

2023

Diseño de un sistema embebido para alarmas de seguridad vehicular
por medio de tecnología de Sistema de Posicionamiento Global Tracker.

Elías Gabriel Nápoles Ramos



Universidad Autónoma de Querétaro
Facultad de Ingeniería.



Nombre de la tesis:

Diseño de un sistema embebido para alarmas de seguridad
vehicular por medio de tecnología de Sistema de
Posicionamiento Global Tracker.

Que como parte de los requisitos para obtener el grado
de Ingeniero en Automatización con línea terminal en
Instrumentación y control.

Presenta

Elías Gabriel Nápoles Ramos

Santiago de Querétaro a 18 de abril del 2023



Dirección General de Bibliotecas y Servicios Digitales
de Información



Diseño de un sistema embebido para alarmas de
seguridad vehicular por medio de tecnología de
Sistema de Posicionamiento Global Tracker

por

Elias Gabriel Napoles Ramos

se distribuye bajo una [Licencia Creative Commons
Atribución-NoComercial-SinDerivadas 4.0
Internacional](https://creativecommons.org/licenses/by-nc-nd/4.0/).

Clave RI: IGLIN-257538



Universidad Autónoma de Querétaro

Facultad de Ingeniería



Ingeniero en Automatización con línea terminal en instrumentación y control de procesos

Tesis:

Diseño de un sistema embebido para alarmas de seguridad vehicular por medio de tecnología de Sistema de Posicionamiento Global Tracker.

Que como parte de los requisitos para obtener el grado de Ingeniero en Automatización

Presenta:

Elías Gabriel Nápoles Ramos

Dirigido por:

Dr. José Gabriel Ríos Moreno

Dr. José Gabriel Ríos Moreno
Presidente

_____ Firma

Dr. Mario Trejo Perea
Secretario

_____ Firma

Dr. Roberto Valentín Carrillo Serrano
Vocal

_____ Firma

Dr. José Manuel Álvarez Alvarado
Suplente

_____ Firma

Nombre y Firma
Dr. Manuel Toledano Ayala

Centro Universitario
Querétaro, Qro.
Fecha 24 de marzo del 2023
México

RESUMEN

Uno de los principales problemas a nivel mundial es el hurto de vehículos particulares y la incapacidad, por sistemas de seguridad, para otorgar soluciones dentro de la misma. En el año 2020 más de 248 976 vehículos de motor fueron identificados como robados a nivel mundial. Durante el transcurso del año 2018 al 2021, en México, se reportaron 42, 476 vehículos robados. El objetivo de este trabajo es diseñar e implementar un sistema embebido de alarmas de seguridad vehicular por medio de tecnología de Sistema de Posicionamiento Global (GPS), Sistema Global de Comunicaciones Móviles (GSM) y Bluetooth para evitar el robo de vehículos. Por el cual se mantiene comunicación directa entre el vehículo y un dispositivo móvil, incorporando distintos sistemas de control manipulados por el usuario desde su dispositivo móvil. Se emplea un sistema embebido, para realizar las labores de comunicación, control, procesamiento de datos y alimentación requerida por el prototipo. El sistema embebido posee microcontroladores y diversos módulos de comunicación satelital para mejorar sus características de comunicación. De esta manera se pesquisa producir una labor social en un entorno local, previniendo hurto y recuperando vehículos particulares. Los resultados muestran un sistema embebido enlazado a una aplicación móvil donde el usuario puede operar distintas actividades del vehículo, de la misma manera es posible conocer la ubicación del vehículo mediante un teléfono móvil. El proyecto cuenta con distintas etapas de seguridad, el cual pudiera prevenir el hurto de vehículos. El proyecto limita en temas de tiempo de respuesta y tamaño principalmente.

SUMMARY

Globally, a common trouble is the inefficiency of security systems about cars, several vehicles are stolen for this reason a year. The purpose on this paper is make an embedded system to vehicular security. This system has direct communication with a smart cellphone, adding some control systems manipulated by user from his smart cellphone. It has an electronic card, as embedded system, it makes the work of communication, control, data process and voltage source. This system has microcontrollers, satellite communication modules, to improve its communication characteristics, they make high efficiency. This way, we look for to do a social work in the local area, preventing and recover vehicles with this trouble. The project has some security phases that only the user can access to take the control of the vehicular system. The main disadvantage of this project is the response time through embedded system to user.

*A mis padres, Benjamín y Martha,
a mis hermanos, Emmanuel y Benjamín.
Por su apoyo en cada paso.*

AGRADECIMIENTOS

Primeramente, agradeciendo a la Universidad Autónoma de Querétaro por proporcionarme las facilidades para obtener una educación completa y de calidad.

De igual manera quiero agradecer a todos los maestros de la facultad por compartir de excelente manera sus conocimientos y experiencias educativas, científicas y laborales.

Agradeciendo al Dr. José Gabriel Ríos Moreno por compartir sus conocimientos, sugerencias, apoyo y fina atención durante la elaboración de este trabajo. Así como por alentarme a realizar un trabajo de esta calidad.

Por último, agradecer a mis padres Benjamín y Martha este trabaja es sólo un poco del fruto de su esfuerzo y dedicación hacia mi parte que sin sus consejos nada de esto sería posible.

ÍNDICE

RESUMEN	i
SUMMARY	ii
AGRADECIMIENTOS	iv
ÍNDICE	v
ÍNDICE DE FIGURAS	ix
ÍNDICE DE TABLAS	xiii
CAPÍTULO I	1
1. Introducción.....	1
1.1. Robo de vehículos en México	2
1.2. Robo de vehículos en Querétaro	4
1.3. Antecedentes	5
1.3.1. Diseño e implementación de rastreo de vehículo y sistema de control de robo.....	5
1.3.2. Seguridad automotriz con autorización y rastreo vía GPS.	6
1.3.3. Rastreo escalable y monitoreo en tiempo real con arquitectura para logística y transporte en terminales.	7
1.3.4. Sistema de seguridad vasado en biométricos en tiempo real con tecnología GPS y GSM.	8
1.4. Hipótesis.....	10
1.5. Objetivos	10
1.5.1. Objetivo general.....	10
1.5.2. Objetivos específicos	10
1.6 Justificación.....	10
CAPÍTULO II	13
2. Fundamentación teórica.....	13

2.1.	Sistemas de seguridad	13
2.2.	Sistemas embebidos	13
2.3.	Software embebido.....	14
2.4.	Requerimientos del sistema embebido.....	14
2.5.	Hardware de un Sistema Embebido	15
2.6.	Dualidad Software/Hardware	16
2.7.	Sistemas operativos de tiempo real	16
2.7.1.	Tiempo de respuesta	17
2.7.2.	Sensibilidad.....	18
2.7.3.	Control de usuario.....	18
2.7.4.	Tipos de RTOS	18
2.7.5.	Procesos	18
2.8.	Control.....	19
2.8.1.	Señal de control	19
2.8.2.	Planta	19
2.8.3.	Señal de salida	19
2.8.4.	Sistemas de control en lazo abierto	20
2.8.5.	Control ON/OFF	20
2.9.	Comunicación.....	21
2.9.1.	Comunicaciones inalámbricas	22
2.9.2.	Wireless Personal Area Network (WPAN)	22
2.9.3.	Wireless Local Area Network (WLAN).....	22
2.9.4.	Wireless Metropolitan Area Network (WMAN).....	23
2.9.5.	Wireless Wide Area Network.....	23
2.10.	Programación	24

2.10.1. Leguaje C.....	24
CAPÍTULO III	25
3. Metodología.....	25
3.1. Descripción del lugar a implementar	25
3.2. Diagrama de flujo del sistema.....	27
3.3. Materiales	28
3.3.1. Alimentación.....	28
3.3.2. Microcontroladores.....	30
3.3.3. Módulos de comunicación.....	32
3.4. Actuadores.....	34
3.4.1. Actuador de seguros eléctricos	34
3.4.2. Diagrama electrónico.....	36
3.5. Comunicación.....	39
3.5.1. Comunicación entre microcontrolador y Pantalla LCD	39
3.5.2. Comunicación entre microcontrolador y Módulo SIM808.....	41
3.5.3. Comandos AT módulo SIM808.....	42
3.5.4. Comunicación entre módulo SIM808 y APP en Android	43
3.6. Desarrollo de Software.....	45
3.6.1. CCS C COMPILER	45
3.6.2. PIC Kit 3.....	46
3.6.3. Android Studio.....	51
3.7. Diseño e instalación	58
3.8. Controlador.....	58
3.9. Montaje final del sistema	64
CAPÍTULO IV	66

4. RESULTADOS	66
4.1. Diseño electrónico y pruebas de comunicación	66
4.2. Visualización de información en el sistema embebido	68
4.3. Visualización de información en la APP desarrollada	74
4.4. Pruebas del sistema embebido.....	78
4.5. Pruebas de la Aplicación desarrollada	83
4.6. Instalación del sistema embebido.....	88
4.6.1. Instalación de la fuente de alimentación.....	88
4.6.2. Instalación módulo de relevadores	90
4.6.3. Conexión entre actuador de seguros y módulo de relevadores.....	91
4.6.4. Alimentación de módulo SIM808	92
4.6.5. Alimentación de la Placa perforada	95
4.6.6. Conexiones entre la placa perforada y el módulo de relevadores.....	96
4.7. Prueba de funcionamiento.....	98
4.8. Tiempo de conexión entre APP y el sistema embebido	103
4.9. Exactitud del sistema embebido.....	107
CAPÍTULO V	110
5. CONCLUSIONES.....	110
REFERENCIAS	111
ANEXOS	113

ÍNDICE DE FIGURAS

Figura 1. Estados con mayor índice de robo de vehículos en México (AMIS, 2019).	3
Figura 2. Robo de vehículos por cada 100 mil habitantes (SESNSP, 2019).....	4
Figura 3. Diagrama de bloques del sistema de control de seguridad (Ahmed, 2015).	6
Figura 4. Diagrama de comunicación (Wirarespati, 2019).	7
Figura 5. Arquitectura del sistema de seguridad (Amrou y Boulmakoul, 2019). ...	8
Figura 6. Diagrama de bloques del sistema de seguridad vehicular (Kiruthiga y Latha 2015).....	9
Figura 7. Estatus de un requerimiento del Sistema embebido.....	15
Figura 8. Diagrama de un sistema operativo de tiempo real.	17
Figura 9. Relación entre un programa y un proceso (Aceves & Ramos, 2011). ...	19
Figura 10. Representación de un proceso de control.....	20
Figura 11. Representación de un lazo abierto.	20
Figura 12. Gráfica Control ON/OFF (José Carlos Villajulca, 2019).	21
Figura 13. Características de conexiones inalámbricas (Santos, 2009).	23
Figura 14. Vehículo Aveo 2017.	25
Figura 15. Espacio destinado para el sistema embebido.	26
Figura 16. Diagrama de flujo del sistema.....	28
Figura 17. Batería del vehículo.	29
Figura 18. Convertidor de voltaje LM2596.....	30
Figura 19. Vista lateral microcontrolador PIC18F4550.....	31
Figura 20. Dispositivo Pickit3.....	31
Figura 21. Vista lateral módulo SIM808.....	32
Figura 22. Pantalla LCD 16x2.....	33
Figura 23. Actuador universal de seguros eléctricos.....	34
Figura 24. Módulo de relevadores.....	36
Figura 25. Diagrama electrónico del módulo de relevadores.....	36
Figura 26. Diagrama electrónico del sistema embebido.	37

Figura 27. Diagrama de conexiones del módulo LM2597.....	38
Figura 28. Diagrama de conexiones entre el actuador y módulo de relevadores..	38
Figura 29. Diagrama de Conexiones entre microcontrolador y pantalla LCD 16x2.	40
Figura 30. Diagrama de conexiones entre el microcontrolador y módulo SIM808 para comunicación serial.	41
Figura 31. Instalación de tarjeta SIM con módulo SIM808.	43
Figura 32. Constante de numero a enviar información.	44
Figura 33. Comando AT configurando el receptor de la información.	45
Figura 34. Inicio de la programación del microcontrolador.....	46
Figura 35. Interfaz PicKit3.	47
Figura 36. Diagrama de conexiones entre microcontrolador y dispositivo Pickit3.	48
Figura 37. Lectura del microcontrolador mediante interfaz Pickit3.....	49
Figura 38. Revisión de comunicación entre el microcontrolador e interfaz PicKit3.	50
Figura 39. Correcta comunicación entre microcontrolador e interfaz Pickit3.	51
Figura 40. Configuración de versión de dispositivos Android.....	52
Figura 41. Menu de Actividades dentro de la APP.	53
Figura 42. Pantalla de inicio de la App desarrollada.....	54
Figura 43. Actividades a realizar dentro de la App desarrollada.	56
Figura 44. Ubicación dentro de la App desarrollada.	57
Figura 45. Diseño electrónico del sistema embebido.....	58
Figura 46. Librerías y definiciones.....	59
Figura 47. Fusibles y configuración del microcontrolador.	60
Figura 48. Variables a utilizar.	61
Figura 49. Funciones a utilizar.	62
Figura 50. Inicialización de variables y configuración de entradas y salidas.	63
Figura 51. Ciclo infinito.	64
Figura 52. Conexión física de relevadores.	65
Figura 53. Conexión del sistema embebido a la alimentación.	65

Figura 54. Conexiones de prueba del sistema embebido.	66
Figura 55. Componentes del sistema embebido.	67
Figura 56. Amperaje consumido del sistema embebido.	68
Figura 57. Conexión entre el actuador y el banco de relevadores.	68
Figura 58. Encendido del sistema embebido.	69
Figura 59. Conexión exitosa entre el sistema embebido y módulo SIM808.	69
Figura 60. Diagrama de flujo para el envío de la ubicación.	70
Figura 61. Correcto encendido del GPS.	70
Figura 62. Sincronización con los satélites.	71
Figura 63. Correcta sincronización con los satélites.	71
Figura 64. Correcto apagado del módulo GPS.	71
Figura 65. Mensaje de verificación con el usuario.	72
Figura 66. Encendido módulo GSM.	73
Figura 67. Envío de información mediante módulo GSM.	73
Figura 68. Estado de espera de recepción de SMS.	73
Figura 69. Filtro de seguridad de la APP desarrollada.	74
Figura 70. Acciones a realizar por parte del usuario.	75
Figura 71. Ubicación del vehículo.	77
Figura 72. Respuestas a comando AT del módulo SIM808.	78
Figura 73. Placa de prueba para el sistema embebido.	79
Figura 74. Cordenadas del vehículo.	80
Figura 75. Enviando SMS al usuario.	80
Figura 76. Mensajes clave para la realización de actividades.	81
Figura 77. Respuesta del sistema embebido.	82
Figura 78. Colocación de usuario y contraseña.	83
Figura 79. Acceso denegado por contraseña incorrecta.	84
Figura 80. Mensajes predeterminados para el sistema embebido.	85
Figura 81. Lectura de SMS recibidos mediante la APP desarrollada.	86
Figura 82. Colocación de Pin en las coordenadas recibidas vía SMS.	87
Figura 83. Calibración del módulo LM2596.	88

Figura 84. Comprobación de salida de voltaje del módulo LM2596 con 3000rpm	89
Figura 85. Conexiones principales.	90
Figura 86. Unión de terminales entre el actuador y módulo de relevadores.	91
Figura 87. Conexión física entre relevadores y terminales del actuador.....	92
Figura 88. Conexiones mínimas del módulo SIM808.....	93
Figura 89. Conexión DC044 para alimentación del módulo SIM808.....	93
Figura 90. Conexión física de alimentación al módulo SIM808.....	94
Figura 91. Correcta alimentación del módulo SIM808.	94
Figura 92. Correcto encendido del módulo SIM808.	95
Figura 93. Diagrama de conexiones de alimentación del sistema embebido.	95
Figura 94. Conexiones físicas de alimentación del sistema embebido.	96
Figura 95. Diagrama de conexiones entre el microcontrolador y módulo de relevadores.....	96
Figura 96. Conexión física entre módulo de relevadores y señales del sistema embebido.	97
Figura 97. Cableado de las señales para el actuador.	98
Figura 98. Conexión de antenas con módulo SIM808.	99
Figura 99. Revisión de entrada de voltaje al sistema embebido.....	100
Figura 100. Comprobación de continuidad entre terminales.....	101
Figura 101. PCB con pantalla LCD 16x2.....	102
Figura 102. Prueba de encendido de placa PCB.	102
Figura 103. Conexión correcta entre placa PCB y el módulo SIM808.	103
Figura 104. Tiempo de sincronización con satélites.	104
Figura 105. Total, de coordenadas enviadas por el módulo SIM808.....	107
Figura 106. Coordenadas enviadas por el sistema embebido truncadas a 7 decimales.	108
Figura 107. Exactitud obtenida por el sistema embebido.....	109

ÍNDICE DE TABLAS

Tabla 1. Países con mayor robo de vehículos (Knoema, 2014).	1
Tabla 2. Vehículos asegurados con más índice de robo en México (Miriam Santillán, 2022).....	3
Tabla 3 Temperaturas internas de un vehículo en Querétaro.	27
Tabla 4 Características LM2596.	29
Tabla 5 Características Arduino Mega.	30
Tabla 6. Características Pickit3.	32
Tabla 7 Características SIM808.	33
Tabla 8. Características pantalla LCD 16x2.....	34
Tabla 9 Características del actuador universal de seguros automáticos.....	35
Tabla 10 Características principales del banco de relevadores.	35
Tabla 11 Comandos AT para el Módulo de comunicación SIM808.....	42
Tabla 12 Actividades de la APP desarrollada en Android Studio.....	53
Tabla 13 Componentes del Sistema Embebido.	67
Tabla 14 Mensajes codificados de la APP.	76
Tabla 15 Comandos AT y respuesta del módulo de comunicación SIM808.	79
Tabla 16. Comparación de voltaje de entrada y salida respecto a las rpm's del vehículo.	90
Tabla 17. Tiempos de sincronización satélites.....	103
Tabla 18. Tiempo de respuesta en acción Subir seguros.....	104
Tabla 19. Tiempo de respuesta en acción bajar seguros.	105
Tabla 20. Tiempo de respuesta en acción subir vidrios.	105
Tabla 21. Tiempo de respuesta en acción bajar vidrios.	106
Tabla 22. Tiempo de respuesta en acción Obtener ubicación.	106

CAPÍTULO I

1. Introducción

Una alarma vehicular es un sistema de seguridad con el objetivo de evitar robos a vehículos utilizando medios visuales y auditivos para alertar sobre un posible robo de vehículos a nivel mundial (Tixce, 2017).

El término “Robo de vehículos” tiene un concepto amplio, este puede relacionarse con venta de auto partes, tráfico de estupefacientes, personas, terrorismo, entre otras. Este tipo de delincuencia es de las más extensas a nivel internacional, según un estudio por la Interpol más de 6,300,000 vehículos fueron identificados como robados en más de 130 países (Knoema, 2022).

Como se muestra en la Tabla 1 el mayor continente con robo vehicular es América (incluye motocicletas, autobuses y camiones) un dato relevante es el hecho de que las cifras incrementan en los países de este mismo continente en comparación con los países del continente europeo que las cifras van en decadencia al transcurrir los años.

Tabla 1. Cantidad de vehículos robados por país 2021-2018 (Knoema, 2022).

Posición	País	2021	2020	2019	2018
1	Estados Unidos	765, 50	713, 10	686, 80	700, 30
2	Francia	161, 51	168, 07	169, 08	173, 23
3	Italia	156, 94	165, 65	176, 46	183, 48
4	México	115, 46	94, 52	98, 98	115, 80
5	Canadá	78, 71	78, 80	74, 01	72, 51
6	Alemania	59, 63	56, 56	58, 40	59, 52
7	Australia	51, 01	46, 54	45, 08	47, 49
8	Colombia	39, 38	35, 00	31, 67	32, 13
9	Indonesia	37, 87	38, 38	42, 16	42, 50
10	Japón	35, 95	49, 30	59, 82	72, 97

Esto es por consecuencia de medidas preventivas y correctivas que se toman en estos países. Una de estas medidas fue tomada por la Interpol mediante un grupo especializado, conformado por funcionarios de policía e investigadores del sector privado. Esta medida de seguridad solo involucra al continente europeo y algunos países del continente asiático.

El 70% de estos robos han sido sin violencia y solo el 37% de los mismos han sido recuperados, por ende, se infiere la falta de seguridad en vehículos y la incertidumbre dada por las dependencias correspondientes para la búsqueda de los mismos (Knoema, 2016).

1.1. Robo de vehículos en México

Desde el 2015 hasta el 2017 México presenta un incremento en robo de vehículos de acuerdo con un estudio realizado por la Interpol, cerrando el año 2017 con 198 mil casos, según el Sistema Nacional de Seguridad Pública (Interpol, 2018).

En México como principal motivo de robo de vehículos es la venta de auto partes a bajo costo, para los vehículos más económicos y de gran popularidad. Como sucesor es por consecuencia de delincuencia organizada, que juega un papel importante en México, como venta de vehículos remarcados (AMIS, 2019).

De acuerdo los datos del Asociación Mexicana de Instituciones de Seguros (AMIS) se reporta un incremento de 31% para este caso específico de delincuencia. Los principales estados con estas incidencias son: Estado de México, Ciudad de México, Puebla, Guanajuato, Veracruz, Hidalgo, Michoacán, Guerrero y Querétaro como se muestra en la Figura 1 (AMIS, 2019).



Figura 1. Estados con mayor índice de robo de vehículos en México (AMIS, 2019).

La Tabla 2 muestra los vehículos asegurados con más índice de robo en México, de acuerdo a la marca y modelo; se observa que los robos de vehículos son de gama media, vehículos con déficit de seguridad.

Tabla 2. Vehículos asegurados con más índice de robo en México (Miriam Santillán, 2022).

Posición	Marca	Modelo
1	Nissan	NP300
2	Nissan	Tsuru
3	Nissan	Versa
4	Chevrolet	Aveo
5	Camiones Kenworth	Kenworth
6	Volkswagen	Vento
7	Italika	Motocicleta 111-250
8	Nissan	Sentra
9	Volkswagen	Jetta
10	Honda	CR-V

1.2. Robo de vehículos en Querétaro

Para el año 2018 Querétaro estaba posicionado en el tercer estado con mayor robo de vehículos, únicamente durante los primeros 9 meses del mismo año, se tenían 218.80 vehículos robados por cada 100 mil habitantes, según la base estadística del secretario ejecutivo del Sistema Nacional de Seguridad Pública (SESNSP). En los últimos 2 años Querétaro registra un repunte en el robo de vehículos, aproximadamente el 36% de los vehículos son llevados a otros estados para ser vendidos en mercado negro o por partes como se muestra en la Figura 2.

La principal causa de robo de vehículos es por la presencia de grupos de delincuencia organizada, ulteriormente la inconformidad en torno a servicios privados y la cifra minoritaria utiliza los mismos para realizar otros actos ilícitos.

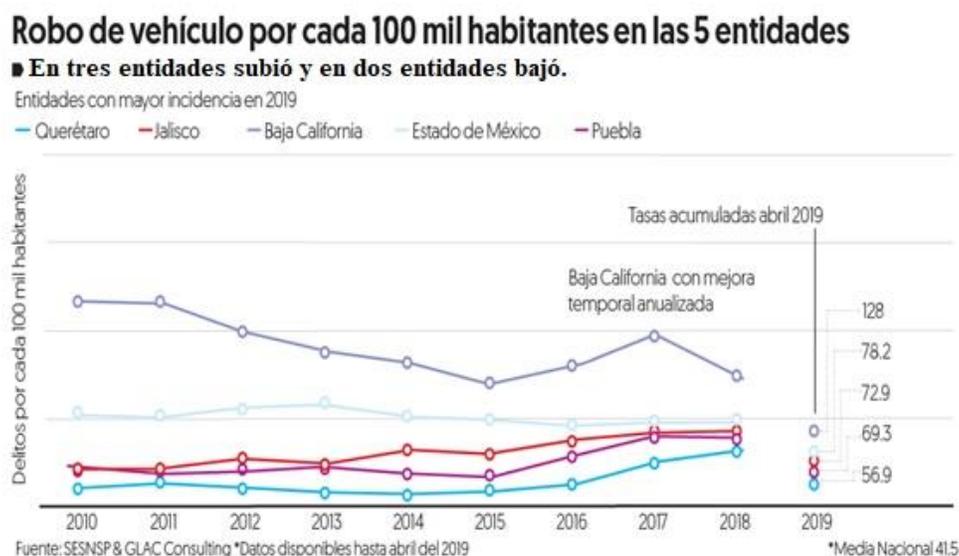


Figura 2. Robo de vehículos por cada 100 mil habitantes (SESNSP, 2019).

Durante los primeros 4 meses del año 2019 se registró el robo de 472 vehículos, esta cifra es idéntica en comparación del mismo periodo del año 2018. Por consiguiente, las medidas de seguridad tomadas por el gobierno son poco eficientes. De igual manera, esta

cifra se distribuye municipalmente de la siguiente manera: 56% Querétaro Capital, 16% San Juan del Río, 10% El Marqués, 7% Corregidora, 3% Tequisquiapan, 3% Pedro Escobedo, 1% Amealco, 1% Ezequiel Montes, 1% Colón, 1% Humilpan (SESNP, 2018).

Un estudio realizado por la AMIS, mostró un incremento, desde el 2016 hasta marzo del 2019, de aproximadamente el 50% en este delito sólo en el estado de Querétaro. Agregando que el 38% de estos casos han podido ser recuperados lo que indica que el robo de autos en México todavía es muy alto y la recuperación muy baja (AMIS, 2019).

1.3. Antecedentes

1.3.1. Diseño e implementación de rastreo de vehículo y sistema de control de robo.

(Ahmed, 2015), realizó un sistema de seguridad vehicular que consiste en 2 unidades que trabajan en conjunto; la primera unidad tiene un diseño estrecho, para ser ocultado dentro del vehículo, esta unidad contiene el Sistema de Posicionamiento Global (GPS) el cual permite rastrear para obtener su ubicación; la segunda unidad consiste en una computadora que sirve como interfaz para el usuario, la principal tarea de esta unidad es recibir, almacenar y procesar la información recibida de la unidad 1.

El uso de tecnología de canal de radio frecuencia (RFC) otorga mayor cobertura que otras tecnologías de comunicación, esto se traduce en más confianza para recuperar el vehículo por parte del usuario.

Otro agregado al diseño es el monitoreo de distintas unidades con el mismo sistema, gracias a la interfaz que otorga la segunda unidad es capaz de estar recibiendo información desde más de 2 unidades al mismo tiempo, lo cual ataca otras problemáticas para las industrias que poseen muchos vehículos privados.

En la Figura 3 se puede observar la trayectoria de información que se comparte entre las ambas unidades. Para el valor agregado que presentan en este proyecto se sugiere añadir los bloques vehiculares necesarios.



Figura 3. Diagrama de bloques del sistema de control de seguridad (Ahmed, 2015).

1.3.2. Seguridad automotriz con autorización y rastreo vía GPS.

Wirarespati y Rasjid (2019), presentaron su trabajo durante la Conferencia Internacional en Ciencias de la Computación e Inteligencia Computacional (ICCSCI), en Indonesia, el cual consiste en un sistema para vehículos enfocado al robo de los mismos. Este sistema tiene como principal operador, los microcontroladores basados en Arduino y el uso de un GPS de bajo costo, para introducirlo al mercado, y el uso de Android Studio Kit de desarrollo de software (SDK) para una aplicación móvil. Como se muestra en la Figura 4 el microcontrolador tiene constante comunicación con el GPS para obtener la posición y a su vez transmitirla al usuario. El usuario será capaz de observar la información emitida por el microcontrolador mediante la interfaz. La principal ventaja de este dispositivo es el uso de tecnología de Sistema Global de Comunicaciones Móviles (GSM) el cual evita el uso del internet y mantiene comunicación con el usuario siempre y cuando haya cobertura con alguna línea telefónica.

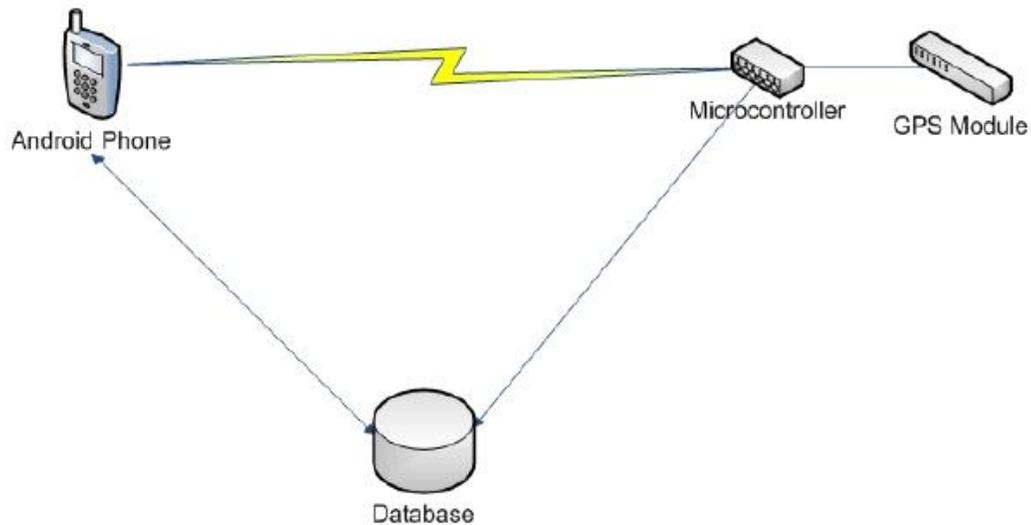


Figura 4. Diagrama de comunicación (Wirarespati, 2019).

1.3.3. Rastreo escalable y monitoreo en tiempo real con arquitectura para logística y transporte en terminales.

De acuerdo a Amrou y Boulmakoul (2019); presentaron un nuevo sistema de monitoreo y rastreo en tiempo real dirigido a vehículos de gama media y a empresas que se dedican a la logística del transporte.

Este sistema es más confiable por la implementación de tecnologías de Identificación por Radio Frecuencia (RFID), GPS. Por lo que se propone el uso de otras arquitecturas como auto – ID tecnologías, QRcode, Barcode, magnético, etc. (Amrou y Boulmakoul, 2019).

El sistema consiste en monitoreo del vehículo en tiempo real, agregando un Barcode partiendo de tecnologías ID para el reconocimiento del usuario del vehículo. Esta estrategia otorga al responsable mayor confianza y comodidad ya que al no tener una aceptación del ID, corta el suministro de energía y envía una señal al responsable para tomar medidas correspondientes. Este modelo proporciona un sistema de logística dinámica, así se pueden identificar problemas de flujo de tráfico en autopistas y reducir el tiempo de entrega. Como

se muestra en la Figura 5 la arquitectura proporciona capacidad de rastreo en tiempo real; rastrea el vehículo en cada área de la terminal a través de un portal de identificación de usuario. Este sistema desarrolla y mejora el proceso de logística aportando seguridad vehicular y conocimiento de las áreas del mismo.



Figura 5. Arquitectura del sistema de seguridad (Amrou y Boulmakoul, 2019).

1.3.4. Sistema de seguridad basado en biométricos en tiempo real con tecnología GPS y GSM.

Kiruthiga y Latha (2015), trabajaron en un sistema de seguridad para vehículos, el prototipo propuesto tiene como objetivo el crear un sistema de seguridad capaz de proteger un vehículo de una persona sin autorización de utilizar el vehículo mediante reconocimiento de huella dactilar. Simultáneamente comunicar al propietario la ubicación del vehículo mediante tecnología GPS, GSM. Todo esto utilizando microcontroladores embebidos para un costo lo más bajo posible. Así como proporcionando una unidad de control de carga por si el ladrón corta o desconecta el suministro de energía, de esta manera el sistema de seguridad puede trabajar con carga o sin carga y conocer su posición. En la Figura 6 se muestra la comunicación entre el microcontrolador y los módulos de información.

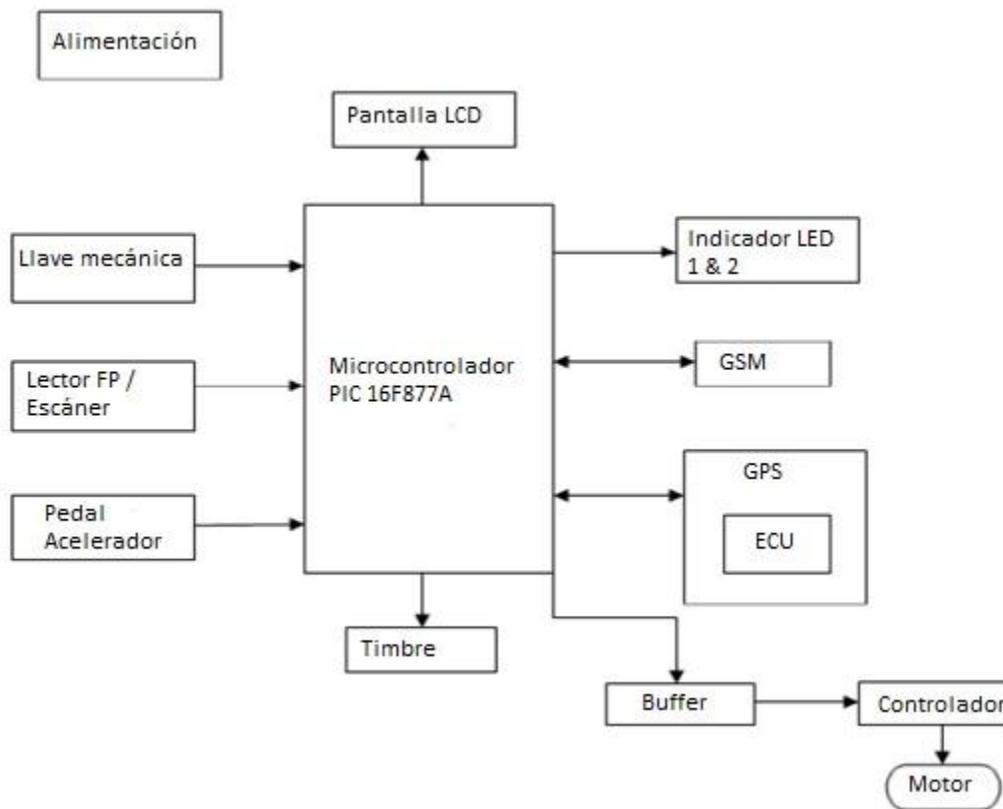


Figura 6. Diagrama de bloques del sistema de seguridad vehicular (Kiruthiga y Latha 2015).

Los fabricantes de automóviles y los proveedores se centran en tecnologías, las cuales son muy similares en todas las marcas, el reto está en mejorar la seguridad de los vehículos, estas industrias están buscando innovaciones para tener un valor agregado a diferencia de la competencia, en una era en la que la seguridad de evitar el robo del vehículo determinará la confiabilidad y decisión de compra, la innovación en esta rama de la ingeniería hará la diferencia de mercado mejorando considerablemente la economía y confiabilidad de los usuarios al evitar el robo de sus vehículos.

1.4. Hipótesis

La integración de un dispositivo de ubicación y control de vehículo mediante tecnología GSM en un sistema embebido, presenta una respuesta de menor tiempo y menor costo económico al usuario para una acción en fin de evitar el robo de sus vehículos en comparación con la alarma Coban TK403B y COMPUSTAR CSX2WQ900A, mejorando la confiabilidad de los sistemas de seguridad. Esto refiere a ventajas del usuario contra la delincuencia.

1.5. Objetivos

1.5.1. Objetivo general

Diseñar e implementar un sistema embebido de alarmas de seguridad vehicular por medio de tecnología de Sistema de Posicionamiento Global (GPS), Sistema Global para las Comunicaciones Móviles (GSM) y Bluetooth para evitar el robo de vehículos.

1.5.2. Objetivos específicos

- a. Obtener la ubicación utilizando tecnología GPS mediante comandos AT con uso del sistema embebido para el uso de dicha información.
- b. Realizar una aplicación móvil Android como interfaz de usuario para la interpretación de los datos recibidos desde el vehículo.
- c. Diseñar una interface mediante el módulo bluetooth para el control de seguros y vidrios como prueba para la seguridad.
- d. Enviar información desde el vehículo mediante módulo GSM utilizando comandos AT con uso del sistema embebido para manejo de la información.

1.6 Justificación

El uso de vehículos privados es una tarea común en nuestra nación ya sea para traslado o como medio de trabajo. En México el 87% de las familias tienen al menos 1 vehículo para su uso particular esto describe gran parte del mercado de transporte y la importancia que existe dentro de nuestra nación. Sin embargo, otro tema de importancia para México es el hurto de vehículos y la baja seguridad que existe para los mismos. Gran parte

del porcentaje de los vehículos en México es de gama media o baja, mismos que son los vehículos con menor seguridad dentro de toda la gama del sistema de protección, con estos datos obtenemos un gran problema, simultáneamente se tiene una ventana de oportunidad para desarrollar un dispositivo de seguridad dirigido para este mercado en particular. Partiendo del costo competitivo, valorar incluir esta alarma para autos usados e impactar de manera más rotunda en el robo de vehículos.

La implementación de un sistema embebido de alarmas de seguridad vehicular por medio de tecnología de GPS evitará en gran medida el robo de vehículos y con ello poder prevenir este tipo de delitos; otorgando al usuario seguridad y el respaldo de su inversión, lo que representará una mejora en su economía, evitando el hurto de su unidad o en su defecto la recuperación pronta del mismo.

En la Tabla 3 se muestran las principales características del sistema embebido y su comparación con alarmas de características similares junto a su precio correspondiente. Con este proyecto prevee un impacto social positivo a los vehículos cuya seguridad sea insuficiente para los usuarios, otorgando una solución alcanzable económicamente. Apoyando a la sociedad en uno de los principales incidentes que ocurren dentro del país. Utilizando dispositivos electrónicos de uso diario para la mayor parte de la población dentro del país, asimismo, otorgando información específica para la recuperación del mismo.

Tabla 3. Comparación precio y características con alarmas en el mercado

Marca	Modelo	Costo	Características
	Sistema embebido	\$1,237.52	-GPS -GSM -GPRS -APP
Coban	TK403B	\$1,469	-GPS -APP -Alarma de movimiento
COMPUSTAR	CSX2WQ900A	\$6,251	-Control LCD -GPS -Sirena -APP con costo -Control de arranque
Coban	4G	\$1,599	-GPS -GSM -Control remoto -Sirena
Coban	TK303G	\$1,798	-GPS -APP -GSM-Control remoto

CAPÍTULO II

2. Fundamentación teórica

2.1. Sistemas de seguridad

Los sistemas de seguridad son grupos de elementos instalados e intercomunicados entre sí que previenen, detectan o actúan ante intrusiones, intentos de robos y otros eventos como incendios.

Tradicionalmente, se asocian los ‘sistemas de seguridad’ con ‘alarmas antirrobo’; sin embargo, los sistemas de alarmas son la solución moderna de mecanismos de protección que el ser humano siempre ha implementado.

Estos conjuntos de dispositivos varían mucho dependiendo de las características del lugar a proteger, el presupuesto y si cuentan además con planes de atención y monitoreo las 24 horas.

Además, los sistemas de seguridad pueden centrarse en proteger inmuebles, personas y salvaguardar procesos industriales. En su concepto más general, estos sistemas están diseñados desde la protección de una vivienda común y corriente hasta controlar los fallos en cuartos de máquinas.

2.2. Sistemas embebidos

Se puede definir los sistemas embebidos como un sistema electrónico de procesamiento, programado para realizar funciones o tareas para un objetivo específico. Cuando los sistemas son más grandes, se pueden incluir además de elementos electrónicos y de software, partes mecánicas, eléctricas y electromecánicas (Aceves & Ramos, 2011).

En términos de limitaciones de hardware, esto significa las limitaciones en el desarrollo del procesamiento, consumo de alimentación, memoria y funcionalidades. En

términos de software, las limitaciones comúnmente son relativas al PC, como aplicaciones (Nooergard, 2013).

2.3. Software embebido

El software embebido difiere del software convencional de una computadora por las siguientes características (Aceves & Ramos, 2011):

- Los sistemas embebidos pueden ser de tiempo real.
- Tienen una interfaz directa con el hardware del dispositivo.
- Suele considerarse las pruebas de hardware como parte del desarrollo del sistema embebido.
- Suele haber más control del tiempo de respuesta, manejo de recursos como memoria.
- Requiere recursos humanos sumamente especializados en las áreas informáticas, electrónicas, entre otras.

2.4. Requerimientos del sistema embebido

Existen distintos tipos de requerimientos. De esta manera simple, se pueden separar los requerimientos en funcionales y no funcionales. El estándar para los sistemas de tiempo real de acuerdo al Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) estándar 830 define los siguientes tipos de requerimientos (Aceves & Ramos, 2011):

- Funcionales.
- Interfaces externas.
- Rendimiento.
- Bases de datos lógicas.
- Limitaciones de diseño.

Además, los requerimientos funcionales incluyen una descripción de todas las entradas del sistema y la secuencia de operaciones asociada con cada set de entradas. En este sentido, la secuencia exacta de las tareas y las respuestas del sistema deben ser listadas para cada entrada posible.

Esto debe realizarse incluso para situaciones anormales. Las situaciones anormales pueden ser manejo de errores y como se recupera el sistema de ellos. Para optimizar el sistema embebido y evitar errores en la Figura 7 se puede apreciar el diagrama de flujo que sigue para realizar apropiadamente el sistema embebido.

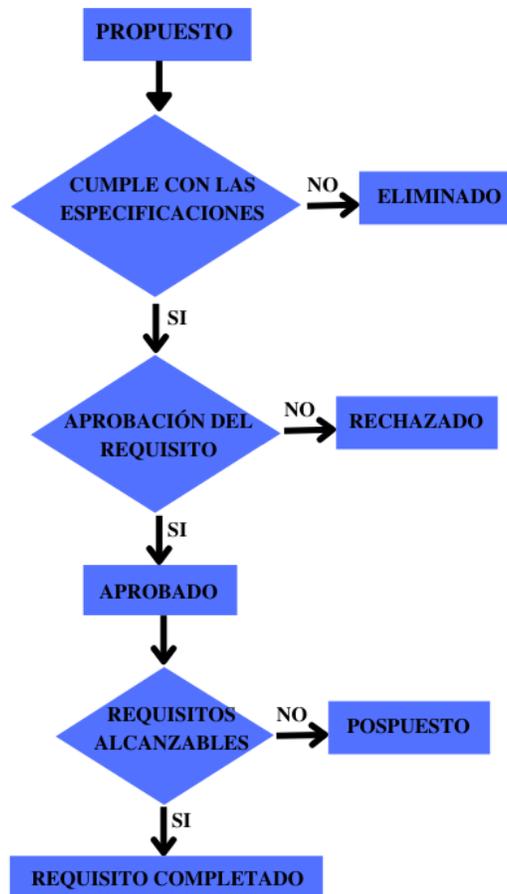


Figura 7. Estatus de un requerimiento del Sistema embebido.

2.5. Hardware de un Sistema Embebido

Diseñar un hardware para un sistema embebido es más que seleccionar un procesador “poderoso” y colocarlo a unos periféricos. La decisión de como particionar el

diseño de tal forma que no pierda su funcionalidad tanto de parte del software como del hardware es fundamental en el diseño (Aceves & Ramos, 2011).

Existen una serie de consideraciones para poder elegir los elementos de hardware, además de los criterios a seguir para la selección del microprocesador a usar. Dada la gran variedad de fabricantes, para la mayoría de las aplicaciones existe más de una solución. Se recomienda tomar en cuenta los siguientes criterios para la elección de hardware:

- Número de entradas/salidas.
- Interfaces requeridas.
- Consideraciones de tiempo real.
- Velocidad de procesamiento.
- Consumo de energía.
- Ciclo de vida.

2.6. Dualidad Software/Hardware

Realizar la partición entre software y hardware no solo es posible, sino necesaria debido a la dualidad entre software y hardware.

Idealmente, la decisión de particionar no debe de realizarse hasta que se entienda todas las maneras alternativas de resolver el sistema. Mientras más tiempo pase para hacer esa decisión, más factible será para saber que partes del sistema se pueden implementar en hardware y cuales en software (Aceves & Ramos, 2011).

La decisión de particionar el hardware y el software es un problema de optimización compleja. Muchos diseños de sistemas embebidos requieren de las características que deben ser tomadas en cuenta para realizar la partición:

- Costo software/hardware.
- Estándares.
- Dueño de la tecnología.
- Competitividad.

2.7. Sistemas operativos de tiempo real

Los sistemas operativos de tiempo real (RTOS) son sistemas informáticos que interactúan repetidamente con su entorno físico y responden dentro de un plazo de tiempo determinado. Es decir, los RTOS exigen un comportamiento predecible ya que se requiere reaccionar a un estímulo del medio ambiente dentro de los parámetros definidos para la respuesta. “Tiempo real” no se refiere a inmediatamente; sin embargo, está relacionado con la velocidad de respuesta del sistema en cuestión (Aceves & Ramos, 2011).



Figura 8. Diagrama de un sistema operativo de tiempo real.

Los sistemas operativos de tiempo real aíslan las tareas y los procesos, permitiendo a cada desarrollador de un grupo a escribir código como si tuviera uso exclusivo del hardware y sus recursos. Sin embargo, el diseño en su conjunto para un sistema embebido debe de considerar los problemas adherentes de datos compartidos y multiprocesos. Los RTOS deben ser personalizados para el hardware en el cual será implementado.

Por lo regular Sistema Operativo de tiempo real suele tener la misma arquitectura general que un Sistema Operativo convencional, pero su diferencia radica en que proporciona mayor prioridad a los elementos de control y procesamiento que son utilizados para ejecutar los procesos o tareas. En la Figura 8 se aprecia el lazo de un sistema operativo en tiempo real.

2.7.1. Tiempo de respuesta

Los sistemas operativos de tiempo real deben responder a estímulos externos en el ambiente dentro de un predeterminado límite de tiempo. Los RTOS deben de producir un resultado correcto y realizarlo a tiempo. Esto implica que el tiempo de respuesta es tan importante como lo correcto de los resultados (Aceves & Ramos, 2011).

Los RTOS deben ser diseñados para poder cumplir con estas expectativas de tiempo. Para poder responder ante estas expectativas, la correcta decisión del diseñador entre los procesos que deben desarrollar el software y el hardware es crucial.

2.7.2. Sensibilidad

La sensibilidad es el tiempo que consume el sistema operativo de tiempo real en dar servicio a la interrupción incluye (Aceves & Ramos, 2011):

- Tiempo necesario para iniciar la gestión de la interrupción y comenzar la ejecución de su rutina de tratamiento (ISR).
- Tiempo para ejecutar la ISR.
- El efecto del anidamiento de interrupciones.

2.7.3. Control de usuario

El control de usuario es generalmente mucho mayor en un sistema operativo de tiempo real que en un sistema operativo ordinario esto se debe a que en un sistema en tiempo real resulta esencial permitir al usuario un control preciso sobre la prioridad de las tareas. El usuario debe poder distinguir entre tareas rígidas y flexibles, a su vez especificar prioridades relativas dentro de cada clase (Aceves & Ramos, 2011).

2.7.4. Tipos de RTOS

Un sistema en tiempo real debe satisfacer las restricciones de tiempo de respuesta sin sufrir la degradación del rendimiento de sistema. Si el sistema sufre la degradación del rendimiento pero no falla, el sistema se conoce como sistema de tiempo real blando. De lo contrario, si el sistema falla, al sistema se le conoce como tiempo real duro, crítico (Aceves & Ramos, 2011).

2.7.5. Procesos

Un proceso es la unidad básica de programación que un sistema operativo puede controlar. Cada sistema operativo define un proceso de manera diferente. Un proceso implementa una función computacional (Aceves & Ramos, 2011).

Un programa es simplemente una secuencia estática de instrucciones. Sin embargo, la ejecución de instrucciones es dinámica en el sentido en el que diversas propiedades cambian con respecto al tiempo en el que se ejecutan y la secuencia de instrucción que se le da al hardware. En este sentido, un proceso es creado por el RTOS para encapsular toda la información involucrada en la ejecución del programa. En la Figura 9 se muestra como el programa es solo una parte de la tarea.

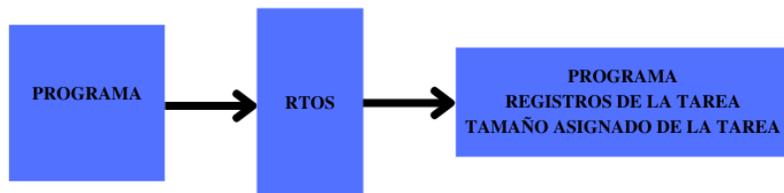


Figura 9. Relación entre un programa y un proceso (Aceves & Ramos, 2011).

2.8. Control

2.8.1. Señal de control

También conocida como entrada, la señal de control es la cantidad o condición que el controlador modifica para efectuar el valor de la variable controlada.

2.8.2. Planta

Una planta puede ser una parte de un equipo, una máquina cuya razón de ser es efectuar una operación particular para modificar la variable controlada.

2.8.3. Señal de salida

Esta señal o variable es la respuesta deseada del sistema. Lo que se busca en control es que esta señal o variable debe ser semejante a la señal de entrada, teniendo esta condición se puede considerar una respuesta correcta del sistema mediante el sistema de control tal como se ilustra en la Figura 10.



Figura 10. Representación de un proceso de control.

2.8.4. Sistemas de control en lazo abierto

Los sistemas en los cuales la salida no tiene efecto sobre la acción de control se denominan sistemas de control en lazo abierto (Ogata, 2010). En un sistema de control de lazo abierto no se mide la salida ni se realimenta para compararla con la salida. Así, a cada entrada de referencia le corresponde una condición de operación fija; como resultado de ello, la precisión del sistema depende de la calibración esto puede ser visualizado en la Figura 11.

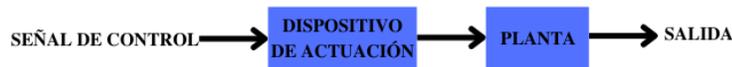


Figura 11. Representación de un lazo abierto.

2.8.5. Control ON/OFF

Este tipo de controlador, también llamado Todo o Nada, usa un algoritmo simple para solamente revisar si la variable de proceso está por encima o por debajo de un SetPoint determinado. En términos prácticos, la variable manipulada o la señal de control cambia entre “totalmente ON” o totalmente OFF, sin estados intermedios. Este tipo de accionamiento provoca un control muy impreciso de la variable de proceso como se observa en la Figura 12.

Solo algunos pocos procesos industriales reales se podría usar este tipo de control, como por ejemplo el control de nivel de un tanque en el que simplemente el objetivo es evitar derrames o evitar que se vacíe completamente, con lo que se pondría un Up SetPoint (USP) de 90% y un Low SetPoint (LSP) de 20% por ejemplo.

No solo es un problema de precisión, sino que este tipo de control usado para una banda de control muy pequeña o un solo SetPoint en específico fatigaría al elemento final de control, por ejemplo, desgaste de componentes mecánicos. Otro factor es la pérdida de energía que se produce porque constantemente se estaría calentando el fluido a temperaturas mayores a la necesaria. Así mismo, este sobre calentamiento o sobre-enfriamiento indudablemente afectara la calidad del producto final, otro punto de tener en cuenta.

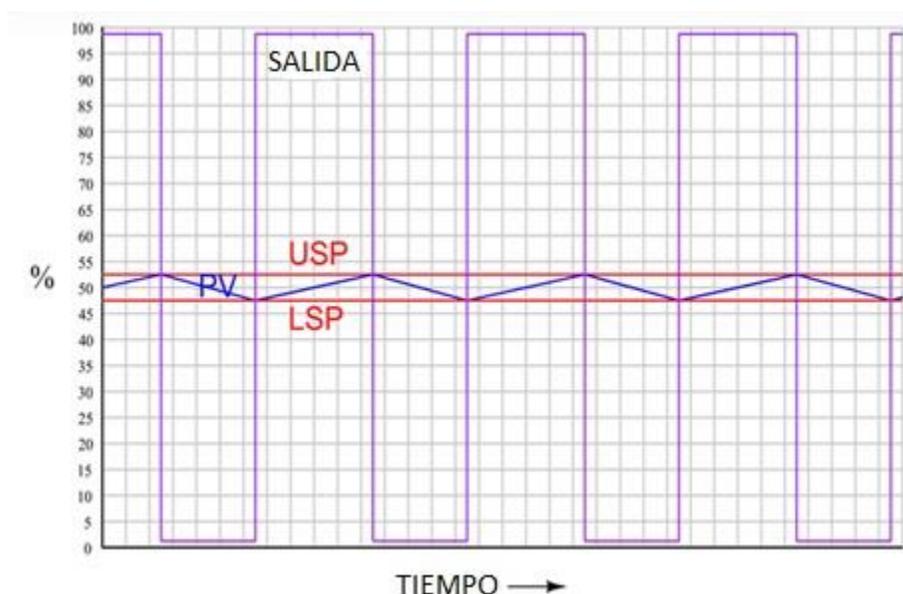


Figura 12. Gráfica Control ON/OFF (José Carlos Villajulca, 2019).

2.9. Comunicación

Actualmente, para establecer una comunicación entre dos puntos se pueden emplear, en primera instancia, dos alternativas: una conexión mediante cable o bien una tecnología inalámbrica. Cada opción tiene sus ventajas y sus inconvenientes y la elección depende de la aplicación concreta (Santos, 2009).

Sin embargo, desde hace tiempo, la alternativa más común para realizar una comunicación entre un origen y un destino ha sido el cable. Aunque actualmente la transmisión por cable sigue siendo muy usada, la alternativa de la transmisión inalámbrica va ganando terreno y es una solución cada vez más utilizada.

2.9.1. Comunicaciones inalámbricas

Debido al creciente desarrollo de las comunicaciones inalámbricas y la reducción en el coste de los dispositivos que incorporan esta tecnología, han surgido, en un corto intervalo de tiempo, una gran cantidad de iniciativas. Esto ha dado lugar a la aparición de multitud de tecnologías cuyo principal objetivo es permitir el intercambio de información entre dispositivos inalámbricos y que se diferencian unas de otras por la tasa de transmisión y el rango de alcance.

El estándar IEEE 801.11ac establece que la comunicación inalámbrica opera en la banda de los 5 GHz con un máximo ancho de banda de canal de 160 MHz. El ancho de banda es el rango de frecuencias en el que se concentra la mayor parte de la potencia de la señal. A pesar de los avances obtenidos con 801.11ac, su banda de frecuencias constituye un recurso limitado pues su espectro es utilizado por otras aplicaciones (Leyva, 2016).

2.9.2. Wireless Personal Area Network (WPAN)

Permiten a los usuarios establecer comunicaciones inalámbricas entre dispositivos (PDA, portátiles, teléfonos móviles, etc.) en un área reducida. Las tecnologías inalámbricas cuyas características permiten la construcción de redes WPAN son las siguientes:

- Bluetooth.
- Banda Ultra Violeta (UWB).
- Zigbee.
- Identificación de Radio Frecuencia (RFID).

2.9.3. Wireless Local Area Network (WLAN)

Permiten a los usuarios establecer conexiones inalámbricas dentro de un área local (edificio corporativo, campus empresarial, aeropuerto, etc.). Se pueden utilizar en lugares donde la instalación de un cableado extenso es prohibitiva, o bien para complementar a una red local de cable existente de manera que los usuarios puedan trabajar en diferentes lugares dentro de un edificio. Las tecnologías principales WLAN son: Fidelidad sin cables o inalámbrica (WiFi).

2.9.4. Wireless Metropolitan Area Network (WMAN)

Permite a los usuarios establecer conexiones inalámbricas entre varias ubicaciones dentro de un área metropolitana, evitando el coste de instalación de cables de cobre o fibra y el alquiler de las líneas. Se pueden destacar tres familias de tecnologías WMAN:

- Worldwide Interoperability for Microwave Acces (WiMAX).
- Manegement By Wandering Around (MBWA.).

2.9.5. Wireless Wide Area Network

Permiten a los usuarios establecer conexiones inalámbricas a través de redes remotas públicas o privadas. Estas conexiones se pueden mantener a través de zonas geográficas extensas, como ciudades o países, mediante el uso de antenas en varias ubicaciones o sistemas satélite. En función de la altitud del satélite, se pueden definir varias redes, como se muestra en la Figura 13:

- Satélites geoestacionarios.
- Satélites de baja órbita.
- Satélites de media órbita.

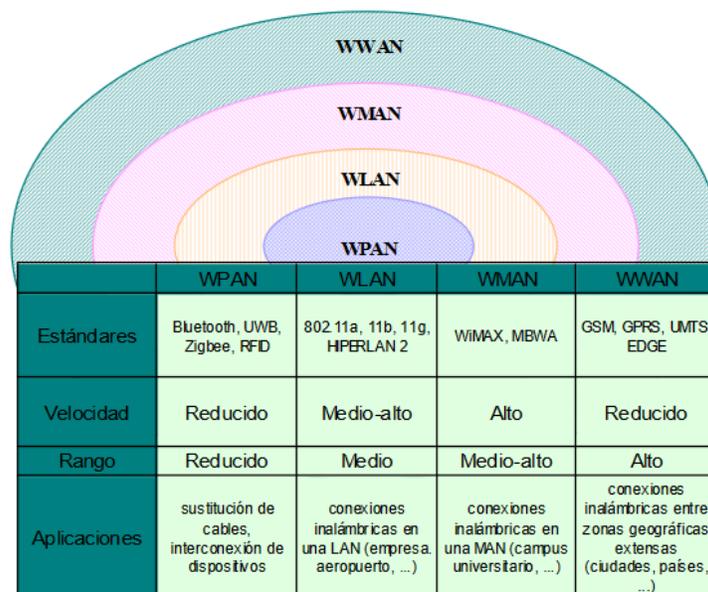


Figura 13. Características de conexiones inalámbricas (Santos, 2009).

2.10. Programación

De acuerdo con la naturaleza del funcionamiento de las computadoras, se dice que estas siempre ejecutan órdenes en un formato que les resulta inteligible; dichas órdenes se agrupan en programas, conocidos como software, el cual, para su estudio, a su vez, se divide en dos partes: el formato de representación interno de los programas, que constituye el lenguaje máquina o código ejecutable, y el formato de presentación externa, que es un archivo o un conjunto de archivos, que puede o no estar en un formato que puede ser visto/leído por el usuario (Juganaru, 2014).

2.10.1. Leguaje C

El lenguaje C es un lenguaje para programadores en el sentido de que proporciona una gran flexibilidad de programación y una muy baja comprobación de incorrecciones, de forma que el lenguaje deja bajo la responsabilidad del programador acciones que otros lenguajes realizan por sí mismos. El lenguaje C no es rígido en la comprobación de tipos de datos, permitiendo fácilmente la conversión entre diferentes tipos de datos y la asignación entre tipos de datos diferentes (Bonet, 2005).

CAPÍTULO III

3. Metodología

Como primer paso en realizar un sistema embebido se estableció como base vehículos de gama baja y media que puedan ser adaptados para la implementación del sistema embebido, por practicidad se utilizó un vehículo de marca Chevrolet, modelo Aveo de año 2013 como se puede observar en la Figura 14. Con base en el mismo se pudo diseñar, y programar un sistema embebido capaz de ser controlado a distancia en tiempo real utilizando como actuadores los propios del vehículo. Este vehículo fue elegido a conveniencia gracias a sus cualidades de seguridad que a continuación se enlistan:

- Seguridad propia de fábrica baja.
- Seguros eléctricos.



Figura 14. Vehículo Aveo 2017.

Debido a las propiedades del vehículo, no se realizó ninguna implementación para los seguros. Por lo que se implementó un sistema de control de lazo abierto para el control de seguros.

3.1. Descripción del lugar a implementar

El desarrollo del sistema se diseñó de manera que pueda ser implementado en el lugar mayormente conveniente para el usuario. Para este caso en particular se utilizó como espacio apropiado en la guantera del copiloto como se puede observar en la Figura 15. Por su facilidad de instalación, su cercanía con los actuadores necesarios para el sistema y por la

ubicación de la alimentación. A su vez se realizó un análisis de temperatura dentro del vehículo para seleccionar los materiales que se pueden utilizar dentro del mismo sin ser afectada la parte operativa del sistema.



Figura 15. Espacio destinado para el sistema embebido. Estas mediciones de temperatura se tomaron dentro del vehículo completamente cerrado durante la etapa de verano para considerar las temperaturas más altas dentro del mismo, se tomaron en el mismo sitio durante distintas horas del día, estas se muestran en la Tabla 4. Gracias a esta misma se concluyó que el lugar es apropiado para la colocación del sistema embebido, debido a que las temperaturas internas del vehículo no exceden las temperaturas máximas de operación de los componentes electrónicos que se emplearon en el sistema. Otro aspecto importante que se consideró fueron las dimensiones que tiene el lugar para la instalación del sistema embebido, la guantera del copiloto cuenta con unas dimensiones de 42 cm de ancho por 23cm de largo y 22cm de profundidad por ende tuvo una factibilidad mayor para colocación del sistema embebido. Como último punto a considerar en el sitio de implementación fue el material del sitio, ya que para evitar daños electrónicos se evitó estar cerca de algún metal que pudiese producir corto circuito o que llegase a afectar en la señal necesaria para los módulos GPS y GSM. La guantera cuenta con un diseño completo de materiales plásticos genera un aislante directo a los componentes del sistema embebido por lo cual el sitio donde fue montado fue uno de los más convenientes para su instalación.

Tabla 4. Temperaturas internas de un vehículo en Querétaro.

Hora	Temperatura
09:00	20°C
12:00	29.4°C
14:00	31.1°C
16:00	29.0°C
18:00	23.9°C

3.2. Diagrama de flujo del sistema

Previo a elegir los componentes a utilizar para el sistema embebido se realizó un diagrama de flujo que pueda representar y visualizar cada etapa del sistema. En este punto fue necesario contemplar los alcances y objetivos anteriormente establecidos para el sistema embebido. Como se ilustra en la Figura 16 se desglosa cada etapa del sistema. Como primer paso se contempló el inicio del sistema, posteriormente la inicialización de una pantalla con la cual sea visible cada etapa posterior del sistema. Como consiguiente fue la configuración de entradas y salidas necesarias para el sistema embebido en el cual se contempló los módulos a utilizar. Como siguiente paso es la comprobación de la conexión correcta de los módulos a utilizar. Una vez comprobando la conexión con los módulos se consideró un mensaje de aviso al usuario donde se le especifica la conexión correcta del sistema embebido. Y como último punto fue un ciclo donde se busca la existencia de algún mensaje no leído para realizar la acción correspondiente.

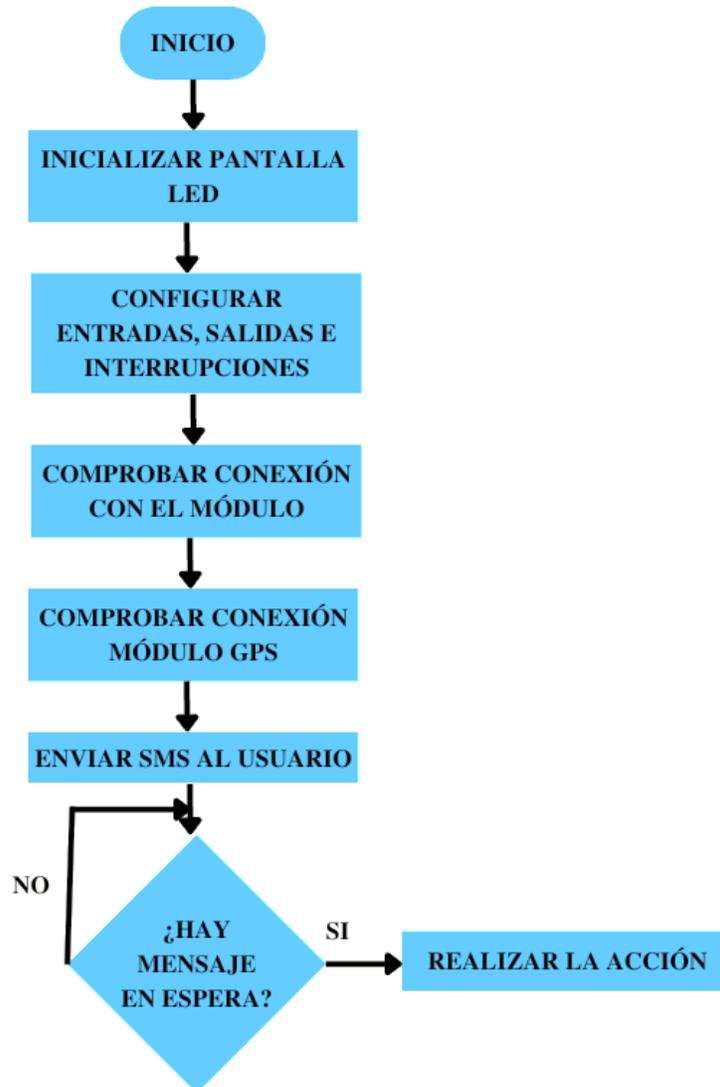


Figura 16. Diagrama de flujo del sistema.

3.3. Materiales

3.3.1. Alimentación

La alimentación del sistema embebido se tomó de la batería eléctrica del vehículo usado, es una batería marca GONHER con salida de 12 V de corriente directa con potencia de 720Wh como la que se muestra en la Figura 17. Debido a estas características no exigió alguna otra fuente de alimentación para compensar potencia o voltaje. Dicha batería es de venta libre al público. Para este punto específico no es necesario contemplar la marca de la misma, siempre y cuando se cumplan con las especificaciones para mantener una

alimentación constante del sistema embebido sin afectar su carga necesaria para su uso al encender el vehículo.



Figura 17. Batería del vehículo.

Se usó el módulo convertidor de voltaje LM2596. El voltaje de salida se selecciona mediante un potenciómetro multivuelta. Gracias a este módulo se ha reducido el espacio del sistema. Se tuvo la facilidad de propiamente reducir el voltaje de entrada al sistema embebido sin problemáticas de formas de señal debido a que la fuente de alimentación es de corriente directa al igual que la necesaria para los componentes electrónicos del sistema embebido. Todos los componentes electrónicos empleados a este sistema utilizan la misma cantidad de voltaje por lo que se usó aventajadamente este módulo y se dispuso únicamente de uno, aunado a esto el módulo limita su voltaje de salida a lo establecido por el potenciómetro multivuelta sin considerar variaciones mayores voltaje, esto implicó una ventaja en la salida de voltaje sin importar los picos de voltaje al encender el vehículo o al acelerar el vehículo. En la Tabla 5 se puede apreciar sus principales características de este módulo, asimismo, en la Figura 18 se muestra el convertidor de voltaje mediante una imagen.

Tabla 5. Características LM2596.

Característica	Valor
Rango de alimentación de voltaje	4.5 a 40 V
Rango voltaje de salida	1.23 a 37 V

Amperaje de salida	3 A
Dimensiones	48.5 x 23mm

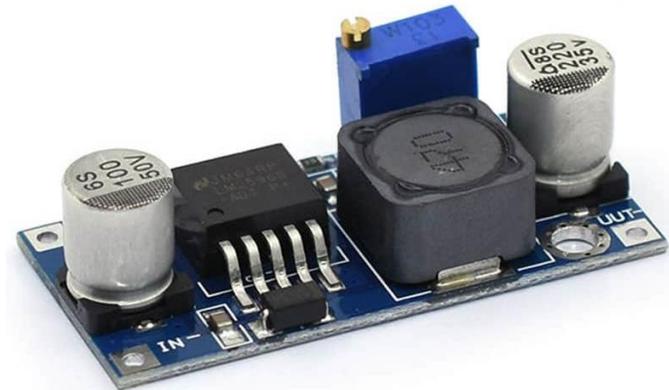


Figura 18. Convertidor de voltaje LM2596.

3.3.2. Microcontroladores

Se usó un PIC18F4550 como microcontrolador del sistema, con el mismo se redujeron costos, espacio e implementación de circuitos electrónicos varios en otra tarjeta electrónica. El propósito del mismo es tener el control de la información recibida mediante los módulos de comunicación. Asimismo, del control para los actuadores dentro del vehículo. En la Tabla 6 se muestran las características del microcontrolador a detalle.

Tabla 6. Características Arduino Mega.

Característica	Valores
Voltaje operativo	5[V]
Pines digitales	40 pines
Pines analógicos	16
PWM	15 pines de salida PWM
Voltaje de salida en cada pin	5[V]
Corriente de salida en cada pin	40[mA]
Reloj	8[MHz]
EEPROM	4[kB]

En la Figura 19 se muestra visualmente el microcontrolador utilizado para el sistema embebido.



Figura 19. Vista lateral microcontrolador PIC18F4550.

Para la programación del microcontrolador fue de alta importancia contar con un programador capaz de soportar la información y el modelo del microcontrolador a utilizar. Dentro del mercado existen muchos programadores con la capacidad de ser compatibles con el microcontrolador elegido, sin embargo, para este proyecto se decidió por un programador PICKit-3 como se muestra en la Figura 20.



Figura 20. Dispositivo Pickit3.

En la Tabla 7 se pueden apreciar sus principales características por las cuales fue elegido este programador.

Tabla 7. Características Pickit3.

Característica	Valores
Voltaje operativo	5[V]
Voltaje de salida	1.8 – 5[V]
Dimensiones	60 x 36 x 3.5mm
Comunicación	Estándar ICSP

3.3.3. Módulos de comunicación

Debido a la gran variedad de módulos de comunicación se decidió por los más comerciales, con bajas dimensiones físicas, exactitud en respuesta y de costo accesible para mejorar el costo del proyecto y su implementación en el vehículo. El primer módulo de comunicación es el SIM808 dentro de este módulo se contienen los encapsulados de los módulos GPS, GPRS y GSM. Este módulo otorgó al microcontrolador la ubicación en tiempo real del módulo GPS. También tuvo la tarea de recibir comunicación vía GSM y transferirla al microcontrolador. En la Figura 21 se muestra el módulo de comunicación utilizado, asimismo.



Figura 21. Vista lateral módulo SIM808.

En la Tabla 8 se detallan sus respectivas características.

Tabla 8. Características SIM808.

Características	Valor
Dimensiones	40[mm] x 55[mm]
Voltaje de entrada	3.3[V] a 5[V]
Interfaces	UART / I2C
Precisión	2.5[m]
Control	Comandos AT

Para la interfaz entre el sistema embebido y el usuario es necesario agregar una pantalla de comunicación unidireccional. Contemplando el precio, su competitividad en el mercado y su tamaño se optó por usar una pantalla LCD 16x2 como la que se observa en la Figura 22.



Figura 22. Pantalla LCD 16x2.

Esta pantalla tiene la practicidad de ser compatible con el microcontrolador a utilizar y debido a sus características resulta mayormente factible para el sistema embebido. La Tabla 9 se detallan las características principales de la pantalla LCD 16x2.

Tabla 9. Características pantalla LCD 16x2.

Características	Valor
Dimensiones	80[mm] x 36[mm]
Voltaje de entrada	3.3[V] a 7[V]
Rango de temperatura	0 a +50 °C
Pines de Control	14 pines

3.4. Actuadores

3.4.1. Actuador de seguros eléctricos

El vehículo cuenta con seguros de puertas electrónicos los cuales no son propios de agencia, los mismos fueron instalados posteriormente. Estos actuadores son muy comunes en el mercado y mayormente utilizados por vehículos de gama baja y media, se pueden observar en la Figura 23 que están diseñados para adaptarse a las necesidades de cada vehículo. Lo cual fue conveniente para el sistema.



Figura 23. Actuador universal de seguros eléctricos.

En la Tabla 10 se aprecian las características del mismo, estas fueron utilizadas para el diseño electrónico y sistema de control empleado por el sistema embebido.

Tabla 10. Características del actuador universal de seguros automáticos.

Característica	Valor
Rango de alimentación de voltaje	8 a 15v
Potencia	1.23 a 37 W
Amperaje	3 A
Dimensiones	48.5 x 23mm

3.4.1.1. Banco de relevadores

Para el uso e implemento de los actuadores se diseñó un control ON/OFF para monitorear los actuadores correspondientes. Esto se logró con el uso de un banco de 4 relevadores con las especificaciones que se muestran en la Tabla 11.

Tabla 11. Características principales del banco de relevadores.

Característica	Valor
Rango de alimentación de voltaje	3.3 a 8v
Potencia	1.23 a 37 W
Amperaje	3 A
Dimensiones	48.5 x 23mm

El banco que se muestra en la Figura 24, es el mismo modelo que se implementó para el sistema embebido, con este modelo en específico se pudo enlazar directamente con el microcontrolador debido a que manejan los mismos niveles de voltaje como se puede apreciar en las especificaciones del mismo.

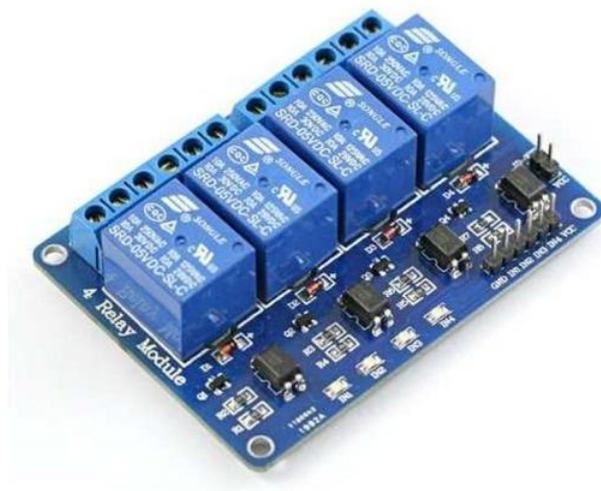


Figura 24. Módulo de relevadores.

En la Figura 25 se puede observar su circuito electrónico interno del módulo de relevadores.

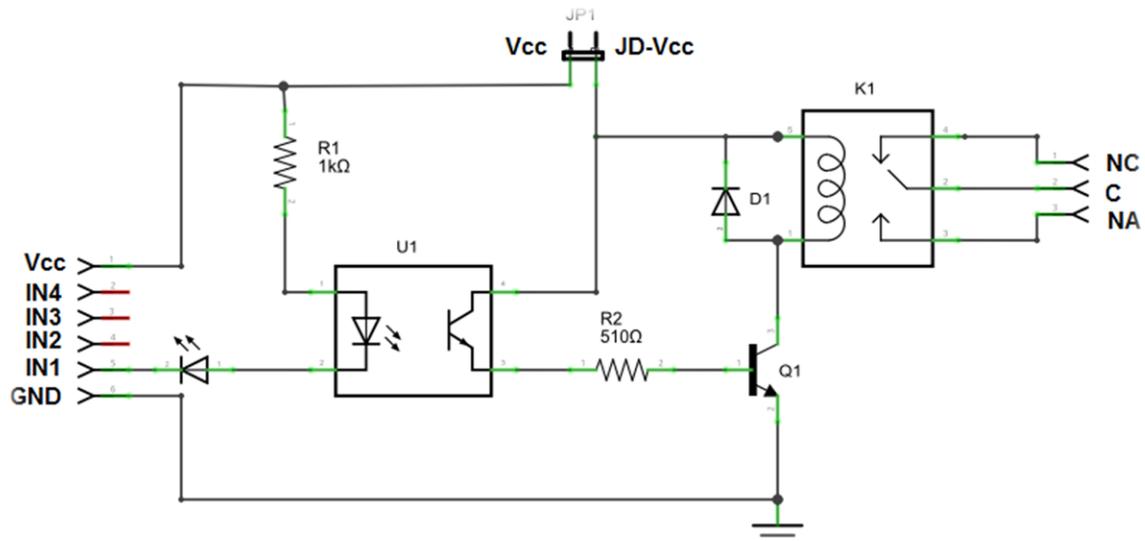


Figura 25. Diagrama electrónico del módulo de relevadores.

3.4.2. Diagrama electrónico

El sistema embebido fue pensado y diseñado para manejar los mismos niveles de voltaje en cada punto del circuito, a su vez poder controlar otros niveles de voltaje para los

actuadores. De esta manera se implementó el sistema embebido con la misma fuente de alimentación que utiliza el vehículo, todo esto se puede observar en la Figura 26.

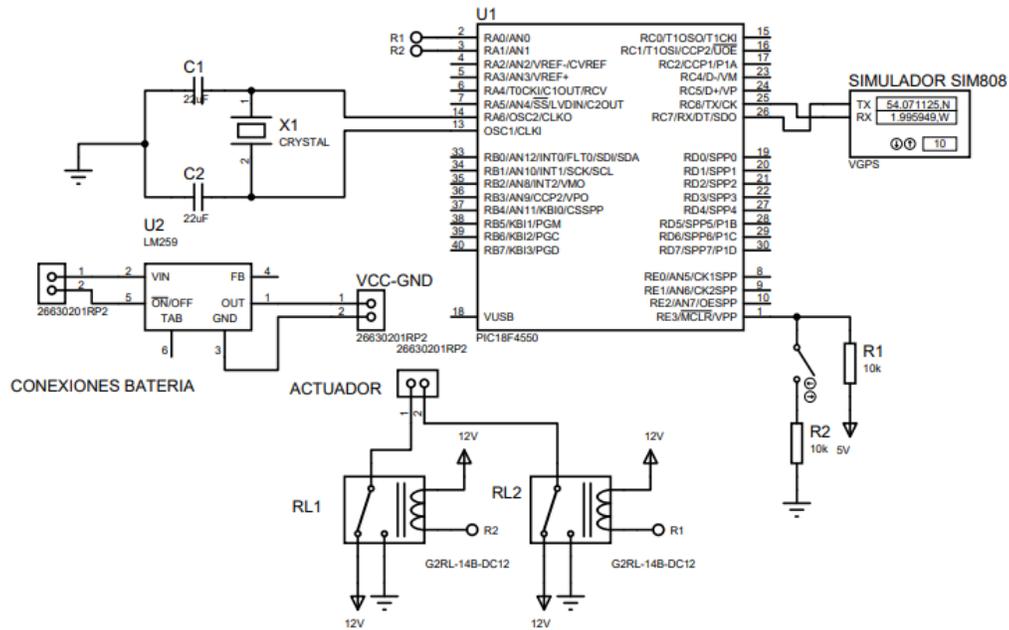


Figura 26. Diagrama electrónico del sistema embebido.

Como se estableció en los objetivos este circuito debe ser de bajo costo para el usuario por lo que el diagrama electrónico cuenta con los dispositivos necesarios para su correcto funcionamiento y con la seguridad electrónica básica para su implementación.

Para la estructura electrónica del sistema embebido fue indispensable contar un regulador de voltaje fijo pensando en los picos de voltaje al encender el vehículo y mantener un bajo consumo de corriente para evitar descargar la fuente de alimentación. A su vez únicamente se realizó el estudio de los dispositivos implementados en el espectro de corriente directa. Como se muestra en la Figura 27 la conexión de batería fue directamente con el regulador de voltaje LM2596, de esta manera se asegura que todo el sistema embebido, a excepción de los actuadores que se manejan al mismo nivel de voltaje.

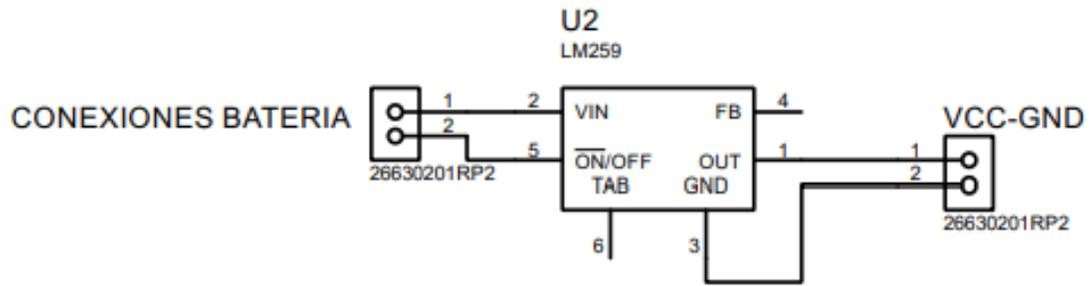


Figura 27. Diagrama de conexiones del módulo LM2597.

Para utilizar los actuadores siguiendo las especificaciones del fabricante en temas de voltaje y corriente, se realizó un circuito electrónico controlado por el microcontrolador donde es manipulado por las salidas R1 y R2 conectadas a relevadores usando GND como señal, asimismo, el común de cada relevador fue conectado a cada terminal del actuador y cada pin Normalmente abierto fue conectado a GND y cada pin Normalmente cerrado fue conectado a VCC como se muestra en la Figura 28.

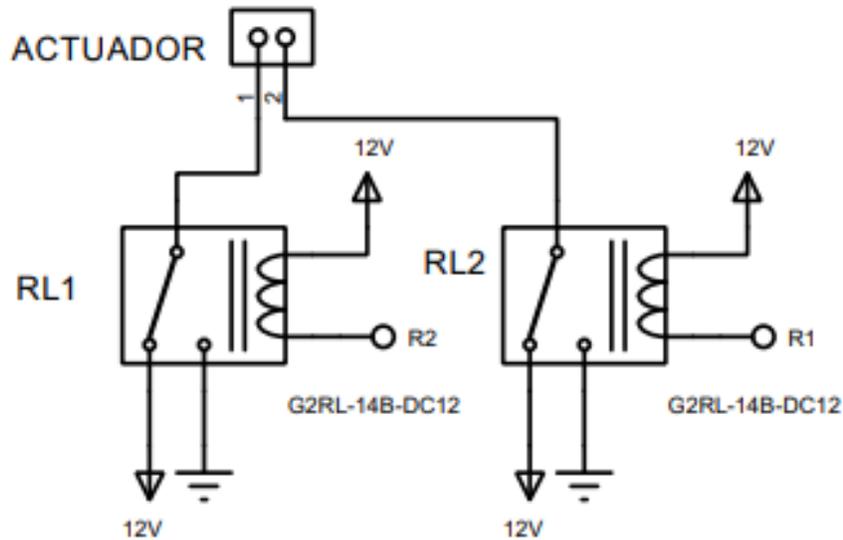


Figura 28. Diagrama de conexiones entre el actuador y módulo de relevadores.

3.5. Comunicación

Para la comunicación del sistema embebido se contempló todos los dispositivos que requieren conexión entre estos para el intercambio de información, se dividió en las siguientes partes:

- Comunicación entre microcontrolador y pantalla LCD.
- Comunicación entre microcontrolador y modulo SIM808.
- Comandos AT para el módulo SIM808.
- Comunicación entre módulo SIM808 y APP en Android.

3.5.1. Comunicación entre microcontrolador y Pantalla LCD

La pantalla LCD se implementó para la etapa de pruebas, durante la implementación no se utilizó por motivos de reducción de consumo de energía. Sin embargo, fue una etapa importante para las pruebas debido a que con la misma se utilizó para comprobar las etapas de procesamiento de información, asimismo contempló como apoyo para futuras fallas que pueda tener el sistema embebido en el vehículo.

Como pantalla LCD se utilizó una pantalla 16x2 de modelo 1602^a, este modelo de pantalla contiene un controlador ST7066U-0A con interfaz disponible de 4bits y 8bits, para las funciones requeridas únicamente se utilizó la interfaz de 4bits. Las conexiones mínimas necesarias para su comunicación se pueden observar en la Figura 29. Para la parte de programación se utilizó una librería de uso libre para pantallas LCD donde los pines de la pantalla LCD se denominan y se conectan de la siguiente manera:

- El pin denominado E es el pin Enable este fue utilizado para habilitar y deshabilitar la pantalla esto significa que este pin al tener un valor de 0 no permite escribir ni leer en la pantalla LCD, por el contrario, cuando tiene un valor igual a 1 este permite escribir o leer en la pantalla LCD.
- El pin R/W es utilizado para seleccionar entre escritura y lectura de la pantalla LCD cuando este pin tiene valor lógico igual a 0 configura la pantalla LCD permite a la pantalla realizar escrituras sobre la misma, de lo contrario cuando tiene un valor lógico igual a 1 configura la pantalla LCD para realizar una lectura de su información.

- El pin RS es un control/registro de datos, cuando este pin tiene un valor lógico igual a 0 selecciona el registro de control y cuando el pin tiene un valor lógico igual a 1 este se selecciona como registro de datos.
- Pines D0-D7 estos pines son utilizados como bus de datos bidireccional.
- Pin V0 este debe tener un valor analógico entre 0v y 5v que permite cambiar el contraste del cristal líquido.
- Pin A este pin debe tener un valor analógico entre 0v y 5v el cual permite tener una retroalimentación positiva de pantalla.
- Pin K este pin debe tener un valor analógico entre 0v y 5v el cual permite tener una retroalimentación negativa de pantalla.

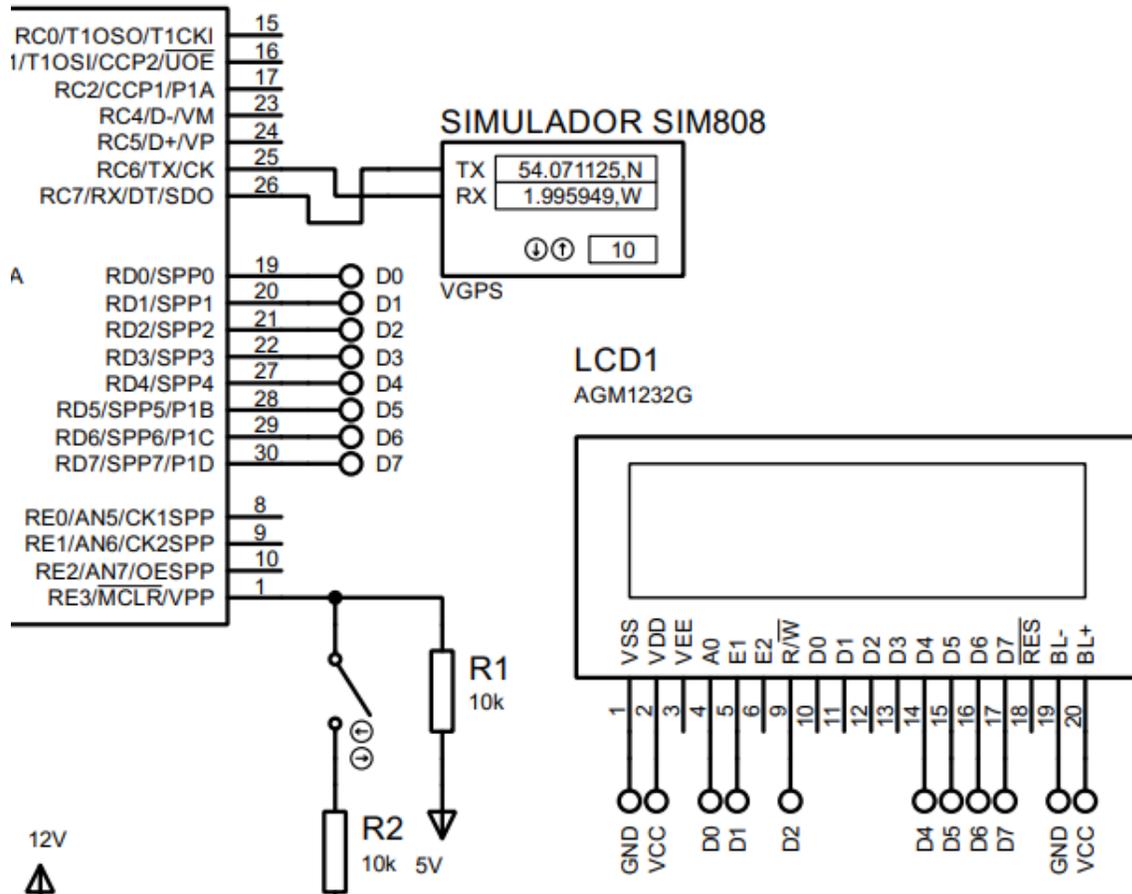


Figura 29. Diagrama de Conexiones entre microcontrolador y pantalla LCD 16x2.

3.5.2. Comunicación entre microcontrolador y Módulo SIM808

La comunicación entre el microcontrolador y el módulo de comunicaciones fue esencial y una de las partes más importantes de sistema embebido. El módulo SIM808 tiene la posibilidad de trabajar con protocolo de comunicación I2C y UART. Se eligió el protocolo de comunicación UART por que es el que viene preestablecido en el módulo, asimismo, debido a que los pines de conexión ya están presoldados para su uso, por el contrario, para utilizar el protocolo de comunicación I2C se tienen que hacer mínimos cambios.

Una vez establecido el protocolo de comunicación a utilizar se procedió a realizar las mínimas conexiones necesarias para el uso del módulo y el microcontrolador, estas conexiones se pueden apreciar en la Figura 30.

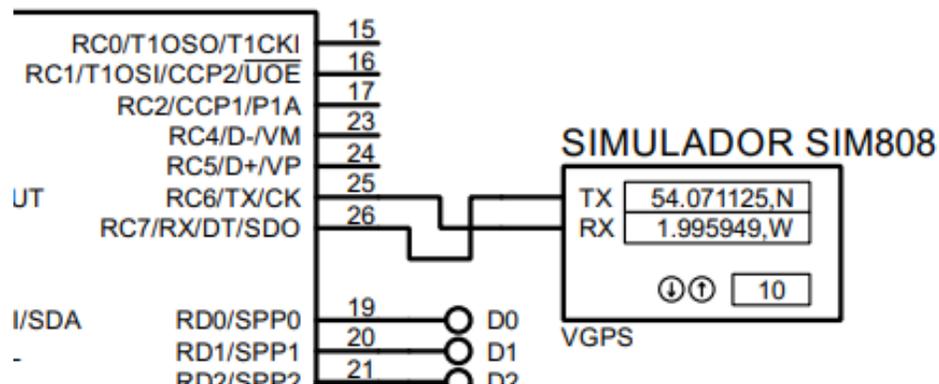


Figura 30. Diagrama de conexiones entre el microcontrolador y módulo SIM808 para comunicación serial.

Como siguiente paso fue establecer los parámetros necesarios para el protocolo de comunicación los cuales fueron los siguientes:

- No se cuenta con Bit de paridad.
- Velocidad en Baudios de 9600.
- 8 Bits de comunicación.
- Conexión Tx que es el transmisor de información.
- Conexión Rx que es el identificado como el receptor de información.

Un punto importante que se tomó en cuenta es que el módulo SIM808 y el microcontrolador trabajan de manera asíncrona por lo cual cada uno tiene su cristal oscilador que maneja su velocidad, pero al momento de la comunicación ambos deben tener la misma velocidad de transmisión y recepción. En las conexiones fue importante colocar de manera cruzada los pines, el Tx de módulo SIM808 se conecta al Rx del microcontrolador y de fue de manera similar para los pines faltantes el Tx del microcontrolador se conecta con el pin Rx del módulo SIM808.

3.5.3. Comandos AT módulo SIM808

Los comandos AT tienen como objetivo ejecutar instrucciones específicas para el módulo SIM808, ya que sin las mismas no habría instrucciones que pueda comprender el módulo. Estos comandos vienen preestablecidos por el fabricante y se debe recurrir a la hoja de datos para comprender cual es la instrucción específica de cada uno. Para el sistema embebido fueron utilizados los comandos AT que se muestran en la Tabla 12.

Tabla 12. Comandos AT para el Módulo de comunicación SIM808.

Comando AT	Función
AT+CMGS="\+524424501556\"	Enviar mensaje al número
AT+CMGF=1	Activar módulo GSM
AT+CGPSPWR=1	Encender módulo GPS
AT+CGPSSTATUS?	Verificar si el módulo está conectado a los satélites
AT+CGPSINF=0	Solicitar ubicación al módulo GPS
AT+CGPSPWR=0	Apagar módulo GPS
AT+CMGR=1	Verificar si hay SMS sin leer

De igual manera fue importante tener la comunicación UART funcionando correctamente ya que el módulo debe de recibir correctamente el comando AT de lo contrario no podrá ejecutar ninguna instrucción.

3.5.4. Comunicación entre módulo SIM808 y APP en Android

Para esta comunicación no se requiere una conexión física entre ambos dispositivos electrónicos; sin embargo, se tuvo que especificar y programar de manera tal para que los dispositivos se puedan comunicar directamente. Para esto se instaló un chip independiente al módulo SIM808 con número conocido y con línea telefónica abierta como se muestra en la Figura 31.



Figura 31. Instalación de tarjeta SIM con módulo SIM808.

A su vez se especificó en la APP aplicación el número telefónico que tiene el chip del módulo de comunicación SIM808, como se resaltó en la Figura 32 se definió como una constante que se utilizó durante toda la programación de la APP.

```
};

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_actividades);

    intentFilter = new IntentFilter();
    intentFilter.addAction("SMS_RECEIVED_ACTION");

    String numero = "4428217509";

    //4423972804 numero padre 4428217509karen
    //-----PERMISO PARA ENVIAR MENSAJES-----
    if(ActivityCompat.checkSelfPermission(
        context: Actividades.this, Manifest.permission.SEND_SMS)!=
        PackageManager.PERMISSION_GRANTED && ActivityCompat.checkSelfPermission(
        context: Actividades.this, Manifest.permission.SEND_SMS) !=
        PackageManager.PERMISSION_GRANTED)
    {
        ActivityCompat.requestPermissions( activity: Actividades.this,new String[]
            {Manifest.permission.SEND_SMS}, requestCode: 1000);
    }else
    {
        //Toast.makeText(this, "Permisos no aprobados", Toast.LENGTH_LONG).show();
    }
}
```

Figura 32. Constante de número a enviar información.

Para finalizar la comunicación entre ambos componentes es necesario especificar al módulo de comunicación SIM808 el número telefónico del receptor de la información. Para se implementó en la programación del microcontrolador un comando AT donde se especifica el número del receptor. En la Figura 33 se muestra la línea exacta de programación donde se realiza lo anterior mencionado.

```
output_high(BAJARVIDRIOS);

//ModuleGetCoordinates();

do
{
    puts("AT+CMGF=1"); //SE CONFIGURA EL MODULO PARA RECEPCION Y ENVIO DE SMS
    delay_ms(1000); // ESPERAMOS RESPUESTA DEL MODULO
}while(flag!=1); // WAITING "OK"

    memset(buff,NULL,15); //RESET BUFF
    flag=0;

    output_high(LED);
    delay_ms(1000);
    output_low(LED);
    do
    {
        puts("AT+CMGS=\\\"+524424501556\\\" ");
        delay_ms(50);
        printf("GPS CONECTADO");
        delay_ms(10);
        putc(0x1A);
    }while(flag!=1); //WAITING "OK"
    flag=0;
    output_high(LED);
    delay_ms(1000);
    output_low(LED);
```

Figura 33. Comando AT configurando el receptor de la información.

3.6. Desarrollo de Software

A continuación, se describen a detalle todos los softwares empleados para la programación del controlador, así como el de la aplicación del dispositivo móvil.

3.6.1. CCS C COMPILER

En este software de programación se utilizó para el algoritmo del microcontrolador. Este software tiene la ventaja de tener un lenguaje de alto nivel, esto a su vez permite simplificar el algoritmo de programación ya que cuenta con librerías, uso y manejo de fusibles del microcontrolador, manejo de puertos de microcontrolador, entre otros.

Como se puede observar en la Figura 34 se estableció el microcontrolador a utilizar, los fusibles a emplear y su modo de uso, se definieron algunos valores predeterminados que se utilizaron dentro del programa.

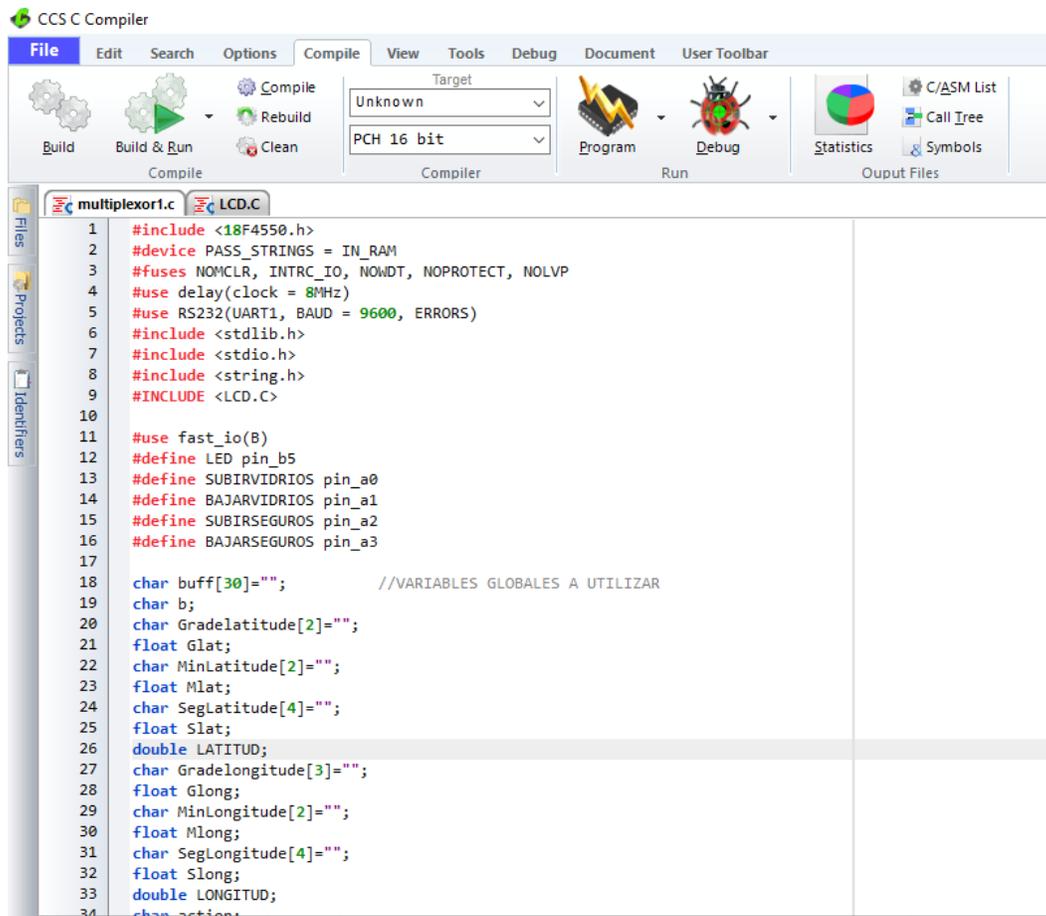


Figura 34. Inicio de la programación del microcontrolador.

3.6.2. PIC Kit 3

Este software de compilación fue necesario para poder cargar todo el algoritmo, previamente programado en CCS, al microcontrolador. De igual manera este software fue útil para pruebas debido a que como se muestra en la Figura 35 existe la opción de activar salida de voltaje para pruebas. Este mismo permitió la configuración de manera rápida el voltaje a utilizar.

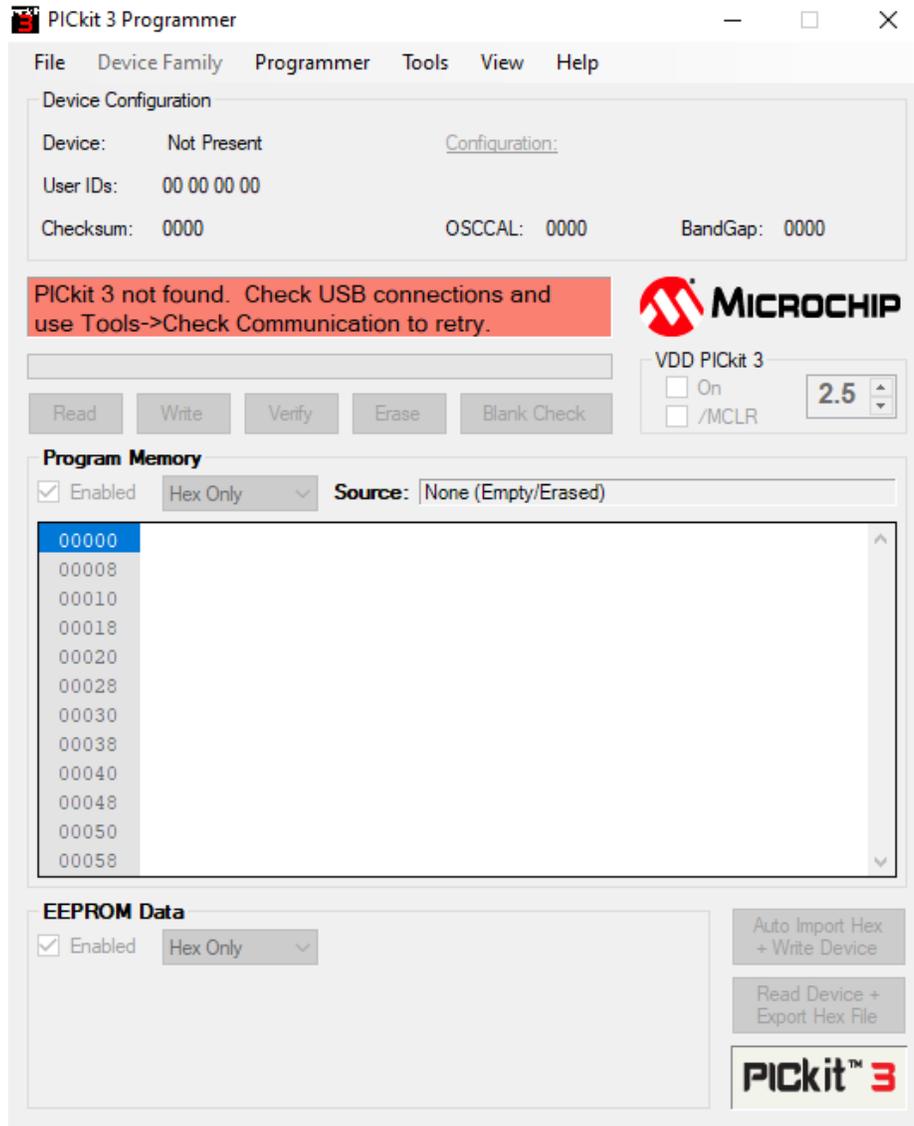


Figura 35. Interfaz PicKit3.

De igual manera es necesario establecer las conexiones necesarias en los pines correspondientes para que se pueda obtener la correcta transferencia de datos, estos se pueden apreciar en la Figura 36.

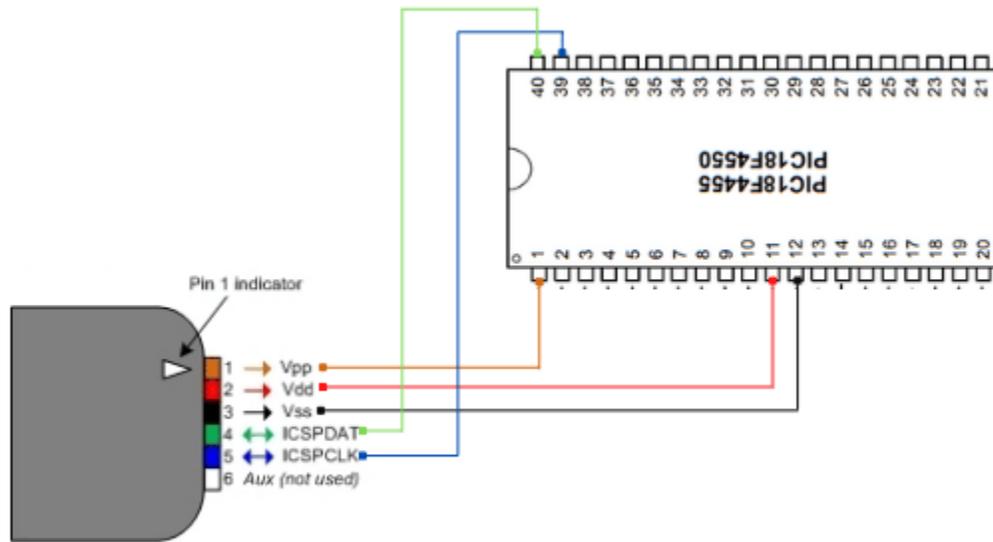


Figura 36. Diagrama de conexiones entre microcontrolador y dispositivo Pickit3.

Una vez teniendo las conexiones necesarias correctamente instaladas, se procede a presionar el botón “Tools” este botón se muestra en la Figura 37.

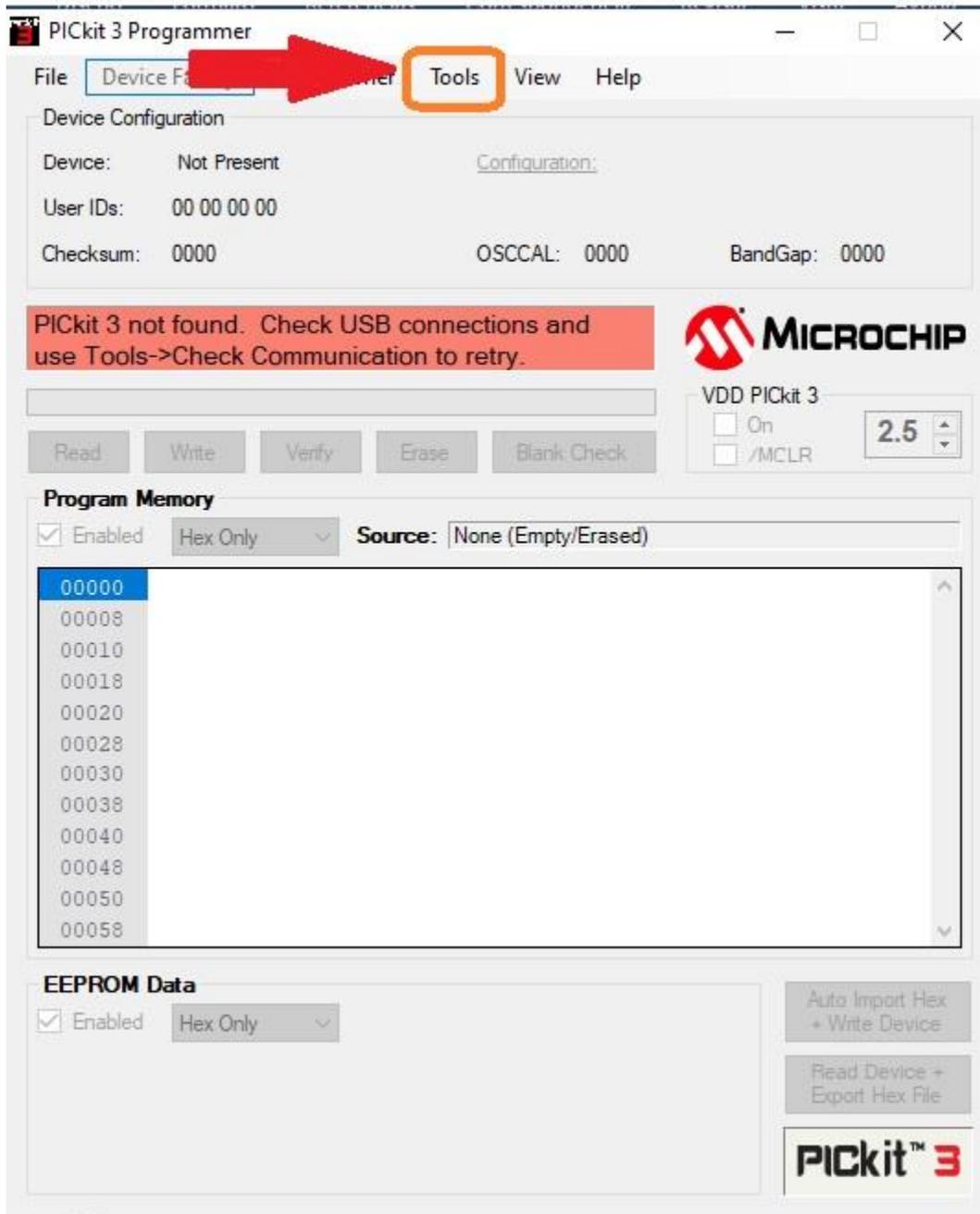


Figura 37. Lectura del microcontrolador mediante interfaz Pickit3.

Posteriormente nos desplegará una ventana con distintas opciones, de las cuales la opción a utilizar es “Check Communication” que se muestra en la Figura 38 esta opción nos va a permitir verificar la comunicación entre en microcontrolador y la computadora.

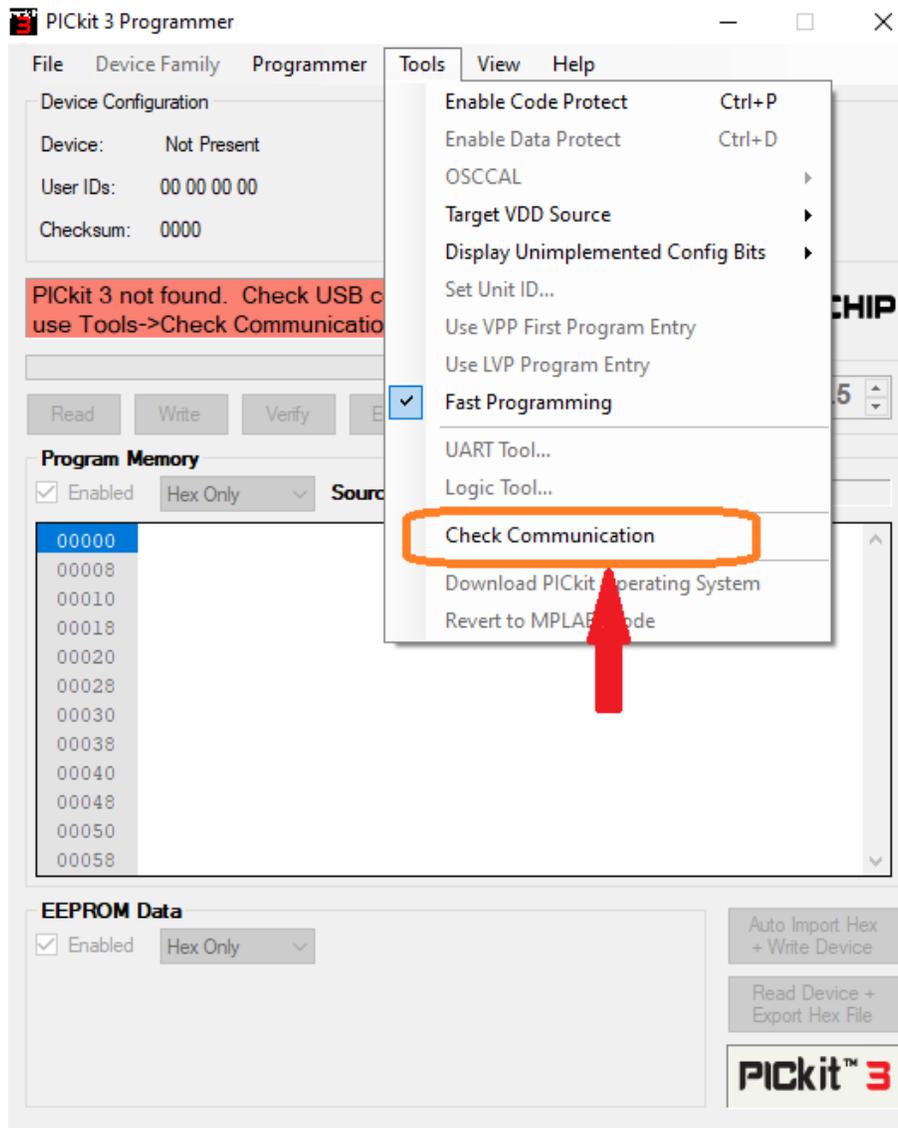


Figura 38. Revisión de comunicación entre el microcontrolador e interfaz PicKit3.

Para este punto se hace una verificación automática del modelo del microcontrolador usado y su correcta comunicación. Si la comunicación entre ambas partes es exitosa se mostrará el microcontrolador conectado y una leyenda donde nos indica una correcta comunicación tal y como se observa en la Figura 39.

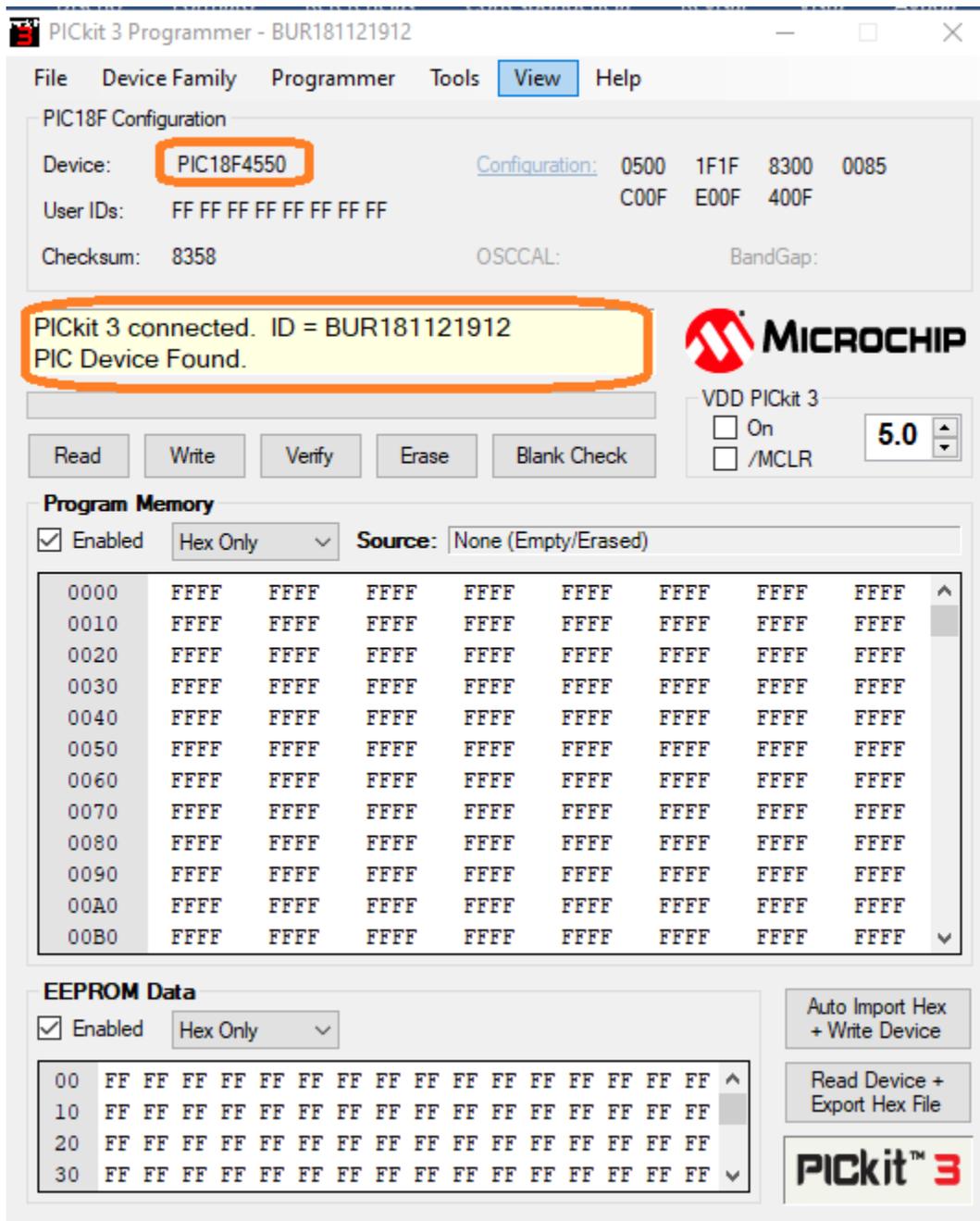


Figura 39. Correcta comunicación entre microcontrolador e interfaz Pickit3.

3.6.3. Android Studio

Este software se utilizó para realizar la interfaz gráfica entre el teléfono móvil y el módulo sim808 que a su vez se comunica con el microcontrolador, este software permitió poder ampliar la gama de celulares compatibles con la App desarrollada tal y como se muestra en la Figura 40.

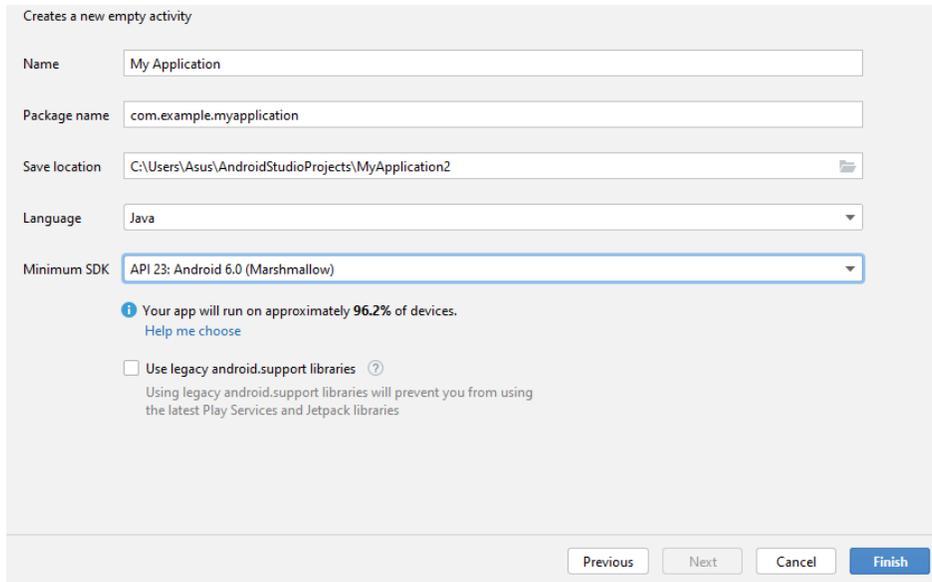


Figura 40. Configuración de versión de dispositivos Android.

Por disposición del software también se pudo elegir entre 2 lenguajes de programación (Kotlin y java). Para este caso en específico se utilizó el lenguaje de programación en java. Esto por practicidad que tiene al hacer uso de diferentes actividades dentro del programa. Como se puede observar en la Figura 41 se utilizaron 4 distintas actividades.

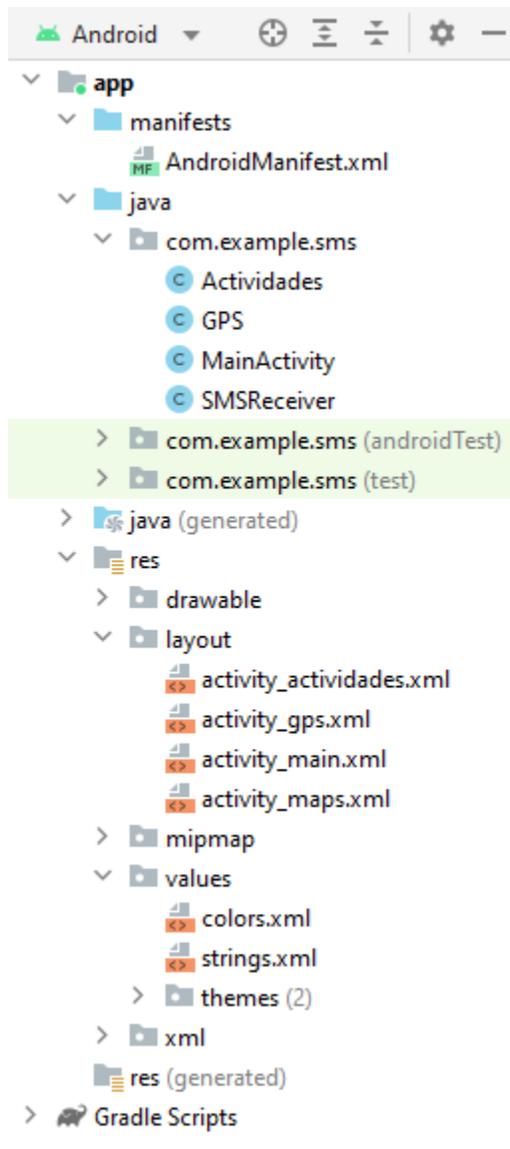


Figura 41. Menú de Actividades dentro de la APP.

Las actividades se enlistan y se describen en la Tabla 13.

Tabla 13 Actividades de la APP desarrollada en Android Studio.

Actividad	Descripción
MainActivity	Filtro de acceso de usuario
Actividades	Mostrar los botones de las acciones a realizar en el vehículo
GPS	Mostrar un mapa GPS con pin colocado en la ubicación actual del vehículo
SMSReceiver	Poder leer los SMS entrantes al dispositivo móvil mediante de la APP

3.6.3.1. MainActivity

Para esta actividad se desarrolló uno de los filtros de seguridad para el usuario, aunado a esto se diseñó una imagen amigable para el usuario con los logos de los creadores. Esto de manera que fuese atractiva y amigable para el usuario indicando visualmente el lugar donde colocar su usuario y su contraseña, del mismo modo se tiene un botón para poder acceder. Dicha imagen se puede observar en la Figura 42.



Figura 42. Pantalla de inicio de la App desarrollada.

3.6.3.2. Actividades

La función de esta actividad es tener distintos botones para el usuario donde se puedan realizar las acciones deseadas por el usuario, a continuación, se enlistan las actividades que se pueden realizar mediante la APP:

- Subir Vidrios.
- Bajar Vidrios.
- Subir ventanas.
- Bajar ventanas.
- Obtener ubicación del vehículo.
- Mostrar en mapa la ubicación del vehículo.

Como se puede observar en la Figura 43 se muestran todas las acciones que se pueden realizar mediante la APP, esto de manera amigable y bastante visual para el usuario.



Figura 43. Actividades a realizar dentro de la App desarrollada.

3.6.3.3. SMSReceiver

Esta actividad fue de gran importancia a pesar de no tener una imagen visual dentro de la APP. La función de esta actividad es poder controlar los mensajes SMS recibidos al teléfono móvil. Esta actividad tuvo como objetivo específico recibir cualquier mensaje de texto proveniente desde cualquier teléfono móvil y poder leer su contenido. Una vez obteniendo el número telefónico del remitente se verifica si este proviene del sistema embebido por lo que si esto es corroborado se procede a leer el mensaje.

3.6.3.4. GPS

El objetivo principal de esta actividad fue mostrar en pantalla un mapa mundial. Como segundo objetivo se estableció el poder colocar un pin en la ubicación recibida por la actividad “SMSReceiber”. Como se muestra en la Figura 44 se observa el PIN colocado con la ubicación del vehículo, esta misma actividad tuvo la capacidad de acercar o alejar el mapa para fines deseados por el usuario.



Figura 44. Ubicación dentro de la App desarrollada.

3.7. Diseño e instalación

Para el diseño electrónico se utilizó el software Proteus, este software fue de gran ayuda para poder definir los puertos a utilizar de salidas, entradas, pines de comunicación y pines de alimentación. Como se puede observar en la Figura 45 se pueden observar todas las conexiones mínimas necesarias para el funcionamiento correcto del sistema embebido. En este software se pudo realizar un enlace electrónico correcto entre el módulo de comunicación, microcontrolador, alimentación y módulos de voltaje para su posterior instalación en placa perforada.

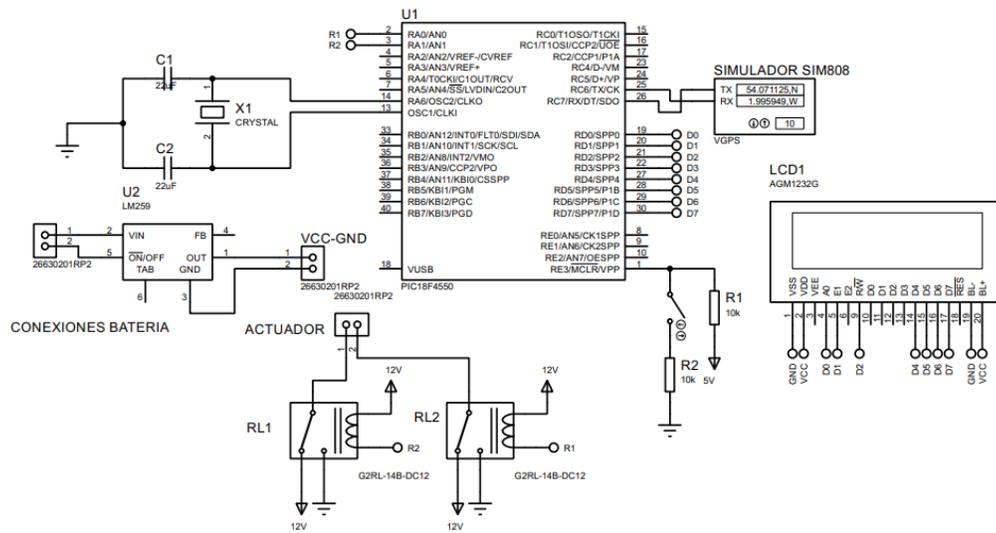


Figura 45. Diseño electrónico del sistema embebido.

3.8. Controlador

El controlador es la parte más importante del sistema debido a que se debe considerar los actuadores y los módulos de comunicación. Como primer paso se ha establecido las librerías que serán utilizadas mediante el proceso. En estas librerías se incluyen las necesarias para dispositivos externos como la pantalla LCD 16x2, comunicación serial entre otros, dichas librerías se muestran en la Figura 46 resaltadas en color azul. Asimismo, se les asignaron sus pines correspondientes para su montaje final, estos pines se pueden apreciar en la Figura 46 resaltadas en color naranja.

The screenshot shows the CCS C Compiler IDE with the file `multiplexor1.c` open. The code includes several preprocessor directives and variable declarations. Two sections are highlighted with colored boxes and arrows:

- A blue box highlights the include statements: `#include <stdlib.h>`, `#include <stdio.h>`, `#include <string.h>`, and `#INCLUDE <LCD.C>`. A blue arrow points to this box.
- A red box highlights the define statements: `#define LED pin_b5`, `#define SUBIRVIDRIOS pin_a0`, `#define BAJARVIDRIOS pin_a1`, `#define SUBIRSEGUROS pin_a2`, and `#define BAJARSEGUROS pin_a3`. A red arrow points to this box.

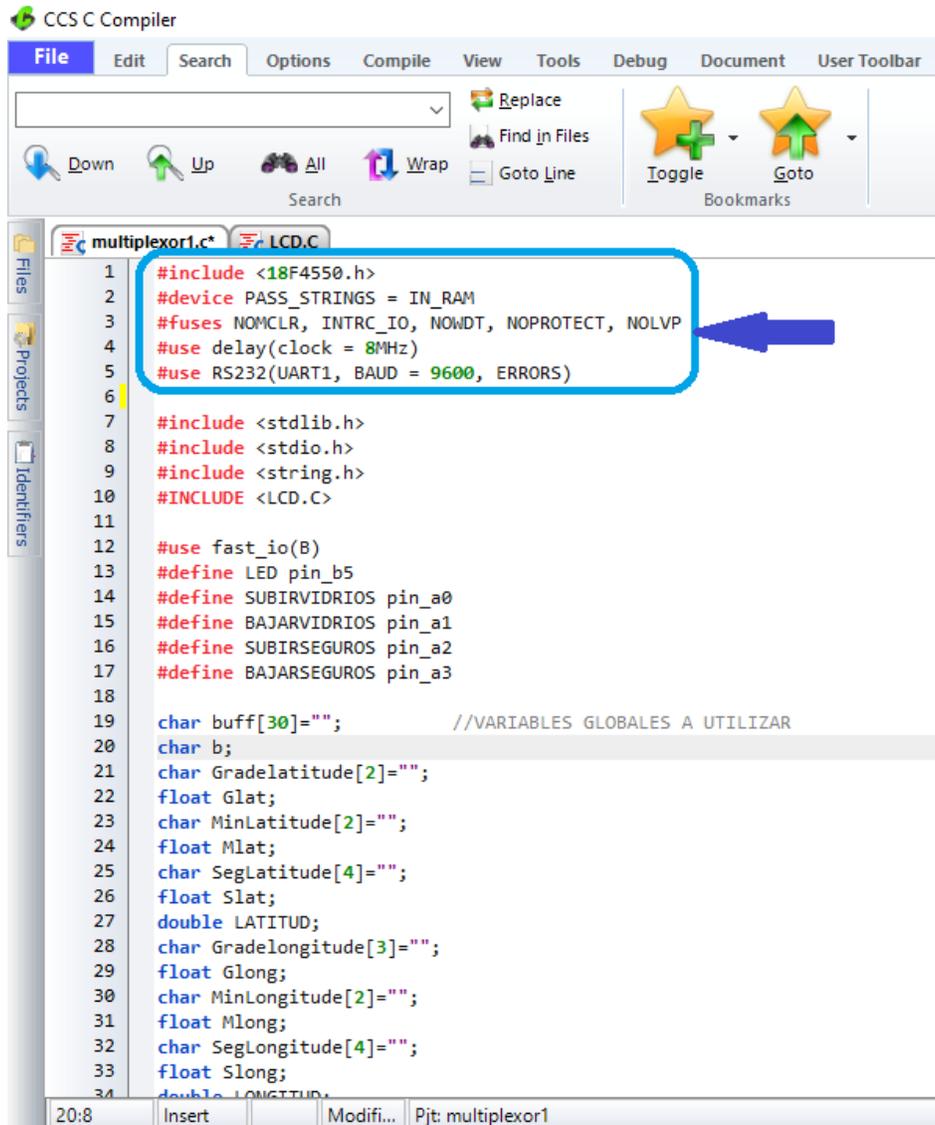
```
1 #include <18F4550.h>
2 #device PASS_STRINGS = IN_RAM
3 #fuses NOMCLR, INTRC_IO, NOWDT, NOPROTECT, NOLVP
4 #use delay(clock = 8MHz)
5 #use RS232(UART1, BAUD = 9600, ERRORS)
6
7 #include <stdlib.h>
8 #include <stdio.h>
9 #include <string.h>
10 #INCLUDE <LCD.C>
11
12 #use fast_io(B)
13 #define LED pin_b5
14 #define SUBIRVIDRIOS pin_a0
15 #define BAJARVIDRIOS pin_a1
16 #define SUBIRSEGUROS pin_a2
17 #define BAJARSEGUROS pin_a3
18
19 char buff[30]=""; //VARIABLES GLOBALES A UTILIZAR
20 char b;
21 char Gradelatitude[2]="";
22 float Glat;
23 char MinLatitude[2]="";
24 float Mlat;
25 char SegLatitude[4]="";
26 float Slat;
27 double LATITUD;
28 char Gradelongitude[3]="";
29 float Glong;
30 char MinLongitude[2]="";
31 float Mlong;
32 char SegLongitude[4]="";
33 float Slong;
34 double LONGITUD;
```

Figura 46. Librerías y definiciones.

Posteriormente se realizó la configuración de fusibles, esto para poder controlar detalladamente la velocidad de comunicación, velocidad del reloj y protecciones para el código, en el mismo apartado se detallan otros puntos, los cuales son:

- Microcontrolador usado.
- Fusibles varios.
- Configuración de la comunicación serial.
- Distintos fusibles.

Para detallar visualmente estos puntos se requiere observar la Figura 47 donde están resaltadas en color azul.



```
1  #include <18F4550.h>
2  #device PASS_STRINGS = IN_RAM
3  #fuses NOMCLR, INTRC_IO, NOWDT, NOPROTECT, NOLVP
4  #use delay(clock = 8MHz)
5  #use RS232(UART1, BAUD = 9600, ERRORS)
6
7  #include <stdlib.h>
8  #include <stdio.h>
9  #include <string.h>
10 #INCLUDE <LCD.C>
11
12 #use fast_io(B)
13 #define LED pin_b5
14 #define SUBIRVIDRIOS pin_a0
15 #define BAJARVIDRIOS pin_a1
16 #define SUBIRSEGUROS pin_a2
17 #define BAJARSEGUROS pin_a3
18
19 char buff[30]=""; //VARIABLES GLOBALES A UTILIZAR
20 char b;
21 char Gradelatitude[2]="";
22 float Glat;
23 char MinLatitude[2]="";
24 float Mlat;
25 char SegLatitude[4]="";
26 float Slat;
27 double LATITUD;
28 char Gradelongitude[3]="";
29 float Glong;
30 char MinLongitude[2]="";
31 float Mlong;
32 char SegLongitude[4]="";
33 float Slong;
34 double LONGITUD;
```

Figura 47. Fusibles y configuración del microcontrolador.

Como siguiente paso se definieron constantes y variables que se utilizaran durante todo el código, en la Figura 48 se muestran dichas variables, asimismo se aprecian su tipo de variable y valores inicializados.

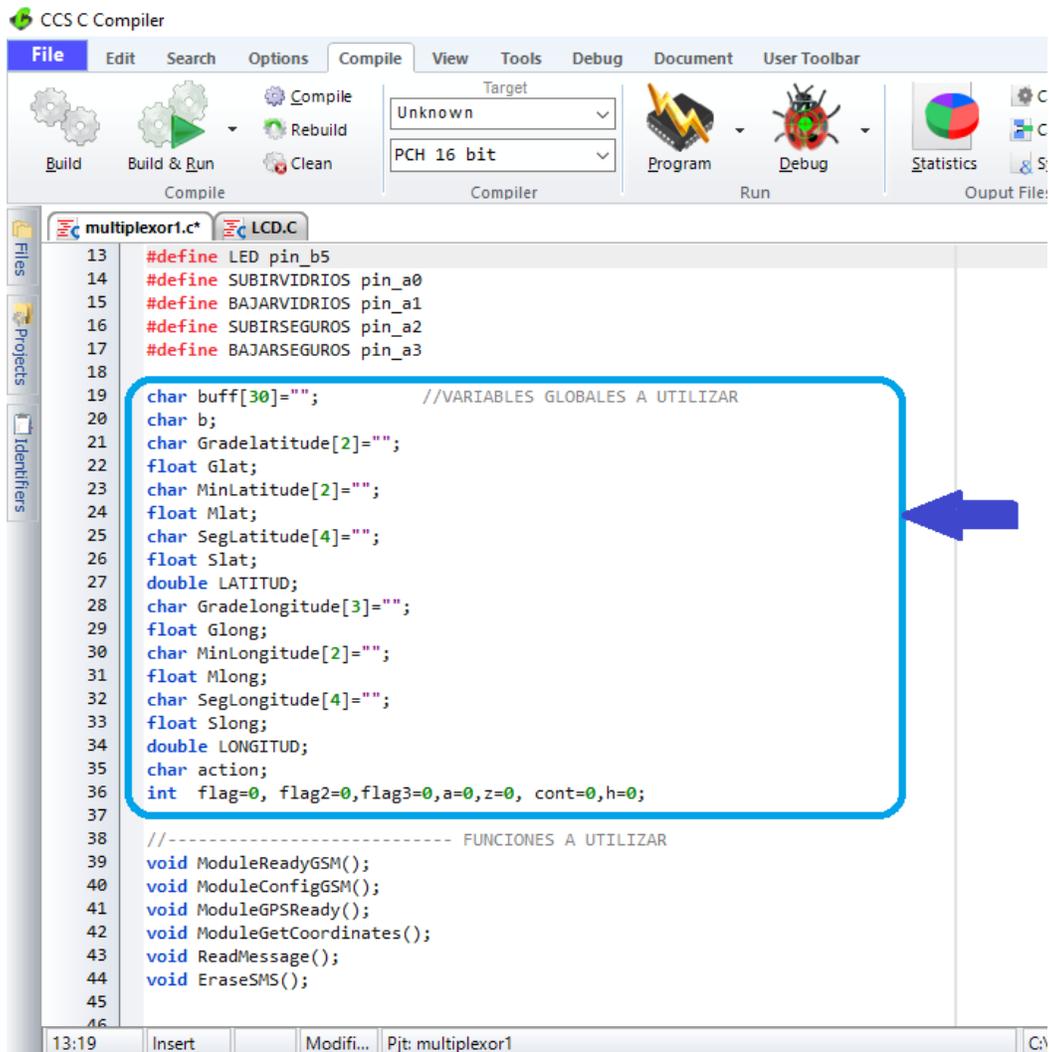


Figura 48. Variables a utilizar.

A continuación, se establecen y programan las funciones a utilizar, considerando todas las librerías, fusibles, constantes y variables previamente programadas. En la Figura 49 únicamente se resalta la inicialización de las funciones, posteriormente en el código se programa el funcionamiento de cada una de estas.

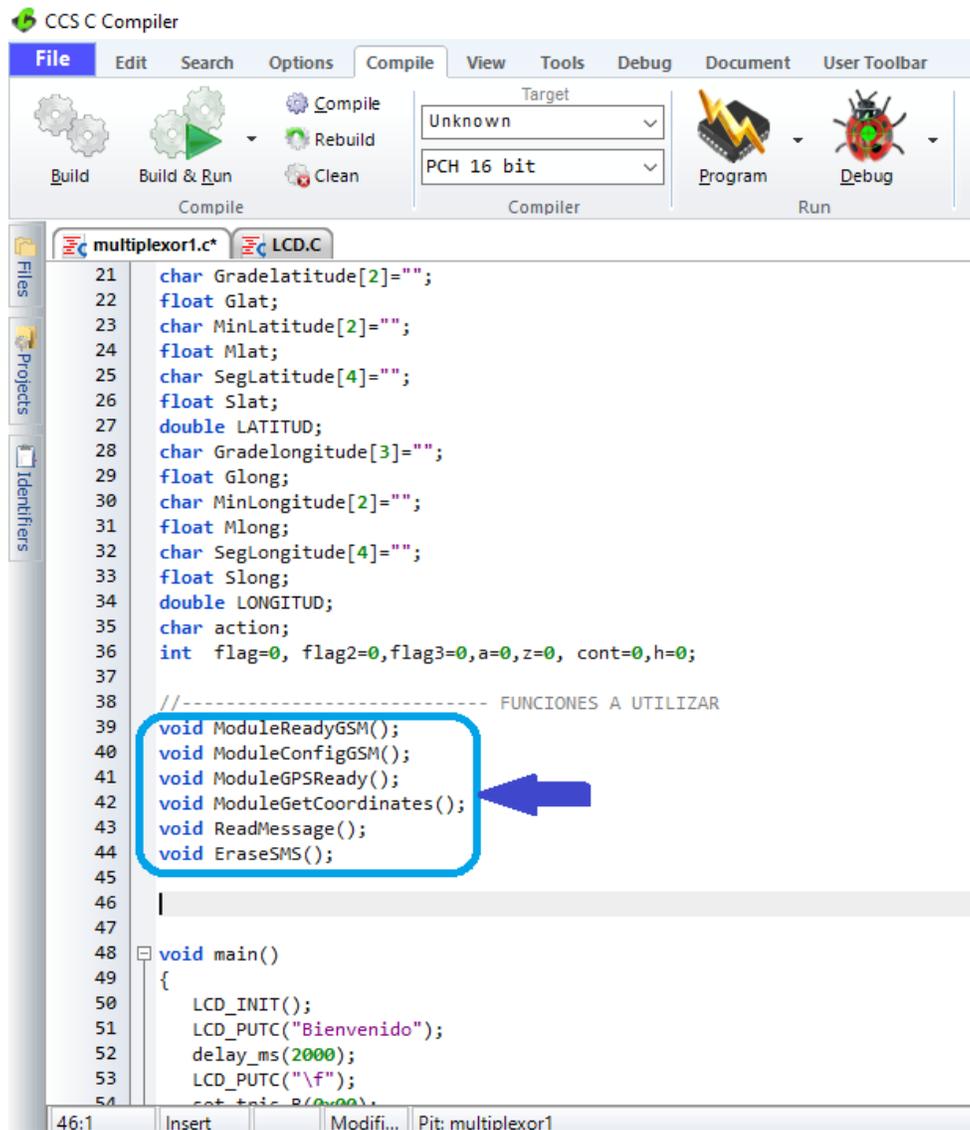


Figura 49. Funciones a utilizar.

Para la función void Main() se inicializan las variables, se terminan de configurar los pines correspondientes a cada salida y entrada del sistema embebido como se aprecia en la Figura 50. Aunado a esto se establece un ciclo de verificación del módulo de comunicación.

```

48 void main()
49 {
50     LCD_INIT();
51     LCD_PUTC("Bienvenido");
52     delay_ms(2000);
53     LCD_PUTC("\f");
54     set_tris_B(0x00);
55     set_tris_A(0x00);
56     setup_oscillator(OSC_8MHZ);
57     enable_interrupts(int_rda);
58     enable_interrupts(GLOBAL);
59     output_B(LED);
60     output_A(SUBIRVIDRIOS);
61     output_A(BAJARVIDRIOS);
62     output_A(SUBIRSEGUROS);
63     output_A(BAJARSEGUROS);
64     ModuleReadyGSM();
65     ModuleGPSReady();
66     output_high(SUBIRSEGUROS);
67     output_high(BAJARSEGUROS);
68     output_high(SUBIRVIDRIOS);
69     output_high(BAJARVIDRIOS);
70
71     //ModuleGetCoordinates();
72
73
74 do
75 {
76     puts("AT+CMGF=1"); //SE CONFIGURA EL MODULO PARA RECEPCION
77     delay_ms(1000); // ESPERAMOS RESPUESTA DEL MODULO
78 }while(flag!=1); // WAITING "OK"
79
80     memset(buff,NULL,15); //RESET BUFF

```

Figura 50. Inicialización de variables y configuración de entradas y salidas.

Para finalizar el ciclo while(true) se mantiene en un ciclo infinito esperando la llegada de algún SMS y poder dirigir a la función correspondiente para su ejecución del sistema embebido como se ilustra en la Figura 51.

```
84     delay_ms(1000);
85     output_low(LED);
86     do
87     {
88         puts("AT+CMGS=\"+524424501556\"");
89         delay_ms(50);
90         printf("GPS CONECTADO");
91         delay_ms(10);
92         putc(0x1A);
93     }while(flag!=1);    //WAITING "OK"
94     flag=0;
95     output_high(LED);
96     delay_ms(1000);
97     output_low(LED);
98
99
100    while(TRUE)
101    {
102
103        ReadMessage(); delay_ms(2000);
104    }//WHILE
105 }//MAIN
106
107 //----- RECEIVER INTERRUPT
108 #int_rda
109 void rda_isr()
110 {
111     b=getc();
112     buff[a]=b;
113     if(buff[a]=='K' && buff[a-1]=='0')flag=1;
114     if(buff[a]=='D' && buff[a-1]=='3')flag=2;
115     if(buff[a]=='R' && buff[a-1]=='G'){flag3=1;}
116     if(buff[a]=='F' && buff[a-1]=='N'){flag2=3;z=0;flag=3;}
117     if(buff[a]=='*' && flag3=1) action=1; //SOLICITA GPS
```

Figura 51. Ciclo infinito.

3.9. Montaje final del sistema

Para concluir con la metodología de diseño e instalación del sistema embebido se procedió a instalar todo el sistema en el diseño previamente establecido para el sistema embebido. Este montaje fue bastante sencillo por el diseño del mismo, solo fue necesario colocar las salidas correspondientes del banco de relevadores a los actuadores necesarios, esto es posible visualizar en la Figura 52.

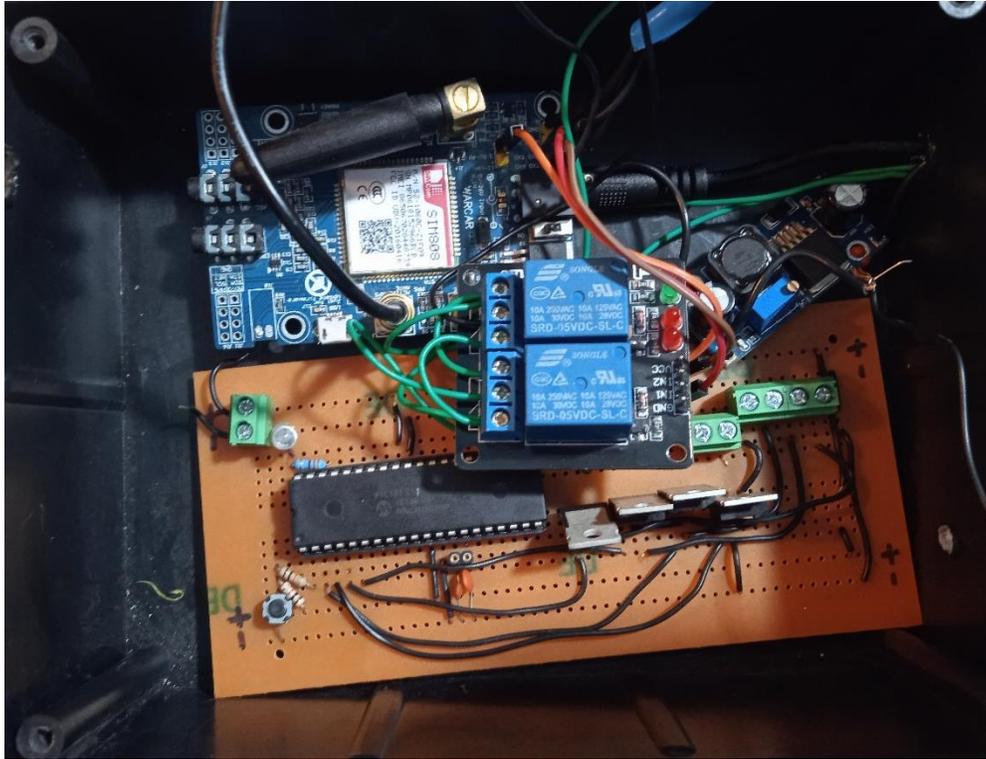


Figura 52. Conexión física de relevadores.

Posteriormente se necesitó colocar la alimentación directamente a la batería del vehículo tal y como se muestra en la Figura 53.



Figura 53. Conexión del sistema embebido a la alimentación.

CAPÍTULO IV

4. RESULTADOS

4.1. Diseño electrónico y pruebas de comunicación

Una vez diseñado el sistema electrónico en protoboard y verificando su correcto funcionamiento se procedió al montaje en una Placa perforada, asimismo, realizaron pruebas de comunicación con el módulo SIM808 y con el regulador de voltaje. En este punto el sistema embebido trabajó correctamente lo cual se prosiguió a instalar en el vehículo seleccionado con la misma fuente de alimentación como se puede observar en la Figura 54. Aunado a esto se realizaron las conexiones necesarias con los actuadores correspondientes del sistema. De este modo el vehículo queda equipado con una alarma funcional capaz de ver su ubicación en tiempo real.

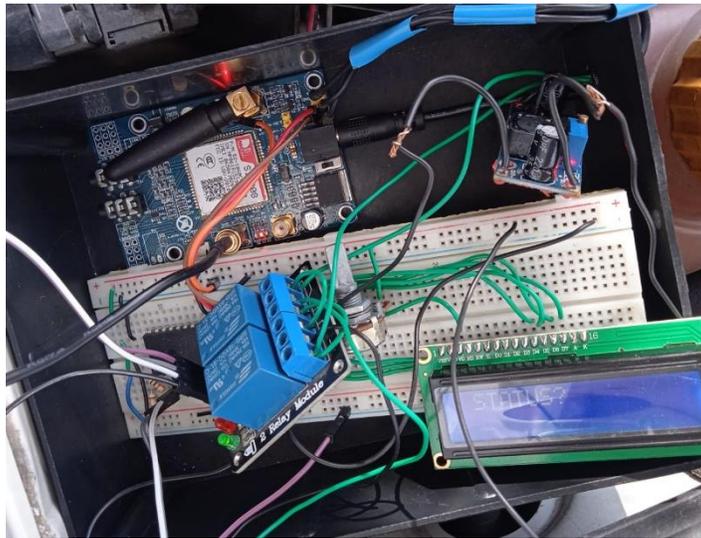


Figura 54. Conexiones de prueba del sistema embebido.

En la Figura 55 se pueden observar todos los componentes por separado para su acceso rápido en cuestión de algún reemplazo, asimismo, en la Tabla 14 se enlistan los componentes utilizados y su ubicación:

Tabla 14. Componentes del Sistema Embebido.

Componente	Ubicación
Microcontrolador pic18f4550	1
Regulador de voltaje LM	2
Banco de relevadores	3
Pantalla LCD 16x2	4
Módulo SIM808	5

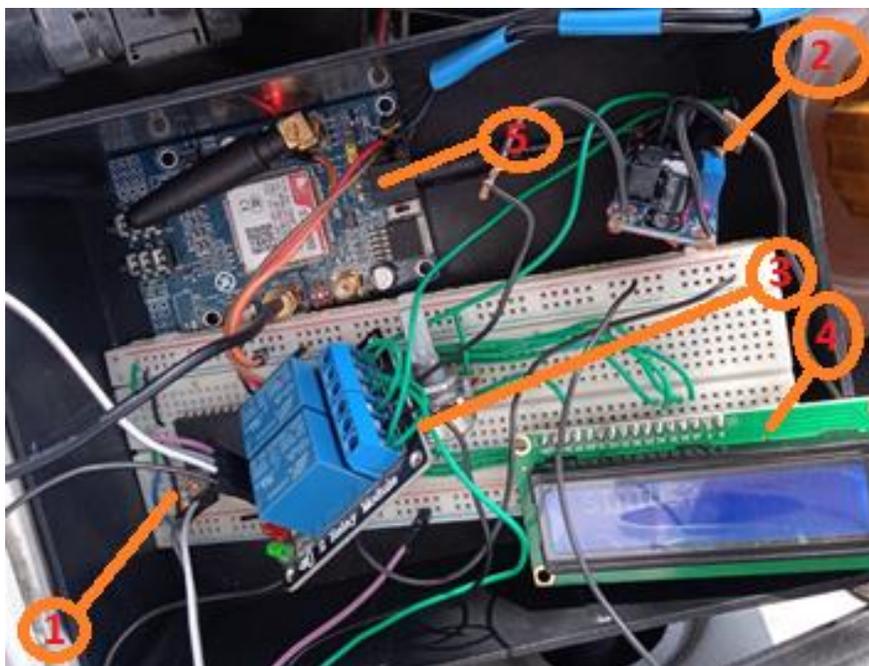


Figura 55. Componentes del sistema embebido.

El protoboard (1), los módulos de relevadores (3), el módulo SIM808 (5), el regulador de voltaje LM2596 (2) y la pantalla LCD 16x2 (4) se encuentran ubicadas dentro del gabinete del sistema embebido. Lo único externo al gabinete del sistema embebido es la antena GPS, las salidas de los actuadores y las entradas de alimentación al sistema. De esta manera se pretende evitar manipulación del sistema embebido por parte del usuario y a su vez si es necesario realizar alguna modificación del sistema se puede tener como guía la pantalla LCD 16x2.

El gabinete es completamente cerrado por lo que se realizó una perforación de aproximadamente 1cm de diámetro esto con el propósito de mantener siempre cerrado el gabinete y poder sacar los cables necesarios para alimentación y salidas de actuadores. Para dichas conexiones se utilizó cable de calibre 22 AWG. Esto en base a la potencia del sistema

embebido para la cual se midió el amperaje que consume el sistema. Como se ilustra en la Figura 56 el amperaje consumido por el sistema en general es de 24.7mA por lo cual el cable de calibre 22 AWG satisfizo las necesidades del sistema sin calentarse.

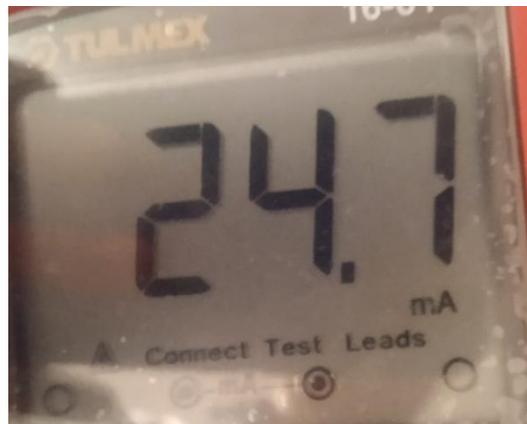


Figura 56. Amperaje consumido del sistema embebido.

En la Figura 57 se pueden observar las conexiones que se realizaron con el actuador de los seguros eléctricos.

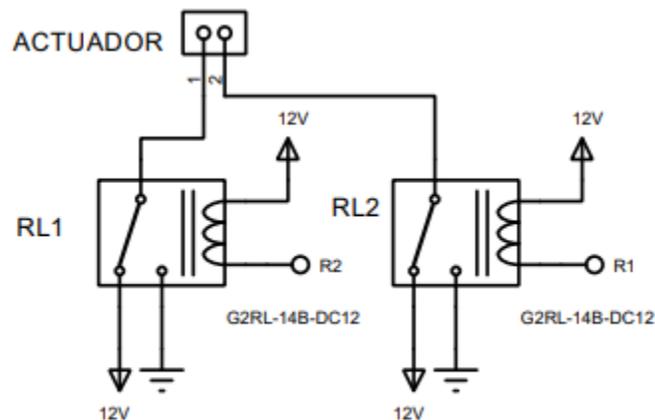


Figura 57. Conexión entre el actuador y el banco de relevadores.

4.2. Visualización de información en el sistema embebido

Una vez encendido el sistema embebido la pantalla LCD 16x2 mostrará las funciones por las cuales pasa en cada proceso. Cuando el sistema es encendido después de

no tener alimentación lo primero que mostrará en la pantalla es la leyenda “Bienvenido” como se puede apreciar en la Figura 58.



Figura 58. Encendido del sistema embebido.

Una vez transcurrido el tiempo de bienvenida el sistema pasará a una función donde se verificará la correcta comunicación con el módulo SIM808, una vez haya tenido respuesta del módulo la pantalla mostrará la leyenda “Modulo conectado” lo cual nos indica que ya tienen una comunicación continua esto se ilustra en la Figura 59.



Figura 59. Conexión exitosa entre el sistema embebido y módulo SIM808.

Posteriormente el sistema embebido realizará una serie de comunicación para verificar el correcto funcionamiento del GPS siguiendo el diagrama de flujo que se muestra en la Figura 60.

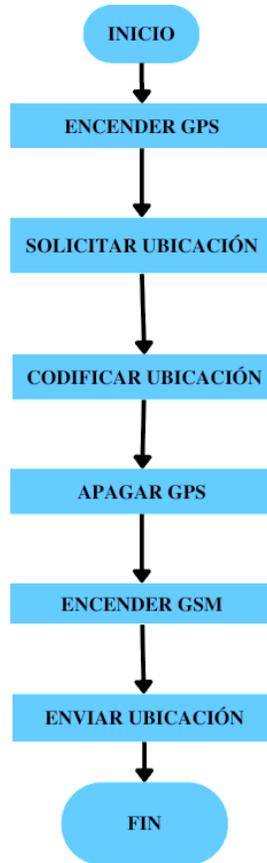


Figura 60. Diagrama de flujo para el envío de la ubicación.

Siguiendo el diagrama de flujo como primer paso se debe encender el módulo GPS esto se logra mediante un comando AT. Al enviar este comando en la pantalla LCD se mostrarán las palabras “GPS POWER ON” como se observa en la Figura 61.



Figura 61. Correcto encendido del GPS.

El siguiente paso es de gran importancia debido a la conexión con los satélites ya que el sistema embebido se quedará en este paso hasta que verifique que el módulo SIM808 este sincronizado con los satélites. Mientras no esté sincronizado correctamente en la pantalla LCD 16x2 se mostrará el mensaje “Statuts?” así como se puede observar en la Figura 62.



Figura 62. Sincronización con los satélites.

Una vez sincronizado con los satélites en la pantalla LCD 16x2 se mostrará el mensaje “3D Fix” por unos segundos como se ilustra en la Figura 63 para posteriormente seguir con su proceso.



Figura 63. Correcta sincronización con los satélites.

Para finalizar con el proceso de sincronización con los satélites es necesario apagar el GPS, de igual manera que al encender el GPS se realiza mediante comando AT. En este último paso la pantalla LCD muestra la leyenda “GPS POWER OFF” como se aprecia en la Figura 64.



Figura 64. Correcto apagado del módulo GPS.

Una vez finalizado el anterior proceso la siguiente función es avisar, al usuario con el cual se encuentra sincronizado, que está en funcionamiento. Este mensaje se puede observar en la Figura 65.



Figura 65. Mensaje de verificación con el usuario.

De igual manera durante este mensaje de verificación sigue el diagrama de flujo que se muestra en la Figura 60.

Como primer paso se activa el modo SMS del módulo con comandos AT. Para realizar este paso mayormente visual se muestra en la pantalla LCD las palabras “MODO SMS” como se muestra en la Figura 66.



Figura 66. Encendido módulo GSM.

El sistema mostrará en la pantalla LCD 16x2 la leyenda de “ENVIANDO MENSAJE” como se muestra en la Figura 67. De esta manera se puede verificar visualmente desde ambos puntos el correcto funcionamiento del sistema una vez recibido el mensaje.



Figura 67. Envío de información mediante módulo GSM.

Como último paso el sistema embebido se queda en espera de recibir algún mensaje para su lectura y procesamiento para realizar alguna acción. En este punto la pantalla LCD muestra “WAIT SMS...” como se muestra en la Figura 68. Mientras llega un mensaje el sistema embebido queda en un bucle que se rompe hasta encontrar una respuesta por parte del usuario.



Figura 68. Estado de espera de recepción de SMS.

4.3. Visualización de información en la APP desarrollada

Para la comunicación a distancia entre el sistema embebido y el usuario se desarrolló una aplicación compatible con los dispositivos móviles que cuenten con sistema operativo Android.

Al acceder a esta aplicación lo primero que nos muestra es un usuario y una contraseña esto está diseñado como seguridad de control de acceso, esta actividad se muestra en la Figura 69.



Figura 69. Filtro de seguridad de la APP desarrollada.

Para poder acceder a la siguiente actividad de la aplicación desarrollada se colocó como usuario “Gabo” y como contraseña “1234” para fines prácticos.

Una vez accediendo el control de seguridad interno de la aplicación la siguiente actividad nos muestra todas las acciones que se pueden realizar al vehículo mediante botones en la aplicación móvil, así como se puede apreciar en la Figura 70.



Figura 70. Acciones a realizar por parte del usuario.

Cada una de estas acciones manda un mensaje vía SMS específico al sistema embebido que a su vez al recibirlo y leerlo realiza una acción en específico. La Tabla 15

muestra las acciones que se pueden realizar junto con los mensajes vía SMS enviados al sistema embebido.

Tabla 15. Mensajes codificados de la APP.

Acción	Mensaje
Obtener ubicación	*
Bajar vidrios	@
Subir vidrios	%
Bajar seguros	&
Subir seguros	i

El único botón que no envía mensaje vía SMS es el botón nombrado como “Mostrar Mapa”, este botón nos dirige a otra actividad, donde nos muestra un Mapa de uso libre con un PIN que es colocado en la ubicación del vehículo como se muestra en la Figura 71.

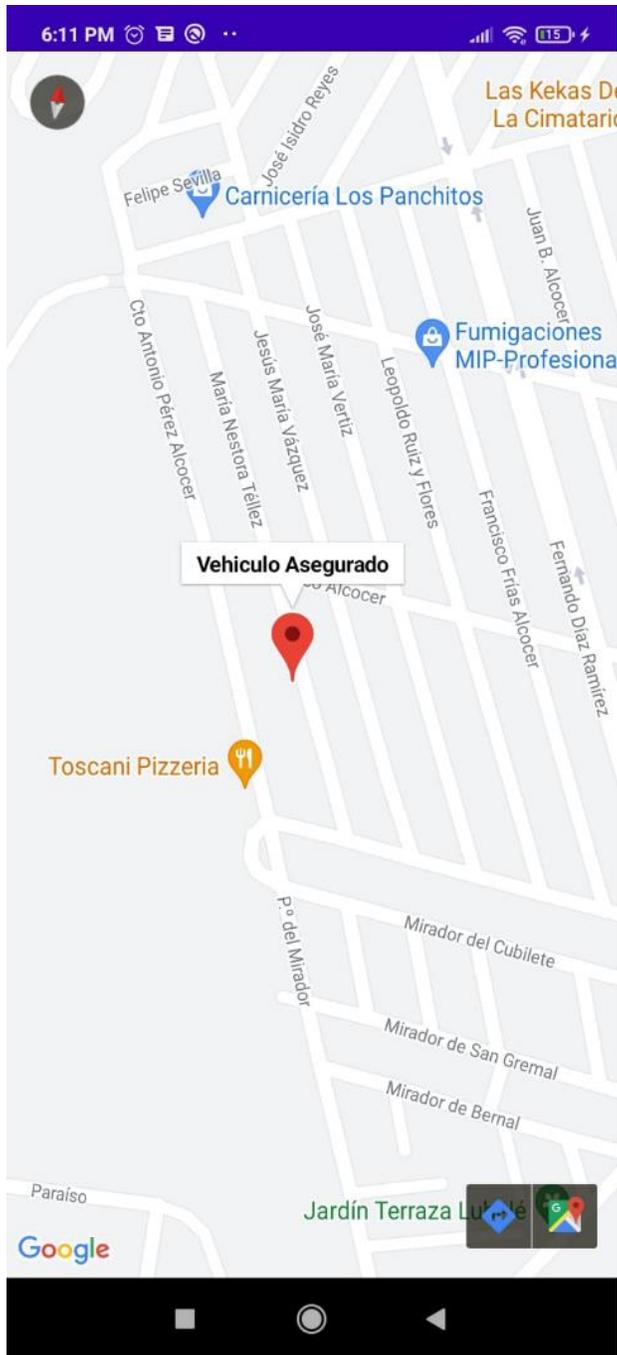
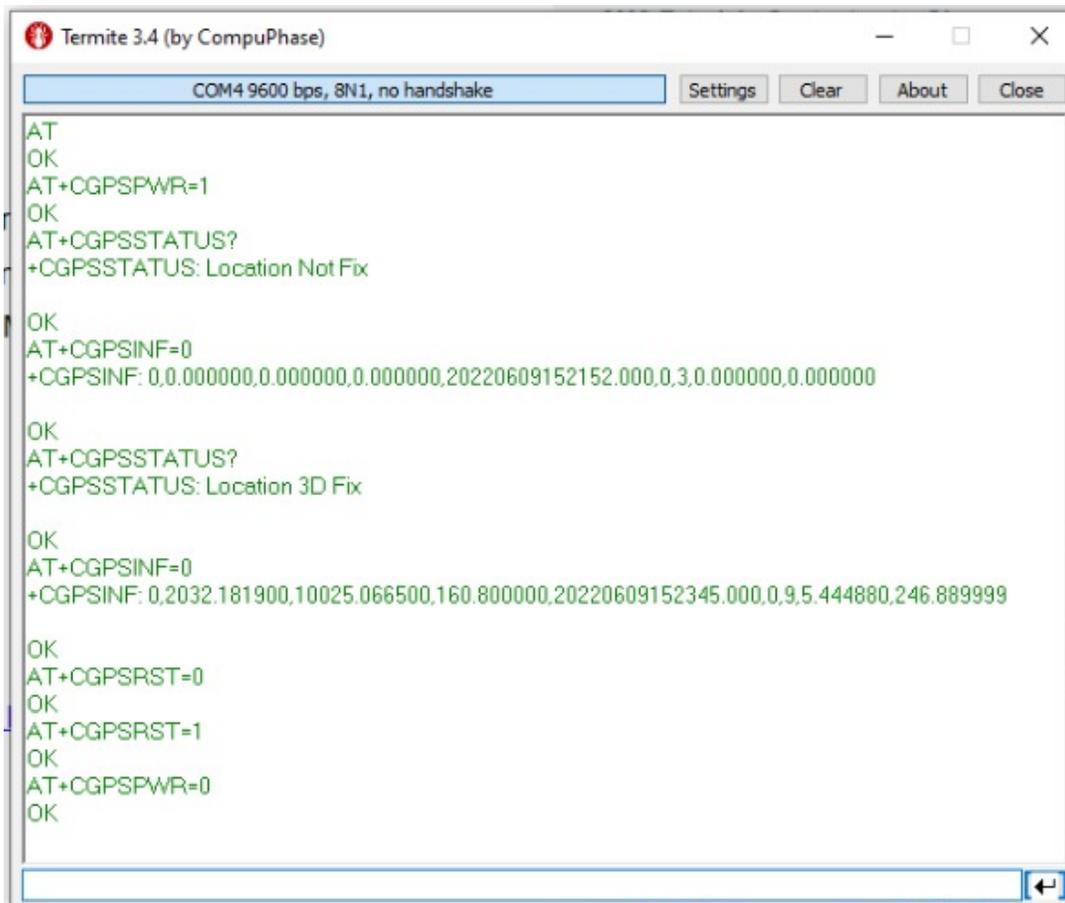


Figura 71. Ubicación del vehículo.

4.4. Pruebas del sistema embebido

Como primera prueba de funcionamiento se utilizó el módulo SIM808 junto con un TTL, como medio de comunicación entre el módulo SIM808 y la computadora, de esta manera se enviaron los comandos AT para observar la respuesta del propio módulo. Como se puede observar en la Figura 72 el módulo envía una respuesta a cualquier comando AT. En la Tabla 16 se muestran los comandos AT utilizados, su función y la respuesta del módulo.



```
AT
OK
AT+CGPSPWR=1
OK
AT+CGPSSTATUS?
+CGPSSTATUS: Location Not Fix

OK
AT+CGPSINF=0
+CGPSINF: 0.0.000000,0.000000,0.000000,20220609152152.000,0.3,0.000000,0.000000

OK
AT+CGPSSTATUS?
+CGPSSTATUS: Location 3D Fix

OK
AT+CGPSINF=0
+CGPSINF: 0.2032.181900,10025.066500,160.800000,20220609152345.000,0.9,5.444880,246.889999

OK
AT+CGPSRST=0
OK
AT+CGPSRST=1
OK
AT+CGPSPWR=0
OK
```

Figura 72. Respuestas a comando AT del módulo SIM808.

Tabla 16. Comandos AT y respuesta del módulo de comunicación SIM808.

Comando AT	Respuesta	Función
AT+CMGF=1	OK	Activar función GSM
AT+CMGS="\'+524424501556\'"	-	Numero a enviar SMS
AT	OK	Verificar conexión con el módulo
AT+CGPSPWR=1	OK	Activar función GPS
AT+CGPSSTATUS?	STATUS 3D FIX	Verificar si el GPS tiene conexión
AT+CGPSINF=0	NEMA	Obtener ubicación
AT+CGPSPWR=0	OK	Desactivar función GPS
AT+CMGD=1	OK	Borrar SMS
AT+CMGR=1	GR	Verificar si existe algún mensaje disponible

Posteriormente se hicieron distintas pruebas de funcionamiento en una protoboard, utilizando el diagrama electrónico de conexiones se acoplaron los componentes donde se realizaron las primeras pruebas, esto se puede apreciar en la Figura 73.

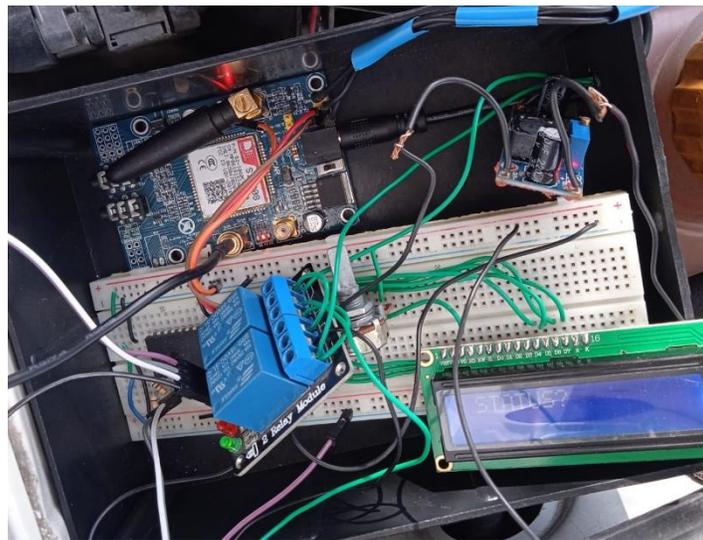


Figura 73. Placa de prueba para el sistema embebido.

Una vez comprobando el correcto acoplamiento de los componentes se procedió a realizar distintas pruebas, como se puede apreciar en la Figura 74 se programó el microcontrolador para que pudiera enviar los comandos AT respectivos para obtener la localización del dispositivo.



Figura 74. Cordenadas del vehículo.

Al comprobar el correcto funcionamiento del GPS se procedió a programar el microcontrolador de manera que pueda enviar los comando AT necesarios para enviar mensajes SMS, como se puede observar en la Figura 75 se envía la ubicación obtenida por el GPS del módulo.



Figura 75. Enviando SMS al usuario.

Una vez obteniendo la ubicación y el envío de mensajes SMS de manera manual corresponde poder leer mensajes SMS. En la Figura 76 se puede observar el SMS enviado al número del chip de pruebas.

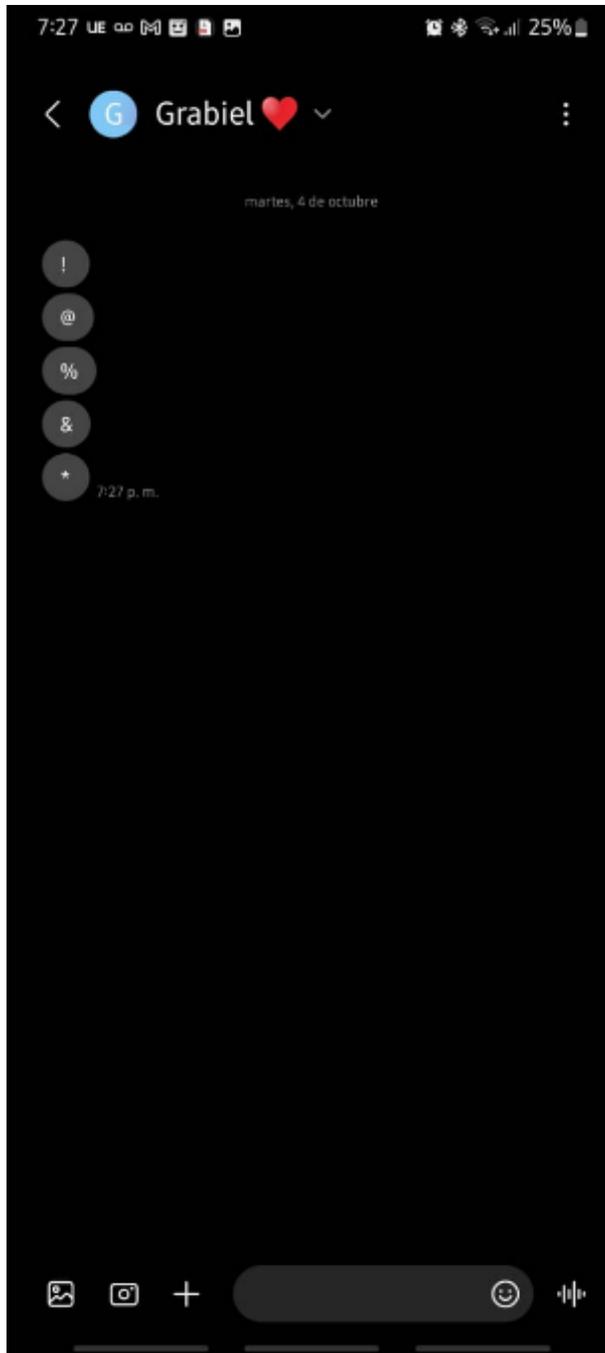


Figura 76. Mensajes clave para la realización de actividades.

Posteriormente comprobando la correcta lectura de los mensajes SMS se procede a programar el microcontrolador para que pueda realizar lo necesario para obtener la ubicación del GPS y ser enviada por mensaje SMS una vez reciba un mensaje en específico por parte del usuario tal como se muestra en la Figura 77.

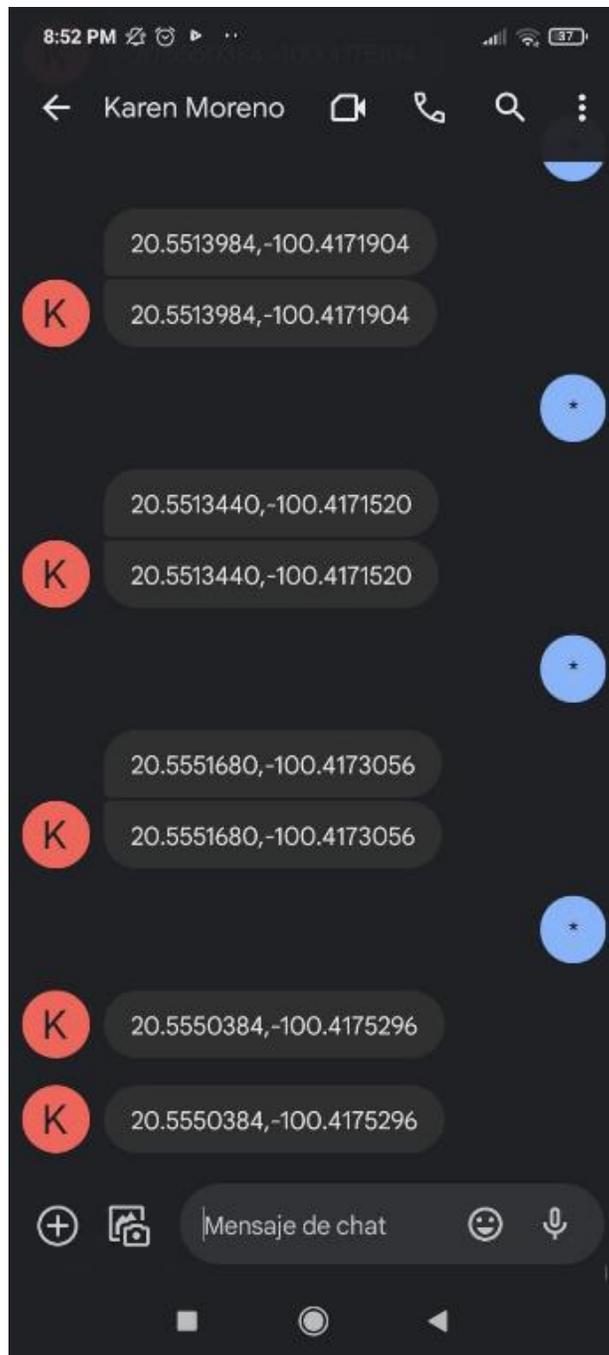


Figura 77. Respuesta del sistema embebido.

4.5. Pruebas de la Aplicación desarrollada

Como primera prueba de la aplicación es considerar el filtro de seguridad del usuario, donde es necesario colocar el usuario y contraseñas para poder acceder a las siguientes actividades de la aplicación. Como se muestra en la Figura 78 el usuario es “Gabo” y la contraseña es “1234” al presionar el botón “entrar” se accede a la siguiente actividad.

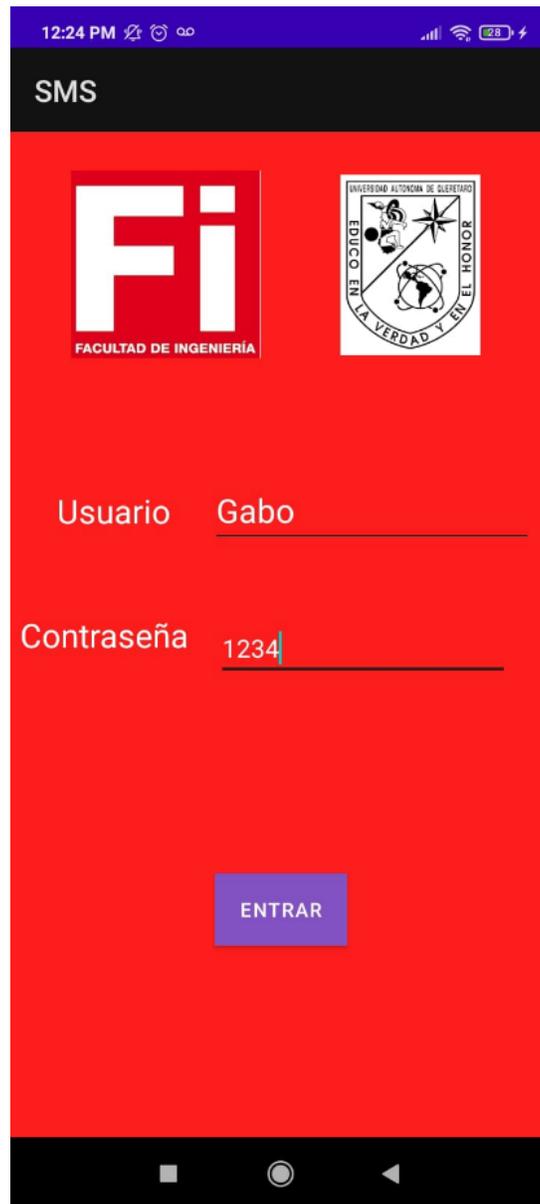


Figura 78. Colocación de usuario y contraseña.

De manera contraria se muestra un mensaje “acceso denegado” en caso de tener algún error con la contraseña o usuario como se puede apreciar en la Figura 79.

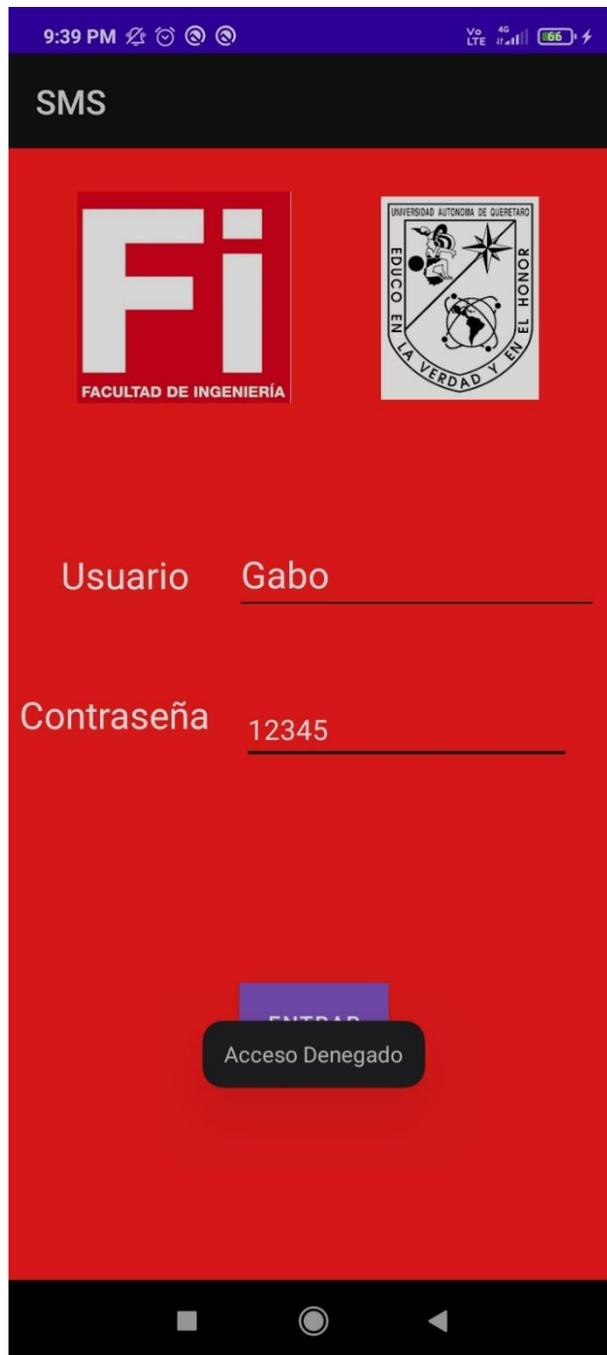


Figura 79. Acceso denegado por contraseña incorrecta.

Como segunda prueba es necesario poder enviar mensajes SMS predeterminados a través de la aplicación. Esto se muestra en la Figura 80 donde podemos apreciar los mensajes predeterminados que le llegan a otro chip, fue necesario considerar que los mensajes

predeterminados deben ser los mismos que se programaron en el microcontrolador para que los pueda leer y ejecutar las acciones programadas anteriormente.

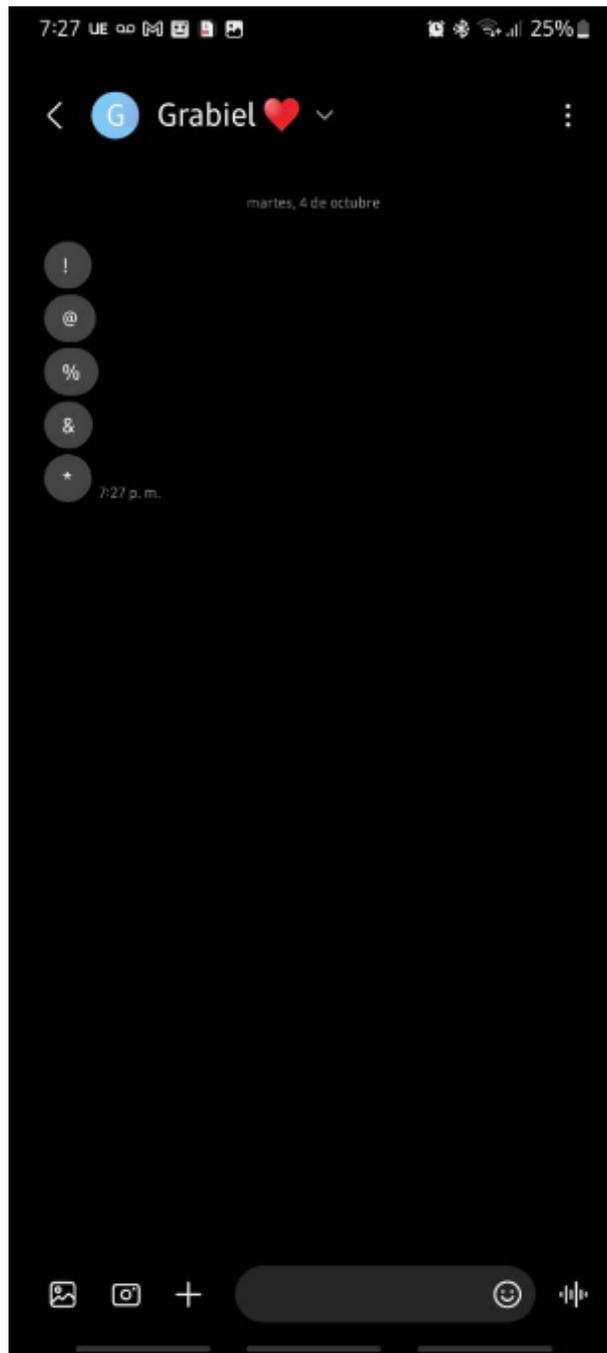


Figura 80. Mensajes predeterminados para el sistema embebido.

Posterior a enviar mensajes SMS corresponde la lectura de los mensajes SMS desde la propia aplicación, para poder comprobar esto al recibir un mensaje SMS se muestra en un

“TextView” como se señala y muestra en la Figura 81 donde se aprecia claramente la ubicación enviada desde el sistema embebido.



Figura 81. Lectura de SMS recibidos mediante la APP desarrollada.

Como última prueba independiente de la aplicación desarrollada es el uso del Mapa, una vez obtenida la ubicación del sistema embebido es de suma importancia mostrarla en un

mapa donde sea mayormente visual la posición del sistema embebido. Esto se logra mediante otra actividad de la aplicación, como se aprecia en la Figura 82 podemos observar la ubicación del sistema embebido en un mapa con un pin colocado para apreciar mayormente los detalles.

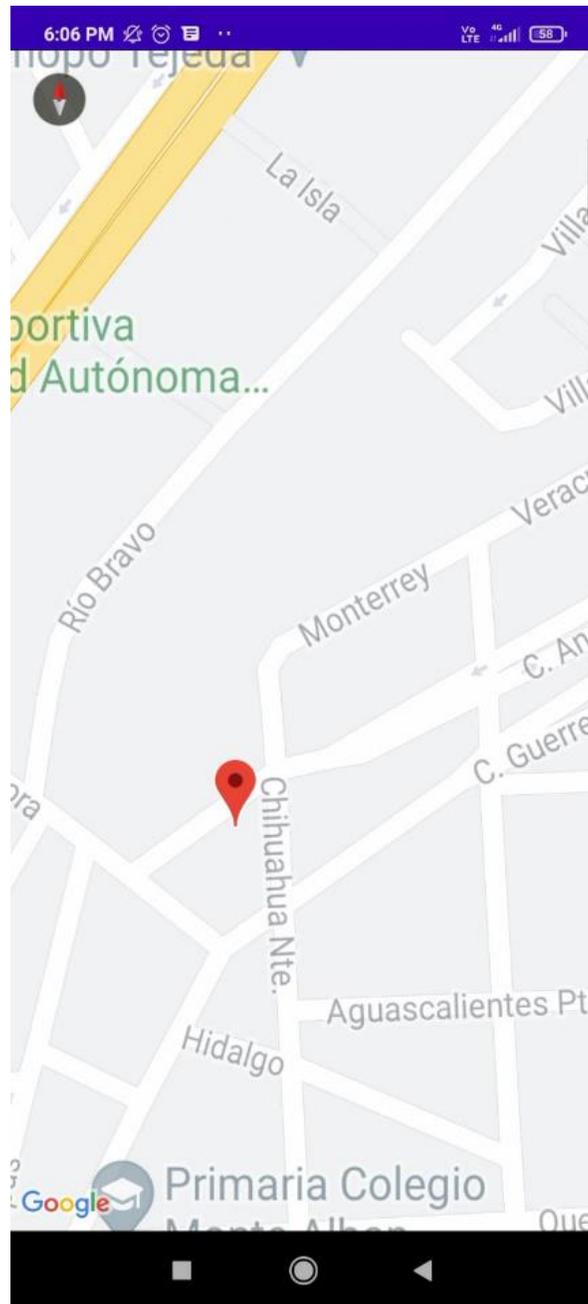


Figura 82. Colocación de Pin en las coordenadas recibidas vía SMS.

4.6. Instalación del sistema embebido

Al tener un sistema embebido funcionando apropiadamente se procede a su instalación dentro del vehículo. Para esto se ha establecido los puntos principales para evitar daños al sistema embebido.

4.6.1. Instalación de la fuente de alimentación

Como primer paso es instalar el circuito LM2596 a la batería y calibrar con su potenciómetro multivuelta a la salida deseada, para esto es verificar la salida de voltaje con un multímetro como se observa en la Figura 83.



Figura 83. Calibración del módulo LM2596.

Posteriormente se enciende el vehículo y se toma la lectura de voltaje de salida con el multímetro como se aprecia en la Figura 84 se obtiene el mismo voltaje de salida. Para esta

prueba de igual manera se revolucionó el vehículo hasta 3000 rpm esto con la finalidad de comprobar el correcto funcionamiento del circuito LM2596.



Figura 84. Comprobación de salida de voltaje del módulo LM2596 con 3000rpm

Una vez realizadas estas pruebas se tiene una salida de voltaje constante a pesar de tener una entrada de voltaje variable. Como se muestra en la Tabla 17 se aprecian las variaciones de voltaje de entrada al módulo LM2596 y la salida de voltaje del módulo LM2596 en relación con los rpm del vehículo.

Tabla 17. Comparación de voltaje de entrada y salida respecto a los rpm del vehículo.

RPM	Voltaje de entrada	Voltaje de salida
0	11.8	4.83
1000	13.2	4.82
2000	13.6	4.83
3000	14.1	4.84
4000	14.4	4.83

4.6.2. Instalación módulo de relevadores

Una vez teniendo en optimo funcionamiento la fuente alimentación se procedió a la instalación del módulo de relevadores. Como instalación primaria con fines de pruebas de funcionamiento se realizaron las conexiones que se muestran en la Figura 85.

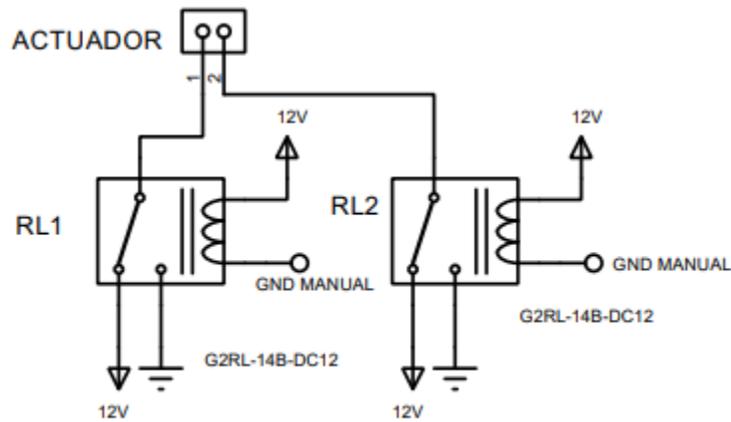


Figura 85. Conexiones principales.

Para poder realizar las pruebas correspondientes de cambio de posición de los relevadores se conectaron las señales directo al negativo de la batería del vehículo como se muestra en la Figura 85. Conexiones principales. todo esto de manera manual. Gracias a este método se pudo corroborar el correcto funcionamiento del módulo de relevadores con señal negativa.

4.6.3. Conexión entre actuador de seguros y módulo de relevadores

Al corroborar el correcto funcionamiento del módulo de relevadores se procede a realizar las conexiones pertinentes entre el actuador de los seguros automáticos del vehículo en cuestión junto con el módulo de relevadores ya conectados.

Como se muestra en la Figura 86 únicamente es necesario unir las terminales del actuador junto con los comunes de cada relevador.

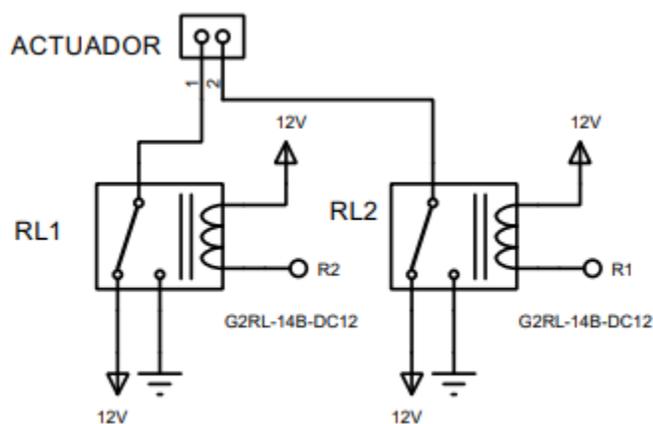


Figura 86. Unión de terminales entre el actuador y módulo de relevadores.

En la Figura 87 podemos observar las conexiones anteriormente mencionadas. En esta imagen se puede apreciar que tienen una protección aislante ambas terminales, esto para evitar algún contacto con el chasis de la puerta. Para este caso se instaló un recubrimiento de termofit transparente como aislante de electricidad.



Figura 87. Conexión física entre relavadores y terminales del actuador.

4.6.4. Alimentación de módulo SIM808

El módulo SIM808 cuenta con una gran ventaja en la alimentación debido a que gracias a sus especificaciones cuenta con un regulador de voltaje interno, por lo cual el módulo de comunicación puede ser alimentado directo de la batería sin sufrir daño ya que su rango de voltaje de entrada puede ser de RANGO RANGO. Las conexiones necesarias de alimentación se muestran en la Figura 88.

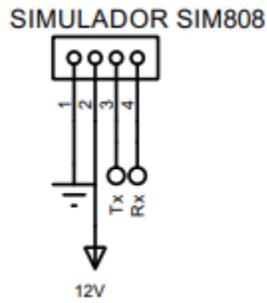


Figura 88. Conexiones mínimas del módulo SIM808.

En la Figura 89 se aprecia que la conexión de alimentación es mediante una conexión DC044 por lo cual es necesario tener dicha conexión. Considerando el botón de encendido se necesita realizar la conexión de alimentación y realizar una prueba de funcionamiento.

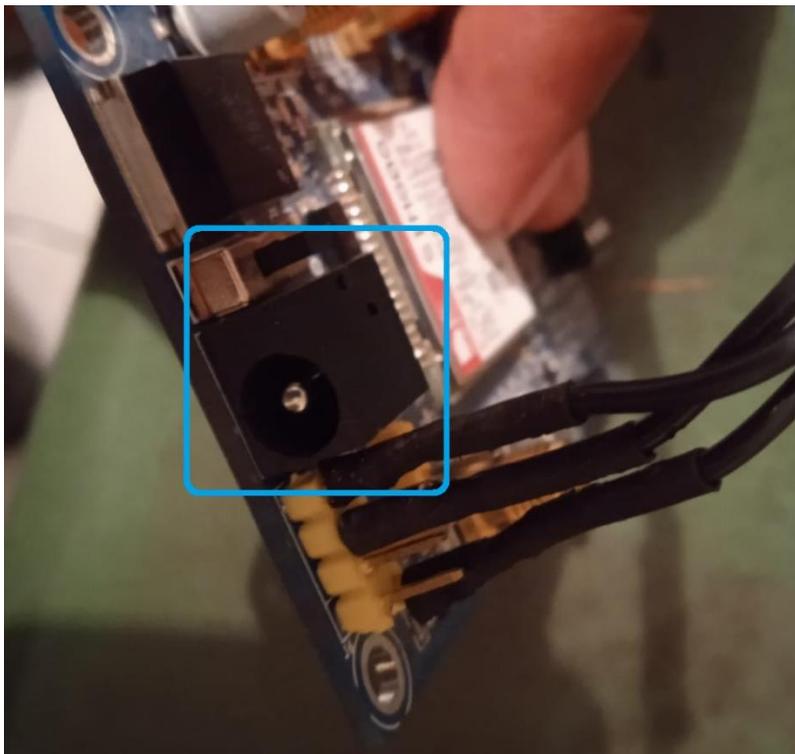


Figura 89. Conexión DC044 para alimentación del módulo SIM808.

Como se aprecia en la Figura 90 siguiendo el diagrama de conexiones del módulo SIM808 se coloca la alimentación de manera correspondiente.



Figura 90. Conexión física de alimentación al módulo SIM808.

En la Figura 91 podemos observar que el switch de encendido está activado. De igual manera se puede apreciar un led rojo encendido lo cual, según el manual del fabricante, se tiene una alimentación adecuada pero el módulo aún no este encendido.

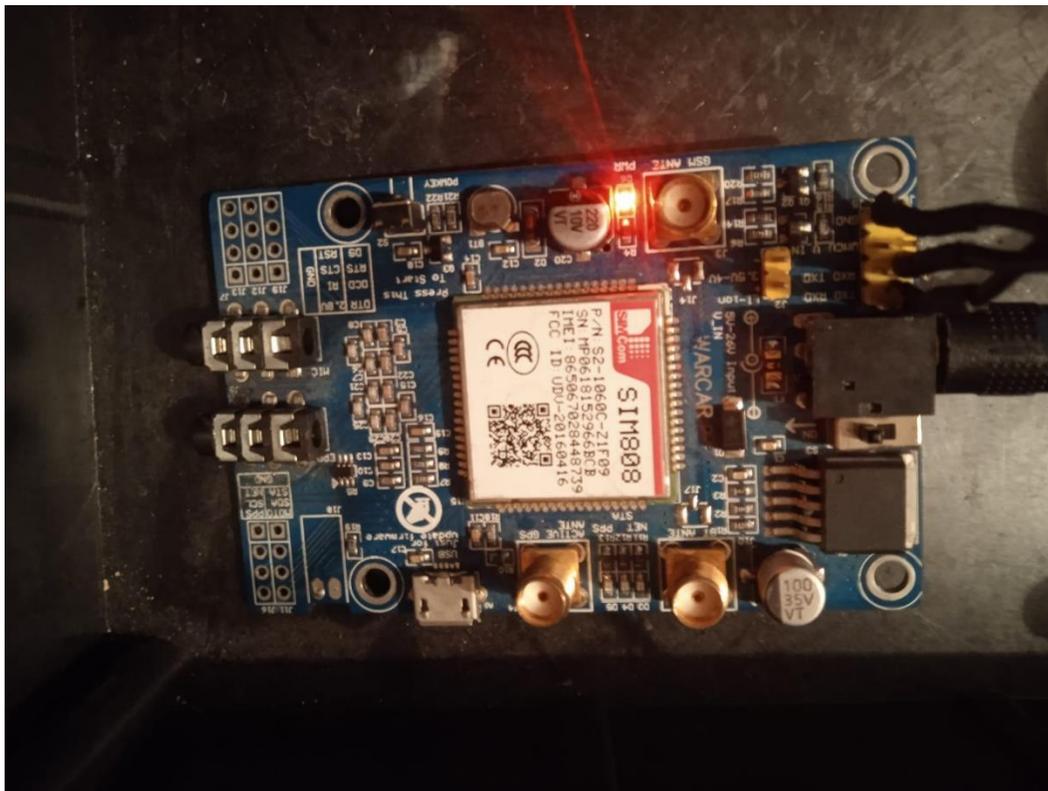


Figura 91. Correcta alimentación del módulo SIM808.

En la Figura 92 podemos ver otro LED anteriormente mencionado con una luz más brillante lo cual nos indica que el módulo SIM808 está encendido correctamente. Por lo cual

únicamente faltaría conectar sus antenas correspondientes y esperar a la sincronización con los satélites.

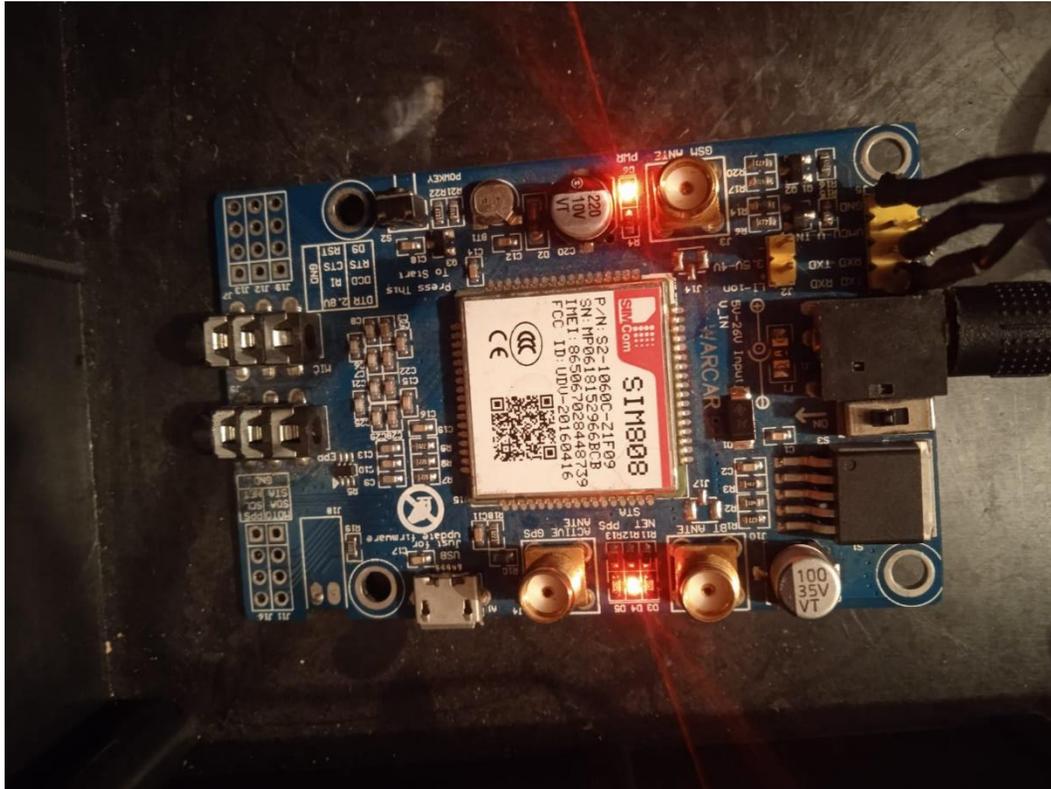


Figura 92. Correcto encendido del módulo SIM808.

4.6.5. Alimentación de la Placa perforada

La conexión entre la PCB y su alimentación se muestra en la Figura 93. Como se aprecia es necesario conectar la salida de voltaje del módulo LM2596 con la entrada de voltaje de la placa perforada.

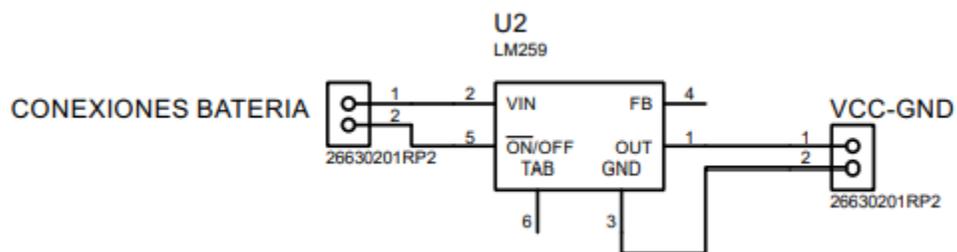


Figura 93. Diagrama de conexiones de alimentación del sistema embebido.

En la Figura 94 se muestran las conexiones realizadas en campo. Debido a que ambos componentes se ubican dentro del sistema embebido no hubo mayor inconveniente.

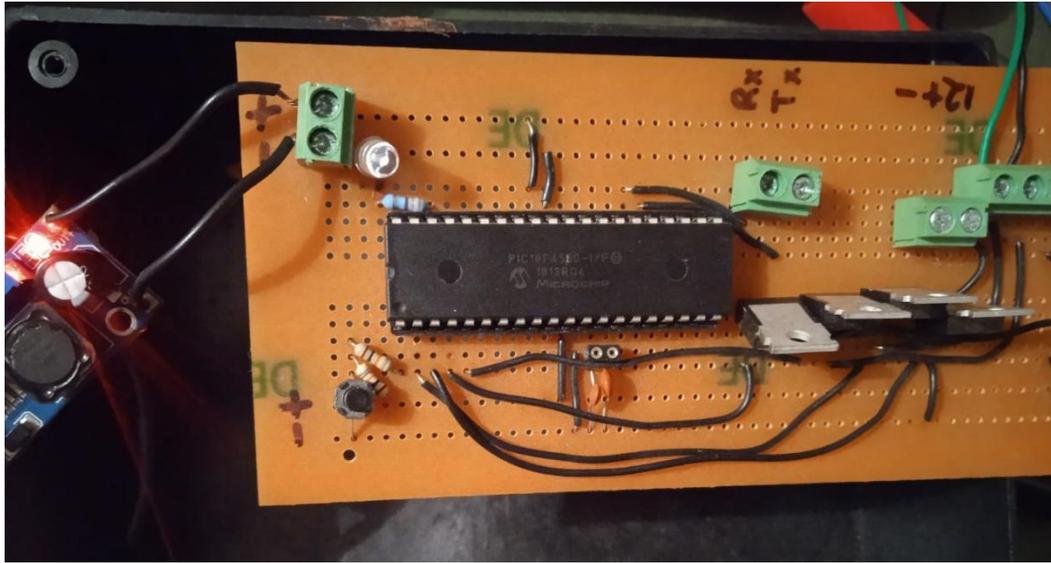


Figura 94. Conexiones físicas de alimentación del sistema embebido.

4.6.6. Conexiones entre la placa perforada y el módulo de relevadores

En la Figura 95 podemos observar las conexiones correspondientes entre la placa perforada y el módulo de relevadores. Para este punto se descartan conexiones de alimentación por cada uno de los componentes por lo cual únicamente tenemos la señal de la placa perforada con los pines correspondientes del módulo de relevadores.

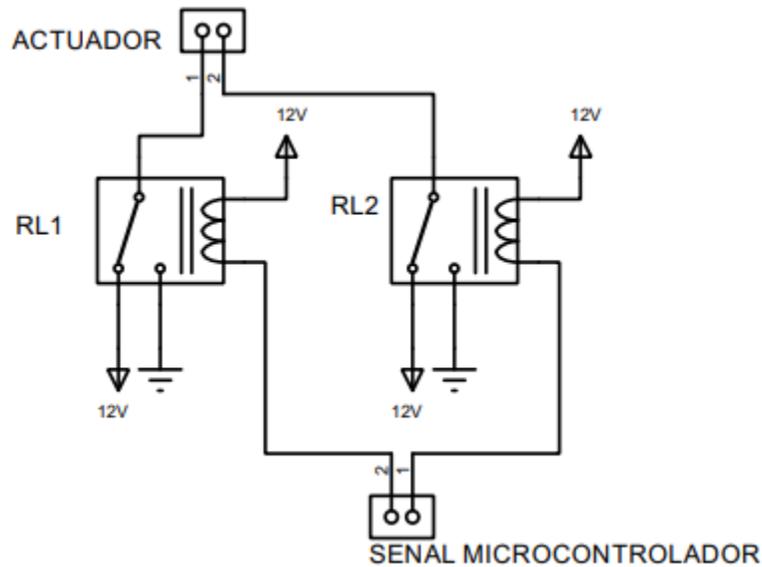


Figura 95. Diagrama de conexiones entre el microcontrolador y módulo de relevadores.

Como se muestra en la Figura 96 se tienen ambos elementos conectados de manera correcta. En este paso es necesario considerar la distancia entre el módulo de relevadores y el actuador de los seguros automáticos.

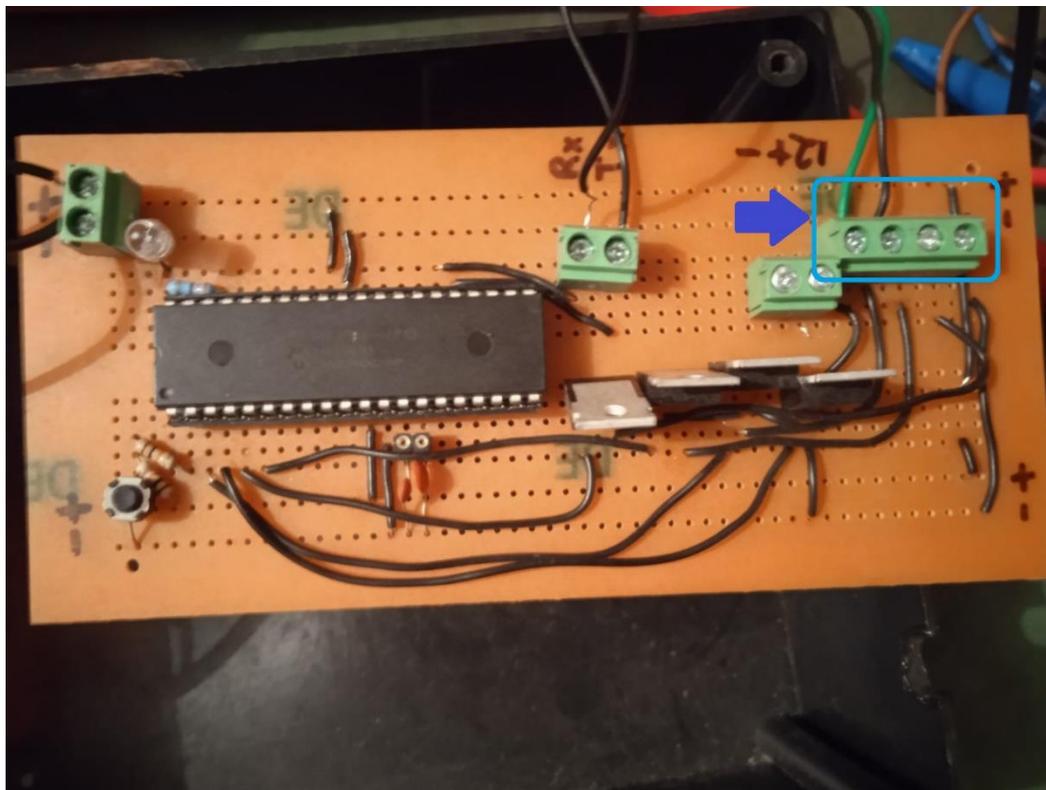


Figura 96. Conexión física entre módulo de relevadores y señales del sistema embebido.

Se busca una instalación visualmente estética por que los cables de las señales para el actuador pasan por debajo del tablero del vehículo. Como se resalta en la Figura 97 los cables de señal a los actuadores pasan a un costado del bus de cables del tablero.

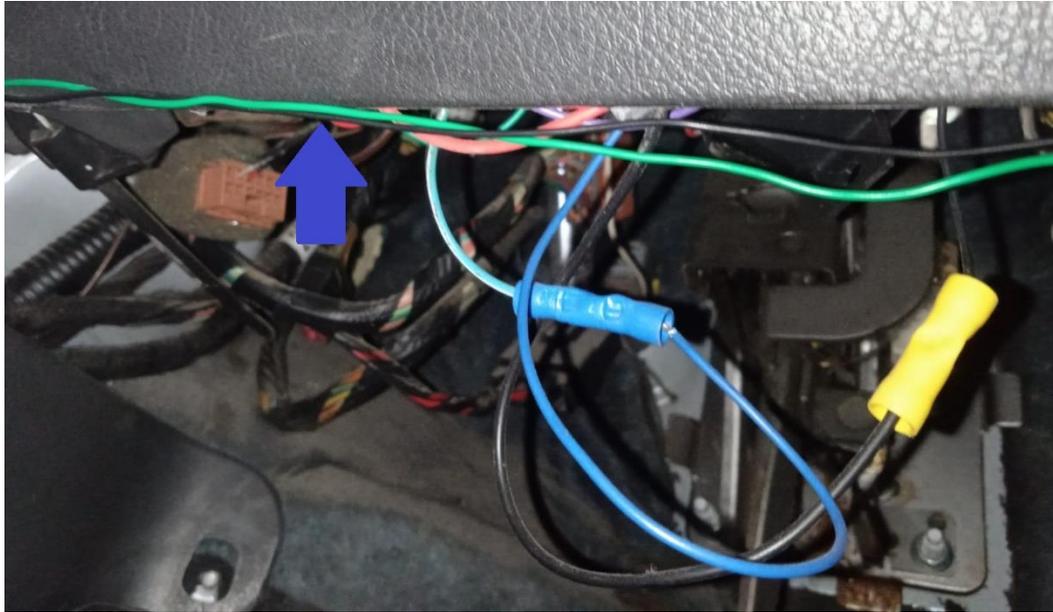


Figura 97. Cableado de las señales para el actuador.

4.7. Prueba de funcionamiento

Una vez instalados todos los componentes se procede a conectar las antenas correspondientes como se muestra en la Figura 98. Para este punto es necesario recordar que una vez conectado el sistema embebido nos va a enviar un mensaje de conexión correcta con los satélites correspondientes. Además, se le agregó un led de visualización que enciende por 2 segundo cada que ejecuta una acción.



Figura 98. Conexión de antenas con módulo SIM808.

Se esperó un aproximado de 1 hora donde no se recibió el mensaje verificación por parte del sistema embebido por lo cual se procede a revisar las alimentaciones para verificar que esté en optimas condiciones, esto se puede apreciar en la Figura 99.

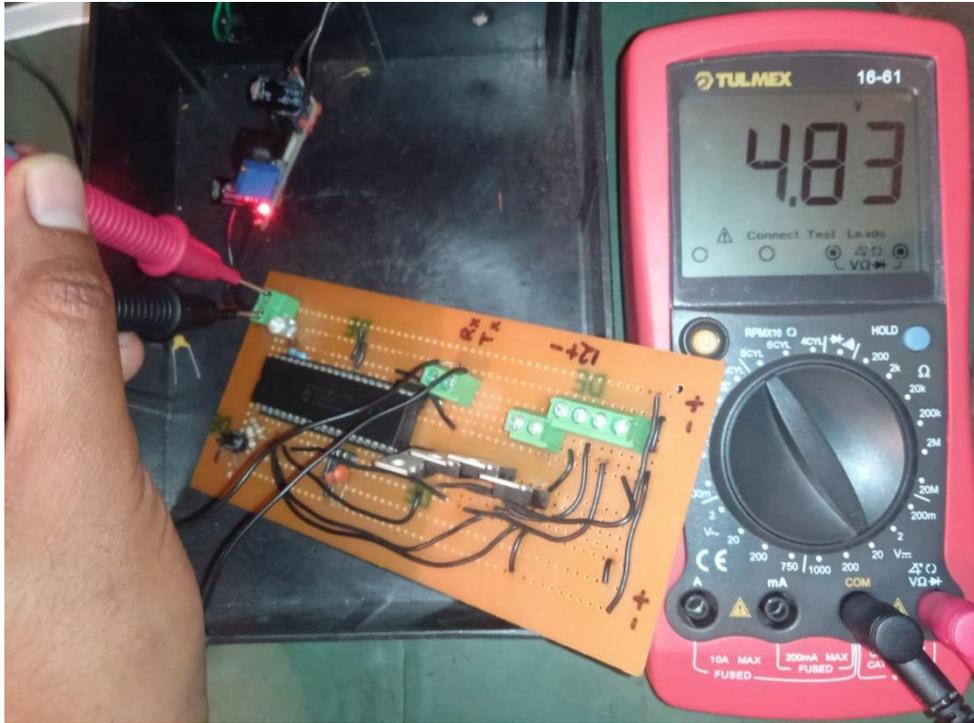


Figura 99. Revisión de entrada de voltaje al sistema embebido.

Como se tiene una buena alimentación en todas las partes de la placa perforada y en los elementos del sistema embebido se procedió a verificar con el led de visualización y el código para corroborar los pasos del sistema embebido por lo que se detectó que el microcontrolador no avanzaba en la línea de comprobación de comunicación con el módulo SIM808. Para verificar este paso comprobamos la continuidad entre las terminales con un multímetro esto se visualiza en la Figura 100. Como resultado de esta prueba se obtiene que existe una continuidad correcta entre las terminales, sin embargo, se detecta que las terminales están alejadas entre ellas por lo que se sospecha que se puede tener interferencia por ruido lo cual influye en la correcta comunicación de los elementos.

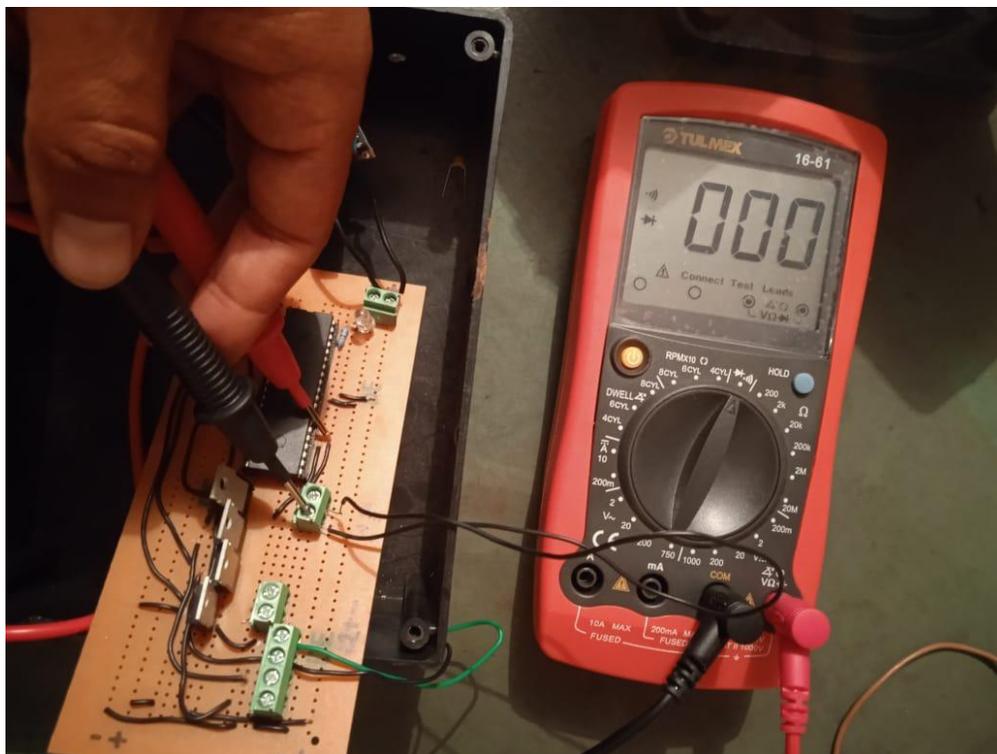


Figura 100. Comprobación de continuidad entre terminales.

Para solucionar este inconveniente se ha optado por realizar nuevamente una placa perforada contemplando la distancia entre todas las terminales de comunicación y agregando la pantalla LCD 16x2 como se muestra en la Figura 101 esto último con la finalidad de tener un apoyo visual de la etapa en la que se encuentra el sistema embebido y poder detectar algún otro inconveniente de manera rápida y eficaz.

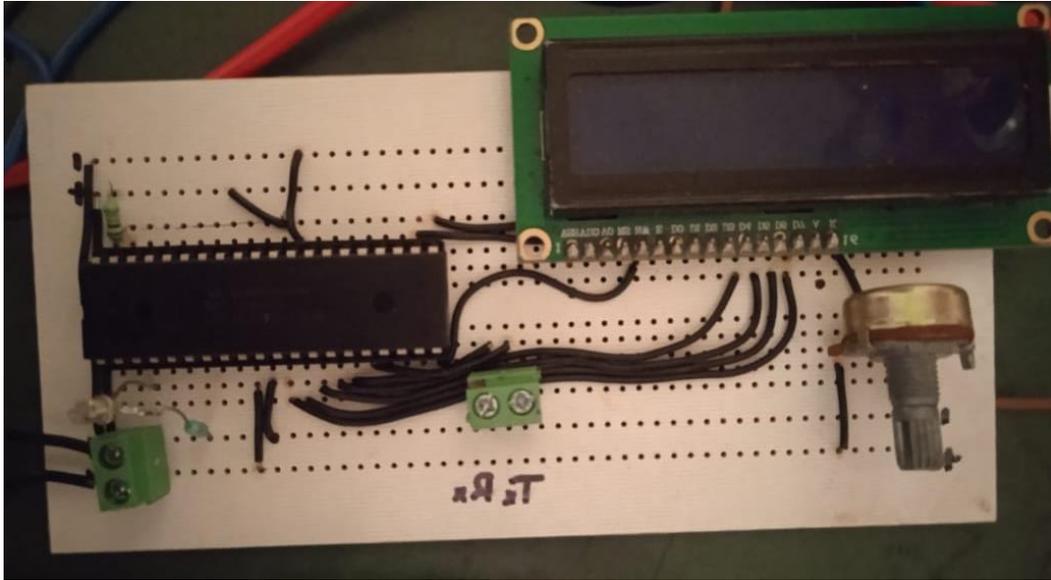


Figura 101. PCB con pantalla LCD 16x2.

Posteriormente se vuelven a instalar los elementos del sistema embebido y se revisa su correcto funcionamiento. Como se muestra en la Figura 102 ya se encuentra una comunicación apropiada entre los elementos del sistema embebido.

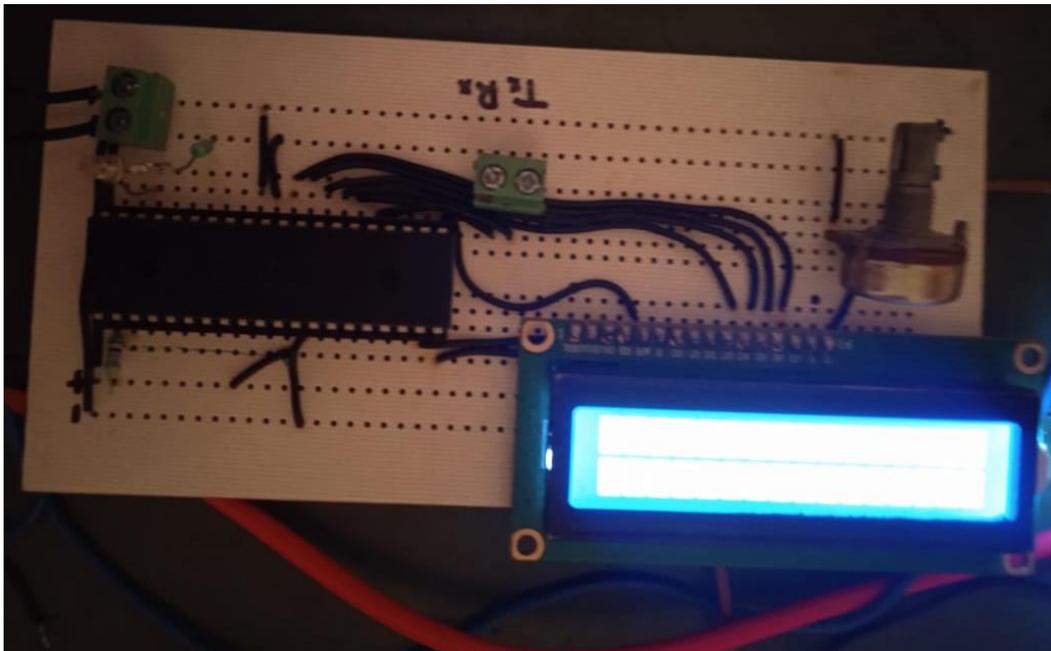


Figura 102. Prueba de encendido de placa PCB.

Se procede a esperar lo necesario para recibir el mensaje de verificación por parte del sistema embebido como se muestra en la Figura 103. En esta prueba fue tiempo aproximado de 23 min en conectarse a los satélites correspondientes.



Figura 103. Conexión correcta entre placa PCB y el módulo SIM808.

4.8. Tiempo de conexión entre APP y el sistema embebido

En la Tabla 18 se muestran los tiempos de conexión con los satélites. Para poder tomar estos valores se conectó y el módulo SIM808 y se desconectó por 5 minutos para volver a conectar nuevamente y de esta manera subsecuente. Con la finalidad de cada vez desconectando el módulo de comunicación se pierda la conexión con los satélites.

Tabla 18. Tiempos de sincronización satélites.

Cantidad de veces conectado	Tiempo de sincronización con satélites
1	23 min
2	13 min
3	17 min
4	13 min
5	15 min
6	15 min
7	16 min
8	14 min
9	21 min
10	16 min

Una vez considerando el tiempo de conexión entre el módulo SIM808 y los satélites se procede a comprobar el tiempo de transmisión y recepción de información mediante la APP desarrollada y el sistema embebido. Por lo cual mediante la APP desarrollada se envía cada acción individualmente y con ayuda de la pantalla LCD 16x2 se verifica el tiempo de recepción. Cada acción se envió 10 diferentes ocasiones para poder tener un promedio de

tiempo de recepción. Como se muestra en la Figura 104 El promedio de recepción de información es de aproximadamente 15 minutos.

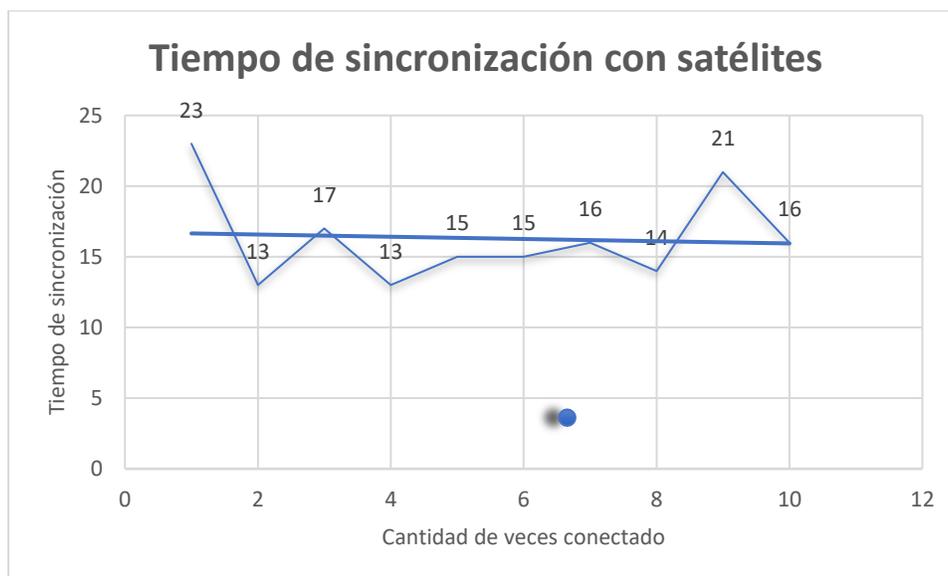


Figura 104. Tiempo de sincronización con satélites.

Como siguiente paso es verificar el tiempo en el cual se ejecuta la acción correspondiente desde que es accionado por el usuario desde la APP desarrollada para esto se considera el tiempo una vez presionado el botón directamente en la APP desarrollada y se detiene el tiempo una vez que la acción a concluido correctamente. Como se muestra en la Tabla 19 se envió en reiteradas ocasiones la acción para elevar seguros y los tiempos de respuesta.

Tabla 19. Tiempo de respuesta en acción Subir seguros.

Cantidad de veces activando la acción "Subir Seguros"	Tiempo de Respuesta
1	3 min
2	2 min
3	2 min
4	2 min
5	3 min
6	2 min
7	2 min
8	4 min
9	3 min
10	2 min

La toma de tiempos de respuesta se realizó para cada acción. Como se coloca en la Tabla 20 se muestran los tiempos de respuesta para la acción bajar seguros. Para los valores de esta se consideraron los mismos puntos que en la tabla anterior.

Tabla 20. Tiempo de respuesta en acción bajar seguros.

Cantidad de veces activando la acción “Bajar Seguros”	Tiempo de Respuesta
1	2 min
2	3 min
3	2 min
4	3 min
5	2 min
6	2 min
7	2 min
8	2 min
9	3 min
10	2 min

La siguiente acción a medir fue subir vidrios, para esta acción se consideró el tiempo desde que se ejecuta la acción de la APP hasta que comienza la elevación del vidrio. El tiempo de ejecución se muestra en la Tabla 21.

Tabla 21. Tiempo de respuesta en acción subir vidrios.

Cantidad de veces activando la acción “Subir Vidrios”	Tiempo de Respuesta
1	1 min
2	3 min
3	2 min
4	3 min
5	1 min
6	2 min
7	2 min
8	3 min
9	3 min
10	3 min

Como siguiente acción se considera el bajar vidrios, al igual que en la tabla anterior en la Tabla 22 se considera el tiempo desde la ejecución desde la APP hasta comenzar la ejecución de bajar los vidrios.

Tabla 22. Tiempo de respuesta en acción bajar vidrios.

Cantidad de veces activando la acción “Bajar Vidrios”	Tiempo de Respuesta
1	3 min
2	2 min
3	2 min
4	1 min
5	2 min
6	2 min
7	2 min
8	3 min
9	1 min
10	3 min

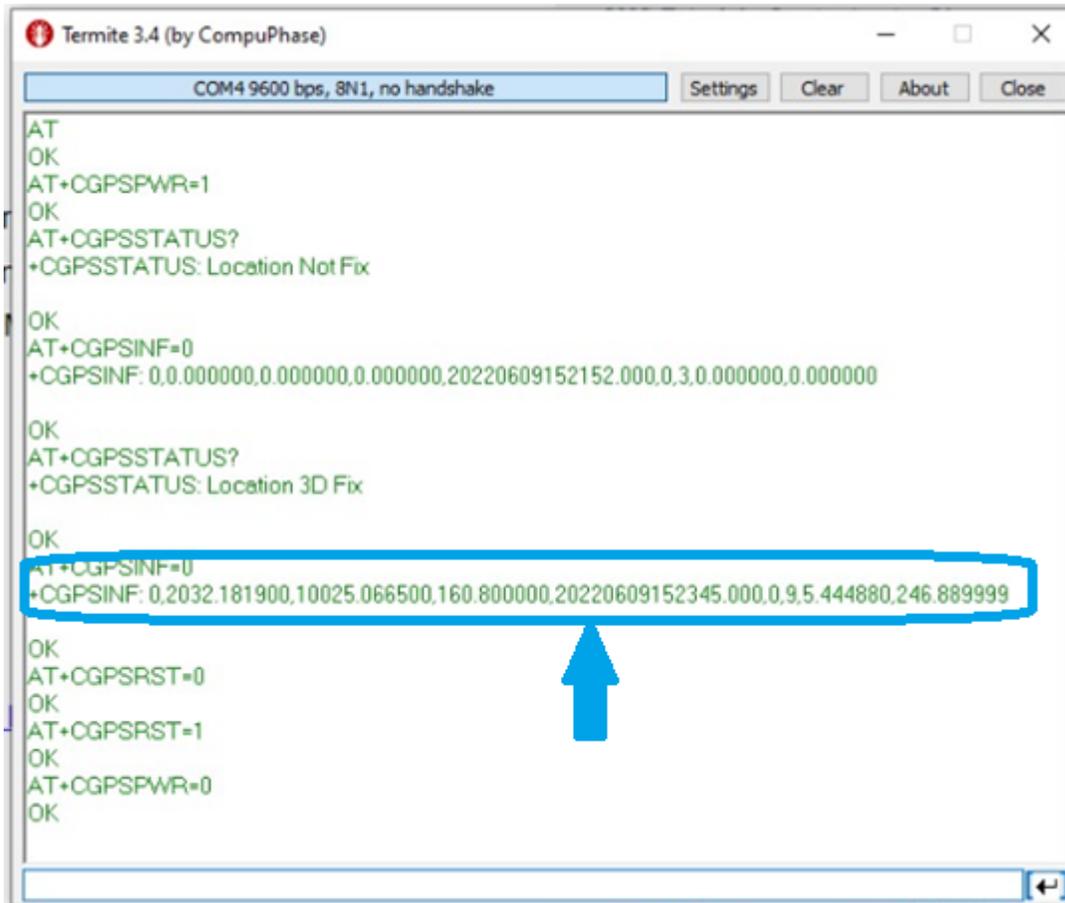
La acción donde se tiempo un tiempo de respuesta más longevo es el adquirir la ubicación del vehículo asegurado que tiene un tiempo de respuesta de aproximadamente 4.7min. Para homogeneizar los tiempos se considera desde que se presiona el botón desde la APP hasta que llega el mensaje con la ubicación del sistema embebido. Lo anterior se consideró para los valores que se muestran en la Tabla 23.

Tabla 23. Tiempo de respuesta en acción Obtener ubicación.

Cantidad de veces activando la acción “Obtener Ubicación”	Tiempo de Respuesta
1	4 min
2	5 min
3	4 min
4	6 min
5	5 min
6	5 min
7	4 min
8	4 min
9	4 min
10	6 min

4.9. Exactitud del sistema embebido

Como último aspecto a considerar es la exactitud y precisión que tiene el sistema embebido. Como se muestra en la Figura 105 el módulo SIM808 recibe las coordenadas con 6 decimales.



```
Termite 3.4 (by CompuPhase)
COM4 9600 bps, 8N1, no handshake
AT
OK
AT+CGPSPWR=1
OK
AT+CGPSSTATUS?
+CGPSSTATUS: Location Not Fix

OK
AT+CGPSINF=0
+CGPSINF: 0,0.000000,0.000000,0.000000,20220609152152.000,0,3,0.000000,0.000000

OK
AT+CGPSSTATUS?
+CGPSSTATUS: Location 3D Fix

OK
+CGPSINF: 0,2032.181900,10025.066500,160.800000,20220609152345.000,0,9,5.444880,246.889999

OK
AT+CGPSRST=0
OK
AT+CGPSRST=1
OK
AT+CGPSPWR=0
OK
```

Figura 105. Total, de coordenadas enviadas por el módulo SIM808.

Durante la programación del microcontrolador se consideró la cantidad de bits que tiene cada variable, al utilizar variables tipo flotantes por practicidad con las funciones de las librerías se ha truncado a 7 decimales por motivos del procesamiento de las coordenadas como se muestra en la Figura 106.

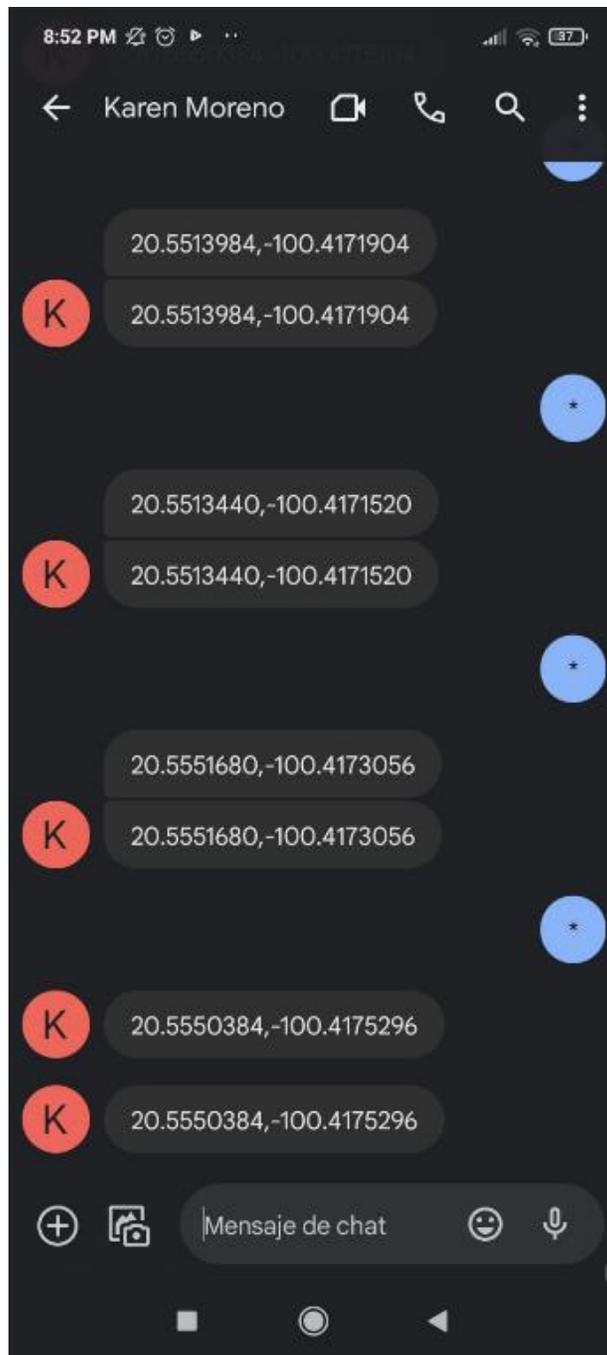


Figura 106. Coordenadas enviadas por el sistema embebido truncadas a 7 decimales.

En la Figura 107 se muestra la exactitud de la ubicación obtenida por el sistema embebido y visualizada por la APP desarrollada en comparación con la exactitud de la ubicación obtenida mediante otra aplicación de fabricada por el mismo aparato móvil de

comunicación y ubicación con un dispositivo móvil. Para esta comparación solo se considera el uso de latitud.

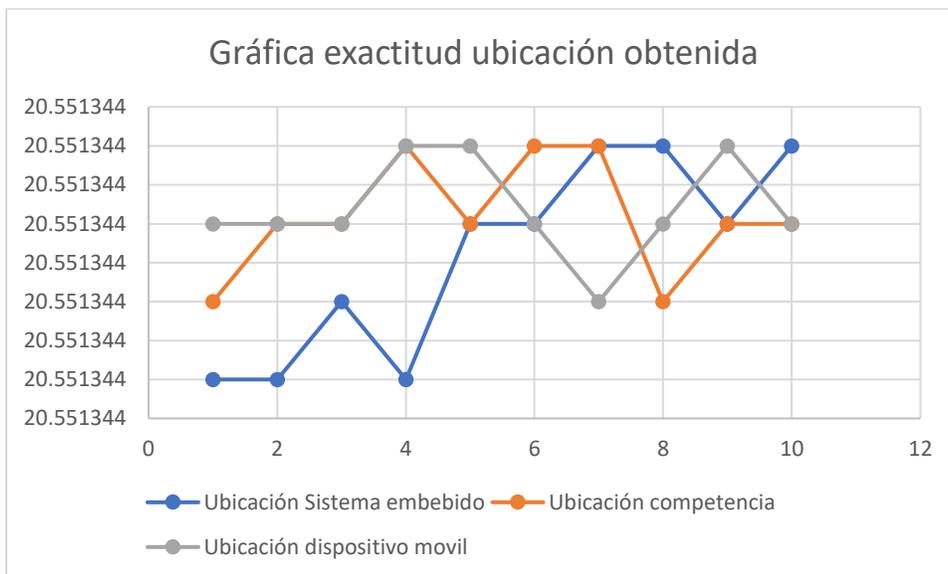


Figura 107. Exactitud obtenida por el sistema embebido.

CAPÍTULO V

5. CONCLUSIONES

De acuerdo con las pruebas realizadas del sistema embebido, se pudo observar una ubicación muy exacta del vehículo asegurado por el sistema embebido. Este sistema cuenta con facilidad de instalación y de uso universal para cualquier vehículo sin depender de alguna marca o modelo en específico, al considerar los voltajes de las señales y los voltajes necesarios para los actuadores.

En comparación con otras alarmas vehiculares, con las mismas características, podemos observar una disminución en el precio de un 20%. A pesar de considerar un módulo de comunicación con uno de los precios más altos del mercado. En este punto se tiene una ventaja para la venta y distribución del sistema embebido.

Contemplando que en Querétaro se tiene un promedio de vehículos de gama media y baja esto implica que el sistema embebido es compatible con la mayor parte de los vehículos propios de la ciudad.

Al realizar las pruebas en campo se tiene un promedio de tiempo de 2 min de respuesta desde el envío de la acción por parte del usuario hasta que el actuador haya hecho su función correspondiente; sin embargo, para el uso de la ubicación se tiene un promedio de 5 min para la recepción de la acción, procesar la ubicación y el envío de la misma al dispositivo móvil del usuario.

La aplicación móvil es amigable con el usuario, fácil de utilizar y gracias a sus filtros es segura para el propio usuario. Con la misma aplicación el usuario no debe considerar la comunicación con el sistema embebido ya que esta realiza todo lo necesario para poder tener una comunicación correcta y eficaz con la misma.

Por otro lado, la aplicación es compatible con el 80% de los celulares con sistema operativo Android considerando que en Querétaro el 70% de la población cuenta con este sistema operativo. Por ende, se concluye una gran oportunidad para implementar este sistema embebido a la población en general.

En comparación de respuesta por parte del sistema embebido con respecto a la competencia en el mercado se tiene una oportunidad de mejora para el sistema debido a que utilizando los mismos componentes se pueden realizar distintas modificaciones para mejorar este aspecto sin incrementar el costo del propio sistema. A su vez queda la oportunidad de una mejora en su diseño electrónico para optimizar el espacio y los componentes necesarios para una mejora en precio y diseño.

REFERENCIAS

A. A. Ahmed, A. M. E. Ahmed, A. H. Mohammed and M. A. A. Akram, "Design and implementation of vehicle tracking and theft control system," *2015 International Conference on Computing, Control, Networking, Electronics and Embedded Systems Engineering (ICCNEEE)*, Khartoum, Sudan, 2015, pp. 181-186, doi: 10.1109/ICCNEEE.2015.7381358.

Aceves Fernández, M. & Ramos Arreguin, J. 2011. Fundamentos de sistemas embebidos: Mediante Lenguajes descriptivos de hardware. Asociacion Mexicana de Mecatronica.

Camacho, J. (2019, junio 13). En Puebla, Queretaro y Jalisco crece el robo a vehículos. Recuperado 25 de abril de 2020, de <https://www.elfinanciero.com.mx/nacional/en-puebla-queretaro-y-jalisco-crece-el-robo-a-vehiculos>

Flores, F. F. Francisco. (2019, 30 mayo). 56% de los robos de autos se cometen en la capital. Recuperado 4 abril, 2020, de <http://www.eluniversalqueretaro.mx/metropoli/56-de-los-robos-de-autos-se-cometen-en-la-capital>

- Newman, G. (2020, enero 2). Delincuencia relacionada con los vehículos. Recuperado 25 de abril de 2020, de <https://www.interpol.int/es/Delitos/Delincuencia-relacionada-con-los-vehiculos>.
- Newman, G. (2020b, enero 2). Lucha contra la delincuencia relacionada con los vehículos. Recuperado 4 de abril de 2020, de <https://www.interpol.int/es/Delitos/Delincuencia-relacionada-con-los-vehiculos/Lucha-contra-la-delincuencia-relacionada-con-los-vehiculos>.
- Núñez, A. (2018, noviembre 29). Querétaro, tercer lugar en robo de vehículos. Recuperado 25 de abril de 2020, de <https://www.capitalqueretaro.com.mx/local/queretaro-tercer-lugar-en-robo-de-vehiculos/>
- Orozco, G. F. M. (2020). Electrónica básica: Electrónica analógica, digital y de potencia (Spanish Edition). Independently published.
- Ortega, H. (2018, agosto 13). México, único de 126 países con alza en robo de autos. Recuperado 4 de abril de 2020, de <https://www.informador.mx/Mexico-unico-de-126-paises-con-alza-en-robo-de-autos-l201808130002.html>.
- Rivera, A. (2019, noviembre 2). Robo de autos con violencia, imparable en todo el país. Recuperado 11 de abril de 2020, de <https://www.milenio.com/autos/robo-de-autos-con-violencia-imparable-en-todo-el-pais>.
- Santos, F.J. (2007). Modelado y simulación de redes basado en el estándar IEEE 802.16. Proyecto Fin de Carrera (Universidad de Málaga).
- Tixce, C. (2017, junio 12). La alarma automotriz: su historia y los diferentes tipos existentes. Recuperado 25 de abril de 2020, de <https://www.motoryracing.com/coches/noticias/alarma-automotriz-historia-tipos-existentes/>
- Trujillo, E. (2019, octubre 29). Si es miércoles y tu auto es blanco, preocúpate: esto dicen las estadísticas de robo de autos en México. Recuperado 11 de abril de 2020, de <https://www.motorpasion.com.mx/industria/robo-autos-mexico-2019>.
- Villajulca, J. C. (2022, 8 mayo). Control ON/OFF o Todo/Nada. Instrumentacion y Automatizacion Industrial. <https://instrumentacionycontrol.net/control-on-off-o-todo-nada/>

ANEXOS

Los anexos que a continuación se adjuntan se adjuntan son las pruebas en formato de video del sistema embebido utilizado en campo, asimismo dentro de los videos se narran los pasos y los estatus en los cuales se encuentra el sistema embebido.

https://drive.google.com/file/d/17i0buPIZ9uaMYXr9JJFZ4QrJD4Ivwwip/view?usp=share_link

https://drive.google.com/file/d/17htLJuDZjNudEXSE8r8bUlx9brgFTrMy/view?usp=share_link

https://drive.google.com/file/d/1KZ_e1UrqDSha2pxBLpPNIzwaKQkxC67c/view?usp=share_link

https://drive.google.com/file/d/1K_pP5vwbAEMV8k6QOKR620_BQkQfh1XF/view?usp=share_link

https://drive.google.com/file/d/1Kar-y61zdw0YVruON9qq9hAaSG1Nzczl/view?usp=share_link

https://drive.google.com/file/d/1KcZzW_4kGNmiKIYS3pbLW7QnWbxnDgoh/view?usp=share_link

https://drive.google.com/file/d/1KcZzW_4kGNmiKIYS3pbLW7QnWbxnDgoh/view?usp=share_link

https://drive.google.com/file/d/1KcZzW_4kGNmiKIYS3pbLW7QnWbxnDgoh/view?usp=share_link