



**CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS
AVANZADOS DEL INSTITUTO POLITÉCNICO NACIONAL**

UNIDAD ZACATENCO

DEPARTAMENTO DE INGENIERÍA ELÉCTRICA

SECCIÓN DE BIOELECTRÓNICA

**"Desarrollo de un olfatómetro controlado por computadora para
investigación clínica"**

T E S I S

Que presenta

LUIS FERNANDO VALDEZ GARDUÑO

Para obtener el Grado de

MAESTRO EN CIENCIAS

EN LA ESPECIALIDAD DE

INGENIERÍA ELÉCTRICA

Director de la Tesis:

Dr. Juan Manuel Gutiérrez Salgado

México, D.F.

Diciembre 2014

"The most exciting phrase to hear in science, the one that heralds new discoveries, is not 'Eureka!' but 'That's funny...'"

Isaac Asimov

Agradecimientos

Gracias a mi familia por su apoyo incondicional, a mi madre por soportarme a pesar de todo. A mis hermanos Edgar, Alejandra y Miguel por molestarme, escucharme, darme consejos y ayudarme cuando fue necesario, a Dante por despertarme sin falta y acompañarme hasta que le fue permitido.

A mi novia Ana por su constante presencia y cariño, por su insistencia y consejo respecto a mis proyectos en particular por la frase 'piensa en el usuario' que procuró tuviera siempre presente.

Un especial agradecimiento a mi asesor el Dr. Juan Manuel Gutiérrez por su confianza y apoyo. Y a los sinodales el Dr. Roberto Muñoz y el Dr. Moisés León por tomarse el tiempo de revisar el trabajo.

Gracias a mis compañeros de clase, profesores y amigos que conocí en el transcurso de la maestría de los cuales trate de aprender toda clase de cosas e hicieron ameno mi paso por la maestría. En especial a todos los que integran el laboratorio 11 por sus consejos tanto académicos como personales y por haberme hecho participe de todas las convivencias y tradiciones que implican ser parte del mismo.

Gracias al Consejo Nacional de Ciencia y Tecnología(CONACYT) por haberme apoyado con la beca que hizo posible mis estudios de maestría.

Por último a todos los que no he mencionado pero que de una u otra forma me ayudaron a generar ideas para la resolución de los problemas que se fueron presentando, GRACIAS.

Resumen

En el caso del ser humano, es particularmente interesante identificar las afecciones en la percepción del olfato por alteraciones del tipo biológico como del entorno, incluyendo claro está a aquellos estados que se consideran patológicos como es el caso de algunas enfermedades neurodegenerativas. De igual forma resulta interesante, estudiar el efecto de los olores en el comportamiento del ser humano, debido a que su presencia pueden desencadenar respuestas hormonales o emocionales. En lo particular, se sabe que el olor se relaciona con la aceptación o rechazo de un alimento apetecible o potencialmente dañino.

El objetivo principal de la presente tesis es el diseño instrumental de un sistema que pueda ser utilizado en estudios relacionados con el olfato. Dicho sistema conocido como olfatómetro activo que permitirá estimular a pacientes con diferentes olores y tiempos de estimulación de forma automatizada haciendo uso de una computadora personal. Del mismo modo, se incluye la adquisición de registros suplementarios relacionados con los potenciales evocados olfativos (OEPs) mediante Electroencefalografía (EEG).

En primera instancia se tiene como resultado el prototipo totalmente funcional de un olfatómetro activo con arquitectura modular. Sus características principales son: Contar con cuatro vías, capacidad de manejar muestras sólidas o líquidas, tener un intervalo de trabajo para aire olorizado de 0.6 a 1.5 l/min y para aire limpio de 0.6 a 3 l/min, contar con interfaces gráficas para implementar estimulaciones individuales o cíclicas y permitir el registro de señales suplementarias. Además de no requerir instalaciones especiales al contar con su propia fuente de aire limpio y ser portátil (Dimensiones 30 x 25 x 30 cm y Peso 4 Kg).

El prototipo fue puesto a prueba utilizándose un sistema de registro de EEG (Emotiv®, EPOC) para la creación de una base de datos conteniendo las señales de EEG producto de la estimulación con 3 olores diferentes y un blanco, en 5 sujetos de prueba. Obteniéndose como producto adicional el software para la segmentación automática de los registros de EEG, siendo esto posible gracias al envío de marcadores por parte de sus interfaces gráficas.

Abstract

On humans, a particularly interesting topic is how olfaction is affected by biological or environmental changes, also including the pathological states that can be shown on some neurodegenerative diseases. Another interesting topic is how odors can affect human behavior, cause it's presence may unleash hormonal or emotional responses. It's particularly well known the odor play an important role on accepting or rejecting an appetizing or a potentially harmful food item.

The principal objective of this thesis is the instrumental design of a system that can be used on olfaction research. A system known as active olfactometer, it will automatically stimulate study subjects with different odors and stimulation times using a personal computer. As well it includes the acquisition of supplementary signals related to Olfactory Evoqued Potentials(OEPs) by Electroencephalography(EEG).

As the main product lies a fully functional active olfactometer with modular architecture. The main characteristics of the prototype are: Having four ways, handling of solid and liquid odor sources, operational range for odor airflow goes from 0.6 to 1.5 l/min and 0.6 to 3 l/min for clean airflow, having graphical interfaces for individual and cyclic stimulation and letting supplementary signal recording. It also has the advantages of no need of special facilities by having its own clean air source and being portable(Size 30 x 25 x 30 cm y Weight 4 Kg).

The prototype was tested while making a EEG database containing the signals of 3 different odor stimulation and one control signal of 5 study subjects using an EEG record tool(Emotiv®, EPOC). Additional software for the automatic segmentation of the EEG records was obtained, segmentation was possible thanks to the markers sent by the graphical interface software.

Contenido

Resumen	i
Abstract.....	ii
Contenido	iii
Capítulo 1	1
1.1 Introducción.....	1
1.2 Justificación	3
1.3 Objetivo principal	4
1.4 Objetivos particulares	4
1.5 Estructura de la tesis	5
Capítulo 2	7
2.1 Marco teórico	7
2.1.1 El sentido del olfato	7
2.1.2 Anatomía.....	8
2.1.3 El código olfatorio	12
2.1.4 Olores.....	13
2.1.5 Practica clínica.....	14
2.1.6 Olfatometría	15
2.2 Estado del arte.....	18
2.2.1 Olfatómetros	18
2.2.2 Olfatómetros pasivos	18
2.2.3 Olfatómetros activos	20
Capítulo 3	27
3.1 Propuesta de diseño	27
3.2 Diseño y construcción del olfatómetro	28
3.2.1 Fuente de Aire	30
3.2.2 Tratamiento de Aire	30

3.2.3	Selección de odorantes	31
3.2.4	Control de flujo	32
3.2.5	Sección de administración.....	34
3.3	Diseño electrónico y programación.....	34
3.4	Interfaz de Usuario	39
3.5	Adquisición de la señal de EEG	41
3.6	Protocolo de Prueba.....	41
Capítulo 4	43
4.1	Módulos de potencia.....	44
4.2	Sensores de presión	44
4.3	Tarjeta de interconexión	45
4.4	Diseño de carcasa	45
4.5	Caracterización del canal de aire limpio.....	46
4.7	Interfaces Gráficas	48
4.8	Sensores auxiliares(Sensor de respiración)	48
4.9	Registro de EEG y Segmentación	49
Capítulo 5	55
Capítulo 6	61
6.1	Conclusiones.....	61
6.2	Trabajo a futuro	61
Referencias:	63
Anexo A (Programación)	- 1 -
Anexo B (Diagramas eléctricos y PCB)	- 45 -
Anexo C (Diseño de carcasa)	- 51 -
Anexo D (Artículos de congreso)	- 61 -

Capítulo 1

Introducción

1.1 Introducción

Los procesos sensitivos son la manera en que un ser vivo puede obtener retroalimentación sobre su interacción con el ambiente; permitiendo determinar su reacción ante cada estímulo. Uno de estos procesos es el olfato que en conjunto con el gusto son conocidos como los sentidos químicos. El olfato en particular es el sentido encargado de detectar y procesar los compuestos químicos volátiles transportados por el aire y que son percibidos al inspirarlos [1]. El papel del olfato en relación con el gusto, radica en que los alimentos que ingresan en la boca también despiden moléculas que son arrastradas hasta el epitelio sensitivo olfatorio. Este hecho tiene como consecuencia que la sensación predominante al comer no provenga del gusto que se inicia en la lengua sino del olfato. Dicha percepción puede provocar una amplia gama de reacciones las cuales pueden ir desde aceptar o rechazar un alimento apetecible o potencialmente dañino, hasta desencadenar una respuesta hormonal o emocional específica [1]–[3].

Aunque los procesos *estímulo-reacción* han sido estudiados ampliamente en las últimas décadas, aún existen muchas incógnitas respecto de aquellos procesos relacionados con los sistemas olfatorios [4]–[6]. En el caso del ser humano, es particularmente interesante identificar como se ve afectada la percepción del olfato por distintos cambios tanto biológicos como en el entorno, incluyendo claro está, a los estados que se consideran patológicos como es el caso de algunas enfermedades neurodegenerativas [7]. Otras interrogantes por su parte, se encuentran relacionadas con las reacciones que producen distintos estímulos olfatorios y la influencia que tienen tanto en el estado físico como en el psicológico de un individuo [8]. De igual manera, el estudio del sistema olfatorio en otros seres vivos es de gran importancia, pues muchas

otras especies lo emplean específicamente para determinar su comportamiento y realizar ciertas actividades primarias (como la reproducción o alimentación). Un ejemplo típico de lo anterior es la relación insectos-agricultura, debido a que el comportamiento de la mayoría de los insectos considerados como plaga se ve influenciado por algún compuesto químico volátil producido por la planta presa [9], [10].

Finalmente, en el área de las llamadas tecnologías bioinspiradas, existe también un profundo interés por comprender las capacidades sensoriales para emular la capacidad humana de relacionar características del entorno con su correspondiente sentido conceptual. Uno de los primeros intentos para definir este tipo de sistemas fue establecido por Dood y Pearsud [11] con la introducción de la Nariz Electrónica (NE), un dispositivo capaz de imitar el sistema olfativo de los mamíferos para la detección de olores. Las narices electrónicas han tenido un gran desarrollo y hasta hoy su campo es uno de los que más publicaciones tiene respecto a tecnologías bioinspiradas. Su aplicación se encuentra en varios sectores por ejemplo: Evaluación de calidad y aroma de alimentos, seguridad alimentaria, perfumería, manejo de combustibles y explosivos, monitoreo ambiental, biomedicina y procesos industriales [5].

Para lograr entender a fondo el papel del olfato es necesario contar con el instrumental adecuado a fin de identificar las reacciones que producen los estímulos olfatorios. En general, la mayoría de experimentos olfativos en humanos exigen tiempos de estímulos rápidos y estables, así como un control preciso de la duración de la estimulación. El sistema capaz de realizar estas tareas se conoce como olfatómetro. "Un olfatómetro es el instrumento utilizado para medir la sensibilidad del sentido del olfato con respecto a intensidad, concentración y calidad de un olor" [12] que en conjunto con herramientas de diagnóstico, tales como la Electroencefalografía (EEG) y la Resonancia magnética funcional (RMf), hace posible estudiar las reacciones que se producen en el cerebro ante diferentes estímulos olfatorios haciendo posible la detección de desórdenes y/o alteraciones.

De forma convencional, los estudios de olfatometría no se encuentran restringidos a los descritos anteriormente, sin embargo, son un claro ejemplo de las tantas variantes que existen de aplicación. Esta situación, provoca que los olfatómetros existentes posean particularidades específicas para cada prueba. A pesar de ello, es posible distinguir dos tipos principales de olfatómetro: *los activos* que entregan los olores en una corriente de aire portadora y *los pasivos* que requieren de la inhalación del sujeto o de una corriente de aire externa para transportar el olor. Así, mientras los olfatómetros activos, son los más utilizados en estudios de laboratorio, pues permiten controlar la mayoría de las variables del entorno de prueba; los olfatómetros pasivos suelen emplearse para efectuar pruebas rápidas en consultorio. Una segunda clasificación, está relacionada con el tipo de muestras aromáticas que son capaces de manejar: líquidas, sólidas o gaseosas. Por último, existe la clasificación considerando el tipo sujetos de prueba (insectos, pequeños mamíferos, humanos, etc.) donde la principal diferencia son los dispositivos de administración.

En resumen el campo de aplicación de los olfatómetros va desde el estudio de las reacciones en los sistemas biológicos hasta la calibración o evaluación de los sistemas artificiales. La presente tesis se acota al diseño de un olfatómetro que cumpla con los requerimientos mínimos para estudios en humanos, sin embargo, procura mantener una arquitectura versátil que permita ciertas adaptaciones a otros tipos de estudios en etapas posteriores de desarrollo.

1.2 Justificación

Actualmente, la falta de información respecto a las patologías que involucran al olfato se debe a que en la práctica clínica rara vez se incluyen pruebas para valorar la función olfativa. El diagnóstico, por lo general es establecido cuando el propio sujeto ya ha notado una alteración significativa. Del mismo modo, la falta de difusión de las pruebas entre los médicos y el escaso desarrollo en esta área (al no considerarse incapacitante la pérdida ni la disminución de la función olfatoria) promueven que la evaluación de la función olfativa no sea parte de una valoración médica rutinaria [13].

Contar con un olfatómetro versátil que pueda ser utilizado en un consultorio, permitiría en una primera instancia incluir una valoración clínica más completa del olfato de los pacientes, haciendo posible la detección temprana de anomalías y/o desórdenes. Además de promover en el mediano plazo, la investigación de procesos *estímulo-reacción* al acoplar dicho sistema con estudios de EEG y RMf, al eliminar el componente subjetivo de las pruebas al evaluar la reacción de los sujetos mediante las señales eléctricas o imágenes capturadas.

1.3 Objetivo principal

Diseñar y construir un olfatómetro activo y portable de cuatro vías, capaz de estimular a pacientes con diferentes olores de forma automatizada. Las características del diseño deben permitir una simple selección de los olores y tiempos de estímulo, así como adquisición de registros suplementarios como potenciales evocados olfativos (OEPs) mediante Electroencefalografía (EEG).

1.4 Objetivos particulares

- Diseñar y construir la etapa de control de flujo y selección de odorantes del instrumento.
- Diseñar y construir la fuente de aire del sistema.
- Diseñar y construir la etapa de administración del instrumento.
- Integrar los componentes del sistema y verificar su funcionamiento.
- Establecer un protocolo de pruebas para la adquisición de potenciales evocados olfativos.
- Realizar pruebas y adquirir las señales de EEG con el protocolo propuesto.
- Organizar las señales de EEG adquiridas para su procesamiento en trabajos posteriores.
- Sincronizar el registro de EEG con otras señales suplementarias.

1.5 Estructura de la tesis

Esta tesis se divide en 6 capítulos que mencionan: En el primero, una breve introducción y justificación para el planteamiento de los objetivos de la presente tesis. En el segundo el marco teórico y estado del arte considerados como base para el diseño del instrumento. En el tercero la descripción de la solución propuesta así como sus componentes. En el cuarto las pruebas realizadas al prototipo y los resultados obtenidos. En el quinto una discusión de los resultados obtenidos contrastados con el estado del arte. Finalmente, en el último capítulo las conclusiones así como la perspectiva de trabajo a futuro y posibles mejoras.

Capítulo 2

Antecedentes

2.1 Marco teórico

2.1.1 El sentido del olfato

Los procesos sensitivos son la manera en que un ser vivo puede obtener información sobre su ambiente. El olfato es quizás el más sensible de todos, ya que es capaz de detectar la presencia de un bajo número moléculas incluso de pocas partes por trillón, esto quiere decir que basta una mínima cantidad de materia para estimular las células olfativas. A pesar de su importancia, el olfato ha sido uno de los sentidos menos estudiados en el humano, debido a que sus relaciones estímulo-reacción no están claramente descritas, sin embargo, en la actualidad el avance tecnológico ha permitido el estudio de las respuestas del cerebro y su relación con los potenciales generados por los órganos olfatorios [14].

Analizando de forma más detallada el olfato, podemos decir que es el sentido que se encarga de hacer un análisis de la composición química del entorno por medio de los componentes disueltos en el aire inhalado. En consecuencia es capaz de [14]:

- Identificar elementos potencialmente peligrosos (como alimentos en descomposición).
- Es auxiliar al sentido del gusto en la identificación de sabores.
- Genera algunas respuestas digestivas.
- Generar respuestas hormonales y emocionales.

2.1.2 Anatomía

De los cinco sentidos del cuerpo humano, el olfato es el más complejo en estructura y organización. Brinda el 80% de la sensación de sabor al comer, desempeñando un rol defensivo al producir una respuesta natural de aversión ante irritantes y "malos olores", haciendo uso de dos nervios principales para dicha tarea: El tracto olfatorio (primer nervio craneal) que transmite la información relativa a la percepción de odorantes químicos y el nervio trigémino (quinto nervio craneal) que transmite la información relativa a irritación o pungencia de compuestos químicos los cuales pueden o no ser odorantes [15].

Normalmente durante el proceso ventilatorio el 10% del aire inhalado pasa por el epitelio olfatorio en la parte superior-trasera de la cavidad nasal. Cuando se produce la acción denominada como olfateo ya sea de forma voluntaria o involuntaria más del 20% del aire inhalado es llevado al área de los receptores debido a la turbulencia producida y se calcula que sólo el 2% de las partículas odorantes llegan a ellos. Además, se ha reportado que la temperatura del aire no afecta en gran medida la percepción siempre y cuando se encuentre en niveles no peligrosos (10 a 35 °C) esto se atribuye a un rápido calentamiento del aire al pasar por la cavidad nasal [16].

El epitelio olfatorio se compone principalmente de células olfatorias y células de sostén como se muestra en la figura 2.1. Las células olfatorias son células nerviosas bipolares derivadas del sistema nervioso central y existen cerca de 100 millones de ellas en el epitelio. En su extremo forman un botón desde el que nacen de 4 a 25 cilios olfatorios que tienen un diámetro de 0.3 a 200 μm y se proyectan hacia el moco que reviste la cara interna de las fosas nasales ofreciendo cerca de 5 cm^2 de superficie de contacto. A su alrededor se encuentran esparcidas pequeñas *glándulas de Bowman*, las cuales secretan moco en la superficie del epitelio el cual actúa como trampa para los odorantes químicos, al pasar el aire por el epitelio olfatorio los odorantes son disueltos en el moco según su solubilidad en agua y otros factores de transferencia de masa entre los que se encuentran temperatura y presión. Entre más solubles al agua sean, se disolverán más fácilmente en el moco. Una vez que el odorante se difunde en el moco entra en contacto con los receptores de membrana de las células olfatorias (en el humano existen alrededor de 350 receptores

distintos que codifican aproximadamente 10,000 olores) ubicados en los cilios, produciendo una respuesta específica según la sustancia y su concentración [15].

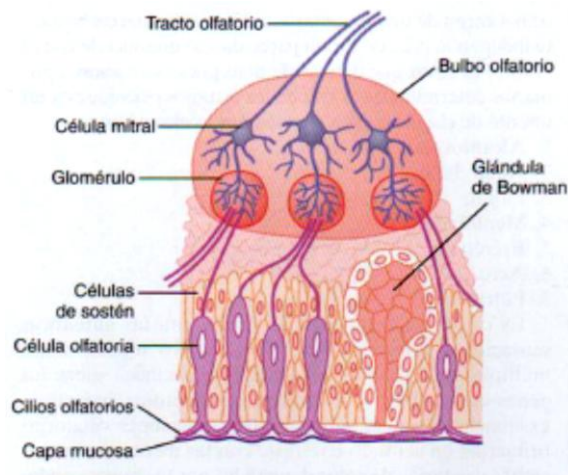


Fig. 1 Composición del epitelio Olfatorio[15].

Cada vez que se recibe una molécula compatible en un receptor se produce una respuesta eléctrica en la célula producto de la activación de la proteína receptora que a su vez activa múltiples moléculas de adenilato ciclasa por dentro de la membrana de la célula olfatoria, la cual convierte muchas moléculas de trifosfato de adenosina (ATP) en varias moléculas de monofosfato de adenosina cíclico (AMPc), quien a su vez activa una cantidad muy superior de canales regulados de ion de sodio produciendo la despolarización de la célula. Por lo cual no es necesaria de una gran cantidad de moléculas odorantes para desatar esa cascada de reacciones explicando la alta sensibilidad de las neuronas olfatorias. Cabe mencionar que la sustancia estimulante debe ser en principio volátil ya que sólo estas pueden inhalarse por las narinas además de tener un carácter hidrosoluble para atravesar el moco y llegar a los cilios [15].

Las porciones olfatorias del encéfalo figuraron entre las primeras estructuras cerebrales desarrolladas en los animales primitivos y gran parte del resto del cerebro se fue formando alrededor de este origen olfatorio. En realidad parte del cerebro que al principio se dedicaba al olfato más tarde evolucionó hacia las estructuras encefálicas basales que controlan las emociones y otros aspectos de la conducta humana: éste es el sistema que llamamos sistema límbico[15].

El bulbo olfatorio está representado en la Fig. 2. Las fibras nerviosas olfatorias que bajan desde el bulbo se llaman par craneal I o tracto olfatorio. Sin embargo, en realidad tanto el tracto como el bulbo constituyen una prolongación anterior del tejido cerebral que emerge desde la base del encéfalo: la dilatación bulbosa de su extremo, el bulbo olfatorio se halla sobre la lámina cribosa que separa la cavidad craneal de los tramos superiores de las fosas nasales. La lámina cribosa presenta múltiples perforaciones reducidas a través de las cuales asciende un número idéntico de pequeños nervios desde la membrana olfatoria en la cavidad nasal para entrar en el bulbo olfatorio dentro de la cavidad craneal. La Fig. 1 pone de manifiesto la íntima relación entre las células olfatorias y el bulbo olfatorio mostrando unos axones cortos que salen de ellas para acabar en múltiples estructuras globulares dentro del bulbo olfatorio que se llaman glomérulos. Cada bulbo posee varios miles de estos glomérulos y cada uno de ellos es el punto de terminación de unos 25,000 axones procedentes de las células olfatorias. Todo glomérulo también es la estación terminal para las dendritas de unas 25 grandes células mitrales y unas 60 células en penacho más pequeñas, cuyos cuerpos celulares se hallan en el bulbo olfatorio por encima de los glomérulos. Estas dendritas reciben sinapsis de las células neuronales olfatorias, y las células mitrales y en penacho envían axones a través del tracto olfatorio para transmitir señales olfatorias hasta niveles superiores en el sistema nervioso central [15].

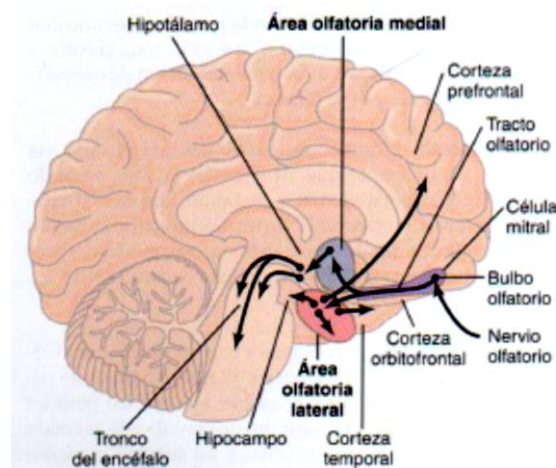


Fig. 2 Conexiones Nerviosas[15].

Algunas investigaciones han hecho pensar que los distintos glomérulos responden a olores diferentes. Es posible que cada glomérulo en cuestión sea el indicio real para analizar las diversas señales olorosas transmitidas hacia el sistema nervioso central y cómo funciona el preprocesamiento de las señales para distinguir olores individuales en una mezcla compleja a la cual es sometido regularmente un sujeto al interactuar con su ambiente [17]–[19].

El tracto olfatorio penetra en el encéfalo a nivel de la unión anterior entre el mesencéfalo y el cerebro; allí, se divide en dos vías, según se observa en la Fig. 2 una que sigue en sentido medial hacia el área olfatoria medial del cerebro y la otra en sentido lateral hacia el área olfatoria lateral. Esta primera estructura representa un sistema olfatorio arcaico, mientras que la segunda constituye la entrada para el sistema olfatorio antiguo y el sistema moderno [15].

El área olfatoria medial consta de un grupo de núcleos situado en las porciones basales intermedias del encéfalo inmediatamente delante del hipotálamo. Más visibles resultan los núcleos septales, los cuales son núcleos de la línea media que se nutren en el hipotálamo y otras porciones primitivas del sistema límbico cerebral. Ésta es la región del cerebro más vinculada con el comportamiento básico. La importancia de esta área olfatoria medial se entiende mejor si se considera lo que sucede en los animales cuando se eliminan las áreas olfatorias laterales de ambos lados del cerebro y no persiste más que el sistema medial. La realización de esta maniobra apenas influye en las respuestas más primitivas al olfato, como lamerse los labios, salivar y otras reacciones a la alimentación ocasionadas por el olor de la comida o por unos impulsos emocionales primitivos asociados a este sentido. En cambio, la supresión de las áreas laterales va a abolir los reflejos olfatorios condicionados complejos [15].

El área olfatoria lateral está compuesta sobre todo por las cortezas prepiriforme y piriforme además de la porción cortical de los núcleos amigdalinos. Desde estas zonas, las vías activadoras se dirigen hacia casi todas las porciones del sistema límbico, en especial hacia las menos primitivas como el hipocampo; que parece más importante para aprender a disfrutar de ciertos alimentos o a aborrecerlos en función de las experiencias personales vividas con ellos. Por ejemplo, se cree que esta área olfatoria lateral y sus abundantes conexiones con el sistema límbico de orden conductual hacen que una persona desarrolle una absoluta aversión hacia las comidas que le hayan provocado náuseas y vómitos. Un rasgo importante del área olfatoria lateral es que muchas vías estimuladoras procedentes de ella también nutren directamente la parte más antigua de la corteza cerebral llamada paleocorteza en la porción anteromedial del lóbulo temporal. Ésta es la única área de toda

la corteza cerebral a la que llegan directamente las señales sensitivas sin pasar antes por el tálamo [15], [18], [20].

Recientemente se ha descubierto una vía olfatoria que atraviesa el tálamo, pasando por su núcleo dorsomedial y llegando después al cuadrante lateroposterior de la corteza orbitofrontal. Según los estudios con monos, este sistema más moderno probablemente interviene en el análisis consciente de los olores. Muchas fibras nerviosas que nacen en las porciones olfatorias del cerebro siguen un trayecto hacia el exterior por el tracto olfatorio hasta el bulbo olfatorio (es decir, «centrífugo» desde el cerebro hacia la periferia). Acaban en una gran cantidad de pequeños granos situados entre las células mitrales y en penacho en el bulbo olfatorio. Los granos envían señales inhibitorias hacia estos dos tipos de células. Se cree que esta retroalimentación negativa podría constituir un medio para acentuar la capacidad específica de distinguir un olor de otro [15].

2.1.3 El código olfatorio

Como ya se menciona los tipos de receptores olfatorios en los humanos son cerca de 350 y para poder hacer una discriminación entre olores se ha encontrado que en promedio 10 de ellos son necesarios, por lo cual es posible hacer 6.8×10^{18} combinaciones entre receptores. Las cuales son muchas más que el número de compuestos químicos posibles. Permitiendo incluso la detección de compuestos que nunca existieron en la naturaleza, tal es el caso de los nuevos sabores y olores creados por el humano de forma sintética o mediante la cocina y procesamiento de los alimentos [21].

En los vertebrados sólo un gen de receptor olfatorio se expresa en cada célula olfatoria, para cada una de estas neuronas sólo habrá un receptor y todas las neuronas que expresan un mismo gen convergen en un mismo glomérulo, resultando en un mapa sensorial establecido. Estos a su vez se conectan con conjuntos establecidos de neuronas en la corteza olfatoria que permiten el procesamiento en paralelo de las señales olfatorias ya que se ubican en diferentes zonas a lo largo de su superficie antes de transportar la información al neocortex y sistema límbico [20].

Otro hallazgo es el hecho de la ausencia de uniformidad en el genoma olfatorio humano, estadísticamente ningún humano tiene el mismo conjunto de genes de receptores olfatorios funcionales. Explicando de alguna manera el porqué no existe una referencia semántica para el olfato, subsistiendo su carácter subjetivo y la imposibilidad de una desambiguación en la clasificación de olores [22]. Por otra parte se ha visto que la falta de expresión de dichos genes o daño en alguno de los glomérulos, produce una pérdida en la percepción de cierto tipo de olores a lo cual se le llama anosmia o hyposmia selectiva según sea el caso [4], [23].

2.1.4 Olores

Para comenzar debemos definir como olor, a la percepción/sensación que despierta un compuesto químico (odorante) que se encuentra en una concentración suficiente para desencadenar una respuesta en un ser mediante su sentido del olfato[24], [25]. Y desde el punto de vista de composición, es una mezcla compleja de varios compuestos orgánicos y algunos inorgánicos[26], [27]. Una parte importante para el estudio del sentido del olfato es el análisis y clasificación de los olores, esta clasificación se torna difícil debido a la subjetividad que representa el agrupar todos los olores en grandes grupos definidos o cualidades del olor. Se han propuesto distintos sistemas de clasificación siendo una de las primeras la clasificación propuesta por Carolous Linoaeus (1707-1778) el cual sugirió clasificar los olores en siete categorías: alcanforados, almizclados, florales, mentolados, con olor a éter, pungentes y pútridos. Posteriormente Hendrik Zwaardemaker (1857-1930) propuso 9 categorías: con olores a éter, aromáticos, fragantes, ambrosiacos, aliáceos, empireumáticos, caprinos, fétidos y nauseabundos. Un sistema un poco más elaborado es el propuesto por Hans Henning en 1916, clasificando los olores con una posición dentro de un prisma, usando 6 olores como olores primarios (frutal, floral, nauseabundo, quemado, resinoso y picante)[16], [28] como se muestra en la Fig. 3.

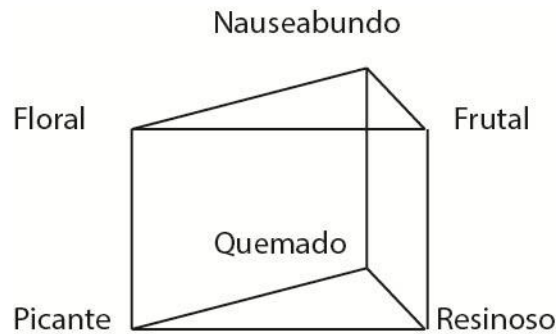


Fig. 3 Prisma de Henning.

Sin embargo, a pesar de los esfuerzos realizados hasta la fecha no existe un acuerdo para una única clasificación de los olores. Esto se debe a que como ya se explicó la percepción de los olores cambia de persona a persona, región o cultura. Ya que además de las diferencias genéticas existentes, para algunos un olor desagradable puede ser común debido a sus actividades diarias y entorno o simplemente puede ser desconocido[22], [28]. Si bien sabemos que existen cerca de 350 receptores y en los casos de anosmia selectiva(en ocasiones producto de ataques químicos, bacterianos o virales) se deja de percibir cierto grupo de olores[29], podríamos clasificar los olores en 350 componentes primarios. Siempre recordando que la percepción final depende de la mezcla presentada y del entrenamiento y/o experiencias del individuo; haciendo de la clasificación de los olores una tarea compleja.

2.1.5 Practica clínica

La subjetividad del sentido del olfato trae como consecuencia la dificultad de identificar sus anomalías y por ende, la necesidad de poder evaluar de una manera objetiva y cuantitativa las respuestas que se tienen ante algún estímulo olfativo. Para evaluar el estado del sentido del olfato es considerar todas las partes que involucran su funcionamiento, por lo que existen una serie de técnicas utilizadas en la actualidad para dicho fin, las cuales se explicaran brevemente a continuación[30].

Exploración básica e historia clínica. Implica una evaluación del historial médico tanto del paciente como de los familiares en busca de anomalías, así como un cuestionario en caso de que el paciente tenga consciencia de una alteración. Además de una exploración exhaustiva de las fosas nasales prestando principal atención al piso superior de las mismas.

También es necesaria la exploración de la totalidad de los pares craneales y el sentido del gusto para descartar probables enfermedades sistémicas.

Biopsia del epitelio olfatorio. La biopsia del epitelio se usa para estudiar el sustrato anatomopatológico en busca de patrones de degeneración de la mucosa en las distintas patologías. No se considera una prueba de rutina debido a lo invasivo de la misma.

Olfactometría. Es una prueba diseñada para controlar y manipular la concentración de los olores, valorar el estado olfativo de los seres vivos, en situación normal o patológica, así como cuantificar los resultados e interpretarlos. Posteriormente se explicara de manera más detallada.

Tomografía Computarizada. Una de las técnicas más aplicada, se utiliza para localizar posibles fracturas sobre el etmoides y tumores que afecten la región o el bulbo olfatorio.

Resonancia magnética (RM) y resonancia magnética funcional (RMf). Es útil para el análisis detallado de los tejidos blandos permitiendo encontrar tumores, malformaciones y lesiones. La RMf agrega el estudio de las zonas de activación del cerebro haciendo posible determinar cuales se activan a distintos estímulos y determinar posibles patologías si estos no son los esperados.

2.1.6 Olfatometría

En general la olfatometría es la ciencia encargada de hacer mediciones ya sea cuantitativas o cualitativas mediante el uso del olfato o sistemas que lo emulan. Actualmente suele enfocarse a la evaluación o caracterización ya sea de un olor o de un sistema olfatorio. A continuación se explicaran estos dos tipos de pruebas en la olfactometría. Al evaluar un olor existen cinco parámetros que pueden ser cuantificados, umbral de olfacción, intensidad del olor, persistencia, tono hedónico y carácter/clasificación [25]. Mientras que durante las pruebas que evalúan un sistema

olfatorio se manipulan: que olores se presentan al "sujeto" o sistema, la concentración de la mezcla y las condiciones ambientales en las que se presenta la estimulación.

En la evaluación de olores existen dos métodos por los cuales se puede llevar a cabo, el primero es la evaluación mediante instrumental en el cual se hace uso de cromatografías, tubos de detección, sensores de compuestos individuales e infinidad de instrumentos de medición para gases incluidas las narices electrónicas, el segundo es la evaluación sensorial u olfatoria en el cual se hace uso de un sistema olfatorio como sensor e instrumental (olfatómetros, bolsas de muestreo, plumillas, etc.) para presentar una muestra a los integrantes de un panel evaluador [25].

El ejemplo más usual en los análisis de olores es el de olfactometría ambiental el cual se encarga de evaluar la calidad del aire en zonas habitadas con el fin de regular posibles emanaciones irregulares que causen enfermedades a la población, este tipo de olfactometría depende de estrictas regulaciones internacionales ya que ayuda a determinar si las emanaciones producidas por algún proceso productivo superan los límites establecidos de contaminación y ayuda calificar las medidas tomadas para corregir los problemas. En esta categoría de olfactometría existen normas que regulan su práctica estableciendo los protocolos que deben llevarse a cabo tanto durante el muestreo como el manejo de las pruebas y selección del panel, como se discute en [31]. La normatividad se divide en dos grandes grupos las normas europeas y las normas americanas sin embargo, existen pocas diferencias entre ellas siendo la más completa la norma EN13725 [32].

También existen pruebas en las cuales se mezclan los métodos de olfactometría con instrumental y la percepción de un panel de asesores entrenados, siendo el caso más común el de la olfactometría en la industria alimentaria donde se busca relacionar la percepción del cliente con los componentes en el producto final o con etapas de control de calidad en el proceso de producción [33]–[35].

En la evaluación de sistemas olfatorios se busca analizar la respuesta de los sistemas a distintos estímulos, variando concentraciones de una sustancia, flujos de presentación, componentes de la mezcla o algún parámetro ambiental como temperatura, ruido, olores de fondo, etc., por lo que se hace uso de instrumental diseñado específicamente para cada

estudio o en su defecto se utiliza instrumental diseñado para olfactometría ambiental que reúne las características necesarias para la prueba. Se puede realizar este tipo de estudio en una gran variedad de seres vivos por ejemplo insectos para analizar su comportamiento ante ciertos olores [36], [37] y seleccionar un posible repelente [38] o en mamíferos de talla chica para ver los efectos de la exposición a ciertas concentraciones de una sustancia [39] o buscar alguna disfunción en su sistema olfatorio producto de alguna enfermedad neurodegenerativa [40].

En el humano la vía más frecuente para la transmisión de un olor es la nasal pero se debe tomar en cuenta que existen otras vías como corresponde a la inoculación por vía intravenosa ya que ciertos componentes pueden llegar a estimular los receptores olfativos debido a la irrigación sanguínea. Generalmente, los olores suelen ser administrados mediante un dispositivo auxiliar como una máscara, cánula o cabina [30]. Dentro de la olfactometría humana existen dos clasificaciones: La individual que como su nombre lo indica analiza la respuesta olfatoria de sólo un sujeto, buscando evaluar en las pruebas tres características del olfato: Identificación, discriminación y umbral de olfacción. Y la olfactometría colectiva o ambiental que como se mencionó anteriormente busca a través de un panel de prueba con sujetos seleccionados/entrenados detectar las condiciones organolépticas ambientales de un sitio determinado que pueden llegar a influir en la comunidad [30].

Las técnicas que se utilizan para evaluar la capacidad de identificación constan de la presentación de una serie de olores al sujeto a una determinada intensidad que se sabe perceptible, para que éste mencione el olor que percibe. En las pruebas de discriminación se presentan una serie de 3 ó 4 muestras de las cuales sólo una es distinta a las demás y se pide al paciente que la identifique, por último, en las pruebas de umbral de olfacción se presenta al sujeto un olor en distintas concentraciones las cuales suelen presentarse en forma ascendente (de menor a mayor concentración) hasta que el sujeto es capaz de percibirlo. Las técnicas antes mencionadas presentan un alto componente de subjetividad ya que dependen totalmente de la respuesta del sujeto [30].

2.2 Estado del arte

2.2.1 Olfatómetros

"Un olfatómetro es el instrumento utilizado para medir la sensibilidad del sentido del olfato con respecto a intensidad, concentración y calidad de un olor" [12] sin embargo, esta definición se ve un poco limitada al ver las aplicaciones que se le dan en la actualidad a este instrumental siendo un poco más apropiada la establecida en [41] que lo define como: "Un dispositivo para suministrar estímulos de olor con concentraciones y duración controlada".

Es difícil la clasificación de los olfatómetros por sus usos ya que uno de ellos puede ajustarse a más de una aplicación por lo que se debe recurrir a una clasificación generalizada como la planteada en [24]:

Olfatómetros pasivos: Aprovechan los movimientos respiratorios para que el olor penetre la cavidad nasal, no se controlan los factores externos como humedad, temperatura ni dilución.

Olfatómetros activos: Transmiten el olor por una corriente permanente de gas que se escapa de las rutas fisiológicas pero pueden ser controlados los factores externos.

2.2.2 Olfatómetros pasivos

En la primera clasificación se encuentran todos los kits de prueba rápida como: los "Sniffin Sticks", pruebas en tarjetas "Scratch & Sniff" (Rasca huele), pruebas con botellas de olor, kits de esencias que se presentan en cotonetes, etc. Ya que existen una gran variedad de estas pruebas se describirán las más usadas.

La prueba denominada Sniffin Sticks consiste en plumillas con punta de felpa, de 14cm de largo y diámetro de 1.3 cm similares a los plumones para pizarrón los cuales su tampón contiene en lugar de pigmentos 4 ml de un odorante líquido o una loción de odorante en propylenglycol, para ser presentados al paciente se destapa por 3 segundos y después se acerca a 2cm de cada una de las fosas nasales, se divide la prueba en 3 secciones como se menciona en [42]:

Prueba de identificación. Consiste en un set de 16 odorantes los cuales cumplen 3 criterios:(i) los sujetos de prueba deben estar familiarizados con los olores a utilizar; (ii) deben ser similares en intensidad y tono hedónico entre sí; y (iii) durante su selección hubo una identificación correcta >75% en pacientes sanos. En la prueba se pide al paciente identificar cada uno de los odorantes por opción múltiple entre 4 opciones, permitiendo al paciente olfatear las veces necesarias por menos de 30s para evitar desensibilación/fatiga. Se seleccionan sólo 16 odorantes para limitar el tiempo de la prueba

Prueba de discriminación. Consiste en 16 tripleteas de odorantes, Se presenta cada plumilla en una tripletea y se pide identificar el olor que es diferente, el paciente debe portar un antifaz para dormir con el fin de prevenir la identificación visual de los odorantes. Los criterios de selección de los odorantes fueron: (i) los olores en una tripletea deben ser similares en intensidad y de ser posible tono hedónico; y (ii) durante su selección hubo una discriminación correcta >75% en pacientes sanos. Sólo se permite al paciente olfatear una vez cada plumilla para mantener el tiempo de la prueba al mínimo, cada tripletea se presenta con un intervalo de 30 s entre ellas y entre cada odorante aproximadamente 3 s.

Prueba de umbral de olfacción, Al igual que la prueba de discriminación consta de 16 tripleteas dentro de las cuales sólo una contiene el odorante n-butanol a una dilución establecida y las otras 2 sólo contienen el diluyente y actúan como blanco(la dilución es establecida por una serie geométrica). Se pide al paciente que identifique la plumilla con el odorante a lo cual debe decir si lo detecta claramente o no. Para comenzar se presenta la plumilla conteniendo la mayor concentración de n-butanol para que el paciente se familiarice con el olor y sea capaz de identificarlo, posteriormente se coloca la mascarilla

para dormir. Las plumillas dentro de la tripleta son presentadas en orden aleatorio y un intervalo de 3 s, las tripletas se presentan en orden ascendente de concentración con un intervalo de 20 s entre cada una. Al producirse dos identificaciones correctas, se procede a comenzar de nuevo desde la tripleta 16 (la más diluida) este procedimiento se repite 7 veces tomando el promedio de las últimas 4 series como el umbral aproximado. También existe una variante en la que se utiliza 2-phenyletanol como odorante.

Finalmente, la prueba University of Pennsylvania Smell Identification Test (UPSIT) como se describe en [43] consta de 4 series de 10 tarjetas "rasca huele" tamaño sobre las cuales contienen un área con espesor de 10 a 50 μm de micro cápsulas de un polímero de formaldehído que contiene a los odorantes para liberarlos al ser frotadas con un lápiz y después presentadas al paciente, el cual debe de responder la pregunta de la tarjeta de la tarjeta utilizando una de sus 4 opciones por ejemplo: "¿Este olor se parece más a? a)chocolate, b) banana, c)cebolla o d)ponche de frutas". Y pueden ser aplicados en sus versiones fraccionadas de 10, 20 y 30 tarjetas cuando se requiere una prueba rápida en el ambiente el ambiente industrial si se sospecha una anomalía para después, aplicar el test completo a los sujetos que resultan con una calificación baja. Los criterios para la selección de los odorantes y preguntas se encuentran descritos en [44] al igual que la forma de calificarlas pruebas ya que depende de edad, género y otros factores.

2.2.3 Olfatómetros activos

En la clasificación de olfatómetros activos se encuentran una infinidad de dispositivos los cuales se pueden clasificar por sujeto de prueba, tipo de prueba, aplicaciones o instrumentos complementarios sin estar definida una clasificación formal; se describirán a continuación en orden ascendente de importancia para esta tesis los diseños de olfatómetros reportados en la literatura y disponibles comercialmente.

Comenzando con los olfatómetros enfocados al estudio de animales pequeños e insectos, estos suelen tratar de inducir una elección o monitorear el comportamiento de los sujetos de prueba ante un olor específico, existen diferentes variaciones de los olfatómetros

que se utilizan en estas aplicaciones, diferenciándose por características como el tipo de dispositivo de administración o el tipo de odorante que se puede utilizar. Algunos ejemplos de olfatómetros y sus aplicaciones son las presentadas en las investigaciones de Reddy, Adhikary, Birkett, Desbeux y Joly . En [45] Reddy utiliza un dispositivo con tubo "Y" para analizar la influencia que tienen algunos compuestos volátiles de diferentes plantas bajo situaciones características (infestadas/sanas/con daño mecánico/con daño por infestación) en el comportamiento de un depredador (*Chrysoperla carnea* Stephens) de ácaros que son considerados plaga (*Tetranychus ludeni* Zarcher). El procedimiento consiste en ubicar una de las plantas que se quiere utilizar en un canal del tubo y se hace pasar una corriente de aire limpio(1.2 l/min) por dos de los extremos del tubo y se coloca al depredador en el restante para ver el grado de predilección que tienen por dirigirse hacia el canal con la planta que funciona como odorante. En [46] Adhikary también utiliza un dispositivo con un tubo en forma de "Y" pero esta vez se hace uso de un extracto de semillas de *Lathyrus sativus* L. como odorante el cual es disuelto y se coloca en papel filtro para ser puesto en uno de los canales del olfatómetro mientras que en el otro canal se pone papel filtro con sólo el diluyente, en el canal restante se coloca una hembra de *Callosobruchus maculatus* para observar su comportamiento, el flujo de aire fue de 150 mL/min. La finalidad del estudio fue determinar que dosis y que variedades de *Lathyrus sativus* L. producen mayor atracción al insecto que suele alimentarse de las semillas después de su cultivo.

Birkett y Desneux en [47] y [48] hacen uso de un dispositivo de cuatro brazos por los que fluyen corrientes de aire iguales hacia una cámara centra en la cual se extrae el aire a un flujo constante para que el flujo de los brazos sea uniforme y no se produzcan turbulencias; en la cámara central se colocan los sujetos de prueba para evaluar si su comportamiento se ve afectado por el olor que fluye por sólo uno de los 4 brazos, [47] evalúa el efecto repelente de un extracto de *Nepeta grandiflora* y *Nepeta clarkei* sobre áfidos *S. avenae*, el flujo utilizado por brazo fue 60 mL/min y del extractor 240 mL/min, mientras que [48] evalúa la toxicidad de diferentes insecticidas sobre un insecto benéfico(Avispa parasítica *Aphidus ervi*) para hacer posible la recomendación de un insecticida que afecte en menor medida a dicho insecto; el flujo utilizado por brazo fue de 200 ml/min. En ambos casos se hizo uso de papel filtro para contener el odorante en uno de los brazos del olfatómetro.

Por último, en [49] Joly hace uso de una cámara con sólo un tubo que hace llegar el olor desde un olfatómetro de ocho canales de muestras líquidas, con el fin de comprobar la capacidad de discriminación de olores en un lémur *Microcebus murinus*, el flujo utilizado fue de 2 L/min (1.95 L/min de aire limpio y 0.05 de aire olorizado). Sin embargo, la arquitectura del olfatómetro usado es muy similar a los utilizados en humanos teniendo una fuente de aire limpio que se entrega de forma constante y otra para el flujo de aire olorizado que se activa y desactiva por computadora.

Comercialmente se pueden encontrar algunos olfatómetros para este tipo de pruebas siendo relativamente baratos ya que no requieren de un control temporal preciso, un ejemplo son los ofrecidos por Analytical Research Systems, Inc. que cuenta con modelos de olfatómetros de tubo en "Y" y de 4 brazos cuyos precios oscilan entre los 1,600 y 5,400 USD, el flujo se ajusta de forma manual mediante los medidores analógicos hasta 6 L/min y permite colocar muestras líquidas y sólidas contando con los accesorios adecuados para cada muestra, para su operación es indispensable disponer de una fuente de aire a presión [50].

Hasta este punto puede observarse que los olfatómetros tienen aplicaciones distintas a los estudios que se realizan en humanos y pueden ser una herramienta útil independientemente del sujeto de prueba, además de poder utilizarse todo tipo de odorantes incluso plantas completas, lo cual abre el panorama sobre el uso de distintos tipos de muestras.

Respecto a los dispositivos para uso en humanos puede encontrarse en el mercado un número reducido de olfatómetros activos diseñados para este fin, la mayor parte de ellos se encuentran enfocados a olfactometría ambiental y requieren que se les entregue la muestra gaseosa en bolsas Tedlar ®, por lo que se requiere de un sistema de muestreo adicional. Algunos de ellos son los olfatómetros de la serie TO de Odournet (Alemania) que manejan diluciones desde 1:4 a 1:65536, flujo máximo de 20 L/min, sistema de control de temperatura, dimensiones de 65 x 65 x 47 cm hasta la versión compacta de 65 x 65 x 47 cm, el flujo es regulado por orificios calibrados y cuenta con software para el cálculo de intensidad y tono hedónico [51]. Otra opción comercial son los sistemas de olfactometría

dinámica como el Nasal Ranger de St. Croix Sensory (EUA) que se utilizan para hacer mediciones *in situ* ya que funcionan diluyendo un flujo de aire ambiental con una corriente de aire limpio generada por un par de bombas al filtrar el aire con carbón activado y papel filtro, sus rangos de dilución son de 1:2 hasta 1:60[52].

Burghart Instruments por su parte, ofrece olfatómetros para estudios clínicos con 3 modelos de características diferentes el OL022 que maneja de 4 a 8 olores, regula el flujo de forma manual y controla la apertura de válvulas por PC; enfocándose a análisis de variables de cambio lento (ritmo cardiaco, comportamiento de memoria, etc.) sus dimensiones son 80 x 70 x 140 cm y peso de 90 kg. El OL023 que maneja estimulación monorinhal y birinhal, de 4 hasta 12 olores, cuenta con calefacción del flujo de aire, humidificación opcional y el flujo es controlado por controladores de flujo de masa, requiere de una fuente de aire externa y tiene una respuesta con flanco de subida de 100 ms (dependiendo de la longitud de la tubería), permite el uso de un disparador externo, sus dimensiones son 170 x 70 x 170 cm y peso de 400 kg; es apto para el estudio de potenciales evocados, electroolfatografía(EOG) y RMf. Por último, el OL024 siendo una versión compacta del OL023 pero de mejor desempeño ya que maneja un flanco de subida de 10ms, al ser una versión compacta sólo maneja 2 canales independientes que soportan muestras líquidas o gaseosa y un flujo de 8 L/min. El OL023 y OL024 son controlados totalmente por la PC mediante un software propio que permite la programación de las pruebas [53].

Las principales desventajas que muestran estos sistemas para la investigación es la difícil adaptación a las pruebas requeridas, ya que alguno de los parámetros no se ajusta a las necesidades de las etapas experimentales o se torna casi imposible su interacción con otros dispositivos de registro, además de ser muy costosas las opciones automatizadas. Dejando a los investigadores con la única opción de construir su propio olfatómetro, en general las propuestas hacen uso de una computadora para controlar de forma sencilla y precisa los tiempos de estimulación además de tratar de seguir los criterios propuestos Stone, Pryor y Steinmetz como se menciona en [54] (i) Contar con un flujo continuo del aire diluyente, (ii) flexibilidad en la selección de estímulos y su concentración/intensidad, (iii) rápida inserción y remoción del estímulo(tiempos de subida y bajada mínimos), (iv) facilidad de operación, registro de datos y limpieza del instrumento. Además de algunos

requerimientos que se consideran opcionales (v) evitar indicios/pistas de activación, (vi) control de temperatura y humedad, (vii) calibración/cambio del flujo de aire y (viii) evaluación objetiva de las concentraciones del odorante.

Las propuestas más recientes de olfatómetros plantean el poder utilizarlos con las técnicas de RMf y EEG que requieren de un control temporal más estricto, algunos ejemplos son [55]–[60] donde en todos los casos se plantea el uso de tubería y componentes que eviten la contaminación por los odorantes, alta resistencia química y carezcan olor propio como el Politetrafluoroetileno (PTFE), manejan un número de canales que oscilan entre 1 y 8 canales, aunque algunos incluso manejan hasta 15 [55], los métodos de control de flujo utilizados son el ajuste manual de válvulas de flujo o el uso de controladores de flujo de masa, el uso de los controladores de flujo de masa mejora el desempeño del sistema pero aumenta significativamente el costo del sistema. El flujo máximo de los sistemas va de 1.5 L/min a 20 L/min, las muestras que soportan suelen ser líquidas con excepción de algunos diseños como los presentados por Sezille y Lundström en [56] y [57] que contempla el uso de muestras sólidas, en [56]–[58] Sezille, Lundström y Johnson hacen uso de un detector de foto-ionización para caracterizar la concentración de salida en algunos de los odorantes que se utilizarían en contraste Benignus y Popp utilizan RMf para comprobar la estimulación en el paciente al analizar directamente las imágenes adquiridas en [54] y [59]. Una última característica es la posibilidad de registro de señales auxiliares como se plantea en el trabajo realizado por Vigouroux donde se hace uso de un software para integrar las mediciones de pletismografía, respiración, reacción electro dérmica y una escala analógica de percepción del olor junto con los registros temporales de estimulación [60].

Una vez analizada la literatura se escogieron los trabajos de Sommer y Sezille como los diseños más representativos por lo que se expondrán a detalle. En [55] Sommer propone un olfatómetro compatible con los estudios de RMf por lo que los componentes que interaccionan con el paciente deben ser no conductivos y no magnéticos. El diseño fue enfocado a ser simple y barato, por lo que se utilizaron componentes estándar (industriales y de laboratorio). Dividiéndose en 3 secciones principales "toma de aire, control y distribución", " Sección de odorantes" y " sección de administración".

La sección de toma de aire, control y distribución se compone por un medidor de flujo el cual se conecta a una toma de aire limpio del hospital o clínica para ser ajustado el flujo mediante la válvula del mismo. El aire posteriormente se divide y pasa por 4 electroválvulas que se conectan antes de los recipientes de los odorantes para la selección de los mismos. Los recipientes de los odorantes son 4 lavadores de gas en los cuales son puestos las muestras líquidas para exponer el aire a la máxima superficie posible del odorante. La sección de administración consta de mangueras que se conectan 10 cm antes de la nariz del sujeto en una vía que mide 5cm la cual termina en una conexión en Y para ser administrado a cada fosa nasal con un flujo total de 2L/min. El software utilizado para su control fue compilado para funcionar en Linux utilizando GNU C++, permite controlar la activación de las 5 válvulas (Encendido y los 4 odorantes) brindando como intervalo mínimo entre instrucciones 50 ms. Se comprobó la correcta estimulación mediante pruebas de MRf a 21 pacientes normósmicos en los cuales se vio la activación típica de las zonas cerebrales.

Por otra parte Sezille en [56] expone el diseño de un olfatómetro enfocado a ser utilizado en estudios de MRf que dentro de sus características principales contempla la portabilidad, bajo costo y el hacer el registro de la señal respiratoria y percepción del paciente dentro del mismo dispositivo. Cuenta con una interfaz USB y RS232 para poder ser controlado mediante la PC. Puede ser descrito en cinco módulos como puede observarse en la Fig. 4: Fuente y tratamiento de aire, estimulador (válvulas y control electrónico) y fuentes de olor, cabeza de mezclado, sensor de flujo respiratorio y caja de respuesta (botones para indicar cuando se percibe el olor) y por último, el software de control (Labview).

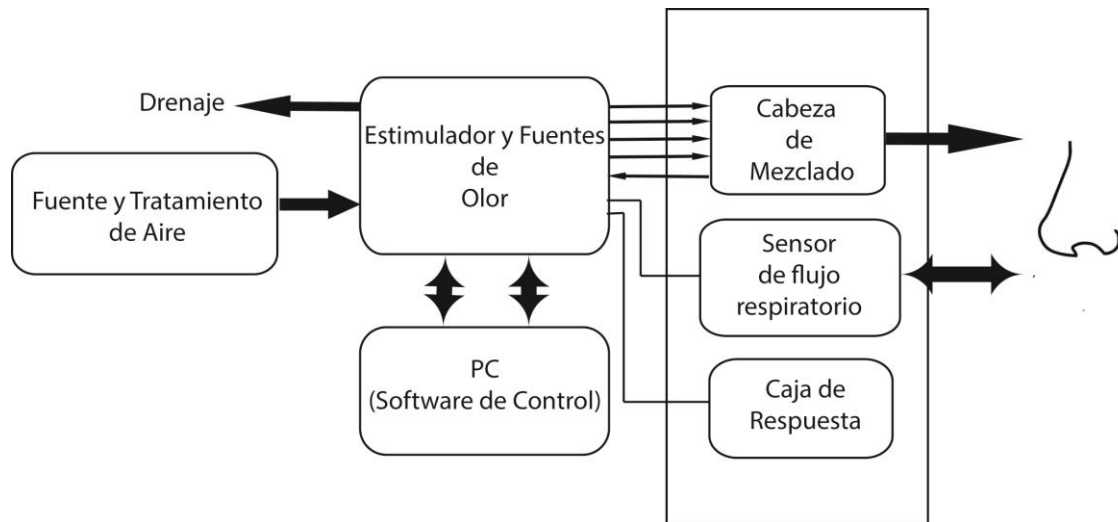


Fig. 4 Diagrama a bloques del olfatómetro propuesto por Sezille.

Este sistema muestra la particularidad de detectar el flujo respiratorio para disparar la estimulación olfativa en el momento de la inhalación, la cantidad de olores que pueden ser mezclados dependen de la cabeza de mezclado y recipientes con los odorantes siendo el máximo 15. Se utilizaron controladores de flujo de masa para regular las concentraciones de los odorantes y se validaron los resultados mediante detección por foto-ionización mostrando que efectivamente la percepción de intensidad en el olor tiene una relación lineal con la concentración del odorante presentado. Por último, las pruebas de fMRI mostraron activación de las zonas olfatorias típicas durante la estimulación.

Una característica común de los dispositivos reportados es el uso de controladores de flujo de masa o reguladores de flujo manuales junto con medidores analógicos a la entrada del sistema para establecer los flujos de salida de los dispositivos. Sin embargo, para que los flujos de entrada y salida de los dispositivos sean los mismos, se debe considerar al aire como un fluido no compresible, lo cual es incorrecto.

Analizados los dispositivos comerciales y los reportados en la bibliografía se puede observar la necesidad de un instrumento que pueda ser manipulado fácilmente, sin la necesidad de instalaciones especiales y sea portable. Lo cual ayudaría a la diversificación de experimentos y registros.

Capítulo 3

Solución Propuesta

El presente capítulo se encuentra dividido en 6 secciones. La primera de ellas describe la solución propuesta en la presente tesis, para conseguir los objetivos propuestos en la investigación. Las secciones restantes describen las etapas de diseño y construcción del olfatómetro desarrollado.

3.1 Propuesta de diseño

Para definir las características del dispositivo se tuvo en cuenta el estado del arte; considerando que existen dispositivos que utilizan desde 1 a 15 canales y las versiones estándar de los dispositivos comerciales manejan un mínimo de 2 ó 4 con opción a expansión se decidió hacer un dispositivo de 4 canales, el cual permitirá la programación de un protocolo cíclico con diferentes olores para evitar la fatiga olfativa en el sujeto de prueba. Como se puede apreciar en el estado del arte ninguno de los puntos descritos considera la portabilidad del instrumental o evitar la necesidad de instalaciones especializadas para su uso por lo que se plantea hacer uso de estos dos criterios. Se propone la fabricación de un olfatómetro portable cuyas características principales sean:

1. Diseño portable(bajo peso y dimensiones) y modular.
2. Disponer de 4 canales independientes para odorantes y un canal extra para aire limpio.
3. Capacidad de controlar mediante PC los tiempos de activación y flujos de salida (aire limpio y aire olorizado).

4. Tener fuente de aire propia junto con un sistema de filtrado que evite su contaminación.
5. Sección de administración independiente.
6. Contar con la posibilidad de conectar de hasta 8 sensores auxiliares y un puerto de entrada/salida digital de 5 bits para posibles mejoras en el diseño.
7. Capacidad de manejar muestras líquidas y sólidas.
8. Permitir la correcta adquisición de los potenciales evocados olfativos (OEPs).
9. Estar fabricado con materiales de baja contaminación a olores y alta resistencia química.

Estas características permitirán la implementación de pruebas de olfacción de forma automatizada y versatilidad para probar diferentes tipos de valoración a distintos grupos de prueba. Del mismo modo, facilitará el intercambio y preparación de las muestras a utilizar. El diseño modular propiciara que el sistema pueda ser compatible con estudios de **RMf** en una segunda fase de implementación realizando cambios mínimos en el diseño del instrumento y su mantenimiento sea fácil de realizar.

3.2 Diseño y construcción del olfatómetro

El diseño general del instrumento se divide en 5 secciones principales: fuente de aire, tratamiento de aire, selección de odorantes, control de flujo y la sección de administración como se muestra en la Fig. 5. Además de incluirse una sección para conectar hasta 8 sensores analógicos auxiliares y contar con 5 pines de entrada/salidas digitales con la finalidad de soportar distintos dispositivos de entrada salida así adaptarse a distintos experimentos de laboratorio.

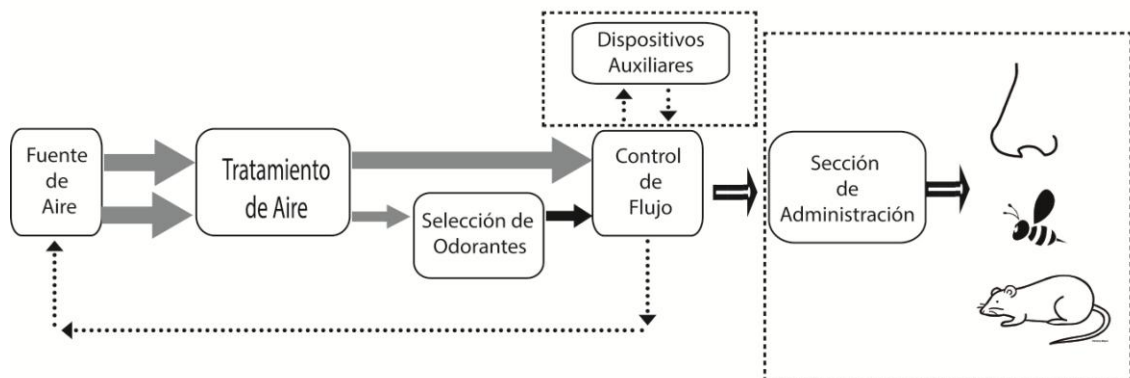


Fig. 5 Diagrama a bloques del olfatómetro propuesto. (Flecha gris corriente de aire limpio, Flecha negra corriente de aire olorizado y flecha punteada control electrónico).

El sistema suministra una corriente de aire limpio constante que se mezclará con otra que porta el olor con el que se quiere estimular por periodos de tiempo determinados para presentar al paciente aire limpio u olorizado según se requiera. La corriente olorizada se genera haciendo pasar una corriente de aire limpio por un recipiente que contiene una solución líquida o muestra sólida. El olfatómetro puede manejar hasta 4 odorantes simultáneamente. La selección del odorante se hace activando una serie determinada de electroválvulas que son controladas por un microcontrolador que recibe las señales desde una interfaz gráfica de PC. La corriente de aire limpio y olorizado se mezclan justo antes de ser presentadas al paciente permitiendo controlar la razón de dilución entre ambos. Los materiales utilizados para la fabricación del instrumento tienen una alta resistencia química y a crecimientos bacterianos con la finalidad de reducir el mantenimiento y evitar su contaminación a lo largo del tiempo. Por ello, para la sección de aire olorizado se utilizaron tuberías de Politetrafluoroetileno_(PTFE) de la empresa Supelco y Tygon® de la empresa Saint-Gobain. En la Fig. 6 se muestra el diagrama neumático del instrumento.

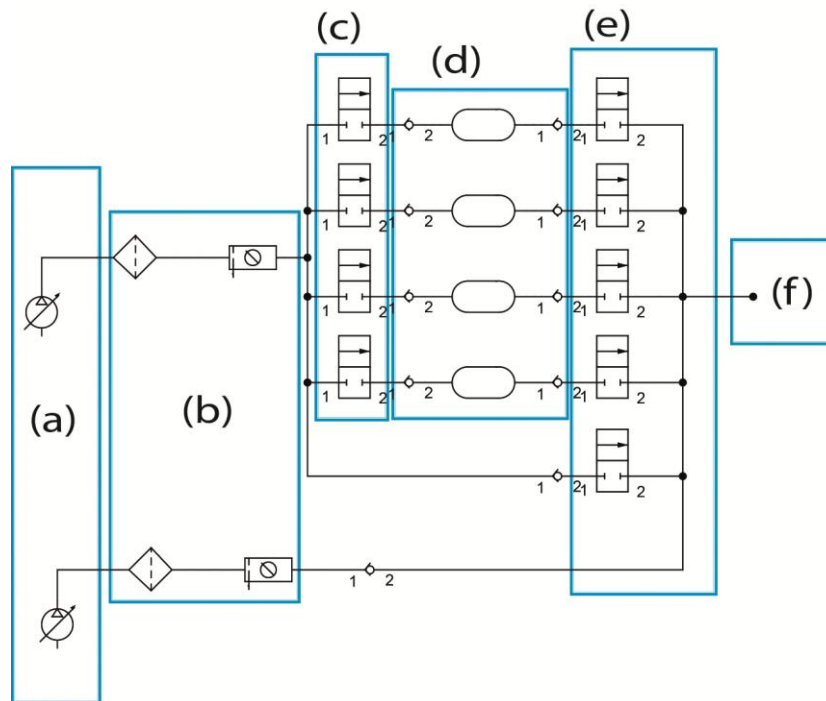


Fig. 6 Diagrama a neumático del olfatómetro (a) Fuente de aire (b) Filtrado y regulador de seguridad (c) Válvulas de selección (d) Contenedores (e) Mezclador (f) Sección de administración (salida).

3.2.1 Fuente de Aire

La fuente de aire se compone por dos bombas de DC que proporcionan un flujo máximo de 12 LPM y presión máxima de 32 PSI (AIRPON, Mod. D2028), una bomba se encarga de alimentar la corriente de aire limpio y la otra el aire que será olorizado facilitando el control independiente de flujos.

3.2.2 Tratamiento de Aire

En esta etapa se filtran los componentes que pudiera contener el aire ambiental para poder entregar un aire libre de olores, esto se logra mediante un filtro de partículas de $5\mu\text{m}$ con regulador y manómetro (FESTO, Mod. MS4-LFR-1/4-D5-CRM-RG) junto con un filtro de carbón activado (FESTO, Mod. MS4-LFX-1/4-R). El regulador con manómetro sirve para obtener lecturas de la presión del sistema, además de funcionar como una medida de seguridad para evitar la apertura no programada de las electroválvulas de los sistemas subsecuentes al evitar un aumento excesivo de presión, fijando la presión máxima en 2 bar.

3.2.3 Selección de odorantes

Esta etapa se encarga de seleccionar los odorantes que serán administrados mediante las señales de un microcontrolador (Texas Instruments, Mod. TMS320F28027) para la activación de una de las 4 electroválvulas (Cole Parmer, Modelo 98302-06) que permiten el ingreso de aire limpio a los contenedores de vidrio con los odorantes. Siendo la restricción contar con boquilla de diámetro 2.5 cm, para permitir la conexión mediante los tapones de goma que tienen los orificios para introducir la tubería que porta el aire limpio, la tubería de lectura del sensor de presión y la válvula check de salida. El volumen máximo sugerido para los contenedores depende del tipo de muestra, de esta forma, para muestras líquidas se manejarán 130 ml mientras que para muestras sólidas es de 70ml. En el caso de las soluciones líquidas el contenedor funciona como burbujeador, ambos contenedores se muestra en la Fig. 7.



Fig. 7 Contenedores para muestras odorantes a) muestras líquidas b) muestras sólidas.

Para un óptimo desempeño del dispositivo se debe tener un volumen libre entre 30 y 50 ml en cada contenedor con el fin de evitar la entrada de muestra a las válvulas de salida. La salida de los contenedores se une en un "mezclador" 5x2 normalmente cerrado (NResearch Inc., Mod 161T101) que permite seleccionar el odorante que se mezclara con la corriente de aire limpio para ser presentada al paciente. Todos los elementos que mantienen contacto con las corrientes del aire se encuentran fabricados de PTFE para evitar contaminación de olores.

3.2.4 Control de flujo

En esta sección se controla el flujo de aire que se desea a la salida, mediante control proporcional tomando como entrada la señal de los sensores de presión (Epcos, Mod. ASR 0.400 V1 TN H30) que toman lectura de la presión dentro de los contenedores para regular mediante PWM la activación de la bomba de aire[61]. La decisión de implementar el control de flujo por presión que se describirá posteriormente se hizo tomando en cuenta que la alteración de la composición de la mezcla de salida afecte en menor medida la lectura del sensor, cosa que no es posible con un controlador de flujo de masa ya que para entregar un resultado confiable se necesita calibrar según la composición del gas a medir (humedad, concentración, temperatura, etc.) lo cual se ignora para la mayoría de las muestras.

$$q = \left(1 - \frac{2\Delta p}{3p_1}\right) N_2 C_v p_1 \sqrt{\frac{\Delta p}{p_1 G_g T_1}} \quad (1)$$

La relación entre presión de entrada y flujo de salida de la válvula check se muestra en la ecuación (1) donde q es el flujo de salida, N_2 la constante de proporcionalidad, C_v el coeficiente de flujo de la válvula, p_1 la presión de entrada, G_g la gravedad específica del gas, T_1 la temperatura de entrada y Δp la diferencia de presión entre entrada y salida. En la Fig. 8 se muestra un diagrama del sistema que representa la ecuación.

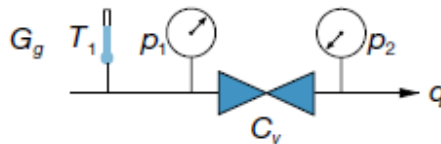


Fig. 8 Diagrama de prueba para la obtención de la ecuación (1).

Las válvulas check utilizadas (Cole Parmer, Mod. 01355-00) cuentan con un $C_v = 0.014$ y una presión de apertura de 0.069 bar además de la constante $N_2 = 6950$ al introducir presiones en [bar] y temperatura en [K] para entregar q en [std L/min]. Siendo el rango de operación de los sensores de presión de 0 bar a 0.4 bar y entregando una salida proporcional entre 0 V y 1 V se introducen al ADC del microcontrolador utilizando la conversión de 12 bits. Resultando la gráfica que se muestra en la Fig. 9 la cual relaciona el

valor del ADC y el flujo a la salida de la válvula check; teniendo en cuenta la presión de apertura de las válvulas.

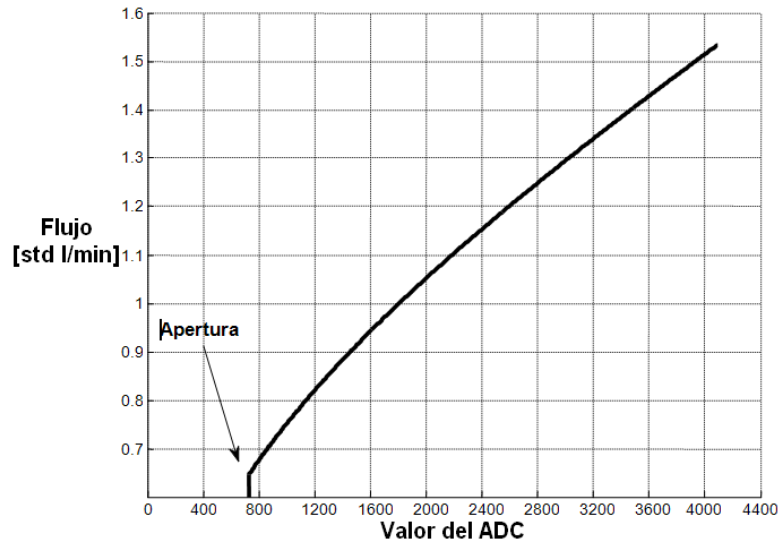


Fig. 9 Gráfica que muestra la relación del valor entregado por el ADC y el Flujo a la salida del sistema.

Las reglas de control implementadas se basan en un control proporcional y están representadas por las ecuaciones (2) y (3) siendo P_o la presión objetivo del sensor y P la lectura actual del sensor, ambos valores expresados por el valor en bits del ADC. El término constante en (3) con valor igual a 5 representa el ciclo de trabajo mínimo con el cual las bombas se ponen en funcionamiento, el segundo término el valor base necesario para mantener la presión objetivo y como último, el ajuste proporcional tomando en cuenta el error del sistema y la presión objetivo.

$$\Delta p = P_o - P \quad (2)$$

$$\%PWM = 5 + \frac{P_o}{100} + \frac{\Delta p}{1024/P_o} \quad \text{para } \Delta p > 0$$

$$\text{sí } \%PWM > 100 \Rightarrow \%PWM = 100 \quad (3)$$

$$\%PWM = 5 + \frac{P_o}{100} \quad \text{para } \Delta p < 0$$

De igual forma se implementó una ecuación de control diferente (4) para preparar las muestras antes de la activación de las válvulas de salida de los contenedores, ya que no existe un flujo salida aumenta la presión rápidamente y ésta sólo disminuye por las

pequeñas fugas de los contenedores se utiliza una constante proporcional más baja y se elimina el segundo término. Además de apagar la bomba al superar la presión objetivo.

$$\%PWM = 5 + \frac{\Delta p}{2048/P_o} \quad \text{para } \Delta p > 0$$

$$\text{sí } \%PWM > 100 \Rightarrow \%PWM = 100 \quad (4)$$

$$\%PWM = 0 \quad \text{para } \Delta p < 0$$

3.2.5 Sección de administración

En esta sección se toma la salida del sistema y se adapta según las necesidades de la prueba o del paciente. Por ejemplo la utilización de una mascarilla o cánula para oxígeno ya sea pediátrica o estándar, el uso de un tubo en Y para pruebas con insectos o simplemente un pequeño tubo para liberar el olor al ambiente. Se debe tomar en cuenta que a mayor longitud de la tubería utilizada mayor será el retraso del estímulo y se recomienda utilizar diámetros pequeños pero no menores a 1mm para no afectar el flujo de salida calculado.

3.3 Diseño electrónico y programación

Como controlador principal del olfatómetro se utiliza una tarjeta de desarrollo Launchpad XL con un microcontrolador TMS320F28027. Éste controla la ejecución de las instrucciones recibidas desde la interfaz gráfica en la PC coordinando la activación de las electroválvulas y bombas de aire en el sistema, de igual manera envía las señales de los sensores auxiliares por medio del puerto USB. Además de ejecutar la rutina de control para el flujo de aire configurado.

La tarjeta de desarrollo se debe conectar a otra que contiene los buses para interconexión de los dispositivos de entrada y salida a modo de facilitar el intercambio de

piezas en caso de requerir mantenimiento. Como entradas tiene el puerto USB para recibir las instrucciones a ejecutar y los 4 sensores de presión para ejecutar la rutina de control. Como salidas se tienen los módulos de potencia 1 y 2 a los cuales se conectan las 9 electroválvulas del sistema y las 2 bombas de aire y el puerto USB para enviar información a la PC del estado del sistema y los datos de los sensores auxiliares. Además, un puerto de entrada/salida de 5 bits para conectar dispositivos auxiliares y 8 entradas para sensores analógicos auxiliares. Las conexiones y flujo de datos del sistema general se muestran en la Fig. 10.

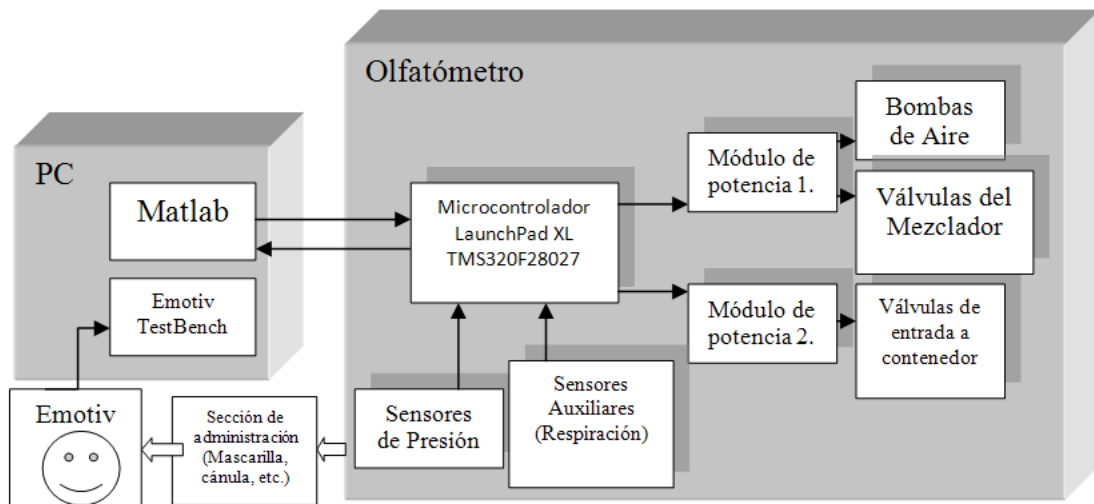


Fig. 10 Diagrama a bloques del sistema (flechas negras flujo de datos- flechas blancas corriente de aire a la salida).

Los módulos de potencia constan de una serie de optocopladores (Fairchild Semiconductor, Mod. 4N35) conectados a transistores MOSFET (ST Microelectronics, Mod. BUZ11) para controlar el encendido y apagado de las electroválvulas y bombas. El control de las bombas de aire se hace mediante el módulo de PWM del microcontrolador a una frecuencia de 100 Hz. El diagrama eléctrico de dicha interfaz para un dispositivo se muestra en la Fig. 11.

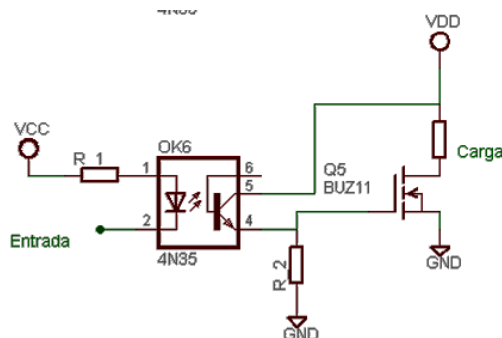


Fig. 11 Diagrama eléctrico de un canal de las interfaces de potencia.

La potencia máxima de las bombas de Aire es de 12 W a 12 V y máxima carga, para las electroválvulas de entrada a los contenedores se tiene una potencia de 4 W a 12 V y para las del mezclador 1.15 W a 12 V. Los voltajes de operación de los dispositivos permiten que se alimenten con la misma fuente. Consumiendo en total un máximo de 47.57 W en la sección de potencia en el caso de encender todas las electroválvulas y bombas. En funcionamiento normal el consumo promedio es de 29.8W en flujo máximo. Una vez calculada la potencia del sistema se determina que en condiciones normales se requiere de una fuente de 12 V 2.5 A para su correcto funcionamiento o bien se podría utilizar una batería de 3300 mAh a 12 V considerando una autonomía de una hora y una profundidad de carga del 75%.

El módulo de potencia 1 contará con 7 canales que son los correspondientes a los dispositivos que son manipulados durante el ciclo de control mientras el módulo 2 contará con cuatro canales que se conectaran a las electroválvulas involucradas en la preparación de la muestra. Resultando en un total de 11 canales de salida los cuales serán manipulados mediante lógica negativa por el microcontrolador.

En cuanto a la obtención de señales de los sensores de presión, se utilizó una referencia fija de 2.048 V (LM4140 de Texas Instruments) al ADC del microcontrolador siendo la mínima permitida 1.98 V, teniendo como resultado que la variación del valor del ADC en un bit exprese un cambio de 0.5 mV en la entrada. Tomando en cuenta que los sensores entregan una señal proporcional a la presión de 0 a 1 V y 2 mA no es necesaria una etapa de adquisición más elaborada. Aunque debe tenerse en cuenta que no se está utilizando el rango completo del ADC y con sólo los 11 bits menos significativos se expresa el valor de los sensores de presión. Los sensores son alimentados por el puerto USB de la computadora ya que soportan una alimentación desde 2.7 V a 5.5 V y demandan una corriente de 2.5 mA mostrando como beneficio extra la separación de la fuente de las secciones de potencia. Para acoplar los sensores con los contenedores se hace uso de una tubería de PTFE con diámetro interno 1mm la cual conecta el interior del contenedor con la superficie de sensado.

Las conexiones disponibles para sensores auxiliares se encuentran conectadas a entradas analógicas del microcontrolador por lo que puede ser utilizadas para conectar cualquier sensor que entregue una salida entre 0 y 2.048 V, como sensor auxiliar se agregó un sensor de temperatura (LM35 de Texas Instruments) dentro de la mascarilla utilizada, a modo de obtener un registro aproximado de las inhalaciones y exhalaciones del sujeto de prueba; para el correcto análisis de las señales de EEG adquiridas en las pruebas realizadas. Considerando que dicho sensor entrega una lectura de 10 mV/°C y teniendo un rango de -55 °C a 150 °C la salida máxima sería de 1.5V por lo que puede conectarse directamente para registrar variaciones de ± 0.05 °C/bit. Siendo suficiente para detectar las variaciones entre la temperatura dentro de la mascarilla durante la inhalación y la exhalación.

Respecto a la programación del microcontrolador se eligió utilizar una estructura de ejecución mediante *polling*, esto se refiere a que una vez inicializado el sistema se comienza un proceso cíclico el cual se ejecuta de manera indefinida modificándose sólo cuando se recibe una instrucción válida mediante la UART/USB. A esta estructura se le agrega un temporizador con frecuencia de 128 Hz el cual genera una interrupción para la adquisición de los datos de los sensores auxiliares y los envía hacia la PC. El diagrama de flujo del programa principal puede observarse en la Fig. 12. Las instrucciones que pueden ser enviadas al olfatómetro son las expresadas en la Tabla 1.

Dichas Instrucciones son coordinadas mediante MATLAB utilizando una interfaz gráfica que se describirá más adelante. O pueden ser enviadas por cualquier otro software que se desee utilizar como Labview o alguna aplicación desarrollada en otros lenguajes de programación.

La programación del microcontrolador se hizo utilizando CodeComposer 5.5.0 en lenguaje C++ con ayuda de las librerías proporcionadas por el desarrollador. Los programas documentados pueden revisarse, en la sección del Anexo A.

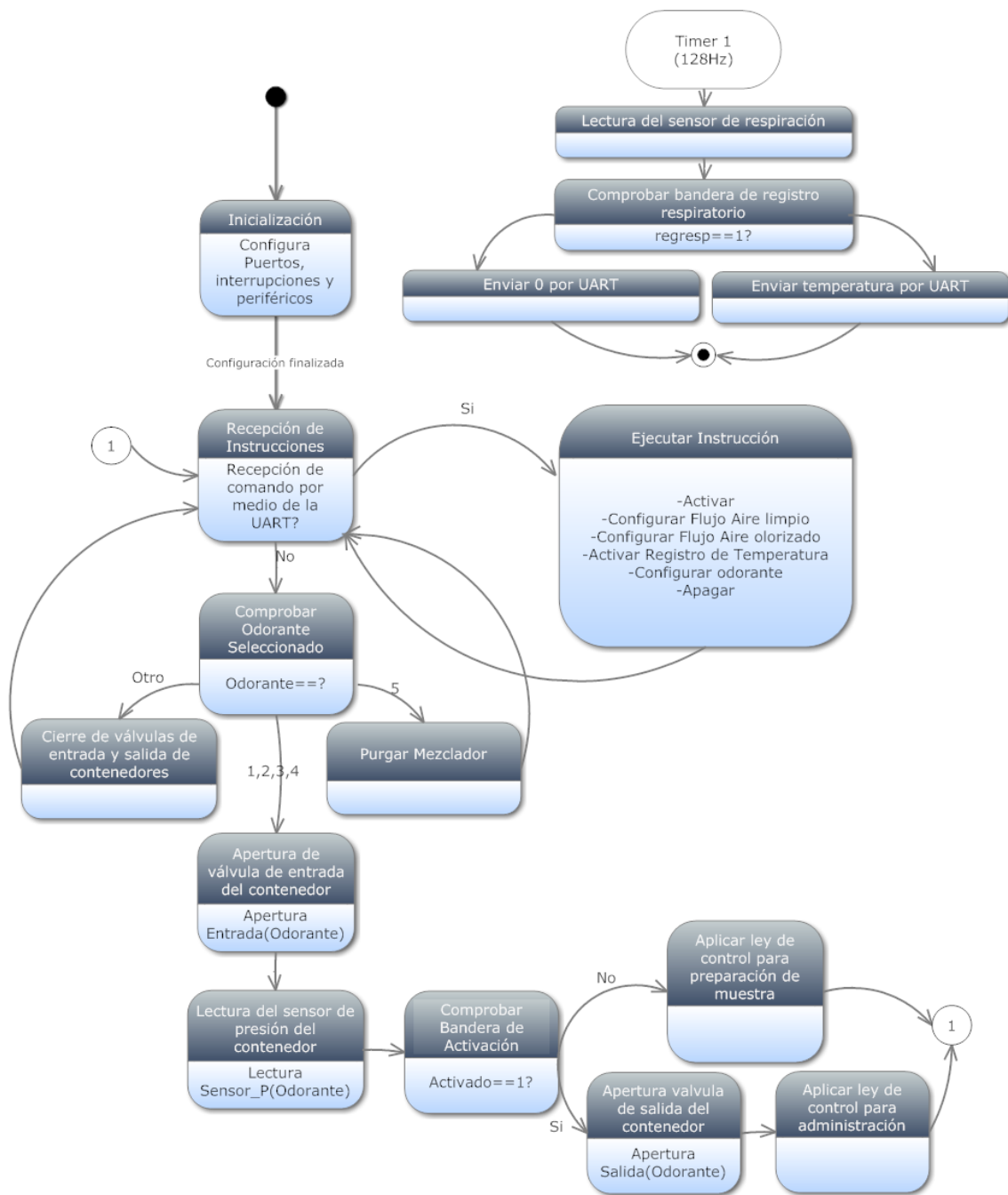


Fig. 12 Diagrama de flujo del programa principal del olfactómetro.

Tabla 1. Instrucciones del dispositivo

Instrucción	Código a enviar por el puerto serie	Acción
Activar	0xF0	Modifica la bandera de activación en el proceso principal.
Configurar flujo de aire limpio	0xFE + (%PWM)	Modifica el valor del PWM que controla la bomba que proporciona la corriente de aire limpio.
Activar registro de temperatura	0xFA + (Valor del ADC de la presión objetivo)	Modifica el valor objetivo utilizado en la ley de control dentro del proceso principal.
Configurar Odorante	0xEF + (Numero del odorante)	Modifica el registro de Odorante utilizado en el proceso principal; selecciona el odorante a utilizar.
Apagar	0xFD	Apaga todas las válvulas y bombas

3.4 Interfaz de Usuario

Para facilitar el manejo del dispositivo se programaron dos interfaces gráficas en Matlab sus códigos pueden observarse en el Anexo A. La primera interfaz permite seleccionar el flujo de aire olorizado, el flujo de aire limpio, editar las etiquetas de los cuatro odorantes, cuál de los cuatro odorantes será administrado y el tiempo del estímulo. También se selecciona el puerto COM asignado al olfatómetro y el puerto para enviar los marcadores para el registro de EEG. Los marcadores envían la información de la etiqueta del odorante, duración del estímulo, inicio del estímulo y el flujo de aire limpio y olorizado. Además de contar con una hyperterminal para poder enviar y recibir datos de forma manual al dispositivo o alguna aplicación de registro esto con fines de depuración del sistema y pruebas de caracterización. La ventana desplegada en MATLAB se muestra en la Fig. 13.

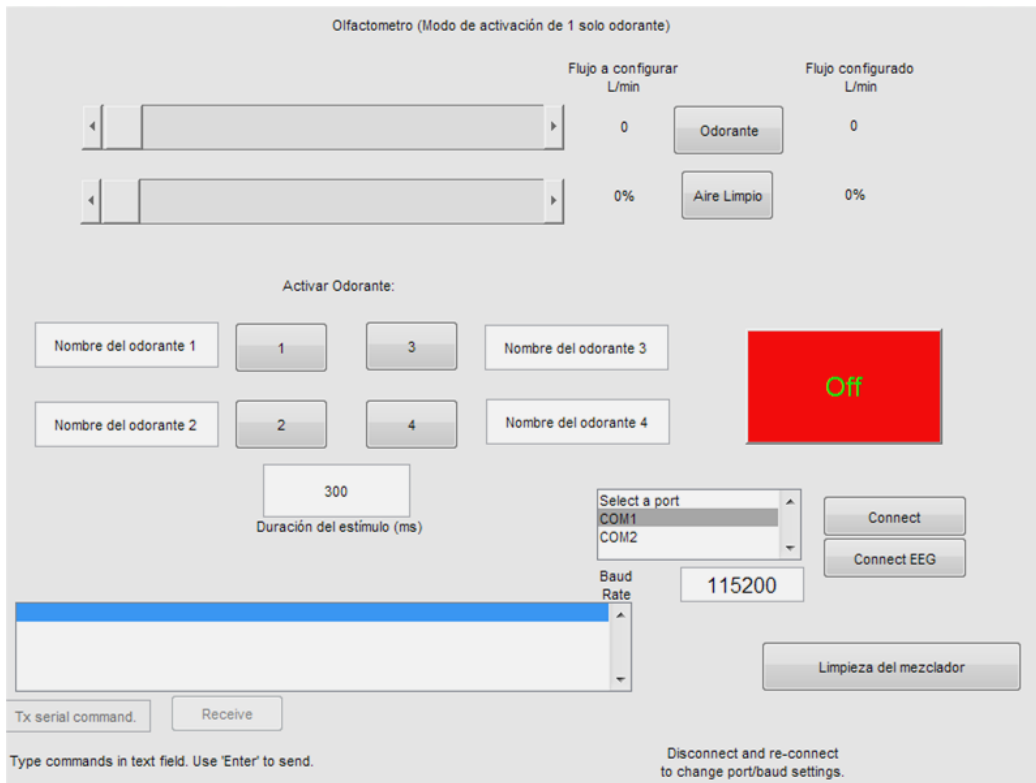


Fig. 13 Ventana de la interfaz de prueba para estímulos individuales.

La segunda permite ejecutar una secuencia automática de 4 estímulos la cual se repite el número de veces y en el orden especificado dejando pasar el tiempo que se desee entre cada estímulo. Los estímulos pueden ser disparados por la señal del sensor de respiración al detectar un flanco de subida (inhalación) en el umbral que se seleccione mediante una barra analógica y guarda el registro del mismo. La ventana desplegada en MATLAB se muestra en la Fig. 14.



Fig. 14 Ventana de la interfaz de prueba para ciclos automáticos de estímulos.

3.5 Adquisición de la señal de EEG

Con la finalidad de facilitar la adquisición de la señal de EEG se utilizó el sistema de adquisición EPOC de Emotiv® que permite la captura simultánea de 14 canales del sistema 10-20 (AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4) además de contar con un sistema de medición de impedancias en cada electrodo para asegurar un registro confiable.

La aplicación utilizada para el registro es el TestBench™ incluido en SDK del sistema, el cual permite recibir por medio de un puerto serie marcadores para integrarlos junto al registro de EEG.

3.6 Protocolo de Prueba

Para verificar el funcionamiento integral del sistema se llevó a cabo el registro de EEG. Teniendo en cuenta que el registro puede ser contaminado por las señales de potenciales evocados auditivos al escucharse el sonido de activación de las bombas y electroválvulas; se utilizaron un par de auriculares para reproducir una pista con ruido blanco a un volumen

en el cual el sujeto de prueba deja de percibir el sonido de activación de los dispositivos. En cuanto a los odorantes se decidió utilizar 3 olores agradables y fácilmente identificables: un olor cítrico (limón), uno frutal (fresa) y uno especiado (canela). En el cuarto canal se colocó agua destilada para registrar un blanco durante la prueba y facilitar el posterior procesamiento de las señales. Las muestras utilizadas consistieron en 1 ml de esencia (DEIMAN, Aditivo para alimentos) de cada olor mencionado y 99 ml de agua destilada.

Como ya se mencionó se utilizó como sensor auxiliar un sensor de temperatura (LM35) en la máscara de oxígeno utilizada para la administración de olores con la finalidad de liberar los olores justo en la inhalación, por lo que se hace el registro de este sensor en paralelo al EEG.

El protocolo que se llevó a cabo fue:

1. Hidratación de las felpas del sistema EPOC de Emotiv.
2. Colocación y verificación de calidad de contacto.
3. Colocación de audífonos con ruido blanco.
4. Colocación de la Máscara.
5. Programación de flujos, tiempos de estimulación y etiquetas de odorantes.
6. Configuración de marcadores y registro en TestBench.
7. Inicio de Prueba.

El ciclo de prueba programado fue: Flujo de aire olorizado 0.9 L/min, flujo de aire limpio 0.6 L/min, tiempo de estimulación 5s, tiempo mínimo interestímulo 10 s y número de ciclos 4. El orden de presentación de los olores agua, canela, fresa y limón; repitiéndose este ciclo hasta terminar el número especificado

Capítulo 4

Resultados

En primera instancia se tiene como resultado el prototipo de olfatómetro con todas sus etapas neumáticas ensambladas y circuitos electrónicos en PCB el cual puede observarse en la Fig. 15, a la salida del sistema se obtuvo un intervalo de trabajo de 0.6 a 1.5 l/min para aire olorizado y de 0.6 a 3 l/min para aire limpio. El intercambio de muestras y limpieza del sistema resultaron sencillos como se planteo en el diseño y no tomo más de 15 minutos incluyendo la preparación de las muestras olfativas.

Posteriormente se describen los resultados parciales obtenidos durante el desarrollo del proyecto: módulos desarrollados, componentes, pruebas de desempeño y software.

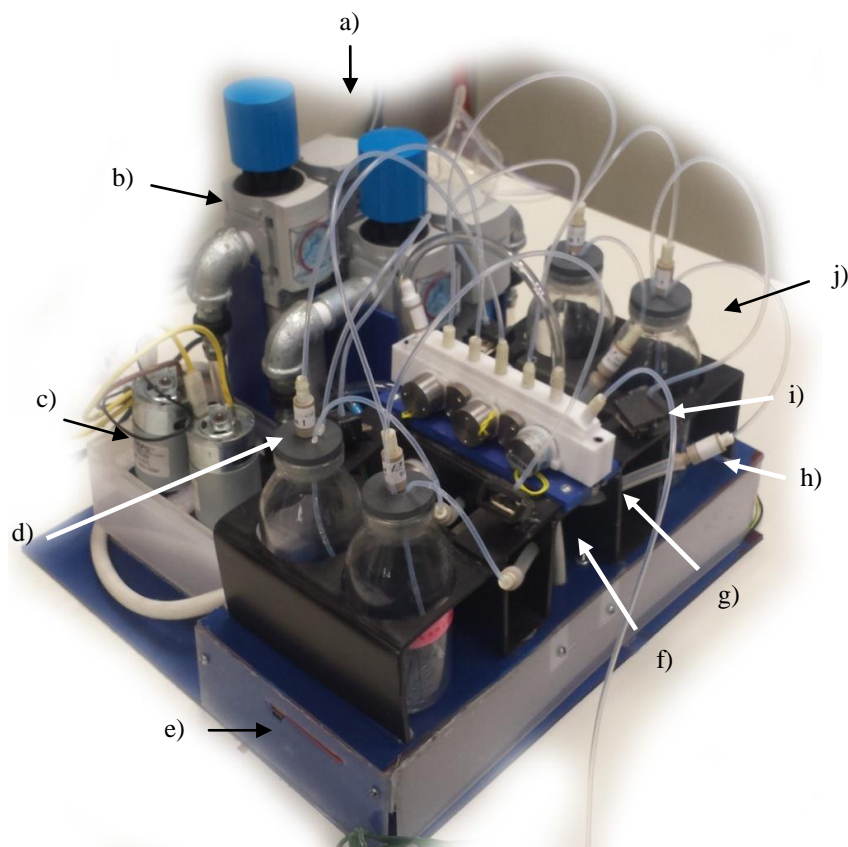


Fig. 15 Olfatómetro ensamblado y ubicación de componentes (a) Filtros de carbón activado b) Filtros de partículas y regulador c)Bombas de Aire d)Válvulas check de salida e)Puerto USB f)Electroválvulas de selección g)Mezclador h)Válvulas check de entrada i)Sensores de presión j)Contenedores de odorantes).

4.1 Módulos de potencia

Respecto a los módulos de potencia se fabricaron los circuitos impresos (PCB) cuyo diseño puede observarse en el Anexo B y se colocaron los componentes para realizar las pruebas de rendimiento y resistencia, las cuales consistieron en conectar los dispositivos a controlar (Bombas y electroválvulas) y activarlos durante un periodo de 12 horas para verificar que no exista un calentamiento excesivo de los módulos o dispositivos. La fuente de alimentación utilizada consta de un adaptador de corriente alterna con salida de 12 V a 4 A. La prueba fue satisfactoria al obtenerse un calentamiento de 15 °C por encima de la temperatura ambiente (23 °C) para los MOSFET al finalizar la prueba aunque se pudo observar que las válvulas de ingreso a los contenedores presentaron una temperatura pico de 70 °C aumentando el consumo total en 5%. Considerando que en funcionamiento normal nunca permanecerán activados los dispositivos por un periodo de tiempo tan prolongado se aprobó el diseño.

4.2 Sensores de presión

Respecto a los sensores de presión se diseñó un PCB para poder conectarlos a la tarjeta principal, se soldaron los sensores y posteriormente se encapsularon con pegamento epóxico para evitar daño mecánico en las pistas y el sensor al conectar las tuberías, se colocó una delgada capa de silicón para formar un sello entre el cilindro metálico del sensor y la tubería de PTFE con diámetro de 1mm termoformada para ajustarse al sensor en un extremo. Además del diseño de una pequeña pieza para asegurar los sensores a las tuberías y evitar fugas o desconexiones involuntarias por parte de los operadores al realizar el intercambio de muestras. El diseño del PCB puede observarse en el Anexo B y el encapsulado y aseguramiento de la tubería en la Fig. 16.

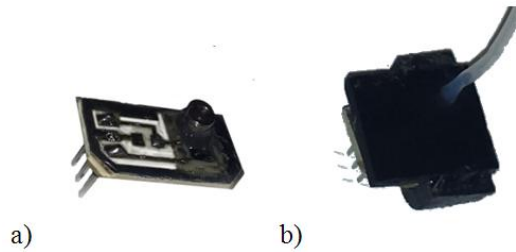


Fig. 16 a) PCB con sensor de presión encapsulado b) Ensamble de la tubería con el PCB

4.3 Tarjeta de interconexión

Se diseñó una tarjeta para conectar todos los módulos con cualquier tarjeta Launchpad con puerto de 40 pines para ser utilizada en el control; dando la opción de intercambiar el microcontrolador posteriormente de ser necesario. La tarjeta permite seleccionar la alimentación de los sensores auxiliares entre 5 V proporcionados por la tarjeta o una fuente externa y entrega la referencia de 2.048 V al ADC por medio del integrado LM4140. Todos los conectores a dispositivos son headers hembra con separación de 2.54 mm y la selección de fuente se hace mediante jumpers. La distribución final puede verse en la Fig. 17 y los PCB en el anexo B.

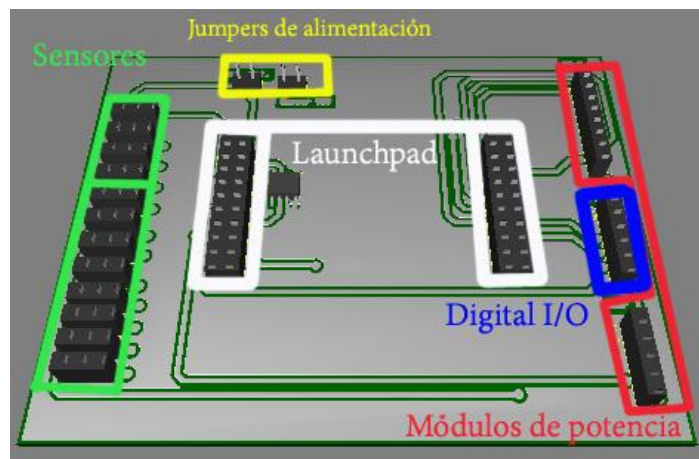


Fig. 17 Distribución final de la tarjeta de interconexión.

4.4 Diseño de carcasa

Siendo uno de los objetivos la portabilidad del sistema se diseñó una carcasa compacta en la cual se pudieran hacer las conexiones eléctricas y neumáticas de todos los

dispositivos, tomando en cuenta que al tratarse de un diseño modular el reemplazo de cada sección pudiera hacerse de forma sencilla y pudieran manipularse los componentes de forma cómoda. Teniendo como prioridad el fácil intercambio de muestras. La carcasa del prototipo mostrado en la Fig. 15 se encuentra compuesta de acrílico y PVC espumado por su facilidad de manejo. La propuesta de diseño para la carcasa final y dimensiones de las piezas que la componen pueden revisarse en el Anexo C y el ensamble se muestran en la Fig. 18. Se sugiere que la carcasa final se fabrique de Polimetilmetacrilato(PLEXIGLAS®) por su alta resistencia y facilidad de manejo.

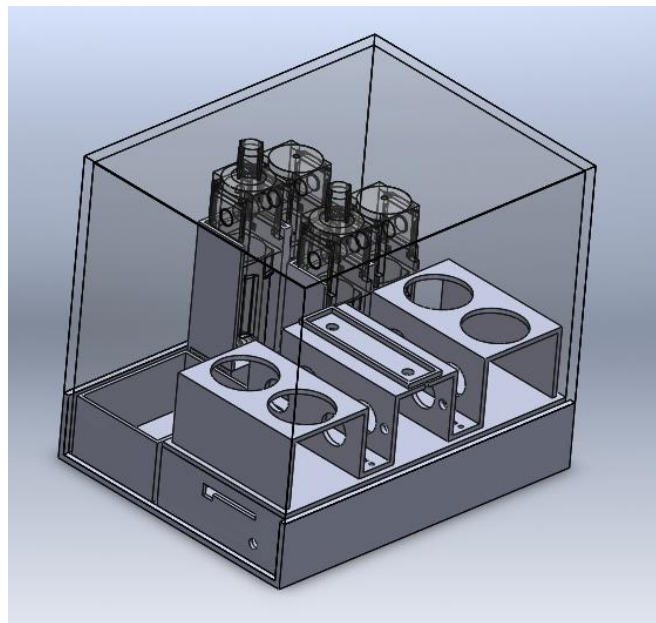


Fig. 18 Ensamble de la carcasa del dispositivo.

4.5 Caracterización del canal de aire limpio

Posteriormente haciendo uso de un canal de PWM como entrada de control de una bomba de aire y alimentación de 12 V se hizo un barrido desde 0% de ciclo de trabajo haciendo incrementos de 1% hasta lograr arrancar la bomba para determinar el porcentaje mínimo que se utilizará al ejecutar la rutina de control, el resultado fue de 6% del ciclo de trabajo. Una vez establecido el ciclo de trabajo mínimo se procedió a caracterizar el canal de aire limpio con ayuda de un sensor de flujo de aire (Honeywell, AWM5104VN) haciendo un barrido desde 10% del ciclo de trabajo hasta 100% con variaciones de 5%.

Interpolando por partes con un polinomio de 4° orden se obtiene la relación ciclo de trabajo-Flujo de aire limpio que se observa en la Fig. 19.

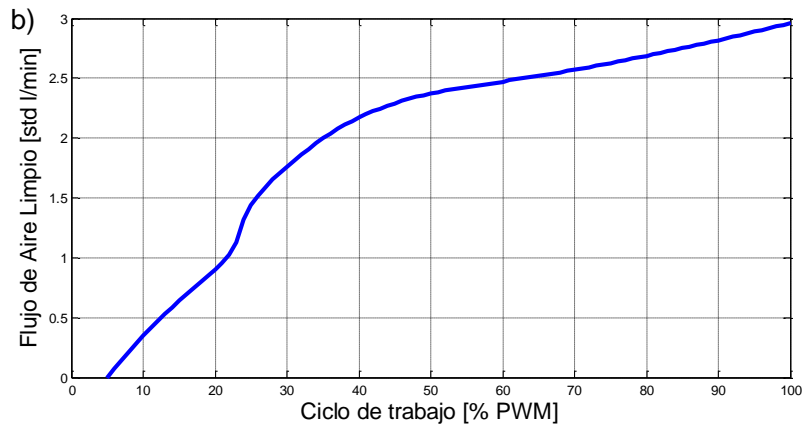


Fig. 19 Curva de caracterización de flujo para distintos ciclos de trabajo en la bomba de aire.

4.6 Evaluación del control de la corriente de aire olorizado

Para evaluar el flujo de aire olorizado se procedió a utilizar el sistema ensamblado en su totalidad y probar el sistema a lo largo del rango de operación de los sensores de presión (0 a 0.4 bar) para evaluar el comportamiento de la regla de control y cómo se explicó el sistema funciona regulando el flujo de salida de acuerdo a la presión dentro de los contenedores con los odorantes resultando en un rango en flujo de 0.6 a 1.5 std l/min. Un ejemplo de la prueba es dado un flujo objetivo de 1 std l/min se obtuvo: Flujo medio de 1.01 std l/min, Desviación estándar de 0.025 std l/min, Flujo pico de 1.06 std l/min y Flujo mínimo de 0.88 std l/min (Fig. 20). Estadísticamente tras el barrido del rango de trabajo se obtuvo un error en el flujo medio de $\pm 1\%$ con una desviación estándar de 3%.

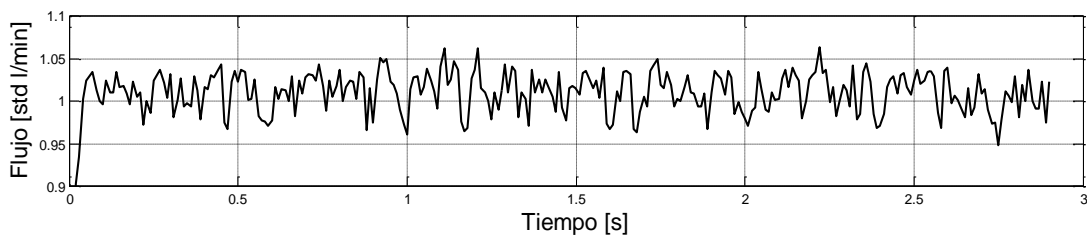


Fig. 20 Gráfica de la señal del flujo entregada por el sensor de presión para un flujo objetivo de 1 std l/min.

4.7 Interfaces Gráficas

Se introdujeron los datos obtenidos durante la caracterización del canal de aire limpio y la ecuación (1) con los datos del sistema para mostrar el flujo a configurar en cada canal al desplazar las barras analógicas, por otra parte se definieron los marcadores a enviar (Tabla 2) y la frecuencia de adquisición de los sensores auxiliares siendo esta 128 Hz para poder ser sincronizada de forma sencilla con los registros de EEG. Los datos de los sensores auxiliares junto con los datos de la prueba(etiquetas de odorantes, flujo de aire limpio, flujo de aire olorizado, tiempo de estimulación y tiempo interestímulo) son almacenados en un archivo .mat para posteriormente ser procesados.

Tabla 2. Marcadores enviados por la interfaz de Matlab mediante puerto serie a una aplicación de registro.

Marcador	Código(valor decimal)
Inicio de segmento de prueba	255
Inicio de estimulación	10
Fin de estimulación	11
Fin de segmento de prueba	128
Odorante	Se definen los marcadores deseados en el programa

Se verifico el funcionamiento de los comandos enviados por las interfaces, posteriormente durante las pruebas no se presentaron problemas por lo que el producto final son dos interfaces gráficas funcionales para el uso del olfatómetro.

4.8 Sensores auxiliares(Sensor de respiración)

Como ya se mencionó se agregó como sensor auxiliar un sensor de temperatura(LM35) dentro de la mascarilla de oxígeno como sensor auxiliar para determinar la inhalación del paciente y así utilizar esta señal como disparo para la liberación del olor seleccionado, la ubicación del sensor puede observarse en la Fig. 21. Durante las pruebas se pudo observar una variación cíclica de la temperatura que se relaciona con la inhalación y exhalación. Al

exhalar se observó un descenso de la temperatura el cual se detenía al comenzar la inhalación junto con un aumento de la temperatura del sensor, esto se atribuye a que al exhalar el flujo de aire es mayor provocando dicha disminución y que al inhalar dado que el flujo de aire es menor el calor corporal provoca un aumento en la temperatura del sensor.

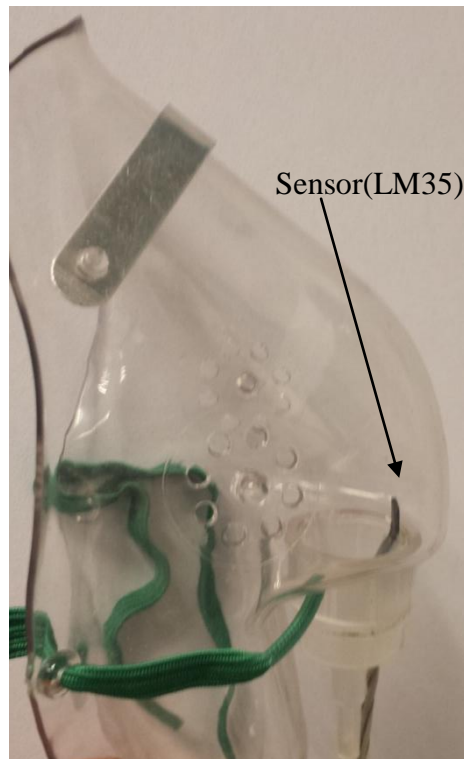


Fig. 21 Mascarilla de oxígeno con el sensor de temperatura colocado.

4.9 Registro de EEG y Segmentación

Se hicieron una serie de pruebas para verificar el funcionamiento del dispositivo en conjunto del sistema EPOC de Emotiv® para verificar su correcto funcionamiento. La primera prueba consistió en presentar un olor a un sujeto de prueba en distintas razones de dilución desde 1:1 hasta 1:5 (olorizado: limpio) siendo el flujo de aire olorizado 0.6 l/min para una solución de 1 ml de esencia de lima y 99 ml de agua haciendo uso de la interfaz para estimulaciones individuales. Como dispositivo de administración se utilizó una mascarilla y posteriormente una cánula de oxígeno desechables.

El registro de las señales de EEG se realizó sin problema alguno y se recibieron correctamente las cadenas de los marcadores con los datos de cada estimulación. El ajuste

en los flujos de aire olorizado y aire limpio permiten regular la intensidad del olor percibida por el sujeto de prueba y éstas serán definidas según las necesidades de la prueba a realizar.

Durante estas pruebas se observó un olor a plástico al utilizar la cánula y cierta interferencia al exhalar en el caso de utilizar la mascarilla; situación que puede afectar la percepción del paciente en los estudios. Por lo que se decidió proceder a las demás pruebas utilizando la mascarilla de oxígeno y el sensor de respiración para sincronizar la liberación del estímulo con la inhalación del sujeto de prueba.

En el segundo bloque de pruebas se aplicó el protocolo descrito en la sección 3.6 a 5 sujetos de prueba los cuales fueron interrogados al finalizar el registro para saber el número de olores percibidos; en todos los casos se identificaron 3 olores. Los registros de EEG generados por TestBench se encuentran en formato .edf y fueron importados a matlab para su segmentación y organización en una base de datos para su posterior análisis. La segmentación se hizo de forma automática utilizando los marcadores enviados por la interfaz, agrupando tramos del registro completo (14 canales de EEG y 22 registros complementarios) desde el inicio de la estimulación hasta el fin del tiempo interestímulo además de añadirse el registro de respiración a la base de datos. En la Fig. 22 se pueden observar un segmento de los registros obtenidos importados y segmentados automáticamente en matlab por medio de los marcadores generados.

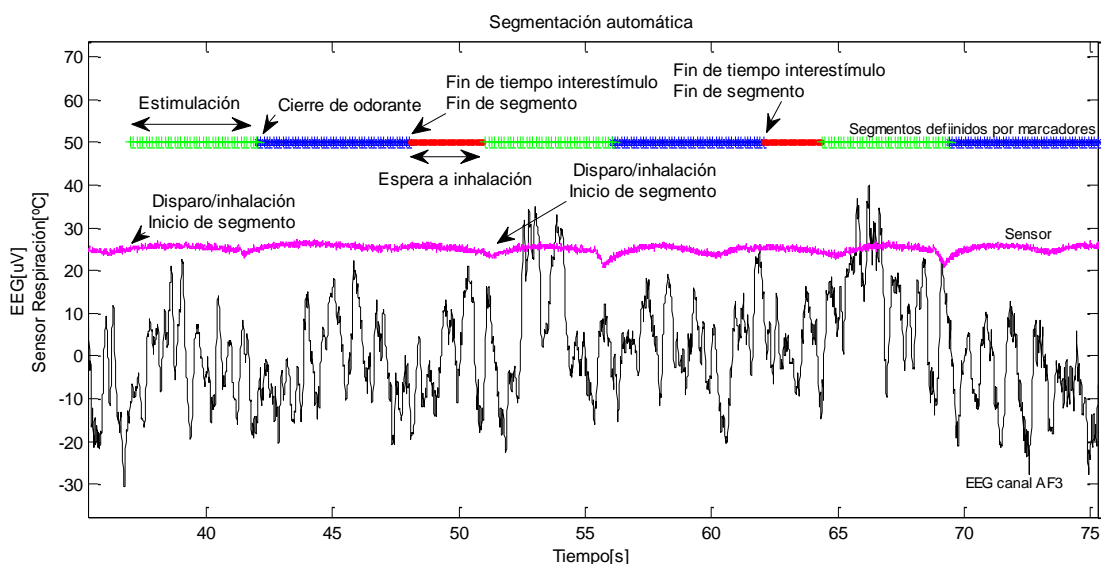


Fig. 22 Segmentación de los registros de EEG por medio de los marcadores enviados y la señal del sensor de respiración cómo disparador durante la inhalación.

Una vez obtenida la base de datos se procesó la información de manera preliminar para analizar si existen patrones visibles en las señales que pudieran diferenciar cuando el sujeto percibe un olor, tomando en cuenta que el área olfatoria del cerebro se encuentra ubicada en el centro se acotaron las pruebas a los canales FC5 y FC6 del sistema 10-20 de EEG. Para esto se muestra en la Fig. 23 los resultados de promediar 6 de los registros segmentados desde la activación del flujo olorizado hasta 5 segundos después de terminada la estimulación, tomando los 3 odorantes utilizados, para 5 de los sujetos de prueba. Se designaron etiquetas S1, S2, S3, S4 y S5 para los 5 sujetos de prueba.

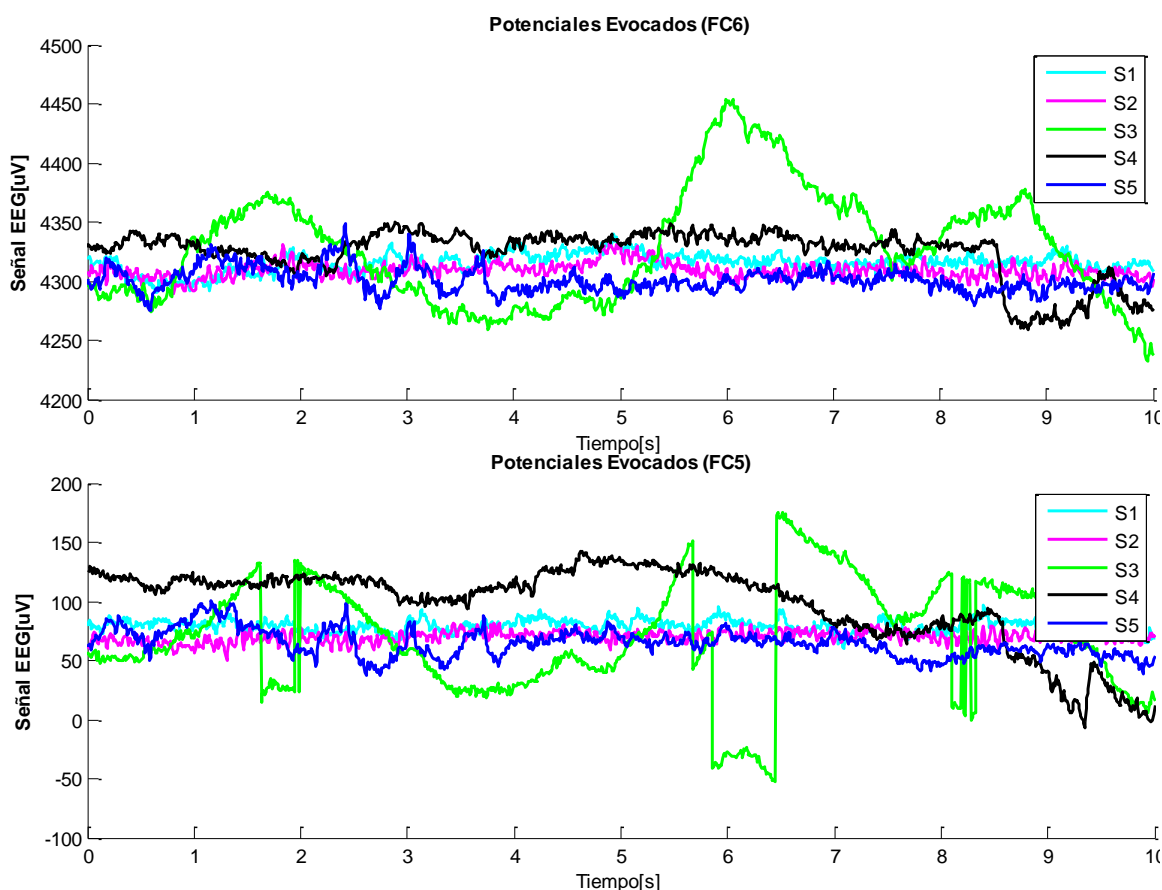


Fig. 23 Señales de potenciales evocados olfativos para los canales FC5 y FC6 promediando el registro de la estimulación con 3 odorantes (2 repeticiones) para 5 sujetos de prueba.

Mientras que en la Fig. 24 se aplicó el mismo procedimiento pero se promediaron los blancos (estimulación con agua bidestilada) de la prueba. A primera vista resalta que existen artefactos en el registro FC5 para S3, se piensa que estos se deben a un error durante la adquisición de la base de datos por parte de TestBench® ya que estos no se mostraron mientras se ejecutaba el registro si no hasta leer el archivo de la base de datos.

De igual forma se puede apreciar que los registros FC5 y FC6 son muy parecidos como era de esperarse al ser simétricos respecto al plano sagital. Respecto a la similitud de las señales entre sujetos no se puede precisar alguna característica común pero al comparar la tendencia entre los blancos y estimulaciones se puede observar que los registros con blanco tienden oscilar más.

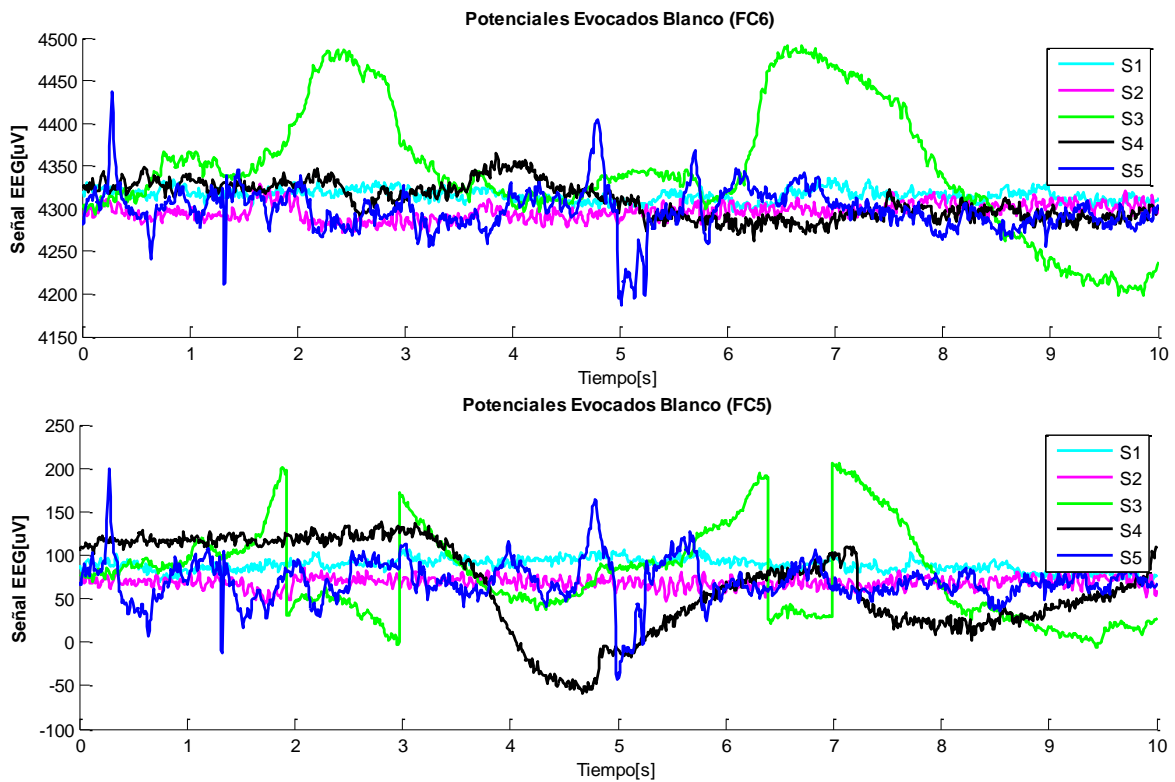


Fig. 24 Señales de potenciales evocados olfativos para los canales FC5 y FC6 promediando el registro de 4 repeticiones estimulando con un blanco (agua bidestilada) para 5 sujetos de prueba.

Otro punto de interés es el determinar si existe similitud entre los registros que corresponden a un mismo olor, ya sea para un mismo sujeto o entre sujetos. Por lo que se muestran las Fig. 25, Fig. 26 y Fig. 27 que corresponden al canal FC5 para estimulaciones con canela, fresa y limón respectivamente en 5 sujetos de prueba. En las gráficas se puede observar una gran similitud en la tendencia de las señales adquiridas para cada sujeto con excepción de S3 aunque la tendencia entre sujetos se ve alterada.

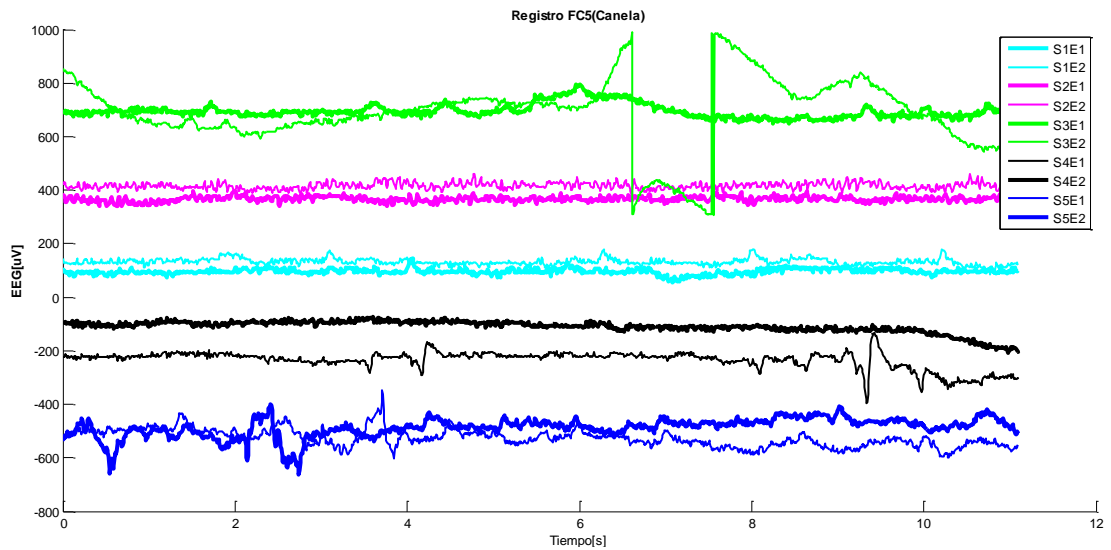


Fig. 25 Registros correspondientes a la estimulación con canela para 5 sujetos de prueba.

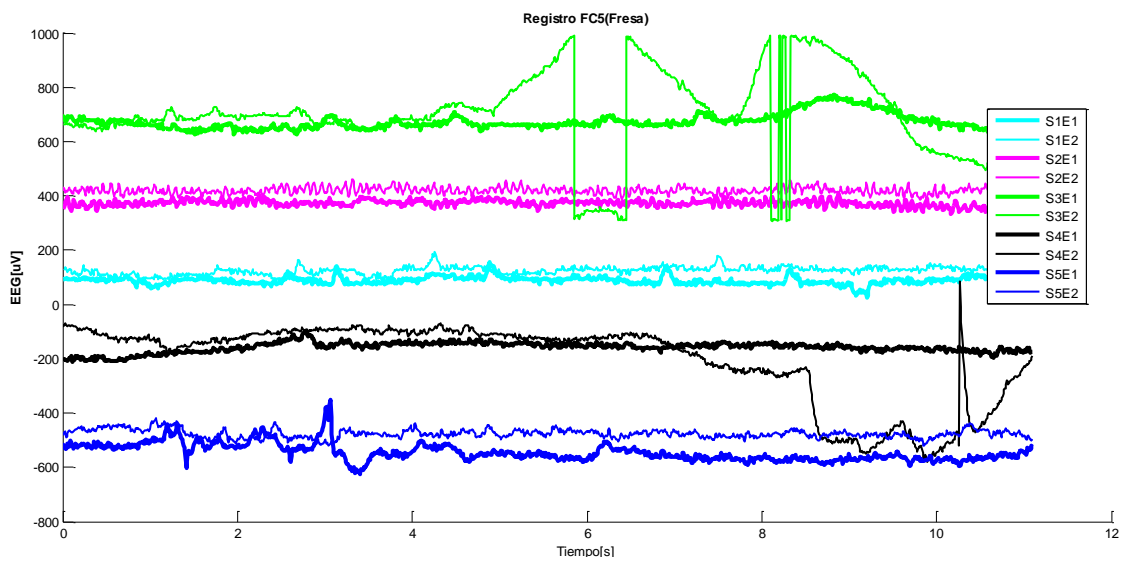


Fig. 26 Registros correspondientes a la estimulación con fresa para 5 sujetos de prueba.

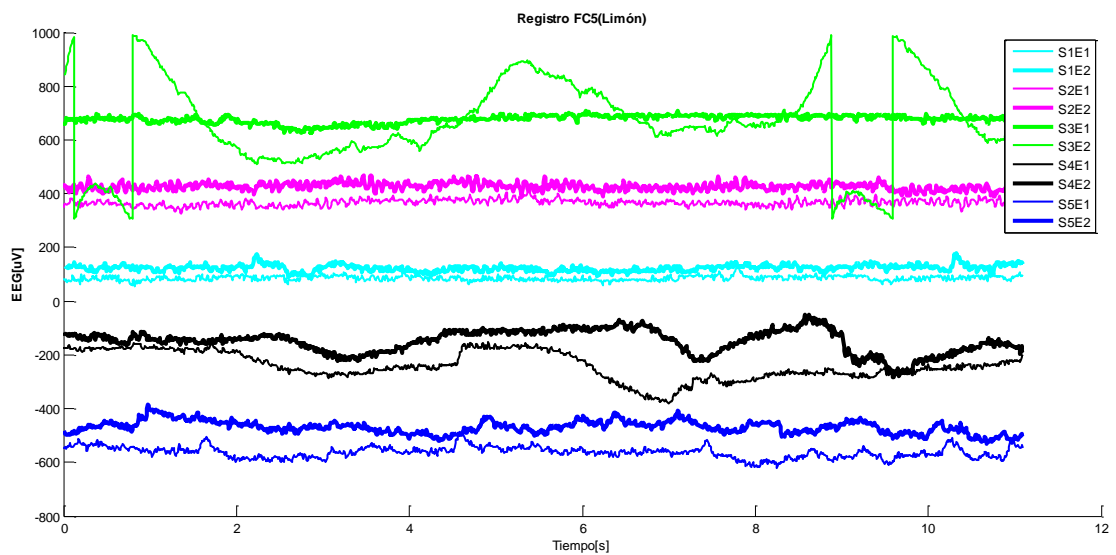


Fig. 27 Registros correspondientes a la estimulación con limón para 5 sujetos de prueba.

Capítulo 5

Discusión de Resultados

Podemos comparar las características obtenidas con las de los dispositivos comerciales y los reportados en la bibliografía como se observa en la tabla 3, enfocando especialmente los olfatómetros de Burghart que son diseñados para aplicaciones clínicas y tienen una buena aceptación en el mercado. En primera instancia se observa que el número de odorantes elegido cumple con los requerimientos mínimos estándar utilizados en la actualidad y se añade el beneficio de poder utilizar muestras sólidas como se plantea en [55] aunque se deja a un lado el uso de muestras gaseosas ya que su uso requiere de instrumental adicional para realizar el muestreo, pero en caso de ser requeridas posteriormente, bastaría con diseñar un módulo para su manejo que podría operar haciendo uso del mismo principio de control de flujo por presión al introducir de manera controlada la muestra gaseosa a los contenedores en lugar de aire.

Uno de los inconvenientes que puede apreciarse en los dispositivos comerciales son las restricciones que se tienen en cuanto a operar los dispositivos mediante su propio software y la incapacidad de conectar otros dispositivos de registro que no sean desarrollados por el mismo fabricante; lo cual limita el desarrollo de sistemas integrales de prueba. Situación que es solventada en el prototipo al plantear:

Su operación por comunicación serie estándar(USB/RS232), ya que los comandos pueden ser enviados por programas desarrollados en cualquier lenguaje de programación o incluso otros dispositivos (no necesariamente una PC) teniendo en mente el poder añadir un panel de control propio.

Incluirse el puerto de entrada/salida digital y las entradas analógicas auxiliares pues puede sincronizarse la activación de otros dispositivos para controlar el entorno durante las pruebas(iluminación, sonidos, indicadores, etc.) o adquirir señales suplementarias directamente (respiración, ECG, EEG, etc.).

Tabla 3 Comparativa de características de olfatómetros reportados con el diseño propuesto

Característica	Dispositivo diseñado	Sezille et al. [56]	Sommer et al. [55]	Serie TO (Odournet) [51]	OLo23 (BurgHart) [53]	OLo24 (Burghart) [53]
Rango de operación	0.6 a 1.5 l/min (olorizado) 0.6 a 3 l/min (limpio)	2 l/min	2 a 15 l/min	20 L/min (En conjunto)	-	8 L/min
Razón de dilución	1:5	1:9	1:3 (usando simultáneamente canales con H ₂ O)	1:4 a 1:65536	-	-
Muestras	Líquidas Sólidas	Líquidas	Líquidas Sólidas	Gaseosa	Líquidas	Gaseosas Líquidas
Número de canales	4	15	4	1	4 a 12	2 independientes
Fuente de aire	Propia	Externa	Externa	Externa	Externa	Propia
Tipo de control	Control de flujo por regulación de presión	3 Controladores de flujo de masa	Manual	Orificios calibrados	Controladores de flujo de masa	Controladores de flujo de masa
Tiempo de Respuesta	20 ms + retraso por recorrido	120 ms + retraso por recorrido	120 ms + retraso por recorrido		100 ms + retraso por recorrido	10 ms + retraso por recorrido
Dimensiones	30 x 25 x 30 cm	53 x 40 x 18 cm	42 x 11 x 23 cm	65 x 65 x 47	170x 70x 170 cm	
Registros externos	10 sensores auxiliares Envío de marcadores(puerto serie)	Botón de percepción (National Instruments multiple function board) Señales por Labview	-	-	1 disparador	1 disparador
Peso	4 kg	12 kg	3kg		400 kg	90 kg
Software	Programable en cualquier lenguaje	Labview	GNU C++	Propio	Propio	Propio
Error	1% error promedio 3% desviación estándar	±5 ml/min		10% desviación estándar	-	-
Costo	2500 USD	12500 USD	-	-	>100,000 USD	>100,000 USD

En cuanto al desempeño de los dispositivos puede observarse que el error de salida no es reportado en los dispositivos de Burghart siendo el error esperado menor de 1% al utilizar controladores de flujo de masa, mientras que en los olfatómetros de la serie TO de Odournet se reporta una desviación estándar en el flujo de salida de 10% (el error se reporta como desviación estándar ya que es la obtenida durante la fabricación de los orificios calibrados usados para regular el flujo) visto esto, se puede considerar como un buen resultado el obtenido; siendo el error promedio de 1% y desviación estándar de 3%.

Es posible mencionar que la regla de control utilizada (control proporcional) muestra un desempeño aceptable aunque puede mejorarse al desempeño del sistema estableciendo un control PID o incluso control neurodifuso. De igual forma, los rangos de la salida son aceptables para las pruebas que se pretende hacer, ya que no requieren de niveles mayores de dilución ni flujos altos pues flujos elevados pueden llegar a producir sensación táctil al sujeto de prueba, provocando interferencia en los registros por ejemplo en el de EEG.

Respecto a controlar la intensidad de los olores haciendo una dilución del flujo olorizado con aire limpio, la razón de dilución máxima del prototipo es de 1:5 la cual puede considerarse baja en comparación con dispositivos de Odournet. Sin embargo, puede llegar a suplirse esta deficiencia al controlar la concentración de las soluciones que se introducen a los recipientes o la cantidad de muestra sólida en ellos.

Existen dos formas que pueden ser utilizadas para ampliar el rango de operación del sistema y por lo tanto la capacidad máxima de dilución sin alterar su arquitectura, la más sencilla es el cambiar las válvulas check utilizadas por otras de mejores características (mayor C_v y menor presión de apertura). La segunda es intercambiar las bombas de aire y aumentar el rango de presión de los sensores de presión. Esto requeriría de un rediseño de los contenedores puesto que el aumento de presión puede provocar que la tapa sea expulsada o en el peor de los casos el contenedor explote como consecuencia de la elevada presión.

Otra posible mejora al prototipo es el agregar un lazo de control al canal de aire limpio haciendo uso de un sensor de presión para monitorear la presión antes de la válvula check de salida y aplicando un control similar al del canal de aire olorizado.

Durante las pruebas se pudo observar que el diseño de los contenedores fue el adecuado, ya que permite una fácil limpieza e intercambio de los mismos; siendo este tiempo no mayor a 10 minutos, al ser contenedores de bajo costo es posible agilizar las pruebas al tener una alta cantidad de muestras ya preparadas en contenedores distintos, para que una vez finalizado un bloque de pruebas se ejecute el procedimiento de limpieza (que se describe más adelante) y se intercambien las muestras.

En cuanto a la contaminación de las tuberías por olores sólo se presentaron problemas en el espacio muerto que va de la válvula check que sale del contenedor de odorantes hacia el mezclador, este problema se soluciono implementando un procedimiento de limpieza que consiste en poner agua bidestilada en contenedores limpios y hacer funcionar el sistema con flujo máximo durante 1 minuto para cada canal. Este procedimiento sólo es necesario si se cambian los odorantes a utilizar. Otra posible solución es el minimizar el espacio muerto al poner la válvula check directamente en el mezclador y fabricar una nueva pieza para conectar la entrada de la válvula con el contenedor.

No se pudieron obtener datos relevantes respecto al mantenimiento del sistema por lo que se sugiere hacer un calendario de revisión y calibración del sistema. Con el fin de mantenerlo en estado óptimo y poder determinar el tiempo promedio para la sustitución de sus componentes, aunque por la naturaleza de los materiales empleados se espera que los ciclos de limpieza descritos en el párrafo anterior sean suficientes. Sólo sería necesario el intercambiar los cartuchos de los filtros de partículas y carbón activado cómo se sugiere por el fabricante aproximadamente cada 1000 horas de funcionamiento.

Otra de las principales ventajas del diseño propuesto sobre los demás dispositivos se encuentran en la portabilidad, ya que no se necesita ningún tipo de instalación especial, basta con tener una PC y una toma de corriente de 127 V o una laptop y una batería de 12 V en caso de requerirse un sistema totalmente portable. Además de presentar dimensiones

y peso mucho más bajo que la mayoría de los dispositivos con excepción del descrito en [55] que es de la mitad de tamaño sin embargo, se debe considerar que este no cuenta con una fuente de aire propio ni la etapa de tratamiento del mismo.

La sección de administración al no estar fija presenta la ventaja de poder conectarse cualquier dispositivo con tubería mayor a 1mm sin afectar el flujo de salida, cuenta con un puerto roscado de 1/4" al cual se conecta un adaptador para el diámetro de la tubería deseada. Permitiendo ajustarse a distintos sujetos de prueba a diferencia de los dispositivos comerciales que utilizan consumibles propios o embudos fijos que no pueden sustituirse.

Un posible problema que se detectó en el sistema son las pequeñas fugas de aire que se pueden llegar a presentar al ensamblar el dispositivo, encontrándose la mayor parte en la sección de filtrado por lo que no afecta en gran medida al sistema, pero para su óptimo desempeño deben sellarse o encontrar una forma más segura de ensamblar los componentes pues las piezas utilizadas no están especialmente diseñadas para utilizar la tubería elegida (Tygon®).

La adquisición de los registros EEG se realizó sin problema alguno y al enviarse los marcadores a la aplicación de registro puede formarse de manera automática una base de datos de las pruebas realizadas que puede ser complementada por los registros de los sensores auxiliares como se mostró con el de respiración, cosa que no prevé ninguno de los sistemas reportados. Se espera que la segmentación automática haga posible implementar técnicas de procesamiento avanzadas como redes neuronales, análisis wavelet u otras para la detección de patrones y así poder llegar a plantear métodos de diagnóstico rápido o evaluación del sentido del olfato. Para facilitar el procesamiento de la base de datos puede y debe optimizarse el protocolo de pruebas por ejemplo controlando otras variables ambientales como la iluminación o regular el grado de concentración del sujeto de prueba al introducir una actividad a realizar durante la prueba, aumentando la repetibilidad en los registros.

También se debe considerar la eliminación de los artefactos en el registro en caso de presentarse o el utilizar un sistema de registro más confiable.

La arquitectura propuesta presenta la ventaja de poder implementar un mezclador de olores que entregue a la salida mezclas ponderadas de los 4 odorantes disponibles, con sólo modificar la programación a diferencia de las arquitecturas presentadas por [55] y [56]. Para esto se debe mantener activada la bomba de aire para mantener la presión fija antes de la entrada a los contenedores gracias al regulador de presión y controlar por PWM las válvulas de entrada a los contenedores, la regla de control a implementar debe determinarse modelando el sistema.

Finalmente, una característica que hay que resaltar es que ninguno de los dispositivos reportados utiliza un control de flujo por presión, que al implementarse disminuye considerablemente el costo y tamaño, aunque lo hace propenso a alterar su desempeño en caso de llegar a dañarse la válvula check caracterizada u obstruirse la tubería de salida. Dicho riesgo puede ser disminuido haciendo un buen uso del sistema y procurando no exceder los niveles de muestra establecidos y no introducir muestras que puedan desprender alguna partícula lo suficientemente grande para obstruir el sistema por ejemplo polvos.

Capítulo 6

Conclusiones y Perspectivas

6.1 Conclusiones

La reducción de costos y dimensiones de un sistema puede llegar a realizarse al cambiar los principios de operación en los que se basa y no necesariamente esperar a que la tecnología empleada en su fabricación sea de menor tamaño o menor costo.

El diseño de instrumental especializado con fines de investigación debe de plantearse buscando la forma de facilitar la interacción con otros sistemas, para que sea posible la construcción de plataformas integrales de prueba cada vez más completas.

Se consideran cumplidos los objetivos de la presente tesis al obtenerse un prototipo de olfactómetro activo, modular y totalmente funcional, que cumple con las características de ser portable, tener cuatro vías y contar con las interfaces que permiten su configuración y correcta comunicación con la aplicación de registro de EEG. Además de añadir la capacidad para obtener de forma automatizada una base de datos de las pruebas realizadas para su posterior análisis.

6.2 Trabajo a futuro

Analizado el estado del arte y observado el desempeño del diseño propuesto, se puede establecer un amplio panorama de aplicaciones para los dispositivos como el diseñado que van desde la aplicación directa en olfactometría a la integración de los mismos en sistemas más complejos. También se pueden establecer las posibles mejoras al sistema, a continuación se enlista el trabajo que se piensa se puede realizar:

- Optimización de las reglas de control e implementación de un lazo de control en el canal de aire limpio.
- Optimización del protocolo de pruebas.

- Agregar el modo de mezclador de olores al dispositivo.
- Desarrollar módulos de adquisición de señales suplementarias como EEG para hacer adquisición directa de las señales por medio del dispositivo.
- Desarrollar una nariz electrónica para obtener datos como la concentración de ciertos componentes a la salida del sistema y poder caracterizar las muestras que se utilicen.
- Diseñar un sistema integral de caracterización de muestras alimentarias, integrando una nariz y una lengua electrónica al dispositivo.
- Hacer uso del sistema para obtener bases de datos de muestras alimentarias para su clasificación o evaluación por medio de una nariz electrónica.
- Hacer uso del dispositivo para llevar a cabo estudios de comportamiento en insectos o pequeños mamíferos.
- Los productos obtenidos como módulos de potencia y tarjeta madre del dispositivo pueden ser empleados como base para el diseño de otro tipo de instrumentación o manipulación de otro tipo de fluidos.
- Obtener una base de datos extensa para distintos olores en un mayor número de sujetos de prueba con la finalidad de detectar posibles patrones.
- Hacer estudios clínicos para determinar si es posible relacionar los registros obtenidos usando el dispositivo con posibles estados patológicos en humanos.

Referencias:

- [1] K. Scott, "Chapter 23 - Chemical Senses: Taste and Olfaction," L. R. S. B. E. B. du L. G. C. B. T.-F. N. (Fourth E. Spitzer, Ed. San Diego: Academic Press, 2013, pp. 513–530.
- [2] B. Auffarth, "Understanding smell--the olfactory stimulus problem.," *Neurosci. Biobehav. Rev.*, vol. 37, no. 8, pp. 1667–79, Sep. 2013.
- [3] J. Feher, "4.6 - The Chemical Senses," J. B. T.-Q. H. P. Feher, Ed. Boston: Academic Press, 2012, pp. 359–369.
- [4] R. Farbiszewski and R. Kranc, "Olfactory receptors and the mechanism of odor perception," *Polish Ann. Med.*, vol. 20, no. 1, pp. 51–55, Sep. 2013.
- [5] S. Sankaran, L. R. Khot, and S. Panigrahi, "Biology and applications of olfactory sensing system: A review," *Sensors Actuators B Chem.*, vol. 171–172, pp. 1–17, Aug. 2012.
- [6] M. Wachowiak, "All in a sniff: olfaction as a model for active sensing.," *Neuron*, vol. 71, no. 6, pp. 962–73, Sep. 2011.
- [7] M. Barresi, R. Ciurleo, S. Giacoppo, V. Foti Cuzzola, D. Celi, P. Bramanti, and S. Marino, "Evaluation of olfactory dysfunction in neurodegenerative diseases.," *J. Neurol. Sci.*, vol. 323, no. 1–2, pp. 16–24, Dec. 2012.
- [8] A. Leer, M. a M. Smeets, P. J. Bulting, and M. a van den Hout, "Odors eliciting fear: a conditioning approach to Idiopathic Environmental Intolerances.," *J. Behav. Ther. Exp. Psychiatry*, vol. 42, no. 2, pp. 240–9, Jun. 2011.
- [9] A. Mukherjee, N. Sarkar, and A. Barik, "Long-chain free fatty acids from *Momordica cochinchinensis* leaves as attractants to its insect pest, *Aulacophora foveicollis* Lucas (Coleoptera: Chrysomelidae).," *J. Asia. Pac. Entomol.*, vol. 17, no. 3, pp. 229–234, Sep. 2014.
- [10] L. . Collins, M. . Wakefield, J. Chambers, and P. . Cox, "Progress towards a multi-species lure: comparison of behavioural bioassay methods for multi-species attractants against three pests of stored grain," *J. Stored Prod. Res.*, vol. 40, no. 3, pp. 341–353, Jan. 2004.
- [11] K. Persaud and G. Dodd, "Analysis of discrimination mechanisms in the mammalian olfactory system using a model nose," *Nature*, vol. 299, no. 5881, pp. 352–355, Sep. 1982.
- [12] Merriam-Webster, "Olfactometer," *Merriam-Webster.com*, 2014. [Online]. Available: <http://www.merriam-webster.com/dictionary/olfactomete>.
- [13] M. Miranda and C. Perez, "¿Porqué evaluar el olfato? y ¿Cómo evaluarlo?," *Rev. Med. Clin. Condes*, vol. 17, no. 3, pp. 120–122, 2006.
- [14] S. C. S. Inc., "A Review of The Science and Technology of Odor Measurement," Lake Elmo, MN, 2005.
- [15] A. C. Guyton, M. D, J. E. Hall, and D. Ph, *Fisiología médica*, 11^a ed. 2006.
- [16] M. A. Amerine, R. M. Pangborn, and E. B. Roessler, "CHAPTER 3 - Olfaction," M. A. Amerine, R. M. Pangborn, and E. B. B. T.-P. of S. E. of F. Roessler, Eds. Academic Press, 1965, pp. 145–219.

- [17] S. I. Korsching, “Odor maps in the brain: Spatial aspects of odor representation in sensory surface and olfactory bulb,” *Cell. Mol. Life Sci. C.*, vol. 58, no. 4, pp. 520–530, 2001.
- [18] D. A. Wilson and R. J. Stevenson, “Olfactory perceptual learning: the critical role of memory in odor discrimination,” *Neurosci. Biobehav. Rev.*, vol. 27, no. 4, pp. 307–328, Jun. 2003.
- [19] M. L. Fletcher and M. Bendahmane, “Chapter 4 - Visualizing Olfactory Learning Functional Imaging of Experience-Induced Olfactory Bulb Changes,” in *Odor Memory and Perception*, vol. Volume 208, E. B. and D. A. W. B. T.-P. in B. Research, Ed. Elsevier, 2014, pp. 89–113.
- [20] J. M. Bekkers and N. Suzuki, “Neurons and circuits for odor processing in the piriform cortex.,” *Trends Neurosci.*, vol. 36, no. 7, pp. 429–38, Jul. 2013.
- [21] J.-C. Pernollet, G. Sanz, and L. Briand, “Olfactory receptors and odour coding.,” *C. R. Biol.*, vol. 329, no. 9, pp. 679–90, Sep. 2006.
- [22] Y. Hasin-Brumshtein, D. Lancet, and T. Olender, “Human olfaction: from genomic variation to phenotypic diversity.,” *Trends Genet.*, vol. 25, no. 4, pp. 178–84, Apr. 2009.
- [23] C. H. Hawkes and B. C. Shephard, “Selective anosmia in Parkinson’s disease?,” *Lancet*, vol. 341, no. 8842, pp. 435–436, Feb. 1993.
- [24] J. a Nicell, “Assessment and regulation of odour impacts,” *Atmos. Environ.*, vol. 43, no. 1, pp. 196–206, Jan. 2009.
- [25] H. K. Patel, *The Electronic Nose: Artificial Olfaction Technology*. New Delhi: Springer India, 2014.
- [26] N. Hudson and G. a Ayoko, “Odour sampling 1: Physical chemistry considerations.,” *Bioresour. Technol.*, vol. 99, no. 10, pp. 3982–92, Jul. 2008.
- [27] N. Hudson and G. a Ayoko, “Odour sampling. 2. Comparison of physical and aerodynamic characteristics of sampling devices: a review.,” *Bioresour. Technol.*, vol. 99, no. 10, pp. 3993–4007, Jul. 2008.
- [28] C. M. Philpott, a Bennett, and G. E. Murty, “A brief history of olfaction and olfactometry.,” *J. Laryngol. Otol.*, vol. 122, no. 7, pp. 657–62, Jul. 2008.
- [29] P. X. Étiévant, “Pathologies et troubles de l’odorat : quels outils en pratique ?,” *Nutr. Clin. Métabolisme*, vol. 22, no. 3, pp. 113–120, Sep. 2008.
- [30] J. C. Martínez, J. M. M. Puebla, and J. A. J. Antolín, “Patología de la olfacción. olfactometría. manejo de los problemas olfativos,” in *Libro virtual de formación en ORL*, pp. 1–14.
- [31] M. A. Mcginley and C. M. Mcginley, “The New European Olfactometry Standard : Implementation, Experience, and Perspectives Authored by :,” Stillwater, MN, 2001.
- [32] Comité Europeen de Normalisation, “Air quality–Determination of odour concentration by dynamic olfactometry (EN13725).” 2003.
- [33] L. D. Falcão, G. de Revel, J. P. Rosier, and M. T. Bordignon-Luiz, “Aroma impact components of Brazilian Cabernet Sauvignon wines using detection frequency analysis (GC–olfactometry),” *Food Chem.*, vol. 107, no. 1, pp. 497–505, Mar. 2008.
- [34] B. d’Acampora Zellner, P. Dugo, G. Dugo, and L. Mondello, “Gas chromatography-olfactometry in food flavour analysis.,” *J. Chromatogr. A*, vol. 1186, no. 1–2, pp. 123–43, Apr. 2008.

- [35] K. a. Theron, M. Muller, M. van der Rijst, J. C. Cronje, M. le Roux, and E. Joubert, "Sensory profiling of honeybush tea (*Cyclopia* species) and the development of a honeybush sensory wheel," *Food Res. Int.*, vol. 66, pp. 12–22, Dec. 2014.
- [36] E. Bakchine, M. H. Pham-Delegue, L. Kaiser, and C. Masson, "Computer analysis of the exploratory behavior of insects and mites in an olfactometer.," *Physiol. Behav.*, vol. 48, no. 1, pp. 183–7, Jul. 1990.
- [37] C. Belda and J. Riudavets, "Attraction of the parasitoid *Anisopteromalus calandrae* (Howard) (Hymenoptera: Pteromalidae) to odors from grain and stored product pests in a Y-tube olfactometer.," *Biol. Control*, vol. 54, no. 1, pp. 29–34, Jul. 2010.
- [38] H. Pan, Y. Lu, and K. a. G. Wyckhuys, "Repellency of dimethyl disulfide to *Apolygus lucorum* (Meyer-Dür) (Hemiptera: Miridae) under laboratory and field conditions," *Crop Prot.*, vol. 50, pp. 40–45, Aug. 2013.
- [39] P. K. Løtvedt, S. K. Murali, L. T. Hernandez Salazar, and M. Laska, "Olfactory sensitivity for 'green odors' (aliphatic C(6) alcohols and C(6) aldehydes)--a comparative study in male CD-1 mice (*Mus musculus*) and female spider monkeys (*Ateles geoffroyi*).," *Pharmacol. Biochem. Behav.*, vol. 101, no. 3, pp. 450–7, May 2012.
- [40] K. M. Roddick, H. M. Schellinck, and R. E. Brown, "Olfactory delayed matching to sample performance in mice: sex differences in the 5XFAD mouse model of Alzheimer's disease.," *Behav. Brain Res.*, vol. 270, pp. 165–70, Aug. 2014.
- [41] C. G. Gross, *Encyclopedia of Neuroscience*. Elsevier, 2009.
- [42] T. Hummel, B. Sekinger, S. R. Wolf, E. Pauli, G. Kobal, and C. Pharmacology, "' Sniffin ' Sticks ' : Olfactory Performance Assessed by the Combined Testing of Odor Identification, Odor Discrimination and Olfactory Threshold."
- [43] R. L. Doty, R. E. Frye, and U. Agrawal, "Internal consistency reliability of the fractionated and whole University of Pennsylvania Smell Identification Test," *Percept. Psychophys.*, vol. 45, no. 5, pp. 381–384, Sep. 1989.
- [44] R. L. Doty, P. Shaman, and M. Dann, "Development of the University of Pennsylvania Smell Identification Test : A Standardized Microencapsulated Test of Olfactory Function," *Physiol. Behav.*, vol. 32, pp. 489–502, 1984.
- [45] G. V. P. Reddy, "Plant volatiles mediate orientation and plant preference by the predator *Chrysoperla carnea* Stephens (Neuroptera : Chrysopidae)," vol. 25, pp. 49–55, 2002.
- [46] P. Adhikary, A. Mukherjee, and A. Barik, "Role of surface wax alkanes from *Lathyrus sativus* L. seeds for attraction of *Callosobruchus maculatus* (F.) (Coleoptera: Bruchidae)," *J. Stored Prod. Res.*, vol. 59, pp. 113–119, Oct. 2014.
- [47] M. a. Birkett, T. J. a. Bruce, and J. a. Pickett, "Repellent activity of *Nepeta grandiflora* and *Nepeta clarkei* (Lamiaceae) against the cereal aphid, *Sitobion avenae* (Homoptera: Aphididae)," *Phytochem. Lett.*, vol. 3, no. 3, pp. 139–142, Sep. 2010.
- [48] N. Desneux, H. Rafalimanana, and L. Kaiser, "Dose-response relationship in lethal and behavioural effects of different insecticides on the parasitic wasp *Aphidius ervi*," *Chemosphere*, vol. 54, no. 5, pp. 619–27, Feb. 2004.
- [49] M. Joly, B. Michel, B. Deputte, and J.-M. Verdier, "Odor discrimination assessment with an automated olfactometric method in a prosimian primate, *Microcebus murinus*," *Physiol. Behav.*, vol. 82, no. 2–3, pp. 325–9, Sep. 2004.

- [50] A. R. S. Inc., “ARS Inc. Bio-Assay Equipment Home,” 2005. [Online]. Available: <http://www.ars-fla.com/mainpages/Bio-Assay/Bio-Assay.html>. [Accessed: 04-Nov-2014].
- [51] Odournet, “Product catalogue,” 2014. [Online]. Available: http://www.odournet.com/fileadmin/uploads/product_catalogues/Odournet_PC_2014_DRAFT_20141031.pdf.
- [52] St. Croix Sensory, “Nasal Ranger,” 2014. [Online]. Available: <http://www.nasalranger.com/>. [Accessed: 04-Nov-2014].
- [53] Burghart, “Burghart-mt,” 2014. [Online]. Available: <http://www.burghart-mt.de/index.php?p1=produkte&p2=olfaktometrie>.
- [54] V. A. Benignus and J. D. Prah, “A computer-controlled vapor-dilution olfactometer,” *Behav. Res. Methods Instrum.*, vol. 12, no. 5, pp. 535–540, 1980.
- [55] J. U. Sommer, W. Maboshe, M. Griebe, C. Heiser, K. Hörmann, B. a Stuck, and T. Hummel, “A mobile olfactometer for fMRI-studies,” *J. Neurosci. Methods*, vol. 209, no. 1, pp. 189–94, Jul. 2012.
- [56] C. Sezille, B. Messaoudi, a Bertrand, P. Jousain, M. Thévenet, and M. Bensafi, “A portable experimental apparatus for human olfactory fMRI experiments,” *J. Neurosci. Methods*, vol. 218, no. 1, pp. 1–10, Aug. 2013.
- [57] J. N. Lundström, A. R. Gordon, E. C. Alden, S. Boesveldt, and J. Albrecht, “Methods for building an inexpensive computer-controlled olfactometer for temporally-precise experiments,” *Int. J. Psychophysiol.*, vol. 78, no. 2, pp. 179–89, Nov. 2010.
- [58] B. N. Johnson and N. Sobel, “Methods for building an olfactometer with known concentration outcomes,” *J. Neurosci. Methods*, vol. 160, no. 2, pp. 231–45, Mar. 2007.
- [59] R. Popp, M. Sommer, and J. Müller, “Olfactometry in fMRI studies : odor presentation using nasal continuous positive airway pressure,” pp. 171–176, 2004.
- [60] M. Vigouroux, B. Bertrand, V. Farget, J. Plailly, and J. P. Royet, “A stimulation method using odors suitable for PET and fMRI studies with recording of physiological and behavioral signals,” *J. Neurosci. Methods*, vol. 142, no. 1, pp. 35–44, Mar. 2005.
- [61] Swagelok, “Valve Sizing,” 2007. [Online]. Available: <http://www.swagelok.com/downloads/webcatalogs/EN/ms-06-84.pdf>. [Accessed: 05-Nov-2014].

Anexo A

(Programación)

Programación del microcontrolador TMS320F28027 (CodeComposer)

```
#####  
//  
// FILE:   olfatometro.c  
//  
// TITLE:  Programa de control del olfatometro  
//  
#####  
  
#include <stdio.h>  
#include <file.h>  
  
#include "DSP28x_Project.h"    // DSP28x Headerfile  
  
#include "f2802x_common/include/adc.h"  
#include "f2802x_common/include/clk.h"  
#include "f2802x_common/include/flash.h"  
#include "f2802x_common/include/gpio.h"  
#include "f2802x_common/include/pie.h"  
#include "f2802x_common/include/pll.h"  
#include "f2802x_common/include/sci.h"  
#include "f2802x_common/include/pwm.h"  
#include "f2802x_common/include/pwr.h"  
#include "f2802x_common/include/sci_io.h"  
#include "f2802x_common/include/timer.h"  
#include "f2802x_common/include/wdog.h"  
  
#define CONV_WAIT 1L  
// Subrutinas y funciones  
void HRPWM1_Config(int); //Declaración rutina de configuración  
void scia_xmit(int a);  
void apagar(void);  
void control_activado(int presion,int presionant,int pobjetivo);  
void control_desactivado(int presion,int pobjetivo);  
interrupt void cpu_timer0_isr(void);  
  
//Variables del programa  
unsigned long timer0IntCount;  
unsigned int pulsos, odorante, activado;  
int marcador= 1;  
volatile uint32_t Xint1Count;  
uint32_t LoopCount;  
Uint16 j, DutyFine, n,update, ciclo,T,A;  
Uint16 presion0, presion1,presion2,presion3,portadora;  
Uint16 presion0ant, presion1ant,presion2ant,presion3ant;  
Uint32 i;  
int16 difpresion;  
Uint16 pobjetivo, apagado;  
Uint16 regresp;  
int16 respiracion;  
PWM_Handle myPwm1, myPwm2;  
int16_t referenceTemp;  
int16_t Serialdata;  
  
// Declarar y renombrar los handles a utilizar  
ADC_Handle myAdc;  
CLK_Handle myClk;  
FLASH_Handle myFlash;  
GPIO_Handle myGpio;  
PIE_Handle myPie;  
SCI_Handle mySci;  
PWR_Handle myPwr;  
TIMER_Handle myTimer0;  
  
//sub-rutina de interrupción para el timer 0  
interrupt void cpu_timer0_isr(void)
```

```

{
    //timer0IntCount++; //acumulador numero de conversiones
    ADC_forceConversion(myAdc, ADC_SocNumber_4); //conversión del sensor de temperatura

    //espera el fin de la conversión
    while(ADC_getIntStatus(myAdc, ADC_IntNumber_5) == 0) {
    }

        // Clear ADCINT5
        ADC_clearIntFlag(myAdc, ADC_IntNumber_5);
    //divide entre 4 la temperatura para enviar el dato por el USART en una trama de 8 bits
    //considerando la temperatura pueda oscilar entre 0 y 50 °C entregando una resolución de
    0.2°C/bit//agrega un offset para diferenciar entre la adquisición de señal y registro en espera
    pues se envía //continuamente el registro
        respiracion =(5+ ADC_readResult(myAdc, ADC_ResultNumber_4)/4);

        // restricción de valores para enviar sólo 8 bits
        if(respiracion>255)
            respiracion=255;
        if(respiracion<5)
            respiracion=5;
        if(regresp==1){
            scia_xmit(respiracion);
        }else {scia_xmit(0);
        }

        // Acknowledge this interrupt to receive more interrupts from group 1
        // PieCtrlRegs.PIEACK.all = PIEACK_GROUP1;
    PIE_clearInt(myPie, PIE_GroupNumber_1);
}

void apagar()
{
    PWM_setCmpA(myPwm1, 0); /// apaga la bomba de aire 1 canal olorizado
    //cierra las valvulas del mezclador
    GPIO_setHigh(myGpio, GPIO_Number_2);
    GPIO_setHigh(myGpio, GPIO_Number_3);
    GPIO_setHigh(myGpio, GPIO_Number_4);
    GPIO_setHigh(myGpio, GPIO_Number_5);
    GPIO_setHigh(myGpio, GPIO_Number_19);
    //cierra las valvulas de entrada a contenedores
    GPIO_setHigh(myGpio, GPIO_Number_12);
    GPIO_setHigh(myGpio, GPIO_Number_6);
    GPIO_setHigh(myGpio, GPIO_Number_7);
    GPIO_setHigh(myGpio, GPIO_Number_18);
    //borra las banderas de activación y selección de odorantes
    activado=0;
    odorante=0;
    // levanta la bandera de sistema apagado
    apagado=1;
}

//Configuración del SCI (puerto serie)
// SCIA 8-bit word, baud rate 115.2kbaud , default, 1 STOP bit, no parity
void scia_init()
{
    CLK_enableSciaClock(myClk);

    // 1 stop bit, No loopback
    // No parity,8 char bits,
    // async mode, idle-line protocol
    SCI_disableParity(mySci);
    SCI_setNumStopBits(mySci, SCI_NumStopBits_One);
    SCI_setCharLength(mySci, SCI_CharLength_8_Bits);

    SCI_enableTx(mySci);
    SCI_enableRx(mySci);
    SCI_enableTxInt(mySci);
    SCI_enableRxInt(mySci);

    // SCI BRR = LSPCLK/(SCI BAUDx8) - 1
    // Configured for 115.2kbps
    #if (CPU_FRQ_60MHZ)
        SCI_setBaudRate(mySci, SCI_BaudRate_115_2_kBaud);
    #elif (CPU_FRQ_50MHZ)

```

```

    SCI_setBaudRate(mySci, (SCI_BaudRate_e)13);
#elif (CPU_FRQ_40MHZ)
    SCI_setBaudRate(mySci, (SCI_BaudRate_e)10);
#endif
//habilita interrupciones de recepción y transmisión
    SCI_enableFifoEnh(mySci);
    SCI_resetTxFifo(mySci);
    SCI_clearTxFifoInt(mySci);
    SCI_resetChannels(mySci);
    SCI_setTxFifoIntLevel(mySci, SCI_FifoLevel_Empty);

    SCI_resetRxFifo(mySci);
    SCI_clearRxFifoInt(mySci);
    SCI_setRxFifoIntLevel(mySci, SCI_FifoLevel_4_Words);

    SCI_setPriority(mySci, SCI_Priority_FreeRun);

    SCI_enable(mySci);

    return;
}

void main()
{
    volatile int status = 0;
    volatile FILE *fid;

    CPU_Handle myCpu;
    PLL_Handle myPll;
    WDOG_Handle myWDog;

    // Inicialización de los handles necesarios
    myAdc = ADC_init((void *)ADC_BASE_ADDR, sizeof(ADC_Obj));
    myClk = CLK_init((void *)CLK_BASE_ADDR, sizeof(CLK_Obj));
    myCpu = CPU_init((void *)NULL, sizeof(CPU_Obj));
    myFlash = FLASH_init((void *)FLASH_BASE_ADDR, sizeof(FLASH_Obj));
    myGpio = GPIO_init((void *)GPIO_BASE_ADDR, sizeof(GPIO_Obj));
    myPie = PIE_init((void *)PIE_BASE_ADDR, sizeof(PIE_Obj));
    myPll = PLL_init((void *)PLL_BASE_ADDR, sizeof(PLL_Obj));
    myPwr = PWR_init((void *)PWR_BASE_ADDR, sizeof(PWR_Obj));
    mySci = SCI_init((void *)SCIA_BASE_ADDR, sizeof(SCI_Obj));
    myWDog = WDOG_init((void *)WDOG_BASE_ADDR, sizeof(WDOG_Obj));
    myPwm1 = PWM_init((void *)PWM_ePWM1_BASE_ADDR, sizeof(PWM_Obj));
    myPwm2 = PWM_init((void *)PWM_ePWM2_BASE_ADDR, sizeof(PWM_Obj));
    myTimer0 = TIMER_init((void *)TIMER0_BASE_ADDR, sizeof(TIMER_Obj));
//Carga de valores iniciales de banderas
    timer0IntCount = 0;
    pulsos=0;
    pobjetivo=0;
    odorante=0;
    portadora=0;

// Inicialización del sistema
    WDOG_disable(myWDog);
    CLK_enableAdcClock(myClk);
    (*Device_cal)();

    //Select the internal oscillator 1 as the clock source
    CLK_setOscSrc(myClk, CLK_OscSrc_Internal);

    // Setup the PLL for x10 /2 which will yield 50Mhz = 10Mhz * 10 / 2
    PLL_setup(myPll, PLL_Multiplier_12, PLL_DivideSelect_ClkIn_by_2);

    // Disable the PIE and all interrupts
    PIE_disable(myPie);
    PIE_disableAllInts(myPie);
    CPU_disableGlobalInts(myCpu);
    CPU_clearIntFlags(myCpu);

    // If running from flash copy RAM only functions to RAM
#ifdef _FLASH
    memcpy(&RamfuncsRunStart, &RamfuncsLoadStart, (size_t)&RamfuncsLoadSize);
#endif
}

```

```

// Inicializa el ADC
ADC_enableBandGap(myAdc);
ADC_enableRefBuffers(myAdc);
ADC_powerUp(myAdc);
ADC_enable(myAdc);
ADC_setVoltRefSrc(myAdc, ADC_VoltageRefSrc_Ext);

ADC_setSocChanNumber (myAdc, ADC_SocNumber_0, ADC_SocChanNumber_A6); //Set SOC0
channel select to ADCINA6
ADC_setSocChanNumber (myAdc, ADC_SocNumber_1, ADC_SocChanNumber_A1); //Set SOC1
channel select to ADCINA1
ADC_setSocChanNumber (myAdc, ADC_SocNumber_2, ADC_SocChanNumber_B4); //Set SOC1
channel select to ADCINB4
ADC_setSocChanNumber (myAdc, ADC_SocNumber_3, ADC_SocChanNumber_A2); //Set SOC1
channel select to ADCINA2
ADC_setSocChanNumber (myAdc, ADC_SocNumber_4, ADC_SocChanNumber_B2); //Set SOC1
channel select to ADCINB2

ADC_setSocSampleWindow(myAdc, ADC_SocNumber_0, ADC_SocSampleWindow_16_cycles); //Set
SOC0 acquisition period to 16 ADCCLK
ADC_setSocSampleWindow(myAdc, ADC_SocNumber_1, ADC_SocSampleWindow_16_cycles); //Set
SOC1 acquisition period to 16 ADCCLK
ADC_setSocSampleWindow(myAdc, ADC_SocNumber_2, ADC_SocSampleWindow_16_cycles); //Set
SOC2 acquisition period to 16 ADCCLK
ADC_setSocSampleWindow(myAdc, ADC_SocNumber_3, ADC_SocSampleWindow_16_cycles); //Set
SOC3 acquisition period to 16 ADCCLK
ADC_setSocSampleWindow(myAdc, ADC_SocNumber_4, ADC_SocSampleWindow_32_cycles); //Set
SOC4 acquisition period to 32 ADCCLK

ADC_setIntSrc(myAdc, ADC_IntNumber_1, ADC_IntSrc_EOC0);
//Connect ADCINT1 to EOC0
ADC_setIntSrc(myAdc, ADC_IntNumber_2, ADC_IntSrc_EOC1);
//Connect ADCINT2 to EOC1
ADC_setIntSrc(myAdc, ADC_IntNumber_3, ADC_IntSrc_EOC2);
//Connect ADCINT3 to EOC2
ADC_setIntSrc(myAdc, ADC_IntNumber_4, ADC_IntSrc_EOC3);
//Connect ADCINT4 to EOC3
ADC_setIntSrc(myAdc, ADC_IntNumber_5, ADC_IntSrc_EOC4);
//Connect ADCINT5 to EOC4

ADC_enableInt(myAdc, ADC_IntNumber_1);
//Enable ADCINT1
ADC_enableInt(myAdc, ADC_IntNumber_2);
//Enable ADCINT2
ADC_enableInt(myAdc, ADC_IntNumber_3);
//Enable ADCINT3
ADC_enableInt(myAdc, ADC_IntNumber_4);
//Enable ADCINT4
ADC_enableInt(myAdc, ADC_IntNumber_5);
//Enable ADCINT5

// Configura las salidas de EPwm1A, EPwm1B
GPIO_setPullUp(myGpio, GPIO_Number_0, GPIO_PuLLUp_Disable);
GPIO_setPullUp(myGpio, GPIO_Number_1, GPIO_PuLLUp_Disable);
GPIO_setMode(myGpio, GPIO_Number_0, GPIO_0_Mode_EPWM1A);
GPIO_setMode(myGpio, GPIO_Number_1, GPIO_1_Mode_EPWM1B);

//Inicialización para las valvulas
GPIO_setMode(myGpio, GPIO_Number_2, GPIO_0_Mode_GeneralPurpose);
GPIO_setMode(myGpio, GPIO_Number_3, GPIO_0_Mode_GeneralPurpose);
GPIO_setMode(myGpio, GPIO_Number_4, GPIO_0_Mode_GeneralPurpose);
GPIO_setMode(myGpio, GPIO_Number_5, GPIO_0_Mode_GeneralPurpose);
GPIO_setMode(myGpio, GPIO_Number_19, GPIO_0_Mode_GeneralPurpose);

GPIO_setMode(myGpio, GPIO_Number_12, GPIO_0_Mode_GeneralPurpose);
GPIO_setMode(myGpio, GPIO_Number_6, GPIO_0_Mode_GeneralPurpose);
GPIO_setMode(myGpio, GPIO_Number_7, GPIO_0_Mode_GeneralPurpose);
GPIO_setMode(myGpio, GPIO_Number_18, GPIO_0_Mode_GeneralPurpose);

GPIO_setDirection(myGpio, GPIO_Number_2, GPIO_Direction_Output);
GPIO_setDirection(myGpio, GPIO_Number_3, GPIO_Direction_Output);
GPIO_setDirection(myGpio, GPIO_Number_4, GPIO_Direction_Output);
GPIO_setDirection(myGpio, GPIO_Number_5, GPIO_Direction_Output);
GPIO_setDirection(myGpio, GPIO_Number_19, GPIO_Direction_Output);

```



```

GPIO_setDirection(myGpio, GPIO_Number_12, GPIO_Direction_Output);
GPIO_setDirection(myGpio, GPIO_Number_6, GPIO_Direction_Output);
GPIO_setDirection(myGpio, GPIO_Number_7, GPIO_Direction_Output);
GPIO_setDirection(myGpio, GPIO_Number_18, GPIO_Direction_Output);

// Inicializa el reloj

CLK_setClkOutPreScaler(myClk, CLK_ClkOutPreScaler_SysClkOut_by_1);

// Setup a debug vector table and enable the PIE
    PIE_setDebugIntVectorTable(myPie);
    PIE_enable(myPie);
// Timer interrupt
PIE_registerPieIntHandler(myPie,PIE_GroupNumber_1,PIE_SubGroupNumber_7,(intVec_t)&cpu_timer0_isr
);
    CPU_enableInt(myCpu, CPU_IntNumber_1);
// Enable Global Interrupts
    CPU_enableGlobalInts(myCpu);

// Inicializa SCIA
scia_init();

FLASH_setup(myFlash);

// Inicializa los pines de la comunicaci3n serie
GPIO_setPullUp(myGpio, GPIO_Number_28, GPIO_PullUp_Enable);
GPIO_setPullUp(myGpio, GPIO_Number_29, GPIO_PullUp_Disable);
GPIO_setQualification(myGpio, GPIO_Number_28, GPIO_Qual_ASync);
GPIO_setMode(myGpio, GPIO_Number_28, GPIO_Mode_SCIRXDA);
GPIO_setMode(myGpio, GPIO_Number_29, GPIO_Mode_SCITXDA);

// Configura la interrupci3n del timer para el envio periodico de los sensores auxiliares
TIMER_stop(myTimer0);
    #if (CPU_FRQ_60MHZ)

TIMER_setPeriod(myTimer0, 468750); //128Hz
#endif
TIMER_setPreScaler(myTimer0, 0);
TIMER_reload(myTimer0);
TIMER_setEmulationMode(myTimer0,TIMER_EmulationMode_StopAfterNextDecrement);
TIMER_enableInt(myTimer0);
TIMER_start(myTimer0);
CPU_enableInt(myCpu, CPU_IntNumber_1);
PIE_enableTimer0Int(myPie);
CPU_enableGlobalInts(myCpu);
CPU_enableDebugInt(myCpu);

//Redirecciona STDOUT to SCI para habilitar el envio de datos por el puesto serie
status = add_device("scia", SSA, SCI open, SCI close, SCI read, SCI write, SCI lseek,
SCI unlink, SCI rename);
fid = fopen("scia","w");
freopen("scia:", "w", stdout);
setvbuf(stdout, NULL, _IONBF, 0);

CLK_disableTbClockSync(myClk);
//Configura el estado inicial de los PWMs
T=62000;
HRPWM1_Config(T); // ePWM1 target, Period = 100Hz
presion0=0;
presion1=0;
presion2=0;
presion3=0;
regresp=0;
CLK_enableTbClockSync(myClk);
A=1000;
apagar();
PWM_setCmpB(myPwm1, 62000);
Main loop
for(;;) {
    // ejecuta el lazo de control mientras no reciba instrucciones
    while(SCI_getRxFifoStatus(mySci) < SCI_FifoStatus_1_Word){

```

```

if(odorante==1){
    GPIO_setLow(myGpio, GPIO_Number_12);
    GPIO_setHigh(myGpio, GPIO_Number_6);
    GPIO_setHigh(myGpio, GPIO_Number_7);
    GPIO_setHigh(myGpio, GPIO_Number_18);
    GPIO_setHigh(myGpio, GPIO_Number_19);
if(activado==1){
    ADC_forceConversion(myAdc, ADC_SocNumber_0);

    while(ADC_getIntStatus(myAdc, ADC_IntNumber_1) == 0) {
    }

    // Clear ADCINT1
    ADC_clearIntFlag(myAdc, ADC_IntNumber_1);

    // Get temp sensor sample result from SOC1
    presion0ant=presion0;
    presion0 = ADC_readResult(myAdc, ADC_ResultNumber_0);
    control_activado(presion0,presion0ant,pobjetivo);
} else{
    ADC_forceConversion(myAdc, ADC_SocNumber_0);
    while(ADC_getIntStatus(myAdc, ADC_IntNumber_1) == 0) {
    }
    // Clear ADCINT1
    ADC_clearIntFlag(myAdc, ADC_IntNumber_1);
    // Get temp sensor sample result from SOC1
    presion0ant=presion0;
    presion0 = ADC_readResult(myAdc, ADC_ResultNumber_0);
    control_desactivado(presion0,pobjetivo);
}
} else if(odorante==2){
    GPIO_setHigh(myGpio, GPIO_Number_12);
    GPIO_setLow(myGpio, GPIO_Number_6);
    GPIO_setHigh(myGpio, GPIO_Number_7);
    GPIO_setHigh(myGpio, GPIO_Number_18);
    GPIO_setHigh(myGpio, GPIO_Number_19); //Force start of conversion on SOC1
    if(activado==1){
        ADC_forceConversion(myAdc, ADC_SocNumber_1);

        //Wait for end of conversion.
        while(ADC_getIntStatus(myAdc, ADC_IntNumber_2) == 0) {
        }

        // Clear ADCINT1
        ADC_clearIntFlag(myAdc, ADC_IntNumber_2);

        // Get temp sensor sample result from SOC1
        presion1ant=presion1;
        presion1 = ADC_readResult(myAdc, ADC_ResultNumber_1);
        control_activado(presion1,presion1ant,pobjetivo);
    } else {
        ADC_forceConversion(myAdc, ADC_SocNumber_1);

        while(ADC_getIntStatus(myAdc, ADC_IntNumber_2) == 0) {
        }
        ADC_clearIntFlag(myAdc, ADC_IntNumber_2);
        presion1ant=presion1;
        presion1 = ADC_readResult(myAdc, ADC_ResultNumber_1);
        control_desactivado(presion1,pobjetivo);
    }
} else if(odorante==3){
    GPIO_setHigh(myGpio, GPIO_Number_12);
    GPIO_setHigh(myGpio, GPIO_Number_6);
    GPIO_setLow(myGpio, GPIO_Number_7);
    GPIO_setHigh(myGpio, GPIO_Number_18);
    GPIO_setHigh(myGpio, GPIO_Number_19);
    //Force start of conversion on SOC2
    if(activado==1){
        ADC_forceConversion(myAdc, ADC_SocNumber_2);

        //Wait for end of conversion.
        while(ADC_getIntStatus(myAdc, ADC_IntNumber_3) == 0) {
        }
    }
}

```

```

// Clear ADCINT1
ADC_clearIntFlag(myAdc, ADC_IntNumber_3);
// Get temp sensor sample result from SOC1
presion2ant=presion2;
presion2 = ADC_readResult(myAdc, ADC_ResultNumber_2);
control_activado(presion2,presion2ant,pobjetivo);

}else{
    ADC_forceConversion(myAdc, ADC_SocNumber_2);
    while(ADC_getIntStatus(myAdc, ADC_IntNumber_3) ==
0) {
        }
        ADC_clearIntFlag(myAdc, ADC_IntNumber_3);
        presion2ant=presion2;
        presion2 = ADC_readResult(myAdc, ADC_ResultNumber_2);
        control_desactivado(presion2,pobjetivo);
    }
} else if(odorante==4){
    GPIO_setHigh(myGpio, GPIO_Number_12);
    GPIO_setHigh(myGpio, GPIO_Number_6);
    GPIO_setHigh(myGpio, GPIO_Number_7);
    GPIO_setLow(myGpio, GPIO_Number_18);
    GPIO_setHigh(myGpio, GPIO_Number_19);

    if(activado==1){
        ADC_forceConversion(myAdc, ADC_SocNumber_3);
        while(ADC_getIntStatus(myAdc, ADC_IntNumber_4) == 0) {
        }
        ADC_clearIntFlag(myAdc, ADC_IntNumber_4);
        presion3ant=presion3;
        presion3 = ADC_readResult(myAdc, ADC_ResultNumber_3);
        control_activado(presion3,presion3ant,pobjetivo);
    } else{
        ADC_forceConversion(myAdc, ADC_SocNumber_3);
        while(ADC_getIntStatus(myAdc, ADC_IntNumber_4) == 0) {
        }
        ADC_clearIntFlag(myAdc, ADC_IntNumber_4);
        presion3ant=presion3;
        presion3 = ADC_readResult(myAdc, ADC_ResultNumber_3);
        control_desactivado(presion3,pobjetivo);
    }
} else if(odorante==5){
    //activa la purga del mezclador
    GPIO_setHigh(myGpio, GPIO_Number_3);
    GPIO_setHigh(myGpio, GPIO_Number_2);
    GPIO_setHigh(myGpio, GPIO_Number_4);
    GPIO_setHigh(myGpio, GPIO_Number_5);
    GPIO_setLow(myGpio, GPIO_Number_19);

    }else {
        apagar();
    }
}
//lectura de instrucciones
Serialdata = SCI_getData(mySci);
if(Serialdata == 0xFE){
    //espera configuración de bomba1
    while(SCI_getRxFifoStatus(mySci) < SCI_FifoStatus_1_Word){ }
    Serialdata = SCI_getData(mySci);
    if(Serialdata < 101){
        A=(Serialdata)*620;
        PWM_setCmpA(myPwm1, A);
    }
} else if(Serialdata == 0xFF){
    //espera configuración de bomba2
    while(SCI_getRxFifoStatus(mySci) < SCI_FifoStatus_1_Word){}
    Serialdata = SCI_getData(mySci);
    if(Serialdata < 101){
        A=62000-(Serialdata)*620;
        portadora=A;
        PWM_setCmpB(myPwm1, A);
    }
} else if(Serialdata == 0xFA){
    // espera flujo objetivo valores en 8 bits de 0 a 255
    while(SCI_getRxFifoStatus(mySci) < SCI_FifoStatus_1_Word){}
}

```

```

        pobjetivo = SCI_getData(mySci);
// escala los valores recibidos a valores del sensor para el loop de control y salida del sensor
    pobjetivo=pobjetivo*8;
    } else if(Serialdata == 0xFD){
// apaga la sección de aire olorizado
        apagar();
        activado=0;
        apagado=1;

    } else if(Serialdata == 0xF0){
// activa el odorante que se encuentre seleccionado
        activado=1;
        apagado=0;

        if(odorante==1){
            GPIO_setLow(myGpio, GPIO_Number_3);
            GPIO_setHigh(myGpio, GPIO_Number_2);
            GPIO_setHigh(myGpio, GPIO_Number_4);
            GPIO_setHigh(myGpio, GPIO_Number_5);
            GPIO_setHigh(myGpio, GPIO_Number_19);

        } else if(odorante==2){
            GPIO_setHigh(myGpio, GPIO_Number_3);
            GPIO_setLow(myGpio, GPIO_Number_4);
            GPIO_setHigh(myGpio, GPIO_Number_2);
            GPIO_setHigh(myGpio, GPIO_Number_5);
            GPIO_setHigh(myGpio, GPIO_Number_19);

        } else if(odorante==3){
            GPIO_setHigh(myGpio, GPIO_Number_3);
            GPIO_setHigh(myGpio, GPIO_Number_4);
            GPIO_setLow(myGpio, GPIO_Number_2);
            GPIO_setHigh(myGpio, GPIO_Number_5);
            GPIO_setHigh(myGpio, GPIO_Number_19);

        } else if(odorante==4){
            GPIO_setHigh(myGpio, GPIO_Number_3);
            GPIO_setHigh(myGpio, GPIO_Number_2);
            GPIO_setHigh(myGpio, GPIO_Number_4);
            GPIO_setLow(myGpio, GPIO_Number_5);
            GPIO_setHigh(myGpio, GPIO_Number_19);

        } else if(odorante==5){
            GPIO_setHigh(myGpio, GPIO_Number_3);
            GPIO_setHigh(myGpio, GPIO_Number_2);
            GPIO_setHigh(myGpio, GPIO_Number_4);
            GPIO_setHigh(myGpio, GPIO_Number_5);
            GPIO_setLow(myGpio, GPIO_Number_19);

        }
        else{
            apagado=0;
//introducir rutina de error de configuración
        }
//scia_xmit(odorante);

    } else if(Serialdata == 0xEF){
//espera odorante a seleccionar
        while(SCI_getRxFifoStatus(mySci) < SCI_FifoStatus_1_Word){
        }
        odorante = SCI_getData(mySci);
    }
    else if(Serialdata == 0xEE){
//espera configuración del registro respiratorio
        while(SCI_getRxFifoStatus(mySci) < SCI_FifoStatus_1_Word){
        }
        regresp = SCI_getData(mySci);
    }
}
}

void HRPWM1_Config(period)
{

```

```

CLK_enablePwmClock(myClk, PWM_Number_1);

// ePWM1 register configuration with HRPWM
// ePWM1A toggle low/high with MEP control on Rising edge
PWM_setPeriodLoad(myPwm1, PWM_PeriodLoad_Immediate);
PWM_setPeriod(myPwm1, period-1); // Set timer period
PWM_setCmpA(myPwm1, period / 2);
PWM_setCmpAHr(myPwm1, (1 << 8));
PWM_setCmpB(myPwm1, period / 2);
PWM_setPhase(myPwm1, 0x0000); // Phase is 0
PWM_setCount(myPwm1, 0x0000); // Clear counter

PWM_setCounterMode(myPwm1, PWM_CounterMode_Up); // Count up
PWM_disableCounterLoad(myPwm1); // Disable phase loading
PWM_setSyncMode(myPwm1, PWM_SyncMode_Disable);
PWM_setHighSpeedClkDiv(myPwm1, PWM_HspClkDiv_by_1); // Clock ratio to SYSCLKOUT
PWM_setClkDiv(myPwm1, PWM_ClkDiv_by_16);

PWM_setShadowMode_CmpA(myPwm1, PWM_ShadowMode_Shadow); // Load registers every ZERO
PWM_setShadowMode_CmpB(myPwm1, PWM_ShadowMode_Shadow);
PWM_setLoadMode_CmpA(myPwm1, PWM_LoadMode_Zero);
PWM_setLoadMode_CmpB(myPwm1, PWM_LoadMode_Zero);

PWM_setActionQual_Zero_PwmA(myPwm1, PWM_ActionQual_Clear);
PWM_setActionQual_CntUp_CmpA_PwmA(myPwm1, PWM_ActionQual_Set);
PWM_setActionQual_Zero_PwmB(myPwm1, PWM_ActionQual_Set);
PWM_setActionQual_CntUp_CmpB_PwmB(myPwm1, PWM_ActionQual_Clear);

//MEP control on Rising edge
PWM_setHrEdgeMode(myPwm1, PWM_HrEdgeMode_Falling);
PWM_setHrControlMode(myPwm1, PWM_HrControlMode_Duty);
PWM_setHrShadowMode(myPwm1, PWM_HrShadowMode_CTR_EQ_0);
}
// Transmit a character from the SCI
void scia_xmit(int a)
{
// while (SciaRegs.SCIFFTX.bit.TXFFST != 0) {}
while(SCI_getTxFifoStatus(mySci) != SCI_FifoStatus_Empty){
}
// SciaRegs.SCITXBUF=a;
SCI_putData(mySci, a);
}

void scia_msg(char * msg)
{
int i;
i = 0;
while(msg[i] != '\0')
{
scia_xmit(msg[i]);
i++;
}
}

void control_activado(int presion,int presionant,int pobjetivo){
Uint16 A;
int16 difpresion;
difpresion= pobjetivo-presion;
if (presionant<presion){
if(difpresion>0){
A=(15+pobjetivo/100+difpresion/(1024/pobjetivo));
if(A>100){
A=62000;
}else A=A*620;
PWM_setCmpA(myPwm1, A);
}else if(difpresion>-15){
A=(10+pobjetivo/100-difpresion/(1024/pobjetivo));
if(A>100){
A=62000;
}else A=A*620;
PWM_setCmpA(myPwm1, A);
}
}
}

```

```

else PWM_setCmpA(myPwm1, 0);
}else{
    if(difpresion>0){
        A=(15+pobjetivo/100+difpresion/(1024/pobjetivo));
        if(A>100){
            A=62000;
        }else A=A*620;
        PWM_setCmpA(myPwm1, A);
    } else if(difpresion>-10){
        A=(10+pobjetivo/100);
        if(A>100){
            A=62000;
        }else A=A*620;
        PWM_setCmpA(myPwm1, A);
    }
    else PWM_setCmpA(myPwm1, 0);
}
}

void control_desactivado(int presion,int pobjetivo){
    Uint16 A;
    int16 difpresion;
    difpresion= pobjetivo-presion;
    if(pobjetivo<180){
        PWM_setCmpA(myPwm1, 0);
    } else{
        if(difpresion>100){
            A=(5+difpresion/(2048/pobjetivo));
            if(A>100){
                A=62000*7/10;
            }else A=A*620*7/10;
            PWM_setCmpA(myPwm1, A);
        } else PWM_setCmpA(myPwm1, 0);
    }
}
//=====
// Fin.
//=====

```

Programación de la interfaz de pruebas para estímulos individuales(Matlab).

```
function varargout = Pruebas_olfactometro(varargin)
% Author: Luis Fernando Valdez Garduño
% Interfaz de estímulos individuales
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',      mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @Pruebas_olfactometro_OpeningFcn, ...
                  'gui_OutputFcn',  @Pruebas_olfactometro_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

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

% --- Executes just before Pruebas_olfactometro is made visible.
function Pruebas_olfactometro_OpeningFcn(hObject, eventdata, handles, varargin)
serialPorts = instrhwininfo('serial');
nPorts = length(serialPorts.SerialPorts);
set(handles.portList, 'String', ...
      [{'Select a port'} ; serialPorts.SerialPorts ]);
set(handles.portList, 'Value', 2);
set(handles.history_box, 'String', cell(1));

handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

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

% --- Outputs from this function are returned to the command line.
function varargout = Pruebas_olfactometro_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;

% --- Executes on selection change in portList.
function portList_Callback(hObject, eventdata, handles)
% hObject      handle to portList (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns portList contents as cell
array
%           contents{get(hObject,'Value')} returns selected item from portList

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

% Hint: listbox 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

function history_box_Callback(hObject, eventdata, handles)
% hObject      handle to history_box (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 history_box as text
%           str2double(get(hObject,'String')) returns contents of history_box as a double

% --- Executes during object creation, after setting all properties.
function history_box_CreateFcn(hObject, eventdata, handles)
% hObject      handle to history_box (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

function Tx_send_Callback(hObject, eventdata, handles)
TxText = get(handles.Tx_send, 'String');
fprintf(handles.serConn, TxText);

currList = get(handles.history_box, 'String');

set(handles.history_box, 'String', ...
    [currList ; ['Sent @ ' datestr(now) ': ' TxText] ]);
set(handles.history_box, 'Value', length(currList) + 1 );

set(hObject, 'String', '');

% --- Executes during object creation, after setting all properties.
function Tx_send_CreateFcn(hObject, eventdata, handles)
% hObject    handle to Tx_send (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 rxButton.
function rxButton_Callback(hObject, eventdata, handles)
try
    RxText = fscanf(handles.serConn);
    currList = get(handles.history_box, 'String');
    if length(RxText) < 1
        RxText = 'Timeout @ ';
        set(handles.history_box, 'String', ...
            [currList ; [RxText datestr(now)] ]);
    else
        set(handles.history_box, 'String', ...
            [currList ; ['Received @ ' datestr(now) ': ' RxText ] ]);
    end
    set(handles.history_box, 'Value', length(currList) + 1 );
catch e
    disp(e)
end

```

```

function baudRateText_Callback(hObject, eventdata, handles)
% hObject    handle to baudRateText (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 baudRateText as text
%         str2double(get(hObject,'String')) returns contents of baudRateText as a double

% --- Executes during object creation, after setting all properties.
function baudRateText_CreateFcn(hObject, eventdata, handles)
% hObject    handle to baudRateText (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 connectButton.
function connectButton_Callback(hObject, eventdata, handles)
if strcmp(get(hObject,'String'),'Connect') % currently disconnected
    serPortn = get(handles.portList, 'Value');
    if serPortn == 1
        errordlg('Select valid COM port');
    else
        serList = get(handles.portList, 'String');
        serPort = serList{serPortn};
        serConn = serial(serPort, 'TimeOut', 1, ...
            'BaudRate', str2num(get(handles.baudRateText, 'String')));

        try
            fopen(serConn);
            handles.serConn = serConn;

            % enable Tx text field and Rx button
            set(handles.Tx_send, 'Enable', 'On');
            set(handles.rxButton, 'Enable', 'On');

            set(hObject, 'String', 'Disconnect')
        catch e
            errordlg(e.message);
        end

    end
else

```

```

        set(handles.Tx_send, 'Enable', 'Off');
        set(handles.rxButton, 'Enable', 'Off');

        set(hObject, 'String','Connect')
        fclose(handles.serConn);
end
guidata(hObject, handles);

% --- 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)
if isfield(handles, 'serConn')
    fclose(handles.serConn);
    fclose(handles.serConn2)
end
% Hint: delete(hObject) closes the figure
delete(hObject);

% --- Executes on button press in od1.
function od1_Callback(hObject, eventdata, handles)
% hObject    handle to od1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
retardo=str2num(get(handles.tiempo, 'String'));
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 1);
fwrite(handles.serConn2, 255);
fprintf(handles.serConn2, get(handles.odorante3e, 'String'));
fprintf(handles.serConn2, get(handles.tiempo, 'String'));
fprintf(handles.serConn2, get(handles.text7, 'String'));
fprintf(handles.serConn2, get(handles.text8, 'String'));
pause(4)
fwrite(handles.serConn, 240);
fwrite(handles.serConn2, 1);
pause(retardo/1000)
fwrite(handles.serConn, 253);
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 5);
pause(1)
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 0);
fwrite(handles.serConn, 253);
pause(3)
fwrite(handles.serConn2, 254);

```

```

% --- Executes on button press in od2.
function od2_Callback(hObject, eventdata, handles)
% hObject    handle to od2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
retardo=str2num(get(handles.tiempo, 'String'));
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 2);
fwrite(handles.serConn2, 255);
fprintf(handles.serConn2, get(handles.odorante3e, 'String'));
fprintf(handles.serConn2, get(handles.tiempo, 'String'));
fprintf(handles.serConn2, get(handles.text7, 'String'));
fprintf(handles.serConn2, get(handles.text8, 'String'));
pause(4)
fwrite(handles.serConn, 240);
fwrite(handles.serConn2, 2);
pause(retardo/1000)
fwrite(handles.serConn, 253);
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 5);
pause(1)
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 0);
fwrite(handles.serConn, 253);
pause(3)
fwrite(handles.serConn2, 254);

% --- Executes on button press in od3.
function od3_Callback(hObject, eventdata, handles)
% hObject    handle to od3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
retardo=str2num(get(handles.tiempo, 'String'));
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 3);
fwrite(handles.serConn2, 255);
fprintf(handles.serConn2, get(handles.odorante3e, 'String'));
fprintf(handles.serConn2, get(handles.tiempo, 'String'));
fprintf(handles.serConn2, get(handles.text7, 'String'));
fprintf(handles.serConn2, get(handles.text8, 'String'));
pause(4)
fwrite(handles.serConn, 240);
fwrite(handles.serConn2, 3);
pause(retardo/1000)
fwrite(handles.serConn, 253);
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 5);
pause(1)
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 0);

```

```

fwrite(handles.serConn, 253);
pause(3)
fwrite(handles.serConn2, 254);

% --- Executes on button press in od4.
function od4_Callback(hObject, eventdata, handles)
% hObject    handle to od4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
retardo=str2num(get(handles.tiempo, 'String'));
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 4);
fwrite(handles.serConn2, 255);
fprintf(handles.serConn2, get(handles.odorante3e, 'String'));
fprintf(handles.serConn2, get(handles.tiempo, 'String'));
fprintf(handles.serConn2, get(handles.text7, 'String'));
fprintf(handles.serConn2, get(handles.text8, 'String'));
pause(4)
fwrite(handles.serConn, 240);
fwrite(handles.serConn2, 4);
pause(retardo/1000)
fwrite(handles.serConn, 253);
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 5);
pause(1)
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 0);
fwrite(handles.serConn, 253);
pause(3)
fwrite(handles.serConn2, 254);

% --- Executes on button press in od5.
function od5_Callback(hObject, eventdata, handles)
% hObject    handle to od5 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 5);

% --- Executes on button press in valv1.
function valv1_Callback(hObject, eventdata, handles)
% hObject    handle to valv1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
fwrite(handles.serConn, 253 );

% --- Executes on button press in valv2.
function valv2_Callback(hObject, eventdata, handles)

```

```

% hObject    handle to valv2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
valor1 = floor(get(handles.slider1,'Value'))
fwrite(handles.serConn, 255);
fwrite(handles.serConn, valor1);

% --- Executes on button press in bombal.
function bombal_Callback(hObject, eventdata, handles)
% hObject    handle to bombal (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
valor1 = floor(get(handles.slider1,'Value'))
if valor1>44
bit=4*valor1;
T1=293;
N2=6950;
gg=1;
p1=0.2+0.000385*bit;
cv=0.014;
dp=p1-0.2;
%figure;
q2=N2.*cv.*p1.*(1-(2.*dp)/(3.*p1)).*(dp/(p1.*gg.*T1)).^0.5;
else
    q2=0;
    valor1=0;
end
set(handles.text7, 'String',num2str(floor(q2*100)/100));
fwrite(handles.serConn, 250);
fwrite(handles.serConn, valor1);

% --- Executes on button press in bomba2.
function bomba2_Callback(hObject, eventdata, handles)
% hObject    handle to bomba2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
load('presionbom2.mat')
valor1 = floor(get(handles.slider2,'Value'))
T1=293;
N2=6950;
gg=1;
p1=0.2+pf(valor1);
cv=0.014;
dp=p1-0.2;
%figure;
q2=N2.*cv.*p1.*(1-(2.*dp)/(3.*p1)).*(dp/(p1.*gg.*T1)).^0.5;
set(handles.text8, 'String',num2str(floor(q2*100)/100));
fwrite(handles.serConn, 255);
fwrite(handles.serConn, valor1);

% --- Executes on button press in apagar.

```

```

function apagar_Callback(hObject, eventdata, handles)
% hObject    handle to apagar (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
fwrite(handles.serConn, 253);
fwrite(handles.serConn, 255);
fwrite(handles.serConn, 0);

% --- Executes on slider movement.
function slider1_Callback(hObject, eventdata, handles)
% hObject    handle to slider1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
h=get(hObject, 'Value');
if h>44
bit=4*floor(h)
T1=293;
N2=6950;
gg=1;
p1=0.2+0.000385*bit;
cv=0.014;
dp=p1-0.2;
%figure;
q2=N2.*cv.*p1.*(1-(2.*dp)./(3.*p1)).*(dp./(p1.*gg.*T1)).^0.5;
else
    q2=0;
end
%hold on
%plot(bit,q2,'r')
set(handles.text4, 'String',num2str(floor(q2*100)/100));

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of slider

% --- Executes during object creation, after setting all properties.
function slider1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

% --- Executes on slider movement.

```

```

function slider2_Callback(hObject, eventdata, handles)
% hObject    handle to slider2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
load('presionbom2.mat');
h=floor(get(hObject, 'Value'));
T1=293;
N2=6950;
gg=1;
p1=0.2+pf(h);
cv=0.014;
dp=p1-0.2;
%figure;
q2=N2.*cv.*p1.*(1-(2.*dp)./(3.*p1)).*(dp./(p1.*gg.*T1)).^0.5;
set(handles.text5, 'String', num2str(floor(q2*100)/100));

% Hints: get(hObject,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of slider

% --- Executes during object creation, after setting all properties.
function slider2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function tiempo_Callback(hObject, eventdata, handles)
% hObject    handle to tiempo (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 tiempo as text
%        str2double(get(hObject,'String')) returns contents of tiempo as a double

% --- Executes during object creation, after setting all properties.
function tiempo_CreateFcn(hObject, eventdata, handles)
% hObject    handle to tiempo (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 pushbutton14.
function pushbutton14_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton14 (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 connect2.
function connect2_Callback(hObject, eventdata, handles)
% hObject    handle to connect2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
if strcmp(get(hObject,'String'),'Connect EEG') % currently disconnected
    serPortn = get(handles.portList, 'Value');
    if serPortn == 1
        errordlg('Select valid COM port');
    else
        serList = get(handles.portList,'String');
        serPort = serList{serPortn};
        serConn2 = serial(serPort, 'TimeOut', 1, ...
            'BaudRate', str2num(get(handles.baudRateText, 'String')));

        try
            fopen(serConn2);
            handles.serConn2 = serConn2;

            set(hObject, 'String','Disconnect EEG')
        catch e
            errordlg(e.message);
        end

    end

else
    set(hObject, 'String','Connect EEG')
    fclose(handles.serConn2);
end
guidata(hObject, handles);

function odorantele_Callback(hObject, eventdata, handles)
% hObject    handle to odorantele (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 odorantele as text
%         str2double(get(hObject,'String')) returns contents of odorantele as a double

% --- Executes during object creation, after setting all properties.
function odorantele_CreateFcn(hObject, eventdata, handles)
% hObject    handle to odorantele (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

function odorante2e_Callback(hObject, eventdata, handles)
% hObject    handle to odorante2e (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 odorante2e as text
%         str2double(get(hObject,'String')) returns contents of odorante2e as a double

% --- Executes during object creation, after setting all properties.
function odorante2e_CreateFcn(hObject, eventdata, handles)
% hObject    handle to odorante2e (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

function odorante3e_Callback(hObject, eventdata, handles)
% hObject    handle to odorante3e (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 odorante3e as text
%         str2double(get(hObject,'String')) returns contents of odorante3e as a double

% --- Executes during object creation, after setting all properties.
function odorante3e_CreateFcn(hObject, eventdata, handles)
% hObject    handle to odorante3e (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

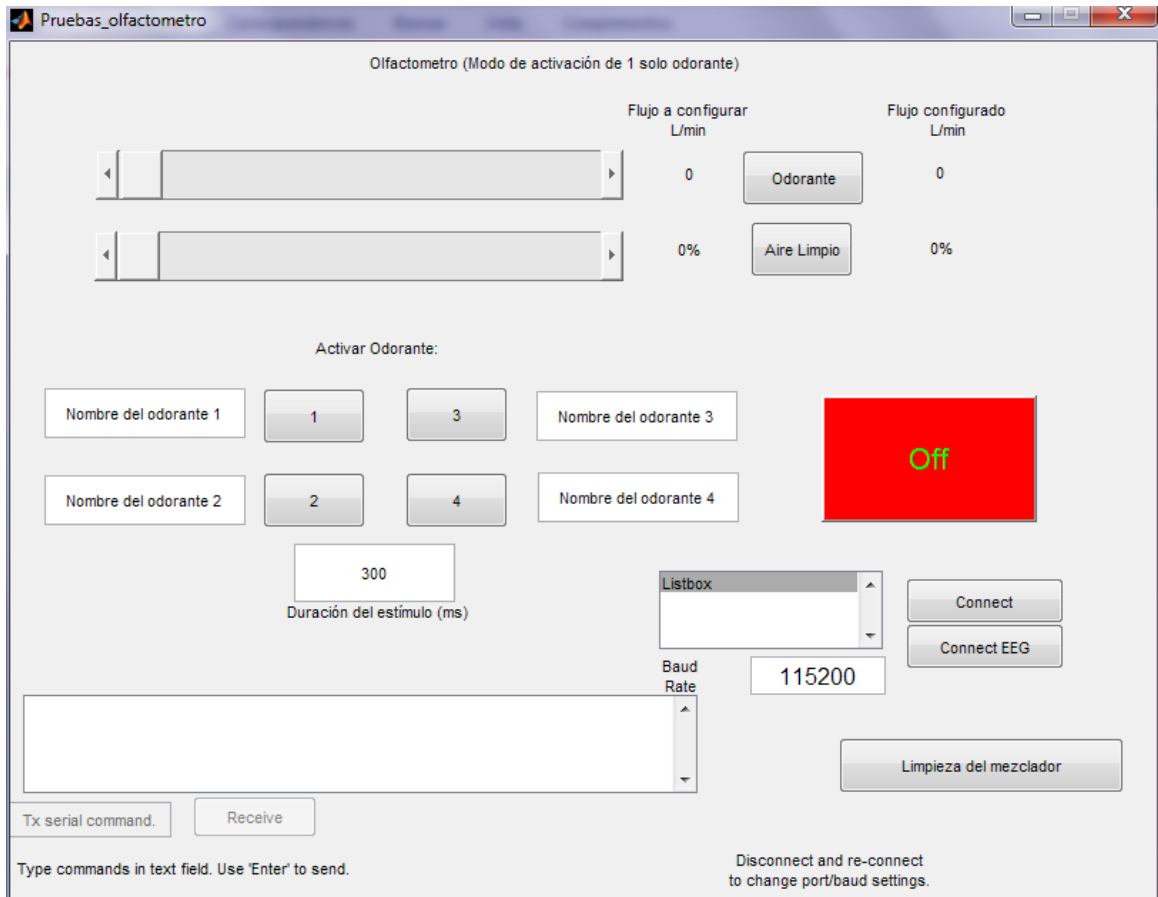
function odorante4e_Callback(hObject, eventdata, handles)
% hObject    handle to odorante4e (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of odorante4e as text
%         str2double(get(hObject,'String')) returns contents of odorante4e as a double

% --- Executes during object creation, after setting all properties.
function odorante4e_CreateFcn(hObject, eventdata, handles)
% hObject    handle to odorante4e (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

```



Distribución de la interfaz gráfica

Programación de la interfaz de pruebas para estímulos ciclicos (Matlab)

```
function varargout = ex_guide_timergui(varargin)
% Author: Luis Fernando Valdez Garduño
% Interfaz de estímulos ciclicos
% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @ex_guide_timergui_OpeningFcn, ...
                  'gui_OutputFcn',  @ex_guide_timergui_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

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

% --- Executes just before ex_guide_timergui is made visible.
function ex_guide_timergui_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 ex_guide_timergui (see VARARGIN)

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

global count
global voltage
count=1;
% START USER CODE
% Create a timer object to fire at 1/10 sec intervals
% Specify function handles for its start and run callbacks
handles.timer = timer(...
    'ExecutionMode', 'fixedRate', ...    % Run timer repeatedly
    'Period', 0.003, ...% Initial period is 1 sec.
    'TimerFcn', {@update_display,hObject}); % Specify callback function
% Initialize slider and its readout text field
voltage=0;
% Create a surface plot of peaks data. Store handle to it.
handles.plot = plot(handles.display,voltage);
```

```

xlim(handles.display,[0 256]);
ylim(handles.display,[0 256]);
handles.serConn= serial('com2' , 'BaudRate', 115200, 'InputBufferSize',1);
    fopen(handles.serConn);
    handles.serConn2= serial('com1' , 'BaudRate', 9600, 'InputBufferSize',1);
    fopen(handles.serConn2);
% END USER CODE
pause(1)
fwrite(handles.serConn, 238);
fwrite(handles.serConn, 0);
% Update handles structure
guidata(hObject,handles);

% --- Outputs from this function are returned to the command line.
function varargout = ex_guide_timergui_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;

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

% START USER CODE
% Only start timer if it is not running
if strcmp(get(handles.timer, 'Running'), 'off')
    start(handles.timer);
end
% END USER CODE

% --- Executes on button press in stopbtn.
function stopbtn_Callback(hObject, eventdata, handles)
% hObject handle to stopbtn (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
% START USER CODE
% Only stop timer if it is running
if strcmp(get(handles.timer, 'Running'), 'on')
    stop(handles.timer);
end

```

```

% END USER CODE

% START USER CODE
function update_display(hObject,eventdata,hfigure)
% Timer timer1 callback, called each time timer iterates.
% Gets surface Z data, adds noise, and writes it back to surface object.
global count
global voltage
handles = guidata(hfigure);
if handles.serConn.BytesAvailable==1
f=fread(handles.serConn);
voltage= [voltage f];
if count<256
set(handles.plot,'YData',voltage);
else
set(handles.plot,'YData',voltage(count-255:count));
end
count=count+1;
end
% END USER CODE

% --- 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)
global voltage
% START USER CODE
% Necessary to provide this function to prevent timer callback
% from causing an error after GUI code stops executing.
% Before exiting, if the timer is running, stop it.
if strcmp(get(handles.timer, 'Running'), 'on')
stop(handles.timer);
end
% Destroy timer
delete(handles.timer)
% END USER CODE
fclose(handles.serConn)
fclose(handles.serConn2)

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

% --- Executes on button press in od1.
function od1_Callback(hObject, eventdata, handles)
% hObject    handle to od1 (see GCBO)

```

```

% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
retardo=str2num(get(handles.tiempo_estimulo, 'String'));
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 1);
fwrite(handles.serConn2, 255);
fprintf(handles.serConn2, get(handles.e1, 'String'));
fprintf(handles.serConn2, get(handles.tiempo_estimulo, 'String'));
fprintf(handles.serConn2, get(handles.flujo_od, 'String'));
fprintf(handles.serConn2, get(handles.flujo_aire, 'String'));
pause(4)
fwrite(handles.serConn, 240);
fwrite(handles.serConn2, 1);
pause(retardo/1000)
fwrite(handles.serConn, 253);
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 5);
pause(1)
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 0);
fwrite(handles.serConn, 253);
pause(3)
fwrite(handles.serConn2, 254);

```

```

% --- Executes on button press in od2.
function od2_Callback(hObject, eventdata, handles)
% hObject handle to od2 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
retardo=str2num(get(handles.tiempo_estimulo, 'String'));
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 2);
fwrite(handles.serConn2, 255);
fprintf(handles.serConn2, get(handles.e2, 'String'));
fprintf(handles.serConn2, get(handles.tiempo_estimulo, 'String'));
fprintf(handles.serConn2, get(handles.flujo_od, 'String'));
fprintf(handles.serConn2, get(handles.flujo_aire, 'String'));
pause(4)
fwrite(handles.serConn, 240);
fwrite(handles.serConn2, 2);
pause(retardo/1000)
fwrite(handles.serConn, 253);
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 5);
pause(1)
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 0);
fwrite(handles.serConn, 253);

```



```

pause(3)
fwrite(handles.serConn2, 254);

% --- Executes on button press in od3.
function od3_Callback(hObject, eventdata, handles)
% hObject    handle to od3 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
retardo=str2num(get(handles.tiempo_estimulo, 'String'));
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 3);
fwrite(handles.serConn2, 255);
fprintf(handles.serConn2, get(handles.e3, 'String'));
fprintf(handles.serConn2, get(handles.tiempo_estimulo, 'String'));
fprintf(handles.serConn2, get(handles.flujo_od, 'String'));
fprintf(handles.serConn2, get(handles.flujo_aire, 'String'));pause(4)
fwrite(handles.serConn, 240);
fwrite(handles.serConn2, 3);
pause(retardo/1000)
fwrite(handles.serConn, 253);
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 5);
pause(1)
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 0);
fwrite(handles.serConn, 253);
pause(3)
fwrite(handles.serConn2, 254);

% --- Executes on button press in od4.
function od4_Callback(hObject, eventdata, handles)
% hObject    handle to od4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
retardo=str2num(get(handles.tiempo_estimulo, 'String'));
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 4);
fwrite(handles.serConn2, 255);
fprintf(handles.serConn2, get(handles.e4, 'String'));
fprintf(handles.serConn2, get(handles.tiempo_estimulo, 'String'));
fprintf(handles.serConn2, get(handles.flujo_od, 'String'));
fprintf(handles.serConn2, get(handles.flujo_aire, 'String'));pause(4)
fwrite(handles.serConn, 240);
fwrite(handles.serConn2, 4);
pause(retardo/1000)
fwrite(handles.serConn, 253);
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 5);

```

```

pause(1)
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 0);
fwrite(handles.serConn, 253);
pause(3)
fwrite(handles.serConn2, 254);

% --- Executes on button press in conf_od.
function conf_od_Callback(hObject, eventdata, handles)
% hObject    handle to conf_od (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
valor1 = floor(get(handles.slider1, 'Value'));
if valor1>44
bit=4*valor1;
T1=293;
N2=6950;
gg=1;
p1=0.2+0.000385*bit;
cv=0.014;
dp=p1-0.2;
%figure;
q2=N2.*cv.*p1.*(1-(2.*dp)./(3.*p1)).*(dp./(p1.*gg.*T1)).^0.5;
else
    q2=0;
    valor1=0;
end
set(handles.flujo_od, 'String', num2str(floor(q2*100)/100));
fwrite(handles.serConn, 250);
fwrite(handles.serConn, valor1);

% --- Executes on button press in conf_aire.
function conf_aire_Callback(hObject, eventdata, handles)
% hObject    handle to conf_aire (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
load('presionbom2.mat')
valor1 = floor(get(handles.slider2, 'Value'))
T1=293;
N2=6950;
gg=1;
p1=0.2+pf(valor1);
cv=0.014;
dp=p1-0.2;
%figure;
q2=N2.*cv.*p1.*(1-(2.*dp)./(3.*p1)).*(dp./(p1.*gg.*T1)).^0.5;
set(handles.flujo_aire, 'String', num2str(floor(q2*100)/100));
fwrite(handles.serConn, 255);
fwrite(handles.serConn, valor1);

```

```

% --- Executes on button press in apagar.
function apagar_Callback(hObject, eventdata, handles)
% hObject    handle to apagar (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
fwrite(handles.serConn, 253);
fwrite(handles.serConn, 255);
fwrite(handles.serConn, 0);

% --- Executes on slider movement.
function slider1_Callback(hObject, eventdata, handles)
% hObject    handle to slider1 (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,'Value') returns position of slider
%        get(hObject,'Min') and get(hObject,'Max') to determine range of slider
h=get(hObject, 'Value');
if h>44
bit=4*floor(h);
T1=293;
N2=6950;
gg=1;
p1=0.2+0.000385*bit;
cv=0.014;
dp=p1-0.2;
%figure;
q2=N2.*cv.*p1.*(1-(2.*dp)./(3.*p1)).*(dp./(p1.*gg.*T1)).^0.5;
else
    q2=0;
end
%hold on
%plot(bit,q2,'r')
set(handles.con_od, 'String', num2str(floor(q2*100)/100));

% --- Executes during object creation, after setting all properties.
function slider1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

```

```

% --- Executes on slider movement.
function slider2_Callback(hObject, eventdata, handles)
% hObject    handle to slider2 (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,'Value') returns position of slider
%         get(hObject,'Min') and get(hObject,'Max') to determine range of slider
load('presionbom2.mat'); % Carga la curva de clalibración del canal de aire
h=floor(get(hObject, 'Value'));
T1=293;
N2=6950;
gg=1;
p1=0.2+pf(h);
cv=0.014;
dp=p1-0.2;
%figure;
q2=N2.*cv.*p1.*(1-(2.*dp)./(3.*p1)).*(dp./(p1.*gg.*T1)).^0.5;
set(handles.con_aire, 'String', num2str(floor(q2*100)/100));

% --- Executes during object creation, after setting all properties.
function slider2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to slider2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

function tiempo_estimulo_Callback(hObject, eventdata, handles)
% hObject    handle to tiempo_estimulo (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 tiempo_estimulo as text
%         str2double(get(hObject,'String')) returns contents of tiempo_estimulo as a
double

% --- Executes during object creation, after setting all properties.
function tiempo_estimulo_CreateFcn(hObject, eventdata, handles)
% hObject    handle to tiempo_estimulo (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

function e1_Callback(hObject, eventdata, handles)
% hObject    handle to e1 (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 e1 as text
%        str2double(get(hObject,'String')) returns contents of e1 as a double

% --- Executes during object creation, after setting all properties.
function e1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to e1 (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

function e2_Callback(hObject, eventdata, handles)
% hObject    handle to e2 (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 e2 as text
%        str2double(get(hObject,'String')) returns contents of e2 as a double

% --- Executes during object creation, after setting all properties.
function e2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to e2 (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

function e3_Callback(hObject, eventdata, handles)
% hObject    handle to e3 (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 e3 as text
%        str2double(get(hObject,'String')) returns contents of e3 as a double

% --- Executes during object creation, after setting all properties.
function e3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to e3 (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

function e4_Callback(hObject, eventdata, handles)
% hObject    handle to e4 (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 e4 as text
%        str2double(get(hObject,'String')) returns contents of e4 as a double

% --- Executes during object creation, after setting all properties.
function e4_CreateFcn(hObject, eventdata, handles)
% hObject    handle to e4 (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

function tiempo_descanso_Callback(hObject, eventdata, handles)
% hObject    handle to tiempo_descanso (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 tiempo_descanso as text
%        str2double(get(hObject,'String')) returns contents of tiempo_descanso as a
double

% --- Executes during object creation, after setting all properties.
function tiempo_descanso_CreateFcn(hObject, eventdata, handles)
% hObject    handle to tiempo_descanso (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

function ciclos_Callback(hObject, eventdata, handles)
% hObject    handle to ciclos (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 ciclos as text
%        str2double(get(hObject,'String')) returns contents of ciclos as a double

% --- Executes during object creation, after setting all properties.
function ciclos_CreateFcn(hObject, eventdata, handles)
% hObject    handle to ciclos (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_prueba.
function Inicio_prueba_Callback(hObject, eventdata, handles)
% hObject    handle to Inicio_prueba (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
global voltage
global count
set(handles.umbra1, 'Value', 70);
retardo=str2num(get(handles.tiempo_estimulo, 'String'));%%tiempo de estimulación
tiempoestimulo=get(handles.tiempo_estimulo, 'String');
tinter=str2num(get(handles.tiempo_descanso, 'String'));%%tiempo inter estimulo
tiempointerestimulo=get(handles.tiempo_descanso, 'String');
nciclos=str2num(get(handles.ciclos, 'String'));%%numero de ciclos
numerodeciclos=get(handles.ciclos, 'String');

etiquetas{1}='111111111'; %%etiqueta deseada
nombre1=get(handles.e1, 'String');
etiquetas{2}='222222222'; %%etiqueta deseada
nombre2=get(handles.e2, 'String');
etiquetas{3}='333333333'; %%etiqueta deseada
nombre3=get(handles.e3, 'String');
etiquetas{4}='444444444'; %%etiqueta deseada
nombre4=get(handles.e4, 'String');
es(1)=str2num(get(handles.estim1, 'String'));
es(2)=str2num(get(handles.estim2, 'String'));
es(3)=str2num(get(handles.estim3, 'String'));
es(4)=str2num(get(handles.estim4, 'String'));

fecha=datestr(now,30);
if strcmp(get(handles.timer, 'Running'), 'off')
    flushinput(handles.serConn);
    start(handles.timer);
end
tic
inicio=count;
pause(0.1)
fwrite(handles.serConn, 238);
fwrite(handles.serConn, 1);

for rep=1:nciclos
    rep
    for numod=1:4
        %% preparación
        fwrite(handles.serConn2, 255);%%inicio datos estimulo
        fwrite(handles.serConn, 239);
        fwrite(handles.serConn, es(numod));
        pause(1)
        while(mean(voltage(count-64:count))<get(handles.umbra1, 'Value'))
            pause(0.001)
        end
    end
end

```



```

end
while (mean(voltage(count-64:count))>get(handles.umbral,'Value'))
    pause(0.001)
end
%%fwrite(handles.serConn2, 255);%%inicio datos estimulo
fprintf(handles.serConn2, etiquetas{es(numod)}); %etiqueta

fwrite(handles.serConn, 240); %% activar
fwrite(handles.serConn2, 10); %% odorante%%
pause(retardo/1000)%% estimulación
fwrite(handles.serConn, 253);%%cerrar valvulas recipientes

fwrite(handles.serConn2, 11); %%cierre odorante%%

fwrite(handles.serConn, 239);
fwrite(handles.serConn, 5);%% Purgar limpiar
pause(1)

fwrite(handles.serConn, 239);
fwrite(handles.serConn, 0); %% cierre mezclador

fwrite(handles.serConn, 253);%% apagar
pause(tinter/1000-1)
fwrite(handles.serConn2, 128); %%%fin ciclo
end

end
'fin'
fwrite(handles.serConn, 238);
fwrite(handles.serConn, 0);
pause(1)
fin=count;
stop(handles.timer)
j=toc;
duracion=num2str(j);
inicio
fin
length(voltage)
prueba=voltage(inicio:fin);
voltage=0;
count=1;
nombre=strcat('registro_', fecha, '_', duracion, '.mat');
datos_prueba=strcat('tiempo del estimulo=', tiempoestimulo, ' tiempo
interestimulo=', tiempointerestimulo, 'numero de ciclos =', numerodeciclos);
lista_odorantes=strcat('odorante 1 =', nombre1, 'odorante 2 =', nombre2, 'odorante 3
=', nombre3, 'odorante 4 =', nombre4);
save(nombre, 'prueba', 'datos_prueba', 'lista_odorantes');

```

```

function estim1_Callback(hObject, eventdata, handles)
% hObject    handle to estim1 (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 estim1 as text
%         str2double(get(hObject,'String')) returns contents of estim1 as a double

% --- Executes during object creation, after setting all properties.
function estim1_CreateFcn(hObject, eventdata, handles)
% hObject    handle to estim1 (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

function estim2_Callback(hObject, eventdata, handles)
% hObject    handle to estim2 (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 estim2 as text
%         str2double(get(hObject,'String')) returns contents of estim2 as a double

% --- Executes during object creation, after setting all properties.
function estim2_CreateFcn(hObject, eventdata, handles)
% hObject    handle to estim2 (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

```

```

function estim3_Callback(hObject, eventdata, handles)
% hObject    handle to estim3 (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 estim3 as text
%        str2double(get(hObject,'String')) returns contents of estim3 as a double

% --- Executes during object creation, after setting all properties.
function estim3_CreateFcn(hObject, eventdata, handles)
% hObject    handle to estim3 (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

function estim4_Callback(hObject, eventdata, handles)
% hObject    handle to estim4 (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 estim4 as text
%        str2double(get(hObject,'String')) returns contents of estim4 as a double

% --- Executes during object creation, after setting all properties.
function estim4_CreateFcn(hObject, eventdata, handles)
% hObject    handle to estim4 (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 od5.
function od5_Callback(hObject, eventdata, handles)
% hObject    handle to od5 (see GCBO)

```

```

% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
fwrite(handles.serConn, 239);
fwrite(handles.serConn, 5);

% --- Executes on slider movement.
function umbral_Callback(hObject, eventdata, handles)
% hObject handle to umbral (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,'Value') returns position of slider
% get(hObject,'Min') and get(hObject,'Max') to determine range of slider

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

% Hint: slider controls usually have a light gray background.
if isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor',[.9 .9 .9]);
end

```



Distribución de la interfaz gráfica para estímulos cíclicos

Segmentación automática de registros de Emotiv® TestBench®

```
clc; clear all;
%%registro de test bench o registro a segmentar
[hdr, record] = edfread('anais-2-26.08.14.16.27.44.edf');
%% registro de las señales suplementarias/ respiración
load('registro_20140826T193139_128.3725.mat')
%%normarilación y corrección de datos en caso de ser necesario
COUNTER=record(1,:);
INTERPOLATED=record(2,:);
AF3=record(3,:);
% af3=AF3;
% AF3=AF3+(max(AF3)*2+(AF3<0)).*(AF3<0);
% record(3,:)=AF3;
%
F7=record(4,:);
% f7=F7;
% F7=F7+(max(F7)*2+(F7<0)).*(F7<0);
% record(4,:)=F7;

F3=record(5,:);
% f3=F3;
% F3=F3+(max(F3)*2+(F3<0)).*(F3<0);
% record(5,:)=F3

FC5=record(6,:);
% fc5=FC5;
% FC5=FC5+(max(FC5)*2+(FC5<0)).*(FC5<0);
% record(6,:)=FC5;

T7=record(7,:);
% t7=T7;
% T7=T7+(max(T7)*2+(T7<0)).*(T7<0);
% record(7,:)=T7;

P7=record(8,:);
% p7=P7;
% P7=P7+(max(P7)*2+(P7<0)).*(P7<0);
% record(8,:)=P7;

O1=record(9,:);
% o1=O1;
% O1=O1+(max(O1)*2+(O1<0)).*(O1<0);
% record(9,:)=O1;

O2=record(10,:);
% o2=O2;
% O2=O2+(max(O2)*2+(O2<0)).*(O2<0);
% record(10,:)=O2;
```

```

P8=record(11,:);
% p8=P8;
% P8=P8+(max(P8)*2+(P8<0)).*(P8<0);
% record(11,:)=P8;
T8=record(12,:);
% t8=T8;
% T8=T8+(max(T8)*2+(T8<0)).*(T8<0);
% record(12,:)=T8;

FC6=record(13,:);
% fc6=FC6;
% FC6=FC6+(max(FC6)*2+(FC6<0)).*(FC6<0);
% record(13,:)=FC6;

F4=record(14,:);
f4=F4;
F4=F4+(max(F4)*2+(F4<0)).*(F4<0);
record(14,:)=F4;
F8=record(15,:);
% f8=F8;
% F8=F8+(max(F8)*2+(F8<0)).*(F8<0);
% record(15,:)=F8;

AF4=record(16,:);
% af4=AF4;
% AF4=AF4+(max(AF4)*2+(AF4<0)).*(AF4<0);
% record(16,:)=AF4;

RAW_CQ=record(17,:);
CQ_AF3=record(18,:);
CQ_F7=record(19,:);
CQ_F3=record(20,:);
CQ_FC5=record(21,:);
CQ_T7=record(22,:);
CQ_P7=record(23,:);
CQ_O1=record(24,:);
CQ_O2=record(25,:);
CQ_P8=record(26,:);
CQ_T8=record(27,:);
CQ_FC6=record(28,:);
CQ_F4=record(29,:);
CQ_F8=record(30,:);
CQ_AF4=record(31,:);
CQ_CMS=record(32,:);
CQ_DRL=record(33,:);
GYROX=record(34,:);
GYROY=record(35,:);
MARKER=record(36,:);

%%segmentación en base a registro de marcadores
inicio=find(MARKER==255);%%ubicación de marcadores de inicio de prueba

```

```

fin=find(MARKER==128);%%ubicación de marcadores de fin de prueba
inest=find(MARKER==255);%%ubicación de marcadores de inicio de estimulación
finest=find(MARKER==11);%%ubicación de marcadores de fin de estimulación

largo=1420; %% segmentos de tiempo que se desea tomar del registro
        %% T estimulo + T interstimulo + T deseado
%% segmentación del registro de respiración
h=find(prueba==0);
j=[h,0]-[0,h];
k=find(j~= 1);
inter=h(k(1));
zerosfinal=[1:(length(AF4)-fin(8))].*0;
zerosinicio=[1:(length(AF4)-length(prueba(1:inter))-length(zerosfinal))].*0;
respiracion=[zerosinicio,prueba(1:inter),zerosfinal];
%% recorta el registro y sincroniza en base a los marcadores de inicio y fin

%% gráfica del sensor de temperatura con el registro de un canal de EEG
% figure
% plot(1/128:1/128:length(AF3)/128,smooth(AF3-mean(AF3),16),'k')
% hold on
% plot(1/128:1/128:length(AF3)/128,20+respiracion/16,'m') %escalamiento temperatura
%% gráfica secciones a segmentar y marcadores
% for numero=1:7
% plot((finest(numero):16:fin(numero))/128,50,'*b')
% plot((inicio(numero):16:finest(numero))/128,50,'+g')
% plot((fin(numero):16:inicio(numero+1))/128,50,'.r')
% end
%%segmenta en base a los marcadores de inicio o fin en caso de perderse un
%%marcador
if length(fin)>length(inicio)
    g=length(fin);
    for muestra=1:g
        registro(:, :,muestra)=[record(:, fin(muestra)-
largo:fin(muestra));respiracion(fin(muestra)-largo:fin(muestra))];
        end
    else
        g=length(inicio);
        for muestra=1:g
            registro(:, :,muestra)=[record(:, inicio(muestra):inicio(muestra)+largo);respiracion(inicio(m
uestra):inicio(muestra)+largo)];
            end
        end
end
end

```


Anexo B

(Diagramas eléctricos y PCB)

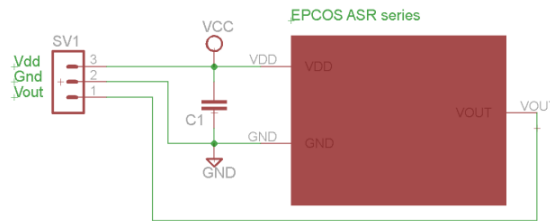


Diagrama eléctrico del módulo del sensor de presión



PCB del módulo del sensor de presión

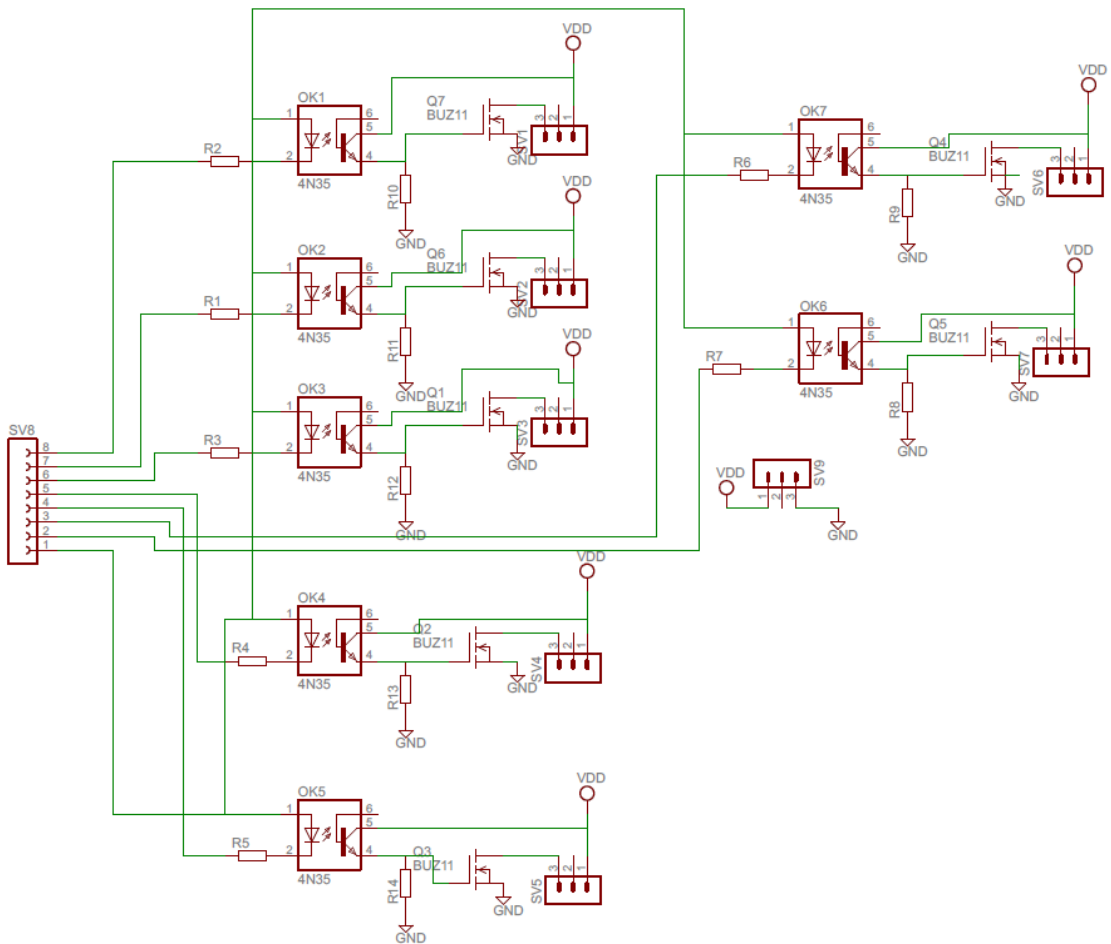
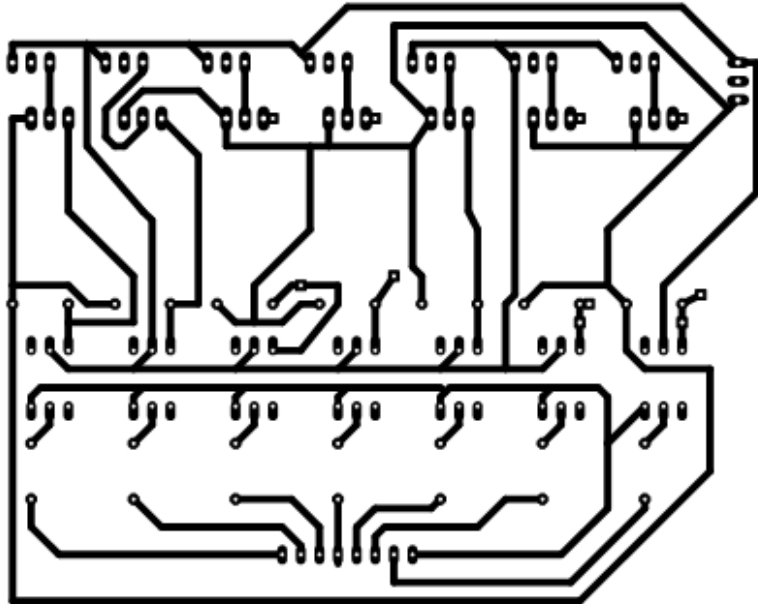
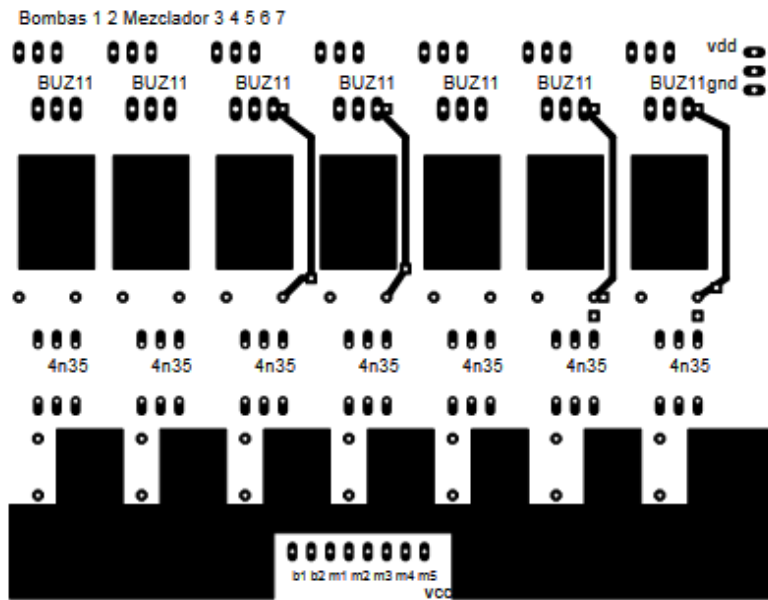


Diagrama eléctrico del módulo de potencia 1



Revés del PCB del modulo de potencia 1



Frente del PCB del modulo de potencia 1

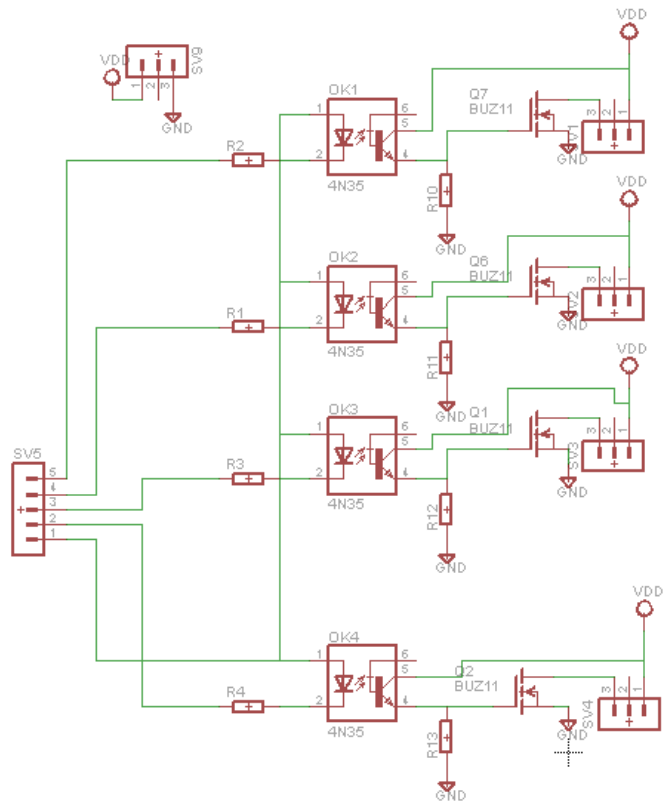
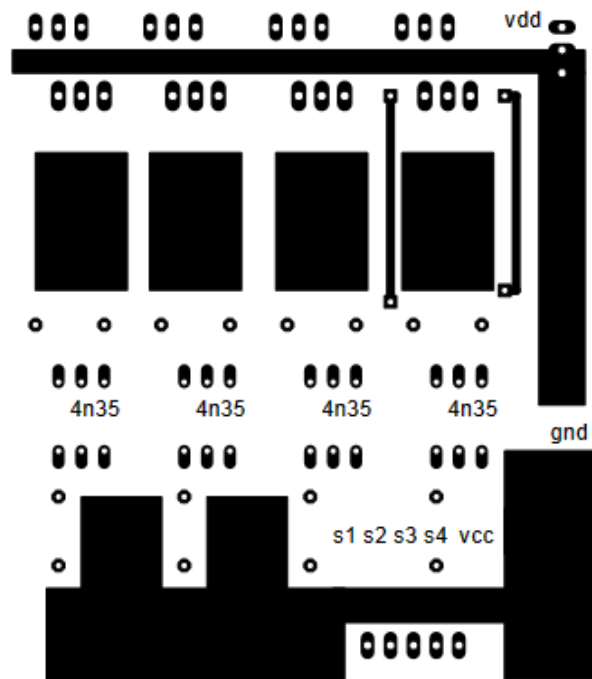
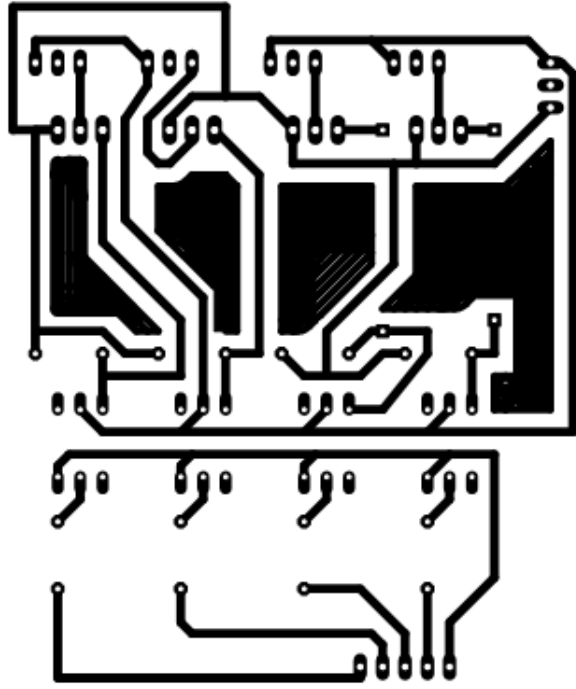


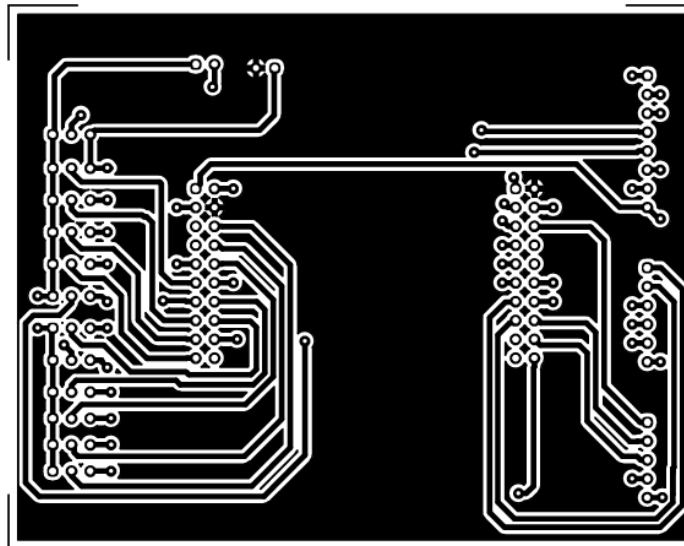
Diagrama eléctrico del modulo de potencia 2



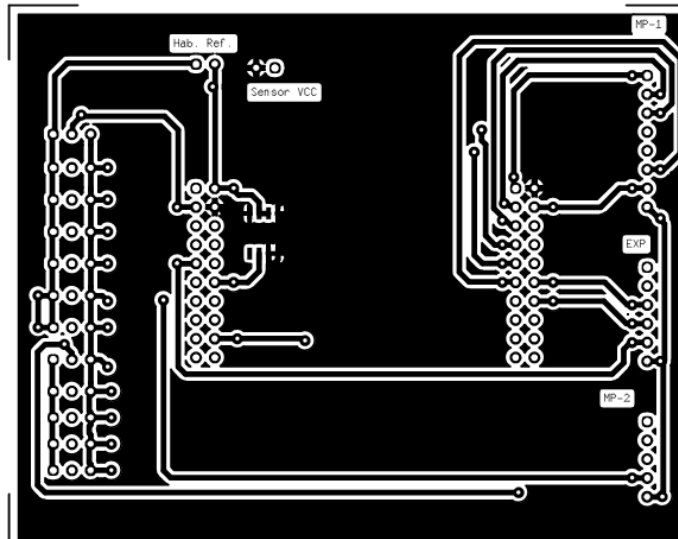
Frente del PCB del modulo de potencia 2



Revés del PCB del modulo de potencia 2



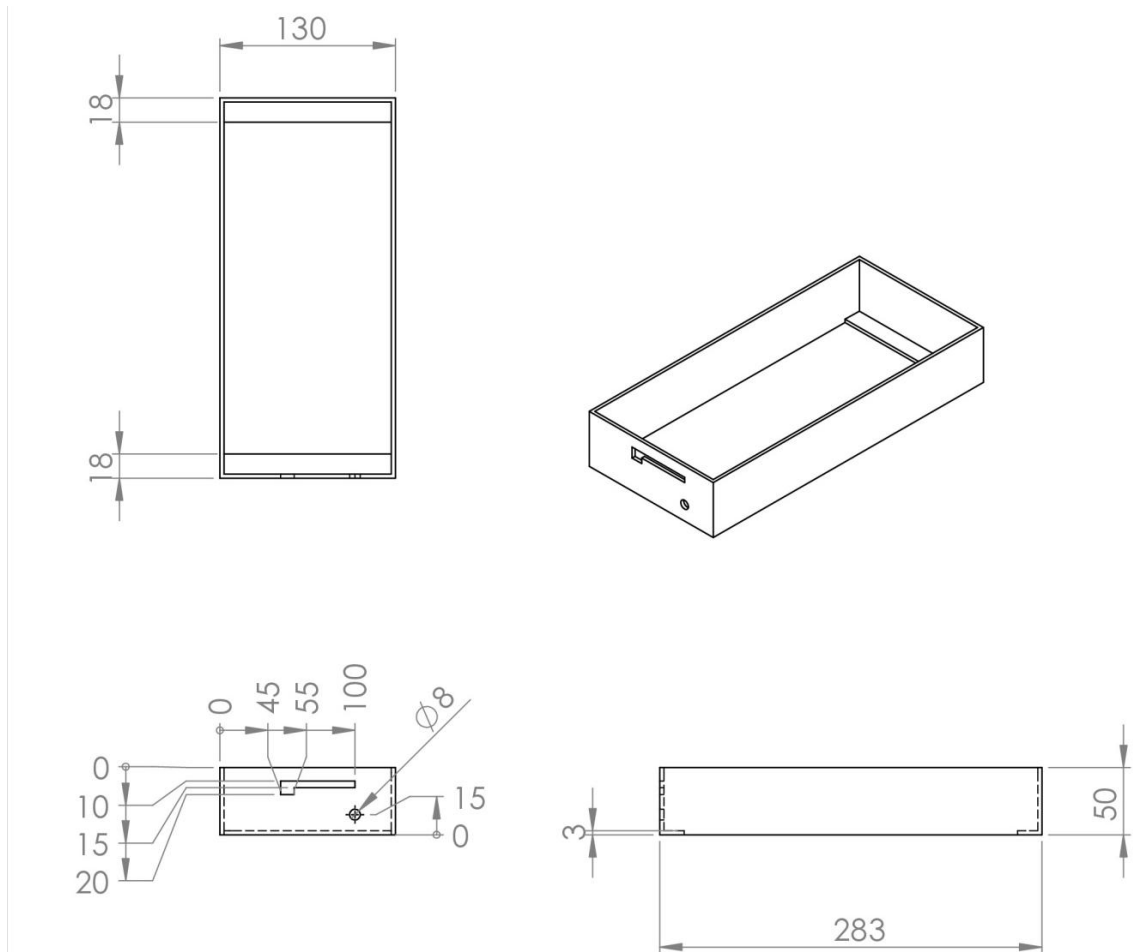
Revés de la tarjeta de interconexión



Frente de la tarjeta de interconexión

Anexo C

(Diseño de carcasa)



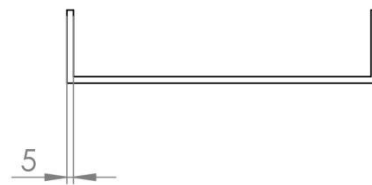
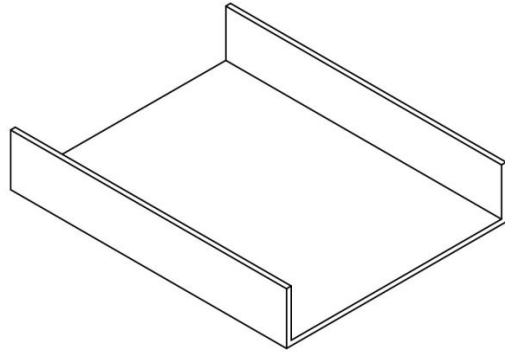
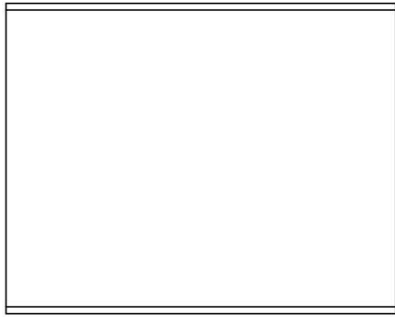
Autor: Luis Fernando Valdez Garduño

Pieza:
base_circuitos

Cinvestav - Bioelectrónica

Escala: 1:5

Unidades:
mm



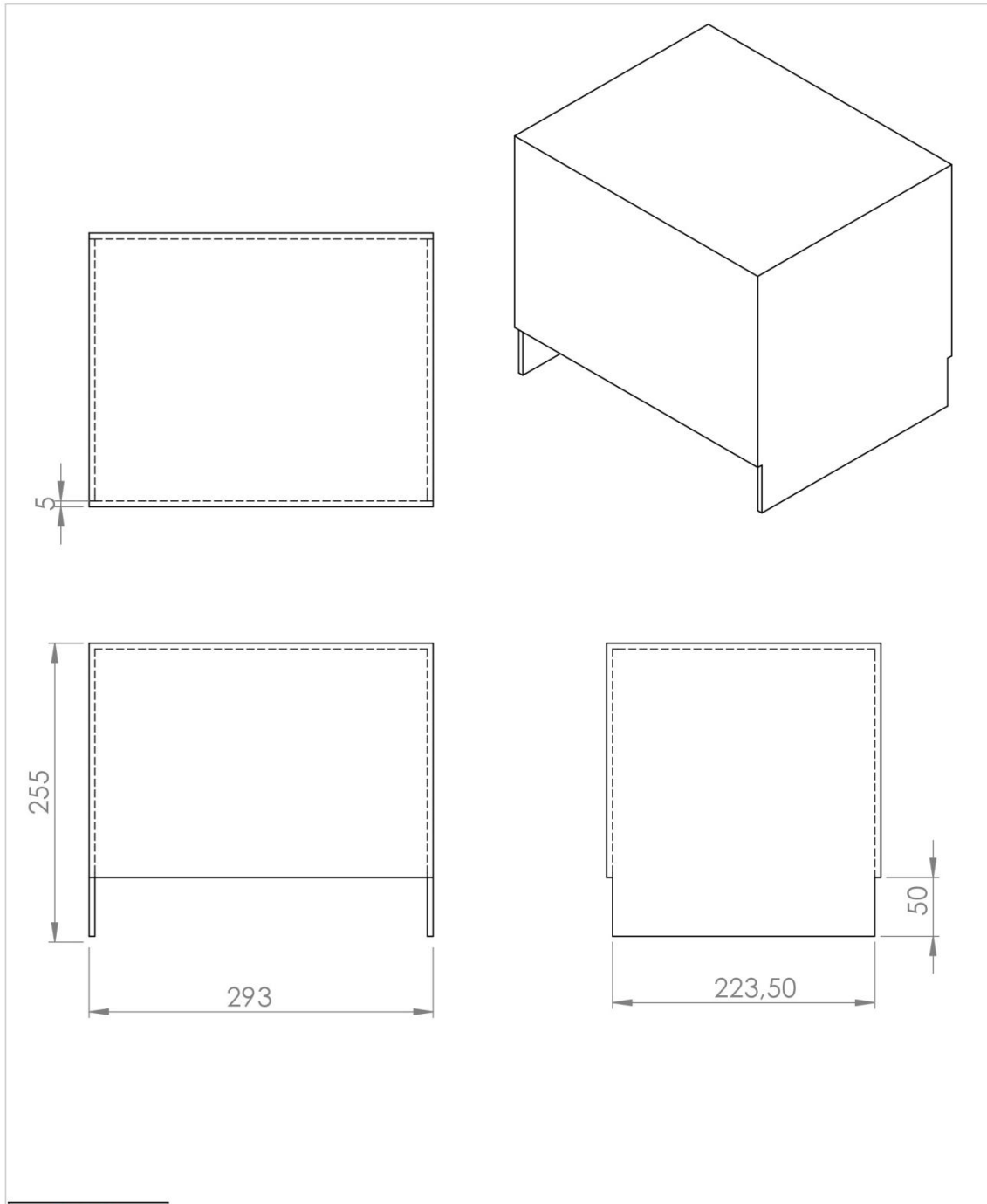
Autor: Luis Fernando Valdez Garduño

Pieza:
base_cubierta

Cinvestav - Bioelectrónica

Escala: 1:5

Unidades:
mm



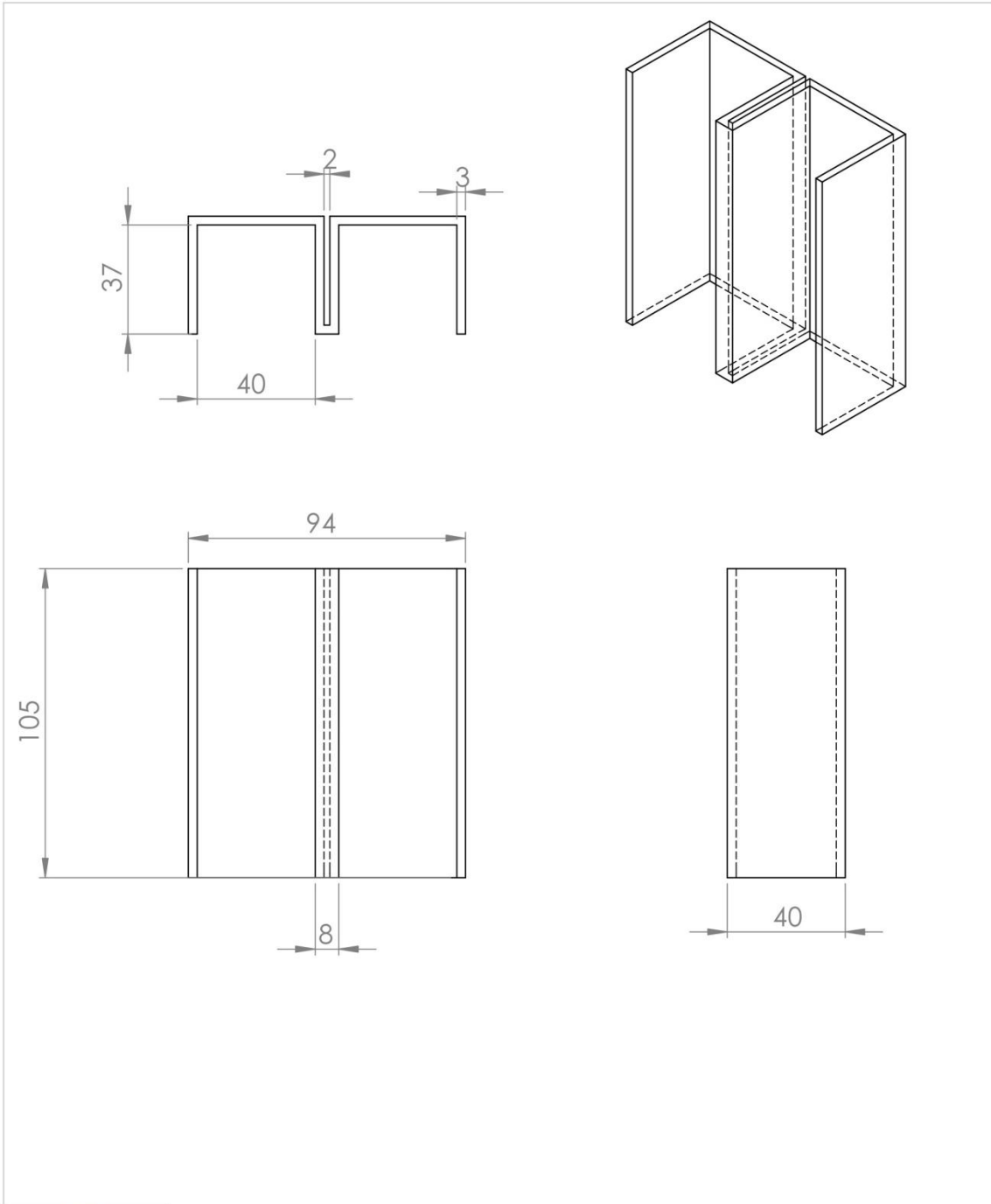
Autor: Luis Fernando Valdez Garduño

Pieza:
cubierta

Cinvestav - Bioelectrónica

Escala: 1:5

Unidades:
mm



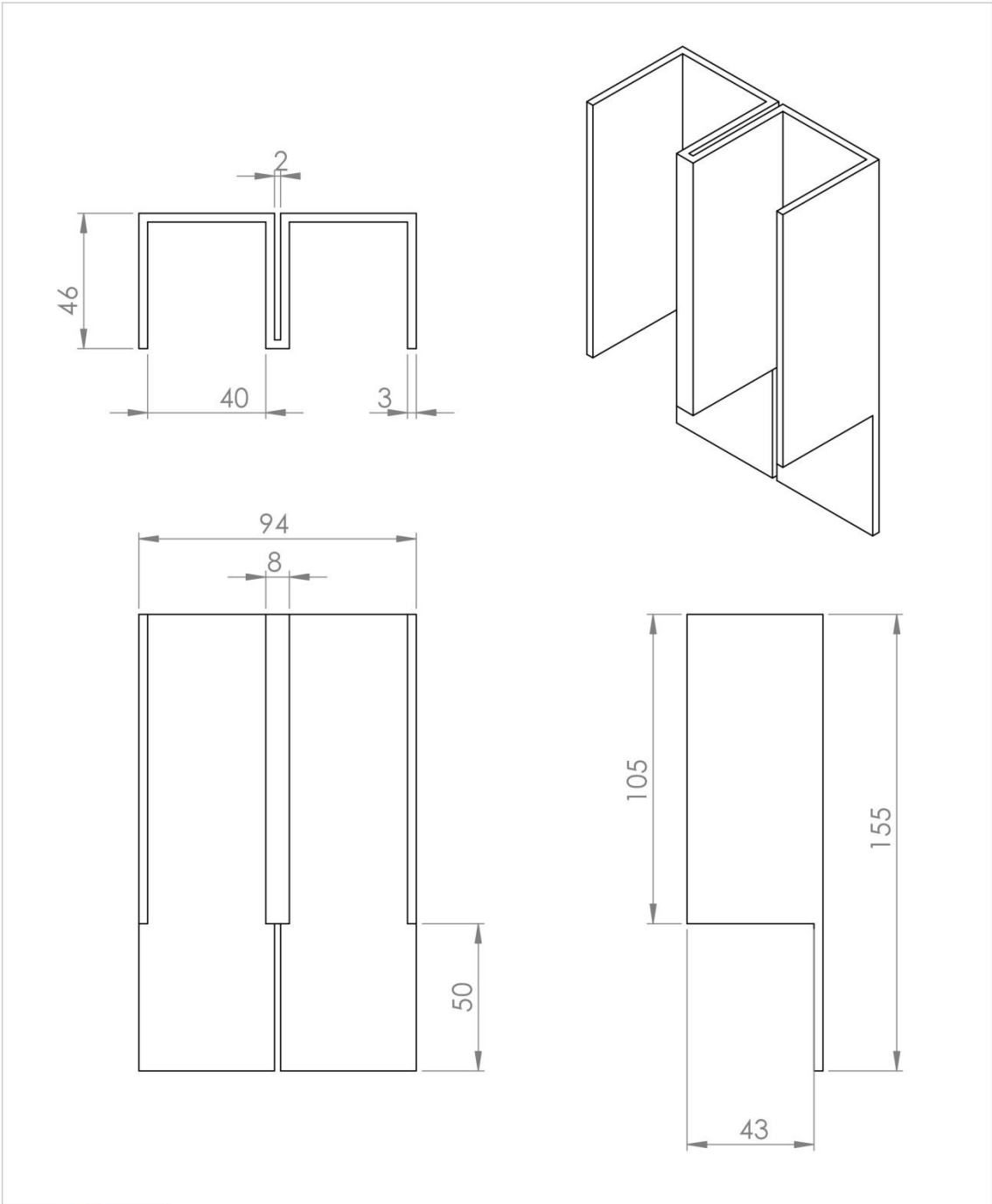
Autor: Luis Fernando Valdez Garduño

Pieza:
Filtro_delantero

Cinvestav - Bioelectrónica

Escala: 1:5

Unidades:
mm



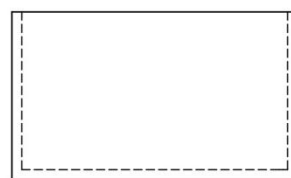
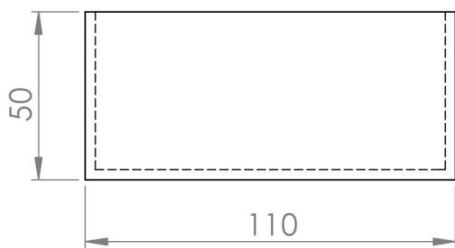
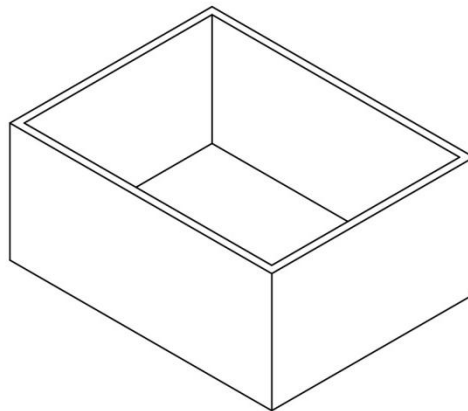
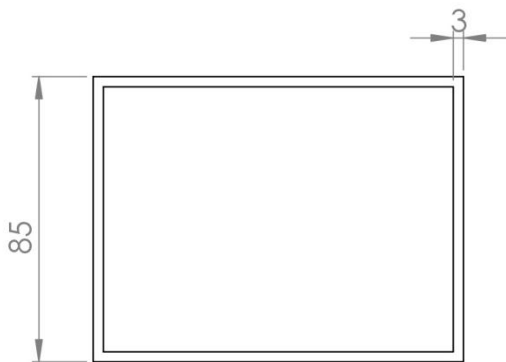
Autor: Luis Fernando Valdez Garduño

Pieza:
Filtro_trasero

Cinvestav - Bioelectrónica

Escala: 1:2

Unidades:
mm



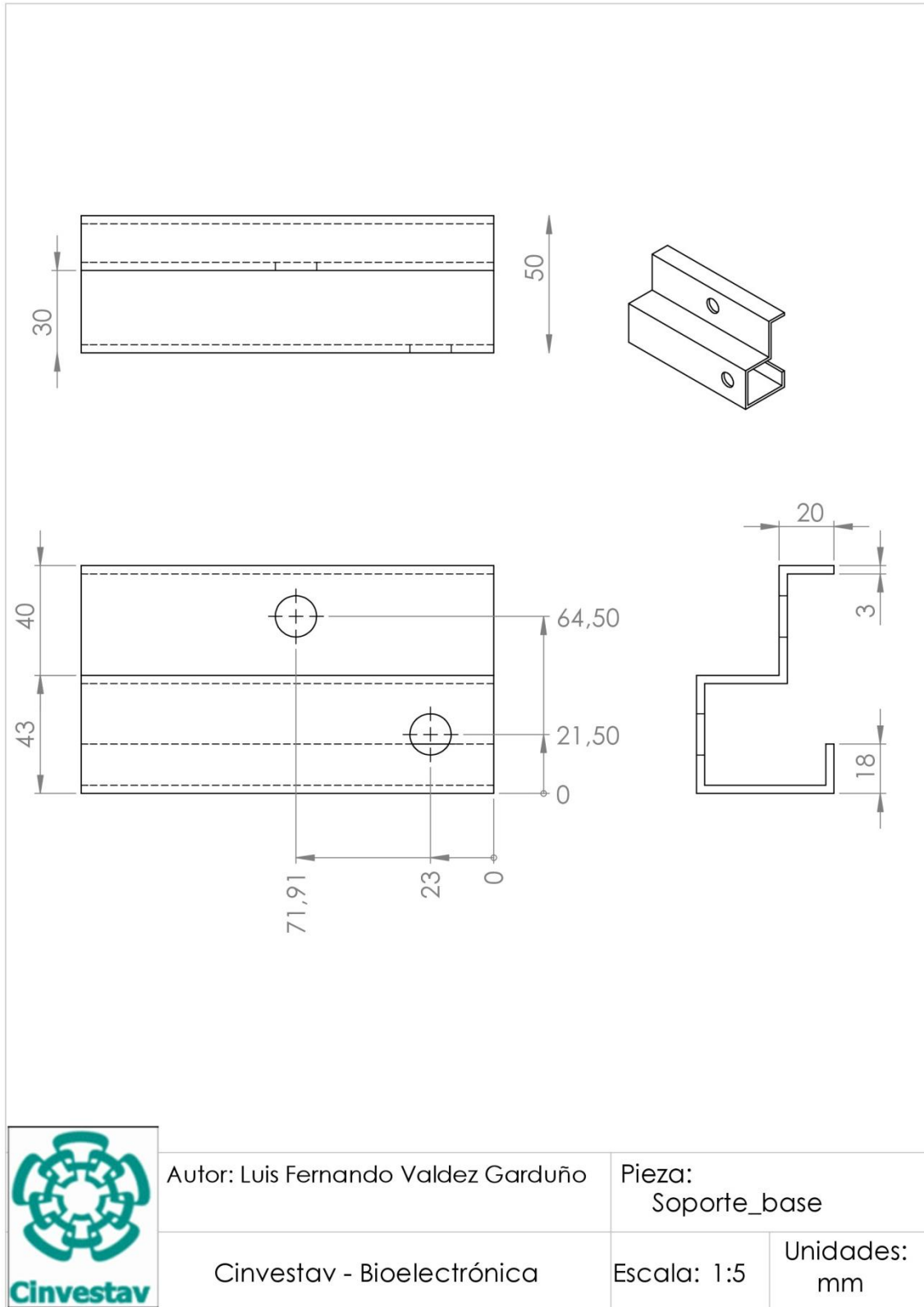
Autor: Luis Fernando Valdez Garduño

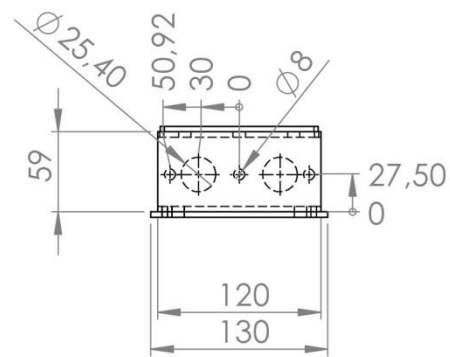
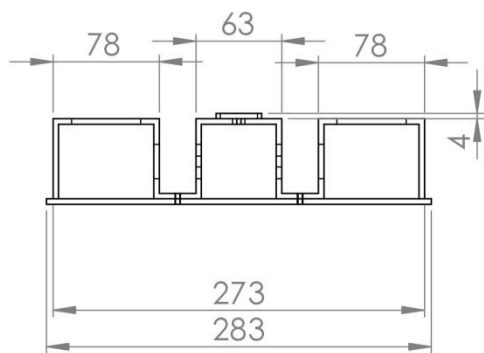
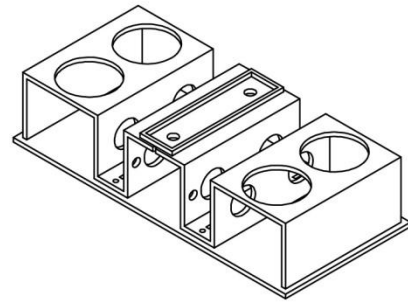
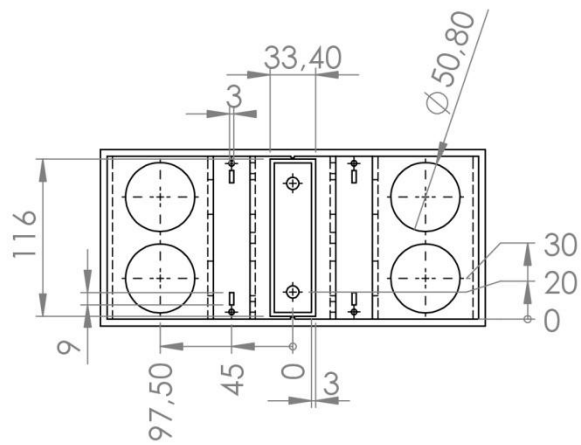
Pieza:
soporte_bombas

Cinvestav - Bioelectrónica

Escala: 1:2

Unidades:
mm





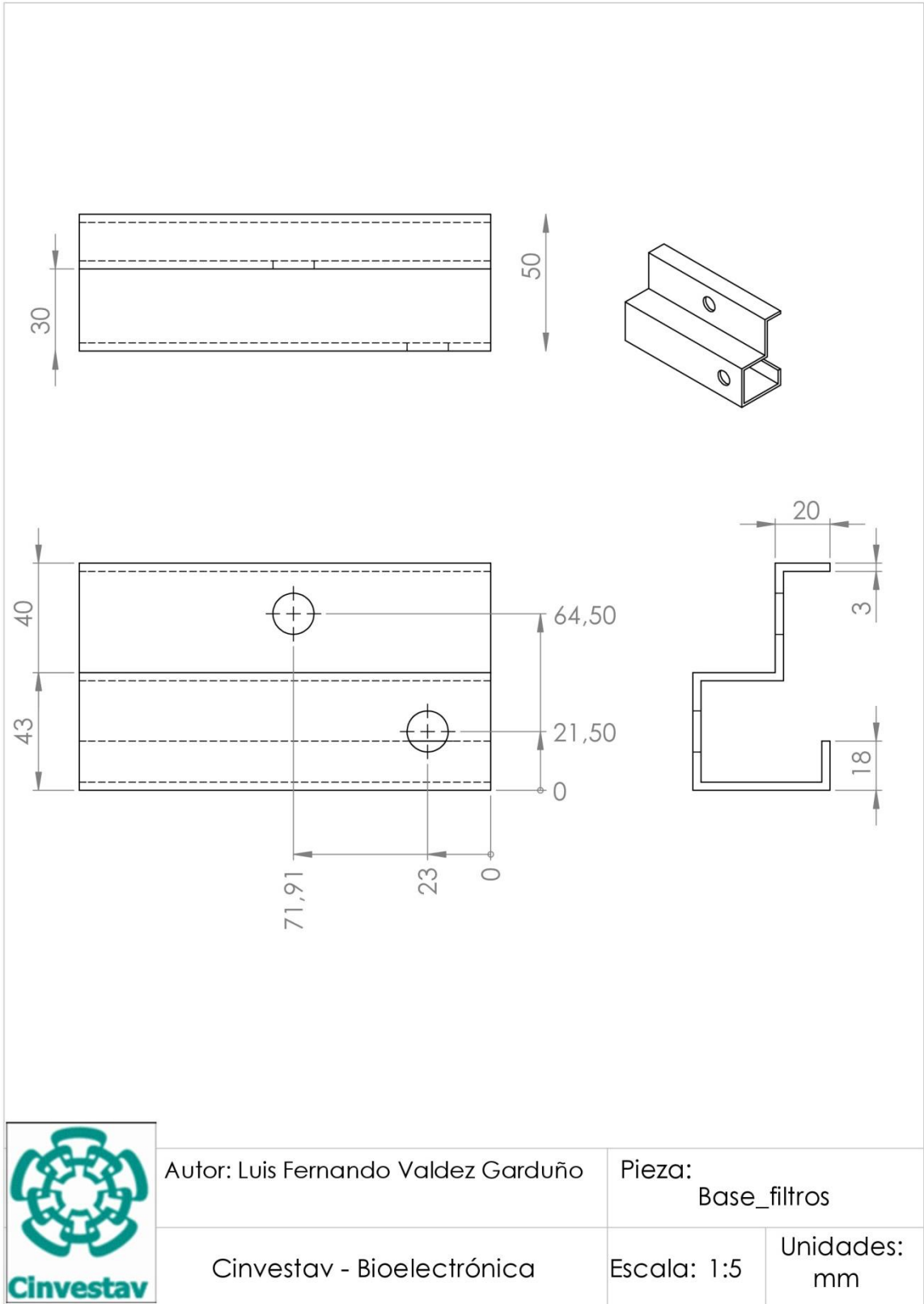
Autor: Luis Fernando Valdez Garduño

Pieza:
Soporte_contenedores

Cinvestav - Bioelectrónica

Escala: 1:5

Unidades:
mm



Anexo D

(Artículos de congreso)



Desarrollo de un olfatómetro controlado por computadora para investigación clínica

Luis Fernando Valdez, Juan Manuel Gutiérrez

Sección de Bioelectrónica
Departamento de Ingeniería Eléctrica
Centro de Investigación y de Estudios Avanzados del I.P.N.
Av. Instituto Politécnico Nacional No. 2508 Col. San Pedro Zacatenco, D.F., México

fvaldez@cinvestav.mx; mgutierrez@cinvestav.mx

RESUMEN

En este trabajo se describe el diseño y funcionamiento de un olfatómetro activo para pruebas de olfactometría y potenciales evocados en laboratorio. El sistema cuenta con 4 canales y es capaz de manejar muestras sólidas o líquidas como odorantes. Los diferentes parámetros de estimulación son enviados por medio de un puerto serie para que sean almacenados junto con el registro electroencefalográfico de 14 canales. Algunas ventajas del sistema están en relación a su portabilidad, bajo mantenimiento y adecuación simple de la sección de administración, por lo que puede ser utilizado en una amplia gama de estudios olfatométricos. En la salida del sistema se obtuvo un intervalo de trabajo de 0.6 a 1.5 l/min para aire olorizado y de 0.6 a 3 l/min para aire limpio.

PALABRAS CLAVE: Olfatómetro, Olfactometría, EEG.

1 INTRODUCCIÓN

Los procesos sensitivos son la manera en que un ser vivo puede obtener retroalimentación sobre su interacción con el ambiente; permitiendo determinar su reacción ante cada estímulo. Uno de estos procesos es el olfato que en conjunto con el gusto son conocidos como los sentidos químicos ya que permiten distinguir ciertos compuestos transportados por el aire o los alimentos. En el caso del olfato dicha percepción puede desencadenar una amplia gama de reacciones las cuales pueden ir desde rechazar un alimento potencialmente dañino o aceptarlo por apetecible y agradable sin la necesidad de probarlo, hasta desencadenar una respuesta hormonal o emocional específica [1].

Si bien se ha avanzado en la comprensión de dichos procesos estímulo-reacción existen aún muchas incógnitas. Una de ellas es cómo se ve afectado el olfato por distintos estados físicos o psicológicos dando mayor peso a los estados que se consideran patológicos [2,3]. En la actualidad existe un número reducido de herramientas que se enfocan a la evaluación de dicho sentido para diagnóstico médico. Comúnmente, las pruebas que se utilizan se enfocan en evaluar tres características del olfato: identificación, discriminación y umbral de olfacción.

Las pruebas que se utilizan para evaluar la capacidad de identificación constan de la presentación de una serie de olores que el paciente debe identificar al percibirlos. El proceso de discriminación se realiza presentando una serie de 3 o 4 muestras de las cuales solo una es distinta a las demás y se pide al paciente que la identifique. Para determinar el umbral de olfacción las muestras olfativas son presentadas al paciente considerando distintas concentraciones (que comúnmente van de menor a mayor) hasta que el paciente es capaz de percibir las. Lamentablemente las pruebas antes mencionadas presentan un alto componente de subjetividad ya que dependen totalmente de la respuesta del paciente [4].

Existen otro tipo de pruebas más objetivas que evalúan la función cerebral producto de la estimulación con ayuda de técnicas como la resonancia magnética funcional (fMRI) o el análisis de potenciales evocados por electroencefalografía (EEG) [5,6,7]. Sin embargo, el empleo de éstas

requiere de instrumental especializado para proporcionar una estimulación controlada; los denominados "olfatómetros" [8]. Comúnmente estos instrumentos suelen ser diseñados para un solo tipo de prueba por lo que su utilización se encuentra limitada.

El objetivo principal de este trabajo es diseñar y construir un instrumento de características portátiles, versátil, de bajo costo, mantenimiento mínimo y capaz de implementar las pruebas de identificación, discriminación y umbral de olfacción de forma controlada.

2 METODO Y MATERIALES

2.1 Propuesta de diseño

Se propone la fabricación de un olfatómetro portable cuyas características permitan la correcta adquisición de los potenciales evocados olfativos (OEPs) y sea pueda ser compatible con estudios de **fRMI** en una segunda fase de implementación realizando cambios mínimos en el diseño del prototipo. Sus características principales son: Disponer de 4 canales para odorantes y uno para aire limpio, poder controlar mediante PC los tiempos de activación y flujos de salida (aire limpio y aire olorizado), tener su propia fuente de aire y contar con una sección de administración independiente. Estas características permitirán la implementación de pruebas de olfacción de forma automatizada.

2.2 Diseño y construcción del olfatómetro

El diseño general del instrumento se divide en 5 secciones: Fuente de aire, tratamiento de aire, selección de odorantes, control de flujo y la sección de administración como se muestra en la Figura 1.

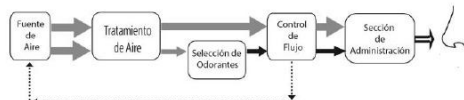


Figura 1. Diagrama a bloques del olfatómetro propuesto. (Flecha gris corriente de aire limpio, Flecha negra corriente de aire olorizado y flecha punteada control electrónico)

El sistema opera entregando una corriente de aire limpio constante al paciente y otra que porta el olor con el que se quiere estimular por periodos de tiempo determinados. La corriente olorizada se genera haciendo pasar una corriente de aire limpio por un recipiente que contiene una solución líquida o muestra sólida. El olfatómetro puede manejar hasta 4 odorantes simultáneamente. La selección del odorante se hace activando una serie de electroválvulas que son controladas por un microcontrolador que recibe las señales desde una interfaz gráfica de PC.

La corriente de aire limpio y olorizado se mezclan justo antes de ser presentadas al paciente permitiendo controlar la razón de dilución entre ambos. Los materiales utilizados para la fabricación del instrumento tienen una alta resistencia química y a crecimientos bacterianos con la finalidad de reducir el mantenimiento y evitar su contaminación a lo largo del tiempo. Por ello, para la sección de aire olorizado se utilizaron tuberías de politetrafluoroetileno (PTFE) de la empresa Supelco y Tygon® de la empresa Saint-Gobain. En la Figura 2 se muestra el diagrama neumático del instrumento.



CCADET
CENTRO DE CIENCIAS APLICADAS
Y DESARROLLO TECNOLÓGICO

SOMI CONGRESO
DE INSTRUMENTACIÓN



XXIX Edición, Puerto Vallarta, Jalisco, México, octubre 2014

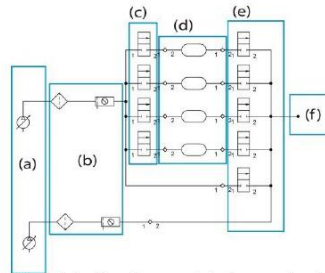


Figura 2. Diagrama a neumático del olfatómetro (a) Fuente de aire (b) Filtrado y regulador de seguridad (c) Válvulas de selección (d) Contenedores (e) Mezclador (f) Sección de administración (salida).

2.2.1 Fuente de Aire

La fuente de aire se compone por dos bombas de DC que proporcionan un flujo máximo de 12 LPM (AIRPON, Mod. D2028), una bomba se encarga de alimentar la corriente de aire limpio y la otra el aire olorizado facilitando el control independiente.

2.2.2 Tratamiento de Aire

En esta etapa se filtran los componentes que pudiera contener el aire ambiental para poder entregar un aire libre de olores, esto se logra mediante un filtro de partículas con regulador y manómetro (FESTO, Mod. MS4-LFR-1/4-D5-CRM-RG) junto con un filtro de carbón activado (FESTO, Mod. MS4-LFX-1/4-R). El regulador con manómetro sirve para obtener lecturas de la presión del sistema, además de funcionar como una medida de seguridad para evitar la apertura de las electroválvulas de los sistemas subsecuentes al evitar un aumento excesivo de presión, fijando la presión máxima en 2 bar.

2.2.3 Selección de odorantes

Esta etapa se encarga de seleccionar los odorantes que serán administrados mediante las señales de un microcontrolador (Texas Instruments, Mod. TMS320F28027) para la activación de una de las 4 electroválvulas (Cole Parmer, Modelo 98302-06) que permiten el ingreso de aire limpio a los contenedores de vidrio con los odorantes. Siendo la restricción contar con boquilla de diámetro 2.5 cm, para permitir la conexión por los tapones de goma que tienen los orificios para introducir: la tubería que porta el aire limpio, la tubería de lectura del sensor de presión y la válvula check de salida. El volumen máximo sugerido para los contenedores depende del tipo de muestra, de esta forma, para muestras líquidas se manejarán 130 ml mientras que para muestras sólidas es de 70ml. En el caso de las soluciones líquidas el contenedor funciona como burbujeador. Para un óptimo desempeño del dispositivo se debe tener un volumen libre entre 30 y 50 ml respectivamente. La salida de los contenedores de muestra se unen en un "mezclador" 5x2 normalmente cerrado (NResearch Inc., Mod 161T101) que permite seleccionar el odorante que se mezclara con la corriente de aire limpio para ser presentada al paciente. Todos los elementos que mantienen contacto con las corrientes del aire se encuentran fabricados de PTFE para evitar contaminación de olores.

2.2.4 Control de flujo

En esta sección se controla el flujo de aire que se desea, esto se hace mediante control proporcional tomando como entrada la señal de los sensores de presión (Epcos, Mod. ASR 0.400

V1 TN H30) en los contenedores y regulando mediante PWM la activación de la bomba de aire. La relación entre presión y flujo de salida después de la válvula check se muestra en la ecuación (1) donde q es el flujo de salida, N_2 la constante de proporcionalidad, C_v el coeficiente de flujo de la válvula, p_1 la presión de entrada, G_g la gravedad específica del gas, T_1 la temperatura de entrada y Δp la diferencia de presión entre entrada y salida. En la Figura 3 se muestra un diagrama del sistema que representa la ecuación [9].

$$q = \left(1 - \frac{2\Delta p}{3p_1}\right) N_2 C_v p_1 \sqrt{\frac{\Delta p}{p_1 G_g T_1}} \quad (1)$$

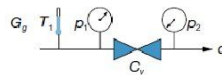


Figura 3. Diagrama de prueba para la obtención de la ecuación (1)

Las válvulas check utilizadas (Cole Parmer, Mod. 01355-00) cuentan con un $C_v = 0.014$ y una presión de apertura de 0.069 bar, la constante $N_2 = 6950$ al introducir presiones en [bar] y temperatura en [K] para entregar q en [std L/min]. Siendo el rango de operación de los sensores de 0 bar a 0.4 bar y entregando una salida proporcional entre 0 V y 1 V se introducen al ADC de 10 bits del microcontrolador utilizado. Resultando la grafica que se muestra en la Figura 4.

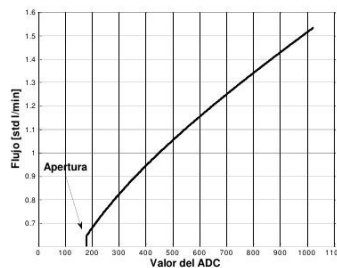


Figura 4. Grafica que muestra la relación del valor entregado por el ADC y el Flujo a la salida del sistema

Las reglas de control implementadas se basan en un control proporcional representadas por las ecuaciones (2) y (3) siendo P_o la presión objetivo del sensor y P la lectura actual del sensor ambos valores expresados por el valor en bits del ADC.

$$\Delta p = P_o - P \quad (2)$$

$$\%PWM = 5 + \frac{P_o}{100} + \frac{\Delta p}{1024/P_o} \quad \text{para } \Delta p > 0 \quad (3)$$

$$\text{sí } \%PWM > 100 \Rightarrow \%PWM = 100$$



CCADET
CENTRO DE CIENCIAS APLICADAS
Y DESARROLLO TECNOLÓGICO

SOMI CONGRESO
DE
INSTRUMENTACIÓN



XXIX Edición, Puerto Vallarta, Jalisco, México, octubre 2014

2.2.5 Sección de administración

En esta sección se toma la salida del sistema y se adapta según las necesidades de la prueba y del paciente. Un ejemplo es la utilización de una mascarilla o cánula para oxígeno. Se debe tomar en cuenta que a mayor longitud de la tubería utilizada mayor será el retraso del estímulo y se recomienda utilizar diámetros pequeños pero no menores a 1mm para no afectar el flujo de salida.

2.3 Diseño electrónico

Se utiliza una tarjeta de desarrollo Launchpad XL con un microcontrolador TMS320F28027 para controlar el dispositivo, manejar la interfaz con la PC por medio del puerto USB y obtener la información de los sensores de presión para determinar el flujo. La interfaz de potencia entre microcontrolador y las válvulas y bombas se hace mediante optocopladores (Fairchild Semiconductor, Mod. 4N35) y transistores MOSFET (ST Microelectronics, Mod. BUZ11). El control de las bombas de aire se hace mediante el módulo de PWM del microcontrolador a una frecuencia de 100Hz.

En cuanto a la obtención de señales de los sensores de presión, se utilizó una referencia fija de 1V al ADC del microcontrolador ya que los sensores entregan una señal proporcional a la presión de 0 a 1V por lo que no es necesaria una etapa de adquisición más elaborada. Para acoplar los sensores con los contenedores se hace uso de una tubería de PTFE con diámetro interno 1mm.

2.4 Adquisición de la señal de EEG

Con la finalidad de facilitar la adquisición de la señal de EEG se utilizó el sistema de adquisición EPOC de Emotiv que permite la captura simultánea de 14 canales del sistema 10-20 (AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4) además de contar con un sistema de medición de impedancias en cada electrodo para asegurar un registro confiable.

La aplicación utilizada para el registro es el TestBench™ incluido en SDK del sistema, el cual permite recibir por medio de un puerto serie marcadores para integrarlos junto al registro de EEG.

2.5 Interfaz de Usuario

Para facilitar el manejo del dispositivo se programó una interfaz gráfica en Matlab. Esta interfaz permite seleccionar el flujo de aire olorizado, el flujo de aire limpio, editar las etiquetas de los cuatro odorantes, cuál de los cuatro odorantes será administrado y el tiempo del estímulo. También se selecciona el puerto COM asignado al olfactómetro y el puerto para enviar los marcadores para el registro de EEG. Los marcadores envían la información de la etiqueta del odorante, duración del estímulo, inicio del estímulo y el flujo de aire limpio y olorizado.

3 RESULTADOS

Como principal resultado se tiene el prototipo de olfactómetro con todas sus etapas neumáticas ensambladas y circuitos electrónicos en PCB como se muestra en la Figura 5. En la salida del sistema se obtuvo un intervalo de trabajo de 0.6 a 1.5 l/min para aire olorizado y de 0.6 a 3 l/min para aire limpio. Y el intercambio de muestras y limpieza del sistema resultaron sencillos y no tomo más de 15 minutos incluyendo la preparación de las muestras olfativas.

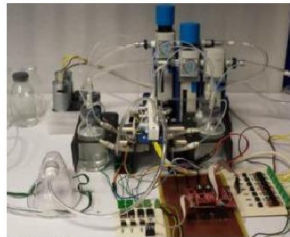


Figura 5. Sistema completo del olfactómetro (Al fondo Fuente de aire y tratamiento de aire, en la parte media selección de odorantes, al frente etapa de control y mascarilla para la administración).

Se realizaron pruebas para evaluar el comportamiento de las reglas de control otorgando los siguientes resultados para la corriente de aire olorizado: Flujo objetivo de 1 std l/min, Flujo medio de 1.01 std l/min, Desviación estándar de 0.025 std l/min, Flujo pico de 1.06 std l/min y Flujo mínimo de 0.88 std l/min (Figura 6a). Estadísticamente tras un barrido del rango de trabajo que va de 0.6 a 1.5 std l/min se obtuvo un error en el flujo medio de $\pm 1\%$ con una desviación estándar de 3%. Para la corriente de aire limpio se caracterizó el sistema haciendo una lectura del manómetro a distintos porcentajes del ciclo de trabajo de la bomba y aplicando la ecuación (1) con las mismas constantes al estar conectada la salida al mismo modelo de válvula check se obtiene el flujo mostrado en la Figura 6b.

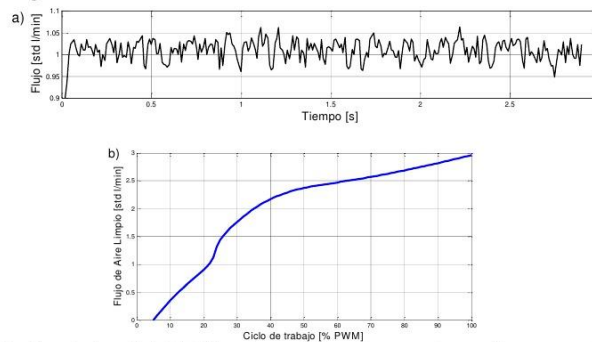


Figura 6. a) Gráfica de la señal del flujo entregada por el sensor de presión para un flujo objetivo de 1 std l/min.

b) Caracterización de flujo para distintos ciclos de trabajo en la bomba de aire

Se hizo una prueba para verificar el funcionamiento del dispositivo en el cual se presentó un olor a un sujeto de prueba en distintas razones de dilución desde 1:1 hasta 1:5 (olorizado: limpio) siendo el flujo de aire olorizado 0.6 l/min para una solución de 1 ml de esencia de lima y 99 ml de agua, se utilizó una mascarilla y posteriormente una cánula de oxígeno desechables. El registro de las señales de EEG se realizó sin problema alguno y se recibieron correctamente las cadenas de los marcadores con los datos de cada estimulación. El ajuste en los flujos de aire olorizado y aire limpio permiten regular la intensidad del olor y éstas serán definidas mediante la interfaz de usuario según el tipo de prueba que se realice. Durante las pruebas se observó un olor a plástico al utilizar



CCADET
CENTRO DE CIENCIAS APLICADAS
Y DESARROLLO TECNOLÓGICO

SOMI CONGRESO
DE
INSTRUMENTACIÓN



XXIX Edición, Puerto Vallarta, Jalisco, México, octubre 2014

la cánula y cierta interferencia al exhalar en el caso de utilizar la mascarilla; situación que puede afectar la percepción del paciente en los estudios.

Los registros de EEG serán analizados posteriormente.

4 DISCUSION DE RESULTADOS

Observando los resultados y al comparar las características obtenidas con las de los dispositivos comerciales (Serie TO de Odournet) cuya desviación estándar del flujo es de 10% es posible mencionar que la regla de control utilizada (control proporcional) muestra un desempeño aceptable aunque puede ser mejorada estableciendo un control PID o incluso control neurodifuso. Igualmente, los rangos de la salida son aceptables para las pruebas que se pretende hacer, ya que no requieren de niveles mayores de dilución además de poder controlarse la concentración de los aromas mediante las soluciones que se introducen a los recipientes o la cantidad de muestra sólida. Durante las pruebas se pudo observar que el diseño de los contenedores fue el adecuado ya que permite una limpieza fácil e intercambio de los mismos siendo este tiempo no mayor a 10 minutos. En cuanto a la contaminación de las tuberías por olores solo se presentó en el espacio muerto de la válvula check que sale del contenedor de odorantes a la válvula del mezclador, este problema se solucionó poniendo agua bidestilada en contenedores limpios y haciendo funcionar el sistema con el flujo máximo durante 1 minuto para cada canal.

El sistema obtenido cumple con el objetivo de ser portable ya que las dimensiones del chasis final son de 30x50x30 cm y la fuente de alimentación consta de un adaptador de corriente alterna con salida de 12V a 4A para la interfaz de potencia y el puerto USB de la PC para la parte de control.

En la literatura se reportan trabajos similares al diseño sin embargo requieren de una fuente de aire comprimido externa, sus pesos oscilan de 3 a 12 kg sus dimensiones de 42x11x23 cm a 64x50x50 cm y tienen de 4 a 16 canales y sus métodos de control de flujo van desde reguladores manuales hasta controladores de flujo de masa [5,10,11,12]. Las principales ventajas del diseño propuesto sobre los olfactómetros comerciales y reportados en la literatura se encuentran en: La portabilidad ya que no se necesita ningún tipo de instalación especial, basta con tener una PC y una toma de corriente de 127V. Envío de marcadores a otras aplicaciones para la sincronización del registro. Y la facilidad de introducir una muestra sólida o líquida sin la necesidad de una preparación exhaustiva solamente estaría restringido el uso de polvos.

Los registros de EEG obtenidos no mostraron errores en la recepción de los marcadores y pudieron ser importados sin problemas a MATLAB para su análisis por lo cual se plantea la creación de un programa que segmente y organice de forma automática los registros de cada prueba para la creación de una base de datos.

Finalmente, para la corrección de los inconvenientes de la etapa de administración se plantea conseguir cánulas de un material que no presente el mencionado olor a plástico y en el caso de la mascarilla modificarla para tener libre la sección de la boca y hacer la exhalación por la misma.

5 CONCLUSIONES

Si bien la razón máxima de dilución que puede entregar el dispositivo es de 1:5 es suficiente para las pruebas de discriminación e identificación; en el caso de las pruebas de umbral de olfacción puede cubrirse esta deficiencia al introducir en los cuatro contenedores el mismo odorante en distintas concentraciones para lograr hacer un barrido más amplio. Respecto de los registros de EEG obtenidos, los marcadores facilitarían la segmentación, procesamiento y análisis de las señales; situación que permitirá extraer características que muestren la diferencia entre los potenciales generados por distintos olores y concentraciones.

El dispositivo cumple con las características mínimas para las pruebas de potenciales evocados que se realizarán posteriormente en el laboratorio e incluso puede ser fácilmente adaptado para funcionar en estudios de fMRI.



CCADET
CENTRO DE CIENCIAS APLICADAS
Y DESARROLLO TECNOLÓGICO

SOMI CONGRESO
DE
INSTRUMENTACIÓN



XXIX Edición, Puerto Vallarta, Jalisco, México, octubre 2014

AGRADECIMIENTOS

Los autores agradecen al Consejo Nacional de Ciencia y Tecnología (CONACYT) por el financiamiento y la beca de posgrado otorgada a Luis Fernando Valdez.

REFERENCIAS

- [1] Auffarth B. Understanding smell-the olfactory stimulus problem. *Neurosci Biobehav Rev.* 2013;37(8):1667–79.
- [2] Leer A, Smeets M a M, Bulsing PJ, et al. Odors eliciting fear: a conditioning approach to Idiopathic Environmental Intolerances. *J Behav Ther Exp Psychiatry.* 2011;42(2):240–9.
- [3] Barresi M, Ciarleo R, Giacoppo S, et al. Evaluation of olfactory dysfunction in neurodegenerative diseases. *J Neurol Sci.* 2012;323(1-2):16–24.
- [4] Martínez JC, Puebla JMM, Antolín JAJ. Patología de la olfacción. olfatometría. manejo de los problemas olfativos. In: *Libro virtual de formación en ORL.*; :1–14.
- [5] Sommer JU, Maboche W, Griebel M, et al. A mobile olfactometer for fMRI-studies. *J Neurosci Methods.* 2012;209(1):189–94.
- [6] Toledano A, Borromeo S, Luna G, et al. Objective Assessment of Olfactory Function Using Functional Magnetic Resonance Imaging. *Acta Otorrinolaringol (English Ed.* 2012;63(4):280–285.
- [7] Barresi M, Ciarleo R, Giacoppo S, et al. Evaluation of olfactory dysfunction in neurodegenerative diseases. *J Neurol Sci.* 2012;323(1-2):16–24.
- [8] Philpott CM, Bennett a, Murty GE. A brief history of olfaction and olfactometry. *J Laryngol Otol.* 2008;122(7):657–62.
- [9] Swagelok®, Valve Sizing Technical Bulletin Available at:
<http://www.swagelok.com/downloads/webcatalogs/EN/ms-06-84.pdf> (Accessed on: February 4, 2014)
- [10] Lundström JN, Gordon AR, Alden EC, Boesveldt S, Albrecht J. Methods for building an inexpensive computer-controlled olfactometer for temporally-precise experiments. *Int J Psychophysiol.* 2010;78(2):179–89.
- [11] Sezille C, Messaoudi B, Bertrand a, Jossain P, Thévenet M, Bensafi M. A portable experimental apparatus for human olfactory fMRI experiments. *J Neurosci Methods.* 2013;218(1):1–10.
- [12] J. Ng1, E. Evaes, D. Carpenter, C. Tang. An MR Compatible Olfactometer for Clinical Research Use. *Proc. Intl. Soc. Mag. Reson. Med.* 19. 2011;31(2):3659.