



Universidad de  
Oviedo



# ESCUELA POLITÉCNICA DE INGENIERÍA DE GIJÓN.

## GRADO EN INGENIERÍA EN TECNOLOGÍAS Y SERVICIOS DE TELECOMUNICACIÓN

ÁREA DE SISTEMAS ELECTRÓNICOS

MINI-OSCILOSCOPIO WEB

D. HUERGA FERNÁNDEZ, Lidia  
TUTOR: D. REVUELTA SANZ, Pablo

FECHA: Octubre, 2021

Lidia Huerga Fernández

# ÍNDICE

<b>1. RESUMEN.....</b>	<b>3</b>
<b>2. INTRODUCCIÓN.....</b>	<b>4</b>
2.1. - SEÑALES ELÉCTRICAS.....	4
2.2. - OSCILOSCOPIOS.....	11
2.3. - CONVERSORES A/D.....	24
2.4. - CONVERSORES DA.....	40
2.5. - MICROCONTROLADORES.....	48
2.6. - PROTOCOLOS DE COMUNICACIÓN.....	56
2.7. - LENGUAJES DE PROGRAMACIÓN.....	59
2.8. - COMUNICACIONES.....	66
2.9. - COMUNICACIONES INALÁMBRICAS.....	71
2.10. - PCB.....	73
<b>3. OBJETIVOS.....</b>	<b>78</b>
<b>4. PLANIFICACIÓN.....</b>	<b>79</b>
<b>5. DISEÑO.....</b>	<b>81</b>
5.1. - HARDWARE.....	81
5.2. - SOFTWARE.....	109
<b>6. PRUEBAS Y RESULTADOS.....</b>	<b>120</b>
6.1. - MEDIDA DE VOLTAJE.....	120
6.2. - MEDIDA DE CORRIENTE.....	126
6.3. - GENERACIÓN DE SEÑALES.....	128
6.4. - AUTONOMÍA.....	135
<b>7. PRESUPUESTO.....</b>	<b>139</b>
<b>8. CONCLUSIONES.....</b>	<b>141</b>
<b>9. TRABAJOS FUTUROS.....</b>	<b>142</b>
<b>10. BIBLIOGRAFÍA.....</b>	<b>144</b>

# 1. Resumen

Esta memoria abarca toda la información necesaria para el desarrollo del Trabajo de Fin de Grado para el Grado en Ingeniería en Tecnologías y Servicios de Telecomunicación de la Escuela Politécnica de Ingeniería de Gijón, titulado “Mini-osciloscopio web”.

Para este proyecto se ha diseñado un osciloscopio manejable desde cualquier dispositivo como puede ser un teléfono móvil o un ordenador. El mini-osciloscopio dispone de una interfaz de usuario intuitiva y sencilla para poder visualizar distintas entradas de señal, así como la capacidad de generarlas.

En la página web que contiene nuestro microcontrolador podemos visualizar las señales y realizar distintas funciones como zoom, trigger, cambio de canal de lectura, etc.

Los componentes principales para el desarrollo del osciloscopio son, por un lado, un microcontrolador donde grabaremos las instrucciones necesarias como de lectura de datos, lanzamiento de un servidor, etc.

Por otro lado, tenemos las distintas entradas de señal, para las cuales necesitaremos un convertidor analógico-digital para transformar los datos en información legible para el microcontrolador. Para nuestro generador de señales necesitaremos el caso contrario, un convertidor digital-analógico programable desde nuestro microcontrolador que sepa transformar datos digitales en una señal analógica de nuestro interés.

El dispositivo creado tiene un coste inferior a 25 euros en componentes a precio de mercado.

## 2. Introducción.

### 2.1.- SEÑALES ELÉCTRICAS.

El objeto de medida de los osciloscopios son las señales eléctricas. Una señal son ondas electromagnéticas propagadas a través de un medio de transmisión.

Uno de los aspectos fundamentales del nivel físico es transmitir información en forma de señales electromagnéticas a través de un medio. Esta información puede ser voz, imagen, datos numéricos, caracteres o código, cualquier mensaje que sea legible y tenga significado para el usuario destino [1].

Para la electrónica: "la señal se define como una cantidad física que varía con el tiempo, el espacio o cualquier variable o variables independientes". Así, desde el punto de vista matemático una función es una señal [1]. Estas ondas se pueden ver en casi todos los artefactos eléctricos que nos rodean, entendiéndose también como distintos niveles de tensión, resistencia, capacidad, inductividad e impedancia que podemos medir con distintos instrumentos.

Los distintos dispositivos con los que podemos medir estas señales son el voltímetro, amperímetro, el óhmetro o megóhmetro, el multímetro y, el de nuestro objeto de estudio, el osciloscopio.

#### 2.1.1.- Parámetros eléctricos.

A continuación explicaremos los distintos parámetros con los que podemos recopilar información de una señal.

- **Periodo.** El periodo de una onda se define como el tiempo en el que tarda una onda en repetir un ciclo. Se mide en segundos y se representa con la letra T.
- **Frecuencia.** La frecuencia es la medida del número de veces que se repite un proceso periódico por unidad de tiempo [2], y se mide en Hertzios. Es decir, las veces que se repite la onda por cada segundo.

$$f(\text{Hz}) = \frac{1}{T} \quad (2.1)$$

Siendo T el periodo.

- **Fase.** La fase de una onda se define como el desplazamiento horizontal o la fracción de periodo transcurrido desde el inicio de la señal. Se mide en radianes o grados [3].
- **Valor instantáneo.** Se trata del valor de la amplitud de la onda en un instante concreto de tiempo.
- **Valor máximo.** El valor máximo o valor de pico es el mayor valor que puede alcanzar una señal.
- **Valor medio.** Representa el promedio de la señal a través del tiempo. Se define como la relación entre la suma de los valores instantáneos de la señal y el período de la onda [4].

$$U_m = \int_0^{-T} U(t) \cdot dt \quad (2.2)$$

- **Valor cuadrático medio o eficaz.** A este parámetro también se le conoce como valor RMS y se define como el valor que produce el mismo efecto que la señal continua equivalente.

$$U_{rms} = \sqrt{\int_0^T U^2(t) \cdot dt} \quad (2.3)$$

- **Valor de pico a pico.** Este parámetro representa la diferencia entre el valor mínimo y máximo de una señal, y su valor es el doble del valor de pico.

Las medidas se pueden realizar tanto en escala lineal como logarítmica. La primera forma de representar las medidas suele utilizarse para medidas en las que los componentes son de aproximadamente el mismo valor, puesto que a valores muy pequeños se complica asignar una magnitud. Sin embargo, si queremos tener una mejor visión de la tendencia de nuestra señal es más recomendable utilizar una escala logarítmica [5].

Con la escala lineal los valores se representan con unidades naturales, mientras que con la logarítmica se utiliza el decibelio. El decibel (dB) es una unidad logarítmica que se utiliza para medir el nivel de sonido, muy utilizado en electrónica, señales y comunicaciones. También suele describir relaciones como de potencia, presión sonora, voltaje, etc [6].

### 2.1.2.- Clasificación de las señales.

Dentro de las señales podemos diferenciar distintos tipos en función de ciertas características.

I. En función de la variable independiente

- **Señales continuas.** Las señales continuas son aquellas cuya variable es continua y puede variar tomando cualquier valor real.

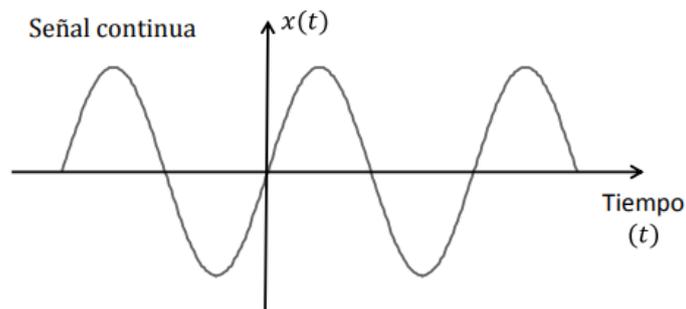


Figura 2.1.- Señal continua [7].

- **Señales discretas.** Las señales discretas por su parte son aquellas cuya variable solo puede tomar un conjunto de valores finito, normalmente tomando valores enteros [7].

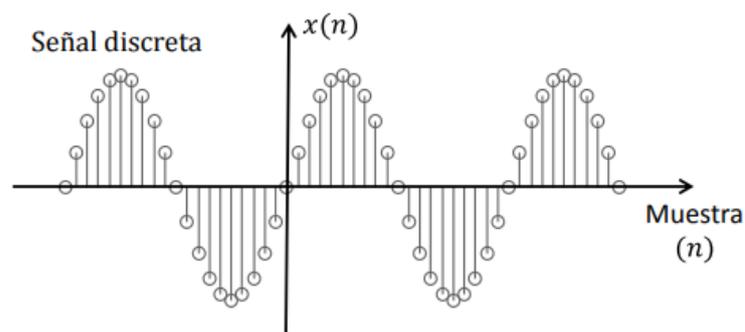


Figura 2.2.- Señal discreta [7].

II. En función de su naturaleza.

- **Señales analógicas.** Son las señales que nos encontramos de forma natural y que varían de forma continua en el tiempo en forma análoga a alguna magnitud física, podemos ver un ejemplo en la figura 2.3. [8]. Este tipo de señales varían entre un límite superior e inferior, y se consideran normalizadas si estos límites coinciden con los que admite un determinado dispositivo. Esto implica una ventaja puesto que se aprovecha mejor la relación señal/ruido del dispositivo.

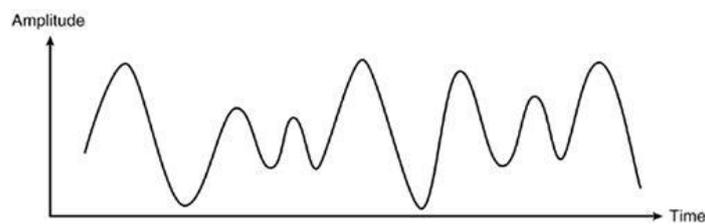


Figura 2.3.- Señal analógica.

- **Señales digitales.** Estas señales son las que varían de forma discreta, típicamente representadas en valores binarios 0 y 1. No obstante, los niveles específicos varían según el tipo de dispositivo utilizado. Una de las grandes ventajas a destacar de las señales digitales es que muestran una mayor inmunidad al ruido que las señales analógicas. En la figura 2.4. podemos ver un ejemplo [8].

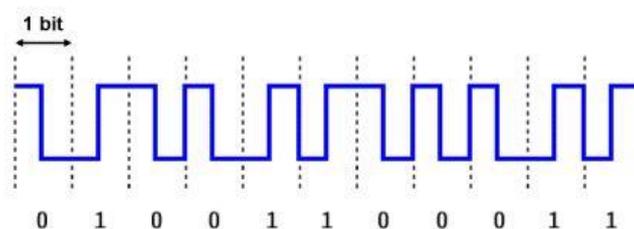


Figura 2.4.- Señal digital.

Dentro de las señales analógicas y digitales se pueden diferenciar tanto señales discretas como continuas. Esto sucede porque la amplitud de la señal puede ser continua o discreta de forma independiente. En la figura 2.5. podemos ver las distintas ondas según su clasificación [1]:

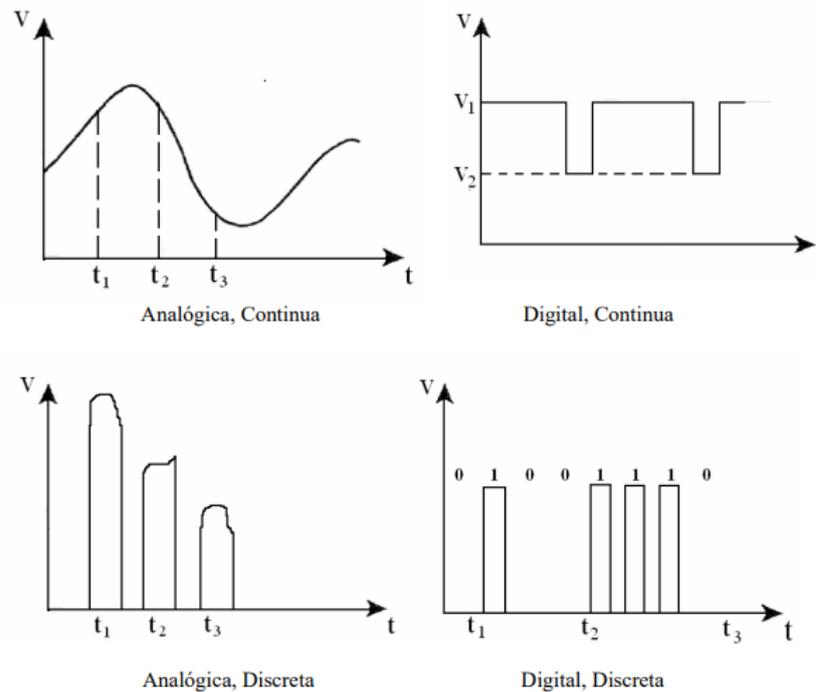


Figura 2.5.- Ejemplos de señales [1].

III. En función de su variación a lo largo del tiempo:

- **Señales constantes.** Son señales que no varían su valor a lo largo del tiempo. Un ejemplo sería el voltaje de una pila. Podemos ver un ejemplo en la figura 2.6 [9].
- **Señales variables.** Son aquellas cuyo valor varía con el paso del tiempo. Un ejemplo sería la voz humana. Dentro de este tipo de señales podemos encontrar las siguientes señales a clasificar:

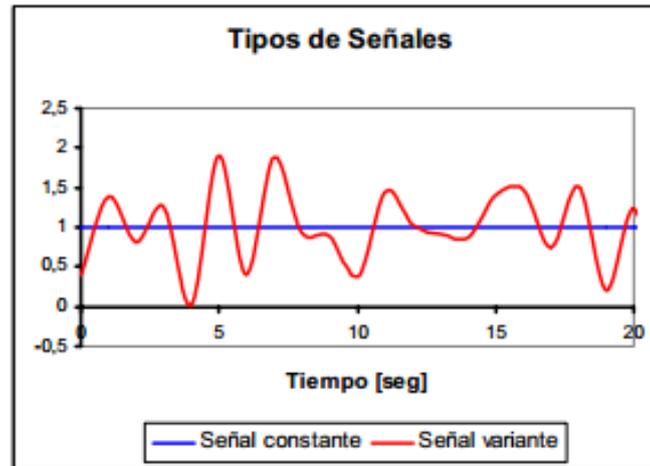


Figura 2.6.- Señal constante y señal variable.

#### IV. Periódicas o no periódicas:

- **Señales periódicas.** Son aquellas que se repiten cada un intervalo de T segundos. A este intervalo se le conoce como período y se mide en segundos. Otro parámetro el cual debemos tener en cuenta es la frecuencia. Las señales periódicas son muy importantes puesto que son las más utilizadas y conocidas. En la siguiente figura 2.7. [10] podemos ver algunas de las más comunes.

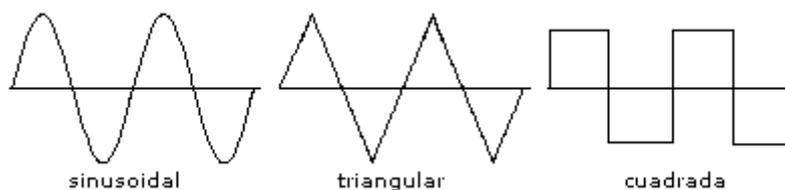


Figura 2.7.- Ondas periódicas.

Matemáticamente las señales periódicas se entienden siguiendo la siguiente función:

$$x(t) = x(t + n \cdot T) \quad -\infty < t < \infty \quad (2.4)$$

- **Señales no periódicas.** Son aquellas que, a diferencia de las anteriores, no siguen un ciclo repetitivo. Esto no significa que sean aleatorias, puesto que dentro de este conjunto podemos encontrar señales crecientes, decrecientes, exponenciales, etc.

## 2.2.- OSCILOSCOPIOS.

El osciloscopio es una herramienta indispensable en los estudios basados en la medición de señales y la electrónica en general. En el mundo de las telecomunicaciones es un dispositivo más que principal a la hora de realizar cualquier tipo de prueba en un laboratorio.

Estos equipos, también conocidos como “oscilógrafos”, sirven para la medición y visualización gráfica de señales eléctricas que pueden variar en el tiempo. Usualmente se suelen utilizar junto a un analizador de espectros.

### 2.2.1.- Historia de los osciloscopios.

El primer osciloscopio se remonta al año 1893. Este primer aparato fue creado a mano y se basaba en un osciloscopio electromagnético con suspensión bifilar. Para representar la señal medida se utilizaba un péndulo con tinta, que con ayuda de una bobina grababa en un papel los valores eléctricos registrados. Éstos primeros osciloscopios carecían de una buena exactitud y su ancho de banda trabajaba entre los 10 y 19 kHz [11].

Con la invención del tubo de rayos catódicos (CRT) en 1897, los osciloscopios evolucionaron gracias a la adaptación en 1932 de esta nueva tecnología, de la mano de la compañía británica A.C. Cossor [11].

A finales de los años 50 se funda la compañía Tektronix, convirtiéndose en el líder mundial en el desarrollo de estos productos. En general, durante estos años la producción de todo tipo

de aparato de medida empezó a crecer por todo el mundo. Tektronix introdujo el primer osciloscopio de exploración retardada, con un ancho de banda de 10MHz [11].



Figura 2.8.- Logotipo de Tektronix [11].

Durante los siguientes años la fabricación de estos dispositivos creció, introduciendo los primeros formatos industriales. Éstos nuevos modelos aumentaron la precisión y eficacia de los osciloscopios, aumentando aún más con la aparición de las versiones digitales. A partir de los años 80, se creó el primer osciloscopio de almacenamiento digital y podemos concluir que ese año fue el inicio de la revolución de esta tecnología, convirtiéndose hoy en día en una herramienta imprescindible [11].

### 2.2.2.- Parámetros de un osciloscopio.

A la hora de elegir o diseñar un osciloscopio se tienen en cuenta los siguientes parámetros:

- **Ancho de banda.** El ancho de banda es el valor que especifica el rango de frecuencias en las que el osciloscopio asegura medir con precisión. Por convenio el ancho de banda se calcula desde 0Hz hasta la frecuencia de corte. Esto se consigue mediante un filtro paso bajo, el cual podemos ver en la figura 2.9.

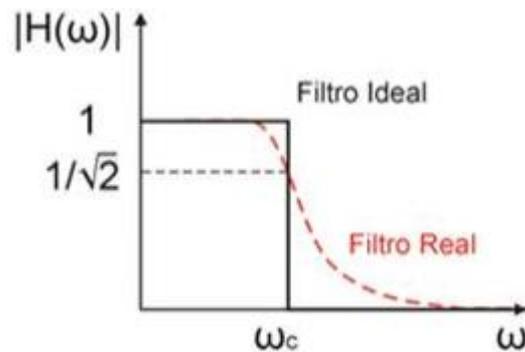


Figura 2.9.- Filtro paso bajo [12].

En la figura 2.9. podemos ver que el filtro paso bajo deja pasar todas las frecuencias por debajo del valor  $\omega_c$ , que es la frecuencia de corte [12]. En la realidad, no es un cuadrado perfecto, sino que a la frecuencia de corte indicada sufre una pérdida en la ganancia. Por lo tanto, a esta frecuencia una señal de tipo senoidal se visualiza a un 70.7% del valor aplicado a la entrada, lo que corresponde con una atenuación de 3dB [13] en unidades logarítmicas.

- **Tiempo de subida.** Es la máxima frecuencia de utilización del osciloscopio. Resulta relevante cuando interesa medir pulsos y flancos.
- **Sensibilidad vertical.** Capacidad de amplificar señales débiles, proporcionado en mV/división vertical.
- **Velocidad.** Indica la velocidad máxima del barrido horizontal. Se proporciona en orden de nanosegundos/división horizontal [13].
- **Exactitud en la ganancia.** Se proporciona en porcentaje máximo de error e indica la exactitud con la que el sistema vertical del osciloscopio amplifica o atenúa la señal.

- **Exactitud de la base de tiempos.** Al igual que la ganancia, se mide en porcentaje máximo de error e indica la precisión con la que el sistema horizontal visualiza el tiempo.
- **Velocidad de muestreo.** Indica la cantidad de muestras por segundo que el osciloscopio digital es capaz de obtener. Este proceso se da en el conversor A/D que contiene el osciloscopio digital.
- **Resolución vertical.** Proporciona el valor de la resolución del conversor analógico-digital, y se mide en bits.
- **Longitud del registro.** Este parámetro indica la cantidad de muestras que se memorizan en un registro para la posterior reconstrucción de la forma de onda, que dependerá de la capacidad de memoria del osciloscopio.
- **Relación señal a ruido (Signal-to-Noise Ratio, SNR).** La relación señal/ruido se define como la proporción existente entre la potencia de la señal que se transmite y la potencia del ruido que la corrompe. Se mide en decibelios (dB) [13]. Su gráfica se ve reflejada en la figura 2.10. [14].



Figura 2.10.- Relación SNR.

### 2.2.3.- Funcionamiento.

Empezando desde la base, y como hemos comentado en la introducción, un osciloscopio se define como un dispositivo que permite la medición de señales eléctricas y su posterior visualización gráfica.

Se emplean los dos ejes X e Y para representar en forma de coordenadas el tiempo y el voltaje, respectivamente. En el eje horizontal aprecia fracciones de tiempo, y dependiendo de la resolución del dispositivo podemos hablar de segundos, milisegundos, etc. En el eje vertical, se controla la amplitud de la tensión, también dependiendo de la resolución. En algunas ocasiones incorpora un “eje Z” o “Cilindro de Wehnelt”. Esta entrada adicional se emplea para controlar la luminosidad del haz y por lo tanto destacar o apagar determinados segmentos de la traza.

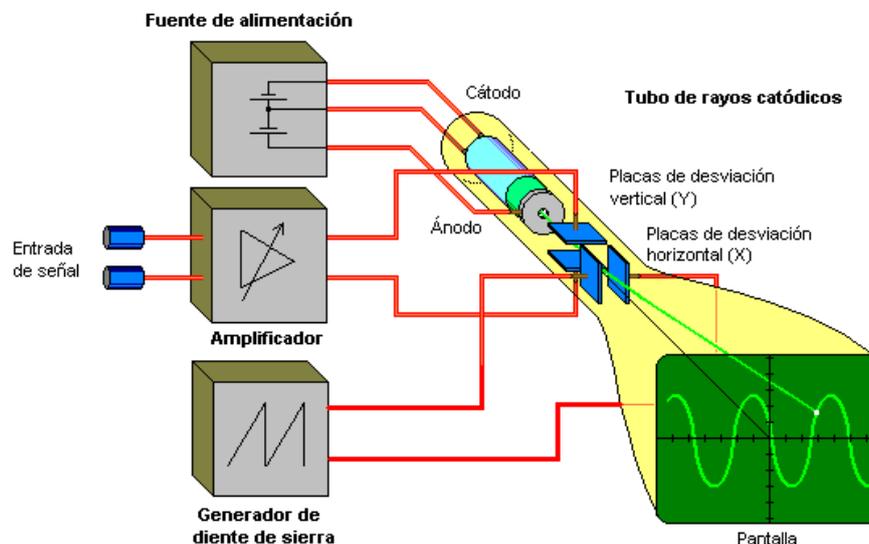


Figura 2.11.- Funcionamiento interno generalizado de un osciloscopio [13].

La imagen que se muestra en pantalla se denomina oscilograma. Algunos dispositivos también muestran otro tipo de valores, como por ejemplo la corriente. Para ello, se necesita disponer de un correcto transductor, el cual nos permitirá medir valores como presión, ritmo cardiaco, nivel de vibraciones en un coche, etc.

Esta característica da a los osciloscopios una gran versatilidad, puesto que puede ser utilizado para un amplio abanico de sectores (por ejemplo, sector de telecomunicaciones o médicos).

#### 2.2.4.- Tipos de osciloscopios.

Atendiendo a las variables con las que se trabaja, existen dos tipos de osciloscopios. Por un lado, los analógicos, los cuales miden variables continuas y trabajan directamente con la señal; por otro lado, los digitales, los cuales miden valores discretos y necesitan de un conversor A/D (analógico-digital) para digitalizar la señal de entrada.

##### 2.2.4.1.- Analógicos.

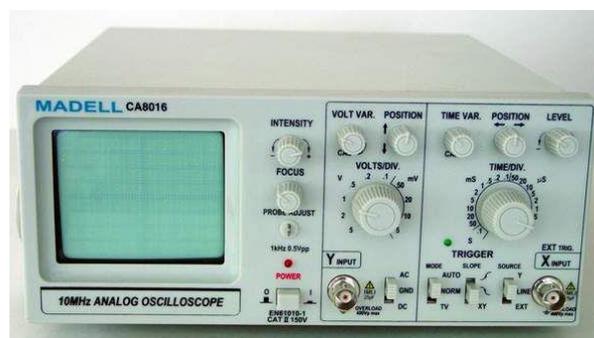


Figura 2.12.- Osciloscopio analógico.

El osciloscopio analógico, el cual podemos ver en la figura 2.12. [15], se compone de un canal horizontal, uno o más canales verticales, un sistema de disparo, una base de tiempo y un módulo de tubo de rayos catódicos (CRT).

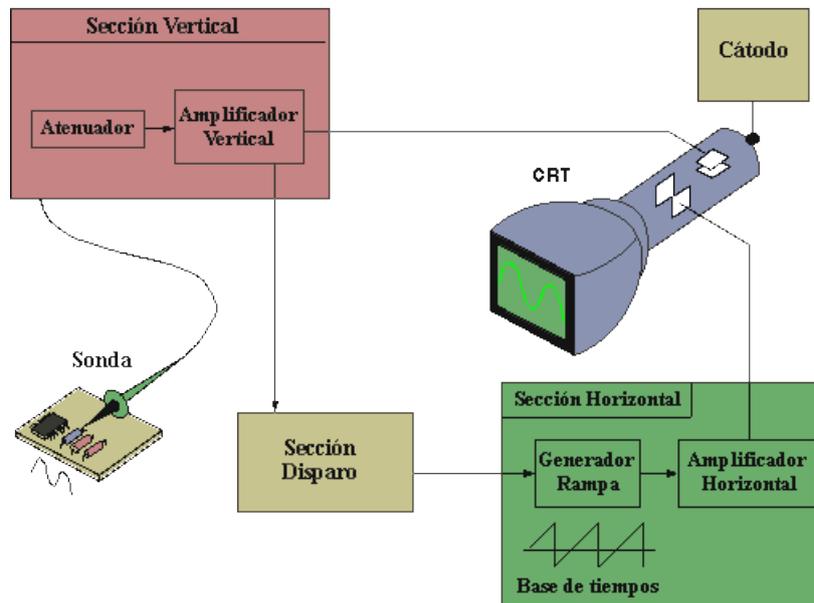


Figura 2.13.- Diagrama de bloques de un osciloscopio analógico [13].

El primer paso se da en la sección vertical, en la cual se introduce la señal medida por la sonda. Dentro de esta sección se encuentra el amplificador vertical, entre otros módulos, que servirá para aumentar o disminuir el nivel de señal hasta el nivel especificado por el/la usuario/a.

En una de las salidas de la sección vertical, se encuentra la sección disparo. Este módulo sirve para iniciar el barrido horizontal.

En las placas de desviación horizontal se aplica una tensión en diente de sierra, generado por un circuito oscilador. Esta sección tiene dos modos de funcionamiento: interno y externo.

La frecuencia a la que trabaja la señal generada por el circuito oscilador puede ser ajustada, adaptándose a la frecuencia de la señal bajo estudio (a esto se le llama “base de tiempos”). La base de tiempos se compone de un amplificador integrador, disparadores y esquemas para sumar e invertir.

En la figura 2.14. podemos ver la imagen de un osciloscopio en funcionamiento [16].

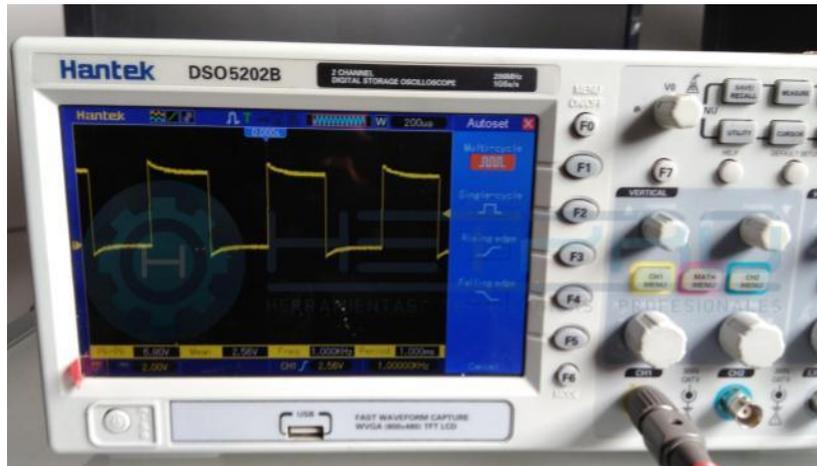


Figura 2.14.- Osciloscopio Hantek en funcionamiento.

#### 2.2.4.2.- Digitales.

En la actualidad los osciloscopios analógicos están comenzando a ser sustituidos por los digitales [17].

El desarrollo de esta digitalización aparece gracias a la revolución tecnológica de convertidores analógico-digitales (ADC), los cuales permiten una mayor rapidez y exactitud en la digitalización de señales de altas frecuencias; los avances en dispositivos de memoria, puesto que se necesita almacenar datos tan rápido como se ejecuta el muestreo y los módulos de pantalla de bajo consumo compactos [11]. También ha sido significativa la facilidad que ofrecen para poder transferir las medidas obtenidas a una computadora personal o una pantalla LCD [17].

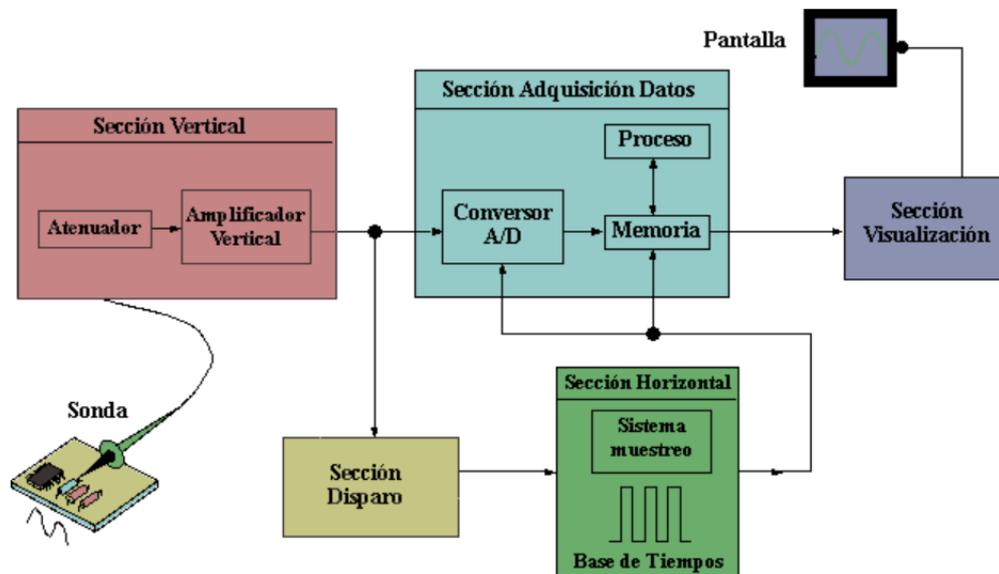


Figura 2.15.- Diagrama de bloques de un osciloscopio digital [13].

Un osciloscopio digital sigue la estructura mostrada en la figura 2.15. Los osciloscopios digitales se componen principalmente de las mismas características que los osciloscopios analógicos, pero se diferencian por contener un sistema adicional de proceso de datos que permite almacenar y ver la señal [13].

Por otro lado, estos osciloscopios necesitan de un convertor analógico digital para digitalizar la señal medida. Esto pone al convertidor A/D como la pieza fundamental del dispositivo, puesto que de él depende la fiabilidad de la señal final [17].

Cuando se conecta la sonda de un osciloscopio digital a un circuito, la sección vertical ajusta la amplitud de la señal de la misma forma que lo hacía el osciloscopio analógico.

El convertor analógico-digital del sistema de adquisición de datos muestrea la señal a intervalos de tiempo determinados y convierte la señal de voltaje continua en una serie de valores digitales llamados *muestras*. En la sección horizontal una señal de reloj determina cuando el convertidor A/D toma una muestra. La velocidad de este reloj se denomina velocidad de muestreo y se mide en muestras por segundo.

Los valores digitales muestreados se almacenan en una memoria como puntos de señal. El número de los puntos de señal utilizados para reconstruir la señal en pantalla se denomina registro. La sección de disparo determina el comienzo y el final de los puntos de señal en el registro. La sección de visualización recibe estos puntos del registro, una vez almacenados en la memoria, para presentar en pantalla la señal.

Dependiendo de las capacidades del osciloscopio se pueden tener procesos adicionales sobre los puntos muestreados, incluso se puede disponer de un predisparo, para observar procesos que tengan lugar antes del disparo.

Fundamentalmente, un osciloscopio digital se maneja de una forma similar a uno analógico, para poder tomar las medidas se necesita ajustar el mando AMPL., el mando TIMEBASE así como los mandos que intervienen en el disparo [13].

Una de las principales características de estos dispositivos digitales es la frecuencia de muestreo, expresada en MS/s, que determina el ancho de banda máximo al que podemos medir con el oscilógrafo. Para el control del conversor, memoria, buffers y demás componentes digitales se utiliza una FGPA (Field Programmable Gate Array).

Dadas las características que presentan estos osciloscopios se añaden ciertas prestaciones al consumidor, como la medida de valores de pico, máximos y mínimos; la captura de transitorios y un largo etc [17].

Dentro de la familia de los osciloscopios digitales, se pueden diferenciar tres tipos:

- **Osciloscopios de memoria (DSO):** Estos osciloscopios nacen de la revolución tecnológica de los convertidores analógico-digitales, lo que permite a este tipo de osciloscopios presentar los datos en formato digital. Utilizan tecnología de muestreo en tiempo real [11].

- **Osciloscopios estroboscópicos (DSaO):** Estos dispositivos utilizan una muestra ordenada/aleatoria de valores instantáneos de la señal a medir y realizan su transformación temporal para reflejar la forma de estas señales. Los DSaO utilizan una medición de valores instantáneos de señales repetidas con ayuda de impulsos estroboscópicos cortos, lo que permite a estos dispositivos tener una banda pasante ancha y una alta sensibilidad. Utilizan una tecnología de muestreo en tiempo equivalente [11].
- **Osciloscopios de fósforo (DPO):** Este tipo de osciloscopio es el más avanzado. La señal es reflejada en tres planos, lo que le da capacidades similares a las de un osciloscopio analógico. Se caracterizan por una alta densidad de muestreo y una capacidad de leer datos conforme a la intensidad de señal investigada. Utilizan tecnologías avanzadas de muestreo y procesamiento de señales [11].

En la figura 2.16. [18] podemos ver un ejemplo de un osciloscopio digital.



Figura 2.16.- Osciloscopio digital.

### 2.2.5.- Modos de funcionamiento

En este punto vamos a hablar de dos tipos de funcionamiento que tienen los osciloscopios: el funcionamiento en modo XT y el funcionamiento en modo XY.

### 2.2.5.1.- Funcionamiento en modo XT

Este método es el más empleado habitualmente, y funciona de la siguiente manera: cuando a las placas horizontales se les aplica una tensión generada internamente en forma de diente de sierra (figura 2.17) y a las verticales se les aplica la señal que queremos estudiar (por ejemplo sinusoidal), la combinación de ambas tensiones hace que la sinusoide quede dibujada en la pantalla.

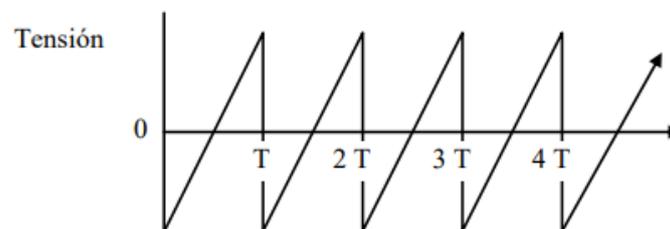


Figura 2.17.- Señal diente de sierra [19].

La tensión horizontal de diente de sierra se denomina voltaje de barrido, puesto que se utiliza para barrer el haz de electrones sobre la pantalla: los electrones son desviados linealmente con el tiempo en la dirección horizontal durante el tiempo  $T$ . A continuación, el haz vuelve muy rápidamente a su punto de partida sobre la pantalla al final de cada periodo.

El osciloscopio suele constar de dos entradas de señal que, en este modo de operar, permiten ver alternativamente las dos señales en función del tiempo o la suma de ambas [19].

### 2.2.5.2.- Funcionamiento en modo XY

Este control consta de un pequeño conmutador en forma de botón que permite desconectar el sistema de barrido interno del osciloscopio, haciendo estas funciones uno de los canales verticales (generalmente el canal II). Este modo nos permite visualizar curvas de respuesta o las famosas figuras de Lissajous, útiles tanto para la medida de fase como de frecuencia [13].

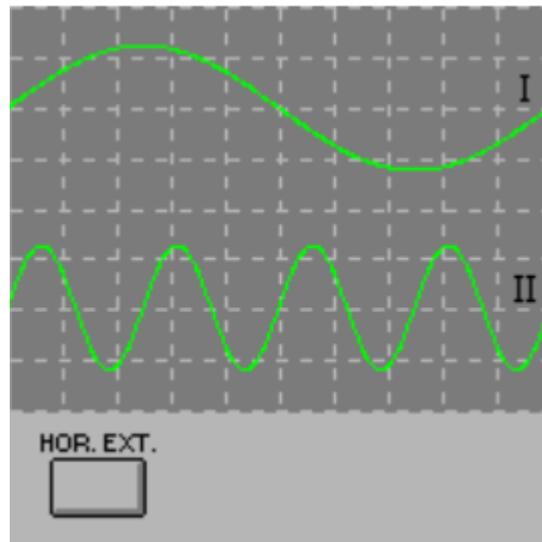


Figura 2.19.- Funcionamiento en modo XT [13].

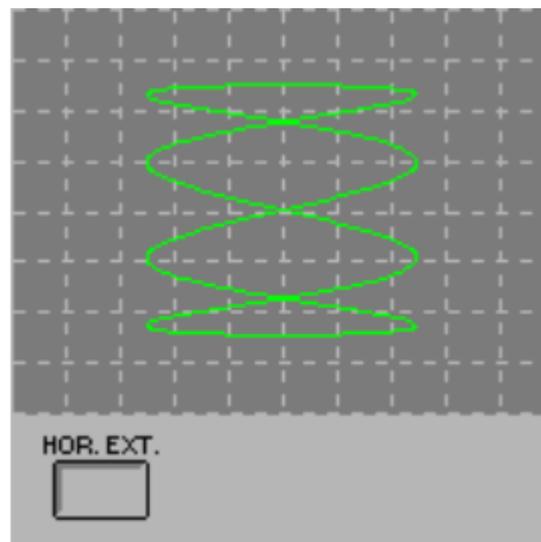


Figura 2.18.- Funcionamiento en modo XY. Figura de Lissajous [13].

Es decir, en este modo de funcionamiento el osciloscopio muestra en la pantalla la composición resultante de dos señales: una se introduce en el canal I y otra en el canal II, aplicada ahora una de ellas a las placas horizontales y la otra a las verticales. Por lo tanto, en la posición XY no actúa la base de tiempos [19].

### **2.3.- CONVERSORES A/D.**

La posibilidad de desarrollar dispositivos con la capacidad de convertir una señal analógica a digital o viceversa surge de la aparición de dispositivos microprocesadores y procesadores digitales de señal (DSP), los cuales han permitido realizar tareas que durante años han sido procesadas por sistemas electrónicos analógicos. Gracias al uso de estos procesadores digitales se puede obtener sistemas de medición más fiables, flexibles y de menor coste.

En segundo lugar, los procesadores digitales no pueden trabajar con señales continuas de forma directa, lo cual supone un problema dado que el mundo físico maneja señales de naturaleza analógica. Esto implica la necesidad de una interfaz bidireccional, que son los llamados convertidores analógico-digital y convertidores digital-analógico.

Por una parte, la conversión mundo analógico – mundo digital consiste en una primera etapa para transformar la magnitud física mediante un transductor en su análoga eléctrica. Esta señal eléctrica entra al convertidor ADC en donde la señal es cuantizada y codificada, para que luego en formato digital pueda ser entendido por el procesador.

Para que dicha señal ingrese al convertidor, ésta debe ser muestreada y retenida previamente, proceso que recibe el nombre de *sampling and holding*. Por otra parte, en el proceso inverso la señal digital que sale del procesador debe pasar por el convertidor digital-analógico en donde se obtiene una señal analógica en escalones. Estos escalones se eliminan mediante un filtro paso bajo [20].

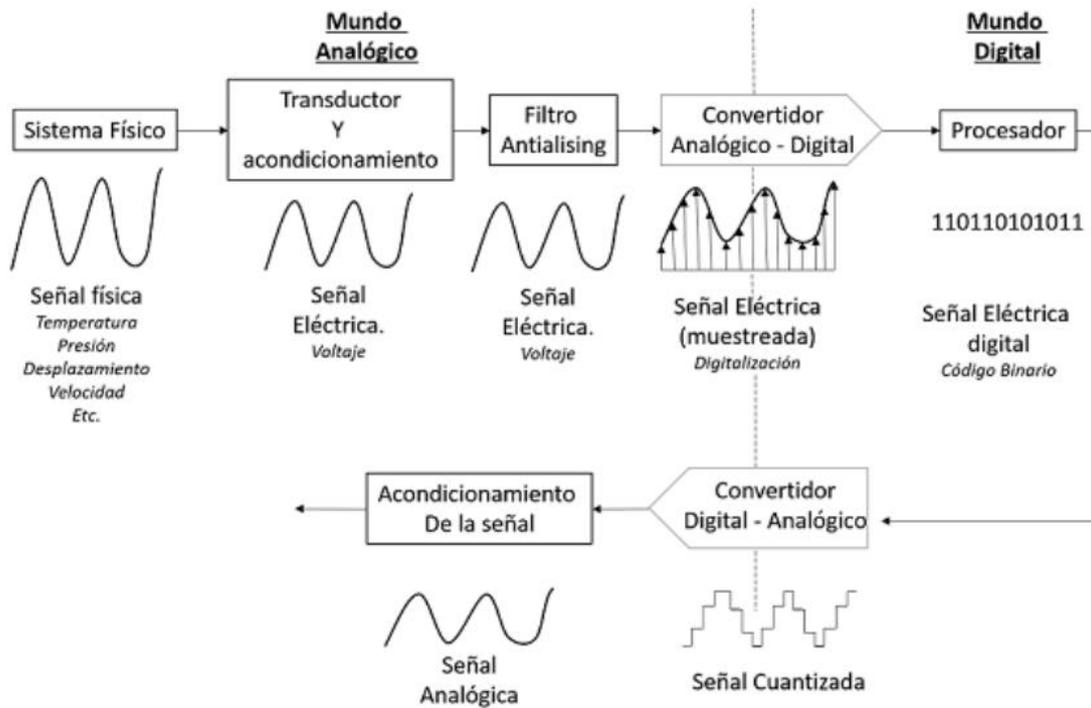


Figura 2.20.- Proceso de conversión [20].

### 2.3.1.- Conceptos básicos de la conversión.

#### 2.3.1.1.- Muestreo

El muestreo es uno de los principales pasos en la conversión analógica-digital. Consiste en medir una señal analógica en intervalos de tiempo llamados instantes de muestreo, o periodo de muestreo ( $T_s$ ).

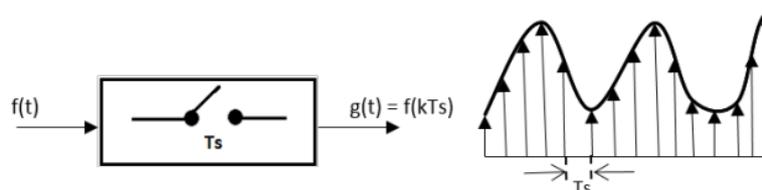


Figura 2.21.- Muestreo [20].

En la figura 2.21 podemos ver expresado en un diagrama de bloques cómo funciona el proceso de muestreo de una señal. El muestreador se comporta como un interruptor, el cual captura la señal en un intervalo de tiempo de duración  $T_s$ .

El valor del tiempo de muestreo tiene mucha importancia en el proceso de conversión ya que la fiabilidad de los datos digitales dependerá del intervalo de muestreo seleccionado. Según el teorema de Nyquist-Shannon nos indica que la frecuencia de muestreo ( $f_s$ ) ha de ser siempre mayor al doble de la máxima frecuencia del espectro de la señal de entrada ( $f_m$ ), para que se pueda garantizar la recuperación de la señal de forma correcta [20].

$$F_s \geq 2 \cdot f_m \quad (2.5)$$

Donde  $f_m$  es la frecuencia máxima de entrada.

Uno de los problemas que se pueden ocasionar a la señal procesada si no se cumple el teorema de Nyquist-Shannon es la aparición del “aliasing”. Éste fenómeno aparece cuando se solapa parte del espectro de la señal, y por lo tanto da a lugar a pérdida de la información. En la figura 2.22. [21] podemos ver más gráficamente el problema de solapamiento o “aliasing”.

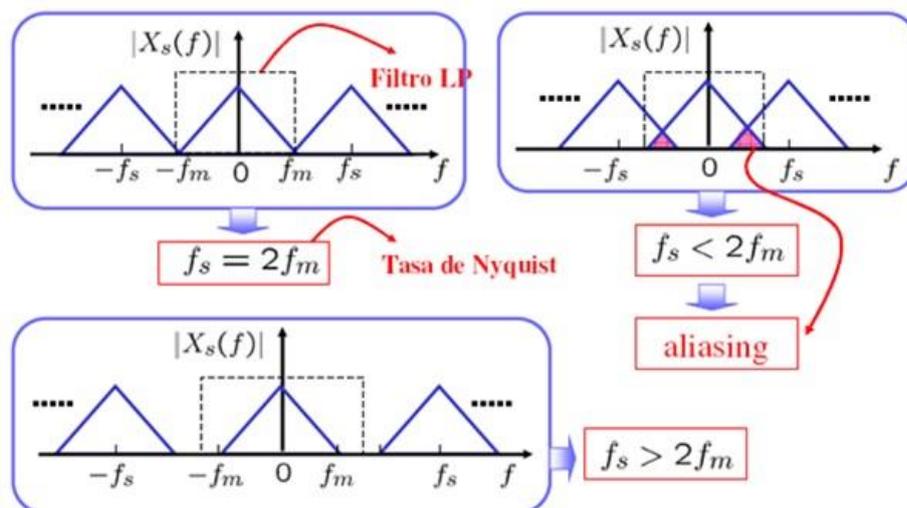


Figura 2.22.- Problema de aliasing.

El aliasing puede ser combatido de dos formas, muestreando a frecuencias mayores que cumplan como mínimo el criterio de Nyquist o filtrando la señal analógica antes de que sea sampleada y quitando por lo tanto las frecuencias que sean mayores de la mitad de la frecuencia de muestreo [22].

### 2.3.1.2.- Retención

En muchos casos, específicamente en el de la conversión analógica-digital, hace falta un valor constante durante cierto intervalo de tiempo para efectuar una conversión correcta. Para lograr esto, es preciso realizar un muestreo con retención.

Un sistema de muestreo y retención (sample and hold) ideal transmitirá instantáneamente y sin errores el valor de su entrada a la salida durante la etapa de muestreo y conservará dicho valor indefinidamente, durante la etapa de retención, hasta que venga otra etapa de muestreo [23]. Con este tipo de sistema podemos controlar que no se generen errores en la señal digital en el caso de que la señal de entrada varíe durante el proceso de conversión.

Típicamente, el tiempo de muestreo será corto, y el de retención más largo.

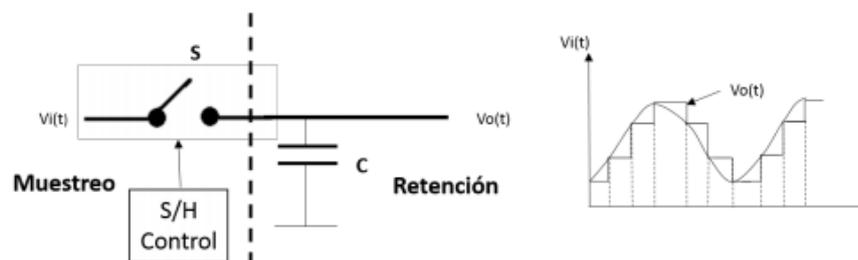


Figura 2.23.- Sistema de muestreo-retención ideal [20].

### 2.3.1.3.- Cuantificación

La cuantificación es el paso de la conversión en el que la señal analógica de entrada, una vez muestreada, se convierte con una aproximación por un conjunto finito de valores discretos. A este paso también se le conoce como redondeo.

Un parámetro para controlar la precisión sería el número de bits con el que representamos la señal. A este número se le conoce como resolución, y a cuanto más resolución tengamos más detalle obtendremos de la señal.

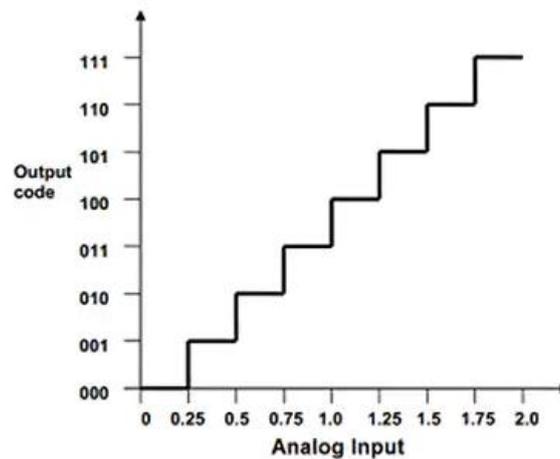


Figura 2.24.- Función de transferencia de un ADC de 3 bits.

En el caso de ejemplo de la figura 2.24, para saber la resolución con la que trabajaríamos se calcularía mediante la expresión:

$$N = 2^n \quad (2.6)$$

Donde n es el número de bits y N el número de niveles. La resolución será de  $2^3 = 8$ .

Los cuantificadores más conocidos se conocen como lineal o uniforme. En este caso el intervalo de cuantificación tiene un valor constante 'q', que corresponde a la resolución del convertidor.

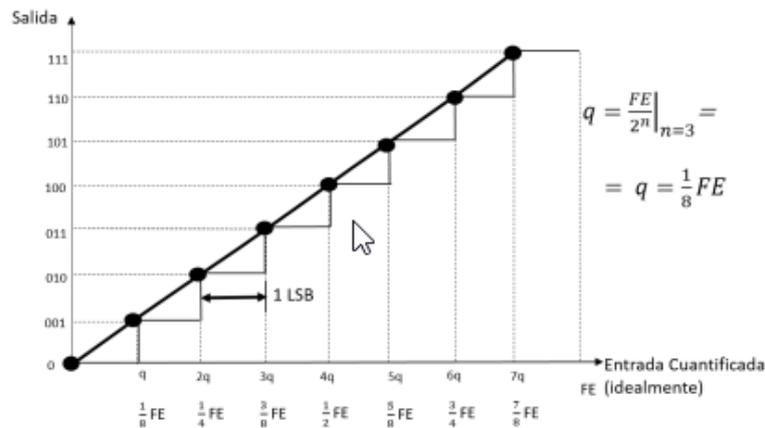


Figura 2.25.- Cuantificador Uniforme Ideal [20].

➤ Error de cuantificación

El error de cuantificación se define como la diferencia existente entre la señal muestreada y la señal discreta cuantificada. Este valor es inversamente proporcional al número de niveles  $N$ , lo que indica como hemos mencionado anteriormente que a mayor número de niveles y bits más fiable será la señal procesada [20].

➤ Ruido de cuantificación

El ruido de cuantificación es un modelo de error de cuantificación que se introduce en el ADC durante este proceso. Es un error de redondeo entre el voltaje de entrada analógica al ADC y el valor digital de salida. Este ruido es lineal y depende de la señal.

En un ADC ideal, donde el error de cuantificación se distribuye uniformemente y la señal tiene una distribución uniforme, la relación señal-ruido de cuantificación (SQNR) puede ser calculado de la siguiente forma [24]:

$$SQNR = 20 \cdot \log_{10} 2^Q \approx 6,02 \cdot Q \text{ (dB)} \quad (2.7)$$

Donde  $Q$  es el número de bits cuantificados.

Cuando la señal de entrada es una onda sinusoidal de amplitud completa, la distribución de la señal ya no es uniforme y se calcularía de la siguiente forma [24]:

$$SQNR \approx 1.761 + 6,02 \cdot Q \text{ (dB)} \quad (2.8)$$

#### 2.3.1.4.- Codificación

La codificación es el paso en la conversión analógica digital en la cual se representan mediante un código binario los diferentes estados del cuantificador.

Para los convertidores se suele utilizar el código binario unipolar de tipo directo o natural. En este tipo de código cada bit tiene asociado una potencia de dos, en función de su posición en la palabra digital. Se rigen por la siguiente expresión:

$$b_{n1} \cdot b_{n2} \cdots b_1 b_0 = b_{n1} \cdot 2_{n-1} + b_{n-1} \cdot 2_{n-2} + \cdots + b_1 \cdot 2_1 + b_0 \cdot 2_0 \text{ donde } b \in [0,1] \quad (2.9)$$

Al bit del extremo izquierdo o bit de mayor peso se le denomina bit más significativo o MSB (Most Significant Bit) y al del extremo derecho o de menor peso bit menos significativo o LSB (Least Significant Bit).

Por otro lado existen otro tipo de códigos binarios, como el código binario bipolar. En este grupo podemos encontrar el binario con signo, el binario en complemento a uno, el binario en complemento a dos o el binario desplazado/binario con offset [20].

Dentro de las técnicas de codificación, podemos encontrar las siguientes:

- **Codificación unipolar.** Se basan en códigos que solo distinguen dos tipos de niveles de tensión, uno para el 0 binario y otro para el 1 binario [25].

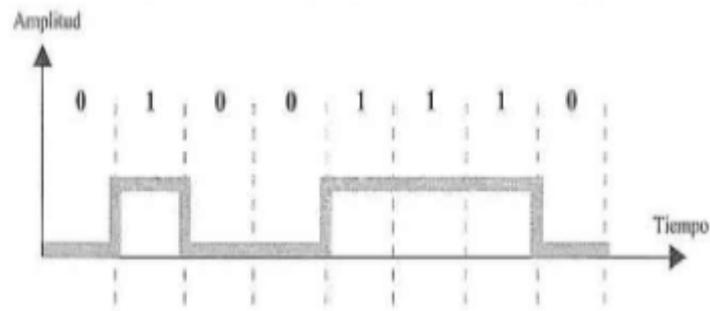


Figura 2.26.- Codificación unipolar [25].

- **Codificación polar.** Se utilizan dos niveles de tensión, positivo y negativo. En este caso existen algunas variantes muy populares como Sin Retorno a Cero (NRZ, No Return to Zero), Manchester, etc [25].

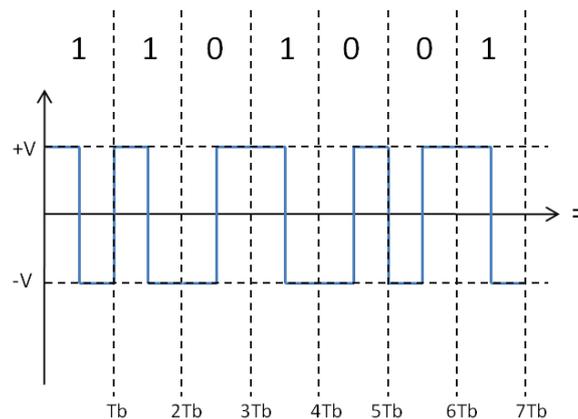


Figura 2.27.- Codificación polar.

- **Codificación bipolar.** En este caso se utilizan tres niveles de tensión: positivo, negativo y nulo. El nivel nulo se representa con el 0, y el positivo y negativo se van representando con el bit 1 alternándose [25]. Podemos ver un ejemplo en la figura 2.28. [26].

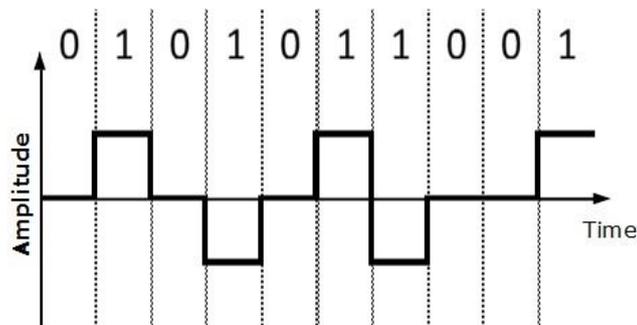


Figura 2.28.- Codificación bipolar.

### 2.3.2.- Características de un ADC.

A continuación se mencionan las dos grandes categorías en las que podemos dividir las características que definen a un convertidor analógico-digital: las características dinámicas y las estáticas. Estas características son importantes a tener en cuenta cuando tratamos con un convertidor para poder entender su funcionamiento y los parámetros que lo definen y poder seleccionar el ADC que mejor se adapte a nuestras necesidades.

#### 2.3.2.1.- Características estáticas

Las características estáticas de un ADC son las que nos permite analizar cuánto se desvían sus características de transferencia de lo que sería la curva ideal. Estas características son de vital importancia en aplicaciones donde se usa el convertidor para medir eventos físicos [22].

A continuación definimos las más importantes:

- **Resolución.** Como hemos mencionado ya anteriormente, se trata de la mínima variación en amplitud a la entrada necesaria para provocar un cambio en la salida.
- **Margen de entrada.** Margen de valores de entrada necesario para que el convertidor pueda devolver un valor a la salida.

- **No linealidad diferencial, DNL (differential Non-Linearity).** Diferencia entre el ancho de un escalón/nivel real y el ideal [20].

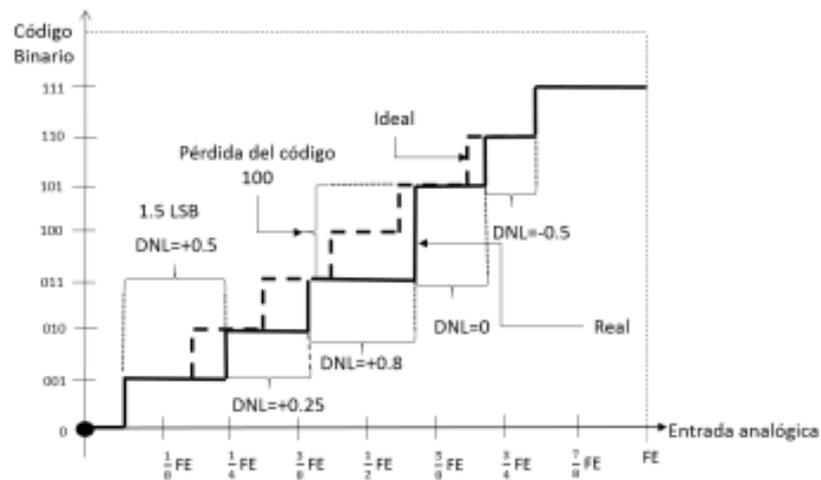


Figura 2.29.- Error DNL [20].

- **No linealidad integral, INL (Integral Non-Linearity).** Indica la máxima variación entre la línea ideal y la real generada al unir los puntos centrales de los escalones reales [20].

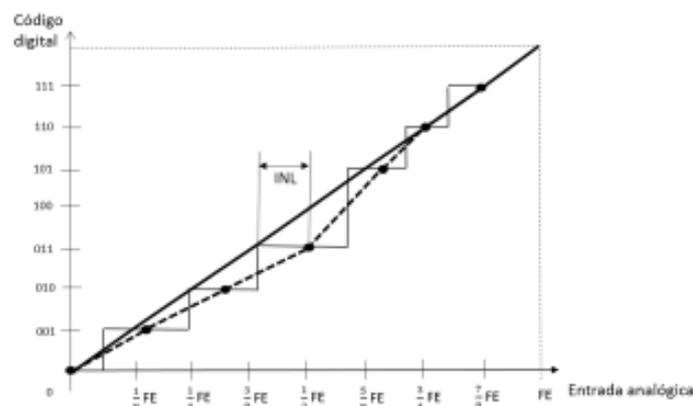


Figura 2.30.- Error INL [20].

- **Pérdida de códigos.** La pérdida de códigos (Missing codes) sólo se produce cuando la DNL es mayor que 0.5 LSB y también es una característica importante para tener en cuenta.

- **Error de offset.** El error de offset es la diferencia entre el punto nominal de offset (cero) y el punto real. Otra manera de definirlo es como el desplazamiento horizontal de la señal. Este error afecta a todas las señales de entrada y puede ser compensado mediante procesos de ajuste [27].
- **Error de ganancia.** El error de ganancia se define como la variación entre el valor de la ganancia esperado y el real. Por ello este valor es proporcional al valor de entrada y se expresa en tanto por ciento.

#### 2.3.2.2.- Características dinámicas

Estas características también son muy importantes para el convertidor y su rendimiento, y surgen a medida que la frecuencia de la señal de entrada incrementa. En estas características, el análisis del dominio de la frecuencia toma más importancia ya que las imperfecciones que aparecen en ella introducen ruido y distorsión en la muestra de salida.

Las características dinámicas o AC de un convertidor nos indican por lo tanto cuánto ruido y distorsión ha sido introducido en una señal muestreada y la precisión del convertidor para cierta frecuencia de entrada y de muestreo [22].

Las características dinámicas más importantes son las siguientes:

- **Tiempo de conversión ( $T_c$ ).** Este parámetro indica el tiempo total que requiere el convertidor en procesar la señal de entrada. Según el ADC puede ser constante o variable.
- **Frecuencia de conversión ( $F_s$ ).** Se mide en Hertzios (Hz) o en muestras por segundos (Samples/s), e indica la frecuencia máxima a la que el convertidor opera de manera estable.

- **Distorsión harmónica total (Total Harmonic Distortion, THD).** Cociente entre la suma del valor RMS de todos los armónicos de la señal de entrada y el valor RMS de la misma a fondo de escala. Este valor se representa en dB, y normalmente en la práctica solo se consideran los 6 primeros armónicos ya que son los que representan la mayoría de la distorsión.

$$THD_{DB} = 20 \cdot \log \sqrt{\frac{V_{f_2}^2 + V_{f_3}^2 + V_{f_4}^2 + V_{f_5}^2 + V_{f_6}^2 + V_{f_7}^2}{V_{f_1}^2}} \quad (2.10)$$

- **Relación señal ruido más distorsión.** Se refiere a la relación directa entre la amplitud RMS de la señal y el valor RMS del ruido de cuantificación, sumando la posible distorsión generada debido a los errores de linealidad.

$$SINAD_{DB} = 20 \cdot \log \sqrt{\frac{V_{f_1}^2}{V_{f_2}^2 + V_{f_3}^2 + V_{f_4}^2 + V_{f_5}^2 + V_{f_6}^2 + V_{f_7}^2 + V_{ruidoRMS}^2}} \quad (2.11)$$

- **Número efectivo de bits.** Proporciona la relación señal ruido más distorsión en termino de bits.
- **Rango dinámico libre de espurios.** Indica la diferencia entre la componente fundamental y el armónico de pico del espectro de frecuencias a la salida del convertidor.

### 2.3.3.- Tipos de convertidores ADC.

A continuación pasamos a describir algunos tipos de convertidores ADC más conocidos.

### 2.3.3.1.- Convertidor de aproximaciones sucesivas

Los convertidores de aproximaciones sucesivas son de tiempo de conversión constante y son uno de los más utilizados en la actualidad para sistemas de adquisición de datos debido a que permite una velocidad de conversión considerable y una alta resolución a un bajo coste. Su estructura se compone de tres bloques principales: un convertidor digital-analógico (DAC), un comparador y una lógica de control (SAR).

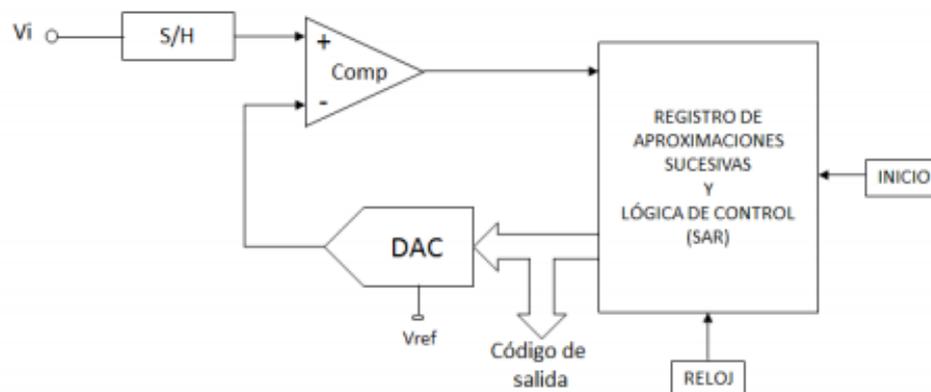


Figura 2.31.- Convertidor ADC de aproximaciones sucesivas [20].

Como podemos ver en la figura 2.31, además de los tres principales bloques también es necesario colocar a la entrada un circuito de muestreo y retención (S/H) para mantener la señal de entrada permanente durante la conversión. Esto es imprescindible puesto que en caso contrario se generarían graves errores en el proceso de conversión.

Los convertidores de aproximaciones sucesivas proporcionan resoluciones de hasta 16 bits con tiempos de conversión muy reducidos. Por ello estos tipos de convertidores suelen utilizarse en aplicaciones en las que se requiere una exactitud moderada de entre 12-16 bits [20].

### 2.3.3.2.- Convertidor flash o paralelo

Los convertidores flash o paralelo se caracterizan tener un tiempo de conversión constante. A pesar de ser muy sencillo, es el convertidor más veloz y complejo a nivel de implementación.

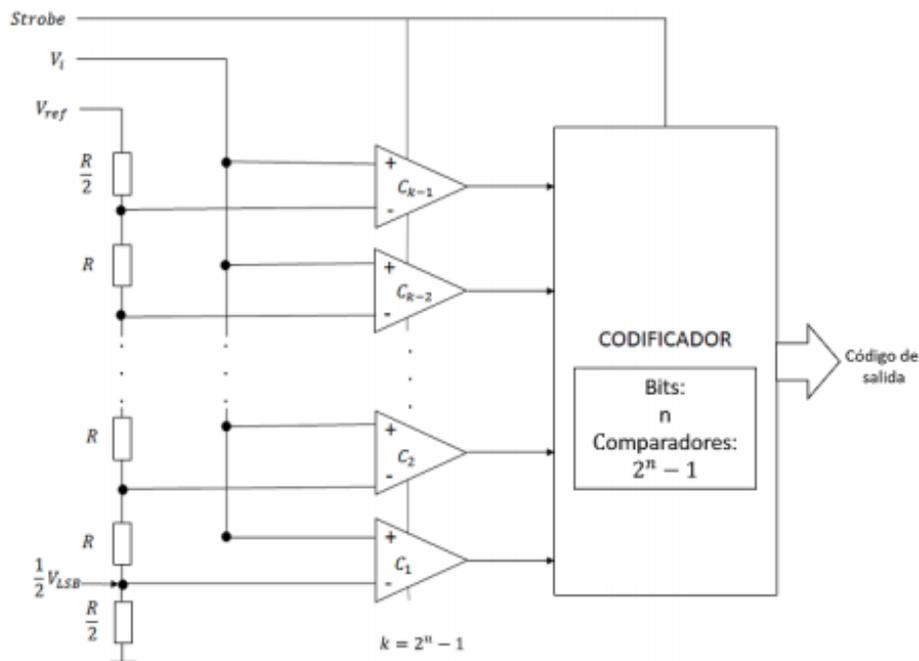


Figura 2.32.- Convertidor ADC de tipo flash [20].

En la figura 2.32 vemos que es necesario un comparador para poder comparar la señal de entrada con un nivel de cuantificación; por consiguiente para obtener una resolución de  $n$  bits hacen falta  $2^n - 1$  comparadores y  $2^n$  resistencias, por lo cual lo convierte en un convertidor muy complejo de diseñar. Por este motivo en la práctica se suelen presentar convertidores de hasta 8 bits [20].

La velocidad de conversión es bastante elevada puesto que solo depende del comparador que se utilice, y por lo tanto consigue alcanzar niveles de 500MS/s.

### 2.3.3.3.- Convertidor semi-paralelo o half-flash

El convertidor semi-paralelo nace de la necesidad de conseguir más resolución a menor coste, aunque esto suponga obtener niveles de velocidad más bajos. Se compone de dos convertidores de tipo flash.

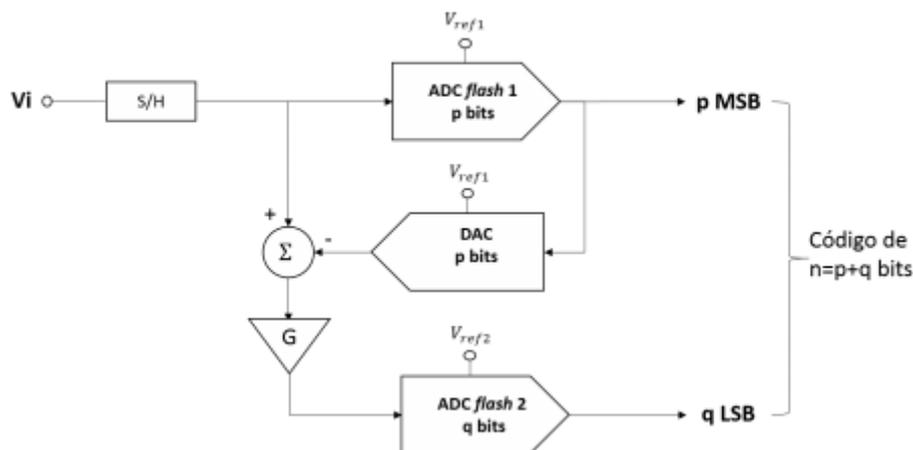


Figura 2.33.- Convertidor ADC de tipo half flash [20].

### 2.3.3.4.- Convertidor Sigma-Delta

Estos convertidores se utilizan principalmente en aplicaciones de baja y media velocidad que requieren una elevada resolución. Son comunes en aplicaciones de audio, instrumentación, sonar, etc. A nivel de implementación es muy sencillo, ya que está conformado por circuitos analógicos activos fáciles de diseñar [20].

En la figura 2.34 [20] podemos distinguir estos circuitos analógicos:

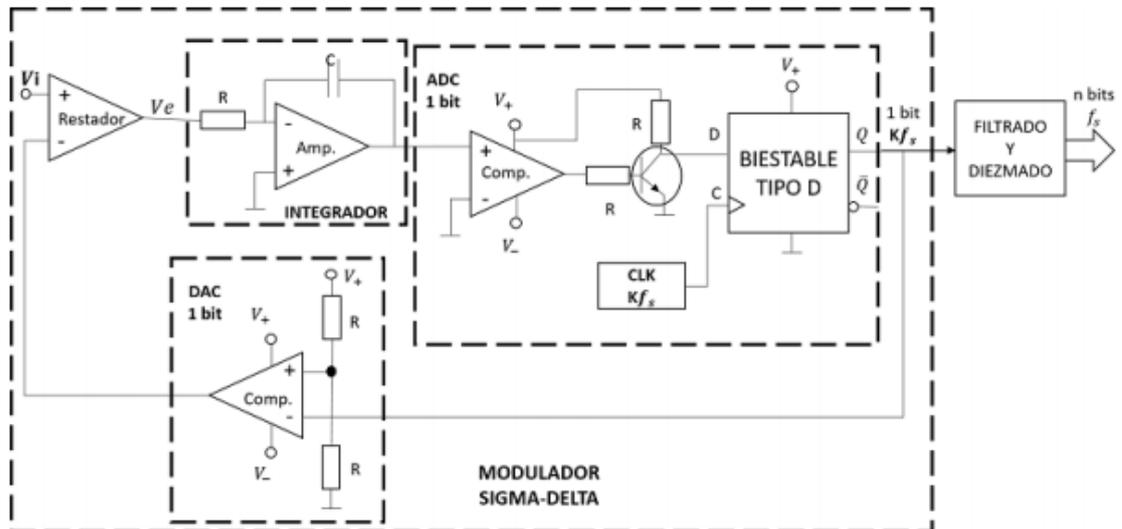


Figura 2.34.- Convertidor ADC Sigma-Delta

Dependiendo del modulador podemos conseguir altas resoluciones del orden de entre 24 y 26 bits y frecuencias de conversión de pocas centenas de kHz.

### 2.3.3.5.- Comparación de los tipos de conversores

A continuación en la tabla 2.1 podemos ver una comparativa de los distintos tipos de conversores previamente mencionados en función de las principales características.

	Aproximaciones sucesivas	Paralelo	Semi-paralelo	Sigma-Delta
Complejidad de implementación	Media	Alta	Media	Media
Velocidad de conversión	Alta	Alta	Media - Alta	Baja-Media
Resolución	Media-Alta	Baja-Media	Media	Alta
Coste	Bajo	Alto	Bajo	Bajo

Tabla 2.1.- Tabla comparativa de los distintos convertidores ADC.

## 2.4.- CONVERSORES DA.

Muchos osciloscopios tienen la funcionalidad no solo de medir señales eléctricas, sino que también la capacidad de generarlas. Para ello se dispone de un conversor digital-analógico. El objetivo principal de un DAC es transformar un número digital en una señal eléctrica analógica, a diferencia de los ADC, que hace la función contraria.

### 2.4.1.- Características de un DAC.

Es importante conocer y tener la capacidad de interpretar las especificaciones de los conversores digital-analógico para su correcta aplicación. Las especificaciones ponen en manifiesto las limitaciones y las verdaderas prestaciones, que la mayoría de las veces difieren de la idealidad.

La primera característica es la función de transferencia ideal, es decir, qué tipo salida se requiere (unipolar o bipolar), y que formato digital de entrada se tiene [23]. En la figura 2.35. [20] podemos ver la función de transferencia de un convertidor con una entrada en

codificación binaria y una salida unipolar. Por otro lado, en la figura 2.36. [20] podemos ver el caso de un convertidor con una entrada en formato binario con offset y una salida bipolar.

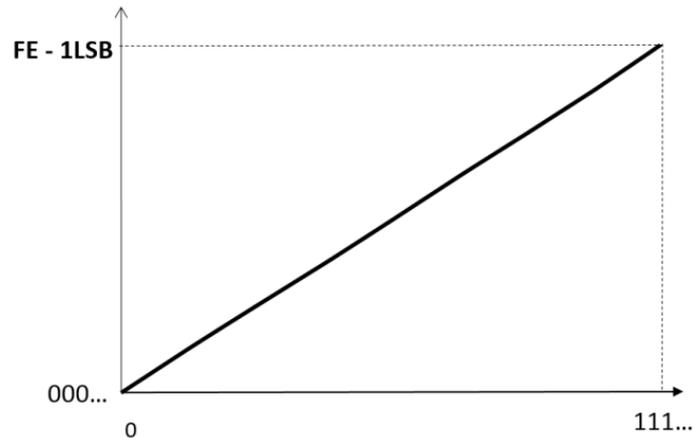


Figura 2.35.- Función de transferencia entrada codificación binaria y salida unipolar.

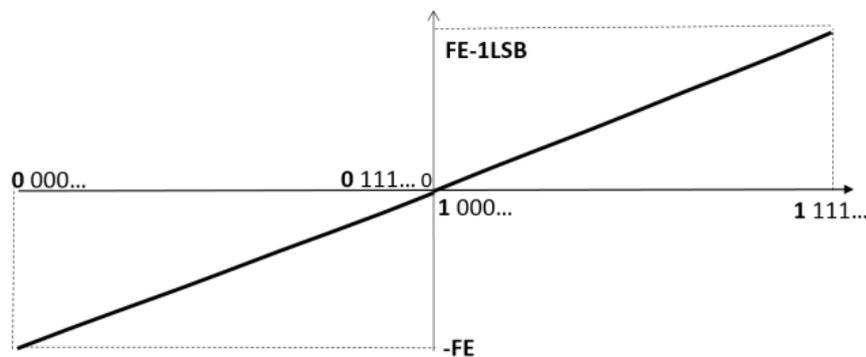


Figura 2.36.- Función de transferencia entrada codificación binaria con offset bipolar y salida en modo bipolar.

#### 2.4.1.1.- Características estáticas

Además de la traza ideal, las características estáticas del convertidor son muy importantes, al igual que para el ADC como previamente hemos comentado. Mencionamos algunas de las más importantes:

- **Resolución.** La resolución indica la cantidad de bits que se acepta a la entrada del convertidor. También se puede definir como la diferencia mínima entre dos códigos adyacentes, siendo esta diferencia de 1 LSB. De esta manera, la resolución puede expresarse como el porcentaje del valor nominal máximo.
- **No linealidad diferencial (DNL).** Indica la diferencia entre resultado de restar los valores de salida correspondientes a dos códigos adyacentes y el valor esperado correspondiente a 1 LSB [20].

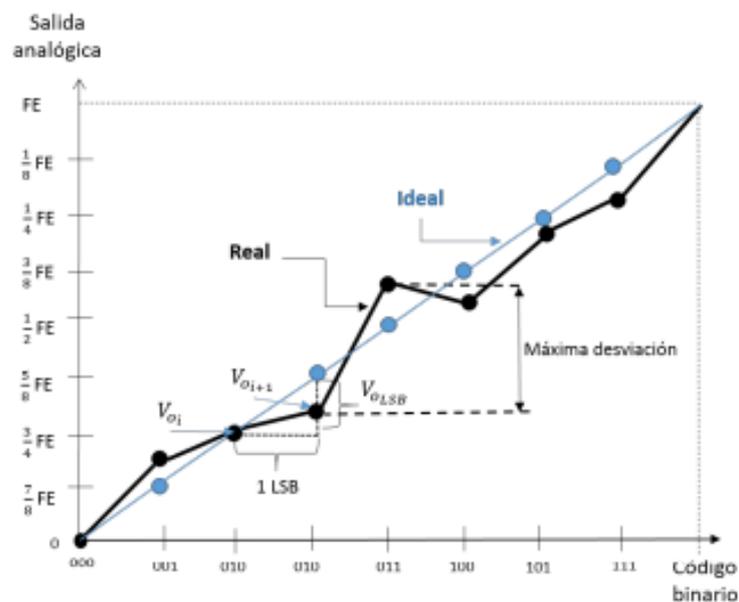


Figura 2.37.- No linealidad diferencial [20].

- **No linealidad integral (INL).** El error de la INL se calcula mediante la máxima no linealidad diferencial.
- **Error de offset.** Se obtiene mediante el valor a la salida obtenido cuando la entrada es nula; también se conoce como el desplazamiento horizontal. Se mide en porcentaje máximo nominal o en LSB.

- **Error de escala.** Al igual que en el caso del ADC, se trata de la desviación máxima a fondo de escala con el valor ideal.

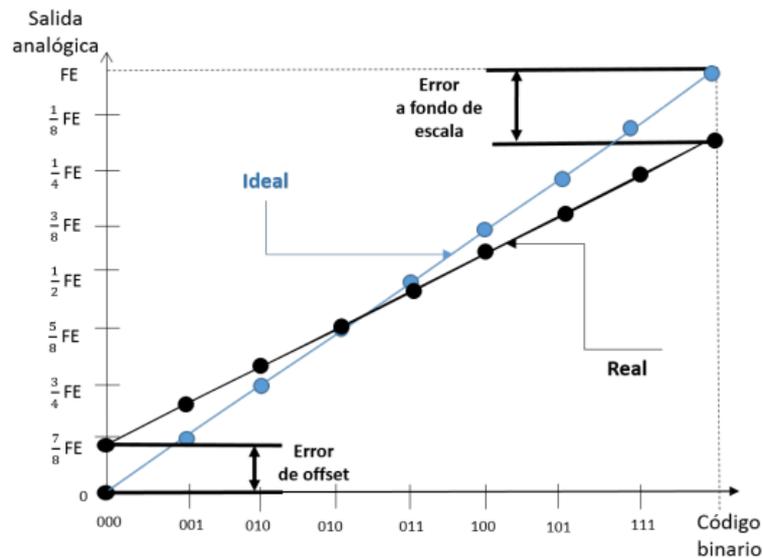


Figura 2.38.- Error de offset y de entrada [20].

#### 2.4.1.2.- Características dinámicas

Estas características se estudian durante el comportamiento del convertidor en régimen permanente. Las más relevantes son:

- **Tiempo de establecimiento (Setting Time, ST).** Se trata del tiempo transcurrido desde que se genera un cambio en la entrada hasta alcanzar el nivel de salida ideal dentro de un margen de error de 0,5 LSB.
- **Rapidez de cambio de salida (Slew Rate, SR).** Indica la pendiente máxima para alcanzar el valor a fondo de escala (FE) partiendo desde el valor de cero [20].
- **Frecuencia de conversión.** Frecuencia máxima a la que el convertidor devuelve la salida sin perder información. Se mide en Hz o en muestras por segundo.

## 2.4.2.- Tipos de conversores DAC.

### 2.4.2.1.- Convertidor con ponderación binaria

Este tipo de convertidores son a nivel de implementación bastante sencillos puesto que su principio de funcionamiento se basa en un amplificador sumador .

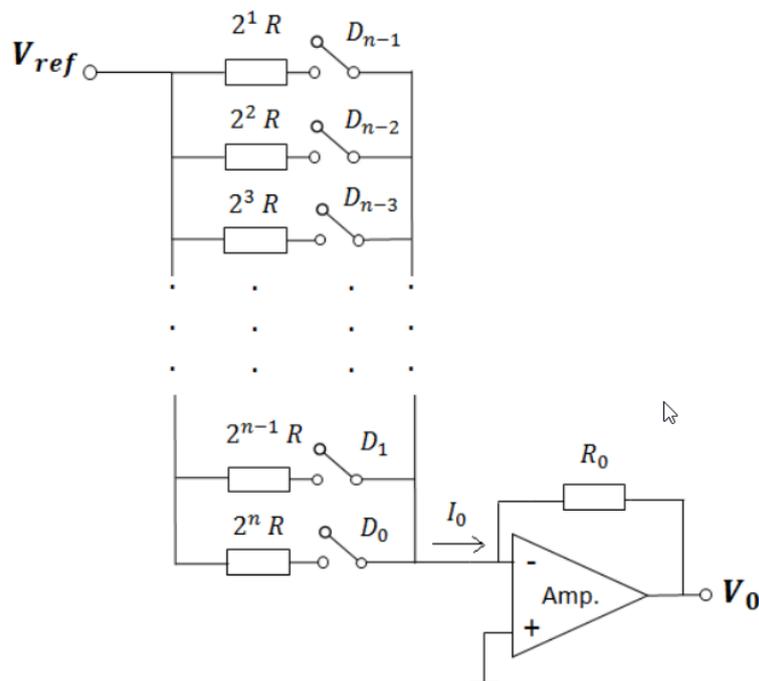


Figura 2.39.- Convertidor DAC con ponderación binaria [20].

Uno de los aspectos negativos de este tipo de convertidor es que está compuesto por un amplio número de resistencias de distintos valores, por lo que es un proceso costoso de realizar.

### 2.4.2.2.- Convertidor con resistencias ponderadas R/2R

Este tipo de convertidor al igual que el de ponderación binaria se basa en un amplificador sumador inversor. La diferencia es la configuración y localización de las distintas resistencias, lo cual hace que esta opción sea más sencilla de implementar.

Gracias a esta variación conseguimos que el convertidor de resistencia ponderadas R/2R no solo sea menos costoso, sino que además alcanza velocidades más altas.

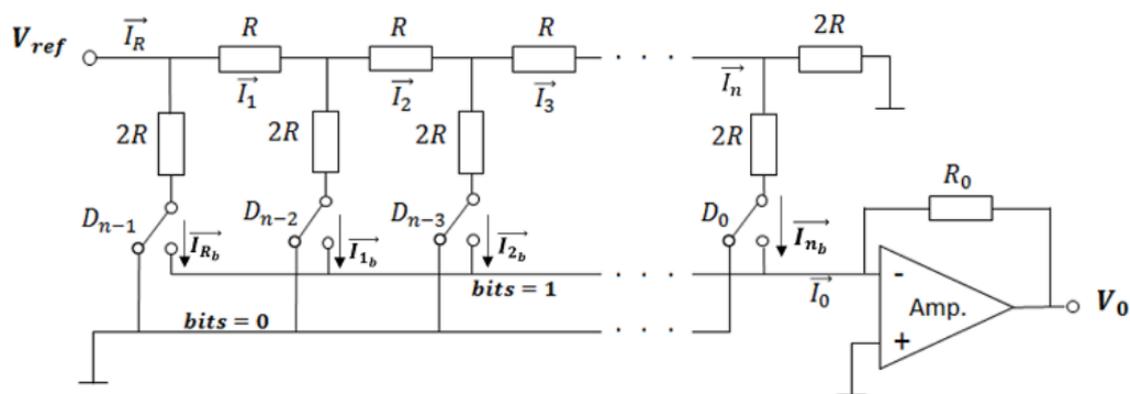


Figura 2.40.- Convertidor DAC con resistencias ponderadas R/2R [20].

### 2.4.2.3.- Convertidor sigma-delta

El convertidor DAC de tipo Sigma-Delta es el más utilizado para el tratamiento de señales de audio digital, puesto que con este tipo de dispositivos se obtiene una alta calidad a un reducido coste. Además de las ventajas que proporciona, son muy sencillos y proporcionan una alta linealidad con resoluciones altas.

En la figura 2.41. [20] podemos ver el diagrama de bloques de un convertidor DAC Sigma-Delta.

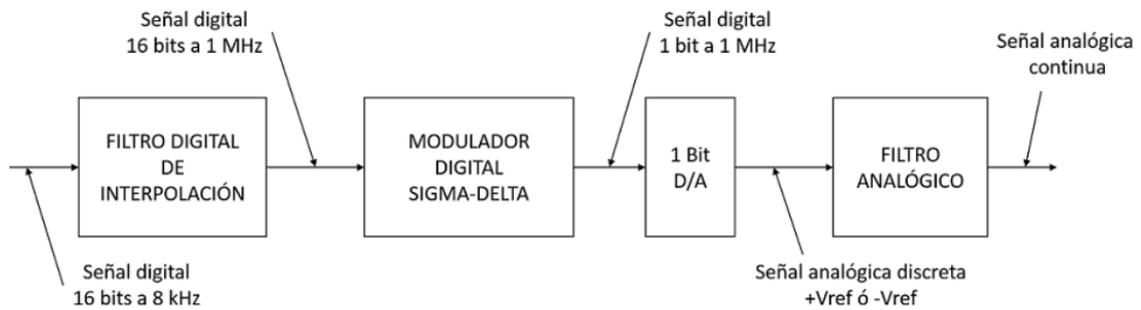


Figura 2.41.- Diagrama de bloques de un DAC Sigma-Delta.

#### 2.4.2.4.- Convertidor PWM (Pulse Width Modulation)

Este convertidor utiliza la misma estructura que el convertidor Sigma-Delta, con la diferencia de que el modulador digital en este caso utiliza una modulación PWM. En la figura 2.42. [28] se muestra la aplicación de la modulación PWM (señal verde) a una señal analógica sinusoidal (señal azul).

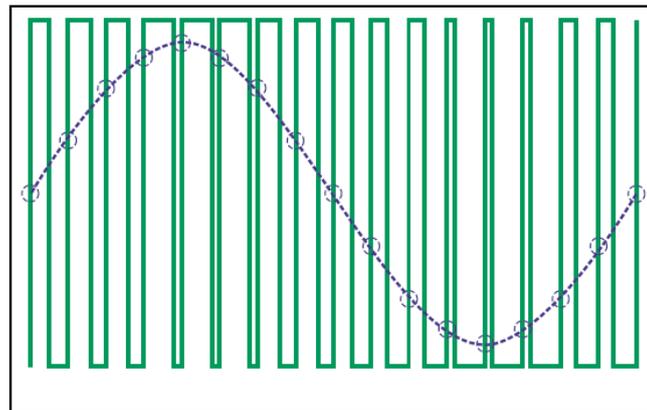


Figura 2.42.- PWM aplicado a una señal sinusoidal.

Esta modulación, también conocida en castellano como modulación por ancho de pulsos, se basa en la variación del ciclo de trabajo de una señal de tipo cuadrada en función del valor de voltaje de entrada. De esta manera, la señal generada será una onda cuadrada cuyos anchos de pulsos van variando.

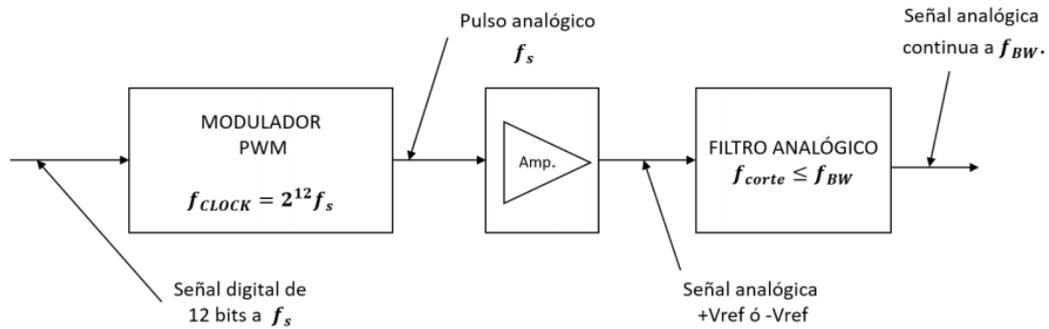


Figura 2.43.- Diagrama de bloques de un convertidor PWM [20].

#### 2.4.2.5.- Comparación de los tipos de conversores

A continuación podemos ver una tabla comparativa de los distintos conversores D/A según las características más importantes [29].

	Ponderación binaria	Resistencias ponderadas	Sigma-Delta	PWM
Complejidad de implementación	Baja	Baja	Media	Media
Velocidad de conversión	Baja	Baja-Media	Baja-Media	Baja
Resolución	Baja	Baja	Alta	Media
Coste	Alto	Baja	Bajo	Medio

Tabla 2.2.- Tabla comparativa de los distintos convertidores DAC.

## **2.5.- MICROCONTROLADORES.**

El Microcontrolador se basa, fundamentalmente, en un microprocesador más algunos periféricos. Se trata de un circuito integrado digital, el cual puede ser usado para un gran número de funciones dado que es programable, y que es el componente principal de un sistema embebido. Se asemeja como una minicomputadora, capaz de controlar elementos de entrada y salida, así como procesar información y automatizar procesos.

Con anterioridad a los microcontroladores los sistemas se utilizaban los autómatas, también llamados máquinas de estado. Estos autómatas funcionan a partir de una tabla de condiciones lógicas que reúnen todas las combinaciones posibles de entrada y los estados de las salidas. Una vez obtenida esta tabla, se diseñaba el sistema digital. Este proceso era muy lento y suponía una gran cantidad de circuitos digitales secuenciales.

Posteriormente, se comenzaron a crear módulos independientes que funcionaban de forma general para múltiples propósitos, como la Unidad Aritmética Lógica (ALU). Sin embargo, este circuito al ser puramente combinacional (sus salidas solo dependen de sus entradas), no es muy útil para ser automatizado. Para esto, se requiere de circuitos secuenciales. Esto dio entrada al diseño de los primeros procesadores [30].

### **2.5.1.- Componentes.**

Un microcontrolador como mínimo contendrá un microprocesador, una memoria y periféricos de entrada y salida. En la figura 2.44. [31] podemos ver la estructura en diagrama de bloques de un microcontrolador.

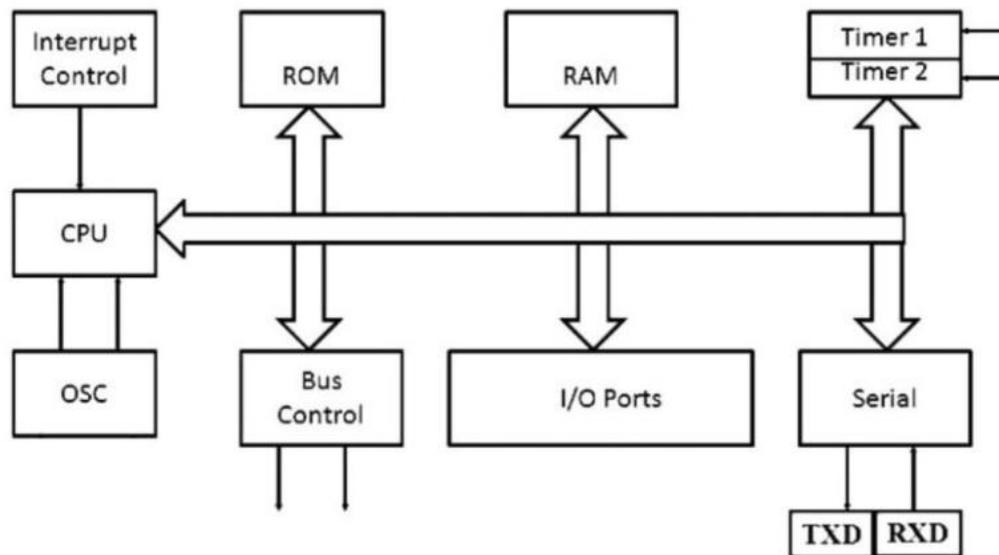


Figura 2.44.- Diagrama de bloques de un microcontrolador.

### 2.5.1.1.- Microprocesador o procesador

Los procesadores deben incluir en sus elementos internos al menos una ALU, una unidad de control y registros. A continuación veremos qué es cada uno de ellos.

- **ALU (Unidad Aritmética y Lógica).** La ALU contiene circuitos electrónicos digitales combinacionales tales como sumadores, multiplicadores, etc. La función principal de esta unidad es realizar operaciones, las cuales pueden dividirse en tres tipos:
  - Lógicas. Son las operaciones realizadas por las puertas lógicas: suma lógica (OR), multiplicación lógica (AND), diferencia lógica (XOR) y negación (NOT). Una operación lógica sólo puede tener como entradas y como salidas una respuesta lógica (0 o 1). Esto se representa con los niveles de voltajes de una señal digital.

- Aritméticas. Estas operaciones son las típicas operaciones de suma, resta, multiplicación y división.
- Misceláneas. Este tipo de operaciones son el resto que no se recogen en los dos grupos anteriores, como por ejemplo la transferencia de bits.
- **Unidad de control.** Esta unidad se encarga de distribuir la lógica de las señales gracias a un conjunto de circuitos digitales secuenciales.
- **Registros.** Las memorias principales de los procesadores se les conoce como registros. Estas memorias funcionan a una mayor velocidad que el resto como la RAM, la FLASH o la CACHÉ puesto que trabajan a la misma velocidad que el procesador.

#### 2.5.1.2.- Periféricos

Los periféricos son los que permiten a los microcontroladores interactuar con el mundo físico. Estos circuitos se encuentran incrustados en el microcontrolador. Estos periféricos permiten realizar distintas funciones, desde controlar los puertos hasta trabajar como convertidores A/D o D/A. Estas funciones se configuran a través de registros de propósito específico [30].

Los periféricos se pueden clasificar según su función en los siguientes tipos:

- **Puertos de entrada/salida paralelos.** Estos puertos tienen capacidades en función del procesador, es decir, si el procesador es de 128 bits el puerto también será de 128 bits.

- **Puertos serie.** Estos periféricos transforman la información digital para poder enviarla por distintas líneas de comunicación. Existen varios tipos, como por ejemplo los puertos serie, I2C, SPI, etc.
  - UART (Unidad de Recepción y Transmisión Serie Asíncrona). Se basa en la transmisión de mensajes binarios a través de dos cables llamados RX (receptor) y TX (transmisor).
  - I2C (Inter-Integrated Circuit). Se conectan dos cables llamados SDA (Serial Data) y SCL (Serial Clock). A través de ellos se envían y reciben datos de varios dispositivos (hasta 127) conectados al mismo bus de datos.
  - SPI (Serial Peripheral Interface). Su función es la de enviar información binaria mediante 4 cables de conexión [30].
- **Periféricos analógicos.** Estos periféricos se comportan como convertidores tanto ADC como DAC.

A continuación podemos ver algunos de los periféricos más típicos y generales:

- **Timer/Contador.** Su función principal es la de contar eventos, ya sea internos (timer) o externos (contador).
- **Controlador de Interrupciones.** Son dispositivos que permiten alterar la secuencia de ejecución del procesador. Esto logra que se pueda interrumpir al procesador para atender el llamado de dispositivos electrónicos de baja velocidad. Por ejemplo, teclados o sensores [30].

- **Controlador de Oscilador.** Permite dividir y/o multiplicar a osciladores internos o externos para poder hacer funcionar a la parte secuencial de la electrónica del microcontrolador [30].

### 2.5.1.3.- Memoria

A diferencia de un SoC (System on a Chip), en el caso del microcontrolador la memoria en la que se ejecutan los programas se encuentra dentro del mismo dispositivo, y no en un chip aparte. La memoria que tienen los microcontroladores es de tipo flash programable. A esta memoria se le accede a través de los pines de entrada y salida del microcontrolador [32].

Una desventaja de este tipo de memoria en vez de una de tipo volátil es que el tiempo de acceso a los datos es mucho más elevado y por lo tanto los microcontroladores funcionan a velocidades más bajas.

La memoria de un microcontrolador puede dividirse en tres bloques:

- **Memoria de programa – FLASH.**

La memoria de programa se necesita cuando estamos programando nuestra aplicación. La información que introducimos en nuestro microcontrolador se convierte en instrucciones máquina que requieren un tamaño de bits para ser guardadas. Estos bits o instrucciones se guardan en la FLASH [30]. Este tipo de memorias permite la lectura y escritura de varios datos a una elevada velocidad, que posteriormente pueden ser borrados, mediante impulsos eléctricos [33].

- **Memoria de datos – RAM (Random Access Memory).**

Esta memoria se utiliza para recopilar información que necesitamos más rápido, como por ejemplo las variables que se declaran inicialmente puesto que se necesitarán para un proceso o cálculo posterior. Típicamente se utiliza una memoria RAM [30]. Las memorias RAM son memorias temporales o volátiles, que recaudan

la información de los programas con los que se está trabajando en ese mismo instante [34].

- **Memoria externa – EEPROM (Electrically Erasable Programmable Read-Only Memory).**

Aunque también se pueden emplear memorias FLASH, la EEPROM se suele seleccionar para este tipo de memoria que necesitamos para ir guardando la información que estamos procesando. Las FLASH suelen funcionar a una velocidad mucho más reducida para protocolos serie como I2C, Serial o SPI, por eso se suele optar más por una EEPROM [30]. Las EEPROM son memorias de solo lectura que pueden borrar su información mediante impulsos eléctricos.

### 2.5.2.- Tipos de arquitectura.

En un microcontrolador podemos definir su arquitectura en función de dos criterios: la organización de las memorias o el ancho de su bus de datos y/o memoria. Las principales son las siguientes:

En función de la memoria, podemos diferenciar dos arquitecturas: Harvard y Von Neumann.

- **Arquitectura Von Neumann.** Cronológicamente hablando, la primera arquitectura fue la Von Neumann; en esta organización, el bus de datos y el bus de memorias son el mismo. Con esta organización se obtiene un ahorro de líneas pero el proceso de ejecución de una instrucción se ralentiza.
- **Arquitectura Harvard.** Esta arquitectura tiene una memoria específica para datos y otra para instrucciones, por lo que se usan dos buses diferentes. Esto aporta una mayor velocidad de ejecución. Es la más utilizada actualmente.

Por otro lado, la segunda clasificación de la arquitectura de un microcontrolador es por el ancho o tamaño en bits de sus buses de datos y/o memoria. Por ejemplo, una arquitectura de 64 bits puede significar que el procesador puede hacer operaciones con operadores o variables de un tamaño de 64 bits e incluso que puede direccionar o guardar hasta 264

localidades de memoria tanto de programa como de datos. Entonces por lo general, un procesador de más bits puede hacer operaciones matemáticas complejas en un menor tiempo [30].

### **2.5.3.- Programación de un microcontrolador.**

Como el hardware ya viene integrado en un solo chip, para usar un microcontrolador se debe especificar su funcionamiento por software a través de programas que indiquen las instrucciones que el microcontrolador debe realizar. En una memoria se guardan los programas y un elemento llamado CPU se encarga de procesar paso por paso las instrucciones del programa [35].

Los microcontroladores necesitan ser programados en lenguaje ensamblador puesto que es el “idioma” que requieren. Esto significa que cada microcontrolador es capaz de entender un conjunto de instrucciones, las cuales pueden clasificarse en función de las operaciones que desempeñan:

- Aritméticas.
- Lógicas.
- Transferencia de bits.
- Misceláneas.

El lenguaje ensamblador presenta una gran curva de aprendizaje, puesto que no es tan intuitivo, y supone una limitación a la hora de desarrollar las aplicaciones de sistemas embebidos. Para ello lo que se utiliza son lenguajes a nivel superior como, por ejemplo, C/C++.

No obstante, el microcontrolador sigue necesitando de un lenguaje ensamblador, lo que significa que debemos traducir nuestro código. Para ello se emplean compiladores.

Para grabar un programa en un microcontrolador se requieren tres elementos:

- **Compilador.** Como ya hemos mencionado, es el encargado de traducir el lenguaje como C/C++ en lenguaje ensamblador.
- **IDE (Integrated Development Environment).** Es el entorno de desarrollo que se emplea para crear y compilar los programas. Uno muy conocido es el Arduino IDE.
- **Programador.** Depende del fabricante o la arquitectura del microcontrolador. No obstante existen algunos conocidos como Universales que se puede emplear para programar diferentes arquitecturas.

#### 2.5.4.- Aplicaciones de uso.

Son múltiples las aplicaciones que tienen los microcontroladores en muchos sistemas digitales que nos rodean en el día a día. A continuación se mencionan algunos ejemplos:

- Alarmas
- Máquinas distribuidoras
- Sistemas de control
- Cargadores
- Dispositivos biomédicos
- Sistemas de medición e instrumentación
- Lectores de huella
- Cámaras

### 2.5.5.- Diferencias entre un microcontrolador y un microprocesador.

Lo que diferencia a un microcontrolador de un microprocesador es su estructura y los elementos que lo componen. Para más exactitud, un microcontrolador es un microprocesador junto a unos periféricos.

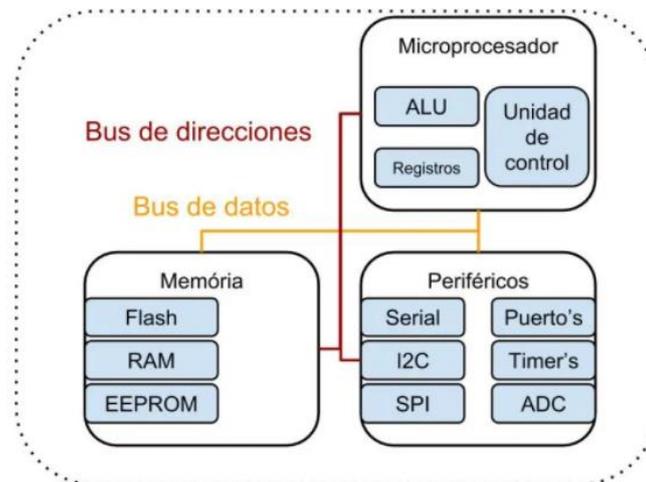


Figura 2.45.- Diagrama de bloques de un microcontrolador [30].

## 2.6.- PROTOCOLOS DE COMUNICACIÓN.

Para poder transferir datos entre dos ordenadores se necesitan unos protocolos de comunicación. Estos protocolos sirven para conectarse y comunicarse con dispositivos periféricos [36].

Dentro de los métodos para la transmisión de datos existen dos tipos:

- **Transmisión serie.** En este caso la información se envía de bit en bit, de manera bidireccional de un host a otro [36].

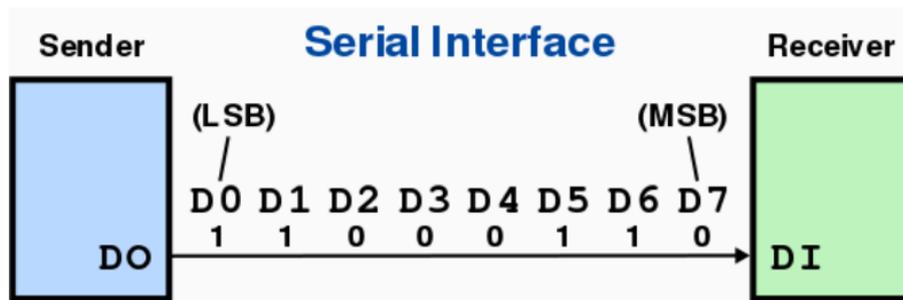


Figura 2.46.- Transmisión de datos en serie [36].

La transmisión en Serie tiene dos subclases: síncrona y asíncrona. En la primera, se añade un bit adicional a cada byte para que el receptor sepa de la llegada de nuevos datos. En la segunda, no se añade este bit y los datos son transferidos en forma de tramas de múltiples bytes [36].

- **Transmisión paralelo.** La información se transmite mediante varios bits de forma simultánea con un solo pulso de reloj [36].

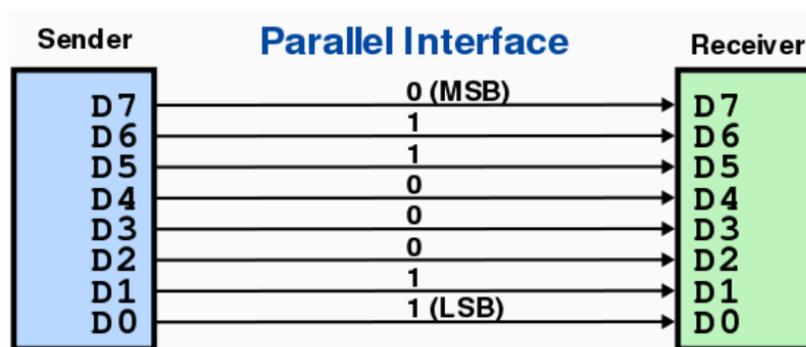


Figura 2.47.- Transmisión de datos en paralelo [36].

La transmisión paralela utiliza un **puerto de 25 pines** que tiene 17 líneas de señal y 8 líneas de tierra. Las 17 líneas de señal se dividen a su vez en [36]:

- 4 para iniciar la comunicación.
- 5 líneas de estado para comunicar y notificar errores.
- 8 para transferir datos.

El bus serie es el más utilizado en sistemas de control y periféricos. Por ello es importante mencionar en detalle algunos de los protocolos más conocidos:

### 2.6.1.- SPI (Serial Peripheral Interface).

El protocolo SPI es dentro de los protocolos seriales el de más alta velocidad. En su versión estándar utiliza 4 pines: uno para enviar datos (MOSI, Master Out Slave In), otro para recibirlos (MISO, Master In Slave Out), otro de reloj (SCK, Serial Clock) y uno de habilitación (CS, Chip Select) [37].

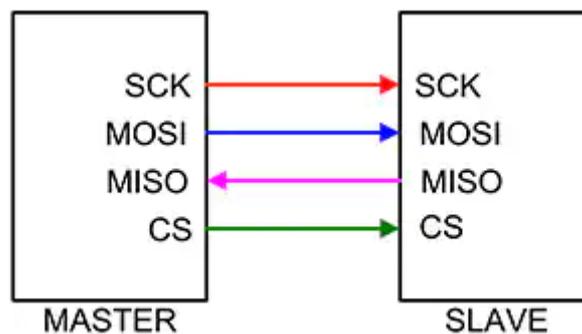


Figura 2.48.- Estructura Maestro-Esclavo protocolo SPI [38].

Usa un modelo de maestro-secundario con un maestro simple y puede manejar varios dispositivos secundarios usando comunicaciones dúplex que operan a velocidades de reloj de hasta 50 MHz [38].

### 2.6.2.- I2C (Inter-Integrated Circuit).

Este protocolo es el único de los seriales que está diseñado para conectar múltiples dispositivos al mismo bus de datos. Esto supone una mayor complejidad de hardware y software. Solo requiere, por otro lado, dos pines: uno de datos (SDA, Serial Data) y un pin de reloj (SCL, Serial Clock) [37].

El protocolo permite conectar hasta 127 dispositivos esclavos con esas dos líneas, con hasta velocidades de 100, 400 y 1000 kbits/s [39].

Dentro del protocolo se distinguen dos elementos básicos: un esclavo y un maestro. El maestro controla el cable de reloj y se encarga de iniciar y parar la comunicación. El esclavo I2C suministra la información de interés al maestro [39].

## **2.7.- LENGUAJES DE PROGRAMACIÓN.**

A continuación mencionaremos algunos lenguajes de programación que posteriormente utilizaremos para el desarrollo de este proyecto. Esto se debe a que necesitaremos una interfaz de usuario en la que mostrar los datos que queremos leer, implementar funciones para poder interactuar como pueden ser botones, sliders, entradas de texto, etc.

### **2.7.1.- HTML**

El Lenguaje de Marcas de Hipertexto, más conocido como HTML (HyperText Markup Language), es el principal componente de la mayoría de las páginas web. Con esta estructura de lenguaje podemos definir el significado y la estructuración del contenido web. HTML suele venir acompañado de otras tecnologías como puede ser CSS (para mejorar su apariencia) o JavaScript (para aportar funcionalidad) [40].

Es un estándar de la World Wide Web Consortium o Consorcio WWW, que se encarga de estandarizar la mayoría de las tecnologías ligadas a las aplicaciones web. Fue creado por Tim Berners-Lee en 1991.

El hipertexto se refiere a los enlaces que se usan para conectar páginas y textos entre sí. El lenguaje de marcado, por su parte, son las marcas que indican a los servidores web la estructura y estilo del contenido.

HTML no es considerado un lenguaje de programación puesto que no genera funcionalidades dinámicas. De hecho, como hemos mencionado anteriormente, utiliza una tecnología externa para ello (JavaScript).

### 2.7.1.1.- Elementos

Los elementos en HTML se distinguen del resto mediante ‘etiquetas’, que se basan en una estructura de tipo “<nombre>”. Podemos clasificar los elementos de un código HTML en dos tipos: elementos de bloque y elementos en línea.

#### ➤ Elementos de bloque

Un elemento de bloque ocupa todo el ancho de la página. Siempre comienza una nueva línea en el documento. Algunos ejemplos son <html>, el elemento raíz; <head>, que contiene información como el título; y <body>, que encierra el resto de contenido [41].

```
<html>
<head>
<!-- INFORMACION META-->
</head>
<body>
<!-- CONTENIDO DE LA PAGINA -->
</body>
</html>
```

Otros tipos de elementos de bloque pueden ser:

- **Etiquetas de encabezados.** <h1>, <h2>, etc.
- **Etiquetas de párrafos.** <p>
- **Etiquetas de listas.** <ol>, <ul>, <li>, etc.

### ➤ Elementos en línea

Estas etiquetas tienen como función dar formato al contenido interno al que nos referimos dentro de un elemento de bloque. Un buen ejemplo sería las etiquetas que se utilizan para añadir links, vídeos, imágenes, o cambiar el formato de la fuente.

#### 2.7.1.2.- Aplicaciones

Como hemos mencionado, HTML conforma la mayoría de las páginas web conocidas, por lo que es una tecnología ampliamente usada en todo el mundo. Alguno de los usos más comunes de HTML son los siguientes:

- **Desarrollo web.** Diseño de páginas web que incluyen textos, hipervínculos, archivos multimedia, etc.
- **Navegación por internet.** Enlazar entre sí páginas web gracias a los hipervínculos.
- **Documentación web.** Proporciona herramientas para organizar y dar formato a documentos.

#### 2.7.1.3.- Ventajas y desventajas

Entre las ventajas podemos encontrar que:

- ✓ Es un código limpio e intuitivo, con una curva de aprendizaje pequeña.
- ✓ Hay muchos recursos e información.
- ✓ Es de código abierto, gratuito y por lo tanto accesible.
- ✓ Es compatible con lenguajes de backend tales como PHP y Node.js.

Por otro lado, existen algunos inconvenientes:

- ✗ No es dinámico.
- ✗ Es independiente, se tienen que crear varias páginas individuales.
- ✗ Algunos navegadores no son del todo compatibles con el código.

Actualmente se ha desarrollado HTML5, el cual incluye nuevas etiquetas y atributos, así como nuevos tipos de controles de formularios.

### **2.7.2.- JAVASCRIPT**

JavaScript (JS) es un lenguaje de programación ligero, interpretado, o compilado justo-a-tiempo (just-in-time) con funciones de primera clase [42]. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico [43]. Se utiliza principalmente del lado del cliente, implementado como parte de un navegador web, otorgando funcionalidad.

La historia de JavaScript comienza en los años 90 de la mano de Brendan Eich, dándole el nombre inicial de LiveScript. Nació de la necesidad de minimizar la pérdida de información a la hora de conseguir una validación en el navegador, puesto que los mensajes de error tardaban mucho en llegar al lado del cliente.

Este lenguaje de programación es el único que es leído por los navegadores de forma nativa, es decir, sin necesidad de terceros.

El estándar para este lenguaje es ECMAScript, el cual se empezó a desarrollar en 1997, surgiendo de la necesidad de estandarizar el lenguaje. Sin embargo, para hacer esta tarea de compatibilidad más sencilla surgieron librerías como jQuery y otras muy semejantes. La librería jQuery permitía programar JavaScript de forma sencilla y compatible para todos los navegadores [44].

Posteriormente los navegadores fueron adaptando el estándar y finalmente JavaScript se convirtió en un lenguaje compatible con todos los navegadores, sin la necesidad de utilizar las librerías de jQuery.

### 2.7.2.1.- Características

#### ➤ Lenguaje del lado del cliente

Como hemos mencionado anteriormente, JS se define como el lenguaje que se ejecuta al lado del cliente, a diferencia de otros lenguajes como PHP que se ejecuta en el lado del Servidor.

#### ➤ Lenguaje orientado a objetos

El lenguaje de JavaScript se define como orientado a objetos porque utiliza estructuras de clases y objetos. Esta organización permite desarrollar el código de forma más simple y permite la reutilización de líneas de código.

#### ➤ De tipado débil (o no tipado)

El lenguaje de tipado débil se define como aquel que no necesita que se especifique el tipo de dato al crear una variable nueva. Supone una gran ventaja a la hora de programar en cuanto a velocidad.

#### ➤ De alto nivel

Como hemos mencionado anteriormente en el apartado de los microprocesadores, estos necesitan un lenguaje de tipo ensamblador (bajo nivel). Sin embargo, para nosotros es más intuitivo un lenguaje de más alto nivel, que indica que se encuentra lo más alejado de la máquina. Este es el caso de JavaScript.

### 2.7.3.- CSS

Hojas de Estilo en Cascada (CSS, Cascading Style Sheets) es el lenguaje de estilos utilizado para describir la presentación de documentos HTML o XML [45]. Se sigue desarrollando por el World Wide Web Consortium.

Se desarrolló en los años 90, y a diferencia de otras hojas de estilo ya existentes en aquel momento, CSS permitía usar una sola hoja de estilos para distintas páginas web independientes.

Mientras que JavaScript es el lenguaje que da dinamismo y funcionalidad a una página de HTML, CSS se encarga de definir el diseño. El navegador interpreta las líneas de código de forma independiente al código HTML y se presenta de forma más atractiva al cliente.

Con la ayuda de unas sencillas instrucciones, los elementos del sitio web como el diseño, el color y la tipografía, pueden adaptarse como se desee. Gracias a las hojas de estilo en cascada, la estructura semántica y el contenido del documento no se ven afectados [46].

#### 2.7.3.1.- Tipos de integración

A la hora de integrar CSS en un código HTML, existe distintas formas: hojas de estilo externas, hojas de estilo internas y estilo inline.

##### ➤ Hojas de estilo externas

En este caso las hojas de CSS son un archivo externo e independiente al de la página HTML. Para poder hacer referencia a él se le añade al archivo HTML mediante una etiqueta de tipo 'link', dentro de la zona de cabecera.

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="stylesheet.css">
</head>
<body>
<h1> Esto es un encabezado </h1>
<p>Esto es un párrafo</p>
</body>
</html>
```

Este es el estilo más utilizado cuando se utilizan códigos muy amplios y se van a reutilizar para distintas páginas. Es el método más usado.

### ➤ Hojas de estilo internas

En algunas ocasiones puede ser necesario o más sencillo añadir el código CSS directamente en las líneas de la página HTML.

```
<!DOCTYPE html>
<html>
<head>
<style>
h1 {color: blue; font-size: 12px;}
</style>
</head>
<body>
<h1>Esto es un encabezado</h1>
<p>Esto es un párrafo</p>
</body>
</html>
```

Con este método se simplifica el número de archivos necesario pero solo será más efectivo si el código CSS es pequeño y si no se va a reutilizar, puesto que en este caso solo valdría para la página HTML en la que se está definiendo.

### ➤ Estilo inline

En este caso, las instrucciones de la hoja de estilos se declaran en el mismo archivo HTML al igual que en el caso anterior. En este caso, a diferencia de los de estilo interno, se definen

dentro de cada atributo, lo que los hace aún más independientes y sin posibilidad de reutilizarse.

```
<!DOCTYPE html>  
<html>  
<body>  
<h1 style="color: blue; font-size:12px;"> Esto es un encabezado </h1>  
<p> Esto es un párrafo </p>  
</body>  
</html>
```

Este tipo de integración es ventajosa cuando no se quiere dar muchas instrucciones de diseño.

## 2.8.- COMUNICACIONES.

### 2.8.1.- Ajax

AJAX, acrónimo de Asynchronous JavaScript And XML (JavaScript asíncrono y XML), es una técnica de desarrollo web para crear aplicaciones interactivas o RIA (Rich Internet Applications) [47].

Para poder entender el funcionamiento de la comunicación con AJAX, debemos saber que la web funciona bajo el protocolo HTTP. Este protocolo es el conjunto de reglas que deben seguir cliente/servidor para comunicarse efectivamente. Las peticiones HTTP no tienen información de estado, por eso se manda desde “frontend” una información de sesión (cookies), y un proxy permite crear cachés y balancear las cargas.

También cuenta con un lenguaje de programación interpretado orientado a objetos, como lo es “JavaScript”, del cual hablaremos más adelante.

Además, la web cuenta con el objeto creado por Microsoft, XMLHttpRequest o XHR, cuya función es poder hacer peticiones solo a ciertas partes de la página sin necesidad de recargarlas, estas consultas retornaban como código XML.

Por lo tanto podemos definir AJAX como un modo de programación que no bloquea la ejecución de otros procesos mientras se resuelven distintas peticiones [48]. Esta técnica de desarrollo web consigue realizar dichas acciones de manera asíncrona, por lo que podemos obtener cambios en la página sin necesidad de refrescar. Esto consiguió proporcionar una navegación más ágil y rápida al consumidor [49].

#### 2.8.1.1.- Funcionamiento

AJAX se comprende básicamente de las siguientes tecnologías:

- **HTML/XHTML** para el lenguaje principal y **CSS** para la presentación.
- **El Modelo de objetos del documento (DOM)** para datos de visualización dinámicos y su interacción.
- **XML** para el intercambio de datos y **XSLT** para su manipulación. Muchos desarrolladores han comenzado a reemplazarlo por **JSON** porque es más similar a **JavaScript** en su forma.
- El objeto **XMLHttpRequest** para la comunicación asíncrona.
- Finalmente, el lenguaje de programación **JavaScript** para unir todas estas tecnologías [50].

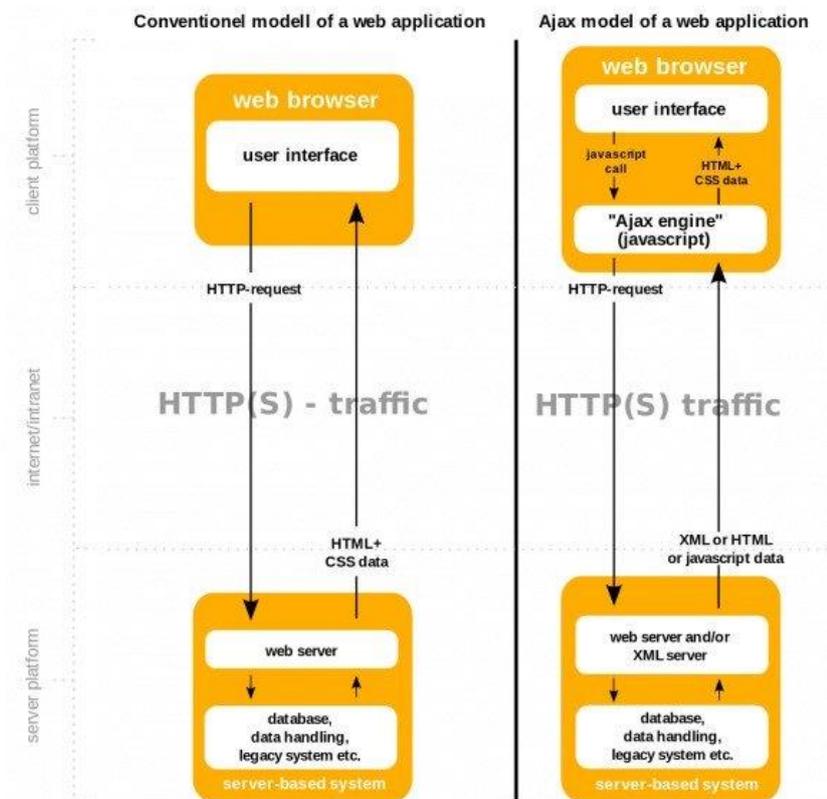


Figura 2.49.- Diagrama de funcionamiento del modelo AJAX [50].

- Modelo convencional. En este modelo se envía una solicitud HTTP desde el cliente al servidor. Este último recibe los datos desde el cliente y devuelve los solicitados. El navegador recibe esta información y recarga la página para que sea visualizada.
- El navegador crea una llamada JavaScript que activará XMLHttpRequest. Por detrás el cliente crea una solicitud HTTP al servidor. El servidor recibe, recupera y posteriormente envía los datos correspondientes al navegador. Finalmente el navegador mostrará directamente la información de la página, sin necesidad de recargar [50].

## 2.8.2.- WebSockets

Los WebSockets sirven para realizar conexiones de manera persistente entre el navegador del usuario y el servidor. Los WebSockets comparten datos rápidamente y solo necesitan crear la conexión una vez. Por lo tanto, son particularmente útiles si tenemos que obtener datos de manera constante, ya que así evitamos que el servidor tenga que procesar un gran número de peticiones innecesarias [51].

Esta tecnología utiliza los puertos 80 y 443, y fue diseñada para evitar los problemas generados por proxies y firewalls. La manera para poder utilizar HTTP con WebSockets, aunque empleen el mismo puerto, usan un protocolo de comunicación distinto y por lo tanto se utiliza ws:// en la URL. Similar pasa con el puerto de HTTPS.

### 2.8.2.1.- Antecedentes en la comunicación cliente-servidor

Inicialmente solo se podía acceder a la información de forma estática, por lo que no podía navegarse de una página a otra. Con el paso del tiempo se fueron desarrollando sitios web más dinámicos con los que se podía ir cargando información, pero de manera ineficiente puesto que daba lugar a un gran número de interacciones.

La primera solución que se desarrolla es el XMLHttpRequest, una función JavaScript creada por Microsoft. Con este método se puede invocar una URL y obtener información igual que al recargar la página. Este método fue implementado por un gran número de navegadores puesto que supuso un gran avance en Internet. Gracias a esta innovación se generaron nuevas aplicaciones webs, lo que se conoce como AJAX (Asynchronous JavaScript and XML), donde XML hace referencia a XMLHttpRequest.

Sin embargo, esta solución presenta un gran número de limitaciones y desventajas, siendo el principal problema la interacción. Para generar esas ‘recargas’, el cliente debe hacer una petición ya sea mediante un temporizador o una interfaz gráfica como un botón. Por lo tanto, no se pueden enviar eventos desde páginas distintas conectadas a nuestro servidor. Otro

problema es que se genera un gran número de peticiones innecesarias que llevan una gran carga en infraestructura.

La solución que se desarrolló fueron los WebSockets, los cuales permiten una relación de tipo bidireccional entre el servidor y el cliente.

### 2.8.2.2.- Funcionamiento

La conexión entre el navegador y el servidor se mantiene activa en todo momento siempre y cuando la pestaña permanezca abierta. Los elementos necesarios para mantener la conexión activa y bidireccional del cliente y el servidor son los siguientes:

- **WebSockets en el lado del cliente.** Mediante este WebSockets se inicia la conexión con el servidor.
- **WebSockets en el lado del servidor.** Acepta esa conexión y da lugar al intercambio de datos.

### 2.8.3.- Comparativa.

En primer lugar es importante remarcar que Ajax no pretende sustituir a los WebSockets ni viceversa. Ambos son dos métodos de comunicación los cuales proporcionan ciertas ventajas y desventajas y por lo tanto son más o menos beneficiosos según la aplicación y el resultado que se desee.

- AJAX utiliza el protocolo HTTP y puede realizar peticiones utilizando los métodos POST/GET desde el cliente al servidor. Por su lado, WebSockets proporciona un protocolo bidireccional entre el cliente y el servidor, distinto de HTTP. En nuestro caso necesitaremos realizar un amplio número de peticiones desde el cliente al servidor, y no viceversa. En el caso de necesitar alguna de las características específicas que nos proporcionan los WebSockets sería recomendable, pero en este caso no es necesario y existe un mayor número de herramientas disponibles en Internet con relación a Ajax.

- Cuando se realiza una petición desde Ajax, el servidor responde y la comunicación acaba. Utilizando WebSockets una vez se inicia la comunicación se mantiene activa. Como hemos mencionado necesitaremos realizar varias peticiones de cliente a servidor para poder muestrear muchos datos por segundo, por lo que necesitaremos que el resto de volumen de peticiones sea inferior para no afectar al propósito principal [52].

Finalmente y como hemos mencionado para la aplicación que vamos a desarrollar es más conveniente el uso de una comunicación a través de Ajax puesto que la diferencia que nos puede aportar el uso de WebSockets salen del alcance de nuestro proyecto.

## 2.9.- COMUNICACIONES INALÁMBRICAS

Las comunicaciones inalámbricas son el tipo de comunicación en el que no se utiliza un medio de propagación físico. En esta comunicación se emplea la modulación de las ondas electromagnéticas. Los únicos dispositivos físicos se encuentran los emisores y los receptores [53].

### 2.9.1.- Clasificación de redes inalámbricas.

Las comunicaciones inalámbricas se pueden clasificar según su alcance [54]:

- Redes inalámbricas de área corporal o WBAN (Wireless Body Area Network) cubren distancias de 1 ó 2 metros.
- Redes inalámbricas de área personal o WPAN (Wireless Personal Area Network). Se utilizan para interconectar los distintos dispositivos de un usuario

(por ejemplo, el ordenador con la impresora). Éste es el caso de la tecnología Bluetooth.

- Redes inalámbricas de área local o WLAN (Wireless Local Area Network) cubren distancias de unos cientos de metros. Crean un entorno de red local entre ordenadores situados en un mismo edificio o grupos de edificios. Éste es el caso de Wi-Fi.
- Redes inalámbricas de área metropolitana o WMAN (Wireless Metropolitan Area Network) pretenden cubrir el área de una ciudad o entorno metropolitano.
- Redes inalámbricas de área extensa o WWAN (Wireless Wide Area Network), pueden cubrir toda una región (país o grupo de países). Estas redes se basan en tecnología celular, como el caso de las redes de telefonía móvil conocidas como 2,5G o 3G.
- Otras: Redes inalámbricas de área regional (WRAN), están constituidas por una integración de las redes WLAN y WMAN, bajo el estándar IEEE 802.22.

### **2.9.2.- Tipos de redes inalámbricas.**

Existen distintos tipos de conexiones inalámbricas, siendo los más importantes y conocidas el Bluetooth y el Wi-Fi, los cuales describiremos a continuación.

- Bluetooth. Esta tecnología es un enlace radio de corto alcance [54]. Permite la conectividad personal entre cualquier tipo de dispositivo que cumpla con las especificaciones inalámbricas Bluetooth.

Las conexiones son uno a uno, con un rango máximo de 10 metros, aunque mediante el uso de repetidores se puede lograr un alcance de hasta 100 metros con algo de distorsión [53].

- Wi-Fi (Familia IEEE 802.11). Esta tecnología es el sistema más extendido, llegando a ser un estándar de facto [54]. Wi-Fi es un sistema que permite crear redes inalámbricas, también conocidas como Wireless [53]. Permite distancias de 100 metros sin distorsión y una velocidad mucho más elevada.

En nuestro proyecto utilizaremos el Wi-Fi como la red de comunicación puesto que nos aporta una mayor velocidad de transmisión de datos, lo cual resulta muy importante puesto que necesitaremos que los datos medidos por el osciloscopio se envíen a la mayor velocidad posible al servidor [55].

Por otro lado, Wi-Fi nos permite establecer una red de dispositivos conectados entre sí simultáneamente. Esto significa que nos podrá aportar la capacidad de que varios usuarios se conecten a la vez a nuestro servidor, a diferencia de Bluetooth que solo nos permitiría de uno en uno [55].

## **2.10.- PCB.**

Un circuito impreso o PCB (Printed Circuit Board) es una tarjeta o placa que se utiliza para realizar el emplazamiento de los distintos elementos que conforman un circuito y las interconexiones eléctricas entre ellas [56]. Estos componentes pueden ser chips condensadores, diodos, resistencias, etc. La primera PCB fue diseñada por el ingeniero Paul Eisler en 1936 [57].

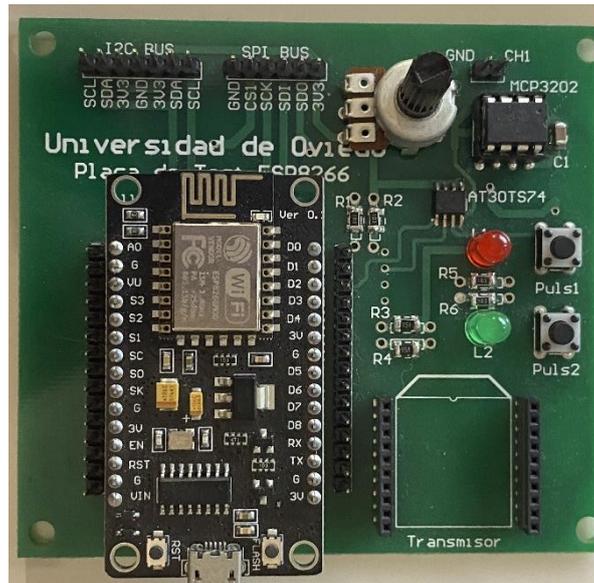


Figura 2.50.- PCB.

### 2.10.1.- Diseño.

Los circuitos impresos se forman a partir de una serie de capas conductoras, dependiendo de la complejidad del circuito. Estas capas se separan mediante un material aislante conocido como sustrato, el cual puede ser de distintas composiciones.

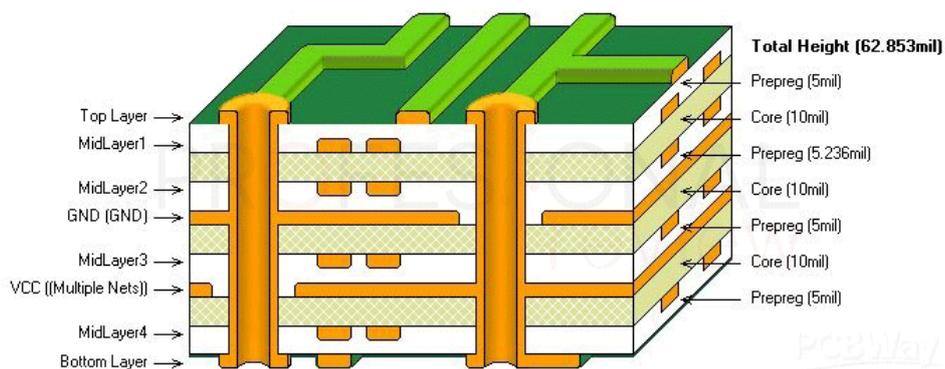


Figura 2.51.- Capas de una PCB [57].

Para conectar entre sí los distintos componentes de una PCB se utilizan pistas conductoras de cobre, típicamente muy finas. Estas pistas, que se pueden encontrar en las distintas capas

de la PCB, se conectan entre si mediante unos orificios llamados vías. Las vías pueden atravesar completamente de lado a lado la PCB o solo de forma parcial.

### 2.10.2.- Proceso.

1. El diseño de los circuitos impresos puede ser realizado mediante software como por ejemplo Eagle, Altium y Proteus, entre otras. Una vez diseñada se pasa a la creación física de la PCB, proceso llamado trazado fotográfico donde se graba mediante láser un gráfico con las máscaras de conexiones de los elementos electrónicos.
2. Se imprimen en la PCB las pistas eléctricas internas. Este paso se realiza si la PCB dispone de capas internas con pistas conductoras.
3. Se comprueba el correcto funcionamiento de las pistas de forma automatizada. A este paso se le conoce como inspección y verificación (AOI).
4. A continuación se realiza un tratamiento de óxido a las distintas láminas para mejorar capacidades y durabilidad. Lo siguiente será construir la PCB definitiva uniendo cada una de las capas.
5. En caso de tener que unir las capas y las pistas de cobre se necesitará realizar una serie de orificios con la ayuda de un taladro.

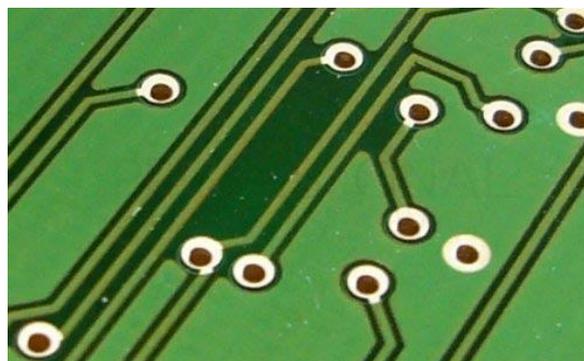


Figura 2.52.- Orificios de la PCB [57]..

6. Para que estos orificios tengan conexión se deben metalizar mediante un proceso de chapado con una fina película de cobre en el interior de los agujeros.
7. Se procede a crear las pistas conductoras, que siguen el mismo procedimiento que el indicado en el paso 3 para las pistas internas. A continuación la PCB se someterá a un proceso de galvanoplastia y finalmente se le añadirá otra capa de estaño encima de la de cobre.
8. Strip each strip es el penúltimo paso, en el cual se elimina el cobre sobrante de la PCB. Posterior a este paso, se retira también el estaño mediante un tratamiento químico.
9. Finalmente se pasa a aplicar una máscara de soldadura para la poder soldar los componentes.
10. Una vez ya se han realizado todos los pasos anteriores, se pasa a añadir los componentes [57].

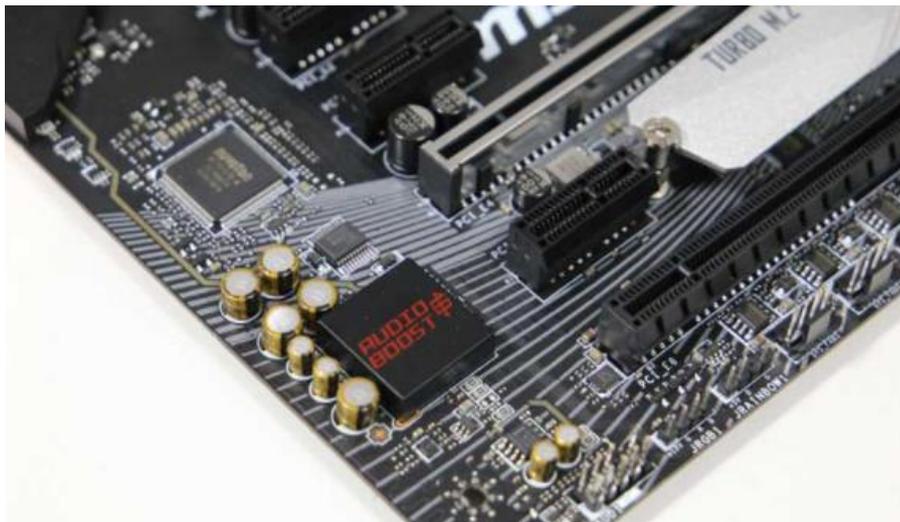


Figura 2.53.- PCB con componentes [57].

### **2.10.3.- Aplicaciones.**

Actualmente la mayoría de los productos están compuestos por tarjetas electrónicas de distintas formas, características, tamaños, etc. Los circuitos integrados son muy usados cuando se trabaja en diseño de electrónica, mecatrónica, eléctrica o incluso en investigación, producción y manufactura, etc [58].

Antes de la aparición de las PCB y la extensión de su uso, los componentes electrónicos estaban repletos de cables, circuitos integrados y resistencias que se repartían por todo el chasis.

Las PCB no son sólo utilizadas para la creación de las placas base, si bien son el mejor ejemplo que hemos podido utilizar para ver toda la utilidad que tienen. También son fundamentales en dispositivos electrónicos como los smartphones, en estos dispositivos el proceso se ha llevado aún más allá y son PCB «de doble cara» dónde en cada cara de la placa hay unos componentes totalmente diferentes. Esto consigue reducir en gran medida el espacio utilizado y poder hacer, así, dispositivos más pequeños [59].

### 3. Objetivos.

El diseño de este proyecto tiene como objetivo principal el diseño y desarrollo de un osciloscopio al que podemos acceder desde cualquier dispositivo con navegador web.

Esta página web tendrá como objetivo principal la compatibilidad con distintos clientes web. Puesto que la primera característica que buscamos es la accesibilidad, será necesario que nuestra página se adapte de forma correcta a cualquier dispositivo y navegador.

El mini-osciloscopio tendrá, por un lado, entradas de señal que servirán para medir tanto señales de voltaje como de corriente. En cuanto a las características técnicas se busca la posibilidad de obtener un ancho de banda de medición de, al menos, 10kHz. El valor de voltaje máximo que podamos recibir rondará los 25V.

Por otro lado, el osciloscopio dispondrá de un generador de señales controlable controlar a través de la interfaz web. Se podrán generar señales de distinto tipo (senoides, señales cuadradas, etc) y con amplitud y frecuencia manejables por el cliente. La máxima amplitud de generación de señal será de 5V y con una frecuencia máxima de 1KHz.

Aunque existen actualmente osciloscopios más ‘de bolsillo’ en el mercado, cierto es que su precio es bastante elevado, rondando los 50€ como mínimo por dispositivos con muy poca funcionalidad. El objetivo de este proyecto es, no solo conseguir una reducción en el tamaño del dispositivo-, sino también una reducción en el coste para que sea más accesible. La idea inicial se propone un mini-osciloscopio con un coste total no superior a los 25€ (dispositivo de visualización excluido).

En último lugar, nuestro osciloscopio dispone de un PowerBank con el cual buscamos una autonomía de hasta 24h.

## 4. Planificación.

Antes de empezar con un proyecto es importante una previa organización temporal para poder guiarnos a la hora de saber qué pasos debemos seguir, cuando vamos a dar esos pasos y la posible duración final del proyecto.

Guiándome por el documento de ejemplo que la Universidad de Oviedo nos proporciona en la Intranet, he organizado este proyecto de la siguiente manera:

- Marzo 2021. Búsqueda de información que servirá de base para nuestro objetivo en este proyecto y servirá de introducción para un entendimiento inicial de hacia dónde va dirigido este trabajo. También nos servirá para ir escogiendo según la información recopilada los distintos componentes, lenguajes de comunicación, etc.
- Mayo 2021. Desarrollo de la parte hardware del proyecto. Pruebas en el laboratorio para posteriormente poder diseñar la PCB definitiva.
- Junio 2021. Una vez realizadas las pruebas se envía la PCB a imprimir. Comienzo de la parte software del programa.
- Julio 2021. Trabajo de soldar los componentes en la PCB. Una vez soldado, programación del ESP32.
- Septiembre 2021. Unión de toda la información recopilada para unificarla en la memoria, con las decisiones que hemos ido tomando. Continuación de la programación tanto del ESP32 como del servidor web.
- Octubre 2021. Perfeccionamiento de los últimos puntos del proyecto, finalización de la memoria, entrega y defensa.



## 5. DISEÑO

### 5.1.- Hardware.

En este apartado se describe todo el desarrollo de la parte de Hardware del mini-osciloscopio. En él se engloba tanto el circuito completo, como los distintos circuitos subinternos, los componentes y la impresión y soldadura de la PCB.

#### 5.1.1.- Circuito.

El osciloscopio se ha diseñado para poder medir señales tanto de voltaje como de corriente, y además una salida con la que se podrá generar distintas señales.

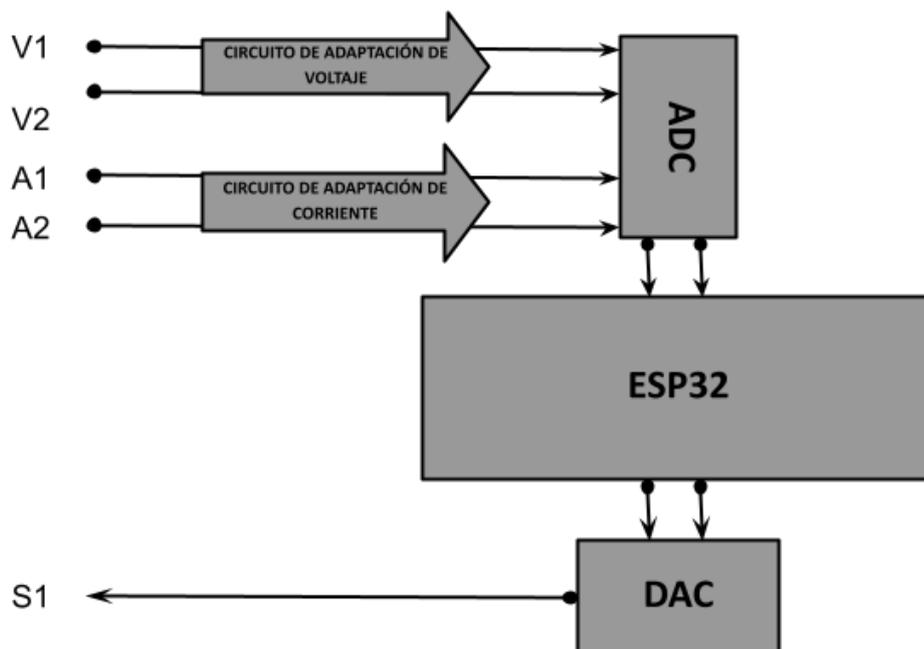


Figura 5.1.- Diagrama de bloques del mini-osciloscopio.

En la figura 5.1. podemos ver el diagrama de bloques del osciloscopio. En primer lugar se encuentran 4 entradas distintas, dos de voltaje y dos de corriente. Estas cuatro señales pasarán al convertidor analógico digital para posteriormente poder tratar los datos con el microcontrolador.

Tanto las señales de voltaje como las de corriente necesitan de un circuito de adaptación para poder ser recibidas por el ADC.

#### **5.1.1.1.- Circuito de adaptación de voltaje.**

Para nuestro osciloscopio, como hemos mencionado anteriormente, utilizaremos un ADC que convertirá nuestra señal analógica en una señal digital para poder tratarla en nuestro microcontrolador.

El uso de este ADC nos supone una limitación: un dispositivo digital no puede leer valores que caigan por debajo de 0. Más concretamente, este ADS1015 solo leerá valores que oscilen entre los 0 y los 5V, es decir, el voltaje de la alimentación.

Por lo tanto necesitamos que, sea cual sea la señal de entrada, ésta siempre entre al ADC entre esos dos valores, y luego ya hacer el ajuste por código.

Para ello utilizaremos el circuito de adaptación que vemos en la figura 5.2., en el que R2 corresponde al potenciómetro digital.

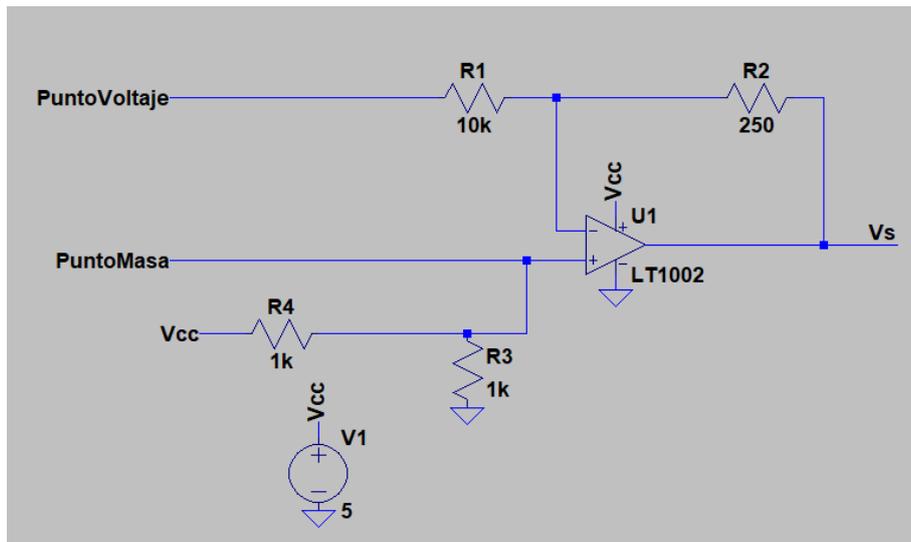


Figura 5.2.- Circuito de adaptación de voltaje.

La masa virtual se encuentra a 2,5V, puesto que se obtiene ese valor gracias al divisor resistivo formado por las resistencias R4 y R3. A partir de la función 5.1. obtenemos el valor de voltaje de nuestra masa virtual.

$$PuntoMasa = \frac{R3}{R4+R3} \cdot Vcc \quad (5.1)$$

$$PuntoMasa = \frac{1000\Omega}{1000\Omega + 1000\Omega} \cdot 5V = 2,5V$$

Este punto lo que hará será añadirle ese ‘offset’ a todas las señales que midamos. Es decir, en nuestro osciloscopio tendremos un par de cables que actuarán como sondas, con su contacto para medir la señal y su contacto para colocar en la ‘masa’ del circuito a medir. Esta última irá conectada a esta masa virtual, por lo que todas las señales que midamos van a ser medidas respecto a una ‘masa’ que en vez de estar a 0V, estará a 2,5V.

Una vez establecida esta masa virtual, utilizaremos el amplificador operacional para amplificar nuestra señal o disminuirla según nos interese. En el osciloscopio dispondremos de dos entradas de señal que se utilizarán una u otra en función de la amplitud de señal que queramos medir:

- Señales entre 0 y 3 V. Esta sonda se utilizará para medir señales más bajas. El circuito será el siguiente:

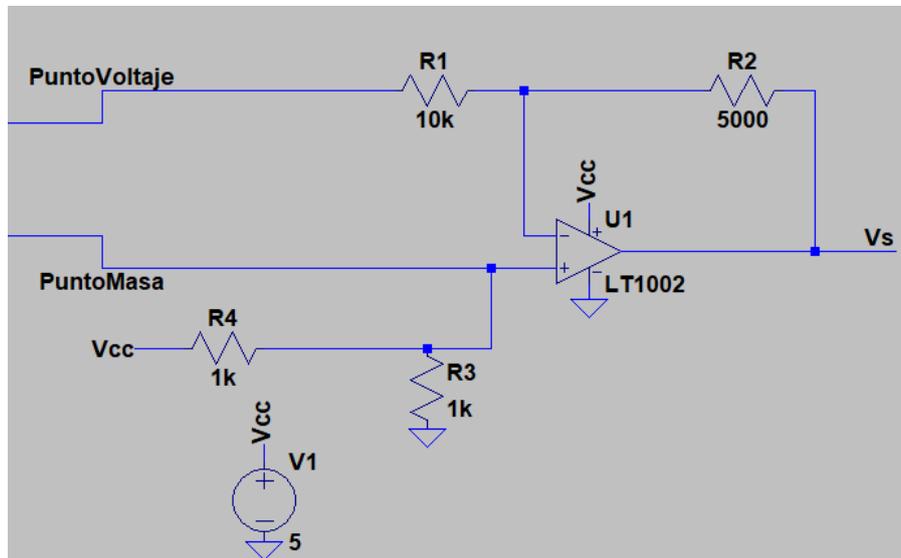


Figura 5.3.- Circuito de adaptación de voltaje para señales inferiores a 3V.

Para esta entrada el valor del potenciómetro será de  $5000\Omega$ . La señal llegará invertida pero, al igual que su amplitud que vendrá multiplicada por algún factor, será por código cuando volveremos a invertirla para que llegue correctamente al osciloscopio.

A continuación realizaremos un ejemplo con una entrada de 2V:

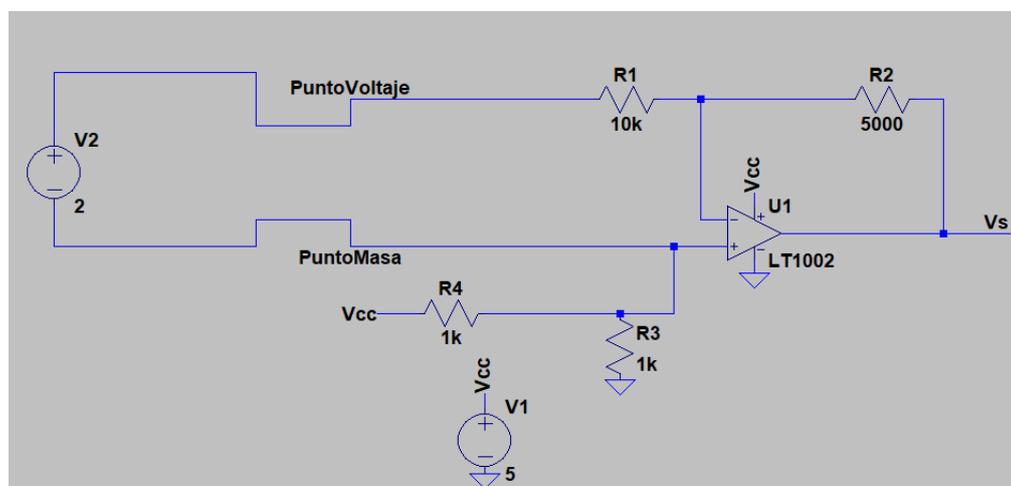


Figura 5.4.- Ejemplo de entrada de una señal de 2V.

El valor de señal de salida (que será la entrada al ADC) es de 1,4V como podemos ver en la figura 5.5.

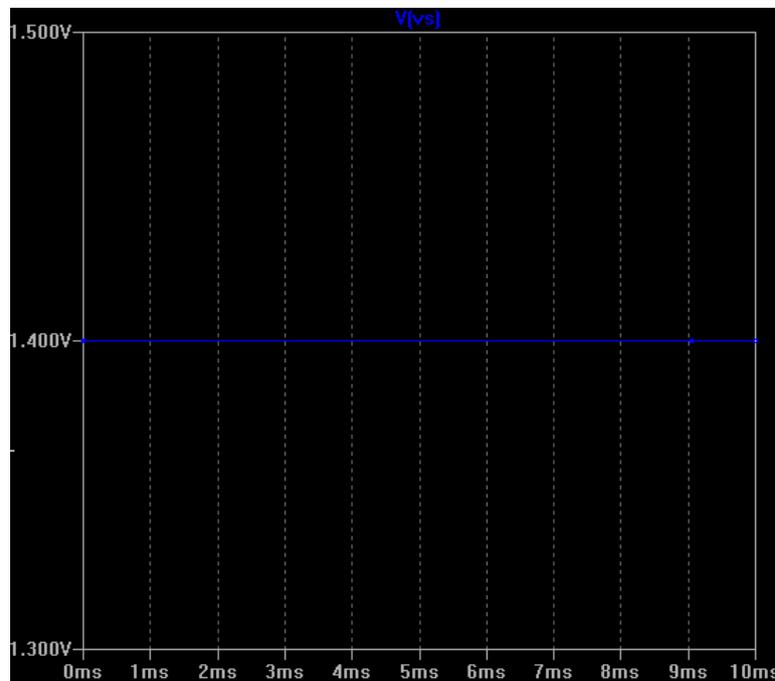


Figura 5.5.- Salida del circuito.

Esta señal es la que leeremos en nuestro ADC, que viene reducida, con un offset 2,5V e invertida. Por lo tanto debemos hacer el siguiente cálculo en el código de nuestro ESP32:

$$V_{invertida} = (5V - 1,4V) = 3,6 V$$

$$V_{final} = 3,6 V - 2,5V = 1,1V$$

Buscamos una relación directa entre la salida y la entrada:

$$\frac{V_e}{V_s} = \frac{2 V}{1,1 V} = 1,8181$$

Ahora sabemos que, para poder pintar la señal tal y como entró a nuestro osciloscopio, debemos realizar los pasos de volver a invertir la señal, restarle los 2,5V de offset y finalmente aplicarle esa relación para obtener el valor real de la señal medida.

- Señales entre 3 y 20V. En este caso necesitamos que la señal sea reducida lo máximo posible para poder alcanzar el mayor valor posible de voltaje de entrada. Como el mínimo valor de nuestro potenciómetro es de  $150\Omega$ , será este el utilizado. El máximo voltaje que admite a la entrada con ese valor de resistencia sin saturarse es una señal de 20V. En este caso el circuito sería el mostrado en la figura 5.5.

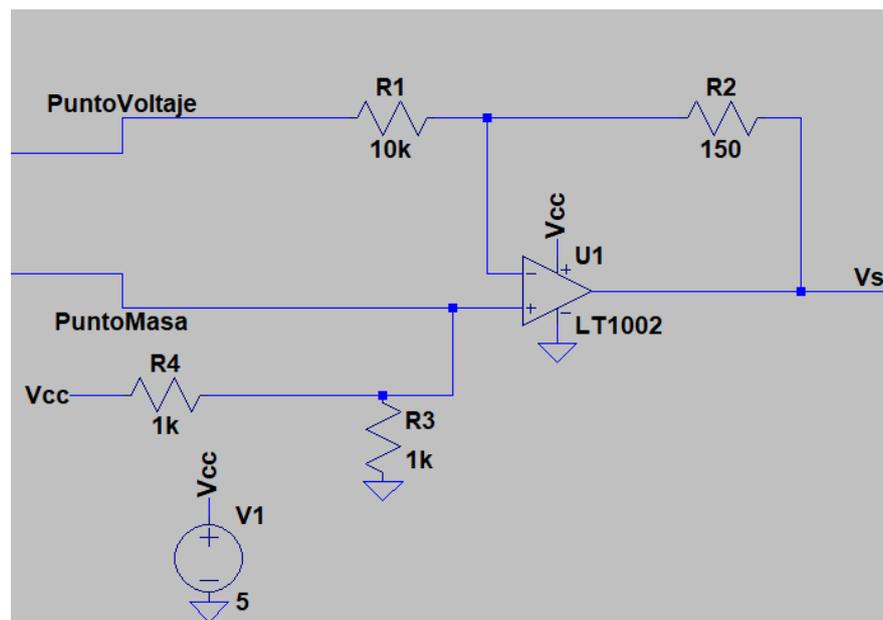


Figura 5.6.- Circuito de adaptación para medir señales con amplitud mínima de 3V.

Para poner un ejemplo de cómo funcionaría vamos a simular un circuito en el que la señal de entrada sea una senoide con un voltaje máximo de 10V y una frecuencia de 1000 Hz, aunque este dato no afecta al proceso de adaptación de la señal. El circuito de ejemplo es el mostrado en la figura 5.6.

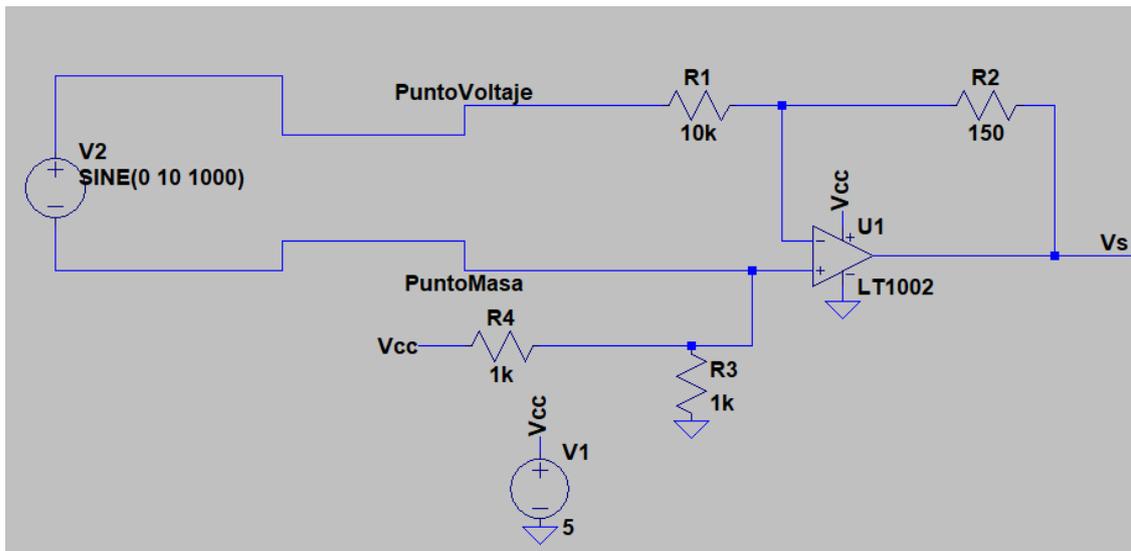


Figura 5.7.- Circuito de adaptación con una entrada senoidal de 10V de amplitud.

En este caso, a la salida de nuestro circuito obtenemos el siguiente valor:

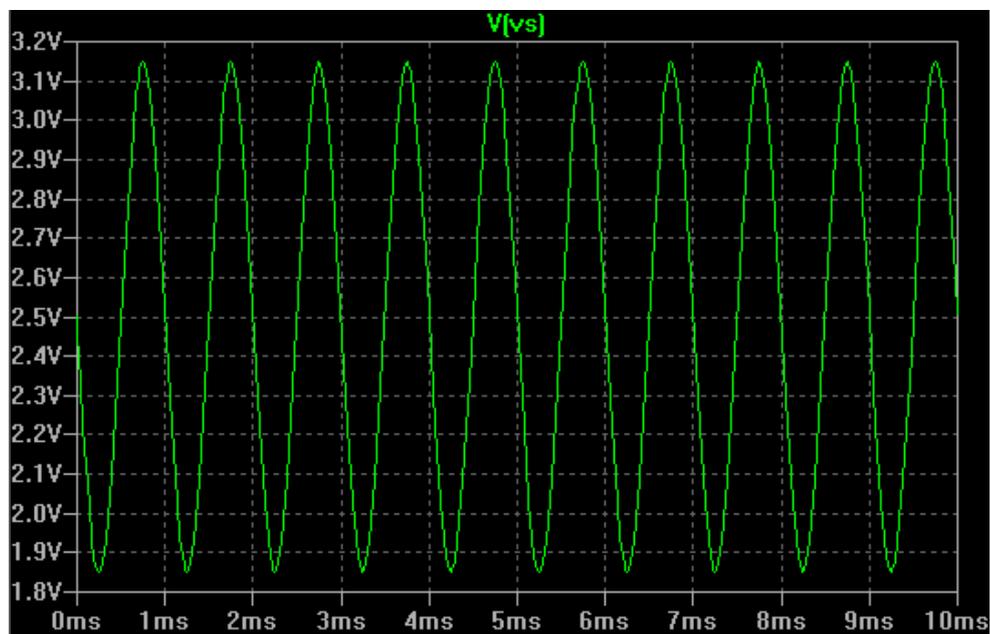


Figura 5.8.- Señal de salida del circuito de adaptación.

Como hemos indicado en el punto anterior, esta señal es la que leeremos en nuestro ADC, que viene reducida, con un offset 2,5V e invertida. Es decir, el punto más bajo

(aproximadamente 1,85V) equivale al punto más alto (los 10V). Por lo tanto debemos hacer el siguiente cálculo en el código de nuestro ESP32:

$$V_{invertida} = (5V - 1,85) = 3,15 V$$

$$V_{final} = 3,15 V - 2,5V = 0,65 V$$

Buscamos una relación directa entre la salida y la entrada:

$$\frac{V_e}{V_s} = \frac{10 V}{0,65 V} = 15,38$$

Ahora sabemos que, para poder pintar la señal tal y como entró a nuestro osciloscopio, debemos realizar los pasos de volver a invertir la señal, restarle los 2,5V de offset y finalmente aplicarle esa relación para obtener el valor real de la señal de entrada.

Para poner un ejemplo de como utilizaríamos este valor de 15,38V, hacemos lo siguiente. Si tenemos un voltaje de 20V a la entrada, según la relación que acabamos de obtener sabemos que a la salida del circuito obtendremos un valor de 1,3 V.

$$\frac{20 V}{15,38} = 1,3 V$$

Ahora simularemos esta situación de la misma forma, con una señal de entrada senoidal de 20V de amplitud. En la figura 5.8. podemos ver el circuito a simular.

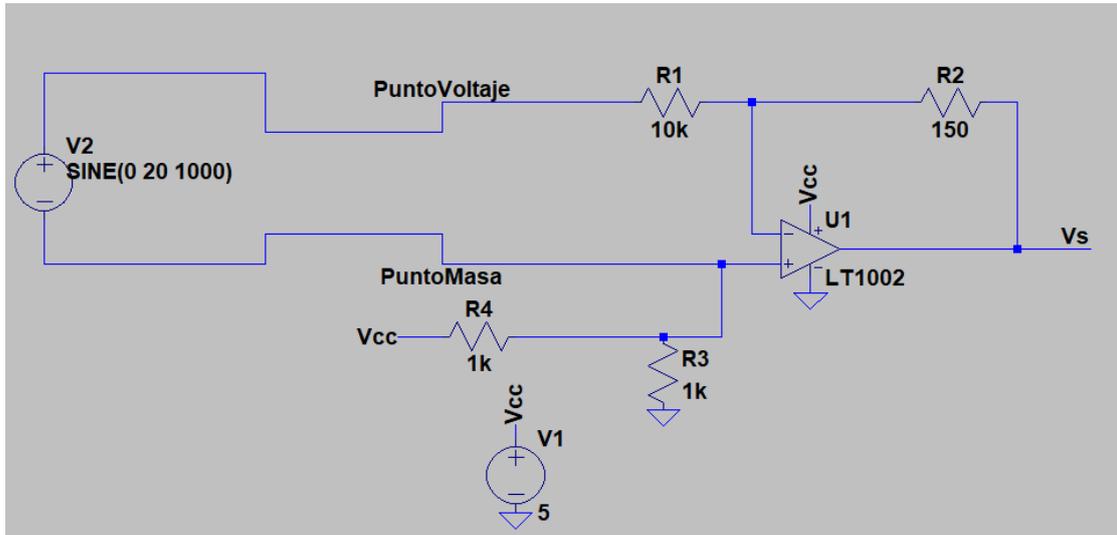


Figura 5.9.- Circuito de adaptación con una senoide de 20V a la entrada.

Si los cálculos son correctos, a la salida deberíamos obtener una señal senoidal invertida, reducida y con un offset de 2,5V.

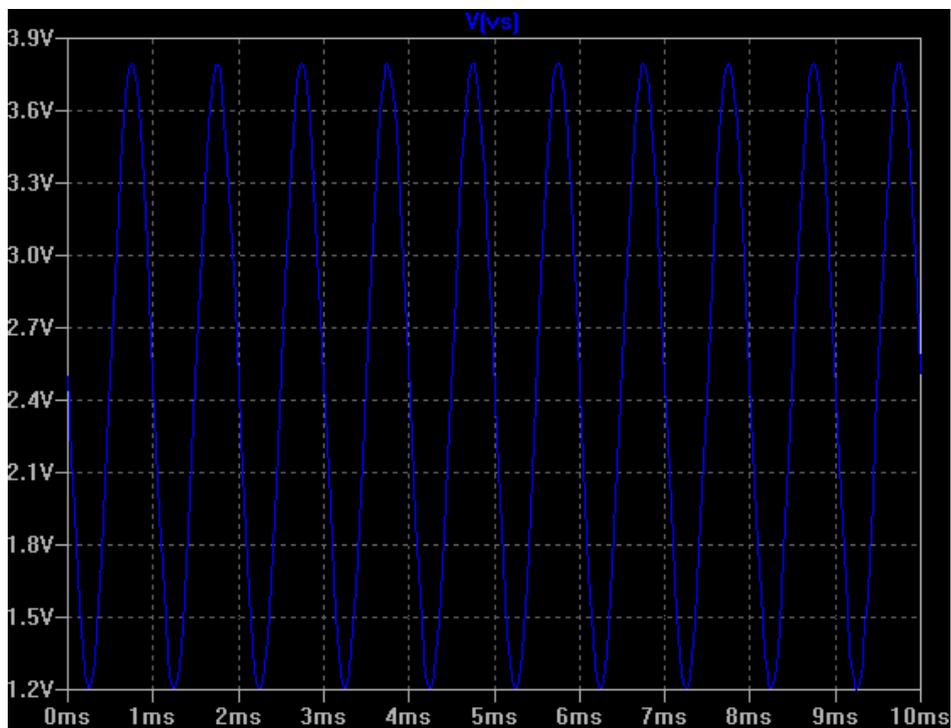


Figura 5.10.- Señal de salida del circuito de adaptación para las condiciones de la figura 5.8.

Como podemos ver en la figura 5.9. aparece una señal de salida (entrada al ADC) de unos 1,2V en el punto que sería máximo de la senoide. Si volvemos a realizar los cálculos, nos sale lo siguiente:

$$V_{invertida} = (5V - 1,2) = 3,8 V$$

$$V_{final} = 3,8 V - 2,5V = 1,3 V$$

Lo que coincide con nuestros cálculos previos.

#### 5.1.1.2.- Circuito de adaptación de corriente.

El circuito de adaptación de la corriente es mucho más sencillo que el anterior, puesto que en este caso solo hemos utilizado una resistencia. El circuito es el siguiente:

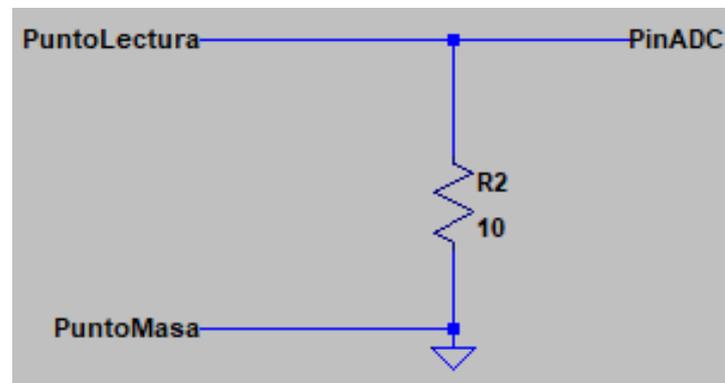


Figura 5.11.- Circuito de adaptación de corriente.

En este caso tenemos, como en todas las entradas, dos cables de entrada que utilizaremos para medir. La resistencia R2 es una resistencia que hemos implementado a nivel interno y cuyo valor es de 10  $\Omega$ . Es importante que el valor de esta resistencia sea bastante bajo para que haya la menor distorsión en corriente del circuito a medir posible.

El funcionamiento sería como el mostrado en la figura 5.11.

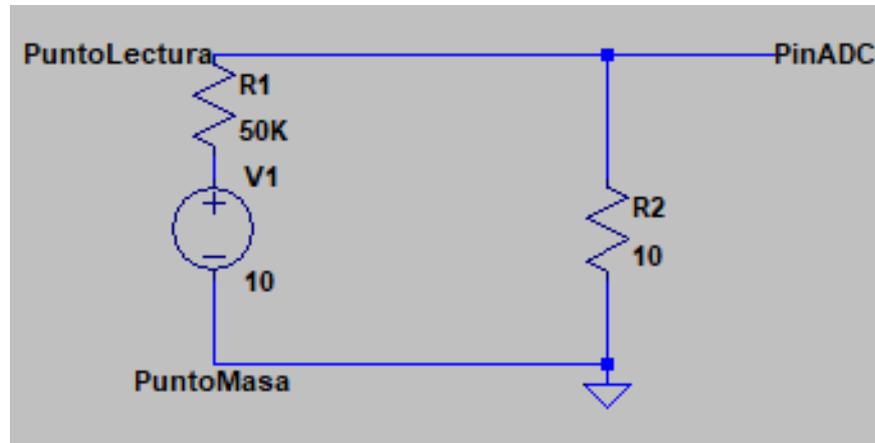


Figura 5.12.- Circuito de adaptación de corriente conectado a una resistencia de  $50\text{k}\Omega$  y una fuente de voltaje de 10V.

Tenemos en el ejemplo un circuito muy sencillo de una fuente de voltaje de 10V y una resistencia  $R1$  de  $50\text{k}\Omega$  en serie. Queremos saber cuál es la corriente de esta línea, y para ello colocamos el amperímetro en serie a la línea de corriente a medir.

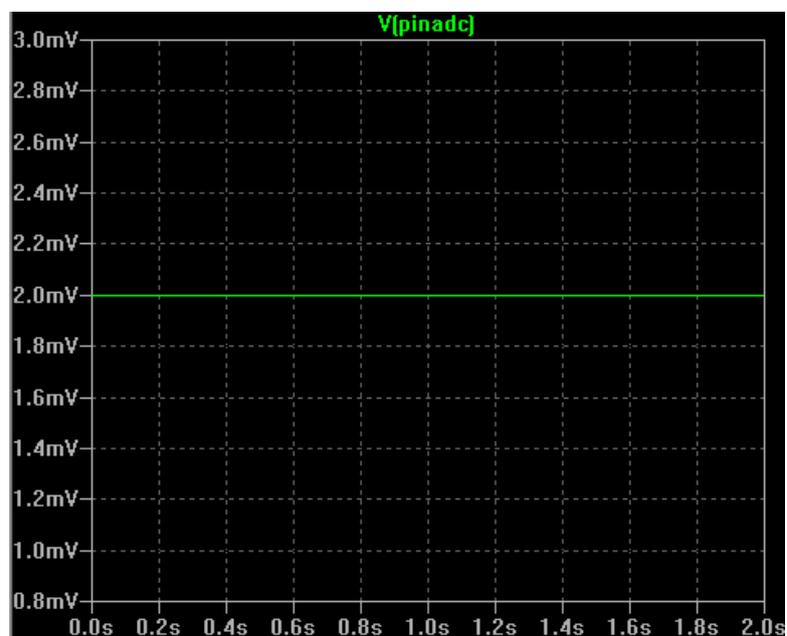


Figura 5.13.- Voltaje de salida a nuestro pin ADC.

Al punto  $V_{\text{salida}}$  llegará el voltaje que recorre a nuestra resistencia R2. Al ADC llegará este valor del voltaje, pero sabiendo el valor de nuestra resistencia interna, en nuestro ESP32 simplemente realizaremos la ley de Ohm para obtener el valor de corriente.

Siendo el valor de voltaje en el punto  $V_{\text{salida}}$  de 2 mV, el cálculo es el siguiente:

$$I = 2 \text{ mV}/10 = 0,2 \text{ mA} = 200 \text{ }\mu\text{A}$$

El cual coincide con el la corriente del circuito medido, aunque con algo de caída en la amplitud:

$$I = 10/50\text{k} = 0,0002 \text{ A} = 0,2 \text{ mA} = 200 \text{ }\mu\text{A}$$

La máxima corriente medible corresponde con aquella que genere en la resistencia de shunt ( $10 \text{ }\Omega$ ) el voltaje de saturación del ADC, el cual se corresponde con los de la fuente de alimentación (5V). Siguiendo de nuevo la ley de Ohm:

$$I_{\text{MÁX}} = 5 \text{ V} / 10 \text{ }\Omega = 0,5 \text{ A} = 500 \text{ mA}$$

### 5.1.2.- Componentes.

A continuación se detallarán los distintos componentes seleccionados para el desarrollo del osciloscopio, así como sus características principales y las ventajas y desventajas a tener en cuenta.

#### 5.1.2.1.- Microcontrolador: ESP32.

El ESP32 es una familia de microcontroladores de bajo coste y bajo consumo. Es el sucesor de ESP8266. Estos controladores disponen de un chip con WiFi y Bluetooth de modo dual integrados. Suponen una gran ventaja a la hora de desarrollar puesto que tiene un bajo consumo de energía, múltiples entornos de código abierto disponibles y bibliotecas.

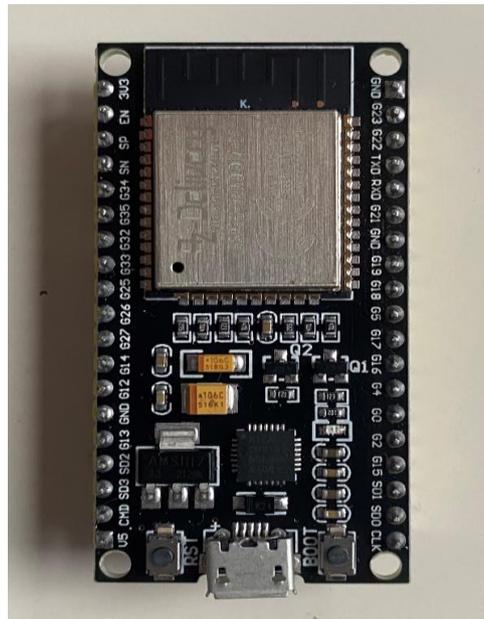


Figura 5.14.- ESP32.

➤ **Características técnicas.**

En la figura 5.14 [60] podemos ver la circuitería del ESP32 y en la figura 5.15 la estructura en diagrama de bloques que define al microprocesador.

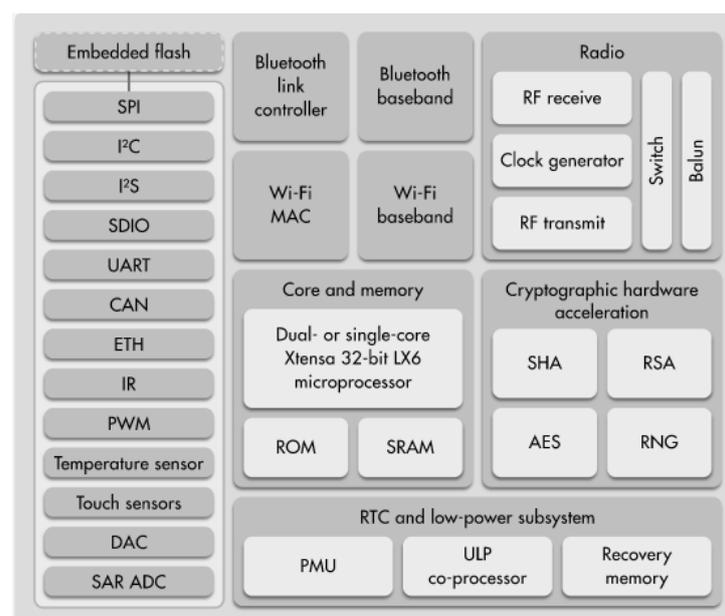


Figura 5.15.- Diagrama de bloques del ESP32.

Processors	Main processor	Tensilica Xtensa 32-bit LX6 microprocessor	
		Core	2 or 1
		Clock frequency	Up to 240 MHz
		Performance	Up to 600 DMIPS
	Ultra low power co-processor	Allows you to do ADC conversions, computation, and level thresholds while in deep sleep	
Wireless connectivity	Wi-Fi	802.11 b/g/n/e/i (802.11n @ 2.4 GHz up to 150 Mbit/s)	
	Bluetooth	v4.2 BR/EDR and Bluetooth Low Energy (BLE)	
Memory	Internal memory	ROM	448 KiB
		SRAM	520 KiB
		RTC fast SRAM	8 KiB
		RTC slow SRAM	8 KiB
		RTC slow SRAM	8 KiB
		eFuse	1 Kibit
		Embedded flash	0 MiB (ESP32-D0WDQ6, ESP32-D0WD, and ESP32-S0WD chips)
			2 MiB (ESP32-D2WD chip)
			4 MiB (ESP32-PICO-D4 SiP module)
			4 MiB (ESP32-PICO-D4 SiP module)
External flash & SRAM	ESP32 supports up to four 16 MiB external QSPI flashes and SRAMs with hardware encryption based on AES to protect developers' programs and data.		
Peripheral input/output	36 GPIO		
	16 ADC 12-bit resolution		
	2 DAC 8-bit		
	Up to 16 PWM channels		
	2 I2C channels & 4 SPI channels		
Security	IEEE 802.11 standard security features all supported, including WPA, WPA/WPA2 and WAPI		

Tabla 5.1.- Características técnicas del microcontrolador ESP32 [60].



- ✓ Posibilidad de utilizar más sensores de lecturas similares sin necesidad de multiplexar.
- ✓ Bluetooth Low Energy (BLE), muy atractivo para proyectos de IoT.

### 5.1.2.2.- Amplificador operacional OP2277.

El amplificador operacional OP2277 pertenece a la familia de amplificadores OPAx277. Estos dispositivos son de precisión y reemplazan al estándar industrial OP-177. Ofrecen un ruido mejorado, una variación más amplia en el voltaje de salida y el doble de velocidad en reposo con la mitad de corriente [62].



Figura 5.17.- Chip OPA2277.

Estos componentes trabajan con voltajes de entrada de  $\pm 2V$  a  $\pm 18V$ . Se caracterizan por ser bastante sencillos de utilizar y estar liberados de inversión de fase y problemas de sobrecarga. La versión utilizada es la dual, lo que significa que dentro del mismo componente obtenemos dos amplificadores independientes.

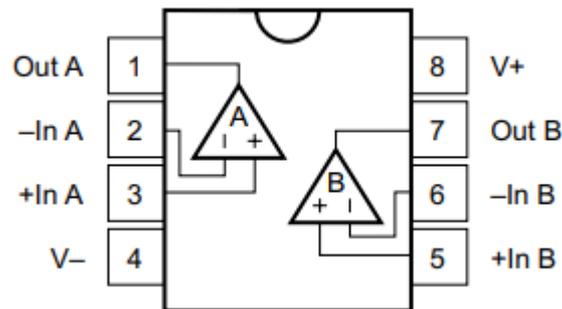


Figura 5.18.- Estructura interna del OPA2277 dual [62].

En cuanto al tipo de encapsulado, en este caso utilizamos un OPA2277 con montaje superficial, también conocido como SMD (Surface Mounting Device). Este tipo de componentes se sueldan de forma directa a la superficie de la PCB a través de los pads [63]. El caso contrario son los componentes de tecnologías de agujeros pasantes o throughhole, los cuales se sueldan en la capa contraria a la que se coloca el dispositivo.

➤ **Detalles técnicos.**

En la tabla 5.2. podemos ver un resumen de algunos de los detalles técnicos que proporciona el ADC

Ultralow Offset Voltage: 10 $\mu$ V
Ultralow Drift: $\pm 0.1 \mu$ V/ $^{\circ}$ C
High Open-Loop Gain: 134 dB
High Common-Mode Rejection: 140 dB
High Power Supply Rejection: 130 dB
Low Bias Current: 1-nA maximum
Wide Supply Range: $\pm 2$ V to $\pm 18$ V
Low Quiescent Current: 800 $\mu$ A/amplifier
Single, Dual, and Quad Versions

Tabla 5.2.- Detalles técnicos del Amplificador Operacional OPA2277.

### 5.1.2.3.- ADC ADS1015.

El ADC ADS1015 es un convertidor analógico-digital de mayor precisión con 12 bits de resolución, realizando lecturas de hasta 3300 muestras/segundo. Opera con un voltaje de entrada de 2 a 5 V.

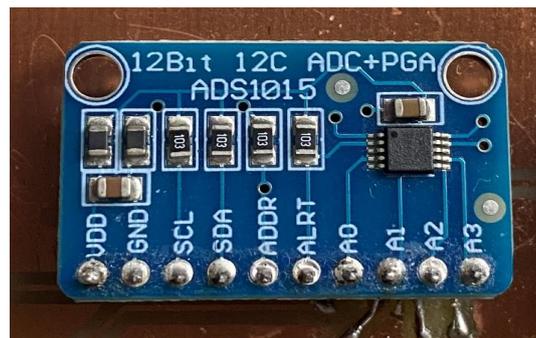


Figura 5.19.- Chip ADS1015.

Los datos se envían a través del protocolo I2C, disponiendo de 4 direcciones I2C para seleccionar. Estos cuatro canales pueden ser configurados como cuatro entradas distintas o como dos canales diferenciales gracias a un multiplexor que tiene integrado. Adicionalmente, el AD proporciona un amplificador de ganancia programable [64].

A pesar de que nuestro microcontrolador ESP32 ya dispone de una entrada ADC, hemos decidido utilizar un convertidor externo como este para poder obtener una mayor precisión, lo que es un punto clave al tratarse de un osciloscopio.

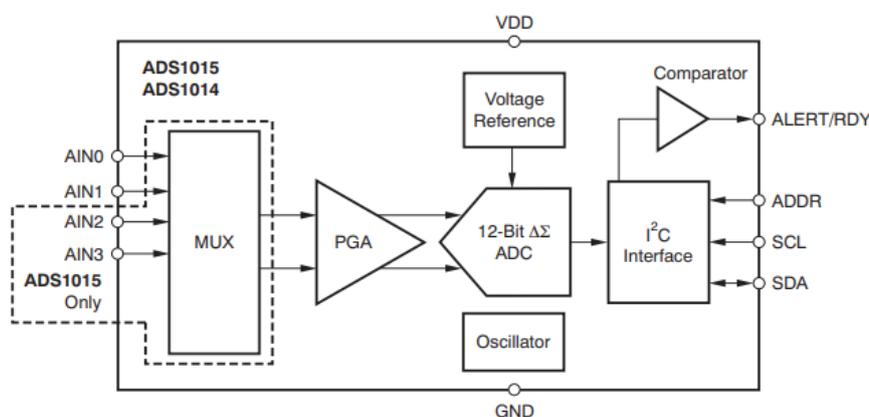


Figura 5.20.- Diagrama de bloques de ADC ADS1015 [64].

➤ **Detalles técnicos.**

En la tabla 5.3 podemos ver un resumen de algunos de los detalles técnicos que proporciona el ADC

ULTRA-SMALL QFN PACKAGE: 2mm × 1,5mm × 0,4mm
WIDE SUPPLY RANGE: 2.0V to 5.5V
LOW CURRENT CONSUMPTION: Continuous Mode: Only 150µA Single-Shot Mode: Auto Shut-Down
PROGRAMMABLE DATA RATE: 128SPS to 3.3kSPS
INTERNAL LOW-DRIFT VOLTAGE REFERENCE
INTERNAL OSCILLATOR
INTERNAL PGA
I2C INTERFACE: Pin-Selectable Addresses
FOUR SINGLE-ENDED OR TWO DIFFERENTIAL INPUTS
PROGRAMMABLE COMPARATOR

Tabla 5.3.- Características técnicas del ADS1015

**5.1.2.4.- DAC MCP4725.**

El convertidor digital-analógico MCP4725 es un dispositivo de salida de voltaje con un buffer de 12 bits, con un único canal, de baja potencia y alta precisión. También dispone de una memoria no volátil EEPROM.

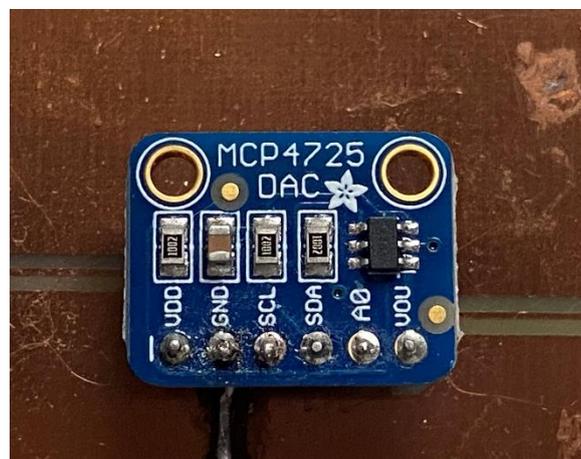


Figura 5.21.- Chip MCP4724.

La comunicación con el DAC se transmite a través del protocolo de I2C, a través del cual podemos convertir señales digitales a analógicas, siendo controlado a través de controladores o microcontroladores. El DAC MCP4725 se controla con un voltaje de entrada entre 2.7V y 5.5V.

En la figura 5.21. podemos ver la estructura en diagrama de bloques del DAC.

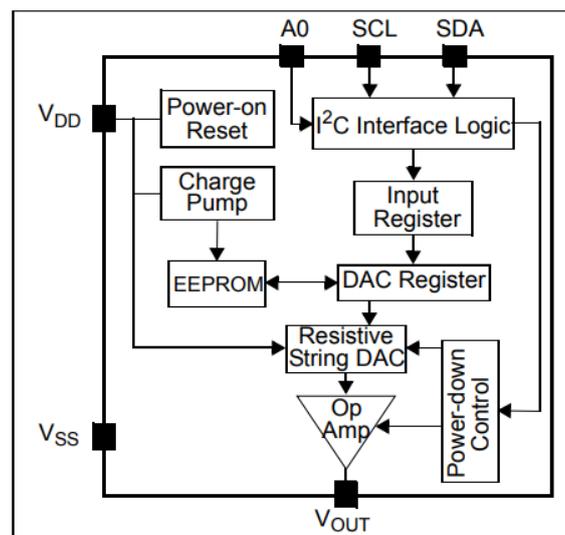


Figura 5.22.- Diagrama de bloques del DAC MCP4725.

### ➤ Detalles técnicos.

El DAC MCP4725 proporciona una serie de características técnicas que podemos ver resumidas en la tabla 5.4 .

12-Bit Resolution
On-Board Non-Volatile Memory (EEPROM)
±0.2 LSB DNL (typical)
External A0 Address Pin
Normal or Power-Down Mode
Fast Settling Time: 6 μs (typical)
External Voltage Reference (VDD)
Rail-to-Rail Output
Low Power Consumption
Single-Supply Operation: 2.7V to 5.5V
I2C Interface
Small 6-lead SOT-23 Package
Extended Temperature Range: -40°C to +125°C

Tabla 5.4.- Características técnicas del DAC MCP4725.

#### 5.1.2.5.- Potenciómetro digital MCP42050

El potenciómetro digital MCP42050 pertenece a la familia de MicroChip MCP42XXX. Se trata de un dispositivo de 255 posiciones, con dos canales diferentes en un encapsulado de 14 pines. Estos componentes operan con un voltaje de entrada de entre 2,7V y 5,5V.



Figura 5.23.- Chip MCP42050.

La posición en la que se coloca el potenciómetro varía de forma lineal y se puede controlar a través de una interfaz SPI. En nuestro caso nuestro potenciómetro proporciona valores de resistencia entre  $150\Omega$  y  $50k\Omega$ , con un cambio mínimo de  $200\Omega$ .

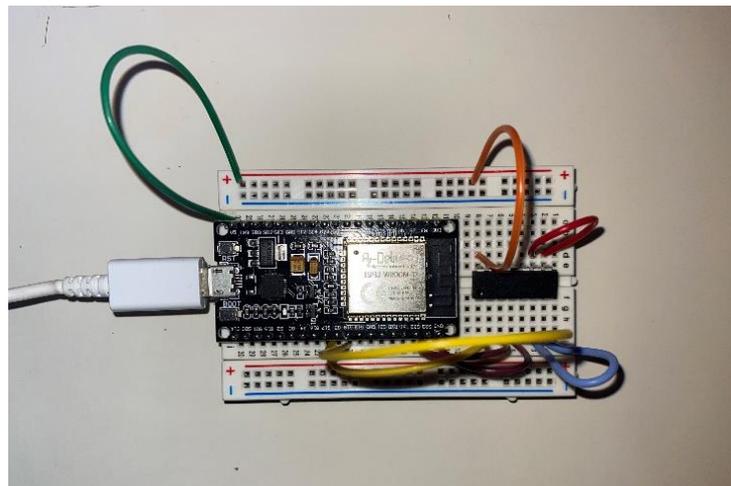


Figura 5.24.- Potenciómetro digital MCP42050 conectado con un ESP32.

En la figura 5.24. podemos ver el diagrama de bloques de estos dispositivos.

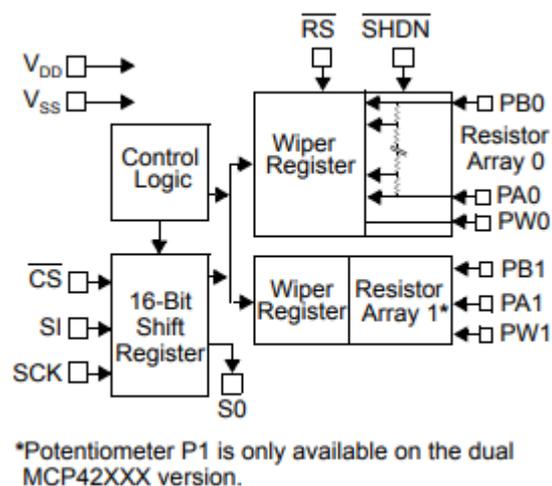


Figura 5.25.- Diagrama de bloques de un MCP42050 [65].

➤ **Detalles técnicos.**

En la tabla 5.5 podemos ver las características técnicas aportadas por el datasheet [65] del chip MCP42050.

256 taps for each potentiometer
Potentiometer values for 10 k $\Omega$ , 50 k $\Omega$ and 100 k $\Omega$
Single and dual versions
SPI™ serial interface (mode 0,0 and 1,1)
$\pm 1$ LSB max INL & DNL
Low power CMOS technology
1 $\mu$ A maximum supply current in static operation
Multiple devices can be daisy-chained together (MCP42XXX only)
Shutdown feature open circuits of all resistors for maximum power savings
Hardware shutdown pin available on MCP42XXX only
Single supply operation (2.7V - 5.5V)

Figura 5.26.- Características técnicas del MCP42050.

### 5.1.3.- Diseño de la placa.

Para el diseño del circuito integrado, como hemos mencionado en la introducción, existe un gran número de aplicaciones disponibles para su desarrollo. En este caso he hecho uso de la aplicación Eagle, de la casa de Autodesk.

En primer lugar hay que realizar un esquemático del circuito, utilizando los distintos componentes que posteriormente colocaremos en la placa. Muchos de estos componentes no están disponibles por defecto en el programa, y por lo tanto hay que descargarse distintas librerías para poder obtenerlos.

El esquemático es el mostrado en la figura 5.26.



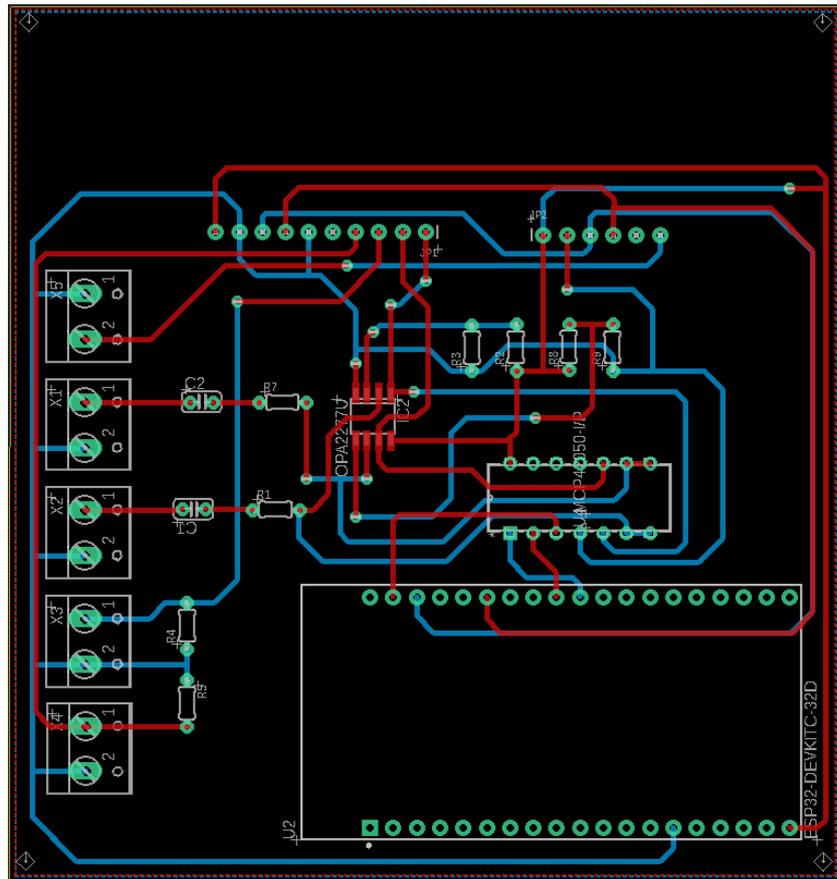


Figura 5.28.- Board en Eagle.

#### 5.1.4.- PCB.

Una vez diseñada la PCB se imprimió en la propia Escuela Politécnica de Ingeniería de Gijón.

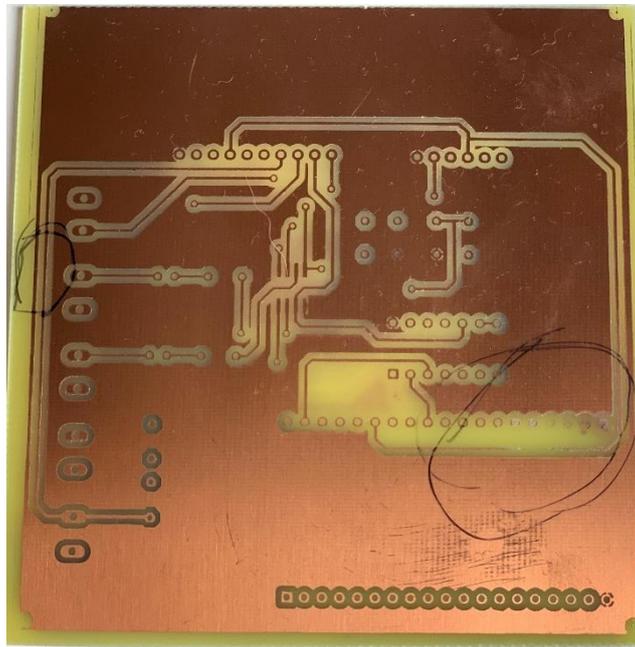


Figura 5.29.- Capa TOP de la PCB.

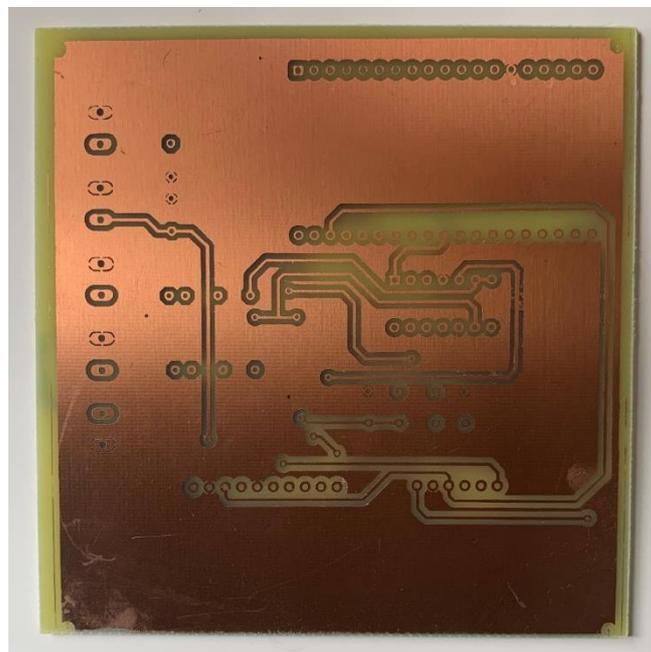


Figura 5.30.- Capa bottom de la PCB.

Inicialmente, como se puede ver la capa TOP venía con muchos defectos como pistas borradas o cortadas. Posteriormente se solucionó colocando un cable en la pista perdida y uniendo los cortes.

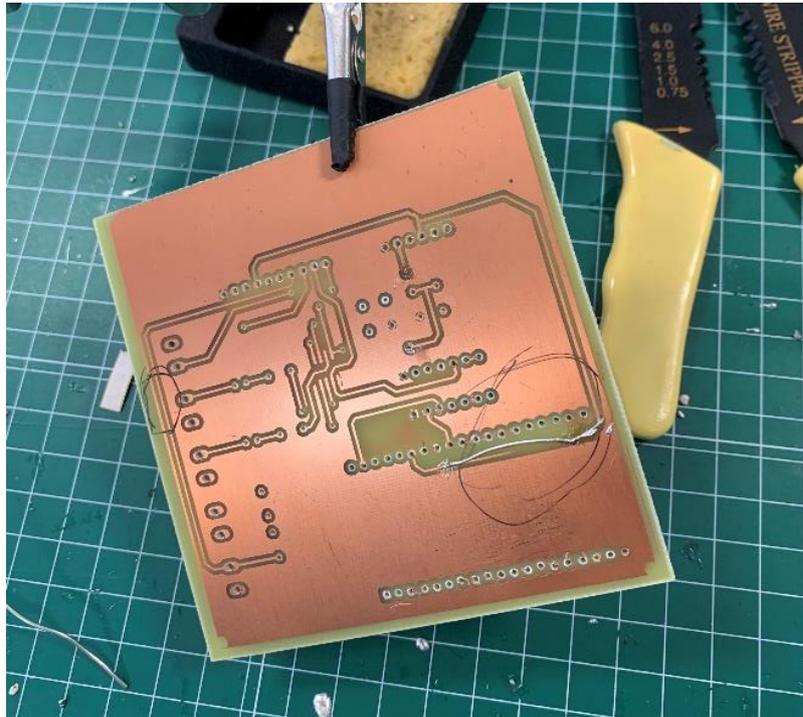


Figura 5.31.- PCB en el laboratorio preparada para soldar.

En la figura 5.30. se ve que la unión se realizó con un cable de metal pero posteriormente se cambió por un cable para evitar el mayor número de pérdidas posible.

Una vez soldada la placa con los componentes colocados, se tuvieron que hacer modificaciones puesto que a la hora de diseñar la placa no se tuvo en cuenta, por ejemplo, el punto de conexión de la señal de entrada con la masa virtual. Posteriormente esto se solucionó soldando un cable a ese punto y unido a la entrada de la masa de la señal.

En la siguiente imagen podemos ver la PCB definitiva con todos los componentes soldados y las modificaciones pertinentes<sup>1</sup>. Los cables blancos son todos los de señal y los marrones y morado son los de las masas.

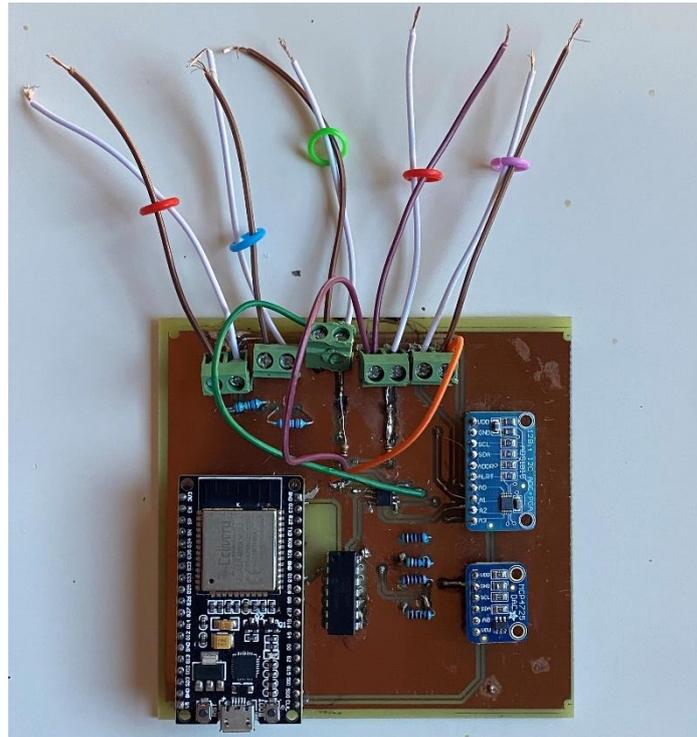


Figura 5.32.- Mini-osciloscopio.

<sup>1</sup> Esta imagen es la de la PCB final, posterior a realizar todas las pruebas del apartado 6. Para las pruebas se han quitado cables y se han tenido que ajustar algunos puntos de soldadura, por eso aparecen algo distintos. Esta imagen es el resultado final de ajustar todos los puntos correctamente.

## 5.2.- Software.

En este apartado describiremos la parte de software que hemos tenido que desarrollar en este proyecto. Esto engloba por un lado el código necesario para programar nuestro microcontrolador ESP32 y por otro lado el código para el desarrollo de la interfaz web.

### 5.2.1.- Código Arduino

Como hemos mencionado en la introducción, para que un microcontrolador ejecute alguna función debemos utilizar un software para ello. El software que hemos seleccionado para este caso es el IDE de Arduino, que utilizando un lenguaje C y gracias al compilador que contiene podemos mandar distintas funciones a nuestro microcontrolador.

En primer lugar el IDE de Arduino necesita de librerías de código, y ha sido necesario añadir alguna extra para poder utilizar, por ejemplo, nuestros componentes ADC o DAC.

Por un lado, necesitamos programar nuestro ADC. El ADS1015, como hemos comentado antes, se comunica mediante protocolo I2C. La idea es que nuestro osciloscopio lea de forma constante todos los valores que tiene a la entrada de sus cuatro pines.

Por otro lado, tenemos el DAC, que se comunica también a través de I2C. En este caso, generaremos 4 tipos de señales: senoidales, triangulares, cuadradas y continuas.

Nuestro osciloscopio cuenta también con un potenciómetro digital, cuyo valor de resistencia variaremos a través del código.

Finalmente, necesitamos un servidor para alojar nuestra página web. Utilizando el protocolo Ajax declararemos un servidor en el puerto 80 en el que alojaremos nuestra página web. La IP vendrá dada de nuestro router, al cual nos tendremos que conectar cada vez que se encienda de nuevo el ESP32.

Para las peticiones con Ajax, tenemos cinco funciones:

- La primera levantará nuestra página principal.
- Otra función a la que llamará el cliente desde la página de forma periódica para recibir los datos del ADC. A cada llamada que se realice, el ESP32 devolverá una cadena con todos los datos almacenados de uno de sus pines del ADC. La selección de estos pines se realiza a través de las siguientes funciones.
- Como hemos mencionado, existirá otra función a través de la cual el cliente enviará el ‘canal’ que quiere leer, es decir, de cuales de los pines quiere recibir la información.
- Una cuarta función se define para que el cliente pueda avisar al ESP32 de que quiere apagar o encender el generador. Esto se ha implementado para que el DAC no esté funcionando en segundo plano y perdamos velocidad.
- La última función sirve para que el cliente envíe al ESP32 los valores de la señal que quiere generar con el DAC. Estos valores son el tipo de señal, la amplitud y su frecuencia.

En la siguiente figura podemos ver el diagrama de flujo del comportamiento de nuestro ESP32.

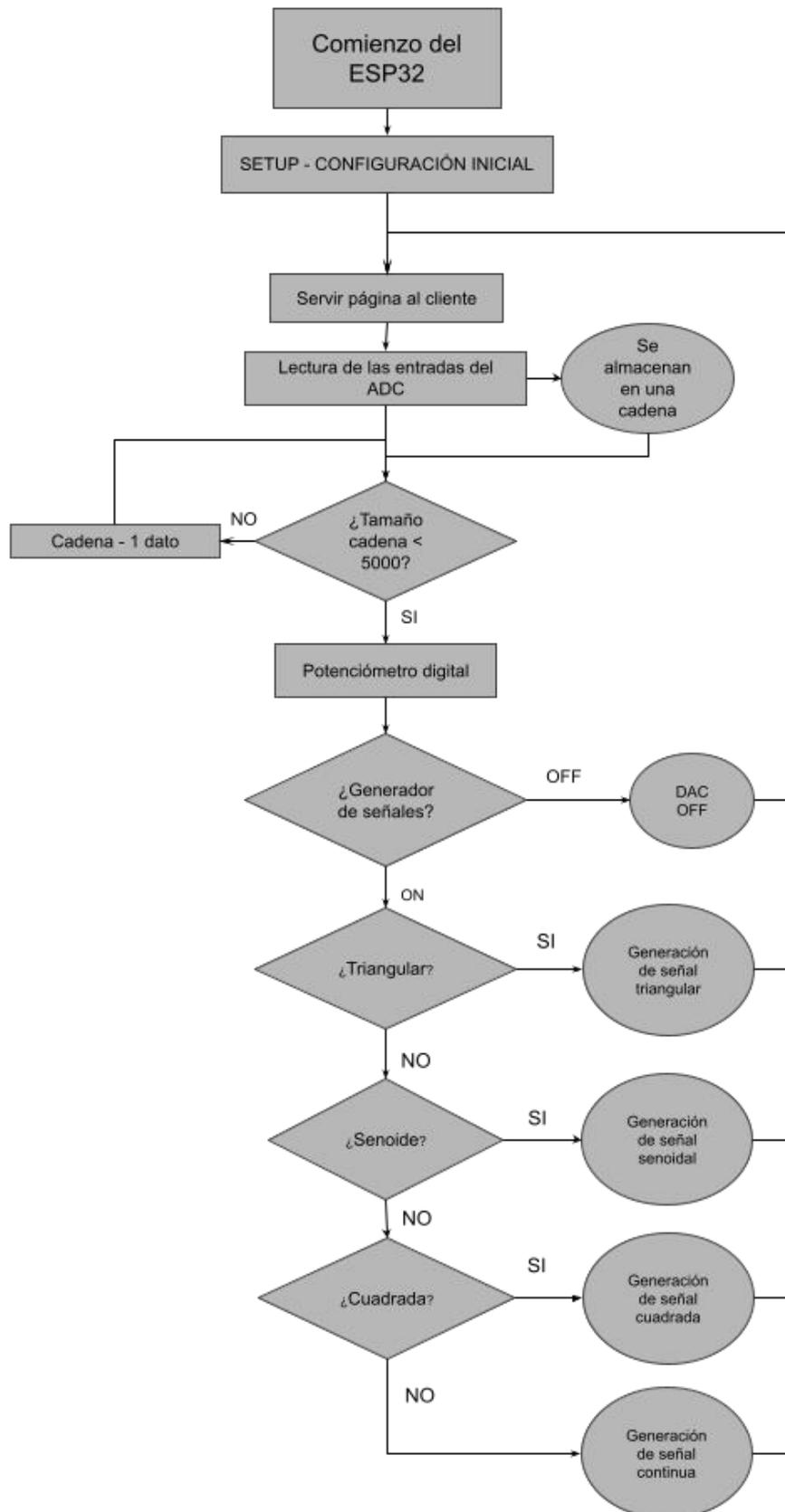


Figura 5.33.- Diagrama de flujo del código del ESP32.

Se puede acceder al código ESP32 en el zip que se adjunta con el resto del trabajo.

### 5.2.2.- Código página.

Para el código de la página hemos utilizado html, un par de estilos de CSS y código JavaScript para aportar la funcionalidad. La idea de este osciloscopio es, como ya hemos mencionado antes es la de obtener un osciloscopio al cual podamos acceder desde cualquier dispositivo con acceso a internet como, por ejemplo, un teléfono móvil. Por lo tanto, es importante que la interfaz sea sencilla, con un diseño más adaptable a un dispositivo móvil y bastante intuitivo.

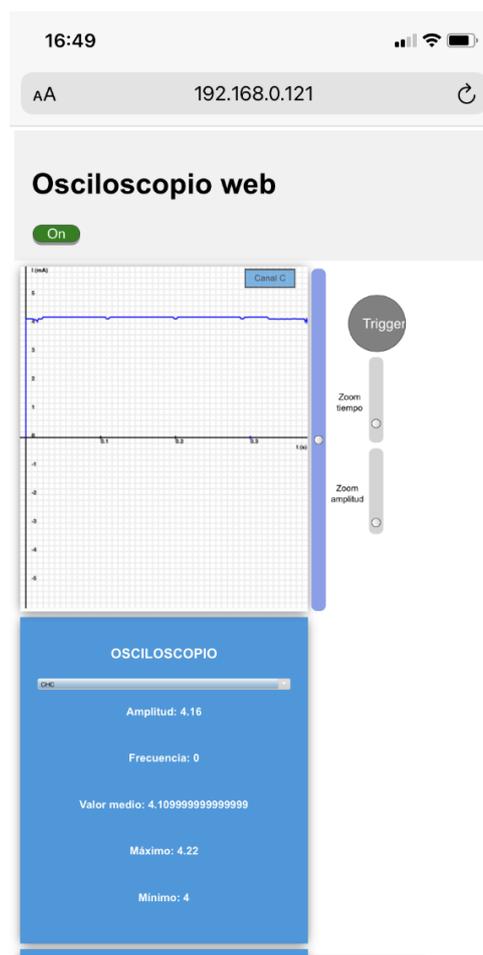


Figura 5.34.- Captura mini-osciloscopio desde un teléfono móvil (a)

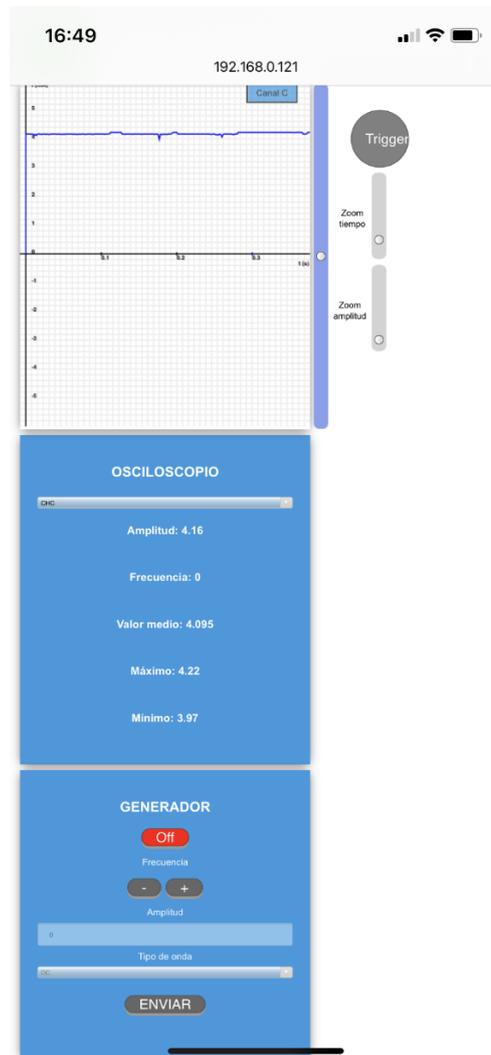


Figura 5.35.- Captura mini-osciloscopio desde un teléfono móvil (b)

Inicialmente nos encontraremos con un osciloscopio apagado, para no realizar peticiones innecesarias. Disponemos de un botón inicial “Off” para poder encender nuestra pantalla del osciloscopio. Una vez la encendamos, se empezarán a realizar peticiones periódicas cada 500ms al ESP32 para ir recibiendo los valores muestreados por nuestro ADC. Al apagarse, estas peticiones se pausan.

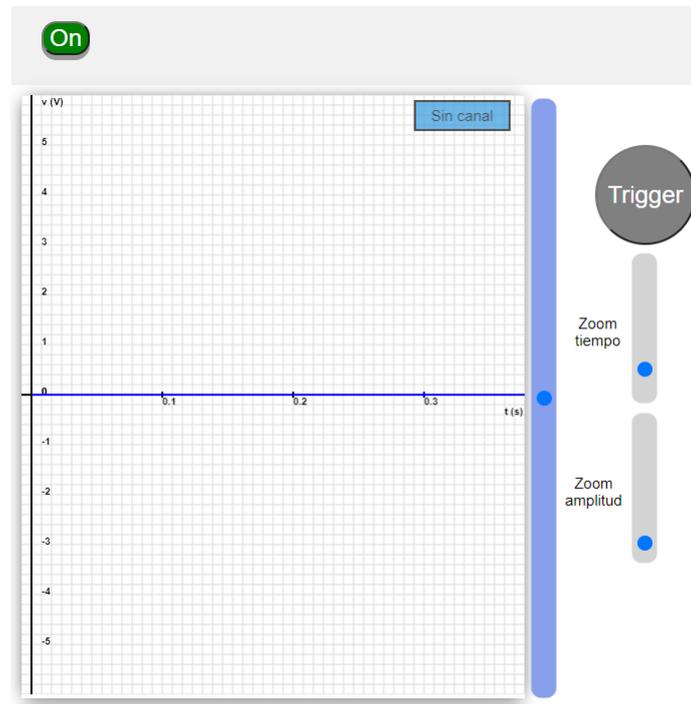


Figura 5.36.- Señal inicial con valor 0 y funciones con valores por defecto.

En primer lugar al encender nuestra pantalla podemos ver unos ejes. Estos ejes trabajan en tiempo y voltaje, pero éste último se cambiará en función del canal que estemos seleccionando, puesto que también podremos medir corriente.

Por otro lado, en la esquina superior derecha de la pantalla podemos encontrar un recuadro que marca “Sin canal”. Este campo mostrará el canal que hemos seleccionado para ver. En el caso de no tener ninguno seleccionado, se muestra ese valor.

Situadas a la derecha de la pantalla tenemos un par de funciones para nuestra señal. Por un lado, tenemos un botón para actuar como trigger, el cual nos parará la señal para poder realizar mediciones. Por otro lado, disponemos de tres sliders: el primero, nos ampliará o reducirá el eje de tiempos; el segundo, actuará igual pero en el eje de la amplitud; y por último, un slider para subir o bajar la referencia de la señal a nuestro antojo.

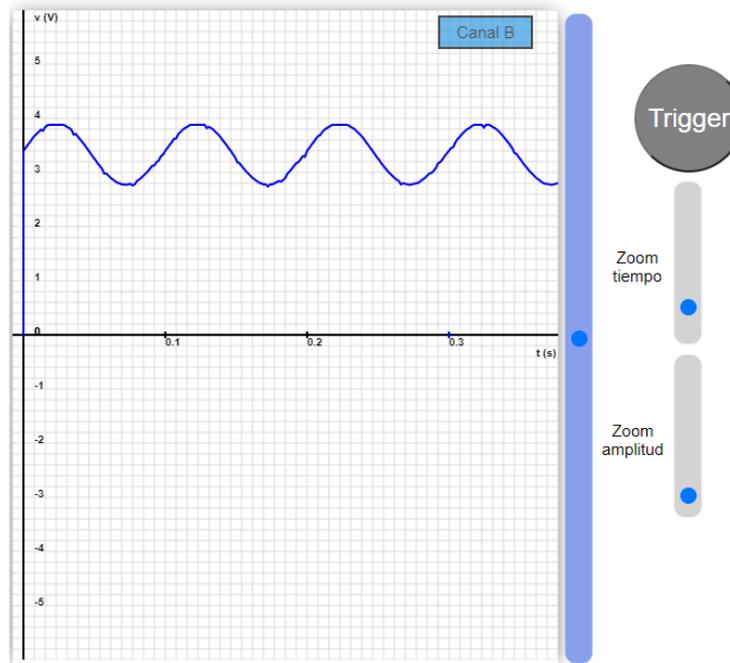


Figura 5.38.- Señal antes de usar alguna función.

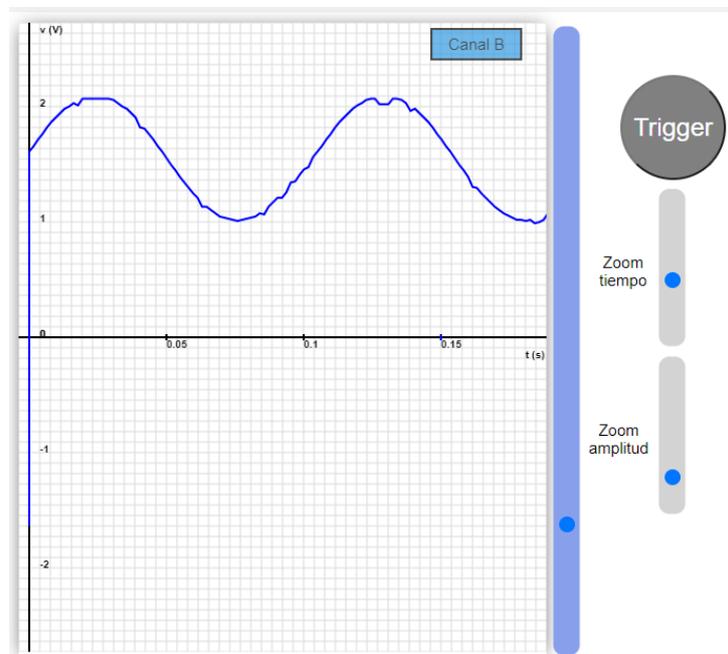


Figura 5.37.- Señal después de aplicar algunas de las funciones.

En la figura 5.37. podemos ver cómo la señal se ha desplazado hacia abajo tras aplicar el tercer slider. También se ha hecho zoom en el eje de la amplitud y se ha aumentado el eje de tiempos. Por último hemos seleccionado el canal B, como podemos ver en el recuadro superior derecha.

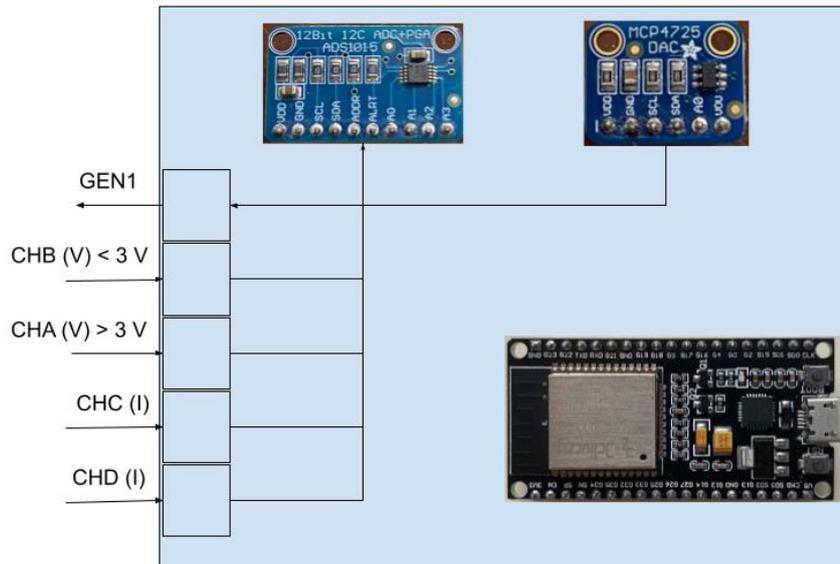


Figura 5.39.- Asignación de canales.

La selección del canal se obtiene en el siguiente apartado de la página, en el cual podemos seleccionar qué señal queremos ver y nos devuelve algunos de los parámetros más importantes de ésta: amplitud, valor máximo, valor mínimo, frecuencia y valor medio. Estos valores se obtienen a partir del intervalo de señal que se está imprimiendo en la pantalla.

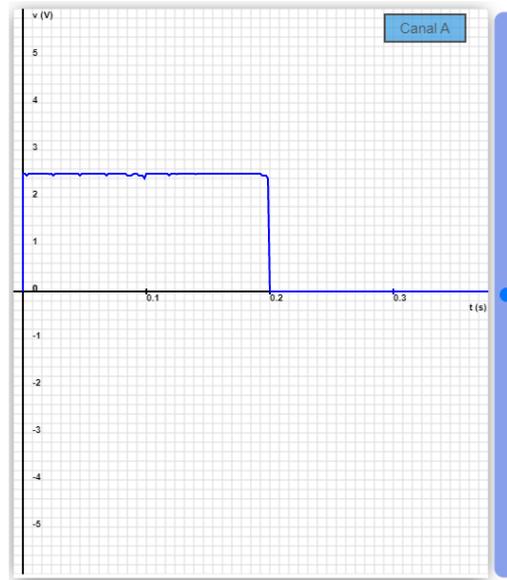


Figura 5.40.- Captura señal en Canal A.



Figura 5.41.- Captura datos osciloscopio.

En el caso de que el osciloscopio esté apagado, el valor de todos estos parámetros pasará automáticamente a cero hasta que se vuelva a encender de nuevo y se realicen nuevas peticiones.

En último lugar nos encontraremos con el generador de señal, inicialmente apagado. Lo apagamos para que, como hemos indicado en el código de nuestro ESP32, no se estén generando señales si el cliente no lo solicita.

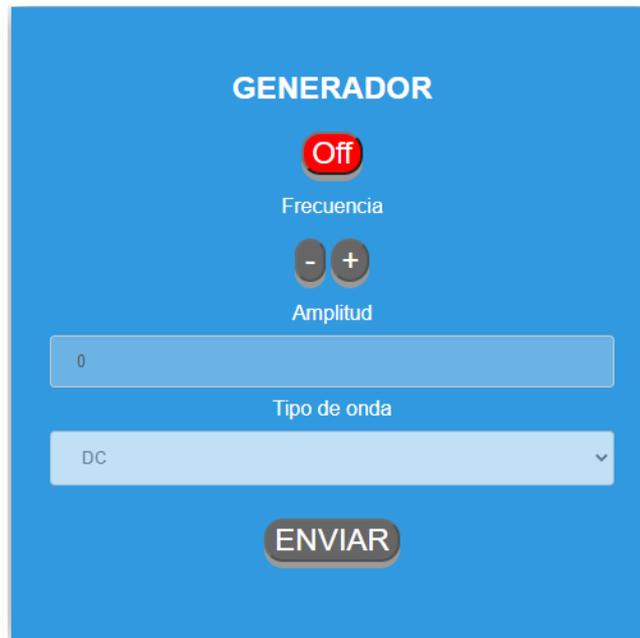


Figura 5.42.- Generador de señales apagado.



Figura 5.43.- Generador de señales encendido.

Una vez encendido el generador podemos seleccionar el tipo de onda (senoidal, cuadrada o continua), su amplitud y su frecuencia. Al pulsar el botón de “ENVIAR” se enviarán estos datos al ESP32. Si se vuelve a apagar de nuevo, no se generará ninguna señal y tampoco se guardarán los últimos datos enviados.

Como nuestro DAC tiene un límite en amplitud para generar señales, se ha decidido bloquear cualquier petición en la cual la amplitud de la señal supere este límite de 2.5V y se notifique del error al cliente. Este caso es para señales senoidales y cuadradas, para el caso de una señal de continua el límite está en 5V.

## 6. Pruebas y resultados.

En este punto se van a realizar varias pruebas acordes con los objetivos buscados en el desarrollo de este proyecto.

### 6.1.- MEDIDA DE VOLTAJE.

Lo primero que vamos a comprobar es el correcto funcionamiento de la medida de señales de voltaje. Como ya hemos mencionado tenemos 2 entradas, organizado como en la figura 5.38.

En primer lugar vamos a probar a realizar la medida de señales de la entrada 1. Para esta prueba vamos a utilizar una pila de 1,5V de continua. El escenario es el siguiente:

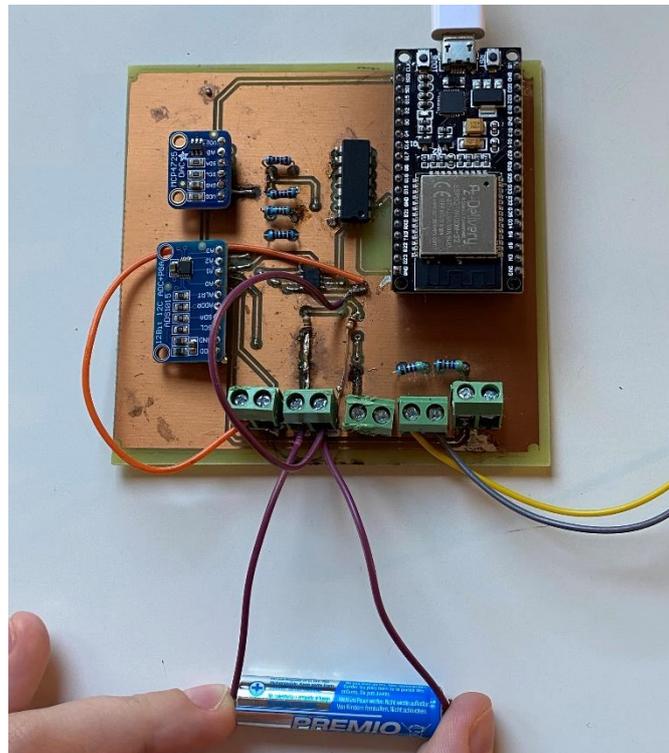


Figura 6.1.- Pila de 1,5V conectada a la entrada del mini-osciloscopio.

Comprobamos el valor de voltaje de la pila a medir:



Figura 6.2.- Pila de 1,5V.

Finalmente comprobamos que el valor que se imprime a la entrada del osciloscopio coincide.

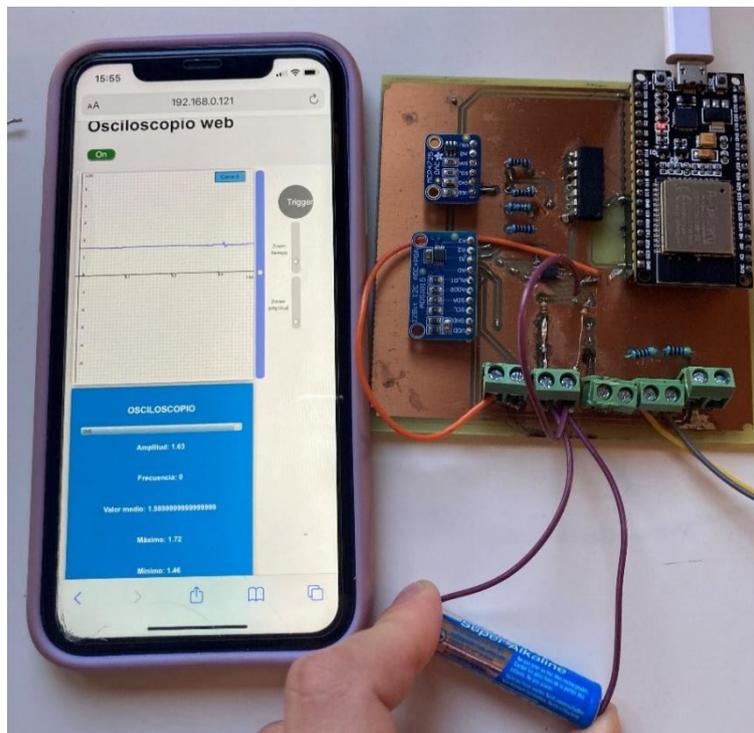


Figura 6.3.- Página web del mini-osciloscopio con el circuito.

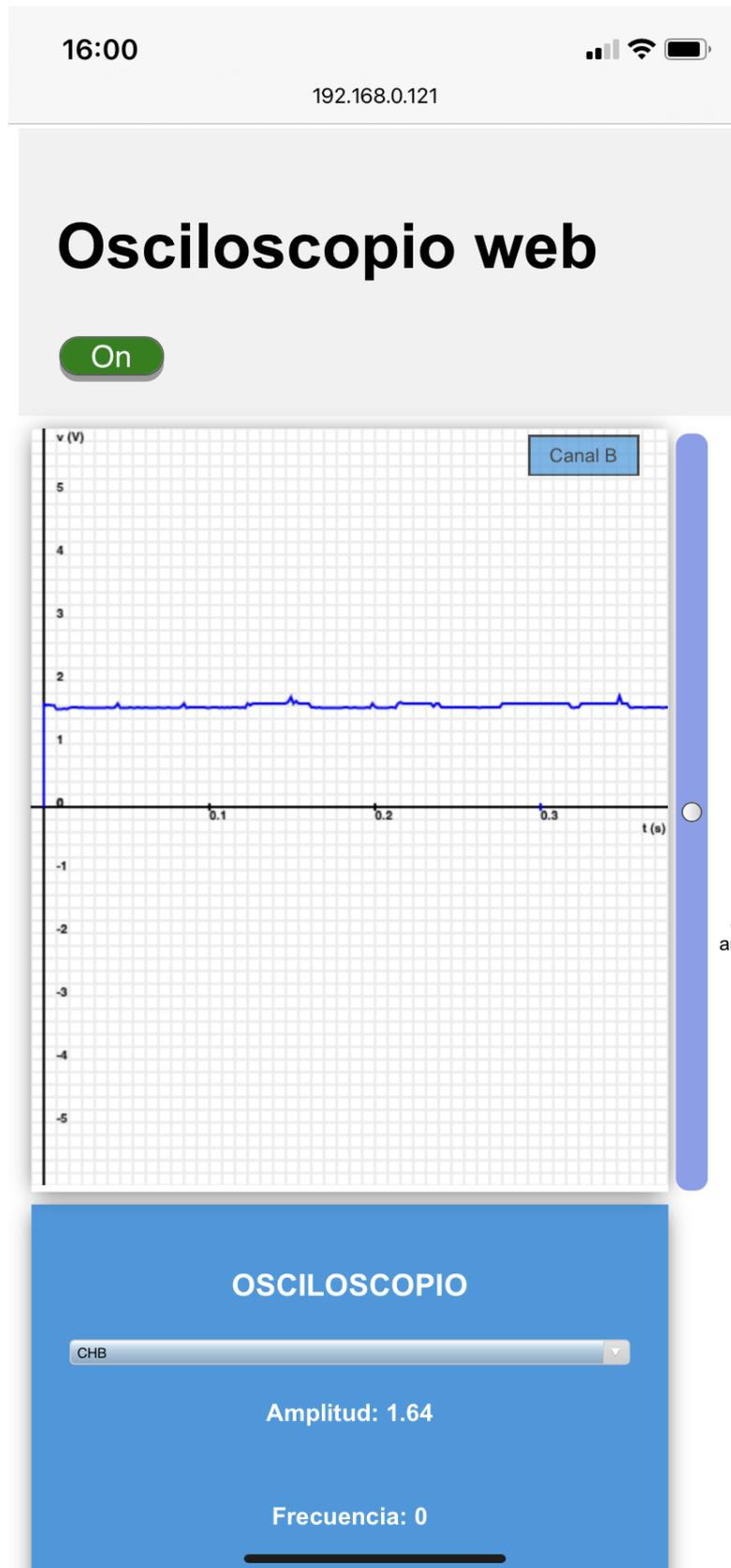


Figura 6.4.- Captura de pantalla de la página en funcionamiento.

Aunque pueda parecer raro, sale 1,6V de amplitud por la sencilla razón de que la pila medida con el polímetro se ve que no tiene exactamente 1,5V, sino más bien cerca de los 1,6V. El resto de desfase se puede deber a un redondeo en el cálculo digital.



Figura 6.5.- Valor real de la pila medida con un polímetro.

A continuación probaremos la entrada de señal de mayor amplitud, usando de ejemplo una señal de 9V. Para ello utilizamos una pila que, para ser más exactos, su valor es el mostrado en la figura 6.6.



Figura 6.6.- Valor de la pila de 9V medido con un polímetro.

Para realizar la medida de esta pila la situación es la mostrada en la figura 6.7.

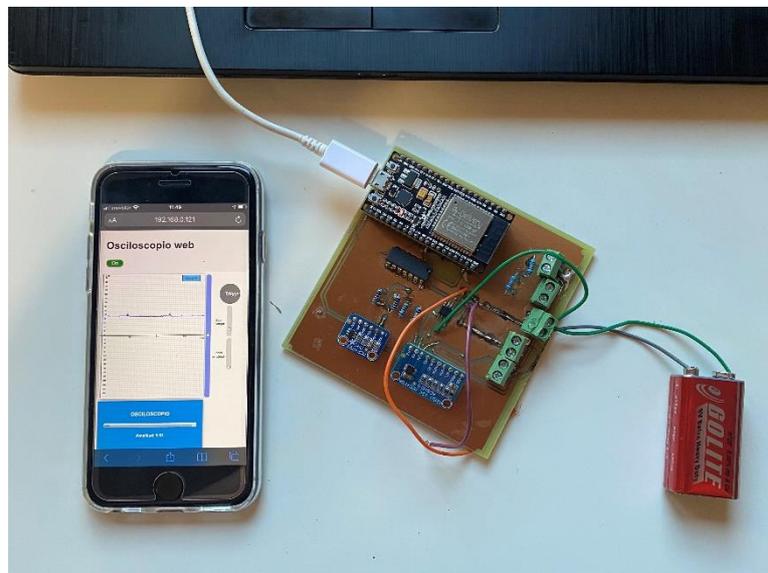


Figura 6.7.- Situación de medida de la pila de 9V.

Conectamos la pila a la entrada de señal óptima para las señales entre 3 y 23 V. Una vez realizada esa conexión, en la página web podemos comprobar que el valor que llega es el correcto.

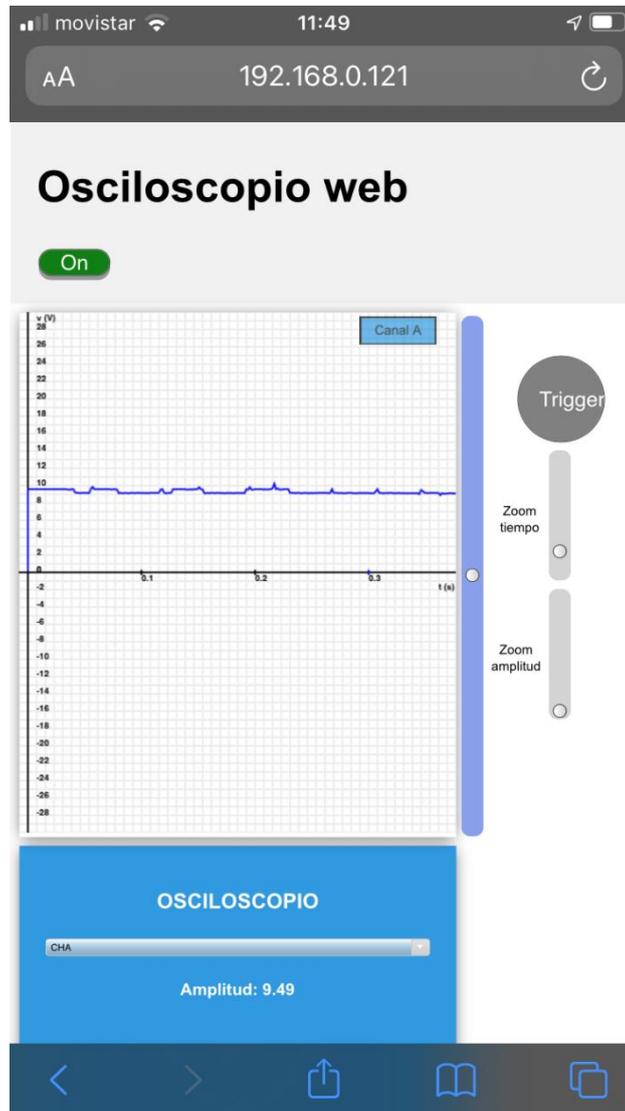


Figura 6.8.- Captura de la página web mostrando el valor medido por el mini-osciloscopio.

Nos muestra un valor de 9,49V, aproximadamente el valor de la pila. Se pierden algunos decimales ya sea por las pérdidas del circuito o por redondeo en el cálculo digital.

## 6.2.- MEDIDA DE CORRIENTE.

Para la prueba de medida de corriente utilizaremos una pila de 1,5V y una resistencia de 1k  $\Omega$ . El circuito sería el siguiente:

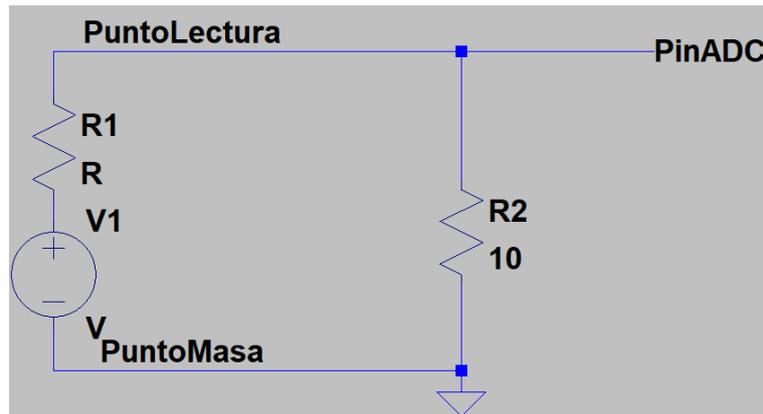


Figura 6.9.- Circuito de prueba en LtSpice.

Para ello realizamos el conexionado con un generador de tensión con valor de 1,5V y una resistencia de 1k $\Omega$ . La situación de medida es la mostrada en la figura 6.10.

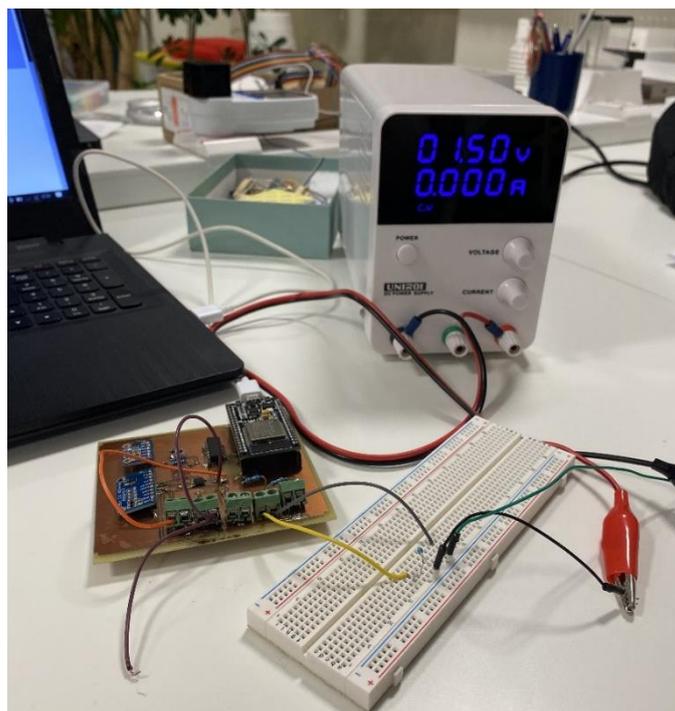


Figura 6.10.- Situación para la medida de corriente.

Una vez realizado este conexionado, el valor que nos imprime en la página es el siguiente:

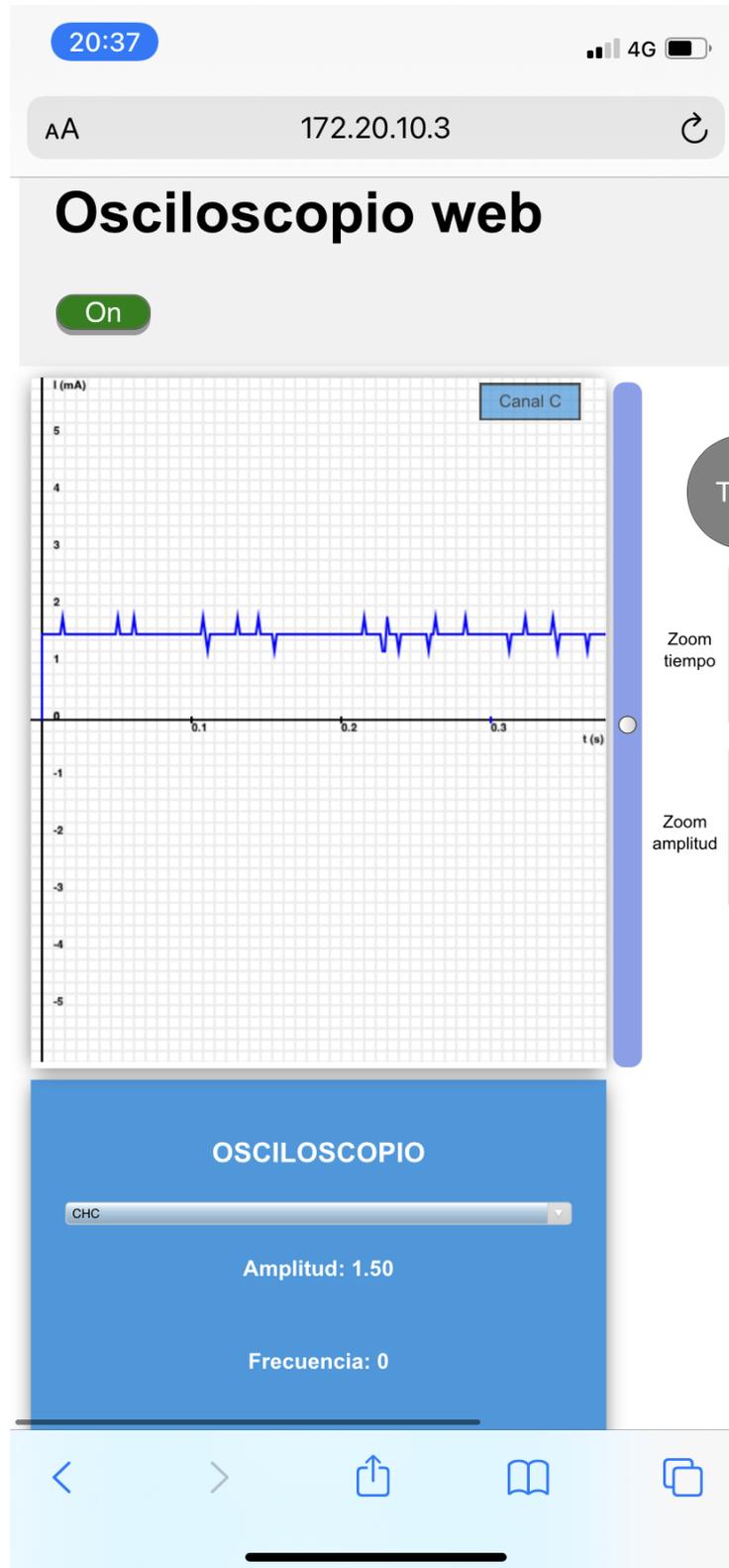


Figura 6.11.- Captura de pantalla de la página web mostrando el valor medido.

Lo que coincide con el valor real de la corriente en el circuito según los cálculos:

$$I(mA) = \frac{V(V)}{R(k\Omega)} = \frac{1,5}{1.000} = 1,5mA$$

### 6.3.- GENERACIÓN DE SEÑALES.

En este apartado vamos a probar el generador de señales que hemos incluido en el mini-osciloscopio. Para las pruebas conectamos los dos cables de salida del DAC a un aparato de medida externo. En este caso se ha hecho uso de un Osciloscopio digital Hantek.

El escenario de medida es el siguiente:



Figura 6.12.- Escenario de medida del generador de señales.

Una vez hechas las conexiones pertinentes, se accede a la página web y se selecciona la señal deseada. Haremos distintas pruebas.

➤ **Señal de continua.**

En esta prueba intentaremos generar una señal de continua de 1,5V y 7V. En primer lugar probamos a generar la señal de 1,5V, para comprobar que no se pierden muchos decimales en la generación digital.

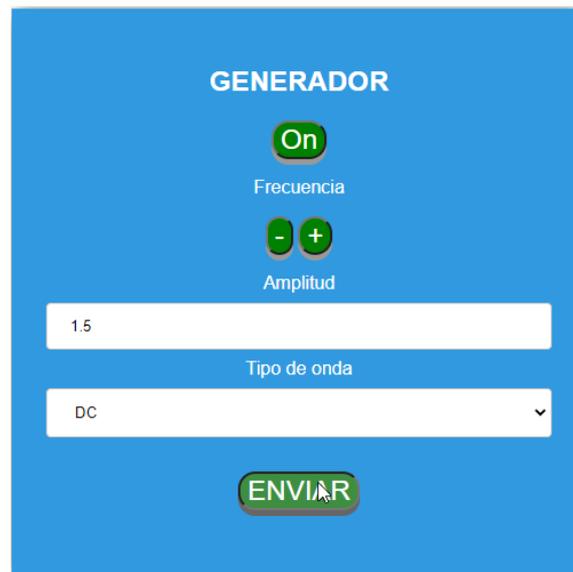


Figura 6.14.- Datos enviados al generador de señales.

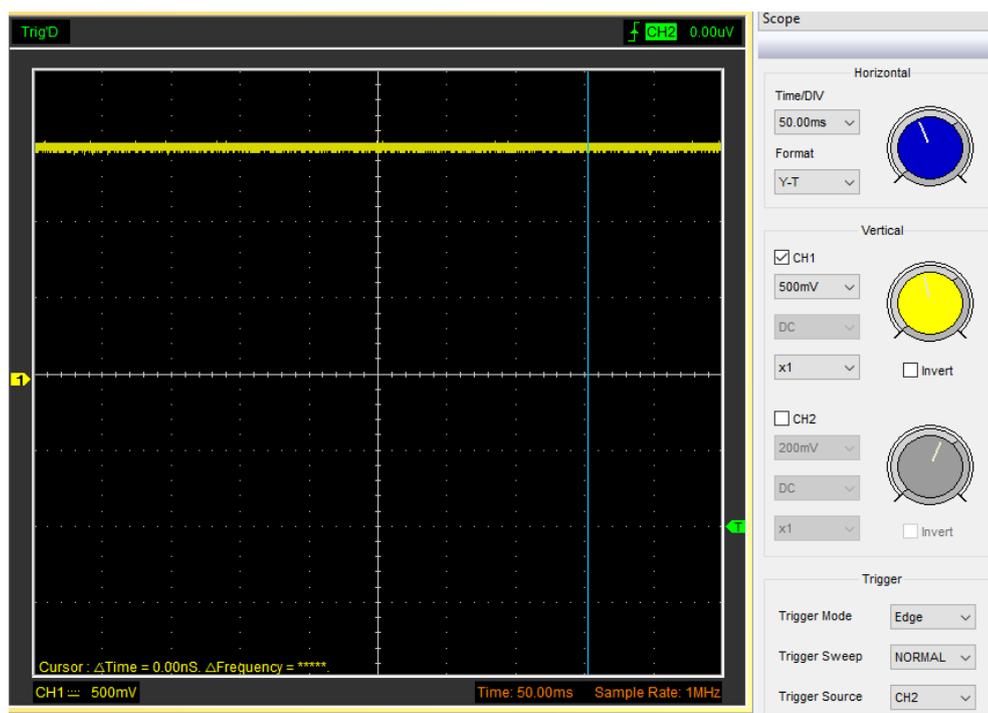


Figura 6.13.- Señal de salida del generador medida con el osciloscopio Hantek.

Podemos comprobar en la figura 6.14 de la aplicación del osciloscopio Hantek que la señal se genera correctamente y sin una notable caída en la amplitud. A continuación probaremos a generar una de 7V, nos dará error puesto que el DAC trabaja entre 0V y la tensión de alimentación de 5V.

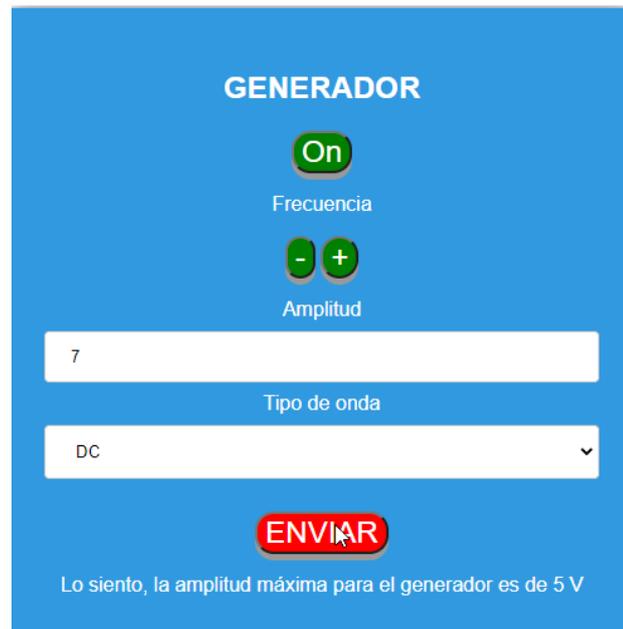


Figura 6.15.- Mensaje de error para señales de continua de más de 5V.

### ➤ Señal senoidal.

A continuación probaremos a generar una senoide. Este tipo de señal, al igual que la cuadrada, solo podrá generarse entre 0 y 2,5V de amplitud. Su valor de voltaje de pico a pico será, en efecto, de 5V como máximo.

En este caso está limitado el máximo a 2,5V. Primero probaremos a generar una señal que esté dentro de nuestro rango y a jugar con los botones de “+” y “-“ para ir variando su frecuencia. Por ejemplo, una señal de 2V de amplitud.



Figura 6.17.- Datos enviados al generador de señales.

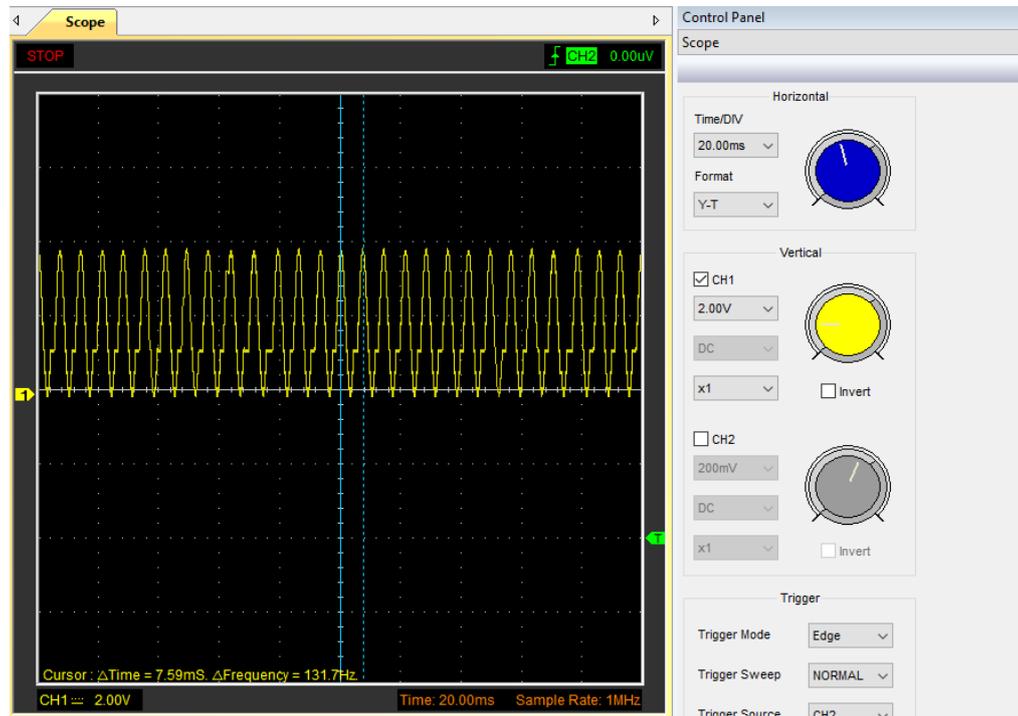


Figura 6.16.- Señal de salida medida por el osciloscopio Hantek.

Esta es la máxima frecuencia a la que podemos generar senoides, de aproximadamente 132 Hz, sin que se pierda del todo la forma natural de la señal. Esto se ha conseguido submuestRANDO, por eso la forma “cuadrada” de la señal.

Con el botón de “-“ podemos ir disminuyendo esa frecuencia:

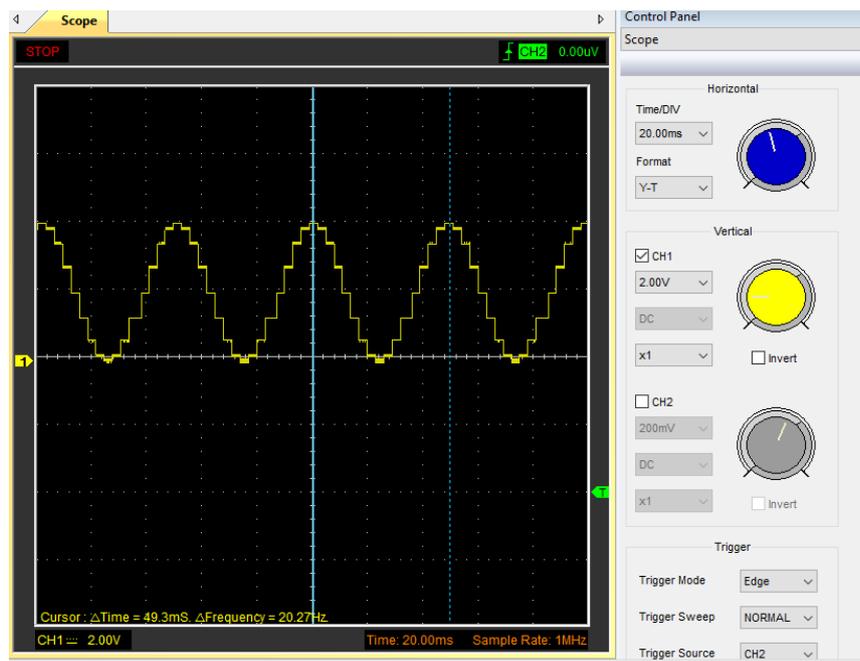


Figura 6.18.- Señal con una frecuencia de 20,27 Hz.

La señal se genera correctamente aunque se aprecian unos escalones que se debe a la forma en la que se ha conseguido disminuir la frecuencia. Esto se ha buscado mediante *delays* que, aunque pueden suponer un problema para generar la señal más acorde a su naturaleza, es el método más sencillo para controlar un aumento o disminución de la frecuencia de la señal.

### ➤ Señal cuadrada.

Esta señal es la que más frecuencia puede alcanzar, puesto que requiere menos datos. Al igual que la anterior, su amplitud máxima será de 2,5V, es decir, un valor de pico a pico de 5V. Probamos a generar una señal de 2,5V.

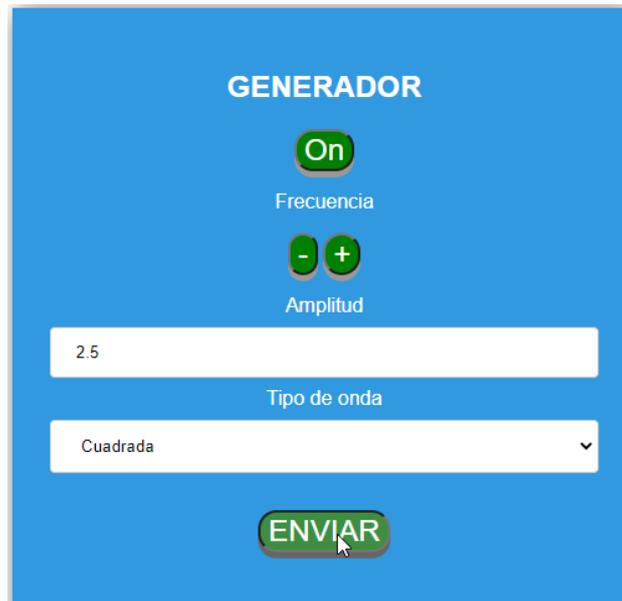


Figura 6.19.- Datos enviados al generador de señales.

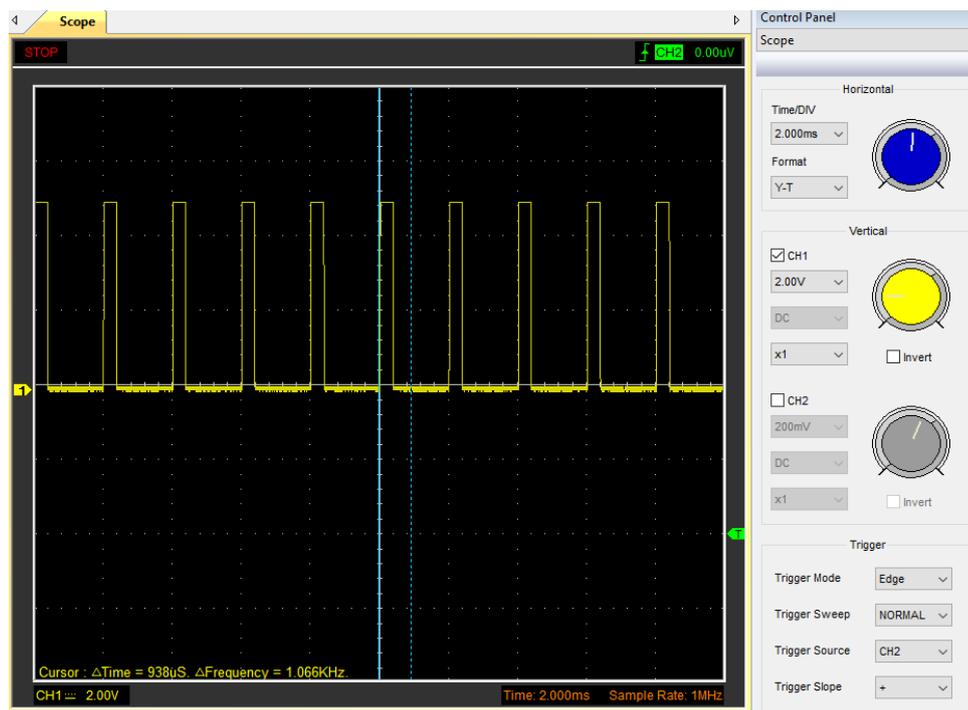


Figura 6.20.- Señal medida por el osciloscopio Hantek.

Como podemos ver en la figura 6.20., a la mayor frecuencia que es de 1kHz, la señal no llega a ser exactamente periódica. Esto se debe a que cuando se establece el valor del DAC en 0V para generar la parte “negativa” de la onda cuadrada, lo que sucede es la vuelta al

inicio del “loop” del ESP32, que procesa mucha información y retrasa la generación de la señal.

Si aumentamos un poco la frecuencia con los botones, llegamos a que el mínimo valor en el que la señal se ve más correcta es a una frecuencia de unos 170Hz, como podemos ver en la figura 6.21.

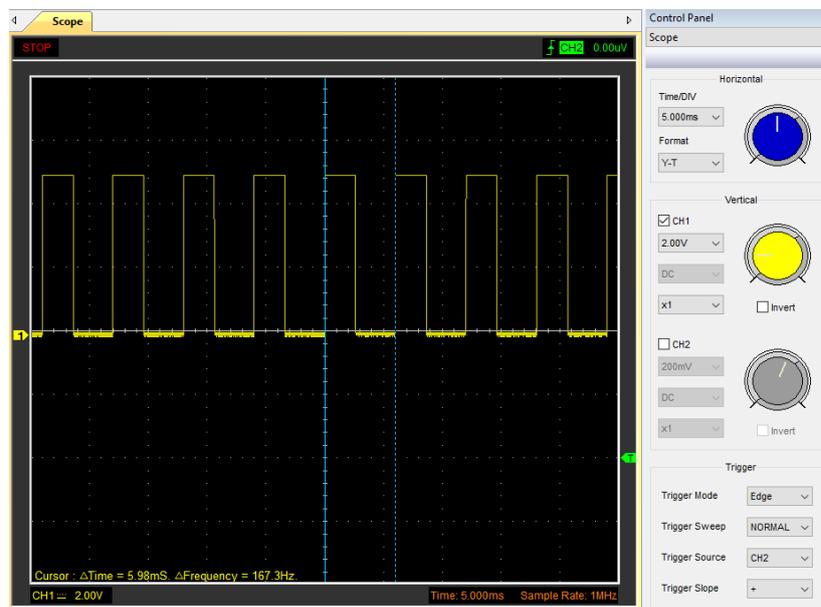


Figura 6.21.- Señal a 170Hz.

Al igual que en el caso de la corriente continua, en la senoide y la onda cuadrada se genera un mensaje de error al intentar generar una señal cuyo valor de pico a pico es superior a los 5V, es decir, con una amplitud de 2,5V:



Figura 6.22.- Mensaje de error para ondas senoidales o cuadradas de más de 2,5 V de amplitud.

En definitiva, tenemos un generador de señales que trabaja con frecuencias máximas de 170Hz para ondas cuadradas y de unos 120Hz para ondas senoidales. Se nos escapa un poco del objetivo inicial, pero teniendo en cuenta la cantidad de datos que se necesitan procesar es un resultado cuanto menos suficiente para el inicio de un proyecto como este.

#### 6.4.- AUTONOMÍA.

Uno de los objetivos que buscamos con este osciloscopio es el de ser un dispositivo fácilmente manejable. Una propiedad que probaremos será la de su autonomía empleando un PowerBank de 5.600mAh.

El escenario de media es el siguiente. Se ha colocado el osciloscopio con un cronómetro y conectado a la batería portátil. También accederemos con el teléfono móvil a nuestra página web y, encenderemos el generador y el osciloscopio simultáneamente, para que esté trabajando al máximo rendimiento.

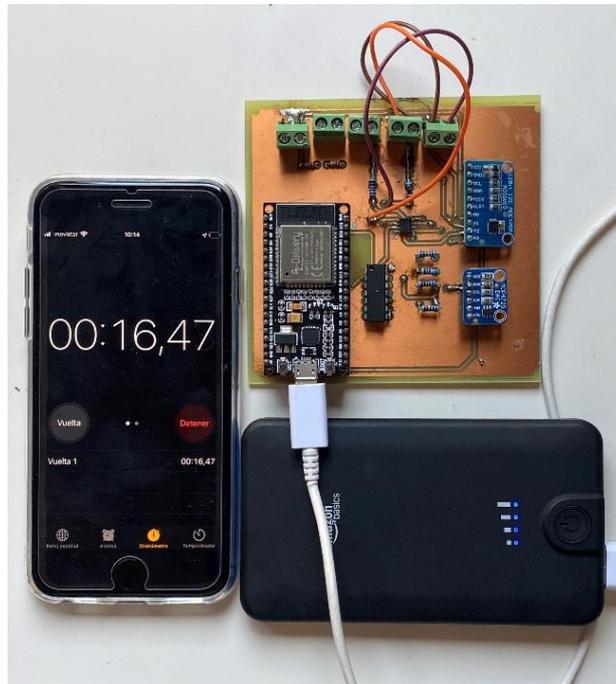


Figura 6.24.- Situación de medida con un PowerBank (a)

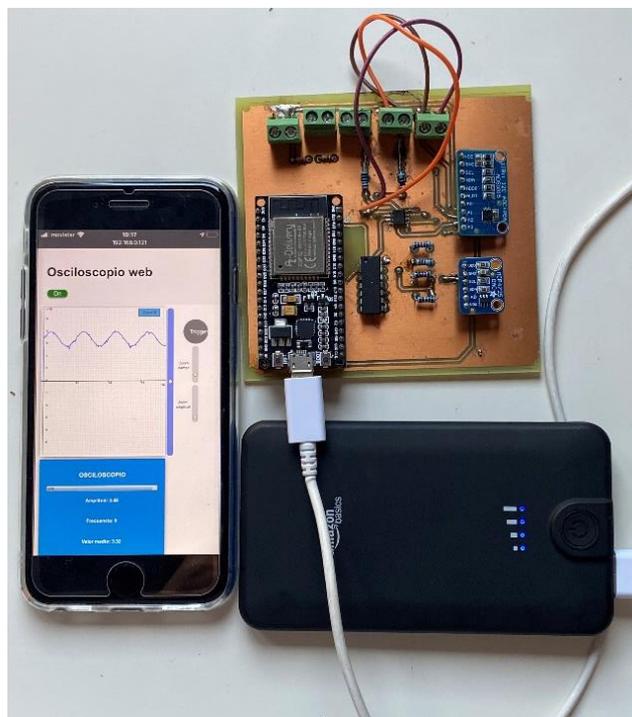


Figura 6.23.- Situación de medida con un PowerBank(b)

La duración total de la autonomía resultó de 31 horas y 23 minutos, lo que resulta muy similar a los cálculos hechos por la siguiente estimación:

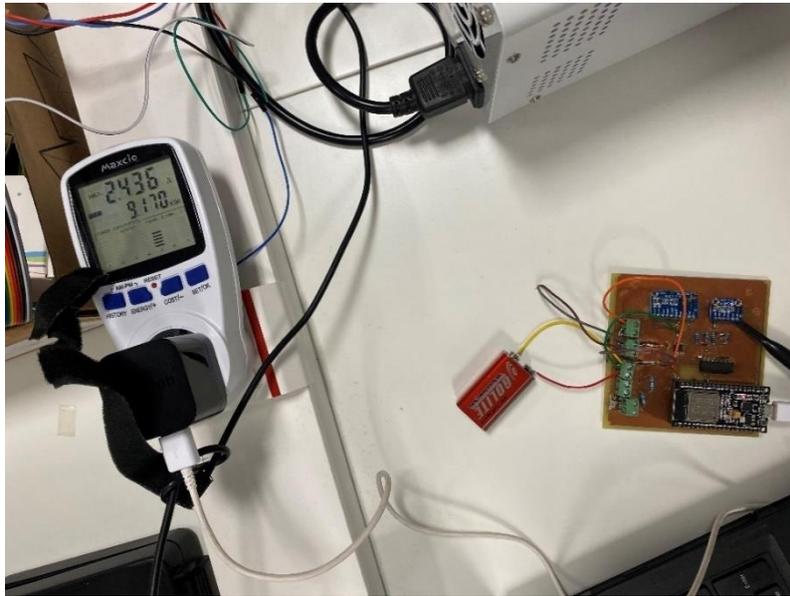


Figura 6.25.- Medida del consumo.

Se midió el consumo de nuestro osciloscopio mientras realizaba medidas y a la vez generaba una señal en el generador, es decir, al máximo nivel de funcionamiento. El valor obtenido de consumo de potencia fue el siguiente:



Figura 6.26.- Consumo de potencia.

Teniendo en cuenta que el voltaje de alimentación es de 5V, obtenemos una corriente de 0,12 A.

$$I \cdot V \Rightarrow I = \frac{P}{V} = \frac{0,8w}{5V} = 0,16 A$$

Por lo tanto, para una batería portátil de 5600mAh, la estimación de duración será la siguiente:

$$Autonomía = \frac{5600 mAh}{160 mA} = 35 horas$$

## 7. Presupuesto.

A continuación recopilaremos el precio de cada uno de los materiales necesarios para el desarrollo de este proyecto. La duración del proyecto según lo establecido en la planificación es de 6 meses, con un total de horas de 300 correspondientes al número de créditos de la asignatura (12). A partir de estos datos, los cálculos del presupuesto son los siguientes:

COMPONENTES	Precio
ADC ADS1015	2,35 €
DAC MCP4724	4,29 €
ESP32	1,50 €
MCP42050	2,35 €
OPA2277	1,71 €
PCB	10,00 €
<b>TOTAL</b>	<b>22,20 €</b>

Tabla 7.1.- Cálculo del presupuesto de los componentes.

HERRAMIENTAS	Precio total	Vida útil (años)	Vida útil (meses)	Duración proyecto (meses)	Precio amortización(€)
ORDENADOR	750,00 €	9	108	6	41,67 €
OSCILOSCOPIO HANTEK DIGITAL USB	70,00 €	10	120	6	3,50 €
SOLDADOR	20,00 €	2	24	6	5,00 €
<b>TOTAL</b>					<b>50,17 €</b>

Tabla 7.2.- Cálculo de la estimación de amortización de los equipos usados.

TRABAJADORA	Tiempo total (horas)	Precio (€/hora)	Total
TRABAJADORA	300,00	25,00	7.500,00 €

Tabla 7.3.- Cálculo de las horas trabajadas y el precio para la trabajadora.

MATERIALES	PRECIO
COMPONENTES	22,20 €
HERRAMIENTAS	50,17 €
TRABAJADORA	7.500,00 €
<b>TOTAL</b>	<b>7.572,37 €</b>

Tabla 7.4.- Cálculo del presupuesto total.

Ascendiendo el precio total a siete mil quinientos setenta y dos euros con treinta y siete céntimos.

## 8. Conclusiones.

Tras estos seis meses de desarrollo de proyecto, las sensaciones son muy positivas puesto que finalmente se ha podido conseguir la mayoría de puntos clave que se buscaban en este trabajo. Un osciloscopio es una herramienta que, al fin y al cabo, he tenido que manejar una gran cantidad de veces a lo largo de este grado y poder desarrollar una máquina tan potente dentro de un dispositivo tan sencillo me deja muy satisfecha con el trabajo realizado.

Por otro lado, me quedan muchas ideas pendientes que me hubiera gustado añadir a este proyecto. En el siguiente apartado de trabajos futuros mencionaré algunos de los que estoy segura que se podrán implementar en este proyecto.

Aunque el proyecto no sea perfecto ni cumpla todos los objetivos buscados al detalle, el resultado final creo que es muy eficiente, reconfortante y sobre todo me dan más ilusión de aprender cada día un poco más.

## 9. Trabajos futuros.

Durante el desarrollo de este proyecto son muchas las ideas que van surgiendo según se van consiguiendo objetivos, muchas de estas ideas quedan fuera del alcance de este trabajo y finalmente no se llevan a cabo. Sin embargo, sí que es importante tenerlas en cuenta puesto que a pesar de que esto sea un proyecto simplificado, a posteriori podría seguir desarrollándose y crecer mucho más.

En primer lugar, en un proyecto futuro basado en este mini-osciloscopio lo primero que habría que abordar es ampliar las capacidades a nivel de ancho de banda, amplitud máxima de lectura, etc. Una forma de hacerlo sería dividir el trabajo del ESP32 en los distintos *cores* que tiene. En ese proyecto este punto se ha intentado pero dio bastantes problemas y se decidió mantenerlo al margen.

En segundo lugar, poder darle uso al potenciómetro. La idea inicial era poder manejar por código el valor del potenciómetro de forma automática. Sin embargo, decidimos simplificarlo e ir más directos a medidas sencillas de realizar y de forma más estática.

También sería muy interesante el modo XY y podría ser algo que aportaría a este osciloscopio una gran característica. También sería interesante y sencillo poder añadir líneas verticales y horizontales en la pantalla de la página para poder marcar puntos de medida de interés.

Por otro lado también podría añadirse la capacidad de mostrar señales en el espectro de frecuencia. Este punto sería muy interesante desarrollarlo en un futuro porque esta función no la tienen todos los osciloscopios y, generalmente, suelen tener un precio bastante elevado.

Es importante hacer hincapié en que dentro de los distintos trabajos futuros que podamos desarrollar de este proyecto, el punto más importante y clave es la búsqueda de las mejores prestaciones dentro de un precio muy reducido. Los osciloscopios son aparatos de medida

muy importantes y su precio actual es muy elevado, por lo que es de vital importancia no perder de vista esta característica que buscamos.

En definitiva, esto es un proyecto inicial al que se le podrían añadir a posteriori una gran cantidad de funcionalidad extra que haría de este dispositivo una máquina bastante interesante y, sobre todo, con una grandísima utilidad a un precio muy asequible.

## 10. Bibliografía.

- [1] D. F. Villafana Gamboa, «Fundamentos de Telecomunicaciones». Accedido: jul. 10, 2021. [En línea]. Disponible en: [https://www.itmerida.mx/panel/posgrado/archivos/mi/Fundamentos%20de%20Telecomunicaciones%20Unidad%201%20\(1\).pdf](https://www.itmerida.mx/panel/posgrado/archivos/mi/Fundamentos%20de%20Telecomunicaciones%20Unidad%201%20(1).pdf)
- [2] «RAE- Definición de frecuencia.», *Real Academia de la Lengua Española*. Accedido: oct. 10, 2021. [En línea]. Disponible en: <https://dle.rae.es/frecuencia>
- [3] «Parametros de la corriente alterna», *Electrónica fácil*. <https://www.electronicafacil.net/tutoriales/Parametros-corriente-alterna.html> (accedido oct. 10, 2021).
- [4] K. A. Polo Jaramillo y W. J. Rodriguez León, «Obtención de los principales parámetros de las señales de tensión y corriente presentes en un sistema monofásico utilizando un sistema digital.», Trabajo Fin de Grado, Universidad industrial de Santander., Bucaramanga, 2015. Accedido: oct. 12, 2021. [En línea]. Disponible en: <http://tangara.uis.edu.co/biblioweb/tesis/2015/156385.pdf>
- [5] «Escalas Lineales y Logarítmicas de Amplitud», *Azimadli*. <http://www.azimadli.com/vibman-spanish/escalaslinealesylogartmicasdeamplitud.htm> (accedido oct. 25, 2021).
- [6] «dB: What is a decibel?», *UNSW Physclips*. <https://www.animations.physics.unsw.edu.au/jw/dB.htm> (accedido oct. 10, 2021).
- [7] I. Bosch Roig, J. Gosálvez, R. Miralles, y L. Vergara Domínguez, *Señales y sistemas. Teoría y problemas.*, 1.<sup>a</sup> ed. Universitat Politècnica de València, 2015. Accedido: oct. 12, 2021. [En línea]. Disponible en: [https://gdocu.upv.es/alfresco/service/api/node/content/workspace/SpacesStore/5a1651eb-f46e-4752-995a-125a77bf045e/TOC\\_0377\\_04\\_01.pdf?guest=true](https://gdocu.upv.es/alfresco/service/api/node/content/workspace/SpacesStore/5a1651eb-f46e-4752-995a-125a77bf045e/TOC_0377_04_01.pdf?guest=true)
- [8] «LOS PINES CUASI ANALÓGICOS», *Prometec*. <https://www.prometec.net/pines-cuasi-analogicos/> (accedido oct. 27, 2021).
- [9] Luis, «Formas de onda, ondas constantes, periódicas y no periódicas», *Ingeniería Electronica.org*, mar. 28, 2016. <https://ingenieriaelectronica.org/formas-de-onda-ondas-constantes-periodicas-y-no-periodicas/> (accedido oct. 15, 2021).

- [10] «Introducción Tutorial Generador de Funciones». <http://www.ing.unp.edu.ar/electronica/asignaturas/ee016/tutoriales/generador/generador.htm> (accedido oct. 15, 2021).
- [11] «Osciloscopios: historia y clasificación», nov. 24, 2015. <https://toolboom.com/es/articles-and-video/oscilloscopes-history-and-classification/> (accedido jul. 06, 2021).
- [12] «Función de atenuación. Ancho de banda y frecuencia de corte: Concepto de filtro.», *Universitas Miguel Hernandez*. [https://repositorio.innovacionumh.es/Proyectos/P\\_19/Tema\\_5/UMH\\_03.htm](https://repositorio.innovacionumh.es/Proyectos/P_19/Tema_5/UMH_03.htm) (accedido oct. 20, 2021).
- [13] Juan Carlos, «EL OSCILOSCOPIO», *La web de Juanki - Universidad de Granada*. <https://www.ugr.es/~juanki/osciloscopio.htm> (accedido jul. 06, 2021).
- [14] «Relación señal/ruido», *Megafonia y Sonorización Aleksandar Penchev*. <https://sites.google.com/site/megafoniaysonorizacionalekspen/10-megafonia-y-sonorizacion/04-equipos-y-transductores-electroacusticos-amplificador-y-altavoces/4-2-caracteristicas-tecnicas-de-los-amplificadores/4-2-5-relacion-senal-ruido> (accedido oct. 27, 2021).
- [15] «Osciloscopio analógico - Madell», *Direct Industry*. <https://www.directindustry.es/prod/madell-technology/product-66775-1703662.html> (accedido oct. 27, 2012).
- [16] «Osciloscopio ¿Qué es? y ¿Cómo se usa?», *HETPRO*. <https://hetpro-store.com/TUTORIALES/osciloscopio/> (accedido oct. 27, 2021).
- [17] «DEFINICION, USO Y TIPOS DE OSCILOSCOPIOS». <https://www.equiposylaboratorio.com/portal/articulo-ampliado/definicion-uso-y-tipos-de-oscilloscopios> (accedido jul. 06, 2021).
- [18] A. Llorente, «Aplicaciones para osciloscopios digitales», *Instrumentacion hoy*, oct. 23, 2013. <https://www.instrumentacionhoy.com/aplicaciones-para-oscilloscopios-digitales/> (accedido oct. 20, 2021).
- [19] «El osciloscopio». Accedido: oct. 07, 2021. [En línea]. Disponible en: <http://webs.ucm.es/centros/cont/descargas/documento14637.pdf>
- [20] R. S. Castro Rios, «Estudio e implementación de un convertidor analógico digital y digital analógico Sigma Delta», Proyecto Final de Grado, Universidad Politécnica de

- Catalunya, Escola d'Enginyeria de Terrassa, 2015. Accedido: jun. 29, 2021. [En línea]. Disponible en: <https://core.ac.uk/download/pdf/46111717.pdf>
- [21] «Aliasing», *Ingeniería de telecomunicaciones*, sep. 18, 2015. <http://aiturrih.com/tratamiento-de-senales/aliasing/> (accedido sep. 15, 2021).
- [22] J. J. Cerdá, «DISEÑO DE UN SAR ADC PARA APLICACIONES MULTICANAL CON TECNOLOGÍA CMOS DE 180nm», Trabajo Fin de Grado, Universitat Politècnica de València, Valencia, 2020. Accedido: oct. 05, 2021. [En línea]. Disponible en: <https://riunet.upv.es/bitstream/handle/10251/156879/Juan%20-%20Dise%C3%B1o%20de%20un%20convertidor%20anal%C3%B3gico%20digital%20capacitivo%20de%20aproximaciones%20sucesivas%20para%20apl....pdf?sequence=1>
- [23] F. Miyara, «Conversores D/A y A/D». 2004. Accedido: jun. 27, 2021. [En línea]. Disponible en: <https://www.fceia.unr.edu.ar/enica3/da-ad.pdf>
- [24] «Quantization (signal processing)». Accedido: oct. 22, 2021. [En línea]. Disponible en: [https://en.wikipedia.org/wiki/Quantization\\_\(signal\\_processing\)#Quantization\\_error\\_models](https://en.wikipedia.org/wiki/Quantization_(signal_processing)#Quantization_error_models)
- [25] L. F. Monter Martínez y D. I. Rios Casañas, *Comunicaciones en Redes, Digital*. Accedido: oct. 20, 2021. [En línea]. Disponible en: <http://cidecame.uaeh.edu.mx/lcc/mapa/PROYECTO/libro27/index.html>
- [26] «¿Cómo diseña un circuito para implementar la técnica de codificación de línea bipolar?», abr. 23, 2015. <https://electronica.guru/questions/52320/como-disena-un-circuito-para-implementar-la-tecnica-de-codif>
- [27] J. A. Rubia Mena, «Nociones básicas sobre adquisición de señales». dic. 2002. Accedido: sep. 05, 2021. [En línea]. Disponible en: <http://redeya.bytemaniacos.com/electronica/tutoriales/adatos/adquisicion.html>
- [28] «PWM Image». <https://www.researchgate.net/profile/Jimy-Cortes/publication/277184041/figure/fig2/AS:391896049438721@1470446707113/Figura-3-PWM-aplicado-a-una-senal-Sinusoidal-2.png>
- [29] R. Ramos Lara, *Sistemas digitales de instrumentación y control*. 2007. Accedido: oct. 20, 2021. [En línea]. Disponible en: <https://upcommons.upc.edu/bitstream/handle/2117/6122/TEMA2.pdf>

- [30] R. E-Marmolejo, «Microcontrolador – qué es y para que sirve», *Microcontrolador – qué es y para que sirve*. <https://hetpro-store.com/TUTORIALES/microcontrolador/> (accedido jul. 07, 2021).
- [31] «¿Qué es un microcontrolador? Introducción para principiantes». <https://descubrearduino.com/microcontrolador/>
- [32] J. Roca, «Microcontroladores, arquitectura y usos de este tipo de CPU», *Todos los chips tienen uno pero, ¿sabes qué son los microcontroladores?*, mar. 16, 2021. <https://hardzone.es/reportajes/que-es/microcontroladores/> (accedido oct. 07, 2021).
- [33] «Memoria Flash», *Escuela Técnica Superior de Ingeniería y Sistemas de Telecomunicación*. <https://www.etsist.upm.es/estaticos/ingeniatic/index.php/tecnologias/item/511-memoria-flash%3Ftmpl=component&print=1.html> (accedido oct. 18, 2021).
- [34] «TIPOS DE MEMORIA RAM», *Area Tecnológica*. <https://www.areatecnologia.com/TECNOLOGIA%20EN%20IMAGENES/TIPOS%20DE%20MEMORIAS%20RAM.htm> (accedido oct. 20, 2021).
- [35] «¿Qué es un microcontrolador?», *Microcontroladores*. <http://sherlin.xbot.es/microcontroladores/introduccion-a-los-microcontroladores/que-es-un-microcontrolador> (accedido sep. 10, 2021).
- [36] «Transmisión de datos en Serie y Paralelo», *Pc-Solucion*, abr. 03, 2018. <https://pc-solucion.es/2018/04/03/transmision-de-datos-en-serie-y-paralelo/> (accedido oct. 21, 2021).
- [37] F. Leon, «Buses de datos en electrónica Paralelo, UART, SPI, I2C», *Dynamo Electronics*, abr. 28, 2021. <https://dynamoelectronics.com/buses-de-datos-en-electronica-paralelo-uart-spi-i2c/> (accedido oct. 22, 2021).
- [38] A. Pini, «Por qué y cómo usar la interfaz periférica serial para simplificar las conexiones entre distintos dispositivos», *Digi-Key*, feb. 14, 2019. <https://www.digikey.es/es/articles/why-how-to-use-serial-peripheral-interface-simplify-connections-between-multiple-devices> (accedido oct. 22, 2021).
- [39] «I2C – Puerto, Introducción, trama y protocolo», *Hetpro*. <https://hetpro-store.com/TUTORIALES/i2c/> (accedido oct. 22, 2021).
- [40] «HTML: Lenguaje de etiquetas de hipertexto», *HTML*. <https://developer.mozilla.org/es/docs/Web/HTML> (accedido sep. 05, 2021).

- [41] G. B., «¿Qué es HTML? Explicación de los fundamentos del Lenguaje de marcado de hipertexto», *Hostinger tutoriales*, ago. 09, 2021. <https://www.hostinger.es/tutoriales/que-es-html> (accedido sep. 05, 2021).
- [42] «JavaScript», *MDN Web Docs*. <https://developer.mozilla.org/es/docs/Web/JavaScript> (accedido sep. 06, 2021).
- [43] «JavaScript», *Wikipedia*. Accedido: sep. 04, 2021. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/JavaScript>
- [44] R. Ramos, «¿Qué es JavaScript y para qué sirve?», *Soy Rafa Ramos*. <https://soyrafaramos.com/que-es-javascript-para-que-sirve/> (accedido sep. 05, 2021).
- [45] «CSS», *MDN Web Docs*. <https://developer.mozilla.org/es/docs/Web/CSS> (accedido sep. 07, 2021).
- [46] «¿Qué es CSS? Definición y aplicación.», jul. 12, 2021. <https://www.ionos.es/digitalguide/paginas-web/disenio-web/que-es-css/> (accedido sep. 07, 2021).
- [47] «Ajax/¿Qué es AJAX?», en *Ajax*, 2011. [En línea]. Disponible en: [https://es.wikibooks.org/wiki/Ajax/%C2%BFQu%C3%A9\\_es\\_AJAX%3F](https://es.wikibooks.org/wiki/Ajax/%C2%BFQu%C3%A9_es_AJAX%3F)
- [48] J. Díaz, «Ajax y Websockets guía completa», *EDteam*, 2019. <https://ed.team/blog/ajax-y-websockets-guia-completa> (accedido sep. 20, 2021).
- [49] «Apuntes sobre Ajax USAL». [En línea]. Disponible en: <https://www.um.es/docencia/barzana/DAWEB/2017-18/Apuntes-sobre-ajax-USAL.pdf>
- [50] G. B., «¿Qué es AJAX y cómo funciona?», *Hostinger tutoriales*, jul. 20, 2021. <https://www.hostinger.es/tutoriales/que-es-ajax> (accedido sep. 20, 2021).
- [51] «WebSockets, ¿qué son y cómo funcionan?», *WebSockets, ¿qué son y cómo funcionan?*, nov. 25, 2019. <https://fp.uoc.fje.edu/blog/websockets-que-son-como-funcionan/> (accedido oct. 11, 2021).
- [52] *Learning WebSockets*. FREE eBook. Accedido: oct. 10, 2021. [En línea]. Disponible en: <https://riptutorial.com/Download/websockets.pdf>
- [53] «La conectividad Inalámbrica: un enfoque para el alumno», ene. 2010, [En línea]. Disponible en: <https://www.feandalucia.ccoo.es/docu/p5sd6785.pdf>
- [54] J. L. Camargo Olivares, «MODELO DE COBERTURA PARA REDES INALÁMBRICAS DE INTERIORES», Proyecto Final de Grado, Universidad de Sevilla, Sevilla, 2009. Accedido: oct. 20, 2021. [En línea]. Disponible en:

[http://bibing.us.es/proyectos/abreproy/11761/fichero/Volumen1%252F5-](http://bibing.us.es/proyectos/abreproy/11761/fichero/Volumen1%252F5-Cap%252FADtulo1+-+Introducci%C3%B3n+a+las+redes+inal%C3%A1mbricas.pdf)

[Cap%252FADtulo1+-+Introducci%C3%B3n+a+las+redes+inal%C3%A1mbricas.pdf](http://bibing.us.es/proyectos/abreproy/11761/fichero/Volumen1%252F5-Cap%252FADtulo1+-+Introducci%C3%B3n+a+las+redes+inal%C3%A1mbricas.pdf)

[55] E. García, «Wi-Fi vs. Bluetooth: diferencias, ventajas y desventajas», *Adsl Zone*, mar. 11, 2021. <https://www.adslzone.net/reportajes/wifi/wifi-vs-bluetooth/> (accedido oct. 20, 2021).

[56] M. Patricio, «Conceptos y terminología utilizada en Circuitos Impresos (PCB)». <http://www.pcb.electrosoft.cl/04-articulos-circuitos-impresos-desarrollo-sistemas/01-conceptos-circuitos-impresos/conceptos-circuitos-impresos-pcb.html> (accedido sep. 26, 2021).

[57] J. A. Castillo, «Qué es una PCB o Placa de Circuito Impreso. Uso, cómo se fabrica», feb. 11, 2019. <https://www.profesionalreview.com/2019/02/11/pcb-que-es/> (accedido sep. 26, 2021).

[58] A. Laverde, «¿PCB qué es y para que sirve?», *Diseño con normas y certificaciones*, mar. 17, 2017. <https://www.aldestatec.com/blog-diseno-con-normas-y-certificaciones/pcb-que-es-y-para-que-sirve/> (accedido sep. 27, 2021).

[59] M. Santos, «PCB: Qué es y por qué son tan importantes en los dispositivos electrónicos», mar. 17, 2018. <https://hardzone.es/2018/03/17/pcb-importancia-dispositivos-electronicos/> (accedido sep. 27, 2021).

[60] «ESP32» [En línea]. Disponible en: <http://esp32.net/>

[61] C. Cinjordiz, «NodeMCU ESP32.», *Infotec.net*, may 27, 2017. <https://www.infootec.net/nodemcu-esp32/> (accedido jun. 10, 2021).

[62] «OPAx277 High Precision Operational Amplifiers datasheet (Rev. B)». Accedido: oct. 15, 2021. [En línea]. Disponible en: <https://www.ti.com/lit/ds/symlink/opa2277.pdf?ts=1634508925219>

[63] «¿Qué son los componentes SMD?», *Surtel electrónica*, abr. 17, 2019. <https://www.surtel.es/blog/que-son-los-componentes-smd/> (accedido oct. 15, 2021).

[64] «Ultra-Small, Low-Power, 12-Bit ADC with Internal Reference (Rev. C)». Texas Instruments, may 2009. Accedido: oct. 05, 2021. [En línea]. Disponible en: <https://www.openhacks.com/uploadsproductos/ads1015.pdf>

[65] «Single/Dual Digital Potentiometer with SPI™ Interface». Accedido: oct. 05, 2021. [En línea]. Disponible en: <https://ww1.microchip.com/downloads/en/DeviceDoc/11195c.pdf>