



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLÁN

**ADQUISICIÓN DE DATOS, PREPROCESAMIENTO Y
RECONOCIMIENTO DE PATRONES PARA MATRICES
DE SENSORES SAW-UV.**

TESIS

Que para obtener el título de

INGENIERO MECÁNICO ELECTRICISTA

PRESENTA:

BERENICE VÁZQUEZ GONZÁLEZ

ALEJANDRO HERNÁNDEZ MORA

ASESOR:

Ing. JOSÉ LUIS BARBOSA PACHECO

CO-ASESOR:

Dr. DANIEL MATATAGUI CRUZ



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLÁN
 SECRETARÍA GENERAL
 DEPARTAMENTO DE EXÁMENES PROFESIONALES

ASUNTO: VOTO APROBATORIO



DEPARTAMENTO DE
 EXÁMENES PROFESIONALES

M. en C. JORGE ALFREDO CUÉLLAR ORDAZ
 DIRECTOR DE LA FES CUAUTITLÁN
 PRESENTE

ATN: I.A. LAURA MARGARITA CORTAZAR FIGUEROA
 Jefa del Departamento de Exámenes Profesionales
 de la FES Cuautitlán.

Con base en el Reglamento General de Exámenes, y la Dirección de la Facultad, nos permitimos comunicar a usted que revisamos el: Trabajo de Tesis

“Adquisición de datos, preprocesamiento y reconocimiento de patrones para matrices de sensores SAW-UV”

Que presenta el pasante: ALEJANDRO HERNÁNDEZ MORA

Con número de cuenta: 30617622-7 para obtener el Título de la carrera: Ingeniería Mecánica Eléctrica

Considerando que dicho trabajo reúne los requisitos necesarios para ser discutido en el EXAMEN PROFESIONAL correspondiente, otorgamos nuestro VOTO APROBATORIO.

ATENTAMENTE

“POR MI RAZA HABLARÁ EL ESPÍRITU”

Cuautitlán Izcalli, Méx. a 27 de Agosto de 2018.

PROFESORES QUE INTEGRAN EL JURADO

	NOMBRE	FIRMA
PRESIDENTE	Ing. Nicolás Calva Tapia	
VOCAL	M. en T.I. Jorge Buendía Gómez	
SECRETARIO	Ing. José Luis Barbosa Pacheco	
1er. SUPLENTE	M. en C. Leopoldo Martín Del Campo Ramírez	
2do. SUPLENTE	Dr. David Tinoco Varela	

NOTA: los sinodales suplentes están obligados a presentarse el día y hora del Examen Profesional (art. 127).

LMCF/ntm*



UNIVERSIDAD NACIONAL
AUTÓNOMA DE
MÉXICO

FACULTAD DE ESTUDIOS SUPERIORES CUAUTITLÁN
SECRETARÍA GENERAL
DEPARTAMENTO DE EXÁMENES PROFESIONALES



ASUNTO: VOTO APROBATORIO

DEPARTAMENTO DE
EXÁMENES PROFESIONALES

M. en C. JORGE ALFREDO CUÉLLAR ORDAZ
DIRECTOR DE LA FES CUAUTITLÁN
PRESENTE

ATN: I.A. LAURA MARGARITA CORTAZAR FIGUEROA
Jefa del Departamento de Exámenes Profesionales
de la FES Cuautitlán.

Con base en el Reglamento General de Exámenes, y la Dirección de la Facultad, nos permitimos comunicar a usted que revisamos el: Trabajo de Tesis

"Adquisición de datos, preprocesamiento y reconocimiento de patrones para matrices de sensores SAW-UV"

Que presenta la pasante: BERENICE VÁZQUEZ GONZÁLEZ

Con número de cuenta: 41100555-7 para obtener el Título de la carrera: Ingeniería Mecánica Eléctrica

Considerando que dicho trabajo reúne los requisitos necesarios para ser discutido en el EXAMEN PROFESIONAL correspondiente, otorgamos nuestro VOTO APROBATORIO.

ATENTAMENTE

"POR MI RAZA HABLARÁ EL ESPÍRITU"

Cuautitlán Izcalli, Méx. a 27 de Agosto de 2018.

PROFESORES QUE INTEGRAN EL JURADO

	NOMBRE	FIRMA
PRESIDENTE	Ing. Nicolás Calva Tapia	
VOCAL	M. en T.I. Jorge Buendía Gómez	
SECRETARIO	Ing. José Luis Barbosa Pacheco	
1er. SUPLENTE	M. en C. Leopoldo Martín Del Campo Ramírez	
2do. SUPLENTE	Dr. David Tinoco Varela	

NOTA: los sinodales suplentes están obligados a presentarse el día y hora del Examen Profesional (art. 127).

LMCF/ntm*

AGRADECIMIENTOS

No hay palabras que puedan expresar la profunda gratitud que representa la culminación de este proyecto y a todas aquellas personas que estuvieron involucradas durante todo este tiempo. Quiero agradecer a mis primeros maestros David y Lilia, que me educaron con los valores que me han formado como el ser humano que soy ahora, gracias por mostrar fortaleza en los tiempos difíciles y por compartir tanto los momentos agradables como los desafortunados, y gracias porque sé que sin importar lo que suceda y donde quiera que nos encontremos, siempre estaremos presentes para seguir apoyándonos. Los amo mucho papá y mamá.

Agradezco al laboratorio de dispositivos biomédicos del Instituto de Ciencias Aplicadas y Tecnología de la UNAM por abrirnos sus puertas e instalaciones y a los proyectos del PAPPIT IA103016 y TA100118 que permitieron el desarrollo de este proyecto así como a todos los integrantes que conforman el laboratorio, en especial al Dr. Daniel Matatagui Cruz por brindarnos su tiempo, conocimiento y paciencia. A la facultad de estudios superiores Cuautitlán campo 4, que en sus aulas descubrí el interés por la ciencia, tecnología e innovación y sobre todo quisiera expresarle un profundo agradecimiento al Ing. José Luis Barbosa Pacheco que mediante sus clases y proyectos me inculco el gusto por el desarrollo tecnológico. Por medio de estas líneas quisiera dedicarles un agradecimiento a nuestros sinodales por su apoyo y tiempo que dedicaron en la revisión y corrección de esta tesis.

Agradezco a mi familia, a mi abuelita Irma, a mi tío Oscar y a mis tías Leticia, Araceli y Nora que estuvieron presentes en cada momento, otorgando sus palabras de aliento y mucho más para seguir adelante y concluir esta etapa.

Agradezco a la persona que me acompañó durante la carrera, en la cual pasamos grandes momentos y otros no muy gratos, que a pesar de todo decidió concluir este camino junto conmigo. Muchas gracias Berenice por tu paciencia, cariño y por todo lo que has compartido conmigo.

Y por último, pero no por ello menos importantes agradecer a todos aquellos que han estado presentes desde el inicio de esta aventura en la UNAM, gracias por brindarme su cariño y amistad, noches de desvelo y tardes de diversión, algún consejo o regaño, por ser más que amigos mis hermanos.

Alejandro Hernández Mora

Son varias las personas que han contribuido en el proceso y la culminación de esta tesis, en primer lugar quiero agradecer a esas personas que estuvieron conmigo desde el inicio de mi vida, a mis padres Blanca y Leonides, gran parte de mis logros se los debo a ustedes porque con su constante motivación me han impulsado a alcanzar mis metas, gracias por ser mis maestros de vida y acompañarme en cada paso que doy.

Agradezco profundamente al Dr. Daniel Matatagui Cruz por dirigir esta tesis, por su paciencia, por el tiempo que empleó en transmitirme sus conocimientos e ideas y por ser una guía en cada una de las etapas que conforman este proyecto. Quiero mostrar mi más profundo agradecimiento al Ing. José Luis Barbosa Pacheco por su apoyo incondicional, por brindarme la oportunidad de aprender cosas nuevas y por todos esos consejos que me ayudaron a crecer profesionalmente. Le agradezco al Instituto de Ciencias Aplicadas y Tecnología de la UNAM por permitirme formar parte de su comunidad durante el desarrollo de esta tesis y al PAPPIT por otorgarme la oportunidad de colaborar en los proyectos IA103016 y TA100118, de igual forma quiero expresar mi gratitud a la Facultad de Estudios Superiores Cuautitlán Campo 4, por las diversas experiencias que adquirí dentro de ella, tanto a nivel profesional como personal, a cada uno de los profesores que contribuyeron a mi formación académica por compartir sus conocimientos, quiero agradecerles también por cada momento que dedicaron a aclarar mis dudas. No quería pasar por alto la oportunidad de agradecer a cada uno de nuestros sinodales por su ayuda, comprensión y sobre todo por el tiempo que invirtieron en la revisión de este proyecto.

A través de estas líneas te agradezco Alejandro por todo el camino que hemos recorrido juntos, porque a pesar de que no todas nuestras experiencias han sido buenas siempre has estado allí para apoyarme, gracias a ti y a tu familia por permitirme formar parte de sus vidas, por brindarme su cariño, paciencia y por tratarme como un miembro más de la familia. Asimismo quiero agradecer a todas aquellas personas que me brindaron una amistad sincera, gracias por sus consejos, su apoyo, por cada palabra de aliento, por la confianza que depositan en mí y por hacerme ver que de los errores también podemos obtener grandes enseñanzas.

Berenice Vázquez González

Índice

Índice de Figuras	iii
Índice de Tablas	vi
Introducción	1
Hipótesis	4
Marco de Trabajo	5
Justificación	5
Objetivo	6
Objetivos particulares	6
Capítulo 1. Fundamentos teóricos de los sensores de ondas acústicas superficiales (SAW)	7
1.1 Generación de ondas acústicas en sustratos piezoeléctricos	7
1.1.1 Efecto piezoeléctrico	8
1.2 Sensores basados en ondas acústicas	8
1.2.1 Ondas de Volumen (Bulk Acoustic Wave, BAW)	10
1.2.2 Ondas Acústicas Superficiales (Surface Acoustic Waves, SAW)	11
1.2.3 Ondas de Rayleigh	14
1.2.4 Ondas Horizontales Transversales SAW (SH-SAW)	14
1.2.5 Ondas Love (LW)	15
1.2.6 Ondas Horizontales Transversales (SH-Acoustic Plate Mode, APM)	17
1.2.7 Ondas Lamb (Flexural Plate Waves)	18
1.3 ¿Por qué sensores de ondas acústicas?	19
1.4 Comparación de rendimiento	20
Capítulo 2. Adquisición de datos	23
2.1 Sistema de adquisición de datos (DAQ)	26
2.2 Hardware para adquisición de datos	27
2.2.1 Características de la Beaglebone Black Rev C (BBB)	27

2.2.2 La Beaglebone Black como hardware de adquisición de datos	29
2.2.3 RTL-SDR	30
Capítulo 3. Procesamiento de datos	39
3.1 Reducción de variables	39
3.1.1 Análisis de Componentes Principales (PCA)	40
3.2 Análisis y clasificación de patrones	44
3.2.1 Elección del modelo	49
3.2.2 Redes Neuronales Artificiales (ANN, Artificial Neural Network)	50
3.2.3 Entrenamiento del clasificador	54
3.2.4 Validación de los datos	58
Capítulo 4. Implementación del sistema	61
4.1 Matriz de sensores	61
4.2 Estructura del software	66
4.2.1 Adquisición de datos	67
4.2.2 Reducción de variables (PCA)	74
4.2.3 Clasificador (ANN)	77
4.3 Interfaz gráfica de usuario	82
Conclusiones	93
Bibliografía	95
Apéndice	99

Índice de Figuras

Introducción

Figura 1. Clasificación de los sensores de estado sólido más comunes	3
Figura 2. Esquema general del análisis de compuestos volátiles	4

Capítulo 1. Fundamentos teóricos de los sensores de ondas acústicas superficiales (SAW)

Figura 1.1. Clasificación de los sensores de onda acústica (según la onda propagada)	10
Figura 1.2. Balanza de cuarzo y su modo fundamental de oscilación	11
Figura 1.3. Esquema de un sensor de ondas acústicas superficiales	12
Figura 1.4. Perfil de un IDT con los electrodos polarizados, generando diferentes modos de una onda	12
Figura 1.5. Vista superior de un IDT que genera una onda al aplicar una diferencia de potencial alterno entre electrodos	13
Figura 1.6. Sensor basado en la propagación de ondas Rayleigh	14
Figura 1.7. Esquema de un dispositivo SH-SAW	15
Figura 1.8. Esquema de un dispositivo LW y esquema del perfil de un sensor Love	16
Figura 1.9. Esquema de un dispositivo basado en ondas SH-APM	17
Figura 1.10. Esquema de un dispositivo basado en ondas Lamb y sus diferentes modos de propagación	18

Capítulo 2. Adquisición de datos

Figura 2.1. Reflexión interna de las ondas superficiales en un transductor de Electrodoes dobles	24
Figura 2.2. Configuración de un oscilador	25
Figura 2.3. Beaglebone Black Rev C y sus elementos	28
Figura 2.4. Dispositivo RTL-SDR y sus componentes internos	30

Figura 2.5. Diagrama del funcionamiento del RTL-SDR	30
Figura 2.6. a) Representación del proceso downconverter b) Diagrama de los Puertos del mezclador incluido en el sintonizador	32
Figura 2.7. Bloques del muestreo por cuadratura	32
Figura 2.8. Comportamiento del sensor	36
Capítulo 3. Procesamiento de datos	
Figura 3.1. Elementos básicos de una Red Neuronal Artificial	53
Figura 3.2. Ejemplo de una red neuronal multicapa	57
Capítulo 4. Implementación del sistema	
Figura 4.1. Diagrama general del proyecto	61
Figura 4.2. Diseño de la matriz de sensores	62
Figura 4.3. Esquema del proceso de ionización	63
Figura 4.4. Interior de la matriz de sensores	64
Figura 4.5. Circuito de leds y sensor de temperatura	64
Figura 4.6. Diagrama del circuito del sensor de temperatura ChipCap2 (CC2)	65
Figura 4.7. Matriz de sensores en funcionamiento	65
Figura 4.8. Diagrama del proceso general de la nariz electrónica	66
Figura 4.9. Diagrama del proceso de adquisición de datos	67
Figura 4.10. Digitalización de la señal de entrada	68
Figura 4.11. Señal en el dominio de la frecuencia	69
Figura 4.12. Punto de la espiga más grande	71
Figura 4.13. Señal respuesta obtenida con una frecuencia de fuente de 158.5MHz	71
Figura 4.14. Respuesta para diferentes frecuencias de fuente	72
Figura 4.15. Respuesta de un sensor expuesto a acetona	73
Figura 4.16. Diagrama de flujo de la reducción de variables	71
Figura 4.17. Agrupación de las componentes principales	76

Figura 4.18. Pesos de las dos primeras componentes principales	76
Figura 4.19. Diagrama de flujo del clasificador	77
Figura 4.20. Ventana de entrenamiento	79
Figura 4.21. Gráfica del rendimiento de la red en sus diferentes etapas de aprendizaje	80
Figura 4.22. Gráfica del ajuste entre los valores estimados y predichos	80
Figura 4.23. Gráfica de la matriz de porcentajes de acierto y error en la clasificación	81
Figura 4.24. Asignación de las clases	81
Figura 4.25. Ventana de inicio	82
Figura 4.26. Ventana de adquisición de datos	83
Figura 4.27. Configuración del RTL-SDR	83
Figura 4.28. Adquisición de la señal	84
Figura 4.29. Exportación de datos	84
Figura 4.30. Ventana de reducción de variables	85
Figura 4.31. Importación de la matriz respuesta	85
Figura 4.32. Valor de la componentes principales	86
Figura 4.33. Gráficas de los PC's	86
Figura 4.34. Exportación de los PC's	87
Figura 4.35. Ventana del proceso de clasificación	87
Figura 4.36. Importación de datos	88
Figura 4.37. Creación de la red neuronal	88
Figura 4.38. Entrenamiento de la red	89
Figura 4.39. Gráfica del estado de entrenamiento	89
Figura 4.40. Gráfica del rendimiento de la red en las tres etapas de aprendizaje	90
Figura 4.41. Gráfica de la matriz de confusión	90
Figura 4.42. Ventana de resultados	91
Figura 4.43. Exportación de nuevos datos	91

Figura 4.44. Ventana de resultados para nuevos datos	92
Figura 4.45. Sistema finalizado	92

Índice de Tablas

Capítulo 1. Fundamentos teóricos de los sensores de ondas acústicas superficiales (SAW)

Tabla 1.1. Resumen de las características de los dispositivos AW	21
Tabla 1.2. Ventajas y desventajas de los diferentes dispositivos AW	22

Capítulo 2. Adquisición de datos

Tabla 2.1. Tipos de sintonizadores y sus frecuencias	31
--	----

Capítulo 3. Procesamiento de datos

Tabla 3.1. Resultados de un modelo clasificador de 100 compuestos	47
---	----

Capítulo 4. Implementación del sistema

Tabla 4.1. Autovalores de la matriz diagonal	75
--	----

Introducción

La nariz humana como herramienta en la detección de enfermedades a través del aliento ha sido una técnica empleada desde la antigüedad, *Hipócrates de Cós* (460-377 a.C.) enseñaba a sus alumnos que el aroma de la respiración humana podía proporcionar pistas para la detección de múltiples enfermedades, pero con el paso del tiempo fue quedando relegada debido al surgimiento de nuevos métodos de diagnóstico. Sin embargo, en la actualidad podemos encontrar múltiples estudios que refieren cómo determinadas enfermedades producen un desequilibrio en los compuestos químicos que integran el aire que exhalamos, a partir de estos estudios se ha retomado el interés por integrar el olfato como un método de diagnóstico de enfermedades.

Las *pruebas organolépticas olfativas* son un recurso muy usado en estudios epidemiológicos, básicamente se trata de medir el aliento de una persona usando el olfato de evaluadores humanos (“jueces”). La complejidad de este tipo de pruebas se manifiesta al momento de seleccionar a los jueces, estos deben ser previamente testeados para confirmar su capacidad de percibir olores, entrenados para que sepan aplicar una escala correctamente (ya que la prueba la realizan distintos jueces a un solo paciente, se busca que los jueces tengan valores similares entre sí); después de que cada uno de los jueces evalúa al paciente, se hace un análisis estadístico con el fin de comprobar la correlación y la concordancia entre los jueces [1]. Todo este proceso además de tomarse un tiempo considerable para llevarse a cabo y ser bastante costoso, es subjetivo ya que está sometido a variables psicológicas y fisiológicas.

Con el gran avance tecnológico se ha logrado sustituir casi por completo las pruebas organolépticas, aumentando así la precisión de la medición del aliento. Actualmente existen muchos sistemas comerciales que se encargan de analizar el aliento, sin embargo, los sistemas que poseen alta sensibilidad y selectividad por lo regular son altamente costosos, robustos y de difícil manejo.

Con el afán de mejorar y reducir el costo de las técnicas de detección olfativa se han implementado instrumentos que emulan el mecanismo del sistema olfativo humano, estos instrumentos son conocidos como *narices electrónicas*, estas son definidas como “un instrumento que comprende una agrupación de sensores químicos con sensibilidades

parcialmente solapadas junto a un sistema de reconocimiento de patrones, capaz de reconocer aromas simples o complejos”. A pesar de que su término nació en la década de los 80’s, su fabricación comenzó en la segunda mitad de los 90’s [2], quizá la principal razón de este retraso se deba al ambiente multidisciplinar que conlleva desarrollar este tipo de sistemas y, a que en ese tiempo se carecía de la tecnología que tenemos en el presente, pero gracias a que se ha podido profundizar y adquirir un conocimiento más extenso en el campo de la inteligencia artificial y al surgimiento de la microelectrónica, que ha permitido la fabricación de sensores químicos integrados, se han llegado a construir equipos que realizan un proceso similar al del olfato humano.

Los receptores olfativos biológicos están representados por una matriz de sensores (grupo de sensores químicos) que producen una señal eléctrica dependiente del tipo de compuesto¹ detectado, el bulbo olfativo humano recibe estas señales para posteriormente enviarlas al cerebro a través de los nervios que, de forma muy similar a las técnicas de pre procesado, preparan la señales reduciendo el volumen de la información y minimizan el ruido que introducen los receptores. Esto último facilita la clasificación o identificación de la muestra usando sistemas de reconocimiento de patrones [3].

Así, como el olfato biológico no necesita identificar cada compuesto de una mezcla para poder reconocerlo, el olfato artificial valora las muestras en su conjunto, sin identificar los componentes básicos que constituyen un aroma complejo [3], pero es muy importante tener en cuenta que los sensores químicos que se integran dentro de la matriz no son específicos, es decir, que no son cien por ciento selectivos a un compuesto químico dado, pero sí ligeramente más sensibles a determinadas familias químicas (solventes orgánicos, ácidos grasos, gases sulfurosos, etc.). Es así como la respuesta que emiten los sensores producen señales características para cada mezcla química, siendo sensibles a una amplia variedad de compuestos [4]. Los sensores son una parte fundamental para el funcionamiento de la nariz electrónica y, a pesar de que existen una innumerable cantidad de sensores en el mercado, el sistema de detección usado en este proyecto está basado en *sensores de estado sólido*, que se eligieron debido a su fácil manejo y a su bajo costo, la desventaja aparente es que carecen de sensibilidad y selectividad cuando se aplican en casos reales, pero al combinarlos con nanotecnología se puede hacer de estos sensores una herramienta efectiva

¹ Se han detectado más de 3000 compuestos distintos en el aliento y se ha demostrado que el aire exhalado por una persona puede albergar más de 200 compuestos orgánicos volátiles (CVO’s).

en la detección de compuestos, lo que nos permite detectar muy bajas concentraciones de los marcadores de las enfermedades.

En la figura 1 se muestra un esquema de los diferentes tipos de sensores de estado sólido que han sido utilizados comúnmente en estudios recientes para conformar la matriz de sensores de los sistemas de detección, la presente tesis se concentra en el estudio de los sensores de *ondas acústicas superficiales* SAW (por sus siglas en inglés *Surface Acoustic Waves*) que se encuentran dentro de la tipología de los sensores basados en ondas acústicas AW, cuya descripción y funcionalidad serán descritos con mayor profundidad en el capítulo 1.

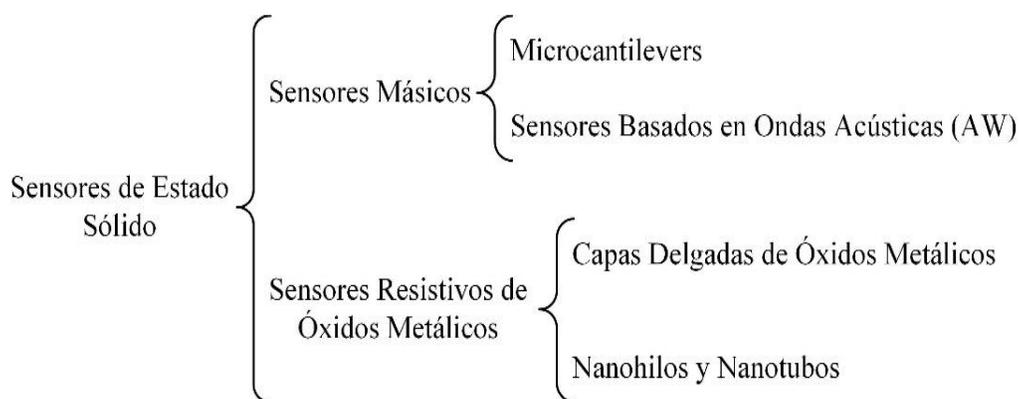


Figura 1. Clasificación de los sensores de estado sólido más comunes.

La matriz de sensores forma únicamente el principio, pero para llegar a nuestra meta que es identificar, clasificar e interpretar los diferentes compuestos volátiles que conforman el aliento por medio de un sistema de olfato electrónico, es necesario tener presente que existen otros módulos que son igual de importantes dentro de dicho análisis.

En la figura 2 ilustramos los tres módulos que conforman el proceso general para el análisis de compuestos volátiles; sin embargo, este proyecto se encuentra enfocado en las secciones de electrónica y software.

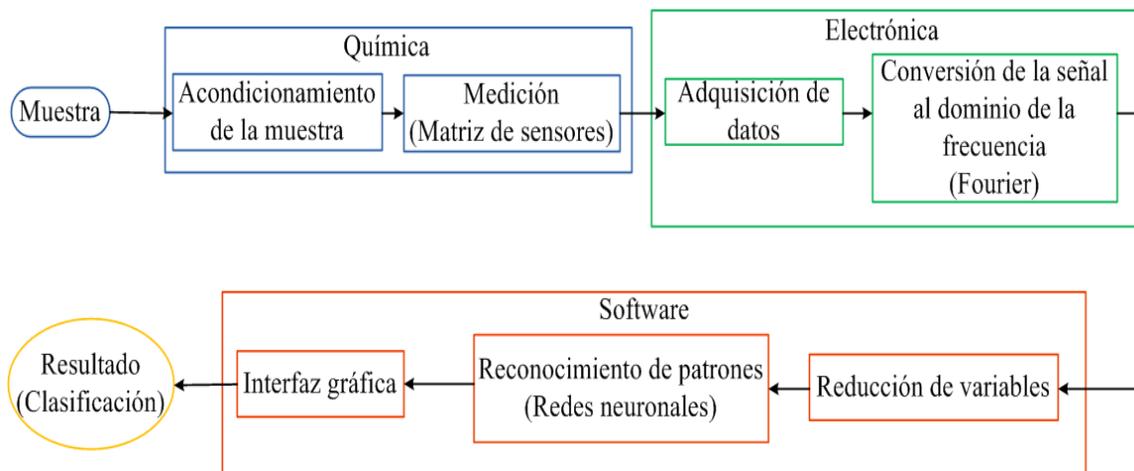


Figura 2. Esquema general del análisis de compuestos volátiles.

La ciencia y la tecnología combinadas han arrojado resultados sumamente favorables en el campo médico así como en otras disciplinas y han permitido, a través de la experimentación científica, emplear las matrices de sensores de estado sólido complementadas con técnicas de adquisición de datos y reconocimiento de patrones (usando modelos de redes neuronales), para obtener un método de diagnóstico no invasivo, de bajo costo, sencillo de utilizar, portátil y que además de todo permite identificar alteraciones pequeñas de los compuestos químicos, lo que abre una ventana a la detección temprana de enfermedades con el objetivo de minimizar los efectos negativos sobre los pacientes.

Hipótesis

- 1) Al aplicar iluminación UV-visible es posible mejorar la sensibilidad de los sensores de estado sólido, así como aumentar la velocidad de toma de muestras para su medición.
- 2) Las matrices de sensores proporcionan una huella única para cada muestra, por lo que es viable el uso de técnicas de reconocimiento de patrones para clasificar y discriminar las respuestas de cada uno de los sensores que conforman la matriz.

Marco de Trabajo

La presente tesis se llevó a cabo en el CCADET-UNAM bajo el *Programa de Apoyo a Proyectos de Investigación e Innovación Tecnológica* (PAPIIT) IA103016 y TA100118, titulado “Matrices de sensores basados en capas sensibles nano-estructuradas con aplicación en la detección temprana de enfermedades”.

Justificación

La detección selectiva de diversos compuestos volátiles en los olores muestreados del cuerpo humano es de gran importancia para el diagnóstico clínico temprano. Para poder investigar las características de olor de las muestras, se han aplicado diversos métodos, siendo la evaluación humana la principal herramienta de diagnóstico durante años, sin embargo, actualmente existe una creciente demanda de diagnósticos económicos, no invasivos, simples y rápidos. El análisis del aliento por medio de sensores se ha vuelto cada vez más significativo en aplicaciones médicas, puesto que permiten la obtención de los perfiles del olor y la detección temprana de los marcadores de la enfermedad, de una forma rápida, precisa y no invasiva, y con ello determinar el tipo de tratamiento requerido por el paciente. Esta técnica nos proporciona una valiosa herramienta para detectar y controlar la enfermedad, minimizando los efectos que pueden resultar perjudiciales en los pacientes.

Las narices electrónicas pueden resultar de gran ayuda en casos en los que el paciente necesite una constante monitorización de parámetros diarios (tales como la glucemia y la urea) ya que pueden realizar esta tarea eficientemente y sin causar daño. Debido a que cada sensor mide las diferentes especies químicas del aliento exhalado, al juntar cada una de las mediciones en una matriz se obtiene una respuesta con características únicas que ayuda a diagnosticar el tipo de enfermedad que presenta el paciente. Sin embargo, los sensores deben responder ante trazas de partes por millón (ppm) y partes por billón (ppb) para poder llegar a obtener una respuesta de las matrices de sensores ante las variaciones de especies químicas en el aliento. Por esta razón, se espera que al excitar los sensores con radiación UV aumenten su sensibilidad y mejoren su tiempo de respuesta. Además, debido al alto

número de especies químicas que se encuentran en el aliento se requieren de algoritmos de pre-procesado de señales de reconocimiento de patrones para poder llevar a cabo el diagnóstico de las enfermedades.

Objetivo

Desarrollar una plataforma que permita excitar los sensores químicos con radiación UV para aumentar su sensibilidad, complementando dicha instrumentación con adquisición de datos y reconocimiento de patrones para llevar a cabo medidas de especies químicas *in situ* y en tiempo real.

Objetivos particulares.

- ❖ Diseñar e implementar la infraestructura que permita adquirir la respuesta de los sensores para usarlos como sistema de análisis.
- ❖ Desarrollar el software para adquirir, procesar, clasificar y discriminar la respuesta de salida utilizando una tarjeta de desarrollo con procesador ARM Cortex A8 (BeagleBone Black).
- ❖ Desarrollar la interfaz hombre-máquina que muestre los resultados de forma sencilla y fácil de interpretar.

Capítulo 1

Fundamentos teóricos de los sensores de ondas acústicas superficiales (SAW)

Podemos encontrar diversas aplicaciones en las que han sido empleados los dispositivos de ondas acústicas (*Acoustic Wave, AW*); en la industria de las telecomunicaciones se encuentran como filtros, resonadores o líneas de retardo para altas frecuencias, mientras que como sensores son usados principalmente en los sistemas de detección en diversos campos como la medicina (detección de virus, toxinas, bacterias, temperatura ...), medio ambiente (detección de gases), alimentación (detección de aromas, toxinas, etc.), detección de magnitudes físicas (presión, medidas de alta tensión, temperatura...), etc. [5]

El descubrimiento del efecto piezoeléctrico, tanto directo como inverso, han dado pie a importantes avances en el desarrollo de sensores basados en ondas acústicas, ya que éstos basan su funcionamiento en el efecto piezoeléctrico para transformar la energía de una señal eléctrica alterna en una onda acústica y viceversa. La piezoelectricidad fue descubierta en 1880 por *Jacques y Pierre Curie*, quienes observaron que cuando se aplicaba una tensión a ciertos cristales como la turmalina, el cuarzo, el topacio y la sal Rochelle², aparecía una carga eléctrica, cuyo voltaje era proporcional a la tensión aplicada. Los Curie, sin embargo, no predijeron el efecto piezoeléctrico inverso, que se dedujo matemáticamente de principios fundamentales termodinámicos por *Gabriel Lippmann* en 1881 [6].

1.1 Generación de ondas acústicas en sustratos piezoeléctricos.

Sabemos que al arrojar una piedra a las aguas tranquilas de un estanque, alrededor del punto donde ha caído la piedra se forman ondas concéntricas que se propagan por toda la superficie del agua hasta llegar a los bordes del estanque. Algo similar ocurre con las ondas acústicas, las partículas no se desplazan con la onda, sino que vibran en torno a su posición de equilibrio (posición de energía mínima), existiendo únicamente desplazamiento de

² De los materiales mencionados, solo el cuarzo se utiliza hoy en día comercialmente, todos los demás cristales piezoeléctricos importantes desde el punto de vista práctico se obtienen de forma artificial.

energía a lo largo del medio, dicho en otras palabras, lo único que viaja a través del material es la perturbación.

Las ondas acústicas, al igual que la mayoría de las ondas conocidas, necesitan un medio elástico para propagarse, ya que en el vacío no existe transmisión sonora. En el caso de las ondas acústicas, pueden ser transmitidas a través de gases, líquidos y sólidos, viajando más rápido en estos últimos [5].

La naturaleza del fenómeno piezoeléctrico hace posible el uso de determinados materiales en sensores de onda acústica, dentro de los materiales más utilizados podemos distinguir dos grupos: los que poseen carácter piezoeléctrico de forma natural como el cuarzo (SiO_2) y los llamados ferroeléctricos³, que presentan propiedades piezoeléctricas tras ser sometidos a una polarización como el niobato de litio (LiNbO_3), el tantalato de litio (LiTaO_3), el óxido de zinc (ZnO) o el nitruro de aluminio (AlN) [5,7].

1.1.1 Efecto piezoeléctrico.

La piezoelectricidad (del griego *piezein*, “estrujar o apretar”) es la capacidad que tienen algunos cristales para generar energía eléctrica al ser sometidos a tensiones mecánicas, es decir, generan tensión eléctrica al ser golpeados o deformados [7].

Los cristales piezoeléctricos al ser sometidos a tensiones mecánicas adquieren una polarización eléctrica en su masa, lo que produce una diferencia de potencial y la aparición de cargas eléctricas en la superficie del mismo. Posteriormente, y dependiendo que es lo que queramos conseguir, esas cargas se aprovechan de una forma u otra. A este fenómeno se le conoce como efecto piezoeléctrico directo, pero este efecto funciona también de forma inversa, ya que al aplicar un campo eléctrico al cristal, este presenta distorsiones mecánicas (efecto piezoeléctrico inverso).

1.2 Sensores basados en ondas acústicas.

Este tipo de sensores forman parte de los llamados “*sensores másicos*” también conocidos como “*sensores gravimétricos*”, este concepto surgió por primera vez en 1959, cuando

³ Cuando se aplica un campo eléctrico externo al material los dipolos eléctricos se alinean en la misma dirección del campo eléctrico aplicado, pero cuando este campo deja de aplicarse no todos los dipolos vuelven a poseer una polarización o carga neta igual a cero (estado de mínima energía), si no que muchos se quedan apuntando en la dirección en la que el campo fue aplicado, por lo tanto los materiales ferroeléctricos presentan una polarización espontánea sin que sea necesario aplicarle un campo eléctrico externo.

Sauerbrey demostró que existe una relación en el cambio de masa en la superficie de una microbalanza de cuarzo (*Quartz Crystal Microbalance*, QCM), con el cambio observado en la frecuencia de oscilación del cristal [5].

Las partes fundamentales que conforman un sensor másico son:

- ❖ **Sustrato.** Este puede ser piezoeléctrico o estar cubierto de una delgada capa de material piezoeléctrico para poder transformar la energía eléctrica a mecánica y viceversa. Por el sustrato viaja la onda con una determinada velocidad de fase (V_p)⁴, la variación de esta velocidad debido a la interacción de la onda con los analitos⁵, es el principio de detección de los sensores gravimétricos [5].
- ❖ **Capa sensible.** Permite que el analito interactúe con la onda acústica y está integrado por un elemento sensible a especies químicas o de elementos biológicos de reconocimiento (inmunosensores) [5].
- ❖ **Transductor.** Electrodo que transforma la detección de analitos en un cambio de frecuencia en la señal de salida. Este cambio es proporcional a la concentración del analito que ha sido absorbido en la capa sensible [5].

Ya que el efecto piezoeléctrico permite que ciertos materiales vibren por la aplicación de un campo eléctrico externo, esto genera ondas acústicas que se propagan e interactúan con el medio que les rodea, de forma que, el grado de interacción o las propiedades del medio pueden ser medidos a partir de las características del campo eléctrico del propio sensor. Estos sensores se comportan como guía de ondas acústicas y pueden responder a la variación de una amplia gama de cantidades físicas como presión, temperatura, masa añadida en la superficie, densidad o viscosidad de los fluidos circundantes [8]. Además, al colocar sobre su superficie un recubrimiento, que actúa como receptor selectivo de determinadas sustancias, permite su amplia utilización como biosensores, en los que suele aprovecharse su sensibilidad a los cambios de masa o de densidad-viscosidad en líquidos.

Puede hacerse una distinción importante entre los tipos de sensores de acuerdo con la naturaleza de las ondas acústicas, clasificándose principalmente en tres grupos: dispositivos basados en *ondas de volumen* (Bulk Acoustic Waves, BAW), dispositivos basados en

⁴ La velocidad de fase es aquella que relaciona directamente la longitud de onda (λ), con la frecuencia síncrona o característica (f_0).

⁵ Es un término utilizado sobre todo en la química analítica, análisis químico, etc., donde hace referencia a una sustancia, la cual puede ser un ion, un elemento, o incluso un compuesto determinado, que posee un interés en nuestra muestra, pues es la parte que deseamos analizar.

ondas acústicas superficiales (Surface Acoustic Waves, SAW) y dispositivos basados en *ondas horizontales transversales* (Acoustic Plate Mode, APM). Los dispositivos BAW son aquellos en los que las ondas se propagan por todo el volumen del material mientras que en los dispositivos SAW las ondas se propagan únicamente en la superficie. Finalmente, los dispositivos APM, en los cuales las ondas se propagan en sustratos con un espesor de unas pocas longitudes de onda, permiten que se propaguen por el volumen con cierto ángulo respecto a la superficie y así pueden reflejarse en ambas caras del material [5,7,8]. La figura 1.1 muestra la clasificación de diferentes BAW, SAW y APM.

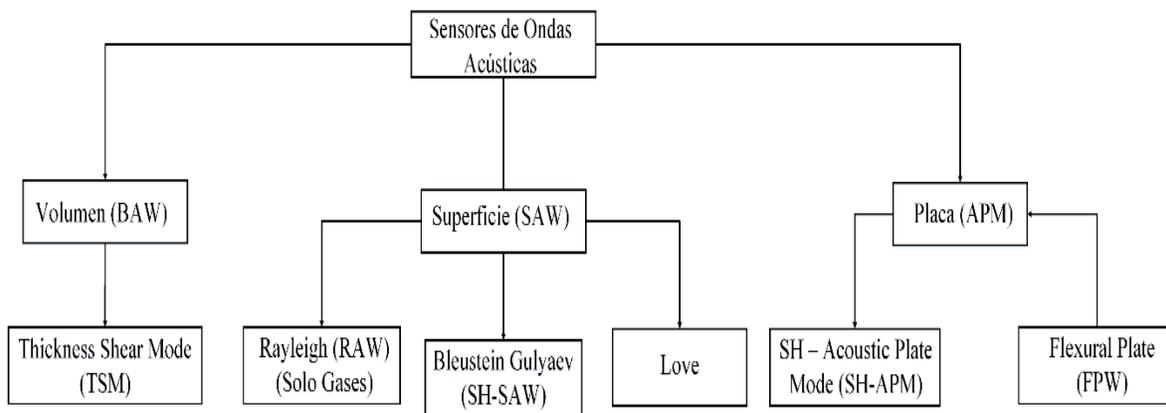


Figura 1.1. Clasificación de los sensores de onda acústica (según la onda propagada).

1.2.1 Ondas de Volumen (Bulk Acoustic Wave, BAW).

La aplicación más antigua del cristal de cuarzo en sensores es la QCM, la cual se ilustra en la figura 1.2. Estos sensores consisten típicamente en una delgada placa de cuarzo con electrodos metálicos⁶ circulares en ambas superficies paralelas principales del cristal. Las ondas de volumen (BAW) se generan aplicando una señal eléctrica de alta frecuencia a los electrodos, dicha diferencia de potencial ocasiona una deformación transversal del cristal y, cuando este potencial es alterno, se excita una onda que se propaga en todo el volumen del sustrato (*Thickness Shear Mode, TSM*) [5,8,9,10].

El efecto que tienen en común toda clase de sensores de ondas acústicas, es la disminución en la frecuencia de resonancia causada por una masa superficial añadida en forma de película, este efecto también llamado “*gravimétrico*” se debe a que la frecuencia

⁶ Para el caso particular de los biosensores se usa oro, plata o platino.

depende exclusivamente de las propiedades físicas intrínsecas del cristal y de su espesor. Al colocar una película de cualquier material lo suficientemente uniforme, rígida y delgada como para que pueda considerarse una pequeña extensión de su espesor, se puede estimar el efecto de cambio en la densidad superficial mediante el desplazamiento de la frecuencia de resonancia.

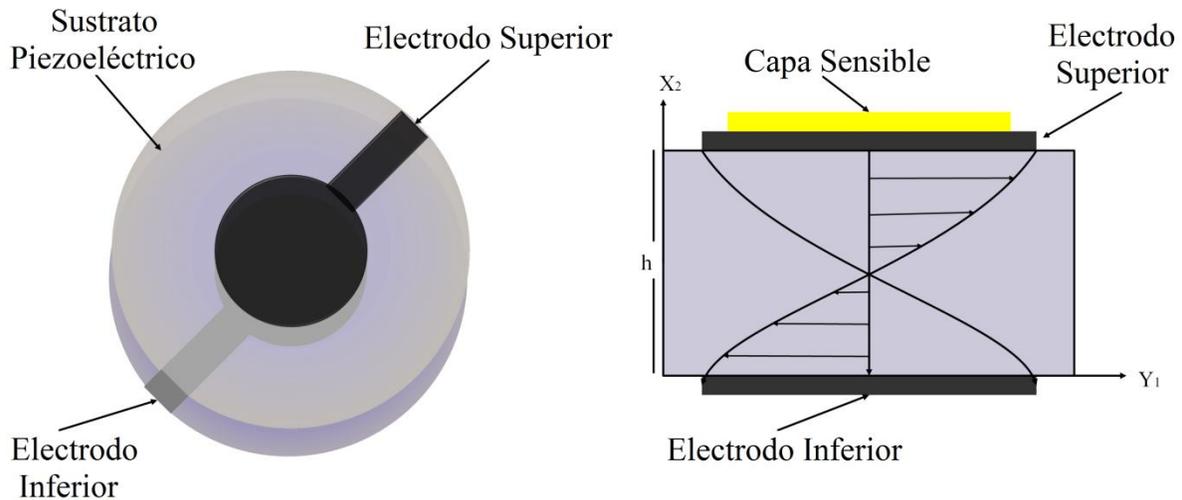


Figura 1.2. Balanza de cuarzo y su modo fundamental de oscilación.

Los QCM probablemente se han convertido en los sensores de onda acústica más comunes pero, debido a que en los únicos lugares en los que se genera la oscilación son en el electrodo de la cara superior y el de la cara inferior, las vibraciones del cristal se concentran en el centro del disco perdiendo amplitud más allá de las dimensiones de los electrodos por lo que para aumentar la sensibilidad de una microbalanza, debe aumentarse su frecuencia de resonancia y para ello es necesario reducir el espesor del cristal, haciendo difícil trabajar con cristales con frecuencias por encima de los 30 MHz debido a que son muy frágiles [5,11].

1.2.2 Ondas Acústicas Superficiales (Surface Acoustic Waves, SAW).

La generación de ondas superficiales se logra mediante una configuración particular de electrodos llamada “*transductor interdigital*” (Inter Digital Transducer, IDT), que tal como lo indica la figura 1.3 consta de dos estructuras idénticas en forma de peine cuyos dedos

respectivos están dispuestos en la superficie de un material piezoeléctrico en un patrón alterno intercalado, el conjunto de estos elementos conforma un dispositivo SAW.

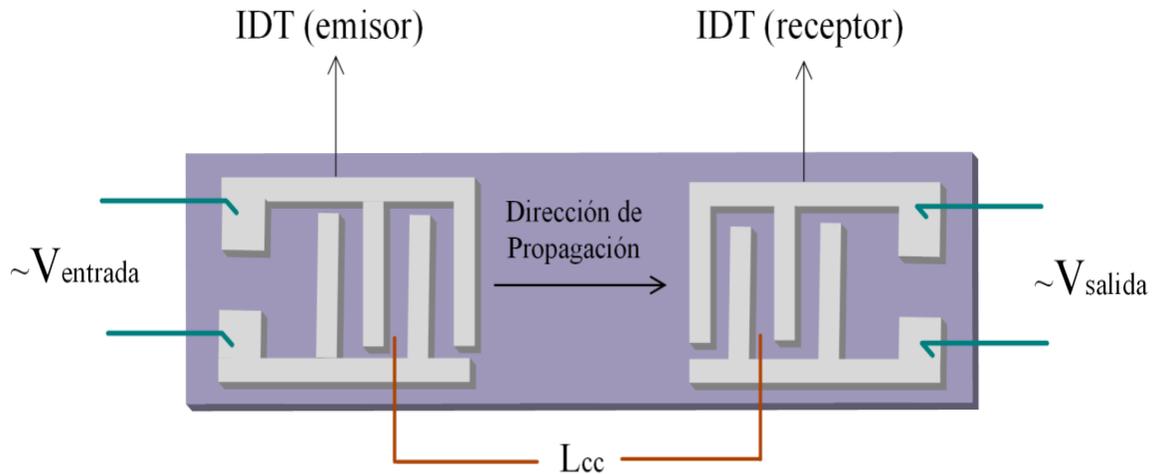


Figura 1.3. Esquema de un sensor de ondas acústicas superficiales.

En principio, los IDT pueden generar un espectro de ondas transversales polarizadas horizontal y verticalmente, así como ondas longitudinales, que se propagan en la superficie o en el volumen del material piezoeléctrico y es por ello que las propiedades del material, el corte del cristal y la geometría del sensor son responsables de qué modos aparecen y en qué medida. En la figura 1.4 podemos observar la generación de diferentes modos de onda [8].

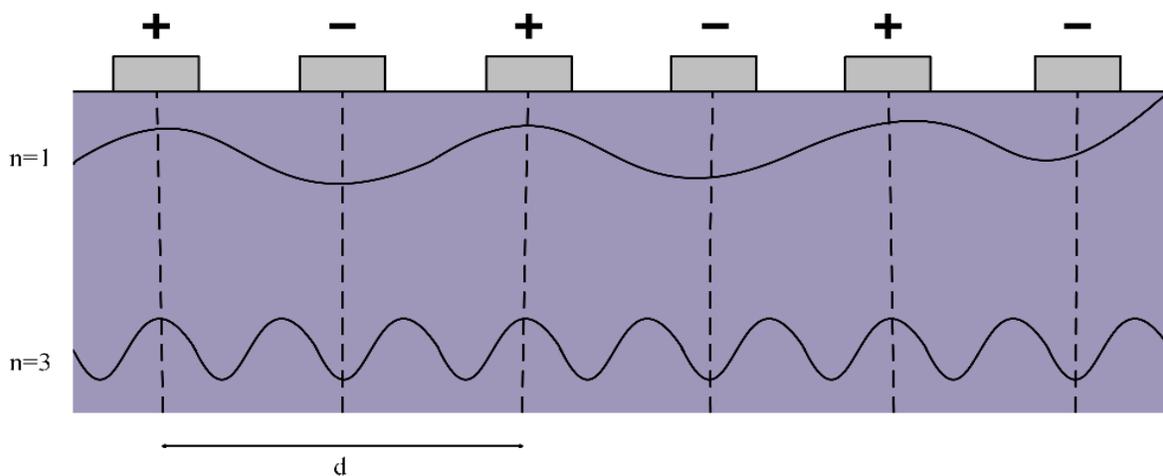


Figura 1.4. Perfil de un IDT con los electrodos polarizados, generando diferentes modos de una onda.

Al colocar dos IDT uno enfrente del otro, el área sensible de nuestro dispositivo será el área comprendida entre los dos electrodos y cuando el primer IDT (emisor) recibe una señal alterna, esta crea una deformación dinámica en el material piezoeléctrico lo que crea una onda acústica de superficie, que viaja a través del sustrato hasta el segundo IDT (receptor) donde es transformada de nuevo en una señal eléctrica alterna [5,8], cuya frecuencia será una de las magnitudes más útiles, debido a que esta magnitud es más sensible que la amplitud a las variaciones en la densidad de masa de la superficie del sensor, y estas pueden ser medidas sin importar lo pequeñas que sean a través del desplazamiento de la frecuencia de resonancia (f_0)⁷ en un determinado tiempo y con un nivel de ruido muy pequeño, con lo que obtenemos una mayor sensibilidad de las SAW en aplicaciones gravimétricas. La velocidad de fase de la onda es característica para la dirección de propagación en un material determinado. Para cada modo de propagación la longitud de onda será:

$$\lambda = \frac{d}{n}, \text{ para } n = 1, 3, \dots \quad (1.1)$$

Donde d es la distancia periódica entre los electrodos con la misma polarización y n representa los modos de la onda. Por lo tanto la frecuencia síncrona o característica la determinará la siguiente relación:

$$V_p = \lambda \cdot f_0 \quad (1.2)$$

La importancia de determinar la magnitud de f_0 , es debido a que de esta depende la sensibilidad másica, la cual es normalmente proporcional a f_0^2 .

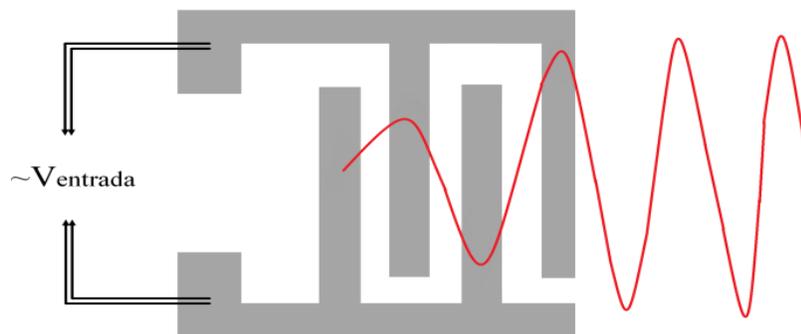


Figura 1.5. Vista superior de un IDT que genera una onda al aplicar una diferencia de potencial alterno entre electrodos.

⁷ Es la frecuencia característica del cristal cuando este alcanza el grado máximo de oscilación.

1.2.3 Ondas de Rayleigh.

En 1885 el científico *John William Strutt* (tercer barón de Rayleigh) predijo la existencia de este tipo de ondas, las cuales se crean por la interacción conjunta de las ondas longitudinales (ondas que desplazan la energía en la misma dirección que la perturbación inicial) y transversales (ondas que desplazan la energía en sentido perpendicular al avance de la onda), estas realizan movimientos elípticos retrógrados al mismo tiempo que avanzan, y como la energía se encuentra confinada en una zona cercana a la superficie, su amplitud decrece con la profundidad [12].

La figura 1.6 ilustra un sensor basado en ondas Rayleigh, y ya que estas ondas necesitan la presencia de ondas transversales y éstas para poder generarse necesitan que el medio por el cual se propagan cuente con cierta rigidez, solo tienen aplicación como sensores de gases, puesto que al ocupar un medio líquido la onda se disipa por completo en distancias de unas pocas longitudes de onda [5,12].

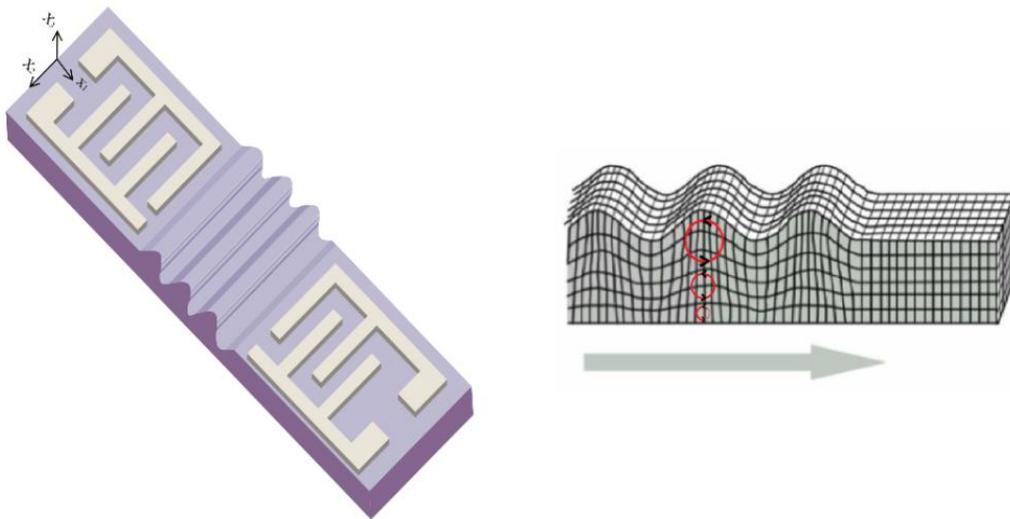


Figura 1.6. Sensor basado en la propagación de ondas Rayleigh.

1.2.4 Ondas Horizontales Transversales SAW (SH-SAW).

Los sensores de ondas transversales de superficie (STW) son dispositivos en los que la energía se encuentra confinada en una capa delgada cerca de la superficie y en ellos son predominantes las ondas transversales⁸, generalmente este tipo de ondas son más lentas que

⁸ También conocidas como ondas secundarias, de corte o cizalla.

las ondas longitudinales pero poseen mayor amplitud y mayor energía. La figura 1.7 muestra el esquema de un sensor SH-SAW.

En el momento en el que una onda se propaga, el desplazamiento de la energía en el material puede realizarse en cualquier dirección perpendicular a la dirección de propagación inicial, pero en algunas ocasiones el desplazamiento de la energía se mueve en una sola dirección, al ocurrir esto se dice que las ondas están polarizadas, debido a esto encontramos ondas transversales polarizadas verticalmente (SV) y ondas transversales polarizadas horizontalmente (SH).

La ventaja de las ondas SH-SAW sobre las Rayleigh, es que al no existir una componente perpendicular al sustrato, la amortiguación de la onda es moderada en líquidos. Por otra parte, la energía que se encuentra confinada en la superficie en las ondas Rayleigh causa una sensibilidad limitada en las SH-SAW [5].

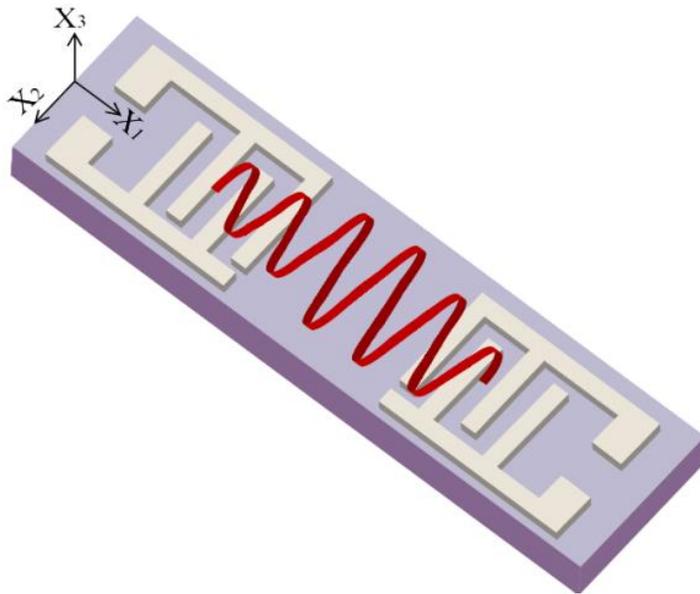


Figura 1.7. Esquema de un dispositivo SH-SAW.

1.2.5 Ondas Love (LW).

Este tipo de ondas, al igual que las Rayleigh, antes de aplicarlas al uso de sensores fueron ampliamente estudiadas en sismología. El efecto físico que provocan las ondas Love fue descubierto originalmente por el matemático *Augustus Edward Rough Love* quien en 1911 desarrolló un modelo matemático de este tipo de ondas sísmicas, al observar que existían

ondas destructivas lejos del epicentro de un terremoto, planteo la posibilidad de que hubiera ondas sísmicas guiadas propagándose como consecuencia de la diferencia de densidad entre la corteza y el manto terrestre [5,10].

En la figura 1.8 observamos que los dispositivos de ondas Love están compuestos por un sustrato que excita principalmente ondas SH-SAW, que posteriormente quedan confinadas en una fina capa guía situada en la parte superior del sustrato y los IDT. Para que las ondas Love se presenten, la velocidad de propagación del material de la capa guía debe ser menor que la del sustrato, esta condición provoca que prácticamente toda la energía de la onda sea confinada en la capa guía, debido a esto, la penetración de la onda en el sustrato es menor y esto hace que los dispositivos LW sean muy sensibles a cualquier cambio que ocurra en la superficie del sensor.

La sensibilidad de estos dispositivos está determinada por el grado de confinamiento de la energía, es decir, que cuanto mayor sea el confinamiento de la energía en la capa guía, mayor es su sensibilidad [10]. Además, debido a que el movimiento de las partículas es paralelo a la superficie del dispositivo, pueden ser usados tanto en líquidos como en gases, y ya que los IDT se encuentran igualmente cubiertos por la capa guía, estos están protegidos del medioambiente y de los líquidos, lo que los hace ideales para trabajar en ambientes hostiles sin que sus IDT se dañen o se destruyan [5].

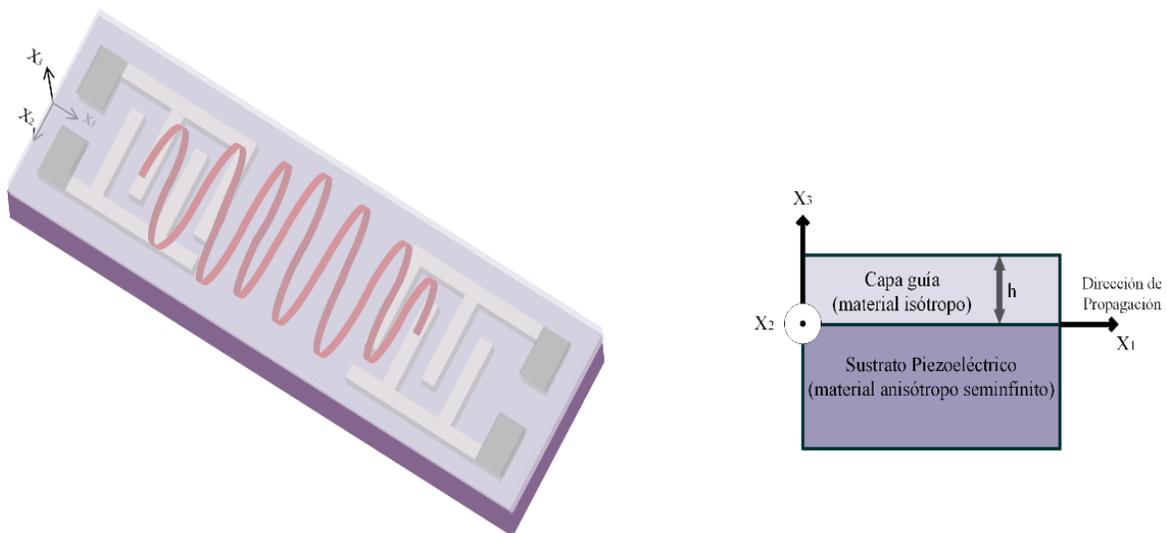


Figura 1.8. Esquema de un dispositivo LW y esquema del perfil de un sensor Love.

1.2.6 Ondas Horizontales Transversales (SH-Acoustic Plate Mode, APM).

Al igual que ocurre en los dispositivos SH-SAW, al generarse ondas de tipo transversales horizontales (SH), el desplazamiento de la energía es predominantemente paralelo a la superficie de la placa y perpendicular a la dirección de propagación a lo largo de la trayectoria de separación entre los IDTs, por lo tanto, estos sensores pueden usarse en aplicaciones con líquidos.

A diferencia de los dispositivos SH-SAW, los SH-APM están constituidos por placas de cuarzo con un espesor de algunas longitudes de onda que actúan como una capa guía, y conforme la onda se propaga por el sustrato, este almacena energía acústica entre las caras superior e inferior de la placa, la figura 1.9 muestra como dicha energía se distribuye por el volumen con cierto ángulo respecto a la superficie y es reflejada en ambas caras, esto implica que la sensibilidad del dispositivo es mayor cuanto más delgado es el sustrato, lo cual representa una desventaja en el momento de su fabricación. Por otra parte al contar con antinodos en ambas superficies del dispositivo, cualquiera de las dos puede ser usada como superficie de detección, lo que facilita su uso en líquidos, ya que los IDTs pueden estar colocados en una cara y detectar los líquidos por la otra y de esta forma los IDTs quedan completamente aislados [5,8].

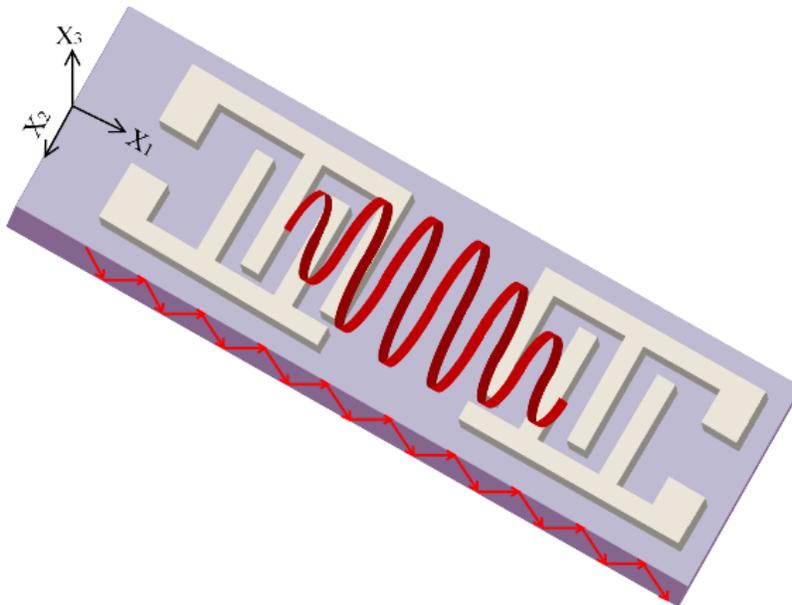


Figura 1.9. Esquema de un dispositivo basado en ondas SH-APM.

1.2.7 Ondas Lamb (Flexural Plate Waves, FPW).

Las *ondas de placa de flexión*, también conocidas como ondas Lamb, fueron descritas por el matemático inglés *Horance Lamb*, los dispositivos basados en ondas FPW están contruidos por placas delgadas que cuentan con un espesor menor que la longitud de onda, debido al confinamiento de la energía acústica en estas placas y a que son afectadas significativamente por las ondas superficiales, se tiene como resultado una alta sensibilidad de masa [5,8]. El esquema de este tipo de sensores se encuentra representado en la figura 1.10.

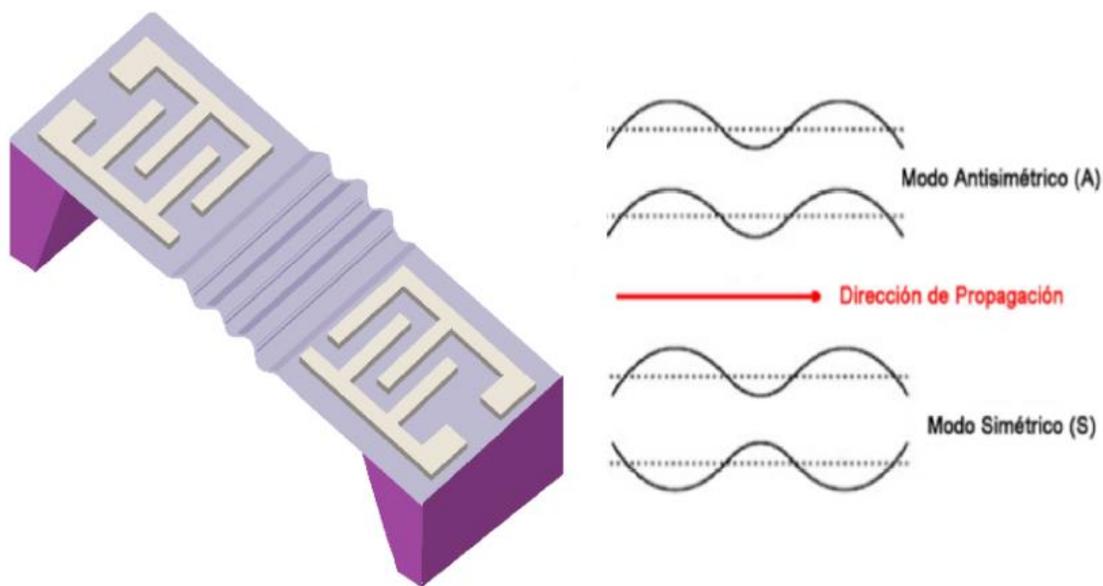


Figura 1.10. Esquema de un dispositivo basado en ondas Lamb y sus diferentes modos de propagación.

Este dispositivo funciona con una onda de placa con desplazamientos de partículas en forma vertical. Las mediciones en gas y en medios líquidos se logran gracias al hecho de que la velocidad de las ondas FPW es menor que la velocidad de compresión del sonido en líquidos, lo que significa que este dispositivo no combina las ondas longitudinales en líquido, solo produce una menor dispersión de energía [10] y al igual que los dispositivos SH-APM, tienen la ventaja de que se puede trabajar con la cara opuesta a la que contiene los IDTs, lo cual evita que estos entren en contacto con los líquidos.

Cuando el grosor del sustrato es menor que la profundidad de penetración, se produce una interacción entre los modos guiados en ambas caras del sustrato y luego se generan una serie de ondas Lamb, las cuales tienen un desplazamiento de partículas similar a las ondas

Rayleigh, es decir, los movimientos de partículas describen una elipse retrógrada tanto normales como paralelas a la superficie. Podemos considerar que una onda Lamb está compuesta por dos ondas Rayleigh que se propagan a cada lado de una placa en modos simétricos o antisimétricos. La ventaja del modo antisimétrico, es que la velocidad del primer modo es mucho más pequeña que las de otros modos posibles y puede disminuir hasta valores menores que la velocidad de propagación del sonido en líquidos y de esta forma se logra que el sustrato funcione como una guía de ondas. La velocidad de la onda depende del espesor de la placa, entre más delgada sea, la velocidad tiende a disminuir y esto es el principal inconveniente de estos sensores puesto que los hace frágiles y además cuentan con una baja reproducibilidad [5,8,10].

1.3 ¿Por qué sensores de ondas acústicas?

En los apartados anteriores se ha hablado sobre los distintos tipos de dispositivos basados en ondas acústicas y como es que gracias a que la mayoría de las interacciones químicas son susceptibles a ser evaluadas y monitoreadas en términos de transferencia de masa a través de una interfaz apropiada, es posible para estos, usar las técnicas gravimétricas para poder realizar la detección de ciertos compuestos, esto permite plantear esta tecnología como una alternativa rentable frente a los métodos de detección existentes, puesto que son capaces de combinar su detección directa, monitoreo en tiempo real, alta sensibilidad y capacidades de selectividad con un costo reducido, y a diferencia de los sensores resistivos, los sensores de ondas acústicas pueden trabajar a temperatura ambiente.

El termino detección directa, se refiere a que la medición se realiza sin que sea necesario purificar la muestra como ocurre en la cromatografía de líquido o gas [10], por lo tanto la cantidad de pasos involucrados en el proceso se reduce, trayendo consigo muchos beneficios, incluyendo ahorros de tiempo y costos significativos.

La portabilidad de estos dispositivos los hace adecuados cuando es requerido un control de la muestra en tiempo real, esto evita que las muestras sean enviadas a los laboratorios reduciendo el tiempo y el costo del análisis de la muestra. Existen dos cosas que pueden mejorar considerablemente la portabilidad y estas son la miniaturización del sensor y la caracterización del mismo. Sumado a todo lo anterior, las magnitudes clave de medición de estos dispositivos proporcionan una ventaja respecto a los demás sistemas de detección,

debido a que la frecuencia, la magnitud y la fase de una señal pueden procesarse de forma sencilla y precisa; es por todo esto que se optó por usar la tecnología de ondas acústicas para desarrollar ésta tesis.

1.4 Comparación de rendimiento.

Como se ha visto en los apartados anteriores, existe una amplia variedad de dispositivos de ondas acústicas y el objetivo principal de llevar a cabo el análisis y la comparación de los diferentes dispositivos AW es encontrar la alternativa que mejor satisfaga el objetivo, en este caso se buscó un dispositivo capaz de mejorar la sensibilidad de masa y el límite de detección. Primero que nada e independientemente de las características únicas que cada sensor posee, estos deben cumplir con requisitos muy estrictos en términos de selectividad y sensibilidad, puesto que deben ser capaces de distinguir compuestos específicos de otros presentes en la muestra (selectividad) y generar una señal de salida lo más intensa posible cuando el compuesto que se busca este presente, incluso en concentraciones muy bajas (sensibilidad) [9,10]. La tabla 1.1 presenta las características de los diferentes dispositivos AW y en la tabla 1.2 se muestran sus ventajas y desventajas.

Elegir un tipo de sensor sobre los otros no es una tarea sencilla, al final la tecnología más apropiada dependerá de diferentes cuestiones, además de la sensibilidad y el límite de detección (LDO); se deben tener en cuenta características como la facilidad de uso y de manejo, la reproducibilidad, la robustez y el bajo costo, puesto que todo desempeña un papel importante en su selección. Existe un factor más que puede marcar una diferencia considerable que es la posibilidad de integrar un gran número de sensores en un dispositivo (matrices de sensores).

Tomando en cuenta todas las características antes mencionadas, al hacer un contraste con los métodos tradicionales (cromatográficos o espectroscopios) y debido a su fácil miniaturización, se llegó a la conclusión de que los sensores LW son los que mejor se ajustan a las necesidades del proyecto. Además, se eligieron porque, al colocarlos en arreglos o matrices, estos sensores ofrecen una capacidad discriminativa cuya magnitud se incrementa en base al número de sensores en el arreglo, es por ello que en la actualidad los arreglos de sensores LW son unos de los de mayor interés en el área del análisis de compuestos volátiles.

Tabla 1.1. Resumen de las características de los dispositivos AW [10].

Dispositivo	Movimiento de partícula	Líquidos	Frecuencia fundamental de operación f_0 (MHz)	Sensibilidad de masa $S\sigma$ (cm²/g)	$S\sigma$ depende de	Materiales de sustrato
QCM	Bulk Shear Horizontal	Si	5-30	12-70	-Espesor del sustrato	AT-Cuarzo
SAW	Surface Vertical and Horizontal	No	40-400	100-200	-Espaciado de los IDTs -Materiales	LT, LN, Cuarzo, Langasite
SH-SAW	Surface Shear Horizontal	Si	30-500	100-180	-Espaciado de los IDTs -Materiales	LT, LN, ST-cuarzo, Langasite, Niobato de Potasio
LW	Surface Shear horizontal	Si	80-300	150-950	-Espaciado de los IDTs -Materiales -Espesor de la capa de guía	ST y AT-cuarzo, LT, LN
SH-APM	Plate Bulk Shear horizontal	Si	25-200	20-50	-Espesor del sustrato -Materiales -Espaciado de los IDTs	ST-cuarzo, LN
FPW	Plate vertical and horizontal	Si	2-20	200-1000	-Espesor del sustrato -Materiales -Espaciado de los IDTs	Si ₃ N ₄ , ZnO y Sustrato de Si

Tabla 1.2. Ventajas y desventajas de los diferentes dispositivo AW [10].

Dispositivo	Ventajas	Desventajas
QCM	<ul style="list-style-type: none"> -Tecnología bien establecida -Comerciante disponible -Robusto -Bajo costo 	<ul style="list-style-type: none"> -$S\sigma$ aumenta cuando el sustrato disminuye y la frecuencia aumenta, pero esto conduce a dispositivos frágiles.
SAW	<ul style="list-style-type: none"> - f_0 más alta 	<ul style="list-style-type: none"> -No es bueno para la detección de líquidos.
SH-SAW	<ul style="list-style-type: none"> -f_0 más alta 	<ul style="list-style-type: none"> -Atenuación bajo inmersión fluida -Menor sensibilidad que en los LW -Corto tiempo de vida.
LW	<ul style="list-style-type: none"> -Aislamiento de IDTs de líquido -Mayor sensibilidad -Robusto 	<ul style="list-style-type: none"> -Baja disponibilidad en el mercado -Alto costo
SH-APM	<ul style="list-style-type: none"> -Los IDTs se pueden ubicar en la parte posterior 	<ul style="list-style-type: none"> -Difícil operar con un solo modo -Baja sensibilidad
FPW	<ul style="list-style-type: none"> -Alto $S\sigma$ a frecuencias más bajas 	<ul style="list-style-type: none"> - $S\sigma$ disminuye considerablemente cuando se trabaja en líquidos. -Alta complejidad de fabricación

Capítulo 2

Adquisición de datos

Como se sabe los sensores LW se basan en la modulación de las ondas acústicas de superficie para detectar un fenómeno físico, el sensor convierte una señal eléctrica de entrada en una onda Love que, a diferencia de una señal eléctrica, puede ser fácilmente influenciada por fenómenos físicos⁹, después el dispositivo transforma esta onda de nuevo en una señal eléctrica, con esto se obtienen cambios de amplitud, fase, frecuencia o retardo entre las señales de entrada y salida, los cuales pueden ser usados para medir la presencia de los compuestos volátiles. Los sensores LW se encuentran conectados a un circuito oscilador, que basa su funcionamiento en líneas de retardo, el diseño básico de una línea de retardo consta de dos IDT depositados en un sustrato de material piezoeléctrico, uno de los cuales será la entrada del dispositivo y el otro la salida. Cuando se aplica un voltaje de entrada se genera un campo eléctrico entre los dientes o electrodos del primer IDT, dicho campo produce en el sustrato una deformación, la cual se propaga a lo largo del mismo en forma de onda Love, cuando esta onda viaja a través del sustrato, crea deformaciones en este, las cuales a su vez, crean campos eléctricos, debido a esto cuando la onda alcanza el segundo IDT los electrodos de este se polarizan, por lo cual se obtiene un voltaje a la salida. En este caso la señal de salida será igual a la señal de entrada pero retardada en un factor que dependerá de la velocidad de la onda a través del sustrato.

La respuesta que se obtiene del dispositivo LW está íntimamente relacionada con la geometría de los IDT que la componen. De igual forma, esta geometría también se utiliza para eliminar o minimizar problemas que comprometan la eficiencia del oscilador, como por ejemplo, las ondas reflejadas indeseadas. Estas ondas se producen cuando la onda Love original alcanza el segundo IDT; parte de esta onda se refleja en dirección al primer IDT y nuevamente se refleja en este último en dirección al segundo, mezclándose con la señal de salida, lo cual produce ruido en la respuesta [13].

Un método que se utiliza para contrarrestar el inconveniente anterior, es utilizar transductores de doble electrodo. En la figura 2.1 se observa que dichos transductores

⁹ Como por ejemplo temperatura, presión, fuerzas de inercia, sustancias químicas, campos magnéticos, radiación, etc.

tienen electrodos pares en el mismo peine, lo que significa que, la distancia entre electrodos consecutivos será un cuarto de λ , lo que provoca que las ondas reflejadas de los electrodos sucesivos difieran en fase por 180° y por ende sean canceladas, debido a esto los transductores de doble electrodo son empleados con mayor frecuencia.

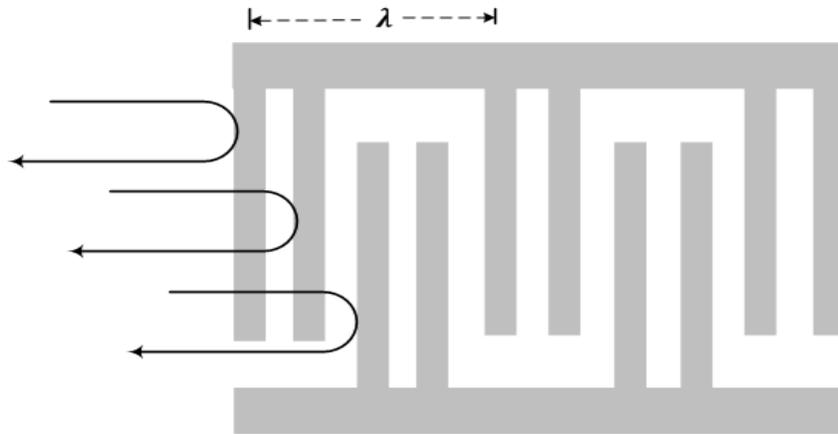


Figura 2.1. Reflexión interna de las ondas superficiales en un transductor de electrodos dobles.

Una línea de retardo puede ser utilizada como un filtro de compresión de pulso, ya que el espaciamiento de los electrodos del segundo IDT tiene una periodicidad graduada, de manera que dos electrodos consecutivos responderán solo a una onda cuya longitud sea dos veces la separación entre ellos; por lo tanto, el retardo en la señal de salida variará con la frecuencia [13].

Tal como lo indica la figura 2.2, el circuito oscilador consta principalmente de un amplificador, un acoplador y un filtro. En este dispositivo, cada etapa de amplificación está precedida por un atenuador para evitar acoplamientos, una etapa de filtrado (pasa banda) ya que en algunas líneas de retardo existen varios modos de propagación y por último, un acoplador que nos permite dividir la señal en dos, una que realimenta el circuito y la otra como entrada al sistema de adquisición de datos [5].

Este circuito debe cumplir con las condiciones de ganancia unitaria en lazo cerrado lo que sucederá a una frecuencia determinada f_0 (frecuencia de oscilación), también son conocidas como condiciones de Barkhausen, estas son:

$$G_x = 0^+ dB \tag{2.1}$$

$$G_{línea_de_retardo} + G_{amplificador} + G_{otros_elementos} = 0^+ dB \tag{2.2}$$

$$2\pi f_0 \frac{L_{cc}}{V_p} + 2\Phi_{IDT} + \Phi_E = 2\pi n \quad (2.3)$$

Donde G_x es la ganancia de cada elemento, $2\pi f_0 \frac{L_{cc}}{V_p}$ es la fase introducida por la línea de retardo, Φ_{IDT} y Φ_E son la diferencias de fase introducidas por los IDTs y el dispositivo amplificador, que son mucho menores que la línea de retardo [5].

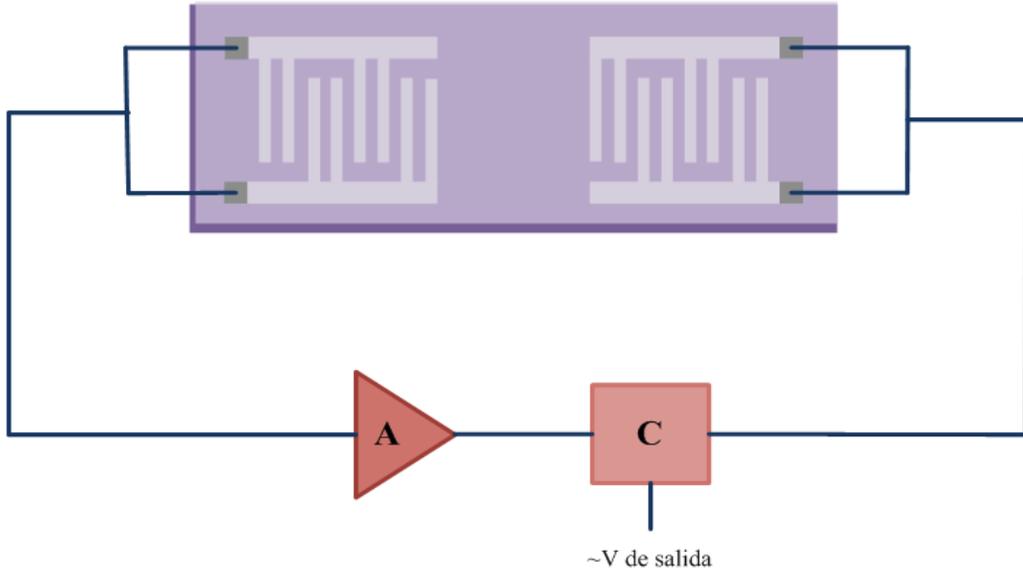


Figura 2.2. Configuración de un oscilador.

Puesto que la señal que se obtiene a la salida del acoplador es una señal analógica y esta no puede ser adquirida por una computadora, es necesario introducir una etapa de procesamiento de señales, esta etapa es muy importante en la construcción de un sistema de adquisición de datos (DAQ), debido a que este permite representar, transformar y manipular la información contenida en las señales. La condición para que una señal pueda ser procesada en una computadora digital, es que la amplitud de la señal debe ser manejada en forma discreta, lo que implica que la amplitud de la señal sólo puede tener un número finito de valores diferentes, por lo cual es necesario realizar un procesamiento digital de señales (*Digital Signal Processing, DSP*), en el que además de la amplitud, las señales son discretas en el tiempo o en el espacio, lo que nos permite conocer el valor de la señal en instantes o puntos específicos, para ello se necesita una conversión analógica-digital previa y una decodificación posterior en el momento de la recepción, debido a esto existe una

pérdida de información al convertirla de continua a discreta y por insignificante que esta resulte ser, siempre hay un error de cuantificación que impide que la señal digital sea exactamente igual a la señal original; sin embargo, al trabajar con una señal discreta (digital) la información se puede transmitir, almacenar y manipular más fácilmente y es menos sensible a las interferencias.

2.1 Sistema de adquisición de datos (DAQ).

El sistema de adquisición de datos está constituido por cuatro elementos principales: sensores, hardware, software y PC; y de las características y funcionalidad de cada uno de ellos dependerá en gran medida la eficiencia del sistema [14]. Es verdad que elegir correctamente tanto los sensores como el hardware para el acondicionamiento y la conversión de señales es clave para conseguir una lectura correcta de las mismas; pero de igual forma se debe considerar que el software que utilizado para desarrollar la aplicación, debe ser lo suficientemente capaz de analizar y presentar correctamente la información al usuario, y de ser necesario, dar cierta flexibilidad para que en caso de que las necesidades de la aplicación cambien, el software pueda adaptarse a dichos cambios.

En general, el sistema de adquisición interpreta órdenes recibidas, las ejecuta y procesa los datos para que estos sean recibidos por la PC, debido a todo este proceso es necesario otorgar al sistema cierta capacidad de cómputo y procesamiento, en este caso se emplea una computadora de placa única (*Single Board Computer*, SBC) con el objetivo de proporcionarle al sistema una alta capacidad de procesamiento en tiempo real con lo que la PC solamente actúa como una interfaz entre el usuario y el sistema de adquisición de datos o cuando mucho se encarga de realizar operaciones con alto grado de abstracción.

Ya que la SBC solo tiene que ocuparse de las entradas y salidas que interactúan con la matriz de sensores y no tiene que atender otras tareas del sistema operativo como lo haría una computadora de propósito general, es posible disminuir la potencia exigida a la computadora [15], de forma que la mayoría de las computadoras del mercado contarán con los suficientes recursos para poder gobernar el sistema de adquisición de datos, dependiendo únicamente de la interfaz con la que se quiera dotar al sistema.

2.2 Hardware para adquisición de datos.

El hardware es una pieza fundamental dentro del sistema de adquisición de datos, puesto que es el encargado de transferir y procesar todos los datos adquiridos de forma continua. Para poder realizar esta tarea eficazmente se ha seleccionado una SBC ya que, a diferencia de las tarjetas de desarrollo que utilizan un microcontrolador, estas por lo general tienen incorporados procesadores de tipo ARM (*Advanced RISC¹⁰ Machine*), lo cual les otorga capacidades de procesamiento similares a los de una computadora [16]. Debido a la gran cantidad y diversidad de aplicaciones que usan las SBC's para su desarrollo, en la actualidad se puede encontrar una extensa gama de ellas en el mercado. Se eligió trabajar con una tarjeta que, además de cumplir con las características de rendimiento de una PC, también cuenta con la compatibilidad de hardware que poseen la mayoría de los microcontroladores modernos, y dado que la *Beaglebone Black Rev C* (BBB) tiene una amplia variedad de características técnicas, se dispone de diferentes herramientas en un solo dispositivo.

En adición a la Beaglebone fue necesario utilizar un dispositivo RTL-SDR (*Software Defined Radio – Radio Definida por Software*), que tiene como función recibir la frecuencia proveniente de los sensores y convertir dicha señal analógica en una señal discreta para su posterior análisis. Se eligió trabajar con este dispositivo debido a que posee una alta compatibilidad con la BBB, ya que es viable usarlo tanto en *Windows* como *Linux*.

2.2.1 Características de la Beaglebone Black Rev C.

Como se mencionó anteriormente, la Beaglebone Black, mostrada en la figura 2.3, es una computadora de placa única usada como plataforma de desarrollo de bajo costo, que cuenta con un microprocesador Sitara AM3358BZCZ100 ARM Cortex-A8 de 32bits de Texas Instruments, que funciona a 1GHz y es capaz de procesar 2000 millones de instrucciones por segundo, posee 512Mb de RAM y 4Gb de memoria incorporada, que puede ampliarse añadiendo una tarjeta microSD. Debido a la gran cantidad de puertos y periféricos integrados en la BBB se cuenta con diversas opciones que permiten la comunicación con otros dispositivos. Dicha comunicación puede establecerse mediante el host USB, el puerto Ethernet, el puerto HDMI (*High Definition Media Interface - Interfaz Multimedia de Alta*

¹⁰ RISC (*Reduced Instruction Set Computer* – Ordenador con Conjunto Reducido de Instrucciones).

Definición) o por el puerto *Serial Debug* de seis pines, en este último es posible tener comunicación directa con el procesador por medio de uno de los módulos UART (*Asynchronous Receiver/Transmitter – Transmisor/Receptor Asíncrono Universal*), sin embargo, se considera como un puerto de emergencia ya que mientras la BBB no presente ningún problema es mejor establecer comunicación por otros medios.

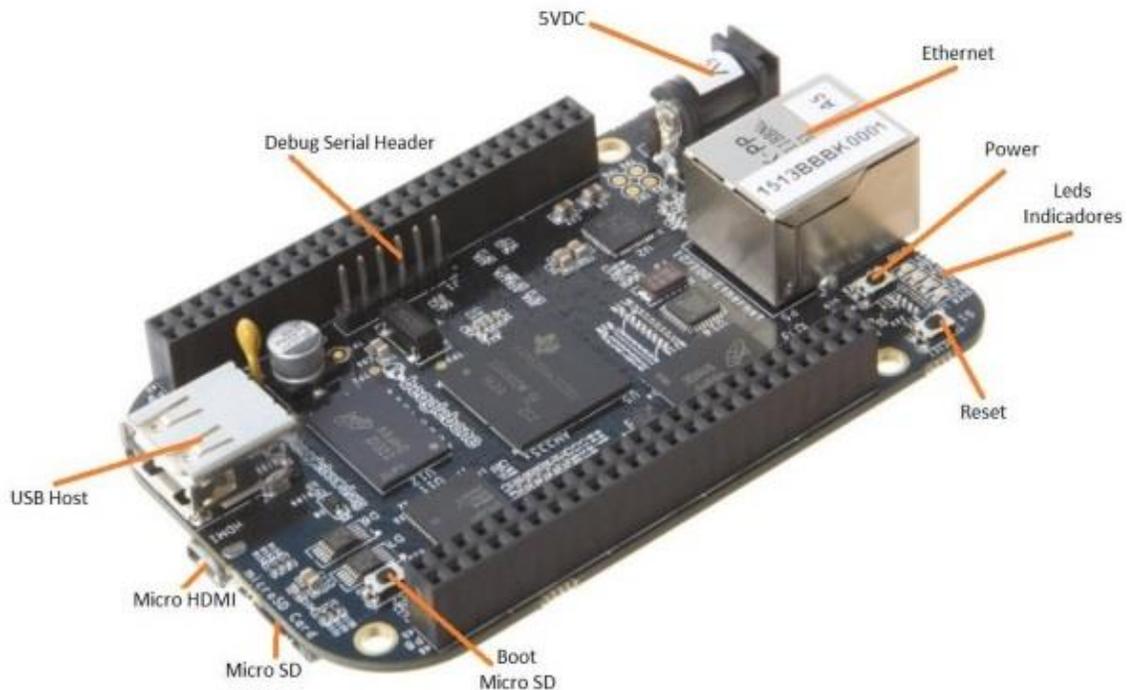


Figura 2.3. Beaglebone Black REV-C y sus elementos.

Los GPIO's (*General Purpose Input/Output – Entrada/Salida de Propósito General*) son otros elementos sumamente útiles, puesto que gracias a ellos podemos generar señales PWM (*Pulse-Width Modulation – Modulación de Ancho de Pulsos*) en determinados pines, así mismo disponer de entradas y salidas analógicas, temporizadores y buses sumamente útiles como I2C (*Inter Integrated Circuits – Circuitos Integrados Internos*), UART, SPI (*Serial Peripheral Interface – Interfaz Periférica Serial*), GPMC (*General-Purpose Memory Controller – Controlador de Memoria de Propósito General*), entre otros. Al hacer uso de los GPIO's, es muy importante tener en cuenta que algunos de los pines de la BBB están multiplexados, lo que significa que no es factible hacer uso simultaneo de los buses y algunos elementos de la tarjeta. Todo lo anterior se encuentra colocado en una

placa que no supera las dimensiones de una tarjeta de crédito, lo que la hace ideal para disminuir la robustez de un sistema.

Otra de las ventajas que proporciona esta tarjeta, es que es capaz de soportar distintos sistemas operativos como por ejemplo: Debian, Angstrom, Ubuntu, ArchLinux, Android, entre otros. Para la realización de la presente tesis se utilizó el sistema operativo preestablecido por el fabricante, que es una versión de *Debian* diseñada especialmente para procesadores ARM y personalizada para la Beaglebone, ya que es el sistema que ofrece mayor estabilidad entre la tarjeta y sus periféricos, permitiendo así la utilización de todos los elementos incorporados dentro de ella y, al ser un hardware de código abierto, es posible ejecutar software de código abierto, y con ello emplear diferentes lenguajes de programación como son: C++, Python, Java, entre otros.

2.2.2 La Beaglebone Black como hardware de adquisición de datos.

Existen diversas razones por las que se eligió trabajar con la BBB, entre ellas: por su velocidad de procesamiento, por ser la computadora de placa reducida con mayor número de puertos de entrada/salida (E/S), porque es posible establecer conexión con una red local o externa, lo que le permite comunicarse con los dispositivos que se encuentren conectados a la misma red y de esta forma controlar diversas funciones de la tarjeta, gracias a su tamaño el sistema se transporta con relativa facilidad y, ya que es un hardware de bajo costo, se pueden realizar distintas aplicaciones sin necesidad de alterar el presupuesto del proyecto.

Aunque todas las características anteriores son igualmente importantes, existen dos motivos principales que hicieron concluir que era adecuado usar este dispositivo como la base del sistema de adquisición de datos:

- 1) La BBB integra al resto de sus componentes una unidad programable en tiempo real conocida como PRU-ICSS (*Programmable Real-Time Unit and Industrial Communication Subsystem – Unidad Programable de Tiempo Real y Subsistema de Comunicación Industrial*), que como su nombre lo indica, permite el desarrollo de aplicaciones que requieren procesamiento en tiempo real.
- 2) Permite ejecutar aplicaciones independientes de otros entornos de desarrollo.

2.2.3 RTL-SDR.

El RTL-SDR ilustrado en la figura 2.4, es un dispositivo SDR de bajo costo, cuyos componentes internos se muestran en la figura 2.5, está basado en el chip RTL2832U de la familia RTL de *Realtek* (chip demodulador originalmente creado para decodificar una señal de emisión HDTV, *High Definition Television – Televisión de Alta Definición*) que puede recibir frecuencias entre los 22-2200 MHz, las cuales varían dependiendo del sintonizador con el que este acompañado. La tabla 2.1 muestra los diferentes tipos de sintonizadores.

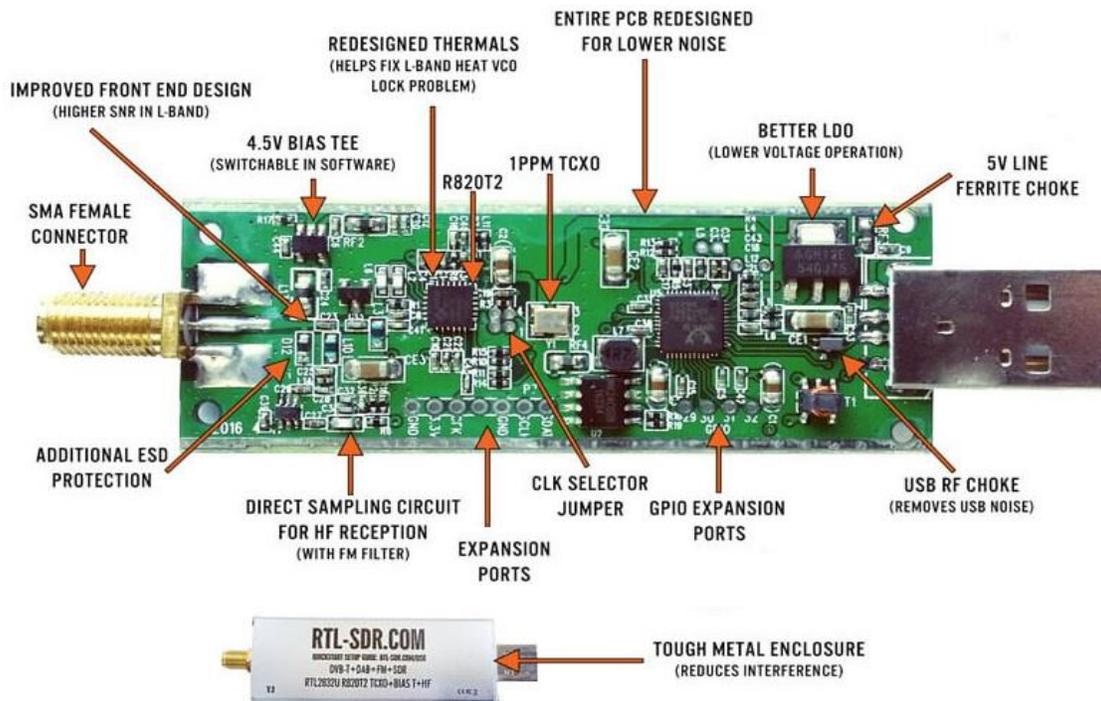


Figura 2.4. Dispositivo RTL-SDR y sus componentes internos [17].

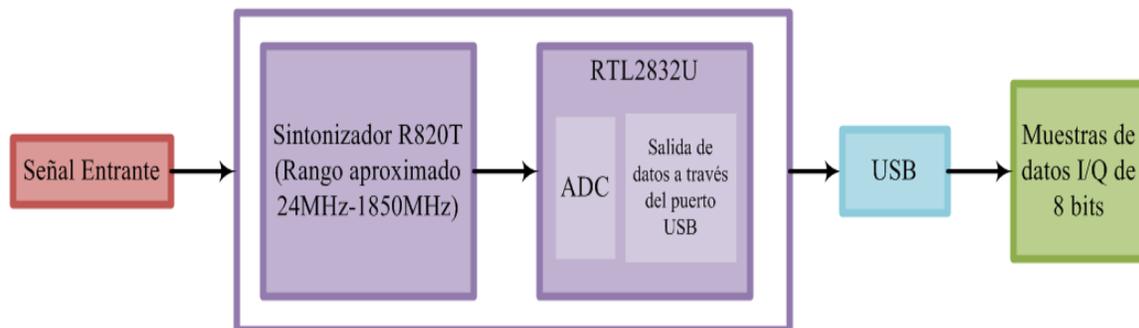


Figura 2.5. Diagrama del funcionamiento del RTL-SDR.

Tabla 2.1. Tipos de sintonizadores y sus frecuencias [20].

Sintonizador	Rango de Frecuencia
Elonics E400	52 - 2200 MHz con un espacio de 1100 MHz a 1250 MHz
Rafael Micro R820T/2	24 - 1766 MHz
Fitipower FC0013	22 – 1100 MHz
Fitipower FC0012	22 – 948.6 MHz
FCI FC2580	146 - 308 MHz y 438 - 924 MHz (espacio intermedio)

La principal característica de este dispositivo es reducir la complejidad del hardware analógico (filtros, amplificadores, moduladores, etc.) emulándolo a través de un programa informático que puede ser implementado en una computadora o en un sistema embebido, de ahí el nombre de “Radio Definida por Software”. Este módulo cuenta con un ancho de banda máximo de 3.2 MHz (estabilidad aproximada a 2.8 MHz), un *convertidor analógico digital (Analog to Digital Converter, ADC)* de 8 bits con un rango dinámico aproximado de 50 dB y una impedancia de entrada de 75 Ohm [18].

El sintonizador tiene integrado un mezclador¹¹, que es utilizado tanto en transmisores como en receptores de señales. En la figura 2.6b se muestran los tres puertos que constituyen el mezclador, estos son: puerto de entrada de radio frecuencia (RF), puerto de entrada del oscilador local (LO) y puerto de salida de frecuencia intermedia (IF). El propósito del mezclador es cambiar la banda de frecuencia en la que se encuentra una señal, preservando las otras características (fase y amplitud) de la señal inicial.

Por el puerto RF se capta una frecuencia lo más parecida posible a la que se quiere detectar, en el puerto LO se asigna una frecuencia central establecida por el usuario, por medio del proceso de *conversión descendente* que se ilustra en la figura 2.6a (“downconverter”), ambas señales se mezclan resultando en el valor absoluto de la diferencia de ambas, dando como salida una señal de frecuencia intermedia a través del puerto IF [19].

¹¹ Si el mezclador es parte de un receptor, tendrá funciones de convertidor reductor de frecuencia y si forma parte de un transmisor actuará como un convertidor elevador de frecuencia.

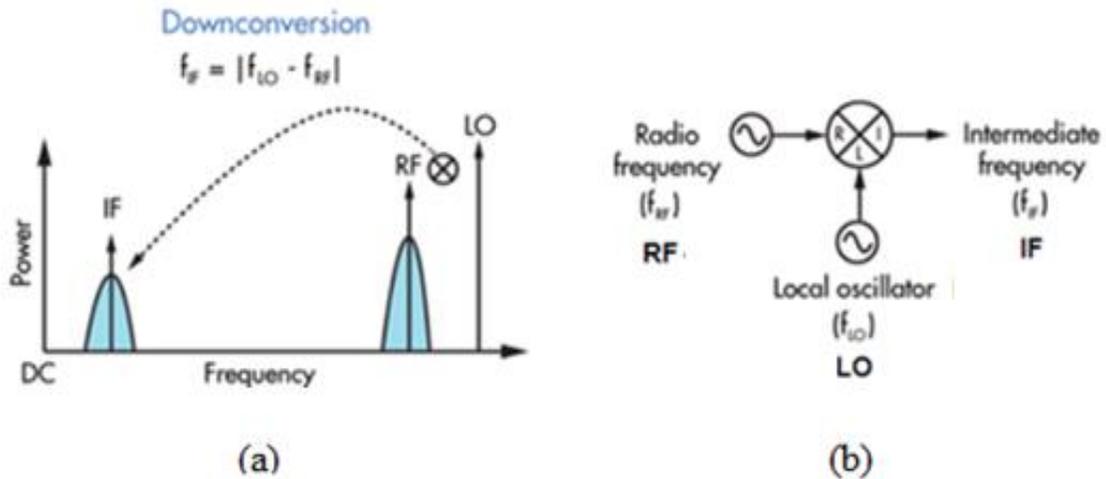


Figura 2.6. (a) Representación del proceso downconverter. (b) Diagrama de los puertos del mezclador incluido en el sintonizador [19].

Una vez obtenida la frecuencia intermedia, el ADC del chip¹² se encarga de digitalizar dicha frecuencia en una cadena de valores que indican la amplitud y fase (Información I/Q¹³) de 8 bits, la secuencia compleja de salida muestra valores alternos de parte real (amplitud) seguido por una parte imaginaria (desplazamiento):

$$X_{1real}, X_{1imaginaria}, X_{2real}, X_{2imaginaria}, \dots, X_{nreal}, X_{nimaginaria}$$

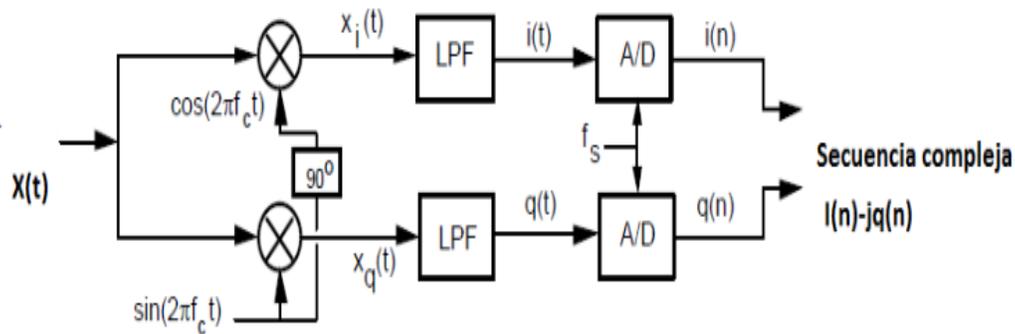


Figura 2.7. Bloques del muestreo por cuadratura.

La digitalización se logra con el proceso denominado muestreo de cuadratura, el cual se encuentra representado por un diagrama de bloques en la figura 2.7, en dicho proceso la señal analógica se transforma en dos señales iguales desfasadas 90° , a la señal que no está

¹² El chip RTL2832U tiene una etapa de procesamiento digital, incluyendo filtrado y disminución de la tasa de muestreo de la señal entregada por el mezclador.

¹³ Inphase/Quadrature - Fase/Cuadratura

desplazada se le denomina como señal en fase (*Inphase signal*) o señal I, mientras que la señal desfasada es nombrada como señal en cuadratura (*Quadrature Signal*) o señal Q. Las frecuencias que conforman las señales I y Q que estén por encima o por debajo de la frecuencia del oscilador local se filtran, como resultado a la salida se obtiene la secuencia compleja $I(n) - jQ(n)$.

Si se usa la frecuencia de muestreo máxima de 3.2 MHz por cada medida, al realizar la digitalización, idealmente se toman 3.2 millones de datos por segundo, datos que son procesados para obtener la frecuencia de resonancia, no obstante, el tiempo de exposición del sensor al compuesto no es solo de un segundo, sino que es de 1 minuto, tiempo en el cual se adquiere una gran cantidad de datos. Después del minuto de exposición, es necesario que los sensores entren en una etapa de recuperación que consta de 10 minutos, en dicha etapa los sensores son expuestos a un gas que contiene carbón activo, el cual ayudará a limpiar las impurezas acumuladas en la capa guía.

Los datos digitalizados son enviados al host a través del puerto USB, idealmente no debería existir pérdida de información durante la transferencia, no obstante en algunos casos ocurre y esto puede pasar cuando la tasa de digitalización del RTL-SDR resulta ser mayor a la tasa de transferencia del puerto USB del host.

La tasa de digitalización es la velocidad con la que los datos son muestreados y se calcula de la siguiente forma:

$$DR = CR_{RTL-SDR} \times f_s \quad (2.4)$$

Dónde $CR_{RTL-SDR}$ es la velocidad de conversión del ADC el cual es un valor fijo equivalente a 8bits por muestra y f_s es la frecuencia de muestreo a usar. Si usáramos la máxima frecuencia de muestreo, al sustituirla en la ecuación 2.11, el valor de la tasa de transferencia sería:

$$DR = 8 \frac{bit}{sample} \times 3.2 \frac{Msamples}{seg} = 25.6 \frac{Mbits}{seg}$$

Dado que 25.6Mbits/seg es una velocidad más pequeña que la velocidad teórica máxima de un puerto USB 2.0 (la cual equivale a 480Mbits/seg), hipotéticamente no debería existir pérdida alguna. Sin embargo, en [18] se menciona que al usar la frecuencia máxima hay pérdidas de información debido a que dicha frecuencia no es lo suficientemente estable, por

lo cual se usa una frecuencia de muestreo de 2.4MHz. Pero si se conecta el RTL-SDR a un puerto USB 3.0 es posible usar frecuencias de muestreo de 2.8 MHz y 3.2 MHz.

Una ventaja que presenta el RTL-SDR es poder modificar el ancho de banda o frecuencia de muestreo a través del software. Sí bien el ancho de banda (espectro donde se concentra la mayor potencia de la señal) y la frecuencia de muestreo (número de muestras en un tiempo) no son lo mismo, para efectos de configuración del RTL-SDR se encuentran relacionados. Por ejemplo, si se tiene una frecuencia de muestreo de 3.2Mpsps¹⁴ se obtiene un ancho de banda (B) de 1.6 MHz, esto indica que la forma de onda debe ser muestreada lo suficientemente rápido para producir un registro de tiempo digitalizado que represente fielmente la forma de onda original, esto se asienta en el criterio de Nyquist el cual establece que una señal debe ser muestreada a una velocidad mayor que el doble de la frecuencia más alta presente en la señal. La velocidad de muestreo mínima aceptable se denomina velocidad o frecuencia de Nyquist:

$$f_s > 2B \quad (2.5)$$

Regresando al ejemplo anterior, al tener una frecuencia de muestreo de 3.2MHz y al despejar la ecuación 2.12 se obtiene la frecuencia de Nyquist:

$$3.2MHz = 2B$$
$$B = \frac{3.2MHz}{2} = 1.6MHz$$

Esta frecuencia representa el ancho de banda en el cual las señales digitalizadas en el dominio del tiempo serán convertidas al dominio de la frecuencia, es decir, que solo las frecuencias que se encuentren dentro del espectro limitado por el ancho de banda hasta la frecuencia central serán transformadas.

Las series de Fourier nos permiten obtener el espectro de frecuencia de señales continuas y periódicas y la transformada de Fourier nos permite obtener el espectro de frecuencias de señales aperiódicas. Al aplicar estas operaciones a una señal podemos ver como distribuyen las diferentes frecuencias que la componen y esto a su vez permitirá manipular la señal de entrada de manera más sencilla, pues su distribución frecuencial es una función discreta y no continua.

¹⁴ Mega muestras por segundo

Las señales son representadas matemáticamente como funciones de una o más variables independientes, siendo el tiempo la variable independiente más común, pero en este caso es conveniente describir o representar las señales en el dominio de la frecuencia. Debido a que las series y la transformada de Fourier nos permiten ver la distribución de frecuencias de la señal y almacenar los valores de frecuencias para cada tiempo determinado de forma digital, son dos herramientas muy útiles en el procesamiento digital [20].

Las *series de Fourier* son series infinitas que convergen puntualmente a una función periódica y continua por partes, dicho de otra forma, estas series son una herramienta matemática básica para analizar funciones periódicas a través de la descomposición de dicha función en una suma infinita de funciones sinusoidales¹⁵ mucho más simples [20,21].

Las series de Fourier tienen la forma:

$$f(t) \sim \frac{a_0}{2} + \sum_{n=1}^{\infty} \left[a_n \cos \frac{2n\pi}{T} t + b_n \sin \frac{2n\pi}{T} t \right] \quad (2.6)$$

Donde a_0 , a_n y b_n son los coeficientes de Fourier que toman los valores:

$$a_0 = \frac{2}{T} \int_{-T/2}^{T/2} f(t) dt, \quad (2.7)$$

$$a_n = \frac{2}{T} \int_{-T/2}^{T/2} f(t) \cos \left(\frac{2n\pi}{T} t \right) dt, \quad (2.8)$$

$$b_n = \frac{2}{T} \int_{-T/2}^{T/2} f(t) \sin \left(\frac{2n\pi}{T} t \right) dt, \quad (2.9)$$

Por la identidad de Euler, las formulas anteriores pueden expresarse en su forma compleja:

$$f(t) \sim \sum_{n=-\infty}^{\infty} C_n e^{2\pi i \frac{n}{T} t} \quad (2.10)$$

En el caso de tener funciones no periódicas que cumplan ser continuas a tramos en todo intervalo finito, y que tengan derivada a derecha e izquierda en todo punto y que sean absolutamente integrables se pueden representar mediante la integral de Fourier¹⁶:

¹⁵ Como combinación de senos y cosenos con frecuencias enteras.

¹⁶ Se muestra la forma compleja ya que es más reducida.

$$f(t) \sim \int_{-\infty}^{+\infty} F(\omega) e^{i\omega t} dt \quad (2.11)$$

Donde:

$$F(\omega) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} f(t) e^{-i\omega t} dt \quad (2.12)$$

$F(\omega)$ es la *transformada de Fourier* de $f(t)$, e implica que para cada punto en el espacio de t le hace corresponder un punto en el espacio de frecuencias de ω

Cuando los componentes de la señal se han transformado al dominio de la frecuencia, se toma el valor de la frecuencia de la primera armónica, está frecuencia nos permitirá observar el comportamiento de los sensores cuando se exponen a los diferentes compuestos volátiles.

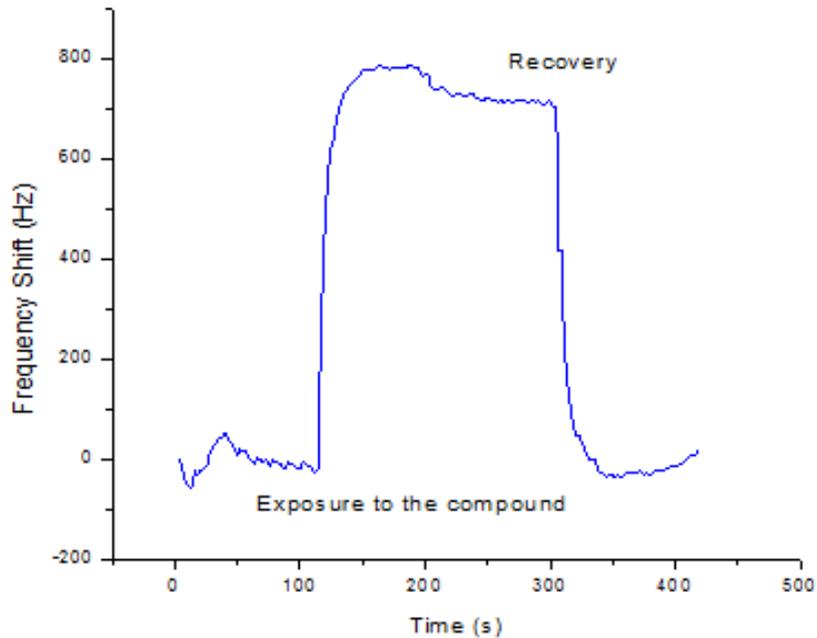


Figura 2.8. Comportamiento del sensor.

En la figura 2.8 se puede notar que existe un incremento en la frecuencia fundamental cuando el sensor es expuesto al compuesto volátil, una vez que la exposición se detiene el sensor necesita un tiempo de recuperación, es decir, al sensor se le aplica aire (limpio) para que se limpie la capa guía de impurezas provenientes del compuesto volátil que puedan afectar a la medición, al concluir el tiempo de recuperación la frecuencia vuelve a su estado inicial para una nueva exposición.

El dato se necesita para poder saber a qué valor de frecuencia pertenece el compuesto volátil es la *frecuencia de cambio*, esta frecuencia es la diferencia entre el inicio de la exposición del compuesto al sensor (f_1) y el final del tiempo de recuperación (f_2).

$$\Delta f = f_2 - f_1 \quad (2.13)$$

Una vez calculadas las frecuencias de cambio, los valores se guardan en columnas, cada columna representa al sensor que realizó la medida, esto permite saber que tan sensible es el sensor a cierto compuesto volátil.

Capítulo 3

Procesamiento de datos.

Una vez obtenidas las señales procedentes de la matriz de sensores, es necesario implementar técnicas de procesamiento de datos, para que posteriormente, estos sean interpretados de acuerdo con la aplicación que se desea evaluar. En este caso se usan técnicas de reconocimiento de patrones que, correctamente aplicadas aseguran que las señales de los sensores serán analizadas apropiadamente con el propósito de obtener modelos confiables para la predicción, identificación o clasificación de determinadas muestras. Sin embargo, en muchas ocasiones el vector de características que resulta de la etapa de adquisición de datos no es adecuado para ser procesado, puesto que cuenta con una gran cantidad de información y cada una de las muestras, características o variables que se introducen para que sean procesadas, pueden incrementar el tiempo en que el sistema tarda en realizar el análisis de los datos. De igual forma, se debe considerar el espacio adicional de almacenamiento que supone guardar los valores de las nuevas variables, muchas de las cuales son redundantes con respecto a otras, así que no aportan información adicional; problema que puede afectar significativamente al sistema debido a las selectividades parcialmente solapadas de los sensores. A causa de todo esto es necesario introducir la información a una etapa previa conocida como *reducción de variables*, que permitirá extraer y normalizar la información relevante que proporciona la matriz de sensores.

3.1 Reducción de variables.

El objetivo al usar métodos de preprocesamiento, es obtener un vector de datos descriptivos de cada medida que pueda ser procesado con mayor facilidad, lo que se hace para lograr dicho objetivo, básicamente es escoger un subconjunto de variables, de tal manera que el espacio de características quede reducido a un conjunto de datos más pequeño empleando técnicas de análisis multivariante.

Las técnicas de análisis multivariante, utilizan a su vez técnicas lineales de transformación de las variables originales a nuevos conjuntos más reducidos, las variables contenidas dentro de los nuevos conjuntos poseen dos cualidades importantes: total

independencia entre ellas y mayor representatividad, lo cual permite simplificar la información y hacer más sencillo el análisis de los resultados. Sin embargo, no solo se trata de comprimir información sino que también se debe extraer un conjunto voluminoso de datos que contenga toda la información que resulte ser esencial, ya que en la mayoría de los casos dicha información no depende de variables aisladas sino de la interrelación de las mismas, debido a esto, se usa el algoritmo de *Análisis de Componentes Principales* (PCA, *Principal Component Analysis*) puesto que permite reducir, representar y extraer información de forma simultánea.

El PCA es probablemente la técnica más antigua y más conocida del análisis multivariante, pero también la más usada por todos aquellos que trabajan en la investigación de narices electrónicas, por este motivo trataremos de explicar lo mejor posible en que consiste y como se aplica.

3.1.1 Análisis de Componentes Principales (PCA).

El objetivo principal del PCA es descubrir la verdadera dimensionalidad de los datos, aunque se estén midiendo p variables, la dimensión real de los datos puede que sea menor que p , al aplicar el PCA es posible reemplazar las variables originales por un número menor de variables, con lo que se puede determinar la dimensión real de la información y extraer así la información más importante, sin perder la información inicial [22].

Las respuestas de cada sensor a cada muestra realizada, constituyen los vectores respuesta, que se encuentran agrupados en forma de columnas dentro de una matriz denominada matriz respuesta y con el PCA es factible expresar la información de la matriz respuesta usando un menor número de variables más descriptivas llamadas componentes principales, de esta forma reducimos tanto los datos redundantes como la dimensionalidad de la información, conservando al mismo tiempo la mayor cantidad posible de la variación presente en el conjunto de datos.

Las componentes principales (PC's) se encuentran ordenadas de tal forma que la primera componente retiene la máxima variación del conjunto original de datos, la segunda componente retiene la máxima variación no recogida por la primera componente y así sucesivamente. La cantidad de componentes necesarias puede determinarse en base a diferentes parámetros, sin embargo, todos estos deben de ir ligados a la varianza

representada por cada una de ellas. Usualmente la mayor cantidad de la variabilidad existente en los datos se encuentra contenida en las dos primeras PC's, lo que hace innecesario calcular una alta cantidad de componentes, esto hace que los cálculos sean relativamente sencillos, la complejidad del análisis quizá radica en que existe una amplia variedad de aplicaciones y derivaciones diferentes ya que además de deducirse estadísticamente, las PC's pueden encontrarse utilizando argumentos puramente matemáticos incluso optimizar diversos criterios algebraicos, que se pueden encontrar en [22,23].

Considerando un grupo de variables (x_1, x_2, \dots, x_p) , cada una con el mismo número de muestras. Las variables de dicho grupo registran información de un vector del mismo tamaño, el cual en forma matricial se representa de la siguiente manera:

$$X = \begin{bmatrix} x_{11} & \dots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \dots & x_{np} \end{bmatrix}$$

Donde n es el número de sensores en el array y p es el número de muestras experimentadas, por lo tanto el espacio de respuestas es n -dimensional. La matriz X constituye la tabla de datos u observaciones, el arreglo de columnas y renglones en conjunto representan el 100% de la información. A partir de esta matriz se calcula un nuevo conjunto (C_1, C_2, \dots, C_p) no correlacionados entre sí, en el cual sus varianzas vayan decreciendo progresivamente. Cada elemento C_j ($j = 1, \dots, p$) representan una combinación lineal de las variables originales (x_1, x_2, \dots, x_p) [22,24].

$$C_j = a_{j1}x_1 + a_{j2}x_2 + \dots + a_{jp}x_p \quad (3.1)$$

$$C_j = a'_j x \quad (3.2)$$

Donde $a'_j = (a_{1j}, a_{2j}, a_{pj})$ y $x = [x_1 \dots x_p]^T$, a'_j es una matriz conformada por los coeficientes (a) de las componentes y representa el peso de las variables en cada componente. Estas componentes explican una parte de la varianza total, pretendiendo encontrar k componentes que representen casi toda la varianza de la información.

La primera componente principal representa la mayor varianza observada sujeta a cumplir la condición de ortonormalidad, ecuación 3.3, el segundo componente principal se obtiene calculando los valores de a_2 de tal manera que C_1 y C_2 sean no correlacionados

(ortogonales). Del mismo modo se eligen los siguientes componentes cada uno con menor varianza que el anterior.

$$a_j' a_j = \sum_{k=1}^p a_{kj}^2 = 1 \quad (3.3)$$

Para obtener la primera componente principal es necesario considerar una matriz de datos en desviaciones respecto a su media S , la primera componente viene dada por:

$$\begin{bmatrix} C_{11} \\ \vdots \\ C_{1n} \end{bmatrix} = \begin{bmatrix} x_{11} & \cdots & x_{1p} \\ \vdots & \ddots & \vdots \\ x_{n1} & \cdots & x_{np} \end{bmatrix} \begin{bmatrix} a_{11} \\ \vdots \\ a_{1p} \end{bmatrix} \leftrightarrow C_1 = X a_1 \quad (3.4)$$

donde C_1 es una variable estandarizada con respecto a su media. La máxima varianza de C_1 está definida de la siguiente forma:

$$Var(C_1) = a_1' S a_1 \quad (3.5)$$

donde $S = X X' \frac{1}{n}$.

Se utilizan *multiplicadores de Lagrange* para maximizar la función sujeta a las condiciones de ortonormalidad, ecuación 3.3, y máxima varianza, ecuación 3.5, donde a_1 es la incógnita que representa el vector con la combinación lineal óptima, con esto se busca uno de los extremos, en este caso el máximo [24].

$$L = a_1' S a_1 - \lambda (a_1' a_1 - 1) \quad (3.6)$$

De la ecuación 3.6 se obtiene:

$$S a_1 = \lambda I_p a_1 \quad (3.7)$$

Resolviendo se obtienen p raíces características, donde I_p representa la matriz identidad. Para que el sistema tenga una solución distinta de cero, la matriz $(S - \lambda I_p)$ tiene que ser singular, es decir, que su determinante debe ser igual a cero $|(S - \lambda I_p)| = 0$, tomando el mayor de los autovalores λ_1 se obtiene el vector característico asociado a a_1 , este vector brinda la mejor combinación lineal de las variables originales representando la mayor varianza. Se puede asociar a C_1 al eje principal de la nube de datos en el un nuevo sistema de referencia, lo que significa que sobre C_1 se representa la mayor distribución de los datos. Se cumple que [22]:

$$Var(C_1) = X a_1 = \frac{1}{n} a_1' X' X a_1 = a_1' S a_1 = a_1' \lambda_1 I_p a_1 = \lambda_1 a_1' a_1 = \lambda_1 \cdot 1 = \lambda_1$$

Para obtener la segunda componente, se aplica el mismo procedimiento, los coeficientes a_2 del segundo componente principal C_2 , corresponde al segundo mayor autovalor λ_2 . De la misma forma se obtienen los demás componentes, cada uno de los cuales puede ser

representado como el producto de una matriz formada por los autovectores multiplicada por el vector X que contiene las variables originales, con lo que, la matriz de covarianzas del nuevo conjunto optimizado tendrá valores diferentes de cero solo en su diagonal [24]:

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_p \end{bmatrix}$$

Lo cual es de esperarse, puesto que, C_1, C_2, \dots, C_p forman un conjunto ortogonal, lo que significa que no existe correlación entre ellos.

$$\Lambda = \text{Var}(C) = a' \text{Var}(X) a = a' S a, \text{ por propiedades } S = a \Lambda a'$$

Por propiedades matriciales se reduce drásticamente el número de autovalores, cada autovalor λ_i corresponde a la varianza de su respectiva C_i y al sumar todos los autovalores se obtiene la varianza total de las componentes [25].

$$\sum_{i=1}^p \text{Var}(C_i) = \sum_{i=1}^p \lambda_i = \text{Traza}(\Lambda) \quad (3.8)$$

$$\text{Traza}(\Lambda) = \text{Traza}(a' S a) = \text{Traza}(S) = \sum_{i=1}^p \text{Var}(X) \quad (3.9)$$

Con las ecuaciones anteriores se demuestra que la suma de las varianzas originales y la suma de las varianzas de las CP's son idénticas. La conservación de la varianza permite cuantificar en porcentaje la dispersión que recoge cada componente y del mismo modo, cuantificar el porcentaje acumulado de los primeros k componentes [25].

$$\frac{\sum_{i=1}^k \lambda_i}{\text{Traza}(\Lambda)}, \text{ donde se pretende que } k \ll p \quad (3.10)$$

Finalmente se llega a la presentación de un nuevo conjunto de variables sintético; cada componente constituye un porcentaje de la información total.

$$X = \begin{bmatrix} C_{11} & \cdots & C_{1p} \\ \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{np} \end{bmatrix}$$

Conociendo el aporte de cada componente, es posible decidir el número k de componentes a utilizar, que representen un porcentaje definido menor al 100%, reduciendo significativamente la cantidad de variables. El coeficiente de correlación r_{ij} entre una componente C_j y una variable X_i , se calcula multiplicando el peso de la variable en ese componente por la raíz cuadrada de su valor propio [25].

$$r_{ij} = \sqrt{\lambda_j} a_{ij}, \quad i = 1, \dots, p \text{ y } j = 1, \dots, k \quad (3.11)$$

Estos coeficientes se denominan pesos o cargas factoriales. La matriz de cargas es parte de la dispersión total de la variable expuesta por el conjunto k componentes, además, la suma de todos estos factores coincide con el porcentaje de dispersión definido por las k componentes.

Las componentes principales permiten hacer una representación gráfica en pocas dimensiones de los hechos más sobresalientes de una tabla de datos, los gráficos establecen la relación de las componentes en representaciones bidimensionales o tridimensionales, puesto que, las componentes principales son consideradas como los ejes sobre el plano del cual se proyectan las respectivas coordenadas o valores de carga para cada variable en relación a cada una de las componentes. En ellos se trata de identificar los grupos de variables que se forman teniendo en cuenta que aquellas que tienen mayor carga o proyección aparecerán al final de un eje [22,26].

3.2 Análisis y clasificación de patrones.

Una vez que se han reducido las variables es necesario analizarlas introduciendo los resultados obtenidos con el PCA a un sistema de reconocimiento de patrones, el cual actúa como un sistema de procesamiento de datos, cuyo objetivo es asignar un patrón¹⁷ a la clase¹⁸ a la que pertenece, a esta etapa se le conoce como clasificación. En esta fase pueden presentarse dos situaciones distintas: se puede dar un conjunto de muestras para que sean etiquetadas, pero las clases e incluso el número de grupos es desconocido, por lo cual el sistema debe determinar la clase de la cual se obtiene la muestra o bien, saber con certeza el número de clases, en cuyo caso el objetivo es establecer una regla con la cual sea posible clasificar una nueva muestra en uno de los grupos ya existentes, aquí el sistema calcula las categorías del conjunto de muestras usando un conjunto de entrenamiento previo. El primer caso es conocido como *aprendizaje no supervisado o agrupamiento* y el segundo es llamado *aprendizaje supervisado o discriminación*.

Para comenzar el proceso de clasificación es necesario especificar las salidas que va a proporcionar el sistema, de igual forma es preciso definir el conjunto de clases, pero para esto es necesario considerar la naturaleza de dichas clases, basado en [27] se distinguen tres

¹⁷ Un patrón es un conjunto de características.

¹⁸ Una clase es un conjunto de patrones similares.

casos comunes, pero en la práctica los conjuntos de datos pueden ser mezclas de estos tipos o pueden encontrarse en un punto intermedio:

- ❖ Cuando las clases corresponden a etiquetas para diferentes poblaciones. Las etiquetas para las diferentes poblaciones están dadas por las clases, es decir, que la pertenencia a las diversas poblaciones no está en cuestión. Por ejemplo, la acetona y el arsénico son dos compuestos volátiles muy diferentes, y se sabe con certeza, si un compuesto es acetona o arsénico o ninguno de los dos, por lo cual la asignación a una de las clases se determina independientemente de cualquier atributo o variable en particular.
- ❖ Las clases son el resultado de un problema de predicción. En esencia, las clases son un resultado que debe predecirse a partir del conocimiento de las variables. Si se desea verlo desde el punto de vista estadístico, las clases son una variable aleatoria. Por ejemplo, al plantear la siguiente pregunta ¿el paciente está enfermo? Las clases se determinan a partir de todos los atributos con los que se dota al sistema y las clases serían: sí (clase 1) y no (clase 2).
- ❖ Las clases están predefinidas por una partición del espacio de muestra. Esto quiere decir que las clases se determinan de las variables mismas, se puede decir que las clases son funciones de los atributos. Por lo tanto, un paciente puede clasificarse como enfermo si algunos atributos están fuera de los límites predeterminados, y no enfermo en caso contrario. Aquí existe una regla que ya ha clasificado los datos de los atributos: el problema es crear una regla que imite la regla real lo más cerca posible.

Muchos problemas exigen como resultado el cálculo de una probabilidad cuyos valores se encuentran entre 0 y 1, el 0 indica la inexistencia de correlación entre la variable dependiente y el modelo desarrollado, mientras 1 hace referencia a la existencia de una correlación perfecta. Sin embargo, debido a que los distintos tipos de compuestos pueden compartir características similares durante la etapa de pruebas, en algunos casos un patrón que representa un determinado compuesto será clasificado en una categoría que no le corresponde. Es aceptable tener un modelo que nos da un valor de correlación de 0.9995 para un determinado compuesto, entonces este modelo predice que es muy probable que este compuesto sea acetona, por ejemplo, y para otro compuesto con un valor de 0.0003 el

mismo modelo indica que es muy probable que no sea acetona, pero ¿qué ocurre con un compuesto con un valor de 0.6?

Para tomar la decisión sobre si el compuesto debe clasificarse dentro de la clase positiva o negativa es importante medir un umbral de clasificación (umbral de decisión) y cualquier valor por encima de este umbral indica que el compuesto pertenece a la clase positiva (es acetona), mientras que cualquier valor inferior al umbral pertenece a la clase negativa (no es acetona). Resulta muy común que se tome un umbral de clasificación de 0.5, sin embargo, la elección depende de cada problema y en algunos casos será necesario ajustar el umbral, parte de elegir un umbral es evaluar qué impacto tendrá cometer un error en la clasificación [28].

Debido a lo anterior, las predicciones ahora se miden en cuatro grupos en función de la respuesta conocida real, así como la respuesta predicha: verdadero positivo (*VP*), verdadero negativo (*VN*), falso positivo (*FP*), falso negativo (*FN*).

Un verdadero positivo es un resultado en el que el modelo predice correctamente la clase positiva. De forma similar un verdadero negativo se produce cuando el modelo predice acertadamente la clase negativa. Contrariamente, un falso positivo se origina cuando el modelo predice erróneamente la clase positiva y un falso negativo ocurre cuando el modelo predice incorrectamente la clase negativa. Estos grupos son los elementos en los que se basan las métricas binarias¹⁹ para evaluar a los clasificadores [29].

La **exactitud** es una de dichas métricas y es considerada la fracción de predicciones que el modelo realizó correctamente.

$$Exactitud = \frac{\text{Número de predicciones correctas}}{\text{Número total de predicciones}} \quad (3.5)$$

También puede ser calculada en términos de positivos y negativos de la siguiente manera:

$$Exactitud = \frac{VP + VN}{VP + VN + FP + FN} \quad (3.6)$$

Imaginando que se construye un modelo ficticio para clasificar 100 compuestos como “acetona” (clase positiva) o en “no acetona” (clase negativa). La tabla 3.1 muestra los resultados de dicho modelo.

¹⁹ Métricas usadas para clasificar los patrones en dos clases.

Tabla 3.1. Resultados de un modelo clasificador de 100 compuestos.

Verdadero Positivo (VP)	Falso Positivo (FP)
<ul style="list-style-type: none"> • Realidad: acetona. • Predicción del modelo: acetona. • VP: 1 	<ul style="list-style-type: none"> • Realidad: no acetona. • Predicción del modelo: acetona • FP: 1
Falso Negativo (FN)	Verdadero Negativo (VN)
<ul style="list-style-type: none"> • Realidad: acetona. • Predicción del modelo: no acetona • FP: 8 	<ul style="list-style-type: none"> • Realidad: no acetona. • Predicción del modelo: acetona. • VP: 90

Aplicando la ecuación 3.6 se calcula la exactitud del modelo:

$$Exactitud = \frac{VP + VN}{VP + VN + FP + FN} = \frac{1 + 90}{1 + 90 + 1 + 8} = 0.91$$

Lo cual indica que el modelo es 91% exacto, es decir que 91 predicciones de las 100 totales fueron correctas, a simple vista esto podría parecer un resultado muy favorable, sin embargo, si hacemos un análisis más detallado, quizá el resultado no sea tan bueno.

De los 91 compuestos que fueron clasificados como “acetona” el modelo identificó 90 correctamente, lo que es realmente bueno, pero de los 9 compuestos que clasificó como “no acetona”, el modelo identificó solamente 1 correctamente, esto significa que 8 de cada 9 compuestos son clasificados erróneamente, lo que implica que el modelo no es mejor que otro que tenga una capacidad predictiva de cero para distinguir acetona de los compuestos que no lo son. Con este análisis se comprueba que la exactitud por sí sola no muestra todo el panorama cuando hay una disparidad significativa entre el número de etiquetas positivas y negativas, es por ello que para poder evaluar un clasificador es necesario que intervengan dos métricas más: *precisión* y *exhaustividad*.

La **precisión** se refiere a la proporción de identificaciones correctas y es definida de la siguiente forma:

$$Precisión = \frac{VP}{VP + FP} \tag{3.7}$$

Calculando la precisión del ejemplo anterior obtenemos:

$$Precisión = \frac{VP}{VP + FP} = \frac{1}{1 + 1} = 0.5$$

Este modelo tiene una precisión de 0.5, es decir, cuando predice que un compuesto es acetona, acierta el 50% de las veces.

La **exhaustividad**²⁰ es la proporción de positivos reales que se identificó correctamente, se define de la siguiente forma:

$$Exhaustividad = \frac{VP}{VP + FN} \quad (3.8)$$

Con los resultados de la tabla 3.1, calculamos la exhaustividad del modelo:

$$Exhaustividad = \frac{VP}{VP + FN} = \frac{1}{1 + 8} = 0.11$$

Esto quiere decir que el modelo identifica correctamente 11% de los compuestos como “acetona”.

Para evaluar completamente la efectividad de un modelo clasificador es importante examinar la precisión y la exhaustividad, desafortunadamente, con frecuencia existe tensión entre estas dos métricas, debido a que al ajustar el umbral de clasificación para mejorar la precisión, generalmente la exhaustividad se reduce y viceversa. En función de la aplicación es como se evalúa que tan efectivo es el modelo, en otras palabras, se podría necesitar un clasificador que este muy seguro de que las predicciones positivas sean realmente positivas (alta exhaustividad) y podría permitirse la clasificación incorrecta de algunos ejemplos positivos como negativos (precisión moderada), o bien, requerir un clasificador que prediga correctamente el mayor número de ejemplos positivos posibles (alta precisión), con lo que se acepta que algunos ejemplos negativos se clasifiquen incorrectamente como positivos (exhaustividad moderada) [28,29].

Las métricas típicas que se utilizan en casos de clasificación binaria, son las mismas usadas en problemas de clasificación multiclase, una manera de abordar problemas de clasificación multiclase es descomponerlo en problemas binarios, es decir, que por cada clase se genera un clasificador binario, donde los patrones positivos serán los que tengan asociada dicha clase y los negativos el resto de las clases. Dado un patrón desconocido, este se clasificará teniendo en cuenta la salida para cada uno de estos clasificadores binarios.

Una vez que se tienen bien definidas las clases se continúa con la construcción del clasificador, construcción que hemos dividido en tres etapas: la elección del modelo, el aprendizaje del clasificador, también conocido como entrenamiento, y verificación de los

²⁰ También es llamada recuperación.

resultados. Es importante examinar algunos aspectos que pueden afectar considerablemente el desempeño del clasificador, tales como:

- ❖ Velocidad. Este factor es de suma importancia cuando las lecturas y la detección de compuestos debe hacerse de forma automática y en tiempo real, inclusive, se podría decir que es más importante que la exactitud, sin embargo, si se llegase a tener un clasificador que es 90% confiable pero 100 veces más rápido que uno que tiene un 95% de exactitud, dependiendo de la aplicación a la que se encuentre destinada la red, se escogería uno u otro.
- ❖ Tiempo de aprendizaje. Al encontrarse en un entorno que cambia de prisa, puede ser necesario que el clasificador aprenda una nueva regla de clasificación en un periodo de tiempo considerablemente breve o incluso realizar ajustes a una regla en tiempo real. El término “aprender rápidamente” en algunos casos implica usar un número reducido de características u observaciones para establecer la nueva regla.
- ❖ En cuanto al procedimiento de clasificación, este debe ser fácil de entender para el usuario, de lo contrario pueden llegar a ocurrir errores al momento de aplicar la regla.

3.2.1 Elección del modelo.

Dado que hay diferencias entre los diversos compuestos, es viable pensar en cada uno de ellos como distintos modelos que poseen distintas descripciones o variables que ayudarían a hipotetizar cada clase con sus cualidades, procesando los datos recogidos por el sensor, eliminar ruido o defectos por cada patrón percibido y generar un modelo que describa exactamente la semántica del patrón. Debido a que el reconocimiento de patrones resulta ser muy útil para escoger o categorizar datos en una forma más práctica y con menos errores, es que se ha optado por construir el clasificador usando técnicas de reconocimiento de patrones.

Actualmente se encuentran una gran cantidad de técnicas de reconocimiento que incluyen el uso de métodos estadísticos convencionales, *redes neuronales artificiales* (ANN, *Artificial Neural Network*), así como una combinación de ambos. Entre los métodos estadísticos se distinguen el análisis de componentes principales, *mínimos cuadrados parciales* (PLS, *Partial Least Square*), análisis discriminante (DA, *Discriminant Analysis*),

análisis discriminante factorial (DFA, *Discriminant Factorial Analysis*) y *análisis de agrupamientos* (CA, *Clusters Analysis*), estos métodos usualmente se utilizan como técnicas complementarias de las ANN, puesto que al ser combinadas se pueden producir clasificadores y agrupamientos que resultan ser más robustos que aquellos implementados con técnicas individuales [27,30].

En la mayoría de los casos, la naturaleza de los datos obtenidos puede llegar a ser muy compleja, por ello es necesario emplear una técnica de reconocimiento de patrones, que permita tratar con datos no lineales, para tal fin se han desarrollado métodos tales como las ANN, los *sistemas de inferencia difusos* (FIS, *Fuzzy Inference Systems*) y los *algoritmos genéticos* (GA, *Genetic Algorithms*), entre otros, que además poseen ventajas sobre los métodos convencionales tales como capacidad de aprendizaje, autoorganización, generalización y tolerancia al ruido. Debido a su sencilla implementación, su rápida respuesta y los buenos resultados que se han obtenido con las ANN, estas se han convertido en la técnica de reconocimiento de patrones más usada hoy en día [27,30].

3.2.2 Redes Neuronales Artificiales (ANN, Artificial Neural Network).

De forma general, la función de una Red Neuronal es modelar la manera en que el cerebro realiza una función o una tarea determinada que es de particular interés, esto se lleva a cabo empleando una interconexión masiva de células informáticas simples, que son denominadas *neuronas* o unidades de procesamiento, las cuales ayudan a mejorar el rendimiento de la red. En [31] una ANN es definida como "una estructura de información paralela y distribuida masivamente, formada por unidades de procesamiento simples, que tienen una propensión natural para almacenar conocimiento experimental y hacerlo disponible para su uso".

Lo que se busca al dotar a las ANN's con capacidades tanto para adquirir como para almacenar conocimiento del medio, es emular las funciones elementales de la red nerviosa del cerebro humano. La adquisición se realiza a través de un proceso de aprendizaje y para almacenar la información se utiliza las fuerzas de conexión entre neuronas, mejor conocidas como *pesos sinápticos*. Gracias a su habilidad para extraer significado de datos complicados o imprecisos las ANN's son usadas para extraer patrones y determinar tendencias que son muy complejas como para ser detectadas tanto por humanos como por

otras técnicas informáticas, estas pueden ser entrenadas para tomar la decisión de asignar un dato de entrada a una clase establecida (clasificación), como un proceso de recuperación de información a partir de una entrada relacionada con datos almacenados (asociación), cuando no se conoce una clasificación de los datos de entrada la red genera esta clasificación (agrupamiento); puede solucionar problemas de optimización de naturaleza complicada, permite anticipar el estado futuro de un sistema y controlar sistemas dinámicos complejos (predicción y control) [31,32].

En la literatura se halla una gran cantidad de ventajas que se obtienen al implementar un sistema basado en redes neuronales, entre las que destacan:

- ❖ Aprendizaje adaptativo. Esto hace referencia a la habilidad que posee la red para aprender cómo realizar determinadas tareas basándose en los datos disponibles para el entrenamiento o como experiencia inicial. Esta cualidad favorece al diseñador puesto que su función es únicamente la obtención de la arquitectura adecuada, dicho en otras palabras, el diseñador no debe preocuparse por el cómo la red aprenderá a discriminar, no obstante si es necesario que desarrolle un buen algoritmo de aprendizaje que le proporcione a la red dicha capacidad, mediante un entrenamiento con patrones [32]. Las redes neuronales son sistemas dinámicos autoadaptativos, debido a la capacidad de autoajuste que poseen las neuronas que componen el sistema, lo que les permite cambiar constantemente para poder adaptarse a las nuevas condiciones.
- ❖ Autoorganización. Esta característica consiste en la modificación completa de la red neuronal para llevar a cabo un objetivo específico, es decir, cuando las redes neuronales se usan para reconocer ciertas clases de patrones, ellas autoorganizan la información usada. Por ejemplo, la red Backpropagation, creará su propia representación característica, mediante la cual puede reconocer ciertos patrones.
- ❖ Operación en tiempo real. La comparación entre los datos, así como el análisis de la relación que existe entre uno y otro, y en general los cálculos que llevan al resultado deseado, pueden llevarse a cabo simultáneamente.
- ❖ Tolerancia a fallos. Hay dos aspectos distintos respecto a la tolerancia a fallos, el primero es respecto a los datos, la red puede aprender a reconocer patrones con ruido, distorsionados o incompletos. Segundo, si se produce un fallo en un pequeño

número de neuronas, aunque el sistema se ve influenciado este no sufre una caída repentina, es decir, que pueden seguir realizando su función con cierta degradación aunque se destruya parte de la red. Las redes neuronales son tolerantes a fallos debido a que la información está distribuida en las conexiones entre neuronas, lo que implica que existe cierto grado de redundancia en este tipo de almacenamiento. A diferencia de la mayoría de los ordenadores algorítmicos y sistemas de recuperación de datos, las ANN's almacenan información no localizada, por lo cual la mayoría de las interconexiones entre los nodos de la red tendrán unos valores en función de los estímulos recibidos, y esto generará un patrón de salida que represente la información almacenada [33].

- ❖ **Facilidad de uso.** Si bien es cierto que el usuario necesita tener cierto conocimiento de cómo seleccionar y preparar los datos, de cómo seleccionar una red neuronal apropiada o de cómo interpretar los resultados, el nivel de conocimiento requerido para aplicar exitosamente las redes neuronales es mucho menor que el necesario para aplicar otros métodos estadísticos no lineales [32].

Todas estas características han contribuido a hacer de las redes neuronales un popular método de reconocimiento de patrones.

Según [34], las diversas arquitecturas neuronales las ANN's se estructuran de la siguiente manera:

- ❖ *Según el número de capas* las redes se dividen en *redes monocapa* y *multicapa*, las primeras son redes muy sencillas ya que se conforman de una capa de neuronas que proyectan las entradas a una capa de neuronas de salida, donde se realizan diversos cálculos. En las segundas existe un conjunto de capas intermedias entre la capa de entrada y salida, que son llamadas capas ocultas, este tipo de red puede conectarse total o parcialmente.
- ❖ *Según el tipo de conexiones* se tiene la siguiente clasificación; redes neuronales *no recurrentes* en las cuales la propagación de las señales se produce en un solo sentido, lo cual impide la existencia de retroalimentación y por ello este tipo de red no tiene memoria y la redes *recurrentes* que se caracterizan por sus lazos de retroalimentación, lo cuales pueden establecerse entre neuronas de diferentes capas,

de la misma capa o entre una misma neurona, esta última estructura se enfoca principalmente en el estudio de la dinámica de sistemas no lineales.

- ❖ Según el grado de conexión existen las redes *totalmente conectadas* y redes *parcialmente conectadas*, en el primer caso las neuronas de una capa se encuentran conectadas con las neuronas de la siguiente capa (no recurrentes) o con las neuronas de la capa anterior (recurrentes). En el segundo caso no hay conexión total, las estructuras neuronales se pueden conectar entre sí para dar lugar a estructuras mayores.

Los distintos modelos de ANN's siguen filosofías de diseño, reglas de aprendizaje y funciones de construcción de las respuestas muy distintas, sin embargo todos poseen los mismos elementos principales que se indican en la figura 3.1.

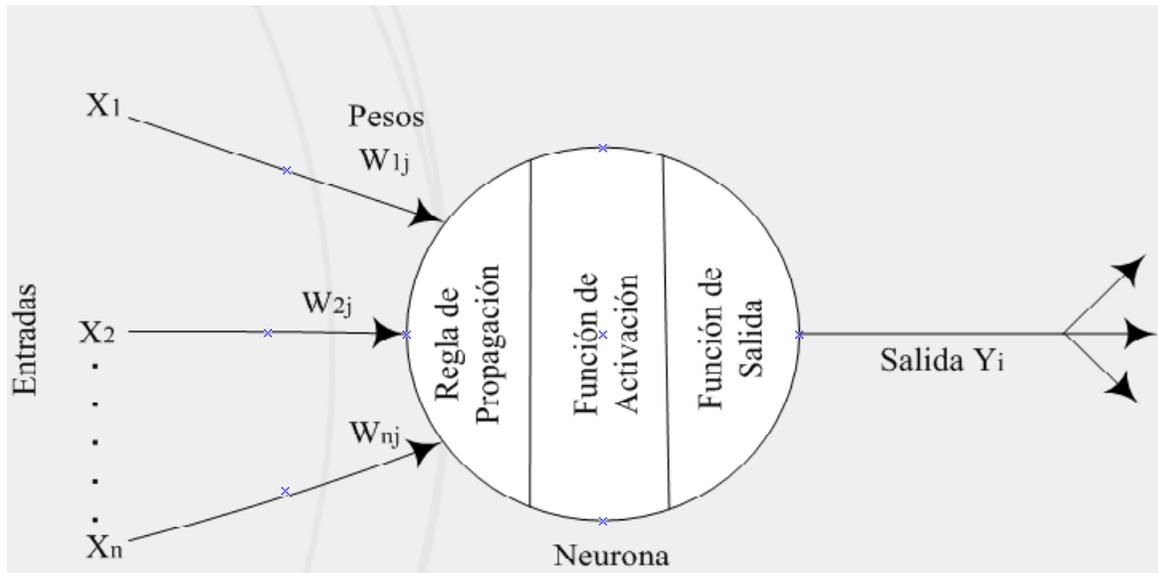


Figura 3.1. Elementos básicos de una Red Neuronal Artificial.

- ❖ Neurona artificial. Es el elemento de procesamiento que recibe información ya sea de los sensores o de otras neuronas artificiales, que realizan operaciones simples en los datos y después proporcionan una salida.
- ❖ Conjunto de entradas. Son un vector formado por la información que proporciona el entorno, en este caso serán las señales recibidas del sensor, una vez que estas señales entran al sistema de adquisición y posteriormente al PCA, los datos que arroja este último pasan a las neuronas de la capa de entrada de la red.

- ❖ Pesos sinápticos. Son coeficientes que se pueden adaptar dentro de la red, estos determinan la intensidad de la señal de entrada registrada por una neurona artificial. Cada neurona suele recibir múltiples entradas simultáneas, a las cuales se les asigna un peso relativo que proporciona la importancia de la entrada, esto es importante debido a que algunas entradas tienen mayor efecto sobre el procesamiento de la neurona al combinarse para producir una respuesta.
- ❖ Regla de propagación. Esta regla utiliza las entradas y los pesos para determinar el valor de potencial postsináptico de la neurona, usualmente se realiza la suma ponderada de cada entrada, para ello se agrupan las entradas y los pesos en dos vectores y sobre estos dos vectores se hace el producto escalar, sin embargo, la función de propagación puede llegar a ser mucho más compleja, ya que las entradas y los pesos se pueden combinar de diferentes maneras antes de pasarle el valor a la función de activación y pueden usarse diversos algoritmos de normalización. El algoritmo específico para la función de propagación de las entradas está determinado por la elección de la arquitectura de la red [34].
- ❖ Función de activación. También conocida como función de transferencia, la cual es un proceso algorítmico para transformar el resultado de la regla de propagación en la salida real de la neurona y su valor es normalmente no lineal ya que el uso de funciones lineales tiene muchas limitaciones debido a que el valor de la salida es proporcional al valor de la entrada [34].
- ❖ Función de salida. Cada elemento de procesamiento tiene permitido una única salida, que puede estar asociada con un número elevado de otras neuronas [34].

3.2.3 Entrenamiento del clasificador.

Generalmente las ANN's dan resultados rápidamente y son eficientes con el procesamiento de la información, pero también presentan un grado de complejidad, puesto que a veces puede llegar a ser bastante difícil elegir los parámetros óptimos de la red, al igual que los procedimientos de entrenamiento, sin embargo, se puede partir del hecho de que las ANN, al igual que las personas, aprenden como procesar la información mediante el ejemplo, antes de que puedan ser usadas en una aplicación específica. El proceso de entrenamiento de una ANN involucra el ajuste de los pesos de entrada en cada neurona

hasta que la red se aproxime a la salida deseada. Este proceso involucra los datos de cada nodo de entrada y la respuesta deseada por cada nodo de salida de la red, al concluir el entrenamiento, sólo los datos de entrada son provistos a la red, la cual “recuerda” la respuesta que “aprendió” durante el entrenamiento.

El aplicar aprendizaje supervisado implica que es necesario entrenar un clasificador para relacionar los valores detectados a etiquetas de compuestos específicos. Entre los algoritmos supervisados empleados en narices electrónicas destacan: el entrenamiento de propagación hacia atrás (*backpropagation trained*), redes alimentadas hacia adelante (*feed forward networks*), cuantificadores de vectores de aprendizaje y *ARTmaps difusos*, contrariamente los algoritmos no supervisados no requieren de clases de compuestos predeterminados como entrenamiento, si no que desarrollan agrupamientos de datos dentro de grupos similares basados en los atributos medidos o en las características que sirven como entradas para los algoritmos, los mapas auto organizados (SOM's – *Self Organized Maps*) y la teoría de resonancia adaptativa en redes, son ejemplos de redes neuronales no supervisadas [35]. Sea cual sea el algoritmo seleccionado para lograr que la red aprenda satisfactoriamente es indispensable entrenarla.

La etapa de entrenamiento busca que las redes neuronales calculen las predicciones de las variables requeridas por el usuario y cada vez que obtienen un valor lo comparan con el valor real que tuvo la variable que se intentó predecir; de esta forma la red puede percibir la proporción de su error, para después modificar los pesos sinápticos y con ello disminuir dicho error. Habitualmente los pesos se inicializan con valores aleatorios e incluso se pueden igualar a cero (antes de que la red empiece a aprender), de esta forma la red tiene un punto de partida, desde el cual puede comenzar a adaptar los pesos. La manera más común para ir adaptando los pesos es mediante una distribución proporcional del error cometido en la predicción entre todas las neuronas según el aporte de cada una al resultado final, en otras palabras entre mayor sea la aportación de la neurona o de la conexión, mayor parte del error se le asignará a ella para modificar los pesos.

Para estimar la eficiencia de un algoritmo de entrenamiento se emplea un método llamado *hold-out* de test, este es considerado un método básico debido a que es el más extendido y el más sencillo de aplicar puesto que mide el error que comete el algoritmo sobre un conjunto de muestras que se mantienen al margen durante el proceso de entrenamiento [32],

lo que quiere decir que un conjunto de muestras denominado conjunto de *test* es separado y reservado con el fin de comprobar la generalización de un modelo una vez entrenado; sin embargo, para poder obtener una medida más exacta de la estimación cuando las muestras son limitadas es necesario utilizar otro método, siendo la *validación cruzada* el utilizado en estos casos.

La validación cruzada *k-fold* distribuye los datos disponibles en k subconjuntos con aproximadamente el mismo número de muestras y supone k iteraciones, de las cuales el algoritmo utiliza $k-1$ subconjuntos para entrenar la red y deja fuera uno de ellos que habitualmente es usado como conjunto de test. El subconjunto que es dejado fuera va rotando en cada iteración de forma que, al final del proceso todos los datos han formado parte del entrenamiento y del test en algún momento. Por lo tanto la medida del error cometido se determina a través de la media aritmética de los errores cometidos en las diferentes iteraciones. Cuando k es igual al número de muestras totales (N), se produce el método denominado *leave-one-out* (LOO), el cual evalúa N veces el modelo y cada prueba la realiza con una muestra distinta, este tipo de validación trabaja bien cuando se utilizan funciones de error continuas, pero también es cierto que posee desventajas en cuanto al tiempo de computo que requiere para llevarse a cabo, además es posible encontrar dificultades al emplearlo en tareas de clasificación debido a que en su mayoría son funciones discontinuas y su falta de continuidad lo hace poco recomendable en la selección de modelos, puesto que un pequeño cambio en los datos puede dar lugar a un cambio radical en el modelo escogido [31,32].

Existe un tercer algoritmo llamado *submuestreo aleatorio*, que también es un método de validación cruzada, pero a diferencia del LOO, dado un conjunto de muestras este extrae un número determinado de ellas que utiliza para entrenar la red y el resto lo deja para testar, el proceso se repite un número determinado de veces, pudiéndose extraer varias veces la misma muestra para entrenar y finalmente se promedia el error de test obtenido en cada una de las repeticiones. Este algoritmo también es conocido como *bootstrap* ya que utiliza muestreo con reemplazamiento para generar las muestras que se usaran para entrenar la red.

Es importante prestar especial atención a la elección de la arquitectura y a los pesos iniciales de la red porque de ellos depende en gran medida el éxito obtenido en el proceso de entrenamiento. Las redes neuronales multicapa con entrenamiento *backpropagation*, son

las más populares en cuanto a problemas de clasificación se trata, por ende es la topología en la que se basa el sistema implementado para este proyecto. En esta red, para poder tener suficientes grados de libertad computacional se han organizado los elementos de procesamiento en una arquitectura regular de tres grupos distintos de neuronas, tal como se muestra en la figura 3.2, estos son: capa de entrada, capa oculta y capa de salida; únicamente en las dos últimas capas, las unidades de procesamiento son neuronas, con lo que se tienen solo dos capas de pesos.

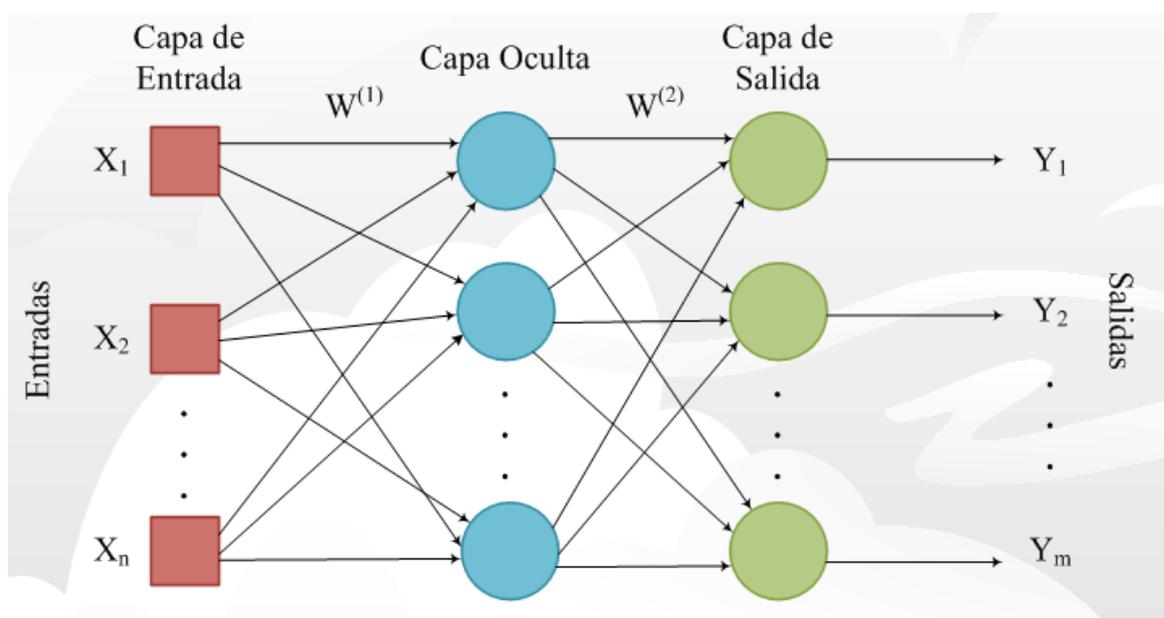


Figura 3.2. Ejemplo de una red neuronal multicapa.

Dado que la arquitectura y los parámetros deben determinarse de forma experimental, el tiempo que tome el hallar una red óptima puede ser considerable usando reglas generales que a menudo se utilizan para establecer la arquitectura de la red sin emplear una gran cantidad de tiempo, estas reglas estipulan que la cantidad de nodos en la entrada usualmente se determina para que corresponda a la cantidad de sensores en la matriz o el número de características consideradas para el conjunto de sensores; el número de neuronas en la capa oculta se determina experimentalmente, pero este no debe ser mayor que el número de neuronas en las otras dos capas y el número de neuronas en la capa de salida no es mayor al número de compuestos que se clasificará puesto que generalmente la cantidad de compuestos analizados definen el número de neuronas que se tendrán en la salida [32].

Al entrenar la red con un algoritmo backpropagation la fase de aprendizaje de la red es supervisado, por lo que se emplea un conjunto de vectores de entrenamiento, para comenzar es necesario proporcionar un conjunto de entrenamiento, una vez hecho esto las neuronas se encargaran de calcular sus entradas ponderadas y realizar una transformación no lineal de esta suma usando una función de activación dada, por ejemplo una función de transferencia tangente sigmoideal que restringe la salida a un valor entre $[0, +1]$ o $[-1, +1]$, esta función comprime las entradas que tienen un rango infinito a un rango finito de salida y se caracterizan debido a que sus pendientes deben aproximarse a cero cuando las entradas son grandes. Los pesos asociados a las neuronas son modificados hasta minimizar la función de error entre las salidas obtenidas y las salidas deseadas. La propagación de las señales tiene dos fases, la fase *forward*, en donde las señales se propagan desde las entradas hacia las salidas, y la fase *backward*, en la cual se realiza la corrección de los pesos en sentido contrario, es decir, de las salidas hacia la entrada, este proceso se repite continuamente hasta que la red haya caído en un nivel de error sea pequeño y constante, (para medir el rendimiento de la red se puede usar la suma cuadrática del error), al estabilizar el error, se dice que la red ha sido entrenada.

3.2.4 Validación de los datos.

Una vez que la red ha sido entrenada, se puede usar para clasificar la pertenencia de nuevas muestras, generalmente el vector respuesta de la nueva muestra se introduce en la capa de entrada de la red neuronal y esta arroja la adecuada clasificación de dicha muestra, sin embargo existen casos en los que la red memoriza las muestras que se le han presentado a la entrada durante el entrenamiento, de tal manera que incluso se ajusta al ruido existente en éstas; a esto se le conoce como *sobre entrenamiento (overfitting)* y cuando esto ocurre, la generalización con respecto a muestras nuevas se verá afectada significativamente, para evitar esta complicación es importante detener el entrenamiento a tiempo. Llegados a este punto una pregunta que surge muy a menudo es ¿cómo saber cuándo se debe parar el entrenamiento de la red? Para determinar esto se utiliza el criterio denominado *early-stopping*, y este consiste en extraer un subconjunto de muestras del conjunto entrenamiento, el cual recibe el nombre de conjunto de *validación* y utilizarlo de manera auxiliar durante el entrenamiento.

La función que desempeña el conjunto de validación es que, tras cada iteración o tras cada cierto número de iteraciones, determina el error de la red y el momento en que este comienza a aumentar. Debido a que el conjunto de validación es dejado al margen durante el entrenamiento, el error que se comete sobre él es un buen indicativo del error que la red cometerá sobre el conjunto de test y en el momento en que el error del conjunto de validación aumente se procederá a detener el entrenamiento de la red y se conservarán los valores de los pesos de la iteración anterior. De esta forma se comprueba que la red neuronal esta lista para clasificar nuevas muestras sin que su generalización se vea afectada.

Capítulo 4

Implementación del sistema.

Cada uno de los bloques contenidos dentro del diagrama ilustrado en la figura 4.1, forman parte fundamental en la construcción de una nariz electrónica. La huella formada por el conjunto de respuestas de los sensores de la matriz entra a la etapa de adquisición de datos en donde el RTL-SDR recibe la señal de salida de los sensores, que tiene un valor aproximado de 160 MHz y la transforma a una señal digital que será procesada por la BBB, debido a que la señal obtenida por el RTL-SDR es una señal mezclada fue necesario utilizar las series de Fourier para adquirir la frecuencia de oscilación y la transformada de Fourier para encontrar la máxima frecuencia y con ello analizar el comportamiento del sensor cuando interacciona con el compuesto, sin embargo, como a la salida de la adquisición se obtuvieron una gran cantidad de datos fue necesario introducir dichos datos a una etapa de reducción de variables, misma que se llevó a cabo mediante la aplicación del PCA. Después de aplicar el análisis de componentes principales se usaron los tres ejes más representativos para entrenar una red neuronal con el fin de clasificar los compuestos.

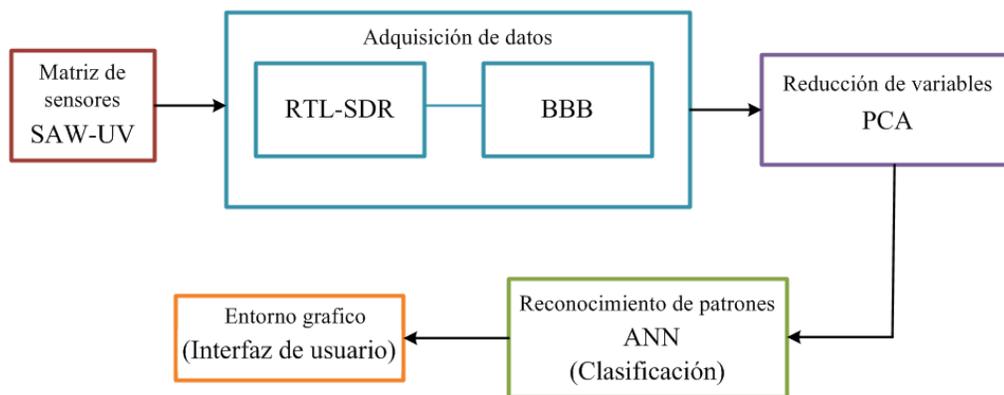


Figura 4.1. Diagrama general del proyecto.

4.1 Matriz de sensores.

La elección de los sensores así como su configuración se realiza teniendo en cuenta las propiedades físico-químicas tanto del sensor como del gas, el conjunto de las señales procedentes de la matriz será característico para cada analito, permitiendo discriminarlos y

clasificarlos. Los sensores individuales, como dispositivos de detección de gases presentan muchas limitaciones, a causa de que su sensibilidad está orientada a un tipo de compuesto en particular, identificando sólo algunos tipos de gases, es por esto que es necesario utilizar una matriz de sensores, no obstante, al usar un arreglo de dos sensores siguen existiendo las mismas limitaciones que se tienen al usar los sensores de forma individual, es por ello que los arreglos se hacen a partir de tres sensores. Dado que uno de los objetivos de esta tesis es reducir la robustez del sistema se buscó usar la menor cantidad de sensores, sin alterar la respuesta optima de la matriz, a causa de esto, el arreglo que se utilizó en este proyecto está formado por 4 sensores que se colocaron en forma de cuadrado puesto que de esta forma fue más sencillo conectar los sensores al circuito oscilador. En la figura 4.2 se observa el diseño de la matriz. Existen trabajos de investigación enfocados a comprobar la eficiencia de la respuesta de los sensores posicionados de diferentes formas dentro de la matriz, ya que no es nuestro objetivo probar con que posicionamiento se encuentran mejoras en la eficiencia de los sensores, únicamente se tomó un modelo funcional.

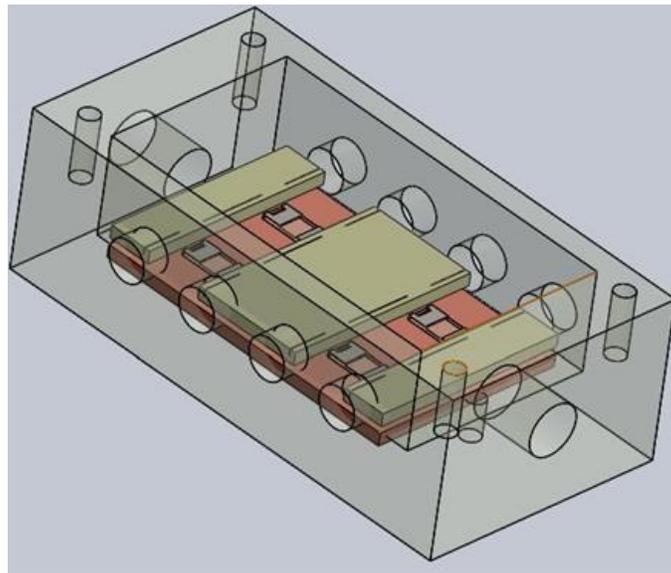


Figura 4.2. Diseño de la matriz de sensores.

Para este proyecto se utilizó como método de excitación la radiación UV, que al interactuar con la materia se considera indirectamente ionizante, debido a esto, al colocar los leds UV sobre los sensores es posible cambiar las propiedades de la capa sensible

mediante la ionización²¹ de la misma, es decir, que pueden existir modificaciones en la conductividad, polarización, activación de enlaces, etc., lo cual hace que las partículas se adhieran con mayor facilidad a la capa guía y por tanto esta capta con mayor facilidad cambios de densidad en su superficie, lo que permite que sean capaces de medir variaciones en ppm e incluso hasta ppb. La figura 4.3 representa el esquema del proceso de ionización.

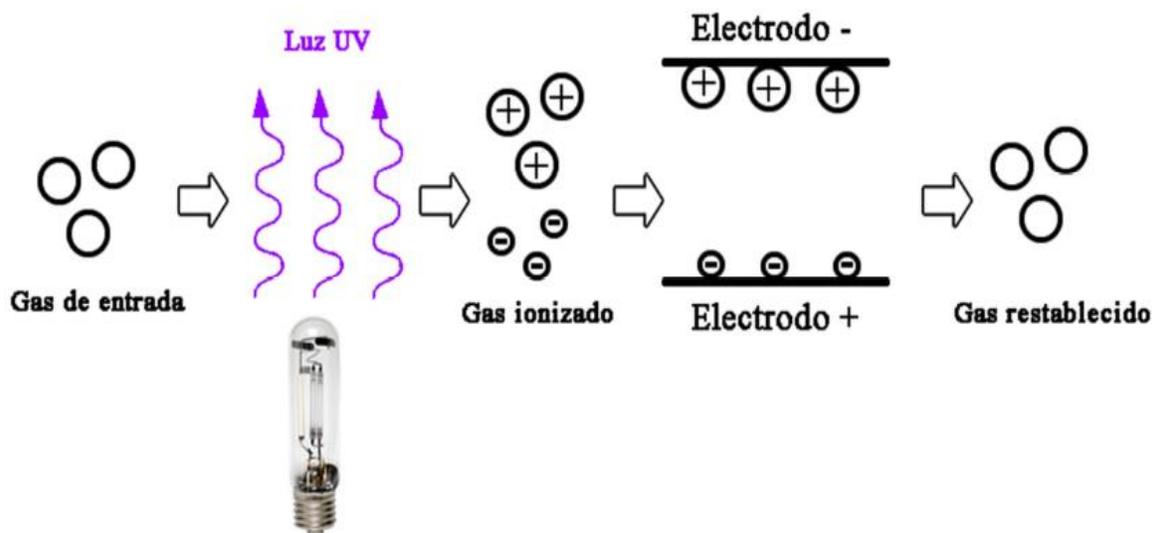


Figura 4.3. Esquema del proceso de ionización.

En la figura 4.4 observamos el interior de la matriz, en la cual los sensores se colocaron sobre soportes soldados a una placa PCB, los contactos de los IDTs se soldaron a las pistas que están dirigidas a los conectores SMA, la primer placa se soldó a una segunda que funge como tierra, la cámara fue hecha con filamento de *ácido poliáctico* (PLA) y construida con una impresora 3D, en la figura 4.5 se ilustra el diseño del circuito de iluminación UV que se colocó en la tapa de la matriz, donde se soldaron cuatro leds UV y se agregó el sensor de humedad y temperatura ChipCap® 2, cuyo diagrama se muestra en la figura 4.6. En la figura 4.7 podemos observar el sistema de iluminación UV en funcionamiento.

²¹ Proceso mediante el cual se forman átomos o moléculas (iones) cargadas eléctricamente debido al exceso o falta de electrones respecto a un átomo o molécula neutra.

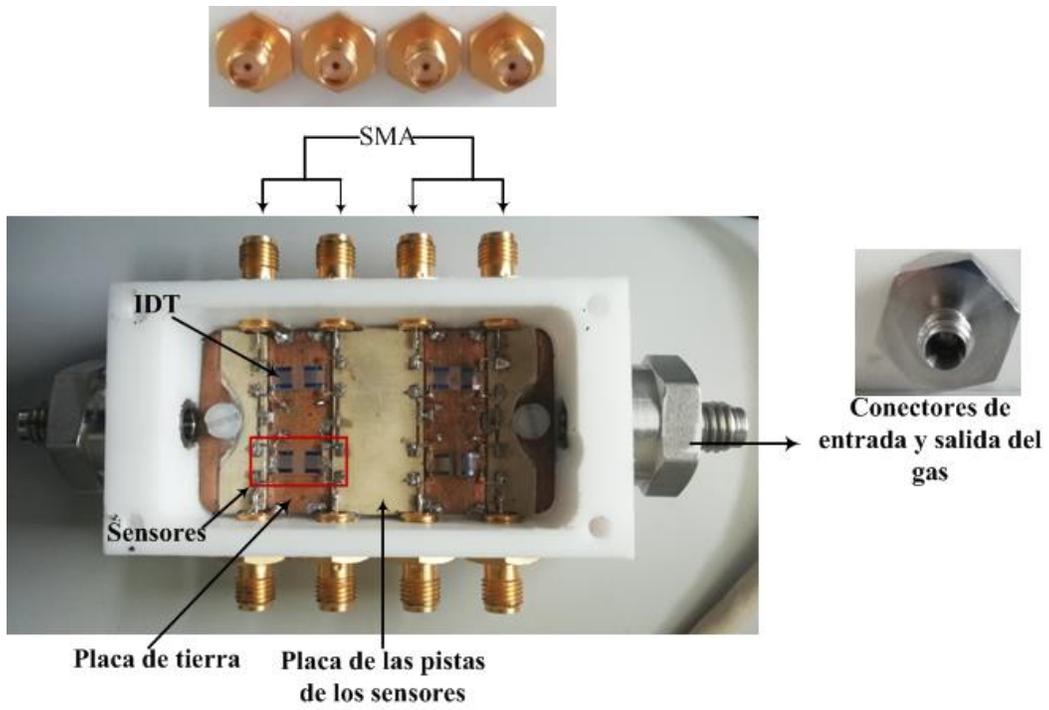


Figura 4.4. Interior de la matriz de sensores.

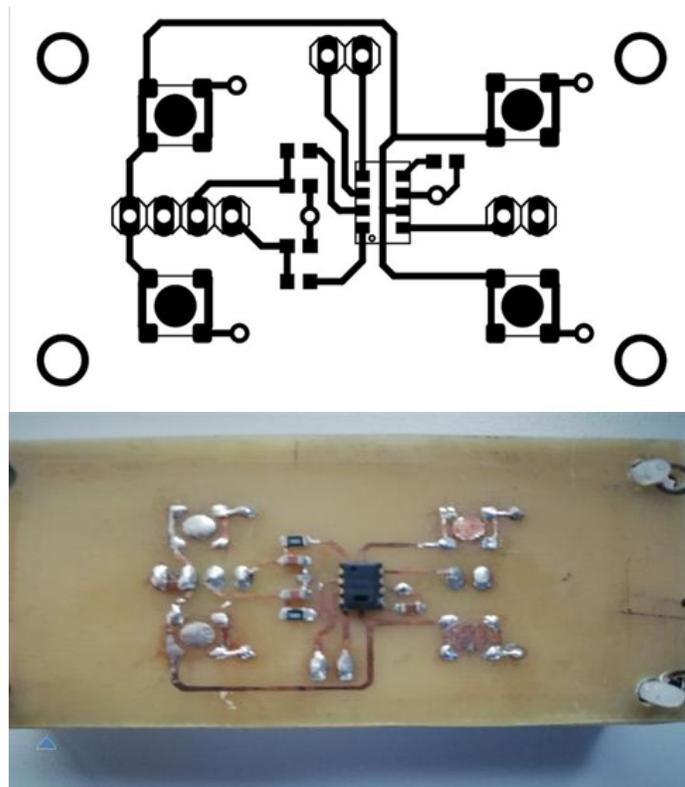


Figura 4.5. Circuito de leds y sensor de temperatura.

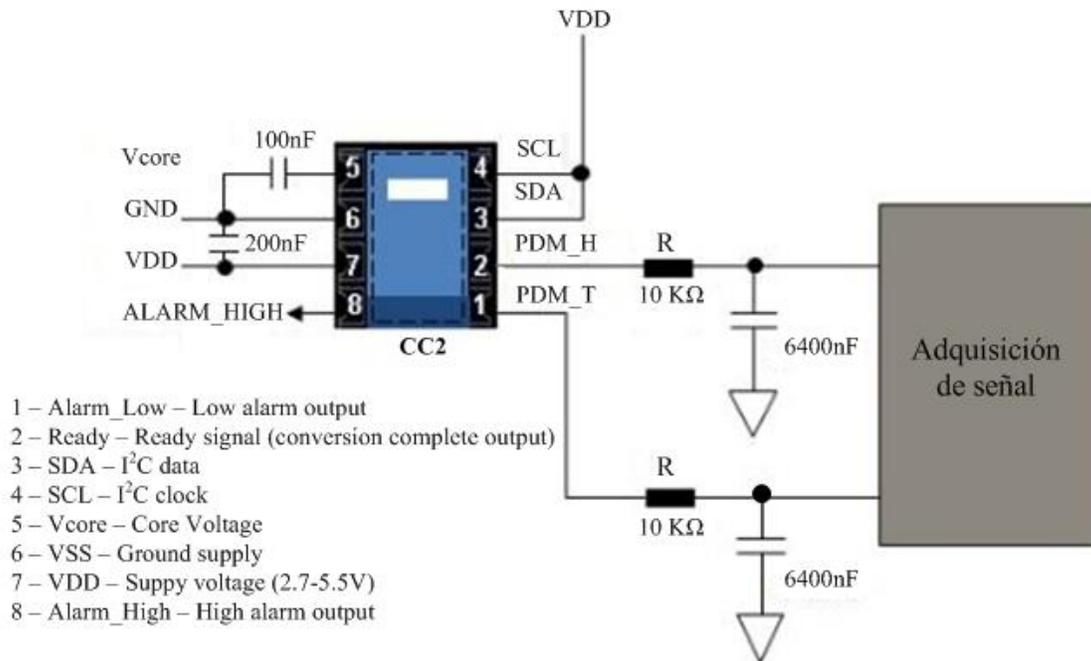


Figura 4.6. Diagrama del circuito del sensor de temperatura ChipCap2 (CC2).

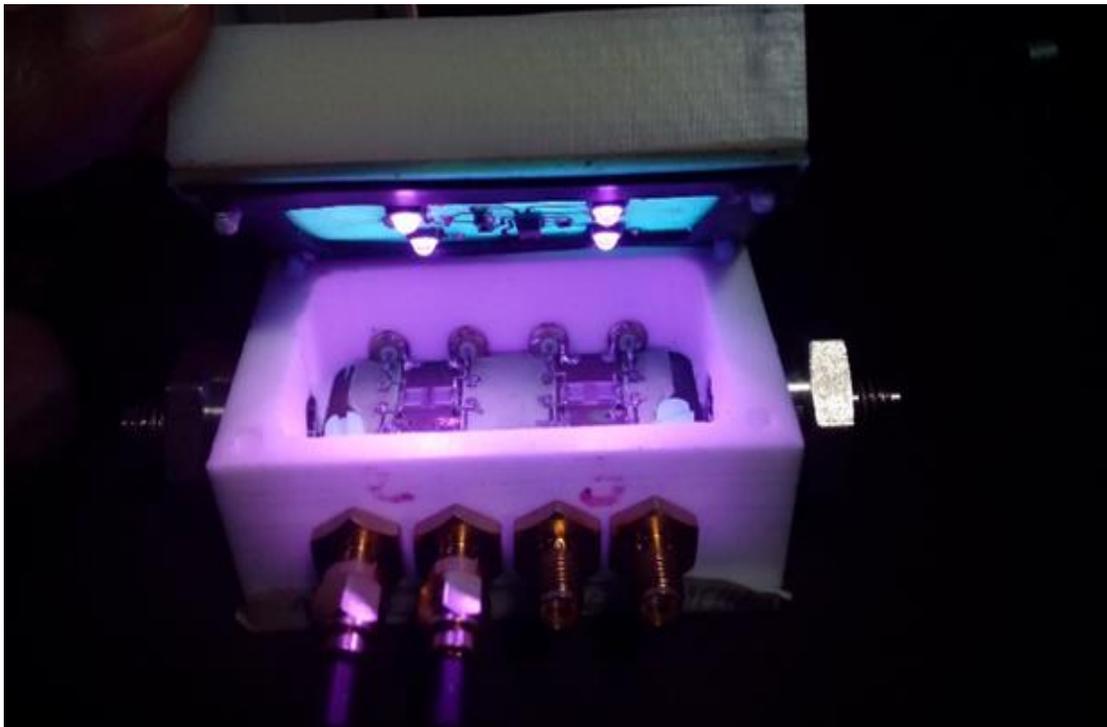


Figura 4.7. Matriz de sensores en funcionamiento.

4.2 Estructura del software.

En este apartado se van a describir los distintos programas y sistemas que fueron usados para desarrollar este proyecto. Todos los programas que están incluidos en el diagrama de flujo ilustrado en la figura 4.8 fueron desarrollados en MATLAB (*MATrix LABoratory*) versión 2010, el cual es considerado un software interactivo, que permite la ejecución del cálculo numérico de forma rápida y precisa, acompañado de características gráficas que facilitan la visualización de los datos obtenidos, y es particularmente práctico cuando el formato de datos esta dado en forma de matriz o vector [36]. Con este software es posible procesar una gran cantidad de series de datos, y aunque pueda parecer que sus funciones son básicas, estas pueden ampliarse de forma significativa usando cajas de herramientas, que realizan tareas específicas correspondientes a diversos campos de la ingeniería.

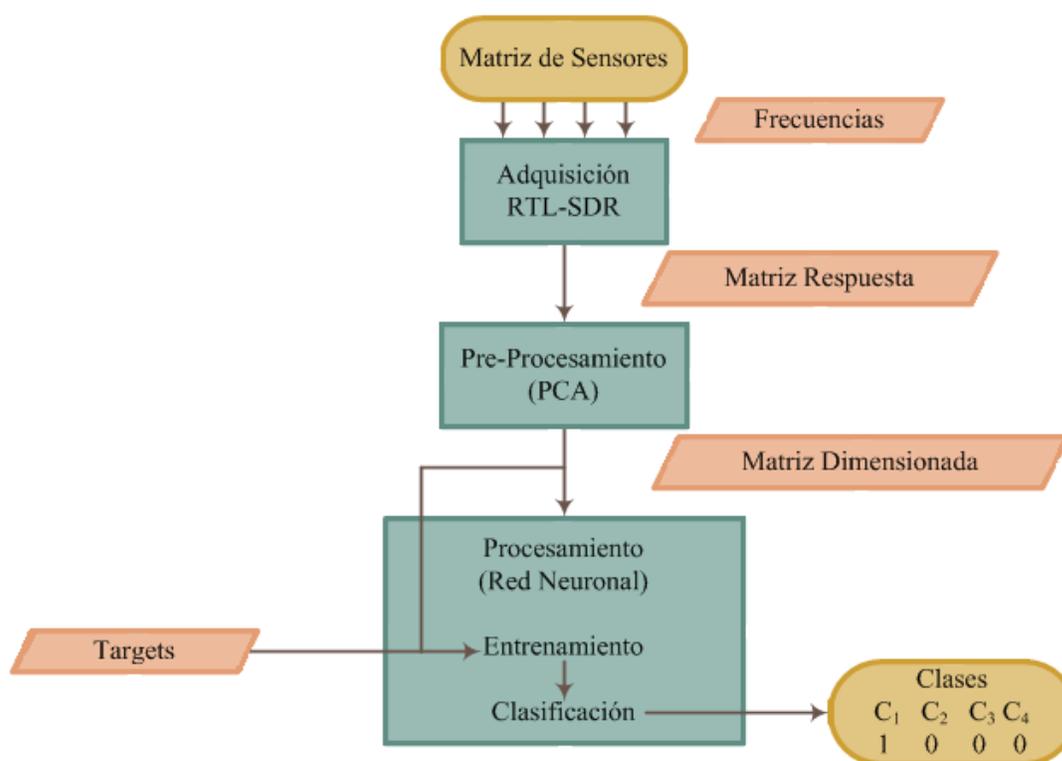


Figura 4.8. Diagrama del proceso general de la nariz electrónica.

MATLAB brinda un entorno de programación abierto, con un lenguaje de codificación basado en matrices, vectores y arrays. Gracias a que su arquitectura es abierta y

ampliamente extensible es factible establecer una relación con Excel, C, Fortran y diversas aplicaciones externas ampliamente utilizadas [25] e incluso el código escrito en el lenguaje de MATLAB puede ser traducido a C de forma inmediata. De igual forma, permite la operatividad entre plataformas posibilitando trabajar con distintos sistemas operativos y relacionar el trabajo realizado en las distintas plataformas e incluso es posible comunicarse a través de él con distintos dispositivos de hardware como la BBB y el RTL-SDR, además de crear interfaces de usuario de una forma sencilla.

Por todo lo descrito anteriormente, MATLAB se ha convertido en una herramienta muy útil tanto en el ambiente académico, como en el desarrollo tecnológico e investigación científica y es por esto que se decidió usarlo para crear los distintos programas que se encuentran involucrados desde la etapa de adquisición de datos hasta la clasificación de la muestra.

4.2.1 Adquisición de datos.

La estructura del programa se muestra en la figura 4.9.

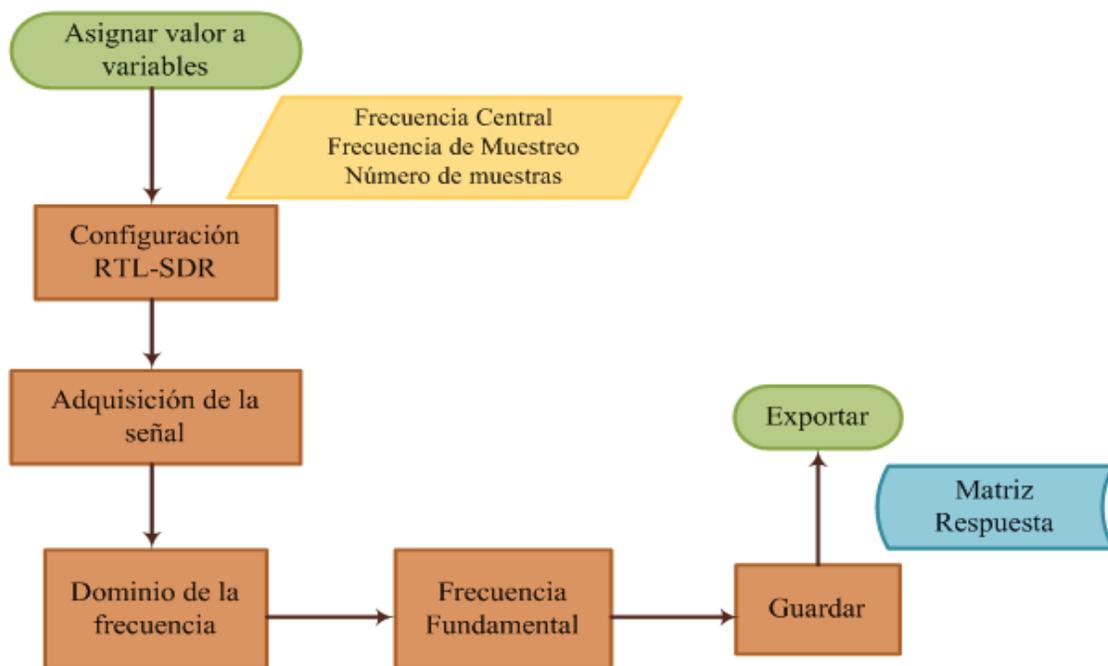


Figura 4.9. Diagrama del proceso de adquisición de datos.

Asignar valor a variables. Se asignan los valores con los que se configurará el dispositivo RTL-SDR, las variables a considerar son el número de muestras (N), la frecuencia de muestreo (F_s) y la frecuencia central (C_f).

Configuración RTL-SDR. Con MATLAB es posible configurar el RTL-SDR directamente con la ayuda del objeto *comm.SDRRTLReceiver*²², que se encuentra dentro del paquete *Support Package for RTL-SDR* del toolbox *Communications System*, el cual permite la interacción entre el dispositivo (RTL-SDR) y el software (Matlab), pero para esto es necesario instalar el soporte del paquete²³.

Adquisición de la Señal. Una vez configurado el dispositivo, se usa la función `step` para recibir datos de señal y control provenientes del RTL-SDR. La señal de salida mostrada en la figura 4.10, es una cadena de datos compleja de la cual solo se utiliza la parte real que representa la magnitud de la señal de salida ya que no interesa saber el desfase de la señal.

En este caso el dispositivo se configuró a una frecuencia de muestreo de 2.4MHz, una frecuencia central de 160MHz y el número de muestras igual a 4096. El valor de la frecuencia central varía dependiendo del compuesto volátil al que el sensor está expuesto.

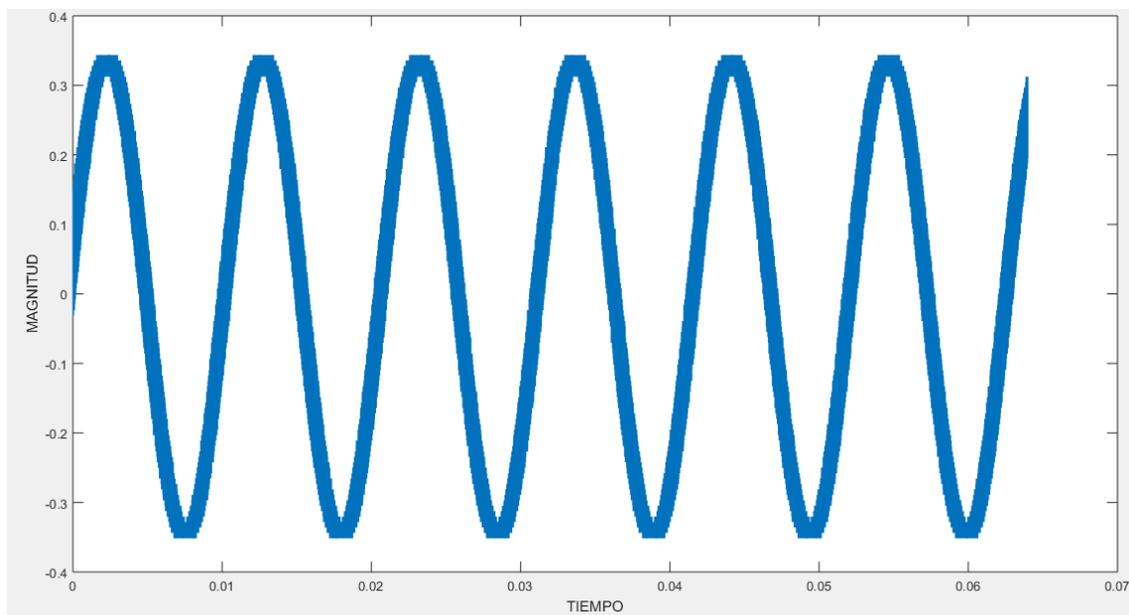


Figura 4.10. Digitalización de la señal de entrada.

²² Las propiedades de este objeto pueden consultarse en [37].

²³ La instalación del paquete puede verse en [38].

Dominio de la Frecuencia. Una vez extraídos los valores reales de la señal de salida, se guardan en una matriz dentro de una nueva variable, utilizando el comando FFT (Fast Fourier Transform) se transforma el dominio del tiempo al dominio de la frecuencia.

En la figura 4.11 se observa la señal de respuesta en el dominio de la frecuencia, en la gráfica se notan diferentes espigas, éstas espigas representan los armónicos que conforman la señal de entrada, sin embargo, solo se considera la mitad izquierda debido a que su salida incluye frecuencias negativas que existen puramente como propiedades matemáticas de la Transformada de Fourier, es decir la primera mitad de la salida de FFT contiene frecuencias desde 0 Hz hasta la frecuencia de Nyquist²⁴ en orden ascendente, la segunda mitad del arreglo es una imagen con frecuencias negativas²⁵.

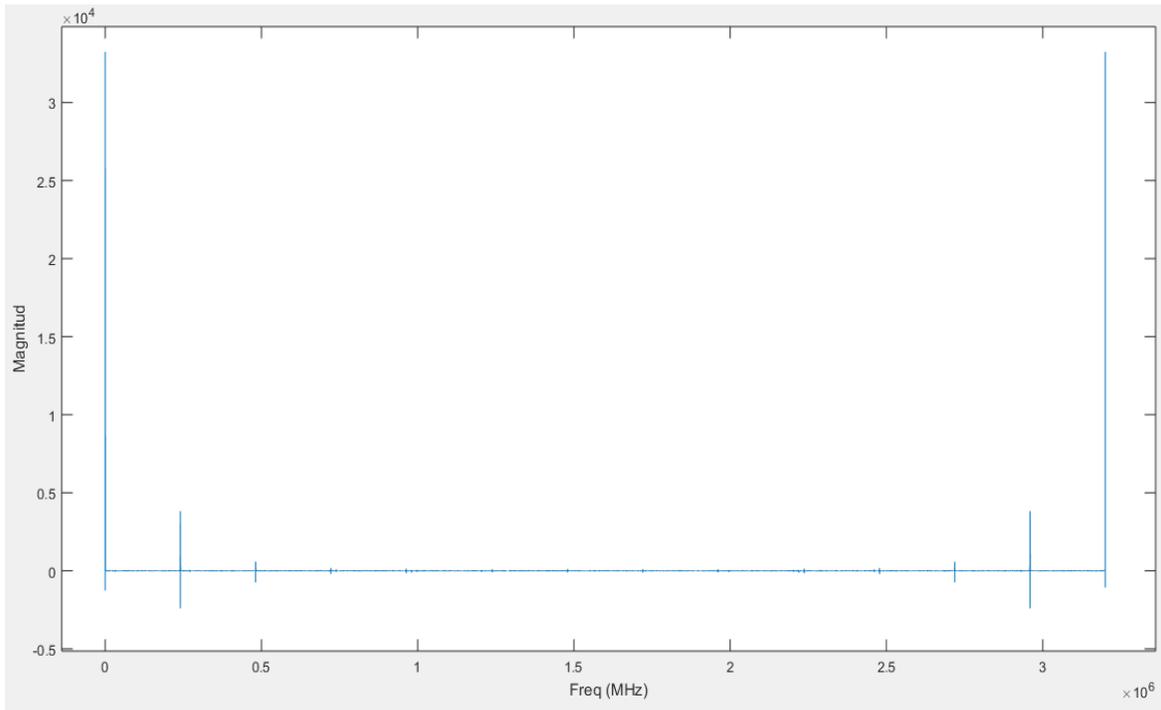


Figura 4.11. Señal en el dominio de la frecuencia

Frecuencia Fundamental. El RTL-SDR al ser un dispositivo receptor de ondas de radiofrecuencia tiene la posibilidad de trabajar con un rango de frecuencias menores o

²⁴ Sección 2.2.3 - Ecuación 2.5

²⁵ Efecto espejo

mayores a la frecuencia central que no excedan el ancho de banda permitido, para saber los límites inferior y superior de ancho de banda se realizan las siguientes operaciones:

$$LI = B - FC \quad (4.1)$$

$$LS = B + FC \quad (4.2)$$

Dónde B es el ancho de banda calculado en la ecuación 2.5 y FC es la frecuencia central que configuró el usuario, estos límites permiten trabajar con un umbral de ancho de banda más grande por lo que es viable hacer uso de un mayor número de frecuencias.

La frecuencia fundamental está representada por la espiga más grande observada en la figura 4.12, dicha espiga es el espectro de frecuencia del primer armónico. Teóricamente se obtiene haciendo una diferencia de la frecuencia central (FC) configurada del RTL-SDR, menos la frecuencia de la fuente (f_0).

$$\Delta f = FC - f_0 \quad (4.3)$$

Esto significa que si se trabaja con un valor de 160 MHz, tanto para la frecuencia central como para la frecuencia de fuente, al hacer la diferencia esta es idealmente 0 Hz, no obstante en la práctica esto es un tanto diferente. El valor de la frecuencia de la espiga más grande puede calcularse experimentalmente a partir de los siguientes parámetros:

$$\frac{\text{Punto de la espiga más grande}}{\text{Número Total de Datos}} \times \text{Frecuencia de Muestreo} \quad (4.4)$$

$$\frac{7}{204800} \times 3.2 \text{MHz}$$

$$\text{Frecuencia de la espiga} = 109.375 \text{Hz}$$

Cuando se hace la digitalización, solo se obtiene el número de puntos que representan la reconstrucción de la señal, por lo que se debe transformar el dato donde se encuentra la espiga a frecuencia, para este propósito se usa la frecuencia de muestreo.

Al comparar la frecuencia de la espiga obtenida prácticamente con el valor ideal, la primera resulta ser considerablemente mayor, no obstante, es un valor aceptable debido a que no está ni cerca de sobrepasar el límite de ancho de banda o frecuencia de Nyquist²⁶.

²⁶ Sección 2.2.3 - Frecuencia de Nyquist = 1.6MHz

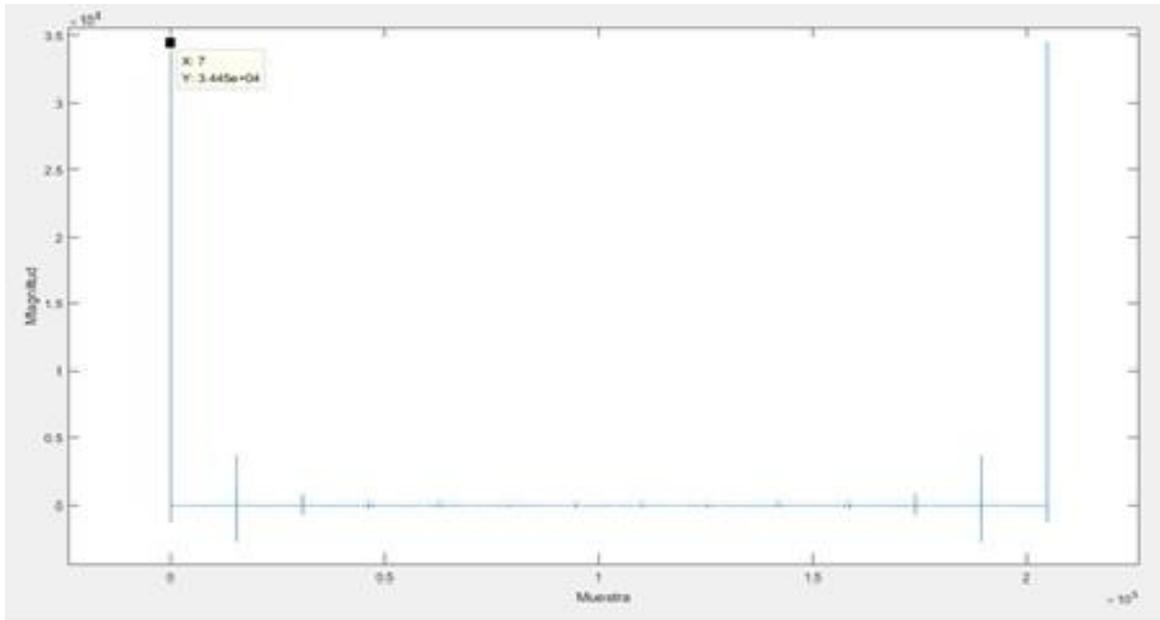


Figura 4.12. Punto de la espiga más grande.

El comportamiento de la señal en el dominio de la frecuencia varía dependiendo si se modifica la frecuencia de la fuente o la frecuencia central. Por ejemplo, si se usa una frecuencia de fuente de 158.5MHz y se mantiene la frecuencia central a 160MHz, las espigas se comportan como se ilustra en la figura 4.13.

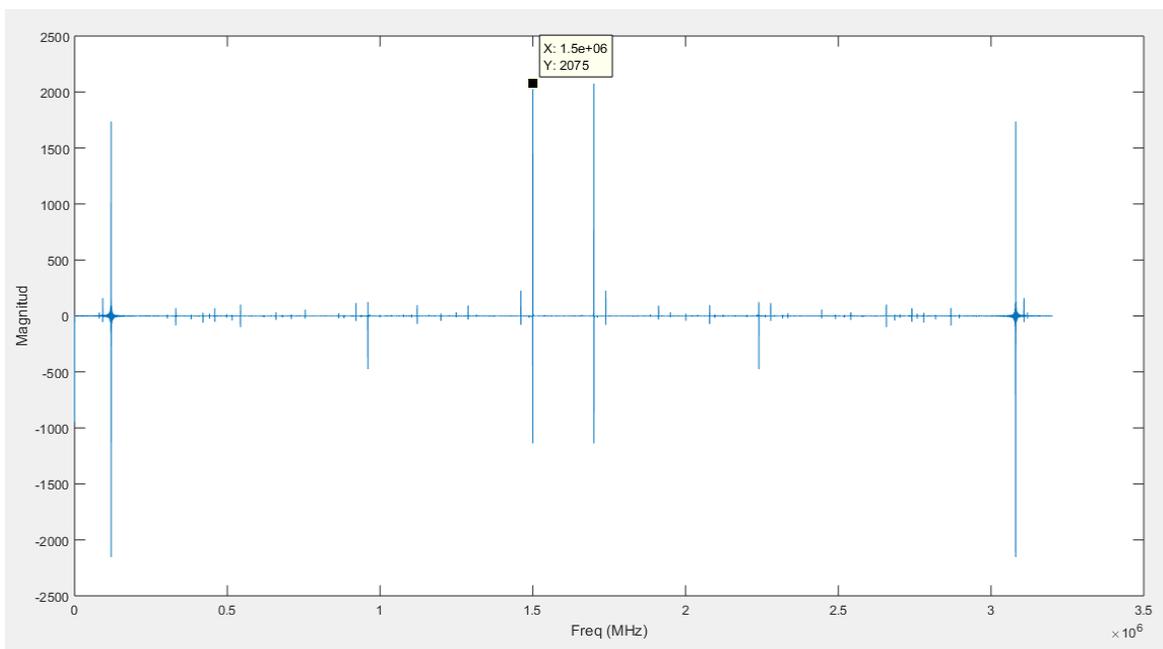


Figura 4.13. Señal respuesta obtenida con una frecuencia de fuente de 158.5 MHz.

La frecuencia resultante de la espiga al hacer la diferencia es de 1.5 MHz, frecuencia que no sobrepasa el límite permitido, por lo cual aún es visible en la gráfica.

Dependiendo si se cambia la frecuencia de fuente o la frecuencia central, entonces la señal en el dominio de la frecuencia tiende a comportarse tal como se muestra en la figura 4.14.

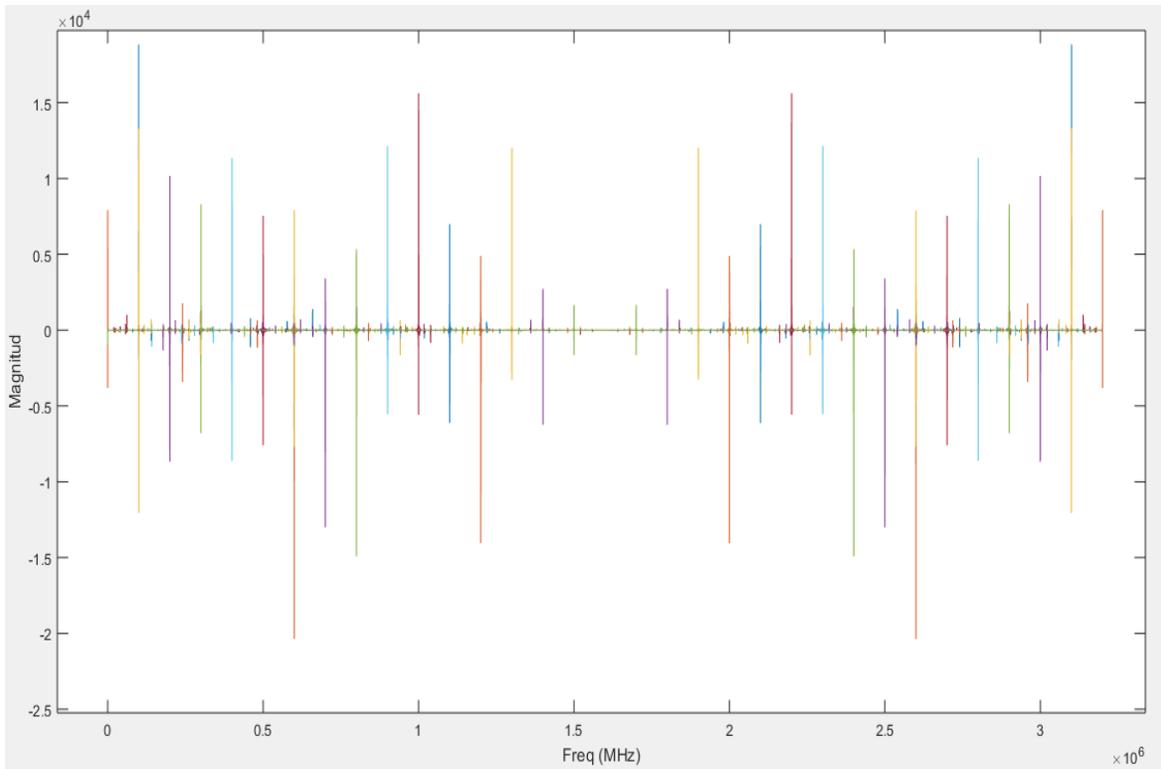


Figura 4.14. Respuesta para diferentes frecuencias de fuente.

En este caso, la frecuencia central se mantuvo a 160 MHz y la frecuencia de fuente se fue variando desde 158.4 MHz hasta 160 MHz con un incremento de 0.1 MHz para el límite inferior y desde 160 MHz hasta 161.6 MHz con un incremento de 0.1 MHz para el límite superior. El comportamiento de las espigas se debe a que entre más cerca se encuentre la frecuencia de la fuente al límite del ancho de banda las espigas se acercan más entre ellas, pero si la frecuencia de la fuente se acerca al valor de la frecuencia central las espigas se alejan²⁷.

²⁷ Efecto espejo explicado en la sección 4.1 en el apartado del *Dominio de la frecuencia*

Guardar y Exportar: Para tener una lectura continua de la frecuencia, se estableció un ciclo que es definido por el usuario para poder capturar los cambios de frecuencia que se presentan en la interacción de los sensores con los compuestos que se están midiendo. Una vez concluido el ciclo se guardan los valores en un vector respuesta para su posterior exportación a un archivo de hoja de cálculo.

Como se explicó en la sección 2.2.3, la frecuencia fundamental da una visualización del comportamiento del sensor, una vez que se exportan los valores de frecuencia en forma matricial, se revisa la señal resultante obtenida de las medidas realizadas por el sensor.

En la figura 4.15 se muestra el comportamiento de un sensor expuesto a Acetona de 20 ppm, aplicando la ecuación 2.13 para saber la frecuencia de cambio.

$$\Delta f = 810.987318 \text{ Hz} - 39.1927669 \text{ Hz}$$

$$\Delta f = 771.7945511 \text{ Hz}$$

El valor calculado de la frecuencia de cambio se guarda en una matriz denominada “matriz respuesta” que se usará posteriormente para el desarrollo de la reducción de variables

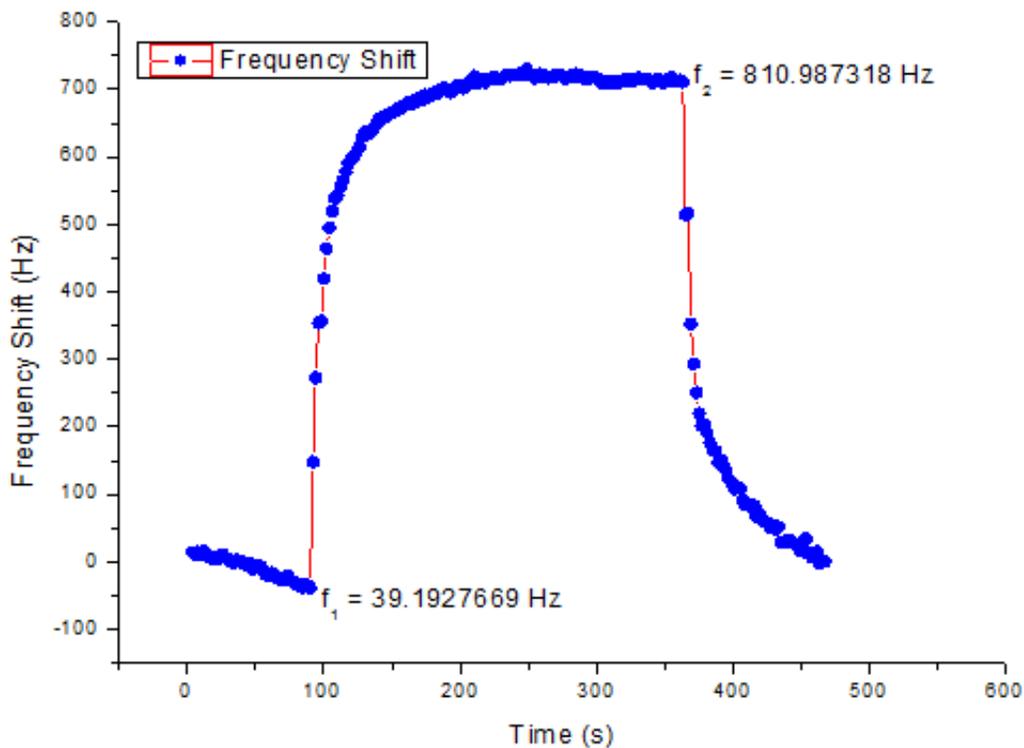


Figura 4.15. Respuesta de un sensor expuesto a acetona.

4.2.2 Reducción de variables (PCA).

Las subrutinas básicas que se usan para desarrollar un PCA en MATLAB se encuentran dentro de un paquete multivariante denominado PLS-Toolbox, sin embargo, ha sido necesario programar una serie de funciones auxiliares, ya que las funciones que contiene dicho paquete son muy generales y no se ajustan al objetivo de este proyecto. La estructura del PCA se puede observar en la figura 4.16.

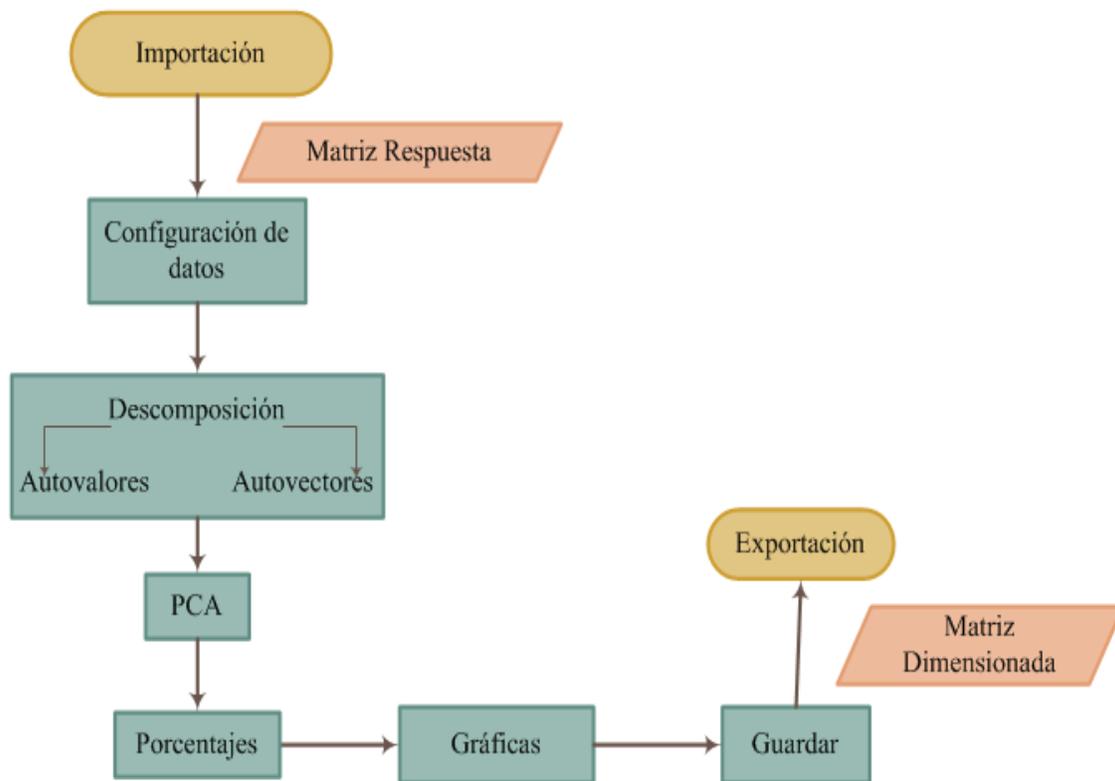


Figura 4.16. Diagrama de flujo de la reducción de variables.

Importación: En este primer paso se transporta la matriz respuesta de un archivo de hoja de cálculo a MATLAB usando el comando `xlsread`.

Configuración de datos: Para que el método funcione correctamente, es necesario establecer una referencia que permita al programa reconocer el número de lecturas que realizó un sensor, acto seguido se normalizan los datos con la técnica *escalado estándar*, que ayuda a evitar redundancia en los datos.

$$nor = \frac{\text{Matriz Respuesta} - \text{Promedio de la Matriz}}{\text{Desviación Estandar de la Matriz}} \quad (4.5)$$

Una vez normalizados, se procede a calcular la matriz de covarianza con el comando `cov`.

Descomposición: Al obtener la matriz de covarianza, se calculan los *autovalores* y *autovectores*. Usando el comando `svd`²⁸ se obtienen dos matrices, una es la matriz diagonal que contiene los autovalores ordenados de mayor a menor, y la segunda matriz contiene los autovectores o mejor conocidos como los coeficientes de las componentes principales.

PCA: A partir de una matriz X, que contiene los datos originales y aplicando el proceso visto en el apartado 3.1.1, se calculan las tres primeras componentes principales.

Porcentajes: Para calcular el porcentaje de información de cada una de las componentes principales se usan los autovalores de la tabla 4.1 contenidos en la matriz diagonal.

Tabla 4.1. Autovalores de la matriz diagonal.

403.0328	0	0	0	0	0
0	100.6229	0	0	0	0
0	0	75.9416	0	0	0
0	0	0	25.9455	0	0
0	0	0	0	17.0478	0
0	0	0	0	0	15.1940

Los porcentajes se calculan con la siguiente ecuación:

$$\% = \frac{\text{Valor de Autovalor}}{\text{Sumatoria de los Autovalores}} \times 100 \quad (4.6)$$

Gráficas: En el primer plano representado en la figura 4.17, se grafican los datos con mayor semejanza (varianza). Pero antes de poder representar las PC's es necesario hacer una serie de cálculos que involucran la técnica de puntajes estandarizados z ²⁹ y la regresión lineal. En la figura 4.18 se muestran los pesos de los coeficientes de cada componente principal.

²⁸ Singular Value Decomposition

²⁹ `zscore` – Esta función devuelve la matriz original de tal forma que sus datos estén centrados por columnas para tener una media igual a 0 y además escalada para que su desviación estándar sea igual a 1.

Guardar y Exportar: Por último, se guarda en una hoja de cálculo la matriz de componentes principales resultante de la sección PCA.

4.2.3 Clasificador (ANN).

La figura 4.19 ilustra los pasos que se siguieron para construir el modelo del clasificador usando redes neuronales.

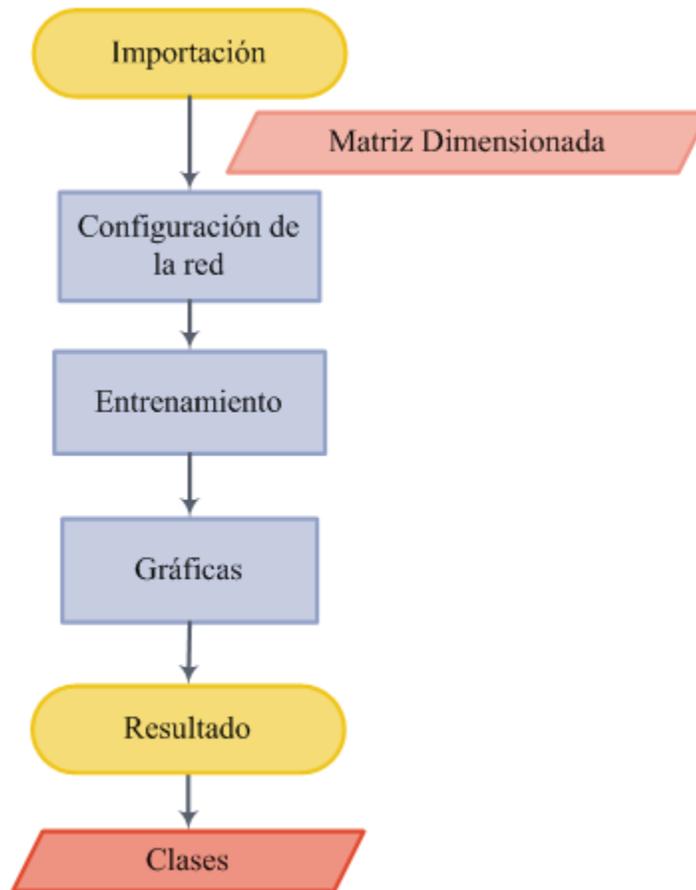


Figura 4.19. Diagrama de flujo del clasificador.

Importación. Se asignan los datos que fungirán como entradas a la red y se definen los objetivos que usará la red para asignar las clases.

Configuración de la Red. Para configurar la red se usa la siguiente metodología:

- ❖ Con el comando `newff` se crea una red *feedforward*. Al crear la red es necesario asignarle los datos de entrada, los targets y el número de neuronas que la conformarán. Después se establecen las conexiones entre las diferentes capas, los

vectores de entrada y salida, y el número de neuronas tanto de la capa oculta como de la capa de salida.

- ❖ Cada tipo de red tiene asignada una función de activación por defecto pero, en este caso se modificó esta característica usando `net.layers{i}.transferFcn`³⁰. A la capa oculta se le asignó la función *tangente hiperbólica* (`tansig`) y a la capa de salida la función *exponencial normalizada* (`softmax`).
- ❖ Es importante tomarse un tiempo para verificar las propiedades asociadas a cada capa ya que de esto depende que la red neuronal funcione como se espera.
- ❖ Se definen los pesos y el umbral de activación³¹ con la propiedad `net.biasConnect`, si los valores de entrada netos superan un cierto umbral la neurona devolverá una señal de activación positiva, por ejemplo “1”, y si no lo superan devolverá una señal negativa, como por ejemplo “0” o “-1”. Con la propiedad `net.initFcn = 'initlay'` se indica que inicie los pesos y el umbral de activación establecidos por defecto en el software.
- ❖ Se asigna la función de control³² a *entropía cruzada* (`crossentropy`) y la función de entrenamiento a *gradiente conjugado escalado* (`trainscg`).
- ❖ Se dividen los datos para establecer los conjuntos de entrenamiento (`trainRatio`), validación (`valRatio`) y testeo (`testRatio`).
- ❖ Por último se inicia la red.

Entrenamiento: El entrenamiento de la red consta de los siguientes pasos:

- ❖ Normalizar los valores
- ❖ Establecer el número de iteraciones (`net.trainParam.epochs`), la tasa de aprendizaje (`net.trainParam.lr`) y el factor de momento (`net.trainParam.mc`).
- ❖ Configurar la visualización de la ventana de entrenamiento, la cual se representa en la figura 4.20.
- ❖ Entrenar la red `[net, tr] = train(net, Entradas, Targets);`
- ❖ Revisar el número de iteraciones realizada por el entrenamiento y el tiempo que se tardó en realizar todo el proceso de entrenamiento, prueba y validación.

³⁰ Donde *i* representa el número de capa de la red.

³¹ El umbral de activación es conocido como bias.

³² Que es la encargada del conjunto de validación.

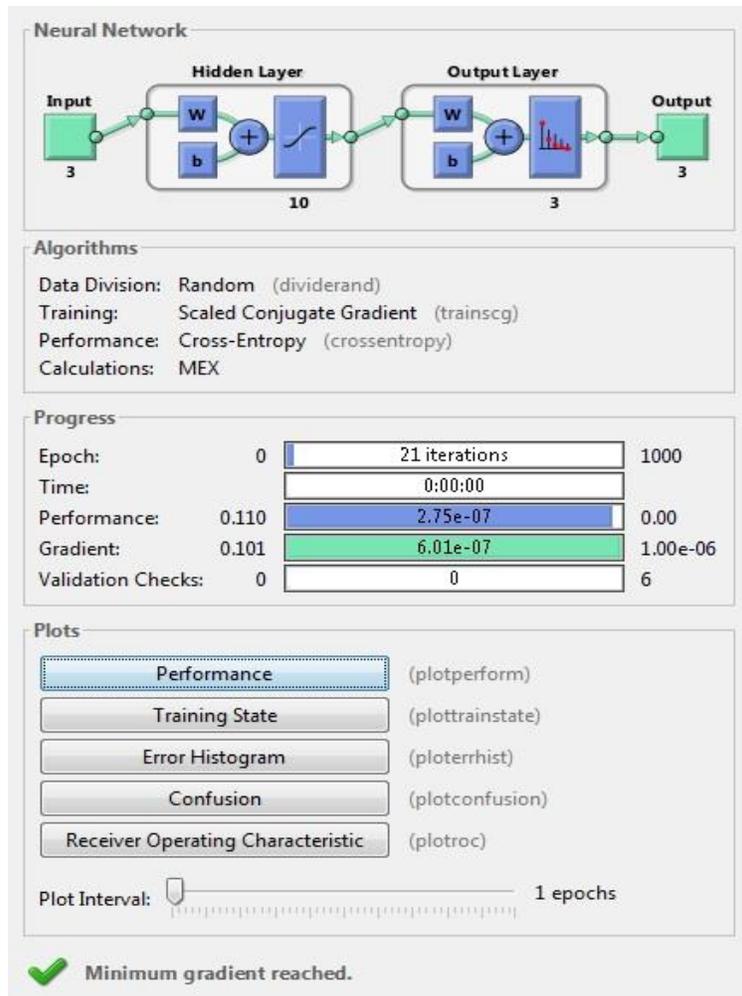


Figura 4.20. Ventana de entrenamiento.

Gráficas: MATLAB arroja diversas gráficas, sin embargo, son de mayor interés tres de estas. En la primera gráfica representada en la figura 4.21 se visualiza el rendimiento de la red en el entrenamiento, validación y prueba (`plotperform`), en la figura 4.22 se observa la segunda gráfica que indica el ajuste entre los valores estimados y predichos (`plotregression`), y la figura 4.23 muestra la tercera gráfica, la cual señala a que clase los datos fueron asignados, y el número de porcentaje de acierto y error en la clasificación (`plotconfusion`)³³. A partir de los resultados observados en estas gráficas el usuario decidirá si es necesario volver a entrenar la red.

³³ La visualización de las gráficas se puede configurar directamente en la sección de configuración de la red o por separado tecleando los comandos.

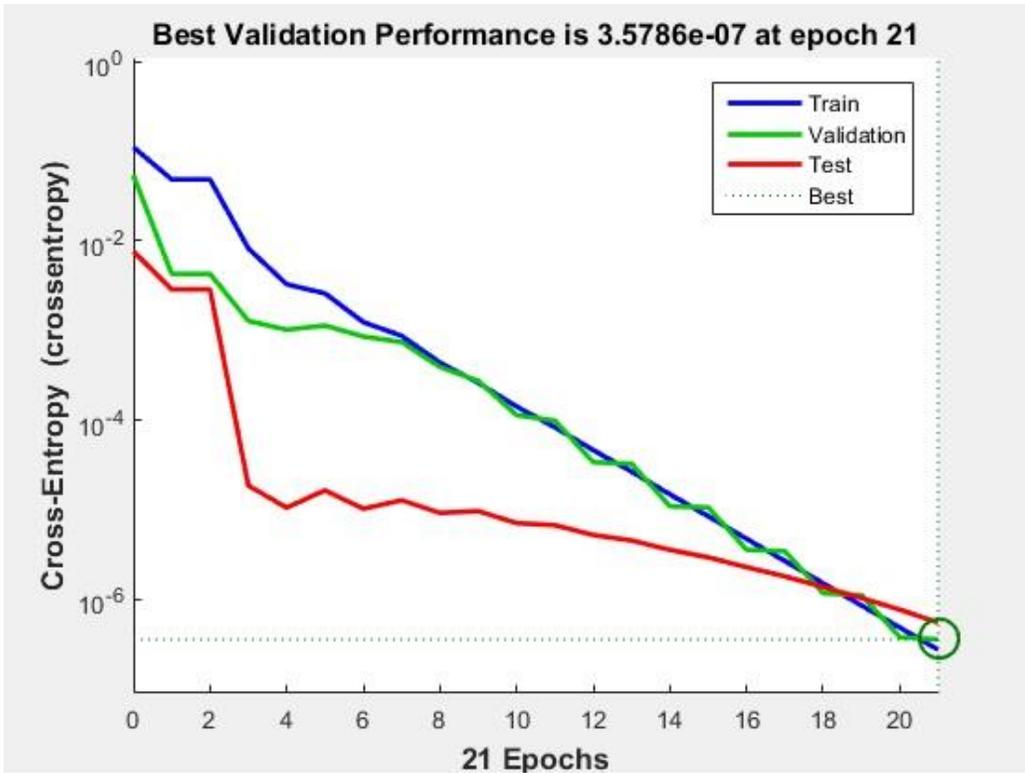


Figura 4.21. Gráfica del rendimiento de la red en sus diferentes etapas de aprendizaje.

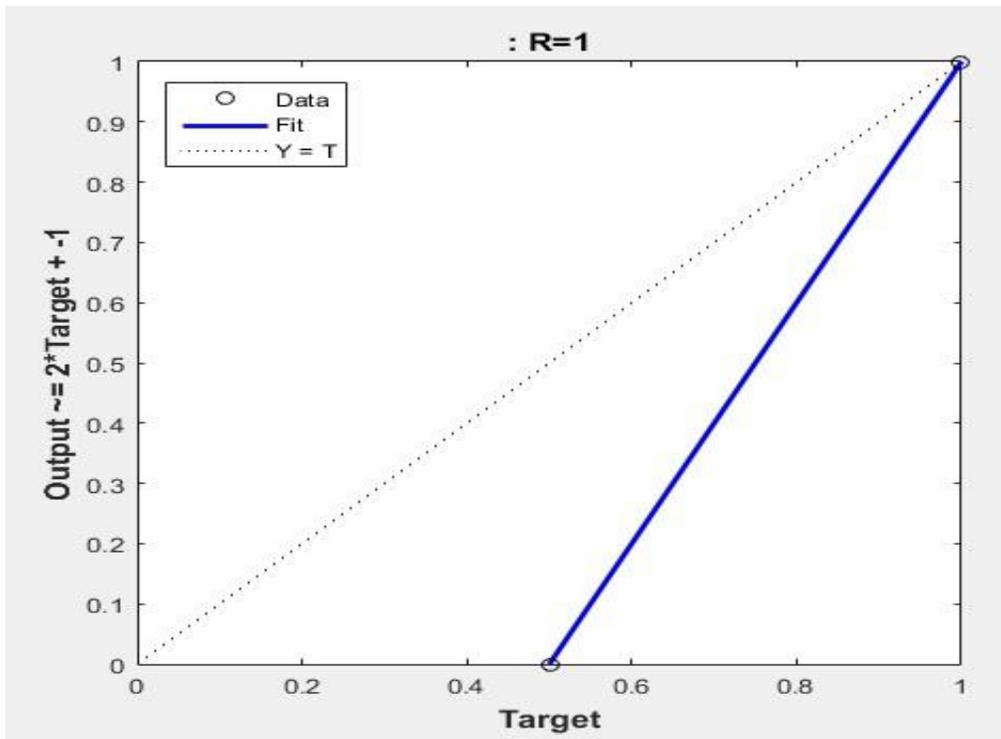


Figura 4.22. Gráfica del ajuste entre los valores estimados y predichos.

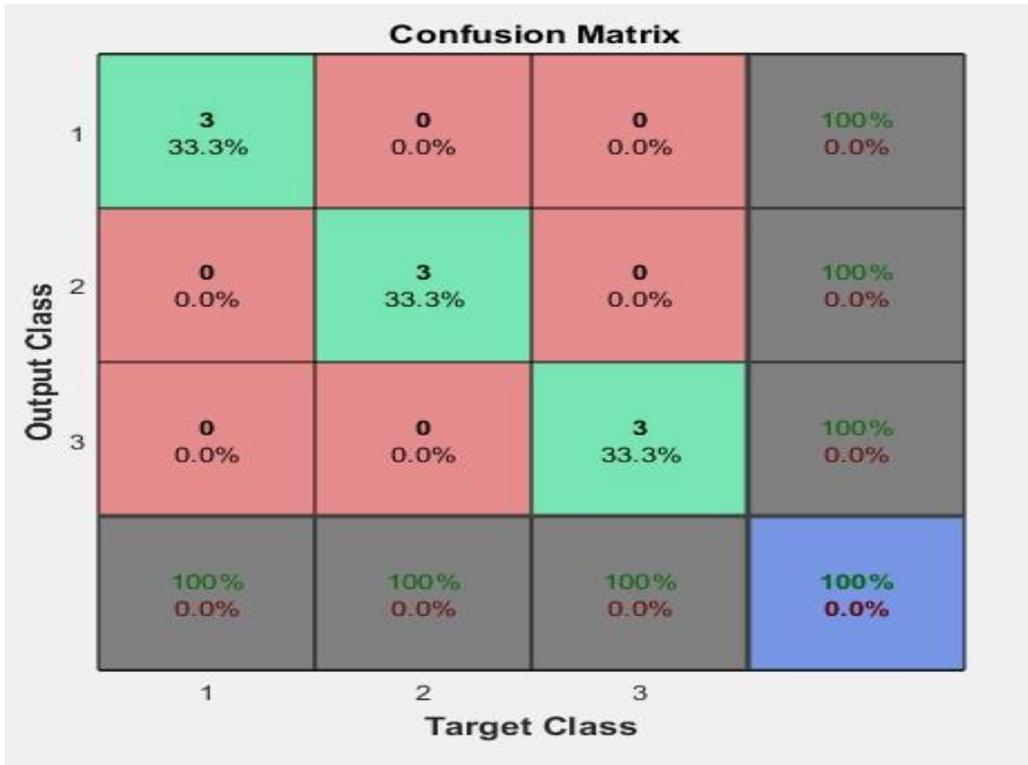


Figura 4.23. Gráfica de la matriz de porcentajes de acierto y error en la clasificación.

Resultado: Una vez concluido el entrenamiento, la red clasifica los datos, tal como se muestra en la figura 4.24.

```

Número de iteraciones: 21
El proceso ha tardado 12.36 segundos
salida =

1.0000  1.0000  1.0000  0.5000  0.5000  0.5000  0.5000  0.5000  0.5000
0.5000  0.5000  0.5000  1.0000  1.0000  1.0000  0.5000  0.5000  0.5000
0.5000  0.5000  0.5000  0.5000  0.5000  0.5000  1.0000  1.0000  1.0000

classes =

1  1  1  2  2  2  3  3  3

```

Figura 4.24. Asignación de las clases.

4.3 Interfaz gráfica de usuario.

El cómo se presentan al usuario los resultados que se obtienen al ejecutar los programas es de suma importancia, ya que la persona que use el software debe visualizar la información de una forma sencilla y ordenada. La interfaz, al igual que el resto del software, se diseñó en MATLAB que en versiones recientes incluyó la posibilidad de usar un editor especializado denominado “GUIDE”³⁴. La interfaz gráfica está dividida en cuatro ventanas:

- ❖ **Inicio.** En esta ventana ilustrada en la figura 4.25, se da la bienvenida al usuario y se ofrece la opción de poder escoger con que etapa del proceso se desea trabajar.



Figura 4.25. Ventana de inicio.

- ❖ **Adquisición.** En la figura 4.26 se visualiza la ventana de la primera etapa en la que se recibe la frecuencia de la fuente, se guarda la frecuencia de la armónica fundamental y se construye la matriz respuesta.

³⁴ En [39] se muestra una guía para poder crear una interfaz gráfica.

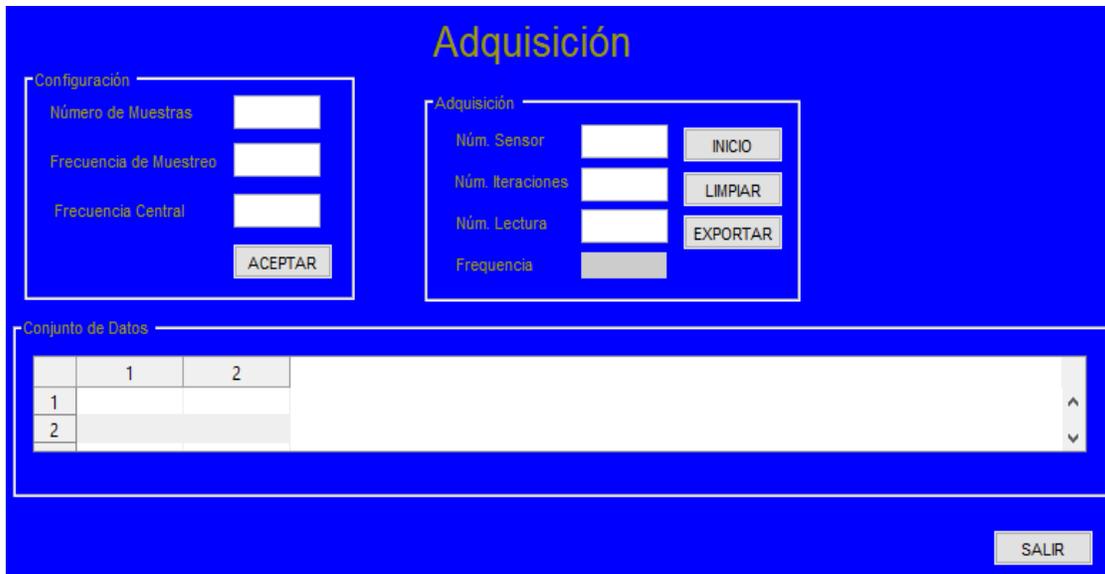


Figura 4.26. Ventana de adquisición de datos.

Lo primero que se hace, es realizar la configuración del RTL-SDR asignando los valores para el número de muestras, frecuencia de muestreo y frecuencia central, como se observa en la figura 4.27.

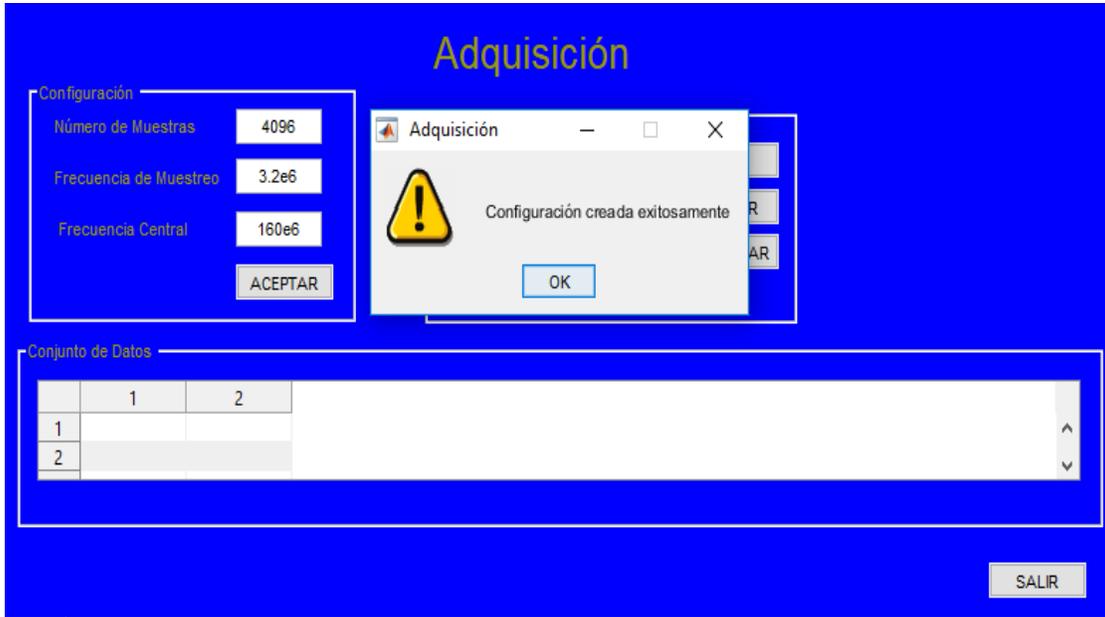


Figura 4.27. Configuración del RTL-SDR

Para iniciar la adquisición se debe introducir el número del sensor con el que se dese trabajar, así como las veces que se quiere realizar la adquisición y el número de

lectura. Cuando se inicie el programa aparecerá la frecuencia fundamental de cada lectura y en una tabla se mostrará cada una de las frecuencias previamente tomadas, tal como se ilustra en la figura 4.28.

La figura 4.29 muestra cómo una vez que las iteraciones concluyeron, solo resta exportar los valores obtenidos a una hoja de cálculo.

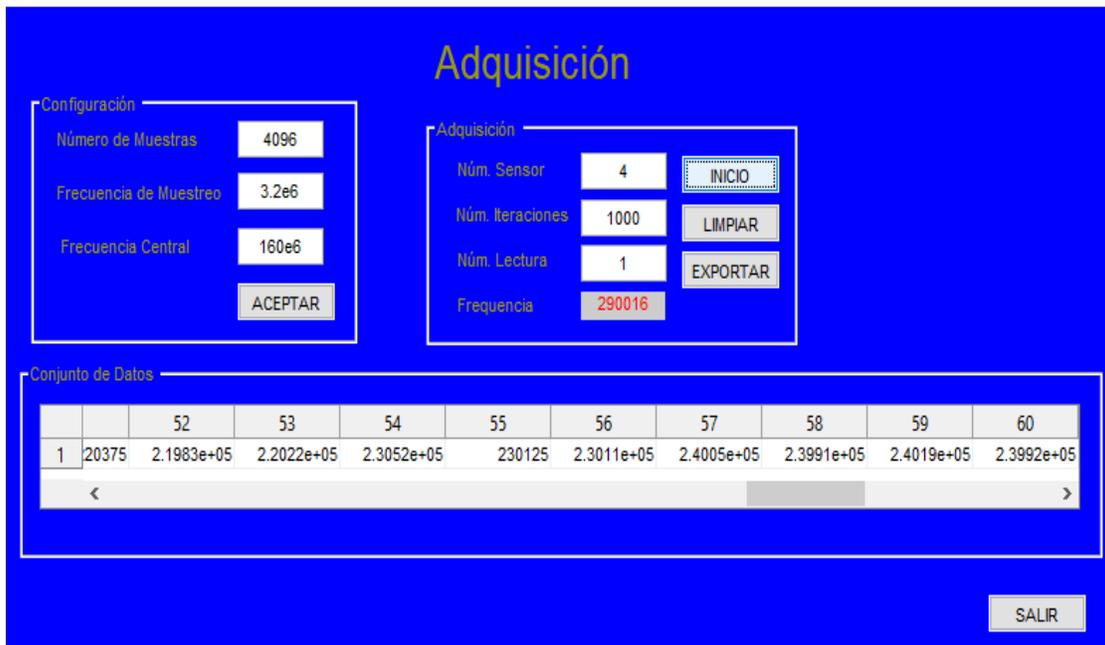


Figura 4.28. Adquisición de la señal.



Figura 4.29. Exportación de datos.

❖ **Pre-Procesamiento.** En la figura 4.30 se representa el inicio de la segunda etapa, que es reducir las variables y descartar la información redundante, empleando el PCA.



Figura 4.30. Ventana de reducción de variables.

Primero se importan los datos que se vayan a optimizar, ya sea del archivo previamente guardado de la etapa anterior, o en su defecto cualquier otra hoja de cálculo, tal como se observa en la figura 4.31.



Figura 4.31. Importación de la matriz respuesta.

Por último, en la figura 4.34 se puede ver como son exportados los nuevos datos obtenidos del PCA.

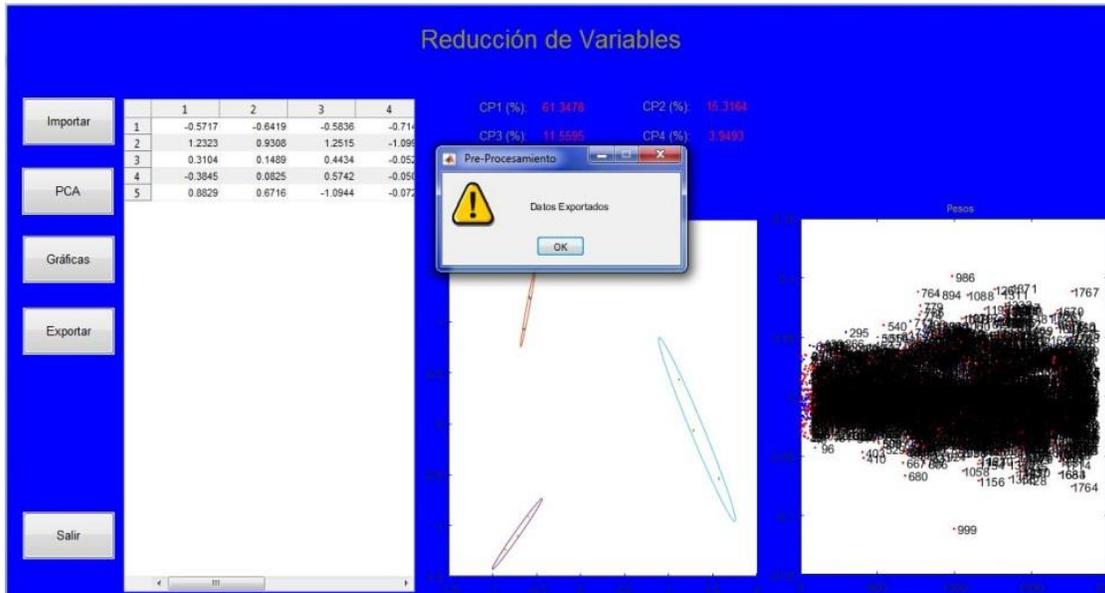


Figura 4.34. Exportación de las PC's.

❖ **Procesamiento.** La figura 4.35 representa la ventana de procesamiento, en la que se encuentra todo lo relativo a la red neuronal y a la clasificación de los datos que nos proporcionó el PCA.



Figura 4.35. Ventana del proceso de clasificación.

En la figura 4.36 se ilustra el proceso de importación de los datos que serán clasificados, una vez ejecutado el programa se apreciarán en la ventana el número de muestras y sensores usados, además de los objetivos asignados.



Figura 4.36. Importación de datos.

En la figura 4.37 se observa la creación de la red, pero para iniciar la red es necesario proporcionar el número de neuronas tanto para la capa oculta como para la capa de salida.



Figura 4.37. Creación de la red neuronal.

Al iniciar con el entrenamiento, en la ventana representada por la figura 4.38 se muestra el número de iteraciones y el tiempo en el que realizó el entrenamiento.



Figura 4.38. Entrenamiento de la red.

Para analizar de forma gráfica el rendimiento de la red se escoge cualquiera de las tres gráficas mostradas en las figuras 4.39, 4.40 y 4.41.

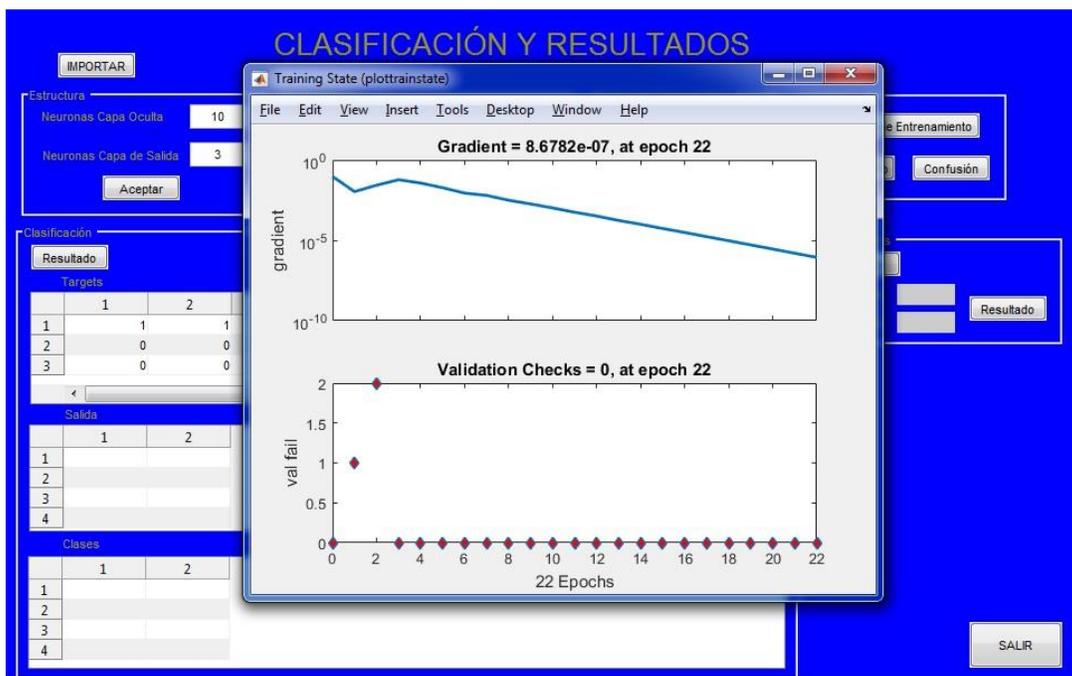


Figura 4.39. Grafica del estado de entrenamiento.

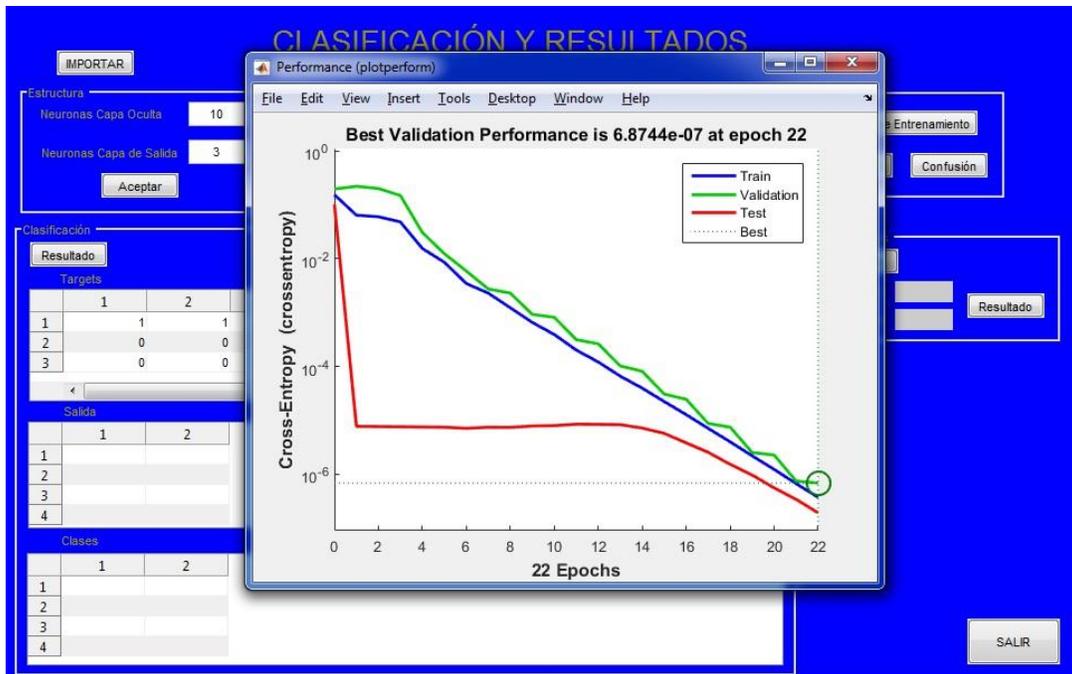


Figura 4.40. Grafica del rendimiento de la red en las tres etapas de aprendizaje.

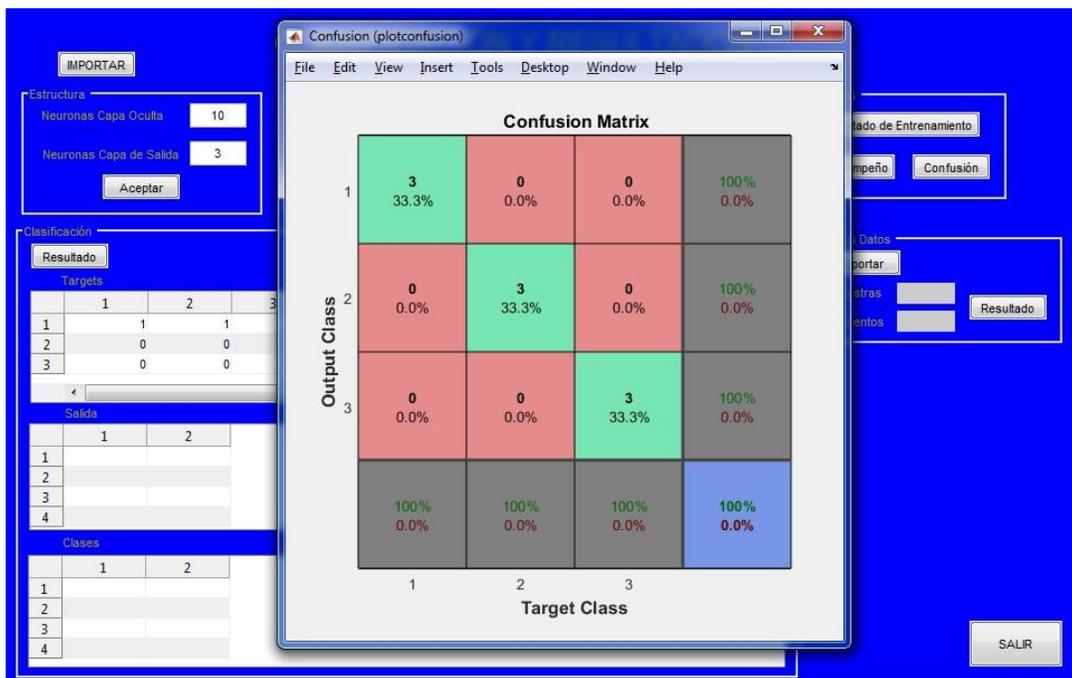


Figura 4.41. Grafica de la matriz de confusión.

Finalmente en la figura 4.42 se ilustran las salidas calculadas y las clases en las cuales fueron asignadas.



Figura 4.42. Ventana de resultados.

En la figura 4.43 se puede ver que existe la posibilidad de exportar otros datos, sin la necesidad de volver a entrenar la red, simplemente se da clic en botón importar, que se encuentra en la sección de nuevos datos, posteriormente los resultados obtenidos con los nuevos datos podrán observarse en la ventana, como se muestra en la figura 4.44.



Figura 4.43. Exportación de nuevos datos.



Figura 4.44. Ventana de resultados para nuevos datos.

El sistema en funcionamiento, desde la interacción de la cámara de sensores con el RTL-SDR y la BeagleBone Black hasta la interfaz gráfica de usuario, se observa en la figura 4.45.

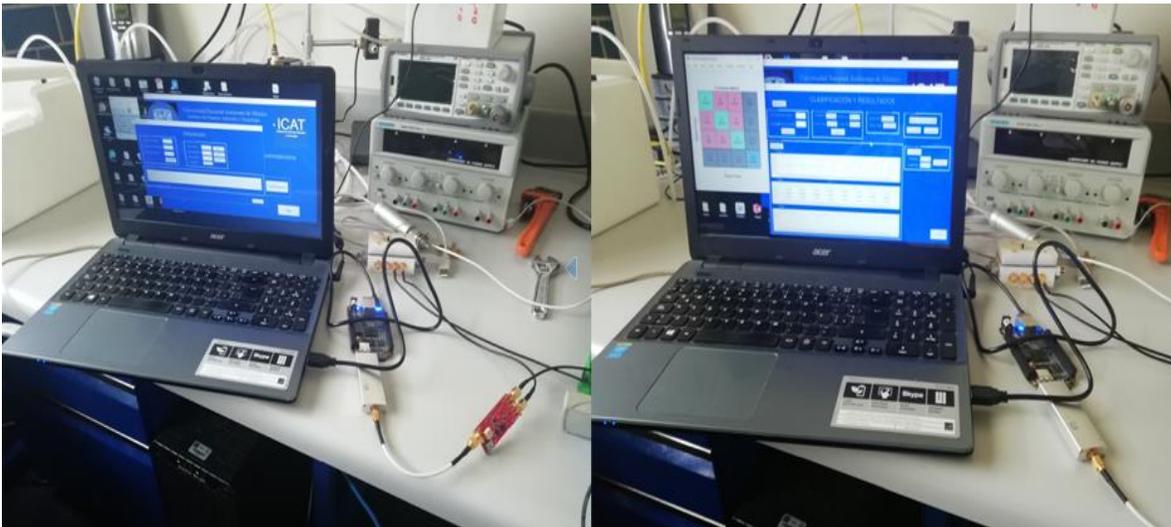


Figura 4.45. Sistema Finalizado.

Conclusiones

A lo largo de este proyecto de tesis hemos visto que la detección de enfermedades a través del aliento es una técnica efectiva de diagnóstico pero, a pesar de que existen diversos sistemas para dicho fin, su uso se ve limitado por varios factores, entre ellos los siguientes: la elaborada preparación que requiere la muestra, la compleja instrumentación que los acompaña; lo que implica que para ser operados es necesario contar con personal técnicamente entrenado, el costo elevado y en muchos de los casos los sistemas no son portátiles.

Debido a los factores antes mencionados, en los últimos años se ha buscado implementar dispositivos de estado sólido que sean capaces de mejorar la eficiencia del proceso de la detección de compuestos volátiles, así como disminuir tanto sus dimensiones como su costo.

Entre los sensores de estado sólido más prometedores encontramos los SAW, para los cuales, en el presente trabajo se ha diseñado una celda completamente sellada donde se puede medir la muestra deseada, además de poder llevar a cabo la ionización de sus capas sensibles para poder ver el cambio en su respuesta.

Una finalidad de este trabajo fue desarrollar un sistema que fuese más sencillo de manipular por los usuarios. Por esta razón se decidió emplear una computadora de placa única (BeagleBone Black), que además de reducir sustancialmente la carga y el tiempo de procesamiento y obtener mediciones en tiempo real, también nos permitió obtener un sistema SAW de detección portátil acoplado un sistema de adquisición de datos de bajo costo basados en un RTL.

Los datos en los que nos basamos para construir nuestro sistema, fueron tomados de un experimento similar al nuestro, que se llevó a cabo previamente dentro del laboratorio de sensores biomédicos, sin embargo, con la información que utilizamos podemos afirmar que los sistemas basados en sensores de estado sólido que utilizan arreglos matriciales de sensores pueden discriminar y clasificar especies, lo que permite generar un patrón de respuesta estrechamente ligado a la selectividad que posee determinado sensor hacia los diferentes compuestos orgánicos volátiles. Gracias a este patrón de respuesta es posible clasificar o identificar los compuestos usando métodos de reconocimiento de patrones, así

como un método de reducción de variables para eliminar los datos redundantes que pudieran afectar el procesamiento y el resultado del sistema, en este proyecto se optó por usar el método de componentes principales, el cual nos permitió obtener los datos más relevantes de cada compuesto orgánico expuesto en los sensores, sin pérdidas importantes de información. Para entrenar la red neuronal backpropagation usamos las tres componentes principales, obteniendo en la mayoría de los casos una correcta clasificación, que demuestra una efectiva separación y una óptima selectividad de los sensores.

Todo trabajo de investigación contribuye a la formulación de preguntas y de nuevas ideas que pueden expandir el alcance de sistemas como el que presentamos en este proyecto, las bases que planteamos en esta tesis muestran solo un inicio para desarrollar herramientas que permitirán diagnosticar afecciones que afecten al cuerpo humano.

Una propuesta interesante podría ser utilizar otro tipo de arreglo en los sensores con la finalidad de optimizar la respuesta de los mismos. La evolución de la tecnología nos presenta la posibilidad de usar diversos dispositivos, como microprocesadores, microcontroladores, placas únicas, FPGA (*Field Programmable Gate Array-Matriz de Puertas Programables*), que no solo se emplearían para mejorar el procesamiento de datos en tiempo real o la portabilidad del sistema, sino que a través de ellos podríamos integrar complementos como LCD (*Liquid Cristal Display –Pantalla de Cristal Líquido*) o pantallas táctiles, con los cuales seríamos capaces de ejecutar la interfaz de usuario directamente dentro del dispositivo elegido y con ello evitaríamos tener que conectar una PC para interactuar con los programas. Podría resultar beneficioso agregar módulos de conexión a la red para poder exportar la información a una base de datos y subirla a la nube para su posterior consulta.

Incluso valdría la pena probar distintos métodos de reducción de variables, así como de otras técnicas de reconocimiento de patrones para optimizar la clasificación y además saber que sensor o conjuntos de sensores tiene más peso en cada diagnóstico.

Bibliografía

- [1] Dr. Jonas Nunes. (2017) Instituto del aliento. [Online]. <http://www.institutodelaliento.com>
- [2] Philip Bartlett Julian W. Gardner, "A brief history of electronic noses," *Sensors and Actuators B: Chemical*, vol. XVIII, no. 1-3, pp. 210-211, 1994.
- [3] Cristhian Manuel Durán Acevedo, "El qué, cómo y cuándo de los sistemas de olfato electrónico," in *Diseño y optimización de los subsistemas de un sistema de olfato electrónico para aplicaciones agroalimentarias.*, pp. 2-25.
- [4] J.M. Elliott, J.W. Gardner P.N. Bartlett, "Electronic noses and their applications in the food industry," *Food Technology*, vol. XII, no. 58, pp. 44-48, 1997.
- [5] Daniel Matatagui Cruz, *Sensores Másicos para la Detección de agentes de Guerra Química y Biológica*. Madrid: UAM (Universidad Autónoma de Madrid, 2012.
- [6] Fundación Española para la Ciencia y la Tecnología (FECYT), "Materiales Piezoeléctricos," *Boletín VT (Vigilancia Tecnológica)*, vol. IV, no. 3, pp. 2-9, 2010.
- [7] Griselda Stephany Abarca Jiménez, Jesús Mares Carreño Leonel Germán Corona Ramírez, *Sensores y Actuadores: Aplicaciones con Arduino*. México, D.F.: Grupo Editorial PATRIA, 2014.
- [8] Antonio Arnau Vives, *Piezoelectric Transducers and Applications*. Berlin-Heidelberg: Springer , 2008.
- [9] Mario De Miguel Ramos, *High Sensitivity Biosensors Based on Shear Mode*. Madrid: Universidad Politécnica de Madrid, 2015.
- [10] M. I. Rocha-Gaso, *Analysis, implementation and validation of a love mode surface acoustic wave device for its application as sensor of biological processes in liquid media*. Valencia: Universidad Politécnica de Valencia, 2013.
- [11] Andrés José Piñón Pazos, *Implementación de un nuevo método de caracterización de biosensores basados en balanza de cristal de cuarzo (QCM) mediante instrumentos virtuales multipropósito*. España: Universidade da Coruña, 2014.
- [12] Josep M. Trigo-Rodríguez Mar Tapia, "¿En qué se parecen un bólido y un terremoto?: Procesos geofísicos asociados a las caídas de meteoritos," *Enseñanzas de las Ciencias de la Tierra*, vol. III, no. 23, pp. 243-253, 2013.
- [13] Alejandro Erasmo Ortega, "Diseño y construcción de un oscilador con base en ondas acústicas superficiales ," Universidad Nacional Autónoma de México , CDMX, Tesis de Maestría 2018.
- [14] Fernando Gabino Ramírez Neyra, *Sistema de Adquisición de Datos de Sensores Analógicos y*

Digitales, Primera ed., ESIME, Ed. México D.F., México: IPN, 2009.

- [15] Manuel Ortega Cantero, *Informática Educativa: Realidad y Futuro*, Primera ed., Ed. Madrid, Castilla-La Mancha: Servicio de Publicaciones de la Universidad de Castilla-La Mancha, 1995.
- [16] Alberto Daniel Picasso González, *Análisis Prueba y Utilización de una Tarjeta de Desarrollo con Procesador ARM CórteX A8*, Primera ed. Cuautitlán Izcali, Edo. México: UNAM, 2016.
- [17] Comunidad RTL-SDR. (2013, Octubre) RTL-SDR.com. [Online]. www.rtl-sdr.com/about-rtl-sdr/
- [18] Carl Laufer, *The hobbyist's guide to the RTL-SDR really cheap software defined radio.*, Primera ed., 2014.
- [19] Juan Ramón Solís Escobedo, "Diseño de un sistema de radio definido por software sobre plataformas ARM y "front-end" analógico básico para comunicaciones de baja velocidad entre boyas sobre la banda HF," Centro de Investigación Científica y de Educación Superior de Ensenada, Ensenada, Baja California, Tesis de Maestría 2016.
- [20] Martin Carrizo, "Fourier y el procesamiento digital de señales," Universidad Nacional del Sur, Argentina, Conferencia 2014.
- [21] Lucas Montané, "Diferentes aplicaciones de Fourier en el procesamiento digital del sonido," Universidad Nacional del Sur, Argentina, Informe 2013.
- [22] G. Figueroa M., E.Carrera R., and A. Jiménez R., "Análisis de Componentes Principales en Paralelo," Instituto Tecnológico de Costa Rica, Costa Rica, Proyecto de fin de curso 2012.
- [23] David Spiegelhalter, Charles Taylor Donald Michie. (1994, Febrero) Machine Learning, Neural and Statistical Classification - Bad Request. [Online]. <https://www1.maths.leeds.ac.uk/~charles/statlog/whole.pdf>
- [24] Julián Zapotitla Román, "Análisis de Componentes principales," Universidad Nacional Autónoma de México, Ciudad de México, 2011.
- [25] I.T. Jolliffe, *Principal Component Analysis*, Segunda ed. New York, U.S.A: Springer, 2002.
- [26] Pedro López Roldán and Sandra Fachellii, *Metodología de la investigación social*. Barcelona, España: Universidad Autónoma de Barcelona, 2015.
- [27] M. en I.S.C. Francisco Jacob Ávila Camacho, M. en I.S.C. Juan Manuel Stein Carrillo, and M.A.N. Adolfo Melendez Ramírez. (2013, Septiembre) ResearchGate. [Online]. <https://www.researchgate.net/publication/259977213>
- [28] Amazon Machine Learning. (2018) Clasificación binaria. [Online]. https://docs.aws.amazon.com/es_es/machine-learning/latest/dg/binary-classification.html
- [29] Google Developers. (2018, Marzo) Curso intensivo de aprendizaje automático. [Online]. <https://developers.google.com/machine-learning/crash-course/classification/true-false->

[positive-negative?hl=es-419](#)

- [30] Simon Haykin, *Neural Networks a Comprehensive Foundation*, Segunda ed. Delhi, India: Prentice Hall, 2001.
- [31] Fernando Mateo Jiménez, "Redes Neuronales y Preprocesado de Variables para Modelos y Sensores en Bioingeniería," Universidad Politécnica de Valencia, Valencia, Tesis Doctoral 2012.
- [32] Advanced Tech Computing Group UTPL. (2007, Septiembre) Elementos básicos de una red neuronal parte II. [Online]. <https://advancedtech.wordpress.com/2007/09/26/elementos-baiscos-de-una-red-neuronal-artificialparte-ii/>
- [33] Alan Delgado. Redes Neuronales. [Online]. <http://disi.unal.edu.co/~lctorress/RedNeu/LiRna008.pdf>
- [34] Paul E. Keller, "Overview of electronic nose algorithms," Pacific Northwest National Laboratory, Conference paper PNNL-SA-31428, Marzo 2015.
- [35] André Quinquis, *Digital Signal Processing Using MATLAB*, Primera ed., André Quinquis, Ed. Great Britain: ISTE, 2008.
- [36] César Pérez López, *MATLAB y sus aplicaciones en las ciencias y la ingeniería*, Primera ed., Isabel Capella, Ed. Madrid, España: Prentice Hall, 2002.
- [37] The MathWorks Inc. (1994-2018) comm.SDRRTLReceiver System object. [Online]. <https://www.mathworks.com/help/supportpkg/rtlsdrradio/ug/comm.sdrrtlreceiver-system-object.html>
- [38] The MathWorks Inc. Installation and Setup. [Online]. <https://www.mathworks.com/help/supportpkg/rtlsdrradio/installation-and-setup.html>
- [39] The Mathworks Inc. GUI de MATLAB. [Online]. <https://es.mathworks.com/discovery/matlab-gui.html>

A.2 Datasheet led uv.



365nm UV LED Gen 2 Emitter

LZ1-00UV00



Key Features

- 365nm UV LED with 1200mW flux output at 2.7W power dissipation
- Ultra-small foot print – 4.4mm x 4.4mm
- Highest Radiant Flux density
- Surface mount ceramic package with integrated glass lens
- Very low Thermal Resistance (4.2°C/W)
- JEDEC Level 1 for Moisture Sensitivity Level
- Lead (Pb) free and RoHS compliant
- Reflow solderable (up to 6 cycles)
- Emitter available on Standard or Miniature MCPCB (optional)

Typical Applications

- Curing
- Printing
- PCB Exposure
- Sterilization
- Medical
- Currency Verification
- Fluorescence Microscopy
- Inspection of dyes, rodent and animal contamination
- Forensics

Description

The LZ1-00UV00 UV LED emitter provides superior radiometric power in the wavelength range specifically required for applications like curing, printing, sterilization, currency verification, and various medical applications. With a 4.4mm x 4.4mm ultra-small footprint, this package provides exceptional optical power density. The patented design has unparalleled thermal and optical performance. The high quality materials used in the package are chosen to optimize light output, have excellent UV resistance, and minimize stresses which results in monumental reliability and radiant flux maintenance.



COPYRIGHT © 2018 LED ENGIN. ALL RIGHTS RESERVED. LZ1-00UV00 (3-D-00)/11/18

LED Engin | 801 River Oaks Parkway | San Jose, CA 95134 USA | ph +1 408 922 7200 | fax +1 408 922 0158 | em sales@ledengin.com | www.ledengin.com

Absolute Maximum Ratings

Table 4:

Parameter	Symbol	Value	Unit
DC Forward Current ⁽¹⁾	I_F	1000	mA
Peak Pulsed Forward Current ⁽²⁾	I_{FP}	1000	mA
Reverse Voltage	V_R	See Note 3	V
Storage Temperature	T_{stg}	-40 ^o +150	°C
Junction Temperature	T_J	115	°C
Soldering Temperature ⁽³⁾	T_{sol}	260	°C
Allowable Reflow Cycles		8	
ESD Sensitivity ⁽⁴⁾		> 2,000 V HBM Class 2 IEC61010-1	

Notes for Table 4:

- Maximum DC forward current is determined by the overall thermal resistance and ambient temperature. Follow the curves in Figure 11 for current derating.
- Pulse forward current conditions: Pulse Width < 10ms; and Duty Cycle < 10%.
- LEDs are not designed to be reverse biased.
- Solder conditions per JEDEC J203. See Reflow Soldering Profile Figure 3.
- LED Engin recommends taking reasonable precautions towards possible ESD damages and handling the LI-001000 in an electrostatic protected area (EPA). An EPA may be adequately protected by ESD controls as outlined in ANSI/ESD S6.1.

Optical Characteristics @ $T_C = 25^\circ\text{C}$

Table 5:

Parameter	Symbol	Typical	Unit
Radiant Flux (@ $I_F = 700\text{mA}$)	Φ	1200	mW
Radiant Flux (@ $I_F = 1000\text{mA}$)	Φ	1680	mW
Peak Wavelength ⁽¹⁾	λ_p	565	nm
Viewing Angle ⁽²⁾	$2\theta_{1/2}$	70	Degrees
Total Included Angle ⁽³⁾	θ_{tot}	105	Degrees

Notes for Table 5:

- When operating the UV LED, observe safety precaution given in EC61471 Risk Group 1. Avoid eye and skin exposure to unshielded product.
- Viewing Angle is the off-axis angle from emitter centerline where the radiometric power is 1/2 of the peak value.
- Total Included Angle is the total angle that includes 80% of the total radiant flux.

Electrical Characteristics @ $T_C = 25^\circ\text{C}$

Table 6:

Parameter	Symbol	Typical	Unit
Forward Voltage (@ $I_F = 700\text{mA}$)	V_F	5.8	V
Temperature Coefficient of Forward Voltage	$\Delta V_F/\Delta T_J$	-1.3	mV/°C
Thermal Resistance (Junction to Case)	$R\theta_{JC}$	4.2	°C/W

A.3 Conector SMA

SMA Bulkhead Jack Front Mount



Specifications:

Electrical

Frequency Range	: DC-18GHz
Nominal Impedance	: 50Ω
Working Voltage Max	: 500V
Proof Voltage	: 1,000V
VSWR Max	: 1.3

Materials

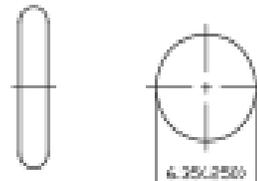
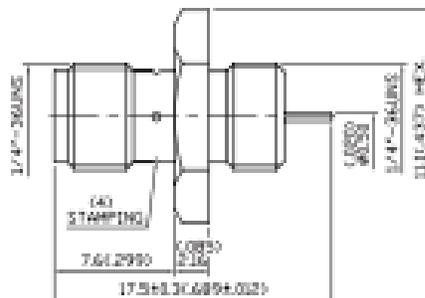
Body	: Stainless Steel per JIS SUS 303
Center Contact	: Beryllium Copper per ASTM B195 UNS C173000
Insulation	: PTFE per ASTM D1457 and ASTM D1710

Mechanical

Interface in Accordance with	: MIL-STD-348A
------------------------------	----------------

Environmental

Operating Temperature Range	: -65°C to +165°C
Connector must withstand Waterproof Lead Test Pressure 3kg/Cm ² Min.	



Mounting Hole

Dimensions : Millimetres (Inches)

Finish and Others

No	Description	Material	Qty	Finish
1	Body	Stainless Steel	1	Gold 3µ" Min.
2	Insulation	PTFE	1	None
3	Centre Contact	Beryllium Copper	1	Gold 50µ" Min.
4	O-Ring	Silicone	1	Red (26.D7 ± 1.78)

Part Number Table

Description	Part Number
Jack, SMA, Bulkhead, Front Mount	H19-016-10-0032210LL

Important Notice : This data sheet and its contents (the "Information") belong to the members of the Premier Farnell group of companies (the "Group") or are licensed to it. No license is granted for the use of it other than for information purposes in connection with the products to which it relates. No license of any intellectual property rights is granted. The Information is subject to change without notice and replaces all data sheets previously supplied. The Information supplied is believed to be accurate but the Group assumes no responsibility for its accuracy or completeness, any error in or omission from it or for any use made of it. Users of this data sheet should check for themselves the Information and the suitability of the products for their purpose and not make any assumptions based on information included or omitted. Liability for loss or damage resulting from any reliance on the Information or use of it (including liability resulting from negligence or where the Group was aware of the possibility of such loss or damage arising) is excluded. This will not operate to limit or restrict the Group's liability for death or personal injury resulting from its negligence. Multicomp is the registered trademark of the Group. © Premier Farnell plc 2012

www.element14.com
www.farnell.com
www.newark.com



A.4 Sensor de humedad y temperatura ChipCap® 2

TELAIRE

ChipCap® 2

A Fully Calibrated Humidity and Temperature Sensor

Features

- Fully Calibrated & Temperature Compensated
- Digital or Analog Output with Alarm Function
- Precision & Accuracy ($\pm 2\%$ RH, $\pm 0.3^\circ\text{C}$, 14 bit)
- Free Operating Voltage (min 2.7V to max 5.5V)
- Low Current Consumption
- SMD Package for Automated Assembly
- Reliable in Harsh Environments

Applications

- Energy Saving HVAC Control – Air Conditioning, Refrigeration, Indoor Air Quality, Vent Fans, Home Appliances, Humi/Dehumidifiers
- Process Control & Instrumentation – Medical Instruments, Handheld Devices, Weather Stations, Food Processing, Printers, RFIDs



- Automobile & Transportation – Cabin Climate Control, Defogging Control, Condensing Preventive Device
- Medical – Nebulizers, Oxygen air, CPAP/ Sleep Apnea devices
- OEM assemblies available

ChipCap® 2 offers the most advanced and cost effective humidity and temperature sensing solution for virtually any type of application.

A capacitive polymer sensor chip and a CMOS integrated circuit with EEPROM are integrated into one embedded system in a reflow solderable SMD package.

Individually calibrated and tested, ChipCap 2 performs at $\pm 2\%$ from 20% to 80% RH ($\pm 3\%$ over entire humidity range), and is simple and ready to use without further calibration or temperature compensation.

ChipCap 2 provides linear output signals in various interfaces to customer requirements:

- I²C Interface
- PDM convertible to analog signal
- Alarm function for preset control at min/ max humidity

Amphenol
Advanced Sensors

Sensor Performance

Relative Humidity (RH%)

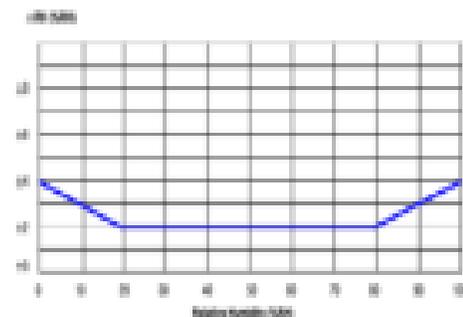
Resolution	1.6 bit (0.015625%)
Accuracy ¹	±0.0156% (0.00015625%)
Repeatability	±0.0156%
Hysteresis	±0.0156%
Linearity	±0.0156%
Response time ²	7.0 sec (40%)
Temp. Coefficient	Max 0.13156%/°C (at 10-80°C, 10-80%RH)
Operating Range	0 - 100% RH (Non-Condensing)
Long Term Drift	±0.156%/yr (Normal condition)

1. Custom Accuracy Tolerance Available

2. Accuracies measured at 25°C, 5.0V

3. Measured at 25°C. Inflow airflow for achieving 40% of step from 25%RH to 80%RH

Typical %RH Accuracy



Electrical Specifications

Supply Voltage

min 2.3V to max. 5.5V

Supply Current (IDD)

750 μ A (typical)

Sleep Current (I_{sleep})

0.6 μ A (typical)

Environmental

Operating Temperature Range

-40 to 125°C

Operating RH Range

0 to 100 % RH (non-condensing)

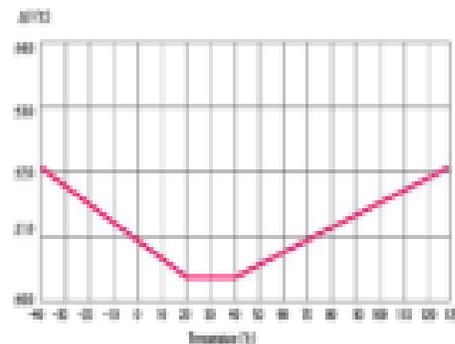
Temperature (°C)

Resolution	1.6 bit (0.07°C)
Accuracy ¹	±0.3°C
Repeatability	±0.1°C
Response time ²	5.0 sec (40%)
Operating range	-40 to 125°C
Long term drift	±0.007°C/yr (Normal condition)

1. Accuracies measured at 25°C, 5.0V

2. Min 5.0 sec, Max 20 sec

Typical Temperature Accuracy



Absolute Maximum Rating

Parameter	Min	Max
Supply Voltage (VDD)	-0.3V	6.0V
Storage Temp (T _{stg})	-55°C	150°C
Junction Temp (T _j)	-55°C	150°C

Soldering Information

Standard or IR Solder Reflow

T_p: 240°C, t_p: 40 sec. (quality Pb free profile)

Note: After soldering reconditioning will be required. Details for this process can be found in the ChipCap[®] 2 application note (918-127).

Package Contents

Capacitive polymer RH Sensor, PTA (Proportional to Absolute) Temperature sensor integrated ASIC chip in LCC (Leadless Chip Carrier) package, SMD, RoHS compliant

A.5 Código Matlab para la adquisición con el RTL-SDR

```
Fs = 2.4e6; %Frecuencia de Muestreo "Sample Rate" (Mínima 250e3 / Máxima
3.2e6)
N = 4096; %Número de Muestras "SamplesPerFrame"
dt = 1/Fs; %Intervalo de tiempo
Cf = 160e6;
%j = 1;
fre_final=[];
fr_fi = [];

rxsdr =
comm.SDRRTLReceiver('0','CenterFrequency',Cf,'SampleRate',Fs,'SamplesPerF
rame',N,'EnableTunerAGC',true,'OutputDataType','single');
infostruct = info(rxsdr);

Y =step(rxsdr);

for (j = 1:200)
    Y_final = [];
    Freq = [];
    newFr = 0;

    for (i = 1:100)
        Y = step(rxsdr);
        Y_final = [Y_final;Y];

        a = real(Y_final);
        [ma,na] = size(a);

        aT = a';
        YA = fft(aT,ma);% Transformada de Fourier
        RYA = real(abs(YA));

        F = max(RYA);
        Fr = find(RYA==F);
        newFr = newFr + Fr;
    end

    Fr = newFr / 100;
    Fre = Fr(:,1);
    FreqMax = (Fre/ma)*Fs
    FrMx(j) = [fr_fi;FreqMax];
    %j
    %release(rxsdr)
end

% %Gráficas
% %Dominio del Tiempo
%     T = [0:dt:dt*((ma-1))];
%     figure(1);
%     plot(T,a);
%     xlabel('Tiempo')
%     ylabel('Amplitud')
%
```

```

% %Dominio de la Frecuencia
% [mr,nr] = size(RYA);
% for (h = 1:nr)
%   FreMa = (h/ma)*Fs;
%   fre_final=[fre_final;FreMa];
% end
% fre_final = fre_final';
% figure(2);
% %plot(real(RYA))
% plot(fre_final,real(RYA))
% %hold on
% xlabel('Freq (MHz)')
% ylabel('Magnitud')

% figure(3);
% stem(fre_final,real(RYA))
% xlabel('Freq (Hz)')
% ylabel('Magnitud')

```

A.6 Código Matlab para la reducción de variables (PCA)

```
% Importar matriz respuesta
X = xlsread('Nombre_Archivo.xlsx');
ref=[1 1 1 2 2 2 3 3 3];
n_muestra=max(ref);
p=X;

% Normalizar los datos de la matriz respuesta
[medidas, sensores]=size(p);
p_desv = std(p);
p_mean = mean(p);
p_norm = (p-repmat(p_mean,medidas,1))./repmat(p_desv,medidas,1);
XTM=p_norm;

% Calcular la matriz de covarianza
covariance = cov(XTM');

% Encontrar los "Autovalores" y "Autovectores"
[PC, V, S] = svd(covariance);%V=Autovalores PC=Componentes Principales
% Extraer los valores diagonales de la matriz de Autovalores
KDiag = diag(V);

% Cálculo de los porcentajes de los componentes principales
VT = sum(diag(V));
L1=[KDiag(1)]'; L2=[KDiag(2)]';
L3=[KDiag(3)]';
VPC1=(L1/VT)*100; VPC2=(L2/VT)*100;
VPC3=(L3/VT)*100;

% Proyección de los datos originales
XT = X;
[M,N] = size(XT);

% Restar el promedio para cada dimensión
mn = mean(XT,2);
XT = XT - repmat(mn,1,N);

%Componentes Principales
signals = PC'*XT;

%Primeros cinco componentes principales estandarizados
ppc = zscore(signals(1,:)); spc = zscore(signals(2,:));
tpc = zscore(signals(3,:)); cpc = zscore(signals(4,:));
qpc = zscore(signals(5,:));
mpccp = [ppc ; spc ; tpc ; cpc ; qpc];

%Standardized values for graphics
x_mean = zscore(signals(1,:));
y_mean = zscore(signals(2,:));

% Para las gráficas
for i=1:n_muestra
    posición_1=find(ref==i);
    X1=x_mean(posición_1)';%1ra Componente Principal(1er Columna)
    Y1=y_mean(posición_1)';%2da Componente Principal(2da Columna)
    MX1 = mean(X1);
    MY1 = mean(Y1);
    XM1=X1-mean(X1);%Proyección de los puntos de 1ra 1CP
    YM1=Y1-mean(Y1);%Proyección de los puntos de 2da 1CP
    M1=[XM1,YM1]; %XM1 y YM1 son convertidos a matriz
    mdl=LinearModel.fit(XM1,YM1);%Ajuste de proyección
    EST=mdl.Coefficients(2,1);
    M=table2array(EST); %Extraer el valor EST(ángulo)
    X_fit=XM1;%X_fit y y_fit representan la regresión lineal
    Y_fit=XM1*M;%que servirán de referencia para la elipse
    angle=atan(M);%Calculo del ángulo (con arco tangente)
```

```

    Rot=[cos(angle),sin(angle);-sin(angle),cos(angle)];%Rotación del
    ángulo (grados de inclinación)
    X_Rot=(Rot*M1)';%Rotación de la regresión lineal
%Cálculo de la distancia de los puntos hacia la línea
    X1_dist=fitdist(X_Rot(:,1),'normal');
    X2_dist=fitdist(X_Rot(:,2),'normal');
%Cálculo de la campana Gaussiana para el eje "x" y eje "y"
%Para el eje "x":
    pd=fitdist(X_Rot(:,1),'Normal');
    x_values=-1:0.001:1;
    y=pdf(pd,x_values);
    [valor,posicion]=min(abs(y-max(y)*0.1));%0.1 equivale a 90%
    x_value_pro=abs(x_values(posicion));
    PCX=x_mean(posicion_1);
    PCY=y_mean(posicion_1);
    PCFinalX(:,i)=PCX;
    PCFinalY(:,i)=PCY;
    LDAX=[PCFinalX PCFinalY];
%Se gráfica la campana de Gauss
    plot(x_values,y,'-')
%Para el eje "y":
    pd = fitdist(X_Rot(:,2),'Normal');
    x_values = -1:0.001:1;
    y = pdf(pd,x_values);
    [valor,posicion]=min(abs(y-max(y)*0.1));
    y_value_pro=abs(x_values(posicion));
%Se establecen los valores de los ejes para evitar circunferencia
    if (x_value_pro <= y_value_pro)
        x_value_pro = y_value_pro;
    end
    ecc = axes2ecc(x_value_pro,y_value_pro);
    ecc_new=[x_value_pro ecc];
%Función que permite establecer las características de la elipse
    [elat,elon] = ellipset1(mean(X1),mean(Y1),ecc_new,angle*180/pi);
%Gráfica que encierra los componentes principales en una elipse
    figure(1)
    plot(x_mean(posicion_1),y_mean(posicion_1),'.',elat,elon,'-');
    if i==1
        title('Análisis de Componentes Principales. ');
        xlabel(['1a. Comp Prin (' num2str(round(VPC1))
        '%)'], 'fontsize',12);
        ylabel(['2a. Comp Prin (' num2str(round(VPC2))
        '%)'], 'fontsize',12);
        hold
    end
    matrizX{i} = x_mean(posicion_1);
    matrizY{i} = y_mean(posicion_1);
end
prueba1 = matrizX{1,3};
legend('Sensor 1','','Sensor 2','','Sensor 3')
%Gráfica que muestra el comportamiento de las lecturas de cada sensor
Y_SPC1=S(:,1);
Y_SPC2=S(:,2);
[MY1,NY1] = size(Y_SPC1);%Matrix size: M=Rows N=Columns
X_SPC1=1:MY1;
X_SPC1=X_SPC1';
[MX1,NX1] = size(X_SPC1);%Matrix size: M=Rows N=Columns

```

```

figure(2);
plot(X_SPC1,Y_SPC1,'b. ');
for j=1:MX1
    lab(j,:) = sprintf('%4g',j);
    pj1=X_SPC1(j,1);
    pj2=Y_SPC1(j,1);
    text(pj1,pj2,lab(j,:));
end
hold on;
[MY2,NY2] = size(Y_SPC2);
X_SPC2=1:MY2;
X_SPC2=X_SPC2';
[MX2,NX2] = size(X_SPC2);
plot(X_SPC2,Y_SPC2,'r. ');
legend('Weights PC1','Weights PC2')
for k=1:MX1
    lab(k,:) = sprintf('%4g',k);
    pk1=X_SPC2(k,1);
    pk2=Y_SPC2(k,1);
    text(pk1,pk2,lab(k,:));
end

```

A.7 Código Matlab para el procesamiento (Red Neuronal)

```
%Importar datos de entrada
X0 = xlsread('Nombre_Archivo.xlsx');
%Establecer las entradas y los targets
XI = X0;
XT = [1 1 1 0 0 0 0 0 0 ; 0 0 0 1 1 1 0 0 0 ; 0 0 0 0 0 0 1 1 1];
%TARGETS
% %----- ESTRUCTURA DE LA RED-----
%Red de 1 entrada y dos capas, con 10 neuronas en la capa oculta y 3
%neuronas en la capa de salida, por lo que la notación establecida es
1:10:3
    net=newff(XI,XT,10);%newff = Crea una red feedforward
%Definimos el número de vectores de entrada
net.numInputs = 1;
%Establecer las capas restantes (Oculta y Salida)
net.numLayers = 2;
%Establecer las neuronas de cada capa
net.layers{1}.size = 10;
net.layers{2}.size = 3;
%Conectar la entrada con la capa oculta (primera capa), el número 1 es la
propiedad de conexión
net.inputConnect(1,1) = 1;
%Conectar el resto de las capas, definida por la matriz de conectividad
net.layerConnect(2,1) = 1;
%Definimos cuál es la capa de salida poniendo la propiedad a 1 para la
capa 2 y 0 para el resto de capas para que no se conecten a ella
net.outputConnect(1) = 0;
net.outputConnect(2) = 1;
%Establecer que función de activación que se utilizará para cada capa.
%La función de activación para la primera capa será: Tangente Hiperbólica
--> tansig
%La función de activación para la capa de salida: Exponencial Normalizada
--> softmax
    net.layers{1}.transferFcn = 'tansig';
    net.layers{2}.transferFcn = 'softmax';
%Observamos como diseñamos la estructura de la red
    %view(net)
%Verificamos las propiedades asociadas a cada capa que se ha creado
    Pro_1 = net.layers{1};
    Pro_2 = net.layers{2};
%Definimos los biases para cada etapa, al asignar 1 a la propiedad
%significa que la capa tiene biases establecidos
    net.biasConnect = [1 ; 1]; % [Capa Oculta ; Capa de Salida ]
%Establecemos el procedimiento por defecto para inicializar los pesos y
biases (A menos que se tenga una rutina personalizada)
    net.initFcn = 'initlay';
%Establecemos la funcion de control (conjunto de validación) a "Entropía
Cruzada"
    net.performFcn = 'crossentropy';
%Establecemos la función de entrenamiento
%Funciones estandar --> trainlm (Levenberg-Marquardt). trainbfg (BFGS
Quasi-Newton)
%Funciones Reconocimiento de Patrones --> trainscg (Gradiente conjugado
escalado). trainrp (Backpropagation resistente)
    net.trainFcn = 'trainscg';
```

```

%Modificamos como se dividiran los datos para el
entrenamiento(trainRatio),validación(valRatio) y prueba(testRatio)
net.divideFcn = 'dividerand'; % Divide data randomly
net.divideMode = 'sample'; % Divide up every sample
net.divideParam.trainRatio = 70/100;
net.divideParam.valRatio = 15/100;
net.divideParam.testRatio = 15/100;
%Establecemos que gráficos queremos que aparezcan
net.plotFcns = {'plotperform','plottrainstate','ploterrhist', ...
'plotconfusion','plotroc'};
%Iniciamos la red
net = init(net);
%Valores de los pesos
pesosEntrada_1 = net.IW{1,1};%Capa de Entrada
pesosCapa_1 = net.LW{2,1};%De la capa oculta a la capa de salida
bias_capa_oculta_1 = net.b{1};%Biases de cada neurona en la capa
oculta
bias_capa_salida_1 = net.b{2};%Biases de cada neurona en la capa de
salida
%-----ENTRENAMIENTO-----
%Normalizamos los valores de la red con la función "mapminmax", asume los
valores de cada celda como números reales finitos y con al menos uno de
diferente de los demás, y en el caso de que se repitiesen estos son
"removeconstantrows" elimina del grupo los valores que no aportan
información adicional a la red.
net.input.processFcns = {'removeconstantrows','mapminmax'};
net.output.processFcns = {'removeconstantrows','mapminmax'};
%Establecemos además el número de iteraciones (epochs), la tasa de
%aprendizaje (lr)[Controla el cambio de los pesos en cada iteración],
%el factor momento (mc) [Encargado de acelerar la convergencia]
net.trainParam.epochs = 1000;%Número máximo de iteraciones
net.trainParam.lr = 0.05;%El valor mayor posible entre 0.05 y 0.5,
sin que provoque grandes oscilaciones
net.trainParam.mc = 0.95;%Valores entre 0.95 y 1
net.trainParam.showWindow = 1;%1=para mostrar ventana 0=para no
mostrar el GUI de entrenamiento
%Entrenamos la red
[net,tr] = train(net,XI,XT);
%-----VALORES OBTENIDOS-----
mdl=tr;
%Número de iteraciones
ITE = mdl.num_epochs;
fprintf('Número de iteraciones: %g\n ', ITE);
%Cálculo del tiempo
TM=mdl.time(1,:);%Se guardan los valores de tiempo de cada
iteración
t = sum(TM);%Se suma el tiempo total
fprintf('El proceso ha tardado %g segundos', t);
%Valores que tomas los pesos de entrada y de las capas así como las
biases una vez realizado el ajuste de la red
pesosEntrada_2 = net.IW{1,1};
pesosCapa_2 = net.LW{2,1};
bias_capa_oculta_2 = net.b{1};
bias_capa_salida_2 = net.b{2};
%-----SALIDA-----
salida = sim(net,XI)
%-----CLASES-----

```

```

y = net(XI);
perf = perform(net,XT,y);
targets = vec2ind(XT);
classes = vec2ind(y)
%-----ERROR CUADRATICO MEDIO-----
%Valor para el análisis del funcionamiento de la red
d = (salida-XT).^2;
mse = mean(d);
%-----GRÁFICAS-----
%Se gráfica el funcionamiento de la red en las etapas de entrenamiento,
validación y prueba, así como el mejor resultado durante estas fases
figure(1)
plotperform(tr)
%Se gráfica el ajuste entre los valores estimados y predichos
figure(2)
plotregression(salida,XT)
%Se gráfica el porcentaje de error y acierto de la clasificación
figure(3)
plotconfusion(XT,y)

```