

UNIVERSIDAD DEL VALLE DE GUATEMALA

Facultad de Ingeniería



Sistema de localización y rastreo por medio de alarma de pánico personal

Trabajo de graduación presentado por Víctor Alberto Vargas Contreras
para optar al grado académico de Licenciado en Ingeniería en
Electrónica

Guatemala

2013

Sistema de localización y rastreo por medio de alarma de pánico personal

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería




Sistema de localización y rastreo por medio de alarma de pánico personal

Trabajo de graduación presentado por Víctor Alberto Vargas Contreras
para optar al grado académico de Licenciado en Ingeniería en
Electrónica


Guatemala
2013

Vo. Bo. :

(f) 

(Ing. Manuel Cotero)

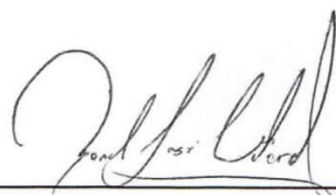
Tribunal Examinador:

(f) 

(Ing. Carlos Esquit)

(f) 

(Ing. Pedro Romero)

(f) 

(Ing. Manuel Cotero)

Fecha de aprobación: **Guatemala, 3 de diciembre de 2013**

ÍNDICE

CAPÍTULOS	PÁGINA
ÍNDICE	vi
LISTA DE FIGURAS	xiii
LISTA DE TABLAS	xviii
RESUMEN.....	xix
I. INTRODUCCIÓN.....	1
II. DIAGRAMA DE BLOQUES DEL PROYECTO.....	2
III. JUSTIFICACIÓN.....	3
IV. OBJETIVOS	4
A. OBJETIVO GENERAL	4
B. OBJETIVOS ESPECÍFICOS	4
V. MARCO TEÓRICO	5
A. BASE DE DATOS (INFORMÁTICA).....	5
1. DEFINICIÓN.....	5
2. SISTEMAS GESTORES DE BASES DE DATOS	5
3. LENGUAJE DE CONSULTA ESTRUCTURADO.	5
4. ESTRUCTURA DE UNA BASE DATOS	6
B. DISEÑO DE PÁGINAS WEB.....	7
1. DEFINICIÓN	7
2. PROTOCOLO HTTP.....	7
3. LENGUAJE HTML.	8
4. LENGUAJE PHP.	9
5. LENGUAJE JAVASCRIPT.....	10

6. CARPETAS EN PÁGINAS WEB.....	11
C. REDES DE TELEFONÍA INALÁMBRICA.....	11
1. FUNCIONAMIENTO GENERAL.....	11
2. SISTEMA GLOBAL PARA LAS COMUNICACIONES MÓVILES.....	13
3. MÓDULOS DE COMUNICACIÓN CON LA RED CELULAR.....	17
D. COMANDOS HAYES	18
1. COMANDO AT	19
2. COMANDO AT+CGMI.....	19
3. COMANDO ATE0.....	19
4. COMANDO AT+CMEE.....	19
5. COMANDO AT+CGDCONT.....	20
6. COMANDO AT#SCFG.....	20
E. SISTEMAS M2M, MACHINE TO MACHINE.....	21
F. MICROCONTROLADOR	21
1. DEFINICIÓN.....	21
2. PROGRAMACIÓN.....	22
3. INTERRUPCIÓN.....	26
4. COMUNICACIÓN SERIAL.....	26
5. CONTADORES INTERNOS.....	27
6. ESTADO SLEEP.....	27
G. SISTEMAS DE POSICIONAMIENTO GLOBAL.....	29
1. GLOBAL POSITIONING SYSTEM (GPS).....	29
2. GLOBAL'NAYA NAVIGATSIONNAYA SPUTNIKOVAYA SISTEMA (GLONASS).....	29
3. GALILEO.....	29
4. BEIDOU.....	29
5. FUNCIONAMIENTO GENERAL DE LOS SISTEMAS DE POSICIONAMIENTO GLOBAL.....	30

H.	MULTIVIBRADORES	31
1.	CONFIGURACIÓN ASTABLE	31
2.	CONFIGURACIÓN MONOESTABLE.....	31
3.	CONFIGURACIÓN BIESTABLE	32
4.	CIRCUITO INTEGRADO 555.	32
I.	REGULADOR DE VOLTAJE.....	33
1.	DROPOUT VOLTAJE	34
J.	DEFINICIÓN GEOFENCE.....	35
K.	FÓRMULA DE HAVERSINE	35
L.	HANDSHAKE ENTRE SISTEMAS DE COMUNICACIONES	36
VI.	DIÁLOGO GENERAL ENTRE SESIÓN DE DATOS Y SERVIDOR	38
A.	VARIABLE 'ORG'	38
1.	PARÁMETRO REQUEST EN BASE DE DATOS.....	39
2.	MENSAJES DE RESPUESTA DEL SERVIDOR.....	40
3.	EJEMPLO DEL FLUJO DE DATOS.....	40
VII.	ESTRUCTURA Y MODOS DE FUNCIONAMIENTO.....	45
A.	FUNCIONAMIENTO EN MODO DE RASTREO.....	46
1.	COMPORTAMIENTO DEL DISPOSITIVO.....	46
2.	COMPORTAMIENTO PLATAFORMA WEB.....	47
B.	FUNCIONAMIENTO EN MODO ALARMA	48
1.	COMPORTAMIENTO DEL DISPOSITIVO.....	48
2.	COMPORTAMIENTO PLATAFORMA WEB.....	50
C.	FUNCIONAMIENTO DE RASTREO MÁS ALARMA	51
1.	COMPORTAMIENTO DEL DISPOSITIVO.	51
2.	COMPORTAMIENTO PLATAFORMA WEB.....	52
D.	FUNCIONAMIENTO EN GEOFENCE	54
E.	FUNCIONAMIENTO EN PETICIÓN	55

VIII. COMUNICACIÓN Y ACCESO A LA RED CELULAR.....	56
A. DISEÑO	56
1. HANDSHAKE ENTRE DISPOSITIVO GSM Y RED CELULAR.....	56
2. HANDSHAKE ENTRE SESIÓN DE DATOS DEL MÓDULO Y RED CELULAR....	56
3. ENVÍO DE DATOS POR MEDIO DEL PROTOCOLO TCP.....	57
4. VERIFICACIÓN DE LOS COMANDOS ENVIADOS Y RECIBIDOS.....	58
B. RESULTADOS.....	59
1. HANDSHAKE ENTRE MÓDULO GSM Y RED CELULAR.....	59
2. HANDSHAKE ENTRE SESIÓN DE DATOS DEL MÓDULO Y RED CELULAR.	60
3. ENVÍO DE DATOS POR MEDIO DEL PROTOCOLO TCP.....	61
C. DISCUSIÓN.....	61
IX. PROGRAMACIÓN DEL MICROCONTROLADOR 18F25K22.....	63
A. DISEÑO	63
1. CÓDIGO FUENTE BASE.....	63
2. CONFIGURACIÓN DEL RELOJ INTERNO.	63
3. COMUNICACIÓN SERIAL.....	63
4. RUTINAS RECURSIVAS.....	64
5. INTERRUPCIONES.....	69
6. ACTIVACIÓN DE PORTC PARA RELÉS DE ESTADO SÓLIDO.....	70
B. DISCUSIÓN	71
X. MÓDULO DE POSICIONAMIENTO GLOBAL GPS	73
A. DISEÑO	73
B. RESULTADOS.....	74
C. DISCUSIÓN	74
XI. PLATAFORMA WEB	76
A. DISEÑO	76
1. LENGUAJES DE PROGRAMACIÓN UTILIZADOS	76

2. ALOJAMIENTO DE LA PÁGINA.....	77
3. PROGRAMACIÓN DE LA INTERFAZ.....	77
4. ENVÍO DE DATOS ENTRE CARPETAS.	78
5. INTERFAZ CON LA BASE DE DATOS.....	79
6. CORREOS DE ALERTA.....	80
B. RESULTADOS.....	81
1. ALOJAMIENTO DE LA PLATAFORMA WEB.	81
2. SERVICIO MARKER.	82
3. SERVICIO PATH.	83
4. SERVICIO GEOFENCE.....	84
5. SERVICIO REQUEST.	85
6. CORREOS DE ALERTA.....	86
C. DISCUSIÓN.....	86
XII. SUB MÓDULOS DE SERVICIO	88
A. DISEÑO	88
1. ENCENDIDO Y APAGADO DE LOS MÓDULOS GPS Y GSM.....	88
2. SEGUIDORES DE VOLTAJE.	88
3. OSCILADOR ASTABLE 555.....	89
B. RESULTADOS.....	90
1. RELÉ DE ESTADO SÓLIDO	90
2. NEGADORES CON TRANSISTORES	91
3. OSCILADOR ASTABLE 555.....	91
C. DISCUSIÓN.....	92
XIII. MÓDULO DE POTENCIA.....	93
A. CORRIENTE CONSUMIDA POR EL DISPOSITIVO	93
B. DISEÑO	96
1. ELECCIÓN DE LA BATERÍA SUMINISTRADORA.	96

2. ELECCIÓN DE REGULADOR DE VOLTAJE.....	97
3. CARGADOR DE BATERÍA.....	97
4. TIEMPO DE FUNCIONAMIENTO.....	98
C. RESULTADOS.....	98
1. BATERÍA Y REGULADOR SELECCIONADOS.....	98
2. CARGADOR DE BATERÍA.....	99
3. TIEMPO DE FUNCIONAMIENTO.....	99
D. DISCUSIÓN.....	99
XIV. ELABORACIÓN DEL PCB.....	101
A. DISEÑO.....	101
1. CIRCUITO ESQUEMÁTICO.....	101
2. FORMA DEL PCB.....	104
3. CREACIÓN DEL PCB EN ALTIUM.....	105
4. FABRICACIÓN DEL PCB.....	106
B. RESULTADOS.....	106
1. COMPONENTES REMOVIDOS DE LA PLATAFORMA GSM CLICK.....	106
2. PCB CON COMPONENTES.....	107
C. DISCUSIÓN.....	108
XV. ADAPTACIÓN DEL MÓDULO A PRENDA DE VESTIR.....	110
A. DISEÑO.....	110
B. RESULTADOS.....	110
1. CONECTORES TIPO HEADER EN PCB Y MÓDULOS.....	110
XVI. PRUEBA DE CAMPO EN MODO RASTREO MÁS ALARMA.....	116
A. DISEÑO.....	116
B. RESULTADOS.....	116
C. DISCUSIÓN.....	118
XVII. CONCLUSIONES Y RECOMENDACIONES.....	119

A. CONCLUSIONES	119
B. RECOMENDACIÓN	119
XVIII. BIBLIOGRAFÍA.....	121
XIX. ANEXOS.....	124
A. CÓDIGO MICROCONTROLADOR Y PLATAFORMA WEB	124

LISTA DE FIGURAS

Figura	Página
Figura 1. Ejemplo de lenguaje estructurado	6
Figura 2. Ejemplo de una tabla con varias filas y columnas	6
Figura 3. Formato básico para método POST	8
Figura 4. Ejemplo de despliegue básico en HTML	9
Figura 5. Ejemplo de despliegue en PHP	9
Figura 6. Formularios básicos en PHP	10
Figura 7. Cuadrícula celular en diferentes zonas	12
Figura 8. Diagrama de bloques de la red celular	14
Figura 9. Intercambio y flujo de datos en una red telefónica	15
Figura 10. Representación de generación GSM y sus velocidades de operación	16
Figura 11. Circuito integrado TELIT GL865	17
Figura 12. GSM click	18
Figura 13. Sub módulos de un microcontrolador	22
Figura 14. Estructura de los niveles de programación	22
Figura 15. Ejemplo de código en lenguaje ensamblador	23
Figura 16. Ejemplo de código en lenguaje de alto nivel.....	23
Figura 17. Plataforma de desarrollo MikroC	24
Figura 18. Plataforma de desarrollo EasyPic7 de MikroE	25
Figura 19. Comunicación entre dos terminales y trama de datos serial	26
Figura 20. Secuencia de activación y desactivación al ejecutar la instrucción Sleep	28
Figura 21. Trilateración utilizando círculos	30
Figura 22. Representación de un multivibrador astable	31
Figura 23. Representación de un multivibrador monoestable	32
Figura 24. Respuesta de un oscilador biestable	32

Figura 25. Circuito para un oscilador 555 y sus cálculos respectivos	33
Figura 26. Representación de un regulador de voltaje	34
Figura 27. Representación de Geofence en 4 puntos	35
Figura 28. Distancias en una esfera	36
Figura 29. Handshake para una sesión TCP convencional	37
Figura 30. Diagrama de flujo para nivel dispositivo -servidor.....	41
Figura 31. Diagrama de flujo del dispositivo en reacción a los mensajes de respuesta del servidor	43
Figura 32. Interrupciones posibles para despertar al microcontrolador y origen de estas	45
Figura 33. Diagrama de flujo del dispositivo para el modo de rastreo	47
Figura 34. Diagrama de flujo de plataforma web en modo de rastreo	48
Figura 35. Diagrama de flujo del dispositivo en modo alarma	49
Figura 36. Diagrama de flujo de plataforma web en modo alarma.....	50
Figura 37. Diagrama de flujo del dispositivo en modo rastreo mas alarma.....	52
Figura 38. Diagrama de flujo de plataforma web en modo de rastreo más alarma	53
Figura 39. Diagrama de flujo de la plataforma web en modo Geofence	54
Figura 40. Diagrama de flujo de la plataforma web en modo petición	55
Figura 41. Handshake inicial entre módulo GSM y red de telefonía.	56
Figura 42. Diagrama de intercambio de información para configurar la sesión de datos en el módulo GSM	57
Figura 43. Diagrama de intercambio de información para el envío de datos y respuesta del servidor.	58
Figura 44. Interfaz de hyperterminal en MikroC.....	59
Figura 45. Comando enviado por el microcontrolador a la red celular.....	59
Figura 46. Comandos recibidos por la red celular	60
Figura 47. Comandos para la configuración de sesión de datos	60
Figura 48. Activación del contexto en el módulo GSM	60
Figura 49. Comandos del protocolo HTTP para el envío de datos mediante POST	61

Figura 50. Respuesta del servidor al mensaje HTTP POST	61
Figura 51. Inicialización del puerto serial 1	64
Figura 52. Inicialización del puerto serial 2.....	64
Figura 53. Diagrama de flujo general del microcontrolador e identificación de las fuentes de interrupción	65
Figura 54. Identificación de la fuente de interrupción en la sección principal.....	65
Figura 55. Comandos AT para el acceso a la red celular	66
Figura 56. Comandos AT para configuración de la sesión de datos en el módulo GSM..	66
Figura 57. Arrays contenedores de la estructura del método POST en el microcontrolador..	67
Figura 58. Método de envío de caracteres por el puerto serial.....	67
Figura 59. Rutina del GPS para obtener las coordenadas geográficas	68
Figura 60. Configuración para la interrupción del puerto B.....	69
Figura 61. Configuración para el Timer1	70
Figura 62. Configuración para las interrupciones	70
Figura 63. Activación de módulos y sub módulos mediante PORTC.....	70
Figura 64. Modulo GPS LS20031 utilizado.....	74
Figura 65. Formato NMEA obtenido del módulo LS20031	74
Figura 66. Flujo de datos entre carpetas	79
Figura 67. Página principal de la plataforma web.....	81
Figura 68. Opciones para el despliegue de la opción Marker	82
Figura 69. Interfaz para la opción Marker.....	82
Figura 70. Opciones para el despliegue Path.....	83
Figura 71. Interfaz para la opción Path	84
Figura 72. Opciones para el despliegue de la opción Geofence.....	84
Figura 73. Interfaz de la opción Geofence.....	85
Figura 74. Opciones para el despliegue de Request.....	85
Figura 75. Despliegue para la interfaz de Request.....	86

Figura 76. Correo de alerta enviado en violación a Geofence	86
Figura 77. Diagrama esquemático para oscilador 555 y valores de componentes utilizando SS555	89
Figura 78. Circuito simulado en LiveWire	90
Figura 79. Relés de estado sólido en PCB de prueba	90
Figura 80. Configuración doble negador en PCB de prueba	91
Figura 81. Simulación del oscilador con los valores establecidos por SS555.....	91
Figura 82. Consumo de corriente en modo de petición	94
Figura 83. Batería seleccionada para alimentar al dispositivo	98
Figura 84. Regulador seleccionado en PCB de prueba.....	98
Figura 85. Esquemático para el oscilador astable TS555.....	101
Figura 86. Esquemático para configuración doble negador con transistores y resistencias.	102
Figura 87. Esquemático para el regulador de voltaje seleccionado.....	102
Figura 88. Esquemático para la configuración de los relés de estado sólido	103
Figura 89. Esquemático para el botón que acciona la interrupción PORTB	103
Figura 90. Esquemático para el microcontrolador 18f25k22.....	104
Figura 91. Forma del PCB basado en módulo GSM Click	105
Figura 92. Diagrama del PCB final con todos sus componentes de placa.....	105
Figura 93. Módulos GSM, parte posterior con y sin modificaciones	106
Figura 94. Módulos GSM, parte frontal con y sin modificaciones	107
Figura 95. Parte superior del dispositivo	107
Figura 96. Parte inferior del dispositivo	108
Figura 97. Conectores header de tres posiciones macho y hembra	110
Figura 98. Módulo GPS con conector header de tres posiciones	111
Figura 99. Batería con conector header de dos posiciones	111
Figura 100. Conector tipo header de dos posiciones, conexión entre PCB y batería.....	112
Figura 101. Abrigo utilizado para instalar el dispositivo	112

Figura 102. Bolsa alojadora de PCB y batería.....	112
Figura 103. Bolsa alojadora de módulo GPS	113
Figura 104. PCB y batería instalados en abrigo	113
Figura 105. PCB y batería instalados en abrigo	113
Figura 106. Módulo GPS instalado en el abrigo	114
Figura 107. Dispositivo oculto completamente instalado en abrigo	114
Figura 108. Dispositivo visible instalado en abrigo.	115
Figura 109. Posiciones encontradas en la prueba de campo	117
Figura 110. Vista satelital última ubicación encontrada.	117

LISTA DE TABLAS

Tabla		Página
Tabla No.1.	Variables utilizadas por el módulo para ingresar datos	38
Tabla No.2.	Definición de estados para la variable 'org'	39
Tabla No.3.	Estado de variable REQUEST y acciones que se realizan.....	39
Tabla No.4.	Respuestas del servidor al dispositivo.....	40
Tabla No.5.	Servicios disponibles y carpeta a la que se re dirigido	79
Tabla No.6.	Parámetros para la tabla Usuarios	80
Tabla No.7.	Parámetros para la tabla UBICA	80
Tabla No.8.	Cuerpo del correo enviado en modo alarma.....	81
Tabla No.9.	Corriente consumida por los módulos principales y sub módulos.....	93
Tabla No.10.	Módulos y sub módulos alimentados en todo momento	94

Resumen

El siguiente trabajo de consiste en diseñar y construir un dispositivo el cual pueda darle seguimiento a la persona que lo esté portando. Este seguimiento puede ser activado periódicamente o manualmente para indicar si la persona se encuentra en peligro.

Se lograra el funcionamiento anteriormente descrito implementado dos grandes tecnologías. Una de ellas es el sistema de posicionamiento global (GPS). Esta estará en funcionamiento en conjunto con el enlace entre el objeto y el usuario por medio de las redes de telefonía inalámbrica para él envío de información tanto de ubicación como de variables.

Los datos proporcionados por el GPS serán alojados en una base de datos para que mediante una interfaz web puedan desplegarse los datos con diferentes formas de visualización.

Por ser de índole portable, el dispositivo puede ser fácilmente adaptado no solo a personas sino también a vehículos, mercancías, paquetes, mascotas y demás posesiones de valor.

I. Introducción

El siguiente trabajo de graduación discute como fue diseñado y construido un dispositivo portable de localización personal y de objetos.

La forma en que esta tarea se llevó a cabo fue utilizando las redes de telefonía celular, módulos de posicionamiento global (GPS) y microcontroladores como sistemas principales.

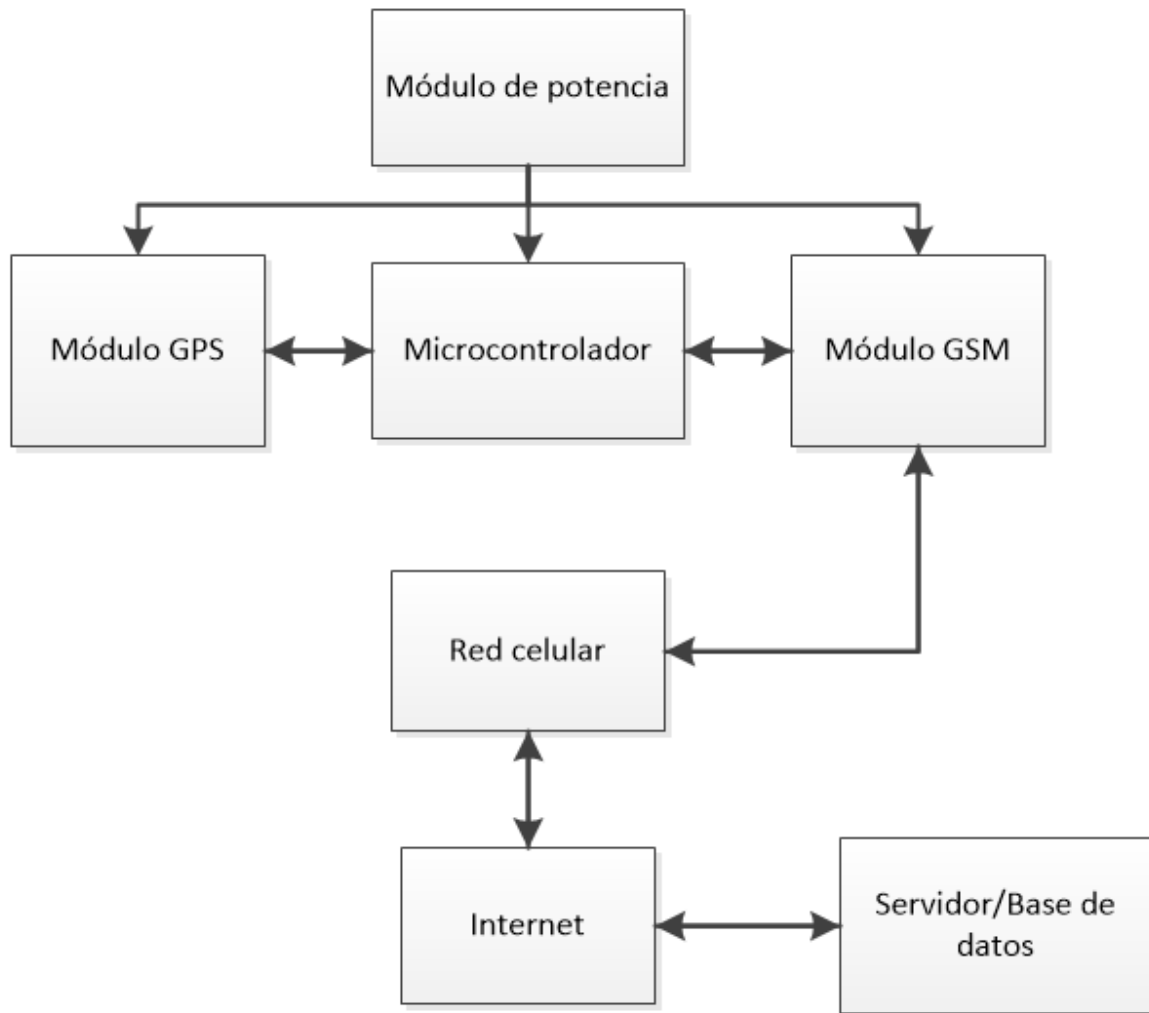
La síntesis del tráfico de información empieza con los datos proporcionados por el GPS que son enviados mediante sesiones de datos a un servidor remoto en cual utilizando herramientas propias de gestores de bases de datos, toma la información la cual será almacenada para su posterior despliegue a través de una plataforma web.

Esta plataforma web contiene herramientas esenciales para realizar funciones de localización funcionando en sincronía con el dispositivo móvil para establecer la frecuencia de datos enviados.

Se discute cómo fue posible la programación con el microcontrolador así como este influye en la comunicación con la red celular y los módulos de posicionamiento. Además como fue la adaptación de este dispositivo a una prenda de vestir para volverlo oculto y discreto.

Entre las conclusiones más grandes que se obtuvo fue que se logró establecer el vínculo entre los sistemas mencionados logrando una sinergia total entre hardware y software.

II. Diagrama de bloques del proyecto



III. Justificación

Un impulso para realizar este trabajo de graduación es que en Guatemala no se cuenta con un sistema de rastreo de personas y alerta de pánico formalmente. Además, en el mercado guatemalteco existen revendedores de sistemas similares los cuales cuentan con dos problemas que este proyecto busca resolver. Uno es tener el respaldo en información y hardware en caso los sistemas fallen se puede actuar de formas alternas para encontrar al dispositivo y por ende a la persona de interés. El segundo problema es que estos son de dimensiones notorias que es lo que busca como mejora principal este proyecto, crear un dispositivo el cual pueda pasar desapercibido ante por ejemplo un delincuente disfrazándolo en alguna prenda de vestir y así poder indicar a las personas de interés que se encuentra en peligro y en donde (Coordenadas geográficas) se encuentra. Se tiene además diferentes aplicaciones a las que este dispositivo podría fácilmente ser adaptado no solo en seguridad personal sino también en aplicaciones médicas, industriales y automotrices. Para limitar el trabajo de graduación se enfocó únicamente en localizar personas.

IV. Objetivos

A. Objetivo general

- Crear un sistema de localización y alerta de pánico personal.

B. Objetivos específicos

- Adaptar un sistema en el cual las personas de interés puedan conocer la ubicación de la persona con y sin consentimiento de esta por medio de una plataforma web y celular.
- Utilizar las redes de telefonía celular para el envío de información del dispositivo a las plataformas.
- Utilizar sistemas de posicionamiento global para obtener las coordenadas del dispositivo.
- Crear una plataforma web para visualizar el contenido proporcionado por el módulo de posicionamiento global.

V. Marco teórico

A. Base de datos (informática)

1. Definición. Una base de datos en informática es un espacio que nos permite guardar grandes cantidades de información de un mismo contexto en un formato digital de forma organizada para que luego podamos encontrar y utilizar fácilmente. Esta información es introducida y modificada mediante programas llamados gestores de base de datos que facilitan la manipulación de los datos alojados en ellas.

La información no se guarda como fue introducida, más bien se codifica para obtener aún mayor seguridad en caso los datos fuesen robados.

2. Sistemas gestores de bases de datos. Un Sistema de Gestión de Bases de Datos (SGBD) es un conjunto de programas que permiten el almacenamiento, modificación y extracción de la información en una base de datos, además de proporcionar herramientas para añadir, borrar, modificar y analizar los datos (MaestrosWeb, 2013).

Los Sistemas de Gestión de Base de Datos (en inglés DataBase Management System) son un tipo de software muy específico, dedicado a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan. Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta (MaestrosWeb, 2013).

Entre los principales gestores de bases de datos tenemos a MySQL, PostgreSQL, Oracle DB, Microsoft Access y Microsoft SQL server. Todos estos se basan en el lenguaje Structural Query Language (Lenguaje de consulta estructurado).

3. Lenguaje de consulta estructurado. El lenguaje estructurado de consultas, más comúnmente llamado SQL, lo conforman una serie de comandos, cláusulas y funciones que permiten realizar cualquier operación sobre la información almacenada en la base de datos (Instituto de tecnologías educativas, 2013).

SQL no es exactamente un lenguaje de programación pero lleva implícita la complejidad de estas herramientas. En realidad, los sistemas gestores de bases de datos tienen como principal fin ocultar el lado oscuro de este lenguaje mediante el uso de utilidades gráficas (Instituto de tecnologías educativas, 2013).

Figura 1. Ejemplo de lenguaje estructurado

```
*INSERT INTO `UBICA` (`ID`, `LAT`, `LON`, `MES`, `DIA`, `ANIO`, `HORA`)
VALUES ([value-1],[value-2],[value-3],[value-4],[value-5],[value-6],[value-7])
*SELECT * FROM `UBICA` WHERE `MES` == $MES
*DELETE FROM `UBICA` WHERE `HORA` == 3 && `MES` == 2
```

Observamos en la figura número 1 que el lenguaje estructurado se asemeja mucho a una petición que haríamos en la vida real. No necesitamos de ciclos ni condiciones elegantes para realizar tareas relacionadas con las bases de datos.

4. Estructura de una base datos. Se tienen tres acciones básicas en las bases de datos que permiten realizar todas las funciones para que esta sea operativa.

La primera son las tablas que almacenan los datos en filas y columnas. Todas las bases de datos contienen una o más tablas sin importar su índole.

La segunda son las consultas que recuperan y procesan los datos. En estas especificamos elegantes peticiones mediante el lenguaje estructurado para obtener solo la información deseada.

Por último la tercera es la generación de informes que resumen e imprimen los datos para una visualización más dinámica para de esta forma analizarla.

Figura 2. Ejemplo de una tabla con varias filas y columnas

←T→		ID	LAT	LON	MES	DIA	ANIO	HORA
<input type="checkbox"/>	Edit	14	14.6414	-90.5116	8	29	2013	9
<input type="checkbox"/>	Edit	15	14.6641	-90.5116	8	29	2013	9
<input type="checkbox"/>	Edit	16	14.6408	-90.474	8	29	2013	9
<input type="checkbox"/>	Edit	17	14.6259	-90.5108	8	29	2013	9
<input type="checkbox"/>	Edit	18	14.6259	-90.474	8	29	2013	9

En la figura anterior se tiene un ejemplo de una estructura para una tabla en una base de datos. Esta tabla cuenta con 7 columnas y 5 filas las cuales utilizando el gestor de base de datos podemos modificarlas con las opciones que se muestran del lado izquierdo de la tabla.

B. Diseño de páginas web

1. Definición. Una página web es el nombre de un documento o información electrónica adaptada para la World Wide Web y que puede ser accedida mediante un navegador. Esta información se encuentra generalmente en formato HTML o XHTML. Las páginas web frecuentemente incluyen otros recursos como hojas de estilo en cascada, guiones (scripts) e imágenes digitales, entre otros (Wikipedia, 2013).

Las páginas web más elaboradas no solo se basan en HTML o XHTML sino implementan código incrustado de demás lenguajes de programación web como lo es PHP, JavaScript, CSS, Flash, entre otros. Cada uno con sus funciones específicas para lograr una página web más dinámica.

Las páginas web pueden estar almacenadas en un equipo local o un servidor web remoto. El servidor web puede restringir el acceso únicamente para redes privadas, p. ej., en una intranet corporativa, o puede publicar las páginas en la World Wide Web. El acceso a las páginas web es realizado mediante su transferencia desde servidores utilizando el protocolo de transferencia de hipertexto (HTTP) (Wikipedia, 2013).

2. Protocolo HTTP. El protocolo de transferencia de hipertexto (HyperText Transfer Protocol) es un protocolo de nivel de aplicación usado para la transferencia de información entre sistemas ya sean computadoras o bien dispositivos móviles (Álvarez, 2013).

Para el envío de dicha información utiliza dos métodos principales y mediante ellos es pasada la información entre terminales. Cada uno varía entre otro tanto en la declaración del encabezado así como la ubicación de los datos en el mensaje.

a. Método GET. Es la petición a un recurso específico o bien a la página web completa. Sus datos viajan en el encabezado de la página a solicitar como por ejemplo:

`www.mipagina.com/carpeta?nombre=Jose&edad=34`

Si vemos las variables de nombre y edad son transmitidas a la carpeta llamada 'carpeta' y esta se encargara de interpretar los datos recibidos.

b. Método POST. Es el envío de datos mediante el protocolo donde estos se incluyen en el encabezado del protocolo. Para este método el datos no viajan en el encabezado como lo es el método anterior sino en una estructura más compleja.

Figura 3. Formato básico para método POST

```
POST /carpeta.php HTTP/1.1
Host: mipagina.com
Connection: keep-alive
Content-Length: 13
Origin: http://vicvicvar.com
nombre=Jose&edad=34
```

Su ventaja es que es más difícil ver los datos enviados ya que no están en el encabezado lo cual lo vuelve el método más seguro para enviar contraseñas, usuarios y demás información confidencial.

3. Lenguaje HTML. HTML es el acrónimo de HyperText Markup Language (Lenguaje de Marcado de Hipertexto) y es el lenguaje que se utiliza para crear las páginas web. Este lenguaje indica a los navegadores cómo deben mostrar el contenido de una página web (Masadelante, 2013).

El lenguaje se basa en etiquetas para poder indicar que tipo de elemento se está tratando. Esto lo hace con cada elemento presente para que luego el navegador o interprete HTML pueda desplegarlo conforme el programador diseño la página.

Figura 4. Ejemplo de despliegue básico en HTML

```

<!DOCTYPE html>
<html>
<body>

Titulo 1

<h1>Titulo 1</h1>
<p>Texto de prueba</p> Texto de prueba

</body>
</html>

```

En la figura anterior se hace énfasis a las etiquetas de títulos y párrafos que son los que interpretara el navegador para seleccionar que elemento es. En este ejemplo los interpretes saben que al tener una etiqueta de <h1> es el título principal de la página y por lo tanto tiene mayor relevancia (mayor tamaño y negrilla).

4. Lenguaje PHP. El PHP (acrónimo de PHP: HyperText Preprocessor), es un lenguaje interpretado de alto nivel embebido en páginas HTML y ejecutado en el servidor (MaestrosWeb, 2013).

El PHP es un lenguaje de script incrustado dentro del HTML. La mayor parte de su sintaxis ha sido tomada de C, Java y Perl con algunas características específicas de sí mismo. La meta del lenguaje es permitir rápidamente a los desarrolladores la generación dinámica de páginas (MaestrosWeb, 2013).

Figura 5. Ejemplo de despliegue en PHP

```

<!DOCTYPE html>
<html>
<body>

My first PHP script!

<?php
echo "My first PHP script!";
?>

</body>
</html>

```

Como podemos observar, el código PHP es insertado en código HTML mediante sus etiquetas respectivas y utiliza sintaxis más formal estableciendo terminación de líneas.

a. FORMS en PHP. Las formas o como mejor se conocen FORMS son formularios que una carpeta de la página web envía para que otra página la reciba y procese estos datos. Si tenemos por ejemplo una página que vende artículos de ropa vamos a tener un buscador. En este buscador especificamos más a detalle que es lo que buscamos. Supongamos que buscamos zapatos.

Una vez especificamos presionamos el botón de enviar y este envía la variable “zapatos” a otra carpeta la cual es la encargada de buscar en la base de datos que zapatos son los que tiene disponible.

Figura 6. Formularios básicos en PHP

```
<html>
<body>

<form action="buscador.php" method="POST">
Articulo: <input type="text" name="articulo"><br>
Color: <input type="text" name="color"><br>
<input type="submit">
</form>

</body>
</html>
```

En la imagen anterior vemos que los formularios en php están incrustados en código HTML para hacer la página más dinámica. En la primera línea después de la etiqueta body tenemos la acción la cual le decimos a que archivo fuente nos referimos. Continuo especificamos el método con el que se envía el cual es POST.

La siguientes dos líneas especifican las variables que se enviaran a la carpeta /buscador.php/. En las últimas dos líneas especificamos el tipo entrada y cerramos el formulario.

5. Lenguaje JavaScript. JAVA Script es un lenguaje interpretado, multiplataforma, orientado a eventos con manejo de objetos, cuyo código se incluye directamente en el mismo documento, usado para el desarrollo de aplicaciones cliente-servidor en páginas HTML (Risco, 2013).

JavaScript no debe confundirse con el lenguaje de programación Java. Este último es un lenguaje de programación orientado a objetos más robusto mientras que JavaScript es un código alojado en un servidor el cual es interpretado y ejecutado por los navegadores web. Se tiene la ventaja con JavaScript de crear páginas dinámicas las cuales interactúan con el usuario reaccionando a eventos como posición del mouse en la pantalla, clicks a ciertas áreas, contadores para refrescar la página con nuevo contenido, entre otros.

6. Carpetas en páginas web. Las carpetas son las extensiones a los archivos de una página web en específico. Estas refieren a nuevo código el cual responderá diferente al código principal. Para ver más claras a las carpetas consideremos la siguiente dirección web:

www.mipagina.com

El código de esta página estará alojado en el archivo “/index/” de nuestro servidor. Pero solo un archivo de código no es suficiente para la capacidad que tienen las páginas web por lo que necesitamos que existan carpetas para direccionarnos a nuevo código.

Como por ejemplo queremos ir a la sección de autenticación de www.mipagina.com lo haríamos de la siguiente manera:

www.mipagina.com/login.php

Entonces el navegador de internet solicitará el archivo “/login.php/” el cual contiene código adecuado para la autenticación de personas.

C. Redes de telefonía inalámbrica

El término se refiere a las comunicaciones entre las terminales móviles (comúnmente llamados celulares) y las redes de telefonía móvil.

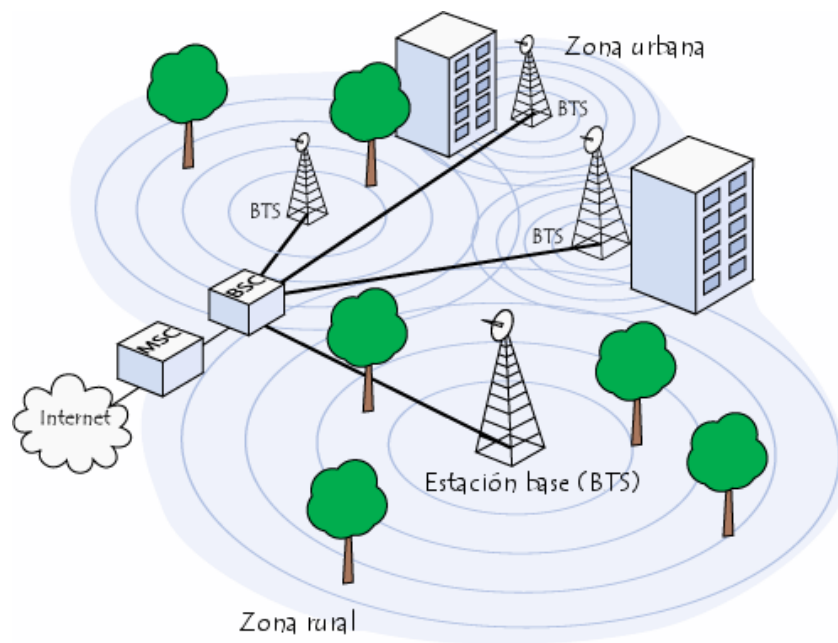
1. Funcionamiento general. Los teléfonos móviles funcionan enviando y recibiendo señales de radio de baja potencia. Las señales se intercambian con antenas que están conectadas a transmisores y receptores de radio, comúnmente conocidos como

estaciones base de telefonía móvil. Las estaciones base están conectadas al resto de las redes de telefonía fija y móvil y pasan la señal (conteniendo mensajes, llamadas, multimedia, etc.) a esas redes (Emfexplained, 2013).

Una red móvil generalmente se diseña en base a una “cuadrícula celular” que cubre un área geográfica. Las estaciones base se colocan ya sea en el centro de cada celda o en el vértice de un grupo de celdas. La cantidad de estaciones base requeridas para un área dada dependerá del terreno y la cantidad de personas que utilicen teléfonos móviles (Emfexplained, 2013).

a. Cuadrícula celular o red de celdas. Es la unidad geográfica básica del sistema de telefonía celular. Es el área cubierta por una estación base o por un subsistema o sector de antena de esa estación base (Servidor Alicante, 2013).

Figura 7. Cuadrícula celular en diferentes zonas.



(Kioskea, 2013)

En la figura anterior vemos cómo cambia la cantidad de radio bases en zonas urbanas. Esto debido a que tendremos a más potenciales usuarios que en un rural donde la densidad de población es menor.

b. Radio base. Una estación base (en inglés: Base Transceiver Station (BTS)) dispone de equipos transmisores/receptores de radio, en la banda de frecuencias de uso (850 / 900 / 1800 / 1900 MHz) en GSM y (1900/2100Mhz) en UMTS que son quienes realizan el enlace con el usuario que efectúa o recibe la llamada (o el mensaje) con un teléfono móvil. Las antenas utilizadas son exclusivas para amplificar estas señales y se sitúan en lo más alto de nuestra radio base (Kioskea, 2013).

La estación base dispone de algún medio de transmisión, vía radio o cable, para efectuar el enlace con la Central de Conmutación de Telefonía Móvil Automática, que a su vez encamina la llamada hacia el teléfono destino, sea fijo o móvil.

En zonas densamente pobladas hay muchas estaciones base, próximas entre sí (células pequeñas) para suplir la demanda de usuarios hacia una potencial conexión de muchas terminales conectadas a la vez. En cambio, en las zonas de baja densidad las radio bases están alejadas unas de otras y transmiten a elevada potencia para asegurar la cobertura en una célula extensa ya que no se esperan demasiadas conexiones (Kioskea, 2013).

2. Sistema global para las comunicaciones móviles. GSM, o Sistema Global para las telecomunicaciones móviles es un sistema estándar completamente definido, usado para la comunicación entre teléfonos móviles basada en la tecnología digital. Permite que cualquier usuario pueda conectarse a través del teléfono a su PC personal, permitiéndole interactuar por e-mail, fax, acceder a Internet, y un acceso seguro a redes LAN o Intranet. También existe la posibilidad de envío de texto corto entre terminales (SMS) (Wikipedia, 2013).

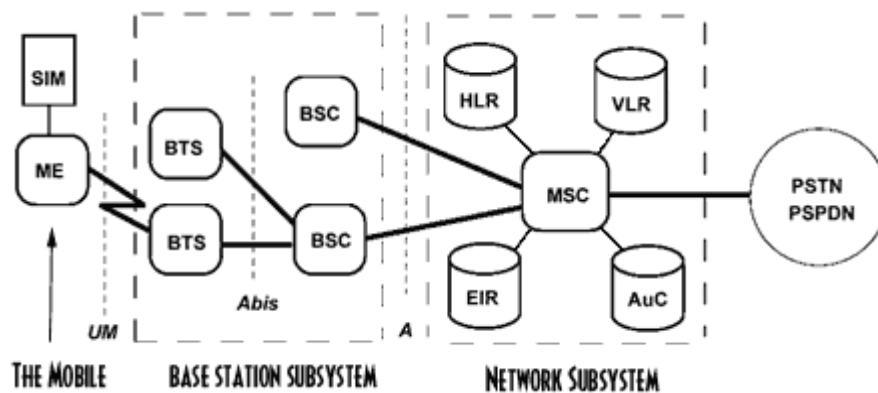
a. Arquitectura de la red GSM. En una red GSM, la terminal del usuario se llama estación móvil. Una estación móvil está constituida por una tarjeta SIM (Módulo de identificación de abonado), que permite identificar de manera única al usuario y a la terminal móvil, o sea, al dispositivo del usuario (normalmente un teléfono portátil) (Ingeniatic, 2013).

Las terminales (dispositivos) se identifican por medio de un número único de identificación de 15 dígitos denominado IMEI (Identificador internacional de equipos

móviles). Cada tarjeta SIM posee un número de identificación único (y secreto) denominado IMSI (Identificador internacional de abonados móviles). Este código se puede proteger con una clave de 4 dígitos llamada código PIN (Ingeniatic, 2013).

Por lo tanto, la tarjeta SIM permite identificar a cada usuario independientemente de la terminal utilizada durante la comunicación con la estación base. Las comunicaciones entre una estación móvil y una estación base se producen a través de un vínculo de radio, por lo general denominado el espacio libre (Ingeniatic, 2013).

Figura 8. Diagrama de bloques de la red celular.

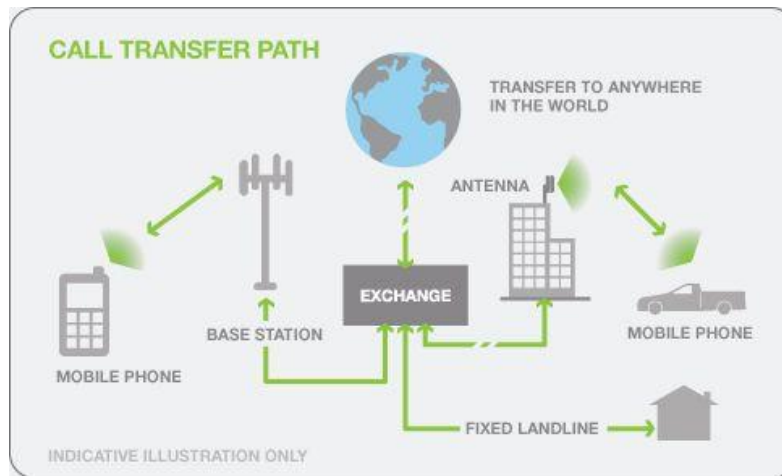


(PrivateLine, 2013)

Todas las estaciones base de una red celular están conectadas a un controlador de estaciones base (o BSC), que administra la distribución de los recursos. El sistema compuesto del controlador de estaciones base y sus estaciones base conectadas es el Subsistema de estaciones base (o BSS) (Ingeniatic, 2013).

Por último, los controladores de estaciones base están físicamente conectados al Centro de conmutación móvil (MSC) que los conecta con la red de telefonía pública y con Internet. El MSC pertenece a un Subsistema de conmutación de red (NSS) que gestiona las identidades de los usuarios, su ubicación y el establecimiento de comunicaciones con otros usuarios (Ingeniatic, 2013).

Figura 9. Intercambio y flujo de datos en una red telefónica.



(Emfexplained, 2013)

El MSC se conecta a bases de datos que proporcionan funciones adicionales como lo son:

El Registro de ubicación de origen (HLR): almacena la posición del usuario dentro de la red, si está conectado o no y las características de su abono (servicios que puede y no puede usar, tipo de terminal, etcétera). Es de carácter más bien permanente; cada número de teléfono móvil está adscrito a un HLR determinado y único, que administra su operador móvil.

Al recibir una llamada, el MSC pregunta al HLR correspondiente al número llamado si está disponible y dónde está (es decir, a qué BSC hay que pedir que le avise) y en ruta la llamada o da un mensaje de error (Kioskea, 2013).

El Registro de ubicación de visitante (VLR): es una base de datos más volátil que almacena, para el área cubierta por un MSC, los identificativos, permisos, tipos de abono y localizaciones en la red de todos los usuarios activos en ese momento y en ese tramo de la red. Cuando un usuario se registra en la red, el VLR del tramo al que está conectado el usuario se pone en contacto con el HLR de origen del usuario y verifica si puede o no hacer llamadas según su tipo de abono. Esta información permanece almacenada en el VLR mientras el terminal de usuario está encendido y se refresca periódicamente para evitar fraudes (por ejemplo, si un usuario de prepago se queda sin saldo y su VLR no lo

sabe, podría permitirle realizar llamadas) (Kioskea, 2013).

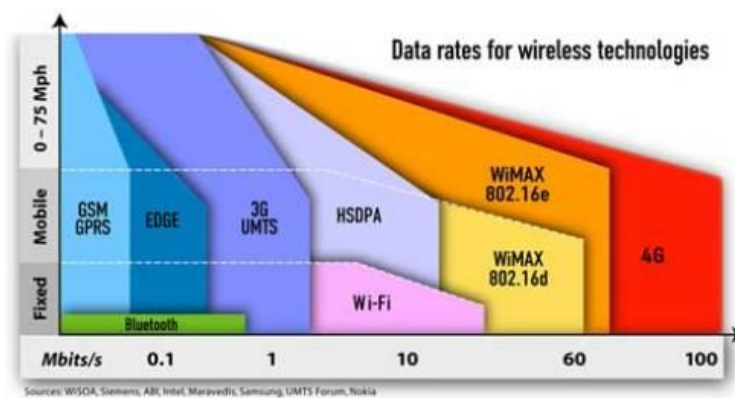
El Registro de identificación del equipo (EIR): guarda listas de permiso de acceso al terminal, al que identifica unívocamente mediante su número de serie o IMEI, para evitar que los terminales robados y denunciados puedan usar la red (Kioskea, 2013).

El Centro de autenticación (AUC): verifica las identidades de los usuarios. La red celular compuesta de esta manera está diseñada para admitir movilidad a través de la gestión de traspasos (movimientos que se realizan de una celda a otra) (Ingeniatic, 2013).

GSM permite acuerdos entre operadores para compartir la red, de modo que un usuario en el extranjero –por ejemplo— puede conectarse a una red (MSC, VLR y capa de radio) de otro operador. Al encender el teléfono y realizar el registro en la red extranjera, el VLR del operador extranjero toma nota de la información del usuario, se pone en contacto con el HLR del operador móvil de origen del usuario y le pide información sobre las características de abono para permitirle o no realizar llamadas. Así, los distintos VLRs y HLRs de los diferentes operadores deben estar interconectados entre sí para que todo funcione (Kioskea, 2013).

b. Evolución de la tecnología GSM. Las primeras terminales en basarse en telefonía celular contaban con tasas de transmisión de datos aceptables. Pero como era de esperarse la demanda en la velocidad de información fue aumentando por lo que GSM fue adaptándose a estos cambios teniendo hasta ahora la más rápida denominada 4G.

Figura 10. Representación de generación GSM y sus velocidades de operación



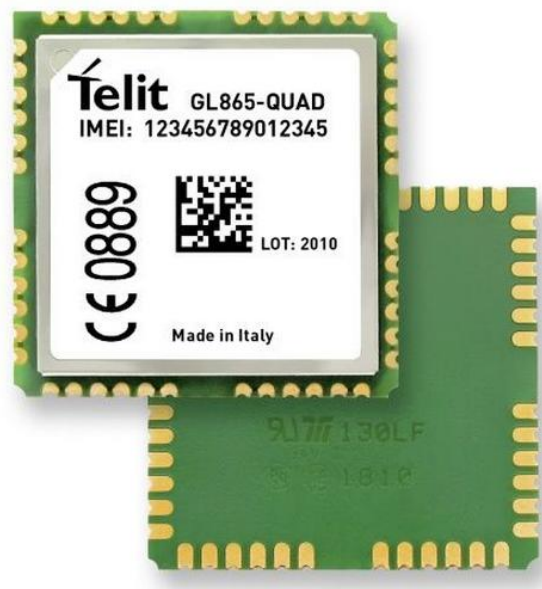
(Softhandover, 2013)

En la anterior figura vemos como es de influyente la velocidad en la que se encuentra el usuario móvil descendiendo por grandes factores la velocidad de transmisión (hasta un 90% a altas velocidades).

3. Módulos de comunicación con la red celular. Estos módulos o bien circuitos integrados con los responsables que nuestro teléfono celular, Tablet, sistema de telemetría, automóvil y computador, entre otros, logre comunicarse con las redes de telefonía inalámbrica. En ellos se encuentra la circuitería necesaria para enviar y recibir datos de las redes telefónicas.

a. Módulo TELIT GL865. El módulo GL865 de la empresa TELIT es uno de los tantos que permiten realizar el acceso a la red telefonía. Es de tecnología GSM de generación 2. Entre sus características principales se encuentra poder crear sesiones TCP y rangos de operación dinámicos.

Figura 11. Circuito integrado TELIT GL865.



(Telit, 2013)

La figura anterior nos demuestra un empaquetado fácil de implementar ya que tiene todos sus pines expuestos dejando por un lado las configuraciones más elegantes como BGA.

b. Plataforma GSM click. La plataforma GSM click de la empresa Mikroelectronica es una adaptación del módulo GL865 para ser usado en sus diferentes plataformas de programación. Facilita el proceso de pruebas ya que cuenta con el integrado ya adaptado a las líneas de transmisión serial TX y RX además que viene soldado a la placa ahorrándonos este trabajo.

Figura 12. GSM click.



(MikroE, 2013)

Cuenta además con espacio para insertar una tarjeta SIM y conversores de voltajes para no tener que agregar circuitería adicional. Viene desbloqueado para ser usado con cualquier empresa de telefonía a nivel mundial con estándares GSM.

D. Comandos Hayes

Cada fabricante de dispositivos móviles tiene su lenguaje con el cual se comunica con la red celular. El más comúnmente usado es el de comandos AT. Estos comandos fueron originalmente desarrollados por Hayes Microcomputer Company para su módems y a lo largo del tiempo de la comunicación celular se han convertido en el estándar de la mayoría de módulos que se conectan con la red celular.

El programa de comunicaciones debe ser capaz de seleccionar los parámetros necesarios para el correcto funcionamiento con el destino al cual se conecta. Para que el programa de comunicaciones pueda "hablar" con el módem debe haber algún tipo de protocolo, este es objetivo de los comandos AT.

Ocurre un dialogo de comandos en el cual cualquiera de los dos puede inicializarlo.

1. Comando AT. Este es el comando base de los comandos Hayes. Con este comando podemos comprobar que nuestro canal es el adecuado para el modulo y que todos los parámetros están correctos. Además podemos ver si algún proceso ya termino esperando respuesta OK por parte de la red celular.

Se le llama el comando base ya que todos los comandos llevan el prefijo AT.

```
Envio    "AT"  
Respuesta "OK"
```

2. Comando AT+CGMI. Comando utilizado para conocer el fabricante del dispositivo. Nótese que lleva el comando base AT el cual es concatenado a CGMI.

```
Envio    AT+CGMI=?  
Respuesta OK  
Respuesta AT+CGMI  
Respuesta TELIT  
Respuesta OK
```

3. Comando ATE0. Comando utilizado para que la red celular no haga un 'echo' o repetición del comando enviado.

Si por ejemplo ejecutamos el comando AT+CGG la red celular nos devolvería la información asociada a ese comando y continua nos devolvería otra vez AT+CGG. Con el comando ATE0 se ahorra este doble envío.

Como es su naturaleza, este va al principio de toda la comunicación así empezamos un dialogo sin repeticiones en ningún momento.

4. Comando AT+CMEE. Este comando está relacionado con el despliegue de errores que se generan dentro del módulo. Es un comando al cual se le puede asignar un valor numérico para saber cómo será su comportamiento. Por defecto es asignado al valor '0' de esta forma "AT+CMEE=0" y en este caso si se genera un error ya sea por parte del módulo o red celular obtendremos el mensaje "ERROR" sin especificar a qué se

debió.

Si le asignamos el valor de '1', el módulo al detectar un error nos lo devolverá en el siguiente formato:

“ERROR (error#), (descripción)”

Con este formato es más fácil identificar los errores ya que no solo nos da el número del error si no también nos describe resumidamente que fue lo que paso (si en caso no tuviéramos el manual de errores disponible).

5. Comando AT+CGDCONT. El comando llamado “Define PDP Context”, especifica parámetros como el APN de la red celular (en este caso TIGO Guatemala) y el protocolo con el cual nos estaremos comunicando (en este caso el Internet Protocol, IP). Lo que busca es configurar el contexto que estaremos trabajando. Para módulos del fabricante Telit se pueden tener múltiples contextos.

La estructura del comando es la siguiente:

“AT+CGDCONT = #contexto, protocolo, APN”

El contexto es el primer parámetro a especificar, el segundo parámetro es el protocolo de red que estaremos usando (podría ser diferente a IP como por ejemplo más avanzados como IPV6), por último se tiene el APN (Acces point name) o el nombre de acceso de punto lo cual refiere a una dirección con la cual se accede a la red de datos telefónica de un proveedor en específico.

6. Comando AT#SCFG. El comando llamado “Socket configuration” es el encargado de especificar los parámetros al socket se conectara remotamente.

La estructura del comando es la siguiente:

“AT#SCFG= <connId>,<cid>, <pktSz>,<maxTo>, <connTo>”

Su primer parámetro es el identificador de la conexión. El siguiente define el contexto del protocolo a los paquetes a enviar. El tercero es el tamaño de paquetes a enviar por el protocolo. El cuarto es el tiempo de inactividad para cerrarse, si no ha habido respuesta dentro de este tiempo se cierra la sesión. Por último se tiene el valor máximo para espera y generar errores. Si no fue posible la conexión dentro de este tiempo, el modulo nos regresa un mensaje de error.

E. Sistemas M2M, Machine to Machine

El concepto M2M (Machine to machine) engloba una serie de tecnologías que se utilizan en el proceso de intercambio de información entre dos máquinas remotas, ya sea mediante redes fijas o móviles y sin intervención humana (PrivateLane, 2013).

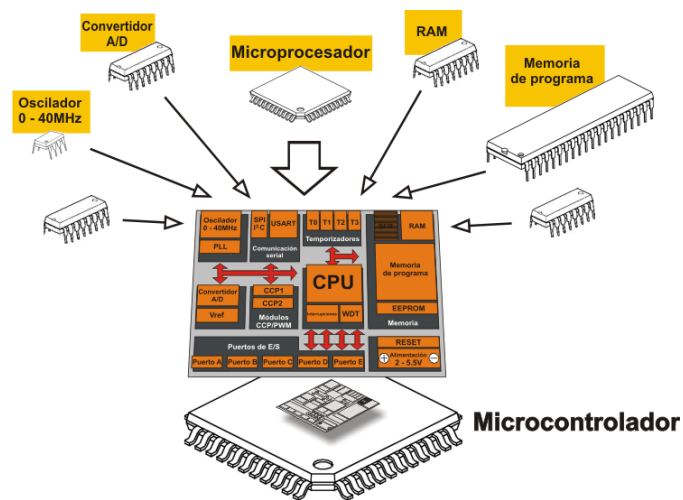
Algunos ejemplos de estos dispositivos son: contadores de consumo de servicios públicos de suministro, como el gas o la electricidad, que leen el consumo realizado cada cierto tiempo y almacenan la información para luego ser extraída manualmente o remotamente utilizando algún medio de transporte. Telemetría que permite tanto la manipulación como la observación de variables de estado de algún proceso o sistema en específico.

F. Microcontrolador

1. Definición. Un microcontrolador es un circuito integrado o chip que incluye en su interior las tres unidades funcionales de una computadora: CPU, Memoria y Unidades de E/S, es decir, se trata de una computadora completa en un solo circuito integrado programable y se destina a gobernar una sola tarea con el programa que reside en su memoria. Sus líneas de entrada/salida soportan el conexionado de los sensores y actuadores del dispositivo a controlar (Emfexplained, 2013).

El microprocesador ejecuta instrucciones almacenadas como números binarios organizados secuencialmente en la memoria principal reaccionados a eventos o simplemente a un ciclo principal.

Figura 13. Sub módulos de un microcontrolador.

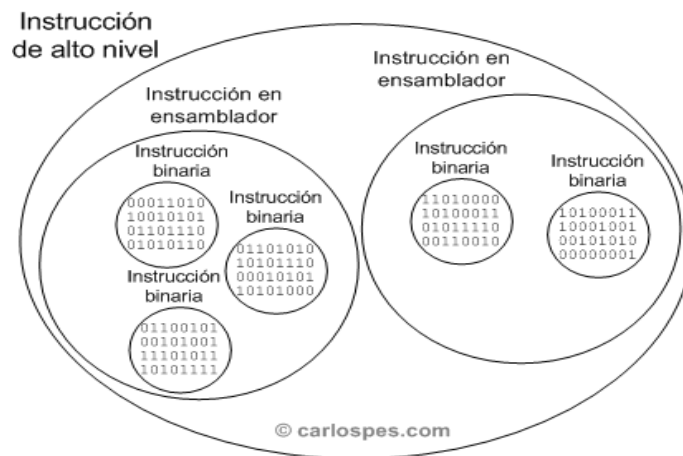


(MikroE, 2013)

Desde el punto de vista lógico, singular y funcional, el microprocesador está compuesto básicamente por varios registros de propósito general y propósito específico, unidad de control, unidad aritmético-lógica, entradas y salidas, memorias RAM ROM y módulos internos dependiendo del fabricante.

2. Programación. Los microcontroladores soportan diversos lenguajes de programación entre los cuales encontramos de bajo y alto nivel. Una vez tenemos el código fuente, este es descargado al microcontrolador mediante herramientas específicas de cada fabricante

Figura 14. Estructura de los niveles de programación



(Carlospes, 2013)

Cuando hablamos de bajo nivel hablamos del lenguaje ensamblador. Como sabemos el único lenguaje que entienden los microcontroladores es el código máquina formado por ceros y unos del sistema binario. El lenguaje ensamblador expresa las instrucciones de una forma más natural al hombre a la vez que muy cercana al microcontrolador, ya que cada una de esas instrucciones se corresponde con otra en código máquina. El lenguaje ensamblador trabaja con nemónicos, que son grupos de caracteres alfanuméricos que simbolizan las órdenes o tareas a realizar (Wikipedia, 2013).

Figura 15. Ejemplo de código en lenguaje ensamblador

```
BSF      BAUDCON+0, 3, 0
CLRF    SPBRGH+0
MOVLW   138
MOVWF   SPBRG+0
BSF     TXSTA+0, 2, 0
CALL    _UART1_Init+0, 0
```

Cuando hablamos de alto nivel nos referimos a lenguajes de programación que se asemejan más a instrucciones o comandos que el ser humano haría llamar para cumplir una función. Por ejemplo en lenguajes de bajo nivel no contamos con condiciones básicas como lo es IF o de ciclo como lo son FOR y WHILE por lo que un lenguaje de alto nivel se encarga en convertir estas funciones en lenguaje ensamblador para que el usuario no tenga que realizar complejas funciones en lenguaje de bajo nivel.

Figura 16. Ejemplo de código en lenguaje de alto nivel.

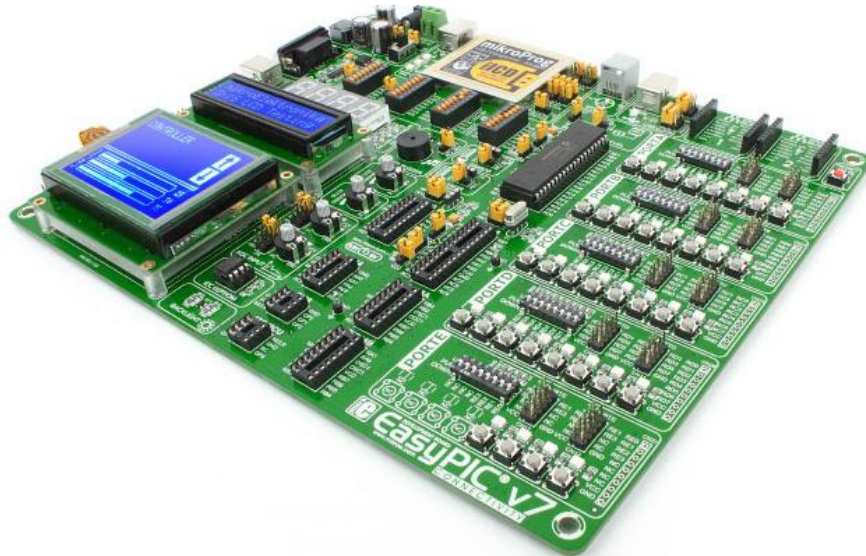
```
while(1)
{
    if(var1 == 3 $$ var2 != 8)
    {
        Funcion();
        Funcion2(var3);
    }
}
```

Para descargar nuestro código final utilizamos herramientas denominadas programadores. Estos se encargan, mediante circuitería conectada a los microcontroladores, de pasar nuestro programa en forma de unos y ceros para que sea

b. Plataforma de desarrollo EasyPic 7. EasyPic 7 es la séptima generación de plataformas de desarrollo para los microcontroladores de la serie PIC. Es fabricada por Mikroelectronica y proporciona muchas herramientas tanto hardware como software para programar microcontroladores y módulos adicionales para lograr aplicaciones de varias índoles.

Facilita el desarrollo teniendo LEDS en cada puerto para ver qué es lo que está ocurriendo, además cuenta con un juego de resistencias que nos permite variar en triestado cada puerto individualmente.

Figura 18. Plataforma de desarrollo EasyPic7 de MikroE.



(MikroE, 2013)

Entre sus módulos de comunicación se encuentran puertos seriales e interfaz para las computadoras mediante un MAX232. A esto se le suma la capacidad de soportar opciones graficas como lo es pantallas LED y pantallas Touch.

Estas cualidades y muchas otras más hacen que esta plataforma de desarrollo sea muy completa y gama de aplicaciones que puede desarrollarse se extiende aún más con la adición de sus nuevos puertos llamados Click donde se interconectan plataformas ya hechas pero con circuitos dedicados para realizar alguna función (por ejemplo altímetros, comunicación con fibra óptica, protocolo 485, etc.).

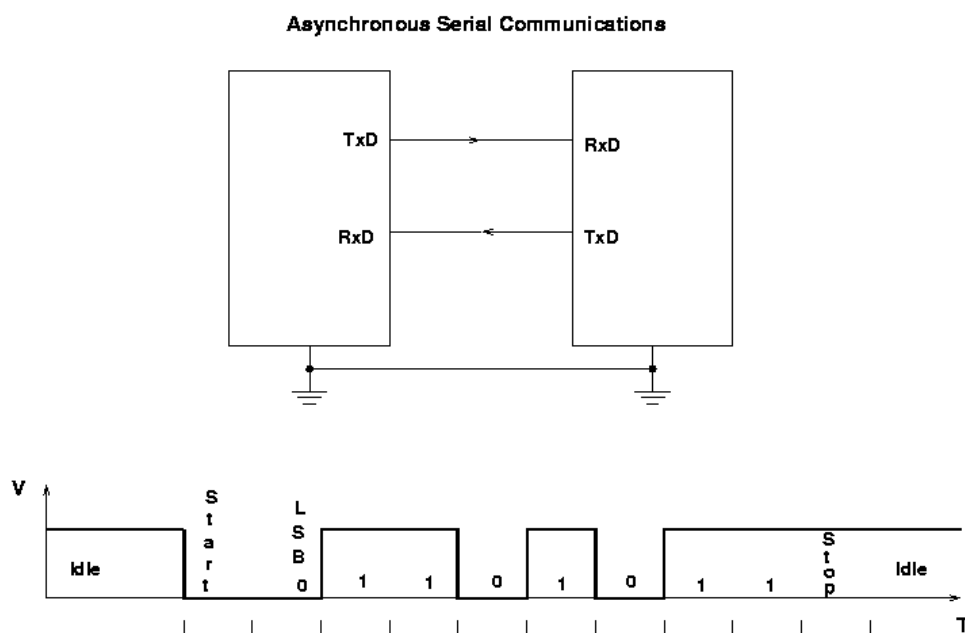
3. Interrupción. Interrupción es una señal recibida por el procesador de un ordenador o microcontrolador, indicando que debe "interrumpir" el curso de ejecución actual y pasar a ejecutar código específico para tratar esta situación (Wikipedia, 2013).

Una interrupción es una suspensión temporal de la ejecución de un proceso, para pasar a ejecutar una subrutina de servicio de interrupción, la cual, por lo general, no forma parte del programa principal de ejecución comúnmente llamado principal o main. Luego de finalizada dicha subrutina, se reanuda la ejecución del programa principal (Wikipedia, 2013).

4. Comunicación serial. La comunicación serial consiste en el envío de un bit de información de manera secuencial, esto es, un bit a la vez y a un ritmo acordado entre el emisor y el receptor (Softhander, 2013).

Existen en la actualidad diferentes ejemplos de puertos que comunican información de manera serial (un bit a la vez). El conocido como "puerto serial" ha sido gradualmente reemplazado por el puerto USB (Universal Serial Bus) que permite mayor versatilidad en la conexión de múltiples dispositivos (Softhander, 2013).

Figura 19. Comunicación entre dos terminales y trama de datos serial.



(Jlozano, 2013)

Además de realizar las conexiones físicas entre las dos terminales para lograr la comunicación, para que pueda establecerse la comunicación serial debe existir un acuerdo previo en la manera cómo van a ser enviados los datos. Este acuerdo debe incluir los niveles de voltaje que serán usados, el tamaño y formato de cada uno de los mensajes (número de bits que constituirán el tamaño de la palabra, existirá o no un bit de inicio y/o de parada, se empleará o no un bit de paridad), el tipo de lógica empleada (qué voltaje representará un cero o un uno), el orden en que serán enviados los datos (será enviado primero el bit de mayor peso o el de menor peso) y la velocidad de envío de datos (Softlander, 2013).

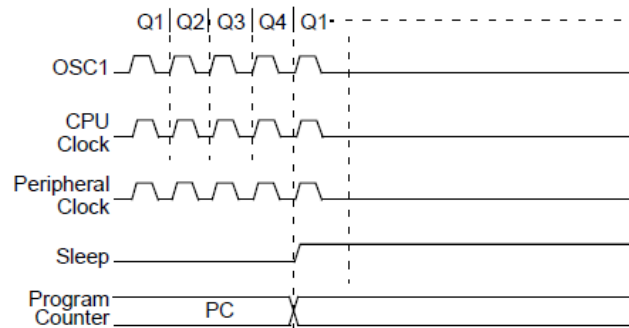
5. Contadores internos. Muchos de los microcontroladores cuentan con módulos llamados TIMERS. Estos timers son muy utilizados en tener referencias de tiempo, captura y medición de eventos, realizar interrupciones y secuencias en el que el programa se ejecutara.

Por lo general se tiene más de uno y con configurables para que su cuenta vaya conforme el reloj de microcontrolador o bien vaya a un paso más despacio para fines propios de la aplicación.

La aplicación típica de estos contadores es que al llenarse sus registros y que ocurra un desborde, generan una interrupción. Sabiendo que la interrupción entra en un lapso de cada unidad de tiempo podemos asignar tareas periódicas.

6. Estado SLEEP. En muchos de los microcontroladores es un estado en el cual se obtiene un menor consumo de corriente debido a que se desconectan relojes internos. Entre estos relojes encontramos principalmente el reloj del CPU, el reloj de los periféricos y osciladores internos primarios.

Figura 20. Secuencia de activación y desactivación al ejecutar la instrucción Sleep.



(Microchip, 2013)

El consumo de corriente reduce drásticamente optimizando nuestros recursos disponibles.

Entrar a este estado supone que haremos tareas periódicas.

Lo contrario a este estado SLEEP es “hacerlo despertar” o como muchos fabricantes describen “Wake-up on SLEEP”. Las fuentes que pueden hacer despertar a un microcontrolador son diversas entre las cuales podemos encontrar perturbaciones en ciertos puertos, desborde de contadores y recepción de datos seriales entre otros.

Para ejemplificar una tarea periódica en donde suceda este caso SLEEP-WAKE consideremos un tanque de agua grande que está siendo abastecido por un flujo pequeño de líquido y un sensor de nivel para medir la altura del tanque.

Sería impráctico obtener lecturas constantes del nivel de agua para saber dónde se encuentra. Por lo que podemos programar una secuencia sencilla que al tomar la medida de agua y observarla en el microcontrolador, que se ejecute el comando SLEEP para que un tiempo suficiente después pueda volver a ser medido el nivel. Esta nueva medición se hará cuando alguna de nuestras fuentes obligue al microcontrolador a despertarse o de virtualmente ejecutarse el “WAKE UP”.

En microcontroladores 18F del fabricante Microchip el consumo en estado SLEEP tiene un valor típico de 20 nano amperios hora.

G. Sistemas de posicionamiento global

Los sistemas de posicionamiento global es un sistema de navegación por satélite que permite determinar en el globo terráqueo la posición de una persona, objeto y/o vehículo motorizado entre otros con una precisión aceptable siempre y cuando se cuente con el equipo receptor adecuado. Todo esto gracias a satélites que se encuentran en órbita con la Tierra llamados constelaciones los cuales transmiten información para ser decodificada y procesada.

1. Global Positioning System (GPS). El Sistema de Posicionamiento Global (GPS) es un servicio propiedad de los Estados Unidos de América que proporciona a los usuarios información sobre posicionamiento, navegación y cronometría (MikroE, 2013).

Diseñado para apoyar los requerimientos de navegación y posicionamiento precisos con fines militares, en la actualidad es una herramienta importante para aplicaciones de navegación, posicionamientos de puntos en tierra, mar y aire tanto para personal militar como civil alrededor de todo el mundo (Telefonica, 2013).

2. Global'naya Navigatsionnaya Sputnikovaya Sistema (GLONASS). Es un Sistema Global de Navegación por Satélite (GNSS) desarrollado por la Unión Soviética siendo hoy administrado por la Federación Rusa y que representa la contrapartida al GPS estadounidense y al futuro Galileo europeo (Grayeb, 2013).

La constelación de GLONASS se mueve en órbita alrededor de la Tierra con una altitud de 19.100 km (diecinueve mil cien kilómetros) algo más bajo que el GPS (20.200 km) y tarda aproximadamente 11 horas y 15 minutos en completar una órbita (Grayeb, 2013).

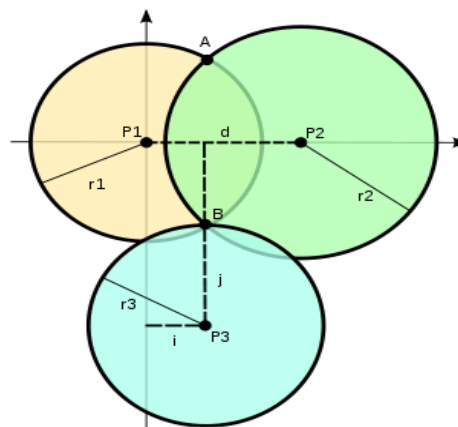
3. Galileo. Galileo es un sistema global de navegación por satélite (GNSS) desarrollado por la Unión Europea (UE), con el objeto de evitar la dependencia de los sistemas GPS y GLONASS.¹ Al contrario de estos dos, será de uso civil y será abierto al público en el año 2014 (MikroE, 2013).

4. Beidou. Beidou es el proyecto impulsado por la República Popular de China para

contar con su propio sistema de posicionamiento global. A diferencia de los demás este busca solamente servir a su país desarrollador y vecinos potenciales. Este servicio se encuentra activo desde el año 2000 el cual se ofrece en versiones militares y civiles.

5. Funcionamiento general de los sistemas de posicionamiento global. En principio el receptor busca engancharse a por lo menos cuatro o más satélites de la respectiva constelación que se esté utilizando. Una vez enganchado a los satélites, se logra la ubicación del dispositivo mediante un proceso llamado Trilateración el cual consiste en obtener posiciones relativas de objetos y utilizando geometría de triángulos se realizan cálculos para obtener las latitudes y longitudes deseadas.

Figura 21. Trilateración utilizando círculos.



(Wikipedia, 2013)

Como podemos ver en la imagen nos ubicamos en el punto B y queremos saber nuestra posición respecto a los puntos P1 y P2 que en este caso representan nuestros satélites en vista por el receptor. Mediante cálculos geométricos el receptor computa la distancia entre el punto B y los otros dos puntos llamadas estas distancias r_1 y r_2 . Conociendo estas distancias reducimos nuestra área de interés a un intercepto de circunferencias. Si a este cálculo le agregamos un tercer satélite en un punto P3 y una distancia r_3 reduciríamos el plano de interés aún más hasta llegar a una medición óptima. En este ejemplo los planos de los círculos se intersecan en el punto B que es donde estaría el receptor.

El proceso continúa hasta encontrar un número máximo de satélites donde 12

satélites es la capacidad típica de un receptor.

El proceso descrito anteriormente es un caso en 2D, para cual el sistema de localización utiliza tres dimensiones para una exactitud aún mejor.

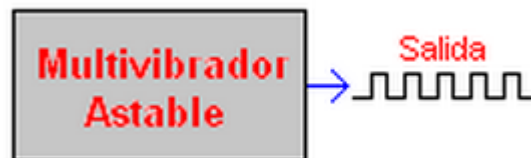
Como primer paso para la localización 3D el receptor debe de encontrar la distancia de tres satélites a las cuales entre ellos conocen su distancia.

H. Multivibradores

En la electrónica se conoce como multivibradores a los circuitos capaces de producir señales cuadradas periódicas o no periódicas. Dependiendo de su configuración estos pueden tener varias respuestas entre las cuales las más comunes son estable, monoestable y biestable.

1. Configuración estable. En esta configuración, el multivibrador produce una señal sin necesidad de una entrada que le dicte cuando ejecutarse. Son muy usados como relojes para circuitos o bien actuadores periódicos.

Figura 22. Representación de un multivibrador estable



(Lopez, 2013)

2. Configuración monoestable. En esta configuración, a comparación de los estable, se necesita de una señal que le active su circuitería para producir una señal no periódica. Se conocen también como multivibradores de disparo ya que se activa con una señal y este genera una conocida como pulso de duración por lo general corta.

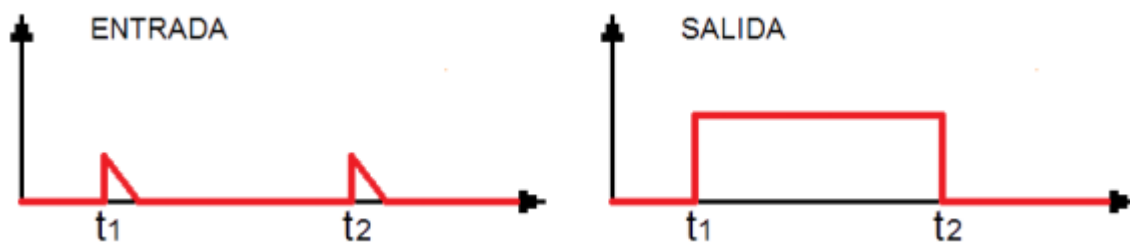
Figura 23. Representación de un multivibrador monoestable.



(López, 2013)

3. Configuración biestable. Este es un caso intermedio entre el astable y monoestable ya que al recibir una entrada este genera una señal que se quedará en “alto” conceptualmente hasta el infinito hasta que otra señal le indique que debe regresar al estado “bajo”.

Figura 24. Respuesta de un oscilador biestable.



(González, 2013)

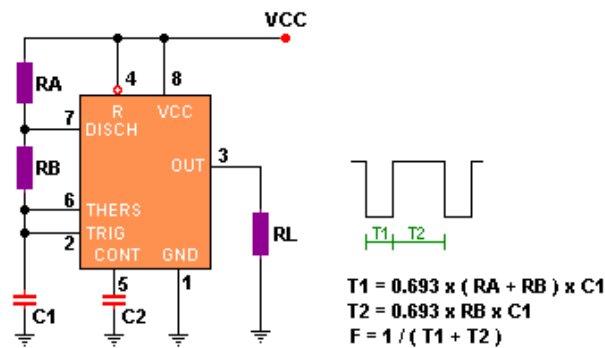
4. Circuito integrado 555. El dispositivo 555 es un circuito integrado muy estable cuya función primordial es la de producir pulsos de temporización con una gran precisión y que, además, puede funcionar como oscilador (MikroE, 2013).

Su uso se extiende aún más y realiza funciones de generador de ondas pudiendo obtener las señales rectangulares y rampas.

Con las configuraciones adecuadas es posible que responda de las maneras anteriormente descritas. Entre estas astable, monoestable y biestable.

a. Configuración integrado 555 como astable. En su versión más simple, un astable puede configurarse de la siguiente forma:

Figura 25. Circuito para un oscilador 555 y sus cálculos respectivos.



(Xbot, 2013)

b. Integrado TS555. Según el tipo de fabricante se le da a cada versión de sus integrados 555 con las denominaciones más comunes como: TLC555, LMC555, uA555, NE555C, MC1455, NE555, LM555, etc. aunque se lo conoce como "el 555" a todas ya que todas responden de manera muy similar.

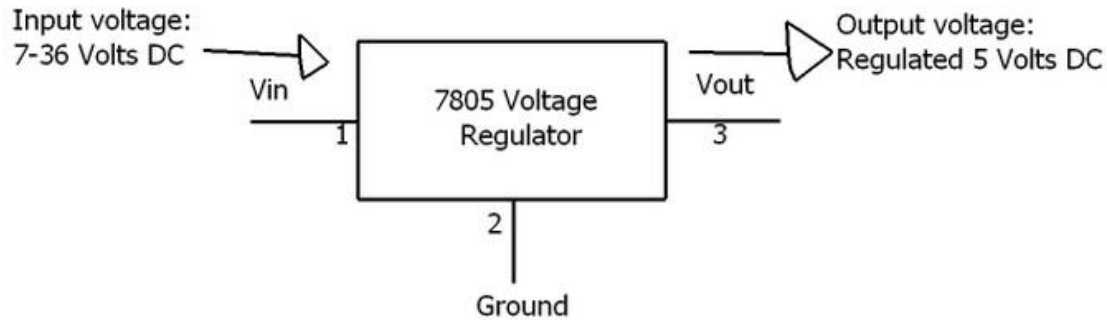
Existe un integrado con la denominación TS555 el cual su mayor característica es su rango de voltaje de operación. Este integrado a comparación de las demás versiones opera a voltajes de 3 voltios a mayores a 15 por lo que lo hace más versátil. Por esta característica el integrado consume menos corriente.

I. Regulador de voltaje

Un regulador de tensión o regulador de voltaje es un dispositivo electrónico diseñado para mantener un nivel de voltaje constante.

Los reguladores electrónicos de tensión se encuentran en dispositivos como las fuentes de alimentación de los computadores, donde estabilizan los voltajes DC usados por el procesador y otros elementos.

Figura 26. Representación de un regulador de voltaje.



(LearningElectronics, 2013)

Cada fabricante diseñó los reguladores con características específicas entre las cuales las más consultadas son:

- Voltaje regulado: voltaje que se mantendrá constante durante la operación del regulador.
- Rango de entrada: se refiere al rango de voltajes el cual el regulador opera correctamente.
- Corriente de entrada máxima: la corriente máxima de entrada al circuito regulador que hará que funcione si bien en su límite, funcionara correctamente.
- Corriente de salida máxima: es la corriente máxima que puede proporcionar el integrado.
- Dropout voltaje: diferencia entre el voltaje de entrada y voltaje regulado.

Todas estas variables si son aumentadas arriba de su rango el fabricante no garantiza su operación por lo que medidas de seguridad deben ser tomadas al escoger un regulador.

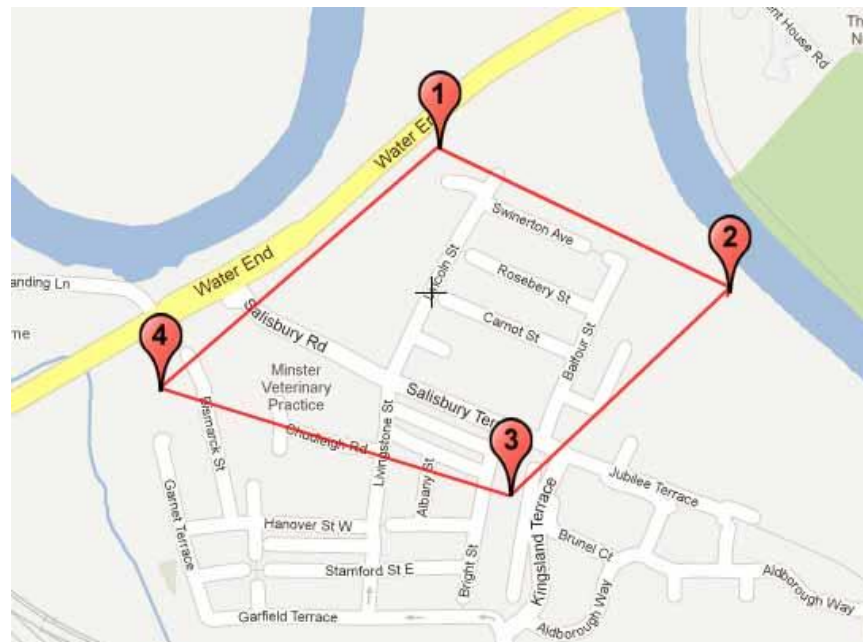
1. Dropout voltaje. Este es un parámetro importante para los reguladores de voltaje ya que es la tensión mínima para que el regulador mantenga el valor deseado. Por ejemplo el dropout voltaje de un regulador de 6 voltios puede ser de 1 volt. Esto nos dice que nuestro voltaje de entrada debe de ser por lo menos de 7 voltios. Más bajo el regulador no se comportara como el fabricante lo diseño.

J. Definición Geofence

Geofence o geo-valla es un perímetro virtual de un área geográfica en específico. Delimita los espacios físicos en el que algún objeto puede movilizarse.

Esta delimitación puede ser el radio de un círculo o bien una serie de líneas unidas para formar un polígono y así delimitar con mayor precisión.

Figura 27. Representación de Geofence en 4 puntos.

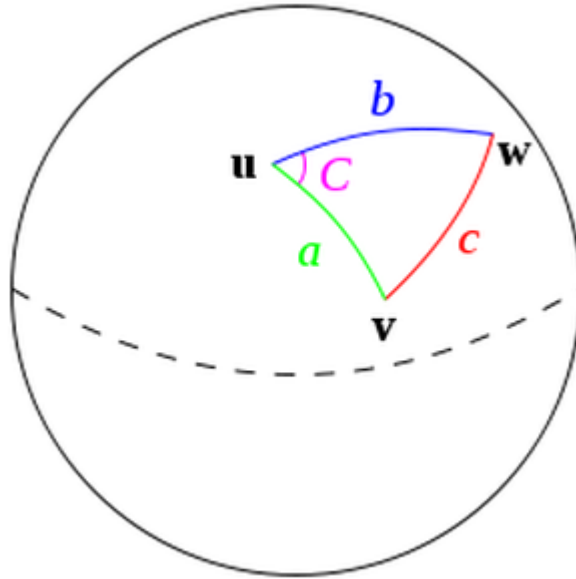


(Genovo, 2013)

K. Fórmula de Haversine

La fórmula de Haversine es utilizada para encontrar la distancia entre dos puntos en el globo terráqueo dadas sus posiciones geográficas (Latitud y longitud). Esta fórmula se basa en la geometría de círculos y debido a esto los cálculos finales son una muy buena aproximación considerando que la tierra no es una esfera uniforme si no un ovalo irregular.

Figura 28. Distancias en una esfera.



(Wikipedia, 2013)

Respecto a la anterior imagen, teniendo los puntos (u, w, v) en una esfera uniforme es posible encontrar las distancias entre cada punto (a, b, c) utilizando la siguiente fórmula:

$$\varphi = \textit{latitud}$$

$$\lambda = \textit{longitud}$$

$$R(\textit{radio de la tierra}) = 6371 \textit{ (km)}$$

$$a = \sin^2 \frac{\Delta\varphi}{2} + \cos \varphi_1 * \cos \varphi_2 * \sin^2 \frac{\Delta\lambda}{2}$$

$$c = 2 * \tan^{-1}(\sqrt{a}) * \tan^{-1}(1 - a)$$

$$d = R * c$$

Siendo d nuestra distancia entre los dos puntos.

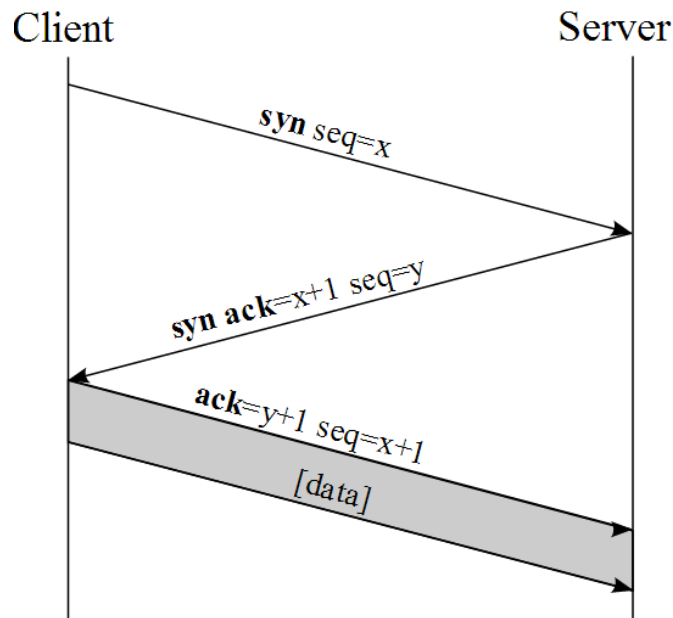
L. Handshake entre sistemas de comunicaciones

Es el protocolo de comienzo de comunicación entre dos máquinas o sistemas.

Involucra hardware y software para establecer y/o mantener dos sistemas en sincronización. Esta conexión requiere a menudo el intercambio de mensajes o paquetes entre las dos partes para establecer tasas y velocidades de transmisión.

Para ejemplificar esta acción de “Apretón de manos” podemos ver el diagrama de una sesión de datos TCP:

Figura 29. Handshake para una sesión TCP convencional.



(Wikipedia, 2013)

De primero el cliente envía un mensaje de respuesta con el valor ‘x’, esperando que el servidor le conteste con este valor ‘x’ incrementado en uno y otro mensaje adicional. Una vez el cliente recibe la información que puede engancharse e intercambiar datos, se empieza la transmisión teniendo control del flujo de datos como lo es tablas de chequeo (Checksum) y constantes acknowledge como métodos de controles internos en la trama

VI. Diálogo general entre sesión de datos y servidor

Para entender cómo es que el dispositivo y el servidor intercambian parámetros, se describe la estructura de los datos y mensajes de respuesta entre cada uno.

En esta sección se describe paso a paso que es lo que ocurre con el modo a nivel de operación y los caminos que puede tomar así como variables y banderas especiales a considerar debido a su alta importancia.

La sesión de datos TCP creada en el módulo GSM enviara datos al servidor mediante el protocolo HTTP las siguientes variables:

Tabla No.1. Variables utilizadas por el módulo para ingresar datos.

Variables utilizados por el módulo para ingresar datos	
Variable	Descripción
User	Es el identificador del usuario en el módulo.
Lat	Latitud respecto a datos obtenidos del GPS
Lon	Longitud respecto a datos obtenidos del GPS
Org	Origen de los datos a ingresar a la base de datos. Datos pueden ser de tipo request o de ingreso a base de datos

A. Variable 'org'

De las cuatros variables la más ambigua es Org por lo cual se procede a su explicación.

Se pueden tener dos escenarios básicos, el primero es en el cual el dispositivo pregunta al servidor si se ha solicitado que envíe su ubicación. En este caso tenemos un estado de petición o request.

El segundo es en cual el dispositivo siempre este enviando sus datos sin importar si ha sido solicitado. En este caso tenemos un estado de rastreo.

Para ambos casos la variable 'org' se modifica de esta forma:

Tabla No.2. Definición de estados para la variable 'org'.

Definición de estados para la variable 'org'	
Valor de variable 'org' (lógico)	Significado
0	El mensaje enviado es para conocer si se ha solicitado la ubicación del dispositivo
1	El mensaje enviado es para guardarse en base de datos directamente.

La variable 'org' entonces define qué tipo de datos son los que está enviando el dispositivo al servidor por lo que esta variable define las acciones que posteriormente se harán en él.

1. Parámetro REQUEST en base de datos. De los parámetros en la base de datos tenemos una variable llamada REQUEST. Esta está relacionada con la petición de la ubicación del dispositivo y es modificada en la página web.

Tabla No.3. Estado de variable REQUEST y acciones que se realizan.

Estado posibles de la variable REQUEST y acciones	
0	No se ha solicitado la ubicación, enviar un mensaje de respuesta al dispositivo ('NOQQ')
1	Se ha solicitado la ubicación, enviar un mensaje de respuesta al dispositivo ('REQQ'),

Los mensajes de respuesta son descritos en la siguiente sección.

Si es que se hizo la solicitud se envía el mensaje de respuesta 'REQQ' y por parte del servidor este le asigna el valor de 0.

Como se ha estado mencionando, el dispositivo puede preguntar si se ha pedido su

ubicación. Esto lo realizara alguna persona de confianza la cual desea saber nuestra ubicación por lo que al ingresar a la página web puede cambiar el estado de esta variable REQUEST para que cuando el dispositivo pregunte si se solicitó ubicación, este pueda enviar los datos requeridos.

2. Mensajes de respuesta del servidor. El dispositivo está configurado para enviar información al servidor y este se mantiene a la espera de una respuesta para saber si la información fue ingresada correctamente o si se tiene alguna solicitud. De lo contrario vuelve a intentar el envío para asegurar la transmisión de datos.

El servidor cuenta con respuestas a eventos las cuales se describen en la siguiente tabla:

Tabla No.4. Respuestas del servidor al dispositivo.

Respuesta del servidor al dispositivo	
Respuesta	Descripción
'REQQ'	Si el dispositivo recibe esta respuesta del servidor, entonces el dispositivo debe de enviar su ubicación.
'NOQQ'	Si el dispositivo recibe esta respuesta del servidor, no se ha solicitado envío de datos.
'GOOD'	Si el dispositivo recibe esta respuesta del servidor, la información que se quería ingresar a la base de datos fue ingresada correctamente.

Dependiendo de la respuesta obtenida, así procederá el dispositivo. Ya sea a seguir alguna rutina o bien a 'dormirse' y esperar algún evento.

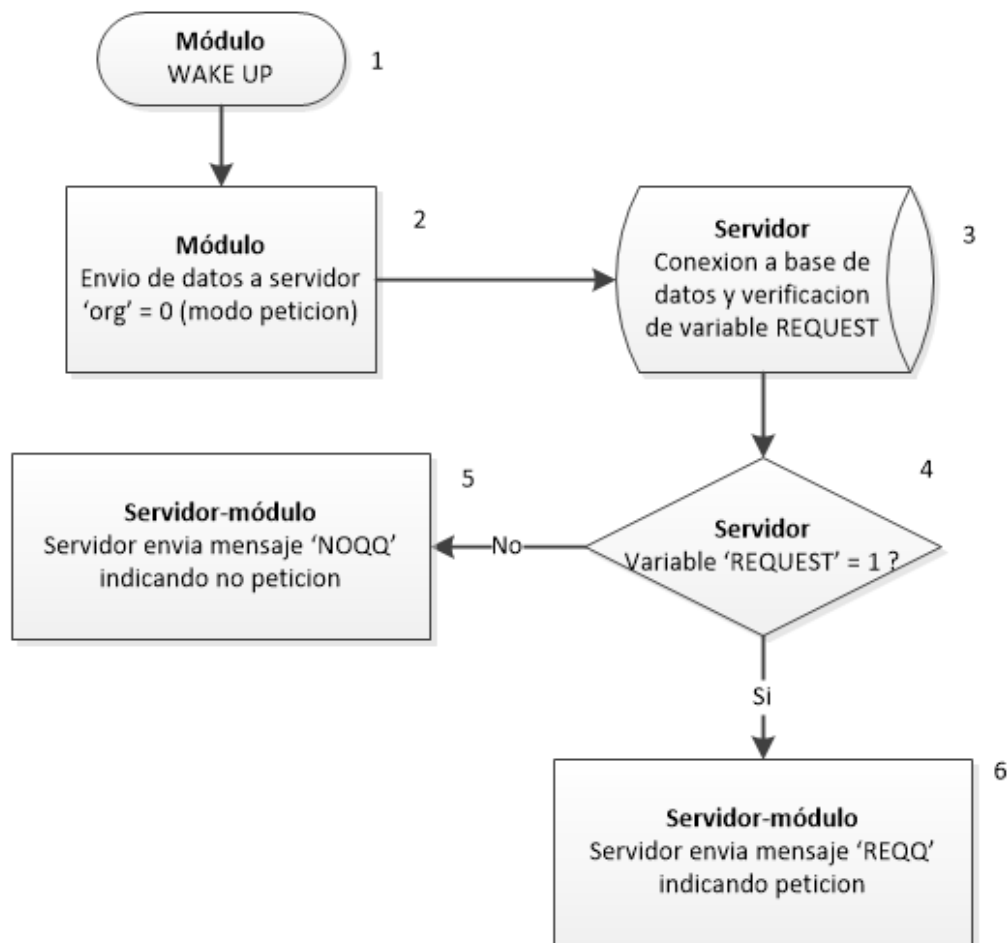
3. Ejemplo del flujo de datos. Para ejemplificar como es que el dispositivo y el servidor realizan la comunicación y reacción ante diferentes eventos consideremos el siguiente caso:

Un padre de familia instaló el dispositivo de ubicación en el bolsón de su hijo. Su deseo es conocer la ubicación indirecta de su hijo por medio del bolsón cuando él lo desee.

El padre procede a conocer la ubicación por lo que ingresa a la página web y modifica esta variable de REQUEST la cual pasa de ser 0 a 1.

A nivel de dispositivo-servidor el siguiente flujo de información y rutinas ocurre entre el módulo GSM en sesión de datos y el servidor:

Figura 30. Diagrama de flujo para nivel dispositivo -servidor.



Detalladamente:

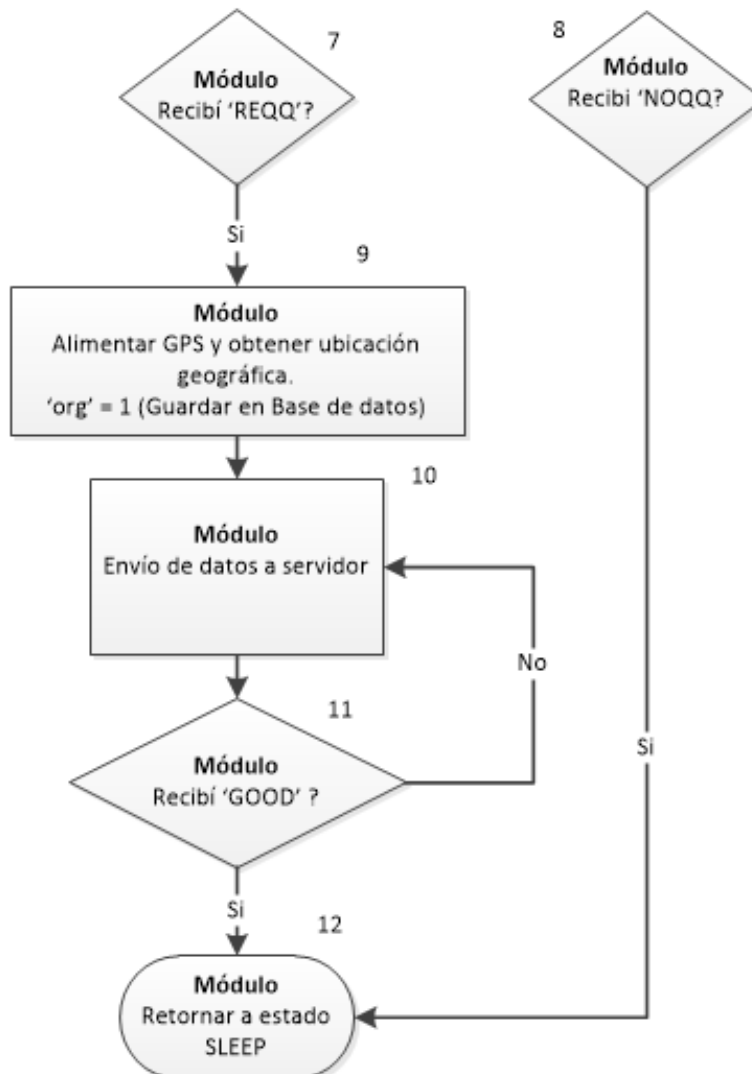
1. El dispositivo es despertado por alguna de sus interrupciones para realizar su código principal.

2. El dispositivo al ser despertado periódicamente desea saber que acción tiene que hacer por lo que contacta al servidor en modo petición para saber si debe de enviar su ubicación. En este punto la variable 'org' está en modo petición o en un cero lógico.
3. El servidor recibe esta información y dependiendo del usuario verifica sus parámetros. Entre estos si su variable de REQUEST ha sido modificada.
Si la variable en cuestión se encuentra en '1' el servidor envía un mensaje de respuesta 'REQQ' el cual hace referencia a 'REQUEST'. Por el lado contrario si encuentra que su variable REQUEST se encuentra en '0', el servidor le envía al dispositivo el mensaje 'NOREQQ' el cual hace referencia a 'NO REQUEST'.
4. El servidor hace un "querry" o petición a la base datos para acceder a la variable REQUEST. Dependiendo del valor que tenga la variable, se hará un echo al dispositivo para darle instrucciones.
5. Si REQUEST se encuentra en cero se responde al dispositivo 'NOQQ'. En otras palabras, que no envíe datos por que no se ha solicitado.
6. Si REQUEST se encuentra en uno responde al dispositivo 'REQQ'. En otras palabras que si envíe datos por que han sido solicitados.

Considerando el ejemplo anteriormente descrito, la variable REQUEST se encuentra en '1' por lo que su mensaje de respuesta al módulo será de 'REQQ'. Se le asigna en el servidor el valor de '0' a la variable REQUEST ya que su función ya fue cumplida. Si se deseara saber de nuevo la ubicación se debe modificar en la página web (Esto por ser en modo petición, en modo rastreo esta variable siempre contara el valor de '1').

Las acciones tomadas por el dispositivo y el servidor correspondientes se representan en el siguiente diagrama:

Figura 31. Diagrama de flujo del dispositivo en reacción a los mensajes de respuesta del servidor.



Detalladamente:

7. Por parte del dispositivo se verifica los caracteres que se recibieron y toma la decisión sobre que segmento de código ejecutar.
8. Si en caso se recibiera 'NOQQ', el dispositivo vuela a su estado de ahorro de energía. Para el presente ejemplo no sucedería ya que la bandera REQUEST fue alterada a tener el valor de '1'.
9. Ya que este ejemplo lo indica, recibí los caracteres correctos ('REQQ') para que envíe los datos al servidor por lo que se debe de alimentar el GPS y obtener las

coordenadas geográficas.

10. Se empaquetan los datos y se crea la sesión TCP para enviar los datos vía HTTP y se queda abierta la conexión a espera de respuesta del servidor.
11. Se tiene programado que hasta que se reciba una respuesta válida del servidor se continúe con el programa. En este caso se espera a la respuesta 'GOOD'. Si en caso no se recibe nada en el tiempo de espera, se vuelve a establecer la sesión de datos y el envío de los mismos para asegurar que la información sea efectivamente almacenada en la base de datos.

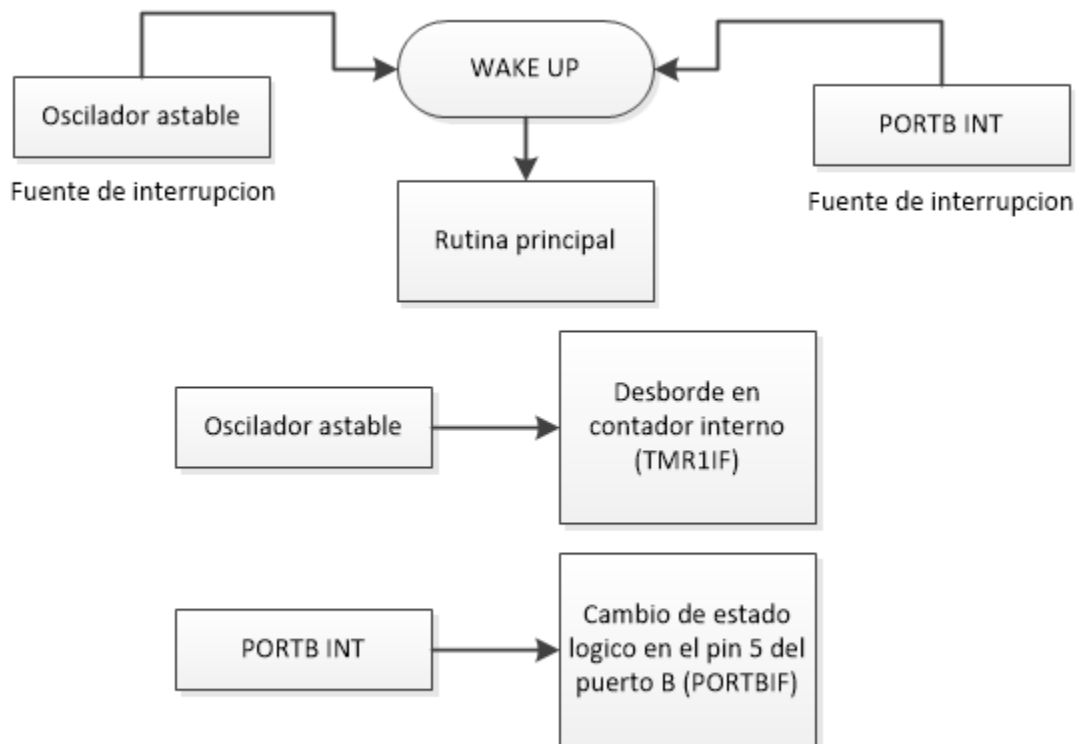
Este fue un ejemplo en donde se utilizan la mayoría de funciones del dispositivo y de opciones que pueden tomarse a lo largo de su operación. Los datos que son almacenados en la base de datos son desplegados mediante la plataforma web con la cual puede analizarse su ubicación.

En las secciones posteriores se describirá como el servidor responde y despliega los datos en la plataforma web.

VII. Estructura y modos de funcionamiento

El presente trabajo se diseñó teniendo tres modos de operación principal y dos extensiones de estos que realizan la misma función de transmitir la ubicación del dispositivo pero varían en la frecuencia en que esta información se envía y eventos por los que ha sido enviada.

Figura 32. Interrupciones posibles para despertar al microcontrolador y origen de estas.



En la rutina principal se tiene el envío de datos provenientes del GPS a la página web la cual puede desplegar los datos de diversas formas. Esta rutina principal es ejecutada periódicamente tanto para enviar información así como para saber si se ha solicitado en el envío de la posición del dispositivo.

Como modo de despliegue se basa en el API de Google Maps con el que tenemos la opción de solo desplegar las últimas posiciones o bien trazar una línea entre cada

posición conforme fueron encontradas para visualizar una trayectoria.

Como modo de control utilizando una GeoFence se traza un círculo en el API de Google Maps y la última posición encontrada para saber si se ha salido de sus límites permisibles la persona.

Como modo de petición, desde la página web podemos ingresar a solicitar que el dispositivo nos envíe su ubicación actual.

A. Funcionamiento en modo de rastreo

En este modo, el dispositivo se encuentra en estado de espera. Más específicamente el modulo con la red celular y GPS se encuentran desconectados de las fuentes de alimentación. Los únicos sistemas que se encuentran activos en todo momento son el oscilador astable con el integrado TS555 y el microcontrolador 18F25k22 en estado SLEEP.

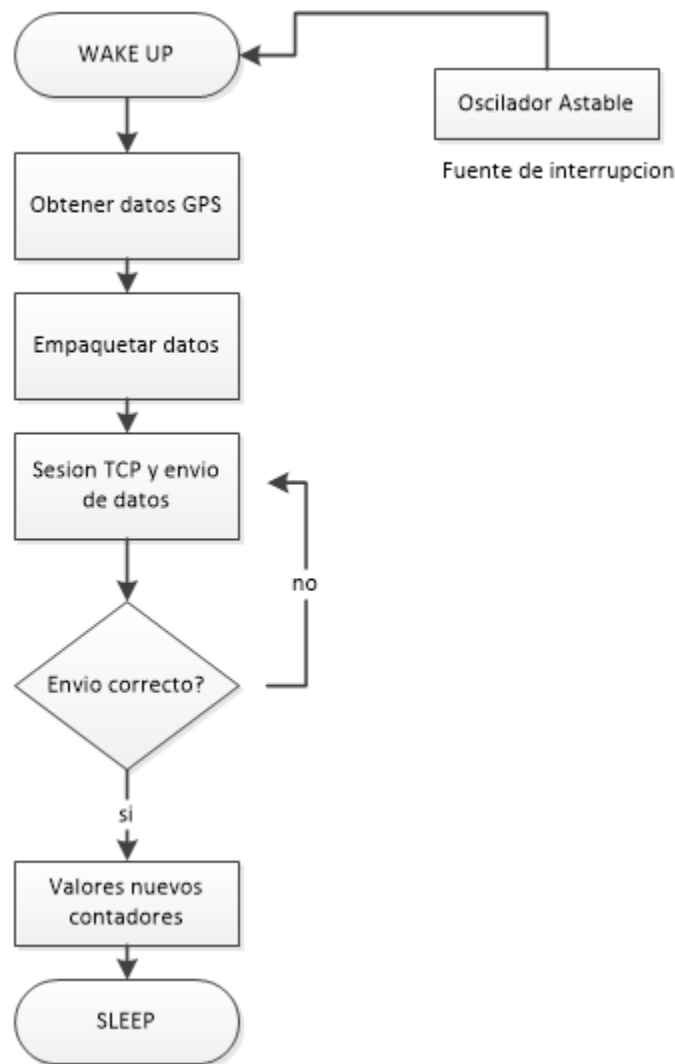
La base de datos se encuentra en espera a que la sesión de datos sea inicializada para así guardar las posiciones provenientes del dispositivo.

1. Comportamiento del dispositivo. El oscilador se encarga de llenar los contadores internos del microcontrolador para que ocurra un desborde y de esta forma pues “despertarse” y entrar en su modo operativo.

El microcontrolador al estar en modo activo proporciona las señales necesarias para que el módulo de GPS y GSM sean suministrados.

Con estos módulos ya activos, se entra a la rutina para extraer información proveniente del GPS. Se tiene un intervalo de tiempo en el que debe responder. Una vez se tiene la información del GPS se procede a empaquetar los datos para ser transmitidos mediante una sesión TCP utilizando el módulo GSM.

Figura 33. Diagrama de flujo del dispositivo para el modo de rastreo.



Estos datos son enviados a una base de datos donde se almacenan para su posterior despliegue.

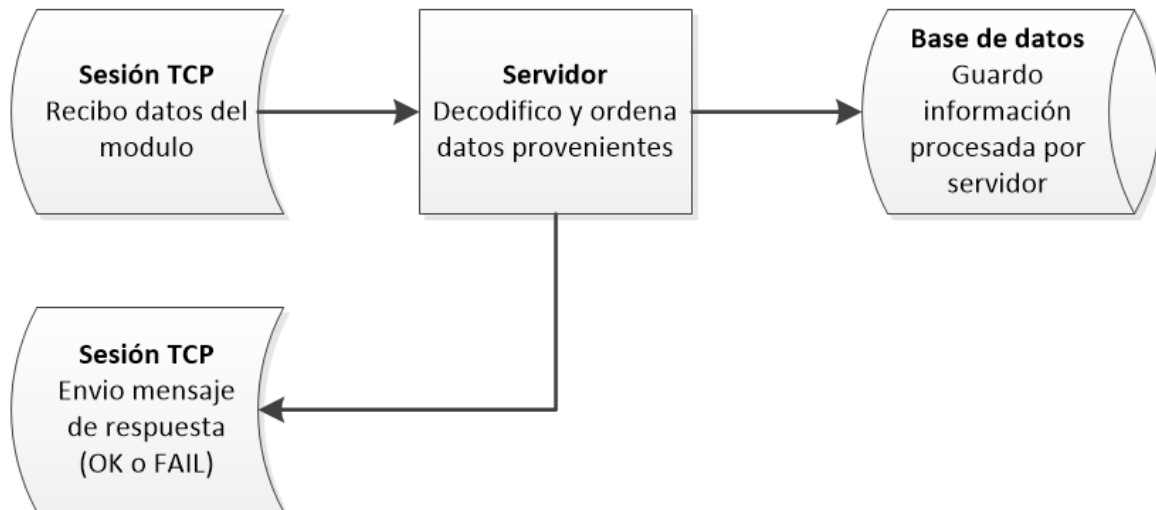
Si se creó una sesión de datos correcta y los datos fueron ingresados a la base de datos satisfactoriamente el servidor responderá con una secuencia de caracteres. Si no es así volvemos a enviar los datos hasta que obtengamos la respuesta.

2. Comportamiento plataforma web. La plataforma web se encuentra en modo escucha a que datos sean ingresados. Se tiene validaciones para saber de quién proviene los datos.

Un código desarrollado decodifica las señales providentes enviadas por el protocolo

de envío HTTP y de esta forma sabe cómo ingresar a la base de datos la información del GPS.

Figura 34. Diagrama de flujo de plataforma web en modo de rastreo.



Además de la latitud y longitud proporcionados por el GPS, la base de datos agrega la hora, fecha y año en el que fue ingresada esta información con el objetivo de poder tener una búsqueda más personalizada.

B. Funcionamiento en modo alarma

Este modo de funcionamiento varía del de rastreo en que no es activado periódicamente por un oscilador externo. Si no, es activado accionando un botón que lleva al microcontrolador a “despertarse” y ejecutar su código principal.

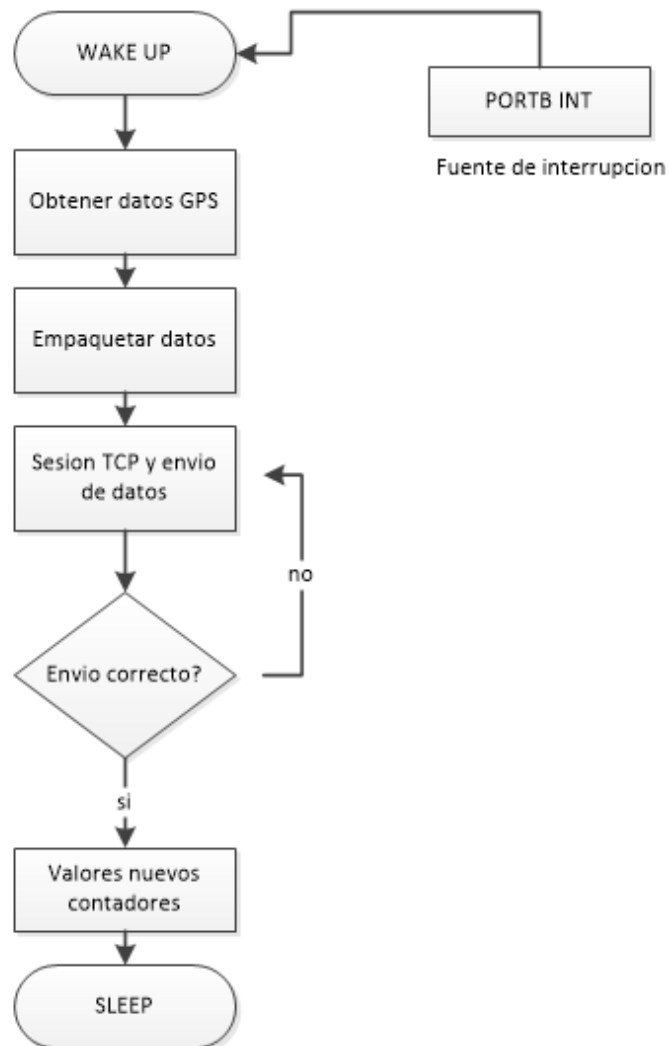
La intención de este funcionamiento es poder mandar una alerta que se encuentra en peligro la persona o que necesita algún tipo de asistencia. Se envía además un correo a la persona de interés previamente registrada en la base de datos con la información necesaria para ubicar a la persona.

1. Comportamiento del dispositivo. El microcontrolador es despertado al ser presionado el botón en el dispositivo. Este genera una interrupción con la que logra salir del modo SLEEP.

El microcontrolador al estar en modo activo proporciona las señales necesarias para que el módulo de GPS y GSM sean suministrados.

Con estos módulos ya activos, se entra a la rutina para extraer información proveniente del GPS. Se tiene un intervalo de tiempo en el que debe de responder. Una vez se tiene la información del GPS se procede a empaquetar los datos para ser transmitidos mediante una sesión TCP utilizando el módulo GSM.

Figura 35. Diagrama de flujo del dispositivo en modo alarma.



Estos datos son enviados a una base de datos donde se almacenan para su posterior despliegue.

Si se creó una sesión de datos correcta y los datos fueron ingresados a la base de

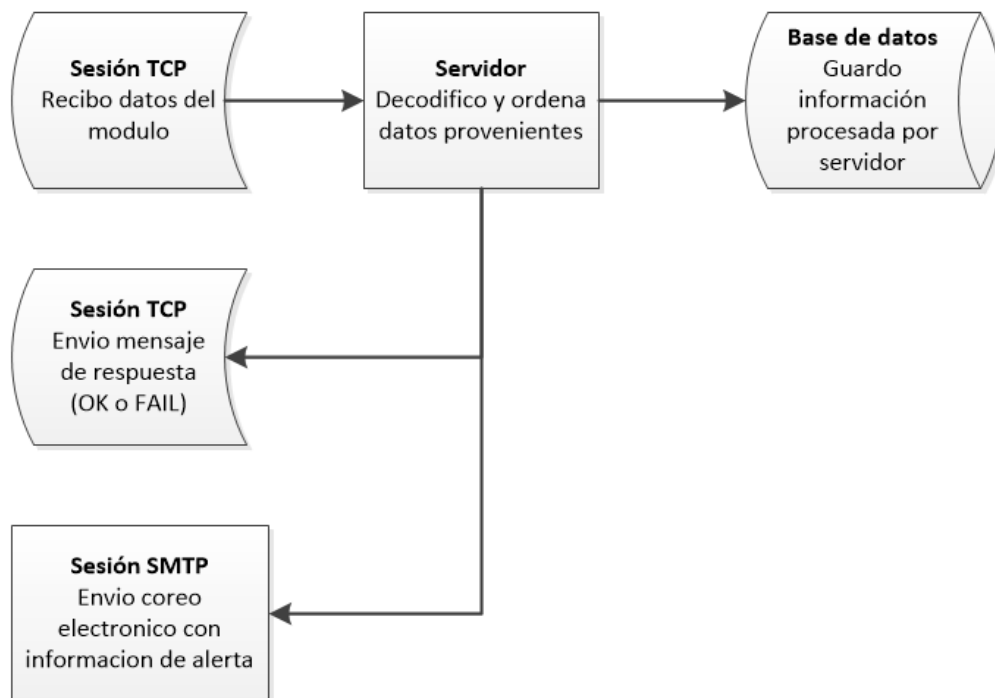
datos satisfactoriamente el servidor responderá con una secuencia de caracteres. Si no es así volvemos a enviar los datos hasta que obtengamos la respuesta.

2. Comportamiento plataforma web. La plataforma web se encuentra en modo escucha a que datos sean ingresados. Se tiene validaciones para saber de quién proviene los datos.

Un código desarrollado decodifica las señales provenientes enviadas por el protocolo de envío HTTP y de esta forma sabe cómo ingresar a la base de datos la información del GPS.

Además de la latitud y longitud proporcionados por el GPS, la base de datos agrega la hora, fecha y año en el que fue ingresada esta información con el objetivo de poder tener una búsqueda más personalizada.

Figura 36. Diagrama de flujo de plataforma web en modo alarma.



Se tiene programada como alerta que se envíe un correo al destinatario de confianza conocer donde se encuentra la persona y hacerle saber que necesita ser encontrada.

C. Funcionamiento de rastreo más alarma

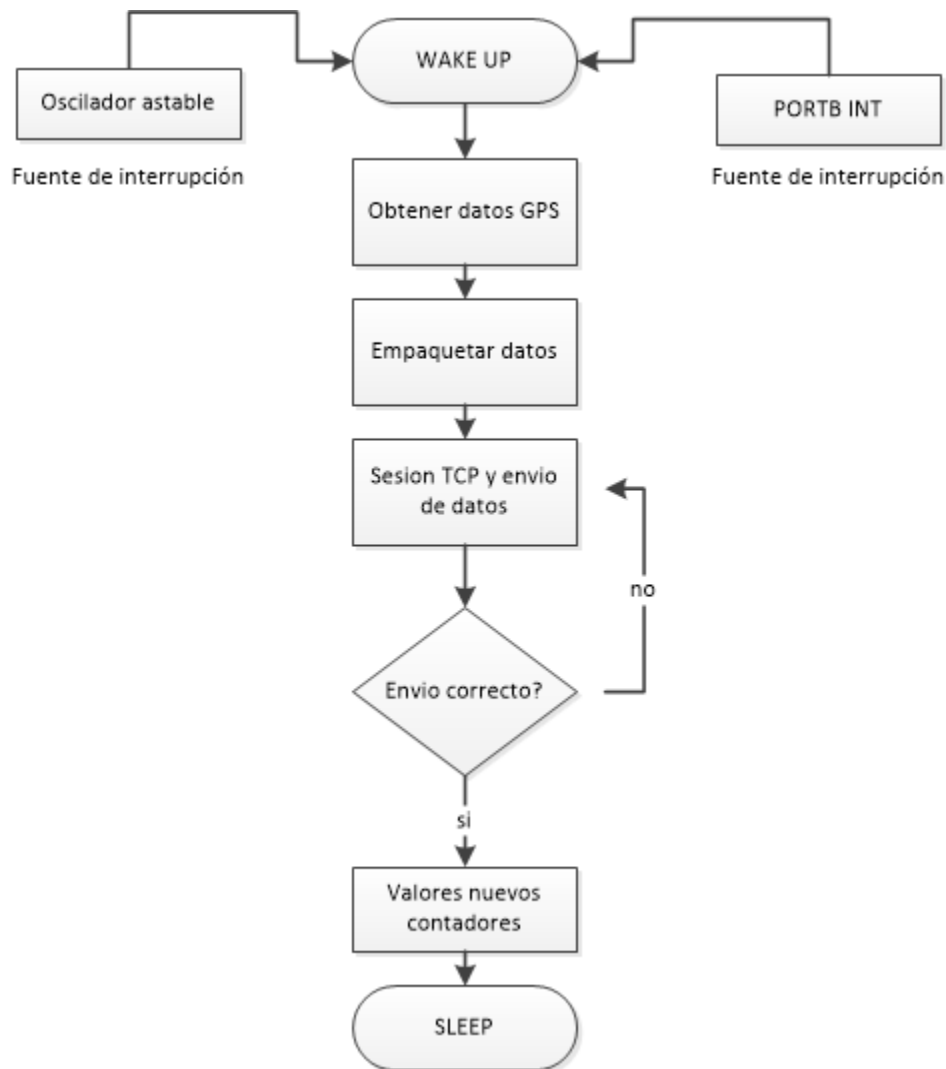
Esta opción en la combinación de los dos funcionamientos anteriormente descritos. Tanto el dispositivo estará enviando con cierta frecuencia, así como el usuario puede decidir manualmente cuando es que quiere que esta información sea transmitida.

1. Comportamiento del dispositivo. El microcontrolador es despertado por dos posible interrupciones. Debido al contador estable o bien por el accionamiento del botón en el módulo.

El microcontrolador al estar en modo activo proporciona las señales necesarias para que el módulo de GPS y GSM sean suministrados.

Con estos módulos ya activos, se entra a la rutina para extraer información proveniente del GPS. Se tiene un intervalo de tiempo en el que debe de responder. Una vez se tiene la información del GPS se procede a empaquetar los datos para ser transmitidos mediante una sesión TCP utilizando el módulo GSM.

Figura 37. Diagrama de flujo del dispositivo en modo rastreo más alarma.



Estos datos son enviados a una base de datos donde se almacenan para su posterior despliegue.

Si se creó una sesión de datos correcta y los datos fueron ingresados a la base de datos satisfactoriamente el servidor responderá con una secuencia de caracteres. Si no es así volvemos a enviar los datos hasta que obtengamos la respuesta.

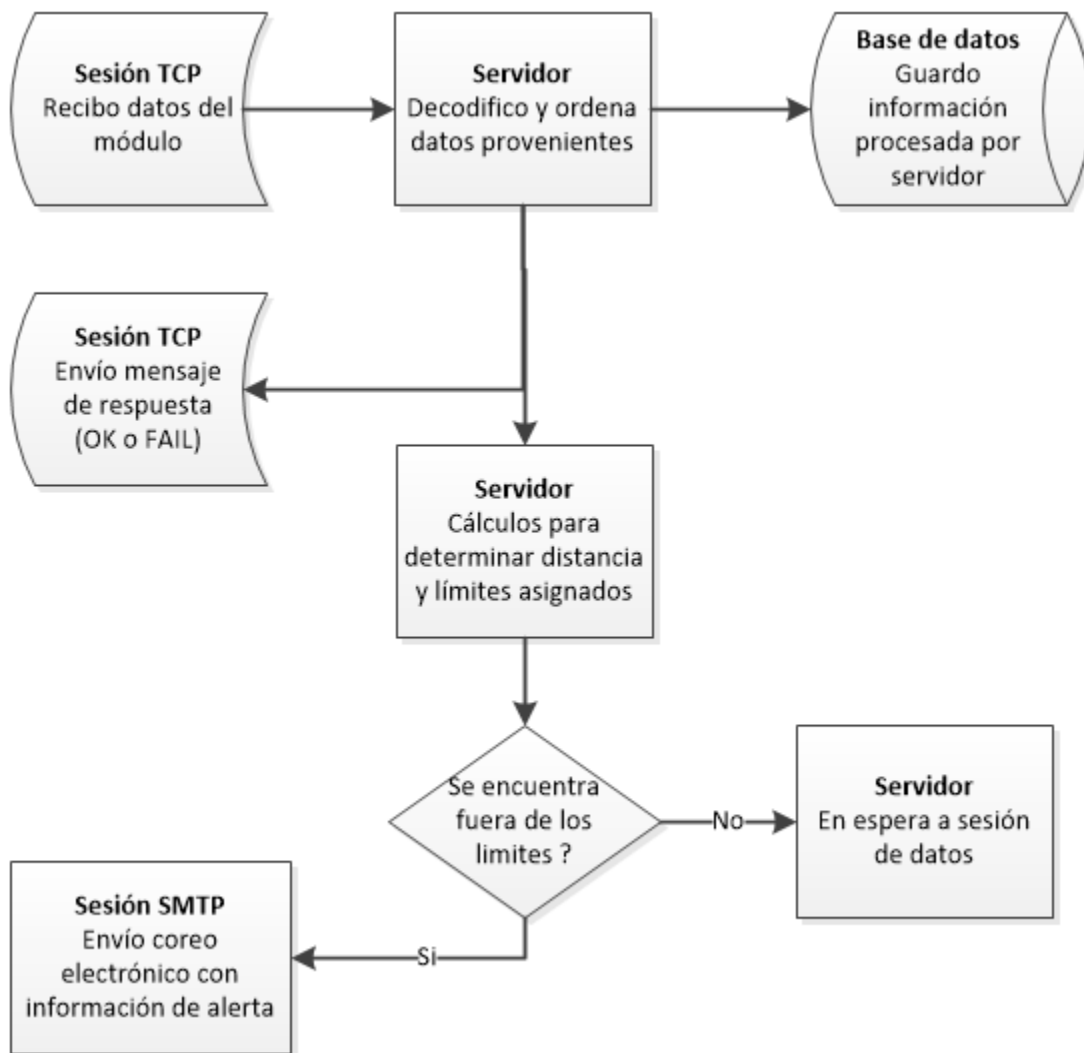
2. Comportamiento plataforma web. La plataforma web se encuentra en modo escucha a que datos sean ingresados. Se tiene validaciones para saber de quién proviene

los datos.

Un código desarrollado decodifica las señales provenientes enviadas por el protocolo de envío HTTP y de esta forma sabe cómo ingresar a la base de datos la información del GPS.

Además de la latitud y longitud proporcionados por el GPS, la base de datos agrega la hora, fecha y año en el que fue ingresada esta información con el objetivo de poder tener una búsqueda más personalizada.

Figura 38. Diagrama de flujo de plataforma web en modo de rastreo más alarma.



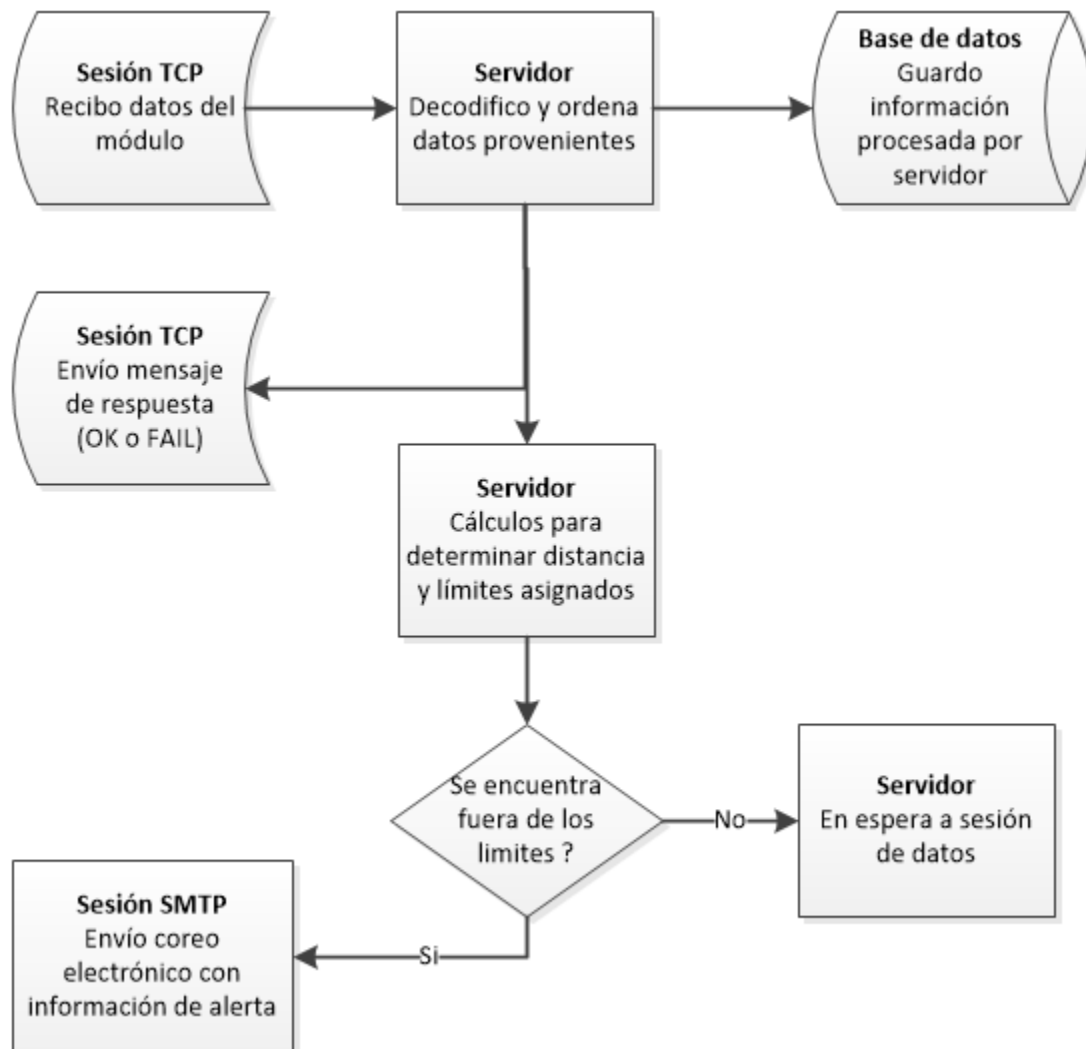
Si es que fue accionada la alarma mediante el botón se tiene programada como alerta

que se envíe un correo al destinatario de confianza conocer donde se encuentra la persona y hacerle saber que necesita ser encontrada.

D. Funcionamiento en Geofence

Este modo de funcionamiento es similar al de rastreo. Por parte del módulo realiza las mismas funciones basándose en la interrupción del oscilador astable para despertarse. Su diferencia está en la plataforma web en donde se despliega de manera diferente los datos ya que se traza un círculo y dependiendo de las últimas posiciones se puede determinar si se ha salido de los límites de este círculo utilizando la fórmula de Haversine.

Figura 39. Diagrama de flujo de la plataforma web en modo Geofence.

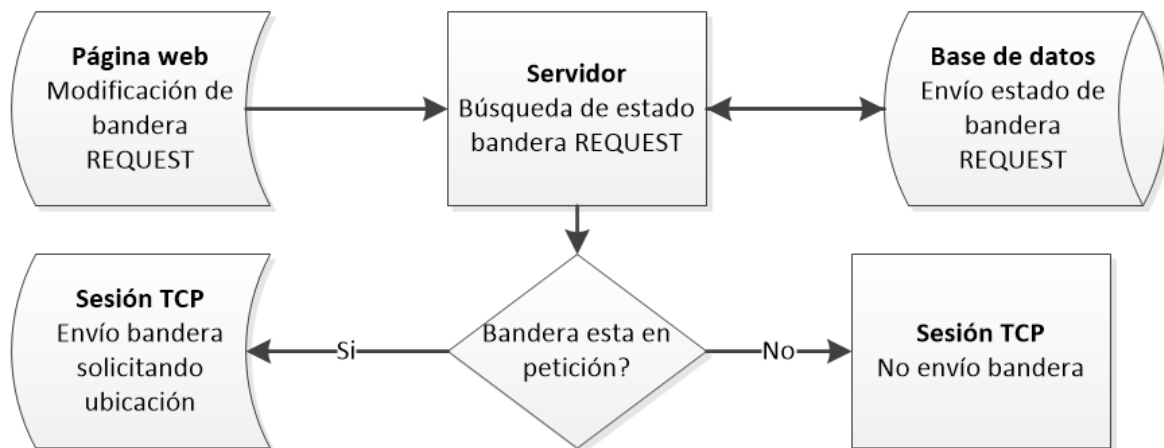


Esto con el objetivo de formar una valla virtual en donde el dispositivo debería de encontrarse y si no es así generar alertas a gente de confianza.

E. Funcionamiento en petición

Este modo de funcionamiento es similar al de rastreo. Por parte del módulo realiza las mismas funciones basándose en la interrupción del oscilador astable para despertarse. La diferencia más grande es que en este modo de operación no siempre estará enviando los datos del GPS, únicamente cuando se solicite en la página web. El módulo GSM enviara mensajes HTTP al servidor y este le responderá si tiene o no que enviar la ubicación. Si es así se empieza la secuencia para suministrar el GPS y obtener sus datos para posteriormente ser enviados.

Figura 40. Diagrama de flujo de la plataforma web en modo petición.



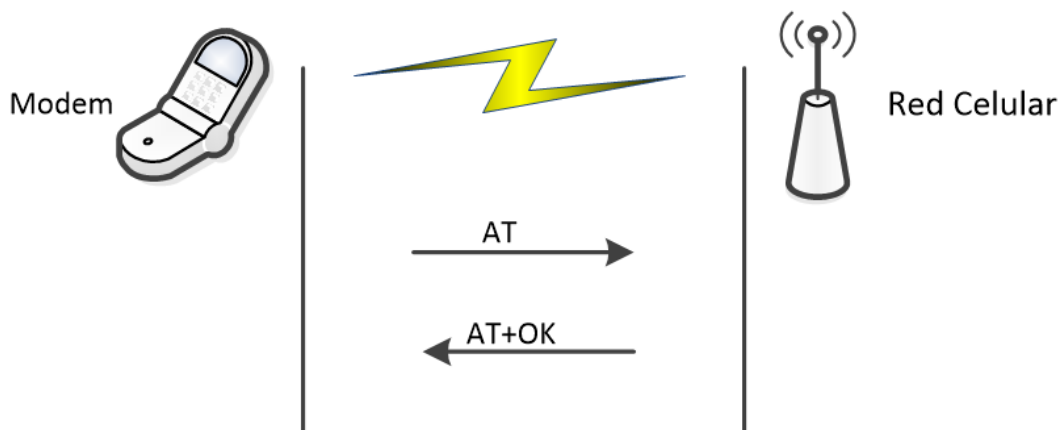
Se juega con una bandera llamada “request” en donde como lo indica si la petición está en alto debo de enviar mi ubicación, por lo contrario si está en un lógico cero el dispositivo vuelve a dormirse.

VIII. Comunicación y acceso a la red celular

A. Diseño

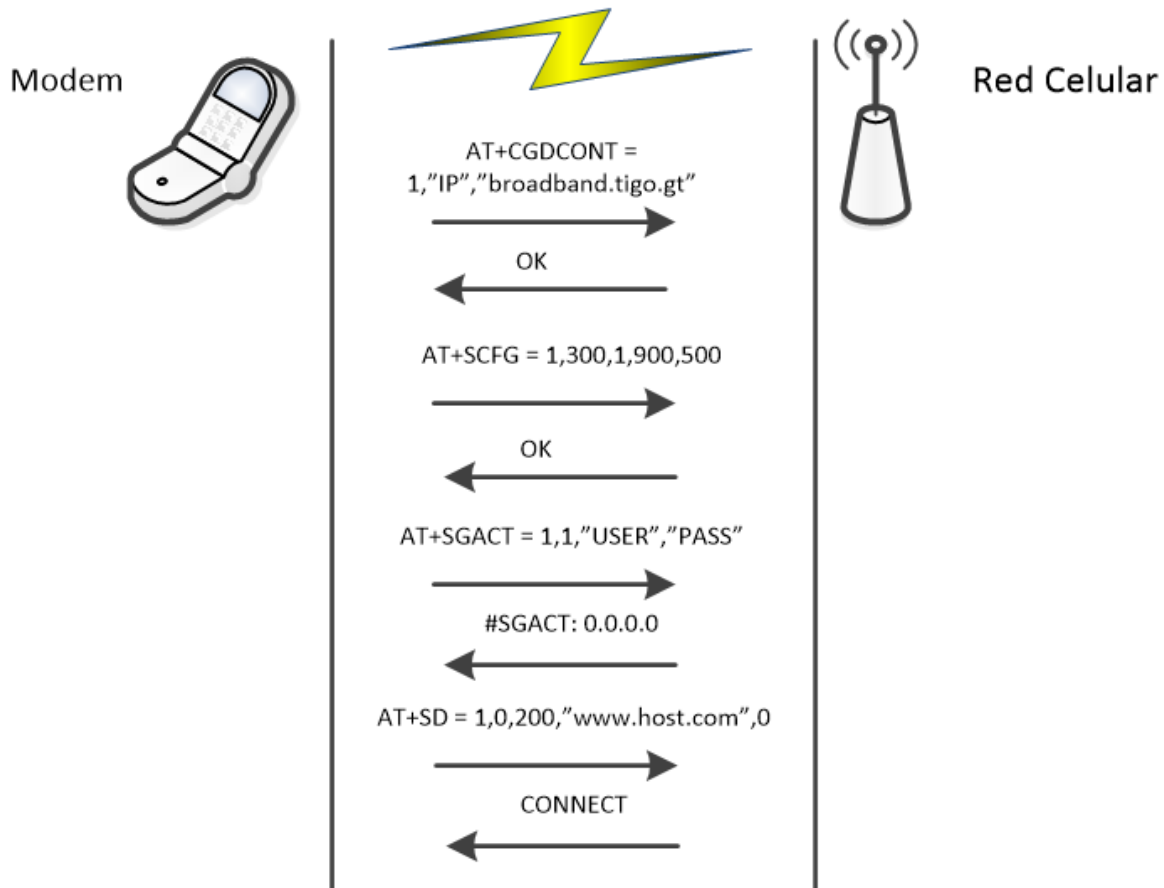
1. Handshake entre dispositivo GSM y red celular. Para establecer de primero comunicación con la red celular se envió el comando base hasta obtener respuesta por parte de la red. Una vez se tuvo respuesta (Los caracteres 'OK'), se procedió a enviar los demás comandos, siempre y cuando se recibía el comando base después de haber enviado alguna otra instrucción.

Figura 41. Handshake inicial entre módulo GSM y red de telefonía.



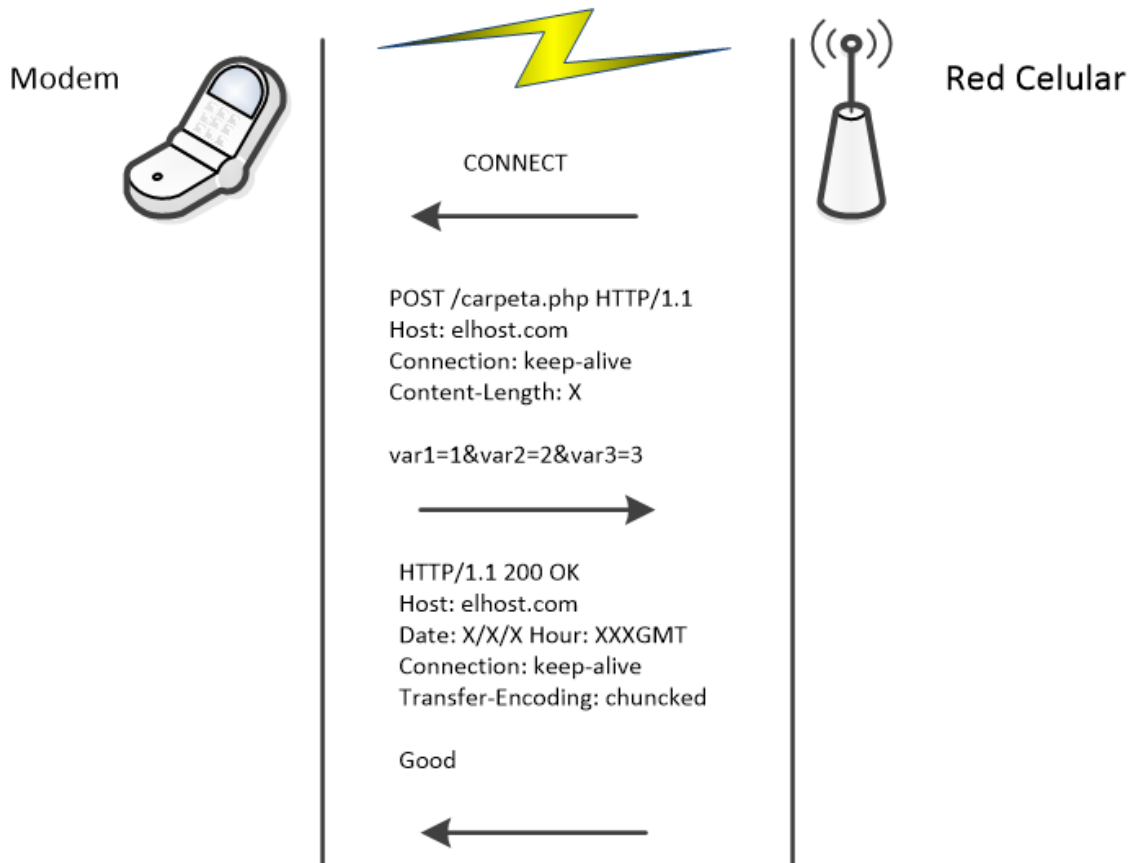
2. Handshake entre sesión de datos del módulo y red celular. Una vez se tuvo comunicación con la red celular, tanto protocolo y velocidad de transmisión se procedió a enviar los comandos necesarios para establecer una sesión de datos con algún servidor remoto. Los comandos se enviaron en el siguiente orden:

Figura 42. Diagrama de intercambio de información para configurar la sesión de datos en el módulo GSM.



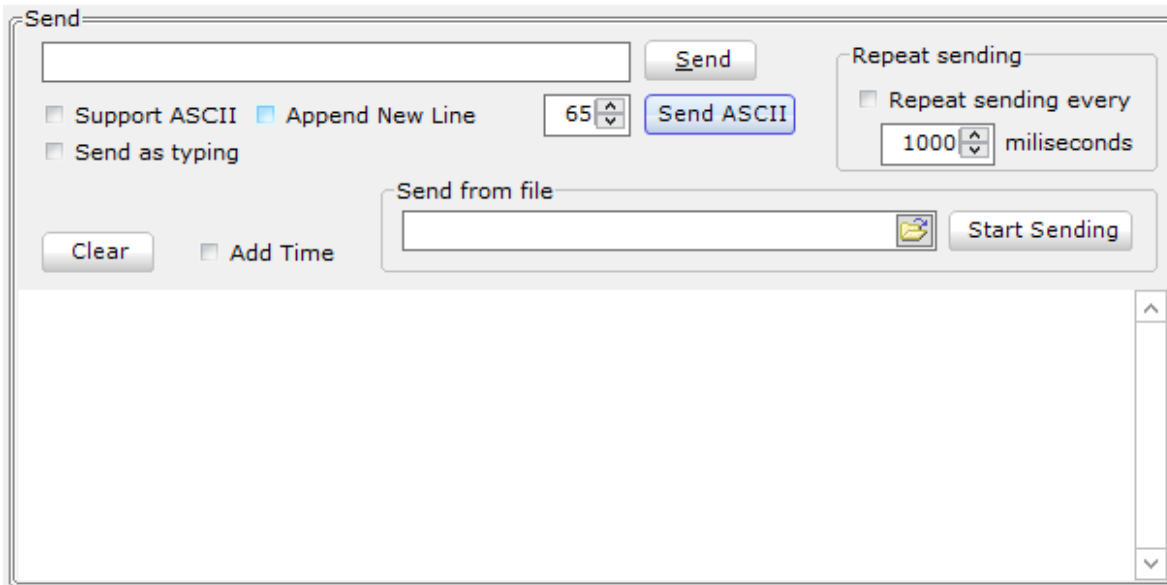
3. Envío de datos por medio del protocolo TCP. Una vez tenemos la respuesta de 'CONNECT', procedemos a enviar el paquete de datos. Los comandos ejecutados se ejecutarán en el siguiente orden:

Figura 43. Diagrama de intercambio de información para el envío de datos y respuesta del servidor.



4. Verificación de los comandos enviados y recibidos. Para realizar pruebas con la secuencia de códigos anteriormente descritos se conectaron dos cables de USB a serial. En el primero vemos el código que se está enviando por parte del módulo y en el segundo vemos las respuestas de la red celular. Se utilizaron las hyperterminales de MikroC a una velocidad de 9600bps, sin control de flujo y sin bit de paridad para la visualización de cada código.

Figura 44. Interfaz de hyperterminal en MikroC.



En la interfaz se cuenta con una opción llamada “Append New Line” la cual es utilizada ya que los comandos deben enviarse con un carácter de retorno de carro, de otra forma no sería reconocido el comando enviado.

B. Resultados

1. Handshake entre módulo GSM y red celular. Las siguientes imágenes fueron tomadas de la hyperterminal de MikroC para observar el flujo de datos entre ambas partes.

a. Comando enviado por el modulo a la red celular

Figura 45. Comando enviado por el microcontrolador a la red celular.

AT

b. Comandos enviados por la red celular

Figura 46. Comandos recibidos por la red celular.

AT

OK

2. Handshake entre sesión de datos del módulo y red celular. Las siguientes imágenes fueron tomadas de la hyperterminal de MikroC para observar el flujo de datos entre ambas partes.

c. Comando enviado por el módulo GSM a la red celular

Figura 47. Comandos para la configuración de sesión de datos.

```
AT+CGCLASS?
AT+CGDCONT=1,"IP","broadband.tigo.gt
AT#SCFG=1,1,300,90,600,50
AT#SGACT=1,1,"Victor V.,"Password :
AT#SD=1,0,2000,"network.me",0,0
```

d. Comandos enviados por la red celular

Figura 48. Activación del contexto en el módulo GSM.

OK

OK

OK

#SGACT: 10.69.105.60

OK

CONNECT

3. Envío de datos por medio del protocolo TCP

e. Comando enviado por el módulo GSM a la red celular

Figura 49. Comandos del protocolo HTTP para el envío de datos mediante POST.

```
POST /guardarPos HTTP/1.1
Host: netowork.me
Connection: keep-alive
Content-Length: 23

user=Hoaa&lat=23&long=4
```

f. Comandos enviados por la red celular

Figura 50. Respuesta del servidor al mensaje HTTP POST.

```
CONNECT
HTTP/1.1 200 OK
X-Powered-By: Express
Date: Fri, 26 Jul 2013 20:56:07 GMT
Connection: keep-alive
Transfer-Encoding: chunked

4
Good
0
```

C. Discusión

Es muy importante, a la hora de comunicarse con la red celular especificar que protocolo de comunicación y velocidad de transmisión que se está usando. Estos parámetros se especifican internamente en el módulo GSM. Si esta configuración no es posible, la redes de telefonía de hoy en día son capaces de reconocer estos dos parámetros automáticamente. Para la comunicación mencionada en la parte de diseño se utilizaron los comandos AT y un baud rate de 9600 los cuales no los configuramos en ningún momento dentro del módulo GSM simplemente esperamos a que la red respondiera por sí sola.

Una técnica para realizar esto es enviar el comando base muchas veces por intervalos de tiempo pequeños hasta que recibamos por parte de la red celular el mensaje de 'OK'. En este punto el Handshake para especificar protocolo y velocidad ya están establecidos.

Cuando ya se tiene el protocolo y velocidad establecida se debe de esperar a que la red celular responda con los caracteres 'OK' antes de volver a enviar algún otro comando. La respuesta 'OK' es un indicador que nos dice que ya termino de ejecutar el código solicitado. Por ejemplo hay código que demoran hasta diez segundos en ejecutarse y hasta que la red responda con 'OK' podemos seguir dialogando con la red. De forma contraria no se tiene una comunicación adecuada y la generación de errores es inevitable.

Se utilizaron dos cables en vez de solo uno debido a que la plataforma EasyPic 7 que se estaba utilizando causaba que los datos recibidos nos fuesen legibles. Esto se debió a que para comunicarse con la computadora se cambiaron de estado unos interruptores, más específicamente los del integrado MAX232, que eleva el voltaje. Esta elevación de voltaje hacia que en vez de tener una señal de 0 a 3.3 volts, obtuviéramos -15 a 15 volts. Este no es un rango de operación para el módulo GSM y por parte de la hyperterminal no se tenía recepción de ningún dato. Por eso se utilizaron dos para aislarlos y con este procedimiento fue posible la visualización en la hyperterminal de los datos enviados y recibidos.

IX. Programación del microcontrolador 18F25k22

En esta sección se describe a detalle como el código del microcontrolador fue diseñado y programado. Se explica también como se configuraron sus diferentes módulos internos para que respondieran de acuerdo a esta aplicación. El código final de la aplicación se encuentra en la sección de anexos.

A. Diseño

1. Código fuente base. Para esta aplicación se utilizó la plataforma GSM click de Mikroelectronica. El fabricante recomienda el uso de ciertos programas ejemplos para que sea más fácil el desarrollo de aplicaciones basadas en las redes celulares.

Se estudió el ejemplo recomendado por Mikroelectronica el cual está alojado en la página www.Libstock.com. Este ofrece dos ejemplos muy útiles tanto para enviar mensajes de texto (SMS) así como realizar y recibir llamadas telefónicas.

Una vez estudiado el código se procedió a modificarlo para que respondiera a las necesidades de este trabajo. Las únicas secciones que quedan del código original son los métodos para el envío por medio de USART y parcialmente el parser o decodificador de los mensajes recibidos por serial.

2. Configuración del reloj interno. Para esta aplicación se eligió un reloj interno de 64Mhz, esto fue logrado utilizando el bloque de oscilador interno del microcontrolador. Este bloque interno puede generar un reloj de 16Mhz, para lo cual utilizamos otra característica del microcontrolador llamada PLL que permite multiplicar por factores de x1, x2, x3 y x4 la frecuencia ya sea interna o externa. En este caso el oscilador interno de 16Mhz es multiplicado por el factor x4 del módulo PLL para obtener una frecuencia de operación de 64Mhz.

3. Comunicación serial. Debido a que se tiene comunicación con la red celular y el GPS fue necesario utilizar dos módulos seriales para la establecer comunicación con ambos sistemas.

a. Puerto serial 1. El puerto serial 1 fue configurado a una velocidad de 9600bps para comunicarse con la red celular.

Figura 51. Inicialización del puerto serial 1.

```
UART1_Init(9600);
```

Se utilizó la inicialización de MikroC para que el módulo GSM fuese operativo.

b. Puerto serial 2. El puerto serial 2 fue configurado a una velocidad de 57600bps para comunicarse con el modulo GPS.

Figura 52. Inicialización del puerto serial 2.

```
UART2_Init(57600);
```

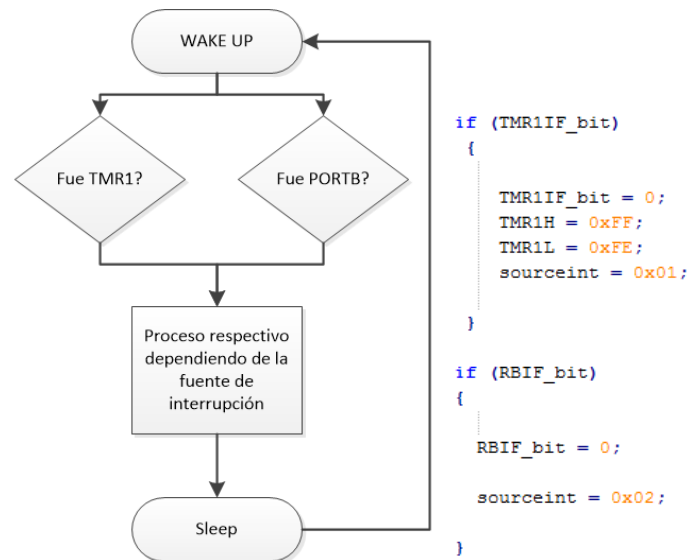
Se utilizó también la inicialización de MikroC sin necesidad de configuraciones adicionales.

4. Rutinas recursivas. En todo el código ejecutado por el microcontrolador se tienen dos rutinas principales encargadas de conectarse a la red celular y obtener datos del GPS. Adicionales a estas dos se tiene la principal o “Main” en donde es llamado a dormir el microcontrolador.

a. Rutina MAIN. La rutina MAIN como lo indica es la principal y en ella se contienen las subrutinas para lograr la función de esta aplicación.

Como de talle a observar, la rutina al principio de su código cuenta con la instrucción ‘SLEEP’ para que entre a su modo de ahorro de energía. Se queda a la espera de alguna interrupción para ser despertado y realizar sus funciones.

Figura 53. Diagrama de flujo general del microcontrolador e identificación de las fuentes de interrupción.



Como podemos observar de las imágenes anteriores, el microcontrolador estará en modo SLEEP hasta que una interrupción intervenga. Cuando se entra al servicio de interrupción se verifica de quien fue la interrupción mediante la variable 'sourceint'. Esta tomará el valor de '1' o '2' dependiendo de quién fue el responsable de activarla.

Una vez tenemos identificado quien provoco la interrupción, en el código main se determina a que condición se debe ir para realizar para realizar las funciones de la aplicación.

Figura 54. Identificación de la fuente de interrupción en la sección principal.

```

//Rutina MAIN
while(1)
{
    asm Sleep;
    //Fue TMR1
    if(sourceint==0x01)
    {
        //Codigo respectivo
    }
    //Si no fue tmr1 fue PORTB
    else
    {
        //Codigo respectivo
    }
}

```

Como vemos en el código anterior la rutina MAIN siempre estará en ejecución y la primer instrucción que ejecutara es la de dormirse o SLEEP.

b. Método INIT_COM_GPRS. Como su nombre lo indica es el encargado de comunicarse con la red celular y establecer la sesión de datos con el servidor. Cuenta con todas las partes necesarias como lo es identificación de mensajes recibidos, comunicación con mensajes AT a la red celular y envío del mensaje HTTP al servidor.

1) *Comandos para la conexión con la red celular.* Debido a que se basó en conexión de datos, los comandos AT para el acceso a la red celular fueron solamente 3:

Figura 55. Comandos AT para el acceso a la red celular.

```
const char atc0[] = "AT"           //Atencion
const char atc1[] = "ATE0"        //Comando echo
const char er0[] = "AT+CMEE=1"    //Error extendido
```

2) *Comandos para la configuración de datos.* La configuración para sesión de datos es más compleja y lleva parámetros tanto de hardware como software. Para establecer conexión con un servidor es necesario llamar a la siguiente rutina y enviarla por la terminal serial:

Figura 56. Comandos AT para configuración de la sesión de datos en el módulo GSM.

```
a1[] = "AT+CGDCONT=1,\"IP\", \"broadband.tigo.gt\"
a2[] = "AT#SCFG=1,1,300,1,900,50"
a3[] = "AT#SGACT=1,1,\"User\", \"Pass\"
a4[] = "AT#SD=1,0,80,\"vicvicvar.com\",0,0"
```

Si el módulo GSM se conectó correctamente obtendremos un mensaje de CONNECT lo cual indica que estamos listos para intercambiar datos con el servidor contactado.

3) *Comandos para el envío del paquete HTTP.* Una vez se tiene conexión con el servidor debemos de enviar los datos de la forma en que el módulo GSM acepta. Se tiene un formato estándar para realizar transferencia de datos mediante HTTP el cual vemos en la siguiente imagen:

Figura 57. Arrays contenedores de la estructura del método POST en el microcontrolador.

```
//Codigo para insertar datos
const char bc1[] = "POST /parser.php HTTP/1.1"      ;
const char bc2[] = "Host: tesis.vicvicvar.com"      ;
const char bc3[] = "Connection: keep-alive"        ;
const char bc4[] = "Content-Length: 47"            ;
const char bc6[] = ""                               ;
const char bc7[] = "user=LM7474&lat=LAT&lon=LON&org=0" ;
```

4) *Método GM862_Send.* Todos los comandos anteriormente mencionados (atc, bc, ax) son almacenados en variables de tipo arreglo de caracteres. En las imágenes proporcionadas podemos ver su declaración mediante corchetes y su tipo de dato con las comillas antes y después de todo el texto. Estos caracteres son enviados mediante el método GM862_Send(atc) el cual simplemente se envía mediante el puerto serial nuestro comando y le agrega al final un carácter de retorno de carro (Carriage return) el cual se identifica con el ASCII 0x0D.

Este último comando es necesario ya que la red celular termina todos sus comandos con este carácter de retorno de carro.

Figura 58. Método de envío de caracteres por el puerto serial.

```
void GM862_Send(const char *s)
{
    while(*s) {
        UART_Wr_Ptr(*s++);
    }

    UART_Wr_Ptr(0x0D);
}
```

Como vemos el método recibe la serie de caracteres y los envía uno por uno por el puerto serial (UART) para al final agregarle un carácter extra e indicarle a la red celular que ese comando ya fue completado y que no espere caracteres adicionales.

c. *Método GPS_ON_OFF*. El método se encarga de esperar cierto de periodo de tiempo para obtener datos del GPS. Se tiene un contador el cual después de cierto tiempo de espera se incrementa en uno. Si este contador llega a un valor específico y todavía no se tiene respuesta del GPS la función principal no puede continuar y el microcontrolador es puesto a dormir de nuevo.

Figura 59. Rutina del GPS para obtener las coordenadas geográficas.

```
void GPS_ON_OFF()
{
    gpstry=0x00;
    gps_fail_ok = 0x00;

    while(gpstry !=0x04 )
    {
        Delay_ms(10000);
        if(gpsdataok == 0x01)
        {
            gpsdataok = 0x00;
            gpstry = 0x04;
            //GPS OK
            gps_fail_ok = 0x01;
        }
        else
        {
            //GPS Fail
            gpstry++;
        }
    }
}
```

Como vemos se tiene la variable 'gpstry' que ve cuantas veces se ha intentado obtener datos del GPS. El código se encuentra en un ciclo el cual se ejecuta mientras no se haya llegado a la cuenta máxima de intentos.

Vamos a obtener datos validos del GPS cuando este se enganche a los satélites necesarios. Al hacerlo envía unos varios comandos el cual de más interés es el GLL que incluye las posiciones geográficas.

El decodificador se encarga de verificar que hayan llegado correctamente datos. De lo contrario no modifica la variable 'gpsdataok' e incrementa en uno los intentos. En este

imagen se tiene un tiempo de espera de 10 segundos y 4 intentos para enganchar con el GPS. Este tiempo e intentos son configurables para tener mayor tiempo de espera y darme más oportunidad al GPS.

5. Interrupciones. La presente aplicación se basa en el uso de 4 interrupciones las cuales sirven para comunicar el microcontrolador a módulos externos y para marcar eventos importantes para su operación.

a. Interrupción PORTB. La interrupción de puerto B fue configurada para reaccionar a cambios en el pin 4 y corresponde al botón de alarma de pánico.

La configuración fue la siguiente:

Figura 60. Configuración para la interrupción del puerto B.

```
//Configuracion PORTB interrupt
TRISB.f4=1;
RBIE_bit = 1;
IOCB.IOCB4 = 0x01;
```

Como vemos el puerto 4 lo configuramos como entrada modificando su bit del registro TRISB. Siguiendo habilitamos la interrupción del puerto B y por ultimo indicamos que puerto puede accionar la interrupción, en este caso con el registro IOCB modificamos su cuarto bit haciendo uno.

b. Interrupción TMRI. La interrupción del Timer 1 fue configurada para que reaccionara a cambios en RC0, el cual es la entrada al reloj correspondiente del Timer ya que se programó para que fuese externo. Este recibe las señales de un oscilador astable externo y la interrupción del TMR1 es llamada cuando existe un desborde en sus registros TMR1L:TMR1H que incrementan en uno cada vez que el reloj externo mande. La configuración programa fue la siguiente:

Figura 61. Configuración para el Timer1.

```

//Configuracion timer 1
TRISC.F0 = 1;           //Entrada para TMR1
T1CON = 0x00;          //Limpio variables del timer1
TMR1GE_bit=0;          //Apago el gate
TMR1ON_bit = 1;        //Enciendo timer
TMR1CS1_bit = 1;       //Reloj externo
TMR1CS0_bit = 0;       //Reloj externo
T1SOSCEN_bit = 0;      //Oscildor secundario
T1SYNC_bit = 1;        //No sincronizar
T1RD16_bit = 0;        //2 registros de 8 bits
TMR1H = 0xFF;          //Valores iniciales
TMR1L = 0xFE;          //Valores iniciales
TMR1IE_bit = 1;        // enable Timer1 interrupt

```

Como vemos en la configuración seleccionamos que el reloj de cuenta sea externo el cual será reconocido en el puerto RC0. Además se configuro para que sean dos registros los que acumulen la cuenta.

c. Interrupción serial 1 y serial 2. Se configuraron las interrupciones de los puertos seriales las cuales serán llamadas cuando reciba un carácter.

Figura 62. Configuración para las interrupciones.

```

// Habilitacion
RC1IE_bit = 1;
RC2IE_bit=1;

```

Como se ha mencionado, el puerto serial 1 es dedicado a la comunicación con la red celular y el puerto serial 2 es utilizado para la comunicación con el GPS.

6. Activación de PORTC para relés de estado sólido. En el diseño del PCB se cuentan con relés de estado sólido que suministran la energía a los módulos principales. Para darles su estado de activación el puerto C fue configurado como salida a las entradas de estos relés:

Figura 63. Activación de módulos y sub módulos mediante PORTC.

```

PORTC=0b0000010;

```

Por ejemplo para la activación del puerto 1, se alimenta el relé del módulo GSM y este se vuelve activo. Al terminar de usar el dispositivo se limpia todo el puerto y el único sistema suministrado es el microcontrolador.

B. Discusión

El microcontrolador 18f25k22 fue elegido debido a diversos factores. Entre los principales esta que el código fuente recomendado por Mikroelectronica se basa en integrados de la familia 18F, más específico en el 18F45k22. La migración de una serie más alta a una más baja (18F25k22) no fue problema ya que se mantuvo la relación entre puertos seriales y puertos de servicio.

Se eligió además este microcontrolador por que tiene un número menor de pines (28) lo que lo hace más compacto que es lo que busca esta aplicación.

Se escogió el reloj interno del microcontrolador ya que el sistema final pretende ser de tipo portable y debido a esto agregar más componentes a la placa la haría potencialmente más grande. Por lo que se optó por un oscilador interno y no externo para no tener que interconectar más componentes y obtener una placa más pequeña.

El puerto serial 1 se configuro con una velocidad de 9600bps la cual podría haber sido mayor ya que el módulo GSM soporta mayores velocidades. Pero se optó por esta velocidad ya que es un estándar en muchas de las comunicaciones de este tipo. Además el flujo de datos es muy pequeño por lo que una velocidad muy rápida no hubiera sido significativa.

El puerto serial dos se configuro a una velocidad de 115000bps debido a que el GPS ya trae por defecto esa velocidad de comunicación. Si bien esta es configurable el fabricante recomienda no modificarla ya que el modulo GPS puede llegar a dejar de funcionar. Por lo tanto se trabajó con la velocidad por defecto.

En muchas partes de la programación se tienen dos condiciones a evaluar, como por ejemplo para saber quién ocasiono la interrupción se tendría que hacer dos IF y ver quien el responsable. Estas condiciones se mapean a una serie de instrucciones las cuales

ocupan mucho espacio del programa en el microcontrolador y ya que MikroC tiene un límite (Este es un límite de código en el cual el programa puede ser compilado y descargado al microcontrolador sin tener que pagar por una licencia) se debió de buscar la forma de limitar el uso del programa. Debido a esto en vez de usar IF para toda la evaluación, agregamos a la evaluación final el comando ELSE ya que este no tiene ninguna evaluación que hacer, simplemente nos lleva a otro pedazo de código.

Así como esta técnica de utilizar un ELSE, se modificaron los métodos para que solo fuesen 2 principales. En un principio se tenían 6 métodos con el objetivo que el programa fuese más ordenado, pero conforme se fue aumentando la programación el programa no pudo ser compilado y descargado debido a que se llegaba al límite permitido por MikroC, por eso se unificaron los métodos los cuales al parecer ocupan bastante espacio en el programa al ser mapeado a su lenguaje ensamblador.

El TMR1 fue configurado con dos registros de 8 bits para que se tuviera mayor rango de tiempo en el cual deseamos que el programa se desborde. Para efectos de ejemplo se tienen los contadores a un solo incremento de desbordarse, pero ya aplicado no queremos que esto pase, sino queremos que sea periódica esta señal de desborde. Teniendo dos registros esta frecuencia de eventos será mejor controlada.

X. Módulo de posicionamiento global GPS

A. Diseño

Para la elección del chip designado a ser el modulo GPS se consideraron las siguientes variables propias de estos sistemas:

- Número de canales: para que el tiempo de primer enlace (time to first fix) fuese muy pequeño.
- Voltaje de operación: considerando el voltaje disponible en el PCB, el modulo debe de ser operable a 3.3 voltios.
- Tipo de antena: que sea de tipo embebida para no tener cables adicionales.
- Precisión: el modulo GPS debe de tener una precisión de por lo menos de +/- 20 metros y debe de operar en ambientes extremos donde la señal de los satélites sea de menor potencia (Adentro de casas, áreas urbanas, etc).
- Tamaño: debe ser de pequeñas dimensiones para lograr el objetivo de poder ocultarlo en una prenda.

De todos los módulos GPS disponibles en el mercado, que fuese posible su adquisición por personas particulares (en cantidades pequeñas) y que tuviera todas sus partes integradas, el que completaba muy bien las especificaciones fue el GPS LS20031 de la marca LOCOSYS.

Este módulo de GPS presenta las siguientes características:

- Voltaje de operación de 3.3V @ 41mA.
- 66 canales para vista de satélites.
- Dimensiones de 3cm*3cm.
- Memoria de satélites vistos anteriormente.
- Precisión de hasta +/- 5 metros con 12 satélites enganchados.

Figura 64. Modulo GPS LS20031 utilizado.



Se procedió a ordenarlo de su proveedor indirecto SparkFun Electronics y se realizaron pruebas para comprobar su funcionamiento.

B. Resultados

Alimentado el modulo, se configuro un puerto serial y se observó la trama de datos la cual se despliega de la siguiente forma:

Figura 65. Formato NMEA obtenido del módulo LS20031.

```
$GPGLL,1436.3179,N,09029.2861,W,001032.800,A,A*40
$GPGSA,A,3,24,29,02,15,21,22,,,,,,,,,1.89,1.27,1.40*0B
$GPGSV,2,1,07,29,68,211,50,51,64,230,46,15,62,034,36,21,34,335,28*73
$GPGSV,2,2,07,24,31,135,44,22,13,273,41,02,07,117,37*4A
$GPRMC,001032.800,A,1436.3179,N,09029.2861,W,0.00,118.33,051013,,,A*79
$GPVTG,118.33,T,,M,0.00,N,0.01,K,A*34
$GPGGA,001033.000,1436.3179,N,09029.2861,W,1,6,1.27,1536.9,M,-2.8,M,,*61
```

De todas las tramas la que nos interesa es la que empieza con \$GPGLL ya que continua a esta se encuentra la latitud y longitud.

C. Discusión

Se enfatiza que el modulo pueda ser adquirido por personas particulares ya que si uno ingresa a la página del fabricante, su venta es realizada solo a socios o clientes

registrados que hacen pedidos de miles de módulos por lo que no es una forma viable de obtener un GPS. Además se habla de que debía de incluir todas sus partes integradas ya que por lo general el fabricante solo vende el circuito integrado y a este le debemos de sumar su antena la cual el acople debe ser el correcto para no tener pérdidas. Por lo que lo ideal es trabajar con módulos ya ensamblados.

Para la prueba del módulo ocurría un caso extraño en el cual no enganchaba satélites si tenía carga en su línea de TX. Por lo que si queríamos observar que información nos entregaba el GPS, era necesario de primero alimentarlo y después de aproximadamente 2 segundos ya era posible la conexión del cable serial a USB para ver sus datos provenientes. Si se hacía paralelamente esta operación no indicaba operación y no se obtenían las sentencias NMEA.

XI. Plataforma web

En esta sección se describe como fue el diseño y la implementación de la plataforma web y sus demás secciones describiendo funcionalidad y usos.

A. Diseño

1. Lenguajes de programación utilizados. Para lograr esta plataforma web se utilizaron tres lenguajes de programación orientados principalmente al desarrollo de páginas web y de aplicaciones dinámicas dentro de la página. Se utilizó adicionalmente un cuarto lenguaje llamado Structured Query Language para comunicarse con la base de datos.

a. Lenguaje HTML. Se utilizó este lenguaje para el despliegue de la interfaz de opciones y como encargado de contenido. Este lenguaje pertenece a lado visual de la plataforma web y es lo que descarga el navegador para visualizar su contenido.

Se programó como encargado de contenido desplegando textos e imágenes relacionadas con la plataforma. Se programaron secciones para que pueda alojar el código del API de Google para visualizar los mapas ya que este soporta incrustaciones de otros lenguajes como lo es PHP y JavaScript.

b. Lenguaje PHP. Este lenguaje fue utilizado para realizar las tareas lógicas, aritméticas y conexión con la base de datos. Este lenguaje corresponde al lado del servidor y su código no es accesible públicamente por lo que en él se alojan datos importantes como credenciales a las bases de datos.

Su herramienta de formularios se utilizó en la mayoría de secciones para intercambiar información entre carpetas y así interactuar entre ellas. Este lenguaje a comparación de HTML permite tener un código ejecutándose en el servidor y un ejemplo del código que se programo fue que recibiera las variables enviadas por los formularios y que las asignara correctamente para que se almacenaran en la base de datos.

c. Lenguaje JavaScript. Este lenguaje se utilizó únicamente debido a que el API de Google Maps se basa en el, por lo que ninguna programación esencial fue hecha en el más que como servicio de despliegue de la API.

2. Alojamiento de la página. Debido a que se requiere que la plataforma web sea accesada en todo momento se compró un servicio de alojamiento de la página BlueHost.com, en ella se ofrece un servidor compartido con disponibilidad las 24 horas al día y muchas herramientas útiles para la aplicación que se pretende.

3. Programación de la interfaz. Para el diseño de la página web se utilizó la herramienta llamada Content Manager la cual permite agregar objetos como imágenes, texto, vínculos, audio, etc. A la página web tan simple como un “drag & drop”. Esto ahorra mucho tiempo ya que no tenemos que conocer tan a fondo lenguajes como HTML o JavaScript para lograr páginas con diseños aceptables.

El servicio de alojamiento adquirido proporciono esta herramienta para poder desplegar de una forma atractiva la interfaz web.

En la página se tienen diferentes servicios los cuales son utilizados con el fin de localizar o controlar el dispositivo en discusión. Estos servicios son MARKER, PARTH, GEOFENCE y REQUEST.

En la página principal de la interfaz se despliegan pestañas con los servicios mencionados a las cuales se puede acceder seleccionando los parámetros para cada servicio.

Para una referencia más detallada avocarse a la sección de anexos donde esta explicado cada parte del código fuente de los servicios.

a. Servicio Marker. En este servicio el programa obtiene la información ingresada en su sección, busca en la base de datos y si encuentra algo despliega en pantalla las ultimas 5 posiciones encontradas.

La información ingresada en su sección se refiere a campos para ingresar el rango de

horas en los que se desea saber la ubicación, el día que se desea saber la ubicación y el mes de interés.

b. Servicio Path. En este servicio el programa obtiene la información ingresada en su sección, busca en la base de datos y si encuentra algo despliega en pantalla las últimas 5 posiciones encontradas uniéndolas con líneas para indicar el recorrido.

La información ingresada en su sección se refiere a campos para ingresar el rango de horas en los que se desea saber la ubicación, el día que se desea saber la ubicación y el mes de interés.

c. Servicio Geofence. En este servicio se tienen los límites en donde el dispositivo no debe de salirse. De ser así se activan alarmas para informar a las personas de interés.

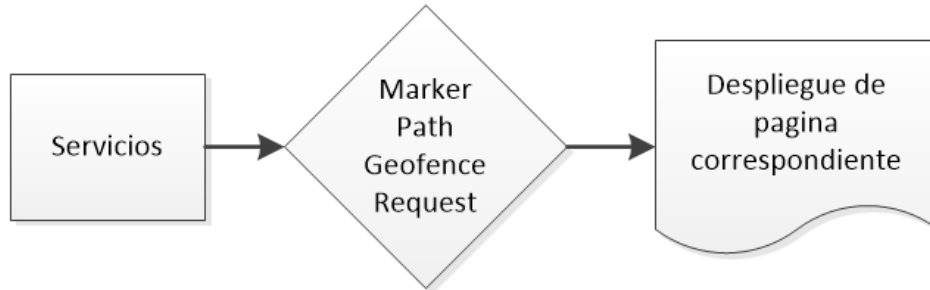
d. Servicio Request. Este es un comando para indicarle al módulo que envíe su información para conocer su ubicación. Dependiendo en qué modo este, este puede ser un parámetro fijo.

4. Envío de datos entre carpetas. El envío de datos entre carpetas se refiere al intercambio de parámetros entre la carpeta principal y las de los servicios. Para ejemplificar esto, en la carpeta principal se encuentra el archivo /index.html el cual despliega nuestra interfaz web. En ella se encuentran las pestañas de servicios. Si seleccionamos un servicio, tenemos la opción de ingresar parámetros para que sean procesados, dependiendo de servicio así serán sus opciones. Si ingresamos al servicio de MARKER, seleccionamos los parámetros y le presionamos enviar, la carpeta /index.html le enviara estos datos a /despliegue.php para que los procese y realice las tareas necesarias para desplegar en pantalla a base de los parámetros enviamos por /index.html.

La carpeta /index.html enviara los datos por medio de HTTP más específico con el comando POST. Por lo tanto /despliegue.php también debe de programarse para recibir los comandos de POST y no por medio de otro método (por ejemplo GET o PUT).

El envío entre carpetas tiene el siguiente diagrama de flujo:

Figura 66. Flujo de datos entre carpetas.



Al seleccionar una opción se es redirigido a la carpeta correspondiente como lo muestra la siguiente tabla:

Tabla No.5. Servicios disponibles y carpeta a la que se re dirigido.

Servicios disponibles y carpeta a la que se dirige		
Numero de opción	Servicio	Carpeta destino
1	Marker	/despliegue.php
2	Path	/despleguepath.php
3	Geofence	/desplieguefence.php
4	Request	/hacerequest.php

Cada carpeta tiene su función específica. Avocarse a la sección de anexos para poder ver el código fuente de cada una.

5. Interfaz con la base de datos. Para alojar los datos se utilizó la base de datos MySQL. En el servicio de alojamiento adquirido se incluyen herramientas para modificar, crear y analizar el contenido en cada tabla de la base de datos. Para este caso se eligió el gesto de base de datos llamado PHPmyAdmin el cual permite realizar tareas de manera simple sin necesidad de conocimiento vasto de lenguajes de consulta estructurados.

La base de datos se diseñó teniendo dos tablas principales, una dedicada a

información del usuario que estaría portando el dispositivo y la segunda para registrar la posición geográfica y datos adicionales a la hora de ingresarla a la base de datos (Hora, día y mes).

La tabla dedicada a la información del usuario se nombró Usuarios y tendrá la siguiente forma:

Tabla No.6. Parámetros para la tabla Usuarios.

Parámetros para la tabla de Usuarios				
ID	Nombre	Apellido	Request	Usuario

La variable REQUEST se incluye en esta tabla y es la responsable de indicar si se ha solicitado o no una petición.

La tabla dedicada a la información de las posiciones geográficas se nombró UBICA y tendrá la siguiente forma:

Tabla No.7. Parámetros para la tabla UBICA.

Parámetros para la tabla de UBICA						
ID	LAT	LON	Mes	Día	Anio	Hora

6. Correos de alerta. Debido a los diferentes modos de operación del módulo, se pueden tener casos en los que necesitamos que alguien de confianza sepa que necesitamos asistencia. O bien puede ser el caso que alguien se salió de su zona asignada y se debe notificar este evento.

En el caso de alarma (descrito en la sección de estructura y modos de funcionamiento), cuando se es presionado el botón de pánico, el dispositivo se encarga de enviar su ubicación e indicar que se está en peligro.

En el caso de Geofence periódicamente se revisa la ubicación y los datos son

enviados al servidor para verificar que este dentro de sus límites permisibles. Si llega a ocurrir un evento que necesite de la notificación a otra persona se enviara un correo electrónico con los siguientes datos:

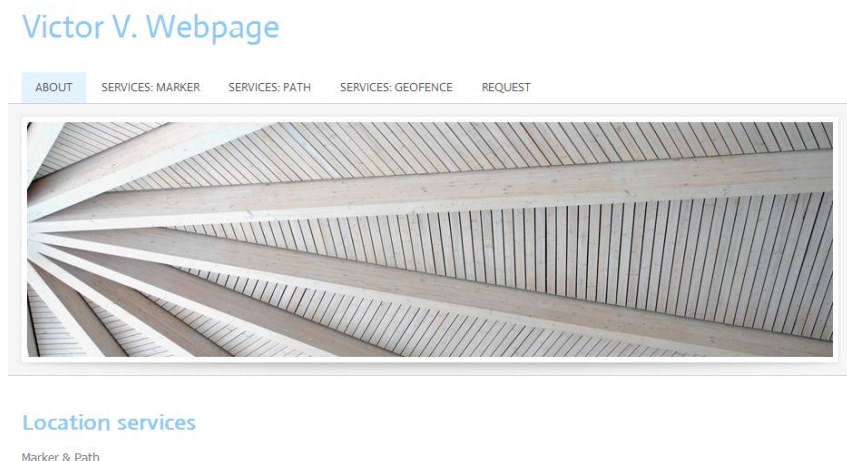
Tabla No.8. Cuerpo del correo enviado en modo alarma.

Cuerpo del correo electrónico	
Parámetro	Descripción
Modulo ID	Identificador del módulo por usuario
Hora y fecha	Variables para tener conocimiento cuando fue generada la alerta
Posiciones geográficas	Potencial ubicación de donde se encontraría el dispositivo y por ende la persona
Link de posición	Link que hace referencia a la posición geográfica en la plataforma de Google maps

B. Resultados

1. Alojamiento de la plataforma web. Como se mencionó anteriormente se adquirió un servicio de alojamiento con el cual se incluyó el dominio www.vicvicar.com haciendo referencia a las iniciales del autor de este trabajo. Si se ingresa a la dirección web mencionada se obtendrá la siguiente interfaz:

Figura 67. Página principal de la plataforma web



Si vemos en la sección superior encontramos 5 pestañas, en donde las últimas 5 son las de servicios.

2. Servicio Marker. Si se hace click a la pestaña llamada Marker se encontrara con las siguientes opciones:

Figura 68. Opciones para el despliegue de la opción Marker.

Seleccione el mes y día de interes

Agosto ▼ 29 ▼

Seleccione rango de hora

De

09:00 ▼

A

12:00 ▼

Submit

Si se encontraron ubicaciones respecto a los parámetros ingresados se re direccionara a una nueva carpeta la cual desplegara su información de la siguiente manera:

Figura 69. Interfaz para la opción Marker



Los indicadores rojos son las posiciones anteriormente encontradas mientras que el único marcador verde es la última posición registrada.

3. Servicio Path. Si se hace click a la pestaña llamada Path se encontrara con las siguientes opciones:

Figura 70. Opciones para el despliegue Path

Seleccione el mes y dia de interes

Agosto ▼ 29 ▼

Seleccione rango de hora

De

09:00 ▼

A

12:00 ▼

Submit

Si se encontraron ubicaciones respecto a los parámetros ingresados se re direccionara a una nueva carpeta la cual desplegara su información de la siguiente manera:

Figura 71. Interfaz para la opción Path.



Los indicadores verdes son las posiciones anteriormente encontradas mientras que el único marcador rojo es la última posición registrada. La línea azul traza el camino de donde tuvo recorrido el dispositivo.

4. Servicio GeoFence. Si se hace click a la pestaña llamada Path se encontrara con la siguiente opción:

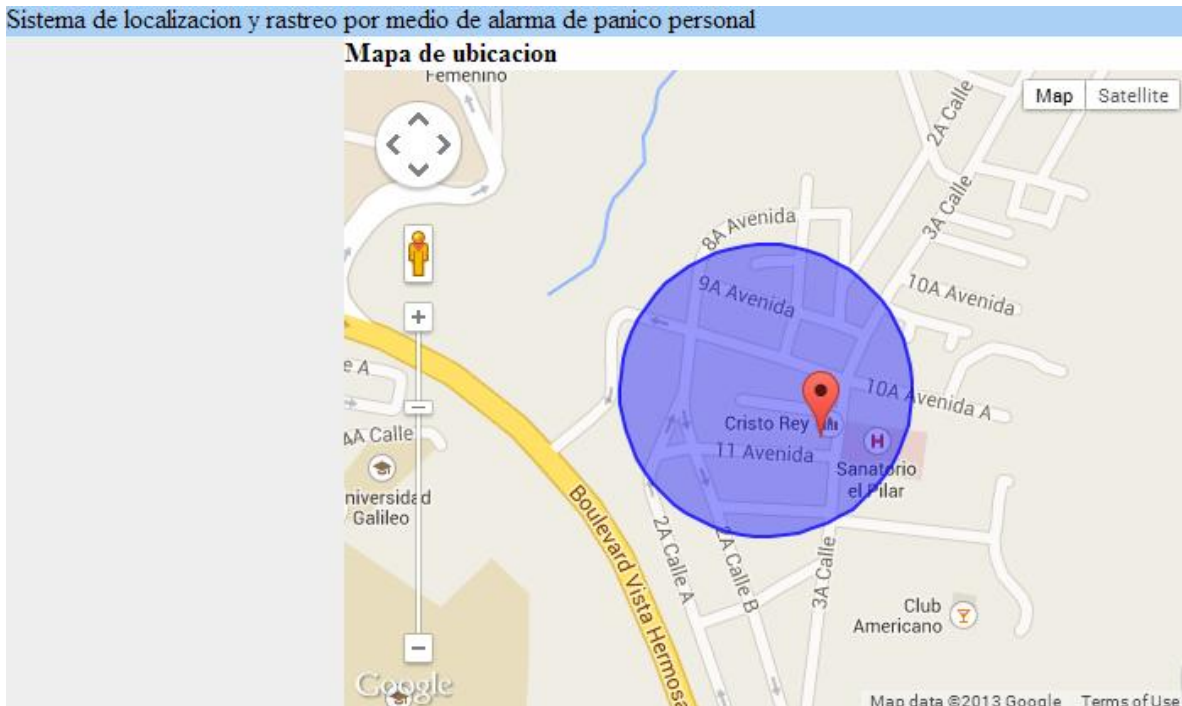
Figura 72. Opciones para el despliegue de la opción Geofence.

Seleccione el usuario de interes

LM741CN

Si se encontraron ubicaciones respecto al usuario ingresado se redireccionará a una nueva carpeta la cual desplegara su información de la siguiente manera:

Figura 73. Interfaz de la opción Geofence.



El círculo azul es el área permitida en donde el dispositivo puede permanecer. El marcador rojo es la posición más reciente que se obtuvo.

5. Servicio Request. Si se hace click a la pestaña llamada Path se encontrara con la siguiente opción:

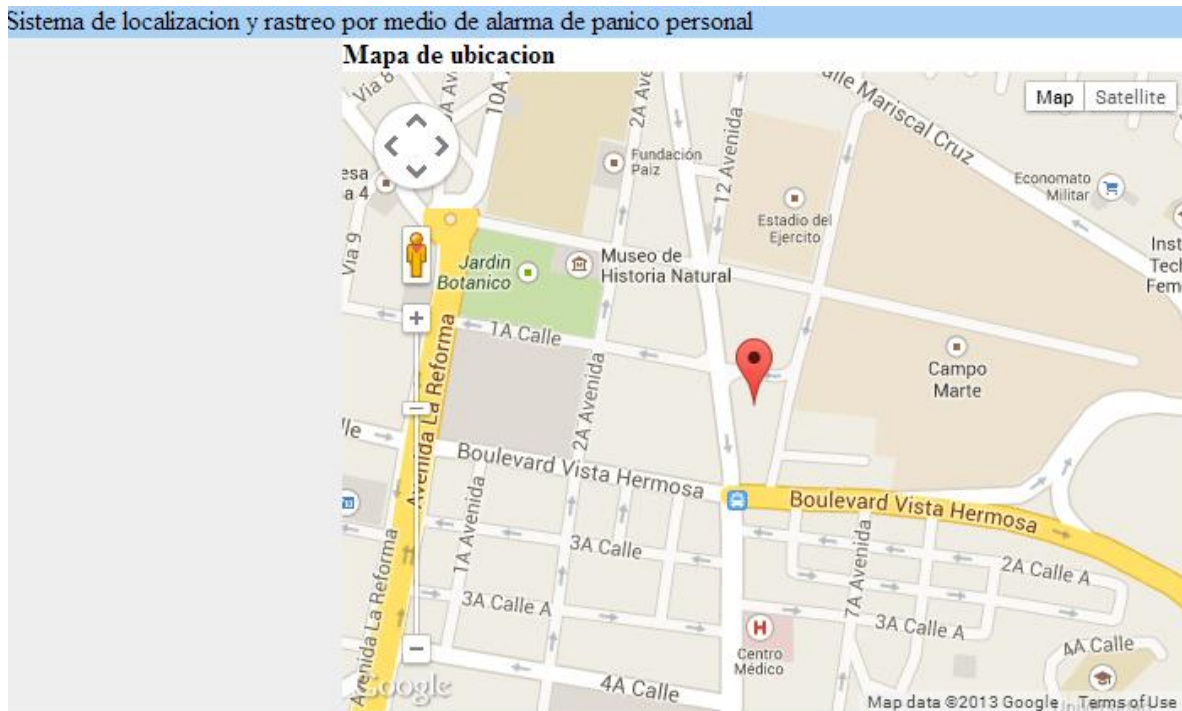
Figura 74. Opciones para el despliegue de Request.

Seleccione el usuario de interes

LM741CN

Si se encontraron ubicaciones respecto al usuario ingresado se redireccionará a una nueva carpeta la cual desplegará su información de la siguiente manera:

Figura 75. Despliegue para la interfaz de Request.



El marcador rojo es la última ubicación del módulo cuando se le fue solicitada su posición.

6. Correos de alerta.

Figura 76. Correo de alerta enviado en violación a Geofence.

Mensaje de alerta
 Identificación: LM741CN
 Hora y fecha: 25/9/2013, 18:10 horas
 Descripción: dispositivo fuera de área permitida
 Última ubicación: 14.557769, -90.513133
 Link de última ubicación: <https://maps.google.com/maps?q=%2B38%C2%B0+34'+24.00%22,+109%C2%B0+32'+57.00%22&um=1&ie=UTF-8&sa=N&tab=w>

C. Discusión

En todas las partes donde se intercambiaban datos se utilizaron los formularios de

HTML para enviar la información ya que es un método seguro para el envío ya que la información no viaja en el encabezado de la página sino viaja en el protocolo en sí.

Se utilizó la aplicación de mapas de Google ya que es la más completa y usada mundialmente. Además es una herramienta gratuita a la cual le podemos hacer peticiones sin límite.

Para la recepción de los correos electrónicos de alerta se tiene que estar conscientes que se enviara con el dominio que se adquirió. Por lo que puede ser que el mensaje sea de un remitente por ejemplo alerta@midominio.com lo cual a los servidores más grandes de correo, este no es un usuario válido y colocara el mensaje en la carpeta de no deseados. De ser así debemos habilitar el filtro para que los mensajes de alerta sean correctamente enviados.

XII. Sub módulos de servicio

Los sub módulos en este trabajo se describen como aquellos que permiten a los módulos principales comunicarse, alimentarse y conectarse entre ellos.

Su función no es tan relevante (como por ejemplo la de un microcontrolador) pero de igual forma son vitales para la operación y sin ellos no se lograría el objetivo de este trabajo. No se consideran como módulos principales ya que estos solo le brindaran servicios a los módulos para que ellos hagan las tareas principales.

A. Diseño

1. Encendido y apagado de los módulos GPS y GSM. Ya que sería impráctico tener todos los módulos principales alimentados, se necesita algún dispositivo electrónico que permita encender y apagar los módulos a discreción y que esta señal pueda ser la salida de un microcontrolador.

Además debe poder suministrar la corriente para que opere en sus límites más altos de cada módulo.

Para el encendido y apagado de los módulos se eligió al dispositivo electrónico Relé más específico de estado sólido para realizar la tarea. El seleccionado fue el de la marca IXYS con la numeración CPC1002NTR el cual puede ser activado con una señal de 3.3 voltios y un corriente en el orden de los micro amperios. La corriente máxima continua que puede suministrar es de 700mA/h lo cual sobre dimensiona la alimentación de cada módulo.

2. Seguidores de voltaje. Debido a que la salida del módulo GSM tiende a atenuarse se necesita de una configuración de seguidor para señales digitales. Este debe de poder cambiar de estado lo suficiente rápido para poder seguir a la señal a la velocidad seleccionada en el microcontrolador.

Para seguir la señal proveniente del módulo GSM se eligió la configuración de dos

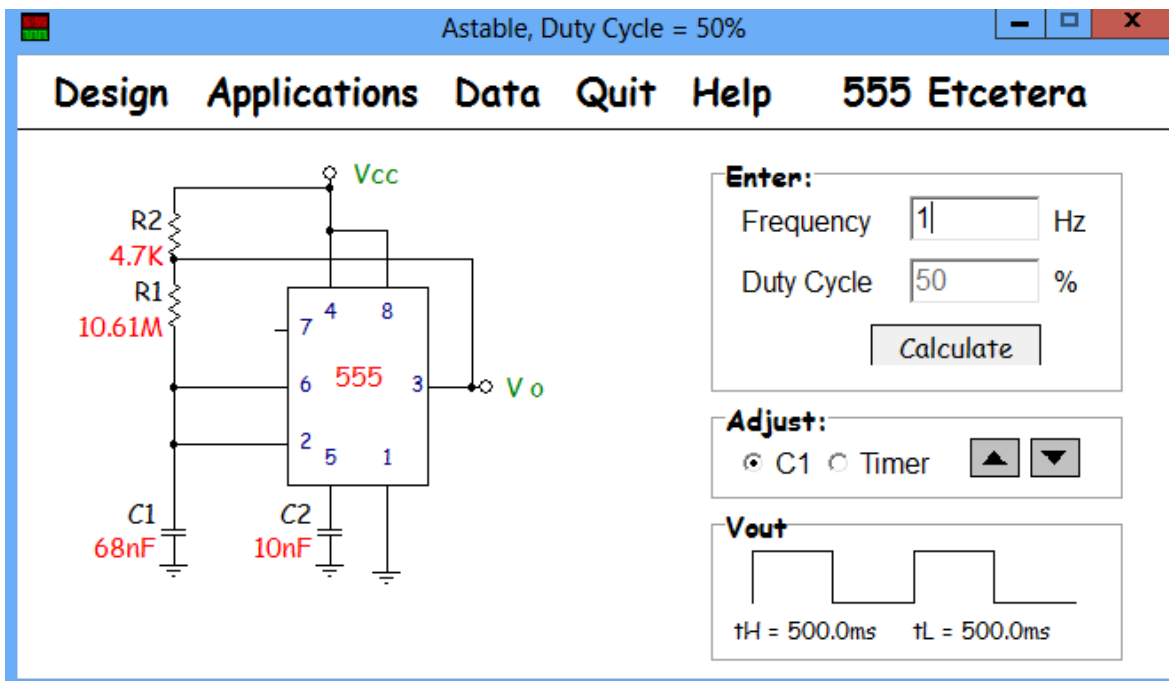
negadores con los transistores 2N3904.

3. Oscilador astable 555. Para alimentar a los contadores de timer1 y poder ocasionar su interrupción se necesita que un reloj este constantemente incrementando el valor de sus registros. Si bien existen circuitos que funcionan en modo oscilador como lo es una resistencia más capacitor, se decidió utilizar el integrado 555 que es el oscilador por excelencia. Ofrece además el voltaje y la corriente de salida necesaria para que el microcontrolador interprete sus salidas como un incremento válido.

Se utilizó la herramienta de software SS555 para diseñar y realizar los cálculos de los valores para que oscile con ciertas características. Se simuló para comprobar su operación con LiveWire.

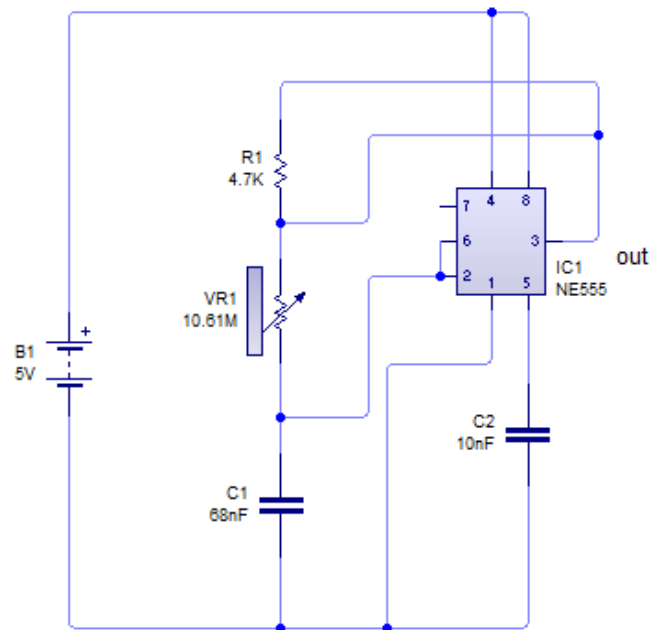
La frecuencia de oscilación seleccionada fue de 1Hz.

Figura 77. Diagrama esquemático para oscilador 555 y valores de componentes utilizando SS555.



Con esta frecuencia ya fija el valor a cambiar para establecer la frecuencia de la interrupción son sus contadores internos.

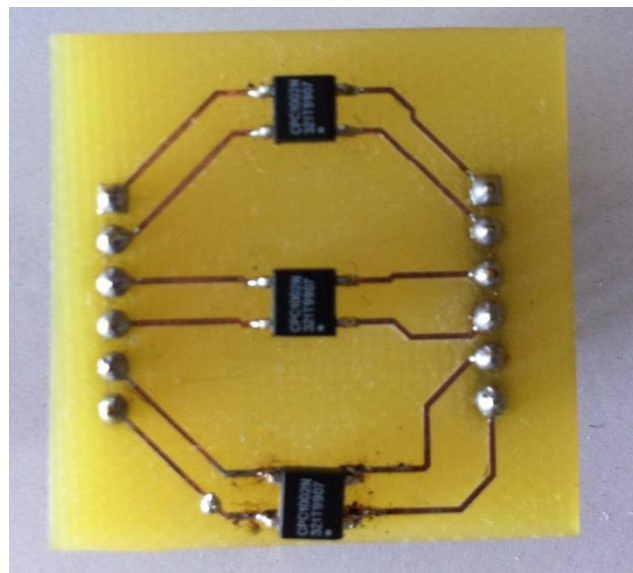
Figura 78. Circuito simulado en LiveWire.



B. Resultados

1. Relé de estado sólido

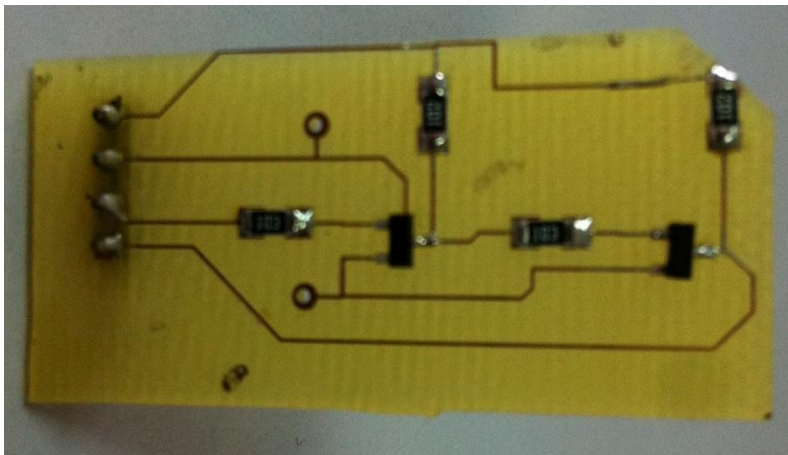
Figura 79. Relés de estado sólido en PCB de prueba.



Los relés se soldaron a placas PCB para poder experimentar con ellos demostrando un desempeño adecuado para la aplicación.

2. Negadores con transistores

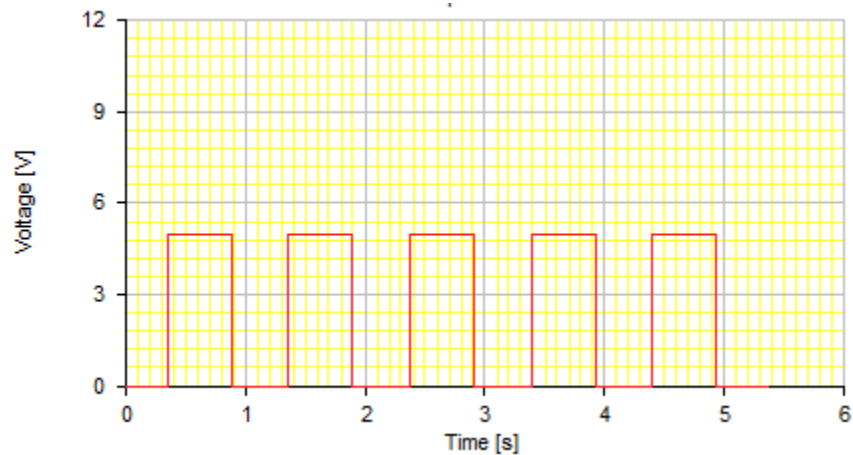
Figura 80. Configuración doble negador en PCB de prueba.



Los transistores y resistencias se soldaron a una placa PCB y se puso a prueba su funcionamiento conectándole una señal proveniente de un microcontrolador la cual la amplifico y siguió como se esperaba.

3. Oscilador astable 555

Figura 81. Simulación del oscilador con los valores establecidos por SS555.



C. Discusión

El módulo del GPS utiliza dos relés de estado sólido, uno para su alimentación y otro para la salida de su comunicación. Esto se debió a que realizando pruebas, al encender el GPS, este si detectaba una carga en su salida TX (en este caso la carga será el RX del microcontrolador) este no empezaba a operar correctamente. Por esta razón a la salida se le adiciono un relé de estado sólido el cual se activa cuando se está en la rutina del GPS, encendiendo de primero el GPS (sin carga en su TX) y luego el canal de comunicación.

En la simulación del oscilador podemos ver que no obtendremos una frecuencia de 1Hz. Esto es debido a que en papel y lápiz los cálculos concuerdan pero en la práctica se deben de considerar factores internos del integrado como por ejemplo el voltaje de operación para alcanzar ciertas frecuencias. La numeración del integrado influye también para saber si se está diseñando con un circuito especial que alcance tanto frecuencias altas como bajas.

XIII. Módulo de potencia

A. Corriente consumida por el dispositivo

La siguiente tabla describe la corriente consumida de cada módulo y sub módulos al ser alimentados:

Tabla No.9. Corriente consumida por los módulos principales y sub módulos.

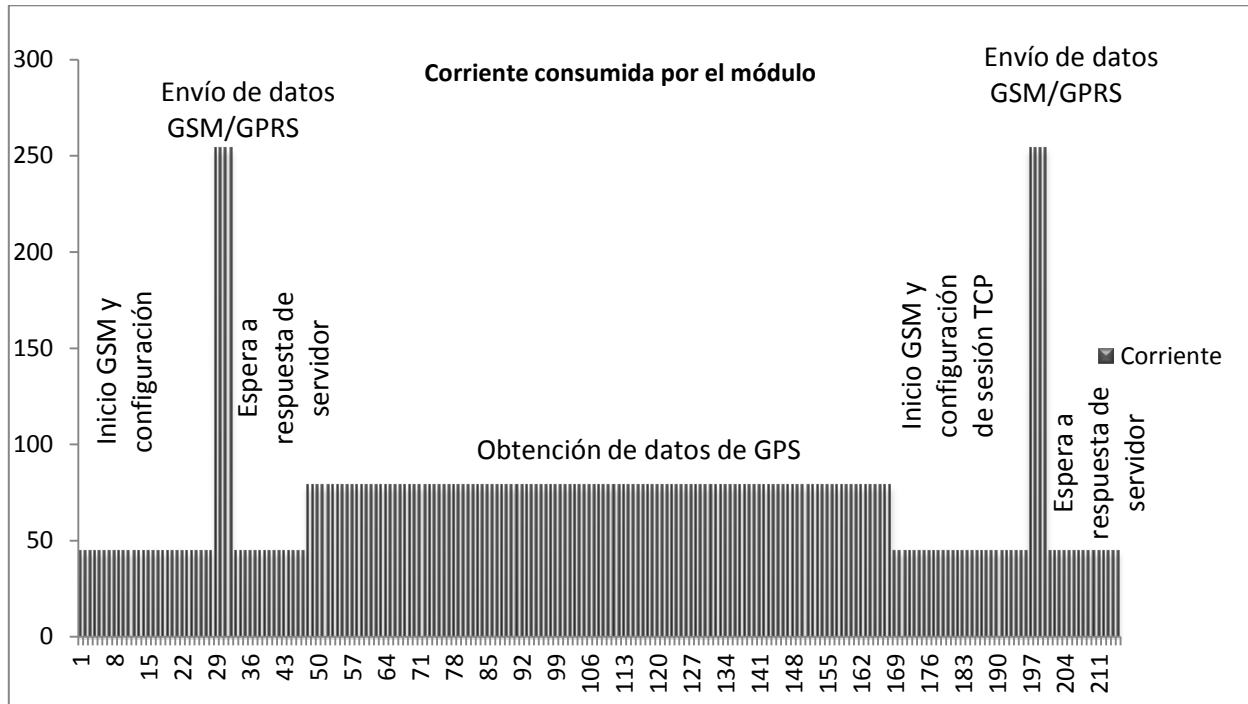
Corriente consumida por módulos y sub módulos	
Módulo	Corriente promedio(mA/h)
GSM datos GPRS	230
GSM modo espera	21
GPS	70
Microcontrolador (Activo)	13
Microcontrolador(Durmiendo)	0.0156
Regulador de voltaje	0.12
Relé de estado solido	2
Oscilador TS555	.09
Configuración seguidores de voltaje	3

Para saber cuánta corriente promedio consumirá el dispositivo se consideró el caso en el que más corriente consume que es cuando se encuentra en modo de petición.

En modo de petición el dispositivo es despertado por alguna de sus interrupciones, continuamente alimenta su módulo GSM para verificar si ha tenido alguna petición. Suponiendo que si fue activada la petición el dispositivo deja de alimentar el módulo GSM y enciende el GPS con un tiempo máximo de dos minutos para obtener las coordenadas geográficas. Al obtener las coordenadas el dispositivo desalimenta el GPS y vuelve a encender el módulo GSM para enviar los datos que se obtuvieron.

Considerando este caso el siguiente consumo de corriente ocurre dependiendo de las siguientes cuatro fases:

Figura 82. Consumo de corriente en modo de petición.



Se considera en la figura anterior que ciertos sistemas siempre estarán alimentados. Estos se representan en la siguiente tabla con su consumo respectivo:

Tabla No.10. Módulos y sub módulos alimentados en todo momento.

Módulos y sub módulos alimentados en todo momento				
Modulo/sub modulo	Regulador	3 *Relé	TS555	Seguidor
Corriente (mA/h)	0.12	6	1	3.5

La suma de estos módulos y sub módulos que siempre estarán alimentados más la corriente del microcontrolador al estar activo nos da una corriente de 23.62 mA/h.

Siendo el caso de petición en el que mayor consumo de corriente se tiene por parte del dispositivo, un aproximado para su consumo promedio se calcula mediante una suma

ponderada entre cada fase y su tiempo de funcionamiento.

La ecuación a utilizar es la siguiente:

$$\text{Promedio de consumo} = (I) * \frac{T}{T_r}$$

Donde I es la corriente consumida por cada fase, T es el tiempo de duración de la fase y T_r es el tiempo de referencia (para dar el promedio en amperios/hora, amperios/minuto o en la que se desee). Los dos parámetros de tiempo se ingresan en segundos y la corriente en miliamperios para obtener los siguientes datos de cada fase:

$$\text{Inicio GSM y configuracion} = 44.62 * \left(\frac{27}{3600}\right) = 0.33465 \text{ mA/h}$$

$$\text{Envio de datos GSM/GPRS} = 253.62 * \left(\frac{4}{3600}\right) = 0.2818 \text{ mA/h}$$

$$\text{Espera respuesta del servidor} = 44.62 * \left(\frac{15}{3600}\right) = 0.1859 \text{ mA/h}$$

$$\text{Obtencion de datos GPS} = 78.62 * \left(\frac{120}{3600}\right) = 2.6207 \text{ mA/h}$$

$$\text{Microcontrolador en SLEEP} = 0.0156 * \left(\frac{3386}{3600}\right) = 0.01467 \text{ mA/h}$$

Ya que tenemos fases repetidas el cálculo para la corriente promedio en una hora será:

$$I_{prom} = 2 * (\text{Inicio GSM y config.}) + 2 * (\text{Envio de datos}) + 2 * (\text{Respuesta Servidor}) \\ + \text{Obtencion de datos GSM} + \text{Microcontrolador en SLEEP}$$

$$I_{prom} = 2 * (0.33465) + 2 * (0.2818) + 2 * (0.1859) + 2.6207 + 0.01467$$

$$I_{prom} = 4.2401 \text{ mA/h}$$

Si consideramos el ciclo de vida de la batería de 80% nuestra corriente promedio será:

$$I_{prom} = \frac{4.2401}{0.8} = 5.3 \text{ mA/h}$$

Por lo que, si el dispositivo operara una vez por hora, nuestro tiempo de operación será el resultado de la siguiente ecuación:

$$\text{Tiempo de funcionamiento} = \frac{\text{Capacidad de la batería}}{\text{Corriente demanda a la batería}}$$

Con una batería con capacidad de 500mA/h nuestro tiempo de funcionamiento será de:

$$\text{Tiempo de funcionamiento} = \frac{500}{5.3} = 94.37 \text{ Horas}$$

Ya que una petición por hora es una frecuencia muy baja, se considera el caso donde se realizan 4 peticiones por hora por lo que nuestro tiempo se reduce a la siguiente expresión:

$$\text{Tiempo de funcionamiento} = \frac{94.37}{4} = 23.59 \text{ Horas}$$

Este tiempo de funcionamiento de cuatro peticiones por hora entra dentro de un rango considerable ya que podemos suponer que la persona u objeto que porte el dispositivo tendrá personas de interés que se interesen al desconocer su ubicación en un lapso menor a 24 horas.

B. Diseño

1. Elección de la batería suministradora. Para el suministro de todos los módulos se necesita una batería que pueda cumplir con la demanda de por lo menos dos módulos alimentados.

La corriente con la que nos basaremos en la elección de la batería, será cuando tengamos el consumo máximo y este ocurre cuando dos módulos están encendidos.

Por defecto siempre estará alimentado el microcontrolador (su mayor parte en modo SLEEP), y nunca tendremos más de dos módulos encendidos debido al diseño que se eligió. Por lo tanto los dos módulos encendidos en cualquier instante que más corriente consumen es el microcontrolador y el módulo GSM enviando datos.

En cuanto al voltaje de la batería, debido a que el módulo GSM tiene un voltaje mínimo de operación de 3.3 voltios, se debe elegir una batería que tenga mayor voltaje para que el regulador pueda operar (debido a su dropout voltaje). Por lo tanto el requisito del voltaje es que sea mayor a 3.3 voltios.

Se eligió la batería de Litio polímero de la serie GMB del distribuidor PowerStream la cual tiene un voltaje de operación de 3.7 Voltios y una corriente de suministro máxima de 550mA/h.

2. Elección de regulador de voltaje. Se debe considerar que el voltaje de la batería será muy similar al voltaje de operación. Esto nos indica que el dropout voltaje del regulador tiene que ser muy pequeño.

Se debe agregar además la restricción que por lo menos debe suministrar una corriente constante de 253.62 mA/h para que el pico de corriente sea abastecido.

El regulador de voltaje seleccionado fue el MCP1826S-3302E/DB del fabricante Microchip el cual tiene un dropout voltaje de 250mV siendo suficiente para la batería que es de 3.7 voltios.

3. Cargador de batería. Según la teoría de cargadores se necesita que este suministre un voltaje de 0.5 a 1 voltio arriba del voltaje nominal de la batería y una corriente menor a un sexto de la corriente máxima suministrada.

Con estos datos se podría optar por un cargador de celular debido a la facilidad de obtención.

4. Tiempo de funcionamiento. El dispositivo debe de por lo menos funcionar 24 horas lo cual es un tiempo de operación aceptable considerando los posibles casos que se pudieran dar.

C. Resultados

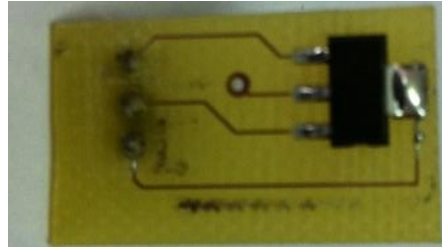
1. Batería y regulador seleccionados. La batería seleccionada sobredimensiona el voltaje mínimo de operación por 0.4 voltios y es suficiente para algunos reguladores de voltaje para realizar su trabajo. Se experimentó con ella en conjunto el regulador de voltaje y una carga 'dummy' para que suministrara 400mA/H. La batería se manifestó de manera correcta y aun con la alta demanda no se sobrecalentó.

Figura 83. Batería seleccionada para alimentar al dispositivo.



La capacidad de 550mAh de la batería sobre dimensiona los picos de corriente y cumple la demanda mínima. El regulador seleccionado suministra constantemente una corriente de 1000mA lo cual sobre dimensiona los picos que se tendrán en la operación.

Figura 84. Regulador seleccionado en PCB de prueba.



El regulador se soldó a una placa de prueba y se comprobó su funcionamiento a una carga de 550mA/h.

2. Cargador de batería. El cargador seleccionado es del fabricante Sony para la cámara DSC-W150 y su modo de operación es de 4.2 voltios y 200mA/h de carga constante. Este cumple con nuestros cálculos teniendo una carga pequeña la cual será alimentada a la batería.

Además el voltaje esta 0.5 voltios arriba del necesitado lo cual está dentro de nuestros estándares de carga.

3. Tiempo de funcionamiento. Según los datos obtenidos en la sección de corriente consumida por el dispositivo y la batería seleccionada se obtendrá un tiempo de funcionamiento de:

$$\text{Tiempo de funcionamiento} = \frac{550(\text{capacidad batería})}{5.3(\text{corriente demandada})} * \frac{1}{4} (\text{peticiones en una hora})$$

$$\text{Tiempo de funcionamiento} = 25.94 \text{ Horas}$$

D. Discusión

Se eligió una batería plana debido a que entre los objetivos de este trabajo especifican que el dispositivo debe ser portable y con esta batería logramos reducir el tamaño total.

Idealmente la batería a elegir hubiese sido aún más pequeña, pero al ser así el voltaje se reduce drásticamente y nuestro regulador seleccionado no hubiera tenido el mínimo Voltage dropout para su operación.

XIV. Elaboración del PCB

En esta sección se describe como fue la elaboración y diseño del PCB para el dispositivo final.

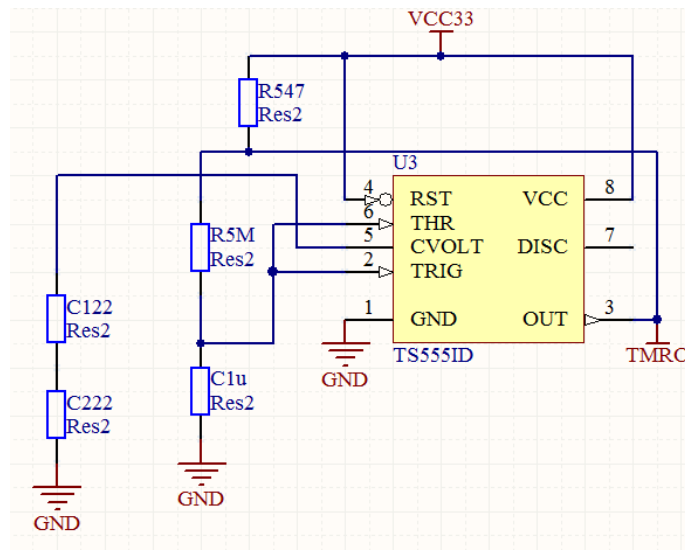
A. Diseño

1. Circuito esquemático. Para elaborar el circuito de esquema se utilizó el programa Altium Designer el cual incluye la mayoría de librerías de los componentes a utilizar. Fue necesario realizar el diseño en PCB de los siguientes componentes:

- Relé de estado sólido.
- Botón para alarma de pánico.

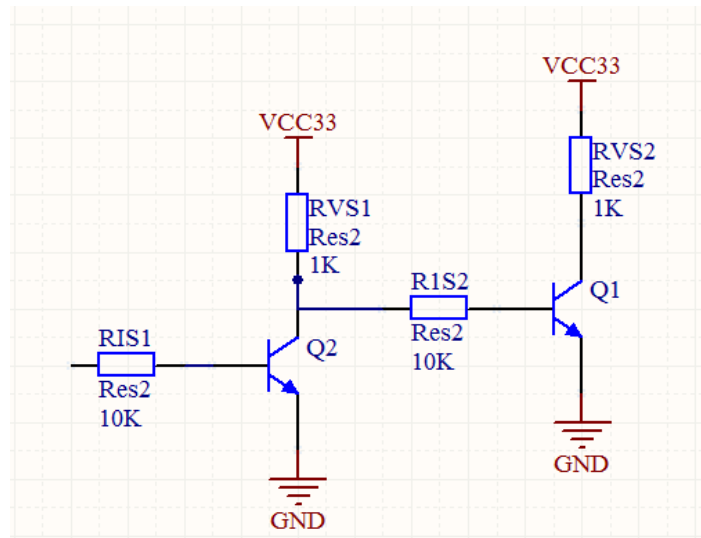
a. Esquemático para el TS555

Figura 85. Esquemático para el oscilador astable TS555.



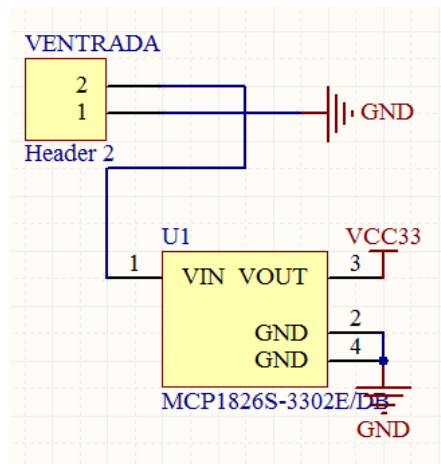
b. Esquemático para el seguidor de señal

Figura 86. Esquemático para configuración doble negador con transistores y resistencias.



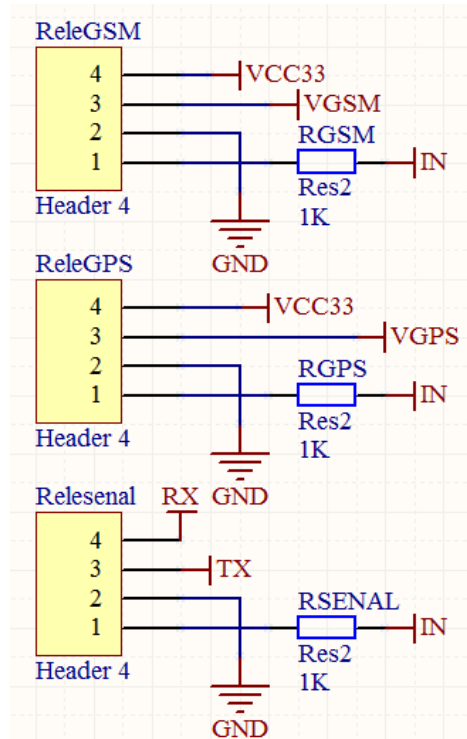
c. Esquemático para regulador de voltaje

Figura 87. Esquemático para el regulador de voltaje seleccionado.



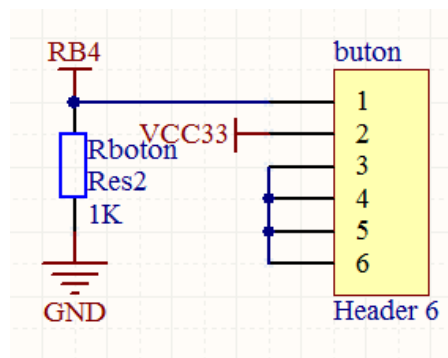
d. *Esquemático para relés de estado solido*

Figura 88. Esquemático para la configuración de los relés de estado sólido.



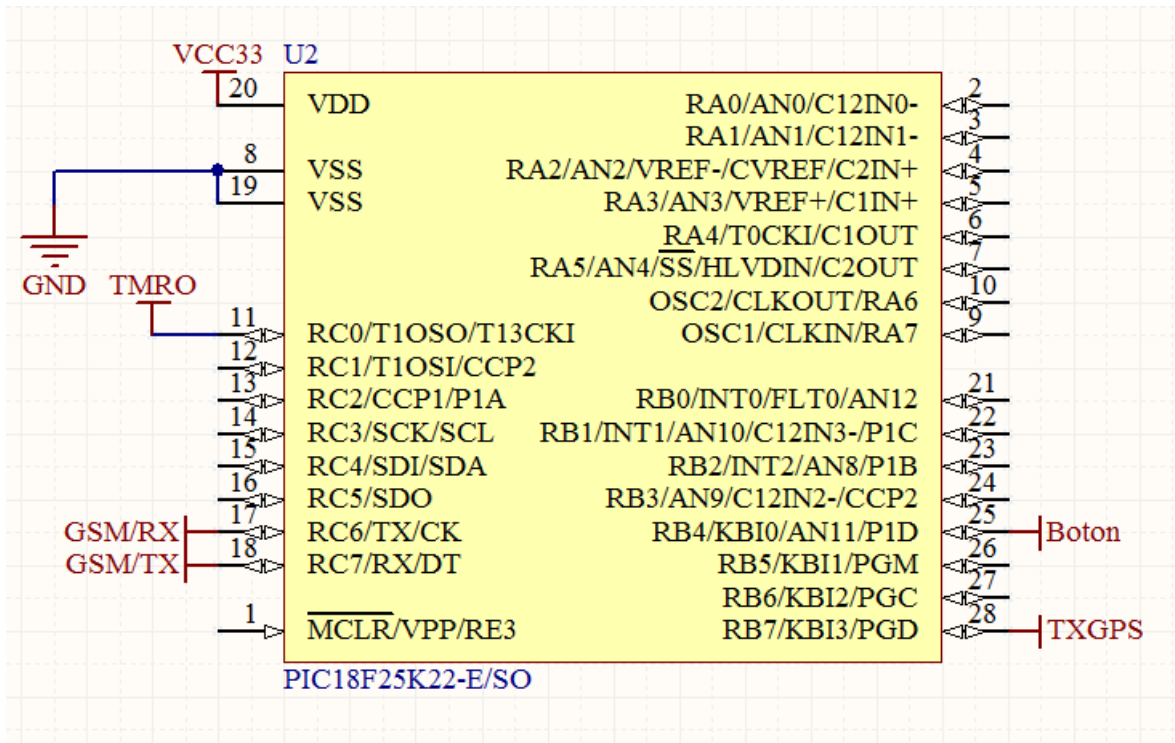
e. *Esquemático para el botón*

Figura 89. Esquemático para el botón que acciona la interrupción PORTB.



f. Esquemático para microcontrolador

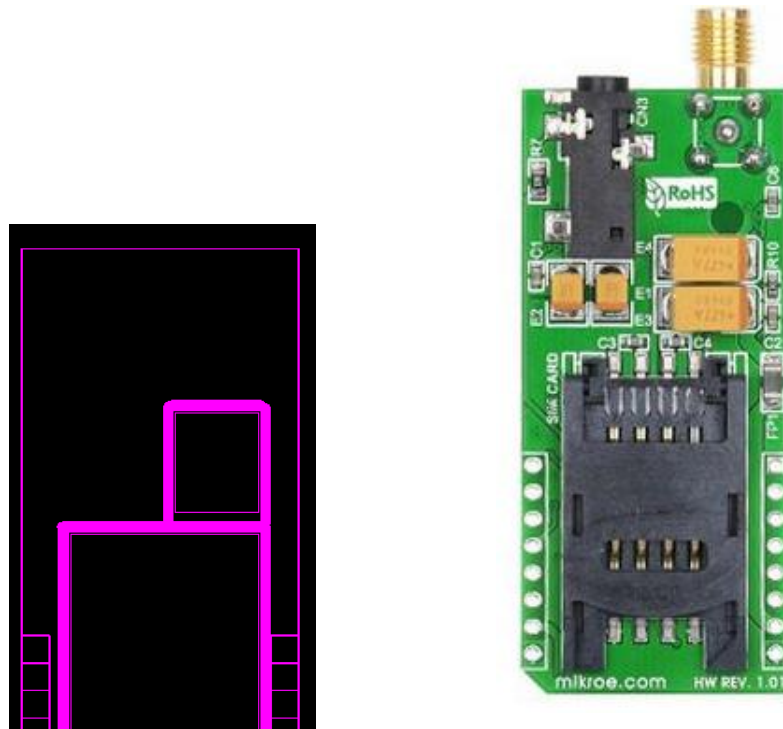
Figura 90. Esquemático para el microcontrolador 18f25k22.



2. Forma del PCB. El dispositivo final será la unión de la plataforma GSM click y el PCB con los demás módulos y sub módulos. Para poder unirlos se diseñó el PCB final con la forma para que hubiera espacio a los componentes electrónicos del módulo GSM. Se removieron además componentes que no iban a ser necesarios como lo es toda la sección de audio incluyendo capacitores y Jack para audífonos.

El diseño de la placa se creó en AutoCAD 2013 y se importó a Altium para basarnos en esa figura.

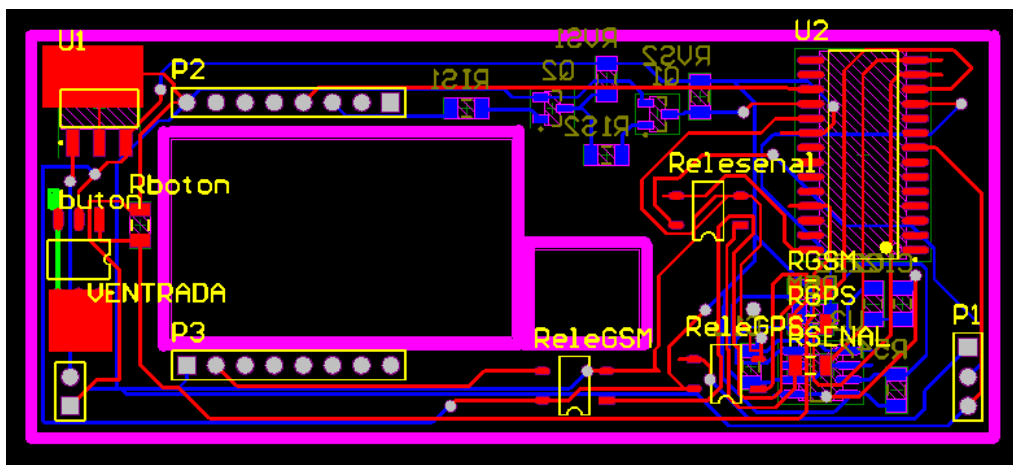
Figura 91. Forma del PCB basado en módulo GSM Click.



Como vemos se dejó espacio para insertar la tarjeta SIM y se considera el espacio donde está el Jack de audio como área útil para colocar componentes. La antena BNC fue reemplaza por una antena de superficie.

3. Creación del PCB en Altium. Para la creación del PCB se utilizó la herramienta de PCB Wizard de Altium y con esta se importaron los componentes del proyecto.

Figura 92. Diagrama del PCB final con todos sus componentes de placa.

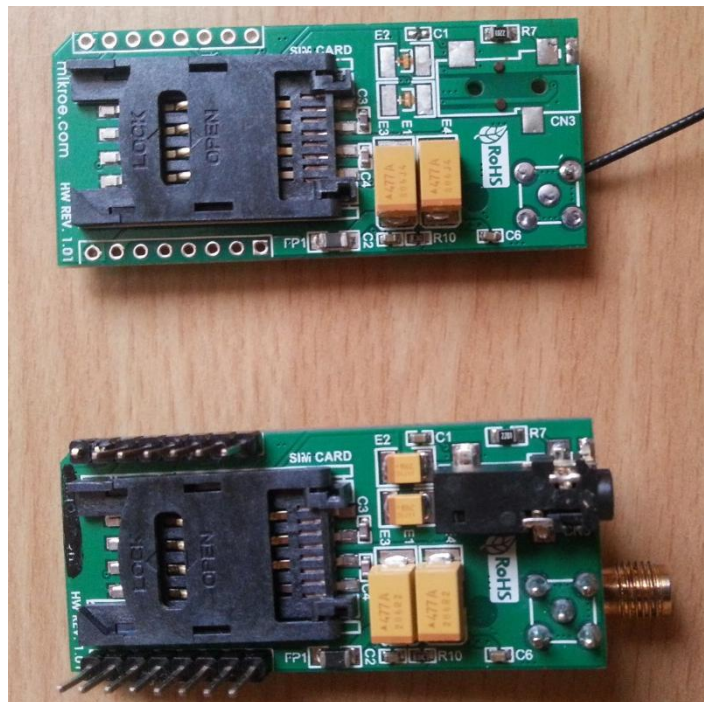


4. Fabricación del PCB. Para la fabricación del PCB se utilizó la máquina del fabricante LPKF la cual realizo las pistas y corto la placa como se tenía diseñada para que el módulo GSM entrara sin problemas. Adicionalmente removió todo el cobre que estaba de más para que solo estuviesen los caminos a utilizar y fuese más fácil su soldadura.

B. Resultados

1. Componentes removidos de la plataforma GSM click. Como se mencionó anteriormente el módulo de audio y antena BNC fueron removidos.

Figura 93. Módulos GSM, parte posterior con y sin modificaciones.



La antena BNC fue removida y se soldó una de superficie para hacer al módulo de menores dimensiones.

Figura 94. Módulos GSM, parte frontal con y sin modificaciones.

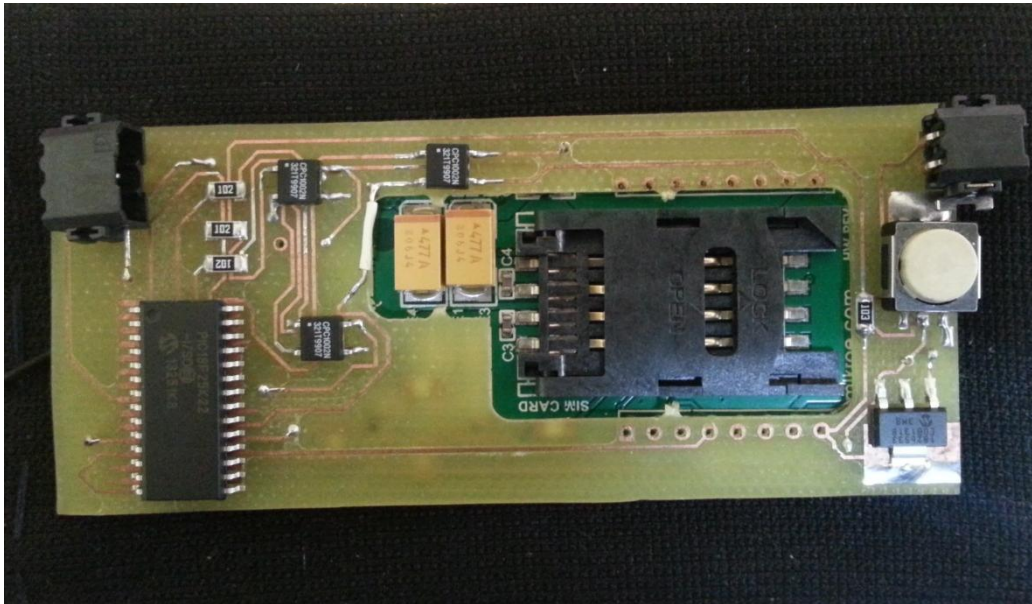


2. PCB con componentes

Figura 95. Parte superior del dispositivo.



Figura 96. Parte inferior del dispositivo.



En la figura anterior se encuentra en la esquina superior izquierda y derecha los conectores para el modulo GPS y batería respectivamente.

C. Discusión

El archivo de AutoCAD que fue importado a Altium debe de respetar ciertas reglas para que su formato sea a escala 1:1. Primero en AutoCAD debemos de especificar con las dimensiones que se está trabajando en el PCB. Dependiendo de las dimensiones, en el PCB de Altium se nos pedirá a que escala es que queremos de importar. Para esto es importante seleccionar la escala más grande y no sub escalas. Por ejemplo si se está trabajando en pulgadas en Altium, debemos de elegir pulgadas de salida en AutoCAD y a la hora de importar elegir que sea traducido también a pulgadas y no mils como que es la opción por defecto.

Para la soldadura de los componentes se debe de tener en cuenta que hay vías en medio de ciertos componentes como lo es en el microcontrolador y en un relé. Para esto debe de realizarse una vía que no estropee el componente y quede con cierta pendiente causando que si por error es presionado, la soldadura se venza y por ende se desprenda el componente.

Para la selección de componentes en Altium, debido a que su librería es muy extensa, debemos de observar cuidadosamente cada número de identificación. Por ejemplo, en Altium no es lo mismo importar el componente LM778 y el componente LM778s. Como se menciona, sus librerías son muy extensas y pequeñas numeraciones como estas hacen que el 'footprint' del componente varíe. Este fue un error en la elaboración de placas de prueba para los sub módulos y ciertos componentes no fueron posibles soldarlos.

XV. Adaptación del módulo a prenda de vestir

A. Diseño

Como objetivo de este trabajo de graduación, el dispositivo debe ser portable y de poca notoriedad, por eso se modificó un abrigo para alojar al dispositivo y demás componentes.

Se utilizó un abrigo al cual se le fabricaron bolsas para alojar al dispositivo y su alimentación. A los módulos que no pertenecen a la misma placa como lo son el GPS y la batería, se le adaptaron conectores para que fuesen compatibles y se adaptaran al PCB fácilmente.

B. Resultados

1. Conectores tipo header en PCB y módulos. Se utilizaron conectores de dos posiciones para la batería y de tres para el GPS.

Figura 97. Conectores header de tres posiciones macho y hembra.



Figura 98. Módulo GPS con conector header de tres posiciones.

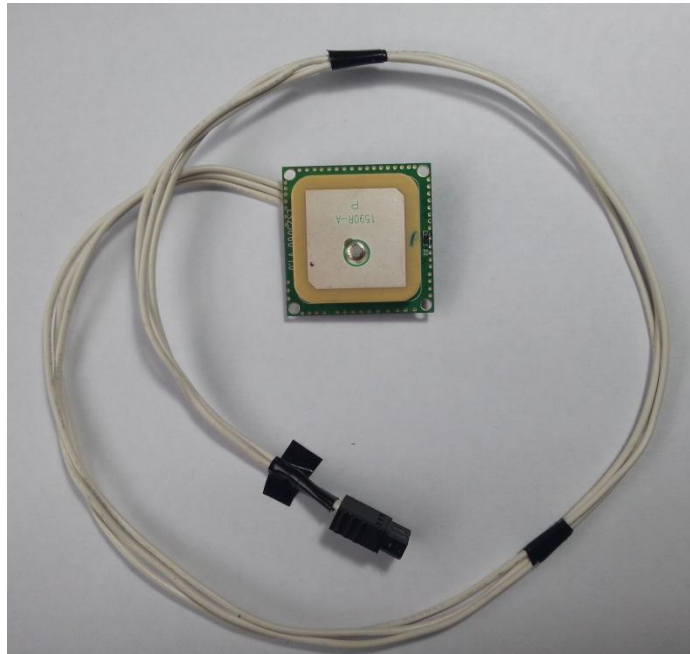


Figura 99. Batería con conector header de dos posiciones.



Figura 100. Conector tipo header de dos posiciones, conexión entre PCB y batería.

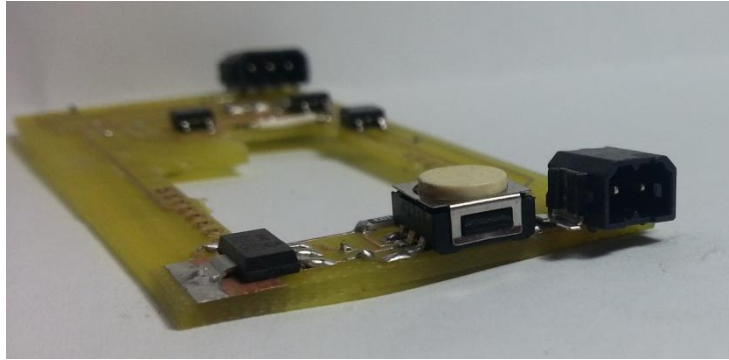


Figura 101. Abrigo utilizado para instalar el dispositivo.



Figura 102. Bolsa alojadora de PCB y batería.



Figura 103. Bolsa alojadora de módulo GPS.



Figura 104. PCB y batería instalados en abrigo.



Figura 105. PCB y batería instalados en abrigo.



Figura 106. Módulo GPS instalado en el abrigo.



Figura 107. Dispositivo oculto completamente instalado en abrigo.



Figura 108. Dispositivo visible instalado en abrigo.



XVI. Prueba de campo en modo rastreo más alarma

A. Diseño

Para poner a prueba el dispositivo y realizar una prueba de campo, se tomó la configuración en donde el dispositivo está alojado en un abrigo. Se programó el microcontrolador para que el dispositivo funcionara en modo rastreo más alarma y que mandara información aproximadamente cada 7 minutos.

Se hizo un recorrido en carro a velocidad promedio de 45 kilómetros por hora y fueron abarcadas las zonas 9, 10 y 15.

El abrigo se llevaba intencionalmente en la parte del copiloto para que el GPS pudiera enganchar a los satélites.

Como punto de parada fue Gourmet Center de zona 10 en el cual intencionalmente se acciono el botón para que ingresara la ubicación actual del dispositivo. Esta última acción con el abrigo puesto.

B. Resultados

El módulo en total envió una cantidad de tres ubicaciones en un periodo de aproximadamente 7 minutos más una ubicación enviada minutos después que fue activada manualmente con el botón de pánico.

Figura 109. Posiciones encontradas en la prueba de campo.

Mapa de ubicacion



Figura 110. Vista satelital última ubicación encontrada.

Mapa de ubicacion



C. Discusión

Para la prueba de campo, ya que se estaba transportando en un vehículo las posiciones encontradas varían en más de 300 metros ya que se están en movimiento. El factor principal en este caso fue que la sesión de datos toma tiempo para subir al servidor las posiciones por lo que si tomamos una posición en un instante, esta se subiera en un tiempo máximo de 4 minutos y si viajamos a altas velocidades tendremos un margen de error bastante grande.

Para la posición final ya que se estaba estático, la posición registrada fue una muy buena aproximación y debido a que no teníamos el factor movimiento, nuestra ubicación no tuvo retardo. El error fue menor a 10 metros y se estuvo afuera del vehículo por 5 minutos a esperar que la posición fuese ingresada.

Entre la última posición y la penúltima no hay tanta diferencia debido a que al llegar al lugar destino fue accionado el botón para indicar que se registrara nuestra posición y ya que acaba de ocurrir la función de rastreo estas dos se encuentran muy cercanas,

XVII. Conclusiones y recomendaciones

A. Conclusiones

- El tiempo máximo para que el dispositivo realice todas sus funciones en modo petición es de cuatro minutos.
- El promedio de la corriente consumida en el modo petición es de 5.3mAh.
- El tiempo de operación del dispositivo con una batería de mA/h y cuatro peticiones por hora es de 23.59 horas.
- La comunicación con la red celular y el envío de datos con el microcontrolador fue exitosa intercambiando información con un servidor remoto.
- La obtención de coordenadas geográficas con el GPS fue exitosa teniendo un margen de posición de +/- 15 metros como máximo con doce satélites enganchados.
- La plataforma web fue exitosamente operativa con cuatro funciones para el despliegue de datos.
- La comunicación de la plataforma web con la base de datos fue posible y el flujo de datos entre ellas fue transparente.
- El microcontrolador fue programado exitosamente con un contador, cuatro interrupciones, tres rutinas esenciales y dos puertos seriales activos.
- El PCB del dispositivo fue exitosamente fabricado y todos sus módulos y sub módulos fueron adaptados sin complicaciones.
- La configuración de doble negador amplificó correctamente la señal proveniente del módulo GPS.
- Los relés de estado sólido en conjunto con la señal del microcontrolador alimentaron y desalimentaron los módulos y sub módulos exitosamente.

B. Recomendación

- Para el envío de datos es preferible que sean enviados por el protocolo TCP y no UDP para asegurar que las dos partes afirmen recepción y transmisión de información.
- Al programar en MikroC, ya que tiene un límite de código, se debe de tratar de

optimizar el código para que sea posible su compilación. Para lograr esto, una opción es importar solo librerías esenciales y encontrar la forma de tener menos condiciones y evaluaciones.

- Al trabajar con módulos GPS, se debe de enfocar en la rapidez que estos se enganchan con los satélites. Esto lo define el parámetro de canales que al tener más nuestro tiempo de obtención de coordenadas se reducirá drásticamente.
- Para la comunicación y desarrollo con plataformas de red celular es importante especificar el orden de los comandos AT. De otra forma no se llevara a cabo debido a conflictos de Handshake.
- Para la elaboración de placas se debe de tener en cuenta que líneas en donde se transmita radio frecuencia deben de estar en asolación correctamente o bien con una capa en contra que haga una referencia neutra. De lo contrario nuestra ganancia de la antena amplificara también ruido de los alrededores causando distorsiones en la información enviada.
- Para obtener aun mayor cobertura de la posición del dispositivo es altamente recomendable incluir en el diseño algún tipo de triangulación mediante la red celular para los casos en donde la señal del GPS no pudiese llegar con suficiente potencia. De esta forma se garantiza en aun en los ambientes más extremos siempre tendremos una referencia de posición.
- Respecto a la plataforma web, debido a que se trata de información importante y de carácter privado, es muy recomendable implementar algún tipo de seguridad y/o manejo de sesiones para usuarios. Con esto garantizando la visualización correspondiente a cada uno sin poder observar a los demás dispositivos a los que no se tenga privilegios para obtener sus posiciones.

XVIII. Bibliografía

- [1] Carlospes. (2013, Septiembre) Lenguaje de alto nivel. [Online].
www.carlospes.com/minidiccionario/lenguaje_de_alto_nivel.php
- [2] Emfexplained. (2013, Septiembre) Explicacion sobre redes moviles. [Online].
<http://www.emfexplained.info/spa/?ID=25196>
- [3] Emfexplained. (2013, Septiembre) Explicacion sobre redes moviles. [Online].
<http://www.emfexplained.info/spa/?ID=25195>
- [4] Enrique Risco. (2013, Sentiembre) Que es JavaScript. [Online].
<http://www.pablin.com.ar/computer/cursos/cursos/js1.htm>
- [5] Federal space agency. (2013, Septiembre) GLONASS system. [Online]. <http://glonass-iac.ru/en/>
- [6] Galaxi0. (2013, Septiembre) El puerto serial. [Online]. <http://galaxi0.wordpress.com/el-puerto-serial/>
- [7] Galeon. (2013, Septiembre) Microcontroladores. [Online]. <http://axnm.galeon.com/>
- [8] Geonovo. (2013, Septiembre) Geofence - Assisted Living Safety Technology. [Online].
<http://www.geonovo.com/geofence-elderly-wanderers.html>
- [9] Global positoning systems. (2013, Septiembre) El Sistema de Posicionamiento Global. [Online]. www.GPS.gov
- [10] Ingeniatic. (2013, Septiembre) GSM (Sistema Global para las telecomunicaciones móviles). [Online]. <http://ingeniatic.euitt.upm.es/index.php/tecnologias/item/471-gsm-sistema-global-para-las-telecomunicaciones-m%C3%B3viles>
- [11] Instituto de tecnologias educativas. (2013, Septiembre) Iniciación al lenguaje estructurado de consultas (SQL). [Online].
<http://www.ite.educacion.es/formacion/materiales/93/cd/m6/index.html>
- [12] Instituto nacional de estadistica y geografia. (2013, Septiembre) Sistemas de posicionamiento global. [Online]. <http://www.inegi.org.mx/geo/contenidos/geodesia/gps.aspx>
- [13] Jlozano. (2013, Septiembre) Interfaces fisicas. [Online].
www.dtic.upf.edu/~jlozano/interfaces/interfaces6.html
- [14] Jorge L. (2013, Septiembre) Multivibradores. [Online]. www.ladelec.com/teoria/electronica-digital/197-multivibradores
- [15] Jose Alvarez. (2013, Septiembre) Protocolo HTTP. [Online].
http://www.uhu.es/josel_alvarez/NvasTecnProg/recursos/ProtocoloHTTP.pdf
- [16] Kioskea. (2013, Septiembre) Estandar GSM sistema global de comunicaciones moviles.

- [Online]. <http://es.kioskea.net/contents/681-estandar-gsm-sistema-global-de-comunicaciones-moviles>
- [17] Kioskea. (2013, Septiembre) Estandar GSM, sistema global de comunicaciones moviles. [Online]. <http://es.kioskea.net/contents/681-estandar-gsm-sistema-global-de-comunicaciones-moviles>
- [18] LearningAboutElectronics. (2013, Septiembre) What is a LM7805 Voltage Regulator? [Online]. <http://www.learningaboutelectronics.com/Articles/What-is-a-LM7805-voltage-regulator>
- [19] Luis Gongazales. (2013, Septiembre) Multivibradores. [Online]. <http://jair.lab.fi.uva.es/~josgavi/descargas/Multivibradores>
- [20] Maestros de la Web. (2013, Septiembre) Maestros de la Web. [Online]. <http://www.maestrosdelweb.com/editorial/%C2%BFque-son-las-bases-de-datos/>
- [21] Maestros de la web. (2013, Septiembre) Que es PHP. [Online]. <http://www.maestrosdelweb.com/editorial/phpintro/>
- [22] MarinJl. (2013, Septiembre) 555. [Online]. <http://www.uv.es/marinjl/electro/555.htm>
- [23] Masadelante. (2013, Septiembre) HTML. [Online]. <http://www.masadelante.com/faqs/html>
- [24] Microchip. (2013, Septiembre) PIC18f45k22. [Online]. <http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en546239>
- [25] Mikroelectronica. (2013, Septiembre) EasyPic7. [Online]. <http://www.mikroe.com/easypic/>
- [26] Mikroelectronica. (2013, Septiembre) El mundo de los microcontroladores. [Online]. <http://www.mikroe.com/chapters/view/79/capitulo-1-el-mundo-de-los-microcontroladores/>
- [27] Mikroelectronica. (2013, Septiembre) GSM click. [Online]. www.mikroe.com/click/gsm
- [28] PrivateLine. (2013, Septiembre) GSM network. [Online]. <http://www.privateline.com/PCS/GSM03.html>
- [29] Samira Grayeb. (2013, Septiembre) Los microcontroladores. [Online]. <http://losmicrocontroladores.blogspot.com/>
- [30] Servidor-alicante. (2013, Septiembre) Celula. [Online]. <http://glosarios.servidor-alicante.com/telecomunicaciones/celula>
- [31] Softhandover. (2013, Septiembre) Transition from 2g to 4g. [Online]. <http://softhandover.wordpress.com/2010/09/29/transition-from-2g-to-4g/>
- [32] Telefonica. (2013, Septiembre) Machine to machine. [Online]. <http://www.tid.es/es/Tecnologia/Paginas/M2M.aspx>
- [33] Telit. (2013, Septiembre) GSM/GPRS modules-GL865. [Online]. www.telit.com/en/products.php?p_ac=show&p=110
- [34] Universidad de Oviedo. (2013, Septiembre) Ensamblador. [Online].

<http://www.unioviedo.es/ate/alberto/TEMA3-Ensamblador.pdf>

- [35] Wikipedia. (2013, Septiembre) Estacion base. [Online].
http://es.wikipedia.org/wiki/Estaci%C3%B3n_base
- [36] Wikipedia. (2013, Septiembre) Formula del Haversine. [Online].
http://es.wikipedia.org/wiki/F%C3%B3rmula_del_Haversine
- [37] Wikipedia. (2013, Septiembre) Pagina Web. [Online].
http://es.wikipedia.org/wiki/P%C3%A1gina_web
- [38] Wikipedia. (2013, Septiembre) Sistema de navegacion Galileo. [Online].
http://es.wikipedia.org/wiki/Sistema_de_navegaci%C3%B3n_Galileo
- [39] Wikipedia. (2013, Septiembre) Sistema global para las comunicaciones moviles. [Online].
http://es.wikipedia.org/wiki/Sistema_global_para_las_comunicaciones_m%C3%B3viles
- [40] Wikipedia. (2013, Septiembre) Transmission Control Protocol. [Online].
http://es.wikipedia.org/wiki/Transmission_Control_Protocol
- [41] Wikipedia. (2013, Septiembre) Trilateracion. [Online].
<http://es.wikipedia.org/wiki/Trilateraci%C3%B3n>
- [42] Xbot. (2013, Septiembre) 555. [Online]. <http://r-luis.xbot.es/icdatos/555.html>

XIX. Anexos

A. Código microcontrolador y plataforma web

El código fuente del microcontrolador y de la plataforma web incluyendo se encuentran en el siguiente repositorio:

<https://github.com/vicvicvar/Tesis>

Contiene todo el código comentando y despliegue de palabras clave resaltadas para mejorar su entendimiento.

