajes de texto recibidos y ejecutarla. El procesamiento de datos es el núcleo de todo el sistema, está conformado por los siguientes componentes:

- **Módulo Procesador de Datos**

Es el encargado de realizar las operaciones lógicas y el procesamiento de los datos obtenidos de todo el sistema.

- **Módulo SIM**

Se encarga de enviar y recibir de mensajes de texto y tener comunicación a internet.

- **Módulo GPS**

Se encarga de receptar las coordenadas geográficas de latitud y longitud del automóvil.

- **Módulo de Radio Frecuencia**

Se utiliza para interactuar con la alarma del Automóvil para replicar los códigos de apertura y cierre de puertas.

- **Bloqueo de Motor**

Se encarga de bloquear la conexión del cable ON en la llave de giro del volante del automóvil, para de esta manera impedir el paso de corriente para encender el vehículo.

- **Adquisición de Señal de Sirena**

Se encarga de adquirir la señal de la sirena de la alarma de 12V cuando está ha sido activada y acoplarla al sistema para poder trabajar con ella.

- Alimentación del Sistema

Se encarga de regular las tensiones del automóvil y acoplarlas a una tensión de 5V para trabajar con los equipos electrónicos que se procede a seleccionar.

- Respaldo de Baterías

Se utiliza como seguridad para dar al sistema una fuente de alimentación alternativa en caso de que la batería del automóvil dejase de funcionar, o fueran cortados sus cables de alimentación en caso de robo.

- Visualización

Se utiliza para visualizar los procesos que realiza el sistema y demostrar su funcionamiento.

Envío de Datos

El envío de datos utiliza la conexión a la red GSM y GPRS para enviar los datos obtenidos del sistema hacia el exterior. Cuando se ejecuta el comando “posición” o cuando se activa la alarma del automóvil, el sistema utiliza únicamente la red GSM para enviar mensajes de texto hacia un teléfono celular, con los datos de latitud y longitud; o notificaciones de la alarma activada. Cuando se ejecuta el comando “rastreo” el sistema utiliza la red GPRS para establecer una comunicación con el servidor web en internet y envía los datos de latitud, longitud y velocidad del automóvil, los cuales son almacenados en el servidor y visualizados en una página web.

4.1.2. Sistema de Monitoreo

El sistema de monitoreo consta de dos interfaces, el servidor web y el teléfono del propietario. Estos dos dispositivos se encargan de recibir la información que envía el sistema de rastreo.

- **Servidor Web**

Se encarga de almacenar todos los datos que se envían desde el sistema en el automóvil, para procesarlos y exponerlos en una página WEB que esté disponible en internet. El servidor web no se encuentra junto al sistema en el automóvil, se encuentra ubicado en un lugar externo, este puede ser el hogar del propietario del sistema.

Además, el servidor debe poseer Internet ilimitado y una IP fija para colocar el servidor en internet.

- **Dispositivo Móvil**

Es el encargado de recibir las notificaciones del sistema, ya sea las notificaciones de la alarma del automóvil o mensajes de posición del automóvil. La información que se despliegue hacia nuestro terminal puede ser configurada en el sistema de rastreo dependiendo de qué datos se desee conocer. Se puede modificar los mensajes de alertas y también los datos obtenidos por el módulo GPS, en la programación.

4.2. Análisis de Requerimientos

Para analizar los requerimientos técnicos del proyecto, se realizó el estudio en dos partes, los requerimientos para el sistema en el automóvil y los requerimientos para el servidor web, debido a que el primero se encuentra en el automóvil y el segundo en el hogar del propietario y las condiciones para analizarlos no serán las mismas. A continuación, se describen los requerimientos de cada uno de estos:

4.2.1. Requerimientos del Sistema en el Automóvil

El automóvil utilizado en el desarrollo de este proyecto es un Hyundai Modelo i10 del año 2013 que cuenta con un sistema de alarma y cierre centralizado de puertas.

Alimentación del Sistema.

Todos los componentes electrónicos dentro del automóvil funcionan con la tensión otorgada por la batería, la cual es de marca EXIDE de 12V y capacidad de 40Ah. El rango de voltaje suministrado por la batería varía de 10V a 13.5V.

Los componentes electrónicos del proyecto requieren un voltaje de 5V para su operación, debido a esto se requiere reguladores de voltaje que reduzcan o eleven el voltaje de trabajo a 5V. Además, para adquirir la señal proveniente de la sirena es necesario realizar una etapa de acoplamiento de 12V a 5V por medio de un optoacoplador, para que pueda ser interpretada como 1 o 0 lógico.

Conexión a las redes móviles

El sistema requiere de conexión con las redes móviles GSM y GPRS para enviar los datos hacia el servidor o el teléfono celular del propietario. Las empresas que ofrecen

servicios de telefonía móvil en la Republica de Ecuador son, Claro (CONECEL), Movistar(OTECEL) y CNT E.P. La ARCOTEL [40] es el organismo que regula el espectro radioeléctrico en el país y otorga la concesión de bandas de frecuencias para operadores de telefonía móvil. Las bandas de Frecuencias de estas operadoras se visualizan en la Tabla 4.1

Tabla 4.1 Bandas de Frecuencias de Operadoras Móviles en Ecuador

Operadoras Móviles	Banda de Frecuencia
Claro (CONECEL)	850MHz y 1900MHz
Movistar (OTECEL)	850MHz y 1900MHz
CNT E.P.	1900MHz

Fuente: Agencia de Regulación y Control de Telecomunicaciones. [40]

Los módulos que se utilizan en el proyecto deben operar a las frecuencias de cualquiera de estas operadoras móviles.

Servicio de Posicionamiento

Para obtener coordenadas geográficas de latitud y longitud que establezcan la ubicación del automóvil, se necesita un receptor de las señales que emiten los sistemas de posicionamiento global.

Las constelaciones satelitales de posicionamiento global existentes en el mundo son GPS de Estados Unidos, GLONASS de Rusia, GALILEO de la Unión Europea y BEIDOU de China, las características de estas constelaciones se describen en la Tabla 4.2

Tabla 4.2 Sistemas de Posicionamiento Global

Características	GPS	GLONASS	GALILEO	BEIDOU
Números de Satélites	24	24	27	30
Planos Orbitales	6	3	3	3
Inclinación Orbital (°)	55	64.8	56	55
Período de Satélite (h)	12h	11h y 15 min	14	12
Altura Sobre la Tierra	22200	19100	23222	21528
País	EEUU	Rusia	Unión Europea	China

Fuente: Autor.

La operación de los sistemas de navegación GALIELO y BEIDOU aún no están habilitados para el país, por lo tanto, los receptores de estos sistemas no son viables. Los sistemas GPS y GLONASS tienen una cobertura mundial, sin embargo, en el mercado civil no existen receptores que trabajen solo con el sistema de posicionamiento global GLONASS. En este proyecto se ha utilizado un receptor de GPS debido a que el sistema satelital del mismo brinda cobertura en el país y posee una buena precisión.

Temperatura

La temperatura incide en el funcionamiento de los equipos electrónicos, de esta manera se procedió a analizar las temperaturas a las cuales el sistema va estar expuesto.

En primer lugar, se procede a analizar las temperaturas ambientales en las posibles regiones del país en las cuales el automóvil podría movilizarse. En el Boletín Climatológico Semestral [41] de Enero a Junio del año 2016 del Instituto Nacional de Meteorología e Hidrología INAMHI se registran las temperaturas máximas y mínimas del ambiente alcanzadas en las tres regiones que posee el país, región insular, región andina y región amazónica; cada una de estas regiones posee ecosistemas diferentes dando así temperaturas variadas para cada una de estas. en la Figura 4.2 se describen los valores máximos de temperatura alcanzados por las diversas estaciones meteorológicas y en la Figura 4.3 se describen los valores mínimos de temperatura.

Figura 4.2 Temperatura media de Máxima Absoluta.

LOCALIDADES	Normal Temperatura Media Máxima Absoluta Semestral	Temperatura Media Máxima Absoluta Semestral 2016	Anomalia
REGION LITORAL/INSULAR			
ESMERALDAS AER.	31,8	31,1	-0,7
LA CONCORDIA	32,3	32,2	-0,1
STO. DOMINGO AER.	30,2	30,5	0,3
PUERTO ILA	32,1	32,8	0,7
PORTOVIEJO	34,4	35,2	0,7
PICHILINGUE	33,1	33,6	0,5
MILAGRO	33,1	34,1	0,9
GUAYAQUIL AER.	34,1	34,1	0,0
SANTA ROSA AER.	32,4	33,0	0,6
S. CRISTOBAL-GAL.	31,3	32,6	1,3
REGION INTERANDINA			
TULCAN AER.	21,2	21,0	-0,2
SAN GABRIEL	20,8	21,6	0,8
INGUINCHO	17,8	18,4	0,5
IBARRA	26,0	27,2	1,2
OTAVALO	23,9	25,1	1,2
TOMALON	23,9	26,0	2,1
QUITO-ÑAQUITO	24,0	25,6	1,6
LA TOLA	25,5	25,7	0,2
IZOBAMBA	20,7	21,8	1,1
LATACUNGA AER.	22,3	23,3	1,0
RUMIPAMBA	24,2	24,4	0,2
QUEROCHACA	22,3	22,1	-0,2
RIOBAMBA AER.	24,8	24,9	0,2
CUENCA AER.	24,1	26,5	2,4
LA ARGELIA LOJA	24,9	26,2	1,3
LA TOMA CATAMAYO	33,2	34,0	0,8
CELICA	22,3	24,0	1,7
REGION ORIENTAL			
LAGO AGRIO AER.	33,7	34,0	0,3
EL COCA AER.	34,3	33,9	-0,4
NVO. ROCAFUERTE	34,0	35,3	1,3
PASTAZA AER.	28,2	30,2	2,0
PUYO	29,4	29,8	0,4
MACAS AER.	30,0	30,0	0,0

Fuente: Instituto Nacional de Meteorología e Hidrología [41].

En este mismo boletín se encuentran temperaturas record que alcanzaron ciertas estaciones meteorológicas, registrando la temperatura más alta en el país de 36.5 °C en la estación meteorológica de Portoviejo.

Con esta información y analizando la Figura 4.2 se estima que la temperatura ambiental máxima a la cual el automóvil puede ser expuesta es de 36,5°C.

Figura 4.3 Temperatura media de Mínima Absoluta.

LOCALIDADES	Normal Temperatura Media Mínima Absoluta Semestral	Temperatura Media Mínima Absoluta Semestral 2016	Anomalia
REGION LITORAL/INSULAR			
ESMERALDAS AER.	21,2	22,6	1,4
LA CONCORDIA	19,8	21,5	1,7
STO. DOMINGO AER.	19,0	20,3	1,2
PUERTO ILA	20,0	21,9	1,9
PORTOVIEJO	20,2	21,7	1,5
PICHILINGUE	20,6	22,0	1,5
MILAGRO	20,8	22,2	1,3
GUAYAGUIL AER.	21,4	22,8	1,4
SANTA ROSA AER.	21,5	22,2	0,7
ZARUMA	15,8	17,0	1,1
S. CRISTOBAL-GAL.	20,8	20,6	-0,2
REGION INTERANDINA			
TULCAN AER.	2,9	5,0	2,1
SAN GABRIEL	3,2	4,9	1,7
INGUINCHO	4,2	6,3	2,1
IBARRA	7,6	9,2	1,6
OTAVALO	4,7	6,7	2,0
TOMALON	6,8	8,0	1,2
QUITO-IRAGUITO	7,4	9,3	1,9
LA TOLA	6,6	7,7	1,0
IZOBAMBA	3,6	3,7	0,1
LATACUNGA AER.	5,6	4,6	-1,0
RUMIPAMBA	4,8	5,5	0,7
QUEROCHACA	3,7	6,4	2,7
RIOBAMBA AER.	5,1	6,9	1,8
CUENCA AER.	7,6	9,0	1,3
GUALACEO	7,7	9,3	1,6
LA ARGELIA LOJA	8,8	10,2	1,5
LA TOMA CATAMAYO	15,2	16,2	0,9
CELICA	9,9	12,2	2,3
REGION ORIENTAL			
LAGO AGRIO AER.	19,8	20,9	1,2
EL COCA AER.	20,3	20,6	0,3
NVO. ROCAFUERTE	20,3	21,0	0,7
PASTAZA AER.	16,1	17,4	1,3
PUYO	14,6	16,5	1,9
MACAS AER.	15,5	16,2	0,7

Fuente: Fuente: Instituto Nacional de Meteorología e Hidrología [41].

En temperaturas mínimas no se registran records en relación al año 2015, según este informe. Con relación a los datos de la Figura 4.3 se estima que la temperatura mínima de ambiente a la cual el automóvil podría ser expuesto es de aproximadamente 4,6°C.

De esta manera se obtiene que el rango de temperaturas ambiente, a las cuales el automóvil podría estar expuesto es de 4,6°C a 36,5°C en cualquier región del país.

Dentro del vehículo, debido a las condiciones de sol, el sellamiento del automóvil, el color del mismo y otros factores; provocan que las temperaturas dentro del automóvil se eleven. El Real Automóvil Club de Cataluña (RACC) junto con el Automóvil Club Suizo (TCS) [42] ha realizado un estudio para demostrar los riesgos que implican las altas temperaturas que se pueden llegar a registrar en el interior de un vehículo expuesto a la radiación solar.

En su estudio indican que, con una temperatura ambiente de 35°C, dentro del automóvil se obtienen temperaturas aproximadas de 55°C, además el color del auto provoca un aumento de temperatura, situando los automóviles de color negro con temperaturas más elevadas que automóviles de otros colores en un 17%. Además, las temperaturas donde el sol tiene más incidencia, como son el volante, el cambio de marchas y el tablero de control del automóvil registran temperaturas de 72.5°C, en su superficie.

Tomando en cuenta este estudio, se determinó la relación del cambio de temperatura que existe dentro del automóvil, la cual incrementa en un valor de 20°C en lugares no expuestos al sol y en un valor de 37.5°C con exposición directa al sol. En la Tabla 4.3 se observan la temperatura ambiente, la temperatura dentro del automóvil con incidencia directa al sol y la temperatura dentro del automóvil en el lugar donde se instaló el sistema.

Tabla 4.3 Análisis de Temperaturas en el país.

Temp.	Lugar Ambiente	Automóvil con exposición directa al sol	Lugar de instalación del sistema
Temp. Max	36,5 °C	74°C	56,5°C

Fuente: Autor

La ubicación del sistema en el automóvil es muy importante, se debe ubicarlo en un lugar donde no este expuesto directamente al sol, para que los componentes electrónicos internos del sistema no alcancen temperaturas demasiado altas. De acuerdo con el análisis realizado de temperaturas los componentes electrónicos para el sistema deben operar a temperaturas de 4,6°C a 56,5°C sin problemas

Campos Electromagnéticos

En el automóvil se puede producir un elevado número de ruidos u ondas parásitas producidas por cualquier aparato eléctrico o mecánico, en el cual se crea un campo magnético variable. Los dispositivos más significativos que provocan estos ruidos son:

- El Sistema de encendido.
- El dinamo o el alternador y el regulador de tensión.

- El motor de arranque.
- El motor limpiaparabrisas.
- Las bocinas.
- La bomba eléctrica de gasolina.

De acuerdo al análisis de interferencia en el automóvil el sistema debe cumplir con los requerimientos que se mencionan a continuación:

- Para evadir las ondas parasitas que son inevitables, se requiere que el sistema se encuentre rodeado de un material metálico, el cual debe estar conectado a tierra del automóvil, de esta manera se simula una jaula de Faraday que protege a los componentes electrónicos dentro de la caja de las ondas electromagnéticas del exterior; con el fin de repeler estos campos electromagnéticos no deseados y otorgarle estabilidad al mismo. Se ha utilizado un recubrimiento de papel aluminio debido a que no es recomendable la utilización de cajas metálicas, porque incrementan el nivel de vibración y temperatura del sistema en comparación con las cajas de plástico diseñadas para altas temperaturas. Adicionalmente las antenas GPS, GSM y de Radio Frecuencia utilizadas deben tener una conexión al exterior para su funcionamiento.
- Debido las variaciones de voltaje generadas por el funcionamiento de los componentes eléctricos del automóvil, se requiere que la fuente de alimentación del sistema suprima estos picos de voltaje mediante el uso de condensadores.

Vibración

El automóvil en buenas condiciones por defecto experimenta cierto nivel de vibración, debido a la propia mecánica del mismo y a que se encuentra en constante movimiento, además influyen factores como pavimento irregular por la presencia de empedrados, reductores de velocidad, vías en mal estado, entre otros. Si no se toman las debidas precauciones estas vibraciones pueden causar desgaste y cierto ruido en los dispositivos electrónicos del sistema.

La vibración se mide mediante los parámetros aceleración, velocidad y desplazamiento presentes en la misma, mientras mayor es la aceleración de la vibración tendrá mayor efecto al confort del sistema, además se la determina mediante el rango de frecuencias que genera el ruido de la vibración.

Según la publicación adicional dos de la Norma ISO 2631, se define el término “mal transporte”, el cual indica que el grado de vibración en un automóvil se encuentra en el rango de frecuencias muy bajas (entre 0.1 y 0.63Hz).

Considerando que el tipo de vibración del automóvil según las características físicas del entorno vibracional, pertenece al grupo de muy bajas frecuencias, es posible controlar y reducir la vibración, evitando así las desventajas de la misma en el sistema, para ello se deben tomar en cuenta los siguientes requerimientos:

- El sistema debe fijarse correctamente en el lugar del automóvil en que es camuflado, mediante un soporte adecuado con el fin de evitar que el desequilibrio y la desalineación del mismo, puedan generar vibración.
- Los dispositivos electrónicos alojados dentro del sistema, deben encontrarse correctamente sujetos a la placa electrónica, para evitar vibración producida por piezas flojas o escasos puntos de anclaje. [43] [44]

Humedad

La humedad ambiental se la puede definir como la cantidad de vapor de agua que contiene el aire, y se la expresa en forma de humedad absoluta, específica y relativa.

Para este análisis se ha tomado en cuenta la humedad relativa, que es la relación entre la cantidad de vapor de agua real del ambiente y la cantidad de vapor de agua que se necesita contener para saturarse a idéntica temperatura, la cual se expresa en porcentajes. Cuando el aire se satura la humedad relativa debe ser igual al 100 % y se llega al punto de rocío.

La saturación de aire, acelera la corrosión de las placas electrónicas formando soluciones conductivas que, posteriormente, se convierten en electrolitos y esto a su vez, provoca cortocircuitos o inestabilidad del sistema con señales falsas o interrupción de procesos. [45]

Se ha realizado un análisis de humedad, debido a que las regiones Costa y Oriente del país, son regiones cálidas húmedas, con el fin de determinar los niveles de humedad relativa existentes.

En la región costa del país se tiene una humedad relativa que se encuentra en el rango de 70% y el 85%. Según un estudio realizado por la organización de estados americanos, la humedad media en la región amazónica oscila entre 85% y 90%. La mínima mensual registrada en las estaciones analizadas fue de 79% y la máxima de 92%. [46]. En la región sierra del país se tiene que la humedad relativa media oscila entre el 50% y 80%.

Adicional a esto se realizó un estudio en cada una de las estaciones meteorológicas que posee el INAMHI, de la humedad relativa y la precipitación que existen alrededor de estas estaciones. Dando así valores de humedad relativa de 27% a 92% dependiendo del lugar analizado [47].

Considerando que la saturación se produce por un descenso de temperatura con la misma humedad relativa, se puede determinar que los elementos que se encuentren dentro del automóvil y no están expuestos al contacto directo de agentes atmosféricos, tienen una baja probabilidad de ser afectados por la saturación, a pesar de los altos niveles de humedad en el exterior, porque la temperatura interior del automóvil no cambia abruptamente como lo hace el ambiente exterior.

Mediante este análisis se han tomado en cuenta los siguientes requerimientos con el fin de evitar la corrosión en el sistema de monitoreo web:

- Los dispositivos electrónicos del sistema deben estar protegidos por un material anticorrosivo, para evitar los efectos de la saturación y proteger al sistema del contacto directo con agentes atmosféricos, por esta razón se ha seleccionado una caja de plástico capaz de soportar altas temperaturas.
- La caja donde se encuentra el sistema, debe tener cavidades, con el fin de que el aire se filtre y se evite la acumulación de polvo y humedad.

4.2.2. Requerimientos del Servidor Web

Los parámetros necesarios para el servidor son el acceso continuo e ilimitado a internet. Este debe poseer una IP Pública fija o variable para que pueda ser reconocido en internet, adicional a esto se requiere de un dominio web para asignarlo a la IP pública.

Debido a que el servidor va a ser utilizado solo por el propietario del automóvil, las peticiones de información al mismo son pocas, en comparación a servidores con una

gran cantidad de usuarios que realizan peticiones diariamente, solicitando al servidor un gran ancho de banda y gran procesamiento de datos. En este proyecto los requerimientos de procesamiento y el ancho de banda requerido no son exigentes. Por lo tanto, se requiere un ancho de banda mínimo de 2Mbps.

Considerando que el servidor se encuentra ubicado en el hogar del propietario, debe utilizar un espacio físico reducido y de un bajo consumo energético.

4.3. Selección de Hardware

Para la selección de hardware se tomó como referencia la utilización de productos de bajo costo que sean de hardware libre, además de tomar las consideraciones del análisis de requerimientos del apartado 4.2. El sistema será construido con módulos y podrá ser escalable, todos los diseños del circuito son libres y se encuentran en el CD de este proyecto de titulación, pueden ser utilizados sin ninguna restricción.

4.3.1. Hardware de Sistema en el Automóvil

El sistema en la automóvil cuenta de nueve etapas, las cuales están coordinados por el Modulo Procesador de Datos,

En la Figura 4.4 se representa todas las etapas del sistema en el automóvil.

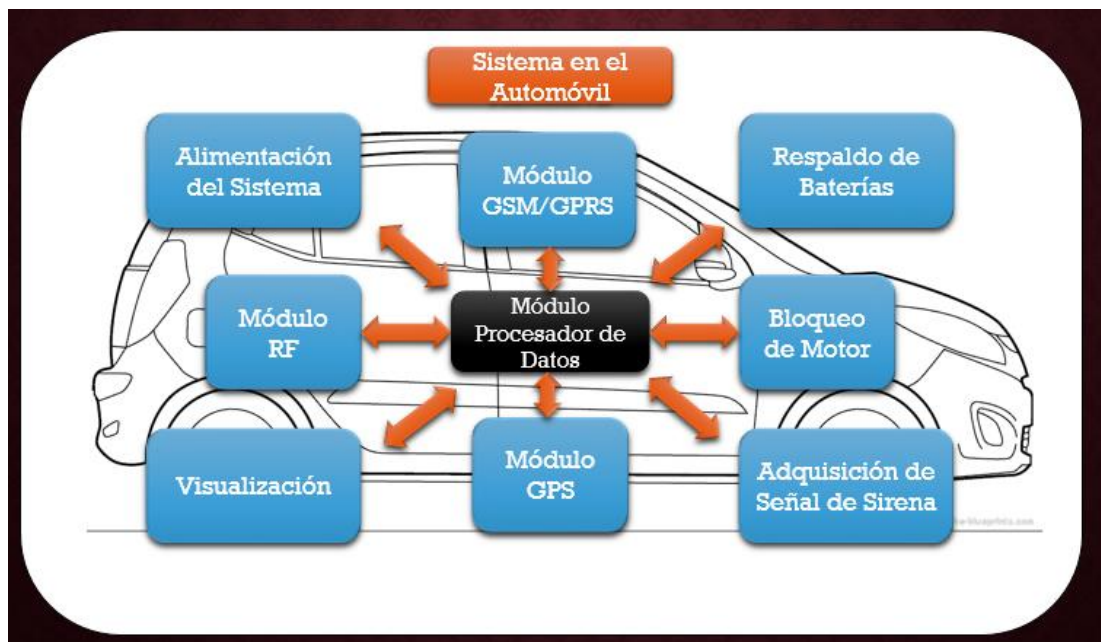


Figura 4.4 Composición del Módulo Procesador de Datos




Fuente: Autor

4.3.1.1. Módulo Procesador de Datos

Esta etapa realiza todos los procesos del sistema en el automóvil, se encarga de recibir los mensajes de texto y compararlos para realizar las diversas acciones del sistema, obtener las coordenadas geográficas de localización del automóvil, bloquear el motor, abrir y cerrar puertas, notificar cuando la alarma se activa, y enviar datos a través de mensajes de textos o accediendo a la red GPRS por medio de internet. El módulo procesador de datos debe estar conformado por un microcontrolador que gestione todas estas acciones.

Se propone el uso de placas de desarrollo para realizar el proyecto, debido a la gran facilidad que existe para trabajar con las mismas, además de tener los diseños y la programación de nuestros circuitos como OPEN SOURCE para compartirlos sin ningún problema. En la Tabla 4.4 se describen algunas tarjetas de desarrollo con su respectiva empresa que la fábrica.

Tabla 4.4 Tarjetas de Desarrollo y Empresas.

Empresas	Raspberry Pi	Arduino	Netduino
Detalles	 RaspberryPi		
Placa	Raspberry Pi3	Arduino Uno	Netduino 3
Microcontrolador	ARM Cortex A53	Atmega 328p	STMicro32bit
Voltaje de Funcionamiento	5V	6-20V	7,5-12V
Voltaje de Salida	5V, 3,3V	5V, 3,3V	5v, 3,3V
Pines I/O	40	14	22
Corriente Máxima por Pin	50mA	40mA	25mA
Velocidad de Reloj	1.2 GHz	16Mhz	168Mhz
RAM	1GB	2KB SRAM,1KB EEPROM, 32KB Flash	164KB SRAM, 384 KBFlash
USB	Micro Usb	Mini Usb	Micro Usb
Memoria Flash	1Gb	2-8Kb	164Kb
Software de Programación	Python, C, C++	Software Arduino IDE	Microsoft Visual C# Express
Dimensiones	8,6cmx5,7cm	6,8cmx5,3xm	8,13cmx5,33cm
Peso	45g	26g	45g
Costo	\$70	\$16	\$50

Fuente: Autor

La placa Raspberry Pi3 tiene buenas prestaciones, pero para los bajos procesos requeridos de sistema, no se utiliza su hardware debidamente, además posee un costo elevado para el propósito del sistema. Esta placa cuenta con un microprocesador en vez de un microcontrolador; lo que la convierte en un ordenador, posee una arquitectura de 64bits y es capaz de ejecutar sistemas operativos. Dentro de sistema en el automóvil no se utilizó esta placa, pero para la implementación del servidor web se consideró el uso de esta.

Netduino es una placa de desarrollo que utiliza el .Net Micro Framework, las prestaciones de esta placa son un poco mayores a las de Arduino, siendo también su costo más elevado. Dentro de Netduino existen placas de desarrollo que poseen compatibilidad con los dispositivos que se usan en Arduino, pero no posee placas en pequeñas dimensiones.

Arduino posee las prestaciones necesarias para el sistema, además tiene un bajo costo y la diversidad de placas que tiene hacen que Arduino sea el entorno de desarrollo escogido para el proyecto. Además de su alta comercialización en el país, es mucho más fácil encontrar una placa Arduino en comparación a las placas descritas anteriormente

Dentro de Arduino existen una variedad de placas de desarrollo, la placa comparada fue un Arduino UNO la cual tiene unas dimensiones 6.8cmx5.3cm con un peso de 25g, una placa de desarrollo que tiene las mismas funcionalidades, pero en tamaño y peso reducidos, es la placa Arduino NANO que tiene unas dimensiones de 4.5cmx1.8cm con un peso de 5g. Debido a que Arduino NANO tiene las mismas características que un Arduino UNO, a un tamaño y peso reducido, se optó por la placa de desarrollo Arduino NANO para trabajar en el sistema. En la Tabla 4.5 se describen las características técnicas de un Arduino NANO.

Tabla 4.5 Especificaciones Técnicas de Arduino NANO

Placas	Arduino NANO
Microcontrolador	ATmega328P
Voltaje de Operación	5V
Voltaje de Entrada Limite	6v-20V
Pines I/O	14
PWM	6
UART, SPI, I2C	SI
Pines de entrada Analógica	8
Corriente por I/O Pin	40mA
Corriente por 3.3 v Pin	50mA
Memoria Flash	16KB(ATmega168) 32KB(ATmega328)
SRAM	1KB
EEPROM	512b-Atmega168 1KB-Atmega328
Velocidad de Reloj	16MHz
Conector USB	Mini USB
Largo	45mm
Ancho	18mm
Temperatura de Trabajo	-40°C a 105°C
Peso	5g
Costo	\$8

Fuente: Autor, basado en [48]

De acuerdo a la hoja de datos [49] del microcontrolador ATmega 328p se encuentra que el rango de temperatura de operación de este microcontrolador es de -40°C a 105°C, de esta manera se cumple con el requerimiento técnico de temperatura


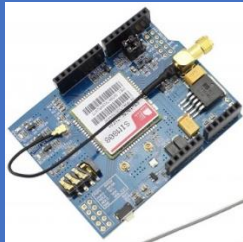

4.3.1.2. Módulo GSM/GPRS

Se encarga de enviar y recibir SMS para controlar las acciones del sistema, además este módulo debe poseer la capacidad de conectarse a Internet. Con estas características se buscó en el mercado módulos que trabajen en GSM y GPRS. La mejor opción dentro de módulos GSM y GPRS es la compañía SIMCOM la cual produce chips de telefonía móvil que tienen la posibilidad de conectarse a redes GSM/GPRS. La empresa SIMCOM produce solo los microchips, no produce los módulos para trabajar con microcontroladores, estos módulos son creados por terceras empresas que fabrican placas con estos chips para trabajar con ellos de una manera

más sencilla. Estos módulos son compatibles con las placas Arduino y tienen una comunicación serial para interactuar con ellas.

A continuación, en la Tabla 4.6 se muestra una comparación de módulos GSM/GPRS existentes en el mercado.

Tabla 4.6 Comparación de Módulos GSM/GPRS

	Sim900 Shield	Sim808 Shield	Sim 900
Modelos			
Características			
Costo	\$45	\$60	\$50
Chip Sim	SIM900	SIM808	SIM900
Banda de Operación	850/900/1800/1900 MHz	850/900/1800/1900 MHz	850/900/1800/1900 MHz
Dimensiones	8.5x5.7x2 cm	6.5x5.4x1.3cm	5x4x0,5cm
Voltaje de Trabajo	5V	5V	5V
Comunicación	Serial	Serial	Serial
RED	GSM/GPRS	GSM/GPRS	GSM/GPRS
GPS Interno	NO	SI	NO

Fuente: Autor

Se escogió trabajar con el módulo sim 900 de reducidas dimensiones, debido a que la placa seleccionada es un Arduino NANO y no se pueden acoplar uno encima de otro. Este módulo posee el chip SIM900 que es el más comercializado y tiene todas las funciones estándar de un celular que opera en las bandas de GSM y GPRS. Las bandas de frecuencias que opera este módulo es de 850MHz, 900MHz, 1800MHz y 1900MHz de esta manera cumple con el requerimiento de conexión a las redes GSM y GPRS, por lo tanto, es compatible con cualquier operadora de telefonía móvil del país

Para el proyecto se optó por utilizar la red de telefonía móvil Movistar, pero puede ser cambiada por cualquier operadora de telefonía móvil del país. En la Figura 4.5 Se observa el rango de cobertura de la señal 2G de Movistar en el país.

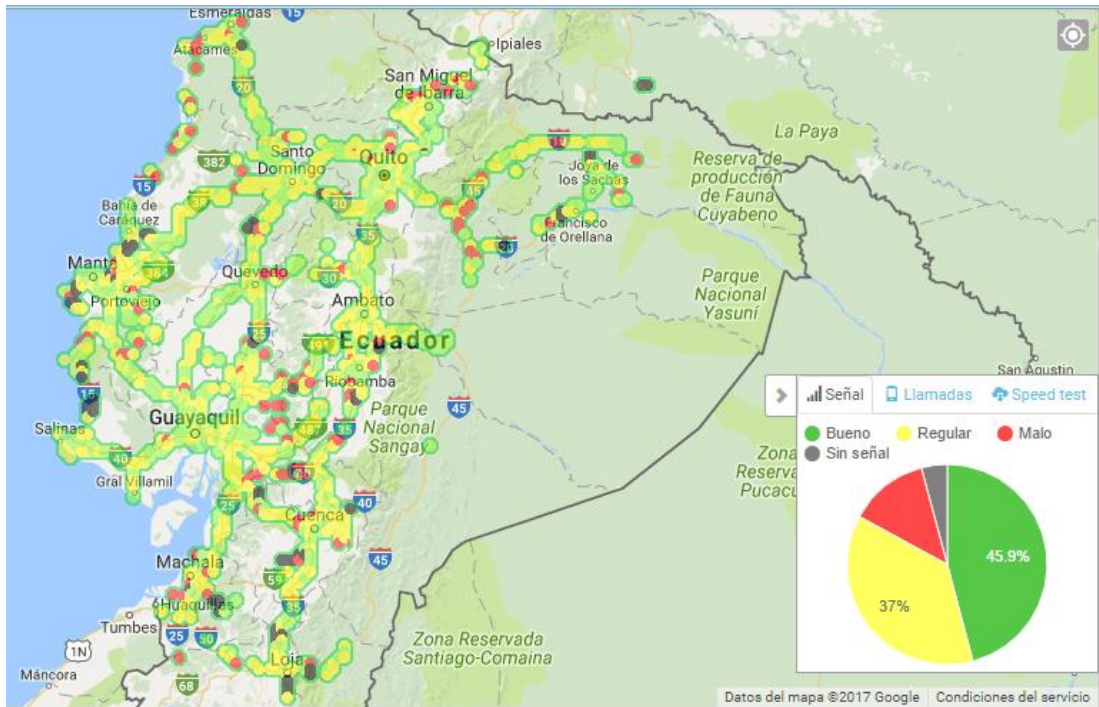


Figura 4.5 Cobertura de la Señal 2G de Movistar en el país.

Fuente: [50]


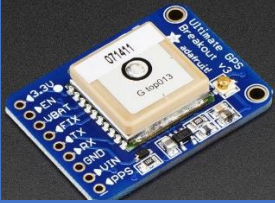
Con la ayuda de la hoja de datos del chip sim900 se observa que el rango de operación de este módulo es de -30°C a 80°C , estando dentro de los requerimientos técnicos de temperatura. Las características completas del microchip SIM900 se encuentran en el Anexo D.

4.3.1.3. Módulo GPS

Este módulo se encarga de recibir coordenadas geográficas de latitud y longitud, y enviarlas al Arduino NANO para procesarlas y realizar las acciones correspondientes.

Actualmente existen módulos receptores que trabajan con los cuatro sistemas de posicionamiento, pero su costo es elevado. Gracias a esto se buscó en el mercado módulos receptores que trabajen con el sistema de posicionamiento GPS; los requerimientos para este módulo es su tamaño, voltaje de alimentación, costo y temperatura de trabajo. En la Tabla 4.7 se representan los diversos módulos GPS en el mercado.

Tabla 4.7 Comparación de Módulos GPS

Módulo	NEO UBLOX 6M	Adafruit GPS Breakout
Características		
Costo	\$25	\$50
Dimensiones	25mm x 35mm	25.5mm x 35mm
Tensión de Operación	2.7 – 3.3V	3.0-5.5VDC
CHIP	NEO 6m	MTK3339
Baud Rate	9600	9600
Sensibilidad de Tracking	-161dBm	-165dBm
Arranque en frio	27s	34s
Canales y Frecuencia	50 a 5Hz	66 a 10Hz
Características Adicionales	UART NMEA USB SPI	UART NMEA

Fuente: Autor

Ambos módulos presentan características similares y dimensiones similares, por lo tanto, se escogió el módulo GPS NEO UBLOX 6M por su costo en comparación con la placa producida por Adafruit Industries.

En la hoja de datos del chip NEO UBLOX 6M se visualiza que las temperaturas de trabajo de este módulo son -40°C a 85°C lo cual cumple con los requerimientos de temperatura del sistema. El datasheet completo del microchip Ublox 6M se encuentra en el Anexo C

4.3.1.4. Módulo de Radio Frecuencia

El automóvil Hyundai i10 posee un sistema de alarma convencional con un bloqueo centralizado de puertas, este tipo de alarmas vehiculares poseen un transmisor en forma de llavero que envía comandos hacia la alarma de manera inalámbrica, dicho llavero puede abrir, cerrar, silenciar o activar la sirena del automóvil, trabajando a una frecuencia de 340 a 430 MHz enviando una serie de comandos codificados hacia su receptor. Por medio de una librería en Arduino se pudo decodificar dichos comandos enviados hacia la alarma y se utilizan nuevamente en el sistema de rastreo para abrir y cerrar las puertas del vehículo por estos códigos.

Para recrear el envío de estos códigos se utilizó una librería en Arduino y un módulo transmisor RF. Normalmente se puede diseñar un transmisor de radio frecuencia, pero el costo en comparación con adquirir un módulo ya fabricado es mucho más barato, por esta razón se utilizó un módulo prefabricado para realizar esta acción. El módulo de Radio Frecuencia utilizado se observa en la figura 4.6.



Figura 4.6 Módulos de Recepción y Transmisión RF

Fuente: [51]

Este módulo posee un transmisor y receptor, pero para este propósito se utiliza solo el transmisor para comunicarse con el receptor ya existente dentro de la alarma del automóvil. El transmisor posee tres pines DATA, VCC y GND, se utiliza una comunicación serial para enviar los datos desde Arduino NANO hacia el pin DATA del módulo.

4.3.1.5. Bloqueo de Encendido de Motor

Para la operación de bloquear el motor del automóvil, es decir, apagarlo si este se encuentra en movimiento e impedir su encendido, únicamente se corta la tensión que existe en la llave de giro en la posición ON. En la figura 4.7 se observa como puede ser realizada esta acción.

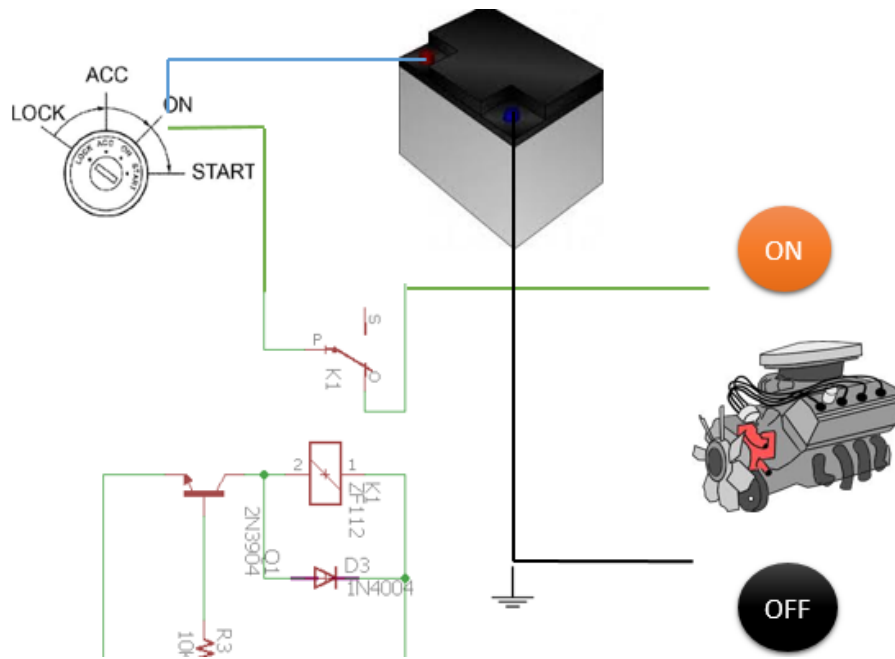


Figura 4.7 Etapa de Apagado de Motor

Fuente: Autor

Se utiliza un relé de 5V en la posición normalmente cerrado para romper la tensión que va hacia el control principal del sistema y apaga todo el automóvil, sin dejarlo encender hasta que el relé pase nuevamente a su estado normalmente cerrado. Los pines de Arduino NANO otorga una corriente máxima por pin de 40mA y se va a utilizar un relé de 5V que consume 70mA de corriente cuando se energiza su bobina; por lo tanto, el pin de salida de Arduino no es capaz de activar el relé y puede ocasionar daños a la placa.

Para resolver este problema se utilizó un transistor 2N3904 para activar y desactivar este relé. El transistor 2N3904 va a trabajar en corte y en saturación, mas no en su zona de trabajo normal, por lo tanto, se procede a calcular el circuito para que trabaje en saturación. En la figura 4.8 se visualiza el circuito para activar y desactivar el relé de 5V.

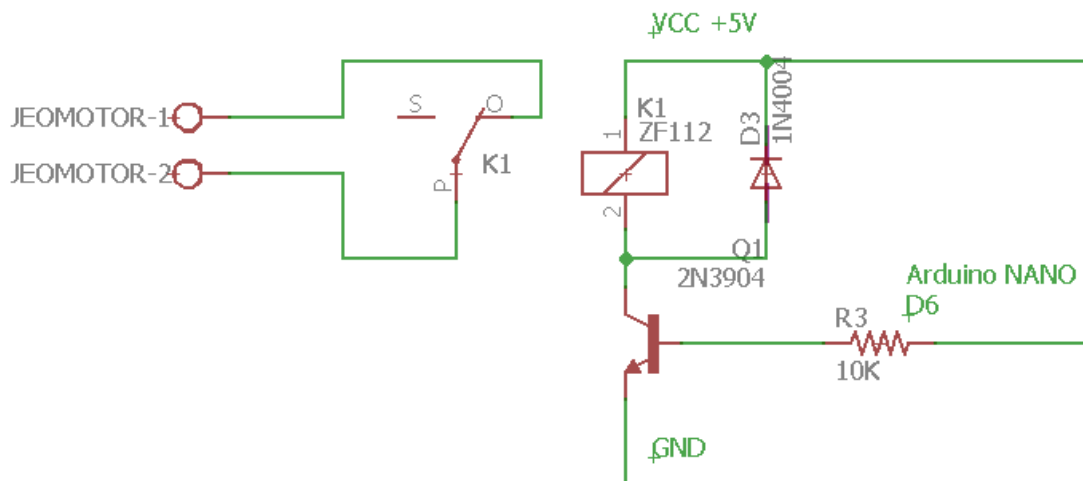


Figura 4.8 Esquema de Relé con transistor 2N3904

Fuente: Eagle-Autor

Cálculos:

Se necesita calcular la resistencia de base R3 del transistor 2N3904 de la figura 4.8 para que trabaje en corte y en saturación, conociendo que la corriente que consume el relé en su activación es de 75mA, se la aproxima a 80mA para un caso máximo de consumo de corriente. Esta corriente de activación del relé es la corriente de colector I_c . La Ecuación 4.2 ayuda a calcular la resistencia de base de un transistor para trabajar en saturación.

$$R_b = \frac{(V_{in} - V_{BE}) * HFE}{I_c} \quad (4.2)$$

En donde $I_c=80mA$, V_{in} es el voltaje de entrada en la base, V_{be} es el voltaje de polarización del diodo 0.7V y HFE es la ganancia del transistor. Según la hoja de datos del transistor 2N3904 ubicada en el Anexo G y la medición con un multímetro se obtiene con una temperatura ambiente de 25°C una ganancia es de 160. A medida que se incrementa la temperatura del transistor la ganancia HFE aumenta, suponiendo la temperatura del automóvil máxima a 80°C se tomó un HFE de 200. Remplazando todos los valores en la ecuación 4.1 se obtiene:

$$R_b = \frac{(5V - 0.7V) * 200}{80mA} \quad R_b = 10750\Omega \quad (4.1)$$

Con el valor de resistencia de base encontrada se procede a buscar una resistencia con ese valor en el mercado y se decidió usar una resistencia de 10K Ω para la resistencia

de base. De esta manera el transistor se comporta como un interruptor en corte (abierto) y saturación (cerrado). El Datasheet del transistor 2N3904 se encuentra en el Anexo G.

4.3.1.6. Adquisición de Señal de Sirena

Para la adquisición de señales de un automóvil se debe tener en cuenta, que todas las señales dentro de este funcionan con un voltaje de 12V, sea de puertas, seguros, vidrios, sirena, luces, etc. Para utilizar esta señal dentro del sistema se utiliza un opto acoplador 4N25 para unir la tensión de 12V de la sirena a un voltaje de 5V del sistema que pueda ser leído por los pines del Arduino NANO. En la Figura 4.9 se detalla la etapa de adquisición de la señal de 12V de la sirena del automóvil.

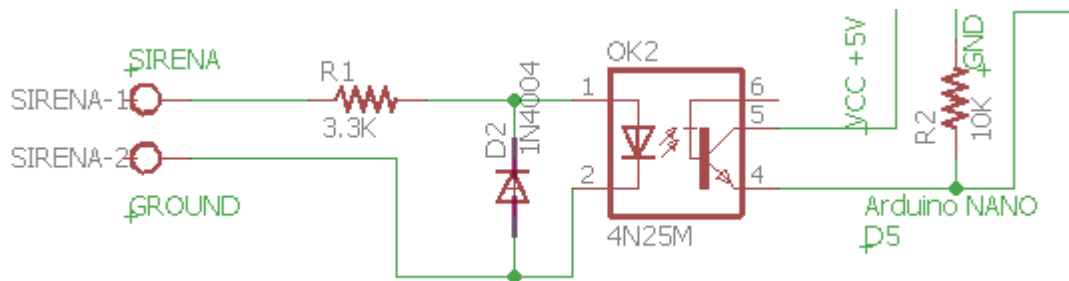


Figura 4.9 Etapa de Adquisición de Señal de Sirena

Fuente: Eagle-Autor

El circuito está conformado por una resistencia R1 de $3.3K\Omega$ que funciona para regular la corriente que entra al diodo led del opto acoplador, un diodo D2 1N4004 en anti paralelo protege al led del opto acoplador, de picos de voltaje provocados por la conmutación de bobinas de relés o cargas inductivas, situación muy común en cualquier automóvil; y una resistencia R2 de $10K\Omega$ que funciona como resistencia pull-down que activa o desactiva la entrada hacia el microcontrolador

Cálculos:

Conociendo que la batería del automóvil trabaja a un voltaje mínimo de 10V y un máximo de 13.5V, se tomó como voltaje de entrada 10V para un cálculo en las peores condiciones de la batería del auto, además una caída de voltaje de 1.35V del led del opto acoplador y una resistencia de Led de $3.3K\Omega$. La corriente que circula por el led I_F se define por la ecuación 4.2:

$$I_F = \frac{V_{min} - V_{IRLED}}{R} \quad (4.2)$$

$$I_F = \frac{10V - 1.35V}{3.3K\Omega}$$

$$I_F = 2.62mA$$

Este opto acoplador tiene una corriente maxima de led I_f de 100mA, dentro de este margen de corriente el led interno del opto acoplador trabaja correctamente, como se tiene una corriente de led de 2.62mA el led interno no sufre daños. Cada opto acoplador posee un CTR (Current Transfer Ratio) que es un numero mediante el cual se conoce cuanta corriente pasa del diodo emisor a la base del fototransistor del opto acoplador. En otras palabras, si se tiene un CTR de 1, quiere decir que por cada mA que entre por el led emisor, se obtendrá la misma cantidad de mA a la salida del fototransistor.

Con ayuda de la Figura 4.10 del Datasheet del opto acoplador 4N25 se puede observar que para una corriente de 2.62mA se obtiene un CTR de aproximadamente 0.9 por lo tanto la corriente que circula entre colector y emisor del fototransistor es de 2,36mA lo cual es suficiente para polarizar el pin D5 del Arduino NANO.

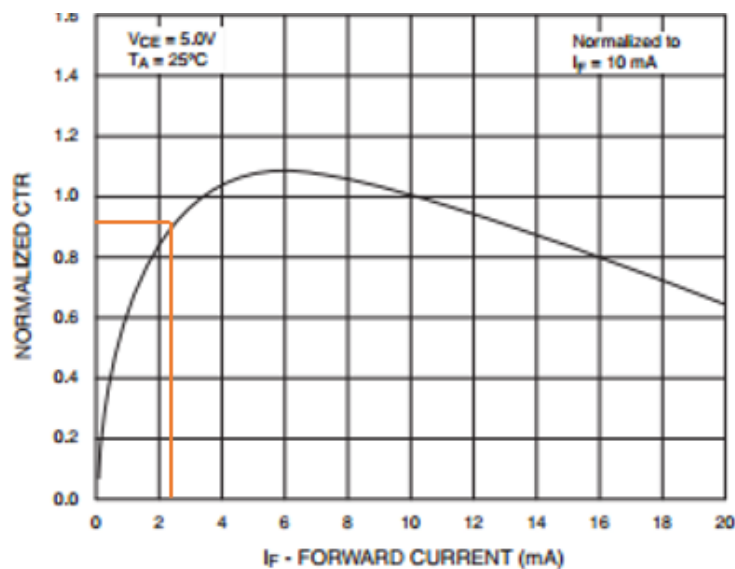


Figura 4.10 Representación Gráfica de I_F vs CTR

Fuente: Anexos F

El valor de la resistencia pull-down debe ser un valor grande alrededor de $10K\Omega$ para que no exista problemas de funcionamiento. El Datasheet completo se encuentra en el Anexo F

4.3.1.7. Alimentación del Sistema

En este apartado se analiza la alimentación del sistema de rastreo para el correcto funcionamiento de sus módulos. La placa Arduino NANO tiene un regulador de voltaje interno que regula su voltaje de entrada de 6-22V a un voltaje de 5V. Esto lo realiza mediante un regulador de voltaje 7805 que tiene incorporado en su placa, el cual toma el voltaje del pin Vin y lo transforma a 5V. Normalmente la placa Arduino NANO puede soportar la tensión de la batería del automóvil que es de 10V a 13.5V en su pin Vin; y podría operarse todo los demás módulos del sistema con el voltaje de salida de este regulador en el pin +5V, pero esto no es recomendable debido a que toda la carga del sistema pasaría primero por la placa Arduino NANO para su regulación y luego se distribuirá a los módulos, en caso de cortocircuito la placa principal quedaría inoperable. Otro punto importante para no usar la alimentación regulada de Arduino es su regulador de voltaje 7805 que posee poca eficiencia energética, debido a que disipan en calor el sobrante de voltaje que se regula, adicional a esto otorgan poca corriente de salida que puede ocasionar fallos de operación del sistema.

Con estos inconvenientes se escogió usar un regulador externo que otorgue directamente 5V a todo el sistema y no dependa de la regulación de Arduino NANO por su pin Vin. En la actualidad se puede obtener módulos de regulación de voltaje variable a un costo muy reducido con una alta eficiencia energética que no gastan potencia en forma de calor excesivamente, a diferencia si se diseña un regulador con componentes electrónicos como 7812 o 7805 el costo es similar y la eficiencia es mejor en dichos módulos. Por esta razón se escogió el módulo regulador de voltaje variable XL4005 BUCK CONVERTER de la figura 4.11.

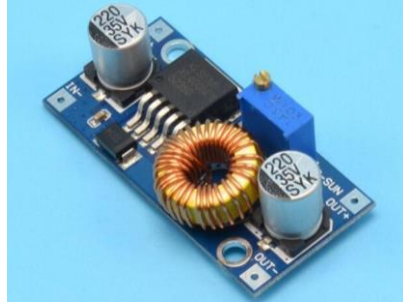


Figura 4.11 Módulo XL4005, Reductor de Voltaje (BUCK CONVERTER)

Fuente: [52]

Como se aprecia en la figura 4.11 estos módulos reductores de voltaje son regulables y de buena calidad, otorgan una eficiencia muy estable y tiene dimensiones reducidas.

Las características de este módulo son las siguientes:

- Voltajes de entrada: 5V a 30V
- Voltajes de Salida: 0,8V a 24V
- Corriente máxima de salida: 5A
- Eficiencia de Conversión Máxima: 90%
- Dimensiones: 43mm * 21mm * 14mm
- Temperatura de Trabajo: -40°C a 85°C

4.3.1.8. Respaldo de Baterías

El sistema cuenta con un sistema de baterías independientes de la batería del automóvil, estas baterías son recargables y se activan únicamente si la fuente de alimentación principal, la batería del automóvil, llegasen a fallar. Se usaron dos baterías 18650 de Li-ion con una capacidad de 3000mAh a 3.7V cada una, de la marca UltraFire.



Figura 4.12 Baterías Recargables UltraFire

Fuente: [53]

Adema de las baterías se utilizó un módulo cargador de baterías, que carga y controla la sobretensión de las mismas, así como cortocircuitos. El módulo TP-4056 es un cargador y protector de baterías 18650 que utiliza un conector micro-USB para cargar las baterías a una corriente de carga máxima de 1A.

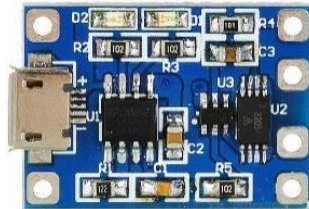


Figura 4.13 Módulo Cargador y Protector de Baterías TP-4056

Fuente: [52]

El voltaje de operación de estas baterías es de 3.7 V y no es suficiente para funcionar completamente todo el sistema de rastreo, por esta razón se elevó dicho voltaje a 5V con la ayuda de un módulo elevador de voltaje, así mismo como para la alimentación del sistema, se utilizó un módulo reductor de voltaje Buck Converter por su eficiencia energética, de la misma manera se utilizó un elevador de voltaje en forma de módulo para aprovechar la conversión de tensiones con la máxima eficiencia posible. Para este caso se seleccionó trabajar con el módulo elevador de voltaje Boost Converter XL6009 que tiene las mismas dimensiones que el módulo reductor XL4005.



Figura 4.14 Módulo Elevador de Voltaje Boost Converter XL6009

Fuente: [52]

Las características de este módulo son las siguientes:

- Voltajes de entrada: 32V a 3,2V
- Voltajes de Salida: 4V a 38V
- Corriente máxima de salida: 4A
- Eficiencia de Conversión Máxima: 90%
- Dimensiones: 43mm * 21mm * 14mm
- Temperatura de Trabajo: -40°C a 85°C



De esta manera se eleva el voltaje de las baterías de 3.7V a 5V para poder alimentar de manera correcta el sistema de rastreo cuando la batería del automóvil deje de funcionar.

4.3.1.9. Visualización

El módulo de visualización es el encargado de indicar al usuario las respectivas acciones que realiza el sistema. Este módulo puede ser descartado del sistema de rastreo o incluido dependiendo del requerimiento del propietario. En este proyecto se lo coloco para indicar cada una de las acciones que realiza el sistema, mostrando indicadores que facilitaron la demostración del sistema completo.

Se optó por usar una pantalla LCD en la cual se visualizó por texto acciones que realiza el sistema. En la Tabla 4.8 se indica las diversas pantallas LCD de venta en el mercado para trabajar con Arduino.

Tabla 4.8 Comparación de Pantallas para Arduino

Pantallas	LCD 16X2	LCD 16x2 I2C	OLED 0.96" 128x32
Características			
Precio	\$5	\$10	\$15
Dimensiones	8cmx3,5cm	8cmx3,5cm	2,5cmx2,5cm
Pines que necesita	12	4	4
Librerías Arduino	LiquidCrystal.h	LiquidCrystalI2C.h	U8GLIB Adafruit
Gráficos	Texto	Texto	Texto Gráficos
Comunicación	N/A	I2C	I2C

Fuente: Autor

Para el sistema se usó la pantalla más pequeña la OLED 0.96" 128x32 que también ahorra pines a la hora de acoplarlo con Arduino. Esta pantalla posee una librería para Arduino la cual es U8GLIB [54], esta librería permite imprimir texto e imágenes en los colores que acepte la pantalla, esta pantalla posee solo los colores blancos y negros, a diferencia de las más costosas que poseen los pines RGB para usar todos los colores. La pantalla utiliza una comunicación serial para acoplarse con Arduino NANO en los pines SDA y SCL además tiene una tensión de trabajo de 5V.

En esta pantalla se imprimió todas las acciones que realiza el sistema, para ahorrar más espacio y consumo energético se puede prescindir de esta y trabajar con leds indicadores.

4.3.2. Hardware para Sistema de Monitoreo

Dentro del sistema de monitoreo se cuenta con el servidor que almacena los datos que envía el sistema de rastreo y también el teléfono celular que recibe notificaciones del mismo. Las notificaciones que se envían al teléfono celular son:

- Alertas de Alarma.

Envía un SMS advirtiendo que la alarma del vehículo ha sido activada. Este mensaje llega cuando la sirena de la alarma esta activa por 5 segundos, después de este tiempo el sistema se conecta a la red GSM y envía un SMS al teléfono del propietario, el tiempo de envío de SMS varía dependiendo de la señal y conectividad a la red GSM.



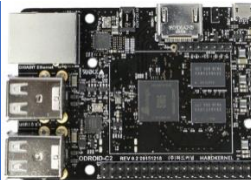
- Posición del Vehículo.

Envía un SMS con un link en GOOGLE MAPS con las coordenadas de latitud y longitud exactas del vehículo además de su velocidad. Este link puede ser usado en teléfonos con sistema operativo Android o IOS para desplegar la aplicación de GOOGLE MAPS e interpretar el link enviado por el sistema. En el apartado del teléfono celular no se analiza ningún hardware porque solo recibe SMS y no hay necesidad de procesar datos. En cambio, en el servidor web se analizó los diversos dispositivos que permiten almacenar un servidor.

Los requerimientos de un servidor son muy amplios y dependen de la operación que vaya a realizar el servidor. Debido a que el propietario es el dueño del servidor, los usuarios que requieran información de este van a ser pocos, en comparación a las empresas que ofrecen estos servicios, las cuales tienen muchas peticiones de información. El servidor de monitoreo cuenta con una página web que despliega un mapa y ubica los datos enviados por el sistema de rastreo, además posee una base de datos en la cual almacena la información para luego ser procesada por el servidor. Un servidor puede ser instalado en cualquier computador, el requisito para el proyecto es que sea un computador de dimensiones pequeñas y con un bajo consumo de energía para que pueda estar conectado las 24 horas del día en la casa del propietario sin ocupar

demasiado espacio y no consumir mucha energía. En la Tabla 4.9 se realiza una comparación con distintos microcomputadores.

Tabla 4.9 Comparación de Mini Ordenadores OpenSource.

Marca			
Modelo	Orange Pi PC 2	Raspberry Pi3	ODROID C2
Procesador	Cortex A53 64 bit	ARMv8 64-bit	ARMv8
Velocidad	1.2GHz	1.2GHz	1.5GHz
Memoria RAM	1GB	1GB	2GB
Almacenamiento	microSD	microSD	microSD
Conectividad	Ethernet HDMI	WiFi 802.11n Bluetooth 4.1 Ethernet Port HDMI	Ethernet Gigabit Wifi 802.11n HDMI
GPIO	40	40	40
Consumo de Potencia	5W	4W	6W
Precio	\$80	\$70	\$100

Fuente: Autor

En el mercado existen muchas empresas, aparte de las citadas en la tabla 4.9, que se dedican a la construcción de mini computadores, pero se realizó el análisis con las más sobresalientes. Las características de cada una son similares, pero detrás de cada una de las empresas existe una comunidad de desarrolladores que mejoran el sistema de cada una de estas placas.

De estas se escogió la marca Raspberry Pi por tener una comunidad muy amplia, su costo bajo, su bajo consumo energético y una gran variedad de sistemas operativos que son compatibles. Existen mucha información en internet y mucho desarrollo que se ha hecho con este minicomputador, por esta razón se escogió la placa Raspberry Pi 3 [55] que es el último modelo que ha sacado al mercado la compañía Raspberry Pi..

4.4. Selección de Software

4.4.1. Software de Programación

Para la programación del microcontrolador se utilizó el software desarrollado por Arduino, es totalmente gratuito. El software para programar estas placas Arduino se conoce como ARDUINO IDE [56], el cual es un programa ejecutable para diversos sistemas operativos como MAC OSX, LINUX y WINDOWS. En la Figura 4.15 se observa el entorno de programación de Arduino.

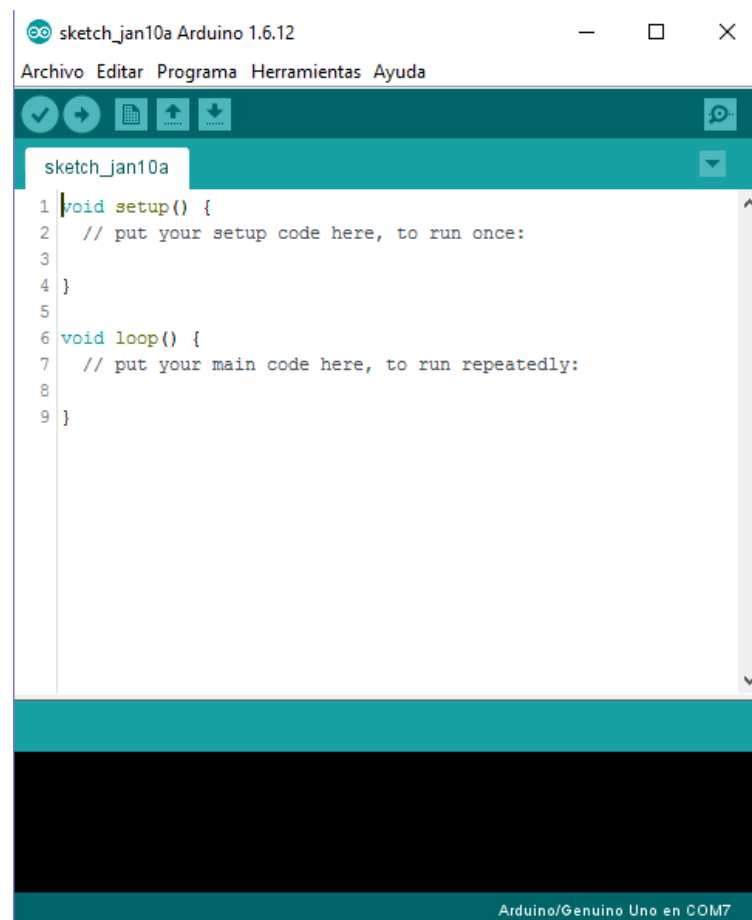


Figura 4.15 Entorno de Programación Arduino.

Fuente: Autor

La programación en Arduino se realiza en C con algunas funciones de C++, el software IDE puede programar todas las placas que Arduino ha fabricado con sus diferentes microcontroladores ATMEL, además posee un repositorio de librerías en línea que ayudan de gran manera a simplificar procesos de programación muy complicadas. Estas librerías son diseñadas por Arduino o por la comunidad de programadores que ayuda a mejorar este sistema con sus microcontroladores. Todo el entorno de

programación y los diseños de las tarjetas Arduino están disponibles para cualquier usuario de manera gratuita.

4.4.2. Software de Diseño de Circuitos PCB

Para elaborar el circuito del proyecto, así como su PCB se escogió el software EAGLE [57], que es totalmente gratis con una licencia Freeware, en su versión 7.7.

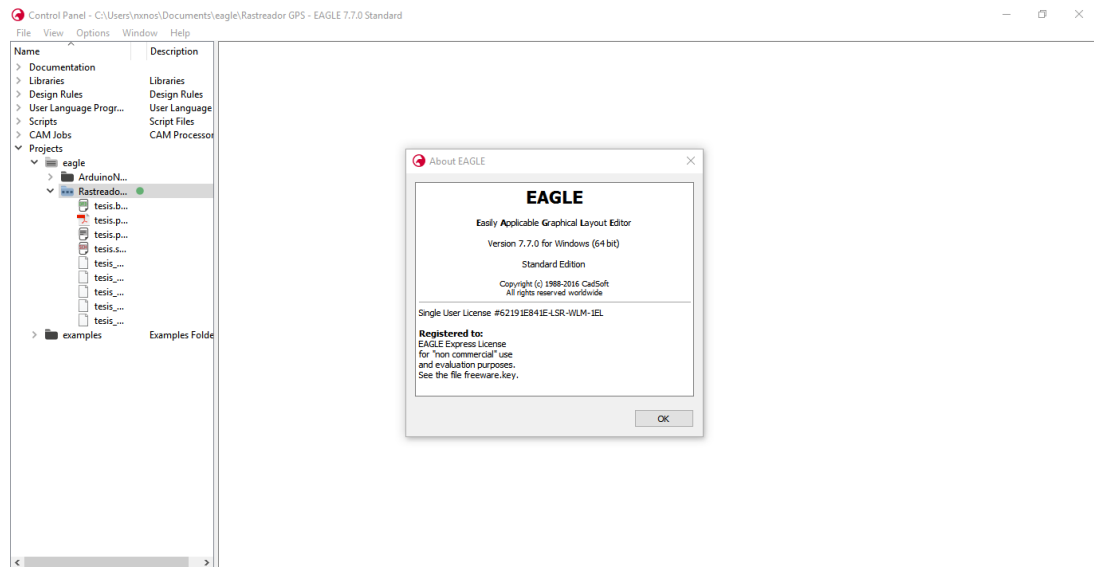


Figura 4.16 Entorno de Desarrollo EAGLE

Fuente: Autor

Este software es el más usado en el mundo de desarrollo DIY, está destinado para ingenieros y estudiantes. Empresas como Arduino Tienen diseñados sus placas en este software, los archivos de diseños que comparte Arduino son para EAGLE. Además, posee de muchas librerías y componentes en línea que ayudan a complementar todas sus funcionalidades.

4.4.3. Software de Servidor Web

Actualmente se puede instalar un servidor en cualquier computador dependiendo de su sistema operativo, se puede escoger entre:

- XAMP para cualquier sistema Linux, MAC OSx o WINDOWS
- LAMP Solo para sistemas Linux
- WAMP Solo para sistemas Windows
- MAMP Solo para sistemas MAC OSx

Debido a que se seleccionó la placa Raspberry Pi3 para trabajar como servidor web, se necesita de un sistema operativo compatible con la misma, el sistema recomendado para trabajar con estas placas es Raspbian. Raspbian es el sistema operativo nativo para las placas Raspberry Pi, es una distribución de Debían (Linux) que ha sido optimizada para el perfecto uso de todos los componentes de estas placas, tiene el respaldo de la empresa Raspberry que continuamente corrige errores en el sistema y siempre lo mantiene actualizado.

Debido a que se trabaja con Raspbian, se procede a instalar el Servidor Web con LAMP la cual es un conjunto de aplicaciones para habilitar servidores en entornos Linux, estas aplicaciones son:

Apache: Es una aplicación que permite instalar el servidor web en la Raspberry Pi3, esta aplicación se encarga de gestionar el almacenamiento dedicado para las páginas web, así como el lenguaje HTML que se encarga de interpretarlas, en un directorio raíz conocido como `/var/www/html/`.

MySQL: Es una aplicación que instala un gestor de base de datos con MySQL que sirve para almacenar datos en grandes cantidades, además de esto instala phpMyAdmin que es un gestor de bases de datos MySQL que facilita la interacción de estas bases de datos.

PHP: Es una aplicación que instala los componentes necesarios para utilizar PHP en las páginas web que se almacenan en la Raspberry Pi, PHP es un lenguaje de programación indispensable en toda la web.

Todas estas aplicaciones en conjunto con los servicios FTP y SSH convierten a la Raspberry Pi3 en un servidor web operable, en el cual se ubicarán todos los programas de la aplicación web del proyecto.

4.4.4. Software para Programación de Aplicación Web

La mayoría de páginas web utilizan diferentes tecnologías de programación, una página web no está construida en solo un lenguaje de programación, sino que se agrupan diversas tecnologías para desarrollarla; y cada una de estas tiene un lenguaje de programación diferente. Las tecnologías básicas para crear una página web son: HTML, PHP, JavaScript. Existen muchas más tecnologías y lenguajes de programación para diseñar una página web, pero estas son las más fundamentales.

Debido a la diversidad de lenguajes de programación que existen en una página web se necesita un editor de texto que permita realizar todas estas programaciones.

Para la realización del proyecto se utilizó dos programas que realizan estas acciones.

SUBLIME TEXT [58]: Es un editor de texto que permite la utilización de diversos lenguajes de programación, entre ellos HTML, PHP, JavaScript, AJAX, etc. Este programa es totalmente gratis y de uso libre, en su versión básica no puede ofrecer grandes características, pero puede instalarse paquetes de lenguajes que faciliten la utilización y programación de los mismos. Solo puede editar texto mas no visualizar el resultado de la página que se está diseñando.

DREAM WEAVER: Es un software para diseño Web de licencia pagada, producido por ADOBE, cuenta con una interfaz gráfica que permite visualizar la página web que se está construyendo, este software es compatible con todos los lenguajes de programación web existentes. Contiene muchas funcionalidades extras tales como conectarse al servidor donde se va a alojar la página web y editarlo remotamente. Este software está pensado para mejorar la maquetación con HTML.

4.5. Procesamiento de Información Recibida de GPS

El módulo GPS UBLOX NEO 6M que fue escogido con el sistema de rastreo, utiliza una comunicación serial, en esta comunicación se envían datos constantemente desde el módulo GPS hacia la placa Arduino NANO, estos datos son enviados con un baud rate de 9600.

En el datasheet del módulo GPS se encuentra que estos datos utilizan el protocolo NMEA 0183, protocolo creado por National Marine Electronics Association, que fue fundada por un grupo de fabricantes de electrónica para obtener un sistema común de comunicación entre las diferentes marcas de electrónica naval. Actualmente todos los dispositivos receptores GPS utilizan el protocolo NMEA 0183 para recibir la información de los satélites GPS. Todos los códigos de NMEA se pueden observar en la figura 4.17.

```

$GPBOD - Bearing, origin to destination
$GPBWC - Bearing and distance to waypoint, great circle
$GPGGA - Global Positioning System Fix Data
$GPGLL - Geographic position, latitude / longitude
$GPGSA - GPS DOP and active satellites
$GPGSV - GPS Satellites in view
$GPHDT - Heading, True
$GPR00 - List of waypoints in currently active route
$GPRMA - Recommended minimum specific Loran-C data
$GPRMB - Recommended minimum navigation info
$GPRMC - Recommended minimum specific GPS/Transit data
$GPRT - Routes
$GPTRF - Transit Fix Data
$GPSTN - Multiple Data ID
$GPVBW - Dual Ground / Water Speed
$GPVTG - Track made good and ground speed
$GPWPL - Waypoint location
$GPXTE - Cross-track error, Measured
$GPZDA - Date & Time

```

Figura 4.17 Tramas del protocolo NMEA 0183

Fuente: [59]

Normalmente conectado al computador, mediante Arduino, el módulo GPS envía cada segundo tramas en formato NMEA como la Figura 4.18.

```

COM7 (Arduino/Genuino Uno)
$GPR,44,07,19,342,24,08,45,007,34*76
$GPGSV,3,2,09,09,48,294,34,17,09,221,29,22,23,163,35,23,67,210,39*72
$GPGSV,3,3,09,27,10,027,28*4B
$GPGLL,0117.01320,S,07838.09096,W,021904.00,A,A*6D
$GPRMC,021905.00,A,0117.01322,S,07838.09089,W,0.031,,120117,,A*7F
$GPVTG,,T,,M,0.031,N,0.057,K,A*23
$GPGGA,021905.00,0117.01322,S,07838.09089,W,1,09,1.01,2804.2,M,12.6,M,,*65
$GPGSA,A,3,23,09,01,08,03,22,07,27,17,,,,,1.63,1.01,1.28*0B
$GPGSV,3,1,09,01,48,154,45,03,24,181,44,07,19,342,25,08,45,007,34*77
$GPGSV,3,2,09,09,48,294,35,17,09,221,29,22,23,163,35,23,67,210,39*72
$GPGSV,3,3,09,27,10,027,28*4B
$GPGLL,0117.01322,S,07838.09089,W,021905.00,A,A*60
$GPRMC,021906.00,A,0117.01324,S,07838.09090,W,0.009,,120117,,A*79
$GPVTG,,T,,M,0.009,N,0.016,K,A*2D
$GPGGA,021906.00,0117.01324,S,07838.09090,W,1,09,1.01,2804.2,M,12.6,M,,*68
$GPGSA,A,3,23,09,01,08,03,22,07,27,17,,,,,1.63,1.01,1.28*0B
$GPGSV,3,1,10,01,48,154,45,03,24,181,44,07,19,342,25,08,45,007,34*7F
$GPGSV,3,2,10,09,48,294,35,14,,,,,23,17,09,221,28,22,23,163,35*46
$GPGSV,3,3,10,23,67,210,38,27,10,027,28*7B
$GPGLL,0117.01324,S,07838.09090,W,021906.00,A,A*6D
$GPRMC,021907.00,A,0117.01328,S,07838.09089,W,0.027,,120117,,A*70
$GPVTG,,T,,M,0.027,N,0.049,K,A*2B
$GPGGA,021907.00,0117.01328,S,07838.09089,W,1,09,1.01,2804.2,M,12.6,M,,*6D
$GPGSA,A,3,23,09,01,08,03,22,07,27,17,,,,,1.63,1.01,1.28*0B
$GPGSV,3,1,09,01,48,154,45,03,24,181,43,07,19,342,26,08,45,007,34*73
$GPGSV,3,2,09,09,48,294,35,17,09,221,28,22,23,163,35,23,67,210,38*72
$GPGSV,3,3,09,27,10,027,28*4B

```

Figura 4.18 Códigos NMEA receptados por GPS UBLOX NEO6M

Fuente: Autor

La trama que empieza con \$GPGGA es la más utilizada al momento de analizar los datos completos enviados por el satélite GPS.

\$GPGGA,021905.00,0117.01322,S,07838.09089,W,1,09,1.01,2804.2,M,12.6,M,,*65

Esta trama acumula todos los datos enviados del GPS separados por comas. En la Tabla 4.10 se clasifica cada dato de la Trama NMEA-GGA

Tabla 4.10 Distribución de Datos de la Trama GGA

Trama	Significado
\$GPGGA	Tipo de trama, puede ser cualquiera de la Figura 4.17
021905.00	Hora UTC en formato hhmmss.ss
0117.01322	Latitud en formato ddm.ddd
S	Indicador Norte N o Sur S
07838.09089	Longitud en formato ddm.ddd
W	Indicados Este E u Oeste W
1	Indicador de validez de posición 0: Posición no válida. El sistema no ha conseguido deducir una posición coherente con las distancias a los satélites 1: Posición válida en Modo SPS (Standard Positioning System). 2: Posición válida en Modo SPS (Standard Positioning System) con información Diferencial. Además de la señal de los satélites, el GPS considera la señal emitida por un emisor terrestre (Modo diferencial)
09	Numero de Satélites
1.01	Indicación de Precisión <1: Ideal 1-2: Excelente 2-5: Buena 5-10: Moderada 10-20: Aproximada >20: Pobre
2804.2	Separación entre el Geoide y el Elipsoide de Referencia. Altitud Elipsoidea = [Altitud sobre el nivel del Mar] + [Separación Geoide]
M	Unidad de la Separación Geoide.
12.6	Segundos de antigüedad de la corrección diferencial
M	Unidad de la Separación Geoide
*65	Checksum

Fuente: Autor

El módulo GPS trabaja únicamente como receptor de códigos del satélite GPS, y envía este tipo de tramas NMEA por su pin Tx serial. Dentro de Arduino se debe calcular dichas tramas y extraer los datos que se encuentran separados por comas, el proceso de transformar estas tramas NMEA a datos legibles se llama “PARSEADO”. Existen muchos algoritmos de PARSEADO que nos ayudan a obtener de manera más representativa los datos de latitud y longitud, pero son muy complejos. Gracias a Arduino se puede disponer de librerías para reducir la complejidad de estos cálculos.

TinyGPS ++ es una librería para Arduino que simplifica notablemente el proceso de parseado, el cual es realizado en segundo plano cuando se llama a esta función, y entrega los datos de latitud, longitud, tiempo, satélites, curso u otras funcionalidades

que se necesiten. Esta librería fue diseñada por Mikal Hart y es de uso libre, se puede conseguir en su página oficial. [60]

Gracias a esta librería se obtiene toda la información deseada del GPS como en la Figura 4.19.



```
COM7 (Arduino/Genuino Uno)

ample.ino
A simple demonstration of TinyGPS++ with an attached GPS module
Testing TinyGPS++ library v. 0.92
by Mikal Hart

DeviceExample.ino
A simple demonstration of TinyGPS++ with an attached GPS module
Testing TinyGPS++ library v. 0.92
by Mikal Hart

Location: -1.283519,-78.634788 Date/Time: 1/12/2017 02:33:57.00
Location: -1.283519,-78.634788 Date/Time: 1/12/2017 02:33:57.00
Location: -1.283519,-78.634788 Date/Time: 1/12/2017 02:33:57.00
Location: -1.283519,-78.634788 Date/Time: 1/12/2017 02:33:57.00
Location: -1.283519,-78.634788 Date/Time: 1/12/2017 02:33:57.00
Location: -1.283519,-78.634788 Date/Time: 1/12/2017 02:33:57.00
Location: -1.283521,-78.634796 Date/Time: 1/12/2017 02:33:58.00
Location: -1.283521,-78.634796 Date/Time: 1/12/2017 02:33:58.00
Location: -1.283521,-78.634796 Date/Time: 1/12/2017 02:33:58.00
Location: -1.283521,-78.634796 Date/Time: 1/12/2017 02:33:58.00
Location: -1.283521,-78.634796 Date/Time: 1/12/2017 02:33:58.00
Location: -1.283521,-78.634796 Date/Time: 1/12/2017 02:33:58.00
Location: -1.283521,-78.634796 Date/Time: 1/12/2017 02:33:58.00
Location: -1.283522,-78.634796 Date/Time: 1/12/2017 02:33:59.00
Location: -1.283522,-78.634796 Date/Time: 1/12/2017 02:33:59.00
Location: -1.283522,-78.634796 Date/Time: 1/12/2017 02:33:59.00
Location: -1.283522,-78.634796 Date/Time: 1/12/2017 02:33:59.00
Location: -1.283522,-78.634796 Date/Time: 1/12/2017 02:33:59.00
```

Figura 4.19 Datos de GPS con Librería TinyGPS ++

Fuente: Autor

4.6. Envío de Datos a través de GSM/GPRS

Para enviar datos por GSM y GPRS utilizando el módulo Sim 900, escogido para el sistema de rastreo, se analizó la comunicación básica que existe entre este módulo. Las comunicaciones básicas para interactuar con cada una de las acciones de los chips de SIMCOM es por medio de comandos AT. Los comandos AT son un conjunto de instrucciones llamadas Hayes, que es un lenguaje desarrollado por la compañía Hayes Communications, dichos comandos se volvieron un estándar en todos los modems de comunicación.

La interacción se realiza con una comunicación serial a través de los pines Tx y Rx del módulo; desde el microcontrolador se envían diversos comandos AT que el chip del módulo identifica y realiza las diversas acciones para las que está programado y devuelve una respuesta hacia el microcontrolador con el estado de la acción. Por ejemplo, al enviar hacia el módulo el comando AT, el módulo responderá con la palabra OK. Los comandos AT son idénticos en todos los modelos de chips SIMCOM en su nivel básico de operación, cada chip se diferencia con un grupo de comandos AT predefinidos para ciertas acciones adicionales que ofrecen, como ejemplo el chip SIM808 tiene un GPS interno y para controlar dicho módulo se necesitan de un grupo de comandos AT que el chip SIM900 no posee, porque no tiene un GPS interno. En el proyecto se tiene dos envíos de datos:

4.6.1. Envío de SMS por Comandos AT

En la Tabla 4.11 se observan los comandos AT utilizados para enviar un SMS a un teléfono celular desde el módulo sim900.

Tabla 4.11 Comandos AT para enviar un SMS

Comando	Acción
AT+CLIP=1	Activa la identificación de números telefónicos en llamadas o SMS
AT+CMGF=1	Configura el modo texto para recibir o enviar SMS
AT+CNMI=2,2,0,0,0	Muestra todos los SMS que reciba el SIM900
AT+CREG?	Comprueba la conexión a la red GSM
AT+CMGF=1	Selecciona el formato del SMS
AT+CMGS=XXXX	Envía un SMS al XXXX Numero

Fuente: Autor

En la programación de Arduino se utilizó cada uno de estos comandos para enviar los diferentes SMS al celular del propietario. Se utilizó el puerto serie de Arduino NANO para enviar los comandos hacia el sim900.

4.6.2. Envío de Datos a Servidor por GPRS

Para el envío de datos hacia el servidor se debe realizar una conexión cliente-servidor, entre el módulo sim900 del sistema de rastreo y el servidor con la Raspberry Pi3. Una vez hecha la conexión con el servidor se procede a enviar los datos. Los datos se envían primeramente a la Base de Datos MySQL para ser almacenados y luego visualizados en el Mapa de la aplicación de la página web.

El método que se utiliza para subir los datos hacia el servidor es el método GET, gracias a este se simplifica de gran manera el envío de datos, debido a que los datos se cargan en la base de datos por medio de la ejecución de un Link en el cual se colocan previamente los datos de latitud, longitud, velocidad y número de satélites. En la Tabla 4.12 se muestra los comandos AT para el envío de datos hacia un Servidor WEB

Tabla 4.12 Comandos AT para enviar Datos a Servidores

Comando	Acción
AT+CGDCONT=1,"IP","APN"	Define APN de Red Móvil
AT+CSTT="internet.movistar.com.ec", "movistar","movistar"	Define APN y contraseña de la red móvil
AT+CIICR	Se conecta a la red GPRS
AT+CIFSR	Obtiene IP de red móvil
AT+CIPSTART="TCP","rastreoivanchulde.ddns.net","80"	Se conecta al servidor
AT+CIPSEND	Activa el envío de datos al servidor
AT+CIPCLOSE	Cierra la Conexión TCP
AT+CIPSHUT	Desactiva la conexión con la APN

Fuente: Autor

Como ejemplo del método GET se puede decir que los datos son almacenados en el sistema con la ejecución del siguiente link:
rastreoivanchulde.ddns.net/datosviagprs.php?lat=XX&long=XX&speed=XX&satélites=XX

Donde cada XX representa los datos de latitud, longitud, velocidad y satélites. Estos datos faltantes son llenados por la placa Arduino NANO cada vez que se requiera conocer la ubicación del automóvil. Los comandos utilizados pueden encontrarse en el Anexo E.

4.7. Diagrama de Flujo del Sistema de Rastreo

El código de programación de la placa Arduino NANO se basa en el diagrama de flujo de la Figura 4.20.

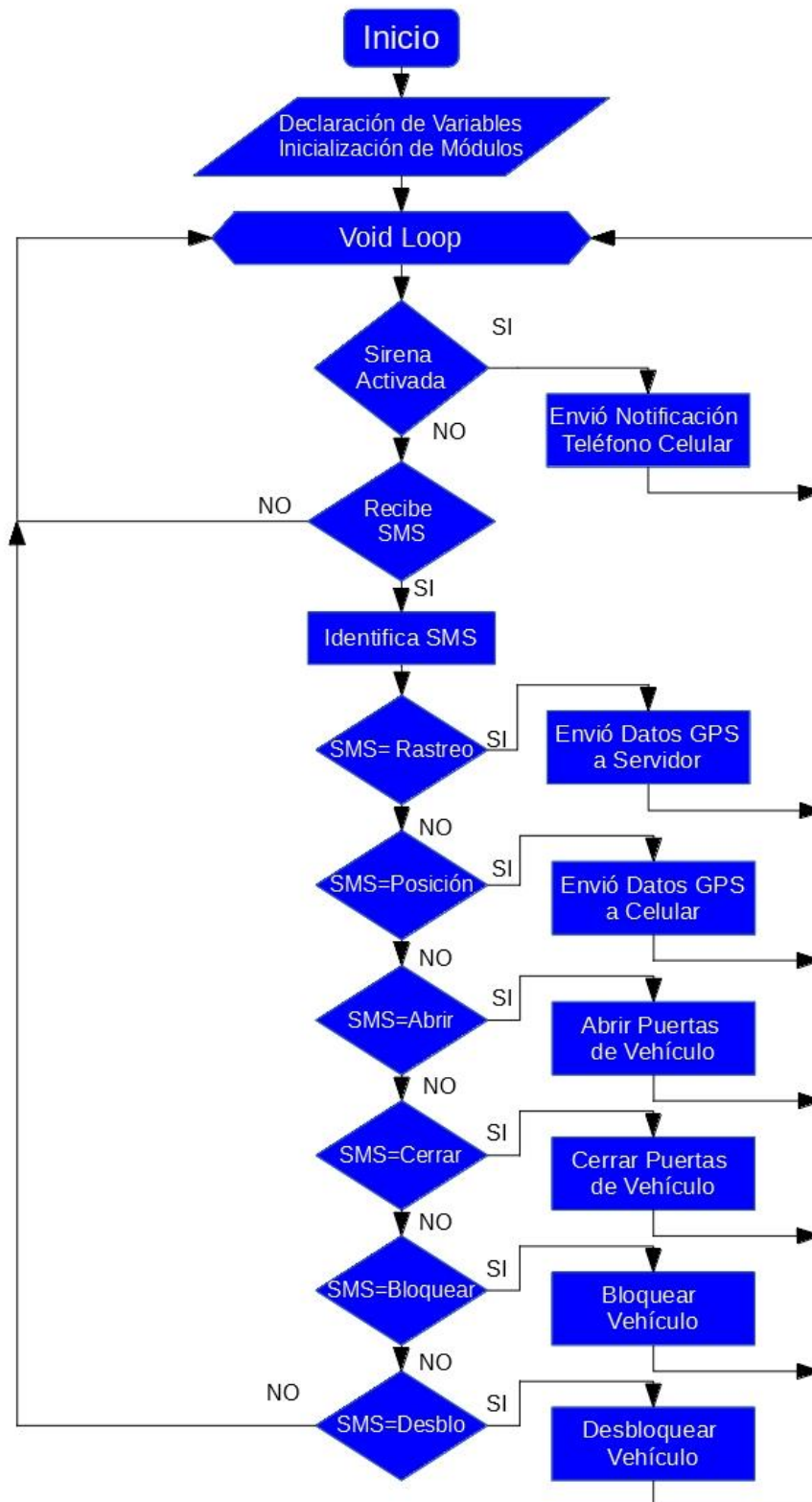


Figura 4.20 Diagrama de Programación de Sistema de Rastreo

Fuente: Autor

Al encenderse el dispositivo inicializa todos los módulos, activa las comunicaciones seriales y configura las entradas y salidas de la placa Arduino NANO. Después de inicializar todo el sistema, se accede a un ciclo infinito, donde va a ser monitoreada la señal de la sirena y los mensajes SMS que se reciban.

Si existe una activación de la sirena de a alarma del automóvil, Arduino NANO envía una notificación al celular del propietario informando que la alarma se activó. Si recibe un SMS, el sistema pasa a examinar dicho SMS y lo compara con los comandos que tiene programados en su interior si se asemeja a cualquier código activa la acción correspondiente y siempre después de cada acción vuelve al bucle infinito de monitoreo de alarma y mensajes SMS. Este es el diagrama de flujo en el cual se basó la programación de la placa Arduino NANO. La programación completa se encuentra en el Anexo A.

4.8. Implementación de Servidor

A continuación, se describen los pasos realizados para la creación del servidor web en la Raspberry Pi3. Los materiales que se necesitan son:

- Raspberry Pi2 o 3,
- MicroSD Card Clase 10 de 8Gb o mayor
- Cargador Micro USB de 2.0A Mínimo
- Mouse, Teclado y Cable HDMI
- Cable Ethernet

4.8.1. Instalación de Raspbian

Primeramente, se instaló el sistema operativo Raspbian en la placa. Se procede a descargarse de manera gratuita de la página oficial Raspbian en su última versión. Se obtiene un archivo con terminación .img. A continuación, se procede a usar el software Win32DiskImager, de uso libre, para crear la micro SD con el sistema operativo. Una vez creada la imagen se coloca dentro de la Raspberry Pi3 y se conectan el teclado, el mouse, el cable HDMI hacia un monitor y se procede a alimentar la placa con el cargador micro USB. El proceso de instalación de un sistema operativo se visualiza en la Figura 4.21

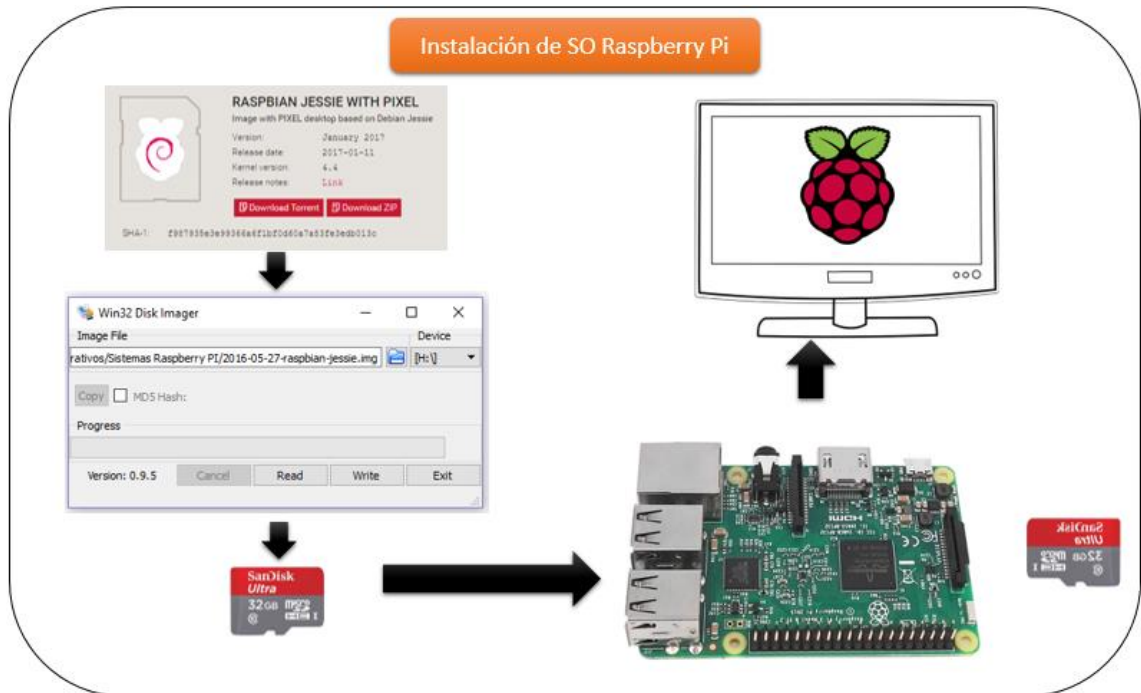


Figura 4.21 Instalación de Raspbian en Raspberry Pi3

Fuente: Autor

En el siguiente enlace <https://www.youtube.com/watch?v=-ZN3eEDaYcQ> [61] se detalla la instalación de Raspbian en una Raspberry Pi3. Una copia del video se encuentra alojado en un CD en la Biblioteca de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato.

4.8.2. Instalación de LAMP

A continuación, en la Tabla 4.13 se muestra un conjunto de comandos que se debe ingresar en la terminal de Raspbian para crear un servidor Web y un servidor FTP.

Tabla 4.13 Pasos para instalar LAMP y FTP en Raspbian.

Pasos	Comandos
1. Asignar IP Estática a Raspberry Pi:	sudo nano /etc/network/interfaces auto eth0 iface eth0 inet static address 192.168.0.2 netmask 255.255.255.0 gateway 192.168.0.1
2. Crear grupo www-data para servidor apache:	sudo groupadd www-data sudo usermod -a -G www-data www-data
3. Actualizar Repositorios de Raspbian:	sudo apt-get update sudo apt-get upgrade
4. Instalación de Apache y PHP	sudo apt-get install apache2 php5 libapache2-mod-php5 sudo /etc/init.d/apache2 restart
5. Activar interfaz de loopback:	sudo ifup lo
6. Instalar MySQL y PHPMyAdmin:	sudo apt-get install mysql-server mysql-client php5-mysql phpmyadmin
7. Editar el fichero php.ini:	sudo nano /etc/php5/apache2/php.ini Antes de la línea “Dynamics Extensions”: extension=mysql.so
8. En caso de no funcionar ejecutar los siguientes comandos	sudo ln -s /etc/phpmyadmin/apache.conf /etc/apache2/conf.d/phpmyadmin.conf sudo /etc/init.d/apache2 reload
9. Comprobar PHP creando un archivo con	sudo <?php phpinfo(); ?>nano /var/www/html/info.php
Instalar Servidor FTP	
10. Instalar FTP	sudo apt-get install vsftpd
11. Editar el Archivo vsftpd.conf	sudo nano /etc/vsftpd.conf local_enable=YES write_enable=YES
12. Reiniciar FTP	sudo service vsftpd restart

Fuente: Autor

Para utilizar el servicio de FTP se utilizó el programa FileZilla, que es totalmente gratuito, el cual crea una conexión FTP entre el servidor y el pc para subir los datos de

la página web creada a la dirección que almacena las paginas en APACHE la cual es /var/www/html/.

En el siguiente enlace <https://www.youtube.com/watch?v=-HCE2aWJ50&t=1s> [62] se detalla la instalación de LAMP en Raspbian. Una copia del video se encuentra alojado en un CD en la Biblioteca de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato.

4.8.3. Diseño de la Interfaz Web

Dentro del servidor web se encuentra alojado todos los archivos necesarios que conforman la página web de rastreo. Estos archivos son los que aparecen en la Figura 4.22

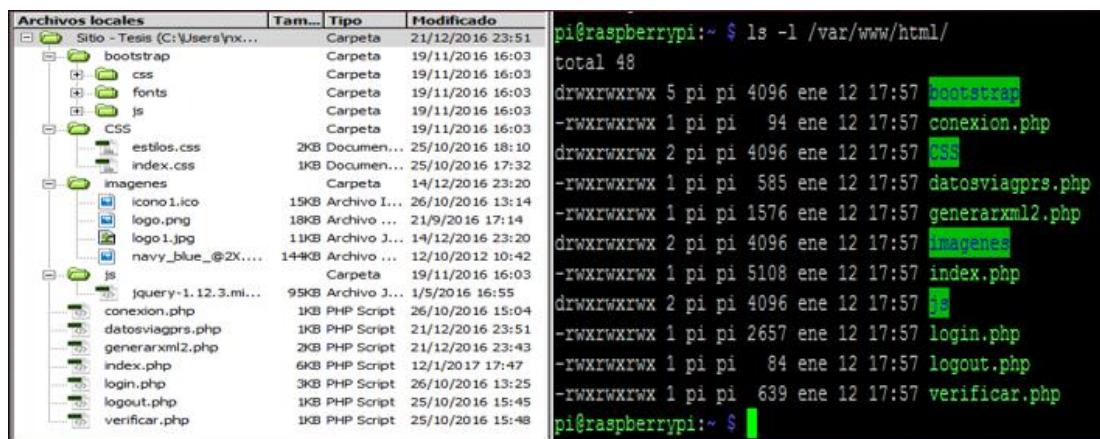


Figura 4.22 Archivos de Aplicación WEB

Fuente: Autor

A continuación, se explica la funcionalidad de cada uno de estos archivos. La programación de cada uno de estos se encuentra en el Anexo B.

Carpeta Bootstrap: Primeramente, Bootstrap es un framework desarrollado y liberado por Twitter que tiene como objetivo facilitar el diseño web. Permite crear de forma sencilla webs de diseño adaptable, es decir, que se ajusten a cualquier dispositivo y tamaño de pantalla y siempre se vean igual de bien. Dentro de esta carpeta se encuentra toda la configuración de Bootstrap para ser usada en la web del proyecto.

Carpeta CSS: Esta carpeta almacena todos los archivos de estilo .css de una página web, en esta se encuentra descrito el tamaño y tipo de letra, el fondo de pantalla, etc. CSS es considerado como los estilos de HTML y todos estos archivos están en esta carpeta, que serán llamados cuando se ejecute la página solicitada.

Carpeta Imágenes: Esta carpeta almacena todos los dibujos, imágenes, fondos de pantalla, etc. todos los archivos de imagen están guardados y son llamados cada vez que se requiera.

Carpeta JS: Dentro de esta carpeta se encuentra toda la programación en java script, esta programación puede ir dentro del mismo documento HTML o fuera en un archivo .js como en esta carpeta. En este proyecto se usó java script dentro de HTML por lo que esta carpeta solo almacena el paquete de programación JQUERY que es una librería que facilita la programación de java script de manera significativa.

conexion.php: Este archivo contiene los datos de conexión, usuarios, contraseñas, nombres de bases de datos, etc. de las diversas bases de datos que se utiliza en el proyecto.

datosviagprs.php: Script PHP que se utiliza para recibir los datos enviados por GPRS desde el sistema de rastreo; y los almacena en MySQL.

generarxml2.xml: Archivo de extensión .xml que se utiliza para extraer la información de la base de datos y pasarla a la página index.php para su procesamiento.

index.php: Es la página principal del sistema, que contiene la aplicación web del Mapa. Se utilizó una API de Google Maps para poder instalar un mapa de la empresa de google en la página web diseñada, la API es libre siempre y cuando no se trate de comercializar. En este archivo de extensión. php se realiza todo el proceso de la

información. En la Figura 4.23 se observa la interfaz principal de la página web.



Figura 4.23 Diseño de Pagina Web Principal.

Fuente: Autor

login.php: Este Script es la página de inicio de sesión de la aplicación web. Dentro de toda la página web se maneja el inicio de sesiones, lo cual si se intenta acceder a la página principal index.php sin haber iniciado sesión automáticamente se retorna siempre a la página de inicio de sesión. En la Figura 4.24 se observa el diseño de la página de inicio de sesión.



Figura 4.24 Página de Inicio de Sesión.

Fuente: Autor

logout.php: Script PHP que se ejecuta cuando se pulsa el botón salir de la página principal, termina la sesión actual y deja seguro el sistema.

verificar.php: Sirve para identificar el usuario y contraseña al momento de iniciar sesión, el usuario y contraseña están alojados en una base de datos, este script se conecta a la base de datos y analiza si los datos son correctos permitiendo el inicio de sesión.

4.9.Servidor en Internet

Normalmente con el paquete básico que ofrece CNT de internet, no es posible poner nuestro servidor en la web. CNT posee un filtrado NAT dentro de la red de distribución de internet, debido al escaso número de IPs que posee, es decir una IP Pública se comparte con diversas IP privadas dentro de la red de distribución. Un servidor necesita una IP Pública para salir a Internet que puede ser fija o variable.

En el paquete básico de internet de fibra óptica o de ADSL no posee acceso a la IP Pública debido al filtrado NAT. Algunos paquetes de CNT si otorgan el acceso a la IP Pública y existen dos tipos, IP Pública Fija o Variable. Se contrató un servicio de IP Pública variable que tiene un costo menor. Debido a esto la IP va a cambiar con el tiempo y no será la misma, para eso se utilizó un servicio DDNS de No-IP. [63] El cual otorga dominós DNS gratis o de pago y los actualiza a la IP variable que se contrató.

Dentro de la Raspberry Pi3 se ejecuta un Script de No-IP para actualizar periódicamente la IP Pública con la que se tenga acceso, de esta manera por medio del dominio se puede acceder al servidor por medio de internet desde cualquier parte del mundo.

Además de tener un dominio para la IP variable del Servidor se necesita habilitar los puertos de nuestro modem de acceso a internet. Los puertos que se utilizan para un servidor web son HTTP con el puerto 80, FTP con el puerto 21 y SSH con el puerto 22. Para el proyecto se utilizó el modem ADSL HG520C que ofrece CNT en sus planes de internet FastBoy, en la Figura 4.25 se muestra la configuración de puertos en el modem de CNT.

EchoLife HG520c

- ⊕ Status
- ⊖ Basic
 - ADSL Mode
 - WAN Setting
 - LAN Setting
 - DHCP
 - NAT
 - IP Route
 - Wireless Lan
 - ATM Traffic
- ⊕ Advanced
- ⊕ Tools

NAT - Virtual Server

NAT - Virtual Server

Virtual Server for	Single IP Account	
Rule Index	1 ▼	
Application	HTTP_Server	HTTP_Server ▼
Protocol	ALL ▼	
Start Port Number	80	
End Port Number	80	
Local IP Address	192.168.1.8	
Start Port(Local)	80	
End Port(Local)	80	

Virtual Server Listing

Rule	Application	Protocol	Start Port	End Port	Local IP Address	Start Port(Local)	End Port(Local)
1	HTTP_Server	ALL	80	80	192.168.1.8	80	80
2	FTP	ALL	21	21	192.168.1.8	21	21
3	SSH	ALL	22	22	192.168.1.8	22	22
4	-	-	0	0	0.0.0.0	0	0
5	-	-	0	0	0.0.0.0	0	0
6	-	-	0	0	0.0.0.0	0	0
7	-	-	0	0	0.0.0.0	0	0
8	-	-	0	0	0.0.0.0	0	0

Figura 4.25 Apertura de Puertos para Servidor en Modem HG520C de CNT

Fuente: Autor

4.10. Recepción de Datos GPRS

Los datos que son enviados desde el sistema de rastreo al servidor son en forma de peticiones GET, en forma de link HTTP. Dentro del servidor se encuentra un script que reconoce las peticiones GET hacia este y procede a almacenar estos datos en la Base de Datos MySQL. En la Figura 4.26 se visualiza el script utilizado para reconocer las peticiones GET.

```
datosviagprs.php
1 <?php
2 include("conexion.php");
3 //Conectar a base de datos MySQL
4 $con=mysql_connect($host,$user,$pw)or die("Problema al Conectar server");
5 mysql_select_db($db,$con)or die("Problema al conectar bd");
6 //Guarda los valores recibidos del link GET del sistema de rastreo
7 $latitud=$_GET["lat"];
8 $longitud=$_GET["long"];
9 $speed=$_GET["speed"];
10 $satelites=$_GET["satelites"];
11 $fecha=date("Y-m-d");
12 $hora=date("H:i:s");
13
14 //Coloca los Datos en la Base de Datos MySQL|
15
16 $sql="INSERT INTO `informacion`(`latitud`,`longitud`,`speed`,`satelites`,`fecha`,`hora`)
17 VALUES ('".$latitud."','".$longitud."','".$speed."','".$satelites."','".$fecha."','".$hora."')";
18 $result=mysql_query($sql);
19
20
21 ?>
22
```

Figura 4.26 Script para Recibir los Datos desde el Sistema de Rastreo.

Fuente: Autor

Cada vez que se ejecuta el link:

rastreoivanchulde.ddns.net/datosviagprs.php?lat=XX&long=XX&speed=XX&satelites=XX

desde el sistema de rastreo, se activa este script PHP en el servidor y lo primero que realiza es conectarse a la base de datos en MySQL, luego de la conexión procede a almacenar en variables locales los valores que vienen en los XX del link, después de esto, las variables son enviadas a la Base de Datos MySQL y guardadas para ser usadas por la aplicación Web.

De esta manera se reciben los datos de latitud, longitud, velocidad y satélites provenientes del sistema de rastreo y son almacenados en el servidor web.

4.11. Estructura Eléctrica del Automóvil

En la Figura 4.27 se observa la distribución eléctrica del automóvil, con sus respectivos dispositivos.

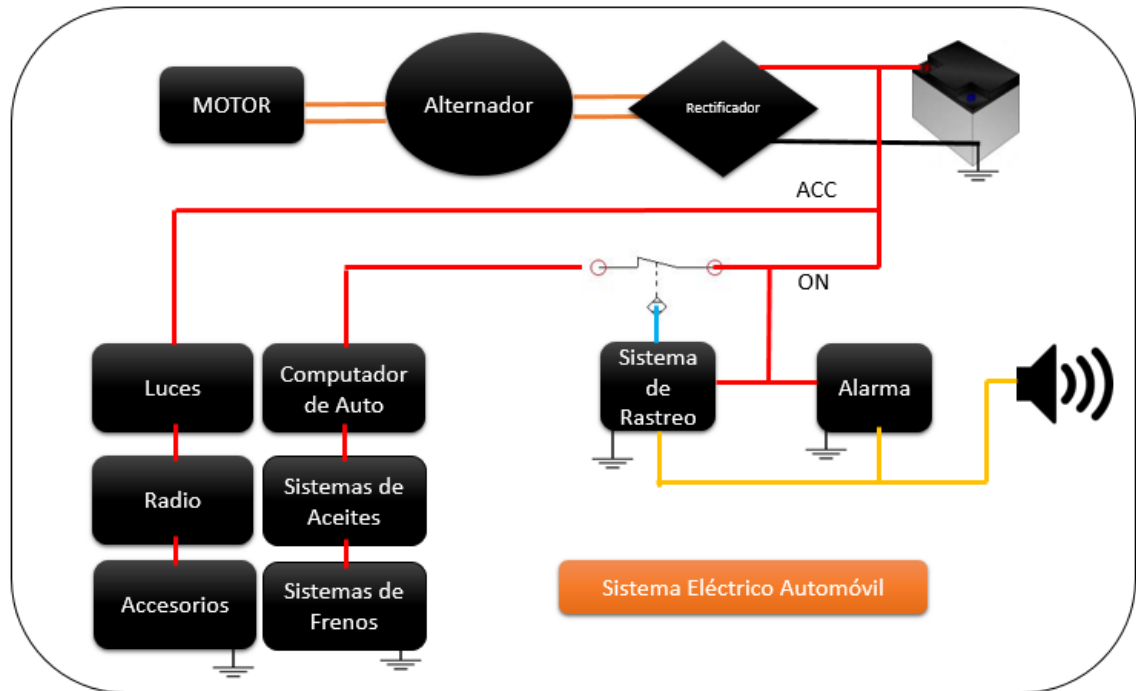


Figura 4.27 Sistema Eléctrico del Automóvil

Fuente: Autor

La energía del automóvil viene de dos fuentes, de su batería de 12V que suministra voltaje continuo; y la otra fuente que es el alternador del automóvil cuando esta encendido. Existen tres modos de suministrar el voltaje al automóvil dependiendo de la posición de la llave de giro ubicada en el volante.

La primera es ACC que suministra corriente de batería a los dispositivos de señalización del automóvil como por ejemplo las luces de parqueo, direccionales, y luces bajas; además suministra tensión a los diferentes accesorios del automóvil como lo son la radio, salidas de cargadores de 12V, Calefacción, etc.

La segunda posición es ON en la cual funciona completamente el automóvil siguiendo la condición de apagado o encendido. Si la llave del automóvil se ubica en ON antes de encender el automóvil, todos los sistemas de seguridad y sistemas de control del automóvil se activan, pero consumen el voltaje de la batería de 12V, como, por ejemplo, la computadora del automóvil, el sistema de aceites, el sistema de frenos, etc. Si la llave del automóvil se ubica en ON después de haber sido encendido el automóvil, se activan de la misma manera todos los sistemas del automóvil, pero esta vez se consume el voltaje suministrado por el alternador. Normalmente un alternador de automóvil convierte la energía mecánica en eléctrica, pero en alterna, es mucho más

eficiente producir voltaje alterno que continuo. Para transformar de alterno a continuo se utiliza un rectificador, de esta manera la batería no suministrara la energía cuando el automóvil esta encendido.

La tercera posición de la llave es ignición, sirve para encender el automóvil, utilizando un motor de arranque compuesto de un motor eléctrico que da el impulso al motor mecánico del automóvil para que empieza marcha. Este modo solo se activa por periodos cortos de tiempo y solo al poner en marcha el motor. Para obtener los cables correctos para el sistema de rastreo se utilizó un multímetro, que mida 12V en la sirena cuando este activada, y 12V adicionales de la fuente de alimentación de la alarma ya instalada. De esta manera se acoplo el sistema de rastreo al automóvil para su funcionamiento. Se reconoció los cables de alimentación de 12V, el cable de alimentación de la sirena y el cable de la posición ON de la llave de ignición para su ingreso en el automóvil. En la Figura 4.28 Se observa el cableado del automóvil.

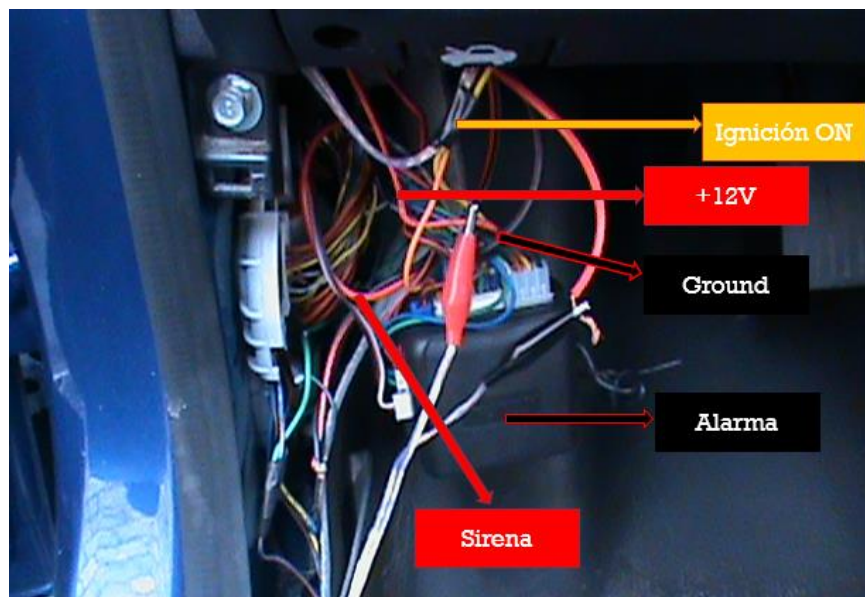


Figura 4.28 Cableado del Automóvil

Fuente: Autor

4.12. Diagramas del Circuito y PCB

El diseño de los circuitos y su PCB se realizaron en el software EAGLE, tomando en cuenta los cálculos de los componentes electrónicos a utilizarse y el tamaño de cada uno de los módulos. En la Figura 4.29, 4.30, 4.31 y 4.32 se puede observar el esquemático completo del circuito del sistema en el automovil, su PCB para fabricarse, la baquelita resultante y el sistema construido.

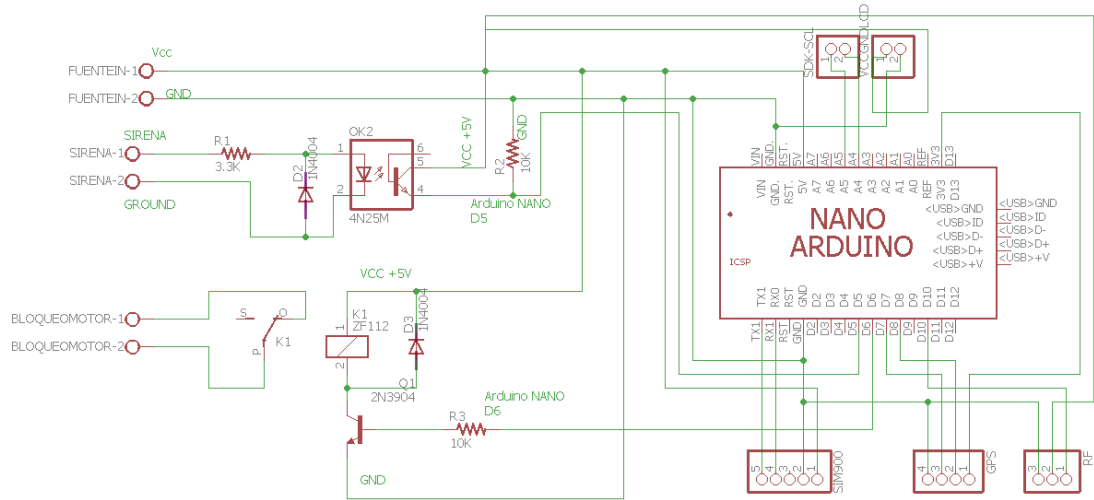


Figura 4.29 Esquemático del Sistema de Rastreo

Fuente: Autor

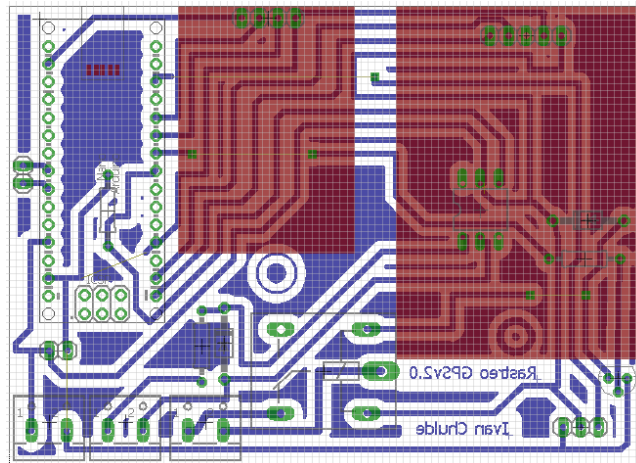


Figura 4.30 PCB del Circuito de Rastreo Satelital

Fuente: Autor

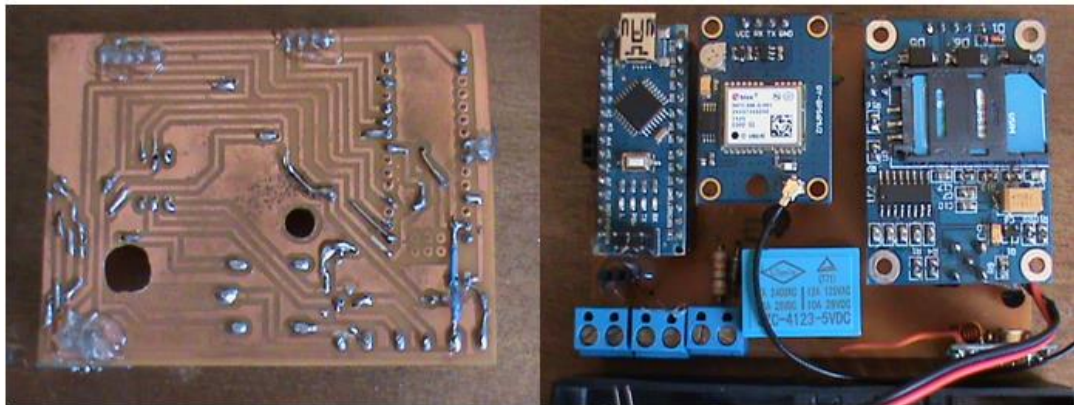


Figura 4.31 Baquelita de Prototipo

Fuente: Autor

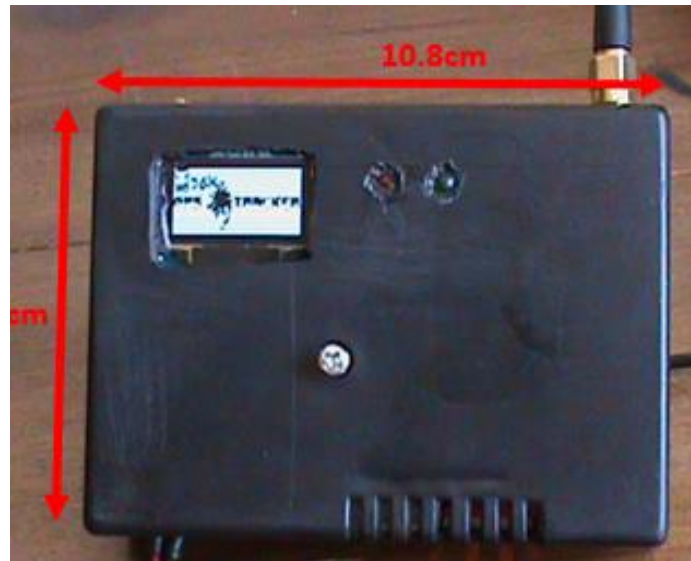


Figura 4.32 Equipo de Sistema de Rastreo

Fuente: Autor

4.13. Pruebas de Funcionamiento

Para presentación del sistema en el proyecto se lo coloco debajo del volante del conductor. En la Figura 4.33 se aprecia el sistema de rastreo instalado en el automóvil.



Figura 4.33 Sistema de Rastreo Instalado en Automóvil

Fuente: Autor

El sistema posee una antena GPS y una antena GSM para conectarse a la red satelital y la red móvil de datos, esta antena debe estar siempre descubierta y en posición hacia el cielo. La ubicación de esta antena puede ser interna o externa, dependiendo de las

necesidades del usuario. Para este proyecto se colocó la antena dentro del automóvil encima del tablero de controles para obtener mayor señal, debido a que la señal de GPS es débil en comparación con la señal de GSM. En la Figura 4.34 se aprecia la antena GPS/GPRS utilizada en el proyecto.



Figura 4.34 Antena GPS ubicada en Automóvil

Fuente: Autor

Al encender el sistema de rastreo, se comienzan a cargar los diferentes módulos que contiene e indica en su pantalla el logo del prototipo, cuando el sistema está listo para operar refleja en su pantalla “Sistema Listo Esperando Orden” En la Figura 4.35 se aprecia estas dos etapas.



Figura 4.35 Encendido del Sistema

Fuente: Autor

Una vez encendido el sistema completamente se puede enviar los comandos desde el teléfono celular de propietario para realizar las diversas operaciones. En las Figuras

4.36, 4.37 y 4.38 se muestran los estados de pantalla de cada una de las acciones recibidas mediante mensajes de texto.



Figura 4.36 Bloqueo y Desbloqueo de Motor

Fuente: Autor



Figura 4.37 Cierre y Apertura de Puertas

Fuente: Autor



Figura 4.38 Envío de Datos a Servidor y Celular

Fuente: Autor

El sistema de rastreo también notifica vía mensaje de texto cuando la alarma del automóvil se activa, es decir cuando la sirena empieza a sonar; en la Figura 4.39 se aprecia el envío de las notificaciones de la alarma vía mensaje de texto al propietario cuando se activa la alarma del automóvil.



Figura 4.39 Envío de Notificaciones de Alarma

Fuente: Autor

Consumo de Corriente

Con la ayuda de una pinza amperimétrica se obtuvo el consumo máximo de corriente que tiene el sistema, siendo de 163mA, ilustrado en la Figura 4.40



Figura 4.40 Medición del sistema con Pinza Amperimétrica

Fuente: Autor

Utilizando las baterías de respaldo, se obtuvo un funcionamiento ininterrumpido de alrededor de 24 horas. De esta manera, en caso de ser desactivada la batería del automóvil, se tiene 1 día de duración de batería para realizar cualquier acción sobre esta.

Comandos “abrir”, “cerrar”, “bloqueo” y “desbloqueo”

Con la ayuda de un cronometro se tomó muestras de tiempo de cada una de las acciones que realiza el sistema. En la tabla 4.14 Se visualiza los tiempos obtenidos de cada acción del sistema.

Tabla 4.14 Tiempos al Ejecutar los comandos en el Sistema.

Comandos	Muestra 1	Muestra 2	Muestra 3	Muestra 4	Muestra 5
Apertura de Puertas	4,4s	6,7s	6,5s	4,9s	5,8s
Cierre de Puertas	6,6s	5,7s	4,9s	5,2s	5,7s
Bloqueo	6,3s	4,4s	4,7s	4,3s	4,0s
Desbloqueo	8,4s	9,1s	7,5s	7,6s	8,0s

Fuente: Autor

Los tiempos varían, debido a la red de telefonía móvil de movistar, dependiendo de la señal que esta operadora tenga en el lugar.

Comando “rastreo”

Para el analisis de los tiempos en el comando “rastreo” se tomo como referencia los tiempos en los cuales fueron registrados los datos en el servidor. Estos valores se visualizan en la figura 4.41

-1.266439	-78.624908	0	5	2017-04-10	11:56:31
-1.266439	-78.624908	0	5	2017-04-10	11:56:44
-1.266435	-78.624908	0	5	2017-04-10	11:56:56
-1.266438	-78.624908	0	5	2017-04-10	11:57:07
-1.266441	-78.624908	0	5	2017-04-10	11:57:20
-1.266441	-78.624908	0	5	2017-04-10	11:57:32
-1.267066	-78.637794	22	5	2017-03-31	14:51:01
-1.267855	-78.637817	0	5	2017-03-31	14:51:19
-1.268245	-78.637825	0	5	2017-03-31	14:51:35
-1.269348	-78.637878	37	5	2017-03-31	14:51:48
-1.269680	-78.636162	25	5	2017-03-31	14:52:46
-1.269026	-78.635223	39	5	2017-03-31	14:52:59

Figura 4.41 Tiempos de Registro de cada muestra en Servidor Web

Fuente: Autor

De acuerdo a las muestras registradas en la base de datos de MySQL del servidor web se tiene una diferencia de tiempos en la tabla 4.15

Tabla 4.15 Tiempos Registrado en Servidor

	Muestra 1	Muestra 2	Muestra 3	Muestra 4	Muestra 5
Hora de Registro en el Servidor	11:56:31	11:56:44	11:56:56	11:57:07	11:57:20
Segundos por Muestra	0:00:13	0:00:12	0:00:11	0:00:13	0:00:12
Hora de Registro en el Servidor	14:51:01	14:51:19	14:51:35	14:51:48	14:52:46
Segundos por Muestra	0:00:18	0:00:16	0:00:13	0:00:58	0:00:13

Fuente: Autor

De esta manera se comprueba que las muestras son registradas en el servidor cada 13 segundos, constantemente cuando existe señal GPRS. Si no existe señal o un fallo en conexión al servidor web, esto puede variar dependiendo de la red móvil de Movistar.

Comando “posiscion”

Para el comando “posicion”, el cual envía las coordenadas de ubicación al teléfono del propietario, se tomaron los tiempos que se tarda en reconocer, procesar y volver enviar los datos, el tiempo que se demora en recibir el mensaje en el teléfono y la demora que existe por la operadora en enviar el mensaje por su red. Los tiempos obtenidos se visualizan en la Tabla 4.16

Tabla 4.16 Tiempos medidos para el comando “posición”

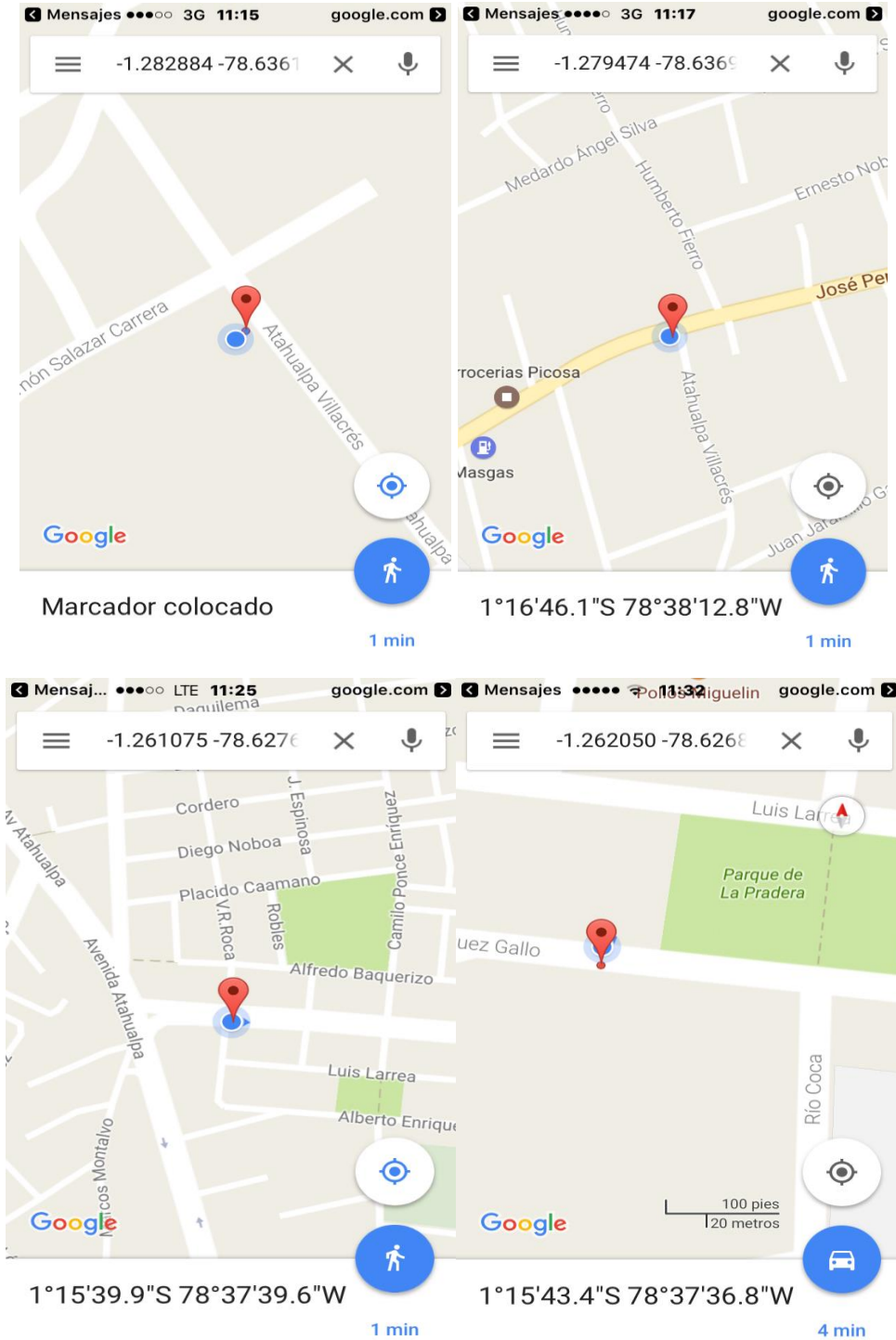
Comando: "posicion"	Muestra 1	Muestra 2	Muestra 3	Muestra 4	Muestra 5
Reconocer, Proceso y Enviar	3,9s	4,0s	4,4s	5,1s	4,9s
Mensaje Recibido en Teléfono	10,8s	11,3s	13,4s	12,2s	12,4s
Demora del Operador Móvil	6,8s	7,3s	9,0s	7,1s	7,5s

Autor

Con relación a los datos obtenidos se observa que los tiempos que tarda la operadora móvil en enviar el mensaje con los datos de ubicación son altos siendo el máximo 9 segundos; estos tiempos no se pueden controlar, dependen de la señal de la red móvil de Movistar. Los tiempos que se demora en reconocer el mensaje, procesarlo y enviar los datos, es de máximo 5 segundos, así mismo en la recepción del mensaje de texto depende de la red móvil.

Para comprobar la precisión en la ubicación del sistema, se lo comparó con el sistema de posicionamiento de un teléfono inteligente, en este caso un Iphone SE. En las Figuras 4.42 se visualizan las pruebas de precisión del sistema de rastreo donde el punto rojo es la ubicación obtenida por el sistema en el automóvil y el punto de color azul es la ubicación obtenida por el Iphone SE.

Figura 4.42 Comparación de Ubicación del Sistema



Fuente: Autor

Servidor Web

La Raspberry Pi3 se encuentra ubicada en el hogar del propietario cerca del modem de acceso a internet y a una fuente de alimentación de 5V. Para el proyecto se utilizó un Access Point TP.Link que extiende la red del Modem de CNT para más capacidad. En la Figura 4.43 se visualiza la ubicación de la Raspberry Pi3 en el hogar junto con el Access Point que tiene acceso a internet por medio del Modem de CNT.

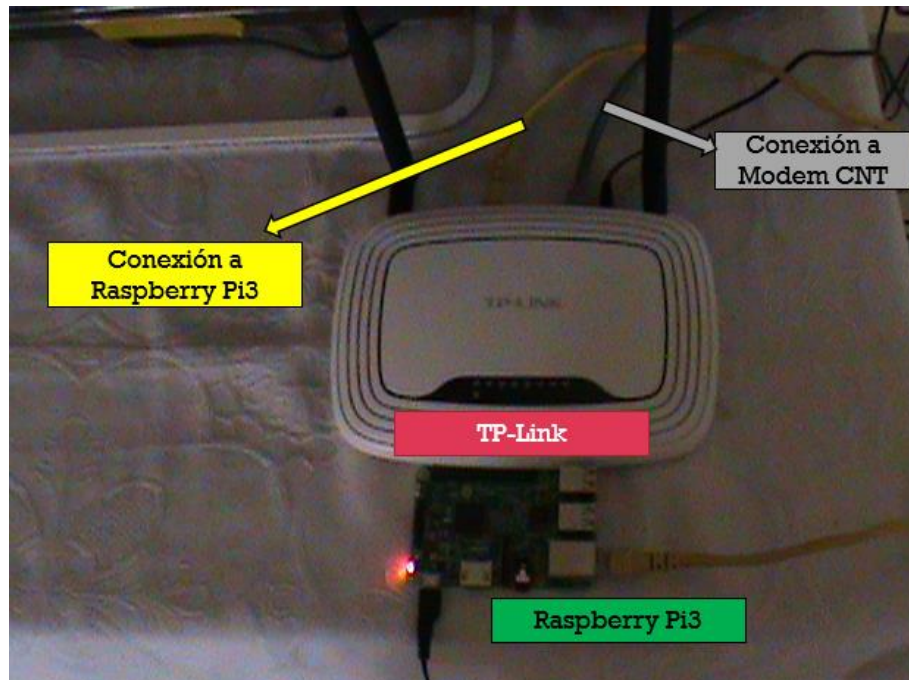


Figura 4.43 Raspberry Pi3 con Acceso a Internet

Fuente: Autor

Todas las acciones del sistema de rastreo se cumplieron satisfactoriamente, en el siguiente enlace <https://www.youtube.com/watch?v=eHusu8SmPAE> [64] se encuentra un video más detallado de las pruebas de funcionamiento. Una copia del video se encuentra alojado en un CD en la Biblioteca de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial de la Universidad Técnica de Ambato.

4.14. Presupuesto

En este apartado se detalla el presupuesto total del sistema propuesto, así como, se analiza el precio de los servicios que ofrecen las diversas empresas de monitoreo web para la localización de un automóvil existentes en el mercado.

4.14.1. Costo del Proyecto.

En la Tabla 4.17, se detalla el costo total del proyecto.

Tabla 4.17 Presupuesto del Sistema.

Presupuesto Total			
Item	Cantidad	Valor Unitario	Valor Total
Raspberry Pi3	1	\$ 70,00	\$ 70,00
Cable de Alimentación de 2A	1	\$ 4,00	\$ 4,00
Tarjeta Micro SD 16GB clase 10	1	\$ 9,00	\$ 9,00
Cable Ethernet 3M	1	\$ 5,00	\$ 5,00
Caja para Raspberry Pi3	1	\$ 6,00	\$ 6,00
Arduino NANO	1	\$ 8,00	\$ 8,00
Módulo GPS	1	\$ 25,00	\$ 25,00
Módulo GSM/GPRS	1	\$ 50,00	\$ 50,00
Módulo RF	1	\$ 4,00	\$ 4,00
Antena GPS/GSM	1	\$ 15,00	\$ 15,00
Módulo XI4005	1	\$ 5,00	\$ 5,00
Módulo XL6009	1	\$ 5,00	\$ 5,00
Módulo TP-4056	1	\$ 3,00	\$ 3,00
Baterías 18650	2	\$ 6,00	\$ 12,00
Porta Baterías	1	\$ 1,00	\$ 1,00
Caja para sistema de rastreo	1	\$ 3,50	\$ 3,50
Pantalla OLED 0,96"	1	\$ 15,00	\$ 15,00
Conector SMA a uFL	1	\$ 4,86	\$ 4,86
Leds	2	\$ 0,50	\$ 1,00
Resistencias(3.3K , 10K, 10K)	3	\$ 0,02	\$ 0,06
Relé 5V	1	\$ 0,50	\$ 0,50
Baquelita	1	\$ 5,00	\$ 5,00
Cloruro Férrico	2	\$ 0,75	\$ 1,50
Estaño	1	\$ 2,00	\$ 2,00
4N25	1	\$ 0,40	\$ 0,40
2N3904	1	\$ 0,10	\$ 0,10
Borneras	3	\$ 0,50	\$ 1,50
Cables 5m	1	\$ 3,00	\$ 3,00
Espagueti Térmico	1	\$ 0,50	\$ 0,50
Espadines	1	\$ 3,00	\$ 3,00
Total			\$ 263,92

Fuente: Autor

El costo del prototipo fue de \$263,92 este costo es por una unidad de todo el sistema de monitoreo web que incluye el sistema de rastreo para el automóvil y el servidor de

rastreo web, puede servir como referencia para un usuario que necesite replicar el sistema haciéndolo por sí mismo, obviando los costos de instalación, mano de obra y otros gastos extras que los cubre la misma persona.

En caso de comercializar el producto se tomó un 30% del valor del mismo para cubrir costos de utilidad, además de costos de instalación de cada uno de los servicios. Se tomó como costo de instalación el valor que cobran los diversos comerciales de autolujos en la ciudad de Ambato para instalar cualquier equipo electrónico en un automóvil.

A continuación, en la Tabla 4.18 se detalla los costos de comercialización para el producto.

Tabla 4.18 Costo de Comercialización

Costo de Fabricación	
Instalación en Automóvil	\$ 30,00
Instalación de Servidor	\$ 30,00
Utilidad del 30% del Producto	\$ 79,18
Costo de Materiales	\$ 263,92
Total	\$ 403,10

Fuente: Autor

De esta manera se obtiene un costo de \$403,10 para comercializar el producto.

4.14.2. Empresas de Servicios de Monitoreo Web.

A continuación, se describen algunas empresas que ofrecen los servicios de monitoreo web para un automóvil, los costos fueron obtenidos de las respectivas páginas web o su Call Center, cotizando los valores para un automóvil Hyundai i10 del año 2013.

En la Tabla 4.19 se describen los servicios que brinda la empresa Hunter.

Tabla 4.19 Servicios de Hunter

Servicio	Hunter Lojack	Hunter Full Plan Básico
Apertura y Cierre de Puertas vía SMS		
Bloqueo y Desbloqueo de Motor vía SMS		
Posición actual del Vehículo		X
Rastreo continuo GPS/GPRS		X
Monitoreo Web de Posicionamiento		X
Rastreo Lojack	X	X
Costo	\$473 + iva	\$850 + iva
Renovación Anual	\$319 + iva	\$384 + iva

Fuente: Autor, basado en [65].

En la Tabla 4.20 se describe el servicio de Chevystar de la empresa CHEVROLET dedicado únicamente a esta marca de automóviles.

Tabla 4.20 Características de Servicios Chevystar

Servicio	Chevystar
Apertura y Cierre de Puertas vía SMS	X
Bloqueo y Desbloqueo de Motor vía SMS	X
Posición actual del Vehículo	X
Rastreo continuo GPS/GPRS	X
Monitoreo Web de Posicionamiento	X
Costo	\$711,48
Renovación Anual	\$300

Fuente: Autor, basado en [66]

En la Tabla 4.21 se visualiza los servicios que ofrece TRACLINK para un automóvil.

Tabla 4.21 Características de Servicios de Tracklink

Servicio	Interactivo	Control
Apertura y Cierre de Puertas vía SMS		X
Bloqueo y Desbloqueo de Motor vía SMS		X
Posición actual del Vehículo	X	X
Rastreo GPS/GPRS	X	X
Monitoreo Web de Posicionamiento		X
Costo	\$511,86	\$660,60
Renovación Anual	\$250	\$320

Fuente: Autor, basado en [67]

En la Tabla 4.22 se describen los servicios de Traigo GPS para un automóvil.

Tabla 4.22 Soluciones de TraigoGPS

Servicio	Servicio Rastreo Satelital	Servicio Premium de Monitoreo
Apertura y Cierre de Puertas vía SMS	X	X
Bloqueo y Desbloqueo de Motor vía SMS	X	X
Posición del Vehículo vía SMS	X	X
Rastreo GPS/GPRS	X	X
Monitoreo Web		X
Costo	\$420	\$603
Renovación Anual	-----	\$240

Fuente: Autor, basado en [68]

A continuación, en la Tabla 4.23 se compara el costo del prototipo con los sistemas analizados en el apartado 4.2. Los costos de todos los sistemas analizados fueron cotizados para un automóvil Hyundai i10 del año 2013 y para uso personal o privado, mas no un servicio empresarial. Además, todos los servicios cuentan con Sistema de Rastreo por medio de página web, por mensajes de textos y aplicaron móvil; apertura y cierre de puertas remotamente y bloqueo del motor del automóvil. Algunos sistemas ofrecen servicios extras, pero en general las características de los sistemas de monitoreo que se han comprado se ajustan al diseño del prototipo desarrollado. Los costos de instalación y de mantenimiento de las empresas analizadas es gratis.

Tabla 4.23 Comparación de Sistemas

Característica	Prototipo	Hunter	ChevyStar	TrackLink	TRAIGO GPS
Servicio	Sistema de Resteo + Servidor Web	Hunter Full	Chevystar	Control	Servicio Premium de Monitoreo
Costo del sistema	\$403	\$850	\$711,48	\$660,60	\$603
Instalación	0\$	\$0	\$0	\$0	\$0
Mantenimiento		\$0	\$0	\$0	\$0
Renovación Anual	\$0	\$384	\$300	\$320	\$240
Total 1er año	\$403	\$850	\$711,48	\$660,60	\$603
Total en 2do año	\$403	\$1234	\$1011,48	\$980,60	\$843
Total en 3er año	\$403	\$1618	\$1311,48	\$1300,60	\$1083

Fuente: Autor

De esta manera se observa que el prototipo desarrollado es una alternativa viable frente al mercado actual de monitoreo y rastreo GPS de un automóvil, teniendo las mismas funcionalidades, con la posibilidad de ser mejorado por el usuario para sus necesidades, a un costo reducido, gracias al uso de hardware y software libre.

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones

- En el análisis de precisión de posicionamiento del módulo GPS utilizado en el sistema, se estableció que posee un margen de variación de 1 a 3m con respecto a al GPS de un teléfono móvil, siendo este un margen mínimo que garantiza la correcta localización del vehículo.
- En el análisis de tiempos de respuesta de los comandos “abrir”, “cerrar”, “bloquear” y “desbloquear” enviados desde el teléfono móvil mediante mensajes de texto GSM, se determinó que tienen un rango de respuesta de 4.3 a 8 segundos, los cuales establecen la rapidez y eficacia de estas aplicaciones en el sistema.
- Se determinó, que mediante el uso de componentes de hardware y software libre, el sistema de monitoreo posee las ventajas de ser personalizable, optimizable, económico, con un costo de \$403 sin cargos de renovación, el cual es accesible para los estratos sociales, medio y medio bajo.
- En las pruebas de funcionamiento de posición y rastreo, del sistema se determinó que las coordenadas geográficas de latitud y longitud se obtienen en un lapso de tiempo de 10 a 14 segundos para el comando posición; y las muestras de localización se almacenan en el servidor web en un intervalo de tiempo de 13 a 58 segundos, estos tiempos dependen tanto de la conexión al satélite GPS, como de la conexión a la red móvil GSM/GPRS.

5.2. Recomendaciones

- Para mejorar el funcionamiento del sistema de rastreo, se debe utilizar un número reducido de librerías de programación y optimizar el uso de variables en el programa, debido a que el procesamiento del microcontrolador se vuelve inestable con un valor igual o mayor al 75% de memoria dinámica ocupada.

- Los dispositivos electrónicos adecuados para el sistema de rastreo deben soportar una temperatura de trabajo entre de 4,6°C a 56,5°C, para asegurar la estabilidad del sistema.
- Se recomienda el uso del presente sistema de monitoreo en futuros proyectos investigativos gracias a que fue desarrollado en código abierto y diseñado con hardware libre, se tiene la libertad de agregarle nuevas funcionalidades según los intereses del investigador.
- Se recomienda el uso de módulos reductores y elevadores de tensión XL40005 y XL6009 en lugar de los reguladores de voltaje 7805, 7809 o 7812, con el fin de mejorar la transferencia energética, evitando pérdidas por disipación de calor y otorgándole al sistema fuentes de alimentación más eficientes.

BIBLIOGRAFÍA

- [1] M. Alpizar, «La Opinión - "Julio, el mes con más autos robados en California",» 10 Julio 2015.
- [2] «Encuesta de Victimización y Percepción de Inseguridad,» INEC-Ecuador, 2011.
- [3] «4,8 autos por día robaron en Quito en 2014,» *El Universo*, 19 Enero 2015.
- [4] INEC, «Instituto Nacional de Estadística y Censo,» [En línea]. Available: <http://www.ecuadorencifras.gob.ec/encuesta-de-estratificacion-del-nivel-socioeconomico/>. [Último acceso: 22 Noviembre 2016].
- [5] INEC, «Instituto Nacional de Estadística y Censos,» 2011-2012. [En línea]. Available: <http://www.ecuadorencifras.gob.ec/inec-presenta-resultados-de-la-encuesta-de-ingresos-y-gastos/>. [Último acceso: 22 Noviembre 2016].
- [6] J. D. Reyes Huertas y M. F. Huertas Vitery, «Diseño e Implementación de un Módulo de Alarma para el Monitoreo y Control del Vehículo a través del Sistema GSM y GPS,» Repositorio - ESPE, Latacunga, 2014.
- [7] J. G. Acosta Calderón, «"SISTEMA DE CONTROL Y MONITOREO VEHICULAR UTILIZANDO TECNOLOGÍA RFID Y ENVÍO DE ALERTAS MEDIANTE MENSAJES DE TEXTO.",» Repositorio Universidad Técnica de Ambato, Ambato, 2015.
- [8] O. L. Matza Chuncha, «"SISTEMA ELECTRÓNICO DE CONTROL DE VELOCIDAD DE AUTOBUSES, PARA LA COOPERATIVA DE TRANSPORTES SANTA",» Repositorio Universidad Técnica de Ambato, Ambato, 2014.
- [9] E. P. Rodríguez Martínez, «"SISTEMA DE LOCALIZACIÓN AUTOMÁTICA DE VEHÍCULOS MEDIANTE GPS, PARA DETERMINAR LA UBICACIÓN DE LAS VOLQUETAS DEL GAD MUNICIPAL DE PATATE",» Repositorio Universidad Técnica de Ambato, Ambato, 2014.
- [10] V. C. Aguirre Salas y D. M. Zambrano Rosado, «Utilización de módulos GPS en combinación con módulos de radio frecuencia para la transmisión de datos de posicionamiento hacia control central empleando microcontroladores,» Repositorio Escuela Superior Politécnica del Litoral, Guayaquil, 2010.
- [11] M. L. Chuquitarco Chuquitarco y R. J. Naranjo Santiana, «Diseño e Instalación de un sistema de rastreo satelital mediante GPS y GPRS para el vehículo Chevrolet-Aveo de la escuela de conducción de ESPE-LATACUNGA,» Repositorio ESPE, Latacunga, 2012.
- [12] T. M. Ortiz Correa, «Aplicación WEB para rastreo Satelital Vehicular,» Repositorio Universidad del Azuay, Cuenca, 2015.

- [13] K. R. Anguizaca Fuentes, A. M. Remache López, A. G. Rodríguez Miranda y S. G. Santana Pincay, «Comunicación Satelital Utilizando el Sistema Globalstar en el Ecuador,» Repositorio Escuela Superior Politécnica del Litoral, Guayaquil, 2000.
- [14] T. Wayne, «Comunicaciones Satelitales,» de *Sistemas de Comunicaciones Electrónicas*, México, Pearson Educación, 2003, pp. 793-832.
- [15] Universidad-Privada-Dr.-Rafael-Belloso-Chacín, «Fundamentos de Comunicación Satelital,» [En línea]. Available: <http://www.urbe.edu/info-consultas/web-profesor/12697883/articulos/Comunicaciones%20Satelites%20y%20Celulares/Fundamentos%20de%20la%20Comunicacion%20Satelital.pdf>. [Último acceso: 23 Noviembre 2015].
- [16] UBLOX, «GPS Essentials of Satellite Navigation,» 2009.
- [17] «Altitud de Satelites Sobre la Tierra,» [En línea]. Available: https://upload.wikimedia.org/wikipedia/commons/b/b4/Comparison_satellite_navigation_orbits.svg.
- [18] GPS, «GPS.gov,» U.S government, [En línea]. Available: <http://www.gps.gov/systems/gps/spanish.php>. [Último acceso: 21 11 2016].
- [19] GPS.gov, «Satélites de la constelacion GPS,» [En línea]. Available: <http://www.gps.gov/systems/gps/space/>.
- [20] GPS.gov, «Segmento de Control de GPS,» [En línea]. Available: <http://www.gps.gov/multimedia/images/GPS-control-segment-map.pdf>.
- [21] GLONASS, «INFORMATION AND ANALYSIS CENTER FOR POSITIONING, NAVIGATION AND TIMING,» [En línea]. Available: <https://www.glonass-iac.ru/en/>.
- [22] IAC, «INFORMATION AND ANALYSIS CENTER FOR POSITIONING, NAVIGATION AND TIMING,» GLONASS, [En línea]. Available: <https://www.glonass-iac.ru/en/guide/>. [Último acceso: 17 11 2016].
- [23] BEIDOU, «BeiDou Navigation Satellite System,» CSNC, [En línea]. Available: <http://en.beidou.gov.cn/index.html>. [Último acceso: 22 11 2016].
- [24] BeiDou, «BeiDou Navigation System Signal In Space Interface Control Document,» Noviembre 2016. [En línea]. Available: <http://www.beidou.gov.cn/attach/2016/11/07/21212.pdf>. [Último acceso: 22 Noviembre 2016].
- [25] ESA, «Agencia Espacial Europea,» ESA, [En línea]. Available: http://www.esa.int/Our_Activities/Navigation/Galileo/What_is_Galileo. [Último acceso: 22 Noviembre 2016].

- [26] E. Mauricio y M. Inga.Ortega, «La Telefonía móvil de cuarta Generación 4G Long Term Evolution,» [En línea]. Available: <http://ingenius.ups.edu.ec/>. [Último acceso: 22 Noviembre 2015].
- [27] J. M. Huidobro Moya y R. C. Pastor, *Sistemas de Telefonía*, Magallanes: Thomson - Parainfo, 2006.
- [28] GRADIANT, «Centro Tecnológico de Telecomunicaciones de Galicia "LTE: la evolución de la red móvil de acceso,» [En línea]. Available: <http://www.gradiant.org/es/>. [Último acceso: 23 Noviembre 2016].
- [29] T. Wayne, «Sistema Global Para Comunicaciones Móviles,» de *Sistemas de Comunicaicones Electronicas*, Mexico, Pearson Educación, 2003, pp. 888-890.
- [30] 3GPP, «3GPP,» [En línea]. Available: <http://www.3gpp.org/technologies/keywords-acronyms/102-gprs-edge>. [Último acceso: 23 11 2016].
- [31] GNU, «El Sistema Opertivo GNU,» Free Software Foundation, [En línea]. Available: <https://www.gnu.org/>. [Último acceso: 23 Noviembre 2016].
- [32] A. Lazalde, J. Torres y D. Vila-Viñas, «Hardware Libre. Recomendaciones para el fomento de la innovación ciudadana,» FLOK Society, Ecuador, 2015.
- [33] aprenderaprogramar.com, «Aprender a Programar,» [En línea]. Available: http://www.aprenderaprogramar.com/index.php?option=com_content&view=article&id=542:que-es-un-servidor-y-cuales-son-los-principales-tipos-de-servidores-proxydns-webftppop3-y-smtp-dhcp&catid=57:herramientas-informaticas&Itemid=179. [Último acceso: 26 Noviembre 2016].
- [34] linuxhispano y R. Hornero Luque, «Linux Hispano,» 16 Febrero 2010. [En línea]. Available: <http://www.linuxhispano.net/2010/02/16/distribuciones-linux-para-servidores/>. [Último acceso: 26 Noviembre 2016].
- [35] W3C, « World Wide Web Consortium,» [En línea]. Available: <https://www.w3.org/html/>. [Último acceso: 26 Noviembre 2016].
- [36] PHP, «PHP.net,» [En línea]. Available: <http://php.net/manual/es/intro-whatis.php>. [Último acceso: 26 Noviembre 2016].
- [37] W3C, «World Wide Web Consortium,» [En línea]. Available: <http://www.w3c.es/Divulgacion/GuiasBreves/HojasEstilo>. [Último acceso: 26 Noviembre 2016].
- [38] D. Pérez Valdés, «Maestros del WEB,» [En línea]. Available: <http://www.maestrosdelweb.com/los-diferentes-lenguajes-de-programacion-para-la-web/>. [Último acceso: 26 Noviembre 2016].

- [39] «sabelotodo.org,» [En línea]. Available: <http://www.sabelotodo.org/automovil/siselectrico.html>. [Último acceso: 10 Dic 2016].
- [40] ARCOTEL, «Agencia de Regulación y Control de Telecomunicaciones,» [En línea]. Available: <http://www.arcotel.gob.ec/espectro-radioelectrico/>. [Último acceso: 22 Diciembre 2016].
- [41] INAMHI, «Boletín Meteorológico Semestral enero-junio 2016,» Instituto Nacional de Meteorología e Hidrología, Quito, 2016.
- [42] ABC-MOTOR, «abc.es,» 6 Agosto 2015. [En línea]. Available: <http://www.abc.es/motor-reportajes/20150805/abci-segun-reciente-estudio-racc-201508051507.html>. [Último acceso: 22 Diciembre 2016].
- [43] Universidad de Huelva, «Vibraciones: Clasificación y efectos de la exposición,» [En línea]. Available: <http://www.uhu.es/servicio.prevencion/menuservicio/info/higiene/vibraciones.pdf>. [Último acceso: 25 Marzo 2017].
- [44] «Medición de las Vibraciones de las Piezas de Ensamble con Pruebas Físicas,» [En línea]. Available: http://catarina.udlap.mx/u_dl_a/tales/documentos/lim/tapia_r_c/capitulo5.pdf. [Último acceso: 25 Marzo 2017].
- [45] AUTOCASION, «La Corrosión en las Carrocerías,» [En línea]. Available: <http://www.autocasion.com/actualidad/reportajes/la-corrosion-en-las-carrocerias>. [Último acceso: 25 Marzo 2017].
- [46] OAS, «3.1 Climatología,» [En línea]. Available: <https://www.oas.org/dsd/publications/Unit/oea32s/ch13.htm>. [Último acceso: 28 Marzo 2017].
- [47] INAMHI, «Anuario Meteorológico,» 2011. [En línea]. Available: <http://www.serviciometeorologico.gob.ec/wp-content/uploads/anuarios/meteorologicos/Am%202011.pdf>. [Último acceso: 25 Marzo 2017].
- [48] ARDUINO, «Arduino,» [En línea]. Available: <https://www.arduino.cc/en>. [Último acceso: 22 Diciembre 2016].
- [49] ATMEL, «328P-328 Microcontroller,» [En línea]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Summary.pdf. [Último acceso: 17 Enero 2016].
- [50] Señal Móvil Ecuador, [En línea]. Available: <http://smovilecuador.arcotel.gob.ec/SenalMovilEcuadorWeb/mapas.html>. [Último acceso: 22 Diciembre 2016].

- [51] Mercado Libre, «Módulo RF para Arduino,» MGSYSTEM, [En línea]. Available: http://articulo.mercadolibre.com.ec/MEC-410514144-mgssystem-arduino-433mhz-rf-modulos-comunicacion-tx-rx-_JM. [Último acceso: 5 Enero 2017].
- [52] Mercado Libre, «Módulos para Arduino,» MGSYSTEM, [En línea]. Available: <http://listado.mercadolibre.com.ec/mgssystem-arduino>. [Último acceso: 5 Enero 2017].
- [53] Mercado Libre, «Baterías de Li-ion 18650,» [En línea]. Available: http://articulo.mercadolibre.com.ec/MEC-410813032-pila-bateria-recargable-litio-37v-5000ma-ultrafire-18650-_JM. [Último acceso: 5 Enero 2017].
- [54] olikraus, «Librería para Pantallas Oled de 128x32,» [En línea]. Available: <https://github.com/olikraus/u8glib>. [Último acceso: 6 Enero 2017].
- [55] Raspberry, «Raspberry Pi3,» [En línea]. Available: <https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>. [Último acceso: 6 Enero 2017].
- [56] Arduino, «Software de Programación Arduino,» [En línea]. Available: <https://www.arduino.cc/en/Main/Software>. [Último acceso: 5 Enero 2017].
- [57] EAGLE, «Software de Circuitos y PCB,» [En línea]. Available: <https://cadsoft.io/#>. [Último acceso: 5 Enero 2017].
- [58] SublimeText, «Editor de Texto para desarrollo Web,» [En línea]. Available: <https://www.sublimetext.com/>. [Último acceso: 6 Enero 2017].
- [59] GPS-NMEA, «Coordenadas NMEA 0183 de GPS,» [En línea]. Available: <http://aprs.gids.nl/nmea/#gga>. [Último acceso: 6 Enero 2017].
- [60] M. Hart, «arduiniana.org Librería para GPS,» [En línea]. Available: <http://arduiniana.org/libraries/tinygpsplus/>. [Último acceso: 6 Enero 2017].
- [61] I. Chulde, «Video de Youtube" Instalación de Raspbian en Raspberry Pi3",» 17 Enero 2017. [En línea]. Available: <https://www.youtube.com/watch?v=-ZN3eEDaYCQ>.
- [62] I. Chulde, «Video de Youtube"Servidor Web en Raspberry Pi3",» 18 Enero 2017. [En línea]. Available: https://www.youtube.com/watch?v=-_HCe2aWJ50&t=1s.
- [63] NO-IP, «Servicio de DDNS No-IP,» [En línea]. Available: <https://www.noip.com/>. [Último acceso: 6 Enero 2016].
- [64] I. Chulde, «Video de Youtube" Pruebas de Funcionamiento del Sistema",» 18 Enero 2017. [En línea]. Available: <https://www.youtube.com/watch?v=eHusu8SmPAE>.

- [65] HUNTER, «Hunter Soluciones Tecnologicas en Seguridad,» [En línea]. Available: <http://www.hunter.com.ec/productos/division-rastreo.aspx>. [Último acceso: 22 Dic 2016].
- [66] CHEVROLET, «Quiero mi Chevystar,» [En línea]. Available: <http://www.chevrolet.com.ec/quiero-chevystar.html>. [Último acceso: 22 Dic 2016].
- [67] TRACKLINK, «TRACKLINK,» [En línea]. Available: <http://www.tracklink.com.ec/rastreo-de-vehiculos-plan-individual/>. [Último acceso: 22 Dic 2016].
- [68] TRAIGOGPS, «Traigo GPS,» [En línea]. Available: <http://traigogps.com/precio-rastreo-satelital/>. [Último acceso: 22 Dic 2016].
- [69] Raspberry.org, «Sistemas Operativos de Raspberry Pi,» [En línea]. Available: <https://www.raspberrypi.org/downloads/>. [Último acceso: 6 Enero 2017].
- [70] inocar, «Información General de la Republica de Ecuador,» [En línea]. Available: https://www.inocar.mil.ec/docs/derrotero/derrotero_cap_1.pdf. [Último acceso: 25 Marzo 2017].

ANEXOS

Anexo A Programación Arduino NANO

```
//Programa de Rastreo GPS
/*Indicaciones:
 * Se envia un mensaje al telefono movil del Automovil y este realiza la accion determinada*/
// El Sistema siempre va a estar monitoreando la sierena de la Alarma y los SMS recibidos

//*****Librerias*****//
#include "U8glib.h" //Libreria de Display Oled 0.96"
#include <TinyGPS++.h> //Librería para el manejo del Módulo gps
#include <SoftwareSerial.h> //Librería para crear puertos virtuales en arduino
#include <RCSwitch.h> //Libreria para Módulo RF

//*****Declaracion de Metodos*****//
U8GLIB_SSD1306_128X32 u8g(U8G_I2C_OPT_NONE); //Metodo u8g para usa Display
RCSwitch mySwitch = RCSwitch(); //Metodo mySwitch para usar Módulo RF
SoftwareSerial GPSSerial(7, 8); //Pines Tx y Rx para modulo GPS
TinyGPSPlus gps; //Metodo para usar libreria GPS

//*****Declaracion de Variables*****//

#define vocina 5 //Vocina pinD5
#define bloqueom 6 //Bloqueo de Motor PinD6
int tiempo; //Almacena tiempo de encendido de sirena
int tiempo2; //Almacena tiempo de encendido de sirena
int tiempo3; //Almacena tiempo de encendido de sirena
float latitud,longitud,velocidad; //Almacena las variables del sistema de rastreo
int satelites; //Almacena numeros de satelites
boolean gps_status=0; //Estado del GPS para salir de bucle de reoger muestras

//Variables para indentificar SMS
char incoming_char = 0; //Variable que guarda los caracteres que envia el
String mensaje = ""; //Variable que almacena los mensajes recibidos
char movil_propietario[]="0982322923"; //Número de Teléfono del Propietario

//Logos del sistema
const uint8_t logo[] PROGMEM = {
  0xC0, 0x03, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0xA5, 0x3D, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0x6F, 0xE9, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0x6F, 0xE5, 0xFF, 0xFF, 0xCF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0x8F, 0xE5, 0xCD, 0xF9, 0x67, 0x9F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0xEF, 0xED, 0x81, 0xF1, 0x64, 0x9F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0x6F, 0xED, 0xF9, 0xC0, 0xE8, 0x3E, 0x3F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0xEF, 0xE1, 0xF9, 0xCC, 0x50, 0x52, 0x8F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0x90, 0x05, 0xF9, 0x88, 0x8E, 0x85, 0x3F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0xC2, 0x31, 0xC5, 0xC1, 0x6F, 0xD0, 0xA3, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0xC0, 0x03, 0xFF, 0xF3, 0xEF, 0xE1, 0x0B, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0xEF, 0xFF, 0xFF, 0xFF, 0xFF, 0xC0, 0x02, 0x7F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0xFF, 0xFF, 0xFF, 0xFF, 0xFE, 0x00, 0x00, 0x7F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
  0xFE, 0xDF, 0xFB, 0xEF, 0xE1, 0x80, 0x00, 0x3B, 0xFF, 0xFF, 0xDD, 0xFF, 0xBF, 0xAF, 0xFF, 0xFB,
  0x80, 0xC0, 0x10, 0x0F, 0xFF, 0x00, 0x01, 0x38, 0x04, 0x01, 0x00, 0xC0, 0x22, 0x08, 0x04, 0x01,
  0x08, 0x40, 0x10, 0x1F, 0xFE, 0x00, 0x00, 0x1E, 0x1C, 0x01, 0x00, 0x07, 0xE0, 0x38, 0x3C, 0x01,
  0x10, 0x40, 0x30, 0x0F, 0xFC, 0x40, 0x00, 0x1E, 0x1C, 0x01, 0x00, 0x07, 0xE0, 0x38, 0x1C, 0x01,
  0x2D, 0xF7, 0xFF, 0x8F, 0xE4, 0x00, 0x00, 0x3E, 0x1C, 0x79, 0x98, 0x77, 0xE8, 0x1B, 0x7C, 0x39,
  0xBF, 0xFF, 0xFF, 0xEF, 0x80, 0x80, 0x04, 0x1E, 0xDD, 0xBF, 0xFF, 0x7F, 0xEF, 0x7F, 0xFD, 0xBD,
  0xF7, 0xEF, 0xF0, 0xBF, 0xC6, 0x02, 0x0E, 0x1F, 0xBF, 0xFB, 0xBF, 0xFD, 0x77, 0xDF, 0x4F, 0xFB,
  0xFB, 0xEF, 0xFF, 0xFF, 0xF8, 0x00, 0x8C, 0x5F, 0xFF, 0xFB, 0xBE, 0xFB, 0xFF, 0xBF, 0xFF,
  0xFF, 0xFF, 0xFF, 0xFF, 0x91, 0x06, 0xBF, 0xFF, 0xFF, 0xFF, 0xFF, 0xEF, 0xFF, 0xFF, 0xFF,
  0xFF, 0xFF, 0xFF, 0xFF, 0xC3, 0xE5, 0x7F, 0xDF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
}
```

```

0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xE7, 0xFD, 0x7F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xEF, 0xFC, 0x7F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xF0, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xD1, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xEF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xD7, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFE, 0x1F, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF, 0xFF,
};

//*****Método de Configuración Inicial*****//
void setup(){
  power();    //Metodo de Encendido del Sistema
  iniciar();  //Metodo que Inicia todas las configuraciones iniciales
}
//*****Método Principal Repetitivo*****//
void loop(){
  //Condición para leer la variable de la Sirena
  if(digitalRead(vocina)== HIGH){
    milisSirena();    //Llama al metodo milisSirena
  }
  //Condición para leer el puerto serie del Módulo Sim900
  if(Serial.available(>0){
    incoming_char = Serial.read();    //Guarda los carácter enviados de GPRS
    mensaje = mensaje + incoming_char ; // Guarda los caracteres en un String
    identifica_SMS();    //Llama al metodo identifica SMS
  }
}
//*****Métodos del Programa*****//
//*****Método de Tiempo de Sirena*****//
//Analiza el tiempo que el pin5 de sirena esta encendido//
void milisSirena(){
  tiempo=millis();
  while(digitalRead(vocina)==HIGH){
    tiempo2=millis();
    tiempo3=(tiempo2-tiempo)/1000;
    if(tiempo3==5){ //Condición de tiempo de sirena en segundos
      notialarma();
      lcd_sistemalista();
    }
  }
}
//*****Metodo de Identificar SMS*****//
//Analiza el SMS recibido y procede a realizar la acción correspondiente//
void identifica_SMS()
{
  //Claves del Sistema
  int Rastreo = mensaje.indexOf("1600454878rastreo");
  int Posicion = mensaje.indexOf("1600454878posicion");
  int AbrirP = mensaje.indexOf("1600454878abrir");
  int CerrarP = mensaje.indexOf("1600454878cerrar");
  int BloqueoMotor=mensaje.indexOf("1600454878bloquear");
  int DesbloqueoMotor=mensaje.indexOf("1600454878desbloquear");
  //Si se cumple la condición del SMS procede a realizar las acciones
  if (Rastreo >= 0)
  {
    rastreoGPS(); //Método de Envío de datos a Servidor
    mensaje = "" ;
    lcd_sistemalista();
  }
  if (Posicion >= 0)

```

```

{
  envioposicion();//Metodo de Envio de Datos por SMS
  mensaje = "" ;
  lcd_sistemalista();
}
if (AbrirP >=0)
{
  lcd_abrirP();
  mySwitch.send(2794274, 24); //Envia los comandos por RF para abrir puerta
  mensaje = "" ;
  delay(1000);
  lcd_sistemalista();
}
if (CerrarP >=0)
{
  lcd_cerrarP();
  mySwitch.send(2794273, 24); //Envia los comandos por RF para cerrar puerta
  mensaje = "" ;
  delay(1000);
  lcd_sistemalista();
}
if (BloqueoMotor >=0)
{
  lcd_motorbloq();
  digitalWrite(bloqueoM,HIGH); //Activa el rele para impedir el paso de energia al motor
  mensaje = "" ;
  delay(1000);
  lcd_sistemalista();
}
if (DesbloqueoMotor >=0)
{
  lcd_motordes();
  digitalWrite(bloqueoM, LOW); //Desactiva el rele para permitir el paso
  mensaje = "" ;
  delay(1000);
  lcd_sistemalista();
}
}

//*****Metodo de SMS de Alarma*****//
//Envia un SMS de notificacion de la alarma activada//
void notialarma()
{
  lcd_envionotialar();
  repetirsmsalarm:
  if (enviarAT("AT+CREG?", "+CREG: 0,1", 1000) == 1) //comprueba la conexion a la red
  {
    char aux_str[50]; //Variable para concatenar el SMS
    enviarAT("AT+CMGF=1\r", "OK", 1000); //Comando AT para mandar un SMS
    sprintf(aux_str, "AT+CMGS=\"%s\"", movil_propietario); //Numero al que vamos a enviar el mensaje
    //Texto del mensaje
    if (enviarAT(aux_str, ">", 10000) == 1)
    {
      //Texto del SMS a enviar
      Serial.print(F("Alarma de Vehiculo Activada.\r" ));
      Serial.print(F("Porfavor tome precauciones con su vehiculo\r"));
      Serial.print(F("\r"));
      delay(1000);
      Serial.write(0x1A); // Ctrl+Z para enviar datos
    }

    lcd_envioSMS2();
  }
}

```

```

else
{
  goto repetirsmsalarm; //Sino hay señal se vuelve a enviar el SMS
}
}
//*****Metodo de SMS Posicion*****//
//Envia un SMS con los datos del GPS//
void envioposicion()
{
  lcd_envioSMS();
  enviarsms:
  if (enviarAT("AT+CREG?", "+CREG: 0,1", 1000) == 1) //comprueba la conexion a la red GSM
  {
    char aux_str[50]; //Variable para concatenar el SMS
    enviarAT("AT+CMGF=1\r", "OK", 1000); //Selecciona Formato para enviar SMS
    getgps(); //Obtiene los datos del GPS
    sprintf(aux_str, "AT+CMGS=\"%s\"", movil_propietario); //Numero al que vamos a enviar el mensaje
    if (enviarAT(aux_str, ">", 10000) == 1)
    {
      //Texto del mensaje
      Serial.print(F("Ubicacion:\rhttps://maps.google.com/maps?q=")); //Colocamos la url de google maps
      Serial.print(latitud,6); //Obtemos los datos de latitud del módulo gps y se lo enviamos al módulo gsm
      Serial.print(F("+"));
      Serial.println(longitud,6); //Obtemos los datos de longitud del módulo gps y se lo enviamos al módulo gsm
      Serial.print(F("Velocidad:\r"));
      Serial.print(velocidad);
      Serial.print(F("Km/h:\r"));
      Serial.print(F("\r"));
      delay(1000);
      Serial.write(0x1A); //Ctrl+Z para enviar SMS
    }
    //Colocar las variables del GPS en 0 para usar la proxima vez
    latitud=0;
    longitud=0;
    velocidad=0;
    lcd_envioSMS2();
  }
  else
  {
    goto enviarsms; //Sino hay señal se envia repite todo el proceso
  }
}
//*****Metodo de Rastreo*****//
//Envia Datos del GPS a Servidor en Raspberry Pi//
void rastreogps(){
  int i=0;
  int j=5; //Numero de muestras a enviar a servidor
  //Ciclo de envio de muestras
  for(i;i<=j;i++){
    volver:
    lcd_envioserver();
    //Configuracion para Acceder a GPRS
    enviarAT("AT", "OK", 1000);
    enviarAT("AT+CGDCONT=1,\"IP\", \"internet.movistar.com.ec\", \"OK\", 1000); //Designacion de PDP
    enviarAT("AT+CSTT= \"internet.movistar.com.ec\", \"movistar\", \"movistar\", \"OK\", 1000); //Designacion de APN
    enviarAT("AT+CIICR\r", "OK", 5000); //Conexion a GPRS
    Serial.print(F("AT+CIFSR\r")); //IP de red GPRS
    delay(2000);
    if(enviarAT("AT+CIPSTART= \"TCP\", \"rastreoivanchulde.ddns.net\", \"80\", \"CONNECT OK\", 10000) == 1){
  //Conexion a Servidor
  Serial.print(F("AT+CIPSEND\r")); // Envias Datos TCP o UDP
  delay(2000);

```

```

getgps(); //Metodo para obtener datos del GPS
Serial.print(F("GET /datosviagprs.php?lat=")); // Envio de Datos por medio de GET con un link
Serial.print(latitud,6);
Serial.print(F("&long="));
Serial.print(longitud,6);
Serial.print(F("&speed="));
Serial.print(velocidad);
Serial.print(F("&satelites="));
Serial.print(satelites);
Serial.print(F(" HTTP/1.1\r\n"));
Serial.print(F("Host: rastreoivanchulde.ddns.net\r\n\r\n"));
Serial.write(0x1A);
delay(3000);
//Variables de GPS a 0 para ocupar otra vez
latitud=0;
longitud=0;
velocidad=0;
}else{
//En caso de no haber conectado a servidor
lcd_envioservererror();
latitud=0;
longitud=0;
velocidad=0;
enviarAT("AT+CIPSHUT\r", "SHUT OK", 2000);
delay(1000);
goto volver;
}
//Cierra la conexion GPRS
lcd_envioserver2();
enviarAT("AT+CIPCLOSE\r", "CLOSE OK", 2000);
enviarAT("AT+CIPSHUT\r", "SHUT OK", 2000);
delay(1000);
}
}
//*****Metodo de GPS*****//
//Obtiene los datos enviados por el GPS, los pasa por la libreria TinyGPS++//
// y los convierte datos legibles para ser ocupados por los diversos metodos//
void getgps(){
while(1){ //Ingresa a un ciclo infinito hasta que obtenega nuevos datos
while(GPSSerial.available()>0){ //Lee los datos seriales de los pines 7 y 8 del modulo GPS
gps.encode(GPSSerial.read()); //envia los tramas NMEA a la libreria para procesarlas
if(gps.location.isUpdated()){
gps_status=1;
//Obtiene las coordenadas en forma legible
satelites=(gps.satellites.value(),5);
latitud=(gps.location.lat());
longitud=(gps.location.lng());
velocidad=(gps.speed.kmph());
}
}
}
//Condicion para romper el Ciclo infinito cuando halle datos GPS
if(latitud!=0 || longitud!=0){
gps_status=1;
}else{
gps_status=0;
}
if(gps_status)
break;
}
}
//*****Método de Encendido del Sistema*****//
//Enciende el Sistema y sus componentes e inicializa variables//
void power(void)

```

```

{
  //Configuración de Pantalla para inicializar
  if ( u8g.getMode() == U8G_MODE_R3G3B2 ) {
    u8g.setColorIndex(255); // white
  }
  else if ( u8g.getMode() == U8G_MODE_GRAY2BIT ) {
    u8g.setColorIndex(3); // max intensity
  }
  else if ( u8g.getMode() == U8G_MODE_BW ) {
    u8g.setColorIndex(1); // pixel on
  }
  else if ( u8g.getMode() == U8G_MODE_HICOLOR ) {
    u8g.setHiColorByRGB(255,255,255);
  }
  //Configura el pin de trabajo del modulo RF
  mySwitch.enableTransmit(10);
  lcd_iniciologo(); //Inicia el Logo del Sistema en la Pantalla
  pinMode(vocina, INPUT); //Declara pin de 5 como entrada
  pinMode(bloqueoM, OUTPUT); //Declara pin de 6 como salida
  digitalWrite(bloqueoM, LOW); //inicializa el pin 6 como apagado
  Serial.begin(9600); //Configura velocidad de modulo GPRS a 9600 baudios
  GPSSerial.begin(9600); //Configura velocidad de modulo GPS a 9600 baudios
  // Comprueba que el modulo SIM900 esta arrancado
  int respuesta = 0;
  if (enviarAT("AT", "OK", 2000) == 0)
  {
    //Descomentar si se trabaja con shield para arduino uno,
    //Donde el se enciende por medio de pin9
    /*pinMode(9, OUTPUT);
    digitalWrite(9,LOW);
    delay(1000);
    digitalWrite(9,HIGH);
    delay(2000);
    digitalWrite(9,LOW);
    delay(3000);*/

    while(respuesta ==0){//Ciclo que se ejecuta hasta que se encienda el modulo
      respuesta=enviarAT("AT","OK",2000);
    }
  }
}
//*****Método de Configaración de SIM900*****//
//Despues de encenderse el sistema se configura con comandos AT para que funcione//
void iniciar()
{
  //espera hasta estar conectado a la red movil
  while ( enviarAT("AT+CREG?", "+CREG: 0,1", 1000) == 0 )
  {
  }
  enviarAT("AT+CLIP=1\r", "OK", 1000); // Activamos la identificacion de llamadas
  enviarAT("AT+CMGF=1\r", "OK", 1000); //Configura el modo texto para enviar o recibir mensajes
  enviarAT("AT+CNMI=2,2,0,0,0\r", "OK", 1000); //Configuramos el modulo para que nos muestre los SMS
  recibidos por comunicacion serie
  lcd_sistemalista();
}

//*****Método para enviar comandos AT*****//
//Envia Comandos AT de manera sencilla esperando su respuesta a Sim900//
//Se tuliza para esperar la respuesta del modulo SIM900 y realizar un control mejor//
int enviarAT(String ATcommand, char* resp_correcta, unsigned int tiempo)
{
  int x = 0;
  bool correcto = 0;

```

```

char respuesta[100];
unsigned long anterior;

memset(respuesta, '\0', 100); // Inicializa el string
delay(100);
while ( Serial.available() > 0) Serial.read(); // Limpia el buffer de entrada
Serial.println(ATcommand); // Envia el comando AT
x = 0;
anterior = millis();
// Espera una respuesta
do {
  // si hay datos el buffer de entrada del UART lee y comprueba la respuesta
  if (Serial.available() != 0)
  {
    respuesta[x] = Serial.read();
    x++;
    // Comprueba si la respuesta es correcta
    if (strstr(respuesta, resp_correcta) != NULL)
    {
      correcto = 1;
    }
  }
}
// Espera hasta tener una respuesta
while ((correcto == 0) && ((millis() - anterior) < tiempo));
return correcto;
}

/*****Metodos para Gráficos de LCD 128x32*****/
/*****Logos*****/
void lcd_iniciologo(){
  u8g.firstPage();
  do {
    u8g.drawBitmapP( 0, 0, 16, 32, logo);
  } while( u8g.nextPage() );
  delay(3000);
}

/*****Texto Estatico LCD *****/

void lcd_sistemalista(){
  u8g.firstPage();
  do {
    u8g.setFont(u8g_font_6x12);
    u8g.drawStr( 22, 10, "Sistema Listo");
    u8g.drawStr( 17, 25, "Esperando Orden");
  } while( u8g.nextPage() );
}
void lcd_envioSMS2(){
  u8g.firstPage();
  do {
    u8g.setFont(u8g_font_6x12);
    u8g.drawStr( 45, 10, "SMS");
    u8g.drawStr( 40, 25, "Enviado");
  } while( u8g.nextPage() );
}
void lcd_envioserver2(){
  u8g.firstPage();
  do {
    u8g.setFont(u8g_font_6x12);

```

```

    u8g.drawStr( 25, 10, "Datos Subidos");
    u8g.drawStr( 33, 25, "a Servidor");
  } while( u8g.nextPage() );
  delay(2000);
}
void lcd_envioservererror(){
  u8g.firstPage();
  do {
    u8g.setFont(u8g_font_6x12);
    u8g.drawStr( 25, 10, " No Conexion");
    u8g.drawStr( 33, 25, "a Servidor");
  } while( u8g.nextPage() );
  delay(2000);
}

/*****Animaciones de Cargando ...*****/
void lcd_envioSMS(){
  for(int i=1; i<=120;i+=5){
  u8g.firstPage();
  do {
    u8g.setFont(u8g_font_6x12);
    u8g.drawFrame(0,12,128,9);
    u8g.drawBox(3,14,1+i,5);
    u8g.drawStr( 40, 7, "Enviando");
    u8g.drawStr( 35, 30, "SMS Espere");
  } while( u8g.nextPage() );
  }
}

void lcd_envioserver(){
  for(int i=1; i<=120;i+=5){
  u8g.firstPage();
  do {
    u8g.setFont(u8g_font_6x12);
    u8g.drawFrame(0,12,128,9);
    u8g.drawBox(3,14,1+i,5);
    u8g.drawStr( 25, 7, "Enviando Datos");
    u8g.drawStr( 35, 30, "a Servidor");
  } while( u8g.nextPage() );
  }
}

void lcd_envionotialar(){
  for(int i=1; i<=120;i+=5){
  u8g.firstPage();
  do {
    u8g.setFont(u8g_font_6x12);
    u8g.drawFrame(0,12,128,9);
    u8g.drawBox(3,14,1+i,5);
    u8g.drawStr( 20, 7, "Alarma Activada");
    u8g.drawStr( 25, 30, "Enviando SMS");
  } while( u8g.nextPage() );
  }
}

void lcd_abrirP(){
  for(int i=1; i<=120;i+=5){
  u8g.firstPage();
  do {
    u8g.setFont(u8g_font_6x12);
    u8g.drawFrame(0,12,128,9);
    u8g.drawBox(3,14,1+i,5);

```



```

        u8g.drawStr( 40, 7, "Abriendo");
        u8g.drawStr( 42, 30, "Puertas");
    } while( u8g.nextPage() );
}
}

void lcd_cerrarP(){
for(int i=1; i<=120;i+=5){
    u8g.firstPage();
    do {
        u8g.setFont(u8g_font_6x12);
        u8g.drawFrame(0,12,128,9);
        u8g.drawBox(3,14,1+i,5);
        u8g.drawStr( 40, 7, "Cerrando");
        u8g.drawStr( 42, 30, "Puertas");
    } while( u8g.nextPage() );
}
}

void lcd_motorbloq(){
for(int i=1; i<=120;i+=5){
    u8g.firstPage();
    do {
        u8g.setFont(u8g_font_6x12);
        u8g.drawFrame(0,12,128,9);
        u8g.drawBox(3,14,1+i,5);
        u8g.drawStr( 35, 7, "Bloqueando");
        u8g.drawStr( 47, 30, "Motor");
    } while( u8g.nextPage() );
}
}

void lcd_motordes(){
for(int i=1; i<=120;i+=5){
    u8g.firstPage();
    do {
        u8g.setFont(u8g_font_6x12);
        u8g.drawFrame(0,12,128,9);
        u8g.drawBox(3,14,1+i,5);
        u8g.drawStr( 30, 7, "Desbloqueando");
        u8g.drawStr( 47, 30, "Motor");
    } while( u8g.nextPage() );
}
}
}

```

Anexo B Programación de Pagina WEB

Programación de Pagina WEB

A continuación, la programación de cada archivo de la página web utilizado en el proyecto

conexión.php

```
<?php
$host = "localhost";
$user = "root";
$pw = "1600454878";
$db = "datosdegps";
?>
```

datosviagprs.php

```
<?php
include("conexion.php");

$con=mysql_connect($host,$user,$pw)or die("Problema al Conectar server");
mysql_select_db($db,$con)or die("Problema al conectar bd");

$latitud=$_GET["lat"];
$longitud=$_GET["long"];
$speed=$_GET["speed"];
$satelites=$_GET["satelites"];

$fecha=date('Y-m-d');
$hora=date("H:i:s");

$sql="INSERT INTO `informacion`(`latitud`, `longitud`, `speed`, `satelites`, `fecha`,
`hora`)
VALUES ('".$latitud."', '".$longitud."', '".$speed."', '".$satelites."', '".$fecha."',
'".$hora."')";
$result=mysql_query($sql);
?>
```

index.php

```
<?php
include("conexion.php");
session_start();
if(!isset($_SESSION["user"])){
    header("location:login.php");
}

//Codigo para Establecer Comunicacion con la Base de Datos MySQL
//Solo Sirve para extraer el ultimo punto de latitud y longitud de la Base de Datos y
//centrar el mapa

$con=mysql_connect($host,$user,$pw)or die("Problema al Conectar server");
mysql_select_db($db,$con)or die("Problema al conectar bd");
$registro = mysql_query("SELECT * FROM informacion");

while($reg = mysql_fetch_array($registro)){
    $latitud=$reg["latitud"];
    $longitud=$reg["longitud"];

}
?>

<!DOCTYPE html>
<html>
<head>
    <title>GPS TRACKER</title>
    <link rel="shortcut icon" href="imagenes/icono1.ico">
    <link href="CSS/estilos.css" rel="stylesheet" type="text/css">
    <link href="https://fonts.googleapis.com/css?family=Baloo+Tamma" rel="stylesheet">

    <script src="js/jquery-1.12.3.min.js" charset="utf-8"></script>
</head>
<body>
<div id="cuerpo">

    <div id="encabezado">

        <div id="logo"></div>
        <div id="cerrar"> <a href="logout.php">Salir</a></div>
        <div id="titulo">
            Sistema de Rastreo GPS
        </div>

        <div id="usuariolog">
            <p1> Bienvenido Usuario: <?php echo " $_SESSION["user"]." ?> </p1>
        </div>
    </div>
</div>
<div id="map"></div>
<script type="text/javascript">
```

```

var map;
function initMap() {
mysql      var coordenada1= "<?php echo $latitud; ?>";          //Guardamos la latitud de

      var coordenada2= "<?php echo $longitud; ?>"; //Guardamos la longitud de mysql
      //Variable como metodo de lat y lon de google maps
      var pos = new google.maps.LatLng(coordenada1, coordenada2);

      map = new google.maps.Map(document.getElementById('map'), {
        center: pos,
        zoom: 17,
        mapTypeId: google.maps.MapTypeId.HYBRID
      });

      var infoWindow = new google.maps.InfoWindow;
      //llamado de la función downloadUrl para que cargue el archivo XML, que a su vez carga
los datos de toda la base de datos
      downloadUrl("generarxml2.php", function(data) {
        var xml = data.responseXML;
        var markers = xml.documentElement.getElementsByTagName("marker");
        //Ciclo for para mostrar mostrar todos los marcadores con su respectiva informacion
        for (var i = 0; i < markers.length; i++) {
variables          //Extraccion de los datos del archivo XML y almacenarlos en

          var speed = markers[i].getAttribute("speed");
          var satelites = markers[i].getAttribute("satelites");
          var time = markers[i].getAttribute("hora");
          var date = markers[i].getAttribute("fecha");
          //Variable con metodo de latitud y longitud para guardar cada coordenada de la
base de datos y ponerla a un marcador respectivo
          var point = new google.maps.LatLng(
            parseFloat(markers[i].getAttribute("lat")),
            parseFloat(markers[i].getAttribute("lng")));
          //Variable que guarda que informacion se muestra en la venta de
informacion de cada marcador al hacer clic.
          var info = "<b>Fecha: " + date + "</b> <br/>Hora: " + time + "</b>
<br/>Velocidad: " + speed + "Km/h<br/>Satelites: " + satelites;
          //Variable que imprime cada marcador deacuerdo con la
configuracion que se tenga
          var marker = new google.maps.Marker({
            map: map,
            position: point,
          });
          //LLlamado de la funcion binInfoWindow, la encargada de visualizar los datos al
hacer clic en cada marcador, en su interior estan las variables que se declararon posteriormente.
          bindInfoWindow(marker, map, infoWindow, info);
          var flightPath = new google.maps.Polyline({
            path: point,
            geodesic: true,
            strokeColor: '#FF0000',
            strokeOpacity: 1.0,
            strokeWeight: 2
          });
          flightPath.setMap(map);
        }
      });

```

```

    }
    //Funcion para mostrar ventanas de informacion al hacer clic en un amrcador
    function bindInfoWindow(marker, map, infoWindow, html) {
        google.maps.event.addListener(marker, 'click', function() {
            infoWindow.setContent(html);
            infoWindow.open(map, marker);
        });
    }
    //Funcion para obtener un archivo XML alojado en el sitio
    function downloadUrl(url, callback) {
        var request = window.ActiveXObject ?
            new ActiveXObject('Microsoft.XMLHTTP') :
            new XMLHttpRequest;
        request.onreadystatechange = function() {
            if (request.readyState == 4) {
                request.onreadystatechange = doNothing;
                callback(request, request.status);
            }
        };

        request.open('GET', url, true);
        request.send(null);
    }

    function doNothing() {}

</script>
<script async defer
    src="https://maps.googleapis.com/maps/api/js?key=AlzaSyDGhwy_wPpV3m9GaFsaxDYxTy-
mLHejIMg&callback=initMap">
</script>
</body>
</html>

```

login.php

```
<?php

session_start();
if (isset($_SESSION["user"])) {
    header("location:index.php");
}

?>

<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <title>Login</title>
    <link rel="shortcut icon" href="imagenes/icono1.ico">
    <link href="https://fonts.googleapis.com/css?family=Baloo+Tamma" rel="stylesheet">
    <link rel="stylesheet" href="bootstrap/css/bootstrap.min.css" media="screen" title="no title"
charset="utf-8">
    <link href="CSS/index.css" rel="stylesheet" type="text/css">
    <script src="js/jquery-1.12.3.min.js" charset="utf-8"></script>
    <script src="bootstrap/js/bootstrap.min.js" charset="utf-8"></script>
</head>
<body>
    <div class="container">
        <div class="row">
            <div class="col-md-6 col-md-offset-3">
                <form method="post">
                    <br><br>
                    <h1><p class="text-center">Sistema de Rastreo GPS</p></h1>
                    <div class="imagen">
                         </div>
                    <br><br>
                    <div class="form-group1">
                        <label for="user">Usuario</label>
                        <input type="text" name="user" id="user" class="form-control">
                    </div>
                    <div class="form-group2">
                        <label for="pass">Password</label>
                        <input type="password" name="pass" id="pass" class="form-control">
                    </div>
                    <br><br>
                    <div class="form-group3">
                        <input type="button" name="login" id="login" value="Login" class="btn btn-default">
                    </div>
                    <br>
                    <span id="result"></span>
                </form>
            </div>
        </div>
    </div>
</body>
</html>
<script>
$(document).ready(function() {
```

```

$('#login').click(function(){
    var user = $('#user').val();
    var pass = $('#pass').val();
    if($.trim(user).length > 0 && $.trim(pass).length > 0){
        $.ajax({
            url:"verificar.php",
            method:"POST",
            data:{user:user, pass:pass},
            cache:"false",
            beforeSend:function() {
                $('#login').val("Conectando...");
            },
            success:function(data) {
                $('#login').val("Login");
                if (data=="1") {
                    $(location).attr('href','index.php');
                } else {
                    $("#result").html("<div class='alert alert-dismissible alert-danger'><button type='button'
class='close' data-dismiss='alert'>&times;</button><strong>jError!</strong> las credenciales son
incorrectas.</div>");
                }
            }
        });
    }
});
</script>

```

logout.php

```

<?php
session_start();
session_destroy();
header("location:login.php");
?>

```

verificar.php

```

<?php
session_start();
$conn = mysqli_connect("localhost","root","1600454878","usuarios");
if(isset($_POST["user"]) && isset($_POST["pass"])){
    $user = mysqli_real_escape_string($conn, $_POST["user"]);
    $pass = mysqli_real_escape_string($conn, $_POST["pass"]);
    $sql = "SELECT user FROM informacion WHERE user='$user' AND pass='$pass'";
    $result = mysqli_query($conn, $sql);
    $num_row = mysqli_num_rows($result);
    if ($num_row == "1") {
        $data = mysqli_fetch_array($result);
        $_SESSION["user"] = $data["user"];
        echo "1";
    } else {
        echo "error";
    }
} else {
    echo "error";
}
?>

```

generarxml2.xml

```
<?php
include("conexion.php");
session_start();
if(!isset($_SESSION["user"])){
    header("location:login.php");
}
$fechaconsulta=$_POST['fecha'];
echo $fechaconsulta;
function parseToXML($htmlStr)
{
$xmlStr=str_replace('<','&lt;',$htmlStr);
$xmlStr=str_replace('>','&gt;',$xmlStr);
$xmlStr=str_replace('"','&quot;',$xmlStr);
$xmlStr=str_replace("'",'&#39;',$xmlStr);
$xmlStr=str_replace("&","&amp;",$xmlStr);
return $xmlStr;
}
// Opens a connection to a MySQL server
$conection=mysql_connect ('localhost', $user, $pw);
if (!$conection) {
    die('Not connected : ' . mysql_error());
}
// Set the active MySQL database
$db_selected = mysql_select_db($db, $conection);
if (!$db_selected) {
    die ('Can\'t use db : ' . mysql_error());
}
// Select all the rows in the markers table
$query = "SELECT * FROM `informacion` WHERE 1 ";
$result = mysql_query($query);
if (!$result) {
    die('Invalid query: ' . mysql_error());
}
header("Content-type: text/xml");
// Start XML file, echo parent node
echo '<markers>';
// Iterate through the rows, printing XML nodes for each
while ($row = @mysql_fetch_assoc($result)){
    // ADD TO XML DOCUMENT NODE
    echo '<marker ' ;
    echo 'id="' . $row['id'] . '" ' ;
    echo 'lat="' . $row['latitud'] . '" ' ;
    echo 'lng="' . $row['longitud'] . '" ' ;
    echo 'speed="' . $row['speed'] . '" ' ;
    echo 'satelites="' . $row['satelites'] . '" ' ;
    echo 'hora="' . $row['hora'] . '" ' ;
    echo 'fecha="' . $row['fecha'] . '" ' ;
    echo '>';
}
// End XML file
echo '</markers>';
?>
```


NEO-6 series

Versatile u-blox 6 GPS modules

Highlights

- UART, USB, DDC (I²C compliant) and SPI interfaces
- Available in Crystal and TCXO versions
- Onboard RTC crystal for faster warm and hot starts
- 1.8 V and 3.0 V variants



NEO-6:
12.2 x 16.0 x 2.4 mm

Features

- u-blox 6 position engine:
 - Navigate down to -162 dBm and -148 dBm coldstart
 - Faster acquisition with AssistNow Autonomous
 - Configurable power management
 - Hybrid GPS/SBAS engine (WAAS, EGNOS, MSAS)
 - Anti-jamming technology
- Simple integration with u-blox wireless modules
- A-GPS: AssistNow Online and AssistNow Offline services, OMA SUPL compliant
- Backward compatible (hardware and firmware); easy migration from NEO-5 family or NEO-4S
- LCC package for reliable and cost effective manufacturing
- Compatible with u-blox GPS Solution for Android
- Based on GPS chips qualified according to AEC-Q100
- Manufactured in ISO/TS 16949 certified production sites
- Qualified according to ISO 16750

Product description

The NEO-6 module series brings the high performance of the u-blox6 position engine to the miniature NEO form factor. u-blox6 has been designed with low power consumption and low costs in mind. Intelligent power management is a breakthrough for low-power applications. These receivers combine a high level of integration capability with flexible connectivity options in a small package. This makes them perfectly suited for mass-market end products with strict size and cost requirements. The DDC interface provides connectivity and enables synergies with u-blox LEON and LISA wireless modules. All NEO-6 modules are manufactured in ISO/TS 16949 certified sites. Each module is tested and inspected during production. The modules are qualified according to ISO 16750 - Environmental conditions and electrical testing for electrical and electronic equipment for road vehicles.

Product selector

Model	Type	Supply	Interfaces	Features
	Standalone GPS Standalone GLONASS Timing & Raw Data Dead Reckoning	1.75 V - 2.0 V 2.7 V - 3.6 V	UART USB SPI DDC (I ² C compliant)	Programmable (Flash) FW update Oscillator RTC crystal Antenna supply and supervisor Configuration pins Timepulse External interrupt / Wakeup
NEO-6G	•	•	• • • •	T • ○ 3 1 •
NEO-6Q	•	•	• • • •	T • ○ 3 1 •
NEO-6M	•	•	• • • •	C • ○ 3 1 •

○ = requires external components and integration on application processor

C = Crystal / T = TCXO



Receiver performance data

Receiver type	50-channel u-blox 6 engine GPS L1 C/A code SBAS: WAAS, EGNOS, MSAS	
Navigation update rate	up to 5 Hz	
Accuracy ¹	Position	2.5 m CEP
	SBAS	2.0 m CEP
Acquisition ²	NEO-6G/Q	NEO-6M
	Cold starts:	26 s
	Aided starts ² :	< 3 s
	Hot starts:	1 s
Sensitivity ³	NEO-6G/Q	NEO-6M
	Tracking:	-162 dBm
	Cold starts:	-148 dBm
	Hot starts:	-157 dBm

¹ All SV @ -130 dBm

² Dependent on aiding data connection speed and latency

³ Demonstrated with a good active antenna

Electrical data

Power supply	2.7 V – 3.6 V (NEO-6Q/6M) 1.75 V – 2.0 V (NEO-6G)
Power consumption	111 mW @ 3.0 V (continuous) 33 mW @ 3.0 V Power Save Mode (1 Hz) 68 mW @ 1.8 V (continuous) 22 mW @ 1.8 V Power Save Mode (1 Hz)
Backup power	1.4 V – 3.6 V, 22 μ A
Supported antennas	Active and passive

Interfaces

Serial interfaces	1 UART 1 USB V2.0 full speed 12 Mbit/s 1 DDC (I ² C compliant) 1 SPI
Digital I/O	Configurable timepulse 1 EXTINT input for Wakeup
Serial and I/O	Voltages: 2.7 – 3.6 V (NEO-6Q/6M) 1.75 – 2.0 V (NEO-6G)
Timepulse	Configurable 0.25 Hz to 1 kHz
Protocols	NMEA, UBX binary, RTCM

Legal Notice

u-blox reserves all rights to this document and the information contained herein. Products, names, logos and designs described herein may in whole or in part be subject to intellectual property rights. Reproduction, use, modification or disclosure to third parties of this document or any part thereof without the express permission of u-blox is strictly prohibited.

The information contained herein is provided "as is". No warranty of any kind, either express or implied, is made in relation to the accuracy, reliability, fitness for a particular purpose or content of this document. This document may be revised by u-blox at any time. For most recent documents, please visit www.u-blox.com.

Copyright © 2011, u-blox AG

Specification applies to FW 7

Package

24 pin LCC (Leadless Chip Carrier): 12.2 x 16.0 x 2.4 mm, 1.6 g

Pinout



Environmental data, quality & reliability

Operating temp. –40° C to 85° C

Storage temp. –40° C to 85° C

RoHS compliant (lead-free)

Qualification according to ISO 16750

Manufactured in ISO/TS 16949 certified production sites

Support products

u-blox 6 Evaluation Kits:

Easy-to-use kits to get familiar with u-blox 6 positioning technology, evaluate functionality, and visualize GPS performance.

EVK-6H: u-blox 6 Evaluation Kit with TCXO, suitable for NEO-6G, NEO-6Q

EVK-6P: u-blox 6 Evaluation Kit with crystal, suitable for NEO-6M

Ordering information

NEO-6G-0	u-blox 6 GPS Module, 1.8V, TCXO, 12x16mm, 250 pcs/reel
NEO-6M-0	u-blox 6 GPS Module, 12x16mm, 250 pcs/reel
NEO-6Q-0	u-blox 6 GPS Module, TCXO, 12x16mm, 250 pcs/reel

Available as samples and tape on reel (250 pieces)

Contact us

HQ Switzerland
+41 44 722 7444
info@u-blox.com

EMEA
+41 44 722 7444
info@u-blox.com

Americas
+1 703 483 3180
info_us@u-blox.com

APAC – Singapore
+65 6734 3811
info_ap@u-blox.com

China
+86 10 68 133 545
info_cn@u-blox.com

Japan
+81 3 5775 3850
info_jp@u-blox.com

Korea
+82 2 542 0861
info_kr@u-blox.com

Taiwan
+886 2 2657 1090
info_tw@u-blox.com

Anexo D SIM 900 Chip Especificaciones

SIM900
The GSM/GPRS Module for M2M applications

SIM900 GSM/GPRS Module



The SIM900 is a complete Quad-band GSM/GPRS solution in a SMT module which can be embedded in the customer applications.

Featuring an industry-standard interface, the SIM900 delivers GSM/GPRS 850/900/1800/1900MHz performance for voice, SMS, Data, and Fax in a small form factor and with low power consumption. With a tiny configuration of 24mm x 24mm x 3 mm, SIM900 can fit almost all the space requirements in your M2M application, especially for slim and compact demand of design.

- SIM900 is designed with a very powerful single-chip processor integrating AMR926EJ-S core
- Quad - band GSM/GPRS module with a size of 24mmx24mmx3mm
- SMT type suit for customer application
- An embedded Powerful TCP/IP protocol stack
- Based upon mature and field-proven platform, backed up by our support service, from definition to design and production

General features

- Quad-Band 850/ 900/ 1800/ 1900 MHz
- GPRS multi-slot class 10/8
- GPRS mobile station class B
- Compliant to GSM phase 2/2+
 - Class 4 (2 W @850/ 900 MHz)
 - Class 1 (1 W @ 1800/1900MHz)
- Dimensions: 24* 24 * 3 mm
- Weight: 3.4g
- Control via AT commands (GSM 07.07 ,07.05 and SIMCOM enhanced AT Commands)
- SIM application toolkit
- Supply voltage range 3.4 ... 4.5 V
- Low power consumption
- Operation temperature: -30 °C to +80 °C

Specifications for fax

- Group 3, class 1

Specifications for data

- GPRS class 10: max. 85.6 kbps (downlink)
- FBCCH support
- Coding schemes CS 1, 2, 3, 4
- CSD up to 14.4 kbps
- USSD
- Non transparent mode
- PPP-stack

Specifications for SMS via GSM Pin Assignment

/ GPRS

- Point-to-point MO and MT
- SMS cell broadcast
- Text and PDU mode

Drivers

- MUX Driver

Specifications for voice

- Tricodec
 - Half rate (HR)
 - Full rate (FR)
 - Enhanced Full rate (EFR)

- Hands-free operation (Echo suppression)
- AMR
 - Half Rate(HR)
 - Full Rate(FR)

Interfaces

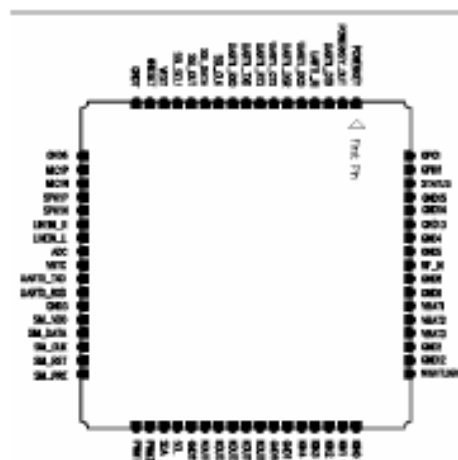
- Interface to external SIM 3V/ 1.8V
- analog audio interface
- RTC backup
- SPI interface
- Serial interface
- Antenna pad
- I2C
- GPIO
- PWM
- ADC

Compatibility

- AT cellular command interface

Approvals (in planning)

- CE
- FCC
- ROHS
- PTCRB
- GCF
- AT&T
- IC
- TA



More about SIM900 module, Please contact: Tel:+86 21 32523300
 Fax:+86 21 32523301
 Email:simcom@sim.com

Anexo E TCP Client Connection



SIM900_TCPIP_Application Note

UDP client and TCP server.

3.1.1 How to establish a TCP client connection

Firstly, before any TCP/UDP related operation, the module should be connected to GSM or GPRS network. You can use the commands "AT+CREG?" and "AT+CGATT?" to query the GSM network registration status whether the module has been attached to GPRS service.

AT+CPIN?

+CPIN: READY

OK

AT+CSQ

CSQ: 20,0

OK

AT+CREG?

+CREG: 0,1

OK

AT+CGATT?

+CGATT: 1

OK

Secondly, you should use the command group AT+CSTT, AT+CIICR and AT+CIFSR to start the task and activate the wireless connection. Lastly, you can establish TCP connection between SIM900 and server by AT command (AT+CIPSTART="TCP","IP Address of server", "port number of server"). If the connection is established successfully, response "CONNECT OK" will come up from the module. Now you can send data to server with "AT+CIPSEND". "AT+CIPSEND" will return with promoting mark ">", you should write data after ">" then issue CTRL+Z (0x1a) to send. If sending is successful, it will respond "SEND OK". And if there is data coming from server, the module will receive the data automatically from the serial port. You can close the TCP connection with "AT+CIPCLOSE" command. Below is an example of TCP connection to remote server.

AT+CGATT?

+CGATT: 1

OK

```
AT+CSTT="CMNET" // Start task and set APN. See Note[1].

OK

AT+CIICR // Bring up wireless connection (GPRS or
CSD)

OK

AT+CIFSR // Get local IP address

10.78.245.128

AT+CIPSTART="TCP","116.228.221.51", "8500" // Start up the connection

OK

CONNECT OK // The TCP connection has been established
successfully.

AT+CIPSEND

> hello TCP server //Send data to remote server, CTRL+Z (0x1a) to
send. See Note[2]

SEND OK // The remote server receives the data.

hello sim900 // Received data from remote server

CLOSED // Remote server closed the connection
```

Note[1]: The default APN is "CMNET", with no username or password. Please ask your local GSM provider to get the APN.

Note[2]: You should write data only after the promoting mark ">", and then use CTRL+Z to send. You can use command "AT+CIPSPRT" to set whether echo promote ">" after issuing "AT+CIPSEND".

3.1.2 How to establish a UDP client connection

The process of establishing UDP connection is similar to TCP.

```
AT+CGATT?

+CGATT: 1

OK

AT+CSTT="CMNET"
```

Anexo F Opto acoplador 4N25



GENERAL PURPOSE 6-PIN PHOTOTRANSISTOR OPTOCOUPLERS

4N25
4N37

4N26
H11A1

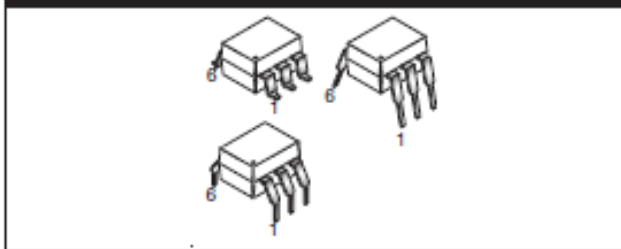
4N27
H11A2

4N28
H11A3

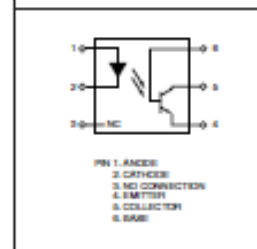
4N35
H11A4

4N36
H11A5

WHITE PACKAGE (-M SUFFIX)



SCHEMATIC



BLACK PACKAGE (NO -M SUFFIX)



DESCRIPTION

The general purpose optocouplers consist of a gallium arsenide infrared emitting diode driving a silicon phototransistor in a 6-pin dual in-line package.

FEATURES

- Also available in white package by specifying -M suffix, eg. 4N25-M
- UL recognized (File # E90700)
- VDE recognized (File # 94766)
 - Add option V for white package (e.g., 4N25V-M)
 - Add option 300 for black package (e.g., 4N25.300)

APPLICATIONS

- Power supply regulators
- Digital logic inputs
- Microprocessor inputs

4N25 4N37	4N26 H11A1	4N27 H11A2	4N28 H11A3	4N35 H11A4	4N36 H11A5
--------------	---------------	---------------	---------------	---------------	---------------

ELECTRICAL CHARACTERISTICS ($T_A = 25^\circ\text{C}$ unless otherwise specified)

INDIVIDUAL COMPONENT CHARACTERISTICS

Parameter	Test Conditions	Symbol	Min	Typ*	Max	Unit
EMITTER						
Input Forward Voltage	($I_F = 10\text{ mA}$)	V_F		1.18	1.50	V
Reverse Leakage Current	($V_R = 6.0\text{ V}$)	I_R		0.001	10	μA
DETECTOR						
Collector-Emitter Breakdown Voltage	($I_C = 1.0\text{ mA}$, $I_F = 0$)	BV_{CEO}	30	100		V
Collector-Base Breakdown Voltage	($I_C = 100\ \mu\text{A}$, $I_F = 0$)	BV_{CBO}	70	120		V
Emitter-Collector Breakdown Voltage	($I_E = 100\ \mu\text{A}$, $I_F = 0$)	BV_{ECO}	7	10		V
Collector-Emitter Dark Current	($V_{CE} = 10\text{ V}$, $I_F = 0$)	I_{CEO}		1	50	nA
Collector-Base Dark Current	($V_{CB} = 10\text{ V}$)	I_{CBO}			20	nA
Capacitance	($V_{CE} = 0\text{ V}$, $f = 1\text{ MHz}$)	C_{CE}		8		pF

ISOLATION CHARACTERISTICS

Characteristic	Test Conditions	Symbol	Min	Typ*	Max	Units
Input-Output Isolation Voltage	(Non '-M', Black Package) ($f = 60\text{ Hz}$, $t = 1\text{ min}$)	V_{ISO}	5300			Vac(rms)
	('-M', White Package) ($f = 60\text{ Hz}$, $t = 1\text{ sec}$)		7500			Vac(pk)
Isolation Resistance	($V_{LO} = 500\text{ VDC}$)	R_{ISO}	10^{11}			Ω
Isolation Capacitance	($V_{LO} = 8$, $f = 1\text{ MHz}$)	C_{ISO}		0.5		pF
	('-M' White Package)			0.2	2	pF

Note

* Typical values at $T_A = 25^\circ\text{C}$

4N25 4N37	4N26 H11A1	4N27 H11A2	4N28 H11A3	4N35 H11A4	4N36 H11A5
--------------	---------------	---------------	---------------	---------------	---------------

TRANSFER CHARACTERISTICS (T _A = 25°C Unless otherwise specified.)							
DC Characteristic	Test Conditions	Symbol	Device	Min	Typ*	Max	Unit
Current Transfer Ratio, Collector to Emitter	(I _F = 10 mA, V _{CE} = 10 V)	CTR	4N35 4N36 4N37	100			%
			H11A1	50			
			H11A5	30			
	4N25 4N26 H11A2 H11A3		20				
	4N27 4N28 H11A4		10				
	4N35 4N36 4N37		40				
	(I _F = 10 mA, V _{CE} = 10 V, T _A = -55°C)		4N35 4N36 4N37	40			
	(I _F = 10 mA, V _{CE} = 10 V, T _A = +100°C)		4N35 4N36 4N37	40			
Collector-Emitter Saturation Voltage	(I _C = 2 mA, I _F = 50 mA)	V _{CE(SAT)}	4N25 4N26 4N27 4N28			0.5	V
	(I _C = 0.5 mA, I _F = 10 mA)		4N35 4N36 4N37			0.3	
			H11A1 H11A2 H11A3 H11A4 H11A5			0.4	
AC Characteristic							
Non-Saturated Turn-on Time	(I _F = 10 mA, V _{CC} = 10 V, R _L = 100Ω) (Fig.20)	T _{ON}	4N25 4N26 4N27 4N28 H11A1 H11A2 H11A3 H11A4 H11A5		2		μs
Non Saturated Turn-on Time	(I _C = 2 mA, V _{CC} = 10 V, R _L = 100Ω) (Fig.20)	T _{ON}	4N35 4N36 4N37		2	10	μs

4N25 4N26 4N27 4N28 4N35 4N36
4N37 H11A1 H11A2 H11A3 H11A4 H11A5

TRANSFER CHARACTERISTICS ($T_A = 25^\circ\text{C}$ Unless otherwise specified.) (Continued)							
AC Characteristic	Test Conditions	Symbol	Device	Min	Typ*	Max	Unit
Turn-off Time	($I_F = 10 \text{ mA}$, $V_{CC} = 10 \text{ V}$, $R_L = 100\Omega$) (Fig.20)	T_{OFF}	4N25 4N26 4N27 4N28 H11A1 H11A2 H11A3 H11A4 H11A5		2		μs
	($I_C = 2 \text{ mA}$, $V_{CC} = 10 \text{ V}$, $R_L = 100\Omega$) (Fig.20)		4N35 4N36 4N37		2	10	

* Typical values at $T_A = 25^\circ\text{C}$

4N25
4N37

4N26
H11A1

4N27
H11A2

4N28
H11A3

4N35
H11A4

4N36
H11A5

TYPICAL PERFORMANCE CURVES

Fig. 1 LED Forward Voltage vs. Forward Current (Black Package)

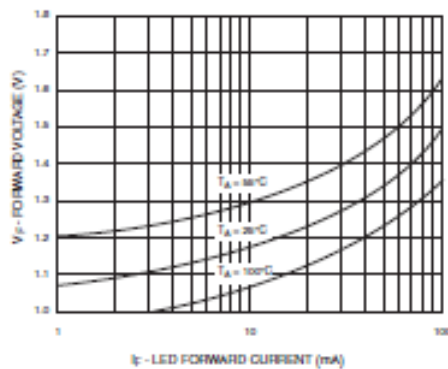


Fig. 2 LED Forward Voltage vs. Forward Current (White Package)

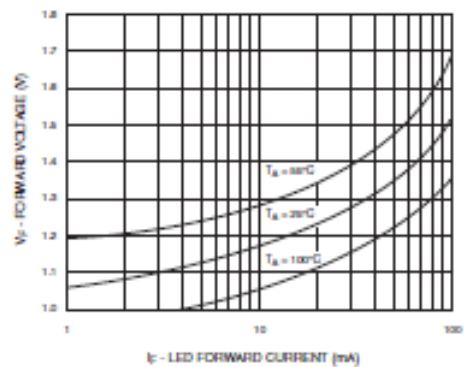


Fig. 3 Normalized CTR vs. Forward Current (Black Package)

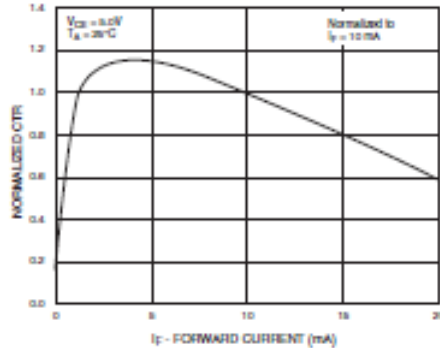


Fig. 4 Normalized CTR vs. Forward Current (White Package)

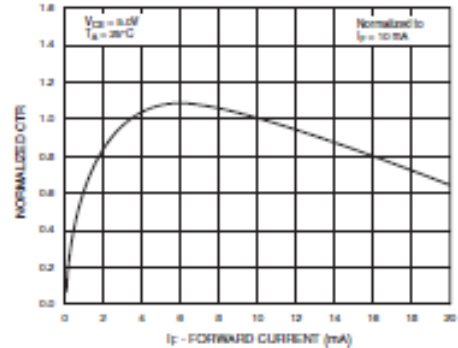


Fig. 5 Normalized CTR vs. Ambient Temperature (Black Package)

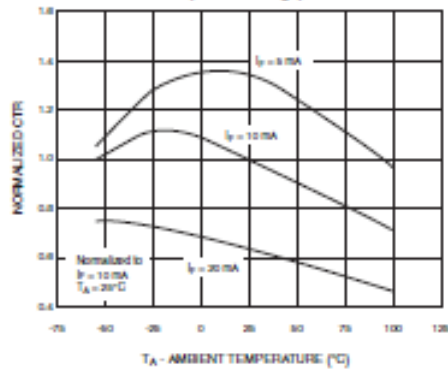
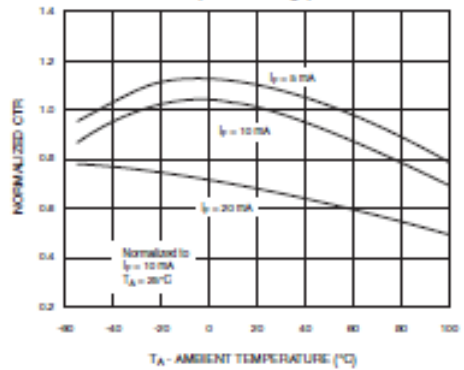


Fig. 6 Normalized CTR vs. Ambient Temperature (White Package)



4N25
4N37

4N26
H11A1

4N27
H11A2

4N28
H11A3

4N35
H11A4

4N36
H11A5

Fig. 7 CTR vs. RBE (Unsaturated)
(Black Package)

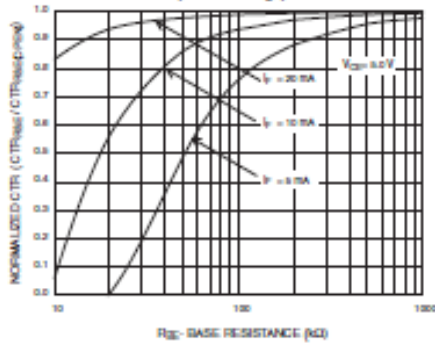


Fig. 8 CTR vs. RBE (Unsaturated)
(White Package)

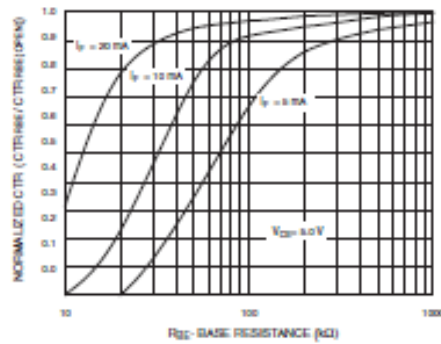


Fig. 9 CTR vs. RBE (Saturated)
(Black Package)

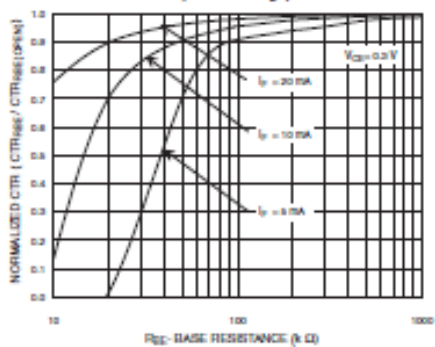


Fig. 10 CTR vs. RBE (Saturated)
(White Package)

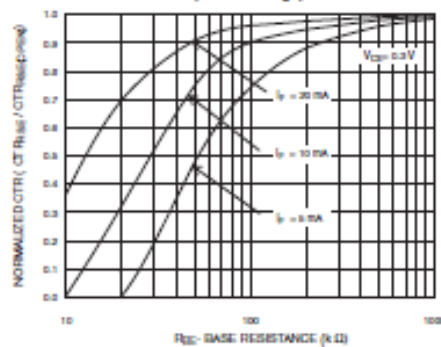


Fig. 11 Collector-Emitter Saturation Voltage vs. Collector Current
(Black Package)

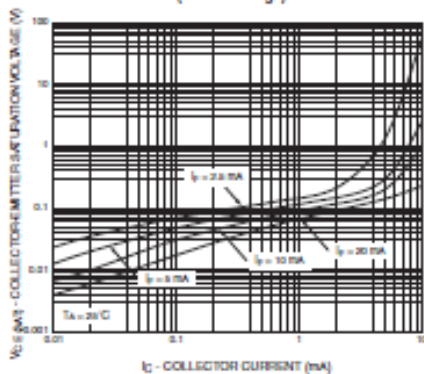
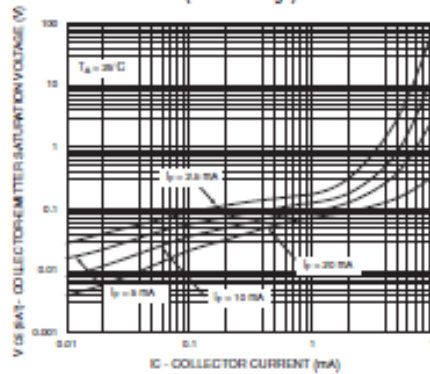


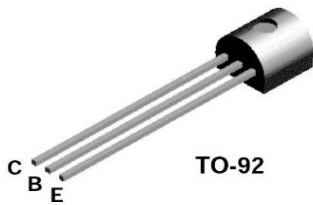
Fig. 12 Collector-Emitter Saturation Voltage vs. Collector Current
(White Package)



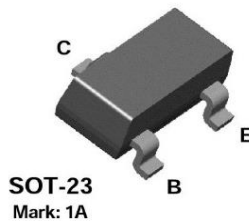
Anexo G Transistor 2N3904



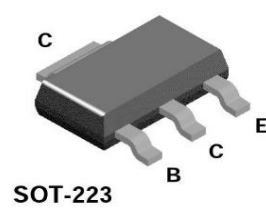
2N3904



MMBT3904



PZT3904



2N3904 / MMBT3904 / PZT3904

NPN General Purpose Amplifier

This device is designed as a general purpose amplifier and switch. The useful dynamic range extends to 100 mA as a switch and to 100 MHz as an amplifier.

Absolute Maximum Ratings* T_A = 25°C unless otherwise noted

Symbol	Parameter	Value	Units
V _{CEO}	Collector-Emitter Voltage	40	V
V _{CBO}	Collector-Base Voltage	60	V
V _{EBO}	Emitter-Base Voltage	6.0	V
I _C	Collector Current - Continuous	200	mA
T _J , T _{stg}	Operating and Storage Junction Temperature Range	-55 to +150	C

*These ratings are limiting values above which the serviceability of any semiconductor device may be impaired.

NOTES:

- 1) These ratings are based on a maximum junction temperature of 150 degrees C.
- 2) These are steady state limits. The factory should be consulted on applications involving pulsed or low duty cycle operations.

Thermal Characteristics T_A = 25°C unless otherwise noted

Symbol	Characteristic	Max			Units
		2N3904	*MMBT3904	**PZT3904	
P _D	Total Device Dissipation	625	350	1,000	mW
	Derate above 25 C	5.0	2.8	8.0	mW/ C
R _{JC}	Thermal Resistance, Junction to Case	83.3			C/W
R _{JA}	Thermal Resistance, Junction to Ambient	200	357	125	C/W

* Device mounted on FR-4 PCB 1.6" X 1.6" X 0.06."

** Device mounted on FR-4 PCB 36 mm X 18 mm X 1.5 mm; mounting pad for the collector lead min. 6 cm².

NPN General Purpose Amplifier

(continued)

Electrical Characteristics

$T_A = 25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Test Conditions	Min	Max	Units
OFF CHARACTERISTICS					
$V_{(BR)CEO}$	Collector-Emitter Breakdown Voltage	$I_C = 1.0\text{ mA}, I_B = 0$	40		V
$V_{(BR)CBO}$	Collector-Base Breakdown Voltage	$I_C = 10\text{ A}, I_E = 0$	60		V
$V_{(BR)EBO}$	Emitter-Base Breakdown Voltage	$I_E = 10\text{ A}, I_C = 0$	6.0		V
I_{BL}	Base Cutoff Current	$V_{CE} = 30\text{ V}, V_{EB} = 3\text{ V}$		50	nA
I_{CEX}	Collector Cutoff Current	$V_{CE} = 30\text{ V}, V_{EB} = 3\text{ V}$		50	nA

ON CHARACTERISTICS*

h_{FE}	DC Current Gain	$I_C = 0.1\text{ mA}, V_{CE} = 1.0\text{ V}$ $I_C = 1.0\text{ mA}, V_{CE} = 1.0\text{ V}$ $I_C = 10\text{ mA}, V_{CE} = 1.0\text{ V}$ $I_C = 50\text{ mA}, V_{CE} = 1.0\text{ V}$ $I_C = 100\text{ mA}, V_{CE} = 1.0\text{ V}$	40 70 100 60 30	300	
$V_{CE(sat)}$	Collector-Emitter Saturation Voltage	$I_C = 10\text{ mA}, I_B = 1.0\text{ mA}$ $I_C = 50\text{ mA}, I_B = 5.0\text{ mA}$		0.2 0.3	V V
$V_{BE(sat)}$	Base-Emitter Saturation Voltage	$I_C = 10\text{ mA}, I_B = 1.0\text{ mA}$ $I_C = 50\text{ mA}, I_B = 5.0\text{ mA}$	0.65	0.85 0.95	V V

SMALL SIGNAL CHARACTERISTICS

f_T	Current Gain - Bandwidth Product	$I_C = 10\text{ mA}, V_{CE} = 20\text{ V}, f = 100\text{ MHz}$	300		MHz
C_{obo}	Output Capacitance	$V_{CB} = 5.0\text{ V}, I_E = 0, f = 1.0\text{ MHz}$		4.0	pF
C_{ibo}	Input Capacitance	$V_{EB} = 0.5\text{ V}, I_C = 0, f = 1.0\text{ MHz}$		8.0	pF
NF	Noise Figure	$I_C = 100\text{ A}, V_{CE} = 5.0\text{ V}, R_S = 1.0\text{ k}\Omega, f = 10\text{ Hz to } 15.7\text{ kHz}$		5.0	dB

SWITCHING CHARACTERISTICS

t_d	Delay Time	$V_{CC} = 3.0\text{ V}, V_{BE} = 0.5\text{ V}, I_C = 10\text{ mA}, I_{B1} = 1.0\text{ mA}$		35	ns
t_r	Rise Time	$V_{CC} = 3.0\text{ V}, I_C = 10\text{ mA}, I_{B1} = 1.0\text{ mA}$		35	ns
t_s	Storage Time	$V_{CC} = 3.0\text{ V}, I_C = 10\text{ mA}, I_{B1} = I_{B2} = 1.0\text{ mA}$		200	ns
t_f	Fall Time	$I_{B1} = I_{B2} = 1.0\text{ mA}$		50	ns

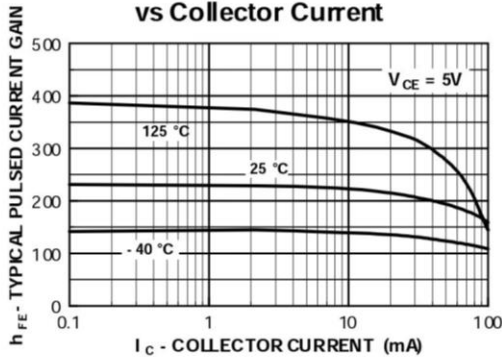
*Pulse Test: Pulse Width 300 μ s, Duty Cycle 2.0%

Spice Model

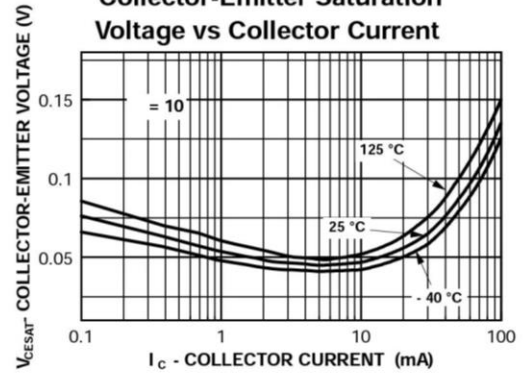
NPN (Is=6.734f Xti=3 Eg=1.11 Vaf=74.03 Bf=416.4 Ne=1.259 Ise=6.734 Ikf=66.78m Xtb=1.5 Br=.7371 Nc=2 Isc=0 Ikr=0 Rc=1 Cjc=3.638p Mjc=.3085 Vjc=.75 Fc=.5 Cje=4.493p Mje=.2593 Vje=.75 Tr=239.5n Tf=301.2p Itf=.4 Vtf=4 Xtf=2 Rb=10)

Typical Characteristics

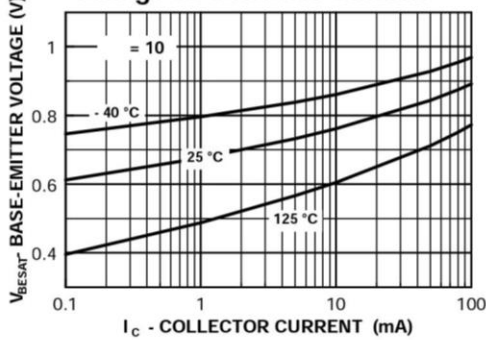
Typical Pulsed Current Gain vs Collector Current



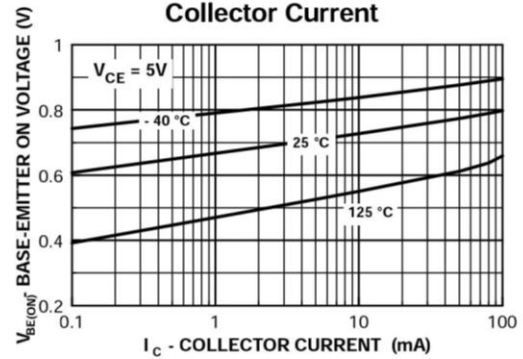
Collector-Emitter Saturation Voltage vs Collector Current



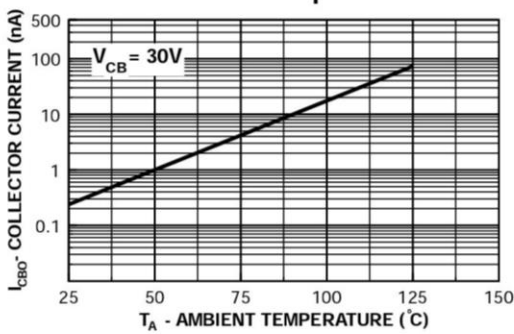
Base-Emitter Saturation Voltage vs Collector Current



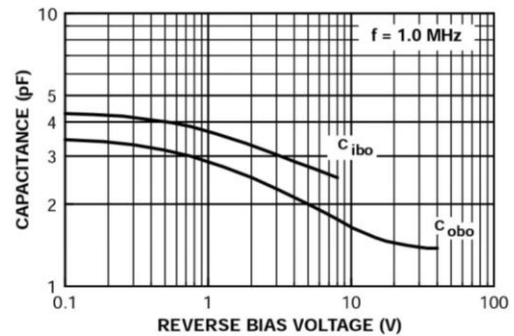
Base-Emitter ON Voltage vs Collector Current



Collector-Cutoff Current vs Ambient Temperature

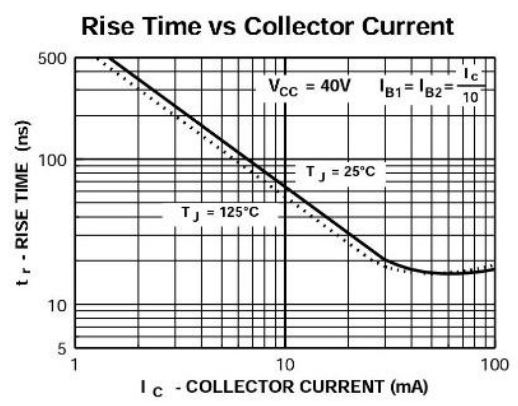
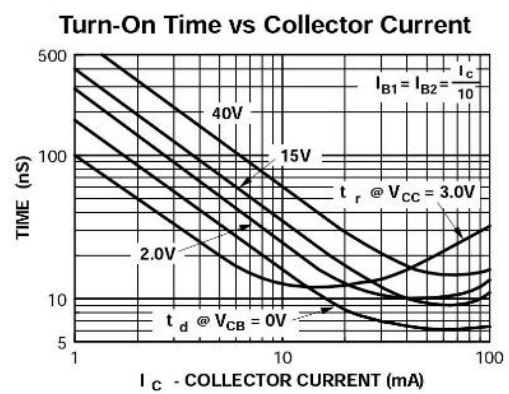
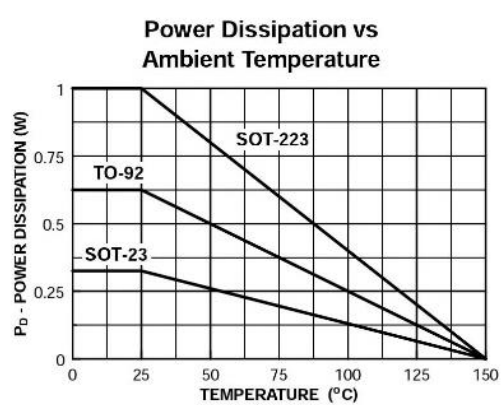
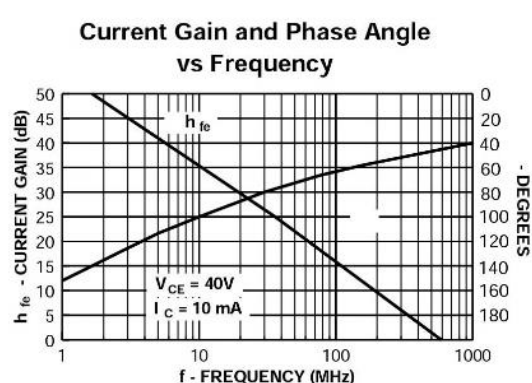
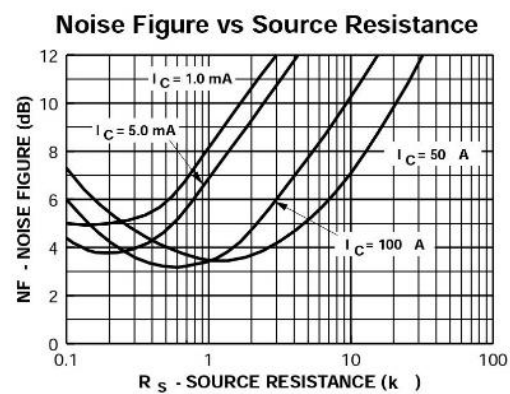
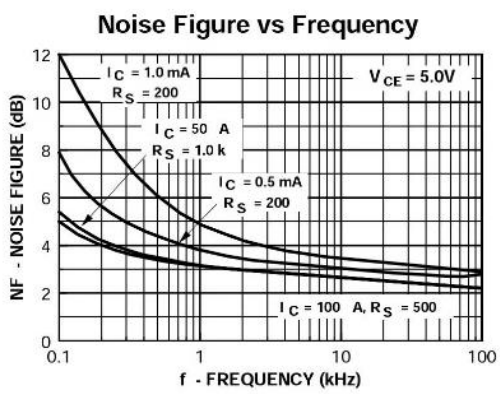


Capacitance vs Reverse Bias Voltage



NPN General Purpose Amplifier
(continued)

Typical Characteristics (continued)



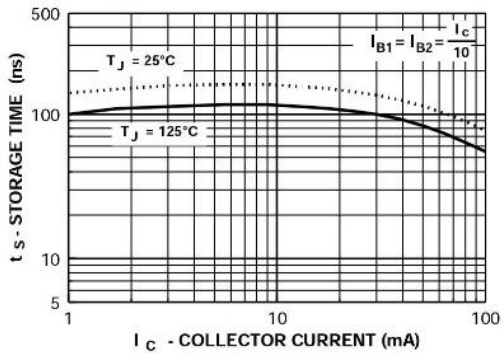
NPN General Purpose Amplifier

(continued)

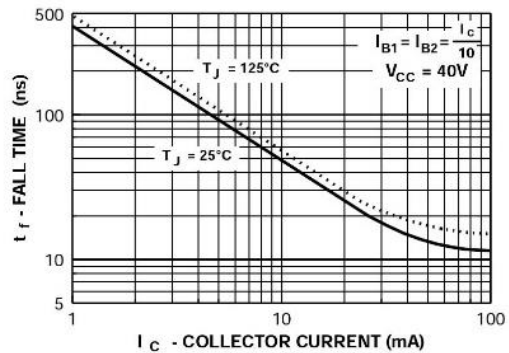
2N3904 / MMBT3904 / PZT3904

Typical Characteristics (continued)

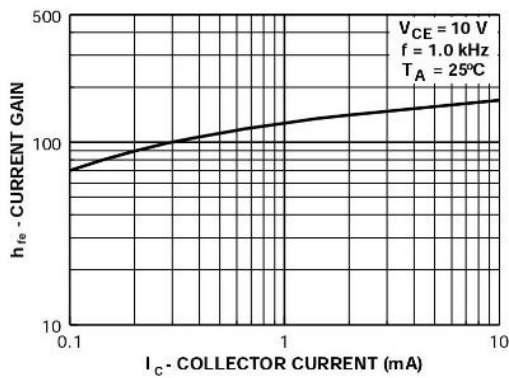
Storage Time vs Collector Current



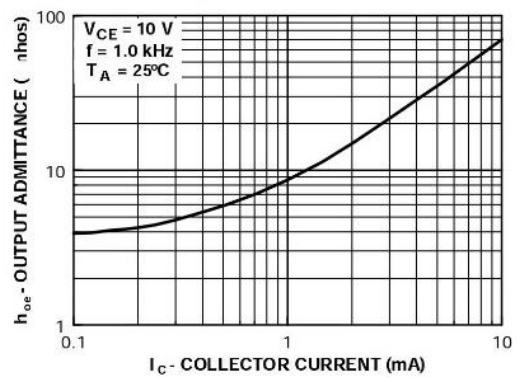
Fall Time vs Collector Current



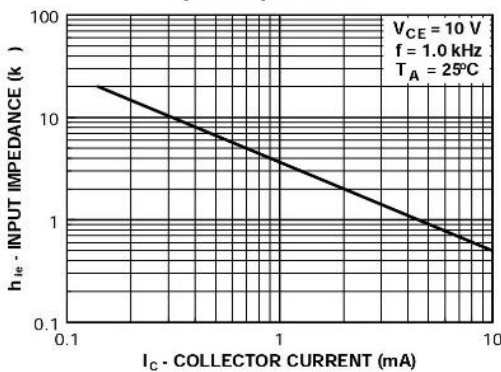
Current Gain



Output Admittance



Input Impedance



Voltage Feedback Ratio

