



Universidad de Extremadura
Escuela Politécnica

Grado en Ingeniería de Sonido e Imagen
en Telecomunicación

Trabajo Fin de Grado

Diseño de sonómetro de medición continua
con conectividad Wi-Fi

Jaime Borrallo Rivera
Febrero 2018



Universidad de Extremadura

Escuela Politécnica

Grado en Ingeniería de Sonido e Imagen en
Telecomunicación

Trabajo Fin de Grado

**Diseño de sonómetro de medición continua
con conectividad Wi-Fi**

Autor: Jaime Borrallo Rivera

Tutor: Antonio Gordillo Guerrero

Co-Tutor/es: Pedro Atanasio Moraga

Tribunal Calificador

Presidente: Juan Miguel Barrigón Morillas

Secretario: Horacio Gonzalez Velazco

Vocal: Marino Linaje Trigueros

Índice general

1. Introducción	1
1.1. Objetivos	3
1.1.1. Objetivo principal	3
1.1.2. Objetivos específicos	3
1.2. Limitaciones	3
1.3. Antecedentes/Estado del arte	4
2. Conceptos generales	7
2.1. Sonómetros	7
2.1.1. Partes de un sonómetro	7
2.1.1.1. Micrófono	8
2.1.1.2. Amplificador	8
2.1.1.3. Redes de ponderación	8
2.1.1.4. Ponderación temporal	10
2.1.2. Tipos de sonómetros	10
2.1.2.1. Sonómetros generales	10
2.1.2.2. Sonómetros integradores-promediadores	10
2.2. Parámetro utilizado para la evaluación y gestión del ruido ambiental según la Directiva 2002/49/CE del Parlamento Europeo y del Consejo de 25 de junio de 2002	11
2.2.1. Nivel día-tarde-noche L_{den}	11
2.3. Tarjeta de desarrollo	12
2.3.1. Arduino	12
2.3.2. Wemos	13
3. Metodología	16
3.1. Desarrollo experimental	16
3.2. Desarrollo técnico	17
3.2.1. Diseño y desarrollo del sonómetro	17

3.2.1.1.	Diseño y desarrollo del pre amplificador	18
3.2.1.2.	Diseño y desarrollo de la red de ponderación	21
3.2.1.3.	Diseño y desarrollo del rectificador de media onda y circuito RC integrador	23
3.2.1.4.	Diseño y desarrollo del amplificador logarítmico	26
3.2.1.5.	Alimentación del circuito	29
3.2.2.	Diseño de PCB con Kicad	30
3.2.3.	Diseño 3D del soporte para la PCB y la alimentación	33
3.2.4.	Consideraciones a tener en cuenta para la programación del dispositivo Wemos	34
3.2.5.	Calibración del sonómetro	35
4.	Análisis y resultados	39
5.	Conclusiones y líneas futuras	43
	Anexos	46
A.	Presupuesto	48
B.	Código de programación en entorno Arduino	49
C.	Diagrama de Gantt	54

Índice de tablas

2.1. Tabla comparativa Wemos D1 Mini vs Arduino Due.	14
3.1. Especificaciones del dispositivo.	18

Índice de figuras

1.1. Diagrama de bloques del algoritmo implementado por Josué Roberto Hernández Juárez. Diseño de un sonómetro.	4
1.2. Diagrama de bloques del algoritmo para programación del sonómetro.	5
2.1. Diagrama de bloques genérico de un sonómetro. (11).	8
2.2. Curvas de Fletcher & Munson o curvas isofónicas, expresadas en dB (re $10^{-12} W/m^2$).	9
2.3. Curvas de ponderación A, B, C (12).	9
2.4. Algunos modelos comerciales de Arduino (16).	13
2.5. Dos modelos comerciales Wemos con el chip ESP-12S (18).	13
3.1. Micrófonos comerciales para Arduino.	16
3.2. Diagrama de bloques del desarrollo a seguir para el diseño del dispositivo.	17
3.3. Diagrama de bloques para el diseño del circuito electrónico.	18
3.4. Respuesta en frecuencia del micrófono WM-61A de Panasonic (19). .	19
3.5. Configuración básica de un micrófono de condensador electret (19). .	19
3.6. Prototipo del pre amplificador en la tarjeta de desarrollo.	20
3.7. Circuito preamplificador realizado con Eeschema (Kicad) (21).	21
3.8. Filtro de ponderación A realizado con Eeschema (Kicad).	22
3.9. Prototipo del filtro de ponderación A.	22
3.10. Respuesta en frecuencia del filtro de ponderación A simulado con LTSpice (24).	22
3.11. Rectificador de precisión de media onda.	24
3.12. (Izq. a Dcha.) Circuito RC, rectificador de media onda realizado con Eeschema (Kicad).	25
3.13. (Rojo) Prototipo del circuito RC, (Azul) Prototipo del rectificador de media onda.	25
3.14. Entrada y salida del rectificador de media onda con filtro RC.	26

3.15. Amplificador logarítmico para entrada positiva realizado con Eschema (Kicad).	27
3.16. Amplificador de salida y divisor de voltaje para el calibrado del umbral de la señal medida. Realizado con Eschema (Kicad).	28
3.17. Circuito de alimentación realizado con Eschema (Kicad).	29
3.18. Pila alcalina 6LP3146 de 9V.	29
3.19. Entorno de Kicad.	31
3.20. Diseño del circuito electrónico completo del sonómetro con Eschema (Kicad).	31
3.21. Diseño final del circuito electrónico del sonómetro con Pcnw. (V1.1)	32
3.22. Vista 3D del circuito electrónico del sonómetro (V1.1).	32
3.23. PCB del circuito electrónico del sonómetro (V1).	32
3.24. Diseño 3D del soporte.	33
3.25. Ruido rosa con el eje de frecuencias logarítmico (36).	35
3.26. Fuente sonora omnipower modelo 4292-L. (37)	36
3.27. Calibración del dispositivo.	37
3.28. Aproximación polinómica del nivel equivalente ponderado A frente al voltaje.	37
4.1. Gráfica de prueba. Entorno de la página web de Adafruit.	39
4.2. Evolución temporal del L_{Aeq} con el sonómetro 2250 Light Brüel & Kjaer.	40
4.3. Evolución temporal del L_{Aeq} con el dispositivo.	40
4.4. Comparativa de los dos instrumentos de medida.	41
5.1. Amplificador logarítmico basado en transistor bipolar [27].	44

*Si he logrado ver más lejos,
ha sido porque he subido
a hombros de gigantes.
Isaac Newton*

Resumen

En el presente proyecto se ha diseñado e implementado un dispositivo capaz de medir el nivel de presión sonora y de manera remota. El instrumento de medida se caracteriza por permitir el registro y la visualización del parámetro acústico L_{Aeq} en tiempo real.

En el desarrollo del dispositivo de medida se ha tenido en cuenta en los puntos en los que ha sido posible la norma de la Comisión Electrotécnica Internacional IEC 61672-1:2014 (1).

Se ha diseñado una placa de circuito impreso para la adaptación analógica de la señal, basada en hardware libre. Para el envío de las muestras se ha implementado un zócalo en la PCB para el dispositivo Wemos D1 Mini, el cual se encarga de conectarse a la red y al servidor para la transmisión de los datos recogidos. El dispositivo Wemos está programado utilizando el IDE Arduino, basado en software libre.

Se concluye que el dispositivo brinda confiabilidad para realizar mediciones en un ambiente académico o de investigación con facilidad de comunicación a través de la red.

Capítulo 1

Introducción

En los últimos años la contaminación auditiva se ha visto incrementada debido al desarrollo industrial y al alto índice de vehículos presente en las ciudades, lo que ha propiciado un incremento significativo del nivel sonoro.

Para prevenir las posibles patologías que se podrían originar por el alto nivel sonoro, disponemos de instrumentos efectivos que permiten cuantificar el nivel al que una persona se expone durante un tiempo determinado.

El desarrollo de las redes de sensores es una de las tecnologías actuales que está en auge gracias a su abaratamiento. Una red de monitoreo en tiempo real para medir el nivel sonoro sería una herramienta útil para determinar la exposición de las personas a niveles sonoros presentes en un entorno.

La tecnología lleva unos años sufriendo una revolución. Existen plataformas de reconocido prestigio basadas en software libre que abren fronteras a técnicos cualificados, poniendo al alcance de cualquier individuo con conocimientos en electrónica y programación un medio eficiente y práctico para el montaje de prototipos y proyectos.

Los microcontroladores son los encargados de dominar cualquier dispositivo electrónico. La mayoría de los procesadores fabricados son utilizados para dispositivos digitales, variando sus capacidades de procesamiento, memoria y puertos GPIO (General Purpose Input/Output). Desde un simple procesador de 4 bits que controla un juguete o varios de ellos trabajando juntos en una red de sensores, hasta procesadores más especializados como pueden ser DSP (Digital signal processor) o procesadores gráficos.

En la actualidad, la contaminación acústica es un problema cada vez más grave, y se precisa un monitoreo continuo para tener un seguimiento representativo de su evolución. La Directiva Europea del ruido de 25 de junio de 2002 sobre evaluación y gestión del ruido ambiental (2) recomienda el uso de los mapas de ruido para determinar la exposición al ruido ambiental.

Para realizar dicho seguimiento, se necesita un estudio extenso que a menudo es costoso y que requiere una inversión de tiempo muy grande debido a la gran cantidad de datos recolectados.

Surge la idea de medir la presión sonora, abaratando costes y facilitando la obtención de medidas a través de un dispositivo inalámbrico, de tal forma que se pueda implementar como una estación de monitoreo fijo, cubriendo así, amplias zonas de estudio, monitoreadas simultáneamente durante periodos extensos de tiempo y recolectando los datos a través de un servidor.

1.1. Objetivos

En este proyecto se pretende diseñar un dispositivo para la medición del nivel de presión sonora de forma fiable, precisa y sencilla, haciendo uso de las posibilidades que ofrecen las tecnologías de software y hardware libre.

1.1.1. Objetivo principal

Desarrollar un sistema de bajo coste, basado en software y hardware libre, que permita la adquisición y digitalización del nivel de presión sonora lo más fielmente posible y enviarlo vía Wi-Fi a un servidor para posteriormente graficarlo en tiempo real.

1.1.2. Objetivos específicos

- Implementación de una red de sensores para graficar el nivel equivalente ponderado A en tiempo real.
- Implementar una base de datos con las muestras obtenidas por el dispositivo para realizar análisis en largos periodos de tiempo.

1.2. Limitaciones

- El dispositivo no tendrá ningún tipo de validación a nivel de normativa.
- Los datos que presente el dispositivo serán únicamente con ponderación A.
- El dispositivo no podrá realizar ningún tipo de análisis en frecuencia.
- Sólo medirá en intervalos de tiempo de 125 ms (Fast).

1.3. Antecedentes/Estado del arte

La revolución de la tecnología ha permitido desarrollar nuevos sistemas capaces de analizar y estudiar fenómenos que se presentan en todo el mundo. Algunos autores como Josué Roberto Hernández Juárez (3), han desarrollado tres etapas fundamentales en el diseño de un sonómetro. En este proyecto desarrollado en 2009 en México se diseñó mediante el entorno de programación LabView un medidor de señales de ruido. Basó su algoritmo para el procesamiento de la señal en el esquema de la figura 1.1.

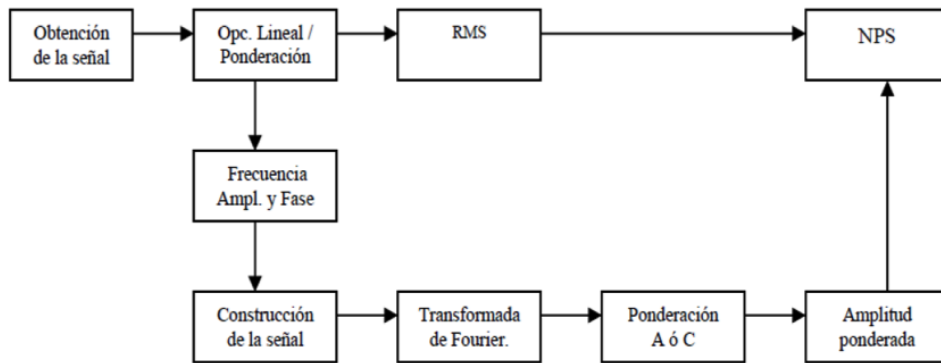


Figura 1.1: Diagrama de bloques del algoritmo implementado por Josué Roberto Hernández Juárez. Diseño de un sonómetro.

Fundamentalmente, el algoritmo es capaz de captar la señal, calcular el valor eficaz, descomponer la señal en frecuencia y ponderar la señal en dBA ó dBC.

Juan Carlos Merlo Bodenhorst (4), diseñó e implementó un prototipo de estación de monitoreo de ruido de bajo coste. El autor propone en su proyecto el uso de microcontroladores para la creación de un instrumento de metrología acústica.

Otro proyecto muy interesante ha sido el desarrollado por Diego Torres Domínguez (5), el cual desarrolló en 2013 un sonómetro profesional para iOS. Uno de los objetivos de este estudio era almacenar los datos en la nube o transmitir las medidas acústicas realizadas.

Luis Daniel García Ibarra (6), diseñó un sonómetro digital de bajo coste, basándose en el esquema de la figura 1.2 en México en 2011.

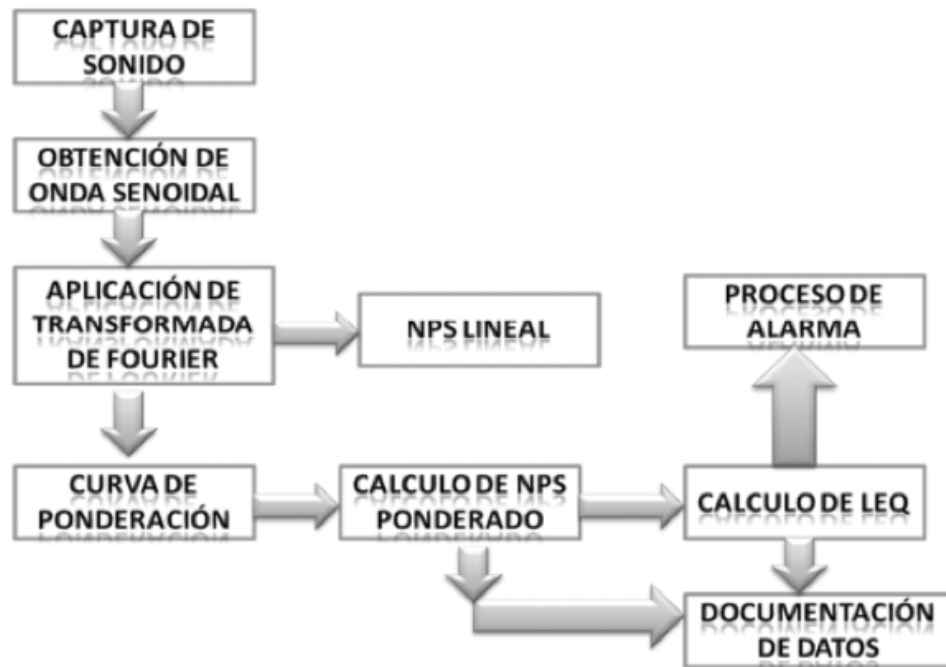


Figura 1.2: Diagrama de bloques del algoritmo para programación del sonómetro.

El algoritmo utilizado por Luis Daniel, a diferencia del explicado anteriormente, primero descompone la señal en frecuencia para después obtener el nivel de presión sonora o ponderar la señal para calcular el nivel equivalente.

Existen proyectos que tratan sobre el análisis de sonido en Smart Cities, con dispositivos de bajo coste, en concreto, “Sound Analysis in Smart Cities” (7) de los autores Juan Pablo Bello, Charlie Mydlarz y Justin Salamon que tratan sobre redes densas de sensores para el monitoreo de la contaminación acústica.

Siguiendo con los mismos autores, nos encontramos con otros proyectos muy interesantes; “The Implementation of Low-cost Urban Acoustic Monitoring Devices” (8), realizan un estudio de los entornos sonoros urbanos de la ciudad de New York, mediante dispositivos de detección acústica estáticos, inteligentes y de bajo coste, basados en hardware de consumo.

En el mismo contexto, el proyecto, “Designing and evaluating the performance of a wireless sensor network for environmental noise monitoring applications” (9) de los autores Kostas Sotirakopoulos, Richard Barham, Ben Piper, realizan una red de sensores acústicos utilizando la computadora Raspberry Pi 2B.

Capítulo 2

Conceptos generales

2.1. Sonómetros

El sonómetro es un instrumento de medida que permite obtener el nivel de presión sonora en un determinado lugar y un momento dado, conformado por un micrófono y un módulo de procesamiento.

Hay diferentes tipos de sonómetros según su grado de precisión. Se clasifican en clase 0, 1, 2 y 3 siendo la clase 0 los de mayor precisión (tolerancias más pequeñas), utilizados como patrones, y la clase 3 que son los de menor precisión.

Concretamente la norma que rige estas clases o tipos de sonómetros, para los casos usuales de tipo 1 y 2 son; IEC 61672-1:2014, ANSI S1.4 (10).

Las normas UNE-EN 61672-1:2014 y ANSI S1.4 tienen como objeto y campo de aplicación dar especificaciones del funcionamiento electroacústico para el diseño de un sonómetro.

2.1.1. Partes de un sonómetro

En la figura 2.1 se muestra un diagrama con los bloques básicos con los que tiene que contar un sonómetro.

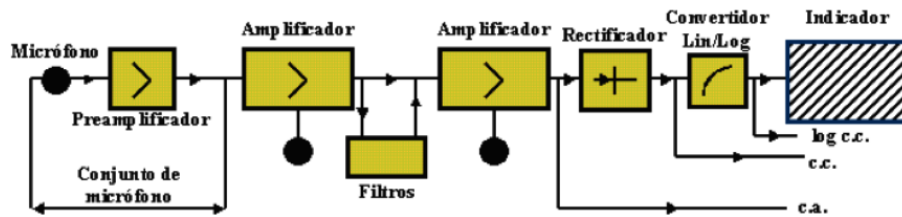


Figura 2.1: Diagrama de bloques genérico de un sonómetro. (11).

2.1.1.1. Micrófono

El micrófono constituye uno de los elementos más importantes de un medidor de nivel sonoro. Es el encargado de transformar la variable física, medida en Pascales, en una variable eléctrica, medida en Voltios. Debe tener un comportamiento que no perturbe la señal, en otras palabras, una respuesta en frecuencia lo más plana posible, así como también baja distorsión en altos niveles de presión sonora, buena sensibilidad y bajo ruido.

Por lo general para medir presión sonora, se recomienda utilizar capsulas de condensador de alta calidad y reducido tamaño. Este tipo de microfones hace variar la distancia entre las placas, que previamente han sido polarizadas, cuando le incide una onda sonora y permite obtener una excelente respuesta en frecuencia sin perturbar el campo acústico.

2.1.1.2. Amplificador

La señal entregada por el micrófono es de unos pocos mili voltios, por lo que se hace necesario un amplificador inmediatamente después del micrófono antes de ser procesada.

2.1.1.3. Redes de ponderación

La respuesta en frecuencia del oído humano no es plana. Para aproximar la respuesta en frecuencia del oído, se crearon las curvas de ponderación en frecuencia, conocidas como las curvas de Fletcher & Munson (12) mostradas en la figura 2.2.

Los sonómetros disponen de varias redes de ponderación (A,B,C,D) que se encargan de que la respuesta en frecuencia del sonómetro, sea similar a la del oído humano.

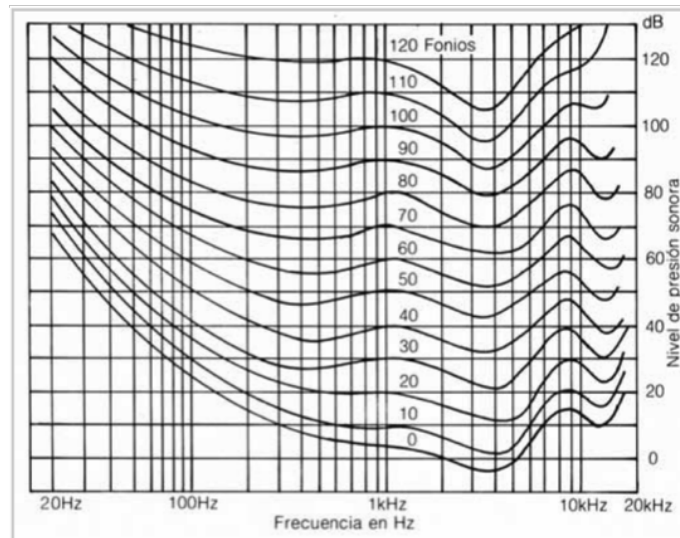


Figura 2.2: Curvas de Fletcher & Munson o curvas isofónicas, expresadas en dB (re 10^{-12} W/m^2).

Las curvas isofónicas relacionan la intensidad, medida en decibelios (dB), con la frecuencia, de dos sonidos para que éstos sean percibidos con igual intensidad por el oído, es decir, son curvas de igual sonoridad.

Una ponderación A asigna a cada frecuencia un valor ponderado que se relaciona con la sensibilidad del oído a esa frecuencia. La características de dBA fueron diseñadas basándose en las curvas isofónicas de igual sonoridad a 40 phons.

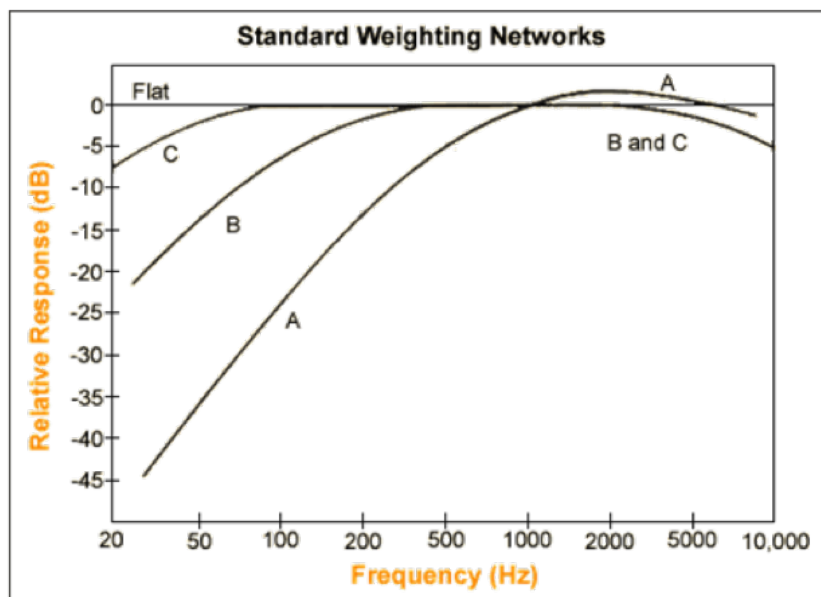


Figura 2.3: Curvas de ponderación A, B, C (12).

2.1.1.4. Ponderación temporal

Para evitar diferencias de lecturas sobre señales sonoras idénticas, los períodos de ponderación o constantes de tiempo están unificados por varias normas internacionales (UNE EN 61672-1:1204) y son los siguientes (13):

- Lento (Slow, S): valor promedio de un segundo. Se utiliza cuando la señal fluctúa ampliamente.
- Rápido (Fast, F): valor promedio sobre 125 milisegundos. Se utiliza para señales estacionarias.
- Por impulso (Impulsive, I): valor promedio de subida de 35 milisegundos y de caída 1 segundo. Mide la respuesta del oído humano ante sonidos de corta duración.
- Por pico (Peak, P): valor de pico. Valor promedio entre 50 y 100 microsegundos. Este valor sirve para evaluar el riesgo de daños en el oído, ante un impulso muy corto pero muy intenso.

2.1.2. Tipos de sonómetros

Principalmente hay dos tipos de instrumentos para medir niveles de presión sonora con diferentes características entre ellos.

2.1.2.1. Sonómetros generales

Muestran el nivel de presión sonora instantáneo en decibelios (dB). Estos instrumentos son útiles para realizar mediciones aproximadas del ambiente sonoro, sin ser medidas precisas y que no vayan a figurar en un informe técnico.

2.1.2.2. Sonómetros integradores-promediadores

Estos sonómetros son capaces de calcular el nivel continuo equivalente (L_{eq}), cálculo de percentiles (L_{10} , L_{90} , L_{95} , L_{max} , etc), análisis en frecuencia, diferentes tipos de ponderaciones y además incorporan funciones para el posterior análisis de los datos.

2.2. Parámetro utilizado para la evaluación y gestión del ruido ambiental según la Directiva 2002/49/CE del Parlamento Europeo y del Consejo de 25 de junio de 2002

La Directiva Europea 2002/49/CE de 25 de junio trata, entre otras cosas, en proporcionar una base para desarrollar y completar el conjunto de medidas comunitarias existentes sobre el ruido emitido por las principales fuentes, en concreto, vehículos, carreteras, aeronaves, etc.

El ámbito de aplicación de esta norma contempla la evaluación del ruido ambiental al que estén expuestos los seres humanos en zonas urbanizadas, no al ruido producido por la propia persona, o por actividades domésticas, o por los vecinos, o en el lugar de trabajo, etc.

Para evaluar un sonido, existen distintos parámetros que nos brindan información sobre los sucesos sonoros en determinados periodos de tiempo. Estos pueden ser valores instantáneos o valores máximos o mínimos, promedios energéticos, percentiles, etc. Los indicadores utilizados en la evaluación del ruido según la Directiva 2002/49/CE sobre evaluación y gestión del ruido ambiental son los siguientes:

2.2.1. Nivel día-tarde-noche L_{den}

El nivel día-tarde-noche L_{den} en decibelios (dB) se determina aplicando la fórmula [2.1]:

$$L_{den} = 10 \cdot \log \frac{1}{24} (12 \cdot 10^{(L_{day}/10)} + 4 \cdot 10^{(L_{evening}+5/10)} + 8 \cdot 10^{(L_{night}+10/10)}) \quad (2.1)$$

Donde:

- L_{day} es el nivel sonoro medio a largo plazo ponderado A definido en la norma ISO 1996-2:1987 (14), determinando a lo largo de todos los periodos diurnos de un año.
- $L_{evening}$ es el nivel sonoro medio a largo plazo ponderado A definido en la norma ISO 1996-2:1987, determinando a lo largo de todos los periodos vespertinos de un año.

- L_{night} es el nivel sonoro medio a largo plazo ponderado A definido en la norma ISO 1996-2:1987, determinando a lo largo de todos los periodos nocturnos de un año.

Consideraciones a tener en cuenta:

- Al día le corresponden 12 horas, a la tarde 4 horas y a la noche 8 horas.
- Un año corresponde al año considerado para la emisión de sonido y a un año medio por lo que se refiere a las circunstancias meteorológicas.
- El sonido que se tiene en cuenta es el sonido incidente.

2.3. Tarjeta de desarrollo

Estos sistemas basados en software libre van haciéndose hueco en todo tipo de proyectos de ingeniería. Permiten modificar su programa de control fácilmente, lo cual le da una gran versatilidad, además de su gran capacidad de procesamiento, pequeño tamaño y bajo coste. Son muy útiles para implementar redes de sensores, así como, dispositivos de comunicación, de registro o de control, convirtiéndolo en una herramienta capaz de analizar y estudiar fenómenos sonoros.

2.3.1. Arduino

Arduino (15) es una plataforma de prototipos electrónica de código abierto (open-source) basada en hardware y software flexibles y fáciles de usar. Está pensada para artistas, diseñadores, como hobby y para cualquier interesado en crear objetos o entornos interactivos. Es una herramienta con la que sentir y controlar el entorno. Las posibilidades que ofrece Arduino para crear proyectos son infinitas, interactuando con sensores, luces, motores y otros dispositivos.

Su modo de funcionamiento es simple, consta de una plataforma de código abierto basada en una placa con un microcontrolador y un IDE (Integrated Development Environment) desde la que desarrollar los programas para la placa. El microcontrolador de la placa se programa usando el Arduino Programming Language (basado en Wiring) y el Arduino Development Environment (basado en Processing). La posibilidad de funcionar en comunicación con algún otro software determinado o de manera totalmente autónoma, amplía enormemente su rango de aplicación. Atendiendo a las virtudes del sistema es fácil comprender que sean miles los usuarios de

la plataforma y multitud los proyectos llevados a cabo.

En la figura 2.4 se muestran algunos de los modelos de Arduino.

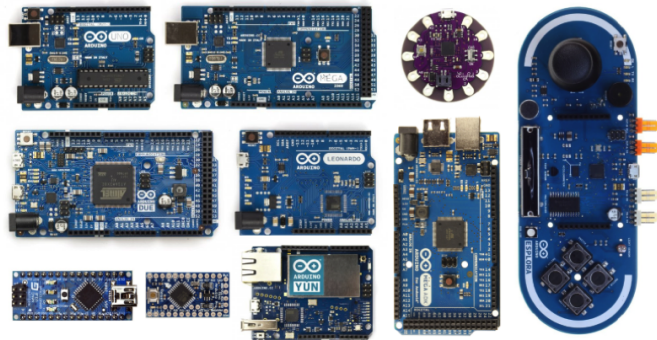


Figura 2.4: Algunos modelos comerciales de Arduino (16).

2.3.2. Wemos

Los dispositivos Wemos (17) se basan en los mismos principios que Arduino, una plataforma de prototipos electrónica de código abierto (open-source) basada en hardware y software flexibles y fáciles de usar, además de integrar en todos sus dispositivos el sensor Wi-Fi ESP8266.

A continuación en la figura 2.5 se muestran algunos de los dispositivos que están actualmente en el mercado bajo la plataforma WEMOS.

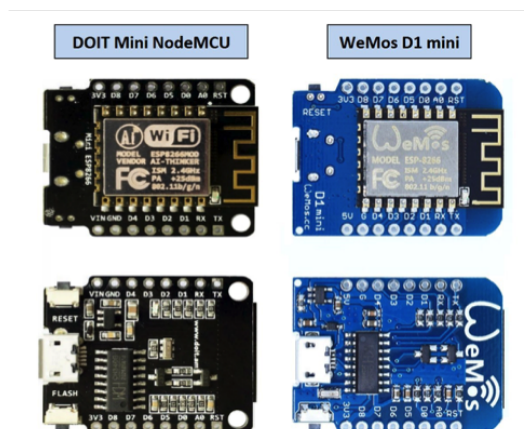


Figura 2.5: Dos modelos comerciales Wemos con el chip ESP-12S (18).

A modo informativo se muestra una tabla comparativa con los modelos que se han utilizado para la realización del proyecto.

Tabla 2.1: Tabla comparativa Wemos D1 Mini vs Arduino Due.

Especificaciones	Wemos D1 Mini	Arduino Due
Tipo de microcontrolador	ESP-8266EX	AT91SAM3X8E
Velocidad de reloj	80/160 MHz	84 MHz
Pines digitales E/S	12	54
Entradas analógicas	1	12
Salidas analógicas	0	2 (DAC)
Memoria de programa (Flash)	4M	512 Kb
Bits convertidor A/D	10	12

Capítulo 3

Metodología

En este capítulo se lleva a cabo la determinación de las herramientas de estudio que se van a emplear, que ayudaran de manera factible a la realización del presente trabajo. Se explicará con detenimiento el desarrollo del prototipo para la adquisición y visualización del nivel de presión sonora utilizando la tecnología que mejor se adapte a las necesidades del sistema. Encontraremos explicaciones sobre la adaptación de los componentes al sistema. Se describirán los elementos hardware que integra el prototipo y sus respectivas especificaciones técnicas. Por último, se detallan los diferentes softwares utilizados.

3.1. Desarrollo experimental

Inicialmente se realizaron varios montajes simples con Arduino Due y con distintos tipos de micrófonos (figura 3.1).

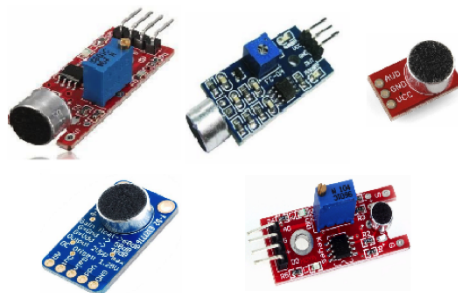


Figura 3.1: Micrófonos comerciales para Arduino.

Pronto se pensó en cambiar de estrategia. El rango dinámico, 40 dB, que nos ofrecen estos tipos de micrófonos es bajo, ya que, no son micrófonos para medir presión sonora. Algunos de los micrófonos cuentan con ganancia ajustable o vienen autoamplificados, es decir, poseen algún tipo de amplificación lineal y esa es la característica

por la que se descartó seguir por esta vía. La amplificación debe ser de tipo logarítmica. Esto último se explicará con detalle en el apartado 3.2.1.3 Diseño y desarrollo del amplificador logarítmico.

3.2. Desarrollo técnico

El diseño se basa en cuatro etapas fundamentales (figura 3.2).

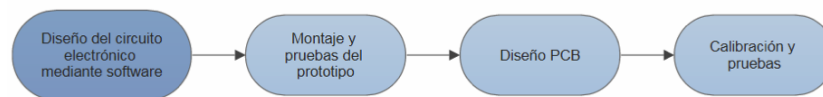


Figura 3.2: Diagrama de bloques del desarrollo a seguir para el diseño del dispositivo.

3.2.1. Diseño y desarrollo del sonómetro

Para el respectivo diseño y desarrollo del sonómetro se toma como base las siguientes especificaciones plasmadas en la norma UNE-EN 61672-1:2014, al momento de adquirir el nivel de presión sonora.

- Sonómetro con ponderación A.
- Rango de frecuencias 16 Hz- 16000 Hz.
- Velocidad de integración: F (Fast)
- El rango dinámico debe ser al menos de 60 dB a 1 KHz.

El desarrollo del sonómetro está compuesto por varias etapas:

- La primera es una etapa de hardware donde se desarrolla el proceso analógico de las señales de audio.
- La segunda es una etapa de software donde se digitalizan las señales de audio y se transmiten desde el dispositivo Wemos D1 Mini.

En la figura 3.3 se muestra el diagrama de bloques a seguir para el diseño del hardware.

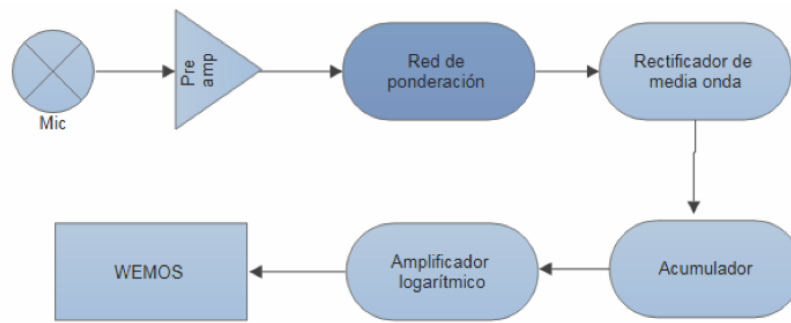


Figura 3.3: Diagrama de bloques para el diseño del circuito electrónico.

En la etapa de hardware se presta especial atención a los elementos que puedan modificar las señales analógicas generadas por el micrófono. Este hardware acondiciona las señales para ser digitalizadas posteriormente en la etapa de software.

Tabla 3.1: Especificaciones del dispositivo.

Características del sonómetro	Valores
Resolución del convertidor A/D	10 BITS
Sensibilidad del micrófono	17,78 mv/Pa
Protocolo de comunicación	IEE 802.11b
Rango de frecuencias	50Hz ~ 16 KHz
Memoria	4 Mb
Rango dinámico	45 dB ~ 105 dB
Parámetro de evaluación	L_{Aeq}
Ponderación	A
Filtro temporal	Fast

3.2.1.1. Diseño y desarrollo del pre amplificador

Para este proyecto se ha utilizado el micrófono Omnidireccional Back Electret Condenser Microphone Cartridge WM-61A que tiene una sensibilidad de $-35\text{dB} \pm 4\text{dB}$ ($0\text{ dB} = 1\text{ V/Pa}$, 1 KHz), relación señal a ruido mayor a 62 dB .

En la figura 3.4 se muestra la respuesta en frecuencia del micrófono WM61-A y en primera aproximación podemos concluir un comportamiento prácticamente constante dentro del espectro audible de frecuencias, entre los 50 Hz y los 15 KHz .

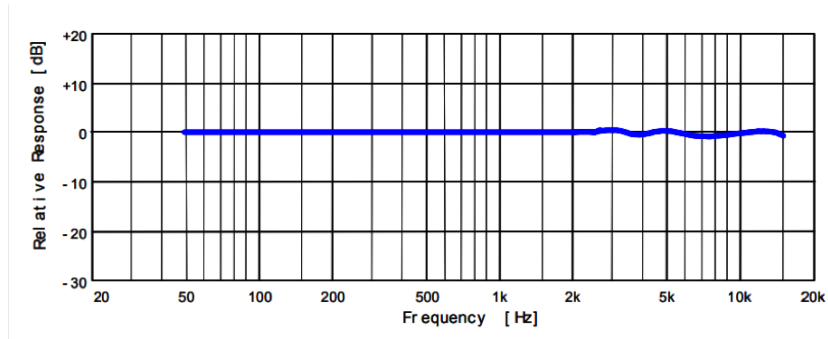


Figura 3.4: Respuesta en frecuencia del micrófono WM-61A de Panasonic (19).

Este micrófono lleva un amplificador con un transistor FET y se conecta como se muestra en la figura 3.5.

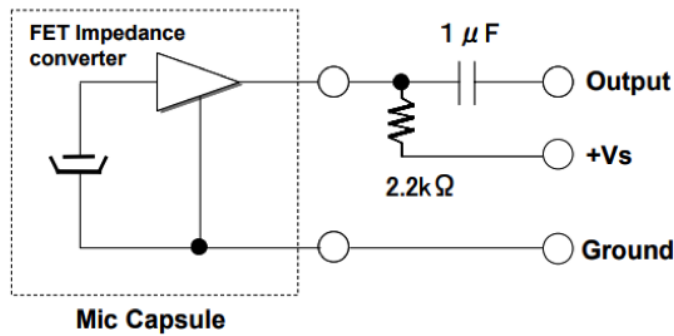


Figura 3.5: Configuración básica de un micrófono de condensador electret (19).

Para obtener la ganancia mínima necesaria del pre amplificador se procede a realizar algunos cálculos.

$$S_{dB} = 20 \cdot \log\left(\frac{S}{S_{ref}}\right) \quad (3.1)$$

A partir de la fórmula 3.1 obtenemos la sensibilidad siendo S_{ref} 1 V/ Pa.

$$-35 = 20 \cdot \log\left(\frac{S(V/Pa)}{1(V/Pa)}\right) \rightarrow S = 0,01778 \text{ (V/Pa)} \quad (3.2)$$

Ahora, procedemos a calcular los voltajes máximos y mínimos entregados por el micrófono, de acuerdo a la sensibilidad calculada en 3.2.

- Voltaje mínimo entregado por el micrófono:

$$2 \text{ mPa} \rightarrow 40dB \rightarrow V = S \cdot P \rightarrow V = 0,01778 \cdot 0,002 = 35,5 \text{ } \mu V \quad (3.3)$$

- Voltaje máximo entregado por el micrófono:

$$6,32 Pa \rightarrow 110dB \rightarrow V = S \cdot P \rightarrow V = 0,01778 \cdot 6,32 = 112,4 mV \quad (3.4)$$

A continuación, calculamos la ganancia mínima de nuestro pre amplificador.

$$G = \frac{V_s}{V_0} \quad (3.5)$$

De la expresión [3.5], sustituyendo procedemos al cálculo de la ganancia mínima necesaria.

$$G = \frac{V_s}{V_0} = \frac{5}{0,1124} = 44,5 \quad (3.6)$$

El voltaje máximo que queremos para la señal proporcionada por el micrófono es el voltaje máximo suministrado por la fuente de alimentación (5V). Se ha colocado un potenciómetro para ajustar la ganancia. Si se desea que 6,32 Pascales sea el máximo valor de presión que capte el sonómetro debe tener una ganancia mínima de 44.5.

Finalmente para el diseño del pre amplificador se optó por poner dos amplificadores lineales seguidos, con el integrado TL071CP (20) que cuenta con un amplificador operacional en cada integrado, uno con ganancia fija de $20 K\Omega$ y otro con un potenciómetro de $50 K\Omega$, para no sobrecargar el primer amplificador y evitar ruidos eléctricos indeseados.

La implementación del pre amplificador se llevó a cabo en una tarjeta perforada universal para el desarrollo de circuitos electrónicos, con el fin de realizar pruebas, como se muestra en la figura 3.6 y en la figura 3.7.

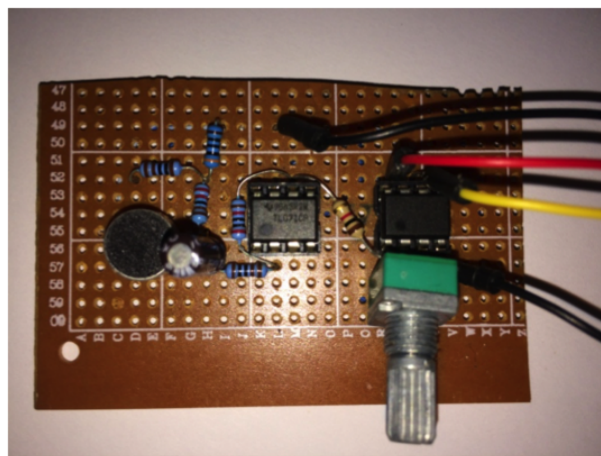


Figura 3.6: Prototipo del pre amplificador en la tarjeta de desarrollo.

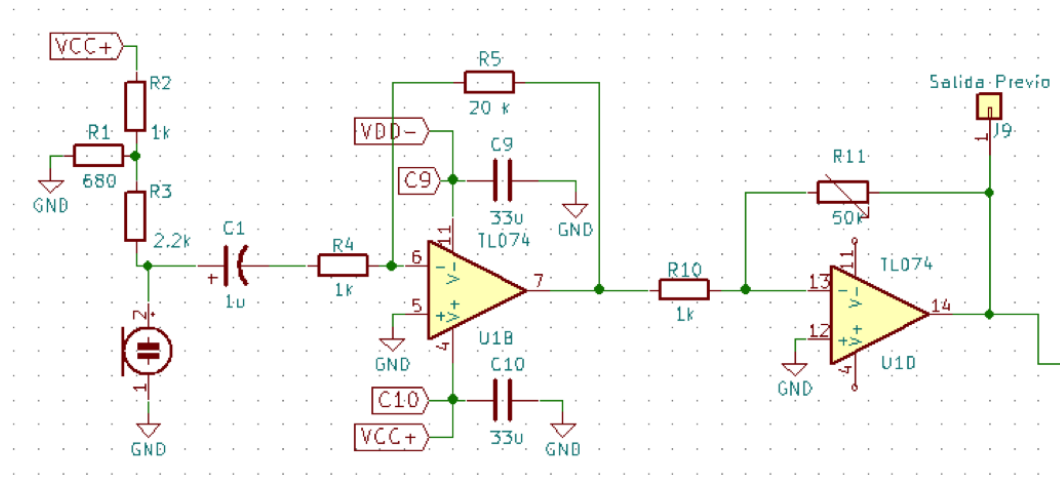


Figura 3.7: Circuito preamplificador realizado con Eeschema (Kicad) (21).

3.2.1.2. Diseño y desarrollo de la red de ponderación

El sonómetro se ha diseñado para medir únicamente la red de ponderación A.

Cabe mencionar, que el diseño del filtro está sacado del proyecto del Doctor Ingeniero Miguel Ángel Gantuz y del Ingeniero Ignacio Peacock (22) y está basado en la norma Argentina IRAM 4074 (23), en la que se especifica que la característica de compensación A es teóricamente realizada con dos polos en el plano complejo de frecuencias, situado en el eje real a 20,6 Hz para proveer la caída en baja frecuencia, y cuatro polos en el eje real en las frecuencias 107,7 Hz, 737,9 Hz y dos polos en 12000 Hz para producir la caída de alta frecuencia.

En la misma norma también aclara que las características de compensación en frecuencia A deben realizarse con circuitos pasivos de resistencias y condensadores.

A continuación se muestra la función de transferencia para el filtro de ponderación A (H_A) (22).

$$H_A(s) = \frac{4\pi^2 \cdot 12200^2 \cdot s^4}{(s + 2\pi \cdot 20,6)^2 \cdot (s + 2\pi \cdot 12200)^2 \cdot (s + 2\pi \cdot 107,7) \cdot (s + 2 \cdot \pi \cdot 738)} \quad (3.7)$$

En las figuras 3.8, 3.9 y 3.10 se muestra el filtro en el software de diseño, el prototipo en la tarjeta de desarrollo y la respuesta en frecuencia de dicho filtro simulado con LTSpice, respectivamente.

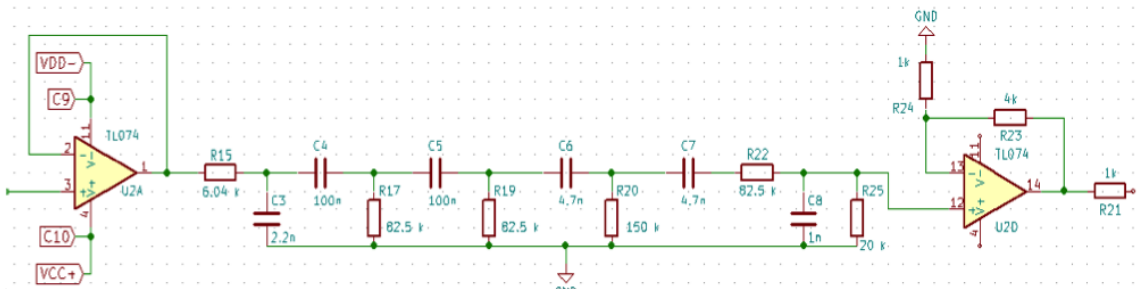


Figura 3.8: Filtro de ponderación A realizado con Eeschema (Kicad).

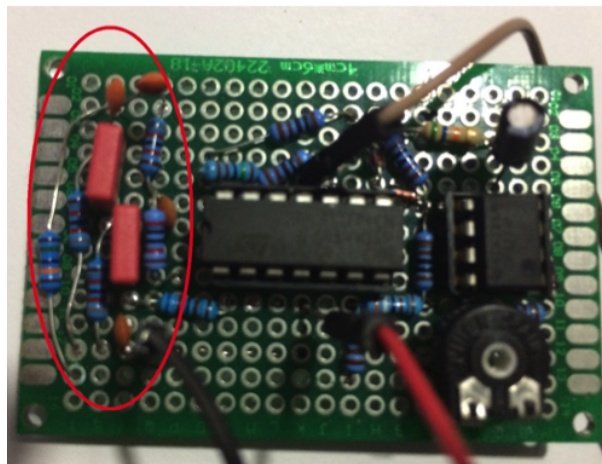


Figura 3.9: Prototipo del filtro de ponderación A.

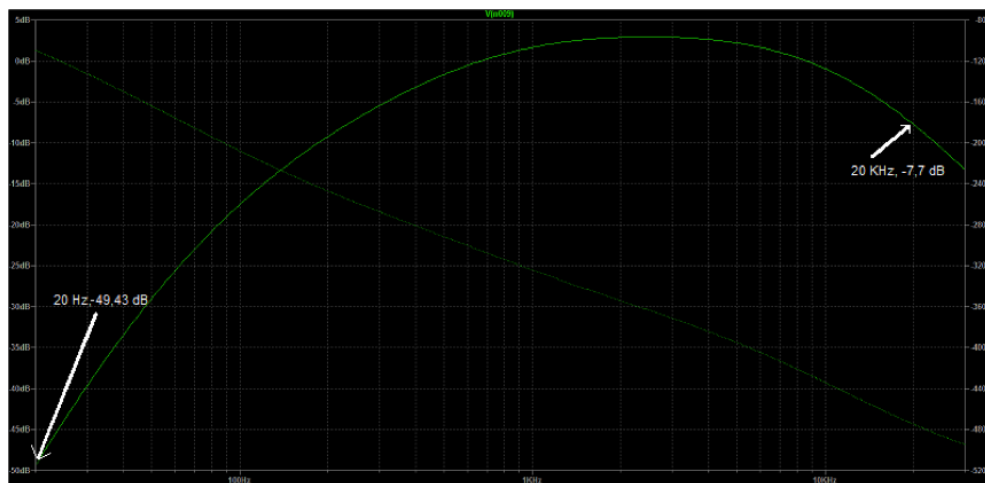


Figura 3.10: Respuesta en frecuencia del filtro de ponderación A simulado con LTSpice (24).

En la figura 3.8 se puede observar que antes del filtro de ponderación hay un amplificador operacional realimentado negativamente, este tiene la función de entregar

la señal con una impedancia muy baja, es decir, coge la señal de alta impedancia entregada por el micrófono y el preamplificador y entrega la señal al filtro de ponderación con una impedancia muy baja.

A la salida del filtro se ha colocado un amplificador para compensar las caídas de tensión del circuito RC.

Para el prototipo se utilizó el integrado TL074CN (25), que cuenta con cuatro amplificadores operacionales en cada integrado.

En la figura 3.10 se observa que la respuesta en frecuencia del filtro se asemeja a la curva estandarizada de ponderación A.

3.2.1.3. Diseño y desarrollo del rectificador de media onda y circuito RC integrador

La idea original era muestrear la señal de audio con Arduino a 44100 Hz cumpliendo el teorema de Nyquist, lo que suponía una carga computacional demasiado elevada cuando hablamos de un microcontrolador, es este caso Arduino Due.

Es por eso que se pensó en obtener el valor energético de cada onda sonora que recogiera el micrófono. Para ello, es necesario un rectificador de media onda seguido de un circuito RC que actúe como integrador. Además, como hablamos de tensiones muy pequeñas cuando queremos medir niveles de presión sonora muy bajos, necesitamos un rectificador de media onda de precisión. Lo que queremos obtener es una magnitud proporcional a la suma de las energías de los diferentes componentes de la onda sonora. En otras palabras, queremos obtener el valor eficaz de la señal.

El diodo no es una unión PN (26) ideal, por lo que presenta varios problemas a la hora de utilizarlo. Necesita una resistencia de carga para permitir el paso de corriente necesaria para polarizar el diodo. Por otro lado, el amplificador operacional pasa de zona lineal a saturación cuando cambia de signo la señal de entrada, por lo que, su respuesta es bastante lenta, dejando la señal tal cual, sin amplificarse ni atenuarse (27).

Existen diversas estructuras con varios diodos y resistencias para impedir que el amplificador operacional abandone la zona lineal. En la figura 3.11 se muestra una configuración típica de rectificador inversor de media onda de precisión con salida

positiva.

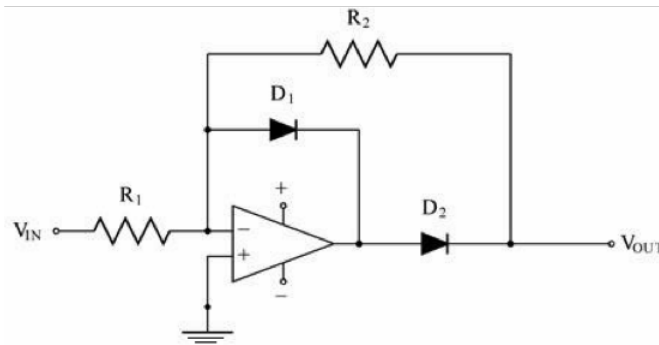


Figura 3.11: Rectificador de precisión de media onda.

El estudio de esta estructura es sencillo. En primer lugar, debe suponerse la entrada V_{IN} bien positiva, bien negativa. A continuación, deben estudiarse las dos posibles combinaciones de estado de D_1 y D_2 llegando a las conclusiones siguientes.

- Si la entrada V_{IN} es positiva, es fácil demostrar que el único estado coherente es aquel en el que el diodo D_1 conduce y D_2 no conduce, entonces el lazo de realimentación se cerraría con D_1 , y $V_{OUT} = 0$.
- Si la entrada V_{IN} es negativa, D_1 no conduce y D_2 conduce, con lo que se cerraría el bucle de realimentación con R_2 y con D_2 , entonces $V_{OUT} = -V_{IN} \cdot \left(\frac{R_2}{R_1}\right)$, lógicamente si hacemos que $R_2 = R_1$, se consigue una rectificación perfecta.

Con esta estructura conseguimos una rectificación precisa y dado que el amplificador operacional nunca abandona la zona lineal, la frecuencia máxima de trabajo aumentaría. Así la frecuencia de trabajo estaría limitada ahora por las capacidades de los diodos y por las propiedades del amplificador operacional en zona lineal (producto ganancia-ancho de banda- slew rate). Finalmente, con esta estructura se consigue rectificar la parte negativa de la señal (27).

Por otro lado, para calcular el valor eficaz de la señal, se barajó la opción de hacer un circuito integrador con un amplificador operacional, para integrar la señal rectificada, pero pronto se dio con otra solución igual de válida para obtener el valor eficaz de la señal. Esta solución es un simple circuito RC, el cual es el encargado de la integración temporal F (Fast). El producto $R \cdot C = 56700 \cdot 2,2 \cdot 10^{-6} = 0,12474$ seg nos da la constante de tiempo del circuito (τ), y equivale al tiempo que el condensador tardaría en cargarse, aproximadamente 125 milisegundos. En la figura 3.12 se observa el circuito rectificador de media onda y el circuito RC.

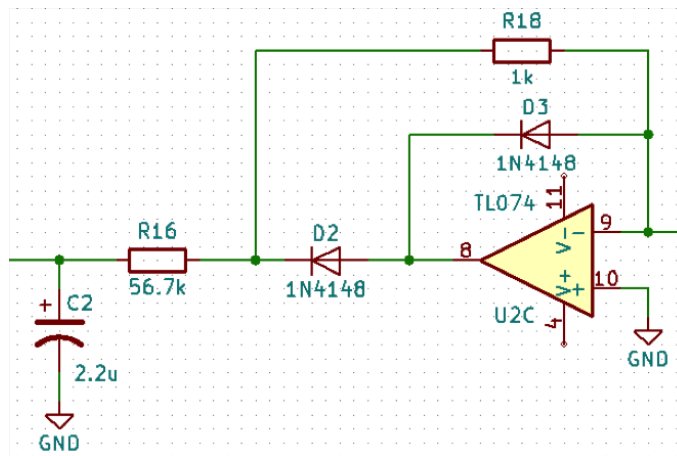


Figura 3.12: (Izq. a Dcha.) Circuito RC, rectificador de media onda realizado con Eeschema (Kicad).

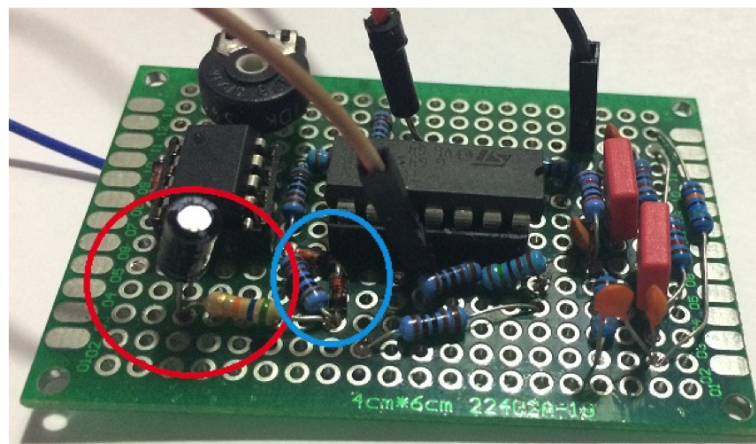


Figura 3.13: (Rojo) Prototipo del circuito RC, (Azul) Prototipo del rectificador de media onda.

El circuito RC detrás del rectificador de media onda, en la configuración que se encuentra, es un filtro paso bajo. Sin embargo, debe tenerse en cuenta que debido a la no linealidad del circuito que lo precede, el filtro no se limita a mantener el valor de continua (valor medio) de la onda rectificada y rechazar los armónicos.

Haciendo referencia a la figura 3.14, suponemos que inicialmente el condensador está descargado. A medida que V_s crece hacia valores positivos, el diodo se polarizará en forma directa y por lo tanto conducirá y la tensión de salida seguirá a la de la entrada. Este proceso continuará hasta el momento t_1 en que la tensión de entrada disminuya más rápidamente que la descarga del condensador a través de la resistencia de carga. A partir de ese momento la tensión de salida se desvincula de la de la

entrada, siguiendo la evolución temporal de la descarga del condensador a través de la resistencia de carga. Mientras tanto, la entrada continuará con su variación senoidal. En un instante t_2 la caída exponencial de la salida se cruzará con el ascenso senoidal de la entrada, y a partir de entonces el diodo volverá a conducir, repitiéndose el proceso anterior.

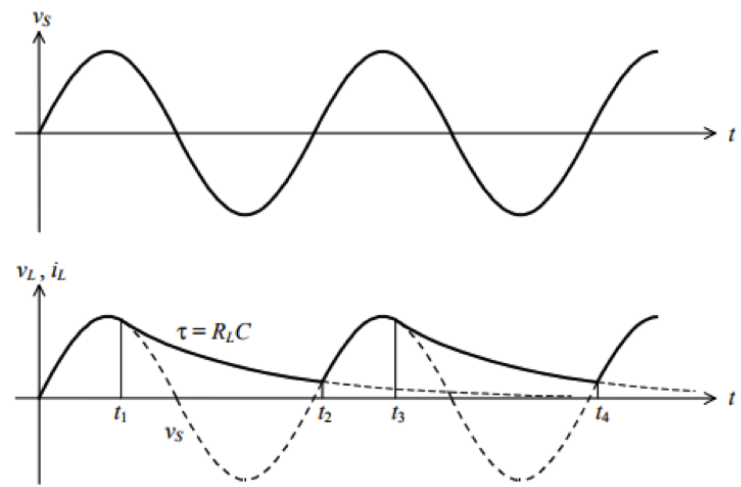


Figura 3.14: Entrada y salida del rectificador de media onda con filtro RC.

A alta frecuencia, es decir, cuando $\omega \gg \frac{1}{RC}$, el condensador no tiene tiempo suficiente para cargarse y la tensión en los bornes permanece pequeña.

Así: $V_R \sim V_{IN}$ y la intensidad del circuito vale por tanto $I \sim V_{IN}/R$. Como $V_c = \frac{1}{C} \int_0^t I dt$, se obtiene:

$$V_c = \frac{1}{RC} \int_0^t V_{IN} dt \quad (3.8)$$

Al final de esta etapa, se intercalará un amplificador para compensar las pérdidas y adaptar la señal para el siguiente proceso.

3.2.1.4. Diseño y desarrollo del amplificador logarítmico

Llegados a este punto, ya es posible obtener una magnitud proporcional al nivel sonoro.

Pero continua el mismo problema que en el desarrollo experimental, el rango dinámico de la señal de entrada al conversor A/D es muy pequeño. Si nos fijamos en una onda de presión sonora de 40 dB que corresponde con una presión de 0.002 Pa y una de onda de 110 dB que corresponde con una presión de 6.32 Pa, y sabemos que la sensibilidad de nuestro micrófono es de $-35\text{dB} \pm 4 \text{ dB}$, podemos calcular los valores

en voltios de la señales entregadas por el micrófono. Sabemos que 40 dB, genera una señal de $35.5 \mu V$, y 110 dB, genera una señal de 112.4 mV. Si el convertor dispone de 10 bits de rango de entrada unipolar de 0 V a 3.3 V, se podrían representar 1024 niveles de tensión, con una resolución de 0.00322 V por nivel. Es decir, para niveles bajos de presión sonora, el dispositivo tendría muy mala resolución.

Por lo tanto, finalmente se optó por un amplificador logarítmico para resolver el problema anterior. La función del amplificador logarítmico es comprimir la señal proporcionalmente al logaritmo de la señal de entrada, permitiéndonos tener un mayor rango dinámico.

En la figura 3.15 se muestra el amplificador logarítmico diseñado para este trabajo. Esta configuración es una de las más sencillas que existen pudiéndose observar que para estabilizar el circuito, la realimentación se realiza a través del terminal inversor, ya que, en el fondo, un diodo no es sino una resistencia fuertemente no lineal. De esta forma, conseguimos que la salida sea proporcional al logaritmo de la entrada.

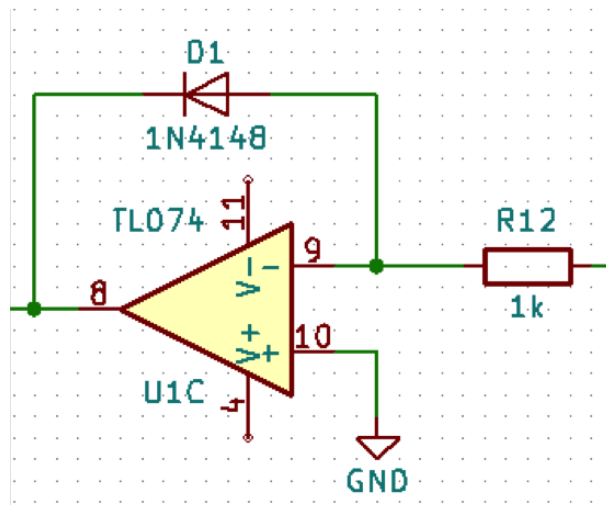


Figura 3.15: Amplificador logarítmico para entrada positiva realizado con Eschema (Kicad).

Si suponemos un punto en la entrada negativa del operacional a la que llamaremos V_{-9} , podemos deducir, sabiendo que la impedancia de entrada al amplificador es infinita, que toda la corriente que atraviesa R_{12} se deriva hacia el diodo. Entonces:

$$I_D = \frac{V_{IN} - V_{-9}}{R_{12}} = I_s \cdot \left[e^{\frac{V_9 - V_{OUT}}{N \cdot V_T}} - 1 \right] \quad (3.9)$$

Siendo I_s y N parámetros característicos del diodo, V_T el voltaje térmico, es decir, $k\frac{T}{q}$, siendo k la constante de Boltzmann de valor $1.38 \cdot 10^{-23}$ (J/K), y q la carga del electrón, de valor $1.609 \cdot 10^{-19}$ C y T la temperatura en grados Kelvin.

En el punto V_{-g} se forma una tierra virtual, es decir, $V_{-g} = 0$, entonces el diodo se polarizará en directa y la ecuación anterior pasará a ser.

$$\frac{V_{IN}}{R_{12}} = I_s \cdot e^{\frac{V_{OUT}}{N \cdot V_T}} \rightarrow V_{OUT} = -N \cdot V_T \cdot \ln\left(\frac{V_{IN}}{R_{12} \cdot I_s}\right) \quad (3.10)$$

Como se puede observar en la fórmula anterior, la salida del amplificador logarítmico depende fuertemente de la temperatura, además de ser negativa. En este punto se ha colocado un amplificador lineal, convirtiendo la señal en positiva, además se intercaló un potenciómetro de 10 K Ω para ajustar la señal de salida.

En la patilla positiva del último amplificador operacional se ha diseñado un divisor de voltaje para calibrar el umbral de la señal medida, tal y como puede apreciarse en la figura 3.16.

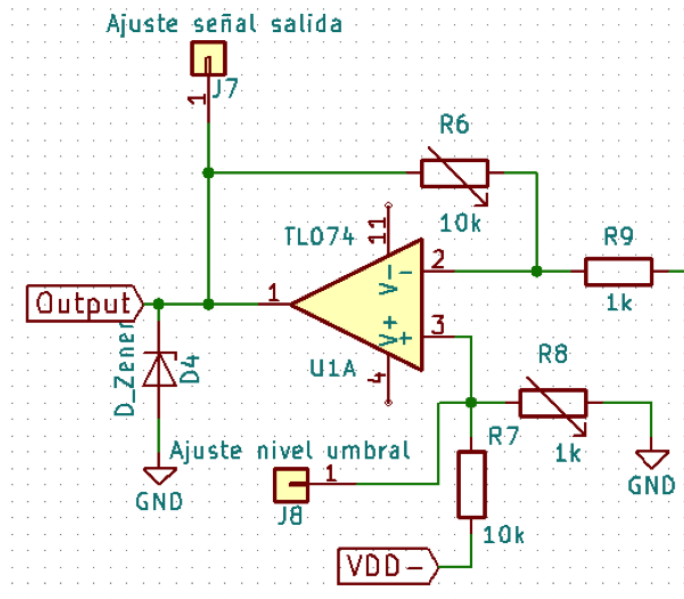


Figura 3.16: Amplificador de salida y divisor de voltaje para el calibrado del umbral de la señal medida. Realizado con Eschema (Kicad).

En la figura 3.16 se puede apreciar un diodo zener al final de la salida del proceso analógico. Este diodo tiene la función de limitar la salida a 3.3 V, protegiendo así el dispositivo Wemos D1 Mini.

3.2.1.5. Alimentación del circuito

Para la alimentación del dispositivo se diseñó un pequeño circuito electrónico con el fin de obtener el voltaje simétrico necesario para la alimentación de los circuitos integrados.

El dispositivo estará alimentado con una pila de 9V o un transformador, de los cuales sacaremos 9V y -9V, con el integrado 7660S CPAZ (28). Estos voltajes, pasarán por unos reguladores, concretamente el LM7905C (29) para la salida negativa y el LM7805C (30) para la positiva, para finalmente alimentar todo el circuito con 5V y -5V, consiguiendo así, un mayor rendimiento del dispositivo.

En la siguiente figura se muestra el circuito de alimentación.

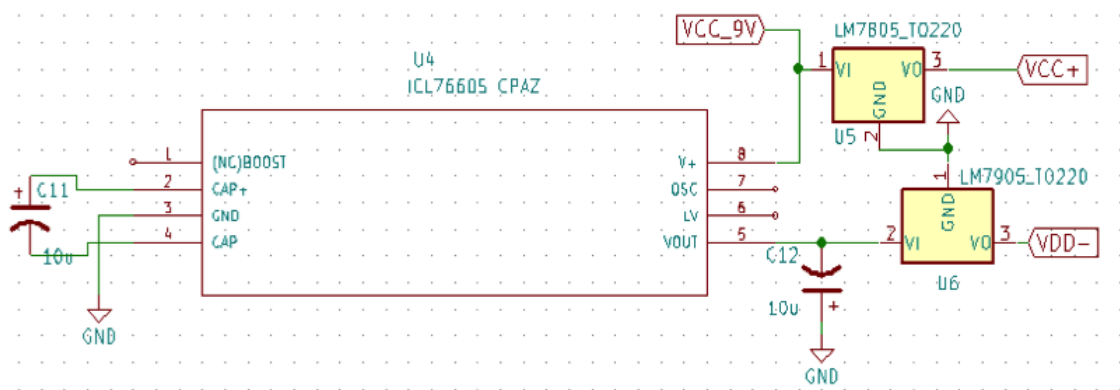


Figura 3.17: Circuito de alimentación realizado con Eschema (Kicad).



Figura 3.18: Pila alcalina 6LP3146 de 9V.

Se podría hacer una estimación de la duración de la pila, de una forma muy sencilla. Conocemos el consumo del dispositivo, es decir, el consumo total del circuito eléctrico, del Wemos y de los leds y es de aproximadamente 110 mA.

Si sabemos que una pila alcalina como la de la figura 3.18 tiene una capacidad mínima de 550 mAh, solo hay que dividir la capacidad de la pila entre el consumo de nuestro dispositivo, así obtenemos el tiempo en horas que la pila nos está suministrando una corriente suficiente para que el dispositivo pueda funcionar correctamente.

$$\frac{\textit{Capacidad pila}}{\textit{Corriente consumida}} = \frac{550 \textit{ mAh}}{110 \textit{ mA}} = 5 \textit{ horas} \quad (3.11)$$

Esas cinco horas son aproximadas. Se ha podido comprobar experimentalmente que el dispositivo deja de funcionar correctamente cuando la pila suministra un voltaje inferior a 7 V.

3.2.2. Diseño de PCB con Kicad

Para el diseño de la PCB se ha utilizado Kicad. Es un entorno de software usado para el diseño de circuitos eléctricos, muy flexible y adaptable, en el que se pueden crear y editar un gran número de componentes.

Nos permite el diseño de circuitos impresos de forma sencilla e intuitiva. Además, en Pcbnew, los circuitos se pueden visualizar con múltiples capas y ser visualizados en 3D.

Kicad está organizado en cinco partes:

- Kicad: administrador de proyectos.
- Eeschema: editor de esquemáticos.
- Cvp pcb: seleccionador de huellas de los componentes usados en el esquemático.
- Pcbnew: entorno de diseño de los circuitos impresos (PCB).
- Gerbview: visualizador de archivos Gerber.

cido posible. Se ha incluido el zócalo para el dispositivo Wemos D1 Mini.

En la figura 3.22 y 3.23 se observa el diseño final para la presentación de este proyecto.

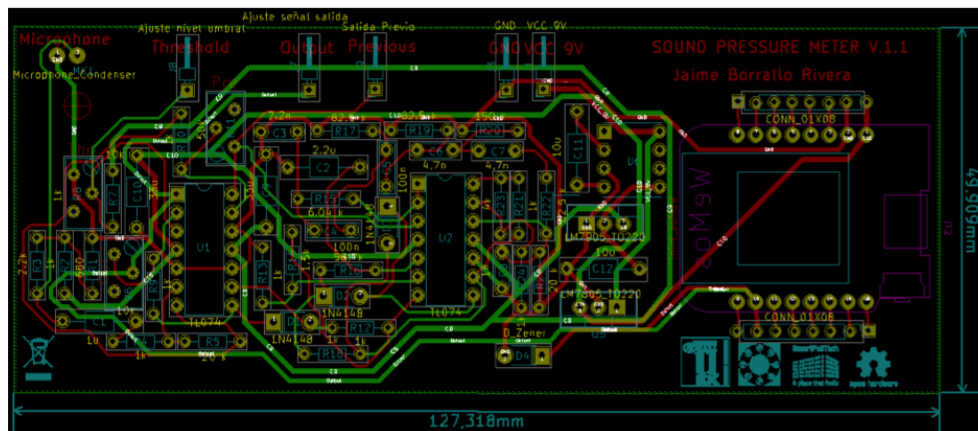


Figura 3.21: Diseño final del circuito electrónico del sonómetro con Pcnw. (V1.1)

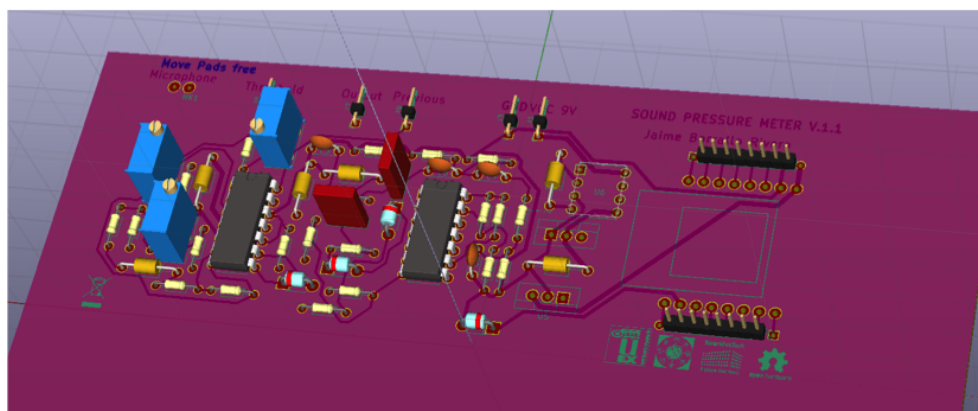


Figura 3.22: Vista 3D del circuito electrónico del sonómetro (V1.1).

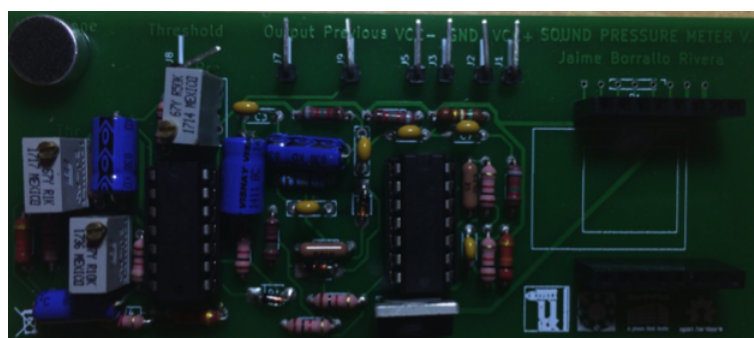


Figura 3.23: PCB del circuito electrónico del sonómetro (V1).

Como se puede apreciar, se han colocado varios potenciómetros y se han sacado líneas externas, para la calibración del circuito. También se han incluido los logotipos de la Universidad de Extremadura, de Smart Open Lab (SOL), de SmartPolitech y el símbolo de hardware abierto, además del título del proyecto y el nombre del autor.

La PCB se encargó en Seeed Studio (31). Esta empresa además de imprimir placas de circuito impreso, trabaja para combinar radio amateur con hardware de código abierto, contribuyen al desarrollo de IoT.

La versión de la PCB expuesta en este proyecto es la V1.1, y la versión montada es la V1. Esto se ha debido a algunos errores cometidos en el proceso del diseño y al añadido del circuito de alimentación que fue posterior al diseño de la PCB.

3.2.3. Diseño 3D del soporte para la PCB y la alimentación

Se ha creado e imprimido un diseño 3D, donde irán alojadas tanto la PCB para el procesado analógico de la señal, como el circuito de alimentación y la pila, además de un interruptor y dos LEDs. El de color rojo nos indica que el circuito está alimentado y el otro de color azul tiene la misión de hacernos saber cuándo se ha conectado al servidor y cuando ha enviado una muestra. Si al encender el dispositivo parpadea una vez lentamente, quiere decir que se ha conectado con éxito al servidor. Una vez que se ha conectado, parpadeará una vez más rápido que la anterior para indicarnos que ha enviado la muestra con éxito.

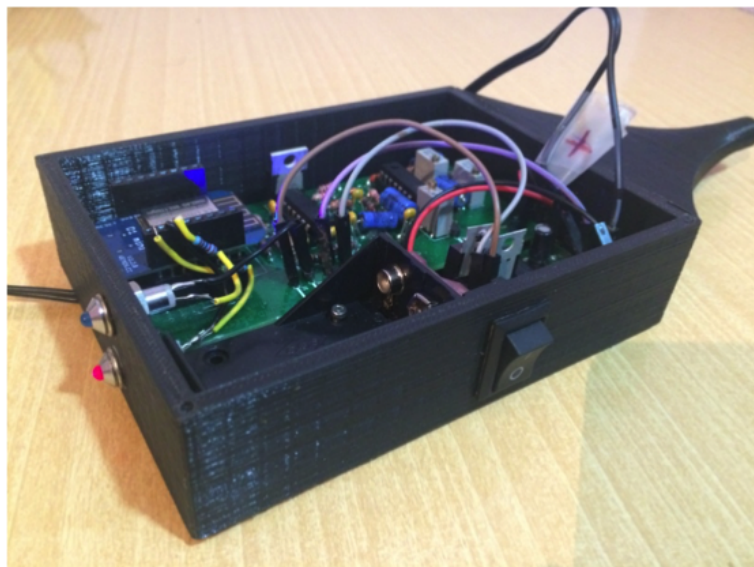


Figura 3.24: Diseño 3D del soporte.

El software de diseño 3D utilizado es Autodesk Inventor Professional (32), este software nos ofrece herramientas profesionales para el diseño mecánico, como el modelado paramétrico, simulación de sólidos en 3D, etc.

3.2.4. Consideraciones a tener en cuenta para la programación del dispositivo Wemos

Para la programación del dispositivo Wemos se han tenido en cuenta las siguientes consideraciones:

- Tiempo de medida

Dado que el producto RC nos da la constante de tiempo del circuito, es decir, el tiempo de carga del condensador, tomaremos una medida teniendo en cuenta ese producto RC.

- Envío de las muestras

Para el envío de las muestras necesitamos conectar el dispositivo Wemos a la Wi-Fi de SmartPolitech. Para ello hacemos que el dispositivo Wemos sea cliente de un servidor MQTT.

Los servidores MQTT básicos permiten a las aplicaciones y dispositivos móviles, soportados por las bibliotecas de clientes de MQTT, intercambiar mensajes.

MQTT (33) es un protocolo de transporte (Protocolo abierto de comunicación a través de mensajes) de mensajes Cliente/Servidor basado en publicaciones y suscripciones a los denominados “tópicos”. Cada vez que un mensaje es publicado será recibido por el resto de dispositivos adheridos a un tópico del protocolo. Está diseñado específicamente para redes y dispositivos de recursos limitados y con bajas tasas de transmisión.

- Ahorro de energía

Para maximizar la energía se barajó la idea de utilizar la función deepSleep (34) que soporta Wemos, pero se pudo comprobar que no era muy eficiente, ya que poner en suspensión el dispositivo por cortos intervalos de tiempo supone un consumo mayor que no suspenderlo. El pico de consumo máximo lo hace al iniciar la Wi-Fi del Wemos.

- Modificación del software de control

Si se quiere modificar el software, con el objeto de cambiar el tiempo de medida

por ejemplo, es decir, aumentar o disminuir el tiempo que tarda el dispositivo en enviar las muestras al servidor, se puede hacer por vía inalámbrica.

El ESP8266 cuenta con una biblioteca que acepta la programación OTA (Over The Air) (35), es decir, acepta el volcado de programas desde el IDE Arduino, de forma inalámbrica vía Wi-Fi.

3.2.5. Calibración del sonómetro

Para calibrar nuestro dispositivo es necesario utilizar un sonómetro profesional como referencia. Además, necesitamos una fuente acústica omnidireccional que se aproxime aun radiador isotrópico esférico. A través de una etapa de potencia, emitimos ruido rosa. El ruido rosa es un ruido patrón que se caracteriza por una disminución de 3 dB en la presión sonora cada vez que aumenta la banda de octava. Si hacemos pasar este ruido por un filtro de octavas, conseguimos que todas las bandas de octavas tengan el mismo nivel sonoro, esto es cierto dado que el ancho de banda de las bandas superiores es mayor que las inferiores. Los filtros de octava son proporcionales, es decir, cada vez que subimos una octava, duplicamos el ancho de banda.

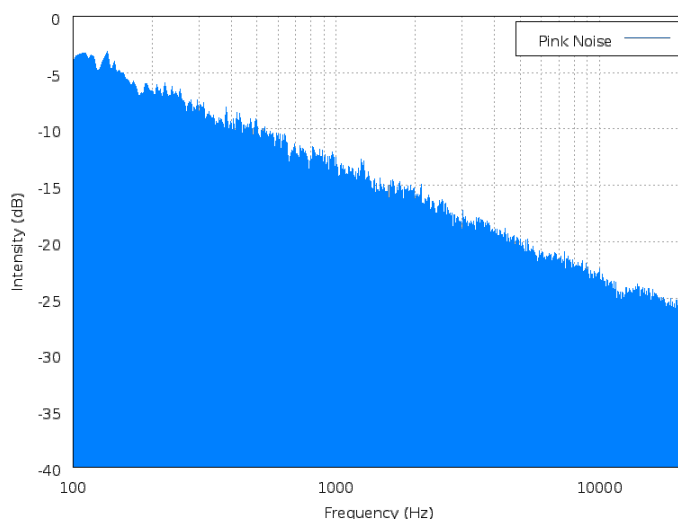


Figura 3.25: Ruido rosa con el eje de frecuencias logarítmico (36).

El sonómetro facilitado por la Universidad de Extremadura, es el 2250 Light de Brüel & Kjaer, el cual se ha calibrado previamente en cada medición. Por las características propias del micrófono que tiene instalado, está diseñado para medir presión sonora en campo libre.

La fuente sonora dodecaédrica es el modelo 4292-L de Brüel & Kjaer (37), la cual cuenta con doce altavoces para emitir sonido de forma uniforme.



Figura 3.26: Fuente sonora omnipower modelo 4292-L. (37)

La disposición de los distintos elementos que intervienen en la calibración se hizo de acuerdo a la norma ISO 1996-1 (38) e ISO 1996-2 (14), en la que se indica que el sonómetro tiene que estar siempre a la misma distancia de separación a la fuente, en nuestro caso 1m desde el eje central del altavoz. La fuente sonora y el sonómetro deben estar a 1.5 m del suelo y a una distancia superior a 1,5 m de cualquier superficie reflectora.

En el momento de la calibración la temperatura era de 11.5°C, la humedad relativa del 67%, y la velocidad del viento era de 10 Km/h.

Se fue variando el nivel de presión sonora a razón de 1 dB aproximadamente. Se fueron tomando los datos del nivel de presión ponderado A, y ponderación temporal Fast, y el nivel de voltaje recogido por nuestro dispositivo, para así obtener una relación entre la presión y el voltaje.

Los potenciómetros se fijaron en los siguientes valores; el potenciómetro para ajustar el umbral (Threshold) se fijó en 800 Ω , para obtener un voltaje negativo de aproximadamente -0.34 V y así ajustar el umbral de salida a 0 V. El potenciómetro para ajustar la salida (Output) se fijó en 7.5 K Ω , de esta forma limitamos el voltaje máximo entregado por el circuito que corresponde con una onda de presión sonora de 110 dB aproximadamente. El potenciómetro del previo se fijó en 3.8 K Ω para obtener la mayor ganancia posible sin saturar la señal.

En la figura 3.27 se puede observar la distribución de los distintos elementos que han intervenido en el proceso de calibrado.



Figura 3.27: Calibración del dispositivo.

El proceso de calibración se llevó a cabo varias veces con el objeto de obtener un ajuste más preciso, obviando algunas de ellas por que las condiciones climatológicas o el entorno sonoro no eran los apropiados.

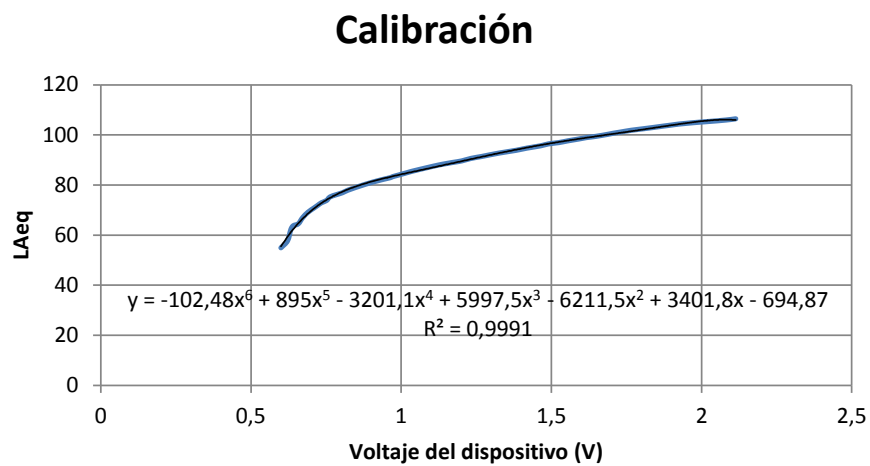


Figura 3.28: Aproximación polinómica del nivel equivalente ponderado A frente al voltaje.

Capítulo 4

Análisis y resultados

Para el análisis de los resultados se ha utilizado a modo de servidor donde recibir los datos, la pagina web de Adafruit (39). Esta empresa desarrolla hardware de código abierto y en su página web ofrece una forma sencilla de transmitir, registrar e interactuar con los datos, así como hacer gráficas, descargar los datos en una hoja de cálculo, indicadores, registros, etc. Lo que permite analizar la evolución temporal de los datos registrados (figura 4.1).

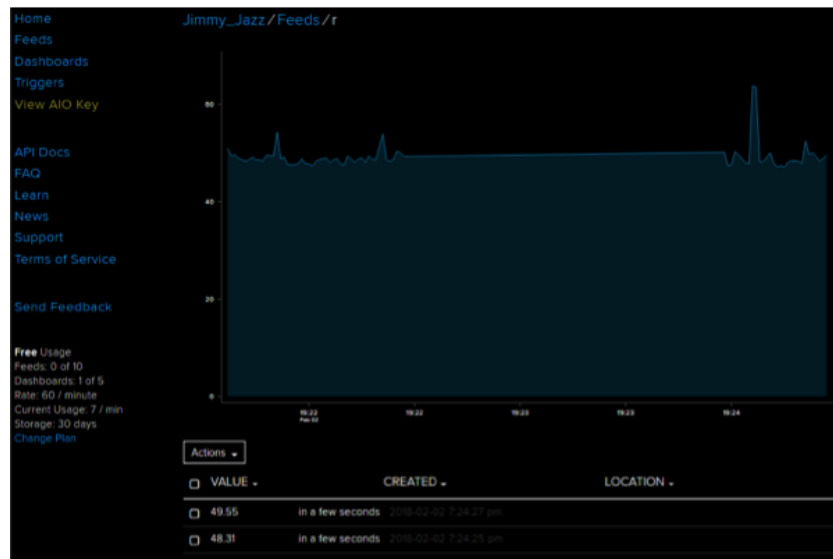


Figura 4.1: Gráfica de prueba. Entorno de la página web de Adafruit.

Para analizar el error de nuestro sonómetro se han realizado varias mediciones con el mismo sonómetro de referencia utilizado en el proceso de calibración, el 2250 Light de Brüel & Kjaer y excitando el ambiente sonoro con ruido rosa. Hemos ido tomando nota del L_{Aeq} en intervalos de 10 segundos durante 10 minutos, tanto en el sonómetro de referencia como en nuestro dispositivo y hemos ido incrementando

el nivel sonoro aleatoriamente. Obteniendo así las gráficas mostradas en las figuras 4.2, 4.3 y 4.4.

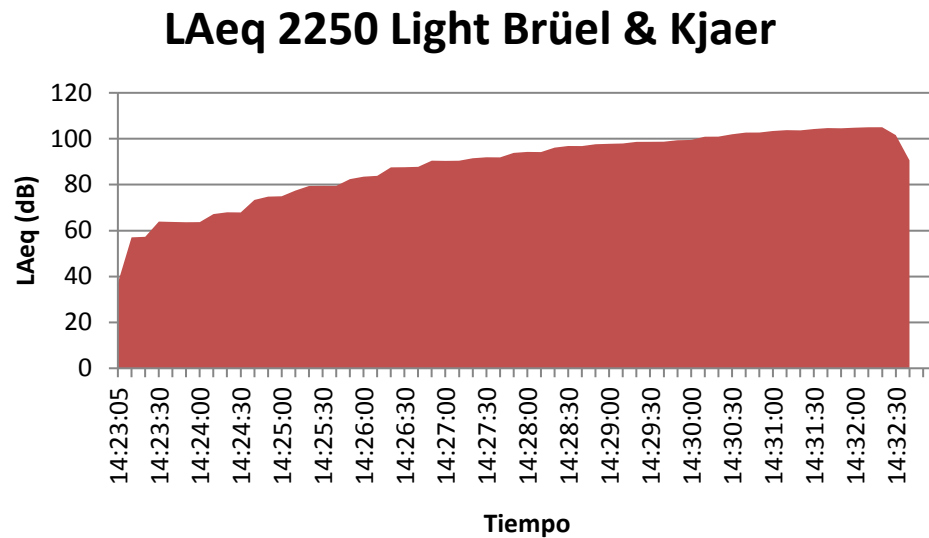


Figura 4.2: Evolución temporal del L_{Aeq} con el sonómetro 2250 Light Brüel & Kjaer.

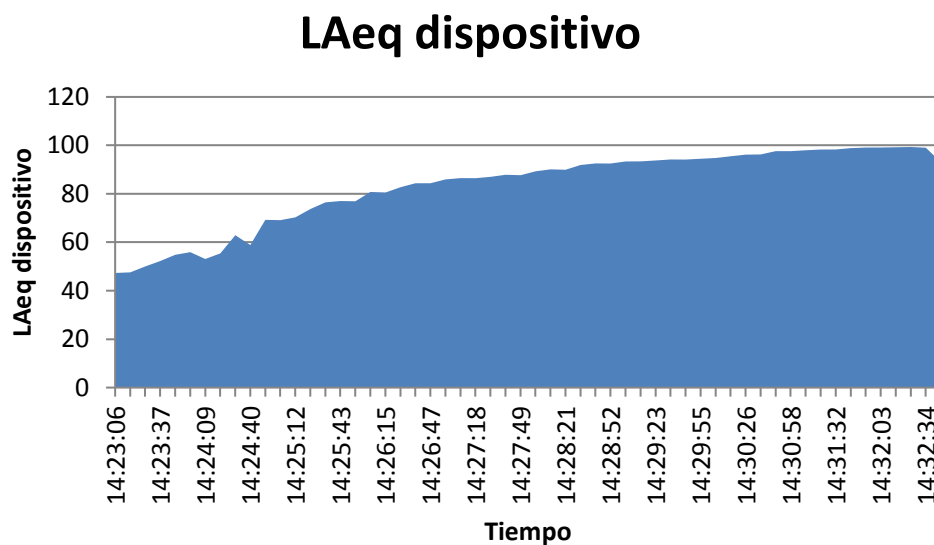


Figura 4.3: Evolución temporal del L_{Aeq} con el dispositivo.

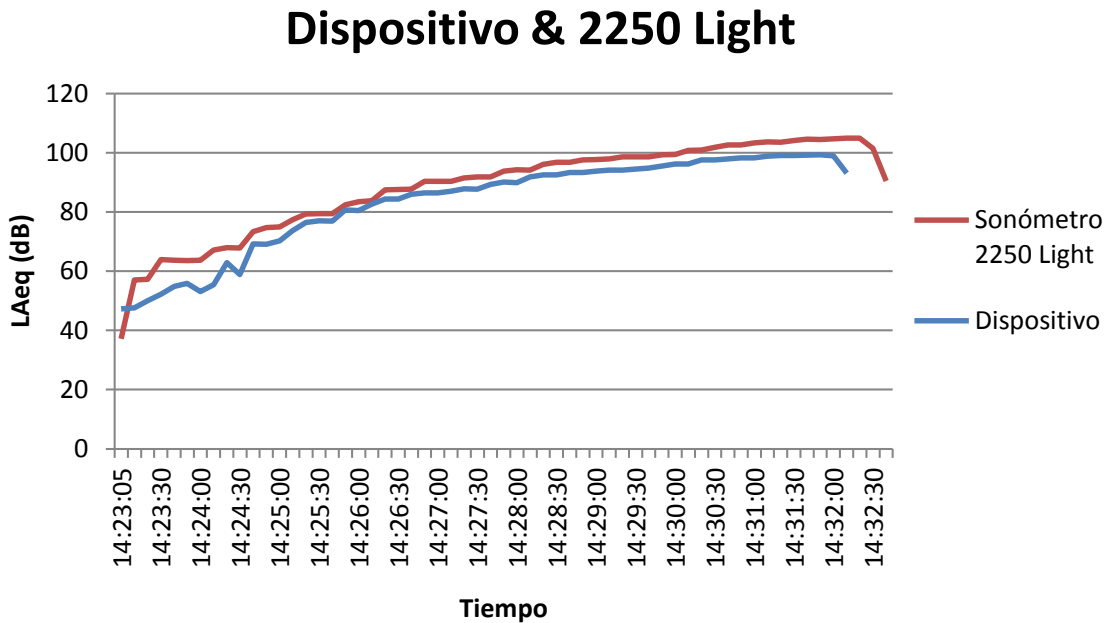


Figura 4.4: Comparativa de los dos instrumentos de medida.

Como se puede apreciar en las gráficas, el dispositivo es capaz de seguir el patrón de la evolución temporal que sigue el sonómetro de referencia pero con menos resolución, sobre todo a niveles bajos. En la figura 4.4, se observa como la línea (azul) que sigue el dispositivo en el intervalo entre 40 dB y 60 dB, difiere de la línea de referencia (roja), pero a medida que elevamos el nivel de presión sonora, la curva azul se va aproximando a la de referencia.

En términos generales podemos concluir que nuestro dispositivo tiene una variación con respecto al sonómetro de referencia de unos 5 dB, lo cual, lo hace muy interesante para conocer de una manera aproximada el entorno sonoro de una zona concreta.

Capítulo 5

Conclusiones y líneas futuras

Se ha desarrollado un medidor de niveles de presión sonora ponderado A de bajo coste, basado en hardware y software libre. La mayor parte del proceso de la señal se hace de manera analógica, mediante un circuito electrónico que acondiciona la señal para su posterior digitalización. El microcontrolador Wemos D1 Mini está programado con el fin de leer el puerto analógico y enviar las muestras recogidas al servidor vía Wi-Fi.

Los resultados obtenidos han sido muy satisfactorios teniendo en cuenta que una parte del objetivo principal era desarrollar un medidor de presión sonora con un coste relativamente bajo. Además, de haber adquirido conocimientos importantes en el desarrollo de circuitos electrónicos, de impresión 3D, de programación, etc.

Las diferencias de nivel que presentan las gráficas de nuestro dispositivo con el sonómetro de referencia son debidas a varios factores; principalmente hay que destacar, la fuerte dependencia de la salida del amplificador logarítmico con la temperatura, la sensibilidad del micrófono y el método de calibración.

Además, los componentes utilizados no son ideales, las tolerancias de los componentes hacen que el proceso analógico no sea muy preciso, pudiendo alterar las señales. La lectura del puerto analógico del Wemos no es todo lo precisa que se requiere para el desarrollo de un sonómetro.

El proyecto se ha desarrollado para un ámbito educativo o de investigación, cumpliendo parte del objetivo principal propuesto al inicio del mismo.

Queda abierta la posibilidad de añadir mejoras al dispositivo tanto a nivel de software

como a nivel de hardware. A modo de mejoras, se incluye el apartado de líneas futuras.

- Circuito analizador por frecuencias, por octavas y por tercios de octava.
- Incluir diferentes tipos de ponderación, tanto en frecuencia como en tiempo.
- Pantalla OLED para la visualización de las muestras.
- Circuito con transformador para la alimentación del dispositivo, integrado en la misma PCB.
- El amplificador logarítmico esta realizado con un amplificador operacional y un diodo, pero este tipo de configuración tiene sus limitaciones. La corriente que atraviesa un diodo en polarización directa es la suma de dos factores exponenciales, uno asociado a las corrientes de difusión y otro asociado a las corrientes de generación-recombinación. Por ello, para minimizar este efecto se podría haber recurrido a diversas alternativas. Una de ellas es utilizar diodos Schottky o de germanio, cuyo comportamiento es prácticamente ideal en comparación con los de silicio (27). Otra solución, es utilizar un transistor en lugar de un diodo como se muestra en la figura 5.1.

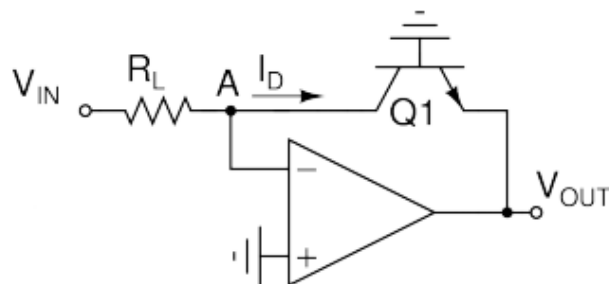


Figura 5.1: Amplificador logarítmico basado en transistor bipolar [27].

Al polarizar el transistor de esta manera se comporta como un diodo con una ventaja sobre la unión PN sencilla como podría ser la unión BE. Al intervenir la corriente de colector, la componente de difusión de la corriente I_B se ve amplificada por un factor β_F o, lo que es lo mismo, el diodo equivalente sería similar a la unión BE tras haber disminuido un factor β_F las corrientes de generación-recombinación. Así, se construye un falso diodo mucho más cercano a la idealidad.

Anexos

Anexo A

Presupuesto

Cantidad	Reference	Value	MFN	Source Price (€)
1	R1	300	Vishay / BC Components	0,153
1	R3	2.2k	Vishay/BC Components	0,153
9	R4, R2, R10, R24, R21, R18, R13, R12, R9	1k	Yageo	1,305
1	R11	50k	BI Technologies / TT Electronics	1,07
1	R8	1k	BI Technologies / TT Electronics	1,07
1	R6	10k	BI Technologies / TT Electronics	1,05
2	R5, R25	20 k	Vishay / BC Components	0,306
1	R15	6.04 k	Vishay / BC Components	0,255
3	R17, R19, R22	82.5 k	Vishay/BC Components	0,435
1	R20	150 k	Vishay / BC Components	0,153
1	R23	4k	Vishay / Dale	0,519
1	R16	56.7k	Vishay / Dale	0,094
1	R14	1.5k	Vishay / BC Components	0,128
1	R7	10k	Vishay / Dale	0,604
1	C1	1u	Vishay / BC Components	0,493
1	C3	2.2n	Vishay / BC Components	0,162
2	C4, C5	100n	Vishay / Sprague	0,17
2	C6, C7	4.7n	KEMET	0,544
1	C8	1n	Vishay / BC Components	0,153
1	C2	2.2u	Vishay / BC Components	0,374
2	C9, C10	33u	Vishay/BC Components	1,208
2	C11, C12	10u	Vishay/BC Components	0,952
3	D3, D2, D1	1N4148	ON Semiconductor/ Fairchild	0,255
2	U1, U2	TL074	Texas Instruments	1,088
1	U4	ICL7660S CPAZ	Renesas / Intersil	1,48
1	U5	LM7805C	Texas Instruments	1,31
1	U6	LM7905C	Texas Instruments	1,73
1	D4	BZX79B3V3 R0G	Taiwan Semiconductor	0,023
1	PCB	s/n	Seed Studios	3
Total Componentes:				Total € sin iva:
46				20,24 €

Anexo B

Código de programación en entorno Arduino

Se adjunta el archivo con el código Arduino y algunas explicaciones al proyecto.

```
1 #include <ESP8266WiFi.h>
2 #include <ESP8266mDNS.h>
3 #include <WiFiUdp.h>
4 #include <ArduinoOTA.h>
5 #include "Adafruit_MQTT.h"
6 #include "Adafruit_MQTT_Client.h"
7 // #define DEBUG
8
9 /***** WiFi Access Point *****/
10 #define WLAN_SSID      "MiFibra-A7EF"
11 #define WLAN_PASS      "6jNVTUYW"
12
13 /***** Server Setup *****/
14 #define AIO_SERVER      "io.adafruit.com"
15 #define AIO_SERVERPORT  1883
16 #define AIO_USERNAME    "Jimmy_Jazz"
17 #define AIO_KEY          "78ee0365382d4da282ca19727450df2f"
18
19 int sensorPIN = A0, sample, i=0;
20 int tiempo_de_envio=59999;
21 unsigned long tiempo_inicial, tiempo_final=0, tiempo_medida,
    tiempo_desde_ult_medida;
22 unsigned long periodo_medidas=125;
```

```

23 float volts, dB, promedio, suma=0;
24
25 WiFiClient client;
26
27 Adafruit_MQTT_Client mqtt(&client, AIO_SERVER, AIO_SERVERPORT
    , AIO_USERNAME, AIO_KEY);
28
29 Adafruit_MQTT_Publish muestra = Adafruit_MQTT_Publish(&mqtt
    , AIO_USERNAME"/feeds/Laeq");
30
31 void setup()
32 {
33     #ifdef DEBUG
34         Serial.begin(115200);
35         Serial.println("Adafruit_MQTT");
36         Serial.println(); Serial.println();
37         Serial.print("Conectado_a_");
38         Serial.println(WLAN_SSID);
39     #endif
40
41     pinMode(D4, OUTPUT);
42     WiFi.mode(WIFI_STA);
43     WiFi.begin(WLAN_SSID, WLAN_PASS);
44     while (WiFi.status() != WL_CONNECTED) {
45         delay(500);
46
47         #ifdef DEBUG
48             Serial.print(".");
49         #endif
50     }
51     ArduinoOTA.setPassword((const char *)"1234");
52     ArduinoOTA.begin();
53
54     #ifdef DEBUG
55         Serial.println();
56         Serial.println("WiFi_conectado");
57         Serial.println("Direccion_IP:_");

```

```

58     Serial.println(WiFi.localIP());
59 #endif
60 }
61
62 void loop()
63 {
64     ArduinoOTA.handle();
65     medida();
66     envio();
67 }
68
69
70 void medida()
71 {
72     tiempo_inicial=millis();
73     if((tiempo_inicial - tiempo_final) > periodo_medidas) {
74         sample = analogRead(sensorPIN);
75         volts = ((sample * 3.3) / 1023);
76         dB=- (102.48*pow(volts, 6))+(895*pow(volts, 5))- (3201.1*
           pow(volts, 4))+(5997.5*pow(volts, 3))- (6211.5*pow(
           volts, 2))+(3401.8*volts)-694.87;
77         suma+=dB;
78         i++;
79         tiempo_final=millis();
80     }
81 }
82
83
84 void envio()
85 {
86     tiempo_medida=millis();
87     if((tiempo_medida-tiempo_desde_ult_medida)>
           tiempo_de_envio) {
88         MQTT_connect();
89         promedio=suma/i;
90         if(muestra.publish(promedio)) {
91

```

```
92     digitalWrite (D4, HIGH);
93     delay (10);
94     digitalWrite (D4, LOW);
95
96     #ifdef DEBUG
97         Serial.println ("Muestra_enviada");
98         Serial.print ("Tiempo_desde_la_ultima_medida_(ms):")
99             ;
100        Serial.println (tiempo_medida-
101            tiempo_desde_ult_medida);
102        Serial.print ("Valor_medido_(LAeq):_");
103        Serial.println (promedio);
104    #endif
105
106    i=0;
107    promedio=0;
108    suma=0;
109    }
110    tiempo_desde_ult_medida=millis ();
111    }
112
113 void MQTT_connect ()
114 {
115     int8_t ret;
116
117     if (mqtt.connected() {
118         return;
119     }
120
121     uint8_t retries = 3;
122     while ((ret = mqtt.connect()) != 0) {
123
124         #ifdef DEBUG
125             Serial.println (mqtt.connectErrorString (ret));
126             Serial.println ("Reconectando_MQTT...");
```

```
127     #endif
128
129     mqtt.disconnect();
130     delay(1000);
131     retries--;
132     }
133
134     #ifdef DEBUG
135         Serial.println("MQTT_Conectado");
136     #endif
137
138     digitalWrite(D4, HIGH);
139     delay(500);
140     digitalWrite(D4 , LOW);
141 }
```

Anexo C

Diagrama de Gantt

Se adjunta a este proyecto un archivo Excell con el diagrama de Gantt.

Diagrama de Gantt. Sonómetro

1 de Octubre de 2017 a 4 de Febrero de 2018

Fecha inicio (dd/mm/aaaa)	Fecha fin (dd/mm/aaaa)	ACTIVIDAD	INICIO DEL PLAN	DURACIÓN DEL PLAN	INICIO REAL	DURACIÓN REAL	PORCENTAJE COMPLETADO
1/10/17	8/10/17	Ideas TFG	1	7	1	3	3,0%
9/10/17	22/10/17	Pruebas iniciales	6	14	8	10	9,9%
23/10/17	29/10/17	Busqueda de información	18	7	18	9	8,9%
30/10/17	26/11/17	Diseño circuito electrónico	25	28	25	28	27,7%
27/11/17	10/12/17	Montaje prototipo	53	14	55	10	9,9%
11/12/17	24/12/17	Diseño PCB	67	14	69	14	13,9%
8/1/18	10/1/18	Calibración	81	3	83	3	3,0%
11/1/18	19/1/18	Análisis	85	9	87	7	6,9%
20/1/18	4/2/18	Redacción TFG	94	17	96	17	16,8%
Total días:						101	

Bibliografía

- [1] AENOR. www.aenor.es. https://www.aenor.es/aenor/normas/iec/fichanormaiec.asp?codigo=36790#.Wmt_Hq7iaCg.IEC61672-1:2014
- [2] PARLAMENTO EUROPEO. www.boe.es. https://www.boe.es/diario_boe/txt.php?id=DOUE-L-2002-8128. «Directiva 2002/49/CE del Parlamento Europeo y del Consejo sobre la evaluación y gestión del ruido ambiental».
- [3] JUÁREZ, JOSUÉ ROBERTO HERNÁNDEZ. *Diseño de un sonómetro*. Instituto Politécnico Nacional. Escuela superior de Ingeniería Mecánica y Eléctrica. México 2009. Tesis. <http://tesis.ipn.mx/bitstream/handle/123456789/3972/DISENOSONOMETRO.pdf?sequence=1>
- [4] BONDERHORST, JUAN CARLOS MERLO. Ecuador, Quito. SONAC 2014. <http://www.udla.edu.ec/ojs/index.php/sonac/article/view/60>
- [5] DOMINGUEZ, DIEGO TORRES. *Desarrollo y evaluación de un sonómetro profesional para iOS*. Escuela Universitaria de Ingeniería Técnica de Telecomunicación. Universidad Politécnica de Madrid. Madrid, 2013. Proyecto fin de carrera. http://oa.upm.es/22083/1/PFC_DIEGO_TORRES_DOMINGUEZ.pdf
- [6] IBARRA, LUIS DANIEL GARCÍA. *Diseño de sonómetro digital*. Escuela superior de Ingeniería Mecánica y Eléctrica. Ingeniería en Comunicaciones y Electrónica. México, 2011. Tesis. <http://tesis.ipn.mx/jspui/bitstream/123456789/10129/1/47.pdf>
- [7] JUAN PABLO BELLO, CHARLIE MYDLARZ y JUSTIN SALAMON. *Sound analysis in Smart Cities*. M. D. Plumbley, and D. P. W. Ellis In T. Virtanen. « Computational Analysis of Sound Scenes and Events. 2017 », págs. 373-397. https://steinhardt.nyu.edu/marl/publications/zp_1_bello_chapter_2018

- [8] CHARLIE MYDLARZ, JUSTIN SALAMON y JUAN PABLO BELLO. *The Implementation of Low-cost Urban Acoustic*. New York, 2016. Investigación. <https://arxiv.org/pdf/1605.08450.pdf>
- [9] KOSTAS SOTIRAKOPOULOS, RICHARD BARHAM y BEN PIPER. *Designing and evaluating the performance of a wireless sensor network for National Physical Laboratory*. Londres, 2016. Investigación. <http://www.sea-acustica.es/fileadmin/Oporto16/28.pdf>
- [10] AMERICAN NATIONAL STANDARD ELECTROACOUSTICS. www.ansi.org. ANSI/ASA S1.4-2014/Part 1/IEC 61672:1-2014. 2014. <https://webstore.ansi.org/RecordDetail.aspx?sku=ANSI%2FASA+S1.4-2014%2FPart+1+%2F+IEC+61672%3A1-2013>
- [11] RAMIRO, CRESPO YÁNEZ DIEGO. *Diagnóstico de un ambiente laboral mediante el diseño y construcción de un dispositivo portátil evaluador de CO2 y ruido para el área industrial de la Espel*. Escuela Politécnica del ejército. Latacunga, Ecuador, 2011. Proyecto fin de carrera. <http://repositorio.espe.edu.ec/bitstream/21000/4001/1/T-ESPEL-0775.pdf>
- [12] MAGGIOLO, DANIEL. [Citado el: 10 de Enero de 2018]. <http://www.eumus.edu.uy/docentes/maggiolo/acuapu/son.html>
- [13] MARCIA ELIZABETH YUCTA INGUILLAY y EDUARDO GIOVANNY TENEZACA CARPIO. *Diseño e implementación de un prototipo multimodal sonómetro para medir la frecuencia del nivel sonoro en la ciudad de Riobamba*. Escuela Politécnica de Chimborazo. Facultad de Informática y Electrónica. Escuela de Ingeniería electrónica en Telecomunicaciones y Redes. Riobamba, Ecuador, 2016. Proyecto fin de carrera. <http://dspace.espoch.edu.ec/bitstream/123456789/6379/3/98T00111.pdf>
- [14] AENOR. www.aenor.es. 16 de Octubre de 2016. [Citado el: 11 de Enero de 2018]. <http://www.aenor.es/aenor/normas/normas/fichanorma.asp?tipo=N&codigo=N0043828#.Wm6Iaa7iaCh>
- [15] ARDUINO. [Citado el: 12 de Enero de 2018]. <https://www.arduino.cc>
- [16] GARCÍA, JON.ARDUINO. ¿Que es Arduino? 2014. [Citado el: 12 de Enero de 2018]. <http://spacediy.es/?p=266>
- [17] WEMOS. [Citado el: 12 de Enero de 2018]. <https://www.wemos.cc>

- [18] ESPloradores. [Citado el: 12 de Enero de 2018]. <http://www.esploradores.com/recursos/doitwemos-d1-mini-nodemcu/>
- [19] Datasheet WM-61A Electret. Omnidirectional Back Electret Condenser Microphone Cartridge. [Citado el: 9 de Noviembre de 2017]. <https://media.digikey.com/pdf/Data%20Sheets/Panasonic%20Electronic%20Components/WM-61A.pdf>
- [20] Datasheet TL071CP. Texas Instruments. [Citado el: 8 de Diciembre de 2017]. <http://www.ti.com/lit/ds/symlink/tl071.pdf>
- [21] Kicad EDA. Kicad. [Citado el: 15 de Diciembre de 2017]. <http://kicad-pcb.org>
- [22] DR. ING. MIGUEL ÁNGEL GANTUZ y ING. IGNACIO PEACOCK. Sonómetro graficador en tiempo real. Facultad de Ingeniería. Universidad de Mendoza. www.um.edu.ar/ojs-new/index.php/FAI/article/download/798/787.
- [23] Instituto Argentino de Normalización y certificación. IRAM. 5 de Junio de 1988. [Citado el: 4 de Diciembre de 2017]. <http://web.iram.org.ar/index.php?vernorma&id=0D3A.IRAM-4074-1>.
- [24] LTSpice. LTSpice IV XVII Build. 14 de Abril de 2017. [Citado el: 26 de Noviembre de 2017]. <http://ltspice-iv.en.lo4d.com/>.
- [25] Datasheet TL074CN. Texas Instruments. [Citado el: 8 de Diciembre de 2017]. <http://www.ti.com/lit/ds/symlink/tl074.pdf>.
- [26] NAVE, M OLMO R. *Semiconductores en Electrónica*. Department of Physics and Astronomy. Georgia State University. Atlanta, Georgia, 2001. Académico. <http://hyperphysics.phy-astr.gsu.edu/hbasees/Solids/pnjun.html>.
- [27] PELÁEZ, FRANCISCO J. FRANCO. *Electrónica Analógica: aplicaciones no lineales de los amplificadores operacionales*. Facultad de Física de la Universidad Complutense de Madrid. Madrid. Académico. https://cv3.sim.ucm.es/access/content/group/portal-uatducma-43/webs/material_original/apuntes/PDF/08_Aplicaciones_no_lineales_opamp.pdf.

- [28] Datasheet ICL7660S. Renesas. [Citado el: 22 de Enero de 2018]. <https://www.mouser.com/ds/2/698/icl7660s-a-1278394.pdf>.
- [29] Datasheet LM7905C. Texas Instruments. [Citado el: 22 de Enero de 2018]. <http://www.ti.com/lit/ds/symlink/lm79.pdf>.
- [30] Datasheet LM7805. Texas Instruments. [Citado el: 22 de Enero de 2018]. <http://www.ti.com/lit/ds/symlink/lm340.pdf>.
- [31] Seeed. Seeed Studios. [Citado el 21 de Diciembre de 2017]. <https://www.seeedstudio.com>
- [32] Autodesk. Inventor. [Citado el: 25 de Enero de 2018]. <https://www.autodesk.es/products/inventor/overview>.
- [33] Arquitectura y Gestión de la IoT. GONZÁLEZ, DANA RODRÍGUEZ. No. 3, s.l.: ISSN 1729-3804, septiembre-diciembre de 2013, Revista Telem@tica, Vol.12, págs. 49-60.
- [34] ESPloradores. [Citado el: 3 de Enero de 2018]. <http://www.esploradores.com/practica-9-modos-de-ahorro-de-energia-deep-sleep/>
- [35] Prometec. [Citado el: 16 de Enero de 2018]. <https://www.prometec.net/esp8266-programacion-ota/>.
- [36] NAVE, M OLMO R. Concepto de reproducción de sonido. Department of Physics and Astronomy. Georgia State University. Atlanta, Georgia, 2001. Académico. <http://hyperphysics.phy-astr.gsu.edu/hbasees/Audio/equal.html>
- [37] DATASHEET 4292-L Brüel & Kjaer. Sound Sources and Impact Sound Source for Building Acoustics. 2013. Datasheet 4292-L. <https://www.bksv.com/-/media/literature/Product-Data/bp1689.ashx>.
- [38] AENOR. 28 de Octubre de 2014. [Citado el: 12 de Enero de 2018]. <http://www.aenor.es/aenor/normas/normas/fichanorma.asp?tipo=N&codigo=N0033866#.WnNLn67iaCg>.
- [39] ADAFRUIT. [Citado el: 11 de Diciembre de 2017]. <http://io.adafruit.com>.

