

**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE QUITO**

CARRERA: INGENIERÍA ELECTRÓNICA

Tesis previa a la obtención del título de: INGENIEROS ELECTRÓNICOS

**TEMA:
DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE ALERTA CON VOZ Y
REFERENCIACIÓN CON MÓDULO GPS PARA PICO Y PLACA EN LA
CIUDAD DE QUITO.**

AUTORES:

JOHANNA ELIZABETH LÓPEZ TOAPANTA

CRISTIAN FERNANDO PEREIRA VIVES

DIRECTOR:

MANUEL RAFAEL JAYA DUCHE

Quito, marzo del 2015

**DECLARATORIA DE RESPONSABILIDAD Y AUTORIZACIÓN DE USO
DE TRABAJO DE GRADO**

Nosotros autorizamos a la Universidad Politécnica Salesiana la publicación total o parcial de este trabajo de titulación y su reproducción sin fines de lucro.

Además declaramos que los conceptos, análisis desarrollados y las conclusiones del presente trabajo son de exclusiva responsabilidad de los los autores.

Johanna Elizabeth López Toapanta

CC: 1721053492

Cristian Fernando Pereira Vives

CC: 1720814654

DEDICATORIA

Dedico este proyecto a Dios por darme salud y vida para llegar al lugar en que me encuentro; a mis padres, José y Myriam, quienes han sido el pilar fundamental en mi vida y que gracias a su amor y confianza he logrado tomar las decisiones más acertadas; a mis hermanos Erika y Joel que han sabido brindarme la ayuda y el cariño necesario para enfrentar las diferentes adversidades que se han presentado; a mis abuelitos Carmen, Juan y Ángel ya que gracias a sus consejos he podido salir adelante; a toda mi familia por su apoyo moral incondicional pero en especial a mi tío Fernando (†) que siempre estuvo pendiente y alentándome durante su vida y que a pesar de no estar presente sé que se sentirá muy orgulloso de mi; a todos mis amigos, quien con su apoyo y confianza alentaron a la culminación de mis estudios y en especial a mi compañero del proyecto de titulación Cristian Pereira por contribuir a sacar adelante y culminar nuestra tan anhelada meta.

Johanna Elizabeth López Toapanta

Hoy que he culminado con este proyecto de tesis lo dedico a Dios por darme fortaleza y ser mi guía en los momentos más difíciles, por haber puesto en mi vida aquellas personas que han creído en mí y que han sido soporte a lo largo de estos años de formación académica.

A mis padres Rosa y Marcelo que con su amor incondicional y ejemplo han hecho de mí un hombre de bien, gracias a ustedes pues sus consejos y valores me han enseñado a ser constante y perseverante. A mis hermanos Karina y Alex por su amor e incondicional apoyo.

A mi novia Johanna por permanecer a mi lado dándome el coraje para seguir adelante, por ser comprensiva amorosa y paciente, a mi hija Abigail por ser la luz de mis ojos y mi fortaleza.

A mi compañera Johanna López con quien e transcurrido gran parte de mis estudios y ahora estoy culminando este proyecto de titulación.

Cristian Fernando Pereira Vives

AGRADECIMIENTO

Extendemos nuestro agradecimiento de manera especial al Ing. Rafael Jaya Duche por su ayuda y colaboración desinteresada para la realización de este proyecto de titulación.

A nuestros profesores quienes con nobleza y entusiasmo, depositaron en nosotros sus vastos conocimientos.

A nuestra prestigiosa Universidad por las enseñanzas recibidas.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1.....	2
PLANTEAMIENTO DEL PROBLEMA.....	2
1.1. Problema a resolver.....	2
1.2. Objetivos	3
1.2.1. Objetivo General	3
1.2.2. Objetivos Específicos.....	3
1.3. Justificación Del trabajo.....	4
1.4. Alcance.....	4
1.5. Metodología	5
1.5.1. Investigación bibliográfica.....	5
1.5.2. Desarrollo del software y hardware.	5
1.5.3. Implementación del sistema.....	6
1.5.4. Documentación del proyecto	6
CAPÍTULO 2.....	7
FUNDAMENTOS TEÓRICOS	7
2.1. Pico y Placa en el Ecuador.....	7
2.1.1. Aplicación del pico y placa en Quito.	7
2.1.2. Horarios, restricción por placas y salvoconductos.....	8
2.1.3. Límites del pico y placa en Quito.	9
2.1.4. Multas y sanciones.	10
2.1.5. Proceso de retención.	11
2.1.6. Proceso para la liberación de vehículos retenidos.....	11
2.1.7. Estacionamientos internos y patios de retención vehicular del pico y placa. 12	
2.2. Sistema de Posicionamiento Global (GPS).....	12
2.2.1. Definición de GPS.	13

2.2.2.	Estructura del GPS.	13
2.2.3.	Funcionamiento del GPS.	14
2.3.	Microcontroladores	15
2.3.1.	PIC´s.....	16
2.3.2.	AVR´s.	17
2.4.	Módulos o Tarjetas Electrónicas de Control.....	18
2.4.1.	Tarjetas de plataforma Arduino.	19
2.4.1.1.	Hardware para Arduino.....	19
2.5.	Comunicaciones Seriales	22
2.5.1.	Protocolo de comunicación RS-232.....	22
2.5.2.	Protocolo I ² C.....	24
2.5.2.1.	RTC (Reloj en tiempo real).....	24
2.6.	Comunicación USB.....	25
2.6.1.	Funcionamiento del USB	25
2.7.	Almacenamiento de Audio Digital	26
2.7.1.	Digitalización del sonido.....	27
2.7.2.	Amplificador de audio.....	28
2.8.	Lenguaje de Programación.....	28
2.8.1.	Lenguaje BASIC	29
2.8.2.	Lenguaje C	29
2.8.3.	Lenguaje Ensamblador (Assembler)	29
2.9.	Entorno de Desarrollo Integrado (IDE).....	29
	CAPÍTULO 3.....	32
	ANÁLISIS DE HARDWARE Y SOFTWARE	32
3.1.	Estudio de Factibilidad de Hardware	32
3.2.1.	Tarjetas de desarrollo	32
3.1.1.1.	ChipKIT MAX32.....	32
3.1.1.2.	Arduino MEGA 2560.....	33

3.1.1.3.	Jerarquización de la Factibilidad.....	34
3.1.2.	Módulos GPS	34
3.1.2.1.	Laipac Tech PG-11	34
3.1.2.2.	Arduino GPS Shield.....	35
3.1.2.3.	Jerarquización de la Factibilidad.....	36
3.1.3.	Amplificadores de Audio	36
3.1.3.1.	Amplificador estéreo TDA1554.....	36
3.1.3.2.	Amplificador TDA2003	37
3.1.3.3.	Jerarquización de la Factibilidad.....	38
3.2.	Descripción del hardware seleccionado	38
3.2.1.	Arduino Mega 2560	38
3.2.1.1.	Características Generales	39
3.2.1.2.	Microcontrolador ATmega2560.....	40
3.2.1.3.	Memorias	41
3.2.1.3.	Alimentación de la placa.....	41
3.2.1.4.	Pines de entrada y salida	42
3.2.1.5.	Pines de interrupciones externas	42
3.2.1.6.	Pines designados para conexión serial	43
3.2.1.7.	Pines con funciones varias	43
3.2.2.	Arduino GPS Shield.....	44
3.2.2.1.	Características principales del módulo.....	45
3.2.2.2.	Chipset SiRFstar III	45
3.2.2.3.	Sentencia NMEA	46
3.2.3.	Modulo Amplificador TDA2003	47
3.3.	Estudio de Factibilidad de Software	48
3.3.1.	Lenguajes de Programación de microcontroladores.	49
3.3.1.1.	Assembler.....	49
3.3.1.2.	Basic.....	50
3.3.1.3.	Lenguaje C	50
3.3.1.4.	Jerarquización de la Factibilidad.....	51
3.3.2.	Entornos de Desarrollo IDE.....	52

3.3.2.1. Labview	52
3.3.2.2. Matlab	53
3.3.2.3. Arduino	53
3.3.2.4. Jerarquización de la Factibilidad.....	54
3.4. Descripción del software seleccionado	55
3.4.1. Lenguaje de programación C	55
3.4.2. IDE Arduino.....	55
3.4.2.1. Estructura	59
CAPÍTULO 4.....	60
DESARROLLO DE HARDWARE Y SOFTWARE	60
4.1. Diseño de Hardware	60
4.1.1. Diagrama de Bloques	60
4.1.1.1. Etapa de adquisición y procesamiento de datos.	61
4.1.1.1.1. Fuente de alimentación.....	61
4.1.1.1.2. Arduino Mega 2560.....	61
4.1.1.2. Etapa de posicionamiento	63
4.1.1.2.1. Módulo GPS	63
4.1.1.2.2. Módulo RTC DS1307	63
4.1.1.2.3. Módulo WTV020SD	64
4.1.1.3. Etapa de visualización y alertas	65
4.1.1.3.1. Amplificador de audio.....	65
4.1.1.3.2. Pantalla LCD	65
4.1.1.4. Diagrama de la placa secundaria.....	67
4.1.1.5. Diagrama Esquemático de control del sistema de alerta.....	68
4.2. Diseño de Software	70
4.2.1. Diagrama de Flujo.....	70
CAPÍTULO 5.....	75
PRUEBAS Y RESULTADOS	75
5.1. Repetitividad del sistema	75
5.2. Linealidad del sistema.....	76
5.3. Pruebas del funcionamiento del sistema	82

CONCLUSIONES	94
REFERENCIAS	95
ANEXOS	98

ÍNDICE DE FIGURAS

Figura 1. Límites de restricción vehicular	10
Figura 2. Segmentos del GPS	14
Figura 3. Principio de triangulación y posicionamiento del GPS	15
Figura 4. Estructura general de un microcontrolador	16
Figura 5. Arquitectura Harvard	16
Figura 6. Tarjetas electrónicas de control	19
Figura 7. Estructura de la tarjeta Arduino	20
Figura 8. Comunicación Serial.....	22
Figura 9. Conector DB-9 y DB-25	23
Figura 10. Transmisión de bit utilizando el Protocolo I ² C	24
Figura 11. Nomenclatura de pines de los conectores USB	25
Figura 12. Digitalización del sonido	27
Figura 13. Etapas de amplificación del audio	28
Figura 14. Estructura del IDE	30
Figura 15. Partes de la placa Arduino Mega 2560	40
Figura 16. Estructura del Arduino GPS Shield	44
Figura 17. Amplificador TDA2003	47
Figura 18. Diseño del amplificador TDA2003	48
Figura 19. Entorno de desarrollo de Arduino	56
Figura 20. Menú Herramientas de Arduino.	57
Figura 21. Programa para encender y apagar un led.	57
Figura 22. Menú Herramientas de Arduino.	58
Figura 23. Diagrama de bloques del hardware	60
Figura 24. Fuente de alimentación	61

Figura 25. Conexiones Arduino Mega 2560.....	62
Figura 26. Esquema GPS.	63
Figura 27. Conexiones RTC DS1307.	64
Figura 28. Circuito WTV020SD.....	64
Figura 29. Circuito amplificador de audio.....	65
Figura 30. Conexiones pantalla LCD.....	66
Figura 31. Ruteado de pistas del circuito diseñado para la placa secundaria	67
Figura 32. Vista 3D de la placa electrónica secundaria	68
Figura 33. Diagrama de conexión del sistema de alerta para pico y placa	69
Figura 34. Implementación del sistema de alerta para pico y placa.	69
Figura 35. Diagrama de Flujo Principal.....	71
Figura 36. Diagrama de Flujo Principal.....	72
Figura 37. Diagrama de Flujo de subrutina de sonido.....	73
Figura 38. Diagrama de Flujo de subrutina del módulo GPS.....	74
Figura 39. Linealidad de los datos de latitud	81
Figura 40. Linealidad de los datos de longitud	81
Figura 41. Pruebas de alertas de proximidad tomando como referencia el Puente de Guajaló.....	83
Figura 42. Pruebas de alertas de proximidad tomando como referencia Chillogallo.....	84
Figura 43. Pruebas de alertas de proximidad tomando como referencia la Autopista General Rumiñahui	85
Figura 44. Pruebas de alertas de proximidad tomando como referencia el Puente Oswaldo Guayasamín.....	86
Figura 45. Pruebas de alertas de proximidad tomando como referencia la Av. de los Granados	87
Figura 46. Pruebas de alertas de proximidad tomando como referencia la entrada a Carcelén.....	88

Figura 47. Pruebas de alertas de proximidad tomando como referencia el ingreso a la Av. Mariscal Sucre.....	89
Figura 48. Pruebas de alertas de proximidad tomando como referencia la Av. Mitad del Mundo.	90
Figura 49. Pruebas de alertas de restricción tomando como referencia la Av. El Inca.....	92
Figura 50. Pruebas de alertas de restricción tomando como referencia la Av. Mariscal Sucre.....	93

ÍNDICE DE TABLAS

Tabla 1. Vehículos retenidos por el pico y placa desde el 2012	3
Tabla 2. Días de restricción según el número de placa	8
Tabla 3. Límites de restricción vehicular	9
Tabla 4. Multas por infringir la restricción vehicular Pico y Placa.....	11
Tabla 5. Patios de Retención vehicular para Pico y placa.....	12
Tabla 6. Gamas de microcontroladores PIC's.....	17
Tabla 7. Familias de microcontroladores AVR's.....	18
Tabla 8. Características de los Módulos Arduino	21
Tabla 9. Pines del Conector DB-9 y DB-25.....	23
Tabla 10. Matriz de evaluación de la tarjeta ChipKIT MAX32.....	33
Tabla 11. Matriz de evaluación de la tarjeta Arduino MEGA 2560.	33
Tabla 12. Cuadro comparativo de puntajes de tarjetas de desarrollo.....	34
Tabla 13. Matriz de evaluación del módulo Laipac Tech PG-11.....	35
Tabla 14. Matriz de evaluación del Arduino GPS Shield	35
Tabla 15. Cuadro comparativo de puntajes de los módulos GPS.	36
Tabla 16. Matriz de evaluación del Amplificador TDA1554	37
Tabla 17. Matriz de evaluación del Amplificador TDA2003	37
Tabla 18. Cuadro comparativo de puntajes de los amplificadores.....	38
Tabla 19. Características principales del Arduino	39
Tabla 20. Características de ATmega2560	40
Tabla 21. Pines de alimentación del Arduino Mega 2560	41
Tabla 22. Pines de Interrupciones Externas del Arduino Mega 2560.....	42
Tabla 23. Pines de conexiones seriales disponibles en Arduino Mega 2560.....	43
Tabla 24. Otros pines de la placa Arduino	43

Tabla 25. Características del Arduino GPS Shield	45
Tabla 26. Códigos NMEA principales	46
Tabla 27. Matriz de evaluación del lenguaje de programación Assembler	49
Tabla 28. Matriz de evaluación del lenguaje de programación Basic.....	50
Tabla 29. Matriz de evaluación del lenguaje de programación C.....	51
Tabla 30. Cuadro comparativo de puntajes del lenguaje de programación.	51
Tabla 31. Matriz de evaluación del software Labview	52
Tabla 32. Matriz de evaluación del software Matlab	53
Tabla 33. Matriz de evaluación del software Arduino	54
Tabla 34. Cuadro comparativo de puntajes de los IDE's de programación.	54
Tabla 35. Conexiones LCD.....	66
Tabla 36. Cálculo de repetitividad del sistema.	75
Tabla 37. Coordenadas obtenidas con el módulo GPS.	76
Tabla 38. Coordenadas obtenidas desde un celular.....	77

ÍNDICE DE ECUACIONES

Ecuación 1. Ecuación Lineal	77
Ecuación 2. Reemplazo de los valores de a	78
Ecuación 3. Reemplazo de los valores de b	78
Ecuación 4. Ecuación lineal obtenida con los datos receptados por el celular	79
Ecuación 5. Reemplazo de los valores de a	79
Ecuación 6. Reemplazo de los valores de b	80
Ecuación 7. Ecuación lineal obtenida con los datos receptados por el celular	80

ÍNDICE DE ANEXOS

Anexo 1. Detalle de las sentencias NMEA en función de la tabla 26	98
Anexo 2. Arduino GPS Shield	100
Anexo 3. Arduino Mega 2560	102
Anexo 4. ATmega 2560	103

RESUMEN

Este proyecto tiene como objetivo desarrollar un sistema de alerta para pico y placa, el cual se encargará de alertar mediante señalización por voz a los conductores de vehículos en Quito acerca de la restricción para circular dentro del perímetro vehicular donde se aplica esta medida, misma que fue adoptada a partir del año 2010 en esta ciudad con la finalidad de reducir la congestión en las horas pico (de 7h00 a 9h30 y de 16h00 a 19h30). En su diseño e implementación, se utilizará un módulo GPS para obtener la posición del vehículo (longitud y latitud), las coordenadas serán procesadas en el módulo Arduino y comparadas en función de los valores obtenidos anteriormente para delimitar el perímetro de restricción, este módulo será montado sobre la tarjeta de control Arduino Mega 2560, que se encargará de realizar el control principal de todo el sistema, también se incluirá un módulo RTC para obtener datos de hora y día, un módulo reproductor de audio que trabajará con una tarjeta Micro SD donde se encontrarán almacenadas las pistas, además fue necesario conectar el reproductor a un circuito amplificador para conseguir una mejor percepción del sonido por parte del usuario una vez emitidas las alertas.

En cuanto a los resultados obtenidos al realizar una comparación entre el módulo GPS que será utilizado para la implementación del dispositivo de alerta y el GPS de un celular fue factible verificar el porcentaje de error, el cual arrojó un 0.001 % en longitud y 0.018 % en latitud.

ABSTRACT

This project aims to develop warning system for peak and plate, which will alert drivers of Quito's vehicles through signaling of voice about restriction for circular within the perimeter where applied this measure, same as was adopted from the year 2010 in this city in order to reduce congestion during peak hours since 7.00 am to 9.30 am and since 04.00 p.m. to 07.30 p.m. In its design and implementation was used a GPS module to obtain the vehicle position (latitude and longitude), these coordinates will be processed on the Arduino software and will be compared to previously obtained values to delimit the perimeter restriction set by the Quito's Municipality, this module will be mounted on a Arduino Mega 2560 card, which will perform the main control of the entire alert system, also is included a Real Time Clock (RTC) module to obtain the information of day and time, also will use an audio module that will work with a Micro SD card where tracks are stored, it was also necessary to connect the player to amplifier circuit for better sound perception by the user, once issued alerts.

As for the results obtained from a comparison between the GPS module used to implement the alert's device and the cellphone's GPS was feasible verify that the error rate is 0.001% in length and 0.018% in latitude.

INTRODUCCIÓN

En los últimos años, el parque automotor ha evidenciado un incremento considerable en todas las regiones del Ecuador, donde la congestión vehicular dentro del Distrito Metropolitano de Quito se ha convertido en una de las principales problemáticas sociales, desencadenando grandes gastos en la movilidad, contaminación ambiental, molestias, enfermedades en la población, entre otros.

Sin embargo, con el fin de mejorar el tránsito vehicular las autoridades competentes han puesto énfasis en reducir el impacto que este problema acarrea, siendo una opción la construcción de nuevas carreteras y desvíos, la cual presenta ciertos impedimentos por las condiciones geográficas de la ciudad, mismas que limitan la expansión de la infraestructura vial, consecuentemente para contribuir con este problema decidieron implementar una medida que regula el tránsito, conocida como pico y placa.

Esta restricción vehicular originó grandes molestias en la ciudadanía a causa de las multas y sanciones a las cuales se encuentran expuestos los infractores de dicha medida, no obstante, siendo muchos los motivos que llevan a los capitalinos a violentar el pico y placa, la facilidad de olvido de las personas es la causa principal, para la cual se encontró pertinente diseñar un dispositivo de alerta para los conductores de vehículos mediante un sistema de señalización de voz y referenciación con módulo GPS, el cual se encargará de tomar las coordenadas que permitirán conocer cierta información (hora, día, lugar, posición del vehículo) que con ayuda del método de cuadrangulación se podrá delimitar el perímetro de restricción donde se aplica el pico y placa, enviando mediante un reproductor mp3 alertas de voz que permitan recordar al usuario el día en que su vehículo no puede circular dentro del perímetro de restricción, además en caso de que él ignore esta alerta y se esté aproximando a dicho perímetro, el dispositivo le advertirá nuevamente al riesgo que se expone si sobrepasa el límite donde se aplica esta medida, de tal manera que se pueda evitar multas y trámites que implica el no acatar la medida de restricción.

CAPÍTULO 1

PLANTEAMIENTO DEL PROBLEMA

En el presente capítulo, se da a conocer el problema a resolver con la realización del proyecto, así como también sus objetivos, justificación, antecedentes, alcances y metodología de investigación.

1.1. Problema a resolver

Este proyecto surge como una iniciativa de contribuir con los conductores de vehículos para evitar las molestas multas y sanciones que han sido impuestas a los quiteños a causa de infringir una medida de restricción vehicular, misma que fue adoptada a partir del año 2010 en la ciudad de Quito y a la cual se denomina “Pico y placa”, cuya finalidad es disminuir la congestión vehicular en horas pico siendo estas de 7h00 a 9h30 y de 16h00 a 19h30.

Estas multas y sanciones van desde la retención del vehículo hasta una multa equivalente a la tercera parte de una remuneración básica unificada, en caso de reincidencia por segunda ocasión, la multa equivale a la mitad de una remuneración básica unificada y por tercera ocasión o más, con una multa equivalente a una remuneración básica unificada.

A pesar de las fuertes multas y sanciones a las cuales deberán someterse los infractores, algunos conductores de vehículos particulares infringen esta norma, ya sea por olvido o imprudencia, comprometiéndose de tal modo a cumplir con las sanciones impuestas. A continuación en la tabla 1, se presenta una estadística de vehículos retenidos por violentar el pico y placa desde el año 2012.

Tabla 1.

Vehículos retenidos por el pico y placa desde el 2012

Año	N° Vehículos Retenidos
2012	44580
2013	48132
2014	34308
TOTAL	127020

Nota. (Quito P. M., s.f.)

Elaborado por: Johanna López y Cristian Pereira

Por los inconvenientes mencionados anteriormente se ha planteado la idea de diseñar un sistema de alerta con voz y referenciación con módulo GPS, mediante el cual los usuarios se mantendrán informados acerca del día en el que tienen prohibido la circulación con su vehículo dentro de los límites establecidos e impuesto por la medida de restricción vehicular pico y placa, además este contribuirá a que el conductor evite ingresar al perímetro restringido por dicha medida.

1.2. Objetivos

1.2.1. Objetivo General

Diseñar e implementar un sistema de alerta con voz y referenciación con módulo GPS para alertar a los conductores de vehículos el día, hora y lugar de pico y placa en la ciudad de Quito.

1.2.2. Objetivos Específicos

- Realizar el levantamiento de información geográfica y cartográfica de la ciudad de Quito.
- Analizar los componentes en software y hardware con los que se implementará el prototipo electrónico.
- Diseñar y desarrollar el sistema electrónico de alerta de pico y placa.
- Implementar el dispositivo en un automóvil y realizar las pruebas de funcionamiento correspondientes para verificar los estándares de calidad.

1.3. Justificación Del trabajo

En la actualidad, debido a la aplicación de la medida de restricción vehicular pico y placa en la ciudad de Quito y principalmente por la facilidad de olvido de las personas, se ha propuesto diseñar un sistema de alertas que permita dar aviso al usuario si presenta prohibición para circular dentro del perímetro donde se aplica la restricción vehicular, contribuyendo con este dispositivo a que el conductor evite las multas y los incómodos trámites que conlleva infringir esta medida.

1.4. Alcance

El sistema de alerta con voz y referenciación con módulo GPS será capaz de realizar lo siguiente:

- Una vez que el vehículo sea encendido, el GPS se encargará de tomar coordenadas para encontrar la posición del vehículo.
- Un LCD acoplado al dispositivo mostrará la fecha y hora actual, además se podrá configurar el número de placa del vehículo mediante pulsadores.
- El sistema de alerta se encontrará en la capacidad de comparar el último dígito de la placa con el día, hora y lugar de restricción del “pico y placa”.
- En el día que se presente restricción del número de la placa del vehículo, el sistema emitirá la siguiente alerta de voz: “Recuerde, hoy su vehículo tiene pico y placa”, en caso de que el usuario ignore esta alerta, el sistema deberá advertir al conductor que se está aproximando al perímetro de restricción con una nueva alerta “Atención, está acercándose al perímetro de pico y placa”.
- Si el día de restricción corresponde al de la placa del vehículo, y se encuentre dentro del perímetro de circulación limitado por las leyes del pico y placa, se escuchará la siguiente alerta: “Recuerde, hoy su vehículo tiene pico y placa y está dentro del perímetro”.

El sistema no realizará lo siguiente:

- No funciona para rastreo del vehículo.
- No indica la velocidad a la que el vehículo viaja.

1.5. Metodología

En este proyecto se utilizaron los siguientes métodos de investigación de acuerdo al desarrollo del proyecto: Investigación bibliográfica; Desarrollo del software y hardware; Implementación del sistema, y finalmente la documentación del proyecto.

1.5.1. Investigación bibliográfica

Se realizó investigaciones acerca de contenidos asociados a Sistema de Posicionamiento Global, tarjetas de adquisición de datos y tarjetas de control, módulos de reproducción, módulos de amplificación de audio, software de programación Arduino.

Se recolecto información acerca de la tarjeta electrónica de control a usarse por las ventajas que presenta en función de la capacidad de memoria, número de entradas analógicas/digitales, puertos disponibles, fácil adquisición de datos, también se estudió el módulo GPS que será utilizado para la adquisición de datos dado a su compatibilidad con la tarjeta de control.

1.5.2. Desarrollo del software y hardware.

Se implementó un circuito de control que permitirá la adquisición de datos (longitud, latitud, hora y fecha), los cuales serán utilizados dentro de la programación realizada en el software Arduino y que permitirá delimitar el perímetro de restricción utilizando el método de cuadrangulación, además se utilizó una pantalla LCD encargada de mostrar los campos adquiridos por el GPS.

Para el funcionamiento de las alertas se realizó un circuito de almacenamiento y amplificación de sonido, además fue necesario la utilización de un módulo RTC que permitirá mediante la programación diferenciar los días de restricción.

1.5.3. Implementación del sistema

Se incorporó una tarjeta secundaria utilizada para la comunicación entre los dispositivos (LCD, pulsadores, tarjeta Micro SD, LED's) y la tarjeta principal de control Arduino Mega 2560, que permitirá mantener el orden y un correcto funcionamiento del sistema de alertas.

1.5.4. Documentación del proyecto

Se desarrolló un documento, el cual describe el proceso del desarrollo e implementación del proyecto: diseño y construcción del sistema de alertas para pico y placa mediante la programación realizada en el software Arduino, el cual deberá cumplir el siguiente orden: planteamiento del problema, fundamentos teóricos, análisis de hardware y software, desarrollo del hardware y software y finalmente un análisis de resultados.

CAPÍTULO 2

FUNDAMENTOS TEÓRICOS

En este capítulo, se presentará breves antecedentes de la medida de restricción vehicular pico y placa aplicado en la ciudad de Quito y los fundamentos teóricos en los que se basa el sistema de alerta con voz y referenciación con módulo GPS.

2.1. Pico y Placa en el Ecuador

El congestionamiento vial es uno de los principales problemas que se presenta en las capitales alrededor del mundo, y que durante el transcurso de los años se ha ido incrementado de una manera casi incontrolable, causando malestares a la ciudadanía y al mismo tiempo impulsando a encontrar una solución que ayude a mitigar en algo el tráfico dentro de las grandes ciudades, para lo cual se creó una medida conocida con el nombre de “Pico y Placa”, la cual tuvo sus inicios en la ciudad de Bogotá durante la alcaldía de Enrique Peñalosa, en el año de 1998, posteriormente ha sido implementado haciendo uso del mismo nombre en otras ciudades como: Colombia, Venezuela y Ecuador.

Este último, al igual que los demás países encontró en esta medida de restricción vehicular una solución para el problema de movilidad, pretendiendo al implementar dicha medida dentro Quito la capital del Ecuador agilitar la fluidez del tráfico, obtener una circulación segura y mayor comodidad para la ciudadanía.

2.1.1. Aplicación del pico y placa en Quito.

Según la Resolución N° 0017 que contiene el reglamento de la Ordenanza Metropolitana N° 0305 emitida por el Consejo Metropolitano de Quito a cargo del Sr. Augusto Barrera Guarderas (Alcalde del Distrito Metropolitano de Quito), el 5 de Marzo del 2010 se creó la modalidad de restricción a la circulación vehicular en la ciudad de Quito correspondiente a la denominada Pico y Placa, “la cual se basa en la prohibición de la circulación de un grupo de vehículos, durante un día laborable de la semana y en los períodos de hora pico de la circulación vehicular.” (Barrera Guarderas, 2010, pág. 4)

Esta medida fue implementada para motivar a los ciudadanos a adoptar cambios en la movilidad, promover el consumo de energía; lo cual contribuirá con la reducción de gastos por concepto de subsidios de combustibles fósiles, disminuir la contaminación ambiental, y principalmente para controlar el tráfico vehicular en las horas picos de tal manera que se pueda aprovechar la capacidad vial existente y acortar los tiempos de viaje.

2.1.2. Horarios, restricción por placas y salvoconductos.

El tránsito vehicular se encuentra restringido para un cierto grupo de vehículos seleccionados de acuerdo al último dígito de su placa, los cuales tendrán prohibido su circulación un día laborable a la semana y en las horas consideradas pico dentro de la ciudad. (Barrera Guarderas, 2010, pág. 4)

La medida de restricción para pico y placa se aplicará de lunes a viernes en los siguientes horarios: en la mañana de 7h00 a 9h30 y en la tarde y noche de 16h00 a 19h30. A continuación, en la Tabla 2 se indica el día correspondiente a cada uno de los números de placa.

Tabla 2.

Días de restricción según el número de placa

Día de la semana	Último dígito de la placa del vehículo
Lunes	1 y 2
Martes	3 y 4
Miércoles	5 y 6
Jueves	7 y 8
Viernes	9 y 0
Sábados, domingos y feriados	Ninguno

Nota. (Libro I del Código Municipal para el Distrito Metropolitano de Quito, 2010, pág. 4)

Dentro de esta restricción se encuentran excluidos del cumplimiento de la misma los siguientes: (Barrera Guarderas, 2010, pág. 5)

- Vehículos Oficiales del Presidente y Vicepresidente de la República.
- Vehículos del cuerpo diplomático y consular, es decir aquellos que tengan las placas de color azul con letras CD, CC, AT y OI.
- Vehículos que aporten con tareas de asistencia social y emergencias pertenecientes al Estado y ambulancias privadas.
- Vehículos de transporte que cuenten con salvoconducto, dentro de estos se encuentran los vehículos particulares que son conducidos por personas con discapacidad, con movilidad reducida; y vehículos particulares que trasladan personas con discapacidad.
- Vehículos conducidos por personas de la tercera edad.
- Vehículos de transporte público, transporte de carga liviana, escolar, institucional, turístico y taxis debidamente autorizados por el Municipio.

2.1.3. Límites del pico y placa en Quito.

Según la Sección II de la Resolución N° 0017, el área de restricción se aplicará dentro de la zona urbana de la ciudad de Quito, comprendida entre los límites viales de referencia descritos a continuación en la tabla 3 e ilustrados en la figura 1. (Barrera Guarderas, 2010, pág. 7)

Tabla 3.

Límites de restricción vehicular

Límites	Con restricción	Sin restricción
Occidente	Av. Mariscal Sucre, entre Av. Morán Valverde y Av. Manuel Córdova Galarza.	Av. Mariscal Sucre entre Av. Manuel Córdova Galarza y Av. Diego de Vásquez de Cepeda.
Norte		Av. Diego de Vásquez de Cepeda, entre Av. Galo Plaza y Av. Manuel Córdova Galarza, y Panamericana Norte, entre la Av. Simón Bolívar y Av. Galo Plaza.
Oriente		Av. Simón Bolívar
Sur		Av. Morán Valverde, entre Av. Simón Bolívar y Av. Mariscal Sucre.

Nota.

Elaborado por: Johanna López y Cristian Pereira

Límites de restricción vehicular

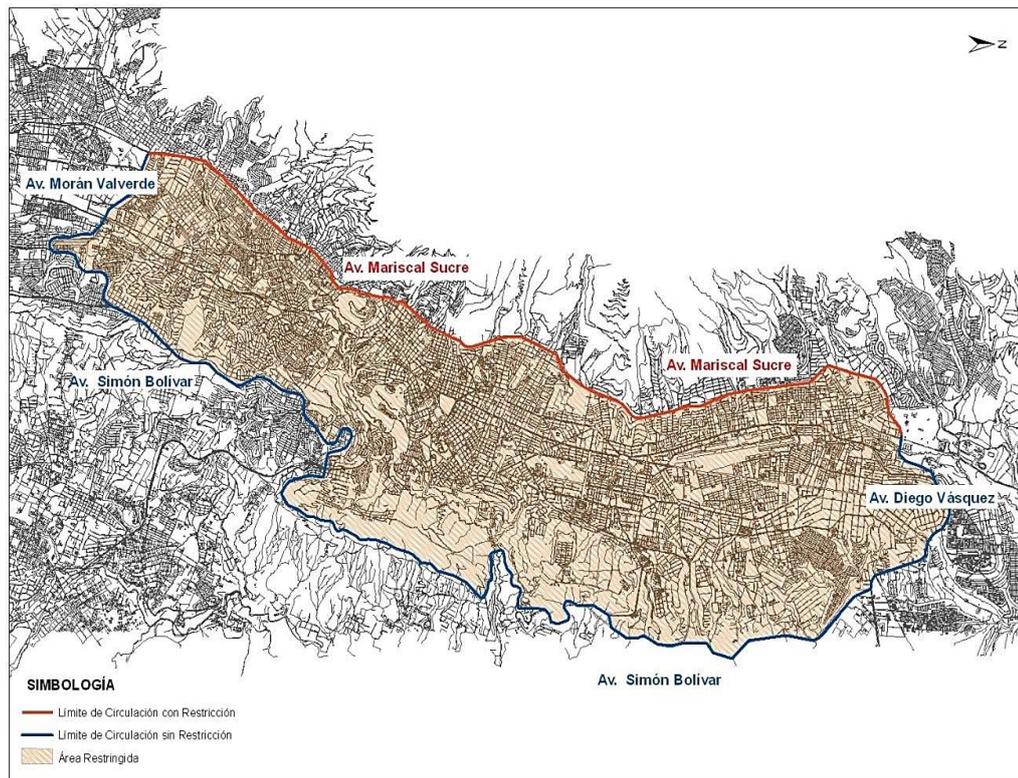


Figura 1. Libro I del Código Municipal para el Distrito Metropolitano de Quito, 2010, pág. 8

2.1.4. Multas y sanciones.

Los propietarios de los vehículos que infrinjan la restricción de esta medida, deberán acatar ciertas sanciones que serán impuestas por el órgano competente del Distrito Metropolitano, la comisión de la infracción administrativa, las cuales se observan en la tabla 4, donde se presenta los valores de los 2 últimos años, ya que dichas multas se calculan en función de la remuneración básica unificada (RBU) en el país. Sin embargo a partir del mes de Junio, con el nuevo Alcalde Mauricio Rodas, estas multas fueron reducidas al 50 % de los rubros especificados más adelante, además de eliminar la retención del vehículo, y únicamente dependerá del pago para retirar el mismo inmediatamente.

Tabla 4.
Multas por infringir la restricción vehicular Pico y Placa

Infracción	Retención de vehículos	% de RBU	Multa 2013	Multa 2014	Multa Junio 2014
Primera vez	1 día	30	106,00	113,33	56.67
Segunda vez	3 días	50	159,00	170,00	79.50
Tercera vez en adelante	5 días	100	318,00	340	170

Nota. (ecuador-vial.com)

Elaborado por: Johanna López y Cristian Pereira

2.1.5. Proceso de retención.

Una vez detectado al vehículo infractor circulando en horarios y días no permitidos, la brigada de control conformada por la Policía Nacional, Metropolitana y fiscalizadores de la EPMMOP inician el proceso de formalización de la infracción cometida al vulnerar la medida de restricción pico y placa, quienes procederán a informar verbalmente al conductor sobre la infracción, recoger evidencia (física y fotográfica), verificar reincidencia de la infracción en la base de datos u otra vía, registrar, suscribir y entregar al conductor el formulario correspondiente, proceder a la retención del vehículo y conducirlo al patio de retención vehicular (mientras se realice el pago de la multa), y finalmente reportar al Municipio para su registro y emisión de la respectiva orden de pago. (Barrera Guarderas, 2010, pág. 11)

2.1.6. Proceso para la liberación de vehículos retenidos.

Para la liberación de vehículos retenidos es importante presentar los siguientes requisitos en el patio de retención vehicular:

- Los originales y copias del pago de la multa, dichos valores pueden ser cancelados en los puntos de pago como son: Agencias de Servipagos y Produbanco.
- La cédula de identidad de la persona que retira el automotor.
- La matrícula del vehículo.
- Copia del parte de Ingreso del automotor ha dicho patio. (Sandoval, 2014)

2.1.7. Estacionamientos internos y patios de retención vehicular del pico y placa.

El Municipio del Distrito Metropolitano de Quito ha destinado sitios exclusivamente para la detención de los vehículos infractores conocidos como patios de retención vehicular, ubicados en diferentes lugares de la ciudad como se indica en la tabla 5.

No obstante, a causa de la gran demanda de vehículos también se cuenta con estacionamientos de demanda interna (Iñaquito y Cumandá) para los conductores que se encuentran en el hipercentro durante las horas pico.

Además, existen estacionamientos ubicados en la periferia del hipercentro, conocidos como estacionamientos de borde, los cuales están destinados para atender la demanda de los vehículos que no pueden ingresar a causa de la restricción vehicular Pico y Placa entre los cuales se tiene los parqueaderos del Condado, Carapungo, Monte Olivo y Cuscungo.

Tabla 5.

Patios de Retención vehicular para Pico y placa

#	Patio de Retención	Dirección
1	La Río Coca	Av. Río Coca y Los Laureles
2	La Y	Av. 10 de Agosto, junto al Terminal Trolebús La Y
3	Gualaquiza	Calle Gualaquiza y Luis Tufiño
4	6 de Diciembre	Av. 6 de Diciembre y Av. Galo Plaza
5	Cumandá	Antiguo Terminal Terrestre
6	Rumichaca	Av. Rumichaca y Matilde Álvarez
7	Guamaní	Av. Maldonado, frente al Beaterio

Nota. (juanpch, 2013)

Elaborado por: Johanna López y Cristian Pereira

2.2. Sistema de Posicionamiento Global (GPS)

En la actualidad, los grandes avances tecnológicos contribuyen a que las personas conozcan la localización de cualquier punto alrededor del mundo, brindando varios beneficios a sus usuarios, sin embargo, han tenido que transcurrir varios años.

Según, Fernando Cuartero, profesor de Informática en la Universidad de Castilla, menciona en uno de sus artículos que: “El lanzamiento del satélite espacial estadounidense Vanguard en 1959 puso de manifiesto que la transmisión de señales de radio desde el espacio podría servir para orientarnos y situarnos en la superficie terrestre” (Cuartero, 2012), siendo esta una pieza fundamental para que el Departamento de Defensa de los Estados Unidos de América, pusiera en funcionamiento el sistema de localización por satélite GPS, el mismo que, según David Zurdo tuvo sus inicios “como un proyecto que pretendía superar tecnológicamente el sistema militar de satélites Transit, el cual fue creado por la Marina Estadounidense y que empleaban el efecto Doppler para establecer posiciones” (Zurdo, 2012), fue entonces cuando el Departamento de Defensa Estadounidense ordenó el desarrollo de un sistema unificado, dando origen al proyecto Navstar y con este el servicio de GPS.

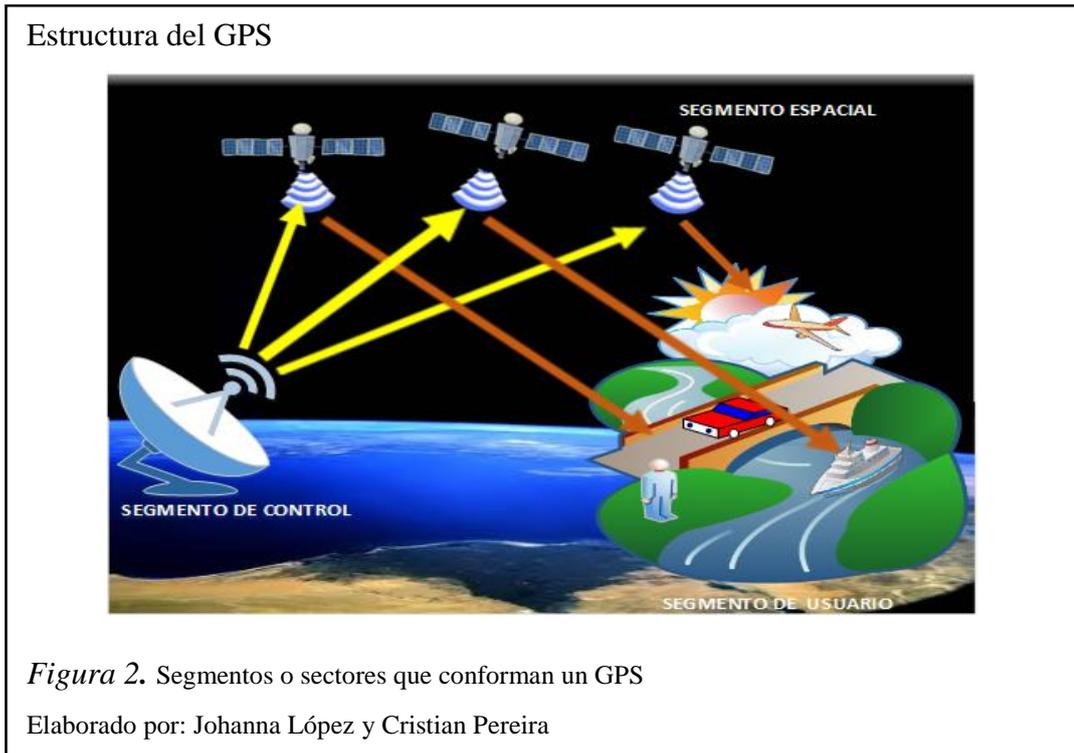
2.2.1. Definición de GPS.

“El GPS (Global Positioning System o Sistema de Posicionamiento Global) es un sistema de navegación por satélite que permite determinar en todo el mundo la posición de un objeto, una persona, un vehículo o una nave, bajo cualquier condición meteorológica y con una precisión hasta de centímetros.” (Cerdeja Sánchez & Pazmiño Días , 2011, pág. 6)

2.2.2. Estructura del GPS.

El sistema GPS tiene una estructura claramente definida, la misma que se encuentra dividida en tres sectores o segmentos, cada uno de ellos tiene un ámbito de acción, y realizan funciones distintas.

- **Segmento espacial:** Este segmento se compone por los satélites activos que forman el sistema, tanto de navegación como de comunicación, así como las diferentes señales que envían y reciben cada uno de los receptores.
- **Segmento de control:** Se encuentra formado por el conjunto de estaciones terrenas que recogen los datos de los satélites y monitoriza el sistema GPS.
- **Segmento de usuario:** Se encuentra conformado por los equipos receptores que reciben las señales del segmento espacial.

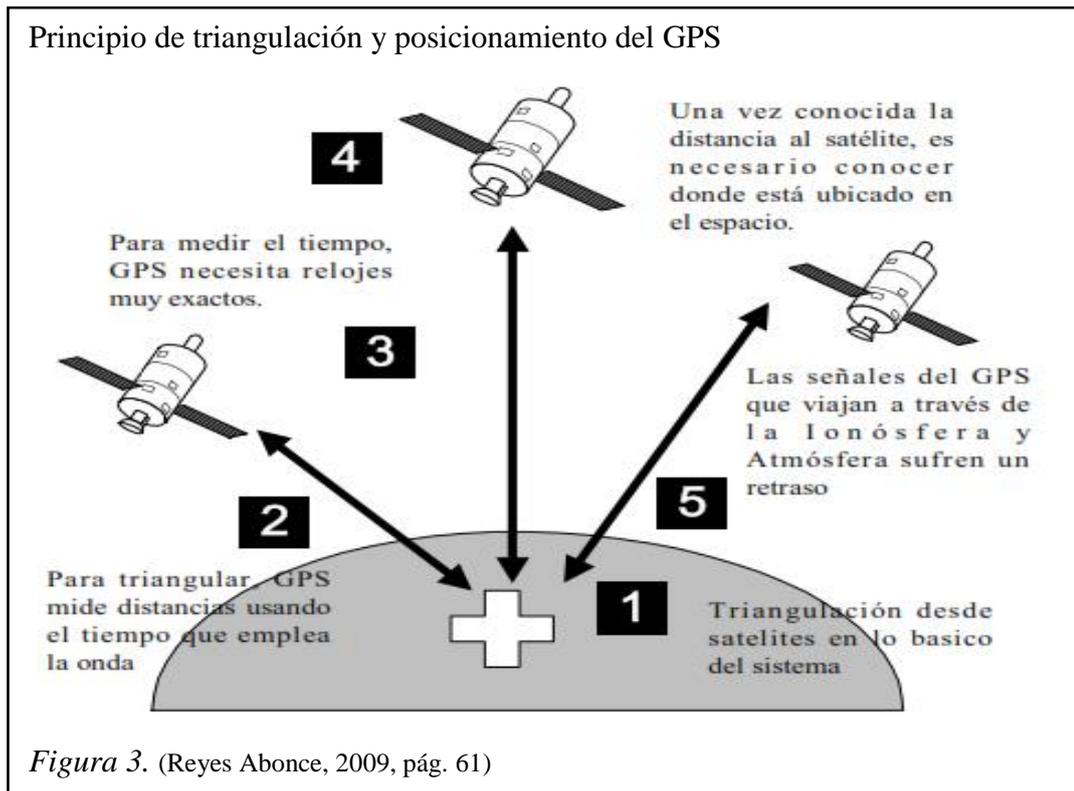


2.2.3. Funcionamiento del GPS.

El servicio GPS funciona mediante una red de 24 satélites en órbita que componen la constelación Navstar (Navigation Satellite Timing and Ranging o Satélite de Navegación Cronometría y Distanciometría), los cuales consiguen cubrir el cielo extendiendo su uso alrededor del mundo en todo momento, es decir este sistema que antes era exclusivo para uso militar, en la actualidad contribuye en ramas como: la geodesia, la topografía, la cartografía, la navegación, otras.

Para que este sistema encuentre un punto o coordenada necesita encontrar al menos tres satélites que proporcionarán la localización satelital mediante la triangulación de la posición de los satélites captados, los cuales proporcionaran datos de Longitud, Latitud y Altitud.

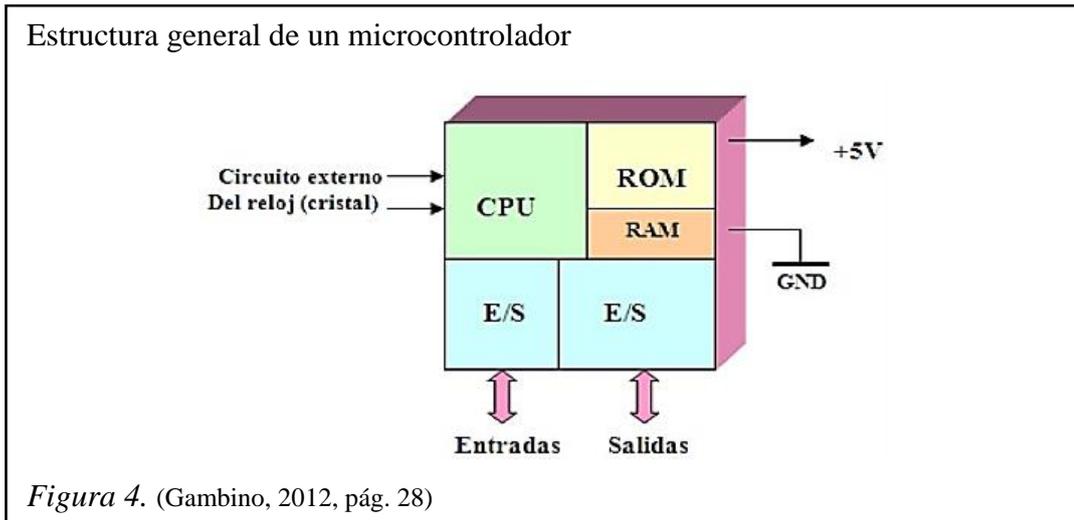
“La triangulación se basa en determinar la distancia de cada satélite respecto al punto de medición” (Cerde Sánchez & Pazmiño Días , 2011, pág. 6), y una vez que se ha obtenido las distancias de al menos tres de los satélites, es más fácil determinar la posición absoluta o coordenadas reales de un punto, como se observa en la figura 3.



2.3. Microcontroladores

“Un microcontrolador es un dispositivo electrónico encapsulado en un circuito de alto nivel de integración” (Clavijo Mendoza, 2011, pág. 17), en efecto, su arquitectura se asemeja a la de una computadora, ya que cuenta con un procesador, un bloque de memoria para almacenamiento de datos y uno para almacenamiento del programa, puertos de entrada y salida (Figura 4), permitiendo crear aplicaciones a pequeña escala, haciendo uso de teclados, displays, sensores, leds, entre otros.

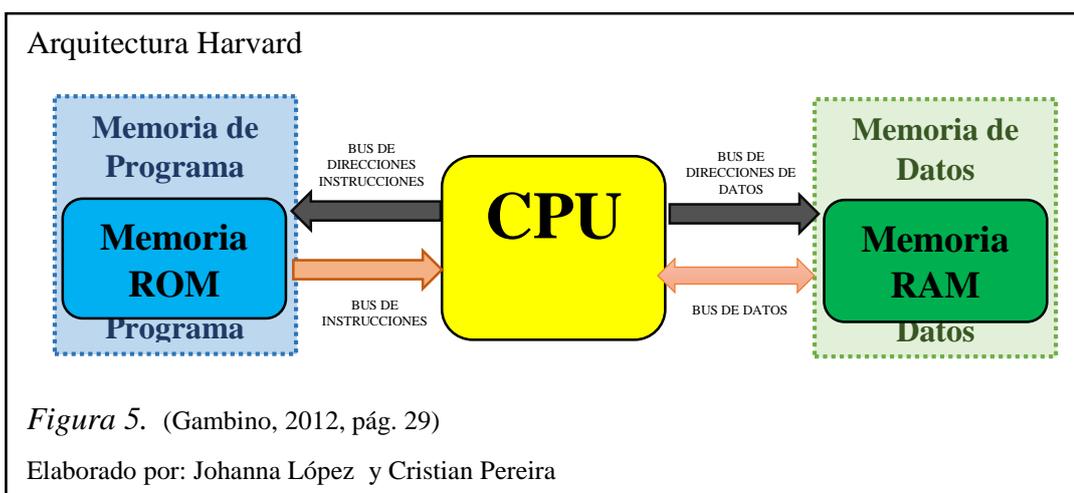
Existen una variedad de microcontroladores en el mercado, sin embargo, en esta ocasión se destacará la familia PIC y la familia AVR, ya que su popularidad es alta, dependiendo de las necesidades y requerimientos de sus consumidores, ya sea por su costo, nivel de integración, rendimiento, arquitectura o su lenguaje de programación.



2.3.1. PIC's.

Los PIC's son una familia de microcontroladores fabricados por la empresa estadounidense MICROCHIP y basados en la arquitectura Harvard, es decir cuenta con dos memorias interconectadas a través de buses diferentes al CPU permitiendo que los instrucciones y los datos se almacenen en espacios de memoria separados para mejorar su rendimiento, como se indica en la Figura 5. (Flores Cortez, 2009, pág. 2)

La memoria de datos es de 8 bits y la memoria de programa destinada para almacenar únicamente las instrucciones del programa es de 12, 14, 16 y 32 bits, según el modelo.



Los microcontroladores PIC's se clasifica en 4 gamas o grupos, aunque todas las versiones están construidas bajo una misma arquitectura, cada uno de ellos se ajustan a las necesidades, de acuerdo al tamaño de los datos que manejan las instrucciones, el

tipo y capacidad de las memorias, el número de líneas de E/S y las funciones auxiliares que presentan, como se observa en la tabla 6.

Tabla 6.

Gamas de microcontroladores PIC's

GAMA	MODELO	PINES	ROM palabras	RAM bytes	INSTRUCCIONES bits
Enana	PIC12Fxx	8	< 1k	< 128	12
Baja	PIC12Fxx/ PIC16Fxx	8 a 28	< 1k	< 256	12
Media	PIC16Fxx	8 a 68	< 8k	< 368	14
Alta	PIC17Fxx	18 a 68	< 16k	< 1536	16
Mejorada	PIC18Fxx	18 a 80	< 24k	< 3968	16

Nota. (Cano Infantes, 2012, pág. 5)

Elaborado por: Johanna López y Cristian Pereira

2.3.2. AVR's.

El AVR es una familia de microcontroladores que cuentan con 32 registros de 8 bits, fabricados por la compañía noruega ATMEL y al igual que los PIC's utilizan arquitectura Harvard, cuentan con un CPU RISC (Computadores de Juego de Instrucciones Reducido), una memoria FLASH o ROM para almacenamiento del programa, memoria de datos estática SRAM, memorias internas EEPROM y un puerto serial (ISP) formado por 3 pines para la programación de la memoria, que se conectan a un programador y este a un puerto del computador para quemar el código dentro del microcontrolador. (Flores Cortez, 2009, pág. 2)

Estos microcontroladores permiten la ejecución de instrucciones mediante la metodología 'pipeline', la misma que cuenta con dos etapas (cargar y ejecutar), es decir, mientras una instrucción está siendo ejecutada, la siguiente es pre-capturada de la memoria de programa.

La familia de microcontroladores AVR es bastante extensa y todas comparten el mismo núcleo AVR, pero tienen distintos periféricos y cantidades de memoria RAM y ROM, como se observa en la tabla 7.

Tabla 7.

Familias de microcontroladores AVR's

MODELO	PINES	MEMORIA FLASH	INSTRUCCIONES bits
tinyAVR	8 a 32	1 – 16 KB	8
MegaAVR	28 a 100	4 – 256 KB	8
AVR XMEGA	44 a 100	16 – 384 KB	8 / 16
UC3 AVR32	48 a 144	16 – 256 KB	32
AP7 AVR32	196 a 256	UP 32 KB	32

Nota.

Elaborado por: Johanna López y Cristian Pereira

2.4. Módulos o Tarjetas Electrónicas de Control

Son plataformas de hardware libre diseñadas con el propósito de facilitar la implementación de prototipos electrónicos, y son utilizadas especialmente por diseñadores, artistas o aficionados en la creación de entornos interactivos.

Estas pequeñas placas cuentan con un microcontrolador y son controladas por un programa diseñado en cualquier tipo de lenguaje de programación, además que su software funciona con licencia abierta y funcionan sin necesidad de conectarse a una computadora.

En el mercado existen tarjetas realizadas por fabricantes conocidos que ofrecen calidad y comodidad a sus usuarios, sin embargo estas pueden ser diseñadas por personas entendidas en la materia sin ningún problema. Existen varios tipos de tarjetas electrónicas, todo dependerá del proyecto o de las necesidades del mismo, “es importante tener en cuenta factores importantes como el tamaño, sus prestaciones, su programación e interacción con otros dispositivos.” (Oña Caisatoa & Viteri Tigse, 2013, pág. 10)

Tarjetas electrónicas de control



Figura 6. (<http://www.element14.com/>)

2.4.1. Tarjetas de plataforma Arduino.

Los módulos o tarjetas Arduino son plataformas de desarrollo de código abierto, basados en hardware sencillo y fácil de manipular, además posee software propietario con un entorno de desarrollo amigable para sus usuarios.

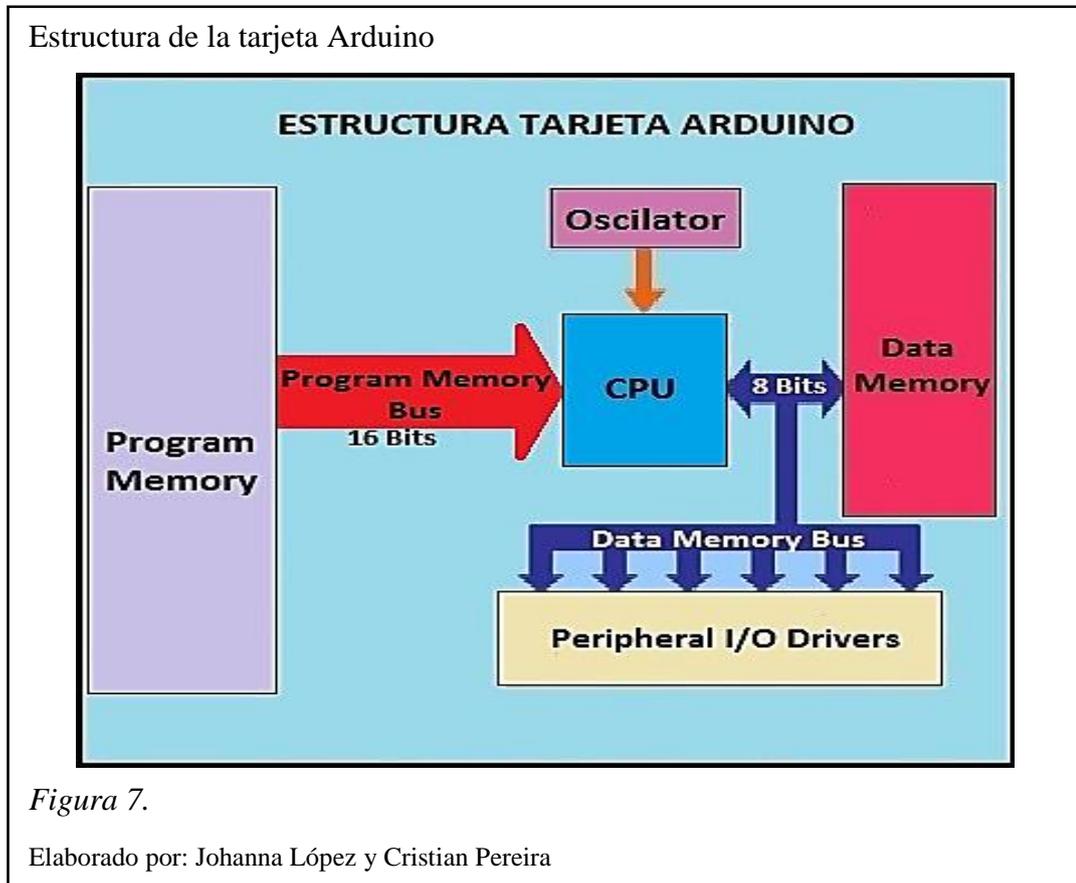
2.4.1.1. Hardware para Arduino

Los diseños y esquemas de Arduino cuentan con una plataforma open-hardware (libre distribución y utilización), la cual se basa en una sencilla placa con entradas y salidas (E/S), analógicas y digitales. Su elemento principal es un microcontrolador de la familia Atmegaxx8, el cual es un chip sencillo y de bajo costo que permite el desarrollo de múltiples diseños.

Este tipo de tarjetas parten de una misma base, por tanto todas presentan iguales características en su entorno de desarrollo, mismas que serán detalladas más adelante y que se observan en la figura 7.

- Microcontrolador Atmegaxx8
- Entradas/salidas digitales, dentro de las cuales algunas proporcionan salidas PWM.
- Entradas analógicas
- Pines de alimentación
- Memorias (flash, SRAM, EEPROM)

- Puerto serial que permite realizar comunicación con periférico
- Puerto serial una conexión USB
- Pulsador para resetear en caso de existir fallas en los procesos, etc.



Además, la marca Arduino cuenta con una gran variedad de placas con parecidas características, pero presentan diferencias que pueden ser determinantes a la hora de elegir una u otra, como es: el tamaño, el número de entradas, el número de salidas, el voltaje de alimentación, la necesidad de programadores externos, el nivel de conocimientos requerido, etc. (<http://www.reflexiona.biz/>, 2010)

A continuación, en la tabla 8 se presenta las especificaciones de cada uno de los módulos encontrados en el mercado (Arduino Mini, Arduino Nano, Arduino Uno, Arduino Mega y Arduino Mega ADK).

Tabla 8.

Características de los Módulos Arduino

Características	Mini	Nano	Uno	Mega	Mega ADK
Microcontrolador	ATmega168	ATmega168 o ATmega328	ATmega328	ATmega1280	ATmega2560
Voltaje de funcionamiento	5V	5 V	5 V	5 V	5 V
Voltaje de entrada	7-9 V	7-12 V	7-12 V	7-12 V	7-12 V
Pines E/S digital	14 (6 son salidas PWM)	14 (6 son salidas PWM)	14 (6 son salidas PWM)	54 (14 son salidas PWM)	54 (15 son salidas PWM)
Pines entrada analógica	8	8	6	16	16
DC Corriente continua por pin E/S	40 mA	40 mA	40 mA	40 mA	40 mA
Memoria Flash	16 KB	ATmega168 16 KB o ATmega328 32 KB	32 KB	128 KB	256 KB
SRAM	1 KB	ATmega168 1 KB o ATmega328 2 KB	2 KB	8 KB	8 KB
EEPROM	512 bytes	ATmega168 512 bytes o ATmega328 1 KB	1 KB	4 KB	4 KB
Velocidad de reloj	16 MHz	16 MHz	16 MHz	16 MHz	16 MHz

Nota.

Elaborado por: Johanna López y Cristian Pereira

2.5. Comunicaciones Seriales

Es uno de los protocolos más comunes que permiten la comunicación entre dispositivos, en el cual se envía un bit de información a la vez y a un ritmo acordado entre el emisor y el receptor, permitiendo alcanzar mayores distancias.

La mayor parte de microcontroladores, entre ellos Arduino, poseen un puerto de comunicación serial, por tanto para comunicarse con las PC's que poseen únicamente puerto USB es necesario un conversor USB-Serial, para ello Arduino utiliza el protocolo RS-232, el cual le permitirá recibir y enviar datos a un computador de manera serial como se muestra en la figura 8.



2.5.1. Protocolo de comunicación RS-232

El protocolo RS-232, conocido por sus siglas en inglés como Recommended Standard 23 es una interfaz que permite el intercambio de datos binarios entre un DTE (Equipo terminal de datos) y un DCE (Equipo de Comunicación de datos).

El protocolo RS-232 soporta comunicaciones a velocidades bajas con distancias cortas, y permite la transmisión de datos de forma síncrona y asíncrona. El RS-232 consiste en un conector tipo DB-25 o DB-9, siendo este último el más usado, mismo

que utiliza solo 3 pines para transmisión y los restantes para control, petición, envío, otros, como se observa en la tabla 9 y en la figura 9.

Tabla 9.

Pines del Conector DB-9 y DB-25

Señal		Función	Pines DB-9	Pines DB-25
Detectada portadora	CD	El modem indica a la computadora que el canal de comunicación está ocupado.	1	8
Recepción de datos	RxD	Recibir datos	2	3
Transmisión de datos	TxD	Transmitir datos	3	2
Terminal de datos preparado	DTR	La computadora y el RS232 están listos	4	20
Señal de tierra	GND	Tierra o voltaje.	5	1
Equipo de comunicación preparado	DSR	El modem está listo.	6	6
Petición de transmisión	RTS	La computadora indica al modem que quiere transmitir a la máquina remota	7	4
Canal libre para la transmisión	CTS	El modem indica a la computadora que puede transmitir.	8	5
Indicador de llamada	RI	Se activa cuando detecta una señal (llamada)	9	22

Nota.

Elaborado por: Johanna López y Cristian Pereira

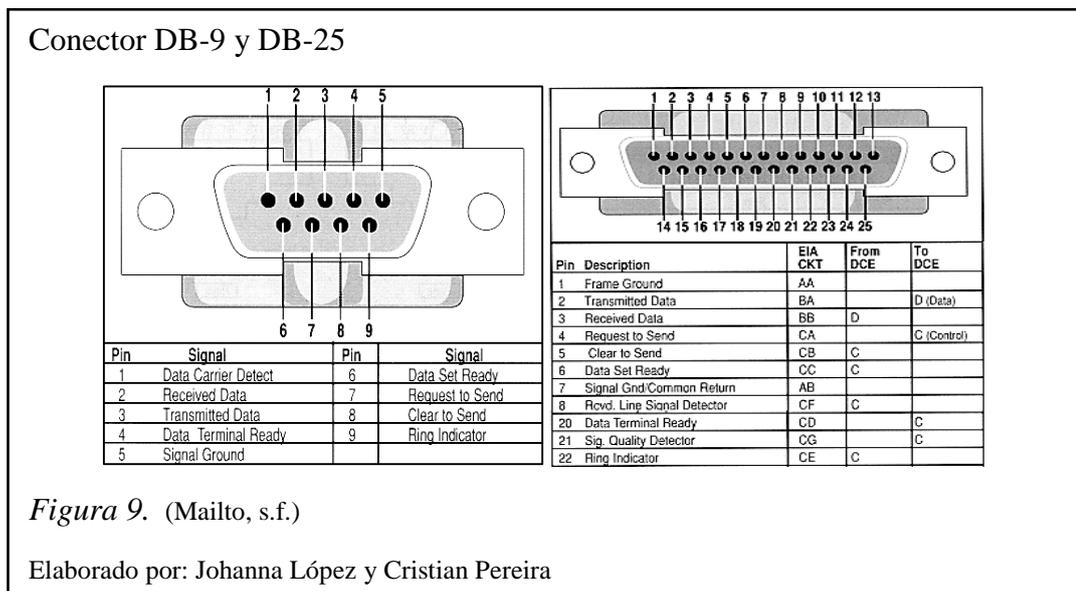


Figura 9. (Mailto, s.f.)

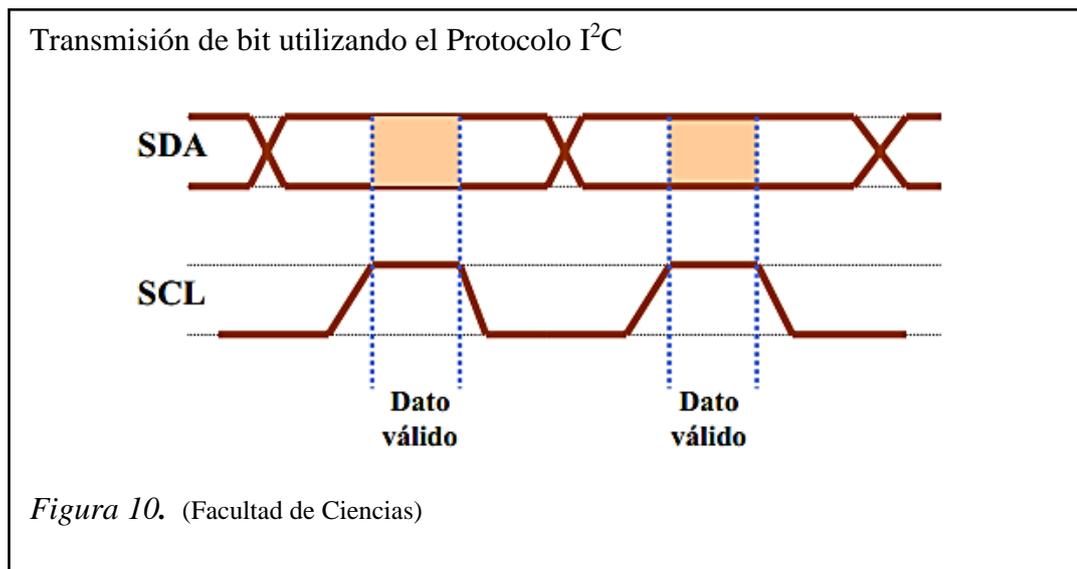
Elaborado por: Johanna López y Cristian Pereira

2.5.2. Protocolo I²C

El protocolo I²C permite la comunicación o transmisión de datos de forma bidireccional entre dispositivos o sistemas diferentes controlados por un microcontrolador.

La comunicación I²C es controlada por una señal de reloj común y está formado por dos líneas de transmisión, conocidas como SDA y SCL.

- **SDA (Serial Data Line):** Es la línea por donde se transfieren los datos, y por cada bit de información se requiere un pulso SCL.
- **SCL (Serial Clock line):** Es la línea de reloj por donde viaja una señal que permitirá la sincronización en la transferencia de los datos SDA, y los datos solo pueden cambiar cuando SCL está a nivel bajo.



2.5.2.1. RTC (Reloj en tiempo real)

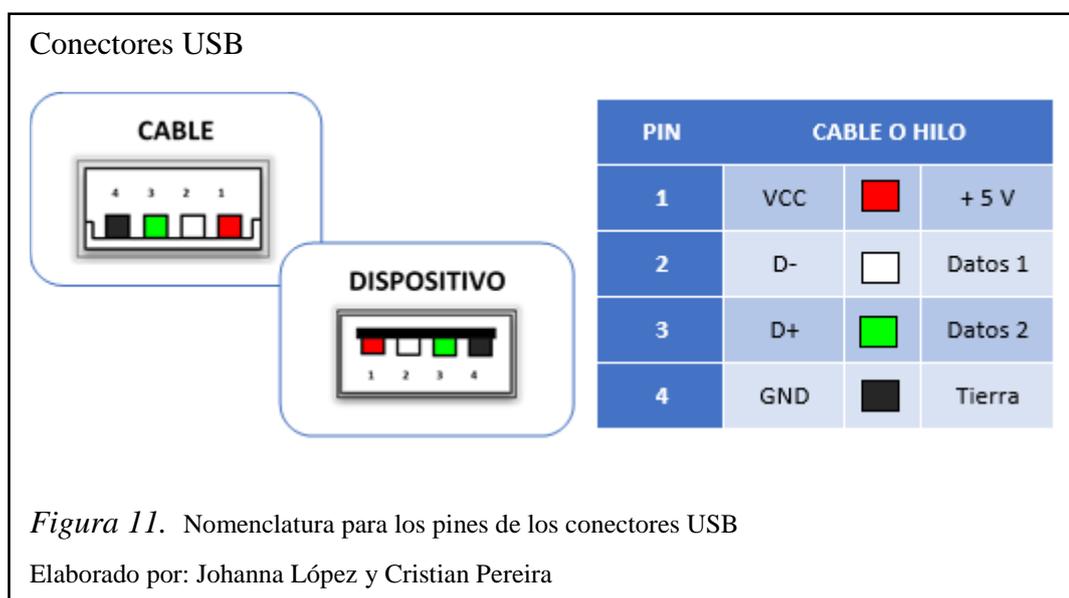
El RTC es un circuito integrado o módulo digital utilizado para almacenar y proporcionar información sobre tiempo y fecha actual, incluso si se encuentra apagado, este se comunica con los microcontroladores a través de la comunicación serial I²C. (MikroElektronika, 1998)

2.6. Comunicación USB

USB por sus siglas en inglés Universal Serial Bus, es una interfaz de entrada/ salida basada en arquitectura tipo serial creada para transmitir datos de forma más rápida, reemplazando a los puertos seriales estándar y fue desarrollado por “empresas líderes del sector de las telecomunicaciones y de los ordenadores como: Compaq, DEC, IBM, Intel, Microsoft, NEC y Northern Telecom.” (Universidad de Sevilla, 2011)

2.6.1. Funcionamiento del USB

El USB funciona sin necesidad de instalación de hardware adicional, ya que detecta configuraciones automáticamente, permitiendo que los dispositivos se encadenen mediante una topología bus o estrella, además consta de 4 hilos (conexión a tierra, alimentación del bus, 2 hilos de datos), soporta “plug and play” y proporcionan fuente de alimentación (5V – 15V) a los dispositivos con los que se conecta, como se observa en la figura 11.



A partir del año 1995, el estándar USB ha desarrollado una amplia gama de versiones:

- **USB 1.0:** ofrece dos modos de comunicación (modo de alta velocidad 12Mbps y velocidad baja 1,5Mbps)
- **USB 1.1:** fue la primer versión popular de USB y no cambia los rasgos de velocidad
- **USB 2.0:** su principal mejora es el incremento en la tasa de transferencia a alta velocidad (480Mbps).
- **USB 3.0:** utiliza una tasa de transferencia de 5Gbps.

2.7. Almacenamiento de Audio Digital

“El audio digital es la representación de señales sonoras mediante un conjunto de datos binarios” (López Martín, pág. 4), es decir una secuencia de unos y ceros, obtenidos al realizar el muestreo de una señal analógica, por ende se dice que la tecnología digital, emplea tecnología analógica para captar los sonidos que serán grabados de forma digital para después reproducirlos y escucharlos, sin embargo esto que parece tan sencillo, en realidad tiene que pasar por un gran proceso, el cual se detalla a continuación:

- Para captar el sonido se necesita micrófonos o fuentes auxiliares, el audio es analógico antes de realizar la grabación digital.
- Para proceder a digitalizar el sonido es necesario transformar los valores de tensión o voltaje obtenidos en valores numéricos binarios, que permitirán crear archivos de sonido digital.
- Los unos y ceros obtenidos después de realizar la conversión analógica a digital no son audibles ni visibles, por tanto es necesario realizar la conversión de digital a analógica para obtener el sonido similar al captado inicialmente.
- Una vez obtenida la señal analógica, después de realizar las respectivas conversiones se amplifica dicha señal para que pueda ser audible.

2.7.1. Digitalización del sonido

Es un proceso de conversión de señales analógicas a digitales, para la cual, se toman muestras de dicha señal cada cierto tiempo para ser medida su intensidad, y almacenarlas en una secuencia de dígitos binarios por cada muestra, como se observa en la figura 12.a.

Una vez realizado el muestreo de la señal, se procede a cuantificarla, de tal manera que se asigna valores analógicos discretos a las muestras, limitando los valores de amplitud de una señal y definiendo una serie no continua de valores posibles.

Por último se realiza el proceso de codificación, la cual consiste en asignar un código binario o conjunto de bits a cada uno de los valores posibles de las muestras de la señal.

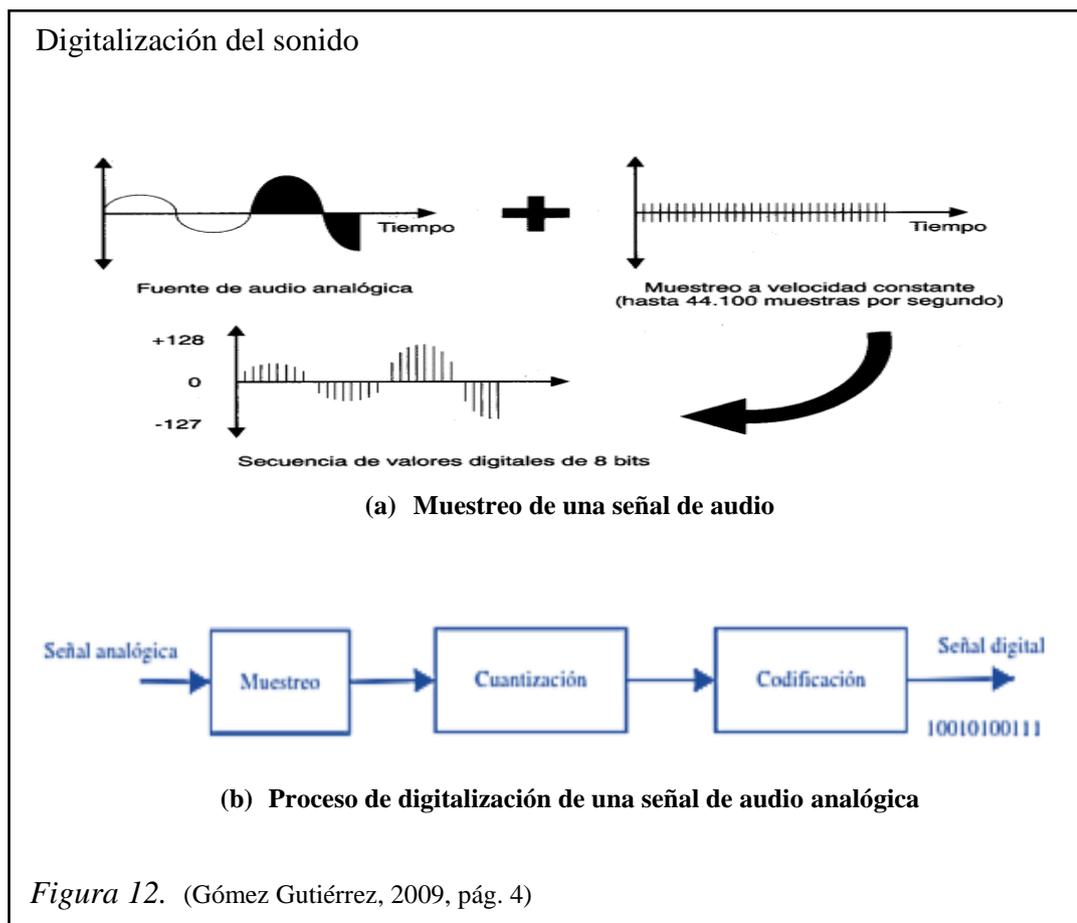
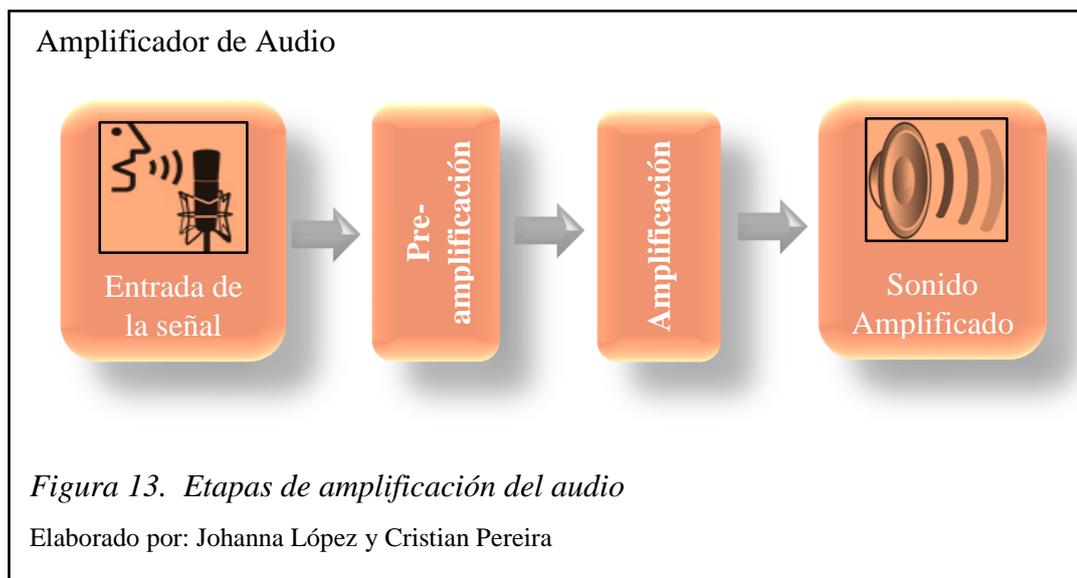


Figura 12. (Gómez Gutiérrez, 2009, pág. 4)

2.7.2. Amplificador de audio

El proceso de amplificación de audio consiste en aumentar el voltaje y corriente de una señal eléctrica anteriormente obtenida, sin alterar o deformar dicha señal, esto implica aumentar la amplitud de dicha señal, y a la vez hace que aumenten su sonoridad y su volumen, este proceso de amplificación consta de ciertas etapas, mismas que serán detalladas a continuación:

- **Preamplificación:** esta es la primera etapa y se encarga de acondicionar la señal para entregarle un mínimo de potencia, la cual no es muy evidente.
- **Amplificación:** esta etapa es la encargada de captar la señal entregada después de la preamplificación y amplificarla hasta obtener suficiente potencia en la fuente externa (parlantes), convirtiéndola en energía eléctrica .



2.8. Lenguaje de Programación

El lenguaje de programación consiste en un conjunto de órdenes o comandos que describen un proceso, cada uno de estos lenguajes tienen sus propias instrucciones y expresiones que se combinan para controlar el comportamiento físico y lógico de una máquina.

2.8.1. Lenguaje BASIC

El BASIC es un lenguaje de programación de alto nivel que consta de instrucciones, comandos y funciones que realizan conjuntamente una tarea determinada, creado con fines pedagógicos, es considerado como uno de los lenguajes más fáciles de aprender ya que provee pocas estructuras para facilitar al programador la construcción de programas, sin embargo, las restricciones que se presentan como consecuencia de la carencia estructuras de programación lo sitúa en una gran desventaja respecto a los otros.

2.8.2. Lenguaje C

“El C es un lenguaje moderno de propósito general que combina las características de un lenguaje de alto nivel (programación estructurada, tipos y estructura de datos, recursividad, etc.) con una serie de características más propias de lenguajes de más bajo nivel” (Instituto Tecnológico Superior de Calkiní, 2014) que posee pocas instrucciones, es un lenguaje muy eficiente especialmente en el desarrollo de software de sistemas y aplicaciones.

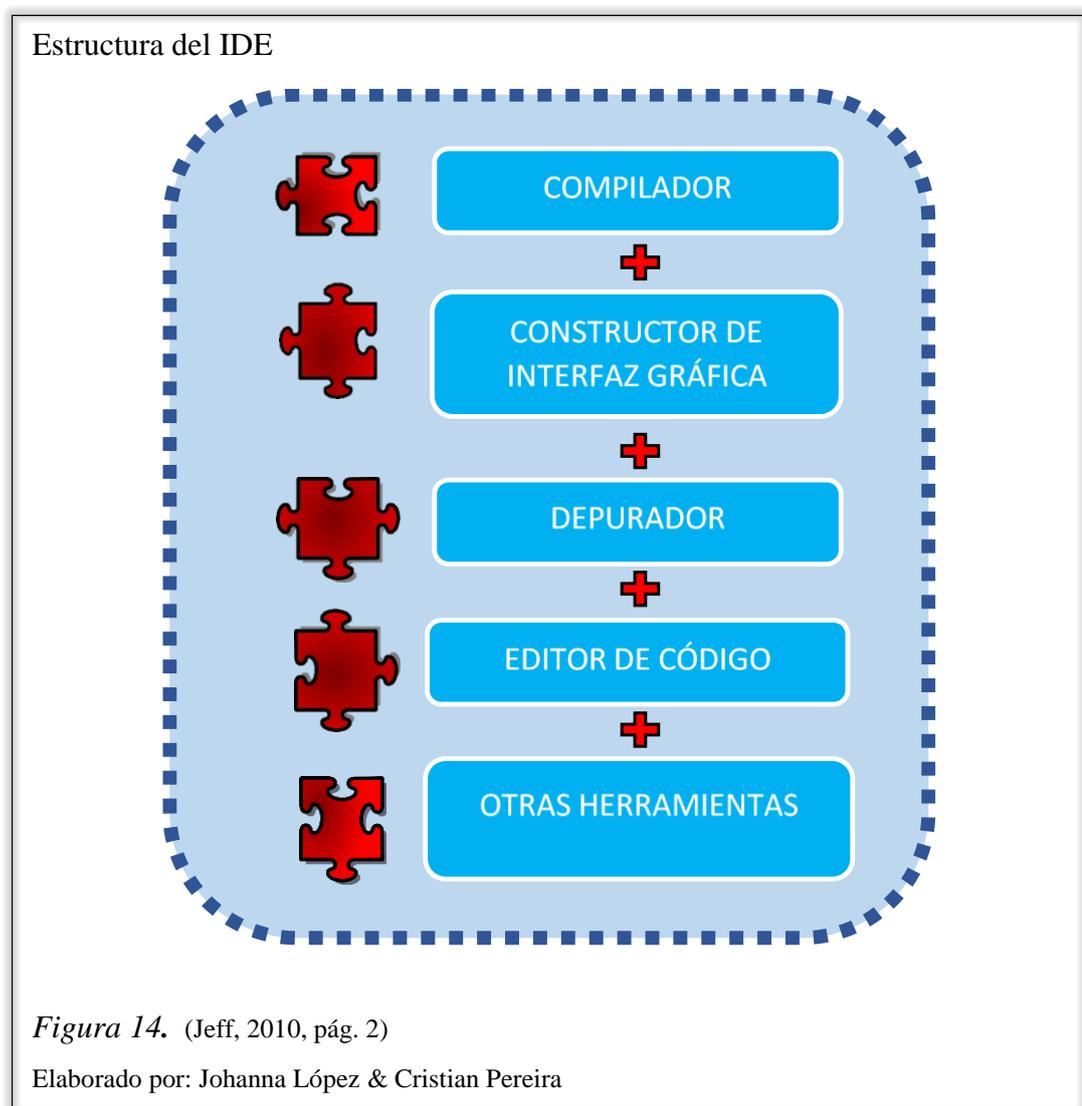
2.8.3. Lenguaje Ensamblador (Assembler)

El lenguaje ensamblador fue creado en sus inicios para reemplazar el código de máquina, convirtiéndose en un lenguaje de programación de bajo nivel que constituye la representación más directa del código máquina comprensible por los programadores, anteriormente muy utilizado, en la actualidad muy pocas personas lo requieren.

2.9. Entorno de Desarrollo Integrado (IDE)

Un IDE es un entorno de programación que consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica o visual, que se encarga de dialogar internamente con cada uno de ellos, misma que proveen un marco de trabajo amigable que permite trabajar cómodamente, en donde:

- **El editor de código:** Sirve específicamente para editar el código fuente de las aplicaciones informáticas.
- **El compilador:** Funciona como un traductor de código fuente, es decir traduce a un lenguaje que sea entendible para las máquinas.
- **El depurador:** Es una aplicación que tiene como función probar y eliminar posibles errores en un programa en desarrollo.
- **El constructor de interfaz gráfica:** Es una herramienta utilizada para crear y diseñar las interfaces con las cuales interactúa entre la aplicación y el usuario.



Existen una gran variedad de IDE's, sin embargo para este trabajo se hablará de algunos de ellos, que son comúnmente utilizados:

- **Microsoft Visual Studio:** Es un IDE que utiliza software licenciado, por tanto, es uno de los más completos, cuenta con autocompletado inteligente y corrección automática de errores, sin embargo se puede conseguir una versión Express.
- **NetBeans:** Es un IDE multiplataforma escrito en Java, pero que sin embargo permite editar, compilar, ejecutar y programar en varios lenguajes de programación, no requiere licenciamiento y es totalmente accesible para su uso.
- **Eclipse:** Es un IDE multilenguaje y multiplataforma basados en software libre, que a pesar de ser escrito completamente en Java permite desarrollar aplicaciones en C, C++, PHP, JavaScript y Perl, entre otros.
- **Arduino:** Es un IDE de desarrollo de software de código abierto, basado en uno de los lenguajes de programación más simple (C/C++), constituido por un entorno gráfico que permite programar directamente al hardware de manera fácil, además presenta una comunidad de desarrollo que permite acceso a referencias y ejemplos de proyectos.

CAPÍTULO 3

ANÁLISIS DE HARDWARE Y SOFTWARE

En este capítulo se realiza un análisis comparativo el cual ayudará en la selección adecuada de los materiales, al igual que el software adecuado que se utilizará para la construcción y programación del dispositivo de alerta propuesto en este proyecto de tesis.

3.1. Estudio de Factibilidad de Hardware

El funcionamiento correcto de un dispositivo depende tanto de hardware como software, tomando esta consideración en este literal se realizará un análisis minucioso de los distintos elementos existentes en el mercado para tomar la decisión acerca de cuáles se emplearán en la creación del prototipo, más adelante se realizará un análisis similar para determinar el software adecuado, permitiendo obtener un complemento perfecto entre estos dos, mismo que se verá reflejado en el desempeño del dispositivo.

3.2.1. Tarjetas de desarrollo

Existen un sin número de tarjetas de desarrollo, sin embargo para realizar el análisis de factibilidad, el cual permitirá elegir la mejor opción para la implementación de este proyecto se tomó en consideración dos modelos en especial, ya que son los más comercializados en el mercado.

3.1.1.1. ChipKIT MAX32

Esta tarjeta de desarrollo fue lanzada por la compañía Digilent y Microchip y utiliza un microcontrolador de 32 bits PIC32, posee 83 pines de entrada/salida digitales, 16 pines de entradas analógicas, 5 salidas PWM y puertos de comunicaciones (serial y USB).

Al realizar el análisis de factibilidad para la utilización de esta placa en el diseño del prototipo, se evidenció los siguientes resultados presentados en la tabla 10.

Tabla 10.

Matriz de evaluación de la tarjeta ChipKIT MAX32.

Solución	Puntaje 100%	Método de Evaluación
ChipKIT MAX32	70	Costos
	50	Proveedores
	80	Características Técnicas
	100	Herramienta de Desarrollo
	70	Documentación

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.1.1.2.Arduino MEGA 2560

Esta placa es basada en el microcontrolador ATmeg1280, cuenta con 54 entradas/salidas digitales, 14 salidas PWM, 16 entradas analógicas, puertos serie y conexión USB.

En la tabla 11, se muestra los resultados obtenidos al realizar el análisis de factibilidad de la tarjeta Arduino MEGA 2560.

Tabla 11.

Matriz de evaluación de la tarjeta Arduino MEGA 2560.

Solución	Puntaje 100%	Método de Evaluación
Arduino MEGA 2560	80	Costos
	100	Proveedores
	90	Características Técnicas
	100	Herramienta de Desarrollo
	100	Documentación

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.1.1.3. Jerarquización de la Factibilidad

Los criterios tomados en cuenta para jerarquizar la factibilidad de cada solución son los siguientes: Costos, proveedores, características técnicas, herramientas de desarrollo y documentación disponible.

Tabla 12.

Cuadro comparativo de puntajes de tarjetas de desarrollo.

Parámetros	ChipKIT MAX32			Arduino MEGA 2560		
	Puntaje	Peso	Total	Puntaje	Peso	Total
Costos	7	0,3	2,1	8	0,3	2,4
Proveedores	5	0,1	0,5	10	0,1	1
Características Técnicas	8	0,4	3,2	9	0,4	3,6
Herramientas de Desarrollo	10	0,1	1	10	0,1	1
Documentación	7	0,1	0,7	10	0,1	1
Puntaje Total	7,5	9,00

Nota.

Elaborado por: Johanna López y Cristian Pereira

Tomando en cuenta los valores obtenidos en la tabla 12, se logró identificar que la tarjeta adecuada para la realización del proyecto es la Arduino Mega 2560, la cual se describirá detalladamente más adelante.

3.1.2. Módulos GPS

3.1.2.1. Laipac Tech PG-11

Este componente se caracteriza por su alto desempeño gracias a la arquitectura GPS SiRF Star II y su chipset de bajo consumo (menor a 250 mW). Soporta el protocolo estándar NMEA0183, característico de las comunicaciones marítimas.

Se realiza un análisis de factibilidad para determinar si este módulo es el indicado para utilizar en el proyecto y se obtuvo los siguientes resultados evidenciados en la tabla 13.

Tabla 13.

Matriz de evaluación del módulo Laipac Tech PG-11.

Solución	Puntaje 100%	Método de Evaluación
Laipac Tech PG-11	70	Costos
	70	Proveedores
	80	Características Técnicas
	60	Herramienta de Desarrollo
	80	Documentación

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.1.2.2.Arduino GPS Shield

Es un módulo GPS que posee interfaz SD, permitiendo almacenar datos de posición u otro tipo de información en una tarjeta microSD de alta capacidad, una de sus principales ventajas se encuentra en la compatibilidad que posee con las placas de desarrollo de Arduino.

En la tabla 14, se realiza el análisis de factibilidad del módulo Arduino GPS Shield y se indica los resultados obtenidos.

Tabla 14.

Matriz de evaluación del Arduino GPS Shield

Solución	Puntaje 100%	Método de Evaluación
Arduino GPS Shield	80	Costos
	80	Proveedores
	100	Características Técnicas
	80	Herramienta de Desarrollo
	90	Documentación

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.1.2.3. Jerarquización de la Factibilidad

Los criterios tomados en cuenta para jerarquizar la factibilidad de cada solución son los mismos que han sido mencionados en el literal 3.1.1.3.

Tabla 15.

Cuadro comparativo de puntajes de los módulos GPS.

Parámetros	Laipac Tech PG-11			Arduino GPS Shield		
	Puntaje	Peso	Total	Puntaje	Peso	Total
Costos	7	0,3	2,1	8	0,3	2,4
Proveedores	5	0,1	0,5	8	0,1	0,8
Características Técnicas	8	0,4	3,2	10	0,4	4
Herramientas de Desarrollo	6	0,1	0,6	8	0,1	0,8
Documentación	8	0,1	0,8	9	0,1	0,9
Puntaje Total	7,2	8,9

Nota.

Elaborado por: Johanna López y Cristian Pereira

De acuerdo, a los valores obtenidos en la tabla 15, el módulo que se utilizará para la implementación del proyecto es el Arduino GPS Shield, el cual se describirá detalladamente más adelante.

3.1.3. Amplificadores de Audio

3.1.3.1. Amplificador estéreo TDA1554

Este amplificador es un circuito integrado basado en el amplificador TDA1554, de donde se deriva su nombre, a pesar de requerir una cantidad muy pequeña de elementos para su funcionamiento, este puede entregar 44 watts de potencia sonora.

Se realiza el análisis de factibilidad del amplificador TDA1554 y se indica los resultados obtenidos, los cuales serán expuestos en la tabla 16.

Tabla 16.

Matriz de evaluación del Amplificador TDA1554

Solución	Puntaje 100%	Método de Evaluación
Amplificador TDA1554	80	Costos
	70	Proveedores
	70	Características Técnicas
	60	Herramienta de Desarrollo
	70	Documentación

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.1.3.2. Amplificador TDA2003

El amplificador de sonido TDA2003 utiliza el circuito integrado TDA2003, el cual posee la misma configuración y distribución del amplificador TDA2002, una de sus ventajas es que usa muy pocos componentes externos y es muy sencillo de construir, además de mantener un bajo costo.

Se realiza el análisis de factibilidad del amplificador TDA2003, para lo cual se indica los resultados obtenidos en la tabla 17.

Tabla 17.

Matriz de evaluación del Amplificador TDA2003

Solución	Puntaje 100%	Método de Evaluación
Amplificador TDA2003	100	Costos
	80	Proveedores
	80	Características Técnicas
	80	Herramienta de Desarrollo
	80	Documentación

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.1.3.3. Jerarquización de la Factibilidad

Los criterios tomados en cuenta para jerarquizar la factibilidad de cada uno de los amplificadores son los descritos en la tabla 18.

Tabla 18.

Cuadro comparativo de puntajes de los amplificadores.

Parámetros	Amplificador estéreo TDA1554			Amplificador TDA2003		
	Puntaje	Peso	Total	Puntaje	Peso	Total
Costos	8	0,3	2,1	10	0,3	3
Proveedores	7	0,1	0,7	8	0,1	0,8
Características Técnicas	7	0,4	2,8	8	0,4	3,2
Herramientas de Desarrollo	6	0,1	0,6	8	0,1	0,8
Documentación	7	0,1	0,7	8	0,1	0,8
Puntaje Total	7,1	8,6

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.2. Descripción del hardware seleccionado

3.2.1. Arduino Mega 2560

Arduino Mega es una de las placas más grandes existentes en el mercado, ya que cuenta con una gran cantidad de pines digitales y analógicos, los cuales brindan la posibilidad de generar señales PWM. Sin embargo, a pesar del número de pines su tamaño es poco considerable (10,16 cm x 5,33 cm) y puede ser usada proyectos que necesitan optimizar el espacio.

Además, cuenta con varios pines de alimentación (5V y GND) permitiendo conexiones simultáneas sin necesidad de placas adicionales y puede ser alimentada desde una fuente externa o simplemente a través de una computadora, cuenta con la opción reset, misma que puede ser ejecutada mediante el código de programación o simplemente haciendo uso del botón para resetear que posee la placa.

3.2.1.1. Características Generales

Arduino Mega 2560 es una de las placas de desarrollo más completas en el mercado, permitiendo realizar en ella una variedad de diseños, en la tabla 19 se describe brevemente las características principales por las cuales se decidió utilizar esta placa para la implementación de este proyecto de titulación.

Tabla 19.

Características principales del Arduino

Arduino Mega 2560	
Microcontrolador	ATmega2560
Voltaje de funcionamiento	5V
Voltaje de entrada (recomendado)	7 - 12V
Voltaje de entrada (límite)	6 - 20V
Pines E/S digitales	54 (15 proporcionan salida PWM)
Pines de entrada analógica	16
Intensidad por pin I/O	40mA
Intensidad en pin 3.3V	50mA
Memoria Flash	256 KB (8 KB usadas por el gestor de arranque o bootloader)
SRAM	8 KB
EEPROM	4KB
Velocidad de reloj	16 MHz
Puertos serie	4

Nota.

Elaborado por: Johanna López y Cristian Pereira

A continuación, se realiza una descripción más detallada de cada uno de las partes de la tarjeta de desarrollo Arduino Mega 2560, mostradas en la figura 15.

Partes de la placa Arduino Mega 2560

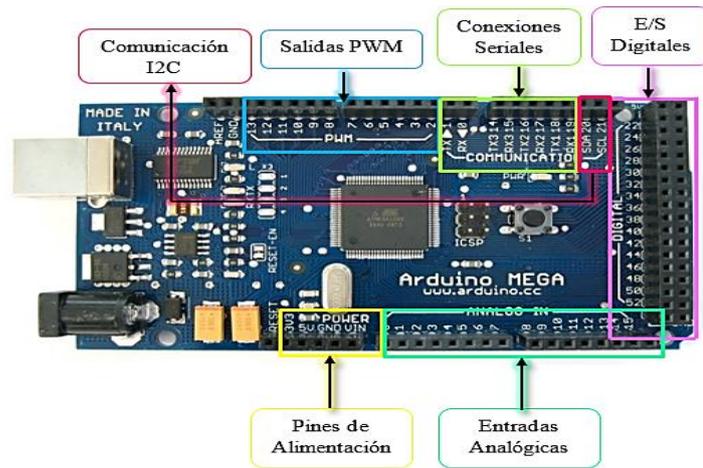


Figura 15. Descripción de cada una de las partes de la tarjeta de control

Elaborado por: Johanna López y Cristian Pereira

3.2.1.2. Microcontrolador ATmega2560

ATMega2560 es un integrado simple, de bajo consumo y costo perteneciente a la familia de los AVR's, es utilizado en las versiones actuales de la placa Arduino para los modelos mega. A continuación, se indican las características principales del microcontrolador en la tabla 20.

Tabla 20.

Características de ATmega2560

Parámetros	Características
Núcleo del Dispositivo	AVR
Arquitectura	RISC
Ancho del Bus de datos	8 bits
Frecuencia Máxima	16 MHz
Puertos de I/O programables	86
Memoria RAM	8KB
Memoria Flash	256KB
Memoria EEPROM	4KB

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.2.1.3. Memorias

Arduino Mega 2560 se basa en el microcontrolador AVR ATmega2560, el cual cuenta con una memoria flash, una RAM y una EEPROM, las cuales se detallan a continuación:

- **Memoria Flash:** El tamaño de esta memoria es de 256KB y es destinada exclusivamente para almacenar el código, dentro de los cuales 4KB son usados para el arranque del sistema (bootloader).
- **Memoria RAM:** Esta memoria de tipo volátil es el espacio donde se almacenan los sketches, su tamaño es de 8 KB y es de uso exclusivo para el programa en ejecución.
- **Memoria EEPROM:** Este tipo de memoria no volátil posee 4KB donde se puede leer y escribir datos que serán usados a largo plazo.

3.2.1.3. Alimentación de la placa

Las tarjetas de desarrollo Arduino, en general cualquier modelo pueden ser alimentadas a través de conexión USB o utilizando una fuente de alimentación externa (entre 7 y 12 voltios), las cuales son reconocidas automáticamente, sus pines de alimentación son: VIN, 5V, 3V3 y GND, descritos más adelante en la tabla 21.

Tabla 21.

Pines de alimentación del Arduino Mega 2560

PIN	Descripción
VIN	Entrada de voltaje del Arduino, es decir a través de este pin es posible proporcionar alimentación a la placa.
5V	Fuente de voltaje usada para alimentar al microcontrolador y a los componentes de la placa.
3V3	Fuente de voltaje generada en el chip FTDI integrado en la placa, de la cual se puede obtener un voltaje de 3.3V y una corriente de 50mA.
GND	Pines de toma de tierra.

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.2.1.4. Pines de entrada y salida

Los pines de E/S trabajan a 5 voltios con una intensidad máxima de 40mA, se tiene pines de entrada/salida digital y analógica, además se cuenta con ciertos pines que realizan funciones especializadas.

- **E/S Digitales:** La placa cuenta con 54 pines digitales, los cuales pueden ser usados como entradas y salidas, dependiendo de la configuración que se le asigne a cada uno de ellos. En el mismo bloque de pines, podemos encontrar arriba dos pines de 5V y abajo otros dos pines de GND. En el pin 13 se encuentra integrado un LED en la placa, cuando este pin tiene un valor HIGH el LED se enciende y cuando este tiene un valor LOW este se apaga.
- **Salidas PWM:** Son 16 pines que proporcionan salidas PWM (Pulse Wave Modulation), poseen 8 bits de resolución (valores de 0 a 255) y pueden ser definidos mediante programación con la función `analogWrite()`.
- **Entradas Analógicas:** Esta Arduino posee 16 entradas analógicas, cada una de ellas proporciona una resolución de 10bits (1024 valores), el nombre de los pines es alfanumérico y van precedidos de la letra 'A' para identificarlos.

3.2.1.5. Pines de interrupciones externas

Estos pines pueden ser configurados para enviar una interrupción en un valor LOW correspondiente a 0V, en flancos de subida o bajada para realizar un cambio de LOW a HIGH correspondiente a 5V o viceversa, y estos se encuentran en los pines mostrados en la tabla 22.

Tabla 22.

Pines de Interrupciones Externas del Arduino Mega 2560

Interrupción	PIN	Interrupción	PIN	Interrupción	PIN
0	2	1	3	2	21
3	20	4	19	5	18

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.2.1.6. Pines designados para conexión serial

Los pines para conexiones seriales son 4 y utilizan dos pines un RX para recibir y un TX para transmitir datos a través de puerto serie TTL, en la tabla 23 se puede observar cada uno de los seriales con sus pines correspondientes. Además, cabe recalcar que el serial 1 se encuentra conectado a los pines correspondientes del chip FTDI USB-a-TTL, es decir, que están conectados a la conexión a través de USB con el ordenador.

Tabla 23.

Pines de conexiones seriales disponibles en Arduino Mega 2560

	PIN RX	PIN TX
Serial 0	0	1
Serial 1	19	18
Serial 2	17	16
Serial 3	15	14

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.2.1.7. Pines con funciones varias

En la placa Arduino se tienen varios pines que realizan funciones específicas, mismas que serán detalladas en la tabla 24.

Tabla 24.

Otros pines de la placa Arduino

NOMBRE	PIN	FUNCIÓN
SDA	20	Soportan el protocolo de comunicación I2C
SCL	21	Soportan el protocolo de comunicación I2C
AREF	NA	Voltaje de referencia para las entradas analógicas.
Reset	NA	Suministra un valor LOW(0V) para reiniciar el microcontrolador

Nota.

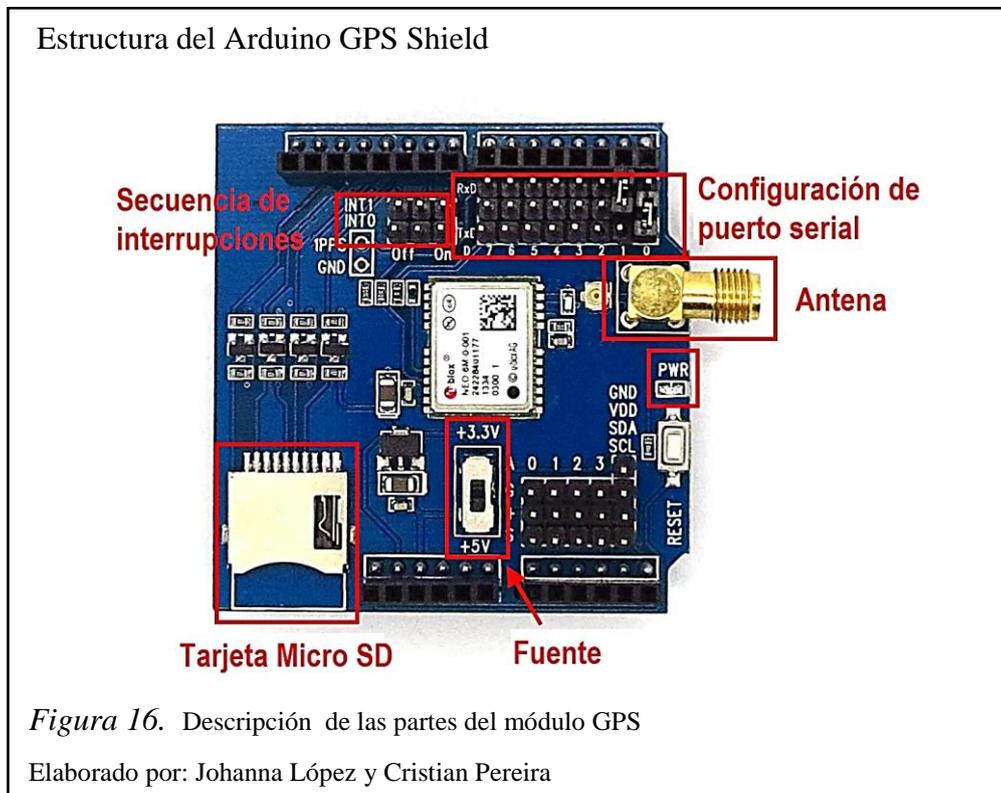
Elaborado por: Johanna López y Cristian Pereira

3.2.2. Arduino GPS Shield

Es un módulo receptor para el Sistema de Posicionamiento Global que admite a las placas Arduino conectarse a un sistema GPS, el cual permite encontrar posiciones, puntos u objetos alrededor del mundo y que además cuenta con una tarjeta micro SD integrada, la cual facilita la grabación de los datos de posición captadas por el GPS.

Su diseño utiliza el chipset SiRFstar III, que le permite mejorar notablemente el rendimiento del dispositivo GPS ofreciendo una precisión de localización inigualable, ya que tiene la capacidad de rastrear hasta 20 satélites a la vez, y realizar TTFF (Time to First Fix ó tiempo para el primer posicionamiento) rápido en entornos de señal débil.

Por otra parte, es una plataforma de hardware libre que posee librerías Open source disponibles en la web, que trabaja con bajo consumo de energía (5V/3.3V) y utiliza los pines de salida RX (para la recepción) y TX (para la transmisión) para comunicarse con las placas Arduino, así como también cuenta con una pequeña área para prototipo en donde se pueden agregar algunos componentes adicionales.



3.2.2.1. Características principales del módulo

El módulo GPS presenta varias características, dentro de las principales se tiene las descritas en la tabla 25.

Tabla 25.

Características del Arduino GPS Shield

Arduino GPS Shield	
Microcontrolador	SiRFstar III
Voltaje de funcionamiento	5V /3.3V
Interruptor DLINE/UART	salidas TX/RX
Interruptor ON/OFF	Alimentación del GPS
Botón RESET	Reiniciar el módulo
Intensidad en pin 3.3V	50mA
Zócalo	Batería para la alimentación backup del GPS

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.2.2.2. Chipset SiRFstar III

SiRFstarIII es una gama de chips microcontroladores GPS de alta sensibilidad fabricados por SiRF Technology, que interpretan las señales de los satélites GPS y determinar la posición del receptor GPS. Entre sus principales características se tiene las siguientes:

- Tiene un receptor de 20 canales, que puede procesar las señales de todos los GPS visibles.
- Consume 62 mW en funcionamiento continuo.
- Los GPS asistidos pueden reducir TTFF a menos de un segundo.
- La sensibilidad del receptor es de -159 dBm de tiempo de seguimiento.

3.2.2.3. Sentencia NMEA

NMEA por sus siglas en inglés National Marine Electronics Association, es un protocolo estándar de comunicación para la transmisión de los datos de posición del GPS entre diferentes dispositivos, compatible con la comunicación RS-232.

El código NMEA consta de una línea de texto expresada con caracteres ASCII, la cual puede contener un máximo de 82 caracteres y cada una de estas comienzan siempre con el carácter “\$”, un código de identificación de 2 dígitos conocido como “talker ID”, un código de identificación de 3 dígitos conocido como “sentence ID”, seguidos de un número determinado de campos (los que contienen la información) separados por comas, que finalizan con el checksum opcional (* y un código de 2 dígitos). (Jelsoft Enterprises Ltd. , 2003)

La sentencia NMEA que corresponden a los receptores GPS son las siguientes: GGA, GLL, GSA, RMC y VTG, mostradas en la tabla 26 y detalladas en el ANEXO 1.

Tabla 26.

Códigos NMEA principales

Sentencia	Denominación	Formato
GPGGA	Posicionamiento Global de Datos Fijos del Sistema	\$GPGGA,hhmmss.ss,llll.ll,a,mmmmm.mm,b,x,yy,z,z,c.c,M,v.v,M,d.d,www*hh
GPGSA	GPS DOP (calidad de la señal) y satélites activos	\$GPGSA,x,y,z1,z2,z3,z4,z5,z6,z7,z8,z9,z10,z11,z12,p,p,h,h,v.v*ff
GPGSV	Información de cada satélite	\$GPGSV,x,y,zz,aa,ee,rr,ii,s1,s2,s3,s4,t1,t2,t3,t4,c1,c2,c3,c4*ff
GPGLL	Posición geográfica, latitud y longitud	\$GPGLL,llll.ll,a,mmmm.mm,b,hhmmss.ss,A*ff

Nota.

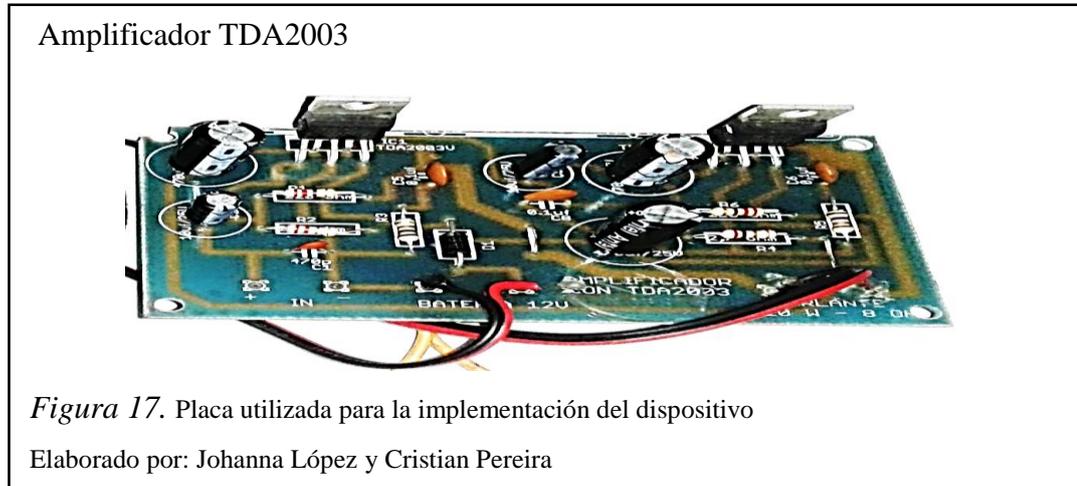
Elaborado por: Johanna López y Cristian Pereira

3.2.3. Modulo Amplificador TDA2003

Este amplificador está basado en dos circuitos integrados TDA2003, los cuales cuentan con la misma configuración y distribución de los pines del TDA2002 ya que es una versión mejorada del este, su uso es variado permitiendo amplificar la señal de una radio, un mp3, un micrófono, otros y a pesar de la sencillez de su circuito, se puede obtener una calidad de audio excelente.

La configuración de estos dos circuitos utilizados para el amplificador, está diseñada para que en sus salidas presenten señales desfasadas entre sí 180°, logrando que uno de ellos funcione como amplificador inversor.

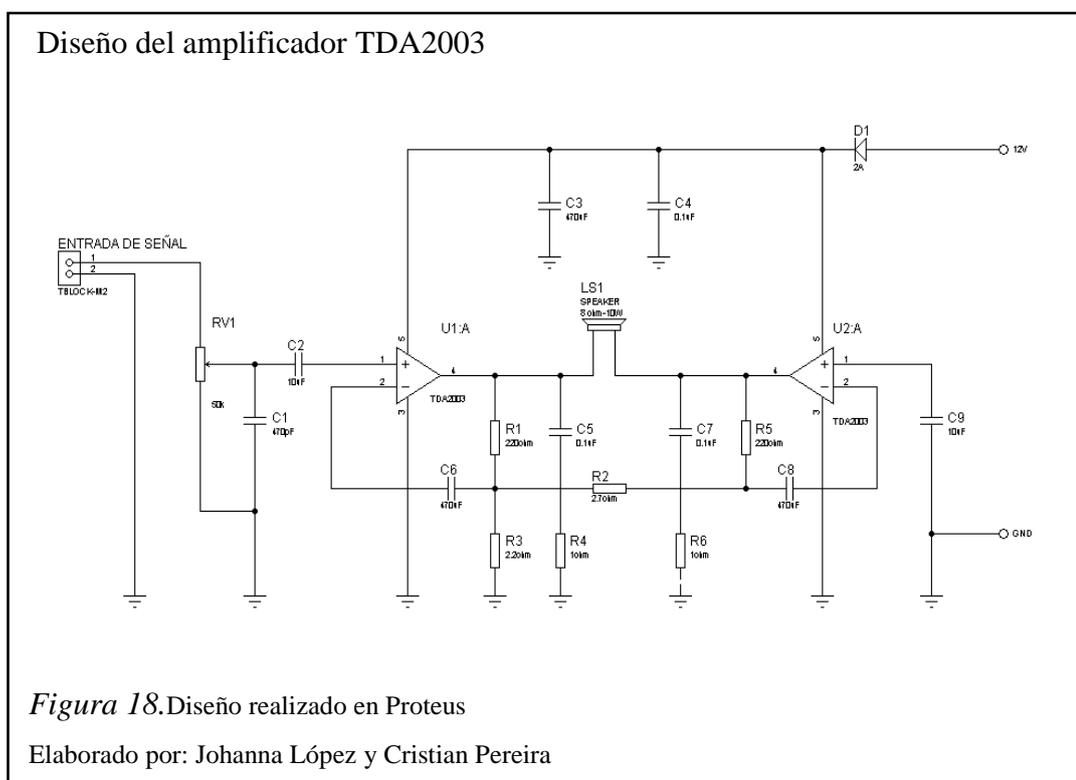
El TDA2003 proporciona una corriente de salida de hasta 3,5 A, posee buena potencia y baja distorsión armónica o cross-over y posee protección contra corto circuito tanto para corriente alterna como para corriente continua, por lo que garantiza la operación con bajo ruido.



Los valores recomendados para este circuito amplificador son los descritos a continuación y mostrados en la figura 18.

- Capacitor de 10 μ F, el cual es utilizado para el desacople de entrada, si se utiliza un capacitor de mayor valor limita la respuesta en baja frecuencia, y se utiliza uno de menor valor, este puede presentar ruido e interferencias.

- Capacitor de 0.1 μ F, el cual permite que las señales de alta frecuencia que pudieran llegar, no sean amplificadas.
- Capacitor de 470 pF, este es utilizado como capacitor de acople de salida.
- Capacitor de 470 μ F, funciona como un estabilizador de frecuencia.
- Resistencias de 220 Ω y 2.2 Ω son aquellas que se encargan de fijar la ganancia del circuito TDA2003.
- Resistencia de 1 Ω , trabaja junto el capacitor de 0.1 μ F para mantener la estabilidad de desempeño del amplificador.



3.3. Estudio de Factibilidad de Software

Un dispositivo electrónico necesita de un conjunto de instrucciones lógicas para hacer posible su funcionamiento y la realización de todas las tareas a las cuales será encomendado, sin embargo no solo necesita de estas instrucciones sino más bien de la combinación tanto de la parte lógica como de su parte física.

3.3.1. Lenguajes de Programación de microcontroladores.

En la actualidad, se pueden encontrar varios lenguajes de programación que poseen similares características y funcionalidades, los cuales pueden ser usados de acuerdo a las necesidades del programador, para lo cual, es indispensable realizar un análisis detallado en función de las propiedades y facilidades que cada uno de los lenguajes de programación puedan ofrecer.

3.3.1.1. Assembler

El lenguaje ensamblador fue uno de los lenguajes de programación más utilizados tiempos atrás para la creación de programas, sin embargo, en la actualidad es usado pero no con mucha frecuencia solo en casos especiales cuando es necesario mantener el control directo de un equipo.

La ventaja principal de assembler es que contribuye en el aprovechamiento de recursos del microcontrolador, además que permite controlar los tiempos y registros de cada uno de los bits.

En la tabla 27, se realiza un estudio de factibilidad en el cual se toman en cuenta ciertos parámetros para realizar dicha evaluación.

Tabla 27.

Matriz de evaluación del lenguaje de programación Assembler

Solución	Puntaje 10%	Método de Evaluación
Assembler	3	Interfaz de usuario
	6	Costo de adquisición
	5	Costo de mantenimiento
	4	Tipo de certificación ISO

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.3.1.2. Basic

Este es uno de los lenguajes de programación de alto nivel diseñado para programadores principiantes, ya que no requiere tener conocimientos extensos en programación y es muy fácil de aprender.

Una de sus principales ventajas es que este lenguaje es muy parecido al usado por los humanos en la vida diaria haciendo muy sencillo trabajar y entender sus instrucciones.

Se realiza un análisis de factibilidad, el cual tiene como objetivo evaluar ciertos parámetros importantes que ayuden a la elección del lenguaje apropiado para la realización de este proyecto, como se observa en la tabla 28.

Tabla 28.

Matriz de evaluación del lenguaje de programación Basic

Solución	Puntaje 10%	Método de Evaluación
Basic	7	Interfaz de usuario
	8	Costo de adquisición
	7	Costo de mantenimiento
	7	Tipo de certificación ISO

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.3.1.3. Lenguaje C

C es un lenguaje de programación que presenta varias características de un lenguaje de bajo nivel y aunque al igual que Basic presenta estructuras de alto nivel, es considerado más bien como un lenguaje de programación de nivel medio, el cual cuenta con licencia libre y su sintaxis es muy sencilla, siendo de esta forma uno de los lenguajes de programación más utilizados para el desarrollo de software orientado a sistemas.

En la tabla 29, se realiza una evaluación de parámetros importantes al momento de elegir un lenguaje de programación que brinde todo lo necesario al programador.

Tabla 29.

Matriz de evaluación del lenguaje de programación C

Solución	Puntaje 10%	Método de Evaluación
C	8	Interfaz de usuario
	9	Costo de adquisición
	9	Costo de mantenimiento
	10	Tipo de certificación ISO

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.3.1.4. Jerarquización de la Factibilidad

Los criterios tomados en cuenta para jerarquizar la factibilidad de cada solución serán evaluados en función de la interfaz que presenta al usuario, el costo de adquisición, el costo de mantenimiento y el tipo de certificación, los cuales son los parámetros más importantes para conocer cada uno de los lenguajes de programación y sus bondades.

Tabla 30.

Cuadro comparativo de puntajes del lenguaje de programación.

Parámetros	Assembler			Basic			C		
	Puntaje	Peso	Total	Puntaje	Peso	Total	Puntaje	Peso	Total
Interfaz de usuario	3	0,3	0,9	7	0,3	2,1	8	0,3	2,4
Costo de adquisición	6	0,4	2,4	8	0,4	0,5	9	0,4	3,6
Costo de mantenimiento	5	0,1	0,5	7	0,1	3,2	9	0,1	0,9
Certificación ISO	4	0,1	0,5	7	0,1	0,7	10	0,1	1
Puntaje Total	4,5	8,2	8,8

Nota.

Elaborado por: Johanna López y Cristian Pereira

De acuerdo a los resultados arrojados en la tabla 30, el lenguaje de programación que presenta mayores ventajas es el lenguaje C, el cual, se describirá más detalladamente a continuación.

3.3.2. Entornos de Desarrollo IDE.

Los IDE's proporcionan un conjunto de herramientas que facilitan la realización de programas, asociando varios lenguajes de programación y de forma sencilla ya que permite utilizar dichos lenguajes en una manera interactiva. Para el desarrollo de este proyecto se han tomado como base para el análisis de factibilidad Labview, Matlab y Arduino.

3.3.2.1. Labview

Labview es un software que utiliza lenguaje de programación gráfico, se basa principalmente en la relación humano - máquina y es orientado a realizar grandes y pequeños procesos a nivel industrial o educativo, como es el caso de la adquisición y procesamiento de datos mediante las tarjetas de desarrollo de la marca Arduino. El software Labview permite la posibilidad de incorporar aplicaciones realizadas en otros lenguajes de programación, además de ser de fácil aprendizaje presenta un compilador gráfico basado en diagramas de bloques y adaptabilidad con varios protocolos de comunicación, la desventaja mayor que presenta Labview es el costo elevado de las licencias .

Tabla 31.

Matriz de evaluación del software Labview

Solución	Puntaje 10%	Método de Evaluación
Labview	8	Facilidades de aprendizaje
	10	Interfaz de usuario
	10	Costo de adquisición
	10	Costos de actualizaciones
	9	Documentación

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.3.2.2. Matlab

Matlab es un software potente basado en lenguaje de programación C que utiliza vectores y matrices para realizar operaciones lógicas y matemáticas, este utiliza librerías denominadas toolboxes que permiten incorporar más procesos para facilitar la programación, estos permiten también la adaptación de hardware que no necesariamente son pertenecientes al software. Entre las principales ventajas que posee este software es que permite la integración con dispositivos hardware, se puede encontrar variada documentación, es de fácil manejo y cuenta con alto rendimiento en la ejecución de procesos, su única desventaja es el costo de adquisición.

Tabla 32.

Matriz de evaluación del software Matlab

Solución	Puntaje 10%	Método de Evaluación
Matlab	6	Facilidades de aprendizaje
	7	Interfaz de usuario
	1	Costo de adquisición
	1	Costos de actualizaciones
	10	Documentación

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.3.2.3. Arduino

Arduino es un software libre basado en el lenguaje C de alto nivel diseñado para el desarrollo de código de programación de manera sencilla, cuenta con un entorno gráfico muy amigable para sus usuarios, permitiendo que en los últimos años tenga una gran acogida en la realización de proyectos, ya que permite desde el encendido y apagado de un led, así como también el diseño de sistemas de control, por lo que es considerado como una alternativa eficiente y de alto rendimiento.

Tabla 33.

Matriz de evaluación del software Arduino

Solución	Puntaje 100%	Método de Evaluación
Arduino	10	Facilidades de aprendizaje
	7	Interfaz de usuario
	10	Costo de adquisición
	10	Costos de actualizaciones
	8	Documentación

Nota.

Elaborado por: Johanna López y Cristian Pereira

3.3.2.4. Jerarquización de la Factibilidad

Los criterios tomados en cuenta para jerarquizar la factibilidad de cada solución serán evaluados en función de la facilidad de aprendizaje del software, la interfaz del usuario, el costo de adquisición, el costo de actualizaciones y la documentación presentada al usuario, los cuales son los parámetros más importantes para conocer cada uno de los lenguajes de programación y sus bondades.

Tabla 34.

Cuadro comparativo de puntajes de los IDE's de programación.

Parámetros	Labview			Matlab			Arduino		
	Puntaje	Peso	Total	Puntaje	Peso	Total	Puntaje	Peso	Total
Facilidades de aprendizaje	8	0,3	2,4	6	0,3	1,8	10	0,3	3
Interfaz de usuario	10	0,1	1	7	0,1	0,7	7	0,1	0,7
Costo de adquisición	10	0,4	4	1	0,4	0,4	10	0,4	4
Costos de actualizaciones	10	0,1	1	1	0,1	0,1	10	0,1	1
Documentación	9	0,1	0,9	10	0,1	1	8	0,1	0,8
Puntaje Total	---	---	9,3	---	---	4	---	---	9,5

Nota.

Elaborado por: Johanna López y Cristian Pereira

De acuerdo a los resultados arrojados en la tabla 34, el entorno de desarrollo que evidencia mayores ventajas es Arduino, el cual, se describirá más detalladamente a continuación.

3.4. Descripción del software seleccionado

3.4.1. Lenguaje de programación C

Es un lenguaje de programación estructurado, considerado de nivel medio ya que combina tanto el lenguaje de programación de alto y de bajo nivel, este programa proporciona flexibilidad al momento de programar, dando como resultado que el código desarrollado sea exclusivamente responsabilidad de los programadores.

Las características y ventajas que presenta el lenguaje de programación C son:

- Diseñado para crear aplicaciones y software de sistemas.
- Es un lenguaje orientado a objetos
- Lenguaje con comandos similares al vocablo humano
- Posee librerías que ayudan a la programación
- Se acopla con nuevas plataformas de hardware

Este lenguaje de programación también presenta desventajas con respecto a otros lenguajes, estas son:

- Utiliza más recursos de memoria a diferencia del lenguaje assembler.
- No utiliza toda la potencia de hardware, lo cual impide el uso de todos los recursos que posee este en su totalidad.

3.4.2. IDE Arduino

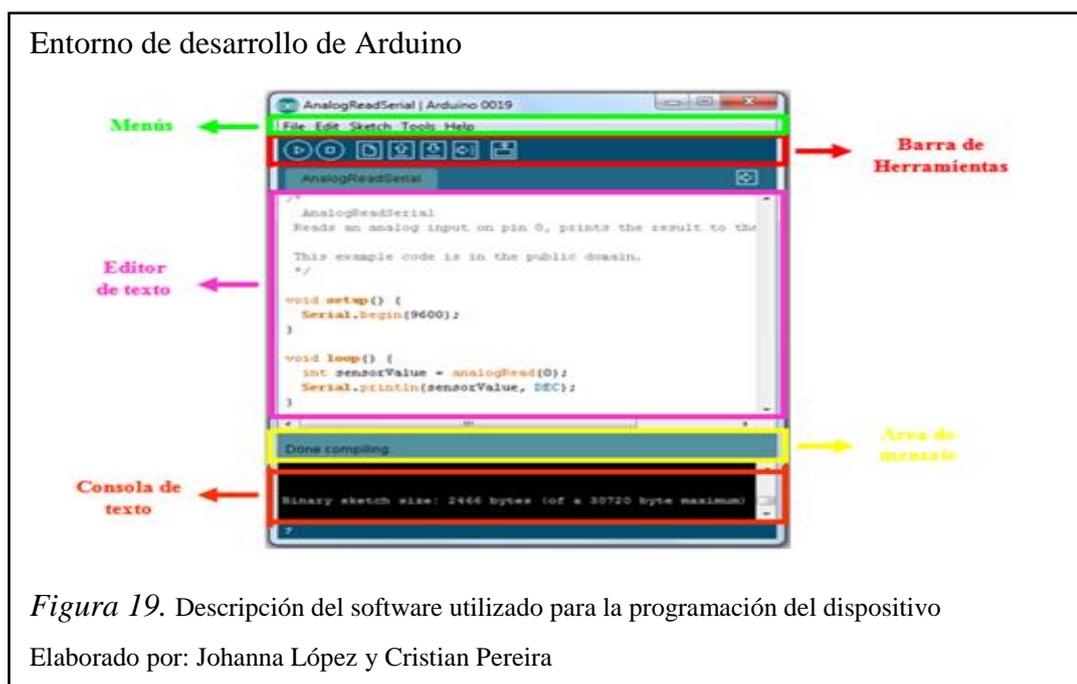
La plataforma Arduino cuenta con un software propietario de código abierto, basado en el lenguaje de programación Writing y el entorno de desarrollo en Processing escrito en Java, sin embargo, es posible utilizar otros lenguajes de programación y aplicaciones afines al lenguaje de programación de Arduino. El entorno de desarrollo

conocido como IDE, “está constituido por un editor de textos para escribir el código, una área de mensajes, una consola de textos, una barra de herramientas con botones para las funciones comunes y una serie de menús” (Pazmiño Pineda , 2012, pág. 35), como se observa en la figura 19.

La barra de herramientas cuenta con las siguientes opciones:

-  **Verify:** Explora el código en busca de errores.
-  **Stop:** Detiene la monitorización serie y oculta otros botones.
-  **New:** Crea un nuevo sketch o programa.
-  **Open:** Presenta un “sketchbooks” o librería sketch.
-  **Save:** Guarda el sketch.
-  **Upload:** Compila el código y lo carga a la placa Arduino.
-  **Serial Monitor:** Inicializa la monitorización serial.

Dentro de la barra de menús se pueden encontrar algunas de las opciones de la barra de herramientas, además de otras que permiten abrir archivos existentes, guardar cambios, imprimir, editar, copiar, pegar, encontrar, etc. En efecto, el software Arduino cuenta con un entorno de programación fácil de manejar por personas entendidas en el tema e incluso por aquellos principiantes que deseen hacer uso del mismo.



Antes de empezar a trabajar en la programación dentro del entorno Arduino, se debe configurar los siguientes parámetros dentro del menú "herramientas":

- Para configurar la comunicación entre la placa Arduino y la computadora se dentro de la opción "Puerto Serial" se elegirá el puerto serial al cual se conectará.
- Dentro de la opción "Tarjeta" seleccionar la tarjeta Arduino con la que se trabajará.

Menú Herramientas de Arduino.

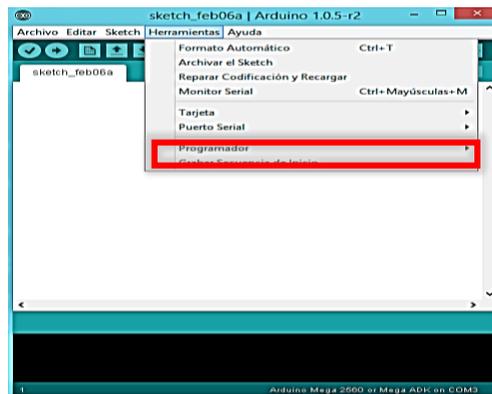


Figura 20. Configuración del puerto de comunicación (Computadora - Tarjeta de control)

Elaborado por: Johanna López y Cristian Pereira

Una vez realizadas estas configuraciones el entorno para crear un programa estará listo, como se muestra en la figura 21.

Programa para encender y apagar un led.

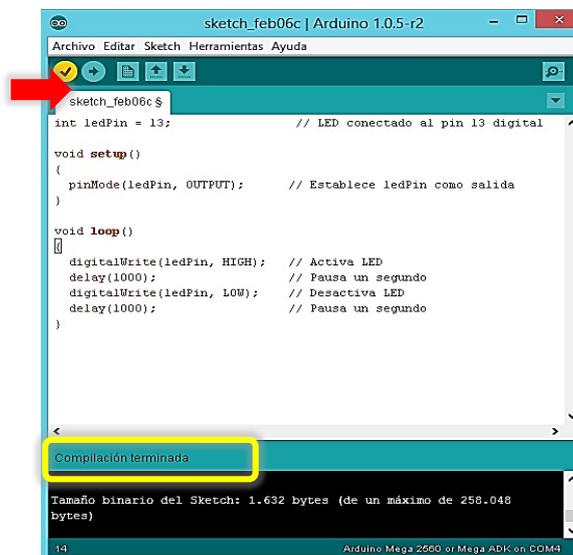
```
sketch_feb06c | Arduino 1.0.5-r2
Archivo Editar Sketch Herramientas Ayuda
sketch_feb06c $
int ledPin = 13; // LED conectado al pin 13 digital
void setup()
{
  pinMode(ledPin, OUTPUT); // Establece ledPin como salida
}
void loop()
{
  digitalWrite(ledPin, HIGH); // Activa LED
  delay(1000); // Pausa un segundo
  digitalWrite(ledPin, LOW); // Desactiva LED
  delay(1000); // Pausa un segundo
}
```

Figura 21. Código de programación

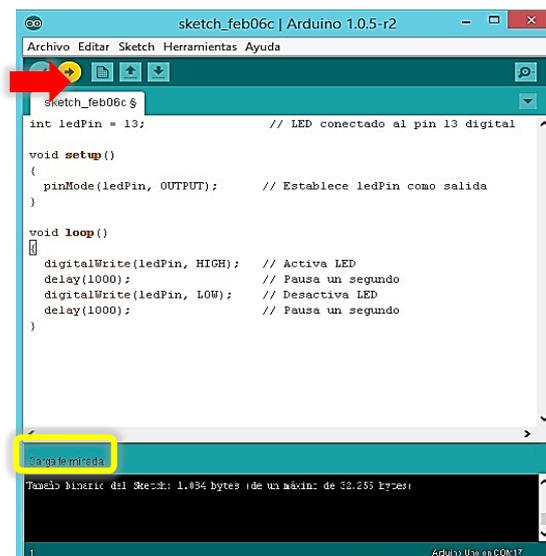
Elaborado por: Johanna López y Cristian Pereira

Al terminar el código de programación primero es importante comprobar que el código fuente sea correcto, para ello se deberá dar clic en el botón de verificación de código como se observa en la figura 22.a si es correcto se desplegará un mensaje en la parte inferior de la interfaz indicando "Compilación Terminada", luego de que el código haya sido verificado se procederá a cargarlo en la tarjeta de control, si la carga es correcta se desplegará un mensaje en la parte inferior de la interfaz indicando "Carga Terminada", como se observa en la figura 22.b.

Menú Herramientas de Arduino.



a. Compilación del programa



b. Carga del programa

Figura 22. Compilación y carga del programa en la tarjeta Arduino

Elaborado por: Johanna López y Cristian Pereira

3.4.2.1.Estructura

El lenguaje de programación Arduino tiene como estructura básica dos partes fundamentales conocidas como funciones, y estas son:

voidsetup(): se ejecuta para inicializar las comunicaciones e iniciar un sketch, además se hace uso de esta instrucción para establecer el estado de los pines e iniciar variables.

voidloop(): esta instrucción permite que el sketch se ejecute continuamente, es decir es el núcleo del programa, ya se realiza la mayor parte del trabajo.

La estructura que en este lenguaje se utiliza es básicamente muy similares a la de C y C++.

CAPÍTULO 4

DESARROLLO DE HARDWARE Y SOFTWARE

En este capítulo, se analizará el diseño de hardware y el desarrollo de software de programación basada en el entorno de desarrollo Arduino, el cual tiene como objetivo permitir la implementación del sistema de alerta para pico y placa.

4.1. Diseño de Hardware

En el diseño del hardware basado en los componentes que forman parte del sistema de alerta para el pico y placa, el cual se encuentra estructurado por una tarjeta de adquisición y procesamiento de datos Arduino Mega 2560, sensor Arduino GPS Shield, fuente de alimentación e indicadores visuales y auditivos.

4.1.1. Diagrama de Bloques

El diagrama de bloques que se muestra en la figura 23, representa sistema de alerta para pico y placa, el cual está conformado básicamente por la etapa de adquisición y procesamiento de datos, la etapa de posicionamiento y la etapa de visualización y alerta, mismos que se detallan a continuación.

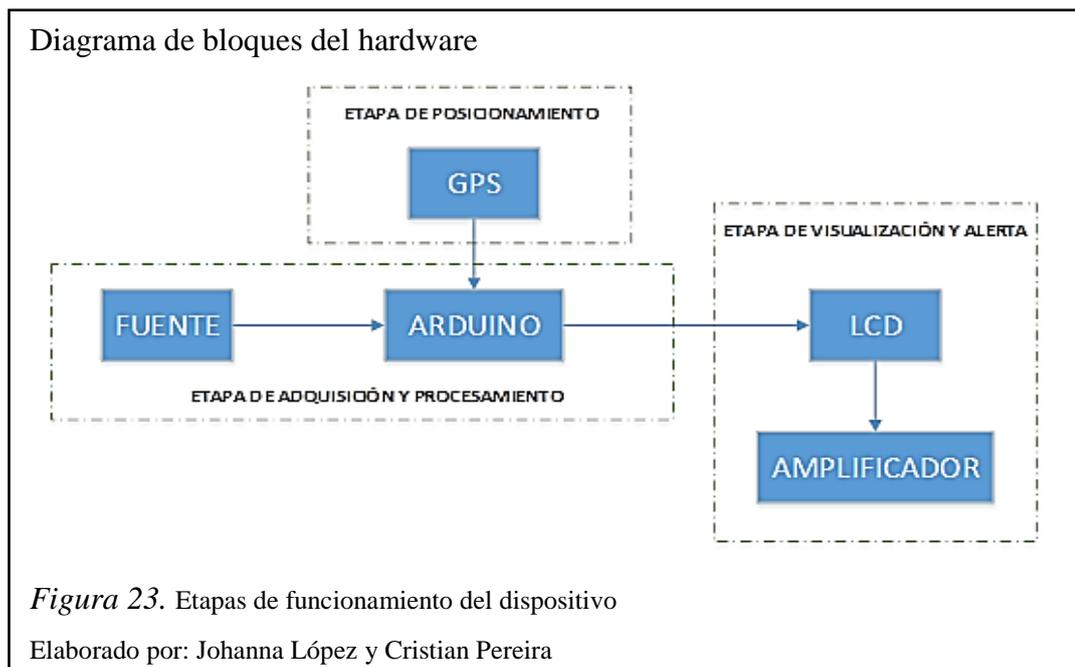


Figura 23. Etapas de funcionamiento del dispositivo

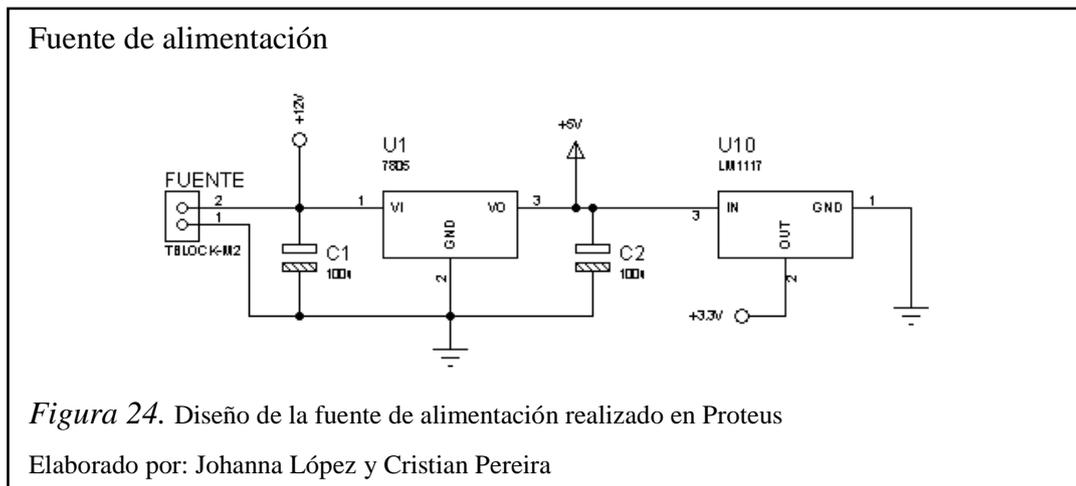
Elaborado por: Johanna López y Cristian Pereira

4.1.1.1. Etapa de adquisición y procesamiento de datos.

4.1.1.1.1. Fuente de alimentación

El sistema de alerta para pico y placa se encuentra alimentado por una fuente de alimentación DC de 12V que toma el voltaje de la batería del automóvil y mediante el circuito de la figura 24 se baja al nivel de voltaje para la alimentación de la tarjeta de control Arduino, para el módulo GPS y para la etapa de visualización y alerta.

El circuito se encuentra conformado por un regulador de voltaje 7805, el cual es un dispositivo electrónico que permite bajar los niveles de voltaje a 5V, además de constar con un regulador LM1117, que permite bajar los niveles de voltaje a 3.3 V.



4.1.1.1.2. Arduino Mega 2560

Este módulo fue el encargado de recibir la señal emitida del módulo GPS a los pines 18 y 19 correspondientes al de recepción y transmisión respectivamente, estos datos luego fueron utilizados en la programación para comparar las zonas de restricción del pico y placa, además utilizó un módulo RTC DS1307 conectado a los pines 20 y 21 correspondientes a SDA y SCL para la comunicación I2C, el cual permite obtener la hora, fecha y día. También se utilizó una pantalla LCD conectada a los pines 22, 23, 24, 25, 26, 27 correspondientes a D7, D6, D5, D4, RS y E para la visualización de los datos adquiridos por el módulo GPS, las alertas de voz son controladas mediante un

reproductor WTV020SD que se encuentra conectado a los pines 36, 37, 38, 39. El número de placa correspondiente al vehículo se ingresa mediante pulsadores configurados con resistencias pull up, mismas que se encuentran conectadas en los pines 28, 29, 30, 31 del arduino.

El esquema de conexión de la tarjeta Arduino y los componentes utilizados para la implementación del sistema de alerta para el pico y placa, se muestran en la figura 25.

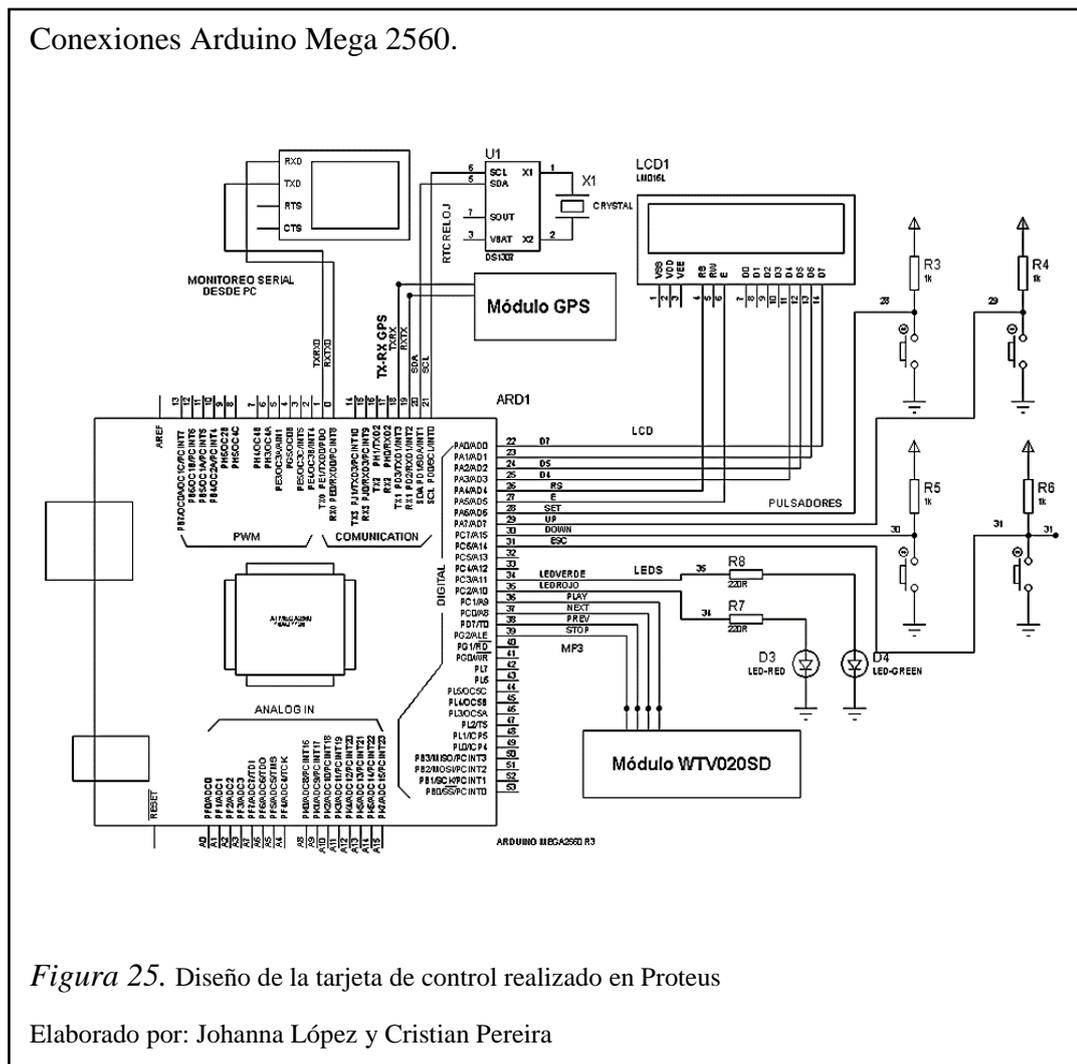


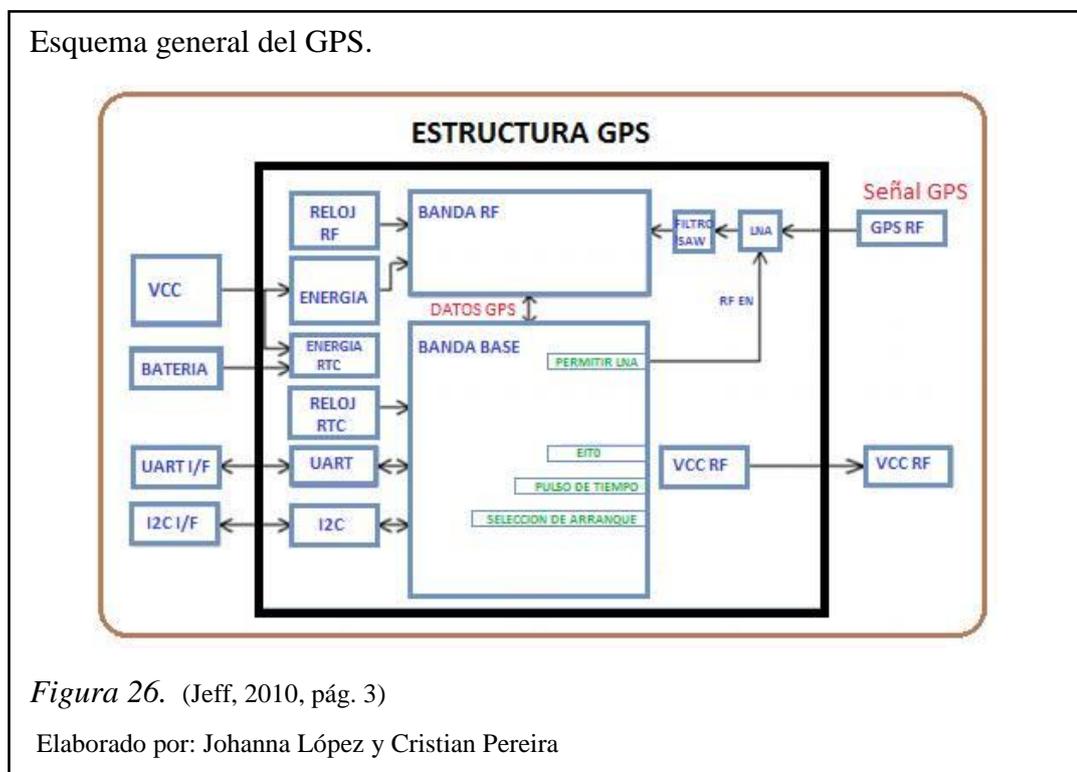
Figura 25. Diseño de la tarjeta de control realizado en Proteus

Elaborado por: Johanna López y Cristian Pereira

4.1.1.2. Etapa de posicionamiento

4.1.1.2.1. Módulo GPS

El módulo GPS se encarga de la recepción de valores cuantitativos de latitud y longitud; provee señal serial mediante la conexión en los pines RX y TX los cuales son vinculados a los pines respectivos de la tarjeta Arduino. Este módulo utiliza una antena magnética con conector tipo SMA (SubMiniature versión A, un tipo de conector roscado para cable coaxial), que trabaja a 3MHz.



4.1.1.2.2. Módulo RTC DS1307

RTC DS1307 utiliza comunicación I2C y se conecta a la tarjeta arduino para proveer información de los segundos, minutos, horas y fecha, además posee una batería, la cual es encargada de proveer energía a la memoria RAM para almacenamiento de datos.

La conexión del módulo RTC DS1307 se muestra a continuación en la figura 27.

Conexiones RTC DS1307.

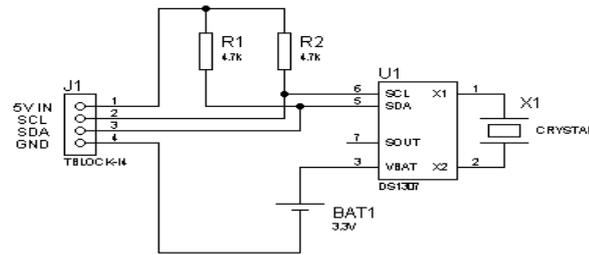


Figura 27. Diseño de conexión del reloj de tiempo real realizado en Proteus

Elaborado por: Johanna López y Cristian Pereira

4.1.1.2.3. Módulo WTV020SD

El WTV020SD es un pequeño módulo que permite reproducir audio, esta placa cuenta con una ranura que permite insertar una tarjeta MicroSD, para su funcionamiento y reproducción de las pistas el módulo requiere que estas sean almacenadas en la tarjeta en un formato especial (4-bit ADPCM y frecuencia de muestreo ad4 de 6-32kHz), ya que no soporta directamente archivos WAV o MP3 y se encuentra conectado en la placa Arduino Mega 2560 mediante modo serial haciendo uso de los pines que proporciona DATA y CLOCK.

El circuito del módulo WTV020SD se muestra a continuación en la figura 28.

Circuito WTV020SD.

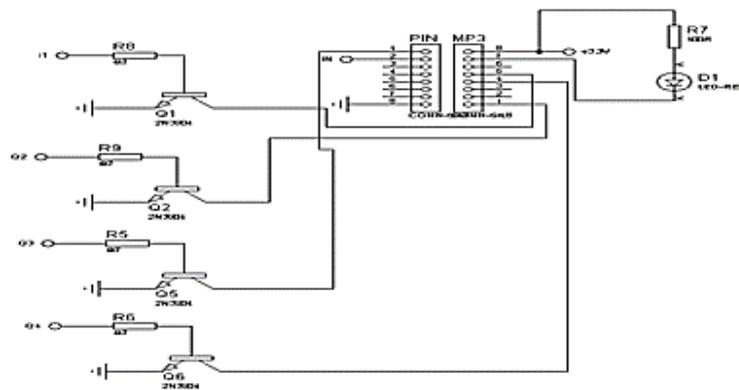


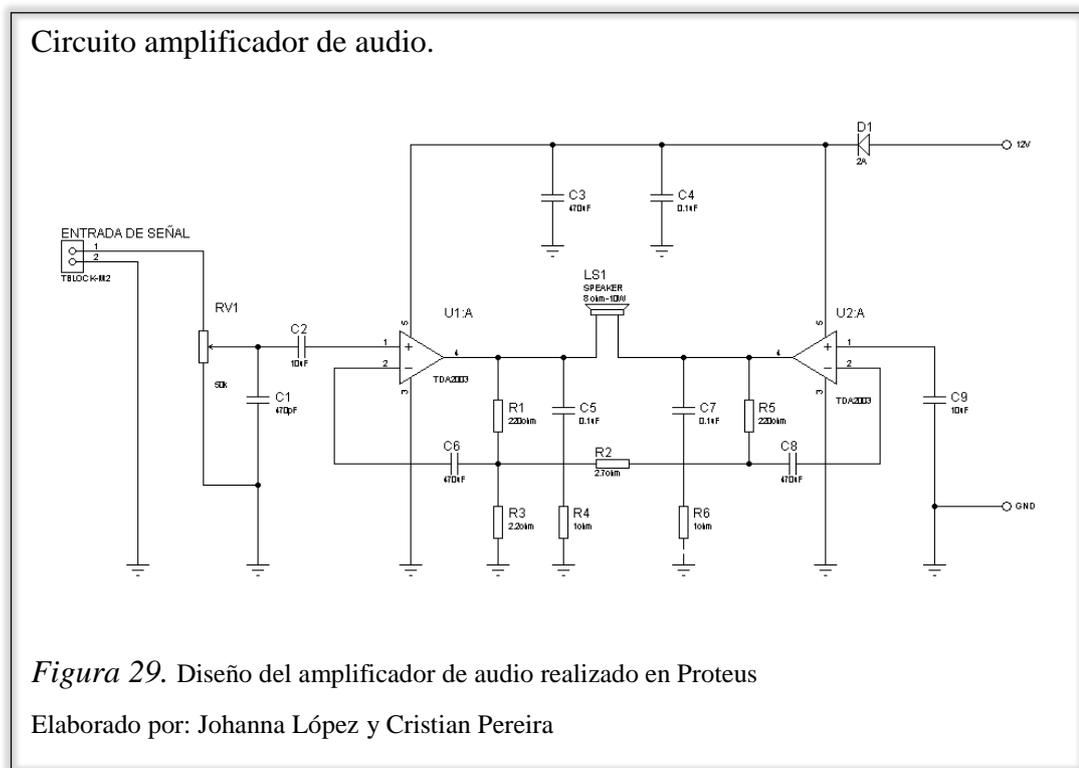
Figura 28. Diseño del reproductor de audio realizado en Proteus

Elaborado por: Johanna López y Cristian Pereira

4.1.1.3. Etapa de visualización y alertas

4.1.1.3.1. Amplificador de audio

El amplificador de audio es el encargado de emitir los mensajes de voz correspondientes a la alerta de pico y placa, está conformado por 2 circuitos integrado TD2003 y cuenta con una protección contra corto circuitos, el voltaje para el amplificador es tomado de la tarjeta secundaria y se encuentra conectado a la salida del reproductor de audio. El circuito del amplificador de audio se muestra a continuación en la figura 29.



4.1.1.3.2. Pantalla LCD

Esta pantalla permite visualizar los valores correspondientes a la longitud y latitud obtenida del módulo GPS, se encuentra conectado como se indica a continuación en la tabla 35.

Tabla 35.

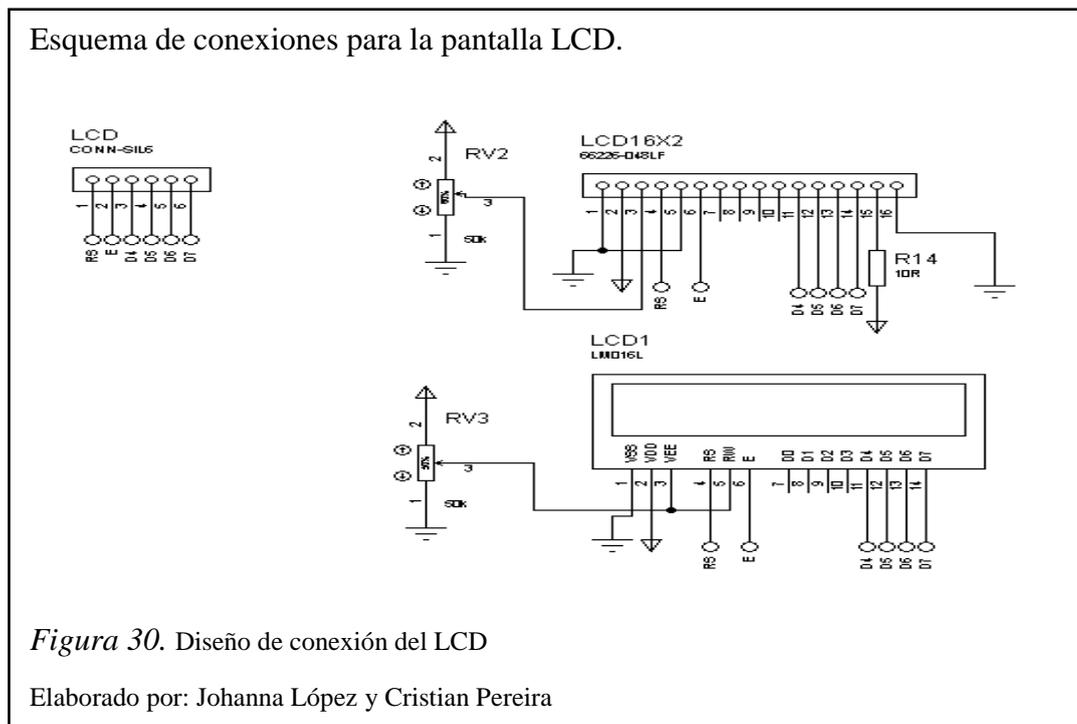
Conexiones de la pantalla LCD

	LCD (Pines)	Tarjeta Secundaria	Módulo Arduino (Pines)
1	RS	NA	26
2	E	NA	27
3	D4	NA	25
4	D5	NA	24
5	D6	NA	23
6	D7	NA	22
7	VSS	GND	NA
8	VDD	5V	NA
9	VO	Potenciómetro	NA

Nota.

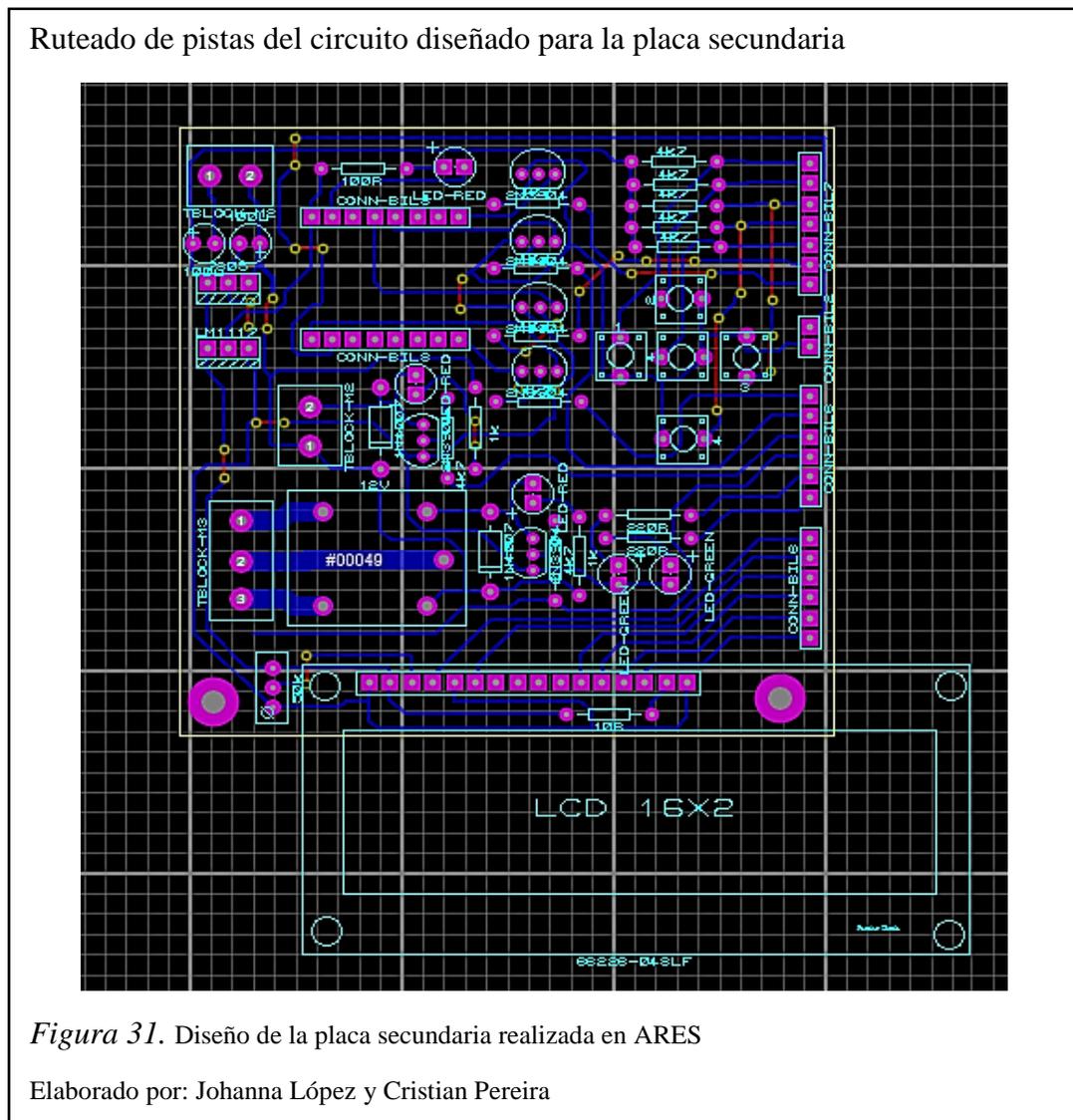
Elaborado por: Johanna López y Cristian Pereira

La conexión de la Pantalla LCD y la placa Arduino se muestra a continuación en la figura 30.



4.1.1.4. Diagrama de la placa secundaria

Se diseñó e implemento una placa electrónica secundaria, la cual es utilizada para integrar cada uno de los circuitos utilizados para la construcción del sistema de alerta para pico y placa. Esta placa fue desarrollada primeramente en el software PROTEUS, una vez realizada la simulación con todos los elementos que conforman el circuito, fue pasada al software ARES usado comúnmente para la fabricación de placas electrónicas, en donde se realizó el ruteo de las pistas, obteniendo el diseño de la placa tal como se muestra en la figura 31 y 32.



Vista 3D de la placa electrónica secundaria

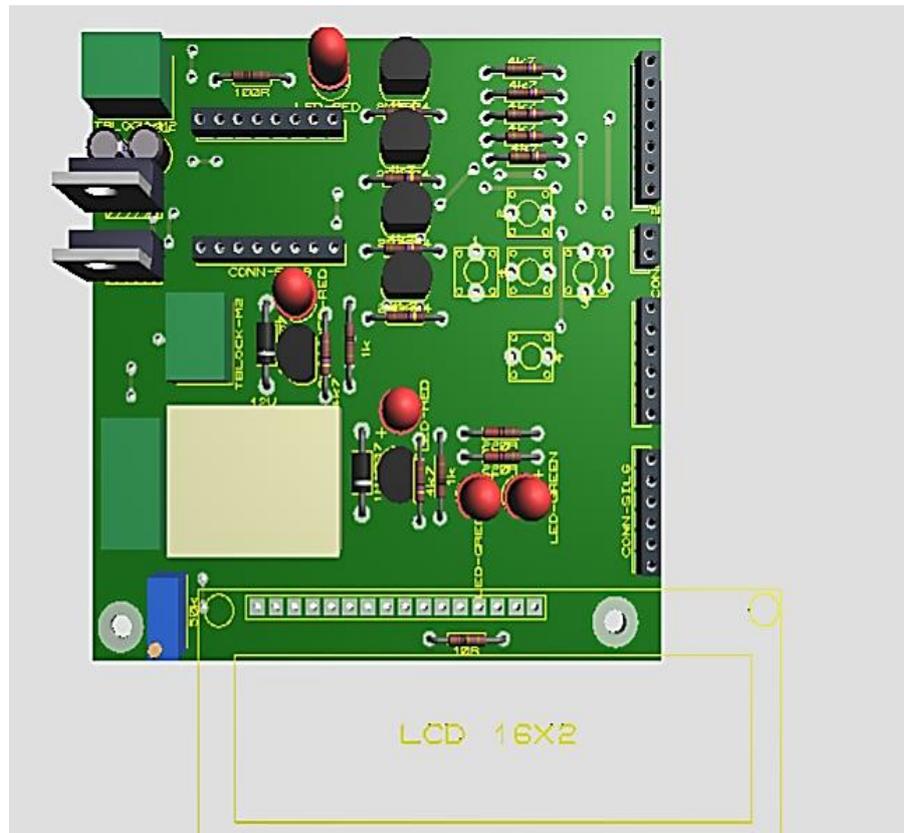


Figura 32. Diseño de la placa secundaria realizada en ARES

Elaborado por: Johanna López y Cristian Pereira

4.1.1.5. Diagrama Esquemático de control del sistema de alerta

En la figura 33, se muestra el diagrama de conexión completo correspondiente al sistema de alerta para pico y placa, el cual se encuentra conformado por cada uno de los elementos mencionados anteriormente en este capítulo.

Diagrama de conexión del sistema de alerta para pico y placa

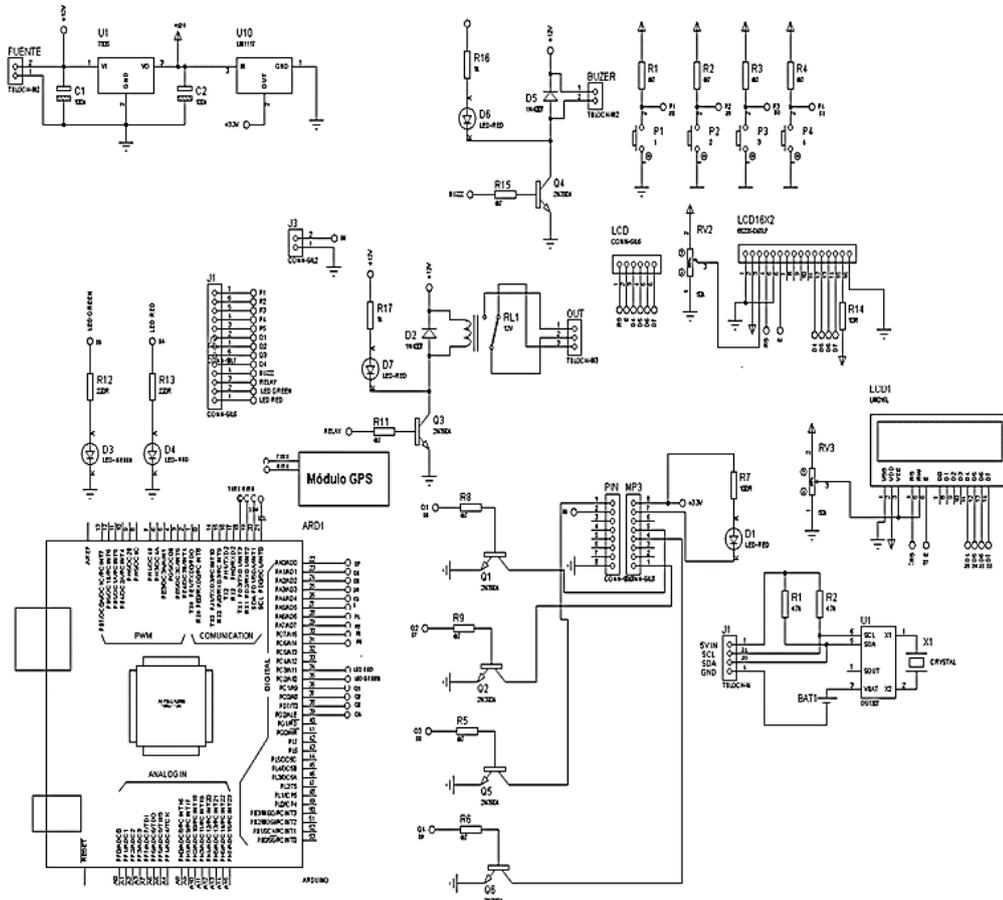


Figura 33. Diseño del circuito completo realizado en Proteus para el dispositivo

Elaborado por: Johanna López y Cristian Pereira

Implementación del sistema de alerta para pico y placa.



Figura 34. Dispositivo de alerta para Pico y Placa implementado

Elaborado por: Johanna López y Cristian Pereira

4.2. Diseño de Software

El diseño del software se encuentra basado en un grupo de instrucciones lógicas que permiten el control del sistema de alerta para pico y placa, para lo cual, se analizará el diagrama de flujo de la programación a ser cargada en el microcontrolador.

4.2.1. Diagrama de Flujo

La lógica de programación implementada en el control del sistema de alerta para pico y placa será descrita a continuación en cada uno de sus bloques funcionales. En la figura 35, se muestra el diagrama principal diseñado para el funcionamiento del dispositivo, mismo que es explicado a continuación:

- 1.** Inicio del programa.
- 2.** Inicialización y declaración de variables a ser utilizadas, variables tipo int, float y unsigned long.
- 3.** Adquisición el número de la placa que fue configurado previamente, para realizar una comparación y validar si el día y número de placa corresponde al de restricción de pico y placa, emite la alerta “Recuerde, hoy su vehículo tiene pico y placa” de acuerdo a lo descrito a continuación, caso contrario regresa a pedir el número de la placa:
 - Si el día es igual a uno la placa corresponde al número 1 y 2.
 - Si el día es igual a dos la placa corresponde al número 3 y 4.
 - Si el día es igual a tres la placa corresponde al número 5 y 6.
 - Si el día es igual a cuatro la placa corresponde al número 7 y 8.
 - Si el día es igual a cinco la placa corresponde al número 9 y 0.
- 4.** Adquisición de longitud y latitud proporcionada por el GPS.
- 5.** Comparación de los datos recibidos de las zonas de pico y placa para validar la ubicación del vehículo.
 - 5.1.** Si la hora y fecha coinciden con el horario establecido y se encuentra el vehículo dentro de dicho perímetro, como se muestra en la figura 35, emitirá la siguiente alerta de voz “Recuerde, hoy su vehículo tiene pico y placa y está dentro del perímetro”.

5.2. Si la hora y fecha coinciden con el horario establecido y el vehículo se encuentra acercándose al perímetro de restricción para pico y placa, como se muestra en la figura 35 y 36, se emitirá la siguiente alerta de “Atención, está acercándose al perímetro de pico y placa”.

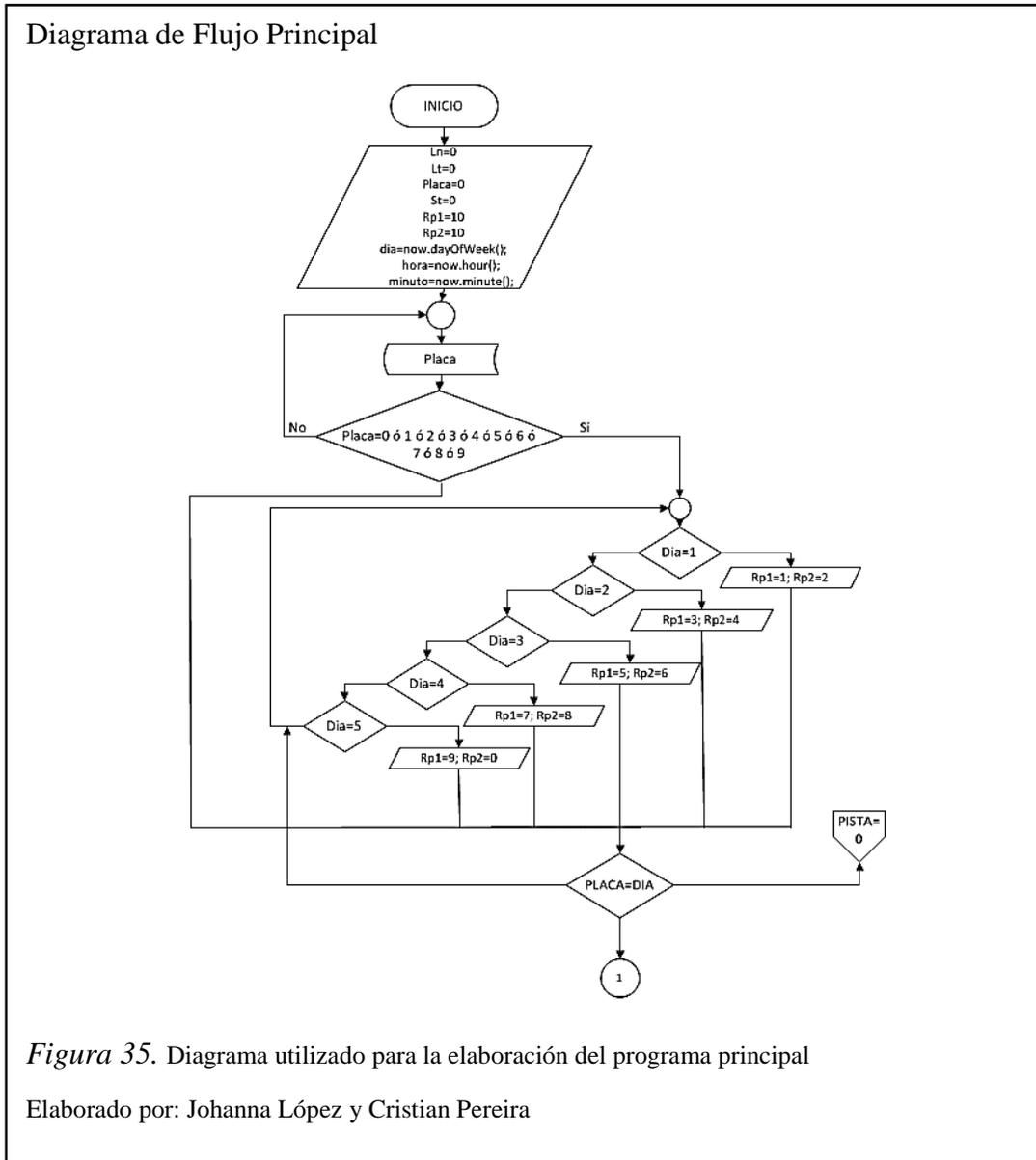
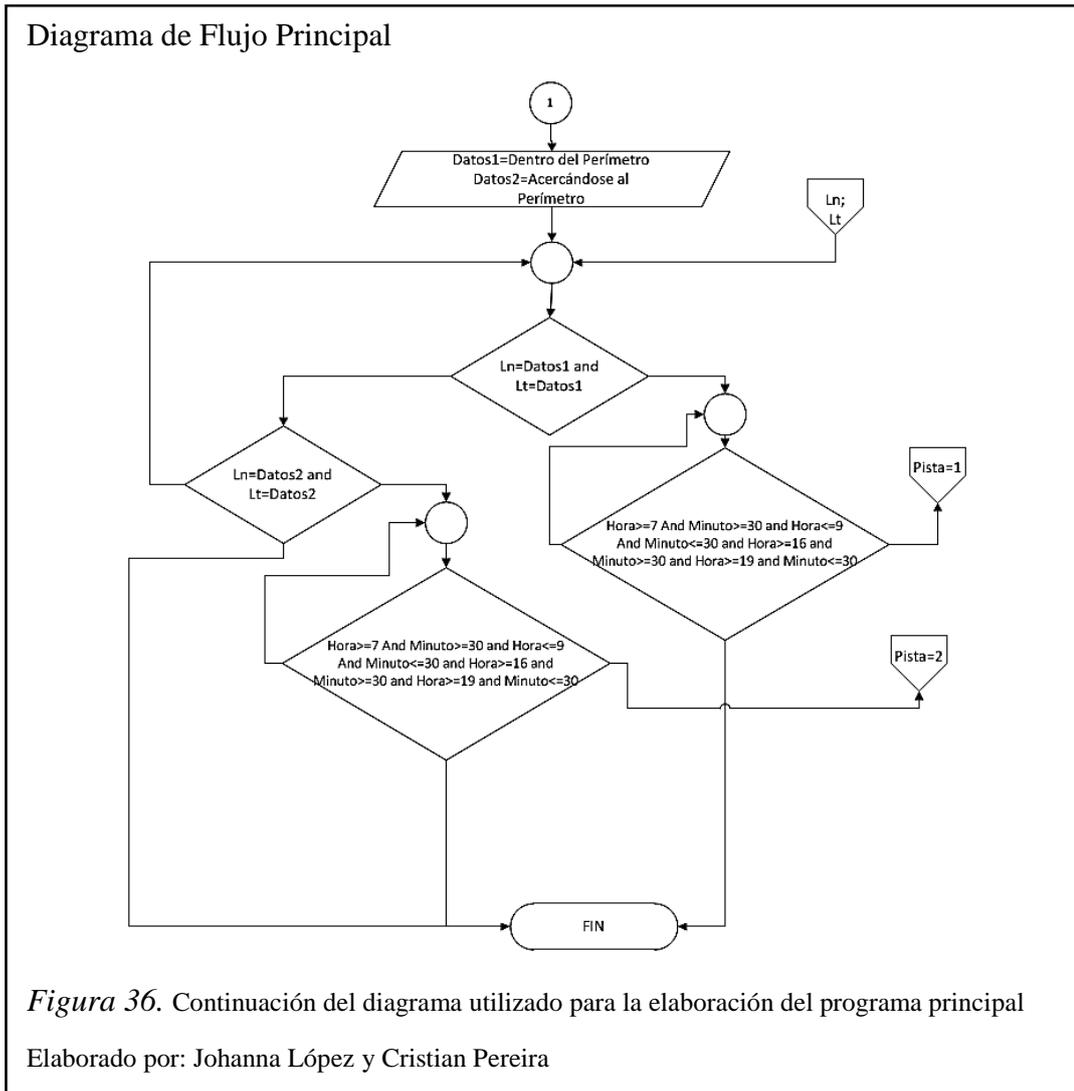


Figura 35. Diagrama utilizado para la elaboración del programa principal

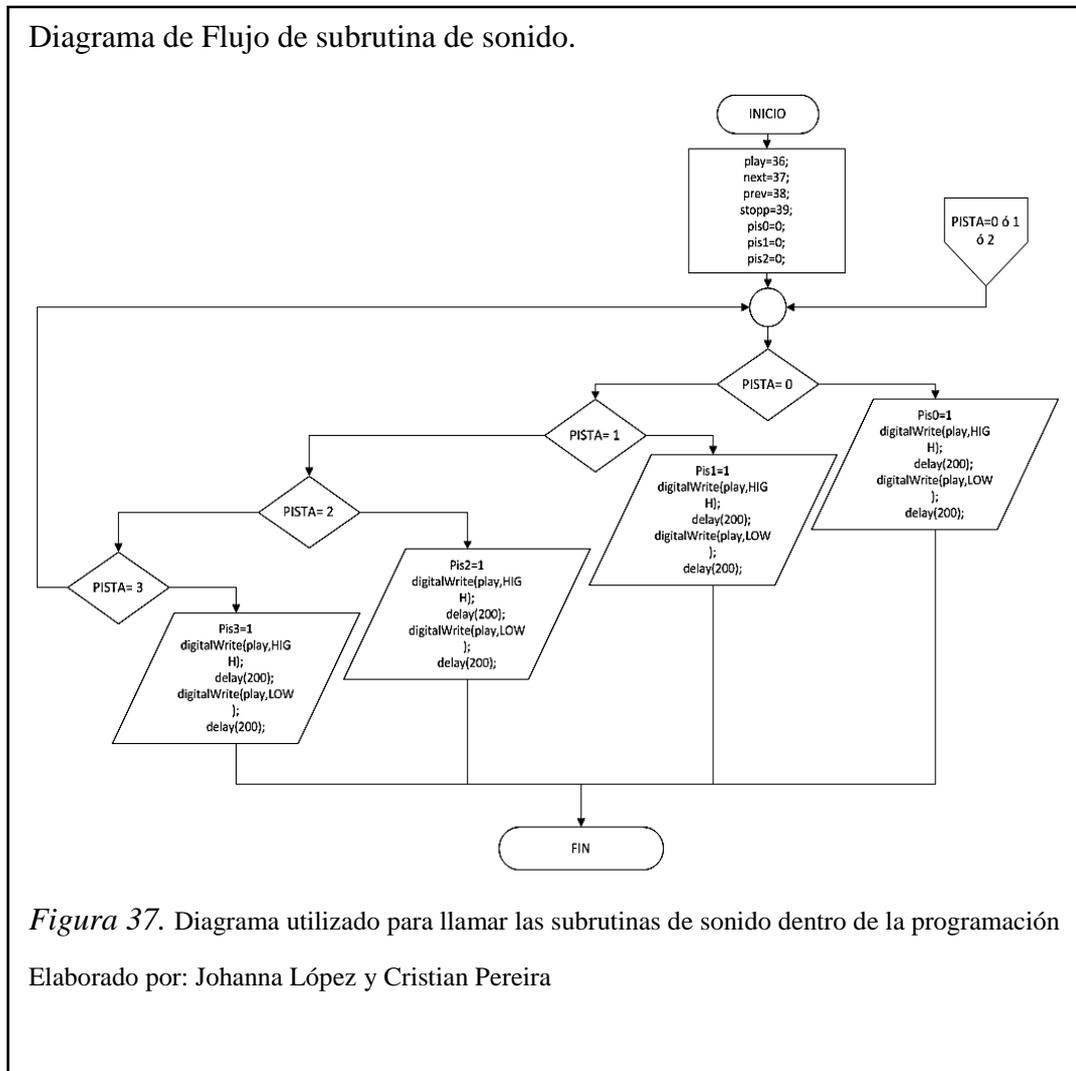
Elaborado por: Johanna López y Cristian Pereira



El diagrama de flujo mostrado en la figura 37, hace referencia a la subrutina de sonido, el cual es explicado a continuación:

1. Inicio del programa.
2. Inicialización y declaración de variables a ser utilizadas tipo int.
3. Comparación de requerimiento de pistas, en donde la variable “play” cambia el estado de low a high y la variable “pista” es comparada con los siguientes valores:
 - Si Pista= 0, se reproduce la siguiente alerta de voz: “Recuerde, hoy su vehículo tiene pico y placa”.
 - Si Pista= 1, se reproduce la siguiente alerta de voz: “Recuerde, hoy su vehículo tiene pico y placa y está dentro del perímetro”.

- Si Pista= 2, se reproduce la siguiente alerta de voz: “Atención, está acercándose al perímetro de pico y placa”.



El diagrama de flujo mostrado en la figura 38, describe la subrutina utilizada para el módulo GPS, el cual es explicado a continuación:

1. Inicio del programa.
2. Inicializar la librería del GPS.
3. Declaración e inicialización de las variables longitud (ln=0) y latitud (lt=0).
4. Adquisición de la posición del vehículo mediante el módulo GPS.
5. Escalamiento de los datos (longitud y latitud) adquiridos por el módulo GPS para un mejor procesamiento.

Diagrama de Flujo de subrutina del módulo GPS.

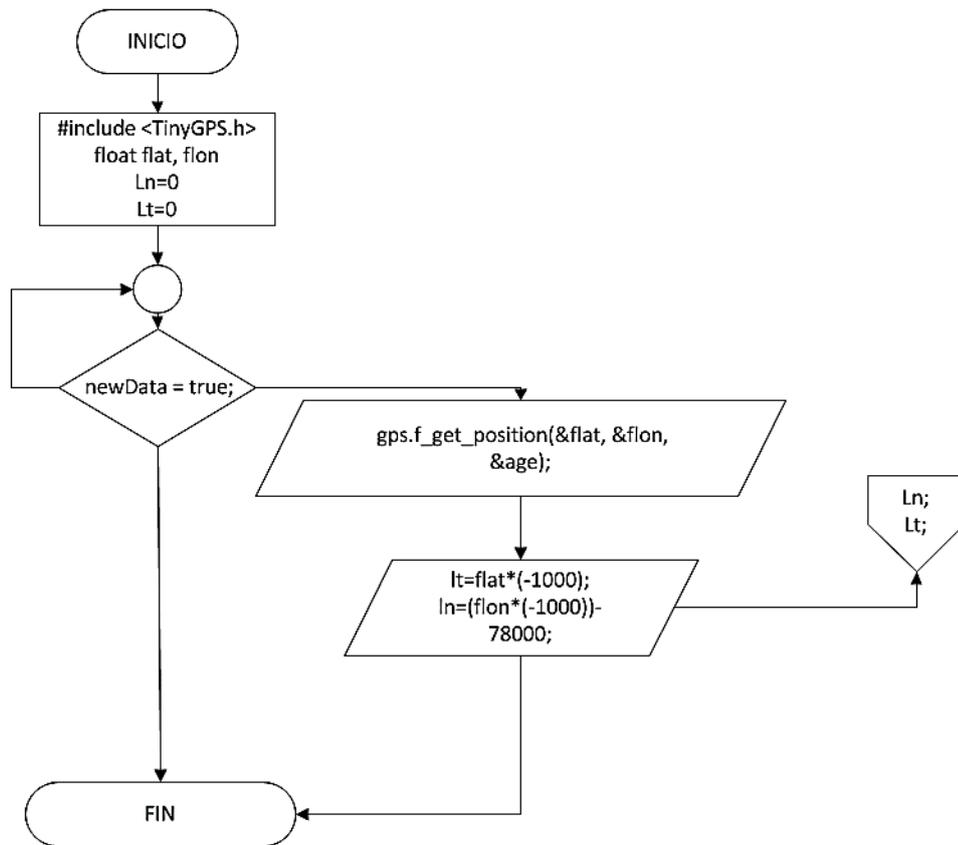


Figura 38. Diagrama utilizado para programar la adquisición de datos con el GPS

Elaborado por: Johanna López y Cristian Pereira

CAPÍTULO 5

PRUEBAS Y RESULTADOS

En este capítulo, se analizará los resultados obtenidos luego de la implementación del dispositivo y una vez realizadas las pruebas de funcionamiento del sistema de alerta para pico y placa, dentro de las pruebas realizadas se tiene: repetitividad, linealidad y calidad, las cuales serán expuestas más adelante.

5.1. Repetitividad del sistema

La repetitividad del sistema fue calculada tomando diez muestras en dos posiciones (inicial y final), estas se las realizó en un mismo lugar para obtener tanto los valores de latitud como los de longitud inicial y final, mismos que se muestran a continuación en la tabla 36.

Tabla 36.

Cálculo de repetitividad del sistema.

LATITUD INICIAL	MEDIA	LATITUD FINAL	MEDIA	LONGITUD INICIAL	MEDIA	LONGITUD FINAL	MEDIA
0,33	0,33	0,33	0,33	78,54	78,54	78,54	78,54
0,33	0,33	0,33	0,33	78,54	78,54	78,54	78,54
0,33	0,33	0,33	0,33	78,54	78,54	78,54	78,54
0,33	0,33	0,33	0,33	78,54	78,54	78,54	78,54
0,33	0,33	0,33	0,33	78,54	78,54	78,54	78,54
0,33	0,33	0,33	0,33	78,54	78,54	78,54	78,54
0,33	0,33	0,33	0,33	78,54	78,54	78,54	78,54
0,33	0,33	0,33	0,33	78,54	78,54	78,54	78,54
0,33	0,33	0,33	0,33	78,54	78,54	78,54	78,54
0,33	0,33	0,33	0,33	78,54	78,54	78,54	78,54
Repetitividad=	0,006%	Repetitividad=	0,003%	Repetitividad=	0,009%	Repetitividad=	0,009%

Nota.

Elaborado por: Johanna López y Cristian Pereira

Para encontrar la repetitividad del sistema por medio del GPS, los valores de latitud y longitud fueron tomados en las mismas posiciones, obteniendo en latitud los valores 0,006 y 0,003, mientras que en longitud 0,009.

5.2. Linealidad del sistema

En las pruebas de linealidad se trabajó con posiciones obtenidas por el módulo GPS y las entregadas por un celular, mismas que son mostradas en la tabla 37 y 38.

Tabla 37.

Coordenadas obtenidas con el módulo GPS.

Item	LATITUD	LONGITUD
1	-0,333345	-78,541765
2	-0,333547	-78,541795
3	-0,333750	-78,541831
4	-0,334018	-78,541831
5	-0,334270	-78,541925
6	-0,334450	-78,541938
7	-0,335130	-78,541954
8	-0,335427	-78,541950
9	-0,335968	-78,541961
10	-0,336528	-78,541965
11	-0,337120	-78,541975
12	-0,337682	-78,541985
13	-0,337988	-78,541985
14	-0,338200	-78,541982
15	-0,338640	-78,542026
16	-0,339355	-78,542045
17	-0,339565	-78,542095

Nota.

Elaborado por: Johanna López y Cristian Pereira

Tabla 38.

Coordenadas obtenidas desde un celular.

Item	LATITUD	LONGITUD
1	-0,333339	-78,541754
2	-0,333581	-78,541780
3	-0,333754	-78,541840
4	-0,334030	-78,541844
5	-0,334271	-78,541930
6	-0,334441	-78,541930
7	-0,335155	-78,541940
8	-0,335544	-78,541945
9	-0,335912	-78,541960
10	-0,336609	-78,541960
11	-0,337230	-78,541965
12	-0,337780	-78,541970
13	-0,337985	-78,541970
14	-0,338240	-78,541980
15	-0,338710	-78,542026
16	-0,339430	-78,542034
17	-0,339595	-78,542080

Nota.

Elaborado por: Johanna López y Cristian Pereira

Para el cálculo se utiliza la ecuación lineal detallada a continuación: $y = ax + b$

Ecuación 1.

Ecuación Lineal

$$y = mx + n$$

Donde la constante m es la pendiente de la recta y n es la ordenada en el origen, la cual indica el punto de corte de la recta con el eje de las ordenadas.

- Se procede a realizar los cálculos respectivos en función de la ecuación lineal con los datos obtenidos por el celular, de tal manera que se obtenga la linealidad del sistema.

Dados los siguientes puntos, en función de los datos obtenidos, donde a son los valores del ítem 1 en la tabla 38 y b son los valores del ítem 17 de la tabla 38:

$$a = (-0,333339; -78,541754)$$

$$b = (-0,339595; -78,54208)$$

Se reemplaza en ecuación 1 los valores tanto de y (longitud) como de x (latitud) en función de a y b , y se procede a despejar m y n hasta obtener su valor correspondiente.

$$y = mx + b$$

Ecuación 2.

Reemplazo de los valores de a

$$-78,541754 = -0,333339m + n$$

Ecuación 3.

Reemplazo de los valores de b

$$-78,542080 = -0,339595m + n$$

Se multiplica * (-1) la ecuación 2, y se realiza una suma de ecuaciones de la siguiente manera:

$$\begin{array}{r}
 +78,541754 = +0,333339m - n \\
 -78,542080 = -0,339595m + n \\
 \hline
 -0,000326 = -0,006256m + 0
 \end{array}$$

Despejando m para obtener su valor, donde: $m = 0,0521099744$

Reemplazando en la ecuación 3 el valor de m y despejando hasta obtener el valor correspondiente a n , de la siguiente manera:

$$-78,542080 = -0,339595 (0,0521099744) + n$$

$$n = -78,5243837132$$

Remplazando nuevamente los valores tanto de m como de n en la Ecuación 1, obteniendo lo indicado a continuación:

Ecuación 4.

Ecuación lineal obtenida con los datos receptados por el celular

$$y = 0,0521099744x - 78,5243837132$$

- Se realiza los cálculos respectivos utilizando la ecuación lineal con los datos obtenidos por el módulo GPS, para obtener la linealidad del sistema.

Dados los siguientes puntos, en función de los datos obtenidos, donde a son los valores del ítem 1 en la tabla 38 y b son los valores del ítem 17 de la tabla 37:

$$a = (-0,333345; -78,541765)$$

$$b = (-0,339565; -78,542095)$$

Se reemplaza en ecuación 1 los valores tanto de y (longitud) como de x (latitud) en función de a y b , y se procede a despejar m y n hasta obtener su valor correspondiente.

$$y = mx + b$$

Ecuación 5.

Remplazo de los valores de a

$$-78,541765 = -0,333345m + n$$

Ecuación 6.

Reemplazo de los valores de b

$$-78,542095 = -0,339565m + n$$

Multiplicando * (-1) la ecuación 5, y se realiza una suma de ecuaciones de la siguiente manera:

$$\begin{array}{r} +78,541765 = +0,333345m - n \\ -78,542080 = -0,339595m + n \\ \hline -0,000330 = -0,006220m + 0 \end{array}$$

Despejando m para obtener su valor, donde: $m = 0,0530546624$

Reemplazando en la ecuación 6 el valor de m y despejando hasta obtener el valor correspondiente a n , de la siguiente manera:

$$\begin{aligned} -78,542095 &= -0,339565(0,0530546624) + n \\ n &= -78,5240794936 \end{aligned}$$

Reemplazando nuevamente los valores tanto de m como de n en la Ecuación 1, obteniendo lo indicado a continuación:

Ecuación 7.

Ecuación lineal obtenida con los datos receptados por el celular

$$y = 0,0530546624x - 78,5240794936$$

Para el cálculo de la linealidad del sistema se realizó ajuste por mínimos cuadrados para encontrar la línea de tendencia y el error producido al comparar la latitud y longitud proporcionadas con el celular y el módulo GPS, obteniendo los resultados que se indican en la figura 39 y 40. A continuación, se muestra la gráfica realizada en EXCEL, una herramienta que facilita los cálculos para obtener la linealidad.

Linealidad de los datos de latitud

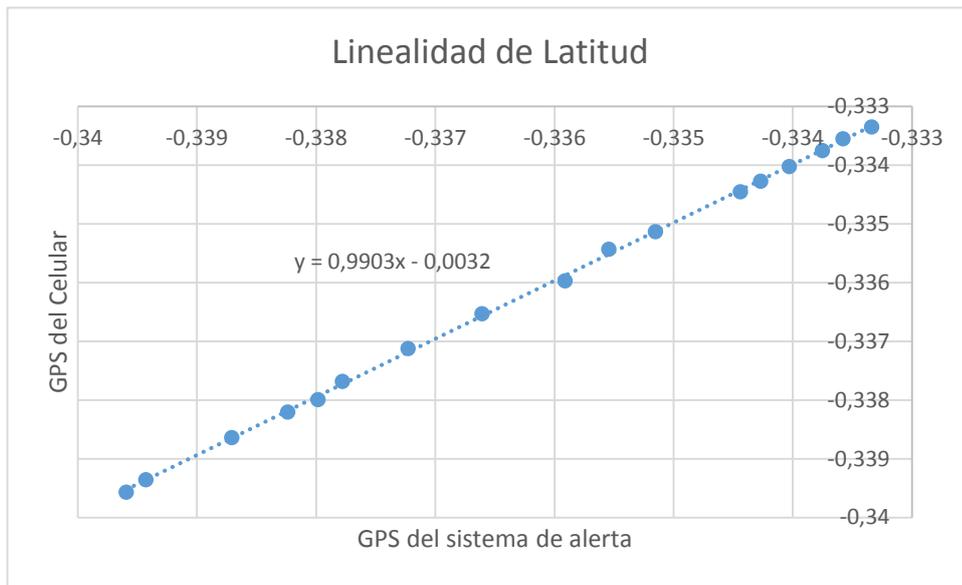


Figura 39. Grafica de la linealidad para la latitud obtenida en Excel

Elaborado por: Johanna López y Cristian Pereira

Linealidad de los datos de longitud

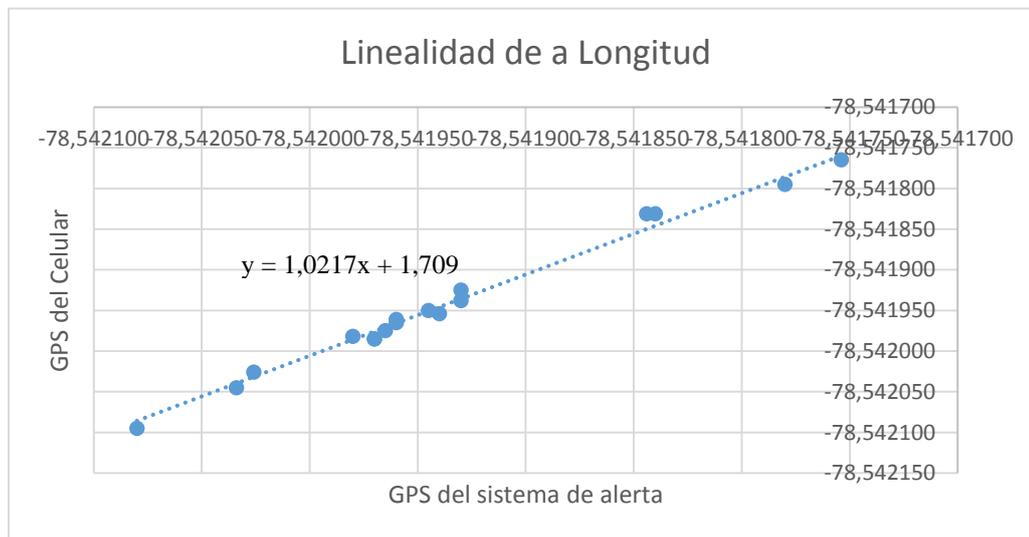


Figura 40. Grafica de la linealidad para la longitud obtenida en Excel

Elaborado por: Johanna López y Cristian Pereira

5.3. Pruebas del funcionamiento del sistema

La calidad que se asegura en este sistema es de $\pm 15\text{m}$ en los sectores no muy concurridos por los agentes metropolitanos ya que en las entradas principales se garantiza que la alerta suene siempre antes de que ingresemos en los perímetros de restricción.

Una vez ya implementado el sistema de alerta para pico y placa, fue necesario realizar varias pruebas dentro de la ciudad, donde se pudo corroborar el correcto funcionamiento del dispositivo de alerta, a continuación se muestran las pruebas realizadas en lugares estratégicos de la ciudad de Quito tomados como referencia para la redacción de este documento.

- En la primera prueba realizada se validó el funcionamiento del dispositivo al emitir la alerta de proximidad, para lo cual se tomó como referencia sectores como el puente de Guajaló (figura 41), Chillogallo (figura 42) la autopista General Rumiñahui (figura 43), el puente Oswaldo Guayasamín (figura 44) , Av. de los Granados (figura 45), Entrada a Carcelén (figura 46), Ingreso a la Av. Mariscal Sucre (figura 47), y la Av. Mitad del Mundo (figura 48), Las figuras mostradas a continuación fueron capturadas al instante que el dispositivo emitió la alerta “Atención, está acercándose al perímetro de pico y placa”, adicionalmente se adjunta la captura realizada en el mismo instante de la información enviada del sistema al computador en el momento de cada alerta.

Alertas de proximidad



a) Fotografía tomada al instante que la alerta fue emitida

```
1t2911401n538968 CHARS=124976 SENTENCES=737 CSUM ERR=95
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.291140 LON=-78.538970 SAT=11 PREC=802014/10/4 7:0:36
1t2911401n538968 CHARS=125328 SENTENCES=738 CSUM ERR=96
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.291140 LON=-78.538970 SAT=11 PREC=802014/10/4 7:0:39
1t2911401n538968 CHARS=125883 SENTENCES=741 CSUM ERR=96
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.291140 LON=-78.538970 SAT=10 PREC=1002014/10/4 7:0:43
1t2911401n538968 CHARS=126226 SENTENCES=743 CSUM ERR=96
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.291140 LON=-78.538970 SAT=11 PREC=802014/10/4 7:0:46
1t2911401n538968 CHARS=126572 SENTENCES=745 CSUM ERR=97
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.291162 LON=-78.538978 SAT=11 PREC=802014/10/4 7:0:49
1t2911621n538976 CHARS=127047 SENTENCES=746 CSUM ERR=97
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.291077 LON=-78.538940 SAT=11 PREC=802014/10/4 7:0:52
1t2910771n538944 CHARS=127403 SENTENCES=749 CSUM ERR=97
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNZONA DE PROXIMIDADLAT=-0.290282 LON=-78.538314 SAT=11 PREC=802014/10/4 7:1:12
1t2902821n538312 CHARS=127760 SENTENCES=751 CSUM ERR=98
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.290353 LON=-78.538131 SAT=11 PREC=802014/10/4 7:1:15
1t2903531n538128 CHARS=128256 SENTENCES=754 CSUM ERR=99
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.290402 LON=-78.537933 SAT=11 PREC=802014/10/4 7:1:18
1t2904021n537936 CHARS=128614 SENTENCES=756 CSUM ERR=99
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.290373 LON=-78.537750 SAT=11 PREC=802014/10/4 7:1:22
1t2903731n537752 CHARS=128975 SENTENCES=757 CSUM ERR=100
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.290287 LON=-78.537452 SAT=11 PREC=802014/10/4 7:1:25
1t2902871n537456 CHARS=129541 SENTENCES=761 CSUM ERR=101
```

b) Información enviada del sistema a la computadora al momento de cada alerta.

Figura 41. Pruebas de alertas de proximidad tomando como referencia el Puente de Guajaló.

Elaborado por: Johanna López y Cristian Pereira

En la figura 41.b, se puede observar que una vez que el vehículo se está aproximando al Puente de Guajaló, sector donde comienza la restricción de pico y placa, el dispositivo envía al computador un mensaje (únicamente para las pruebas) y al mismo tiempo la alerta se reproduce para que sea escuchada por el usuario.

Alertas de proximidad



a) Fotografía tomada al instante que la alerta fue emitida

```
lt2830731n553840 CHARS=542935 SENTENCES=2794 CSUM ERR=478
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.282922 LON=-78.554016 SAT=11 PREC=802014/10/4 8:46:8
lt2829221n554016 CHARS=543503 SENTENCES=2796 CSUM ERR=478
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.282840 LON=-78.554092 SAT=11 PREC=802014/10/4 8:46:12
lt2828401n554096 CHARS=543860 SENTENCES=2798 CSUM ERR=478
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.282817 LON=-78.554115 SAT=11 PREC=802014/10/4 8:46:15
lt2828171n554112 CHARS=544204 SENTENCES=2799 CSUM ERR=479
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.282773 LON=-78.554153 SAT=11 PREC=802014/10/4 8:46:18
lt2827731n554152 CHARS=544770 SENTENCES=2802 CSUM ERR=479
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.282640 LON=-78.554199 SAT=11 PREC=802014/10/4 8:46:22
lt2826401n554200 CHARS=545130 SENTENCES=2804 CSUM ERR=479
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.282315 LON=-78.554122 SAT=11 PREC=802014/10/4 8:46:25
lt2823151n554120 CHARS=545489 SENTENCES=2806 CSUM ERR=480
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.281980 LON=-78.553947 SAT=11 PREC=802014/10/4 8:46:28
lt2819801n553944 CHARS=545983 SENTENCES=2807 CSUM ERR=480
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.281593 LON=-78.553749 SAT=11 PREC=802014/10/4 8:46:32
lt2815931n553752 CHARS=546341 SENTENCES=2810 CSUM ERR=480
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.281092 LON=-78.553504 SAT=11 PREC=802014/10/4 8:46:35
lt2810921n553504 CHARS=546702 SENTENCES=2812 CSUM ERR=481
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNZONA DE PROXIMIDAD
```

Desplazamiento automático

b) Información enviada del sistema a la computadora al momento de cada alerta.

Figura 42. Pruebas de alertas de proximidad tomando como referencia Chillogallo.

Elaborado por: Johanna López y Cristian Pereira

En la figura 42.b, se observa el mensaje que se envía al computador desde el dispositivo (únicamente para las pruebas) cuando el vehículo se está aproximando al Puente de Chillogallo donde comienza la restricción de pico y placa, al mismo tiempo la alerta se reproduce para que sea escuchada por el usuario.

Alertas de proximidad



a) Fotografía tomada al instante que la alerta fue emitida

```
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.235015 LON=-78.488212 SAT=10 PREC=902014/10/4 7:11:16
1t2350151n488216 CHARS=201018 SENTENCES=1120 CSUM ERR=164
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.235920 LON=-78.488090 SAT=10 PREC=902014/10/4 7:11:20
1t2359201n488088 CHARS=201584 SENTENCES=1122 CSUM ERR=165
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.236473 LON=-78.487930 SAT=11 PREC=802014/10/4 7:11:23
1t2364731n487928 CHARS=201944 SENTENCES=1124 CSUM ERR=165
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.236920 LON=-78.487548 SAT=11 PREC=802014/10/4 7:11:26
1t2369201n487552 CHARS=202304 SENTENCES=1126 CSUM ERR=165
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.237177 LON=-78.486999 SAT=11 PREC=802014/10/4 7:11:30
1t2371771n487000 CHARS=202872 SENTENCES=1127 CSUM ERR=165
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNZONA DE PROXIMIDADLAT=-0.238147 LON=-78.482810 SAT=11 PREC=802014/10/4 7:11:49
1t2381471n482808 CHARS=203320 SENTENCES=1129 CSUM ERR=165
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.238663 LON=-78.482658 SAT=11 PREC=802014/10/4 7:11:52
1t2386631n482656 CHARS=203680 SENTENCES=1131 CSUM ERR=165
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.239198 LON=-78.482719 SAT=11 PREC=802014/10/4 7:11:56
1t2391981n482720 CHARS=204044 SENTENCES=1132 CSUM ERR=166
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.239912 LON=-78.482864 SAT=11 PREC=802014/10/4 7:11:59
1t2399121n482864 CHARS=204417 SENTENCES=1134 CSUM ERR=166
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.240468 LON=-78.482948 SAT=11 PREC=802014/10/4 7:12:2
1t2404681n482952 CHARS=204777 SENTENCES=1136 CSUM ERR=166
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓNLAT=-0.241062 LON=-78.483016 SAT=10 PREC=902014/10/4 7:12:6
1t2410621n483016 CHARS=205135 SENTENCES=1138 CSUM ERR=166
```

b) Información enviada del sistema a la computadora al momento de cada alerta.

Figura 43. Pruebas de alertas de proximidad tomando como referencia la Autopista General Rumiñahui

Elaborado por: Johanna López y Cristian Pereira

En la figura 43.b, se puede observar que el vehículo se está aproximando al desvío a la Autopista General Rumiñahui que permite conectarse al Norte Quito, donde se aplica la restricción de pico y placa, momento en el cual el dispositivo envía un

mensaje al computador (únicamente para las pruebas) y al mismo tiempo la alerta se reproduce para que sea escuchada por el usuario.

Alertas de proximidad



a) Fotografía tomada al instante que la alerta fue emitida

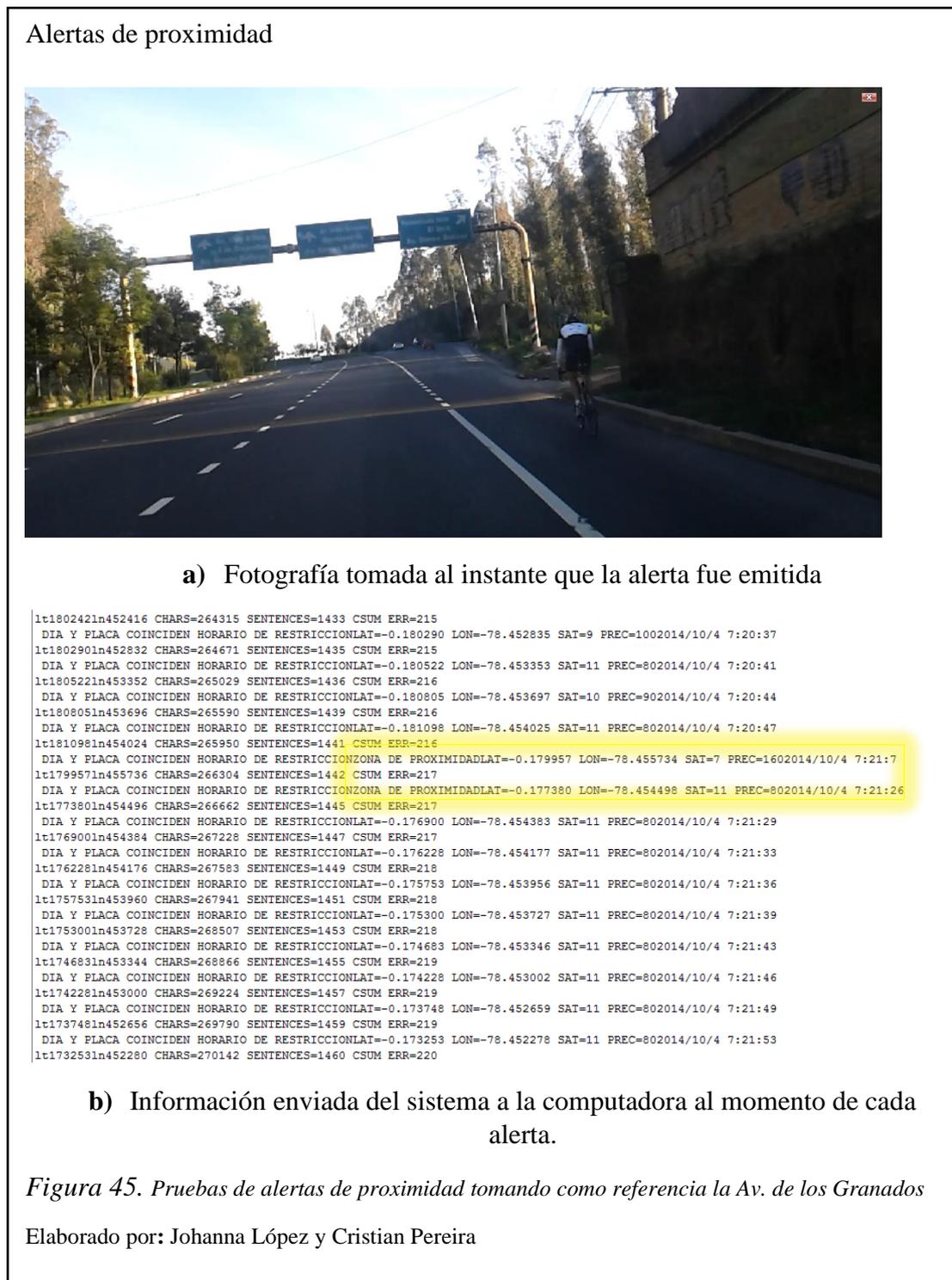
```
lt191833ln458072 CHARS=255054 SENTENCES=1388 CSUM ERR=208
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.191893 LON=-78.457473 SAT=9 PREC=1102014/10/4 7:19:5
lt191893ln457472 CHARS=255408 SENTENCES=1390 CSUM ERR=208
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.192110 LON=-78.456680 SAT=9 PREC=1102014/10/4 7:19:9
lt192110ln456680 CHARS=255768 SENTENCES=1392 CSUM ERR=209
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.192300 LON=-78.456062 SAT=9 PREC=1102014/10/4 7:19:12
lt192300ln456064 CHARS=256124 SENTENCES=1394 CSUM ERR=209
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.192428 LON=-78.455398 SAT=9 PREC=1102014/10/4 7:19:15
lt192428ln455400 CHARS=256478 SENTENCES=1396 CSUM ERR=209
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.192475 LON=-78.454605 SAT=9 PREC=1102014/10/4 7:19:19
lt192475ln454608 CHARS=256836 SENTENCES=1398 CSUM ERR=210
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.192137 LON=-78.454154 SAT=11 PREC=802014/10/4 7:19:22
lt192137ln454152 CHARS=257194 SENTENCES=1400 CSUM ERR=210
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.191572 LON=-78.454025 SAT=11 PREC=802014/10/4 7:19:25
lt191572ln454024 CHARS=257550 SENTENCES=1402 CSUM ERR=210
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.190972 LON=-78.454071 SAT=11 PREC=802014/10/4 7:19:29
lt190972ln454072 CHARS=257908 SENTENCES=1403 CSUM ERR=211
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONZONA DE PROXIMIDADLAT=-0.186738 LON=-78.453926 SAT=11 PREC=802014/10/4 7:19:48
lt186738ln453928 CHARS=258264 SENTENCES=1405 CSUM ERR=211
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.186113 LON=-78.453796 SAT=11 PREC=802014/10/4 7:19:51
lt186113ln453800 CHARS=258622 SENTENCES=1407 CSUM ERR=211
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.185585 LON=-78.453514 SAT=11 PREC=802014/10/4 7:19:54
lt185585ln453512 CHARS=259188 SENTENCES=1409 CSUM ERR=211
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.185100 LON=-78.452873 SAT=10 PREC=902014/10/4 7:19:58
lt185100ln452872 CHARS=259545 SENTENCES=1410 CSUM ERR=212
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.184893 LON=-78.452293 SAT=8 PREC=1202014/10/4 7:20:1
lt184893ln452296 CHARS=259897 SENTENCES=1413 CSUM ERR=212
```

b) Información enviada del sistema a la computadora al momento de cada alerta.

Figura 44. Pruebas de alertas de proximidad tomando como referencia el Puente Oswaldo Guayasamín.

Elaborado por: Johanna López y Cristian Pereira

En la figura 44.b, se puede observar que el vehículo se está aproximando al desvío hacia el Puente de Oswaldo Guayasamín, donde se aplica la restricción de pico y placa, momento en el cual el dispositivo envía un mensaje al computador (únicamente para las pruebas) y al mismo tiempo la alerta se reproduce para que sea escuchada por el usuario.



En la figura 45.b, se puede observar que cuando el vehículo se está aproximando hacia la Av. de los granados, donde se aplica la restricción de pico y placa, el dispositivo envía un mensaje al computador (únicamente para las pruebas) y al mismo tiempo la alerta se reproduce para que sea escuchada por el usuario.

Alertas de proximidad



a) Fotografía tomada al instante que la alerta fue emitida

```

DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.094200 LONG=-78.664462 SAT=10 PREC=902014/10/4 7:33:53
129820012664444 CARRS=336627 SENTENCIAS=0792 CSUM ERR=275
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.097845 LONG=-78.664407 SAT=10 PREC=902014/10/4 7:33:53
129784512664408 CARRS=337205 SENTENCIAS=0794 CSUM ERR=275
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.097803 LONG=-78.664305 SAT=10 PREC=902014/10/4 7:33:41
129780312664305 CARRS=337645 SENTENCIAS=0795 CSUM ERR=274
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.097115 LONG=-78.665303 SAT=10 PREC=902014/10/4 7:33:45
129711512665304 CARRS=332192 SENTENCIAS=0799 CSUM ERR=277
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.094659 LONG=-78.665342 SAT=10 PREC=902014/10/4 7:33:48
129465912665342 CARRS=335590 SENTENCIAS=0801 CSUM ERR=277
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.094630 LONG=-78.665427 SAT=10 PREC=902014/10/4 7:33:51
129463012665427 CARRS=335905 SENTENCIAS=0802 CSUM ERR=277
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.094430 LONG=-78.665360 SAT=10 PREC=902014/10/4 7:33:55
129443012665360 CARRS=339475 SENTENCIAS=0805 CSUM ERR=278
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.094352 LONG=-78.666067 SAT=9 PREC=902014/10/4 7:33:58
129435212666067 CARRS=339627 SENTENCIAS=0808 CSUM ERR=278
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.094302 LONG=-78.666278 SAT=8 PREC=1202014/10/4 7:34:01
129430212666278 CARRS=342179 SENTENCIAS=0810 CSUM ERR=278
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.094227 LONG=-78.666354 SAT=10 PREC=902014/10/4 7:34:05
129422712666352 CARRS=340737 SENTENCIAS=0812 CSUM ERR=278
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.094059 LONG=-78.666345 SAT=10 PREC=902014/10/4 7:34:08
129405912666345 CARRS=341095 SENTENCIAS=0814 CSUM ERR=279
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.094078 LONG=-78.666773 SAT=10 PREC=902014/10/4 7:34:11
129407812666774 CARRS=342455 SENTENCIAS=0814 CSUM ERR=279
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLAT=-0.093997 LONG=-78.667071 SAT=10 PREC=902014/10/4 7:34:15
129399712667072 CARRS=342000 SENTENCIAS=0819 CSUM ERR=280
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLONGA DE RESTRICCIONLAT=-0.093937 LONG=-78.667422 SAT=10 PREC=902014/10/4 7:34:24
129393712667424 CARRS=342489 SENTENCIAS=0819 CSUM ERR=280
DIA Y PLACA COINCIDE HORARIO DE RESTRICCIONLONGA DE RESTRICCIONLAT=-0.093898 LONG=-78.668095 SAT=10 PREC=902014/10/4 7:34:33
129389812668094 CARRS=342947 SENTENCIAS=0822 CSUM ERR=280

```

b) Información enviada del sistema a la computadora al momento de cada alerta.

Figura 46. Pruebas de alertas de proximidad tomando como referencia la entrada a Carcelén.
Elaborado por: Johanna López y Cristian Pereira

En la figura 46.b, se puede observar que el vehículo se está aproximando a la entrada a Carcelén, donde se aplica la restricción de pico y placa, momento en el cual el dispositivo envía un mensaje al computador (únicamente para las pruebas) y al mismo tiempo la alerta se reproduce para que sea escuchada por el usuario.

Alertas de proximidad



a) Fotografía tomada al instante que la alerta fue emitida

```

DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.091397 LONG=-78.4731908 EAX=10 FREC=902014/10/4 7:35:42
11921971247294 CARRAS=347080 SENTENCIAS=1242 CSOM ERR=123
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.091310 LONG=-78.473472 EAX=9 FREC=902014/10/4 7:35:44
11921971247242 CARRAS=347414 SENTENCIAS=1243 CSOM ERR=124
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.090922 LONG=-78.473363 EAX=10 FREC=902014/10/4 7:35:49
11921971247282 CARRAS=347955 SENTENCIAS=1245 CSOM ERR=124
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.090782 LONG=-78.474487 EAX=10 FREC=902014/10/4 7:35:52
11907812474489 CARRAS=348293 SENTENCIAS=1249 CSOM ERR=124
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.090739 LONG=-78.473181 EAX=10 FREC=902014/10/4 7:35:54
119073912473182 CARRAS=348449 SENTENCIAS=1250 CSOM ERR=125
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.090782 LONG=-78.473424 EAX=9 FREC=902014/10/4 7:35:55
11907812474424 CARRAS=349099 SENTENCIAS=1251 CSOM ERR=125
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.090908 LONG=-78.474097 EAX=9 FREC=902014/10/4 7:36:2
119090812474094 CARRAS=348455 SENTENCIAS=1253 CSOM ERR=125
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.091118 LONG=-78.474870 EAX=9 FREC=902014/10/4 7:36:18
1190111812474848 CARRAS=349615 SENTENCIAS=1254 CSOM ERR=124
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.091518 LONG=-78.477187 EAX=10 FREC=902014/10/4 7:36:19
1190151812477180 CARRAS=350128 SENTENCIAS=1256 CSOM ERR=124
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.091512 LONG=-78.477384 EAX=9 FREC=902014/10/4 7:36:12
1190151212477384 CARRAS=350539 SENTENCIAS=1258 CSOM ERR=124
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.092088 LONG=-78.477973 EAX=10 FREC=902014/10/4 7:36:13
119208812477974 CARRAS=350297 SENTENCIAS=1260 CSOM ERR=124
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.092449 LONG=-78.478492 EAX=10 FREC=902014/10/4 7:36:19
119244912478496 CARRAS=351255 SENTENCIAS=1261 CSOM ERR=127
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.092718 LONG=-78.478874 EAX=10 FREC=902014/10/4 7:36:22
119071812478872 CARRAS=351613 SENTENCIAS=1263 CSOM ERR=127
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.092947 LONG=-78.479252 EAX=10 FREC=902014/10/4 7:36:23
119294712479200 CARRAS=351871 SENTENCIAS=1265 CSOM ERR=127
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.093298 LONG=-78.479437 EAX=10 FREC=902014/10/4 7:36:29
11932981247943 CARRAS=352330 SENTENCIAS=1267 CSOM ERR=128
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.093448 LONG=-78.480020 EAX=10 FREC=902014/10/4 7:36:32
119344812480000 CARRAS=352688 SENTENCIAS=1269 CSOM ERR=128
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLONGA DE PROXIMIDADLAT=-0.094322 LONG=-78.482025 EAX=10 FREC=902014/10/4 7:36:13
119432212482024 CARRAS=353046 SENTENCIAS=1271 CSOM ERR=128

```

b) Información enviada del sistema a la computadora al momento de cada alerta.

Figura 47. Pruebas de alertas de proximidad tomando como referencia el ingreso a la Av. Mariscal Sucre.

Elaborado por: Johanna López y Cristian Pereira

En la figura 47.b, se puede observar que el vehículo está circulando en la Magdalena y se está aproximando a la Av. Mariscal Sucre donde se tiene la restricción de pico y placa, por tanto el dispositivo envía un mensaje al computador (únicamente para las pruebas) y al mismo tiempo la alerta se reproduce para que sea escuchada por el usuario.

Alertas de proximidad



a) Fotografía tomada al instante que la alerta fue emitida

```

DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.101893 LONG=-78.488642 SAT=9 FREC=902014/10/4 7:37:48
1a101893a64864 CARRS=384880 SENTENCS=2478 CDM ERR=289
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.102348 LONG=-78.487322 SAT=10 FREC=902014/10/4 7:37:52
1a102348a64732 CARRS=385038 SENTENCS=2881 CDM ERR=289
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.102688 LONG=-78.487493 SAT=10 FREC=902014/10/4 7:37:55
1a102688a64749 CARRS=385594 SENTENCS=2883 CDM ERR=289
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.102482 LONG=-78.488075 SAT=10 FREC=902014/10/4 7:37:58
1a102482a64807 CARRS=385755 SENTENCS=2884 CDM ERR=290
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.102433 LONG=-78.488325 SAT=10 FREC=902014/10/4 7:38:12
1a102433a64832 CARRS=386113 SENTENCS=2887 CDM ERR=290
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.101925 LONG=-78.490463 SAT=10 FREC=902014/10/4 7:38:20
1a101925a64946 CARRS=386662 SENTENCS=2890 CDM ERR=291
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.099297 LONG=-78.490097 SAT=10 FREC=902014/10/4 7:38:28
1a099297a49096 CARRS=387019 SENTENCS=2892 CDM ERR=292
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.099918 LONG=-78.489410 SAT=10 FREC=902014/10/4 7:38:37
1a099918a48440 CARRS=387374 SENTENCS=2894 CDM ERR=292
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.099910 LONG=-78.489501 SAT=10 FREC=902014/10/4 7:38:38
1a099910a49504 CARRS=387728 SENTENCS=2894 CDM ERR=293
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.097092 LONG=-78.488842 SAT=10 FREC=902014/10/4 7:38:40
1a097092a48840 CARRS=388297 SENTENCS=2899 CDM ERR=294
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.097358 LONG=-78.489439 SAT=10 FREC=902014/10/4 7:38:42
1a097358a48840 CARRS=388445 SENTENCS=2901 CDM ERR=294
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.097485 LONG=-78.489738 SAT=10 FREC=902014/10/4 7:38:46
1a097485a49738 CARRS=389005 SENTENCS=2903 CDM ERR=294
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.099282 LONG=-78.489913 SAT=10 FREC=902014/10/4 7:38:50
1a099282a48993 CARRS=389549 SENTENCS=2908 CDM ERR=295
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.098705 LONG=-78.490081 SAT=9 FREC=1002014/10/4 7:38:53
1a098705a49081 CARRS=389825 SENTENCS=2908 CDM ERR=295
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.099095 LONG=-78.490288 SAT=10 FREC=902014/10/4 7:38:56
1a099095a49092 CARRS=340253 SENTENCS=2910 CDM ERR=295
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.099873 LONG=-78.490394 SAT=10 FREC=902014/10/4 7:38:40
1a099873a49092 CARRS=340245 SENTENCS=2912 CDM ERR=294
DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIÓN DOMA DE RESTRICCIÓNLAT=-0.100355 LONG=-78.490339 SAT=10 FREC=902014/10/4 7:38:43
1a100355a49039 CARRS=341103 SENTENCS=2914 CDM ERR=294

```

b) Información enviada del sistema a la computadora al momento de cada alerta.

Figura 48. Pruebas de alertas de proximidad tomando como referencia la Av. Mitad del Mundo.
Elaborado por: Johanna López y Cristian Pereira

En la figura 48.b, se puede observar que el vehículo está circulando desde la Mitad del Mundo y se está aproximando a la Av. Mariscal Sucre, el dispositivo envía un mensaje al computador (únicamente para las pruebas) y al mismo tiempo la alerta se reproduce para que sea escuchada por el usuario.

- En la segunda prueba realizada se validó el funcionamiento del dispositivo al emitir las alertas que indican estar dentro del perímetro de restricción, para lo cual se tomó como referencia diferentes puntos como la Av. El Inca (Figura 49), Av. Mariscal Sucre (Figura 50), las figuras mostradas a continuación fueron capturadas al instante que el dispositivo emitió la alerta “Recuerde, hoy su vehículo tiene pico y placa y está dentro del perímetro”, adicionalmente se adjunta la captura realizada en el mismo instante de la información enviada del sistema al computador en el momento de cada alerta.

Alertas de restricción



a) Fotografía tomada al instante que la alerta fue emitida

```
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.164735 LON=-78.496437 SAT=9 PREC=100014/10/4 7:47:52
1215675910494440 CMAS=393714 SENTENCIAS=2074 CSOM ERS=331
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.164432 LON=-78.496307 SAT=10 PREC=902014/10/4 7:47:55
1215941210494304 CMAS=394072 SENTENCIAS=2074 CSOM ERS=331
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.160113 LON=-78.496162 SAT=10 PREC=902014/10/4 7:47:58
1214011310494360 CMAS=394430 SENTENCIAS=2074 CSOM ERS=331
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.161027 LON=-78.495966 SAT=10 PREC=902014/10/4 7:48:02
1214102710494360 CMAS=394745 SENTENCIAS=2074 CSOM ERS=332
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.164898 LON=-78.495790 SAT=9 PREC=100014/10/4 7:48:05
1214489810495752 CMAS=395340 SENTENCIAS=2080 CSOM ERS=333
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.167748 LON=-78.496443 SAT=9 PREC=110014/10/4 7:48:08
1214774810494440 CMAS=395890 SENTENCIAS=2081 CSOM ERS=334
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.168247 LON=-78.496670 SAT=9 PREC=110014/10/4 7:48:11
1214824710496972 CMAS=396014 SENTENCIAS=2085 CSOM ERS=334
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.168848 LON=-78.496314 SAT=10 PREC=902014/10/4 7:48:14
1214884810496914 CMAS=396418 SENTENCIAS=2087 CSOM ERS=334
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.169738 LON=-78.496572 SAT=9 PREC=902014/10/4 7:48:17
1214973810496974 CMAS=396973 SENTENCIAS=2088 CSOM ERS=335
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.170435 LON=-78.496614 SAT=10 PREC=902014/10/4 7:48:20
1217043510496914 CMAS=397381 SENTENCIAS=2090 CSOM ERS=335
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.171137 LON=-78.500093 SAT=10 PREC=902014/10/4 7:48:24
1217113710490098 CMAS=397594 SENTENCIAS=2092 CSOM ERS=335
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCIONLAT=-0.172047 LON=-78.500381 SAT=9 PREC=902014/10/4 7:48:27
1217204710490094 CMAS=398240 SENTENCIAS=2094 CSOM ERS=334
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.172140 LON=-78.500427 SAT=10 PREC=902014/10/4 7:48:31
1217214010490014 CMAS=398618 SENTENCIAS=2094 CSOM ERS=334
...DIA Y PLACA COINCIDEN HORARIO DE RESTRICCION ZONA DE RESTRICCIONLAT=-0.174857 LON=-78.502047 SAT=10 PREC=902014/10/4 7:48:39
1217485710492044 CMAS=399098 SENTENCIAS=2097 CSOM ERS=334
```

b) Información enviada del sistema a la computadora al momento de cada alerta.

Figura 50. Pruebas de alertas de restricción tomando como referencia la Av. Mariscal Sucre.

Elaborado por: Johanna López y Cristian Pereira

En la figura 50.b, se puede observar que el vehículo está circulando en la Av. Mariscal Sucre, el dispositivo envía un mensaje al computador (únicamente para las pruebas) y al mismo tiempo la alerta se reproduce para que sea escuchada por el usuario indicando que se encuentra dentro del perímetro donde rige esta norma.

CONCLUSIONES

- La distancia a la cual el dispositivo procesa y emite las alertas depende del número de satélites a los que se conecte el módulo GPS, ya que depende mucho de las coordenadas (longitud y latitud) que obtenga el dispositivo para mejorar su precisión el valor estimado es de 15 metros, pero el valor podría oscilar entre 20 a 25 m cuando tenga conexión con menos satélites.
- Los valores de longitud varían en función del movimiento de vehículo, es decir de sur a norte los valores aumentan y en sentido contrario disminuyen, en el caso de la latitud si el vehículo viaja de este a oeste estos disminuyen y en sentidos contrarios estos aumentan.
- Se pudo verificar y observar que el procesamiento de valores en las autopistas se ve afectado por la velocidad, dado que esta se encuentra en el rango de 90 – 100 Km/h lo cual provoca un retardo de 30s en la obtención de coordenadas.
- La variación en los datos entregados por el sistema de alerta implementado para este proyecto es mínima, ya que de acuerdo a los valores arrojados en las pruebas de repetitividad realizadas oscilan entre un 0,006 y 0,003 para la latitud y 0,009 para la longitud, siendo menores al valor considerado como repetitividad máxima que es el de $\pm 0.01\%$.
- Se observó que de acuerdo a los valores obtenidos con el módulo GPS tanto en latitud como en longitud comparados con los obtenidos en el celular la variación es mínima, en longitud varia un 0,004% y en latitud en un 0,001%.
- Una vez comprobada la confiabilidad y eficacia de este prototipo es recomendable su fabricación a gran escala la única limitante seria su alto costo de distribución en el mercado, sin embargo no deja de ser un excelente dispositivo que permitirá prevenir a los conductores infringir contra la normativa del pico y placa.

REFERENCIAS

- (s.f.). Obtenido de www.juntadeandalucia.es/:
http://www.juntadeandalucia.es/averroes/albora/Departamentos/Informatica/3_digitalizacin_del_sonido.html
- <http://www.reflexiona.biz/>. (2010). Obtenido de
<http://www.reflexiona.biz/shop/content/10-guia-de-compra-de-arduino>
- <http://www.malditonerd.com>. (16 de 01 de 2011). Obtenido de
<http://www.malditonerd.com/howto-extender-un-cable-usb-por-utp-usb-a-rj45/>
- Barrera Guarderas, A. (2010). ORDENANZA METROPOLITANA No. 0305. *ORDENANZA METROPOLITANA REFORMATORIA DE LA SECCIÓN IV*, 13. Quito, Pichincha, Ecuador.
- Barrera Guarderas, A. (2010). Resolución N° 0017. En *Libro I del Código Municipal para el Distrito Metropolitano de Quito* (pág. 13). Quito.
- Cano Infantes, D. (20 de Mayo de 2012). <http://www.slideshare.net/>. Obtenido de Introducción a los Microcontroladores PIC:
<http://www.slideshare.net/soharca/microcontroladores-pic-13005280>
- Cerda Sánchez , D. A., & Pazmiño Días , I. P. (2011). <http://repositorio.espe.edu.ec/>. Obtenido de DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA CON GPS Y CONTROL DE SEGURIDAD VEHICULAR CON COMUNICACIÓN GSM: <http://repositorio.espe.edu.ec/bitstream/21000/3784/1/T-ESPEL-0800.pdf>
- Clavijo Mendoza, J. R. (2011). *Diseño y simulación de sistemas microcontrolados en lenguaje C*. Colombia: Colombia.
- consulvial-ecuador. (02 de 04 de 2013). *CONSULTORA EN EDUCACION Y SEGURIDAD VIAL DEL ECUADOR*. Obtenido de <http://consulvial-ecuador.blogspot.com/>: <http://consulvial-ecuador.blogspot.com/2013/04/horarios-de-pico-y-placa-en-quito-2013.html>
- Cuartero, F. (05 de Junio de 2012). www.hablandodeciencia.com. Obtenido de <http://www.hablandodeciencia.com/articulos/2012/06/05/el-sistema-de-posicionamiento-global-gps/>
- ecuador-vial.com. (Mayo de 2011). www.ecuador-vial.com. Obtenido de <http://www.ecuador-vial.com/quito-con-nuevas-multas-por-pico-y-placa>
- Facultad de Ciencias, U. (s.f.). <http://www.fciencias.uaslp.mx/>. Obtenido de http://galia.fc.uaslp.mx/~cantocar/microcontroladores/EL_BUS_I2C/TRANSPARENCIASI2C.PDF
- Flores Cortez, O. O. (Septiembre de 2009). <http://microcontroladores2utec.wordpress.com/>. Obtenido de BATALLA DE MICROCONTROLADORES ¿AVR o PIC?:

http://microcontroladores2utec.files.wordpress.com/2009/11/180909_articulo_colaboracion_boletin_fica_omar_otoniel_flores.pdf

Gambino, F. M. (2012). *Programación de microcontroladores AVR Nivel inicial*. San Miguel de Tucumán, Argentina: GambrinoGroup.com.ar.

Gómez Gutiérrez, E. (17 de septiembre de 2009). <http://www.dtic.upf.edu/>. Obtenido de Digitalización del Sonido:
<http://www.dtic.upf.edu/~egomez/teaching/sintesi/SPS1/Tema2-Digitalizacion.pdf>

<http://www.element14.com/>. (s.f.). Obtenido de element14 community:
<http://www.element14.com/community/community/raspberry-pi/raspberry-pi-accessories/gertduino/blog/2014/01/14/getting-up-and-running-with-the-gertduino>

Instituto Tecnológico Superior de Calkiní. (2014). <http://www.itescam.edu.mx/>. Obtenido de
<http://www.itescam.edu.mx/principal/sylabus/fpdb/recursos/r102293.PDF>

Jeff. (19 de 07 de 2010). <http://libooks.net/>. Obtenido de LiBooks:
<http://libooks.net/read/ac1893c525719cd45d5a4ba3f862e91a9b7c4617>

Jelsoft Enterprises Ltd. . (24 de Agosto de 2003). <http://www.todopocketpc.com/>. Obtenido de <http://www.todopocketpc.com/software-de-navegacion-en-carretera-y-pna/chuleta-para-interpretar-las-sentencias-nmea-20117.html>

juanpch. (07 de 06 de 2013). *foros ecuador.ec*. Obtenido de <http://www.forosecuador.ec/forum/aficiones/autos-y-motos/1974-patios-de-retenci%C3%B3n-vehicular-en-quito>

López Martín, A. (s.f.). <http://www.analfatecnicos.net/>. Obtenido de Manual para Radialistas Analfatécnicos:
<http://www.analfatecnicos.net/archivos/32.FormatosDeAudioDigital.pdf>

Mailto. (s.f.). <http://www.arcelect.com/>. Obtenido de
<http://www.arcelect.com/rs232.htm>

Martín Pernía, A. (s.f.). <http://www.unioviedo.es/>. Obtenido de MANUAL DE MICROCONTROLADORES PIC:
<http://www.unioviedo.es/ate/alberto/manualPic.pdf>

MikroElektronika. (1998). <http://www.mikroe.com/>. Obtenido de
<http://burgath.com/wp-content/uploads/2012/06/Manual-RTC-datasheet.pdf>

Oña Caisatoa, D. G., & Viteri Tigse, H. P. (Enero de 2013). <http://repositorio.espe.edu.ec/>. Obtenido de DISEÑO, IMPLEMENTACIÓN DE UN SISTEMA DE MONITOREO PARA EL VEHÍCULO MAZDA BT-50 DE LA ESCUELA DE CONDUCCIÓN DE LA ESPE-L:
<http://repositorio.espe.edu.ec/bitstream/21000/6376/1/T-ESPEL-CDT-0992.pdf>

- Pazmiño Pineda , D. N. (Septiembre de 2012). Obtenido de Diseño e Implementación de un Sistema de Control de Ruta Vehicular Mediante Comparación con el uso de GPS :
<http://186.42.96.211:8080/jspui/bitstream/123456789/156/1/SISTEMA%20CONTROL%20DE%20RUTA%20VEHICULAR%20MEDIANTE%20COMPARACION%20CON%20EL%20USO%20DE%20GPS.pdf>
- Quito, C. M. (2010). Reglamento a la ordenanza 305 pico y placa. *Reglamento a la ordenanza 305 pico y placa* (pág. 13). Quito: Quito.
- Quito, P. M. (s.f.). <http://policiametropolitanaquito.gob.ec/>. Obtenido de <http://policiametropolitanaquito.gob.ec/ESTADISTICAS/estadisticas2013/PICO%20Y%20PLACA%202011.pdf>
- Reyes Abonce, S. (09 de 09 de 2009). <http://itzamna.bnct.ipn.mx/>. Obtenido de SISTEMA DE POSICIONAMIENTO GLOBAL GPS APLICADO AL CONTROL DE MAQUINARIA PESADA:
http://itzamna.bnct.ipn.mx:8080/bitstream/handle/123456789/4937/409_SISTEMA%20DE%20POSICIONAMIENTO%20GLOBAL%20GPS%20APLICADO%20AL%20CONTROL%20DE%20MAQUINARIA%20PESADA.pdf?sequence=1
- Sandoval, F. (30 de 05 de 2014). Municipio de Quito reduce a un 50% las multas de Pico y Placa y zona azul. *EL TELÉGRAFO*.
- Universidad de Sevilla. (Febrero de 2011). <http://www.forpas.us.es/aula/>. Obtenido de http://www.forpas.us.es/aula/hardware/dia2_USB.pdf
- Zurdo, D. (24 de 12 de 2012). *El sistema GPS y sus competidores*. Obtenido de <http://www.acta.es/>:
http://www.acta.es/medios/articulos/ciencias_y_tecnologia/038055.pdf

ANEXOS

Anexo 1.

Detalle de las sentencias NMEA en función de la tabla 26

Sentencias	Interpretación	
GPGGA	hhmmss.ss	Hora UTM referente a la toma del dato (horas, minutos y segundos)
	llll.ll	Latitud en grados y minutos
	A	Tipo de Latitud (N=norte, S=sur)
	mmmmmm.mm	Longitud en grados y minutos
	b	Tipo de longitud (E=este, W=oeste)
	x	Calidad de la señal GPS (0=no válida, 1= corrección GPS, 2= DCPS)
	yy	Número de satélites en uso
	z.z	HDOP
	c.c	Altitud sobre el nivel del mar
	M	Unidad de medida de la altitud (metros)
	v.v	Distancia entre el elipsoide WGS-84 y el nivel del mar
	M	Unidad Unidades de separación (metros)
	d.d	Tiempo desde la última actualización DGPS (segundos)
	wwwwww	Identificación de la estación DGPS
*hh	Checksum	
GPGSA	x	Modo de operación 2d/3d (A=automático, M>manual)
	y	Modo de corrección de datos (1=no disponible, 2=2D, 3=3D)
	z1-z12	Identificación PRN de los satélites usados para corrección (Max. 12)
	p.p	PDOP
	h.h	HDOP
	v.v	VDOP
	*ff	Checksum
GPGSV	x	Número total de mensajes de este tipo en este ciclo
	y	Número de mensajes con respecto al total
	zz	Número total de satélites a la vista
	aa	Identificación PRN del satélite
	ee	Elevación en grados (Max. 90)
	rr	Azimut en grados (desde el norte verdadero, Max. 359,9)

	ii	Intensidad de la señal en decibelios (Max. 99)
	s1	Identificación PRN del 2° satélite
	s2	Elevación en grados (Max. 90) del 2° satélite
	s3	Azimut en grados (desde el norte verdadero, Max. 359,9) del 2° satélite
	s4	Intensidad de la señal en decibelios (Max. 99)del 2° satélite
	t1	Identificación PRN del 3° satélite
	t2	Elevación en grados (Max. 90) del 3° satélite
	t3	Azimut en grados (desde el norte verdadero, Max. 359,9) del 3° satélite
	t4	Intensidad de la señal en decibelios (Max. 99)del 3° satélite
	c1	Identificación PRN del 4° satélite
	c2	Elevación en grados (Max. 90) del 4° satélite
	c3	Azimut en grados (desde el norte verdadero, Max. 359,9) del 4° satélite
	c4	Intensidad de la señal en decibelios (Max. 99)del 4° satélite
	*ff	Checksum
GPGLL	llll.ll	Latitud en grados y minutos
	a	Tipo de Latitud (N=norte, S=sur)
	mmmm.mm	Longitud en grados y minutos
	b	Tipo de longitud (E=este, W=oeste)
	hhmmss.ss	Hora UTM referente a la toma del dato (horas, minutos y segundos)
	A	Validez del dato (A=válido, V=no válido) del 4° satélite
	*ff	Checksum

Anexo 2.

Arduino GPS Shield



Arduino GPS Shield

-A high performance GPS shield for Arduino

Overview



Arduino GPS shield is a GPS module breadout board designed for Global Positioning System receiver with SD interface. It is easy to use for recording the position data into SD card. 5V/3.3V compatible operation voltage level make it compatible with Arduino boards, leaf maple, IFlat32 and other arduino compatible boards.

Features

- With Micro SD interface
- Active antenna design with high receive sensitivity, compatible normal antenna
- Extremely fast time to first fix at low signal level
- UART interface
- Operation temperature: $-40^{\circ}\text{C} \sim +85^{\circ}\text{C}$

Specifications

PCB size	55.88mm X 54.1mm X 1.6mm
Indicators	PWR,
Power supply	compatible with Arduino
Communication Protocol	UART
RoHS	Yes

Electrical Characteristics

Specification	Min	Type	Max	Unit
Power Voltage	4.5	5	5.5	VDC
Input Voltage VH:	3	-	5.5	V
Input Voltage VL:	-0.3	0	0.5	V
Baud rate		38400		bps

Hardware

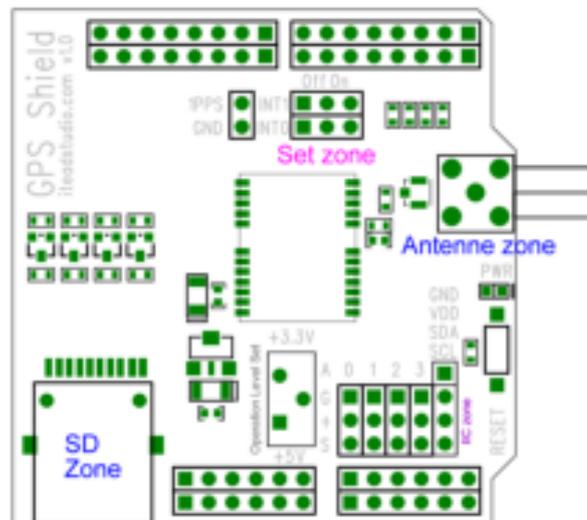


Figure 1 Top Map

Anexo 3.

Arduino Mega 2560

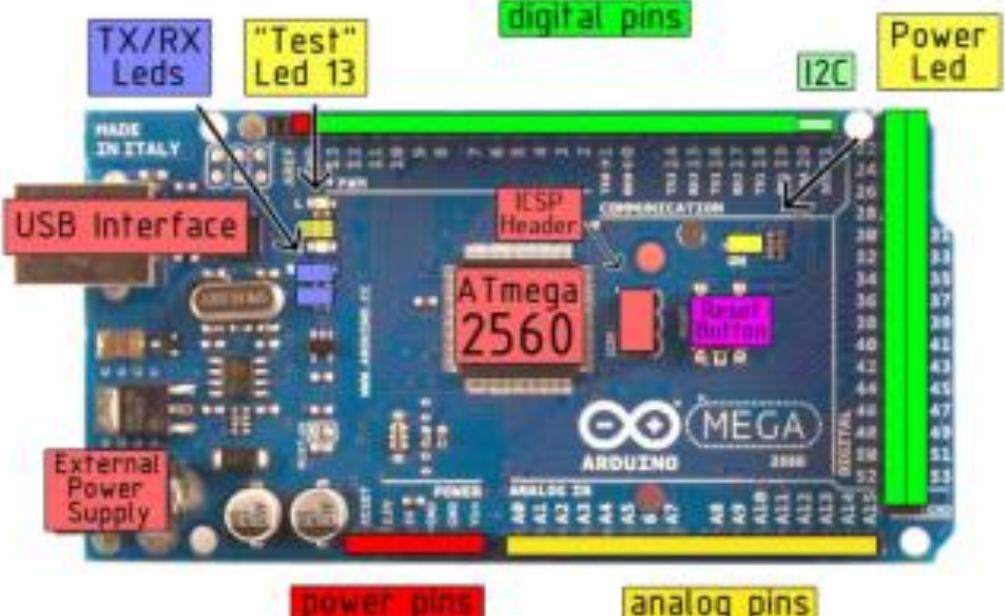
Technical Specification

EAGLE files: [arduino-mega2560-reference-design.zip](#) Schematic: [arduino-mega2560-schematic.pdf](#)

Summary

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	54 (of which 14 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz

the board



The diagram shows the Arduino Mega 2560 board with the following labels:

- TX/RX Leds**: Located at the top left.
- "Test" Led 13**: Located at the top left, next to the TX/RX LEDs.
- digital pins**: A green bar along the top edge of the board.
- Power Led**: A yellow bar along the top right edge of the board.
- I2C**: A green bar along the top right edge of the board.
- USB Interface**: A red box pointing to the USB Type-B connector.
- External Power Supply**: A red box pointing to the DC power jack.
- ATmega 2560**: The central microcontroller chip.
- ICSP Header**: A red box pointing to the In-Circuit Serial Programming header.
- power pins**: A red bar along the bottom edge of the board.
- analog pins**: A yellow bar along the bottom edge of the board.



radiospares

RADIONICS



Anexo 4.

ATMega 2560

ATmega640/1280/1281/2560/2561

1. Pin Configurations

Figure 1-1. TQFP-pinout ATmega640/1280/2560

