

PROYECTO FINAL DE CARRERA

**PRUEBA DE CONCEPTO: DISEÑO Y DESARROLLO DE UN PROTOTIPO
DE COMUNICACIÓN BASADO EN EL MOVIMIENTO LINGUAL
PARA PERSONAS CON CUADRIPLEJIA**

Alumnos:

AWRUCH, FEDERICO EZEQUIEL - 57269

y

DONADU, TOMÁS EZEQUIEL - 57324

Tutor:

ESP. BIOING. PASCHETTA, FEDERICO

Asesor:

RÍOS ESQUIVEL, ESTEBAN ANTONIO



Departamento de Ciencias de la Vida
Instituto Tecnológico de Buenos Aires
Septiembre 2022

Índice general

	Página
1. Introducción	2
2. Marco Teórico	3
2.1. Anatomía de la lengua	3
2.2. Fisiopatología	5
3. Concepción	7
3.1. Alcance del trabajo	7
3.2. Dificultades y limitaciones	8
3.3. Características Distintivas	8
4. Objetivos	10
5. Diseño e implementación	11
5.1. Sensores	11
5.1.1. Definición de tecnología de sensado	11
5.1.2. Definición del sensor óptico	16
5.1.3. Interfaz de conexión: SPI	19
5.1.4. Características técnicas del sensor BB8520	22
Pinout	22
Operaciones	24
Registros	27
5.2. Comunicación	31
5.2.1. Definición del tipo de comunicación	31
5.2.2. Determinación del modulo Bluetooth	31
Perfiles	31

Módulos Bluetooth	33
Interfaz de conexión	34
5.2.3. Características técnicas del modulo RN42	35
5.3. Hardware	37
5.3.1. Definición del microcontrolador	37
5.3.2. Diseño y desarrollo de Placa de Prueba	39
Planificación del diseño	40
Definición del diseño	41
Desarrollo de la placa	42
Armado de placa complementaria	43
Sistema integrado de la tecnología	44
5.3.3. Diseño y desarrollo de Placa Final	45
Diseño de doble placa	46
Características adicionales	47
Desarrollo de ambas placas	48
5.3.4. Alimentación	50
5.4. Software	53
5.4.1. Configuración Inicial	55
5.4.2. Lazo	56
5.5. Gabinete	58
5.5.1. Definición del gabinete	58
5.5.2. Diseño de gabinete	60
Primera iteración	60
Segunda iteración	61
Tercer iteración	62
5.5.3. Impresión y ensamblado	63
6. Pruebas y resultados	65
6.1. Comunicación	65
6.2. Sensores	66
6.2.1. Trackball	67
6.2.2. Joystick	67

6.2.3. Botones	67
6.2.4. Pad Capacitivo	67
6.2.5. Trackpad	68
6.3. Trackpad en placa de prueba	68
6.4. Software	69
7. Discusión	70
7.1. Prototipo funcional ensamblado	70
7.1.1. Funcionamiento	70
7.1.2. Dificultades durante su uso	71
Sensor	71
Mordillo	72
7.2. Mejoras	72
7.2.1. Hardware	72
7.2.2. Software	73
7.2.3. Gabinete	74
7.3. Hitos del proyecto	75
7.3.1. Comunicación Bluetooth en perfil HID	75
7.3.2. Definición y prueba del sensor	75
7.3.3. Definición y finalización del mordillo	76
8. Conclusión	77
Referencias	81
Apéndices	
A. Diseño de placas en Altium	83

Índice de figuras

2.1. Grupos musculares de la lengua.	4
2.2. Anatomía de la medula espinal	5
2.3. Efectos de la lesión espinal	6
5.1. Funcionamiento de un sensor óptico de ratón. El LED emite luz visible o IR, esta atraviesa un primer lente que la dispersa e ilumina la superficie. El haz de luz rebota y atraviesa un segundo lente que converge en el sensor óptico.	16
5.2. Funcionamiento de un trackpad óptico. Se desliza el dedo sobre la superficie del mismo y los cambios detectados por la cámara del mismo son traducidos a movimientos de un cursos.	18
5.3. Sensor óptico seleccionado y conector necesario para utilizarlo	19
5.4. Interfaz SPI entre un dispositivo maestro y uno o mas dispositivos periféricos esclavos	20
5.5. Distintos modos de operar la comunicación entre maestro y esclavo en una interfaz SPI	21
5.6. Pin-out del trackpad del <i>Blackberry</i> ® BB8520	22
5.7. Operación de escritura	25
5.8. Operación de escritura	26

5.9. Tiempo entre dos operaciones de escritura	26
5.10. Tiempo entre operaciones de escritura y lectura	27
5.11. Tiempo entre operaciones lectura y escritura o lectura y lectura	27
5.12. Byte correspondiente al registro a la dirección Prod_ID	28
5.13. Byte correspondiente al registro a la dirección Motion	28
5.14. Byte correspondiente al registro a la dirección Delta_X	29
5.15. Byte correspondiente al registro a la dirección Delta_Y	29
5.16. Byte correspondiente al registro a la dirección SQUAL	30
5.17. Byte correspondiente al registro a la dirección Configuration_bits	30
5.18. Comparativa entre una conexión HID cableada típica y comunicación in- alámbrica mediante Bluetooth	33
5.19. Esquema de implementación de alto nivel de comunicación Bluetooth me- diante UART	34
5.20. Microcontrolador seleccionado y la placa accesoria necesario para progra- marlo	39
5.21. Esquemático de placa de prueba realizado en <i>Altium Designer</i> ®.	42
5.22. Placa de prueba inicial luego impresa	42
5.23. Izquierda, 5.23a: Medidas de conector en pulgadas (y en milímetros). De- recha, 5.23b: Conector hembra del trackpad BB8520 soldado a la placa de prueba.	43
5.24. Placa complementaria finalizada. De izquierda a derecha se conectan, el modulo Bluetooth, el Arduino, el conversor lógico y por ultimo la placa del sensor	44

5.25.Primer a integración funcional del sensor y comunicación en conjunto	45
5.26.Implementación en bloques del sistema, cada componente separado en módulos con sus respectivos componentes.	46
5.27.Esquemático correspondiente a la placa de procesamiento	49
5.28.Placa procesamiento (dorso y revés) sin soldar.	49
5.29.Esquema correspondiente a la placa del sensor	50
5.30.Placa sensor impresa sin soldar.	50
5.31.Diagrama de bloques de alimentación del sistema.	51
5.32.Arriba (5.32a, 5.32b,5.32c): M6dulo de sensado finalizado. Abajo (5.32d): M6dulo de procesamiento y alimentación finalizados	53
5.33.Diagrama de flujo del software implementado	54
5.34.Distintas vistas de la primera evolución del diseo, los componentes se visualizan en verde y la caja en gris	61
5.35.Distintas vistas de la segunda evolución del diseo	62
5.36.Distintas vistas de la tercer evolución del diseo	63
5.37.Distintas vistas del ensamble final del diseo.	64
A.1. Placa de prueba diseoada en <i>Altium Designer</i> ®.	83
A.2. Dibujo del circuito complementario antes de ser armado	83
A.3. Diseo de placa correspondiente al PCB del sensor	84
A.4. Diseo de placa correspondiente al PCB de procesamiento	84

Índice de tablas

5.1. Comparación entre tecnologías consideradas pero no utilizadas para el sentido de movimiento. [7], [8], [9]	13
5.2. Especificaciones de distintos sensores ópticos que se tuvieron en cuenta. [12] [13] [14] [15]	17
5.3. Modos de funcionamiento de interfaz SPI.	21
5.4. Tiempos requeridos entre distintas operaciones de lectura o escritura.	26
5.5. Tabla de registros y direcciones de memoria utilizadas para obtener información del sensor.	28
5.6. Especificaciones de las distintas placas de Arduino que se tuvieron en cuenta. [23][24][25]	38
5.7. Características electrónicas de cada componente.	40

Glosario

- SNC: Sistema nervioso Central
- C: Cervical
- IoT: Internet of Things
- LED: Light Emitting Diode
- DPI: Dots Per Inch
- SPI: Serial Peripheral Interface
- I2C: Inter-integrated Circuit
- SDIO: Serial Digital Input Output
- UART: Universal Asynchronous Receiver-Transmitter
- IPS: Inches Per Second
- CPI: Counts Per Inch
- OFN: Optical Finger Navigation
- OTS: Optical Tracking Sensor
- IR: Infrarrojo
- V: Volts
- mV: Milivolts
- A: Ampere
- mA: Miliampere
- mA/h: Miliamperio hora
- MOSI: Master Out Slave In
- MISO: Master In Slave Out
- SS: Slave Select
- NCS/ CS: Chip Select
- CLK: Clock
- SCLK: Serial Clock
- CPOL: Clock Polarity
- CPHA: Clock Phase
- SMD: Surface-mounted Device
- GND: Ground
- NC: Not connect
- SQUAL: Surface Quality
- BT: Bluetooth
- SW: Software
- USB: Universal Serial Bus
- HID: Human Interface Device
- SPP: Serial Port Protocol
- HCI: Host Communication Interface
- kbps: Kilo Bits per Second
- Hz: Hercio
- mHz: Milihercio
- mm: Milímetro

Capítulo 1

Introducción

En el siguiente proyecto se desarrolló una prueba de concepto sobre el diseño e implementación de un prototipo de interfaz lingual que asemeja las funciones de un mouse. El dispositivo, idealmente, debe ser accionado y utilizado únicamente con la boca, debido a que está dirigido a pacientes con parestesia, es decir, personas con lesiones nerviosas o mecánicas en las extremidades superiores. La idea del mismo surge debido a las limitaciones que poseen estos pacientes a la hora de utilizar dispositivos electrónicos que, hoy en día son de uso cotidiano en la sociedad, en este caso, específicamente la computadora o dispositivos móviles.

Una vez conocidas el alcance y las limitaciones de un proyecto de estas características, se realizó una investigación sobre los diferentes dispositivos comercializados o en desarrollo, de características similares y que tenían el mismo fin que el dispositivo propuesto. En ese momento, se descubrió el poco desarrollo experimental con el que cuentan los dispositivos relacionados a esta problemática. Adicionalmente, la comercialización de ellos es poco accesible y no suplen en su totalidad la funcionalidad de un mouse.

Durante el desarrollo de este informe se plantean múltiples desafíos como: la elección del sensor a utilizar, el diseño de la electrónica, la comunicación computadora-dispositivo, el diseño de software, la alimentación, el diseño de un primer gabinete, entre otros.

Capítulo 2

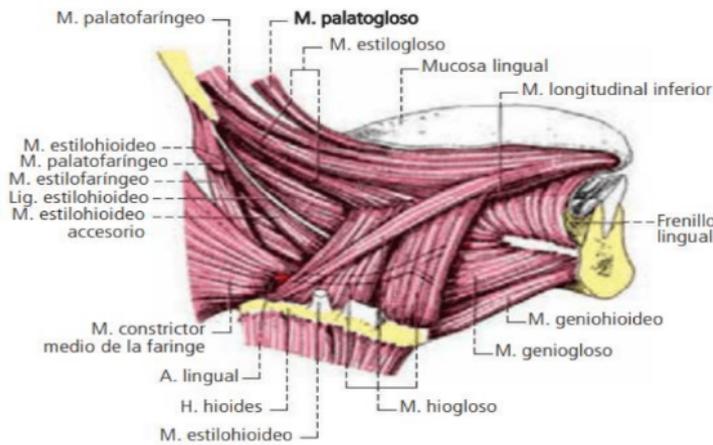
Marco Teórico

2.1. Anatomía de la lengua

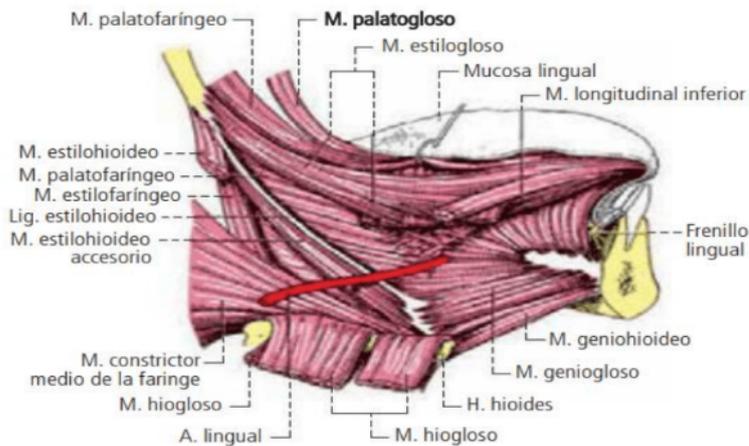
La lengua, órgano impar mediano y simétrico, es una formación muscular móvil, revestida de mucosa. Desempeña una acción esencial como la masticación, la deglución, la succión y la fonación. Es asimismo el órgano receptor de las sensaciones gustativas.

Cuando la boca está cerrada, la lengua, apoyada sobre el piso de la boca, está en contacto con el paladar por arriba, los arcos dentales adelante y a los lados. Cuando la boca está abierta, la lengua forma por su cara superior una saliente convexa que sobrepasa el nivel de los dientes inferiores. La lengua, puede desplazarse en el plano en ambos planos, vertical y horizontal, y, sobre todo, proyectar hacia adelante, por fuera de la cavidad bucal.

La constitución anatómica de la lengua está compuesta por, un armazón osteofibroso, numerosos músculos y un revestimiento mucoso. Por razones relacionadas con el dispositivo, nos centraremos en los numerosos músculos con los que cuenta este órgano para poder accionar el prototipo diseñado. Se pueden distinguir los siguientes músculos en la lengua, un músculo impar y mediano, el músculo longitudinal superior, y los siguientes músculos pares y laterales, que son el geniogloso, el hiogloso, el condrogloso, el estilogloso, el longitudinal inferior, el transverso de la lengua, el vertical de la lengua y el palatogloso.



(a) Músculos superficiales de la lengua.



(b) Músculos profundos de la lengua.

Figura 2.1 Grupos musculares de la lengua.

Adicionalmente, cabe mencionar la inervación de la lengua, la transmisión de las sensaciones de contacto, de temperatura, de dolor, de presión y de posición.

La sensibilidad general de la mucosa lingual y la sensibilidad propioceptiva de los músculos de la lengua es asegurada por tres nervios:

- **Nervio lingual:** Ramo del nervio trigémino, para los dos tercios anteriores
- **Nervio glossofaríngeo:** A través de los ramos linguales para la raíz de la lengua
- **Nervio laríngeo superior:** Ramo del nervio vago, para los pliegues glosopiglóticos, las valléculas epiglóticas y la epiglotis.

2.2. Fisiopatología

Para poder hablar sobre la fisiopatología que llevó a la idea del prototipo, se debe hablar, en primer lugar, de la médula espinal. Está, junto con el cerebro, forma parte del sistema nervioso central. La médula espinal recorre el canal medular y es la encargada de transmitir la información sensorial y motora, solicitando la ejecución de una acción. Adicionalmente, controla: la propiocepción, los reflejos, el control de esfínteres y el dolor [1].

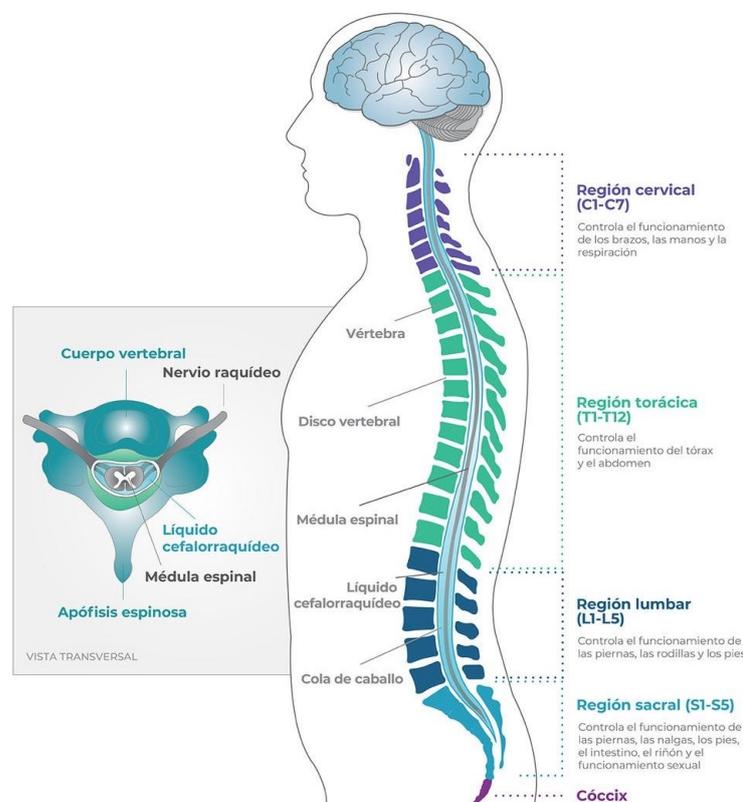


Figura 2.2 Anatomía de la médula espinal

La médula espinal puede sufrir lesiones a lo largo de ella, ver figura 2.2 y la sintomatología depende de la altura de la columna donde se haya generado la misma, ver Figura 2.3 [2]. Estas lesiones se pueden dividir en:

- **Completas:** Toda la función sensorial y toda la capacidad para controlar el movimiento se pierden por debajo de la lesión

- Incompletas: Se conserva cierta actividad motora o sensorial debajo de la zona afectada

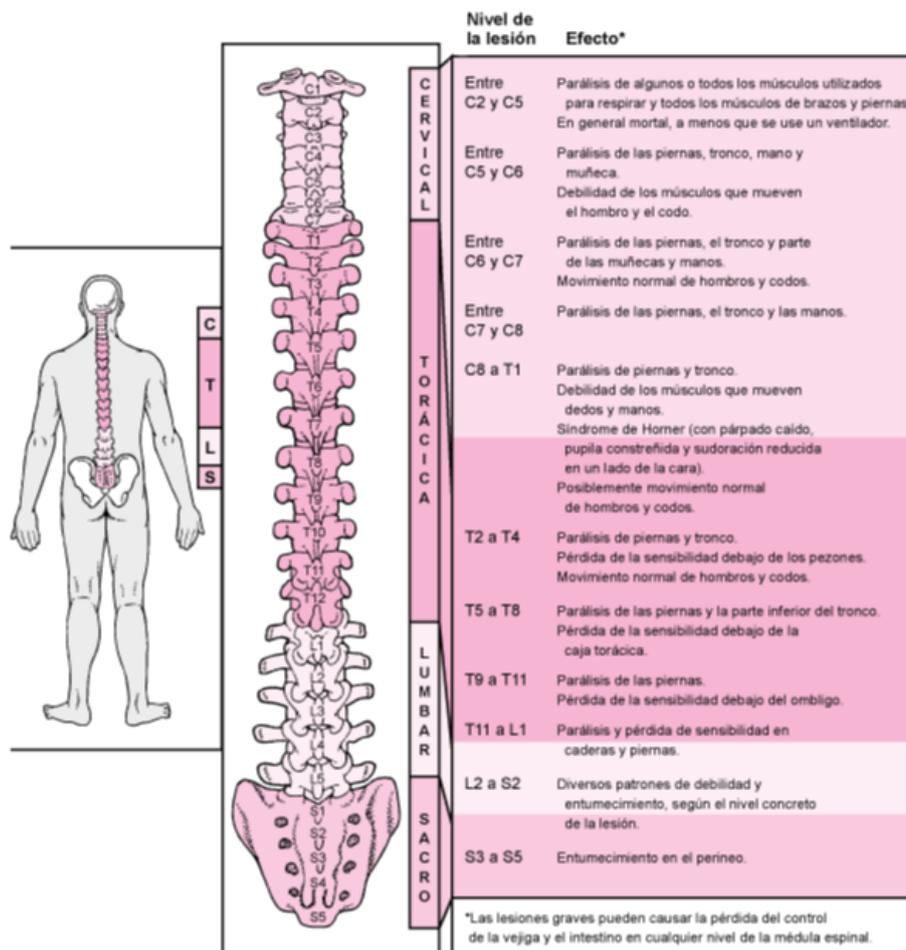


Figura 2.3 Efectos de la lesión espinal

A su vez, la parálisis por una lesión medular espinal se puede dividir en:

- Cuadriplejia o tetraplejia, cuando los brazos, las manos, el tronco, las piernas y los órganos pélvicos están afectados
- Paraplejia: afecta la totalidad o parte de las piernas y los órganos pélvicos

Como podemos observar en la Figura 2.3, este dispositivo está destinado a pacientes con lesión medular espinal en los segmentos cervicales (considerado cuadriplejia o tetraplejia). Cabe aclarar que esto es una generalización de todos los casos en donde se observa este tipo de lesiones, en cada caso particular se debería estudiar al paciente para determinar sus funciones motrices y sensitivas.

Capítulo 3

Concepción

El proyecto surge como respuesta a cubrir necesidades y facilitar el acceso a ciertas tecnologías a pacientes con lesiones medulares a partir de C7, como también con daños en sus extremidades superiores. Se buscó realizar un prototipo que explore la aplicación de distintas tecnologías con el fin de realizar un dispositivo periférico que reemplace el típico mouse de computadora y cuyo usuario sea capaz de accionar utilizando la lengua, esto es, desplazar el cursor en pantalla de una computadora, dispositivo móvil o similar y poder realizar los click necesarios para utilizar dichos dispositivos.

3.1. Alcance del trabajo

A la hora de contextualizar la concepción del dispositivo, se debe dimensionar el número de pacientes que se ve afectado por este tipo de lesiones. Para esto se investigaron distintos estudios realizados.

En 2013, se realizó un estudio en los Estados Unidos en donde se descubrió que aproximadamente 1 de cada 50 personas viven con parestesia (aproximadamente 5,4 millones de personas). Este número representa el 1,7 % de los Estados Unidos. También, se describen las causas de parálisis: ACV (34 %), daños en la médula espinal (27 %), esclerosis múltiple

(19%), entre otras [3]. Adicionalmente, en 2020, el Centro de Estadísticas Nacional de Lesiones de Médula Espinal publicó un trabajo de investigación en donde se encuentran algunas estadísticas generales: aproximadamente existen 18 mil casos nuevos de lesiones de médula espinal por año en Estados Unidos, el 78% de los pacientes son masculinos, el promedio de edad de estos pacientes es de 43 años, entre otros. A su vez, se especifican las causas de las lesiones: accidente vehicular (39%), caídas (32%), violencia (14%), deportes (8%), médico/quirúrgicas (4%) y otras (3%) [4].

3.2. Dificultades y limitaciones

Los pacientes con estas lesiones se ven afectados con múltiples síntomas como: imposibilidad de caminar y mantener el equilibrio; entumecimiento, hormigueo o pérdida de sensibilidad en las manos, los dedos, los pies o los dedos de los pies; pérdida de control en la vejiga; dolor o presión en la cabeza, el cuello o la espalda; entre otros. Estos síntomas se transforman en dificultades y limitaciones durante su vida cotidiana del paciente. El grupo decidió focalizarse en la pérdida de una de las actividades cotidianas más importantes en los últimos años, como lo es, no poder utilizar la computadora o un dispositivo móvil. A su vez, uno de los objetivos del dispositivo, es intentar suplir la gran dificultad que sufren estos pacientes a la hora de la inserción laboral, gracias a sus limitaciones motoras.

Debido a la versatilidad del prototipo planteado, en el futuro, se podría evaluar la posibilidad de ampliar su en aplicaciones de domótica a través del IoT. Es decir, la utilización de dispositivos y sistemas integrados hogareños.

3.3. Características Distintivas

Teniendo en cuenta la concepción, las personas a las que este dispositivo está dirigido y el dispositivo en sí, se plantean las siguientes características con el fin de obtener un

prototipo lo más cercano a un dispositivo final:

- Intuitivo y fácil uso: Importante que el dispositivo requiera de la mínima intervención de otra persona a la hora de ser utilizado. Idealmente el dispositivo, luego de ser instalado, pueda ser utilizado ni bien esté encendido
- Comunicación inalámbrica: Imprescindible que el dispositivo sea inalámbrico, esto aportaría mayor libertad al usuario y, por lo tanto, mayor comodidad
- Compatibilidad con computadoras y sistemas Android: La capacidad de conectar un periférico de estas características tanto a computadoras como a dispositivos móviles aumenta la versatilidad del mismo al ser posible utilizarlo en dispositivos que son casi imprescindibles en el día a día
- Ergonomía adaptada a la mandíbula: Idealmente el dispositivo debe funcionar dentro de la boca, esto supone que la forma del periférico debe ser única a cada usuario para que el mismo sea cómodo y se reduzca la fatiga al utilizarlo
- Conservar las funciones típicas de un mouse: Se entiende por funciones típicas de un mouse la capacidad de mover al cursor en pantalla, y la de realizar dos clicks con las que se realizan todas las acciones básicas

Capítulo 4

Objetivos

El proyecto se orientó al diseño y desarrollo de un prototipo con funciones similares a las de un mouse de computadora pero accionado por la lengua, como prueba de concepto. La intención fue realizar un dispositivo fácil de utilizar e intuitivo para el usuario.

Para la realización del mismo, se propusieron los siguientes objetivos:

■ **Objetivos de Mínima:**

- Definición del protocolo de comunicación y composición de mensajes
- Diseño y desarrollo de hardware, electrónica y firmware
- Integración en prototipo de interfaz lingual externa a la boca
- Cumplimiento de características técnicas

■ **Objetivos de Máxima:**

- Diseño y desarrollo de dispositivo que pueda ser utilizado en la boca
- Evaluar viabilidad de tecnología y futuros desarrollos

Capítulo 5

Diseño e implementación

5.1. Sensores

Un transductor es un elemento que convierte una forma de energía a otra, por ejemplo, sonido a electricidad como en el caso de un micrófono o electricidad en sonido como en el caso de un amplificador de audio [5]. De esta manera un sensor es un transductor que responde a un estímulo, o a una entrada, generando un salida de alguna manera procesable o mensurable. En el caso de un micrófono, este mismo hace de sensor transformando el sonido en una señal eléctrica, medible y procesable, y el sonido es la propiedad física a ser medida [6]. Como se explicó anteriormente en este proyecto, se busca que distintos movimientos de la lengua sean de alguna forma detectados y procesados con el fin de poder mover un cursor en una pantalla y así utilizar una computadora o un teléfono móvil. Para lograr esto primero se debe determinar la tecnología del sensor, o tipo de sensor, y luego el sensor en sí.

5.1.1. Definición de tecnología de sensado

Existen distintos tipos de sensores que se encargan de medir el movimiento. Algunos de ellos son: resistivos, capacitivos, ópticos, de presión (mecánicos), magnéticos. Cada

uno de ellos posee distintas características particulares, las cuales se tuvieron en cuenta a la hora de la elección del sensor. Por esta razón, es fundamental definir correctamente las magnitudes a medir y la precisión necesaria.

Ahora bien, la tecnología para el dispositivo desarrollado tenía como primera limitación el tamaño debido a que el usuario debe accionarlo con su lengua. A su vez, este se debe asemejar lo más posible a la precisión que posee un mouse para computadora para que el usuario no pierda funcionalidad a la hora de utilizarlo. Otro problema identificado fue la estanqueidad de la tecnología, ya que se encuentra en contacto directo con la saliva del usuario por lo que este no debería transmitir ninguna corriente eléctrica.

Para minimizar el impacto de las limitaciones, se buscó una tecnología que pueda persistir dentro de la boca. No obstante, se pensó para su funcionamiento fuera de la boca y luego dentro de la misma, cumpliendo así el objetivo de mínima del trabajo. Este diagrama ayudó a pasar las primeras barreras en cuanto al diseño del dispositivo, ya que se necesitaría una gran inversión para pensar el dispositivo directamente dentro de la boca.

En la tabla 5.1 se enumeran todas las tecnologías analizadas que se tuvieron en cuenta y se descartaron por distintos motivos. En esta tabla se encuentran, las características de cada tecnología y las razones por las cuales no fueron utilizadas en el prototipo final.

Sensor	Características	Justificación
Trackball	<ul style="list-style-type: none"> . Sensor mecánico . Detecta movimiento hacia: arriba, abajo, izquierda, derecha y diagonales . Botón interruptor central iluminado . Tamaño pequeño 	<ul style="list-style-type: none"> . El usuario debe realizar una gran fuerza continúa con la lengua para utilizar este sensor en forma de mouse . La bola que se utiliza para detectar movimiento esta en directo contacto con la electrónica del sensor, la estanqueidad de este sensor es casi nula
Joystick	<ul style="list-style-type: none"> . Sensor mecánico . Considerable tamaño . Interfaz analógica y de fácil procesamiento . Botón incluido 	<ul style="list-style-type: none"> . Gran tamaño para ser utilizado en la boca . Dificultad al utilizarlo con la lengua] . Baja estanqueidad
Resistivo	<ul style="list-style-type: none"> . Piezoresistivo . Poca sensibilidad en comparación a otros sensores . Económicos 	<ul style="list-style-type: none"> . Poca sensibilidad . Implementación compleja . Baja estanqueidad
Capacitivo	<ul style="list-style-type: none"> . Compuestos por cuatro capas conductivas que al interactuar entre ellas se generan los componentes X/Y . Utilizan el campo eléctrico del humano . Gran disponibilidad comercial 	<ul style="list-style-type: none"> . Afectado por la humedad . Alto costo . Implementación compleja debido a la baja estanqueidad y humedad

Tabla 5.1 Comparación entre tecnologías consideradas pero no utilizadas para el sensado de movimiento. [7], [8], [9]

Cada una de las tecnologías de sensores mencionadas en la Tabla: 5.1 fueron probadas realizando pruebas imitando su uso el dispositivo final. La prueba más importante, fue utilizar cada uno de ellos con la lengua para comprobar la comodidad y usabilidad del sensor para el usuario final, tomando en cuenta que este dispositivo está pensado para ser utilizado durante un tiempo prolongado.

En base a las pruebas realizadas, se optó por utilizar un sensor óptico principalmente, porque su precisión se asemeja a la que posee un mouse de computadora y mas importante no son afectados por la saliva dentro de la boca. A continuación, se encuentran algunas de las características de los distintos tipos de sensores ópticos, que justifican la elección del

sensor final.

- Foto-detectores: Los materiales semiconductores sensibles a la luz se han utilizado en una gran variedad de componentes electrónicos. Los foto-detectores van desde simples fotocélulas resistivas hasta foto-diodos y transistores. El detector debe formar parte de un circuito de conmutación o amplificación; por sí solos, pueden transportar pequeñas cantidades de corriente. Algunos ejemplos de su utilización son para: controlar los cierres de las puertas de los ascensores, los contadores de piezas de las cadenas de montaje y los sistemas de seguridad.
- Fibra óptica: La fibra óptica tiene ventajas sobre el cableado eléctrico estándar para algunos entornos. Las fibras no transportan corriente, por lo que son inmunes a las interferencias eléctricas. No suponen ningún peligro de chispas o descargas si el cable se daña. Dependiendo del diseño, la luz de la fibra puede utilizarse como un sensor en sí mismo, o puede servir como una vía de señal para un paquete de sensores separado.
- Pirometro: Los objetos irradian luz según su temperatura y producen los mismos colores a las mismas temperaturas. Un pirómetro estima la temperatura de un objeto detectando el color de la luz que emite. Un pirómetro óptico es un dispositivo más antiguo; el operador compara un filamento brillante en un visor con un objeto caliente para determinar su temperatura. Los pirómetros electrónicos utilizan un semiconductor sensible a la luz para medir la temperatura automáticamente. Los pirómetros se utilizan cuando el contacto directo es inconveniente, inseguro o imposible. Sus aplicaciones incluyen la supervisión de hornos de fundición y la determinación de la temperatura de las estrellas.
- Detector de proximidad: Los detectores de proximidad en miniatura utilizan la luz para detectar la proximidad de los objetos. Contienen una fuente LED y un detector

para medir la luz reflejada. Al medir unos pocos milímetros de lado, son lo suficientemente pequeños como para ser utilizados en pequeños aparatos electrónicos y teléfonos móviles. Tienen un alcance de unos pocos centímetros, útil para determinar la alineación del papel en una fotocopiadora, por ejemplo.

- **Infrarrojo:** Los sensores infrarrojos se utilizan en situaciones en las que la luz visible sería inconveniente o contraproducente. Pueden utilizarse para saber si hay alguien en una habitación, por el calor que desprende el cuerpo de una persona. Los infrarrojos también son útiles para la señalización y constituyen la base de los mandos a distancia de vídeo y audio.
- **Sensor óptico de ratón:** El sensor óptico de un ratón necesita de un LED, generalmente infrarrojo o de luz roja, aunque los hay con LEDs azules, lentes y el sensor propiamente dicho, que es una matriz de píxeles. La luz emitida por el LED pasa a través de un lente que la dispersa e ilumina la superficie debajo del ratón, esta rebota, atraviesa otro lente, pasa a través de una pequeña apertura y finalmente incide sobre la matriz de píxeles. Se puede decir que el sensor óptico de un ratón es una cámara de baja resolución, esta cámara no captura color si no que obtiene un mapa topográfico de la sección de la superficie sobre la cual está apoyado en una escala de grises. El sensor óptico del ratón no tienen la capacidad de almacenar las imágenes sucesivas capturadas por el mismo. El sensor captura las imágenes a una tasa de refresco elevada, que varía con cada sensor, y las compara obteniendo la diferencia sucesiva entre imágenes consecutivas calculando el cambio en el eje X e Y utilizando un algoritmo de correlación [10]. Los cambios en los ejes son enviados al microcontrolador del ratón que se encarga de enviarlos a la computadora. La cantidad de píxeles está directamente relacionada a la sensibilidad del ratón. La sensibilidad, se mide en DPI, es decir puntos por pulgada, esta medida técnicamente indica cuántos píxeles desplaza la punta del cursor cuando el ratón se desplaza una pulgada. A mayor DPI e igual

desplazamiento el cursor se desplazará más píxeles por lo tanto se tiene un configuración más sensible y menos precisa [11]. Inversamente, a menor DPI se tiene mayor precisión y menor sensibilidad, un ratón configurado con pocos DPI requerirá largos movimientos o levantar el ratón de la superficie y apoyarlo donde se lo comenzó a mover.

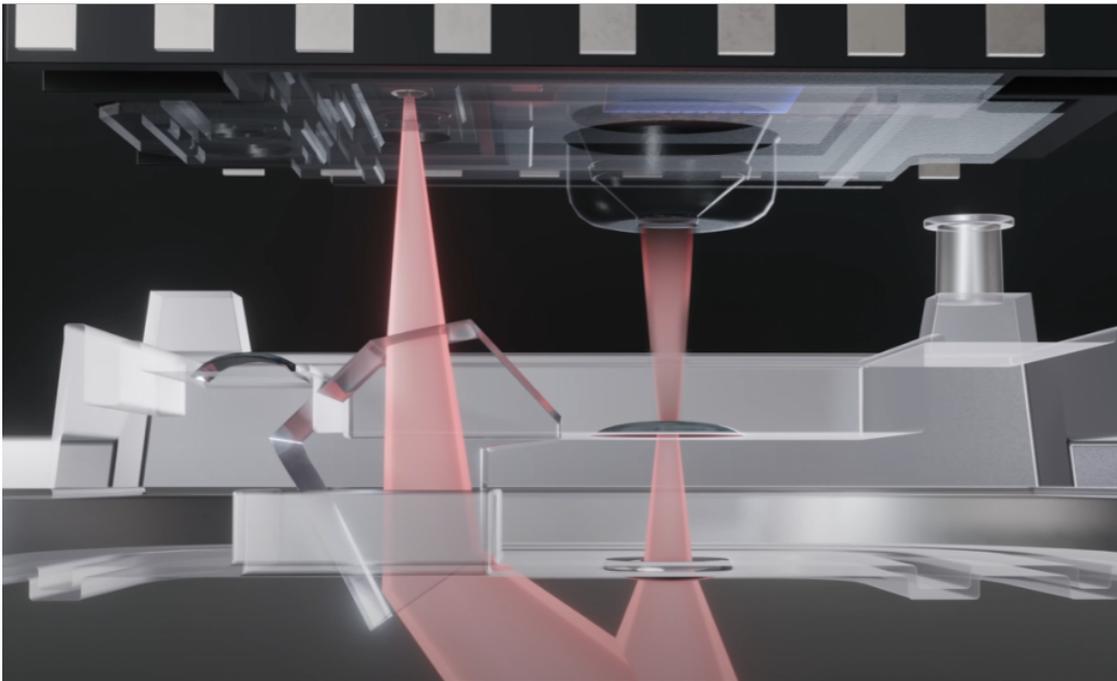


Figura 5.1 Funcionamiento de un sensor óptico de ratón. El LED emite luz visible o IR, esta atraviesa un primer lente que la dispersa e ilumina la superficie. El haz de luz rebota y atraviesa un segundo lente que converge en el sensor óptico.

Como se explicó en la sección superior, se decidió utilizar un sensor óptico de mouse para detectar los movimientos de la lengua y poder realizar el prototipo del dispositivo diseñado.

5.1.2. Definición del sensor óptico

Una vez definida la tecnología y el tipo de sensor a utilizar resta definir el sensor. Si bien al optar por sensores ópticos se elimina el problema de la saliva y superficies capacitivas o

el esfuerzo extra requerido para accionar un sensor mecánico con la lengua, aún quedan limitaciones pendientes como la miniaturización o la estanqueidad del sensor. Dentro de los sensores ópticos y más específicamente dentro de la familia de los sensores ópticos para mouse, ya sean o no infrarrojos, hay distinta variedad en cuanto a encapsulado del sensor y especificaciones técnicas de los mismos. Toda esta familia de sensores requiere de un emisor LED, un lente que permite dispersar la luz emitida por el mismo y un segundo lente que permite converger la luz reflejada al sensor para detectar movimiento sobre una superficie. Ambos lentes deben cumplir con las especificaciones provistas por el fabricante del sensor para asegurar su correcto funcionamiento. Se tuvieron en cuenta los siguientes sensores:

	Tension[V]	Corriente[mA]	Velocidad [ips]	Resolución[cpí]	Interfaz	Formato	Luz
ADNS-3060	3.3	60	40	400, 800	SPI	OTS	Rojo
ADNS-2610	5.0	15	12	400	SDIO	OTS	Rojo
PAW3204DB	2.7	3	28	400 - 1600	SDIO	OTS	Rojo
PAW-A350	1.7 2.1	2.9	20	125 - 1250	SPI/I2C	OFN	IR

Tabla 5.2 Especificaciones de distintos sensores ópticos que se tuvieron en cuenta. [12] [13] [14] [15]

Las especificaciones de estos sensores son similares, por lo tanto el criterio de decisión fue en base al tamaño y la facilidad de implementación del sensor, por ejemplo, para utilizar un sensor del tipo OTS hay que tener en cuenta que se requiere del uso de un lente específico para detectar movimiento. Por lo tanto, en un primera instancia se optó por el PixArt PAW-A350, es un sensor óptico para navegación con dedos, en su encapsulado incluye, el sensor óptico, el LED, en este caso infrarrojo, y el lente para enfocar la luz emitida por el LED. Optando por este sensor se evita el uso de un lente y un LED externo, dos elementos a tener en cuenta tanto en el diseño de software y hardware como en el molde del prototipo final. Este sensor tiene dos grandes desventajas, la primera, el conector requerido para conectar el trackpad y poder utilizarlo es poco común y no es fácil de conseguir. La segunda, y no en particular a este trackpad en específico pero a esta tecnología en general,

es la poca variedad y disponibilidad del mismo. PixArt no cuenta con representante local en el país y vende únicamente al por mayor y los contactos que se intentaron hacer no fueron fructíferos, por lo tanto adquirir pocas unidades de este módulo no fue posible. Si bien a partir de este modelo de trackpad se logró definir el tipo de sensor a utilizar, se tuvo que buscar alternativas más accesibles para poder continuar con el diseño del dispositivo y comenzar con el desarrollo de un primer prototipo.

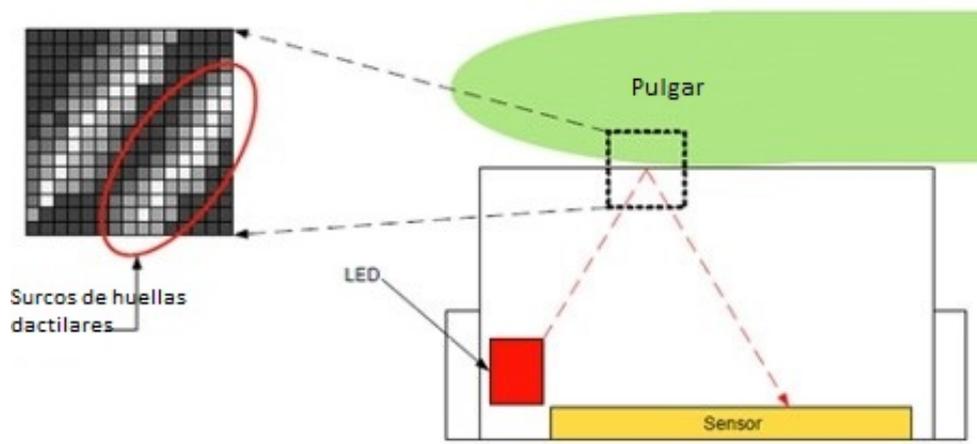


Figura 5.2 Funcionamiento de un trackpad óptico. Se desliza el dedo sobre la superficie del mismo y los cambios detectados por la cámara del mismo son traducidos a movimientos de un cursos.

De esta forma finalmente se optó por el trackpad óptico del *Blackberry*® BB8250 (Nro. parte: HDW-24168) [16]. Este trackpad es muy similar al trackpad anterior, incluye los componentes necesarios para que el sensor funcione, con la ventaja que tiene disponibilidad en el mercado local y el zócalo hembra para conectarlo es más fácil de conseguir que aquel requerido para conectar el PixArt PAW-A350. Lamentablemente la documentación de este sensor es casi nula y no existen hojas de datos del mismo disponibles al público, pero sí existen proyectos que hacen uso de este sensor con documentación disponible que explican y sugieren la interfaz y pin-out del trackpad [17], [18].



(a) Trackpad óptico utilizado en el teléfono móvil BlackBerry® BB8520

(b) Conector hembra que se debe utilizar para conectar el sensor seleccionado. Denominación *DF30FB - 20DS - 0,4V*

Figura 5.3 Sensor óptico seleccionado y conector necesario para utilizarlo

5.1.3. Interfaz de conexión: SPI

Si bien existen distintas interfaces de conexión entre dispositivos digitales es propio del diseño de cada dispositivo admitir una o más interfaces. Dependiendo de la complejidad, antigüedad de cada sensor y del fabricante, los sensores ópticos usualmente admiten interfaz SPI o I2C. En este caso, el sensor seleccionado admite únicamente interfaz SPI, por lo tanto se evitó la comparación entre las dos interfaces.

La interfaz SPI (Serial Peripheral Interface) permite comunicación dúplex, serial y síncrona entre un controlador (maestro) y uno o más dispositivos periféricos (esclavo).[19] La transferencia y recepción de información se da en ambas direcciones y se transmite/-recibe información de a un bit por vez. La comunicación se da entre el controlador y un periférico a la vez, el controlador determina en qué momento cada dispositivo esclavo opera y sincroniza la transferencia y recepción utilizando los pulsos del reloj [20]. La interfaz está compuesta por 4 señales o líneas:

- MOSI (Master Out Serial In): Pin utilizado para transmitir información fuera del maestro hacia el esclavo
- MISO (Master In Slave Out): Inverso al anterior este pin es utilizado para transmitir información fuera del esclavo hacia el maestro

- SS o CS (Slave Select o Chip Select): Se utiliza para seleccionar el periférico con el que el maestro se estará comunicando. Es con este pin que el maestro habilita o no la comunicación de a un periférico a la vez
- SCLK: Se utiliza para enviar la señal del clock con la que se sincronizan los esclavos con el maestro

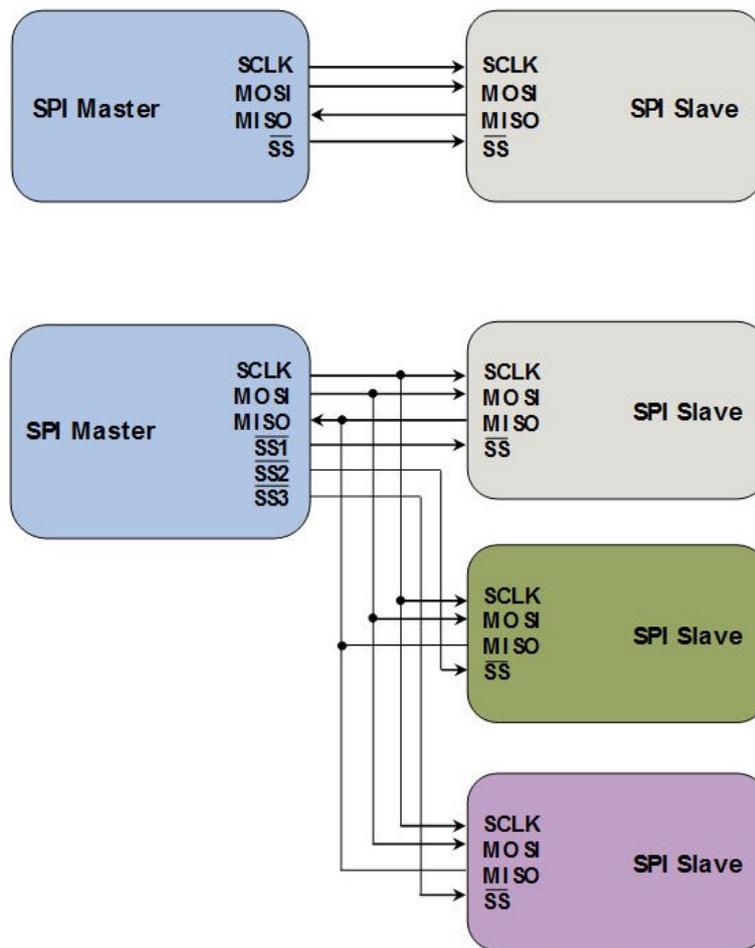


Figura 5.4 Interfaz SPI entre un dispositivo maestro y uno o más dispositivos periféricos esclavos

El clock determinará el modo en que la información fluye del controlador al dispositivo periférico y para esto se utilizan la polaridad (*CPOL*) y la fase de la señal del reloj (*CPHA*). Alterando la polaridad se especifica si el reloj está activo en un flanco alto

($CPOL = 0$) o bajo ($CPOL = 1$) y no afecta el formato de transferencia. Con la fase se determina el tiempo de muestreo de los bits de datos en relación con los pulsos del reloj.[19] Especifica si los bits de datos del maestro a esclavo y del esclavo al maestro se muestrea en el borde trasero de los pulsos del clock ($CPHA = 1$) o si los bits de datos se muestrea en el frente de los pulsos del reloj ($CPHA = 0$) [19], [20]. Al haber dos estados para $CPOL$ y para $CPHA$ se tienen 4 modos de operación:

SPI MODE	CPOL	CPHA	Fundamento
0	0	0	Se muestrea los bits en el frente ascendente de los pulsos del clock
1	0	1	Se muestrea los bits en el borde trasero descendente de los pulsos del clock
2	1	0	Se muestrea los bits en el frente descendente de los pulsos del clock
3	1	1	Se muestrea los bits en el borde trasero ascendente de los pulsos del clock

Tabla 5.3 Modos de funcionamiento de interfaz SPI.

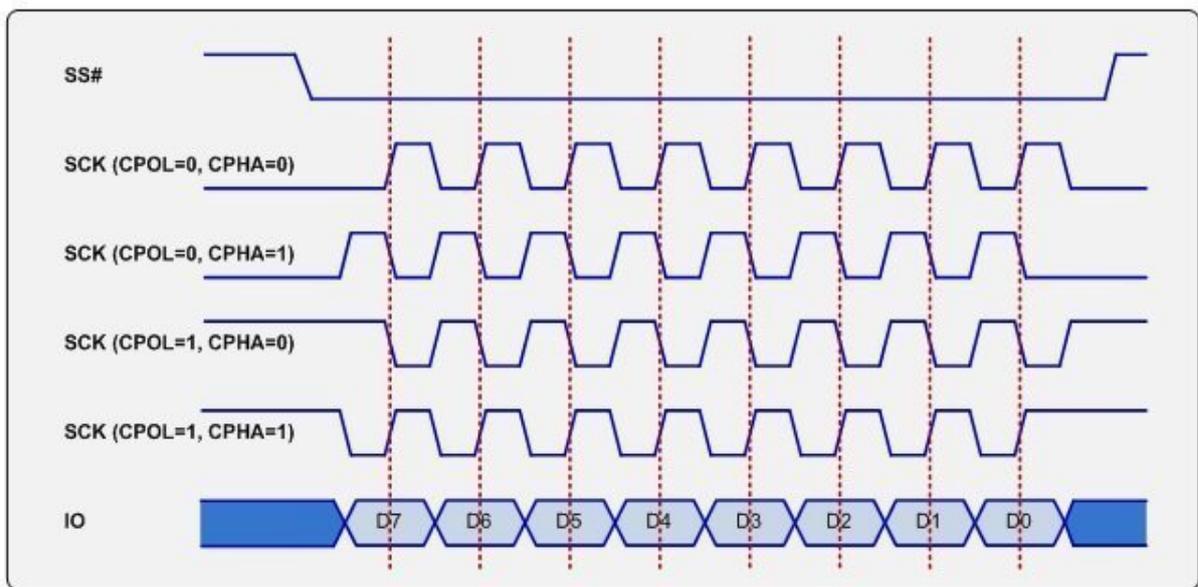


Figura 5.5 Distintos modos de operar la comunicación entre maestro y esclavo en una interfaz SPI

Si bien tener el tipo de interfaz determinado por el sensor puede parecer ventajoso, se está perdiendo la posibilidad de elección entre otras interfaces que pueden facilitar el diseño general. Por ejemplo, si bien la conexión SPI tiene mayor velocidad que una conexión I2C, SPI es más compleja de realizar ya que requiere de 4 líneas de comunicación

en vez de 2 [21]. En esta aplicación en particular al tener solamente un sensor conviene priorizar velocidad a la simplificación de la electrónica pero de todas formas siempre es ventajoso que se tenga más de una opción a la hora de diseñar un sistema.

5.1.4. Características técnicas del sensor BB8520

La alimentación típica del trackpad es de 2.85V y la comunicación con el mismo se da a través de una interfaz SPI. Al tener una tensión de operación de 2.85V el nivel de lógica que utiliza este trackpad es también de 2.85V, importante a la hora de diseñar la electrónica del prototipo y la comunicación con el microcontrolador [17].

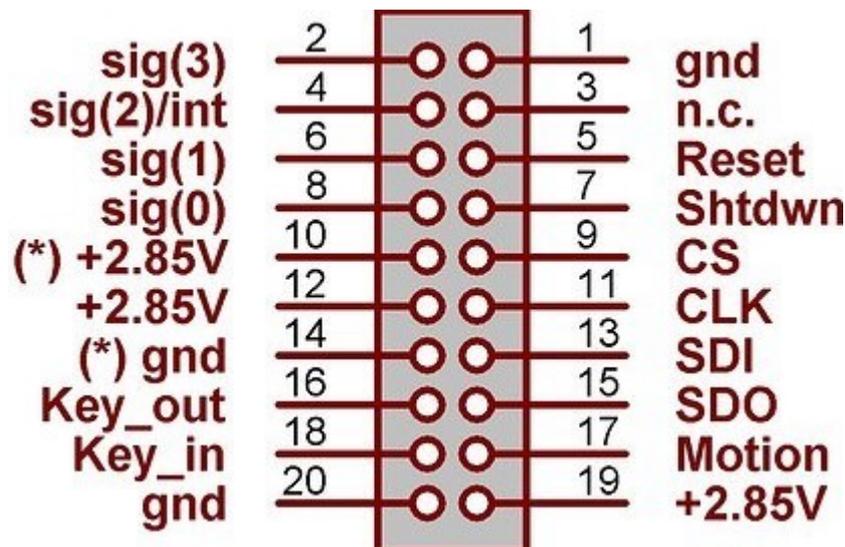


Figura 5.6 Pin-out del trackpad del BlackBerry® BB8520

Pinout

- Los pines de señales del tipo $sig(x)$ o $sig(2)/int$ se pueden omitir ya que no son utilizados por el trackpad
- Aquellos pines indicados con (*) se deben conectar en serie a una resistencia de 220Ω
- El pin *n.c.* (Not Connect) como el nombre indica no se debe conectar

- Pines SPI:
 - CS: Chip Select
 - CLK: Serial Clock
 - SDO (MISO): Serial Data Out
 - SDI (MOSI): Serial Data In
- Motion: Pin de interrupción, se cambia su estado a 0 cuando hay información disponible para leer o cambia a 1 cuando no la a hay
- Shtdwn y Reset como su nombre lo indica son para apagar el trackpad o resetearlo, respectivamente
- Key_out y Key_in se utilizan para el control del pulsador incluido en el sensor

De todos los pines los de mayor relevancia y utilidad son los correspondientes a la interfaz SPI, Motion, Reset y Shtdwn y los necesarios para alimentar al trackpad.

Al no tener hoja de datos del trackpad no es posible conocer las especificaciones técnicas del mismo (resolución, velocidad de detección, etc.) pero con la información disponible si es posible obtener registros y la forma de operarlo. A partir del pinout, la interfaz de comunicación y la alimentación necesaria se desarrolló una placa de prueba (Ver sección Desarrollo de Hardware) para conectar el trackpad a un Arduino y realizar pruebas de lectura y escritura de registros. Como guía de funcionamiento de este trackpad se tomaron distintas hojas de datos de sensores ópticos ya que para un funcionamiento básico, los registros y dirección de memoria son similares entre todos. Se programó un código en el IDE de Arduino para realizar la comunicación entre microcontrolador y trackpad utilizando la librería de interfaz SP3 de Arduino para obtener las secuencias de lectura, escritura y valores de registros del trackpad (Ver sección Desarrollo de Software).

Como referencia inicial utilizamos las hojas de datos de los 4 ya mencionados sensores y de los sensores ADNS-3000, ADNS-5095 y ADNS-3050. Finalmente según los resultados

obtenidos utilizamos como referencia principal la hoja de datos del sensor ADNS-3060 de AVAGO [12].

Operaciones

La comunicación con el sensor BB8250 se maneja a través del pin NCS (Slave Select o Chip Select), como se explicó anteriormente cuando el Pin se encuentra en estado LOW el sensor es capaz de recibir la señal de reloj del dispositivo maestro y recibir o enviar datos al mismo. En otras palabras, dependiendo del estado del pin NCS se activa la comunicación SPI o se la ignora. Este pin es extremadamente útil a la hora de brindar estabilidad a la conexión ya que al cambiar su estado a HIGH se inactiva y por lo tanto se resetea el puerto, el cambio de estado de este pin permite minimizar el tiempo de comunicación entre maestro y esclavo minimizando la probabilidad de errores. Esto permite resetear la comunicación en caso de errores, abortar operaciones de escritura o lectura, o mismo encuadrar operaciones, iniciando cada operación con el cambio de estado a LOW del pin y finalizando la comunicación con el pin en HIGH.

La operación de lectura consiste de 2 bytes, el primer byte se envía a través del pin MOSI (del maestro al esclavo) y en el bit más significativo contiene un "0", este bit indica la dirección en la que fluyen los datos ya que la comunicación es primero del maestro al esclavo y luego del esclavo al maestro, el resto de bits contiene la dirección del registro que se requiere leer. El segundo byte contiene el valor guardado en la dirección de memoria indicada por el registro del primer byte y se envía a través del pin MISO, es decir, del esclavo al maestro. Desde el punto de vista del dispositivo esclavo, se envían datos a través del pin MISO en cada caída del pulso del reloj y se leen datos provenientes del maestro (MOSI) en cada subida del flanco del pulso del reloj. La operación de lectura si bien es rápida no es instantánea ya que el sensor tiene un tiempo de respuesta determinado entre el primer byte y el segundo. El tiempo de respuesta se lo conoce como t_{RAD} o $t_{RAD-MOT}$, siendo este último el tiempo de respuesta para la lectura de registros que contengan

información del movimiento (ver Tabla 5.4). El primero es un tiempo de mínimo $50 \mu s$ mientras que el segundo es de $75 \mu s$, hay que tener en cuenta estos tiempos para una correcta lectura y estabilidad del software. De esta manera la operación de lectura se realiza primero cambiando el estado de NCS a LOW, luego se envía a través de MOSI la dirección de memoria a ser leída, se espera t_{RAD} o $t_{RAD-MOT}$, se obtiene el valor guardado en ese registro y la comunicación finaliza con el cambio de estado de NCS a HIGH.

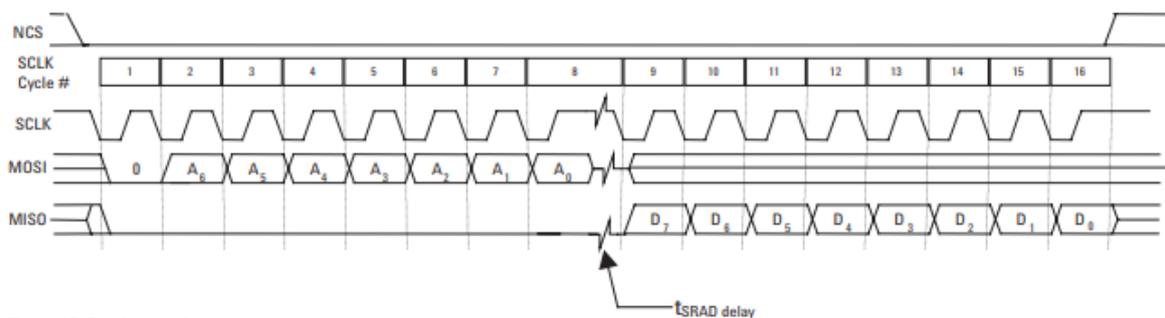


Figura 5.7 Operación de escritura

De forma similar la escritura de registros contiene dos bytes, el primer byte contiene la dirección del registro a ser escrito o reescrito y también se transmite a través de MOSI, pero, a diferencia de la lectura, el bit más significativo contiene un 1, ya que la comunicación solamente es de maestro a esclavo. El segundo byte contiene el valor con el cual se reescribe dicho registro y también se envía a través del pin MOSI. Nuevamente el sensor lee el pin MOSI en cada subida del flanco de la señal del reloj. De esta manera, una correcta operación de lectura inicia con el cambio de estado del pin NCS a LOW, sigue con el envío del registro a ser escrito en el primer byte y con el valor a reescribir ese registro en el segundo byte y continúa con el cambio de estado de NCS a HIGH para terminar la comunicación. Por último siempre es conveniente corroborar que la escritura haya sido exitosa realizando una operación de lectura sobre ese registro.

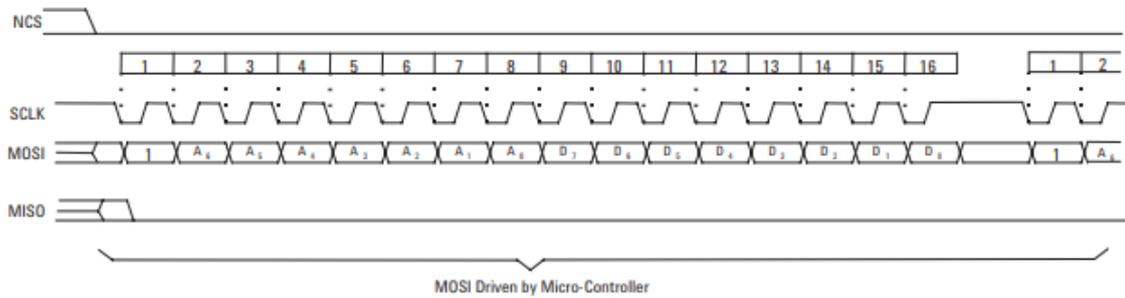


Figura 5.8 Operación de escritura

Además del tiempo que se debe esperar entre el primer byte y el segundo en una operación de lectura también hay tiempos requeridos entre operaciones en el puerto serie. Estos tiempos cambian según las operaciones que se realicen.

Símbolo	Tiempo [μs]
$t_{s_{WW}}$	50
$t_{s_{WR}}$	50
$t_{s_{RW}}$	0.25
$t_{s_{RR}}$	0.25

Tabla 5.4 Tiempos requeridos entre distintas operaciones de lectura o escritura.

Escritura - Escritura

Para asegurar que dos operaciones de lectura consecutivas se completen correctamente, el flanco ascendente del pulso del reloj para el último bit del segundo comando de escritura debe ocurrir al menos $50\mu s$, luego del flanco ascendente del reloj para el último bit del primer comando de escritura. Es decir, el tiempo en el que la primera y segunda operación de escritura termina debe ser mayor o igual a $50\mu s$.

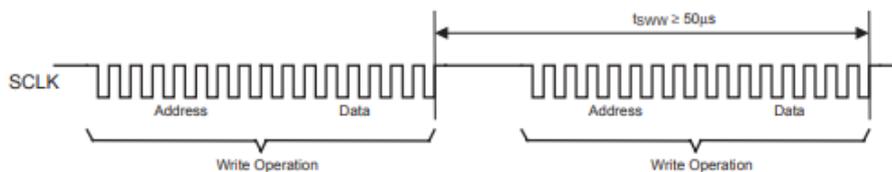


Figura 5.9 Tiempo entre dos operaciones de escritura

Escritura - Lectura

De forma similar ocurre con una operación de escritura seguida de una operación de lectura, el tiempo en que ambas terminan debe ser mayor o igual a $50 \mu s$.

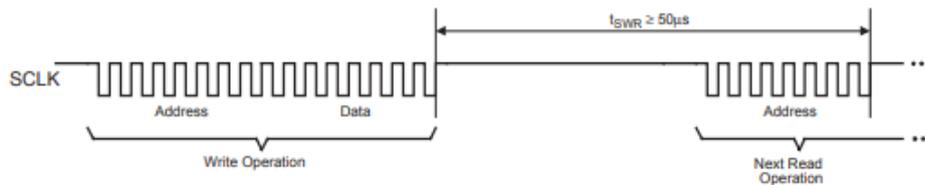


Figura 5.10 Tiempo entre operaciones de escritura y lectura

Lectura - Escritura o Lectura - Lectura

En el caso de operaciones seguidas a un comando de lectura, el flanco descendente del pulso del reloj del comando de escritura o lectura debe ser al menos $250 ns$ después del último flanco ascendente del pulso del reloj del comando de lectura anterior. Es decir, entre lectura y escritura o lectura y lectura debe haber un tiempo mayor o igual a $250 ns$.

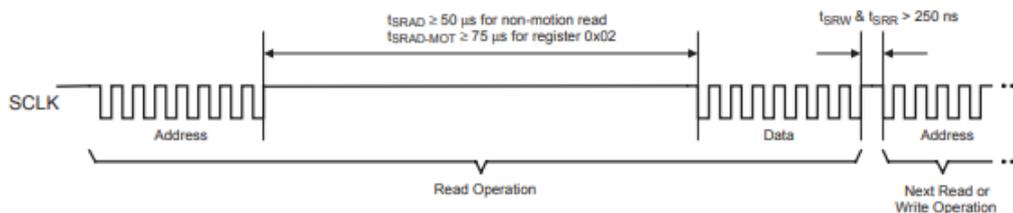


Figura 5.11 Tiempo entre operaciones lectura y escritura o lectura y lectura

Registros

El sensor ADNS-3060 tiene 39 registros que son accesibles mediante el puerto serie. Todos los registros admiten lectura, ya que guardan datos relevantes al movimiento o datos de la configuración del sensor y algunos admiten escritura, estos son únicamente para la configuración del sensor. A cada registro le corresponde una dirección de memoria que permite acceder a ese registro ya sea para escritura o lectura del mismo. No hay que olvidar que nos basamos en el sensor ADNS-3060 por lo tanto no podemos definir

con exactitud todos los registros de nuestro sensor, pero si pudimos obtener los registros necesarios para poder utilizar el sensor BB8250. Si bien las direcciones de memoria son las mismas el valor que contiene cada registro puede variar pero cumple con la misma función.

Dirección	Registro	Escritura/Lectura [R/W]	Predeterminado [μ s]
0x00	Product_ID	R	0x0D
0x00	Motion	R	0x00
0x00	Delta_X	R	0x00
0x00	Delta_Y	R	0x00
0x00	SQUAL	R	0x00
0x00	Configuration_bits	R/W	0x09

Tabla 5.5 Tabla de registros y direcciones de memoria utilizadas para obtener información del sensor.

Dirección 0x00: Product_ID

Bit	7	6	5	4	3	2	1	0
Field	PID ₇	PID ₆	PID ₅	PID ₄	PID ₃	PID ₂	PID ₁	PID ₀

Figura 5.12 Byte correspondiente al registro a la dirección Prod_ID

- Tipo de dato: Entero sin signo de 8 bits

Este registro contiene un valor de identificación único asignado al sensor. El valor en este registro no cambia nunca y se utiliza para comprobar que la comunicación serie funciona correctamente.

Dirección 0x02: Motion

Bit	7	6	5	4	3	2	1	0
Field	MOT	Reserved	Reserved	OVF	Reserved	Reserved	Reserved	RES

Figura 5.13 Byte correspondiente al registro a la dirección Motion

- Tipo de dato: Campo de bits

Permite determinar si el sensor registró movimiento, para esto se debe observar el bit número 7 de dicho registro. Si se registra movimiento se deben leer las direcciones 0x03 y 0x04 para obtener el movimiento acumulado. También permite saber la resolución actual del sensor y hay overflow del buffer de los registros Delta_X y Delta_Y.

Dirección 0x03: Delta_X

Bit	7	6	5	4	3	2	1	0
Field	X ₇	X ₆	X ₅	X ₄	X ₃	X ₂	X ₁	X ₀

Figura 5.14 Byte correspondiente al registro a la dirección Delta_X

- Tipo de dato: Complemento a dos de un numero de 8 bits

Contiene el valor del desplazamiento acumulado en el eje X, puede ser negativo como positivo. Una vez leído el registro este se borra y vuelve a cero.

Dirección 0x04: Delta_Y

Bit	7	6	5	4	3	2	1	0
Field	Y ₇	Y ₆	Y ₅	Y ₄	Y ₃	Y ₂	Y ₁	Y ₀

Figura 5.15 Byte correspondiente al registro a la dirección Delta_Y

- Tipo de dato: Complemento a dos de un numero de 8 bits

Contiene el valor del desplazamiento acumulado en el eje Y, puede ser negativo como positivo. Una vez leído el registro este se borra y vuelve a cero.

Cabe destacar que tanto el registro Delta_Y como el Delta_X tienen buffers internos ya sea para acumular lectura de movimientos o para almacenar más bits de movimiento. Al leer el bit de movimiento del registro Motion los valores de los registros Delta_X y Delta_Y se congelan. Puede suceder que los buffers de estos registros se llenen y saturen (overflow) en cuyo caso habría que leer consecutivamente los valores de los registros de movimiento hasta que el bit MOT del registro Motion vuelva a cero.

Dirección 0x05: SQUAL

Bit	7	6	5	4	3	2	1	0
Field	SQ ₇	SQ ₆	SQ ₅	SQ ₄	SQ ₃	SQ ₂	SQ ₁	SQ ₀

Figura 5.16 Byte correspondiente al registro a la dirección SQUAL

- Tipo de dato: Primer 8 bits mas relevantes de un entero de 10 bits sin signo

Indica la calidad de la superficie, el registro contiene una medida equivalente a un cuarto del número de características de la superficie visibles al sensor en su posición actual. Por lo tanto si no hay superficie sobre el sensor SQUAL es cero y varía con el movimiento de la superficie sobre el sensor ya que cambia la superficie. A mayor SQUAL mayor características tiene la superficie y por lo tanto es más fácil detectar el movimiento.

Dirección 0x0a: Configuration_bits

Bit	7	6	5	4	3	2	1	0
Field	0	LED_MODE	Sys Test	RES	Reserved	Reserved	Reserved	Reserved

Figura 5.17 Byte correspondiente al registro a la dirección Configuration_bits

- Tipo de dato: Campo de bits

Permite cambiar la configuración del sensor. Cada bit corresponde a una configuración, de interés son el bit que configura el estado del LED (LED_MODE) si está prendido constantemente o es intermitente y el bit de resolución del sensor (RES) que permite cambiar la resolución del sensor entre 400 u 800 CPI (count per inch).

5.2. Comunicación

5.2.1. Definición del tipo de comunicación

Otro de los componentes importantes a definir de la tecnología fue el método de comunicación entre el sensor y la computadora. Se tuvieron en cuenta múltiples opciones, pero se terminó definiendo entre comunicación Bluetooth y antena USB (hace de receptor mientras que está conectado a la computadora). Para tomar la decisión se tuvieron en cuenta: la conectividad, la aplicación de determinada comunicación y la compatibilidad.

Por las razones mencionadas anteriormente se definió utilizar la comunicación Bluetooth utilizando un módulo que se conecta a la placa de Arduino. Adicionalmente, al elegir esta comunicación, el dispositivo puede ser conectado tanto a una computadora como a un celular.

Una vez elegida la comunicación Bluetooth se tuvo que investigar respecto a los requerimientos de dicha comunicación. Dentro de sus requerimientos encontramos: el perfil, la compatibilidad con Windows, que permite tanto ser utilizado como esclavo o maestro, entre otros. Estos requerimientos jugaron un papel importante en la elección del módulo a utilizar.

5.2.2. Determinación del modulo Bluetooth

Perfiles

Un perfil de comunicación Bluetooth se refiere a una especificación sobre la comunicación wireless entre dispositivos, es una de las especificaciones que reside en el núcleo ya que es de suma importancia. Este determina la forma en la que el dispositivo utiliza el Bluetooth para comunicarse, ya que posee distintas funciones para realizar la misma. Los perfiles proporcionan normas que los fabricantes siguen para permitir que los dispositivos utilicen Bluetooth de la manera prevista. Como mínimo, cada perfil contiene información

sobre: dependencias de otros formatos, formatos de interfaz de usuario sugeridos y partes específicas de la pila del protocolo Bluetooth utilizadas por el perfil.

Al estudiar la compatibilidad del perfil Bluetooth y Windows, se encontró que existen distintos módulos que necesitan del desarrollo de un Driver compatible con Windows que reconozca el dispositivo Bluetooth como un periférico, que es la funcionalidad que se estaba buscando. Al investigar los distintos perfiles de Bluetooth que existen, se encontraron que existen más de 35 perfiles, de los cuales se eligió utilizar el perfil Human Interface Device Profile (HID).

El perfil HID define el protocolo entre [22]:

- Dispositivo HID: facilita la entrada y salida de datos del usuario desde y hacia el host
- Host: Utiliza o solicita los datos y servicios del dispositivo HID

Este perfil permite controlar el conjunto de funciones del dispositivo y el reporte HID, aquel que el host utiliza para interpretar los datos recibidos como valores ASCII. El reporte HID sigue el ya conocido protocolo estándar HID del USB (Universal Serial Bus) para aprovechar los controladores existentes del host, es decir, los Drivers de la computadora con Windows o un dispositivo Android existente. En un escenario de uso típico un módulo Bluetooth configurado con un perfil HID reemplaza el cable USB. Por ejemplo en el caso de un teclado Bluetooth, el valor ASCII originado luego de la pulsación de una tecla se convierte en un código de escaneo en un informe HID sin procesar, que luego el módulo Bluetooth envía a través del enlace Bluetooth al host. El software del controlador del host decodifica el informe HID sin procesar y pasa los valores clave a la aplicación que se ejecuta en la PC.

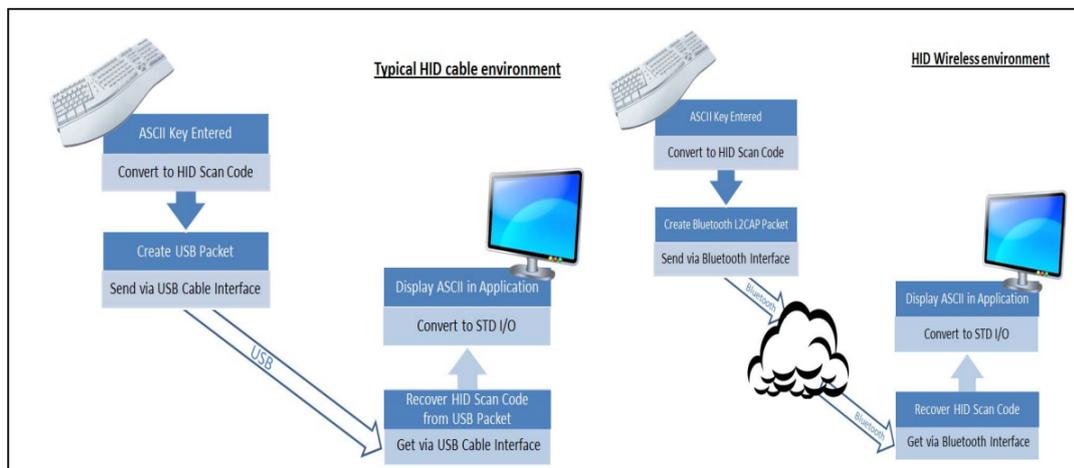


Figura 5.18 Comparativa entre una conexión HID cableada típica y comunicación inalámbrica mediante Bluetooth

Módulos Bluetooth

Inicialmente se tuvieron en cuenta módulos Bluetooth HID o módulos capaces de ser configurados con el perfil HID. El módulo ideal para esta aplicación es el RN42 de Roving Networks, este módulo cuenta con 5 perfiles y 6 modos configurables y documentación disponible al público. Lamentablemente este módulo no se encuentra disponible en el mercado local y los costos de importación del mismo son altos, más si se trata de una sola unidad. Por lo tanto se buscaron posibles soluciones, desde retroceder con el desarrollo e implementar una antena USB hasta utilizar módulos Bluetooth alternativos. Se realizaron pruebas con los módulos más accesibles y disponibles, el HC-06 y el HC-05, ambos módulos compatibles únicamente con el perfil SPP (Serial Port Protocol) con la diferencia que el HC-06 es únicamente esclavo y el HC-05 es tanto esclavo y maestro. De fábrica ninguno de los dos módulos anteriores son capaces de ser configurados con perfiles HID con el detalle que el microcontrolador del HC-05 es idéntico al micro controlador del módulo RN42.

Finalmente, se optó por la utilización del HC-05 con el firmware del RN42, a fines prácticos se denomina al nuevo módulo como HC-05*. Esto se logró utilizando un chip conversor del tipo FTDI para poder conectarlo directo a la computadora y luego poder

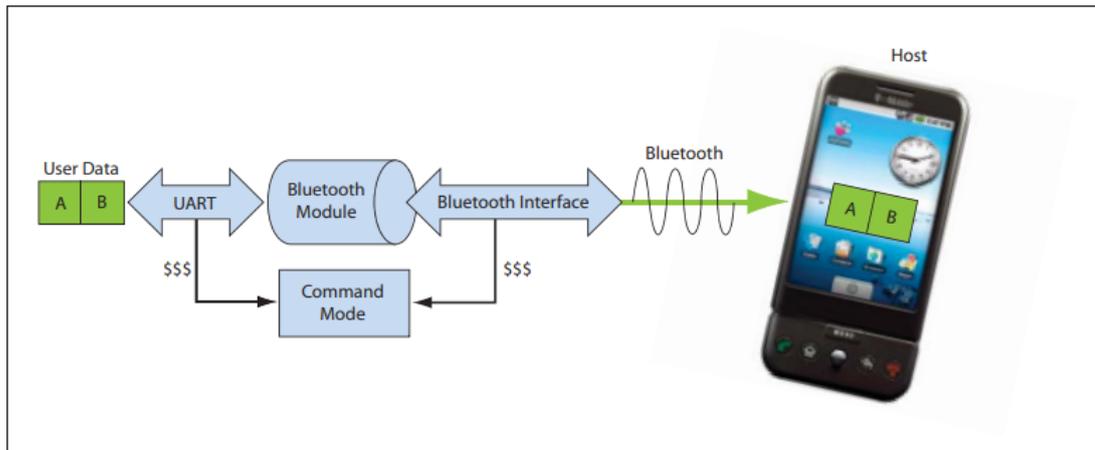


Figura 5.19 Esquema de implementación de alto nivel de comunicación Bluetooth mediante UART

extraer el firmware original, eliminarlo del módulo y escribir en el mismo el firmware del RN42. Una ventaja adicional es que el HC-05 de fábrica cuenta únicamente con 6 pines fáciles de conectar a un Arduino mientras que el formato de fábrica del RN42 es de montaje superficial con una interfaz más rigurosa que hubiese implicado un estudio exhaustivo de la hoja de datos para resolver su conectividad.

Interfaz de conexión

Una vez modificado el firmware del módulo HC-05, se determinó la interfaz con la que el host iba a comunicarse con el huésped (HCI). Dado el firmware y el módulo utilizado, se contaba con dos opciones: HCI sobre UART (Receptor/transmisor asíncrono universal) o HCI sobre USB.

HCI sobre UART

En este modo, la interfaz del hardware entre el controlador host y el módulo Bluetooth es el UART. Es necesario interconectar las señales de control de flujo entre el controlador anfitrión y el módulo Bluetooth para que la interfaz HCI funcione. Si no lo hace, puede hacer que el procesador anfitrión y el módulo Bluetooth se de-sincronizan, rompiendo así el enlace Bluetooth.

HCI sobre USB

En este modo, la interfaz de hardware entre el procesador anfitrión y el módulo Bluetooth es el USB. En esta arquitectura, el módulo Bluetooth es el esclavo USB y el procesador anfitrión es el anfitrión USB. La información se envía en serie y se tiene a través de los pines que van desde el host al huésped y del huésped al host.

Por las características descritas fue que decidimos elegir HCI sobre UART.

A su vez, la interfaz HCI sobre UART cuenta con dos modos operacionales: modo data (default) y modo comando. En el primer modo, el módulo Bluetooth funciona solamente como un tubo de datos. Cuando este recibe los datos, se desprende de los títulos y trailers del Bluetooth y solo pasa los datos del usuario al UART. Cuando los datos son escritos en el UART, el módulo construye el paquete Bluetooth y lo envía. Entonces, el proceso entero de recibir y enviar datos al huésped es totalmente transparente hasta el final del microcontrolador.

Se puede configurar el módulo ingresando al modo comando y enviando comandos ASCII por un puerto serial o por el link Bluetooth. Una vez que se cambian los parámetros de configuración, estos se mantienen hasta realizar un reseteo de fábrica. Se ingresa al modo comando enviando el string '\$\$\$' al módulo a lo que el módulo devuelve 'CMD', si la comunicación es correcta, indicando que se encuentra en modo comando.

Una vez en modo comando, podemos enviar comandos de configuración vía la terminal. Al ingresar un comando válido el módulo devuelve 'AOK'. Al ingresar un comando invalido, devuelve 'ERR'. Para comandos no reconocidos, devuelve '?' [22].

5.2.3. Características técnicas del modulo RN42

Algunas de las características que se encontraron sobre el RN42 hablan de su extensa funcionalidad. Cómo se utilizó el firmware de dicho módulo se utilizaron las funcionalidades específicas a la configuración de perfiles y conectividad con otros dispositivos. La configuración default del módulo son:

- Modo Bluetooth: Esclavo
- Pin Code = 1234
- Puerto Serie: 115200 Kbps de baud rate, 8 bits, sin paridad y 1 bit de stop
- Control de flujo a través del puerto serie: Deshabilitado
- Modo de baja potencia: Apagado

A su vez, el módulo cuenta con una gran cantidad de modos operacionales y comandos, a continuación se encuentran los utilizamos para la utilización del módulo RN42:

- **\$\$\$**: Causa que el dispositivo ingrese a modo comando. El módulo envía caracteres como datos hasta que encuentra esa secuencia
- **- - -** : Produce que el módulo salga del modo comando
- **GK**: Indica el status actual del módulo. Donde 1,0,0 indica que está conectado y 0,0,0 indica que no está conectado
- **S ,6**: Setea el perfil, donde el número (en este caso 6) indica el perfil HID
- **SM,1**: Setea el modo del modulo, donde el número (en este caso 1) indica el modo esclavo
- **SH,0220**: La bandera de registro HID es un registro mapeado por bits que es configurado mientras se encuentra en modo comando. El registro enviado en este comando corresponde a un número de 10, donde cada bit implica una configuración específica independiente del resto, a excepción de los bits entre el 7 y el 4, que configuran el descriptor del modo HID. En este caso para configurarlo como ratón basta con que esos 4 bits sean 0010, que corresponde al registro '0220'
- **R,1**: Reinicia el modulo

- **I,6:** Tiempo en el cual busca dispositivos cercanos, en este caso son 6 segundos
- **CFI:** Genera que el modulo se conecte al primer dispositivo que encuentra y queda configurado en modo de data rápida

Según los comandos mencionados anteriormente, a la hora de conectar, el módulo es reconocido como un dispositivo HID mas específicamente un mouse. La conexión se administra desde el dispositivo maestro y una vez conectado el mismo modulo es capaz de realizar la conexión por si solo, si encuentra dicho dispositivo.

5.3. Hardware

El hardware consiste en todo componente interconectado en un sistema que tiene la capacidad de realizar operaciones lógicas para almacenar o transformar información. Una vez definidos dos de los componentes de mayor relevancia en cuanto al funcionamiento electrónico del dispositivo se comenzó con la definición, el diseño y la implementación del hardware.

5.3.1. Definición del microcontrolador

Un microcontrolador es un circuito integrado programable, es el cerebro de un sistema informático y el motor que activa el funcionamiento de un equipo. Usualmente integra memoria tanto volátil como no volátil y tiene capacidad de procesamiento dada por una unidad central de procesamiento.

A la hora de seleccionar un microcontrolador se optó la familia de placas Arduino con procesadores relativamente simples. Esto fue así ya que se tenía experiencia previa manipulando estas placas y programando sobre el ambiente de programación de Arduino, que es amigable y más simple que la programación de bajo nivel que puede llegar a ser necesaria con otros controladores, además, debido a la gran comunidad que utiliza Arduino y

a la abundante documentación. Se decidió optar por placas que utilicen el procesador **ATmega328** o **ATmega324**, dos controladores casi idénticos con la diferencia que el último cuenta con soporte USB integrado y no necesita del circuito integrado FTDI para poder comunicarlo con una computadora. La elección de procesadores es tal ya que no se requería alta capacidad de procesamiento y dentro de las opciones que ofrece la gama de productos de Arduino estos controladores son de los más simples.

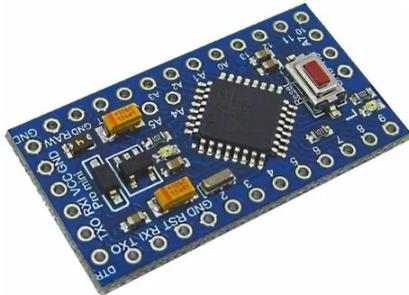
Ahora bien hay más de una placa que hace uso de este procesador y se tuvieron en cuenta las siguientes, priorizando tamaño:

	Dim [mmxmm]	Interfaces	Conectividad	Tension [V]	Clock [MHz]
Arduino Nano	18 x 45	SPI:1 I2C: 2 UART: 1	USB-B Mini	5-12	16
Arduino Pro Mini	18 x 32	SPI:1 I2C: 1 UART: 1	USB-B Mini	3.3-12	16
Arduino Pico	15 x 15	SPI:1 I2C: 1 UART: 1	USB-B Mini	5-12	16

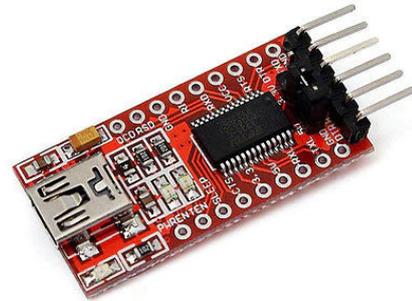
Tabla 5.6 Especificaciones de las distintas placas de Arduino que se tuvieron en cuenta. [23][24][25]

Si se prioriza tamaño observando la Tabla 5.6 la opción ideal es el Arduino Pico, mismas características que el resto, cumple con los requerimientos de interfaz, es diminuto y corre ventaja de poseer un conector micro USB, más reciente y común que el USB-B Mini del Arduino Nano. Lamentablemente no fue posible acceder a ese modelo. Por lo tanto se optó por el segundo modelo de menor tamaño, el **Arduino Pro Mini**, la gran desventaja de esta placa es que no cuenta con un USB de ningún tipo, por lo que la conexión con una computadora para su programación debió hacerse a través de una placa extra, en este caso se utilizó un chip FTDI **FT232RL**. Para programarlo se conectan los pines GND, VCC, RX, TX y DTR del Pro Mini con los correspondientes del FT232RL y este por USB Mini a una

computadora.



(a) Arduino Pro Mini



(b) Placa FTDI FT232RL

Figura 5.20 Microcontrolador seleccionado y la placa accesoria necesario para programarlo

5.3.2. Diseño y desarrollo de Placa de Prueba

Se desarrolló una placa de prueba con el propósito de probar el sensor seleccionado sin estar conectado al *Blackberry*® [16] para el cual fue diseñado. Si bien se probó el funcionamiento del trackpad con la lengua con resultados exitosos, la misma se realizó sobre un móvil *Blackberry*® ya que no era posible hacer uso del sensor fuera del mismo, debido a que el conector hembra que utiliza es de montaje superficial y de contactos diminutos. A su vez, tener una placa de prueba fue extremadamente útil para probar el sensor, la comunicación Bluetooth y el funcionamiento general del dispositivo. Una vez determinado el sensor, el módulo de comunicación Bluetooth y el controlador se tuvieron los requerimientos electrónicos del sistema.

Planificación del diseño

	Tensión [V]	Regulador interno	Nivel de lógica [V]	Corriente [mA]
Arduino Pro Mini	3.3 - 12	Sí	3.3 o 5	19
HC-05*	5 - 6	Sí	3.3	25
BB8520	2.85	No	2.85	3(*)

(*): Estimada a partir de sensores conocidos

Tabla 5.7 Características electrónicas de cada componente.

En base a estos requisitos, inicialmente, se diseñó un PCB que únicamente contenga la electrónica necesaria para que el sensor pueda ser conectado al Arduino, desde la reducción de tensión al nivel indicado hasta los contactos necesario para soldar el conector del sensor y los pines para luego conectarlo al Arduino.

- **Regulador de Tensión:** Se optó por utilizar el LT1117-2.85 de Linear Technology. Este es un regulador de caída de tensión para proporcionar una corriente estable de salida de hasta 800mA a 2.85V. A una entrada de 5V tiene un consumo de corriente de 10mA. Este componente se utilizó para alimentar el sensor con la tensión indicada [26].
- **Traductor Bidireccional de 8 Bits:** Dado que el sensor opera a un nivel de lógica máximo de 2.85V, no basta únicamente con alimentarlo con esa tensión. La lógica interna está diseñada para funcionar a 2.85V por lo tanto se perdería información del sensor al Arduino o se dañaría el sensor cuando enviamos información del Arduino al trackpad. Es por esto que se optó por utilizar por un conversor lógico de 8 bits, el componente en cuestión es el TXB00108 de *Texas Instruments*. Se eligió este en particular debido a la cantidad de pines requeridos y a la lógica de 8 bits del trackpad en cuestión. Tiene un rango de tensión de entre 0 y 6.5V y se lo debe suministrar con las tensiones a convertir, en este caso con 2.85V y 5V, con un consumo continuo máximo de 50mA [27].

- **Capacitores de Desacople:** Se utilizaron única y exclusivamente para eliminar ruido de línea y de alta frecuencia para así estabilizar la conexión entre componentes. La alimentación del circuito nunca es perfecta y del todo estable y es importante que en comunicaciones entre circuitos lógicos que las señales sean lo mas precisas posibles para no tener perdida de información. El Arduino funciona con 5V, normalmente los voltajes por debajo de 2.5 V se leerán como una señal BAJA y las tensiones por encima de 2,5 V se leerán como una señal ALTA. Ruido en la señal podría implicar un desencadenamiento de señales BAJAS o ALTAS erróneas [28].

Definición del diseño

Se continuo con el diseño en si de la placa, el mismo se realizo en *Altium Designer* ® y se utilizo el complemento gratuito *SnapEDA* ® para obtener los esquemas y huellas de los componentes no incluidos en las librerías de *Altium Designer* ®, de este complemento se descargaron las huellas y diseños del regulador de tensión y del conector del sensor.

Se incluyeron en el esquemático todos los componentes mencionados anteriormente a excepción del conversor lógico, debido a su tamaño se optó por dejarlo fuera de la placa para facilitar armado, dibujo de pistas y reducir el tamaño de la primer placa significativamente. Luego se inicio con el dibujo de pistas y ubicación de los componentes dentro de la placa.

Desarrollo de la placa

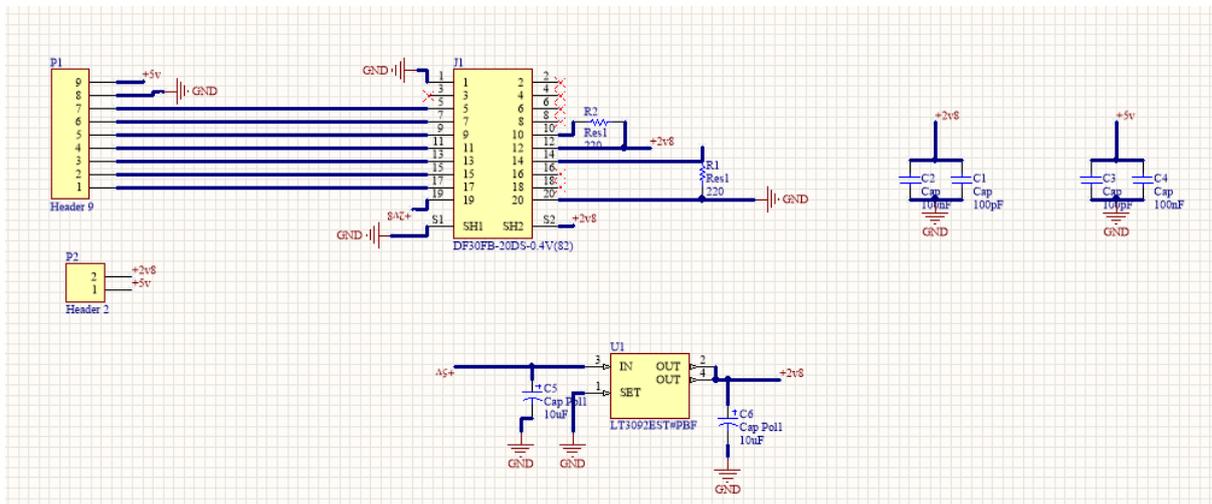


Figura 5.21 Esquemático de placa de prueba realizado en Altium Designer ®.

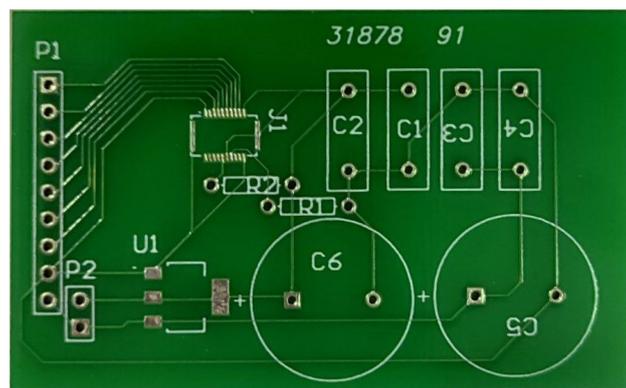


Figura 5.22 Placa de prueba inicial luego impresa

Debido al tamaño de los contactos del conector del sensor fue imposible fabricar la placa con métodos no industriales, por lo tanto se envió a fabricar un pequeño lote de estas placas luego de una rigurosa inspección de conexiones del circuito como también del espacio entre pistas y ubicación de los componentes para que todo se pueda conectar.

La mayoría de los componentes se soldaron relativamente fácil con un poco de estaño y un soldador genérico. En cambio, el conector del sensor fue todo un desafío teniendo que implementar técnicas nuevas. Utilizando un lupa, el conector se soldó primero colocando estaño es pasta y flux sobre los contactos de la placa y se fundió con un soldador de punta

fina. Luego, Con una pinza se colocó cuidadosamente el conector sobre los contactos ya estañados. Una vez alineado el conector sobre la huella, cuidadosamente se soldó, con el soldador, los terminales de los extremos del conector a la placa. Una vez con el conector fijo, se colocó estaño en pasta sobre el resto de los pines y sobre la pasta se inyectó más flux. Para lograr soldar la totalidad del conector se entregó calor con una pistola de aire caliente realizando movimientos circulares sobre el conector. Una vez derretido el estaño se removió el exceso del mismo cuidadosamente con el soldador de punta fina, el resultado se puede ver en la Figura 5.23b

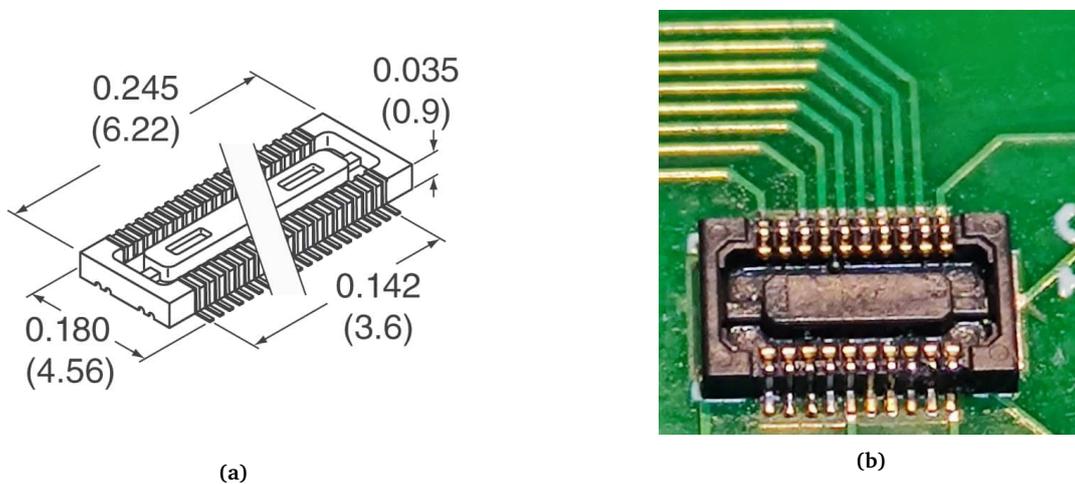


Figura 5.23 Izquierda, 5.23a: Medidas de conector en pulgadas (y en milímetros). Derecha, 5.23b: Conector hembra del trackpad BB8520 soldado a la placa de prueba.

Las primeras pruebas se realizaron con un Arduino MEGA ya que debido a su tamaño y mayor cantidad de pines fue más fácil conectar para realizar pruebas preliminares, por ejemplo, establecer comunicación con el sensor y realizar lecturas de los registros de movimiento.

Armado de placa complementaria

Una vez realizadas las primeras pruebas y saber con certeza que el sensor funcionaba, se decidió armar una placa accesoria. Esta placa se fabricó con el objetivo de utilizar toda

la electrónica en conjunto y realizar las primeras pruebas completamente inalámbricas, donde el conjunto estuviera conectado únicamente a la alimentación y pueda comunicarse con una computadora mediante Bluetooth. Se realizó un dibujo a mano de la misma y se utilizó una placa multiperforada, pines y cables para el armado de la misma.

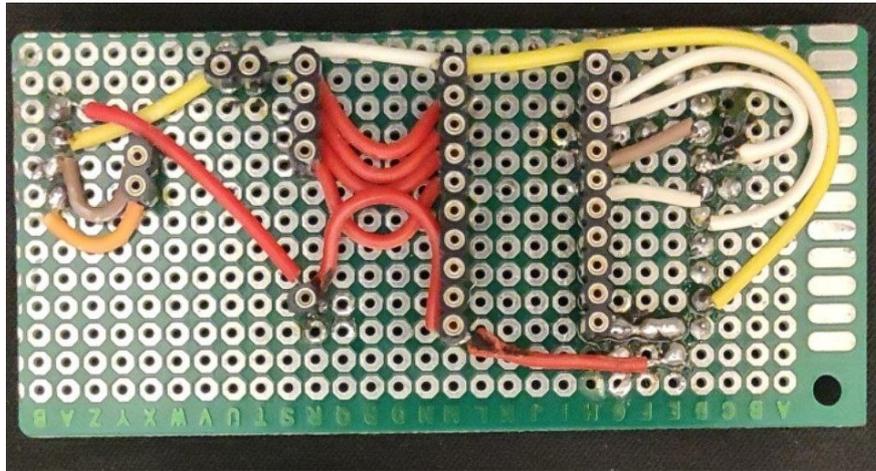


Figura 5.24 Placa complementaria finalizada. De izquierda a derecha se conectan, el modulo Bluetooth, el Arduino, el conversor lógico y por ultimo la placa del sensor

Sistema integrado de la tecnología

Finalmente se tuvo una primera implementación de la tecnología funcional, capaz de conectarse a un dispositivo móvil o una computadora y capaz de mover el cursor inalámbricamente. Es importante destacar que este sistema se desarrolló únicamente para probar si la implementación del sensor y de la comunicación eran factibles.

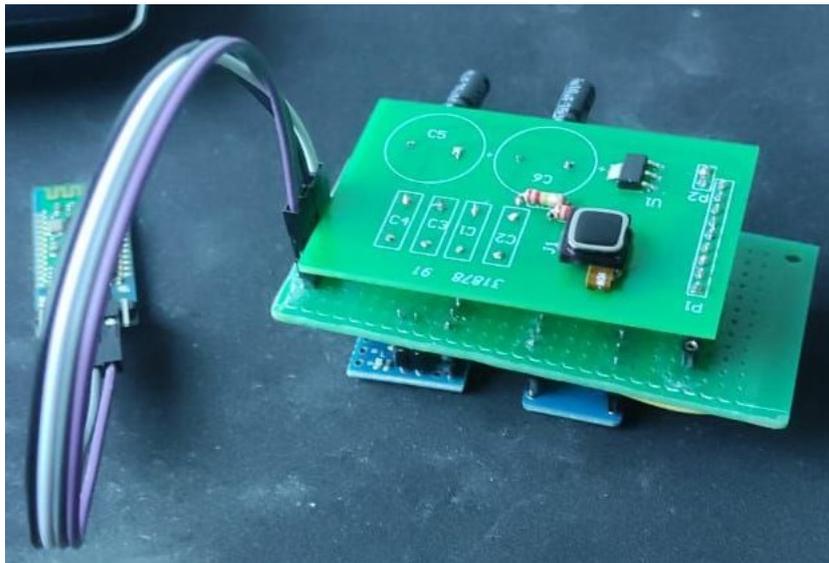


Figura 5.25 Primera integración funcional del sensor y comunicación en conjunto

5.3.3. Diseño y desarrollo de Placa Final

En base a las placas en la sección anterior se definió la implementación de alto nivel de la electrónica:

1. Bloque Alimentación: Batería y regulador de tensión
2. Bloque procesamiento: Microcontrolador, Modulo Bluetooth y electrónica del sensor, incluye conversor lógico, capacitores de desacople y regulador de tensión
3. Bloque Detección: Sensor y Click

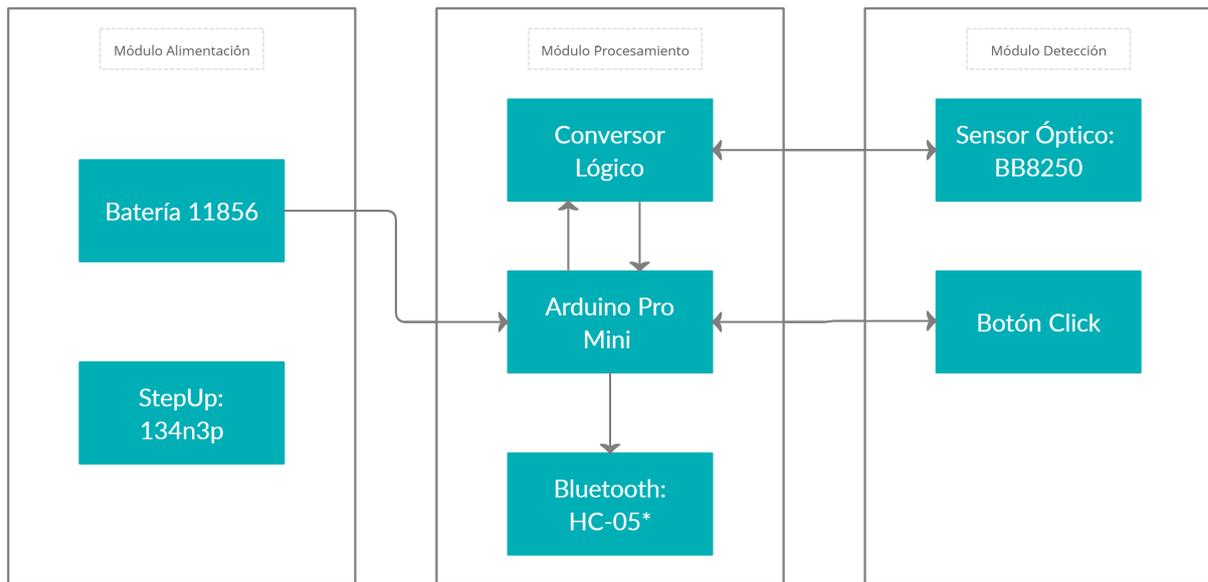


Figura 5.26 Implementación en bloques del sistema, cada componente separado en módulos con sus respectivos componentes.

Diseño de doble placa

La integración del sensor y comunicación permitió observar el espacio y volumen que ocupa la electrónica. Si bien se priorizaron aquéllos componentes de menor tamaño, no se contaba con la electrónica lo suficientemente pequeña como para tener un dispositivo que integre los tres módulos ya mencionados dentro, o en las cercanías de la boca y esto es sin que se tenga en cuenta el volumen de la batería. Por lo tanto, se decidió separar la electrónica en dos placas distintas. Una placa correspondiente al bloque de procesamiento y otra de tamaño considerablemente menor correspondiente únicamente al sensor. Ambas placas se conectan a través de 6 cables, 2 correspondientes a la alimentación del trackpad y 4 correspondientes a la interfaz SPI. Nuevamente, las placas se diseñaron en *Altium Designer*® y al tratarse de un primer prototipo se tuvieron en cuenta las siguientes características:

- **Orientación de los pines:** Al tener dos placas conectadas por varios cables, los pines correspondientes se orientaron de forma tal que la conexión entre placas sea directa y no haya posibles enredos ni forcejeos por parte de los cables. Si bien

se busco reducir el tamaño del cable y tener un conector mas robusto, al tratarse de una primera iteración se opto por usar pines rectos para facilitar la conexión y desconexión entre las placas, este descarto la utilización de conectores tipo JST o la utilización de cables planos.

- **Modular:** Al tratarse de un primer prototipo se considero fundamental la capacidad de poder extraer los componentes mas importantes de la placa en caso de falla o programación de los mismos. Si bien esto requiere de una electrónica mas voluminosa, habiendo separado el diseño en dos placas esto ya no era un restricción del lado del procesamiento y alimentación. Tanto el Arduino, el convertidor de lógica y el módulo Bluetooth se conectarían a la placa a través de pines típicos, donde el PCB hacia de zócalo para los módulos. Por el lado del sensor ya se contaba con un conector específico.
- **Incorporación del Click:** Se incorporó el circuito del click al PCB de procesamiento. El click por razones de tamaño no fue incluido en el PCB del sensor pero si en una placa accesoria, esto facilito la ubicación del mismo dentro del prototipo final y flexibilizó el diseño.
- **Alimentación de doble vía:** A la hora de diseñar estas placas no se había definido la alimentación del sistema, por lo tanto, la placa se diseñó teniendo en cuenta las dos posibles alimentaciones admitidas por el Arduino Pro Mini. Es posible alimentarlo a través del pin V_{CC} con una tensión regulada de 5V o también a través del pin V_{RAW} con una tensión sin regular de entre 6V a 12V. El cambio entre alimentaciones estaba dado por un jumper incluido en la placa.

Características adicionales

Adicionalmente, se agregaron características que todo dispositivo de estas condiciones suele tener, estas características no necesariamente mejoran el funcionamiento particular del dispositivo, por ejemplo, la detección de movimiento, pero si agregan a su robustez y

su funcionalidad:

- **Ground Plane:** Método de diseño común en circuitos comerciales. Se deja un plano de la placa como Tierra (GND) esto ayuda a la hora de diseñar y fabricar la placa ya que agiliza el proceso al haber menos material que se debe deshacer. Además, reduce posibles ruidos que el circuito pueda sufrir [29] y al haber mayor área de cobre asegura que no haya caídas significativas de tensión, es decir, garantiza que la conexión a tierra de todos los componentes tenga el mismo potencial de referencia.
- **Batería:** Se incluyó una vía específicamente para medir la tensión de la batería y poder determinar o aproximar su carga. Se utiliza el conversor analógico digital de 10 bits del Arduino Pro Mini junto con un divisor resistivo, incluido en caso que sea necesario. Se agregó la posibilidad de implementar un divisor en caso de que la batería seleccionada para alimentar el circuito sea mayor a los 5V de nivel de lógica que utiliza el Arduino.
- **Indicador LED:** Se incluyó en un pin la posibilidad de conectar un LED, de esta manera se podría visualizar si el dispositivo esta encendido y hasta indicar cuando la batería es baja.

Desarrollo de ambas placas

Nuevamente las placas se enviaron a fabricar y se soldaron con métodos convencionales a excepción del conector del trackpad que se soldó con el método ya mencionado. Se realizaron las pruebas al igual que con la placa de prueba y el resultado fue el mismo.

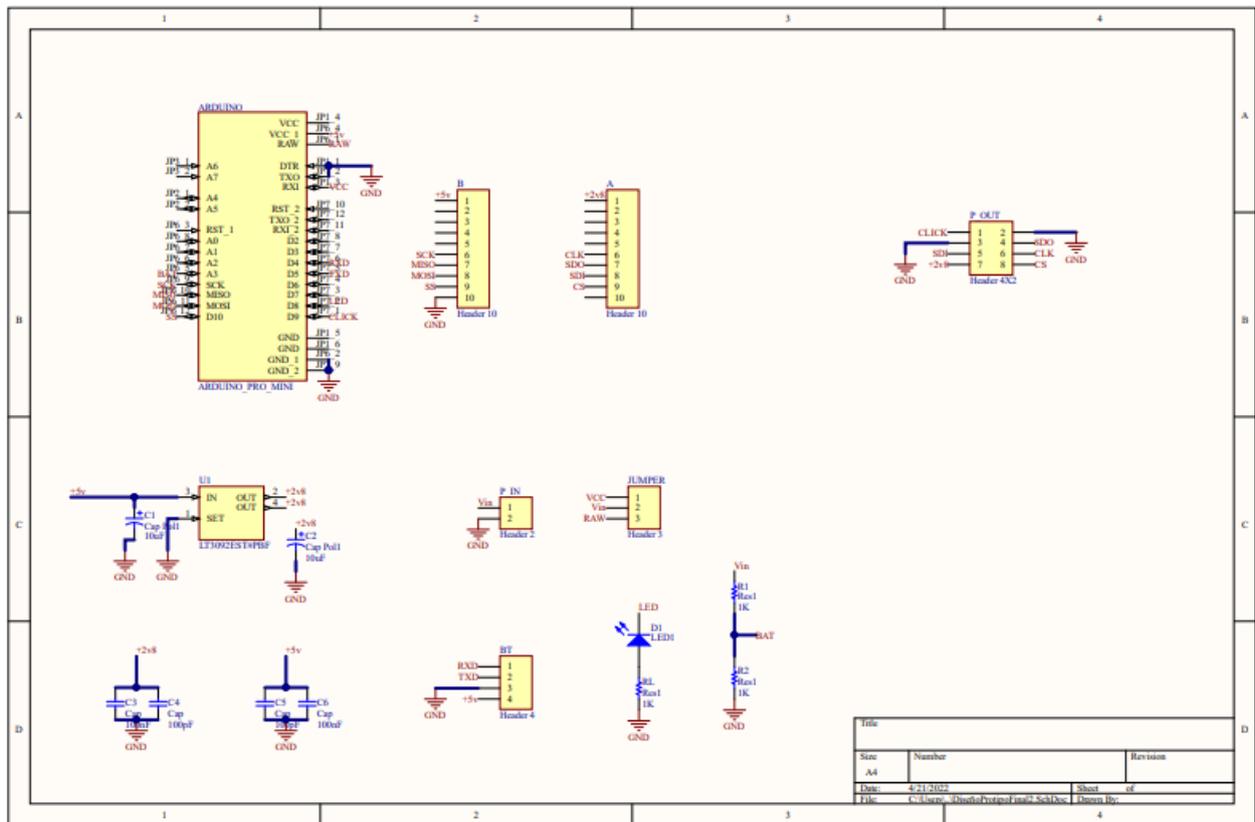


Figura 5.27 Esquemático correspondiente a la placa de procesamiento

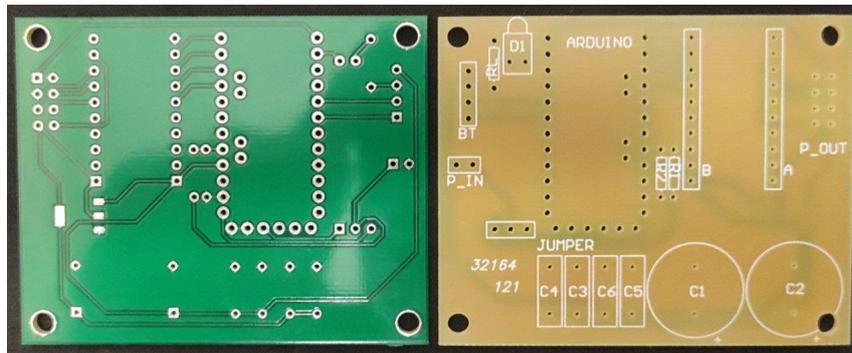


Figura 5.28 Placa procesamiento (dorso y revés) sin soldar.

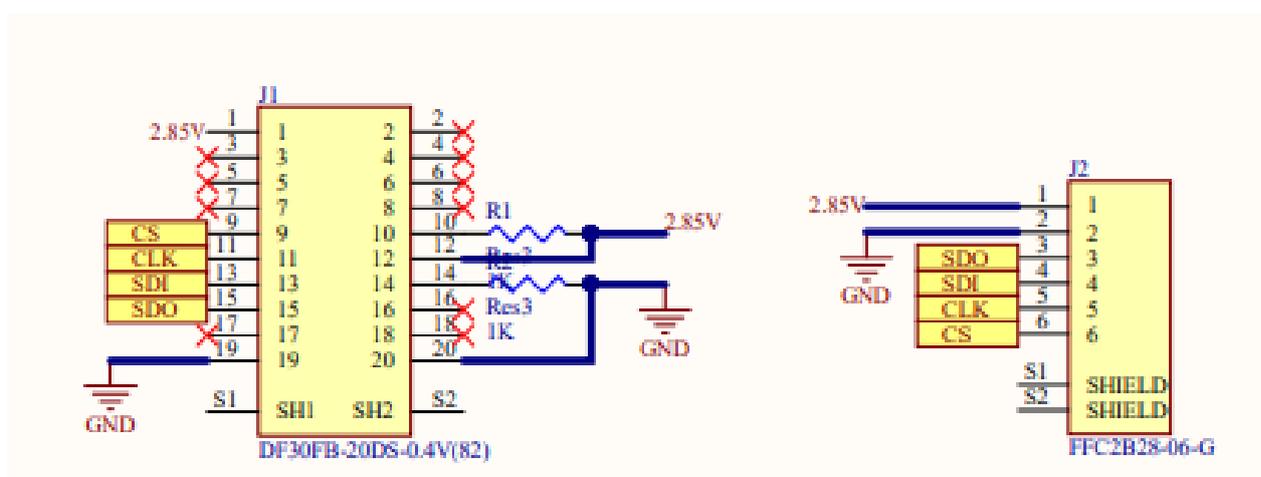


Figura 5.29 Esquema correspondiente a la placa del sensor

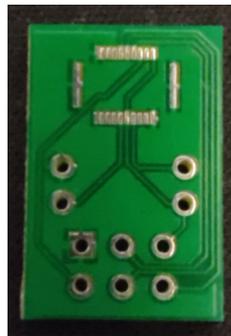


Figura 5.30 Placa sensor impresa sin soldar.

5.3.4. Alimentación

Una vez diseñada la placa final, se debía definir la alimentación utilizada en el dispositivo. El circuito se diseñó con la alimentación de todos los componentes proveniente desde el Arduino dado que el mismo soporta una carga máxima de 40mA por pin, por lo tanto, basta con alimentarlo como para que el circuito entero funcione [24]. Según lo requerido por el Arduino se debe administrar el circuito con 5V regulados o más de 6V sin regular. Un requerimiento no obligatorio pero fundamental es que al tratarse de un dispositivo inalámbrico es una gran ventaja que el mismo sea recargable.

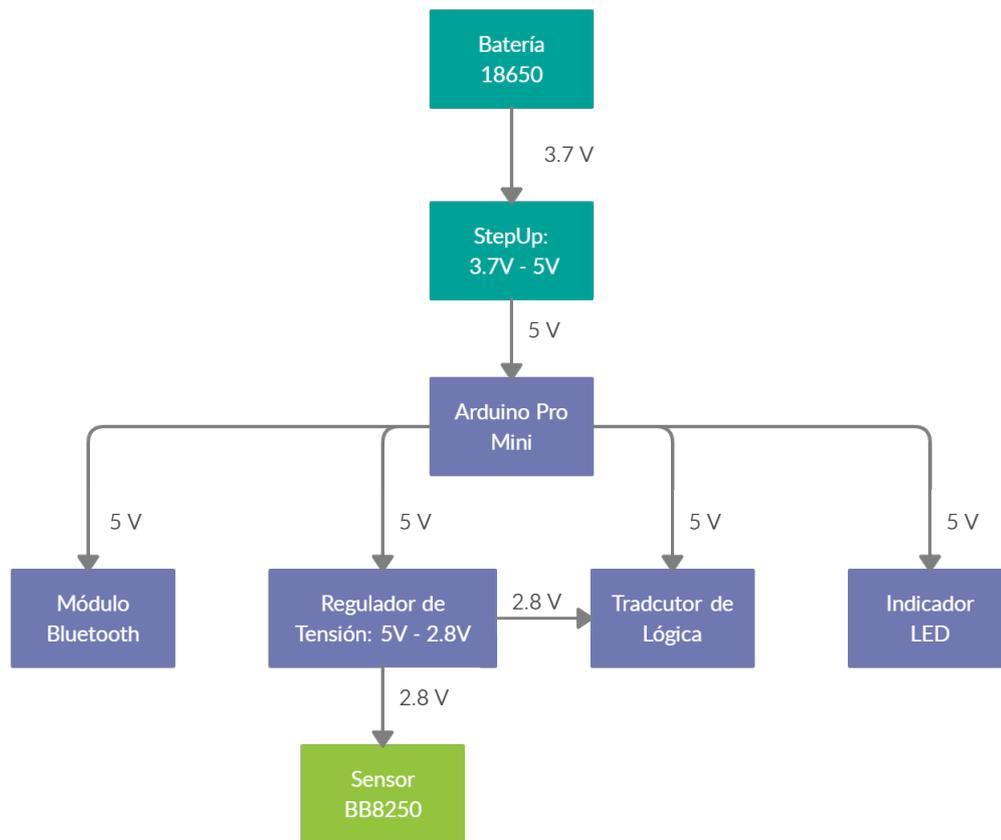


Figura 5.31 Diagrama de bloques de alimentación del sistema.

Se analizaron múltiples opciones, sin embargo se terminó definiendo entre estas tres posibilidades:

- Dos pilas en serie de 3.7V para tener así 7.4V sin regular: Tener dos pilas en serie aumenta el peso considerablemente y no aumenta la carga del mismo. Se tendría un dispositivo de igual carga que con una batería pero mas pesado
- Una batería de 9V: Similar a lo anterior, batería pesada y con poca carga. Un dispositivos e estas características no suelen ser alimentados por estas baterías.
- Un módulo step-up que aumente la tensión de 3.7V (u otro valor comercial por debajo de 5V) a 5V regulados: Agrega complejidad al circuito adicional a el modulo de recarga de baterías

Se optó por el módulo step-up, ya que las dos primeras opciones proveen tensión excesiva para la necesidad del dispositivo y agregan peso innecesario. Al analizar brevemente distintos modelos de módulos de estas características, se eligió el módulo step-up **134n3p** que además tenía el circuito integrado para recargar baterías. El módulo tiene una tensión de entrada y salida de 3.7V y 5V respectivamente, corriente de salida máxima de hasta 1A y una eficiencia de descarga del 85%. Esto significa que a una salida máxima de 1A (o 5W) se tiene una corriente de entrada de 1.6A (o 5.9W). En cuanto a la batería, sin utilizar el módulo step-up, se midió la corriente del dispositivo alimentando con 5V regulados utilizando un tester. Se realizaron dos mediciones:

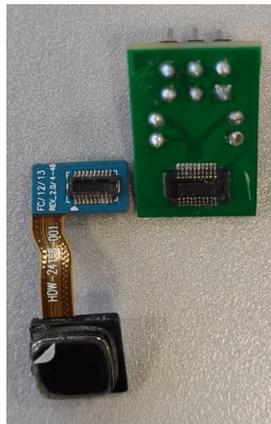
- **Tensión:** Agregó una resistencia de $1k\Omega$ al terminal positivo del dispositivo y se midió la caída de tensión sobre la misma. Luego se obtuvo la corriente dividiendo la tensión medida sobre el valor de la resistencia.
- **Corriente:** Se midió directamente la corriente del dispositivo conectando un tester en serie entre la alimentación y el dispositivo modo de medición de corriente.

En ambos casos la corriente fue cercana a 70mA, por lo tanto, el step-up debería tener de entrada 3.7 y 115mA aproximadamente. Es decir, que para obtener una duración de funcionamiento continua de 30 horas se debería utilizar una pila de aproximadamente 3360mAh. Finalmente, si bien, la duración de batería del dispositivo no es de suma importancia para este primer prototipo, se eligió una pila del tipo 18650 con una capacidad de 8.800mAh con una tensión de 3.7V de UltraFire®. Si los valores declarados por el fabricante son reales, esto implicaría una duración del mas del doble de la mencionada.

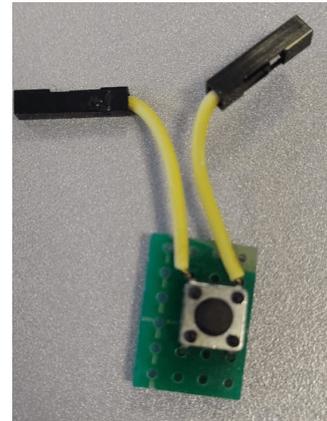
Resumiendo la alimentación, la pila elegida se encuentra conectada al módulo step-up, y la salida, en esta instancia, es de 5V que son la entrada del Arduino. Luego, el Arduino alimenta al módulo Bluetooth, al conversor lógico y al regulador de tensión, Por último, el regulador de tensión alimenta al sensor.



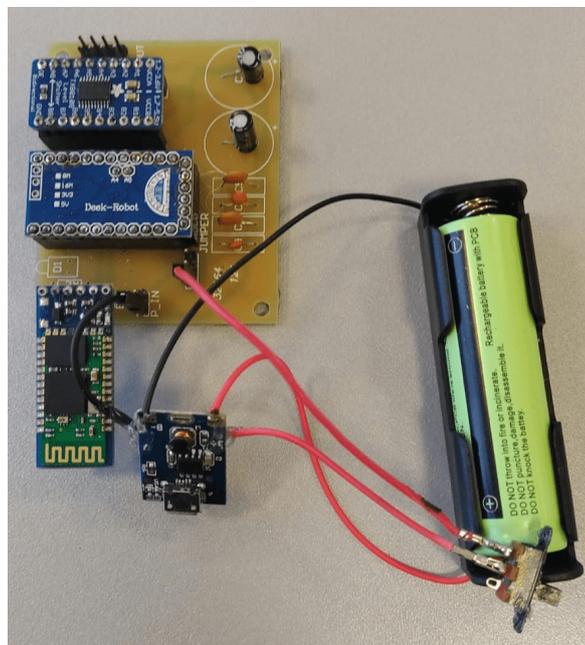
(a) Placa de sensor con trackpad desconectado.



(b) Placa sensor finalizada



(c) Placa del botón para realizar el click.



(d) Placa de procesamiento finalizada junto al modulo de alimentación.

Figura 5.32 Arriba (5.32a, 5.32b, 5.32c): Modulo de sensado finalizado. Abajo (5.32d): Modulo de procesamiento y alimentación finalizados

5.4. Software

La restricción más grande con la que se contó durante el diseño de software fue que el dispositivo cuenta únicamente con un botón, el cual debía ejercer todas las funciones de los distintos clicks del ratón, o al menos las básicas.

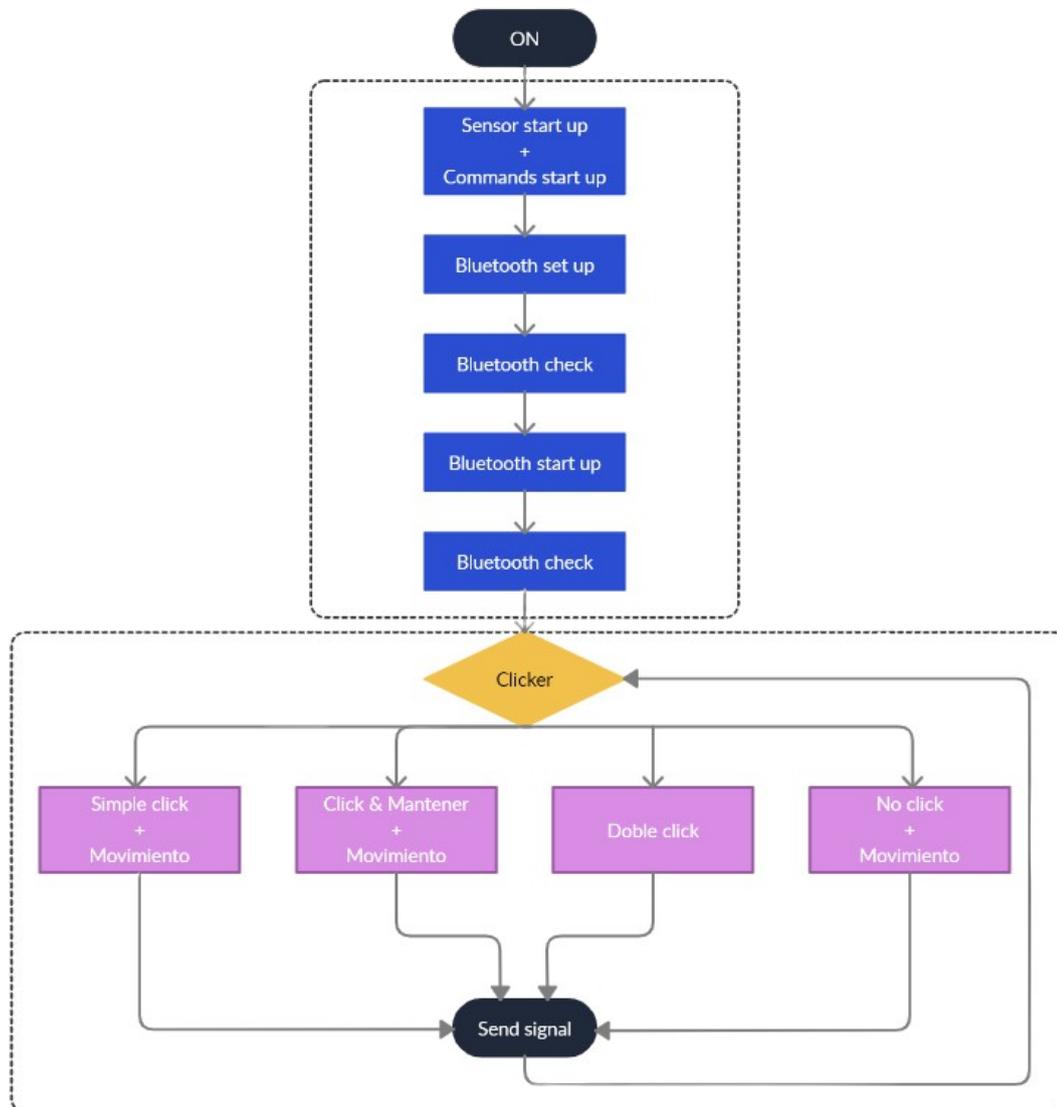


Figura 5.33 Diagrama de flujo del software implementado

Como todo Arduino, el código está dividido en la función `setup()` y el `loop()`. En la primera se definen tanto las configuraciones como las variables iniciales y se ejecuta ni bien se resetea o enciende el Arduino. La segunda se ejecuta en lazo y dentro de ella corre el código principal del Arduino. Todas las líneas de código de Arduino y por lo tanto las funciones que contenga se ejecutan en cascada.

5.4.1. Configuración Inicial

Dentro de `setup()` se establece la conexión con el sensor óptico y se inicializa y configura el módulo Bluetooth.

En primer lugar, se inicia la comunicación con el sensor y se comprueba si el sensor se encuentra en su correcto funcionamiento. En este paso primero se configura e inicializa la comunicación SPI definiendo los 4 pines necesarios (CS, MISO, MOSI, SCLK), el orden significativo de los bits (bit más significativo primero) y el modo de comunicación SPI (Modo 3), luego se ejecuta la función “**com_start()**”, que resetea el pin NCS para inicializar la comunicación con el sensor, esto si bien no es necesario hacerlo todo el tiempo aporta redundancia al sistema y a fin de cuentas asegura una conexión exitosa. Para comprobar la conexión se necesita que el la dirección de memoria `PROD_ID` leída sea igual al registro contenido en esa dirección, es decir, el número `0x0D` (13), si esto es así significa que el mismo se encuentra conectado correctamente.

En el siguiente paso, se ejecutan tres funciones que comandan al módulo Bluetooth:

- **BT_setup()**: Inicia el puerto serie que comunica al Arduino con el módulo Bluetooth. Se configura el módulo bluetooth para que siempre se encuentre en modo HID que permite que sea reconocido por otros dispositivos como un Mouse y se establece el modo esclavo
- **BT_Check()** Verifica si hay enlace con algún dispositivo, si no la hay se ejecuta `BT_startup()`
- **BT_startup()**: Intenta conectar el prototipo con el primer dispositivo que encuentre siempre y cuando este ya haya sido conectado previamente, aporta redundancia al sistema a cambio de tiempo de inicialización

Una aclaración importante es que, una vez ya conectado a un dispositivo maestro, no es posible modificar ninguna de las características al módulo BT desde el Arduino (por ejem-

plo, si este no se encuentra en modo *Command*, el Arduino no puede intentar modificarlo), solo es posible enviar comandos.

5.4.2. Lazo

Dentro del lazo corre la función **Clicker()**, esta puede detectar cuatro estados, como se ve en la figura 5.33, estos son: click simple + movimiento, click y mantener + movimiento, doble click y no click + movimiento. El quinto se da cuando no se ejecutan ninguna de las opciones, no hay, ni cambios en el estado del botón ni movimiento, es decir no ocurre nada. Se definen variables como tiempos de click y distintos tipos de click, se puede entender este script como una máquina de estados. A continuación se detallan las funciones:

- **getClick():** Lee y devuelve el estado del click
- **clicker():** Función principal del lazo de Arduino, funciona en base al estado del click y al tiempo en que este esté presionado. La función lee el estado del click y guarda el estado anterior del click cronometrando cada vez que cambia el estado del mismo. Al correr en loop lo primero que realiza esta función es la lectura del estado actual del click:
 - Escenario 1: Estado actual del click presionado (1) y estado pasado sin presionar (0), se presionó el click. Arranca el cronómetro y cambia el estado pasado a presionado
 - Escenario 2: Estado actual del click presionado (1) y estado pasado presionado (1). Este estado depende del tiempo del cronómetro y permite mover el mouse con el click presionado y eventualmente si se dan las condiciones (movimientos no detectados y 4 segundos de botón presionado) realiza un click derecho
 - Escenario 3: Estado actual del click 0 y estado pasado 1. Escenario que detecta que el botón se dejó de presionar

- Escenario 4: Estado actual del click 0 y pasado 0, se utiliza este escenario para enviar el comando de click a través del modulo HC-05*, teniendo en cuenta el cronometraje necesario para la posibilidad de un segundo click
- Escenario 5: click sin presionar y movimiento en el sensor detectado, permite el movimiento del mouse sin haber acción del click

Cabe destacar que siempre que se realiza una lectura de los registros de movimiento se corrobora que efectivamente dicho movimiento exista según el registro de detección de movimiento explicado anteriormente.

En base a estos escenarios podemos determinar los estados previamente mencionados:

- Click simple y doble click dependiendo de la necesidad del usuario
- Click derecho si se mantiene presionado el pulsador por más de 4 segundos y no hay movimientos del mouse
- Click & drag: Click presionado y movimiento del sensor
- Movimiento sin click

Para facilitar el desarrollo y organización, el código se separó en 3 archivos headers (.h) más el script propio de Arduino (.ino). Cada header contiene las funciones y variables necesarias de cada módulo, sensor, HC-05* y click. A su vez, se introdujeron señales dentro del código las cuales funcionan como debuggers muy útiles para verificar su correcto funcionamiento y arreglar errores dentro del mismo. Cabe mencionar que se utilizó GitHub, el cual permite realizar seguimiento de las distintas versiones del código, tener el código en la nube y cuenta con branches, una herramienta muy útil que permite masificar el código. De esta manera se pueden tener distintas versiones sin perder la anterior. En este caso esta opción fue útil a la hora de desarrollar el click, ya que, se tenía un programa funcional y se ramificó para realizar el click y no romper el código ya escrito.

5.5. Gabinete

El diseño de un gabinete o una carcasa que encapsule la electrónica y a la vez sea funcional es tan importante como la electrónica en sí. No solo contiene toda la electrónica si no que aporta a la funcionalidad para que el dispositivo sea utilizado en la boca y funcione correctamente. En un primera instancia se trabajó en el diseño del gabinete a la par que el diseño de la electrónica al no saber el sensor ni el tamaño de los componentes a utilizar.

5.5.1. Definición del gabinete

Se tuvieron en cuenta distintas implementaciones al realizar el primer prototipo:

- **Paladar:** Armar un dispositivo integrado que se ubique por debajo del paladar y de alguna manera sensar el movimiento de la lengua sobre el mismo, el click estaría dado al presionar la lengua sobre el dispositivo y apretar el botón de click, se podía dividir el paladar en secciones para agregar aún más interruptores. [30]
- **Mordillo:** Diseñar un dispositivo el cual el usuario pueda morder y desplazar la lengua sobre la cara del mordillo dentro de la boca, el click estaría dado por uno o dos botones al ejercer fuerza con la mandíbula sobre el mordillo.
- **Arco a lo largo de la dentadura:** Similar a un protector bucal, se capta el movimiento de la lengua a lo largo de los dientes, ya sea con distintos botones o sobre la superficie del dispositivo. Podría incluir más de dos botones para realizar distintas funcionalidades. [31][32]
- **Diseño fuera de la boca:** Colocar el dispositivo fuera de la boca y aproximarlos a la misma con un soporte similar al que utilizan las armónicas. Implementación más fácil entre todas las opciones pero no la ideal ya que supone un mayor esfuerzo por parte del usuario.

Inicialmente se optó por un diseño por debajo del paladar ya que es la implementación ideal para un dispositivo de estas características, completamente portátil, inalámbrico, compacto, de fácil uso, funcional y tiene una mayor área de sensado con respecto a los otros diseños. Así como es la solución ideal también es la de mayor dificultad de implementación, el diseño debe incluir el sistema entero en un volumen considerablemente menor al del paladar, no todos los paladares son iguales por lo tanto el dispositivo final se determina con la forma paladar del usuario, la implementación es íntegramente dentro de la boca por lo que la estanqueidad es un factor importantísimo a tener en cuenta. Sobre este diseño se avanzó en la obtención de un molde de yeso de una mandíbula superior y la reconstrucción digital de la misma a través de fotogrametría, una técnica de reconstrucción digital de objetos a través de fotografías del mismo, para así comenzar el diseño en SolidWorks. Lamentablemente, al definir el controlador y el módulo de comunicación Bluetooth, se tuvo que cambiar el diseño ya que no era posible caber todo dentro de la boca.

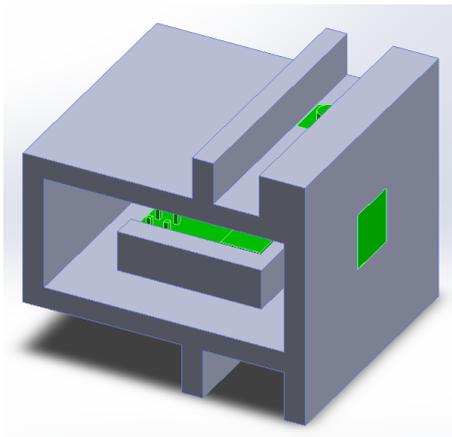
Finalmente se optó por el mordillo acompañado del diseño de doble PCBs, este tiene una implementación más simple, permite tener la electrónica más voluminosa por fuera de la boca, la estanqueidad del dispositivo es más flexible y simple ya que los lugares por donde puede filtrar agua son menores así como la posibilidad de producir alguna descarga en el usuario. Como desventaja el diseño estará dividido en dos partes, disminuyendo la practicidad del mismo, el modelo debe ser diseñado teniendo en cuenta la mandíbula y mordedura de cada usuario, el material debe soportar la tensión constante al ser mordido, más aún cuando el click se acciona de esta forma y es posible que el uso prolongado de un dispositivo de estas características produce más fatiga que uno por debajo del paladar o por fuera de la boca. Los componentes no incluidos dentro del mordillo se colocan en una caja a parte que se sujeta al brazo del usuario.

5.5.2. Diseño de gabinete

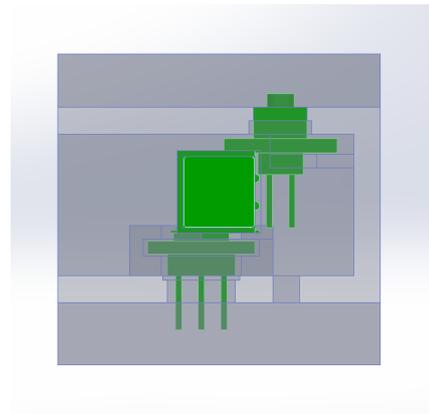
Primera iteración

Para obtener una primera aproximación de modelo a diseñar se armó una pequeña caja de cartón, con el objetivo de obtener las medidas iniciales del mordillo y la ubicación de la electrónica dentro de la misma. El armado de esta caja permitió definir la ubicación del sensor en la cara del mordillo ubicada dentro de la boca, el largo aproximado de 30 milímetros necesario para ubicar el sensor y la placa del sensor, la disposición del conector de la placa del sensor y, más importante, el alto aproximado de 20 milímetros de la caja.

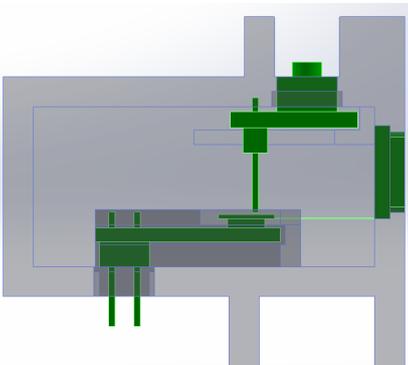
Una vez definida la ubicación de la electrónica y las medidas necesarias, se realizó un primer bosquejo del modelo, se pudo definir la forma en que se ensambla y la ubicación del click. El modelo consta de una caja en donde toda la electrónica se desliza o engancha dentro de la misma. El botón del click se ubica hacia la derecha del centro de la mandíbula superior por encima del sensor fijado. Esta primer evolución se diseño en SolidWorks® y se pueden observar en la Figura 5.34, para ello se tuvieron que modelar los componentes individuales para poder visualizar la pieza modelada.



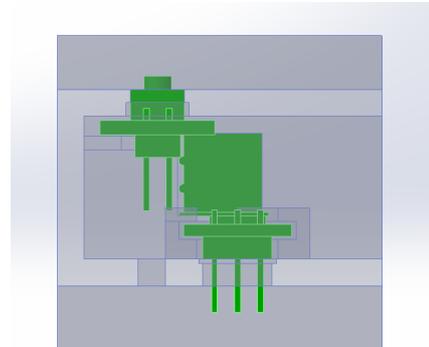
(a) Vista isométrica de la primera evolución del diseño



(b) Vista de frente del diseño, caja transparentada



(c) Vista de perfil del diseño, caja transparentada



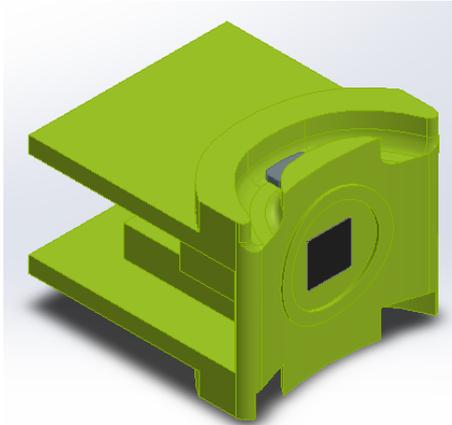
(d) Vista de atrás del diseño, caja transparentada

Figura 5.34 Distintas vistas de la primera evolución del diseño, los componentes se visualizan en verde y la caja en gris

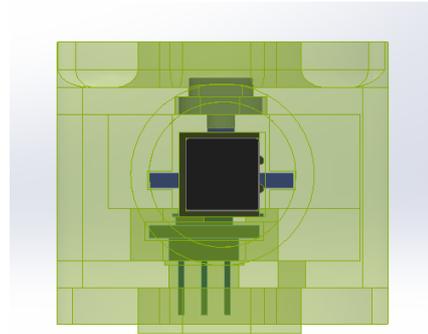
Segunda iteración

Se optó por continuar el diseño en Siemens NX® al tener más herramientas que facilitaron el diseño final y requerir menos especificaciones de la computadora en donde se diseñó el mordillo. En esta evolución se definió curvatura inferior y superior del mordillo, a partir de las medidas obtenidas del molde de yeso fabricado para realizar la reconstrucción por fotogrametría. A su vez esto permitió darle curvatura a la cara frontal del mordillo para minimizar el espacio que ocupa esta cara dentro de la boca, además, para que la cara del sensor acompañe a la curvatura de la dentadura. Se delimitó el área del sensor con un anillo con relieve para ubicar fácilmente el sensor con la lengua y delimitar más eficientemente la zona del sensor. Por último se definió el diseño del botón utilizado para el

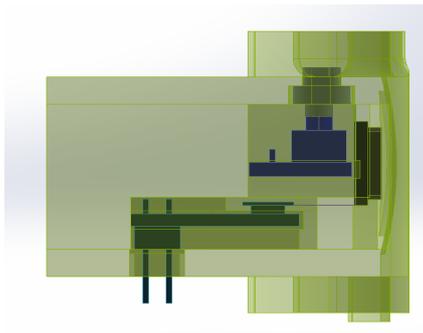
click, el mismo es una pieza extra que se coloca en una abertura propia del mordillo. El botón aumentaba la apertura de la boca por lo tanto se buscó minimizar la altura total del diseño, resultando en una apertura máxima de 22 milímetros con botón incluido.



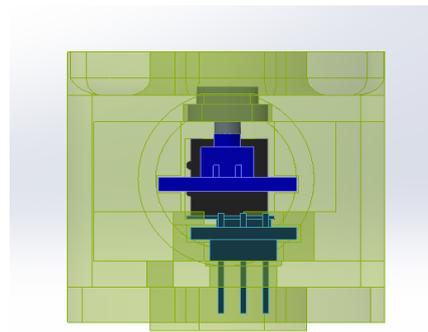
(a) Vista isométrica de la segunda evolución del diseño



(b) Vista de frente del diseño, caja transparentada



(c) Vista de perfil del diseño, caja transparentada



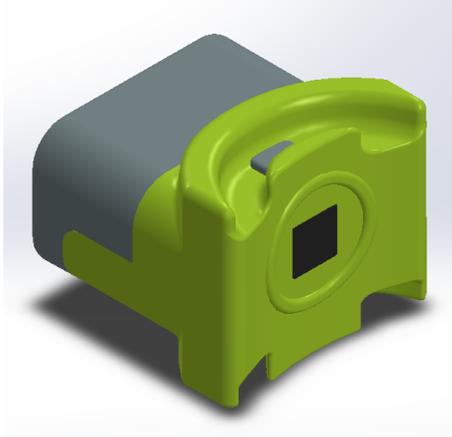
(d) Vista de atrás del diseño, caja transparentada

Figura 5.35 Distintas vistas de la segunda evolución del diseño

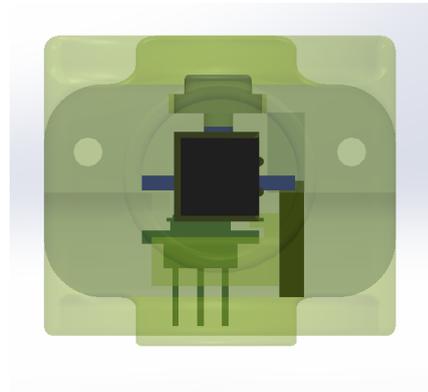
Tercer iteración

En la última evolución del diseño se definió la sección trasera del mordillo, el mismo se cierra con un tapa que fija los componentes electrónicos en su lugar sujeta por dos tornillos. Para no aumentar el largo total del mordillo se aprovechó la flexibilidad del cable flex que utiliza el sensor, esto permitió desplazar el PCB del sensor hacia adelante y en consecuencia la posición de la tapa trasera. Además, para facilitar la impresión en 3D y mejorar la calidad se optó por un diseño sólido minimizando los espacios huecos,

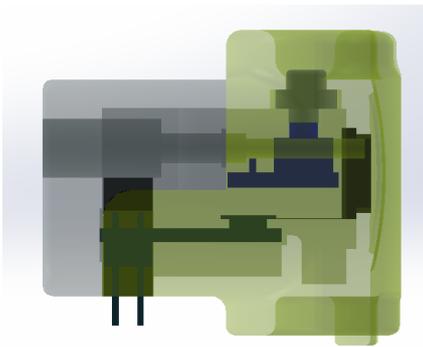
esto evita posibles deformación a la hora de la solidificación del material. Por último se realizaron las terminaciones generales eliminando la mayor cantidad de bordes rectos.



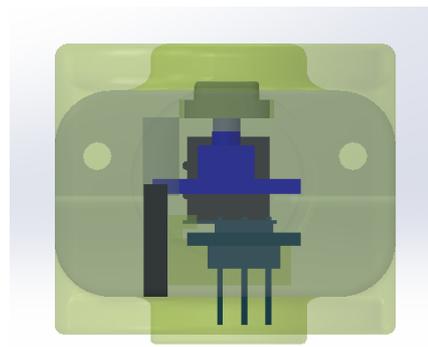
(a) Vista isométrica de la tercer evolución del diseño



(b) Vista de frente del diseño, caja transparentada



(c) Vista de perfil del diseño, caja transparentada



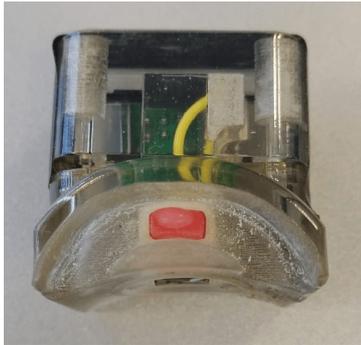
(d) Vista de atrás del diseño, caja transparentada

Figura 5.36 Distintas vistas de la tercer evolución del diseño

5.5.3. Impresión y ensamblado

El diseño se realizó teniendo en cuenta la ergonomía y simpleza del dispositivo con las limitaciones impuestas por el tamaño y características de los componentes utilizados. La impresión final se realizó en polyjet debido a la disponibilidad de material a disposición, esto resultó ideal debido a la transparencia del material, útil para observar el interior del dispositivo. Una vez impreso el modelo, el ensamblaje del mismo se realiza dando vuelta el mordillo y colocando el botón para el click, luego se desliza la placa del sensor con el trackpad conectado y se encastra a presión el trackpad en la ranura correspondiente,

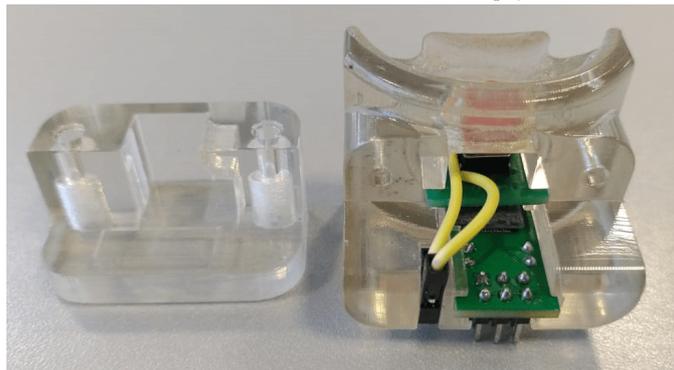
deslizando de la misma forma se coloca la placa que contiene el botón para el click y se acomodan los conectores de la misma en su respectivas ranuras. Por último, se coloca la tapa trasera y se conecta el mordillo a la placa que contiene el Arduino. De esta forma se tiene el primer prototipo o prueba de concepto funcional del dispositivo propuesto.



(a) Mordillo ensamblado, electrónica incluida



(b) Maridillo finalizado incluyendo los cables de conexión (se encuentra apoyado sobre la cara superior).



(c) Vista trasera sin tapa del diseño.

Figura 5.37 Distintas vistas del ensamble final del diseño.

Capítulo 6

Pruebas y resultados

A lo largo del desarrollo se realizaron distintas pruebas con el objetivo de determinar y justificar el descarte o la elección de cada componente. Además de la determinación de componentes, las pruebas ayudaron al desarrollo de software, ya que, se tuvo que desarrollar un código particular para cada prueba, distintas partes de cada código realizado constituyeron gran parte del código final. Las pruebas se realizaron, en su mayoría, utilizando un Arduino MEGA 2650, debido que al momento de realización de las pruebas no se había determinado el controlador y el tamaño y cantidad de pines de la placa facilitaba la realización de las mismas.

6.1. Comunicación

La prueba de comunicación se dividió en dos etapas, cada etapa fue clave a la hora de validar y determinar el éxito de la modificación realizada y la configuración del módulo bluetooth y el funcionamiento del mismo. La prueba se realizó con el fin de obtener un módulo Bluetooth capaz de ser configurado con un perfil HID y por lo tanto ser reconocido como periférico. La primera etapa se realizó luego de cambiar el firmware del módulo HC-05 por el del RN42, se conectó el módulo con el Arduino y se observó que el patrón de

encendido de los LED integrados en el módulo había cambiado acorde al funcionamiento del módulo RN42 y se logró comunicación entre el módulo y el Arduino enviando comandos ya mencionados propios del RN42 y obteniendo la respuesta esperada en el puerto serie del Arduino. Según el éxito de la primera etapa se procedió a configurar el perfil HID en el módulo modificado según los comandos detallados anteriormente en la sección 5.2. La segunda etapa consistió en lograr comunicar el módulo modificado con una computadora y mover su cursor. Con el módulo ya configurado se escribió un script de Arduino para que el HC-05* envíe los comandos necesarios para que el cursor de la computadora se mueva en círculos. El éxito de la prueba está determinado por el éxito de cada etapa y finalmente por poder desplazar el cursor de una computadora a través del módulo.

Los resultados fueron exitosos, se repitió el proceso en un segundo módulo para tener un reemplazo y realizar pruebas en paralelo. El código utilizado para configurar el módulo permitió experimentar con las distintas configuraciones del mismo hasta obtener la configuración seleccionada para el prototipo

6.2. Sensores

Se realizaron pruebas con los distintos sensores planteados, cada sensor se evaluó de forma similar teniendo en cuenta sus características. Las pruebas consistieron en utilizar cada sensor con la lengua o en condiciones de humedad similares y poder mover el cursor en la pantalla de una computadora. Como cada sensor funciona de forma diferente, en caso de ser necesario, se desarrolló un script particular para cada uno de los sensores.

Para el criterio de aprobación se tuvo en cuenta el tamaño de cada sensor, la facilidad de movimiento del cursor primero con las manos y luego con lengua.

6.2.1. Trackball

Se utilizó un modelo de prototipado compatible con Arduino. Un modulo de tamaño considerable comparado al tamaño real del sensor, esto es por la cantidad de pines requeridos para que el mismo funcione. La tecnología es intuitiva pero se hizo imposible utilizarla con la lengua ya que se moja fácilmente el mecanismo interno del sensor y al utilizarlo prolongadamente deja de funcionar. El sensor no superó el criterio de aprobación.

6.2.2. Joystick

Tamaño considerable, imposible de adaptar para su uso dentro de la boca. Difícil uso con las manos, requiere de un ajuste fino de la sensibilidad y aceleración del cursor luego de la obtención de la señal. El uso prolongado al ser utilizado con la lengua provoca fatiga. El sensor no superó el criterio de aprobación.

6.2.3. Botones

Tamaño adaptable para su uso dentro de la boca. Uso con los dedos intuitivo, requiere aceleración del cursor para facilitar su uso. Su uso con la lengua resulta poco intuitivo y lento al tener que cambiar constantemente entre 4 botones para poder desplazar el cursor en pantalla. El sensor no superó el criterio de aprobación.

6.2.4. Pad Capacitivo

La prueba se realizó sobre una computadora portátil, primero se humedeciendo el pad y luego se lo utilizo con la lengua. Si bien su uso fue el más intuitivo y fácil de todos los sensores probados hasta ahora, la humedad y el agua hicieron de su implementación y uso con la lengua imposibles en el largo plazo. El sensor no superó el criterio de aprobación.

6.2.5. Trackpad

Se probó el trackpad de un teléfono *Blackberry*® Bold 9900. Se utilizó con la lengua, resultó igual de intuitivo a utilizarlo con los dedos, con la desventaja que requiere acostumbrar y posicionarla de forma tal que la lengua apoye completamente sobre el sensor. Es posible implementarlo dentro de la boca, es pequeño y viene ensamblado de forma tal que minimiza el problema de estanqueidad. Al ser un sensor óptico, su superficie no se ve afectada por el agua. El sensor superó el criterio de aprobación.

6.3. Trackpad en placa de prueba

Se probó el correcto funcionamiento tanto de la placa de prueba, como del trackpad y por lo tanto se verificó que la técnica de soldado utilizada era exitosa. Una vez ensamblada la placa, la prueba consistió en conectar la misma al Arduino utilizando el conversor de lógica y obtener lectura de ciertos registros del sensor, por ejemplo el registro de ID del sensor, el de movimiento y los que guardan los movimientos en X e Y.

El criterio de éxito de esta prueba está dado por la correcta lectura de cada registro y que cada lectura sea coherente al registro leído. Por ejemplo, la lectura del registro del ID del sensor debe ser siempre el mismo número, y la lectura de los registros de movimientos en X y en Y deben ser coherentes al movimiento del pulgar sobre el sensor.

La prueba fue exitosa, no solo se comprobó el correcto funcionamiento del sensor y de la placa de prueba, también, se logró determinar el modo de funcionamiento de la interfaz SPI utilizado por este sensor y la obtención de distintos registros del mismo. Debido al resultado de la prueba, sobre esta placa se desarrolló gran parte del código relacionado al sensor además del ajuste fino de la comunicación SPI.

6.4. Software

El código se realizó de forma modular e iterativa. A la hora de integrar cada modulo e implementar el click, una vez que se tenía un código funcional se probó la estabilidad del mismo. Para esto se utilizó un joystick, ya que no se contaba con un prototipo final y se quería verificar únicamente el funcionamiento del software, y se reemplazó el mouse convencional de la computadora por un hora. Durante esta hora se utilizó el joystick de la misma forma que se utiliza un mouse de computadora y se realizaron todas las acciones típicas del click derecho e izquierdo. Además, se observó que el tiempo de respuesta del joystick sea similar o comparable con aquel de un mouse inalámbrico convencional.

El éxito de esta prueba está determinado por el correcto funcionamiento del joystick durante una hora continua, pudiendo haber realizado todas las acciones indicadas y teniendo la menor cantidad de fallas en el funcionamiento del joystick. Por ejemplo, “saltos” del cursor en la computadora o registrar clicks incorrectamente.

El código se comportó de forma estable durante la duración de la prueba. La prueba fue exitosa, no se observaron clicks falsos ni saltos del cursor en pantalla.

Capítulo 7

Discusión

7.1. Prototipo funcional ensamblado

Una vez finalizadas e integradas las distintas etapas del diseño e implementación del prototipo, se tuvo un dispositivo funcional capaz de conectarse vía Bluetooth tanto a una computadora como a un dispositivo Android. Se pudo utilizar el prototipo tanto con un teléfono como con una computadora únicamente moviendo la lengua y sin la asistencia de las manos.

7.1.1. Funcionamiento

La primera conexión se debe realizar de forma manual, el prototipo no cuenta con pin ni contraseña para ser emparejado con un dispositivo. Con el prototipo encendido se debe encender el Bluetooth del teléfono móvil o computadora y se lo debe seleccionar de entre la lista de dispositivos disponibles. Una vez realizada la primera conexión, siempre y cuando el prototipo no haya sido conectado a otro dispositivo, no es necesario repetir estos pasos, ya que siempre se intentará conectar al último dispositivo al que estuvo conectado.

Ya conectado el prototipo se puede comenzar a utilizar, se coloca el mordillo en la boca ubicando los dientes dentro de los surcos de la cara frontal con el botón apuntado hacia

la mandíbula superior. Luego basta con desplazar la lengua sobre el sensor para mover el cursor en pantalla y morder levemente para realizar el click. Como ya se mencionó el botón del click abarca todos los usos de tanto el click izquierdo como el derecho de un mouse convencional. Se puede realizar click izquierdo simple y doble, click izquierdo y desplazar el cursor para seleccionar y manteniendo más de 4 segundos presionado el botón se realiza el click derecho.

Teniendo en cuenta que se trata de una de una primera iteración, el dispositivo es cómodo. El mordillo se diseñó a medida según la mandíbula de uno de los integrantes del proyecto y calza cómodamente en la boca del mismo. La lengua no se encuentra totalmente retraída hacia el fondo de la boca con el mordillo colocado y se encuentra relajada cuando no se la está desplazando sobre el sensor.

7.1.2. Dificultades durante su uso

Sensor

El desplazamiento de la lengua sobre el sensor es esperado de un sensor utilizado en móviles *Blackberry*®. En los teléfonos móviles que hace uso de este sensor se requiere desplazar el dedo repetidas veces para recorrer longitudes relativamente largas dentro de la pantalla. El prototipo desarrollado no es ajeno a esta característica lo que termina resultando en fatiga durante usos prolongados. Es una característica propia del sensor, ya que es un sensor relativamente simple, pequeño y de bajos DPIs. Esto se puede solucionar en futuras iteraciones agregando aceleración al movimiento del cursor o aumentando la sensibilidad, que si bien puede no estar permitido por el mouse, se puede escalar el movimiento detectado por un valor fijo o variable. Como último recurso se podría cambiar el sensor por uno más moderno pero este requeriría un rediseño de gran parte de la electrónica y software.

Mordillo

Al tratarse de una primera iteración el encastre del sensor en el mordillo no es perfecto por lo tanto no sobresale de la cara frontal del mordillo, el sensor tiene una cara con una curvatura mínima y se buscó que sobresalga del mordillo. Como esto no sucede se debe presionar levemente la lengua sobre el sensor para que el mismo detecte movimiento, lo que nuevamente provoca fatiga durante periodos largos de uso.

Este diseño obliga al usuario tener la boca abierta 22 mm aproximadamente, esto dificulta el tragado lo que en consecuencia aumenta la cantidad de saliva en la boca. Además, dependiendo de cómo se trague, se puede accionar el botón o mover el cursor en pantalla involuntariamente.

Por último, cabe destacar que el material utilizado en este prototipo no es el apto para un dispositivo de estas características y si bien se podría utilizar la impresión por PolyJet se deben utilizar materiales dentales como aquellos utilizados en ortodoncia [33].

7.2. Mejoras

Si bien se pueden listar un gran cantidad de mejoras, se listaran las mejoras inmediatas que se podrían aplicar en la siguiente iteración, diferenciando por módulo.

7.2.1. Hardware

- **Reducción de tamaño de electrónica:** Utilizando componentes electrónicos tipo SMD o de montaje superficial se puede reducir significativamente el tamaño de las placas y diseñando placas de doble faz, se puede lograr implementar un dispositivo integrado en un solo módulo. En cuanto al Arduino y al módulo HC-05* se los puede soldar directamente sobre las placas. Esto implica agrandar el tamaño del mordillo actual eliminando la caja externa con la electrónica.

- **Cambio de tensión de operación:** Cambio obligatorio que se debe hacer en futuras iteraciones, es posible tener los componentes necesarios para que el dispositivo funcione a 3.3V, esto mejoraría, por ejemplo, el consumo de energía. Para que esto ocurra se debería cambiar el regulador de tensión, el Arduino por un Arduino similar pero que funcione a 3.3V y por último el módulo de recarga
- **Uso de conectores adecuados:** De ser necesario, por ejemplo, para el uso del botón, se pueden agregar conectores del tipo JST para realizar conexiones cableadas. Este tipo de conectores evitan la desconexión de componentes dando mayor robustez a la electrónica
- **Agregar conectores para programar/debuggear:** El diseño actual no cuenta con los pines requeridos para conectar el Arduino a una computadora, agregar estos pines puede facilitar la programación del Arduino y HC-05*

7.2.2. Software

- **Mejora en conectividad Bluetooth:** Conexión automática del dispositivo a la lista de direcciones MAC guardadas en memoria. Actualmente el dispositivo busca la dirección del último dispositivo al que estuvo conectado y no hace uso de la capacidad de guardar direcciones MAC en su memoria
- **Capacidad de configurar el dispositivo a través del click:** Agregar una secuencia capaz de configurar ciertos parámetros del dispositivo como puede ser la sensibilidad del mismo o la aceleración del cursor en pantalla. Se deberían agregar indicadores LED para indicar el estado actual de la configuración del dispositivo
- **Aplicación dedicada:** Desarrollar una aplicación tanto de escritorio como móvil para configurar fácilmente el dispositivo. Hoy en día es común que distintos periféricos,

ya sean cableados o inalámbricos, cuenten con aplicaciones dedicadas que facilitan al usuario la configuración del dispositivo

7.2.3. Gabinete

- **Redimensionar la ranura donde encastra el sensor:** Si bien el sensor cabe en la ranura del gabinete, se pueden redefinir ciertas medidas para que el mismo entre más cómodamente y sobresalga apenas del mordillo. Esto mejoraría la detección del movimiento de la lengua sobre el mismo y como resultado se disminuye la fatiga del usuario
- **Disminuir la longitud de la curvatura de la mandíbula superior:** Si bien el radio de la mandíbula superior es el correcto y la tolerancia para que encastran los dientes es suficiente, en arco es muy largo y choca contra los incisivos laterales. Achicando el arco se puede evitar esta incomodidad sin perder agarre que igualmente estaría dado por la dentadura inferior y los incisivos centrales
- **Explorar estanqueidad del dispositivo:** Actualmente el dispositivo puede tener filtraciones de saliva, asegurar la estanqueidad del mismo es de suma importancia a la hora de realizar un dispositivo de estas características
- **Tener en cuenta la orientación de la impresión 3D:** La cara frontal del mordillo por como fue impresa resulta áspera al tacto de la lengua, se debe cambiar la orientación en la que se imprime para obtener un acabado listo y facilitar el uso del periférico
- **Disminuir la apertura de la boca:** Esto mejoraría la ergonomía del dispositivo, actualmente la apertura es de 22 mm, reducir esto en algunos milímetros mejoraría el uso del dispositivo durante periodos de uso prolongados. Una posible solución de conseguir esto es alargar el mordillo hacia afuera de la boca y disminuir su altura para que quepa la electrónica.

7.3. Hitos del proyecto

A lo largo de la realización del proyecto final hubo ciertos avances, resultados y decisiones que fueron clave al desarrollo del mismo. Estos hitos marcaron un antes y un después en el desarrollo ya que cada uno de ellos permitió el avance del diseño del prototipo, la ausencia o fracaso de estos hitos hubiese resultado en un regresión del mismo.

7.3.1. Comunicación Bluetooth en perfil HID

Desde la concepción se propuso realizar un dispositivo inalámbrico. Durante la etapa de investigación y con el objetivo de contar con un abanico más amplio de soluciones se tuvieron en cuenta desde distintas conexiones inalámbricas hasta cableadas. El hecho de encontrar una solución apropiada y reemplazar el firmware de un módulo existente por otro y que pueda ser reconocido por una computadora como un teclado, cursor o joystick permitió la definición completa del tipo de comunicación cumpliendo una de las características técnicas propuestas. Si esto no sucedía de esta forma probablemente la solución hubiese requerido del desarrollo de controladores y programación de bajo nivel para lograr comunicar el prototipo con la computadora.

7.3.2. Definición y prueba del sensor

La definición tanto de la tecnología como del sensor mismo fue trabajoso y requirió de investigación extensiva. La definición del sensor a su vez determinaría el diseño final del prototipo, cada cambio del sensor en la fase de diseño implicaba un cambio, no solamente en la forma final del dispositivo, sino en la ubicación y forma de uso del mismo. Por ejemplo, de haber seleccionado un joystick como método de input el dispositivo se hubiera ubicado por fuera de la boca. También se analizó la posibilidad de desarrollar un pad resistivo para ubicar por debajo del paladar, si bien fue una de las primeras soluciones

estudiadas se comenzó a diseñar una prueba de concepto de la misma. Hasta que se dio con el sensor utilizado.

7.3.3. Definición y finalización del mordillo

Una vez definido el sensor restaba definir la forma del prototipo final, nuevamente y dado el tamaño del sensor elegido se estudiaron distintas formas que podía tomar el prototipo. Finalmente y respetando lo mejor posible las características técnicas propuestas, se definió el diseño de mordillo, en un principio integrado en un solo módulo y luego en dos separados. El diseño del mismo comenzó en lápiz y papel, pasando por maquetas seguidas de SolidWorks™ y Siemens NX™ y finalmente impreso en PolyJet. El ensamblado y correcto funcionamiento del prototipo comenzó con el cierre y finalización del proyecto final.

Capítulo 8

Conclusión

Dada la funcionalidad del prototipo desarrollado se puede concluir que un dispositivo de estas características es realizable, la tecnología es aplicable, funcional, conocida y accesible. A lo largo del desarrollo se pudieron cumplir tanto los objetivos de mínima, incluyendo el cumplimiento de las características distintivas propuestas, y los objetivos de máxima. El diseño propuesto funciona, en parte, dentro de la boca y dada la funcionalidad obtenida se puede continuar con el desarrollo del mismo.

A pesar del desarrollo planteado y de los resultados obtenidos, se debe mencionar, que este prototipo no se encuentra en condiciones de ser utilizado por usuarios. Se trata de una prueba de concepto que explora la aplicación de distintas tecnologías en un dispositivo de estas características. Hay aspectos funcionales y, más importante, de seguridad que deben ser considerados, investigados y aplicados para que un dispositivo así pueda ser utilizado por distintos usuarios. Dicho esto, el dispositivo es ampliamente mejorable y se podrían continuar con futuros desarrollos, hasta ser posible realizar pruebas con pacientes.

Dentro del diseño del mordillo, con el desarrollo de electrónica dedicada típica de mouses comerciales y un sensor más complejo y avanzado, el mismo puede o no ser dedicado, y un encapsulado propio a medida de cada usuario, se puede obtener un dispositivo apto y funcional. Adicionalmente, implementando el diseño para ser utilizado íntegramente

dentro de la boca debajo del paladar, se podría agregar un teclado similar al de los teléfonos móviles de principios de la década pasada [30]. Agregando funcionalidad se podría analizar el uso de un periférico de estas características en otros dispositivos y facilitar de alguna forma a las personas que lo necesiten.

Finalmente, a pesar de las frustraciones durante el proyecto, propias de un proyecto de estas características, el resultado obtenido fue satisfactorio. Si bien se aplicaron conocimientos aprendidos en carrera y se aprendieron nuevos, la naturaleza de un proyecto como éste, obligó al grupo de trabajo a hacer uso de la creatividad y pensamiento lateral a la hora superar distintos desafíos.

Referencias

- [1] I. N. del Cancer. «Anatomía y funciones de los tumores de cerebro y médula espinal.» (2020), dirección: [https://www.cancer.gov/rare-brain-spine-tumor/espanol/tumores/anatomia#:~:text=La%5C%20m%5C%C3%5C%A9dula%5C%20espinal%5C%20conecta%5C%20el,sistema%5C%20nervioso%5C%20central%5C%20\(SNC\).](https://www.cancer.gov/rare-brain-spine-tumor/espanol/tumores/anatomia#:~:text=La%5C%20m%5C%C3%5C%A9dula%5C%20espinal%5C%20conecta%5C%20el,sistema%5C%20nervioso%5C%20central%5C%20(SNC).)
- [2] G. Mao. «Lesiones de la médula espinal y de las vértebras.» (2021), dirección: <https://www.msmanuals.com/es-ar/hogar/traumatismos-y-envenenamientos/lesiones-medulares/lesiones-de-la-m%C3%A9dula-espinal-y-de-las-v%C3%A9rtebras.>
- [3] B. S. A. E. A. C.-L. M. H. F. H. Fredine y A. Cahill, «Prevalence and Causes of Paralysis—United States, 2013,» Tesis doct., <https://ajph.aphapublications.org/doi/full/10.2105/AJPH.2013.>
- [4] Volodymyr. «SPINAL CORD INJURY STATISTICS.» (2019), dirección: [https://vlukash.com/2019/01/15/trackpad-in-keycap-corne-crkbd-keyboard/.](https://vlukash.com/2019/01/15/trackpad-in-keycap-corne-crkbd-keyboard/)
- [5] A. Agarwal y J. B. Lang, *Foundations of Analog and Digital Electronic Circuits*, 1.^a ed. ELSEVIER, 2005.
- [6] K. Kalantar-zadeh, *Sensors, An Introductory Course*, 1.^a ed. Springer, 2013.
- [7] «Sistemas táctiles,» EOS Iberica. (2022), dirección: [https://eosiberica.es/blog-sistema-tactiles/.](https://eosiberica.es/blog-sistema-tactiles/)
- [8] «Ventajas Y Desventajas Del Panel Táctil Capacitivo,» El Sexto Elemento materiales Technology Co. (2022), dirección: [http://es.c6th.com/news/capacitive-touch-panel-advantages-and-disadvan-7622676.html.](http://es.c6th.com/news/capacitive-touch-panel-advantages-and-disadvan-7622676.html)
- [9] «Módulo trackball,» TecBolivia. (2022), dirección: [http://tecbolivia.com/index.php/venta-de-componentes-electronicos-11/sensores/modulo-trackball-detail.](http://tecbolivia.com/index.php/venta-de-componentes-electronicos-11/sensores/modulo-trackball-detail)
- [10] G. B. G. D. L. K. R. B. T. Hartlove, «Seeing eye mouse for a computer system,» Tesis doct., Patent Number: 6281882, 1998.
- [11] X. Lee. «What is Mouse DPI? Does DPI Matter?» (2021), dirección: [http://xahlee.info/kbd/mouse_dpi.html.](http://xahlee.info/kbd/mouse_dpi.html)

-
- [12] *Agilent ADNS-3060 High-performance Optical Mouse Sensor*, Agilent Technologies, Inc., 2013.
- [13] *ADNS-2610 Optical Mouse Sensor*, Agilent Technologies, Inc., 2008.
- [14] *PAW3204 LOW COST WIRELESS MOUSE SENSOR*, PixArt Imaging, Inc., 2009.
- [15] *PAW-350: Optical Finger Navigation (OFN) Module*, PixArt Imaging, Inc., 2018.
- [16] A. Nusca. «What's behind new BlackBerry optical trackpad.» (2009), dirección: <https://www.cnet.com/tech/mobile/iphone-users-remember-to-clear-your-iphone-cache-regularly/>.
- [17] M. N. «thumbMouse.» (2020), dirección: <https://hackaday.io/project/167075-thumbmouse>.
- [18] T. M. Project. «Trackpad in a Keycap for Corne/CrKbd Keyboard.» (2020), dirección: <https://www.themiamiproject.org/resources/statistics/>.
- [19] *SPI Block Guide V03.06*, Motorola, Inc., 2000.
- [20] «SPI Explained, An Introduction To The Standard Serial Peripherals Bus,» electricImpDevCenter. (2022), dirección: <https://developer.electricimp.com/resources/spi>.
- [21] «Introduction to I2C and SPI protocols,» Byte Paradigm. (2022), dirección: <https://www.byteparadigm.com/applications/introduction-to-i2c-and-spi-protocols/?/>.
- [22] *Bluetooth Data Module Command Reference and Advanced Information User's Guide*, Roving Networks, Inc., 2013.
- [23] «Arduino Nano,» Arduino Inc. (2022), dirección: <https://store-usa.arduino.cc/products/arduino-nano?selectedStore=us>.
- [24] «Arduino Pro Mini,» Arduino Inc. (2022), dirección: <https://docs.arduino.cc/retired/boards/arduino-pro-mini>.
- [25] «Arduino Pico,» Mell Bell Electronics. (2022), dirección: <https://mellbell.cc/products/pico>.
- [26] *LT1117/LT1117-2.85 LT1117-3.3/LT1117-5. 800mA Low Dropout Positive Regulators Adjustable and Fixed 2.85V, 3.3V, 5V*, Linear Technology Corporation, 1993.
- [27] *TXB0108 8-Bit Bidirectional Voltage-Level Translator with Auto-Direction Sensing and ±15-kV ESD Protection*, Texas Instruments, 2022.
- [28] «¿Qué es un Condensador de Desacoplo?» Learning about Electronics. (2017), dirección: <http://www.learningaboutelectronics.com/Articulos/Condensador-de-desacoplo.php>.

-
- [29] «Understanding 2 Layer PCB Ground Planes,» Altium Limited. (2018), dirección: <https://resources.altium.com/p/understanding-ground-planes-your-two-layer-pcb>.
- [30] F. et al., «Tongue Activated Communications Controller,» Tesis doct., Patent Number: 5523754, 1996.
- [31] H. Park y M. Ghovanloo, «An Arch-Shaped Intraoral Tongue Drive System with Built-in Tongue-Computer Interfacing SoC,» *MDPI Sensors*, 2014.
- [32] L. Wright. «Tongue-controlled computer mouse earns Ontario teen a science prize,» CBC News. (2016), dirección: <https://www.cbc.ca/news/science/tongue-controlled-computer-mouse-1.3677813>.
- [33] *Advanced materials for superior digital dentistry and orthodontics*. Strasys, 2014.

APÉNDICES

Apéndice A

Diseño de placas en Altium

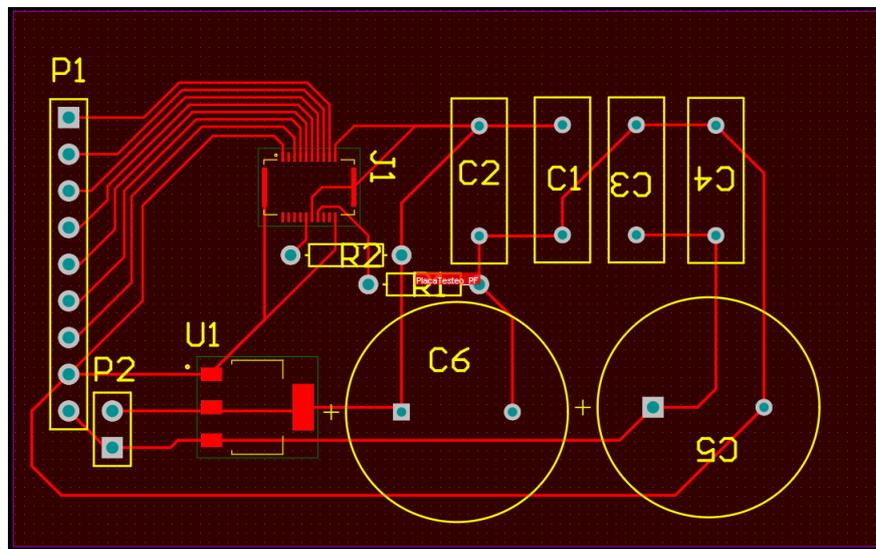


Figura A.1 Placa de prueba diseñada en Altium Designer ®.

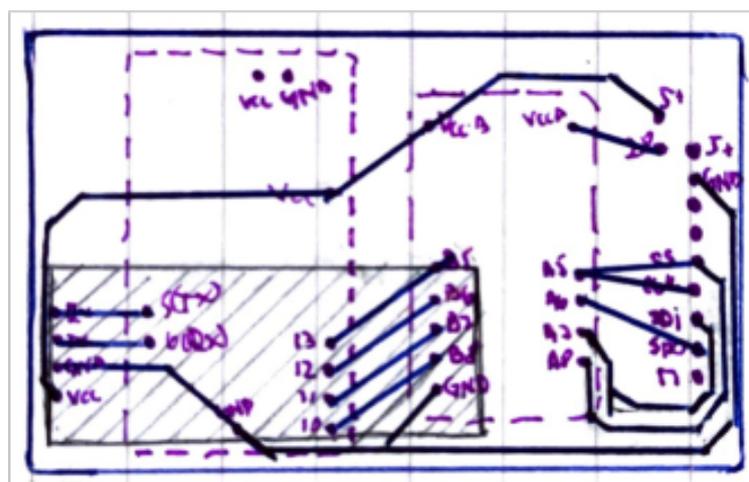


Figura A.2 Dibujo del circuito complementario antes de ser armado

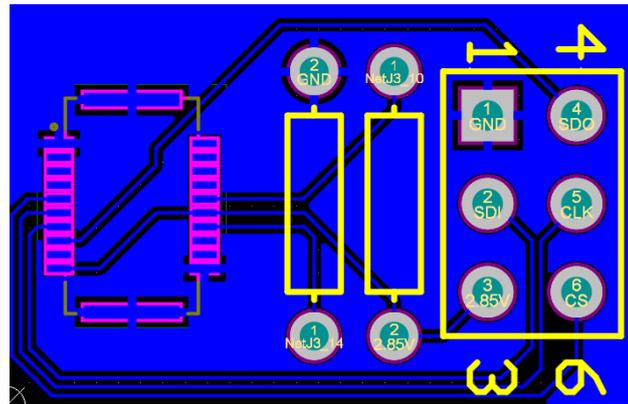


Figura A.3 Diseño de placa correspondiente al PCB del sensor

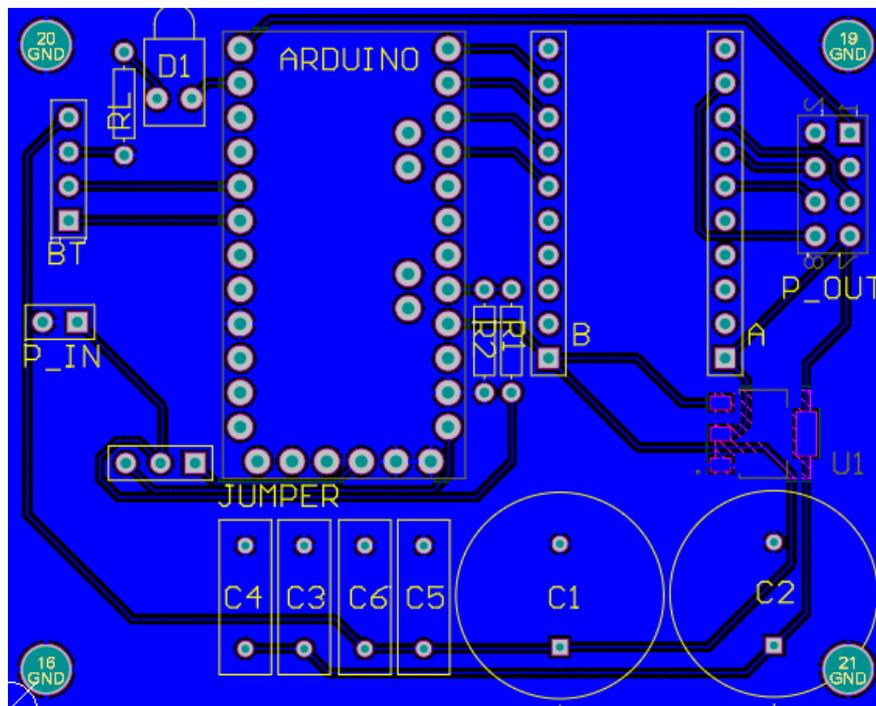


Figura A.4 Diseño de placa correspondiente al PCB de procesamiento