

RAE

1. **TIPO DE DOCUMENTO:** Trabajo de grado para optar por el título de INGENIERO DE SONIDO.
2. **TÍTULO:** DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE ILUMINACIÓN DOMÓTICO AUTOMATIZADO POR COMANDOS DE VOZ
3. **AUTORES:** Daniel Ricardo Torres Espinosa y Edwin Ángel Angarita puentes.
4. **LUGAR:** Bogotá D.C.,
5. **FECHA:** Junio de 2012.
6. **PALABRAS CLAVE:** Reconocimiento de voz, MatLab, Formantes, interface, Arduino, simulink, domótica.
7. **DESCRIPCIÓN DEL TRABAJO:** El uso de comandos verbales ofrece un amplio campo de aplicación convirtiéndose en la base para el diseño de un sistema domótico de iluminación, el cual le permitirá al beneficiario desempeñar labores básicas del hogar tan sencillas como lo son el encender y apagar las luces, a través del sistema eléctrico de una vivienda sin tener la necesidad de realizar un desplazamiento físico al momento de accionar el interruptor, e inclusive para activar o desactivar todas luces con un solo comando desde cualquier lugar en donde haya cubrimiento del sistema.
El funcionamiento de este sistema trabajara a partir del reconocimiento automático de palabras aisladas por parte de un usuario, logrando cumplir las funciones anteriormente especificadas por medio de un software, para que en el momento de reconocer y aceptar el comando, envíe la información por medio de una interfaz al sistema eléctrico encargado de la iluminación de la maqueta o espacio propuesto para este proyecto.
8. **LÍNEAS DE INVESTIGACIÓN:** Universidad de San Buenaventura/Tecnologías Actuales y Sociedad. Sub línea de facultad de ingeniería: Procesamiento de señales digitales y/o analógicas. Campo temático del programa: Diseño de sistema de sonido.
9. **FUENTES CONSULTADAS:** HENAO, Oscar. "Hardware y Software domótico" Universidad Pontificia Bolivariana. Colombia. 2006., BERNAL BERMÚDEZ. "Reconocimiento de voz y fonética acústica". Madrid, España. 2000., RAMIREZ, Ángela, BOJACA, Julián. "Diseño de un modulo para el control de iluminación de hardware domótico a través de un teléfono celular en un ambiente distribuido." Universidad El Bosque. Bogota D.C., Colombia. 2007., Gemello, R., Albesano D., Moisa L. and de Mori R., "Integration of Fixed and Multiple Resolution Analysis in a Speech Recognition System", ICASSP 2001, 2001., Biem A., Katagiri S., McDermott E. and Juang, B. H., "An Application of Discriminative Feature Extraction to Filter-Bank-Based Speech Recognition", IEEE Trans. Speech and Audio Proc., 9 (2), pp. 96-110, febrero 2001., Ferreiros, J., "Aportación a los Métodos de Entrenamiento de Modelos de Markov para Reconocimiento de Habla", Tesis Doctoral, TESIT, Universidad Politécnica de Madrid, 1996., Bernal, J. "METODOLOGIA PARA LA VISUALIZACIÓN DE LOS FORMANTES, BASADA EN LA TRANSFORMADA DE FOURIER, Y SU USO EN EL ESTUDIO ESPECTRAL DE LA FONÉTICA ACÚSTICA ESPAÑOLA", Universidad Politécnica de Madrid, España 1999.
10. **CONTENIDOS:** MATLAB es un programa de cálculo técnico y científico, cuando puede ejecutar sus funciones en código nativo con los tamaños más adecuados para aprovechar sus capacidades de vectorización. En otras aplicaciones resulta bastante más lento que el código equivalente desarrollado en C/C++ o Fortran. Arduino es una plataforma de electrónica abierta para la creación de prototipos basada en software y hardware flexibles y fáciles de usar. Se creó para artistas, diseñadores, aficionados. El reconocimiento a través de características fisiológicas se realiza a partir de las medidas físicas de partes del cuerpo humano (huellas dactilares, mano, iris). Mientras que las basadas en el comportamiento tienen en cuenta cómo realiza cada persona determinadas acciones (voz, firma, escritura en teclado). La interfaz gráfica de usuario, es una aplicación proporcionada por MatLab que permite generar la relación con el usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador.
11. **METODOLOGÍA:** Es de carácter empirico-analitico, con un enfoque metodológico con base en el estudio y diseño de un sistema de reconocimiento de voz implementado en domótica.
12. **CONCLUSIONES:** El reconocimiento de voz para este algoritmo es limitado, debido a que solamente se permite la opción de dos sílabas, además existen formantes de la voz humana que son muy similares en palabras con la misma composición vocal (ejemplo: casa, sala, cama). Es indispensable tener una etapa de entrenamiento al momento de programar el algoritmo ya que el software encargado de reconocimiento debe asimilar la totalidad de las formantes y demás propiedades de la voz del locutor; así mismo el usuario debe utilizar un lenguaje verbal fluido sin léxicos desconocidos, una buena pronunciación y entonación. Además es indispensable garantizar el uso de un micrófono con respuesta en frecuencia que este en el rango de la voz humana y tenga una relación señal ruido excelente. Es importante realizar la comparación de rangos entre cada comando de voz ya que esto permite evaluar qué posibilidades existen para validar dos o más comandos al tiempo o de no ser así puede ayudar a ampliar el rango de posibilidades y de esta manera aumentar porcentaje de efectividad del mismo. En conclusión es posible reconocer voz en base a un software, y que este mismo se encargue de ordenar por puerto USB la ejecución de diferentes tareas en el mundo real. Además la efectividad del diferente software de reconocimiento de voz es directamente proporcional a aspectos inherentes del locutor como la fuerza de pronunciación, el estado anímico, la correcta pronunciación o enfermedades bucofaringeas que impidan la correcta entonación de las palabras. El porcentaje de error para este algoritmo de reconocimiento de voz varía en un rango entre 72% y 78%.

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE ILUMINACIÓN
DOMÓTICO AUTOMATIZADO POR COMANDOS DE VOZ**

DANIEL RICARDO TORRES ESPINOSA

EDWIN ÁNGEL ANGARITA PUENTES

UNIVERSIDAD DE SAN BUENAVENTURA

FACULTAD DE INGENIERÍA

PROGRAMA INGENIERÍA DE SONIDO

BOGOTÁ D.C.

2012

**DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE ILUMINACIÓN
DOMÓTICO AUTOMATIZADO POR COMANDOS DE VOZ**

PRESENTADO POR:

DANIEL RICARDO TORRES E.

ing.daniel.torres@hotmail.com

EDWIN ÁNGEL ANGARITA PUENTES

edwinangar18@hotmail.com

UNIVERSIDAD DE SAN BUENAVENTURA

FACULTAD DE INGENIERÍA

PROGRAMA INGENIERÍA DE SONIDO

BOGOTÁ D.C.

2012

Nota de aceptación

**Firma del jurado
Nelson Rosas**

**Firma del jurado
Mijail Posada**

Bogotá D.C, 2012.

AGRADECIMIENTOS

Quiero hacer constar mi agradecimiento antes que nada a DIOS, por llevarme a su lado a lo largo de esta vida siempre llenándola de alegría y gozo al permitirme alcanzar un triunfo más, a mi madre Blanca Puentes por esos años de entrega, desvelos y apoyo a lo largo de mi tesis, formación educativa, y sobre todo en mi vida diaria. A mi padre Horacio Angarita por alentarme a hacer lo que quiero y ser como soy, es en este momento donde no puedo dejar de recordar tu cercanía, devoción y ejemplo, a todos mis amigos, compañeros y maestros en la Universidad de San Buenaventura, que siempre estarán en mis recuerdos.

MIL GRACIAS A TODOS.

EDWIN ÁNGEL ANGARITA PUENTES

Quiero dar gracias a Dios y a la virgen por regalarme este título y alumbrar mi camino siempre; a mi mama Liliana Sofía Torres Espinosa quien aparte de darme sus apellidos, hizo todo esto posible; fue quien me acompañó en madrugadas y traspasos durante este largo proceso; a mis tíos y mi abuelos quienes nunca me dejaron desfallecer y siempre me impulsaron a seguir adelante, a mi hermanita y mis primos que inconscientemente me impulsaron a darles ejemplo, a mis amigos y compañeros por sus consejos y compañía, a todos los maestros que contribuyeron en mi formación y a todas y cada una de esas personas que alguna vez elevaron una oración por mí.

GRACIAS

DANIEL RICARDO TORRES ESPINOSA

TABLA DE CONTENIDO

	Pág.
INTRODUCCIÓN	10
1. PLANTEAMIENTO DEL PROBLEMA	11
1.1. ANTECEDENTES	11
1.1.1. Estado del arte	11
1.1.2. Antecedentes teóricos	13
1.2. DESCRIPCIÓN DEL PROBLEMA	14
1.3. JUSTIFICACIÓN	14
1.4. OBJETIVOS	15
1.4.1. Objetivo general	15
1.4.2. Objetivos específicos	15
1.5. ALCANCES	15
1.6. LIMITACIONES	15
2. METODOLOGÍA	16
2.1. ENFOQUE DE LA INVESTIGACIÓN	16
2.2. LÍNEA DE INVESTIGACIÓN / SUBLÍNEA DE LA FACULTAD / CAMPO TEMÁTICO DEL PROGRAMA	16
2.2.1. Línea de Investigación de la Universidad: Tecnologías Actuales y Sociedad	16
2.3. SUBLINEAS DE INVESTIGACION DE LA FACULTAD	17
2.3.1. Procesamiento de señales digitales y/o analógicas	17
2.4. CAMPO DE LA FACULTAD	18
2.4.1. Diseño de sistema de sonido	18
2.5. RECOLECCIÓN DE INFORMACIÓN	20
2.6. HIPÓTESIS	20
2.7. VARIABLES	20
2.7.1. Variables dependientes	20
2.7.2. Variables independientes	20
3. MARCO DE REFERENCIA	21
3.1. MARCO TEÓRICO	21
3.1.1. MATLAB	21
3.1.1.1. Simulink	21
3.1.1.2. Interfaz gráfica de usuario (GUI)	22
3.1.2. ARDUINO	22
3.1.2.1. Arduino UNO	23

3.1.3. D.S.P (Procesamiento Digital de la señal)	23
3.1.3.1. Captura de la señal	23
3.1.3.2. Sistemas de reconocimiento biométrico	23
3.1.4. RECONOCIMIENTO DELA VOZ	24
3.1.4.1. Reconocimiento Automático del habla	24
3.1.4.2. Descripción de un sistema de RAH	24
3.1.4.3. Parametrización o formantes	25
3.2. MARCO LEGAL Y NORMATIVO	26
4. DESARROLLO INGENIERIL	28
4.1. SOFTWARE	28
4.2. HARDWARE IMPLEMENTADO	28
4.3. SELECCIÓN DE COMANDOS	31
4.4. ELABORACIÓN DE ALGORITMO	38
4.5. INTERFAZ GRAFICA	43
4.6. MAQUETA	44
5. ANÁLISIS DE RESULTADOS	50
5.1. PRUEBA DE ENSAYO Y ERROR	50
6. CONCLUSIONES Y RECOMENDACIONES	58
6.1. CONCLUSIONES	58
6.2. RECOMENDACIONES	59
BIBLIOGRAFIA	61
REFERENCIAS DE PAGINA WEB	62
ANEXO A	63
ANEXO B	78
ANEXO C	80

TABLA DE IMÁGENES

Imagen 4.1. Arduino	29
Imagen 4.2. Mapa del circuito eléctrico	30
Imagen 4.3. Opto acoplador	30
Imagen 4.3. Circuito final	31
Imagen 4.4.1. Diagrama de flujo	37
Imagen 4.4.2. Diagrama de flujo 2	38
Imagen 4.4.3. Diagrama de flujo 3	38
Imagen 4.4.4. Diagrama de flujo 4	39
Imagen 4.4.5. Diagrama de flujo 5	39
Imagen 4.4. Diagrama de flujo final	41
Imagen 4.5. Diagrama de bloques	42
Imagen 4.6. Interface gráfica	43
Imagen 4.6.1. Interface gráfica – Botón 1	43
Imagen 4.6.2. Interface gráfica – Botón 2	43
Imagen 4.6.3. Interface gráfica – Botón 3	44
Imagen 4.7. Planta	45
Imagen 4.8. Diseño 3D	45
Imagen 4.9. Render vista 1	46
Imagen 4.10. Render vista 2	46
Imagen 4.11. Arduino acoplado en el techo	47
Imagen 4.12. Maqueta construida cielo raso	47
Imagen 4.13. Maqueta construida fachada	48
Imagen 4.14. Maqueta construida planta	48
Imagen 4.15. Sistema funcionando	49
Imagen 4.15. Producto final	49

TABLAS

Tabla 1	Sala	33
Tabla 2	Jardín	33
Tabla 3	Balcón	34
Tabla 4	Piso	34
Tabla 5	Mesón	35
Tabla 6	Tina	35
Tabla 7	Uno	36
Tabla 8	Día 1	51
Tabla 9	Día 2	53
Tabla 10	Día 3	53
Tabla 11	Día 4	54
Tabla 12	Prueba y error	56
Tabla 13	Porcentaje total	57

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE ILUMINACIÓN DOMÓTICO AUTOMATIZADO POR COMANDOS DE VOZ

INTRODUCCIÓN

La comunicación verbal entre las personas y los software se ha incrementado durante las últimas décadas con el fin de sustituir de cierta forma las interfaces tradicionales como lo son teclados, paneles o dispositivos similares con el fin de incorporar un control que realice estas funciones de una manera más precisa, ahora se presenta un sistema de fácil control que ofrece comodidad durante su vida cotidiana; en situaciones en las cuales, la persona carece de tener sus manos libres, en ambientes donde se posee poca visibilidad o cuando el contacto manual es de difícil acceso o imposible para el control de un dispositivo.

El uso de comandos verbales ofrece un amplio campo de aplicación convirtiéndose en la base para el diseño de un sistema domótico de iluminación, el cual le permitirá al beneficiario desempeñar labores básicas del hogar tan sencillas como lo son el encender y apagar las luces, a través del sistema eléctrico de una vivienda sin tener la necesidad de realizar un desplazamiento físico al momento de accionar el interruptor, e inclusive para activar o desactivar todas luces con un solo comando desde cualquier lugar en donde haya cubrimiento del sistema.

El funcionamiento de este sistema trabajara a partir del reconocimiento automático de palabras aisladas por parte de un usuario, logrando cumplir las funciones anteriormente especificadas por medio de un software, para que en el momento de reconocer y aceptar el comando, envíe la información por medio de una interfaz al sistema eléctrico encargado de la iluminación de la maqueta o espacio propuesto para este proyecto.

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE ILUMINACIÓN DOMÓTICO AUTOMATIZADO POR COMANDOS DE VOZ

1. PLANTEAMIENTO DEL PROBLEMA

1.1. ANTECEDENTES

1.1.1. Estado del arte

Los orígenes del RAH se remontan a los años 40, momento en el que se desarrollan los primeros espectrógrafos, que permitían observar el espectrograma de una señal, es decir, la evolución temporal de la energía en las distintas bandas de frecuencia, dato que podía servir para caracterizar y reconocer la voz humana. El primer dispositivo automático de reconocimiento, desarrollado en 1952 para distinguir los diez dígitos ingleses pronunciados de forma aislada por un único locutor, se basaba en identificar las frecuencias de resonancia de la parte vocálica de los dígitos. En otro trabajo paralelo e independiente (Laboratorios RCA, 1956) se trataba de reconocer sílabas mediante distancias espectrales obtenidas a partir de un banco de filtros analógico. Otros trabajos de la época también se basaban en dispositivos analógicos que obtenían información acerca del contenido espectral de las señales, y utilizaban como criterio de clasificación las frecuencias de resonancia de las vocales.¹

En los años 60 se publicaron ideas fundamentales sobre reconocimiento de patrones. De hecho, los primeros trabajos que emplearon medios informáticos aplicados al RAH datan de esta época. Estos trabajos se centraban en el reconocimiento de palabras aisladas mono locutor, y utilizaban técnicas de programación dinámica para comparar la secuencia de vectores de entrada, mediante alineamiento temporal no lineal («Dynamic Time Warping», DTW), con los patrones de las palabras del diccionario. Además de RCA y AT&T, entran en escena los laboratorios japoneses de NEC, a los que se suman los trabajos realizados en la Carnegie Mellon University (CMU), que continuarán hasta nuestros días.

En los años 70, parcialmente solucionada la tarea de reconocimiento de palabras aisladas, se atacó con optimismo la tarea de reconocimiento de discurso continuo.

¹ PERSPECTIVA DESDE UN GRUPO DE INVESTIGACIÓN DEL PAIS VASCO. Reconocimiento automático del habla. <http://www.euskomedia.org/PDFAnlt/congresos/13/13317323.pdf>, (1996).

Perfilaban dos aproximaciones al problema: los Modelos Estructurales Estocásticos (MEE) y los Sistemas Basados en el Conocimiento (SBC). Se abordaron grandes proyectos de investigación, de los cuales el más ambicioso y conocido de la historia del RAH -que tomaba como referencia la segunda de las aproximaciones-, fue el proyecto ARPASUR («Advanced Research Projects Agency - Speech Understanding System»), iniciado en 1971 y financiado por el Departamento de Defensa de los Estados Unidos, proyecto que nunca alcanzó sus objetivos pero que contribuyó en gran medida a conocer los mecanismos de producción del habla y a tomar conciencia sobre la verdadera magnitud del problema. En IBM se forma también un grupo de reconocimiento del habla, que ataca varias tareas sobre grandes vocabularios. En este grupo se apuesta desde el principio por sistemas estadísticos-probabilísticos basados en aprendizaje inductivo. Mientras tanto, en AT&T continúan las investigaciones con palabras aisladas y DTW, ahora tratando de obtener sistemas independientes del locutor, para lo cual se desarrollan algoritmos de agrupamiento de muestras que generen patrones robustos.

Demostrada en los primeros 80 la ineficacia de los SBC, se invierte todo el esfuerzo en desarrollar sistemas capaces de extraer conocimiento de forma inductiva, es decir, a partir de muestras. Se utiliza a partir de entonces, siguiendo los trabajos de IBM, una modelización acústica basada en Modelos Ocultos de Markov, discretos y continuos, y se optimizan los algoritmos de aprendizaje para entrenar los sistemas a partir de grandes bases de datos. Se mejoran también los sistemas de DTW para el reconocimiento de palabras conectadas, más concretamente se desarrollan algoritmos de búsqueda eficientes con los que determinar la secuencia optima de patrones para una secuencia de vectores acústicos.

Mediada la década de los 80 se presenta la aproximación conexionista como alternativa a la aproximación estadístico- probabilística. Las redes neuronales artificiales («Artificial Neural Networks», ANN) comparten con los HMMs su carácter inductivo, es decir, el aprendizaje a partir de muestras, pero sus configuraciones clásicas -como los perceptrones multicapa («Multi-Layer Perceptron», MLP)- no son capaces de representar fenómenos dinámicos como la señal de voz, por lo cual tuvieron que desarrollarse arquitecturas recursivas específicas, con objeto de superar estas limitaciones. Otros autores han optado

por configuraciones híbridas en las que un MLP es utilizado para estimar las probabilidades de emisión de un HMM.²

Actualmente la investigación se concentra, por un lado, en mejorar el rendimiento de la modelización acústica (generación automática de unidades acústicas contextuales, entrenamiento discriminativo de los modelos) y, por otro lado, en la integración de niveles de conocimiento superiores (estrategias de búsqueda heurísticas en grandes autómatas que representan modelos del lenguaje). También se está invirtiendo un gran esfuerzo en diseñar y adquirir grandes bases de datos para el entrenamiento de sistemas de reconocimiento de discurso continuo.

1.1.2. Antecedentes teóricos

- En el 2006 Oscar David Henao Merchán presento como proyecto de grado titulado **HADWARE Y SOFTWARE DOMÓTICO**, a la facultad de ingeniería eléctrica y electrónica de la universidad pontificia bolivariana en la ciudad de Medellín, en la cual describe las bases del diseño de sistemas domóticos.
- En diciembre de 2007 los estudiantes Ángela Viviana Ramírez Acevedo y Julián Augusto Bojaca Medina presentaron el proyecto titulado **DISEÑO DE UN MODULO PARA EL CONTROL DE ILUMINACIÓN DE UN HARDWARE DOMOTICO A TRAVÉS DE UN TELEFONO CELULAR EN UN AMBIENTE DISTRIBUIDO**; a la facultad de ingeniería de sistemas, especializada en el área de tecnología de telecomunicaciones de la universidad del bosque, en la ciudad de Bogotá D.C. basados en la implementación de un sistema domótico de iluminación controlado por un teléfono celular.
- En el año 2008 en la Universidad de San Buenaventura, sede Bogotá el estudiante Nicolai Abdón Becerra Rojas, presento el proyecto de grado titulado **DISEÑO E IMPLEMENTACIÓN DE UN CONTROL LUMÍNICO POR MEDIO DE LA VOZ**, fundamentado en el reconocimiento de voz para controlar el nivel de intensidad de la luz.

² HILDA YELITZA CONTRERAS. Procesamiento del lenguaje natural basado en una “gramática de estilos” para el idioma español. http://www.saber.ula.ve/bitstream/123456789/13157/1/hc_propuestatesis.pdf, (2001).

1.2. DESCRIPCIÓN DEL PROBLEMA

En la actualidad se puede observar muy fácilmente como los avances tecnológicos determinan el hábitat de los seres humanos en su vida cotidiana, implementando la automatización de distintos dispositivos que están al alcance de sus manos, para el desarrollo de determinadas actividades diarias, de esta manera se busca facilitar labores comunes por medio de dichos dispositivos y facilitarlas en caso de existir una discapacidad o personas con necesidades especiales.

Estas labores podrían facilitarse en nuestras vidas si simplemente con un aplauso o decir nuestro nombre se ejecutaran las funciones como prender la luz, cambiar nuestro entorno visual entre otros.

La implementación de un software que por medio de una interfaz sea capaz de controlar algunas de nuestras actividades en el mundo real, como lo es el encender o apagar una luz con simplemente decir una palabra. Todo esto nos lleva a plantearnos la siguiente pregunta.

¿Es posible diseñar un dispositivo capaz de reconocer la voz de un usuario y que cumpla funciones específicas dentro de un sistema eléctrico?

1.3. JUSTIFICACIÓN

El reconocimiento de voz es un avance tecnológico que ha tomado mucha importancia en actualidad, por otra parte la automatización domótica en Colombia se ha venido implantando, innovando técnicas en todos los sistemas de los hogares ofreciendo al público sistemas que facilitan las labores, aumentando su comodidad y calidad de vida.

La investigación es muy conveniente para adaptar nuevas tecnologías de automatización por medio de reconocimiento de voz además incentiva a iniciar nuevas investigaciones sobre el tema.

La proyección social que tiene el proyecto es aumentar la calidad de vida de los usuarios por medio de la implementación de este tipo de sistemas, apoyados en el uso de la voz.

Así mismo el proyecto proporciona a las personas con deficiencias físicas un mejor estilo de vida ya que se podrían valer de sí mismos para realizar ciertas acciones que no le es posible realizar en la actualidad.

1.4. OBJETIVOS

1.4.1. Objetivo general

- Diseñar un sistema de iluminación automatizado por medio de comandos de voz para un usuario, implementado en la maqueta de una vivienda.

1.4.2. Objetivos específicos

- Diseñar el sistema de reconocimiento de voz para mínimo 5 comandos.
- Diseñar y construir la interfaz que permita la comunicación entre el sistema de reconocimiento de voz y el sistema eléctrico de la maqueta
- Diseñar y construir la maqueta de una vivienda acoplando el sistema eléctrico necesario para la implementación de la interfaz anteriormente mencionada.

1.5. ALCANCES

- Hacer que este sistema maneje una base de datos de usuarios y así mismo crear una interface gráfica que permita al usuario visualizar el estado de la iluminación de su hogar.
- Implementar esta interfaz en el sistema eléctrico de una vivienda a escala real.
- Comercializar este sistema en el mercado de finca raíz.

1.6. LIMITACIONES

- El sistema de reconocimiento voz es dependiente del usuario, ya que el tono de voz varía según el estado de ánimo físico y de salud.
- Acoplar la totalidad del sistema y realizar las pruebas necesarias para evaluar su funcionamiento en busca de la optimización del mismo.

2. METODOLOGÍA

2.1. ENFOQUE DE LA INVESTIGACIÓN

El enfoque del proyecto se puede describir como un método empírico-analítico porque se hace un análisis de lo que se pretende lograr con este proyecto de acuerdo a las necesidades imperantes en las personas que tienen un vínculo con los diseños de sistemas de sonido y la programación de un software de reconocimiento de voz, con esto, se quiere decir que para desarrollar un hardware que controla información que será de gran ayuda.

2.2. LÍNEA DE INVESTIGACIÓN / SUBLÍNEA DE LA FACULTAD / CAMPO TEMÁTICO DEL PROGRAMA

2.2.1. LÍNEA DE INVESTIGACIÓN DE LA UNIVERSIDAD: TECNOLOGÍAS ACTUALES Y SOCIEDAD

Las tecnologías con las que se trabajan actualmente permiten facilitar la recolección de información o datos, por tal motivo se dispone aún más de herramientas en el campo Ingenieril. Para lograr esto se incorporan áreas de conocimiento como la informática, la electrónica y el procesamiento digital de señales con el fin de efectuar las respectivas aplicaciones y desarrollo en el instrumento tratado, para una futura manipulación por parte de los estudiantes de la universidad de San Buenaventura.

✓ Conocimientos Implicados

- ✓ Aplicaciones tecnológicas de sistemas de sonido
- ✓ Adaptación y/o adopción de tecnologías

✓ Resultados esperados

- ✓ Soluciones viables en el campo tecnológico
- ✓ Obtención de información por medio de esta herramienta

✓ Aplicabilidad

- ✓ Este instrumento es utilizado para solucionar problemas de los usuarios frente al manejo de dispositivos externos.

✓ **Recursos**

- ✓ Recurso humano calificado y capacitado (investigadores, docentes y estudiantes)
- ✓ Se permitirá encontrar un software completamente actualizado Recurso humano calificado y capacitado (investigadores, docentes y estudiantes)
- ✓ Bibliografía y recursos actualizados sobre tecnologías de punta

2.3. SUBLINEAS DE INVESTIGACION DE LA FACULTAD

2.3.1. Procesamiento de señales digitales y/o analógicas

El campo de interés de esta sub-línea de investigación es la teoría y aplicación de transmisión, detección, análisis, registro y/o reproducción de señales en el dominio del tiempo y frecuencia, mediante el uso de dispositivos o técnicas digitales o análogas. El término señal incluye audio, video, voz, imagen, sonar, radar, musical, entre otras pero para este proyecto se trabajara solo con voz.

El procesamiento de señales es una gran necesidad en la actualidad, porque se requiere tener una solución que permita adquirir información a partir de señales, almacenarla y manejarla matemáticamente, integrar datos de diferentes fuentes, predecir comportamientos etc., para ser aplicada en diferentes áreas. Para abordar los proyectos pertenecientes a esta sub-línea, es necesario tener una sólida formación matemática en el proceso de señales particularmente, además de poseer conocimientos en el manejo de programas digitales utilizados para cumplir tareas específicas de este tema.

Los proyectos desarrollados abordarán el diseño de dispositivos que garanticen el adecuado procesamiento de señales, atendiendo a las necesidades que se presenten en procesamiento de señales de algún proceso proveniente de los diversos sectores, ya sea el aeronáutico, electrónico, o acústico, entre otros.

✓ **Conceptos generales**

- ✓ Procesamiento: Someter algún dato o señal a una transformación, elaboración, tratamiento de información.
- ✓ Señal: Fenómeno físico en el cual pueden variar una o más características para representar información.

- ✓ Señal analógica: Señal continua cuya amplitud puede adoptar un intervalo continuo de valores.
- ✓ Señal digital: Señal discreta con amplitud cuantificada. Dicha señal se puede representar mediante una secuencia de números, por ejemplo, binarios.
- ✓ **Conocimientos implicados**
 - ✓ Procesamiento digital de señales.
- ✓ **Resultados esperados**
 - ✓ Diseño de dispositivos para procesamiento de cualquier tipo de señal digital y/o analógicas.
- ✓ **Aplicabilidad**
 - ✓ Análisis de señales, diseño de sistemas de sonido.
- ✓ **Recursos**
 - ✓ Recurso humano calificado y capacitado (investigadores, docentes y estudiantes)
 - ✓ Equipos para captura, almacenamiento y análisis de señales.
 - ✓ Hardware y Software para la manipulación de señales.
 - ✓ Bibliografía actualizada y acceso a redes de información.

2.4. CAMPO DE LA FACULTAD

2.4.1. Diseño de sistema de sonido

El campo de investigación de Diseño de Sistemas de Sonido, requiere de un alto valor formativo en Ingeniería que suministran las ciencias básicas aplicadas a las temáticas específicas del espectro de posibilidades de la Ingeniería de Sonido.

El campo laboral nacional e internacional exige que un Ingeniero de Sonido sea competente, para asumir problemas y generar soluciones al mundo globalizado. En este campo de investigación, el Ingeniero de Sonido se apropia del conocimiento desarrollando habilidades y destrezas, que le permiten asumir cada

una de las aplicaciones requeridas y poseer una mente abierta a nuevas aplicaciones a través del trabajo interdisciplinario.

✓ **Conceptos generales**

- ✓ Este campo busca desarrollar competencias en análisis y diseño de sistemas radiantes y de captación sonora, el procesamiento control y transmisión de señales mediante la medición, simulación y estudio de los parámetros eléctricos.

✓ **Conocimientos implicados**

- ✓ Física mecánica, electromecánica, de oscilaciones y ondas.
- ✓ Instrumentación electrónica, Circuitos eléctricos, Electrónica análoga, digital.
- ✓ Procesamiento digital de señales.
- ✓ Señales y sistemas.

✓ **Resultados esperados**

- ✓ Fortalecimiento del trabajo interdisciplinario
- ✓ Desarrollo, implementación e innovación de productos de sistemas de sonido, que satisfagan el mercado nacional e internacional con miras al mejoramiento de la calidad de vida.

✓ **Aplicabilidad**

- ✓ Análisis de señales, procesamiento digital de señales.
- ✓ Diseño e implementación de sistemas radiantes, diseño de software para sonido.

✓ **Recursos**

- ✓ Personal docente calificado y capacitado.
- ✓ Software actualizado, sistemas de desarrollo, Chips, DSP.
- ✓ Equipos de cómputo disponible, Equipo de laboratorio suficientes.

2.5. RECOLECCIÓN DE INFORMACIÓN

Para la recolección de información necesaria para la elaboración de este proyecto se recurrirá a textos y documentos basados en el reconocimiento de la voz, las técnicas más utilizadas y los métodos implementados por otros software de reconocimiento, también se investigaran documentos basados en la fisionomía del aparato fonador, para así entender su funcionamiento y características que nos permitan evaluar más adelante la caracterización de un locutor y el posterior reconocimiento de la voz.

De igual modo se recurrirá al tutoriales en programación y a las múltiples ayudas del sistema de programación MatLab para seleccionar las posibles funciones que nos permitan la consecución del objetivo principal del proyecto.

2.6. HIPÓTESIS

Es posible crear un interfaz que basado en un software de reconocimiento de voz sea capaz de realizar una función específica lo cual puede llegar a facilitar las labores cotidianas básicas como prender la luz siendo de gran ayuda para el usuario del dispositivo.

2.7. VARIABLES

2.7.1. Variables dependientes

- ✓ El ruido de fondo del sitio donde se realizaran las pruebas del sistema.
- ✓ La efectividad en la aplicación del software e interfaz.

2.7.2. Variables independientes

- ✓ El estado de la voz del interlocutor que simulara al usuario que interactúa con el sistema.
- ✓ Los componentes eléctricos con que se va a construir la interfaz.

3. MARCO DE REFERENCIA

3.1. MARCO TEORICO

En este punto se especifican cada una de las aplicaciones y la información más relevante sobre el desarrollo del proyecto

3.1.1. MATLAB

MATLAB es un programa de cálculo técnico y científico, cuando puede ejecutar sus funciones en código nativo con los tamaños más adecuados para aprovechar sus capacidades de vectorización. En otras aplicaciones resulta bastante más lento que el código equivalente desarrollado en C/C++ o Fortran.³

MATLAB es un programa para realizar cálculos numéricos con vectores y matrices. Como caso particular puede también trabajar con números escalares (tanto reales como complejos), con cadenas de caracteres y con otras estructuras de información más complejas. Una de las capacidades más atractivas es la de realizar una amplia variedad de gráficos en dos y tres dimensiones

3.1.1.1. Simulink

Simulink es una aplicación que permite construir y simular modelos de sistemas físicos y sistemas de control mediante diagramas de bloques. El comportamiento de dichos sistemas se define mediante funciones de transferencia, operaciones matemáticas, elementos de MatLab y señales predefinidas de todo tipo.⁴

Simulink dispone de una serie de utilidades que facilitan la visualización, análisis y guardado de los resultados de simulación. Simulink se emplea profundamente en ingeniería de control.

3.1.1.2. Interfaz gráfica de usuario (GUI)

³ ZÚÑIGA, Ramón. Introducción al uso de MATLAB.
<http://www.geociencias.unam.mx/~bole/eboletin/Matlabintro0408.pdf>, (2009).

⁴ CERVANTES PÉREZ, Jazmín Elizabeth. Simulink practica 4.
<http://bloglabcom.files.wordpress.com/2009/09/p4-simulink-elizabeth-cervantes.pdf>, (2009).

La interfaz gráfica de usuario⁵, es una aplicación proporcionada por MatLab que permite generar al relación con el usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador.

Habitualmente las acciones se realizan mediante manipulación directa, para facilitar la interacción del usuario con la computadora. Surge como evolución de las interfaces de línea de comandos que se utilizaban para operar los primeros sistemas operativos y es pieza fundamental en un entorno gráfico

En el contexto del proceso de interacción persona-ordenador, la interfaz gráfica de usuario es el artefacto tecnológico de un sistema interactivo que posibilita, a través del uso y la representación del lenguaje visual, una interacción amigable con un sistema informático.

3.1.2. ARDUINO

Arduino es una plataforma de electrónica abierta para la creación de prototipos basada en software y hardware flexibles y fáciles de usar. Se creó para artistas, diseñadores, aficionados y cualquiera interesado en crear entornos u objetos interactivos.⁶

Arduino puede tomar información del entorno a través de sus pines de entrada de toda una gama de sensores y puede afectar aquello que le rodea controlando luces, motores y otros actuadores. El microcontrolador en la placa Arduino se programa mediante el lenguaje de programación Arduino (basado en Wiring) y el entorno de desarrollo Arduino (basado en Processing). Los proyectos hechos con Arduino pueden ejecutarse sin necesidad de conectar a un ordenador, si bien tienen la posibilidad de hacerlo y comunicar con diferentes tipos de software (p.ej. Flash, MatLab, Processing, MaxMSP).

⁵ http://es.wikipedia.org/wiki/Interfaz_gr%C3%A1fica_de_usuario, (2004).

⁶ ARDUINO, Home page. <http://arduino.cc/es/>, (2010).

3.1.2.1. Arduino UNO

El Arduino Uno es una placa basada en el microcontrolador ATmega328. Cuenta con 14 pines entradas/salidas digitales (de los cuales 6 pueden ser utilizados como salidas PWM), 6 entradas analógicas, un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, conector ICSP y un botón de reinicio. Contiene todo lo necesario para apoyar el microcontrolador, simplemente se conecta a un computador con un cable USB o a un adaptador AC a DC o a una batería.⁷

3.1.3. D.S.P. (Procesamiento Digita de la Señal)

3.1.3.1. Captura de la señal

La captura de la voz se realiza siempre con un A.C.D o convertidor análogo digital, el cual generalmente siempre es un micrófono, dispositivo con el cual la señal análoga generada por la voz se convierte en impulsos eléctricos para que después de esto se convierta en una onda.⁸

3.1.3.2. Sistemas de reconocimiento biométrico

La biometría es la disciplina que permite identificar a las personas basándose en características fisiológicas o de comportamiento. El reconocimiento a través de características fisiológicas se realiza a partir de las medidas físicas de partes del cuerpo humano (huellas dactilares, mano, iris). Mientras que las basadas en el comportamiento tienen en cuenta cómo realiza cada persona determinadas acciones (voz, firma, escritura en teclado).⁹

Existen 2 clases de reconocimiento biométrico definidos de la siguiente manera:

Autenticación de usuario biométrica y no biométrica: el sistema debe ser capaz de confirmar la identidad del usuario utilizando métodos alternativos al biométrico. En estos métodos alternativos el sistema debe

⁷ ARDUINO UNO. <http://www.juguetrónica.com/tienda/detalles/1516/251/robotica/arduino-y-accesorios/arduino-uno.html>, (2010).

⁸ Música y tecnología, <http://www.iesizpisuabelmonte.es/Adjuntos/profesor/p7804E20MusicaUd01.pdf>, (2010).

⁹ DÍAZ, Alejandro. Boletín electrónico. http://www.ieec.uned.es/Investigacion/ieec_dieec/sb/boletin/boletin%20n2final.pdf, (2004).

tener la posibilidad de autenticar una persona sin necesidad de características biométricas, manteniendo, como mínimo, las mismas características de fiabilidad y robustez que el método biométrico.

Sistema Multi biométrico y multi dispositivo: De cara a la definición de una arquitectura descentralizada, el sistema debe permitir la integración de distintos tipos de dispositivos biométricos (PDA, Móvil, sistemas embebidos, PC) así como distintas características biométricas (huella, voz, escritura, entre otros).

3.1.4. RECONOCIMIENTO DE LA VOZ

En este numeral se mencionan los fundamentos básicos para el reconocimiento de la voz así como los métodos más conocidos para el mismo.

3.1.4.1. Reconocimiento Automático del habla

Bajo la denominación de Reconocimiento Automático del Habla (RAH) se refiere a tareas de diferente complejidad: reconocimiento de palabras aisladas, reconocimiento de palabras conectadas, identificación de palabras clave en discurso continuo, reconocimiento de discurso continuo, etc. Son las características propias del habla: continuidad temporal, variabilidad en la pronunciación, redundancia informativa y multi-interactividad de niveles de conocimiento y de capacidades perceptivas, que afectan negativamente al proceso de reconocimiento, las que nos obligan a imponer ciertas, entre ellas acotar el nivel de ruido o las condiciones ambientales esperables en la señal de entrada, establecer el tamaño del vocabulario (reducido, medio o grande), la dependencia o independencia del locutor, la capacidad de adaptación a nuevos locutores, el tipo de gramática de la aplicación, etc.¹⁰

3.1.4.2. Descripción de un sistema de RAH

Un sistema de RAH se compone de una etapa de Pre proceso (modelado de la señal vocal), una etapa Acústico-Fonética (modelado acústico de unidades sub-léxicas y/o Léxicas) y una etapa Sintáctico-Semántica (modelado del lenguaje). Estas dos últimas pueden combinarse de forma secuencial o integrarse en un único módulo. El sistema parte de una señal de voz, a la cual se aplican técnicas de procesamiento de señal y

¹⁰ BERNAL BERMÚDEZ, Jesús. Reconocimiento de voz y fonética acústica. (2000).

reconocimiento de patrones para generar una cadena de unidades acústicas y a continuación, dependiendo de la aplicación, producir una interpretación semántica que, a su vez, genere una acción u otro tipo de representación de alto nivel.

Habitualmente estas etapas se realizan secuencialmente mediante módulos independientes que por sí mismos definen áreas específicas de investigación. Así, el módulo de adquisición proporciona una señal digital, muestreada a intervalos fijos y con niveles discretos de cuantificación. Esta señal constituye la entrada del módulo de parametrización, donde se realiza un análisis en tiempo y en frecuencia de sucesivos segmentos de señal, generando como salida una secuencia de vectores de parámetros acústicos. Esta transformación debe producir una representación compacta con una pérdida mínima de información. A continuación se procede a la decodificación acústico-fonética (DAF) de la secuencia de vectores, es decir, a la generación de la secuencia óptima de unidades acústicas - habitualmente fonemas- según el conjunto de modelos disponible. En muchos sistemas (por ejemplo, en los HMM discretos), entre el módulo de parametrización y el módulo DAF se inserta un módulo de cuantificación vectorial («Vector Quantization», VQ), proceso en el que cada vector de características es sustituido por la etiqueta del vector centroide más cercano según una cierta métrica definida en el espacio de los parámetros acústicos. En este proceso se pierde parte de la información, pero también se reduce drásticamente el costo computacional del módulo de DAF. Como paso previo, deberá generarse el diccionario de centroides, con el criterio de minimizar el error de cuantificación para un conjunto estadísticamente significativo de vectores (habitualmente se utiliza el algoritmo LBG). El módulo de modelización del lenguaje toma como entrada la secuencia óptima de fonemas, o típicamente las N secuencias «mejores», a partir de las cuales, y según unas ciertas reglas gramaticales más o menos restrictivas, se determinará la secuencia «correcta» y, por tanto, el mensaje que pretendía extraerse de la señal acústica. La figura 1 ilustra las distintas etapas del proceso.¹¹

3.1.4.3. Parametrización o formantes

Los algoritmos de reconocimiento de patrones requieren reducir drásticamente el volumen de datos de la señal de voz. Para ello deberá

¹¹ BERNAL BERMÚDEZ, Jesús. Reconocimiento de voz y fonética acústica. (2000).

eliminarse toda información redundante o inútil, y mantener sólo información relevante, a ser posible mediante un número pequeño de parámetros. Esta discriminación de información y reducción del volumen de datos es lo que trata de efectuarse, en primer término, mediante la parametrización. La primera etapa de procesamiento consiste en la conversión A/D de la señal de voz (filtrado «antialiasing», muestreo y cuantificación). De ella se obtiene una secuencia de números, tratable computacionalmente, que no contiene toda la información acústica de la señal de voz original, pero sí toda la información que interesa a efectos de reconocimiento. Esta secuencia de números se divide en pequeños segmentos consecutivos y solapados, cada uno de los cuales se analiza por separado y produce un vector de parámetros o vector característico. Pueden aplicarse distintos tipos de análisis, ya sea en el tiempo (energía, cruces por cero), en la frecuencia (banco de filtros, transformada de Fourier), o paramétricos (predicción lineal). La representación más utilizada consiste en la transformada inversa de Fourier del espectro, denominada «cepstrum», que modela la articulación del tracto vocal (envolvente espectral) mediante su respuesta a impulsos.¹²

Es posible generar un único vector reuniendo dos o más representaciones distintas. De hecho, la representación paramétrica suele incluir primeras y segundas derivadas que ayudan a describir el carácter dinámico de la señal de voz. Se trata de almacenar el máximo de información en un espacio mínimo. Para conseguir este objetivo, en ocasiones se aplican técnicas de análisis lineal discriminante («Linear Discriminant Analysis», LDA) generando una representación paramétrica reducida, formada por aquellas componentes del vector característico que más información aportan en el proceso de reconocimiento. Otra técnica empleada habitualmente es el denominado filtrado cepstral o «liftering», que modula la envolvente espectral y disminuye su sensibilidad a la posición exacta de las frecuencias de resonancia, permitiendo por tanto una menor dependencia del locutor y de las condiciones de transmisión de la señal (ruidos, etc.).

3.2. MARCO LEGAL Y NORMATIVO

Con El fin de tener una orientación en base a los sistemas de las instalaciones eléctricas, nos basaremos en sus normas y reglamentos, esta reglamentación establece las condiciones mínimas que deberán cumplir las instalaciones

¹² RODRÍGUEZ FUENTES, Luis Javier. Estudio comparativo de varias representaciones paramétricas para el reconocimiento automático del habla. <http://gtts.ehu.es/gtts/NT/fulltext/RodriguezTR94.pdf>. (1994).

eléctricas para preservar la seguridad de las personas y de los bienes, así como asegurar la confiabilidad de su funcionamiento. Rige para las instalaciones en inmuebles destinados a viviendas, comercios, oficinas y para las instalaciones en locales donde se cumplan funciones similares, inclusive las temporarias o provisorias, con tensiones alternas de hasta 1.000 V (valor eficaz) entre fases y frecuencia nominal de 50 Hz (ver **norma IRAM 2001**). Del mismo modo con el mismo objetivo nos basaremos en el código eléctrico colombiano establecido por la norma NTC 2050; igualmente el Reglamento Técnico de Instalaciones Eléctricas – RETIE el cual se aplica a toda instalación eléctrica nueva, ampliación y remodelación de la misma

4. DESARROLLO INGENIERIL

Se realizó una evaluación de los requerimientos necesarios para el desarrollo del sistema de reconocimiento de voz, de acuerdo con los objetivos propuestos.

Los requerimientos primordiales en esta evaluación fueron:

- Un micrófono capaz de capturar señal en el rango de la voz y que la relación señal/ruido sea relativamente baja.
- Una interfaz que pueda comunicar el computador con el mundo real.
- Un computador que permita la capturar audio, procesarlo y ejecutar comandos por puerto USB.
- Una maqueta diseñada para ensamblar y mostrar todos los componentes anteriormente mencionados.

4.1. SOFTWARE

Inicialmente se hizo una investigación del software más conveniente para el desarrollo de reconocimiento de voz, donde se optó por utilizar el programa MatLab 7.1.0.0 (R 2010a) para Windows 7. Ultimate) y realizar la programación del algoritmo que permitirá reconocer la voz para realizar el encendido de los bombillos.

4.2. HARDWARE IMPLEMENTADO

En cuanto al hardware se utilizó el Arduino Uno, este dispositivo suple las necesidades para el enlace con el sistema eléctrico del dispositivo, además ofrece el privilegio de utilizar el microchip AT Mega 328P-PU, para de esta manera conectarse por cable USB al ordenador y operar como esclavo del MatLab por medio de un algoritmo, y una programación pre establecida que se inserta en el micro-controlador con la ayuda de un programa llamado Arduino (V.0022) brindado por el proveedor del hardware.

Imagen 4.1. Arduino



http://www.tuxbrain.com/sites/default/files/ArduinoUNO_Front.jpg

Se utilizo el Arduino Uno ya que es un dispositivo de código abierto (Open Source) diseñado para este tipo de proyectos.

Paso seguido a la salida del cada pin del Arduino se envía esta señal a un opto acoplador cuya función es pasar la señal de un circuito al otro sin la necesidad que haya conexión eléctrica entre ellos; para esto se utilizo el circuito eléctrico (Imagen 4.3)y se instalo en la baquela (Imagen 4.4).

En la figura 4.5 se muestra el circuito final para las 5 salidas del Arduino y posterior conexión al bombillo de 110V.

Imagen 4.2. Mapa del circuito eléctrico

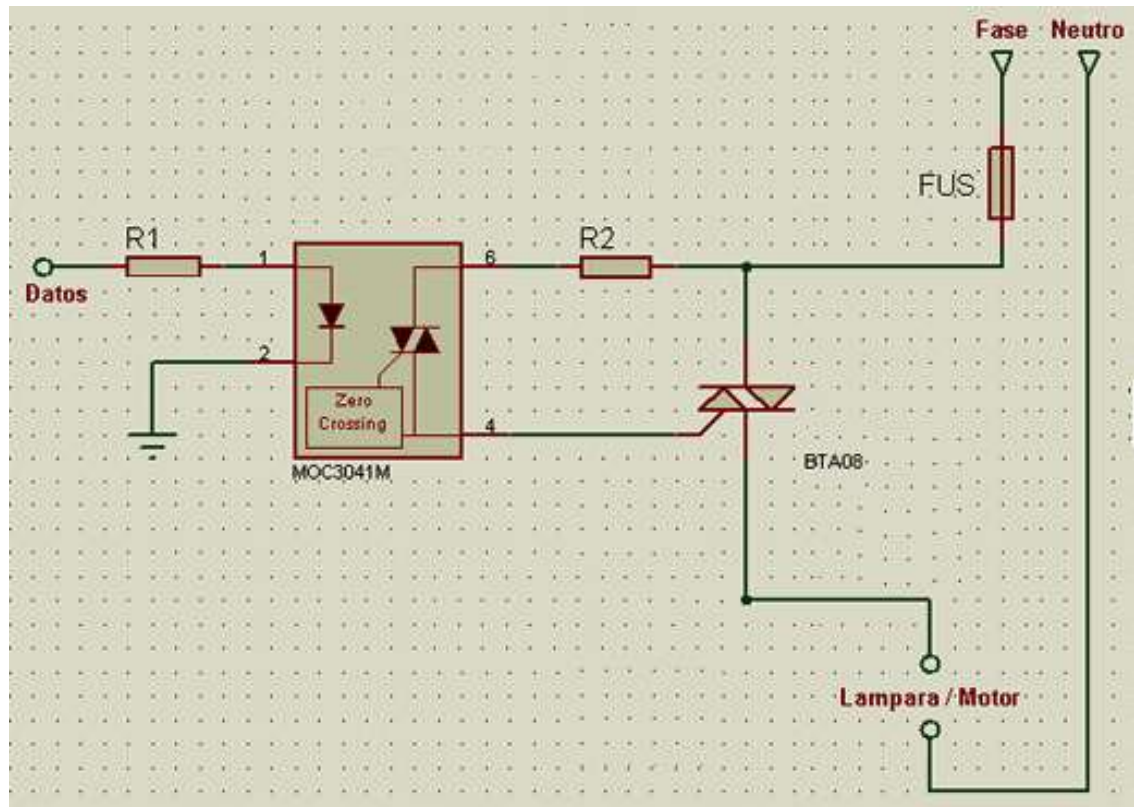


Imagen 4.3. Opto acoplador

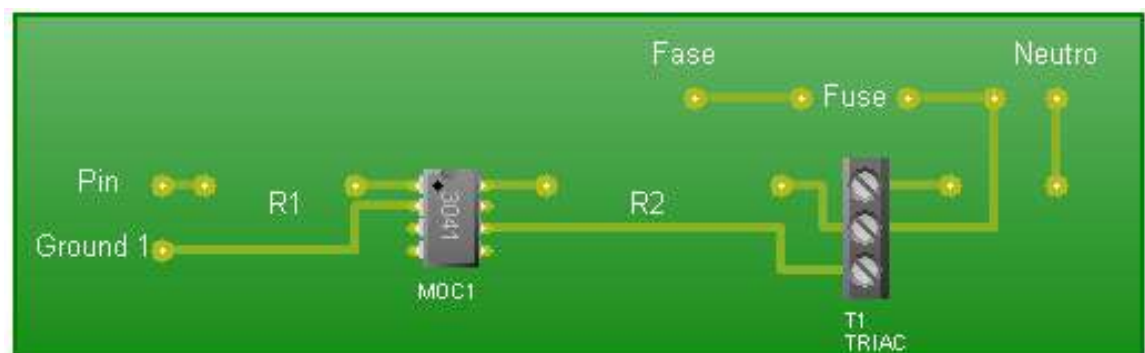
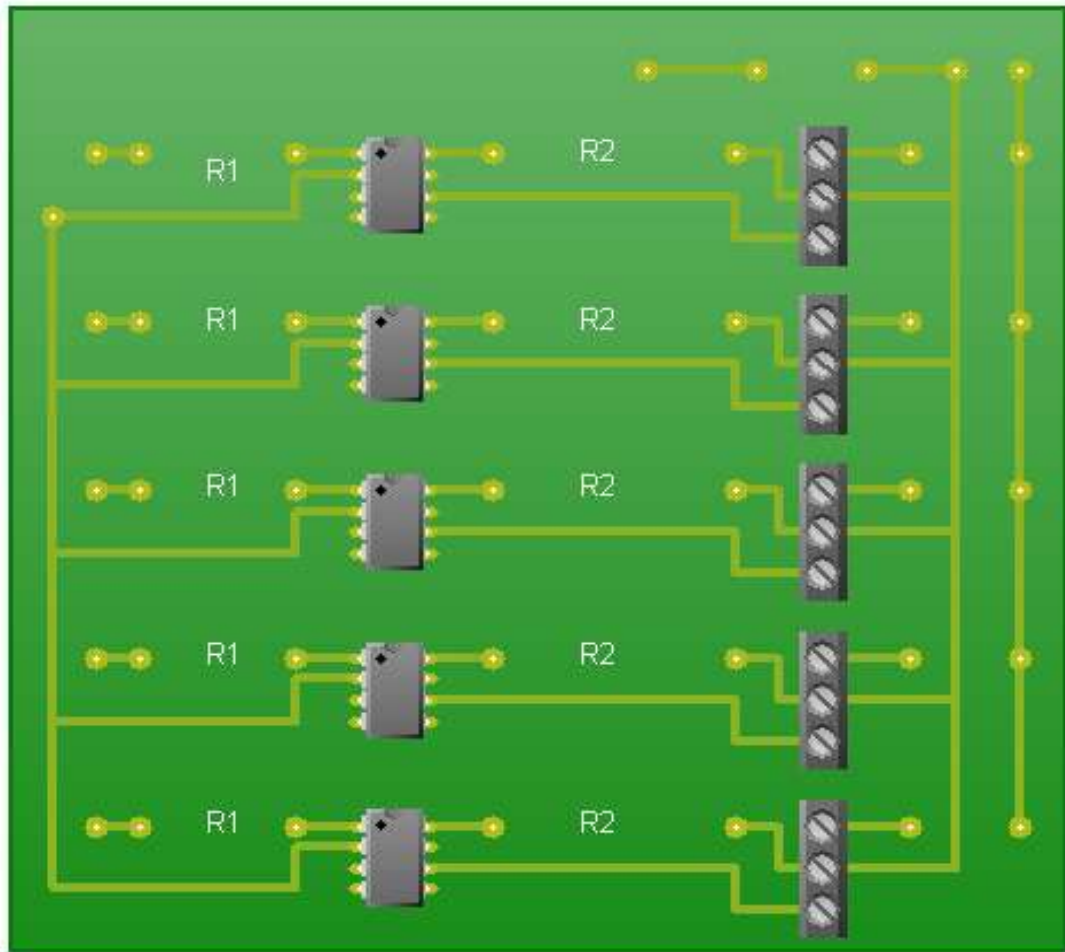


Imagen 4.4. Circuito final



4.3. Selección de comandos

Una de las principales dificultades al momento de reconocer una palabra, radica en la caracterización de un determinado registro vocal ya que estas varían según su sexo, edad y pronunciaciones típicas correspondientes a distintas zonas geográficas del país. Todas estas razones hacen que dos personas que pronuncian la misma palabra, posean patrones frecuenciales y temporales en dicha pronunciación radicalmente distintos y que en apariencia no guardan similitud alguna, es por esta razón que para el diseño de este sistema se acoplo un software que inicialmente fuera compatible para un único usuario sin dejar de lado la posibilidad que en un futuro se pueda configurar y

entrenar para una base de datos más amplia, donde se incrementen el número de locutores.

Después se hace la selección de palabras a trabajar que para el objetivo inicial propuesto se trata de 5 comandos de voz para un usuario, para esto se realizó una lista conformada por las partes de un apartamento compuestas de dos sílabas, que a su vez no tuvieran problemas de solapamientos unas con otras, se realizaron más de 12 análisis para diversos comandos de voz donde se estudia principalmente su contenido armónico a lo largo de cada sílaba que la conformaba (dos veces por sílaba). Después de este proceso se llega a la conclusión que la selección de comandos de voz es uno de los procesos más importantes a tener en cuenta durante el desarrollo del dispositivo, ya que de estos depende el grado de acierto del sistema.

Todas las grabaciones fueron realizadas por un mismo locutor, realizando veinte repeticiones por muestra en donde se obtuvieron los valores correspondientes a cada formante y se almaceno en una base de datos diseñada para el desarrollo de este proyecto obteniendo cuatro formantes para cada palabra, a su vez se promedió el valor de cada formante y también se obtuvieron los valores máximos y mínimos. De esta manera se hizo una selección para los 7 comandos de voz que hicieran parte del análisis del algoritmo(*Tablas 1-7*).

Tabla 1. Sala

SALA (Hz)	f1	f2	f3	f4
Rep1	881	1511	787	1543
Rep2	692	1511	787	1543
Rep3	692	1543	692	1511
Rep4	724	1511	724	1574
Rep5	692	1574	787	1417
Rep6	692	1543	724	1543
Rep7	850	1511	755	1511
Rep8	692	1543	787	1448
Rep9	818	1543	787	1448
Rep10	724	1480	818	1574
Rep11	724	1606	881	1574
Rep12	692	1574	787	1511
Rep13	818	1480	724	1574
Rep14	881	1417	787	1417
Rep15	881	1448	818	1417
Rep16	850	1480	850	1448
Rep17	881	1480	818	1448
Rep18	881	1543	818	1543
Rep19	755	1574	850	1511
Rep20	724	1511	850	1511
Minimo	692	1417	692	1417
Promedio	777	1519	792	1503
Maximo	881	1606	881	1574

Tabla 2. Jardín

JARDIN (Hz)	f1	f2	f3	f4
Rep1	818	1511	346	3055
Rep2	787	1637	346	2803
Rep3	692	1543	314	2488
Rep4	724	1574	314	2866
Rep5	818	1543	346	3023
Rep6	755	1543	346	2456
Rep7	787	1480	314	2866
Rep8	755	1448	314	2488
Rep9	787	1606	346	2960
Rep10	755	1543	314	2866
Rep11	755	1480	346	2551
Rep12	755	1543	314	2897
Rep13	692	1385	346	3086
Rep14	755	1354	346	2929
Rep15	850	1543	346	2897
Rep16	787	1606	346	2803
Rep17	755	1511	346	2960
Rep18	629	1606	346	3023
Rep19	787	1480	346	2897
Rep20	850	1574	346	2992
Minimo	629	1354	314	2456
Promedio	765	1526	336	2845
Maximo	850	1637	346	3086

Tabla 3. Balcón

BALCON (Hz)	f1	f2	f3	f4
Rep1	724	1417	472	1669
Rep2	724	1354	629	1291
Rep3	692	1448	472	1354
Rep4	661	1385	472	1322
Rep5	692	1448	629	1165
Rep6	692	1448	503	1039
Rep7	755	1511	346	1039
Rep8	692	1448	472	1574
Rep9	629	1385	661	1322
Rep10	598	1417	535	1102
Rep11	692	1574	598	1669
Rep12	629	1480	472	1133
Rep13	724	1385	598	1385
Rep14	692	1385	472	1291
Rep15	692	1322	661	1165
Rep16	724	1448	566	1133
Rep17	755	1385	472	1385
Rep18	724	1385	472	1448
Rep19	755	1385	472	1606
Rep20	629	1385	346	1070
Minimo	598	1322	346	1039
Promedio	694	1420	516	1308
Maximo	755	1574	661	1669

Tabla 4. Piso

PISO (Hz)	f1	f2	f3	f4
Rep1	2425	3023	472	1196
Rep2	2519	3149	566	1196
Rep3	2519	3086	440	1070
Rep4	2488	3086	566	1291
Rep5	2551	3086	598	1070
Rep6	2299	2929	472	1291
Rep7	2488	2992	503	1070
Rep8	2488	3055	535	1165
Rep9	3086	4000	472	1196
Rep10	2551	3118	472	976
Rep11	2551	3023	566	1291
Rep12	2362	2929	283	1322
Rep13	3086	3653	409	1196
Rep14	2488	3055	409	1070
Rep15	2992	3464	409	1165
Rep16	3055	3653	409	976
Rep17	2992	3905	440	1165
Rep18	2047	2519	472	1070
Rep19	2488	3086	503	1070
Rep20	2456	2992	503	1165
Minimo	2047	2519	283	976
Promedio	2597	3190	475	1151
Maximo	3086	4000	598	1322

Tabla 5. Mesón

MESON (Hz)	f1	f2	f3	f4
Rep1	598	2960	472	1291
Rep2	535	2015	472	1133
Rep3	629	2708	535	1228
Rep4	629	2047	535	1102
Rep5	566	2299	472	1291
Rep6	566	2204	283	1165
Rep7	472	2362	472	1007
Rep8	472	2204	472	1165
Rep9	598	2236	472	1196
Rep10	472	3023	472	1165
Rep11	692	1984	346	1165
Rep12	472	2393	472	1228
Rep13	598	1921	472	1165
Rep14	472	2330	283	1196
Rep15	692	2204	314	1291
Rep16	566	2456	472	1165
Rep17	566	2141	314	1165
Rep18	598	2110	346	1196
Rep19	566	2047	472	1165
Rep20	472	2960	472	1165
Minimo	472	1921	283	1007
Promedio	562	2330	431	1182
Maximo	692	3023	535	1291

Tabla 6. Tina

TINA (Hz)	f1	f2	f3	f4
Rep1	2897	3370	283	787
Rep2	2551	3023	283	818
Rep3	2551	3055	283	818
Rep4	3055	3590	283	787
Rep5	2582	3086	283	818
Rep6	3023	3496	283	787
Rep7	2897	3370	283	787
Rep8	2803	3590	283	818
Rep9	2582	3118	283	755
Rep10	2519	3023	283	787
Rep11	2488	2992	283	818
Rep12	2519	2992	283	787
Rep13	3275	3937	283	818
Rep14	2834	3496	314	881
Rep15	2803	3370	283	787
Rep16	3023	3496	283	755
Rep17	2551	3055	283	787
Rep18	2740	3244	283	818
Rep19	3023	3622	283	818
Rep20	2614	3307	283	755
Minimo	2488	2992	283	755
Promedio	2767	3312	285	799
Maximo	3275	3937	314	881

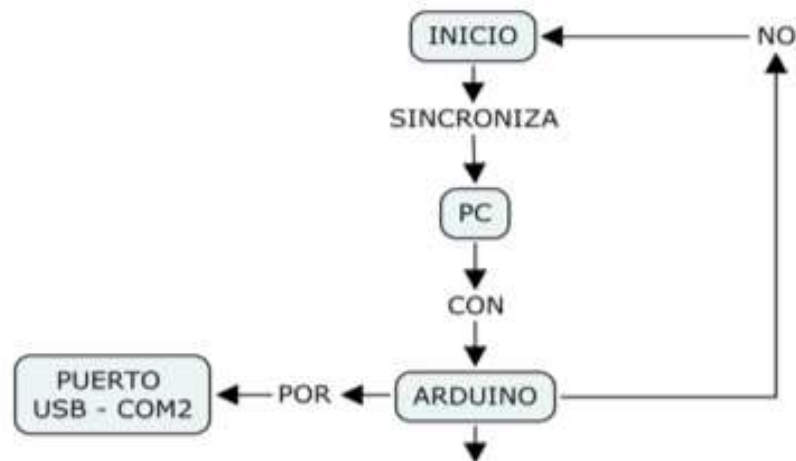
Tabla 7. Uno

UNO (Hz)	f1	f2	f3	f4
Rep1	818	1291	283	1102
Rep2	818	1543	283	1039
Rep3	724	1385	283	1102
Rep4	881	1417	283	1322
Rep5	818	1574	283	1165
Rep6	818	1543	283	1133
Rep7	724	1417	283	1070
Rep8	818	1417	283	1133
Rep9	755	1385	283	1133
Rep10	692	1385	283	1102
Rep11	787	1511	283	1165
Rep12	818	1543	283	1165
Rep13	787	1417	283	1102
Rep14	724	1228	283	1133
Rep15	818	1511	283	1196
Rep16	818	1448	283	1133
Rep17	755	1448	283	1102
Rep18	818	1417	283	1228
Rep19	850	1543	283	1102
Rep20	818	1511	283	1165
Minimo	692	1228	283	1039
Promedio	793	1447	283	1140
Maximo	881	1574	283	1322

4.4. ELABORACIÓN DE ALGORITMO

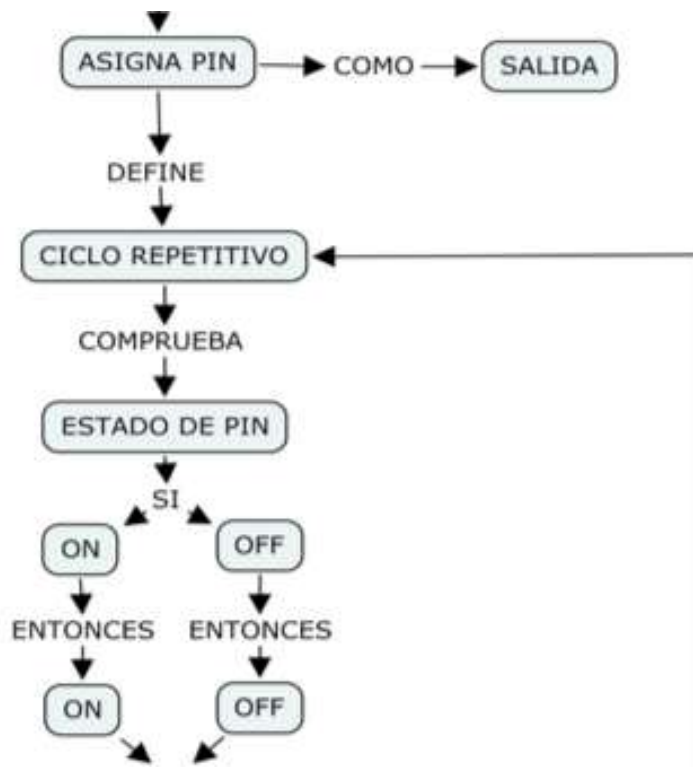
En la primera parte del algoritmo (Anexo A) se sincroniza el hardware con MatLab donde además se definen las variables con las que se va a trabajar a lo largo de todo el algoritmo, por otro lado es necesario configurar el mismo para que su transferencia de datos sea por un puerto serial específico, que para este caso se decidió por (COM 2),

Imagen 4.4.1. Diagrama de flujo 1



Seguidamente se configuran todos los pines como las salidas de nuestro hardware, para luego hacer una comprobación del estado actual de cada una de ellas (encendido o apagado de los LED), para que al momento que se realiza el ciclo, el algoritmo mantenga los valores con los que se terminó en el ciclo anterior y únicamente modifique el valor de la salida al verificar un comando de voz.

Imagen 4.4.2. Diagrama de flujo 2



Una vez terminado este paso se procede a hacer la captura de la muestra; para esto se destinó un LED que se encienda al momento de la captura y vuelve a apagarse al finalizar. Esta captura se realiza a 8000 muestras por segundo, se almacena en el computador y se carga nuevamente en una variable denominada comando.

Imagen 4.4.3. Diagrama de flujo 3



Luego del procesamiento de la señal restamos un pico innecesario al inicio de la muestra, luego procedemos a hacer el inventariado y separación de las silabas; para esto se calcula la energía en corto tiempo y se busca el valor más bajo ente las dos silabas para luego poder inventanar las dos silabas por separado; para después normalizar la señal. Una vez que se realizó el inventariado de la señal,

se procede a dividirla por sílabas, buscando el punto mínimo de la señal y la posición del vector para después asignar a cada parte el nombre de *silaba uno* y de *silaba dos* correspondientemente.

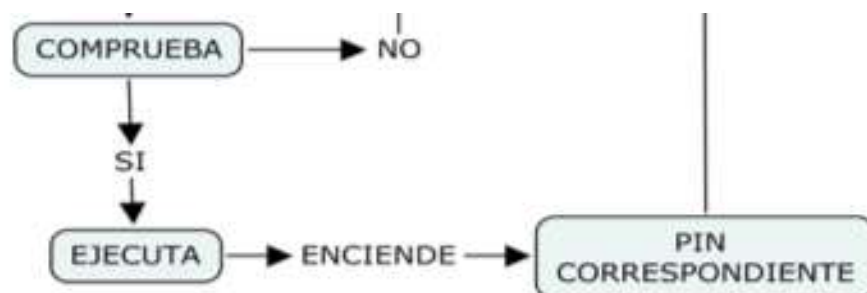
Luego de obtener la *silaba uno* se procede a realizar la transformada rápida de Fourier, que para una mejor comprensión a lo largo de este desarrollo lo denominaremos por medio de las siglas FFT, y de esta manera obtener la primera formante “f1”; ya encontrado este valor a partir de “f1” se busca la segunda formante realizando nuevamente la FFT la cual denominaremos “f2”, para la sílaba dos se repite el proceso y se designan los valores “f3” para la primera formante y “f4” para la segunda formante.

Imagen 4.4.4. Diagrama de flujo 4



Posteriormente terminado el entrenamiento del sistema y habiendo obtenidos los rangos aproximados para las variables anteriormente mencionadas (f1, f2, f3y f4), se construye la etapa de confirmación de los comandos de voz, y en caso de existir alguna coincidencia en todos los parámetros se pide al software el valor actual de la salida del led y se envía la orden de encender o apagar el bombillo dependiendo del estado actual y ejecutar la orden inversa.

Imagen 4.5.5. Diagrama de flujo 5



Este proceso tiene un ciclo repetitivo que dura aproximadamente 15 segundos, esto quiere decir que el software da la oportunidad de ordenar nuevamente en este lapso de tiempo.

Para la ejecución del programa es indispensable tener conectado el hardware en el puerto pre establecido y dar click en Inicio.

Imagen 4.4. Diagrama de flujo final

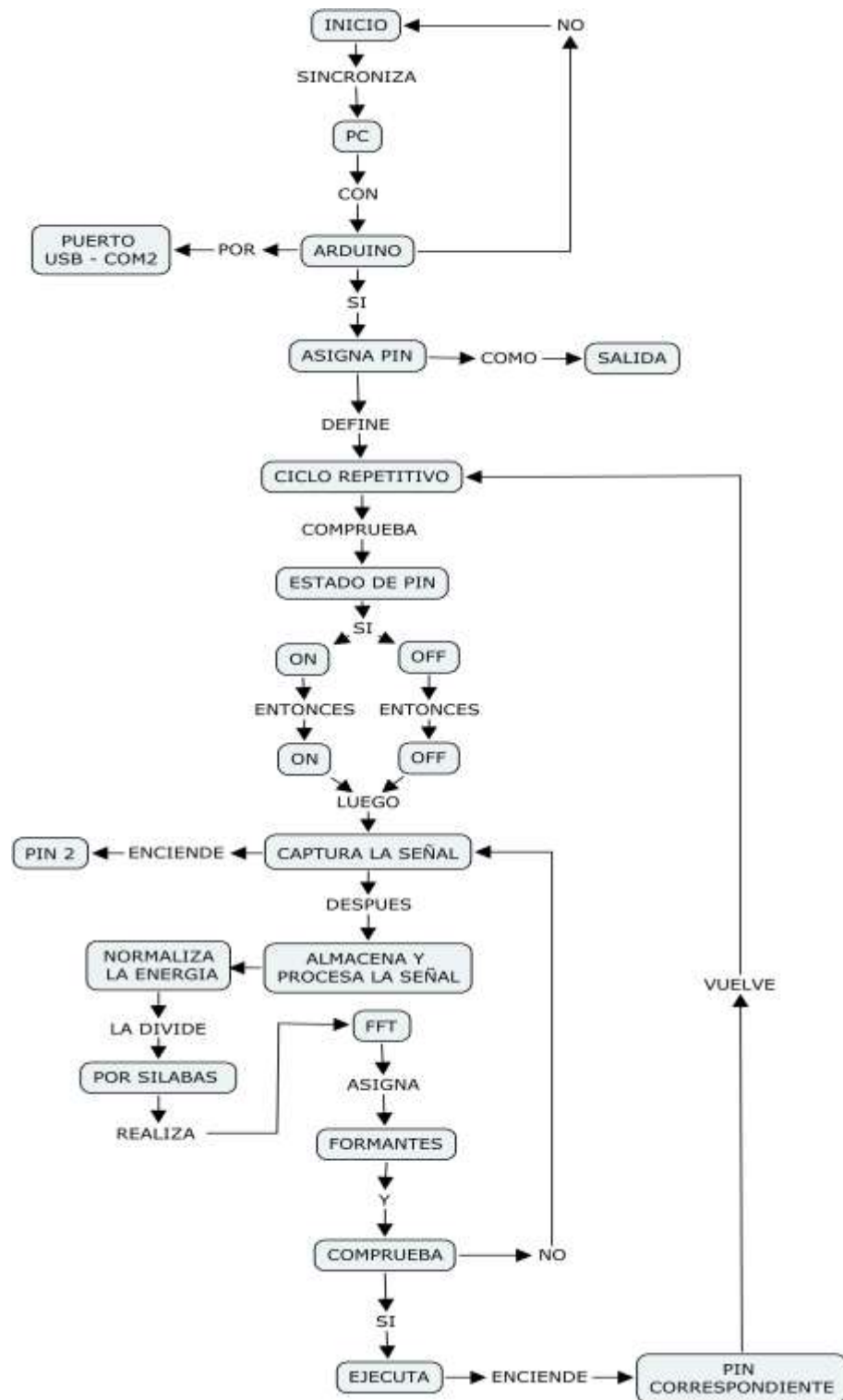
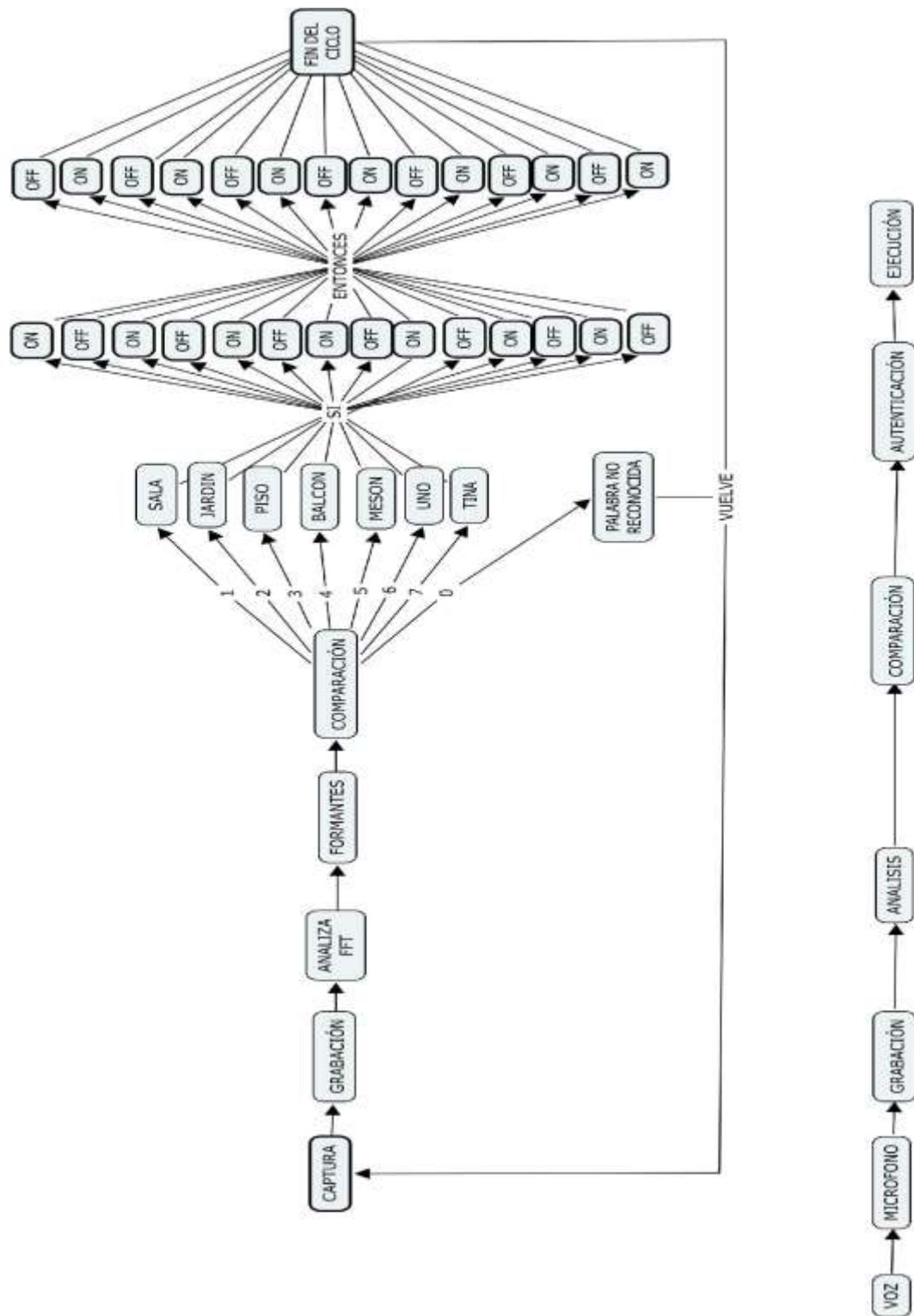


Imagen 4.5. Diagrama de bloques



4.5. INTERFAZ GRAFICA

La interfaz gráfica de usuario del software implementado para el reconocimiento de voz fue realizada en GUI, aplicativo del mismo MatLab que permite hacer este tipo de entornos de una manera eficaz y rápida. Esta interface comprende tres botones como se muestra en la imagen 4.7.

Imagen 4.6. Interface gráfica



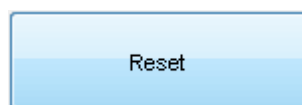
El botón de inicio da comienzo al algoritmo de reconocimiento y al ciclo repetitivo de grabación, extracción de formantes y comprobación de datos, en el punto final del ciclo se comprueba la relación de las cuatro formantes de la muestra grabada, procede a ejecutar la orden designada y encender el LED que esta correlacionado con la palabra, en caso de no reconocer comprobar y validar la muestra con ninguno de los datos suministrados simplemente se repite el ciclo.

Imagen 4.6.1. Interface gráfica – Botón 1



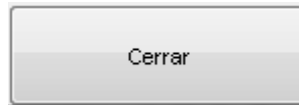
Posteriormente encontramos el botón Reset que como su nombre lo indica es el encargado de reiniciar el Arduino, la interface y el algoritmo en caso de presentarse algún error en la ejecución del mismo; fue necesaria la implementación de este botón debido a que en algunas ocasiones el Arduino quedaba ocupando el puerto serial destinado para la transferencia de datos y simplemente al tratar de iniciar de nuevo el programa el MatLab indicaba que el puerto ya estaba siendo ocupado por otro programa.

Imagen 4.6.2. Interface gráfica – Botón 2



Por último encontramos el botón cerrar, que es el encargado de dar fin al ciclo repetitivo ya mencionado y cerrar la interfaz gráfica dejando así mismo el puerto serial libre para ser utilizado por cualquier otro dispositivo que sea conectado al ordenador.

Imagen 4.6.3. Interface gráfica – Botón 3



Esto se ve reflejado en el algoritmo con una serie de comandos que el mismo GUI de MatLab inserta (Anexo B).

4.6. MAQUETA

El diseño del espacio en que realizara la simulación y prueba del sistema fue realizado en base a un apartamento tipo loft a escala 1:25. Para esto se utilizó el software AutoCad, programa especializado para este tipo de trabajos.

El recinto cuenta con un espacio amplio destinado para la sala/comedor, una cocina, un balcón ubicado en una esquina cubriendo gran parte de la sala/comedor, el cuarto de equipos está dispuesto hacia el centro del recinto ya que este proporcionara la alimentación de energía hacia los distintos espacios de la casa, una alcoba y un baño.

Posterior a tener el diseño final en planta del recinto se procedió a extrudar los muros y a ubicar las ventanas y puertas de vidrio corredizas que tendría el modelo a escala real (Imagen 4.8 – 4.9).

Después de realizar el diseño en 3D se diseñó una cubierta a se aplicaron texturas para el render final (Imagen 4.10 - 4.11). y se sacaron los moldes para la construcción.

Imagen 4.7. Planta

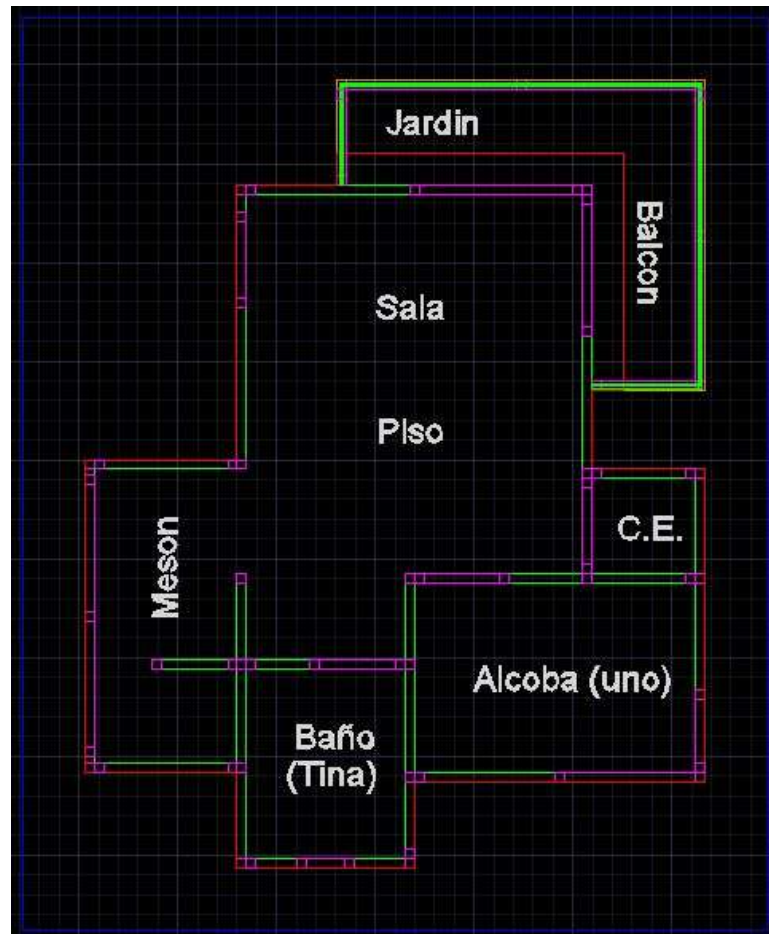


Imagen 4.8. Diseño 3D

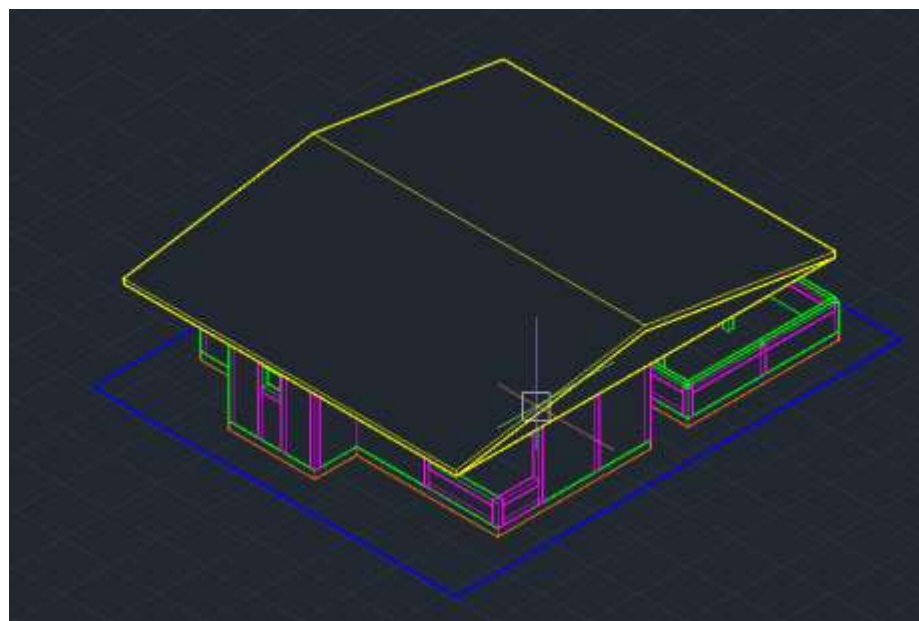


Imagen 4.9. Render vista 1

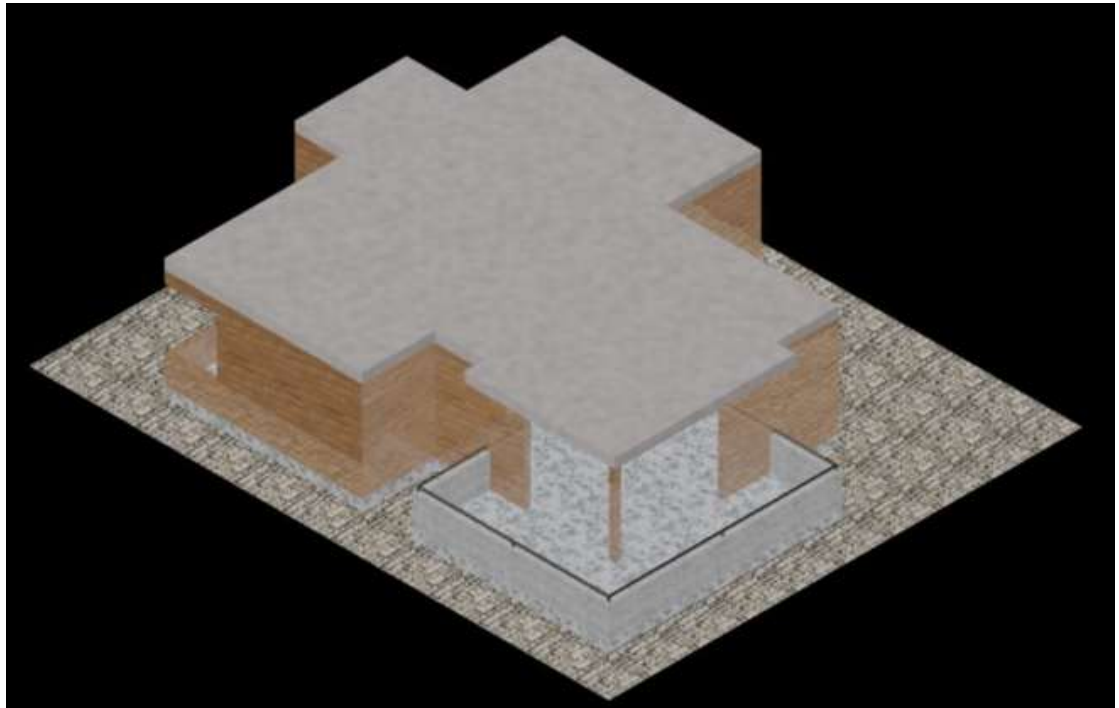
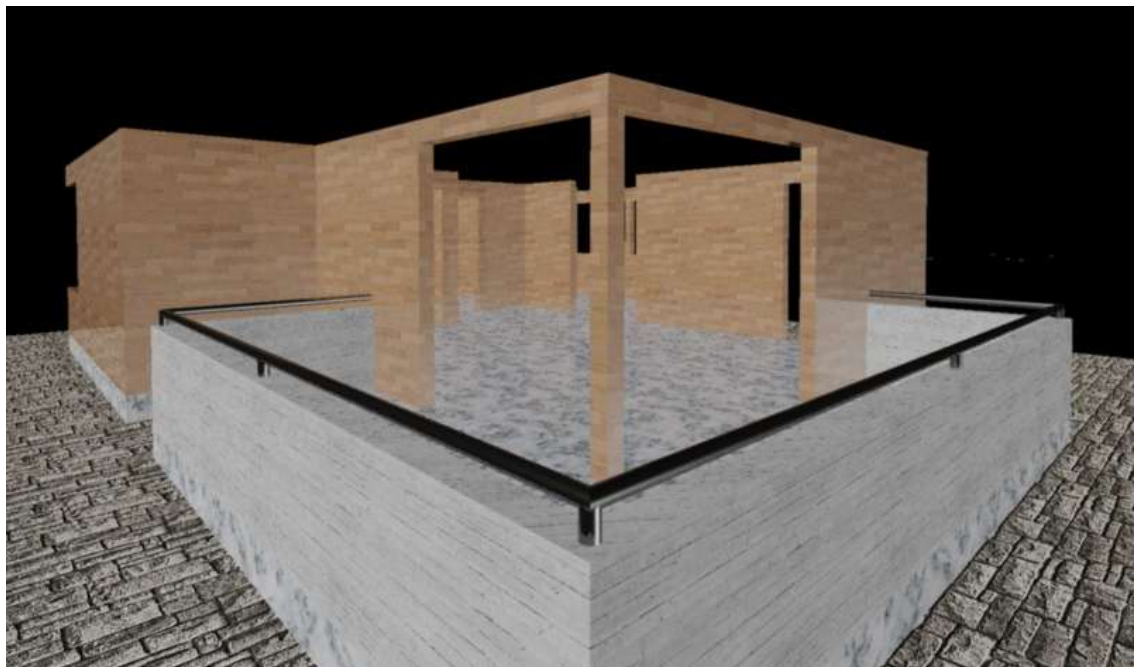


Imagen 4.10. Render vista 2



Basados en los diseños, planos de planta y cortes del modelo digital se procedió a su construcción en base a materiales como cartón paja blanco, balsa de 5 mm, acetato, madera, icopor entre otros y finalizado esto se inserto el cableado y los LED. Luego de la construcción de la estructura se procedió a cablear el cielo raso y a ubicar las instalaciones eléctricas y el Arduino (Imagen 4.12 – 4.13). Una vez instalado el sistema eléctrico, el sistema de reconocimiento y el cableado que conforman el diseño (Imagen 4.14 - 4.16) se ubica la cubierta y se procede a probar el sistema completo.

Imagen 4.11. Arduino acoplado en el techo



Imagen 4.12. Maqueta construida cielo raso

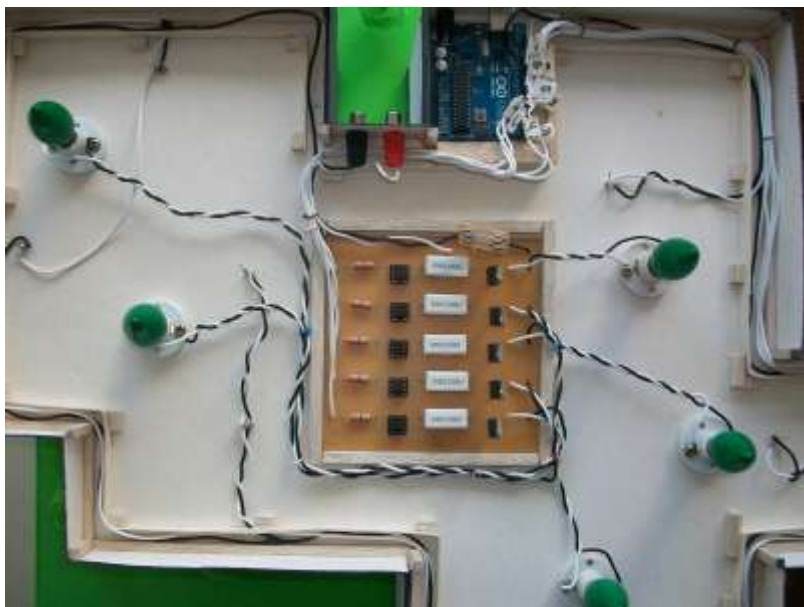


Imagen 4.13. Maqueta construida fachada



Imagen 4.14. Maqueta construida planta



Imagen 4.15. Sistema funcionando

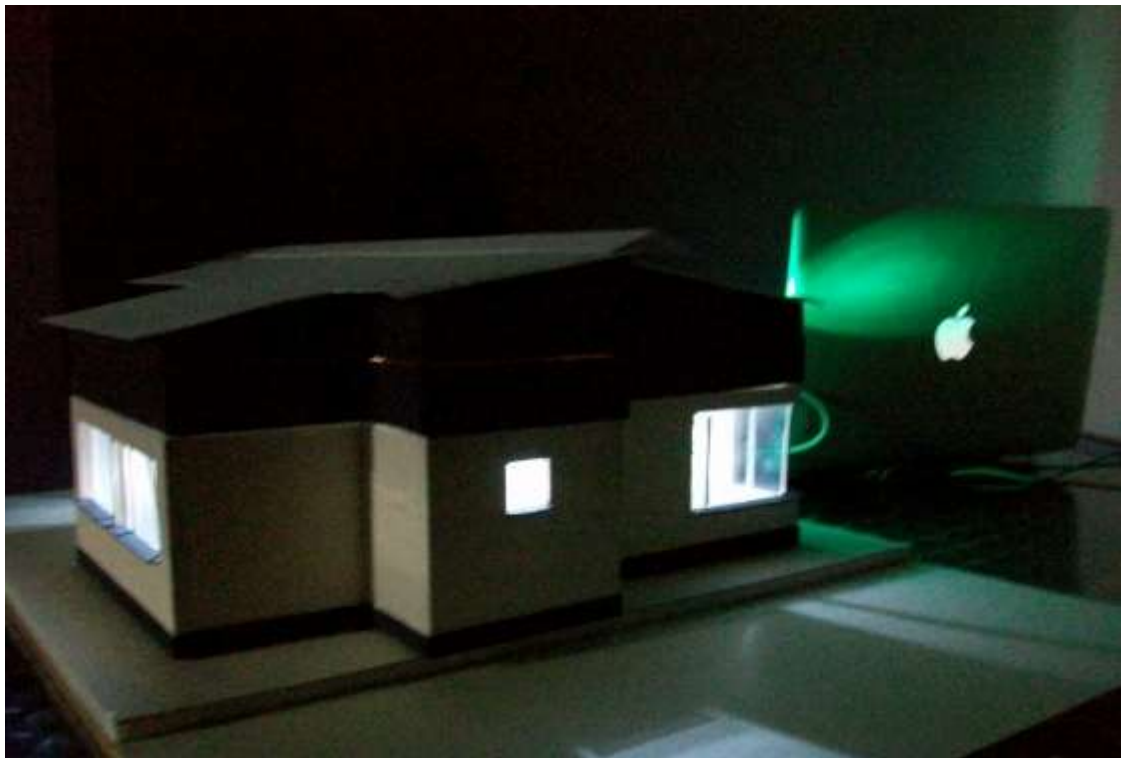


Imagen 4.16. Producto final



5. ANÁLISIS DE RESULTADOS

5.1. PRUEBA DE ENSAYO Y ERROR

Al momento de terminar el desarrollo del software, el hardware y la programación necesaria para el funcionamiento adecuado del sistema se realizaron diferentes pruebas con el fin de conocer el porcentaje de error en la captura, reconocimiento y ejecución del producto final.

Para esto se citó al locutor en 4 oportunidades, ambientes diferentes con mayor y menor ruido de fondo y se le pidió repetir 25 veces cada comando predefinido anteriormente (Sala, Tina, Balcon, Meson, Piso, Jardin y Uno).

Día 1:

En el primer día de prueba se cito al locutor de 9:00am a 11:00 am, este día se probó el sistema con un bajo ruido de fondo si ningún tipo de electrodoméstico encendido y con las ventanas cerradas para evitar al máximo cualquier fluctuación de ruido dando como resultado los siguientes datos por palabra. La prueba se hizo tomando 25 repeticiones de cada comando (*Tabla 8*).

Tabla 8. Día 1

Sala	Tina	Balcon	Meson	Piso	Jardin	Uno
Dia 1						
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	0	1	1	0	1
1	1	1	0	1	1	0
0	1	0	1	1	0	1
1	1	1	0	1	1	0
0	0	1	1	0	1	1
1	1	0	1	1	0	1
0	0	1	1	0	1	1
1	1	0	1	1	0	1
0	1	1	1	1	1	1
1	1	0	1	1	0	1
1	1	1	1	1	1	1
1	1	1	0	1	1	0
1	1	1	1	1	1	1
0	1	1	0	1	1	0
1	0	0	1	0	0	1
0	1	1	0	1	1	0
1	0	0	1	0	0	1
0	1	1	1	1	1	0
1	0	1	1	0	1	1
0	1	0	0	1	0	0
1	1	1	1	1	1	1
1	0	0	0	0	0	0
1	1	1	1	1	1	1
17	19	16	18	19	16	17

Teniendo en cuenta que se trataba de la primera prueba que se realizaba el locutor presentaba algo de inseguridad al pronunciar los comandos y de alguna manera era muy variable el contenido en frecuencias de las formantes por lo cual se procedió a analizar la pronunciación y los acentos con los cuales fuera más natural y homogénea la toma. Una vez practicados y repasados los comandos seleccionados se prosiguió con la evaluación de los datos obtenidos.

En el momento de evaluación de los datos se puede observar que de las 25 repeticiones que se realizaron para determinada palabra el menor número que fue reconocido de comandos fue de 16 repeticiones lo cual representa el 64% de porcentaje de reconocimiento con un máximo de 76% para el primer día de pruebas.

Día 2:

Para las siguientes pruebas se quiso cambiar el ambiente sonoro por lo que el locutor fue citado en un domicilio diferente a donde se realizaron las pruebas del primer día, para este caso se trataba de una casa ubicada cerca de una calle que posee un nivel de tráfico medio, la prueba fue realizada de 4:00pm a 6:00pm en un día entre semana donde el tránsito se comportaba con mayor flujo vehicular en una hora pico para la ciudad de Bogotá lo cual generaba un alto nivel de ruido de fondo, para este caso se ubicó el sistema en un segundo piso donde se abrió la ventana buscando la variación con respecto al ruido de fondo que se presentó para el primer día (*Tabla 9*).

Como resultado se logró observar que en este caso los valores del porcentaje de reconocimiento total máximos y mínimos conservaron el valor de la primera prueba con un 76% y 64% respectivamente, pero a diferencia del primer día se incrementaron el número de palabras con un porcentaje de reconocimiento de 76% para este caso 4 de los 7 comandos presentaron este porcentaje, lo cual hace que se incremente el porcentaje total de evaluación del sistema.

Día 3:

Para el tercer día se realizaron las pruebas en un domicilio que está ubicado en zona tranquila en horas matutinas por lo que el locutor fue citado de 8:00am a 10:00am, además se buscó incrementar el nivel de ruido de fondo al interior del hogar encendiendo electrodomésticos, como lo fueron un televisor y un equipo de sonido a un nivel moderado (*Tabla 10*).

Tabla 9. Día 2

Sala	Tina	Balcon	Meson	Piso	Jardin	Uno
Dia 2						
1	1	1	0	1	1	1
1	1	1	1	1	1	1
0	1	1	1	1	1	1
1	0	1	1	0	1	1
0	1	1	1	1	1	1
1	0	1	1	0	1	1
1	1	0	1	1	0	0
0	1	1	1	1	1	1
1	1	0	0	1	0	0
0	1	1	1	1	1	1
1	1	1	0	1	1	1
0	1	1	1	1	1	1
1	1	1	0	1	1	1
1	0	1	1	0	1	1
1	1	1	1	1	1	1
1	0	1	1	0	1	1
0	1	0	0	1	0	0
1	0	1	1	0	1	1
0	1	0	1	1	0	0
1	1	1	1	1	1	1
1	1	0	1	1	0	0
0	0	1	0	0	1	1
1	1	1	1	1	1	1
0	0	0	0	1	0	0
1	1	1	0	1	1	1
16	18	19	17	19	19	19

Tabla 10. Día 3

Sala	Tina	Balcon	Meson	Piso	Jardin	Uno
Dia 3						
1	1	0	0	0	0	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	0	1	1	1	1	1
1	1	1	1	1	1	1
0	0	0	0	0	0	0
1	1	1	1	1	1	1
1	1	0	0	0	0	0
1	1	1	1	1	1	1
1	1	0	0	0	0	0
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
0	0	0	0	1	1	1
1	1	1	1	1	1	1
0	0	1	1	1	1	1
1	1	1	1	1	1	1
0	0	1	1	1	1	0
1	1	0	0	1	1	1
1	1	1	1	1	1	1
0	0	0	0	1	1	1
1	1	0	0	0	1	1
19	19	17	17	20	21	20

Para este caso lo niveles de reconocimiento se incrementaron considerablemente con respecto a las pruebas anteriores, en este día se obtuvo un valor máximo de 84% y un valor mínimo de 68% lo cual aumenta el porcentaje de reconocimiento total considerablemente con respecto a las pruebas anteriores, esta curva ascendente se debe probablemente a que el locutor esta mas familiarizado con los

comandos y debido a la práctica de los mismos estos son pronunciados con mayor confianza.

Día 4:

Para el cuarto día o prueba final se busco realizar la toma de valores en un ambiente lo mas similar a una casa en la que habitan 4 miembros familiares, para esto se hicieron las tomas en un horario donde se encontraban cuatro personas realizando sus actividades normalmente, estas tomas fueron hechas en un horario de 6:00pm a 8:00pm (Tabla 11).

Tabla 11. Día 4

Sala	Tina	Balcon	Meson	Piso	Jardin	Uno
Dia 4						
0	1	0	1	1	1	1
1	1	1	1	1	1	1
1	1	1	0	1	1	1
1	1	1	1	1	1	1
1	1	1	0	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	0	0
1	0	1	0	0	1	1
0	1	0	1	1	0	0
1	0	1	0	0	1	1
0	1	0	1	1	1	1
1	0	1	0	0	1	1
0	1	0	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	0	1	1	0	1	1
1	1	1	0	1	0	0
1	1	1	1	1	1	1
1	1	1	0	1	0	0
1	1	1	1	1	1	1
0	1	0	1	1	0	0
1	1	1	0	1	1	1
1	1	1	1	0	1	1
1	1	1	0	1	0	0
1	1	1	1	1	1	1
20	21	16	16	20	19	19

Para este caso los valores presentaron homogeneidad con un valor máximo de reconocimiento de 84% y un valor mínimo de 64%, a lo largo de todo el proceso de evaluación del sistema se puede observar que el porcentaje total de reconocimiento de voz presento un incremento proporcional con los días en los que se realizaron las pruebas ya que la etapa de entrenamiento es fundamental para el acople entre el sistema y el locutor, por otro lado observamos que el ruido de fondo no afecta considerablemente el funcionamiento del sistema ya que no se presentaron variaciones drásticas entre los valores incluso modificando el ruido de fondo para cada prueba.

Para esto se utilizo la *Tabla 8* en la cual se registra el día de prueba y al mismo tiempo cual fue la cantidad de aciertos por día, marcados con 1 y los desaciertos con un 0, para mostrar los resultados de cada una de estas pruebas y sus resultados. Luego de esto se realiza la sumatoria de cada día y se saca el promedio del total de 100 muestras, lo cual nos permite observar con mayor facilidad cual es el porcentaje de efectividad del sistema.

Como se puede observar (*Tabla 8*) el porcentaje de error está entre el 22% y el 28% lo cual nos indica que el programa está respondiendo bien con respecto al objetivo.

Tabla 12. Prueba y error

Sala				Tina				Balcon				Meson				Piso				Jardin				Uno			
dia 1	dia 2	dia 3	dia 4	dia 1	dia 2	dia 3	dia 4	dia 1	dia 2	dia 3	dia 4	dia 1	dia 2	dia 3	dia 4	dia 1	dia 2	dia 3	dia 4	dia 1	dia 2	dia 3	dia 4	dia 1	dia 2	dia 3	dia 4
1	1	1	0	1	1	0	1	1	1	0	1	1	0	1	1	1	1	0	1	1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1
1	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1
0	0	1	1	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1
1	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1
0	1	0	1	0	1	1	1	1	0	1	0	1	1	1	1	0	1	1	1	1	0	1	0	1	0	1	0
1	0	1	1	1	1	1	0	0	1	1	1	1	1	0	0	1	1	1	0	0	1	1	1	1	1	1	1
0	1	0	0	0	1	0	1	1	0	0	0	1	0	1	1	0	1	0	1	1	0	0	0	1	0	0	0
1	0	1	1	1	1	1	0	0	1	1	1	1	1	0	0	1	1	1	0	0	1	1	1	1	1	1	1
0	1	1	0	1	1	0	1	1	1	0	1	1	0	1	1	1	1	0	1	1	1	0	1	1	1	0	1
1	0	1	1	1	1	1	0	0	1	1	1	1	1	0	0	1	1	1	0	0	1	1	1	1	1	1	1
1	1	1	0	1	1	0	1	1	1	0	1	1	0	1	1	1	1	0	1	1	1	0	1	1	1	0	1
1	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	1	1	1	1	0	1	0	1	1	1	1	0	1	0	1	1	0	1	0	1	1	1	1	0	1	1	1
1	0	0	1	0	1	0	1	0	0	1	0	1	0	1	0	0	1	1	1	0	0	1	0	1	0	1	0
0	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1
1	0	0	1	0	1	1	1	0	0	1	0	1	1	1	0	0	1	1	1	0	0	1	0	1	0	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
1	1	0	0	0	1	1	1	1	0	0	0	1	1	1	1	0	1	1	1	1	0	1	0	1	0	0	0
0	0	1	1	1	0	0	1	0	1	1	1	0	0	1	0	1	0	1	1	0	1	1	1	0	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	0	0	1	0	0	0	1	0	0	1	0	0	0	1	0	0	1	1	1	0	0	1	0	0	0	1	0
1	1	1	1	1	1	0	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1
17	16	19	20	19	18	17	21	16	19	20	19	18	17	21	16	19	19	20	20	16	19	21	19	17	19	20	19
Prom	0,72			Prom	0,75			Prom	0,74			Prom	0,72			Prom	0,78			Prom	0,75			Prom	0,75		

Como se puede observar en la tabla 8. solamente en uno de los casos (Piso.) se obtuvieron valores muy similares ya que en los demás casos la efectividad del sistema varía entre tres y cinco aciertos. Estas variaciones probablemente se deben a la variación del estado de ánimo del locutor, las condiciones de ruido de fondo, hora y lugar donde se realizaron las pruebas.

Tabla 13. Porcentaje total

Sala	72%
Tina	75%
Balcon	74%
Meson	72%
Piso	78%
Jardin	75%
Uno	75%
Total	74%

En la *Tabla 9.* se muestra el porcentaje de efectividad para cada palabra, después de tomar las muestras y promediar el resultado para obtener el valor total de efectividad de un determinado comando que opera el sistema de reconocimiento de voz, de esta manera se establece que el rango mínimo de efectividad es del 72 %, con un máximo de 78% lo cual indica q el sistema posee un porcentaje de error relativamente bajo.

6. CONCLUSIONES Y RECOMENDACIONES

6.1. CONCLUSIONES

- ✓ El reconocimiento de voz para este algoritmo es limitado, debido a que solamente se permite la opción de dos sílabas, además existen formantes de la voz humana que son muy similares en palabras con la misma composición vocal (ejemplo: casa, sala, cama).
- ✓ Es indispensable tener una etapa de entrenamiento al momento de programar el algoritmo ya que el software encargado de reconocimiento debe asimilar la totalidad de las formantes y demás propiedades de la voz del locutor; así mismo el usuario debe utilizar un lenguaje verbal fluido sin léxicos desconocidos, una buena pronunciación y entonación.
- ✓ Es indispensable garantizar el uso de un micrófono con respuesta en frecuencia que este en el rango de la voz humana y además tenga una relación señal ruido excelente.
- ✓ Es importante realizar la comparación de rangos entre cada comando de voz ya que esto permite evaluar qué posibilidades existen para validar dos o más comandos al tiempo o de no ser así puede ayudar a ampliar el rango de posibilidades y de esta manera aumentar porcentaje de efectividad del mismo.
- ✓ El sistema por reconocimiento de voz basado en comparación de formantes utilizado en este algoritmo es muy práctico en palabras de dos silabas aunque la separación entre las mismas debe ser muy marcada para que facilite al programa diferenciar donde inicia la primera silaba, donde finaliza la segunda silaba y la división entre ellas.
- ✓ La implementación de este dispositivo en hogares a nivel comercial es totalmente viable ya que en base a la investigación previa a este proyecto se encontró que ninguna empresa de domótica en el país comercializa este tipo de dispositivos.
- ✓ Es posible reconocer voz en base a un software, y que este mismo se encargue de ordenar por puerto USB la ejecución de diferentes tareas en el mundo real.
- ✓ La efectividad del diferente software de reconocimiento de voz son directamente proporcionales a aspectos inherentes del locutor como la

fuerza de pronunciación, el estado anímico, la correcta pronunciación o enfermedades bucofaríngeas que impidan la correcta entonación de las palabras.

- ✓ El porcentaje de error para este algoritmo de reconocimiento de voz varia en un rango entre 72% y 78%.
- ✓ La implementación de un modelo a escala es fundamental y bastante practico para el mejor entendimiento del funcionamiento del dispositivo como también para su posible comercialización.
- ✓ La interface gráfica permite al usuario tener el control del dispositivo y facilitar su manipulación.
- ✓ El Arduino Uno es un excelente dispositivo y gracias a ser Open Source facilita el desarrollo de este tipo de proyectos basados en la interacción de software con el mundo real permitiendo el intercambio de datos entre los mismos.

6.2.RECOMENDACIONES

- ✓ Para la realización de un software de reconocimiento de voz es importante elegir inicialmente el método de reconocimiento para así buscar un software que cuente con todas las cualidades y funciones necesarias para la programación y posterior ejecución del dispositivo. De igual manera la elección del micrófono es fundamental en la elaboración del programa y aplicación del mismo.
- ✓ Es importante tener una etapa de entrenamiento del sistema ya que así como el software necesita reconocer los comandos a utilizar, el tono y las demás propiedades de la voz del locutor, éste debe conocer los alcances del programa, la forma de pronunciar debidamente las palabras y procurar repetirlas siempre de la forma más similar posible.
- ✓ Hay que tener en cuenta que el sistema es totalmente dependiente de la voz del locutor, por esto es importante evitar hacer entrenamientos del programa o ejecución del mismo en caso de gripa, carraspera, desgaste físico del aparato fonador e incluso tener en cuenta el estado anímico ya que esto puede ocasionar el aumento en el porcentaje de error al momento de utilizar la aplicación.

- ✓ El sistema está obligado a tener una etapa de re-entrenamiento y actualización de su base de datos debido a los cambios de la voz, ya que esta característica del ser humano es un parámetro variable por naturaleza de manera gradual, preferiblemente que el tiempo de este no supere los tres meses.
- ✓ Los métodos de redes neuronales, red neuronal back propagation y correlación entre otros son muy usados a nivel mundial pero requieren un mayor procesamiento y así mismo una mayor complejidad al momento de entender y programar la red, y de igual forma al relacionar las palabras o comandos establecidos.

BIBLIOGRAFIA

- ✓ HENAO, Oscar. "Hardware y Software domótico" Universidad Pontificia Bolivariana. Colombia. 2006.
- ✓ BERNAL BERMÚDEZ. "Reconocimiento de voz y fonética acústica". Madrid, España. 200.
- ✓ RAMIREZ, Ángela, BOJACA, Julián. "Diseño de un modulo para el control de iluminación de hardware domótico a través de un teléfono celular en un ambiente distribuido." Universidad El Bosque. Bogota D.C., Colombia. 2007.
- ✓ Gemello, R., Albesano D., Moisa L. and de Mori R., "Integration of Fixed and Multiple Resolution Analysis in a Speech Recognition System", ICASSP 2001, 2001.
- ✓ Biem A., Katagiri S., McDermott E. and Juang, B. H., "An Application of Discriminative Feature Extraction to Filter-Bank-Based Speech Recognition", IEEE Trans. Speech and Audio Proc., 9 (2), pp. 96-110, febrero 2001.
- ✓ Ferreiros, J., "Aportación a los Métodos de Entrenamiento de Modelos de Markov para Reconocimiento de Habla", Tesis Doctoral, TESIT, Universidad Politécnica de Madrid, 1996.
- ✓ Morales, G. "RECONOCIMIENTO DE COMANDOS DE VOZ CON MAQUINAS DE SOPORTE VECTORIAL A TRAVEZ DE BANDAS ESPECTRALES", Universidad tecnológica de Pereira, Colombia 2007.
- ✓ De Luna, Carlos, Martínez, Julio, Mora, Miguel. "RECONOCIMIENTO DE VOZ CON REDES NEURONALES, DTW Y MODELOS OCULTOS DE MARKOV", Instituto tecnológico de Aguascalientes, México 2006.
- ✓ Delores M. Etter. "SOLUCION DE PROBLEMAS DE INGENIERIA CON MATLAB", Department of Electrical and Computer Engineering, University of Colorado.
- ✓ Peralta, Fernando, Cotrina, Anibal. "ALGORITMO COPER PARA LA DETECCIÓN DE VOZ", Facultad de Ingeniería Electrónica de la UMMSM, Perú 2002.

- ✓ Bernal, J. "METODOLOGIA PARA LA VISUALIZACIÓN DE LOS FORMANTES, BASADA EN LA TRANSFORMADA DE FOURIER, Y SU USO EN EL ESTUDIO ESPECTRAL DE LA FONÉTICA ACÚSTICA ESPAÑOLA", Universidad Politécnica de Madrid, España 1999.
- ✓ Oropeza, J. "ALGORITMOS Y MÉTODOS PARA EL RECONOCIMIENTO DE VOZ EN ESPAÑOL MEDIANTE SILABAS", Centro de investigación en Computación-IPN, México 2006.
- ✓ BECERRA, Nicolai. Diseño e implementación de un control lumínico por medio de la voz. Universidad de San Buenaventura. Bogota D.C., Colombia. 2008

REFERENCIAS DE PAGINAS WEB

- ✓ <http://es.librosintinta.com/busca/MAX%252FMSP/pdf/>
- ✓ <http://arduino.cc/playground/Main/InterfacingWithHardware#Input>
- ✓ <http://www.adictosaltrabajo.com/tutoriales/tutoriales.php?pagina=complementosr>
- ✓ http://www.tuxbrain.com/sites/default/files/ArduinoUNO_Front.jpg
- ✓ <http://www.bricogeek.com/shop/305-945-home/arduino-uno.jpg>
- ✓ <http://www.mathworks.com/matlabcentral/fileexchange/27843>.
- ✓ http://classes.engr.oregonstate.edu/engr/spring2011/engr421-001/group/gp8/MATLAB_CODE/arduino.m.pdf.

ANEXO A

```
clc

% 1) Sincronizacion de MatLab con arduino

delete(instrfind({'Port'},{'COM2'}))

%arduino.m

a=arduino('COM2');

global loop energia t norpa noren Y fourier form reco

a.pinMode(13,'output');
a.pinMode(12,'output');
a.pinMode(11,'output');
a.pinMode(10,'output');
a.pinMode(9,'output');
a.pinMode(8,'output');
a.pinMode(7,'output');
a.pinMode(6,'output');
a.pinMode(2,'output');

a.digitalWrite (2, 1);

% 2)Comprobacion de encendido o apagado de led

loop=1;

if a.digitalRead(12)==1;
a.digitalWrite (12, 1);

else

a.digitalWrite (12,0);

end
```

```
if a.digitalRead(11)==1;
a.digitalWrite (11, 1);
else
a.digitalWrite (11,0);
end

if a.digitalRead(10)==1;
a.digitalWrite (10,1);
else
a.digitalWrite (10, 0);
end

if a.digitalRead(9)==1;
a.digitalWrite (9,1);
else
a.digitalWrite (9,0);
end

if a.digitalRead(8)==1;
a.digitalWrite (8,1);
else
a.digitalWrite (8, 0);
end

if a.digitalRead(7)==1;
a.digitalWrite (7,1);
else
a.digitalWrite (7, 0);
```

```

end

if a.digitalRead(6)==1;
a.digitalWrite (6,1);
else
a.digitalWrite (6, 0);
end

if a.digitalRead(2)==1;
a.digitalWrite (2,1);
else
a.digitalWrite (2, 0);
end

% 3)Captura de la muestra
for lop = 1:500
Fs=8000;
pause(2)
a.digitalWrite (12, 1);
y=wavrecord(2*Fs,Fs,1);
a.digitalWrite (12, 0);
wavwrite(y,Fs,'muestra.wav');
comando=wavread('muestra.wav' );

% 4)Procesamiento de seÒal
pico=length(comando);
comando=comando(2000:1:pico,1);
z=comando;

```

```

winLen=301;
winOverlap=300;
wHamm=hamming(winLen);
sigFramed=buffer(z, winLen, winOverlap, 'nodelay');
sigWindowed = diag(sparse(wHamm)) * sigFramed;
energyST = sum(sigWindowed.^2,1);
t = (0:length(z)-1);
ecero=zeros(1,149);
for e=1:length(ecero)
    energia(1,e)=ecero(1,e);
end
for e=1:length(energyST)
    energia(1,149+e)=energyST(1,e);
end
% 5) Normaliza la energia
normalizar=1;
maxpa=max(z);
difpa=maxpa-normalizar;
porpa=difpa/maxpa;
for tor=1:1:length(z)
    norpa(tor,1)=z(tor,1)-(z(tor,1)*porpa);
end
difpa=max(energia)-normalizar;
poren=difpa/max(energia);

```

```

for tore=1:1:length(energia)
    noren(1,tore)=energia(1,tore)-(energia(1,tore)*poren);
end

clear sigWindowedttortorewHammwinLenwinOverlapmaxpanni
clear difpaeeceroenergySTijknormalizarporenporpasigFramed

% 6) Division por silabas

for d=length(noren):-1:1
    if noren(1,d)>0.15
        enti=noren(1,d);
    end
end

for ddd=1:length(noren)
    if enti==noren(1,ddd)
        posen=ddd;
    end
end

for d2=1:1:length(noren)
    if noren(1,d2)>0.108
        enti2=noren(1,d2);
    end
end

for ddd2=1:length(noren)
    if enti2==noren(1,ddd2)
        posen2=ddd2;
    end
end

```

```

end
end
energia2=noren(posen:1:posen2);
cortesil=min(energia2);
for y=1:length(energia2)
if cortesil==energia2(1,y)
    posminen=y;
end
end
posminpa=posminen+posen;
clear dd2dddddd2entienti2
silaba1=norpa(posen:posminpa);
for d2=1:1:length(noren)
if noren(1,d2)>0.05
    enti2=noren(1,d2);
end
end
for ddd2=1:length(noren)
if enti2==noren(1,ddd2)
    posibfin=ddd2;
end
end
silaba2=norpa(posminpa+1:posibfin);
clear cortesild2ddd2energiaenergia2enti2posminenyz

```

```

% 7)Transformada rapida de fourier y determinacion de "f1","f2","f3" y
% "f4"

z=silaba1;
times=30;
lenz=length(z);
lenven=((Fs*2)*(times/1000))/2;
sol=50;
lensol=(lenven*sol)/100;
vens=(floor(lenz/lenven)*2)-1;
v = zeros(lenven,vens);
vseg = zeros(lenven,vens);
for i=0:(vens-1)
    v(:,i+1)=z((lensol*i)+1:1:(lensol*i)+lenven);
    w = hamming(lenven);
    vseg(:,i+1) = w.*v(:,i+1);
    NFFT = 2^nextpow2(length(vseg));
    Y(:,i+1) = fft(vseg(:,i+1),NFFT)/lenven;
    f = Fs/2*linspace(0,1,NFFT/2);
    fourier(:,i+1)=2*abs(Y(:,i+1));
end
fourier2=fourier(1:128,1:vens);
fourier3 = max(fourier2');
max1=max(fourier3(16:length(fourier3)));
posib1=0;

```

```

for x=1:length(fourier3)
    if max1==fourier3(1,x)
        posib1=x;
    end
end

f1=f(posib1);
eli=15;
max2=max(fourier3(posib1+eli:length(fourier3)));

for w=1:length(fourier3)
    if max2==fourier3(1,w)
        posib2=w;
    end
end

f2=f(posib2);
z=silaba2;
times=30;
lenz=length(z);
lenven=((Fs*2)*(times/1000))/2;
sol=50;
lensol=(lenven*sol)/100;
vens=(floor(lenz/lenven)*2)-1;
v = zeros(lenven,vens);
vseg = zeros(lenven,vens);

```

```

for i=0:(vens-1)
    v(:,i+1)=z((lensol*i)+1:1:(lensol*i)+lenven);
    w = hamming(lenven);
    vseg(:,i+1) = w.*v(:,i+1);
    NFFT = 2^nextpow2(length(vseg));
    Y(:,i+1) = fft(vseg(:,i+1),NFFT)/lenven;
    f = Fs/2*linspace(0,1,NFFT/2);
    fourier(:,i+1)=2*abs(Y(:,i+1));
end

fourier2=fourier(1:128,1:vens);
fourier4 = max(fourier2');
max3=max(fourier4(10:length(fourier4)));
posib3=0;
for x=1:length(fourier4)
    if max3==fourier4(1,x)
        posib3=x;
    end
end

f3=f(posib3);
eli=12;
max4=max(fourier4(posib3+eli:length(fourier4)));
for w=1:length(fourier4)
    if max4==fourier4(1,w)
        posib4=w;
    end
end

```

```

end
end

f4=f(posib4);
f1=fix(f1);
f2=fix(f2);
f3=fix(f3);
f4=fix(f4);
form=[f1 f2 f3 f4];
clear max2max3max4soltimesvvensvsegwxz
clear posenposen2posib1posib2posib3posib4posibfin
clear NFFTYfourierfourier2fourier3fourier4ilensollenvenlenzmax1

% 8) Comprobacion de comandos por formantes y ejecucion de orden
reco=0;
% SALA
if 590<f1 && f1<890
if 1390<f2 && f2<1610
if 590<f3 && f3<890
if 1410<f4 && f4<1580
reco=1;
if a.digitalRead(11)==1;
a.digitalWrite (11, 0);
else
a.digitalWrite (11, 1);
end

```

```

end
end
end
end

% TINA

if 2400<f1 && f1<3300

if 2900<f2 && f2<4000

if 280<f3 && f3<320

if 740<f4 && f4<890

        reco=2;

if a.digitalRead(10)==1;

        a.digitalWrite (10, 0);

else

        a.digitalWrite (10, 1);

end

end

end

end

end

end

% BALCON

if 570<f1 && f1<780

if 1300<f2 && f2<1590

if 320<f3 && f3<680

if 1000<f4 && f4<1700

```

```

        reco=3;
    if a.digitalRead(9)==1;
        a.digitalWrite (9, 0);
    else
        a.digitalWrite (9, 1);
    end
end
end
end
end
end
end
% MESON
if 450<f1 && f1<700
if 1900<f2 && f2<3100
if 270<f3 && f3<550
if 1000<f4 && f4<1300
        reco=4;
    if a.digitalRead(8)==1;
        a.digitalWrite (8, 0);
    else
        a.digitalWrite (8, 1);
    end
end
end
end
end
end

```

```

end

% JARDIN

if 620<f1 && f1<850

if 1340<f2 && f2<1650

if 300<f3 && f3<360

if 2450<f4 && f4<3090

        reco=5;

if a.digitalRead(7)==1;

        a.digitalWrite (7, 0);

else

        a.digitalWrite (7, 1);

end

end

end

end

end

end

% UNO

if 465<f1 && f1<478 || 650<f1 && f1<900

if 1210<f2 && f2<1580

if 270<f3 && f3<520 || 580<f3 && f3<720

if 1000<f4 && f4<1240

        reco=6;

if a.digitalRead(6)==1;

        a.digitalWrite (6, 0);

```

```

else
    a.digitalWrite (6, 1);
end
end
end
end
end
end
% PISO
if 2000<f1 && f1<3100
if 2500<f2 && f2<4100
if 270<f3 && f3<650
if 950<f4 && f4<1350
reco=7;
    a.digitalWrite (11,1);
a.digitalWrite (10,1);
    a.digitalWrite (9,1);
    a.digitalWrite (8,1);
    a.digitalWrite (7,1);
a.digitalWrite (6,1);
end
end
end
end
end

```

ANEXO B

```
function varargout = sonic(varargin)

% SONIC M-file for sonic.fig

%   SONIC, by itself, creates a new SONIC or raises the existing
%   singleton*.

%   H = SONIC returns the handle to a new SONIC or the handle to
%   the existing singleton*.

%   SONIC('CALLBACK', hObject,eventData,handles,...) calls the local
%   function named CALLBACK in SONIC.M with the given input arguments.

%   SONIC('Property','Value',...) creates a new SONIC or raises the
%   existing singleton*. Starting from the left, property value pairs are
%   applied to the GUI before sonic_OpeningFcn gets called. An
%   unrecognized property name or invalid value makes property application
%   stop. All inputs are passed to sonic_OpeningFcn via varargin.

%   *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
%   instance to run (singleton)".

% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help sonic

% Last Modified by GUIDE v2.5 27-Oct-2011 21:47:53

% Begin initialization code - DO NOT EDIT

gui_Singleton = 1;

gui_State = struct('gui_Name',    mfilename, ...
'gui_Singleton', gui_Singleton, ...
'gui_OpeningFcn', @sonic_OpeningFcn, ...
```

```

'gui_OutputFcn', @sonic_OutputFcn, ...
'gui_LayoutFcn', [], ...
'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

% End initialization code - DO NOT EDIT

% --- Executes just before sonic is made visible.

function sonic_OpeningFcn(hObject, ~, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to sonic (see VARARGIN)

% Choose default command line output for sonic
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes sonic wait for user response (see UIRESUME)

```

```
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.

function varargout = sonic_OutputFcn(~, ~, handles)

% varargout cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

clc
```

ANEXO C

Manual de funcionamiento

Sonic 1.0

Luego de conectar nuestro ordenador por medio de USB al dispositivo e iniciar el software Sonic 1.0, el ordenador procede a mostrarlos la interfaz gráfica de usuario que le permite interactuar con el software de una forma muy simple.

Imagen 1. Interface grafica



Ya abierta la interface grafica (Imagen 1.) se encuentran tres botones que permiten la operación del hardware. El botón de inicio da la orden al software de empezar el ciclo y al mismo tiempo enciende un led verde que indica que el sistema esta en funcionamiento. Al iniciar el ciclo repetitivo se enciende un LED azul cada 15 segundos aproximadamente durante dos segundos tiempo en el cual el usuario debe mencionar la palabra que quiere ser identificada; después de esto el software pasa a hacer la comprobación de los datos adquiridos y por ultimo a ejecutar la orden designada y encender o apagar el led asignado dependiendo su estado inicial, en caso de no reconocer la palabra simplemente se repite el ciclo.

Imagen 2. Interface grafica – Iniciar



Posteriormente encontramos el botón Reset que como su nombre lo indica es el encargado de reiniciar el software en caso de presentarse algún error en la ejecución del mismo; fue necesaria la implementación de este botón debido a que en algunas ocasiones el hardware mantiene ocupando el puerto serial destinado para la transferencia de datos y simplemente al tratar de iniciar de nuevo el software indicaba que el puerto ya estaba siendo ocupado por otro programa.

Imagen 3. Interface grafica – Reset



Por ultimo encontramos el botón cerrar, que es el encargado de dar fin al ciclo repetitivo ya mencionado y cerrar la interfaz gráfica dejando así mismo el puerto serial libre para ser utilizado por cualquier otro dispositivo que sea conectado al ordenador.

Imagen 4. Interface grafica – Cerrar

