



UNIVERSIDAD TÉCNICA DE AMBATO

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

**CARRERA DE INGENIERÍA EN ELECTRÓNICA Y
COMUNICACIONES**

Tema:

“Sistema Electrónico por comando de voz para la seguridad física, iluminación y accionamiento automático de los accesorios primarios de un automóvil.”

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

Línea de investigación: Sistemas Electrónicos

Autor: Edwin Ismael Torres Valverde

Profesor revisor: Ing. Santiago Mauricio Altamirano Meléndez, Mg.

Ambato – Ecuador

Diciembre 2015

APROBACIÓN DEL TUTOR

En mi calidad de tutor del trabajo de investigación sobre el tema: “**Sistema Electrónico por comando de voz para la seguridad física, iluminación y accionamiento automático de los accesorios primarios de un automóvil**”, del señor Edwin Ismael Torres Valverde, estudiante de la Carrera de Ingeniería en Electrónica y Comunicaciones, de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial, de la Universidad Técnica de Ambato, considero que el informe investigativo reúne los requisitos suficientes para que continúe con los trámites y consiguiente aprobación de conformidad con el Art. 16 del Capítulo II, del Reglamento de Graduación para obtener el Título Terminal de Tercer Nivel de la Universidad Técnica de Ambato.

Ambato, Diciembre del 2015

Ing. Santiago Mauricio Altamirano Meléndez, Mg.

TUTOR

AUTORÍA

El presente trabajo de investigación titulado: “**Sistema Electrónico por comando de voz para la seguridad física, iluminación y accionamiento automático de los accesorios primarios de un automóvil**”, es absolutamente original, auténtico y personal, en tal virtud, el contenido, efectos legales y académicos que se desprenden del mismo son de exclusiva responsabilidad del autor.

Ambato, Diciembre del 2015

Edwin Ismael Torres Valverde

C.C: 0503171811

APROBACIÓN DEL TRIBUNAL DE GRADO

La Comisión Calificadora del presente trabajo conformado por los señores docentes Ing. Geovanni Danilo Brito Moncayo, Mg. e Ing. Mario Geovanni García Carrillo M. Sc., revisó y aprobó el informe final del trabajo de graduación titulado **“Sistema Electrónico por comando de voz para la seguridad física, iluminación y accionamiento automático de los accesorios primarios de un automóvil”**, presentado por el señor Edwin Ismael Torres Valverde de acuerdo al Art. 17 del Reglamento de Graduación para obtener el Título Terminal de Tercer Nivel de la Universidad Técnica de Ambato.

Ing. José Vicente Morales Lozada, Mg.
PRESIDENTE DE TRIBUNAL

Ing. Geovanni Danilo Brito Moncayo, Mg.
DOCENTE CALIFICADOR

Ing. Mario Geovanni García Carrillo M. Sc.
DOCENTE CALIFICADOR

DEDICATORIA

Al ser maravilloso que me dio la vida, mujer luchadora que con su esfuerzo supo salir adelante tras cualquier adversidad, mujer que con su actitud firme forjó mi destino encarrilándome siempre en el camino del bien, mujer que con su ejemplo me enseñó la diferencia entre lo bueno y lo malo, mujer que con su aliento levanta mis ánimos y me motiva para continuar y culminar mis metas, a usted Madre mía, por ser quien eres y gracias a usted soy quien soy.

Ismael Torres Valverde

AGRADECIMIENTO

Primeramente a Dios por la oportunidad de vivir, ya que sin su presencia espiritual en nosotros seríamos simples seres sin ningún propósito en la vida.

A mi Familia que con su apoyo constante infundieron la avidez necesaria para lograr mi superación personal y emocional para llegar a ser un verdadero ser humano.

A los Docentes que en todo el transcurso de mi vida estudiantil dentro de esta noble institución, compartieron sus conocimientos, convirtiéndome en un buen profesional, capacitado para desarrollar de la mejor manera en el hábito laboral.

A demás Personas y Amigos que estuvieron conmigo en los buenos y malos momentos, apoyándonos incondicionalmente, llegando a formar parte de mi vida personal por su sincero cariño y afecto.

Ismael Torres Valverde

ÍNDICE DE CONTENIDOS

APROBACIÓN DEL TUTOR.....	1
AUTORÍA.....	ii
APROBACIÓN DEL TRIBUNAL DE GRADO	iii
DEDICATORIA	iv
AGRADECIMIENTO	v
ÍNDICE DE CONTENIDOS	vi
ÍNDICE DE FIGURAS.....	ix
ÍNDICE DE TABLAS	xii
RESUMEN.....	xiii
ABSTRAC	xiv
GLOSARIO DE TÉRMINOS.....	xv
INTRODUCCIÓN	xvii
CAPÍTULO I.....	1
EL PROBLEMA	1
1.1. TEMA DE INVESTIGACIÓN	1
1.2. PLANTEAMIENTO DEL PROBLEMA.....	1
1.3. DELIMITACIÓN.....	2
1.4. JUSTIFICACIÓN	3
1.5. OBJETIVOS	4
1.5.1. OBJETIVO GENERAL	4
1.5.2. OBJETIVOS ESPECÍFICOS.....	4
CAPÍTULO II	5
MARCO TEÓRICO.....	5
2.1. ANTECEDENTES INVESTIGATIVOS.....	5
2.2. FUNDAMENTACIÓN TEÓRICA.....	7
2.2.1. SISTEMA DE SEGURIDAD VEHICULAR	7
2.2.2. SISTEMAS Y ACCESORIOS DEL VEHÍCULO.....	9
2.2.3. RECONOCIMIENTO DE VOZ	16
2.2.4. MICROCONTROLADOR	30

2.2.5.	ARDUINO	31
2.2.6.	PROPUESTA DE SOLUCIÓN	34
CAPÍTULO III.....		35
METODOLOGÍA		35
3.1.	MODALIDAD DE LA INVESTIGACIÓN	35
3.2.	RECOLECCIÓN DE LA INFORMACIÓN	36
3.3.	PROCESAMIENTO Y ANÁLISIS DE DATOS.....	36
3.4.	DESARROLLO DEL PROYECTO	37
CAPÍTULO IV.....		38
DESARROLLO DE LA PROPUESTA.....		38
4.1.	SISTEMAS DE RECONOCIMIENTO DE VOZ PARA VEHÍCULOS	39
4.1.1.	App Drivestyle Digital	39
4.1.2.	Pioneer Zypr.....	40
4.1.3.	Ford Sinc	41
4.1.4.	Android Auto	42
4.2.	ANÁLISIS DE FACTIBILIDAD	43
4.2.1.	Factibilidad Técnica	43
4.2.2.	Factibilidad Operativa	44
4.2.3.	Factibilidad Económica.....	44
4.3.	SELECCIÓN DISPOSITIVOS ELECTRÓNICOS	44
4.3.1.	Arduino Mega 2560	44
4.3.2.	EasyVR SHIELD 2.0	47
4.3.3.	Pantalla GLCD 12864ZW	49
4.3.4.	Componentes Electrónicos.....	53
4.3.5.	Cálculo Componentes Electrónicos	58
4.4.	SOFTWARE	61
4.4.1.	Arduino IDE.....	61
4.4.2.	Balabolka	62
4.4.3.	EasyVR Commander.....	63
4.4.4.	Sensory QuickSynthesis 5.....	64
4.4.5.	Proteus.....	65
4.5.	ESTRUCTURA	66

4.5.1.	Módulo Central	67
4.5.2.	Módulo de Control	68
4.5.3.	Interfaz de Usuario.....	69
4.6.	CARACTERÍSTICAS DEL SISTEMA.....	70
4.7.	PROGRAMACIÓN	71
4.8.	COMANDOS DEL SISTEMA	75
4.9.	SIMULACIONES	77
4.10.	DISEÑO DE PLACAS ELECTRÓNICAS.....	83
4.11.	MONTAJE	87
4.12.	PRUEBAS DE FUNCIONAMIENTO	90
4.13.	ANÁLISIS FINAL DE COSTOS	93
CAPÍTULO V		96
CONCLUSIONES Y RECOMENDACIONES.....		96
5.1.	CONCLUSIONES	96
5.2.	RECOMENDACIONES	97
BIBLIOGRAFÍA		98
ANEXOS		103

ÍNDICE DE FIGURAS

Figura 2.1: Sistema de iluminación general de un vehículo	10
Figura 2.2: Esquema del Sistema eléctrico de iluminación	12
Figura 2.3: Ejemplo general de un sistema de alarma.....	13
Figura 2.4: Proceso general entrenamiento, reconocimiento de la voz.....	19
Figura 2.5: Señal de voz sonorizada	20
Figura 2.6: Señal de voz no sonorizada	20
Figura 2.7: Interpretación de la voz del modelo Source-Filter.....	21
Figura 2.8: Ejemplo de una señal analógica separada en bloques.....	22
Figura 2.9: Señal senoidal aplicada ventana rectangular	23
Figura 2.10: Transformada de Fourier de señal analógica	23
Figura 2.11: Gráfica función ventana de Hamming	24
Figura 2.12: Señal analógica aplicada ventana de Hamming.....	24
Figura 2.13: DTF de la señal resultante al utilizar ventana de Hamming	25
Figura 2.14: Espectro de una señal vocal tratada con ventana de Hamming	26
Figura 2.15: Espectro logarítmico de la señal resultante al utilizar ventana de Hamming	26
Figura 2.16: Procesamiento de la señal analógica a Cepstrum	27
Figura 2.17: Parte real de una señal Cepstrum.....	27
Figura 2.18: Esquema básico de un microcontrolador.....	30
Figura 2.19: Introducción al mundo Arduino	31
Figura 4.1: Aplicación SIRI de Apple.....	39
Figura 4.2: Aplicación Pionner	40
Figura 4.3: Sistema Sinc3 de Ford	41
Figura 4.4: Aplicación Android Auto	42
Figura 4.5: Hardware Arduino Mega2560	45
Figura 4.6: hardware EasyVR SHIELD 2.0	47
Figura 4.7: Pantalla GLCD 12864ZW	50
Figura 4.8: Pines LCD grafico 12864ZW	51
Figura 4.9: Teclado matricial 4x4 configuración interna.....	52
Figura 4.10: Teclado matricial 4x4 estructura externa.....	53

Figura 4.11: Altavoz de 8 ohm.....	53
Figura 4.12: Estructura física de una resistencia eléctrica	54
Figura 4.13: Diodo led	54
Figura 4.14: Relevador 12 VDC 10A	55
Figura 4.15: Diodo 1N4007	56
Figura 4.16: Estructura física Optoacoplador 4N25.....	56
Figura 4.17: Transistor 2N3904	57
Figura 4.18: Circuito regulador de voltaje	58
Figura 4.19: Circuito de potencia básico.....	59
Figura 4.20: Editor de código Arduino IDE.....	62
Figura 4.21: Editor de texto Balabolka Software.....	63
Figura 4.22: Software EasyVR Commander.....	64
Figura 4.23: Cuadro de diálogo Sensory QuickSynthesis 5.....	65
Figura 4.24: Portada Software Proteus.....	66
Figura 4.25: Esquema gráfico del prototipo electrónico	66
Figura 4.26: Diagrama del módulo central.....	67
Figura 4.27: Lista de comandos EasyVR	68
Figura 4.28: Diagrama del módulo de control	68
Figura 4.29: Lista de reproducción SoundTable	70
Figura 4.30: Diagrama de flujo Prototipo Electrónico.....	74
Figura 4.31: Circuito del Sistema Electrónico	78
Figura 4.32: ISIS Circuito A	79
Figura 4.33: ISIS Circuito B	80
Figura 4.34: ARES Placa A	81
Figura 4.35: ARES Placa B.....	82
Figura 4.36: Visualización 3D placa A, Módulo de Control	83
Figura 4.37: Corrosión cobre placas electrónicas	84
Figura 4.38: Perforación placas electrónicas.....	84
Figura 4.39: Placas electrónicas del Módulo de Control	85
Figura 4.40: Prototipo electrónico pruebas iniciales.....	85
Figura 4.41: Prototipo Electrónico por reconocimiento de voz	86
Figura 4.42: Desmontaje del tablero del vehículo.....	87

Figura 4.43: Ubicación de las conexiones eléctricas en el vehículo	88
Figura 4.44: Conectores socket de cableado eléctrico y prototipo.....	89
Figura 4.45: Conexión Prototipo Electrónico	89
Figura 4.46: Final Prototipo Electrónico por reconocimiento de voz	90
Figura 4.47: Prueba de funcionamiento circuito básico Módulo de Control	91
Figura 4.48: Prueba de funcionamiento Prototipo Electrónico	91

ÍNDICE DE TABLAS

Tabla 4.1: Datos técnicos Arduino Mega2560.....	45
Tabla 4.2: Características módulos más eficientes del mercado	46
Tabla 4.3: Cuadro comparativo módulos reconocimiento voz	49
Tabla 4.4: Descripción de pines de pantalla GLCD.....	51
Tabla 4.5: Comandos programados en el EasyVR.....	75
Tabla 4.6: Pruebas de comunicación Prototipo Electrónico	92
Tabla 4.7: Lista de elementos electrónicos	94
Tabla 4.8: Análisis final de inversión	95

RESUMEN

La Electrónica se ha destacado a más del alcance tecnológico por el bienestar personal; ejemplo de ello es la creación de Sistemas Electrónicos para personas especiales con alguna discapacidad física. En el reglamento de la Ley Orgánica de Tránsito, Art.133 tienen derecho de obtener certificado de conducción las personas discapacitadas, pero no pueden acceder al mismo al no poseer sus autos modificados de acuerdo a su discapacidad.

En este proyecto se ha desarrollado un Prototipo Electrónico de asistencia de conducción, para el control del sistema de iluminación, ejecución de dispositivos y un sistema de seguridad; todo controlado por reconocimiento de voz. La implementación del Prototipo permitió al usuario volver a conducir un vehículo de forma segura disminuyendo las distracciones por accionamiento manual, resolviendo los problemas ocasionados por su discapacidad física.

Los resultados fueron favorables diseñando un Prototipo Electrónico accesible, fácil de controlar, de interés colectivo, empleado tanto para personas especiales o no, igualando sus oportunidades, mejorando su estilo de vida.

Palabras claves: Sistema Embebido, Arduino, Discapacidad Física, Comando de voz.

ABSTRAC

The Electronics has outstanding scope technological for the personal development; example is the creation of Electronic Systems for special people with any physical disabilities. In the regulation of Organic Law of Traffic, Art.133 has right to obtain driving certificate disabled people, but can't access for not possessing their cars modified according to their disability.

This project has developed a prototype electronic of assistance for driving, for the control of the lighting system and execution devices, a security system; all controlled by voice recognition. Prototype implementation allowed the user to return to drive a vehicle safely reducing distractions by manually operated, solving the problems caused by their physical disabilities.

The results were favorable designing a Prototype Electronic accessible, easy to control, of collective interest, used both for special people or not, matching their opportunities, improving their life style.

Weywords: Embedded System, Arduino, Physical Disability, Voice Command.

GLOSARIO DE TÉRMINOS

ARES: (Advanced Routing and Editing Software) Software de Edición y Ruteo Avanzado.

Basic Stand: Soporte básico.

Bit: Acrónimo de dígito binario, unidad básica de información.

Byte: Conjunto ordenado de 8 bits.

Bootloader: Gestor de arranque.

Cepstrum: Transformada de Fourier del logaritmo del espectro de una señal.

DDRAM: (Double DRAM) Doble memoria dinámica de acceso randómico.

DFT: (Discrete Fourier Transform) Transformada Discreta de Fourier.

DOC, HTML, PDF: Formatos de archivo de texto.

EEPROM: (Electrically Erasable Programmable Read Only) Memoria de solo lectura electricamente borrable.

FFT: (Fast Fourier Transform) Transformada Rápida de Fourier.

Fourier: Proceso matemático de transformación de una señal.

Glottis: Abertura superior de la laringe.

GLCD: (Graphic Liquid Crystal Display) Pantalla grafica de cristal líquido.

GND: Tierra

GPS: (Global positioning System) Sistema Posicional Global.

GRAM: (Graphic RAM) Memoria gráfica de acceso randómico.

Hamming: Proceso matemático para tratar a una señal.

HMM: (Hidden Markov Model) Modelo Oculito de Markov.

Hz: Hercio

IDE: (Integrated Development Enviroment) Entorno de Desarrollo Integrado.

ISIS: (Intelligent Schematic Input System) Sistema de Enrutamiento de Esquemas Inteligentes.

KEYLESS: Sin llave.

log: Logaritmo

MCU: Unidad central de memoria.

MPU: (Microprocessor Principal Unit) Unidad Principal del Microprocesador.

NPN: Tipo juntura de un transistor.

PICs: (Programmable Interrupt Controller) Controlador Programable de Interrupciones.

Pitch: Tono

Processing: Lenguaje de programación Arduino.

PWM: (Pulse Width Modulation) Modulación por ancho de pulso.

RAM: (Random Access Memories) Memoria de Acceso Randómico.

Shield: Placa

Sketches: Esquema

SRAM: (Static Random Access Memory) Memoria Estática de Acceso Randómico.

UARTs: (Universal Asynchronous Receiver-Transmitter) Transmisor- Receptor Asíncrono Universal.

μC, UC: Microcontrolador

USB: (Universal Serial Bus) Bus Universal en Serie.

VCC: Alimentación

VSM: (Virtual System Modeling) Modelamiento Virtual del Sistema.

WAP, MP3, MP4, OGG o WMA, WAV: Formatos de archivo de audio.

Wiring: Plataforma Arduino.

.hex: archivo en formato hexadecimal.

INTRODUCCIÓN

En el presente proyecto se implementa un Prototipo Electrónico por Reconocimiento de voz, para el control del Sistema de iluminación de un vehículo y Accionamiento de ventanillas, limpiaparabrisas, calefacción sin la intervención manual, todos ejecutados por comandos de voz emitidos por parte del usuario.

Adicionalmente se desarrolló un Sistema de Seguridad, que aprovechó las características únicas de la voz para diseñar el sistema que puede ser activado o desactivado únicamente por el dueño del vehículo. Se ha tomado en cuenta circunstancias adversas que pueden afectar la tonalidad del habla, motivo por el cual se creó un sistema auxiliar de control del sistema de seguridad a través del ingreso de una contraseña.

Se dispone de una interfaz para la mejor comprensión del usuario que consiste en una pantalla GLCD para visualizar el proceso del sistema, un altavoz que emite sonidos de alerta y frases de información previamente gravados en la programación del prototipo.

El Primer capítulo se enfoca al análisis del proyecto, donde trata sobre la situación actual del problema y resalta la importancia del diseño e implementación del mismo. El Segundo capítulo presenta antecedentes sobre la realización de proyectos similares, además de información de los sistemas intervenidos del vehículo, fundamentos técnicos y descripción de dispositivos. El Tercer capítulo describe la metodología utilizada en el proyecto, los procesos de desarrollo del mismo y el análisis de recolección de la información. El Cuarto capítulo explica el proceso del diseño del Prototipo Electrónico por reconocimiento de voz. El Quinto capítulo contiene las conclusiones y recomendaciones obtenidas durante el diseño del Sistema Electrónico.

CAPÍTULO I

EL PROBLEMA

1.1. TEMA DE INVESTIGACIÓN

“Sistema Electrónico por comando de voz para la seguridad física, iluminación y accionamiento automático de los accesorios primarios de un automóvil.”

1.2. PLANTEAMIENTO DEL PROBLEMA

En la actualidad la humanidad se encuentra en la capacidad de resolver problemas presentes en su diario vivir, disponiendo de la tecnología y conocimiento necesario para cumplir dichos propósitos movidos por el deseo de mejorar su estilo de vida; pero el problema nace e influye directamente en un porcentaje de la población que al poseer algún tipo de limitación económica y mucho más importante física se ven truncadas sus expectativas de alcanzar aquel nivel de confort.

En los últimos años el desarrollo de la Electrónica aplicada a la Automotriz ha tomado auge en Ecuador, siendo lo más relevante sistemas de alarmas, con el objetivo de mejorar la seguridad por el temor a ser víctimas de la delincuencia, evitando pérdidas económicas; pero tales sistemas poseen un mínimo de seguridad encontrándose obsoletos y vulnerables, desperdiciando el potencial que ofrece esta área.

Empresas automotoras se enfatizan en optimizar el control del vehículo, diseñando de tal manera que el panel de mando este totalmente al alcance del piloto; pero aun con todas estas distribuciones, no es suficiente llegar a este objetivo ya que ciertas acciones le obligan al chofer realizar maniobras de riesgo, como soltar las manos del volante y más peligroso retirar la vista del frente, que en últimas instancias pueden ocasionar accidentes de tránsito.

El diseño de automóviles especiales para personas con discapacidades físicas, es un problema más para las automotoras, ya que al existir diferentes tipos de lesiones y discapacidades, no es posible la estandarización de los mismos; donde la opción más factible, es el acoplamiento de acuerdo a la necesidad del usuario.

En el último Censo realizado el 2014 se estableció cerca de 382.670 personas con alguna discapacidad física, y a través de la Ley Orgánica de Tránsito, en su reglamento de licencias no profesionales, gran porcentaje de los mismos no pueden acceder a su certificado al no poseer sus automotores adaptados de acuerdo a su discapacidad. [34]

1.3. DELIMITACIÓN

Área: Física y Electrónica

Línea de Investigación: Sistemas Electrónicos

Sublínea de Investigación: Sistemas Embebidos

Delimitación Espacial: El Sistema Electrónico por comando de voz para la seguridad física, iluminación y accionamiento automático de los accesorios primarios de un automóvil se implementó en un vehículo liviano, adaptándolo para la conducción de una persona con capacidad especial en una de sus extremidades superiores.

Delimitación Temporal: El proyecto se realizó en 6 meses a partir de la aprobación del Honorable Consejo Directivo de la Facultad de Ingeniería en Sistemas, Electrónica e Industrial desde el 27 de febrero del 2015 hasta el 27 de agosto del 2015.

1.4. JUSTIFICACIÓN

El mercado encabezado por empresas que desarrollan tecnología con el único propósito de generar capital, no han tomado en cuenta la necesidad de cubrir las limitaciones presentes en personas con capacidades especiales, direccionándonos al ámbito Automotriz que no desarrollan automóviles para personas con algún tipo de discapacidad física. Es aquí donde nace el interés por el desarrollo de un sistema embebido que pueda adaptarse y asistir a personas con alguna precariedad en extremidades superiores.

La importancia de la presente investigación se enfatiza en la asistencia al chofer al momento de la conducción del vehículo, que permitió automatizar acciones de ejecución manual y controlarlas por la voz, de esta manera el piloto no tiene la necesidad de soltar el volante o de quitar la mirada de la carretera para realizar algún cambio de luces, activar ventilación, limpia parabrisas o abrir y cerrar ventanillas; disminuyendo de esta manera la presión al momento de conducir un automóvil y el riesgo de accidente por desconcentración.

El impacto social fue positivo ya que vivimos en una época en donde toda persona tiene las mismas oportunidades y gracias al desarrollo tecnológico se puede cubrir ciertas desventajas que posee una persona con algún impedimento físico ante alguien normal; no obstante los beneficiarios son ambos ya que la eficiencia y funcionalidad generan comodidad y mejora el estilo de vida.

El desarrollo del proyecto fue factible tanto en la parte intelectual, técnica y económica, al disponer de todos los componentes necesarios para el diseño y ensamblaje con una gran variedad de alternativas en el mercado de acuerdo a las necesidades con precios módicos y convenientes además del conocimiento necesario para crear el prototipo electrónico.

1.5. OBJETIVOS

1.5.1. OBJETIVO GENERAL

- Diseñar un Prototipo Electrónico por comando de voz para la seguridad física, iluminación y accionamiento automático de los accesorios primarios de un automóvil.

1.5.2. OBJETIVOS ESPECÍFICOS

- Investigar los Sistemas de Seguridad y su funcionabilidad en el vehículo.
- Analizar las características del dispositivo de reconocimiento de voz.
- Realizar el Módulo de Control para la ejecución de accesorios del vehículo.
- Implementar el Prototipo Electrónico por comando de voz.

CAPÍTULO II

MARCO TEÓRICO

2.1. ANTECEDENTES INVESTIGATIVOS

Óscar Gómez Casado en su proyecto de maestría señala “El proyecto realizado se centra en un entorno importante para la sociedad, como es el tema de la seguridad. La seguridad vista desde el punto de evitar posibles accidentes de tráfico, como consecuencia del consumo excesivo de alcohol, y la seguridad desde el punto de vista de las agresiones físicas a conductores y los atentados a la propiedad privada.” [1]

Del trabajo se concluye que la utilización del reconocimiento de voz implementado en el vehículo eleva el nivel de seguridad del mismo y además resalta la responsabilidad del conductor frente a accidentes, ya que un sistema biométrico es muy confiable pero a la vez requiere que sea exacto, por los rasgos únicos del tono de la voz humana que fácilmente pueden ser alterados por ingesta de alguna sustancia estupefaciente o alcohol que le impediría el activar del vehículo.

Alex Danilo Ruiz Dalgo, Adrián Daniel Rueda Guiracocha en su proyecto de grado argumentan “El proyecto trata sobre el diseño e implementación de un control electrónico automatizado de un sistema de alumbrado de un vehículo con la finalidad de ofrecer una visibilidad óptima al usuario.” [2]

Se resalta la importancia de la percepción visual al momento de conducir el vehículo y la concentración del conductor detrás del volante, que conjuntamente son factores que aumentan el nivel de cansancio en el piloto, que en circunstancias adversas puede ocasionar accidentes; por ello la relevancia del sistema de iluminación en el vehículo, tanto en luces delanteras, traseras, de emergencia y auxiliares.

Fredy Orlando Cando Santo, Alex Mauricio Tipán Suárez en su proyecto de grado señalan “Las aplicaciones electrónicas van avanzando y es imprescindible la ayuda de este sistema para optimizar de diferente manera la utilización de los diferentes accesorios y sistemas para la mejor conducción del automóvil en la ciudad y sus alrededores ya que el mismo mantendrá la concentración del conductor.” [3]

Se concluye la utilidad de la Electrónica en el área Automotriz para mejorar cualquier proceso que conforme el vehículo, creando sistemas de asistencia al chofer, sean éstos para la seguridad, estabilidad del vehículo, iluminación y confort; automatizando con el propósito de disminuir la activación manual de dispositivos, permitiendo de esta manera al chofer mantener su atención en la conducción y evitando distracciones innecesarios que pueden terminar en colisiones.

En un artículo sobre el desarrollo de la industria Automotriz, tratan sobre la historia del uso de sistemas de reconocimiento de ordenes mediante comando de voz, el cual ha estado presente de forma experimental desde hace décadas, aunque hasta los años noventa su utilización no comenzó a plantearse de forma masiva en coches. Cuando lo hizo, fue gracias a diferentes fabricantes, como el chip desarrollado por Oki Telecomm, que permitía controlar con la voz un número limitado de órdenes sencillas para manejar un teléfono móvil.

A mediados de la primera década del 2000, empresas como la japonesa Asahi Kasei desarrollaron diferentes tecnologías para el reconocimiento de voz que fueron incorporadas por los fabricantes de vehículos y también por fabricantes de navegadores GPS y plataformas de entretenimiento. En 2005, el Ford Fiesta incorporaba de serie un sistema de audio con reconocimiento de voz. Este sistema, que admitía hacer llamadas, también reconocía una serie limitada de órdenes de control y comandos. También Pioneer junto con IBM desarrollaron un sistema de reconocimiento de voz que posibilitaba utilizar comandos de voz sencillos. [4]

Por seguridad y comodidad, la forma más sencilla para comunicarse con un coche es mediante la voz. De esta forma, los conductores no tienen que quitar la vista de la carretera ni manejar de forma manual diferentes dispositivos, algo que pondría en peligro la conducción y que está prohibido en la mayoría de las legislaciones sobre tráfico en el mundo. La industria del automóvil ya trabaja en diferentes tecnologías para incorporar programas de reconocimiento de voz intuitivos que permitan comunicarse con el vehículo a través del lenguaje natural.

2.2. FUNDAMENTACIÓN TEÓRICA

2.2.1. SISTEMA DE SEGURIDAD VEHICULAR

La seguridad vehicular es el conjunto de elementos y sistemas ubicados en el automotor para que el usuario se encuentre protegido de todo daño o riesgo en caso de un accidente de tránsito. Estos dispositivos dotan a los vehículos de los más altos niveles de seguridad y de mejores condiciones para una conducción adecuada.

La principal misión de la seguridad vehicular es la de proteger la integridad física y psicológica del conductor y de los ocupantes, antes y en el momento que se provoque el accidente, por medio de la reducción de los riesgos y la limitación de las consecuencias del impacto.

a) Sistema de Seguridad Activo

El objetivo principal de este sistema es disminuir el riesgo de accidentes que se pueden producir durante el uso regular y habitual del vehículo, por lo cual los dispositivos y elementos que lo conforman deben garantizar que el conductor no sufra perturbaciones en la marcha del vehículo y además que se facilite la manipulación de los mandos que permiten que éste pueda funcionar.

De acuerdo a la funcionabilidad de los diferentes dispositivos y elementos que conforman este sistema, se los puede clasificar:

Seguridad de marcha: Tiene por objetivo garantizar el correcto funcionamiento dinámico del vehículo en los mecanismos de dirección, frenos, suspensión y conjuntos de transmisión de las ruedas.

Sistema en la percepción de señales: Se determina por el buen estado y aplicación de los dispositivos de iluminación, dispositivos acústicos de advertencia y medidas de visión.

Seguridad de servicio: Establece la facilidad que brinda el vehículo para poder manipular los mandos y controles durante la conducción de manera comfortable.

b) Sistema de Seguridad Pasivo

Tiene la finalidad de disminuir las consecuencias posteriores a un accidente, para esto se debe tener en cuenta el comportamiento de la estructura del vehículo en el momento del impacto y luego del mismo, a más de la utilización de mecanismos y elementos que detengan o disminuyan la evolución del accidente.

Seguridad interior: Tiene como objetivo precautelar el estado de supervivencia de los ocupantes del vehículo cuando se provoca un accidente buscando reducir el impacto y facilitando la liberación de los accidentados.

Dentro de las medidas de protección se tiene el comportamiento de la deformación de la carrocería en el habitáculo, sistemas de sujeción, fijación de parabrisas, elementos de protección, etc.

Seguridad exterior: Busca reducir las lesiones de las personas que al estar involucradas en el accidente no son ocupantes del vehículo, por lo que en este caso influye el comportamiento de la carrocería ante el impacto y la forma exterior que tiene la misma. Dentro de las medidas de seguridad se debe considerar la disposición y aplicación de elementos como faros, limpiaparabrisas, manecillas, etc. [5]

2.2.2. SISTEMAS Y ACCESORIOS DEL VEHÍCULO

a) Sistema de iluminación del Automóvil

El sistema de iluminación, clave en la seguridad activa de un vehículo proporciona la cantidad adecuada de luz para la correcta conducción del automotor en diferentes situaciones adversas de visibilidad; a través de dispositivos lumínicos tales como luces incandescentes, leds, faros, neón, halógenos, etc., que en conjunto y en una distribución adecuada sirven de localización, información y advertencia de acciones ejecutadas por el conductor.

Normas internacionales del color de luz ayudan a interpretar de mejor manera la funcionalidad de cada dispositivo lumínico; faros traseros de color rojo, laterales o direccionales color ámbar y faros delanteros color amarillo o blanco.

De acuerdo a la ubicación de los dispositivos lumínicos, podemos dividir el sistema de iluminación en tres grupos:

- Faros y luces auxiliares de iluminación delantera.
- Faros frontales, laterales y traseros de señalización.
- Luz interior y otros dispositivos lumínicos.



Figura 2.1: Sistema de iluminación general de un vehículo

Fuente: <http://www.ro-des.com/mecanica/sistema-alumbrado-del-coche-que-es/> [6]

✓ **Faros y luces auxiliares de iluminación delantera**

Constituido principalmente por faros de corto y largo alcance, que deben abarcar todo el frente de visión del conductor y advierten la posición actual del vehículo.

Luces de corto alcance: Emiten una luz afilada y simétrica que evita el deslumbramiento a otros usuarios de la vía.

Luces de largo alcance: Emiten una luz más intensa y centrada, generalmente utilizada para la conducción en carretera.

Faros supletorios: Su finalidad es ensanchar el campo de visión.

Faros antiniebla: El uso es obligatorio en el automotor, utilizados en situaciones climatológicas adversas como nevada, neblina, lluvia.

Luces diurnas: Luces de leds, óptimas para la circulación a plena luz del día.

✓ **Faros frontales, laterales y traseros de señalización**

Luces de posición y direccionamiento: Permite a los demás conductores visualizar nuestra posición al encontrarnos expuestos en zonas de escasa visibilidad.

Indicadores de dirección: Luz intermitente que advierte de la acción a realizar por parte del conductor antes de ser ejecutada, ubicados en los extremos del vehículo.

Luces de emergencia: Sistema de alumbrado auxiliar activado por un interruptor que activa los cuatro intermitentes para advertir una situación de emergencia o avería del automotor.

Luces de marcha atrás: Advierte la ejecución del carro en dirección opuesta que se activa al colocar la marcha de retro en la caja de cambios.

Luces de freno: Emiten una luz de color rojo intensa y continua activada por el pedal de freno.

✓ **Luz interior y otros dispositivos lumínicos.**

Son las demás luces internas del vehículo como la luz de habitáculo o salón, guantera entre otras que permiten mejor iluminación en la noche y generalmente son activadas al abrir las puertas o a su vez por interruptores dedicados. [6]

En la siguiente figura se encuentra representado el sistema eléctrico de iluminación de un vehículo, con sus respectivos interruptores, luces y conexiones, para una mejor interpretación del mismo.

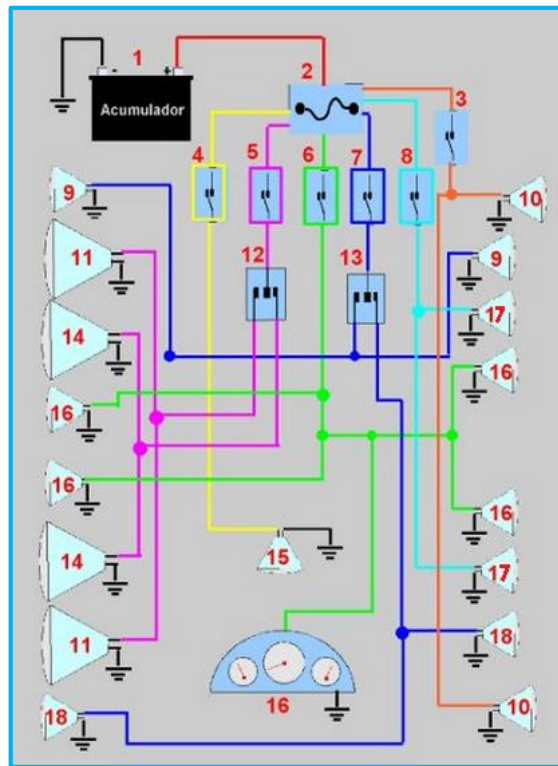


Figura 2.2: Esquema del Sistema eléctrico de iluminación
Fuente: <http://www.sabelotodo.org/automovil/siselectrico.html>

Descripción: 1) acumulador, 2) caja de fusibles, 3) interruptor de luces de reversa, 4) interruptor de luz de cabina, 5) interruptor de luces de carretera, 6) interruptor de luces de ciudad, 7) interruptor de luces direccionales, 8) interruptor de luz de freno, 9) luces de vía a la derecha, 10) luces de reversa, 11) luces altas de carretera, 12) conmutador de luces de carretera. 13) conmutador de luces direccionales, 14) luces bajas de carretera, 15) luces de cabina, 16) luces de ciudad y tablero de instrumentos, 17) luces de freno, 18) luces de vía a la izquierda.

b) Sistema de alarma de seguridad

Un sistema de alarma es un elemento de seguridad activa que permite advertir al conductor de situaciones anormales en el vehículo, y ejecutar una acción rápida de acuerdo al problema presente.

En la actualidad la mayoría de las alarmas modernas están constituidas por distintos tipos de sensores, interruptores, sirenas, receptor de radio para control inalámbrico, batería auxiliar de alimentación del sistema de alarma y una unidad central de monitoreo.

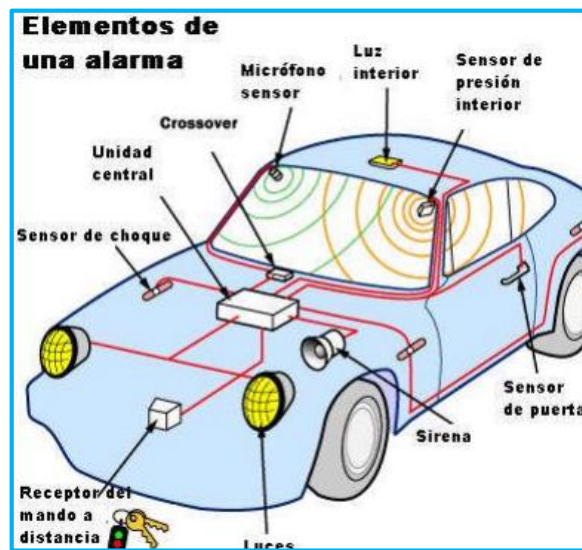


Figura 2.3: Ejemplo general de un sistema de alarma

Fuente: <http://www.aficionadosalamecanica.com/Alarma%20en%20el%20automovil.pdf>

De acuerdo a la tecnología utilizada y al propósito del sistema de alarma, se puede distinguir un sistema antirrobo y un sistema de seguridad centralizado.

✓ Sistema Antirrobo

Es la forma básica de un sistema de alarma, formada por una centralita que gestionara el activar de todos los interruptores, sensores, dispositivos sonoros del vehículo; además de un mecanismo trabavolante y corta corriente.

✓ **Sistema de seguridad centralizado**

Cumple la función principal de todo sistema de alarma convencional y adicionalmente bloquea el vehículo en su totalidad, impidiendo apertura desde el interior y exterior. Utiliza tecnología más sofisticada con una unidad central que puede desconectar si es necesario el computador o cerebro del automotor.

Distintas tecnologías aplicadas como GPS utilizada para la localización exacta a través de un centro de operaciones; una alarma silenciosa que lleva un registro celular que informa vía sms de la situación actual el vehículo; un sistema Keyless para el encendido electrónico sin la necesidad de utilizar llaves.

Vehículos de nueva generación incorporan sistemas biométricos en sus unidades aprovechando las características únicas que estos ofrecen como son lectores de retina, huella dactilar, reconocimiento facial, reconocimiento de voz. [7]

c) **Sistemas auxiliares primarios del vehículo**

✓ **Ventanillas eléctricas**

Sistema eléctrico que permite el control de apertura y cierre de las ventanas del vehículo a través de una acción mecánica ejercida por un motor, con la potencia necesaria para elevar y mantener fijo el cristal.

Elevavidrios, elevavidrios, alzavidrios entre otros nombres es el mecanismo del sistema y generalizando se encuentra constituido por un carril de guía, un cable de acero y el motor para ejercer el torque. [8]

✓ **Sistema de limpiaparabrisas**

Sistema auxiliar del vehículo cuya función consiste en la limpieza de los parabrisas de sustancias que puedan impedir la adecuada visualización al conductor, como puede ser lluvia, nieve, polvo, etc.

El sistema está formado por un motor eléctrico con un sistema de reducción incorporado capaz de vencer la fuerza de rozamiento que ejerce las raquetas o escobillas contra el cristal; un sistema de transmisión que consiste en transformar exteriormente el movimiento circular de la rueda dentada en movimiento alternativo en los brazo de raqueta por medio de un conjunto de bielas y manivelas. [9]

✓ **Sistema de ventilación y calefacción**

Los sistemas de ventilación y calefacción de los automóviles intervienen en las características ambientales del habitáculo para hacerlo más confortable, actuando sobre la temperatura del aire.

El sistema de ventilación facilita la renovación del aire en el interior del habitáculo del vehículo y sistemas más complejos combinan el sistema de calefacción y el de aire acondicionado. Dependiendo de las condiciones que se desee modificar, actuarán unos sistemas u otros, así para tomar aire del exterior será suficiente con contar con un sistema de ventilación, si se desea calentar la temperatura del aire actuará el sistema de calefacción o enfriarlo a través del aire acondicionado.

El sistema de calefacción consiste en calentar el aire ambiente haciéndolo pasar a través de un radiador por el cual circula en paralelo agua caliente procedente del circuito de refrigeración del motor. [10]

2.2.3. RECONOCIMIENTO DE VOZ

El reconocimiento automático del habla o reconocimiento de voz es una aplicación de la biometría que permite la comunicación entre humano – máquina, para ejecutar acciones posteriores al procesamiento de la información, tratando con las características fisiológicas de la persona como hábitos lingüísticos, acentos, idioma, entre otros.

En la Informática el reconocimiento de voz es la capacidad de un ordenador de interpretar la voz humana a un código binario comprensible por el computador, donde la información analógica (voz) es tomada en un determinado periodo de tiempo, para ser analizada a través de cualidades que pueden ser duración, calidad, intensidad, tono, etc.

a) Historia

En 1960, Gunnar Fant, profesor sueco publicó un modelo del habla acústico basado en el análisis de rayos X de individuos produciendo sonido fónicos específicos. En 1970, el Dr. Joseph Perkell utilizó rayos X en movimiento ampliando el modelo de Fant. Los sistemas de reconocimiento de voz utilizaban el resultado promedio de varios filtros analógicos para buscar coincidencias. En 1976, Texas Instruments construyó un sistema prototipo y la adquisición fue por la Fuerza Aérea de Estados Unidos y la corporación MITRE. A medida de los 80, el Instituto Nacional de Estándares y Tecnologías (NIST) desarrolla un grupo de estudio de procesamiento de voz, donde hasta la fecha promueve el continuo avance de la comunidad del reconocimiento de voz.

En la actualidad empresas dedicadas a la informática y tecnología han adoptado esta área para la creación de prototipos y dispositivos inteligentes aplicando el reconocimiento de voz, al ser esta aplicación versátil ganando mercado en el mundo y adaptando cada vez más al diario vivir de las personas. [11]

b) Aplicaciones

El uso del reconocimiento de voz a tenido un largo recorrido desde que era una tecnología utilizada únicamente en complejas máquinas industriales, para ahora es considerada una herramienta útil en nuestra vida cotidiana.

La primera industria que utilizó el software de reconocimiento fue la salud, sin embargo con el desarrollo de la informática, se amplió el uso del reconocimiento de voz aplicando a celulares, automóviles y ordenadores portátiles. El reconocimiento de voz añade un nivel de simplicidad para todos los usuarios, un ejemplo de ello es la compañía de Microsoft con su sistema operativo Vista, con una interfaz versátil y efectiva.

A continuación se hablará sobre las áreas donde el desarrollo del reconocimiento de voz ha tomado una mayor acogida:

✓ Domótica

La automatización del hogar ha tenido un desarrollo significativo, con el propósito de aumentar los niveles de comodidad y seguridad, controlando acciones básicas y reemplazando la ejecución manual de las mismas por dispositivos electrónicos, mecánicos comandados por un ordenador. Al disponer de un sistema domótico centralizado, es posible ejecutarlo simplemente por comandos de voz siendo más amistoso e innovador.

✓ Dispositivos móviles

En esta categoría se encuentran teléfonos celulares, ordenadores portátiles, tabletas, entre otros que cuentan con aplicaciones de reconocimiento de voz, que le permiten al propietario interactuar con su dispositivo sin utilizar las manos; ejemplos de ello la marcación por voz, consultas en internet e incluso redactar un mensaje de texto.

✓ **Personas discapacitadas**

El objetivo de la tecnología de reconocimiento de voz es el de mejorar la experiencia del usuario y aumentar los niveles de accesibilidad a diferentes entornos y necesidades del ser humano, es aquí donde juega un papel importante ya que existen personas con alguna discapacidad física que les priva de un desarrollo normal.

La Electrónica ha permitido la creación de diferentes prototipos que asisten a personas discapacitadas, y de acuerdo a sus necesidades se pueden desarrollar sistemas que mejoren su diario vivir.

✓ **Área de la Salud**

La industria de la salud se adapta al uso de esta tecnología, mejorando la productividad de los médicos y ayudando a obtener resultados más exactos. Profesionales de la salud hacen uso del reconocimiento de voz para dictados y transcripciones, registros de historiales e incluso asistencia médica. [12]

✓ **Mercado industrial**

En el mundo industrial la aplicación del reconocimiento de voz ha permitido mejorar la producción, aumentar ganancias dando mayor seguridad, exactitud en diferentes áreas y procesos. Es mucho más rápido ejecutar una acción oralmente que transcribirla con la posibilidad de ingresar mal algún dato de información.

Empresas automotrices hace pocos años atrás han implementado en sus líneas de producción vehículos con la tecnología de reconocimiento de voz, y en la actualidad este desarrollo es mucho mayor, creando vehículos con la capacidad de interactuar con el conductor, pero el inconveniente se encuentra en que la adquisición de automotores con este sistema son extremadamente caros.

c) Principios de Funcionamiento

Todo sistema de reconocimiento de voz, debe trabajar en dos formas distintas, el Modo de entrenamiento, que es la fase donde se obtienen los patrones y valores de referencia correspondientes a la señal analógica tomada por el locutor o usuario y el Modo de Reconocimiento, fase que a partir de la señal analógica el sistema tomará decisiones acerca de la identidad del locutor.

A continuación se muestra un diagrama en bloques del proceso general de entrenamiento y reconocimiento de la voz:

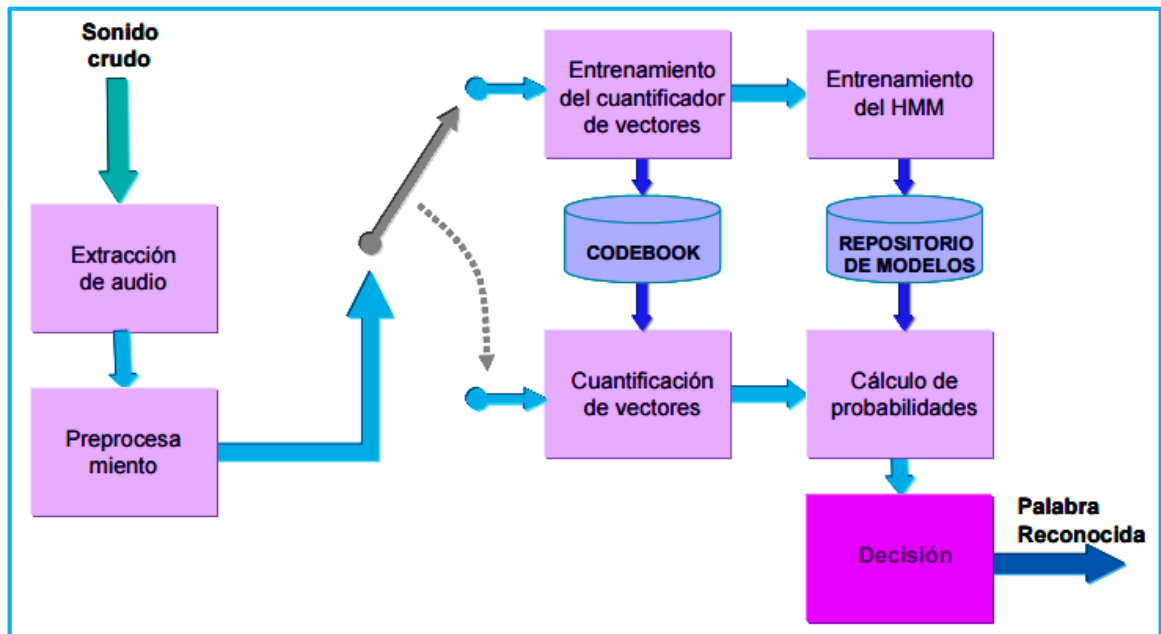


Figura 2.4: Proceso general entrenamiento, reconocimiento de la voz

Fuente: Introducción a la Inteligencia Artificial [14]

Los sonidos vocales son producidos por la acción de aire que impulsan los pulmones sobre el tracto vocal, en la laringe existen dos membranas (cuerdas vocales) que permiten variar el área de la tráquea por la cual circula (glotis). Durante el habla, dichas membranas permanecen en continuo movimiento (abrir y cerrar) lo que origina una mezcla de características en la generación de la voz, que se dan a llamar sonorizado y no sonorizado.

En los tramos sonorizados, las cuerdas vocales están normalmente cerradas y vibran al ser recorridas por la corriente de aire proveniente de los pulmones, su frecuencia o pitch se determina por el largo y tensión de las membranas y están en el orden de 50 a 400Hz. El efecto de este continuo abrir y cerrar de la glotis aparece como un tren de pulsos casi continuo. En la figura se muestra un tramo de voz sonorizada y su espectro:

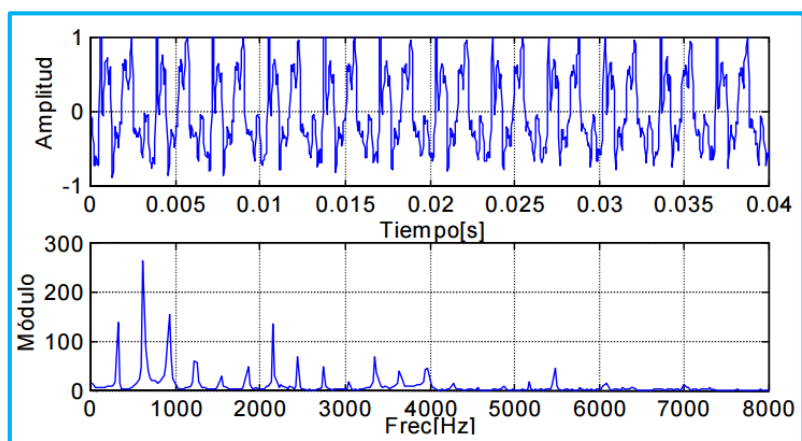


Figura 2.5: Señal de voz sonorizada
Fuente: Introducción a la Inteligencia Artificial [14]

En los no sonorizados, las cuerdas vocales permanecen abiertas y el aire pasa al resto del tracto vocal en forma de turbulencia. En la figura se muestra un tramo de voz no sonorizada y su espectro:

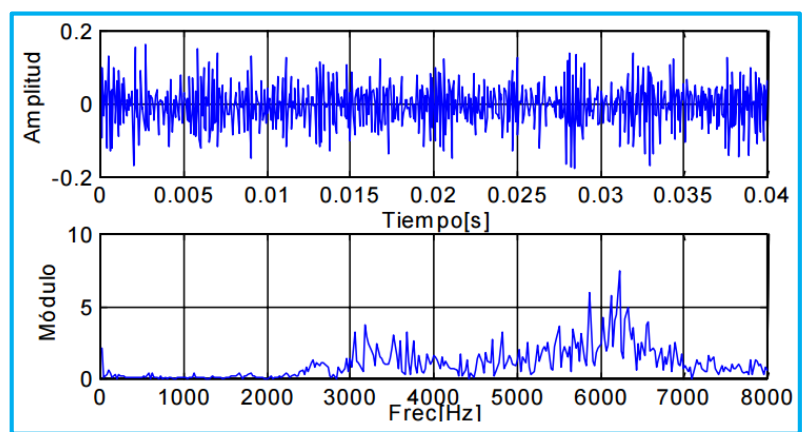


Figura 2.6: Señal de voz no sonorizada
Fuente: Introducción a la Inteligencia Artificial [14]

Queda así establecido que el resultado del habla en la región de la glotis será, una sucesión de pulsos o ruido blanco, según sea sonorizado o no, pero lo que termina de moldear el aspecto de la voz es el resto del tracto vocal que da forma al sonido actuando como un filtro que impone su propia respuesta en frecuencia.

Para poder trabajar con la señal pura de audio, primero es necesario ejecutar dos procesos sobre la misma para lograr caracterizarla. Estos dos procesos son comunes para el entrenamiento y reconocimiento.

Módulo de entrenamiento de audio

Realiza la separación o extracción de la señal propia de audio, de la señal ruido generada que no pertenece a la misma, puede adaptarse dinámicamente a variaciones de los niveles de ruido a través de filtros; el más utilizado es el Modelo Source-Filter, que considera a la voz producida por una señal de excitación en forma de impulsos que provienen de la acción de las cuerdas vocales, alternando de forma aleatoria con ruido blanco. En la siguiente figura se muestra cómo actúa el Modelo Source-Filter en la interpretación de la señal de voz:

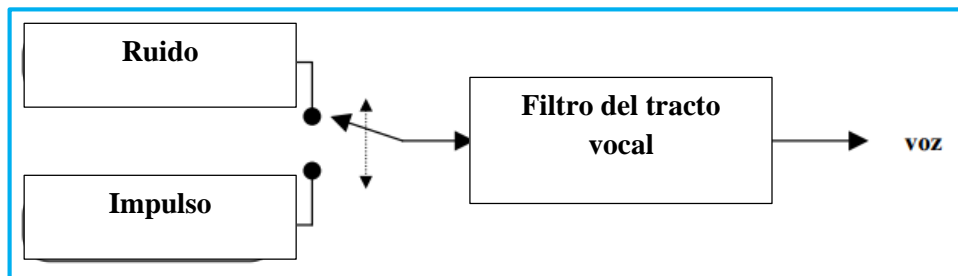


Figura 2.7: Interpretación de la voz del modelo Source-Filter
Fuente: Introducción a la Inteligencia Artificial [14]

Módulo Preprocesamiento

Obtiene las mejores características descriptivas del proceso de la señal de voz humana y trata de reducir la cantidad de información a manipular. Primero se obtiene una muestra con una frecuencia de 8Khz y cuantificación de 8 bits, datos mínimas para el tratado de la señal de la voz, tomando en cuenta que a mayor porcentaje de estos la resolución será mejor.

La señal debe ser separada en bloques de una duración tal que pueda ser considerada estacionaria. Para la señal de voz el ancho de la ventana temporal está comprendido entre 10ms y 45ms. Independientemente los bloques son superpuestos desplazados en una cantidad de muestras. A continuación un ejemplo de bloques de señal:

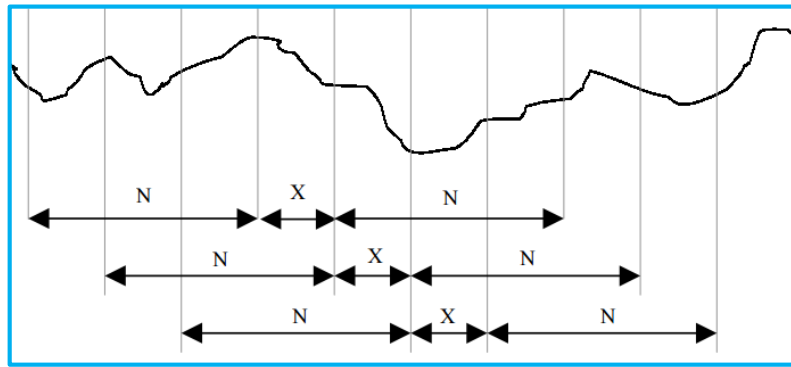


Figura 2.8: Ejemplo de una señal analógica separada en bloques
Fuente: Introducción a la Inteligencia Artificial [14]

Es evidente que una función ventana rectangular sería la más sencilla de implementar pero su uso trae algunos inconvenientes en el dominio de la frecuencia. Supongamos una señal senoidal a la que aplicamos una función de ventana rectangular:

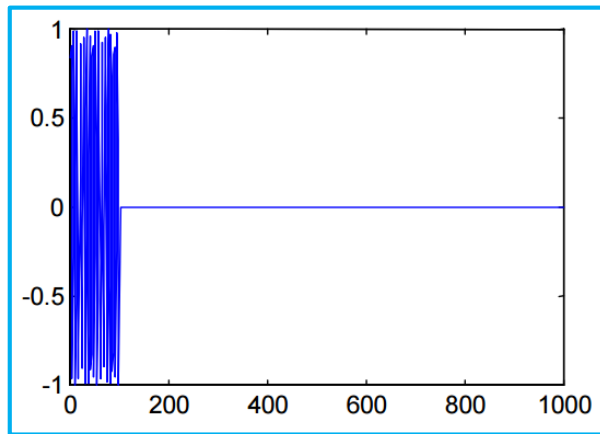


Figura 2.9: Señal senoidal aplicada ventana rectangular
Fuente: Introducción a la Inteligencia Artificial [14]

Aplicando la transformada de Fourier resulta:

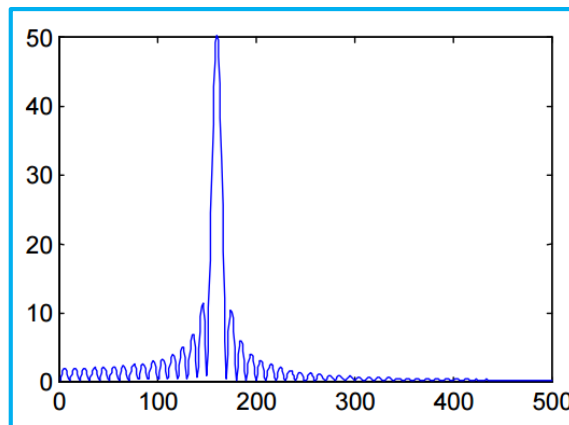


Figura 2.10: Transformada de Fourier de señal analógica
Fuente: Introducción a la Inteligencia Artificial [14]

Sabemos que el espectro de una señal senoidal es un impulso a la frecuencia correspondiente, se observa entonces la deformación de dicho espectro debido al truncamiento temporal y discontinuidad en los extremos.

Este problema puede ser minimizado multiplicando en el dominio temporal por una ventana que reduzca la influencia de los extremos. Una ventana muy utilizada es la ventana de Hamming:

$$W_n = \begin{cases} 0,54 - 0,46 \cos(2\pi n / (N-1)) & \text{Si } 0 \leq n \leq N-1 \\ 0 & \text{Si } 0 > n > N \end{cases}$$

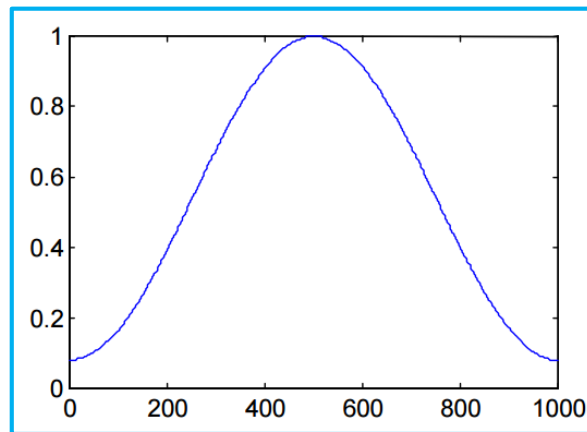


Figura 2.11: Gráfica función ventana de Hamming
Fuente: Introducción a la Inteligencia Artificial [14]

Tomando en cuenta la señal senoidal, aplicando la ventana de Hamming tenemos:

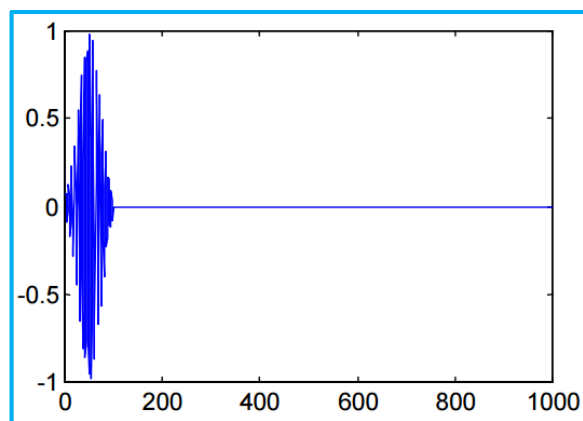


Figura 2.12: Señal analógica aplicada ventana de Hamming
Fuente: Introducción a la Inteligencia Artificial [14]

Su DFT es:

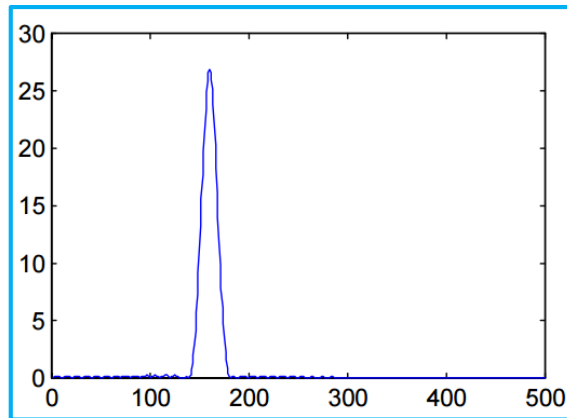


Figura 2.13: DTF de la señal resultante al utilizar ventana de Hamming
Fuente: Introducción a la Inteligencia Artificial [14]

El error queda minimizado, pero se pierde en cuanto a la energía total, ya que resulta menor que la original.

Análisis Cepstral

Continuando con el preprocesamiento de la señal se realiza un análisis Cepstral, donde descompone la señal de voz S , en excitación E y un filtro lineal caracterizado por su respuesta en frecuencia $H(f)$.

Sabemos que en dominio temporal se puede expresar como una convolución:

$$S(t) = H(f) * e(t)$$

Siendo H la respuesta al impulso del filtro

En el dominio de la frecuencia:

$$S(f) = H(f) * e(f)$$

$$\text{Siendo } S(f) = |S(f)|e^{j\varphi}$$

$H(f)$ es la envolvente del espectro y $E(f)$ los detalles que se aproximan a impulsos.

Observemos el siguiente ejemplo del espectro obtenido de la FFT de un segmento de una vocal tratado con una ventana de Hamming:

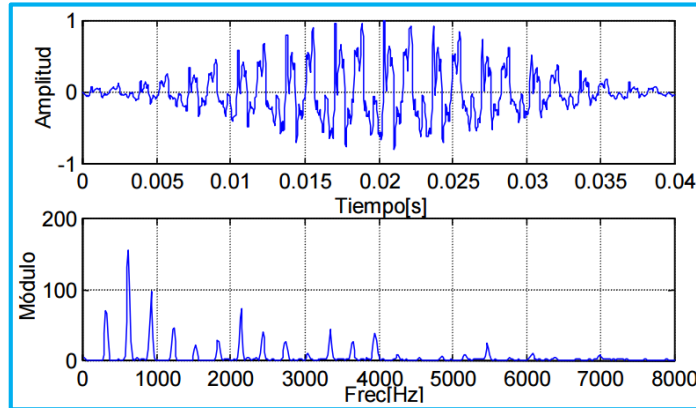


Figura 2.14: Espectro de una señal vocal tratada con ventana de Hamming
Fuente: Introducción a la Inteligencia Artificial [14]

Si aplicamos logaritmo a ambos miembros, sin considerar su fase, que normalmente no es necesaria en el campo de señales de voz, queda:

$$\log|S(f)| = \log|H(f)| + \log|E(f)|$$

En la figura observamos el espectro logarítmico a partir del gráfico anterior:

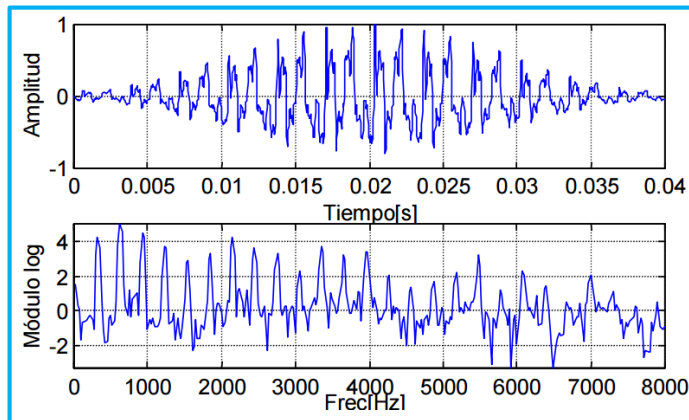


Figura 2.15: Espectro logarítmico de la señal resultante al utilizar ventana de Hamming
Fuente: Introducción a la Inteligencia Artificial [14]

Se logra pasar de una convolución de difícil resolución a una suma de logaritmos, en definitiva a una suma de dos señales de muy distintas características en frecuencia, ya que la señal de excitación tiene variaciones temporales muy superiores a la respuesta del filtro, si se realizara una nueva transformada de Fourier al resultado actual, se separarían claramente las señales en regiones apartadas ya que la respuesta al filtro ocupará las bajas frecuencias y la excitación las regiones de más alta frecuencia.

Se logra separar la envolvente de los impulsos o detalles de la voz, mediante el análisis Cepstral, que se diagrama a continuación:



Figura 2.16: Procesamiento de la señal analógica a Cepstrum
Fuente: Introducción a la Inteligencia Artificial [14]

Si se calcula la parte real del Cepstrum del ejemplo anterior se obtiene:

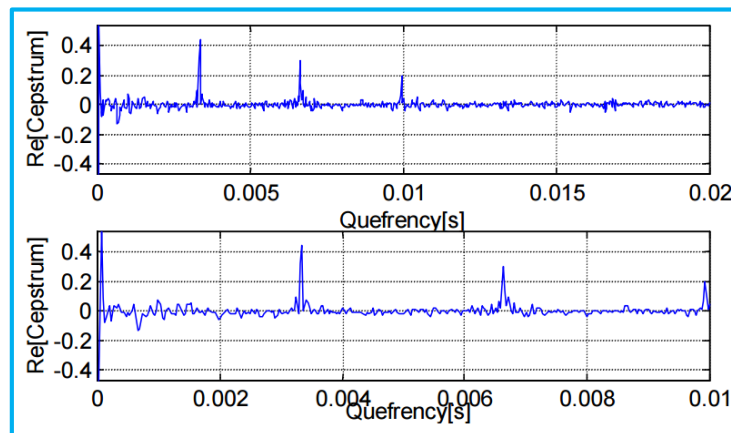


Figura 2.17: Parte real de una señal Cepstrum
Fuente: Introducción a la Inteligencia Artificial [14]

Se observa en dos escalas, la gran cantidad de información cerca del origen y más alejado solo los impulsos aislados que caracterizan el tono separados por el periodo del mismo. Los coeficientes de orden bajo proveen entonces información sobre la envolvente.

En general el análisis Cepstral logra pasar de una convolución a una multiplicación por medio de la transformada de Fourier y luego a una suma con el uso del logaritmo, finalmente con una nueva transformada se logra el Cepstrum, la palabra está originada por la alternación de las letras que la conforman Spectrum ya que en realidad su dominio no es temporal ni frecuencial sino un nuevo dominio Cepstral. Se tiene así un método para separar la información del tracto vocal.

✓ **Vector de Observación**

Al finalizar el tratamiento de la señal de voz, se obtendrá un vector que contendrá la información vocal, características que describen de la mejor manera posible la voz humana. Estos vectores son conocidos en el área del reconocimiento de voz como vectores de Observación.

✓ **Cuantificación de Vectores**

Una parte importante en cualquier tipo de procesamiento de voz es la optimización de los algoritmos en cuanto a velocidad y almacenamiento, entonces, la cuantificación de vectores trae consigo la idea de clasificar un conjunto de vectores, luego de lo cual se buscarán los mejores representantes para reducir el tamaño de la información almacenada. La forma de medir la fidelidad de un cuantificador es determinar el error que éste produce al reemplazar los datos de entrada que recibe por los vectores representantes o Codewords, dicho parámetro es llamada error por distorsión. La finalidad de un cuantificador se obtiene de un conjunto de vectores representativos llamado Codebook, que presente el menor error por distorsión.

Una vez procesada la señal de entrada, se utiliza la misma para entrenar o reconocer, siendo distintos los procesos que se ejecutan sobre los vectores de observación correspondientes dependiendo de la acción requerida.

❖ **Entrenamiento**

✓ **Módulo de entrenamiento del cuantificador de Vectores**

Entrena el cuantificador de vectores de manera no supervisada, sin tomar en cuenta características similares que permiten agrupar señales parecidas para mejorar el almacenamiento de las mismas.

✓ **Módulo de entrenamiento del HMM**

Modelo oculto de Márkov sus siglas HMM (Hidden Márkov Model), es un modelo estadístico que se basa en la probabilidad cuyo objetivo es determinar los parámetros desconocidos a partir de parámetros observables.

Utilizado en el ámbito del reconocimiento de la voz determina un modelo especial característico para cada señal ingresada, a través de similitudes preestablecidas entre palabras entrenadas. Una vez obtenido el modelo, lo ingresa en un repositorio común del sistema, junto con la palabra entrenada (establece la relación modelo - palabra).

❖ **Reconocimiento**

✓ **Módulo cuantificador de vectores**

Se encarga, una vez entrenado de agrupar, reducir y discretizar la cantidad de información, logrando asociar grupos de vectores a un sector determinado que será representativo.

✓ **Módulo Calculador de probabilidades**

Obtiene del repositorio común cada uno de los modelos, calculando la probabilidad de que cada uno de ellos haya podido generar la palabra de entrada.

✓ Módulo de decisión

Compara las probabilidades obtenidas por el módulo calculador y obtiene la máxima. De esta forma asocia el modelo a la palabra a reconocer.

Finalmente obtiene del repositorio la palabra asociada al modelo de mayor probabilidad e informa dicha palabra como palabra reconocida, asociando de esta forma la palabra a reconocer “de entrada” con la palabra reconocida “del repositorio”. [14]

2.2.4. MICROCONTROLADOR

Los microcontroladores (μ C, UC, MCU) son circuitos integrados capaces de ejecutar órdenes posteriormente grabadas en su memoria. Su composición está dada por varios bloques fundamentales, los cuales cumplen una tarea específica.

El termino microcontrolador está dado por dos palabras “Micro”-”Controlador” que en relación sería “pequeño” y “controlar” procesos los cuales son definidos mediante la programación.

Un microcontrolador en su interior se encuentra constituido por las tres unidades fundamentales de un computador que son la unidad central de procesamiento, memoria y periféricos de entrada y salida. Además un circuito oscilador, para controlar la frecuencia de operación.

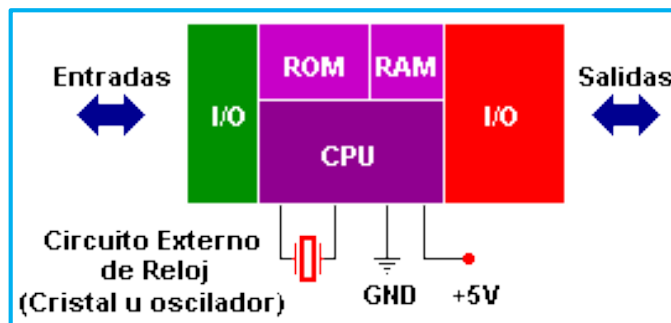


Figura 2.18: Esquema básico de un microcontrolador
Fuente: http://perso.wanadoo.es/luis_ju/pic/pic01.html

La unidad central de procesos: Cuyas siglas en ingles son “CPU” en el núcleo del microcontrolador, que se encarga de ejecutar las instrucciones almacenadas en la memoria, habitualmente llamamos procesador o microprocesador.

La memoria: Entendemos por memoria los diferentes componentes del microcontrolador que se emplean para almacenar información durante un periodo determinado de tiempo, los cual utilizaremos durante la ejecución del programa como son el código y registros. Esto nos permite identificar la memoria de programa y memoria de datos.

La memoria generalmente está constituida por una memoria RAM compuesta por registros que almacena datos temporales y una memoria EEPROM para el almacenamiento del programa que se quiere ejecutar.

Las unidades de entrada y salida: Son los puertos que utiliza el microcontrolador para recibir y enviar datos en forma serial o paralelo.

El reloj principal: Circuito interno que sirve para la sincronización de todas las operaciones del sistema, a través de un dispositivo exterior. [15]

2.2.5. ARDUINO

Arduino es una plataforma de código abierta (Open Hardware) para la creación de prototipos basados en software y hardware flexibles y fáciles de usar. Arduino puede tomar información del entorno a través de sus pines de entrada de toda una gama de sensores y ejecutar acciones con actuadores que controlan procesos exteriores.



Figura 2.19: Introducción al mundo Arduino

Fuente: <http://www.cortoc.com/2011/12/introduccion-arduino.html>

Tiene su acogida en el mercado por ser un entorno de desarrollo totalmente libre de licencia, empezando por la creación de varios tipos de placas o Shields que integran microcontroladores, entradas y salidas analógicas y digitales, etc.; las cuales pueden venir ensambladas o para armar, previamente diseñadas en una fábrica.

Las placas electrónicas se programan mediante el lenguaje de programación Arduino *Wiring* que es una plataforma abierta de prototipos electrónicos compuestos de un entorno de programación, una tarjeta de prototipo, además información adicional sobre proyectos; y el entorno de desarrollo Arduino *Processing* lenguaje de programación de código abierto, inicialmente desarrollado para servir como un software gráfico y para el aprendizaje de los fundamentos de la programación en un contexto visual.

Los proyectos hechos con Arduino pueden ejecutarse sin necesidad de conectar a un ordenador, si bien tiene la posibilidad de hacerlo y comunicar con diferentes tipos de software.

a) Hardware

El hardware de Arduino está basado en un microcontrolador Atmel AVR, por su sencillez y bajo coste, cuya programación se basa en lenguaje C/C++ que utiliza el compilador de código abierto *avr-gcc*. Adicionalmente cuenta con puertos de entrada/salida y el cargador de arranque que se ejecuta en la placa.

b) Software

El entorno de desarrollo de Arduino contiene un editor de texto para escribir el código, un área de mensaje, una consola de texto, barra de herramientas y demás instrucciones. Los programas escritos usando Arduino son llamados “Sketches”. Cuando cargas un Sketch se usa el Bootloader, que es un pequeño programa precargado al microcontrolador, que te permite cargar el código de un proyecto sin usar ningún hardware adicional.

El Bootloader es activado durante unos segundos cuando se reinicia el microcontrolador, después de eso inicia el programa del proyecto cargado.

El software de Arduino realiza cambios en el código escrito para verificar si está correcto. Primero el código del programa pasa por el compilador avr-gcc, el cual convierte el código de alto nivel a lenguaje de máquina, luego es combinado con librerías estándar de Arduino que proveen las funciones. Como resultado se obtiene un archivo .hex que contiene los bytes necesarios para programar la memoria del microcontrolador y luego enviar a través del puerto USB o serial utilizando el Bootloader para cargar el proyecto.

c) Ventajas

- Los dispositivos Arduino son accesibles, ya que las placas electrónicas son relativamente baratas en comparación con otras plataformas de microcontroladores.
- El software de Arduino es multiplataforma, y puede ejecutarse en el sistema operativo Windows, Macintosh OSX y Linux.
- El entorno de programación del Arduino es simple y flexible, ideal para la creación de proyectos de bajo y alto nivel.
- El hardware es extensible, por lo que el diseñador puede crear su propio módulo, ampliándolo y mejorándolo.
- El software de Arduino es libre y no requiere licencia para ser adquirido.
- Posee información adicional en internet, herramientas, librerías.
- El idioma se puede ampliar a través de bibliotecas C++.

d) Desventajas

- Dado que la programación no se realiza en assembler, el precio a pagar por el uso de las librerías es un retraso en la ejecución de las instrucciones, algunos microsegundos que en el caso de dispositivos de uso cotidiano son irrelevantes, pero significativos a la hora de hacer adquisición de datos.

- El hecho de que la plataforma venga ya ensamblada le quita flexibilidad a los proyectos, obligando a usar un espacio y forma acorde con la placa; para superar esto, se debe trabajar con un microcontrolador diferente al de la plataforma y diseñar la placa desde cero como los PICs. [16]

2.2.6. PROPUESTA DE SOLUCIÓN

La implementación del Sistema Electrónico por comando de voz para la seguridad física, iluminación y accionamiento automático de los accesorios primarios de un automóvil, permite al conductor adquirir un sistema de seguridad único de desbloqueo del vehículo y control por medio de comandos de voz a los diferentes sistemas y accesorios inmiscuidos en el proyecto sin la necesidad de la ejecución manual.

CAPÍTULO III

METODOLOGÍA

3.1. MODALIDAD DE LA INVESTIGACIÓN

El proyecto es de carácter Bibliográfico, ya que la información se sustenta de tesis y monografías obtenidas de buscadores educativos de establecimientos nacionales e internacionales, además de links sobre proyectos, artículos, revistas y libros enfocados en la Electrónica.

Se la puede caracterizar en una investigación Experimental, en la que se obtiene la información por medio de la observación de los hechos, y que se encuentra dirigida a modificar la realidad con el propósito de dar solución a los problemas ocurridos en determinadas condiciones.

El proyecto amerita ser catalogado como una investigación de Campo, al inmiscuir al investigador en el ambiente natural en el que conviven las personas a través de la observación, para posteriormente analizar y sacar conclusiones de posibles soluciones de situaciones que agobian el buen vivir de la sociedad.

Se la considera en una investigación Aplicada, al utilizar los conocimientos adquiridos en el proceso de formación superior en la vida cotidiana, con el propósito de sacar provecho y mejorar alguna falencia o necesidad en las personas.

3.2. RECOLECCIÓN DE LA INFORMACIÓN

La recolección de la información se realizó a través de links de internet y fuentes bibliográficas con el propósito de ampliar la visión del proyecto; revistas, tutoriales y hojas de datos técnicos para un enfoque teórico-práctico y manuales de los dispositivos a utilizar para su mejor entendimiento.

3.3. PROCESAMIENTO Y ANÁLISIS DE DATOS

Toda la información recolectada del estudio de los distintos elementos que conforman el proyecto sirvió para el diseño del hardware con el propósito de crear un dispositivo compacto; el estudio de la estructura del vehículo permitió ubicar al prototipo y ser adaptado en una posición adecuada en el mismo sin perder su estética; además la programación se la realizó en un software de código abierto, que cumplió nuestras expectativas, el mismo que es multiplataforma y se puede ejecutar en una PC con sistema operativo Windows para desarrollar la aplicación en el microcontrolador.

La programación del módulo de reconocimiento de voz se lo realizó en el software del mismo, además los diferentes circuitos fueron simulados para su posterior diseño tanto las placas electrónicas y los módulos que conforman el Prototipo Electrónico.

3.4. DESARROLLO DEL PROYECTO

- ✓ Estudio de los sistemas y accesorios del vehículo utilizados en el proyecto.
- ✓ Recolección de información sobre sistema de reconocimiento de voz.
- ✓ Análisis de los diferentes dispositivos del Prototipo Electrónico.
- ✓ Adquisición de los dispositivos electrónicos y medios de interfaz.
- ✓ Diseño de circuitos y Módulos Electrónicos del Prototipo.
- ✓ Simulación de circuitos y Módulos Electrónicos del Prototipo.
- ✓ Adaptación de los diferentes Módulos Electrónicos, dispositivos de interfaz que conforman el Prototipo Electrónico.
- ✓ Implementación del Prototipo Electrónico por reconocimiento de voz en el vehículo.
- ✓ Pruebas de funcionamiento del proyecto.
- ✓ Recolección y análisis de resultados.

CAPÍTULO IV

DESARROLLO DE LA PROPUESTA

En el presente capítulo se muestra el desarrollo de la propuesta, cumpliendo cada uno de los objetivos planteados en el análisis del proyecto.

Al encontrarse enfocado en el área Automotriz, se realizó un análisis de los sistemas electrónicos de reconocimiento de voz existentes en el mercado, identificando características favorables y deficientes para medir la factibilidad del mismo. Posteriormente se propone el desarrollo del prototipo con la selección y análisis de los dispositivos que comprenden la parte de hardware, la programación, diseño y montaje del Sistema Electrónico.

Vale recalcar que el propósito del proyecto es diseñar un Prototipo Electrónico que cubra las limitaciones de una persona discapacitada con deficiencia en una extremidad superior, que le complica el control manual de ciertos dispositivos del vehículo; no obstante eso no priva a otras personas de la utilización del mismo.

4.1. SISTEMAS DE RECONOCIMIENTO DE VOZ PARA VEHÍCULOS

4.1.1. App Drivestyle Digital



Figura 4.1: Aplicación SIRI de Apple

Fuente:http://appleinsider.com/articles/12/02/28/mercedes_benz_to_add_siri_integration_to_new_a_class_line.html

Aplicación creada inicios de este año por la empresa alemana Daimler AG que combina un Kit Drive Plus para iPhone con el sistema multimedia del vehículo, desarrollado explícitamente para la marca Mercedes Benz para sus coches categoría clase A. Todas sus funciones se basan en la aplicación SIRI de Apple, y para su funcionamiento es necesario disponer del iPhone 4S y versiones posteriores y conexión a internet y su aplicación debe ser adquirida al ser un software con licencia. Entre sus principales funciones tenemos:

- Control multimedia del vehículo.
- Acceso a redes sociales como Facebook, Twitter.
- Descarga de canciones a través iTunes Store de Apple.
- Radio por internet.
- Navegación GPS con software Garmin. [17]

4.1.2. Pioneer Zypr

La compañía Pioneer hace un par de años atrás creó una aplicación de reconocimiento de voz por conexión a Internet para teléfonos móviles que reconoce las palabras emitidas por el usuario y las interpreta para cumplir su determinada función.

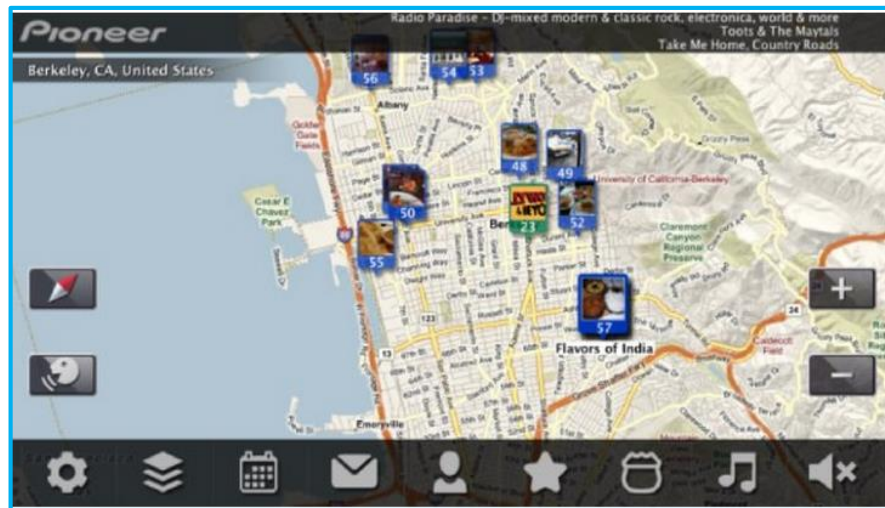


Figura 4.2: Aplicación Pioneer

Fuente: <http://es.engadget.com/2011/11/11/pioneer-zypr-reconocimiento-de-voz-a-lo-siri-gratuito-y-para/>[18]

Es una aplicación flexible que puede conectarse a un modem para acceder a internet y permite la integración de varios dispositivos electrónicos dependiendo de la necesidad. Una ventaja de la aplicación es que es de código abierto, no requiere licencia, pero el financiamiento del servicio de la nube se basa en anuncios publicitarios. Entre sus principales funciones tenemos:

- Búsqueda y navegación mediante GPS.
- Reproducción de música mediante internet.
- Acceso a redes sociales Facebook, Twitter.
- Acceso al buscador Google. [18]

4.1.3. Ford Sinc

Uno de los fabricantes actuales que más apuesta por los sistemas de reconocimiento de voz es Ford. Tiene como soporte a la empresa norteamericana Nuance Communications Inc., una de las principales compañías desarrolladoras de tecnología por voz, conocida por los usuarios domésticos por contar con aplicaciones para ordenadores y dispositivos portátiles como Dragon Dictation. Nuance posee el sistema de reconocimiento de voz Dragon Drive, orientado a la integración con diferentes fabricantes de coches.

Nuance realizó un convenio con la empresa automotora Ford para la creación de un sistema de reconocimiento de voz que se implementó en sus líneas de coches con conexión a internet, creando el sistema Ford Sinc que comprende el lenguaje natural del habla.



Figura 4.3: Sistema Sinc3 de Ford

Fuente: <http://www.bizjournals.com/boston/news/2012/05/22/nuance-dragon-drive.html> [19]

Este sistema se enfoca en la navegación, entretenimiento y las comunicaciones, estrenado en el auto Ford B-Max. Sus expectativas no cumplieron los requerimientos planteados, mejorando su sistema en los presentes años con su versión Sinc2 y Sinc3, que se encuentran disponibles en autos clase A y B modelos 2015. Entre sus principales funciones tenemos:

- Búsqueda y navegación mediante GPS.
- Reproducción de multimedia.
- Reproducción de música mediante internet.

- Acceso a redes sociales Facebook, Twitter.
- Envío de sms y correo electrónico.
- Llamadas telefónicas.
- Consulta en internet.
- Compatibilidad con teléfonos inteligentes. [19]

4.1.4. Android Auto

Plataforma creada por Google para autos con conexión a internet, que se conecta a través de bluetooth a teléfonos y tabletas y puede ser controlado por estos o directamente por la pantalla táctil del panel de mando del vehículo.

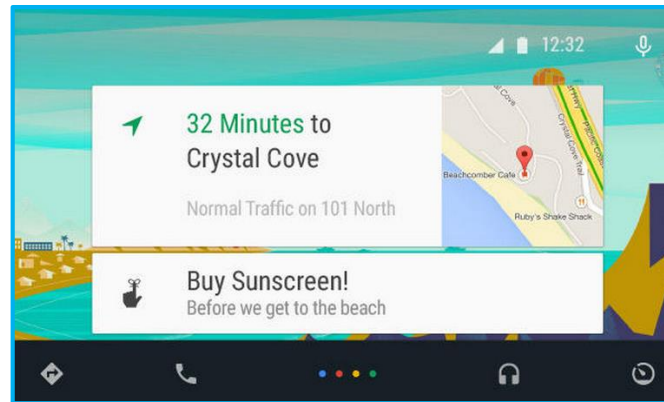


Figura 4.4: Aplicación Android Auto

Fuente: <http://www.cnet.com/es/analisis/google-android-auto/> [20]

La aplicación cuenta con 5 secciones:

Navegación: utiliza Google Map para la navegación GPS, a través de comandos de voz, plataforma tradicional que se encuentra en teléfonos y computadoras portátiles.

Teléfono: Al encontrarse conectado con nuestro teléfono tiene la opción de adquirir información de nuestros contactos más usados frecuentemente y mostrarnos en su panel.

Música: Funciona con Google Music y permite la opción de adelantar, cambiar, detener y reproducir música por comando de voz.

Aplicaciones: Sección abierta para futura creación de aplicaciones a través de los desarrolladores y programadores.

Android Auto Overview: Es una visión general del uso de las aplicaciones del sistema. Un resumen del usuario con sus contactos favoritos, lista de música más escuchada y direcciones más transcurridas. [20]

A través de la investigación sobre los sistemas y aplicaciones de reconocimiento de voz para vehículos existentes en el mercado se ha llegado a la conclusión que la mayoría de ellos se enfocan como prioridad en la navegación para la obtención de direcciones, el entorno de multimedia con el control de reproducción de música y por último el acceso a redes sociales. Todos estos sistemas requieren de una conexión obligatoria a internet, ya sea a través de dispositivos portátiles o del automotor.

En la actualidad empresas Automotoras como Mercedes Benz, Audi, Volvo, Honda, Cadillac, Hyundai, Kia, Ford siendo los más relevantes, implementan Sistemas Electrónicos que integran la navegación, comunicación y control de multimedia, todos por reconocimiento de voz, pero el inconveniente es que son implementados en autos de categoría A y B y sus precios son extremadamente elevados. La adquisición tan solo del sistema electrónico se encuentra en una estimación sobre los 1500 a 2000 dólares, y requiere de ciertos parámetros obligatorios en el vehículo.

4.2. ANÁLISIS DE FACTIBILIDAD

4.2.1. Factibilidad Técnica

Se dispuso de todo los dispositivos necesarios para el diseño del prototipo electrónico con sus respectivos manuales y guías técnicas; además de los conocimientos necesarios por parte del investigador para el diseño, programación y montaje. Todo el trabajo se elaboró rigiéndose a parámetros técnicos y de seguridad.

4.2.2. Factibilidad Operativa

El diseño de la interfaz del prototipo electrónico es muy funcional, su simplicidad le permite al usuario interactuar de la mejor manera a través de una pantalla GLCD que informa la ejecución de algún proceso, adicionalmente sonidos emitidos por un altavoz con mensajes acústicos de acuerdo a la tarea a realizar; todo el prototipo se basa en el control de los diferentes accesorios del vehículo utilizando comandos de voz, donde basta con hablar para controlar el sistema sin la necesidad de un complejo conocimiento tecnológico; como resultado una alta eficiencia en la parte operativa.

4.2.3. Factibilidad Económica

El presupuesto económico que sostuvo el proyecto fue factible y sustentado en su totalidad por parte del investigador, que cubrió los gastos de todo lo requerido para llevar a cabo el proyecto desde el diseño hasta la implementación.

4.3. SELECCIÓN DISPOSITIVOS ELECTRÓNICOS

El Prototipo Electrónico se establece en el entorno Arduino, plataforma de desarrollo libre, gratuito con una extensa gama de dispositivos hardware, y dependiendo de las necesidades del proyecto se ha seleccionado cada dispositivo tratando de explotar al máximo las cualidades y características para un eficiente funcionamiento.

4.3.1. Arduino Mega 2560

El Arduino mega es una placa electrónica basada en el Atmega2560. Cuenta con 54 pines digitales de entrada/salida, de los cuales 15 son de salida PWM. Además con 16 entradas analógicas, 4 puertas seriales UARTs, un oscilador de 16 MHz, conexión USB, conector de alimentación, una cabecera ISCP y un botón de reinicio.

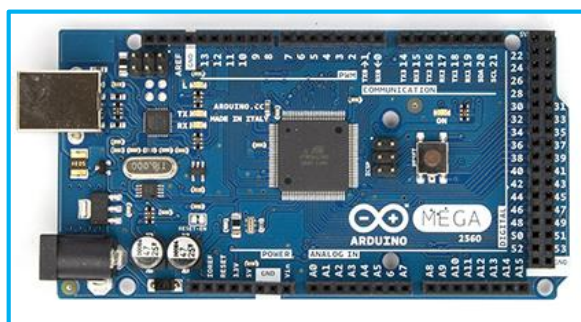


Figura 4.5: Hardware Arduino Mega2560

Fuente: <http://www.arduino.cc/en/Main/arduinoBoardMega2560> [21]

El Arduino Mega puede ser alimentado a través de la conexión USB o con una fuente de alimentación externa la cual se selecciona automáticamente. De fuente externa utilizando una adaptador de CA a CC o una batería conectando los pines GND y Vin de la placa. [21]

✓ Características

Características	Descripción
Microcontrolador	Atmega2560
Tención de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límites)	6-20V
Digitales pines I/O	54 (15 pines salida PWM)
Analógicos pines entrada	16
Corriente DC por pin I/O	40mA
Corriente DC de pin 3.3V	50mA
Memoria Flash	256 KB
SRAM	8KB
EEPROM	4KB
Velocidad de reloj	16 MHz

Tabla 4.1: Datos técnicos Arduino Mega2560

Fuente: El investigador

✓ Evaluación componente

En el mercado se encuentra diferentes dispositivos que cumplen los requerimientos del sistema, pero por sus características se ha optado por Arduino Mega 2560, uno de los más utilizados en la actualidad. En la siguiente tabla se realiza un análisis del mismo con dispositivos de características similares:

Características	Mega 2560	Uno	Leonardo	Due
Tipo de microcontrolador	Atmega2560	Atmega328	Atmega32U4	AT91SAM3X8E
Velocidad de reloj	16 MHz	16 MHz	16 MHz	84 MHz
Pines digitales E/S	54	14	20	54
Entradas analógicas	16	6	12	12
Salidas analógicas	0	0	0	2 (DAC)
Memoria de programa (Flash)	256 kb	32 kb	32 kb	512 kb
Memoria de Datos (SRAM)	8 kb	2 kb	2.5 kb	96 kb
Memoria auxiliar (EEPROM)	4 kb	1 kb	1 kb	0 kb
Precio	\$85,00	\$35,00	\$50,00	\$75,00

Tabla 4.2: Características módulos más eficientes del mercado

Fuente: El investigador

En el **Anexo III** se encuentra el manual técnico de la placa electrónica con información general del módulo, datos técnicos, modos de utilización e información adicional que permitirá un mejor entendimiento del dispositivo.

4.3.2. EasyVR SHIELD 2.0

EasyVR 2.0 es una tarjeta de reconocimiento de voz con múltiples usos, diseñado para añadir capacidades de reconocimiento de voz versátil, robusta y rentable para cualquier aplicación de electrónica. EasyVR es la versión de tercera generación del módulo VRbot y se basa en las características de su predecesor.

Es capaz de reproducir audio de una lista de sonidos previamente grabados en su memoria flash interna; dichos sonidos pueden ser de cualquier tipo que se puedan obtener a partir de archivos formato wav.

EasyVR dispone de un software de desarrollo que permite una rápida familiarización y empleo de todas sus aplicaciones y puede trabajar como host interactuando con cualquier tipo de microcontrolador (Pic, Basic Stand, ATMEL, Arduino, etc) a través de una interfaz UART.

Entre sus aplicaciones principales podemos destacar:

- Sistema de control de propósito general gobernado mediante voz.
- Automatización de aplicaciones en el ámbito doméstico.
- Control de acceso por voz.
- Sistemas robóticos controlados por voz.
- Reproducción de mensajes, sonidos, música, etc.

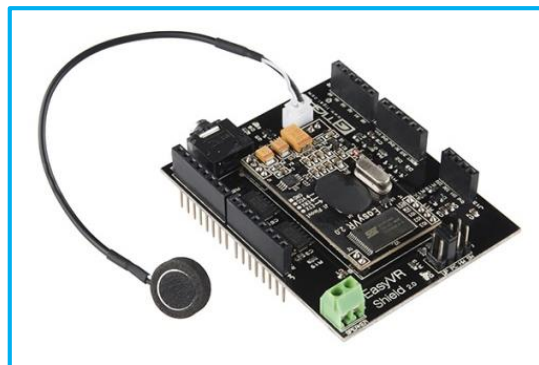


Figura 4.6: hardware EasyVR SHIELD 2.0
Fuente: <http://arduino.cl/easyvr-shield-2-0/> [22]

✓ **Características**

- Dispone de 28 altavoces independientes (SI en idioma inglés, italiano, japonés, alemán, español y francés), todos listos para ejecutar en el programa.
- Soporta hasta 32 Triggers o comandos de parlantes dependientes definidos por el usuario, así como también soporta clave de voz.
- Interfaz de usuario gráfica simple y fácil de usar para programar comandos de voz.
- La tarjeta puede ser usada con cualquier servidor con una interfaz UART energizada con 3.3V – 5V.
- Protocolo simple y robusto para acceso y programación del módulo a través de la tarjeta controlada.
- Incluye Sonicnet para controlar uno o más módulos EasyVR 2.0 wirelessly con fichas de sonido generadas por el módulo u otra fuente de sonido.
- Puede crear los sonidos usando la herramienta Sensory QuickSynthesis4.
- El nuevo EasyVR GUI incluye comandos para procesar y descargar tablas de sonido hechas a medida hacia el módulo.
- Conector para entrada de micrófono.
- Salida de parlantes de 8 ohmios.
- Audífonos Jack.
- Acceso a los pines de entrada y salida del easyVR.
- Led programable para comprobar comunicación durante procesos de reconocimiento.
- Incluye librería Arduino. [22]

✓ **Evaluación componente**

En la siguiente tabla se analiza las características de los módulos de reconocimiento de voz más utilizados en el entorno de desarrollo de proyectos electrónicos, tomando en cuenta la accesibilidad a estos:

Características	Vrbot	EasyVR Shield 2.0	Módulo Reconoc V3
Voltaje Alimentación	3.0 – 5.0 V	3.3 – 5.0 V	3.3 – 5.0 V
Altavoces independientes predefinidos	23	28	28
Capacidad comandos	25	28	80
Shield	No	Sí	Sí
Idiomas	Inglés, Italiano, Japonés, Alemán	Inglés, Italiano, Japonés, Alemán, Español, Francés, Coreano	Inglés, Italiano, Japonés, Alemán, Español, Francés, Coreano
Interfaz UART	No	Sí	Sí
Control comando efectivo simultáneo	No	No	Sí
Precio	\$50,00	\$90,00	\$130,00

Tabla 4.3: Cuadro comparativo módulos reconocimiento voz
Fuente: El investigador

En el **Anexo IV** se encuentra el manual técnico del módulo de reconocimiento de voz con información general, datos técnicos, modos de utilización e información adicional que permitirá un mejor entendimiento del dispositivo.

4.3.3. Pantalla GLCD 12864ZW

Es un dispositivo de visualización ideal para la aplicación en microcontroladores, pantalla GLCD gráfica de 128 píxeles de largo por 64 píxeles de ancho, trabaja con el controlador ST7920, posee retroiluminación y contraste variable, transmisión de datos en forma serial y paralela de 4, 8 bits de información.

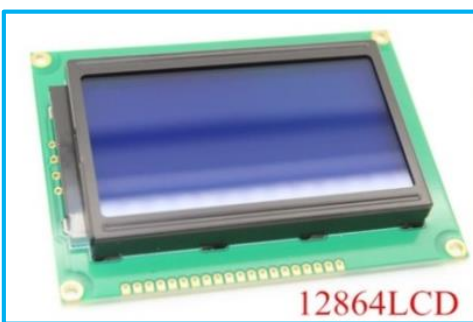


Figura 4.7: Pantalla GLCD 12864ZW

Fuente: <http://es.aliexpress.com/w/wholesale-128x64-graphic-lcd.html>

✓ **Características**

- Pantalla 128 x 64 pixeles.
- Luz de fondo azul.
- Controlador ST7920.
- Voltaje de operación 4.5V a 5.5V
- Soporta transmisión 4 bits, 8 bits, interfaz serial MPU.
- 64 x 16 bits display RAM (DDRAM)
- Soporta 16 palabras x 4 líneas.
- Rango LCD display de 16 palabras x 2 líneas.
- 64 x 256 bits display grafico RAM (GRAM)
- 2 Mbits para generación ROM de caracteres (CGROM)
- Pin de Reset externo.
- Bajo consumo de corriente: Normal 250uA, Modo Stand by 30uA. [23]

✓ **Esquema de Pines**

En la siguiente figura se muestra la disposición de 20 pines para la conexión del display, los mismos que son utilizados para la alimentación, reset, iluminación del display y el modo de transmisión con un bloque de 8 bits para los datos.

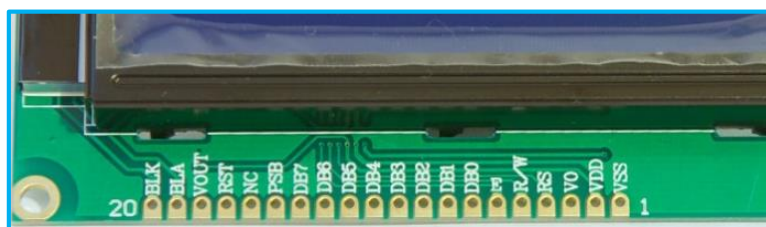


Figura 4.8: Pines LCD grafico 12864ZW

Fuente: El investigador

En la siguiente tabla se describe cada pin de la pantalla GLCD, con su significado y una breve descripción de la función que desempeña en el dispositivo visual:

Pin	Significado	Descripción
VSS/GND	Tierra	-
VDD/VCC	Vin	-
V0	V driver LCD	Ajuste contraste LCD
RS	Register Select	Multifunción si se trabaja en serie o paralelo
R/W	Read/Write line	En paralelo indica el tipo de operación
E	Real/Write Enable	En paralelo señal de comienzo de actividad(lectura de datos) En modo serie (Señal de reloj).
DB0-DB3	Data bus 0 – 3	Bits menos significativos modo paralelo 8 bits.
DB4 – DB7	Data bus 4 – 7	Bits más significativos en modo paralelo 8 bits. Bus de datos en modo paralelo 4 bits.
PSB	Serial/Palallel Control port	Selecciona el modo de funcionamiento (Serie/ Paralelo)
NC	Not Connected	-
RST	Reset	-
Vout	Salida de doblador de voltaje	-
BLA	BackLight(+)	Terminal positivo de retroiluminación
BLK	BackLight(-)	Terminal negativo de retroalimentación

Tabla 4.4: Descripción de pines de pantalla GLCD

Fuente: <http://rduinostar.com/documentacion/datasheets/lcd-st7920-hj12864zw/>

En el **Anexo V** se encuentra el DataSheet de la pantalla GLCD 12864ZW, con información pertinente relacionada a la visualización de símbolos y formas de transmisión de la información desde el microcontrolador.

e) Teclado Matricial Alfanumérico

Es un periférico utilizado para proporcionar datos y generar órdenes a un proceso digital, formado por un conjunto de pulsadores que solo se acciona uno en cada instante. Esto hace que se deba realizar un proceso de conversión de la información para almacenarla en un código numérico específico.

Es ideal para aplicaciones electrónicas con una amplia funcionabilidad, al ser un medio de interacción entre hombre - sistema, permitiendo gobernar ingresando códigos, enviando caracteres que cumplen una función específica en la programación del mismo.

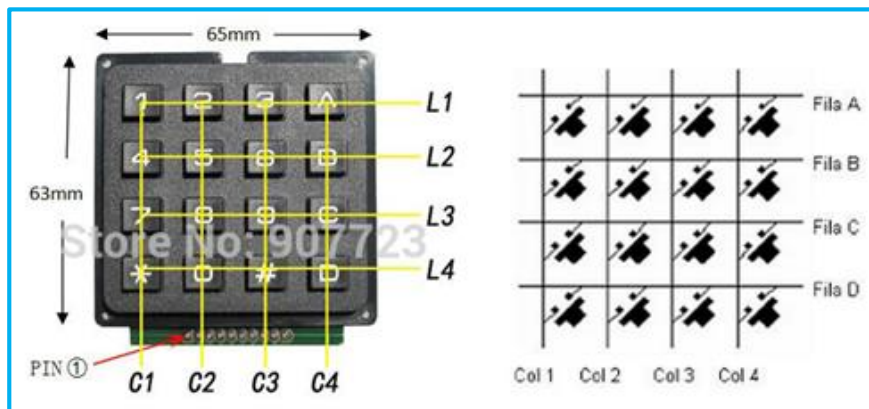


Figura 4.9: Teclado matricial 4x4 configuración interna

Fuente: Libro Microcontroladores PIC [24]

Cuando el número de teclas es pequeño (inferior a 16), éstas se pueden conectar directamente de forma independiente, pero al tener mayor cantidad de teclas es necesario conectar mediante una matriz de hilos conductores. [24]

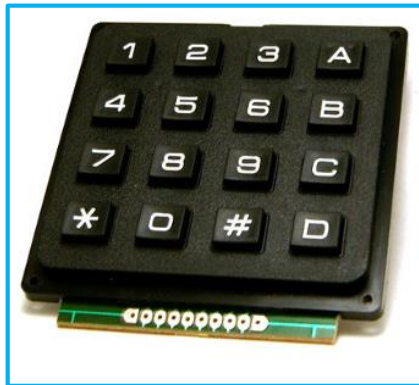


Figura 4.10: Teclado matricial 4x4 estructura externa
Fuente: Libro Microcontroladores PIC [24]

4.3.4. Componentes Electrónicos

Altavoz: Transductor electroacústica utilizado para la reproducción de sonido, que convierte la onda eléctrica en energía mecánica y esta a su vez en ondas de frecuencia acústica. Su impedancia es inversamente proporcional a la potencia de trabajo.



Figura 4.11: Altavoz de 8 ohm
Fuente: <https://djmusica8sanvicente.wordpress.com/2012/01/16/que-son-los-altavoces/>

Resistencias: Dispositivos semiconductores que impide el paso de la corriente, disipa energía en forma de calor. Elemento electrónico pasivo, de acuerdo a la ley de Ohm la resistencia de un material se define como la razón entre la diferencia de potencial eléctrico y la corriente que atraviesa dicha resistencia.



Figura 4.12: Estructura física de una resistencia eléctrica
Fuente: <http://spainlabs.com/foro/viewtopic.php?t=2873&p=49384>

Diodo led: Dispositivo semiconductor. El led es un tipo especial de diodo que trabaja como un diodo normal, pero al ser atravesado por una corriente eléctrica emite luz. Parámetros técnicos señalan un voltaje ideal de operación de 1,4V hasta 3,8V aproximadamente y una corriente de funcionamiento que varía de acuerdo al color generalmente rojos de 10 y 20mA y demás de 40mA.

Su bajo consumo de energía, mantenimiento casi nulo y su extensa vida útil, son características que detallan a un elemento electrónico muy funcional.



Figura 4.13: Diodo led
Fuente: <http://robonet1.blogspot.com/2015/03/led-diodo-emisor-de-luz.html>

Relevador: Relé o relevador es un dispositivo electromecánico que funciona como un interruptor controlado por un circuito eléctrico, constituido por una bobina y un electroimán que accionan un juego de contactos permitiendo abrir o cerrar circuitos eléctricos independientes.

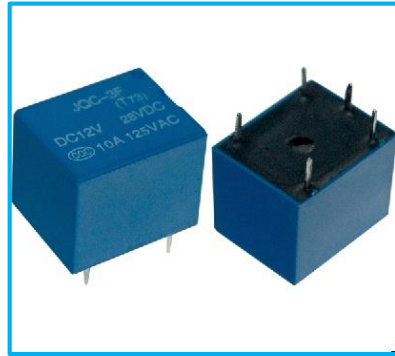


Figura 4.14: Relevador 12 VDC 10A

Fuente: <http://www.taringa.net/posts/info/5800372/Funcionamiento-del-rele.html>

Para el proyecto se utilizó relés de 12 VDC, a continuación sus principales características técnicas:

- Fabricante Techman
- Voltaje operación 12VDC.
- Voltaje funcionamiento DC 28V terminales.
- Voltaje funcionamiento AC 125V terminales.
- Corriente máxima terminal 10A.
- Corrientes de funcionamiento 30mA.
- Resistencia bobina 400 ohm.
- Tiempo de acción 0,10mSeg. [25]

Diodo 1N4007: Dispositivo electrónico de dos terminales que permite la circulación de la corriente en un solo sentido. Diodo semiconductor que por debajo de cierta diferencia de potencial se comporta como un circuito abierto (no existe conducción de corriente) y por encima de ella actúa como un circuito cerrado (conducción de corriente) con una resistencia eléctrica muy pequeña. Elemento compuesto de silicio cuyo voltaje de funcionamiento es de 0,7V, utilizado principalmente como rectificador de señal.



Figura 4.15: Diodo 1N4007

Fuente: <http://www.electronicoscaldas.com/diodos-y-puentes/67-diodo-1n4007.html>

Características principales:

- Baja caída de tensión directa.
- Capacidad de alta corriente.
- Alta fiabilidad.
- Intensidad que soporta 1A.
- Voltaje máximo 100V. [26]

Optoacoplador 4N25: Dispositivo de emisión y recepción que funciona como un interruptor activo mediante la emisión de luz de un diodo led que satura un componente optoelectrónico comúnmente un fototransistor o fototriac. Ambos elementos electrónicos se combinan formando un solo dispositivo semiconductor dentro de un encapsulado con un fotoemisor y un fotoreceptor cuya relación es óptima. Dispositivo ideal para trabajar circuitos que funcionan con fuentes de voltaje y corriente diferentes.



Figura 4.16: Estructura física Optoacoplador 4N25

Fuente: <http://es.farnell.com/avago-technologies/4n25-360e/optocoupler-transistor-o-p-vde/dp/1244502>

Características principales:

- Salida a fototransistor.
- Voltaje funcionamiento 1,2V.
- Voltaje funcionamiento máx 1,5V.
- Voltaje de aislamiento 2500Vrms.
- Corriente Funcionamiento 10mA.
- Corriente directo led máx 80mA.
- Voltaje inverso led máx 6V.
- Voltaje colector-emisor máx 30V.
- Corriente del colector máx 100mA.
- Tiempo de respuesta 3uSeg. [27]

Transistor 2N3904: Dispositivo semiconductor que entrega una señal de salida en respuesta a una señal de entrada. De término inglés “*transfer resistor*” resistor de transferencia, constituye el elemento principal en todos los aparatos electrónicos. Cumple funciones de amplificador, oscilador, conmutador o rectificador.

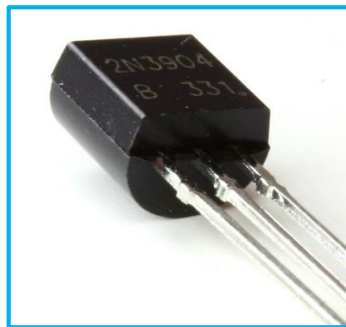


Figura 4.17: Transistor 2N3904

Fuente: <http://www.oddwires.com/2n3904-npn-general-purpose-transistor-30-pack/>

Características principales:

- Juntura NPN.
- Voltaje colector-emisor 40V.
- Voltaje funcionamiento 0,7V.

- Corriente operación 10mA.
- Corriente de colector Max 200mA.
- Potencia total disipada 625mW.
- Frecuencia de trabajo 300 MHz.
- Voltaje máximo 35V. [28]

4.3.5. Cálculo Componentes Electrónicos

El acumulador o batería del vehículo es de 12V teórico, y a través de mediciones se obtuvo una tensión de 12,8V con el motor apagado y 14,4V con el motor encendido. Para el cálculo de los valores de los elementos se trabajó con la media de dichos valores anteriormente mencionados, obteniendo como resultado 13,6V.

La corriente indispensable para el funcionamiento del prototipo electrónico es proporcionada dependiendo de la necesidad del circuito por parte del acumulador. Tomamos en consideración la corriente de funcionamiento de la tarjeta Arduino que es de 40mA.

En la Figura 4.18 se muestra el circuito regulador de 12V a 5V para la alimentación del Módulo Central, constituido por la placa Arduino Mega 2560 y la placa EasyVR, cuyos valores de sus componentes se obtuvieron a través de parámetros establecidos y valores nominales.

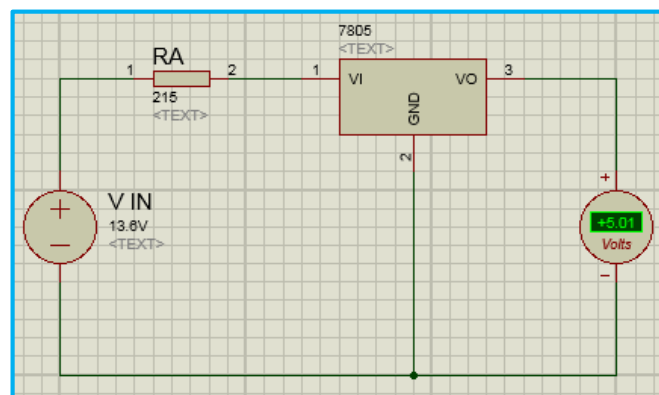


Figura 4.18: Circuito regulador de voltaje
Fuente: El investigador

Cálculo de los componentes electrónicos:

DATOS:

V Motor Apagado = 12,8 V

V Motor Prendido = 14,4 V

V in = 13,6 V

V out = 5V

I funcionamiento Arduino = 40mA

$$\begin{aligned}V_{Disipado} &= V_{in} - V_{out} \\V_{Disipado} &= 13,6V - 5V \\V_{Disipado} &= 8,6V\end{aligned}$$

$$V_{RA} = V_{Disipado}$$

$I_{in} \approx I_{funcionamiento\ Arduino}$

$$R_A = \frac{V_{RA}}{I_{Arduino}} = \frac{8,6V}{40mA} = 215\Omega$$

$$215\Omega \approx 220\Omega$$

Resistencia utilizada $R_A = 220\Omega$

La Figura 4.19 trata del circuito de potencia básico para la conexión de los dispositivos y circuitos eléctricos a intervenir en el vehículo, cuyos valores de sus componentes se obtuvieron a través de parámetros establecidos y valores nominales.

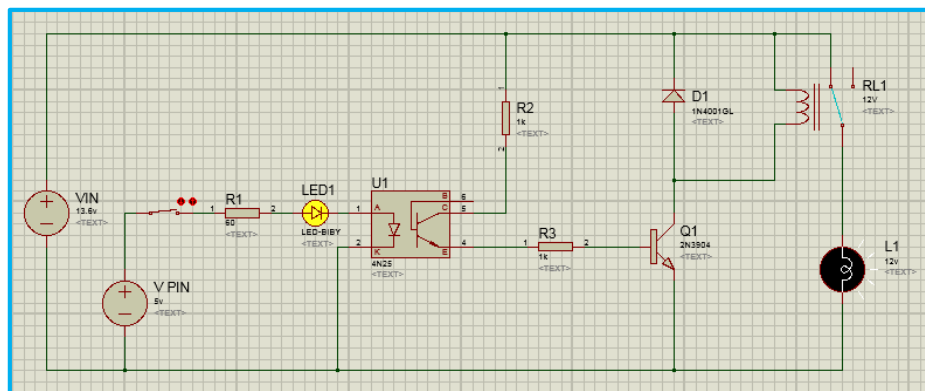


Figura 4.19: Circuito de potencia básico
Fuente: El investigador

Cálculo de los componentes electrónicos:

DATOS:

$$V_{in} = 13,6V$$

$$V_{pin} = 5V$$

$$I_{pin} = 40mA$$

$$V_{funcionamiento\ led} = 1,4V$$

$$I_{funcionamiento\ transistor} = 10mA$$

$$I_{funcionamiento\ Opto} = 10mA$$

$$V_{funcionamiento\ Opto} = 1,2V$$

$$R_{Bobina\ Relé} = 400\Omega$$

$$I_{funcionamiento\ Relé} = 30mA$$

$$VR1 = V_{in} - V_{led} - V_{opto}$$

$$VR1 = 5V - 1,4V - 1,2V$$

$$VR1 = 2,4V$$

$$R1 = \frac{VR1}{I_{pin}} = \frac{2,4V}{40mA} = 60\Omega$$

Resistencia utilizada R1 = 100Ω

R1 = Resistencia de protección Optoacoplador

$$I_{relé} = \frac{V_{in}}{R_{Bobina}} = \frac{13,6V}{400\Omega} = 0,034A$$

$$I_{relé} = 34mA$$

Irelé > I funcionamiento relé

$$34mA > 30mA$$

R2 = Resistencia de protección Optotransistor

$$R2 = 1K\Omega$$

$$IR2 = \frac{V_{in}}{R2} = \frac{13,6V}{1000\Omega} = 13,6mA$$

R2 limita la corriente necesaria para la operabilidad de Transistor

$$IR2 = IR3$$

$$R3 = \frac{Vin}{IR3} = \frac{13,6V}{13,6mA} = 1K\Omega$$

R3 = Resistencia de protección Base Transistor

IR3 > I funcionamiento Transistor

$$13,6mA > 10mA$$

4.4. SOFTWARE

4.4.1. Arduino IDE

Arduino IDE es una plataforma de código abierto escrita en Java utilizada para proyectos de Electrónica, cuyo lenguaje de programación es Processing de alto nivel, que se basa en C y C++. El software es de multiplataforma y se puede ejecutar en los sistemas operativos Windows, Linux, Mac OS X.

Tiene la capacidad de importar en los proyecto bibliotecas que disminuyen las líneas de código, dispone de un editor de texto para el código, un área de mensajes, una consola de texto, barra de herramientas con funciones principales, menús de aplicaciones, ejemplos y otros; que permiten al desarrollador disponer de un entorno amigable y sencillo para la programación del Sketch que es el proyecto en sí a crear. [29]

En la Figura 4.20 se muestra el entorno de desarrollo del software Arduino IDE, ejecutando una simulación básica de la aplicación Blink, que se encuentra en la sección de ejemplos, precargados por el programa.

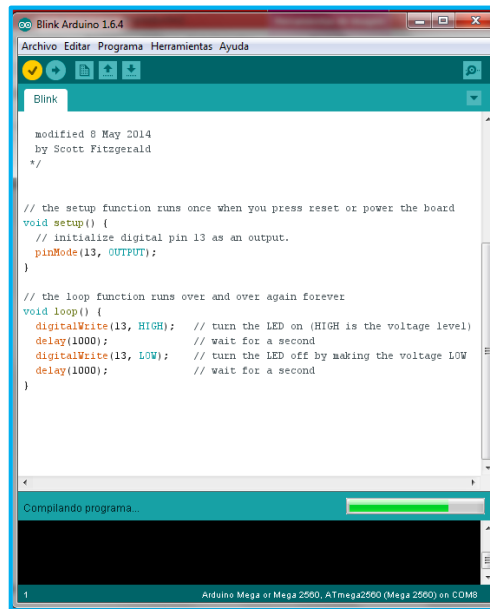


Figura 4.20: Editor de código Arduino IDE
Fuente: El investigador

4.4.2. Balabolka

Es un software de conversión de texto a voz, versión para sistema operativo Windows XP en adelante, que permite guardar el texto leído como un archivo en formato WAP, MP3, MP4, OGG o WMA. Permite modificar los parámetros de voces, velocidad, tono para mejorar la pronunciación en una lectura.

El programa es de licencia gratuita, soporta varios idiomas, permite locutar el texto de un documento en formato DOC, HTML, PDF, entre otros; y a través de su editor de texto puede escribir y de esa manera crear sus propios archivos de sonido. [30]

En la Figura 4.21 se identifica la ventana de trabajo del software, con las opciones de modificación de velocidad, volumen y la parte del editor donde se ingresa el texto que va a ser reproducido en audio.

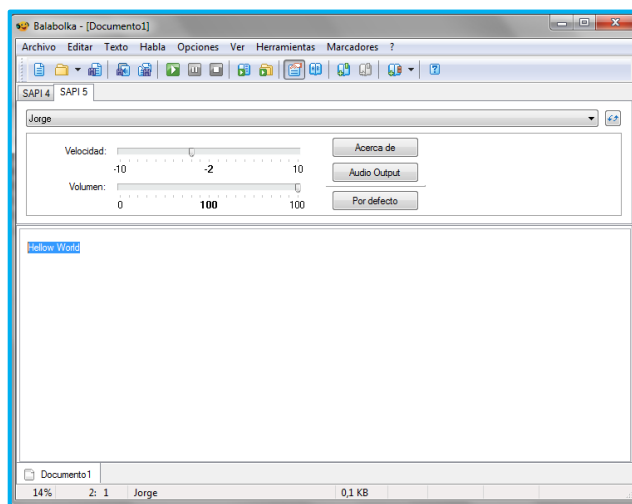


Figura 4.21: Editor de texto Balabolka Software
Fuente: El investigador

4.4.3. EasyVR Commander

Software diseñado para programar y controlar al módulo EasyVR, que permite añadir la capacidad de reconocimiento de voz a través de sonidos incorporados o la opción de nuevos comandos, que se ejecuta en 6 distintos idiomas. EasyVR Commander permite crear y personalizar conjuntos de comandos, entrenar al módulo con la función de reconocer su voz, y reproducir tabla de sonidos creados en el software Sensory QuickSynthesis 5 con una duración de hasta 9 minutos de audio.

En su entorno se dispone de una lista de grupos de comandos, lista de SoundTable, cuadro para la escritura de comandos en un grupo específico, menú principal con las opciones necesarias para crear un comando, barra de herramientas para configurar la tarjeta EasyVR.[31]

En la Figura 4.22 muestra la ventana del software, donde se puede identificar en la parte superior las opciones de programación, en su costado izquierdo las listas de comandos que se despliegan cada uno de ellos a su derecha, dependiendo del grupo de trabajo en que se encuentren.

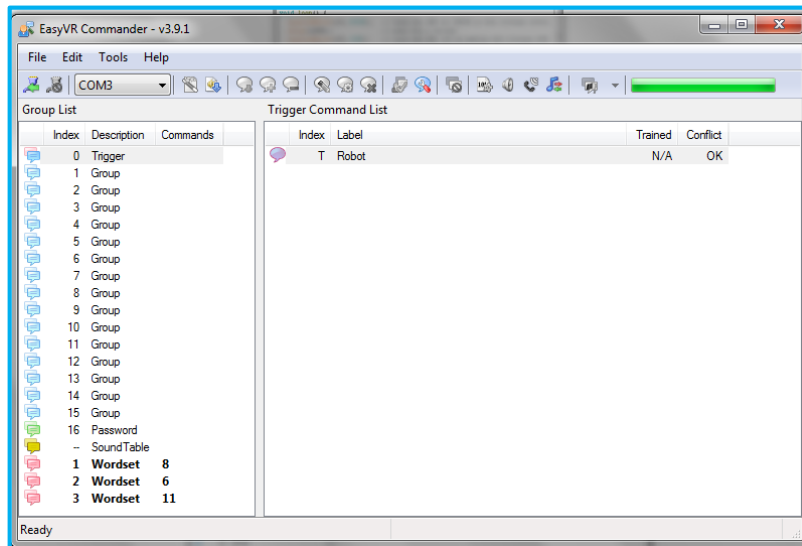


Figura 4.22: Software EasyVR Commander

Fuente: El investigador

4.4.4. Sensory QuickSynthesis 5

Es un software que se instala con EasyVR Commander, herramienta para crear y gestionar expresión y síntesis de sonido para aplicaciones de microcontroladores, que permite a los desarrolladores utilizar una variedad de tecnologías de comprensión de datos con una amplia gama de velocidades de transferencia.

La aplicación de simulación permite escuchar la tabla de audio, antes de crear el fichero de sonido y comprimir los datos de información. En su entorno dispone de un cuadro para la lista de sonido, cuadro de dialogo de transferencia, opción de volumen, velocidad de transmisión y menú principal de mando. [32]

La Figura 4.23 corresponde a la ventana principal del software, con sus respectivos cuadros de trabajo.

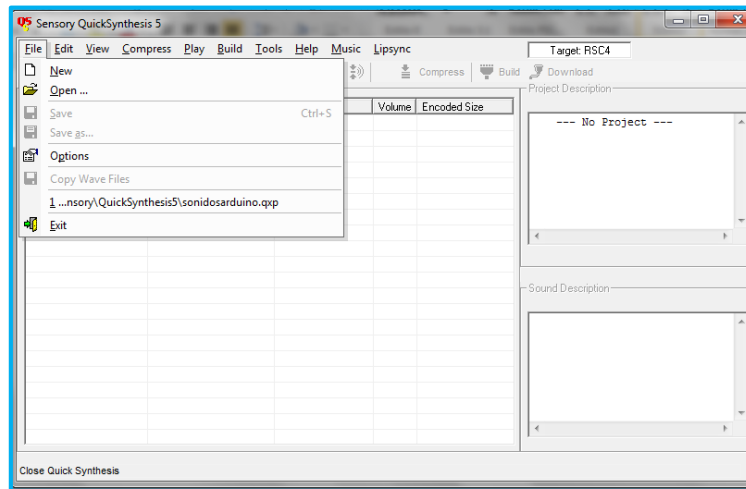


Figura 4.23: Cuadro de diálogo Sensory QuickSynthesis 5

Fuente: El investigador

4.4.5. Proteus

Desarrollado por Labcenter Electronics es una compilación de programas utilizados para el diseño de esquemas eléctricos y simulación de circuitos, con completos paquetes de elementos electrónicos, actuadores, interruptores, etc.; dos programas principales que son Isis y Ares, que en conjunto forman una potente herramienta en el entorno electrónico.

El programa Isis o Sistema de Enrutamiento de Esquemas Inteligentes, sirve para el diseño del plano eléctrico del circuito que se desea realizar que utiliza una extensa gama de componentes y a través del módulo VSM puede simular en tiempo real.

El programa Ares o Software de Edición Y Ruteo Avanzado, sirve para la fabricación de placas de circuito impreso que permite editar la capa superficial (Top Copper) y de soldadura (Botton Copper) para generar pistas. [33]

La Figura 4.24 es la imagen principal de carga al momento de ejecutar el software Proteus en su versión Design Suite 8.1 en un ordenador con sistema operativo Windows XP en adelante.



Figura 4.24: Portada Software Proteus
Fuente: El investigador

4.5. ESTRUCTURA

En la parte de hardware, disponemos de tres módulos electrónicos, el Módulo Central que es la parte principal del sistema, el Módulo de Control encargado de la interacción con los dispositivos del vehículo y la Interfaz de Usuario por medio del cual se controla y visualiza el funcionamiento del sistema. La figura 4.25 describe de una mejor manera la estructura general del Sistema Electrónico:

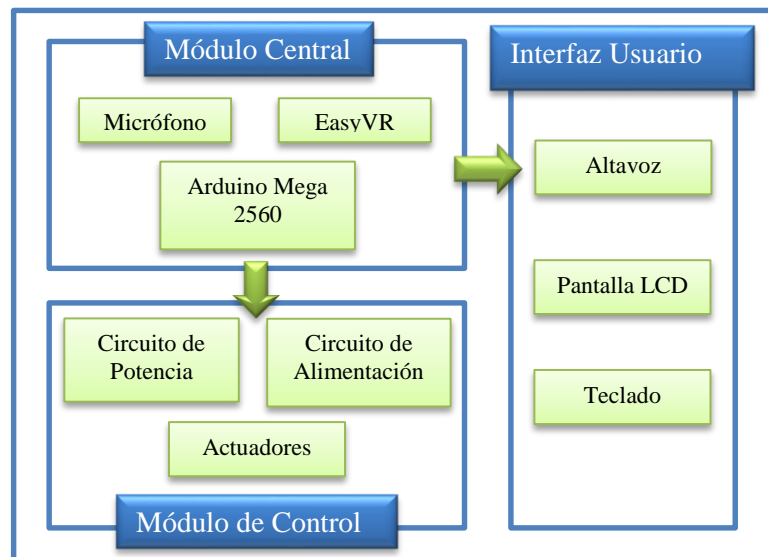


Figura 4.25: Esquema gráfico del prototipo electrónico
Fuente: El investigador

4.5.1. Módulo Central

Este módulo es la parte más importante de Sistema Electrónico que controla, gestiona y ejecuta las diferentes órdenes por parte del usuario emitidas por comandos de voz. Está constituido por la placa Arduino Mega 2560, que contiene toda la programación del sistema, adicionalmente utiliza sus pines digitales como salidas que proporcionaran 5V y 40mA para interactuar con el módulo de control.

La tarjeta EasyVR es el dispositivo que reconoce una señal audible a través de su micrófono omnidireccional, procesa y codifica dicha información para posteriormente entregar a la placa Arduino Mega y ejecutar las órdenes correspondientes. En la figura 4.26 se muestra el esquema básico del Módulo Central:

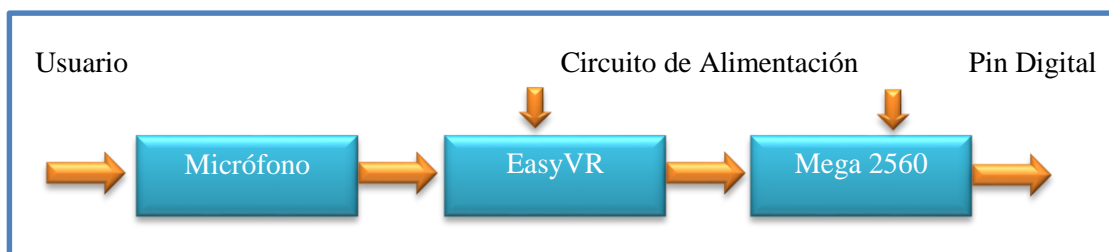


Figura 4.26: Diagrama del módulo central
Fuente: El investigador

El software EasyVR Commander se utiliza para programar la tarjeta EasyVR donde se creó los diferentes comandos que cumplen una acción específica en el código de programación de la tarjeta Arduino Mega 2560. En la Figura 4.27 se identifica el Grupo 1 con 23 comandos de voz que al ser pronunciados por parte del usuario controlarán cada uno de los sistemas y accesorios intervenidos en el vehículo.

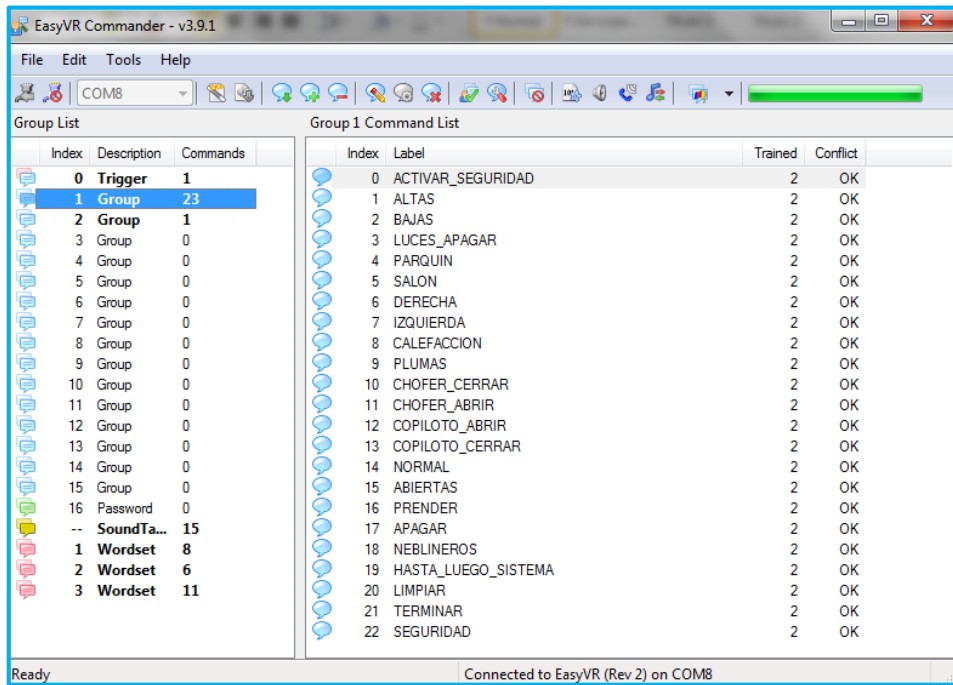


Figura 4.27: Lista de comandos EasyVR
Fuente: El investigador

4.5.2. Módulo de Control

En la Figura 4.28 describe al Módulo de Control constituido por los circuitos de alimentación y de potencia, necesarios para el accionamiento de los actuadores y dispositivos finales.

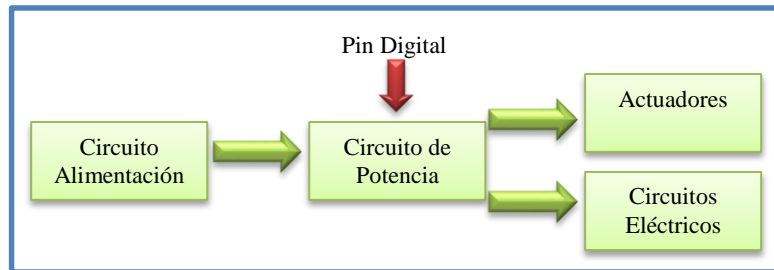


Figura 4.28: Diagrama del módulo de control
Fuente: El investigador

El circuito de potencia es el encargado de la interacción entre ambos módulos al trabajar cada uno de ellos con voltajes y corrientes diferentes, siendo imposible la conexión directa con el riesgo absoluto de daño a la placa Arduino y la tarjeta de reconocimiento de voz. Los optoacopladores permiten la interacción con el Módulo de Control sin la necesidad de una conexión directa al ser esa su funcionalidad, y a través de sus diodos excitan a sus autotransistores permitiendo la circulación de corriente proveniente del acumulador para polarizar un transistor que actúa como interruptor para el accionamiento de los relevadores.

El circuito de Alimentación regula el voltaje y corriente necesaria para el Módulo Central y además tiene dispositivos de protección que evitan el daño de las tarjetas por corto circuitos o sobre cargas de corriente.

Los actuadores son dispositivos finales de control y en nuestro caso se utilizó relevadores comúnmente conocidos como relés, elementos electromecánicos que soportan grandes cantidades de voltaje y corriente.

4.5.3. Interfaz de Usuario

Se considera una interfaz el medio que permite comprender y controlar señales generadas por algún tipo de dispositivo a través de dispositivos sencillos. El prototipo electrónico dispone de un teclado matricial 4x4 alfanumérico, ideal para el ingreso de contraseñas, que constituye un medio auxiliar para desactivar el sistema de Seguridad.

Para interpretar cada proceso ejecutado por el sistema se dispone de una pantalla GLCD 128x64 pixeles, donde se visualiza de una manera agradable líneas de información con el fin de entender el proceso de ejecución del mismo.

La tarjeta EasyVR tiene la opción de incorporar un altavoz de 8 ohm de salida, que en función del proceso realizado por parte del sistema reproduce sonidos que difunden información y alerta al usuario.

El software Balabolka se utilizó para crear archivos de audio con formato WAP, y por medio del programa Sensory QuickSynthesis 5 se crea tablas de sonidos que son cargadas por el programa EasyVR Commander en la tarjeta de reconocimiento de voz EasyVR. En la figura 4.29 se muestra la lista de reproducción creada para el prototipo electrónico.

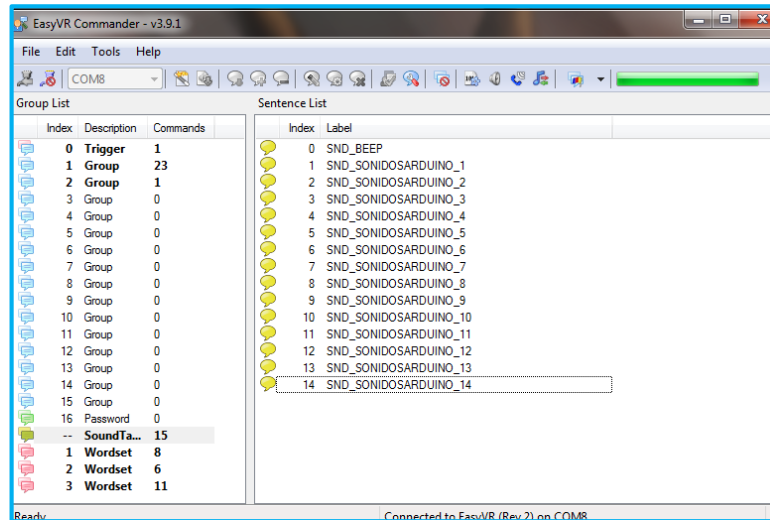


Figura 4.29: Lista de reproducción SoundTable
Fuente: El investigador

4.6. CARACTERÍSTICAS DEL SISTEMA

El sistema Embebido controla diferentes dispositivos del vehículo a través de comandos de voz con la intención de mejorar la seguridad, concentración y maniobrabilidad al momento de conducir. De acuerdo a las necesidades presentes por parte del usuario, se ha tomado en cuenta los siguientes sistemas del vehículo para ser automatizados:

- Sistema de Iluminación
- Sistema de Accesorios primarios
- Sistema de Seguridad

En el Sistema de Iluminación se trabajó con su mayor parte, con los subsistemas de luces altas, luces bajas, luces de emergencia, luces auxiliares neblineros, iluminación de salón y luces de dirección de vía.

El encendido del aire acondicionado, limpiaparabrisas y el control de ventanillas del conductor y copiloto se ejecutarán sin la intervención manual controlados por comandos de voz.

El paso de corriente a los inyectores del vehículo es ideal para impedir el encendido ilícito de un automotor, que al trabajar en conjunto con la alarma se mejoró el sistema de seguridad, que impide el encendido forzado e ilícito y bloquea al automotor para su protección.

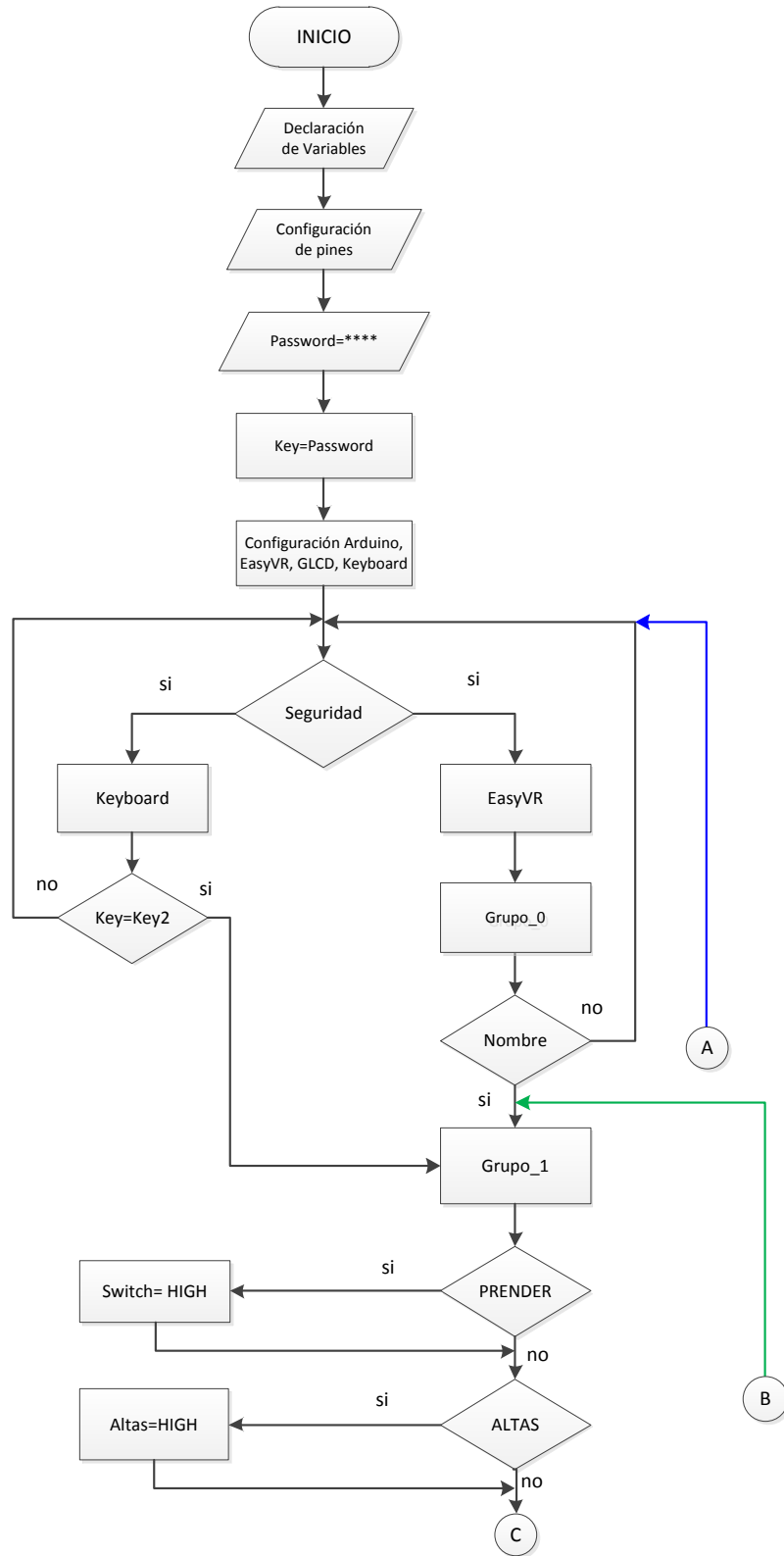
4.7. PROGRAMACIÓN

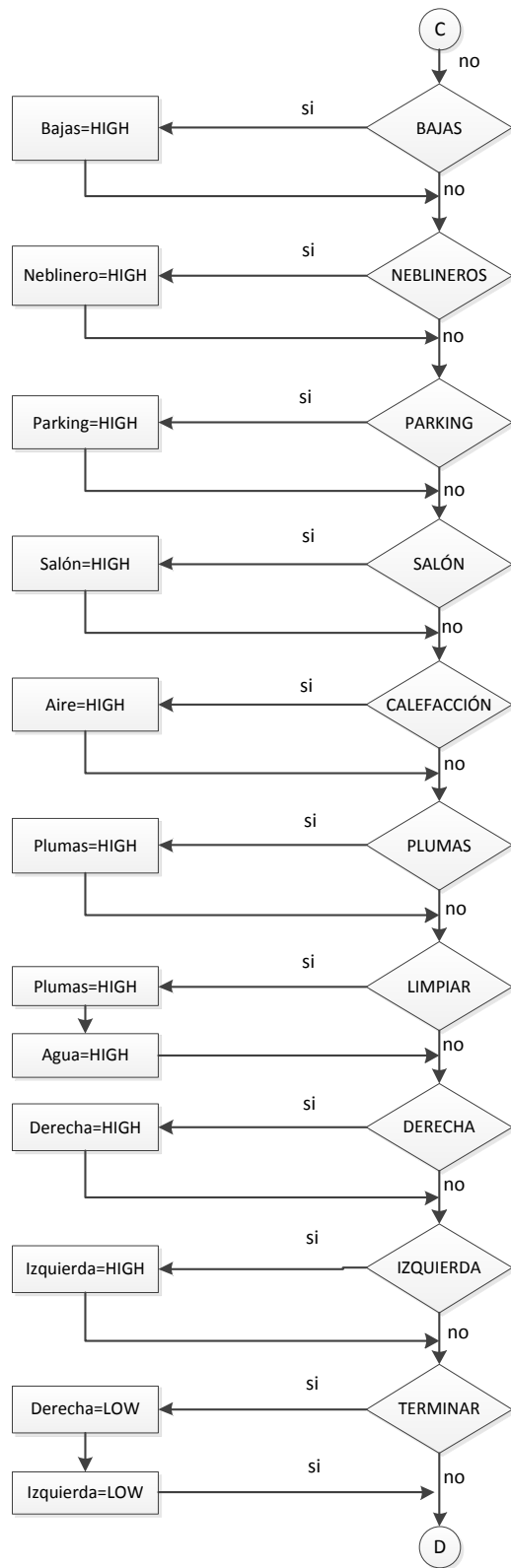
La programación del Sistema Electrónico se realizó en la plataforma de código abierto Arduino que además de disponer del software gratuito posee una variedad de dispositivos físicos, flexibles y compatibles entre sí. En la red se dispone de una gran cantidad de información del lenguaje de programación Processing.

La tarjeta EasyVR es totalmente compatible con el entorno Arduino, razón por la cual se realizó en esta plataforma a más de ser un software libre sin ningún tipo de restricción; para el intercambio de datos se utilizó una comunicación serial, no obstante se la puede realizar en paralela o dedicada por puertos específicos.

En el **Anexo II** se encuentra la programación del sistema, para el estudio y análisis del funcionamiento general del prototipo electrónico.

A continuación se muestra el análisis de la programación en un diagrama de flujos:





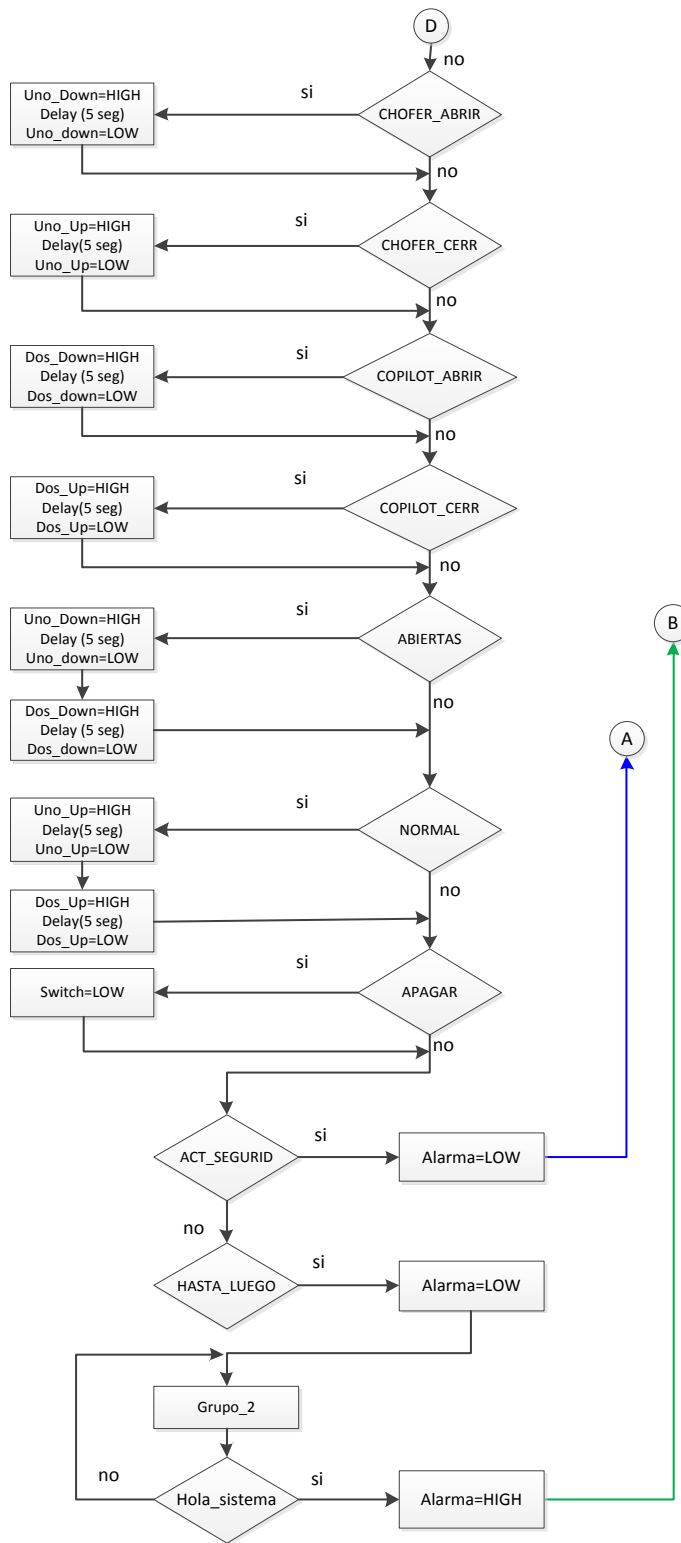


Figura 4.30: Diagrama de flujo Prototipo Electrónico
Fuente: El investigador

4.8. COMANDOS DEL SISTEMA

De acuerdo a la aplicación a realizar por parte del prototipo, se ha desarrollado una lista de comandos que se encuentran divididos en 3 grupos; Trigger donde se crea el comando contraseña, Grupo 1 y Grupo 2 dependiendo de la utilidad.

En la tabla 4.5 se enumera cada uno de ellos en su grupo correspondiente:

Grupo	Item	Nombre
Trigger	0	NOMBRE
Grupo 1	0	ACTIVAR_SEGURIDAD
	1	PRENDER
	2	APAGAR
	3	ALTAS
	4	BAJAS
	5	LUCES_FUERA
	6	NEBLINEROS
	7	PARKING
	8	SALÓN
	9	CALEFACCIÓN
	10	PLUMAS
	11	LIMPIAR
	12	DERECHA
	13	IZQUIERDA
	14	TERMINAR
	15	CHOFER_ABAJO
	16	CHOFER_ARRIBA
	17	COPILOTO_ABAJO
	18	COPILOTO_ARRIBA
	19	ABIERTAS
	20	NORMAL
21	HASTA_LUEGO_SISTEMA	
Grupo 2	0	HOLA_SISTEMA

Tabla 4.5: Comandos programados en el EasyVR

Fuente: El investigador

A continuación se describe la función a realizar por parte de cada uno de los comandos:

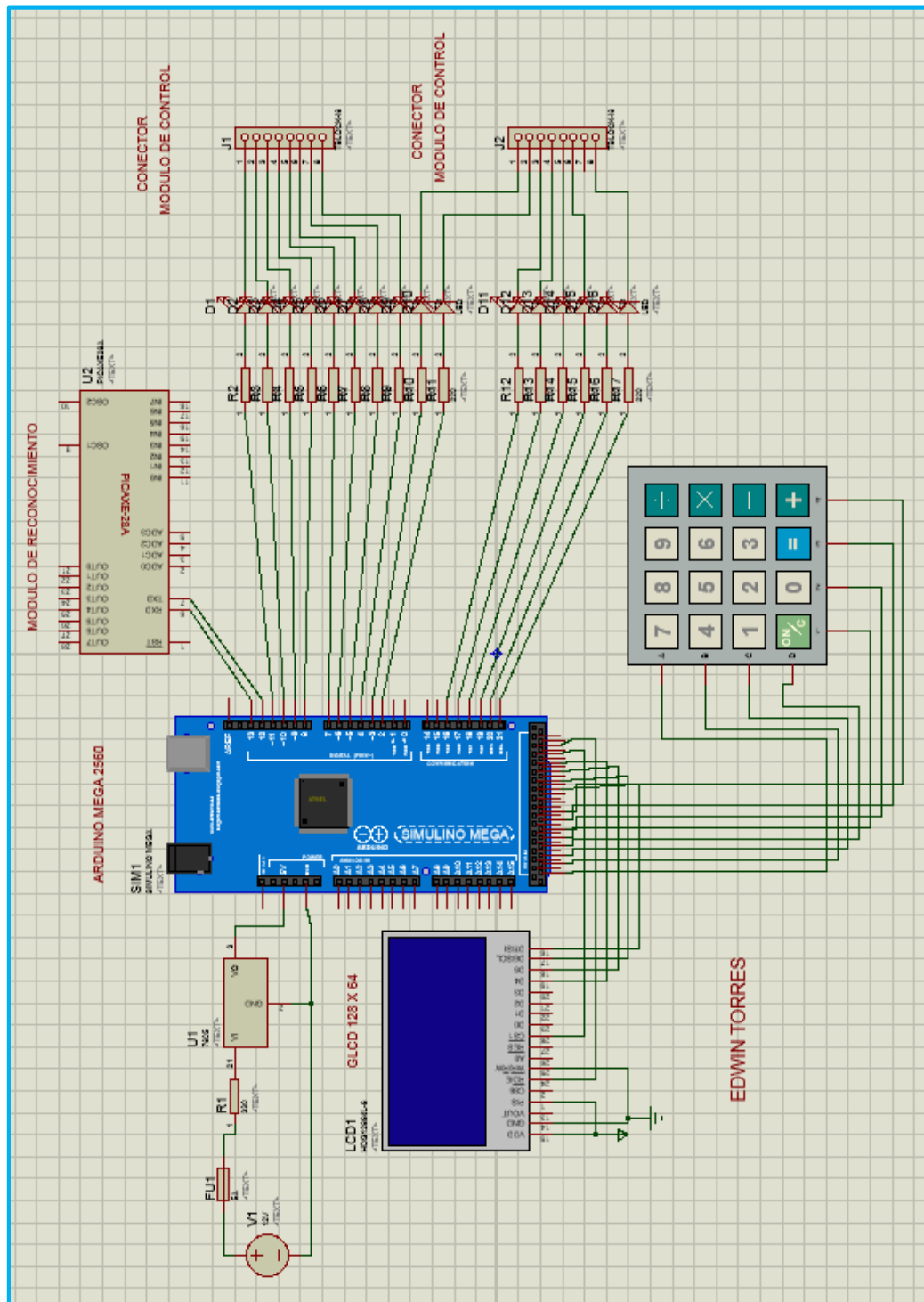
- NOMBRE: Contraseña para desactivar el sistema de Seguridad.
- ACTIVAR_SEGURIDAD: Activa Sistema de Seguridad.
- PRENDER: Prende el vehículo.
- APAGAR: Apaga el vehículo.
- ALTAS: Enciende las luces altas.
- BAJAS: Enciende las luces bajas.
- LUCES_FUERA: Apaga luces altas y bajas.
- NEBLINEROS: Enciende luces auxiliares para neblina.
- PARKING: Enciende luces de emergencia.
- SALÓN: Enciende luces del habitáculo del vehículo.
- CALEFACCIÓN: Activa la calefacción.
- PLUMAS: Activa limpiaparabrisas.
- LIMPIAR: Activa limpiaparabrisas y bomba de agua.
- DERECHA: Enciende luz de dirección de vía derecha.
- IZQUIERDA: Enciende luz de dirección de vía izquierda.
- TERMINAR: Apaga luces de dirección de vía.
- CHOFER_ABAJO: Baja el cristal de la ventana del chofer.
- CHOFER_ARRIBA: Sube el cristal de la ventana del chofer.
- COPILOTO_ABAJO: Baja el cristal de la ventana del copiloto.
- COPILOTO_ARRIBA: Sube el cristal de la ventana del copiloto.
- ABIERTAS: Baja los cristales de las ventanas del chofer y copiloto.
- NORMAL: Sube los cristales de las ventanas del chofer y copiloto.
- HASTA_LUEGO_SISTEMA: Modo Standby prototipo electrónico.
- HOLA_SISTEMA: Despierta prototipo electrónico de Modo Standby.

4.9. SIMULACIONES

La simulación de cada uno de los circuitos se realizó en el software electrónico Proteus, utilizando el programa ISIS, con la opción de simulación en tiempo real, para verificar el correcto funcionamiento de los mismos y evitar futuros inconvenientes.

Se utilizó el programa ARES del software Proteus para el diseño de las placas electrónicas PCB, ya que dispone de completas bibliotecas de elementos electrónicos y nos permite una vista en 3D del resultado final de la placa con la ubicación de cada dispositivo.

A continuación en la Figura 4.31 se muestra la simulación del prototipo electrónico con su Módulo Central, Interfaz de Usuario y conectores para la parte de potencia que son los puertos de entradas para los terminales del Módulo de Control.



EDWIN TORRES

Figura 4.31: Circuito del Sistema Electrónico
Fuente: El investigador

El Módulo de Control se divide en dos circuitos. La Figura 4.32 trata del circuito A, formada por relevadores y elementos electrónicos que controlan el accionamiento de los mismos, comandados por el Módulo Central. Los relés controlan sus respectivos dispositivos y circuitos finales, adicionalmente el circuito regulador de voltaje para la alimentación del Módulo Central.

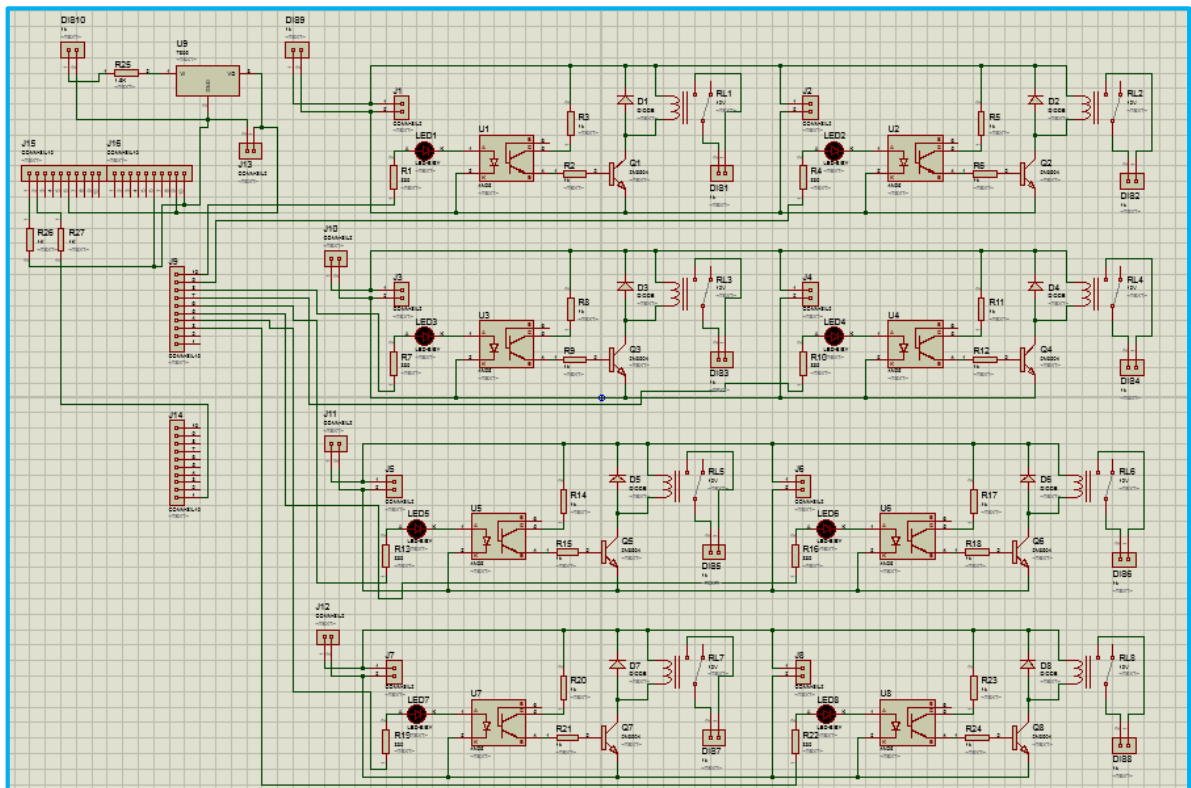


Figura 4.32: ISIS Circuito A
Fuente: El investigador

La Figura 4.33 trata del circuito B, constituido por relevadores y elementos electrónicos que controlan el accionamiento de los mismos, comandados por el Módulo Central.

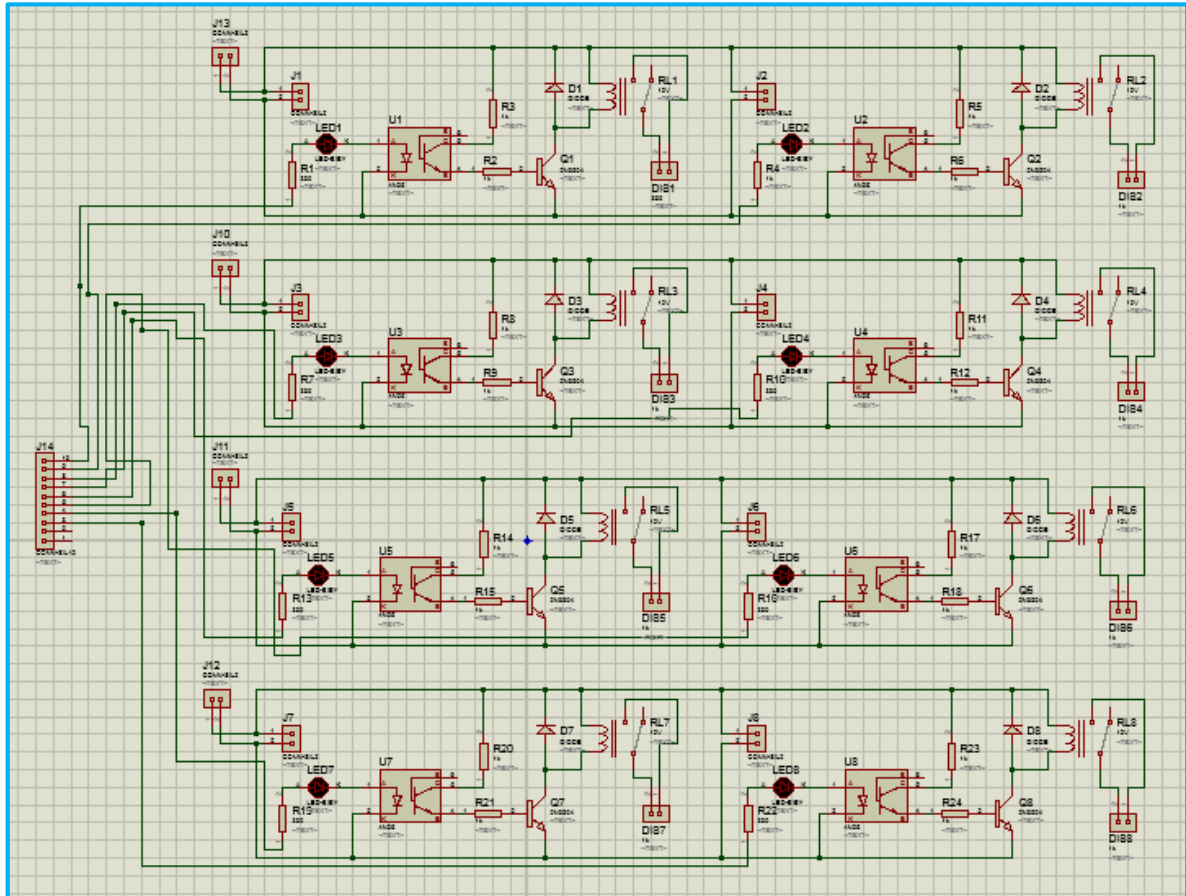


Figura 4.33: ISIS Circuito B
Fuente: El investigador

La Figura 4.34 muestra la placa PCB del circuito A del módulo de Control, donde se puede identificar las pistas de los circuitos electrónicos del prototipo electrónico.

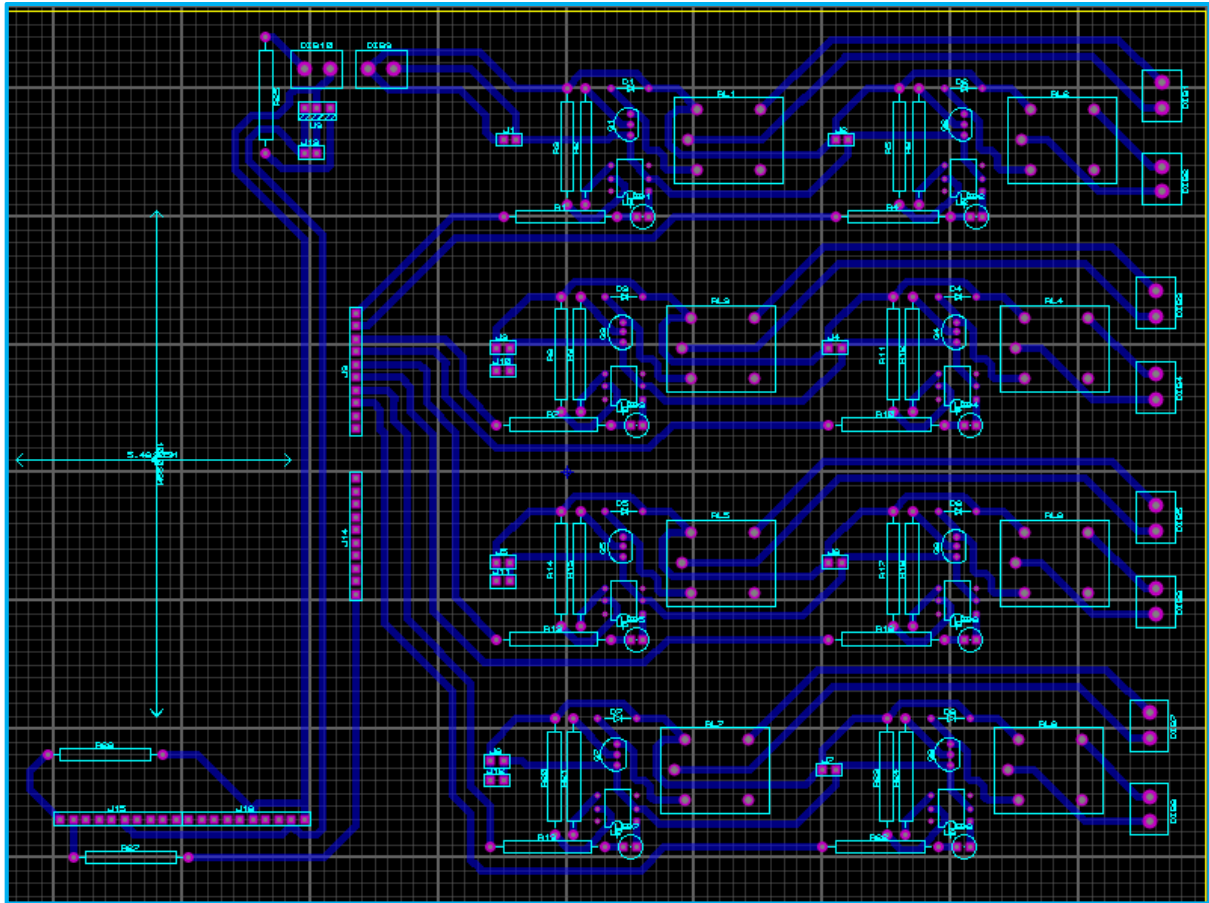


Figura 4.34: ARES Placa A
Fuente: El investigador

La Figura 4.35 muestra la placa PCB del circuito B del módulo de Control, donde se puede identificar las pistas de los circuitos electrónicos del prototipo electrónico.

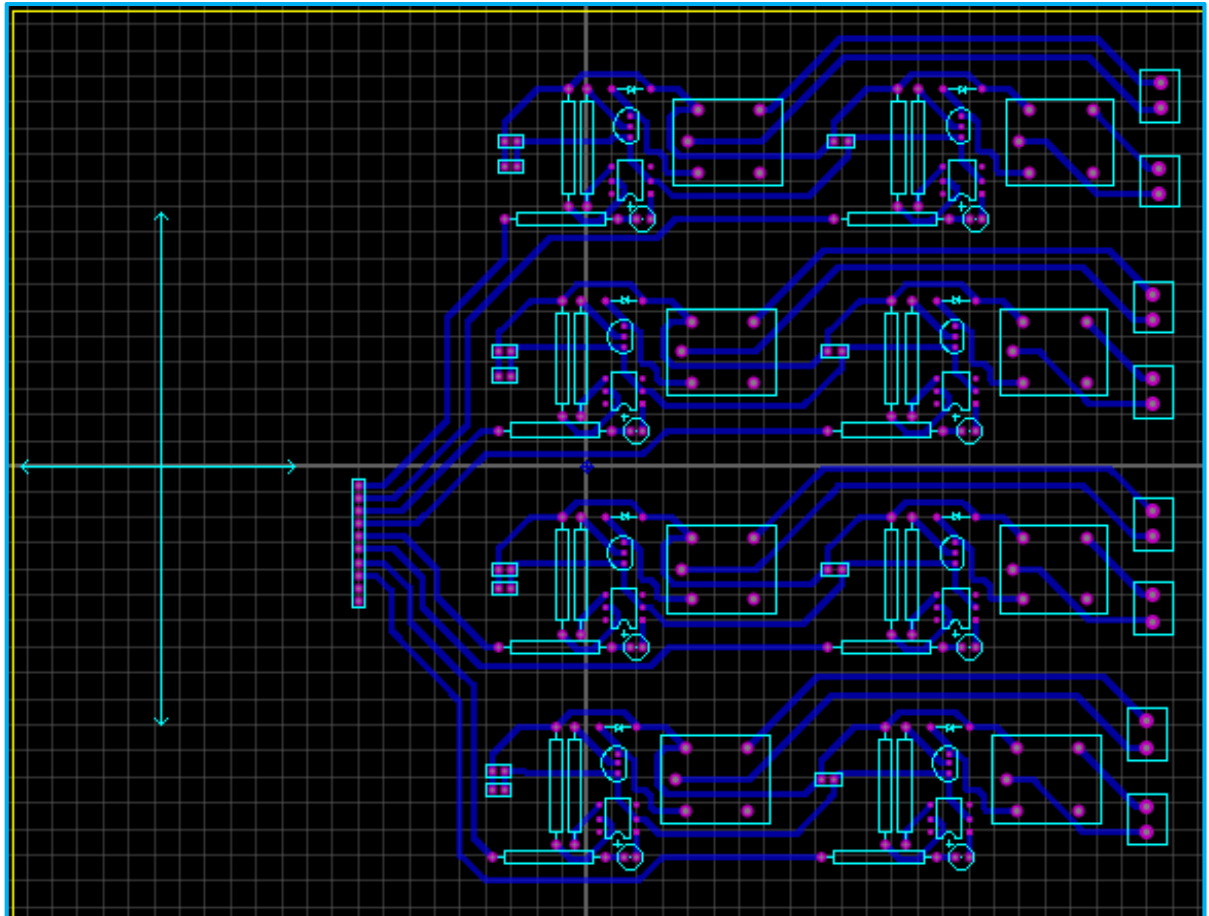


Figura 4.35: ARES Placa B
Fuente: El investigador

La Figura 4.36 es una visualización en 3D en el software Proteus del diseño final de la placa del circuito A, que en conjunto al circuito B constituirán el Módulo del Control.

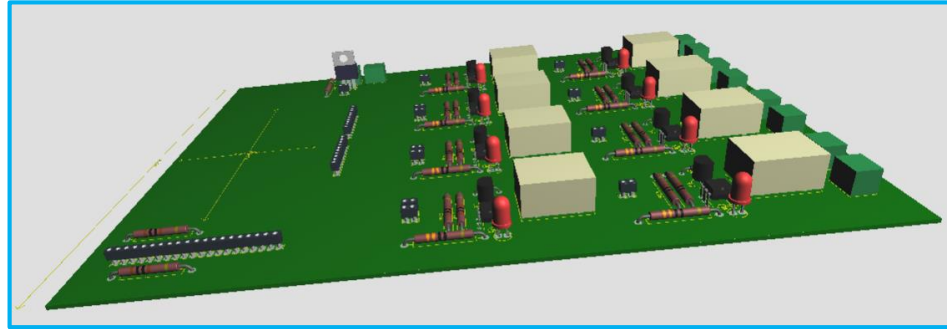


Figura 4.36: Visualización 3D placa A, Módulo de Control
Fuente: El investigador

4.10. DISEÑO DE PLACAS ELECTRÓNICAS

Posterior a la comprobación del funcionamiento a través de las simulaciones en los software respectivos de cada circuito del sistema, se elaboró las diferentes placas electrónicas, para finalizar se ensamblaron todos los módulos y se colocaron en una estructura plástica de material acrílico diseñada de acuerdo a las dimensiones del prototipo para proteger de golpes, sustancias que puedan deteriorar y dañar el sistema disminuyendo su funcionalidad.

En todo el proceso de diseño se realizaron sus respectivas pruebas a priori para descartar fallas y mejorar la eficiencia del sistema embebido. A continuación se describe de una manera más detallada el proceso de diseño de las placas electrónicas.

La Figura 4.37 es una fotografía del proceso de elaboración de pistas de las placas electrónicas, en su proceso de corrosión de cobre por la acción del cloruro férrico.

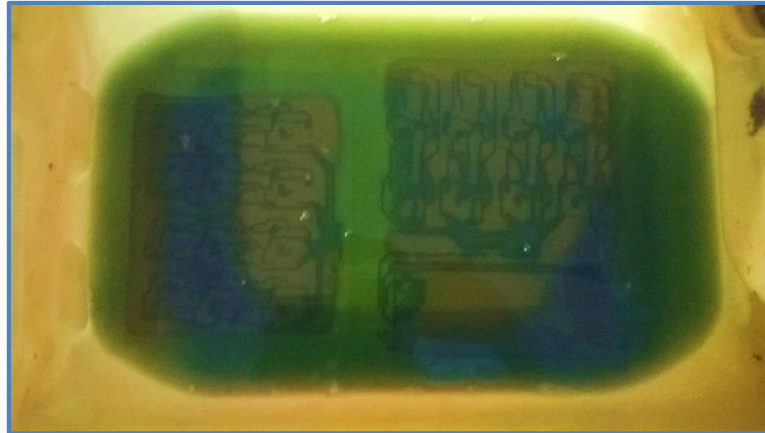


Figura 4.37: Corrosión cobre placas electrónicas
Fuente: El investigador

La Figura 4.38 es una fotografía del resultado final de la elaboración de pistas en las placas, se puede apreciar la perforación de agujeros para la colocación de los elementos electrónicos posterior a la comprobación de continuidad de pistas.

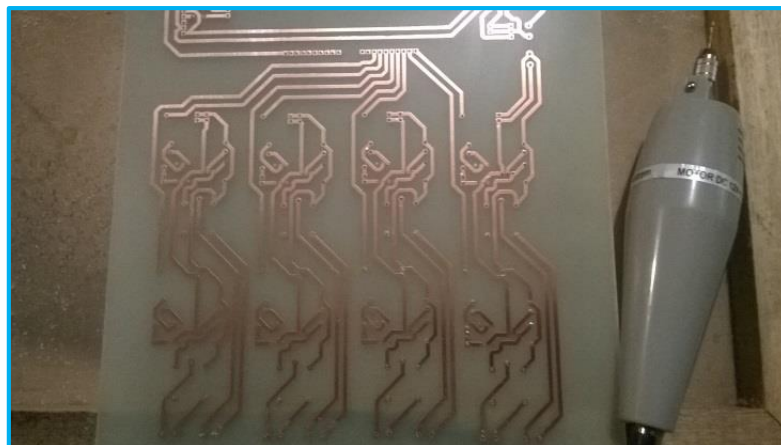


Figura 4.38: Perforación placas electrónicas
Fuente: El investigador

La Figura 4.39 muestra las dos placas electrónicas que conforman el Módulo de Control, con sus respectivos elementos electrónicos.

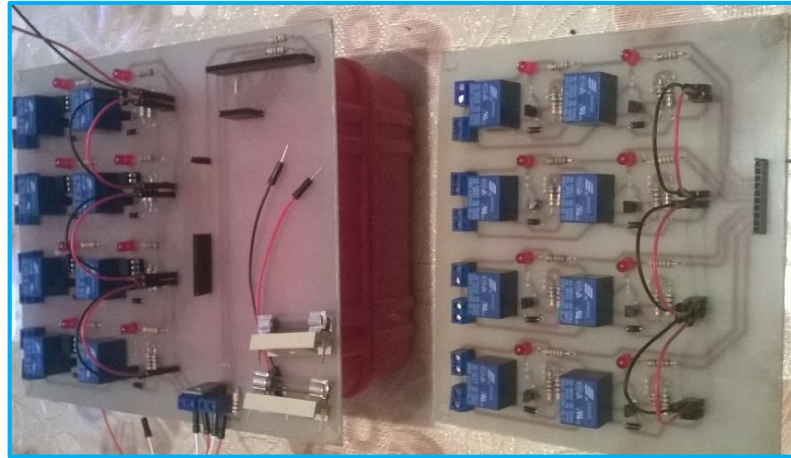


Figura 4.39: Placas electrónicas del Módulo de Control
Fuente: El investigador

La Figura 4.40 es una fotografía de pruebas de funcionamiento del prototipo, antes del ensamblaje final, obteniendo los resultados deseados, comprobando la funcionalidad del sistema.



Figura 4.40: Prototipo electrónico pruebas iniciales
Fuente: El investigador

En la Figura 4.41 se muestra el prototipo electrónico por reconocimiento de voz ensamblado con sus módulos e interfaz de usuario en el interior de la caja plástica de protección.

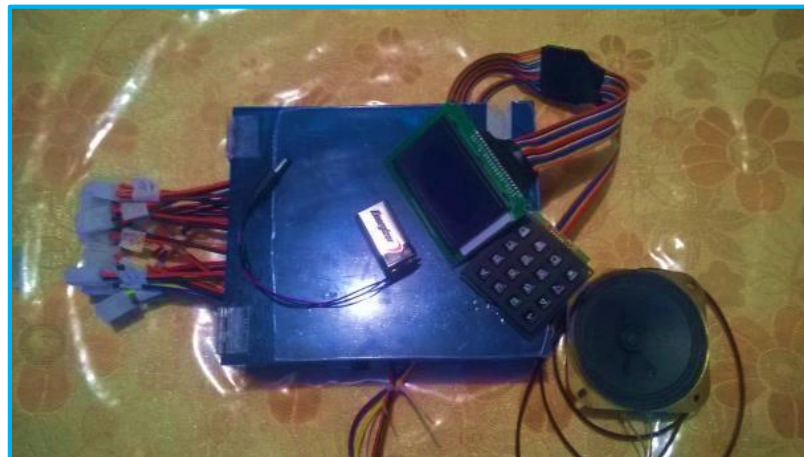


Figura 4.41: Prototipo Electrónico por reconocimiento de voz
Fuente: El investigador

4.11. MONTAJE

Para la adaptación del prototipo se preparó al vehículo organizando el espacio necesario para su colocación, posterior a un análisis de la estructura y disposición del cableado eléctrico. En la Figura 4.42 se muestra el desmontaje de la mascarilla y tablero para realizar las conexiones necesarias en los sistemas y accesorios intervenidos, en donde se utilizó alambre flexible N.18.



Figura 4.42: Desmontaje del tablero del vehículo
Fuente: El investigador

El sistema embebido no tiene que estorbar ni dañar la ergonomía estética del vehículo, motivo por el cual se optó por ubicar al prototipo en el interior de la mascarilla ocupando el espacio de la valija, en el tablero del mismo. La figura 4.43 muestra la ubicación de las conexiones eléctricas que van al prototipo electrónico.

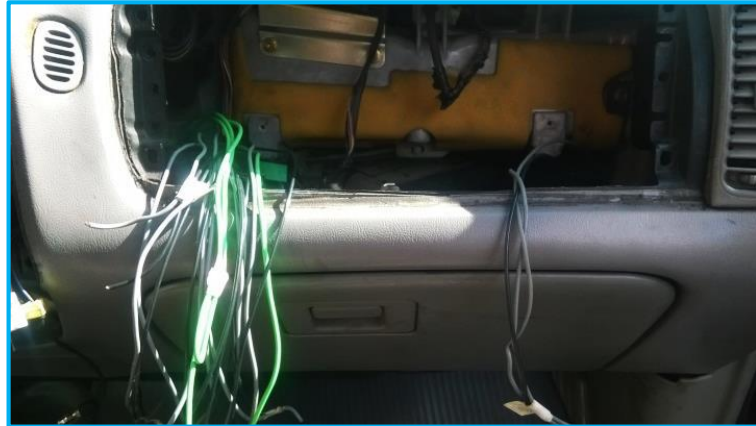


Figura 4.43: Ubicación de las conexiones eléctricas en el vehículo
Fuente: El investigador

Para la conexión del prototipo con los sistemas y accesorios intervenidos en el vehículo se utilizó conectores socket de dos terminales, muy utilizados en el ámbito automotriz para el acoplamiento de luces, aparatos electrónicos, etc. En la Figura 4.44 se observa al prototipo y cableado eléctrico con su respectivo conector.



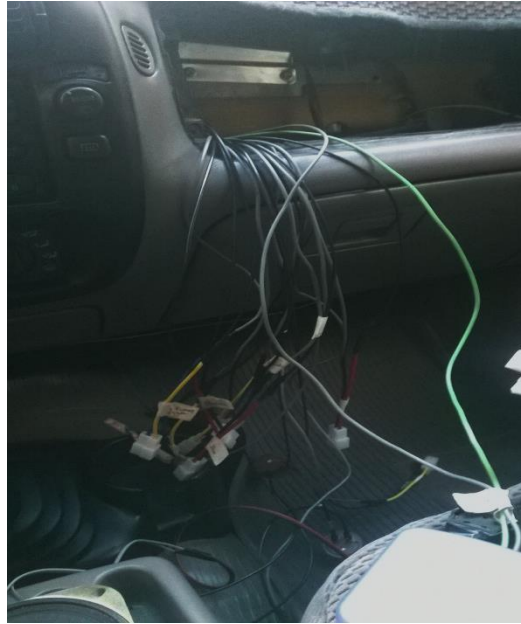


Figura 4.44: Conectores socket de cableado eléctrico y prototipo
Fuente: El investigador

La conexión del prototipo electrónico se realizó primeramente con la revisión de continuidad en terminales y en todas las conexiones eléctricas realizadas en el vehículo, para evitar cortos circuitos y mal funcionamiento. La figura 4.45 muestra el proceso de revisión de terminales y la conexión final del prototipo electrónico.



Figura 4.45: Conexión Prototipo Electrónico
Fuente: El investigador

En la Figura 4.46 se muestra la implementación final del prototipo electrónico por reconocimiento de voz puesto en funcionamiento, cumpliendo todas las expectativas de la investigación y satisfaciendo las necesidades del usuario del vehículo.

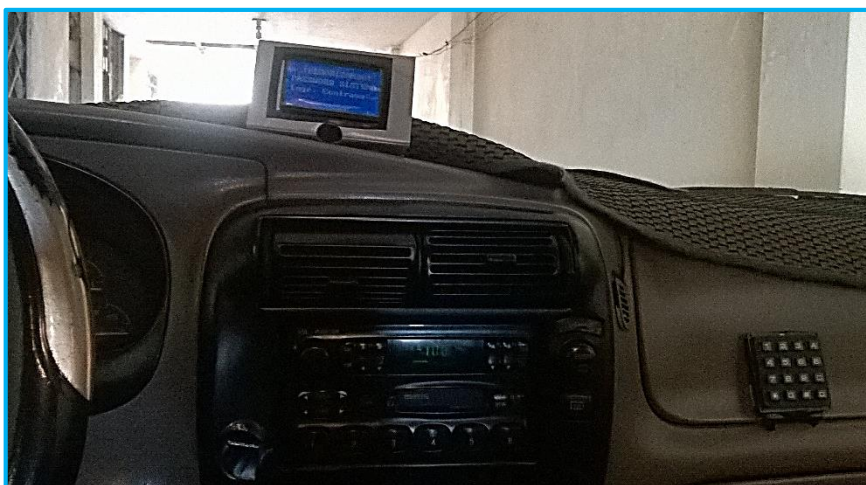


Figura 4.46: Final Prototipo Electrónico por reconocimiento de voz
Fuente: El investigador

4.12. PRUEBAS DE FUNCIONAMIENTO

Las pruebas de funcionamiento fueron necesarias en todo el desarrollo del prototipo electrónico para corregir y mejorar la eficacia del sistema descartando posibles falencias que se pueden presentar tanto en la parte de diseño como de programación, enfocándonos en la operabilidad y comprensión por parte del usuario.

Como primera instancia, se desarrolló los circuitos básicos del prototipo en protoboard sustentándose en sus respectivas simulaciones, evaluando comportamiento y funcionalidad en cada elemento electrónico, evitando sobrecalentamiento, consumo excesivo de corriente, mejorando la eficiencia y comprobando los resultados obtenidos en los cálculos matemáticos. En la Figura 4.47 se ilustra el funcionamiento del circuito básico que constituye el Módulo de Control.

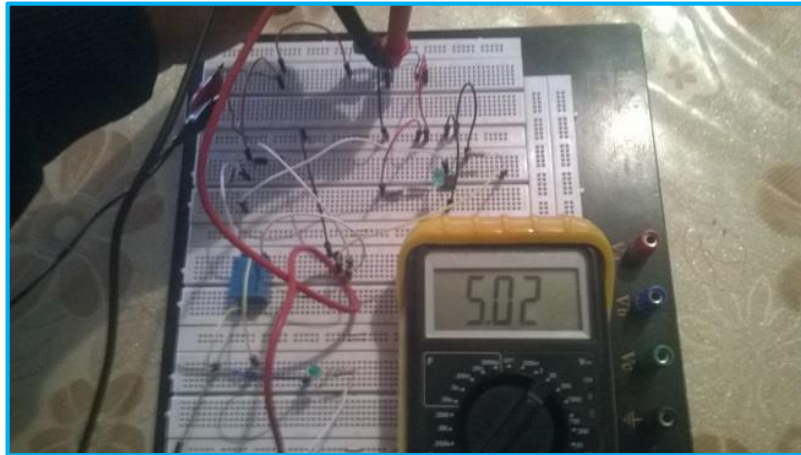


Figura 4.47: Prueba de funcionamiento circuito básico Módulo de Control
Fuente: El investigador

Se realizó pruebas de funcionamiento, antes y después de haber ensamblado el prototipo electrónico, para medir el nivel de eficiencia en la parte de comunicación y control del proyecto. En la Figura 4.48 se muestra el Prototipo Electrónico, realizando las comprobaciones necesarias para su ensamblaje final.

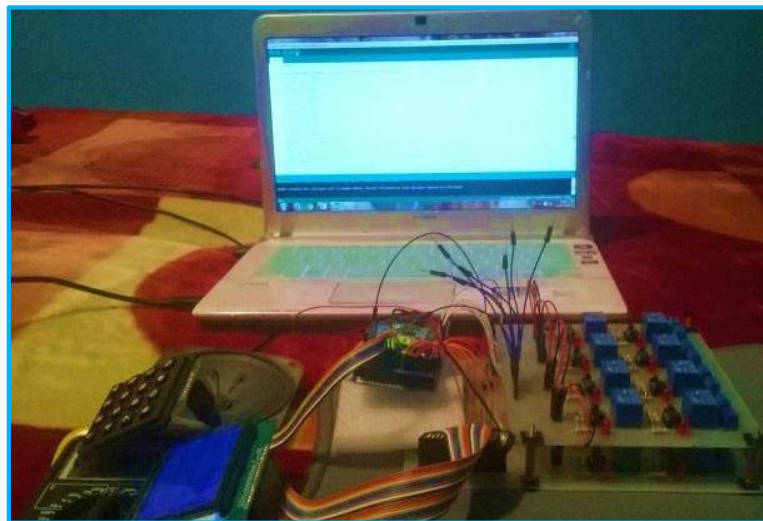


Figura 4.48: Prueba de funcionamiento Prototipo Electrónico
Fuente: El investigador

Las pruebas de comunicación entre el dispositivo de reconocimiento de voz y módulos del sistema fueron realizadas en un tiempo estimado de 2 meses de acuerdo al plan de elaboración del proyecto, durante los meses de junio y julio cuyos resultados se encuentran descritos en la Tabla 4.6.

Comandos	Ambiente Normal (Pre instalación)	Ambiente Ruidoso (Pre instalación)	Ambiente Normal (Vehículo)	Ambiente Ruidoso (Vehículo)	Descripción Problema
Nombre	✓	✓	✓	✓	
Activar_Seguridad	✓	Dificultad	✓	✓	Moderada
Prender	✓	✓	✓	✓	
Apagar	✓	✓	✓	✓	
Altas	✓	✓	✓	✓	
Bajas	✓	✓	✓	✓	
Luces_Fuera	✓	✓	✓	✓	
Neblineros	✓	✓	✓	✓	
Parking	✓	✓	✓	✓	
Salón	✓	✓	✓	✓	
Calefacción	✓	✓	✓	✓	
Plumas	✓	✓	✓	✓	
Limpiar	✓	✓	✓	✓	
Derecha	✓	✓	✓	✓	
Izquierda	✓	✓	✓	✓	
Terminar	✓	✓	✓	✓	
Chofer_abajo	✓	Dificultad	✓	✓	Moderada
Chofer_Arriba	✓	Dificultad	✓	✓	Moderada
Copiloto_Abajo	✓	Dificultad	✓	✓	Moderada
Copiloto_Arriba	✓	Dificultad	✓	✓	Moderada
Abiertas	✓	✓	✓	✓	
Normal	✓	✓	✓	✓	
Hasta_Lueg_Sistema	✓	Dificultad	✓	✓	Moderada
Hola_Sistema	✓	✓	✓	✓	

Tabla 4.6: Pruebas de comunicación Prototipo Electrónico
Fuente: El investigador

El dispositivo de reconocimiento de voz dispone de un micrófono omnidireccional que capta sonidos en cualquier dirección, por lo cual es afectado por interferencias y sonidos procedentes del ambiente. Las pruebas realizadas en el vehículo fueron favorables, ya que el habitáculo es reducido y al momento de reconocer los comandos de voz no tiene mayor inconveniente, cumpliendo las expectativas del proyecto.

En el **Anexo 1** se encuentra el manual de usuario, que es una guía que describe al prototipo electrónico y contiene datos técnica, información del modo de uso y especificaciones de cada comando de voz, adicional advertencias y recomendaciones.

4.13. ANÁLISIS FINAL DE COSTOS

El desarrollo total del Prototipo Electrónico por comando de voz para la seguridad física, iluminación y accionamiento automático de los accesorios primarios de un automóvil se ha realizado por un monto de 799,40 dólares americanos, donde se toma en cuenta todos los gastos generados por parte del investigador.

En la Tabla 4.7 se describe los elementos electrónicos utilizados en el prototipo.

Item	Descripción	Unidad	Cantidad	Valor Unitario(\$)	Valor del Rubro(\$)
1	Relevador 12VDC	c/u	20	0,60	12,00
2	Leds	c/u	20	0,15	3,00
3	Resistencias	c/u	60	0,10	6,00
4	Optoacoplador 4N25	c/u	20	1,50	30,00
5	Transistor 2N3904	c/u	20	0,25	5,00
6	Diodo 1N4007	c/u	20	0,20	4,00
7	Fusible	c/u	2	0,50	1,00
8	Porta fusible	c/u	2	1,00	2,00
9	Conector Socket 2 ter.	c/u	18	3,00	54,00
10	Borneras pequeñas 2 ter.	c/u	20	0,35	7,00
11	Baquelita fibra	c/u	2	5,00	10,00
12	Sockets integrado 8 pines	c/u	20	0,25	5,00
13	Bus de datos 20 hilos	c/u	4	5,00	20,00
14	Jumpers juego	c/u	2	5,00	10,00
15	Batería 9V	c/u	1	5,00	5,00
16	Regulador 12v	c/u	1	0,35	0,35
17	Regulador 5v	c/u	1	0,35	0,35
18	Parlante 8 ohm	c/u	1	7,00	7,00
TOTAL					181,40

Tabla 4.7: Lista de elementos electrónicos
Fuente: El investigador

En la Tabla 4.8 se describe en detalle el monto total de la inversión para el diseño del proyecto:

Items	Descripción	Unidad	Cantidad	Valor unitario(\$)	Valor del rubro(\$)
1	EasyVR Shield 2.0	c/u	1	90,00	90,00
2	Arduino Mega 2560	c/u	1	85,00	85,00
3	Pantalla GLCD 128 x 64	c/u	1	40,00	40,00
4	Teclado matricial alfanumérico	c/u	1	15,00	15,00
5	Elementos electrónicos	c/u	Varios	181,40	181,40
6	Alambre flexible N.18	Metro	50	0,50	25,00
7	Caja plástica acrílico	c/u	1	30,00	30,00
8	Mano de obra[35]	Hora	100	2,33	233,00
9	Gastos otros				100,00
Total					799,40

Tabla 4.8: Análisis final de inversión
Fuente: El investigador

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. CONCLUSIONES

- El prototipo electrónico por reconocimiento de voz cumplió las expectativas por la eficiencia de reconocimiento del habla en el habitáculo del vehículo, ejecutando de una manera correcta cada uno de los procesos en los sistemas que se intervino, no obstante cabe resaltar que el prototipo no afecta el control manual de los mismos, así que existe la opción del control tradicional o utilizando el sistema embebido.
- La visualización del proyecto se desarrolló tomando en cuenta circunstancias extremas para personas especiales con alguna discapacidad física en extremidades superiores, que en cierta manera les impide un adecuado control del vehículo, facilitándoles la interacción con el automotor especialmente en el sistema de iluminación, indispensable al momento de conducir un automotor por ser utilizado para la comunicación con terceros, ya sean transeúntes o demás conductores que utilizan las señales de luces para avisar alguna acción realizada, prevenir e informar algún tipo de anomalía presente en la vía.

- El reconocimiento de voz por sus fiables y únicas cualidades es una gran opción al momento de automatizar o controlar dispositivos, pero tales características también pueden presentarse como desventajas en ciertas situaciones adversas, ya que la voz humana puede cambiar impredeciblemente.
- El micrófono del dispositivo de reconocimiento, tiene una gran sensibilidad al ser del tipo omnidireccional que puede reconocer los comandos de voz en cualquier posición, pero además es afectado por el ruido exterior ya que capta la información de todas las direcciones posibles, afectando en leves ocasiones el entendimiento del sistema de las órdenes emitidas por el usuario.

5.2. RECOMENDACIONES

- Es indispensable que la carga del acumulador del vehículo se encuentre en buenas condiciones, ya que el prototipo electrónico se va a energizar a través de la misma, y al no abastecer la corriente necesaria va a dejar de funcionar correctamente y en la situación de que la batería se encuentre descargada desalimentará al sistema y se borrará la clave de control ingresada al momento de activar por primera vez al sistema embebido.
- Tomar en consideración los factores ambientales, el estado de salud, el estado de ánimos y la ingesta de sustancias estupefacientes o bebidas alcohólicas al momento de conducir el vehículo, ya que dichos elementos alteran la tonalidad de la voz humana, complicando el reconocimiento haciendo que el sistema no reconozca las órdenes emitidas por el usuario.
- La posición del micrófono del prototipo electrónico es un factor fundamental para mejorar la eficiencia al momento de reconocer los comandos emitidos por el usuario, donde la ubicación adecuada es en sentido frontal al locutor con una distancia mínima de 60 cm y máxima de 100 cm a la altura del rostro.

BIBLIOGRAFÍA

- [1] O. Gómez, “Sistemas de Seguridad en vehículos Sisve”, M.S. Tesis, Ingeniería Electrónica, Universidad Politécnica de Madrid, Madrid, 2013.
- [2] A. D. Ruiz, A. D. Rueda, “Control Electrónico automatizado del sistema de alumbrado de un vehículo para diferentes condiciones de circulación”, Tesis, Ingeniería Automotriz, Escuela Superior Politécnica del Ejército, Latacunga, 2013.
- [3] F. O. Cando, A. M. Tipán, “Diseño e Implementación de un sistema inteligente de luces, ventanillas eléctricas y limpiaparabrisas controlado mediante comandos de voz”, Tesis, Ingeniería Automotriz, Escuela Superior Politécnica del Ejército”, Latacunga, 2010.
- [4] A. Delgado, (2012). Eroski Consumer. "Reconocimiento de voz para coches" [En línea]. Disponible en:
<http://www.consumer.es/web/es/tecnologia/software/2012/06/12/210171.php>
- [5] J. E. Llivicula, L. M. Sagnay, “Rediseño de los Elementos y Sistemas de Seguridad del transporte urbano –Bus tipo- de la ciudad de Cuenca”, Tesis, Ingeniería Mecánica Automotriz, Universidad Politécnica Salesiana, Cuenca, 2007.
- [6] Rodes, (2014). Mecánica, El sistema de alumbrado del vehículo [Online]. Aavailable:
<http://www.ro-des.com/mecanica/sistema-alumbrado-del-coche-que-es/>
- [7] Consorcio, (2013). Tu auto seguro, “Última tecnología para evitar el robo de autos” [Online]. Aavailable: <http://www.tuautoseguro.cl/conduccion/cual-es-la-ultima-tecnologia-para-evitar-el-robo-de-tu-auto/>

- [8] Aficionados a la mecánica, (2014). Cursos, “Curso rápido de electricidad de automóvil” [Online]. Available: <http://www.aficionadosalamecanica.net/elevelunas-electrico.htm>
- [9] Aficionados a la mecánica, (2014). Cursos, “Equipo limpia parabrisas” [Online]. Available: <http://aficionadosalamecanica.com/limpiaparabrisas-motor.htm>
- [10] Aficionados a la mecánica (2014). Manual, “Sistema de calefacción en el automóvil” [Online]. Available: <http://www.aficionadosalamecanica.net/manuales/Sistema%20de%20calefaccion%20en%20el%20automovil.pdf>
- [11] Biometría, (2015, April). Métodos Biométricos, “Reconocimiento de Voz” [Online]. Available: <http://www.biometria.gov.ar/metodos-biometricos/voz.aspx>
- [12] St. Franklin, (2014, March). Directory M, “Reconocimiento de voz” [Online]. Available: http://www.articulosinformativos.com/Reconocimiento_de_Voz-a963743.html
- [13] UNAM, (2013). Biometría, “Reconocimiento de voz” [Online]. Available: <http://redyseguridad.fi-p.unam.mx/proyectos/biometria/clasificacionsistemas/clasificacionvoz.html>
- [14] F. Martínez, G. Portale, H. Klein, O. Olmos, (2008). Introducción a la inteligencia Artificial, “Reconocimiento de voz” [Online]. Available: http://www.secyt.frba.utn.edu.ar/gia/IA1_IntroReconocimientoVoz.pdf
- [15] Sánchez S., (2013). Microcontroladores y sus aplicaciones, Introducción y Arquitectura de Microcontroladores [Online]. Available: <https://microcontroladoresesv.wordpress.com/arquitectura-de-los-microcontroladores/>

- [16] Arduino C., (2013). Arduino, “Introducción Arduino” [Online]. Available: <http://arduino.cc/>
- [17] AppleInside, (2012, Febrero). Communication System, “Mercedes Benz to add SIRI Integration” [Online]. Available: http://appleinsider.com/articles/12/02/28/mercedes_benz_to_add_siri_integration_to_new_a_class_line.html
- [18] Engadget, (2011, Noviembre). Pioneer Zipr, “Reconocimiento de voz” [Online]. Available: <http://es.engadget.com/2011/11/11/pioneer-zypr-reconocimiento-de-voz-a-lo-siri-gratuito-y-para/>
- [19] Boston Bussiness, (2013, Mayo). Industry News, “Nuance announces dragon Drive” [Online]. Available: <http://www.bizjournals.com/boston/news/2012/05/22/nuance-dragon-drive.html>
- [20] Cnet., (2014, Junio). Tecnología automotriz, “Android Auto” [Online]. Available: <http://www.cnet.com/es/analisis/google-android-auto/>
- [21] Arduino C., (2013). Arduino, “Arduino Mega 2560” [Online]. Available: <https://www.arduino.cc/en/Main/arduinoBoardMega2560>
- [22] L. Thayer, (2014). Placas Arduino, “EasyVR Shield 2.0” [Online]. Available: <http://arduino.cl/easyvr-shield-2-0/>
- [23] ArduinoStar, (2015). Comunidad Arduino, “LCD 12864WZ con ST7920” [Online]. Available: http://www.digole.com/images/file/Digole_12864_LCD.pdf
- [24] E. Mandado, “Microcontroladores PIC: Sistema integrado para el autoaprendizaje”, Teclado Matricial, Ed. Marcombo, España, 2006, pg. 148 – 149.

- [25] National Instrument. (2006). Datasheet, “RH - 012C”. [Online]. Available: <http://www.hobbytronics.co.uk/datasheets/songle-12v-relay.pdf>
- [26] Diodes Incorporated. (2014). Datasheet, “1N4001-1N4007”. [Online]. Available: <http://www.diodes.com/datasheets/ds28002.pdf>
- [27] Vishay Semiconductors, (2009). Datasheet, “4N25, 4N26, 4N27, 4N28” [Online]. Available: <http://www.vishay.com/docs/83725/4n25.pdf>
- [28] ST, (2006). Datasheet, “2N3904” [Online]. Available: <http://www.st.com/web/en/resource/technical/document/datasheet/CD00002987.pdf>
- [29] Arduino C., (2013). Arduino, “Arduino IDE” [Online]. Available: <https://www.arduino.cc/en/main/software>
- [30] L. Morozov, (2009). Tecno Accesible, “Balabolka” [Online]. Available: <http://www.tecnoaccesible.net/content/balabolka>
- [31] J. Hamblen, (Febrero 2012). ARMMbed, “EasyVR Commander” [Online]. Available: https://developer.mbed.org/users/4180_1/notebook/easyvr/
- [32] Sensory, Inc., (2002). User’s Manual, “Quick Synthesis” [Online]. Available: <file:///C:/Users/USUARIO/Downloads/smart80-0186-c20140122151127.pdf>
- [33] F. Afridi, (2013). Get Into PC, “Proteus 8 Overview” [Online]. Available: <http://getintopc.com/software/electronics/proteus-8-free-download/>
- [34] CONADIS, “Agenda Nacional para la igualdad en Discapacidades”, en Ley Orgánica de Tránsito Terrestre, Tránsito y Seguridad Vial, Ed. Guayaquil, Ecuador, 2014, pp. 32-36.

- [35] CONADES, (2015). Consejo Nacional de Salarios y Comisiones Sectoriales, “Salarios” [Online]. Available: <http://www.relacioneslaborales.gob.ec/el-salario-basico-para-el-2015-sera-de-354-dolares/>

ANEXOS

ANEXO I

MANUAL DE USUARIO

SISTEMA ELECTRÓNICO DE RECONOCIMIENTO DE VOZ

MANUAL DE USUARIO



INFORMACIÓN USUARIO:

El Sistema de reconocimiento de voz es un sistema embebido utilizado para controlar el sistema de iluminación, calefacción y limpiaparabrisas, además permite el accionamiento de las ventanillas eléctricas del vehículo sin la intervención manual; todo por comandos de voz.

Dispone de un sistema de seguridad que protege al vehículo de intentos de encendido ilícitos que puede ser activado o desactivado por el usuario a través del habla o por una contraseña ingresada al momento de activar por primera vez el sistema.

La interfaz de usuario se encuentra constituida por una pantalla donde se visualiza la ejecución de los procesos que controla el sistema; un teclado alfanumérico para el ingreso de una contraseña como medio para la desactivación de la seguridad y un altavoz que reproduce grabaciones del sistema de alertas y ejecución de procesos.

Tiene la Opción de encender y apagar su vehículo por comandos de voz.

INFORMACIÓN TÉCNICA:

- ✓ Voltaje de Alimentación: 12V
- ✓ Salidas de control normalmente abiertas: 14
- ✓ Salidas de control normalmente cerradas: 2
- ✓ Salidas de control dedicadas para ventanillas: 2
- ✓ Porta fusibles de 7A y 15A
- ✓ Teclado alfanumérico
- ✓ Parlante de 8 ohm
- ✓ Fuente auxiliar de alimentación: batería de 9V

REQUERIMIENTOS:

- ✓ Acumulador del vehículo de 12V
- ✓ Ventanillas eléctricas
- ✓ Sistema de seguridad (Alarma)

Nota: Del estado de carga de la batería depende el adecuado funcionamiento del sistema. Al descargarse la batería conectar la fuente auxiliar de alimentación para evitar mal funcionamiento.

COMANDOS:

- ✓ Los comandos se encuentran distribuidos en tres grupos:
 - **Trigger:** Comando contraseña para desactivar sistema de seguridad.
 - **Grupo 1:** Comandos de control del sistema.
 - **Grupo 2:** Comando modo Standby.

En la tabla se describe los diferentes comandos del sistema de reconocimiento de voz.

Grupo	Item	Nombre
Trigger	0	NOMBRE
Grupo 1	0	ACTIVAR_SEGURIDAD
	1	PRENDER
	2	APAGAR
	3	ALTAS
	4	BAJAS
	5	LUCES_FUERA
	6	NEBLINEROS
	7	PARKING
	8	SALÓN
	9	CALEFACCIÓN
	10	PLUMAS
	11	LIMPIAR
	12	DERECHA
	13	IZQUIERDA
	14	TERMINAR
	15	CHOFER_ABAJO
	16	CHOFER_ARRIBA
	17	COPILOTO_ABAJO
	18	COPILOTO_ARRIBA
	19	ABIERTAS
	20	NORMAL
	21	HASTA_LUEGO_SISTEMA
Grupo 2	0	HOLA_SISTEMA

FUNCIÓN DE CADA COMANDO:

- NOMBRE: Contraseña para desactivar el sistema de Seguridad.
- ACTIVAR_SEGURIDAD: Activa Sistema de Seguridad.
- PRENDER: Prende el vehículo.
- APAGAR: Apaga el vehículo.
- ALTAS: Enciende las luces altas.
- BAJAS: Enciende las luces bajas.

- LUCES_FUERA: Apaga luces altas y bajas.
- NEBLINEROS: Enciende luces auxiliares para neblina.
- PARKING: Enciende luces de emergencia.
- SALÓN: Enciende luces del habitáculo del vehículo.
- CALEFACCIÓN: Activa la calefacción.
- PLUMAS: Activa limpiaparabrisas.
- LIMPIAR: Activa limpiaparabrisas y bomba de agua.
- DERECHA: Enciende luz de dirección de vía derecha.
- IZQUIERDA: Enciende luz de dirección de vía izquierda.
- TERMINAR: Apaga luces de dirección de vía.
- CHOFER_ABRIR: Baja el cristal de la ventana del chofer.
- CHOFER_CERRAR: Sube el cristal de la ventana del chofer.
- COPILOTO_ABRIR: Baja el cristal de la ventana del copiloto.
- COPILOTO_CERRAR: Sube el cristal de la ventana del copiloto.
- ABIERTAS: Baja los cristales de las ventanas del chofer y copiloto.
- NORMAL: Sube los cristales de las ventanas del chofer y copiloto.
- HASTA_LUEGO_SISTEMA: Modo StandBy prototipo electrónico.
- HOLA_SISTEMA: Despierta prototipo electrónico del Modo StandBy.

Nota: La pronunciación por primera ocasión de un comando activa la función determinada por el sistema, al repetir por segunda ocasión el mismo comando después de haber ejecutado su proceso, desactiva al mismo.

INSTRUCCIONES:

Posterior a la realización de las conexiones necesarias en el sistema eléctrico, y comprobación de continuidad, se adapta el prototipo y se energiza. La primera ocasión de ejecución del sistema electrónico de reconocimiento de voz, se activa la pantalla en donde se visualiza mensajes con indicaciones a realizar:

Memorizar contraseña a través del teclado:

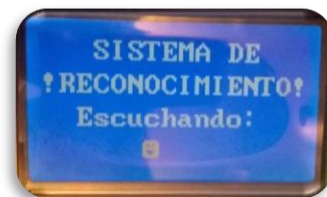


A continuación mensaje de confirmación de contraseña:



Nota: El registro de la contraseña se ejecuta al funcionar por primera vez el sistema, y a continuación, se omite este proceso en el funcionamiento. Al ser des energizado el sistema se genera un reset de la misma.

Después de memorizar la contraseña, se activa el modo escucha del sistema de reconocimiento:



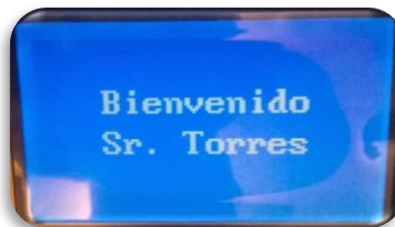
MODO ESCUCHA ACTIVADO

Para ingresar al sistema y controlar los diferentes dispositivos inmiscuidos, como primer paso se debe pronunciación obligatoriamente el comando de seguridad, para comprobar la identidad del usuario.

Al no coincidir el comando de voz, en la pantalla se visualiza:



Al ser aceptado el comando, en la pantalla se visualiza un mensaje de bienvenida al usuario:



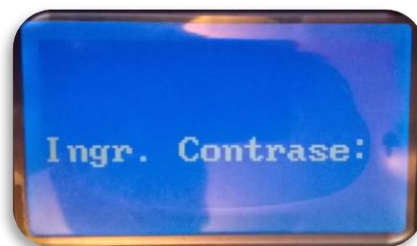
Tras ser aceptado el comando de voz se desactiva el sistema de seguridad, permitiendo continuar con el control del sistema a través de la pronunciación de los comandos del Grupo 1 y el encendido del vehículo.

DESACTIVAR SISTEMA DE SEGURIDAD

Una vez de haber ingresado la contraseña como primer parámetro se tiene la opción de desactivar el sistema de seguridad a través del teclado o por comando de voz. Al activar el sistema por el comando de voz `ACTIVAR_SEGURIDAD`, se restringe al sistema de desactivar la seguridad solo con el ingreso de contraseña:

Se oprime el botón [*]:

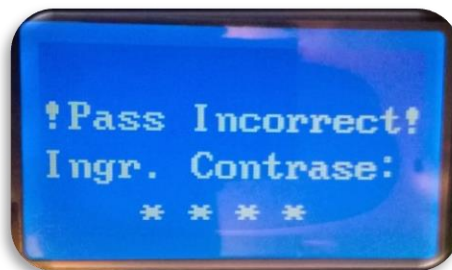
- Ejecuta la función de ingreso de contraseña por teclado, mostrando en la pantalla:



- Proceso de ingreso de contraseña:



- Cuando la contraseña es errónea, en la pantalla se visualiza:



- Cuando la contraseña es correcta, en la pantalla se visualiza:



Se desactiva el sistema de seguridad, permitiendo continuar con el control del sistema a través de la pronunciación de los comandos del Grupo 1 y el encendido del vehículo.

Nota: El botón [#] permite borrar los dígitos ingresados en el caso de error accidental.

MODO STANDBY

Standby es el segundo método de activación de la seguridad, y a su vez pone al sistema en modo de ahorro de energía, utilizado cuando el usuario permanece en el interior del vehículo. Para activar este modo se pronuncia el comando HASTA_LUEGO_SISTEMA.

El modo Standby se puede activar una vez se haya desactivado el sistema de seguridad por teclado, cumpliendo las funciones del mismo. El sistema de seguridad se desactiva por comando de voz, pronunciando HOLA_SISTEMA.

ADVERTENCIA:

- ✓ Mantener descubierto el micrófono para que el sistema pueda recibir los comandos de voz de una manera directa sin disminuir su capacidad de recepción.
- ✓ Evitar golpear, mojar o desmontar el sistema para evitar fallas o destrucción de cada módulo que conforma el sistema electrónico de reconocimiento de voz.
- ✓ La ingesta de sustancias estupefacientes o bebidas alcohólicas altera la tonalidad de habla afectando al reconocimiento del sistema electrónico.
- ✓ Queda estrictamente obligatorio que la instalación del sistema sea realizado por personal capacitado.

ANEXO II
CÓDIGO DE PROGRAMACIÓN

PROGRAMACIÓN DEL PROYECTO

```
/*=====
  SISTEMA DE RECONOCIMIENTO DE VOZ
=====*/

// Librerias
#include "Arduino.h"
#include "EasyVR.h"
#include <LiquidCrystal.h>
#include <SPI.h>
#include <Keypad.h>

#if !defined(SERIAL_PORT_MONITOR)
  #error "Arduino version not supported. Please update your IDE to the latest version."
#endif

#if defined(SERIAL_PORT_USBVIRTUAL)
  // Shield Jumper on HW (for Leonardo and Due)
  #define port SERIAL_PORT_HARDWARE
  #define pcSerial SERIAL_PORT_USBVIRTUAL
#else
  // Shield Jumper on SW (using pins 12/13 or 8/9 as RX/TX)

  #include "SoftwareSerial.h"
  SoftwareSerial port(12, 13);
  #define pcSerial SERIAL_PORT_MONITOR
#endif

EasyVR easyvr(port);

//Grupo de Comandos EasyVR
enum Groups
{
  GROUP_0 = 0,
  GROUP_1 = 1,
  GROUP_2 = 2,
};

enum Group0
{
  G0_NOMBRE = 0,
};

enum Group1
{
  G1_ACTIVAR_SEGURIDAD = 0,
  G1_ALTAS = 1,
  G1_BAJAS = 2,
  G1_LUCES_APAGAR = 3,
  G1_PARQUIN = 4,
  G1_SALON = 5,
  G1_DERECHA = 6,
```

```

G1_IZQUIERDA = 7,
G1_CALEFACCION = 8,
G1_PLUMAS = 9,
G1_CHOFER_CERRAR = 10,
G1_CHOFER_ABRIR = 11,
G1_COPILOTO_ABRIR = 12,
G1_COPILOTO_CERRAR = 13,
G1_NORMAL = 14,
G1_ABIERTAS = 15,
G1_PRENDER = 16,
G1_APAGAR = 17,
G1_NEBLINEROS = 18,
G1_HASTA_LUEGO_SISTEMA = 19,
G1_LIMPIAR = 20,
G1_TERMINAR = 21,
G1_SEGURIDAD = 22,
};

enum Group2
{
    G2_HOLA_SISTEMA = 0,
};

int8_t group, idx;

char password[4]; //vector principal de caracteres de 4x4 para almacenar la contraseña
int almacena = 0; //variable para contar número de caracteres de la contraseña
int i = 0;
char clave[4]; //vector auxiliar de caracteres 4x4

const byte filas = 4; //número de filas
const byte columnas = 4; //número de columnas

char keys[filas][columnas] = { //matriz 4x4 para teclado
    {'1','2','3','A'},
    {'4','5','6','B'},
    {'7','8','9','C'},
    {'*','0','#','D'}
};

byte filasPins[filas] = {45, 43, 41, 39 }; //pines de filas teclado
byte columnasPins[columnas] = {53, 51, 49, 47}; //pines de columnas teclado

Keypad keypad = Keypad( makeKeymap(keys), filasPins, columnasPins, filas, columnas ); //estructura de la matriz
teclado

LiquidCrystal lcd(24, 26, 28, 30, 32, 34); //pines lcd

// variables
int alarma = 11;
int luces = 10;
int contacto = 9;
int apagar = 8;
int neblineros = 7;
int parquin = 6;
int salon = 5;
int aire = 4;
int plumas = 3;

```

```

int agua = 2;
int derecha = 15;
int izquierda = 16;
int chofer_abrir = 17;
int chofer_cerrar = 18;
int copiloto_abrir = 19;
int copiloto_cerrar = 36;

void setup(){

    //pines modo salidas
    pinMode(2,OUTPUT);
    pinMode(3,OUTPUT);
    pinMode(4,OUTPUT);
    pinMode(5,OUTPUT);
    pinMode(6,OUTPUT);
    pinMode(7,OUTPUT);
    pinMode(8,OUTPUT);
    pinMode(9,OUTPUT);
    pinMode(10,OUTPUT);
    pinMode(11,OUTPUT);
    pinMode(15,OUTPUT);
    pinMode(16,OUTPUT);
    pinMode(17,OUTPUT);
    pinMode(18,OUTPUT);
    pinMode(19,OUTPUT);
    pinMode(36,OUTPUT);
    pinMode(22,OUTPUT);

    lcd.begin(20, 4);    //inicializando lcd

    //MENSAJE DE INICIALIZACION DEL SISTEMA
    digitalWrite(22,HIGH);
    lcd.home ();
    lcd.print("=====");
    lcd.setCursor(16,0);
    lcd.print(" !Hola, Usuario! ");
    lcd.setCursor(24,0);
    lcd.print("=====");
    delay (3000);

    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" !MEMORIZANDO! ");
    lcd.setCursor(16,0);
    lcd.print("PASSWORD SISTEMA");
    lcd.setCursor ( 8,0 );
    lcd.print ("Ingr. Contrase:");
    delay (1000);

    while (almacena < 4){

        lcd.setCursor(almacena+26,0);
        lcd.cursor();

        char key = keypad.getKey(); //obteniendo caracteres de la matriz teclado
        key == NO_KEY; //comparando

        if (key != NO_KEY){ //comparando

```

```

    if ((key != '*')&&(key != '#')&&(key != 'A')&&(key != 'B')&&(key != 'C')&&(key != 'D')){ //descarta tipos de
caracteres
    Serial.print(key);
    lcd.print(key); //visualización de contraseña en lcd
    password[almacena] = key; //almacena caracteres ingresados en matriz key a matriz password
    almacena++; //incremento de la variable de conteo
    }
}
}

```

```

if (almacena == 4){ //termina el ingreso de caracteres a la matriz password
    delay(500);

```

```

    //visulizacion de contraseña en lcd
    lcd.noCursor();
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("CLAVE INTRODUC:");
    lcd.setCursor(19,0);
    lcd.print(password[0]);
    lcd.print(password[1]);
    lcd.print(password[2]);
    lcd.print(password[3]);
    delay(3000);
    lcd.clear();
    almacena = 0;
}

```

```

Serial.begin(9600); //inicializando comunicacion serial

```

```

// bridge mode?
int mode = easyvr.bridgeRequested(pcSerial);
switch (mode)
{
case EasyVR::BRIDGE_NONE:
    // Programación Puerto serial EasyVR
    port.begin(9600);
    break;

case EasyVR::BRIDGE_NORMAL:
    // setup EasyVR serial port (low speed)
    port.begin(9600);
    // Software conectado a dos puestos (PC e EasyVR)
    easyvr.bridgeLoop(pcSerial);
    break;

case EasyVR::BRIDGE_BOOT:
    // setup EasyVR serial port (alta velocidad)
    port.begin(115200);
    //Software conectado a dos puestos (PC e EasyVR)
    easyvr.bridgeLoop(pcSerial);
    // si no se ha establecido conexión
    break;
}

while (!easyvr.detect())
{
    Serial.println("EasyVR no detectado");
    delay(1000);
}

```

```

easyvr.setPinOutput(EasyVR::IO1, LOW);
Serial.println("EasyVR detectado");
easyvr.setTimeout(0); //tiempo para escuchar comando ilimitado
easyvr.setLanguage(4); //idioma español

group = EasyVR::TRIGGER; //Grupo de Inicialización
}

//Funciones del Sistema
void action();
void teclado();
void dormir();
char despertar();

void loop(){

easyvr.setPinOutput(EasyVR:IO1, HIGH); // LED ON (Escuchando)
Serial.print("Say a command in Group ");
Serial.println(group);
easyvr.recognizeCommand(group);

do{
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print(" SISTEMA DE ");
  lcd.setCursor(16,0);
  lcd.print("!RECONOCIMIENTO!");
  lcd.setCursor(8,0);
  lcd.print(" Escuchando: ");
  lcd.setCursor(27,0);
  lcd.print(char(2));
  lcd.blink();
  delay(200);
}

while (!easyvr.hasFinished());

easyvr.setPinOutput(EasyVR:IO1, LOW); // LED off (Ejecutando)

idx = easyvr.getWord();
if (idx <= 0)
{
  return;
}

idx = easyvr.getCommand();
if (idx >= 0)
{
  uint8_t train = 0;
  char name[32];
  Serial.print("Command: ");
  Serial.print(idx);
  if (easyvr.dumpCommand(group, idx, name, train))
  {
    Serial.print(" = ");
    Serial.println(name);
  }
  else
    Serial.println();
  easyvr.playSound(0, EasyVR:VOL_FULL);

  lcd.noBlink();
}

```



```

lcd.clear();
lcd.setCursor(0,0);
lcd.print(" SISTEMA DE ");
lcd.setCursor(16,0);
lcd.print("!RECONOCIMIENTO!");
lcd.setCursor(8,0);
lcd.print(" !!Ejecutando!! ");
lcd.setCursor(27,0);
lcd.print(char(3));
delay(2000);
action();
}

else // errors o timeout
{
  if (easyvr.isTimeout())
    Serial.println("Timed out, try again...");
  int16_t err = easyvr.getError();
  if (err >= 0)
  {
    Serial.print("Error ");
    Serial.println(err, HEX);

    lcd.noBlink();
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(" !!ERROR!! ");
    lcd.setCursor(16,0);
    lcd.print(" Accion no ");
    lcd.setCursor(8,0);
    lcd.print(" identificada ");
    lcd.setCursor(24,0);
    lcd.print("VUENV A INTENTAR");
    delay(2000);
  }
}
}

void action()
{
  switch (group)
  {
  case GROUP_0:
    switch (idx)
    {
    case G0_NOMBRE: //desactivar sistema de seguridad
      digitalWrite(11, HIGH);
      digitalWrite(5, HIGH);
      delay(100);
      digitalWrite(11, LOW);
      digitalWrite(5, LOW);
      easyvr.playSound(10, EasyVR::VOL_FULL);
      group = GROUP_1; //salto a grupo 1
      break;
    }
    break;

  case GROUP_1:
    switch (idx)
    {
    case G1_ACTIVAR_SEGURIDAD: //activar sistema de seguridad

```

```

digitalWrite(11, HIGH);
digitalWrite(5, HIGH);
delay(100);
digitalWrite(11, LOW);
digitalWrite(10, LOW);
digitalWrite(9, LOW);
digitalWrite(8, LOW);
digitalWrite(7, LOW);
digitalWrite(6, LOW);
digitalWrite(5, LOW);
digitalWrite(4, LOW);
digitalWrite(3, LOW);
digitalWrite(2, LOW);
digitalWrite(15, LOW);
digitalWrite(16, LOW);
digitalWrite(17, LOW);
digitalWrite(18, LOW);
digitalWrite(19, LOW);
digitalWrite(36, LOW);
easyvr.playSound(9, EasyVR::VOL_FULL);
easyvr.playSound(6, EasyVR::VOL_FULL);
group = GROUP_0; //salto a grupo 0
dormir();
break;

case G1_PRENDER: //prender auto
digitalWrite(9, HIGH);
delay(2000);
digitalWrite(9, LOW);
easyvr.playSound(14, EasyVR::VOL_FULL);
break;

case G1_APAGAR: //apagar auto
digitalWrite(8, HIGH);
delay(1000);
digitalWrite(8, LOW);
easyvr.playSound(13, EasyVR::VOL_FULL);
break;

case G1_ALTAS: // encender luces altas
digitalWrite(10, HIGH);
easyvr.playSound(5, EasyVR::VOL_FULL);
break;

case G1_BAJAS: // encender luces bajas
digitalWrite(10, LOW);
easyvr.playSound(5, EasyVR::VOL_FULL);
break;

case G1_LUCES_APAGAR: // apagar luces altas y luces bajas
digitalWrite(10, LOW);
easyvr.playSound(5, EasyVR::VOL_FULL);
break;

case G1_NEBLINEROS: //activar neblineros
easyvr.playSound(5, EasyVR::VOL_FULL);
if (digitalRead(7)==LOW){
digitalWrite(7,HIGH);
}
else{
digitalWrite(7,LOW);
}
}

```

```

break;

case G1_PARQUIN: //activar luces de parqueo
easyvr.playSound(5, EasyVR::VOL_FULL);
if ((digitalRead(15)==LOW)&&(digitalRead(16)==LOW)){
    digitalWrite(15, HIGH);
    digitalWrite(16, HIGH);
}
else{
    digitalWrite(15, LOW);
    digitalWrite(16, LOW);
}
break;

case G1_SALON: //activar luces del salon
easyvr.playSound(5, EasyVR::VOL_FULL);
if (digitalRead(5)==LOW){
    digitalWrite(5,HIGH);
}
else{
    digitalWrite(5,LOW);
}
break;

case G1_CALEFACCION: //activar calefaccion
easyvr.playSound(5, EasyVR::VOL_FULL);
if (digitalRead(4)==LOW){
    digitalWrite(4,HIGH);
}
else{
    digitalWrite(4,LOW);
}
break;

case G1_PLUMAS: //activar plumas
easyvr.playSound(5, EasyVR::VOL_FULL);
if (digitalRead(3)==LOW){
    digitalWrite(3,HIGH);
}
else{
    digitalWrite(3,LOW);
}
break;

case G1_LIMPIAR: //activar plumas y agua
easyvr.playSound(5, EasyVR::VOL_FULL);
digitalWrite(2, HIGH);
delay(3000);
digitalWrite(2, LOW);
break;

case G1_DERECHA: //activar direccional derecha por 10 sg
easyvr.playSound(2, EasyVR::VOL_FULL);
digitalWrite(15, HIGH);
digitalWrite(16,LOW);
break;

case G1_IZQUIERDA: //activar direccional izquierda por 10 sg
easyvr.playSound(3, EasyVR::VOL_FULL);
digitalWrite(16, HIGH);
digitalWrite(15,LOW);

```

```
break;

case G1_TERMINAR: //termina proceso de direccionales
easyvr.playSound(5, EasyVR::VOL_FULL);
digitalWrite(15, LOW);
digitalWrite(16, LOW);
break;

case G1_CHOFER_ABRIR: //bajar vidrio ventanilla chofer
easyvr.playSound(5, EasyVR::VOL_FULL);
digitalWrite(17, HIGH);
delay(5000);
digitalWrite(17, LOW);
break;

case G1_CHOFER_CERRAR: //subir vidrio ventanilla chofer
easyvr.playSound(5, EasyVR::VOL_FULL);
digitalWrite(18, HIGH);
delay(5000);
digitalWrite(18, LOW);
break;

case G1_COPILOTO_ABRIR: //bajar vidrio ventanilla copiloto
easyvr.playSound(5, EasyVR::VOL_FULL);
digitalWrite(19, HIGH);
delay(5000);
digitalWrite(19, LOW);
break;

case G1_COPILOTO_CERRAR: //subir vidrio ventanilla copiloto
easyvr.playSound(5, EasyVR::VOL_FULL);
digitalWrite(20, HIGH);
delay(5000);
digitalWrite(20, LOW);
break;

case G1_ABIERTAS: //abrir ambas ventanillas
easyvr.playSound(5, EasyVR::VOL_FULL);
digitalWrite(17, HIGH);
digitalWrite(19, HIGH);
delay(5000);
digitalWrite(17, LOW);
digitalWrite(19, LOW);
break;

case G1_NORMAL: //subir ambas ventanillas
easyvr.playSound(5, EasyVR::VOL_FULL);
digitalWrite(18, HIGH);
digitalWrite(36, HIGH);
delay(5000);
digitalWrite(18, LOW);
digitalWrite(36, LOW);
break;

case G1_SEGURIDAD:
break;

case G1_HASTA_LUEGO_SISTEMA: //salta a grupo 2 y pone al sistema en stand bye
digitalWrite(11, HIGH);
digitalWrite(5, HIGH);
delay(100);
digitalWrite(11,LOW);
```

```

digitalWrite(10, LOW);
digitalWrite(9, LOW);
digitalWrite(8, LOW);
digitalWrite(7, LOW);
digitalWrite(6, LOW);
digitalWrite(5, LOW);
digitalWrite(4, LOW);
digitalWrite(3, LOW);
digitalWrite(2, LOW);
digitalWrite(15, LOW);
digitalWrite(16, LOW);
digitalWrite(17, LOW);
digitalWrite(18, LOW);
digitalWrite(19, LOW);
digitalWrite(36, LOW);
easyvr.playSound(9, EasyVR::VOL_FULL);
easyvr.playSound(6, EasyVR::VOL_FULL);
port.write('s'); //Standby EASYVR
port.write('A'); //Se despierta con cualquier sonido
digitalWrite(22, LOW);
group = GROUP_2;
break;
}
break;

case GROUP_2:
switch (idx)
{
case G2_HOLA_SISTEMA: //despierta sistema, salta grupo 0
port.write('w');
digitalWrite(11, HIGH);
digitalWrite(5, HIGH);
delay(300);
digitalWrite(11, LOW);
digitalWrite(5, LOW);
digitalWrite(22, HIGH);
easyvr.playSound(4, EasyVR::VOL_FULL);
group = GROUP_0;
break;
}
break;
}
}

void teclado(){

lcd.clear();
lcd.setCursor (8,0);
lcd.print ("Ingr. Contrase:");

while (almacena < 4){ //condicion para el borrado de contraseña erronea

char key2 = keypad.getKey();

if (key2 == '#'){ //caracter que borrar contraseña erronea ingresada
almacena = 0; //variable de conteo reseteada

//Serial.println("!!BORRADO!!");
//Serial.println("CODIGO: ");

lcd.clear();

```

```

    lcd.setCursor(16,0);
    lcd.print(" !!BORRADO!! ");
    delay(2000);
    lcd.clear();
    lcd.setCursor(8,0);
    lcd.print("Ingr. Contrase:");
}

else
if (key2 != NO_KEY){ //condicion para imprimir datos en lcd

    lcd.setCursor(almacena+26,0);
    lcd.cursor();
    lcd.print(key2);

    //Serial.print(key2);

    clave[almacena] = key2; //caracteres ingresados se almacenan en vector clave
    almacen++;

    delay(100);
    Serial.print("*");

    lcd.noCursor();
    lcd.setCursor(almacena + 25, 0);
    lcd.print("*");
    lcd.setCursor(almacena + 26, 0);
    lcd.cursor();
}
}

if (almacena == 4){ //condicion de finalizacion de ingreso de clave

    //condicion de comparacion de clave ingresada con vector password
    if (clave[0] == password[0] && clave[1] == password[1] && clave[2] == password[2] && clave[3] ==
password[3]){

        //si es correcto la clave ingresada se visualiza en lcd:

        lcd.noCursor();
        lcd.setCursor(16,0);
        lcd.print("!Clave Correcta!");
        almacen = 0;
        delay(2000);
        //Serial.println("SEGURIDAD DESACTIVADO");
        lcd.clear();
        lcd.setCursor(16,0);
        lcd.print(" Bienvenido ");
        lcd.setCursor(8,0);
        lcd.print(" Sr. Torres ");
        delay(2000);
        digitalWrite(22, HIGH);
        port.write('w');
        Serial.print("despierto por teclado");
        delay(2000);
        action();
    }

else{

    //si no es correcta clave ingresada se visualiza en lcd:

```

```

    lcd.noCursor();
    lcd.setCursor(16,0);
    lcd.print("!Pass Incorrect!");
    easyvr.playSound(7, EasyVR::VOL_FULL);
    delay(1000);
    lcd.clear();
    lcd.setCursor(16,0);
    lcd.print("  Vuelva a  ");
    lcd.setCursor(8,0);
    lcd.print(" ingresar clave ");

    delay(1500);
    almacena = 0; //se formatea variable de conteo
    delay(1500);
    teclado();
  }
}

void dormir(){

  digitalWrite(22, LOW);
  Serial.println("en void dormir");
  port.write('s'); // 's' ----> modo standby
  delay(1000);

  Serial.println("standby");
  delay(1000);

  port.write('A'); //'A' -----> se despierta con carácter

  lcd.clear();
  delay(1000);
  despertar();
}

char despertar(){

  char key2 = keypad.getKey();
  Serial.println("en char despertar");
  Serial.print(key2);
  delay(500);

  if(key2=='*'){
    port.write('w');
    delay(1000);
    digitalWrite(22, HIGH);
    Serial.println("TECLADO");
    teclado();
  }

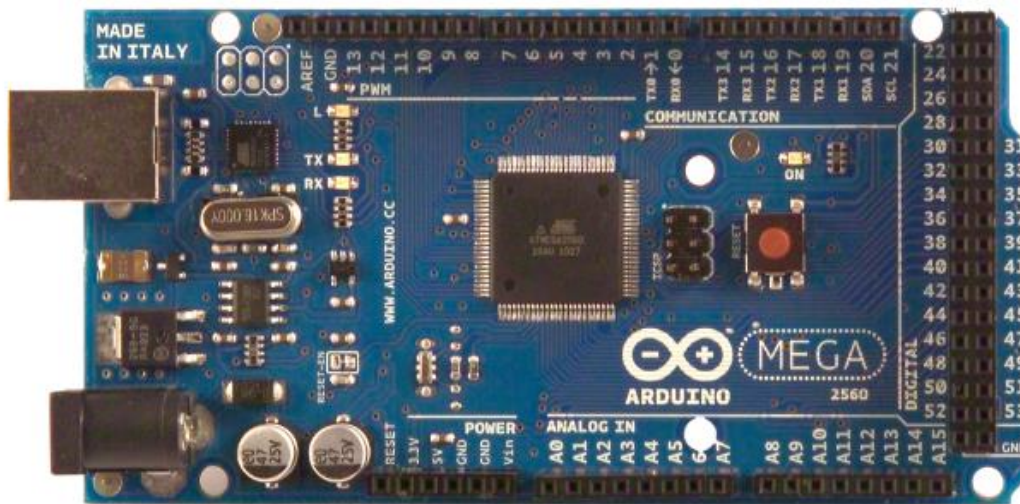
  else{
    Serial.println(" regresando a void despertar");
    delay(500);
    despertar();
  }
}

```

ANEXO III

ARDUINO MEGA 2560

Arduino MEGA 2560



Product Overview

The Arduino Mega 2560 is a microcontroller board based on the ATmega2560 ([datasheet](#)). It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

Index

Technical Specifications

Page 2

How to use Arduino
Programming Environment, Basic Tutorials

Page 6

Terms & Conditions

Page 7

Environmental Policies
half sqm of green via Impatto Zero®

Page 7



radiospares **RADIONICS**



Technical Specification

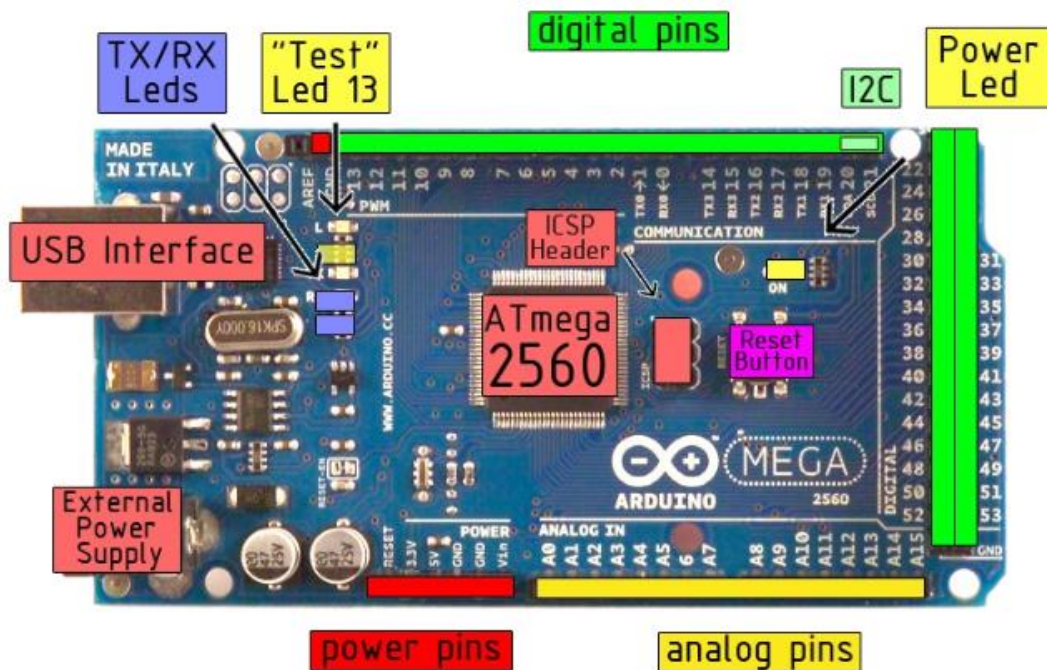


EAGLE files: [arduino-mega2560-reference-design.zip](#) Schematic: [arduino-mega2560-schematic.pdf](#)

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



radiospares

RADIONICS



Power

The Arduino Mega2560 can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The Mega2560 differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The ATmega2560 has 256 KB of flash memory for storing code (of which 8 KB is used for the bootloader), 8 KB of SRAM and 4 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 54 digital pins on the Mega can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX); Serial 1: 19 (RX) and 18 (TX); Serial 2: 17 (RX) and 16 (TX); Serial 3: 15 (RX) and 14 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. Pins 0 and 1 are also connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip .
- **External Interrupts: 2 (interrupt 0), 3 (interrupt 1), 18 (interrupt 5), 19 (interrupt 4), 20 (interrupt 3), and 21 (interrupt 2).** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 0 to 13.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI: 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language. The SPI pins are also broken out on the ICSP header, which is physically compatible with the Duemilanove and Diecimila.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- **I²C: 20 (SDA) and 21 (SCL).** Support I²C (TWI) communication using the [Wire library](#) (documentation on the Wiring website). Note that these pins are not in the same location as the I²C pins on the Duemilanove.

The Mega2560 has 16 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and [analogReference\(\)](#) function.

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.



radiospares

RADIONICS



Communication

The Arduino Mega2560 has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega2560 provides four hardware UARTs for TTL (5V) serial communication. An ATmega8U2 on the board channels one of these over USB and provides a virtual com port to software on the computer (Windows machines will need a .inf file, but OSX and Linux machines will recognize the board as a COM port automatically). The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the board. The RX and TX LEDs on the board will flash when data is being transmitted via the ATmega8U2 chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Mega's digital pins.

The ATmega2560 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation on the Wiring website](#) for details. To use the SPI communication, please see the ATmega2560 datasheet.

Programming

The Arduino Mega2560 can be programmed with the Arduino software ([download](#)). For details, see the [reference](#) and [tutorials](#).

The ATmega2560 on the Arduino Mega comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.



radiospares

RADIONICS



ALLIED ELECTRONICS
AND AUTOMATION

Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Mega2560 is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega2560 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Mega2560 is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Mega2560. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Mega contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

USB Overcurrent Protection

The Arduino Mega has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics and Shield Compatibility

The maximum length and width of the Mega PCB are 4 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.

The Mega is designed to be compatible with most shields designed for the Diecimila or Duemilanove. Digital pins 0 to 13 (and the adjacent AREF and GND pins), analog inputs 0 to 5, the power header, and ICSP header are all in equivalent locations. Further the main UART (serial port) is located on the same pins (0 and 1), as are external interrupts 0 and 1 (pins 2 and 3 respectively). SPI is available through the ICSP header on both the Mega and Duemilanove / Diecimila. **Please note that I²C is not located on the same pins on the Mega (20 and 21) as the Duemilanove / Diecimila (analog inputs 4 and 5).**



radiospares

RADIONICS



How to use Arduino



Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the [Arduino programming language](#) (based on [Wiring](#)) and the Arduino development environment (based on [Processing](#)). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. Flash, Processing, MaxMSP).

Arduino is a cross-platform program. You'll have to follow different instructions for your personal OS. Check on the [Arduino site](#) for the latest instructions. <http://arduino.cc/en/Guide/HomePage>

Linux Install

Windows Install

Mac Install

Once you have downloaded/unzipped the arduino IDE, you can Plug the Arduino to your PC via USB cable.

Blink led

Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

**File>Sketchbook>
Arduino-0017>Examples>
Digital>Blink**

Once you have your sketch you'll see something very close to the screenshot on the right.

In **Tools>Board** select MEGA

Now you have to go to **Tools>SerialPort** and select the right serial port, the one arduino is attached to.

```
int ledPin = 13; // LED connected to digital pin 13

// The setup() method runs once, when the sketch starts

void setup() {
  // initialize the digital pin as an output:
  pinMode(ledPin, OUTPUT);
}

// the loop() method runs over and over again,
// as long as the Arduino has power

void loop()
{
  digitalWrite(ledPin, HIGH); // set the LED on
  delay(1000); // wait for a second
  digitalWrite(ledPin, LOW); // set the LED off
  delay(1000); // wait for a second
}
```



Done compiling.

Press Compile button
(to check for errors)



Upload



TX RX Flashing



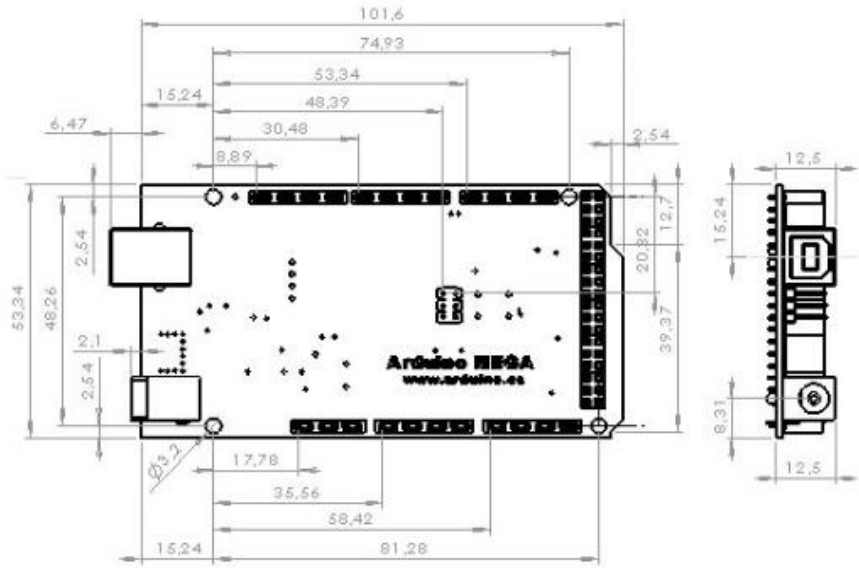
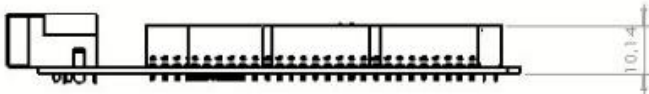
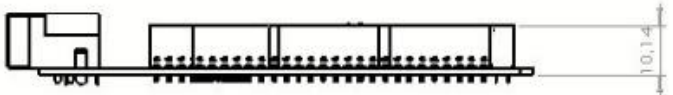
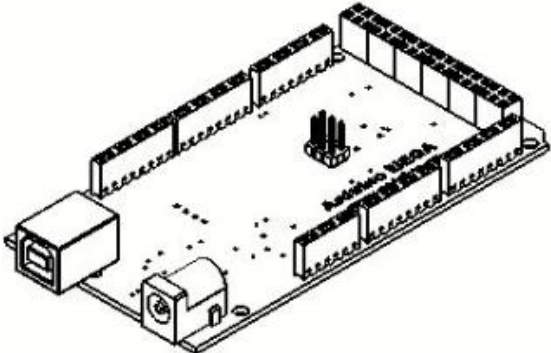
Blinking Led!



radiospares

RADIONICS





radiospares **RADIONICS**



ANEXO IV

EASYVR SHIELD 2.0



EasyVR

User Manual

Release 3.3



www.veear.eu

EasyVR Module

Product Description

EasyVR is a multi-purpose speech recognition module designed to easily add versatile, robust and cost effective speech recognition capabilities to virtually any application.

The EasyVR module can be used with any host with an UART interface powered at 3.3V – 5V, such as PIC and Arduino boards. Some application examples include home automation, such as voice controlled light switches, locks or beds, or adding "hearing" to the most popular robots on the market.

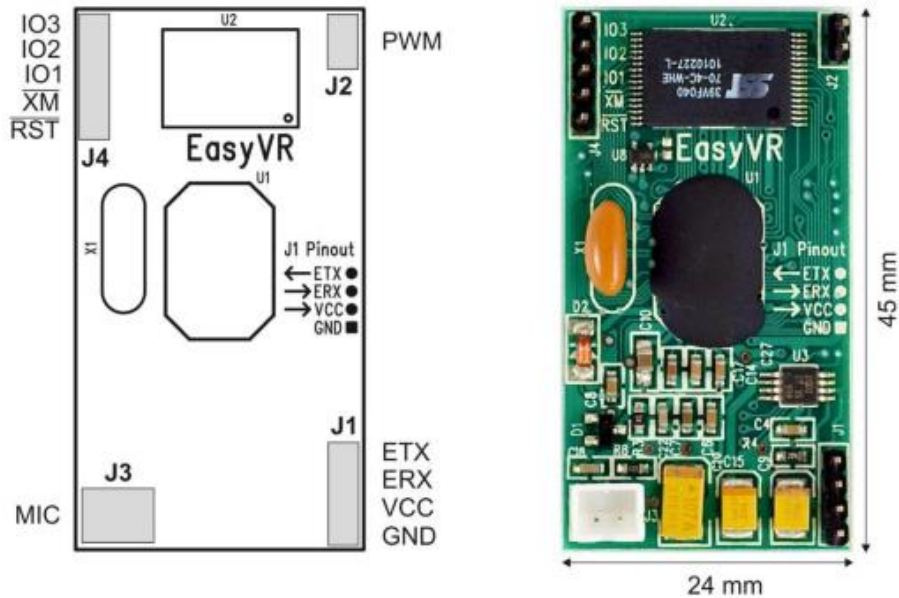


EasyVR features

- A host of built-in Speaker Independent (SI) commands for ready to run basic controls, in the followings languages:
 - English (US)
 - Italian
 - German
 - French
 - Spanish
 - Japanese
- Supports up to 32 user-defined Speaker Dependent (SD) triggers or commands as well as Voice Passwords. SD custom commands can be spoken in ANY language.
- Easy-to-use and simple Graphical User Interface to program Voice Commands and audio.
- Module can be used with any host with an UART interface (powered at 3.3V - 5V)
- Simple and robust documented serial protocol to access and program through the host board
- 3 GPIO lines (IO1, IO2, IO3) that can be controlled by new protocol commands.
- PWM audio output that supports 8Ω speakers.
- Sound playback of up to 9 minutes of recorded sounds or speech.

Technical specifications

Physical dimensions and pin assignment



Connector	Number	Name	Type	Description
J1	1	GND	-	Ground
	2	VCC	I	Voltage DC input
	3	ERX	I	Serial Data Receive (TTL level)
	4	ETX	O	Serial Data Transmit (TTL level)
J2	1-2	PWM	O	Differential audio output (can directly drive 8Ω speaker)
J3	1	MIC_RET	-	Microphone reference ground
	2	MIC_IN	I	Microphone input signal
J4	1	/RST	I	Active low asynchronous reset (internal 100K pull-up)
	2	/XM	I	Boot select (internal 1K pull-down)
	3	IO1	I/O	General purpose I/O (3.0 VDC TTL level)
	4	IO2	I/O	General purpose I/O (3.0 VDC TTL level)
	5	IO3	I/O	General purpose I/O (3.0 VDC TTL level)

Note: the GPIO (J4.3, J4.4, and J4.5) are at nominal 3.0VDC level. Do not connect 5VDC directly to these pins!

Recommended Operating Conditions

Symbol	Parameter	Min	Typ	Max	Unit
VCC	Voltage DC Input	3.3	5.0	5.5	V
Ta	Ambient Operating Temperature Range	0	25	70	°C
ERX	Serial Port Receive Data	0	-	VCC	V
ETX	Serial Port Transmit Data	0	-	VCC	V

Electrical Characteristics

These are applicable to J4 pins only, including IO1-3, /XM and /RST.

Symbol	Parameter	Min	Typ	Max	Unit
V _{IH}	Input High Voltage	2.4	3.0	3.3	V
V _{IL}	Input Low Voltage	-0.1	0.0	0.75	V
I _{IL}	Input Leakage Current (0 < V _{IO} < 3V, Hi-Z Input)		<1	10	µA
R _{PU}	Pull-up Resistance	Strong	10		kΩ
		Weak	200		kΩ
V _{OH}	Output High Voltage (I _{OH} = -5 mA)	2.4			V
V _{OL}	Output Low Voltage (I _{OL} = 8 mA)			0.6	V

Power Supply Requirements

Symbol	Parameter	Min	Typ	Max	Unit
I _{Sleep}	Sleep current		< 1		mA
I _{Oper}	Operating current		12		mA
I _{Speaker}	Audio playback current (with 8Ω speaker)		180		mA (RMS)

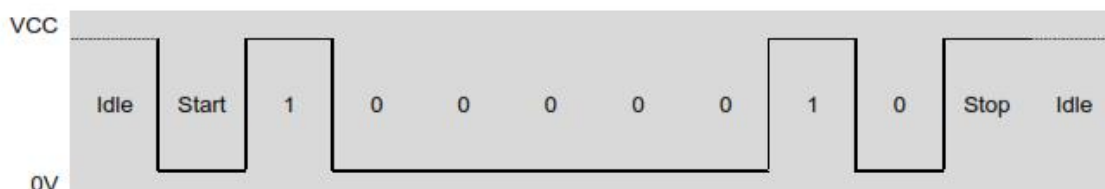
Serial Interface

The EasyVR is a "slave" module communicating via an asynchronous serial interface (commonly known as UART interface), with the following features:

- Baud Rate: **9600** (default), 19200, 38700, 57600, 115200
- Frame: **8** Data bits, **No** parity, **1** Stop bit

The receiver input data line is ERX, while the transmitter output data line is ETX. No handshake lines are used.

Example of a serial data frame representing character "A" (decimal 65 or hexadecimal 41):



See also chapter [Communication Protocol](#) later on this manual for communication details.

Microphone

The microphone provided with the EasyVR module is an omnidirectional electret condenser microphone (Horn EM9745P-382):

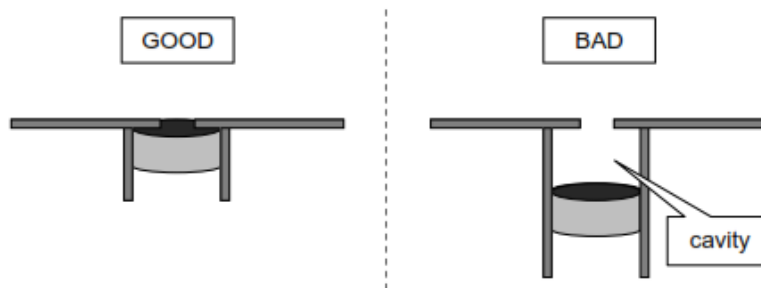
- Sensitivity -38dB (0dB=1V/Pa @1KHz)
- Load Impedance 2.2K
- Operating Voltage 3V
- Almost flat frequency response in range 100Hz – 20kHz

If you use a microphone with different specifications the recognition accuracy may be adversely affected. No other kind of microphone is supported by the EasyVR.

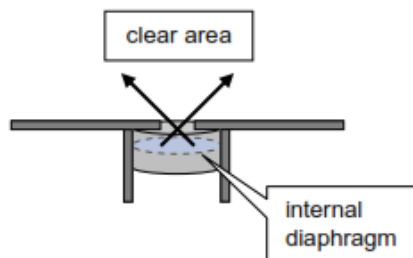
Note: Vocal commands should be given from about 60cm from the microphone, but you can try at greater distances by talking louder.

Please note that improper acoustic positioning of the microphone will reduce recognition accuracy. Many mechanical arrangements are possible for the microphone element, and some will work better than others. When mounting the microphone in the final device, keep in mind the following guidelines:

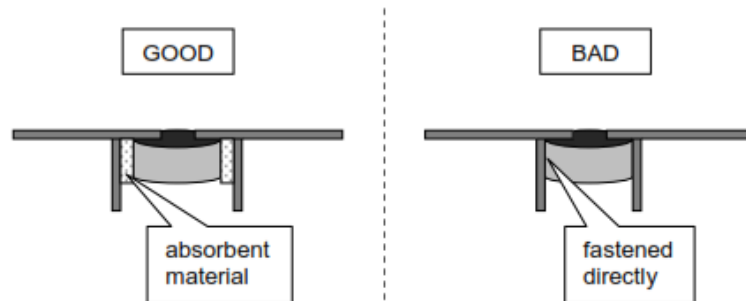
1. **Flush Mounting** - The microphone element should be positioned as close to the mounting surface as possible and should be fully seated in the plastic housing. There must be no airspace between the microphone element and the housing. Having such airspace can lead to acoustic resonance, which can reduce recognition accuracy.



2. **No Obstructions, Large Hole** - The area in front of the microphone element must be kept clear of obstructions to avoid interference with recognition. The diameter of the hole in the housing in front of the microphone should be at least 5 mm. Any necessary plastic surface in front of the microphone should be as thin as possible, being no more than 0.7 mm, if possible.



3. **Insulation** - The microphone should be acoustically isolated from the housing if possible. This can be accomplished by surrounding the microphone element with a spongy material such as rubber or foam. The provided microphone has this kind of insulating foam. The purpose is to prevent auditory noises produced by handling or jarring the device from being "picked up" by the microphone. Such extraneous noises can reduce recognition accuracy.



4. **Distance** - If the microphone is moved from 15 cm to 30 cm from the speaker's mouth, the signal power decreases by a factor of four. The difference between a loud and a soft voice can also be more than a factor of four. Although the internal preamplifier of the EasyVR compensates for a wide dynamic range of input signal strength, if its range is exceeded, the user application can provide feedback to the speaker about the voice volume (see appendix [Error codes](#)).

Audio Output

The EasyVR audio output interface is capable of directly driving an 8Ω speaker. It could also be connected to an external audio amplifier to drive lower impedance loudspeakers.

Note: Connecting speakers with lower impedance directly to the module may permanently damage the EasyVR audio output or the whole module.

It is possible to connect higher impedance loads such as headphones, provided that you scale down the output power according to the speaker ratings, for example using a series resistor. The exact resistor value depends on the headphone power ratings and the desired output volume (usually in the order of 10kΩ).

Note: Connecting headphone speakers directly to the EasyVR audio output may damage your hearing.

General Purpose I/O

Since the EasyVR communication interface takes two pins of the host controller, a few spare I/O pins are provided, that can be controlled with the communication protocol, to get those pins back for basic tasks, such as lighting an LED.

The three I/O pins IO1–IO3 are connected directly to the embedded microcontroller on the EasyVR module, so they are referenced to the internal 3.0V regulated power supply. If you need to interface to circuits using a different supply, there are a number of solutions you can adopt. Some of these are outlined below (here IO1 indicates any one of the three I/O pins of the EasyVR).

Use a pin as an output

All the I/O pins are inputs with weak internal pull-up after power on. You must explicitly configure a pin before you can use it as an output (see the example code [Use general purpose I/O pins](#)).

Quick start for using the module

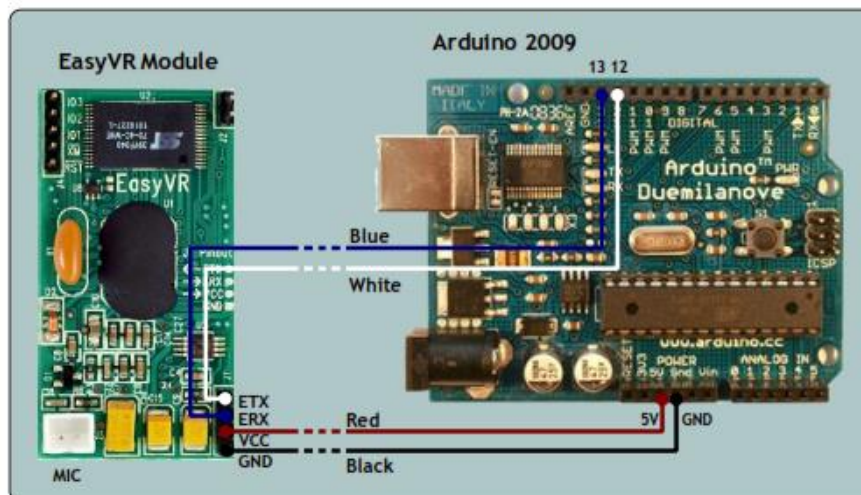
EasyVR on Arduino

You can connect the EasyVR module to an Arduino board basically in two ways:

1. **Bridge mode** – You can control the module using a software serial library and connect to the module with the EasyVR Commander from your PC, with the same pin configuration
2. **Adapter mode** – You can use the Arduino board as a USB/Serial adapter by holding the microcontroller in reset, but you need to change the connections once you want to control the module from the microcontroller

Bridge mode

This is the preferred connection mode, since it allows simple communication with both the Arduino microcontroller and the PC. All the provided examples for Arduino manage the bridge mode automatically when the EasyVR Commander requests a connection.



Automatic bridge mode used to be supported only on Arduino boards with a bootloader implementing EEPROM programming.

The latest version of EasyVR Commander (since 3.1.x) and Arduino libraries (since 1.1) does not rely on that feature anymore, so it should work on all Arduino boards.

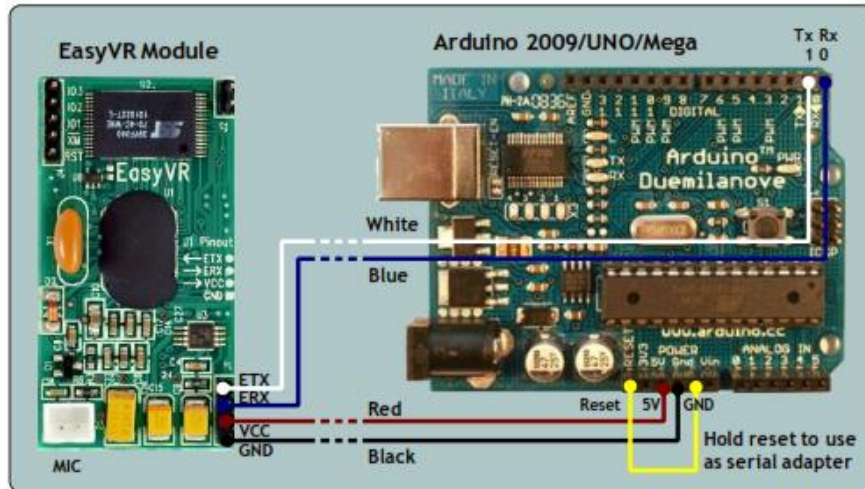
Note: bridge mode cannot be used to download a Sound Table or to perform a flash update. You need to use *adapter mode* or a true USB/Serial adapter.

Adapter mode

This connection scheme has the advantage of working with any Arduino board that has an on-board USB/Serial adapter and not needing a spare input pin to enter bridge mode.

Also, it does not rely on the AVR microcontroller to do any software bridge between communication pins, so it can be used to check your hardware in case of connection problems.

Using this method also allows you to download a Sound Table to the EasyVR module, provided you also configure the module to start in *boot mode* (see paragraph [Flash Update](#)).



This configuration, with Reset shorted to GND, is for connection with the EasyVR Commander. To use the module from the Arduino microcontroller, you need to remove the short (yellow wire) and move the ETX/ERX connection to other pins. The example code uses pin 12 for ETX and pin 13 for ERX, like the above bridge mode.

ANEXO V

PANTALLA GLCD 12864ZW

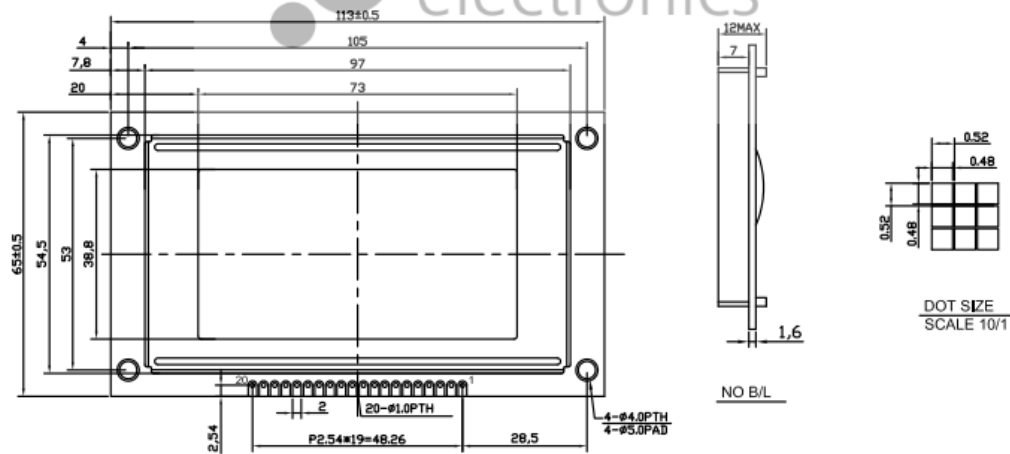
Chapter1. 12864 Graphic LCD Module Brief

Introduction

Physical Data

Item	Contents	Unit
LCD type	STN	---
LCD duty	1/64	---
LCD bias	1/9	---
Viewing direction	6	o'clock
Module size (W×H×T)	112× 66	mm
Viewing area (W×H)	73.4 × 38.8 (2.89" × 1.53")	mm
Number of dots	128 × 64	dots
Dot size (W×H)	0.48 × 0.48 (0.019" × 0.019")	mm
Dot pitch (W×H)	0.52 × 0.52 (0.020" × 0.020")	mm

External Dimensions



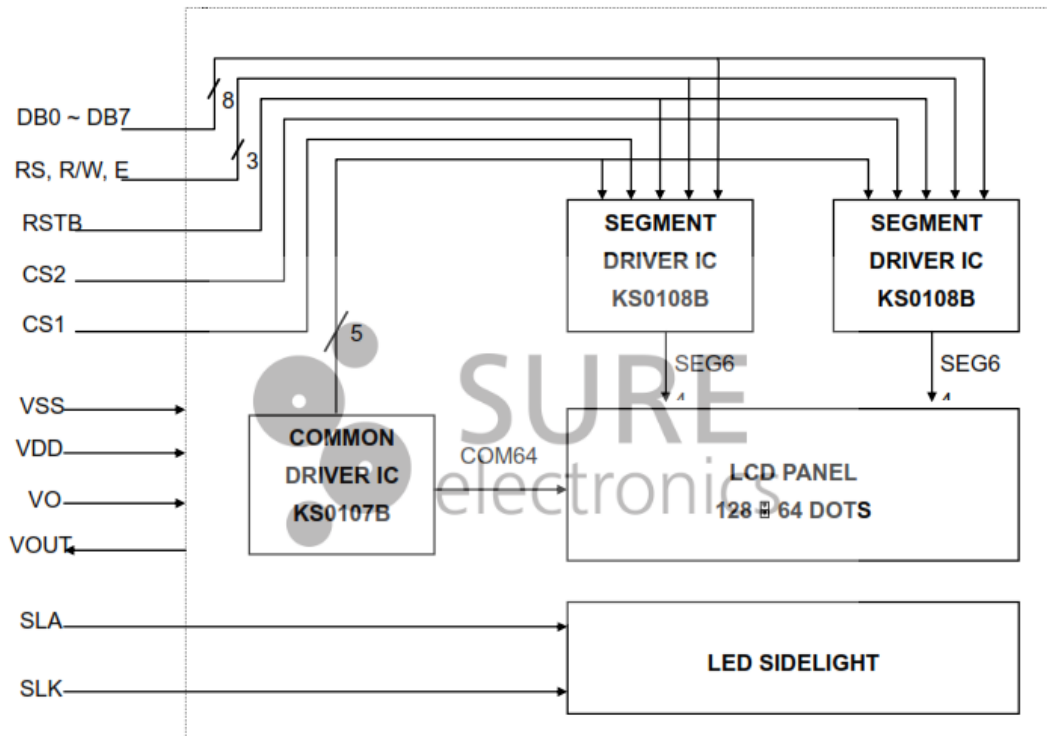
Mechanical Data

Item	Standard Value	Unit
Module Dimension	113×65	mm
Viewing Area	73×38.8	mm
Mounting hole	105×53	mm
Dot pitch	0.52×0.52	mm

12864 Graphic LCD Module Brief Introduction

Block Diagram

20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
SL	SL	VO	RS	CS	CS	DB	DB	DB	DB	DB	DB	DB	DB	E	R/W	RS	VO	VD	VS
K	A	UT	TB	2	1	7	6	5	4	3	2	1	0					D	S



Absolute Maximum Ratings (Ta = 25°C)

Parameter	Symbol	Min	Max	Unit
Supply voltage for logic	VDD	-0.3	7.0	V
Supply voltage for LCD	VDD - VO	-0.3	VDD+0.3	V
Input voltage	VI	-0.3	VDD+0.3	V
Operating temperature	TOP	0	50	°C
Storage temperature	TST	-10	60	°C

12864 Graphic LCD Module Brief Introduction

Electrical Characteristics (VDD = +5V±5%, VSS = 0V, Ta = 25°C)

Parameter	Symbol	Condition	Min	Typ	Max	Unit
Supply voltage for logic	VDD	---	4.5	5.0	5.5	V
Supply current for logic	IDD	---	---	2.02	4	mA
Operating voltage for LCD	VDD - VO	0°C	8.4	8.8	9.2	V
		25°C	8.1	8.5	8.9	V
		50°C	7.4	7.8	8.2	V
Supply voltage for sidelight	VF	---	---	4.2	4.6	V
Supply current for sidelight	IF	VF=4.2V	---	180	300	mA
Input voltage 'H' level	VIH	---	0.7VDD	---	VDD	V
Input voltage 'L' level	VIL	---	0	---	0.3VDD	V

Interface Pin Connections

Pin No.	Symbol	Level	Description
1	VSS	0V	Ground
2	VDD	5.0V	Supply voltage for logic
3	VO	---	Input voltage for LCD
4	RS	H/L	H : Data signal, L : Instruction signal
5	R/W	H/L	H : Read mode, L : Write mode
6	E	H, H → L	Chip enable signal
7	DB0	H/L	Data bit 0
8	DB1	H/L	Data bit 1
9	DB2	H/L	Data bit 2
10	DB3	H/L	Data bit 3
11	DB4	H/L	Data bit 4
12	DB5	H/L	Data bit 5
13	DB6	H/L	Data bit 6
14	DB7	H/L	Data bit 7
15	CS1	H	Chip select signal for KS0108B(1)
16	CS2	H	Chip select signal for KS0108B(2)
17	RSTB	L	Reset signal
18	VOOUT	-5V	Output voltage for LCD
19	SLA	4.2V	Sidelight anode
20	SLK	0V	Sidelight cathode