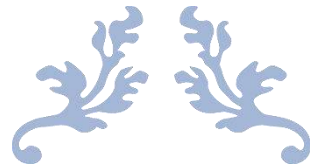


Universidad Nacional de Mar del Plata
Facultad de Ingeniería
Departamento de Ingeniería Electrónica



DISPOSITIVO TELEMÉTRICO DE SEÑALES DE EEG



Agustin Ferreira y Facundo Roldán

Director: Gonzalo Fontanella

Co – Director: Gustavo Meschino

TRABAJO FINAL DE GRADUACIÓN PARA ACCEDER AL TÍTULO DE
INGENIERO ELECTRÓNICO

MAR DEL PLATA



RINFI es desarrollado por la Biblioteca de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata.

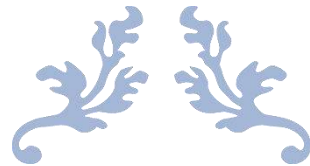
Tiene como objetivo recopilar, organizar, gestionar, difundir y preservar documentos digitales en Ingeniería, Ciencia y Tecnología de Materiales y Ciencias Afines.

A través del Acceso Abierto, se pretende aumentar la visibilidad y el impacto de los resultados de la investigación, asumiendo las políticas y cumpliendo con los protocolos y estándares internacionales para la interoperabilidad entre repositorios



Esta obra está bajo una [Licencia Creative Commons Atribución- NoComercial-CompartirIgual 4.0 Internacional](https://creativecommons.org/licenses/by-nc-sa/4.0/).

Universidad Nacional de Mar del Plata
Facultad de Ingeniería
Departamento de Ingeniería Electrónica



DISPOSITIVO TELEMÉTRICO DE SEÑALES DE EEG



Agustin Ferreira y Facundo Roldán

Director: Gonzalo Fontanella

Co – Director: Gustavo Meschino

TRABAJO FINAL DE GRADUACIÓN PARA ACCEDER AL TÍTULO DE
INGENIERO ELECTRÓNICO

MAR DEL PLATA

Resumen

El presente trabajo tiene como objetivo implementar un prototipo funcional para una solución portátil, telemétrica y de bajo costo de diseño que contribuya con la adquisición de señales fisiológicas. Con dicha finalidad, desarrollamos un sistema de adquisición, transmisión, procesamiento y visualización remota de las señales biomédicas. Las señales a analizar poseen una baja relación señal/ruido; es por ello la importancia de la selección de un sistema de acondicionamiento adecuado. En consecuencia, utilizamos un amplificador de instrumentación y conversor analógico/digital para amplificar y muestrear la señal. Para configurar el dispositivo y gestionar la comunicación remota a través de la red utilizamos un microcontrolador. La comunicación entre el amplificador y el microcontrolador se realizó a través de una Interfaz periférica de puerto serie. El software para gestionar las comunicaciones se desarrolló en una implementación reducida de Python con el propósito de facilitar su programación. La interfaz gráfica en la computadora se desarrolló con el mismo lenguaje.

El beneficiario de este desarrollo es el Laboratorio de Bioingeniería de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata. Si bien el Laboratorio cuenta con un dispositivo para realizar mediciones de señales biomédicas de manera inalámbrica, dicho adquirente es importado, utiliza un protocolo de comunicación propietario, no cuenta con la posibilidad de adquirir más de una señal electroencefalográfica o electrocardiográfica (derivación) en simultáneo, es de código cerrado y no cuenta con la posibilidad de marcar eventos para estudios de reacción a estímulos externos. Con la incorporación de este dispositivo se pretende proveer al Laboratorio una plataforma de desarrollo libre, accesible y fácil de usar que gestiona un sistema de adquisición de señales de varios canales, de bajo costo y de código abierto con una Interfaz de configuración y adquisición amigable para el usuario.

La plataforma posibilitará llevar a cabo futuros proyectos de investigación que se aboquen al campo de la electroencefalografía o estudios de neurociencias. Al ser práctica, liviana e inalámbrica podrá ser utilizada con fines didácticos en la asignatura Introducción a la Bioingeniería. Presenta una alternativa atractiva y económica a los sistemas propietarios convencionales usualmente utilizados en educación.

Los resultados de este trabajo indican que la combinación del hardware y la programación utilizados ofrece una alta precisión en la medición de señales biomédicas y una comunicación inalámbrica confiable para la transferencia de datos. Se espera continuar investigando en esta

área para mejorar la precisión y la eficiencia de los dispositivos telemétricos de señales biomédicas.

Índice

Resumen.....	1
Definiciones, acrónimos y abreviaturas	7
Capítulo 1	8
Introducción	8
1.1 Motivación y clientes	8
1.2 Estado del Arte	9
1.3 Características de las señales de EEG.....	9
Capítulo 2	10
Objetivos y requerimientos funcionales	10
2.1 Objetivo.....	10
2.2 Perspectiva	11
2.3 Funciones del producto.....	11
2.4 Características de los usuarios	12
2.5 Restricciones	12
2.6 Suposiciones y dependencias.....	14
2.7 Requisitos futuros	14
2.8 Requisitos específicos	14
2.8.1 Requerimientos funcionales de hardware	14
2.8.2 Requerimientos funcionales de software	16
Capítulo 3	17
Diseño del hardware	17
3.1 Introducción	17
3.2 Etapa de Adquisición.....	18
3.2.1 Selección del adquisidor.....	18
3.2.2 ADS 1299	20
3.2.3 Red de multiplexado	21

3.2.4 Entradas analógicas.....	22
3.2.5 Señal de polarización.....	23
3.2.6 Detección de conexión de electrodos.....	25
3.2.7 Detección de conexión del electrodo de polarización.....	26
3.2.8 ADC Sigma Delta.....	26
3.2.9 Filtro digital Sinc.....	27
3.2.10 Requerimientos de filtrado.....	27
3.3 Microcontrolador.....	28
3.3.1 Introducción.....	28
3.3.2 Elección del Microcontrolador.....	28
3.3.3 ESP32.....	29
3.3.4 Comunicación entre AFE y microcontrolador.....	30
3.3.5 Comunicación entre microcontrolador y PC.....	32
3.3.6 Circuito cargador de arranque.....	33
3.4 Alimentación eléctrica.....	34
3.4.1 Requerimientos.....	34
3.4.2 Evaluación de alternativas.....	35
3.4.3 Elección de batería.....	37
3.4.4 Diseño de fuentes.....	38
3.4.5 Gestión de carga de la batería.....	40
3.5 Diseño de la PCB.....	42
Capítulo 4.....	43
Diseño del software del dispositivo.....	43
4.1 Introducción.....	43
4.2 Requerimientos.....	44
4.2.1 Requerimiento de tiempo del puerto SPI:.....	46

4.2.2 Requerimiento de tiempo de envío de las señales y tamaño del mensaje:	46
4.3 Funcionamiento y formato de los datos del ADS1299.....	47
4.3.1 Interfaz SPI:	47
4.3.2 Pines y comandos:.....	48
4.4 Arquitectura	51
Capítulo 5	56
Diseño de software de la computadora.....	56
5.1 Introducción	56
5.2 Desarrollo	57
Capítulo 6	64
Pruebas y análisis de resultados	64
6.1 Verificación general.....	64
6.1.1 Medición de continuidad y ausencia de corto circuitos en la placa.	64
6.1.2 Medición de las tensiones de fuentes:.....	65
6.1.3 Verificación del sistema de carga de la batería:.....	65
6.1.4 Depuración del ADS1299:.....	66
6.1.5 Comprobación de la conexión USB y carga del software embebido en el esp32:.....	67
6.1.6 Prueba de funcionamiento de llave de encendido y sistema de puentes	68
6.1.7 Autonomía.....	69
6.2 Verificación del adquisidor.....	70
6.2.1 Medición de señal de prueba interna y verificación de los 4 canales.....	70
6.2.2 Rechazo al modo común CMRR	71
6.2.3 Resolución	72
6.2.4 Ruido equivalente a la entrada	72
6.3 Pruebas de comunicación	73
6.3.1 Distancia entre dispositivos	73

Capítulo 7	75
Conclusiones	75
Bibliografía	77
Apéndice.....	81
A.1 Red de multiplexado	81
A.2 Filtros de entrada.....	82
A.3 Esquemático.....	84

Definiciones, acrónimos y abreviaturas

Nombre	Descripción
RF	Requerimiento Funcional
RNF	Requerimiento No Funcional
EF	Especificación Funcional
ENF	Especificación No funcional
AD	Analógico digital
EEG	Señal de electroencefalografía
PC	Computadora personal
PCB	Circuito impreso
<u>Binding</u>	Es una adaptación de una biblioteca para ser usada en un lenguaje de programación distinto de aquel en el que ha sido escrita.
Widgets	Es una pequeña aplicación o programa, usualmente presentado en archivos o ficheros pequeños que son ejecutados por un motor de <i>widgets</i> o <i>Widget Engine</i> . Entre sus objetivos están dar fácil acceso a funciones frecuentemente usadas y proveer de información visual.
EEG	Señal de electroencefalografía
ESP32	Microcontrolador de Espressif utilizado en el proyecto
ADS1299	Frontend de Texas instruments utilizado en el proyecto

Capítulo 1

Introducción

1.1 Motivación y clientes

El avance de la neurociencia y la electrónica en los últimos años ha hecho posible el desarrollo de dispositivos inalámbricos que adquieran señales de electroencefalografía (*EEG*) para un diverso rango de aplicaciones. Por ejemplo, existen dispositivos para detectar si una persona se está durmiendo [1]; esto es particularmente importante en conductores que recorren largos trayectos al frente de vehículos. Por otro lado, existen videojuegos que funcionan con "control mental" usando dispositivos que procesan señales de *EEG*. De esta manera, es factible pensar que en los próximos años se desarrollen dispositivos *EEG* inalámbricos para controlar prótesis o para detectar con antelación si un paciente está por sufrir una crisis epiléptica. Esto último permitirá que una persona con esta enfermedad pueda realizar actividades que actualmente no puede realizar con seguridad, como por ejemplo conducir un automóvil o nadar. El impacto que genera el avance de la neurociencia y todo el campo en el cual se puede aplicar constituyen importantes motivaciones para este trabajo.

El beneficiario de este desarrollo es el Laboratorio de Bioingeniería de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata.

Si bien el Laboratorio cuenta con un dispositivo para realizar mediciones de señales eléctricas del cerebro, dicho adquisidor es importado, no cuenta con la posibilidad de adquirir más de una derivación en simultáneo, es de código propietario no permitiendo su manipulación.

Con la incorporación de este dispositivo se pretende proveer al Laboratorio de un sistema de adquisición de señales *EEG* que disponga de varios canales, que sea práctico, liviano, inalámbrico, de bajo costo de diseño y de código abierto, para poder de esta forma llevar a cabo proyectos de investigación. Además, el mismo tendrá fines didácticos en la asignatura Introducción a la Bioingeniería, como así también podrá ser utilizado en futuros proyectos que se aboquen al campo de la electroencefalografía o estudios de neurociencias.

1.2 Estado del Arte

La electroencefalografía es la técnica encargada del registro e interpretación del electroencefalograma (*EEG*), que consiste en el registro de las señales eléctricas generadas por la acción cooperativa de las células cerebrales (neuronas). La actividad eléctrica neuronal puede ser registrada por medio de electrodos situados en el cuero cabelludo del paciente.

Actualmente, la electroencefalografía se utiliza como herramienta en la mayoría de los estudios neurológicos.^[2] Los equipos de grado hospitalario, para realizar los estudios, disponen de múltiples canales de adquisición (*desde 32 a 256 canales*), interfaces gráficas y otras prestaciones que suelen ser plataformas cerradas y costosas.

En otra categoría se encuentran los dispositivos de menor cantidad de canales tales como OpenBCI ^[3] y el desarrollo comercial Biopac ^[4].

OpenBCI es una empresa que produce dispositivos telemétricos de EEG de código abierto. Su sistema consta de un hardware personalizado, que incluye un amplificador de instrumentación ADS1299 de 8 canales y una placa que utiliza el microcontrolador ATmega328P de Atmel, que es un microcontrolador AVR de 8 bits con un procesador RISC de baja potencia y alta velocidad de reloj, y un software personalizado que permite la adquisición y el análisis de datos de EEG. El sistema de OpenBCI ofrece una solución de bajo costo para la adquisición de señales de EEG y es muy popular en la comunidad de la neurociencia.

Biopac es otra empresa que produce dispositivos telemétricos de señales de EEG, entre otros sistemas de adquisición de datos biomédicos. Su sistema EEG consiste en un amplificador de instrumentación, un software de adquisición de datos y un software de análisis de las señales adquiridas. El sistema de Biopac es de código propietario, no permitiendo su manipulación.

Ambos sistemas, OpenBCI y Biopac, son soluciones comerciales de adquisición de señales de EEG y ofrecen una alternativa a los sistemas de EEG convencionales. Estos sistemas permiten la monitorización inalámbrica de señales de EEG de calidad en entornos móviles y ambulatorios.

1.3 Características de las señales de EEG

La electroencefalografía es el estudio de la actividad eléctrica cerebral. Mediante ella se procura investigar el comportamiento del cerebro en distintos estados de estimulación y de reposo. Estos registros se realizan midiendo las variaciones de potencial causadas por la actividad neuronal. Actualmente se utiliza para el análisis clínico de cualquier fenómeno que se sospeche de origen cerebral.

La amplitud típica de las señales de *EEG* varía entre $10\mu V$ y $200\mu v$ y presentan un rango de frecuencia entre $0.1Hz$ y $100Hz$. Se clasifican en cinco tipos según su frecuencia [5] [6] [7]:

- Delta: $0.1Hz$ a $3.5Hz$. Asociadas a la etapa de sueño profundo.
- Theta: $4Hz$ a $7.5Hz$. Presentes en las primeras etapas de sueño, fases 1 y 2.
- Alfa: $8Hz$ a $13Hz$. Se originan durante periodos de relajación, con los ojos cerrados, pero todavía despierto. Desaparecen al abrir los ojos.
- Beta: $1Hz$ a $30Hz$. Están asociadas con etapas de vigilia, donde se está despierto y consciente.
- Gamma: Mayor a $30Hz$. Están asociadas a estados de conciencia, aunque no hay un acuerdo unánime respecto a esto.

Capítulo 2

Objetivos y requerimientos funcionales

2.1 Objetivo

El objetivo del proyecto es diseñar y construir un prototipo de dispositivo telemétrico de señales biomédicas y específicamente de electroencefalografía. El proyecto comprende desde el diseño y construcción del hardware y el software para la adquisición de las señales bioeléctricas, hasta su transmisión inalámbrica, procesamiento y visualización en una PC.

Este dispositivo se integra conjuntamente a un casco de malla que dispone de electrodos comerciales aptos para la práctica de EEG. Dicho casco se encuentra disponible en el Laboratorio de Bioingeniería

2.2 Perspectiva

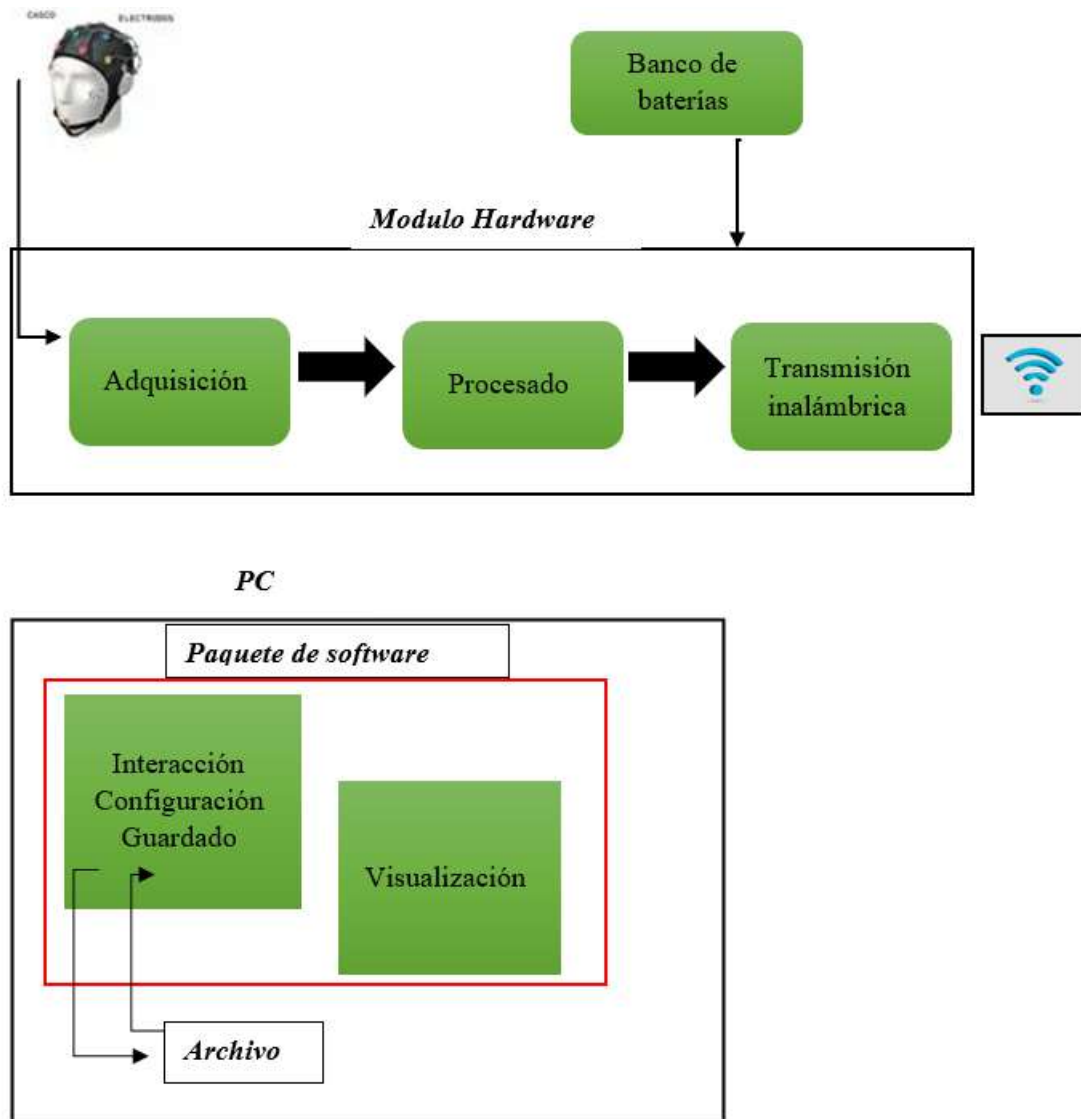
Este producto está enmarcado en el ámbito de investigación y enseñanza de la Universidad Pública. En dicho ámbito, la disponibilidad de recursos para la compra de equipamiento o instrumental para realizar determinadas prácticas es limitada. Además, los requerimientos del investigador y los avances en las técnicas de investigación hacen que los dispositivos adquiridos de códigos y hardware cerrados queden rápidamente obsoletos. En este contexto y considerando una de las necesidades del laboratorio de Bioingeniería de la Facultad de Ingeniería de la Universidad Nacional de Mar del Plata se desarrolla este proyecto.

2.3 Funciones del producto

Este sistema cuenta con un módulo de hardware y un paquete de software.

El módulo de hardware es capaz de adquirir señales de EEG provenientes de electrodos húmedos de montaje superficial montados en el cuero cabelludo del paciente. Cumple con las funciones de aislación eléctrica, filtrado de las señales útiles, muestreo y adquisición. Luego procesa y encapsula la información para enviarla inalámbricamente a una PC.

El paquete de software funciona en una PC, presenta una interfaz gráfica que le permite al cliente interactuar con el módulo de hardware para configurar el estudio. También se le brinda al cliente la posibilidad de guardar el estudio como así también visualizarlo en tiempo real.



2.4 Características de los usuarios

El producto desarrollado está dirigido a destinatarios con conocimientos de EEG, electrónica y programación. Este proyecto se pensó con fines educativos, como complemento a las prácticas de EEG realizadas en asignaturas con contenidos de bioingeniería de grado y posgrado, o bien en el desarrollo de trabajos finales de grado y proyectos de investigación.

2.5 Restricciones

Restricción	Aplica	Explicación
Regulación	Si	<p>La norma IRAM 4020-1 (IEC 60601-1) describe los requisitos generales que deben cumplir los sistemas electro médicos para efectos de seguridad en el paciente en el cual va a ser usado el equipo.</p> <p>Se deberá cumplir con el estándar IEEE 802.11 para comunicaciones inalámbricas en la banda de 2.4 GHz.</p>
Limitaciones de hardware	Si	<p>El proyecto depende de componentes de origen importados cuya disponibilidad puede verse afectada por medidas regulatorias.</p> <p>La computadora sobre la que se ejecute el paquete de software deberá contar con una placa de conexión inalámbrica compatible con el estándar IEE 802.11 y deberá contar con los permisos adecuados para conectarse al módulo de hardware.</p>
Interfaces con otras aplicaciones	Si	Se requiere que los drivers de control de la interfaz inalámbrica sean compatibles con el modelo de protocolos TCP/IP.
Operaciones en paralelo	No	No requiere operaciones en paralelo.
Protocolos de comunicación	Si	Requiere cumplir con el estándar IEEE 802.11 para la comunicación entre el módulo de hardware y la computadora.

Criticidad de la aplicación	Si	Si bien no se ha cuantificado, la aplicación deberá funcionar de forma acorde sin errores previstos y por un lapso continuo de al menos 30 minutos.
Consideraciones de seguridad	Si	El producto deberá cumplir con estándares de seguridad eléctrica ya que será conectado a una persona.

2.6 Suposiciones y dependencias

Se asume que el operador del dispositivo cuenta con una PC con capacidad de conexión inalámbrica wifi y que en dicha computadora corre un sistema operativo Windows 7 o superior compatible con el software del dispositivo.

Se asume que el operador del dispositivo cuenta para la realización del estudio de EEG con los electrodos necesarios y aptos para conectar al dispositivo.

2.7 Requisitos futuros

Este dispositivo es escalable para adquirir 16 canales de señales EEG. Su software permite su adecuación para fines específicos, permitiendo utilizar otras librerías dentro del paquete de software Python que se ajusten a las necesidades del investigador o desarrollador.

2.8 Requisitos específicos

2.8.1 Requerimientos funcionales de hardware

- **Número de canales a adquirir:** el dispositivo deberá contar con cuatro derivaciones que permitan la adquisición simultánea de señales de EEG.

- El usuario podrá conectar electrodos de EEG en todas o en algunas de las cuatro entradas provistas, y seleccionar mediante software la activación y configuración de estas.
- **Tipo de señales a adquirir:** se deberá poder adquirir señales de amplitud entre $10\mu V$ y $200\mu V$, y con una frecuencia de entre 0.1 Hz y 35 Hz .
 - El usuario podrá adquirir las señales provenientes de electrodos de EEG de montaje superficial en un espectro de frecuencias que incluye las principales características de estas.
- **Detección del estado de los electrodos:** el dispositivo deberá contar con un circuito que detecte la desconexión de los electrodos.
 - El usuario dispondrá en la interfaz de pantalla de información que le indique el estado de conexión de los electrodos.
- **Relación de rechazo al modo común:** el dispositivo deberá brindar un CMRR de al menos 80db .
 - El dispositivo deberá tener un CMRR, en la banda de frecuencias de 0.1 Hz a 35 Hz , de por lo menos 80db .
- **Frecuencia de muestreo:** el dispositivo deberá muestrear las señales de EEG con una velocidad de por lo menos 250 muestras por segundo.
 - El usuario dispondrá en la interfaz de pantalla la posibilidad de seleccionar entre diferentes velocidades de transferencia. Estas velocidades de transferencia a su vez determinaran el ancho de banda de las señales a adquirir según la siguiente tabla:

<i>SPS</i>	<i>-3db AB Hz</i>
1000	262
500	131
250	65

- **Resolución:** las señales adquiridas deberán tener una resolución de al menos 12 bits.
 - Los datos adquiridos deberán tener una resolución de al menos 12 bits libres de ruido.

- **conexión con la PC:** el dispositivo deberá generar una red inalámbrica en modo AP con protocolo IEEE 802.11 accesible desde una PC.
 - El usuario desde la PC deberá conectarse a la red generada y luego abrir el software de gestión para conectar con el dispositivo.
- **Velocidad de transferencia:** la transmisión inalámbrica deberá tener un throughput útil de al menos 15 *Kbps* para poder transferir los datos de cuatro derivaciones sin retardos en la señal.

2.8.2 Requerimientos funcionales de software

El software deberá contar con una interfaz gráfica de usuario (GUI, Graphical User Interface), que permita cumplir con los siguientes requerimientos.

- **Conectar con el dispositivo:** deberá disponer de un método que permita conectar y desconectar el dispositivo.
 - El usuario dispondrá de un botón que permita, haciendo clic con el mouse, conectar y desconectar el software con el dispositivo y obtener una verificación visual de que el comando fue efectuado.
- **Indicación de nivel de batería:** deberá contar con alguna indicación del nivel de batería.
 - El usuario tendrá en pantalla una indicación que le permita conocer el estado de carga de la batería.
- **Fecha y hora actuales:** se deberá poder visualizar en pantalla la fecha y la hora actuales.
- **Datos del paciente y observaciones del estudio:** deberá permitir ingresar mediante teclado los datos de nombres y apellidos del paciente, así como observaciones pertinentes del estudio. Deberá indicarse en forma clara un campo donde ingresar los datos.
- **Guardar datos:** deberá permitir guardar en un archivo de texto los datos de fecha, hora, datos del paciente y observaciones.
- **Cargar datos:** deberá permitir cargar datos de un registro previo.

- El usuario, mediante un menú, podrá cargar los datos de un registro realizado
- **Seleccionar la cantidad de canales:** deberá permitir seleccionar la cantidad de canales a registrar.
 - El usuario podrá determinar la cantidad de canales que desea adquirir con un máximo de cuatro.
- **Asociar electrodos:** deberá permitir asociar cada canal seleccionado a un electrodo específico del sistema 10 – 20.
 - El usuario mediante la interfaz podrá asociar cada canal individual con un electrodo posicionado de acuerdo con el sistema 10-20 en el casco.
- **Visualizar el estado de conexión de los electrodos:** Permitirá visualizar el estado de conexión de los electrodos.
 - El usuario podrá verificar mediante un indicador en pantalla, que los electrodos están correctamente colocados, para obtener una medición real de las señales de electroencefalografía.
- **Velocidad de salida de los datos:** Se podrá seleccionar una velocidad de salida de los datos de hasta 1000 muestras por segundo.
 - El usuario dispondrá de la posibilidad de seleccionar la velocidad de salida de los datos del conversor analógico digital entre 250, 500 o 1000 *SPS*.
- **Aplicar filtros digitales:** Permitirá aplicar filtros digitales y seleccionar sus frecuencias de corte inferior y superior.
 - El usuario tendrá la posibilidad de aplicar filtros digitales a las señales adquiridas y modificar sus frecuencias de corte superior e inferior.
- **Iniciar y pausar la visualización:** Deberá permitir iniciar y pausar la visualización.

Capítulo 3

Diseño del hardware

3.1 Introducción

En este capítulo se describe la electrónica diseñada para dar solución a los requerimientos establecidos. Se realiza una breve descripción del funcionamiento de las partes que conforman el dispositivo, la justificación de su diseño y se brinda información de las conexiones entre componentes. En la *figura 3.1.a* se presenta un diagrama con los bloques necesarios.

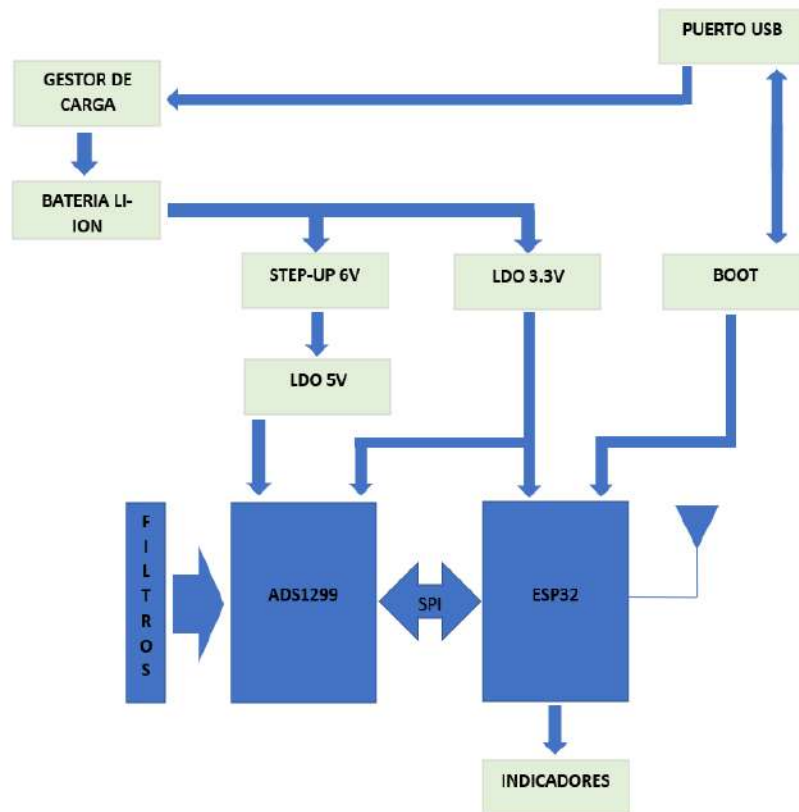


figura 3.1.a

3.2 Etapa de Adquisición

3.2.1 Selección del adquisidor

En función de los requerimientos de hardware y las restricciones descritos en el *capítulo 2* se evaluaron las siguientes alternativas considerando sus ventajas y desventajas.

Diseñar las etapas de amplificación y digitalización con componentes discretos o utilizar un circuito integrado dedicado a la adquisición de señales biomédicas.

En relación con la eficiencia en el diseño y la reducción del tamaño se vio una ventaja en la utilización de un circuito integrado, diseñado específicamente para la adquisición de señales fisiológicas y en particular de EEG. Su integración en un solo chip elimina la necesidad de seleccionar y conectar números componentes discretos, lo que simplifica el diseño del dispositivo y permite reducir el tamaño total del circuito.

Los circuitos integrados son diseñados y fabricados utilizando tecnologías avanzadas, lo que les permite ofrecer un rendimiento superior en términos de velocidad, precisión y consumo de energía en comparación con los circuitos discretos equivalentes. Además, al estar integrados en un solo chip, se reducen las conexiones físicas y se minimizan los problemas de interferencia y ruidos, mejorando la confiabilidad y estabilidad del circuito.

Costos reducidos. Si bien los circuitos integrados pueden tener un costo inicial más alto en comparación con los componentes discretos, a largo plazo pueden resultar más económicos. La reducción en el tiempo de diseño, la facilidad de fabricación, el menor costo de montaje y la menor necesidad de componentes adicionales pueden contribuir a una reducción global en los costos de producción.

La funcionalidad es otra de las ventajas evaluadas. Los circuitos integrados ofrecen una amplia gama de funciones y características adicionales en un solo dispositivo. Por ejemplo, incluyen periféricos integrados tales como convertidores analógico digital, interfaces de comunicación serial, detección de conexión de electrodos, etc. Esto permite agregar funcionalidades avanzadas al sistema sin la necesidad de implementar circuitos adicionales.

En función de las ventajas de los circuitos integrados frente a la implementación discreta, se evalúa en la *tabla 3.2.1.a* las características de dos adquisidores similares de señales biomédicas

Fabricante	Texas Instruments	Intan Technologies
Modelo	ADS1299 [8]	RHD2216 [9]
Canales	4	16

Resolución	24 bits	16 bits
Data rate	250sps - 16ksps	_ 30ksps
CMRR	110dB @ 50Hz	82dB
Ganancia (V/V)	1, 2, 4, 6, 8, 12, 24	192
Comunicación	SPI	SPI
Ruido intrínseco	1,39 μ Vpp@70Hz BW	2,4 μ Vrms
Impedancia de entrada	> 1000M DC	1300M@5H
Valor	US\$29,00	US\$286,00

tabla 3.2.1. a. Comparación de chips.

En vista de las características de ambos circuitos integrados y los requerimientos funcionales del proyecto se optó por seleccionar el dispositivo *ADS1299* de *Texas Instruments*.

3.2.2 ADS 1299

El *ADS1299 – 4* es un *convertor – analógico digital* de cuatro canales simultáneos con arquitectura *Sigma – Delta*. Tiene una resolución de *24 bits* y está diseñado específicamente para la adquisición de señales biomédicas, dentro de las cuales se incluyen: electroencefalografía *EEG*, electro cardiología *EEG*, monitoreo del sueño, índice bi-espectral *BIS* y potenciales de audio evocado *EAP*.

Posee en sus entradas cuatro amplificadores diferenciales con ganancia programable, las cuales pueden seleccionarse modificando los valores de ciertos registros, entre 1; 2; 4; 6; 8; 12 o 24.

Posee un ruido referido a la entrada de aproximadamente $1\mu V_{pp}$, definido para un ancho de banda de *70Hz*, una corriente de polarización máxima de $300\rho A$ y un *CMRR* mínimo de $-110dB$ a *50Hz*.

Cuenta con una red de multiplexado que posibilita realizar diversas acciones, tales como: medir la temperatura del integrado; medir las tensiones de alimentación analógica y digital;

seleccionar el electrodo de referencia, en el modo referencial; inyectar una señal de prueba en un canal; evaluar la impedancia de un electrodo; medir el ruido intrínseco o inyectar la señal de polarización, entre otras.

La comunicación hacia el exterior se realiza a través de una interfaz serial como *SPI* o *I2C*. Puede operar con velocidades de transferencia de datos de entre 250 *SPS* a 16.000 *SPS*, estas velocidades modifican el ancho de banda del filtro digital sinc según *tabla 3.2.2*.

<i>SPS</i>	<i>-3dB AB (Hz)</i>
250	65
500	131
1000	262
2000	524
4000	1048
8000	2096
16000	4193

Tabla 3.2.2 AB filtros digitales

3.2.3 Red de multiplexado

En la *figura 3.2.3.a* se puede observar un bloque del multiplexor de uno de los canales. Las distintas configuraciones del multiplexor se realizan a partir de la modificación de los registros del *ADS1299*. Estos registros son modificados por el usuario a través de la interfaz gráfica *GUI*. Se listan a continuación y se realiza una breve descripción de cada una de las opciones de multiplexado utilizados en el proyecto. Para más información de la red de multiplexado ver [apéndice A.1](#)

- Normal: esta configuración es la utilizada para realizar un estudio de *EEG*.
- Medición de ruido intrínseco del dispositivo: se realiza para obtener una medida del ruido propio del dispositivo.

- Señales de testeo: se trata de señales generadas internamente que sirven para verificar la cadena de señal interna.
- Señales de excitación para la detección de conexión del electrodo: esta configuración se utiliza para enviar al electrodo y al paciente una corriente generada internamente
- Selección de la fuente de polarización: esta configuración permite seleccionar entre una fuente interna o externa para polarizar al paciente.
- Medición de la señal de polarización: esta configuración rutea la señal obtenida en el pin *BIASIN*
- Ruteo de la señal de bias: esta configuración permite derivar la señal de polarización por otro electrodo. Si bien dicha señal tiene su electrodo propio, esta configuración permite derivarla por otro electrodo en caso de falla
- Ruteo de *SRB1* para el modo referencial: con esta configuración se logra conectar todas las entradas *INn* de todos los canales con un mismo electrodo de referencia conectado en *SRB1*

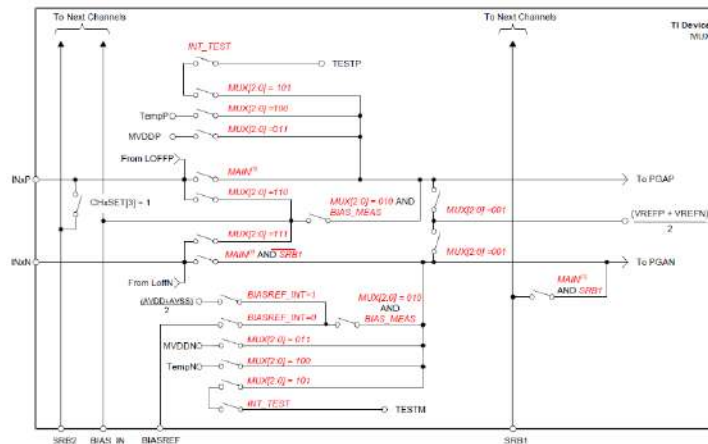


Figura 3.2.3. a Multiplexado ADS. Fuente "Texas Instruments"[8]

3.2.4 Entradas analógicas

El *ADS1299* – 4 cuenta con cuatro entradas analógicas que conectan directamente con los amplificadores internos de bajo ruido, alta impedancia y ganancia programable *figura 3.2.4. a*. Estos amplificadores son puramente diferenciales y se encuentran integrados dentro del chip.

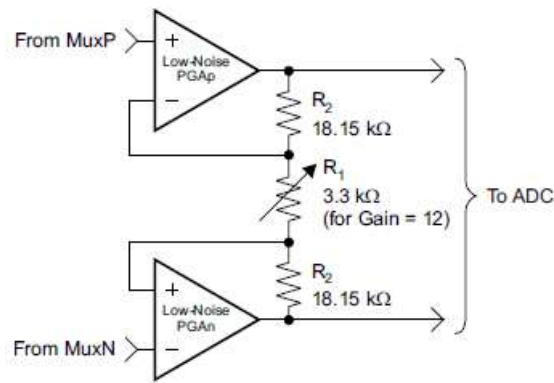


figura 3.2.4. a PGA. Fuente "Texas Instruments"[8]

Cada canal tiene ganancia programable. Esto permite adaptar la sensibilidad del amplificador a las características de la señal que se está midiendo. La elección de la ganancia del amplificador se realiza por medio de la interfaz gráfica. Se pueden seleccionar ganancias de entre siete valores posibles (1; 2; 4; 6; 8; 12; 24). Los valores de ganancia determinan los anchos de banda del amplificador según la *tabla 3.2.4.*

Ganancia	AB (Khz)
1	662
2	332
4	165
6	110
8	83
12	55
24	27

Tabla 3.2.4. Ganancia Vs. Ancho de banda

3.2.5 Señal de polarización

Dado que el *ADS1299* está alimentado con una fuente simple, es necesario polarizar al paciente de forma tal de suministrarle, a las señales de *EEG* a relevar, un voltaje común. Con

dicho voltaje se consigue que el amplificador trabaje en su máximo rango, al estar polarizado en la mitad del suministro, y en zona lineal.

Para lograr esto, el ADS1299 dispone de un amplificador de polarización dedicado a tal tarea. El encendido de este se realiza a través de la interfaz gráfica.

Es sabido que en los sistemas de *EEG*, una gran cantidad de interferencia electromagnética se acopla al sistema a través del cuerpo del paciente, los cables de los electrodos y la circuitería del dispositivo. Para disminuir la incidencia del ruido de modo común, el dispositivo ADS1299 puede utilizar el amplificador de polarización. A través de la interfaz gráfica se puede activar esta funcionalidad. Al activarla, el amplificador de polarización inyecta en el paciente la señal común presente en los electrodos, pero de forma invertida y escalada. Con esta técnica se logra una disminución considerable del ruido generado por las interferencias eléctricas.

En la *figura 3.2.5. a* se puede observar el esquema de la sección de polarización del ADS1299. Según se ve en la figura, la señal de realimentación está conformada por la salida de los amplificadores de entrada. Estas señales se realimentan a través del amplificador de bias, cuando se activan a través de la interfaz gráfica, los registros adecuados. A medida que más cantidad de señales provenientes de los electrodos se realimentan el rechazo a las señales comunes mejora.

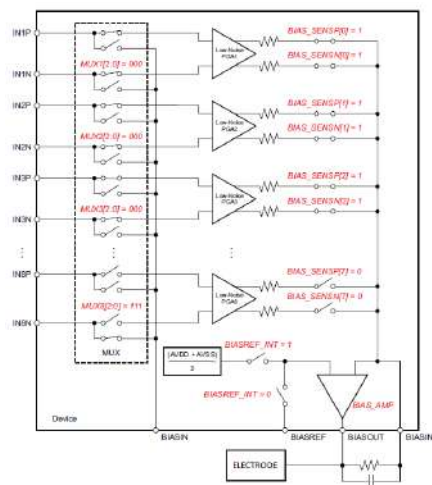


figura 3.2.5. a. Esquema del amplificador de BIAS. Fuente "Texas Instruments"[8]

3.2.6 Detección de conexión de electrodos

La detección del estado de la conexión del electrodo con el paciente es fundamental en cualquier estudio de *EEG*. Una buena medición de la señal de electroencefalografía depende de un camino de baja impedancia desde el cuerpo del paciente hasta el dispositivo de medición. Si hay alguna interrupción entre el cuerpo y el dispositivo, es posible que los resultados informados no se correspondan con precisión a la fisiología del paciente.

De las distintas estrategias para la detección de la conexión del electrodo, en este proyecto se utiliza el método de corriente continua. Este método proporciona un monitoreo del estado de los electrodos durante el estudio y tiene un efecto menor sobre la señal de *EEG*.

Para realizar el monitoreo, el *ADS1299* dispone, para tal tarea, de una serie de fuentes de corriente y comparadores *figura 3.2.6. a*. El proceso de monitoreo comienza proyectando la corriente, generada por las fuentes internas, a través de los electrodos de un mismo canal. Esta corriente al entrar en contacto con la carga genera un voltaje, el cual es comparado con un valor de referencia. La activación de la detección de desconexión de los electrodos, como así el valor de la intensidad de las fuentes de corriente y el valor del voltaje de comparación se realizan modificando los registros pertinentes a través de la interfaz gráfica. Durante el estudio, cuando el voltaje generado por la corriente de detección es superior a la tensión de referencia, se activan sobre la interfaz gráfica unos indicadores de mala conexión del electrodo.

Se debe tener en cuenta que estas corrientes generan, sobre los amplificadores de ganancia programable, un nivel de tensión de continua. Este nivel de tensión varía en función del estado de la conexión de los electrodos, siendo menor cuando la conexión es buena (de baja impedancia) y pudiendo saturar el amplificador cuando hay desconexión de los electrodos.

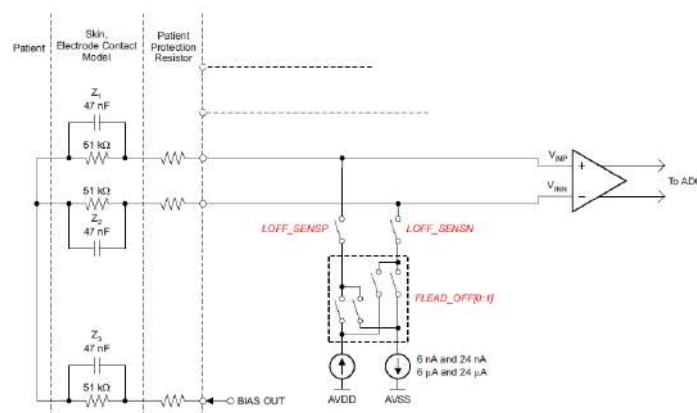


figura 3.2.6. a Lead Off Detection. Fuente "Texas Instruments"[8]

3.2.7 Detección de conexión del electrodo de polarización

El electrodo de polarización *BIAS* también posee un comparador y una fuente de corriente para detectar la correcta conexión del electrodo. Mediante la escritura de los registros adecuados, a través de la interfaz gráfica, se realiza la medición del electrodo de polarización. Se debe tener en cuenta que en este caso la medición no puede efectuarse durante el estudio, ya que el amplificador de *BIAS* debe permanecer apagado para realizar la medición de desconexión del electrodo.

Dado que las tierras del paciente y del dispositivo adquisidor son distintas, esta medición debe hacerse en conjunto con la detección de desconexión de electrodo de alguno de los canales restantes, ya que se debe asegurar un circuito de retorno de la corriente para obtener una medición certera.

3.2.8 ADC Sigma Delta

Los conversores *Sigma – Delta* son un tipo de arquitecturas de conversores analógico digital que presentan una serie de ventajas frente a los conversores tradicionales. Estos conversores son ampliamente utilizados cuando se desea digitalizar señales de variación lenta con una alta resolución [10] [11]. Para lograr esto el convertor utiliza técnicas de sobre muestreo, modelado de ruido, filtrado y decimación digital.

El funcionamiento se basa en el sobre muestreo de la señal a adquirir, es decir a una frecuencia f_s mucho mayor que la mayor frecuencia de la señal de entrada. El modulador convierte la señal analógica de entrada en un flujo de bits digitales, este flujo de bits es luego pasado por un filtro de decimación digital. La *figura 3.2.7. a* muestra el diagrama en bloques típico de un ADC sigma-delta.

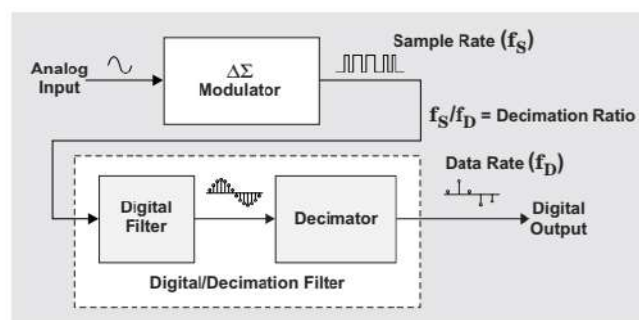


Fig. 3.2.7. a. Fuente "B. Baker, Texas Instruments"[11]

En el proceso de conversión sigma-delta, el filtro digital recibe la salida del modulador sigma-delta y reduce la tasa de datos mediante la técnica de decimación. La cantidad de filtrado se puede ajustar para lograr un equilibrio entre la resolución y la tasa de datos: se puede aumentar el filtrado para lograr una mayor resolución o reducir el filtrado para aumentar la tasa de datos. El filtro digital en cada canal consta de un filtro sinc de tercer orden. La relación de decimación del filtro sinc es $(N = \frac{f_{MOD}/f_{DR}}{f_{DR}}$, donde $f_{MOD} = \frac{f_{CLK}}{2}$)[11]. La tasa de datos f_{DR} se puede ajustar mediante los bits DR en el registro $CONFIG1$. Esta configuración es una configuración global que afecta a todos los canales y, por lo tanto, todos los canales operan a la misma tasa de datos en el dispositivo. Para mas información consultar el [apendice A.4](#)

3.2.9 Filtro digital Sinc

El filtro sinc es un filtro paso bajo de tercer orden con tasa de decimación variable ($N = f_{MOD}/f_{DR}$). Los datos se suministran a esta sección del filtro desde el modulador a la velocidad de f_{MOD} . El filtro sinc atenúa el ruido de alta frecuencia generado por el modulador y luego reduce la tasa de datos en paralelo. La tasa de decimación afecta la tasa global de datos del convertidor. En resumen, el filtro sinc en un convertidor sigma-delta es un filtro de paso bajo de orden variable que reduce la tasa de datos y atenúa el ruido de alta frecuencia generado por el modulador.

$$|H(f)| = \left| \frac{\sin\left(\frac{N\pi f}{f_{MOD}}\right)}{N \times \sin\left(\frac{\pi f}{f_{MOD}}\right)} \right|^3$$

FUNCIÓN TRANSFERENCIA DEL FILTRO SINC[8]

3.2.10 Requerimientos de filtrado

Es necesario colocar en cada una de las entradas de los amplificadores diferenciales del $ADS1299$ un filtro pasa bajos. Estos filtros tienen como objetivo evitar el solapamiento, eliminar la mayor cantidad posible de energía de RF de las líneas de entrada, mantener una impedancia de entrada lo suficientemente alta sobre el ancho de banda de medición y proporcionar una protección adicional contra sobrecarga en la entrada.

Para cumplir con los objetivos del filtrado del proyecto se propone utilizar el esquema que se muestra en la *figura 3.2.10. a.*

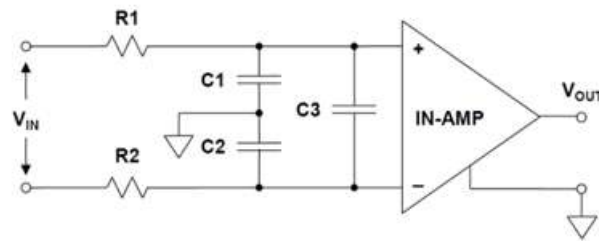


figura 3.2.10. a. Etapa de filtrado analogico[12]

Las ramas $R1$, $C1$ y $R2$, $C2$ conforman los filtros de modo común con una $f_{comun} = 15,4 \text{ KHz}$ cada uno, y el lazo $R1$, $C3$ y $R2$ el filtro diferencial con una $f_{dif} = 769.6 \text{ Hz}$.

Para la selección de la frecuencia de corte diferencial se tuvieron en cuenta los siguientes factores: máxima frecuencia de la señal a adquirir, forma de la banda de transición del filtro y la frecuencia de muestreo del conversor analógico digital. El criterio para la selección de la frecuencia de corte de los filtros de modo común fue que estas estuvieran por lo menos una década por encima de la frecuencia de corte diferencial. Los cálculos se desarrollan en el [apéndice A.6](#)

3.3 Microcontrolador

3.3.1 Introducción

Es el núcleo procesador del sistema, se encarga de recibir las muestras digitales provenientes del *ADS1299*, encapsularlas y luego enviarlas por *Wifi* a la computadora donde corre la interfaz gráfica *GUI*. También es el encargado de enviar los datos para modificar los registros de configuración necesarios del *ADS1299*, monitorear el estado de carga de la batería e indicar al usuario, a través de un led y la interfaz gráfica, el estado de esta.

3.3.2 Elección del Microcontrolador

Se decidió utilizar el dispositivo *ESP32* de *Espressif Systems* [13] para el procesador del proyecto. Dicho dispositivo pertenece a la familia de chips *SoC* de bajo costo y bajo consumo de energía.

La selección se basó principalmente en las siguientes características:

- Contiene los circuitos necesarios para la conectividad inalámbrica. De modo de ahorrar espacio en la *PCB* .
- Cuenta con comunicación *SPI* , necesaria para la comunicación con el dispositivo *ADS1299*.
- Contiene una cantidad suficiente de *GPIOs*, los cuales pueden ser utilizados para proyectos futuros.
- Posee conversores *analógico – digital*. Es necesario uno para medir el nivel de la batería.
- Una cantidad de memoria suficiente para almacenar el programa necesario para el proyecto.
- La existencia de una extensa cantidad de documentación y soporte ya sea en hojas de datos, manuales técnicos y notas de aplicación.
- Disponibilidad y precio accesible.
- Posee la capacidad de correr *MicroPython* , siendo este el lenguaje de programación utilizado en el proyecto.

3.3.3 ESP32

El dispositivo *ESP32* , *figura 3.3.3.a* , está basado en un microcontrolador *Tensilica Extensa LX6* de dos núcleos puede operar a una frecuencia de hasta *240MHz*. En la *figura 3.2.3.b* se pueden observar todos los bloques funcionales que conforman el *SoC ESP32*.

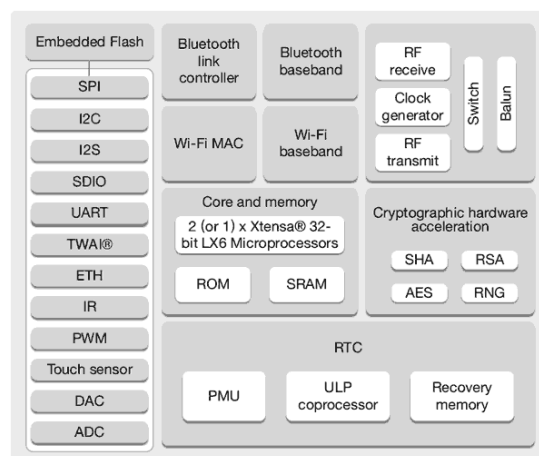


Figura 3.3.3. a. Bloques ESP32. Fuente "hoja de datos ESP32" [13]

- Conectividad inalámbrica: es compatible con 802.11 *b/g/n* en la banda de los 2.4 GHz. Alcanza velocidades de hasta 150 *Mbits/seg*.
- Bloque de radio: integra toda la circuitería necesaria para transformar la señal digital proveniente del módulo *Wifi* y convertirla en una señal electromagnética. También tiene la capacidad de recibir las ondas electromagnéticas provenientes de otros dispositivos para luego digitalizarlas.
- Núcleo: cuenta con dos microprocesadores de bajo consumo *Tensilica Xtensa* de 32 bits *LX6*. Además, cuenta con un *co – procesador* de ultra baja potencia para realizar conversiones *analógico – digitales* y otras operaciones mientras el dispositivo se encuentra en el modo de bajo consumo.
- Memorias:
 - ROM de 448KB donde se almacenan los códigos necesarios para el control de la capa física de *Wifi*, algunas rutinas de propósito general y el bootloader para iniciar el código de la memoria externa.
 - SRAM Interna 520KB utilizada por el procesador para almacenar datos e instrucciones de forma rápida.
 - RTC SRAM 16KB utilizada por el co-procesador cuando el dispositivo se encuentra en modo sleep.
 - Flash embebida de 4MB, para este proyecto, donde se almacena el código del programa.
- Pines digitales: cuenta con 34 pines digitales de entrada salida, la mayoría de estos con *pull – up / pull – down* internos y también con la posibilidad de colocarlos en alta impedancia.
- Conversores: cuenta con un conversor *analógico – digital* de 12 bits de 18 canales y un conversor *digital – analógico*.
- SPI: contiene 3 buses totalmente funcionales con los cuatro modos de transferencia y un *Buffer* de 64 bytes para transmisión y recepción.

3.3.4 Comunicación entre AFE y microcontrolador

Para realizar la comunicación entre los dispositivos *ADS1299* y *ESP32*, se utiliza en este proyecto la interfaz de comunicación serie *SPI*. [14] Este estándar de comunicación se trata de

un enlace de datos en serie, síncrono y full dúplex. Utiliza un modelo maestro-esclavo y opera con velocidades de reloj de hasta 50Mhz.

El *SPI* utiliza cuatro líneas de señal, en este proyecto el *ESP32* es el maestro y suministra y controla las líneas de reloj (*SCLK*) y de selección de chip (*CS*). Las dos líneas restantes se utilizan para enviar datos desde el maestro al esclavo (*MOSI*) y desde el esclavo al maestro (*MISO*). En la *figura 3.3.4.a* se pueden ver las conexiones entre el *ADS1299* y el *ESP32*.

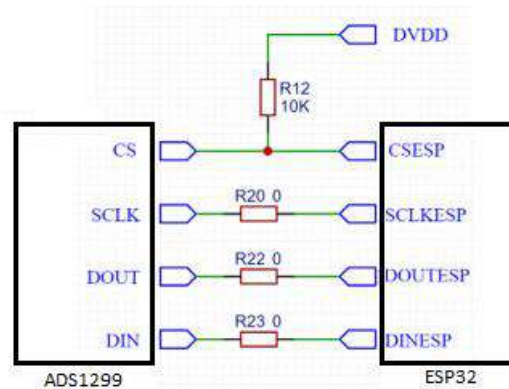


figura 3.3.4.a. Conexiones SPI

La transmisión posee cuatro modos, *figura 3.3.4b*, de cronometro. Los atributos que pueden modificarse del reloj son la polaridad (*CPOL*) y la fase (*CPHA*). *CPOL* establece el nivel lógico del reloj cuando no está siendo utilizado y *CPHA* establece en que flanco del reloj se transmiten los datos. Para este proyecto se utilizó el modo $CPOL = 0$, $CPHA = 1$ para el funcionamiento del *SPI* requerido por el *ADS1299*. En este modo, el reloj en su estado inactivo se encuentra en un nivel lógico bajo y la lectura de los datos se realiza en el flanco descendente.

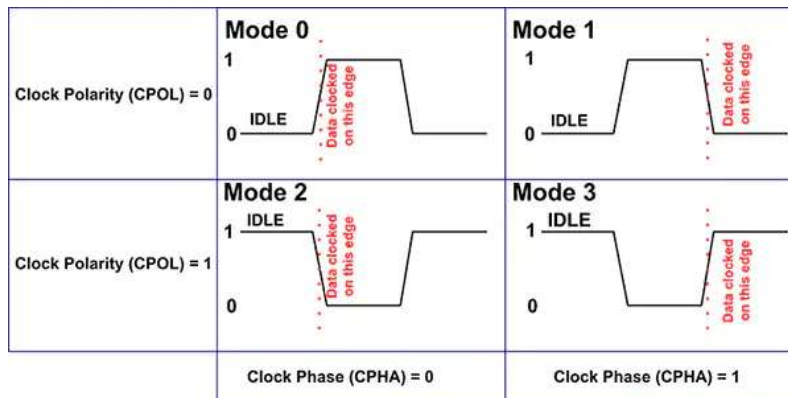


Figura 3.3.4. b. Modos de funcionamiento del SPI. Fuente "Internet"

3.3.5 Comunicación entre microcontrolador y PC

La comunicación entre el dispositivo *EEG* y la computadora se realiza a través de una comunicación *Wifi*. Se decidió utilizarlo por ser un protocolo muy difundido y utilizado, por lo cual muchos problemas ya fueron detectados y solucionados. Tiene como ventajas un *throughout* mayor al necesario, y un alcance suficiente para las necesidades de este proyecto. Como desventajas se pueden mencionar: el alto consumo eléctrico y la interferencia con otros canales cercanos.

En este proyecto se utiliza el ESP32UE *figura 3.3.5. a*. Esta versión del dispositivo cuenta con un conector *ipex* para colocar la antena.



figura 3.3.5. a. ESP32UE. Fuente internet

3.3.6 Circuito cargador de arranque

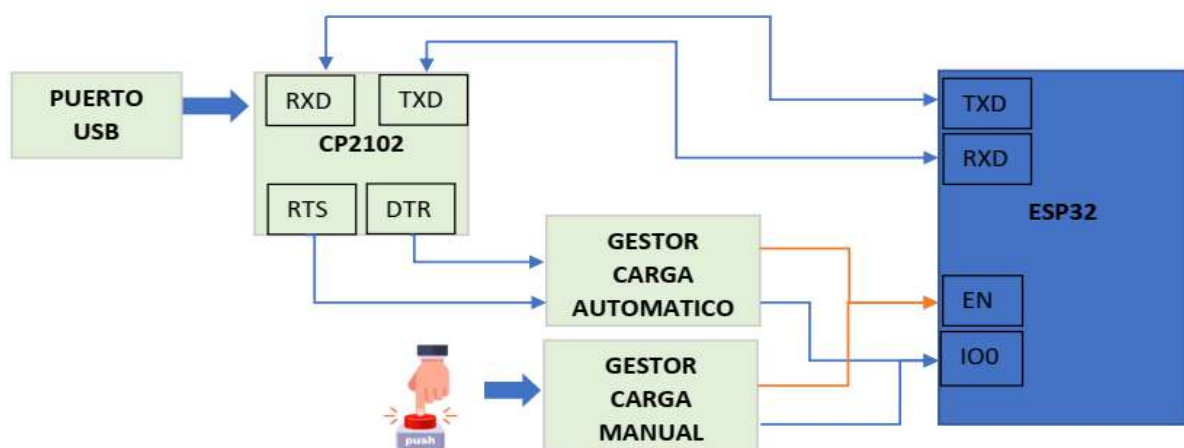


figura 3.3.6. a. Esquema programador

En esta sección se describe el circuito que permite cargar el firmware de MicroPython [30] y las aplicaciones desarrolladas para el dispositivo.

El ESP32 posee un gestor de arranque nativo al cual se accede colocando las entradas digitales Enable(EN) e IO0 en cierto estado, esto permite cargar el firmware de MicroPython [15] en la memoria del dispositivo y cargar las aplicaciones desarrolladas.

De acuerdo con las especificaciones y recomendaciones del fabricante Espressif [16], se implementaron los circuitos esquematizados en la *figura 3.3.6. a.*

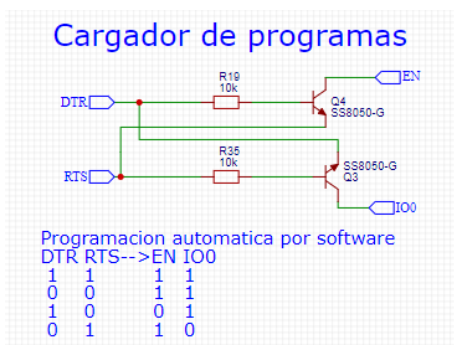
El puerto USB es necesario para la conexión con la computadora, brinda la alimentación y la comunicación de datos durante el modo de programación del microprocesador.

El integrado CP2102 es un convertidor de protocolo USB a Serie [17], que facilita la comunicación entre una PC y un microcontrolador a través de los puertos RXD y TXD. RTS y DTR son pines de control de flujo del puerto serie y son utilizadas para acceder al gestor de arranque del microcontrolador manejando las entradas digitales EN e IO0 del mismo.

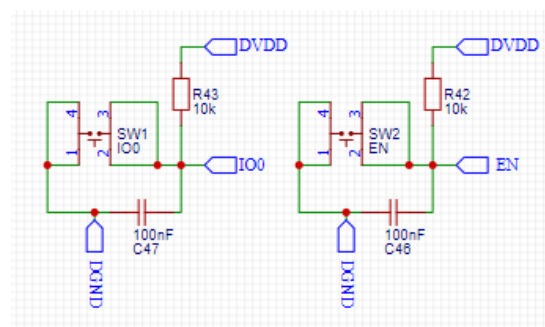
El ESP32 ingresará al cargador de arranque cuando IO0 se mantenga bajo en el reinicio. De lo contrario, ejecutará el programa en flash [18].

El gestor de carga automático de la *figura 3.3.6.b* consta de una configuración de dos transistores, las señales DTR y RTS son controladas mediante el protocolo USB por la herramienta esptool.py y realiza la secuencia de ingreso al cargador de arranque en forma automática.

El gestor de carga manual *figura 3.3.6.c* fue desarrollado dado que cuando se inició el proyecto el IDE Thonny, utilizado para la programación del ESP32, no contaba con soporte para la gestión automática del gestor de carga, mediante estos dos botones debía mantenerse presionado el botón de SW1 conectado al IO0, mientras se pulsaba el botón SW2 conectado en el pin de Enable (EN).



3.3.6. b. Gestor de carga automatico.



3.3.6. c. Gestor de carga manual.

3.4 Alimentación eléctrica

3.4.1 Requerimientos

El ADS1299 requiere para su funcionamiento de dos tensiones. Una tensión analógica que puede ser suministrada con una fuente bipolar de $-2.5V$ y $2.5V$ o una fuente única de $5V$ y una tensión digital que debe estar en el rango de $1.8V$ a $3.6V$.

		MIN	NOM	MAX	[U]
Voltaje	AVDD – AVSS	4.75	5	5.25	V

	<i>DVDD – DGND</i>	1.8	1.8	3.6	<i>V</i>
	<i>AVDD – DVDD</i>	-2.1		3.6	<i>V</i>
<i>Corriente</i>	<i>Corriente AVDD</i>		4.06		<i>mA</i>
<i>BIAS OFF</i>	<i>Corriente DVDD</i>		0.54		<i>mA</i>

Tabla 3.6.1.a Características eléctricas ADS1299

El fabricante recomienda para la fuente de alimentación analógica que esta sea lo más estable posible de modo de no afectar el rendimiento del conversor analógico digital.

El *ESP32* necesita para su funcionamiento una tensión de *3.3V*.

		<i>MIN</i>	<i>NOM</i>	<i>MAX</i>	[<i>U</i>]
<i>Voltaje</i>	<i>VDD</i>	3.0	3.3	3.6	<i>V</i>
<i>Corriente</i>	<i>Corriente VDD</i>	0.5			<i>A</i>

Tabla 3.6.1.b Características eléctricas ESP32

3.4.2 Evaluación de alternativas

Se evaluaron las siguientes alternativas para suministrar energía al dispositivo:

Alternativa 1

Alimentar el circuito con dos baterías de *Li – Ion* en serie. De esta forma se pretendían obtener dos voltajes nominales uno de *3.7* y otro de *7.4V*, conformado por la disposición en serie de las dos baterías. Partiendo de los *7.4V* y por medio de un *LDO* de *5V* obtener la tensión

de alimentación analógica. La tensión digital se obtendría del voltaje de $3.7V$, pasando por un *LDO* de $3.3V$.

Ventajas analizadas:

- La ventaja de esta implementación radica en la no utilización de fuentes conmutadas para la alimentación y con ello la disminución de ruido eléctrico.

Desventajas analizadas:

- Considerando que el consumo de la etapa digital es sustancialmente mayor al de la etapa analógica, se tenía una descarga asimétrica de las baterías.
- El tamaño físico del dispositivo se incrementaba al utilizar dos baterías.
- La diferencia de tensión entre los $7.4V$ de la tensión nominal de las baterías con el necesario para alimentar la tensión analógica daba un rendimiento pobre[19], situación que también se daba en la alimentación digital.

Alternativa 2

Alimentar el circuito con una sola batería de *Li – Ion*. Para esta implementación era necesario una fuente conmutada para conseguir una tensión de $5V$ a partir de los $3.7V$ de la batería y un regulador de voltaje *LDO* de $3.3V$ para la tensión digital.

Ventajas analizadas:

- La ventaja de esta implementación es la utilización de solo una batería de *Li – Ion*, lo que hace posible un tamaño del dispositivo mas reducido y un costo menos elevado, dos requerimientos propuestos por el cliente.

Desventajas analizadas:

- El ruido generado por la fuente conmutada podía deteriorar el rendimiento del conversor analógico digital del ADS1299.
- Necesidad de utilizar filtros para disminuir el ruido generado por la fuente.

Alternativa 3

Alimentar el circuito con una sola batería de *Li – Ion*. Utilizar para la tensión analógica una fuente bipolar de $\pm 2.5V$. Para esta implementación era necesario, a partir de los $3.7V$ de la

batería, utilizar un inversor de voltaje para obtener una tensión negativa. También era necesario utilizar tres reguladores lineales uno de $-2.5V$ y otro de $2.5V$ para la tensión analógica y un regulador más de $3.3V$ para la tensión digital.

Ventajas analizadas:

- La ventaja de esta implementación es la utilización de solo una batería de *Li – Ion*, lo que hace posible un tamaño del dispositivo más reducido y un costo menos elevado, dos requerimientos propuestos por el cliente.

Desventajas analizadas:

- El inversor de voltaje utiliza conmutación para generar el voltaje negativo. El ruido generado por la fuente conmutada podía deteriorar el rendimiento del conversor analógico digital del ADS1299.
- Necesidad de utilizar filtros para disminuir el ruido generado por la fuente.
- Necesidad de utilizar tres LDO, lo que conlleva a un aumento de la cantidad de componentes electrónicos.

3.4.3 Elección de batería

Para la elección de la batería se consideró una autonomía de 2 hs . Se seleccionó para el proyecto una batería de *Li – Ion* (ion de litio) debido a su bajo costo y su alta disponibilidad en el mercado. Se consideró que la batería seleccionada contenga los circuitos de protección necesarios, sobrecarga, sobre-descarga y descarga rápida. El dimensionamiento se realizó estimando el mayor consumo de corriente (peor caso), información que fue provista por las hojas de datos del *ESP32* y el *ADS1299*, en la *tabla 3.4.3. a* se pueden observar los consumos de cada uno de los componentes. Al valor obtenido se le adiciono un 20% de margen.

$$\text{bateria} = 507\text{ mA} \times 1,20 \times 2\text{hs} = 1218\text{ mAh}$$

<i>Componente</i>	<i>Cant.</i>	<i>Consumo unitario</i>	<i>Total</i>
<i>ADS1299 AVDD</i>	<i>1</i>	<i>4,06 mA</i>	<i>4,06 mA</i>
<i>ADS1299 DVDD</i>	<i>1</i>	<i>0,54 mA</i>	<i>0,54 mA</i>
<i>ADS1299 BIAS AMP.</i>	<i>1</i>	<i>1,1 mA</i>	<i>1,1 mA</i>
<i>ESP32 DVDD</i>	<i>1</i>	<i>500 mA</i>	<i>500 mA</i>

<i>TOTAL</i>		<i>507,5 mA</i>	<i>507,5 mA</i>
--------------	--	-----------------	-----------------

tabla 3.4.3. a. Consumos eléctricos

3.4.4 Diseño de fuentes

En la *figura 3.4.4. b* se pueden observar los esquemas electrónicos de alimentación utilizados en este proyecto.

- DC – DC Step Up

Para conseguir el voltaje necesario para alimentar el dispositivo *ADS1299* se utilizó el circuito integrado *LM2704*[20], del fabricante Texas Instruments en conjunto con el circuito integrado *MCP1755S* [22]descrito abajo[21]. Este circuito opera de forma conmutada y puede generar un voltaje de salida ajustable, con un rango de entrada que va desde los *2,2 Volt* hasta los *7 Volt*. Para la selección del circuito integrado se consideró que la frecuencia de conmutación del circuito Step Up sea un múltiplo de la frecuencia de la salida de los datos digitales del *ADS1299*, para disminuir la interferencia eléctrica[8].

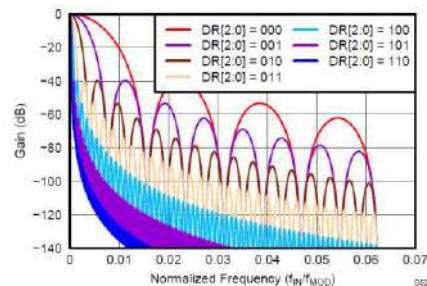


figura 3.4.4. a

Se dimensiono para que la corriente y el voltaje suministrados sean los requeridos por los circuitos analógicos. Los componentes periféricos al circuito integrado se seleccionaron en función de los datos suministrados en la hoja de datos y los requerimientos del dispositivo *EEG*. A continuación, se detallan los cálculos realizados.

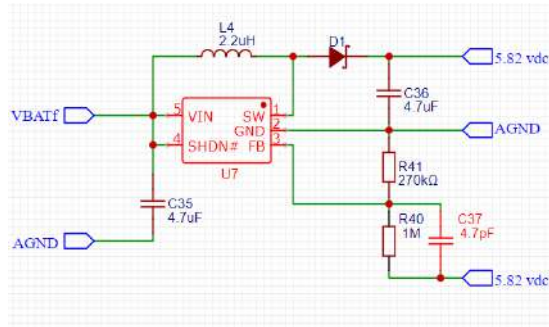


figura 3.4.4. b

Selección de R40 y R41 para determinar el voltaje de salida:

$$V_{OUT} = \left(\frac{R40}{R41} + 1 \right) \times 1,237 \text{ Volt} \cong 5,82 \text{ Volt}$$

Selección de L4:

$$L = \left(\frac{V_{OUT} - V_{INMIN} + V_D}{I_{CL}} \right) \times T_{OFF}$$

- DC – DC LDO 5V

Dadas los requerimientos de alimentación eléctrica necesarios para la etapa analógica del dispositivo ADS1299, se seleccionó un regulador lineal apropiado [21]. Se consideraron las características descritas en la tabla 3.4.4. a

Característica	Requeridos	MCP1755S
PSRR banda de medición	> 40 db	> 60 db
Dropout Voltage	< 300 mV	50 mV @ 50 mA
Output Voltage Tolerance	< 5 %	< 2%
Corriente mínima	30 mA	300 mA
Protección contra corto	Si	Si
Disponibilidad		Si

tabla 3.4.4. a

En vista de los requerimientos se seleccionó el circuito integrado MCP1755S. Las conexiones se realizaron según recomendaciones del fabricante.

- DC – DC LDO 3.3V

Dadas los requerimientos de alimentación eléctrica necesarios para la etapa digital, se seleccionó un regulador lineal apropiado. Se consideraron las características descritas en la *tabla 3.4.4. b*

<i>Caracterisitica</i>	<i>Requeridos</i>	<i>LD39150</i>
<i>Dropout Voltage</i>	$< 150\text{ mV}$	$100\text{ mV @ }600\text{ mA}$
<i>Ouput Voltage Tolerance</i>	$< 9\%$	$< 9\%$
<i>Corriente mínima</i>	500 mA	$1,5\text{ A}$
<i>Protección contra corto</i>	<i>Si</i>	<i>Si</i>
<i>Disponibilidad</i>		<i>Si</i>

tabla 3.4.4. b

En vista de los requerimientos se selección el circuito integrado *LD39150*[23]. Las conexiones se realizaron según recomendaciones del fabricante.

3.4.5 Gestión de carga de la batería

La gestión de carga de la batería de *Li – Ion* se realiza a través del circuito de la *figura 3.4.5* y gestionado por el circuito integrado *MCP73831/2* del fabricante *Microchip Technology*[24].

La carga de la batería se realiza únicamente si el dispositivo se encuentra apagado, con el interruptor en posición de carga, y si el nivel de batería lo requiere.

El circuito integrado alterna entre los modos de funcionamiento, dependiendo del nivel de carga que tenga la batería, en la [tabla 3.4.5](#) se pueden observar los parámetros de umbral.

- Carga lenta (baja corriente): si el nivel de la batería, V_{BAT} , es menor que el voltaje de umbral, V_{PTH} , entonces la carga se realiza con una corriente constante definida por el diez por ciento de I_{REG} .

- Carga rápida (alta corriente): cuando el nivel de batería, V_{BAT} , se encuentra por encima del voltaje umbral, V_{PTH} , la carga se realiza a corriente constante con un valor definido por I_{REG} .
- Carga a voltaje constante: cuando el nivel de batería, V_{BAT} , alcanza el valor de V_{REG} , el cargador sale del modo de carga rápida y comienza a cargar a voltaje constante.
- Terminación de carga: cuando el promedio de la corriente de carga disminuye por debajo del cinco por ciento de I_{REG} el proceso de carga finaliza.

El proceso de carga se indica en el dispositivo *EEG* a través de un diodo led (*LED2*) de color rojo, el cual se enciende mientras se realiza la carga. En la *figura 3.4.5* se puede ver el diagrama de conexión del gestor.

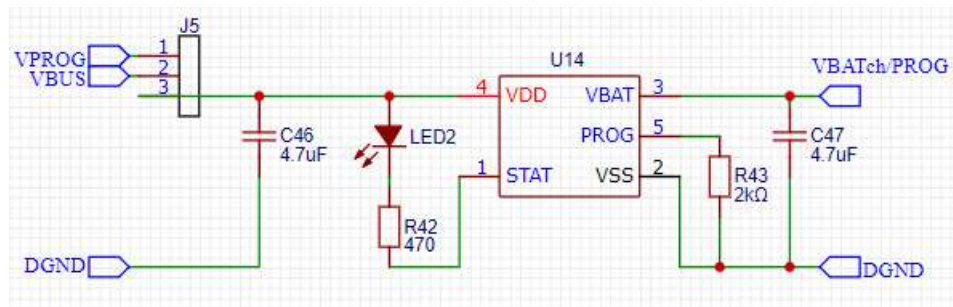


figura 3.4.5. Esquema de conexión del gestor de carga. Fuente "propia"

La corriente del diodo queda determinada por la *ecuación 3.4.5. a*, para este proyecto se decidió utilizar un corriente, lo más baja posible, que permitiera visualizar el led con iluminación normal en el ambiente.

$$\text{ecuación 3.4.5. a. } I_{LED2} = \frac{V_{DD} - V_{LED2}}{R_{42}} = \frac{5V - 1,7V}{470\Omega} \cong 7mA.$$

La corriente de carga rápida queda determinada por R_{43} según la *ecuación 3.4.5. b*, para este proyecto se decidió una corriente de carga de $500mA$.

$$\text{ecuación 3.4.5. b. } I_{REG} = \frac{1000V}{R_{43}} = \frac{1000V}{2K\Omega} = 500mA.$$

La máxima potencia disipada por el dispositivo queda determinada por la ecuación 3.4.5. c. Esto se da cuando la corriente y la diferencia de voltaje entre batería y voltaje de alimentación son máximos.

$$\begin{aligned} \text{ecuación 3.4.5. c. } P_{MAX} &= (V_{DDMAX} - V_{PTHMIN}) \times I_{REGMAX} = (5,5 - 2,7) \times 550\text{mAmp.} \\ &= 1,54 \text{ Watt} \end{aligned}$$

<i>Parámetro</i>	<i>Sim.</i>	<i>Valor típico</i>	<i>Uni.</i>	<i>Condición</i>
<i>Voltaje de salida</i>	V_{REG}	4.20	<i>Volt</i>	—
<i>Corriente de carga rapida</i>	I_{REG}	500	<i>mAmp.</i>	$R43$
<i>Corriente de carga lenta</i>	I_{PREG}	50	<i>mAmp.</i>	$0,1 \times I_{REG}$
<i>Corriente de terminacion</i>	I_{TERM}	25	<i>mAmp.</i>	$0,05 \times I_{REG}$
<i>Voltaje de umbral</i>	V_{PTH}	2,79	<i>Volt</i>	$0,665 \times V_{REG}$

tabla 3.4.5

3.5 Diseño de la PCB

La fabricación del circuito impreso y el montaje de los componentes fue realizada por la Empresa Ernesto Mayer S.A.

Para la realización del diseño se optó por un circuito impreso doble faz, con espesor de cobre de 1 Oz. Se utilizaron reglas de diseño en base a las recomendaciones del fabricante del ADS1299, Texas Instruments y notas de aplicación. [9] [25] [26]. Se adoptaron como reglas de diseño propias la utilización de trazas con un ancho mayor a 11 mils y curvas con ángulos no mayores a 45°, reglas que se obtuvieron considerando las recomendaciones de los fabricantes. Los componentes utilizados fueron todos de montaje superficial y encapsulado 0805 o mayor.

Para calcular el ancho de las trazas de los circuitos de alimentación se utilizaron las fórmulas de la IPC-2221[27]. Estas fórmulas determinan el ancho de un conductor de cobre, en una placa de un circuito impreso, necesario para transportar una determinada cantidad de corriente manteniendo un aumento de la temperatura de la traza por debajo de un límite especificado.

Se consideraron las recomendaciones de los fabricantes para la ubicación de los componentes periféricos a las fuentes de alimentación como así también las áreas de cobre necesarias para una correcta disipación del calor generado.

El circuito se diseñó para que los planos de masa analógico y digital quedaran separados de modo de minimizar interferencias. Ambos planos se dispusieron en ambas caras de la placa y se tuvo en cuenta la no formación de áreas de cobre aisladas.

Se separó el sector de fuentes de alimentación del *AFE ADS1299* y el microcontrolador *ESP32* del resto del circuito.

Las figuras 3.6.a y 3.6.b muestran los layout con sus respectivas conexiones en la capas superior e inferior respectivamente, diseñados utilizando el software *EasyEDA* [28]. .

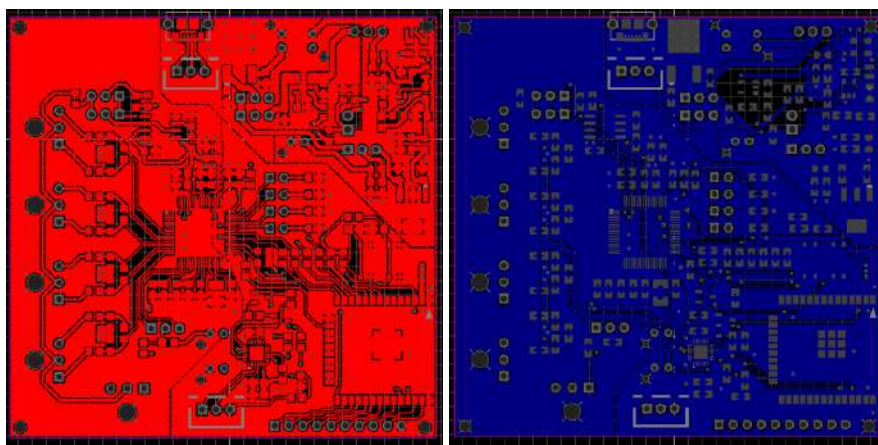


figura 3.6.a

figura 3.6.b

Capítulo 4

Diseño del software del dispositivo

4.1 Introducción

En el presente capítulo se documenta el proceso de diseño del software del dispositivo.

El software fue programado en MicroPython el cual es una implementación de Python-3 que se ejecuta en dispositivos microcontroladores [18]. Es un lenguaje de programación interpretado que permite la interacción con hardware, lo que lo hace ideal para proyectos de IoT (Internet de las cosas), robótica y automatización del hogar.

Para su programación se utilizó el entorno de desarrollo integrado (IDE) Thonny [29] el cual es compatible con MicroPython, gratuito y de código abierto, lo que significa que cualquiera puede descargarlo y modificarlo para adaptarlo a sus necesidades.

Se buscó realizar un programa modular y mantenible que permita cambios y mejoras a futuro. Como el software pertenece a un sistema donde los tiempos son críticos, este fue orientado al tiempo real.

Se diseñó para lograr la funcionalidad del dispositivo, esto implica la adquisición de señales y la comunicación con el PC.

En primer lugar, se expondrán las bibliotecas de MicroPython utilizadas, luego se presentarán los módulos del software implementados y después la estructura del programa.

4.2 Requerimientos

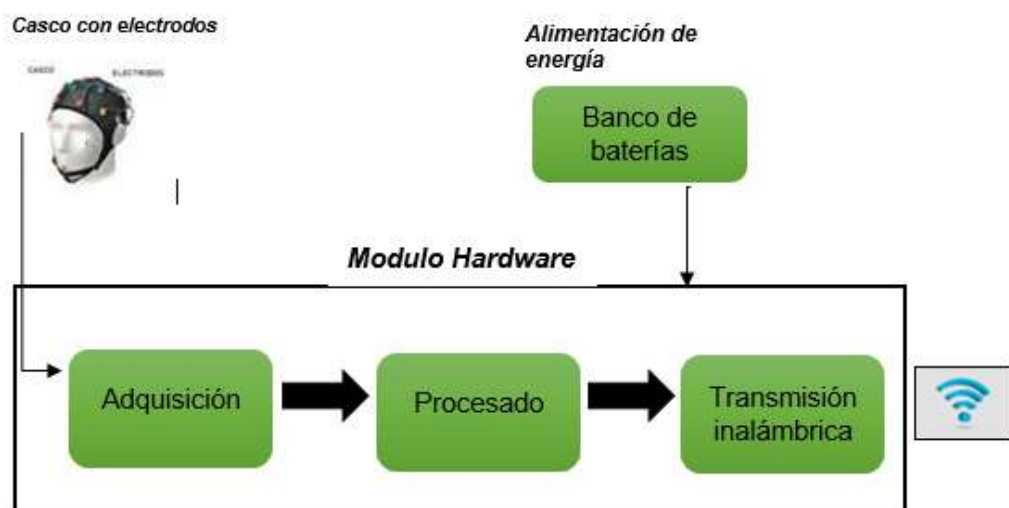


figura 4.2.a

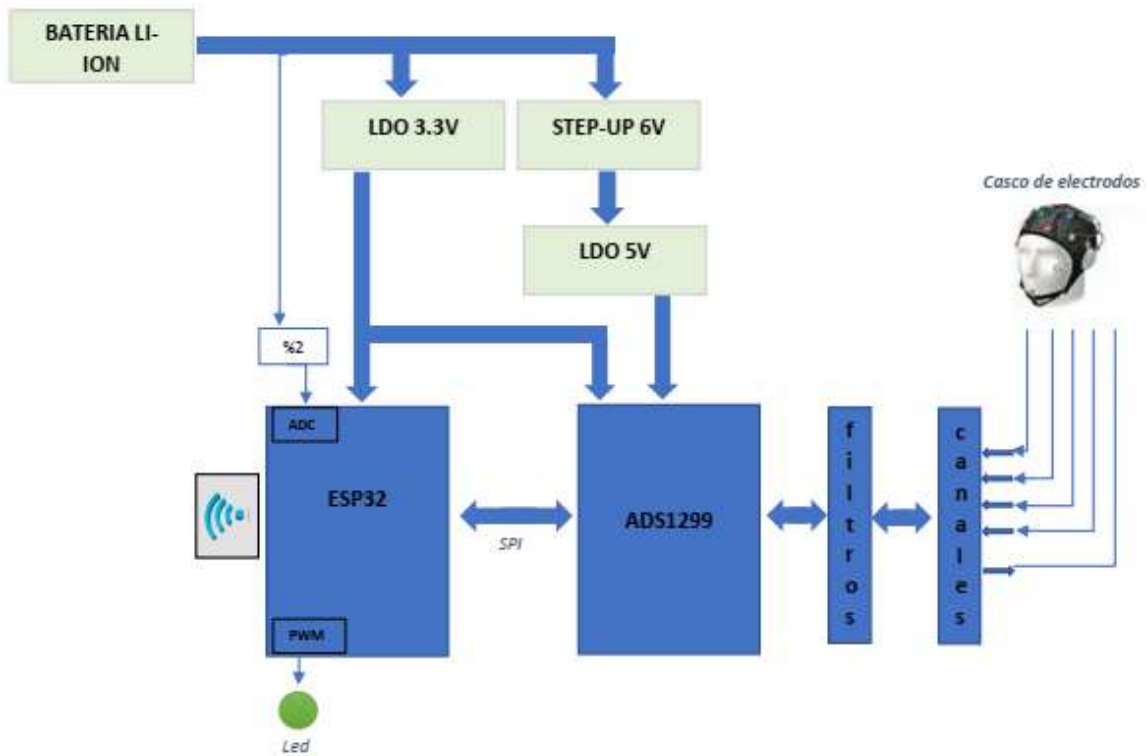


figura 4.2.b

La figura 4.2.a muestra el diagrama en bloques propuesto durante la especificación de requerimientos inicial, y la figura 4.2.b representa un diagrama funcional del dispositivo desarrollado.

El dispositivo genera una red wifi a la que el usuario puede conectarse, y a la cual el software cliente en la computadora envía comandos y datos para configurarlo y desde la cual recibe datos. El software es capaz de recibir estos mensajes, decodificarlos y responder con las acciones requeridas.

El dispositivo se alimenta con una batería de iones de litio de 3,7 volts, a la cual se le supervisa la carga mediante una conexión a un ADC que es el encargado de verificar la tensión durante su funcionamiento. El valor de la batería es informado mediante un led físico conectado en la placa del dispositivo e informado al software de la computadora cada vez que es solicitado. Los posibles estados del led dependen del estado del nivel de la batería y de la conexión del usuario a la red wifi. Al encender el dispositivo se mide el nivel de la batería, si este es menor a 3,4 volts el led destella con una frecuencia de 1 Hz para indicar un nivel de batería que requiere recarga.

De otro modo si la tensión medida es superior a 3,4 volts, el led destellara a una frecuencia de 10hz para indicar que el usuario no se ha conectado a la red wifi. Cuando el usuario se conecte

a la red generada, el led quedara visiblemente fijo hasta que el valor de batería disminuya del nivel fijado en 3,4 volts. A su vez el usuario dispondrá del nivel de batería en el software cliente de la computadora.

Como se describió en el capítulo 3, el dispositivo cuenta con un ESP32 encargado de la gestión de control y un AFE ADS1299 encargado de la conversión de las señales a sus valores digitales, la comunicación entre los mismos es a través de un puerto SPI el cual es gestionado por el ESP32. El software configura y controla el puerto SPI para la comunicación efectiva entre ambos dispositivos.

4.2.1 Requerimiento de tiempo del puerto SPI:

De acuerdo a la información brindada por el fabricante del ADS1299 [8], el periodo de la señal de reloj del puerto SPI debe cumplir con los requerimientos de la ecuación 4.2.1, esta señal es generada por el microcontrolador maestro ESP32.

El caso más desfavorable para el dispositivo desarrollado es con Nbits=24 (resolución de la muestra), Nchannels = 4, tdr=1/1000SPS y tclk=1/2.048 MHz, lo que da un tsclk= 8.32x10⁻⁶, o lo que equivale a un SCLK mínimo de 121KHz. Dado que el sistema lo permite, se configura a 2MHz, lo cual sería suficiente para 4 canales a 16KSPS.

$$t_{sclk} < \frac{t_{DR} - 4 * t_{CLK}}{N_{bits} * N_{channels} + 24}$$

Ecuación 4.2.1

4.2.2 Requerimiento de tiempo de envío de las señales y tamaño del mensaje:

El proceso de envío de señales y datos es un aspecto fundamental en muchas aplicaciones, especialmente en aquellas relacionadas con la monitorización y adquisición de datos en tiempo real. Se evaluaron dos estrategias para el envío de los datos, teniendo en cuenta el requerimiento de tiempo necesario para muestrear la señal a 1KSPS, enviar los datos, procesarlos y graficarlos. Asimismo, se discute el tamaño del mensaje y el tiempo de envío en cada una de las estrategias.

Estrategia 1: Enviar cada muestra. La primera estrategia consiste en enviar cada muestra de la señal en tiempo real. Esto implica el envío de 24 bits por canal, multiplicado por 4 canales, más 24 bits de estado, lo que da un total de 120 bits o 15 bytes de datos. Para cumplir con el requerimiento de tiempo de muestreo, esto debería realizarse en menos de 1 ms. Esta estrategia

es eficiente en términos de latencia, ya que los datos se envían en tiempo real, pero el tamaño del mensaje es muy pequeño, lo que implica un mayor número de paquetes que puede generar una carga mayor en la red y un mayor tiempo total de envío, y disminuyendo el tiempo disponible para procesar y mostrar los datos en pantalla.

Estrategia 2: Tamaño óptimo de paquetes. La segunda estrategia consiste en enviar paquetes de datos más grandes, con el objetivo de reducir la carga de la red y el tiempo total de envío. Para ello, se evalúa un tamaño óptimo de paquetes para la red wifi, que depende de la MTU (Maximum Transmission Unit). El valor predeterminado de MTU para la mayoría de las redes wifi es de 1500 bytes. Las cabeceras de las capas de protocolos pueden sumar entre 40 y 60 bytes, dependiendo del tipo de protocolo y el tamaño de los campos de la cabecera. Por lo tanto, si los datos de la aplicación tienen un tamaño de 1000 bytes, el tamaño total del paquete sería de alrededor de 1040 a 1080 bytes.

En esta estrategia, se propone encapsular 100 muestras de la señal junto con un archivo de cabecera de estados que incluya el encabezado y los registros de control de 24 bits. Este archivo estaría compuesto por un código de control OXC (4 bits), los registros de estado de LOFF (16 bits), 4 bits de los estados de los GPIO, y 8 bits con el valor codificado del nivel de batería. El encabezado tendría un tamaño de 4 bytes. Los datos de las 100 muestras darían un total de 24 bits por canal por 4 canales por 100 muestras, lo que da un total de 1200 bytes. El tamaño total del mensaje sería de 1244 bytes sumando las cabeceras de TCP-IP. Esta estrategia brindaría un tiempo de 100 ms a 1KSPS para enviar los datos, procesarlos y graficarlos, para el caso de 250SPS brindaría 400 ms.

4.3 Funcionamiento y formato de los datos del ADS1299

Para desarrollar el software embebido fue necesario conocer en detalle el funcionamiento y los modos de trabajo y configuración del ADS1299. A continuación, se brinda un resumen de los pines y comandos necesarios para el funcionamiento y un breve detalle del formato de los datos [8].

4.3.1 Interfaz SPI:

Chip Select (CS): El pin CS activa la comunicación SPI. CS debe estar bajo antes de las transacciones de datos y debe permanecer bajo durante todo el período de comunicación SPI. Cuando CS está alto, el pin DOUT entra en un estado de alta impedancia. Por lo tanto, la lectura y escritura en la interfaz serial son ignoradas y la interfaz serial se reinicia. La operación del pin

DRDY es independiente de CS. DRDY todavía indica que se ha completado una nueva conversión y se fuerza a un estado alto como respuesta a SCLK, incluso si CS está alto.

Serial Clock (SCLK): SCLK es el reloj para la comunicación serial. SCLK es una entrada de disparador de Schmitt, pero TI recomienda mantener SCLK lo más libre posible de ruido para evitar que las fallas inadvertidas muevan los datos.

Data Input (DIN): DIN se utiliza junto con SCLK para enviar datos al dispositivo.

Data Output (DOUT): DOUT se utiliza junto con SCLK para leer la conversión y los datos del registro del dispositivo.

4.3.2 Pines y comandos:

La figura 4-3 muestra una lista los comandos disponibles y la figura 4-7 la lista de los registros.

COMMAND	DESCRIPTION	FIRST BYTE	SECOND BYTE
System Commands			
WAKEUP	Wake-up from standby mode	0000 0010 (02h)	
STANDBY	Enter standby mode	0000 0100 (04h)	
RESET	Reset the device	0000 0110 (06h)	
START	Start and restart (synchronize) conversions	0000 1000 (08h)	
STOP	Stop conversion	0000 1010 (0Ah)	
Data Read Commands			
RDATAC	Enable Read Data Continuous mode. This mode is the default mode at power-up. ⁽¹⁾	0001 0000 (10h)	
SDATAC	Stop Read Data Continuously mode	0001 0001 (11h)	
RDATA	Read data by command; supports multiple read back.	0001 0010 (12h)	
Register Read Commands			
RREG	Read <i>n nnnn</i> registers starting at address <i>r rrrr</i>	001 <i>r rrrr</i> (2xh) ⁽²⁾	000 <i>n nnnn</i> ⁽²⁾
WREG	Write <i>n nnnn</i> registers starting at address <i>r rrrr</i>	010 <i>r rrrr</i> (4xh) ⁽²⁾	000 <i>n nnnn</i> ⁽²⁾

(1) When in RDATA mode the RREG command is ignored

FIG. 4-3 COMANDOS

Power-Down (PWDN): Pin de entrada. Cuando PWDN se lleva a bajo, toda la circuitería del chip se apaga. Para salir del modo de apagado, debe colocarse el pin PWDN en alto.

Reset (RESET): Existen dos métodos para resetear el ADS1299-x: colocar en cero el pin de RESET y luego en uno, o enviar el comando RESET por la interfaz SPI.

Data Ready (DRDY): Pin de salida. DRDY es una señal de salida que transita de alta a baja indicando que hay nuevos datos de conversión disponibles.

Start (START): Para comenzar las conversiones en el ADS1299-x, es necesario poner el pin de START en alto durante al menos 2 periodos tCLK, o enviar el comando START. Si START está en

bajo y no se ha enviado el comando START, el dispositivo no emite una señal DRDY (las conversiones se detienen).

Cuando se utiliza el comando START para controlar las conversiones, debes mantener el pin de START en bajo. El ADS1299-x cuenta con dos modos para controlar las conversiones: modo continuo y modo de un solo disparo. El modo de un solo disparo se habilita estableciendo el bit SINGLE_SHOT en el registro CONFIG4 en '1'. En este método, se realiza una única conversión al enviar el comando START.

En este desarrollo se utilizo modo continuo de conversiones. Las conversiones comienzan cuando se lleva el pin START a alto o cuando se envía el comando START, la salida DRDY se pone en alto cuando se inician las conversiones y se pone en bajo cuando los datos están listos para ser leídos en el puerto SPI. Las conversiones continúan indefinidamente hasta que se lleva el pin START a bajo o se transmite el comando STOP.

STOP: Detiene las conversiones.

RDATA: El comando RDATA carga el registro de desplazamiento de salida con los datos más recientes en cada señal DRDY sin necesidad de emitir comandos de lectura de datos posteriores.

SDATA: Detiene el modo de lectura continua de los datos.

RDATA: El comando RDATA carga el registro de desplazamiento de salida con los datos más recientes cuando no está en modo de lectura continua de datos. Se debe emitir este comando después de que DRDY pase a bajo para leer el resultado de la conversión.

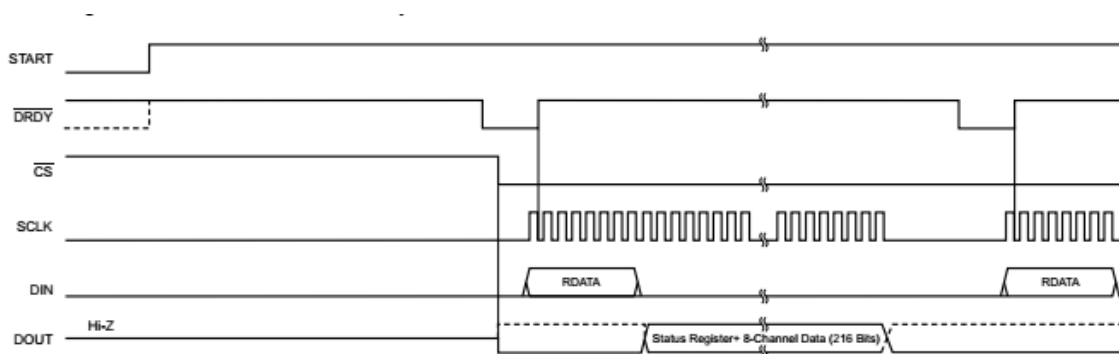


FIG. 4-4 LECTURA DE DATOS

El número de bits en la salida de datos depende del número de canales y el número de bits por canal. Para el ADS1299 de 4 canales, el número de salidas de datos es [(24 bits de estado + 24 bits × 4 canales) = 120 bits].

El formato de los 24 bits de estado es: (1100 + LOFF_STATP + LOFF_STATN + bits [4:7] del registro GPIO). El formato de datos para cada canal es complemento a dos y MSB primero. Cuando se apagan los canales mediante la configuración del registro del usuario, la salida del canal correspondiente se establece en '0'. Sin embargo, la secuencia de salida del canal sigue siendo la misma.

El dispositivo proporciona 24 bits de datos en formato complemento a dos binario. El tamaño de un código (LSB) se calcula mediante la siguiente ecuación.

$$1 \text{ LSB} = (2 \times VREF / \text{Ganancia}) / 2^{24}$$

RREG: Este comando lee datos del registro. El comando de lectura de registro consta de dos bytes seguidos de la salida de los datos del registro. El primer byte contiene el comando y la dirección del registro. El segundo byte de comando especifica el número de registros a leer, menos 1.

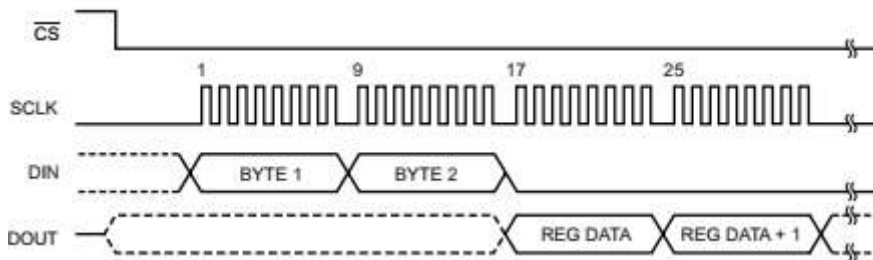


FIG. 4-5 LECTURA DE REGISTROS

WREG: Este comando escribe datos en el registro. El comando de escritura de registro consta de dos bytes seguidos de la entrada de datos del registro. El primer byte contiene el comando y la dirección del registro. El segundo byte de comando especifica el número de registros a escribir, menos 1.

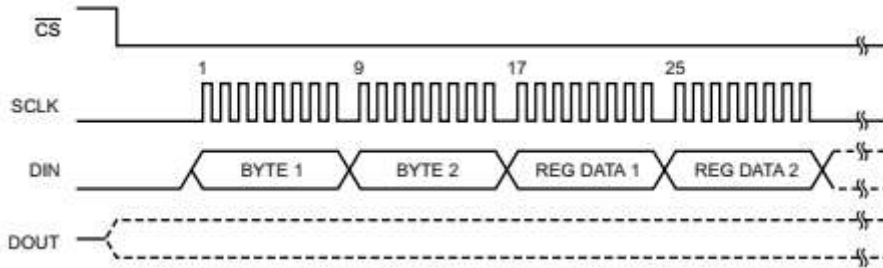


FIG. 4-6 ESCRITURA DE REGISTROS

ADDRESS	REGISTER	DEFAULT SETTING	REGISTER BITS							
			7	6	5	4	3	2	1	0
Read Only ID Registers										
00h	ID	xxh	REV_ID[2:0]			1	DEV_ID[1:0]		NU_CH[1:0]	
Global Settings Across Channels										
01h	CONFIG1	96h	1	DAISY_EN	CLK_EN	1	0	DR[2:0]		
02h	CONFIG2	C0h	1	1	0	INT_CAL	0	CAL_AMP0	CAL_FREQ[1:0]	
03h	CONFIG3	60h	PD_REFBUF	1	1	BIAS_MEAS	BIASREF_INT	PD_BIAS	BIAS_LOFF_SENS	BIAS_STAT
04h	LOFF	00h	COMP_TH[2:0]			0	ILEAD_OFF[1:0]		FLEAD_OFF[1:0]	
Channel-Specific Settings										
05h	CH1SET	61h	PD1	GAIN1[2:0]			SRB2	MUX1[2:0]		
06h	CH2SET	61h	PD2	GAIN2[2:0]			SRB2	MUX2[2:0]		
07h	CH3SET	61h	PD3	GAIN3[2:0]			SRB2	MUX3[2:0]		
08h	CH4SET	61h	PD4	GAIN4[2:0]			SRB2	MUX4[2:0]		
09h	CH5SET ⁽¹⁾	61h	PD5	GAIN5[2:0]			SRB2	MUX5[2:0]		
0Ah	CH6SET ⁽¹⁾	61h	PD6	GAIN6[2:0]			SRB2	MUX6[2:0]		
0Bh	CH7SET ⁽²⁾	61h	PD7	GAIN7[2:0]			SRB2	MUX7[2:0]		
0Ch	CH8SET ⁽²⁾	61h	PD8	GAIN8[2:0]			SRB2	MUX8[2:0]		
0Dh	BIAS_SENSP	00h	BIASP8 ⁽¹⁾	BIASP7 ⁽²⁾	BIASP6 ⁽¹⁾	BIASP5 ⁽¹⁾	BIASP4	BIASP3	BIASP2	BIASP1
0Eh	BIAS_SENSN	00h	BIASN8 ⁽¹⁾	BIASN7 ⁽²⁾	BIASN6 ⁽¹⁾	BIASN5 ⁽¹⁾	BIASN4	BIASN3	BIASN2	BIASN1
0Fh	LOFF_SENSP	00h	LOFFP8 ⁽¹⁾	LOFFP7 ⁽²⁾	LOFFP6 ⁽¹⁾	LOFFP5 ⁽¹⁾	LOFFP4	LOFFP3	LOFFP2	LOFFP1
10h	LOFF_SENSN	00h	LOFFM8 ⁽²⁾	LOFFM7 ⁽²⁾	LOFFM6 ⁽¹⁾	LOFFM5 ⁽¹⁾	LOFFM4	LOFFM3	LOFFM2	LOFFM1
11h	LOFF_FLIP	00h	LOFF_FLIP8 ⁽²⁾	LOFF_FLIP7 ⁽²⁾	LOFF_FLIP6 ⁽¹⁾	LOFF_FLIP5 ⁽¹⁾	LOFF_FLIP4	LOFF_FLIP3	LOFF_FLIP2	LOFF_FLIP1
Lead-Off Status Registers (Read-Only Registers)										
12h	LOFF_STATP	00h	IN8P_OFF	IN7P_OFF	IN6P_OFF	IN5P_OFF	IN4P_OFF	IN3P_OFF	IN2P_OFF	IN1P_OFF
13h	LOFF_STATN	00h	IN8M_OFF	IN7M_OFF	IN6M_OFF	IN5M_OFF	IN4M_OFF	IN3M_OFF	IN2M_OFF	IN1M_OFF
GPIO and OTHER Registers										
14h	GPIO	0Fh	GPIOD[4:1]				GPIOC[4:1]			
15h	MISC1	00h	0	0	SRB1	0	0	0	0	0
16h	MISC2	00h	0	0	0	0	0	0	0	0
17h	CONFIG4	00h	0	0	0	0	SINGLE_SHOT	0	PD_LOFF_COMP	0

- (1) Register or bit only available in the ADS1299-6 and ADS1299. Register bits set to 0h or 00h in the ADS1299-4.
(2) Register or bit only available in the ADS1299. Register bits set to 0h or 00h in the ADS1299-4 and ADS1299-6.

FIG. 4-7 ASIGNACIÓN DE LOS REGISTROS

4.4 Arquitectura

En esta sección se brinda un breve resumen del código y se pretende explicar el funcionamiento del programa.

La figura 4-8 refleja la estructura principal del software del dispositivo desarrollado en MicroPython. Para su implementación se decidió utilizar un paradigma secuencial que puede

ser interrumpido por eventos y se diseñó una estructura modular más fácil de mantener, corregir y modificar.

El programa sigue una estructura lógica que comienza importando las librerías necesarias.

Machine: Control del hardware del ESP32,

Network: creación de red.

socket: Creación del socket TCP-IP servidor.

const: Definición de constantes del sistema.

utime: Cálculo de tiempos y generación de retardos.

uselect: Permite monitorear varios sockets y esperar hasta que uno o más de ellos estén listos para realizar una operación de E/S (lectura o escritura).

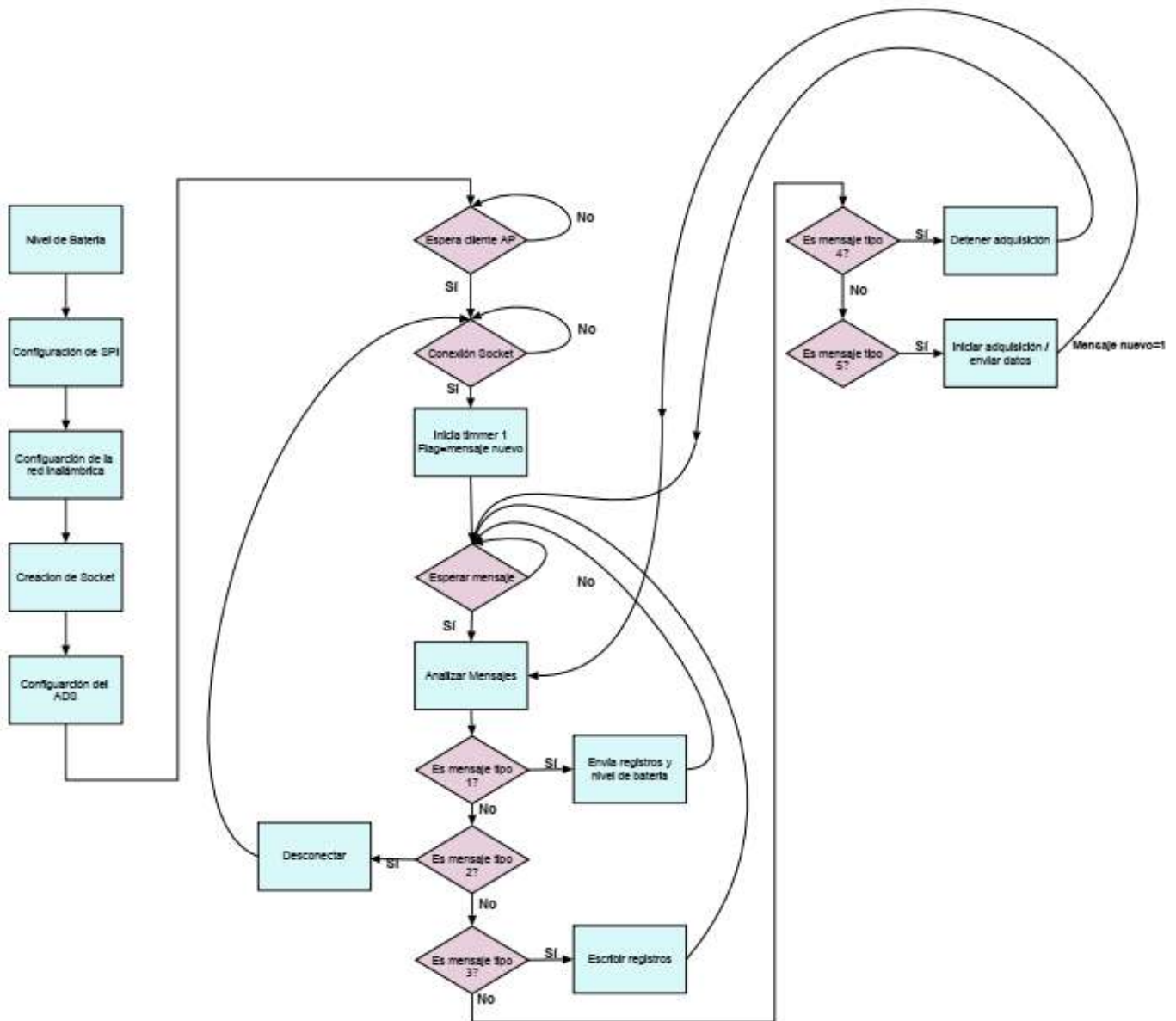


FIG. 4-8 DIAGRAMA DE FLUJO

Luego de importar las librerías necesarias, el código implementa las siguientes definiciones.

- Define las direcciones de los registros y los comandos del ADS1299 de la forma: ID=const(0X00) para las direcciones de los registros.

- START= const(0X08) para el comando de START.

- Inicializa los pines utilizados del ESP32 de la forma “powerdown = Pin (33, Pin.OUT, value = 1)”

- Define el Timer 0, encargado de verificar si existe un mensaje nuevo en el buffer de recepción cada 1 segundo y el Timer 1 encargado de leer el nivel de batería mediante el ADC.

- Crea una instancia llamada “**verificar**” de la clase **Select** que crea una lista de verificaciones de clientes conectados.

- Define las variables necesarias para la ejecución del programa. Además, incluye funciones para la lectura y escritura de registros del ADS1299 mediante la interfaz SPI, y funciones para la comunicación con el cliente mediante sockets.

Una vez importadas las librerías y definidas las variables necesarias, comienza la ejecución del programa.

En concordancia con la figura 4-8 se describen las funciones de los bloques.

-Nivel de batería: inicializa el PWM que se utiliza para variar la frecuencia del led indicador de la batería y del estado de la conexión.

- Se inicializa el led y se indica el estado desconexión de la red

- Se inicializa y activan las interrupciones del Timer 1 que actualiza el valor de la batería cada 60 segundos.

-Configuración del módulo SPI: posibilita la comunicación con el ADS1299 de acuerdo a lo detallado en los incisos 3.3.4 y 4.2.1.

-Configuración del módulo Network: Se crea la red wifi con el nombre “**DitEEG**”.

-Configuración del módulo SOCKET: Se genera un socket servidor capaz de aceptar y registrar clientes.

-Configuración del ADS

1. Se detiene la lectura continua del ADS y las conversiones.
2. Se escribe la configuración predeterminada de los registros del ADS.

Ejecución del loop principal:

- Se espera a que un cliente se conecte a la red wifi.
- Cuando un cliente se conecta, se indica mediante la frecuencia del LED, colocándolo en un estado visiblemente fijo.

- Se espera a que un cliente se conecte al socket.
- Cuando un cliente se conecta, se agrega a la lista de verificaciones para verificar si envía un nuevo mensaje.
- Se activa la interrupción del Timer 0 para verificar si hay un nuevo mensaje cada 1 segundo.
- Se verifica si hay un nuevo mensaje entrante.

Los mensajes recibidos por el dispositivo son de tamaño fijo de 18 Bytes.

1 Byte de verificación 0xFF, 1 Byte que indica el comando de 01 a 05, y 16 Bytes correspondientes a los registros de escritura del ADS1299. [CONFIG1, CONFIG2, CONFIG3, LOFF, CHSET1, CHSET2, CHSET3, CHSET4, BIAS_SENSP, BIAS_SENSN, LOFF_SENSP, LOFF_SENSN, LOFF_FLIP, GPIO, MISC1, CONFIG4]

- Si hay un mensaje entrante, se analiza su integridad y se llama a la función "**AnalizarMensaje**" que recibe dos parámetros: "**_comando**" y "**_dato**". Donde el **_dato** corresponde a los registros y el **_comando** al tipo de mensaje.
- Si el cliente se desconecta, el programa vuelve a esperar otra conexión al socket.

La función "**AnalizarMensaje**" hace una serie de comparaciones utilizando la estructura "if-elif-else" para determinar qué acción se debe tomar en función del valor del parámetro "**_comando**".

Si "**_comando**" es igual a 0x01, la función ejecutará una función llamada "**Conectado()**", que responde al cliente con un mensaje que contiene [0XC0, LOFF_STATP, LOFF_STATN, GPIO, CONFIG3, nivelbat], que corresponde a un byte con un comando de verificación, los registros de estado de conexión de los electrodos, el registro de estado del puerto GPIO del ADS, el registro CONFIG3 y el nivel de batería.

Si "**_comando**" es igual a 0x02, corresponde a un mensaje de desconexión, la función detendrá el "**Timer 0**", removerá el cliente de la lista de verificación y establecerá "salir" en True.

Si "**_comando**" es igual a 0x03, corresponde a un mensaje para guardar la configuración de los registros enviados, asignará el valor de "**_dato**" a la variable "**Valor_W**", creará un diccionario llamado "**Config_W**" y llamará a la función "**EscribirRegistros()**" que actualiza los valores de los registros del ADS1299. La función también enviará una respuesta al cliente para indicar la escritura correcta de los registros.

Si "**_comando**" es igual a 0x04, corresponde a un mensaje para detener las conversiones. Llamará a una función "**DetenerConversiones()**".

Si "**_comando**" es igual a 0x05, corresponde a un mensaje de solicitud de datos, llamará a una función llamada "**IniciarConversiones()**" y luego llamará a una función llamada "**EnviarDatos()**", el dispositivo quedara en este estado indefinidamente hasta que la variable "**mensaje_recibido**" se configure a verdadero por la recepción de un mensaje y retornara a "**AnalizarMensaje**".

Si "**_comando**" no es igual a ninguna de las condiciones anteriores, la función no realizará ninguna acción y volverá a esperar mensaje.

Capítulo 5

Diseño de software de la computadora

5.1 Introducción

Python es un lenguaje de programación popular que se utiliza para una amplia variedad de aplicaciones, incluido el desarrollo de software de escritorio con interfaces gráficas de usuario (GUI). Una de las herramientas más populares para crear interfaces gráficas en Python es Qt Designer[31], que es una herramienta de diseño visual que se integra con el marco de trabajo Qt.

Cuando se desarrolla un software en Python utilizando Qt Designer, el flujo de trabajo típico es el siguiente:

1. Diseño de la interfaz gráfica en Qt Designer: La interfaz de usuario se diseña utilizando el método de arrastrar y soltar, que permite al desarrollador colocar widgets (como botones, etiquetas, cuadros de texto, etc.) y personalizar sus propiedades, como tamaño, posición, colores, fuentes, etc.
2. Exportación de la interfaz gráfica a un archivo .ui: Una vez que se ha diseñado la interfaz gráfica, se exporta a un archivo .ui que contiene toda la información de diseño y configuración necesaria para el software.
3. Integración de la interfaz gráfica en el código Python: El archivo .ui se carga en el código Python y se utiliza para crear la interfaz gráfica en tiempo de ejecución. Esto implica crear objetos de widget y establecer sus propiedades y conexiones de señal y ranura.
4. Desarrollo de la lógica de la aplicación: Una vez que se ha creado la interfaz gráfica, se desarrolla la lógica de la aplicación en Python. Esto puede implicar el uso de bibliotecas

y marcos de trabajo adicionales para realizar tareas específicas, como el acceso a bases de datos, la manipulación de archivos, la comunicación de red, etc.

5. Pruebas y depuración: Una vez que se ha desarrollado el software, se realiza una prueba exhaustiva para detectar errores y asegurar que la aplicación funciona como se espera. Los errores se solucionan mediante la depuración del código y la modificación de la interfaz gráfica en caso necesario.

5.2 Desarrollo

La figura 5.2.a muestra el entorno gráfico de QT Designer.

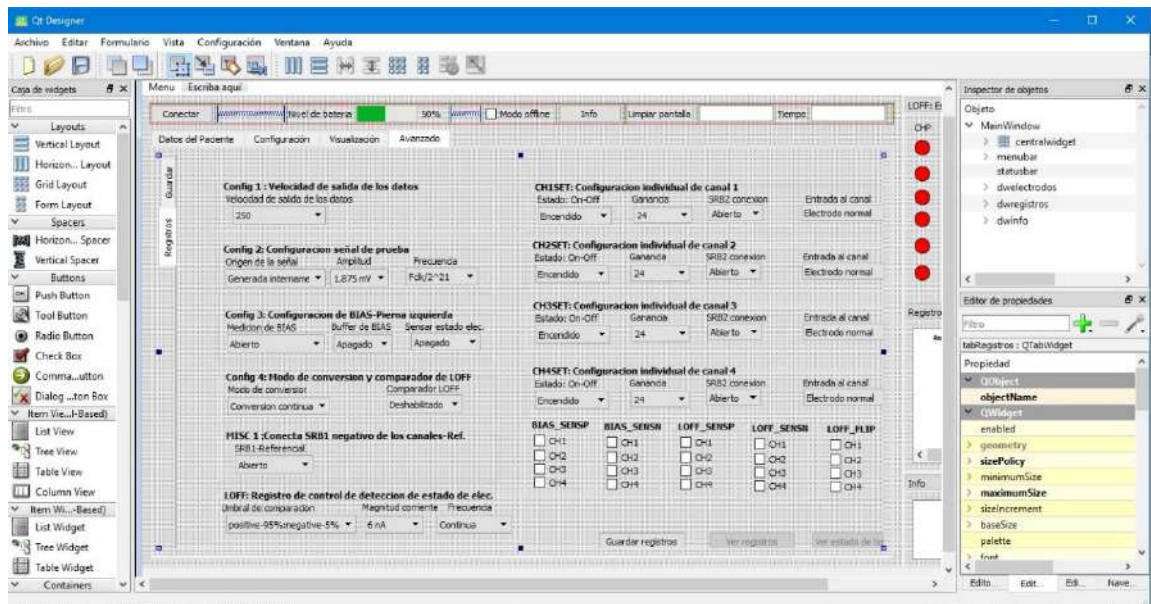


FIG. 5.2.A QT DESIGNER

La interfaz gráfica desarrollada se presenta en las figuras 5.2.b a 5.2.g.

En la figura 5.2.b, recuadrado en verde se puede ver un menú común a todas las pestañas. Este dispone de las siguientes funciones.

Menú: Habilita un menú con la opción “Abrir”, que despliega un explorador de archivos que permite buscar y cargar un archivo generado previamente de extensión .CSV que contiene los datos del estudio.

Para lograr esto se utilizó la propiedad **“triggered”**. En PyQt, esta propiedad se utiliza para representar la señal emitida cuando un elemento de menú es seleccionado. **“connect”**, permite conectar con una función definida por el usuario, en este caso **“cargar_datos()”**.

Esta función guarda en la variable **“archivo_offline”**, la dirección del archivo seleccionado que luego se utilizara para copiar los datos.

Conectar: Realiza la conexión con el dispositivo. Previo a la utilización de este botón el usuario debe conectarse la red wifi **“DitEEG”**, generada por el dispositivo. Este botón habilita la conexión en tiempo real.

Este comando realiza las siguientes acciones:

- Crea el socket y conecta con el dispositivo
- Envía un mensaje tipo 01, con la configuración predeterminada del ADS.
- Lee la respuesta del mensaje enviado y actualiza el valor del nivel de batería.
- Activa el **Timer0** de 60 segundos encargado de enviar al dispositivo un mensaje de este tipo y actualizar la hora del sistema.

Nivel de batería: Es un widget QProgressBar de PyQt, que refleja el nivel de batería del dispositivo cuando está conectado.

Modo Offline: Es un QCheckBox que habilita el modo offline del dispositivo que permite cargar estudios guardados previamente.

Info: Es botón despliega un QDockWidgwet, que muestra información relevante del archivo cargado mediante el menú **“Abrir”**, la frecuencia de muestreo, los canales muestreados y la duración del estudio.

Limpiar pantalla: Se utiliza en el modo offline para limpiar las visualizaciones previas.

Tiempo: Los cuadros de texto a la izquierda y derecha de esta etiqueta permiten seleccionar el tiempo inicial y final que se desea visualizar del estudio cargado. Esto es útil cuando se desea inspeccionar un tramo de interés del estudio efectuado.

Fecha: Muestra la fecha actual del sistema.

Pestaña “Datos del Paciente”: Permite guardar la fecha, los datos de contacto del paciente y observaciones pertinentes al estudio. Luego, mediante el botón “Guardar Datos”, se despliega un explorador de archivos que permite seleccionar donde guardar el archivo y el nombre que se le desea dar.

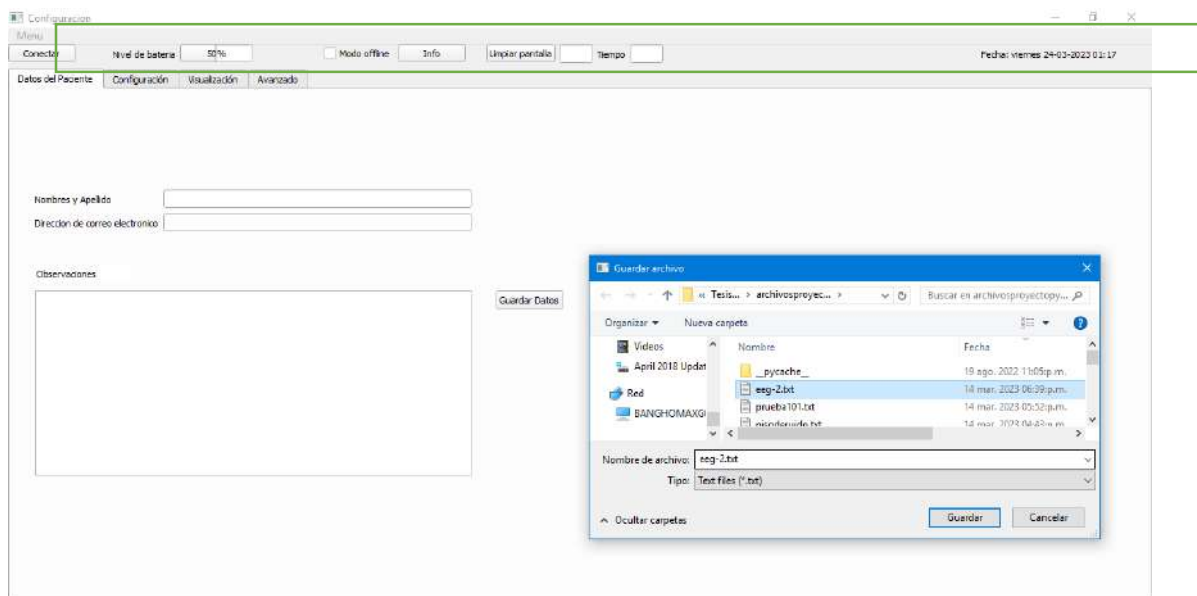


FIG. 5.2.B

Pestaña “Configuración”: Esta pestaña, mostrada en la figura 5.2.c, permite configurar las opciones del estudio, ver el estado de conexión de los electrodos y asociarlos a la disposición del sistema 10-20 y habilita la visualización del estudio. Si bien, esta pestaña está diseñada, la lógica correspondiente no se finalizó y se decidió por la configuración de la pestaña “Registros” de la figura 5.2.f. Queda como desarrollo a futuro terminar la lógica de esta pestaña.

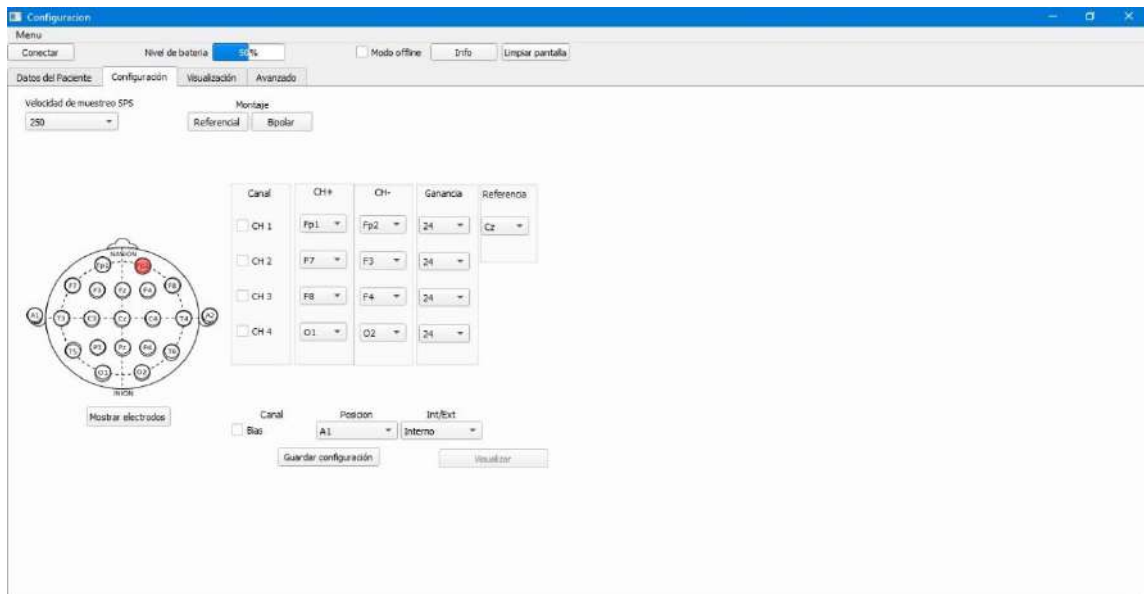


FIG. 5.2.c

Pestaña “Visualización”: La pestaña de la figura 5.2.d, permite el inicio, pausa y detención de la visualización de las señales seleccionadas.

Iniciar: Detiene el **Timer0**. Si el modo offline está activado, llama a la función “**modo_offline()**” y le pasa la variable **archivo_offline**. Esta función lee los archivos .TXT y .CSV relacionados al estudio previamente cargado por el menú “**Abrir**”, lee las variables de tiempo y frecuencias de corte de los filtros seleccionadas y grafica los datos.

Si el modo **offline** no está activo, llama a la función “**Datos()**”. Esta función, envía un mensaje tipo 05, activa el **Timer1** encargado de graficar los datos alojados en el registro **_copy** generado por la función “**procesar_datos()**” e inicializa un hilo (subproceso) de la clase **QThread** de **PyQt** que es el encargado de recibir los datos. Este hilo actualiza las variables de estado de los electrodos, el nivel de batería y convierte los datos binarios recibidos en 3 Bytes, correspondiente a cada muestra, en un dígito de 24 bits, luego emite una señal que llama a la función **procesar_datos** que será la encargada de actualizar el registro **_copy** de 10 mil muestras de los canales activos y aplicar los filtros definidos.

Pausa: Este botón tiene dos estados, **Pausa** y **Continuar**, que desactiva o activa el **Timer1** encargado de graficar los datos

Detener: Este botón llama a la función **Detener**, desactiva el **Timer1**, envía un mensaje **tipo04**, limpia la pantalla e inicializa el **Timer0**.

Filtro notch: Es un QCheckBox de PyQt que permite aplicar un filtro tipo FIR con diseño de ventana de Hamming, de las siguientes características:

- Filtro rechaza banda no causal y fase cero.
- Ventana Hamming con ondulación de banda de paso de 0.0194 y atenuación de banda de rechazo de 53 dB.
- Borde inferior de la banda de paso: 49.00 Hz.
- Ancho de banda de transición inferior: 0.50 Hz (frecuencia de corte de -6 dB: 48.75 Hz).
- Borde superior de la banda de paso: 51.00 Hz.
- Ancho de banda de transición superior: 0.50 Hz (frecuencia de corte de -6 dB: 51.25 Hz).
- Longitud del filtro: depende de la frecuencia de muestreo, se calcula mediante la ecuación $N = \frac{3.3}{\text{Anchodetransicion}} * SPS$ [33] Donde SPS es la frecuencia de salida de los datos muestreados o frecuencia de muestreo. Por ejemplo, para SPS=250 y ancho de transición de 0.50Hz da 1651 muestras (6.604 segundos)

Filtro pasa-banda: Este QCheckBox permite aplicar un filtro FIR de pasa banda definiendo la frecuencia de corte inferior y superior, y elegir el tipo de ventana para su diseño a través de la lista desplegable “Filtro” entre Hamming, Blackman, hanninig, Bartlett y Flat top. El ancho de la banda de transición es de 1HZ y el tamaño del filtro queda definido por la ecuación

$$N = \frac{3.3}{\text{Anchodetransicion}} * SPS$$

Para sus diseños se utilizó la librería Scipy de Python y para su implementación la librería MNE [32] [33].

El círculo verde marcado con 1 en la figura 5.2.d, es una lista de los canales activos, que indica el color correspondiente a cada canal, dando un clic con el mouse sobre él, permite ocultar o visualizar la señal.

Las señales se pueden visualizar en tiempo real en el dominio del tiempo mediante la pestaña “Temporal” o en el dominio de la frecuencia mediante su transformada de Fourier utilizando el algoritmo FFT en la pestaña “FFT”.

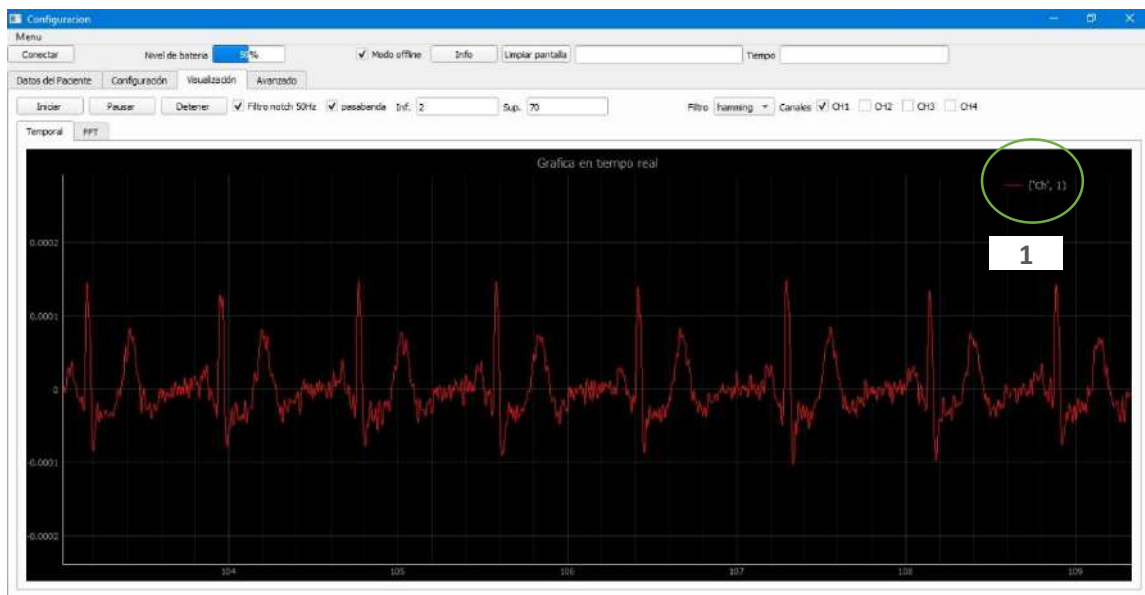


FIG. 5.2.D PESTAÑA DE VISUALIZACIÓN

Pestaña “Avanzado”: Esta pestaña incluye dos sub-pestañas, “guardar” y “registros”, figuras 5.2.e y 5.2.f respectivamente.

Sub-pestaña Guardar: Esta sub-pestaña permite seleccionar los datos que desean guardarse. Observaciones, configuración de los registros y datos del estudio.

Al presionar el botón “Guardar” se despliega un explorador de archivo para definir el nombre y lugar donde se guardará el archivo.

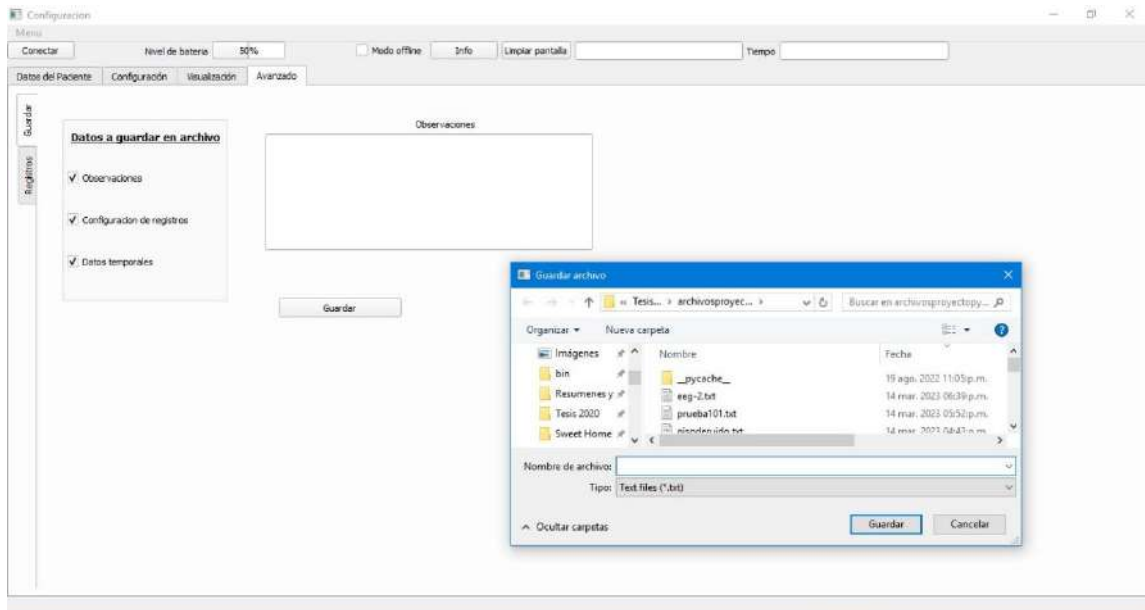


FIG. 5.2.E

Sub-pestaña Registros: Esta sub-pestaña de la figura 5.2.f, permite configurar todos los registros escribibles del ADS1299 mostrados en la tabla de la [figura 4-7](#) asignación de registros.

Ver registros: Este botón despliega una sub-ventana que muestra la última configuración de los registros en formato hexadecimal y binario.

Ver estado de los electrodos: Este botón despliega una sub-ventana que muestra el estado de conexión de los electrodos. Para poder visualizar esto, se debe habilitar el comparador de Lead Off de Config4, tildar los QCheckBox de los canales deseados en LOFF_SENSP y LOFF_SENSN y seleccionar los umbrales de comparación, la intensidad de la corriente y el tipo de señal en el registro LOFF. Además, es posible definir el sentido de la corriente mediante el registro LOFF_FLIP. Tildando los respectivos QCheckBox la corriente fluirá del canal negativo al positivo y de forma inversa si no se tildan. También existe la posibilidad de ver el estado del electrodo de BIAS, para lograr esto se debe apagar el buffer de Bias y colocar en encendido **“sensar estado del electrodo”** en el registro CONFIG3. El estado del electrodo se adquiere del bit 0 del CONFIG3 donde “0” es conectado y “1” desconectado, esto permanece transparente para el usuario y lo visualiza en la sub-ventana desplegada.

Esta medición no es posible de realizar durante el muestreo, ya que requiere apagar el buffer de bias. La circuitería involucrada y el modo de trabajo fue desarrollado en el inciso [3.2.7 Detección de conexión del electrodo de polarización](#).

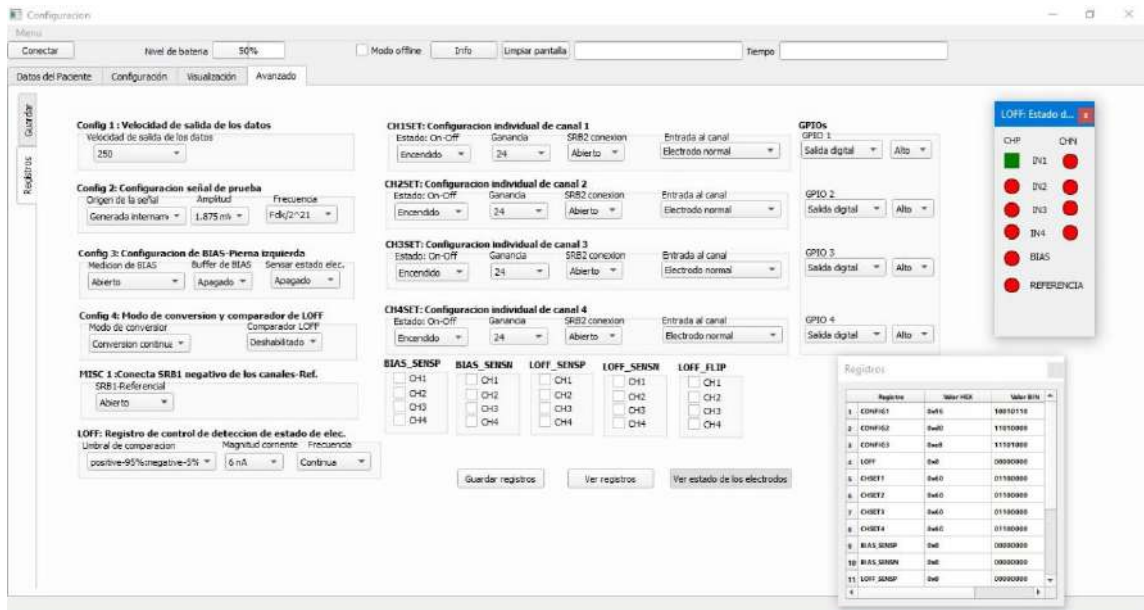


FIG. 5.2.F

Capítulo 6

Pruebas y análisis de resultados

En este capítulo se evalúa el funcionamiento del equipo telemétrico de señales EEG. Se verifican tanto el funcionamiento del software como del hardware. Se analizan los resultados obtenidos y se comparan con los valores declarados en los requerimientos funcionales.

6.1 Verificación general

La verificación general del dispositivo telemétrico de señales EEG es una prueba importante para garantizar su correcto funcionamiento y detectar posibles problemas en la placa. Para ello, se realizaron las siguientes pruebas:

6.1.1 Medición de continuidad y ausencia de corto circuitos en la placa.

1. Se utilizó un multímetro para medir la continuidad de los circuitos de la placa.
2. Se verificó la ausencia de cortocircuitos en la placa.

Resultados:

Los resultados de las pruebas realizadas indican que no hay problemas de continuidad ni cortocircuitos en la placa, lo que asegura su correcto funcionamiento.

6.1.2 Medición de las tensiones de fuentes:

Es importante verificar que las tensiones de fuentes del dispositivo se encuentran dentro de los rangos adecuados para asegurar su correcto funcionamiento [34]. Para ello, se realizaron las siguientes pruebas:

Procedimiento:

Medir con un multímetro las tensiones de alimentación analógica y digital con el dispositivo conectado únicamente a la batería y en condición de máximo consumo. Este procedimiento se realiza con batería con carga máxima y con carga mínima. En la tabla 6.1.2.a se muestran los resultados obtenidos.

Resultados:

Batería	AVDD	AVDD Esperado	DVDD	DVDD Esperado
Mínimo 3.4 V	4.94 V	5 V	3.14	3.3
Máximo 4.2 V	5.03 v	5V	3.31	3.3

Los resultados de las pruebas realizadas indican que las tensiones de fuentes del dispositivo se encuentran dentro de los rangos adecuados.

6.1.3 Verificación del sistema de carga de la batería:

La batería es una parte fundamental del dispositivo telemétrico de señales EEG, por lo que es necesario comprobar que su sistema de carga funciona correctamente. Para ello, se realizaron las siguientes pruebas:

Procedimiento:

1. Con la batería descargada (3,4 V) se coloca el interruptor del dispositivo en la posición de carga.
2. Se conecta una fuente de alimentación externa de 5V con capacidad mínima de 500 mAh al puerto USB del dispositivo.
3. Se verifica que la corriente de carga y el voltaje cumplan con lo especificado en el capítulo 3

4. Se mide durante el proceso de carga las temperaturas de la batería y del circuito gestor de carga utilizando un termómetro laser.

Resultados:

VBat [Volt]	Corriente [mA]	Temperatura [C]	Tiempo [min]
3.2	500	27	0
4.2	55	28,5	246
4.2	0	28	321

Los resultados de las pruebas realizadas indican que el sistema de carga de la batería funciona correctamente. La carga se realiza según lo especificado en el capítulo 3 y las temperaturas de los componentes no sobrepasan los valores recomendados [24] [36]

En conclusión, se puede afirmar que el sistema de carga de la batería cumple con los requisitos de funcionamiento esperados y se encuentra en óptimas condiciones para su uso en el dispositivo telemétrico de señales EEG.

6.1.4 Depuración del ADS1299:

El ADS1299 es el chip encargado de adquirir las señales EEG del paciente, por lo que es importante comprobar que funciona correctamente. Para ello, se realizaron las siguientes pruebas [35].

Procedimiento:

1. Encender el dispositivo.
2. Verificar las tensiones analógica y digital.
3. Medir los voltajes en los terminales VCAP.
4. Habilita el reloj maestro interno.
5. Enviar el comando START.
6. Verificar con un osciloscopio sobre el pin DRDY si el dispositivo está realizando conversiones. El periodo observado entre flancos descendentes debe ser igual a la velocidad de datos predeterminada del dispositivo.

Resultados:

Pin	Valor obtenido	Valor esperado
Vcap 1	1.21	1.2 V
Vcap 2	4.52	4.5V
Vcap 3	6.92	6.9 V
Vcap 4	2.24	2.25 V

Se verificó que las tensiones de verificación se encontraron dentro de los valores esperados y se visualizó correctamente la señal de DRDY con la frecuencia de salida de datos por defecto (250SPS).

6.1.5 Comprobación de la conexión USB y carga del software embebido en el esp32:

La comprobación de la conexión USB y carga del software embebido en el dispositivo es una etapa crítica en el desarrollo de esta plataforma. Esta prueba tiene como objetivo verificar que la conexión USB entre el ESP32 y el equipo está establecida correctamente, y que el software se carga y se ejecuta sin errores en el ESP32.

Para llevar a cabo esta prueba, se utiliza el IDE Thonny v3.2.7 y el firmware MicroPython v1.18. Thonny es un IDE fácil de usar que soporta diferentes plataformas y lenguajes de programación, mientras que MicroPython es un firmware ligero y eficiente que permite programar en Python para plataformas con recursos limitados como el ESP32.

Procedimiento:

1. Colocar el cable trenzado y colocar los jumpers J5, J6 y J7, para modo de programación
2. Desconecte la batería.
3. Colocar la llave de encendido en la posición de carga/prog.
4. Conectar el dispositivo a través del cable USB a un puerto USB disponible en el equipo.
5. Abrir el IDE Thonny en la PC.

6. Ir a la pestaña "Herramientas" en la barra de menú y seleccionar "Opciones...".
7. En la ventana de opciones, seleccionar la pestaña "Interprete".
8. Seleccionar "MicroPython (ESP32)" como intérprete.
9. Seleccionar el puerto serial correspondiente al ESP32 en la lista de puertos disponibles.
10. Hacer clic en "OK" para guardar la configuración.
11. En la pestaña "Archivo", seleccionar "Abrir" para buscar el archivo de código y abrirlo.
12. Hacer clic en "Run" o presionar F5 para ejecutar el código y comprobar que el IDE se conecta correctamente con el ESP32 y que el programa se ejecuta correctamente.
13. Si se produce algún error, verificar la configuración de Thonny, la conexión USB y que el cable sea el adecuado.

Resultados:

Una vez completado el procedimiento, el programa se ejecutó correctamente en el dispositivo. También se verificó que el software se ha cargado correctamente en el ESP32 y que se está ejecutando sin errores.

6.1.6 Prueba de funcionamiento de llave de encendido y sistema de puentes

Esta prueba es importante para comprobar que el sistema de fuentes y encendido se relacionan correctamente.

Procedimiento:

1. Verificar que el cable trenzado de programación no este colocado.
2. Verificar que la batería este correctamente conectada.
3. Colocar los puentes J5, J6 y J7 en la posición de encendido.
4. Colocar la llave de encendido en la posición de encendido.
5. Verificar que el led verde se enciende.
6. El led parpadea rápidamente para indicar que la red Wifi fue generada.

7. Verificar en la computadora que la red se generó correctamente.

Resultados:

Se pudo observar que el led se encendió y la red wifi fue generada exitosamente, verificando la correcta asociación de los puentes y la llave de encendido.

6.1.7 Autonomía

La autonomía de la batería es un factor crucial a considerar en el diseño de dispositivos telemétricos. Es necesario conocer la duración de la carga de la batería del dispositivo para garantizar su correcto funcionamiento durante el tiempo de uso previsto. Por lo tanto, en esta sección se describen los procedimientos utilizados para evaluar la autonomía de la batería del dispositivo telemétrico.

Procedimiento:

Para realizar la evaluación de la autonomía, se cargó completamente la batería del dispositivo telemétrico. Se configuró para adquirir señales EEG a una frecuencia de muestreo de 1 KSPS. Se utilizó una señal de prueba interna para simular la adquisición de señales durante un período de tiempo determinado.

Se registró la duración de la adquisición de señales EEG antes de que la batería se agotara. La duración de la batería se midió en minutos y se realizó en la condición de máximo consumo.

Resultados:

El tiempo de duración de la batería que se midió fue de 248 minutos.

Se observó que la duración de la batería disminuyó significativamente en modo de transmisión debido al consumo de energía necesario para transmitir los datos adquiridos a través de la red inalámbrica.

Estos resultados permiten conocer la duración de la batería del dispositivo en situaciones reales de uso, lo que es importante para la planificación del uso del dispositivo y la toma de decisiones sobre la gestión de energía del mismo.

6.2 Verificación del adquiredor

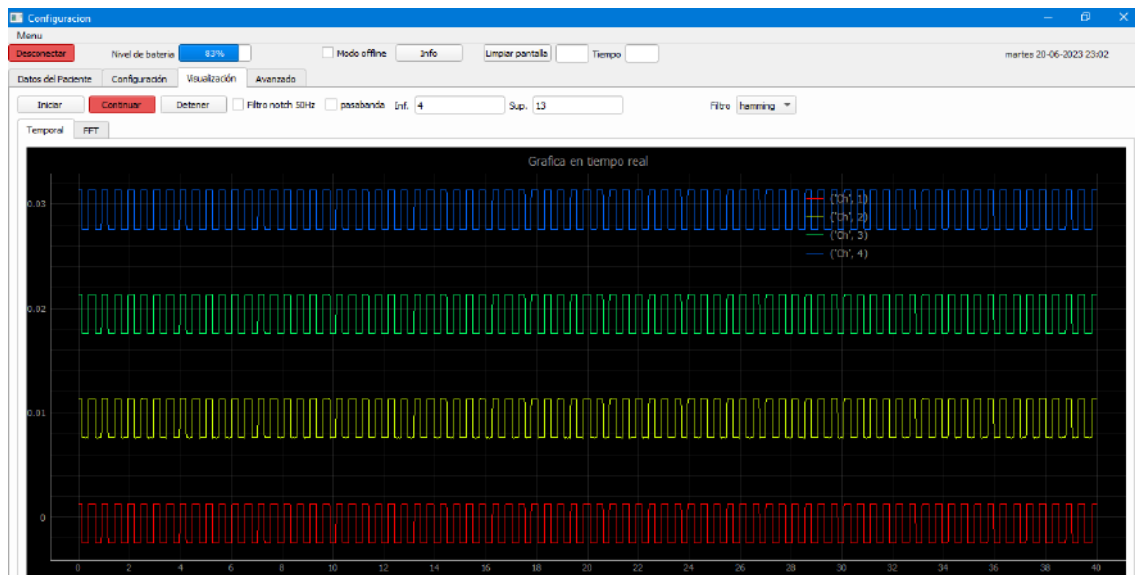
6.2.1 Medición de señal de prueba interna y verificación de los 4 canales.

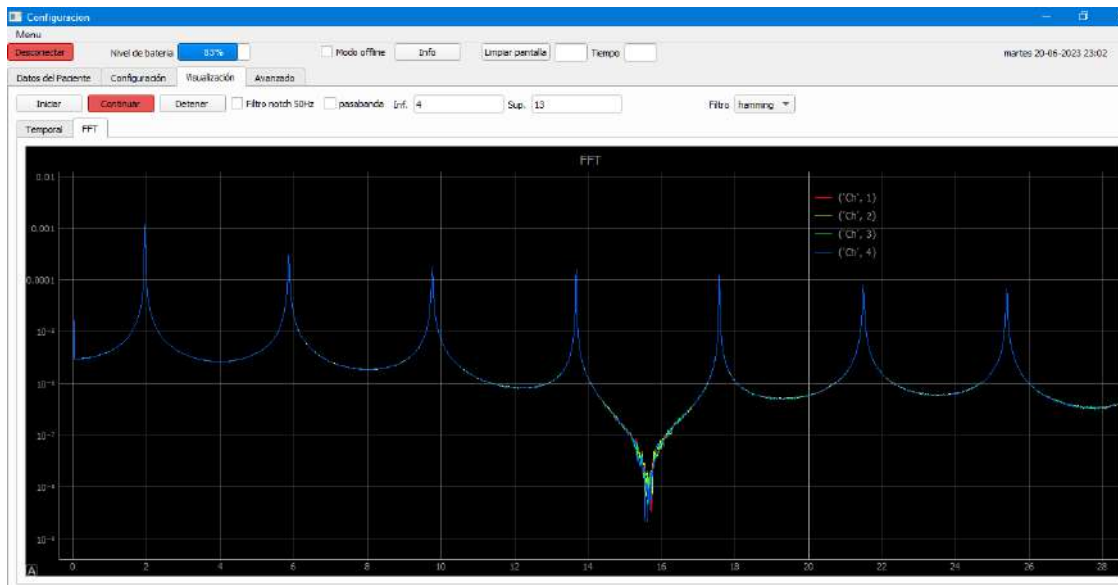
Objetivo Comprobar mediante el uso de una señal de prueba interna que los circuitos propios del ADS1299 funcionan correctamente y verificar la adquisición y trazado de los 4 canales simultáneamente.

Procedimiento:

Configurar en la sección correspondiente de la interfaz gráfica los registros necesarios para generar una señal de prueba interna de amplitud $1,875mV$ y $1,95Hz$ y medirla en los cuatro canales con la siguiente configuración frecuencia de muestro $250 SPS$ y ganancia unitaria.

Resultados: Se registró una onda cuadrada de frecuencia dentro de los márgenes propuestos. se verifico mediante la FFT de la señal adquirida.





6.2.2 Rechazo al modo común CMRR

Objetivo:

Medir el CMRR dentro de la banda de frecuencia de interés.

Procedimiento:

Realizar el siguiente procedimiento:

Utilizando un generador de señales sinodales.

Configuración: Conectar la salida + del generador de señales al canal 1 y al canal de referencia.

La salida - se conecta al canal de bias. El resto de los canales se deja al aire.

1. Conectar la salida del generador de señales en ambas entradas del canal 1.
2. Configurar para adquirir la señal en el canal 1, con frecuencia de muestreo de 250SPS y Ganancia 1 y amplificador de BIAS encendido.
3. Inyectar una senoide de 1Vpp a distintas frecuencias (5, 10, 15, 20, 25, 30 Hz).
4. Relevar la salida.
5. Calcular el CMRR para las distintas configuraciones y frecuencias.

Resultados:

Recordar que como se compensa la ganancia del ADS1299 por software, la ganancia diferencial es 1V/V.

Frecuencia (Hz)	FFT (V/Hz)	CMRR(dB)
5	3.98e-6	108
10	7.2e-6	103
15	1.35e-5	97
20	1.36e-5	97
25	2.25e-5	95
30	2.82e-5	91
50	4.5e-5	87

6.2.3 Resolución

En base al registro de ruido de entrada es posible calcular la resolución efectiva del dispositivo para una ganancia y velocidad de datos determinadas.

$$\text{Resolución efectiva(bits)} = \log_2\left(\frac{FSR}{V_{N,RMS}}\right) \text{ con } FSR = 2 * V_{ref}/Gan \quad [21]$$

$$V_{ref} = 4.5$$

$$\text{Ganancia} = 24$$

$$\text{Frecuencia de muestreo} = 1\text{KSPS}$$

$$V_{N,RMS} = 3.25\text{E-}7$$

$$\text{Resolución efectiva} = 20.13 \text{ bits}$$

6.2.4 Ruido equivalente a la entrada

Objetivo:

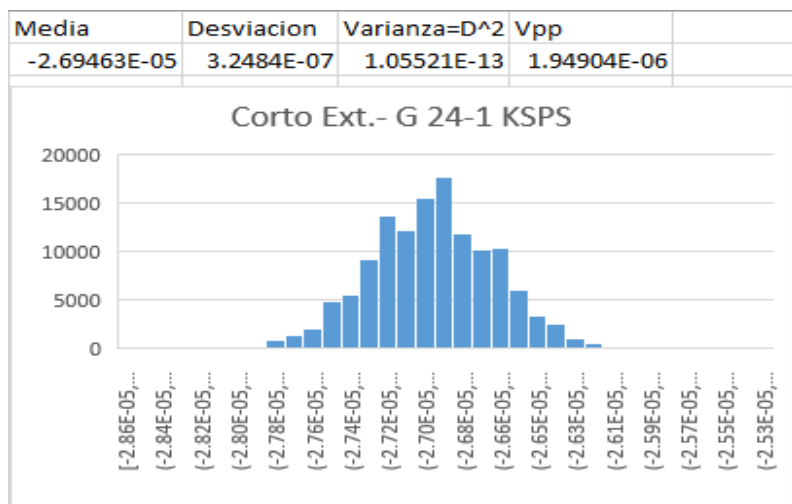
Medir el ruido equivalente en la entrada del dispositivo para verificar que se ajuste a los valores nominales ofrecidos por el fabricante en la hoja de datos del ADS1299.

La configuración elegida es una velocidad de datos de 1KSPS y ganancia 24 V/V.

Procedimiento:

- Generar un cortocircuito entre las entradas P y N del canal 1 y la salida de BIAS.
- Encender el dispositivo y conectarlo con el pc.
- Mediante la interfaz de usuario, seleccionar solo el canal 1 encendido, como entrada al canal seleccionar electrodo normal y ganancia 24. Configurar en velocidad de salida de los datos 1000 SPS.

Resultado: Se realiza un registro de 100 mil muestras, de las cuales se obtiene la gráfica 6.2.5.a. Se calcula el valor pico a pico como 6 desviaciones estándar, obteniendo $V_{pp} = 1.95 \mu V_{pp}$.



6.2.5.a Distribución de ruido

6.3 Pruebas de comunicación

6.3.1 Distancia entre dispositivos

Objetivo:

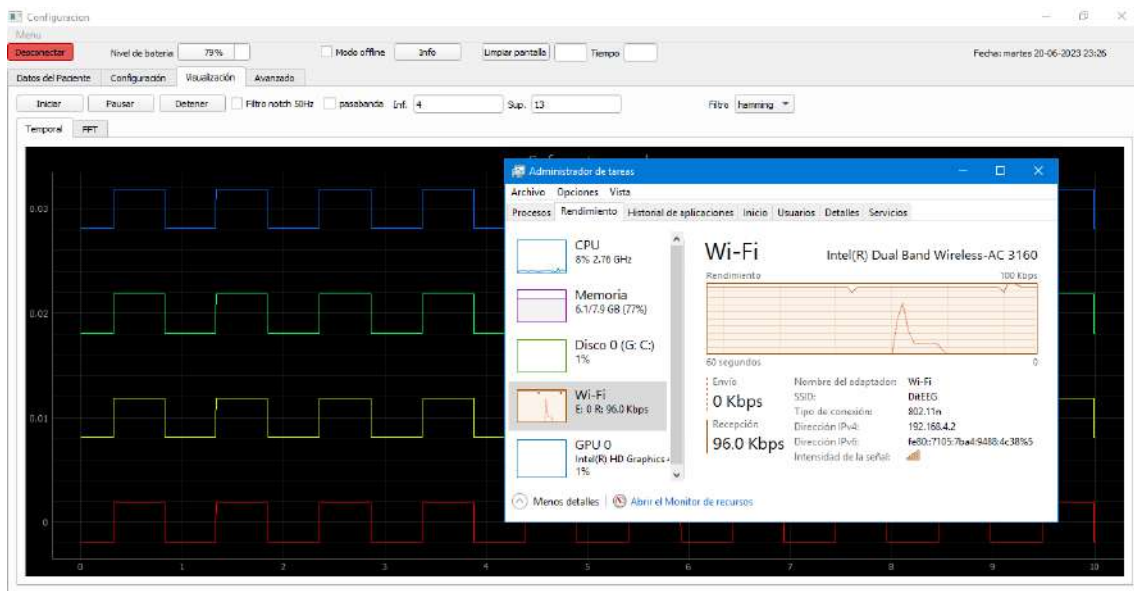
Verificar que el dispositivo se conecta y transmite datos en forma exitosa en un radio de 5 metros.

Procedimiento:

- Encender el dispositivo y conectarlo a la PC
- Iniciar el software y configurarlo para adquirir muestras de la señal de prueba interna desde los 4 canales a 1KSPS.
- Desplazarse con el dispositivo en un radio de 5 metros.

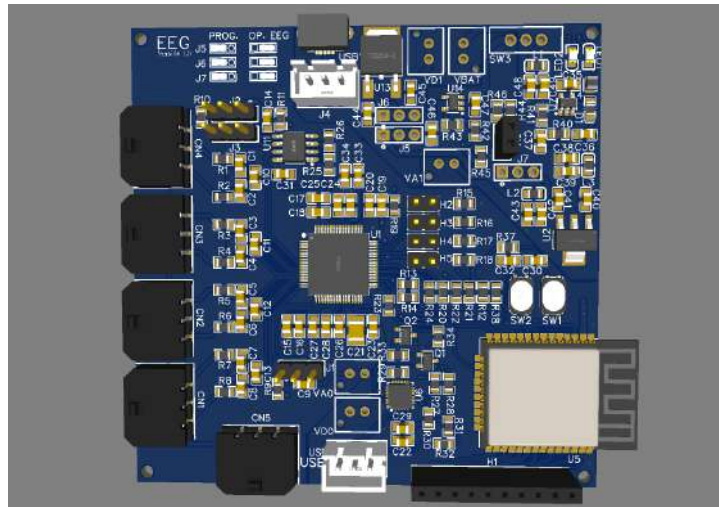
Resultado:

Se verifico que la señal se mantenía constante y la velocidad de la conexión permanecía estable, demostrando el correcto funcionamiento del dispositivo en el ambiente de prueba.



Capítulo 7

Conclusiones



En este proyecto se diseñó y construyó un dispositivo telemétrico de señales EEG para su uso en aplicaciones de investigación. A través del proceso de desarrollo, se llevaron a cabo diversas pruebas y evaluaciones para comprobar la funcionalidad del dispositivo, así como la calidad de las señales registradas. Los resultados obtenidos indican que el dispositivo es capaz de capturar y transmitir señales EEG de manera precisa y estable, lo que lo hace adecuado para su uso en aplicaciones de investigación.

Validamos la confiabilidad en la adquisición de las señales y la facilidad en el manejo de la interface de usuario del desarrollo propio con el sistema comercial adquisidor de señales biomédicas utilizado para educación en el Laboratorio de Bioingeniería. El desarrollo propio resulta ser amigable con el usuario a la vez que la adquiere señales con es de alta precisión. La Plataforma brinda una solución de bajo costo y accesible para la adquisición y análisis de señales, lo que puede resultar beneficioso para la monitorización remota en situaciones donde se requiere una solución portátil.

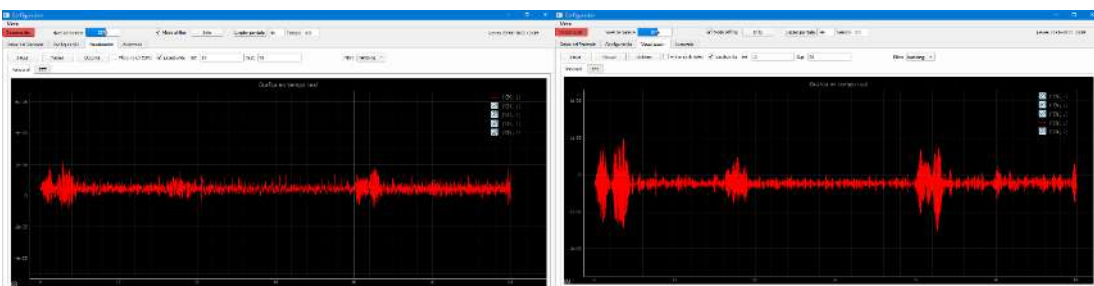
Además, se llevaron a cabo pruebas de autonomía, comprobación de la conexión USB y carga del software embebido en el ESP32, y verificación de funcionamiento de los 4 canales, entre otros. Estas pruebas demostraron que el dispositivo es capaz de funcionar correctamente en diversas situaciones y ambientes.

Aunque se logró un buen rendimiento en las pruebas realizadas, se identificaron algunas áreas de mejora para futuras versiones del dispositivo. Por ejemplo, se podría explorar la posibilidad

de incluir más canales de registro EEG para permitir un mayor número de mediciones simultáneas, así como mejorar la eficiencia energética para prolongar la duración de la batería.

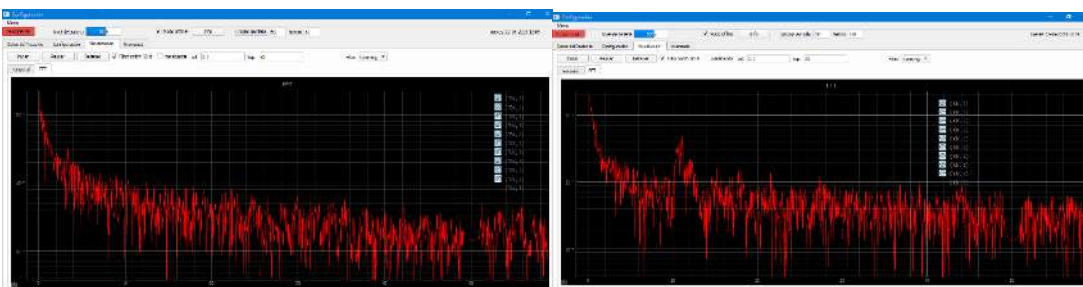
En general, el dispositivo telemétrico de señales EEG desarrollado en este proyecto ofrece una alternativa viable y de bajo costo para la captura y transmisión de señales EEG en aplicaciones de investigación. Los resultados obtenidos y las recomendaciones planteadas abren la puerta para futuras investigaciones y mejoras en el dispositivo.

Las figuras 7.1 a 7.4 son los resultados obtenidos con el dispositivo desarrollado, conectando el canal 1 en las posiciones O1 y Cz del sistema 10-20, donde se visualiza la activación de la onda Alfa al realizar la experiencia de mantener durante 20 segundos los ojos cerrados, 20 segundos de ojos abiertos y 20 segundos con los ojos cerrados.



7.1 – EEG – sin filtrar

7.2 – EEG – 8 a 13 Hz



7.3 – EEG – Espectro ojos abiertos

7.4 – EEG – Espectro ojos cerrados

Bibliografía

- [1] “El dispositivo para detectar somnolencia al conducir ya puso primera”, Edu.ar. [En línea]. Disponible en: <http://www.suplementos.unsj.edu.ar/octubre73/?p=1463>. [Consultado: 07-jun-2023].
- [2] “Electroencefalografía (EEG)”, MayoClinic.org, 19-jul-2022. [En línea]. Disponible en: <https://www.mayoclinic.org/es-es/tests-procedures/eeg/about/pac-20393875>. [Consultado: 07-jun-2023].
- [3] “Openbci”, Openbci.com. [En línea]. Disponible en: <https://openbci.com/>. [Consultado: 07-jun-2023].
- [4] “Data acquisition, loggers, amplifiers, transducers, electrodes”, BIOPAC Systems, Inc, 09-may-2015. [En línea]. Disponible en: <https://www.biopac.com/>. [Consultado: 07-jun-2023].
- [5] J. D. Enderle, J. D. Bronzino, y S. M. Blanchard, Introduction to Biomedical Engineering, 2a ed. Londres, Inglaterra: Academic Press Inc. (London), 2005.
- [6] J. E. Hall, Guyton y Hall. Tratado de Fisiología Medica + Studentconsult, 12a ed. Elsevier, 2014.
- [7] D. Joseph, “The BIOMEDICAL engineering handbook. Bronzino, Joseph D”, 2006.
- [8] “ADS1299 datasheet”, Wwww.ti.com. [En línea]. Disponible en: <https://www.ti.com/document-viewer/ADS1299/datasheet/abstract>. [Consultado: 08-jun-2023].
- [9] “Intan Technologies”, Intan Technologies. [En línea]. Disponible en: https://intantech.com/products_RHD2000.html. [Consultado: 08-jun-2023].
- [10] B. Baker, “How delta-sigma ADCs work, Part 1”, Wwww.ti.com. [En línea]. Disponible en: https://www.ti.com/lit/an/slyt423a/slyt423a.pdf?ts=1686203286271&ref_url=https%253A%252F%252Fwww.google.com%252F. [Consultado: 08-jun-2023].

- [11] B. Baker, "How delta-sigma ADCs work, Part 2", Wwww.ti.com. [En línea]. Disponible en: https://www.ti.com/lit/an/slyt438/slyt438.pdf?ts=1686203912742&ref_url=https%253A%252F%252Fwww.google.com%252F. [Consultado: 08-jun-2023].
- [12] "In-Amp Input RFI Protection", Analog.com. [En línea]. Disponible en: <https://www.analog.com/media/en/training-seminars/tutorials/MT-070.pdf>. [Consultado: 20-jun-2023].
- [13] G. Wifi, Bluetooth, y B. Le Module, "ESP32WROOM32E ESP32WROOM32UE Datasheet", Espressif.com. [En línea]. Disponible en: https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf. [Consultado: 08-jun-2023].
- [14] Wikipedia contributors, "Serial Peripheral Interface", Wikipedia, The Free Encyclopedia. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Serial_Peripheral_Interface&oldid=143065352.
- [15] "MicroPython - Python for microcontrollers", Micropython.org. [En línea]. Disponible en: <http://micropython.org/>. [Consultado: 08-jun-2023].
- [16] "Boot Mode Selection - ESP32 - — esptool.py latest documentation", Espressif.com. [En línea]. Disponible en: <https://docs.espressif.com/projects/esptool/en/latest/esp32/advanced-topics/boot-mode-selection.html>. [Consultado: 13-jun-2023].
- [17] S.-C. U. to U. Transfer y V. C. P. Drivers, "SINGLE-CHIP USB-TO-UART BRIDGE", Silabs.com. [En línea]. Disponible en: <https://www.silabs.com/documents/public/datasheets/CP2102-9.pdf>. [Consultado: 20-jun-2023].
- [18] "Primeros pasos con MicroPython en el ESP32 — Documentación más reciente de MicroPython", Micropython.org. [En línea]. Disponible en: <https://docs.micropython.org/en/latest/esp32/tutorial/intro.html>. [Consultado: 13-jun-2023].
- [19] A. Carvajales, "LDOs Thermal Performance in Small SMD Packages", Wwww.ti.com. [En línea]. Disponible en: https://www.ti.com/lit/an/snva753/snva753.pdf?ts=1686260014434&ref_url=https%3A%2F%2Fwww.google.com%2F. [Consultado: 08-jun-2023].

- [20] “LM2704 micropower step-up DC/DC converter with 550mA peak current limit”, Wwww.ti.com. [En línea]. Disponible en: https://www.ti.com/lit/ds/symlink/lm2704.pdf?ts=1687268918664&ref_url=https%3A%2F%2Fwww.ti.com%2Fproduct%2FLM2704. [Consultado: 20-jun-2023].
- [21] “Fundamentals of precision ADC noise analysis”, Wwww.ti.com, 2020. [En línea]. Disponible en: <https://www.ti.com/lit/eb/slyy192/slyy192.pdf?ts=1687090198285>. [Consultado: 18-jun-2023].
- [22] Microchip.com. [En línea]. Disponible en: <https://ww1.microchip.com/downloads/en/DeviceDoc/MCP1755-MCP1755S-Data-Sheet-DS20005160B.pdf>. [Consultado: 20-jun-2023].
- [23] Wwww.st.com. [En línea]. Disponible en: <https://www.st.com/resource/en/datasheet/ld39150.pdf>. [Consultado: 20-jun-2023].
- [24] Microchip.com. [En línea]. Disponible en: <https://ww1.microchip.com/downloads/en/DeviceDoc/MCP73831-Family-Data-Sheet-DS20001984H.pdf>. [Consultado: 08-jun-2023].
- [25] “AN-1520 A Guide to Board Layout for Best Thermal Resistance for Exposed Packages”, 2006.
- [26] “AN-2020 Thermal Design By Insight, Not Hindsight”, Wwww.ti.com, 2010. [En línea]. Disponible en: <https://www.ti.com/lit/an/snva419c/snva419c.pdf?ts=1686506792715>. [Consultado: 12-jun-2023].
- [27] Developed by the IPC-2221 Task Group of the Rigid Printed Board Committee of IPC, “Generic standard on printed board design”, Ipc.org. [En línea]. Disponible en: <https://www.ipc.org/TOC/IPC-2221A.pdf>. [Consultado: 08-jun-2023].
- [28] “EasyEDA – Simulador de circuitos y diseño de circuitos impresos online”, Easyeda.com. [En línea]. Disponible en: <https://easyeda.com/es>. [Consultado: 19-jun-2023].
- [29] “Thonny”, Thonny.org. [En línea]. Disponible en: <https://thonny.org/>. [Consultado: 08-jun-2023].

- [30] “MicroPython downloads”, Micropython.org. [En línea]. Disponible en: <https://micropython.org/download/>. [Consultado: 13-jun-2023].
- [31] “Qt Designer Manual”, Doc.qt.io. [En línea]. Disponible en: <https://doc.qt.io/qt-5/qt designer-manual.html>. [Consultado: 20-jun-2023].
- [32] “scipy.signal.firwin — SciPy v1.10.1 Manual”, Scipy.org. [En línea]. Disponible en: <https://docs.scipy.org/doc/scipy/reference/generated/scipy.signal.firwin.html#scipy-signal-firwin>. [Consultado: 19-jun-2023].
- [33] “Background information on filtering — MNE 1.4.2 documentation”, Mne.tools. [En línea]. Disponible en: https://mne.tools/stable/auto_tutorials/preprocessing/25_background_filtering.html#disc-filtering. [Consultado: 19-jun-2023].
- [34] A. Kay y T. Green, “Analog Engineer’s Pocket Reference”, Www.ti.com. [En línea]. Disponible en: <https://www.ti.com/seclit/eb/slyw038c/slyw038c.pdf>. [Consultado: 12-jun-2023].
- [35] “[FAQ] ADS129x: How do I verify that my ADS129x device is still functional?”, E2e.ti.com. [En línea]. Disponible en: <https://e2e.ti.com/support/data-converters-group/data-converters/f/data-converters-forum/775288/faq-ads129x-how-do-i-verify-that-my-ads129x-device-is-still-functional/2868597#2868597>. [Consultado: 14-jun-2023].
- [36] Wikipedia contributors, “Batería de polímero de litio”, Wikipedia, The Free Encyclopedia. [En línea]. Disponible en: https://es.wikipedia.org/w/index.php?title=Bater%C3%ADa_de_pol%C3%ADmero_de_litio&ol did=150915253.

Apéndice

A.1 Red de multiplexado

El ADS1299 posee una red de multiplexado por cada canal de entrada *figura A. 1. a*. La red ofrece muchas opciones configurables de conmutación de señales. Esta flexibilidad permite realizar diagnósticos, calibraciones y configuraciones del dispositivo y sus subsistemas. A continuación, se enumeran y se detallan las opciones de multiplexado utilizados en el proyecto:

- Normal: en esta posición el multiplexor, a través de la configuración de los registros adecuados en la *GUI*, coloca sus llaves internas de forma tal de dirigir las señales provenientes de los electrodos hacia los amplificadores internos del *ADS1299*. Esta configuración es la utilizada para realizar un estudio de *EEG*.
- Medición de ruido intrínseco del dispositivo: se realiza para obtener una medida del ruido propio del dispositivo. Para lograr esto, a través de la configuración de los registros adecuados en la interfaz gráfica, el multiplexor coloca las entradas *INp* e *INn*, entradas de un mismo canal, a una misma referencia. El valor obtenido es el ruido.
- Señales de testeo: se trata de señales generadas internamente que sirven para verificar la cadena de señal interna. En este caso, a través de la escritura mediante la interfaz gráfica de los registros adecuados, el multiplexor coloca sus llaves internas de forma tal de enviar hacia el amplificador una señal generada internamente en el *ADS1299*.
- Señales de excitación para la detección de conexión del electrodo: esta configuración se utiliza para enviar al electrodo y al paciente una corriente generada internamente. Con la escritura del registro adecuado, a través de la interfaz gráfica, se selecciona la intensidad de corriente y la frecuencia de las señales de excitación, en este proyecto se utiliza solo *DC*.
- Selección de la fuente de polarización: esta configuración permite seleccionar entre una fuente interna o externa para polarizar al paciente. Se debe, además de configurar el registro adecuado a través de la *GUI*, se debe retirar el puente colocado entre los terminales 2 y 3 de *J2* (bias) en los terminales 2 y 3 de *J3* (bias externo).

- Medición de la señal de polarización: se debe configurar el registro adecuado a través de la interfaz gráfica. Esta configuración rutea la señal obtenida en el pin *BIASIN* hacia el canal *P* del canal que se seleccione. Esta señal es comparada luego con la referencia del bias, la cual se rutea hacia el lado *N* de dicho canal. Para lograr esta medición se debe, retirar el puente de los terminales 2 y 3 de *J2* y colocarlo entre los terminales 1 y 2 de *J2*.
- Ruteo de la señal de bias: esta configuración permite derivar la señal de polarización por otro electrodo. Si bien dicha señal tiene su electrodo propio, esta configuración permite derivarla por otro electrodo en caso de falla del electrodo de bias. Para esta configuración además de modificar, a través de la *GUI* el registro adecuado, se debe colocar el puente ubicado en los terminales 2 y 3 de *J2* en los terminales 1 y 2 de *J2*.

Ruteo de *SRB1* para el modo referencial: esta configuración se realiza modificando el registro adecuado a través de la interfaz gráfica. Con esto se logra colocar las entradas *INn* de todos los canales con un mismo electrodo de referencia conectado en *SRB1*. Se debe colocar el puente entre los terminales 1 y 2 de *J1*

A.2 Filtros de entrada

Los amplificadores diferenciales poseen un alto rechazo de modo común *CMRR*, sin embargo, este disminuye a medida que la frecuencia de las señales de entrada aumenta *figura A. 6. a*. Esto da como resultado que frecuencias altas sean rectificadas en el amplificador y aparezca un error de compensación de *CC*. Por tal motivo se utilizan filtros para filtrar las señales comunes. En la *figura A. 6. b* se ve el esquema del filtro utilizado en el proyecto.

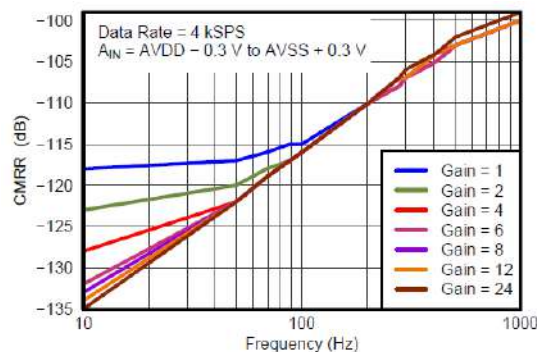


figura A. 6. a. CMRR vs frec. Fuente hoja de datos ADS1299 "Texas Instruments"

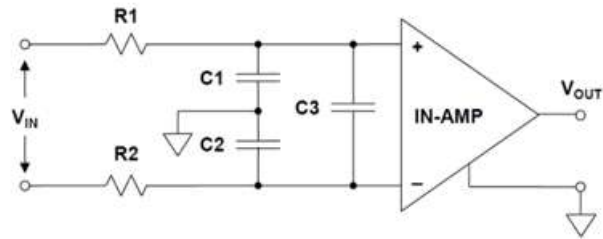


figura A. 6. b. Etapa de filtrado Analógico

En este esquema, las ramas $R1$, $C1$ y $R2$, $C2$ conforman los filtros de modo común, si bien estos pueden utilizarse también como filtros anti-solapamiento, para filtrar la señal diferencial útil, pequeños desajustes entre las constantes de tiempo $\frac{C1}{R1}$ y $\frac{C2}{R2}$ generan un desequilibrio entre las dos ramas que disminuye el rechazo al modo común. Por tal motivo se conecta entre las entradas del amplificador el capacitor $C3$. Este capacitor reduce efectivamente cualquier error de rechazo de modo común de CA debido a la no coincidencia de los componentes.

Para la selección de la topología del filtro diferencial y su frecuencia de corte se tuvieron en cuenta los siguientes factores: máxima frecuencia de la señal a adquirir, forma de la banda de transición del filtro y la frecuencia de muestreo del conversor analógico digital. En vista de que la frecuencia de muestro $f_s = 1.024 \text{ MHz}$ es varios ordenes superior a la máxima frecuencia de la señal adquirir $f_{max} = 50 \text{ Hz}$, y que esta característica disminuye los requisitos sobre la banda de transición *figura A. 6. c* se optó por seleccionar un filtro pasivo de primer orden. Esta selección ahorra componentes, ocupa un espacio reducido en el circuito impreso, no incorpora el ruido de elementos activos, abarata los costos y simplifica el diseño.

Para el filtrado de modo común se decidió por utilizar también un filtro pasivo de primer orden por cada canal y con una frecuencia de corte de por lo menos diez veces la frecuencia de corte del filtro diferencial.

Para mejorar el rechazo al modo común se utilizaron resistencias con una tolerancia del 0,5% y bajo coeficiente de temperatura, además se utilizaron valores de resistencia de modo tal que el ruido generado por estas sea menor al del amplificador. Se utilizaron capacitores clase 1 con una tolerancia del 5%, baja ESR y dielectrico COG ($NP0$). Los circuitos impresos se

diseñaron simétricamente y lo más próximos a los amplificadores, se generaron planos de masa en la capa superior y en la capa inferior y las curvas de las pistas no superaron los 45°. En la *figura A. 6. d* y la *figura A. 6. e* se puede ver el detalle del diseño del circuito impreso.

A continuación, se detallan los cálculos realizados:

$$f_c \text{ Diferencial} = \frac{1}{2\pi*(R_1+R_2)*\left(\frac{C_1*C_2}{C_1+C_2}+C_3\right)}$$

Si se cumple que: $\frac{C_1*C_2}{C_1+C_2} \ll C_3$ entonces se puede considerar que $f_c = \frac{1}{2\pi*(R_1+R_2)*C_3}$.

Para nuestro caso se tiene que:

$$R_1 = R_2 = 2.200 \pm 11 \Omega, C_3 = 47.00 \pm 2.35\eta F, f_c = 769.6 \pm 44.6Hz$$

A.3 Esquemático

En este apéndice se presenta el diagrama esquemático final del prototipo desarrollado en el software EasyEDA v.6.4.31.

