



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

“DISEÑO E IMPLEMENTACIÓN DE UN PROTOTIPO CON
SERVICIOS DE GEOPOSICIONAMIENTO Y ALARMAS DE
SOCORRO PARA LOS BUSES DE LA ESPOL Y DESARROLLO
DE PARADEROS INTELIGENTES USANDO UNA APP
COMERCIAL MEDIANTE LA RED GSM/GPRS”

INFORME DE MATERIA INTEGRADORA

Previa a la obtención del Título de:

INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

KARINA PAOLA MITE PITA

FRANCISCO JOSÉ VELÁSQUEZ PALMA

GUAYAQUIL – ECUADOR

AÑO: 2016

AGRADECIMIENTOS

Agradezco al Señor, creador del cielo y la tierra, por sostenerme y ayudarme a terminar esta etapa de mi vida; en los momentos en que más lo necesitaba. A mi madre que, desde el cielo, sentirá que mi éxito es su éxito; mujer perseverante que me enseñó desde muy niña, que las grandes metas se logran con grandes sacrificios. A mis familiares y amigos que fueron ese apoyo para continuar mi camino, cuando desfallecía.

Karina Paola Mite Pita

Primero agradecer a Dios por mantenerme siempre en el camino correcto, a mi familia y amigos que siempre han estado a mi lado apoyándome. Un especial agradecimiento al startup Enjambre por su colaboración en este proyecto, a Microsigma Service por la confianza y la oportunidad de realizar esta tesis; y al final y no menos importante al Ph.D. Francisco Novillo y al Grupo GICOM por haberme guiado y aconsejado en estos últimos semestres a desenvolverme en el área profesional y humana.

Francisco José Velásquez Palma

DEDICATORIA

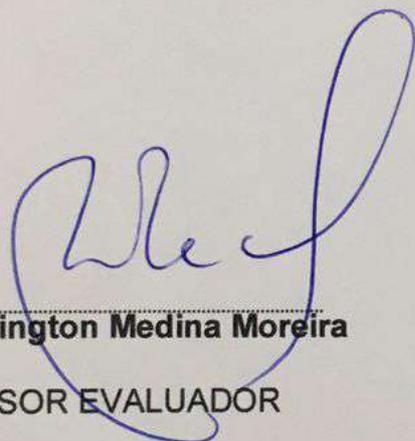
Dedico este proyecto a mi madre la señora Vitalia Pita Vargas, quien a pesar de que ahora ya no está junto a nosotros sigue y seguirá siendo el pilar fundamental de mi vida, el ejemplo a seguir de fortaleza y perseverancia, de amor incondicional y fe, quien con sus consejos y directrices fue quien encamino mi vida por el camino del bien, la unidad, el amor a Dios y a la familia.

Karina Paola Mite Pita

Dedico este proyecto a mi Mama Cuqui Palma (+), quien desde el cielo me guiado cada uno de mis pasos como mi ángel guardián. A mi hermana Susana quien es y seguirá siendo mi motivo de perseverancia y éxito. A mi Mamiñita quien ha desempeñado el papel de segunda madre dándome consejos de vida y quien tanto ha anhelado este momento. A mi papa Jorge Velásquez y a mi Ñaña Patty quienes son mi ejemplo a seguir como profesionales y como personas. A mi hermano mayor Jorge mi mayor ejemplo de lucha constante a pesar de las adversidades de la vida. Los amo y los seguiré amando inmensamente.

Francisco José Velásquez Palma

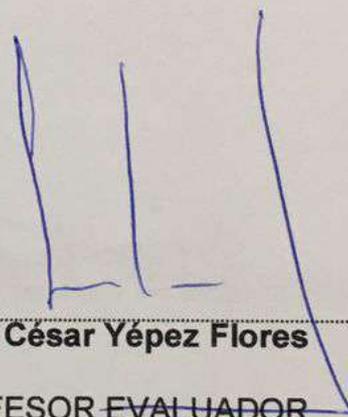
TRIBUNAL DE EVALUACIÓN



A handwritten signature in blue ink, consisting of a large, stylized initial 'W' followed by a cursive 'Medina Moreira'.

M.Sc. Washington Medina Moreira

PROFESOR EVALUADOR



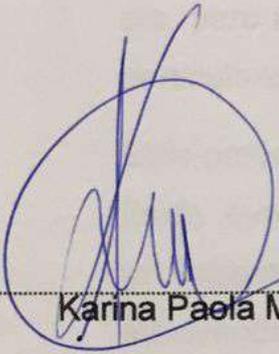
A handwritten signature in blue ink, consisting of a stylized 'C' followed by 'Yépez Flores'.

M.sc. César Yépez Flores

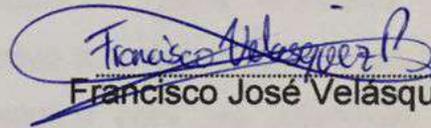
PROFESOR EVALUADOR

DECLARACIÓN EXPRESA

"La responsabilidad y la autoría del contenido de este Trabajo de Titulación, nos corresponde exclusivamente; y damos nuestro consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"



Karina Paola Mite Pita



Francisco José Velásquez Palma

RESUMEN

La avalancha de tecnologías y propuestas que están emergiendo en el sector vehicular tiene un auge cada vez mayor. Con un sinfín de proveedores la competencia dentro de esta área es difícil, siendo estos tipos de servicios costosos, por tal motivo no son tan atractivos para los clientes ya que ofrecen servicios básicos como, por ejemplo: planes de rastreo por medio de mensajes o por medio de aplicativos web, etc.

Este tipo de propuestas resuelven de alguna manera los problemas de hoy en día, pero no de una manera integral, dando a entrever fallos en las soluciones particulares por ser costoso y no tan comercial.

Este proyecto tiene como finalidad presentar una alternativa más económica y fiable para mejorar el servicio de transporte en Espol, mediante el uso de tecnologías de hardware libre y su respectiva instalación dentro de los buses, incluyendo la novedad de botones de pánico para los diferentes tipos de emergencias. Además de esto un paradero inteligente en ubicaciones estratégicas dentro del campus ESPOL prosperina. Con esto se espera mejorar de alguna manera los tiempos de espera y las colas interminables de alumnos en las paradas de la institución. Integrando todo esto en una sola página web adaptativa se podrá hacer visible la ubicación de los buses en tiempo real, facilitando así la vida de los estudiantes.

Gracias al análisis realizado y a los resultados obtenidos se podrá ver que es una solución viable y eficaz para resolver los problemas antes mencionados, sin tener costos elevados de mantenimiento, ni de instalación.

ÍNDICE GENERAL

AGRADECIMIENTOS.....	ii
DEDICATORIA	iii
TRIBUNAL DE EVALUACIÓN	iv
DECLARACIÓN EXPRESA	v
RESUMEN	vi
ÍNDICE GENERAL.....	vii
ÍNDICE DE FIGURAS.....	xii
ÍNDICE DE TABLAS	xiii
CAPÍTULO 1	1
1. ASPECTOS GENERALES.....	1
1.1 Problema a Resolver.....	1
1.2 Justificación.....	4
1.3 Objetivo General y Específicos.....	7
1.3.1 Objetivo General.....	7
1.3.2 Objetivos Específicos.....	7
1.4 Metodología.....	8
1.5 Alcance.....	9
1.6 Estado del Arte.....	10
CAPÍTULO 2.....	12
2. MARCO TEÓRICO.....	12
2.1 Orígenes de la tecnología GSM en Ecuador.....	12
2.2 Sistema celular de una red GSM/GPRS.....	13
2.2.1 Estación móvil (MS).....	15

2.2.2	Subsistema de estación base (BSS).....	16
2.2.3	Subsistema de conmutación y red (NSS).....	17
2.2.4	Subsistema de operación y apoyo (OSS).....	18
2.3	Requisitos, versiones y evolución de la tecnología GSM.	19
2.3.1	GSM-900.....	21
2.3.2	DCS-1800.....	21
2.3.3	PCS-1900.....	22
2.3.4	Serving GPRS support node (SGSN).....	23
2.3.5	Gateway GPRS support node (GGSN).....	23
2.4	Sistemas de Geo Posicionamiento.....	24
2.4.1	Posicionamiento estático.....	27
2.4.2	Posicionamiento dinámico.....	27
2.4.3	Proceso de triangulación y posicionamiento.....	27
2.4.4	Geocercos.....	28
2.5	Sistemas embebidos.....	30
2.5.1	Los microprocesadores.....	31
2.5.2	Los microcontroladores.....	31
2.5.3	La lógica de programación.....	32
2.5.4	Arduino UNO.....	32
2.5.5	Arduino Mega 2560.....	34
2.6	Tipos comunicación y Módulos.....	35
2.6.1	Comunicación serial.....	35
2.6.2	Comunicación I2C.....	36
2.6.3	Comunicación SPI.....	37

2.6.4	Comandos AT.....	38
2.6.5	Módulo de comunicación SIM900.....	40
2.6.6	Módulo de comunicación SIM800.....	41
2.7	Sistemas Discretos.....	42
2.7.1	Capacitores.	42
2.7.2	Tiempo de carga y descarga de los capacitores.	43
2.7.3	Sistemas anti rebotes para pulsos digitales.....	45
2.7.4	Reguladores Lineales.....	46
2.7.5	Reguladores Conmutados.....	47
2.8	Sistemas RFID.	48
2.8.1	Tarjetas y lectores RFID.....	49
CAPÍTULO 3.....		51
3.	DISEÑO E IMPLEMENTACIÓN.	51
3.1	Diseño.	51
3.2	Diagramas de flujo.....	55
3.2.1	Diagrama módulo buses internos.	55
3.2.2	Diagrama módulo buses externos.....	56
3.2.3	Diagrama del paradero inteligente.....	58
3.3	Diseño de códigos.	59
3.3.1	Diseño de código para buses internos.....	59
3.3.2	Diseño de código para buses externos.....	60
3.3.3	Diseño de código para paradero inteligente.	61
3.4	Implementación de los Sistemas	62
3.4.1	Implementación para buses internos y externos.	62

3.4.2 Implementación para paradero Inteligente.	66
3.5 Diseño de las placas.....	70
CAPÍTULO 4.....	73
4. ANÁLISIS Y RESULTADOS.....	73
4.1 Software BeeTracker.....	73
4.2 Prueba de Módulo bus interno.....	74
4.3 Prueba de Módulo bus externo.....	79
4.4 Prueba de Módulo Paradero Inteligente.	82
4.5 Costos de Producción.....	84
CONCLUSIONES Y RECOMENDACIONES	88
BIBLIOGRAFÍA.....	90
ANEXOS.....	94

ÍNDICE DE FIGURAS

Figura 1.1. Grupo de estudiantes en el nuevo terminal de ESPOL en horas pico dentro del campus Gustavo Galindo	3
Figura 2.1. Modelos de tarjeta SIM utilizado para la tecnología GSM [17].	13
Figura 2.2. Red de telefonía celular [19].	14
Figura 2.3. Arquitectura de una red celular GSM [21].	15
Figura 2.4. Elementos que conforman el NSS [21].	18
Figura 2.5. Canalización de la banda GSM-900 [22].	21
Figura 2.6. Canalización de la banda DCS-1800 [22].	21
Figura 2.7. Canalización de la banda PCS-1900 [22].	22
Figura 2.8. Arquitectura GPRS [23].	24
Figura 2.9. Arquitectura global de la red GSM/GPRS [24].	25
Figura 2.10. Operaciones de los satélites [24].	26
Figura 2.11. Ejemplos de diseño de Geocercos [27].	29
Figura 2.12. Sistema embebido.	31
Figura 2.13. Placa Arduino UNO [30].	33
Figura 2.14. Placa Arduino mega 2560 [30].	35
Figura 2.15. Comunicación maestro esclavo de un enlace I2C [31].	37
Figura 2.16. Comunicación maestro esclavo de un enlace I2C [31].	38
Figura 2.17. Placa SIM900 [34].	41
Figura 2.18. Placa SIM800 [35].	42
Figura 2.19. Diferentes tipos de Capacitores.	43
Figura 2.20. Circuito capacitor, resistencia, fuente de voltaje [36].	44
Figura 2.21. Diagrama de un Sistema Anti-rebote [37].	46
Figura 2.22. Esquema de un regulador de voltaje 78XX lineal [38].	47
Figura 2.23. Esquema de un regulador como fuente conmutada [40].	48
Figura 2.24. Rangos de trabajo de la tarjeta RFID [41].	49
Figura 2.25. Etiqueta RFID [42].	50
Figura 3.1. Diagrama de posibles elementos a utilizar en el prototipo.	51
Figura 3.2. Placa GPRS NEO 6M de la empresa UBLOX [43].	53
Figura 3.3. Diagrama de flujo bus interno	56

Figura 3.4. Diagrama de flujo bus externo	57
Figura 3.5. Diagrama de parada inteligente	58
Figura 3.6. Conexión de los pines del Arduino Mega con el módulo SIM900.	63
Figura 3.7 Conexión del Arduino Mega con los módulos SIM900 y GPS.	64
Figura 3.8. Conexión a una entrada de interrupción.....	65
Figura 3. 9. Conexión Arduino UNO y lector RFID.	66
Figura 3.10. Conexión Arduino UNO y SIM900 usando pines por software.	67
Figura 3.11. Conexión de pantalla LCD mediante el módulo I2C.	68
Figura 3.12. Conexión entre Arduino UNO, pantalla LCD y SIM900.	70
Figura 3.13. Módulo del paradero inteligente.	71
Figura 3.14. Diseño de placa para los buses internos y externos.	71
Figura 3.15. Diseño de placa para el Paradero Inteligente.....	72
Figura 4.1. Página de ingreso a BeeTracker.....	73
Figura 4.2. Página donde se muestra el recorrido de los buses de ESPOL.	74
Figura 4.3. Dispositivo asociado a la cuenta BeeTracker.....	75
Figura 4.4. Conexión física de los buses internos y externos.....	75
Figura 4.5. Prototipo del módulo de los buses internos y externos.....	76
Figura 4.6. Prototipo del módulo en tiempo real.....	77
Figura 4.7. Ruta trazada por el bus interno.	77
Figura 4.8. Vista del recorrido del bus con sus respectivas alarmas.	78
Figura 4.9. Alarma con imagen en caso de robo.....	79
Figura 4.10. En caso de alarma de emergencia médica.	80
Figura 4.11. En caso de alarma de incendio.	81
Figura 4.12. Geocercos para enviar mensajes de auxilio al UPC1.	81
Figura 4.13. Geocercos para enviar mensaje de auxilio al UPC2.	82
Figura 4.14. Ruta trazada por el bus externo en BeeTracker.....	83
Figura 4.15. Modulo del paradero inteligente	84
Figura 4.16. Prototipo del paradero inteligente.....	85

ÍNDICE DE TABLAS

Tabla 1. GPS Disponibles en el mercado con algunas características.....	52
Tabla 2. Comparaciones entre los modelos de Arduino.	54
Tabla 3. Módulos de comunicación con la red celular.	54
Tabla 4. Frecuencias de tarjetas RFID.....	55
Tabla 5. Pines de conexión del Arduino Mega con el módulo SIM900.	63
Tabla 6. Pines de conexión entre el Arduino Mega y el modulo GPS.....	64
Tabla 7. Pines de conexión entre el Arduino UNO y el lector RFID.....	67
Tabla 8. Pines Arduino UNO y módulo SIM900.	68
Tabla 9. Pines del Arduino uno y el módulo I2C.....	69
Tabla 10. Costo de elementos de los módulos para los buses internos y externos.	85
Tabla 11. Costo por unidad.....	86
Tabla 12. Costo de producción de los módulos para el paradero inteligente.....	86
Tabla 13. Costo por unidad para los paraderos inteligentes.....	87
Tabla 14. Costos de mantenimiento mensual de BeeTracker	87
Tabla 15. Costo de instalación y mantenimiento	87
Tabla 16. Costo por unidad para la instalación en los buses.....	87

CAPÍTULO 1

1. ASPECTOS GENERALES.

1.1 Problema a Resolver.

En esta última década el desarrollo económico de los países subdesarrollados ha ido en crecimiento, lo que ha generado un progreso en el estatus familiar y en su poder adquisitivo, debido a esto se tiene ahora más facilidad de comprar un bien, por lo que el adquirir un vehículo es más fácil en estos tiempos, lo que ha ocasionado un gran problema para las ciudades pequeñas que no cuentan con vías amplias o vías alternas para la gran cantidad de vehículos transitando. La mayoría de la población en el mundo se encuentran laborando o realizando actividades fuera de su hogar o residencia, desde el más pequeño que tiene que acudir a una guardería mientras sus padres se encuentran dirigiéndose a sus respectivos puestos de trabajo, hasta los más jóvenes que tienen en cambio la responsabilidad de asistir a un establecimiento de enseñanza, sea esta una escuela, colegio o una universidad, las actividades y responsabilidades están inherentes en el diario vivir.

Si a estas actividades le sumamos un número considerable de personas que habitan una población en un mismo horario y en una misma ciudad, nos podemos dar cuenta de algo, pues con un poco de suspicacia y sentido común se puede saber que esto generará en su momento la famosa hora pico, es decir, el momento en el cual todas las personas empiezan a salir llenando una avenida principal, un bus de transporte o una carretera. Las actividades de la mayor parte de los ciudadanos económicamente activos en las distintas ciudades del Ecuador, ha crecido considerablemente. Cada ciudadano realiza labores fuera de su hogar, desde ir de compras al supermercado, llegar puntual a su puesto de trabajo; e incluso poder acudir a una clase importante en alguna de las instituciones [1].

En algunas ciudades limitar la circulación de vehículos particulares, ha sido una iniciativa para aliviar el tráfico vehicular, mediante los números de sus placas, sin mayores resultados [2]. En la ciudad de Quito las causas de los embotellamientos vehiculares son más complejas debido a las condiciones geográficas que cortan el crecimiento de la infraestructura vial, así como también la preferencia a no usar otras alternativas de transporte. Esto no solo es un problema de tiempos de espera, sino también de los problemas medioambientales generados por la acumulación de smog [3].

Por otro lado, en Guayaquil; la creciente demanda de un buen servicio de transporte público por parte de los usuarios y la falta de organización en la gestión del sistema público de transporte, ha hecho que las quejas de quienes usan este medio de transporte urbano, aumenten día a día. Por ejemplo, la llegada con retraso a sus lugares de trabajo, el irrespeto de las unidades de buses en las paradas y hasta los accidentes de tránsito por exceso de velocidad son algunas de las quejas que más se registran para este tipo de transporte urbano [4].

Al ser un problema de los ciudadanos de Guayaquil, también afecta a la comunidad universitaria, grupo que va en aumento cada año. La Escuela Superior Politécnica del Litoral (ESPOL) actualmente cuenta aproximadamente con 10894 estudiantes, de los cuales la mayoría utilizan diariamente los buses de TransEspol y buses urbanos para movilizarse; desde y hacia sus hogares. Debido a la considerable cantidad de estudiantes en esta institución superior y a la frecuencia variable de los buses, se ocasionan largas filas en los distintos paraderos en horas pico, a pesar de que se cuenta con 60 buses. En la Figura 1.1 se puede apreciar un ejemplo de la masiva cantidad de personas esperando los buses en horas pico.



Figura 1.1 Grupo de estudiantes en el nuevo terminal de ESPOL en horas pico dentro del campus Gustavo Galindo.

La aglomeración de estudiantes en los diferentes paraderos conlleva a varios problemas: El principal, la desorganización de los usuarios, lo cual puede ocasionar que cualquier individuo ajeno a la institución ingrese a las unidades de transporte; y, debido a que no existe un control de seguridad, se pueden perpetrar actos delictivos que se hacen más frecuentes especialmente en horas pico. De igual manera, pero con otro enfoque lo hacen los amigos de lo ajeno que los podemos encontrar a la vuelta de la esquina esperando y acechando a su próxima víctima, actualmente la delincuencia sigue siendo un gran problema para nuestra sociedad [5].

Otro problema concurrente es la falta de comunicación hacia los estudiantes, acerca de cambios en los horarios o recorridos de los buses. Estos tipos de cambios no lo hacen por el medio más apropiado ni en el momento más oportuno, lo que ocasiona malestar en los estudiantes, quienes acuden a la institución por información, trámites o clases.

Debido a que el transporte público de Guayaquil no tiene acceso al Campus Gustavo Galindo, el mismo que cuenta con una extensión territorial de “690 hectáreas, de las cuales 80 hectáreas están urbanizadas” dentro de los cuales funcionan varias facultades. El acceso o desplazamiento entre estas localidades es dificultoso para quienes no cuentan con movilización propia [6] Al no existir ningún tipo de sistema de vigilancia en los buses de la ESPOL, como son las cámaras de video del sistema ECU-911 en los buses públicos, las personas que comenten actos delictivos se marchan sin la posibilidad de poderlas reconocer y tener evidencia registrada del mismo. A esto se le puede adicionar la falta de instalación de botones de pánico para el conductor y usuarios, que permitan aumentar la efectividad de perpetrar un siniestro en los buses de TransEspol.

De igual forma también podría suscitarse una emergencia de salud en un usuario mientras el bus se encuentra en su recorrido regular, o el de un incendio mientras hace su recorrido por las zonas perimetrales de la ciudad. Esta clase de eventos se podrían atender y ayudar a solucionar de existir un medio de inmediata comunicación con los UPC más cercanos o el sistema integrado de seguridad ECU-911.

Debido a estos antecedentes es que se propone una solución para la gran demanda de servicio por parte de los estudiantes; y por qué no, de la mayoría de transeúntes y servidores públicos al extrapolar esta idea hacia otros sectores con la misma problemática.

1.2 Justificación.

El crecimiento de la tecnología y la tendencia mundial del internet de las cosas, hace posible el diseño y creación de equipos más avanzados que cumplen objetivos específicos. Los desarrolladores de tecnologías, como, por ejemplo, servicios de rastreo satelital, servicios de autenticación y acceso a comunicaciones inalámbricas, entre otras más, están logrando finalmente solucionar en gran medida la demanda de seguridad y transporte al mismo tiempo.

La tecnología GSM fue el medio para la conectividad de dispositivos móviles más popular y significativo en lo que se refiere a la portabilidad, en otras palabras, el hecho de poder independizar los celulares de la línea telefónica logró un avance y cambió la forma de ver los dispositivos móviles [7]. Hoy en día podemos llevar este concepto a nuevos equipos tecnológicos tales como módems USB, tabletas, computadores personales, y en localizadores GPS. Estos localizadores GPS al poder operar mediante una banda móvil tienen la ventaja de poder ser transportados de un lugar a otro y tener siempre una comunicación remota e inalámbrica [8]. Si se utilizaran esas ventajas sobre el problema de transporte público, sería posible cubrir parte de las necesidades de los estudiantes, en este caso, determinar si los buses están por salir del parqueadero principal o están a punto de llegar hacia el destino donde se encuentren los estudiantes.

En la actualidad, se han implementado sistemas de seguridad con cámaras de vigilancia [9] proporcionadas por el sistema de vigilancia ECU911, sin embargo, no se han realizado estudios basados en geo localización, ni tampoco algún tipo de sistema que permita agilizar el flujo de usuarios dentro de una zona o parada de buses. En todo caso se propone con la ayuda de este proyecto cubrir tal necesidad y permitir mejorar significativamente el servicio de transporte masivo para ser complemento a lo que ya existe actualmente. Si ya se cuenta con cámaras de vigilancia y botones de socorro, porque no complementarlo con un sistema que determine la posición exacta de los transportes. Ayudando así al estudiante a organizar su entrada y salida del establecimiento, Evitando el tráfico vehicular existente y con ello poder dar una mejor atención a los usuarios; y, solo en casos más severos, determinar la posición de la o las víctimas en casos de robos o accidentes dentro de una zona de parada de buses, así finalmente se cubrirán necesidades de transporte, una mejor atención y la seguridad, ante todo.

Los servicios de geo posicionamiento y alarmas de socorro basados en microcontroladores ofrecen al operador de transporte la posibilidad de brindar su servicio como un activo, además su mantenimiento y monitoreo podrá ser realizado por parte de los funcionarios y estudiantes de la Espol.

Otra de las razones por la cual esta propuesta es viable, es que una vez implementada brindará un servicio de bajo costo a los usuarios del sistema de transporte de la Espol, mediante el uso de los datos móviles de sus celulares. También habrá acceso a los usuarios que no posean teléfonos inteligentes a través del uso de una página web, los cuales podrán ver en sus computadoras personales o de escritorio. Permitiendo de esta manera que la comunidad de usuarios en su totalidad disfrute de las ventajas de este servicio.

Esta iniciativa tiene el beneficio de monitorear de forma precisa y constante las unidades de transporte otorgando así un alto grado de confiabilidad para el usuario (compañía), en comparación con otras opciones en materia de vigilancia que se podría implementar en el sistema de transporte en la Espol como lo son, por ejemplo, vigilancia personalizada por agentes de seguridad o vigilancia por circuito cerrado de audio y video en buses y paraderos.

Una parte muy importante a resaltar de este proyecto es el beneficio social que puede brindar al operador y usuarios del mismo, porque se elevaría el nivel de confianza de toda la comunidad de usuarios del sistema de transporte de la Espol al usar las unidades, y se estaría contribuyendo a la seguridad ciudadana de estudiantes, conductores y demás usuarios al momento de transportarse desde y hacia el campus Gustavo Galindo.

Con esto se garantizaría una mejor atención, seguridad, menores colas en las paradas de buses, y sobre todo, se vinculará lo aprendido dentro de la universidad para cubrir necesidades comunes permitiendo que las personas que nos ven desde afuera vean como ejemplo el talento humano, el conocimiento y el desarrollo tecnológico reflejado en los servicios cotidianos, y de ser posible sea imitado este conocimiento fuera de la institución hacia los servicios de transporte masivo que existen en la actualidad como lo son el servicio de metro vía, los taxis y el servicio de buses público promoviendo así la vinculación con la sociedad.

1.3 Objetivo General y Específicos.

1.3.1 Objetivo General.

- Diseñar e implementar dos prototipos, uno con servicio de Geoposicionamiento y alarmas de socorros, y otro para el servicio de paradero inteligente.

1.3.2 Objetivos Específicos.

- Diseñar el acoplamiento del sistema GPS/GSM/GPRS mediante la plataforma de Arduino.
- Diseñar la programación incluyendo de ser posibles números de emergencia.
- Diseñar los geocercos específicos con sus respectivos contactos de las unidades de UPC cercanas para los buses externos.
- Diseñar un sistema de RFID para conteo de estudiantes en las paradas.
- Diseñar los circuitos en placas PCB para ser utilizados en el prototipo de los vehículos de prueba.
- Implementar el prototipo en los vehículos de prueba.
- Analizar los resultados de las pruebas del sistema completo.

1.4 Metodología.

La metodología del proyecto empieza con el acoplamiento de un equipo de GPS existente en el mercado, es un sistema embebido llamado Arduino, la cual nos permitirá procesar los datos recibidos por parte del equipo GPS y estos a su vez ser gestionados y enviados por medio de una trama a un servidor web; o en su defecto dentro de una aplicación para teléfonos inteligentes. Para el envío de esta trama se hará uso de una placa que utiliza un chip llamado SIM900, encargada de la comunicación GSM/GPRS.

Este acoplamiento va de la mano con el diseño del algoritmo y la programación respectiva para el Arduino. La programación utiliza librerías y subrutinas lógicas que ayudan a llevar los datos generados por el equipo GPS y los convierte en resultados útiles para el usuario final.

Al tener todo ya acoplado, configurado y en funcionamiento se procederá a colocar los geocercos con datos de contactos de los UPC cercanos a la institución, de tal manera que se pueda indicar a las personas que transitan que el servicio está disponible y que puedan acudir de ser necesario a dichos contactos en caso de alguna emergencia.

Por otro lado, se diseñará un prototipo de paradero inteligente dentro de la infraestructura existente en la universidad, en este caso el punto será dentro del parqueadero cercano a la garita de entrada a la institución. Luego se diseñará un sistema que funcione mediante los carnets de los estudiantes, usando lectores RFID, contabilizando y auto configurando el envío de buses en las horas picos.

Finalmente se realizará una prueba para los dispositivos que se han diseñado y de esta manera se verificará todos los posibles errores que deben ser corregidos como son tiempos de reacción, visualización en pantalla de la posición, errores de posicionamiento, errores en el medio inalámbrico y se registrará todos los resultados obtenidos a través de la simulación en tiempo real por medio de gráficos, cálculos matemáticos y la experiencia obtenida a través de este simulacro.

1.5 Alcance.

El uso correcto de la tecnología ha permitido avances significativos en el campo de la seguridad. Tenemos como ejemplo en nuestro país que contamos con servicio de seguridad de cámaras de vigilancia dentro de espacios en donde existe mayor afluencia de público tales como parques, centros comerciales, centros educativos, hospitales, buses, taxis, y en todas las calles y avenidas de la ciudad.

Por esos se busca como prioridad un servicio de geo posicionamiento y paradero inteligente para complementar los servicios de vigilancia existente dentro de la universidad, de esa forma se eliminará en lo posible el exceso de colas dentro de una parada y contribuirá a la seguridad de los estudiantes.

Para desarrollar dicho proyecto se utilizarán módulos y microcontroladores de bajo costos y de fácil programación. También se usarán reductores de voltaje, sistemas de acoples, Sim900, módulos GPS, circuitos anti rebotes y lector de tarjetas magnéticas RFID.

Para los buses internos se creará la comunicación por medio de internet a una página web mostrando la ubicación y el tiempo en que el bus llegará a sus distintos destinos ya sea el de partida o llegada, al tener instalada las alarmas de socorro se podrá enviar los mensajes de ayuda, para socorrer algún suceso fuera de lo normal (robo, incendio, alerta médica).

Mientras que, en los buses externos, se desarrollará lo mismo que se usó en los buses internos, a diferencia de que se basará en la ruta por donde hacen los distintos recorridos para añadir los números de emergencia de los UPC que se encuentren en sus respectivas rutas. Esto se logrará mediante el uso de geocercos.

En los paraderos se planea añadir un sistema de control de estudiantes con RFID, el cual funcionara de la siguiente manera: si la demanda de estudiantes es alta, enviara un mensaje a las autoridades competentes de TransEspol para que envíen otro bus.

1.6 Estado del Arte.

En Chile un grupo de jóvenes profesionales desarrollaron la aplicación “Despierta y paradero inteligente”. Es una aplicación para celulares la cual tiene registrado los caminos que toman los transportes públicos y mediante una alarma sonora indica el acercamiento a un paradero, por lo que sería muy útil para personas ciegas [10].

Este grupo tiene varios proyectos aplicados en las diferentes ciudades de Chile, entre el cual podemos nombrar a “Planificador de Viajes”, fue implementado en el 2012 mediante el uso de GoogleMaps, el cual podía ser usado tanto en celulares como en computadores. Este programa les permite a las personas a establecer recorridos de viajes ya sea en transporte público, particular e inclusive marítimo. Este proyecto consiste en otorgar el recorrido que deseen tomar y el tiempo que se tardará en recorrer dicha ciudad en los diferentes transportes. Gracias a esto y a las referencias positivas de la aplicación fue incluida en los transportes dentro de la ciudad [11].

Así también es el caso del “Sistema Automático de tiempos de viaje”, el cual permite medir y obtener información de la autopista en tiempo real en la ciudad de Puerto Montt, esto a su vez, permite a los usuarios de transporte público y privado escoger la ruta óptima hacia donde sea su destino [12].

En Europa, en las ciudades de Barcelona y Cádiz un grupo de empresas privadas, han creado un paradero inteligente llamado la “smartquesina”. Este paradero permite realizar compras de boletos, también permite tener una planificación de la ruta, además de esto es autosustentable energéticamente gracias a los paneles solares y otras fuentes renovables. Es muy eficiente en su capacidad de identificar y contar personas para controlar la luminosidad dentro de la misma y así ahorrar energía; teniendo como único defecto ser sólo para teléfonos con software Android; y, tener lector de escáner [13].

En el Ecuador un grupo de estudiantes en la universidad técnica de Ambato desarrolló un “sistema de alarmas mediante una red de comunicación GSM para la seguridad de la empresa Auplatec en el cantón Pelileo”. Este sistema utilizando sensores detecta la proximidad de un intruso a las instalaciones de

la empresa, y dependiendo del sensor que active, se notificará mediante un dispositivo GSM los distintos mensajes que envían los sensores [14].

CAPÍTULO 2

2. MARCO TEÓRICO.

2.1 Orígenes de la tecnología GSM en Ecuador.

La tecnología celular en el Ecuador permitió avances significativos en cuanto a la comunicación y a la reducción de la brecha digital. Si se remonta a los años noventa, se podrá recordar que fue ahí cuando nuestro país comenzó dando sus primeros pasos en la tecnología móvil. En ese entonces, las empresas como CONECEL S.A [15] y OTECEL S.A [16], desarrollaron tecnología en la ciudad de Guayaquil, otorgando a sus clientes el servicio telefónico inalámbrico, utilizando la tecnología analógica AMPS, debido a la demanda masiva de comunicación estas empresas se vieron obligadas a mejorar la tecnología que ofrecían a sus clientes a medida que estos se incrementaban. Después de algunos años dichas compañías se unieron a diferentes grupos internacionales de telefonía móviles, esto permitió un avance tecnológico debido a que las compañías internacionales de telefonía móviles tenían implementado el sistema global para las comunicaciones móviles (GSM); junto con el servicio general de paquetes vía radio (GPRS).

La tecnología GSM se difundió masivamente en todo el Ecuador y fue el más aceptado por el público en general. Tenía la ventaja de poder reemplazar el equipo móvil utilizando el mismo número telefónico. Antes esto no era posible debido a que los celulares tenían incorporado en su interior el número telefónico. Este número era asignado por parte de la compañía telefónica y era casi imposible realizar un cambio de número y si aparecía un nuevo modelo de teléfono, la persona se veía obligado a esperar horas o días para realizar el cambio de número o en su defecto conformarse con otro número telefónico, pero gracias a la tecnología GSM estos problemas se acabaron.

Esto fue posible ya que el número telefónico se encuentra vinculado por medio de una pequeña tarjeta electrónica llamada tarjeta SIM la cual se la puede adquirir en cualquier tienda y utilizarlo a su vez en cualquier equipo celular. En la Figura 2.1 se pueden apreciar diferentes modelos de tarjeta SIM.



Figura 2.1 Modelos de tarjeta SIM utilizado para la tecnología GSM [17].

La aceptación masiva de las personas, sumada a las grandes posibilidades que implicaba el uso del chip permitió que se realizarán innovaciones importantes que iban a la par en cuanto a la evolución de la tecnología y los nuevos estándares que llegarían en poco tiempo.

2.2 Sistema celular de una red GSM/GPRS.

La telefonía celular en el Ecuador ha ido de evolución en evolución, las empresas telefónicas antes mencionadas comenzaron a implementar nuevos servicios de forma sistemática y compitiendo entre ellas para lograr abarcar con un mayor número de usuarios, pero esto se pudo lograr gracias a una adecuada infraestructura y a la renovación de las estructuras y redes existentes que conformar el sistema celular.

Un sistema celular móvil en particular se compone por un grupo de estaciones base (BS) que operan a través de un centro de control (CC) y a su vez brindan cobertura a un conjunto de terminales móviles (MS) y de ser necesario, las BS [18], requerirán propagar más su cobertura y dependerán de estaciones repetidoras (RS).

Estos sistemas celulares fueron desarrollados con una visión y filosofía, que es la de cubrir áreas de terreno limitado y brindar un servicio telefónico, en la Figura 2.2 se puede apreciar una pequeña distribución de una red celular y los diferentes terminales vinculados a cada estación base, así como también se muestra los elementos de interferentes dentro de la cobertura de la red celular tales como vehículos, árboles, construcciones y estaciones de servicio.

Para lograr una cobertura homogénea y correcta en toda una región se requiere replicar este esquema como si fuese una cadena de células interconectadas, siempre y cuando no se vean interferidas entre ellas y cumplan con el mínimo valor de potencia para poder operar sin problemas de propagación o de salud pública.

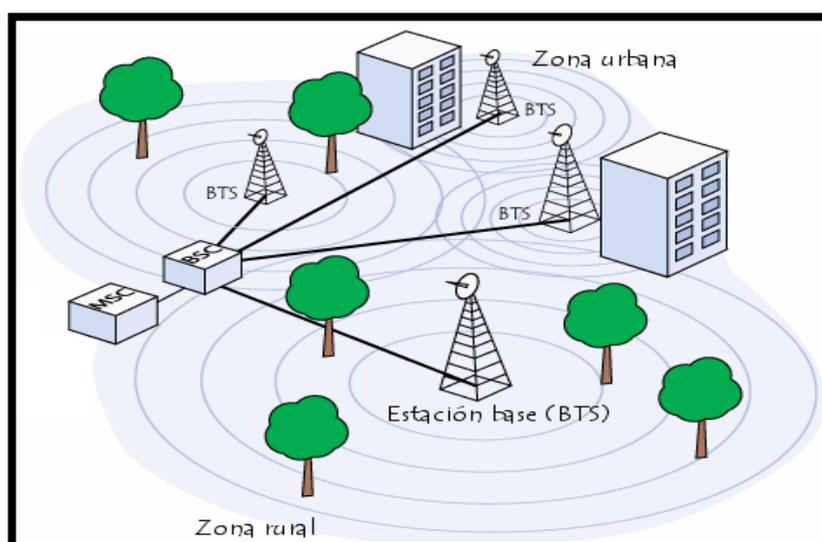


Figura 2.2 Red de telefonía celular [19].

La Figura 2.3 que se muestra a continuación permite explicar en detalle cada componente de una red celular, basada en el estudio de la tecnología GSM [20], y la arquitectura que posee, debido a que para este proyecto se va a requerir utilizar este tipo de red móvil en los dispositivos de geo localización, teniendo presente los beneficios antes mencionados, y se explicará cada uno de los bloques que la conforman.

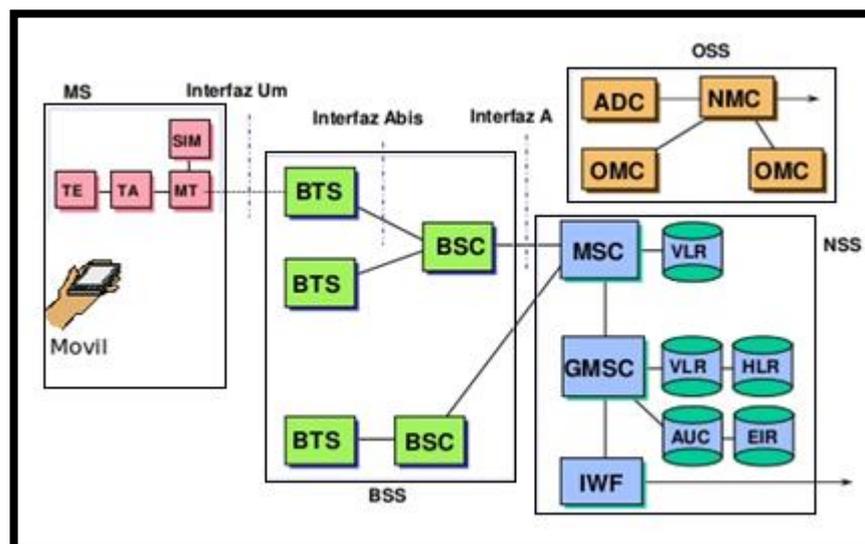


Figura 2.3 Arquitectura de una red celular GSM [21].

La Figura 2.3 que se muestra a continuación permite explicar en detalle cada componente de una red celular, basada en el estudio de la tecnología GSM [20], y la arquitectura que posee, debido a que para este proyecto se va a requerir utilizar este tipo de red móvil en los dispositivos de geo localización, teniendo presente los beneficios antes mencionados, y se explicará cada uno de los bloques que la conforman.

2.2.1 Estación móvil (MS).

Una estación móvil, es la puerta de acceso a la red inalámbrica, puede ser un celular, un modem de internet, o un equipo de geo posicionamiento. Los elementos básicos que lo conforman son los siguientes:

- El equipo móvil (ME).
- El módulo de identificación del suscriptor (SIM).

El equipo móvil es el dispositivo físico utilizado por el usuario para acceder a los servicios de una red GSM, y el módulo de identificación del suscriptor es la asociación del cliente con la red celular. Contiene

el número telefónico dentro de un pequeño chip que el usuario debe de introducirlo en el equipo móvil.

2.2.2 Subsistema de estación base (BSS).

Una estación base tiene como función el procesamiento analógico y digital que luego será propagada a través del aire en forma de ondas electromagnéticas por medio de una antena. Al propagarse la señal cubre un área específica, y la representación de este fenómeno se lo hace por medio de una Figura geométrica llamada celda, que son las áreas cubiertas de una estación base en particular.

La estación base está compuesta de los siguientes elementos:

- Estación transreceptora de base (BTS).
- Estación base de control (BSC).
- El transcodificador (TRX).

Los BTS proporcionan la comunicación entre la estación móvil y la red a través de la interfaz aire, ya que realiza el proceso de sincronización para la operación y el mantenimiento respectivo de una señal. Un BTS contiene el equipo para la transmisión-recepción de señales, equipos de comunicación y antenas por medio de la cual existe la conexión entre la BTS y la BSC. La BSC es el control maestro de la BTS y su función es de realizar los procesos de handovers, los saltos de frecuencia o frequency hopping, y los controles de frecuencia de radio sobre los BTS. Todo esto se lo realiza para mantener la llamada activa, y mantener la adaptación de la velocidad de enlace de radio al estándar de 64 Kbps. El TRX es un elemento que se encarga de convertir la señal de voz en un formato digital y viceversa.

La estación base y los subsistemas que lo conforman realizan en conjunto las siguientes operaciones:

- La señalización de la interfaz aire y la interfaz A.
- El establecimiento de las configuraciones y enlaces entre los MS y los NSS.

- El control de la red celular.
- La recopilación de datos estadísticos.
- La gestión de la movilidad.
- Y finalmente el tratamiento y codificación de la voz, en caso de ser necesario.

2.2.3 Subsistema de conmutación y red (NSS).

Estos subsistemas se encargan de llevar y administrar la comunicación hacia los distintos MS. Las tareas que realizan son las siguientes:

- El control de la llamada.
- El control de la tarifa.
- Los datos del abonado y la gestión de los servicios.
- La seguridad y gestión.
- El control del BSS.
- La señalización de la interfaz A y la PSTN.
- El funcionamiento de redes.
- Y la gestión de la movilidad.

Al igual que el subsistema BSS, este también recopila material estadístico, que luego puede ser usado para propósitos de estudio. Los elementos que conforman este subsistema son:

- El centro de conmutación de servicios móviles (MSC).
- El nodo gateway mobile switching center (GMSC).
- El registro de posición visitante (VLR).
- El centro de autenticación (AuC).
- El registro de posición base (HLR).
- El registro de identificación del equipo (EIR).

En la Figura 2.4 se muestra la comunicación entre los elementos anteriormente mencionados.

El bloque MSC se encarga de gestionar las llamadas, tarifación, y señalización de las interfaces A y PSTN. Mientras que el bloque HLR es una base de datos que contiene los datos de usuario o abonado. Mientras tanto el VLR administra las variables de memoria y las

funciones necesarias para manejar llamadas hacia y desde un usuario móvil dentro del área del VLR.

El GMSC obtiene información del HLR para el enrutamiento de una llamada dirigida hacia un teléfono móvil, es decir, es el medio de comunicación de la red GSM con las demás redes.

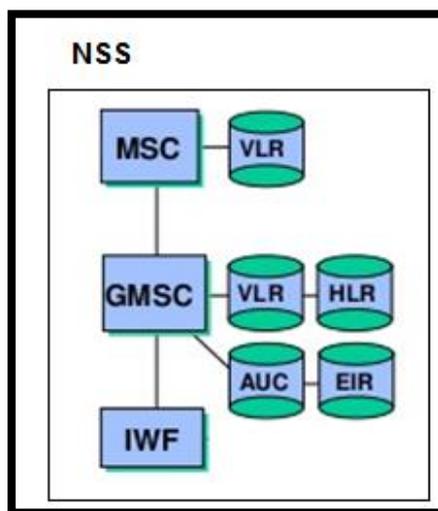


Figura 2.4 Elementos que conforman el NSS [21].

El AuC es un elemento de la red del NSS que se encarga de velar por la seguridad de la red. El EIR proporciona las configuraciones para tomar decisiones sobre la autenticación de usuarios y es capaz de soportar los procesos de encriptación dentro de la red. EL AuC y el NSS hacen su trabajo operando sobre el VLR en el cual se comunican.

2.2.4 Subsistema de operación y apoyo (OSS).

El OSS es un centro de operaciones que se encargan de resolver un número determinado de operaciones, gestiona el funcionamiento de la red y brinda mantenimiento y soporte a la misma. Esto es posible gracias a los bloques OMC y NMC. El OMC por sus siglas es el centro de operación y mantenimiento y el NMC es el centro de gestión de la red, ambos bloques trabajan en común, teniendo identificado

previamente los parámetros de servicio, los registros de datos en transmisión, tráfico y el manejo indirecto de los datos relativos de cada terminal móvil que lo identifica como abonado en la propia red celular.

Cuando se refiere a una estación móvil en particular, se debe tomar en cuenta que este tiene la función de abonado y terminal al mismo tiempo, debido a que el número de abonados se almacenan en la tarjeta SIM, y el terminal es el equipo en el que se coloca la SIM y se vincula con la red móvil gracias a que la SIM posee un código llamado identificador de terminal móvil internacional (IMSI) que le permite al operador GSM identificar claramente el abonado de la red.

2.3 Requisitos, versiones y evolución de la tecnología GSM.

Una estación móvil, sea cual sea su función debe de cumplir los siguientes requisitos para que pueda operar satisfactoriamente dentro de una red celular GSM [21].

El primer requisito que debe cumplir es el registro. El equipo debe estar registrado en el área de localización de una estación base. Los parámetros de configuración y registro se almacenan en la tarjeta SIM y en las bases de datos de la arquitectura GSM llamadas VLR y HLR. Todo esto se lo logra por medio de mecanismos lógicos programados que aseguran el sincronismo y coherencia de la información cada vez que la estación móvil realiza un desplazamiento y decida sin saberlo cambiarse de un área de localización a otra. Esta información se la comunica al VLR de la nueva estación base y simultáneamente se lo hace saber al VLR y al HLR de la estación base anterior para evitar el corte de la comunicación de una estación móvil.

El segundo requisito es la señalización. Los canales de comunicación que usa una estación móvil contienen canales dedicados para una adecuada gestión de los recursos y una mejor transmisión. El principal canal de señalización que nos concierne es aquel que se encarga de manejar el desplazamiento de un equipo móvil desde una estación base a otra. Este proceso lo hemos mencionado con anterioridad cuando explicamos el bloque de control de la estación base y se llama "handover". El hecho solo de migrar de estación a

base a otra implica un cambio de enlace, y en esos casos se debe señalar adecuadamente la comunicación dentro de los canales dedicados para dar la sensación de transparencia y fidelidad mientras se está utilizando la red móvil.

El tercer requisito se encarga de la capacidad de la estación móvil de realizar desplazamientos entre nuevas redes de diferentes operadores móviles sin la pérdida del servicio. A este requisito se le llama roaming, para ello debemos considerar lo siguiente:

- Cada operador móvil trabaja con un estándar de tecnología GSM en diferentes versiones de frecuencia.
- Cada operador maneja sus propias tarifas de consumo y precios por realizar procesos de roaming internacional y garantizar una excelente calidad de comunicación.

Al realizar el proceso de roaming, se utiliza la misma tarjeta SIM del operador local, en este caso, el abonado podrá utilizar su mismo número fuera del área de cobertura.

Las versiones de GSM operan dentro de las bandas 900, 1800 y 1900 MHz, y cada operador móvil opera dentro de las respectivas frecuencias dependiendo del espacio asignado por parte de los organismos de control de cada gobierno. Para el Ecuador el organismo competente es la ARCOTEL, sin embargo, es importante conocer acerca de cada versión con sus respectivas variantes. Por una parte, porque estamos conscientes de que el espectro radioeléctrico es un recurso muy limitado, y se necesita tener opciones para operar sobre una banda de frecuencias. Por otro lado, estas versiones promovieron la innovación de una mejorada red GSM, llamada tecnología GPRS. Aquella tecnología utiliza los recursos de operación de la red GSM para brindar de una manera más atractiva servicios de mensajería multimedia e internet sobre una estación móvil. Es allí donde es posible la utilización de un equipo GPS que sea compatible con una red que utilice una tarjeta SIM. Debido a que estos servicios de localización demandan una conexión permanente a internet y al mismo tiempo deben de ser tan portables como lo podría ser un teléfono celular.

El método de acceso a la red GSM utiliza la combinación de los accesos por división de tiempo y por división de frecuencia (TDMA y FDMA). Las versiones de la tecnología GSM son las mencionadas a continuación.

2.3.1 GSM-900.

Es un sistema que opera con 124 canales sobre dos sub bandas de frecuencia con una capacidad de 25 MHz de ancho de banda cada una. El rango de operación está dentro de los 890 a 915 MHz y los 935 a 960 MHz, cada ancho de banda por canal ocupa los 200 KHz. En la Figura 2.5 se puede observar esta configuración de los canales y se puede verificar que los 45 MHz de espacio libre representan la separación entre las portadoras, en este caso la portadora de bajada (DOWNLINK) y de subida (UPLINK).



Figura 2.5 Canalización de la banda GSM-900 [22].

2.3.2 DCS-1800.

Es un sistema que opera con 374 canales en dos sub bandas de frecuencia con una capacidad de 75 MHz de ancho de banda cada una. El rango de operación está dentro de los 1710 a 1785 MHz y los 1805 MHz hasta los 1880 MHz. Cada ancho de banda por canal ocupa los 200 KHz, esto se lo puede ver en la Figura 2.6, y se puede verificar que los 75 MHz de espacio libre representan la separación entre la portadora de bajada (DOWNLINK) y el de subida (UPLINK).

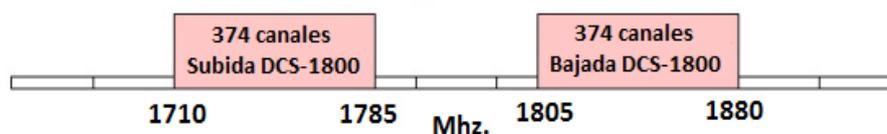


Figura 2.6 Canalización de la banda DCS-1800 [22].

2.3.3 PCS-1900.

Es un sistema que opera con 374 canales sobre dos sub bandas de frecuencia con una capacidad de 75 MHz de ancho de banda cada una. El rango de operación está dentro de los 1850 a 1925 MHz y los 1930 a los 2005 MHz, el ancho de banda por canal ocupa los 200 KHz. En la Figura 2.7 se puede observar esta configuración de los canales y se puede verificar que existe una separación de 1800 MHz entre las portadoras, esta asignación fue aprobada y utilizada en los Estados Unidos.

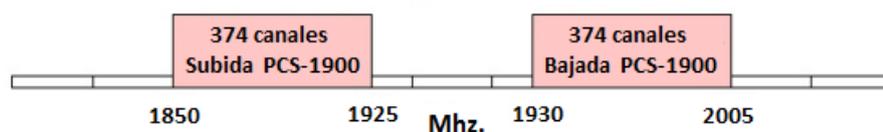


Figura 2.7 Canalización de la banda PCS-1900 [22].

Este tipo de canalizaciones la ocupa de igual manera la tecnología GPRS, esta tecnología es considerada la 2.5 G debido a que ocupa un sitio entre la segunda generación (GSM) y la tercera generación (UMTS), pero se caracteriza por que utiliza una red de segunda generación para brindar altas tasas de transferencias de datos, ideal para los servicios de internet.

El sistema GPRS reemplaza la conmutación de circuitos GSM por una conmutación de paquetes, es decir, si no se está ocupando datos o servicios por parte del abonado, las frecuencias quedan disponibles para ser empleadas por otros abonados pertenecientes a la misma red.

La arquitectura para operar una red GPRS implica mínimas modificaciones dentro de la arquitectura GSM, por lo que no implica una radical modificación de la arquitectura antes mencionada. Una red

GPRS necesita de la instalación de elementos nuevos en la red core, estos elementos serán explicados con más detalle a continuación.

2.3.4 Serving GPRS support node (SGSN).

Esta interfaz se conecta al BSC [22], y es el medio por el cual se enlaza a la red GPRS siempre y cuando los abonados soliciten el servicio. El SGSN intercambia los datos de redes externas y se encarga de encapsular las tramas y los paquetes de datos.

2.3.5 Gateway GPRS support node (GGSN).

Es el dispositivo que se conecta a una red externa, haciendo que la infraestructura de la red GPRS sea vista desde afuera. En la Figura 2.8 se tiene un ejemplo de este hecho, teniendo desde un extremo un teléfono con capacidad GPRS intentando conectarse a un servicio de internet para ser utilizado como un modem de internet sobre una laptop que intenta enviar un archivo y por el otro lado se tiene un usuario revisando la información recibida a través de su computadora. El GGSN se encarga de encaminar los datos provenientes de un teléfono hacia una red externa, pero siempre verificando si la dirección ip se encuentra activa, y de ser así, envía los datos al SGSN, para después ser empaquetada y transmitida hacia las redes externas.

De esta arquitectura, es apreciable que algunos bloques no han cambiado su naturaleza, debido a que GPRS proviene de la evolución del estándar GSM, es por ello que notamos la presencia de la estructura de conmutación de circuitos propia de GSM superpuesta a su vez con la red de conmutación de paquetes proporcionada por GPRS, es por esto que existen nuevas conexiones, actualizaciones de los bloques y nuevas interfaces. Las interfaces del sistema GPRS son:

- Gd: Interfaz entre SMS-Gateway y el SGSN.
- Gs: Interfaz entre el SGSN y la MSC/VLR.
- Gp: Interfaz definida entre dos GSN de la misma PLMN.
- Gb: Interfaz de una BSC al SGSN.

- Gn: Interfaz entre un GSN de la misma PLMN.
- Gi: Interfaz del GGSN a las redes externas X.25 o IP.

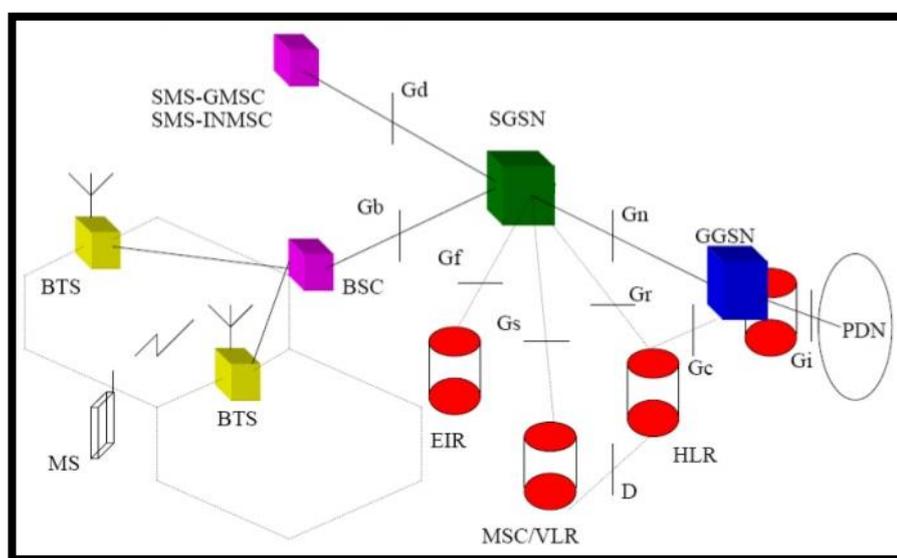


Figura 2.8 Arquitectura GPRS [23].

Debido a la capacidad que tiene el GGSN de “enrutar” los datos del teléfono hacia las redes externas, podemos considerar al GGSN como un “ruteador”, ya que no permite ver las características de la red GPRS, sino que más bien los presenta como si fueran direcciones IP conocidas por las computadoras

En la Figura 2.9 se presenta un esquema mucho más global del sistema GSM coexistiendo con el nuevo sistema GPRS.

2.4 Sistemas de Geo Posicionamiento.

Una vez vistas las características que envuelven al sistema GSM/GPRS [24], su arquitectura, las bandas en la que opera, los requisitos y su evolución, a la par de los beneficios que ofrecen estas redes, es posible continuar explicando sobre la teoría de la geo localización.

Uno de los sistemas encargados de procesar y tomar datos terrestres son los satélites GPS [25], estos satélites trabajan conjuntamente formando una red

compuesta de 24 satélites que orbitan la tierra a una distancia de aproximadamente 20 mil kilómetros de altura. Los satélites se desplazan de manera independiente, pero con trayectorias sincronizadas de tal manera que se eviten colisiones entre ellas.

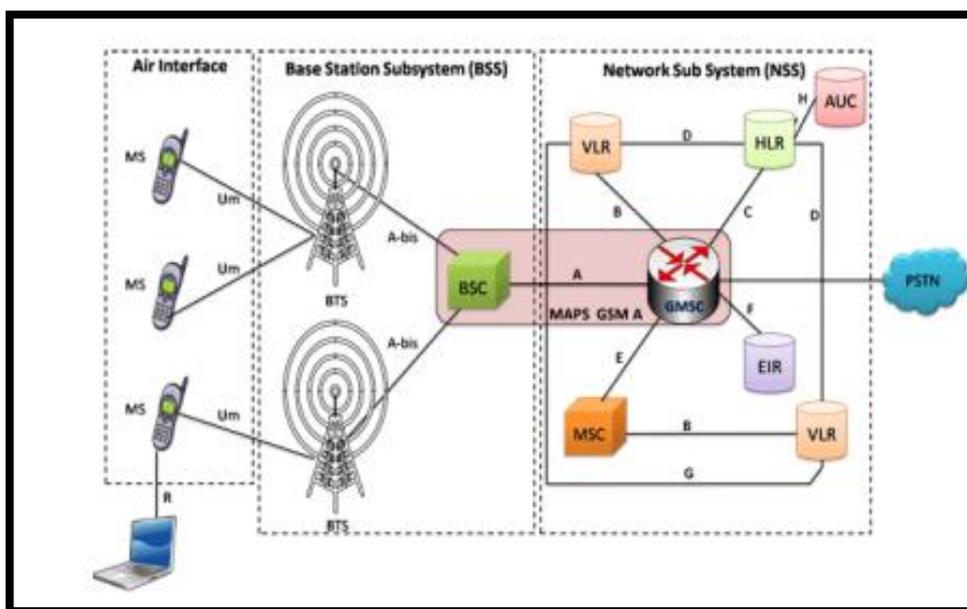


Figura 2.9 Arquitectura global de la red GSM/GPRS [24].

Al estar en el espacio ellos pueden determinar la posición mediante un receptor que se enlaza con un máximo de 4 satélites, los cuales mediante un cálculo de tiempo y distancia son capaces de obtener la posición aproximada de un objeto que se encuentre en movimiento o estático. Este cálculo recibe el nombre de “trilateración inversa” y se basa principalmente en la sincronización del reloj interno del satélite con respecto al resto de satélites y el tiempo en el cual tarda en receptarse la información. Se puede ejemplificar este proceso por medio de la Figura 2.10 en el que se muestra como los satélites trabajan en conjunto para determinar una posición en particular.

Un satélite emite un mensaje de navegación con una velocidad de 50 bit por segundo por medio de una portadora de frecuencia de microondas y para poder sincronizarse hace uso de dos señales, L1 y L2. En el caso de L1 su

frecuencia de operación es de aproximadamente 1576 MHz y L2 es de aproximadamente 1228 MHz, y son de distinta frecuencia para evitar la posible interferencia cocanal.

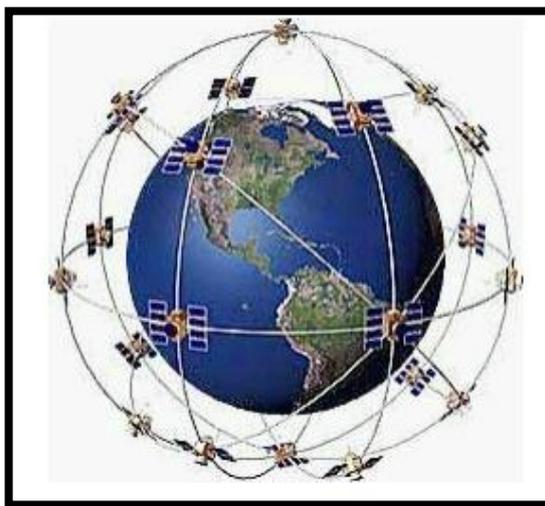


Figura 2.10 Operaciones de los satélites [24].

Una de las características relevantes de estos satélites son la precisión en el tiempo, y esto lo logran gracias a que su reloj interno es el más preciso de todos, el reloj atómico. Estos relojes utilizan materiales de rubidio o cesio para mantenerse con una exactitud de 10^{-9} segundos por día, y de esa forma los satélites mantienen la exactitud de sus mediciones dentro de un rango de 50 ns.

Un sistema de geolocalización se basa en la determinación, ubicación y posicionamiento de un objeto en un determinado sitio. El geo posicionamiento se lo define en términos de una referencia de latitud y longitud utilizando un sistema de coordenadas tridimensionales, y consiste en tomar un punto de origen hacia donde se encuentra el objeto en cuestión y expresar la ubicación en coordenadas de latitud y longitud. Estas coordenadas poseen unidades angulares, en el caso de la latitud representa el ángulo que forma de norte y sur en algún punto con respecto al ecuador, y en el caso de la longitud el ángulo se mide desde un punto de la tierra a lo largo del ecuador. El meridiano

de Greenwich es la longitud 0 en la mayoría de los casos. Cuando es un objeto, es necesario averiguar si este posee movimiento o no, de ser así, es necesario definir el tipo de posicionamiento que se debe tomar y como lo tiene que hacer.

2.4.1 Posicionamiento estático.

Es el método más convencional de un sistema GPS, este método efectúa las observaciones mediante dos o más receptores operando simultáneamente en periodos de tiempo mayores a una hora

2.4.2 Posicionamiento dinámico.

En este método se le asigna a un receptor como base en un punto fijo de coordenadas, esto lo hace mientras el otro receptor empieza a desplazarse y a tomar mediciones acordes al tiempo en el cual ambos receptores están sincronizados. Gracias a esta técnica es posible obtener coordenadas sobre el desplazamiento del receptor que se encuentra espaciado una distancia d , y se determina la velocidad que este posee.

2.4.3 Proceso de triangulación y posicionamiento.

Cuando un grupo de satélites se sincronizan y toman mediciones de distancias entre un satélite con respecto al otro [26], este método se lo denomina triangulación, esto se lo hace con el fin de conocer la posición exacta de un objeto.

El método de triangulación se basa en el envío y recepción de señales de radio. Esta señal le permite al satélite receptor dibujar una especie de esfera imaginaria centrado en un eje de coordenadas tridimensionales de radio D_1 , después un segundo satélite realiza el mismo procedimiento y dibuja una segunda esfera con radio D_2 y al final un tercer satélite realiza el mismo proceso, pero con radio D_3 . La intercepción de estas tres esferas permite determinar la ubicación del objeto, la onda de radio viaja a la velocidad de la luz aproximadamente $3 \times 10^8 \text{ m/s}$.

Estas distancias se basan en la ecuación del movimiento, esta posee la siguiente expresión

$$d = t \times c \quad (2.1)$$

En donde:

d = Distancia entre el satélite y el objeto (m).

t = Diferencia de tiempo entre dos puntos.

c = Velocidad de las ondas electromagnéticas equivalente a la velocidad de la luz.

2.4.4 Geocercos.

Un administrador de un sistema GPS debe asegurarse de señalar adecuadamente un área geográfica en la cual se van a proceder a realizar actividades de rastreo, es por eso que se utilizan estos sistemas de alerta para indicar mediante mensajes de texto o correo electrónico que existen objetos fijos o en movimiento cercanos a las áreas señalizadas.

Un geocercos [27], es una delimitación visual que permite vigilar un área específica con la finalidad de tomar acciones pertinentes sobre el mismo, en nuestro caso permitirá determinar la existencia o no de vehículos de transporte cercanos a nuestra área.

Un geocercos se lo puede realizar mediante un software de rastreo satelital, este software es capaz de alertar a un usuario si un objeto se encuentra fuera o dentro de un perímetro determinado. Por ejemplo, en la Figura 2.11 se puede ver un software en el cual se configura los parámetros y el delineamiento que se sigue de una ruta específica.

Para establecer las propiedades de un geocercos se debe tomar en cuenta las siguientes propiedades:

- El nombre que va a tener un geocercos.
- La descripción o característica que posee dicha área.

- Una escriba o grafico geométrico como puede ser una línea, polígono o círculo, para este último se debe de indicar el radio que este posee en metros.
- Una cuenta de administración, ya que esta permite al usuario final asignarle o no el acceso a un geocercos.
- El área o perímetro que ocupa al crear un geocercos.
- La dirección de origen.
- La ruta de inicio y ruta de fin.
- Y finalmente un color que delimite y diferencie dos o más geocercos cercanos.

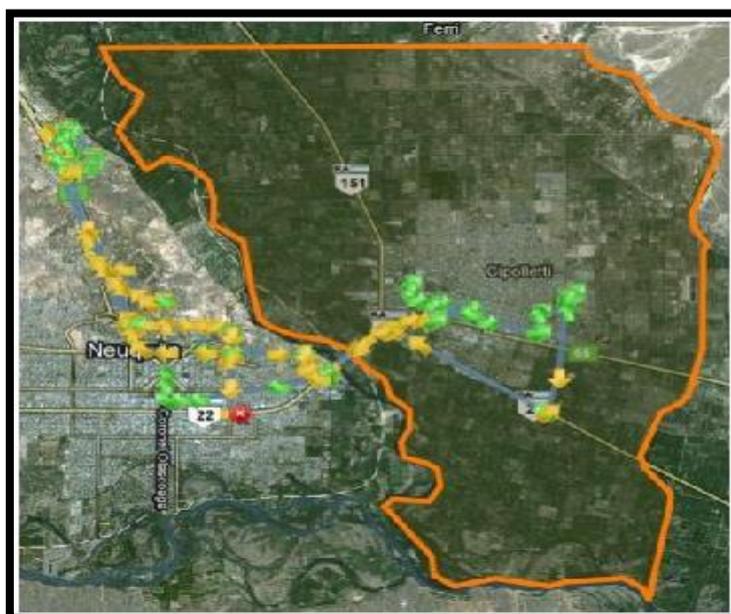


Figura 2.11 Ejemplos de diseño de Geocercos [27].

En pocas palabras, diseñar este tipo de cosas no resulta muy complicado, solo se necesita conocer muy bien el área en donde se va a proceder a hacer el estudio para ir delimitando adecuadamente los geocercos y poseer el software que mejor se ajuste a las condiciones del problema.

2.5 Sistemas embebidos.

Durante mucho tiempo la tecnología digital dependió en cierta medida de integrados y controladores que realizaban procesos pequeños y específicos dentro de un circuito electrónico. Estos procesos han permitido que se den pasos gigantes en diseño y creación de sistemas capaces de realizar tareas muy variadas. Por ejemplo, los módulos GGSN de una red de telefonía GPRS contienen en su interior varios de estos controladores que le ayudan a la red poder ser vista mediante una red IP, los ordenadores y computadores en la actualidad lo tienen, cualquier dispositivo electrónico y digital está compuesto de ellos. En la gran mayoría de los casos siempre hemos dependido de ellos, por ello es imprescindible conocer acerca de ellos y su funcionalidad de manera general.

Los sistemas embebidos [28], han sido creados para cumplir varias o pocas funciones dedicadas, y se les dice embebidos porque forman parte integral de un hardware completo. Estos forman parte de muchos sistemas tecnológicos dentro de la industria en general, por ejemplo en la industria automotriz tenemos los navegadores GPS, en la aviación están los pilotos automáticos y el control de aterrizaje dentro de un avión, en la industria de la telefonía y las telecomunicaciones tenemos los teléfonos celulares, los routers y módems como el caso de los módulos GGSN en la tecnología GPRS, y en la industria domótica están los controladores de microondas, las aspiradoras automatizadas, el cortador automático de césped, y el controlador de una casa domótica.

Un ejemplo de un sistema embebido se lo puede ver en la Figura 2.12 en el cual se muestra una placa FPGA desarrollado por la empresa Papilio.

Los elementos que componen un sistema embebido lo conforman los microprocesadores, los microcontroladores, la lógica programable y los circuitos digitales y analógicos.

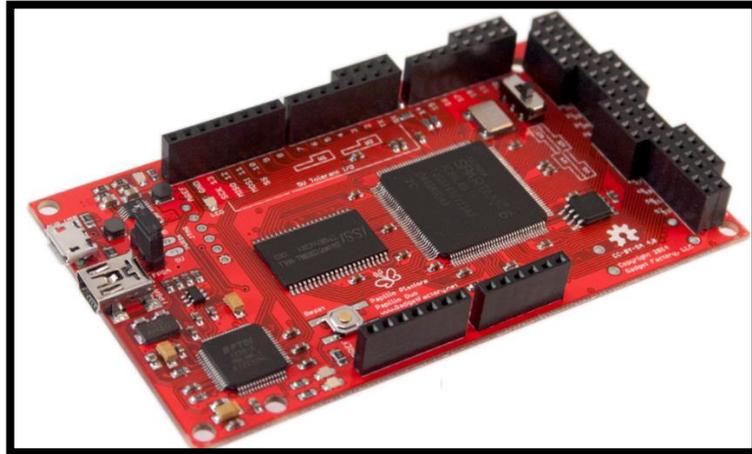


Figura 2.12 Sistema embebido [28]

Todos estos dispositivos integran en conjunto un sistema embebido, a continuación, se explicará en detalle que representa cada componente dentro de un sistema embebido, adicionalmente se dará a conocer los sistemas embebidos que se van a utilizar para el desarrollo de nuestro proyecto, y finalmente se explicará los componentes interconectados dentro de los sistemas embebidos para poder ser ensamblados y acoplados adecuadamente.

2.5.1 Los microprocesadores.

Es un circuito integrado que posee un chip que es el cerebro del sistema tiene la característica de ser programable, se encarga de llevar el proceso de la información y de ejecutar la mayoría de instrucciones que permiten la ejecución de un sistema operativo o de un programa para que el usuario pueda interactuar y realizar operaciones específicas. Estos procesos son resueltos mediante instrucciones de máquina de bajo nivel por medio de las operaciones fundamentales como son la aritmética y la lógica matemática.

2.5.2 Los microcontroladores.

Por su parte, los microcontroladores son encargados de realizar tareas específicas dependiendo de las órdenes grabadas en su memoria,

estos dispositivos están compuestos de una unidad central de procesamientos, memorias y periféricos de entrada y salida. Estos generalmente utilizan palabras de 4 bits y operan con velocidades que van desde los 31 KHz hasta más allá de los 8 MHz, normalmente operan siempre y cuando exista estimulación de los periféricos de entrada.

2.5.3 La lógica de programación.

Es todo tipo de lenguaje de programación orientado al hardware también llamado HDL, se caracteriza por manejar la operación de los circuitos electrónicos digitales. Los lenguajes de máquinas son muy semejantes a los lenguajes de programación comunes, pero se diferencia del resto ya que la noción del tiempo y el manejo de estados o lazos es lo que impera dentro de los lenguajes de máquina. Estos HDL son utilizados para darle especificaciones ejecutables al hardware, de tal manera que un desarrollador pueda modelar y simular el comportamiento electrónico de un dispositivo en particular antes de ser construido físicamente. La ventaja de los HDL es que un usuario puede manipular el funcionamiento de un equipo de tal manera que pueda resolver problemas de la vida cotidiana, cuando se logra implementar estos dispositivos en la vida real se procede al desarrollo de placas o circuitos lógicos reales [29].

2.5.4 Arduino UNO.

Una de los sistemas embebidos más utilizado hoy en día en los proyectos de ingeniería es el Arduino UNO. Este es un sistema embebido que se conformados principalmente por un microcontrolador, memorias y componentes electrónicos activos y pasivos. Todos ellos interconectados de manera lógica formando una red de hardware que realizan actividades específicas regidas por la lógica programable.

El hardware básicamente consiste en una placa de circuito impreso el cual usa un microcontrolador ATmel AVR (AVR es una familia de microcontroladores del fabricante estadounidense Atmel, el cual

cuenta con una arquitectura de diseño simple y fácil programación) [32]. Posee puertos digitales y analógicos de entrada y salida que pueden conectarse a placas de expansión, para ampliar su funcionamiento.

El Arduino UNO utiliza un microcontrolador ATmega328, el cual consta además de una placa sencilla con 14 pines que pueden configurarse como entradas o salidas de los cuales 6 son para la salida de la modulación por ancho de pulso (PWM), 6 que se pueden usar como entradas analógicas, su voltaje de operación es de 5 Voltios, posee una entrada USB que permite la comunicación entre la computadora y la placa, esta tarjeta funciona con un suministro de energía de 6 a 20 Voltios DC. Posee 32 KB de memoria Flash para el almacenamiento del código, de los cuales 0,5 KB son utilizados para el gestor de arranque, 2 KB para la memoria SRAM y 1kb para la memoria EEPROM. En la figura 2.13 se puede observar el esquema de un Arduino uno.

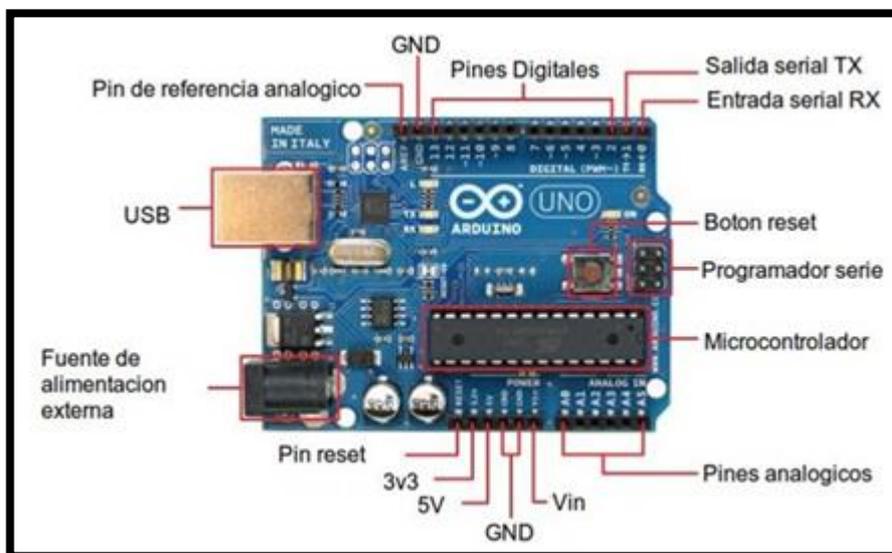


Figura 2.13 Placa Arduino UNO [30].

2.5.5 Arduino Mega 2560.

El Arduino Mega es una placa que posee una entrada USB en la cual permite la comunicación entre el computador y la placa. Esta tarjeta funciona con un suministro de energía de 6 a 20 voltios DC. La placa está conformada por un microcontrolador ATmega2560 que posee una capacidad de memoria flash de 128 KB para el almacenamiento del código, de estos 128 KB, 4 KB son dedicados para el gestor de arranque, 8 KB para la memoria SRAM y 4 KB para la memoria EEPROM. Contiene 54 pines digitales configurable por código como entradas y salidas digitales que manejan un voltaje de 5 voltios, cada pin suministra un máximo de 40 mA con resistencias de pull-up de 20 a 50 kOhms. También cuenta con ciertos pines para funciones específicas.

Para los puertos seriales contamos con el par 0 en los pines 0 y 1, el par 1 en los pines 19 y 18, el par 2 en los pines 17 y 16, y el par 3 en los pines 15 y 14, estos pares de puertos seriales tienen un pin de transmisión llamado TX y un pin de recepción llamado RX que permite enviar y recibir datos por medio de la comunicación serial, el puerto USB de la placa está conectado al par 0 de los pines 0 y 1.

Para el caso de las interrupciones externas, se utilizan el pin 2 para la interrupción 0, el pin 3 para la interrupción 1, el pin 21 para la interrupción 2, el pin 20 para la interrupción 3, el pin 19 para la interrupción 4 y el pin 18 para la interrupción 5. Estas interrupciones operan por medio de un valor bajo, un flanco de reloj ascendente o descendente o un cambio de valor.

Para la modulación PWM se hace uso de los pines del 2 al 13 y del 44 al 46, esta modulación proporciona una salida PWM un tamaño de 8 bits.

Para las entradas analógicas, la tarjeta posee 16 pines de entrada analógica de cual se tiene una resolución de 10 bits, con lo que se tiene un abanico de 1024 valores, medidos desde tierra hasta los 5 voltios.

Para las conexiones SPI se tiene disponibles los pines 50 para MISO, 51 para MOSI, 52 para SKC y 53 para SS, todos ellos admiten la comunicación SPI. Para la comunicación I2C se tiene los pines 20 para SDA y 21 para SCL [30]. En la Figura 2.14 se puede ver el esquema de una placa Arduino mega 2560.

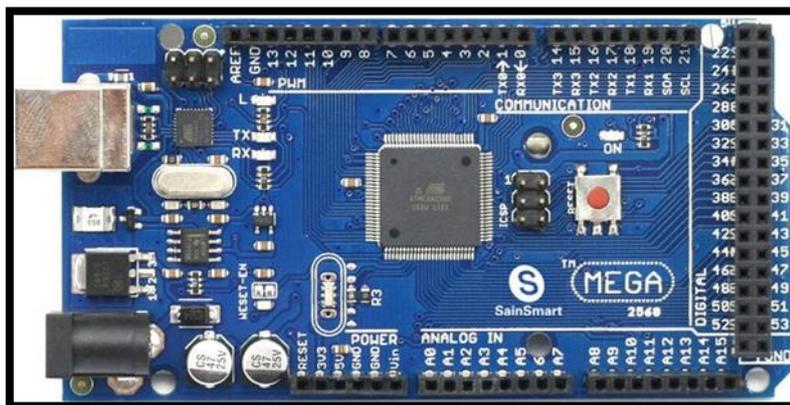


Figura 2.14 Placa Arduino mega 2560 [30].

2.6 Tipos comunicación y Módulos.

Es necesario tener una breve introducción de los tipos de comunicación que manejan los sistemas embebidos, así como también los módulos responsables de la comunicación GSM/GPRS.

2.6.1 Comunicación serial.

La comunicación serial consiste en el envío y recepción de datos de un bit a la vez de manera secuencial por medio de un canal de comunicación o bus. Se diferencia de la comunicación en paralelo en la cual los datos transmitidos llegan al mismo tiempo.

Las ventajas de este tipo de comunicación es que permite usar un número pequeño de líneas de transmisión que en condiciones normales esto mejora el beneficio costo, en Arduino es posible este tipo de comunicación a través de los pines de comunicación serial visto anteriormente, y son configurable por medio de código mediante la

subrutina Seria-begin (rate), en donde el valor de “rate” representa el valor de la velocidad en baudios en la cual se va a comunicar el dispositivo.

2.6.2 Comunicación I2C.

Al igual que la comunicación serial, la comunicación I2C, es un protocolo de comunicación que inicialmente fue utilizado dentro de los reproductores de CD, videojuegos y ordenadores de escritorio, con la finalidad de que los circuitos integrados y dispositivos internos pudiesen comunicarse entre ellos y estos a su vez hacia nuevos periféricos dentro de su arquitectura. Esta comunicación opera mediante la asignación de roles, en este caso, quien envía la orden será llamado maestro, y quien recibe la orden será llamado esclavo, es por eso que el sistema de bus que caracteriza a la comunicación I2C es un bus maestro y esclavo.

Las características de este protocolo son las siguientes:

- La comunicación es controlada mediante la señal de reloj, es por eso que se lo conoce como un sistema de comunicación síncrono.
- La velocidad de transferencias varía entre los 100 Kbits/s hasta los 3,4 Mbits/s.
- Utiliza el protocolo de acceso al bus maestro – esclavo y es capaz de soportar múltiples maestros.
- Cada dispositivo solo puede comunicarse mediante una dirección única dentro del bus

En la Figura 2.15 se puede observar el intercambio de información de un modelo maestro esclavo en función de su nivel de enlace.

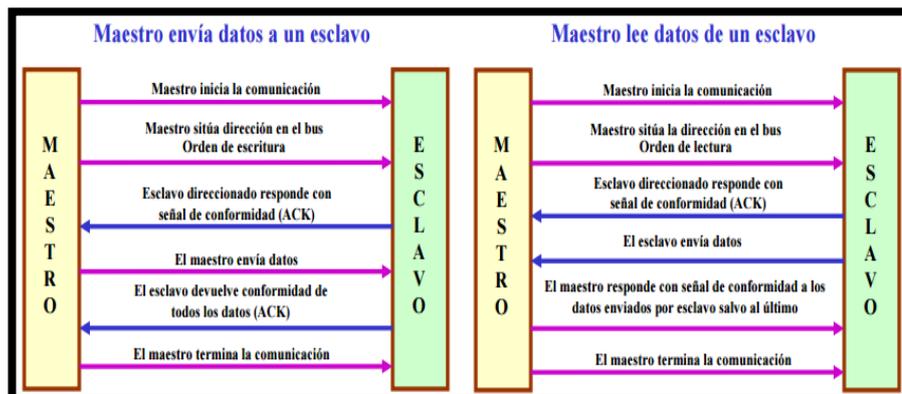


Figura 2.15 Comunicación maestro esclavo de un enlace I2C [31].

A pesar de ser un sistema anticuado y lento, este tipo de comunicación reduce el tamaño de un circuito debido a que ocupa menos espacio y es muy barato en comparación con otros sistemas de comunicación, y debido a esto muchos microcontroladores poseen este tipo de comunicación, esto incluye a los sistemas embebidos como Arduino y los módulos de comunicación, los cuales poseen este tipo de configuraciones.

2.6.3 Comunicación SPI.

Es una comunicación híbrida entre la comunicación serial y la comunicación I2C porque es un sistema serial, usa el rol maestro esclavo y funciona de manera síncrona, el bus SPI contiene una línea de reloj (SCLK), una línea de dato entrante (MOSI), una línea de dato saliente (MISO) y una línea de selección de chip (CS) que permite la conexión/desconexión de la operación de un dispositivo para su posterior sincronización con la línea de reloj.

En la Figura 2.16 se puede observar la interconexión entre un microcontrolador maestro y tres dispositivos esclavos.

Los beneficios de utilizar este tipo de comunicación es que la transferencia no se limita a palabras de 8 bits, es una comunicación full

dúplex y garantiza mayor tasa de transmisión en comparación con el I2C, las desventajas de esta comunicación es que no es posible tener varios servidores conectados a la vez, solo funciona en cortas distancias y no existe una señal de control que evite que un servidor envíe información mientras un cliente no está presente [31].

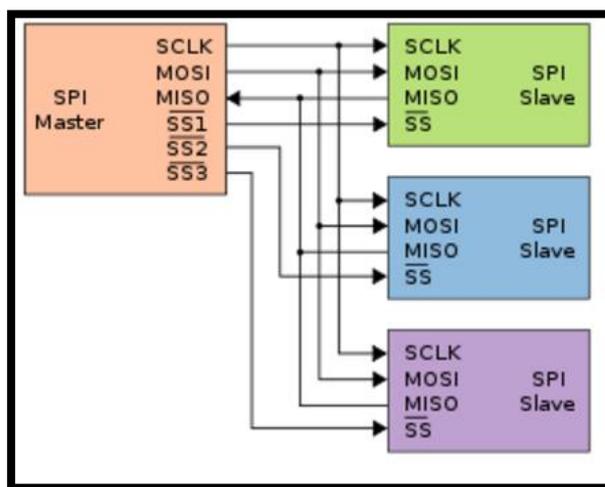


Figura 2.16 Comunicación maestro esclavo enlace I2C [31].

2.6.4 Comandos AT.

Es un lenguaje estándar para la configuración y parametrización de módems, y estándar abierto para distintos módulos de comunicación de radiofrecuencia como lo puede ser el módulo SIM 900. Este lenguaje fue desarrollado por la compañía Hayes y fue adoptado por la ITU-T para el desarrollo de controladores específicos de módems superfluo en el protocolo V.250.

Dentro de las especificaciones, se recomienda que los comandos AT sean escritos en mayúsculas, pero en la actualidad, los dispositivos que operan mediante comandos AT exigen que sean escritos en minúsculas, y gracias a estos comandos es posible ajustar los parámetros de configuración de los dispositivos para que puedan funcionar adecuadamente en una determinada aplicación.

Los módulos de comunicación pueden ser configurados mediante los comandos AT [32], cada comando AT permite modificar las configuraciones internas de los módulos para que trabajen en un rango de operación establecido por el usuario. Los principales comandos AT que se pueden utilizar son:

- AT: Sirve para verificar si el módulo SIM900 está funcionando adecuadamente para entrar en modo comando. Al enviar AT el SIM deberá contestarnos con un OK.
- AT+CGMI: Veremos en nombre del fabricante
- ATI: Ver la información del producto.
- AT+IPR=?: Preguntar el Baud Rate en el que puede operar el SIM
- AT+IPR?: Sirve para preguntar el Baud Rate actual
- AT+IPR=XXXX: Configuremos a la frecuencia deseada
- AT+COPS?: Nombre de la compañía telefónica
- AT+CGSN: Visualizar el IMEI del chip utilizado
- AT+CSCS?: Tipo de texto
- AT+CSCS="XXX": Configurar a tipo de texto
- AT+CMGF?: Ver el formato de un mensaje, ya sea PDU (0) o SMS (1)"
- AT+CMGS=04455XXXXXXXXX: Enviar un SMS Se despliega el símbolo mayor que > Escribir mensaje y al finalizar presiona Ctrl+Z retornará OK si el SMS se envió correctamente.
- AT+CMGL=ALL: Sirve para ver todos los mensajes que nos han llegado al SIM
- ATD04455XXXXXXXXX: Sirve para hacer una llamada a cualquier teléfono móvil
- ATA: Sirve para contestar una llamada
- ATH: Sirve para colgar una llamada

2.6.5 Módulo de comunicación SIM900.

El SIM900 es un sistema embebido fabricado por la empresa SIMCom. Este dispositivo trabaja en la red GSM/GPRS diseñado con un microprocesador AMR926EJ-S core. Debido a su pequeño tamaño permite ser acoplado de una manera sencilla y práctica para múltiples propósitos y funciona con un suministro de voltaje de entre 3,2 hasta los 5 voltios, entre ellos pueden ser aplicaciones de telefax, sistemas de voz y SMS, y conectividad móvil para usos de redes de internet.

Este módulo opera dentro de las bandas desde la 850 hasta el 1900 MHz respectivamente, para el manejo de voz, mensajes de texto, datos, fax y garantiza un bajo consumo de potencia debido a su modo de bajo consumo incorporado en su interior.

El módulo SIM900 cuenta con Quad-Band 850/900/1800/1900 MHz, múltiple ranura GPRS clase 10 y clase 8, posee una función de estación móvil de clase B, cumple con la norma GSM fase 2/2, tiene una antena simple de cancelación de interferencias (SAIC), su temperatura de funcionamiento es entre los -40°C hasta los 85°C y la dimensión de esta tarjeta es de 24 x 24 x 3 mm y tiene un peso de 3,4 gramos [33].

Obedece a comandos AT (GSM 07.07, 07.05 y SIMCOM de comandos AT mejorada) y posee un Kit de herramientas de aplicaciones SIM.

Tiene una alimentación de 3,2 hasta los 4,8 voltios y es configurable para manejar corrientes de 1,0 mA en el modo de reposo.

Además, posee interfaces para poder ser manejado adecuadamente, la interfaz externa SIM de 3V / 1.8 V, una interfaz de audio analógico, un respaldo de RTC, una interfaz SPI, una interfaz serial, un pad conector en caso de utilizar una antena externa. En la Figura 2.17 se puede ver la forma que tiene este módulo GSM/GPRS.

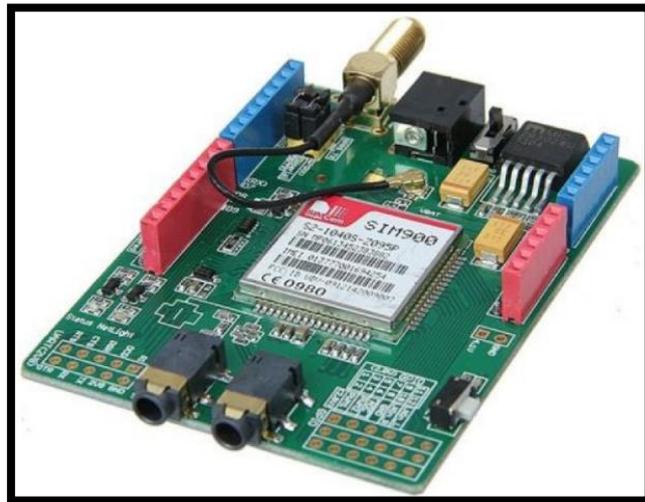


Figura 2.17 Placa SIM900 [34].

2.6.6 Módulo de comunicación SIM800.

El SIM800 tiene características muy parecidas con respecto al SIM900, como por ejemplo trabaja en cuatro bandas 850 / 900 / 1800 / 1900MHz, puede usarse como celular con respecto a voz, datos y mensajes.

Teniendo en cuenta que también puede ser manejado con comandos AT, y su diferencia radica en que puede captar radio FM y usar Bluetooth [35]. En la Figura 2.18 se puede observar la placa del módulo SIM800.

Las características de este módulo son:

- Quad-band 850/900/1800/1900MHz
- Multiple ranura para GPRS clase 12 y clase 10
- Bluetooth
- Controlado mediante comandos AT
- Rango de voltaje de operación 3.4 ~ 4.4V
- Modo de ahorro de energía

- Su temperatura de funcionamiento es entre los -40°C hasta los 85°C .

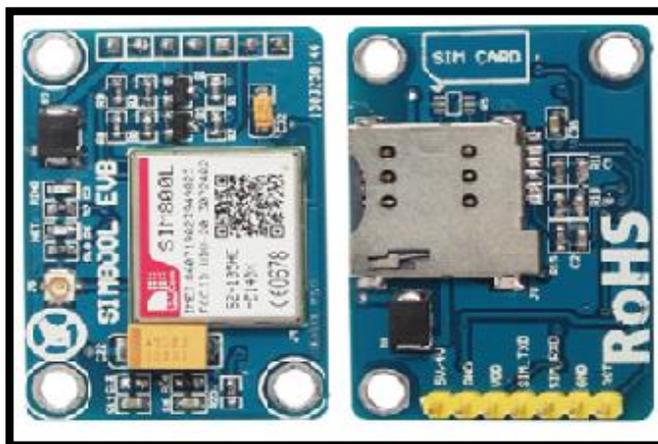


Figura 2.17 Placa SIM800 [35].

2.7 Sistemas Discretos.

En la siguiente sección se comentará acerca de los elementos discretos a utilizarse dentro del proyecto como son los capacitores, los sistemas anti rebotes, los reguladores lineales y los reguladores conmutados.

2.7.1 Capacitores.

Son dispositivos simples capaces de almacenar energía y carga eléctrica en su interior, son normalmente vistos en cualquier dispositivo electrónico como transmisores y receptores de radiofrecuencia, filtros de fuentes eléctricas y flashes electrónicos en cámaras digitales debido a sus propiedades eléctricas y su capacidad de almacenamiento. Se pueden observar varios tipos de capacitores en la Figura 2.19.

Se conforma por dos conductores separados por un material dieléctrico aislante, al ser sometidos ambos extremos a una carga eléctrica, cada conductor tendrá la misma carga eléctrica, pero con diferente signo, esto establece una diferencia de potencial entre ellos y su relación carga sobre diferencia de potencial establece la cantidad constante natural del capacitor llamada capacitancia, cuya unidad es el Faradio.

Los capacitores pueden ser de distintas formas, entre ellas tenemos los capacitores de placas paralelas, los capacitores cilíndricos y muy poco frecuente los capacitores esféricos cuyo ejemplo se lo puede tener al observar el comportamiento terrestre de los mares, la tierra y la atmosfera terrestre, y debido a eso nuestro planeta puede ser considerado como si fuese un gran capacitor natural.

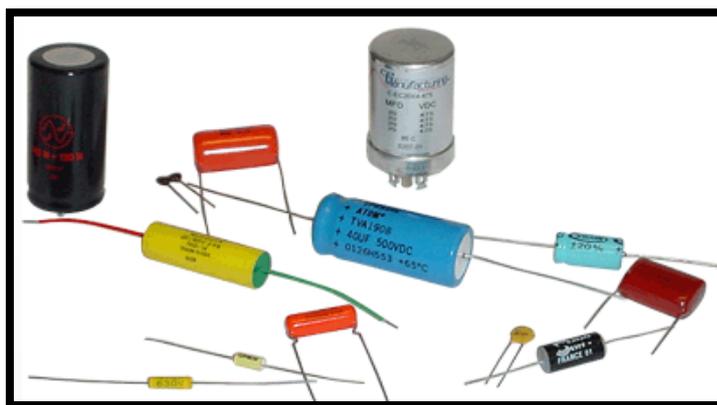


Figura 2.18 . Diferentes tipos de Capacitores [36].

2.7.2 Tiempo de carga y descarga de los capacitores.

Mediante prácticas de laboratorio fue posible la determinación en tiempo real de los tiempos que le toma a un capacitor tener su máxima carga eléctrica y a su vez llegar a su mínima carga eléctrica. Pero las ecuaciones que rigen este comportamiento suelen ser muy complejas debido a su naturaleza y combinación con otros dispositivos como lo son las resistencias. Como se puede observar en la Figura 2.20 a continuación, se tiene un pequeño circuito constituido por una fuente de voltaje U , un capacitor C , una resistencia R y un switch que conecta y desconecta el circuito.

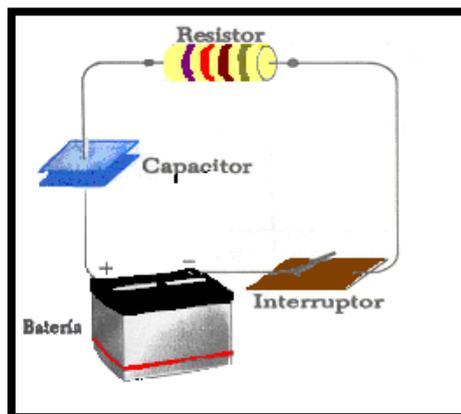


Figura 2.19 Circuito conformado por un capacitor, resistencia, fuente de voltaje [36].

Cuando el capacitor se somete a una diferencia de potencial o voltaje U entre sus extremos, los capacitores por lo general no se cargan inmediatamente, sino que más bien toman un tiempo en función de su capacitancia y de la resistencia R del circuito. A esta función se la denomina constante de tiempo y está en términos del producto del valor del capacitor por el valor de la resistencia, es decir:

$$T = R \times C \quad (2.2)$$

La unidad de esta constante es el segundo y es la medida de rapidez de reacción del capacitor.

Si asumimos que el switch se cierra, existirá una corriente eléctrica circulando alrededor del circuito, esta corriente tiene la siguiente expresión:

$$i = I_0 e^{-t/T} \quad (2.3)$$

En donde la letra e representa la base de los números logaritmos naturales conocido también como constante de Euler cuyo valor aproximado es 2,72, este valor de corriente tiende a ser cero a medida que el capacitor adquiere voltaje que lo proporciona la fuente.

Durante el proceso de carga del capacitor mediante la corriente de la ecuación 2.3, si consideramos un valor de carga q en un instante después de cerrar el switch, entonces su expresión será:

$$q=Q_f (1-e^{-t/\tau}) \quad (2.4)$$

En donde q es la carga que va adquiriendo el capacitor, Q_f es el valor final que el capacitor obtiene, en este caso Q_f puede ser obtenido mediante el producto de la capacitancia por el voltaje de la fuente.

Ahora vamos a suponer que el capacitor ya tiene una carga inicial Q_0 , y además se retira la fuente del circuito. El valor de la carga eléctrica del capacitor será:

$$q=Q_0 e^{-t/\tau} \quad (2.5)$$

De las ecuaciones 2.4 y 2.5, haciendo su respectivo despeje, será posible obtener el valor de la carga y descarga del capacitor [36].

2.7.3 Sistemas anti rebotes para pulsos digitales.

Los sistemas anti rebotes es un tema de mucha importancia, el cual no presentar fallos u errores en las señales. Existen diferentes tipos constituidos básicamente de un circuito resistencia-capacitor, o también usando uso de histéresis como lo es el disparador de Schmitt usado como comparador. De esta manera se filtra un solo pulso y no varios, como sería sin ningún circuito anti rebote [37].

Su configuración se la realiza con el embebido HEX40106, haciendo uso de un capacitor, una resistencia y un negador a la salida, y de esta manera conseguir la lógica requerida. La aplicación de este tipo de comparador son las siguientes:

- Convertidor analógico a digital.
- Interruptor de sobrecarga a tierra.
- Generador de onda cuadrada.
- Comparador de señal

En la Figura 2.23. Muestra el diagrama de un circuito anti-rebote para pulsos digitales

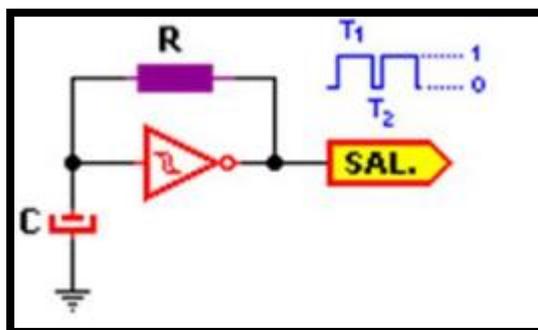


Figura 2.20 Diagrama de un Sistema Anti-rebote, resistencia, capacitor [37].

2.7.4 Reguladores Lineales.

Los reguladores de voltaje o tensión son dispositivos electrónicos capaces de nivelar un voltaje de salida constante, usualmente se los usa en computadores o en proyectos en los cuales es necesario un voltaje invariable. Esta básicamente construido por transistores bjt, resistencias y diodo zenner que estabilizan su salida.

Vienen implementados por medio de un dispositivo integrado y forman parte de la familia de los 78XX y 79XX [38]. La familia 78 entrega voltajes positivos en comparación con la familia de los 79, el valor de las XX representan los voltajes que estos pueden otorgar, su configuración típica se lo puede encontrar en la Figura 2.22 en la que se muestra que a la entrada de voltaje se conecta un capacitor de 0.33 uF que permite eliminar los efectos producidos por los dispositivos inductores tales como son las bobinas, y en la salida de voltaje se conecta un capacitor de 0.1 uF para garantizar una mejor estabilidad en el circuito frente a los fenómenos transitorios

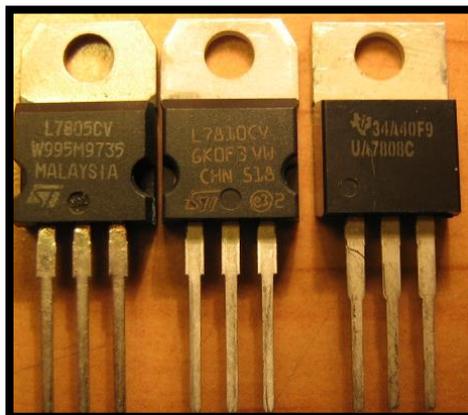


Figura 2.21 Esquema de un regulador de voltaje 78XX lineal [38].

2.7.5 Reguladores Conmutados.

Los reguladores conmutados trabajan por medio de la conmutación que se produce en un transistor, es decir, el transistor algunas veces opera en forma de corte y en otros casos opera en forma saturada [39]. Cuando se tiene este tipo de configuraciones se aporta con un decremento de consumo de potencia. En la Figura 2.23 podemos observar un esquema de una fuente de voltaje conmutada la cual hace uso de este tipo de reguladores.

Para un regulador básico es posible utilizar el uA723 ya que su configuración es muy sencilla y puede operar de varias formas, entre ellas están:

- Como un dispositivo de baja tensión ($V_o=2.0$ hasta 2.7 voltios).
- Como un regulador de alta tensión ($V_o=7.0$ hasta los 7.37 voltios).
- Como regulador con protección fold-back.
- Como regulador conmutado (switching).

Un circuito conmutado normalmente recibe en sus entradas una señal de tipo triangular generados por un oscilador y un voltaje de referencia, cuando los niveles de tensión sobrepasan el valor de referencia, existe

un nivel alto en la salida del comparador, y en caso de ser bajo, existirá un nivel bajo en la salida del comparado.



Figura 2.22 Esquema de un regulador de voltaje uA723 como fuente conmutada [40].

2.8 Sistemas RFID.

Un sistema RFID es capaz de almacenar y recuperar datos, grabados en etiquetas, por lo general algún número de serie único mediante el uso de radio frecuencia [41].

Los sistemas RFID se pueden clasificar específicamente dependiendo del tipo de frecuencia a la que trabajan, siendo frecuencia baja, alta, UHF, VHF y de microondas. En la Figura 2.24 se puede observar los rangos de trabajo de las tarjetas RFID.

Banda de Frecuencias	Descripción	Rango
125 kHz - 134 kHz	LF (Baja Frecuencia)	Hasta 45 cm.
13,553 MHz - 13,567 MHz	HF (Alta Frecuencia)	De 1 a 3 m.
400 MHz - 1.000 MHz	UHF (Ultra Alta Frecuencia)	De 3 a 10 m.
2,45 GHz - 5,4 GHz	Microondas	Más de 10 m.

Figura 2.23 Rangos de trabajo de la tarjeta RFID [41].

2.8.1 Tarjetas y lectores RFID.

Existen tanto lectores como tarjetas, donde las tarjetas son casi siempre pasivas, lo que significa que requieren de excitación para poder responder con su número guardado. Son usadas en múltiples propósitos gracias a su pequeña forma y versátil uso.

Este tipo de etiquetas pasivas tiende a un rango mínimo de alcance de 10 cm hasta unos pocos metros, y según su frecuencia de trabajo, diseño y tamaño de la antena. Hoy en día hay maquinas que las imprimen por millares y en diferentes formas y tamaños.

En contraste con las pasivas se encuentran las activas las cuales tienen su propia fuente de energía, para de esta manera tener más alcance y capacidad de memoria mucho más grande que las pasivas. Siendo propensas a un menor porcentaje de error gracias a que generan su propia señal y no trabajan en respuesta a un impulso. Pueden llegar hasta muchos cientos de metros y tienen vida útil de hasta diez años.

En la actualidad pueden llegar a medir hasta el tamaño de una moneda, rango de hasta 10 metros, con ese tamaño, y varios años de vida útil de la batería [42]. En la Figura 2.25 se puede observar un pequeño tag o etiqueta de RFID.

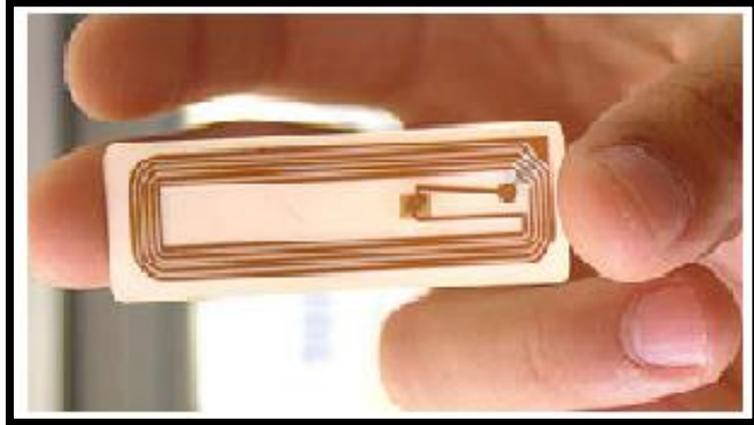


Figura 2.24 Etiqueta RFID [42].

Entre los beneficios que nos brinda este tipo de sistemas podemos mencionar lo siguiente:

- Inventariado de objetos de una manera fácil y sencilla
- Se puede combinar con el internet
- Provee identificación de personas y de objetos en todo momento al pasar por un lector.

Entre otros su uso es extenso teniendo muchas áreas de cobertura en el desarrollo de la vida actual.

CAPÍTULO 3

3. DISEÑO E IMPLEMENTACIÓN.

En la siguiente sección se procederá al diseño e implementación del prototipo con servicio de geoposicionamiento para buses internos y externos, así como también el prototipo para el paradero inteligente, se explica el por qué se escogieron los elementos, la lógica de programación mediante el uso de diagramas de flujo; y también sus conexiones físicas.

3.1 Diseño.

Previo a la implementación del prototipo se realiza un diagrama con todos los elementos que son más fáciles de obtener en el mercado, para su correspondiente análisis de factibilidad y viabilidad.

En la figura 3.1 podemos observar la variedad de elementos disponible en el mercado para GPS, Arduino y demás. Con ello se analizar cada elemento que se podrá escoger para los diferentes módulos del proyecto.

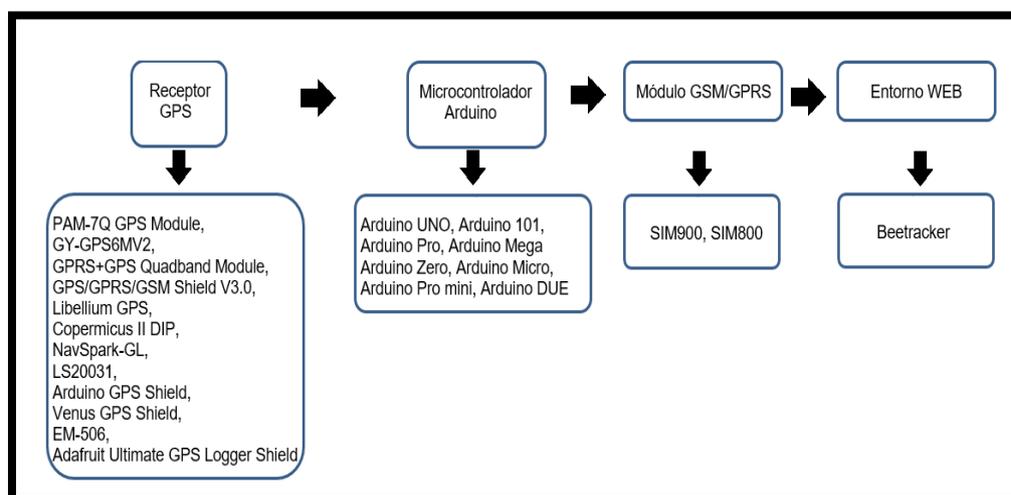


Figura 3.1. Diagrama de posibles elementos a utilizar en el prototipo.

Se contemplan los diferentes tipos de receptores GPS en el mercado, los cuales se enlistan en la Tabla1.

Receptor GPS	Precisión	Antena	Chip	Precio Us
PAM-7Q GPS Module	menor a 2m	SI	u-blox MAX-7	\$ 50.00
GY-GPS6MV2	menor a 2.5m	SI	u-blox 6M	\$ 17.68
GPRS+GPS Quadband Module	menor a 2.5m	SI	SIM 908	\$ 62.99
GPS/GPRS/GSM Shield V3.0	menor a 2.5m	SI	SIM 908	\$ 130.00
Libellium GPS	menor a 2.5m	NO	Vincotech A 1080-B	\$ 47.00
Copermicus II DIP	menor a 2.5m	SI	Trimble Trim Core	\$ 74.95
NavSpark-GL	menor a 2.5m	SI	Venus 8	\$ 25.00
LS20031	menor a 2.5m	SI	MT3339 MTK	\$ 59.95
Arduino GPS Shield	menor a 2.5m	NO	Venus	\$ 39.99
Venus GPS Shield	menor a 2.5m	SI	Venus638FLPx	\$ 49.95
EM-506	menor a 2.5m	SI	SIRF Star IV	\$ 39.95
Adafruit Ultimate GPS Logger Shield	menor a 3m	SI	MT3339	\$ 44.95

Tabla 1. GPS Disponibles en el mercado con algunas características.

El módulo embebido que se escogerá es el GY-GPS6MV2 con una precisión menor a 2.5 m, posee una antena y es uno de los más económicos; comúnmente conocido en el mercado como GPS Neo 6M. Este módulo fue desarrollado por UBLOX [43], el cual ofrece un receptor GPS práctico y fácil de ensamblar, con la garantía de tener un dispositivo de alto rendimiento, para dispositivos móviles con muy estrictas limitaciones de coste y espacio.

Posee en la parte superior 4 pines sueltos que representan las conexiones de fuente, tierra, y la comunicación serial TX y RX respectivamente y en la parte

inferior un conector para la antena. La entrada de voltaje es capaz de soportar un rango que va desde los 3.3 voltios hasta los 5 voltios en corriente directa.

Las características de este módulo son las siguientes:

- Velocidades de 9600/38400/4800/57600 baudios.
- Opera en modo máximo de consumo y en bajo nivel de consumo.
- Antena de gran potencia con ganancia de 50 dB.
- Soporta temperaturas de -40° C hasta los 85° C.
- Sensibilidad de recepción de -161 dBm.
- Un rango de tiempo de pulso que va desde los 0.25 Hz hasta los 10 MHz.
- Canalización de 50 canales y sistema de recepción GSP en frecuencias de L1.

En la Figura 3.2 se muestra un modelo de este receptor GPS, en la cual es posible notar los pines de comunicación serial TX y RX, junto con los pines de voltaje y tierra; y, en el otro lado tenemos un pequeño cable junto a la antena que permite la recepción de la frecuencia L1 de un sistema GPS satelital.

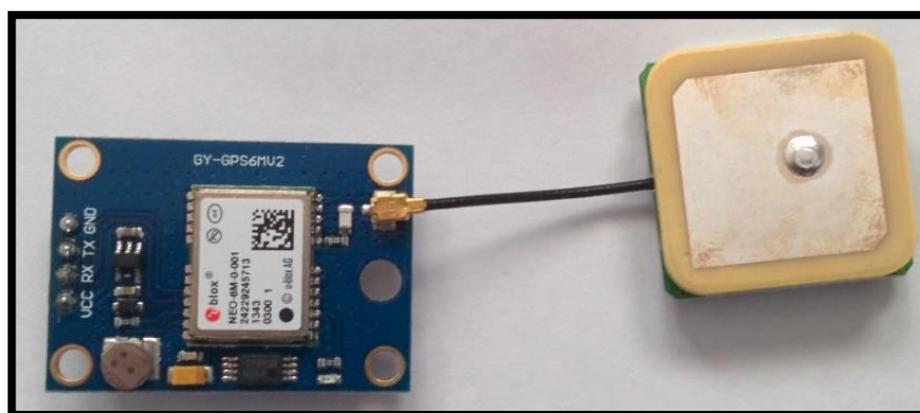


Figura 3.2. Placa GPRS NEO 6M de la empresa UBLOX [43].

En la Tabla 2 se muestran las comparaciones entre los diferentes modelos de Arduino, los cuales son de fácil programación. El criterio para escoger el microcontrolador correcto fue basado en el número de pines de interrupción,

así como también en el número de puertos UART para comunicación entre los módulos GPS y GPRS.

Nombre	Pines de Interrupción	Digital IO/PWM	UART	Precio
101	14	14/4	-	\$ 37.74
Mega 2560	6	54/15	4	\$ 16.90
Micro	5	20/7	1	\$ 10.00
Pro	2	14/6	1	\$ 46.95
Pro Mini	2	14/6	1	\$ 5.49
Uno	2	14/6	1	\$ 13.50
Zero	13	14/10	2	\$ 65.00
Due	54	54/12	4	\$ 39.07

Tabla 2. Comparaciones entre los modelos de Arduino.

Con estas premisas se debe escoger un Arduino Mega 2560, el cual cumple con nuestro requerimiento de pines para interrupción, 3 pines de 6; y, para la comunicación tenemos en uso 2 pares de los 4 pares que posee el Arduino Mega 2560.

La Tabla 3. Muestra las características de los módulos de comunicación con la red Celular para la conexión a la interfaz GSM/GPRS.

Nombre	Bandas de Operación	SMS	Bluetooth	Precio
SIM900	Quad-Band 850/900/1800/1900 MHz	si	no	\$ 26.99
SIM800	Quad-band 850/900/1800/1900M Hz	si	si	\$ 43.99

Tabla 3. Módulos de comunicación con la red celular.

Como se puede observar los dos módulos son muy parecidos, con respecto a su modo de uso, los voltajes y temperatura de operación. Para este proyecto no será necesario el uso de Bluetooth por lo que la mejor opción es el SIM900.

Para el paradero inteligente, se debe usar un sistema el cual se acople a las tarjetas inteligentes. En la Tabla 4 se puede observar los tipos de frecuencia en las que trabajan estas tarjetas.

Frecuencia	Rango de Alcance	Uso	Tipo
< 135 kHz	< 10cm	Control de acceso	pasivo
13.56 MHz	< 1m	Tarjetas Inteligentes, Control de acceso	pasivo

Tabla 4. Frecuencias de tarjetas RFID

Debido a la frecuencia a la cual trabajan los carnets de los estudiantes es de 13.56 MHz, para ello se usará un módulo económico llamado MFRC522, el cual trabaja en esta frecuencia, con un costo de \$13.00 dando la facilidad de lectura y escritura de tarjetas.

3.2 Diagramas de flujo.

Una vez escogidos los elementos a utilizar y previo a las conexiones físicas, se diseña la codificación respectiva para cada módulo, por lo que es necesario tener bien en claro los diagramas de flujo para el desarrollo del código en c++ dentro de la interfaz Arduino IDE.

3.2.1 Diagrama módulo buses internos.

Para poder obtener un resultado óptimo poniendo las ideas en orden se obtiene el diagrama de flujo mostrado en la Figura 3.3 donde: Primero establecemos las variables y encendemos el modulo responsable de la comunicación con la red, posterior a eso preguntamos por la señal GPS, y preguntamos por los estados de los

botones de una manera asíncrona. Y al finalizar enviará la trama con longitud, latitud y estado de alarma.

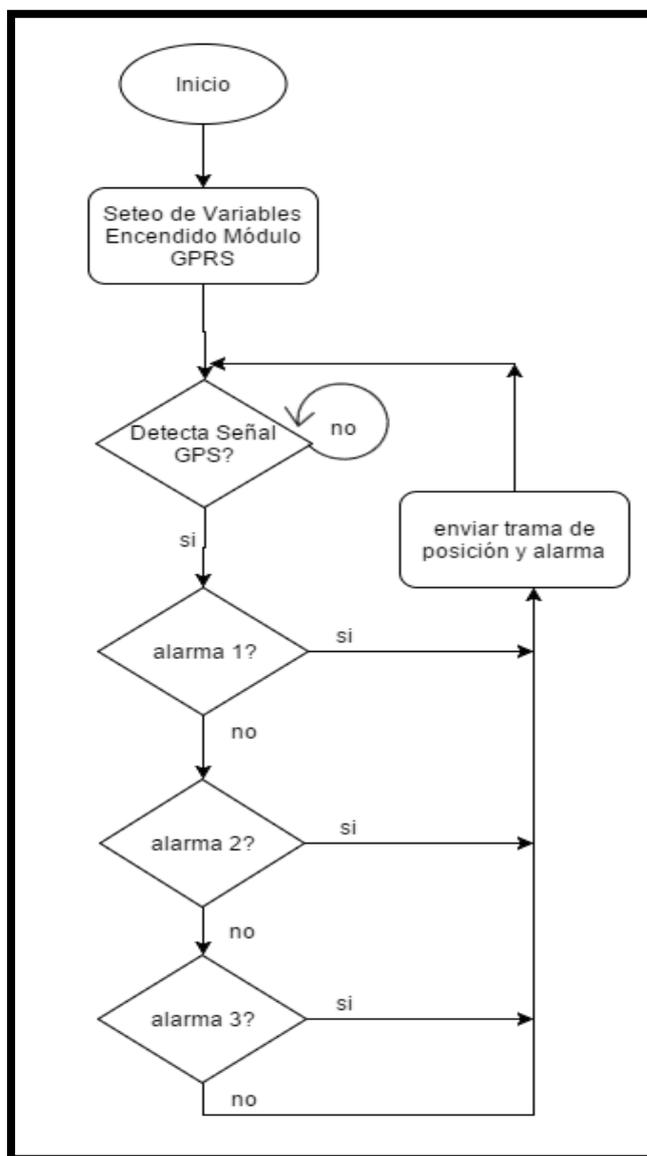


Figura 3.3. Diagrama de flujo bus interno

3.2.2 Diagrama módulo buses externos.

Para el segundo módulo se incluyó los geocercos, los cuales al estar dentro de un bus externo cambiaría el número de celular, dependiendo en que geo cerco se encuentre. Con el objetivo de que envíen un

mensaje al UPC más cercano si alguna alarma es activada, además de la transmisión de la trama al servidor cada intervalo de tiempo. En la Figura 3.4 se puede observar el diagrama de flujo para los buses externos.

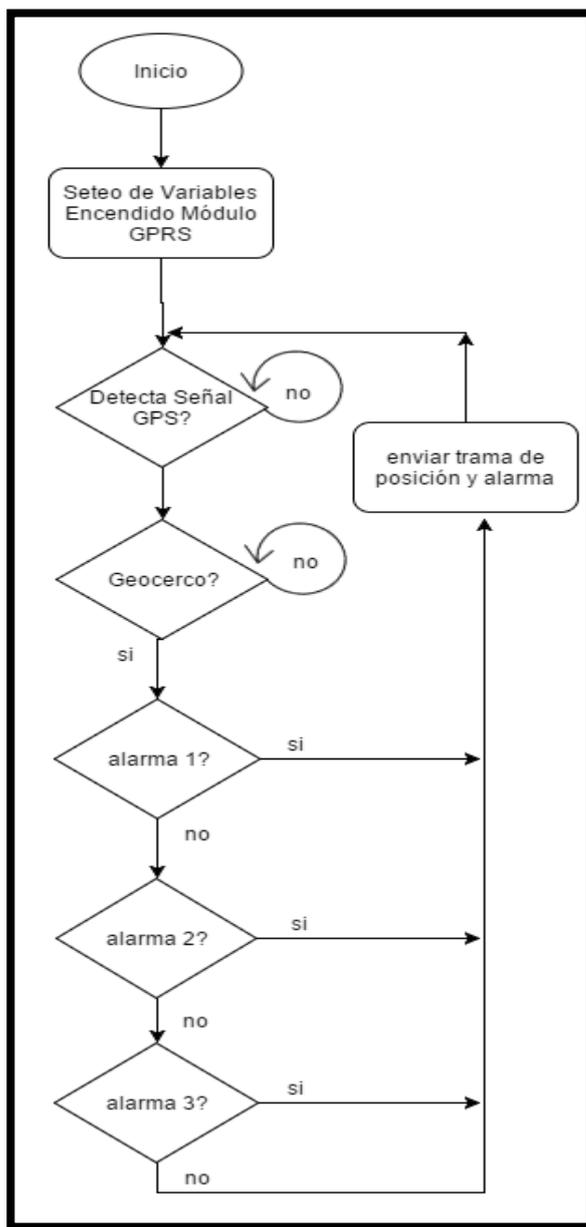


Figura 3.4. Diagrama de flujo bus externo

3.2.3 Diagrama del paradero inteligente.

Para el paradero inteligente primero se establecen las variables responsables del conteo de carnets y de los pines de comunicación. Posterior a eso, en el ciclo, espera si es que detecta un carnet, una vez que ha detectado revisa una matriz para verificar que no se encuentren repetidos. Si llega a los 50 carnets enviaría un mensaje a las autoridades competentes para que manden otra unidad, caso contrario revisa el estado de las alarmas, si ninguna está activada continúa su funcionamiento normal. Todo esto puede ser observado en la Figura 3.5.

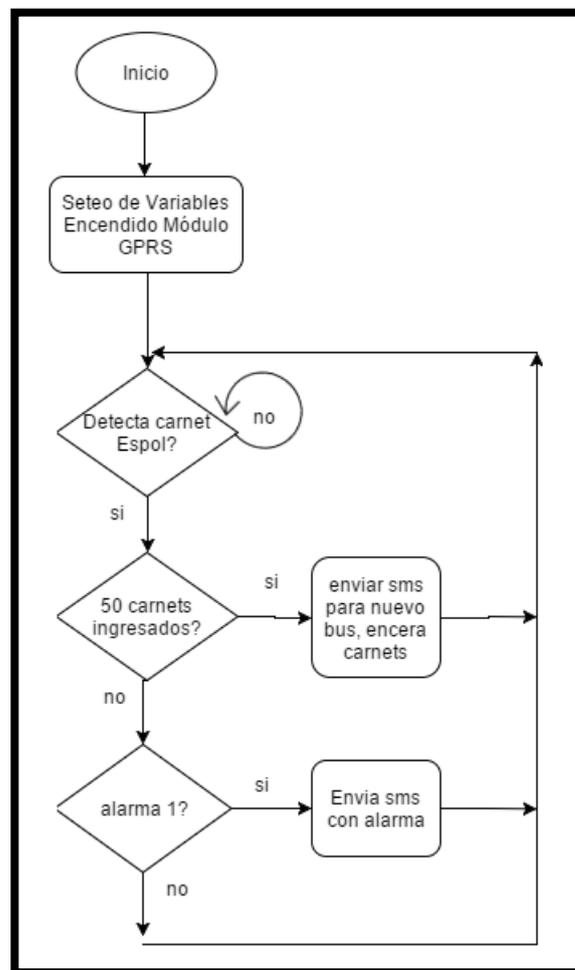


Figura 3.5. Diagrama de parada inteligente

3.3 Diseño de códigos.

En la siguiente sección se describirán los códigos a utilizar, los cuales se encuentran en los anexos.

3.3.1 Diseño de código para buses internos.

Para el diseño del primer módulo es necesario tener en cuenta que todo microcontrolador debe tener una subrutina de configuración y otra de ciclo infinito.

Se incluye la librería "TinyGPS.h" la cual se encargará de crear / manipular objetos y manejar funciones estrictamente de la cadena que será recibida de manera serial por el módulo GPS.

Posterior a eso deberán declararse ciertas variables, las mismas que serán usadas por otras funciones dentro del código.

En la subrutina de configuración se inicia la comunicación serial con el módulo de GPS y el módulo GPRS mediante la función: `Serial.begin(x)`; donde `x` es la velocidad a la cual se comunican estos dispositivos en baudios. También se declaran las variables de carácter asíncrono, como son las interrupciones, por medio de la función `attachInterrupt (alarma1, Interr0, RISING)`; esto es debido a que las alarmas de socorro pueden ocurrir en cualquier momento; y, es necesario que una interrupción pueda levantar una bandera en cualquier momento sin que ésta interrumpa al código principal. También se procede a encender el módulo SIM 900 para que se encuentre conectado a la red, por medio de la función `SIM900POWER`.

La subrutina de lazo espera detectar señal del GPS, para luego decodificarla y de esta manera obtener latitud y longitud. Posteriormente será enviada al servidor encapsulada en una trama cada 5 segundos. La trama es la siguiente:

```
enjambre.ec/bt/point/Id/Latitud/Longitud/Alarma/
```

Donde:

Este código se encuentra en el Anexo 1,

Id: es el identificador del vehículo,

Latitud: es la latitud en grados del vehículo,

Longitud: es la longitud en grados del vehículo.

Alarmas: es el estado de las alarmas el 0 corresponderá a ninguna alarma, 1 para alarma de robo dentro del vehículo, 2 para alarma médica y 3 para alarma de incendio en el punto de envió.

3.3.2 Diseño de código para buses externos.

Para el diseño del código que se utilizará en los buses externos es necesario crear una función; la cual, escogerá los números de celular de la persona encargada del UPC, dentro de la zona que marca el geo cerco donde se encuentre circulando el vehículo.

Haciendo uso de Google Maps se crearán los geocercos. Primero se establecerá un punto sobre el UPC y luego otro punto el cual se encuentre en la ruta del bus. De esta manera se establecerá el Radio en grados.

Para crear los diferentes puntos de los geocercos es necesario basarse en la ecuación del círculo, cambiando el centro y definiendo el radio.

$$(X-A)^2 - (Y -B)^2= R^2 \quad (2.6)$$

Donde:

X: es la latitud obtenida por el GPS,

Y: es la longitud obtenida por el GPS,

A: el centro de latitud del punto,

B: el centro de longitud del punto,

R: el radio medido en grados desde el punto.

Se procede a crear la función smsGeofence () la cual se encarga de hacer el cambio de números según el geocercos, donde se encuentre el

vehículo. También se ha considerado el caso por defecto si es que el vehículo no se encuentra en ningún geo cerco. Este código se encuentra en el Anexo 2.

3.3.3 Diseño de código para paradero inteligente.

Para el diseño del código del paradero inteligente se debe dividir en dos módulos: el módulo uno, que es el responsable del conteo de tarjetas RFID mediante su módulo lector; y, el modulo dos, responsable de la comunicación GSM, las señales lumínicas y la pantalla LCD.

Dentro del módulo uno se hará uso de las librerías “AddicoreRFID.h” y “SPI.h”, de esta manera se configurará para que pueda leer el carnet de los estudiantes. El tipo de comunicación que se utiliza es SPI, el cual se usa para la transferencia de información entre el lector RFID y el Arduino UNO, este método de comunicación ya fue explicado en el capítulo 2 en la sección 2.6.3. Antes de ser enviada la información al módulo dos, previamente revisará un arreglo, creado e inicializado en la subrutina de configuración, en el cual se deberán ir guardando la información de los carnets. En caso de no estar repetido se guardará en el arreglo y se enviará una señal de alto para que indique en la pantalla LCD que se está leyendo una tarjeta. Este código se encuentra en el anexo 3.

En el módulo dos se usará la librería “LiquidCrystal_I2C.h” para poder utilizar la pantalla LCD de 16x2 mediante solo dos pines usando la comunicación I2C. Esta comunicación I2C varía dependiendo del Arduino que se va a utilizar ya que los pines SDA y SCL también varían, este método de comunicación ya fue explicado en el capítulo 2 en la sección 2.6.2 Este código se encuentra en el anexo 4.

La pantalla LCD será responsable de mostrar los mensajes de “Ocupado”, “Paradero Espol”, “personas” y su respectivo contador. Paralelo a la revisión de carnets se hará uso de la función digitalWrite (pin,modo); para que el led verde indique que es posible pasar la tarjeta,

mientras que el led rojo indicará que el módulo se encuentra ocupado, esto significa que está enviando un mensaje. Una vez que dentro del código se encuentren un número determinado de personas procederá a encender el módulo GSM y enviará un mensaje según el caso; el cual puede ser por paradero lleno, o por botón de alarma.

3.4 Implementación de los Sistemas

La siguiente sección se trata de las conexiones físicas de los diferentes módulos prototipos a utilizarse.

3.4.1 Implementación para buses internos y externos.

Como se explicó en la sección 3.1 se hará uso del Arduino Mega debido a que este maneja cuatro puertos seriales independientes; y, también por que el SIM900 y el módulo GPS requieren una comunicación serial independiente para que no exista conflicto en la comunicación. Siendo este un problema grande de sincronización si fuesen puertos virtuales y no físicos al momento de comunicar los diferentes módulos.

Además, para contar con un sistema de alarmas independientes se deben usar interrupciones para enviar los diferentes tipos de señales en la trama de datos enviada al servidor sin que éstas interrumpen el funcionamiento del módulo. El Arduino Mega consta con cuatro interrupciones de las solo cual usaremos tres, haciendo uso de flancos de bajada o de subida independientemente, el cual cambiará una bandera, responsable del tipo de alarma. En la Figura 3.6 se puede observar la unión de los pines del Arduino Mega con el módulo SIM900.

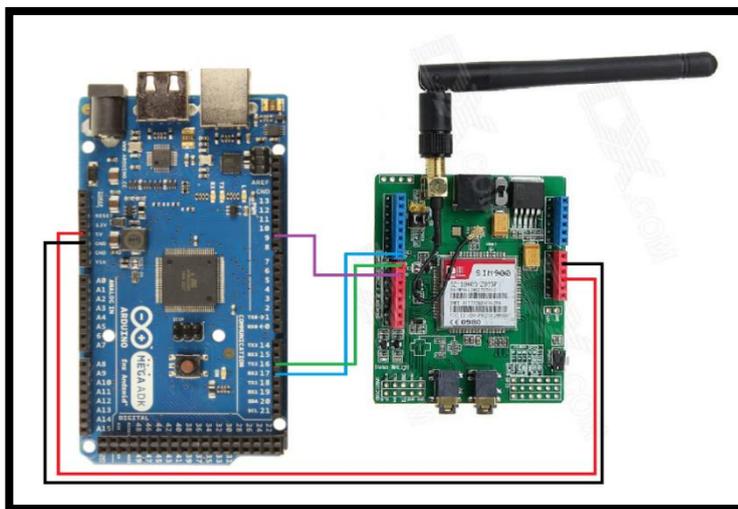


Figura 3.6 Conexión de los pines del Arduino Mega con el módulo SIM900.

En la Tabla 5 se muestra los pines que se están conectando entre el Arduino y el módulo SIM900.

Arduino	SIM900
5v	5v
Gnd	Gnd
Pin 9	Pin 9
Pin 16	Pin 8
Pin 17	Pin 7

Tabla 5. Pines de conexión del Arduino Mega con el módulo SIM900.

Una vez probado que la comunicación entre el Arduino y SIM900 funciona de manera correcta se procede a adicionar el modulo GPS. En la Figura 3.7 se puede observar la conexión de los pines, entre el Arduino Mega, el módulo SIM900 y el módulo GPS.

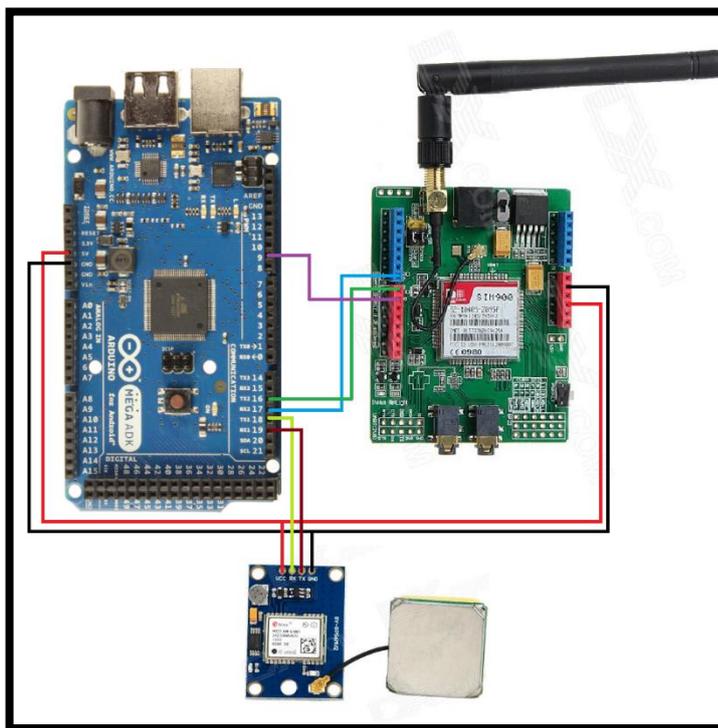


Figura 3.7 Conexión del Arduino Mega con los módulos SIM900 y GPS.

En la Tabla 6 se muestra los respectivos pines de conexión entre el Arduino Mega y el módulo Neo GPS 6M.

Arduino Mega	Neo GPS 6M
5v	Vcc
Gnd	Gnd
Pin 19	tx
Pin 18	rx

Tabla 6. Pines de conexión entre el Arduino Mega y el módulo GPS. Las señales de interrupción obtenidas como resultado de presionar los botones de alarma son altamente sensibles a errores en el medio, es por esto que se implementará un comparador encargado de minimizar en lo posible este tipo de errores que, de no ser corregido, causaría muchos inconvenientes durante las pruebas del prototipo. El sistema que se va

a utilizar es un comparador de histéresis implementado por Otto Herbert Schmitt para prevenir el ruido de una señal original y previniendo que se produzcan errores en los niveles recibidos. Este circuito integrado 40106N es el responsable de lograr esto, accionado por la resistencia y el capacitor. Los valores de la resistencia y el capacitor fueron elegidos en base a la constante de tiempo “ τ ” que se hace al unir estos dos elementos pasivos el cual es:

$$T=RxC=10k\Omega \times 100\mu\text{f}=0.1\text{ms}$$

Para lo cual 0.1 ms es un tiempo que se establece para circuitos antirrebotes. Este comparador procede a realizar conmutaciones en un pin negativo cuando se recibe una entrada que sobrepasa un nivel de referencia previamente establecida. En el capítulo 2 se explicó acerca de anti rebotes para sistemas digitales. En la siguiente Figura 3.8 se ve cómo se conecta una entrada de interrupción por hardware.

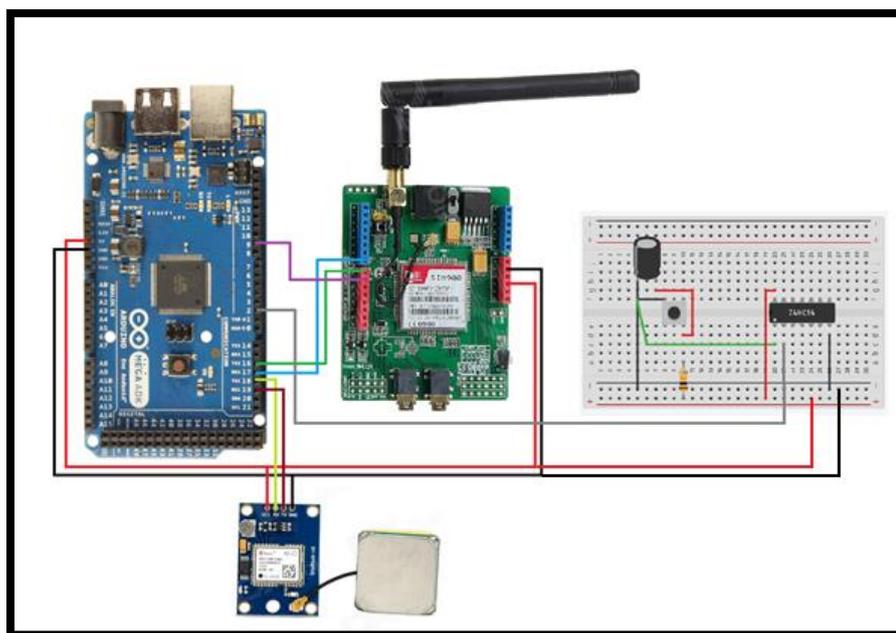


Figura 3.8. Conexión a una entrada de interrupción.

Para poder conectar el prototipo a la fuente de energía del bus, debido a que el bus utiliza dos baterías de 12 voltios en serie, siendo esto una

señal de voltaje no adecuada para alimentar el prototipo, es necesario hacer uso de un regulador de voltaje que limite la cantidad de voltaje recibido y a su vez lo regule de tal manera que evite posibles picos de voltaje que estropeen los dispositivos embebidos. Siendo este un regulador conmutado, comúnmente encontrado en los cargadores de los celulares de los carros.

3.4.2 Implementación para paradero Inteligente.

En primer lugar, se procede a conectar el lector RFID como se puede observar en la Figura 3.9.

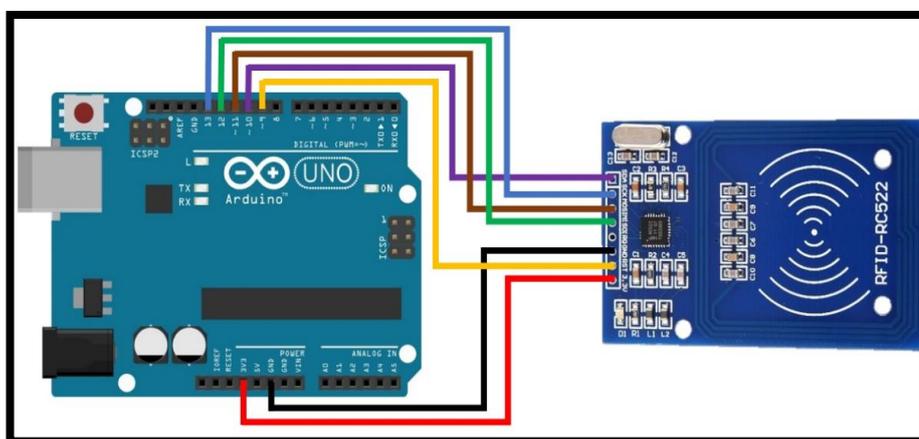


Figura 3.9. Conexión Arduino UNO y lector RFID.

Este lector será el encargado de extraer la información contenida en cada tarjeta y almacenar el primer campo obtenido, en un arreglo de 50 personas, el cual al llegar a ese número enviará una señal al otro módulo, para que envíe un mensaje solicitando otro bus. En la Figura 3.10 se puede observar cómo se conectan el módulo Arduino UNO con un SIM900 mediante la simulación de puertos seriales en los pines 7 y 8.

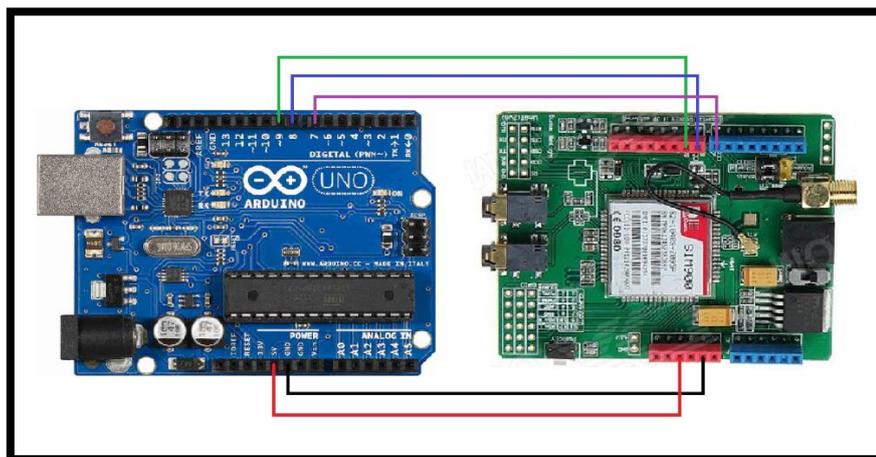


Figura 3.10. Conexión entre el Arduino UNO y el SIM900 usando comunicación por software.

La Tabla 7 muestra los pines de conexión entre el Arduino Uno y el RFID.

Arduino UNO	RFID
3.3v	Vcc
Gnd	Gnd
Pin 10	SDA
Pin 13	SCL
Pin 11	MOSI
Pin 12	MISO
N/A	IRQ
Pin 9	RST

Tabla 7. Pines de conexión entre el Arduino UNO y el lector RFID.

La Tabla 8 muestra los pines de conexión entre el Arduino Uno y el SIM900.

Arduino UNO	SIM900
5v	5v
Gnd	Gnd
Pin 7	Pin 7
Pin 8	Pin 8
Pin 9	Pin 9

Tabla 8. Pines Arduino UNO y módulo SIM900.

La Figura 3.11 muestra cómo se conecta la pantalla LCD 16x2 mediante un módulo I2C, el cual nos facilita la programación y hace uso de sólo 2 pines analógicos, en este caso los pines SDA y SCL.

En la Figura 3.12 se observa la conexión entre el Arduino UNO, la pantalla LCD y el SIM900. De lo cual se puede observar que para un Arduino UNO los puertos responsables de la comunicación I2C son los analógicos 4 y 5, no así para otros Arduinos.

En la Tabla 9 se muestra los pines de conexión entre el Arduino UNO y el módulo I2C.

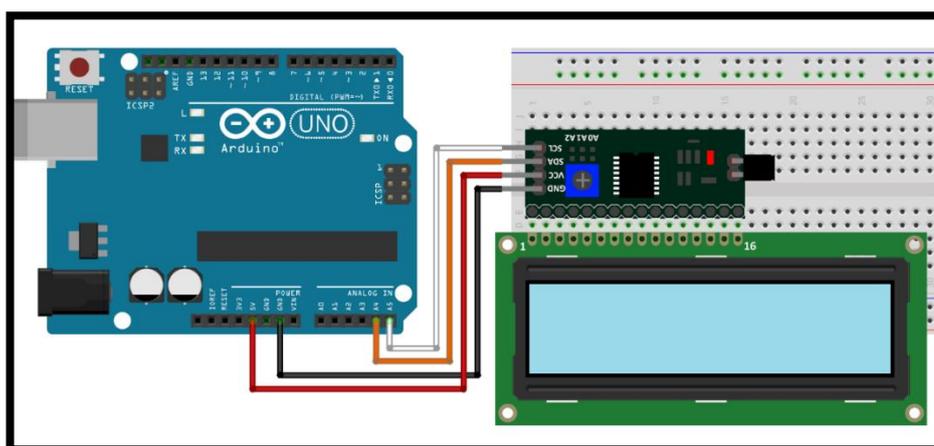


Figura 3.11. Conexión de pantalla LCD mediante el módulo I2C.

Arduino UNO	Módulo I2C
5v	5v
Gnd	Gnd
Analog 4	SDA
Analog 5	SCL

Tabla 9. Pines del Arduino uno y el módulo I2C.

En la Figura 3.13 se puede observar los módulos conectados entre sí para su correcto funcionamiento, los cuales adicionalmente ya tienen conectados los leds, responsables de la respuesta visible para las personas del estado del paradero. Para escoger las resistencias que limitan las corrientes que pasan por los leds es necesario saber la corriente de polarización de cada uno, la cual es 20ma. Siendo 20ma la corriente máxima de salida es necesario disminuirla y sabiendo que el voltaje de salida de cada pin es 5v, hacemos la siguiente ecuación:

$$I=V/R =5/330 = 15\text{ma}$$

Estos 15ma son suficientes para encender el led y no quemarlo, es por eso que usaremos resistencias de 330 ohmios

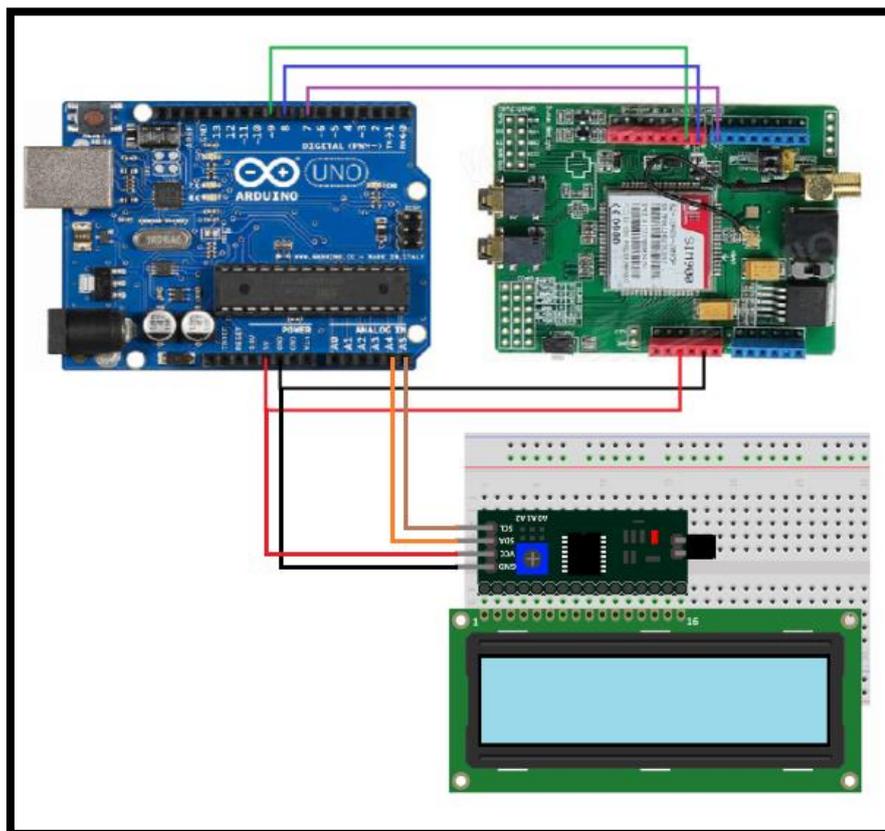


Figura 3.12. Conexión entre Arduino UNO, pantalla LCD y SIM900.

3.5 Diseño de las placas.

Para poder pasar del diseño en protoboard a algo más profesional es necesario hacer un diseño de placas las cuales serán enviadas a imprimir y después se soldarán todos los componentes. En la Figura 3.14 se puede apreciar el diseño en PCB de la placa que se implementará en los buses internos y externos de la ESPOL. El módulo SIM900, el Arduino Mega y las borneras para los botones; estarán alineados de izquierda a derecha respectivamente en la placa.

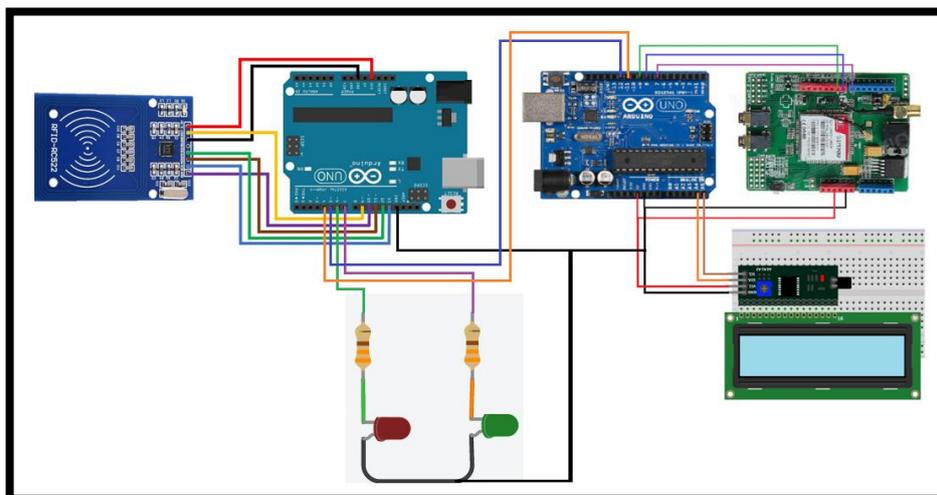


Figura 3.13. Módulo del paradero inteligente.

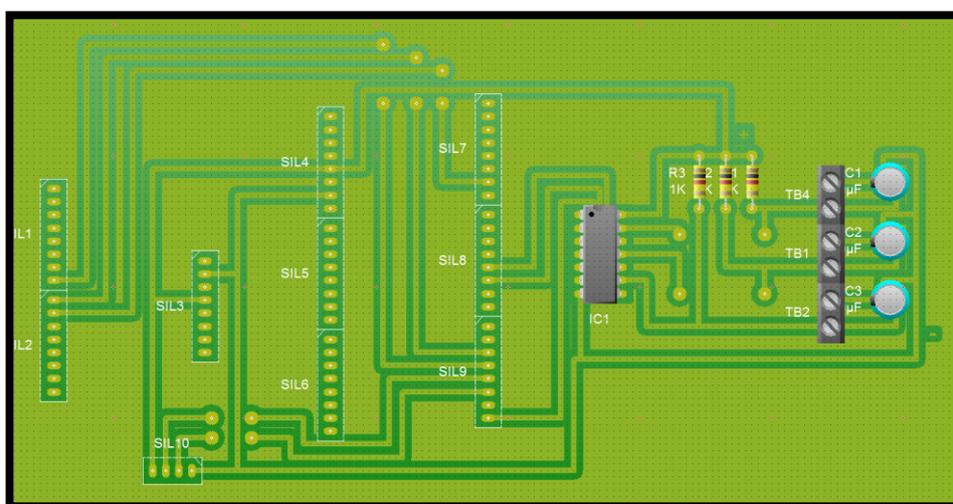


Figura 3.14. Diseño de placa para los buses internos y externos.

En la Figura 3.15 se puede apreciar el diseño en PCB de la placa que se implementará en el paradero inteligente en la garita de la ESPOL, donde estarán: el módulo Arduino UNO con el SIM900 montado encima, el Arduino UNO conectado a la pantalla LCD con el módulo I2C (SIL1) y las borneras para el botón de alarma; alineados de izquierda a derecha respectivamente en la placa.

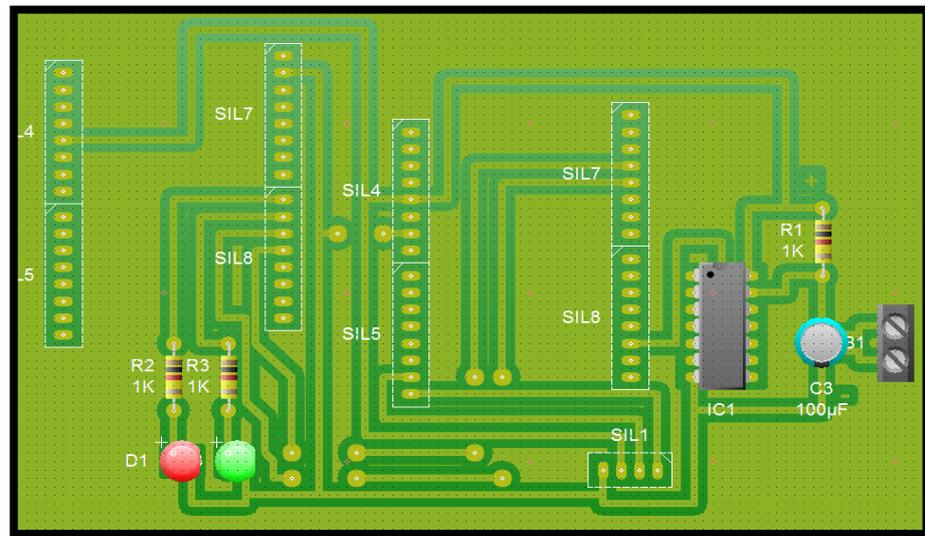


Figura 3.15. Diseño de placa para el Paradero Inteligente

CAPÍTULO 4

4. ANÁLISIS Y RESULTADOS.

En este capítulo se podrán observar los resultados de las pruebas realizadas con el uso de un vehículo particular, así también se presentará el software responsable de la presentación en tiempo real de los módulos. Al final se pondrán una lista de precios para ser tomada en consideración por las autoridades competentes si desearan realizar el proyecto.

4.1 Software BeeTracker.

BeeTracker es un software de presentación gráfica de módulos por medio de telemetría basado en una página web adaptativa, la cual se puede presentar en cualquier dispositivo inteligente, adaptándose a las dimensiones del mismo. Dicho software fue desarrollado por la empresa Enjambre.

En la Figura 4.1 se puede observar la página de ingreso a BeeTracker, donde se introduce el usuario correspondiente y la contraseña.

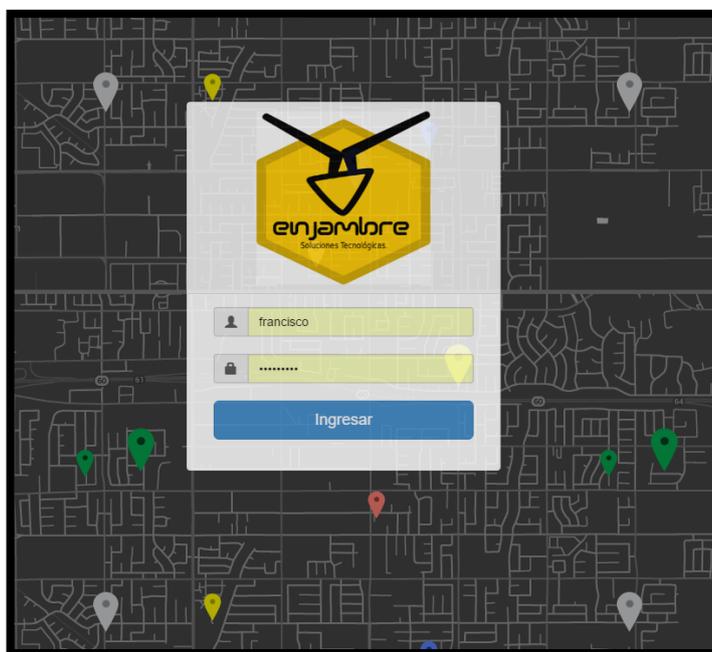


Figura 4.1. Página de ingreso a BeeTracker.

En la Figura 4.2 se puede observar la página principal de Beetracker la cual trabaja con Google Maps de fondo.

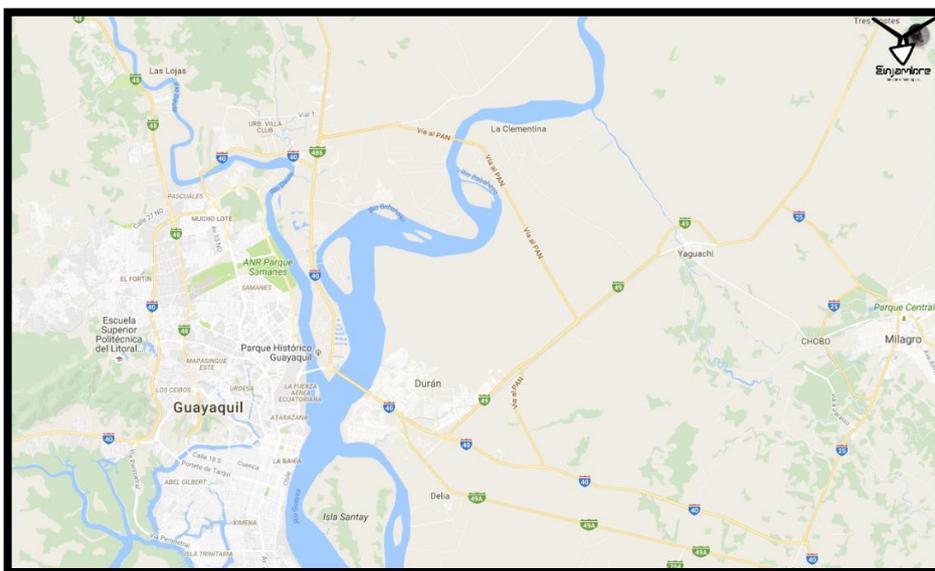


Figura 4.2. Página donde se mostrará el recorrido de los buses de ESPOL.

En la Figura 4.3 se puede visualizar el panel de dispositivos asociados a la cuenta BeeTracker.

4.2 Prueba de Módulo bus interno.

En la Figura 4.4 se observa las conexiones físicas realizadas en un protoboard de los módulos de los buses internos y externos, en la cual se puede apreciar los tres botones que se conectan a la resistencia y al capacitor; pasando al integrado 40106, y las demás conexiones físicas entre el SIM900 y el modulo GPS.

En la Figura 4.5 se pueden observar las conexiones físicas del prototipo en su diseño final para ser instalado y probado. Se ve claramente el Arduino Mega en el fondo, El GPS en la parte superior y el SIM900 en la cara lateral derecha.

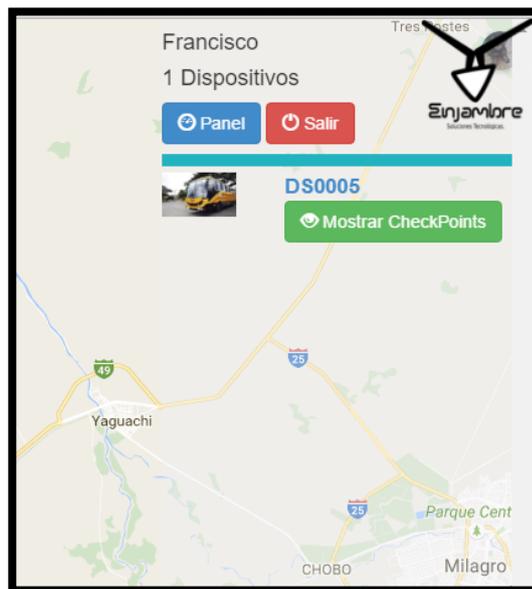


Figura 4.3. Dispositivo asociado a la cuenta BeeTracker.

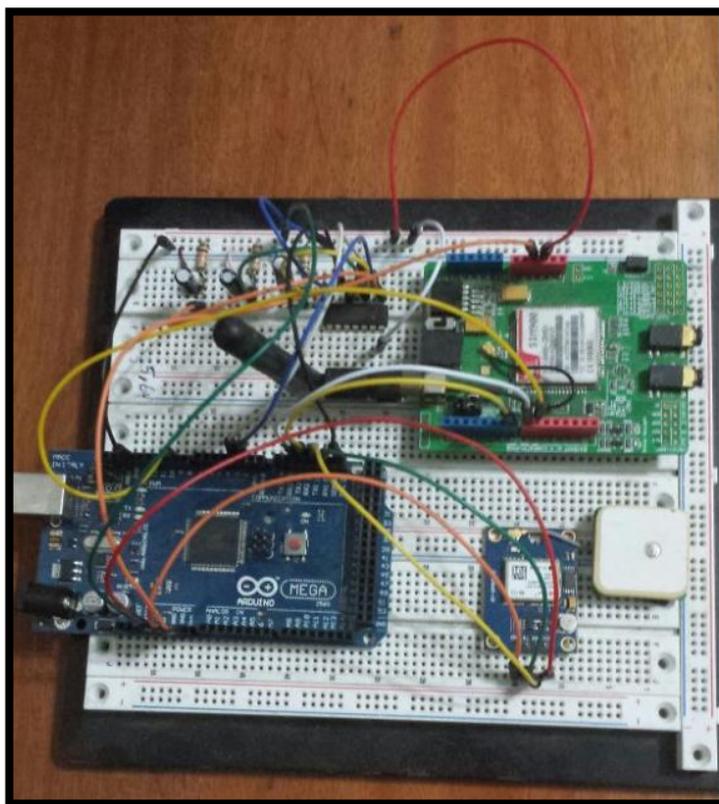


Figura 4.4. Conexión física de los buses internos y externos.

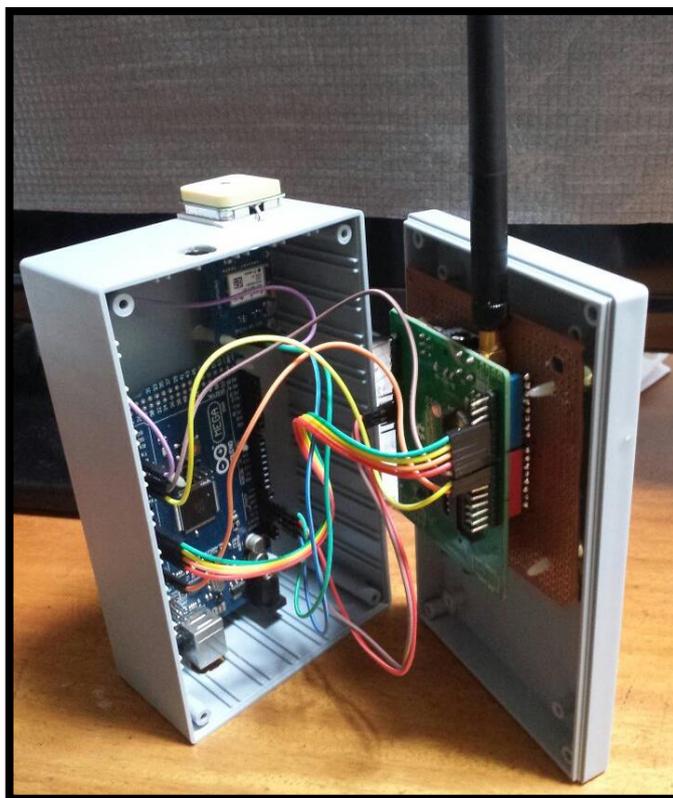


Figura 4.5. Prototipo del módulo de los buses internos y externos.

En la Figura 4.6 se puede observar en el Beetracker el módulo de los buses internos y externos funcionando en tiempo real el cual muestra su nombre o etiqueta en este caso “DS0005”, el número cero es el estado de la alarma, es decir no presenta alarma en el momento, a su vez también muestra los valores de longitud y la latitud respectivamente.

En la Figura 4.7 se observa una de las funcionalidades dentro de Beetracker como lo es su reconstrucción de ruta, el cual nos permite recrear la ruta trazada por el módulo de los buses internos o externos.

En la Figura 4.8 se observa, en imagen satelital, el módulo funcionando en tiempo real, el cual muestra su nombre o etiqueta, en este caso 1 que corresponde al estado de la alarma de Robo, y los siguientes valores corresponden a la longitud y la latitud respectivamente.

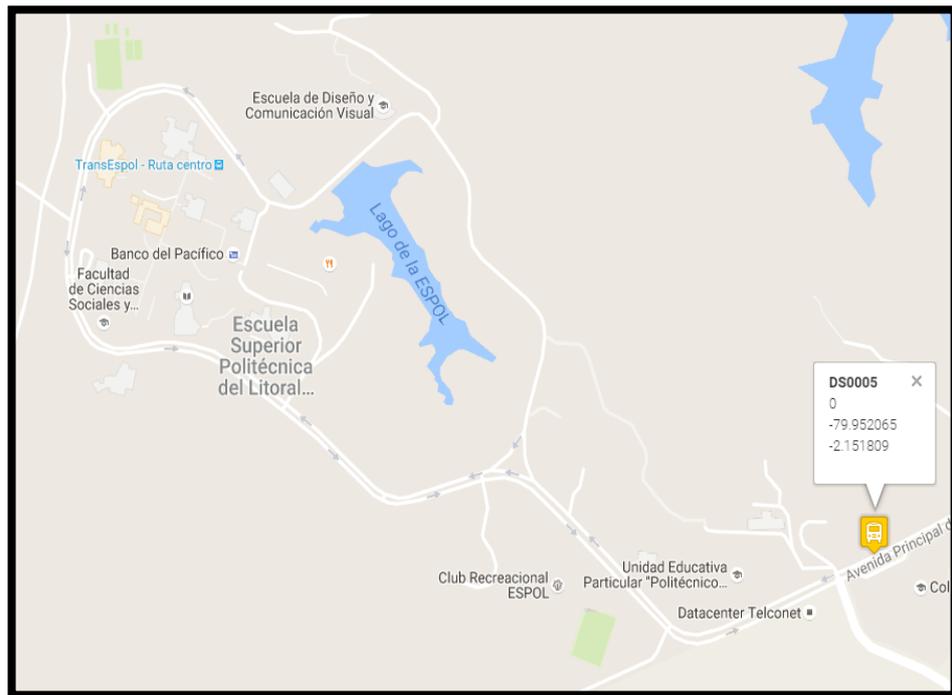


Figura 4.6. Prototipo del módulo en tiempo real.

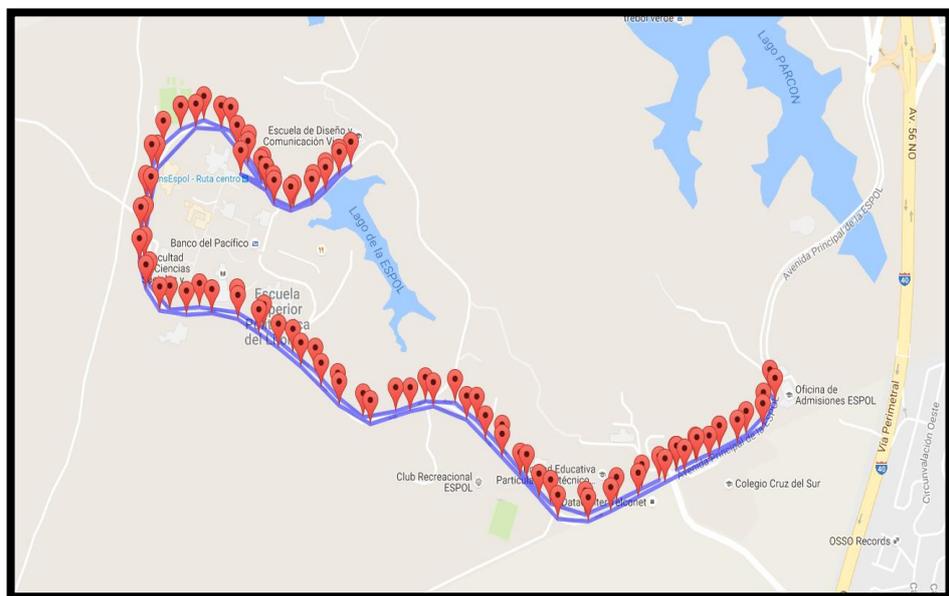


Figura 4.7. Ruta trazada por el bus interno.

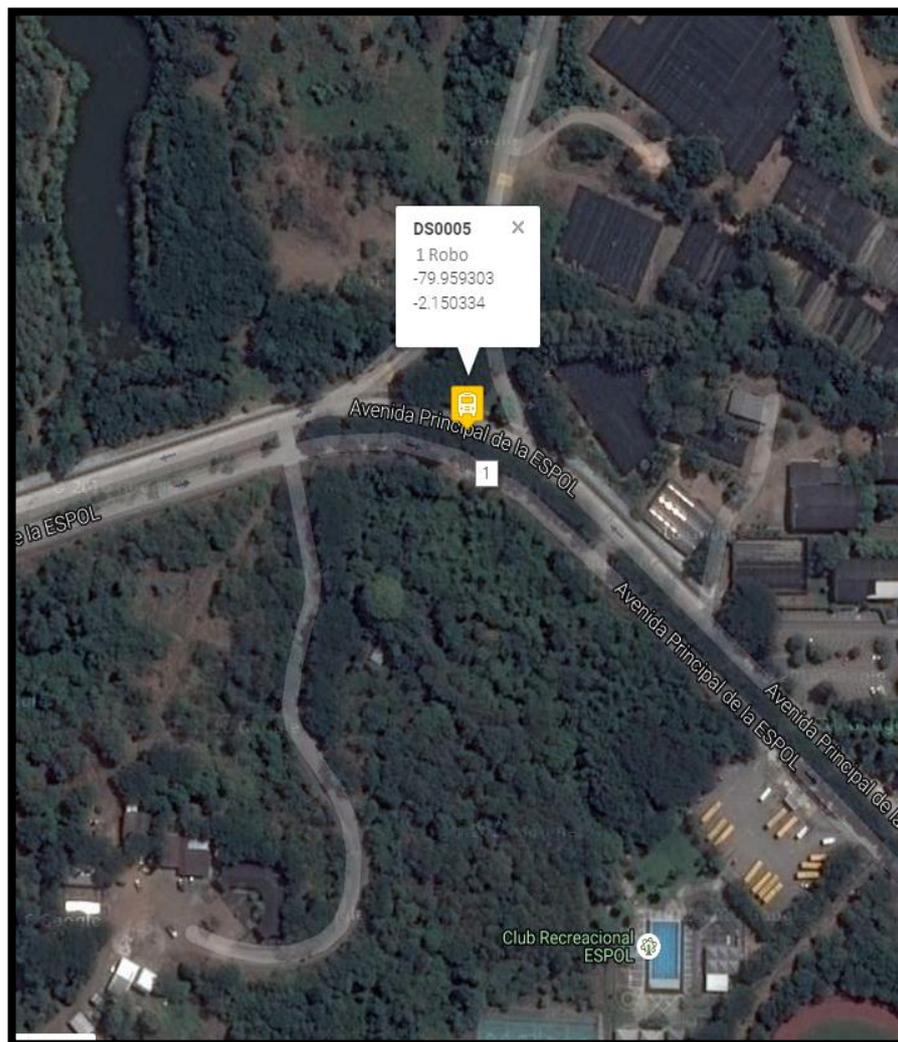


Figura 4.8. Vista del recorrido del bus con sus respectivas alarmas.

En la Figura 4.9 se puede observar otra funcionalidad del software, el cual permite habilitar imágenes en tiempo real a través de BeeTracker la imagen que aparecerá en caso de que se presione la alarma 1 que corresponde a robo, y los siguientes valores corresponden a la longitud y la latitud respectivamente.

En la Figura 4.10 se puede observar en tiempo real a través de BeeTracker la imagen que aparecerá en caso de que se presione la alarma 2 que corresponde a emergencia médica, y los siguientes valores corresponden a la longitud y la latitud respectivamente.

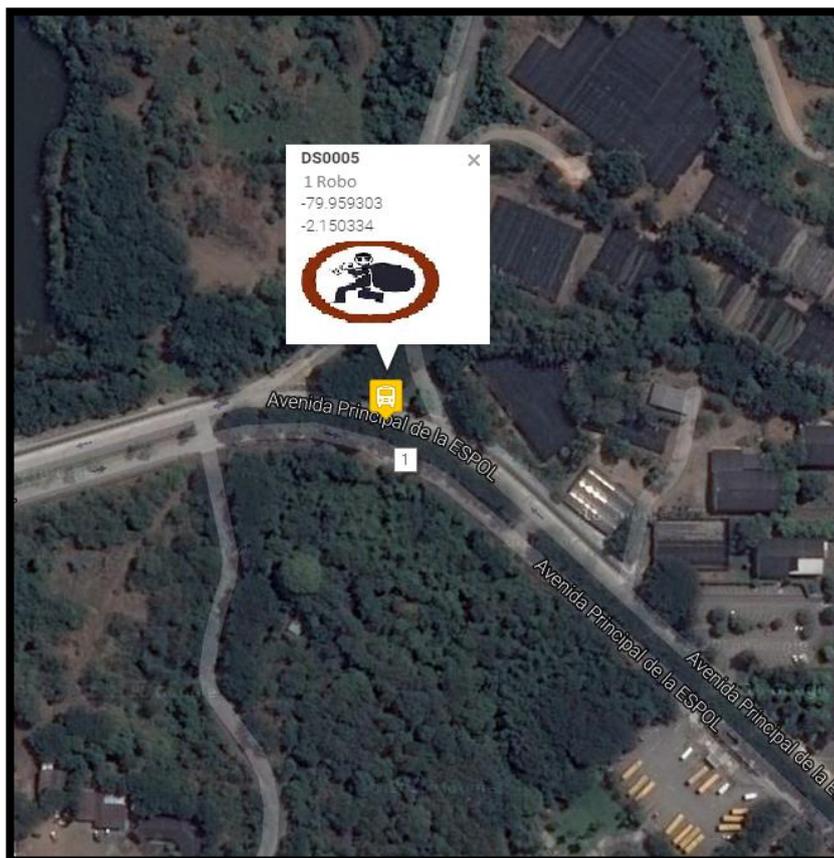


Figura 4.9. Alarma con imagen en caso de robo.

En la Figura 4.11 se puede observar en tiempo real a través de BeeTracker la imagen que aparecerá en caso de que se presione la alarma 3 que corresponde a incendio, y los siguientes valores corresponden a la longitud y la latitud respectivamente.

4.3 Prueba de Módulo bus externo.

Como fue mencionado en el capítulo 3, el módulo externo será muy similar al módulo interno, es por eso que sus ilustraciones son las mismas de las Figuras 4.4 y Figura 4.5, cambiando solamente la programación interna.

En la Figura 4.12 se observa el geo cerco creado dentro de la FIEC, a manera de ejemplo, si la emergencia se suscita dentro de ese sector, el módulo

enviaría un mensaje al número programado del UPC correspondiente a ese geo cerco. Asumiendo que la FIEC es el UPC1.

En la Figura 4.13 se observa el geo cerco creado dentro del paradero del CELEX, a manera de ejemplo si la emergencia se suscita dentro de ese sector el módulo enviaría un mensaje al número programado del UPC correspondiente a ese geo cerco. Simulando que el paradero es el UPC2.

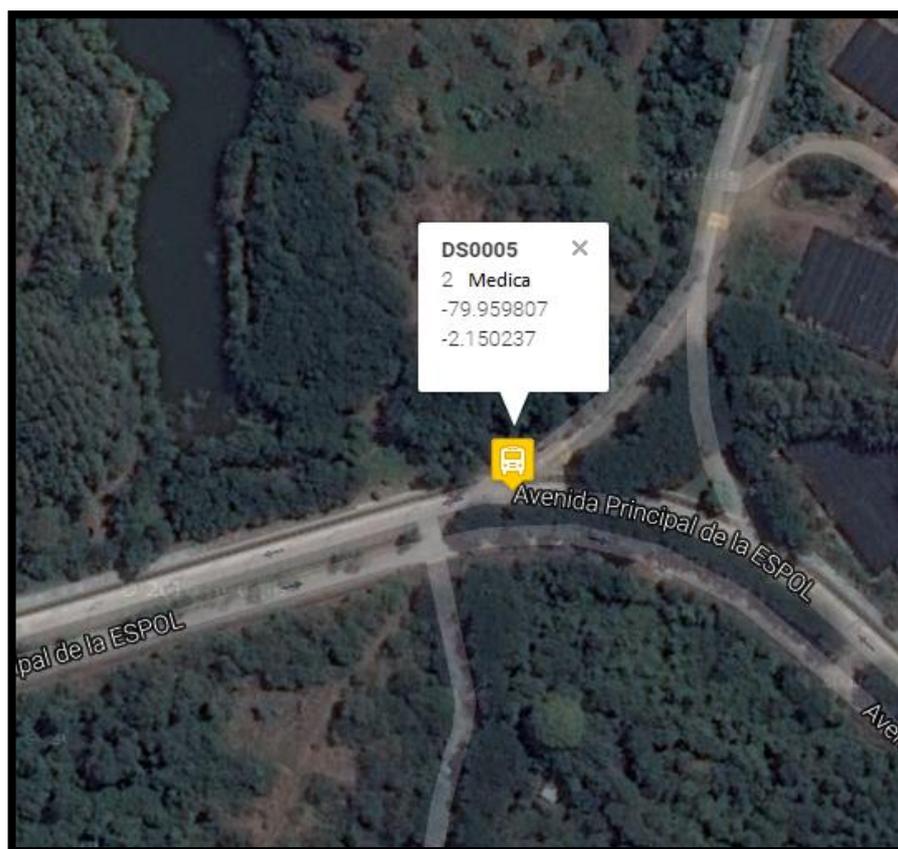


Figura 4.10. En caso de alarma de emergencia médica.

En la Figura 4.14 se puede observar en el BeeTracker la ruta trazada por el bus externo. BeeTracker también nos brinda la posibilidad de recrear rutas para saber si es que el bus se ha desviado de su camino normal, o inclusive saber si va a exceso de velocidad.

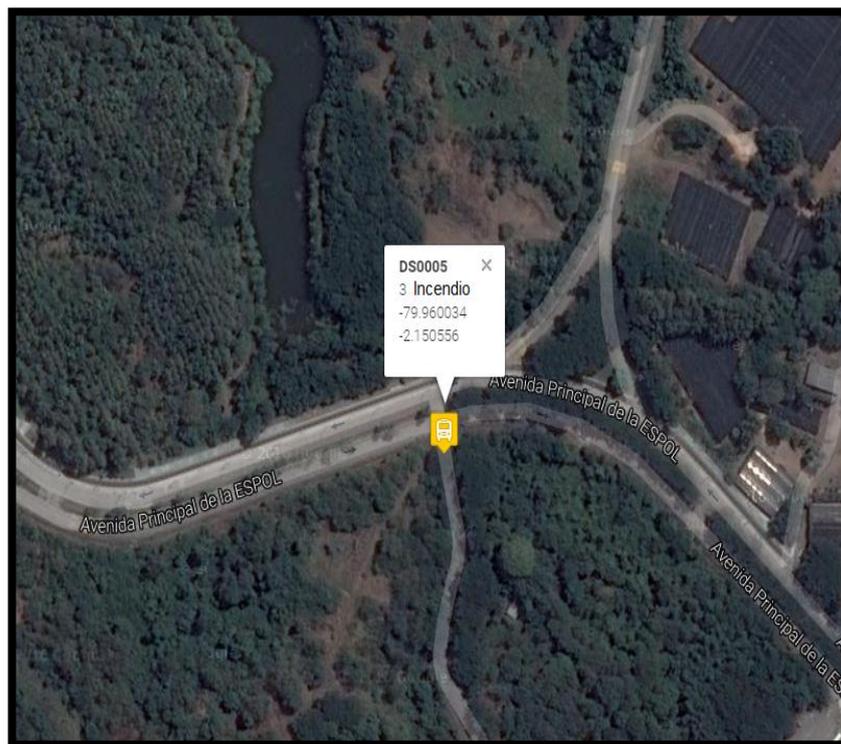


Figura 4.11. En caso de alarma de incendio.

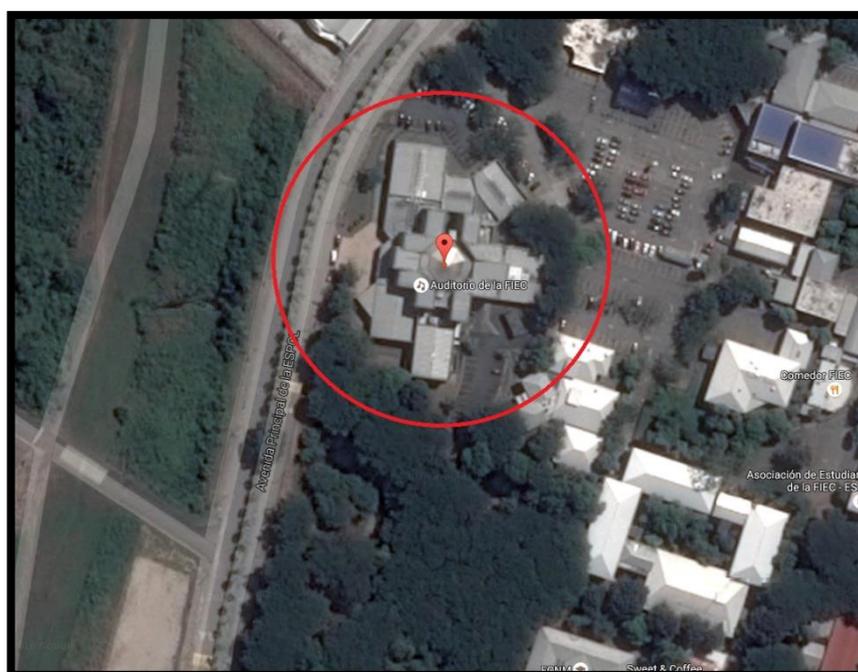


Figura 4.12. Geocercos para enviar mensajes de auxilio al UPC1.



Figura 4.13. Geocercos para enviar mensaje de auxilio al UPC2.

4.4 Prueba de Módulo Paradero Inteligente.

En la Figura 4.15 se puede observar el prototipo del paradero inteligente funcionando con los dos módulos Arduinos, el cual trabaja de la siguiente forma, a medida que van pasando los carnets los estudiantes por el lector este va aumentando el número, hasta que llega al número 50 y envíe un mensaje, informando que manden el siguiente bus. Si ya pasó su carnet ya no será tomado en cuenta dentro del arreglo. Además de eso el módulo irá mostrando el número de carnets ingresados

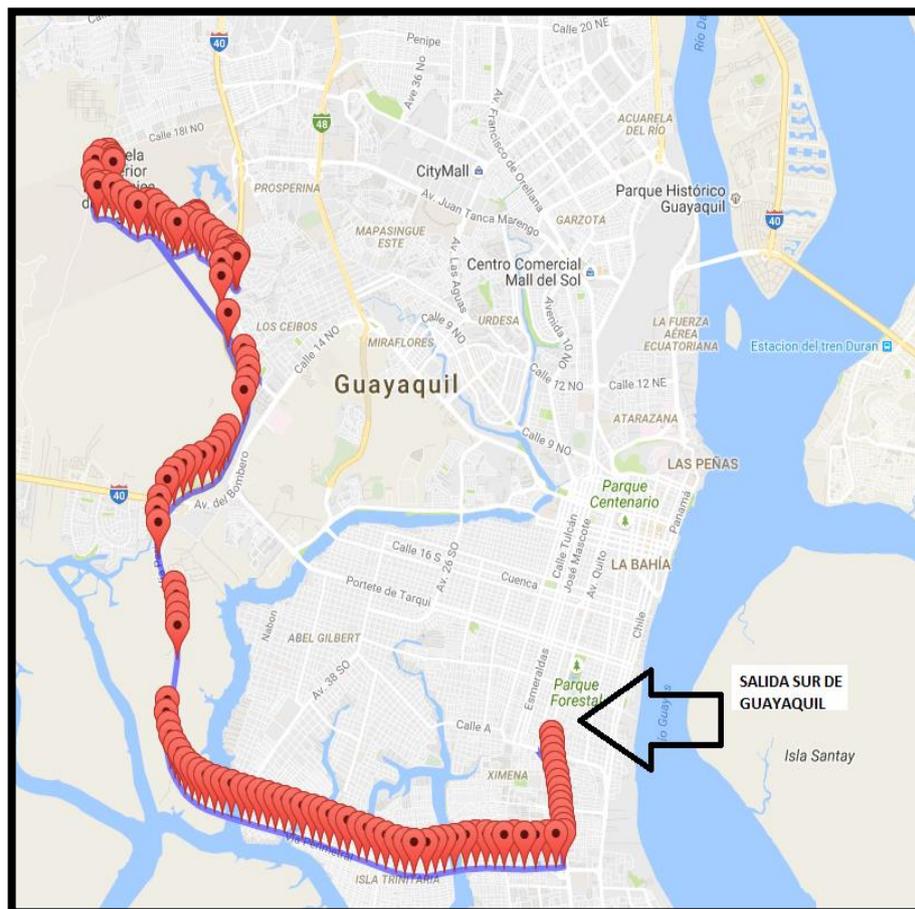


Figura 4. 14. Ruta trazada por el bus externo en BeeTracker

En la Figura 4.16 se puede observar las conexiones físicas del módulo del paradero inteligente en su prototipo final para ser instalado y probado.

4.5 Costos de Producción.

Para poder sacar el costo de producción se tomaron en cuenta varios aspectos como son los costos individuales de cada equipo por módulo, los costos de BeeTracker, y al final el costo de la instalación en los vehículos.

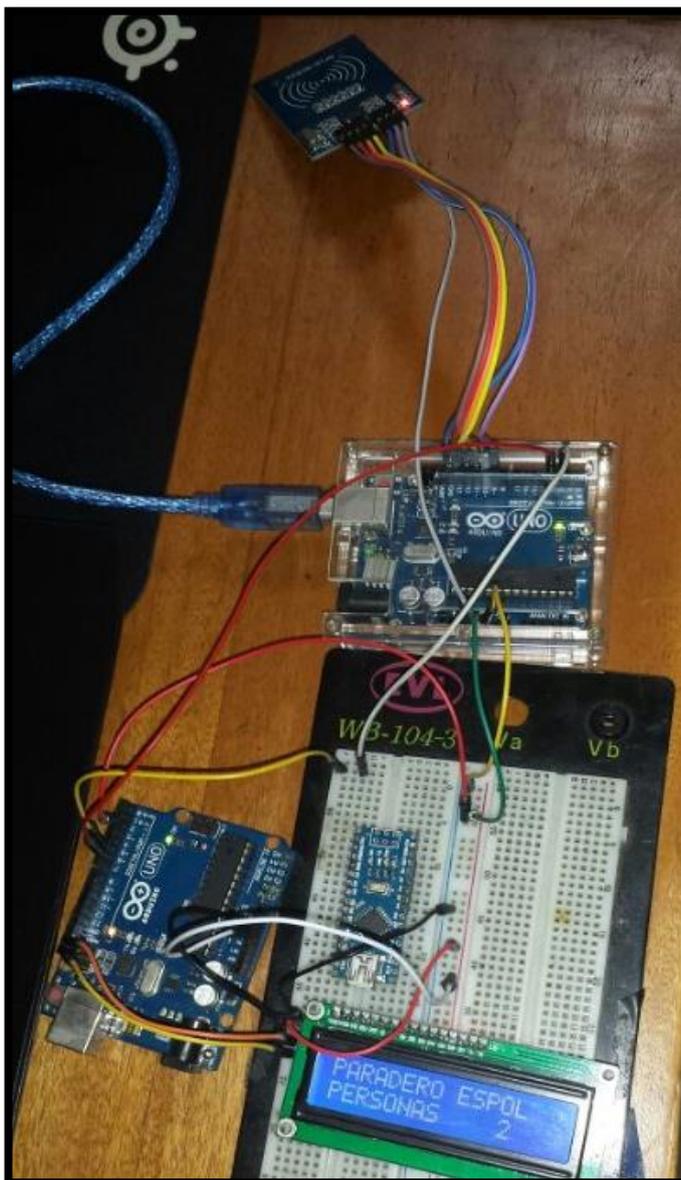


Figura 4. 15. Módulo del paradero inteligente

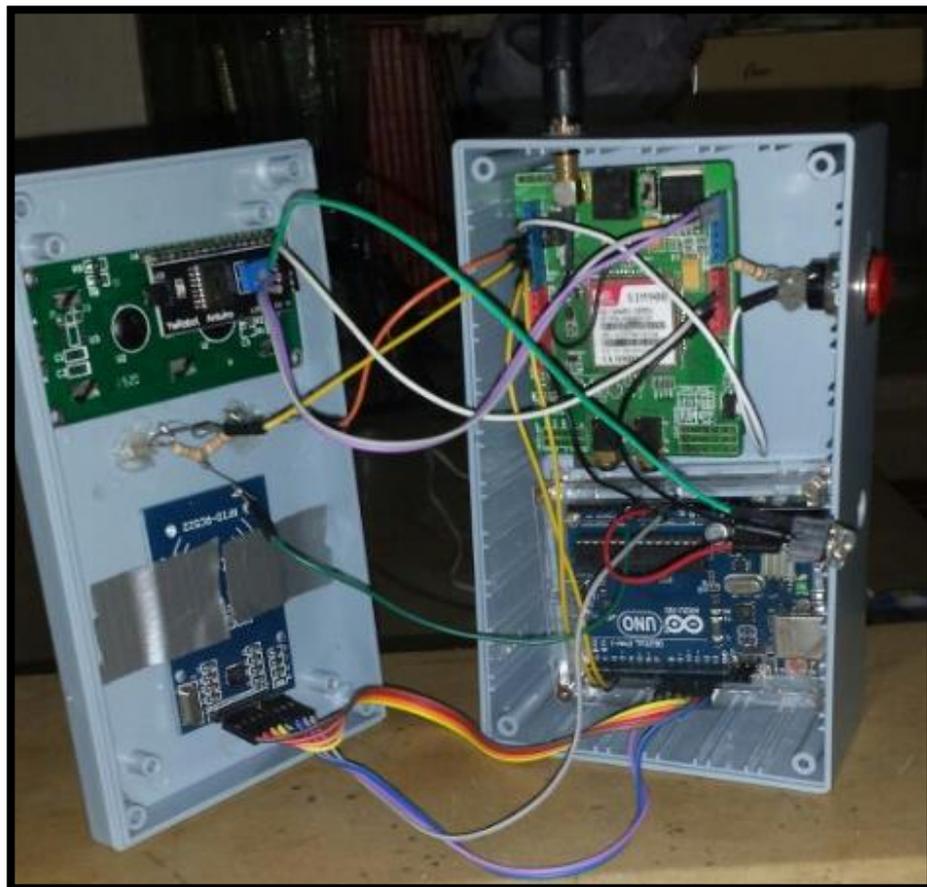


Figura 4. 16. Prototipo del paradero inteligente.

En la Tabla 10 se muestra el detalle de los valores que se invirtió para realizar la implementación del módulo de los buses internos y externos.

Detalle	Costo	Unidad	Total
Arduino Mega	\$35.00	1	\$35.00
Modulo GPS	\$35.00	1	\$35.00
Módulo SIM900	\$45.00	1	\$45.00
Protectores	\$9.00	1	\$9.00
Regulador	\$20.00	1	\$20.00
subtotal			\$169.00
IVA 14%			\$23.66
Total			\$ 192.66

Tabla 10. Costo de elementos de los módulos para los buses internos y externos

En la siguiente Tabla 11 se puede encontrar los costos reducidos por unidad para los módulos de los buses.

Unidades	Costo por unidad
< 5	\$200.00
>5 y <15	\$180.00
>15	\$150.00

Tabla 11. Costo por unidad

En la Tabla 12 se muestra el detalle de los valores que se invirtió para realizar la implementación del prototipo del paradero inteligente.

En la Tabla 13 se puede encontrar los costos reducidos por unidad para el paradero inteligente.

En la Tabla 14 se puede encontrar los costos de mantenimiento mensual de BeeTracker por unidad, el cual para su correcto funcionamiento haría una página libre de credenciales y accesible al público en general.

En la Tabla 15 se encontrará detallado los costos de instalación y de materiales a utilizar para poder tener un sistema sustentable con una batería auxiliar, cables que irían a la batería, y con su respectiva protección de corriente y de sobre voltaje.

Detalle	Costo	Unidad	Total
Arduino UNO	\$15.00	2	\$30.00
Módulo SIM900	\$45.00	1	\$45.00
Pantalla Lcd I2C	\$12.00	1	\$12.00
Módulo RFID reader	\$15.00	1	\$15.00
Placa	\$15.00	1	\$15.00
Pastico protector	\$9.00	1	\$9.00
Subtotal			\$133.00
IVA 14%			\$18.62
Total			\$ 151.62

Tabla 12. Costo de producción de los módulos para el paradero inteligente

Unidades	Costo por unidad
< 5	\$150.00
>5 y <15	\$125.00
>15	\$100.00

Tabla 13. Costo por unidad para los paraderos inteligentes

Unidades	Costo por unidad
< 5	\$5.00
>5 y <15	\$4.00
>15	\$3.00

Tabla 14. Costos de mantenimiento mensual de BeeTracker

Descripción	Costo	Unidad	Total
Cables	\$5.00	1	\$5.00
batería auxiliar	\$18.00	1	\$18.00
mano de obra	\$60.00	1	\$60.00
Instalación	\$80.00	1	\$80.00
subtotal			\$163.00
iva 14%			\$22.82
total			\$185.82

Tabla 15. Costo de instalación y mantenimiento

Unidades	Costo por unidad
< 5	\$180.00
>5 y <15	\$150.00
>15	\$120.00

Tabla 16. Costo por unidad para la instalación en los buses

CONCLUSIONES Y RECOMENDACIONES

Se logró diseñar los sistemas de acople de una manera óptima mediante el uso de puertos de comunicación seriales físicos, con los cuales no hubo problema de sincronización gracias a que el Arduino Mega 2560 cuenta con 4 puertos de este tipo.

No fue necesario el uso de la red 4G debido a los consumos excesivos de 4G con respecto a energía, donde 4G necesita más potencia y trabaja a mayor frecuencia, dando como resultado un consumo innecesario para transmitir una trama de datos muy pequeña.

Se logró diseñar la programación gracias a que se tenía en claro los esquemas a seguir; dejando abierto la posibilidad de ingresar los números de emergencia de los responsables para los casos de alarma sean estos: robo, médica o incendio.

Se diseñó con éxito los geocercos, gracias a los conocimientos matemáticos aprendidos en cursos anteriores, asociados a la ruta del Sur de la ESPOL; dejando documentado sólo para añadir los números correspondientes a las rutas que sigan.

Se implementó los módulos dentro de un vehículo particular, el cual no nos permitió probar su funcionamiento en conexión con el interruptor de arranque del vehículo como entrada para validación del sistema, ni el uso de los botones de emergencia en un caso real.

Se diseñó e implemento con éxito un sistema de conteo de personas mediante los carnets de los estudiantes; dejando preparado el código para enviar el mensaje para otro bus, cuando el número de personas supere la media de los buses que es de 50 personas.

La posición de la antena GPS o del módulo deberá estar en un compartimiento del carro intentando que esté ubicada en 45 grados, con respecto al suelo para poder obtener mejor recepción de señal.

No encerrar el módulo en una jaula de Faraday, ya que esto actuaría como un aislador de señal, o al menos dejar la antena GPRS fuera de la jaula para poder captar la señal celular.

Adicionar un extractor con un PID de temperatura a fin de que siempre se encuentre funcionando dentro de sus estándares de operación, ya que el módulo GPRS es un poco sensible al calor concentrado, haciendo que se reinicie de manera indebida.

BIBLIOGRAFÍA

- [1] R. Gakenheimer. (1998). *Los problemas de la movilidad en el mundo en desarrollo*, EURE (Santiago) [online], vol. 24, nº 72, pp. 33-52. Available : <http://dx.doi.org/10.4067/S0250-71611998007200002>.
- [2] M. Monzón. (2005). *Gestión del transporte metropolitano. Gobernar las metrópolis*, Europa Artes Gráficas (Washington, DC) [online], vol. 519, p. 409. Available : <http://www.bvsde.paho.org/documentosdigitales/bvsde/texcom/cd050644/erojas.pdf>.
- [3] A. Mendoza, "Análisis y solución al congestionamiento vehicular en horas pico utilizando una aplicación móvil con GPS", Tesis de Ingeniería en Sistemas. UPSG, Guayaquil, Ecuador, 2014.
- [4] M. Alcivar, P. Gavilanes, "Análisis de competitividad del transporte público y el transporte privado en la ciudad de Guayaquil-Sector Sauces" Tesis Maestría en transportes ,PUCE, Guayaquil, Ecuador, 2014.
- [5] P. Andrade, "Investigación de mercado para transespol: Implementación de tarjetas prepago como posible solución al problema del transporte de la ESPOL", ESPOL, Guayaquil, Ecuador 2002.
- [6] (Escuela Superior Politecnica del Litoral) [online], Available : <http://www.espol.edu.ec/espol/main.jsp?urlpage=historiaespol.jsp>.
- [7] J. Toaza, "Guía metodológica para el análisis forense en incidentes de teléfono celular con tecnología GSM" Tesis de Ingeniería en Sistemas Computacionales, Universidad de Guayaquil, Guayaquil, Ecuador, 2011.
- [8] J. López, *Gps Tecnología más allá de la Atmosfera*, Observatorio Tecnológico [online], nº 289, p. 820, 13 Junio 2006. Available : <http://recursostic.educacion.es/observatorio/version/v2/eu/equipamiento-tecnologico/didactica-de-la-tecnologia/357-gps-tecnologia-mas-alla-de-la-atmosfera>
- [9] ecu911.gob.ec,(Servicio Integrado de Seguridad ECU 911 | Ecuador), [online]. Disponible en: <http://www.ecu911.gob.ec/>.

- [10] P. Vidal, *Despierta y Paraderos Inteligentes, Ciudades Inteligentes [online]*, (27 Octubre 2014). Available : <http://www.ciudadesinteligentes.cl/emprendedores-chilenos-presentan-aplicaciones-para-resolver-problemas-de-movilidad/>
- [11] P. Vidal, *Planificador de Viaje, Ciudades Inteligentes [online]*, 2012. Available : <http://www.ciudadesinteligentes.cl/portfolio/planificador-de-viajes/>
- [12] P. Vidal, *Sistema Automático de tiempos de viaje, Ciudades Inteligentes [online]*, 2014. Available : <http://www.ciudadesinteligentes.cl/portfolio/sistema-automatico-de-tiempos-de-viaje/>
- [13] I. Espinosa, (2002) Proyecto Artemisa. Articulación Tecnológica para la mejora integral del servicio de transporte en autobus. In III Congreso Nacional sobre sistemas inteligentes de transporte seguridad, información y tecnología *The National Academies of Sciences-Engineering-Medicine [online]*, nº 43, p. 637,.
- [14] M. Carrasco, "Sistemas de alarmas para mejorar la seguridad de la Empresa Auplatec ubicada en el canton Pelileo." Tesis en Ingeniería en Electrónica y Comunicaciones. Ambato, Ecuador 2012.
- [15] Conecel.s.a, 1993. [online]. Available : <http://www.claro.com.ec/portal/ec/sc/personas/institucional/>.
- [16] Otecel.s.a, 10 Sept 1993. [online]. Available : https://www.emis.com/php/company-profile/EC/Otecel_SA_es_2082181.html.
- [17] *Sim Cards Have Finally Been Hacked, And The Flaw Could Affect Millions Of Phones*, 21 Julio 2013. [online]. Available : <http://www.forbes.com/sites/parmyolson/2013/07/21/sim-cards-have-finally-been-hacked-and-the-flaw-could-affect-millions-of-phones/#644c4f144e39>.
- [18] P. Sanchez, "Análisis del Desempeño de las Redes Celulares GSM-GPRS" Tesis Ingeniero en Telecomunicaciones, UNAM, Mexico, 2005.
- [19] E. Quesada, "Sistemas de Provisionamiento para Abonados de Telefonía Móvil" Tesis Ingeniería en Telecomunicaciones, UPM, Madrid, España, 2012.

- [20] J. Lozada, "Sistema de Gestión para la detección temprana de fallas en las BTS de la red digital" Tesis Ingeniería en comunicación, UPM, Madrid, España, 2005.
- [21] P. Burneo, P. Siguenza (2013) *Evolucion del sistema celular hacia UMTS*, [online]. Available : <https://www.researchgate.net>
- [22] J.Timo Halonen, *GSM, GPRS and EDGE Performance: Evolution Towards 3G/UMTS, Second Edition*, John Wiley & Sons, Ltd 15 Mar 2004.
- [23] P. Burneo, P. Siguenza (2013), *Evolucion del sistema celular hacia UMTS*, [online]. Available :: <https://www.researchgate.net>
- [24] J. Astudillo, "Sistema de Localización monitoreo y control vehicular basados en protocolos GPS/GSM/GPRS" Tesis Ing. Electrónico, Univ, Politécnica Salesiana Cuenca, Ecuador, 2012.
- [25] Carturla, J. L "Sistema de Posicionamiento Global (GPS)". Departamento de la Guajira, Colombia, 2007.
- [26] C. Recalde, "Estudio, diseño e implementación de un sistema de monitoreo de control de rondas para la empresa Betancourt & Zambrano Seguridad cía. Ltda utilizando la plataforma de openstreetmaps y software libre" Tesis Electrónica y Telecomunicaciones, Univ. Tec. Israel, Quito, Ecuador 2013.
- [27] A. Tirabassi, "Monitoreo satelital de vehículos mediante una aplicación web," Tesina de Licenciatura, Univ. Nacional de la Plata, La Plata, Argentina 2011.
- [28] G. Llinares, (enero 2010) *Sistemas Embebidos*, [online]. Available : <http://server-die.alc.upv.es/asignaturas/PAEEES/2005-06/A07%20-%20Sistemas%20Embebidos.pdf>
- [29] P. Miguel, *Electronica Aplicada*, Paraninfo f.a, 2016, Ediciones Paraninfo 2ª Edicion p. 415.
- [30] Arduino Genuino, [online]. Available : <https://www.arduino.cc>.
- [31] O.Torrente, *Arduino Curso Practico de formacion, RC libros*, 2013, pp. 79-82.

- [32] S. Wireless, Ed., *SIM900 AT* [online] 2010, p. 229, Available: http://www.espruino.com/datasheets/SIM900_AT.pdf .
- [33] SimCom.EE, *Sim900*, RCLibros [online], Available :http://www.espruino.com/datasheets/SIM900_AT.pdf 2013.
- [34] Golem Electronics, *Modulo Gsm Sim900 para arduino*, 2016. [online]. Available : <http://www.golemelectronics.com/producto/modulo-gsm-sim900-para-arduino/>.
- [35] SimCom, *Wireless Modules > GSM/GPRS Module*, 9 Abril 2015. [online]. Available: <http://simcomm2m.com/En/module/detail.aspx?id=138>.
- [36] F.Weston, *Física moderna*, [online] Pearson Educacion vol. 2, p. 713, 2005.
- [37] R. Tocci., *Sistemas Digitales Principios y Aplicaciones*, vol. 5, J. P. Vonilla, Ed., Hispanoamerica S.A.
- [38] A. Donate, *Electrónica Aplicada*, vol. 2, Barcelona: marcombo formación, 2012, p. 484.
- [39] S. López, *Dispositivos Electrónicos de Potencia*, Toás Di Bella ed., vol. 1, Baja California: UABC, 2002, p. 47.
- [40] Taringa.net, 2013. [Online]. Available: <http://www.taringa.net/post/info/15881961/Fuentes-de-poder-o-conmutadas.html>.
- [41] O. Ginebra, *Consecuencias Sociales y laborales de una mayor utilización de la tecnologías avanzadas destinadas a ninoristas*, vol. 1, Ginebra: TMART, 2006, p. 61.
- [42] *Panda-Id Soluciones*, 1996. [Online]. Available: <http://www.panda-id.com/que-es-una-etiqueta-rfid/>.
- [43] H. Torres, *Hetpro Herramientas tecnológicas profesionales*, 11 Agosto 2014. [Online]. Available: <http://hetpro-store.com/TUTORIALES/gps-ublox-neo-6m-modulo-con-matlab/>.

ANEXOS

```

/*
Código funcional del Arduino mega conectado por conexión serial con el sim 900 y
el modulo gps. Además de eso tiene los botones por interrupción que envían
mensaje cuando son aplastados y envían la posición gps cada cierto tiempo.
Lo que significa que cuando es conectado el modulo se enciende solo y envía
tramas cada 10 segundos
*/
#include <TinyGPS.h>
long lata,latb,lona,lonb;
int id_bus=1;
int alarma=0;
String cadena;
String one="AT";
String two="+";
String tree="HTTTPARA";
String four="=\\";
String five="URL";
String six="\\,\\\\";
String seven="http://enjambre.ec/bt/point/";
String barra="/";
String punto=".";
String fin="^";
long lat,lon;
int alarma1=0;//pin 2
int alarma2=1;//pin 3
int alarma3=2;//pin 21
TinyGPS gps;
void setup()
{
  Serial.begin(9600);
  Serial2.begin(19200);//sim900
  Serial1.begin(9600);//gps
  attachInterrupt(alarma1,Interr0,RISING);
  attachInterrupt(alarma2,Interr1,RISING);
  attachInterrupt(alarma3,Interr2,RISING);
  Serial2power();
}
void loop()
{
  while(Serial1.available())
  {
    if(gps.encode(Serial1.read()))
    {
      gps.get_position(&lat,&lon);
      SubmitHttpRequest();
      if(alarma==1 || alarma==2 || alarma==3)

```

```

    {
        Serial2.print("AT+CMGF=1\r");// AT comando AT para enviar mensaje
        delay(100);
        if(alarma==1)
            Serial2.println("AT+CMGS = \"+593990000000\");//numero de telefono
correspondiente
        if(alarma==2)
            Serial2.println("AT+CMGS = \"+593991529449\");//numero de telefono
correspondiente
        if(alarma==3)
            Serial2.println("AT+CMGS = \"+593991529449\");//numero de telefono
correspondiente
        delay(100);
        if(alarma==1)
            Serial2.println(" ALERTA DE EMERGENCIA ROBO - PROYECTO
INTEGRADOR - ");
        if(alarma==2)
            Serial2.println(" ALERTA DE EMERGENCIA MEDICA - PROYECTO
INTEGRADOR - ");
        if(alarma==3)
            Serial2.println(" ALERTA DE EMERGENCIA INCENDIO- PROYECTO
INTEGRADOR - ");
        Serial2.println();
        Serial2.println(imprimir(lat,lon));
        delay(100);
        Serial2.println((char)26); // comando AT que finaliza el sms con ^Z, el codigo
ascii de 26
        delay(100);
        Serial2.println();
        delay(5000);
    }
    alarma=0;
}
}
}
void SubmitHttpRequest()
{
    Serial2.println("AT+CSQ"); // chequea la calidad de la senal
    delay(100);
    ShowSerialData();//muestra como interactua el shield
    Serial2.println("AT+CGATT?");
    delay(100);
    ShowSerialData();
    Serial2.println("AT+SAPBR=3,1,\"CONTYPE\",\"GPRS\");//setea parametros de
configuracion para usar
    delay(1000);
    ShowSerialData();
    Serial2.println("AT+SAPBR=3,1,\"APN\",\"internet.claro.com.ec\");//setea el APN

```

```

delay(4000);
ShowSerialData();
Serial2.println("AT+SAPBR=1,1");
delay(2000);
ShowSerialData();
Serial2.println("AT+HTTPINIT");
delay(2000);
ShowSerialData();
Serial2.println(imprimir(lat,lon));
delay(1000);
ShowSerialData();
Serial2.println("AT+HTTPACTION=0");
delay(10000);//este retarde ese en base a la cantidad de datos retornado del sitio
web, a mayor datos mayor tiempo de espera
ShowSerialData();
Serial2.println("AT+HTTPREAD");
delay(300);
ShowSerialData();
Serial2.println("");
delay(100);
}
void ShowSerialData()
{
  while(Serial2.available()!=0)
    Serial.write(char (Serial2.read()));
}
void Serial2power()
{
  digitalWrite(9, HIGH);
  delay(1000);
  digitalWrite(9, LOW);
  delay(5000);
}
String imprimir(long lat,long lon)
{
  lata=lat/1000000;
  latb=(lat%1000000)*(-1);
  lona=lon/1000000;
  lonb=(lon%1000000)*(-1);
  cadena=one+two;
  cadena=cadena+tree;
  cadena=cadena+four;
  cadena=cadena+five;
  cadena=cadena+six;
  cadena=cadena+seven;
  cadena=cadena+id_bus;
  cadena=cadena +barra;
  cadena=cadena+lata;
}

```

```

cadena=cadena+punto;
cadena=cadena+latb;
cadena=cadena+barra;
cadena =cadena+lona;
cadena=cadena+punto;
cadena=cadena+lonb;
cadena=cadena+barra;
cadena=cadena+alarma;
cadena=cadena+fin;
return cadena;
}
void Interr0()
{ alarma=1; }
void Interr1()
{ alarma=2; }
void Interr2()
{ alarma=3; }

```

Anexo 1: Código bus interno

```

#include <TinyGPS.h>
long lata,latb,lona,lonb;
int id_bus=1;
int alarma=0;
String cadena;
String one="AT";
String two="+";
String tree="HTTTPARA";
String four="=";
String five="URL";
String six="\\,\\";
String seven="http://enjambre.ec/bt/point/";
String barra="/";
String punto=".";
String fin="^";
long lat,lon;
long lat2,lon2;
int alarma1=0;//pin 2
int alarma2=1;//pin 3
int alarma3=2;//pin 21
//ubicacion gps de la fiec
float a=-2.144651;//CENTRO X
float b=-79.967644;//CENTRO Y
//el radio al cuadrado es igual a (el punto x de radio menos el punto de x de origen)
al cuadrado
// el punto y de radio menos el punto de y de origen) al cuadrado

```

```

//71,53 -2.144653, -79.968268
float radio1=0.0000028;
//ubicacion gps del paradero basico nuevo
float a1=-2.148192;//CENTRO X
float b1=-79.967902;//CENTRO Y
//91,68m -2.147978, -79.967094
float radio2=0.0000028;
TinyGPS gps;

void setup()
{
  Serial.begin(9600);
  Serial2.begin(19200);//sim900
  Serial1.begin(9600);//gps
  attachInterrupt(alarma1,Interr0,RISING);
  attachInterrupt(alarma2,Interr1,RISING);
  attachInterrupt(alarma3,Interr2,RISING);
  Serial2power();
}
void loop()
{
  while(Serial1.available())
  {
    if(gps.encode(Serial1.read()))
    {
      gps.get_position(&lat,&lon);
      lat2=lat;
      lon2=lon;
      Serial.println("SubmitHttpRequest - started" );
      SubmitHttpRequest();
      Serial.println("SubmitHttpRequest - finished" );
      if(alarma==1 || alarma==2 || alarma==3)
      {
        smsGeofence();//funcion de geocerco
      }
      alarma=0;
    }
  }
}
void smsGeofence()
{
  float aux5;
  float aux6;
  float aux7;
  float aux8;
  float aux;
  float aux2;
  float aux3;

```

```

float aux4;
float aux1;
float aux12;
float aux13;
float aux14;
aux=lat2/1000000;
aux2=lat2%1000000;
aux3=aux2*0.000001;
aux4=aux+aux3;
aux1=lon2/1000000;
aux12=lon2%1000000;
aux13=aux12*0.000001;
aux14=aux1+aux13;
//ya tengo las coordenadas del punto ahora preguntar si esta dentro de los
geocercos
aux5=aux4-a;
aux5=pow(aux5,2);
aux6=aux14-b;
aux6=pow(aux6,2);

aux7=aux4-a1;
aux7=pow(aux7,2);
aux8=aux14-b1;
aux8=pow(aux8,2);
Serial2.print("AT+CMGF=1\r");// AT comando AT para enviar mensaje
delay(100);
///CONDICION DENTRO O FUERA DEL GEOCERCO
if(aux5+aux6<=radio1)
{
  Serial2.println("AT+CMGS = \"+593990000000\");//numero de telefono
correspondiente
}
else if(aux7+aux8<=radio2)
{
  Serial2.println("AT+CMGS = \"+593990000000\");//numero de telefono
correspondiente
}
else
{
  Serial2.println("AT+CMGS = \"+593990000000\");//numero de telefono
correspondiente
}
delay(100);
if(alarma==1)
  Serial2.println(" ALERTA DE EMERGENCIA ROBO - PROYECTO
INTEGRADOR - ");
if(alarma==2)

```

```

        Serial2.println(" ALERTA DE EMERGENCIA MEDICA - PROYECTO
INTEGRADOR - ");
        if(alarma==3)
            Serial2.println(" ALERTA DE EMERGENCIA INCENDIO- PROYECTO
INTEGRADOR - ");
        Serial2.println();
        Serial2.println(imprimir(lat,lon));
        delay(100);
        Serial2.println((char)26); // comando AT que finaliza el sms con ^Z, el codigo
ascii de 26
        delay(100);
        Serial2.println();
        delay(5000);
    }

void SubmitHttpRequest()
{
    Serial2.println("AT+CSQ"); // chequea la calidad de la senal
    delay(100);
    ShowSerialData();//muestra como interactua el shield
    Serial2.println("AT+CGATT?");
    delay(100);
    ShowSerialData();
    Serial2.println("AT+SAPBR=3,1,\"CONTYPE\",\"GPRS\"");
    delay(1000);
    ShowSerialData();
    Serial2.println("AT+SAPBR=3,1,\"APN\",\"internet.claro.com.ec\"");
    delay(4000);
    ShowSerialData();
    Serial2.println("AT+SAPBR=1,1");
    delay(2000);
    ShowSerialData();
    Serial2.println("AT+HTTPIPINIT");
    delay(2000);
    ShowSerialData();
    Serial2.println(imprimir(lat,lon));
    delay(1000);
    ShowSerialData();
    Serial2.println("AT+HTTPACTION=0");
    delay(10000);
    ShowSerialData();
    Serial2.println("AT+HTTPREAD");
    delay(300);
    ShowSerialData();
    Serial2.println("");
    delay(100);
}

void ShowSerialData()

```

```

{
  while(Serial2.available()!=0)
    Serial.write(char (Serial2.read()));
}
void Serial2power()
{
  digitalWrite(9, HIGH);
  delay(1000);
  digitalWrite(9, LOW);
  delay(5000);
}
String imprimir(long lat,long lon)
{
  //AT+HTTTPARA="URL",\"http://enjambre.ec/bt/point/1/-2.148162/-
79.9644815/25^"
  lata=lat/1000000;
  latb=(lat%1000000)*(-1);;
  lona=lon/1000000;
  lonb=(lon%1000000)*(-1);;
  cadena=one+two;
  cadena=cadena+tree;
  cadena=cadena+four;
  cadena=cadena+five;
  cadena=cadena+six;
  cadena=cadena+seven;
  cadena=cadena+id_bus;
  cadena=cadena +barra;
  cadena=cadena+lata;
  cadena=cadena+punto;
  cadena=cadena+latb;
  cadena=cadena+barra;
  cadena =cadena+lona;
  cadena=cadena+punto;
  cadena=cadena+lonb;
  cadena=cadena+barra;
  cadena=cadena+alarma;
  cadena=cadena+fin;
  return cadena;
}
void Interr0()
{ alarma=1; }
void Interr1()
{ alarma=2; }
void Interr2()
{ alarma=3; }

```

```

#include <AddicoreRFID.h>
#include <SPI.h>
#define uchar unsigned char
#define uint unsigned int
uchar serNumA[5];
uchar fifobytes;
uchar fifoValue;
AddicoreRFID myRFID;
const int chipSelectPin = 10;
const int NRSTPD = 5;

#define MAX_LEN 16
#define MAX 50//numero maximo de personas que se almacenaran en el arreglo
int a[MAX];
int aux=0;
void setup()
{
  Serial.begin(9600);
  pinMode(2,OUTPUT); // envio de senal
  pinMode(3,INPUT); // recepcion de senal para continuar

  SPI.begin();
  pinMode(chipSelectPin,OUTPUT);
  digitalWrite(chipSelectPin, LOW);
  pinMode(NRSTPD,OUTPUT);
  digitalWrite(NRSTPD, HIGH);
  myRFID.AddicoreRFID_Init();
  Serial.print("Listo");
  encerrar();
}
int cont=1;
int senso=0;
void loop()
{
  if(digitalRead(3)==HIGH);
  {
    senso=0;
    delay(1000);
  }

  if (senso==0)
  {
    uchar i, tmp, checksum1;
    uchar status;
    uchar str[MAX_LEN];
    String mynum = "";
    str[1] = 0x4400;
    status = myRFID.AddicoreRFID_Request(PICC_REQIDL, str);
  }
}

```

```

if (status == MI_OK)
{
}
status = myRFID.AddicoreRFID_Anticoll(str);
if (status == MI_OK)
{
if (cont==3)
{
Serial.print("enviando mensaje");
cont=1;
encerar();
}
aux=str[0];
if(revisar(aux)==1)
Serial.print("ya esta ingresado");
else
{
a[cont-1]=aux;
Serial.print("Ingresado con Exito ");
Serial.println(a[cont-1]);
cont++;
digitalWrite(2,HIGH);
delay(1000);
digitalWrite(2,LOW);
senso=1;
}
}
myRFID.AddicoreRFID_Halt();
}

}
void encerar()
{
int i=0;
for(i=0;i<MAX;i++)
{
a[i]=0;
}
}
/*
retorna 1 si es q si hay un numero repetido en el arreglo
retorna 0 si no hay nada repetido
*/
int revisar(int aux)
{
int i=0;
for (i=0;i<cont;i++)
{

```

```

    if(a[i]==aux)
        return 1;
    }
    return 0;
}

```

Anexo 3: Código paradero inteligente micro 1

```

#include <SoftwareSerial.h>
#include <LiquidCrystal.h>
SoftwareSerial SIM900(7, 8);
#include <Wire.h>
#include <LCD.h>
#include <LiquidCrystal_I2C.h>

#define I2C_ADDR 0x27
#define BACKLIGHT_PIN 3
#define En_pin 2
#define Rw_pin 1
#define Rs_pin 0
#define D4_pin 4
#define D5_pin 5
#define D6_pin 6
#define D7_pin 7
LiquidCrystal_I2C
lcd(I2C_ADDR,En_pin,Rw_pin,Rs_pin,D4_pin,D5_pin,D6_pin,D7_pin);
int cont=0;
int alarma1=0;//pin 2
int alarma=0;
void setup()
{
    pinMode(9,OUTPUT); // encendido sim900
    pinMode(3,OUTPUT); // led verde para proseguir
    pinMode(4,OUTPUT);// led rojo para ocupado
    pinMode(5,INPUT);// recepcion de contador
    pinMode(6,OUTPUT);// envio de seguir contando
    SIM900.begin(19200);
    lcd.begin (16,2);
    lcd.setBacklightPin(BACKLIGHT_PIN,POSITIVE);
    lcd.setBacklight(HIGH);
    lcd.home ();
    lcd.print("PARADERO ESPOL");
    lcd.setCursor(0,1); // Situamos el cursor en la primera columna y la primera fila
    lcd.print("PERSONAS ");
    attachInterrupt(alarma1,Interr0,RISING);
}
void loop()

```

```

{
if(digitalRead(5)==HIGH);
{
  cont++;
  if(cont==50)
  {
    digitalWrite(3,LOW);
    digitalWrite(4,HIGH);
    lcd.clear();
    lcd.print("OCUPADO");
    lcd.setCursor(0,1);
    lcd.print("ENVIANDO SMS");
    digitalWrite(6,HIGH);
    SIM900power();
    EnviarSMS();
    SIM900power();
    digitalWrite(6,LOW);
    digitalWrite(3,HIGH);
    digitalWrite(4,LOW);
    cont=1;
    lcd.home ();
    lcd.print("PARADERO ESPOL");
    lcd.setCursor(0,1);
    lcd.print("PERSONAS ");
    lcd.setCursor(12,1);
    lcd.print(cont);
  }
  digitalWrite(3,LOW);
  digitalWrite(4,HIGH);
  lcd.clear();
  lcd.print("OCUPADO");
  delay(1000);
  digitalWrite(6,HIGH);
  delay(1000);
  lcd.home ();
  lcd.print("PARADERO ESPOL");
  lcd.setCursor(0,1);
  lcd.print("PERSONAS ");
  lcd.setCursor(12,1);
  lcd.print(cont);
  digitalWrite(6,LOW);
  digitalWrite(3,HIGH);
  digitalWrite(4,LOW);
  digitalWrite(6,HIGH);
}
}
if(alarma==1)
{
  digitalWrite(3,LOW);

```

```

digitalWrite(4,HIGH);
lcd.clear();
lcd.print("ALARMA ACTIVADA");
lcd.setCursor(0,1);
lcd.print("ENVIANDO SMS");
SIM900power();
EnviarSMSAlarma();
SIM900power();
digitalWrite(3,HIGH);
digitalWrite(4,LOW);
lcd.home ();
lcd.print("PARADERO ESPOL");
lcd.setCursor(0,1);
lcd.print("PERSONAS ");
lcd.setCursor(12,1);
lcd.print(cont);
alarma=0;
}
lcd.setCursor(12,1);
lcd.print(cont);
}
void SIM900power()
{
digitalWrite(9,HIGH);
delay(1000);
digitalWrite(9, LOW);
delay(3000);
}
void EnviarSMSAlarma()
{
SIM900.print("AT+CMGF=1\r");
delay(100);
SIM900.println("AT + CMGS = \"+5939900000000\");
delay(100);
SIM900.print("ALARMA DE EMERGENCIA ACTIVADA LLAMAR UPC");
delay(100);
SIM900.println((char)26);
delay(100);
SIM900.println();
delay(5000);
}
void EnviarSMS()
{
SIM900.print("AT+CMGF=1\r");
delay(100);
SIM900.println("AT + CMGS = \"+5939900000000\");
delay(100);
SIM900.print("Enviar nuevo bus. El paradero esta lleno");
}

```

```
delay(100);  
SIM900.println((char)26);  
delay(100);  
SIM900.println();  
delay(5000);  
}  
void Interr0()  
{ alarma=1; }
```