

Trabajo Fin de Grado

Desarrollo de un sistema automático de medida con sensores acústicos, térmicos y de presión para modelar el comportamiento de un recipiente con agua hirviendo.

Development an automatic measurement system with acoustic, thermal and pressure sensors to model boiling liquid's behaviour.

Autor/es

Daniel García Padilla

Director/es

José Ramón Beltrán Blázquez.
David Diaz-Guerra Aparicio

Grado en Ingeniería Electrónica y Automática
Escuela Ingeniería y Arquitectura, Zaragoza.
2020/2021

Desarrollo de un sistema automático de medida con sensores acústicos, térmicos y de presión para modelar el comportamiento de un recipiente con agua hirviendo.

Resumen

Basado en la necesidad de incrementar las prestaciones de una placa de inducción, el fin de este trabajo es proporcionar una base de datos destinada a entrenar una red neuronal que sea capaz de distinguir la temperatura de un líquido hirviendo sin necesidad de introducir sensores de temperatura en el interior del recipiente. Para ello, se crea un set experimental para obtener datos simultáneamente de temperatura, sonido y presión acústica del sistema con el objetivo de analizar la viabilidad de estimar la temperatura únicamente por medio de micrófonos o sensores de presión.

En este trabajo nos centramos en el modelado del sistema, la creación del set experimental y el desarrollo de un sistema automático de medida que integre los distintos dispositivos de captura de datos para su captura simultánea. Además, realizamos un análisis preliminar de los datos obtenidos y el alcance y la problemática que puede tener el añadir la red a la cocina.

Development an automatic measurement system with acoustic, thermal and pressure sensors to model boiling liquid's behaviour.

Abstract

Based on the need to increase the performance of an induction cooktop, the aim of this work is to provide a database to train a neural network capable of distinguishing the temperature of a boiling liquid without the need to introduce temperature sensors inside the vessel. For this purpose, an experimental set is created to simultaneously obtain temperature, sound and acoustic pressure data from the system in order to analyze the feasibility of estimating the temperature only by means of microphones or pressure sensors.

In this work we focus on the modeling of the system, the creation of the experimental set and the development of an automatic measurement system that integrates the different data capture devices for simultaneous data capture. In addition, we performed a preliminary analysis of the data obtained and the scope and problems that may arise from adding the network to the kitchen.

Índice

Resumen	2
Abstract	3
Tabla de figuras	5
Capítulo 1. Introducción.	6
Capítulo 2. Modelado del comportamiento del sistema.	8
2.1 Comportamiento general.	8
2.2 Fuentes de sonido en nuestro sistema.	10
Capítulo 3. Diseño del set experimental	12
3.1 Sistema de captura de datos y dispositivos.	12
3.2 Estructura inicial.	15
3.3 Análisis preliminar	16
3.4 Estructura final	19
Capítulo 4. Implementación y programación del sistema de medida.	20
4.1 Problemática y limitaciones de Matlab	20
4.2 Estructura final de la aplicación:	21
Capítulo 5. Análisis global y conclusiones.	26
5.1 Análisis global	26
5.2 Resumen y conclusiones:	33
Bibliografía.	34
Anexo. Código de la aplicación y diagramas de flujo.	35
Script principal.	35
Función Captura de datos.	38
Función Captura de audio.	44

Tabla de figuras

Figura 1. <i>Fases de la ebullición de un líquido</i>	8
Figura 2. <i>Esquema ebullición de transición</i>	9
Figura 3 <i>Esquema ebullición de película.</i>	9
Figura 4. <i>Identificación de las tres primeras fases en el audio capturado y espectrograma.</i>	11
Figura 5. <i>Dispositivos de captura de datos inicial</i>	12
Figura 6. <i>Dispositivos de captura de datos final</i>	13
Figura 7. <i>Vista cenital del set experimental</i>	15
Figura 8. <i>Vista lateral set experimental.</i>	15
Figura 9. <i>Identificación inicio de las fases según la temperatura.</i>	16
Figura 10. <i>Comparativa señal micrófono y piezoeléctrico</i>	17
Figura 11. <i>Temperatura NTCs internas y Temperaturas recogidas por los termopares.</i>	18
Figura 12. <i>Disposición final del dataset utilizada para obtener la base de datos.</i>	19
Figura 13. <i>Estructura simplificada del programa</i>	21
Figura 14. <i>Resumen de la aplicación de medida</i>	25
Figura 15. <i>Análisis conjunto de velocidad en alcanzar la Temperatura de saturación.</i>	26
Figura 16. <i>Análisis conjunto de la duración de la primera fase de nucleación</i>	26
Figura 17. <i>Espectrograma de una de las grabaciones</i>	27
Figura 18. <i>Análisis del espectro de cada una de las fases</i>	28
Figura 19 <i>Análisis del valor RMS en toda la grabación.</i>	29
Figura 20. <i>Comparativa señal del piezoeléctrico interno y los piezoeléctricos en la superficie</i>	30
Figura 21. <i>Comparativa frecuencial de las señales del micrófono y piezoeléctrico interno.</i>	31
Figura 22. <i>Detalle del pico en 50Hz en la señal recogida por el piezoeléctrico interno.</i>	32

Capítulo 1. Introducción

Este trabajo es parte de un proyecto, realizado en colaboración con la empresa BSH Electrodomésticos, que tiene como finalidad estudiar la viabilidad de dotar a una placa de inducción de la capacidad de poder distinguir, sin necesidad de sensores específicos de temperatura, la temperatura de un líquido al hervir. El objetivo final del proyecto es intentar predecir la temperatura a la que se encuentra el líquido utilizando técnicas de aprendizaje profundo contando únicamente con los datos recogidos por sensores acústicos y de presión.

El proyecto ha sido desarrollado en los laboratorios de Imagen y Sonido de la Escuela de Ingeniería y Arquitectura de la Universidad de Zaragoza, donde se encuentra la placa de inducción con la que se han tomado las medidas.

Este trabajo constará de todo lo necesario para obtener los datos necesarios para entrenar la red neuronal, pero no entra en las técnicas de aprendizaje profundo utilizadas. Por lo tanto, se centra exclusivamente en el modelado del comportamiento del sistema, la creación del set experimental para la captura de datos, la integración de los dispositivos de recogida de datos, y el análisis de estos.

El primer paso a realizar ha sido un pequeño modelado del sistema y un estudio de la problemática, luego se ha diseñado el set experimental con el que hemos capturado los datos y finalmente se han realizado las mediciones para después analizar los datos.

Los dispositivos con los que se han trabajado han sido:

- Una tarjeta de sonido de 8 canales donde recogemos el sonido tanto de un micrófono como de cuatro piezoeléctricos,
- UDA (Universal Diagnosis Adapter): Un dispositivo de captura de datos internos de la propia cocina que nos proporcionaba datos como la temperatura interna y la corriente y potencia suministrada a las bobinas.

-Datalogger: Un dispositivo para captura de datos externos como temperaturas externas, diferencias de potencial, corrientes... etc. Para realizar la base de datos se sustituyó por un Pico.

-Pico TC-08. Dispositivo de captura de temperaturas.

Para el sistema de medida y el análisis posterior hemos desarrollado una aplicación con MatLab en el que se toman datos simultáneamente de los distintos dispositivos, se procesan y se analizan. También hemos necesitado otras herramientas como AppDesigner (de Matlab) para la realización de una interfaz de usuario amigable por si hiciera falta realizar futuras medidas, Audacity (software de grabación y procesado de audio multicanal) o CueMix (para la configuración de la tarjeta de sonido).

Este informe recorre todo el proceso y se estructura de la siguiente manera:

Cap 1: Introducción.

Cap 2: Modelado y comportamiento del sistema

Cap 3: Diseño del set experimental y análisis preliminar

Cap 4: Diseño del algoritmo

Cap 5: Análisis global y conclusiones.

Capítulo 2. Modelado del comportamiento del sistema.

2.1 Comportamiento general.

Para desarrollar tanto el diseño del set experimental como el del programa para realizar las grabaciones es necesario entender cómo se comporta el sistema que queremos estudiar [1], [2], [3]. Para ello, es necesario entender que el proceso de ebullición de un líquido pasa por diversas fases que dependen de lo que llamamos exceso de temperatura]. Este exceso es la diferencia entre la temperatura de la superficie caliente del recipiente que contiene el líquido y la temperatura de saturación de este, que en nuestro caso son 100°C , ya que el líquido de prueba va a ser agua exclusivamente.

Como se puede observar en la figura 1, conforme proporcionamos calor al recipiente, primero entran en juego fenómenos únicamente de convección, por lo que el líquido sobrecalentado sube a la interfase líquido-vapor y se evapora (Punto A). Llegados a este punto, los gases disueltos en el líquido condensan formando burbujas, en un fenómeno que se llama ebullición nucleada. Dependiendo de la energía que adquieren, las burbujas explotan debajo de la superficie del líquido (zona A-B) o consiguen llegar a la superficie (zona B-C).

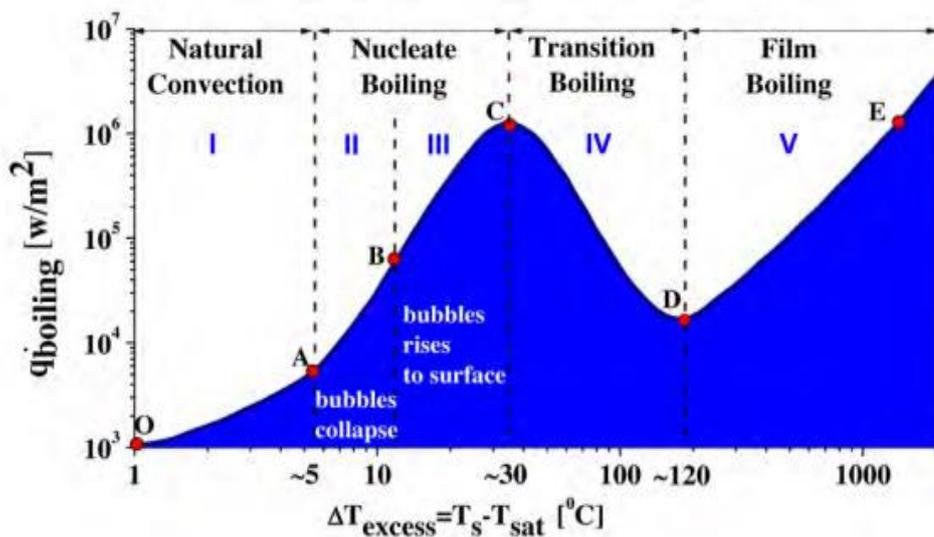


Figura 2. Fases de la ebullición de un líquido [3]

Cuando el exceso de temperatura ronda unos 30°C de diferencia entre la temperatura de la superficie del recipiente y la de saturación del líquido y se supera un cierto valor llamado flujo de calor crítico CHF (correspondiente con el punto C de la gráfica) entramos en la zona de transición de ebullición. El vapor producido tiene una conductividad menor que la del líquido, por lo que puede llegar a formar una película en el fondo del recipiente que actúa como aislante térmico. Esto produce que pasado el punto CHF el coeficiente de transferencia de calor disminuye y la ebullición se vuelve inestable.

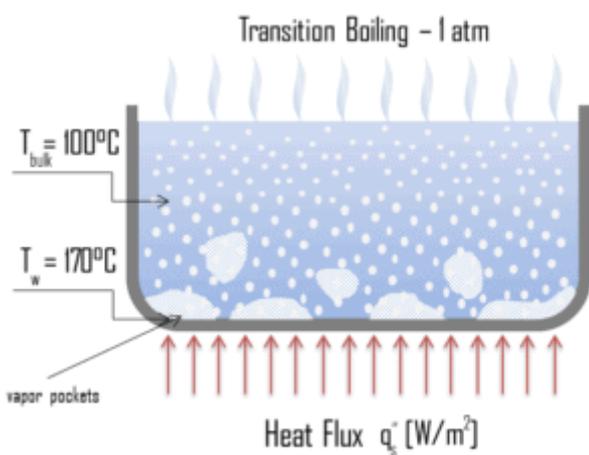


Figura 2. Esquema ebullición de transición. [9]

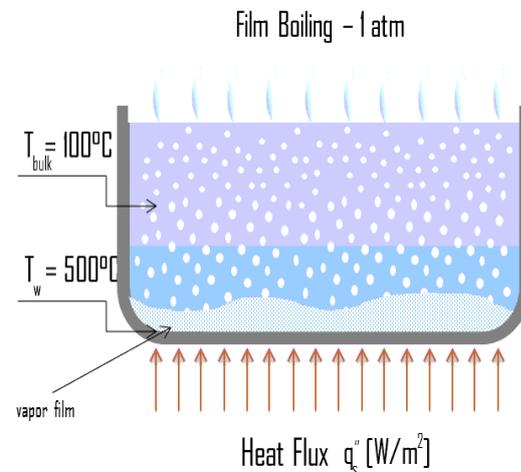


Figura 3 Esquema ebullición de película. [10]

Un aumento adicional del flujo de calor provoca que se forme completamente una película en la superficie del recipiente una vez que llegamos al punto D (llamado punto de Leidenfrost), reduciendo considerablemente el coeficiente de convección e impidiendo el contacto entre la superficie y la fase líquida pasando a transferir el calor mayoritariamente por radiación.

En nuestro caso, y debido a que la potencia que podemos ofrecer es limitada, como se ha comprobado en experimentos previos realizados para diseñar el set experimental, únicamente llegaremos a sobrepasar por poco la zona de ebullición nucleada, por lo que las zonas de estudio serán las comprendidas desde el punto O hasta el C. Para comprender las fuentes de sonido que tienen lugar en el proceso, explicamos en detalle nuestro caso a continuación.

2.2 Fuentes de sonido del sistema.

Como hemos explicado, únicamente estudiamos las tres primeras fases: evaporación en la interfase y los dos tipos de ebullición nucleada (es decir, zonas I, II, III del gráfico de la figura 1).

Entendemos nuestro sistema físico como un único recipiente que contiene agua, al que se le va proporcionando calor en su base y no tiene más entradas ni perturbaciones que la fuente de calor.

Conforme se proporciona calor, el recipiente va incrementando su temperatura y transfiriendo calor al líquido, calentándolo. El líquido, por tanto, entra en estado de ebullición en donde podemos diferenciar dos fases:

- Ebullición subenfriada o local: donde la temperatura del líquido se encuentra por debajo de la temperatura de saturación. En este caso, como el líquido es agua, esta fase se dará cuando la temperatura del agua esté por debajo de los 100°C.
- Ebullición saturada o global: donde la temperatura del líquido ya ha alcanzado la temperatura de saturación y se queda constante en esos 100°C.

En nuestro caso, podemos distinguir sonidos que se corresponden a los distintos estados por los que pasa el líquido en el proceso [2],[3]. Conociendo estos estados, podemos conocer las fuentes de ruido del sistema que son principalmente debidas a tres fenómenos: nucleación, cavitación y el estallido de burbujas en la superficie.

- Nucleación: Al proporcionar el calor en la base existe un gradiente de temperaturas en todo el recipiente, siendo la parte inferior la que está a mayor temperatura. Conforme aumenta la temperatura, la solubilidad de los gases disueltos en el agua disminuye y se forman burbujas en el fondo del recipiente. Cuando estas burbujas alcanzan un determinado tamaño, se liberan del fondo del recipiente y ascienden hacia capas superiores del líquido. Esta es la primera causa de la aparición del sonido al hervir agua y produce un sonido bajo en intensidad y con componentes de baja frecuencia.

- Cavitación: Como hemos dicho, la base del recipiente se encuentra a 100°C, lo que hace posible que se formen burbujas. A medida que la burbuja asciende, pasa a capas de menor temperatura perdiendo energía en el proceso y empieza a colapsar antes de llegar a la superficie y finalmente, estalla. Cuando estalla, deja un hueco en el agua que el líquido intenta rellenar rápidamente, chocando entre sí y produciendo un sonido más intenso que el anterior.
- Estallido de superficie: Cuando la temperatura de todo el líquido está próxima a los 100°C, las burbujas tienen la suficiente energía como para ascender hasta la superficie. La explosión, ahora, se produce en la superficie de forma suave, a diferencia del proceso de cavitación (que se producía de forma violenta y por debajo de la superficie) por lo que el sonido que se produce se atenúa.

Como vemos en la figura 4, al capturar el audio durante todo el proceso de ebullición, se diferencian las tres zonas perfectamente:

-(1) la zona en la que el líquido se está calentando y se empiezan a crear las burbujas en el fondo. El audio es poco intenso y con una frecuencia baja.

-(2) la zona donde predomina el fenómeno de nucleación y las burbujas estallan sin llegar a la superficie. Vemos que el sonido es más intenso y se forma una “una burbuja” en la señal de audio.

-(3) la zona donde las burbujas tienen la suficiente energía para explotar al llegar a la superficie, donde vemos que se suaviza el audio.

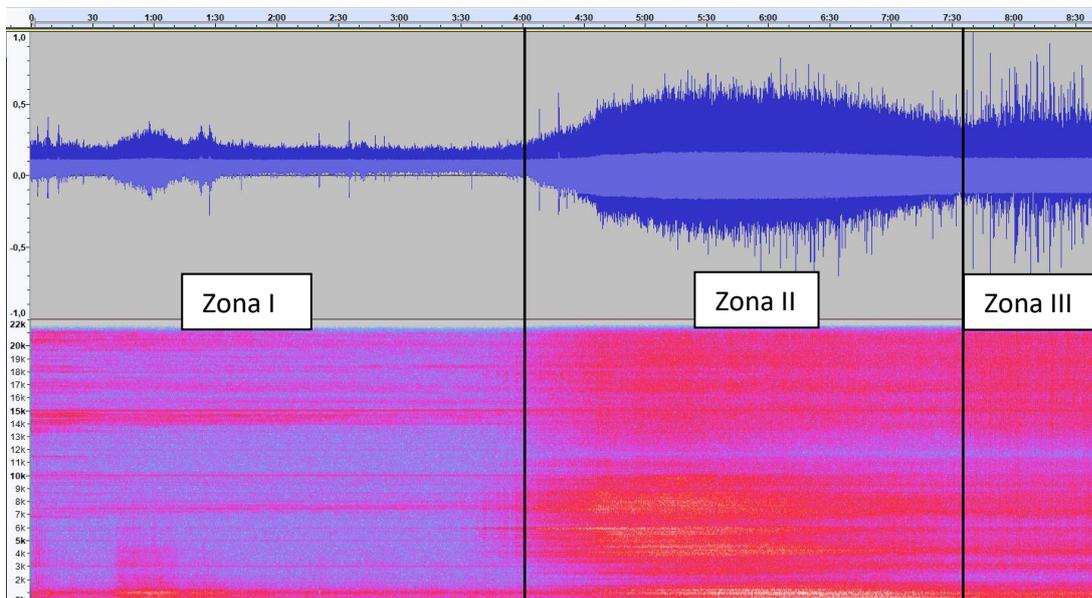


Figura 4. Identificación de las tres primeras fases en el audio capturado y espectrograma.

Capítulo 3. Diseño del set experimental

3.1 Sistema de captura de datos y dispositivos.

Con el problema físicamente entendido, pasamos a diseñar el set experimental que nos servirá para obtener la base de datos. Para ello, el objetivo principal es ser capaces de capturar el sonido que se produce en cada fase de la ebullición y poder correlarlo con la temperatura a la que se produce, pudiendo determinar, al menos, el inicio y el fin de cada fase. Por lo tanto, necesitamos disponer de un sistema de captura de audio y otro de captura de temperaturas.

El sistema de captura de audio consiste en un micrófono Behringer B5 [4] y una tarjeta de sonido MOTU 8 PRE [5], con 8 canales preamplificados y alimentación *phantom* como interfaz entre el PC y el micrófono. El Behringer B5 es un micrófono de condensador omnidireccional con un circuito de entrada FET que elimina la distorsión de baja frecuencia. Además, para analizar si podría ser viable implementar una realimentación acústica prescindiendo del uso de micrófonos, añadimos 4 piezoeléctricos ADELINE AD-35 TRANSDUCTER, conectados también a la tarjeta de



Figura 5. Dispositivos de captura de datos inicial

sonido. Elegimos estos piezoeléctricos por su disponibilidad y porque al ser necesario utilizar una tarjeta de sonido previa, poder conectarlos directamente a esta hace mucho más fácil el sistema de detección de audio. Los piezoeléctricos se han usado pegados al cristal de la cocina para poder medir su vibración y, además, uno se ha pegado por debajo del mismo, ya que, si la información capturada por éste resulta suficiente, permitiría un diseño integrado en el interior de la placa de inducción sin elementos en su exterior.

Además del audio, necesitamos correlar con la temperatura. Para capturarla, en un primer diseño utilizamos un dispositivo de adquisición de datos AGILENT DATALOGGER, [6] al que conectamos tres termopares tipo K: uno para medir la temperatura de la superficie externa de la placa, otro para la temperatura de la base del recipiente y una sonda que nos proporcionaba la temperatura del agua. También nos proporcionaba la posibilidad de capturar algún dato externo a la cocina en el resto de canales si fuera necesario. El gran inconveniente de este dispositivo era el ruido que producía cuando actuaba el relé cada vez que se tomaba una muestra, que provocaba

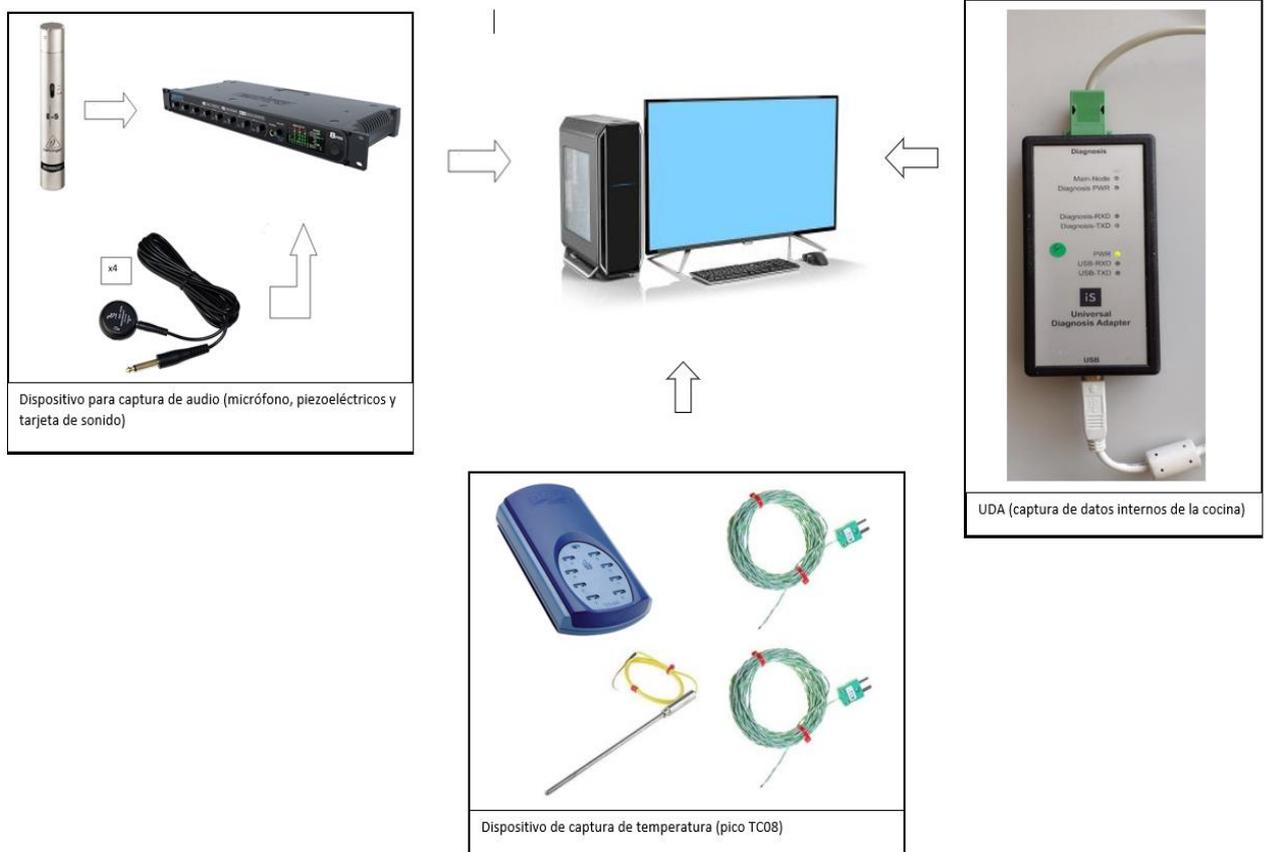


Figura 6. Dispositivos de captura de datos final

interferencias en la grabación de forma periódica. Con este dispositivo únicamente realizamos un análisis preliminar de los datos que podíamos recoger, mitigando este sonido lo máximo posible y, tras unos primeros experimentos, al ver que no podíamos capturar el audio del proceso limpio y no era necesario utilizar más canales, se decidió sustituirlo por una capturadora de datos para termopares PICO TC08 [7], al que se le conectaron los termopares K, obteniendo así únicamente las temperaturas y sin interferir en las grabaciones.

También capturamos la potencia que se está entregando a los inductores para tener una idea del calor que se está aplicando, ya que afecta a la duración de las fases de la ebullición así como otros datos de corriente o temperatura de las NTC's situadas debajo de la placa con un dispositivo de adquisición de datos llamado U.D.A. (Universal Diagnosis Adapter). La potencia y la corriente la relacionamos con el audio y la temperatura de las NTCs internas la recogemos para estudiar cómo disipa el calor la placa, aunque como se ve en el apartado 3.3 de este capítulo, debido a la deriva térmica, la información que podemos obtener de las NTCs no puede usarse de modo sencillo para estimar la temperatura del agua.

3.2 Disposición inicial.

Con todo esto, la disposición para los primeros experimentos fue la que muestra la figura 7.

La placa con la que se cuenta es una placa Flex de Inducción con sistema integrado de ventilación de la marca Gaggenau, con número de serie CV282100 [8]. En cada lado del extractor, la cocina cuenta con cuatro bobinas de las cuales solo usamos las dos indicadas en la figura con rectángulos rojos. Los piezoeléctricos se ubicaron en los límites de las bobinas para estudiar la vibración de toda la placa cercana al recipiente.

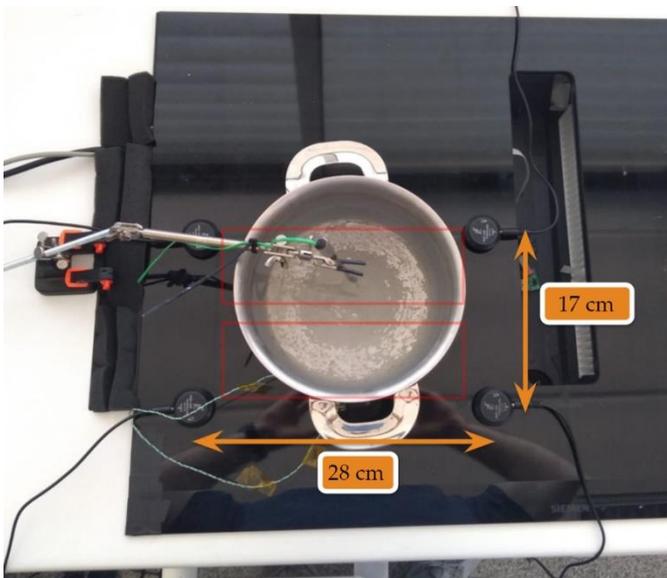


Figura 7. Vista cenital del set experimental

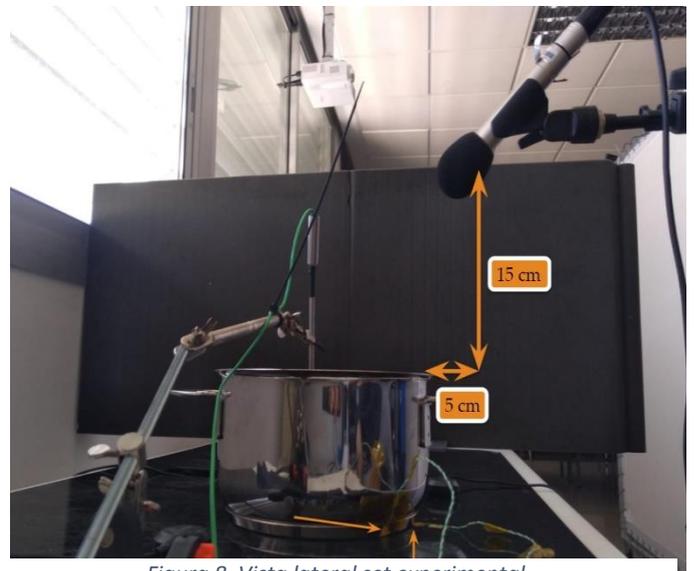


Figura 8. Vista lateral set experimental.

El micrófono que estamos utilizando tiene un patrón de direccionalidad cardiode, por lo que captura más señal en la dirección hacia la que se apunta. Por lo tanto, lo ideal sería colocarlo encima de la cazuela apuntando hacia abajo. Para evitar que la humedad lo estropeará, lo situamos a 15 cm de altura y separándolo 5 cm de la vertical del recipiente con un ángulo de 45°, consiguiendo así que grabara de la mejor manera posible sin que los gases afectasen al micrófono. La disposición de la sonda y de los termopares se refleja en la figura 8. La sonda queda en contacto con el agua y los termopares se colocan de forma que el destinado a medir la temperatura de la superficie (1) quede lo más cerca del recipiente posible, pero sin estar en contacto con

él, y el destinado a medir la temperatura del recipiente (2) se coloca en la base ya que es la zona donde mejor conduce el calor.

3.3 Análisis preliminar

Con todos los dispositivos integrados y una vez desarrollado el algoritmo que explicamos en el capítulo 4, algunos resultados obtenidos de los primeros experimentos son los siguientes:

Como primera observación inicial, una vez conseguido capturar audio y temperatura, en la figura 9 observamos lo esperado teóricamente: los inicios de la fase de nucleación se dan cuando el recipiente se encuentra a 100°C y el inicio de la segunda fase de nucleación, es decir el estallido de las burbujas en la superficie, se da cuando el agua alcanza los 100°C de la temperatura de saturación.

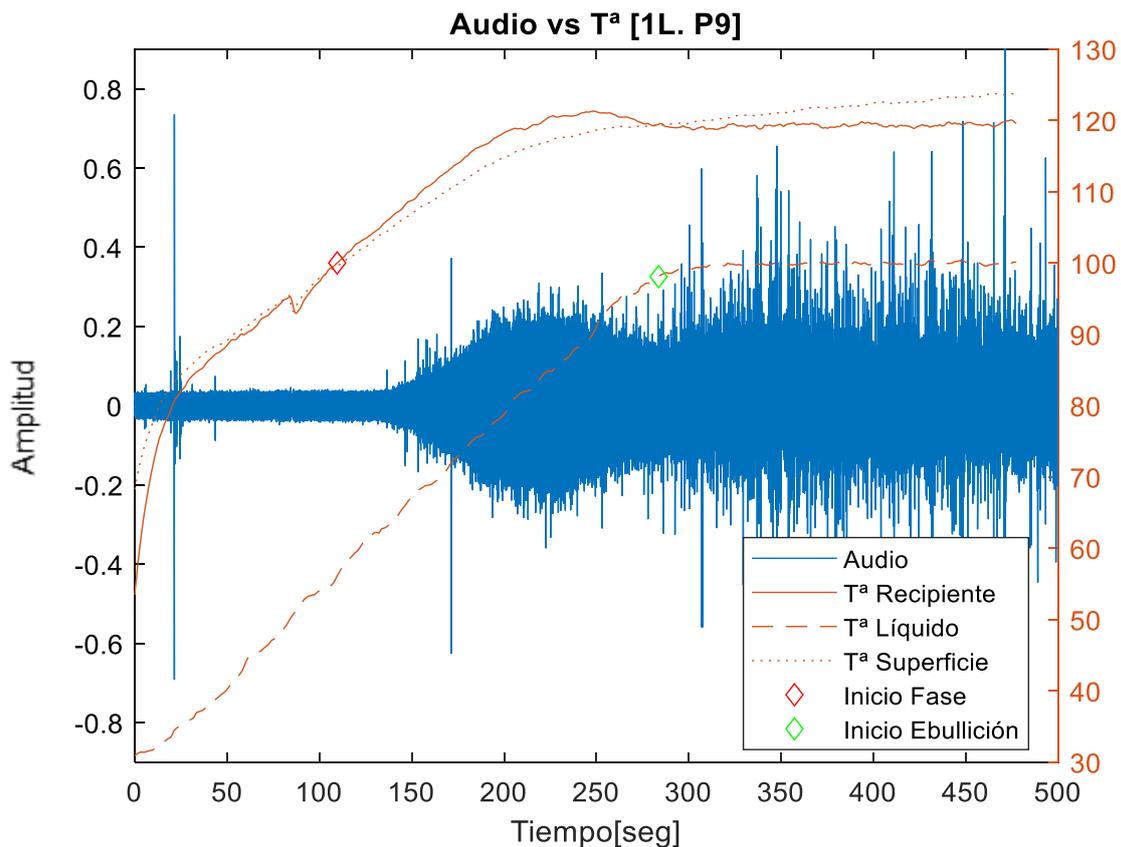


Figura 9. Identificación inicio de las fases según la temperatura.

Además, podemos observar lo ya comentado en el apartado de teoría: claramente, a la máxima potencia que podemos aplicar, la diferencia de temperatura entre la superficie de la placa (120°C) y la temperatura de saturación del líquido (100°C) ronda los 20 grados de diferencia, por lo que según lo expuesto en la figura 1, no llegamos a ese exceso de temperatura de 30°C marcado como el inicio de la fase de ebullición de película y por lo tanto solo podemos estudiar las tres primeras fases: la fase donde domina la convección natural y las dos fases de la ebullición nucleada.

De la misma manera, podemos ver en la figura 10 que las señales recogidas por los piezoeléctricos colocados en el exterior y la del micrófono tienen formas de onda similares:

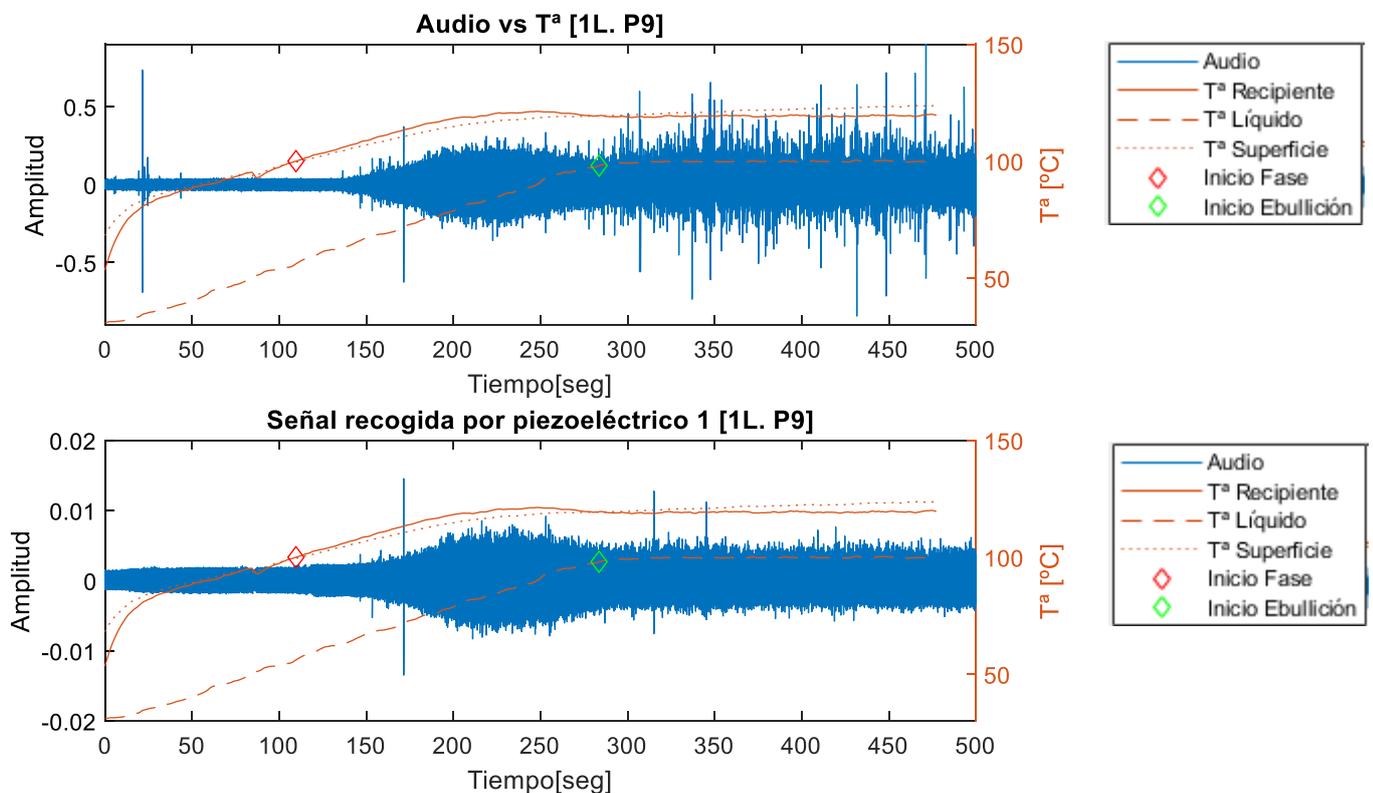


Figura 10. Comparativa señal micrófono y piezoeléctrico

Por lo que proponemos modificar la estructura del set experimental y estudiar si, de cara a una implementación final y como mejora de las prestaciones de la cocina, podemos sustituir el micrófono por un piezoeléctrico alojado en el interior de la placa para recoger información acústica de forma mucho más barata y fácil de incluir en la cocina. Es por ello que la base de datos se recoge con la estructura final que explicamos en el siguiente apartado del capítulo.

En cuanto a la temperatura de las NTCs, podemos observar que se encuentran a temperatura ambiente únicamente cuando la cocina lleva tiempo sin encenderse. En el momento que se hace uso de ella, debido a la deriva térmica, se enfrían muy lentamente y se contamina cualquier medida real que podamos obtener de ellas una vez encendida. Como observamos en la figura 11, se ve claramente cómo la temperatura de las NTCs de la zona correspondiente a los inductores que estamos utilizando (líneas amarilla y morada) están a temperaturas muy altas y no tienen ninguna dependencia con la temperatura de la superficie de la placa, mientras que la NTCs más alejada de esta (línea azul) está a temperatura ambiente. Además, cuando se estabiliza la temperatura del líquido en la temperatura de saturación, la temperatura de las NTCs sigue aumentando.

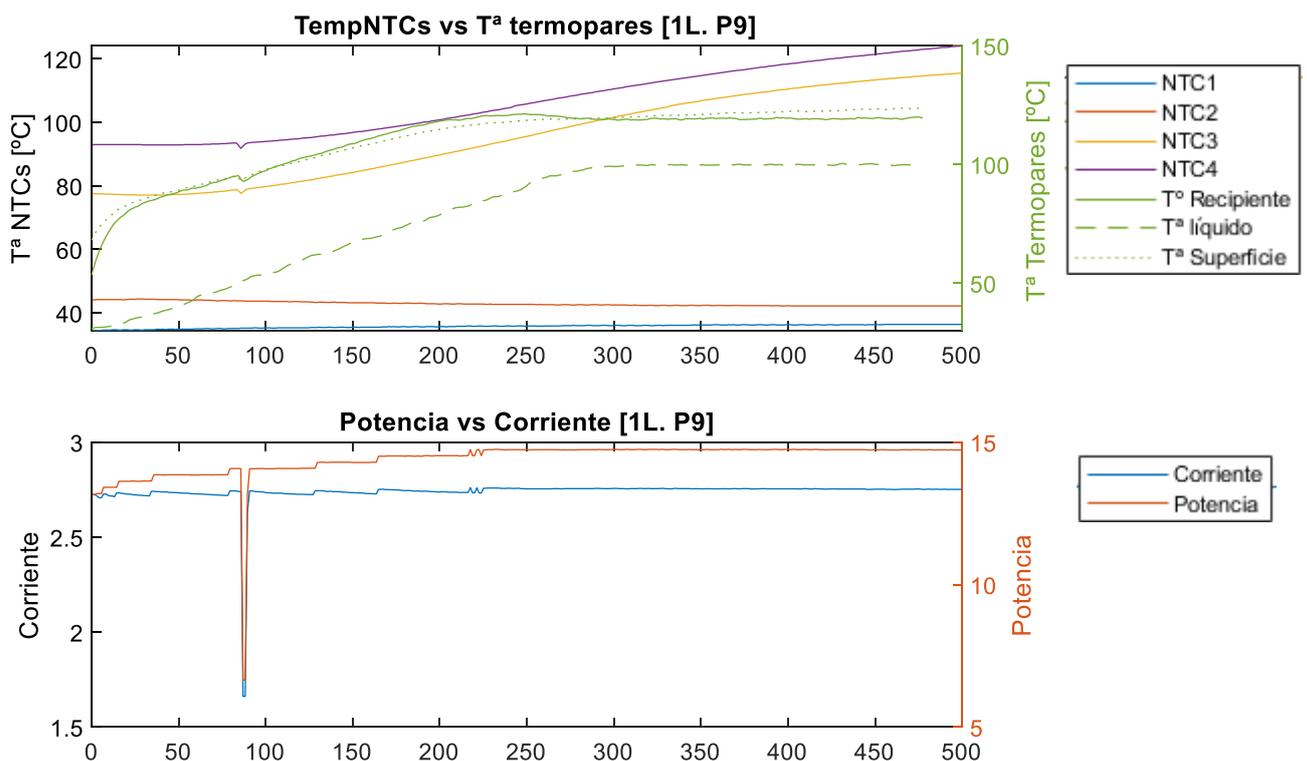


Figura 11. Temperatura NTCs internas y Temperaturas recogidas por los termopares.

Por lo tanto, con todo lo anterior, podemos concluir que la información proporcionada por las NTCs internas no nos permite estimar la temperatura del líquido, ni podemos correlarla con la temperatura de la superficie debido a la deriva térmica de las propias NTCs y el carácter aislante de la placa.

También, para dotar a la base de datos de la máxima variabilidad posible, se decide realizar las grabaciones con tres tipos de recipientes distintos entre sí, tanto de tamaño como de materiales. Para cada uno de los tres recipientes se grabará un proceso de ebullición distinto con potencias 7.5, 8, 8.5 y 9 y se repetirá este proceso con diferentes cantidades de líquido: 1l., 1.5l. y 2l.

3.4 Disposición final

Una vez hecho el análisis preliminar, introduciendo el piezoeléctrico de la esquina inferior derecha de la figura 7 en el interior, y sustituyendo el Datalogger por el Pico TC08, la disposición final del set experimental y con la que realizamos la base de datos es la mostrada en la figura 12. El círculo rojo indica la posición final del piezoeléctrico interior y el círculo amarillo la posición donde se encontraba en los primeros análisis.



Figura 12. Disposición final del dataset utilizada para obtener la base de datos.

Capítulo 4. Implementación y programación del sistema de medida.

Para integrar todos los dispositivos utilizados y automatizar el sistema de medida hemos programado una aplicación que nos permite dar a todos los datos una misma referencia de tiempos y gestionarlos para unificar las medidas independientemente del dispositivo que las ha obtenido. Para ello, explicamos la problemática y la estructura final de la aplicación.

4.1 Problemática y limitaciones de Matlab

Aunque nos encontramos con problemas a la hora de integrar los distintos dispositivos, el problema principal lo encontramos al tener que desarrollar la aplicación en Matlab ya que, al tener que realizar grabaciones multicanal para recoger las señales del micrófono y de los 4 piezoeléctricos, hubo que encontrar la manera de grabar continuamente en todos los canales mientras, simultáneamente, obtenemos datos del resto de dispositivos. Las opciones que nos da Matlab para capturar audio y datos simultáneamente solo son válidas para grabar en mono y estéreo, por lo que no es posible grabar las cinco pistas que necesitamos de forma simultánea y a la vez estar capturando los datos de temperatura, potencia y corriente.

Debido a la secuencialidad del propio Matlab, y ante la falta de herramientas para esto, únicamente podíamos grabar y una vez finalizada la captura de audio, comenzar a recoger datos. Para solventarlo, utilizamos la “Paralell Toolbox” de Matlab, que nos permite usar a la vez varios núcleos del procesador del computador. En nuestro caso, al trabajar con un ordenador que dispone de un microprocesador Intel i3, disponemos de dos núcleos. Esto quiere decir que, haciendo uso de esta Toolbox, nos permite trabajar con los dos núcleos al mismo tiempo y dividir el algoritmo en dos tareas principales que se llevan a cabo a la vez: Capturar datos y Capturar audio. De esta manera podemos grabar independientemente las 5 pistas de audio sin que interfieran en la captura de datos y dándoles a todos los dispositivos una referencia de tiempos común.

Además, a la hora de sustituir el DATALOGGER por el PICO TC08, nos encontramos con problemas a la hora de gestionar la captura de datos. Hasta el momento, en el momento que solicitábamos un dato al UDA, se solicitaba a la par un dato de temperatura del DATALOGGER. El PICO TC08 nos permite trabajar con dos modos de trabajo: asíncrono (donde captura una muestra en cada canal justo en el momento en el que sea preciso) y continuo (donde calcula automáticamente una frecuencia de muestreo y mantiene la captura de datos constante, almacenándola en un buffer limitado), por lo que hay un cambio sustancial en la manera de gestionarlo. El primer modo ralentiza mucho la captura de potencia, corriente y T^a de las NTCs mientras que el uso del modo continuo bloquea el programa e imposibilita la captura de datos con el UDA. Para solucionarlo, se decide concatenar medidas usando ventanas pequeñas. De esta manera, mientras se lanza una captura del UDA, el PICO está capturando información y se almacenan los datos nuevos de ambos dispositivos antes de una nueva medida como se explica más adelante.

4.2 Estructura final de la aplicación:

Como hemos comentado, el set experimental sufrió cambios desde su análisis preliminar hasta el diseño del set experimental final. Algunos significaron una modificación importante en el código, principalmente la sustitución del DATALOGGER por la capturadora de datos para termopares PICO TC08 que se utilizó para obtener la base de datos definitiva y que disponía de una gestión de la información recogida muy diferente a la del DATALOGGER.

Como se muestra en la figura 13, la aplicación se compone principalmente de 3 scripts. Un script principal que se ocupa de iniciar el modo de trabajo en paralelo, mandar las tareas a cada uno de los núcleos del microprocesador e iniciar los procesos de grabación y captura de datos, y otros 2 scripts destinados a desarrollar las funciones que nos permiten capturar tanto sonido como los datos y que se lanzan simultáneamente, pero son independientes entre sí. De esta manera la captura de audio no interfiere en la captura de datos pero sí que tienen la misma referencia de tiempo.

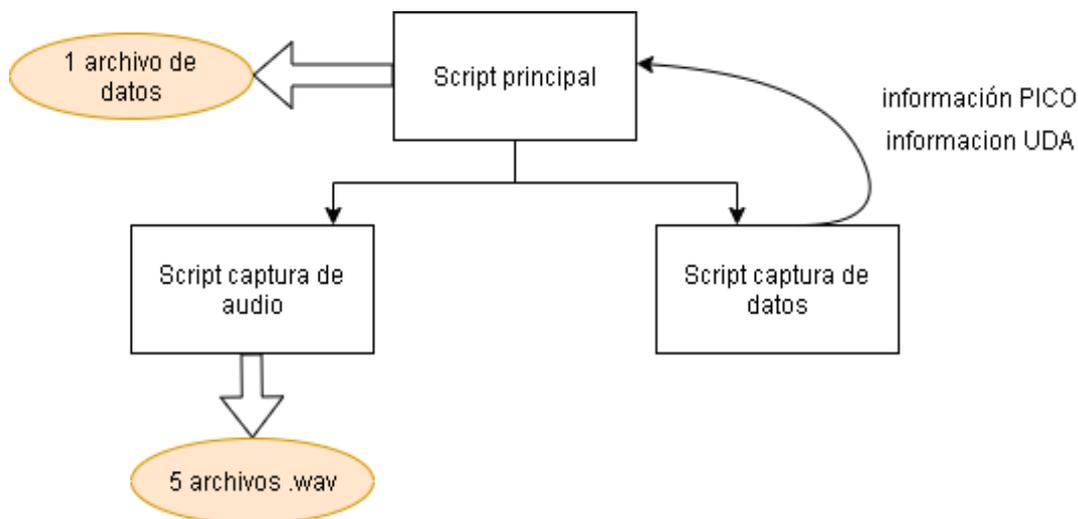


Figura 13. Estructura simplificada del programa

Entrando más en profundidad, pasamos a explicar cada uno de los bloques anteriores:

- **Script principal:** Se encarga principalmente de cargar lo que se llama “Parpool”, configurar e iniciar la siguiente grabación a realizar. Un Parpool es una herramienta que crea un grupo de “trabajadores” que pueden realizar tareas en paralelo. Estos grupos dependen del número de núcleos que tiene el microprocesador del ordenador que usemos. En nuestro caso, al trabajar con un Intel i3, tenemos disponibles 2 trabajadores. En el caso de, por ejemplo un Intel i7, tendríamos 6 núcleos disponibles y podríamos realizar hasta 6 tareas en paralelo.

Una vez indicado el número de trabajadores necesarios, creamos el clúster y configuramos los parámetros de la grabación que vamos a realizar:

- Duración en segundos de la recogida de muestras, tanto de sonido como de los datos.
- Volumen: cantidad de líquido que hemos introducido en el recipiente. Nos sirve principalmente como identificador a la hora de diferenciar las distintas grabaciones conforme vayamos aumentando el dataset.

- Potencia que estamos entregando al recipiente. Es un número del 1 al 9 que indican el nivel de potencia que nos proporciona la cocina. Tiene la misma función que el parámetro anterior ya que la potencia real que proporciona la cocina la medimos en Vatios con el UDA.

Estos parámetros sirven como variables de entrada a las funciones de captura de información.

Una vez acabado el proceso de muestreo, la función de captura de datos devuelve la información que ha obtenido. Estos datos llegan por separado: por un lado los datos capturados por el UDA, que tienen potencia, corriente y la temperatura de las NTCs, y por otro los datos recogidos por el PICO TC08, con los datos de temperatura de la superficie de la placa, de la base del recipiente y del agua. El siguiente paso es unificar estos datos y ordenarlos temporalmente, identificar cuando el recipiente está a 100°C (lo que indica el inicio de la fase de nucleación y la aparición de las primeras burbujas) y cuando el agua alcanza los 100°C (lo que indica que el líquido ha llegado a la temperatura de saturación y las burbujas tienen la suficiente energía para explotar en la superficie). De esta forma, con las características de la grabación (potencia y volumen de esa grabación en concreto) y la referencia temporal tenemos situado cada dato en la fase correspondiente de la ebullición y relacionado con el audio, dándole así las suficientes etiquetas para entrenar la red neuronal a la que van destinados.

Finalmente, con todos los datos ordenados y clasificados, creamos un archivo .csv con toda la información recogida. Este archivo contiene la temperatura recogida por los termopares, la potencia suministrada y la corriente interna de la placa, así como la temperatura de las NTCs situadas en el interior de la placa.

- **Función de grabación de audio:** Es la función que permite recoger las grabaciones del micrófono y de los piezoeléctricos. Recibe como entradas los parámetros de duración, potencia y volumen y genera los archivos .wav de cada uno de los 5 canales de grabación de la tarjeta de sonido que estamos usando: 1 un archivo con la señal recogida por el micrófono y otros 4 cuatro para la señal recogida por cada uno de los piezoeléctricos.

Está estructurado de la siguiente forma: una vez configurado los parámetros de la grabación (frecuencia de muestreo, número de muestras y de canales que estamos usando), crea un objeto DeviceReader, lo que permite grabar cada una de las señales en canales distintos. Mantiene la grabación hasta la duración indicada en la entrada y, al finalizar, escribimos un archivo .wav por cada uno de los canales que hemos utilizado.

- **Función de captura de datos:** Función que nos sirve para integrar todos los dispositivos de captura de datos (UDA y PICOTC08) y lanzar la captura de información a la vez.

Una vez configurado el UDA (iniciando el dispositivo e indicando cual es el área de la cocina y cuáles son los inductores que vamos a utilizar), se configura el PICO TC08 (dándole el número de canales que utiliza y el tipo de termopar, además de cargar las librerías e iniciar el dispositivo). La principal diferencia entre estos dispositivos a la hora de capturar los datos es que mientras el UDA captura un dato en un momento determinado y lo almacena, el PICO TC08 mantiene una grabación constante y los va almacenando en el buffer que hemos creado. Para acceder a los datos es necesario acceder al buffer, donde nos mostrará los últimos datos recogidos que le pidamos. Además, el tiempo de muestreo no es fijo en el UDA y se mueve en un intervalo de una muestra cada 1-1,6 segundos, mientras que el PICOTC08 calcula el tiempo de muestreo automáticamente, que ronda los 400ms.

Una vez configurados ambos dispositivos, la captura de datos se realiza de forma que lanzamos el proceso del PICO TC08 y, mientras está recogiendo datos, pedimos al UDA que recoja una muestra de la potencia, corriente y la temperatura de las NTCs internas. En el tiempo que recoge un dato de cada magnitud, el PICO TC08 recoge unas tres o cuatro muestras (un muestreo del UDA que tarda alrededor de 1 segundo, corresponde con 3 muestras recogidas cada 400ms, pero esto varía ya que el UDA no siempre accede a los datos en un tiempo fijo). Una vez el UDA acaba con la medida, leemos el buffer del PICO TC08i y almacenamos estas tres o cuatro muestras que ha tomado a la vez, teniendo de esta manera la medida de potencia, corriente y temperatura de las

NTC a la vez que distintas medidas de temperatura de la superficie de la placa, del recipiente y del líquido. Almacenamos en registros diferentes las medidas de cada uno de los dispositivos y los pasamos como salida para, como hemos explicado antes, unificarlos, etiquetar las fases y crear el archivo .csv en el script principal. La última misión de esta función es desconectar los dispositivos y liberar recursos para poder realizar nuevas grabaciones.

A modo resumen y como se puede ver en la figura 14, disponemos de un script principal que manda a cada núcleo una de las dos funciones: Capturar audio y Capturar datos. Una vez recogidos los datos, obtenemos 5 archivos con señales acústicas y 1 archivo con diferentes muestras de datos de temperatura (tanto interna como externa de la cocina), potencia y corriente. Se lleva a cabo una primera clasificación de las fases de ebullición y se etiquetan los datos de forma que pueda servir para entrenar la red neuronal a la que van destinados. En el anexo adjunto está disponible tanto el código de la aplicación como los diagramas de flujo de cada una de las funciones principales explicadas.

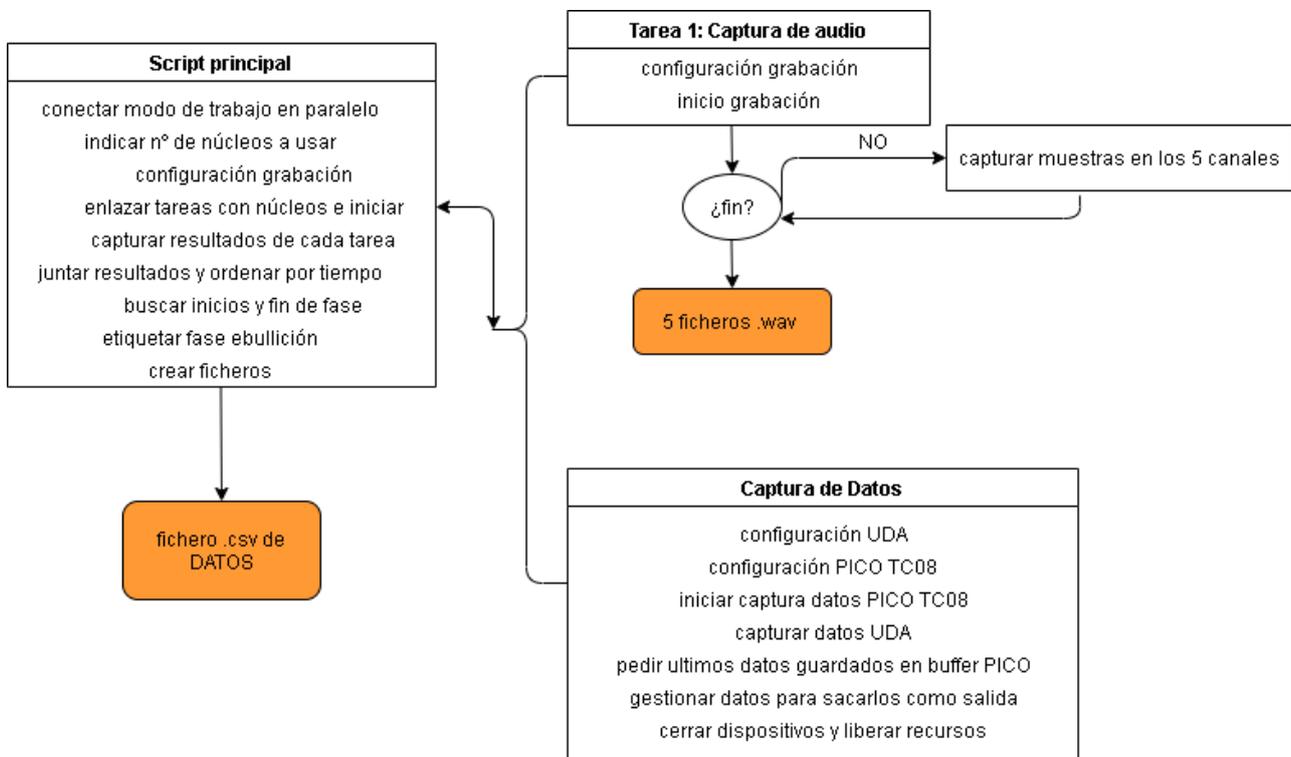


Figura 14. Resumen de la aplicación de medida

Capítulo 5. Análisis global y conclusiones.

5.1 Análisis global

Con la base de datos conseguida, podemos profundizar un poco más en el análisis.

En primer lugar, podemos ver la dependencia de la duración de las fases con la potencia y el volumen de líquido con el que trabajemos. Para verlo, analizamos la velocidad con la que el agua alcanza la temperatura de saturación desde una temperatura de referencia, que tomamos en 60°C, en la figura 15.

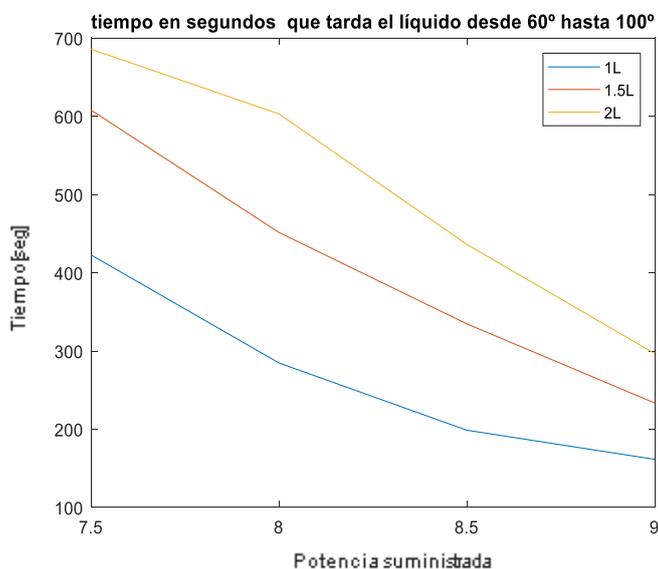


Figura 15. Análisis conjunto de velocidad en alcanzar la Temperatura de saturación.

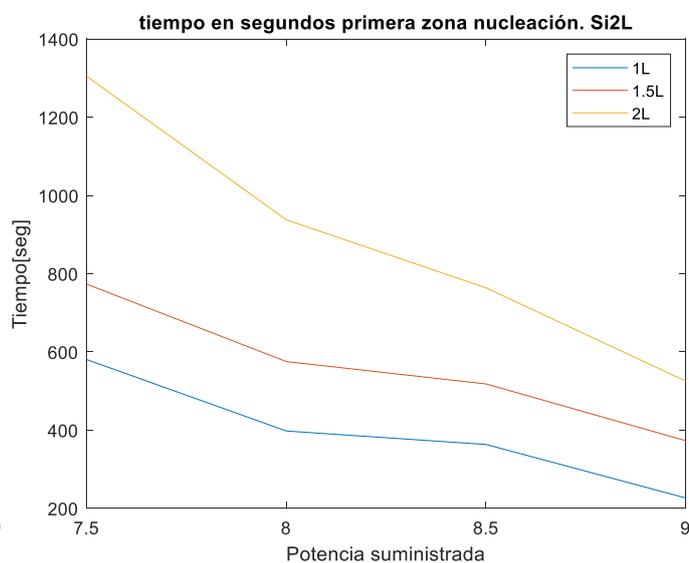


Figura 16. Análisis conjunto de la duración de la primera fase de nucleación

Como es lógico, cuanto mayor potencia apliquemos, más rápido se calienta el recipiente y por tanto, se alcanza la temperatura de saturación más rápidamente. El volumen también afecta: a más volumen de líquido, más tiempo tardará en alcanzarla.

De igual manera, al haber localizado previamente el inicio y final de las fases según la información recogida, podemos estudiar la duración de la burbuja central de la señal de audio, correspondiente a la primera fase de nucleación en función de la potencia y el volumen de trabajo (figura 16). Como era de esperar, la duración aumenta conforme disminuye la potencia aportada.

Además de estos análisis conjuntos, hacemos un estudio frecuencial y analizamos la energía de las señales recogidas.

El análisis frecuencial de la figura 17, donde se refleja el espectrograma de toda la señal, respalda todo lo contado hasta ahora: se puede observar claramente como en la primera fase tenemos frecuencias muy bajas y no es hasta que se alcanza la fase II cuando aparecen frecuencias más altas que se corresponden a ese colapso de burbujas por debajo de la superficie. Conforme llegamos a la fase III, parte de esas frecuencias altas desaparecen.

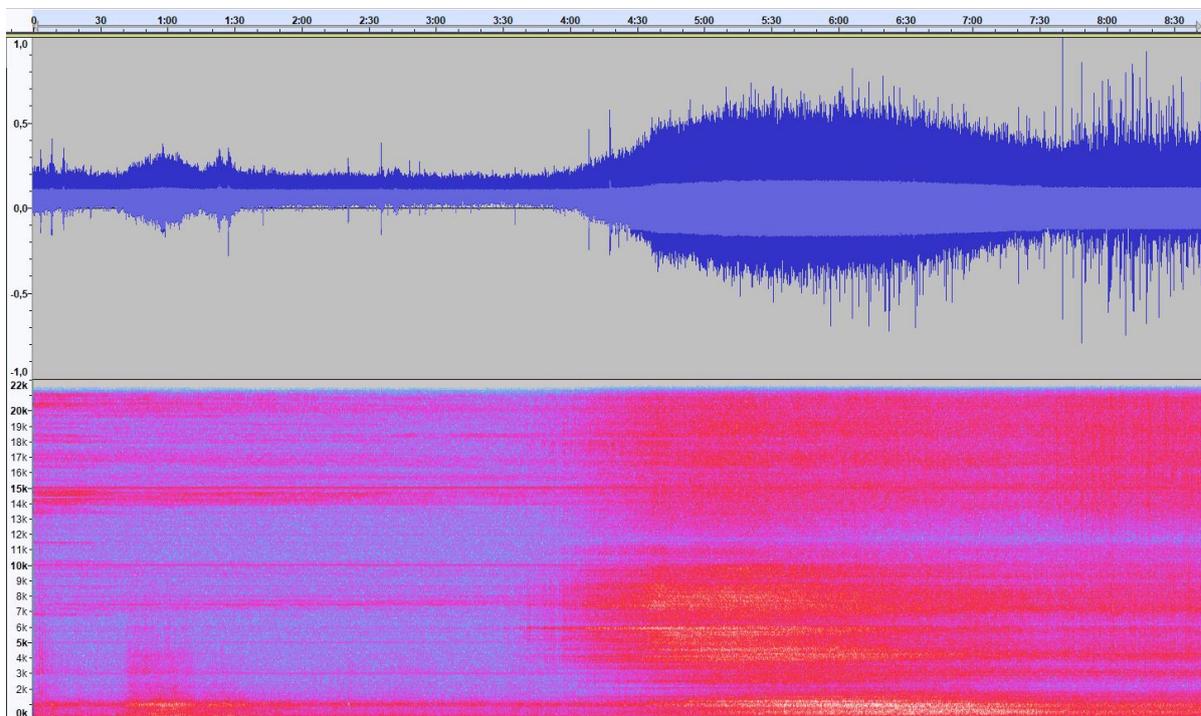


Figura 17. Espectrograma de una de las grabaciones

Analizando más en detalle y comparando el espectro de cada una de las fases (figura 18) vemos que la diferencia reside principalmente en las altas frecuencias. En la segunda fase es mucho mayor el contenido frecuencial entre 500Hz y 7000Hz que en las otras dos, lo que coincide además con que esta sea donde la señal tiene más potencia como se verá después. Además, conforme pasamos a la tercera fase, donde llegan las burbujas a la superficie, vemos como la estructura es más ordenada debido a este burbujeo, a diferencia de las otras dos, donde o el sonido es muy tenue, o se produce el colapso de las burbujas por debajo de la superficie.

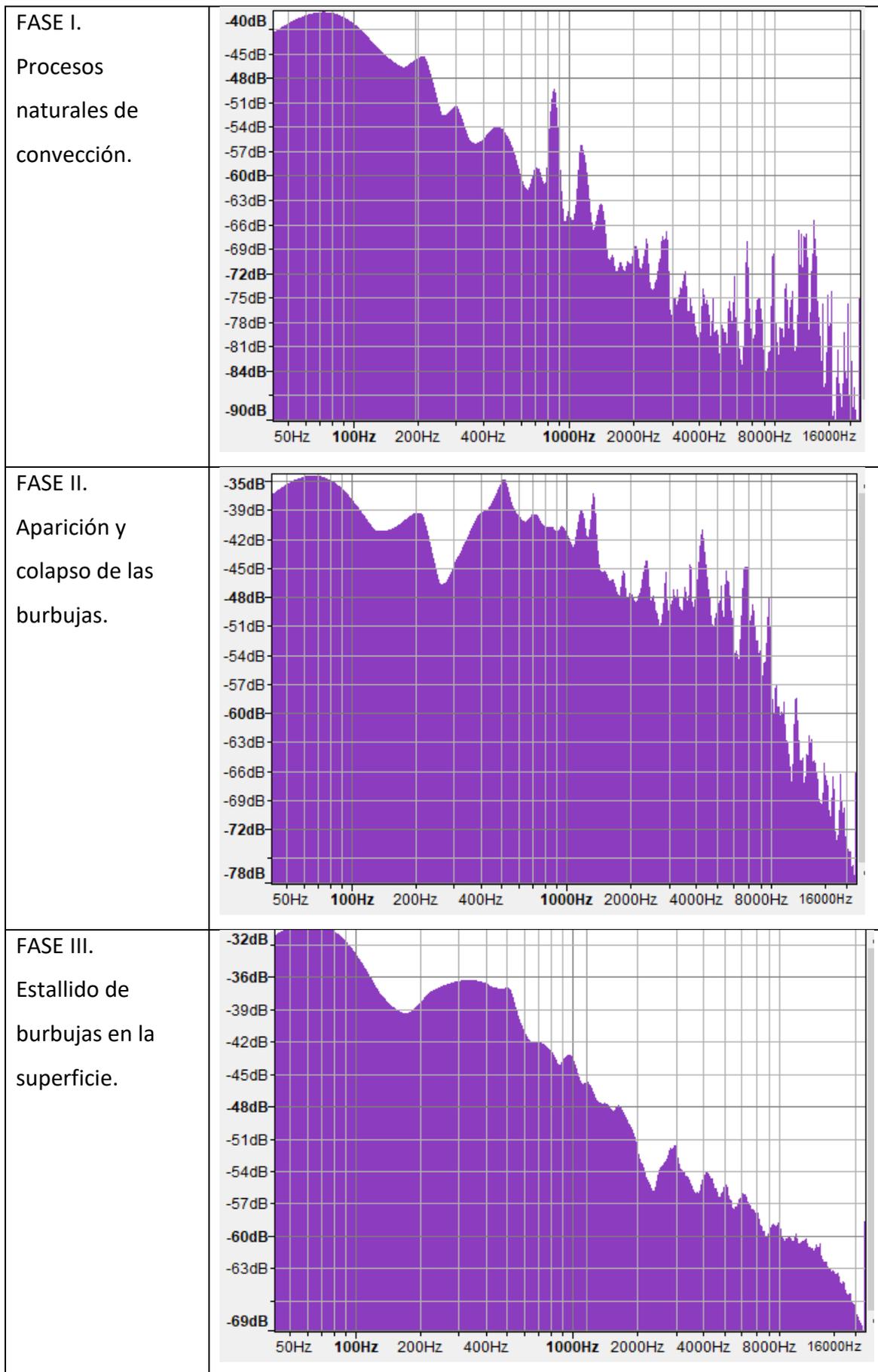


Figura 18. Análisis del espectro de cada una de las fases

Para acabar con este análisis, calculamos la energía de los audios recogidos por el micrófono calculando el valor RMS de la señal de audio. En la figura 19 se observa el valor RMS de cada fase y el valor medio de este y se observa cómo se mantiene prácticamente constante entorno a unos valores medios de 0.01 en la primera fase, 0.05 en la segunda y 0.04 en la tercera, lo que cuadra con el análisis espectral y con lo que también somos capaces de diferenciar cada una de las fases.

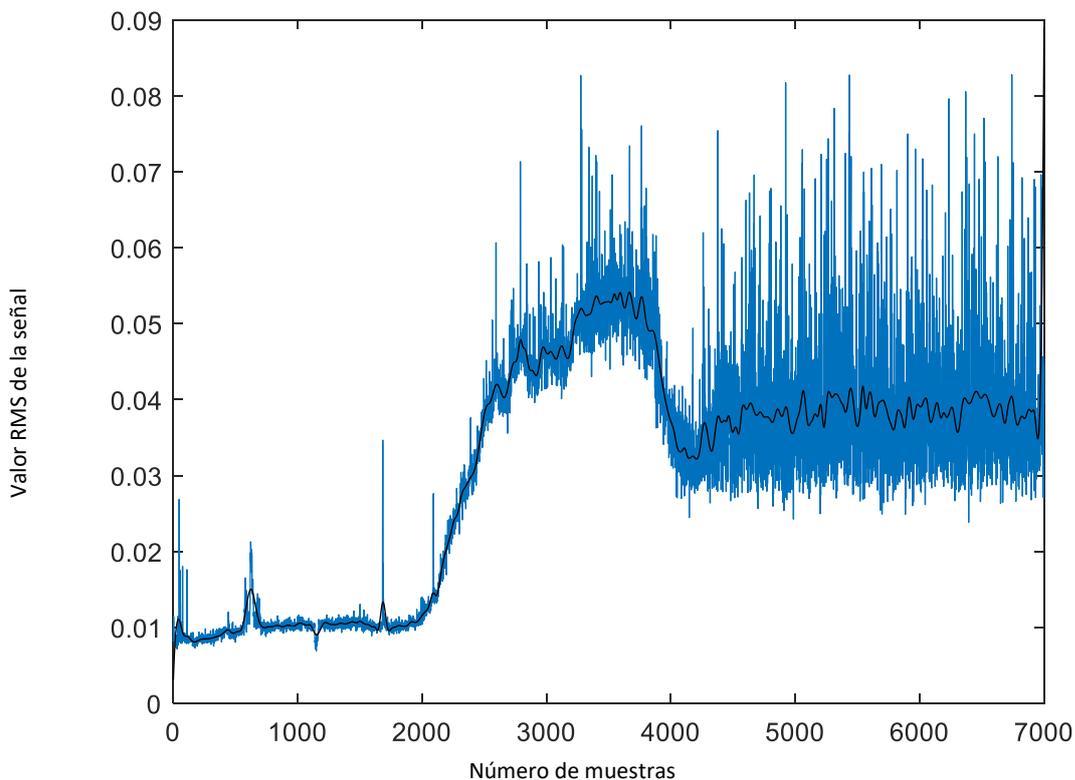


Figura 19 Análisis del valor RMS en toda la grabación.

Para finalizar, vamos a estudiar si el sensor de presión alojado en el interior de la placa que situamos al cambiar la estructura del set experimental nos permite obtener la misma información que el micrófono. En la figura 20 vemos que la señal recogida por el piezoeléctrico alojado en el interior (1) recibe una señal semejante a la de los piezoeléctricos situados en la superficie de esta (2), sin sufrir distorsiones ni interferencias. Con todo lo anterior, concluimos que podemos obtener la misma información que nos proporciona el micrófono con sensores de presión, aunque

debería hacerse un análisis más profundo, ya que, como se refleja en la figura 21, los análisis espectrales de las señales recogidas por el piezoelectrico son distintos a los referentes a la señal recogida por el micrófono, pero podríamos llegar a diferenciar las fases con ellos. Hay que resaltar que la potencia de la señal recogida por los piezoelectricos es mucho menor y el problema principal al usar estos sensores es que existe un pico a 50Hz que domina la señal, haciéndose necesario un filtrado para poder trabajar con esa información.

De cara a una futura implementación para la mejora de las prestaciones de la cocina, obtener la información con estos sensores en lugar de con micrófonos hace el diseño mucho más barato.

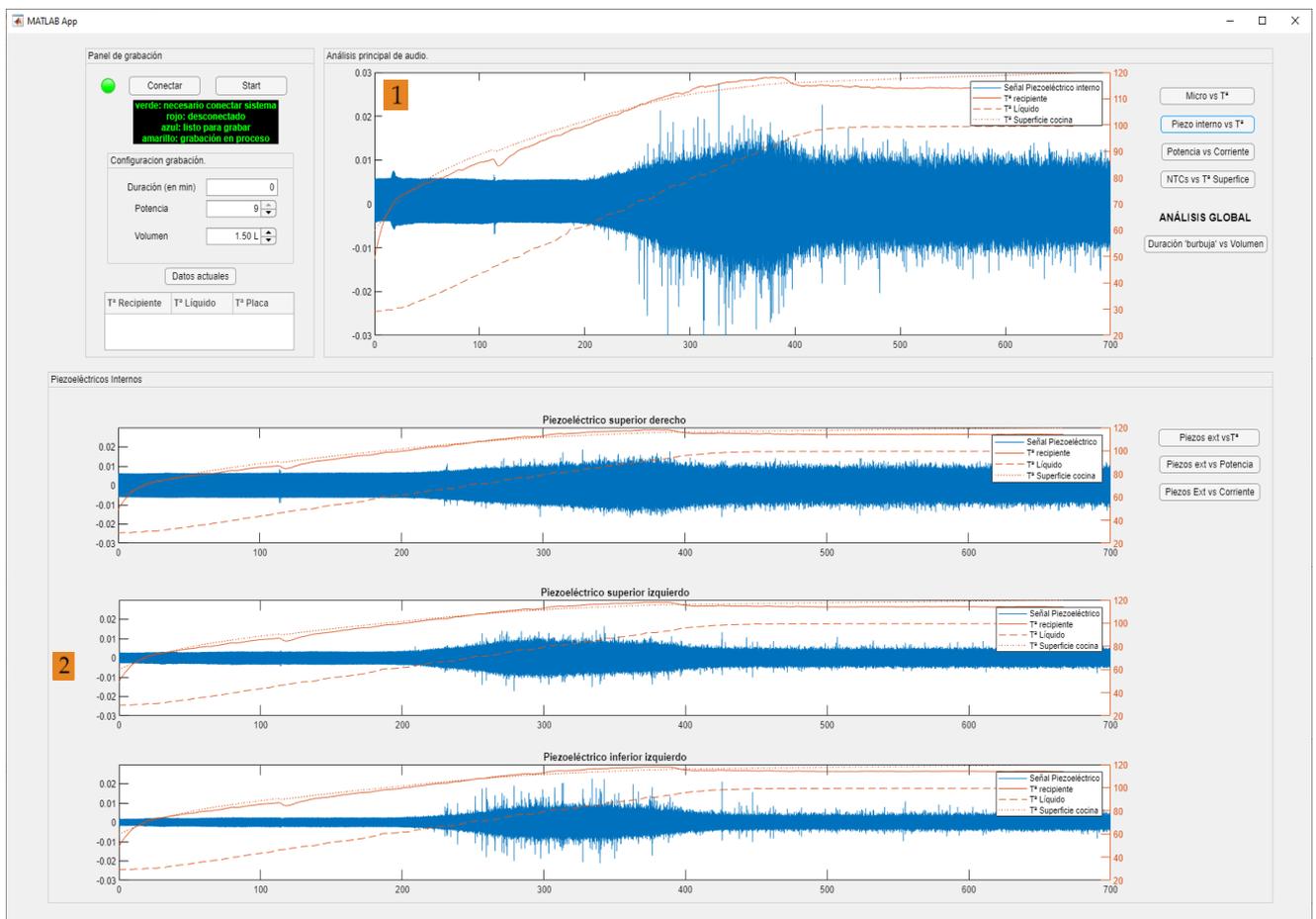


Figura 20. Comparativa señal del piezoelectrico interno y los piezoelectricos en la superficie

La imagen anterior se han sacado de una interfaz de usuario desarrollada para el proceso de grabación y análisis que nos facilita el análisis temporal. Esta interfaz nos permite comparar distintas grabaciones rápidamente, así como realizar nuevas

grabaciones y obtener la temperatura de la superficie del recipiente, de la placa y del líquido en cualquier momento fuera del proceso de grabación.

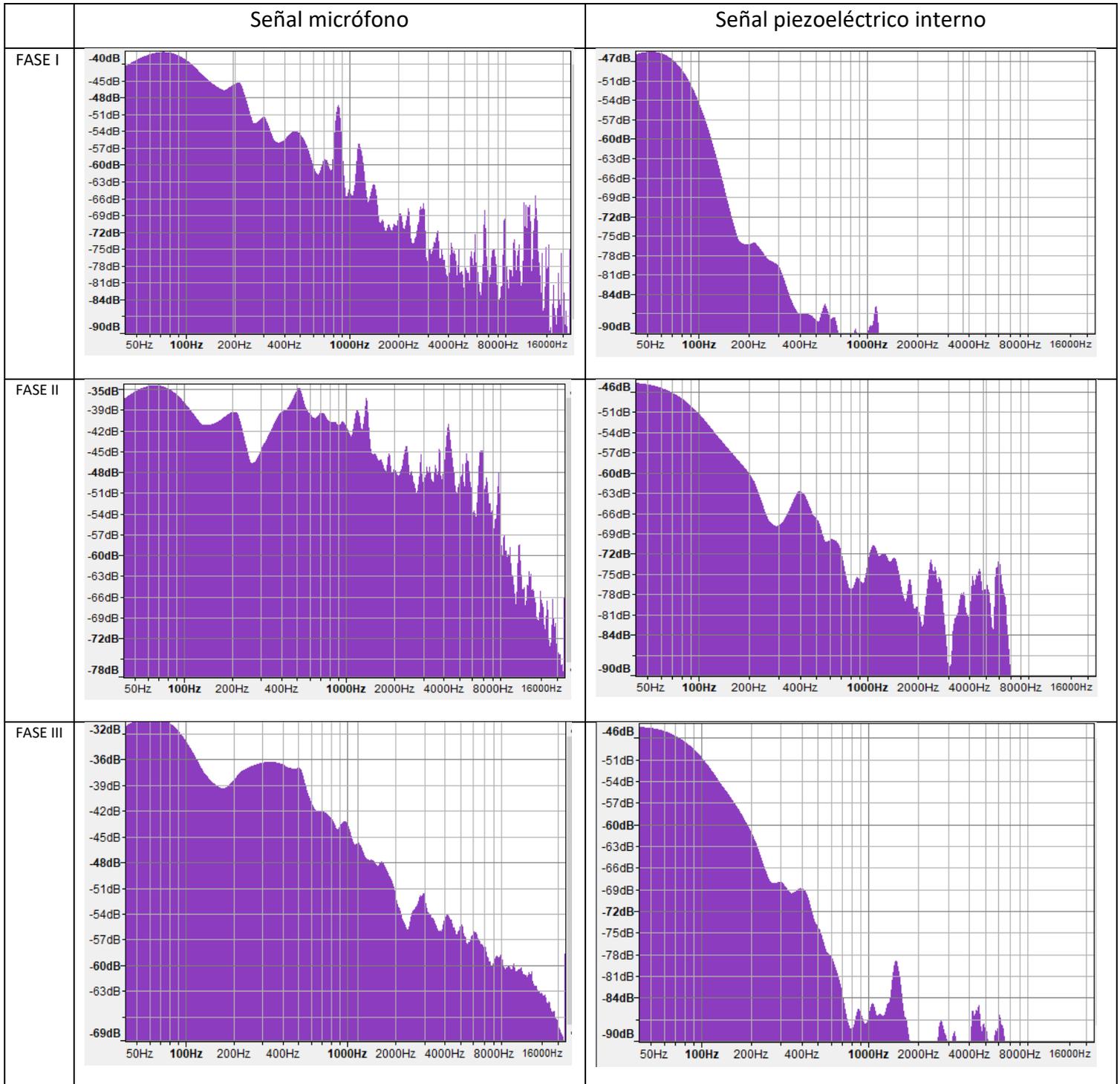


Figura 21. Comparativa análisis espectral de las señales recogidas por el micrófono y el piezoeléctrico interno.

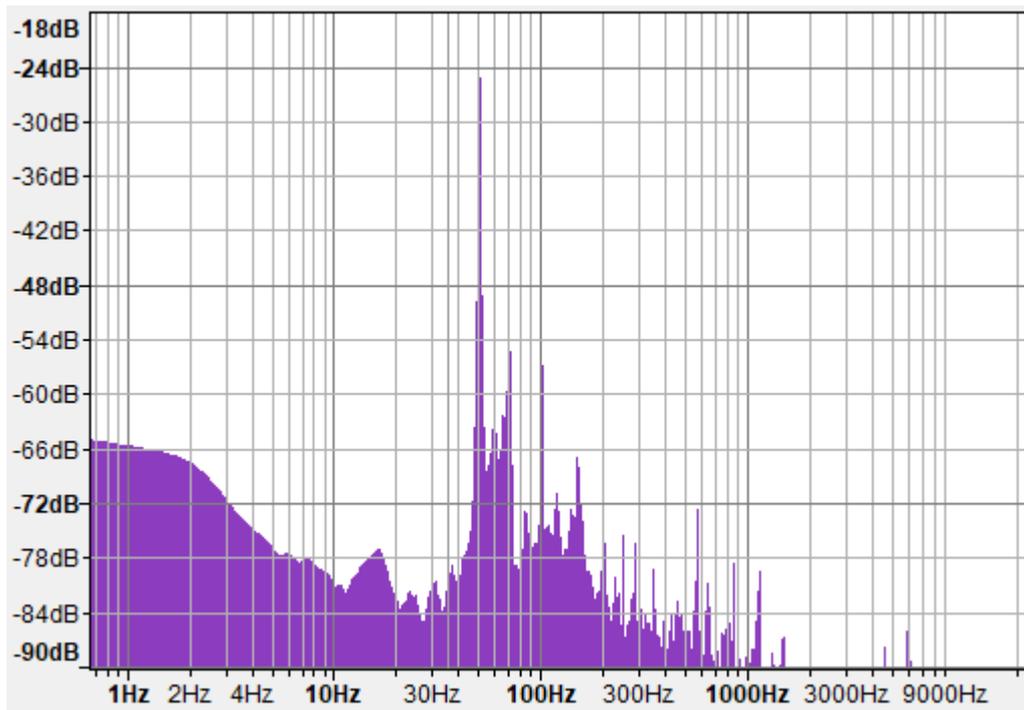


Figura 22. Detalle del pico en 50Hz en la señal recogida por el piezoeléctrico interno.

5.2 Resumen y conclusiones:

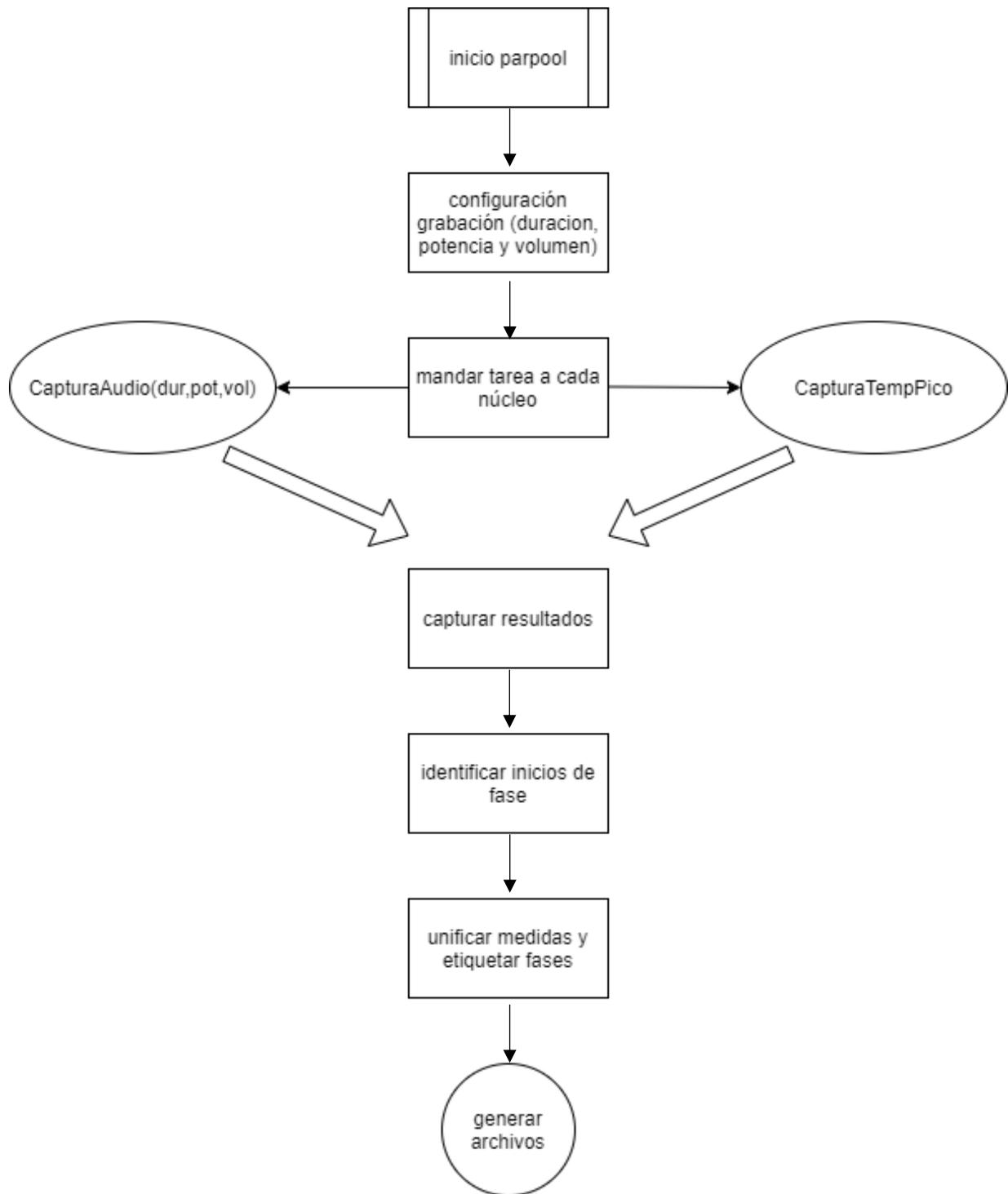
Para concluir y a modo de resumen. Se ha desarrollado un sistema para obtener una base de datos destinada a entrenar una red que tiene como objetivo, al menos, clasificar las distintas fases de la ebullición de un líquido. Pensando que este estudio se realiza para mejorar las prestaciones de una cocina de inducción por medio de técnicas de machine learning, creemos que con los datos recogidos la red será capaz de aprender a diferenciar el inicio y el final de las fases, y en último lugar, poder estimar la temperatura del agua.

En cuanto a lo específico de este trabajo, hemos visto que una vez superados los obstáculos relativos a la programación e integración de los dispositivos en el sistema de medida, hemos diseñado el set experimental y obtenido una base de datos con toda la información necesaria para poder diferenciar las fases de la ebullición. Además, concluimos que la misma información obtenida por medio de las grabaciones de un micrófono se pueden obtener con sensores de presión como piezoeléctricos, mucho más baratos y fáciles de integrar en el interior de una cocina.

Bibliografía.

- [1] Carmi, Rami, Arie Bussiba, Galit Widenfeld, Yosi Aharon, Igal Alon, y Igal Hochbaum. «DETECTION OF TRANSIENT ZONES DURING WATER BOILING BY ACOUSTIC EMISSION», s. f., 9.
- [2] Sinha, Kumar Nishant Ranjan, Vijay Kumar, Nirbhay Kumar, Atul Thakur, y Rishi Raj. «Deep Learning the Sound of Boiling for Advance Prediction of Boiling Crisis». *Cell Reports Physical Science* 2, n.º 3 (marzo de 2021): 100382. <https://doi.org/10.1016/j.xcrp.2021.100382>.
- [3] Tabacchi, M., C. Asensio, I. Pavón, M. Recuero, J. Mir, y M.C. Artal. «A Statistical Pattern Recognition Approach for the Classification of Cooking Stages. The Boiling Water Case». *Applied Acoustics* 74, n.º 8 (agosto de 2013): 1022-32. <https://doi.org/10.1016/j.apacoust.2013.03.001>.
- [4] B-5 Microphone, «Quick start guide». [en línea]. Accedido Septiembre de 2021. Disponible en: https://mediadl.musictribe.com/media/PLM/data/docs/P0224/B-5_M_WW.pdf
- [5] MOTU 8pre, «User's Guide for Windows». [en línea]. Accedido Septiembre de 2021. Disponible en: https://cdn-data.motu.com/manuals/firewire-usb-audio/8pre_User_Guide_Win.pdf
- [6] AGILENT DATALOGGER, «Data Acquisition / Switch Unit Service Guide». [en línea]. Accedido Septiembre de 2021. Disponible en: <https://www.dennlec.com/images/hp-34970a-service-manual.pdf>
- [7] PICO TC-08, «Cloud-connected 8-channel thermocouple data logger». [en línea]. Accedido Septiembre de 2021. Disponible en: <https://www.picotech.com/download/datasheets/usb-tc-08-thermocouple-data-logger-data-sheet.pdf>
- [8] Gaggenau, «Datasheet Placa Flex de Inducción (CV282100)». [En línea]. Accedido Septiembre de 2021. Disponible en: <https://media3.gaggenau.com/Documents/specsheet/es-ES/CV282100.pdf>.
- [9] Ebullición de transición - <<What is Transition Boiling - Partial Film Boiling>> [En línea]. Accedido Septiembre de 2021. Disponible en: <https://www.thermal-engineering.org/what-is-transition-boiling-partial-film-boiling-definition/>
- [10] Ebullición de película <<What is Film Boiling - Definition >> Accedido Septiembre de 2021. Disponible en: <https://www.thermal-engineering.org/what-is-film-boiling-definition/>

Anexo A. Código y diagramas de flujo.
Script principal.



```

p = parpool(2);
durREC=1300      %duración de la grabación.
pot=8           %potencia a la que se va a realizar la captura de datos.
vol=1           %volumen (en Litros) del líquido que contiene el recipiente.
for n=1:2       %%ordenar tarea a cada núcleo del microcontrolador.
    if(n==1)
        f(n) = parfeval(p, @CapturaAudio,2, durREC, vol, pot);
        {f.State}
    else
        f(n) = parfeval (p, @CapturaTempPico,2, durREC)

        {f.State}

    end
    disp(p.FevalQueue.RunningFutures)
end
[fin, sennal] = fetchOutputs(f(1))
[datosUDA,datosTemp] = fetchOutputs(f(2))

%% ordenamos los datos recogidos, unificamos las medidas del UDA y del PICO TC08.
%% hayamos las transiciones de fase y etiquetamos las fases
pucheroa100 = datosTemp(:,2)>=100;
q= find(pucheroa100==1);
q(1);

aguaa100 = datosTemp(:,3) >= 100;
qq= find(aguaa100==1);
qq(1);

clasif=[];
ceros=[zeros(q(1),1)];

cuenta= qq(1)- q(1);
unos=[ones(cuenta,1)];
resto=size(datosTemp,1)-qq(1);
doses=[ones(resto,1)+1];
clasif=[ceros;unos;doses];
datosTemp(:,5)=clasif

A=[];
B=[];
C=[];
D=[];
TrainingData=[];
for u=1:1:size(datosUDA,1)
    B=[B; datosUDA(u,1) 0 0 0 datosUDA(u,2:7) 9999];
end
for k=1:1:size(datosTemp,1)
    A=[A;datosTemp(k,1) datosTemp(k,2:4) 0 0 0 0 0 datosTemp(k,5)];
end
C=[A;B];

```

```
D=sortrows(C);
```

```
for j=1:1:size(D,1);  
    if(D(j,11)==9999);  
        if(D(j-1,11)==0);  
            D(j,11)=0;  
        end  
        if(D(j-1,11)==1);  
            D(j,11)=1;  
        end  
        if(D(j-1,11)==2);  
            D(j,11)=2;  
        end  
    end  
end  
end
```

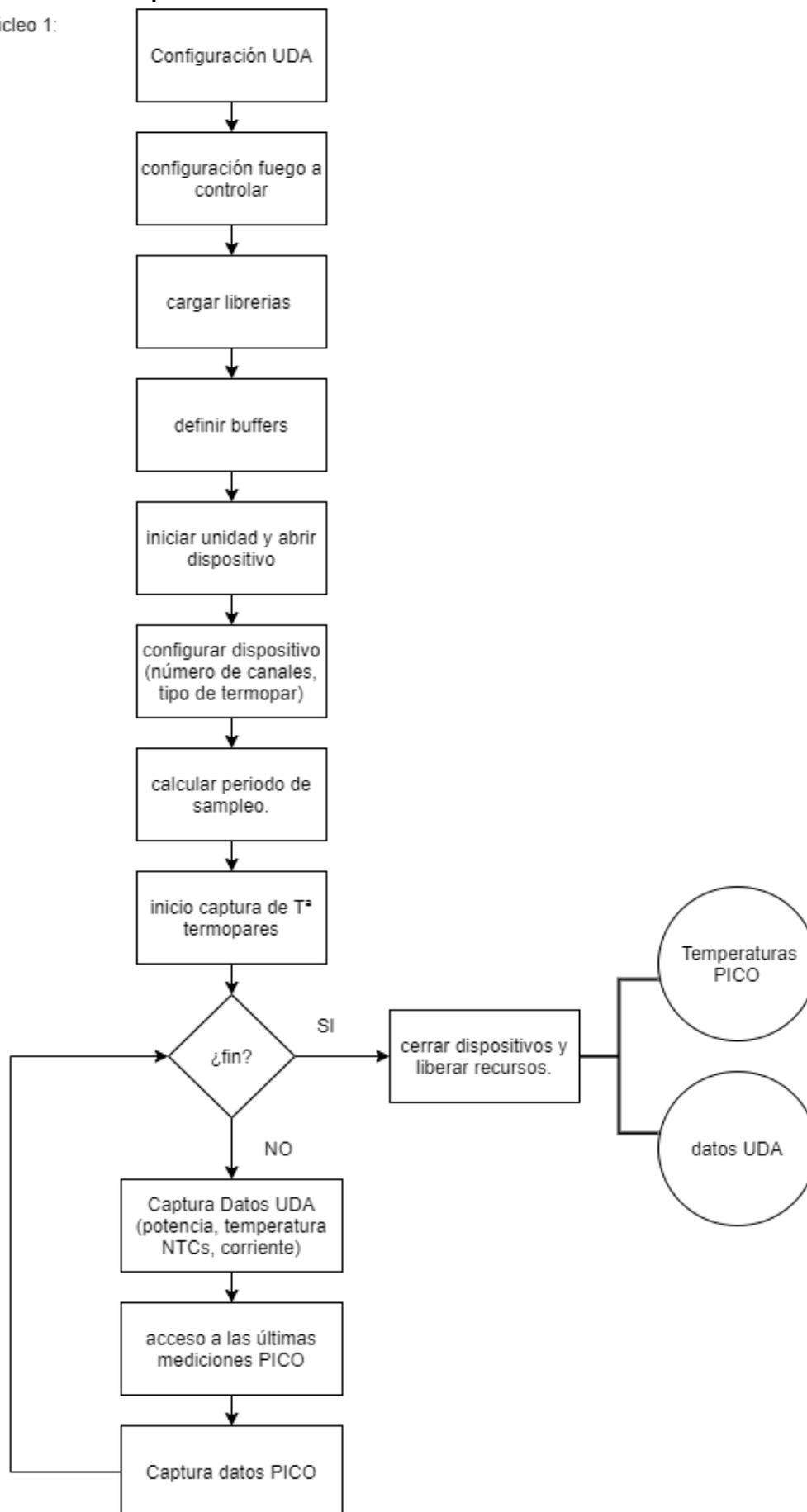
```
%% creamos 3 archivos .csv con los datos. El destinado a entrenar la red será  
%% ARCHIVORED .
```

```
potencia= num2str(pot)  
volumen= num2str(vol)  
TempFile=strcat(volumen,'L_P',potencia,'_MedidasTemperatura.csv');  
Datos=strcat(volumen,'L_P',potencia,'_MedidasUDA.csv');  
Train=strcat(volumen,'L_P',potencia,'_ArchivoRED.csv');  
csvwrite(Datos, datosUDA)  
csvwrite(TempFile, datosTemp)  
csvwrite(Train, D)
```

```
%% análisis. Gráficar y analizar los datos recogidos.  
 analisisIndividual(pot, vol, datosUDA, datosTemp)
```

Función de captura de datos.

Núcleo 1:



```

function [DatosSalida, Temperaturas] = CapturaTempPico(dur)
try
%% Clear Command Window and Figures
clc;
close all;
% clear all;
warning off;
finDatos=0;

%% UDA CONFIG
uda1 = UDA(19200,1000,'C0');
pause(0.1);

%Time configuration
Time.Period = 1;
tic;
%n_iter=numiter

%Configuracion fuego a controlar
PL0.nodel = '70';
PL0.type = 'uint16';

%% Load Configuration Information
USBTC08Config;

%% Load Library
if (ismac())

    loadlibrary('libusbtc08.dylib', @usbtc08MFile, 'alias', 'usbtc08');

    % Check if the library is loaded
    if ~libisloaded('usbtc08')

        error('Library libusbtc08.dylib or usbtc08MFile not found');

    end

elseif (isunix())

    loadlibrary('libusbtc08.so', @usbtc08MFile, 'alias', 'usbtc08');
    % Check if the library is loaded
    if ~libisloaded('usbtc08')

        error('Library libusbtc08.so or usbtc08MFile not found');

    end

elseif (ispc())
    loadlibrary('usbtc08.dll', @usbtc08MFile);
    % Check if the library is loaded

```

```

    if ~libisloaded('usbtc08')
        error('Library usbtc08.dll or usbtc08MFile not found');
    end
else
error('USBTC08Example:OSNotSupported', 'Operating system not supported, please
contact support@picotech.com');
end

%% Define Buffers

numSamples = 3;

pBufferCJ    = libpointer('singlePtr',zeros(numSamples, 1, 'single'));
pBufferCh1   = libpointer('singlePtr',zeros(numSamples, 1, 'single'));
pBufferCh2   = libpointer('singlePtr',zeros(numSamples, 1, 'single'));
pBufferCh3   = libpointer('singlePtr',zeros(numSamples, 1, 'single'));
pBufferCh4   = libpointer('singlePtr',zeros(numSamples, 1, 'single'));
pBufferTimes = libpointer('int32Ptr',zeros(numSamples, 1, 'int32'));
overflow     = libpointer('int16Ptr',zeros(1, 1, 'int16'));

TemperaturaCJ=[];
TemperaturaCh1=[];
TemperaturaCh2=[];
TemperaturaCh3=[];
TiempoCJ=[];
TiempoCh1=[];
TiempoCh2=[];
TiempoCh3=[];

%variables UDA
Tiempo=[];
potencia=[];
corriente=[];
DatosNTCs=[];
NTC1=[];
NTC2=[];
NTC3=[];
NTC4=[];
DatosSalida=[];

%% Open Unit and Display Information
unithandle = calllib('usbtc08', 'usb_tc08_open_unit');
if unithandle > 0
    disp('Connected to USB TC-08 device...');
    fprintf('\n');
elseif unithandle == 0

```

```

        warning('USBTC08Example:NoMoreUnitsFound', 'No more USB TC-08 units found');
    else
        unloadlibrary('usbtc08');
        error('USBTC08Example:UnitFailedToOpen', 'Unit failed to open.');
```

return;

end

% Get unit information

```

infoString = blanks(512);
[status.unitinfo, infostring] = calllib('usbtc08', 'usb_tc08_get_formatted_info', ...
    unithandle, infoString, length(infoString));

disp(infostring);
error = calllib('usbtc08', 'usb_tc08_get_last_error', unithandle);

%% Configure Device
% Set up channels

usbtc08MaxChannels =
usbtc08Enuminfo.enUSBTC08Channels.USBTC08_MAX_CHANNELS;
typek = 'K';

% Enable Cold Junction and Channel 1 and 2
for n = 1:4

    status.setChannel = calllib('usbtc08', 'usb_tc08_set_channel', unithandle, ...
        (n - 1), int8(typek));
end

% Set mains filter to 50Hz
status.mainsFilter = calllib('usbtc08', 'usb_tc08_set_mains', unithandle, 0);

% Find minimum sampling interval
min_interval_ms = calllib('usbtc08', 'usb_tc08_get_minimum_interval_ms', unithandle)

%% Capture Data
n_iter=dur;
fin=0;
TiemposTemp=[];
interval = calllib('usbtc08', 'usb_tc08_run', unithandle, ... min_interval_ms)
Times=[];
TempCh1=[];
TempCh2=[];
TempCh3=[];
TrainingData=[];
```

```

for iter=1:n_iter
    iter;
    tiempoact=toc;
    Tiempo=[Tiempo; tiempoact];

if toc>=dur
    fin=1
end
if fin==0

% %medida de potencia. UDA
[res1, Power11] = uda1.Read(PL0.nodel, '361e', 'uint16');
potencia=[potencia; Power11];

% %medida de corriente. UDA
[res1, Current11] = uda1.Read(PL0.nodel, '3622', 'uint16');
corriente= [corriente; Current11/100];
%
% %medida de Tª NTC. UDA
[res3, NTCglassread1I] = uda1.Read(PL0.nodel, '0aee', 'uint16');
[res4, NTCglassread2I] = uda1.Read(PL0.nodel, '0ace', 'uint16');
[res5, NTCglassread3I] = uda1.Read(PL0.nodel, '0b6e', 'uint16');
[res6, NTCglassread4I] = uda1.Read(PL0.nodel, '0b8e', 'uint16');
%
NTC1=[NTC1; NTCglassread1I];
NTC2=[NTC2; NTCglassread2I];
NTC3=[NTC3; NTCglassread3I];
NTC4=[NTC4; NTCglassread4I];

DatosNTCs= [NTC1 NTC2 NTC3 NTC4];

DatosSalida=[Tiempo potencia corriente DatosNTCs]
%

%% accedemos a las últimas medidas del PICOTC08.
[numValuesCJ, pBufferCJ, pBufferTimes, overflow] = calllib('usbtc08', ...
    'usb_tc08_get_temp', unithandle, pBufferCJ, pBufferTimes, numSamples, ...
    overflow, 0, 0, 0);

[numValuesCh1, pBufferCh1, pBufferTimes, overflow] = calllib('usbtc08', ...
    'usb_tc08_get_temp', unithandle, pBufferCh1, pBufferTimes, numSamples, ...
    overflow, 1, 0, 0);

[numValuesCh2, pBufferCh2, pBufferTimes, overflow] = calllib('usbtc08', ...
    'usb_tc08_get_temp', unithandle, pBufferCh2, pBufferTimes, numSamples, ...
    overflow, 2, 0, 0);

```

```
[numValuesCh3, pBufferCh3, pBufferTimes, overflow] = calllib('usbtc08', ...  
    'usb_tc08_get_temp', unithandle, pBufferCh3, pBufferTimes, numSamples, ...  
    overflow, 3, 0, 0);
```

```
Times=single([Times; pBufferTimes])  
TempCh1=[TempCh1; pBufferCh1]  
TempCh2=[TempCh2; pBufferCh2];  
TempCh3=[TempCh3; pBufferCh3];  
Temperaturas=[Times/1000 TempCh1 TempCh2 TempCh3]  
end  
end
```

```
%% Stop the Device  
stop = calllib('usbtc08', 'usb_tc08_stop', unithandle);  
error = calllib('usbtc08', 'usb_tc08_get_last_error', unithandle);  
uda1.ClosePort();
```

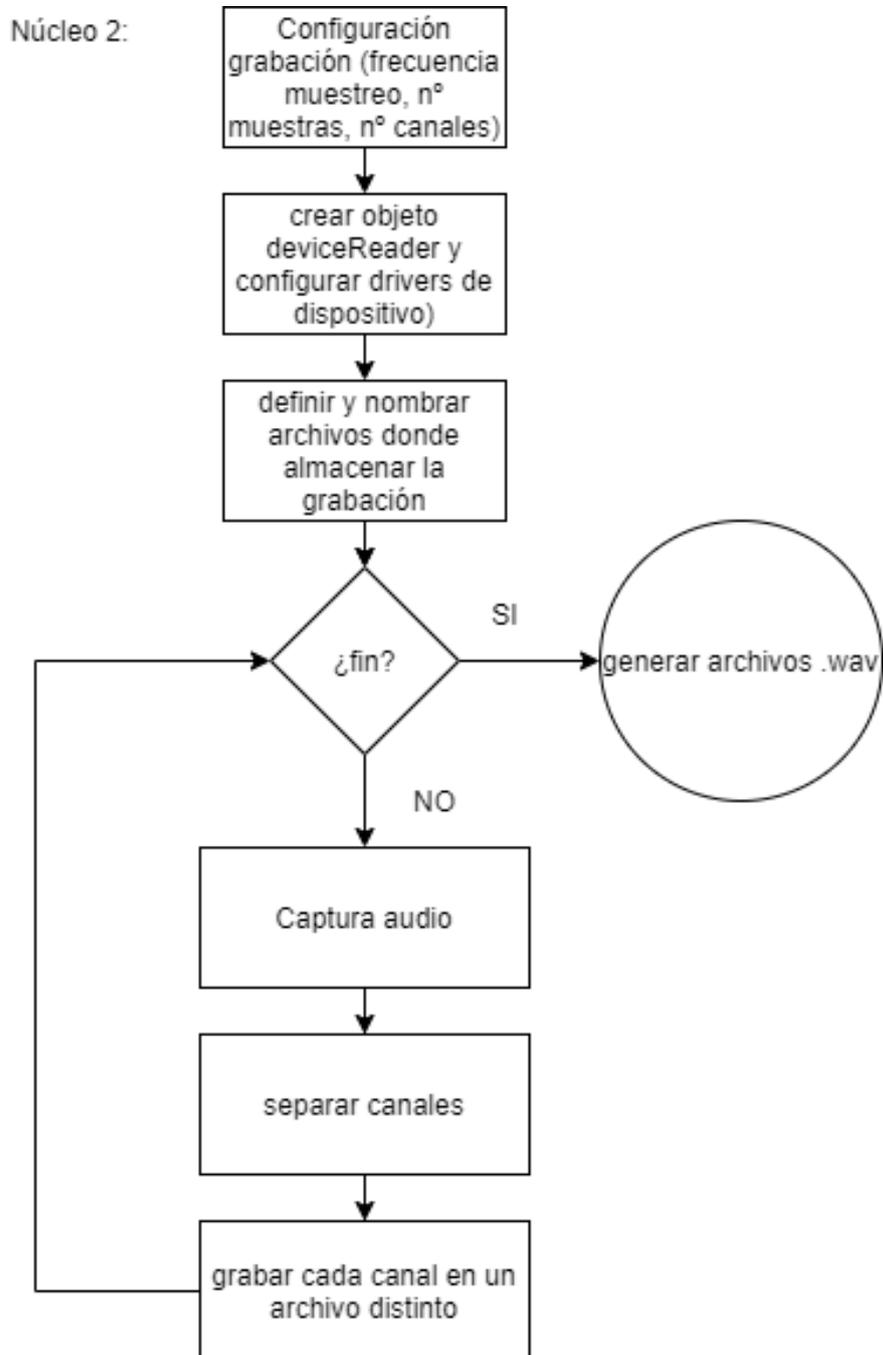
```
%% Close Connection to Unit and Unload Library  
exit = calllib('usbtc08', 'usb_tc08_close_unit', unithandle);  
unloadlibrary('usbtc08');  
finDatos=1;
```

```
catch ME
```

```
stop = calllib('usbtc08', 'usb_tc08_stop', unithandle);  
error = calllib('usbtc08', 'usb_tc08_get_last_error', unithandle);  
uda1.ClosePort();
```

```
%% Close Connection to Unit and Unload Library  
exit = calllib('usbtc08', 'usb_tc08_close_unit', unithandle);  
unloadlibrary('usbtc08');  
finDatos=1;  
end  
end
```

Función de captura de audio.



```

function [fin, fileWriter2, fileWriter1] = CapturaAudio(dur, vol, pot)

Fs=44100;
Samples=256;
nChannels=8;

potencia=num2str(pot)
volumen=num2str(vol)
deviceReader = audioDeviceReader(Fs, Samples, 'Driver' , 'ASIO', 'Device', 'MOTU Audio ASIO' ,
'NumChannels' , nChannels);
setup(deviceReader);

fileWriter1 = dsp.AudioFileWriter(strcat(volumen,'L_P',potencia,'_Micrófono.wav'));
fileWriter2 =
dsp.AudioFileWriter(strcat(volumen,'L_P',potencia,'_Piezoelectrico1_dcha_arriba.wav'));
fileWriter3 =
dsp.AudioFileWriter(strcat(volumen,'L_P',potencia,'Piezo1eléctrico2_izda_abajo.wav'));
fileWriter4 =
dsp.AudioFileWriter(strcat(volumen,'L_P',potencia,'Piezo1eléctrico3_izda_arriba.wav'));
fileWriter5 =
dsp.AudioFileWriter(strcat(volumen,'L_P',potencia,'Piezo1eléctrico4_interno.wav'));

disp('Speak into microphone now.')
tic
fin=0;
while toc < dur
    acquiredAudio = deviceReader();
    fileWriter1(acquiredAudio(:,1));
    fileWriter2(acquiredAudio(:,5));
    fileWriter3(acquiredAudio(:,6));
    fileWriter4(acquiredAudio(:,7));
    fileWriter5(acquiredAudio(:,8));
end
disp('Recording complete.')
fin=1;
release(deviceReader)
release(fileWriter1)
release(fileWriter2)
release(fileWriter3)
release(fileWriter4)
release(fileWriter5)
end

```