



Universidad Nacional Autónoma de México

Facultad de Estudios Superiores Aragón

TESIS

“Desarrollo de dispositivos de control hombre-máquina para un sistema computacional aumentativo y alternativo de comunicación”

Presenta:

Andrés Galicia Pérez

Para obtener el título de:

INGENIERO ELÉCTRICO ELECTRÓNICO

Asesor:

Dr. Ismael Díaz Rangel



Ciudad Nezahualcóyotl, Estado de México (FES Aragón), 2022.



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

Agradecimientos

Agradezco a Dios por haberme permitido llegar hasta este punto, también agradezco a mi tutor el Dr. Ismael Díaz Rangel por la paciencia, el apoyo y el consejo para llevar a cabo este proyecto. De igual forma agradezco a mi madre y a mi padre que en paz descansen, que con su esfuerzo lograron sacarme adelante, a mi hermano por ser mi gran apoyo. Gracias a mis tíos, tías y primos por el apoyo. Al grupo IDEA, a los revisores de tesis, también al proyecto PAPIIT y a la UNAM, en especial la FES Aragón por las experiencias vividas y los conocimientos adquiridos.

Investigación realizada gracias al Programa UNAM-PAPIIT IT103220.

ÍNDICE

| | |
|--|----|
| Agradecimientos..... | i |
| ÍNDICE..... | ii |
| Introducción..... | 1 |
| Objetivo general | 2 |
| Objetivos particulares | 2 |
| Actividades | 2 |
| Justificación | 3 |
| Antecedentes..... | 3 |
| Descripción del capitulado | 4 |
| Capítulo 2. Marco teórico..... | 5 |
| 2.1 Definición del SAAC | 5 |
| 2.1.1 Sistemas de comunicación..... | 5 |
| 2.1.2 Algunos ejemplos de SAAC..... | 7 |
| 2.2 Condiciones de personas que requieren de un SAAC | 10 |
| 2.2.1 Esclerosis lateral | 10 |
| 2.2.2 Esclerosis múltiple..... | 12 |
| 2.2.3 Traumatismos craneoencefálicos..... | 13 |
| 2.2.4 Parálisis cerebral..... | 14 |
| 2.2.5 Otros | 16 |
| 2.3 Métodos de interacción de los SAAC | 17 |
| 2.3.1 Pulsadores y conmutadores | 17 |
| 2.3.2 Emuladores de ratón | 18 |
| 2.3.3 Ratones adaptados y de bola..... | 20 |
| 2.3.4 Comunicadores de voz digitalizada..... | 20 |
| 2.3.5 Comunicadores de voz sintetizada | 21 |
| 2.3.6 Programas específicos para computadora..... | 23 |
| 2.3.7 Aplicaciones móviles..... | 24 |
| 2.3.8 Cámaras | 24 |
| 2.4 Microcomputadora Raspberry Pi..... | 25 |
| 2.4.1 Modelos | 26 |
| 2.4.2 Pines | 27 |
| 2.5 Arduino..... | 28 |

| | | |
|-------------|---|----|
| 2.5.1 | Arduino UNO | 29 |
| 2.5.2 | IDE de Arduino | 30 |
| 2.5.3 | Bibliotecas de Arduino | 31 |
| 2.6 | Microcontroladores..... | 32 |
| 2.6.1 | Microprocesadores..... | 33 |
| 2.7 | ATmega328P..... | 34 |
| 2.8 | Micrófonos | 35 |
| 2.8.1 | Tipos de micrófono..... | 35 |
| 2.9 | Amplificadores operacionales | 37 |
| 2.9.1 | LM386 | 37 |
| 2.10 | Componentes pasivos | 38 |
| 2.10.1 | Resistencia | 38 |
| 2.10.2 | Capacitores | 39 |
| 2.10.3 | Inductores | 39 |
| 2.11 | Software para diseño de PCB..... | 40 |
| 2.11.1 | Proteus | 40 |
| 2.11.2 | EAGLE | 41 |
| 2.12 | Software para diseño de modelos 3D | 42 |
| 2.12.1 | AutoCad..... | 42 |
| 2.12.2 | SolidWorks | 42 |
| 2.12.3 | Tinkercad..... | 43 |
| 2.12.4 | Ultimaker Cura | 43 |
| 2.13 | Interruptores | 44 |
| 2.13.1 | Interruptores | 44 |
| 2.13.2 | Pulsadores..... | 44 |
| 2.13.3 | Conmutadores..... | 45 |
| 2.13.4 | Microinterruptores..... | 45 |
| 2.13.5 | Relés | 46 |
| 2.14 | Acelerómetros y giroscopios | 47 |
| 2.14.1 | MPU6050 | 48 |
| Capítulo 3. | Desarrollo experimental..... | 52 |
| 3.1 | Módulos basados en interruptores mecánicos | 53 |
| 3.1.1 | Módulo para palma-muñeca de la mano..... | 53 |
| 3.1.2 | Módulo para activación por pulgar de la mano..... | 58 |

| | | |
|-------|--|-----|
| 3.1.3 | Módulo para activación por agarre de mano | 64 |
| 3.2 | Módulos basados en soplido..... | 69 |
| 3.2.1 | Diagrama de bloques | 70 |
| 3.2.2 | Diagrama esquemático | 70 |
| 3.2.3 | Diagrama de flujo | 74 |
| 3.2.4 | PCB de variante 1 | 78 |
| 3.2.5 | Modelo 3D y real de variante 1 | 79 |
| 3.2.6 | PCB de variante 2 | 82 |
| 3.2.7 | Modelo 3D y real de variante 2 | 84 |
| 3.3 | Módulo con acelerómetro..... | 86 |
| 3.3.1 | Diagrama de bloques | 87 |
| 3.3.2 | Diagrama esquemático | 87 |
| 3.3.3 | Diagrama de flujo | 89 |
| 3.3.4 | PCB..... | 97 |
| 3.3.5 | Modelo 3D y real..... | 99 |
| 4 | Pruebas y resultados | 103 |
| 4.1 | Pruebas de posibles prototipos | 103 |
| 4.1.1 | Módulo de activación por guiño..... | 103 |
| 4.1.2 | Módulo de activación por un dedo | 104 |
| 4.2 | Pruebas de construcción | 105 |
| 4.2.1 | Pruebas de construcción para el módulo de activación por palma-muñeca de la mano | 105 |
| 4.2.2 | Pruebas de construcción para módulo de activación por pulgar | 107 |
| 4.2.3 | Pruebas de construcción para módulo de activación por agarre..... | 109 |
| 4.2.4 | Pruebas de construcción para módulo basado en soplido..... | 112 |
| 4.2.3 | Pruebas de construcción para módulo con acelerómetro..... | 119 |
| 4.3 | Pruebas de programación..... | 122 |
| 4.3.1 | Pruebas de programación del sensor basado en soplido..... | 122 |
| 4.3.2 | Pruebas de programación del módulo con acelerómetro..... | 125 |
| 4.4 | Pruebas y resultados de módulos finales en el SAAC..... | 127 |
| 4.4.1 | Módulo de activación por palma-muñeca de la mano..... | 127 |
| 4.4.2 | Módulo de activación por pulgar..... | 128 |
| 4.4.3 | Módulo por activación de agarre..... | 129 |
| 4.4.4 | Módulo de activación por soplido | 131 |

| | |
|---|-----|
| 4.4.5 Módulo con acelerómetro | 132 |
| Conclusiones..... | 134 |
| Trabajo a futuro | 135 |
| Referencias | 136 |
| Fuentes de Imágenes..... | 139 |
| Fuentes de tablas..... | 143 |
| Anexo | 144 |
| A. Código del módulo basado en soplo (Dos variantes). | 144 |
| B. Código del módulo con acelerómetro..... | 145 |
| C. Lista de materiales y costos. | 147 |

Introducción

En el mundo existen 7.8 billones de personas (2021) (worldometer, 2021), las cuales están expuestas en todo momento a enfermedades, traumatismos, entre otras adversidades. Dentro de la población mundial existe un aproximado del 15% que vive con alguna discapacidad, unos 1000 millones de personas según el banco mundial (Banco Mundial, 2021). Como parte de los esfuerzos para tener una mejor sociedad se han utilizado diversos recursos y planes para integrar este sector de la población en la sociedad, muchos de estos planes son más sencillos y no necesitan mucha ayuda para que las personas con discapacidad puedan desempeñar sus labores, pero existe un grupo de personas que sufre de trastornos motores o neurodegenerativos que no les permiten comunicarse, un ejemplo de ello pueden ser las personas con esclerosis múltiple que 1 de cada 3000 personas la tiene, esta enfermedad va degenerando el sistema motor hasta no tener control de ningún músculo como el de la boca (Sanabria, 2010), otro ejemplo son las personas con esclerosis lateral amiotrófica la cual le da a 5 personas de cada 100,000 y va degenerando sistemas motores, sensitivos y cerebelosos (Porrás-Betancourt M, 2007), o parálisis cerebral que le da a 3 de cada 1000 personas donde el cerebro no se desarrolla bien y genera problemas motores y del habla (VALDEZ, 2007). Estas enfermedades, entre otras, en un punto avanzado o grave generan una discapacidad del habla y la escritura, en donde sin la ayuda no podrían comunicarse disminuyendo enormemente su calidad de vida, aunque sea en sus últimos momentos.

Como parte de la inclusión y el mejoramiento de la salud de las personas con estos padecimientos, se ha optado por desarrollar dispositivos que permitan la comunicación, ya sea de baja o alta tecnología, y a estos dispositivos se les conoce como Sistemas Aumentativos y Alternativos de la Comunicación (SAAC). Debido a este reto de inclusión y mejoramiento de la calidad de vida de las personas con este tipo de discapacidad, en la FES Aragón en el grupo IDEA se ha desarrollado un SAAC tecnológico de bajo costo, que tiene la capacidad de formar oraciones y reproducirlas a través de una voz sintetizada; este SAAC permite la comunicación de personas con discapacidades de comunicación y escritura, el elemento central es una microcomputadora Raspberry PI, lo cual permite tener un costo asequible; el sistema puede mostrar en cualquier pantalla o monitor una interfaz que consiste en una serie de plantillas, sobre las cuales se muestran pictogramas o un abecedario; un cursor se desplaza automáticamente, y el paciente para realizar la selección, presiona un botón cuando el cursor esta sobre el elemento de su interés, con ello es posible crear mensajes que pueden ser verbalizados (Martínez, 2020). El control es mediante un simple botón puede ser útil para algunos pacientes, pero difícil o imposible de operar para otros, en función del grado de limitación que se tenga en términos de control de movilidad. Por tal razón, este trabajo se enfoca a dotar al SAAC ya existente, con diversos módulos que pueden ser fácilmente conectados, y que permitan ser utilizados para casi cualquier paciente sin importar sus limitaciones motrices.

Objetivo general

Desarrollar módulos eléctricos-electrónicos que sirvan de interfaz hombre-máquina para la interacción entre pacientes y un sistema aumentativo y alternativo de comunicación computacional.

Objetivos particulares

- ◆ Producir señales de acoplamiento al SAAC con niveles lógicos compatibles con una microcomputadora.
- ◆ Desarrollar los dispositivos o módulos de interacción considerando un costo asequible.
- ◆ Crear módulos basados en interruptores mecánicos con acoplamientos para su colocación en diversas extremidades corporales.
- ◆ Crear módulos basados en el flujo de aire del tracto vocal.
- ◆ Crear un módulo basado en el uso de un acelerómetro/giroscopio con acoplamientos para su colocación en diversas extremidades corporales.

Actividades

- ◆ Investigar condiciones de los pacientes susceptibles a requerir el uso de un sistema de comunicación aumentativo y alternativo (SAAC).
- ◆ Investigar las opciones existentes para interacción hombre-máquina de los SAAC tecnológicos.
- ◆ Estudiar las características de la microcomputadora Raspberry Pi, particularmente de sus puertos.
- ◆ Conocer diversas opciones de interruptores.
- ◆ Diseñar carcasas para los módulos mediante herramientas de diseño 3D, que permitan el acoplamiento de interruptores para su colocación en diversas extremidades.
- ◆ Crear módulos que integren interruptores.
- ◆ Indagar las opciones para monitoreo de señales acústicas.
- ◆ Definir un método para ajustar sensibilidad y duración de señal de salida.
- ◆ Diseño de una carcasa y acoplamiento para el paciente.
- ◆ Crear PCB para cada propuesta, así como carcasas y acoplamientos para los pacientes.
- ◆ Crear prototipos de un módulo sensible al flujo de aire.
- ◆ Investigar los diversos tipos de acelerómetros/giroscopios, su funcionamiento y aplicación.
- ◆ Proponer el diseño de un módulo empleado acelerómetros/giroscopios, que tenga ajuste de sensibilidad y duración de señal de salida.

- ◆ Crear PCB, así como carcasas y acoplamientos para los pacientes.
- ◆ Crear un módulo empleando un acelerómetro y giroscopio con sus diferentes interfaces.

Justificación

Las personas que por una razón u otra han perdido o reducido su capacidad de comunicarse tanto verbalmente como por escrito, pueden ver severamente reducida su calidad de vida; pero no solo ellos, también el de las personas que tomar el rol de cuidarlos. En algunos casos, sus funciones cognitivas están intactas, pero su cuerpo ya no les permite comunicarse, es por ello que es importante, utilizar los conocimientos tecnológicos para tratar de ayudar a reducir esta lamentable situación. El problema de los dispositivos de apoyo considerados de alta tecnología es que su costo es bastante elevado, por lo que pocas personas tienen la capacidad económica para adquirir uno de ellos, ante esta situación, estudiantes y académicos de la FES Aragón trabajan en el desarrollo de un SAAC de alta tecnología, con la premisa de lograr un costo accesible, actualmente se cuenta con un prototipo que presenta una interfaz que permite la generación de mensajes de manera simple y eficaz, aunque su modo de interacción entre el posible paciente y el sistema está limitado a la pulsación de un botón, lo cual lo limita a ser utilizado por pacientes que no pueden escribir, pero que si tienen un grado de control en su mano como para manipularlo a voluntad; esto hace que el sistema sea un tanto restrictivo, y que la incorporación de otras formas de interacción puede permitir lograr el objetivo de su creación; es decir, ayudar a cualquier persona que requiera de un sistema de este tipo, casi sin importar sus limitaciones motrices o económicas.

Antecedentes

Como parte de los esfuerzos para poder tener un mundo más inclusivo la tecnología ha ayudado poco a poco a adaptar a las personas con discapacidades al mundo, he incluso mejorando su calidad de vida, aunque sea en los últimos momentos. En el Consumer Electronics Shows (CES) del 2021 se habló de esto, donde este año se abrió paso a exposiciones de artículos tecnológicos para mejorar la vida cotidiana de los usuarios centrada principalmente en la accesibilidad, desde audífonos hasta teclados braille y aplicaciones que guían a los usuarios con discapacidad visual. Dado a que en la población mundial el 15% de las personas tienen alguna discapacidad, abre un mercado de un billón de dólares y por el momento todos estos dispositivos son costosos (AI-Heeti, 2021). Los dispositivos que ayudan a las personas con discapacidad a poder comunicarse como lo es un SAAC (Sistema alternativo y aumentativo de la comunicación) son altamente costosas, y no solo el SAAC si no los dispositivos que se utilizan para controlarlos también lo son, un ejemplo de ellos se puede encontrar en la página de *Inclusive Technology* que es una empresa británica que vende SAAC y diversos controles para los mismos, un ejemplo de estos es: *IntegraMouse Plus* que es un dispositivo que detecta el soplo y succión además de ser una palanca de direcciones controlable por la boca y sirve como mouse para una computadora, no obstante su precio es muy caro dando un costo de aproximadamente 2000 euros más envío que es un precio poco asequible en Latinoamérica (Inclusive Technology, 2021). Otro ejemplo es el *Quha Zono Mause* que es un mouse controlado por un giroscopio que se activa por un leve movimiento, de igual forma cuesta aproximadamente 1000 euros siendo un costo poco asequible en

Latinoamérica (Inclusive Technology, 2021). Además, tienen diversos interruptores, pero con un alto precio, de igual forma tienen SAAC tecnológicos y con menor tecnología.

Como parte de la solución a este problema se toma el SAAC desarrollado en la FES Aragón para abaratar costos, el cual está programado en Python y está cargado en una Raspberry Pi 4, que es mucho más barato que cualquier dispositivo de las empresas inglesas de tecnología inclusiva (Martínez, 2020). El modelo del SAAC no tenía un control adaptado a los pacientes y los existentes eran muy caros, además de que está diseñado para solo funcionar con una lógica inversa que cuando recibe un nivel bajo activa la señal en el SAAC.

Descripción del capitulado

Capítulo 1 Introducción. Se detallan los objetivos generales del proyecto, así como los objetivos particulares, la justificación del trabajo donde se dice porque se realizó este trabajo y con qué fin, los antecedentes que dan un panorama de prototipos parecidos que fueron hechos.

Capítulo 2 Marco Teórico. Se muestra un panorama de conocimientos general para el entendimiento del trabajo, las definiciones del SAAC y condiciones de las personas a usar un SAAC, microcontrolador, software utilizado en el trabajo, sensores, amplificadores, entre otros conocimientos de electrónica.

Capítulo 3 Desarrollo Experimental. En este capítulo se muestra la metodología que se siguió para desarrollar los diferentes dispositivos, detallando el procedimiento de creación como: imágenes de los modelos 3D, modelos impresos, programación, esquemas de bloques, diagramas de flujo, y desarrollo de circuitos.

Capítulo 4 Pruebas y Resultados. En este capítulo se muestran los prototipos de los dispositivos, sus errores, modelos que no fueron funcionales, adaptaciones y cambios que se hicieron para luego probarlos. También muestra los resultados de los dispositivos en funcionamiento con el SAAC, así como la adaptación al cuerpo.

Se continúa proporcionando las conclusiones y las mejoras a realizar como trabajo a futuro; por último, se indican las referencias tomadas y los anexos del trabajo.

Capítulo 2. Marco teórico

En este apartado se proporciona información relativa a los SAAC, personas a quienes están dirigidos, así como la información teórica y técnica referente a los diversos dispositivos considerados para el desarrollo experimental de los diversos módulos creados para satisfacer los objetivos de este trabajo.

2.1 Definición del SAAC

Sistemas Alternativos y Aumentativos de Comunicación o también conocidos como (SAAC), son: instrumentos que aumentan la comunicación de personas, con alteraciones diversas que no les permiten comunicarse en algún lenguaje, y cuyo objetivo es la comunicación mediante procedimientos específicos que introducen un conjunto estructurado de códigos no vocales necesitados o no de algún soporte físico, los cuales, mediante un funcionamiento específico permiten funciones de representación de códigos o algún lenguaje que les permiten llevar a cabo actos de comunicación (funcional, espontánea y generalizable), por si solos, o en conjunción con 7 códigos vocales, o como apoyo parcial a los mismos o en conjunción con otros códigos no vocales (Galván, 2015).

También llamados *Augmentative and Alternative Communication* (AAC) en inglés, que simplificando su definición son: una gama de estrategias y herramientas para ayudar a las personas que tienen problemas con el habla. Un ejemplo de estos dispositivos son tableros de cartas, imágenes o sofisticados sistemas basados en computadoras.

Los sistemas AAC ayudan a las personas a comunicarse de la manera más eficaz posible, en tantas situaciones como sea posible (Matters, 2020).

2.1.1 Sistemas de comunicación

Los sistemas de comunicación pueden clasificarse según precisen o no de algún soporte técnico para su expresión, es este caso los dividiremos en dos sistemas, con ayuda y sistemas sin ayuda: (González, 2003)

- ◆ Los sistemas de comunicación sin ayuda posibilitan las diferentes formas de intercambiar la información usando el cuerpo, en vez de algún tipo de ayuda o herramienta externa. Por ejemplo, las personas que son sordas y usan gestos o lenguaje de signos para comunicarse, lo hacen con sus manos, las expresiones de su cara y a menudo con la manera de estar de pie o de situar su cuerpo. El aprendizaje procedimental de estos sistemas es principalmente: la Dactilología, la Lengua de Signos, el Bimodal y la Palabra Complementada (González, 2003).
- ◆ Los sistemas de comunicación con ayuda son posibles gracias al uso de ayudas o herramientas externas al cuerpo para la comunicación. Estas ayudas permiten una comunicación básica como preguntar, hablar sobre sentimientos, necesidades, tu estado de salud y contar o que te cuenten las cosas que han pasado durante el día. Las herramientas que se usan más frecuentemente son: papel o cartulina, una carpeta o un libro. Éstos tienen dibujos, letras o palabras escritas en ellos con los cuales se pueden elaborar los tableros o plantillas de comunicación. También hay dispositivos

electrónicos que pueden decir o imprimir los mensajes que una persona selecciona o crea. Algunos son muy simples y otros muy sofisticados llamados de alta tecnología (González, 2003).

En la ilustración 2.1 se muestra un esquema de los sistemas de comunicación.

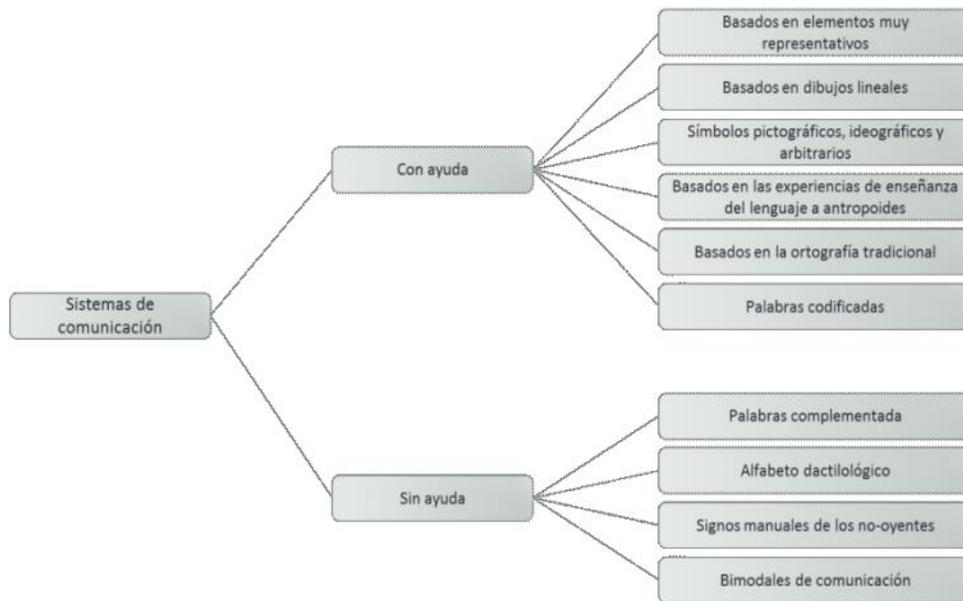


Ilustración 2.1 Esquema: clasificación de los sistemas de comunicación.

Como se explicó en los anteriores párrafos en este caso nos enfocaremos en los sistemas de comunicación con ayuda dado el contexto de este trabajo.

Tipos de Sistemas de Comunicación con Ayuda

Dentro de los sistemas de comunicación con ayuda podemos destacar varios sistemas tales como:

- ◆ Sistema BLISS, que son símbolos gráfico-visuales que representan significados. Estos símbolos se combinan de diversas maneras formando así nuevos significados, con lo que se crea un sistema complejo capaz de expresar conceptos diferentes (González, 2003).

Creado por Charles Bliss (1965) y utilizando como sistema de comunicación para personas con deficiencia motora desde 1971 (Lic. Falcón, 2011).

- ◆ Sistema SPC, que es un sistema de comunicación no oral basado en símbolos pictográficos con gran sencillez y transparencia de los pictogramas que se usan (González, 2003).

El Picture Communication Symbols (PCS, conocido en nuestro país como SPC, Símbolos Pictográficos de Comunicación), desarrollado por Mayer Jonson (1981) (Lic. Falcón, 2011).

- ◆ Sistema Minspeak, que es un sistema de comunicación aumentativa diseñado para una serie de comunicadores con voz. Se basa en la misma idea que los jeroglíficos:

un mismo dibujo puede tener uno u otro significado dependiendo qué rasgo se destaque de él o con cuál otro se combine (González, 2003).

- ◆ Sistemas basados en elementos muy representativos: estos elementos son los objetos, miniaturas, fotos, etc., a que se aludía en la clasificación anterior, y están especialmente recomendados para personas con dificultades serias de comunicación y representación (Lic. Falcón, 2011).
- ◆ Sistemas basados en las experiencias de enseñanzas de lenguajes antropoides: esta categoría está representada principalmente por dos programas de enseñanza; uno es el Non-Slip (Non-speech language initiation program), desarrollado por Carrier y Peak (1975) el objetivo del programa es la enseñanza de habilidades conceptuales previas para la adquisición de la estructura sintáctica de la frase, el otro es el programa de Deich y Hodges (1977) que fue diseñado para permitirle facilitar a un grupo de niños con grave retraso intelectual un sistema funcional de comunicación (Lic. Falcón, 2011).
- ◆ Sistemas basados en la ortografía tradicional: en este grupo los elementos de representación, los cuales, son el alfabeto escrito, es decir, sílabas, palabras o frases que se disponen en el soporte de comunicación. En estos casos, los usuarios deben conocer perfectamente procesos de lectura y escritura. Los sistemas que más posibilidades ofrecen, ya que, por un lado, en poco espacio, donde se puede desplegar un tablero de comunicación, donde se tienen todos los elementos para que la comunicación sea completa en todo momento. Además, es un sistema utilizado comúnmente por la mayoría de las personas hablantes (Lic. Falcón, 2011).
- ◆ Sistemas que utilizan palabras codificadas: es el caso del sistema Braille, en el que los símbolos Braille deben estar dispuestos en el soporte de comunicación para que el usuario los pueda utilizar; este sistema es especialmente indicado para personas con problemas de visión. Otro sistema incluido en esta categoría es el Morse; a veces se utiliza para personas con graves dificultades motoras y alto nivel cognitivo como acceso a sus ayudas técnicas, por ejemplo, al ordenador; es estos casos, las ayudas técnicas deben estar preparadas para traducir las señales Morse utilizadas por el usuario en letras del alfabeto, para que el mensaje se convierta en escritura y pueda ser entendido por los interlocutores (Lic. Falcón, 2011).

2.1.2 Algunos ejemplos de SAAC

Dependiendo del tipo de discapacidad que tenga el sujeto, de las características del contexto y de las indicaciones para su uso, el soporte que se utilizará para el SAAC y los sistemas de control elegidos, será de un tipo u otro:

- ◆ Comunicadores (dispositivos de lenguaje grabado):

Son aparatos que emiten sonido y están destinados a personas con ausencia de habla. Estos comunicadores han supuesto un verdadero avance en el mundo de los SAAC.

Existen dos tipos de comunicadores según el tipo de voz saliente del aparato. Estos son: comunicadores con voz digitalizada y comunicadores con voz sintetizada (González, 2003).

- Comunicadores de voz Digitalizada:

El habla digitalizada, es un registro de voz humana, realizada a través de un digitalizador y que permanece grabada en la memoria de un comunicador o del ordenador. Este tipo de habla tiene la ventaja de su calidad de sonido, así como la de poder adecuarse al tipo de usuario que va a utilizarla, teniendo en cuenta el sexo, edad e idioma. De esta manera, cuando se usa un comunicador que tenga esta prestación, se puede grabar en cada casilla una palabra o mensaje, que además puede contener expresiones familiares. El hecho de poder escuchar el mensaje cuando se elige un signo pictográfico aumenta considerablemente las posibilidades comunicativas de las personas sin habla (González, 2003).

Un ejemplo de estos sistemas de comunicación con voz digital se muestra en la ilustración 2.2.

| MODELO | IMAGEN |
|---|--|
| <p>ALPHATALKER: dispositivo ligero y portátil de comunicación.</p> |  |
| <p>CHAT BOX: ofrece una posibilidad útil y sencilla de comunicarse.</p> <p>SIDE KICK: es un comunicador de bolsillo, ligero, de poco peso y tamaño.</p> |  |
| <p>TECH 8: comunicador digital de ocho casillas.</p> |  |

Ilustración 2.2. Modelos de comunicadores de voz digital.

- Comunicadores de voz Sintetizada:

El habla sintetizada permite la generación artificial del habla a partir de un texto, controlando la lectura letra a letra o por emisión continua. No se basa en el registro de la voz humana, por lo que la voz resulta más robotizada. Su calidad de sonido, por tanto, es inferior a la del digitalizador. El sintetizador de voz no tiene limitación de memoria, por lo que permite usar un vocabulario muy amplio, así

como que el usuario pueda comunicarse en diferentes situaciones y contextos, siempre y cuando sepa escribir. Este tipo de habla es muy útil para personas ciegas, así como para aquellas personas que tengan problemas con la lectura y escritura, ya que el sintetizador de voz lee en voz alta los textos escritos. Algunos sintetizadores permiten adecuar el tipo de voz (masculina, femenina), la velocidad del habla, el volumen, etcétera (González, 2003).

En la ilustración 2.3, se muestra algunos modelos que utilizan la voz sintetizada.

| MODELO | IMAGEN |
|---|---|
| DELTATALKER: comunicador portátil de 128 casillas. |  |
| LIGHTWRITER: permite una natural comunicación frente a frente. |  |
| PATH FINDER: es un poderoso comunicador con teclas ya adaptado para la comunicación. |  |

Ilustración 2.3. Modelos de comunicadores de voz sintetizada.

◆ **Programas Específicos para Ordenador:**

Con el paso del tiempo la tecnología avanza, lo cual, nos permite tener ordenadores y portátiles ligeros, pequeños, resistentes y con mayor autonomía de funcionamiento. Estos modelos tienen una aplicación como comunicadores para personas con acceso fácil a pequeños teclados. Simplemente escribiendo los mensajes con cualquier procesador de texto y activando la tarjeta de voz, o en otro caso con programación y procesamiento de IA (Inteligencia Artificial) puede tener un comunicador con síntesis de voz, que a su vez le da prestaciones para otras funciones (González, 2003).

Un ejemplo de ello se muestra en la ilustración 2.4. que muestra un SAAC digital con una interfaz hombre-máquina para su control.



Ilustración 2.4. SAAC digital de computadora.

2.2 Condiciones de personas que requieren de un SAAC

Los sistemas SAAC son usados como estrategias alternativas a la comunicación oral o como apoyo o ayuda al habla del individuo para potenciar su expresión, en los casos de personas que, por causas congénitas o adquiridas, no pueden usar el habla para comunicarse. Abren un abanico de posibilidades para personas con diferentes déficits en lo que respecta a la comunicación, permitiendo atender a las necesidades y favoreciendo la interacción, fundamental para la vida en sociedad; fomentando que las personas en situación de discapacidad, en este caso los niños con parálisis cerebral, puedan tener una mejor calidad de vida o alguna otra persona que tenga un déficit para comunicarse (TOSCANO CASTILLO, 2016).

2.2.1 Esclerosis lateral

La Esclerosis Lateral Amiotrófica (ELA) es una enfermedad progresiva y de mal pronóstico que provoca debilidad de los músculos voluntarios. La ELA tiene una incidencia de 1 a 3 casos por cada 10,000 personas. Esta enfermedad tiene más ocurrencia en los hombres y la edad de inicio es en promedio a los 56 años, aunque el rango oscila entre los 20 a los 80 años. Usualmente es una enfermedad de aparición esporádica, pero el 10% de los casos puede ser de carácter familiar, con una herencia autosómica dominante.

Lamentablemente la causa de la ELA es aún desconocida, pero se ha logrado entender su patogénesis, principalmente en el área genética debido a mutaciones cromosómicas (Sanabria, 2010).

Los síntomas que conlleva la ELA se caracterizan clínicamente por una combinación de hallazgos de neurona motora superior e inferior. La ELA puede ser una enfermedad tan sutil que es posible que se pasen por alto los síntomas.

Los síntomas de neurona motora superior dada por la ELA incluyen enlentecimiento del habla, espasticidad, hiperreflexia, signo de Hoffman o de Babinski.

Los síntomas de neurona motora inferior dada por la ELA incluyen atrofia, fasciculaciones y debilidad.

En algunos casos, los síntomas iniciales afectan una de las piernas y los pacientes experimentan dificultad para caminar o correr o se dan cuenta de que se tropiezan o tambalean más a menudo. También se puede ver los efectos de la enfermedad al principio en una mano o brazo, esta sucede cuando quieren hacer tareas sencillas que requieren destreza manual como escribir. En estos casos para diagnosticar la ELA se debe de tener los dos síntomas de neurona superior e inferior motora con daño y estas no deben ser atribuida por otras causas físicas.

Aunque la secuencia de los síntomas y la tasa de progresión de la enfermedad varían de persona a persona, eventualmente los pacientes no podrán pararse o caminar, acostarse o levantarse de la cama solos, o usar sus manos y brazos que conlleva a una pérdida de la comunicación.

En las etapas posteriores de la enfermedad, los pacientes tienen dificultad para respirar porque los músculos del sistema respiratorio se debilitan. Eventualmente los pacientes perderán su capacidad de respirar por si solos y estos dependerán de un respirador artificial para sobrevivir. La mayoría de los pacientes mueren de fallo respiratorio entre 3 y 5 años.

Algunos pacientes muestran el síndrome puro de neurona superior motora, el cual se le conoce como esclerosis lateral primaria o se presentan también puramente con el síndrome de neurona inferior motora o atrofia muscular progresiva. Estas patologías con el tiempo tienden a ser esclerosis lateral amiotrófica.

Es necesario saber que en ningún momento se afectan las facultades intelectuales ni los órganos de los sentidos, tampoco existen afectaciones a los órganos sexuales ni a los esfínteres. Debido a los síntomas los pacientes pasan por un estado de depresión ya que su intelecto no es afectado (Sanabria, 2010).

Un caso muy conocido de esta enfermedad es del científico Stephen Hawking quien perdió la habilidad de comunicarse por medio del habla y utilizo un SAAC para poder comunicarse como se muestra en la ilustración 2.5.



Ilustración 2.5. Stephen Hawking.

2.2.2 Esclerosis múltiple

La Esclerosis Múltiple (EM) es considerada una enfermedad autoinmune, crónica, inflamatoria, desmielinizante (afección que provoca un daño en la cubierta protectora que rodea las fibras nerviosas del cerebro, los nervios ópticos y la medula espinal) del Sistema Nervioso Central (SNC), principalmente se presenta en individuos genéticamente susceptibles y que involucra a factores inmunológicos como anticuerpos, complemento y mediadores de respuesta inmune innata.

Es considerada como una enfermedad desmielinizante inflamatorias idiopáticas, y construye una de las causas más frecuentes de discapacidad neurológica en los jóvenes adultos. El 50% de los pacientes es incapaz de caminar a los 15 años después del inicio de la enfermedad.

Los síntomas que conlleva la EM son muy variados en su forma de presentación, gravedad y duración, pero coinciden en que estas manifestaciones se asocian a afección de tractos. Las manifestaciones clínicas más frecuentes son déficit de motores, sensitivos y cerebelosos; afección de nervios craneales, alteraciones autonómicas y psiquiátricas, como se muestra en la tabla 2.1 donde se desglosan los síntomas y manifestaciones que conlleva la EM (Porrás-Betancourt M, 2007).

Tabla 2.1 Principales signos y síntomas de EM
Principales signos y síntomas de EM

| SÍNTOMAS | % |
|--------------------------|--------|
| Motor | |
| Debilidad muscular | 65-100 |
| Espasticidad | 73-100 |
| Alt. de reflejos | 62-98 |
| Sensitivas | |
| Vibración | 48-82 |
| Termoalgesia | 16-72 |
| Dolor | 11-37 |
| Signo de Lhermitte | 1-42 |
| Cerebeloso | |
| Ataxia | 37-78 |
| Temblor | 36-81 |
| Nistagmo | 54-73 |
| Disartria | 29-62 |
| Nervios craneales | |
| Disminución AV | 27-55 |
| Alt. de movs. oculares | 18-39 |
| NC V, VII, VIII | 5-52 |
| Signos bulbares | 9-49 |
| Vértigo | 7-27 |
| Autonómicos | |
| Disfunción vesical | 49-93 |
| Disfunción intestinal | 39-64 |
| Disfunción sexual | 33-59 |
| Sudoración y vasculares | 38-43 |
| Psiquiátricos | |
| Depresión | 8-55 |
| Euforia | 4-18 |
| Alt. cognitivas | 11-59 |
| Misceláneos | |
| Fatiga | 59-85 |

Modificado de Miller JR. Multiple Sclerosis. In: Rowland LP. Merritt's Neurology. 10th ed.

Los síndromes más frecuentes son la neuritis óptica, mielitis transversa, oftalmoplejía internuclear, neurálgica del trigémino, espasmo hemifacial y síndromes polisintomáticos sin alteraciones de la conciencia.

La evolución clínica la EM es variada describiéndose principalmente en cuatro tipos de evoluciones clínicas de la enfermedad estas son: Esclerosis Múltiple Recurrente-Remitente (EMRR), esclerosis múltiple primariamente-progresiva (EMPP), Esclerosis Múltiple Secundariamente-Progresiva (EMSP) y Esclerosis Múltiple Progresiva-Recurrentemente (EMPR). Donde se manifiestan diferentes evoluciones de la EM donde la EMRR es la que se manifiesta el 85 % de los casos y es con intervalos de crisis, las demás van variando en intensidad y momentos de crisis (Porrás-Betancourt M, 2007).

Dada las afectaciones de la esclerosis múltiple que afectan cognitivamente y también ocasiona trastornos del habla, afectaciones motoras que no les permiten una comunicación buena, para las personas que padezcan EM, son personas que podrían necesitar un SAAC para poder comunicarse correctamente y mejorar su calidad de vida.

2.2.3 Traumatismos craneoencefálicos

El Traumatismo Craneoencefálico (TCE), es una importante causa de muchas muertes y padecimientos en cualquier lugar del mundo.

Se define como una alteración en la función cerebral u otra evidencia de patología cerebral, causada por una fuerza extrema, la cual puede consistir en un impacto directo sobre el cráneo, aceleración o desaceleración rápida, penetración de un objeto (arma de fuego) u ondas de choque de una explosión, con la presencia de al menos uno de los siguientes elementos: alteración de la conciencia o amnesia, cambios neurológicos o neurofisiológicos, diagnóstico de fractura de cráneo o lesiones intracraneanas y ocurrencia de muerte como resultado final del traumatismo.

Debido a los parámetros, la gravedad de un TCE puede variar desde leve que es el cambio temporal al estado de conciencia a grave que es un periodo extendido de inconciencia o amnesia después de la lesión.

Muchas personas quedan discapacitadas después de un TCE. En México, el TCE ocupa la cuarta causa de muerte, contribuyendo los accidentes de tráfico con el 75% como se muestra en la ilustración 2.6 y el resto debido a muerte violentas, caídas y lesiones deportivas. En relación de género es 3:1 en hombres y de 15 a 45 años por lo general (Francisco Estrada Rojo, 2012).



Ilustración 2.6 TCE por accidente de tráfico.

Se pueden identificar dos tipos de lesiones en un TCE que son:

- ◆ **Lesión Primaria:** esta ocurre en el momento del impacto y no es reversible, incluye el desgarramiento de los trayectos de materia blanca, con tención focal, hematomas (intra y extra cerebral) y el edema difuso, que hace cambios a nivel celular y afecta la condición de la persona.
- ◆ **Lesión Secundaria:** que corresponde a los efectos tardíos, es un proceso potencialmente reversible, mediante una terapia adecuada. Involucra cambios funcionales, estructurales, celulares y moleculares que provocan daño neuronal (Francisco Estrada Rojo, 2012).

Las lesiones por TCE pueden llegar a provocar una pérdida del sistema del habla o motora, la cual dificultaría la comunicación, por lo que un SAAC podría ayudar en el desempeño de su mejoramiento o comunicación, ya sea ayudando a recuperarse o mejorando su calidad de vida.

2.2.4 Parálisis cerebral

La Parálisis Cerebral (PC), se considera un grupo de síndromes y no una enfermedad como tal, se manifiestan como problemas motores no evolutivos, aunque frecuentemente cambiantes. Son secundarios a lesiones o mal formaciones cerebrales originarias en las primeras etapas del desarrollo, que van desde los 3 a 5 años de vida, cuando el cerebro está inmaduro.

Se define como un trastorno aberrante en el control del movimiento y la postura, aparece tempranamente en la vida debido a una lesión, disfunción o malformación del Sistema Nervioso Central (SNC) y no es resultado de una enfermedad progresiva o degenerativa. Esta anomalía puede ocurrir en etapas pre, peri o postnatales.

Al principio de la enfermedad el diagnóstico será un déficit motor y usualmente el paciente se presenta al médico por retraso en el desarrollo psicomotor, o presentar otros síntomas de disfunción cerebral como retardo mental, retardo en el lenguaje, epilepsia y trastornos sensoriales (VALDEZ, 2007).

EL PC se da a nivel mundial en 2 a 2.5 por mil recién nacidos vivos, estos son resultados de algún traumatismo o un cambio cerebral prematuro.

Aunque aún no hay una clasificación oficial, estos son algunos tipos de parálisis cerebral:

- ◆ Hemiparesia espástica: Alteración motora unilateral. Puede ser congénita o adquirida. En los casos congénitos, cuando la lesión causal se produjo del nacimiento. Se presenta más frecuentemente en varones y habitualmente no son conocidas las causas, predomina en el lado izquierdo cerebral, cerca de dos tercios de los pacientes. La hemiparesia espástica congénita representa entre el 23 a 40% de todos los casos de PC, por lo tanto, es la forma más frecuente.
- ◆ Cuadriparesia espástica: Se presenta por lo regular en recién nacidos con bajo peso para su edad gestacional; es la más grave y representa entre un 10 a 40% de las PC. En la mayoría de los casos, son por problemas prenatales, aunque también se presenta por causas perinatales y postnatales. Las infecciones y las disgenesias cerebrales son las más comunes
- ◆ La paraparesia espástica: Se ha considerado que se presenta del 10 al 33% de los pacientes con PC. Las manifestaciones principales son en las extremidades inferiores más que en las superiores, que incluso no son afectadas en la mayoría de los casos, se ha asociado a la prematuridad y a complicaciones perinatales.
- ◆ Parálisis cerebral extrapiramidal: Hay una alteración en la coordinación del movimiento y en la regulación del tono muscular, que ocasiona posturas anormales y trastornos del movimiento.
- ◆ Parálisis cerebral atónica: La manifestación es de una hipotonía generalizada, con reflejos aumentados, con marcada debilidad de los miembros inferiores y mejor fuerza en los superiores. Se asocia a alteraciones prenatales.
- ◆ Parálisis cerebral atáxica: La mayoría de los casos de este tipo de PC se considera congénita, aunque las manifestaciones clínicas aparecen alrededor del año de edad, cuando el niño inicia la deambulacion.

De los síntomas que se presentan son principalmente de tipo motora, pero no es la única, también existen diferentes tipos de síntomas que conlleva tener una PC:

- ◆ Deficiencia mental: Se presenta en el 70% de los casos y esto sufren retraso mental.
- ◆ Epilepsia: El 46 % aproximadamente tiene este síntoma.
- ◆ Problemas del Lenguaje: Dentro de estas se dividen en 3:
 - A) Disfunción oromotora (producción de palabras).
 - B) Disfunción de procesamiento central (Trastornos del lenguaje).
 - C) Disfunción auditiva (hipoacusia).

Es posible que una misma persona pueda presentar más de una alteración del lenguaje y la disartria es muy común en pacientes con PC.

- ◆ Parálisis pseudobulbar: Como una manifestación frecuente que se manifiesta como una succión débil o inordinada, protrusión lingual o problemas de deglución y esta conlleva a una desnutrición.
- ◆ Trastornos visuales: Son frecuentes los trastornos motores oculares, como el estrabismo, nistagmus y problemas de refracción, ambliopía, defectos de los campos visuales e incluso ceguera, donde el 70% lo presenta.
- ◆ Problemas urinarios: donde no pueden controlar el esfínter por lo tanto se da una incontinencia.
- ◆ Problemas de conducta: Puede crear un déficit de atención e hiperactividad impulsiva.
- ◆ Otros: Sordera o problemas de sueño pueden ser algunos de estos casos (VALDEZ, 2007).

Pese a todas las complicaciones que llegan a tener los niños con PC, tener un sistema que permite darles una mejor calidad de vida como lo es un SAAC y que este adaptado a sus capacidades mejora su calidad de vida, como se muestra en la ilustración 2.7, donde se está utilizando un SAAC pictográfico.



Ilustración 2.7 Niño con parálisis cerebral con un SAAC pictográfico.

2.2.5 Otros

Existen otras discapacidades que afectan al sistema de comunicación, por lo tanto, el uso de un SAAC para mejorar la calidad de vida es necesario y a continuación se muestran algunos casos.

Uno de esos casos es por autismo que se entiende como un defecto de severidad variable en la interacción social recíproca y en la comunicación verbal y no verbal y en la actividad imaginativa, asociado a un repertorio escaso y repetitivo de actividades e intereses (Allen, 1988). Dado los casos más graves que presenta el autismo, el SAAC puede dar un apoyo en el tratamiento o mejoramiento en la calidad de vida como se muestra en la ilustración 2.8.



Ilustración 2.8 Niño con autismo SAAC

Entre los trastornos que afectan la comunicación, se puede tener algún síndrome que degenera el sistema motor o alguna capacidad cerebral, un accidente cardiovascular, algún defecto motor, ataxia, afasia, disartria, o alguna condición ya sea sola o combinada podría llegar a hacer una afectación en la comunicación (Shelat, 2020).

2.3 Métodos de interacción de los SAAC

Como se vio en los anteriores temas ya vimos que existen dos tipos de sistemas de comunicación, los cuales son: los sistemas sin ayuda y los sistemas con ayuda. Como también se vio existen diferentes tipos de SAAC y algunos necesitan de controladores para funcionar y estos van adaptados dependiendo el tipo de discapacidad de cada persona.

Las ayudas Técnicas son todas aquellos instrumentos mecánicos o electrónicos, diseñados para que la persona pueda comunicarse mejor, ya sea aumentando o bien supliendo su habla oral (González, 2003).

Dependiendo del tipo de señalización que tenga el sujeto, de las características del contexto y de las indicaciones para su uso, el soporte que se utilizará para el sistema o sistemas elegidos, será de un tipo u otro (González, 2003).

2.3.1 Pulsadores y conmutadores

Pulsadores y Conmutadores: son dispositivos que sirven para accionar cualquier aparato con batería o conectado a la red electrónica. Se trata de interruptores cuando fijan la función y pulsadores cuando sólo dan la orden de activación (González, 2003).

Un ejemplo de ellos se muestra en la ilustración 2.9 donde se ven varios modelos de pulsadores y conmutadores.

| MODELO | IMAGEN |
|--|--|
| <p>Conmutador de 1 función: es el pulsador sencillo más usado.</p> |  |
| <p>Conmutador de 2 funciones: misma función que el anterior pero con 2 funciones.</p> <p>Conmutador sopló-succión: se activa mediante el soplido o la succión.</p> |  |
| <p>Conmutador luminoso: conmutador que se ilumina para saber que se ha pulsado.</p> |  |
| <p>Conmutador multicontactos: funciona con un desplazamiento de la varilla en cualquier posición.</p> |  |

Ilustración 2.9 Conmutadores y pulsadores.

2.3.2 Emuladores de ratón

Emuladores de ratón: son todas aquellas ayudas, accesorios y técnicas que hacen la función de un ratón para acceder al ordenador o a cualquier comunicador. Estos accesorios emulan a los ratones convencionales (González, 2003).

Un ejemplo de ellos se muestra en la ilustración 2.10 e ilustración 2.11 donde se ven algunos tipos de emulador y su explicación.

| MODELO | IMAGEN |
|--|--|
| <p>Emulador de cabeza: emulador de ratón mediante señalización con la cabeza. Selección mediante soplo-succión.</p> |  |
| <p>Emulador óptico: emulador de ratón mediante unos sensores ópticos.</p> |  |
| <p>Emulador digital: emulador de ratón por pulsaciones.</p> |  |
| <p>Emulador joystick: emulador de ratón manejado mediante joystick.</p> |  |
| <p>Preciso boca: emulador de ratón manejado mediante joystick con la boca. Funciones mediante soplo-succión.</p> |  |
| <p>Preciso mentón: emulador de ratón que se manipula mediante el mentón o barbilla.</p> |  |

Ilustración 2.10 Modelos de simulador de ratón.

| | |
|---|--|
| <p>Emulador mediante barrido: emulador de ratón que se manipula a través de un conmutador o sensor óptico a través de barrido.</p> |  |
| <p>Carcasa para teclado: carcasa de metacrilato para pulsar las teclas sin pulsar las demás.</p> |  |

Ilustración 2.11 Modelos de simulador de ratón.

2.3.3 Ratones adaptados y de bola

Ratones adaptados y de bola: son ratones a los que se les puede adaptar cualquier conmutador para acceder al ordenador o a cualquier comunicador (González, 2003).

Un ejemplo de ellos se muestra en la ilustración 2.12 donde se ven algunos tipos de ratones que fueron adaptados para un uso más específico.

| MODELO | IMAGEN |
|---|--|
| Ratón adaptado: ratón con acceso a varios conmutadores. |  |
| Ratón adaptado y de bola: ratón con acceso a varios conmutadores y de fácil manejo mediante una bola. |  |
| Ratón de bola: permite un mayor control y precisión con ligeros movimientos de brazos y manos. |  |
| Ratón de bola: permite un mayor control y precisión con ligeros movimientos de brazos y manos. |  |

Ilustración 2.12 Modelos ratón adaptados y de bola.

2.3.4 Comunicadores de voz digitalizada

El habla digitalizada, es un registro de voz humana, realizada a través de un digitalizador y que permanece grabada en la memoria de un comunicador o del ordenador. Este tipo de habla tiene la ventaja de su calidad de sonido, así como la de poder adecuarse al tipo de usuario que va a utilizarla, teniendo en cuenta el sexo, edad e idioma.

De esta manera, cuando se usa un comunicador que tenga esta prestación, se puede grabar en cada casilla una palabra o mensaje, que además puede contener expresiones familiares. El hecho de poder escuchar el mensaje cuando se elige un signo pictográfico aumenta considerablemente las posibilidades comunicativas de las personas sin habla (González, 2003).

Los conmutadores de voz digitalizada son SAAC que ya están integrados a su manera de interacción. Algunos de estos modelos se muestran en la ilustración 2.13.

| MODELO | IMAGEN |
|---|---|
| <p>ALPHATALKER: dispositivo ligero y portátil de comunicación.</p> |  |
| <p>CHAT BOX: ofrece una posibilidad útil y sencilla de comunicarse.</p> <p>SIDE KICK: es un comunicador de bolsillo, ligero, de poco peso y tamaño.</p> |   |
| <p>TECH 8: comunicador digital de ocho casillas.</p> |  |
| <p>TECH 32: comunicador digital de treinta y dos casillas.</p> |  |

Ilustración 2.13 Comunicadores de voz digitalizada.

2.3.5 Comunicadores de voz sintetizada

El habla sintetizada permite la generación artificial del habla a partir de un texto, controlando la lectura letra a letra o por emisión continua. No se basa en el registro de la voz humana, por lo que la voz resulta más robotizada; por lo tanto, su calidad de sonido es inferior a la del comunicador de voz digitalizador.

El sintetizador de voz no tiene limitación de memoria, por lo que permite usar un vocabulario muy amplio, así como que el usuario pueda comunicarse en diferentes situaciones y contextos, siempre y cuando sepa escribir.

Este tipo de habla es muy útil para personas ciegas, así como para aquellas personas que tengan problemas con la lectura y escritura, ya que el sintetizador de voz lee en voz alta los textos escritos.

Algunos sintetizadores permiten adecuar el tipo de voz (masculina, femenina), la velocidad del habla, el volumen, etcétera (González, 2003).

En la ilustración 2.14 se muestran algunos comunicadores de voz sintetizada, donde se puede ver un SAAC con un panel táctil y más algunos otros que incorporan pictogramas, generando una voz artificial al ser seleccionados.

| MODELO | IMAGEN |
|---|--|
| DELTATALKER: comunicador portátil de 128 casillas. |  |
| LIGHTWRITER: permite una natural comunicación frente a frente. |  |
| PATH FINDER: es un poderoso comunicador con teclas ya adaptado para la comunicación. |  |
| SPRINGBOARD: es un comunicador flexible, táctil y fácil de usar. |  |
| VANTAGE: es un poderoso y dinámico comunicador portátil. |  |
| VANGUARD: comunicador con pantalla dinámica. |  |
| PDA ENDIKU: pequeño ordenador portátil con SAC incorporado. |  |

Ilustración 2.14 Comunicadores de voz sintetizada.

2.3.6 Programas específicos para computadora

Cada vez más se diseñan ordenadores portátiles ligeros, resistentes y con mayor autonomía de funcionamiento. Estos modelos, así como las agendas electrónicas sí tienen una aplicación como comunicadores para personas con acceso fácil a pequeños teclados. Simplemente escribiendo los mensajes con cualquier procesador de texto y activando la tarjeta de voz, puede tener un comunicador con síntesis de voz, que a su vez le da prestaciones para otras funciones (González, 2003).

La tecnología nos ha permitido cargar estos programas a aplicaciones para móviles o mini computadoras, para hacerlos más portables.

Algunos de estos programas se muestran en la ilustración 2.15 los cuales ya son programas antiguos.

| MODELO | IMAGEN |
|--|--|
| <p>WIVIK: teclado virtual o emulador de teclado con predicción de palabras.</p> |  |
| <p>ZOOMTEXT: es un aumentador de pantalla y lector con voz sintetizada de textos.</p> |  |
| <p>JAWS: lector de pantalla con voz sintetizada.</p> |  |
| <p>BOARDMAKER: programa para la producción de SPC.</p> |  |

Ilustración 2.15 Modelos de software para interacción en un ordenador.

2.3.7 Aplicaciones móviles

Dada la nueva era de la tecnología, se han desarrollado aplicaciones móviles que son un sistema aumentativo y alternativo de la comunicación, que se pueden instalar en una tableta o un smartphone y son accesibles vía internet por medio de una tienda de aplicaciones. Como los dispositivos móviles son en su mayoría touch, aprovechan el recurso para los pictogramas obteniendo una mejor accesibilidad, aunque los celulares son más baratos que un ordenador, para algunos casos no son factibles dado el contexto de alguna persona discapacitada. (M. Fried-Oken, 2013)

Algunas de las aplicaciones son: ARASAAC, Aumentativa 2.0, DictaPicto, Pictoeduca, Pictotraductor, Pictosonidos, Pictocuentos, Wikipicto, Pictoagenda, Pictojuegos, etc (Comunicaciones, 2012).

Un ejemplo de estas aplicaciones es LetMeTalk que te permite formar oraciones con pictogramas como se muestra en la ilustración 2.16.

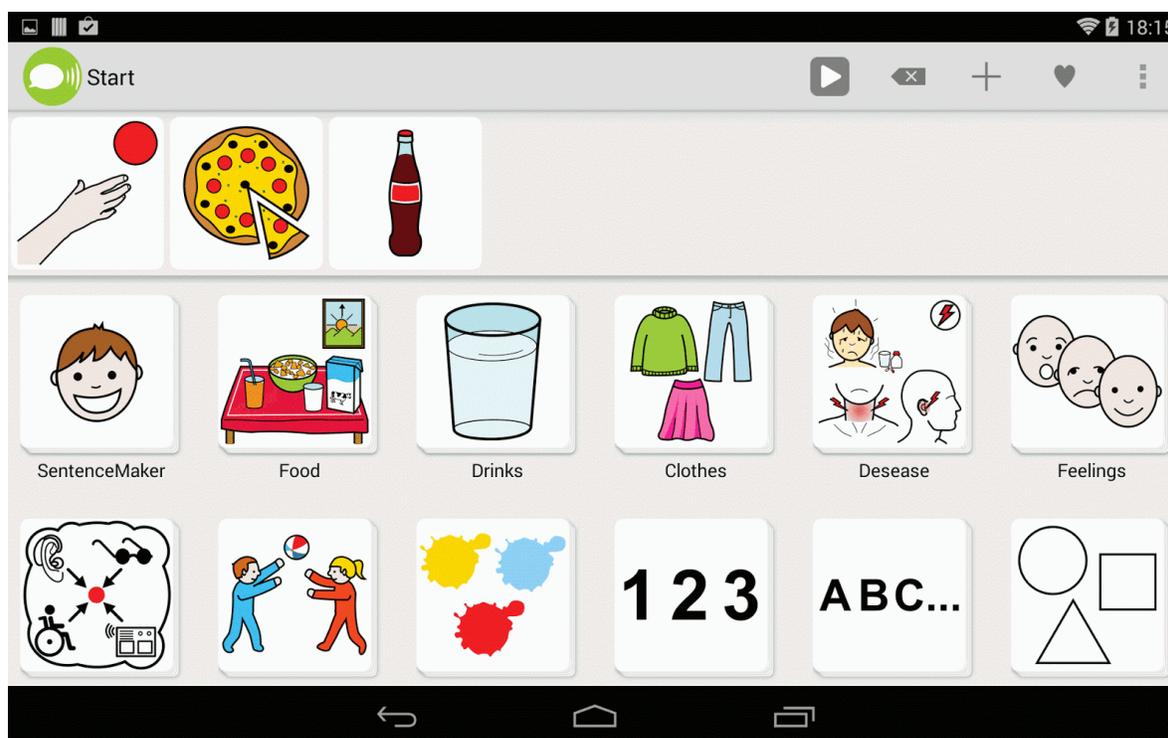


Ilustración 2.16 LetMeTalk aplicación.

2.3.8 Cámaras

Las cámaras utilizadas con software son capaces de detectar movimientos en el rostro como un parpadeo, por lo mismo se ha implementado en algunos SAAC.

Un sistema de comunicación asistido, táctil o por seguimiento ocular, ofrece un inigualable método alternativo de comunicación, capacitación, mejorando el bienestar y la calidad de vida a las personas de todas las edades con alguna de estas discapacidades del habla.

Su principal funcionamiento se basa en lectores que siguen la pupila de los pacientes e interpretan los deseos del mismo a través de una interfaz gráfica muy sencilla y amigable que

permite escribir texto, dibujar imágenes, seleccionar figuras, jugar, escribir en redes sociales, entre otras aplicaciones. La computadora lee al final el escrito que se escribió con los ojos del paciente. De esta manera, este se puede comunicar e interactuar con sus familiares, amigos y el mundo mejorando su calidad de vida y de quienes lo rodean (Asistronic, 2018).

Un ejemplo del funcionamiento de este sistema se muestra en la ilustración 2.17.



Ilustración 2.17 SAAC con Cámara.

Existen otros métodos de interacción aún más específicos dadas las condiciones y contexto de la persona a quien se le quiera adaptar, con los métodos de interacción abarcados se engloba la mayoría de estos.

2.4 Microcomputadora Raspberry Pi

La Raspberry Pi es una microcomputadora de bajo costo y tamaño reducido, tanto es así que cabe en la palma de la mano, pero puedes conectarle una pantalla y un teclado para interactuar con ella exactamente igual que cualquier otra computadora.

Para encontrar su origen tenemos que irnos hasta el Reino Unido, allí es donde nació como una organización caritativa, la Fundación Raspberry Pi en 2009. Su objetivo es ayudar a los niños a aprender informática en las escuelas. Hoy en día es relativamente frecuente encontrar ordenadores y tabletas en las aulas, pero no es un recurso asequible al que puedan acceder todos los centros ni todos los países.

La Raspberry Pi es la placa de un ordenador simple compuesto por un SoC, CPU, memoria RAM, puertos de entrada y salida de audio y vídeo, conectividad de red, ranura SD para almacenamiento, reloj, una toma para la alimentación, conexiones para periféricos de bajo nivel, reloj, entre otros, es prácticamente lo mismo que la parte de atrás de la torre de un ordenador, porque la Raspberry es un ordenador. Eso sí, no tiene interruptor para encenderlo o apagarlo.

Para ponerlo en marcha tenemos que conectar periféricos de entrada y salida para poder interactuar como una pantalla, un ratón y un teclado y grabar un sistema operativo para Raspberry Pi en la tarjeta SD. Ya solo queda conectarlo a la corriente y funcionara (Luis, 2018).

2.4.1 Modelos

La Raspberry Pi ha evolucionado con el paso del tiempo y por lo mismo han surgido nuevos modelos desde el 2012. En la web oficial de Raspberry Pi se encuentran nueve modelos de placa doméstica, cinco variaciones para uso industrial y una buena colección de accesorios, aunque si se busca afuera de la página oficial, las posibilidades crecen exponencialmente en oferta. Restringiéndonos al uso doméstico y englobando opciones con leves diferencias, en ilustración 2.18 y la tabla 2.2 se muestran los diferentes modelos hasta el 2020 (Eva, 2020).

Tabla 2.2 Tabla de modelos Raspberry Pi.

| MODELO | CPU | RAM | CONECTIVIDAD INALÁMBRICA | PUERTOS E/S | PRECIO |
|---------------------------------|--|---------|--------------------------------------|--|---|
| RASPBERRY PI 4B | 1.5-GHz, 4-core Broadcom BCM2711 (Cortex-A72) | 2/4/8GB | 802.11ac / Bluetooth 5.0 | 2x USB 3.0, 2x USB 2.0, 1x Gigabit Ethernet, 2x micro HDMI | 39,95 euros (2GB) 61 euros (4GB) 93 euros (8GB) |
| RASPBERRY PI 3 B+ | 1.4-GHz, 4-core Broadcom BCM2837B0 (Cortex-A53) | 1GB | 802.11ac, Bluetooth 4.2, Ethernet | 4 x USB 2.0, HDMI, 3.5mm audio | 39 euros |
| RASPBERRY PI ZERO W | 1-GHz, 1-core Broadcom BCM2835 (ARM1176JZF-S) | 512MB | 802.11n / Bluetooth 4.1 | 1x micro USB, 1x mini HDMI | 10,53 euros |
| RASPBERRY PI ZERO WH | 1-GHz, 1-core Broadcom BCM2835 (ARM1176JZF-S) | 512MB | 802.11n / Bluetooth 4.1 | 1x micro USB, 1x mini HDMI | 14,60 euros |
| RASPBERRY PI ZERO | 1-GHz, 1-core Broadcom BCM2835 (ARM1176JZF-S) | 512MB | - | 1x micro USB, 1x mini HDMI | 5,40 euros |

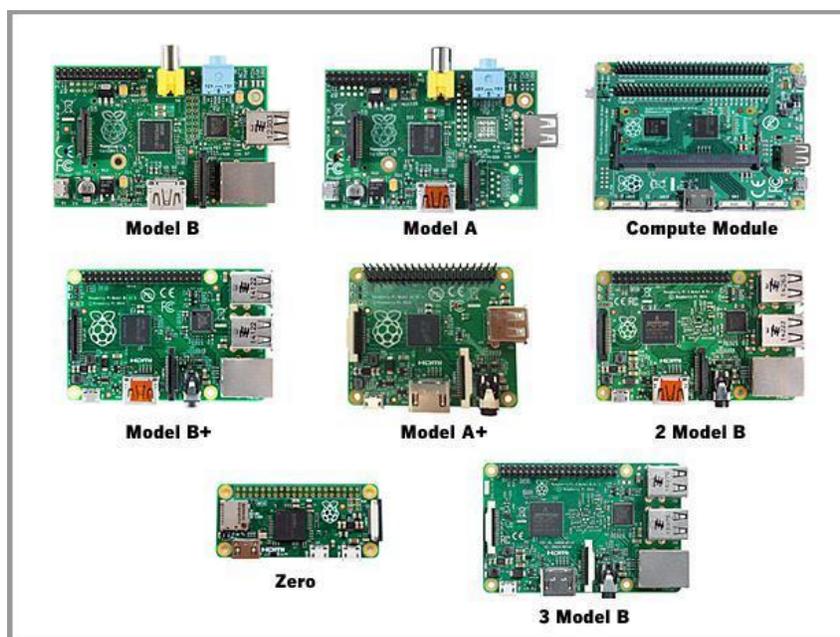


Ilustración 2.18 Modelos Raspberry Pi.

2.4.2 Pines

Los pines GPIO de la placa Raspberry Pi 4, la 3, y también de sus predecesores, dotan a la placa SBC de capacidades similares a las que puede tener Arduino, ya que con ellos puedes crear proyectos electrónicos muy interesantes controlados desde el sistema operativo mediante código en distintos lenguajes, como Python.

GPIO son las siglas de General Purpose Input/Output, es decir, Entrada/Salida de propósito general. Lo pueden tener diferentes productos electrónicos, como los propios chips o ciertas placas PCB como es el caso de esta Raspberry Pi. Como su propio nombre indica, son pines que se pueden configurar para realizar distintas funciones, de ahí que sean de propósito general y no para un uso específico.

Las nuevas placas Raspberry Pi 4 y la versión 3 están dotadas con un gran número de pines GPIO. No todas las versiones ofrecen la misma cantidad, ni están numerados de la misma forma, así que hay que tener cuidado con esto para saber bien cómo debes realizar la conexión según el modelo y revisión que tengas.

Pero lo que sí es más genérico es los tipos de GPIO que puedes encontrar en el puerto de las placas Raspberry Pi. Y estos son los tipos de pines con los que cuenta en la ilustración 2.19 y a continuación sus características:

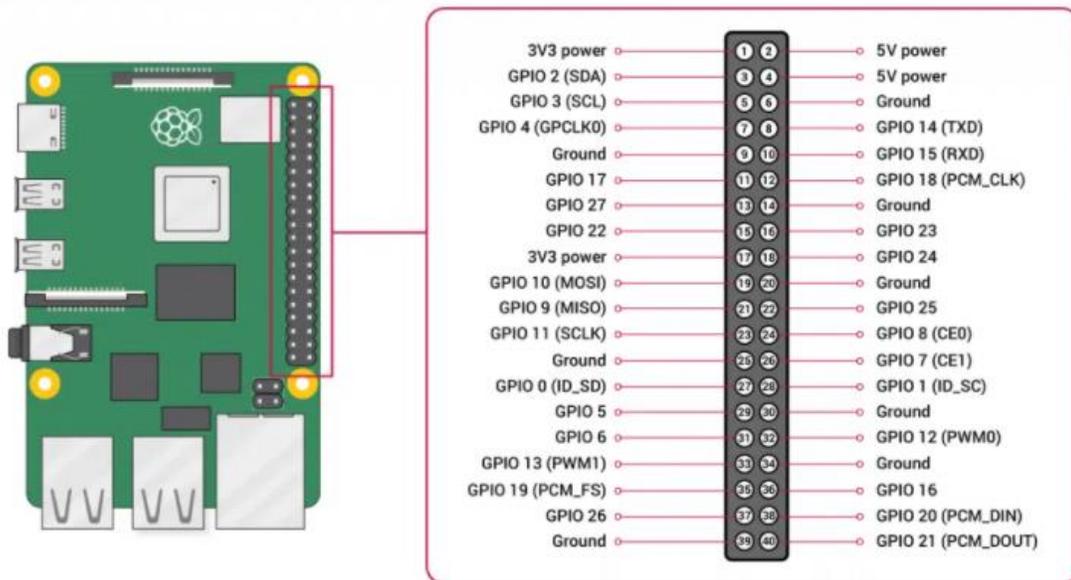


Ilustración 2.19 Pines de la Raspberry Pi 4.

- ◆ Alimentación: estos pines sirven para poder conectar las líneas o cableado de alimentación para circuitos electrónicos. Se corresponden a pines similares a los que tiene la placa Arduino, y que proveen de voltajes de 5v y 3v3 (3.3v limitados a 50mA de carga). Además, tiene tierra (GND o Ground). Si no se usan fuentes externas de alimentación como pilas, baterías, o adaptadores, estos pines sirven de gran ayuda para alimentar tu circuito.

- ◆ DNC (Do Not Connect): son pines que hay en algunas versiones y que no tienen función, pero que en las nuevas placas sí que se les ha dado otro fin. Estos solos los encontrarás en modelos más primitivos de la placa. En las nuevas 3 y 4 estarán marcados como GND por lo general, pudiéndose integrar en el grupo anterior.
- ◆ Pines configurables: son los GPIO normales, y se pueden programar mediante códigos.
- ◆ Pines especiales: son algunas conexiones que están destinados a conexiones o interfaces especiales como UART, conexiones serie TXD y RXD, etc., como ocurre con Arduino. Incluso se encuentran algunos como SDA, SCL, MOSI, MISO, SCLK, CE0, CE1, etc. Entre ellos destacan:
 - PWM, que pueden regular el ancho del pulso. En la Raspberry Pi 3 y 4 son los GPIO12, GPIO13, GPIO18 y GPIO19.
 - SPI es otra interfaz de comunicación. En el caso de las nuevas placas de 40-pin, son los pines (con diferentes canales de comunicación como ves):
 - SPI0: MOSI (GPIO10), MISO (GPIO9), SCLK (GPIO11), CE0 (GPIO8), CE1 (GPIO7)
 - SPI1: MOSI (GPIO20); MISO (GPIO19); SCLK (GPIO21); CE0 (GPIO18); CE1 (GPIO17); CE2 (GPIO16)
 - I2C es otra conexión de comunicación serial entre dos dispositivos digitales. Este bus está compuesto por la señal de datos (GPIO2) y el reloj (GPIO3). Además de EEPROM Data (GPIO0) y EEPROM Clock (GPIO1).
 - Serial, otra comunicación muy práctica con pines TX (GPIO14) y RX (GPIO15) como los que puedes encontrar en la placa Arduino UNO.

Se debe recordar que los GPIO son la interfaz entre la Raspberry Pi y el mundo exterior, pero que tienen sus limitaciones, especialmente eléctricas. Algo que se debe tener muy en cuenta para no estropear la placa es recordar que estos pines GPIO suelen ser unbuffered, es decir, sin buffer. Esto quiere decir que no disponen de protección, por lo que debes vigilar las magnitudes de voltaje e intensidad aplicada para no terminar con una placa inservible (Isaac, 2018).

2.5 Arduino

Arduino es una plataforma de creación de electrónica de código abierto, la cual está basada en hardware y software libre, flexible y fácil de utilizar para los creadores y desarrolladores. Esta plataforma permite crear diferentes tipos de microordenadores de una sola placa a los que la comunidad de creadores puede darles diferentes tipos de uso.

El proyecto nació en 2003, cuando varios estudiantes del Instituto de Diseño Interactivo de Ivrea, Italia, con el fin de facilitar el acceso y uso de la electrónica y programación. Lo hicieron para que los estudiantes de electrónica tuviesen una alternativa más económica a las

populares BASIC Stamp, unas placas que por aquel entonces valían más de cien dólares, y que no todos se podían permitir.

El resultado fue Arduino, una placa con todos los elementos necesarios para conectar periféricos a las entradas y salidas de un microcontrolador, y que puede ser programada tanto en Windows como macOS y GNU/Linux. Un proyecto que promueve la filosofía 'learning by doing', que viene a querer decir que la mejor manera de aprender es practicando.

El Arduino es una placa basada en un microcontrolador ATMEL. Los microcontroladores son circuitos integrados en los que se pueden grabar instrucciones, las cuales las escribes con el lenguaje de programación que puedes utilizar en el entorno Arduino IDE. Estas instrucciones permiten crear programas que interactúan con los circuitos de la placa.

El microcontrolador de Arduino posee lo que se llama una interfaz de entrada, que es una conexión en la que podemos conectar en la placa diferentes tipos de periféricos. La información de estos periféricos que conectes se trasladará al microcontrolador, el cual se encargará de procesar los datos que le lleguen a través de ellos.

El tipo de periféricos que puedas utilizar para enviar datos al microcontrolador depende en gran medida de qué uso le estés pensando dar. Pueden ser cámaras para obtener imágenes, teclados para introducir datos, o diferentes tipos de sensores.

También cuenta con una interfaz de salida, que es la que se encarga de llevar la información que se ha procesado en el Arduino a otros periféricos. Estos periféricos pueden ser pantallas o altavoces en los que reproducir los datos procesados, pero también pueden ser otras placas o controladores (Fernández, 2020).

2.5.1 Arduino UNO

Es un modelo de la plataforma Arduino, siendo la más utilizada en el mercado y con mayor comunidad, la cual le da un soporte en documentación enorme.

El modelo Uno se considera la base a partir del cual incorporar diferentes configuraciones para crear otras placas.

Tiene el tamaño de una tarjeta de crédito, dispone de 14 pines digitales de entrada/salida (de los cuales 6 pueden ser usando con PWM), 5 analógicos, un cristal de 16Mhz, conector jack de alimentación y requiere 5 Volts para funcionar. Puedes alimentarla con un cable USB o con adaptador de corriente.

Tiene incorporado el procesador ATmega328P con 32Kb de memoria donde se guarda el código del proyecto, una capacidad bastante discreta para el estándar actual, razón por la cual si practicas con el Arduino tarde o temprano se te quedará corta. Además, a largo plazo también podrías verte limitado tanto por la cantidad de pines como por la memoria para ejecutar procesos (SRAM) y la de almacenar variables (EEPROM).

Sin embargo, para aprender es perfecta ya que integra suficientes funciones para comenzar, es barata, hay una gran cantidad de material en internet para aprender con este modelo (EVA, 2021).

La placa es del tamaño de una tarjeta aproximadamente y se muestra en la ilustración 2.20.



Ilustración 2.20 Arduino UNO.

2.5.2 IDE de Arduino

Las siglas IDE vienen de Integrated Development Environment, lo que traducido a nuestro idioma significa Entorno de Desarrollo Integrado. Esto es simplemente una forma de llamar al conjunto de herramientas software que permite a los programadores poder desarrollar (es decir, básicamente escribir y probar) sus propios programas con comodidad. En el caso de Arduino, necesitamos un IDE que nos permita escribir y editar nuestro programa (también llamado “sketch” en el mundo de Arduino), que nos permita comprobar que no hayamos cometido ningún error y que además nos permita, cuando ya estemos seguros de que el sketch es correcto, grabarlo en la memoria del microcontrolador de la placa Arduino para que este se convierta a partir de entonces en el ejecutor autónomo de dicho programa (Arteo, 2013).

Un ejemplo de la interfaz de este software se muestra en la ilustración 2.21.

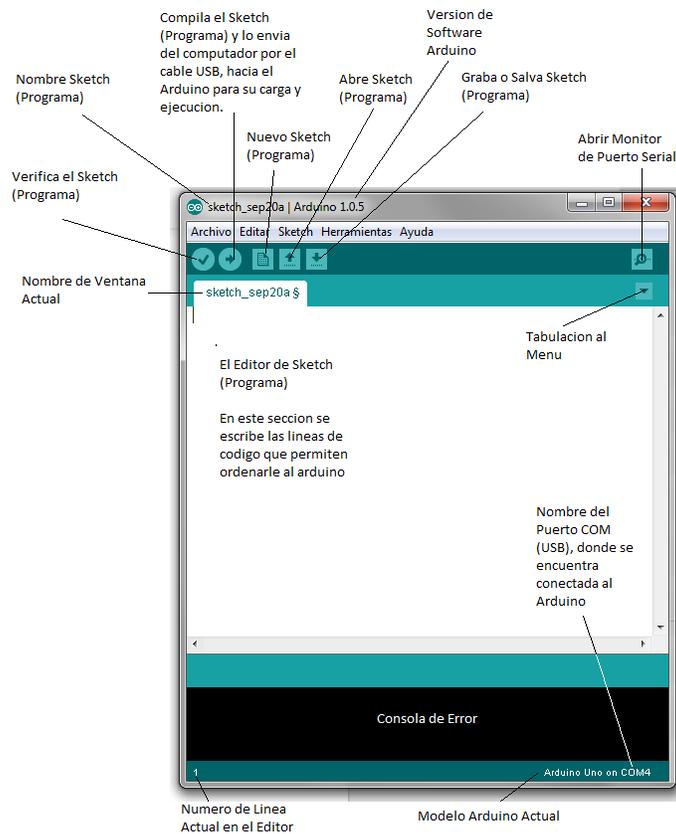


Ilustración 2.21 Arduino IDE.

“Compilar” significa convertir un código escrito en el IDE (utilizando el lenguaje Arduino, y, por tanto, entendible fácilmente por los seres humanos) en el programa realmente ejecutable por el microcontrolador, que no es más que un inmenso conjunto de bits (es decir, 1s y 0s) tan solo entendible por él.

Ya que las placas Arduino incorporan microcontroladores de arquitectura AVR (excepto la Due, que es ARM, pero no lo tendremos en cuenta), es lógico pensar que el compilador incluido en el entorno de programación oficial de Arduino es uno específico para generar código binario compatible con este tipo de chips. Y así es. Sin embargo, dicha herramienta (llamada “gcc-avr”) no compila código escrito en lenguaje Arduino a código binario AVR (concretamente, en formato “.hex”), sino que compila código escrito en lenguaje C/C++ a código binario AVR (Arteo, 2013).

2.5.3 Bibliotecas de Arduino

Las bibliotecas sirven para proporcionar funcionalidad extra en nuestros sketches (ofreciendo la posibilidad de manipular datos en diferentes formatos, de gestionar protocolos de comunicación diversos, de interactuar con hardware variado, etc.).

Para poder utilizar las funciones que proveen, la biblioteca en cuestión primero se ha de importar, o bien mediante la opción “Sketch” “Import library” del menú del IDE, o bien incluyendo al principio del código del sketch la sentencia `#include` pertinente (Arteo, 2013).

2.5.3.1 I2Cdev

I2C (Inter-Integrated Circuit, también conocido con el nombre de TWI de “TWo-wIre”, literalmente “dos cables” en inglés): es un sistema muy utilizado en la industria principalmente para comunicar circuitos integrados entre sí. Su principal característica es que utiliza dos líneas para transmitir la información: una (llamada línea “SDA”) sirve para transferir los datos (los 0s y los 1s) y otra (llamada línea “SCL”) sirve para enviar la señal de reloj. En realidad, también se necesitarían dos líneas más: la de alimentación y la de tierra común, pero estas ya se presuponen existentes en el circuito (Arteo, 2013).

2.5.3.2 MPU6050

Es una biblioteca creada por usuarios que tiene como objetivo facilitar la conexión del sensor MPU6050 que es un giroscopio de 3 ejes y acelerómetro de 3 ejes (naylor mechatronics, 2016).

2.5.3.3 Wire

Permite comunicar mediante el protocolo I2C (también llamado TWI) la placa Arduino con dispositivos externos. Tal como ya se ha comentado varias veces, la comunicación se establece a través de los pines SDA (línea de datos, ubicada en pin analógico 4 en la placa Arduino UNO) y SCL (línea de reloj, ubicada en el pin analógico 5). Esta biblioteca utiliza 7 bits para identificar el dispositivo (lo que da la posibilidad de distinguir hasta 128 dispositivos diferentes) (Arteo, 2013).

2.5.3.4 LiquidCrystal

Permite controlar pantallas de cristal líquido (LCD) basadas en el chip HD44780 de Hitachi (o compatibles, como el KS0066 de Samsung o el ST7065C de Sitronix, entre otros). Este modelo de chip se encuentra en la mayoría de LCD de caracteres. La biblioteca puede trabajar tanto en el modo 4-bit como en 8-bit (es decir: puede utilizar tanto 4 como 8 líneas de datos) (Arteo, 2013).

2.6 Microcontroladores

El Microcontrolador es un circuito integrado que es el componente principal de una aplicación embebida. Es como una pequeña computadora que incluye sistemas para controlar elementos de entrada/salida. También incluye a un procesador y por supuesto memoria que puede guardar el programa y sus variables (flash y RAM). Funciona como una mini PC. Su función es la de automatizar procesos y procesar información (E-Marmolejo, 2017).

Dentro de los microcontroladores existen dos arquitecturas básicas de hardware.

- ◆ Von Neumann: Se caracteriza por tener una memoria única para los datos y las instrucciones del programa. A dicha memoria se accede a través de un sistema de buses único (control, direcciones y datos) como se muestra en la ilustración 2.22 (ptolomeo, Desconocido).

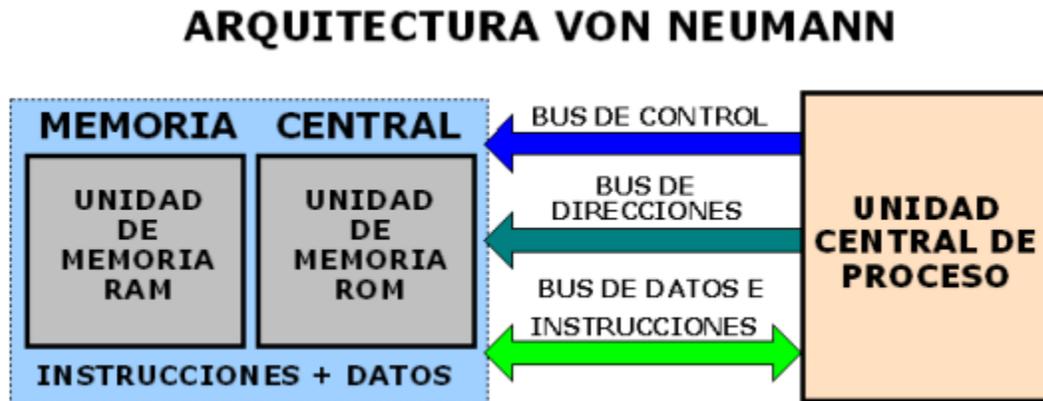


Ilustración 2.22 Esquema Von Neumann.

- ◆ Harvard: Este modelo tiene la unidad central de proceso (CPU) conectada a dos memorias, una con las instrucciones y otra con los datos, por medio de dos buses diferentes. Una de las memorias contiene solamente las instrucciones del programa (Memoria de Programa), y la otra sólo almacena datos (Memoria de Datos) como se muestra en la ilustración 2.23 (ptolomeo, Desconocido).



Ilustración 2.23 Esquema Harvard.

2.6.1 Microprocesadores

Un microprocesador es el conjunto de tres sistemas digitales: ALU, unidad de control y registros. Los microprocesadores fueron los padres de los microcontroladores.

Un procesador incluye al menos tres elementos, ALU, unidad de control y registros.

- ◆ ALU. También conocida como Unidad Aritmética y Lógica. Esta unidad está compuesta por los circuitos electrónicos digitales del tipo combinatorios (compuertas, sumadores, multiplicadores), cuya principal función es el realizar operaciones. Estas operaciones están divididas en tres tipos:
 - Lógicas. Como las operaciones básicas de las compuertas lógicas, como la suma lógica (OR), multiplicación lógica (AND), diferencia lógica (XOR) y negación (NOT). Una operación lógica sólo puede tener como entradas y como salidas una respuesta lógica (0 o 1). Esto dependiendo de los niveles de voltajes de una señal digital.
 - Aritméticas. Las operaciones aritméticas son la suma, resta, multiplicación y división. Dependiendo del procesador (8, 16, 32 o 64 bits) será la rapidez con la que se pueden hacer dichas operaciones.
 - Misceláneas. En estas operaciones caen todas las demás operaciones como la transferencia de bits.
- ◆ Unidad de control. La unidad de control es el conjunto de sistemas digitales secuenciales (aquellos que tienen memoria) que permiten distribuir la lógica de las señales.
- ◆ Registros. Los registros son las memorias principales de los procesadores, ya que funcionan a la misma velocidad que el procesador a diferencia de otras memorias un tanto más lentas (como la RAM, FLASH o la CACHE). Los registros están contruidos por Flip-Flops. Los Flip-Flops son circuitos digitales secuenciales (E-Marmolejo, 2017).

2.7 ATmega328P

Como bien sabrán aquellos que siguen a Arduino, uno de los microcontroladores que más van a encontrar en las plataformas de Arduino será el ATmega328¹. Este es un microcontrolador de la firma Atmel (ahora adquirida por Microchip) y está basado en la arquitectura AVR.

AVR es una arquitectura basada en RISC (Computadora con conjunto de instrucciones reducidas), recordemos que la arquitectura de un microprocesador o microcontrolador, detalla la estructura interna del mismo y como se conforma la distribución, acceso y control de los diversos bloques que componen al microcontrolador (Electgpl, 2016).

En la tabla 2.3 se muestran las características de sus pines en la ilustración 2.24, se muestran los pines del ATmega328P.

Tabla 2.3 Características ATmega328P.

Technical specs

| | |
|------------------------|--|
| Microcontroller | ATmega328P |
| Operating Voltage | 5V |
| Digital I/O Pins | 14 (of which 6 provide PWM output) |
| PWM Digital I/O Pins | 6 |
| Analog Input Pins | 6 |
| DC Current per I/O Pin | 20 mA |
| Flash Memory | 32 KB (ATmega328P) of which 0.5 KB used by bootloader |
| SRAM | 2 KB (ATmega328P) |
| EEPROM | 1 KB (ATmega328P) |
| Clock Speed | 16 MHz |

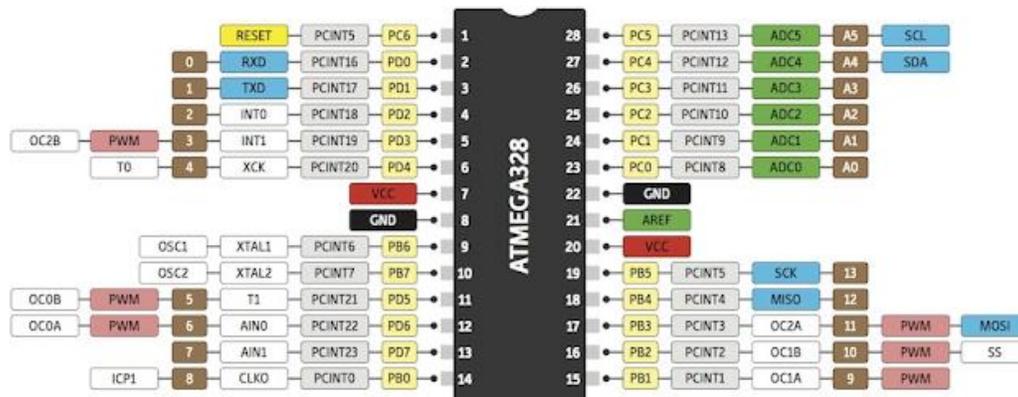


Ilustración 2.24 Atmga328p.

¹ Atmel, ATmega328P Datasheet, recuperado el 10 de diciembre del 2021 de: ATmega328P (microchip.com)2015, Microchip.

El microcontrolador de alto rendimiento Microchip picoPower® de 8 bits basado en AVR® RISC combina una memoria Flash ISP de 32 KB con capacidades de lectura y escritura, EEPROM 1024B, SRAM de 2 KB, 23 líneas de E / S de propósito general, 32 registros de trabajo de propósito general , tres temporizadores / contadores flexibles con modos de comparación, interrupciones internas y externas, USART programable en serie, una interfaz en serie de dos cables orientada a bytes, puerto en serie SPI, un convertidor A / D de 6 canales y 10 bits (8 canales en TQFP y paquetes QFN / MFL), temporizador de vigilancia programable con oscilador interno y cinco modos de ahorro de energía seleccionables por software. El dispositivo funciona entre 1,8 y 5,5 volts.

Al ejecutar poderosas instrucciones en un solo ciclo de reloj, el dispositivo logra rendimientos cercanos a un MIPS por MHz, equilibrando el consumo de energía y la velocidad de procesamiento (Microchip, 2021).

2.8 Micrófonos

El micrófono un transductor capaz de convertir señal sonora en una señal eléctrica. Con mayor precisión, convierte presión sonora en tensión (Miyara, 2006).

2.8.1 Tipos de micrófono

Existen tres tipos de micrófono los cuales son:

- ♦ **Micrófonos dinámicos:** Existen varios mecanismos de conversión de energía sonora en energía eléctrica utilizados en los micrófonos. Los más habituales corresponden a los micrófonos dinámicos y los micrófonos capacitivos. Los micrófonos dinámicos, también denominados de bobina móvil, están constituidos por una bobina con varias espiras de alambre de cobre que se desplaza en forma oscilante a lo largo de un núcleo cilíndrico de imán. La bobina es impulsada por un diafragma que vibra en concordancia con las variaciones de presión de una onda sonora como se muestra en la ilustración 2.25. De la física se sabe que cuando una bobina se mueve dentro de un campo magnético (en este caso el del imán) se genera en sus terminales una tensión eléctrica denominada fuerza electromotriz, y éste es precisamente el principio de operación de los micrófonos dinámicos.

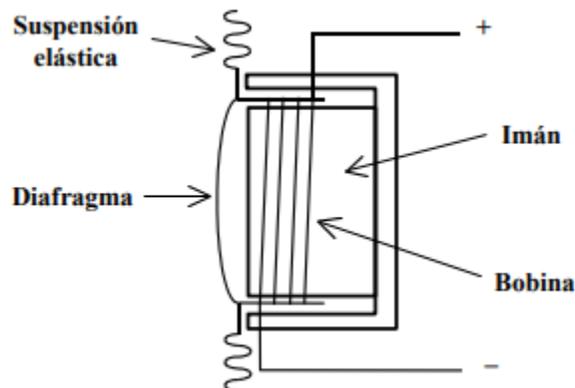


Ilustración 2.25 Micrófono Dinámico.

Los micrófonos dinámicos generan tensiones bastante pequeñas, del orden de 1 a 4 mV/Pa (milivolt por pascal). Para lograr mayores sensibilidades sería necesario que la bobina tuviera muchas espiras, lo cual implicaría aumentar su masa. Esto repercutiría negativamente en la respuesta en alta frecuencia, dado que, a mayor masa, mayor inercia, es decir mayor dificultad para que una onda de alta frecuencia ponga en movimiento al conjunto diafragma o bobina (sería equivalente a intentar sacudir rápidamente un objeto muy pesado) (Miyara, 2006).

- ◆ Los micrófonos capacitivos (también denominados de capacitor, de condensador, o electrostáticos) se basan en la utilización de un campo eléctrico en lugar de un campo magnético. Están formados por un diafragma muy delgado (típicamente, 5 micrones de espesor) bañado en oro, y una placa posterior metálica que normalmente está perforada o ranurada como se muestra en la ilustración 2.26. Ambos forman un condensador cuya capacidad “C” varía con la distancia entre sí, y esta distancia varía al vibrar el diafragma impulsado por las variaciones de presión de la onda sonora. La ecuación fundamental de un condensador es:

$V = q / C$ donde “V” es la tensión en sus terminales, “q” es la carga eléctrica en sus terminales, y “C” es el valor de la capacidad asociada al condensador. Si de alguna manera se consigue cargar las placas del condensador con una carga fija “q”, al variar la capacidad “C” variará también la tensión “V” que se mide entre los terminales. Este es el principio físico de funcionamiento de los micrófonos capacitivos

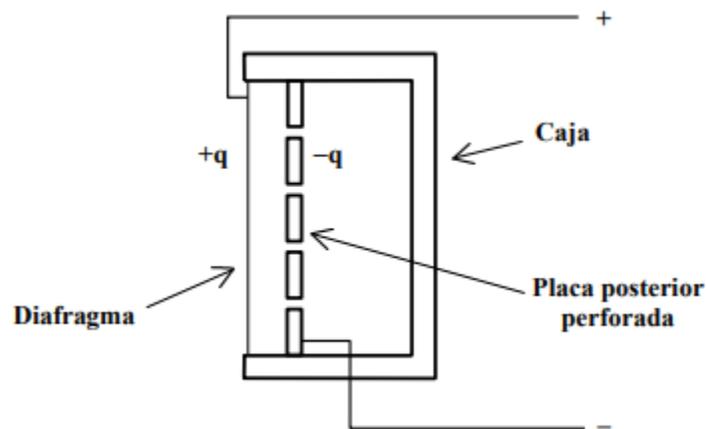


Ilustración 2.26 Micrófono Capacitivo.

La manera de cargar las placas del condensador es por medio de una polarización externa, lo cual se logra conectando el micrófono a una fuente de tensión constante a través de una resistencia.

En la actualidad se encuentra muy difundido un tipo de micrófono a condensador prepolarizado, es decir con una polarización interna intrínseca, que en principio no requiere la utilización de una fuente fantasma. Son los denominados electret, y se caracterizan porque una de las placas contiene una película aislante especial en la cual se han introducido, durante

la fabricación, cargas eléctricas que quedan atrapadas en la estructura interna sin posibilidad de escapar (Miyara, 2006).

2.9 Amplificadores operacionales

Un amplificador operacional (A.O. también op-amp), es un amplificador de alta ganancia directamente acoplado, que en general se alimenta con fuentes positivas y negativas, lo cual permite que obtenga excursiones tanto por arriba como por debajo de masa o punto de referencia que se considere. Se caracteriza especialmente porque su respuesta en: frecuencia, cambio de fase y alta ganancia que se determina por la realimentación introducida externamente. Por su concepción, presenta una alta impedancia (Z) de entrada y muy baja de salida (García, 2010).

En la ilustración 2.27 se muestra el amplificador operacional ideal.

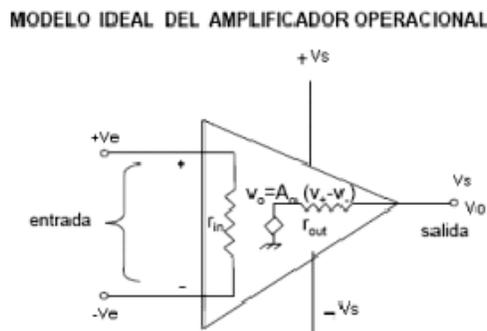


Ilustración 2.27 Modelo Ideal del Amplificador Operacional.

2.9.1 LM386

El LM386 es un amplificador de potencia diseñado para su uso en aplicaciones de consumo de baja tensión². La ganancia se establece internamente en 20 para mantener bajo el recuento de piezas externas, pero la adición de una resistencia externa y un condensador entre los pines 1 y 8 aumentará la ganancia a cualquier valor de 20 a 200. la mitad de la tensión de alimentación. El consumo de energía en reposo es de solo 24 mili watts cuando se opera con un suministro de 6 volts, lo que hace que el LM386 sea ideal para el funcionamiento con batería. En la ilustración 2.28 se muestra una imagen del integrado (Texas Instruments, 2017).

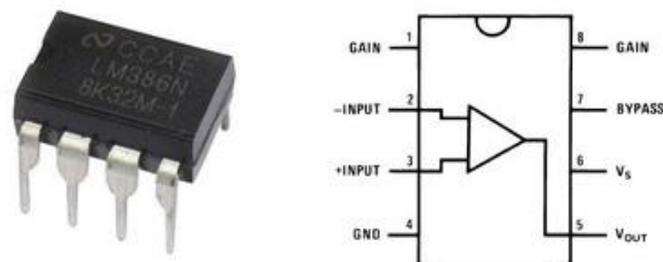


Ilustración 2.28 LM386, amplificador de audio.

² Texas Instruments, 2017, LM386 Datasheet, recuperado el 10 de diciembre del 2021 de: LM386 Low Voltage Audio Power Amplifier datasheet (Rev. C) (ti.com)

En la ilustración 2.29 se muestra la aplicación de recuento mínimo de piezas que se puede implementar con LM386. Su ganancia es internamente establecida en 20 (Texas Instruments, 2017).

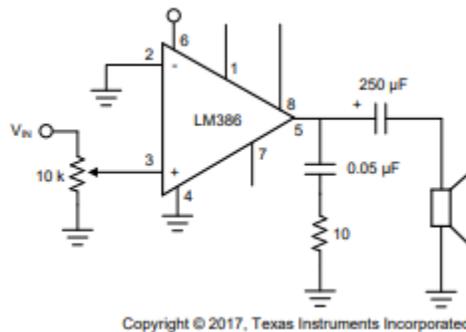


Ilustración 2.29 LM386 con ganancia de 20.

2.10 Componentes pasivos

Podemos definir los componentes electrónicos pasivos como aquellos que no producen amplificación y que sirven para controlar la electricidad colaborando al mejor funcionamiento de los elementos activos (los cuales son llamados genéricamente semiconductores). Los componentes pasivos están formados por elementos de diversas clases que tendremos que considerar independientemente, ya que son diferentes sus objetivos, construcción y resultados, de modo que vamos a dividirlos en tres grandes grupos: (Castellanos, 2005).

1. Resistores.
2. Condensadores.
3. Inductores.

2.10.1 Resistencia

La resistencia eléctrica, un fenómeno físico de la oposición que presenta un material a ser atravesado por una corriente eléctrica. Una resistencia o resistor, un componente usado en electricidad y electrónica asociado a las pérdidas de voltaje entre dos puntos de un circuito.

El ohm es la unidad de medida de la resistencia que oponen los materiales al paso de la corriente eléctrica y se representa con el símbolo o letra griega " Ω " (omega) un ejemplo de estas se muestra en la ilustración 2.30 (Castellanos, 2005) .



2.10.2 Capacitores

El condensador o capacitor es un dispositivo que almacena energía eléctrica, es un componente pasivo. Está formado por un par de superficies conductoras en situación de influencia total que, sometidas a una diferencia de potencial, adquieren una determinada carga eléctrica, positiva en una de las placas y negativa en la otra (siendo nula la carga total almacenada). La carga almacenada en una de las placas es proporcional a la diferencia de potencial entre esta placa y la otra, siendo la constante de proporcionalidad la llamada capacidad o capacitancia. En el Sistema internacional de unidades se mide en Faradios (F) (Castellanos, 2005).

En la ilustración 2.31 se muestran diferentes tipos de capacitores.

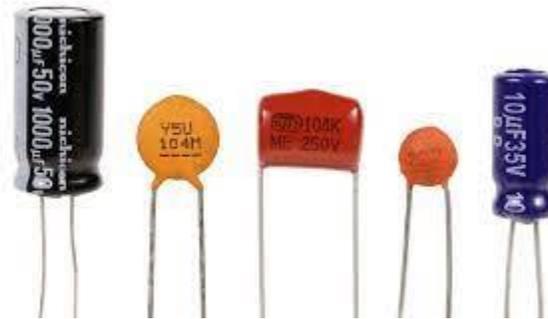


Ilustración 2.31 Diferentes tipos de Capacitores.

2.10.3 Inductores

Un inductor bobina o inductancia es un componente pasivo de un circuito eléctrico que, debido al fenómeno de la autoinducción, almacena energía en forma de campo magnético.

Un inductor (bobina) está constituido usualmente por una cabeza hueca de una bobina de conductor, típicamente alambre o hilo de cobre esmaltado. Existen inductores con núcleo de aire o con núcleo de un material ferroso, para incrementar su capacidad de magnetismo. Los inductores pueden también estar construidos en circuitos integrados, usando el mismo proceso utilizado para realizar microprocesadores.

El inductor consta fundamentalmente de:

Devanado inductor: Es el conjunto de espiras destinado a producir el flujo magnético, al ser recorrido por la corriente eléctrica.

Núcleo: Es la parte del circuito magnético rodeada por el devanado inductor.

Un inductor bobina o inductancia es un componente pasivo de un circuito eléctrico que, debido al fenómeno de la autoinducción, almacena energía en forma de campo magnético. De acuerdo con el Sistema Internacional de Unidades, si el flujo se expresa en weber y la intensidad en amperio, el valor de la inductancia vendrá en henrio (H) (Castellanos, 2005).

En la ilustración 2.32 se muestran algunos tipos de inductores.



Ilustración 2.32 Tipos de Inductores.

2.11 Software para diseño de PCB

El diseño de circuitos impresos se lleva a cabo mediante software de diseño, emulando en computadora el circuito y verificando errores. El proceso para diseñar circuitos impresos combina la colocación y el enrutamiento de componentes para definir la conectividad eléctrica en un circuito impreso manufacturado (AUTODESK, 2020).

Existen diferentes programas que te permiten hacer diseños en PCB incluso simularlos, aquí se mostraran dos muy conocidos.

2.11.1 Proteus

Proteus Design Suite combina la facilidad de uso con un poderoso conjunto de funciones para permitir el diseño, prueba y disposición rápidos de placas de circuito impreso profesionales (Proteus, 2021).

El diseño de PCB en Proteus es una opción profesional que ofrece un diseño basado en restricciones, un potente enrutador automático y soporte dedicado para señales de alta velocidad.

Posee diferentes características como un abundante número de bibliotecas de componentes, con la función de añadir bibliotecas externas, plantillas para los componentes, puedes hacer el routing manual y automáticamente, verificar si hay un error, simular, tener diferentes pestañas para cada proyecto, visualizar en 3D y exportar en diferentes formatos (Proteus, 2021).

En la ilustración 2.33 se muestra una ventana del programa.

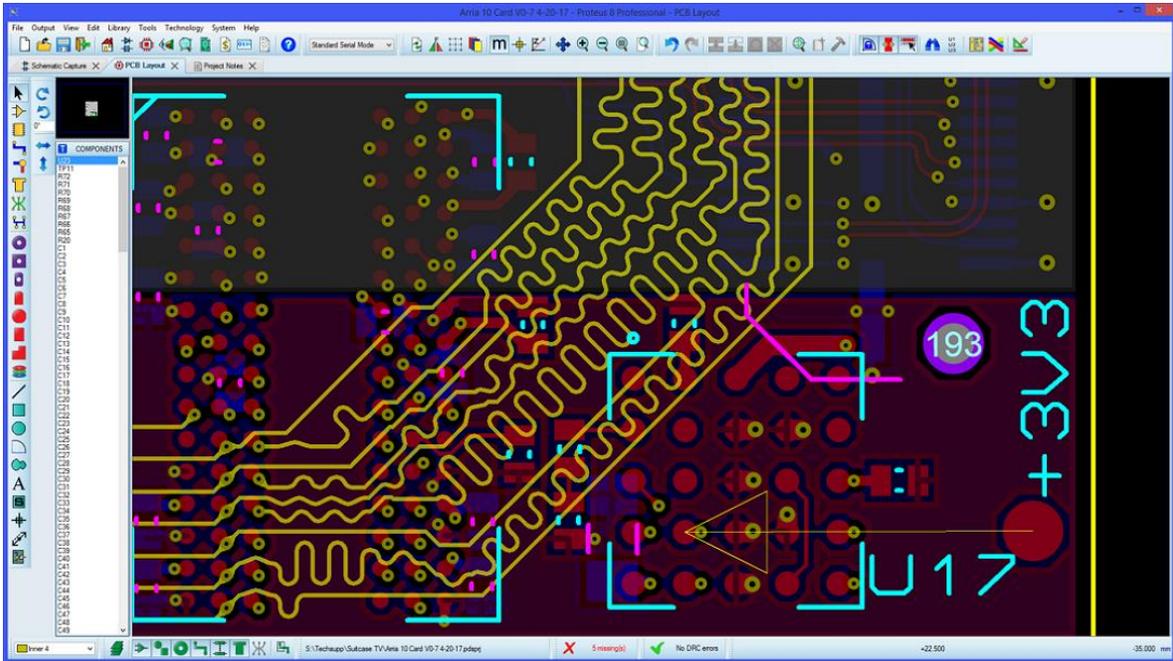


Ilustración 2.33 Proteus.

2.11.2 EAGLE

EAGLE es un software de automatización de diseño electrónico (EDA) que permite a los diseñadores de placas de circuito impreso (PCB) conectar sin problemas diagramas esquemáticos, ubicación de componentes, enrutamiento de PCB y contenido completo de la biblioteca, emulando el circuito para en una etapa posterior poder construirlo (EAGLE, 2021).

En la ilustración 2.34 se muestra una ventana en EAGLE.

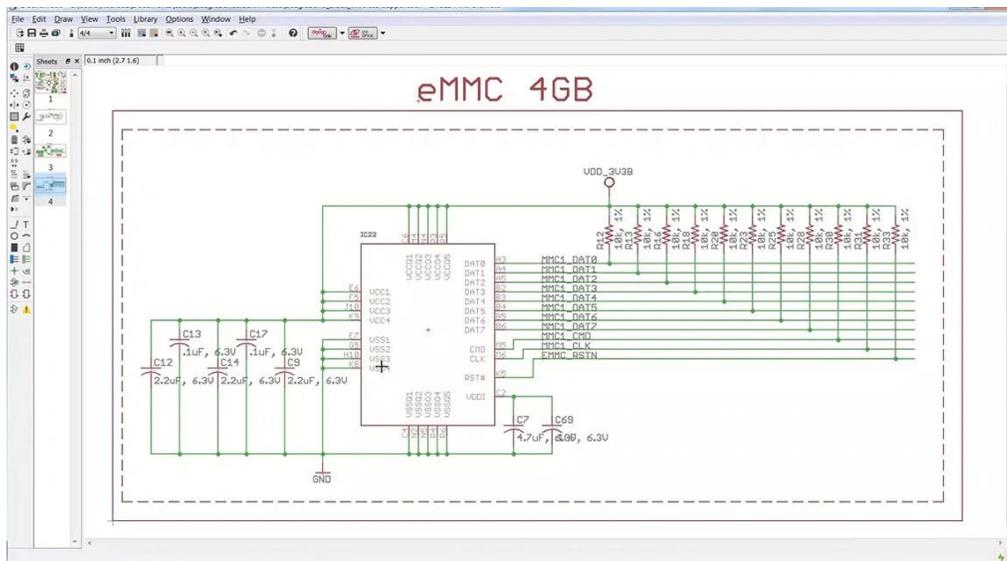


Ilustración 2.34 EAGLE.

2.12 Software para diseño de modelos 3D

El modelado 3D consiste en utilizar software para crear una representación matemática de un objeto o forma tridimensional. El objeto creado se denomina modelo 3D y se utiliza en distintas industrias.

Las industrias del cine, la televisión, los videojuegos, la arquitectura, la construcción, el desarrollo de productos, la ciencia y la medicina utilizan modelos 3D para visualizar, simular y renderizar diseños gráficos (Autodesk, 2020).

En la ilustración 2.35 se muestra un modelado en SolidWorks.

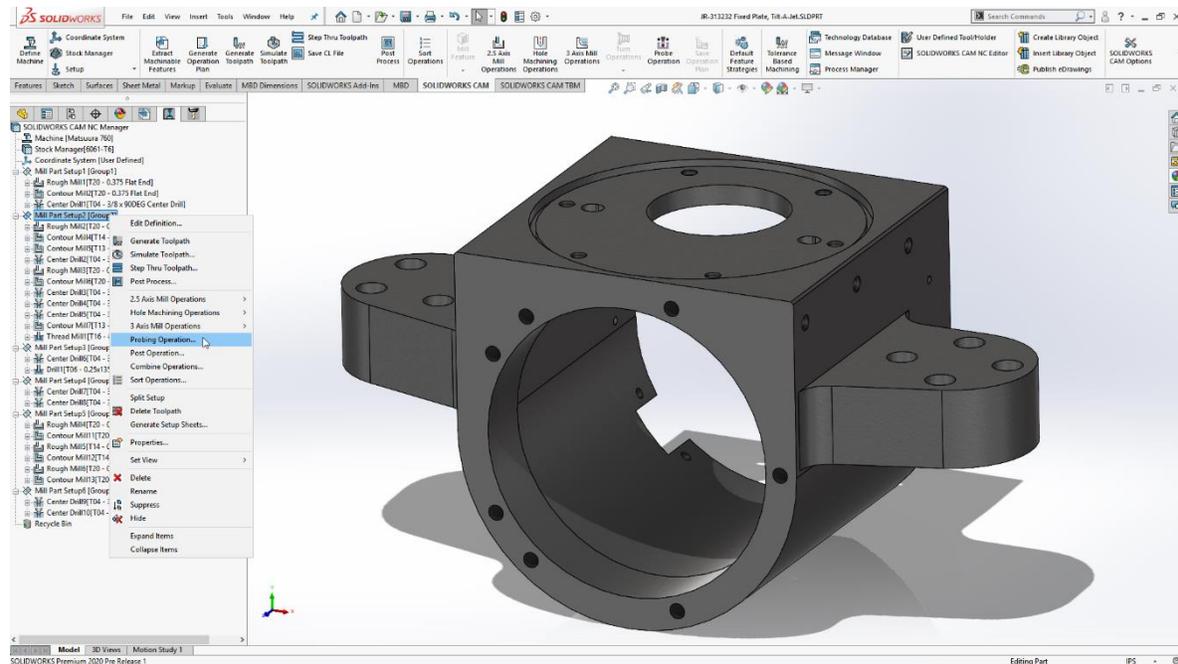


Ilustración 2.35 Modelado en SolidWorks.

2.12.1 AutoCad

AutoCAD es un software de diseño asistido por ordenador (CAD) en el que arquitectos, ingenieros y profesionales de la construcción confían para crear dibujos precisos en 2D y 3D.

- ◆ Permite dibujar, anotar y diseñar geometrías 2D y modelos 3D con caras sólidas, superficies y objetos de malla.
- ◆ Automatiza tareas como la comparación de dibujos, los recuentos, la adición de bloques, la creación de planificaciones y mucho más.
- ◆ Personaliza tareas con ayuda de las API y aplicaciones complementarias (AUTODESK, 2021).

2.12.2 SolidWorks

Durante más de 25 años, SolidWorks ha sido el estándar fiable del sector en diseño e ingeniería. Las soluciones intuitivas de diseño en 3D y desarrollo de productos de SolidWorks ayudan a conceptualizar, crear, validar, comunicar, gestionar y transformar ideas

innovadoras en grandes diseños de productos. Un ejemplo de esto, se muestra en la ilustración 2.35, que es una ventana de SolidWorks.

- ◆ Se puede crear diseños de forma rápida y precisa, incluidos modelos 3D y dibujos 2D de ensamblajes y piezas complejos.
- ◆ Optimiza los costes de diseño y fabricación utilizando herramientas de estimación de costes y realizando comprobaciones de la viabilidad de fabricación.
- ◆ Interactúa con los miembros del equipo y controle las revisiones con herramientas estandarizadas de gestión de datos.
- ◆ Evita errores y repeticiones utilizando herramientas integradas de análisis de la tensión y el movimiento (SOLIDWORKS, 2021).

2.12.3 Tinkercad

Tinkercad es una herramienta de modelado 3D, gratuita y basada en la web, creada para aficionados, docentes y estudiantes de todas las edades. Se pueden crear decoraciones para el hogar, juguetes, modelos de Minecraft y mucho más (Autodesk, 2020).

2.12.4 Ultimaker Cura

Esta es una aplicación diseñada para impresoras 3D, en la que se pueden modificar los parámetros de impresión y después transformarlos a un código para poder imprimir. Fue creada por David Braan, que después de un tiempo trabajaría para Ultimaker, una empresa dedicada al diseño y fabricación de impresoras 3D.

Para que el software se pueda mantener, se hizo disponible bajo una licencia LGPLv3. El desarrollo está alojado en GitHub. El software es utilizado por más de un millón de usuarios en todo el mundo y es del software de impresión 3D más usado en las impresoras 3D.

Ultimaker Cura se caracteriza por proporcionar una interfaz gráfica para preparar modelos para impresión 3D, la cual se ajusta según el modelo y el programa determina el escenario de la impresora 3D durante la aplicación secuencial de cada capa.

En el caso más simple, es suficiente importar el modelo en uno de los formatos compatibles (STL, OBJ, X3D, 3MF, BMP, GIF, JPG, PNG), seleccionar la configuración de velocidad y calidad y enviar el trabajo de impresión.

Existen complementos para la integración con SolidWorks, Siemens NX, Autodesk Inventor y otros sistemas CAD. El motor CuraEngine se utiliza para traducir el modelo 3D al conjunto de instrucciones de la impresora 3D.

De las principales características de Ultimaker Cura se destacan las siguientes:

- ◆ Software de código abierto multiplataforma, disponible de forma totalmente gratuita
- ◆ Cuenta con dos tipos de uso; el modo recomendado o el modo personalizado para configurar más de 300 configuraciones, para un control máximo
- ◆ Cuenta con perfiles configurados, que hacen que la configuración de hardware y material sea simple y rápida, y logra resultados confiables y profesionales
- ◆ Soporte inmediato para formatos de archivo STL, OBJ, X3D y 3MF

- ♦ Capacidad de poder ampliar su funcionalidad principal con complementos
- ♦ Capacidad de poder para administrar una o más impresoras Ultimaker con capacidad de red desde una sola interfaz

Actualmente el software se encuentra en su versión 4.6.1 la cual es básicamente una actualización de emergencia para la versión 4.6 y en ella se proponen nuevos perfiles estándar que automatizan el ajuste teniendo en cuenta el uso de materiales como policarbonato, nylon, CPE (poliestireno) y CPE +.

Además de que la interfaz proporciona la visualización de scripts activos para el procesamiento posterior y se agregó una configuración para expandir todos los agujeros al agregar desplazamiento en cada capa, lo que le permite aumentar o disminuir manualmente los agujeros para compensar la expansión horizontal. En la ventana de vista previa, se agrega la capacidad de hacer que el material auxiliar sea transparente. (Darkcrizt DesdeLinux, 2019)

2.13 Interruptores

El interruptor mecánico en electrónica. Es un dispositivo eléctrico cuya función es detener el flujo de corriente eléctrica entre dos puntos. tradicionalmente, un interruptor sólo tiene dos posiciones: apagado y encendido (Busca Palabra, 2021).

Los elementos de control o de maniobra son dispositivos que nos permiten abrir o cerrar el circuito cuando lo necesitamos. Estos son algunos tipos de interruptores (Búa, 2014).

2.13.1 Interruptores

Un interruptor (simple), permite abrir o cerrar un circuito y permanece en la misma posición hasta que volvemos a presionar.

Un interruptor doble o bipolar es un interruptor que abre y cierra dos circuitos al mismo tiempo (Búa, 2014).

En la ilustración 2.36 Se muestra un interruptor.



Ilustración 2.37 Interruptor.

2.13.2 Pulsadores

Un pulsador permite abrir o cerrar el circuito solo mientras estemos actuando sobre él. Cuando dejamos de presionar vuelve a su posición inicial.

Pulsador Normalmente Abierto (NA): En el estado de reposo el circuito está abierto, y se cierra cuándo se presiona.

Pulsador Normalmente Cerrado (NC): En el estado de reposo el circuito permanece cerrado, y se abre cuándo se presiona (Búa, 2014).

En la ilustración 2.37 se muestra un pulsador.



Ilustración 2.37 Pulsador.

2.13.3 Conmutadores

Los conmutadores o circuito de varias posiciones. Un conmutador es un elemento que establece una asociación entre una entrada y una salida de las múltiples que se tiene. Esta conexión perdura en el tiempo, hasta que volvemos a accionar el conmutador. El conmutador de dos posiciones tiene 3 patillas. La conexión de en medio es la común, y las patillas A y B son las posibles salidas.

Conmutador doble o bipolar: El elemento que puedes ver en la ilustración 2.38 se trata de un conmutador 2 circuitos y dos posiciones (6 contactos). Consiste en dos conmutadores de dos posiciones que se activan al mismo tiempo (Búa, 2014).



Ilustración 2.38 Conmutador.

2.13.4 Microinterruptores

Un microinterruptor o final de carrera es un componente que se acciona mediante una palanca empujada por un elemento en movimiento.

Según la forma de conectarlo, puede comportarse como conmutador o como pulsador, logrando seleccionar la posición inicial como Normalmente Abierta (NO o NA) o Normalmente Cerrada (NC).

Para ser utilizados, se debe identificar las patillas del microinterruptor. La pata que está más cerca del apoyo de la palanca es el común, que se debe conectar siempre. La de en medio es la normalmente abierta, y la última es la normalmente cerrada (Búa, 2014).

En la ilustración 2.39 Se muestra un microinterruptor.



Ilustración 2.39 Microinterruptores.

2.13.5 Relés

Uno relé es un interruptor automático controlado por la electricidad. Los relees permiten abrir o cerrar circuitos eléctricos sin la intervención humana. El relé es el elemento que da la orden para el funcionamiento de algún motor, luces, y multitud de sistemas automáticos.

Su funcionamiento es el siguiente:

Cuando se hace pasar corriente eléctrica a través de la bobina o electroimán, este genera un campo magnético a su alrededor, y atrae la armadura que, con su movimiento, hace que los contactos cambien de posición.

De esta manera estamos actuando sobre un conmutador por medio de la corriente eléctrica, sin contacto físico.

El símbolo del relé varía en función del número de contactos que tenga. Puede ser un interruptor, un conmutador unipolar, un conmutador doble o bipolar, en la ilustración 2.40 Se muestra un relé (Búa, 2014).



Ilustración 2.40 Relevador.

2.14 Acelerómetros y giroscopios

Los acelerómetros y los giroscopios son dos componentes que se complementan, pero que son:

- ◆ Un giroscopio es un dispositivo que sirve para medir, mantener o cambiar la orientación en el espacio de algún aparato o vehículo.

Su diseño consiste en un disco de rotación libre llamado rotor, montado en un eje giratorio en el centro de una rueda más grande y estable. A medida que el eje gira, el rotor permanece estacionario (Castro, 2020).

Un ejemplo de este se muestra en la ilustración 2.41.

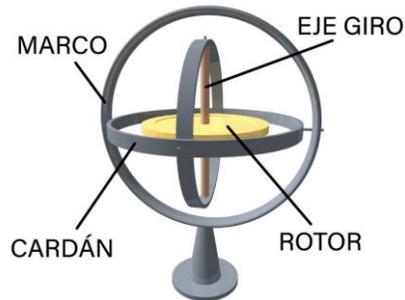
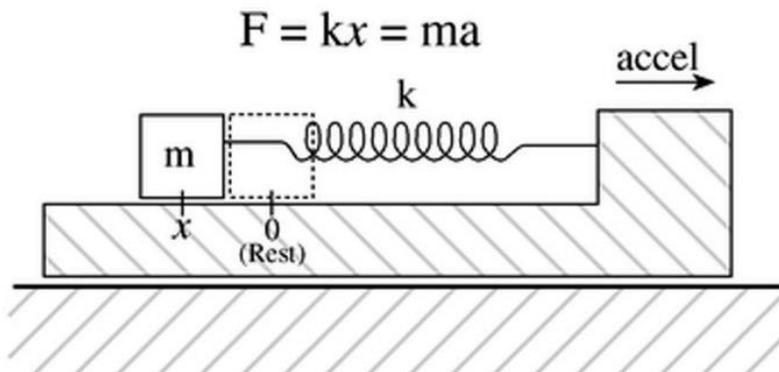


Ilustración 2.41 Giroscopio.

- ◆ Un acelerómetro es un dispositivo compacto diseñado para medir la fuerza de aceleración, ya sea causada por la gravedad o por el movimiento.

El acelerómetro detecta el movimiento, lo que lo convierte en una herramienta ideal para capturar con precisión el cambio posición y la orientación del mismo (Castro, 2020).

En la ilustración 2.42 se muestra el funcionamiento del giroscopio.



La masa (M) se sujeta mediante un resorte (K) y cuando está en reposo está en la posición 0 (reposo). Si tiras del marco hacia la derecha, la masa permanecerá estacionaria durante algún tiempo y luego se moverá con el movimiento. Este retardo es capturado y se convierte en aceleración

Ilustración 2.42 Funcionamiento de un Acelerómetro.

Por lo tanto, si unimos un acelerómetro con un giroscopio estos pueden ser más precisos para movimientos más bruscos como el que se da en un celular al caminar.

2.14.1 MPU6050

MPU6050 es un dispositivo de rastreo de movimiento de 6 ejes basado en MEMS. Tiene un giroscopio y sensores de acelerómetro en el chip junto con un sensor de temperatura. MPU6050 es un dispositivo digital. Este módulo es de tamaño muy pequeño, tiene requisitos de bajo consumo de energía, es muy preciso, tiene alta repetibilidad, alta tolerancia a los golpes, tiene una programabilidad de rendimiento específico para cada aplicación y su precio es barato. El MPU6050 puede ser fácilmente conectado a otros sensores como magnetómetros y microcontroladores (descubrearduino.com, 2021). En la ilustración 2.43 se muestra el sensor MPU6050 con sus ejes y rotaciones detectables.

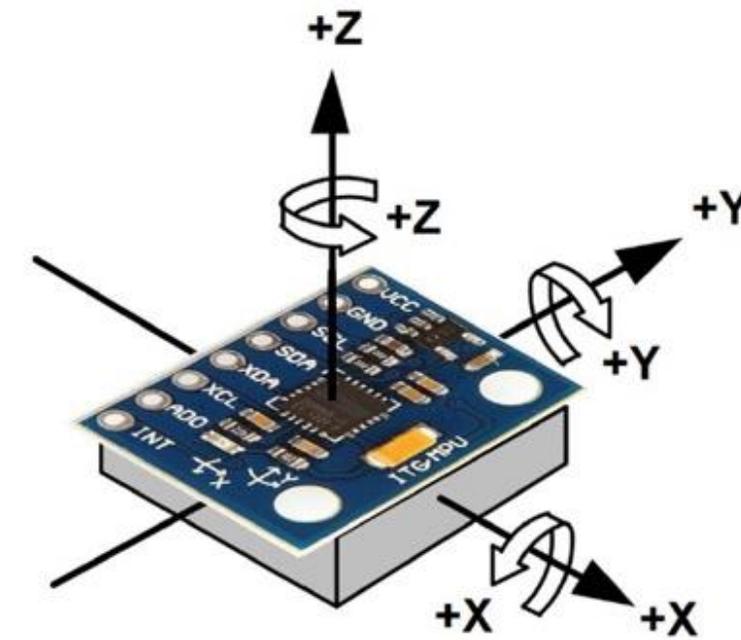


Ilustración 2.43 MPU6050

El módulo MPU6050 está compuesto por los siguientes bloques y funciones:

- ◆ Un sensor giroscópico de 3 ejes MEMS con tres ADC de 16 bits y acondicionamiento de señal.
- ◆ Un sensor acelerómetro MEMS de 3 ejes con tres ADC de 16 bits y acondicionamiento de señal.
- ◆ Un motor de procesador de movimiento digital en el chip.
- ◆ Interfaces primarias de comunicación digital I2C.
- ◆ Interfaces auxiliares I2C para la comunicación con sensores externos como el magnetómetro.
- ◆ Reloj interno.
- ◆ Registros de datos para almacenar los datos de los sensores.

- ◆ Memoria FIFO que ayuda a reducir el consumo de energía.
- ◆ Interrupciones programables por el usuario.
- ◆ Un sensor de temperatura de salida digital.
- ◆ Autotest para el giroscopio y el acelerómetro.
- ◆ LDO y Bias.
- ◆ Bomba de carga.
- ◆ Registros de estado (descubrearduino.com, 2021).

EL módulo Acelerómetro MPU tiene un giroscopio de tres ejes con el que podemos medir velocidad angular y un acelerómetro también de 3 ejes con el que medimos los componentes X, Y y Z de la aceleración. Necesita una biblioteca llamada MPU605 para funcionar (Naylamp mechatronics, 2016).

El MPU6050 está disponible como un pequeño dispositivo de 4×4×0,9 mm. La estructura del MEMS está herméticamente sellada y unida al nivel de las obleas. MPU6050 está disponible como un paquete QFN de 24 pines que es el chip integrado que está en medio del módulo. La descripción de las clavijas de este módulo es la siguiente:

- ◆ Pin-1- CLKIN- es la entrada opcional del reloj de referencia externo. Esta clavija se conecta a tierra cuando no está en uso.
- ◆ Pin-2, el Pin-3, el Pin-4, el Pin-5 son los pines NC. Estos pines no están conectados internamente.
- ◆ Pin-6, AUX_DA, es el pin de datos de la serie maestra de I2C. Este pin se usa para conectar sensores externos.
- ◆ Pin-7, AUX_CL, es el reloj serial maestro de I2C. Esta clavija se usa para conectar sensores externos.
- ◆ Pin-8, VLOGIC, es el pin de voltaje de suministro de E/S digital.
- ◆ Pin-9, AD0, es el pin LSB de dirección de esclavo de I2C.
- ◆ Pin-10, REGOUT, es la conexión del condensador del filtro regulador.
- ◆ Pin-11, FSYNC, es la entrada digital de sincronización de cuadro. Esta clavija se conecta a tierra cuando no se usa.
- ◆ Pin-12, INT, es el pin de salida digital de interrupción.
- ◆ Pin-13, VDD, es el pin de voltaje de la fuente de alimentación.
- ◆ Pin-14, Pin-15, Pin-16, Pin-17, son los pines NC. Estos pines no están conectados internamente.
- ◆ Pin-18, GND, es la tierra de la fuente de alimentación.
- ◆ Pin-19 y el Pin-21 son los pines RESV. Estos pines están reservados.
- ◆ Pin-20, CPOUT, es la conexión del condensador de la bomba de carga.
- ◆ Pin-22, es RESV, el pin reservado.

- ◆ Pin-23, SCL, es el reloj de serie I2C.
- ◆ Pin-24, SDA, es el pin de datos en serie de I2C

En la ilustración 2.44 se muestra los pines que trae el módulo comercial y el significado de cada una de ellas.

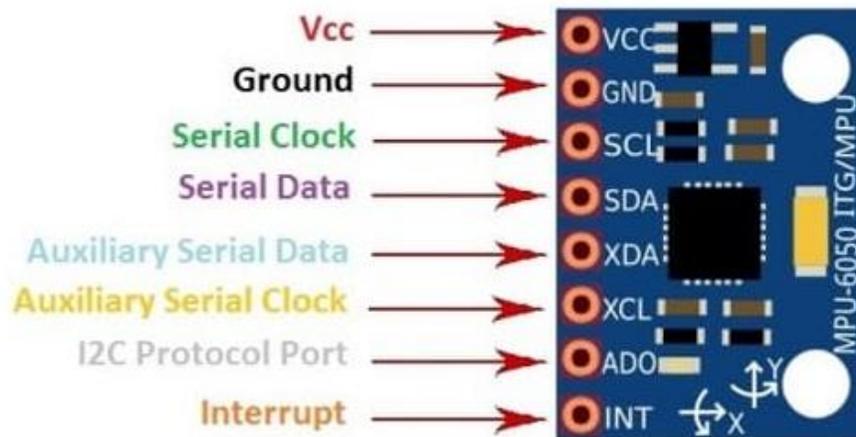


Ilustración 2.44 Puertos del MPU6050

Las que utiliza para la conexión al Arduino son las de alimentación junto con SDA y SCL, donde se emplea la conexión I2C, los demás puertos pueden ser usados según las exigencias del usuario (descubrearduino.com, 2021).

Algunas de las especificaciones del MPU6050 se muestran a continuación:

- ◆ El MPU6050 tiene un giroscopio de 3 ejes, un acelerómetro de 3 ejes y un procesador de movimiento digital integrado en un solo chip.
- ◆ Funciona con la fuente de alimentación de 3V-5V.
- ◆ MPU6050 utiliza el protocolo I2C para la comunicación y la transferencia de datos.
- ◆ Este módulo tiene incorporado un ADC de 16 bits que proporciona una gran precisión.
- ◆ MPU6050 puede ser interconectado con otros dispositivos IIC como los magnetómetros.
- ◆ MPU6050 también tiene un sensor de temperatura incorporado.
- ◆ El bus de sensores I2C le ayuda a recoger datos directamente de la brújula externa de 3 ejes, que utiliza para proporcionar una salida completa de 9 ejes de MotionFusion.
- ◆ Para los usuarios, la MPU6050 elimina la necesidad de selección, calificación e integración a nivel de sistema de dispositivos discretos.
- ◆ Usando su puerto I2C, los sensores no inerciales como el sensor de presión pueden ser interconectados.
- ◆ MPU6050 consta de tres ADC de 16 bits para digitalizar las salidas del giroscopio 0 y tres ADC de 16 bits para digitalizar las salidas del acelerómetro.

- ◆ Se dispone de una gama de giroscopios y de una gama de acelerómetros programables por el usuario para el seguimiento de precisión de movimientos tanto rápidos como lentos.
- ◆ Se dispone de un búfer FIFO de 1024 bytes en el chip que ayuda a reducir el consumo de energía del módulo.
- ◆ La necesidad de agrupar frecuentemente la salida del sensor se reduce al mínimo con la ayuda del DMP en el chip.
- ◆ El MPU6050 también tiene un oscilador en el chip con una variación de $\pm 1\%$.
- ◆ El MPU6050 tiene filtros de paso bajo para el giroscopio, el acelerómetro y el sensor de temperatura.
- ◆ El pin de referencia VLOGIC se utiliza para establecer los niveles lógicos de la interfaz I2C.
- ◆ El rango programable por el usuario del giroscopio presente en el MPU6050 es de ± 250 , ± 500 , ± 1000 y $\pm 2000^\circ/\text{seg}$.
- ◆ La imagen, el vídeo y la sincronización del GPS son compatibles con la clavija de sincronización externa del giroscopio.
- ◆ Este giroscopio ha mejorado el rendimiento del ruido de baja frecuencia.
- ◆ El giroscopio necesita 3,6mA de corriente para funcionar.
- ◆ El filtro de paso bajo del giroscopio es programable digitalmente.
- ◆ El acelerómetro presente en el MPU6050 opera en 500 μ A de corriente.
- ◆ El rango programable a escala completa de este acelerómetro es de $\pm 2g$, $\pm 4g$, $\pm 8g$ y 16g.
- ◆ El acelerómetro también puede detectar la orientación, la detección de golpecitos.
- ◆ Las interrupciones programables por el usuario están presentes para el acelerómetro.
- ◆ Entre los ejes del acelerómetro y del giroscopio hay una sensibilidad mínima en los ejes cruzados.
- ◆ Para comunicarse con todos los registros se utiliza el modo rápido de 400kHz I2C.
- ◆ El DMP presente en el MPU6050 soporta el procesamiento de movimiento en 3D y los algoritmos de reconocimiento de gestos.
- ◆ Se proporciona una lectura de ráfagas para el procesador del sistema. Después de leer los datos del FIFO, el procesador del sistema entra en modo de suspensión de baja potencia mientras que el MPU recoge más datos.
- ◆ Las interrupciones programables admiten características como el reconocimiento de gestos, el desplazamiento, el zoom, el desplazamiento, la detección de pulsaciones y la detección de shack.
- ◆ El MPU6050 también tiene una entrada de reloj externo opcional de 32,768kHz o 19,2Mhz (descubrearduino.com, 2021).

Capítulo 3. Desarrollo experimental

Para la realización del proyecto se planteó: crear prototipos funcionales para el control de un SAAC (Sistema Aumentativo y Alternativo de la Comunicación), como primera propuesta se tuvo como objetivo hacer diferentes modelos con interruptores, después con el desarrollo del proyecto se propusieron modelos más complejos los cuales fueron los módulos basados en la detección del soplo, y un módulo con acelerómetro y giroscopio capaz de detectar el movimiento de una persona con discapacidad.

Para los módulos con interruptores, se acordó crear una carcasa capaz de implementarse a diferentes tipos de discapacidades o las opciones que fueran más cómodas para las personas en esta situación, una de estas opciones fue una carcasa que pudiera activar el interruptor con el hecho de apretar la mano, otra opción fue poder activar el interruptor por medio de la muñeca con el propio peso de la mano y también algún modelo que se activara por medio de la mano.

Para el módulo de detección por soplo, se acordó desarrollar un módulo capaz de detectar el soplo de una persona, que tenga la opción de ser más sensible al ruido o no, además de controlar la salida para activarla por un determinado tiempo, teniendo un led indicador, un switch apagador, una pantalla LCD para indicar los valores de sensibilidad y tiempo, un sensor con su debida adaptación para interactuar con él y la carcasa que contenga la placa de circuito, con los diferentes componentes y sus debidas conexiones. Para controlar el sensor se desarrolló un sistema de amplificación para el sensor, en este caso el micrófono y un programa para el control del sistema por medio de un microcontrolador.

Para el módulo de detección de movimiento con al acelerómetro y el giroscopio, se acordó crear un sistema capaz de detectar el movimiento en una distancia corta como un gesto, para mandar una señal. Para desarrollar el módulo, se hizo un circuito controlado por un microprocesador capaz de cambiar el tiempo de la señal, la sensibilidad del sensor, monitorear la sensibilidad y el tiempo por medio de una pantalla LCD, también monitorear la señal de salida por medio de un led, con otro led señalar la buena conexión con el sensor (acelerómetro y giroscopio) y controlar los valores de sensibilidad y tiempo por medio de dos potenciómetros. Como parte del desarrollo del módulo se hizo una carcasa, la cual, se modelo en 3D y se imprimió, para que el módulo tenga sus conexiones en carcasa, un apagador y los componentes necesarios para su funcionamiento.

Por lo tanto, el proyecto se dividió en tres diferentes dispositivos dos de ellos con variantes. Los tres dispositivos fueron:

- ◆ Módulo con interruptores
- ◆ Módulo con sensor de soplo (Micrófono)
- ◆ Módulo con sensor de movimiento (Acelerómetro)

Los módulos desarrollados, basaron su interacción con el SAAC hecho en la FES Aragón (Martínez, 2020), el cual tiene un puerto del GPIO para poder interactuar; este puerto funciona con lógica inversa a un voltaje de 3.3 volts, todo fue programado en Python para correr en una Raspberry Pi 3 o 4. La señal de salida debe tener una duración inferior al tiempo

que dura el cursor para desplazarse sobre los elementos de la interfaz gráfica, y no hay una duración mínima definida.

3.1 Módulos basados en interruptores mecánicos

Dadas las características del SAAC ya descritas, hace que, para la interacción entre usuario y sistema, el uso de interruptores mecánicos sea una opción económica, simple y funcional desde lo técnico, pero con limitantes importantes, ya que un tipo de pacientes a quienes se dirige el SAAC, son aquellos con problemas de motricidad fina en sus manos, y que presionar un interruptor podría no ser posible. Gracias las ventajas del uso de interruptores, se pensó en que, a pesar de las desventajas, el diseño de módulos ergonómicos para diferentes formas de sujeción, podrían ser útiles para algunos pacientes, consiguiendo así un modo de interacción de bajo costo.

Por lo mencionado, se realizaron variantes de módulos de interacción basados en interruptores. La diferencia principal es el diseño de la carcasa y el tipo de interruptor, lo cual será descrito en los siguientes apartados.

En la ilustración 3.1 se muestra el esquema del funcionamiento básico de los interruptores, todos los módulos con interruptores desarrollados incorporan este mismo principio, al pulsar el interruptor deja pasar el flujo de corriente.



Ilustración 3.1 Esquema de Pulsadores.

3.1.1 Módulo para palma-muñeca de la mano

Para este dispositivo, se planteó la idea de que será útil para personas que estén en cama o reposo y tengan movimiento en la mano o muñeca, de tal manera que con poco esfuerzo puedan apretar un botón, incluso con el mismo peso del brazo recargado en la muñeca puedan accionar el pulsador. Para este caso se decidió hacer un modelo grande para más comodidad.

3.1.1.1 Modelo 3D

La realización de la carcasa fue hecha en SolidWorks, dado que el modelo sería materializado con una impresora 3D, fue necesario considerar en modelo, las limitaciones de la impresora, lo cual llevo a que, para el diseño, se trabajase en 8 piezas, dos de ellas son la base, reposa codo, donde va incrustado el pulsador y los dos tubos con su tapa.

Se tomaron las correspondientes medidas del botón para que no hubiera ningún problema a la hora de ensamblarlo, como parte del diseño, se optó por levantar la parte de adelante para que la persona pueda acomodar su brazo y se sienta más cómodo.

La base se partió en dos, las cuales se unen con tornillos en un borde para a completar la figura y que no se desarme.

La ilustración 3.2 muestra la base del codo la cual tiene cuatro pilares con agujeros para unir la pieza de arriba que es el soporte del codo. La base del codo mide 120x140mm.

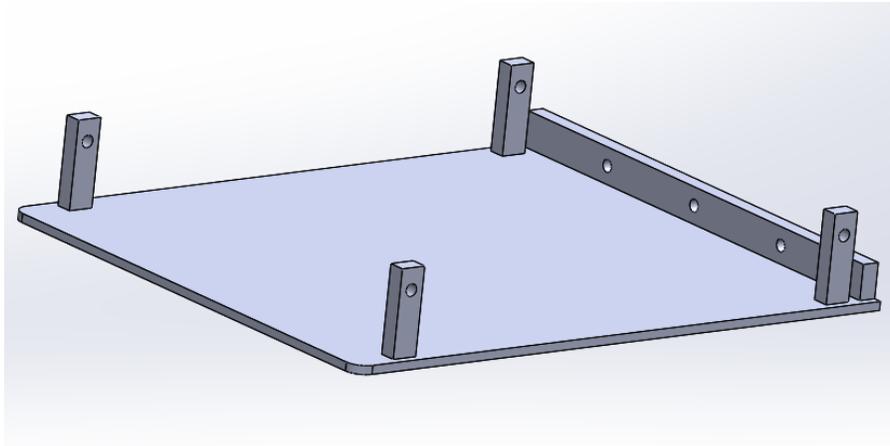


Ilustración 3.2 Base del Codo.

En la ilustración 3.3 se muestra la base de la pieza que lleva el botón, esta es diferente ya que tiene tres paredes, un borde con la unión a la otra base con sus respectivos tornillos, dos de las paredes tienen agujeros que permiten el paso de un tubo para hacer soporte al momento de elevar el botón. Esta pieza mide 200x120mm en la base y una altura de 183mm, tiene los bordes redondeados para no crear alguna incomodidad con el filo de las esquinas. La pared de enfrente es más pequeña ya que es solo para cubrir cuando se alza la pieza del botón.

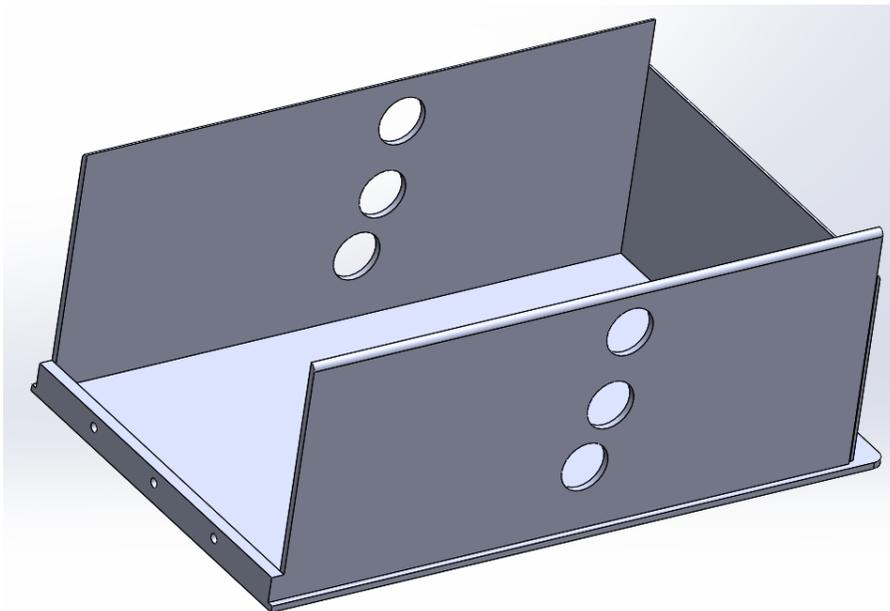


Ilustración 3.3 Base del botón.

Una vez obtenida la base se procedió a hacer las tapas o las piezas de arriba. En la ilustración 3.4 se muestra el codo, que es la pieza que va arriba de la base del codo, esta lleva cuatro agujeros, que es donde pasa el tornillo para sujetar la base del codo, también dispone de una

perforación que va de lado a lado para acoplar con un tubo la pieza del botón y tenga movilidad girando concéntricamente. Por otro lado, baja con un redondeo y deja una zona más abajo, la cual, esta echa para acoplarle algún colchón o esponja. Este mide 120x140 mm con una altura máxima de 85mm y las esquinas están redondeadas. Además, en un costado tiene un agujero por el cual pasan los cables de conexión al SAAC.

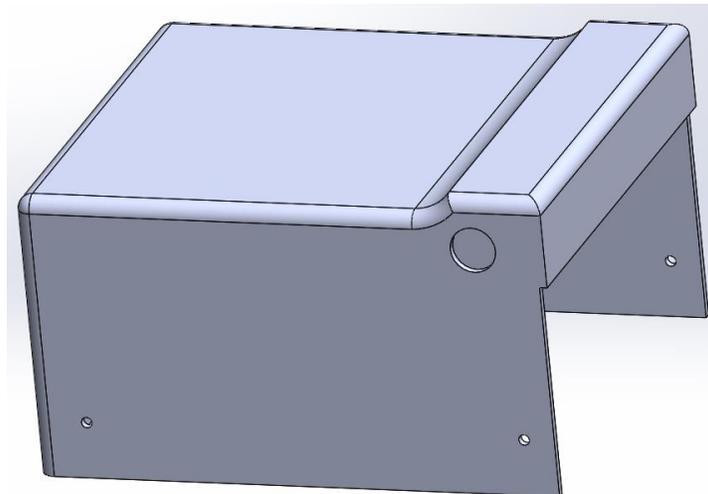


Ilustración 3.4 Pieza Codo.

La pieza que lleva el botón se muestra en la ilustración 3.5, la cual tiene el agujero echo a la medida para el pulsador o botón que fue diseñada, tiene dos pestañas salidas con un agujero para unirse por medio de un cilindro con la tapa del codo, la cual le permite girar concéntricamente, una zona más baja como hueco para acoplar alguna esponja o cojín y un agujero de lado a lado en el costado para ajustar la altura de inclinación. Mide 228.83x120 mm es la pieza más grande.

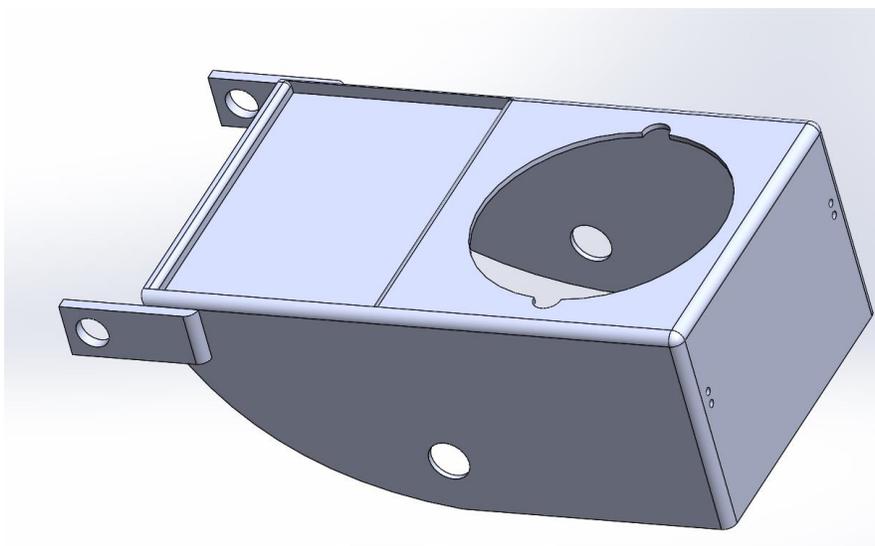


Ilustración 3.5 Pieza Botón.

Por último, se crearon los tubos para dar movimiento y soporte al dispositivo. En la ilustración 3.6 A se muestra el tubo más pequeño con su tapa, la cual une a el codo con la

pieza del botón y permite que gire. Las dos llevan un tope para que no se salga y se unen entre ellas.

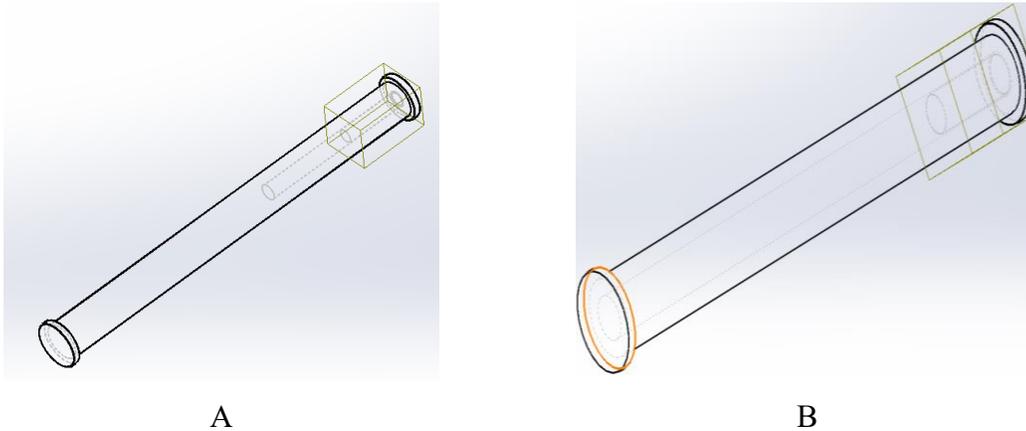


Ilustración 3.6 Tubos Conectores.

El otro tubo soporta a la pieza que contiene al botón y la fija a la base del botón, esta pieza se puede sacar y acomodar en las tres alturas que contiene la base del botón para inclinar el ajuste de la mano. Se muestra en la ilustración 3.6 B el tubo y la tapa.

Como parte final se muestra el ensamblaje de todas las piezas en la ilustración 3.7 A, después en la B se muestra con la primera inclinación y en la C se muestra con la máxima altura.

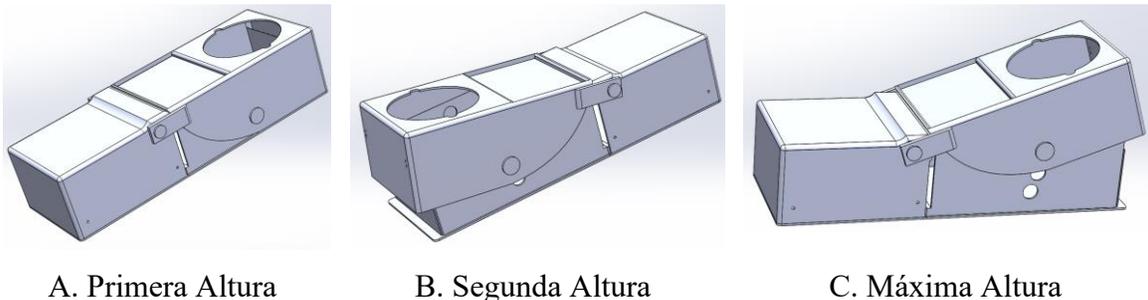


Ilustración 3.7 Botón grande diferentes alturas.

El siguiente paso fue exportarlas a Ultimaker Cura (Software de impresión 3D) y proceder a su impresión.

3.1.1.2 Modelo real

Como parte principal de este módulo, se muestra en la ilustración 3.8 el botón que se tomó como referencia para construir este módulo, el botón se divide en tres partes, una donde se encuentra el switch con sus conexiones y el led que incorpora, la segunda es la base que sirve para sujetar el botón con la tuerca que lleva y por último el botón exterior.

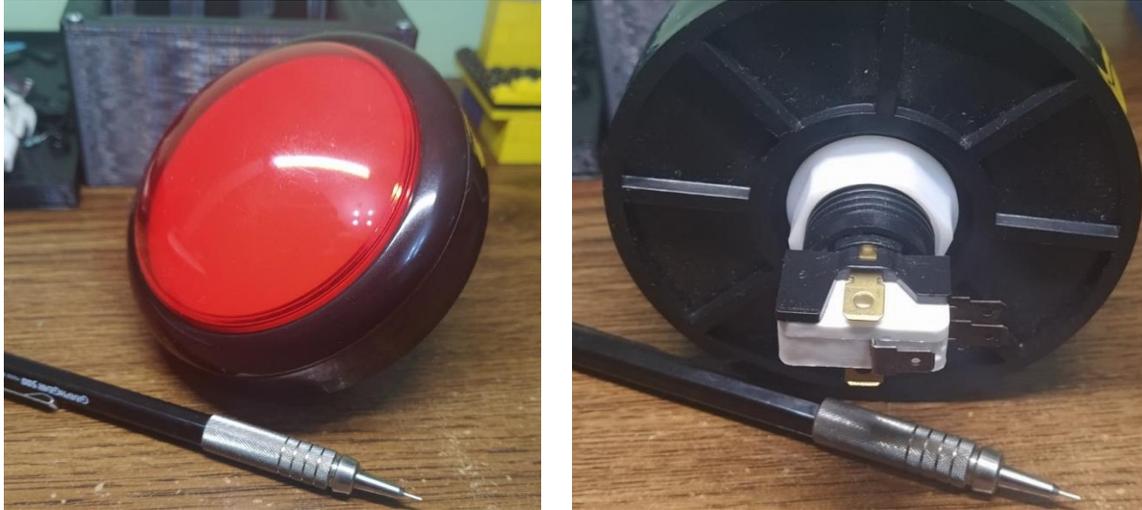


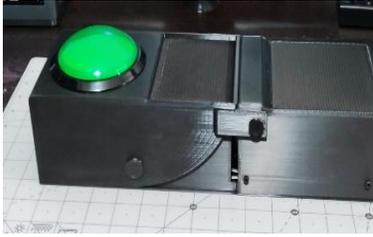
Ilustración 3.8 Pulsador.

Una vez que se tenían los modelos 3D, se exportaron en formato “stl” el cual permite abrirlo en el software Ultimaker Cura, donde se mapeo el recorrido que tenía que hacer la impresora 3D y se exporto para imprimir en 3D como se muestra en la ilustración 3.9.



Ilustración 3.9 Impresión del botón grande.

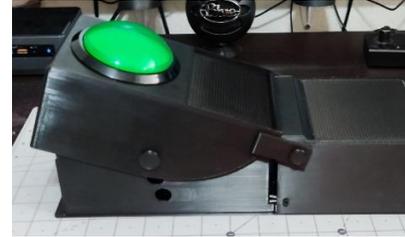
Una vez impresas cada una de las 8 piezas que conformaban el botón grande, se procede al ensamblaje, el cual conlleva atornillar las partes sueltas y unir piezas por medio de los tubos ya mencionados. Como resultado se muestra en la ilustración 3.10 A el botón grande ya ensamblado en su posición básica, después en la ilustración 3.10 B se muestra el botón grande con la primera altura, la cual se hace moviendo el tubo al segundo agujero y, por último, en la posición más alta, donde el tubo se recorrido en el último agujero se muestra en la ilustración 3.10 C.



A. Posición normal



B. Altura Media



C. Máxima Altura

Ilustración 3.10 Botón grande Real.

Como se muestra en las ilustraciones tiene dos plataformas más bajas en la parte de arriba, estas son para colocar algún colchón o soporte para mayor comodidad, además de que de un costado de la pieza del codo tiene un agujero para la salida de los cables que van conectados al botón. Pese a que tiene partes móviles estas son lo suficientemente rígidas para mantenerse firmes.

El tubo que conecta a la pieza del codo y la del botón, la pieza que tapa al tubo, está pegada para evitar que se desarme. El otro tubo se puede intercambiar de agujero para que se ajuste la altura.

3.1.2 Módulo para activación por pulgar de la mano

Este módulo se desarrolló con la idea de servir a las personas que pueden sujetar un pequeño tubo en la mano y tengan la capacidad de mover el dedo pulgar, es otra forma de implementar un pulsador, es menos masiva como la del botón grande, pero requiere la capacidad de interactuar con la mano, por lo que es diferente. Para este dispositivo básicamente se tienen dos variantes y se diferencian por el tipo de pulsador, los cuales veremos a continuación.

3.1.2.1 Modelo 3D de variante 1

Para el módulo de activación por pulgar de mano, de igual manera que los anteriores, se realizó una carcasa, utilizando el mismo software, SolidWorks y se dividió en 2 piezas en este caso, dado el tamaño del dispositivo.

Una vez habiendo medido el pulsador que se muestra en la ilustración 3.11 se procedió a hacer el diseño 3D.



Ilustración 3.11 Pulsador v1 con la tapa.

El primer paso fue hacer la tapa, la cual tiene el agujero para introducir el botón con bordes redondeados, para mayor comodidad, y cuatro pequeños pilares para unirlo con la base, la tapa de la variante 1 se muestra en la ilustración 3.12.

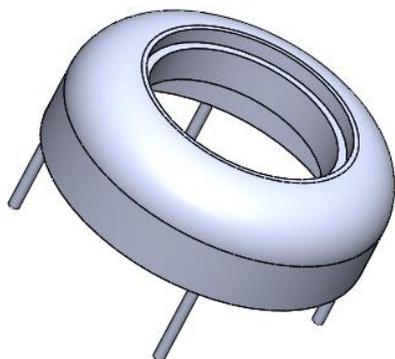


Ilustración 3.12 Tapa Botón Pulgar v1.

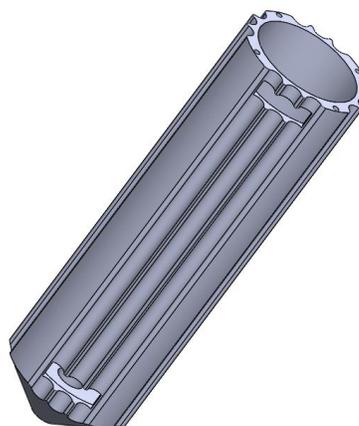


Ilustración 3.13 Base del botón del pulgar v1.

Para la base que se muestra en la ilustración 3.13, se hizo un tubo de aproximadamente 100 mm de altura, con la textura de los bordes redondeada, para tener más agarre, dos agujeros en los costados para tener un resorte que sujetara la mano y un agujero en la parte inferior para pasar los cables que van conectados al SAAC (al GPIO de la Raspberry Pi 4B). También tiene sus respectivos hoyos para la conexión con la tapa.

Una vez modelado las dos piezas, se ensamblaron en el software para comprobar fallos de modelado. Como siguiente paso, fue exportarlas a Ultimaker Cura e imprimirlos, una vez impresos, se hicieron los respectivos ensamblajes, cableado y componentes externos que se verán a continuación.

3.1.2.2 Modelo real de variante 1

Una vez modelado todas las piezas necesarias para la construcción de este módulo, se procedió a imprimirlas, primero exportando en formato “stl” y después mapeando en el software Cura Ultimaker para poder exportar e imprimir. En la ilustración 3.14 se muestra la impresión de una de las piezas.



Ilustración 3.14 Impresión del botón pulgar.

Una vez impresas todas las piezas, se procede al ensamblaje de las mismas, para el botón, antes de ser ensamblado completamente, se soldaron los respectivos cables que conectarán con el SAAC, estos tienen un largo de más de un metro para permitir más movilidad. Una vez soldado se aisló con silicón y después con cinta aisladora.

Una vez soldados y aislados se introdujo los cables por el hoyo que tiene en la base como se muestra en la ilustración 3.15.



Ilustración 3.15 Preensamblado del botón pulgar.

Ya probado el ensamblaje, se procedió a incorporar el resorte, que es, una tira con resorte que permite mayor sujeción a la mano, este va en los agujeros que tiene en los laterales.

Por último, se muestra la pieza completa, la cual se le hizo un trenzado de cables para mayor estética y comodidad, se le incorporaron dos conectores hembra para unir al SAAC, En los pines 26 y GND. En la ilustración 3.16 A, se muestra el botón de pulgar de la variante 1 ya completado, el cual tiene el pulsador, un resorte para sujeción en mano y cables de aproximadamente un metro para la conexión al SAAC, tiene textura anti derrape y bordes redondeados. En la ilustración 3.26 B, se muestra como es el agarre del botón pulgar con una mano.



A. Botón pulgar.



B. Agarre del botón pulgar.

Ilustración 3.16 (A y B) Botón Pulgar v1.

Este modelo, aunque sencillo es una solución muy básica y económica para todas aquellas personas que aún tienen movimiento en la mano y más en el dedo pulgar, este modelo es capaz de sostenerse en la mano sin hacer un agarre, gracias a el resorte que se ajusta a la mano.

3.1.2.3 Modelo 3D de variante 2

Para este módulo de activación por pulgar de mano, se realizó un prototipo diferente, llamado variante 2, el cual mejoraba algunas de sus características, pero hace el mismo funcionamiento que el de la variante 1.

En este módulo de igual manera se realizó el diseño de su carcasa con el programa de modelado 3D SolidWorks, cambiando completamente el diseño del botón, dado a que el pulsador que se incorpora es diferente, mejora el tacto y la capacidad de respuesta. En la ilustración 3.17 se muestra el nuevo pulsador.



Ilustración 3.17 Pulsador variante 2.

Una vez tomadas las medidas del nuevo pulsador, se procedió a realizar los diseños 3D de los modelos. Como el pulsador tenía una rosca, se tomó como referencia, para hacer el agujero de la carcasa, se dividió en dos piezas las cuales son: pieza del pulsador y pieza base.

En la ilustración 3.18, se muestra la pieza del pulsador la cual lleva un agujero grande con tope para detener el botón, esta redondeada en la parte de arriba para más comodidad, tiene un agujero para un resorte, se le dio una textura en las orillas para mejor el agarre y un agujero en la parte inferior donde se acopla la base.

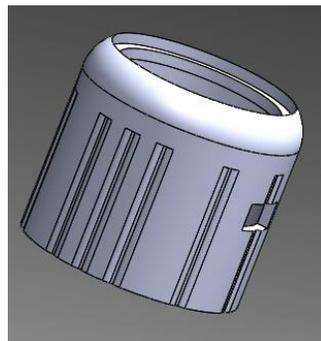
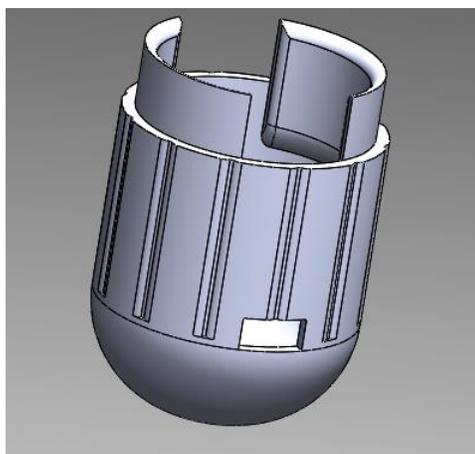
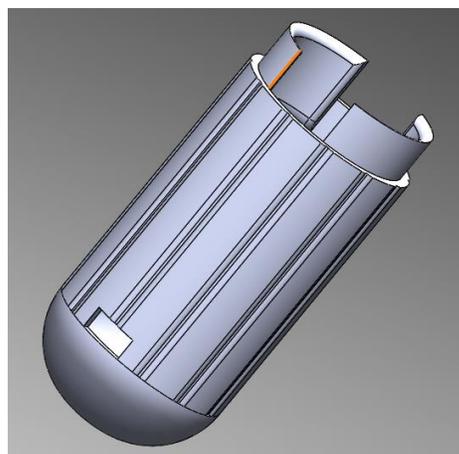


Ilustración 3.18 Pieza del pulsador v2.

Para este módulo de la variante 2, se tienen dos tamaños para las bases, aunque del mismo grosor y en forma de cilindro. El pequeño mide 82mm y el grande mide 113mm de altura. Tienen las mismas características que son: textura a los lados para mejor agarre, un agujero en la parte de abajo para los cables, un agujero en el lateral para el resorte y un terminado redondeado en la parte inferior, así como el agujero principal en la parte superior con dos pestañas que entran a presión en la pieza del pulsador permitiendo conectarlas sin la necesidad de un pegamento. La base pequeña se muestra en la ilustración 3.19 A y en la B se muestra la base grande.



A. Base pequeña

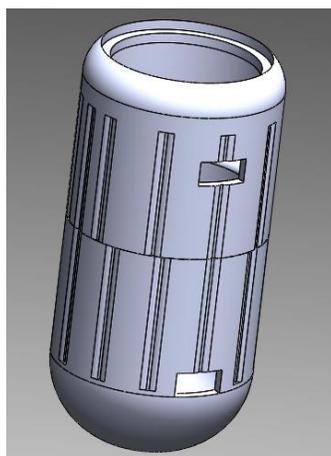


B. Base grande

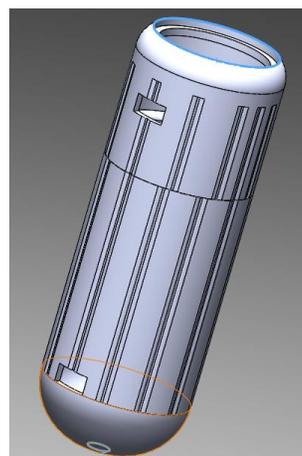
Ilustración 3.19 (A y B) Base módulo botón pulgar v2.

Para la base grande no cambiaron sus características excepto el tamaño, que fue el alargamiento del cilindro, el cual fue posible gracias a la edición de componentes en SolidWorks manteniendo sus características.

Los módulos se ensamblaron en el software y en la ilustración 3.20 A, se muestra el módulo de activación por pulgar de mano versión 2 corto y en la ilustración 3.20 B, se muestra el módulo de activación por pulgar en la mano versión 2 largo.



A. Corto



B. Largo

Ilustración 3.20 (A y B) Ensamblaje del módulo v2.

3.1.2.4 Modelo real de variante 2

Una vez modelados los módulos en 3D, se exporto en formato “stl”, para Ultimaker Cura donde se mapeo la impresión y se mandó a imprimir pieza por pieza.

Ya impresos se procedió a ensamblar el pulsador, sosteniéndolo con la tuerca que incluía el botón, después se soldaron los respectivos pines del módulo con los cables, una vez soldados fueron aislados con silicón y cinta de aislar.

Una vez soldado e impreso, se procedió a ensamblar las piezas no sin antes amarrar a su vez un resorte, que es la cinta de color negro elástica, que permite un mayor agarre del módulo como se muestra en la ilustración 3.21.



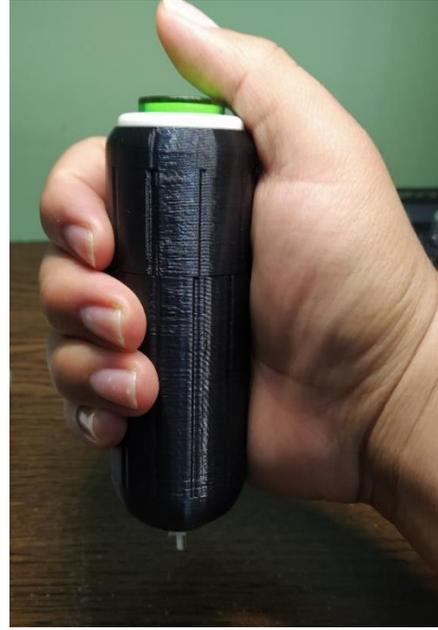
Ilustración 3.21 Pulsador Soldado y resorte.

Al módulo pequeño de esta variante, se le soldó una conexión que directamente es un cable de aproximadamente un metro de distancia, al final del cable contiene un molex hembra que se conecta con el SAAC, el cable está trenzado para mejor comodidad, como se muestra en la Ilustración 3.22 A, donde se ve el módulo por activación de pulgar de la mano, versión 2 corto completo.

Por otro lado, al módulo más largo de esta variante en la parte inferior se le agrego un conector macho de molex el cual sirve para conectar un cable con un molex en uno de sus extremos, este debe ser de dos pines, además de que es más fácil remplazar el cable de conexión con el SAAC ante cualquier daño. El módulo por activación de pulgar de la mano variante 2, largo se muestra en la ilustración 3.22 B, el cual se muestra como un ejemplo de uso.



A. Modelo Corto



B Modelo Largo

Ilustración 3.22 (A y B) Módulo por activación de pulgar de la mano versión 2.

Estos módulos tienen el mismo funcionamiento y se pueden acoplar dependiendo las necesidades de los pacientes ya sea tomando una variante u otra.

3.1.3 Módulo para activación por agarre de mano

Este módulo como su nombre lo indica, es para activación por agarre de mano, se pensó con la idea de que el paciente tenga la capacidad de mover la mano y pueda apretar para activar un botón o microinterruptor. Este módulo tiene dos microinterruptores que están ubicados a las orillas y cualquiera que se active primero mandara la señal, incluso si los dos se activan simultáneamente, en la ilustración 3.23 se muestra el esquema que se utilizó para conectar los dos interruptores al SAAC.

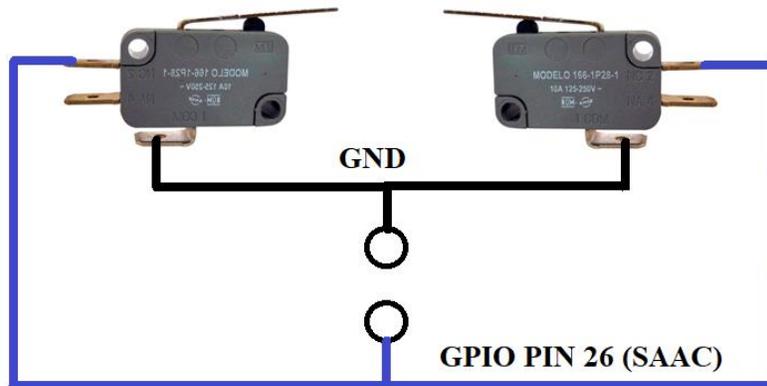


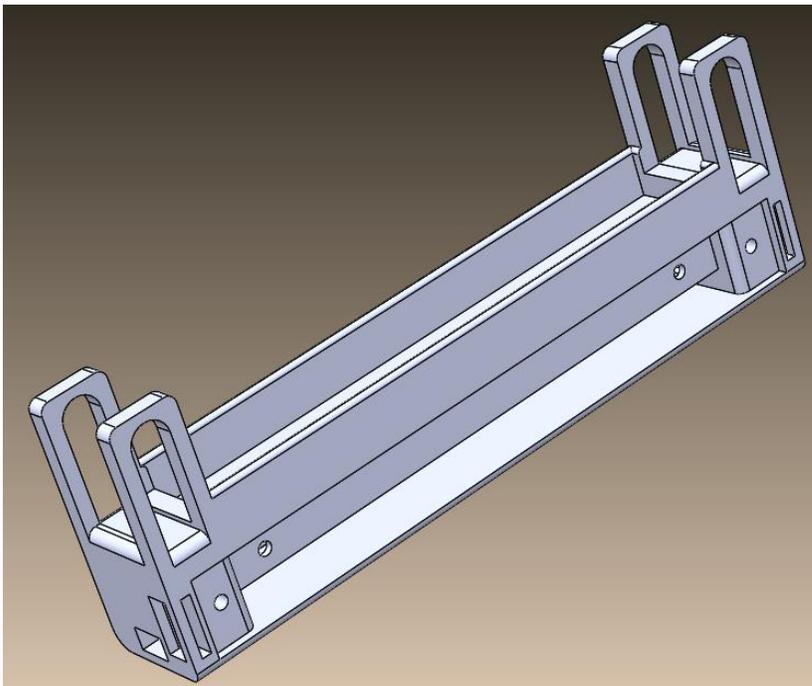
Ilustración 3.23 Esquema del módulo de agarre de mano.

Tiene dos piezas principales, donde en medio de estas lleva dos resortes que hacen el impulso hacia arriba para no mantener la señal activa o no mantener los interruptores apretados, a continuación, se verá el proceso de construcción.

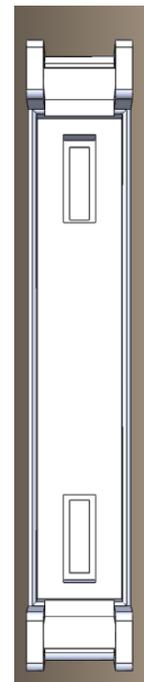
3.1.3.1 Modelo 3D

Igual que los módulos anteriores, los modelos 3D fueron hechos en el software de SolidWorks, este módulo se compone de siete piezas, de las cuales dos son las principales. Como primer paso se pensó en el diseño que estos podrían tener, el cual se hizo en cartón, y luego varios modelos en 3D, se escogió el interruptor adecuado para que en el momento de accionarlo sea inmediata su respuesta y tenga la sensación correcta de haberse activado.

Se midió el interruptor para poder diseñar la pieza. Una vez hecho esto se procedió a modelar la pieza base que contendría los interruptores, la base tiene una forma más compleja que los anteriores módulos, se basó principalmente en un rectángulo de 150 x 25 mm con una altura de 50 mm, se le hicieron diferentes vaciados o cortes para obtener las cavidades del interruptor, la pieza tapa y los cables. Tiene cuatro arcos en cada esquina para dar un tope a la pieza y que pueda pasar el cilindro por ellos, dos cavidades para los interruptores los cuales se fijan por el costado con tornillos, un vaciado en la parte superior donde encaja la pieza de la tapa o el gatillo, de un costado tiene tres vaciados, dos de los vaciados son donde se atora el resorte y uno donde van los tornillos de los interruptores con los cables. En la parte de abajo o de un lado tiene el orificio donde pasan los cables. El resorte se atora en los dos agujeros que pasan de un lado a otro en las dos esquinas. Tiene un pequeño vaciado para que entre la tapa que cubre los cables y sus respectivos agujeros para los tornillos. Por último, se redondearon las esquinas para mayor comodidad, todo esto se muestra en la Ilustración 3.24 A.



A. Base del gatillo.

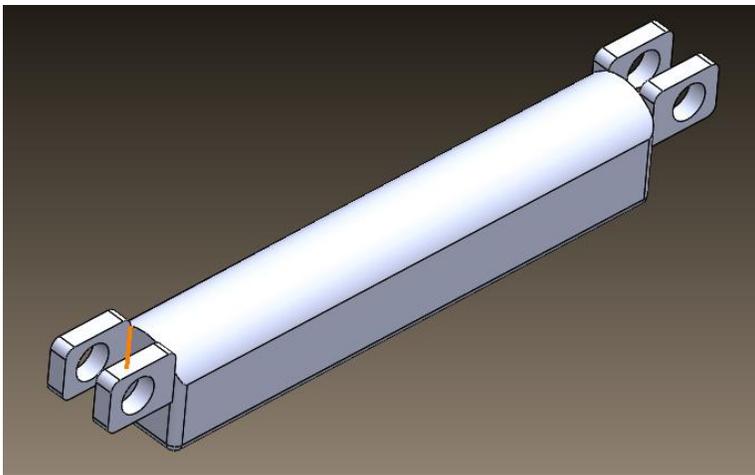


B. Vista por arriba.

Ilustración 3.24 Pieza base gatillo.

En la ilustración 3.24 B, se muestra la pieza base, vista de la parte de arriba. A los arcos se les dio más grosor que para que soportaran la presión ejercida del cilindro y del gatillo que se mostrara más adelante.

Con la pieza base, se procedió a realizar la pieza del gatillo o tapa, esta pieza de igual manera que la anterior se basó en un rectángulo, fue un poco más pequeño y a partir del rectángulo se extrudió, después se redondeó bastante para dar una forma curva y que se ajustara mejor a la mano. De los extremos se hicieron cuatro pestañas con un agujero al centro del mismo diámetro que el agujero de los arcos de la pieza base, cabe resaltar que no están en los costados de la pieza del gatillo, si no, en la cara de los extremos, las pestañas para finalizar fueron redondeadas para mayor comodidad. En la parte inferior de la pieza del gatillo se tiene una cara plana con dos agujeros en forma de dona, estos dos agujeros fueron hechos con el propósito de no permitir que los resortes se muevan, no son tan profundos y no atraviesan la pieza del gatillo. En la ilustración 3.25 A se muestra la pieza gatillo.



A. Pieza gatillo

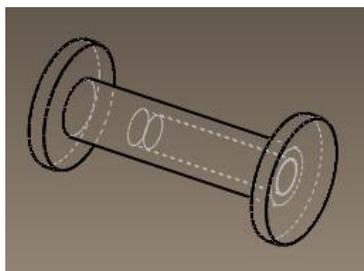


B: Vista por abajo

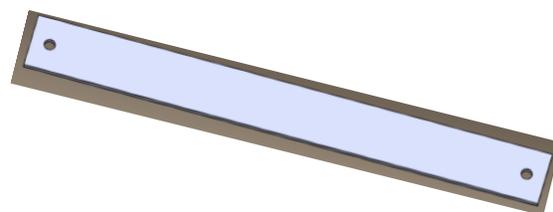
Ilustración 3.25 Pieza gatillo.

En la ilustración 3.25 B, se muestra la pieza gatillo, vista por abajo con sus agujeros que sostienen a los resortes, que consisten de dos agujeros en forma de dona.

También se creó una tapa, esta da acceso a los cables y a los tornillos que sujetan los interruptores, se creó a partir de un rectángulo y se extrudió en aproximadamente 2 mm, después se le hicieron los agujeros para los tornillos que coinciden con los que tiene la pieza de la base. En la ilustración 3.26 B se muestra la tapa.



A. Rodillo para el gatillo



B. Tapa

Ilustración 3.26 Tapa y Rodillos.

Por último, para unir estas dos piezas y además que tengan movilidad, se diseñaron los cilindros que van en las esquinas. Su diseño fue a partir de un cilindro y estos contienen una tapa que choca con los arcos de la pieza base del gatillo, el cilindro pasa por los agujeros de las esquinas y se une con el otro cilindro. En la ilustración 3.26 A, se muestran como son estas dos piezas, las cuales van por duplicado dado a que se ocupa una de cada lado.

Ya con todas las piezas, se hizo el ensamblaje en el programa y se obtuvo el modelo completo, el cual se verificó que cada una de sus piezas encajasen a la medida y sin problema. En la ilustración 3.27 se muestra el ensamblaje del módulo por activación por agarre de mano.

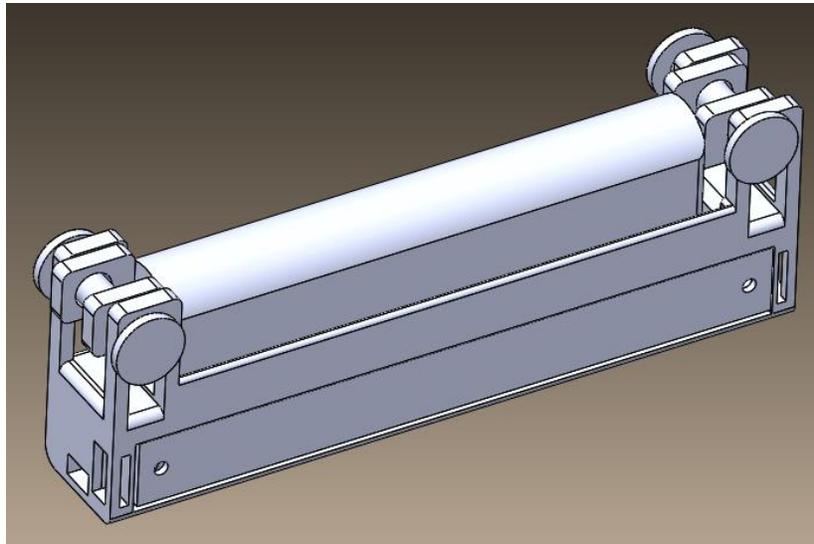


Ilustración 3.27 Ensamblaje del módulo de agarre por mano.

3.1.3.2 Modelo real

Una vez diseñadas todas las piezas de este módulo, se procedió a la construcción del mismo, para esto se exportó en archivo “stl”, el cual acepta el programa Ultimaker Cura que mapea la forma de imprimir la pieza en la impresora 3D, después se imprimieron todas las piezas.

Ya con las piezas construidas, se tomaron los interruptores como el que se muestra en la ilustración 3.28 y se procedió a soldar los cables de acuerdo al esquema de la ilustración 3.26. Los cables van trenzados y aislados, y tienen una extensión menor a un metro para poder conectarse al SAAC.



Ilustración 3.28 Interruptores parecidos al del módulo gatillo.

Se ensamblaron en la pieza base y se atornillaron para que no se puedan mover, además se amarró el resorte o listón elástico en las cavidades de los extremos, de tal forma que abracen la mano para sujetarse, esto se muestra en la ilustración 3.29.

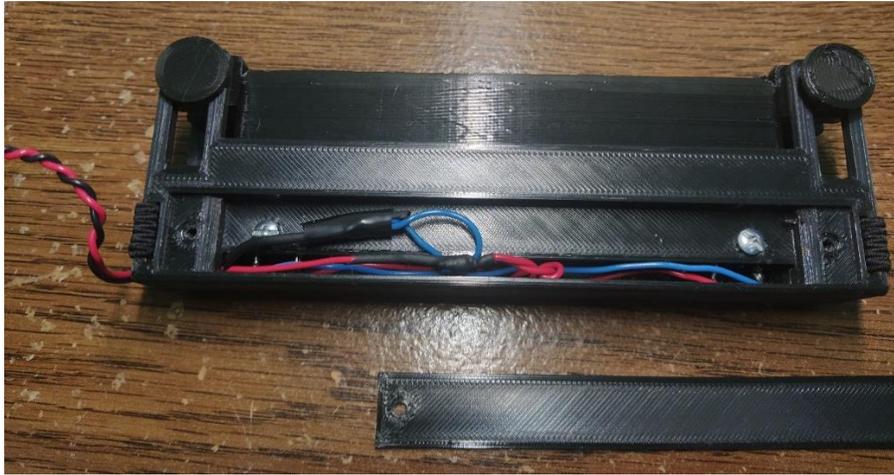


Ilustración 3.29 Módulo de agarre con la tapa abierta.

En la ilustración 3.30, se muestra la pieza del gatillo vista por debajo con los resortes pegados, y en la ilustración 3.31, se muestra la base del gatillo con los interruptores ensamblados, vista de arriba y a un lado los cilindros para unir las dos piezas.

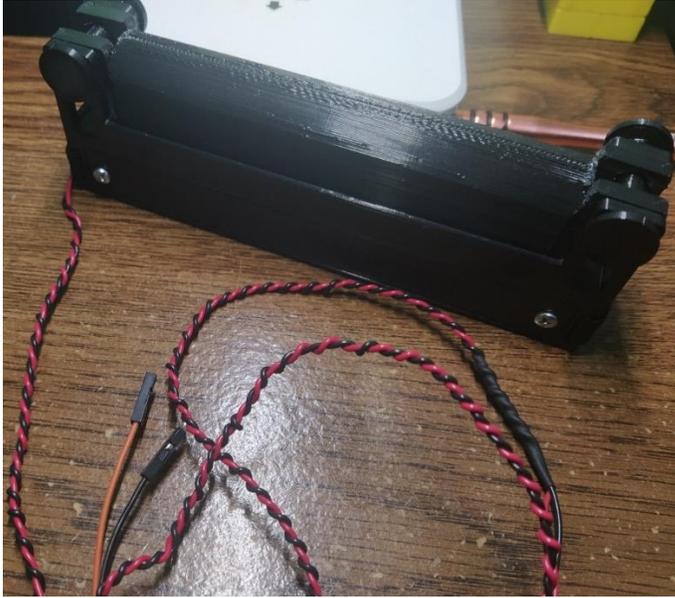


Ilustración 3.30 Pieza gatillo por abajo.



Ilustración 3.31 Base con interruptores.

El ensamblaje del módulo se muestra en la ilustración 3.32 A y un ejemplo del funcionamiento en la mano, se muestra en la ilustración 3.32 B, donde al presionar el gatillo se activan los interruptores. Este modelo también tiene resorte el cual, se sujeta a la mano, para esto se cosió de los dos lados para adherirlo por los respectivos agujeros en las esquinas.



A. Módulo de activación por agarre de mano.



B. Ejemplo de uso.

Ilustración 3.32 (A y B) Módulo por activación de agarre de mano.

Al final del cable se soldaron dos jompers hembra para conectarlo con los pines del SAAC que está implementado en una Raspberry Pi 4B.

3.2 Módulos basados en soplido

Los módulos basados en soplido, se caracterizan por tener un sensor que detecta el flujo del aire y envía una señal a partir de la interacción que este recibe. En este caso se logró desarrollar dos prototipos funcionales con diferentes métodos de interacción, uno mejor que otro y utilizando el mismo programa. Para la realización de los módulos se tuvo que hacer una carcasa con sus diferentes adaptaciones para puertos y aditamentos, la cual fue diseñada en 3D por computadora, después fue impresa, se necesitó un microcontrolador con su debida programación para detectar y regular el tiempo y la sensibilidad del sensor, así como mostrar la información de estos parámetros en un display LCD, de tal manera de que estos puedan variar al mover un potenciómetro. Se utilizó un diseño de un circuito con amplificador para el sensor, en este caso se optó por un micrófono. Este circuito se fabricó en una placa PCB, donde fueron soldados sus componentes. Por último, se le agregaron los puertos, conexiones y complementos que veremos a continuación donde se detalla el desarrollo de estos módulos.

Estos módulos fueron pensados para aquellas personas que ya no tengan movilidad en el cuerpo y puedan respirar adecuadamente, también para aquellas que no tengan control en los movimientos de su cuerpo excepto la respiración, con el soplo sería suficiente para poder comunicarse a través del SAAC.

3.2.1 Diagrama de bloques

En el diagrama de bloques mostrado en la ilustración 3.33, se presentan los componentes más importantes del módulo por soplido.

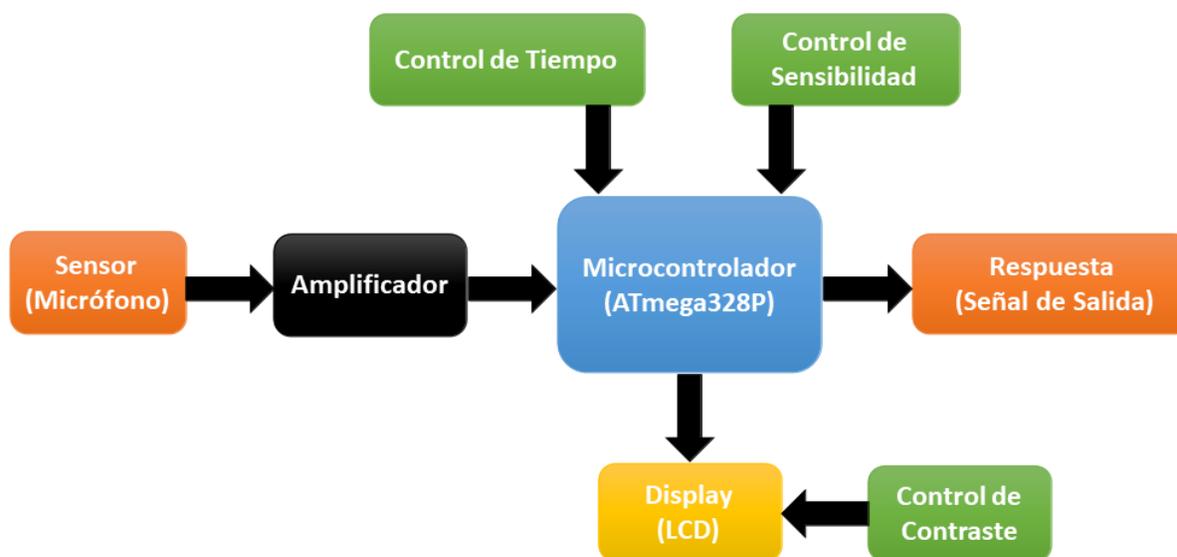


Ilustración 3.33 Diagrama de Bloques módulo de activación por soplido.

Donde:

- ◆ Sensor (Micrófono): Transductor de las señales de audio (en este caso soplo) a eléctricas.
- ◆ Amplificador: Amplifica la señal del sensor para mayor visibilidad.
- ◆ Microcontrolador (Atmga328P): El microcontrolador es el encargado de recibir la señal de entrada, compararla, gestionar la sensibilidad y el tiempo de la señal de salida, enviar la señal de salida y reflejar los valores en el display LCD.
- ◆ Display LCD: Dispositivo que muestra los valores de sensibilidad y tiempo.
- ◆ Control de tiempo: Resistencia variable que controla el tiempo de la señal de salida.
- ◆ Control de sensibilidad: Resistencia variable que controla la sensibilidad para mandar la señal de salida.
- ◆ Control de Contraste: Resistencia variable que controla el contraste del display LCD.
- ◆ Respuesta de Salida: Señal de salida que esta adecuada para el SAAC en voltaje.

3.2.2 Diagrama esquemático

En la ilustración 3.34 se muestra el esquemático que se realizó para este módulo, tanto la variante uno como la variante dos llevan el mismo circuito con tan solo dos pequeños cambios.

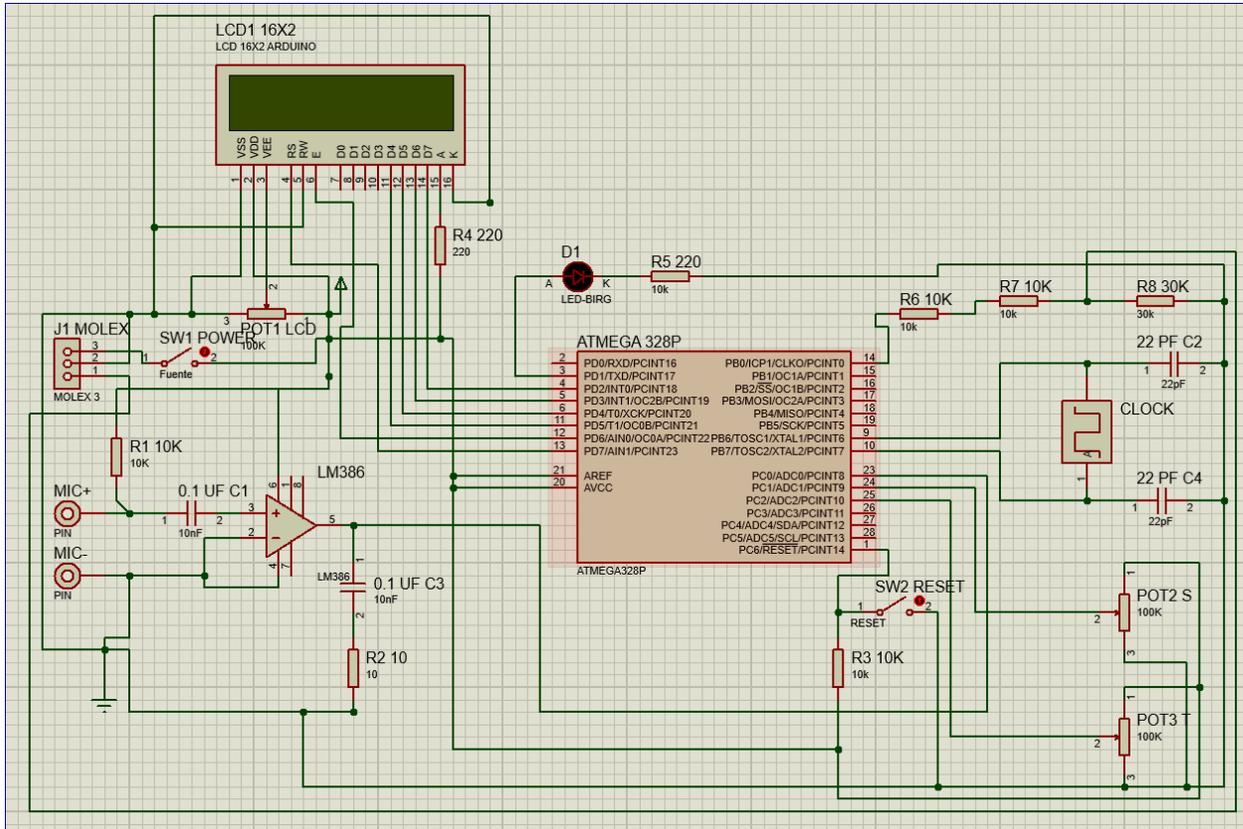


Ilustración 3.34 Esquema del módulo de soplido v2.

Los componentes que se utilizaron para formar el circuito fueron los siguientes:

- ◆ ATmega328P.
- ◆ LM386.
- ◆ LCD16x2.
- ◆ Cristal de 16 MHz.
- ◆ Un micrófono elektret.
- ◆ 3 potenciómetros de 100 KΩ.
- ◆ 1 resistencias de 10 Ω.
- ◆ 1 resistencia de 30 KΩ.
- ◆ 2 capacitores cerámicos de 22 pF.
- ◆ 1 pulsador.
- ◆ 1 switch.
- ◆ 2 capacitores de 0.1 uF.
- ◆ 4 resistencias de 10 KΩ.
- ◆ 2 resistencias de 220 Ω.

- ◆ Molex de 3 pines.
- ◆ 1 LED.

En la ilustración 3.35 se muestra la etapa de amplificación y los puertos del micrófono, para realizar la amplificación se optó por el amplificador LM386, un amplificador muy usado en sistemas de audio, el cual por lo general se utiliza para amplificar la señal de audio a bocinas. En este caso amplificamos la señal que transmite el micrófono, dado a que la señal que produce este sensor está en milivolts y por lo tanto es muy baja. Se utilizó la configuración de amplificación del LM386 para audio y además se metió un filtro con un capacitor con resistencia.

La ganancia del LM386 se dejó en la estándar que fue de 20, dadas las características de alimentación del circuito se utilizaron 5 volts, que es el voltaje mínimo para el funcionamiento del mismo.

Primero pasa por un pequeño filtro que es el capacitor, el cual elimina una parte del ruido, amplifica y la salida de la señal es la pata cinco del LM386, a la cual en paralelo tiene un pequeño filtro pasivo de resistencia con un capacitor de bajo valor. La señal de salida del LM386 (5), se conecta a la patita 23 del ATmega328P, la cual es una entrada analógica.

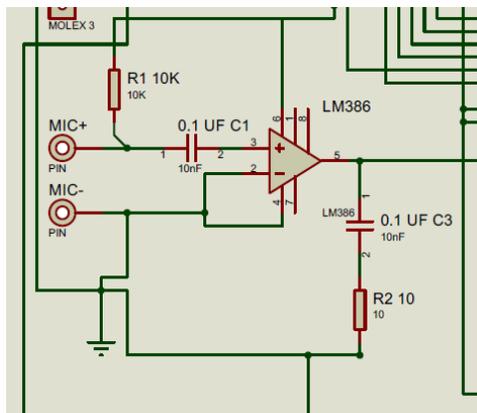


Ilustración 3.35 etapa de amplificación del micrófono.

El ATmega328P es el microcontrolador que realiza todas las operaciones lógicas y aritméticas del circuito, más adelante se explicará su programación. El microcontrolador recibe la señal del amplificador, los potenciómetros que controlan el tiempo y la sensibilidad, un led indicador de salida y el display LCD. El display LCD no utilizo un adaptador I2C, si no que se utilizó la configuración básica que cablea todos sus pines, además de un potenciómetro que controla el contraste de la pantalla.

Como parte del diseño, se utilizó un puerto molex de 3 pines para sacar la señal que va dirigida al SAAC y obtener la alimentación. También se instaló un switch para apagar el circuito, así como un pulsador de reset para el ATmega328P que reinicia el programa instalado.

Se agregó un cristal de 16 MHz para que el ATmega328P funcionara a una mayor velocidad.

Para la señal de salida, dado a que el circuito maneja un voltaje de 5 volts se necesitó un divisor de voltaje que bajara el voltaje a 3.3 volts o menos. Esto nos permite conectar nuestro

módulo con la Raspberry Pi 4B, donde se aloja el programa del SAAC, sin dañarla ya que solo acepta un voltaje de 3.3 volts, si es de 5 volts puede quemar la placa.

Para esto se utilizó un arreglo de resistencias como se muestra en la ilustración 3.36, donde se utilizaron dos resistencias en serie de 10 KΩ para sumar 20 KΩ, las cuales van en serie con una de 30 KΩ y entre la resistencia de 20 KΩ y 30 KΩ se saca la señal de salida para el SAAC.

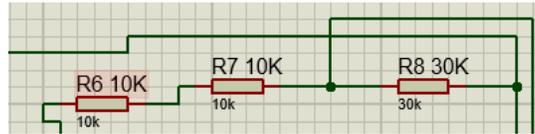


Ilustración 3.36 Divisor de voltaje.

Se utilizó fórmula 3.1, para calcular la señal de salida:

$$V_{out} = \frac{V_{in}R_2}{(R_1 + R_2)} \quad (\text{ec. 3.1})$$

Donde:

- ◆ $V_{out} = 3$ Volts (voltaje de salida).
- ◆ $V_{in} = 5$ Volts (voltaje de entrada).
- ◆ $R_1 = 20$ KΩ (la suma de las dos resistencias de 10 KΩ).
- ◆ $R_2 = 30$ KΩ.

Sustituyendo valores se tiene:

$$\frac{5 * 30000}{(20000 + 30000)} = 3$$

Como último, la diferencia entre la variante 1 y la variante 2 del módulo de activación por soplo, es que la variante uno no tiene el pulsador de reset del programa, y que la señal de salida no está en el pin 14, si no que está en el 3, con un nodo que deriva al LED. Esto se muestra en la ilustración 3.37 que es el esquemático de la variante 1, por lo que, todo lo demás y el funcionamiento es exactamente igual.

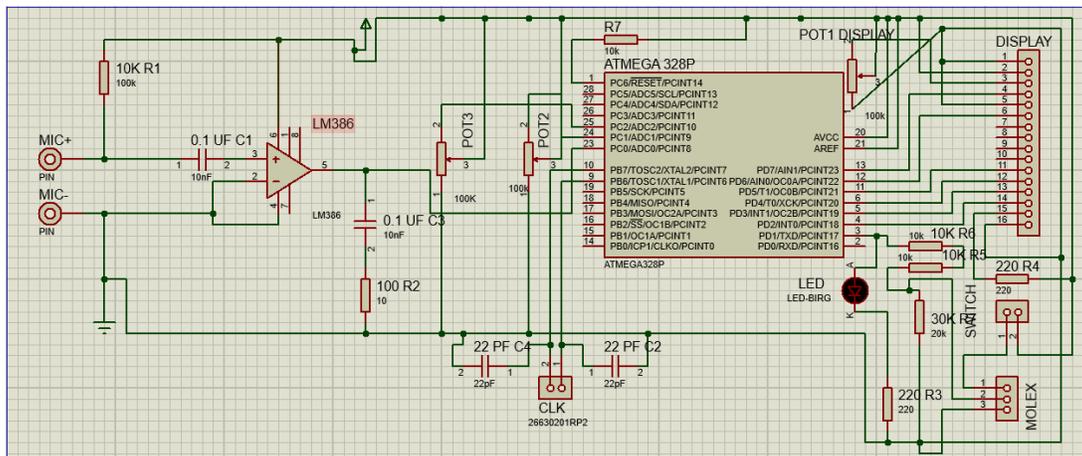


Ilustración 3.37 Esquemático de la variante 1.

3.2.3 Diagrama de flujo

Para controlar todas las señales y enviar datos a un display se necesitó del microcontrolador el cual se usó el ATmega328P, que es el que utiliza la placa Arduino y es posible adaptarlo para poder funcionar de la misma manera. Para el funcionamiento del módulo necesita un programa que administre el tratamiento de las señales y el control de las mismas, como base se utilizó el software IDE de Arduino para programar, este se basa en C/C++ y se puede convertir el código en hexadecimal, para que, al ser grabado en el microcontrolador pueda leer las instrucciones.

A continuación, se muestra en la ilustración 3.38 el diagrama de flujo que describe la programación del módulo de activación por soplo:

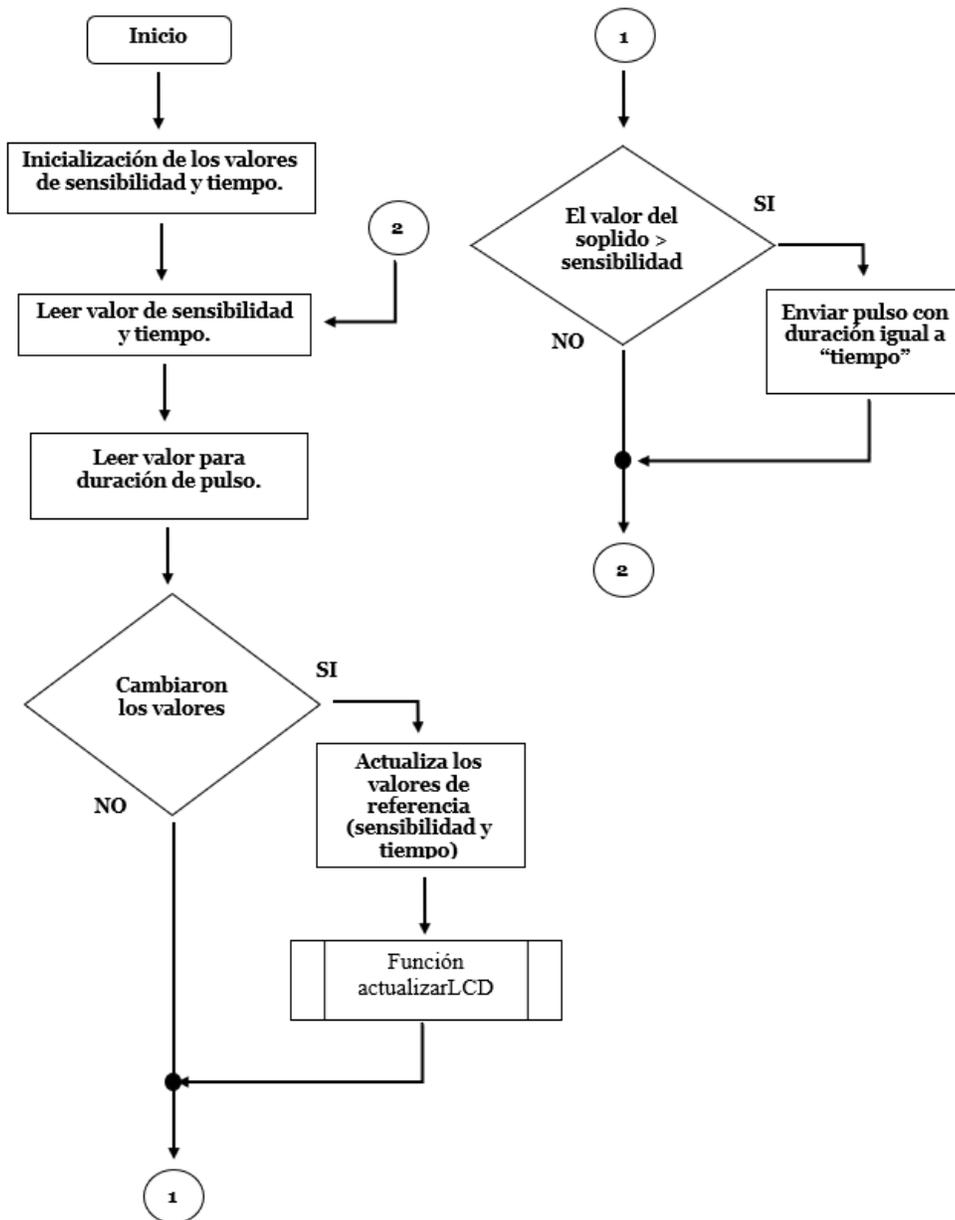


Ilustración 3.38 Diagrama de flujo del módulo de activación por soplo.

Como se muestra en la ilustración 3.38, se mandó a llamar a la función “actualizarLCD()”, por lo tanto, a continuación, se muestra en la ilustración 3.39 su respectivo diagrama de flujo:

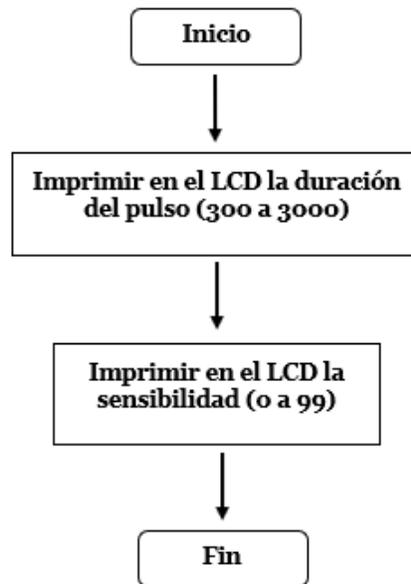


Ilustración 3.39 Diagrama de flujo de la función “actualizarLCD()”

Con los diagramas de flujo, se procedió a la programación para este módulo. Como primer paso se mandaron a llamar a las bibliotecas, dado a que se ocupó una pantalla LCD, para esto se llamó con la función #include y la biblioteca, además de configurar a las patitas del microcontrolador para su conexión.

```
1  #include <LiquidCrystal.h>
2  LiquidCrystal lcd(7,6,5,4,3,2);
```

Con las bibliotecas declaradas se inicializan los valores de tiempo y sensibilidad, para tener el parámetro inicial de referencia.

```
4  int sensibilidad=0;
5  int tiempo=0;
```

Como parte del entorno de Arduino IDE se procede a declarar todas las entradas y salidas, al igual que se inicializa el LCD y se mantienen en alto el puerto 1 y 8.

```
7  void setup()
8  {
9      lcd.begin(16, 2);
10     pinMode(1, OUTPUT);
11     pinMode(8, OUTPUT);
12     digitalWrite(1, HIGH);
13     digitalWrite(8, HIGH);
14 }
```

Con todos los parámetros inicializados, se procede a entrar en el ciclo repetitivo que estará funcionando mientras este prendido el módulo, primero como se muestra en el diagrama de flujo se leen los valores de la sensibilidad y tiempo, estos se toman de dos potenciómetros que están conectados en los puertos analógicos del microcontrolador, estas lecturas fueron grabadas en las variantes s y t, que es la primera letra de sensibilidad y tiempo, más adelante se compararan con las variantes de que se pusieron al inicio del programa, que fueron sensibilidad y tiempo.

```
15
16 void loop()
17 {
18     int s = analogRead(A1); //establecer valor de referencia (0-
1023)
19     int t = analogRead(A2);
```

Después se mapeo a la lectura del tiempo en 300 a 3000, esto quiere decir que, en una variable llamada tLED se guardó el mapeo analógico del potenciómetro que está conectado al puerto A2 del microcontrolador. Al mapear se toma la lectura analógica que se despliega de 0 a 1023 y dependiendo en que valor de resistencia se encuentre el potenciómetro, se tomara de 300 a 3000 como valor, cambiando el intervalo numérico que se le había dado cuando en un inicio.

```
20     int tLED=map(t,0,1023,300,3000);
```

Como siguiente paso se toma una decisión, si han cambiado los valores de los potenciómetros, estos potenciómetros son sensibilidad y tiempo, y están escritos como las variables sensibilidad, tiempo, s, y t. En un principio será verdadera esta condición ya que sensibilidad y tiempo están igualados a cero y s y t son los valores de las lecturas analógicas. Después cambiara dependiendo de la siguiente condición: si el valor de s es mayor o igual a la sensibilidad +10 o si s es menor o igual a sensibilidad -10 y si el valor de t es mayor o igual a tiempo +10 o t es menor o igual a tiempo-10. Si alguna de estas condiciones se cumple es actualiza el valor de referencia igualando sensibilidad a s y tiempo a t, además de que se llama a la función “actualizarLCD()”. Si no se cumple esta condición pasa a la siguiente.

```
21
22     if((s>=sensibilidad+10) || (s<=sensibilidad-10) ||
(t>=tiempo+10) || (t<=tiempo-10))
23     {
24         sensibilidad=s;
25         tiempo=t;
26         actualizarLCD();
27     }
```

Como siguiente paso, se vuelve a poner una condición, la cual es, si el valor del soplido es mayor que la sensibilidad. Esto quiere decir que si, el valor que se recibe del micrófono ya amplificado es mayor que el de la sensibilidad, donde el valor de la sensibilidad es el que nos da el potenciómetro. Para esto se toma lectura de la señal del amplificador entrando por el puerto A0 del microcontrolador, mientras que la lectura de sensibilidad ya fue tomada antes.

Si se cumple la condición y la señal del micrófono es mayor que la de la sensibilidad, se manda un pulso por los puertos 1 y 8 del microcontrolador. Cabe resaltar que se envía una señal con lógica inversa, que en vez de un pulso de 1 o 5 volts se manda uno de 0 o 0 volts, que es como trabaja el SAAC empleado. Ya enviado el pulso se mete un retardo o tiempo con el valor de la variable tLED que esta mapeada para tener un intervalo de 300 a 3000, según la posición del potenciómetro y este valor se puso para indicar de 300 milisegundos a 3 segundos en el tiempo. Ya pasado el tiempo regresan los puertos 1 y 8 del microcontrolador al estado anterior que es de 1 lógico.

Si no se cumple esta condición regresa al inicio del ciclo donde se toman las lecturas de los potenciómetros de sensibilidad y tiempo o s y t.

```
28
29     if (s<=analogRead(A0))
30     {
31         digitalWrite(1, LOW);
32         digitalWrite(8, LOW);
33         delay(tLED);
34         digitalWrite(1,HIGH);
35         digitalWrite(8,HIGH);
36     }
37 }
38
```

Aquí es donde se acaba el programa y vuelve a entrar en el ciclo mientras este encendido el módulo. Para la función “actualizarLCD()” descrita en el diagrama se programó lo siguiente.

Como se van a mostrar los valores de sensibilidad y tiempo en pantalla, se mapea la variable tiempo de 300 a 3000 y se guarda en la variable tiempoLCD, después se mapea la variable sensibilidad de 0 a 99 y se guarda en la variable sensibilidadLCD. Una vez hecho esto se imprime en el primer renglón “Tiempo: ”, dejando ocho espacios para borrar todo lo que tenga el renglón, así dejando limpio el espacio para poner el valor de la variable tiempoLCD y permitir que el valor máximo de la variable se muestre, sin que guarde la antigua información en los espacios que no se ocuparon, además de poner el valor de la variable en el último recuadro del display.

Se agregó un retardo de 10 milisegundos y se procede a imprimir la sensibilidad en el segundo renglón, primero se imprime "Sensibilidad: ", abarcando los 16 espacios del renglón, para borrar toda aquella información anterior, luego se imprime la variable sensibilidadLCD en el espacio 14 del segundo renglón, el cual es suficiente para poner el valor de 0 a 99. Se deja otro retardo de 10 milisegundos y se termina la función actualizarLCD, regresando al ciclo principal.

```
39 void actualizarLCD()
40 {
41     int tiempoLCD = map(tiempo,0,1023,300,3000);
42
43     int sensibilidadLCD = map(sensibilidad,0,1023,0,99);
44
45     lcd.setCursor(0,1);
46     lcd.print("Tiempo:           ");
47     lcd.setCursor(9,1);
48     lcd.print(tiempoLCD);
```

```

50     delay(10);
51
52     lcd.setCursor(0,0);
53     lcd.print("Sensibilidad:  ");
54     lcd.setCursor(14,0);
55     lcd.print(sensibilidadLCD);
57     delay(10);
59 }

```

Se compiló el código después de verificar los errores, creando un archivo hexadecimal y se grabó en la memoria del microcontrolador, en este caso se utilizó el ATmega328P dadas sus características, se configuraron los fusibles internos para poder funcionar con un cristal de 16 MHz y tener los tiempos adecuados. Para hacer esta configuración se le agregaron componentes descritos en la sección 3.2.2 diagrama esquemático.

3.2.4 PCB de variante 1

Una vez hechos los esquemáticos con Proteus, y habiendo realizado las respectivas pruebas del circuito, se procedió a realizar los diseños para la PCB, con el mismo programa.

En la ilustración 3.40 se muestra el diseño del PCB, este fue generado por el esquemático y conectando las pistas por el método automático del programa, además de las definiciones por defecto de dispositivos, se hicieron algunas ediciones en las etiquetas. Se agregó un margen y cuatro agujeros grandes en las esquinas para tornillos de sujeción, la distancia de cuadrículado es de 2.5 mm, se tuvo que realizar un reacomodo de elementos previa al enrutado, ya que al pasar del esquemático al diseño del PCB no venían acomodados por defecto, por lo que se acomodaron dependiendo el acomodo de la carcasa.

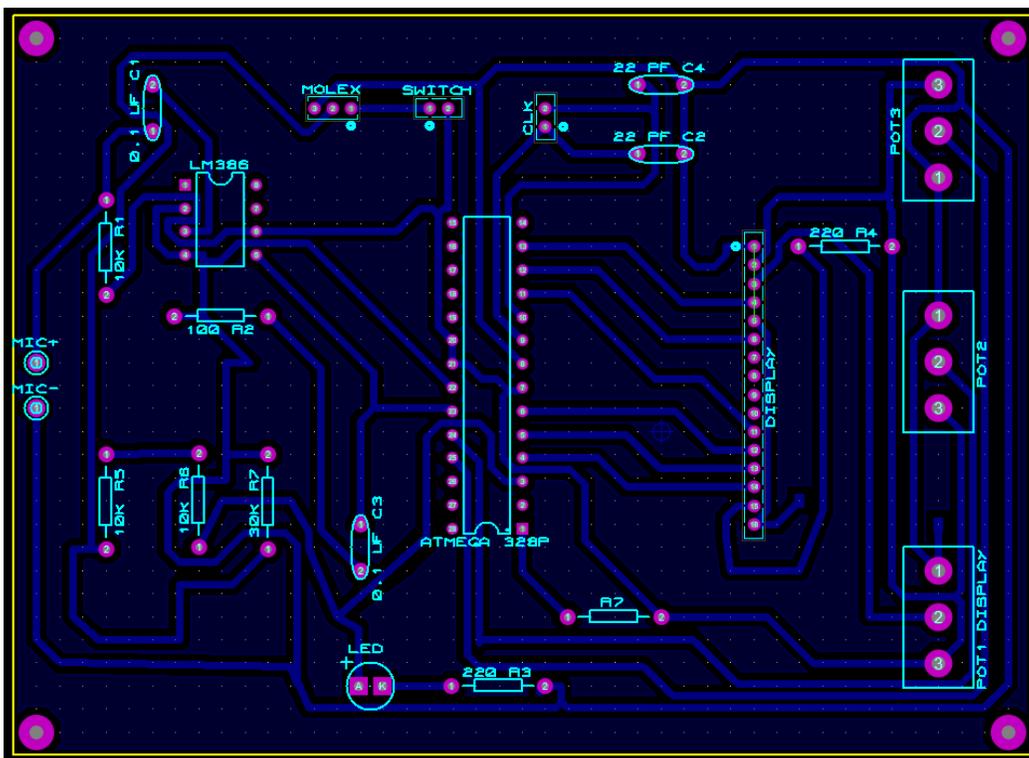


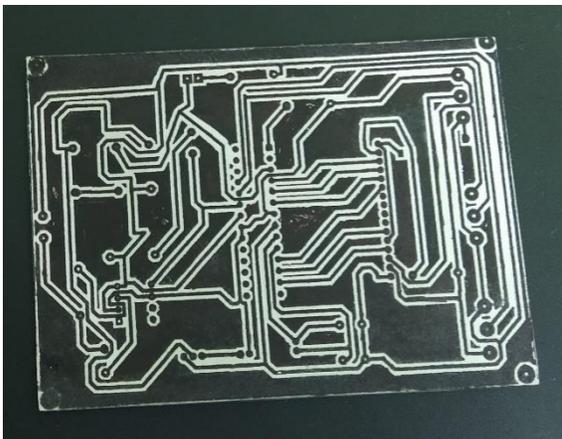
Ilustración 3.40 Diseño PCB variante 1.

Ya con este diseño de la PCB, se exportó en formato PDF, el cual fue definido para dejar solamente las pistas y pads, quedando listo para ser transferido a la placa de cobre. Como parte del traspaso se eligió el método del planchado, el cual consiste en la transferencia del tóner a través del calor de una plancha.

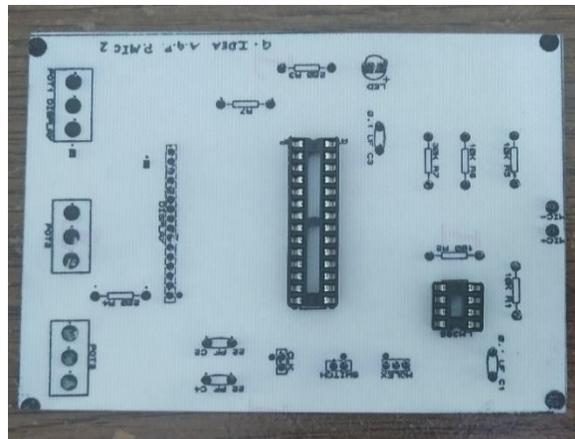
La copia fue impresa en tóner con un negro profundo para asegurar que la cantidad de tóner transferido no deje huecos, después se tomó una placa de cobre recortada a un tamaño de 80mm x 110 mm, la cual se lijó con lana de acero y después se limpió con alcohol para evitar grasa e imperfecciones en la placa de cobre. Después se recortó la impresión y se puso la cara impresa con la del cobre fijándola con cinta adhesiva, una vez fijado se plancho la placa remarcando el papel y transfiriendo el tóner por varios minutos, ya pasados se procedió a meter en agua donde se quitaría el papel y se revelaría el impreso ya en la placa de cobre como se muestra en la ilustración 3.41 A.

Ya impreso en la placa de cobre, se procedió a meter en ácido (Cloruro Férrico) el cual deshizo el cobre excepto el que estaba en las pistas, después se limpió con agua y se quitó el tóner con la lana de acero dejando el cobre descubierto y obteniendo el PCB. Después con un taladro de mano se procedió a hacer los respectivos agujeros con una broca milimétrica. Para esta PCB se utilizó el planchado de la simbología de los componentes en la parte sin cobre de la placa, se hizo de igual manera, imprimiendo en tóner la simbología y pasándola con una plancha a calor.

Ya con la PCB lista se soldaron los componentes que conforman el circuito como se muestra en la ilustración 3.41 B, donde faltan las conexiones a componentes que van en la carcasa, el LM386 y el microcontrolador.



A. PCB con el impreso.



B. PCB parte de arriba.

Ilustración 3.41 (A yB)Placa PCB variante 1.

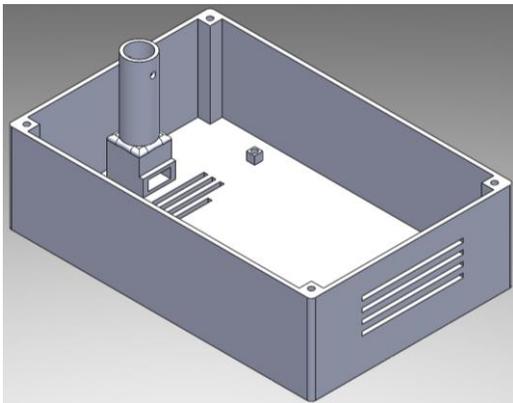
Con el microcontrolador programado se introdujo en el puerto de la PCB, cabe recalcar que los puertos de la PCB están diseñados para que se puedan cambiar, así como los componentes que van conectados por cable ya que están unidos a la carcasa.

3.2.5 Modelo 3D y real de variante 1

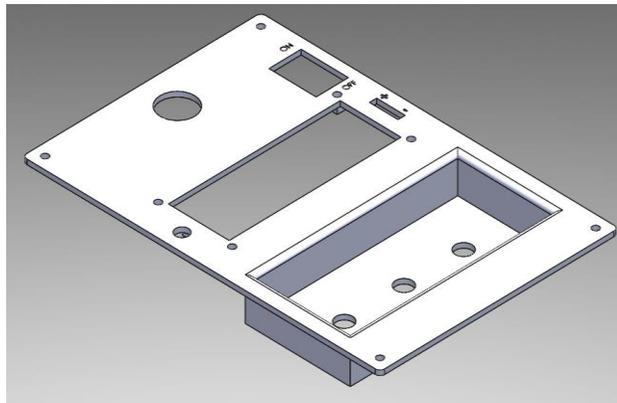
Como parte del diseño se modeló una carcasa 3D en SolidWorks específica para este módulo y variante, este consiste en dos piezas una tapa y una caja.

La caja fue hecha a partir de un rectángulo de 100mm x 150mm en donde se extruyó a una altura de 44mm, las paredes son de un espesor de 2 mm que es suficiente para mantener su estructura, en las esquinas se hicieron columnas más gruesas porque ahí mismo en la parte de arriba se hicieron los agujeros para los tornillos, que fueron de 2.8mm, tiene una rejilla de ventilación de un lado y de otro en la parte de abajo, esto es para sacar el aire y la saliva acumulada. En la base del interior de la caja, se hizo una estructura en forma de tubo que es donde pasa la manguera que desemboca al micrófono de la PCB, en la base, se tiene un conducto, con una rejilla que conecta a la manguera, este es rectangular al inicio y está en forma de L, en la parte que sube es en forma de cilindro, este conducto va directamente al micrófono, esta parte se muestra en la ilustración 3.42 A, que es la caja de la variante 1, además de un hoyo en la parte superior del cilindro que es donde se fija la manguera y por ultimo cuatro soportes con agujero de tornillo en la base para acoplar la PCB, están ubicados según las medidas del PCB.

La pieza de la tapa, que se muestra en la ilustración 3.42 B, de igual manera que la pieza de la caja esta se basa en un rectángulo con las mismas medidas, solo que la altura es de 2mm, tiene cuatro agujeros en cada esquina para los tornillos y estas esquinas están redondeadas para mayor comodidad, cuenta con un agujero en la parte de arriba para la manguera, un agujero para el molex conector de energía y señal, otro para el switch de energía que a su vez tiene marcados “ON” y “OFF”, el agujero para el display con sus respectivos tornillos y en la parte inferior su tope para que queda plano a la tapa en la parte superior, en la parte de abajo un rectángulo que esta vaciado con tres agujeros en su interior para los potenciómetros, este está redondeado para mayor comodidad y un agujero para el LED indicador.



A. Pieza Caja.

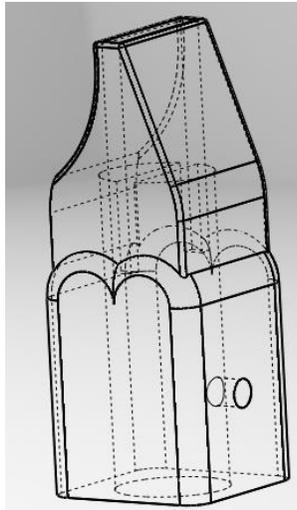


B: Pieza Tapa.

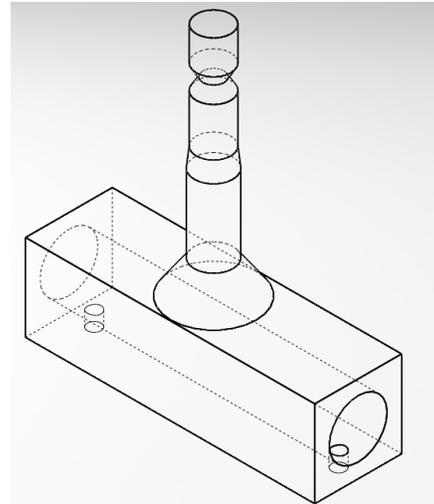
Ilustración 3.42 (A y B) Módulo de soplo 3D variante 1

Ya teniendo las piezas se ensamblaron en el mismo programa, donde se ve que coinciden las medidas de los tornillos y el tubo o cilindro donde entra la manguera.

Como parte opcional se realizó una boquilla en 3D para sujetar la manguera como se muestra en la ilustración 3.43A, también se le hizo una adaptación para colocar la manguera en un brazo para micrófono con una pieza como se muestra en la ilustración 3.43 B.



A. Boquilla.



B. Adaptador a brazo.

Ilustración 3.43 (A y B) Boquilla y adaptador a brazo.

Ya terminadas las piezas modeladas, se exportaron en formato “stl” y luego se pasaron a Ultimaker Cura, donde se mandaron a imprimir pieza por pieza, después con las piezas listas se ensambla la PCB en la caja donde se tienen la base de los tornillos para ajustarla.

En la tapa es un caso distinto ya que en esta, fueron incrustados los componentes con los que interactuaran los usuarios, son los tres potenciómetros en la parte de abajo con sus respectivas perillas, también el LED indicador, así como el LCD que fue unido con los cuatro tornillos, el switch que entró a presión y por último las entradas para la alimentación. Todos estos componentes tienen cables que van soldados tanto a la PCB como a ellos, excepto por el display que se conecta por pines, tal y como se muestra en la ilustración 3.44.



Ilustración 3.44 Tapa unida a PCB.

Con la tapa y sus componentes adheridos se ensambla la tapa con la caja, y se sujetó por medio de los tornillos que están en las esquinas. Por último, se añadió la manguera que llega hasta abajo, se fijó con un tornillo. En la ilustración 3.45 se muestra el modelo final del módulo de activación por soplido variante 1.

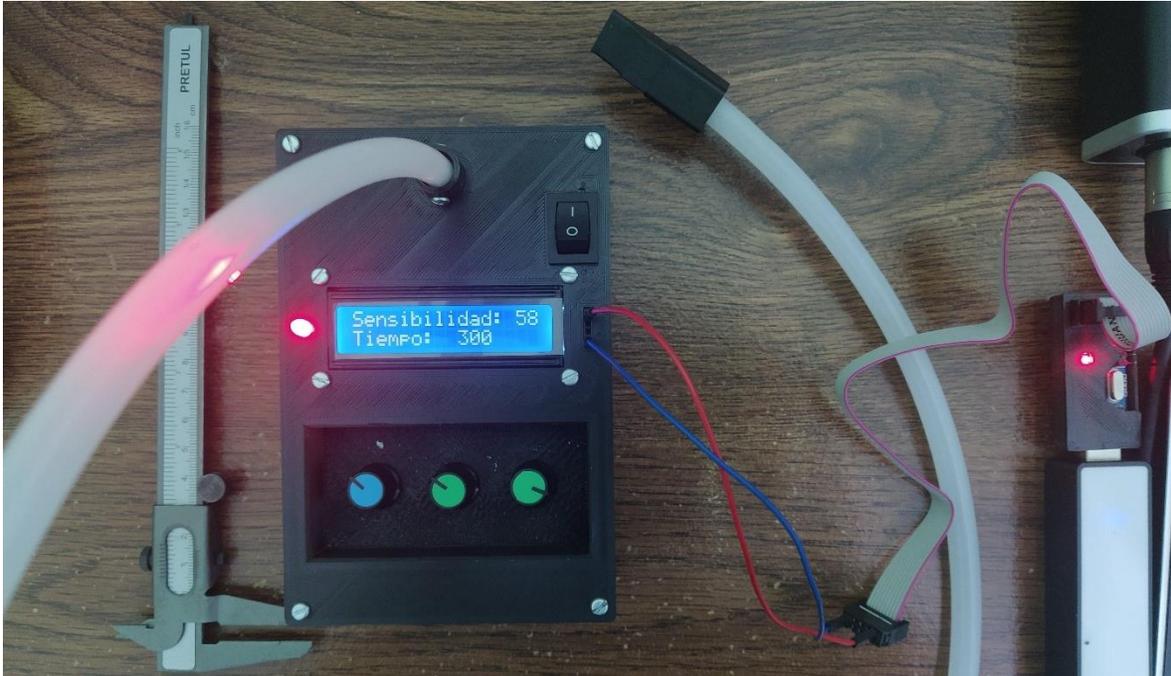


Ilustración 3.45 módulo de soplido variante 1.

Por último, se muestra en la ilustración 3.46 la manguera con la boquilla en el brazo para que no esté suelta y se quede firme apuntando a la boca.



Ilustración 3.46 Boquilla con soporte.

3.2.6 PCB de variante 2

Para la variante 2 al igual que la variante 1, con base al esquemático visto en el capítulo 3.2.2 realizado en Proteus, se procedió a hacer el diseño del PCB, en este caso se cambió algunas características de los componentes, para que sus pines se ajustaran al tamaño de los

componentes, además de ajustar algunos componentes para que solo se colocaran conectores, dado a que no irán soldados a la placa si no que se conectaron por medio de cables. Se hizo un margen para delimitar el espacio conforme a la placa y se acomodaron los componentes, después se auto ruteó con la opción del programa, generando las pistas automáticamente, estas con un grosor previamente especificado, se comprobó que todo estuviera en orden y se realizó un pequeño puente. Para sujetar la PCB a la caja se pusieron orificios en las esquinas, como parte del diseño se rellenaron las partes faltantes para mejor estética como se muestra en la ilustración 3.47, cabe resaltar que para la cuadrícula se tiene un espacio de 2.5 mm.

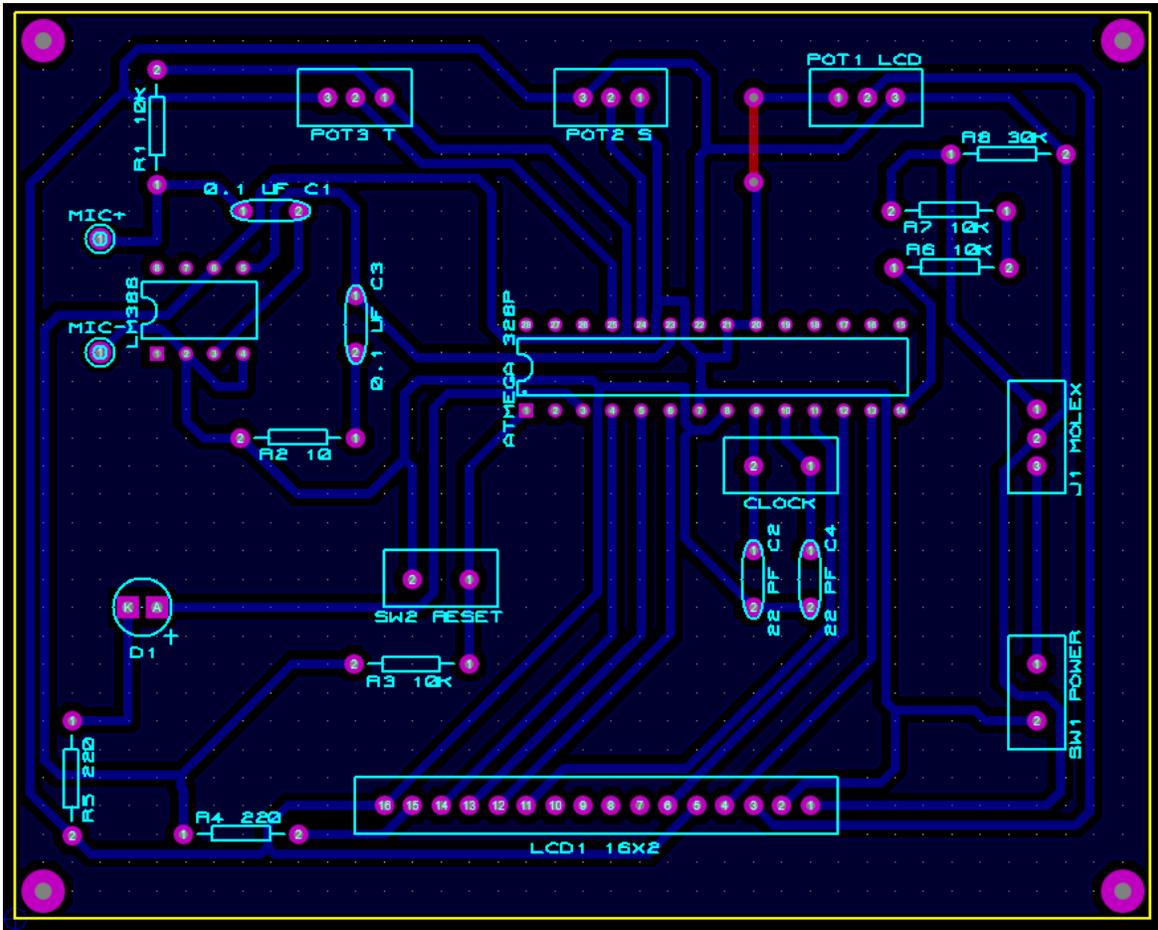


Ilustración 3.47 Diseño PCB variante2.

Una vez teniendo el diseño del PCB se procedió a exportar en PDF, el cual de igual manera se imprimió una copia con tóner que nos serviría para el impreso. Esta vez no se utilizó la técnica del planchado, pero de igual forma se transfirió el tóner.

Para esto se cortó la baquelita en 100 mm x 80 mm, después se lijo la parte del cobre con lana de acero y se limpió con alcohol, ya limpia la baquelita se recortó el circuito impreso en tóner y se pegó en la parte del cobre, de tal manera que quedara cara a cara con el cobre y el tóner. Ya fijo, se usó acetona para la transferencia de tóner, se roció con acetona hasta quedar empapado por encima y se aplastó por un tiempo. Habiendo echo este proceso, se metió en agua tibia donde poco a poco se fue quitando el papel dejando el cobre marcado con el tóner,

para evitar errores de transferencia se remarcó las partes imperfectas con un plumón permanente.

Una vez con la baquelita marcada se metió en Cloruro Férrico donde se deshizo el cobre expuesto y después de un tiempo se sacó y se lavó con agua, todo con sus debidas precauciones dado los materiales usados. Con una lana de acero se lijo para quitar el tóner y se lavó con alcohol.

Ya con las pistas echas se procedió a taladrar cada agujero según los componentes, también se realizó el agujero para los tornillos en las esquinas, para esto se utilizó un taladro de mano con una broca milimétrica.

Una vez hecha la PCB con sus respectivos agujeros, se marcó la simbología de los componentes del lado blanco con un plumón permanente, después se agregaron los componentes y se soldaron, cabe resaltar que no se soldó el microcontrolador si no una base y muchos componentes solo son los puertos molex para poder conectarlos ya que están unidos a la tapa. En la ilustración 3.48 se muestran las dos caras de la PCB, una con los componentes soldados y otra vista por arriba.

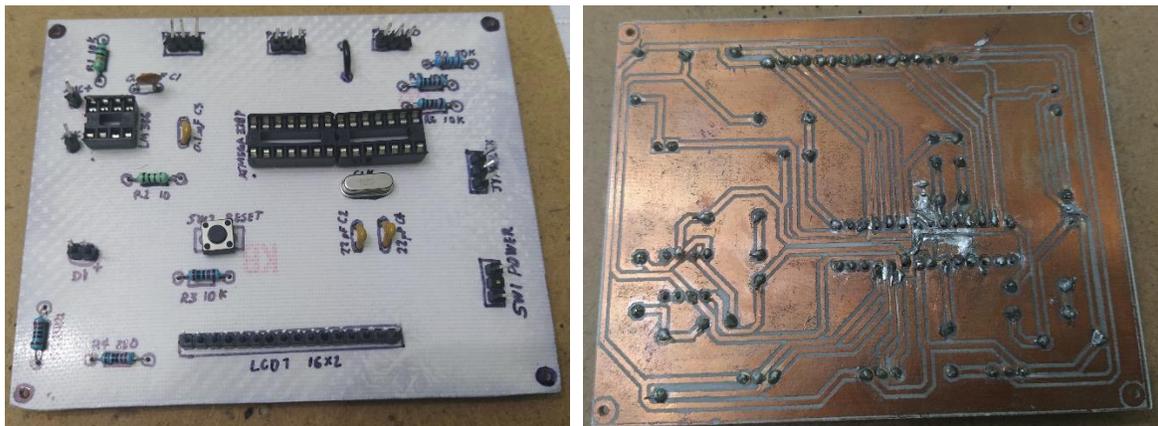


Ilustración 3.48 PCB de los dos lados, módulo de soplo variante 2.

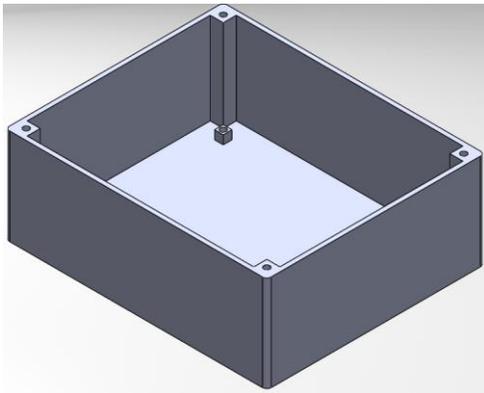
3.2.7 Modelo 3D y real de variante 2

Para este modelo hubo un cambio importante en la hora de interactuar con el sensor, de tal forma que se realizaron cambios en la carcasa. Debido a que se abandonó el uso de la manguera sustituyéndola con una diadema con micrófono. De igual manera se utilizó el programa SolidWorks para modelar en 3D las piezas.

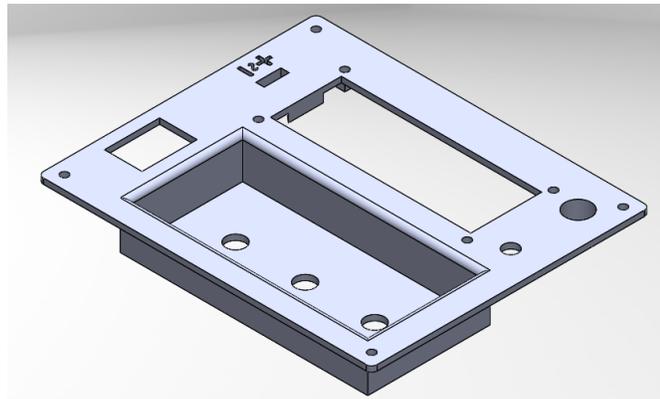
Para este modelo se hicieron dos piezas, una es la caja y otra es la tapa, para la caja se tomó un rectángulo de 100 mm x 120 mm el cual se extrudió y se vació, redondeando sus esquinas, dejando columnas en las esquinas con agujeros de 2.8mm para los tornillos, adicionalmente se puso la base para la PCB que consiste en cuatro cuadros con agujeros para tornillos como se muestra en la ilustración 3.49 A.

Para la tapa se realizó un rectángulo igual que la pieza de la caja solo que en altura fue de 2mm, de igual forma se le hicieron los cuatro hoyos en las esquinas para los tornillos, un agujero para el led indicador, otro para el switch de encendido y apagado, otro para el display con sus cuatro tornillos y el tope en la parte de abajo. También se hizo una cavidad para los

potenciómetros, sacando un rectángulo hacia abajo y en el suelo tres agujeros para los potenciómetros. Para la conexión con el SAAC y la fuente de alimentación se hizo un agujero y en la parte izquierda la simbología para las conexiones, por último, la pieza cambiante un agujero para la conexión de la diadema con el micrófono como se muestra en la ilustración 3.49 B.



A. Caja.

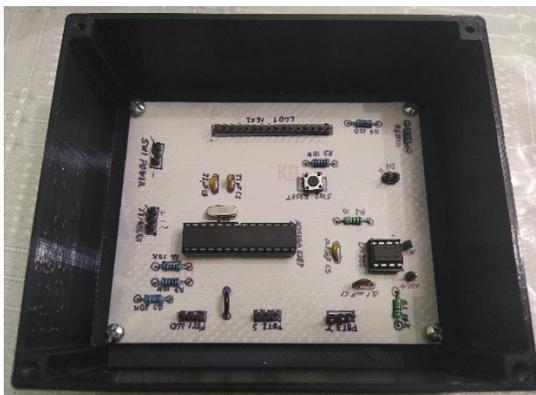


B. Tapa.

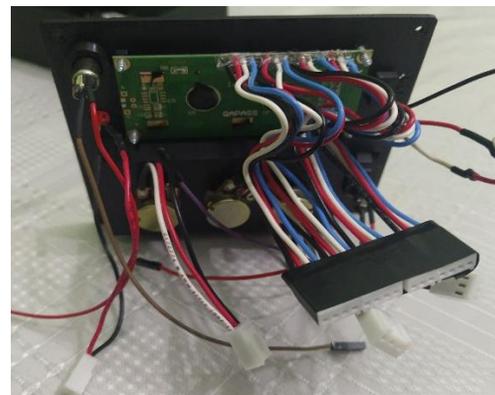
Ilustración 3.49 (A y B) Modelo 3D de la variante 2 soplo.

Una vez teniendo las piezas modeladas, se procede a exportar en formato “stl”, después se mapea la forma de imprimir en Ultimaker Cura y después se imprime pieza por pieza. Ya impresas se procede a ensamblar la PCB a la caja con los tornillos para ajustarla.

Ya con la tapa se adhirió los componentes, a todos estos siguiendo el diseño se les soldaron cables excepto el display, dejando en el extremo del cable puertos molex para solo conectar con la placa PCB, haciendo separable la tapa y haciendo fácil de remplazar los componentes. Para el LCD, se le hizo un arreglo con cables que es un bus de conexión. El Jack hembra para la diadema que es de 3.5mm fue probado antes para saber qué tipo de conexiones deben ir soldadas. Para mejor estética los potenciómetros tienen perillas en las puntas y el molex fue pegado para mayor firmeza a la hora de conectar, al igual que el led, como se ve en la ilustración 3.50 donde en la A se muestra la caja con el PCB ya fijado y del lado B las conexiones con la tapa.



A. Ensamblaje del PCB.



B Tapa con sus componentes.

Ilustración 3.50 (A y B) Ensamblaje del módulo por soplido variante 2.

Con los elementos adheridos a la tapa y la caja se procedió a ensamblar las dos piezas, primero uniendo los componentes de la tapa a la PCB con sus conexiones, después pegando la tapa con los tornillos que están en las esquinas. Para su uso es necesario conectar la diadema en el Jack de 3.5mm. en la ilustración 3.51. Se puede ver el modelo de activación por soplo variante 2, con su respectiva diadema y a un lado sus cables de conexión.



Ilustración 3.51 Módulo de soplo variante 2.

3.3 Módulo con acelerómetro

El módulo con acelerómetro, se caracteriza por el uso de un acelerómetro para medir los movimientos cortos, como el de la cabeza, en este caso para enviar una señal al SAAC. Para este módulo se necesitó un microcontrolador, en este caso se escogió el ATmega328P, para poder analizar la información que le llegaba del acelerómetro y enviar una señal dependiendo de algunas condiciones. De igual forma se necesitó una PCB, una carcasa y los acoplamientos al paciente para su funcionamiento, mostrando en un display el tiempo y la sensibilidad con la que mandamos la señal, todo esto controlado a través de unos potenciómetros y por medio de un programa grabado en el microcontrolador. De igual manera se implementó un LED indicativo para la conexión con el acelerómetro y otro para señalar la señal de salida. Este es un módulo más complejo, que es posible adaptar a diferentes partes del cuerpo y solo requiere que el paciente tenga algún movimiento controlado en una parte de su cuerpo.

Como principal sensor se utilizó el MPU6050 que es un acelerómetro con giroscopio digital que envía una señal por medio de una comunicación I2C al microcontrolador y es de bajo costo, dándole sentido al objetivo de desarrollo de estos módulos que es ser de bajo costo, además de que cubre diferentes discapacidades siendo el que se adapta mejor al cuerpo.

3.3.1 Diagrama de bloques

Para la descripción de los bloques que constituyen al módulo con acelerómetro, se presenta el diagrama de la ilustración 3.52.

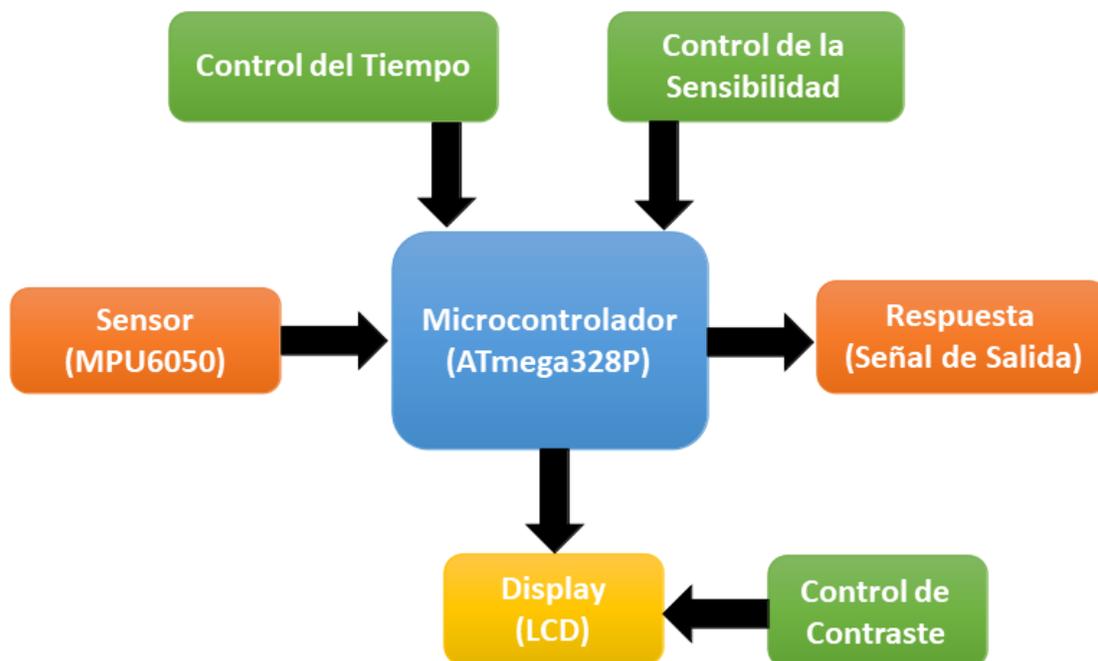


Ilustración 3.52 Diagrama de bloques módulo acelerómetro.

Donde:

- ◆ Sensor (MPU6050): Es el sensor digital que contiene el acelerómetro y un giroscopio, el cual capta el movimiento.
- ◆ Microcontrolador (Atmega328P): El microcontrolador es el encargado de recibir la señal de entrada, compararla, gestionar la sensibilidad y el tiempo de la señal de salida, enviar la señal de salida, con reflejo en un led y reflejar los valores en el display LCD.
- ◆ Display LCD: Dispositivo que muestra los valores de sensibilidad y tiempo.
- ◆ Control de tiempo: Resistencia variable que controla el tiempo de la señal de salida.
- ◆ Control de sensibilidad: Resistencia variable que controla la sensibilidad para mandar la señal de salida.
- ◆ Control de contraste: Resistencia variable que controla el contraste del display LCD.
- ◆ Respuesta de salida: Señal de salida que esta adecuada en voltaje para el SAAC.

3.3.2 Diagrama esquemático

Para la realización de este módulo se necesitó un circuito, para el cual, se realizó un esquemático en el programa Proteus que más adelante nos permitirá exportar el diseño en PCB, en la ilustración 3.53 se muestra el esquemático del módulo con acelerómetro.

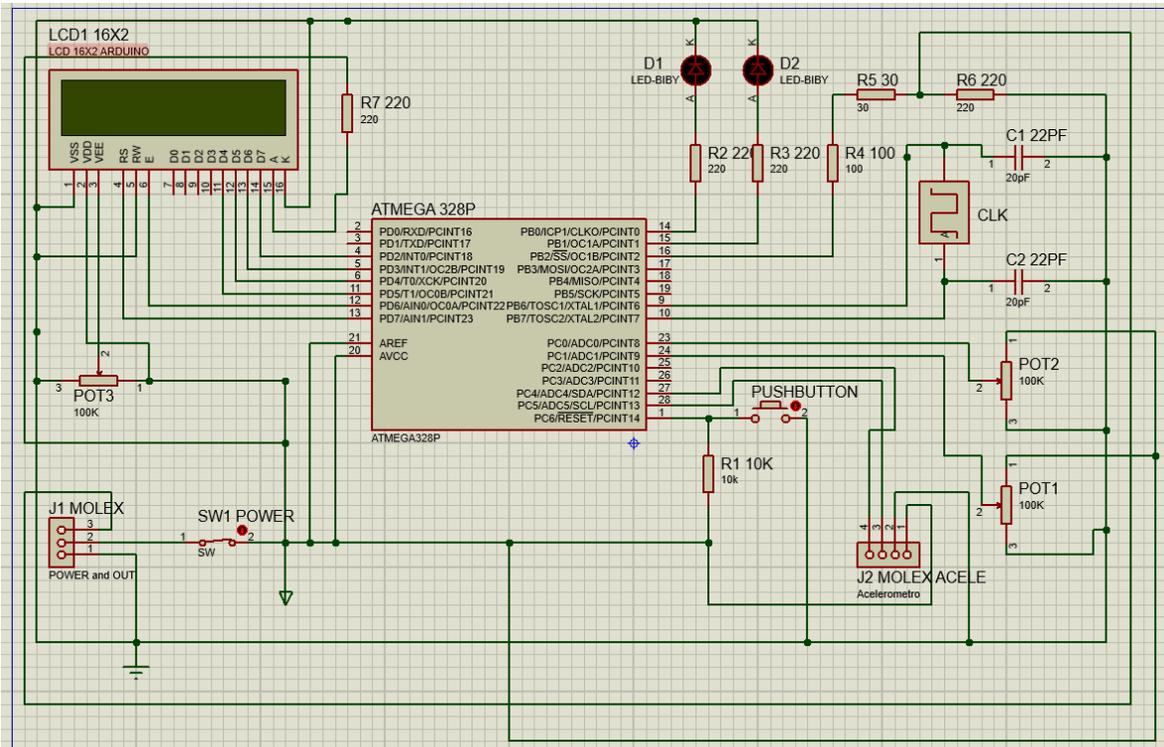


Ilustración 3.53 Esquemático del módulo con acelerómetro.

Los componentes que se utilizaron para formar el circuito fueron los siguientes:

- ◆ ATmega328P.
- ◆ LCD16x2.
- ◆ Cristal de 16 MHz.
- ◆ 3 potenciómetros de 100 K Ω .
- ◆ 1 resistencias de 30 Ω .
- ◆ 1 resistencia de 100 Ω .
- ◆ 2 capacitores cerámicos de 22 pF.
- ◆ 1 pushbutton.
- ◆ 1 switch.
- ◆ 1 resistencias de 10 K Ω .
- ◆ 4 resistencias de 220 Ω .
- ◆ 1 Molex de 3 pines hembra.
- ◆ 1 Molex de 4 pines hembra.
- ◆ 2 LED.

Para la señal de salida, dado a que el circuito maneja un voltaje de 5 volts se necesitó un divisor de voltaje que redujera el voltaje a 3.3 volts o menos, ya que el SAAC no tiene la

capacidad de soportar un voltaje de 5 volts, en el caso que lo conectemos sin ninguna reducción de voltaje quemaría la placa. Por lo tanto, nos permite conectar nuestro módulo con la Raspberry Pi 4B donde se aloja el programa del SAAC sin dañarla.

Para esto se utilizó un arreglo de resistencias como se muestra en la ilustración 3.54, donde se utilizaron dos resistencias en serie de 100 Ω y de 30 Ω para sumar 130 Ω, las cuales van en serie con una de 220 Ω y entre la resistencia de 30 Ω y 220 Ω se saca la señal de salida para el SAAC.

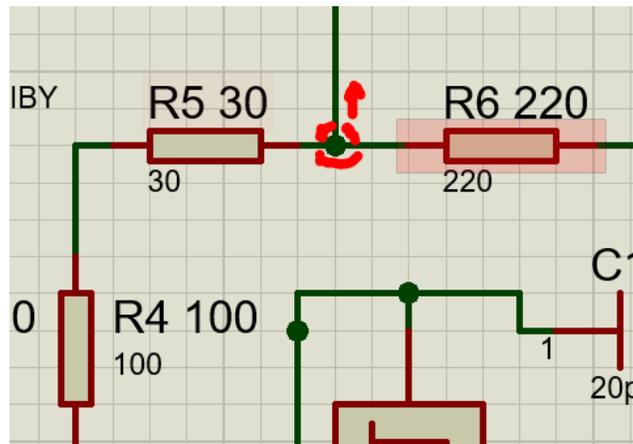


Ilustración 3.54 Divisor de voltaje acelerómetro.

Se utilizó la formula (ec. 3.1), para calcular la señal de salida.

Donde:

- ◆ $V_{out} = 3$ Volts (Voltaje de Salida).
- ◆ $V_{in} = 5$ Volts (Voltaje de entrada).
- ◆ $R_1 = 130 \Omega$ (La suma de las dos resistencias de 100 Ω y de 30 Ω).
- ◆ $R_2 = 220 \Omega$

Sustituyendo en la ecuación se tiene:

$$\frac{5 * 220}{(130 + 220)} = 3$$

Para la conexión con el sensor MPU6050 se utilizó un molex de 4 puertos, los cuales dos son para la alimentación y los otros son los puertos de comunicación, que son SDA y SCL, están conectados a los puertos analógicos del ATmega328P, los demás puertos del MPU6050 no se utilizaron.

3.3.3 Diagrama de flujo

Para controlar todas las señales y enviar datos al display, se utilizó el ATmega328P como microcontrolador y administrador del sistema. Se utilizó la interface de Arduino IDE para realizar el programa. Una vez hecho el programa se exporta o se compila en formato hexadecimal para funcionar en el microcontrolador.

A continuación, se muestra en la ilustración 3.55 el diagrama de flujo que describe la programación del módulo con acelerómetro:

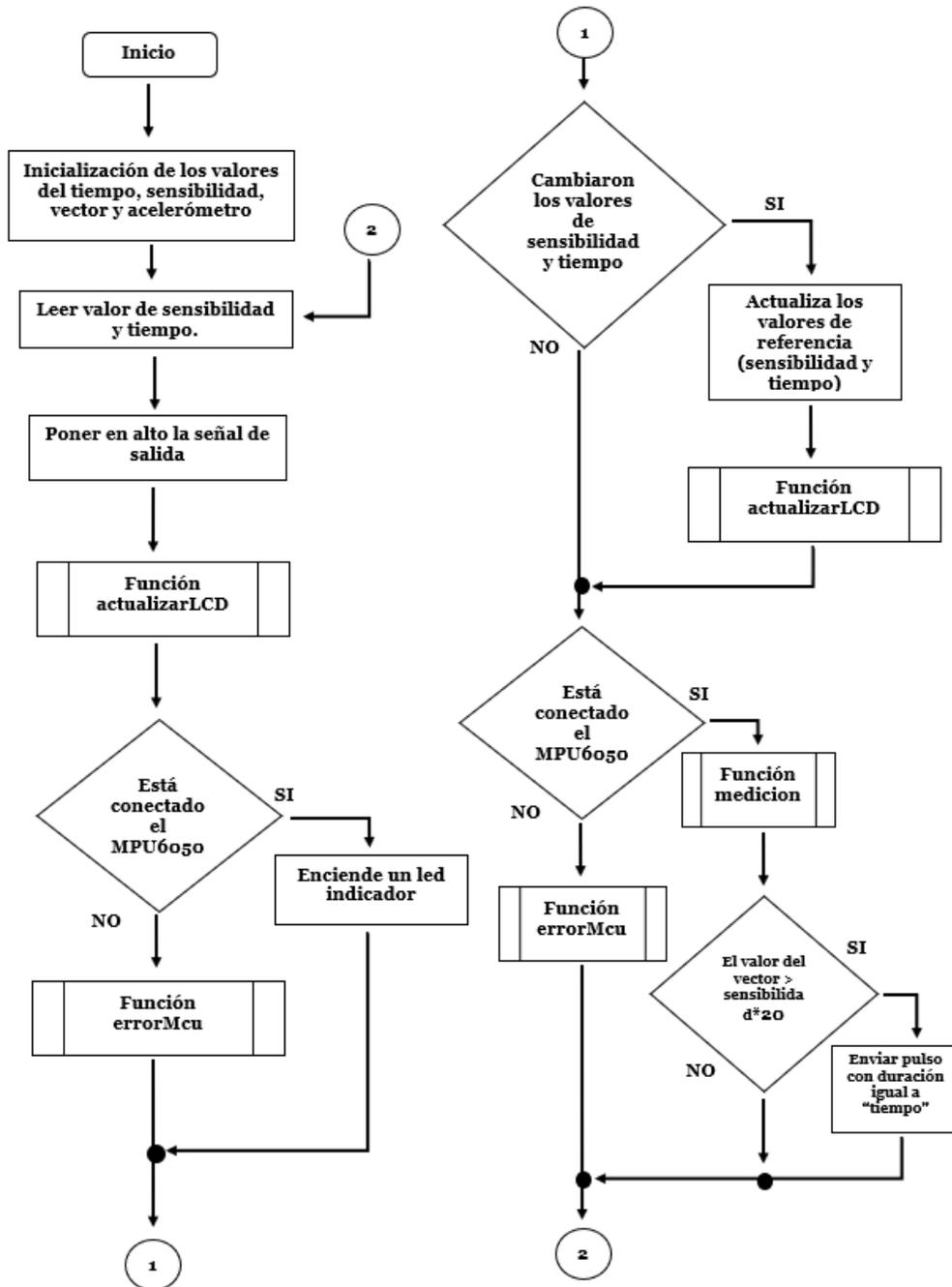


Ilustración 3.55 Diagrama de flujo módulo con acelerómetro.

Como se muestra la ilustración 3.55, se llama a diversas funciones, las cuales realizan una acción aparte del sistema principal. A continuación, se muestran en la ilustración 3.56 la función “medición()”, que describe la obtención de un vector a partir del acelerómetro:

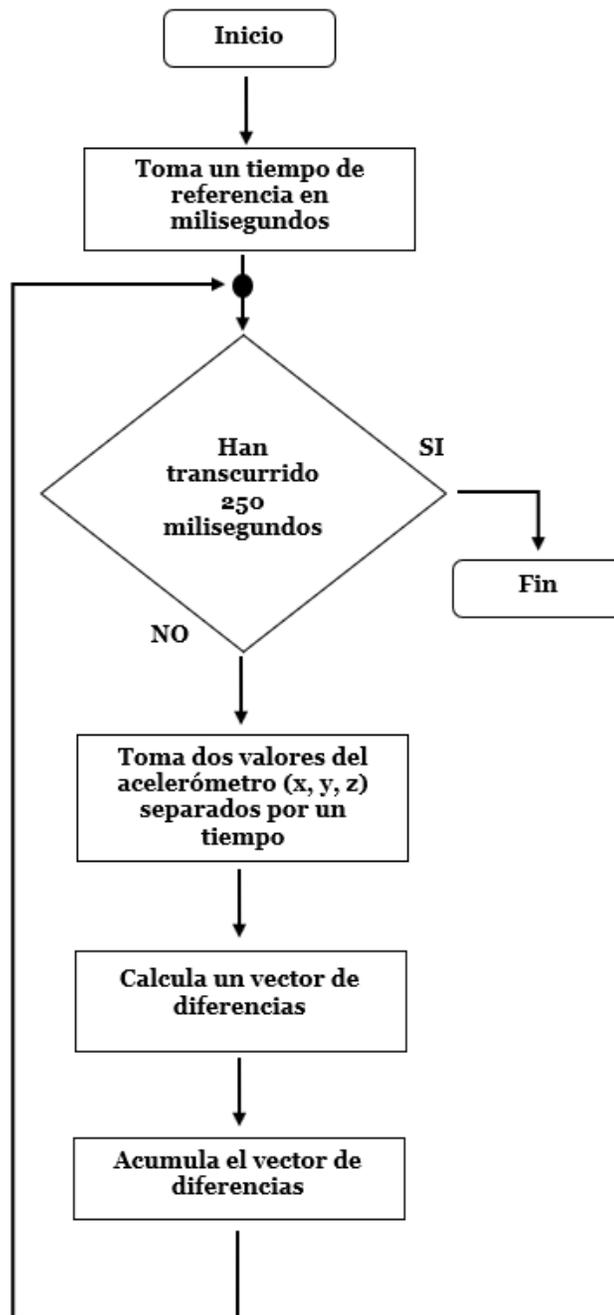


Ilustración 3.56 Diagrama de flujo de la función “medicion”.

Otra función llamada “*errorMcu()*”, que nos sirve para identificar si el acelerómetro está conectado al módulo. A continuación, se muestra en la ilustración 3.57 su diagrama de flujo:

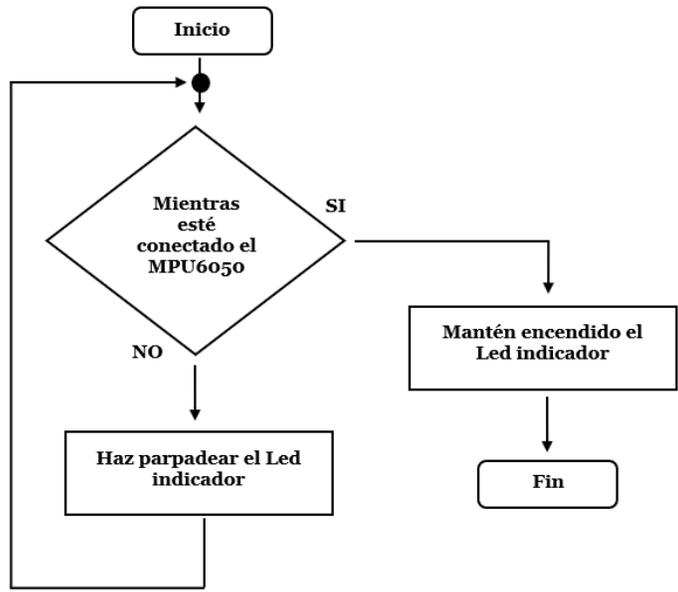


Ilustración 3.57 Diagrama de flujo de la función “errorMCU”.

Por último, se muestra en la ilustración 3.58 la función “actualizarLCD()”, que indica la actualización de los datos mostrados en el display. A continuación, se muestra su diagrama de flujo:

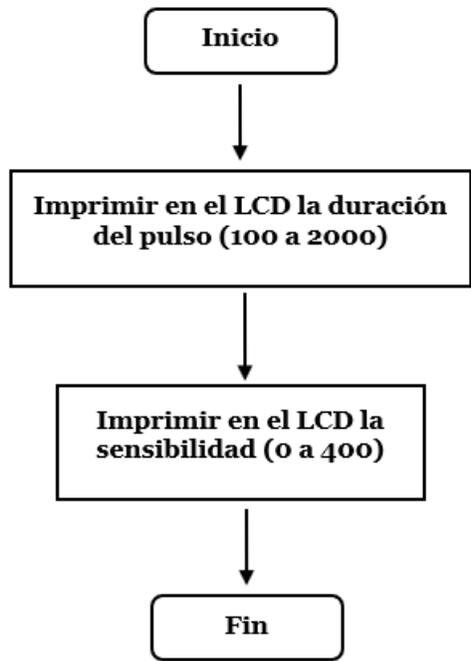


Ilustración 3.58 Diagrama de flujo de la función “actualizarLCD”.

Con los diagramas de flujo, se procedió a realizar la programación del módulo con acelerómetro. Para esto se ocupó el entorno Arduino IDE para después exportarlo al microcontrolador.

Como primer paso se declararon las bibliotecas ya que para utilizar el MPU6050 (acelerómetro/giroscopio) se necesitaban bibliotecas específicas como “wire”, MPU6050 y I2Cdev, que permiten la comunicación con el mismo, además de la biblioteca correspondiente para la pantalla LCD.

```
1 #include "I2Cdev.h"
2 #include "MPU6050.h"
3 #include "Wire.h"
4 #include "LiquidCrystal.h"
5
```

Ya con las bibliotecas declaradas, se inicializaron las variables correspondientes que se iban a ocupar como muestra el diagrama de flujo, se declararon las entradas de la pantalla LCD, las variables para el acelerómetro, un tiempo de referencia, la variable del vector, las variables de sensibilidad y tiempo, así como sus variables que más adelante se comparan, que son s y t.

```
6 MPU6050 mpu(0x68);
7 LiquidCrystal lcd(7,6,5,4,3,2);
8
9 int ax, ay, az;
10 unsigned long tempRef1;
11 float valor=0;
12 int sensibilidad;
13 int tiempo;
14 int s;
15 int t;
16
```

Como parte de la estructura de programación se indicó dónde van a estar conectados los pines de salida de la pantalla y los pines de la señal de salida. Se inicializó biblioteca “wire” y se tomaron las lecturas de s y t, que son los potenciómetros conectados a las entradas analógicas, la lectura fue igualada a sensibilidad y tiempo con un mapeo o cambio de valores, que fue de 0 a 400 en sensibilidad y de 100 a 2000 en tiempo. También se puso en alto o en 1 la señal de salida, porque el SAAC está programado en la entrada como lógica inversa y para indicar que al encender el módulo no mande una señal errónea. Se mandó a llamar a la función “actualizarLCD()” para refrescar el display al inicio.

```
17
18 void setup()
19 {
20   lcd.begin(16, 2);
21   lcd.home();
22   pinMode(9,1); //salida status mcu
23   pinMode(8,1); //replica respuesta led
24   pinMode(10,1); //salida respuesta Señal
25   digitalWrite(10,1);
26   Wire.begin();
27   s=analogRead(A0); //Lectura del Potenciómetro para la
sensibilidad
28   t=analogRead(A1); //Lectura del Potenciómetro para el Tiempo
29   sensibilidad=map(s,0,1023,0,400);
30   tiempo=map(t,0,1023,100,2000);
31   actualizarLCD();
32
```

Como se muestra en el diagrama de flujo, antes de que inicie el ciclo, se tiene una decisión, de que, si está conectado el sensor MP6050 o no. Si está conectado se manda a encender el led indicador de conexión del sensor, que sale por el pin 9, si no está conectado, se manda a llamar a la función “errorMcu()”, que se explicara más adelante.

```
33  mpu.initialize();
34  if (mpu.testConnection())
35  {
36      digitalWrite(9,1);
37  }
38  else
39      errorMcu();
40  }
```

Con la iniciación del programa, se procedió a entrar en el loop o ciclo, que mientras este encendido el módulo, repetirá el mismo ciclo de programación una vez inicializado. El ciclo inicia tomando las lecturas de los potenciómetros de sensibilidad y tiempo, guardando los valores en las variables s1 y t1, estas serán utilizadas para la condicional como se tiene en el diagrama de flujo. La condición es, que si s1 es mayor que s+10 o menor a s-10 y que si t1 es mayor a t+10 o t-10, dicho de otro modo, que si el valor de sensibilidad y tiempo cambio por mover el potenciómetro 10 unidades tanto para arriba o abajo, se cambie los valores que se tomaron como referencia para comparar y se actualicen por los nuevos s1 y t1. También se llama a la función “actualizarLCD()”, ya que si cambia el valor de sensibilidad y tiempo se tienen que ver reflejados en la pantalla LCD. Si no, pasamos al siguiente.

```
41
42  void loop()
43  {
44      int s1=analogRead(A0); //Lectura del Potenciotmetro para la
sensibilidad
45      int t1=analogRead(A1); //Lectura del Potenciotmetro para el
Tiempo
46
47      if((s1>=s+10) || (s1<=s-10) || (t1>=t+10) || (t1<=t-10))
48      {
49          sensibilidad=map(s1,0,1023,0,400);
50          tiempo=map(t1,0,1023,100,2000);
51          s=s1;
52          t=t1;
53          actualizarLCD();
54      }
```

Como parte final del ciclo de trabajo del módulo con acelerómetro, se hace otra comparación de que el MP6050 (sensor) esté conectado, si no está conectado se manda a llamar a la función “errorMCU()”. Si está conectado el módulo, se llama a la función “medicion()”, además adentro se hace otro condicional, este es el que compara, si el valor del vector es mayor a la sensibilidad*20, se manda el pulso de salida en lógica inversa, se multiplica por 20 la sensibilidad ya que el valor del vector que se saca del sensor es muy alto comparado con la sensibilidad, se envía la señal o pulso de salida, por el led indicador y la salida del molex al SAAC, además de que el pulso se manda por un determinado tiempo, que es, la variable tiempo, la cual es controlada a través del potenciómetro. Aquí termina el ciclo, pero este se va a repetir mientras siga energizado el circuito.

```

56  if (mpu.testConnection())
57  {
58      medicion();
59      if(valor>(sensibilidad*20))
60      {
61          digitalWrite(8,1);
62          digitalWrite(10,0);
63          delay(tiempo);
64          digitalWrite(8,0);
65          digitalWrite(10,1);
66          delay(tiempo/2);
67      }
68
69  }
70  else
71      errorMcu();
72  }
73

```

Se crearon diferentes funciones para este programa, como parte de la simplificación del código y segmentar las funciones. La función “actualizarLCD()” se creó para refrescar la pantalla del LCD cada vez que se detectara un cambio en los potenciómetros de sensibilidad y tiempo, como se muestra en su diagrama de flujo y contemplando que las lecturas de los potenciómetros son variables globales, primero se puso el cursor en el renglón 2 y columna uno, después se imprimió en pantalla “Tiempo: ”, dejando los 9 espacios para borrar todo aquello que este escrito antes, después se puso el cursor en la fila 2 columna 9 y se imprimió la variable tiempo, que ya anteriormente en el programa principal se había mapeado o dado un rango de valores de 100 a 2000, esto se hace con la idea que el programa lee 1000 como un segundo. Después, se deja un pequeño tiempo de 10 milisegundos y se procede a hacer lo segundo del diagrama de flujo, que es, imprimir el valor de sensibilidad, como el anterior se indica la fila y la columna, que en este caso fue, fila 1 columna 1, donde se imprimió “Sensibilidad: ”, dejando los tres espacios vacíos pero marcados para eliminar lo que estuviera escrito con anterioridad, después se colocó el cursor en la fila 1 columna 13, y se imprimió el valor de la sensibilidad, que está representado por la lectura del potenciómetro, mapeada de 0 a 400, con el objetivo de mostrar contra que se va comparar la señal de vector. Por último, para esta función se mete un pequeño retraso de 10 milisegundos y se regresa al programa original.

```

110 void actualizarLCD()
111 {
112     lcd.setCursor(0,1);
113     lcd.print("Tiempo:      ");
114     lcd.setCursor(9,1);
115     lcd.print(tiempo);
116
117
118     delay(10);
119
120     lcd.setCursor(0,0);
121     lcd.print("Sensibilidad:  ");
122     lcd.setCursor(13,0);
123     lcd.print(sensibilidad);
124     delay(10);
125
126 }

```

Después se hizo la función “errorMcu()”, la cual indica o hace el testeo de que el sensor MPU6050 esté conectado, para esto empieza con un condicional como el diagrama de flujo, que mientras el sensor o acelerómetro no esté conectado, parpadeará el led indicador de conexión del sensor y no saldrá de este bucle hasta que esté conectado. Si está conectado, mandara a encender el led indicador de conexión del sensor y lo mantendrá encendido hasta que se desconecte el sensor, para saber que está conectado se toma una función especial llamada “!mpu.testConnection()” que permite identificar que el sensor esté conectado. Con esto se regresa al programa principal, que de vez en cuando, llamara a la función “errorMcu()”.

```

98 void errorMcu()
99 {
100  while(!mpu.testConnection())
101  {
102    digitalWrite(9,1);
103    delay(500);
104    digitalWrite(9,0);
105    delay(500);
106  }
107  digitalWrite(9,1);
108 }

```

La función “medicion()”, una de las más importantes del programa ya que esta deduce un vector a partir de los datos obtenidos del sensor. Primero inicializa la variable valor = 0, después como dice el diagrama de flujo de la función “medicion()”, toma un tiempo de referencia en milisegundos, este valor se toma más adelante para saber cuánto tiempo a transcurrido. Después se emplea un condicional que, mientras los milisegundos que han pasado, menos el tiempo de referencia en milisegundos, sea menor que, 250, entonces se toma los valores de los ejes x, y, z de la aceleración y se guardan en una variable cada uno, se deja pasar un retardo de 20 milisegundos y se vuelven a tomar los valores de los ejes del acelerómetro, guardándolas en otras variables, se deja pasar un retardo de 20 milisegundos. Después, se procede a convertir los valores en un vector, para esto se utiliza una variable flotante donde se utiliza una formula ec 3.2.

$$Vector = \sqrt{(ax1 - ax2)^2 + (ay1 - ay2)^2 + (az1 - az2)^2} \quad (ec. 3.2)$$

Donde:

- ◆ ax1, ay1, az1: son los valores de los ejes tomados en la primera lectura del sensor o acelerómetro.
- ◆ ax2, ay2, az3: son los segundos valores tomados del sensor o acelerómetro.

Con esta fórmula, al hacer las lecturas y sustituirlas en la formula se saca un vector, que es igualado a la variable valor. En otras palabras, mientras no pasen 250 milisegundos, genera un vector con dos lecturas de los ejes he iguálalos a la variable valor. Si se cumple la condición se pasa al programa principal donde se compara el valor del vector con la sensibilidad.

```

74 void medicion()
75 {
76  valor=0;
77  tempRef1=millis();

```

```

78  while((millis()-tempRef1)<250)
79  {
80  mpu.getAcceleration(&ax, &ay, &az);
81  int ax1=ax;
82  int ay1=ay;
83  int az1=az;
84  delay(20);
85
86  mpu.getAcceleration(&ax, &ay, &az);
87  int ax2=ax;
88  int ay2=ay;
89  int az2=az;
90  delay(20);
91
92  float mov =sqrt((pow((ax1-ax2),2))+pow((ay1-
ay2),2))+pow((az1-az2),2));
93
94  valor=valor+mov;
95  }
96 }
97

```

Con esto se cumplen las tres funciones llamadas por el programa principal, en resumen, de todo el programa, es tomar los valores del acelerómetro y generar un vector que se compara con una sensibilidad, para mandar una señal al SAAC, controlando el tiempo y la sensibilidad con dos potenciómetros y mostrando en una pantalla LCD los valores, además de tener un sistema que verifica que el acelerómetro o sensor esté conectado.

Después, se compilo todo el programa, ya una vez verificados los errores, creando un archivo hexadecimal y a través de un programa se grabó en el microcontrolador el programa realizado, en este caso se utilizó el ATmega328P dadas sus características. Se le cambiaron los fusibles internos para poder funcionar con un cristal de 16 MHz y tener los mismos tiempos que una placa Arduino. Para hacer esta configuración se le agregaron componentes descritos en el esquemático, que se muestran en el capítulo 3.3.2.

3.3.4 PCB

Una vez hechos los esquemáticos y habiendo probado el circuito con su programación, se procedió a realizar la PCB. Tomando el esquemático del capítulo 3.3.2 hecho en Proteus, se generó el diseño PCB, modificando los componentes para que fueran de las mismas dimensiones que los reales, se hizo un margen del tamaño de la placa y se acomodaron los componentes conforme iba a estar en la tapa de la carcasa. Una vez acomodados se procedió a configurar el tamaño de las pistas de conexión, después se auto ruteó, generando las pistas conforme la configuración y mejor ruta automáticamente. Se checaron que las conexiones estuvieran bien y se procedió a hacer las modificaciones necesarias, como lo es un puente marcado en color rojo que se aprecia en la ilustración 3.59, el llenado de espacios entre pistas para ahorro de materiales, las marcas para los tornillos en las esquinas y el margen. Ya con el diseño del PCB terminado como se muestra en la ilustración 3.59, se procedió a exportar en pdf, donde se imprimiría en tóner en una hoja de papel.

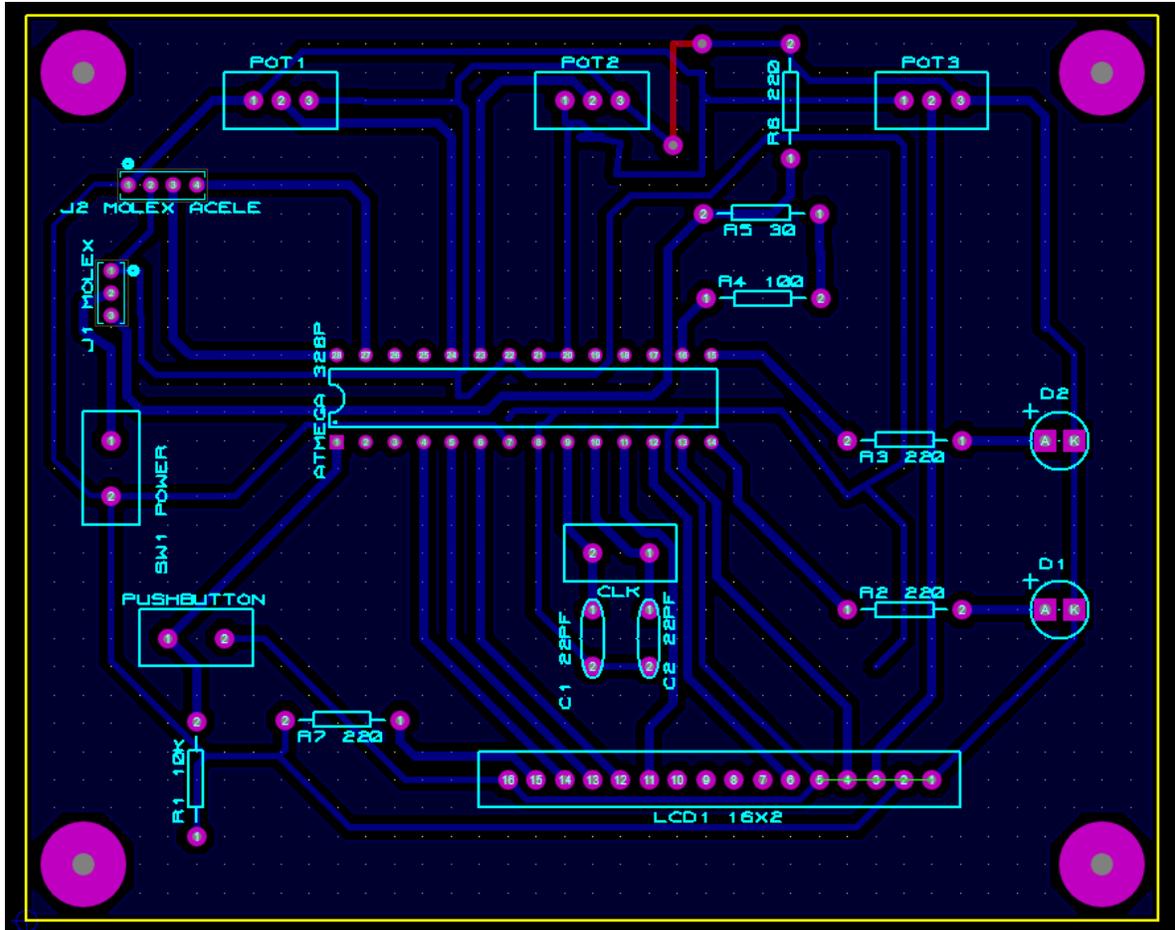


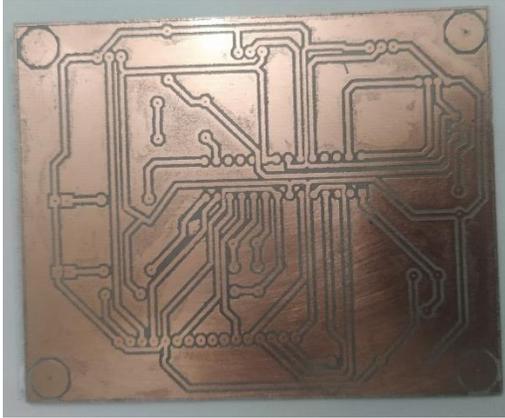
Ilustración 3.59 Diseño PCB del módulo con acelerómetro.

Se tomo una placa fenólica, para poderla usar se tuvo que limpiar bien, primero lijándola con lana de acero para quitar imperfecciones, después se limpió con alcohol para quitar manchas de grasa de los dedos. Ya limpia y con la impresión en tóner, se dispuso a recortar la impresión, después se pegó a la placa fenólica del lado del cobre y se fijó con cinta adhesiva. Para este método se tomó el traspaso de tóner con acetona, el cual consistió en empapar de acetona la impresión fija en la placa fenólica y después aplastándola de tal forma de que poco a poco se transfiriera el tóner del papel a la parte de cobre, después de aplastarlo y remojarlo con acetona por un tiempo, se metió en agua tibia para quitar el papel y dejar solamente el tóner pegado al cobre. Ya una vez impreso en el cobre se procedió a verificar y arreglar las pistas que no se transfirieron bien con un plumón permanente.

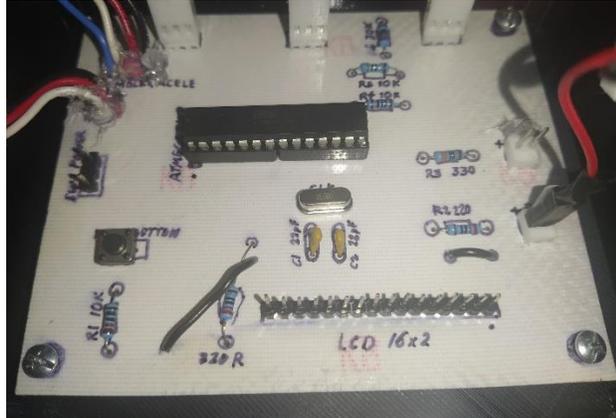
Con la placa impresa con tóner se procedió a deshacer el cobre con Cloruro Férrico, esto se hizo sumergiendo la placa en el ácido y moviendo lentamente por un tiempo hasta que se deshizo el cobre, una vez deshecho el cobre, se mete en agua para limpiar el ácido, todo esto se hizo con sus respectivas precauciones y cuidados.

Para remover el tóner de la placa se lijo con lana de acero y se limpió con alcohol, como se muestra en la ilustración 3.60 A, ya con la placa hecha se procedió a taladrar cada agujero con un mini taladro y brocas milimétricas. Ya con todos los agujeros incluso con los de los tornillos, se procedió a hacer la simbología en la parte del reverso para identificar donde van

los componentes y su dirección de ensamblaje. Se ensamblaron los componentes y la base para el microcontrolador y se procedió a soldar cada uno. Teniendo en cuenta que muchos de los componentes van unidos en la tapa y solo se sueldan los puertos de conexión, como se muestra en la ilustración 3.60 B.



A. Placa PCB



B. Placa por arriba.

Ilustración 3.60 (A y B) Placa PCB del módulo con acelerómetro.

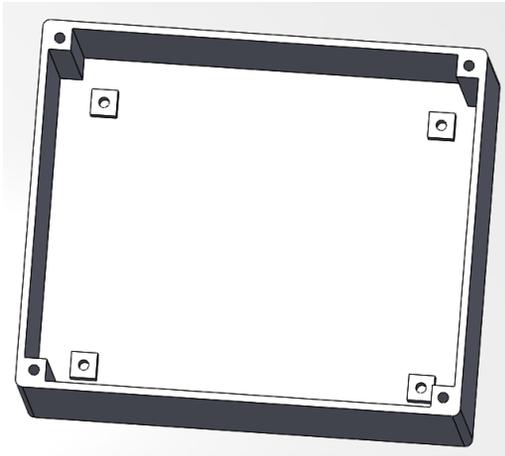
Antes de soldar los componentes, se revisaron pista por pista, para verificar que no existiera ningún corto en sus pistas.

3.3.5 Modelo 3D y real

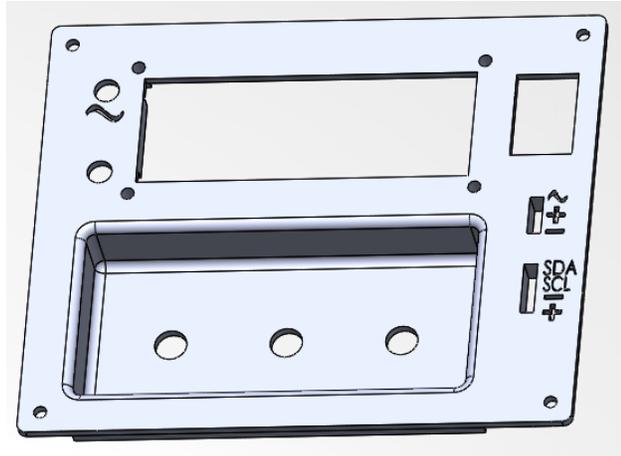
Como parte del desarrollo de este módulo, se diseñó su respectiva carcasa en 3D, donde se ocupó el programa SolidWorks para el modelado, el cual conlleva a hacer dos piezas para la carcasa y tres variantes para sostener el sensor MPU6050 en diferentes partes del cuerpo.

Primero se realizó la base, la cual fue la caja, esta fue hecha a partir de un rectángulo de 100mm x 120mm, donde las esquinas fueron redondeadas y se extruyó con una altura de 57mm, donde se le dejaron columnas gruesas en las esquinas con los agujeros para los tornillos en la parte superior con un diámetro de 2.8mm, esta se vació y en el fondo se construyeron cuatro estructuras con orificios para tornillos, que sostienen a la PCB en cada una de sus esquinas, como se muestra en la ilustración 3.61 A.

Para la tapa se realizó de la misma medida que la pieza de la caja, de igual forma tomando el rectángulo y redondeando las esquinas con sus agujeros para tornillos, el grueso de la tapa es de 2mm, tiene dos agujeros para los LED indicadores, tiene un agujero para el switch, otro para el molex de alimentación y señal de salida, otro para el display y sus cuatro tornillos, otro para el molex que conecta con el MP6050 y un rectángulo vaciado, profundo, donde al final contiene los tres agujeros para los potenciómetros, el cual está diseñado para que no sobresalgan. Para el display tiene un pequeño tope abajo y los agujeros para los tornillos de sujeción, así como las correspondientes simbologías de las conexiones a la alimentación, al molex del sensor y el led de señalización de salida. Todo esto se muestra en la ilustración 3.61B, que contiene al diseño de la tapa.



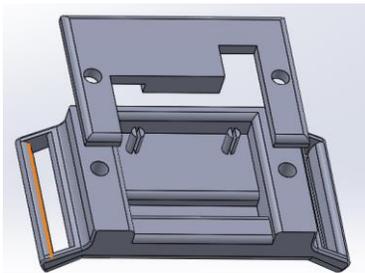
A. Caja



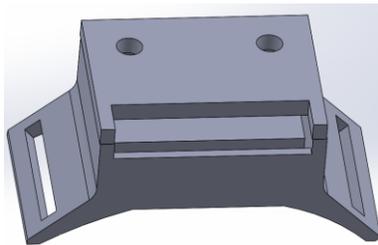
B. Tapa.

Ilustración 3.61 (A y B) Modelos 3D de módulo con acelerómetro.

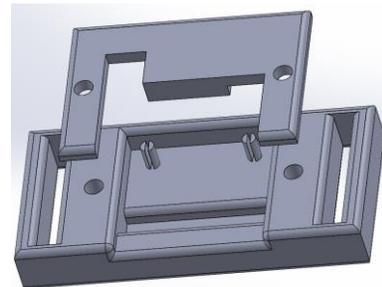
Para la adaptación del sensor, se tuvieron que hacer diferentes carcasas del acelerómetro, con sus respectivos agarres para incorporarlos al cuerpo, básicamente estos consisten en una tapa y una base, la tapa es un rectángulo cortado en la parte de abajo para mostrar el led indicativo, y la base tiene ranuras para fijar el sensor y los agarres para el resorte en las orillas dependiendo del adaptador, estos modelos se muestran en la ilustración 3.62.



A. Cascara pulsera.



B. Anillo.



C. Cascara cabeza.

Ilustración 3.62 (A, B y C) Modelos de cascara del acelerómetro.

Se utilizaron dos del modelo A para pierna y pulsera, el modelo del anillo y el modelo para la cabeza.

Ya con las piezas de los modelados, se exportaron en archivo “stl” y se mapeo la impresión en Ultimaker Cura, se mandaron a imprimir pieza por pieza, se quitaron los soportes y se procedió a la siguiente etapa el ensamblaje.

Primero se ensamblaron los componentes faltantes en la tapa, como lo son los potenciómetros, el display, LED, los puertos molex y el switch de encendido. Estos se fijaron, los LED se pegaron al igual que los puertos molex, el display se atornillo y los potenciómetros se ajustaron y se les puso una perilla para mejor aspecto. Casi todos los componentes de la tapa se les soldó los correspondientes cables con sus puertos para conectar al PCB, excepto el display que tiene un bus como cable para conectarse, como se muestra en la ilustración 3.63.

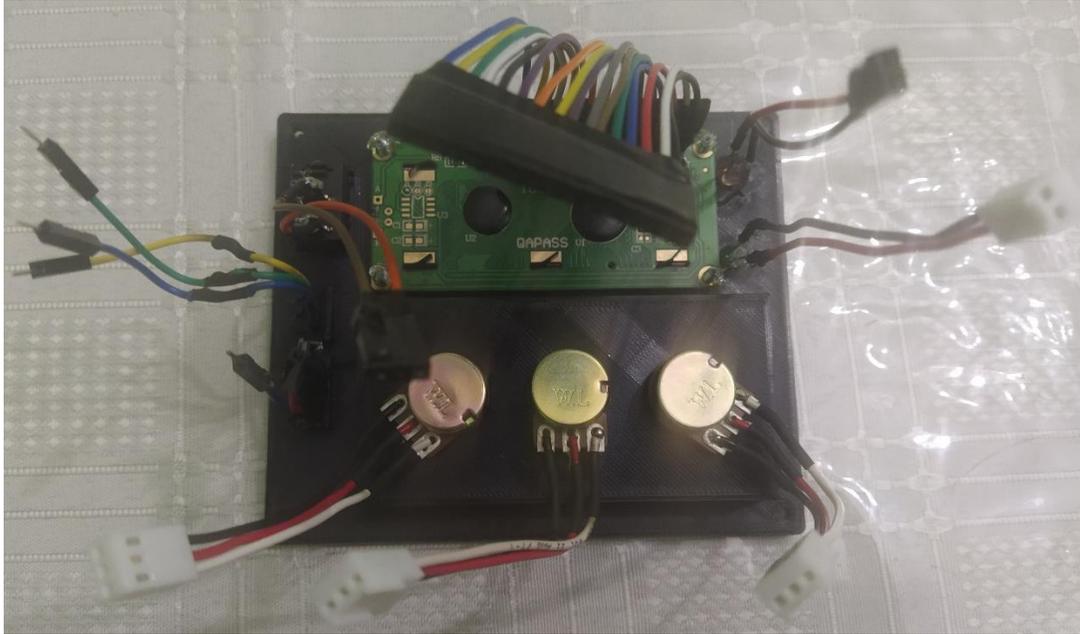


Ilustración 3.63 Tapa con los componentes.

Se ensambla la PCB a la caja por medio de los cuatro tornillos y después se conectaron los componentes que están en la tapa. Se hicieron las correspondientes pruebas de funcionamiento y se cerró fijándola con los tornillos de las esquinas.

Una vez ensamblado el módulo, se ensambla el MPU6050 en una de sus carcasas, este va fijado con tornillos. En la ilustración 3.64 se muestran como son las adaptaciones del sensor probadas en las diferentes partes del cuerpo. Para el modelo del anillo y el de la cabeza se les pega una esponja en la parte inferior para que no fuera a lastimar, además de que el resorte se coció en los agarres de la carcasa para fijarse y este tiene velcro en el final de la cinta para ajustarse a la anchura correspondiente y mantenerse fijo.

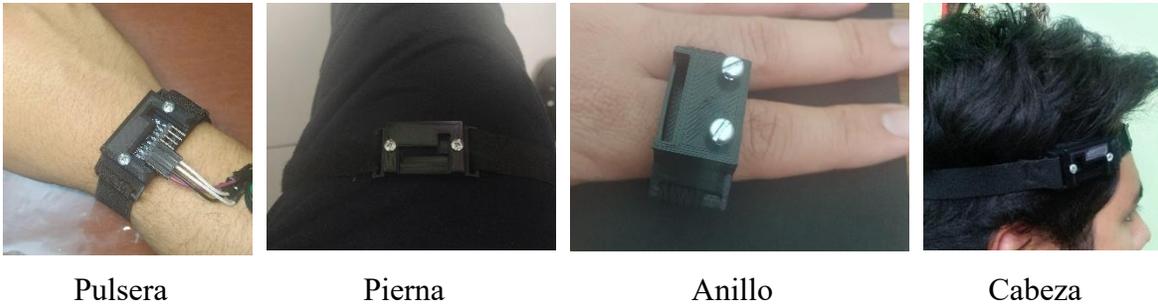


Ilustración 3.64 Cascaras del MP6050 con ejemplos de adaptación.

Por último, hicieron los correspondientes cables trenzados para las conexiones del módulo. En la ilustración 3.65 se muestra el módulo con acelerómetro con sus cables y las diferentes carcasas de interacción.



Ilustración 3.65 Módulo con acelerómetro.

4 Pruebas y resultados

Para poder concretar los diferentes módulos ya descritos en el anterior capítulo, se realizaron diversas pruebas, donde hubo resultados esperados y también no esperados, tanto en los experimentos con circuitos, como en el diseño de modelos 3D, errores al imprimir, errores en programación, cambios en el circuito, en conexión y funcionamiento.

En este capítulo se documentan algunas de las pruebas, y la solución cuando no se obtuvieron los resultados deseados.

4.1 Pruebas de posibles prototipos

Una vez revisados diversos tipos de interruptores y posibilidades, se planteó que prototipos se podrían crear, para esto algunos prototipos se hicieron elementalmente, de materiales como cartón, papel, tubos de plástico, etcétera; Para hacer pruebas de funcionalidad y a partir de ahí proceder a su diseño e impresión en 3D.

Se conformaron tres diseños principales, que se consideraron podrían ser útiles para algunas de las condiciones de los pacientes objetivo; los cuales corresponden al módulo de activación por soplo, módulo con acelerómetro y diversas variantes de interruptores. Para llegar a esto se descartaron dos prototipos principalmente, que se verán a continuación, los demás prototipos descartados solo fueron ideas.

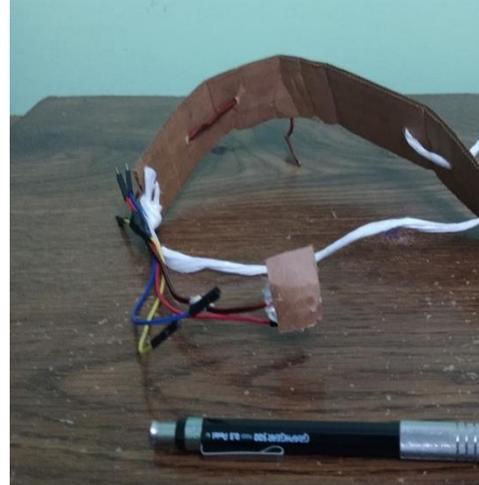
4.1.1 Módulo de activación por guiño

Para este módulo como su nombre lo indica, es para activación por guiño, se fundamenta en que una persona que pueda guiñar el ojo sea capaz de enviar una señal, para esto se planteó utilizar un sensor infrarrojo, pero al investigar se descubrió que es malo para la vista, después se planteó un botón capacitivo, el cual iría pegado al pómulo, donde se activaría al contacto con la piel, pero se descartó porque no era muy cómodo y ante el movimiento se despegaba de la cara. Por último, una cámara que detectara el parpadeo, pero fue descartado por su costo.

En la ilustración 4.1 A se muestra el prototipo construido con base a un botón capacitivo, la estructura esta hecha de cartón con un popote, está conectado a una protoboard con una compuerta lógica inversora, fue pegado con una pistola de silicón y está montado sobre un globo simulando una cabeza. En la ilustración 4.1 B se muestra el prototipo con infrarrojo, de igual forma esta hecho de cartón y un lazo, en la ilustración no se tiene el sensor.



A Prototipo con botón capacitivo.



B Prototipo con sensor infrarrojo.

Ilustración 4.1 (A y B) Prototipo activación por guiño.

4.1.2 Módulo de activación por un dedo

Este módulo tiene un funcionamiento muy sencillo, se activa al doblar un dedo, sirve con dos metales que al doblar el dedo entran en contacto y mandan una señal, también se pensó en una variante con liga que apretaba un interruptor por medio de la mano, fue descartado ya que era algo incomodo e impreciso, además de no concretarse físicamente, también supondría un mayor uso de la fuerza, para poder activar el interruptor.

En la ilustración 4.2 se muestra el modelo que se activaba al doblar el dedo, está construido con palitos de madera, silicón, dos bandas elásticas y pedazos de metal conectados a un cable. Aunque sencillo es incómodo y algo impreciso, además de que no se indago más en una posible mejora.

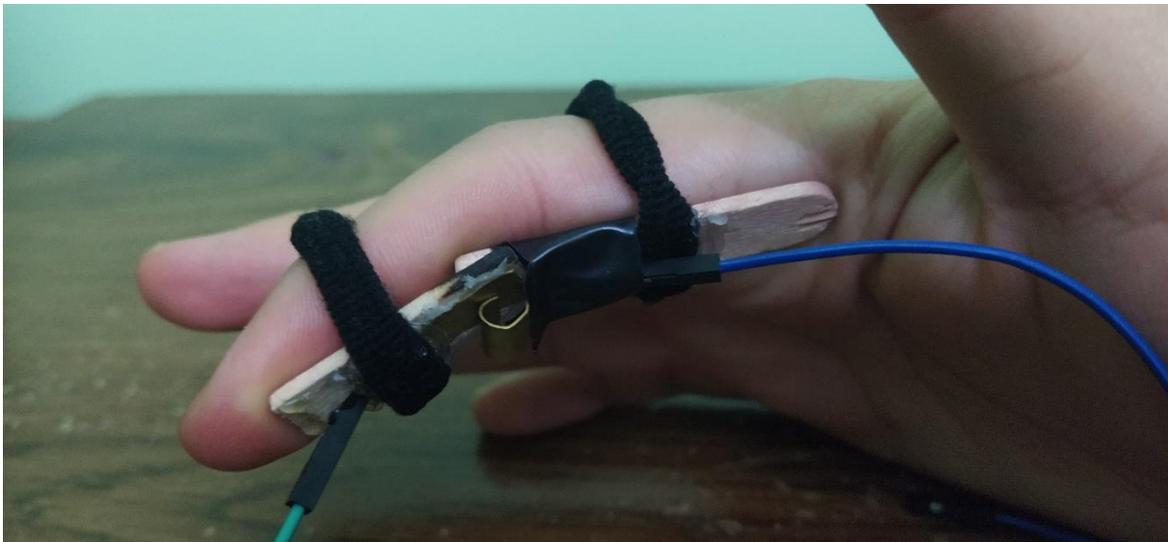


Ilustración 4.2 Prototipo de activación por un dedo.

4.2 Pruebas de construcción

En este capítulo se mostrarán las pruebas de construcción de los módulos descritos en el capítulo 3, como su modelado 3D, su impresión, en algunos el circuito y el ensamblaje. Donde se tuvieron algunos errores, se hicieron algunos cambios y nuevas ideas que fueron implementadas para mejorar los prototipos.

4.2.1 Pruebas de construcción para el módulo de activación por palma-muñeca de la mano

Este modelo fue desarrollado a partir del botón grande visto en la ilustración 3.8, dado a sus características fue necesario hacer bocetos en papel y medir de que tamaño debería ser. Con el prototipo en mente, se construyó un modelo de cartón de para el modelaje en 3D, se hizo a partir de una caja de cartón recortada y pegada con silicón, donde poco a poco fue moldeada para que se viera como la ilustración 4.3.



Ilustración 4.3 Prototipo de botón grande.

Al principio no tenía el soporte de la mano, ni el hoyo en la parte de atrás, fueron mejoras que se añadieron y se implementaron al momento de modelar la carcasa.

Pruebas de impresión y modelaje 3D

Con el prototipo echo de cartón como se muestra en la ilustración 4.3, se procedió a crear el modelo en 3D, para esto se realizó de manera minuciosa, ya que, al ser de gran tamaño, aumentaría el gasto de materiales y tiempo, llevando días para su impresión 3D. Con el afán de agilizar la construcción y material se revisó varias veces el diseño 3D, evitando choques entre las piezas y medidas de ensamblaje exactas.

Una parte que no se vio en el capítulo 3.1.1, fue la creación de una manija de agarre como se muestra en la ilustración 4.4, la cual servía para mover la el brazo más adelante y utilizar la muñeca para presionar el botón, este si fue impreso pero descartado, dado a que no servía mucho como soporte y estaba a una altura más baja de la mano.

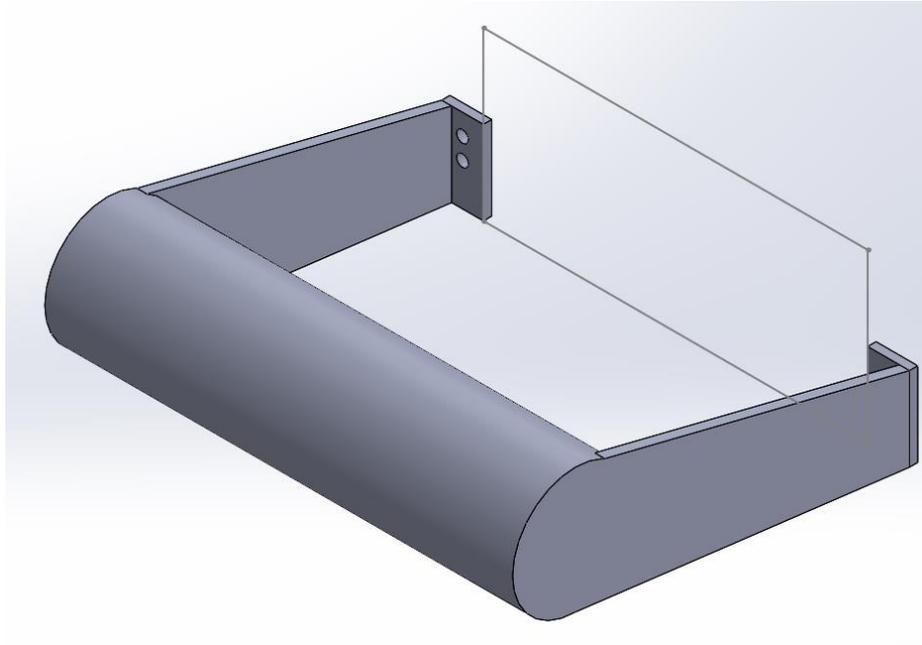


Ilustración 4.4 Agarre de mano.

Como no se tubo complicaciones de choque entre piezas se imprimió, fue cuando se tuvieron complicaciones, dado a que la impresora 3D solo aceptaba piezas del tamaño de 20x20x20 cm lo cual, en algunas piezas modeladas, excedía tales dimensiones, no se podía imprimir. Por lo que se realizó una modificación de seccionamiento de piezas, quitando patas de agarre que se muestran en la ilustración 4.5 y remodelándolas para una impresión externa y ensamblaje a la pieza del botón grande. Siendo unidas por tornillos. Aunque de igual forma se imprimió el modelo original en una impresora 3D más grande que se pudo conseguir.

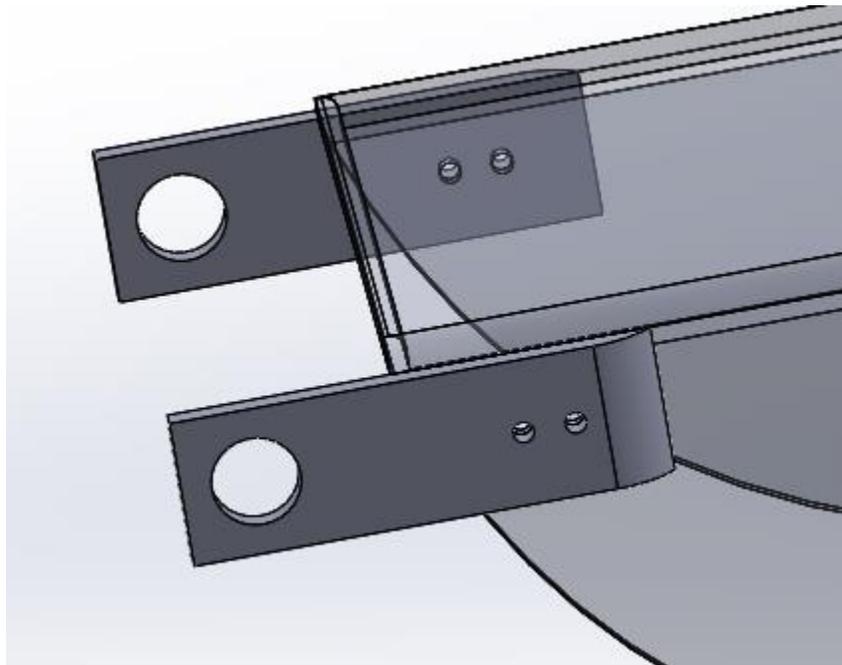


Ilustración 4.5 Patas de agarre.

Pese a que se realizaron las respectivas revisiones del modelo en 3D, se tuvieron errores de impresión, uno fue que la manija de agarre quedo muy baja y no se le vio utilidad, la segunda fue el fallo de la impresora 3D, en la adhesión a la cama, ya que, al ser piezas de gran magnitud, tendían a deformarse con el enfriamiento cometiendo errores al imprimir como se muestra en la ilustración 4.6, que es un error de impresión donde deforme toda la pieza haciéndola inservible. No obstante, con la impresora 3D grande se pudo imprimir toda la pieza sin ningún error y tomando el modelo original, creando el primer modelo físicamente.

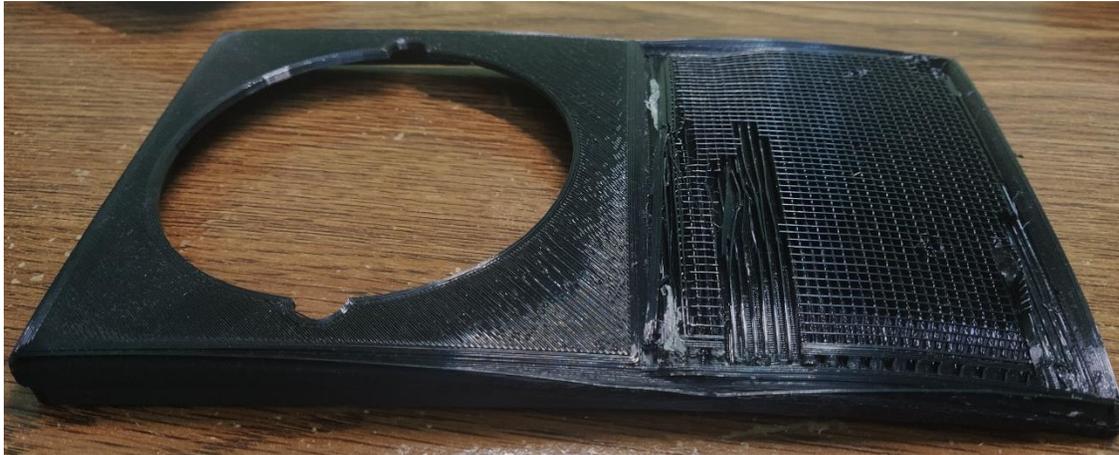


Ilustración 4.6 Error de impresión pieza botón grande.

4.2.2 Pruebas de construcción para módulo de activación por pulgar

Como se puede ver en el capítulo 3.1.2 este módulo tiene diferentes variantes, pero para consolidar los prototipos que se mostraron en el capítulo 3.1.2, se tuvieron que descartar algunos prototipos y otros fueron hechos como modelos de prueba para llegar al resultado.

Este módulo no tuvo ningún modelo previo hecho de cartón o papel, simplemente se tomó el botón deseado y se empezó a diseñar. Fueron dos variantes ya que el primer botón visto en la ilustración 3.11 no era muy sensible, el segundo botón visto en la ilustración 3.17 mejoro totalmente la sensación, aunque incrementando el grosor del prototipo.

Pruebas de impresión y modelaje 3D

Como se empezó por el modelaje en 3D para este tipo de prototipos, fue donde más errores se tuvieron.

Para la primera variante vista en el capítulo 3.1.2.1, pese a haber medido el botón con un calibrador vernier y comprobado la medición con regla a la hora de modelar en 3D e imprimir no se podían ensamblar los componentes, esto ocurre porque la impresora 3D solo puede imprimir con un determinado grosor mínimo, si no entra el grosor dentro de las posibilidades se crea un conflicto a la hora de ensamblarse. Otro error fue crear partes de la pieza demasiado delicadas lo cual a la hora de ensamblar o manipular la pieza se rompían, tomando varios prototipos de las piezas como se muestra en la ilustración 4.7, donde se ven varias de las piezas defectuosas de la variante 1 del módulo de activación por el dedo pulgar.



Ilustración 4.7 Piezas defectuosas variante 1 módulo de activación por pulgar.

Para la variante dos del módulo de activación por el dedo pulgar, se tomó otro botón visto en la ilustración 3.17, que mejoraba el prototipo, pero al ser más grande, se tuvo que cambiar todo el diseño 3D, dando como resultado un modelo más grueso que la variante anterior. Para este modelo, de igual forma que el anterior se hicieron las correspondientes medidas, creando un diseño en 3D e imprimiéndolo, este tuvo un error al ser demasiado largo ya que no se podía ensamblar el botón, dado a que necesitaba más espacio y en el agujero que se tenía no era suficientemente corto para ensamblar el interruptor, en la ilustración 4.8 se muestra algunas de estas piezas prototipo.



Ilustración 4.8 Piezas defectuosas de la variante 2 del módulo de activación por dedo pulgar.

Otra prueba que se hizo, fue hacer la conexión de las dos piezas del módulo, ya que una llevaba el botón y otra era el mango, para unir las se pensó en una especie de patitas con

gancho, pero al quedar muy justas y por el tipo de grosor se rompieron. Para solucionar este problema se realizó una modificación descrita en el capítulo 3.1.2, donde se diseñó un sistema que entra a presión por medio de unas pestañas largas y gruesas para soportar la presión. Una de las piezas con las pestañas rotas se muestra en la ilustración 4.8. Para compensar el tamaño de los modelos, se creó un modelo chico y otro grande.

4.2.3 Pruebas de construcción para módulo de activación por agarre

Para el módulo por activación por agarre, se tuvieron varios prototipos desde lo elemental hasta un modelo completamente diferente al visto en el capítulo 3.1.3. Se creó como una alternativa más a los interruptores, el primer prototipo se hizo de cartón, incluso no es funcional, pero fue tomado como base para crear el prototipo final.

Primero se pensó en poner un botón dentro de un rectángulo y que este se activara por medio de otro que lo aplastara, tal y como se muestra en la ilustración 4.9, que es un prototipo echo de cartón, este solo lo simula, no está bien adaptado y no está agarrado el gatillo en una sola pieza.



Ilustración 4.9 Prototipo de cartón del módulo agarre de mano.

A partir de este modelo se pensó en construir un prototipo funcional y con el mismo sistema del gatillo.

Pruebas de impresión y modelaje 3D

Primero se tomó como base la idea hecha en cartón para desarrollar el primer prototipo, este consistió en el mismo sistema, pero con dos microinterruptores que se accionarían al ser presionados por un gatillo, está sujeto por un lado con un cilindro para dar el movimiento. En la ilustración 4.10 se muestra el modelo 3D del prototipo descartado.

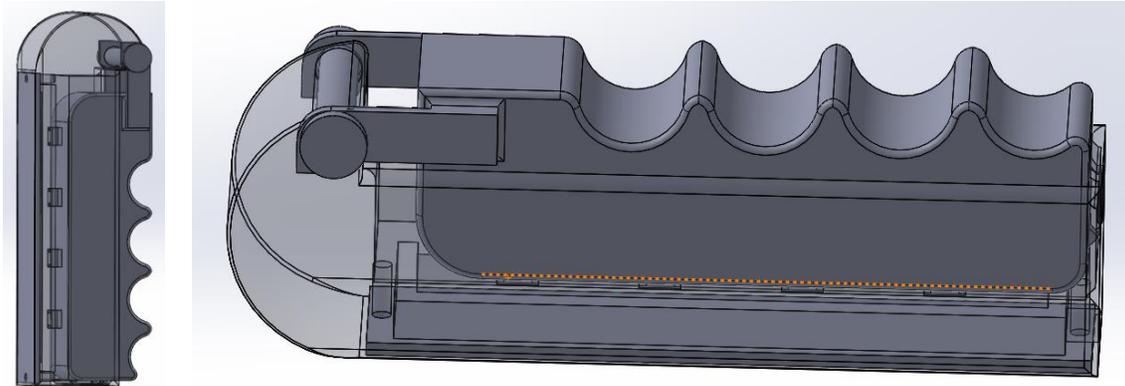


Ilustración 4.10 Modelo 3D prototipo descartado.

Como se puede observar en la ilustración 4.10 no es del todo estético y menos cómodo, ya que es muy grueso. Consta de cinco piezas, las cuales son: la pieza base, el gatillo o empuñadura, una tapa para los circuitos de atrás, dos tubos para el giro del gatillo. La empuñadura para los dedos del gatillo, que es incómoda y muy grande, el gatillo se traba al bajar con los bordes de la pieza base o los que contienen a los interruptores y en conjunto las dos piezas generan un agarre incómodo para la mano, además de que no existe un impulso de regreso, generando un atascamiento y no tiene la sensación para sentir que se presionó el interruptor.

En la ilustración 4.11 se observa el modelo impreso, este modelo tuvo problemas de impresión en unas de sus caras, y no realizaba el funcionamiento esperado, además de que al momento de medir se tuvieron errores, provocando que los botones no se pudieran ensamblar. AL mismo tiempo es muy incómodo, dado a que es muy grueso y cuadrado, lo que crea incomodidad en la mano.



Ilustración 4.11 Prototipo descartado del módulo para activación por agarre de mano.

Dado el funcionamiento del dispositivo y sus errores, se procedió a hacer el prototipo funcional que se muestra en el capítulo 3.1.3, donde de igual forma tuvo algunos errores que fueron corregidos, los cuales se muestran a continuación.

Al imprimir el nuevo prototipo, al momento de ensamblar el componente gatillo en la pieza base, se notó que los soportes de las esquinas eran muy delgados como se muestra en la ilustración 4.12, dando la sensación de que al momento de un golpe se romperían, por lo cual se modificaron, haciéndolos más gruesos para resistir un mayor impacto. En un prototipo de

esta pieza, se rompieron los soportes, ya que eran muy delgados como se puede ver en la ilustración 4.12.

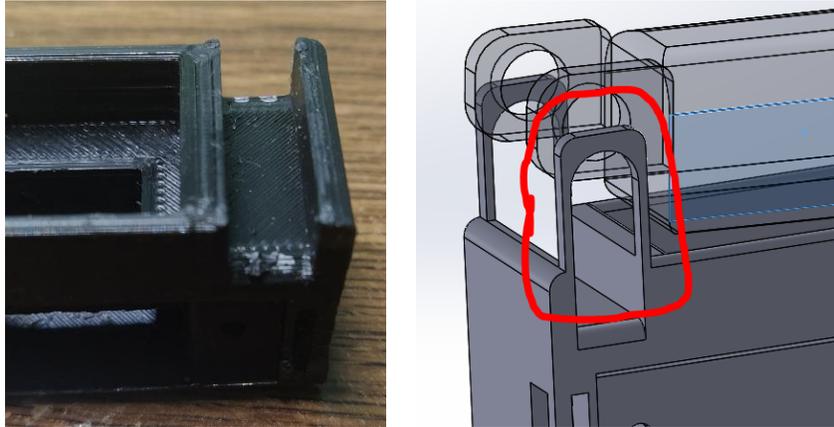


Ilustración 4.12 Error en soporte pieza base.

De igual manera se cambió el gatillo volviendo a modelar e imprimir, para que, entrara a la perfección en la cavidad de la pieza base, además de modificar el redondeo de la empuñadura para mayor comodidad, también se cambiaron las extremidades donde pasa el tubo de sujeción, juntándolas para no hacer más ancho el gatillo y no provocar errores de ensamblaje, en la ilustración 4.13 se muestran dos prototipos de gatillo.

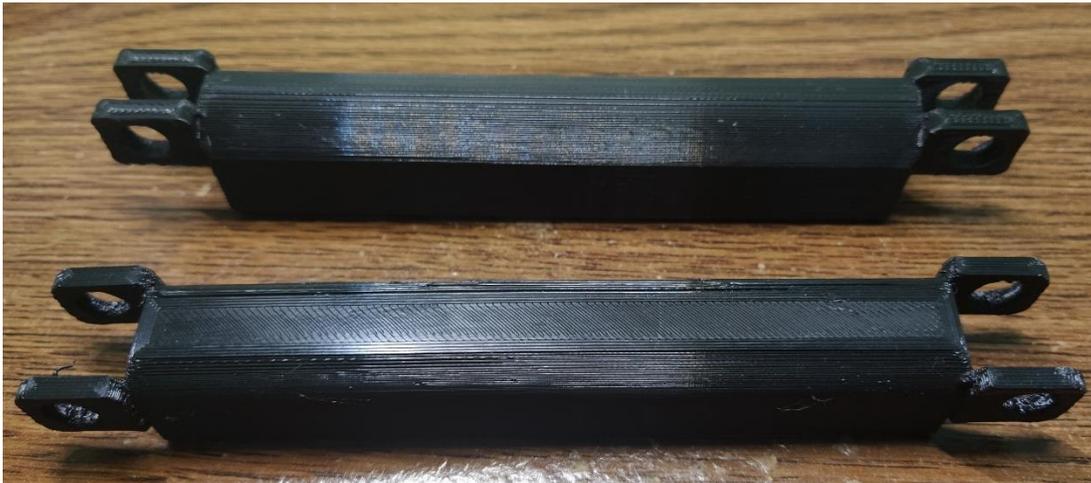


Ilustración 4.13 Gatillos descartados.

Una vez acoplados los interruptores y todos los componentes, se procedió a realizar una prueba de funcionamiento, la cual demostró que en diversas ocasiones al bajar el gatillo se quedaba atascado y los microinterruptores no eran capaces de empujar el gatillo de nuevo hacia arriba, para no mantener siempre la señal, por lo mismo se decidió acoplaron dos resortes pequeños, que se acoplaron en el gatillo, para permitir el regreso del interruptor y poder enviar pulsos, tal y como se muestra en la ilustración 4.14, donde se muestra el modelo 3D del gatillo con los agujeros del resorte.

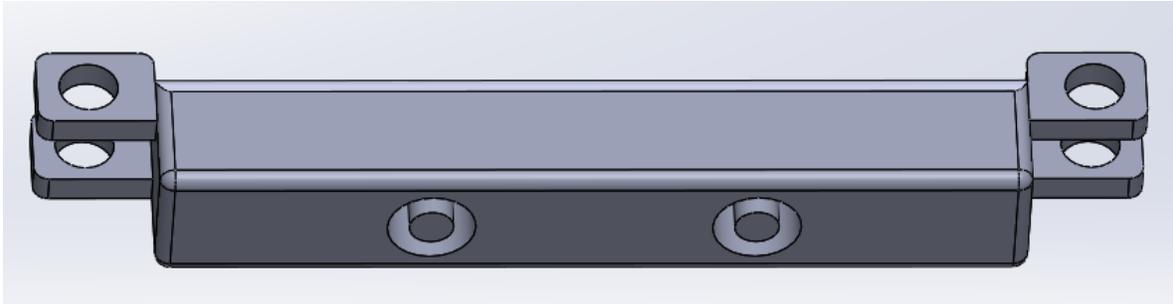


Ilustración 4.14 Gatillo con agujeros para resortes.

Para estos modelos se construyeron pruebas de ensamblaje de los tubos que lleva en las esquinas, probando la cavidad y ensamblaje, como se muestra en la ilustración 4.15. Estos fueron modelados en 3D e impresos varias veces hasta encontrar la medida exacta.



Ilustración 4.15 Pruebas de ensamblaje tubo.

4.2.4 Pruebas de construcción para módulo basado en soplido

Para el módulo de activación por soplido, se construyeron dos prototipos hechos de cartón que sirvieron para consolidar los dos módulos descritos en el capítulo 3.2.

El primer prototipo se basó en un sistema mecánico, como el de un interruptor, el cual, al pasar el flujo del aire una lámina de metal choca con otra, para esto se construyó un pequeño cuadro de papel tipo cartón donde un popote incrustado en el cubo, es el conducto para el flujo del aire, el flujo pega en la parte del metal empujándola contra la otra. Dependiendo cuanto era el flujo del aire accionaba el prototipo, pese a esto se descartó por ser muy impreciso y al moverse se mueve la lámina dando señales erróneas. En la ilustración 4.16 se muestra este prototipo.



Ilustración 4.16 Prototipo 1 en cartón del módulo de activación por soplido.

A partir del primer prototipo se cambió la idea de un funcionamiento mecánico y se adoptó un circuito con un sensor capaz de reconocer la señal. Para esto, como se vio en el capítulo 3.2, se optó por un micrófono, el cual detecta el soplo como si fuera un ruido. Como prototipo se creó una caja de cartón con una manguera incrustada, dentro de la caja se metió una protoboard, donde llevaba el circuito eléctrico, tal y como se muestra en la ilustración 4.17.

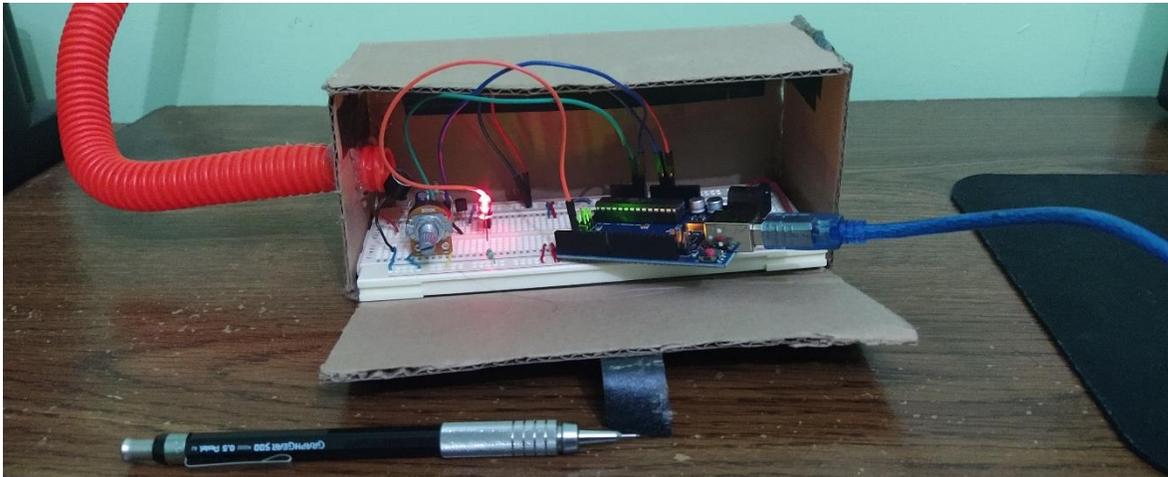


Ilustración 4.17 Prototipo 2 en cartón del módulo de activación por soplo.

Esto fue el modelo base para realizar una carcasa en 3D, un circuito impreso y mejoras para poder interactuar con el sensor.

Pruebas del circuito

Para la realización de este módulo, se tuvo que hacer un circuito tal y como se ve en el capítulo 3.2.2. Pero para llegar a tener el circuito definitivo se pasaron por varias pruebas.

Antes de crear un circuito impreso, se hicieron pruebas en una protoboard, la cual permitió cambiar componentes de manera sencilla. Para controlar el sistema se utilizó un microcontrolador, por lo cual las pruebas en placa se realizaron tanto en un Arduino uno como en el ATmega328P, consolidando el circuito primero con el Arduino uno y después utilizando el ATmega328P para las pruebas finales, tal y como se muestra en la ilustración 4.18, donde se muestra el circuito en sus primeras fases.

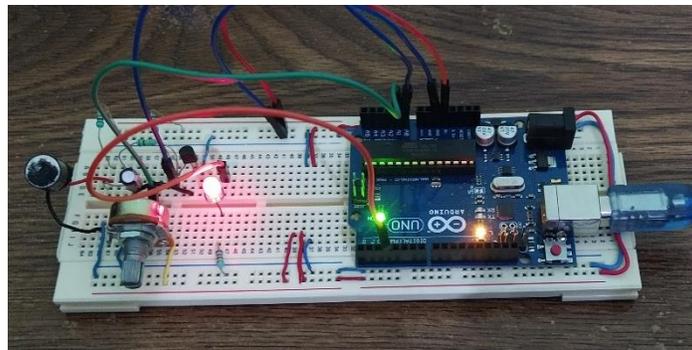


Ilustración 4.18 Protoboard con circuito de prueba.

Para la etapa de entrada de este módulo, se utilizó un micrófono, y se probó un circuito de amplificación antes de usar al LM386, esto se hizo porque los micrófonos no daban el voltaje

suficiente para poder acoplarse al microcontrolador y realizar su digitalización. También se colocaron filtros pasivos para filtrar el posible ruido. Respecto a los circuitos de amplificación, primero se intentó con un transistor, el cual fue armado, pero no funcionó como se esperaba, y por último el LM386 que fue el circuito integrado que se quedó para el prototipo final.

Respecto al experimento con transistor, se utilizó un BC547, el cual podía controlar la amplificación de la señal por medio de un potenciómetro, además acoplaba un pequeño filtro. Se recreo el circuito en Proteus, conectado al ATmega328P, donde presentaba la señal de salida a través de un LED. Con otro potenciómetro se controlaba la sensibilidad o el límite para mandar el pulso de salida. Todo esto se muestra en la ilustración 4.19, que es el esquemático del circuito.

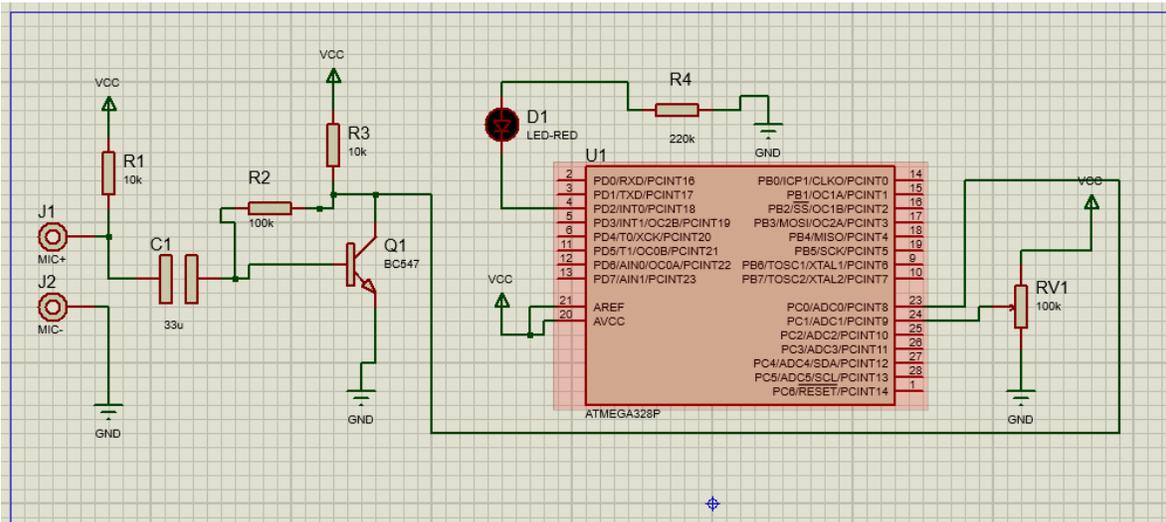


Ilustración 4.19 Circuito de prueba para el módulo de activación por soplo.

Se cambió al LM386 ya permitía un mejor control de la amplificación que el BC547. Para el circuito final, se agregaron más componentes para el control del tiempo y la pantalla LCD, tal y como se muestra en la ilustración 4.20. También en la variante 2 del capítulo 3.2 se agregó un botón de reset para algún caso de reinicio.

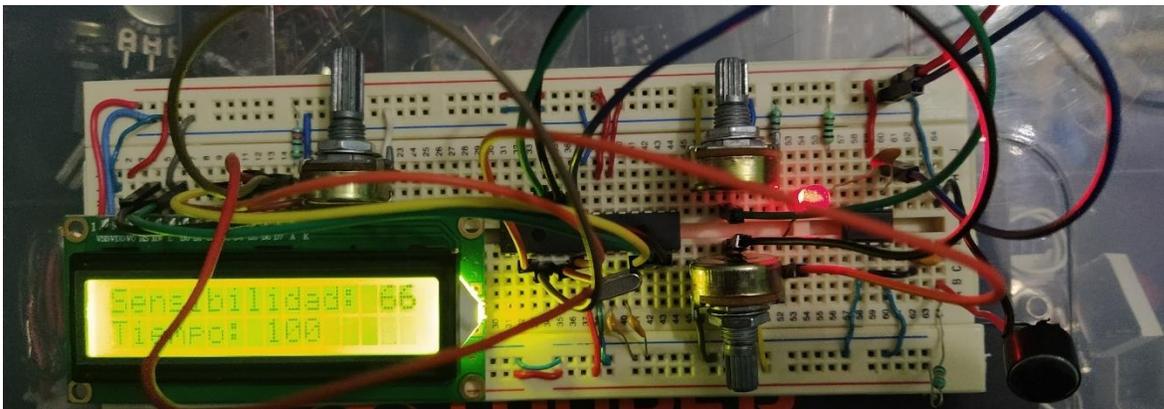


Ilustración 4.20 Circuito en protoboard del módulo de activación por soplo.

Con los circuitos probados se procedió a realizar el PCB, el cual tuvo varias pruebas para un buen funcionamiento.

Con el primer prototipo se utilizó el método del planchado, pero fue ineficiente, dado el tamaño de las pistas que se manejaba, por lo cual se cambió el método como se vio en el capítulo 3.2.6, que es un método de traspaso de tener atreves de acetona, en la ilustración 4.21, se muestra una de las fases de este método.



Ilustración 4.21 PCB en el agua para quitar papel.

Después de haber metido el PCB al ácido, se checó con un multímetro la continuidad a cada pista para averiguar si alguna estuviera en corto, con lo cual se descubrieron varias dado a que por una línea muy delgada del impreso se conectaban dos pistas. Para remover estas fallas se utilizó un cúter, para raspar las líneas y dejar el circuito sin cortos. Una vez soldado de igual manera se probó la continuidad con el multímetro para verificar las fallas en el circuito, se debe realizar esta tarea dado a que con algún corto en las pistas no funcionaría el circuito.

Dado al tamaño de las pistas, al momento de soldar se manchó la placa como se muestra en la ilustración 4.22, con las pruebas de continuidad echas se descartó este problema.

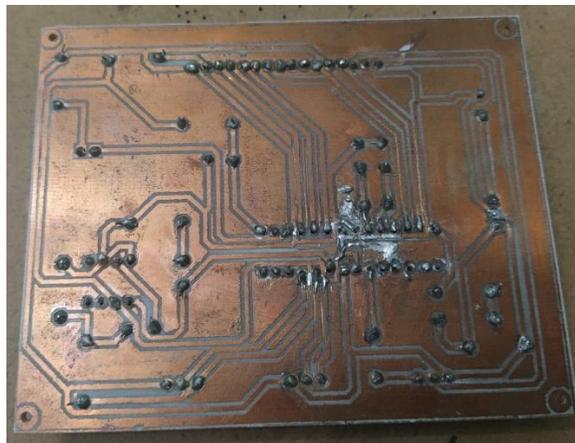


Ilustración 4.22 PCB con estaño.

Pruebas de impresión y modelaje 3D

Para el módulo por activación por sopro, se creó una caja en general para proteger al circuito y una base para la interacción, para llegar al modelo final se tuvieron varias pruebas.

Al modelar las cajas se tuvo que hacer diferentes pruebas, una de ellas fue la medida exacta para los tornillos, para esto se construyeron modelos con diferentes medidas de agujeros para saber cuál era el correcto como se muestra en la ilustración 4.23.

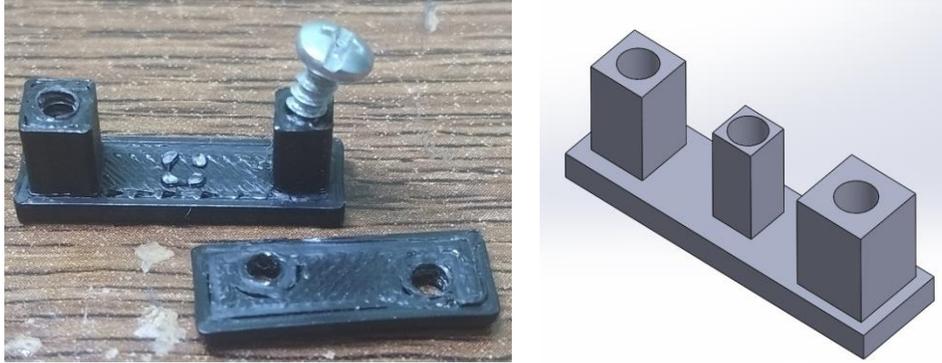


Ilustración 4.23 Prueba de tornillos modelo 3D y real.

De igual forma existe un prototipo que fue descartado debido a que la manguera que se utiliza para el sopro, se atora en la caja, de tal modo que, si se quiere extraer del módulo, se tiene que desarmar todo. Para este modelo se basó en dos piezas, una tapa y una caja. Fueron de un tamaño grande, dado a que la PCB hecha para este prototipo es más grande que los prototipos finales, la caja tiene soporte para PCB, ventilaciones y un conducto que fija la manguera o tuvo, él tuvo fue más pequeño que el prototipo final, el modelo 3D de la caja se muestra en la ilustración 4.24. Para la tapa se tienen los mismos agujeros que el descrito en el capítulo 3.2.5, solamente el agujero del tuvo es más pequeño y los agujeros para los tornillos están pegados en las esquinas, de tal forma que rompieron la tapa cuando se atornillo debido a la presión ejercida, en el prototipo final se corrigió este problema. La tapa se muestra en la ilustración 4.25.

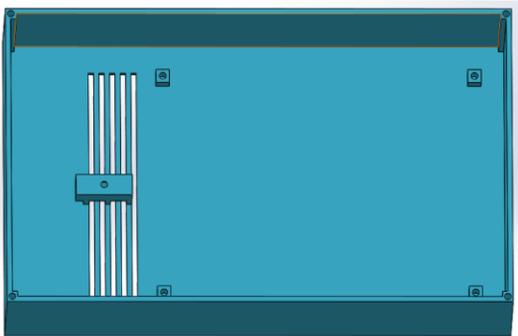


Ilustración 4.24 Caja Prototipo modulo con sopro.

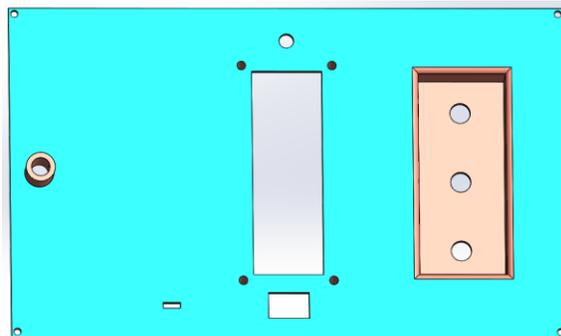


Ilustración 4.25 Tapa prototipo modulo con sopro

Se debe considerar, el tamaño del diseño 3D para poder imprimirlo, además del tiempo en el que se tarda. Para agilizar la construcción y ahorrar en material, el prototipo final se redujo.

Para el módulo de activación por sopro con manguera, en el prototipo final se tiene un conducto impreso hacia el micrófono como se muestra en la ilustración 3.42 A, donde se tuvo

que acortar, para tener más distancia respecto al micrófono y evitar ser llenado de saliva, además de que se modelo una rejilla como se muestra en la ilustración 4.26, que al chocar con el aire genera un mayor ruido ayudando al sensor a captar el soplo. Uno de los inconvenientes de este módulo es la saliva, por lo que en el diseño se crearon rejillas para expulsar la saliva, en la ilustración 4.26 se muestra la colocación de silicón para evitar que se extienda la saliva.

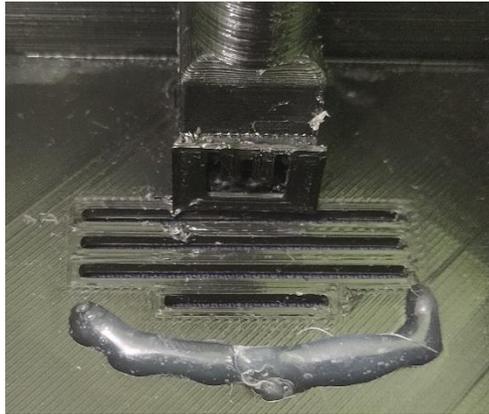


Ilustración 4.26 Rejilla en la caja del módulo por activación de soplo.

Para el soporte de la manguera se modelo una boquilla con una trayectoria curva en vez de lineal, pero fue descartada ya que la lineal era más sencilla y la curva fue inexacta en la medida del agujero, atorando él tuvo, el modelo de esta boquilla se muestra en la ilustración 4.28.



Ilustración 4.28 Soporte de Manguera descartado.

Pruebas de construcción del modelo real

Con todos los elementos listos del módulo, se realizó un ensamblaje que en algunos casos fue imperfecto, realizando las varias pruebas y correcciones.

Se ensamblaron los diferentes componentes en la pieza de la tapa, donde algunos no tuvieron la medida exacta y se debieron hacer correcciones, cortando o agrandando agujeros como el del LED y el display, además de los tornillos que en primer prototipo rompió las esquinas.

En la ilustración 4.29 se muestra el primer prototipo funcional, el cual se descartó debido a sus complicaciones y errores, se puede ver como la parte del display está cortada, dado a que no se incrustaba en la tapa, además de las esquinas rotas por los tornillos.

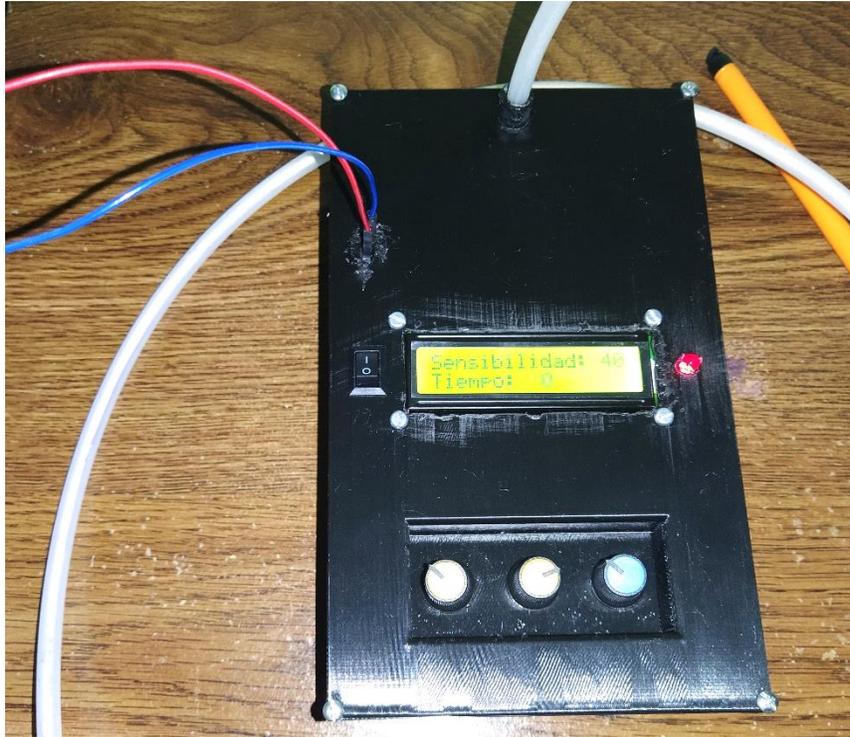


Ilustración 4.29 Prototipo descartado de modulo basado en soplido.

Se probó el encendido de los dispositivos, para verificar un correcto funcionamiento y que todos sus componentes funcionaran, dado a que muchas veces se encontró algún problema de desconexión, esto porque lleva muchas conexiones de la PCB a la tapa y al momento de cerrar la tapa, los cables se aplastaban, llegando a desoldar o desconectar los cables, generando fallas en el sistema. En la ilustración 4.30 se muestra el módulo abierto con sus múltiples conexiones, verificando que todas estuvieran conectadas.

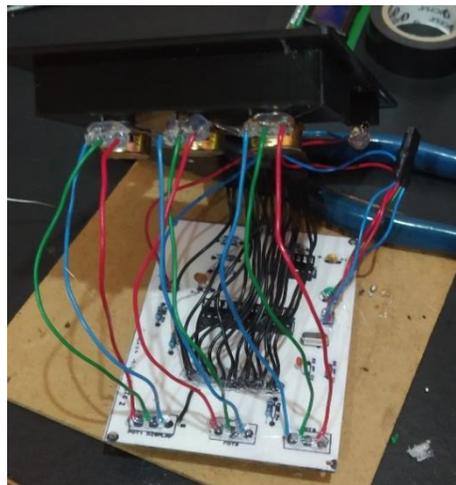


Ilustración 4.30 Conexiones del módulo basado en soplido por dentro.

4.2.3 Pruebas de construcción para módulo con acelerómetro

El módulo con acelerómetro fue el último prototipo en construirse, de igual forma que los anteriores modelos se recreó una base de cartón para tener un modelo de referencia, en la ilustración 4.31 se muestra una banda para la cabeza hecha en cartón con hilo de plástico, el cual iba a ser la adaptación al cuerpo, para este tipo de módulo.

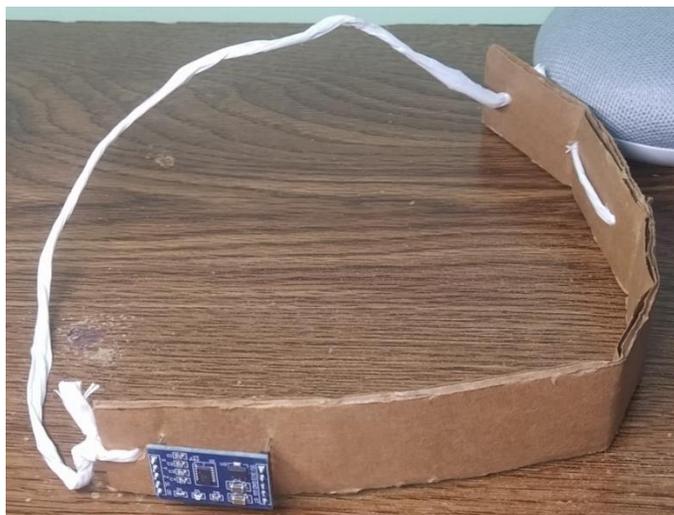


Ilustración 4.31 Banda con acelerómetro de cartón.

Pruebas del circuito

Para este módulo, se realizaron distintas pruebas de funcionamiento, dada la complejidad del módulo era necesario tener un circuito de prueba, el cual fue hecho en una protoboard, que permitía el cambio de piezas, los primeros componentes de prueba fue un Arduino uno y el módulo MPU6050, con estos se realizaron las primeras pruebas de funcionamiento.

Ya que se tenía el programa para controlar al módulo, se cambió el Arduino uno por el ATmega328P, con sus respectivos componentes de funcionamiento, además de que se agregó los potenciómetros para el control, como el display, en donde se realizaron pruebas de funcionamiento, además de la programación, como se muestra en la ilustración 4.32, donde se ve el circuito de prueba.

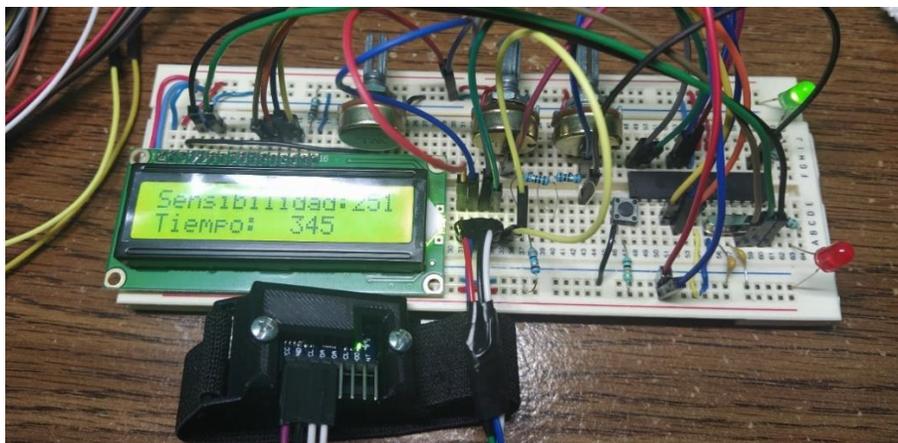


Ilustración 4.32 Circuito de prueba del módulo con acelerómetro.

Para la creación del PCB se tuvieron algunos inconvenientes en el traspaso del tóner, dado a que las pistas del circuito eran muy pequeñas. Ya con la PCB echa, se procedió a verificar la continuidad de cada pista, para evitar cualquier corto o mala conexión, donde se corrigió los cortos cortando con cúter los pequeños hilos de cobre que no se deshicieron con el ácido. Además, se tuvo un error en el trazado de una de las pistas del display estaba erróneamente conectada por lo que se realizó un puente con cable para corregir este error y desgastando la pista para quitar la mala conexión, como se muestra en la ilustración 4.33.



Ilustración 4.33 Puente en el PCB del módulo con acelerómetro.

Dado a que las pistas y los pads o marcas de los componentes eran muy pequeños, al momento de soldar con estaño, se ensucio la placa, pero se verifico la continuidad en estas para evitar cortos y por lo tanto un mal funcionamiento, como se muestra en la ilustración 4.34.

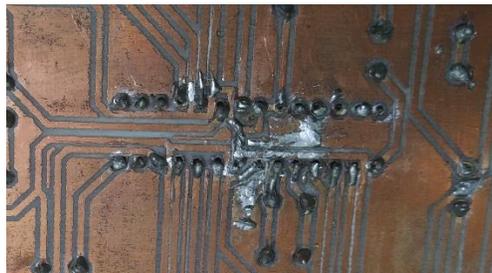


Ilustración 4.34 PCB manchada con estaño.

En el divisor de corriente visto en el esquemático del capítulo 3.3.2, se experimentó con diferentes resistencias, al principio se tuvo resistencias grandes, por lo que fue corregido para bajar el nivel de corriente y así evitar daños, implementando resistencias de menor valor.

Pruebas de impresión y modelaje 3D

Dado a que fue el último modulo, se tomaron muchas medidas de referencia del módulo de activación por sople, por lo cual no hubo más que un problema en su construcción el cual radica, en el momento de cerrar la caja, quedan apretados los cables generando desconexiones, por lo que se realizaron correcciones agrandando la caja 20 mm más aproximadamente. En

la ilustración 4.35 se muestra un modelo en 3D de la caja con mayor tamaño, permitiendo a los cables un mayor espacio para acomodarse y corrigiendo las desconexiones.

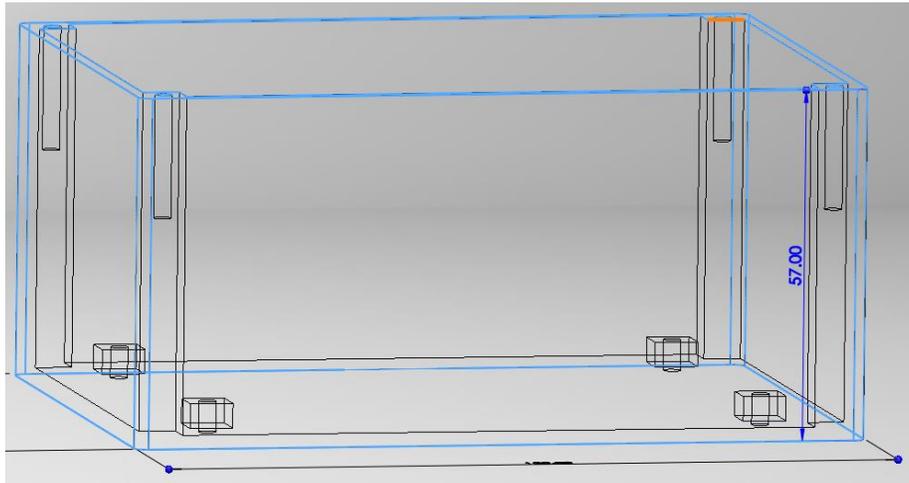


Ilustración 4.35 Modelo 3D de caja más grande.

Para el sensor, se realizaron varios diseños de las tapas que cubrían la carcasa del MPU6050, para lograr un buen recubrimiento, primero se realizó una adaptación en forma de anillo, pero no se logró el ajuste necesario, lo que hacía que el sensor se moviera dentro de la carcasa, por lo que fue rediseñando para un mejor ajuste, creando las carcasas que se muestran en el capítulo 3.3.5. Algunas de las carcasas y tapas descartadas para el sensor se muestran en la ilustración 4.36.



Ilustración 4.36 Tapas descartadas para la carcasa del acelerómetro.

Pruebas de construcción del modelo real

Dado al problema de aplastamiento de la caja, se observó que los cables tendían a desconectarse, por lo que se incorporó moxer para mantener la conexión más fija.

El inconveniente más grande y donde se realizaron más pruebas fue al momento de conectar al SAAC, ya que al tener un corto en la conexión al sensor (MPU6050), se bloqueaba el funcionamiento del módulo, esto se solucionó verificando la conexión del cable que iba conectado al sensor, descubriendo un corto en las terminales, el cual fue remplazado y se pudo solucionar el problema. En la ilustración 4.37 se muestra el cable en corto, realizando una medición de corriente.

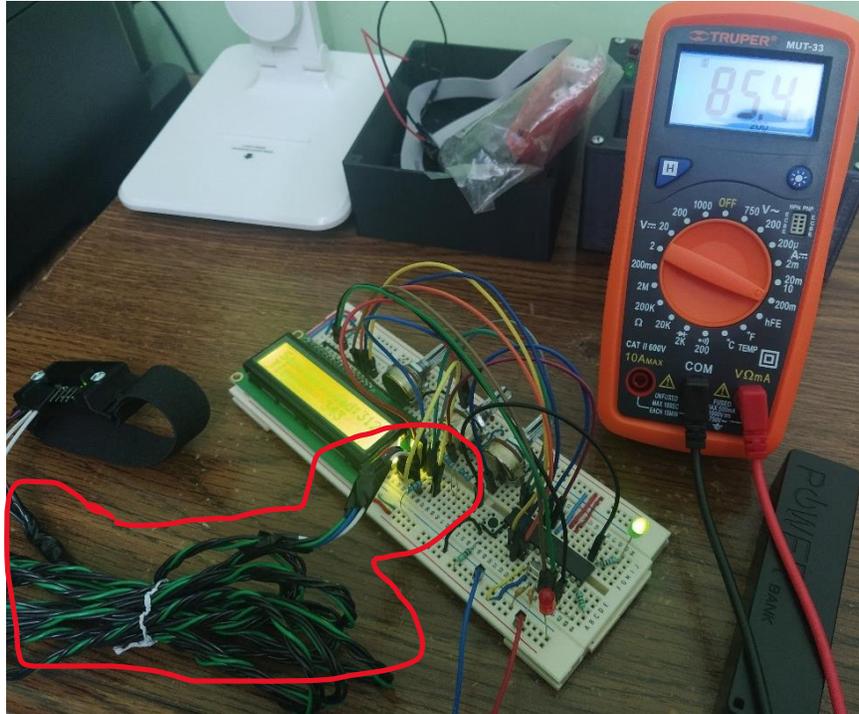


Ilustración 4.37 Cable en corto, medición de la corriente del módulo con acelerómetro.

En un momento se pensó que era la fuente de alimentación del módulo, por lo que se realizaron mediciones de la corriente y se sacó la alimentación de un puerto USB de la microcomputadora, pero no resultó, por lo que se llegó a la conclusión del cable, incluso se redujeron las resistencias para bajar el consumo.

4.3 Pruebas de programación

Para llegar a la versión final de los programas para los módulos, se realizaron diferentes pruebas, donde se cambiaron métodos y se agregaron componentes, dando como resultado diferentes versiones de los programas. El módulo de activación con soplido fue el que mayor número de visiones necesito, a partir de él, se logró reciclar código para el programa del módulo con acelerómetro dejando menos versiones.

4.3.1 Pruebas de programación del sensor basado en soplido

Para este módulo, se realizaron 12 versiones del programa, siendo las primeras versiones las que más cambios presentan.

Para la primera versión se planteó poner una comparación con un “if”, donde se comparaba la señal de entrada con una señal analógica de un potenciómetro, si la señal del micrófono era mayor que la del potenciómetro se mandaba la señal de salida, esta metodología era ineficaz dado a que la señal del potenciómetro era muy baja en comparación que la del micrófono y daba señales erróneas.

```
void setup() {  
  pinMode(2, OUTPUT);  
}
```

```

void loop() {
  int t =200;

  if (analogRead(A0)<=analogRead(A1)){
    //Si el valor es menor o      igual a parametro se enciende el led
    conectado al pin 12, de lo    contrario se apaga
    if (digitalRead(2) == 0){
      digitalWrite(2, HIGH);
      delay(t);
    }
  }else
  {
    if (digitalRead(2)== 1){
      digitalWrite(2, LOW);
      delay(t);
    }
  }
}

```

Después se retomaría este mismo sistema de comparación para llegar a la última versión del programa vista en el capítulo 3.2.3.

En la segunda versión, se intentó realizar un programa cuyo funcionamiento era tomar un determinado tiempo de la señal de entrada; la cual era el micrófono. Dado a que esta fluctúa con el tiempo, se promedió para poder compararla con la señal proveniente del potenciómetro, la cual fue nombrada sensibilidad. Pese a ser un buen calculo, este demoraba mucho y al momento de soplar consecutivamente, no era posible detectar la señal del micrófono y generaba una señal errónea, por lo que fue descartado.

```

  if (analogRead(A0)>=analogRead(A1)){      //Si el valor es menor o
  igual a parametro se enciende el led conectado al pin 12, de lo
  contrario se apaga
    digitalWrite(3, HIGH);

  }else
  {
    tempRef=millis();
    digitalWrite(3, LOW);
    while((millis()-tempRef)<=200)
    {
      //tiempo de retardo
    }
    digitalWrite(3, LOW);
    delay(200);
  }
}

```

A partir de la tercera versión hasta la séptima se adoptó un sistema de comparación, donde se utiliza dos condicionales dentro de otro, en donde, si el valor de la lectura analógica del sensor es igual o menor que la sensibilidad tomada de un potenciómetro, entra en otro

condicional, donde si la señal de salida está en uno manda un pulso, dentro del primer condicional, si no, compara si la señal de salida está en cero y si esta así, lo pone en uno, con su respectivo retardo. Para entender mejor este sistema de comparación es mejor verlo en el siguiente código.

```
if (analogRead(A0)<=s){
  if (digitalRead(1) == 0){
    digitalWrite(1, HIGH);
    delay(tiempo);
  }
}else
{
  if (digitalRead(1)== 1){
    digitalWrite(1, LOW);
    delay(tiempo);
  }
}
}
```

A partir de la segunda versión, se empezó a realizar las pruebas para mostrar a través de un display los valores de tiempo y sensibilidad, que se controlan a través de los potenciómetros. Para estas versiones se tomó la función de mapeo para las señales analógicas, que permite tomar el muestreo de la señal analógica que va de 0 a 1023 y convertirla en el rango que se dese, en los programas se tomó por lo general: 0 a 100 en sensibilidad, por el número que daba la señal de salida y 0 a 2000 en tiempo, porque se representaba el tiempo en milisegundos.

Las versiones del 1 al 7, se implementó el código directamente en la función principal o en el “void_loop()”, a partir de la versión 8 a la 12.5, se implementó una función aparte llamada “actualizarLCD()”, que permitía tener un código más limpio y mejor distribuido. Como el display reflejaba la información de los potenciómetros, y estos cambiaban dependiendo la posición en que se encontraban, la pantalla tenía que actualizarse, lo cual creo errores en la segunda versión, dado a que no borraba el contenido de la pantalla, para esto a partir de la tercera versión se tomó dos condicionales para limpiar la pantalla, en el caso de que se movieran los potenciómetros. Hasta la versión nueve se dejó de limpiar la pantalla con el comando “lcd.clear” y se optó por refrescar la información con espacios vacíos. En la décima versión se tomó un condicional que llamaba a “actualizarLCD()”, solo si se había registrado un cambio en el display. Cabe resaltar que se llama la función antes de iniciar el programa, para tener información inmediata al inicio, este es un error común que se podría tener.

Para la versión 12, se corrigieron valores iniciales, dado a que el programa iniciaba, dando un valor erróneo en la señal de salida, para ello se inicializo la variable, también se detectó un error en el código, la cual, al hacer una comparación, no permitía refrescar el display al inicio de programa, mostrando valores erróneos en el display.

```
sensibilidad=analogRead(A1);
tiempo=analogRead(A2);
}

void loop()
{
  int s = analogRead(A1); //establecer valor de referencia (0-1023)
  int t = analogRead(A2);
```

```

int tLED=map(t,0,1023,300,3000);

if((s>=sensibilidad+50) || (s<=sensibilidad-
50)|| (t>=tiempo+50)|| (t<=tiempo+50))
{
    sensibilidad=s;
    tiempo=t;
    actualizarLCD();
}

```

Esto se debía que, al momento de hacer la comparación, se tomaban dos valores de los potenciómetros, y estos valores habían sido tomados prácticamente uno después del otro, por lo que iban a ser siempre parecidos o iguales y no permitirían actualizar el display, además se tenía código innecesario, por lo que se llegó a la versión 12.5, donde se corregían estos errores y por lo tanto se tomó como la versión final.

4.3.2 Pruebas de programación del módulo con acelerómetro

El programa que lleva el módulo con acelerómetro, fue realizado después de hacer los programas del módulo de activación por soplo, tomando parte de su código y agilizando la realización del nuevo. Para este programa se realizaron cuatro versiones, las cuales cambiaron bastante gracias a que se tenía una base y solo se escribió el código para controlar el sensor y comparar la señal de entrada.

La primera versión, se basó en el ejemplo de funcionamiento del MPU6050, que se encuentre en internet (naylor mechatronics, 2016), se descartaron algunas líneas de código, principalmente que involucraban al giroscopio ya que no era de utilidad para el este módulo, este programa ya incluye las librerías y el test de conexión al sensor, solo se vio gráficamente los valores que salen de los ejes del acelerómetro. Esto se hizo con el objetivo de verificar el funcionamiento y solo graficaba las señales del acelerómetro, por lo cual no daba ninguna señal de salida.

```

#include "MPU6050.h"
#include "I2Cdev.h"
#include "Wire.h"

MPU6050 mpu(0x68);
int ax, ay, az;
int gx, gy, gz;

void setup() {
  Serial.begin(57600); //Iniciando puerto serial
  Wire.begin(); //Iniciando I2C

  mpu.initialize(); //Iniciando el sensor

  if (mpu.testConnection())
    Serial.println("Sensor iniciado correctamente");
  else
    Serial.println("Error al iniciar el sensor");
}

void loop() {

```

```

// Leer las aceleraciones y velocidades angulares

mpu.getAcceleration(&ax, &ay, &az);
mpu.getRotation(&gx, &gy, &gz);

Serial.print("a[x y z] g[x y z]:\t");
Serial.print(ax); Serial.print("\t");
Serial.print(ay); Serial.print("\t");
Serial.print(az); Serial.print("\t");
Serial.print(gx); Serial.print("\t");
Serial.print(gy); Serial.print("\t");
Serial.println(gz);
delay(100);
}

```

En la segunda versión del programa se implementó el funcionamiento principal, el cual está descrito en el capítulo 3.3.3, es casi igual, solo que en algunas señales de salida están en lógica inversa. Este programa incorporó el vector, la señal de salida y el test de conexión con indicador. Esta versión no estaba completa, dado a que no se tenía el control a través de un display.

Para la tercera versión, se tomó el funcionamiento principal de la segunda versión, pero esta vez se incorporó el display LCD, donde se muestran los valores de los potenciómetros, solo que estos se actualizaban cada vez que entraba en el ciclo consumiendo parte del procesamiento, por lo que se pasó a la cuarta versión del programa.

Para la cuarta versión, se incorporó en el sistema, una actualización de pantalla que se activaba cada vez que detectara un cambio en los potenciómetros, para evitar consumir parte del procesamiento y tener una mejor fluidez del programa.

```

void setup()
{
  lcd.begin(16, 2);
  lcd.home();
  pinMode(9,1); //salida status mcu
  pinMode(8,1); //replica respuesta led
  pinMode(10,1); //salida respuesta Señal
  Wire.begin();
  s=analogRead(A0); //Lectura del Potenciómetro para la sensibilidad
  t=analogRead(A1); //Lectura del Potenciómetro para el Tiempo
  sensibilidad=map(s,0,1023,0,400);
  tiempo=map(t,0,1023,100,2000);
  actualizarLCD();

  mpu.initialize();
  if (mpu.testConnection())
  {
    digitalWrite(9,1);
  }
  else
    errorMcu();
}

void loop()

```

```

{
  int s1=analogRead(A0); //Lectura del Potenciómetro para la
sensibilidad
  int t1=analogRead(A1); //Lectura del Potenciómetro para el Tiempo

  if((s1>=s+10) || (s1<=s-10)|| (t1>=t+10)|| (t1<=t-10))
  {
    sensibilidad=map(s1,0,1023,0,400);
    tiempo=map(t1,0,1023,100,2000);
    actualizarLCD();
    s=s1;
    t=t1;
  }
}

```

Pese a funcionar, al probar con el SAAC, se detectó un error en el código generando una señal falsa al iniciar el módulo, además el condicional que se había agregado para detectar el cambio en los potenciómetros, estaba siendo afectado por un error en código he invalidando sus operaciones, por lo que se creó la versión 4.1, que es la última versión del programa que corrige estos bugs del sistema. Para componer este error se inicializaron las variables de salida antes, para evitar señales falsas, se encontró el error de comparación en el condicional de los cambios en el potenciómetro y se re ubico, para no hacer una comparación de sensibilidad y tiempo con los mismos valores, para tener una buena actualización del display.

4.4 Pruebas y resultados de módulos finales en el SAAC

Con los módulos hechos vistos en el capítulo 3, se procedió a realizar pruebas de funcionamiento conectados al SAAC, checando la capacidad de respuesta, la comodidad, el control de la señal y el desarrollo de los módulos en general. Recordando que el SAAC está cargado en una Raspberry Pi 4B, donde los módulos se conectan a través de sus pines y en los módulos por activación por soplo y el módulo con acelerómetro son alimentados desde el mismo SAAC.

4.4.1 Módulo de activación por palma-muñeca de la mano

Este módulo, se basa en el interruptor, por lo cual tiene un funcionamiento muy sencillo, por lo que para conectarse al SAAC, solo se necesita conectarse a dos pines y no necesita una alimentación externa, está construido para usarse en una persona con movilidad en la mano, se recomienda utilizar en posición de sedestación en cama, con el objetivo de evitar incomodidad en el paciente, en el caso que no sea posible operar el módulo en esa posición, se recomienda cambiar por algún otro módulo realizado.

Prueba de funcionamiento en el SAAC

Una vez conectado el módulo al SAAC, se procedió a realizar las pruebas de funcionamiento, las cuales consistieron en realizar una palabra y elegir algún pictograma, tomando el tiempo y observando el funcionamiento. Se realizaron tres pruebas, y los resultados se muestran en la tabla 4.1 La palabra tomada fue Hola, tomando distintos caminos en el selector donde la primera prueba fue la más larga y la tercera prueba fue la selección más rápida. Para el pictograma se tomó Hola de la sección de saludos, donde en la primera prueba, espera el recorrido del cursor y la tercera, es la selección más rápida para llegar. Siendo la segunda el punto intermedio, tanto para la palabra como el pictograma.

Tabla 4.1 Prueba de funcionamiento para el módulo para palma-muñeca de mano.

| | Prueba 1 | Prueba 2 | Prueba3 | Promedio |
|------------|-----------------|-----------------|----------------|-----------------|
| Palabra | 25.64 s | 23.60 s | 12.90 s | 20.71 |
| Pictograma | 12.76 s | 10.55 s | 7.49 s | 10.26 |

Observaciones

Al realizar las pruebas, se pudo observar que este módulo, tiene una capacidad de respuesta muy buena, mandando la señal incluso si el botón no había sido presionado en su totalidad. Con respecto a su comodidad, es buena, no cansa el brazo ni la mano, se puede operar de distintas formas, tanto con la palma de la mano, como con la muñeca. No obstante, si no se acomoda en una posición adecuada, como una mayor altura al brazo o muy pegado al cuerpo, puede llegar a cansar, dado a que se estira la mano, por lo cual se debe tener cerca del cuerpo y la persona debe estar sentada, para tener el brazo descansando arriba del módulo. Las almohadillas en la parte superior del módulo, lo hacen más cómodo.

El módulo con el botón grande para activación por la muñeca-mano, tiene una buena respuesta y comodidad, siempre y cuando se tenga un poco de movilidad en la mano o brazo, en caso de no tener control en el brazo o mano es preferible usar algún otro modulo.

4.4.2 Módulo de activación por pulgar

Este módulo está basado en un pequeño botón, cuenta con dos variantes, las cuales se diferencian por el botón empleado, para conectarse al SAAC, se conectan dos pines, no necesitan alimentación externa y su uso es para aquellas personas que tengan movilidad en la mano, siendo más específicos el dedo pulgar. Es más compacto y se sujeta de la mano con un resorte. En la ilustración 4.38 se muestra la variante 1 y 2.



Varaintel



Variante 2

Ilustración 4.38 Variante 1 y 2 del módulo de activación por pulgar de mano.

Prueba de funcionamiento en el SAAC

Una vez conectado el módulo al SAAC, se procedió a realizar pruebas de funcionamiento, las cuales se tomaron tres lecturas de tiempo, donde se seleccionaba una palabra y un pictograma. Se escogió la palabra Hola y el pictograma fue Hola. La primera prueba fue la más lenta, la segunda un método intermedio y la tercera fue el método más rápido para seleccionar la palabra y el pictograma.

- ◆ Para la variante 1 del módulo de activación por pulgar se muestra en la tabla 4.2.

Tabla 4.2 Prueba de funcionamiento para el módulo de activación por pulgar V1.

| | Prueba 1 | Prueba 2 | Prueba3 | Promedio |
|------------|-----------------|-----------------|----------------|-----------------|
| Palabra | 60 s | 26.14 s | 29.30 s | 38.48 s |
| Pictograma | 16.81 s | 16.84 s | 8.60 s | 14.08 s |

- ◆ Para la variante 2 del módulo de activación por pulgar se muestra en la tabla 4.3.

Tabla 4.3 Prueba de funcionamiento para el módulo de activación por pulgar V2.

| | Prueba 1 | Prueba 2 | Prueba3 | Promedio |
|------------|-----------------|-----------------|----------------|-----------------|
| Palabra | 32.60 s | 27.05 s | 28.60 s | 29.41 s |
| Pictograma | 19.86 s | 13.27 s | 7.34 s | 13.49 s |

Observaciones

Una vez echas las pruebas con el SAAC se observó lo siguiente:

- ◆ Para la variante 1:
El módulo tiene una mala respuesta al tacto, dado a que no envía la señal incluso si se oprime más de la mitad del botón, no envía la señal a menos de que presiones hasta abajo del botón y no se percibe cuando ya se ha activado. Es cómodo el agarre y la sujeción porque es delgado, pero cansa el dedo, dado a que está muy arriba el botón y se ejerce mucha fuerza para poder activarlo, no se recomienda utilizar en periodos largos de tiempo.
- ◆ Para la variante 2:
Pese a ser más grueso el cilindro que la variante 1, tiene una mejor comodidad en el agarre, además de que se tienen dos tamaños, que se adaptan mejor a la mano. El botón empleado mejora mucho con respecto a la variante 1, ya que al presionarlo se puede sentir que se activó, no es necesario presionar todo el botón para activarlo, lo cual mejora su respuesta comparada con la variante 1, da la sensación de que se ha activado y es más grande. Su respuesta es mejor como se ve en la tabla 4.3.

Este módulo es sencillo y bueno, se recomienda para personas con movimiento en la mano, es preferible utilizar la variante 2, dado a su respuesta y comodidad. En el caso de que no se tenga movilidad en la mano, se recomienda utilizar algún otro módulo.

4.4.3 Módulo por activación de agarre

Este módulo está basado en dos interruptores, que son accionados a través de un gatillo, al igual que los módulos con interruptores, para poder conectarse a al SAAC, solo se ocupan

dos pines, y no necesita alimentación externa. Se recomienda para personas que tengan movilidad en la mano y puedan apretar el gatillo con poca fuerza. En la ilustración 4.39 se muestra el uso de este módulo.



Ilustración 4.39 Módulo de activación por agarre ejemplo de uso.

Prueba de funcionamiento en el SAAC

Una vez conectado el módulo al SAAC, se procedió a realizar pruebas de funcionamiento, las cuales se tomaron tres lecturas de tiempo, donde se seleccionaba una palabra y un pictograma. Se escogió la palabra Hola y el pictograma fue Hola. La primera prueba fue la más lenta, la segunda un método intermedio y la tercera fue el método más rápido para poder seleccionar la palabra y el pictograma.

En la tabla 4.4 se muestra los tiempos de las pruebas mostrando la velocidad de respuesta del módulo.

Tabla 4.4 Prueba de funcionamiento para el módulo de activación por agarre.

| | Prueba 1 | Prueba 2 | Prueba3 | Promedio |
|------------|----------|----------|---------|----------|
| Palabra | 34.40 s | 25.09 s | 27.80 s | 29.09 s |
| Pictograma | 11.95 s | 10.18 s | 8.42 s | 10.18 s |

Dado a los tiempos que se muestran en la tabla 4.4, se puede ver que es un módulo bastante eficiente y de rápida respuesta.

Observaciones

Una vez echo las pruebas en el SAAC, se puede observar que, es un módulo con una respuesta buena, no se traba al bajar como los prototipos anteriores de este módulo, puede reaccionar incluso si solo se presiona un lado del gatillo, se siente al tacto que ha sido presionado el interruptor, lo cual es bueno para poder detectar fallos. La comodidad es buena, pero para personas con manos pequeñas puede ser más incómodo, la sujeción atreves del resorte es buena y se debe considerar tener la suficiente fuerza de agarre para poder presionarlo, aunque el módulo no esté tan duro. Se consiguió un buen tiempo de respuesta y es posible manejarlo por largos periodos de tiempo sin sentir algún malestar en la mano.

Si no se tiene la suficiente fuerza en la mano, se recomienda utilizar otro módulo.

4.4.4 Módulo de activación por soplido

Este módulo, incorpora un circuito que gestiona la señal de entrada y la señal de salida. La interacción es a través de un micrófono y dos potenciómetros para modificar la señal de salida. Para incorporar al SAAC, es necesario tener una conexión a la alimentación del SAAC de 5 volts y la señal de salida es a través de un pin con lógica inversa, en un voltaje de 3 volts. Para este tipo de módulo, se tienen dos variantes, las cuales cambian en el método de interacción del usuario, siendo uno a través de un tubo y otro con una diadema que contiene directamente el micrófono. Su uso recomendado es para aquellas personas que no tengan movilidad en el cuerpo, pero tengan la capacidad de controlar su respiración y poder soplar.

Prueba de funcionamiento en el SAAC

Una vez conectado el módulo al SAAC, se procedió a realizar pruebas de funcionamiento, las cuales se tomaron tres lecturas de tiempo, donde se seleccionaba una palabra y un pictograma. Se escogió la palabra Hola y el pictograma fue Hola. La primera prueba fue la más lenta, la segunda un método intermedio y la tercera fue el método más rápido para poder seleccionar la palabra y el pictograma.

- ◆ Para la variante 1 del módulo de activación por soplo se muestra en la tabla 4.5, su método de interacción es por manguera.

Tabla 4.5 Prueba de funcionamiento para el módulo de activación por soplo V1.

| | Prueba 1 | Prueba 2 | Prueba3 | Promedio |
|------------|----------|----------|---------|----------|
| Palabra | 37.39 s | 41.38 s | 35.89 s | 38.22 s |
| Pictograma | 15.30 s | 17.75 s | 10.3 s | 14.45 s |

- ◆ Para la variante 2 del módulo de activación por soplo se muestra en la tabla 4.6, su método de interacción es a través de la diadema con el micrófono.

Tabla 4.6 Prueba de funcionamiento para el módulo de activación por soplo V2.

| | Prueba 1 | Prueba 2 | Prueba3 | Promedio |
|------------|----------|----------|---------|----------|
| Palabra | 35.16 s | 33.82 s | 28.61 s | 32.53 s |
| Pictograma | 12.42 s | 11.43 s | 8.70 s | 10.85 s |

Observaciones

Una vez echas las pruebas con el SAAC, se puede observar que, la variante 1, que es la que contiene la manguera, es menos eficiente que la variante 2, que es la que contiene la diadema, dado a los tiempos de respuesta que se muestran en las tablas 4.5 y 4.6, por lo que justifica a la variante 2 como una mejor versión para este tipo de módulo y que a su vez tiene una mejor respuesta.

- ◆ Para la variante 1:
Se tuvo una mala respuesta a la sensibilidad, pese a ser regulada. Si no se sopla con la suficiente fuerza el sensor no detecta el soplo, en algunos casos dio señales falsas y empeora la calidad de respuesta. Sus controles tienen buena respuesta y conectarlo al SAAC es sencillo. Al soplar por largos periodos se puede observar una

acumulación de saliva, la cual sale por sus rendijas, pero puede llegar a dañar el circuito, por el tipo de manguera es difícil acomodar la boquilla, para un mejor control siempre se debe utilizar el brazo con soporte. Es cómodo una vez acomodado, pero no tiene una buena respuesta.

◆ Para la variante 2:

Para esta variante con diadema, se tiene una buena señal de respuesta, dado a que el sensor está más cerca de la boca, sus controles son buenos, se conecta fácilmente y es cómodo y fácil de ajustar. No es susceptible al ruido del ambiente, pero puede llegar a enviar señales erróneas, gracias al aire que sale de la nariz e interactúa con el sensor. Una de sus ventajas es que no acumula saliva y permite cambiar la diadema fácilmente.

En el caso de contar con una movilidad en una parte del cuerpo, se puede utilizar otro módulo, aunque estos módulos son una buena alternativa para aquellas personas que no tengan movimiento.

4.4.5 Módulo con acelerómetro

Este módulo contiene un circuito que gestiona las señales, su principal característica es un acelerómetro, que detecta los movimientos cortos, para activar la señal de salida. Para poder conectarse al SAAC, es necesario tomar una alimentación de 5 volts del SAAC y mandar la señal de salida a un pin del SAAC, esta señal es en lógica inversa y con un voltaje de 3 volts. Se recomienda para personas con alguna movilidad en el cuerpo, donde se acoplan a través de adaptadores del sensor, como se muestra en la ilustración 4.40, donde se colocó en el brazo.



Ilustración 4.40 Ejemplo de colocación del acelerómetro.

Prueba de funcionamiento en el SAAC

Una vez conectado el módulo al SAAC, se procedió a realizar pruebas de funcionamiento, las cuales se tomaron tres lecturas de tiempo, donde se seleccionaba una palabra y un pictograma. Se escogió la palabra Hola y el pictograma fue Hola. La primera prueba fue la

más lenta, la segunda un método intermedio y la tercera fue el método más rápido para poder seleccionar la palabra y el pictograma.

Para estas pruebas, se tomó el accesorio para el brazo, como el de la ilustración 4.40, cabe resaltar que tiene el mismo funcionamiento para los demás adaptadores. En la tabla 4.7 se muestran los resultados.

Tabla 4.7 Prueba de funcionamiento para el módulo con acelerómetro en el brazo.

| | Prueba 1 | Prueba 2 | Prueba3 | Promedio |
|------------|-----------------|-----------------|----------------|-----------------|
| Palabra | 34.86 s | 26.30 s | 27.8 s | 29.65 s |
| Pictograma | 12.8 s | 11.10 s | 9.7 s | 11.2 s |

Observaciones

Una vez echas las pruebas con el SAAC, se puede observar que, tiene una buena respuesta, la cual es muy veloz, es fácil de ajustar su sensibilidad y se puede aumentar para eliminar temblores, es bastante cómodo y se puede adaptar a diferentes partes del cuerpo. Si el movimiento es muy lento puede mandar señales erróneas cuando se regresa al estado inicial, se debe hacer movimientos cortos si no puede mandar dos veces la señal de salida, también se debe ajustar el tiempo para evitar mantener la señal por mucho tiempo y es importante conectar bien el sensor a la caja, dado a que se pausa el módulo si no tiene una buena conexión. Si se descompone el sensor este puede ser intercambiado fácilmente, además de que permite una conexión sencilla al SAAC.

Es un módulo recomendable, que abarca todo el cuerpo y en caso de no tener movilidad se recomienda el módulo de activación por soplo.

Conclusiones

Para todos los módulos propuestos, se logró generar la señal requerida para la interacción con el SAAC, el cual trabaja sobre una microcomputadora Raspberry Pi, y que debe ser mediante una lógica inversa con un voltaje de 2 a 3.3 volts para niveles altos, y de máximo 0.8 volts para niveles bajos. Para los módulos que utilizan un microcontrolador, fue necesario incorporar un divisor de voltaje. Para los módulos con interruptores, no fue necesario hacer un ajuste especial, ya que la polarización, se tomó de la terminal de 3.3 volts de la misma Raspberry.

Se desarrollaron cinco módulos de interacción para el SAAC (algunos de ellos con variantes), siempre teniendo en cuenta un costo asequible y sin comprometer la calidad de los componentes y su funcionamiento.

Se consideraron diferentes discapacidades que podrían necesitar un SAAC, y a partir de estas, se propusieron diseños de módulos basados en interruptores, que podrán ser utilizados por pacientes con diferentes grados de control en sus manos. Para acoplar los módulos con interruptores al cuerpo, se crearon diseños en 3D y se materializaron con una impresora de fabricación aditiva.

Se desarrolló un módulo con dos variantes, basados en el flujo de aire del tracto vocal, detectando un soplo, y produciendo una señal para la interacción con el SAAC. Una de las variantes de este módulo, cuenta con un tubo con una boquilla para recibir el soplo; el otro, en su lugar incorpora una diadema con micrófono. En ambos, se agregó un control de duración de la señal que se envía al SAAC, otro para la sensibilidad de intensidad de soplo y uno más para el ajuste de contraste de una diminuta pantalla LCD que muestra el valor de los ajustes mencionados, los cuales se realizan mediante perillas. Estos módulos utilizan un microcontrolador, y se diseñaron modelos 3D para el montaje y protección de la circuitería.

Se creó un módulo basado en un acelerómetro/giroscopio, en el cual se utilizó un microcontrolador para el procesamiento de señales de posición y movimiento del sensor, así como para la gestión de los ajustes de usuario que son: control de sensibilidad, tiempo y contraste de un display. Para acoplarlo a diversas extremidades corporales, se crearon diversas soluciones de montaje modeladas en 3D.

Para finalizar, se puede considerar que, con los módulos creados, pacientes con una gran diversidad de limitaciones motrices, podrán hacer uso del SAAC, eligiendo alguna de las soluciones propuestas, las cuales fueron creadas utilizando elementos de bajo costo, y por lo tanto, se logra mantener la esencia primordial del proyecto, que es brindar ayuda a personas sin importar sus limitaciones en términos de nivel de motricidad así como económicas.

Trabajo a futuro

Aunque los objetivos planteados en este trabajo han sido cubiertos, aún es posible realizar mejoras y adición de funcionalidad al SAAC, que permitan un mejor desempeño a los módulos ya creados.

Para lograr identificar las posibles mejoras a realizar en los módulos, es necesario hacer una evaluación de funcionamiento, en un entorno con personas en las condiciones descritas para este proyecto, donde se identificarían las posibles adaptaciones para un mejor acoplamiento al cuerpo, si algún módulo no cubre sus propuestas de uso y si falta alguna adaptación por cubrir. Dado a las condiciones que impone la pandemia en las que se desarrolló este proyecto, no fue posible hacer estas pruebas, pero se espera que un futuro se pueda realizar.

Para el módulo con acelerómetro un trabajo que mejoraría el funcionamiento en la programación, sería que el acelerómetro no solo detectara el movimiento hacia cualquier dirección, sino que también, detectara tanto la ida como la vuelta como un solo movimiento, o algún gesto específico que al realizarlo, activase la señal de salida.

Para la circuitería, se lograría una mejor presentación, produciendo las tarjetas de PCB con algún proceso profesional, esto es, mandarlas realizar con un fabricante especializado, o adquiriendo un equipo para ese propósito. Los circuitos tendrían componentes con soldadura superficial y conectores de mayor calidad para los componentes externos a la PCB. En este sentido de igual manera, una mejora a realizar es contar con el apoyo de un diseñador industrial, para retrabajar los diseños y dejarlos listos para su producción a volumen y no una impresora 3D, lo cual mejoraría su ergonomía, aspecto, durabilidad y potencialmente su costo.

Otra mejora a realizar, sería evaluar si algún interruptor o sensor es mejor que los que están implementados en los módulos, y cambiarlos de ser necesario, por aquellos que tengan una mejor capacidad de respuesta, mejora en el tacto o sensibilidad y calidad del componente.

Otro trabajo a realizar, es indagar para el prototipo de parpadeo que fue descartado, si de alguna manera se puede desarrollar, aunque este cubierto su funcionalidad con algún otro módulo ya realizado.

Y por último, aún se pueden explorar alternativas que aunque actualmente pueden ser costosas, quizá con trabajo pueda encontrarse la manera de disminuir su costo para la incorporación al SAAC, tal es el caso de los sistemas que hacen lectura de ondas cerebrales.

Referencias

- AI-Heeti, A. (16 de Enero de 2021). *Accessibility devices at CES 2021 reflect growing focus on inclusive tech*. Obtenido de cnet: <https://www.cnet.com/health/accessibility-devices-at-ces-2021-reflect-growing-focus-on-inclusive-tech/>
- allen, D. A. (1988). Communication disorders of preschool children: the physician's responsibility. *Developmental and Behavioral Pediatrics*, 9, 164-170.
- Arteo, Ó. T. (2013). Arduino Curso Práctico de Formación. En Ó. T. Arteo, *Arduino Curso Práctico de Formación* (pág. 148). México D.F.: Alfaomega. Obtenido de http://cienciasvirtuales.com/wp-content/uploads/2016/11/Arduino_Curso_Practico_de_Formacion.pdf
- Asistronic. (12 de Marzo de 2018). *Asistronic*. Obtenido de Comunicación Asistida por Seguimiento Ocular: <https://www.asistronic.com/comunicacion-asistida-por-escaneo-ocular/?v=42983b05e2f2>
- Autodesk. (1 de Enero de 2020). *AUTODESK*. Obtenido de Modelado 3D: <https://www.autodesk.mx/solutions/3d-modeling-software>
- AUTODESK. (1 de Enero de 2020). *Diseño de PCB*. Obtenido de AUTODESK: <https://www.autodesk.mx/solutions/pcb-design-software>
- AUTODESK. (1 de Enero de 2021). *AUTODESK*. Obtenido de AutoCAD: <https://www.autodesk.es/products/autocad/overview?term=1-YEAR&tab=subscription>
- Banco Mundial. (1 de enero de 2021). *Banco Mundial*. Obtenido de Discapacidad: <https://www.bancomundial.org/es/topic/disability#1>
- Búa, M. T. (30 de Abril de 2014). *Edu Xunta de Galicia*. Obtenido de Elementos de maniobra y control.: https://www.edu.xunta.gal/espazoAbalar/sites/espazoAbalar/files/datos/1464947843/contido/314_elementos_de_maniobra_y_control.html
- Busca Palabra. (- de - de 2021). *Busca Palabra*. Obtenido de Definición de interruptores: <https://www.buscapalabra.com/definiciones.html?palabra=interruptores>
- Castellanos, C. (2005). Principios de la Electronica. *Instituto Radiofonico Fe y Alegria*, 29.
- Castro, R. (28 de Agosto de 2020). *wikiversus*. Obtenido de Acelerómetro Vs. Giroscopio - ¿cuáles son las diferencias?: <https://www.wikiversus.com/moviles/acelerometro-vs-giroscopio/>
- Comunicaciones, S. d. (1 de enero de 2012). *Argentina.gob.ar*. Obtenido de Comunicación Digital Inclusiva: https://seticcdi.enacom.gob.ar/descarga/index_descarga_pictograma.html
- Darkcrizt DesdeLinux. (1 de Enero de 2019). *DesdeLinux*. Obtenido de Ultimaker Cura: una excelente aplicación para preparar modelos para impresión 3D: <https://blog.desdelinux.net/ultimaker-cura-una-excelente-aplicacion-para-preparar-modelos-para-impresion-3d/#comments>

- descubrearduino.com. (3 de Julio de 2021). *descubrearduino.com*. Obtenido de MPU6050, Diagrama de pines, circuito y conexión con Arduino: <https://descubrearduino.com/mpu6050/>
- EAGLE. (1 de Enero de 2021). *AUTODESK*. Obtenido de EAGLE: <https://www.autodesk.com/products/eagle/overview#pcb-layout>
- Electgpl. (1 de Junio de 2016). *Electgpl*. Obtenido de EL Atmega328p: <http://electgpl.blogspot.com/2016/06/el-atmega328p.html>
- E-Marmolejo, R. (1 de Diciembre de 2017). *hetpro*. Obtenido de Microcontrolador – qué es y para que sirve: <https://hetpro-store.com/TUTORIALES/microcontrolador/>
- Eva, R. d. (2 de Agosto de 2020). *Xataka*. Obtenido de Qué modelo de Raspberry Pi comprar: un repaso a las principales placas y los proyectos más habituales para dar con la mejor: <https://www.xataka.com/seleccion/que-modelo-raspberry-pi-comprar-repaso-a-principales-placas-proyectos-habituales-para-dar-mejor>
- EVA, R. d. (22 de Septiembre de 2021). *Xataka*. Obtenido de Empezar con Arduino: qué placa y kits de iniciación comprar: <https://www.xataka.com/makers/empezar-arduino-que-placa-kits-iniciacion-comprar>
- Fernández, Y. (3 de Agosto de 2020). *Xataka*. Obtenido de Qué es Arduino, cómo funciona y qué puedes hacer con uno: <https://www.xataka.com/basics/que-arduino-como-funciona-que-puedes-hacer-uno>
- Francisco Estrada Rojo, J. M. (2012). Neuroproteccion y traumatismo Craneoencefálico. *Revista de la Facultad de Medicina (México)*, 14.
- Galván, C. d. (2015). *Habilidades de comunicación y promoción de conductas adaptadas de la persona con discapacidad*. España: Ediciones Paraninfo.
- García, V. (9 de Noviembre de 2010). *EPA Electrónica Práctica Aplicada*. Obtenido de Al Amplificador Operacional: <https://www.diarioelectronicohoy.com/blog/el-amplificador-operacional>
- González, P. M. (01 de 01 de 2003). *SISTEMAS ALTERNATIVOS Y AUMENTATIVOS DE COMUNICACION (SAAC) Y ACCESIBILIDAD*. (P. a. Lectura, Ed.) Recuperado el 13 de 05 de 2021, de <https://dialnet.unirioja.es/descarga/articulo/6023011.pdf>
- Inclusive Technology. (1 de enero de 2021). *Inclusive Technology*. Obtenido de Quha Zono Mouse: <http://www.inclusive.co.uk/quha-zono-mouse>
- Inclusive Technology. (1 de enero de 2021). *Inclusive Technology All the help you need!* Obtenido de IntegraMouse Plus: <http://www.inclusive.co.uk/integramouse-plus-p5725>
- Isaac. (1 de Enero de 2018). *Hardware Libre*. Obtenido de GPIO: todo sobre las conexiones de la Raspberry Pi 4 y 3: <https://www.hwlibre.com/gpio-raspberry-pi/>
- Lic. Falcón, V. C. (Junio de 2011). *EFDeportes.com*. (Revista Digital. Buenos Aires) Recuperado el 16 de 07 de 2021, de <https://www.efdeportes.com/efd157/sistema-alternativo-de-comunicacion.htm>

- Luis, E. R. (15 de Julio de 2018). *Xataka*. Obtenido de De cero a maker: todo lo necesario para empezar con Raspberry Pi: <https://www.xataka.com/makers/cero-maker-todo-necesario-para-empezar-raspberry-pi>
- M. Fried-Oken, C. R. (2013). Mobile Technology as Communication Supports for Adult with Primary Progressive Aphasia. *Oregon Health & Science University*, 40. Obtenido de http://aac-lerc.psu.edu/documents/Fried-Oken_mobile_computing_PPA_ATIA_2013_HO.pdf
- Martínez, I. O. (2020). *Sistema computacional aumentativo y alternativo de comunicacion como herramienta de apoyo a personas con discapacidad motora y del habla*. Estado de México, Ecatepec de Morelos: FES Aragon UNAM.
- Matters, C. (01 de 01 de 2020). *Communication Matters*. (National Communication Association) Recuperado el 13 de 05 de 2021, de <https://www.communicationmatters.org.uk/>
- Microchip. (1 de Enero de 2021). *Microchip*. Obtenido de ATmega328P: <https://www.microchip.com/en-us/product/ATmega328P#>
- Miyara, F. (2006). *Acústica y sistemas de sonido*. Argentina, Buenos Aires: Editorial de la Universidad Nacional de Rosario Urquiza.
- Naylamp mechatronics. (05 de Mayo de 2016). *Naylamp mechatronics*. Obtenido de TUTORIAL MPU6050, ACELERÓMETRO Y GIROSCOPIO: https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html
- naylamp mechatronics. (5 de Mayo de 2016). *TUTORIAL MPU6050, ACELERÓMETRO Y GIROSCOPIO*. Obtenido de naylamp mechatronics: https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html
- Porras-Betancourt M, N. O.-Á.-A.-S. (2007). Esclerosis múltiple. *Revista Mexicana de Neurociencia*, 10. Obtenido de <http://previous.revexneurociencia.com/wp-content/uploads/2014/06/Nm071-10.pdf>
- Proteus. (1 de Enero de 2021). *Labcenter Electronics*. Obtenido de PCB Layout Features: <https://www.labcenter.com/pcb-layout/>
- Proteus. (1 de Enero de 2021). *Proteus*. Obtenido de PCB Desing & Simulation Made Easy: <https://www.labcenter.com/>
- ptolomeo. (- de - de Desconocido). *unam*. Obtenido de El Microcontrolador: <http://www.ptolomeo.unam.mx:8080/jspui/bitstream/132.248.52.100/760/4/A4.pdf>
- Sanabria, M. F. (2010). Esclerrosis Lateral Amiotrofica. *Revista Medica de Costa Rica y Centroamerica LXVII(591)*, 4.
- Shelat, A. M. (6 de 23 de 2020). *MedlinePlus*. Obtenido de Deterioro del lenguaje en adultos: <https://medlineplus.gov/spanish/ency/article/003204.htm>
- SOLIDWORKS. (1 de Enero de 2021). *SOLIDWORKS*. Obtenido de Diseño/Ingenieria: <https://www.solidworks.com/es/domain/design-engineering>

Texas Instruments. (1 de Mayo de 2017). *Texas Instruments*. Obtenido de LM386 Low Voltage Audio Power Amplifier: https://www.ti.com/lit/ds/symlink/lm386.pdf?ts=1636654517735&ref_url=https%253A%252F%252Fwww.google.com.mx%252F

TOSCANO CASTILLO, S. (Junio de 2016). *Lenguaje y parálisis cerebral: el uso de los SAAC como medio de comunicación*. Recuperado el 19 de Julio de 2021, de Colibri: <https://www.colibri.udelar.edu.uy/jspui/handle/20.500.12008/8517>

VALDEZ, J. M. (2007). PARALISIS CEREBRAL. *MEDICINA, ACTUALIZACIONES EN NEUROLOGIA INFANTIL*, 7. Obtenido de <http://ardilladigital.com/DOCUMENTOS/DISCAPACIDADES/MOTORA/Paralisis%20Cerebral%20-%20Malagon%20-%20articulo.pdf>

worldometer. (1 de 1 de 2021). *worldometer*. Obtenido de Poblacion mundial anual: [https://www.worldometers.info/es/poblacion-mundial/#:~:text=Poblaci%C3%B3n%20Mundial%3A%207.8%20Billones%20de%20Personas%20\(2021\)%20%2D%20Worldometer](https://www.worldometers.info/es/poblacion-mundial/#:~:text=Poblaci%C3%B3n%20Mundial%3A%207.8%20Billones%20de%20Personas%20(2021)%20%2D%20Worldometer)

Fuentes de Imágenes

- ◆ Ilustración 2.1 Clasificación de los Sistemas de Comunicación Recuperada el 16 de Julio del 2021 de:

A.P. Sofía Aguirre, A.P. Héctor Daniel Pérez, SAAC-Droid: una herramienta de ayuda a la comunicación, General Pico, 2012, pág. 3.

- ◆ Ilustración 2.2 Comunicadores de Voz Digital Recuperada el 16 de Julio del 2021 de:

P. M. González, «SISTEMAS ALTERNATIVOS Y AUMENTATIVOS DE COMUNICACION (SAAC) Y ACCESIBILIDAD,» 01 01 2003. [En línea]. Available: <https://dialnet.unirioja.es/descarga/articulo/6023011.pdf>. [Último acceso: 13 05 2021].

- ◆ Ilustración 2.3 Comunicadores de voz Sintetizada Recuperada el 16 de Julio del 2021 de:

P. M. González, «SISTEMAS ALTERNATIVOS Y AUMENTATIVOS DE COMUNICACION (SAAC) Y ACCESIBILIDAD,» 01 01 2003. [En línea]. Available: <https://dialnet.unirioja.es/descarga/articulo/6023011.pdf>. [Último acceso: 13 05 2021].

- ◆ Ilustración 2.4 SAAC Digital de Ordenador Recuperada el 16 de Julio del 2021 de: CRE Discapacidad y Dependencia de San Andrés del Rabanedo León. (2021, 07 19). *Sistemas de Comunicación Alternativos y Aumentativos.mp4*. Recuperado de: https://www.youtube.com/watch?v=NmAeSlejnzY&t=4s&ab_channel=crediscapacidad

- ◆ Ilustración 2.5 Stephen Hawking Recuperada el 11 de noviembre del 2021 de: ELA (Esclerosis lateral amiotrófica) | Neurología del Sureste Recuperado de:

neurologomerida.com

◆ Ilustración 2.6 Francisco Estrada Rojo, J. M. Recuperada el 12 de Noviembre del 2021 de Neuroproteccion y traumatismo Craneoencefálico. *Revista de la Facultad de Medicina (México)*, 14.(2012)Recuperado de : <http://www.scielo.org.mx/pdf/facmed/v55n4/v55n4a4.pdf>

◆ Ilustración 2.7 Néstor Pardo Recuperada el 12 de noviembre del 2021 de Parálisis cerebral infantil y modalidades aumentativas – alternativas para la comunicación (1/12/2017) Recuperado de:

<https://www.espaciologopedico.com/revista/articulo/166/paralisis-cerebral-infantil-y-modalidades-aumentativas--alternativas-para-la-comunicacion.html>

◆ Ilustración 2.8 Daniel Oscar Rodríguez Boggia Recuperada el 12 de noviembre del 2021 de SAAC. Sistemas alternativos y aumentativos de comunicación (Parte III) (1/12/2017) Recuperado de:

<https://www.espaciologopedico.com/revista/articulo/3368/saac-sistemas-alternativos-y-aumentativos-de-comunicacion-parte-iii.html>

◆ Ilustración 2.9 Pulsadores y Conmutadores Recuperada el 8 de noviembre del 2021 de:

P. M. González, «SISTEMAS ALTERNATIVOS Y AUMENTATIVOS DE COMUNICACION (SAAC) Y ACCESIBILIDAD,» 01 01 2003. [En línea]. Available: <https://dialnet.unirioja.es/descarga/articulo/6023011.pdf>. [Último acceso: 13 05 2021].

◆ Ilustración 2.10 Modelos de simulador de ratón Recuperada el 8 de noviembre del 2021 de:

P. M. González, «SISTEMAS ALTERNATIVOS Y AUMENTATIVOS DE COMUNICACION (SAAC) Y ACCESIBILIDAD,» 01 01 2003. [En línea]. Available: <https://dialnet.unirioja.es/descarga/articulo/6023011.pdf>. [Último acceso: 13 05 2021].

◆ Ilustración 2.11 Modelos de simulador de ratón Recuperada el 8 de noviembre del 2021 de:

P. M. González, «SISTEMAS ALTERNATIVOS Y AUMENTATIVOS DE COMUNICACION (SAAC) Y ACCESIBILIDAD,» 01 01 2003. [En línea]. Available: <https://dialnet.unirioja.es/descarga/articulo/6023011.pdf>. [Último acceso: 13 05 2021].

◆ Ilustración 2.12 Modelos de ratón adaptados y de bola Recuperada el 8 de noviembre del 2021 de:

P. M. González, «SISTEMAS ALTERNATIVOS Y AUMENTATIVOS DE COMUNICACION (SAAC) Y ACCESIBILIDAD,» 01 01 2003. [En línea]. Available: <https://dialnet.unirioja.es/descarga/articulo/6023011.pdf>. [Último acceso: 13 05 2021].

◆ Ilustración 2.13 Comunicadores de voz digitalizada Recuperada el 8 de noviembre del 2021 de:

P. M. González, «SISTEMAS ALTERNATIVOS Y AUMENTATIVOS DE COMUNICACION (SAAC) Y ACCESIBILIDAD,» 01 01 2003. [En línea]. Available: <https://dialnet.unirioja.es/descarga/articulo/6023011.pdf>. [Último acceso: 13 05 2021].

- ◆ Ilustración 2.14 Comunicadores de voz sintetizada Recuperada el 8 de noviembre del 2021 de:
P. M. González, «SISTEMAS ALTERNATIVOS Y AUMENTATIVOS DE COMUNICACION (SAAC) Y ACCESIBILIDAD,» 01 01 2003. [En línea]. Available: <https://dialnet.unirioja.es/descarga/articulo/6023011.pdf>. [Último acceso: 13 05 2021].
- ◆ Ilustración 2.15 Modelos de software para interacción en un ordenador Recuperada el 8 de noviembre del 2021 de:
P. M. González, «SISTEMAS ALTERNATIVOS Y AUMENTATIVOS DE COMUNICACION (SAAC) Y ACCESIBILIDAD,» 01 01 2003. [En línea]. Available: <https://dialnet.unirioja.es/descarga/articulo/6023011.pdf>. [Último acceso: 13 05 2021].
- ◆ Ilustración 2.16 LetMeTalk aplicación Recuperada el 8 de noviembre del 2021 de: Informática para Educación Especial. Available en: <http://informaticaparaeducacionespecial.blogspot.com/p/letmetalk.html>
- ◆ Ilustración 2.17 SAAC con Cámara Recuperada el 8 de noviembre del 2021 de: Asistrinic página web. Available en: <https://www.asistrinic.com/comunicacion-asistida-por-escaneo-ocular/?v=42983b05e2f2>
- ◆ Ilustración 2.18 Modelos Raspberry Pi Recuperado el 8 de noviembre del 2021 de: Kolwidi, ¿Qué es una Raspberry Pi? por Loli Diéguez de: <https://kolwidi.com/blogs/blog-kolwidi/que-es-una-Raspberry-pi>
- ◆ Ilustración 2.19 Pines de la Raspberry Pi 4 Recuperado el 8 de noviembre del 2021 de: HardwareLibre, GPIO: todo sobre las conexiones de la Raspberry Pi 4 y 3, Issac de: <https://www.hwlibre.com/gpio-Raspberry-pi/>
- ◆ Ilustración 2.20 Arduino UNO Recuperado el 8 de noviembre del 2021 de: Xataka Empezar con Arduino: qué placa y kits de iniciación comprar, EVA de: <https://www.xataka.com/makers/empezar-arduino-que-placa-kits-iniciacion-comprar>
- ◆ Ilustración 2.21 Arduino IDE Recuperado el 8 de noviembre del 2021 de: ViaSoluciones, Arduino, Alejandro de: <https://sites.google.com/site/xtalejandrovaldivia/p/proyecto-4>
- ◆ Ilustración 2.22 Esquema de Von Neumann Recuperado el 9 de noviembre del 2021 de: potolomeo unam de: <http://www.ptolomeo.unam.mx:8080/jspui/bitstream/132.248.52.100/760/4/A4.pdf>
- ◆ Ilustración 2.23 Esquema de la Arquitectura Harvard Recuperado el 9 de noviembre del 2021 de: potolomeo unam de: <http://www.ptolomeo.unam.mx:8080/jspui/bitstream/132.248.52.100/760/4/A4.pdf>
- ◆ Ilustración 2.24 ATmega328p Recuperado el 9 de noviembre del 2021 de: Atmga328P, Electgpl (2016) de: <http://electgpl.blogspot.com/2016/06/el-ATmega328p.html>
- ◆ Ilustración 2.25 Micrófono Dinámico Recuperado el 9 de noviembre del 2021 de: Acústica y Sistemas de Sonido, Miraya (2016). De: MIYARA_F_Acustica_y_Sistemas_de_Sonido_Capítulos_1_y_2.pdf (untrefvirtual.edu.ar) pág. 93.

- ◆ Ilustración 2.26 Micrófono Capacitivo Recuperado el 9 de noviembre del 2021 de: Acústica y Sistemas de Sonido, Miraya (2016). De: MIYARA_F_Acustica_y_Sistemas_de_Sonido_Capítulos_1_y_2.pdf (untrefvirtual.edu.ar) pág. 94.
- ◆ Ilustración 2.27 Modelo ideal del Amplificador Operacional Recuperado el 9 de noviembre del 2021 de EPA Electrónica Practica Aplicada, Vicente, (2010), de: <https://www.diarioelectronico hoy.com/blog/el-amplificador-operacional>
- ◆ Ilustración 2.28 LM386 Recuperada el 9 de noviembre del 2021 de: Texas Instruments Datasheet (2017) en: https://www.ti.com/lit/ds/symlink/lm386.pdf?ts=1636654517735&ref_url=https%253A%252F%252Fwww.google.com.mx%252F
- ◆ Ilustración 2.29 LM386 con ganancia de 20 Recuperado el 9 de noviembre de: lifetechnology de: https://lifetechnology.mercadoshops.com.mx/MLM-852133732-lm386-amplificador-de-audio-de-baja-potencia-10-piezas-_JM
- ◆ Ilustración 2.30 Resistencia Recuperado el 9 de noviembre del 2021 de: Nota tecnología, de: <https://notatecnologica.com/dispositivos/resistencia-electrica/>
- ◆ Ilustración 2.31 Diferentes tipos de Capacitores Recuperado el 9 de noviembre del 2021 de: Capacitación Técnica (2020) Nota tecnología, de: <https://capacitaciontecnica.com.ar/2020/04/17/que-son-los-capacitores/>
- ◆ Ilustración 2.32 Tipos de Inductores Recuperado el 9 de noviembre del 2021 de: FIEK, Wikipedia (2011) Nota tecnología, de: https://es.wikipedia.org/wiki/Inductor#/media/Archivo:Aplikimi_i_feriteve.png
- ◆ Ilustración 2.33 Proteus Recuperado el 9 de noviembre del 2021 de: Protesus PCB desing layout (2021) de: <https://www.labcenter.com/pcb-layout/>
- ◆ Ilustración 2.34 EALGE Recuperado el 9 de noviembre del 2021 de: EALGE, Autodesk (2021) de: <https://www.autodesk.com/products/eagle/overview>
- ◆ Ilustración 2.35 Modelado en SolidWorks Recuperado el 10 de noviembre del 2021 de: Novedades en SolidWorks CAM 2020 de: https://blogs.solidworks.com/solidworkslatamyesp/solidworks-blog/solidworks_cam/novedades-en-solidworks-cam-2020/
- ◆ Ilustración 2.36 Interruptor Recuperado el 10 de noviembre del 2021 en: leantec, interruptor de encendido/apagado de: <https://leantec.es/tienda/interruptor-de-encendido-apagado-ac-250-v-3a-2-pin-on-off-i-o-spst-snap/>
- ◆ Ilustración 2.37 Pulsador Recuperado el 10 de noviembre del 2021 en: iberorobotics pulsador 6x6mm en: <https://www.iberorobotics.com/producto/pulsador-6x6mm-mini-pack-5-unidades/>
- ◆ Ilustración 2.38 Conmutador Recuperado el 10 de noviembre del 2021 en: aliexpress interruptor de: <https://es.aliexpress.com/item/32736698358.html>

- ◆ Ilustración 2.39 Microinterruptor Recuperado el 10 de noviembre del 2021 en: Amazon, houseuse de: <https://www.amazon.com/-/es/Houseuse-interruptores-palanca-microinterruptores-color/dp/B07696LBJG>
- ◆ Ilustración 2.40 Relé Recuperado el 10 de noviembre del 2021 en: BricoGeek, Relé de: <https://tienda.bricogeek.com/componentes/260-rele-5v-spdt.html>
- ◆ Ilustración 2.41 Giroscopio Recuperado el 10 de noviembre del 2021 de: wikiversus, Rubén Castro (2020) de: <https://www.wikiversus.com/moviles/acelerometro-vs-giroscopio/>
- ◆ Ilustración 2.42 Funcionamiento de un Acelerómetro Recuperado el 10 de noviembre del 2021 de: wikiversus, Rubén Castro (2020) de: <https://www.wikiversus.com/moviles/acelerometro-vs-giroscopio/>
- ◆ Ilustración 2.43 MPU6050 Recuperada el 10 de noviembre del 2021 de: naylamp mechatronics en: https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html
- ◆ Tabla 2.44 Puertos del MPU6050 Recuperada el 10 de noviembre del 2021 de: descubrearduino en: <https://descubrearduino.com/mpu6050/>

Fuentes de tablas

- ◆ Tabla 2.1 Porras-Betancourt M, N. O.-Á.-A.-S. Recuperada el 11 de Noviembre del 2021 de: (2007). Esclerosis múltiple. *Revista Mexicana de Neurociencia*, 10. Obtenido de <http://previous.revmexneurociencia.com/wp-content/uploads/2014/06/Nm071-10.pdf>
- ◆ Tabla 2.2 Tablas de Modelos Raspberry Pi Recuperado el 8 de noviembre del 2021 de: Xataka Qué modelo de Raspberry Pi comprar: un repaso a las principales placas y los proyectos más habituales para dar con la mejor por EVA de: <https://www.xataka.com/seleccion/que-modelo-Raspberry-pi-comprar-repaso-a-principales-placas-proyectos-habituales-para-dar-mejor>
- ◆ Tabla 2.3 Características ATmega328P Recuperado el 9 de noviembre del 2021 de Atmega328P, electgpl (2016) de: https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html

Anexo

A. Código del módulo basado en sople (Dos variantes).

```
1  #include <LiquidCrystal.h>
2  LiquidCrystal lcd(7,6,5,4,3,2);
3
4  int sensibilidad=0;
5  int tiempo=0;
6
7  void setup()
8  {
9      lcd.begin(16, 2);
10     pinMode(1, OUTPUT);
11     pinMode(8, OUTPUT);
12     digitalWrite(1, HIGH);
13     digitalWrite(8, HIGH);
14 }
15
16 void loop()
17 {
18     int s = analogRead(A1); //establecer valor de referencia (0-1023)
19     int t = analogRead(A2);
20     int tLED=map(t,0,1023,300,3000);
21
22     if((s>=sensibilidad+10) || (s<=sensibilidad-10) || (t>=tiempo+10)
|| (t<=tiempo-10))
23     {
24         sensibilidad=s;
25         tiempo=t;
26         actualizarLCD();
27     }
28
29     if (s<=analogRead(A0))
30     {
31         digitalWrite(1, LOW);
32         digitalWrite(8, LOW);
33         delay(tLED);
34         digitalWrite(1,HIGH);
35         digitalWrite(8,HIGH);
36     }
37 }
38
39 void actualizarLCD()
40 {
41     int tiempoLCD = map(tiempo,0,1023,300,3000);
42
43     int sensibilidadLCD = map(sensibilidad,0,1023,0,99);
44
45     lcd.setCursor(0,1);
46     lcd.print("Tiempo:      ");
47     lcd.setCursor(9,1);
48     lcd.print(tiempoLCD);
49
50     delay(10);
```

```

51
52     lcd.setCursor(0,0);
53     lcd.print("Sensibilidad:  ");
54     lcd.setCursor(14,0);
55     lcd.print(sensibilidadLCD);
56
57     delay(10);
58
59 }
60

```

B. Código del módulo con acelerómetro.

```

1     #include "I2Cdev.h"
2     #include "MPU6050.h"
3     #include "Wire.h"
4     #include "LiquidCrystal.h"
5
6     MPU6050 mpu(0x68);
7     LiquidCrystal lcd(7,6,5,4,3,2);
8
9     int ax, ay, az;
10    unsigned long tempRef1;
11    float valor=0;
12    int sensibilidad;
13    int tiempo;
14    int s;
15    int t;
16
17
18    void setup()
19    {
20        lcd.begin(16, 2);
21        lcd.home();
22        pinMode(9,1); //salida status mcu
23        pinMode(8,1); //replica respuesta led
24        pinMode(10,1); //salida respuesta Señal
25        digitalWrite(10,1);
26        Wire.begin();
27        s=analogRead(A0); //Lectura del Potenciómetro para la
                sensibilidad
28        t=analogRead(A1); //Lectura del Potenciómetro para el Tiempo
29        sensibilidad=map(s,0,1023,0,400);
30        tiempo=map(t,0,1023,100,2000);
31        actualizarLCD();
32
33        mpu.initialize();
34        if (mpu.testConnection())
35        {
36            digitalWrite(9,1);
37        }
38        else
39            errorMcu();
40    }
41
42    void loop()

```

```

43 {
44     int s1=analogRead(A0); //Lectura del Potenciometro para la
sensibilidad
45     int t1=analogRead(A1); //Lectura del Potenciometro para el
Tiempo
46
47     if((s1>=s+10) || (s1<=s-10)|| (t1>=t+10)|| (t1<=t-10))
48     {
49         sensibilidad=map(s1,0,1023,0,400);
50         tiempo=map(t1,0,1023,100,2000);
51         s=s1;
52         t=t1;
53         actualizarLCD();
54     }
55
56     if (mpu.testConnection())
57     {
58         medicion();
59         if(valor>(sensibilidad*20))
60         {
61             digitalWrite(8,1);
62             digitalWrite(10,0);
63             delay(tiempo);
64             digitalWrite(8,0);
65             digitalWrite(10,1);
66             delay(tiempo/2);
67         }
68     }
69     else
70     errorMcu();
71 }
72
73
74 void medicion()
75 {
76     valor=0;
77     tempRef1=millis();
78     while((millis()-tempRef1)<250)
79     {
80         mpu.getAcceleration(&ax, &ay, &az);
81         int ax1=ax;
82         int ay1=ay;
83         int az1=az;
84         delay(20);
85
86         mpu.getAcceleration(&ax, &ay, &az);
87         int ax2=ax;
88         int ay2=ay;
89         int az2=az;
90         delay(20);
91
92         float mov =sqrt((pow((ax1-ax2),2))+ (pow((ay1-
ay2),2))+ (pow((az1-az2),2)));
93
94         valor=valor+mov;
95     }
96 }

```

```

97
98 void errorMcu()
99 {
100     while(!mpu.testConnection())
101     {
102         digitalWrite(9,1);
103         delay(500);
104         digitalWrite(9,0);
105         delay(500);
106     }
107     digitalWrite(9,1);
108 }
109
110 void actualizarLCD()
111 {
112
113     lcd.setCursor(0,1);
114     lcd.print("Tiempo:      ");
115     lcd.setCursor(9,1);
116     lcd.print(tiempo);
117
118     delay(10);
119
120     lcd.setCursor(0,0);
121     lcd.print("Sensibilidad:  ");
122     lcd.setCursor(13,0);
123     lcd.print(sensibilidad);
124
125     delay(10);
126
127 }
128

```

C. Lista de materiales y costos.

| Cantidad | Concepto | Valor unitario M.N. [\$] | Total [\$] |
|---|----------------------------------|-----------------------------|---------------|
| Módulo para palma -muñeca de la mano | | | |
| 1 | Botón Grande | 100 | 100 |
| 3 | Metros de alambre | 3.5 | 10.5 |
| 1 | Rollo PLA (aproximadamente 500g) | 250 | 250 |
| 10 | Tornillos | 0.5 | 5 |
| 1 | Termofit negro (20 cm) | 8 | 8 |

| | | | |
|--|-------------------------------------|-----|-------|
| 1 | Conector hembra | 10 | 10 |
| Total, M | Modulo para palma-muñeca de la mano | | 383.5 |
| Módulo para activación por pulgar de la mano V1 | | | |
| 1 | Botón mediano | 9 | 9 |
| 1 | Metros de resorte | 10 | 10 |
| 1 | Rollo de PLA (150g) | 75 | 75 |
| 2 | Metros de alambre | 3.5 | 7 |
| 1 | Termofit negro (20 cm) | 8 | 8 |
| 1 | Conector hembra | 10 | 10 |
| Total, M | Módulo de activación por pulga V1 | | 119 |
| Módulo para activación por pulgar de la mano V2 | | | |
| 2 | Botón con interruptor | 10 | 20 |
| 1 | Metros de resorte | 10 | 10 |
| 1 | Rollo de PLA (200g) | 100 | 100 |
| 3 | Metros de alambre | 3.5 | 10.5 |
| 1 | Termofit negro (20 cm) | 8 | 8 |
| 3 | Conector hembra | 10 | 10 |
| Total, M | Módulo de activación por pulga V2 | | 158.5 |
| Módulo para activación por agarre de mano | | | |
| 2 | Interruptores | 7 | 14 |

| | | | |
|--|------------------------------------|-----|-------|
| 2 | Resortes pequeños | 10 | 20 |
| 1 | Metros de resorte | 10 | 10 |
| 1 | Rollo de PLA (200g) | 100 | 100 |
| 3 | Metros de alambre | 3.5 | 10.5 |
| 1 | Termofit negro (20 cm) | 8 | 8 |
| 3 | Conector hembra | 10 | 10 |
| Total, M | Módulo de activación por agarre | | 172.5 |
| Módulo para activación por soplo V1 | | | |
| 1 | Placa fenólica | 30 | 30 |
| 3 | Tapas para potenciómetro | 10 | 30 |
| 1 | ATmega328P | 68 | 68 |
| 2 | Metros de manguera | 15 | 30 |
| 10 | Tornillos | 0.5 | 5 |
| 1 | LM386 | 16 | 16 |
| 1 | LCD16x2. | 59 | 59 |
| 1 | Cristal de 16 MHz. | 5 | 5 |
| 1 | Un micrófono elektret. | 40 | 40 |
| 3 | Potenciómetros de 100 K Ω . | 10 | 30 |
| 1 | Resistencias de 10 Ω . | 1 | 1 |
| 1 | Resistencia de 30 K Ω . | 1 | 1 |

| | | | |
|--|-----------------------------------|-----|-----|
| 2 | Capacitores cerámicos de 22 pF. | 5 | 10 |
| 1 | Pulsador. | 6 | 6 |
| 1 | Switch. | 10 | 10 |
| 2 | Capacitores de 0.1 uF. | 5 | 10 |
| 4 | Resistencias de 10 KΩ. | 1 | 4 |
| 2 | Resistencias de 220 Ω. | 1 | 2 |
| 1 | Molex de 3 pines. | 10 | 10 |
| 3 | Molex hembra | 10 | 30 |
| 5 | Metros de alambre | 3.5 | 35 |
| 1 | Termofit negro (20 cm) | 8 | 8 |
| 7 | Conector hembra | 10 | 10 |
| Total, M | Módulo de activación por soplo V1 | | 450 |
| Módulo para activación por soplo V2 | | | |
| 1 | Placa fenólica | 30 | 30 |
| 3 | Tapas para Potenciómetro | 10 | 30 |
| 1 | ATmega328P | 68 | 68 |
| 1 | Diadema con micrófono | 50 | 50 |
| 8 | Tornillos | 0.5 | 4 |
| 1 | LM386 | 16 | 16 |
| 1 | LCD16x2. | 59 | 59 |

| | | | |
|--------------------------------|------------------------------------|-----|-------|
| 1 | Cristal de 16 MHz. | 5 | 5 |
| 1 | Jack hembra de 3.5 mm | 15 | 15 |
| 3 | Potenciómetros de 100 K Ω . | 10 | 30 |
| 1 | Resistencias de 10 Ω . | 1 | 1 |
| 1 | Resistencia de 30 K Ω . | 1 | 1 |
| 2 | Capacitores cerámicos de 22 pF. | 5 | 10 |
| 1 | Pulsador. | 6 | 6 |
| 1 | Switch. | 10 | 10 |
| 2 | Capacitores de 0.1 uF. | 5 | 10 |
| 4 | Resistencias de 10 K Ω . | 1 | 4 |
| 2 | Resistencias de 220 Ω . | 1 | 2 |
| 1 | Molex de 3 pines. | 10 | 10 |
| 3 | Molex hembra | 10 | 30 |
| 5 | Metros de alambre | 3.5 | 17.5 |
| 1 | Termofit negro (20 cm) | 8 | 8 |
| 7 | Conector hembra | 10 | 70 |
| Total, M | Módulo de activación por soplo V2 | | 486.5 |
| Módulo con acelerómetro | | | |
| 1 | Placa fenólica | 30 | 30 |
| 1 | ATmega328P | 68 | 68 |

| | | | |
|----|------------------------------------|-----|-----|
| 1 | MPU6050 | 75 | 75 |
| 3 | Tapas para potenciómetro | 10 | 30 |
| 10 | Tornillos | 0.5 | 5 |
| 4 | Metros de Resorte | 10 | 40 |
| 1 | Rollo de PLA (250g) | 125 | 125 |
| 6 | Metros de alambre | 3.5 | 21 |
| 1 | Termofit negro (50 cm) | 8 | 8 |
| 10 | Conector hembra | 10 | 100 |
| 1 | LCD16x2. | 59 | 59 |
| 1 | Cristal de 16 MHz. | 5 | 5 |
| 3 | Potenciómetros de 100 K Ω . | 10 | 30 |
| 1 | Resistencias de 30 Ω . | 1 | 1 |
| 1 | Resistencia de 100 Ω . | 1 | 1 |
| 2 | Capacitores cerámicos de 22 pF. | 5 | 10 |
| 1 | Pushbutton. | 6 | 6 |
| 1 | Switch. | 10 | 10 |
| 1 | Resistencias de 10 K Ω . | 1 | 1 |
| 4 | Resistencias de 220 Ω . | 1 | 4 |
| 1 | Molex de 3 pines hembra. | 10 | 10 |
| 1 | Molex de 4 pines hembra. | 10 | 10 |

| | | | |
|----------|-------------------------------------|---|------|
| 2 | LED. | 2 | 4 |
| Total, M | Módulo con acelerómetro | | 653 |
| Total | Todos los componentes | | 2423 |
| Total | Total + componentes de construcción | - | 3123 |