

DISEÑO E IMPLEMENTACIÓN DE UN DISPOSITIVO DE ACOPLAMIENTO
ENTRE SENSORES DE NARICES ELECTRONICAS Y UN TERMINAL MOVIL DE
PROPÓSITO ESPECIAL

EDWING ALFONSO ARENAS RUEDA

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICO-MECANICAS
ESCUELA DE INGENIERÍAS ELECTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES
BUCARAMANGA
2016

DISEÑO E IMPLEMENTACIÓN DE UN DISPOSITIVO DE ACOPLAMIENTO
ENTRE SENSORES DE NARICES ELECTRONICAS Y UN TERMINAL MOVIL DE
PROPÓSITO ESPECIAL

EDWING ALFONSO ARENAS RUEDA

Trabajo de grado para optar al título de
Ingeniero Electrónico

Director
HOMERO ORTEGA BOADA
Ingeniero Electrónico, Ph.D

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERIAS FISICO-MECANICAS
ESCUELA DE INGENIERÍAS ELECTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES
BUCARAMANGA

2016

CONTENIDO

	Pág.
INTRODUCCIÓN	16
1. MARCO TEORICO	18
1.1. EL OLOR	18
1.2. EL SENTIDO DEL OLFATO	20
1.3. LOS COMPUESTOS ORGANICOS VOLATILES	21
1.4. GENERALIDADES DE LOS SISTEMAS DE NARIZ ELECTRONICA	23
1.5. TIPOS DE SENSORES	25
1.6. CLASIFICACION DE LOS SENSORES QUIMICOS	25
2. ANTECEDENTES	29
3. DESARROLLO DEL PROYECTO	30
3.1. ETAPA 1: IDENTIFICACION Y DESCRIPCION DE LA NARIZ ELECTRONICA DE PAMPLONA	30
3.1.1. Nariz electrónica – Universidad de Pamplona	30
3.1.2. Hardware de la Nariz Electrónica	31
3.1.3. Software de la Nariz Electrónica	35
3.1.4. Aspectos generales del hardware de la Nariz Electrónica	38

3.2. ETAPA 2: IDENTIFICACION Y DESCRIPCION DEL TERMINAL MÓVIL	40
3.2.1. Características generales del Terminal Móvil	40
3.2.2. Tarjeta Intel Edison	41
3.2.3. Terminal móvil de nariz electrónica de RadioGIS	43
3.2.4. Programación implementada en el Terminal Móvil	44
3.2.5. Aspectos generales del Terminal móvil de nariz electrónica de RadioGIS	45
3.3. ETAPA 3: DISEÑO DEL SISTEMA DE NARIZ ELECTRONICA	46
3.3.1. Metodología de diseño	46
3.3.2. Parte 1: módulo de sensores	47
3.3.2.1 Sensores de gases	51
3.3.3. Parte 2: módulo DAQ	55
3.3.4. Parte 3: módulo central	59
3.3.5. Parte 4: módulo de salida	62
3.3.5.1. Metodología 1	63
3.3.5.2. Metodología 2	64
3.3.6. Parte 5: módulo de Nariz Electrónica	66
3.3.6.1. Cámara de concentración	67
3.3.6.2. Cámara de medida	68
3.3.6.3. Electroválvulas	68
3.3.6.4. Regulador de presión	69
3.3.6.5. Fuente de alimentación para la matriz de sensores	70

3.4. ETAPA 4: EQUIPO FINAL	71
4. PRUEBAS Y RESULTADOS	73
4.1. PRUEBAS PRELIMINARES	73
4.2. ETAPA DE EXPERIMENTACION	75
4.3. MANEJO DE ARCHIVOS DE MEDICION	79
4.4. ENVIO DE DATOS	81
5. CONCLUSIONES	83
6. RECOMENDACIONES	84
CITAS BIBLIOGRAFICAS	85
BIBLIOGRAFIA	89
ANEXOS	90

LISTA DE TABLAS

	Pág.
Tabla 1: Algoritmos principales implementados en la herramienta Bnose	37
Tabla 2: Características generales de los sensores implementados en la Matriz de sensores	50
Tabla 3: Canales de la matriz de sensores	50
Tabla 4: Especificaciones eléctricas generales para sensor de gas	54
Tabla 5: Características generales del integrado TLC2543	56
Tabla 6: Características generales del integrado PCF8574	57
Tabla 7: Características de la tarjeta de desarrollo Intel Galileo Gen 1	60

LISTA DE FIGURAS

	Pág.
Figura 1. Esquema del sistema olfativo del ser humano	20
Figura 2. Vista ampliada de la parte receptora del sistema olfativo	21
Figura 3. Analogía entre el sistema olfativo biológico y un sistema de nariz electrónica	24
Figura 4. Esquema general de un mosfet	26
Figura 5. Respuesta de un sensor de gas (TGS822) ante una muestra de esencia de hinojo	27
Figura 6. Respuesta de un conjunto de sensores de gases ante una concentración de gas Butano (gas de encendedor)	28
Figura 7. Equipo de Nariz Electrónica, desarrollada por la Universidad de Pamplona	30
Figura 8. PCB y visualización 3D de la matriz de sensores	32
Figura 9. Visualización del funcionamiento del sistema neumático	32
Figura 10. Visualización de la fuente de alimentación de 600W, marca UNITEC	33
Figura 11. Visualización del módulo de control implementado	34
Figura 12. Tarjeta de adquisición NI USB-6210	34
Figura 13. Ventana general del programa Bnose	35
Figura 14. Ventana de la interface de adquisición	36
Figura 15. Grafica resultante del proceso de adquisición mediante la herramienta Bnose	36
Figura 16. Ventana de la interface de procesamiento del programa Bnose	37

Figura 17. Equipo de nariz electrónica, vista de sus elementos en su interior	38
Figura 18. Modelo base para hardware de Nariz Electrónica	39
Figura 19. Tarjeta Intel Edison	42
Figura 20. Metodología de Terminal móvil desarrollado	43
Figura 21. Aplicación web, implementada para la visualización de la información transmitida por el terminal móvil	44
Figura 22. Visualización de los datos tomados por el GPS	44
Figura 23. Modelo general de un servicio basado en nariz electrónica	46
Figura 24. Modelo general del sistema de nariz electrónica y bloques desarrollados	47
Figura 25. Esquemático de la matriz de sensores	48
Figura 26. PCB y visualización 3D de la matriz de sensores	49
Figura 27. Esquema interno de los sensores serie 8	51
Figura 28. Versiones de encapsulado de los sensores de la serie 8 tipos R, C y M	52
Figura 29. Esquema de circuito para sensor de gas (a) Fígaro y (b) Hanwei	52
Figura 30. Esquema general de circuito para sensor de gas	53
Figura 31. PCB y visualización 3D del sistema de adquisición	55
Figura 32. Esquema general para el integrado TLC2543	56
Figura 33. Esquema general para el integrado PCF8574	57
Figura 34. Esquemático de sistema de adquisición y control	58
Figura 35. Tarjeta Intel Galileo Gen1	59

Figura 36. Diagrama de flujo del programa implementado para el sistema de adquisición en la tarjeta de desarrollo Intel Galileo	61
Figura 37. Esquema de los pines implementados en la tarjeta de desarrollo Intel Galileo	62
Figura 38. Interfaz de adquisición para equipo de nariz electrónica, implementado mediante la herramienta MATLAB	63
Figura 39. Vista de la interfaz de adquisición implementada mediante la herramienta PYTHON	64
Figura 40. Vista de la interfaz de adquisición de PYTHON, donde se puede observar la configuración de tiempos y el inicio del proceso de medida	65
Figura 41. Modelo general de nariz electrónica	66
Figura 42. Vista general de la cámara de concentración	67
Figura 43. Vista general de la cámara de medida	68
Figura 44. Vista de (a) las electroválvulas y (b) el módulo de adecuación para las electroválvulas	69
Figura 45. Regulador de presión modelo 85040	69
Figura 46. Vista de la fuente de poder utilizada para la alimentación de la matriz de sensores	70
Figura 47. Vista del módulo final de nariz electrónica	71
Figura 48. Vista general del prototipo de nariz electrónica durante un experimento	72
Figura 49. Prueba preliminar de la matriz de sensores y el sistema de Adquisición	73
Figura 50. Respuesta transitoria de la matriz de sensores ante muestras de (a) cloruro, (b) butano y (c) espuma de poliuretano parte A	74
Figura 51. Compuesto comercial conocido como “silicona líquida”	75
Figura 52. Vista de la nariz electrónica durante un experimento para la medida de volátiles de un compuesto comercial conocido como “silicona líquida”	76

Figura 53. Respuesta transitoria de una muestra de olor de “silicona líquida”	76
Figura 54. Comparación entre las respuestas transitorias de la nariz electrónica de (a) RadioGIS y (b) la Universidad de Pamplona	77
Figura 55. Respuesta transitoria del sensor TGS822, resaltando los parámetros de interés	79
Figura 56. Archivos generados en el interior de la tarjeta microSD	80
Figura 57. Muestra de los archivos generados por la interface	80
Figura 58. Modem marca D-Link modelo DWR-921	81
Figura 59. Archivos de medidas transmitidos hacia un servidor de pruebas (260mb)	82
Figura 60. Circuito sistema de adquisición y control (Esquemático)	90
Figura 61. Circuito matriz de sensores (Esquemático)	91
Figura 62. Layout Circuito matriz de sensores (Top-rojo/Bottom-verde)	92
Figura 63. Layout Circuito sistema de adquisición y control (Top-rojo/Bottom-verde)	93

LISTA DE ANEXOS

	Pág.
Anexo A. Esquemáticos y negativos de las tarjetas	90
Anexo B. Código fuente Intel Galileo	94
Anexo C. Código Matlab adquisición	101
Anexo D. Código Matlab interface	105
Anexo E. Código Python DAQ	113
Anexo F. Código Python transmisor FTP	119

RESUMEN

TITULO: DISEÑO E IMPLEMENTACIÓN DE UN DISPOSITIVO DE ACOPLAMIENTO ENTRE SENSORES DE NARICES ELECTRONICAS Y UN TERMINAL MOVIL DE PROPÓSITO ESPECIAL*

AUTOR: EDWING ALFONSO ARENAS RUEDA**

PALABRAS CLAVE: NARIZ ELECTRONICA, OLFATO ELECTRONICO, TERMINAL MOVIL, OLOR, INTEL GALILEO, SENSORES DE GAS, PYTHON.

DESCRIPCIÓN:

En el marco del presente trabajo de grado se ha planteado la necesidad de construir un equipo de nariz electrónica para laboratorio, que permita captar la señal de respuesta de diversos tipos de olores, cuyos datos resultantes puedan transmitirse finalmente hacia un servidor web. Además todas estas cualidades deben representar al desarrollador una opción fácil, rápida y de bajo costo.

Esta solución parte con la propuesta de identificar y describir el desarrollo de un equipo de nariz electrónica de la universidad de pamplona, y de allí extraer una metodológica básica que le permita al desarrollador tener las ideas y bases claves para el diseño de un módulo de nariz electrónica. Y es a partir de esta metodología propuesta que se desarrolla en este trabajo de grado un módulo portable, compacto y versátil para nariz electrónica y con las posibilidades de poder adaptarlo a cualquier aplicación en específico que requiera el desarrollador, de manera que este nuevo dispositivo pueda acoplarse y adaptarse fácilmente a cualquier herramienta computacional, periférico de comunicación o terminales móviles disponibles.

Gracias a este desarrollo, los grupos CEMOS y RadioGIS ya cuentan hoy con el primer prototipo de nariz electrónica e incluso han iniciado varios emprendimientos como la puesta en marcha de proyectos como el uso de las narices electrónicas para identificar el vector del mal de Chagas, con los cuales crecen las expectativas para numerosos proyectos.

*Trabajo de grado

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería Eléctrica, Electrónica y Telecomunicaciones. Director: Homero Ortega Boada.

ABSTRACT

TITLE: DESIGN AND IMPLEMENTATION OF A DEVICE FOR PAIRING SENSORS ELECTRONIC NOSE AND A SPECIAL PURPOSE MOBILE TERMINAL *

AUTHOR: EDWING ALFONSO ARENAS RUEDA**

KEYWORDS: ELECTRONIC NOSE, MOBILE TERMINAL, SMELL, INTEL GALILEO, GAS SENSORS, PYTHON.

DESCRIPTION:

As part of this degree work has raised the need to build an electronic nose for laboratory that captures the response signal various types of smells, which can ultimately resulting data upload to a web server. Also all these qualities should represent the developer an easy option, quick and cheap.

This solution starts with the proposal to identify and describe the development of an electronic nose system of the University Pamplona, and from there extract a basic methodology that allows the developer to have ideas and keys to designing an electronic nose system base. And it is from this proposed methodology that is developed in this degree work a portable module, compact and versatile electronic nose system and possibilities to suit any specific application requiring the developer, so this new device can be coupled and easily adapt to any computational tool, peripheral communication or mobile terminals available.

Thanks to this development, the CEMOS and RadioGIS research groups now have the first prototype of electronic nose system and initiating several projects as the implementation of the use of electronic noses system to identify the vector of Chagas disease, with high expectations for many projects, as the implementation of a solution to the detection of landmines.

* Degree Work

** Faculty of Physical- Mechanical Engineering, School of Electrical, Electronic and Telecommunications Engineering, Director Homero Ortega Boada.

INTRODUCCIÓN

Difícilmente podremos borrar de nuestra mente las imágenes de Dayco, el perro que murió de agotamiento en el terremoto de Ecuador, no sin antes salvar muchas vidas con su poderosa nariz. Precisamente, este trabajo de grado representa un aporte al diseño e implementación de una nariz electrónica que, aunque no puede competir con la de Dayco, puede tener grandes aplicaciones bien sea en la seguridad de las personas, en el cuidado del medio ambiente, en la medicina o en la industria. De esta manera, los grupos de investigación RadioGIS y Cemos, con acompañamiento de la Universidad de Pamplona, se han planteado el reto de contar con una nariz electrónica con la cual espera abrir un amplio abanico de oportunidades de investigación y desarrollo aplicado a diferentes necesidades.

Es a partir de una versión portable y compacta de nariz electrónica, que se plantea la posibilidad de formar más adelante un modelo de servicio basado en estos equipos y que incluirá la tecnología de los dispositivos móviles y aplicaciones basadas en un webservice, todo con el sencillo fin de poder utilizar el sentido de la nariz electrónica para realizar una medición de un olor de un determinado lugar, y transmitir los datos tomados hacia un centro remoto de procesamiento, para posteriormente analizar los datos, y emitir la información a través de las tecnologías de cómputo, aplicaciones web o dispositivos móviles. Para lograr llegar a la visión general de la nariz electrónica como un servicio itinerante, los grupos de investigación mencionados trabajan sobre un conjunto de necesidades y estrategias de desarrollo, como son el diseño de un equipo de nariz electrónica, la metodología para terminales móviles, el sistema de procesamiento de patrones de olores a través del cómputo en la nube, y la búsqueda de potenciales campos de aplicación para ofrecer un equipo que pueda ser adaptado a las necesidades de los usuarios.

De manera específica, para este trabajo de grado se ha planteado la necesidad de crear un dispositivo que permita capturar y adquirir la señal de respuesta de diversos tipos de olores, de manera que los datos resultantes de estos se pre procesen y puedan transmitirse en a un servidor web, para que estos datos puedan ser estudiados de manera posterior; además todas estas cualidades deben representarle al desarrollador una opción fácil, rápida y de bajo costo.

La solución parte con la propuesta de brindar una metodológica básica que le permita a cualquier desarrollador tener las ideas y bases claves para el diseño de un módulo de nariz electrónica. Y es a partir de esta metodología propuesta que se desarrolla en el presente trabajo de grado un módulo portable, compacto y versátil para nariz electrónica, con las posibilidades de poder adaptarlo a cualquier

aplicación en específico que requiera el desarrollador, de tal manera que este nuevo dispositivo pueda acoplarse y adaptarse fácilmente a cualquier herramienta computacional, periférico de comunicación o terminales móviles disponibles. Gracias a este desarrollo, los grupos CEMOS y RadioGIS ya cuentan hoy con el primer prototipo de nariz electrónica e incluso han iniciado varios emprendimientos como la puesta en marcha de un trabajo de maestría para usar las narices electrónicas para identificar el vector del mal de Chagas. Otro destacable es la participación en la implementación de un sabueso para detectar minas antipersonas, con los cuales crecen las expectativas para números proyectos.

1. MARCO TEORICO

1.1. EL OLOR

El olor se puede definir como una sensación producida por la percepción de algún estímulo del sistema olfativo; o también se puede describir el olor como la impresión que da el olfato ante un compuesto químico que se encuentra disipado en el aire [1,2].

Para que nuestro sistema olfativo logre percibir un compuesto químico y lo podamos identificar y asociar en nuestra mente como un olor, se deben tener algunas características básicas de manera que nuestro estímulo del olfato pueda captar dicho olor, además de las cualidades propias de un olor, de entre ellas se pueden resaltar:

- El compuesto químico debe tener volatilidad o que se evapora fácilmente, esta condición es necesaria más no suficiente para poder estimular el sistema olfativo, debido a que algunas sustancias son volátiles pero carecen de olor (ej. Vapor de agua).
- las sustancias olorosas deben ser solubles en agua y lípidos, con el fin de que estos compuestos logren penetrar la película acuosa y la capa de lípido que recubre los receptores olfativos.
- Todos los olores por lo general son mezclas de compuestos químicos.

La percepción y/o apreciación de un olor es por lo general muy subjetiva, debido a que la apreciación de un cierto olor está muy influenciada por diferentes aspectos según cada individuo, incluso este sentido puede ligarse a las emociones.

Hasta ahora no hay un mapa completo de olores que pueda ajustarse a cada ser vivo de manera general. Cada individuo tiene una manera propia para detectar y describir un olor, a su vez cada ser percibe un olor y lo identifica comúnmente mediante el método de comparación con otro olor percibido en algún otro escenario, sin embargo esta apreciación es bastante subjetiva y puede variar en cada ser vivo.

Además se debe tener muy en claro que todos los olores tienen una duración corta o en otras palabras tiene un periodo de tiempo limitado por el ambiente; esto

es debido a que un olor es en sí mismo un resultado de un proceso químico que está sucediendo en la materia orgánica o inorgánica, y su duración depende de cuanto pueda permanecer el proceso químico hasta que el mismo se estabilice y pase a un compuesto totalmente neutro [3].

Aunque hay discusión en la comunidad científica por tratar de clasificar los olores y enmarcarlos como una referencia general para todos los seres vivos, oficialmente se han catalogado en siete olores primarios, esta clasificación fue propuesta por Amoore en 1962 [4]. Este sistema de clasificación optó por estudiar un tipo de olor y relacionarlo a un compuesto natural de nuestro alrededor como una posible norma para su identificación, los cuales son:

- **Floral:** es un olor floral, característico de las flores.
- **Menta:** es un olor agradable, como la planta de hierbabuena.
- **Alcanfor:** se caracteriza por un olor penetrante, como el mismo nombre de la planta.
- **Podrido:** olor de putrefacción, podrido.
- **Etéreo:** es un olor penetrante y sabor dulce.
- **Acre:** es un olor áspero y picante, como el olor a ajo, o fósforos encendidos.
- **Almizcle:** es un olor intenso, relacionado con la secreción de algunos animales.

Por lo general un aspecto que es muy cotidiano como seres humanos, reside en que somos capaces de reconocer entre 2000 y 4000 muestras entre olores, aromas y fragancias diferentes, aunque se estimen cerca de 50000 [1]. También está el hecho de que el ser humano no está muy capacitado para el reconocimiento de olores si este se pone en comparación con otras especies de seres vivos. Por lo general el olor y la forma como las especies pueden percibir el mismo sugiere muchas teorías y estudios que buscan explicar la misma capacidad para percibir los olores, aunque en su mayoría su estudio no se basa en la capacidad biológica para percibirlo, sino en la forma como se interpreta la información sensorial en el cerebro para posteriormente relacionar un aroma como un patrón o huella [5].

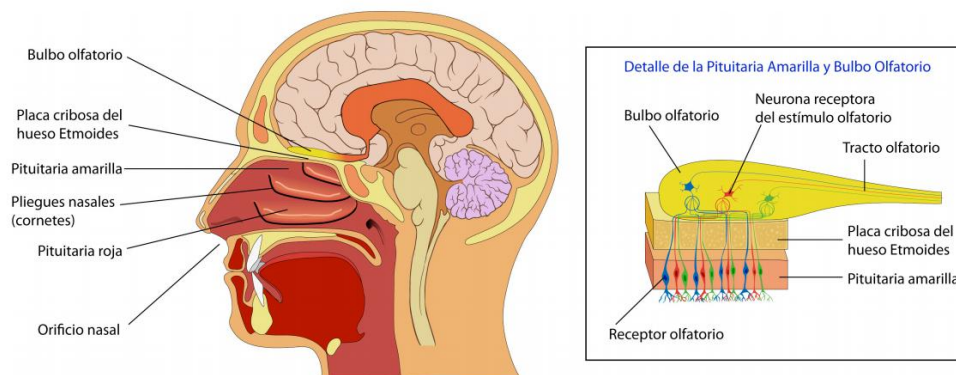
1.2. EL SENTIDO DEL OLFATO

El sentido del olfato es uno de los sentidos más sofisticados donde una especie puede captar información de su ambiente y también desempeñar un papel clave en su sistema sensorial, lo que lo lleva a modificar su comportamiento y gustos hacia determinados alimentos y ambientes.

En el reino animal el olfato es un sentido altamente desarrollado y utilizado para diversos propósitos como la alimentación, la conducta, la orientación y la supervivencia. A diferencia de los animales, el aparato olfativo del ser humano es bastante rudimentario y no es un sentido especialmente desarrollado, posiblemente esta característica se debe a que se ha acentuado más el desarrollo de los demás sentidos y también otras aptitudes [6].

Resumiendo su funcionamiento biológico, este sentido se basa en quimiorreceptores, de entre 10 y 100 millones, que se encuentran en la porción superior de la cavidad nasal, conocida también como la pituitaria amarilla, un tejido epitelial especializado que ocupa una superficie de unos 5cm². En la zona inferior de la cavidad nasal se denomina la pituitaria roja. Las células receptoras se encuentran incluidas entre las epiteliales de sostén y las glándulas olfatorias de Bowman, encargadas de producir mucosidad y humedecer la superficie. La figura 1 nos da una imagen de las diferentes partes que componen el sistema olfativo en el ser humano [6].

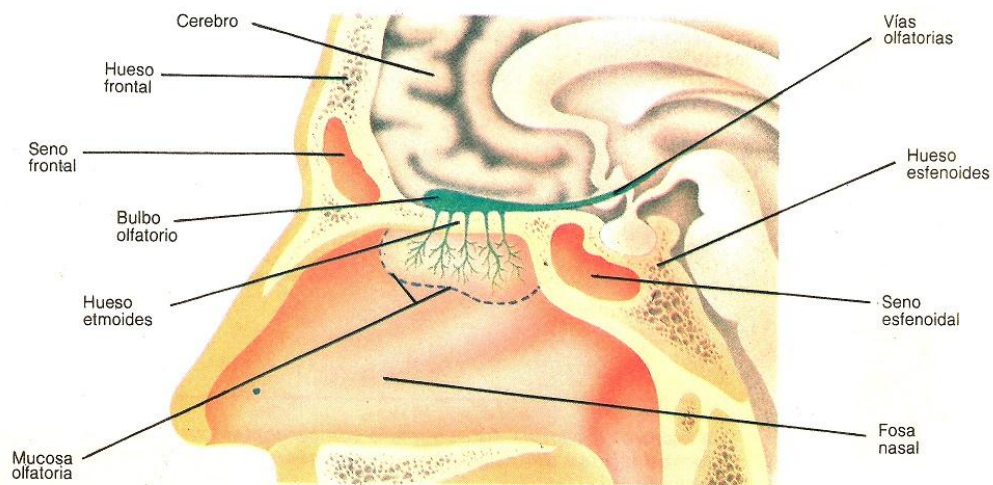
Figura 1. Esquema del sistema olfativo del ser humano



Fuente: Anatomía y Fisiología I. Disponible a Mar-30-2016 en: http://www.elmodernoprometeo.es/Sitio_web/Anatomia_files/sentidos.pdf

Cuando llega una sustancia a la parte receptora del sistema olfativo (figura 2), esta manda un impulso al sistema nervioso central mediante la membrana olfatoria, estas atraviesan los agujeros que tiene el hueso etmoides y que constituye el techo de la cavidad nasal, a esta zona se le denomina placa cribosa y llega a una serie de lóbulos que se encuentran justo por encima del hueso etmoides y que forma parte del encéfalo, y constituye lo que se denomina el bulbo olfativo, de ahí la información pasa a los lóbulos frontales del cerebro. Donde se analizan y se interpreta como un aroma o fragancia [6,7].

Figura 2. Vista ampliada de la parte receptora del sistema olfativo



Fuente: Los Sentidos Gusto Y Olfato. Disponible en Mar-30-2016 en: <http://enfermedadesyremedioscaseros.com/wp-content/uploads/el-olfato.jpg>

1.3. LOS COMPUESTOS ORGÁNICOS VOLÁTILES

Los compuestos orgánicos volátiles, o COV (por sus siglas en español), se pueden definir como sustancias químicas que se convierten fácilmente en vapores o gases. Además de ser compuestos que contiene carbono, contiene también elementos como hidrogeno, oxígeno, flúor, cloro, azufre, o nitrógeno. Los COV son sustancias liberadas por combustibles durante su ignición, como la gasolina, madera, gas natural o carbón. Así también pueden ser liberados por pinturas, disolventes, compuestos químicos industriales, compuestos orgánicos y otros productos que son empleados en nuestro diario vivir. De los compuestos orgánicos volátiles, se pueden ver algunas características como:

- **Volatilidad:** son así llamados los compuestos orgánicos que se evaporan de manera rápida en el ambiente.
- **Liposolubilidad:** son las moléculas orgánicas que son liposolubles, es decir tienen una afinidad por las grasas y se acumulan en los tejidos grasos; producto resultante del metabolismo dentro de un organismo.
- **Inflamabilidad:** son los compuestos que arden con facilidad al contacto con el aire.
- **Toxicidad:** dependiendo de cada compuesto, las propiedades tóxicas pueden variar, así como sus notables efectos en un ser vivo.

Los compuestos orgánicos volátiles por lo general pueden tener una fuente de origen natural o de manera artificial. Una posible fuente natural muy común es el metano, un gas de efecto invernadero generado a partir de la descomposición de la materia orgánica. Por otro lado los COV de fuente artificial tienen su lugar en la industria como por ejemplo la elaboración de químicos para pinturas, calzado, pesticidas, disolventes, cosméticos, plásticos, entre muchos otros.

De manera más general la mayoría de los COV se producen en la realización de actividades industriales. Cabe resaltar que hay muchas actividades en la industria donde es fundamental y susceptible se genere productos a partir de COV. Algunos ejemplos de compuestos orgánicos volátiles serían:

- **Naturales:** isopreno, pineno, limoneno, entre otros.
- **Artificiales:** benceno, tolueno, nitrobenceno, entre otros.

Otro aspecto de los compuestos orgánicos volátiles, colocando a un lado la actividad industrial, da su importancia a los efectos ambientales y los posibles efectos que pueden tener sobre la salud de los seres vivos. Los efectos sobre la salud pueden ir desde un alto grado de toxicidad hasta una ausencia de efectos nocivos [8].

Los efectos perjudiciales provocados por la exposición prolongada a los COV, dependerán en sí mismo de la naturaleza de cada compuesto, del grado de toxicidad y del periodo de exposición al que se somete [9].

1.4. GENERALIDADES DE LOS SISTEMAS DE NARIZ ELECTRONICA

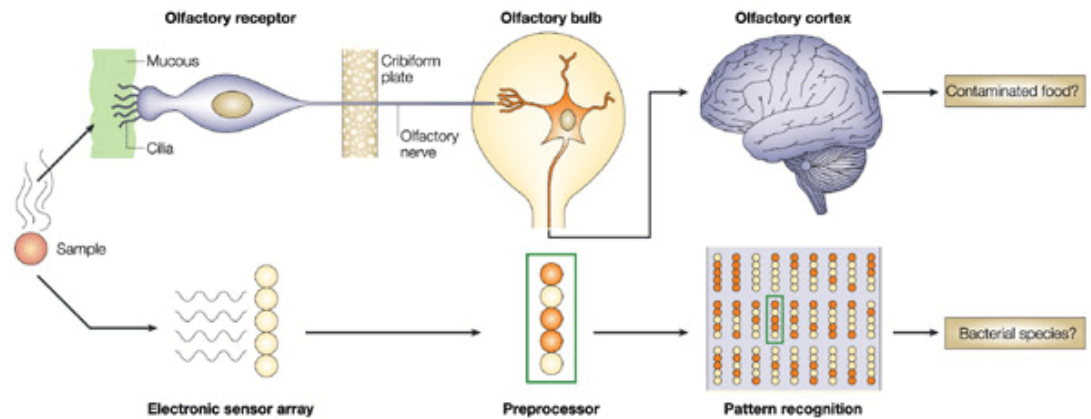
Una de las definiciones más utilizadas para describir lo que es un sistema de nariz electrónica, es la de Gardner y Barlett (Gardner y Barlett, 1999) donde se describe como “un instrumento que comprende una matriz de diferentes tipos de sensores químicos electrónicos con una especificidad parcial, y un sistema computacional de análisis de datos y reconocimiento de patrones, capaz de reconocer olores simples y/o complejos” [10].

Uno de los aspectos esenciales de los sistemas de nariz electrónica, radica en capturar y poder identificar lo que nuestro sistema olfativo percibe como un olor, aroma o fragancia, Independientemente del término que se use. Desde el punto de vista químico, el olor es una sensación, estímulo o percepción producida por el sentido del olfato, producto de la interacción de una sustancia orgánica compuesta por gases, vapores y polvo, con los receptores olfativos de los seres vivos.

El desarrollo de un dispositivo electrónico para la captación de olores y reconocimiento, parte de cómo funciona el sentido del olfato de los seres vivos y a partir del mismo se trata de emular este sentido mediante una versión electrónica que permita imitar la captura y reconocimiento de olores [11].

En un sistema de nariz electrónica (NE), las células receptoras del sistema olfativo son reemplazadas por un arreglo de sensores de gases, que reaccionan a diversos compuestos volátiles. El bulbo olfatorio podría reemplazarse por un medio de adquisición de las señales, acondicionamiento y preprocesado de los diferentes sensores olfativos, como respuesta a un olor. Y por último la respuesta del cerebro para poder reconocer un olor y su intensidad, se puede reemplazar por algún método de reconocimiento de patrones [12]. La figura 3 puede mostrar la analogía entre un sistema olfativo biológico y un sistema de nariz electrónica.

Figura 3. Analogía entre el sistema olfativo biológico y un sistema de nariz electrónica



Fuente: TURNER Anthony P.F., MAGAN Naresh. Electronic noses and disease diagnostics. Nature Reviews Microbiology 2. 2004, February. p. 161-166. ISSN: 1740-1526.

El objetivo final tanto para el sistema de olfato biológico como para el electrónico es poder reconocer patrones de olores y poder identificarlos. Visto desde la capacidad de un ser humano, este no necesita identificar totalmente toda una serie de compuestos que estén presentes en un olor, basta con poder reconocerlo solamente y que pueda ser diferenciado de otros. Visto desde la capacidad de un nariz electrónica, cada sensor tiene un espectro de volátiles que es capaz de percibir, además de poder percibir un volátil en particular, formando así una respuesta variada según el tipo de volátil presente en el entorno, esta característica hace que el nariz electrónica forme un patrón de datos relacionado con el olor percibido.

Sin embargo se debe resaltar que los sistemas actuales para reconocimiento de olores no trabajan exactamente igual que una nariz humana, debido a que no detectan los mismos compuestos volátiles que esta, además de presentar una menor sensibilidad. El número de células receptoras olfativas en relación con los sensores, además de un umbral de detección para cualquier compuesto hacen de la nariz electrónica no trabaje exactamente igual a la nariz humana. Por esto algunos autores prefieren no hacer la asociación directa entre la nariz electrónica y la nariz humana. Por esto prefieren llamar a estos equipos sensores de gases, que permiten identificar una determinada familia de compuestos químicos, a partir de una concentración y sensibilidad presente en la mezcla que se quiere oler. Es a partir de ello, que el desarrollo de un sistema de nariz electrónica se reduzca a dos fundamentos: la capacidad de captura de olor de una matriz de sensores y un

método de análisis de reconocimiento de patrones para un conjunto de muestras de olores [12].

1.5. TIPOS DE SENSORES

En un sistema de nariz electrónica el componente principal es el sensor químico de gases, que es un elemento que sufre un cambio físico o químico reversible en presencia de un gas, para dar pasó a una señal eléctrica resultante dependiendo del tipo de gas y su concentración. Generalmente la base de un sensor químico es un recubrimiento sensible a la presencia de un tipo de gas [11]. Este recubrimiento puede ser de dos tipos:

ORGANICOS: una gran ventaja radica en que pueden ser modificados mediante diferentes métodos de síntesis, con lo cual se puede modificar sus propiedades de sensibilidad. Pero su inconveniente está en la respuesta de los sensores debido a la falta de estabilidad de los compuestos orgánicos.

INORGANICOS: entre ellos están los óxidos de metal (SnO_2 , TiO_2 , entre otros). Una ventaja está en su estabilidad y pureza, aunque resulta más difícil poder modificar y obtener recubrimientos con diferentes propiedades de selectividad y sensibilidad.

1.6. CLASIFICACIÓN DE LOS SENSORES QUIMICOS

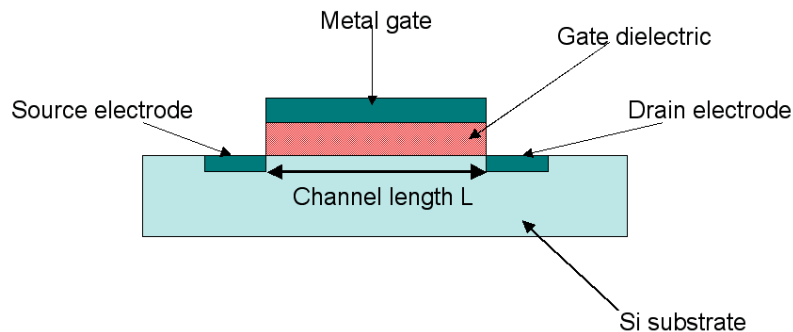
Los sensores químicos se pueden clasificar según su principio de operación que utilice y sus posibles aplicaciones. Entre las propiedades más empleadas están la conductividad, voltaje, capacitancia, calentamiento, masa o constantes ópticas los cuales cambian al variar una concentración de muestra química que interactúa con el recubrimiento sensible del sensor [14]. Los más representativos son:

- Sensores electrolíticos.
- Sensores de efecto de campo.
- Sensores fotométricos.
- Sensores sensibles a la masa.
- Sensores semiconductores de óxidos metálicos.

Sensores electrolíticos: este tipo de sensores se forma por un electrodo metálico que está en contacto con un gas a detectar y un segundo electrodo de referencia. Su principio se basa en el potencial electroquímico que se genera por la concentración de gas.

Sensores de efecto de campo (MOSFET): en este tipo de sensor se emplea un transistor de efecto de campo como elemento. Este utiliza la propiedad de que el metal de la compuerta se modifica ante la presencia de concentraciones de gas (figura 4).

Figura 4. Esquema general de un mosfet



Fuente: MOSFET (Metal-oxide semiconductor field effect transistor). Disponible a Mar-30-2016 en: <http://www.ustudy.in/node/673>

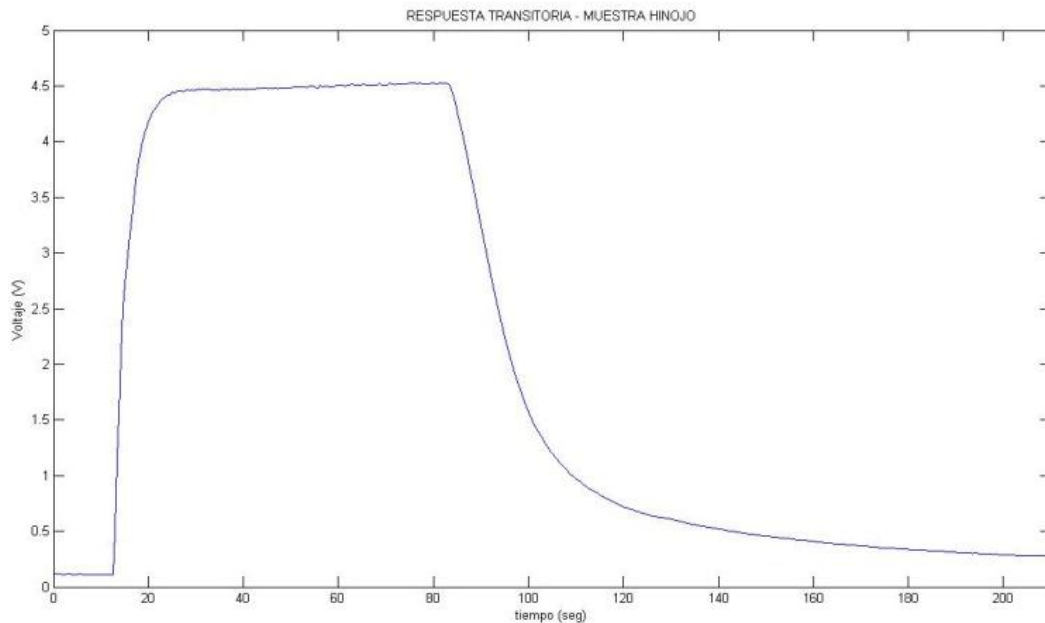
Sensor catalítico: este sensor se encuentra formado por un hilo de platino recubierto de un óxido metálico. Su principio de detección se basa en que una mezcla de gas pasa por la superficie del óxido metálico y se produce una combustión del gas. El calor producto de la combustión hace que el hilo de platino modifique su resistencia.

Sensor sensible a la masa: su funcionamiento se basa en un cristal cuya frecuencia de oscilación depende de la absorción del componente gaseoso. Los sensores conocidos como microbalanza de cuarzo modifican la frecuencia como respuesta a una modificación de la masa del oscilador.

Sensor de gas tipo semiconductor (MOS): consiste en una película de óxido semiconductor depositada en su sustrato. El modo de funcionamiento de estos sensores se basa en el cambio de su conductividad cuando se encuentra ante una concentración de gas. Esta reacción del gas en el semiconductor produce un cambio en la resistencia eléctrica.

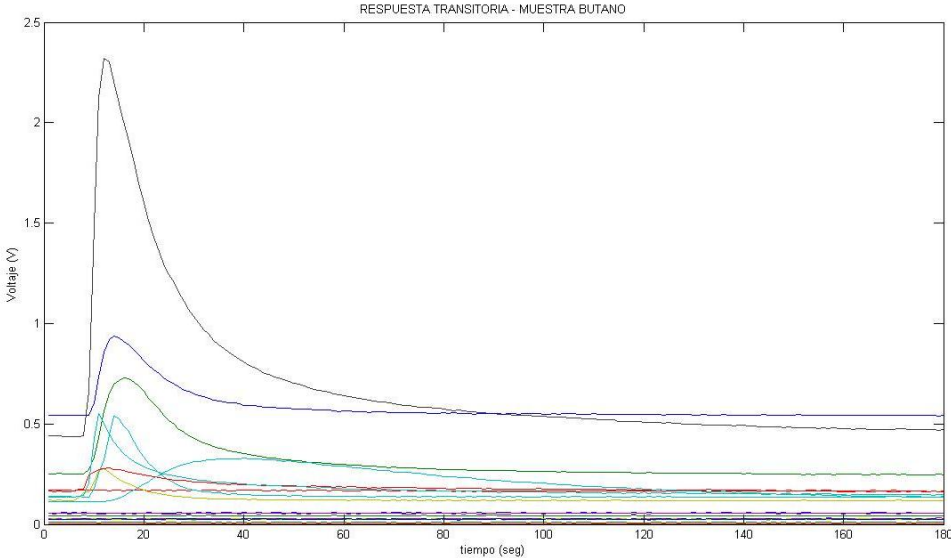
La interacción entre el compuesto químico volátil y la superficie sensitiva del cualquier sensor, produce un cambio físico y químico en el mismo. Este cambio da como resultado una señal eléctrica que puede ser capturada durante el tiempo de medida (figura 5). La respuesta típica del sensor consta de una pendiente inicial en el cual se guardó la absorción de los volátiles por el sensor en su superficie. Tiempo después este alcanza un equilibrio dinámico y los volátiles son absorbidos, lo que produce una señal constante [11]. Cuando el volátil se retira la señal del sensor muestra un descenso en su nivel, hasta retomar su valor inicial.

Figura 5. Respuesta de un sensor de gas (TGS822) ante una muestra de esencia de hinojo



Una cualidad interesante de los sensores de gases radica en su comportamiento a diferentes compuestos químicos. De hecho un sensor de gas puede responder a muchas muestras químicas y muchos sensores pueden responder de manera distinta a una misma muestra química. Por esto en una matriz de sensores, se obtiene una respuesta total que es la agrupación de las diferentes sensibilidades en mayor o menor medida que cada sensor responde a un compuesto (figura 6).

Figura 6. Respuesta de un conjunto de sensores de gases ante una concentración de gas Butano (gas de encendedor)



2. ANTECEDENTES

A nivel global, el campo de las narices electrónicas, ha marcado una creciente investigación, explotando todas las capacidades que este tipo de dispositivos puede aportar a la industria en creciente desarrollo. Las variadas tecnologías de las telecomunicaciones, han dado un punto de avance para el posible desarrollo de diferentes metodologías de terminales móviles. A continuación se presenta algunos proyectos desarrollados en estos entornos:

- En la universidad de Pamplona, se implementó una nariz electrónica, como un medio para determinar el estado de madurez del tomate de árbol, con base en los volátiles que este produce durante su proceso de maduración. Mediante esta investigación, es posible realizar un control de calidad de la fruta, estimando su nivel de madurez óptimo, para su posterior procesamiento en la industria alimenticia [12].
- En México, en el Instituto Politécnico Nacional, se desarrolló una nariz electrónica, con el objetivo de determinar el estado y calidad de los alimentos. Para la investigación, se estudió los sabores del vino, jamón, queso y papas fritas. Posteriormente se estudió mediante un espectrómetro de masas para estimar los compuestos de cada tipo de alimento. A partir de ese resultado, se estimó cuáles eran los 10 compuestos volátiles que describían el sabor de cada alimento. De ese punto de partida, desarrollaron una nariz electrónica, con el fin de evaluar la calidad del alimento [15].
- En la Universidad de la República Oriental de Uruguay, se desarrolló una interfaz USB genérica, con la finalidad de permitir conectar diferentes dispositivos electrónicos, como sensores, actuadores, displays, DAC, ADC, y gestionar estos elementos, mediante las computadoras personales de nuestro entorno. Este proyecto, plantea una contribución significativa, en la posibilidad de mejorar las diferentes problemáticas de conectividad e interacción que se puedan presentar al trabajar con este tipo de dispositivos, permitiendo facilitar la programación y el control, y en general, desarrollar soluciones prácticas [16].
- En la Universidad Industrial de Santander, se realizó una tesis de grado, cuya finalidad fue la implementación de un terminal móvil aplicado a la metodología de nariz electrónica. Este sistema pretendía combinar las capacidades de una nariz electrónica y un terminal móvil, y su desarrollo se basó principalmente en las potenciales aplicaciones que pueda brindar [17].

3. DESARROLLO DEL PROYECTO

Se propuso desarrollar el presente trabajo en 4 etapas, con el objetivo de brindar un orden en la metodología de desarrollo y ofrecer claridad en cómo se planteó la solución, para alcanzar los objetivos propuestos inicialmente.

3.1. ETAPA 1: IDENTIFICACION Y DESCRIPCION DE LA NARIZ ELECTRONICA DE PAMPLONA

3.1.1. Nariz electrónica – Universidad de Pamplona. Para el desarrollo de un Sistema de nariz electrónica (o NE) de bajo costo, practico, multipropósito y con la ventaja de poder ser compatible con las diferentes metodologías de los terminales móviles. El presente trabajo toma como punto de partida el estudio del “sistema multisensorial basado en sensores de gases químicos” desarrollado por la Universidad de Pamplona (Colombia) identificando los elementos en general que componen este equipo y cuál es su función dentro del mismo, tanto en hardware como en software, con el fin de poder brindar al desarrollador los fundamentos claves para conocer el funcionamiento de una NE [11]. En la figura 7 se observa una fotografía del equipo de nariz electrónica desarrollada por la Universidad de Pamplona.

Figura 7. Equipo de Nariz Electrónica, desarrollada por la Universidad de Pamplona



Fuente: Gualdron Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

3.1.2. Hardware de la Nariz Electrónica. El sistema de nariz electrónica o módulo multisensorial, desarrollado por la Universidad de Pamplona, a partir de un conjunto de medidas y pruebas con el fin de conocer el equipo y entender su funcionamiento, y es a partir de las diferentes prácticas que se realizaron, que se puede observar que el sistema de NE que conforman la parte de hardware, se compone en esencia de cinco módulos.

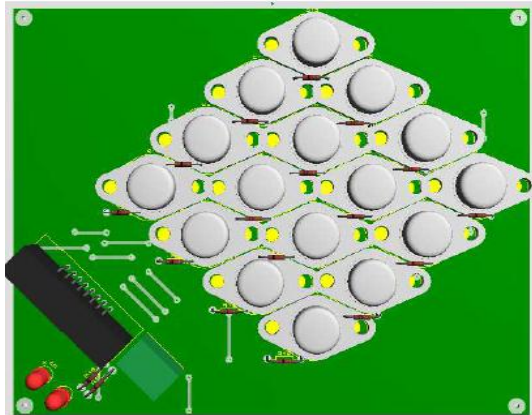
- Módulo de sensado
- Modulo neumático
- Módulo de alimentación
- Módulo de control
- Módulo de adquisición

Cada módulo que interviene en el equipo de nariz electrónica de la Universidad de Pamplona [11], se describe de manera breve y resaltando sus cualidades dentro de todo el sistema.

Módulo de sensado: este módulo corresponde a una cámara de sensores de gases y una cama de muestreo. El sistema permite colocar una muestra de olor en la correspondiente cámara de muestreo, de manera que la medición del respectivo olor se haga sin ninguna perturbación externa y poder llegar a controlar parámetros como volumen de la muestra y concentración. La figura 8 muestra una visualización 3D de la PCB desarrollada para la matriz de sensores del módulo.

La cámara de sensores de gases consta de 16 sensores de la marca Fígaro, modelo TGS. Estos sensores de gases pueden medir diferentes compuestos químicos volátiles además de ser sensibles a un compuesto en particular, esta característica hace que la nariz electrónica pueda tener una funcionalidad específica según el tipo de sensor a utilizar. Esta cámara se encuentra hermética y aislada de cualquier posible cambio en el ambiente. Mediante un sistema neumático y una red de válvulas esta interconectada con la cámara de muestreo, permitiendo un sistema de medida aislado de cualquier perturbación externa, minimizando la variación de cada medida.

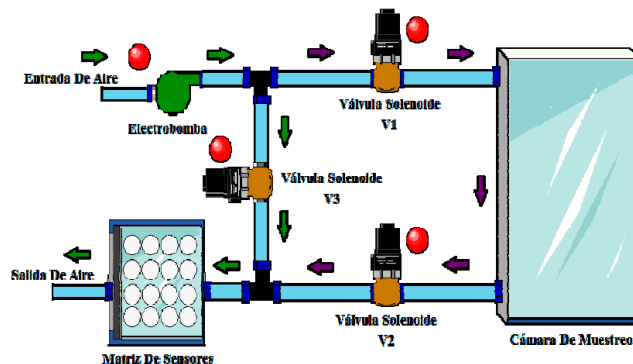
Figura 8. PCB y visualización 3D de la matriz de sensores



Fuente: Galdron Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

Modulo neumático: con el fin de emular de manera mecánica la forma como el sistema olfativo capta el olor y este es transportado hacia los receptores olfativos, la nariz electrónica de la universidad de pamplona trata este mismo método. En la figura 9 se puede apreciar el modelo del sistema neumático, el cual consta de un conjunto de electroválvulas solenoide para el control de flujo de aire entre la cámara de muestras y la cámara de sensores, además de mantener el sistema libre de compuestos volátiles entre cada medida mediante la circulación de aire limpio.

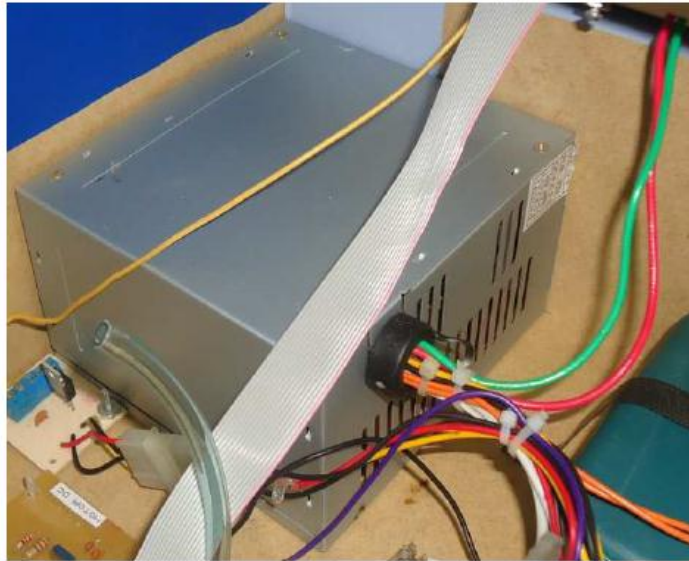
Figura 9. Visualización del funcionamiento del sistema neumático



Fuente: Galdron Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

Módulo de alimentación: la matriz de sensores, además de las válvulas solenoide, la bomba de aire y el sistema de control para las válvulas, representan un consumo eléctrico, que es suministrado por una fuente de voltaje de 600 W marca UNITEC (figura 10).

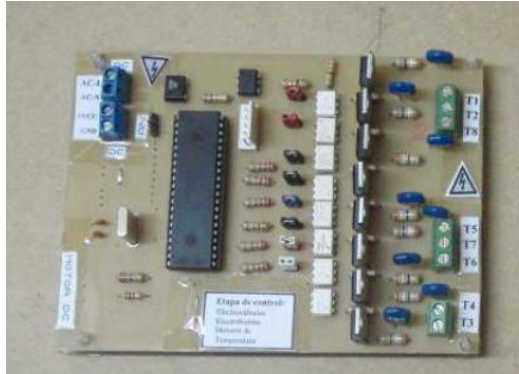
Figura 10. Visualización de la fuente de alimentación de 600W, marca UNITEC



Fuente: Gualdron Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

Módulo de control: este sistema se encarga de controlar la lógica de apertura de las electroválvulas y de encendido de la electrobomba. También el sistema puede realizar algunas operaciones de adquisición y control para dispositivos que puedan brindar un soporte lógico al subsistema. Este sistema se centra en un microcontrolador PIC16f877. La figura 11 muestra la tarjeta implementada para ese propósito.

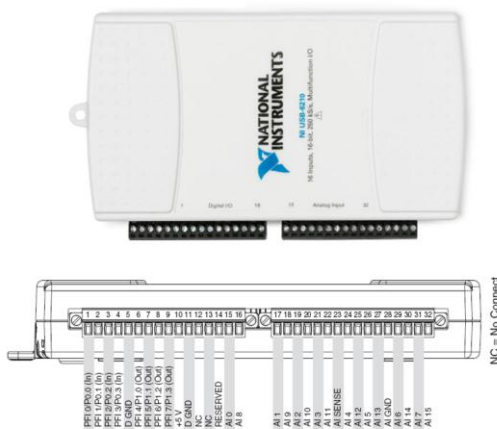
Figura 11. Visualización del módulo de control implementado



Fuente: Gualdrón Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

Módulo de adquisición: el sistema de adquisición para los sensores, como también la que ejerce la parte de control, tiene como soporte una tarjeta NI USB-6210 (figura 12). Es un módulo de adquisición de datos (DAQ) multifunción de la serie M. esta optimizado para una resolución de 16 bits, 250kS/s por canal. Este módulo tiene 16 entradas analógicas, cuatro entradas digitales, cuatro salidas digitales, cuatro rangos de entrada programable por canal, disparo digital y dos contadores/temporizadores. Para una mayor información sobre este módulo, se recomienda visitar la página oficial de National Instruments.

Figura 12. Tarjeta de adquisición NI USB-6210



Fuente: NI USB-6210. Disponible a Mar-30-2016 en:
<http://sine.ni.com/nips/cds/view/p/lang/es/nid/203223>

3.1.3. Software de la Nariz Electrónica. El sistema de hardware de la nariz electrónica es controlado por un programa de control, captura, análisis y reconocimiento de datos, llamado B-NOSE (figura 13). Este programa se desarrolló en la plataforma Windows, utilizando el paquete informático Matlab Versión 2007, mediante la herramienta de interfaz gráfica de usuario GUIDE, con la finalidad de brindar al usuario diferentes actividades y opciones para captura y análisis de los datos resultantes de cada medición.

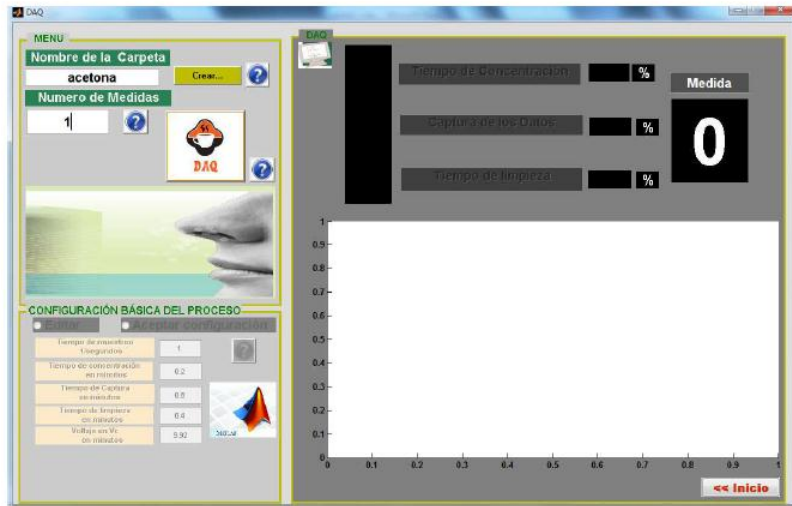
Figura 13. Ventana general del programa Bnose



Fuente: Gualdron Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

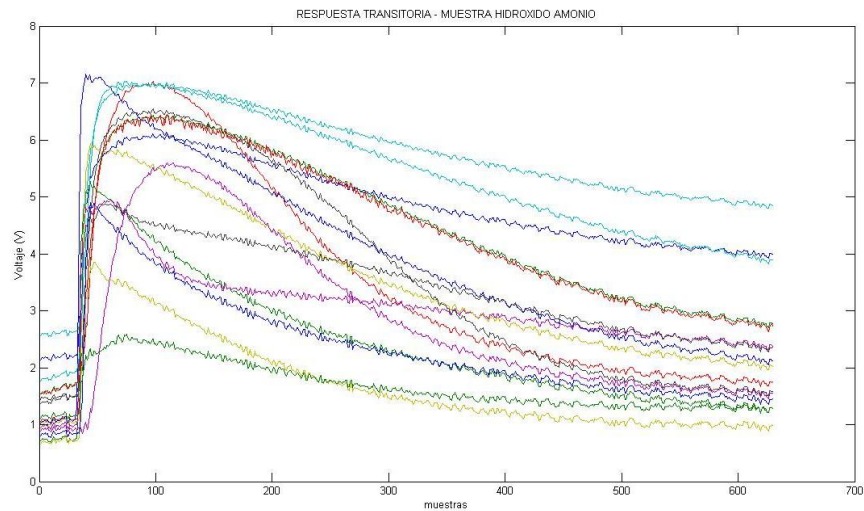
Para este estudio, se revisara los aspectos relevantes del software B-NOSE. En su ventana principal, se puede seleccionar dos accesos: MODULO DAQ encargado de la parte de adquisición y captura de datos, y la opción PROCESAR DATOS encargado de la parte de análisis de datos. En la ventana de adquisición de datos (figura 14) es posible configurar aspectos como la configuración de archivo de salida de los datos de medida, el número de medidas y los parámetros de medición, como es el tiempo de muestreo, tiempo de captura, tiempo de limpieza y voltaje V_c de alimentación. Al fijar los parámetros se puede iniciar el proceso de captura de datos, y al finalizar el proceso se puede obtener una curva de datos resultante de una muestra de olor (figura 15).

Figura 14. Ventana de la interface de adquisición



Fuente: Gualdrón Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

Figura 15. Grafica resultante del proceso de adquisición mediante la herramienta Bnose

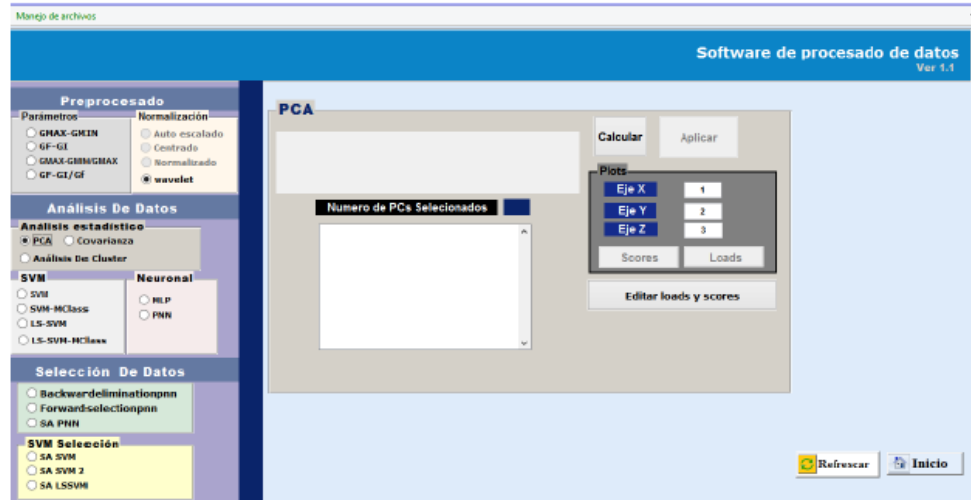


Fuente: Gualdrón Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

En la ventana de procesamiento de datos (figura 16), es posible encontrar diferentes opciones para seleccionar y analizar datos de nariz electrónica

mediante diferentes técnicas multivalentes. En la tabla 1 se resume los algoritmos principales que se encuentran implementados dentro de la herramienta.

Figura 16. Ventana de la interface de procesamiento del programa Bnose



Fuente: Gualdron Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

Tabla 1: Algoritmos principales implementados en la herramienta Bnose

Técnica	Algoritmo	Aplicación
PCA (<i>análisis de componentes principales</i>)	lineal	Clasificación
CA (<i>Análisis de covarianza</i>)	lineal	
MLP (<i>multi-layer perceptrons</i>)	neuronal	Clasificación/cuantificación
PNN (<i>Probabilistic neural network</i>)	neuronal	Clasificación
SVM (<i>Support Vector Machines</i>)	No lineal	Clasificación, regresión

Fuente: Gualdron Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

3.1.4. Aspectos generales del hardware de Nariz Electrónica. El sistema de Nariz Electrónica de la universidad de pamplona [11], propone una solución básica para el desarrollo de equipos de medición de olores, y aporta un avance a la metodología de análisis de datos multivariantes y la aplicación de las redes neuronales, como punto de apoyo hacia el estudio de los sistemas de análisis y reconocimiento de patrones. Observando el hardware más de cerca (figura 17), se puede apreciar el gran tamaño y forma de algunos de los componentes que la conforman, tomando como finalidad la implementación de una versión más compacta y que pueda brindar las mismas características que la nariz electrónica desarrollada por la universidad de pamplona.

Figura 17. Equipo de nariz electrónica, vista de sus elementos en su interior

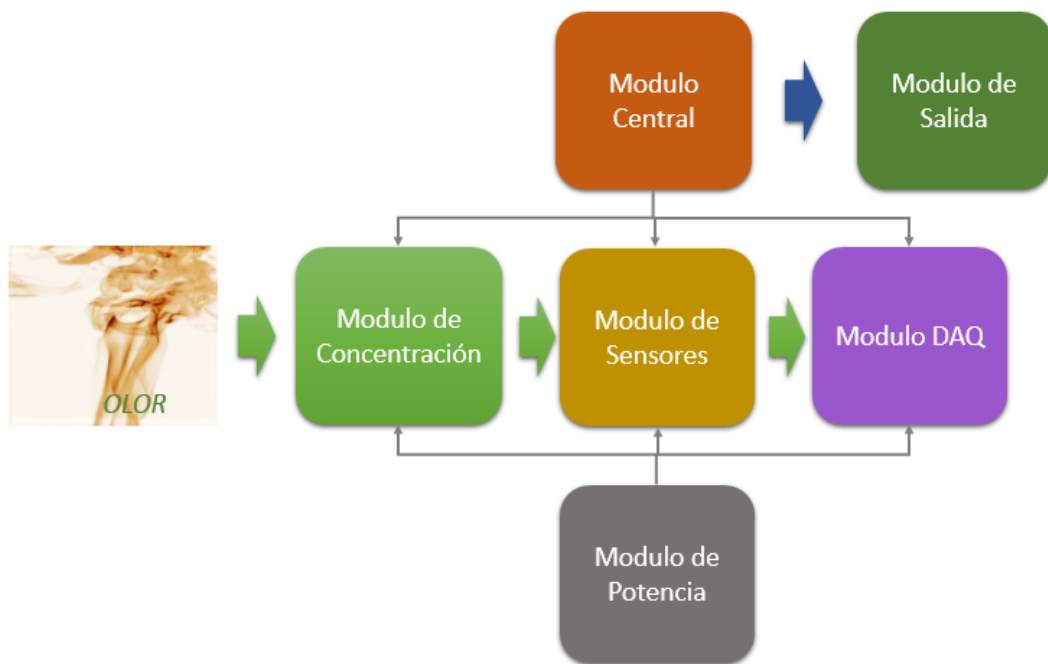


Fuente: Gualdron Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

Para una amplia visión del equipo desarrollado por la universidad de pamplona, detalles de la parte técnica del equipo y la metodología de análisis de datos utilizada, este presente trabajo sugiere al lector dirigirse a la tesis de grado “Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales” [11]. A partir de las observaciones del equipo y su funcionamiento, es posible determinar de manera práctica un modelo para el hardware de una Nariz Electrónica. En la figura 18 se observa el diagrama de bloques que representa el modelo base para la construcción de una Nariz electrónica. Una de las ventajas de observar un equipo como un conjunto de

bloques conectados entre sí, es la posibilidad de modificar o mejorar cada bloque o módulo de manera independiente y sin afectar los diferentes módulos.

Figura 18. Modelo base para hardware de Nariz Electrónica



De cada módulo, se puede describir de manera breve cuál es su función, que elementos componen el mismo y brindar al desarrollador un criterio base acerca de cada uno.

Módulo de concentración: este bloque lo conforma el sistema que permite capturar la muestra de olor. Este módulo puede variar en su topología y forma según las aplicaciones que pueda tener el sistema de nariz electrónica, como el arreglo de la cámara de muestreo, la cámara de sensores y los diferentes dispositivos encargados de controlar el proceso que permita circular la concentración de olores entre estas cámaras, al mismo tiempo de mantener el sistema limpio y con un bajo riesgo de perturbación entre medidas.

Módulo de sensores: lo conforma un arreglo de sensores de gases que intervienen como los elementos de sensado de olores. Según el tipo de sensores a elegir, la

NE puede tener una funcionalidad específica y percibir una variedad de compuestos químicos.

Módulo DAQ: lo comprende el circuito electrónico o modulo que realizara la correspondiente adquisición de datos provenientes del arreglo de sensores de gases. Este módulo debe tener en cuenta el número de sensores, el parámetro de medición de cada sensor, además de brindar una velocidad de muestreo adecuada y una buena resolución que permita captar cualquier comportamiento durante la fase de medida del olor por parte de la matriz de sensores.

Módulo de potencia: comprende la parte de alimentación y potencia del sistema en conjunto. Sea la matriz de sensores, el sistema de adquisición de datos, los controladores de las cámaras de medida o el controlador presente; además de cualquier otro circuito necesario para el buen funcionamiento, se debe tener una fuente acorde con los requerimientos de potencia total del sistema, además de las debidas protecciones en caso de un posible fallo.

Módulo central: es el encargado de dirigir toda la lógica de control en el proceso de medida y adquisición, además esta etapa supone una adecuación de los datos para posteriormente se suministren hacia un periférico de recepción de datos como un PC.

Módulo de salida: ya es la parte final del proceso de medida, donde los datos son transmitidos hacia un PC o son guardados en un sistema de almacenamiento extraíble como una memoria para posteriormente sean analizados y evaluados mediante software.

3.2. ETAPA 2: IDENTIFICACION Y DESCRIPCION DEL TERMINAL MÓVIL

3.2.1. Características generales del Terminal Móvil. El grupo de investigación RadioGIS, ha estado desarrollando diferentes metodologías para aprovechar las redes de comunicación inalámbricas actuales (Wifi, bluetooth, 3G), así también como preparar el camino a nuevas tecnologías de transmisión, procesamiento y plantear nuevas aplicaciones con el fin de desarrollar servicios para los diferentes usuarios de estas tecnologías en curso. Con la finalidad de poder brindar soluciones de ingeniería en la rama de las telecomunicaciones, uno de los desarrollos potenciales para el uso de equipos de medición o control, es el uso de dispositivos que permitan el uso de las redes móviles, aprovechando toda su

flexibilidad y capacidades, brindando un método ágil y de manera práctica para los usuarios de incluir este tipo de tecnologías en sus proyectos en desarrollo.

El fundamento de los terminales móviles (MT), es la posibilidad de conectar equipos de control, medición, o cualquier dispositivo presente con las redes móviles, el internet y los servicios de procesamiento en la nube (Cloud Computing). Por esta razón se debe hacer un planteamiento base de los requisitos de un terminal móvil, con la finalidad de dar una solución completa a un dispositivo que requiera conectarse a una red de manera portable y fácil acceso. Estos requerimientos comprenden:

- Ser una solución sencilla y de bajo costo.
- Pueda transmitir información de manera bidireccional.
- Debe ser escalable, con la posibilidad de mejoras hacia el futuro.
- Pueda utilizar alguna de las distintas redes móviles disponibles.
- Georreferenciar la comunicación.
- La solución pueda ser portable.
- Pueda integrarse con cualquier tecnología, de manera dependiente o independiente.

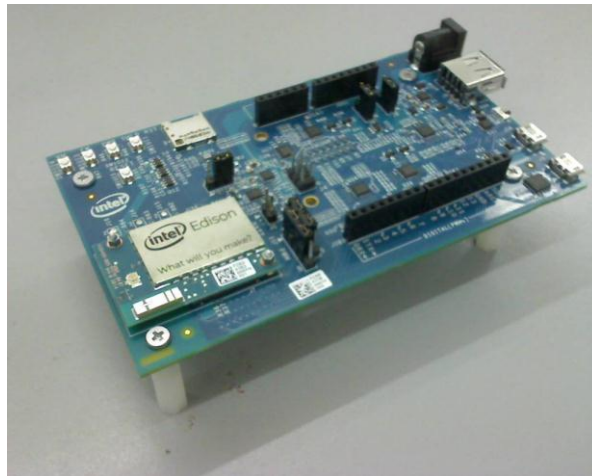
Es a partir de estos requerimientos, que la implementación de un terminal móvil sea base para la solución de diferentes dispositivos que requieran conexión con la red móvil, además de brindar una solución multifuncional y que pueda adaptarse a cualquier solución. El grupo de investigación RadioGIS a través de la tesis *“Sistema de comunicación de un terminal móvil para narices electrónicas basadas en IoT”* se desarrolló una metodología, junto con una solución en hardware y software que permitiera proyectarla como una solución IOT enfocada a la aplicación de una nariz electrónica [17].

3.2.2. Tarjeta Intel Edison. El terminal móvil desarrollado por RadioGIS, tiene como elemento base la tarjeta de desarrollo Intel Edison (figura 19). Esta tarjeta está basada en el microprocesador Dual Intel Quart Soc X1000 de arquitectura x86 a 400Mhz. Esta tarjeta presenta unas características, que permiten al desarrollador ver su potencial, de entre ellas se pueden resaltar:

- Puede programarse mediante la IDE Intel Galileo, de igual manera que las tarjetas de desarrollo Arduino.
- Soporta la mayoría de los módulos o shields para Arduino, lo cual hace que en la tarjeta pueda migrarse cualquier proyecto anterior basado en las tarjetas de desarrollo Arduino.
- Incluye modulo para Micro SD.
- Puede manejar comunicación Wifi y bluetooth.
- Tiene periféricos USB para diferentes posibilidades de comunicación de la tarjeta.

Además como un elemento adicional, mediante la Micro SD que se le añade a la tarjeta, en esta puede instalarse un sistema operativo embebido basado en Linux/Yocto. Este sistema operativo embebido permite extender el marco de desarrollo de cualquier proyecto basado en Linux, permitiendo desarrollar proyectos basados en Arduino, Python, Shell, C y Json. Según las características necesarias para un proyecto, es posible construir el sistema embebido y modificarlo según las necesidades del programador o el proyecto a ejecutar.

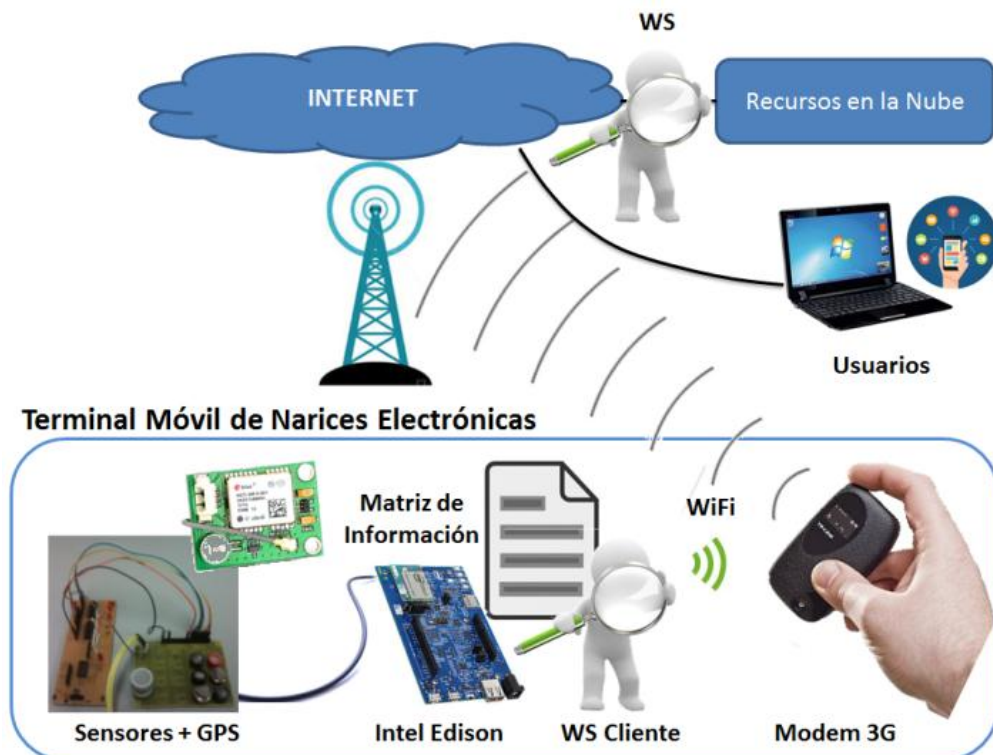
Figura 19. Tarjeta Intel Edison



Fuente: tarjeta Intel Edison. Disponible a Mar-30-2016 en: <http://blog.theinventorhouse.org/wp-content/uploads/2014/12/Picture-9.jpg>

3.2.3. Terminal móvil de nariz electrónica de RadioGIS. En la figura 20, se puede observar de manera general la metodología empleada en la versión del terminal móvil de nariz electrónica desarrollada por el grupo de investigación RadioGIS en la tesis de grado “*Sistema de comunicación de un terminal móvil para narices electrónicas basadas en IoT*” [17]. Básicamente su solución consiste en una programación que es efectuada a partir de la generación de un archivo de medición en .txt de un conjunto de sensores de gases predispuestos. Gracias a la capacidad de la Intel Edison de conectarse fácilmente mediante un módulo Wifi incorporado, este se conecta a un modem Wifi, y mediante un código basado en Python, este toma el archivo .txt y lo transmite hacia un servidor web, donde se encuentra una aplicación que toma la información y permite visualizar los resultados guardados en el archivo transmitido.

Figura 20. Metodología de Terminal móvil desarrollado



Fuente: Conde John, Rangel Johan (2015). Sistema de comunicación de un terminal móvil para narices electrónicas basadas en IoT. Trabajo de grado para optar al título de ingeniero Electrónico, Universidad industrial de Santander.

3.2.4. Programación implementada en el Terminal Móvil. La programación del terminal móvil, se basó en un conjunto de scripts en Wiring, Python y PHP, que permitía tomar la posición actual mediante GPS, y posteriormente toma los datos adquiridos de un sistema de sensores y guardados en un archivo de texto en la memoria SD, para luego poder transmitirlo hacia un servidor web, donde se recibirán los datos transmitidos, se ordenaran y se procesaran dentro del mismo, para finalmente suministrar la información final al usuario mediante una aplicación web (figura 21).

Figura 21. Aplicación web, implementada para la visualización de la información transmitida por el terminal móvil

Inicio · Medidas

Medidas

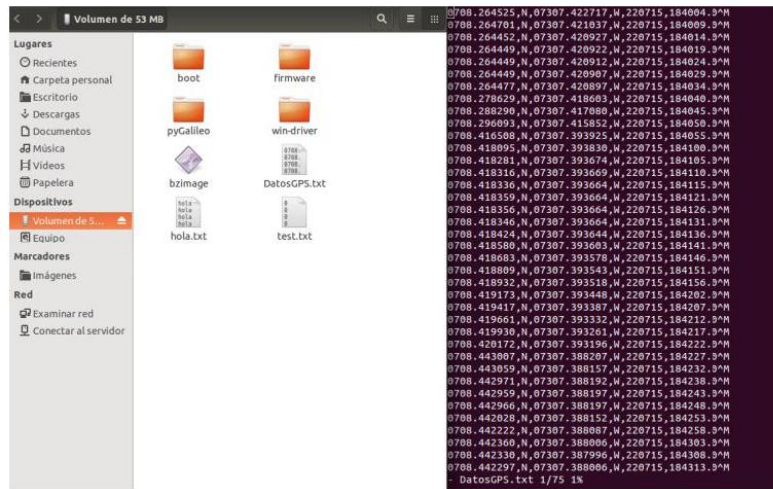
Mostrando 1-20 de 406 elementos.

ID	Fecha Toma	Fecha Registro	Coordenadas	Etiqueta	Sensor	Tipo	Comentario	
1418	28/07/15 16:47:09	28/07/15 16:47:13	07°52'N, 73°27'W	(no definido)	Sensor Nariz	Watts	(no definido)	
1417	28/07/15 16:47:08	28/07/15 16:47:13	07°52'N, 73°27'W	(no definido)	Sensor Nariz	Watts	(no definido)	
1416	28/07/15 16:47:08	28/07/15 16:47:12	07°52'N, 73°27'W	(no definido)	Sensor Nariz	Watts	(no definido)	
1415	28/07/15 16:47:07	28/07/15 16:47:11	07°52'N, 73°27'W	(no definido)	Sensor Nariz	Watts	(no definido)	
1414	28/07/15 16:47:06	28/07/15 16:47:11	07°52'N, 73°27'W	(no definido)	Sensor Nariz	Watts	(no definido)	

Fuente: Conde John, Rangel Johan (2015). Sistema de comunicación de un terminal móvil para narices electrónicas basadas en IoT. Trabajo de grado para optar al título de ingeniero Electrónico, Universidad industrial de Santander.

Además los desarrolladores realizaron diferentes pruebas para georreferenciar la información mediante una aplicación GPS de fácil uso (figura 22).

Figura 22. Visualización de los datos tomados por el GPS



Fuente: Conde John, Rangel Johan (2015). Sistema de comunicación de un terminal móvil para narices electrónicas basadas en IoT. Trabajo de grado para optar al título de ingeniero Electrónico, Universidad industrial de Santander.

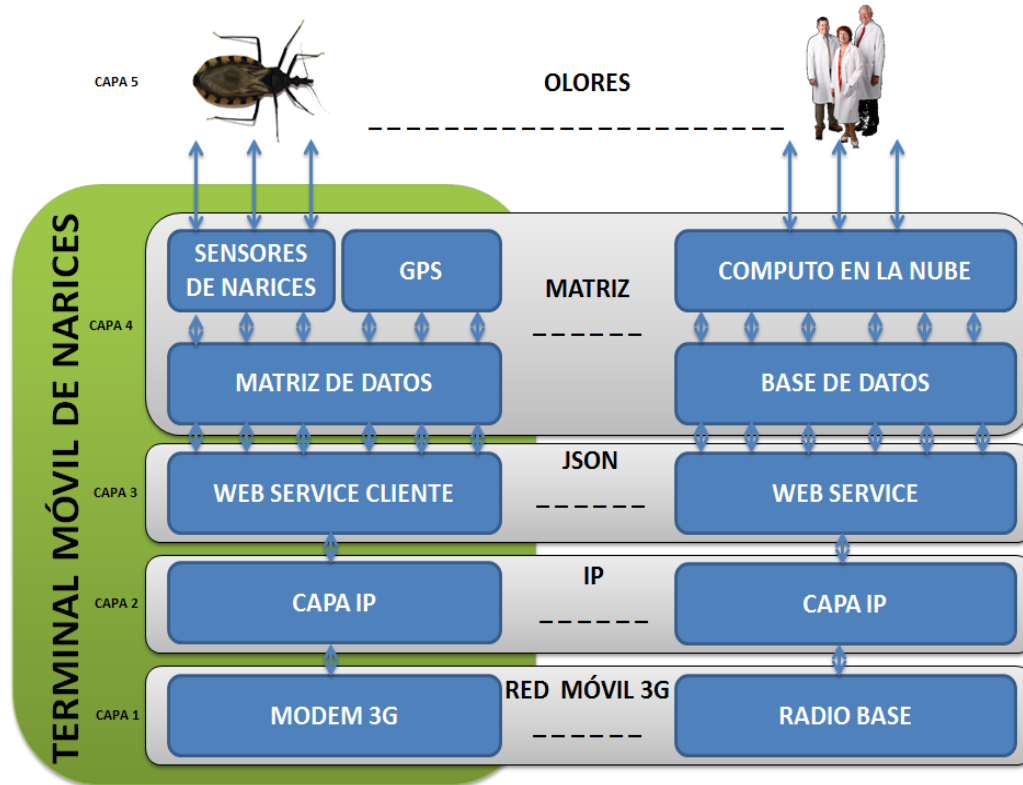
3.2.5. Aspectos generales del Terminal móvil de nariz electrónica de RadioGIS. Esta propuesta de Terminal móvil para nariz electrónica [17], presenta las siguientes características, las cuales son puntos potenciales para destacar en el presente trabajo:

- Gracias a la flexibilidad del sistema operativo embebido que presenta esta tarjeta de desarrollo, la solución puede ser modular, y adaptar su programación a diferentes tarjetas de programación que manejen la misma metodología de sistema embebido (ej.: Raspberry, Beaglebone, Intel galileo, entre otros).
- Esta propuesta de terminal móvil demuestra ser práctica, portable, y puede adaptarse fácilmente a una solución particular, según sea los requerimientos del desarrollador.
- es sencillo, fácil de programar, fácil de usar y pueda servir como material pedagógico para cualquier estudiante o practicante que quiera proponer algún desarrollo o aporte a este tipo de tecnologías.

La metodología de terminal móvil para nariz electrónica propuesta en la tesis *“Sistema de comunicación de un terminal móvil para narices electrónicas basadas en IoT”* propone una solución global, cuya propósito central es un servicio general que pueda satisfacer las necesidades de transmisión de datos, aprovechando al

máximo las diferentes tecnologías actuales, y lograr una solución sencilla y eficiente. En la figura 23 se puede observar el modelo de capas de la solución global.

Figura 23. Modelo general de un servicio basado en nariz electrónica



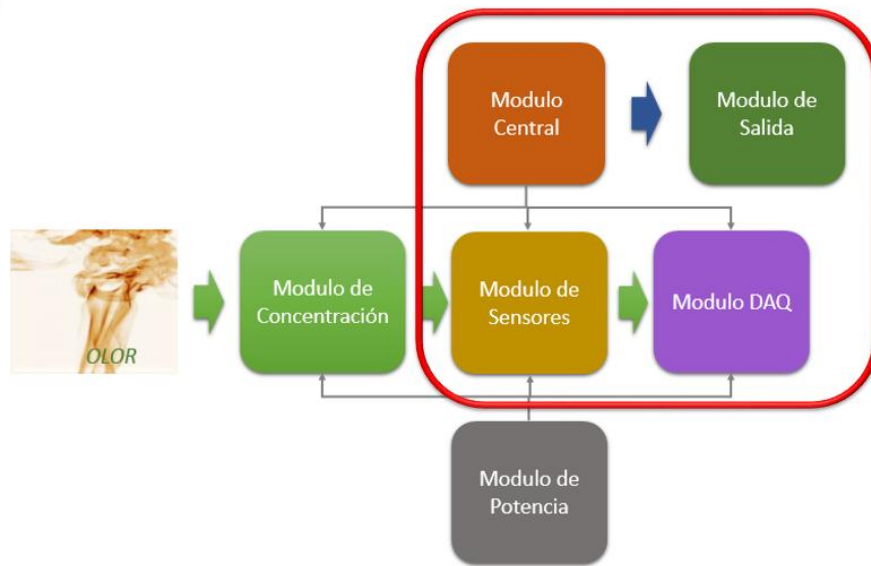
Fuente: Conde John, Rangel Johan (2015). Sistema de comunicación de un terminal móvil para narices electrónicas basadas en IoT. Trabajo de grado para optar al título de ingeniero Electrónico, Universidad industrial de Santander.

3.3. ETAPA 3: DISEÑO DEL SISTEMA DE NARIZ ELECTRÓNICA

3.3.1. Metodología de diseño. Una nariz electrónica es un equipo formado por un conjunto de módulos que trabajan en forma unida con el objetivo de analizar de manera cualitativa y/o clasificar una muestra de olor. Este equipo se compone de 6 partes básicas, de las cuales este trabajo se concentrara en el diseño e implementación de 4 de ellas, que son el módulo de sensores, módulo de adquisición, módulo de control y módulo de salida de información (figura 24). En el

presente trabajo se describe de manera general la metodología de desarrollo de cada módulo.

Figura 24. Modelo general del sistema de nariz electrónica y bloques desarrollados



3.3.2. Parte 1: módulo de sensores. Dentro del módulo de sensores, se encuentra una matriz de sensores de gases. Se diseñó con la intención de que fuera una solución práctica, escalable, y con la posibilidad de fáciles mejoras. En la figura 26 se puede observar la matriz de sensores desarrollada que conforma este módulo. En total se pueden conectar hasta 16 sensores de la marca Figaro o marca Hanwei, serie 8. En la figura 25 se puede observar el esquemático del diseño implementado

Figura 25. Esquemático de la matriz de sensores

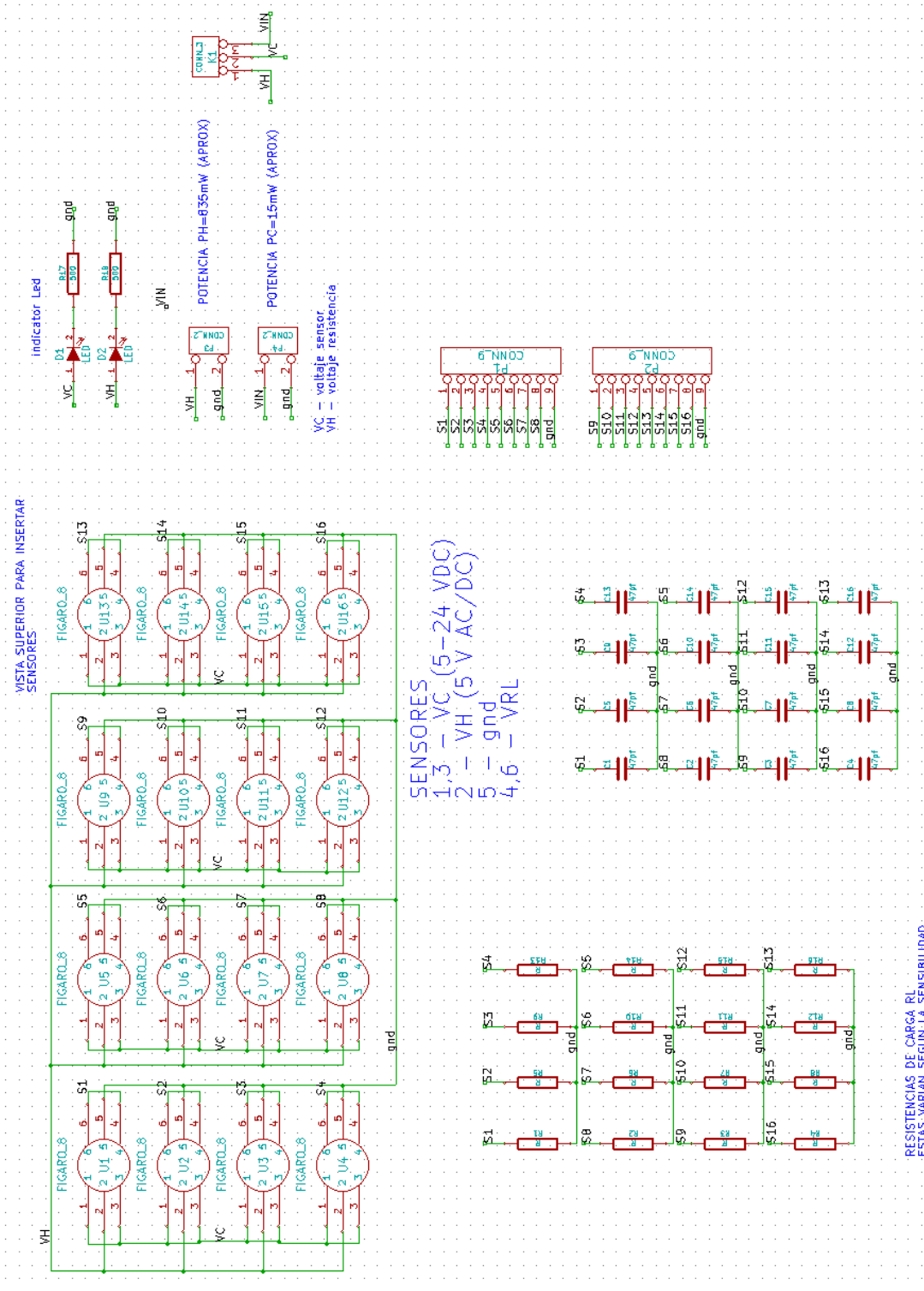
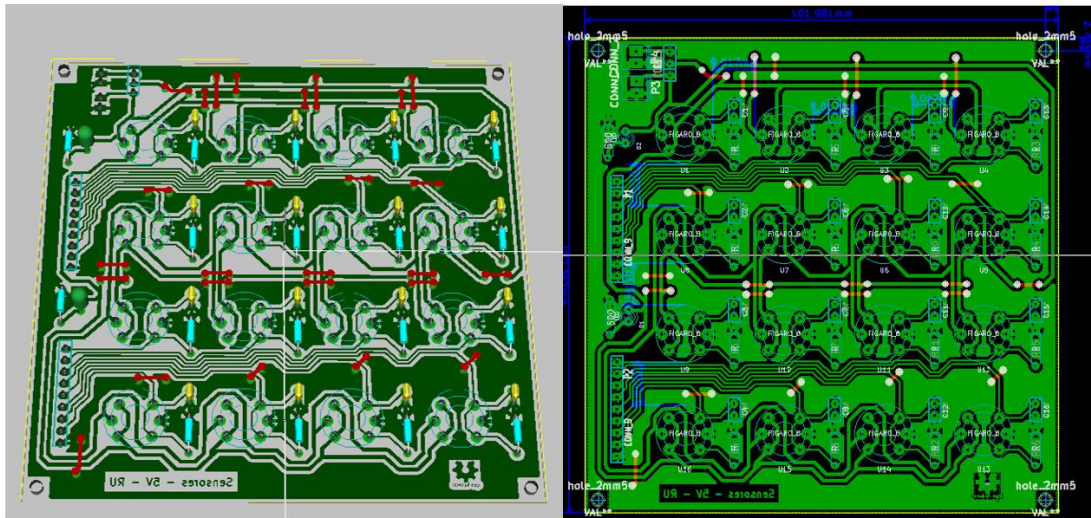


Figura 26. PCB y visualización 3D de la matriz de sensores



La base de la matriz de sensores, son propiamente los sensores de gases, los utilizados finalmente en el proyecto son del tipo semiconductor de óxido de estaño (mosfet). Este tipo de sensor basa su funcionamiento en el cambio de la resistencia presente en su capa activa, ante la presencia de un compuesto volátil, este tipo de sensor se cataloga como sensor resistivo.

Este tipo de sensores son los más utilizados en el mercado de sensores de gases puesto que presentan una alta sensibilidad a diversos compuestos orgánicos volátiles, se puede implementar en equipos portátiles y su costo es bajo. También son sensores de fácil uso y no requieren de instrumentación compleja para su implementación.

La tabla 2 muestra los sensores de la marca Fígaro y Hanwei que conforma el modulo, en ella se especifica el modelo de sensor utilizado, su aplicación general y que tipo de gases detecta con gran sensibilidad. Aunque cada sensor esta optimizado para responder de manera versátil a un determinado volátil o vapor, este puede responder además ante una gran variedad de compuestos y gases, de manera que cada sensor presenta un espectro de olores diferente, según la capacidad propia de la capa activa de cada sensor.

Tabla 2: Características generales de los sensores implementados en la matriz de sensores

Sensor	Detección	Aplicación General
tgs813	Metano, butano, alcohol, propano, hidrogeno, CO.	Gases en general, calidad del aire.
tgs821	Hidrogeno, metano, etanol, CO.	Detección de Hidrogeno.
tgs822	Metano, butano, alcohol, etanol, hexano, CO, benceno, acetona.	Industria, detección de solventes y vapores.
tgs826	Amoniaco, butano, etanol, hidrogeno.	Detección de amoniaco.
tgs832	Etanol, R-22, R-134a, R-410a, R-404a, R-407c.	Detección de gases refrigerantes.
MQ-2	Metano, butano, lpg, smoke, alcohol, propano, hidrogeno, CO.	Industria, hidrocarburos.
MQ-3	Metano, lpg, smoke, alcohol, etanol, hexano, CO.	Detección de alcohol y solventes.
MQ-4	Metano, lpg, smoke, alcohol, propano, CO.	Industria, hidrocarburos.
MQ-6	Metano, butano, lpg, smoke, alcohol, etanol, propano, hidrogeno, CO.	Industria, hidrocarburos, control de calidad del aire.
MQ-7	Metano, lpg, alcohol, etanol, hidrogeno, CO.	Industria, detección de CO.
MQ-9	Metano, lpg, CO.	Hidrocarburos, industria.
MQ-135	metano, smoke, alcohol, etanol, amoniaco, CO, CO2, benceno	Control de calidad del aire.

Fuente: [18]

En la tabla 3 se muestra la disposición de los canales de la matriz de sensores implementados, que servirán de referencia para la conexión con los canales del dispositivo de adquisición.

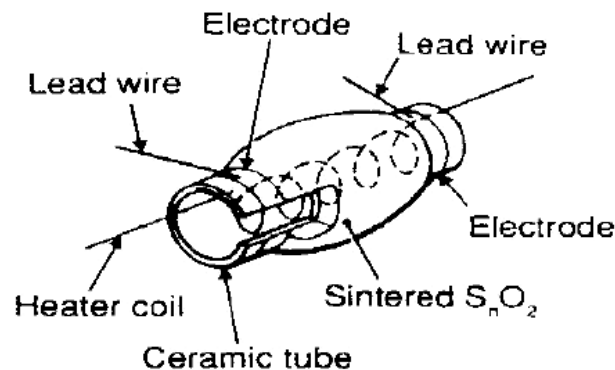
Tabla 3: Canales de la matriz de sensores

Sensor de la matriz	Referencia de sensor
S1	TGS826
S2	TGS822
S3	TGS813
S4	TGS832
S5	TGS821
S6	MQ-9
S7	MQ-2
S8	TGS813
S9	MQ-4
S10	MQ-7
S11	MQ-135
S12	MQ-6
S13	MQ-3
S14	---
S15	---
S16	---

3.3.2.1. Sensores de gases. Los sensores de gases de la marca Fígaro y Hanwei, son catalogados como sensores resistivos, y como una característica propia de este tipo de sensores son su fácil uso y su circuito de alimentación es relativamente práctico para cualquier aplicación.

Para la implementación se han utilizado sensores de la serie 8 (figura 27). Este modelo de sensores se caracteriza por tener un heater adentro de una estructura cerámica de alúmina, depositada sobre una superficie tipo cilindro con dos electrodos en oro. Para conseguir una alta sensibilidad y buena estabilidad del material a largo plazo, este se sintetiza por un largo periodo y bajo condiciones de temperatura y presión controladas.

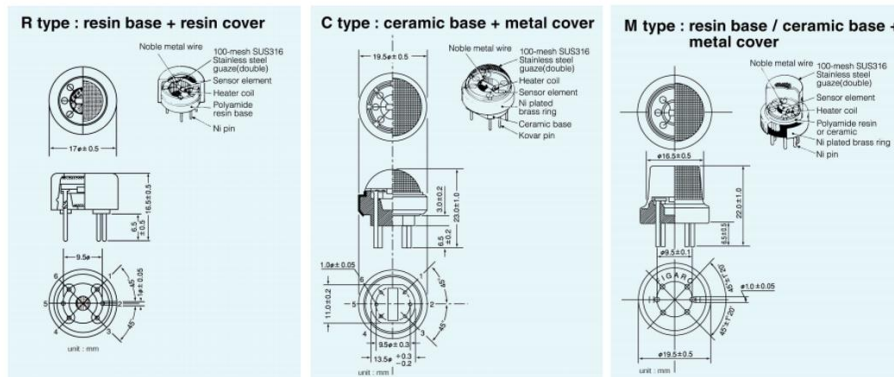
Figura 27. Esquema interno de los sensores serie 8



Fuente: FIGARO GAS SENSORS, Figaro Gas Sensors Product Catalogue. Disponible a Mar-30-2016 en: https://jenslabs.files.wordpress.com/2013/06/figaro_tgs_serien.pdf

Los sensores de la serie 8 presentan 3 tipos de encapsulados, mostradas en la figura 28. El encapsulado tipo R corresponde a una configuración estándar, de base y encapsulado en resina. El tipo C tiene la base en cerámica y el encapsulado en metal. Y por último el encapsulado tipo M tiene la base en resina/cerámica y el encapsulado en metal.

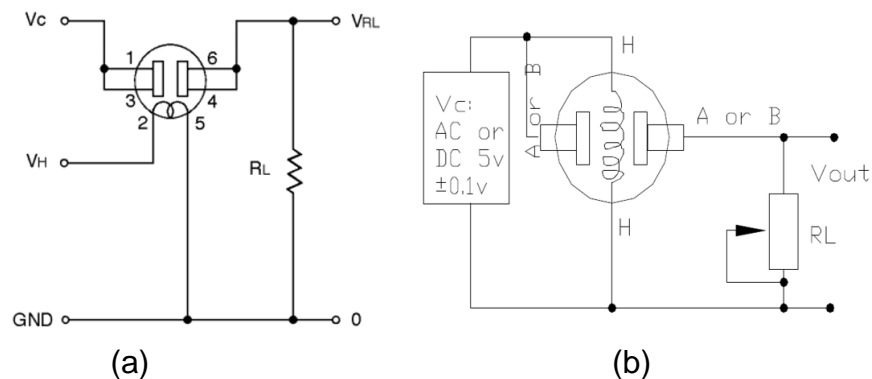
Figura 28. Versiones de encapsulado de los sensores de la serie 8 tipos R, C y M



Fuente: FIGARO GAS SENSORS, Figaro Gas Sensors Product Catalogue. Disponible a Mar-30-2016 en: https://jenslabs.files.wordpress.com/2013/06/figaro_tgs_serien.pdf

En la figura 29 se puede observar el esquema general del circuito del sensor, suministrado por los fabricantes Hanwei y Figaro. El circuito de alimentación es bastante similar en ambas referencias y en general consiste en conectar los pines del sensor de gas con una resistencia R_L , formando con el sensor un circuito divisor de tensión, relacionando el cambio de la resistencia del sensor producto de la exposición a un volátil, con una señal de voltaje resultante que puede ser adquirida por un conversor analógico-digital (ADC) para su lectura y procesamiento. Además para que el sensor funcione adecuadamente en su zona de trabajo, se debe alimentar una resistencia calefactora por los terminales marcados con el símbolo H o V_h , esta resistencia es la que mantiene la capa activa del sensor a una temperatura elevada, del orden de los 300°C.

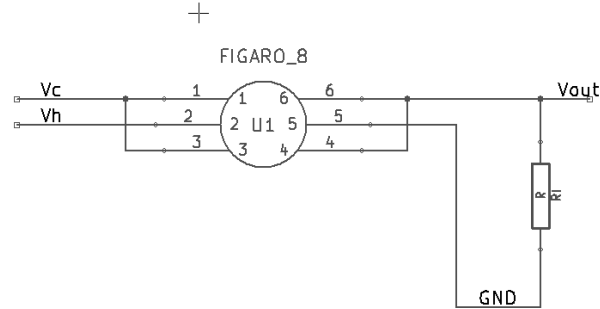
Figura 29. Esquema de circuito para sensor de gas (a) Fígaro y (b) Hanwei



Fuente: FIGARO GAS SENSORS, hojas de datos sensores Figaro. Disponible a Mar-30-2016 en: <http://www.figarosensor.com/>

La figura 30 muestra el esquema de circuito de sensor de gas serie 8 de la marca Fígaro o Hanwei. Para una mejor comprensión, el circuito base se puede apreciar mejor en la figura xx:

Figura 30. Esquema general de circuito para sensor de gas



Dónde:

R_s = Resistencia sensor (Ω)

R_l = Resistencia de carga (Ω)

V_c = Voltaje de entrada (V)

V_h = Voltaje de calefacción (V)

V_{out} = Voltaje de salida (V)

Como se describió antes, el circuito corresponde a un divisor de tensión con voltaje de entrada V_c y una conexión entre la resistencia del sensor (R_s) y la resistencia de carga (R_l), cuando el sensor se encuentra ante la presencia de un volátil, el sensor modifica su resistencia y por siguiente modifica la señal de voltaje de salida (V_{out}), de manera que la señal de voltaje de salida es (1):

$$V_{out} = \frac{R_l}{R_l + R_s} V_c \quad (1)$$

Este valor de voltaje puede ser capturado mediante un módulo de adquisición (ADC) y posteriormente procesar los datos ya almacenados. La señal de voltaje de salida puede ser usada para describir el comportamiento del sensor ante un volátil, también a partir de esta señal de voltaje se puede determinar de manera más directa la resistencia del sensor (R_s) y su cambio ante un volátil, y se puede describir por la ecuación (1):

$$R_s = \frac{Rl(V_c - V_{out})}{V_{out}} \quad (2)$$

De la ecuación (2), podemos relacionar el voltaje de salida del sensor (V_{out}) directamente con la resistencia del sensor (R_s), aunque se puede utilizar el parámetro de conductancia (G_s) mediante la expresión (3):

$$G_s = \frac{1}{R_s} \quad (3)$$

En la tabla 4 se muestra las especificaciones eléctricas generales de un sensor de gas, sea la marca Fígaro o Hanwei sus especificaciones son bastante similares.

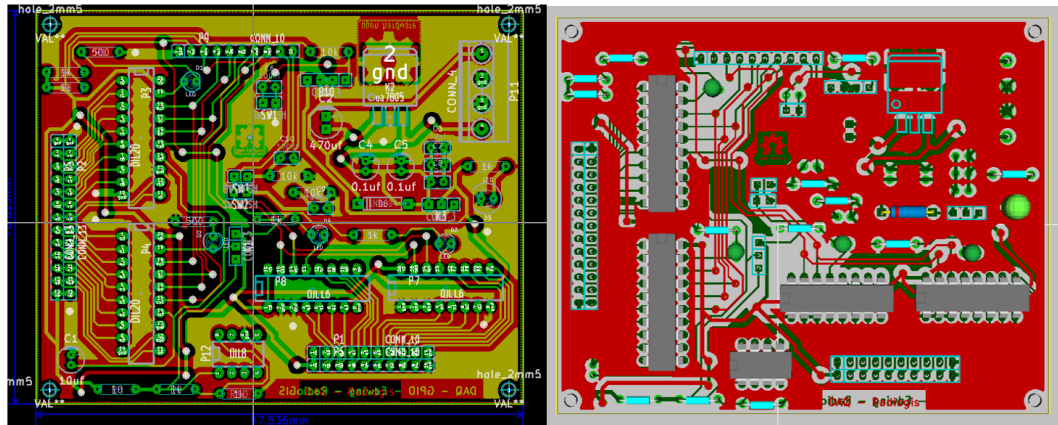
Tabla 4: Especificaciones eléctricas generales para sensor de gas Fígaro o Hanwei

Ítem	Símbolo	Valores nominales
Voltaje de la resistencia calefactora	VH	4,8-5,2 V Ac o DC
Voltaje de circuito	Vc	5-24 V DC
Resistencia de carga	RL	mínimo 0,45K Ohm
Resistencia de sensor	Rs	1K - 10 K Ohm aprox.
Consumo resistencia de sensor	Pc	15 mW aprox.
Consumo resistencia calefactora	PH	700mW@5V aprox.

Fuente: [18]

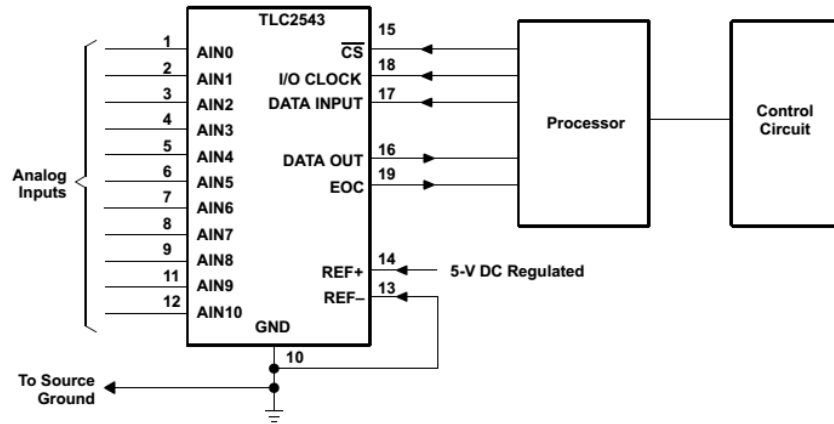
3.3.3. Parte 2: módulo DAQ. Para el diseño del módulo de adquisición, se optó por una solución de muy bajo costo, que permitiera muestrear en general las señales de 16 sensores de gas en tiempo real, con una resolución adecuada, y principalmente que permitiera realizar operaciones de acción o control, convirtiendo el modulo en una herramienta fácil y versátil para el desarrollador. En la figura 31, se presenta el prototipo desarrollado para el sistema de adquisición y control. Además de ser implementado para una nariz electrónica, este módulo puede ser usado ampliamente para diferentes propósitos y fácilmente puede ser adaptado a cualquier tipo de solución. Este módulo se basa en los integrados TLC2543 y PCF8564.

Figura 31. PCB y visualización 3D del sistema de adquisición



Este sistema se centra en 2 integrados, el primero es el TLC2543 de TEXAS INSTRUMENT, un ADC de 11 canales analógicos, 12 bits de resolución (4096 niveles), una velocidad de 66kS/s, programable para diferentes lógicas de adquisición y transmisión, su interface de comunicación maneja el protocolo SPI. Este integrado supone una solución práctica, de bajo costo y ágil para poder realizar la adquisición de múltiples sensores, y brinda una opción para diversas aplicaciones. En la figura 32 se puede observar el diagrama circuital del ADC para implementar con un microcontrolador de manera versátil.

Figura 32. Esquema general para el integrado TLC2543



Fuente: TEXAS INSTRUMENTS, Hoja de datos TLC2543. Disponible a Mar-30-2016 en: <http://www.ti.com/lit/ds/symlink/tlc2543-ep.pdf>

En la tabla 5 se resume las características más relevantes del integrado utilizado.

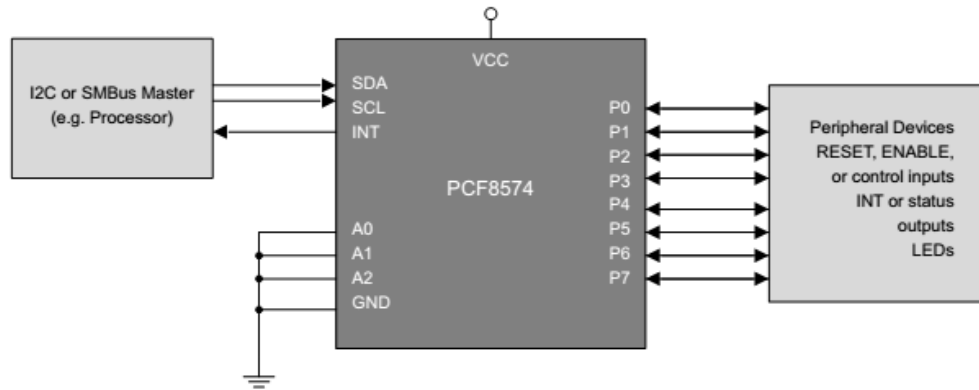
Tabla 5: Características generales del integrado TLC2543

Item	Características
Integrado	TLC2543
Voltaje de alimentación	0.5 a 6.5 V max
Resolución ADC	12 bits (0 – 4096 niveles)
Frecuencia de reloj	4.1 MHz
Tiempo de conversión (aprox)	10 - 16 us aprox.
Numero de canales ADC	11
Interface de comunicación	SPI modo 3

Fuente: [20]

El segundo integrado es el PCF8574, que es un expansor de puertos digitales programable mediante el protocolo de comunicación I2C, implementado para darle una versatilidad en el módulo para operaciones de control y lógica, de la misma manera que lo haría un microcontrolador, con utilizar solo 2 pines para comunicar y controlar dicho integrado (SDA, SCL). En la figura 33 se presenta el diagrama circuital del integrado aplicado a un microcontrolador.

Figura 33. Esquema general para el integrado PCF8574



Fuente: TEXAS INSTRUMENTS, Hoja de datos PCF8574. Disponible a Mar-30-2016 en: <http://www.ti.com/lit/ds/symlink/pcf8574.pdf>

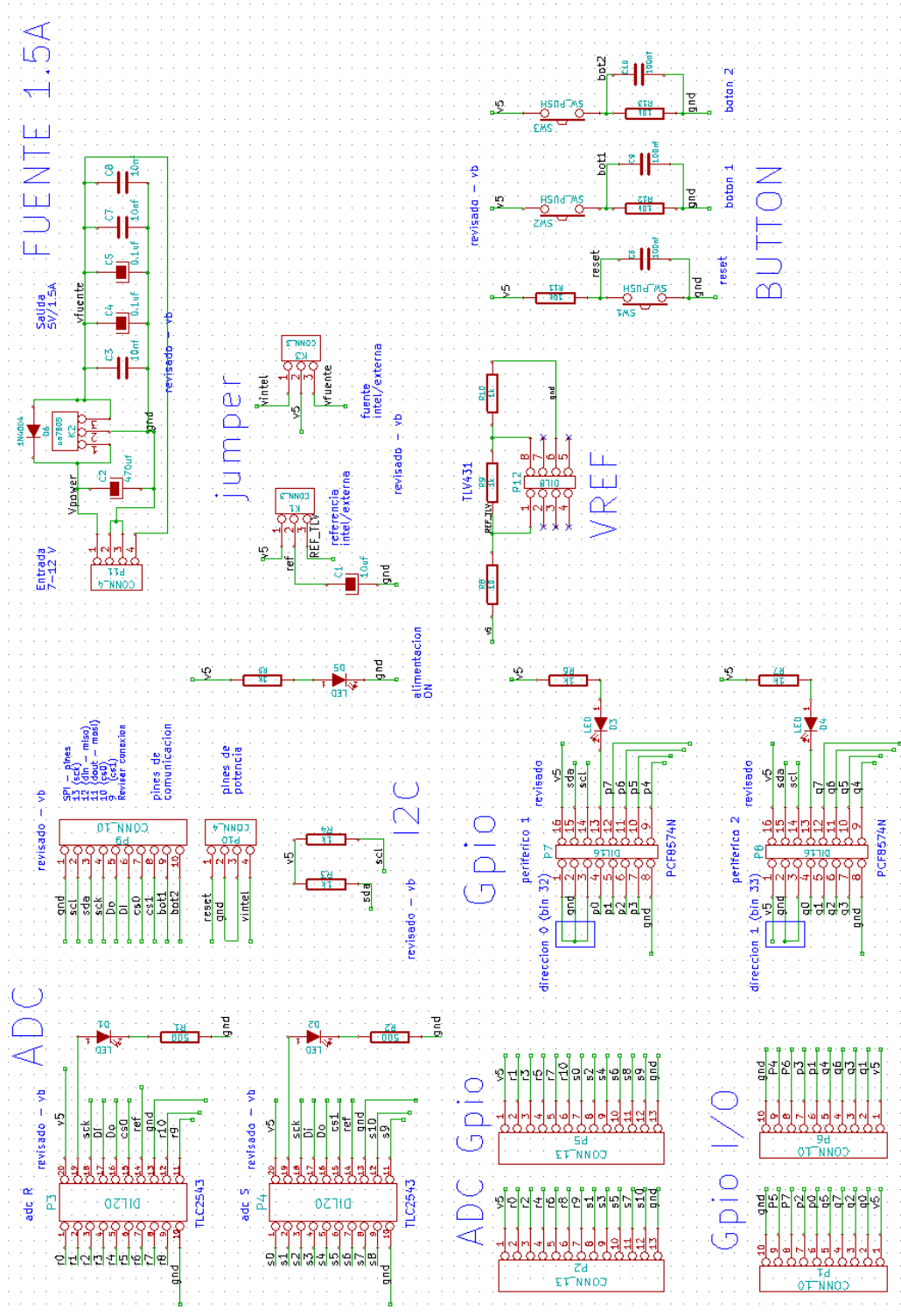
En la tabla 6 se resume las características más relevantes del integrado utilizado.

Tabla 6: Características generales del integrado PCF8574

Item	Características
Integrado	PCF8574
Voltaje de alimentación	0.5 a 7 V max.
Voltaje de pines I/O	0.5 a 7 V max.
Numero pines digitales I/O	8
Corriente util por pin	50 mA max.
Número máximo de módulos conectados entre si	8
Interface de comunicación	I2C

Fuente: [21]

Figura 34. Esquemático de sistema de adquisición y control



Como elementos adicionales al módulo, este puede ser alimentado por un microcontrolador o por una fuente externa, También utiliza una fuente de referencia para el integrado TLC2543 basado en el integrado TLV431, también conexiones para interruptores, jumpers de selección y elementos indicadores, todo con el fin de brindar un fácil uso del módulo en diferentes proyectos que involucre la necesidad realizar adquisición y control. En la figura 34 se puede observar el esquemático general del diseño implementado.

3.3.4. Parte 3: módulo central. Para el diseño y desarrollo del módulo de control, se buscó una solución en una gran variedad de tarjetas desarrolladoras, y que esta solución fuera eficiente, escalable, simple y a un bajo costo. Se optó finalmente por utilizar la tarjeta de desarrollo Intel Galileo GEN 1, ya que representa una estrategia fácil y ágil de implementación para desarrolladores. Esta tarjeta de desarrollo presenta múltiples ventajas para el programador, ya que se puede implementar diferentes lenguajes de programación como Wiring, Python, c, Shell, Json, o cualquier lenguaje instalado en su sistema operativo embebido basado en Linux/Yocto. En la tabla 7 se resume las características más relevantes que se utilizaron de esta tarjeta. La figura 35 muestra una imagen de la tarjeta implementada.

Figura 35. Tarjeta Intel Galileo Gen1



Fuente: Tarjeta Intel Galileo. Disponible a Mar-30-2016 en: <http://electronilab.co/tienda/intel-galileo-board-de-desarrollo-arduino/>

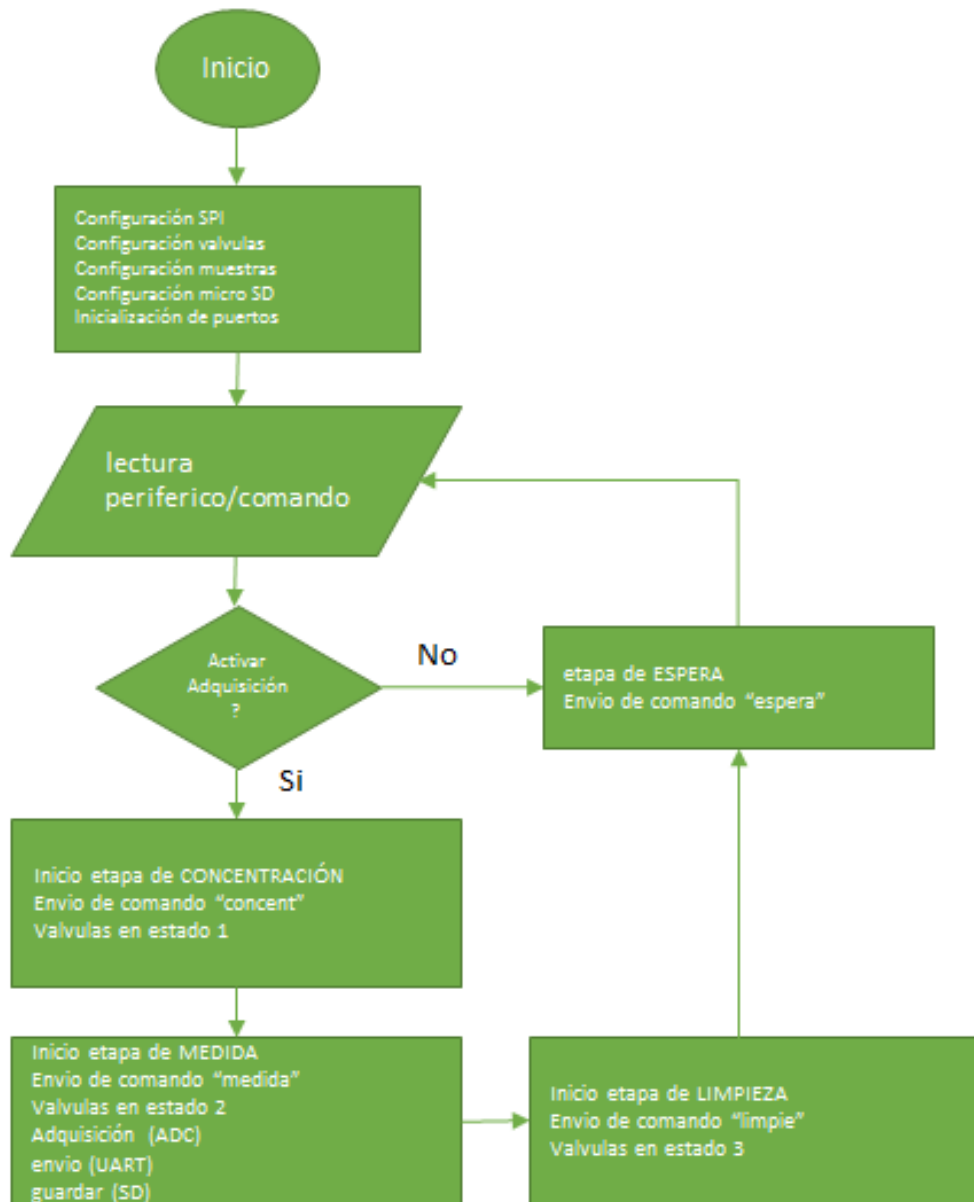
Tabla 7: Características de la tarjeta de desarrollo Intel Galileo Gen 1

Ítem	Características
Tarjeta	<i>Intel galileo</i>
Precio estimado	<i>254.500 COP</i>
Procesador	<i>Intel Quark SoCX1000 – 32bit</i>
Velocidad	<i>400MHz</i>
USB	<ul style="list-style-type: none"> • <i>USB 2.0 Host Port en conector mini- PCIe</i> • <i>Conector USB 2.0 Host</i>
Memoria	<ul style="list-style-type: none"> • <i>8 MByte de flash SPI Flash</i> • <i>512 KByte SRAM</i> • <i>256 Mbytes DRAM</i> • <i>256KByte ~ 512KByte para el almacenamiento del sketch</i> • <i>11 KByte EEPROM</i>
Especificaciones eléctricas	<ul style="list-style-type: none"> • <i>Voltaje de entrada 5V</i> • <i>Pines digitales 14</i> • <i>Entradas analógicas 6</i> • <i>Corriente pin 80mA</i>
Lenguajes de programación	<i>Wiring, Python, C, Shell, Json</i>
Comunicación	<ul style="list-style-type: none"> • <i>Conector Ethernet 10/100</i> • <i>Conector Wifi PCI Express mini</i> • <i>USB 2.0</i> • <i>2 puertos serial UART</i> • <i>I2C</i> • <i>SPI master</i> •
Almacenamiento extraíble	<i>Socket para tarjeta micro SD ofrece hasta 32GByte de almacenamiento</i>

Fuente: [22]

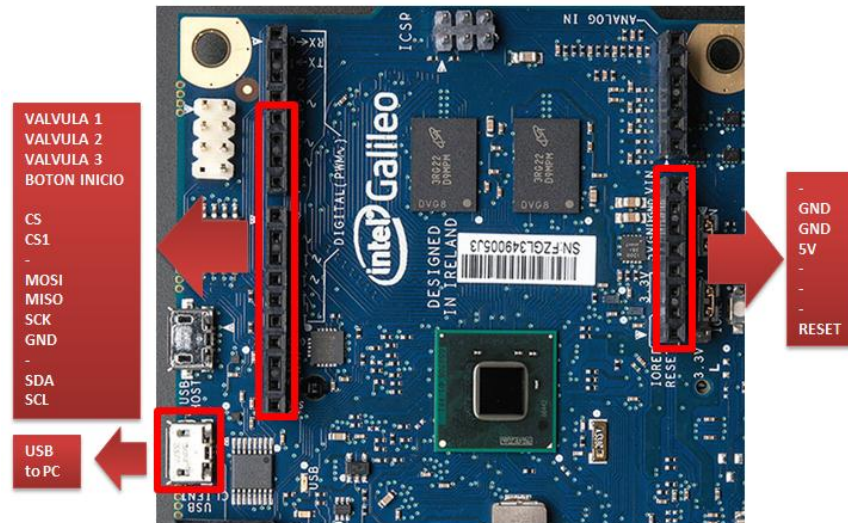
La programación de la tarjeta de desarrollo fue realizada en lenguaje Wiring para la Intel Galileo mediante el software IntelArduino IDE, suministrado por la empresa Intel. Dado que Wiring representa uno de los lenguajes de programación más utilizados e implementados por su fácil sintaxis, además de poder adaptarse a cualquier familia de microcontroladores que tengan soporte en Wiring, su desarrollo representa una opción clave para adaptar la lógica de programación para diferentes propósitos o incluir de manera organizada alguna lógica de control especial y necesaria para cualquier aplicación específica. El código que se utilizó se simplifica en el diagrama de la figura 36 el cual es muy fácil de usar, adaptar y modificar, según sea las necesidades del programador.

Figura 36. Diagrama de flujo del programa implementado para el sistema de adquisición en la tarjeta de desarrollo Intel Galileo



En la figura 37 se observa la implementación de los pines utilizados para la comunicación y control entre la tarjeta de desarrollo Intel Galileo y la tarjeta de adquisición y control.

Figura 37. Esquema de los pines implementados en la tarjeta de desarrollo Intel Galileo



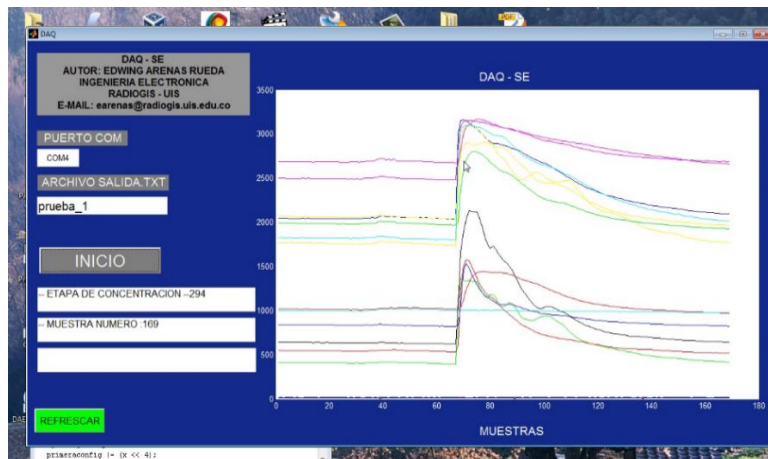
3.3.5. Parte 4: módulo de salida. Este módulo, se desarrolló con 4 propósitos básicos:

- Controlar el sistema de adquisición de los sensores de gas de manera general.
- Visualizar en tiempo real el proceso de adquisición.
- Capturar y guardar los datos adquiridos a un archivo de texto, para poder procesar la información por cualquier herramienta informática, además guardar una gráfica que muestre el comportamiento transitorio de la medida.
- Ofrecer una metodología práctica, de manera que se pueda adaptar de manera rápida a mejoras, diferentes desarrollos o para una aplicación específica.

A partir de estos requisitos, se propuso 2 metodologías para el control y adquisición de datos de los sensores.

3.3.5.1. Metodología 1. Como primera solución se propuso un sistema básico bajo la plataforma Windows utilizando la herramienta de computo MATLAB versión 2010 mediante la herramienta GUIDE que permite crear una interfaz de usuario práctica y aprovechando el poder computacional de Matlab. En la figura 38 se muestra la ventana general donde el usuario debe ingresar en la ventana los datos relacionados con el puerto COM en el cual se encuentra conectado el hardware de adquisición y el nombre de archivo de salida que se quiere generar al finalizar los resultados.

Figura 38. Interfaz de adquisición para equipo de nariz electrónica, implementado mediante la herramienta MATLAB



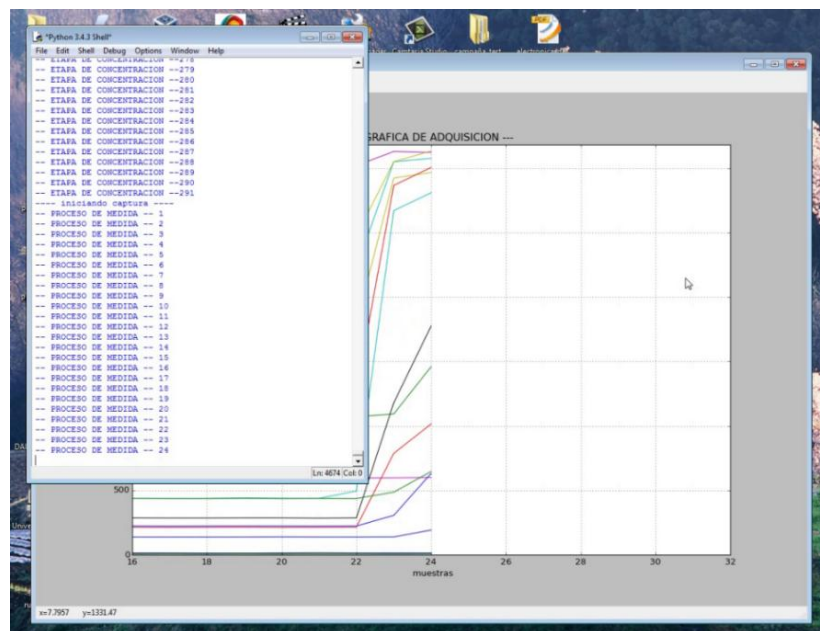
Al presionar el botón INICIO se dará el inicio al sistema de adquisición, y permitirá al usuario observar todo el proceso de concentración, medida y limpieza en tiempo real además de visualizar el estado de cada etapa, junto con el progreso de la medición total. Al finalizar el programa, el sistema dará un mensaje indicando su finalización correcta, siguiendo por una gráfica final de todo el proceso de medida, y en la carpeta actual del programa DAQ se encontrara un archivo con extensión txt y otro con extensión .png cuyo nombre el usuario ha propuesto inicialmente.

Para volver a realizar una nueva captura basta con dar click en el botón REFRESCAR y escribir un nuevo nombre de archivo de salida, verificar el puerto COM y dar click en el botón INICIO para generar un nuevo proceso de adquisición. La programación está ajustada para que el usuario pueda realizar el proceso de manera fácil, evitando posibles errores de ejecución o fallos producto del sincronismo entre la aplicación y la tarjeta de desarrollo, como también la posibilidad de evitar inicios en falso en las medidas.

La inicialización de todo el proceso de medida requiere con anterioridad haber programado en la tarjeta de desarrollo los parámetros de los tiempos de concentración, medida y limpieza, además de ajustar los archivos que serán guardados en la memoria microSD y que servirán de respaldo para eventos donde se realice el proceso de adquisición y los datos resultantes se salven dentro de un archivo en la misma tarjeta de desarrollo.

3.3.5.2. Metodología 2. Como una segunda solución se propuso utilizar el lenguaje de programación PYTHON como una herramienta muy versátil para el sistema de adquisición, también ofrece la gran opción de adaptar esta solución a cualquier plataforma o sistema embebido que tenga soporte en Python. En la figura 39 se muestra el proceso de adquisición mediante la herramienta la IDE de Python, en ella se ofrece la posibilidad de controlar todo el proceso de adquisición mediante comandos de texto, además de configurar los tiempos de concentración, medida y limpieza, haciendo que el proceso sea totalmente práctico para el usuario.

Figura 39. Vista de la interfaz de adquisición implementada mediante la herramienta PYTHON

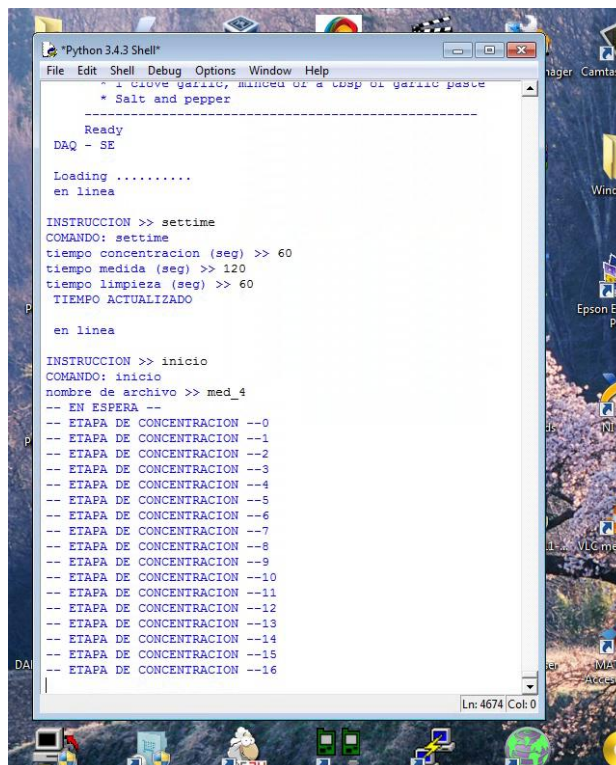


Al dar inicio a la aplicación PYTHON, y posteriormente al cargar el archivo DAQ.py en la carpeta de programas, se ejecutara la ventana de comandos del programa de adquisición para nariz electrónica. En la figura 40 se muestra la interfaz de comandos del programa de adquisición, de una manera práctica es posible

adaptar la lógica de comandos y programación para cualquier tipo de solución, según sean los requerimientos del desarrollador.

Actualmente el sistema está adaptado para responder a 3 comandos de ejecución. El comando settime permite al usuario ingresar los datos relacionados con el tiempo de concentración, medida y limpieza en segundos. El comando inicio permite al sistema de adquisición realizar la actividad de medida. El comando salir dará la finalización al programa. Inmediatamente al dar el comando inicio, el usuario debe establecer el nombre del archivo de salida que al finalizar el proceso, estará en extensión .txt y .png que permitirá al desarrollador tomar los datos finales de la medición para poder analizarlos en cualquier herramienta informática. Al iniciar el proceso de adquisición, el sistema mostrara al usuario el estatus del proceso, así como la variación de los sensores durante todo el proceso de medida en tiempo real, de una manera limpia y sencilla.

Figura 40. Vista de la interfaz de adquisición de PYTHON, donde se puede observar la configuración de tiempos y el inicio del proceso de medida



```
Python 3.4.3 Shell
File Edit Shell Debug Options Window Help
-----
 * Olive Garden, MINDEN of a clump of garlic paste
 * Salt and pepper
-----
Ready
DAQ - SE

Loading .....
en linea

INSTRUCCION >> settime
COMANDO: settime
tiempo concentracion (seg) >> 60
tiempo medida (seg) >> 120
tiempo limpieza (seg) >> 60
TIEMPO ACTUALIZADO

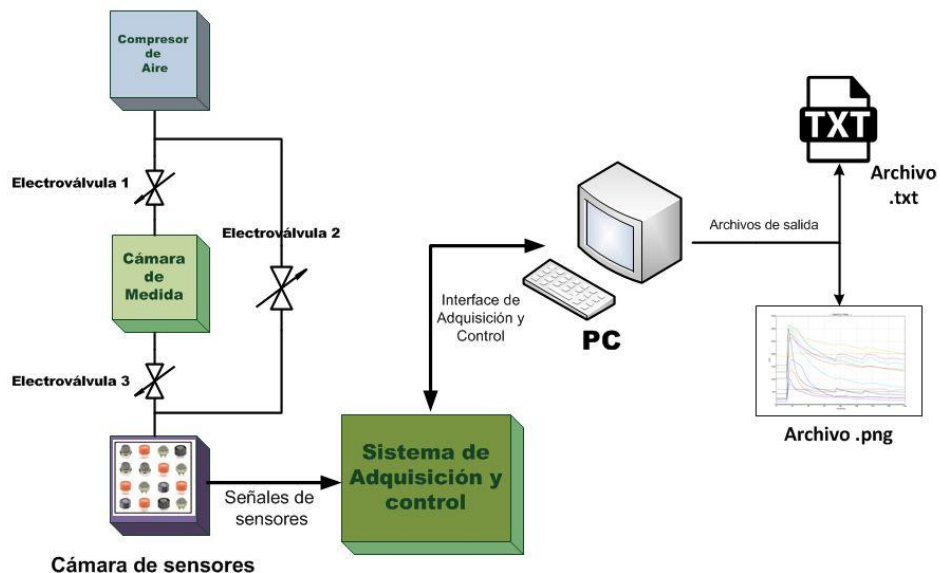
en linea

INSTRUCCION >> inicio
COMANDO: inicio
nombre de archivo >> med_4
-- EN ESPERA --
-- ETAPA DE CONCENTRACION --0
-- ETAPA DE CONCENTRACION --1
-- ETAPA DE CONCENTRACION --2
-- ETAPA DE CONCENTRACION --3
-- ETAPA DE CONCENTRACION --4
-- ETAPA DE CONCENTRACION --5
-- ETAPA DE CONCENTRACION --6
-- ETAPA DE CONCENTRACION --7
-- ETAPA DE CONCENTRACION --8
-- ETAPA DE CONCENTRACION --9
-- ETAPA DE CONCENTRACION --10
-- ETAPA DE CONCENTRACION --11
-- ETAPA DE CONCENTRACION --12
-- ETAPA DE CONCENTRACION --13
-- ETAPA DE CONCENTRACION --14
-- ETAPA DE CONCENTRACION --15
-- ETAPA DE CONCENTRACION --16

Ln: 4674 Col: 0
```

3.3.6. Parte 5: módulo de nariz electrónica. El grupo de investigación RadioGIS, en su interés por proponer un módulo base que permita implementar la metodología de nariz electrónica desarrollada en el presente trabajo, para fines de estudio e investigación y poder encaminarla hacia la solución y aplicación en diversas áreas de interés, desarrollo en unión con la empresa Soluciones Robóticas Integrales, un prototipo de sistema de medida y captura de olores, que unido con el presente trabajo de grado, supone un equipo de nariz electrónica base para el estudio e investigación de olores y reconocimiento de patrones. Con el fin de proporcionar una metodología para la construcción de futuros prototipos, se describirá de manera práctica los elementos que intervienen en este módulo de medida. La figura 41 visualiza el modelo general de nariz electrónica (etapa de medida + etapa de adquisición y control).

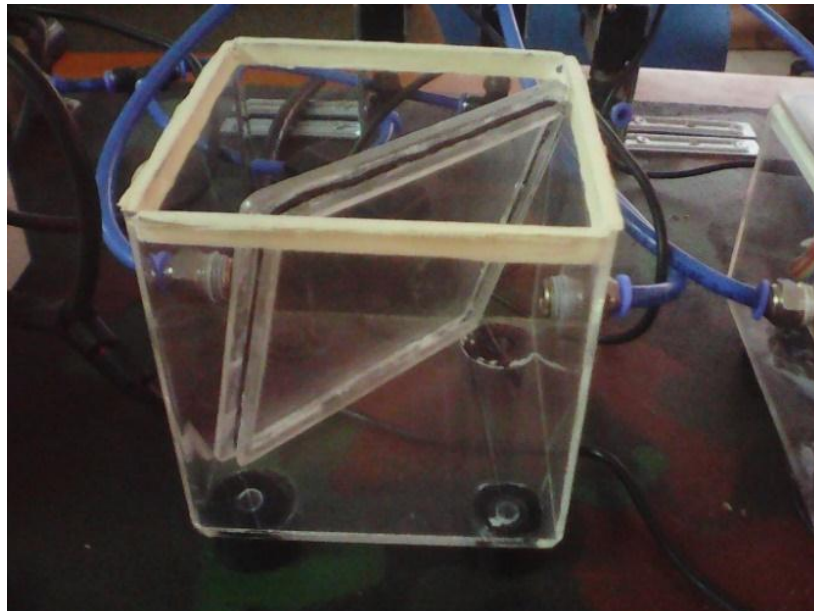
Figura 41. Modelo general de nariz electrónica



El módulo de medida corresponde a un conjunto de elementos como la cámara de concentración, la cámara de medida en donde se ubica la matriz de sensores de gases, las electroválvulas y el sistema de control para las mismas, y por ultimo un dispositivo de flujo de aire (bomba de aire, compresor, aire comprimido, entre otros). Este módulo basa su funcionamiento en generar un flujo de aire, que permita desplazar los componentes volátiles u olores de la cámara de concentración a la cámara de medida.

3.3.6.1. Cámara de concentración. La cámara de concentración consiste un recipiente que permite albergar diferentes objetos sólidos y líquidos. Consiste de una cámara sencilla en acrílico con tapa para cerrar y mantener la muestra aislada del exterior, en sus laterales dispone de una entrada y salida de aire que permite circular los volátiles hasta la cámara de medida según un tiempo determinado. Esta cámara durante el periodo de concentración se encuentra aislada mediante las electroválvulas y durante un tiempo establecido, en esta cámara se formara una concentración de volátiles que luego serán transportados por el flujo de aire hasta la cámara de medida en donde se encuentran los sensores. Sus dimensiones son 12cm ancho, 12cm largo, 12cm alto. La figura 42 muestra una foto de la cámara de concentración.

Figura 42. Vista general de la cámara de concentración



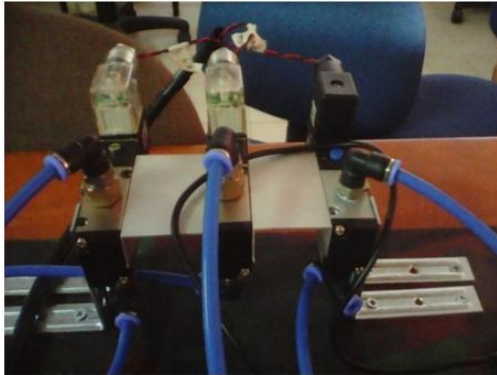
3.3.6.2. Cámara de medida. La cámara de medida es un recipiente de acrílico cuyas dimensiones son 13 cm ancho, 13cm largo, 10cm alto. De la misma manera que la cámara de concentración, esta dispone de una entrada y salida de aire, por el cual circula el flujo de aire constante y durante la fase de medida transporta los volátiles que provienen de la cámara de concentración. Esta cámara costa de un diseño sencillo y práctico, que puede ser adaptado según los sensores empleados y la aplicación final. La figura 43 muestra una foto de la cámara de medida y sus diferentes conexiones con el terminal de adquisición.

Figura 43. Vista general de la cámara de medida

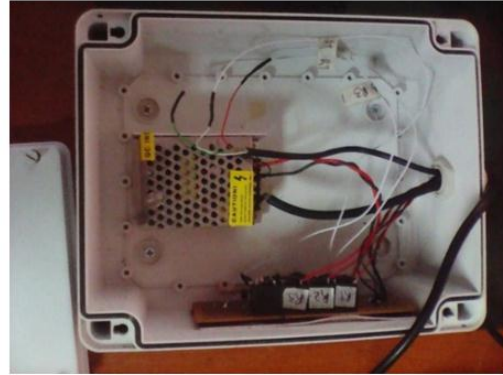


3.3.6.3. Electroválvulas. Las electroválvulas son de tipo solenoide modelo EBCHQ ref. 91283, y modelo MVSC-220-3E1-NC marca MINDMAN. Estas válvulas son del tipo on/off, y solo se encargan de controlar el paso del flujo de aire. La alimentación es de 110/220 AC con consumo de 3 a 5W. Estas electroválvulas cuentan con un circuito de apertura/cierre que se controla mediante la tarjeta de desarrollo Intel Galileo. En la figura 44 se puede observar la distribución de las electroválvulas en el dispositivo.

Figura 44. Vista de (a) las electroválvulas y (b) el módulo de adecuación para las electroválvulas



(a)



(b)

3.3.6.4. Regulador de presión. El modulo cuenta con un regulador de presión marca EBCHQ, modelo 85040 (figura 45). Este regulador permite ajustar la presión de entrada de aire entre 7.2 y 125 PSI, necesario para poder controlar la presión del flujo de aire en caso de utilizar un equipo compresor de aire, aire comprimido o cualquier bomba de aire y así poder mantener una presión constante en todo el sistema de flujo de aire.

Figura 45. Regulador de presión modelo 85040



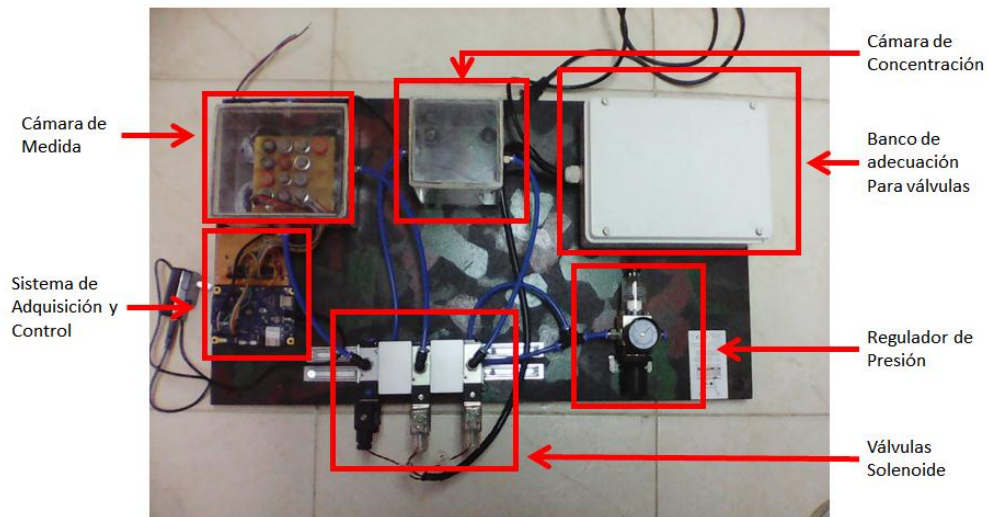
3.3.6.5. Fuente de alimentación para la matriz de sensores. Para la alimentación de los sensores, se usa una fuente de alimentación de triple salida modelo 1672 marca BK Precisión. Esta fuente puede suministrar de 0 a 32 V con suministro de máximo 3 A. esta fuente representa una solución práctica y rápida para alimentar la tarjeta que compone la matriz de sensores y suministrar la corriente necesaria para su correcto funcionamiento. En la figura 46 se observa la fuente de alimentación utilizada. La tarjeta de desarrollo es alimentada por un adaptador de 5V y 2.0A que ya viene incluida en la misma tarjeta.

Figura 46. Vista de la fuente de poder utilizada para la alimentación de la matriz de sensores



De manera general, el modulo final de nariz electrónica se puede visualizar en la figura 47. En él se puede ver la implementación del presente trabajo y el módulo de medida que fue desarrollado por el grupo de investigación RadioGIS y Soluciones Robóticas Integrales, para este proyecto en específico.

Figura 47. Vista del módulo final de nariz electrónica



3.4. ETAPA 4: EQUIPO FINAL

En la figura 48 muestra la implementación final del equipo de nariz electrónica, en el cual se puede distinguir todos los diferentes elementos que la conforman. Desarrollada como una plataforma para la investigación y la búsqueda de potenciales aplicaciones que pueda ofrecer este sistema de nariz electrónica en distintas áreas como la industria, el agro, la salud, entre otros. Así también el objetivo de este primer equipo es brindar las bases para el desarrollo de próximos prototipos enfocados hacia aplicaciones más concretas.

Figura 48. Vista general del prototipo de nariz electrónica durante un experimento



4. PRUEBAS Y RESULTADOS

Esta es la fase final del proyecto, en el cual se integra todas las etapas anteriores con el fin de validar mediante pruebas y medidas el correcto funcionamiento del equipo de nariz electrónica, evaluando el correcto funcionamiento de todos sus módulos.

4.1. PRUEBAS PRELIMINARES

Como una primera etapa se realizaron diferentes pruebas preliminares de la matriz de sensores y una versión del sistema de adquisición y control con algunos compuestos químicos comunes, como se puede observar en la figura 49; con el fin de verificar el correcto funcionamiento de la matriz de sensores, el sistema completo de adquisición y la lógica de control. En la figura 50 se observa la respuesta transitoria de los sensores de la nariz electrónica ante diferentes compuestos químicos como son el cloruro, butano (gas de encendedor) y poliuretano parte A. en cada grafica se puede observar la calidad de la adquisición, la respuesta en niveles de voltaje cuantificados (0 – 4095) que genera cada sensor y su comportamiento antes y después de estar expuesto a las diferentes muestras volátiles.

Figura 49. Prueba preliminar de la matriz de sensores y el sistema de adquisición

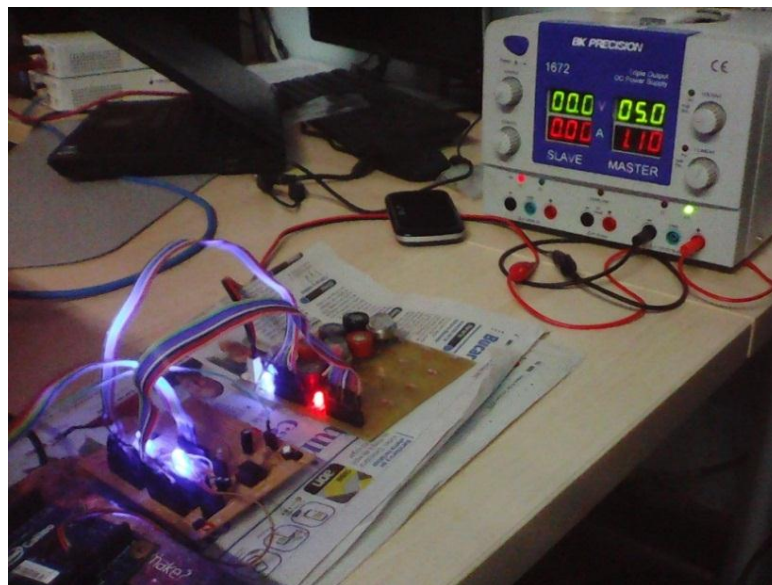
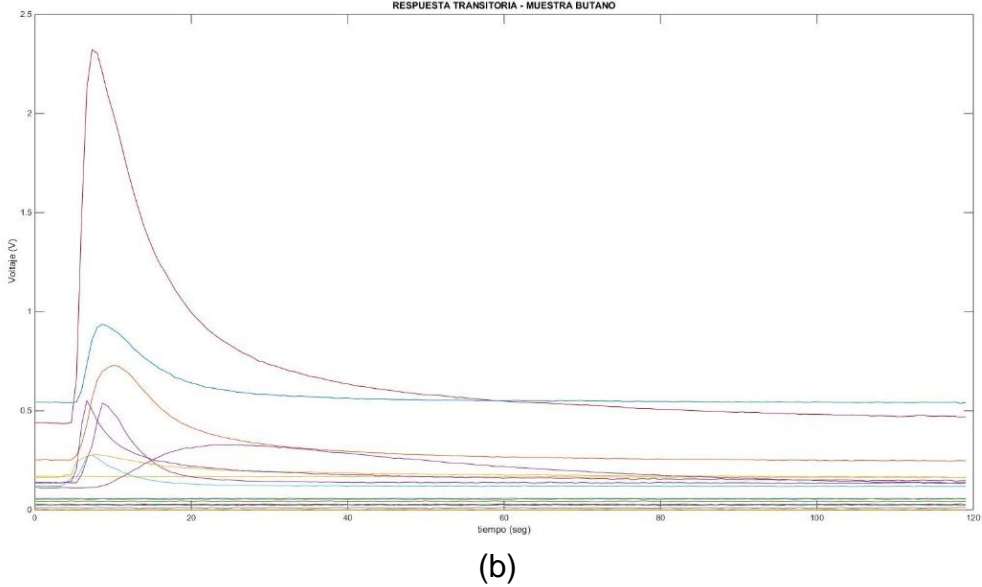
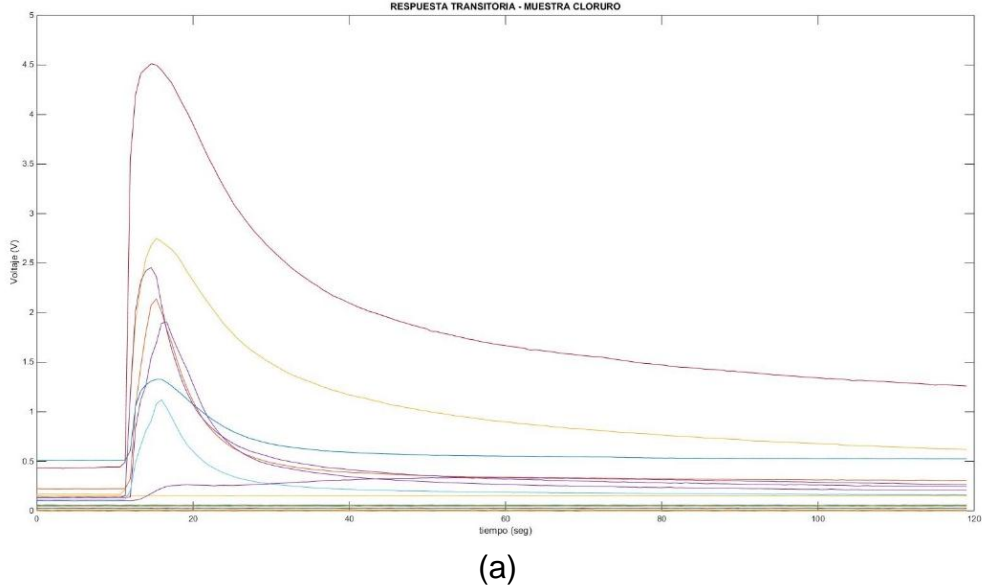
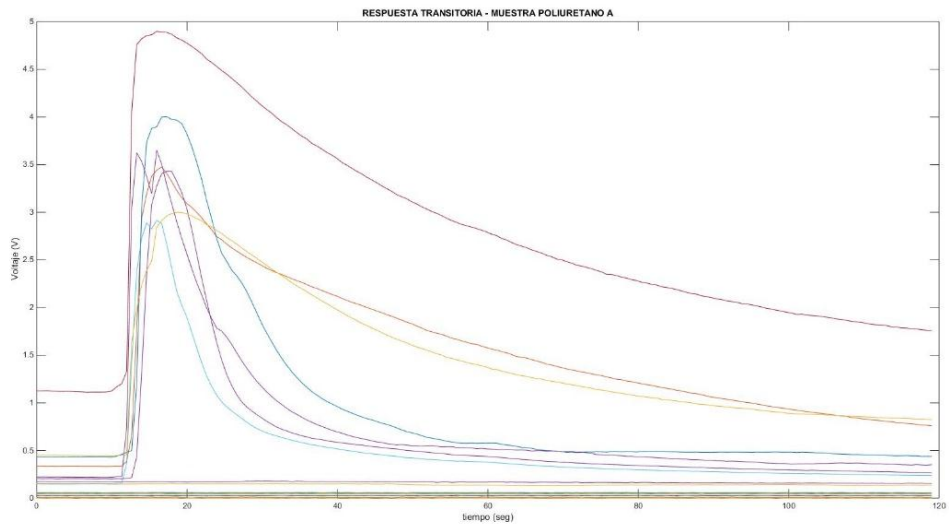


Figura 50. Respuesta transitoria de la matriz de sensores ante muestras de (a) cloruro, (b) butano y (c) espuma de poliuretano parte A





(c)

4.2. ETAPA DE EXPERIMENTACIÓN

Como una segunda etapa y ya integrando el presente trabajo en el módulo de medida, la figura 51 muestra una fotografía del compuesto comercial conocido como “silicona líquida” y el equipo durante un experimento para la medida de volátiles (figura 52). En la imagen se puede observar el equipo de nariz electrónica y el computador que se utilizó para visualizar la respuesta de los sensores en tiempo real, además de controlar todo el proceso del mismo.

Figura 51. Compuesto comercial conocido como “silicona líquida”

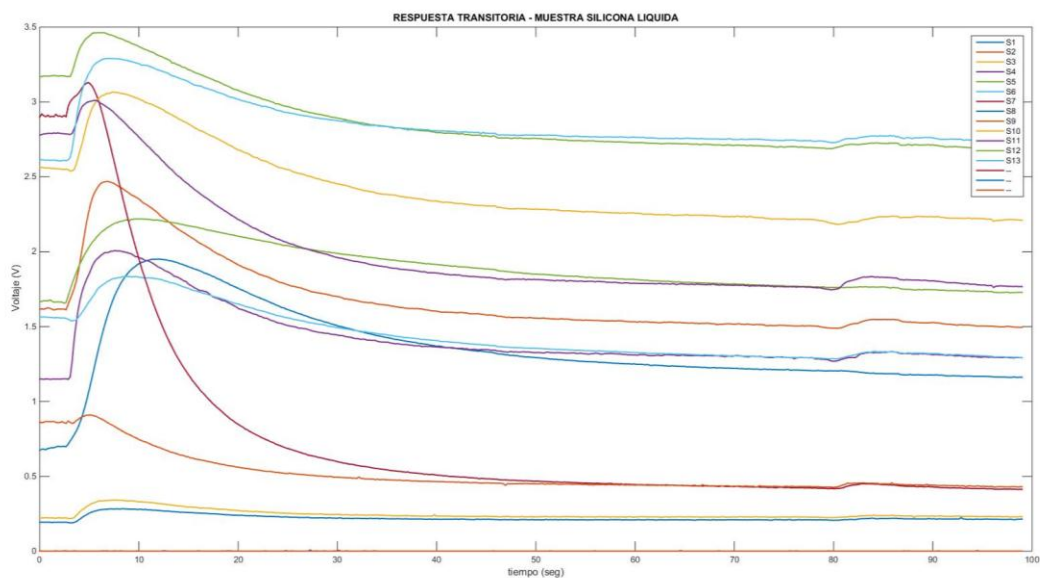


Figura 52. Vista de la nariz electrónica durante un experimento para la medida de volátiles de un compuesto comercial conocido como “silicona líquida”



Durante la medida se realizó el proceso de adquisición de una muestra de olor de “silicona líquida”, dando como resultado de la medida la reacción transitoria de la matriz de sensores ante la exposición de la muestra, como se puede apreciar en la gráfica de la figura 53.

Figura 53. Respuesta transitoria de una muestra de olor de “silicona líquida”

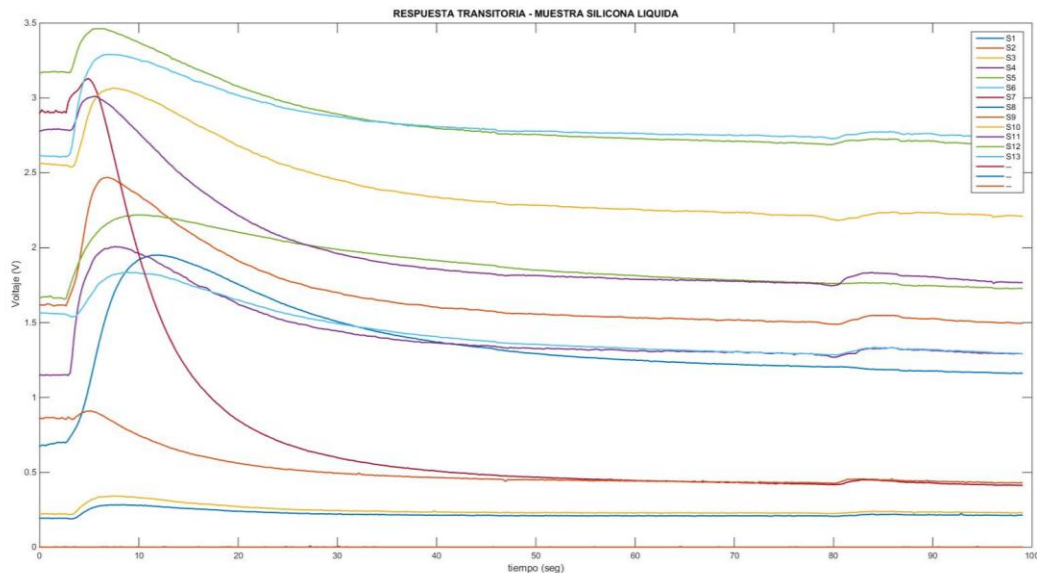


La matriz de sensores responde ante un compuesto volátil, generando una señal transitoria en cada sensor. Debido a que cada sensor está diseñado en su capa

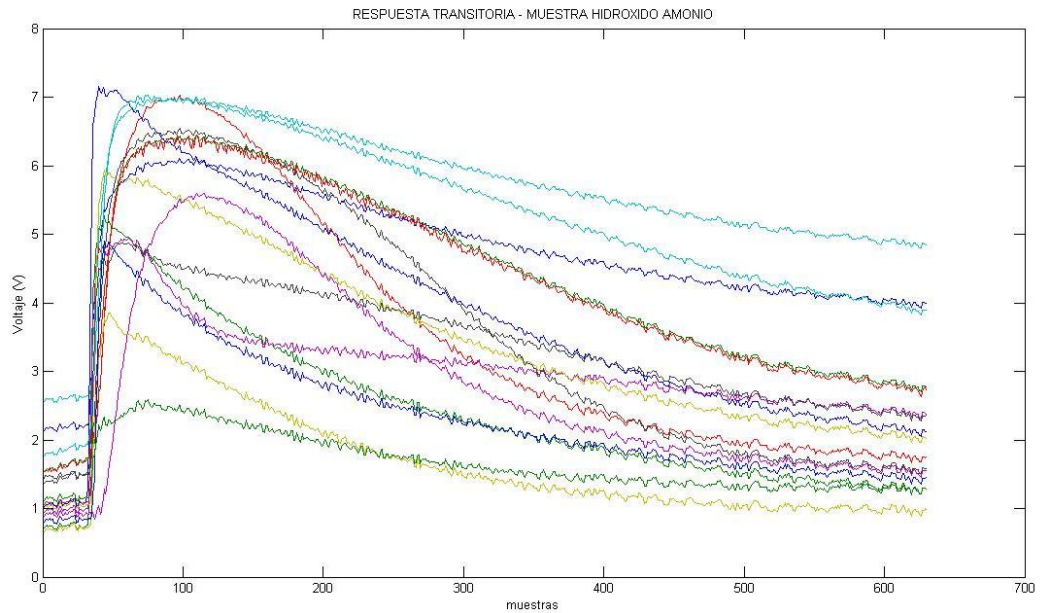
activa para responder ante una gama específica de compuestos volátiles, cada sensor responde de manera diferente ante el mismo volátil al que fue expuesto, generando así un conjunto de datos del volátil medido, que se conoce como “huella”. Esta huella es el patrón de olor percibido por la nariz electrónica y cuya información generada es la base para ingresarlo a un posterior algoritmo de programación en análisis de muestras de datos de olor y reconocimiento de patrones.

Al comparar la medida adquirida de la muestra de silicona con una medida realizada por la nariz electrónica de la Universidad de Pamplona en una muestra de benceno (figura 54), se puede observar un comportamiento transitorio similar generado por la matriz de sensores en los dos equipos en particular ante una muestra de un compuesto volátil.

Figura 54. Comparación entre las respuestas transitorias de la nariz electrónica de (a) RadioGIS y (b) la Universidad de Pamplona



(a)

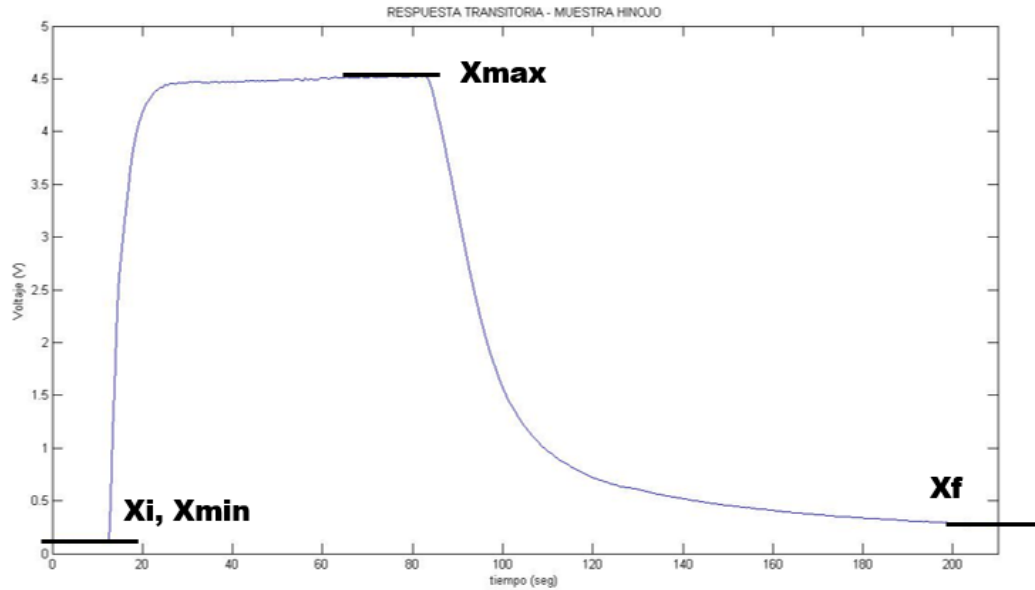


(b)

Fuente: Gualdron Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

Una de las características esenciales para analizar el comportamiento de los sensores, se basa en la respuesta eléctrica de cada sensor ante un compuesto volátil. En la figura 55 se puede observar un ejemplo de la respuesta transitoria del sensor TGS822 ante la presencia de una muestra de esencia de hinojo, en el cual los valores máximos, mínimos, iniciales y finales son muy utilizados para describir el comportamiento del sensor ante el volátil y extraer los parámetros de interés. Sin embargo el presente trabajo no pretende dar profundidad a la metodología de análisis de datos o manejo de parámetros.

Figura 55. Respuesta transitoria del sensor TGS822, resaltando los parámetros de interés



4.3. MANEJO DE ARCHIVOS DE MEDICIÓN

Al realizar la respectiva medición, la programación de adquisición y captura está diseñada para guardar en un archivo de extensión .txt en la tarjeta microSD la respectiva medición realizada y también un archivo de extensión .txt con el historial de mediciones realizadas. En la figura 56 se observa los archivos que se encuentran en el interior de la tarjeta de desarrollo, gracias al programa de acceso WINSCP. En el PC utilizado para la medición se genera también dos archivos del mismo nombre y con extensión .txt y .png resultante de la medida realizada (figura 57).

Figura 56. Archivos generados en el interior de la tarjeta microSD

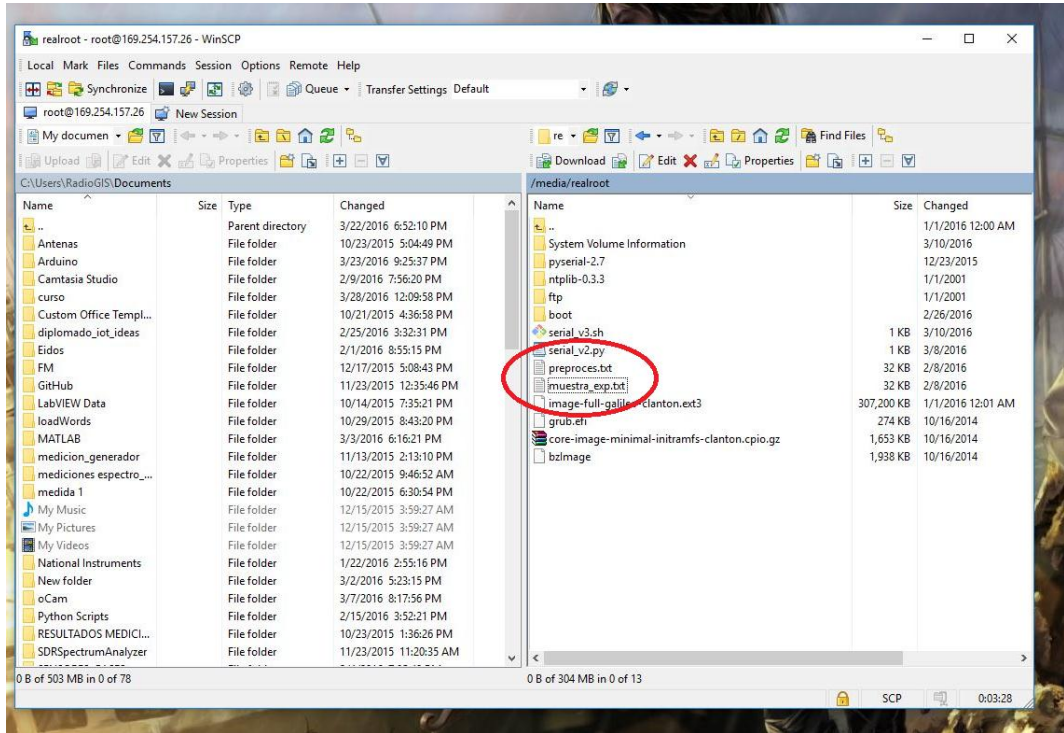
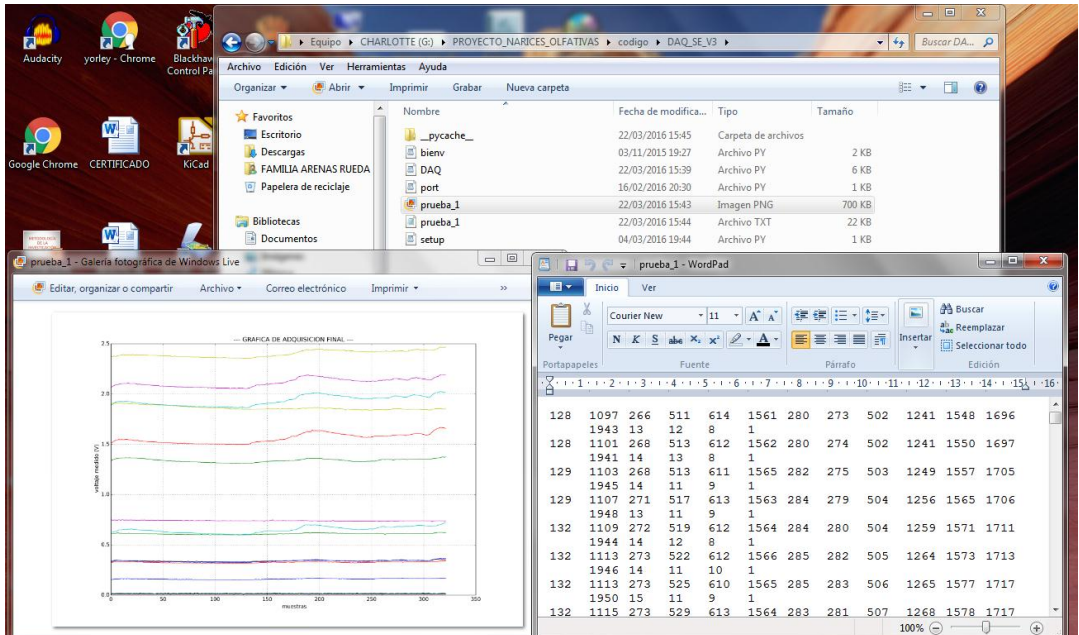


Figura 57. Muestra de los archivos generados por la interface



4.4. ENVÍO DE DATOS

Como una estrategia para implementar en una posible solución particular, donde se necesite enviar un archivo de medición con extensión .txt hacia un servidor ftp o un servicio web, para posteriormente utilizar esta medición para una posible aplicación web o manejo de datos en el cómputo en la nube, se implementó a partir de la solución terminal móvil estudiada en el presente trabajo, una solución básica para transmitir una medición hacia un servidor ftp de manera práctica y de fácil adaptación, utilizando el lenguaje de programación PYTHON y aprovechando la versatilidad de la tarjeta de desarrollo Intel Galileo para conectarse a internet.

Para su implementación, se puede utilizar un router con conexión a internet mediante Ethernet, ya que la tarjeta de desarrollo está totalmente configurada y lista para usarse mediante Ethernet. Para el presente trabajo se utilizó un modem suministrado por el grupo de investigación RadioGIS, marca D-Link modelo DWR-921 (figura 58). Este modem está diseñado a partir de las características de cualquier router para compartir internet de la manera tradicional vía Ethernet, la característica más a resaltar está en la capacidad de poder conectarse a la red 3G y 4G LTE de manera fácil y rápida.

Figura 58. Modem marca D-Link modelo DWR-921

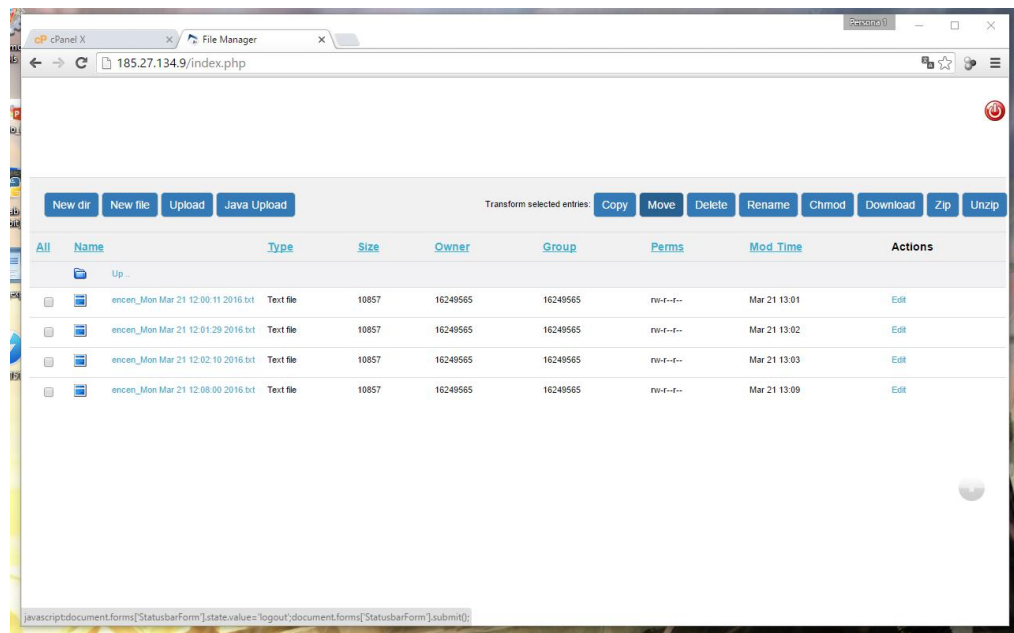


Fuente: hoja de datos modem D-link. Disponible en Mar-30-2016 en: <http://www.dlink.com/es/es/home-solutions/connect/broadband-modems-and-routers/dwr-921-4g-lte-router>

El presente trabajo quiere resaltar la posibilidad de utilizar cualquier modem para que la tarjeta Intel Galileo pueda conectarse y transmitir información y datos que sean de gran interés para el desarrollador, sin embargo se concentrara esta implementación en proponer una posible solución para transmitir información, a partir de las diferentes observaciones y propuestas que el autor da para poder enviar un archivo de datos.

Esta solución se basa en un programa con extensión .py que toma un archivo de medida con extensión .txt guardado en la microSD de la tarjeta de desarrollo y lo transmite hacia un servidor ftp, y agrega a este la fecha y hora de ingreso al servidor, aprovechando toda la capacidad de la tarjeta Intel Galileo para conectarse de manera fácil y sencilla hacia el internet. En la figura 59 se puede observar un ejemplo de un conjunto de archivos de medición transmitidos hacia un servidor de prueba gratuito conocido como “260mb” [24].

Figura 59. Archivos de medidas transmitidos hacia un servidor de pruebas (260mb)



5. CONCLUSIONES

- Se estudió un equipo de nariz electrónica de laboratorio desarrollada por la Universidad de Pamplona, encaminada hacia la identificación y descripción de los elementos que conforman, entre otras características que fueron base para entender su funcionamiento, y de allí plantear una solución base para la construcción de un modelo de nariz electrónica.
- Se adaptó el desarrollo de un terminal móvil al equipo de nariz electrónica, enfocado hacia una solución que permitiera transmitir datos y archivos relacionados a las mediciones generadas por el equipo de nariz electrónica de manera fácil, rápida y sin pérdida de datos o información. Cabe resaltar que la eficacia de este terminal móvil radica en su conectividad con las redes móviles y la cobertura que esta pueda brindar.
- Se implementó una solución que permitiera realizar la adquisición de un número determinado de sensores de gases marca Fígaro y Hanwei, cuya información resultante de todo el proceso de medición representa la respuesta transitoria de cada sensor ante un compuesto químico volátil expuesto, para posteriormente procesar o adecuar los datos para finalmente poder guardarlos en un dispositivo extraíble, transmitirlos de manera práctica a un PC o en el caso de un terminal móvil, poder transmitirlos hacia un servidor web.
- Se implementó 2 metodologías para poder realizar la adquisición mediante un PC y guardar la información para su posterior análisis. La primera metodología utilizó la herramienta de cómputo de Matlab; mediante su interface desarrollada el sistema de adquisición permitía observar el comportamiento de los sensores en tiempo real durante una medida, sin permitir la modificación de parámetros como los tiempos de concentración, medición y limpieza. La segunda metodología utilizó el lenguaje de programación PYTHON, esta opción permitió realizar de manera más práctica la programación, adaptación, todas las operaciones de control de tiempos y manejo de la adquisición de tiempo real, y ofrece al desarrollador una opción más sencilla, ágil y práctica de extender las características del sistema de adquisición hacia aplicaciones generales o adaptarse rápidamente a soluciones específicas.
- La realización de este trabajo como proyecto de grado, pretende dar un modelo de metodología fácil y práctica para el desarrollo de diversos proyectos basado en temas afines al concepto de nariz electrónica. Además sea de material didáctico que ayude a proyectar el conocimiento e investigación relacionada con la temática de nariz electrónica a los próximos desarrolladores.

6. RECOMENDACIONES

- Estudiar e investigar de manera amplia el tema de sensores de gases presentes en el mercado, Encaminado hacia la caracterización del dispositivo y los diferentes compuestos volátiles a los cuales el sensor puede detectar y reaccionar, Con el fin de formar el espectro de olores que un sensor puede percibir.
- Al momento de realizar las diferentes mediciones del módulo completo de nariz electrónica, se debe tener en cuenta la apropiada instalación de Python versión 3.4 o posterior, con sus módulos y librerías necesarias para su correcto funcionamiento. Actualmente es posible obtener paquetes completos de instalación de Python (Canopy, Anaconda, Ninja IDE, WingIDE, entre otros.) que están orientados hacia un fácil entorno de programación y proporciona todas las herramientas para un desarrollador.
- Regularmente se debe estar verificando la calidad de los sensores de gases, debido a que estos con su uso y el tiempo, su capa activa se va deteriorando, provocando una respuesta particular entre cada medición o genera una respuesta inesperada o abrupta ante un volátil. Además en caso de que el sensor se encuentre sucio debido al contacto accidental con un fluido, se recomienda retirar el sensor de la matriz, e inyectarle aire a alta presión o con un equipo de aspiradora, con el fin de remover la impureza del sensor. si un sensor presenta una respuesta errática, aleatoria o que no corresponde con una reacción ante un volátil, se recomienda apartarlo de la matriz de sensores, y buscar su respectivo reemplazo.
- El módulo de medida, realizado por el grupo de investigación RadioGIS en unión con Soluciones Robóticas Integrales, fue realizado con el principal objetivo de formar junto con el presente trabajo de grado, un equipo de nariz electrónica para laboratorio, enfocado hacia el estudio de componentes volátiles, el desarrollo de algoritmos de análisis y reconocimiento de patrones, y propone una base para la experimentación y desarrollo de diferentes dispositivos para aplicaciones específicas.

CITAS BIBLIOGRAFICAS

- [1] Aroma y olor. Disponible a mar-30-2016 en:
http://mazinger.sisib.uchile.cl/repositorio/lb/ciencias_quimicas_y_farmaceuticas/witt_inge01/capitulo01/03.html
- [2] Olores. Definición Y Medición Del Olor. Desodorización Por Vía Seca Y Húmeda. Biofiltros. Técnicas Combinadas. Disponible a Mar-30-2016 en:
http://www.cma.gva.es/comunes_asp/documentos/agenda/val/03.%20OLORES_JESUS%20CANDENAS.pdf
- [3] La Química Del Olor. Disponible a mar-30-2016 en:
<http://www.uv.mx/cienciauv/blog/quimicadelolor/>
- [4] Sancho Valls Josep, Bota Prieto Enric, de Castro Martín Juan José. Introducción al análisis sensorial de los alimentos (1999).
- [5] El Fascinante Mundo De Los Olores. Disponible a mar-30-2016 en:
<https://www.uv.mx/cienciahumano/revistae/vol23num1/articulos/olores/index.html>
- [6] Anatomía y Fisiología I. Disponible a Mar-30-2016 en:
http://www.elmodernoprometeo.es/Sitio_web/Anatomia_files/sentidos.pdf
- [7] Los Sentidos Gusto Y Olfato. Disponible en Mar-30-2016 en:
<http://enfermedadesyremedioscaseros.com/wp-content/uploads/el-olfato.jpg>
- [8] Los Compuestos Orgánicos Volátiles (COV) Peligro Respiratorio. Disponible a mar-30-2016 en:
<http://archivosseguridadlaboral-manueldomene.blogspot.com.co/2013/05/compuesto-organico-volatil-cov-peligro.html>
- [9] Compuestos Orgánicos Volátiles (Covs O Vocs). Disponible a mar-30-2016 en:
<http://cienciaybiologia.com/compuestos-organicos-volatiles-covs-o-vocs/>

[10] Gardner Julin, Bartlet Philip N. Sensors and Sensory Systems for an Electronic Nose (1991).

[11] Gualdron Guerrero Oscar Eduardo (2006). Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral, Universidad Rovira I Virgili.

[12] DURAN ACEVEDO Cristian Manuel, GUALDRON GUERRERO Oscar Eduardo, HERNANDEZ ORDOÑEZ Mariela. Nariz electrónica para determinar el índice de madurez del tomate de árbol. Ingeniería Investigación y Tecnología. 2014, nro. 3. p. 351-362. ISSN 1405-7743 FI-UNAM.

[13] TURNER Anthony P.F., MAGAN Naresh. Electronic noses and disease diagnostics. Nature Reviews Microbiology 2. 2004, February. p. 161-166. ISSN: 1740-1526.

[14] Rodríguez Daniel F (2012). Estudio y desarrollo de películas semiconductoras de óxidos metálicos nanoestructurados para el uso en microsensores de gas. Tesis Doctoral, Universidad Nacional de San Martin.

[15] Díaz Murcio Gerardo (2012). Desarrollo de una metodología de nariz electrónica para el control de calidad en alimentos basada en el espectrómetro de masas y análisis multivariado. Tesis de Maestría, Instituto Politécnico Nacional.

[16] Aguirre Andrés, Fernández Pablo, Grossy Carlos (2007). Interfaz USB genérica para comunicación con dispositivos electrónicos. Trabajo de grado para optar al título de ingeniero Electrónico, Universidad de la República Oriental del Uruguay.

[17] Conde John, Rangel Johan (2015). Sistema de comunicación de un terminal móvil para narices electrónicas basadas en IoT. Trabajo de grado para optar al título de ingeniero Electrónico, Universidad industrial de Santander.

[18] FIGARO GAS SENSORS, hojas de datos sensores Figaro. Disponible a Mar-30-2016 en: <http://www.figarosensor.com/>

[19] FIGARO GAS SENSORS, Figaro Gas Sensors Product Catalogue. Disponible a Mar-30-2016 en: https://jenslabs.files.wordpress.com/2013/06/figaro_tgs_serien.pdf

[20] TEXAS INSTRUMENTS, Hoja de datos TLC2543. Disponible a Mar-30-2016 en: <http://www.ti.com/lit/ds/symlink/tlc2543-ep.pdf>

[21] TEXAS INSTRUMENTS, Hoja de datos PCF8574. Disponible a Mar-30-2016 en: <http://www.ti.com/lit/ds/symlink/pcf8574.pdf>

[22] Características de Intel Galileo. Disponible a Mar-30-2016 en: <http://diymakers.es/intel-galileo-guia-inicial/>

[23] hoja de datos modem D-link. Disponible en Mar-30-2016 en: <http://www.dlink.com/es/es/home-solutions/connect/broadband-modems-and-routers/dwr-921-4g-lte-router>

[24] 260MB, Servicios de Hosting. Disponible a Mar-30-2016 en: <http://260mb.net/>

BIBLIOGRAFÍA

AGUIRRE Andrés, FERNÁNDEZ Pablo, GROSSY Carlos. Interfaz USB genérica para comunicación con dispositivos electrónicos. . Trabajo de grado para optar al título de ingeniero Electrónico. Uruguay: Universidad de la República Oriental del Uruguay. Instituto de Computación - Facultad de Ingeniería, 2007.

BREZMES LLECHA Jesús. Diseño de una nariz electrónica para la determinación no destructiva del grado de maduración de la fruta. Tesis Doctoral. Barcelona (España): Universitat politècnica de Catalunya. Departament De Teoria Del Senyal I Comunicacions, 2001.

CONDE John, RANGEL Johan. Sistema de comunicación de un terminal móvil para narices electrónicas basadas en IoT. Trabajo de grado para optar al título de ingeniero Electrónico. Bucaramanga (Colombia): Universidad industrial de Santander. Escuela De Ingeniería Eléctrica, Electrónica Y Telecomunicaciones, 2015.

DÍAZ MURCIO Gerardo. Desarrollo de una metodología de nariz electrónica para el control de calidad en alimentos basada en el espectrómetro de masas y análisis multivariado. Tesis de Maestría. México: Instituto Politécnico Nacional. Centro de investigación en ciencia aplicada y tecnología avanzada, 2012.

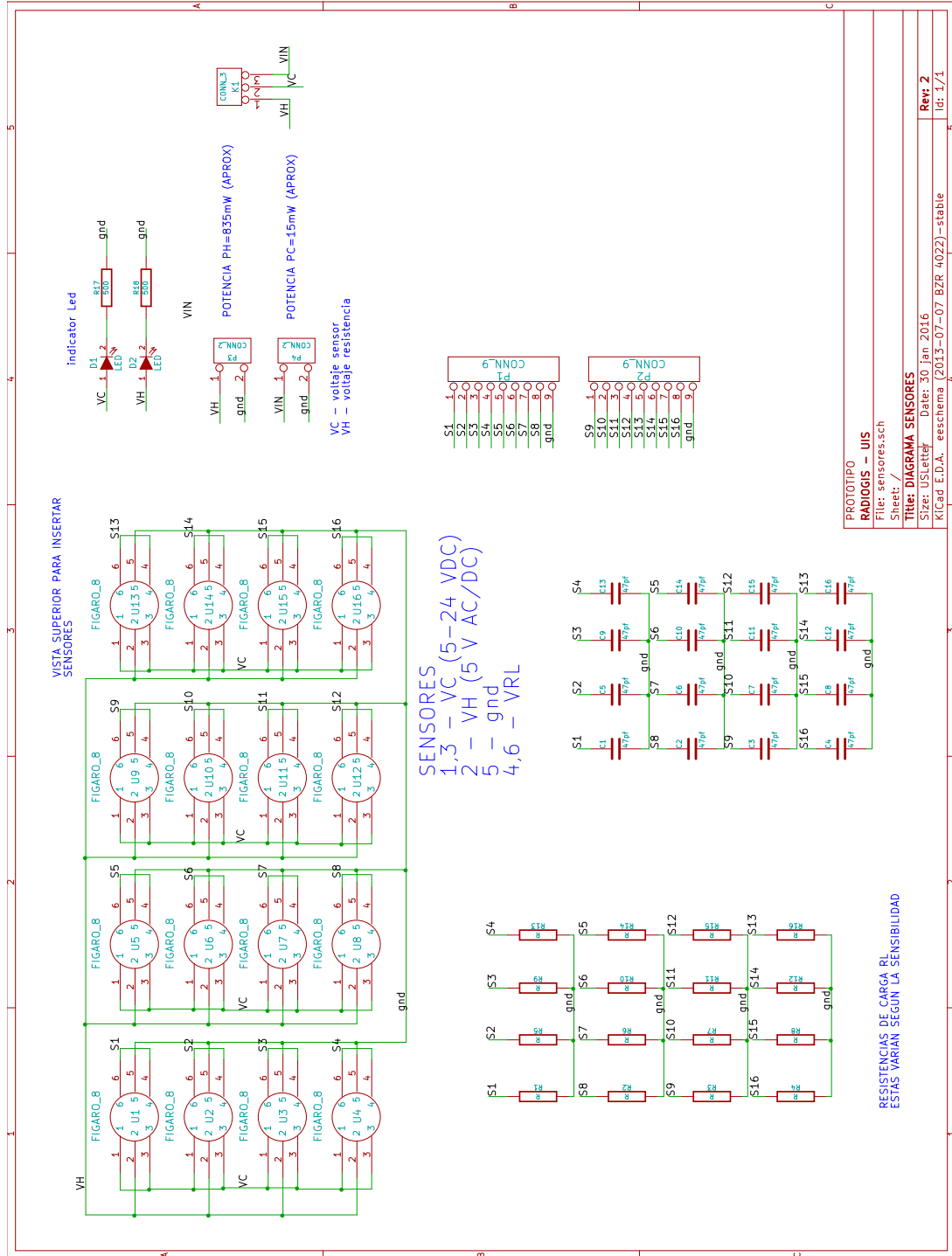
FLORES VEGA Christian Humberto, LI KU Antonio Eugenio. Diseño de una nariz electrónica como discriminador de olores utilizando Algoritmos Genéticos y Redes Neuronales Artificiales. Trabajo de grado para optar al título de Ingeniero Electrónico. Lima (Perú): Universidad Ricardo Palma. Escuela académico profesional de ingeniería electrónica, 2011.

GUALDRON GUERRERO Oscar Eduardo. Desarrollo de diferentes métodos de selección de variables para sistemas multisensoriales. Tesis Doctoral. Tarragona (España): Universidad Rovira I Virgili. Departamento de ingeniería eléctrica, electrónica y automatización, 2006.

PRIETO CASTAÑEDA Natalia. Implementación de un sistema electrónico de evaluación sensorial electrónico para el control de calidad de vinos. Tesis Doctoral. Valladolid (España): Universidad de Valladolid. Departamento De Física De La Materia Condensada Mineralogía Y Cristalografía, 2012.

RODRÍGUEZ Daniel F. Estudio y desarrollo de películas semiconductoras de óxidos metálicos nanoestructurados para el uso en microsensores de gas. Tesis Doctoral. Argentina: Universidad Nacional de San Martín. Instituto de tecnología Jorge A. Sábato, 2012.

Figura 61. Circuito matriz de sensores (Esquemático)



Nota: todas las imágenes se encuentran escaladas

Figura 62. Layout Circuito matriz de sensores (Top-rojo/Bottom-verde)

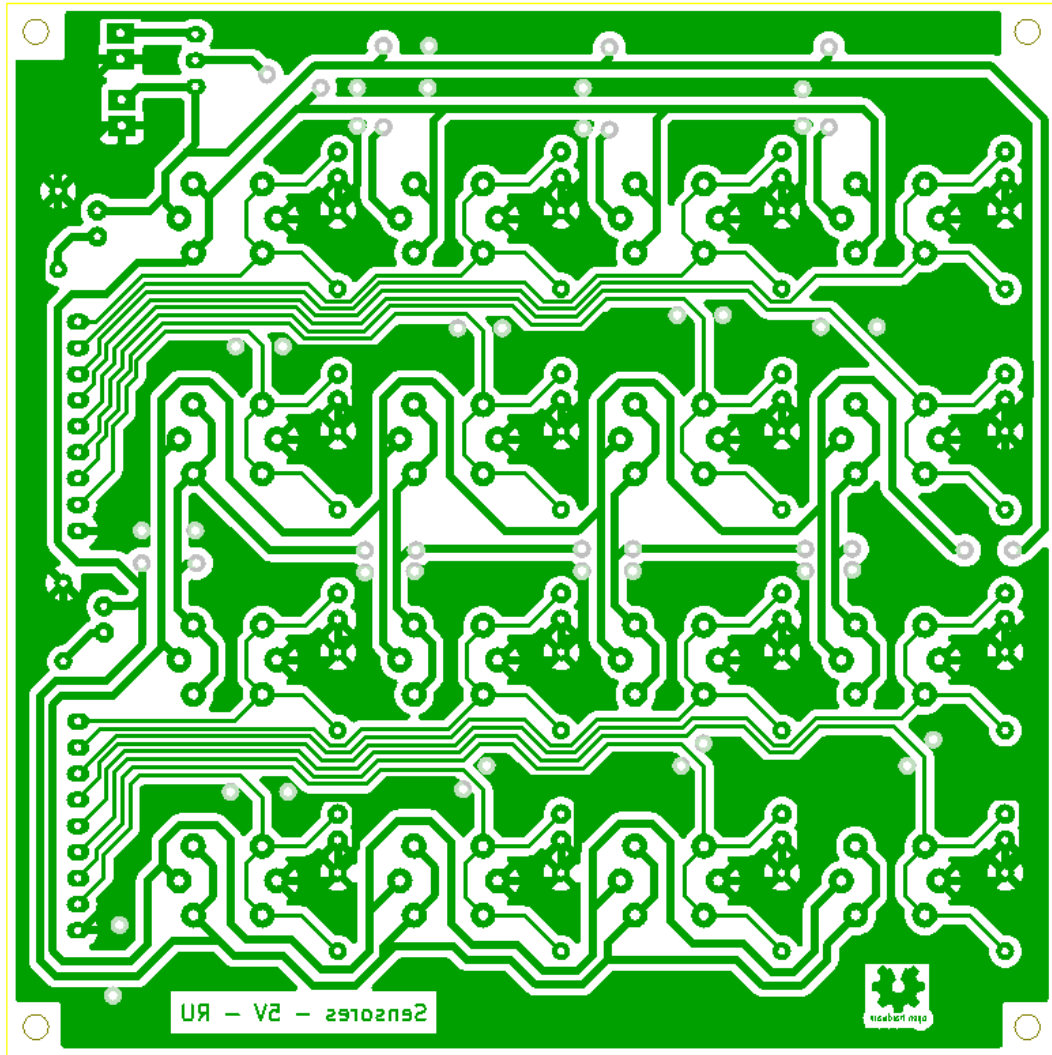
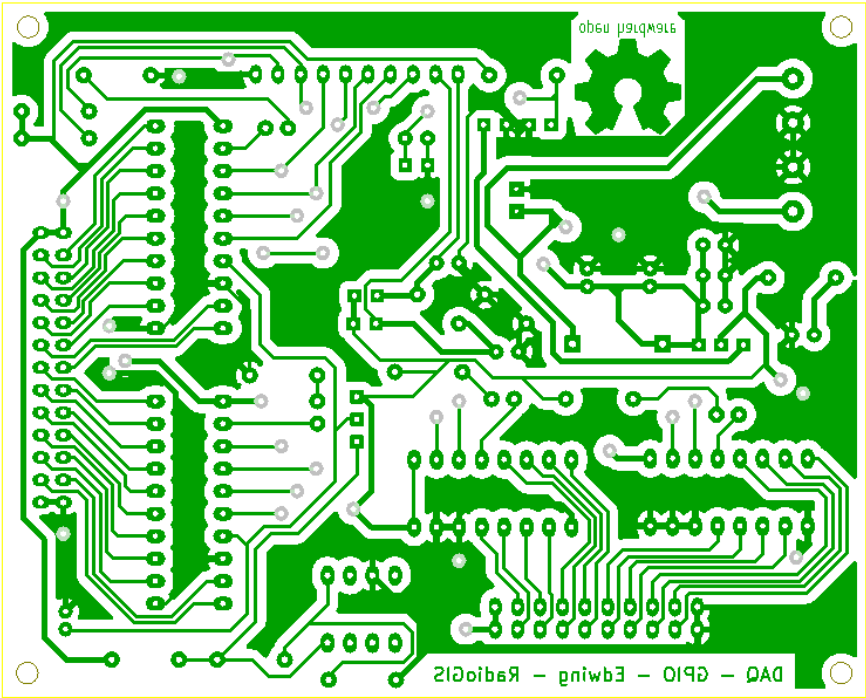
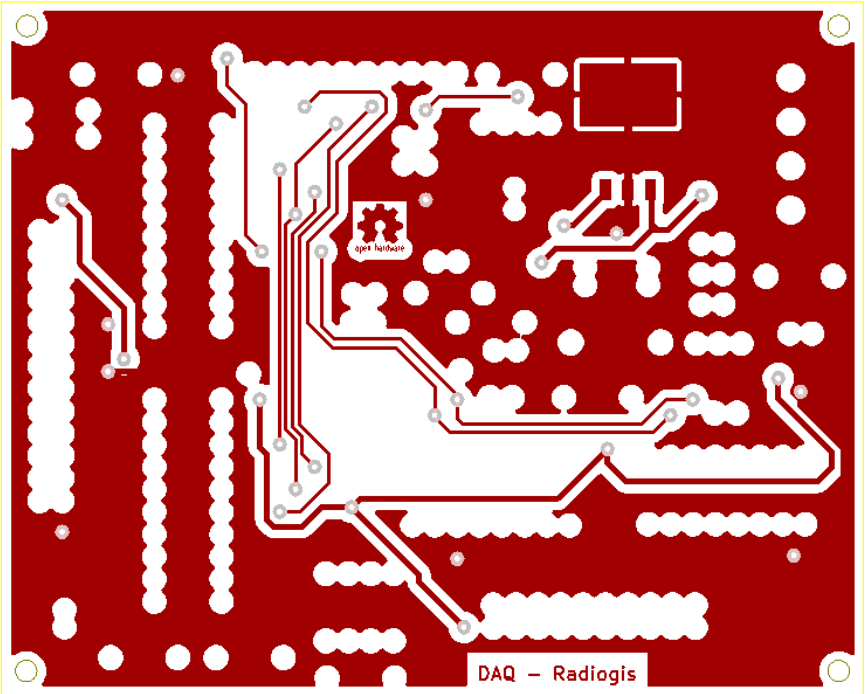


Figura 63. Layout Circuito sistema de adquisición y control (Top-rojo/Bottom-verde)



Anexo B

CÓDIGO FUENTE INTEL GALILEO

```
//-----//
//--- SISTEMA DE ADQUISICION SENSORES - VERSION 2.0 -----//
//-----//

//pines de uso mod 1
// 1 2 3 4 5 6 7 8
//pines de uso mod 2
// 1 2 3 4 5 6 7 8

//-- LIBRERIAS --//
#include <SD.h>
#include <SPI.h>

//-- MODULO DE ADQUISICION --//
String car="";
String getA="";
String getB="";
String getC="";

int cs = 9; // modulo1
int cs1 = 8; // modulo2
int valvula1=4;
int valvula2=5;
int valvula3=6;
int j = 0;
int cnt=0;
int muestras =0;
int muestras_concentracion =0;
int muestras_limpieza =0;
int del = 200;

#####//
#### tiempos de sistema (seg) ####//
int tconcentracion=0;
int tadq = 0;
int tlimpieza= 0;
#####//

//-- VARIABLES DE TOMA DE DATOS --//
int dat[16] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
int minimo[16] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
int maximo[16] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
int prep[16] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
int anterior[16] = {0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0};
int selo=1;
char aux[256];

// VARIABLE DE ARCHIVO //
File archivo;
```

```

File archivo2;

//----- ENTIDAD -----//
void setup()
{
  // PINES I/O
  pinMode(7, INPUT);
  pinMode(cs, OUTPUT);
  pinMode(cs1, OUTPUT);
  pinMode(valvula1,OUTPUT);
  pinMode(valvula2,OUTPUT);
  pinMode(valvula3,OUTPUT);
  digitalWrite(cs, 1);
  digitalWrite(cs1, 1);

  // CONFIGURACION SPI
  SPI.setClockDivider(SPI_CLOCK_DIV16);
  SPI.setBitOrder(MSBFIRST);
  SPI.setDataMode(SPI_MODE3);
  SPI.begin();

  // PUERTO SERIAL
  Serial.begin(57600);

  // VERIFICACION TARJETA SD PARA INICIO
  if (!SD.begin())
  {
    Serial.println(" fallo en la conexion con SD - Revisar ");
    return;
  }

  // MUESTRAS - TIEMPO - valor inicial
  tconcentracion=60;
  tadq = 120;
  tlimpieza=60;
  muestras = (tadq * 200)/74;
  muestras_concentracion= (tconcentracion * 162)/33;
  muestras_limpieza= (tlimpieza * 162)/33;

  //#####//
  // codigo control aqui //
  digitalWrite(valvula1,0);
  digitalWrite(valvula2,1);
  digitalWrite(valvula3,0);
  //#####//

}

//----- ARQUITECTURA -----//
void loop()
{
  inicio:

```

```

#####//
// codigo IOT aqui
while (Serial.available())
{
  car = Serial.readStringUntil('\n');
}

// modificacion tiempos de muestreo
if(car.startsWith("AT"))
{
  getA=car.substring(car.indexOf("AT")+2,car.indexOf("BT"));
  getB=car.substring(car.indexOf("BT")+2,car.indexOf("CT"));
  getC=car.substring(car.indexOf("CT")+2,car.length());

  tconcentracion = getA.toInt();
  tadq = getB.toInt();
  tlimpieza = getC.toInt();

  Serial.println(" TIEMPO ACTUALIZADO ");
  car="";
}
else
{
}
#####//

if (digitalRead(7) == 1 || car=="inicio")
{
  Serial.println("empieza");
  car="";
  goto sistema;
}
else
{
  // ENVIO DE DATOS EN ESTADO INICIAL //
  // selo=6666;
  // sprintf(aux, "%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d",
  // dat[0], dat[1], dat[2], dat[3], dat[4], dat[5], dat[6], dat[7], dat[8],
  // dat[9], dat[10], dat[11], dat[12], dat[13], dat[14], dat[15], selo);
  // Serial.println(aux);
  delay(del);
  goto inicio;
}
#####//

sistema:
// FUNCION DE ADQUISICION Y CAPTURA //

// AJUSTES DE TIEMPOS
ajuste_tiempos();

// CONCENTRACION
concentracion();

```

```

// MEDIDA
medida();

// LIMPIEZA
limpieza();

// ETAPA DE REPOSO
#####//
// codigo control aqui //
digitalWrite(valvula1,0);
digitalWrite(valvula2,1);
digitalWrite(valvula3,0);
#####//

selo=7777;
sprintf(aux, "%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d",
dat[0], dat[1], dat[2], dat[3], dat[4], dat[5], dat[6], dat[7], dat[8],
dat[9], dat[10], dat[11], dat[12], dat[13], dat[14], dat[15], selo);
Serial.println(aux);
}

//----- ajustes tiempos ----//
void ajuste_tiempos()
{
// MUESTRAS - TIEMPO - valor inicial
muestras = (tadq * 200)/74;
muestras_concentracion= (tconcentracion * 162)/33;
muestras_limpieza= (tlimpieza * 162)/33;
}

//----- CONCENTRACION -----//
void concentracion()
{
// ETAPA DE CONCENTRACION
#####//
// codigo control aqui //
digitalWrite(valvula1,0);
digitalWrite(valvula2,1);
digitalWrite(valvula3,0);
#####//

cnt=0;
do
{
selo=8888; // 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
sprintf(aux, "%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d",
dat[0], dat[1], dat[2], dat[3], dat[4], dat[5], dat[6], dat[7], dat[8],
dat[9], dat[10], dat[11], dat[12], dat[13], dat[14], dat[15], selo);
}

```

```

Serial.println(aux);
delay(del);
cnt++;
}while(cnt < muestras_concentracion);

cnt=0;
}

```

```
//---- SISTEMA MEDIDA -----//
```

```
void medida()
{
```

```
    // INICIO ETAPA DE MEDIDA //
```

```
do
```

```
{
    #####//
    // codigo control aqui //
    if(j<15)
    {
        digitalWrite(valvula1,0);
        digitalWrite(valvula2,1);
        digitalWrite(valvula3,0);
    }
    else
    {
        digitalWrite(valvula1,1);
        digitalWrite(valvula2,0);
        digitalWrite(valvula3,1);
    }
    #####//
}

```

```
//----- DAQ -----//
```

```
// MODULO 1
```

```
dat[0] = read_adc(1, 1); // 1
dat[1] = read_adc(2, 1); // 2
dat[2] = read_adc(3, 1); // 3
dat[3] = read_adc(4, 1); // 4
dat[4] = read_adc(5, 1); // 5
dat[5] = read_adc(6, 1); // 6
dat[6] = read_adc(7, 1); // 7
dat[7] = read_adc(8, 1); // 8
```

```
// MODULO 2
```

```
dat[8] = read_adc(1, 2); // 9
dat[9] = read_adc(2, 2); // 10
dat[10] = read_adc(3, 2); // 11
dat[11] = read_adc(4, 2); // 12
dat[12] = read_adc(5, 2); // 13
dat[13] = read_adc(6, 2); // 14
dat[14] = read_adc(7, 2); // 15
dat[15] = read_adc(8, 2); // 16
```

```
//-----//
```

```

//-- ENVIO DE DATOS --//
selo=1;
sprintf(aux, "%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d",
dat[0], dat[1], dat[2], dat[3], dat[4], dat[5], dat[6], dat[7], dat[8],
dat[9], dat[10], dat[11], dat[12], dat[13], dat[14], dat[15], selo);
Serial.println(aux);

//-- GUARDAR --//
sprintf(aux, "%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d",
dat[0], dat[1], dat[2], dat[3], dat[4], dat[5], dat[6], dat[7], dat[8],
dat[9], dat[10], dat[11], dat[12], dat[13], dat[14], dat[15]);
archivo = SD.open("REPORTE.txt", FILE_WRITE);
archivo.println(aux);
archivo.close();

// VALOR MAXIMO //
for (int i = 0; i < 16; i++)
{
  if (dat[i] >= maximo[i])
  {
    maximo[i] = dat[i];
  }
}

// VALOR MINIMO //
if (j >= 2)
{
  for (int i = 0; i < 16; i++)
  {
    if(dat[i] <= anterior[i])
    {
      minimo[i] = dat[i];
    }
  }
}
// VALOR ANTERIOR //
for (int i = 0; i < 16; i++)
{
  anterior[i] = dat[i];
}
//-----//

j++;
delay(del);

} while (j < muestras);

// ETIQUETA DE FINALIZADO //
archivo = SD.open("REPORTE.txt", FILE_WRITE);
archivo.println("//-----//");
archivo.println("//-----//");
archivo.close();

```

```

// PREPROCESADO TIPO: ( MAX - MIN ) //
for (int i = 0; i < 16; i++)
{
    prep[i] = maximo[i] - minimo[i];
}

// INICIO ARCHIVO PREPROCESADO //
sprintf(aux, "%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d",
prep[0], prep[1], prep[2], prep[3], prep[4], prep[5], prep[6], prep[7],
prep[8], prep[9], prep[10], prep[11], prep[12], prep[13], prep[14], prep[15]);
archivo2 = SD.open("PREPROCES.txt", FILE_WRITE);
archivo2.println(aux);
archivo2.println("//-----//");
archivo2.println("//-----//");
archivo2.close();

//---- LIMPIEZA DE VARIABLES ----//
j = 0;
for (int i = 0; i < 16; i++)
{
    prep[i] = 0;
    minimo[i] = 0;
    maximo[i] = 0;
    dat[i] = 0;
    anterior[i] = 0;
}
}

//----- LIMPIEZA -----//
void limpieza()
{
    // ETAPA DE LIMPIEZA
    //#####//
    // codigo control aqui //
    digitalWrite(valvula1,0);
    digitalWrite(valvula2,1);
    digitalWrite(valvula3,0);
    //#####//

    cnt=0;
    do
    {
        selo=9999; // 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
        sprintf(aux, "%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d",
        dat[0], dat[1], dat[2], dat[3], dat[4], dat[5], dat[6], dat[7], dat[8],
        dat[9], dat[10], dat[11], dat[12], dat[13], dat[14], dat[15], selo);
        Serial.println(aux);
        delay(del);
        cnt++;
    }while(cnt < muestras_limpieza);
    cnt=0;
}

```

```

##### NO TOCAR AQUI #####
int read_adc(int x, int pa)
{
    int digitalvalor = 0;
    int paq = 0;
    byte primeraconfig = 0b00000000;
    byte primerbyte = 0b00000000;
    byte segundobyte = 0b00000000;
    primeraconfig |= (x << 4);
    primeraconfig |= (3 << 2);
    if (pa == 1)
    {
        paq = cs;
    }
    else
    {
        paq = cs1;
    }
    noInterrupts();
    digitalWrite(paq, 0);
    SPI.transfer(primerconfig);
    SPI.transfer(0x00);
    digitalWrite(paq, 1);
    delayMicroseconds(20);
    digitalWrite(paq, 0);
    primerbyte = SPI.transfer(0x00);
    segundobyte = SPI.transfer(0x00);
    digitalWrite(paq, 1);

    digitalvalor |= (primerbyte << 8);
    digitalvalor |= (segundobyte);

    return ((digitalvalor >> 4)+1);
}

```

Anexo C

CÓDIGO FUENTE MATLAB ADQUISICIÓN

```

%%-----%%
%%----- SISTEMA DE ADQUISICION SENSORES -----%%
%%-----%%
%% PRUEBA SATISFACTORIA
clc;
clear all;
close all;

```

```

%% VARIABLES INICIALES - datos de adquisicion - 16 SENSORES
PU='COM14'; %-- puerto COM
nombre_ar='muestra_1415'; %-- archivo de salida
extension='.txt';
nombre_archivo=strcat(nombre_ar,extension);
i=1;
dato1=[];
dato2=[];
dato3=[];
dato4=[];
dato5=[];
dato6=[];
dato7=[];
dato8=[];
dato9=[];
dato10=[];
dato11=[];
dato12=[];
dato13=[];
dato14=[];
dato15=[];
dato16=[];

%/-- inicio de puerto COM matlab --//
delete(instrfind({'Port'},{PU}));
disp('---- inicio adquisicion ---- ');

puerto=serial(PU);
set(puerto, 'BaudRate', 57600);
set(puerto, 'FlowControl', 'hardware');
set(puerto, 'DataBits', 8);
set(puerto, 'StopBits', 1);
set(puerto, 'Terminator', 'LF');
set(puerto, 'TimeOut', 1);
set(puerto, 'InputBufferSize', 256);
warning('on', 'MATLAB:serial:fscanf:unsuccessfulRead');
fopen(puerto);
puerto.ReadAsyncMode = 'continuous';
readasync(puerto);

flushinput(puerto);
valor=fscanf(puerto, '%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d')
;
valor=fscanf(puerto, '%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d')
;
%//-----//
%//---- ADQUISICION EN ESPERA ----//
tconcent=0;

while 1==1
    %
    % 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6

valor=fscanf(puerto, '%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d')
;

```



```

dato11(i)=valor(11)+1;
dato12(i)=valor(12)+1;
dato13(i)=valor(13)+1;
dato14(i)=valor(14)+1;
dato15(i)=valor(15)+1;
dato16(i)=valor(16)+1;

x=linspace(1,i,i);
set(l1,'XData',x,'YData',dato1(1:i));
set(l2,'XData',x,'YData',dato2(1:i));
set(l3,'XData',x,'YData',dato3(1:i));
set(l4,'XData',x,'YData',dato4(1:i));
set(l5,'XData',x,'YData',dato5(1:i));
set(l6,'XData',x,'YData',dato6(1:i));
set(l7,'XData',x,'YData',dato7(1:i));
set(l8,'XData',x,'YData',dato8(1:i));
set(l9,'XData',x,'YData',dato9(1:i));
set(l10,'XData',x,'YData',dato10(1:i));
set(l11,'XData',x,'YData',dato11(1:i));
set(l12,'XData',x,'YData',dato12(1:i));
set(l13,'XData',x,'YData',dato13(1:i));
set(l14,'XData',x,'YData',dato14(1:i));
set(l15,'XData',x,'YData',dato15(1:i));
set(l16,'XData',x,'YData',dato16(1:i));

totali='-- MUESTRA NUMERO : ';
ava=num2str(i);
seli=strcat(totali,ava);
disp(seli);
i=i+1;
elseif (valor(17)==7777)
    break;
elseif (valor(17)==9999)
    tlimpieza=tlimpieza+1;
    mens4='--- ETAPA LIMPIEZA --- ';
    tim2=num2str(tlimpieza);
    mens4=strcat(mens4,tim2);
    disp(mens4);
end

end
tlimpieza=0;
i=0;
%//-----//
% // guardar los datos en un archivo .dat //
disp('---- ADQUISICION FINALIZADA ----');

table=[dato1;dato2;dato3;dato4;dato5;dato6;dato7;dato8;dato9;dato10;...
        dato11;dato12;dato13;dato14;dato15;dato16];
tabla=table';

%% NOTA: POR AHORA SOLO 16 SEÑALES, REVISAR
%% metodo guardado 1

```

```

%% csvwrite('prub1.txt',tabla);
%% metodo guardado 2
dlmwrite(nombre_archivo,tabla,'delimiter','\t')
disp('-- guardado finalizado correctamente --');

%%--- grafica final de adquisicion ---
gua_fig=figure(2);
plot(tabla);
set(gca,'fontsize',14)
title('-- GRAFICA ADQUISICION FINAL --')
xlabel(' NUMERO DE MUESTRAS ')
ylabel(' valor ( 0 - 4096 ) ')
axis([1 length(dato1) 0 4100])
legend('s1','s2','s3','s4','s5','s6','s7','s8','s9','s10','s11',...
       's12','s13','s14','s15','s16')

extension='.jpeg';
nombre_archivo=strcat(nombre_ar,extension);
saveas(gua_fig,nombre_archivo)
%%/----- cerrar puerto y limpieza---//
stopasync(puerto);
fclose(puerto);
delete(puerto);
% figure
% plot(res_1)
delete(instrfind({'Port'},{PU}));
clear
clear puerto

%%-----

```

Anexo D

CÓDIGO FUENTE MATLAB INTERFACE

```

function varargout = DAQ(varargin)
% DAQ MATLAB code for DAQ.fig
%   DAQ, by itself, creates a new DAQ or raises the existing
%   singleton*.
%
%   H = DAQ returns the handle to a new DAQ or the handle to
%   the existing singleton*.
%
%   DAQ('CALLBACK',hObject,eventData,handles,...) calls the local
%   function named CALLBACK in DAQ.M with the given input arguments.
%
%   DAQ('Property','Value',...) creates a new DAQ or raises the

```

```

%     existing singleton*. Starting from the left, property value pairs
are
%     applied to the GUI before DAQ_OpeningFcn gets called. An
%     unrecognized property name or invalid value makes property
application
%     stop. All inputs are passed to DAQ_OpeningFcn via varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only
one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

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

% Last Modified by GUIDE v2.5 18-Feb-2016 17:39:49

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @DAQ_OpeningFcn, ...
                  'gui_OutputFcn',  @DAQ_OutputFcn, ...
                  'gui_LayoutFcn',   [] , ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before DAQ is made visible.
function DAQ_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to DAQ (see VARARGIN)

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

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes DAQ wait for user response (see UIRESUME)
% uiwait(handles.figure1);

```

```

% --- Outputs from this function are returned to the command line.
function varargout = DAQ_OutputFcn(hObject, eventdata, handles)
% varargout cell array for returning output args (see VARARGOUT);
% hObject handle to figure
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

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

function puerto_Callback(hObject, eventdata, handles)
% hObject handle to puerto (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of puerto as text
% str2double(get(hObject,'String')) returns contents of puerto as
a double

% --- Executes during object creation, after setting all properties.
function puerto_CreateFcn(hObject, eventdata, handles)
% hObject handle to puerto (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
% See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in inicio.
function inicio_Callback(hObject, eventdata, handles)
% hObject handle to inicio (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
%-----%
%----- ARQUITECTURA DE PROGRAMA -----%
PU=get(handles.puerto,'String');
nombre_ar=get(handles.archiv,'String');
extension='.txt';
nombre_archivo=strcat(nombre_ar,extension);
axes(handles.axes1);
i=1;

```

```

dato1=[];
dato2=[];
dato3=[];
dato4=[];
dato5=[];
dato6=[];
dato7=[];
dato8=[];
dato9=[];
dato10=[];
dato11=[];
dato12=[];
dato13=[];
dato14=[];
dato15=[];
dato16=[];

%//-- inicio de puerto COM matlab --//
delete(instrfind({'Port'},{PU}));

puerto=serial(PU);
set(puerto, 'BaudRate', 57600);
set(puerto, 'FlowControl', 'hardware');
set(puerto, 'DataBits', 8);
set(puerto, 'StopBits', 1);
set(puerto, 'Terminator', 'LF');
set(puerto, 'TimeOut', 1);
set(puerto, 'InputBufferSize', 256);
warning('on', 'MATLAB:serial:fscanf:unsuccessfulRead');
fopen(puerto);
puerto.ReadAsyncMode = 'continuous';
readasync(puerto);

flushinput(puerto);
valor=fscanf(puerto, '%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d')
;
valor=fscanf(puerto, '%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d')
;

%//-----//
%//---- ADQUISICION EN ESPERA ----//
tconcent=0;

while 1==1
    %                1  2  3  4  5  6  7  8  9  0  1  2  3  4  5  6

valor=fscanf(puerto, '%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d,%d')
;
drawnow;
if (valor(17)==6666)
    mens1='-- EN ESPERA --';
    set(handles.pantalla, 'string', mens1);
elseif (valor(17)==8888)

```



```

dato16(i)=valor(16)+1;

x=linspace(1,i,i);
set(l11,'XData',x,'YData',dato1(1:i));
set(l12,'XData',x,'YData',dato2(1:i));
set(l13,'XData',x,'YData',dato3(1:i));
set(l14,'XData',x,'YData',dato4(1:i));
set(l15,'XData',x,'YData',dato5(1:i));
set(l16,'XData',x,'YData',dato6(1:i));
set(l17,'XData',x,'YData',dato7(1:i));
set(l18,'XData',x,'YData',dato8(1:i));
set(l19,'XData',x,'YData',dato9(1:i));
set(l110,'XData',x,'YData',dato10(1:i));
set(l111,'XData',x,'YData',dato11(1:i));
set(l112,'XData',x,'YData',dato12(1:i));
set(l113,'XData',x,'YData',dato13(1:i));
set(l114,'XData',x,'YData',dato14(1:i));
set(l115,'XData',x,'YData',dato15(1:i));
set(l116,'XData',x,'YData',dato16(1:i));

totali='-- MUESTRA NUMERO : ';
ava=num2str(i);
seli=strcat(totali,ava);
set(handles.pantalla,'string',seli);
i=i+1;
elseif (valor(17)==7777)
    break;
elseif (valor(17)==9999)
    tlimpieza=tlimpieza+1;
    mens4='--- ETAPA LIMPIEZA --- ';
    tim2=num2str(tlimpieza);
    mens4=strcat(mens4,tim2);
    set(handles.pantalla2,'string',mens4);
end

end

tlimpieza=0;
i=0;
%//-----//
% // guardar los datos en un archivo .dat //

table=[dato1;dato2;dato3;dato4;dato5;dato6;dato7;dato8;dato9;dato10;...
    dato11;dato12;dato13;dato14;dato15;dato16];
tabla=table';

%% NOTA: POR AHORA SOLO 16 SEÑALES, REVISAR
%% metodo guardado 1
%% csvwrite('prub1.txt',tabla);
%% metodo guardado 2
dlmwrite(nombre_archivo,tabla,'delimiter','\t')
%disp('-- guardado finalizado correctamente--');
mens5='-- guardado finalizado correctamente --';
set(handles.pantalla,'string',mens5);

```

```

%%--- grafica final de adquisicion ---
gua_figu=figure();
plot(tabla);
set(gca,'fontsize',14);
title('-- GRAFICA ADQUISICION FINAL --');
xlabel(' NUMERO DE MUESTRAS ');
ylabel(' valor ( 0 - 4096 ) ');
axis([1 length(dato1) 0 4100]);
legend('s1','s2','s3','s4','s5','s6','s7','s8','s9','s10','s11',...
       's12','s13','s14','s15','s16')

extension='.jpeg';
nombre_archivo=strcat(nombre_ar,extension);
saveas(gua_figu,nombre_archivo);
%%/-- cerrar puerto y limpieza---//
stopasync(puerto);
fclose(puerto);
delete(puerto);
delete(instrfind({'Port'},{PU}));
guidata(hObject, handles);
%-----%
%-----%

function archiv_Callback(hObject, eventdata, handles)
% hObject     handle to archiv (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of archiv as text
%        str2double(get(hObject,'String')) returns contents of archiv as
a double

% --- Executes during object creation, after setting all properties.
function archiv_CreateFcn(hObject, eventdata, handles)
% hObject     handle to archiv (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns
called

% Hint: edit controls usually have a white background on Windows.
%        See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```

% --- Executes on button press in salir.
function salir_Callback(hObject, eventdata, handles)
% hObject    handle to salir (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% --- Executes on button press in refrescar.
function refrescar_Callback(hObject, eventdata, handles)
% hObject    handle to refrescar (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
set(handles.puerto, 'String', 'COM14');
set(handles.archiv, 'String', 'muestra_1');
set(handles.pantalla, 'String', '');
set(handles.pantalla1, 'String', '');
set(handles.pantalla2, 'String', '');
cla(handles.axes1);
grid off;

% --- Executes when user attempts to close figure1.
function figure1_CloseRequestFcn(hObject, eventdata, handles)
% hObject    handle to figure1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: delete(hObject) closes the figure
delete(hObject);

% --- Executes on mouse press over axes background.
function axes1_ButtonDownFcn(hObject, eventdata, handles)
% hObject    handle to axes1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

```



```

elif sys.platform.startswith('darwin'):
    ports = glob.glob('/dev/tty.*')
else:
    raise EnvironmentError('Unsupported platform')

result = []
for port in ports:
    try:
        s = serial.Serial(port)
        s.close()
        result.append(port)
    except (OSError, serial.SerialException):
        pass

return str(result[int(len(result)-1)])

```

- **DAQ**

```

# -*- coding: utf-8 -*-
#-----#
#---- DAQ - NE ----#
#-----#
# \r - espacio
# \n - nueva linea

## librerias ##
import sys
import os
import serial
import time
import bienv
import numpy
from matplotlib import pyplot
from port import *
import warnings

warnings.filterwarnings("ignore")

def conversion( numer ):
    response=float((numer*5)/4095)

    return round(response,3)

## CAPTURA ##

```

```

def captura(archivo):

    # espera
    ser.flush()
    line=str(ser.readline().decode())
    line=str(ser.readline().decode())

    ## etapa de espera ##
    print('-- EN ESPERA --')
    while True:

        line=str(ser.readline().decode())
        a=int(line.find('6666'))
        if (a >= 0):

            line=""

        elif (a < 0):

            break

    ## etapa de concentracion ##
    cnt1=0
    while True:

        line=str(ser.readline().decode())
        a=int(line.find('8888'))
        if (a >= 0):
            print('-- ETAPA DE CONCENTRACION --'+str(cnt1))
            cnt1=cnt1+1
            line=""

        elif (a < 0):

            break

    ## etapa de medida ##

    line=str(ser.readline().decode())
    lineas=line.replace("\n","").replace("\r","").split(',')
    global tam
    tam=len(lineas)

    for i in range(0,tam):
        exec('global p%d'%(i))
        exec('global m%d'%(i))
        exec('m%d=[]'%(i))
        exec('p%d=[0]*25'%(i))

    print("---- iniciando captura ----")
    pyplot.ion()
    pyplot.figure(figsize=(15,10))
    pyplot.title('--- GRAFICA DE ADQUISICION EN TIEMPO REAL ---')
    pyplot.xlabel(' muestras ')

```



```

pyplot.close()
pyplot.figure(figsize=(15,10))
pyplot.title('--- GRAFICA DE ADQUISICION FINAL ---')
pyplot.xlabel(' muestras ')
pyplot.ylabel(' voltaje medido (V) ')
pyplot.grid(True)
for i in range(0,tam):
    exec('r%d, =pyplot.plot(m%d,linewidth=1)%'(i,i))

archi2=archivo+'.png'
pyplot.savefig(archi2,dpi=500) # grafica de salida

cnt2=0
while True:

    line=str(ser.readline().decode())
    a=int(line.find('9999'))
    if (a >= 0):
        print('-- ETAPA DE LIMPIEZA --'+str(cnt2))
        cnt2=cnt2+1
        line=""

    elif (a < 0):
        break

print('-- CAPTURA FINALIZADA -- ARCHIVO DE SALIDA: '+archivo)
pyplot.ioff()
pyplot.close()

return

## SERIAL ##
def open_serial(puerto):
    #---- configuracion puerto ----#
    #ser.port = "/dev/ttyUSB0"

    global ser
    bau='57600'
    ser=serial.Serial()

    ser.port= puerto
    ser.baudrate= bau
    ser.bytesize=8 #bits per bytes
    ser.parity='N' #set parity: no parity
    ser.stopbits=1 #stop bits
    ser.timeout=5
    print(' Loading ..... ')
    time.sleep(1)

    # inicio puerto serial
    ser.open()
    ser.flush()

```

```

#-----#
return 0

def cerrar_serial():
    print(' cerrando serial ..... ')
    ser.flush()
    ser.close()
    print(' conexion serial finalizada ..... ')

#-----#
#----- ARQUITECTURA -----#
if __name__=='__main__':

    bienv.portada()
    print(' DAQ - SE \n')
    #-----#
    open_serial(scan_port())
    #-----#

    while True:

        print(' en linea \n')
        comando=""
        comando=str(input("INSTRUCCION >> "))
        print('COMANDO: '+comando)
        if comando=='inicio':

            comando=comando+'\n'
            nombre=str(input("nombre de archivo >> "))
            ser.write(comando.encode())
            captura(nombre)

        elif comando=='salir':
            break

        elif comando=='settime':

            comando=""
            tconcent=str(input("tiempo concentracion (seg) >> "))
            tmed=str(input("tiempo medida (seg) >> "))
            tlimp=str(input("tiempo limpieza (seg) >> "))
            comando='AT'+tconcent+'BT'+tmed+'CT'+tlimp+'\n'

            ser.write(comando.encode())
            respuest=str(ser.readline().decode())
            print(respuest)

        else:
            print('----- COMANDO NO RECONOCIDO -----')

    cerrar_serial()
    sys.exit(" ----- APLICACION FINALIZADA ----- ")

```

Anexo F

CÓDIGO PYTHON TRANSMISOR FTP

```
## transmisor archivos a ftp ##

# libreria ftp - version python2.7
from ftplib import FTP
import ntplib
from time import ctime

## CONEXION SERVIDOR ##

# hora servidor #
c = ntplib.NTPClient()
response = c.request('south-america.pool.ntp.org')
timzon=-5
dato=str(ctime(float(response.tx_time)+(3600*timzon)))

# direccion servidor
ftp=FTP('xxxxxxxxxxxxxx')
# user, passwd
ftp.login(user='xxxxxxxx',passwd='xxxxxxxx')

# navegar en las carpetas a guardar
ftp.dir()
ftp.cwd('/htdocs/pollo_sabroso/')

# envio de archivo con extension .txt y nombre predeterminado
filename='encen_1.txt'
ftp.storbinary('STOR ' + filename, open(filename,'rb'))
print 'transmision finalizada'

# renombrar archivo
archiv="encen_"+dato+".txt"
ftp.rename("encen_1.txt", archiv)

ftp.quit()
```