Universidad de Alcalá
Escuela Politécnica Superior

Grado en Ingeniería Electrónica de Comunicaciones

Trabajo Fin de Grado

Diseño, implementación y evaluación de un demostrador de captura y procesamiento de audio multicanal en espacios inteligentes

Autor: Sandra Caso Alba

Tutor: Javier Macías Guarasa

2018

UNIVERSIDAD DE ALCALÁ

ESCUELA POLITÉCNICA SUPERIOR

Grado en Ingeniería Electrónica de Comunicaciones

Trabajo Fin de Grado

Diseño, implementación y evaluación de un demostrador de captura y procesamiento de audio multicanal en espacios inteligentes

Autora: Sandra Caso Alba

Tutor: Javier Macías Guarasa

Tribunal:

Presidente: Marta Marrón Romera

Vocal 1º: Juan Carlos García García

Vocal 2º: Javier Macías Guarasa

Calificación: .....

Fecha: .....

A mis padres, por no dejar que me rindiera...

"Un poco más de persistencia, un poco más de esfuerzo, y lo que parecía irremediablemente un fracaso puede convertirse en un éxito glorioso."
Elbert Hubbard

Agradecimientos

Antes de nada, me gustaría agradecer a mis padres todo su esfuerzo y por hacerme ver que abandonar no era una opción, ya que gracias a ellos hoy he llegado hasta aquí y lo he conseguido. Gracias por aguantarme en mis peores días, y también por estar ahí en los mejores. Sobre todo, a mi madre, por alegrarse tanto o más que yo cuando lo he logrado.

También me gustaría agradecer a mis abuelos, que me han apoyado siempre sin dudarlo, y a toda mi familia.

Gracias también a todos los amigos que me llevo, por compartir sufrimientos de días enteros en la universidad, pero también por hacerlos más llevaderos.

Y, por último, gracias a Christian, por apoyarme siempre y no dudar de mi en ningún momento. Gracias por ayudarme a conseguirlo.

Resumen

Este proyecto describe el diseño, desarrollo, implementación y evaluación de un demostrador en tiempo real de captura y procesamiento de audio multicanal para localización de hablantes mediante agrupaciones de micrófonos dentro de espacios inteligentes.

Para conseguir este objetivo se han combinado los resultados de trabajos previos de otros Proyectos Fin de Grado y Fin de Máster en los que se desarrollaron librerías de adquisición y reproducción de audio multicanal, algorítmica de detección de actividad de voz en entornos con agrupaciones de micrófonos, módulos de visualización de entornos virtuales y algorítmica de localización de locutores, integrando su funcionalidad y adaptándola a las exigencias del funcionamiento en tiempo real.

Asimismo, se ha completado el hardware disponible en la sala *ispace*, que consta de dos agrupaciones de micrófonos, implementando las conexiones oportunas para añadir dos nuevos arrays al sistema, estudiando la geometría de estos. Asimismo, se ha diseñado e implementado un quinto array que consta de una nueva topología con el fin de estudiar sus ventajas.

Palabras clave: Localización acústica, detección de actividad de voz, demostrador en tiempo real, procesamiento basado en agrupaciones de micrófonos, espacio inteligente.

Abstract

This project describes the design, development, implementation and evaluation of a real time demonstrator of multi-channel audio capture and processing for localization of speakers using microphone arrays in smart spaces.

To achieve this goal it has been combined the results of previous Thesis which developed multichannel acquisition and reproduction libraries, activity voice detection algorithms in environments with microphone clusters, virtual environment visualization modules and speaker location algorithms, integrating their functionalities and adapting them to real time operating requirements.

Likewise, the hardware available in the room *ispace*, which consists of two groups of microphones, has been completed, implementing the appropriate connections to add two new arrays to the system, studying the geometry of these. In addition, a fifth array consisting of a new topology has been designed and implemented in order to study its advantages.

Keywords: Acoustic localization, voice activity detection, real time demonstration, microphone array processing, smart space.

Resumen extendido

Este proyecto describe el diseño, desarrollo, implementación y evaluación de un demostrador en tiempo real de captura y procesamiento de audio multicanal para localización de hablantes mediante agrupaciones de micrófonos dentro de espacios inteligentes, y tiene como objetivo depurar y corregir las implementaciones previas del sistema.

Para ello se ha partido del resultado de proyectos anteriores, como "Diseño, implementación y evaluación de un demostrador de localización de fuentes de audio en tiempo real" de María Paz González [1], implementación de librerías de audio multicanal como "Diseño, implementación de un sistema de adquisición, procesamiento y generación de audio multicanal en aplicaciones para espacios inteligentes" realizado por Jesús Pablo Martínez González [2], algorítmica de detección de actividad de voz en entornos con agrupaciones de micrófonos del proyecto "Estudio, implementación y evaluación de un sistema de detección de actividad de voz en espacios inteligentes" desarrollado por Rubén Peral Nuño [3], módulos de visualización de entornos virtuales como "Diseño, implementación y evaluación de una interfaz de control multimodal en un espacio inteligente: control gestual" llevado a cabo por David Casillas [4], y algorítmica de localización de locutores del proyecto "Speaker Localization Techniques in Reverberant Acoustic Environments" realizado por Carlos Castro [5].

El proyecto se divide en tres grandes bloques, los cuales van a ser el estudio teórico, el desarrollo y evaluación y resultados. En un primer lugar, se realizará el estudio teórico del funcionamiento del sistema, el cual se ha dividido en una parte hardware y una parte software, con el fin de comprenderlo mejor. Esta parte hardware incluirá los siguientes apartados:

- Micrófonos: en este primer apartado se procederá a describir los micrófonos que se van a utilizar en este proyecto. Para ello, se repasará el tipo de micrófonos existentes, los tipos de patrones de directividad que se pueden tener, y por qué se han elegido micrófonos de condensador con patrón de directividad omnidireccional. Además, también se explicarán los diferentes modelos de micrófonos implicados en este proyecto, describiendo la estructura, el tipo de micrófono, en qué se basa su funcionamiento y las características fundamentales de los mismos.

- Agrupaciones de micrófonos: este apartado tiene como objetivo profundizar en la estructura y distribución de los micrófonos que se encuentran en los arrays implementados. Como se verá posteriormente, se tendrán un total de cinco arrays de micrófonos. Dos de ellos ya se encontraban implementados con anterioridad, y en este proyecto se han implementado los tres arrays restantes, explicándose en la parte de desarrollo los pasos seguidos para ello. Cuatro de los arrays tendrán la misma estructura, estando compuestos por cuatro micrófonos cada uno, formando una estructura en T. Por su parte, el quinto estará compuesto por ocho micrófonos, formando una nueva estructura heptagonal, con el fin de poder observar las ventajas y desventajas que tiene esta estructura frente a la mencionada anteriormente.

- Previo y convertidor A/D: se describirá el funcionamiento de los previos utilizados en este proyecto, siendo un total de tres, los cuales tienen como tarea preamplificar y filtrar la señal procedente de los diversos micrófonos, además de digitalizarla con su Analog to Digital Converter (ADC) interno.

- Tarjeta de adquisición multicanal: en esta parte se explicará el funcionamiento de la tarjeta de adquisición de audio multicanal utilizada en el presente proyecto, la cual se encargará de muestrear la señal digitalizada devuelta por lo diferentes previos con una frecuencia de muestreo indicada.

Continuando con la parte de software, ésta se encuentra dividida en cinco grandes bloques, los cuales se resumen a continuación:

- Bloque de adquisición de audio multicanal: este bloque hará posible la captura de audio de los diferentes micrófonos colocados en los arrays que se encuentran en el espacio inteligente, además del procesamiento de las diferentes señales adquiridas y la generación de audio multicanal en tiempo real.

- Bloque de cálculos de correlación: en él se realizarán cálculos de correlación Cross-power Spectrum Phase (CSP), los cuales estarán basados en diferencias temporales (algoritmo Time Difference Of Arrival (TDOA)) entre las señales adquiridas para cada uno de los pares de micrófonos de los arrays disponibles. Estos datos se almacenan para que posteriormente se puedan utilizar en el bloque de detección de actividad de voz y en los cálculos para la estimación de la posición de la fuente sonora.

- Bloque de detección de actividad de voz: se encargará de decidir qué tramas contienen o no voz. Para ello, se dispone de un bloque encargado del cálculo del umbral de detección, y un bloque que será un autómata o máquina de estados, cuya tarea será decidir qué tramas contienen voz y cuáles son solo periodos de silencio, para poder así descartar aquellas que no interesen. Esto último es muy importante, ya que de esta forma no se malgastan recursos del procesador realizando cálculos sobre señales que no contienen voz.

- Bloque de estimación de la fuente sonora: su función es localizar al hablante dentro del espacio inteligente. Para conseguir esto, se va a utilizar el algoritmo de localización basado en respuesta dirigida (Steered Response Power (SRP)), el cual evalúa la actividad acústica en localizaciones específicas, orientando el patrón de directividad del array utilizando la técnica de $beamforming$. Asimismo, este bloque hará uso de los cálculos de correlación obtenidos en el módulo anteriormente descrito, utilizándolos para la técnica de localización basada en la dirección de llegada (Direction of Arrival (DOA)), haciendo uso de las diferencias de tiempos de llegada de la voz a los distintos micrófonos que componen un array para completar el cálculo de la ubicación de la fuente sonora.

- Bloque de visualización 3 Dimensiones (3D): en este módulo se hará una representación virtual del espacio inteligente donde se han realizado las pruebas en 3 Dimensiones (3D). Este permite la visualización tanto del entorno como de ciertos objetos que permiten una interacción hombre-máquina a modo de actuadores del espacio inteligente. Gracias a esta sencilla e intuitiva representación se puede tener una vista funcional del espacio inteligente que posibilita hacer ajustes de vistas y proyecciones, además de observar cambios en los actuadores del entorno virtual o representar la posición del locutor dentro de dicho entorno.

Posterior a esta parte de estudio teórico, la cual tiene la finalidad de comprender las partes y funcionamiento del sistema completo implementado, se abordarán en el apartado de desarrollo todas las implementaciones llevadas a cabo en este proyecto. Se tendrán los siguientes apartados:

- Implementación de arrays de micrófonos: en este apartado se explicarán las diferentes conexiones necesarias en los micrófonos a la hora de implementarlos en los nuevos arrays. También, se describe el proceso de diseño del nuevo array E implementado, así como la forma de solucionar los problemas encontrados.

- Configuración y conexionado de los elementos hardware: el objetivo de esta parte es realizar una descripción de todas las conexiones y configuraciones a nivel de hardware necesarias para el correcto funcionamiento del sistema, tanto de los micrófonos, como de los previos y de la tarjeta de adquisición multicanal, describiendo, por ejemplo, posiciones en las que se deben encontrar los switches, significado de los indicadores LED, etc. También se describirán los programas utilizados para realizar las configuraciones oportunas a nivel de software.

- Aplicación de detección de pasos por cero: se describirá el funcionamiento de este programa, el cual se necesitará posteriormente en la parte de evaluación de resultados. Tiene como finalidad encontrar todos los pasos por cero que se encuentran en las señales contenidas en ficheros de formato *.wav*, devolviendo el número de muestra donde se han producido estos pasos por cero en unos ficheros *.txt*.

Por último, se tendrá la parte de evaluación y resultados, donde se realizarán diversas pruebas con la finalidad de comprobar y demostrar el correcto funcionamiento del sistema. Contiene los siguientes apartados:

- Verificación de los micrófonos: en esta parte se tiene como objetivo comprobar que los diferentes micrófonos implementados en la sección de desarrollo efectivamente tienen un funcionamiento correcto, para lo cual se realizarán grabaciones con los mismos y se estudiarán los resultados obtenidos.

- Sincronismo del sistema: en este apartado se explicará cómo se va a comprobar la correcta sincronización entre micrófonos, para lo cual se realizarán una serie de grabaciones de tonos de distintas frecuencias, midiendo los diferentes retardos existentes entre cada par de micrófonos.

- Funcionamiento demostrador en tiempo real con los nuevos micrófonos: se explicará cómo se comprueba que el programa funcione correctamente con los nuevos micrófonos añadidos, así como los problemas encontrados en este proceso y cómo se han solucionado los mismos.

- Evaluación perceptual del demostrador: por último, en este apartado se describirán las pruebas realizadas para conseguir tener una visión global de la eficiencia del funcionamiento del sistema con cada uno de los arrays por separado, y posteriormente, con todos juntos.

Tras el análisis de todo el demostrador en tiempo completo, se llega a encontrar y solventar un problema en el manejo de la memoria que no permitía hacer uso de más de doce canales. Con esto y todo el despliegue de hardware realizado, se consigue una mejora del sistema, aunque quedan como líneas futuras el análisis de la comparativa entre los resultados obtenidos en tiempo real y los que se tienen con un funcionamiento offline.

Índice general

Índice de figuras

Índice de tablas

Lista de acrónimos

3D	3 Dimensiones.
ADC	Analog to Digital Converter.
API	Application Programming Interface.
CC	Cross Correlation.
CSP	Cross-power Spectrum Phase.
DOA	Direction of Arrival.
GCC	Generalized Cross Correlation.
GEINTRA	Grupo de Ingeniería Electrónica aplicada a Espacios Inteligentes y Transporte.
ML	Maximum Likelihood.
PHAT	Phase Amplitude Transform.
SRP	Steered Response Power.
TDOA	Time Difference Of Arrival.
VAD	Voice Activity Detector.

Capítulo 1

Introducción

Los espacios inteligentes son entornos que utilizan sensores (como cámaras, agrupaciones de micrófonos, etc.) y algorítmica del procesamiento con la finalidad de conseguir crear ambientes interactivos con el usuario. El objetivo de estos espacios es lograr resolver problemas cotidianos y mejorar así las actividades comunes, y para conseguirlo los diferentes sensores cooperan entre sí, analizando la información recogida del usuario. En este contexto, las tareas de detección, localización y seguimiento de personas son fundamentales para mejorar dichos procesos de interacción con el entorno, o con otras personas u objetos dentro del mismo.

Este proyecto se centra en la evaluación y mejora de un demostrador en tiempo real en el contexto de los sistemas de localización y seguimiento de hablantes en un espacio inteligente, uno de los trabajos orientados a la generación de tecnología basada en el procesamiento de las señales captadas por agrupaciones de micrófonos que propone el *Grupo de Ingeniería Electrónica aplicada a Espacios Inteligentes y Transporte (GEINTRA)* del Departamento de Electrónica siguiendo la línea de investigación y desarrollo enfocada a explotar los diferentes sensores que se encuentran en el espacio inteligente del cual dispone la Universidad de Alcalá.

En este Trabajo Fin de Grado se parte de trabajos previos de otros Proyectos Fin de Grado y Tesis de Máster, como "Diseño, implementación y evaluación de un demostrador de localización de fuentes de audio en tiempo real" de María Paz González [1], implementación de librerías de adquisición y reproducción de audio multicanal como "Diseño, implementación de un sistema de adquisición, procesamiento y generación de audio multicanal en aplicaciones para espacios inteligentes" realizado por Jesús Pablo Martínez González [2], algorítmica de detección de actividad de voz en entornos con agrupaciones de micrófonos del proyecto "Estudio, implementación y evaluación de un sistema de detección de actividad de voz en espacios inteligentes" desarrollado por Rubén Peral Nuño [3], módulos de visualización de entornos virtuales como "Diseño, implementación y evaluación de una interfaz de control multimodal en un espacio inteligente: control gestual" llevado a cabo por David Casillas [4], y algorítmica de localización de locutores del proyecto "Speaker Localization Techniques in Reverberant Acoustic Environments" realizado por Carlos Castro [5].

1.1 Motivación y objetivos

El objetivo fundamental de este proyecto es el diseño, implementación y evaluación de un demostrador de captura y procesamiento de audio multicanal en espacios inteligentes. Los objetivos específicos de este proyecto son los siguientes:

- Diseñar y realizar el despliegue de hardware de captura multicanal en el espacio de demostración del Grupo de Investigación, parcialmente equipado.
- Evaluar y mejorar una librería de adquisición de audio multicanal como soporte al procesamiento en tiempo real de señales captadas por agrupaciones de múltiples micrófonos.
- Realizar estudios de calibración y corrección de los flujos de audio generados por el hardware desplegado en los dos escenarios descritos.
- Diseñar, implementar y evaluar la arquitectura de los demostradores a generar, integrando progresivamente los desarrollos de los siguientes subobjetivos:
  - Integración y evaluación de un sistema de detección de actividad de voz multicanal.
  - Integración y evaluación un sistema de localización de locutores.
  - Integración y evaluación de un sistema de visualización de los resultados del demostrador.

Los requisitos que debe cumplir el trabajo a desarrollar son:

- Ser flexible en el sentido de permitir modificar con facilidad los parámetros de control de los procesos de adquisición, detección de actividad, localización acústica y visualización.
- Ser flexible en el sentido de permitir la cómoda incorporación y control de nuevos dispositivos de captura y reproducción de audio, así como la incorporación de nueva algorítmica en el mismo entorno de demostración.
- Estar bien documentado para facilitar su utilización en futuros proyectos.
- Disponer de un software eficiente y robusto.

1.2 Organización de la memoria

Esta memoria se ha estructurado en cinco capítulos y un anexo, que se describen a continuación:

1. Introducción: en el capítulo 1 se realiza una presentación general del trabajo realizado, incluyendo el punto de partida. También se incluyen los objetivos iniciales y las motivaciones que han impulsado la realización de este trabajo.
2. Diseño del sistema: en el capítulo 2 se van a describir todos los conceptos teóricos de cada una de las partes que componen el sistema completo, tanto de la parte hardware como de la parte software.
3. Desarrollo: en el capítulo 3 se explica paso a paso todo el desarrollo seguido para realizar este proyecto, explicándose la implementación de toda la parte hardware (micrófonos y arrays).
4. Resultados y evaluación: en el capítulo 4 se evalúa el funcionamiento del sistema y se analizan los resultados obtenidos para demostrar su correcta ejecución.
5. Conclusiones y líneas futuras: el capítulo 5 muestra las conclusiones a las que se ha podido llegar y se describen las mejoras que se deben realizar sobre el sistema.
6. Apéndices:
   (a) Herramientas y recursos: en el apéndice A se enumeran las herramientas utilizadas para la elaboración del proyecto.

(b) Especificaciones: en el apéndice B se presentan a modo de resumen todas las especificaciones de los micrófonos, previos, tarjeta de adquisición multicanal, cableado y software.

Capítulo 2

Diseño del Sistema

2.1 Introducción

En esta sección previa al desarrollo del proyecto, se realizará un estudio teórico para llegar a conocer la estructura general del sistema, y se explicará brevemente la funcionalidad de cada uno de los bloques que van a formar parte del demostrador en tiempo real completo, tanto de la parte hardware como de la parte software. La arquitectura de dicho demostrador se puede observar en la figura 2.1.

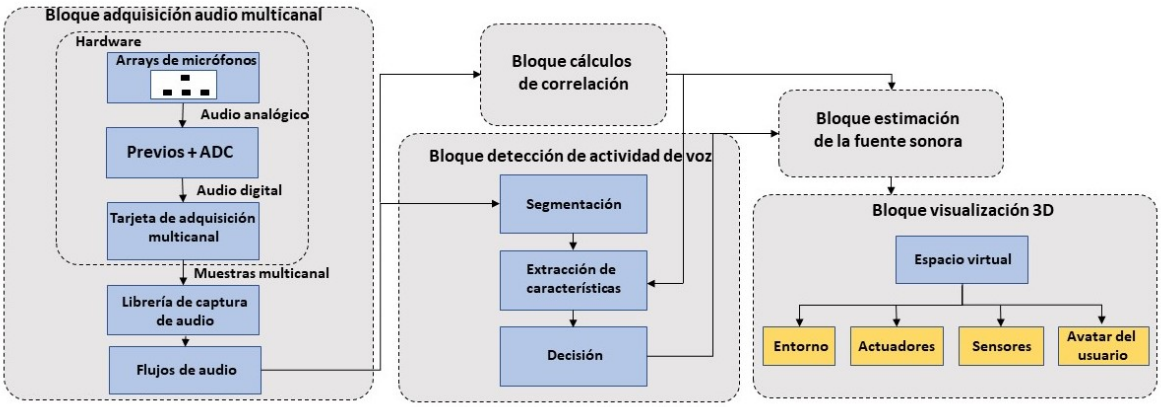


Figura 2.1: Arquitectura del demostrador de localización en tiempo real.

El sistema, en un primer lugar, realizará la adquisición de audio multicanal a través del hardware, el cual está compuesto en primer lugar por los micrófonos que se encuentran situados en los diferentes arrays, teniendo cada array en total cuatro micrófonos, a excepción de uno de ellos, que cuenta con ocho. La señal capturada por estos micrófonos llegará hasta el previo, el cual se encargará de preamplificarla y filtrarla, para posteriormente digitalizarla con el ADC que posee internamente. Por último, la señal digitalizada pasará a la tarjeta de adquisición de audio multicanal, donde será muestreada a una frecuencia de muestreo previamente fijada.

Esta información se transmitirá hasta el ordenador, donde la aplicación se encargará de procesarlo para obtener los resultados buscados. En un primer lugar, llegará hasta los bloques de detección de actividad de voz y cálculos de correlación, los cuales se encargarán de decidir si las tramas de audio contienen voz o no. Posteriormente, las tramas con audio pasarán al bloque de estimación de la posición de la fuente sonora, donde se obtendrá la localización del locutor. Una vez conocida, los resultados obtenidos llegarán

hasta el bloque de visualización 3D, donde se representará esta localización estimada en un entorno que representa la sala ***ispace***. Todas estas tareas se deben realizar en tiempo real, por lo que deben cumplirse unos compromisos de sincronismo entre procesos y tiempo de ejecución de estos.

Para entender mejor el funcionamiento del sistema, en los siguientes apartados se va a separar entre la parte hardware, compuesta por los micrófonos, las agrupaciones de micrófonos, los previos (que cada uno contiene un ADC) y la tarjeta de adquisición, y la parte software, compuesta por la aplicación que contiene cinco bloques: bloque de adquisición de audio multicanal, bloque de cálculos de correlación, bloque detección de actividad de voz (*Voice Activity Detector (VAD)*), bloque de estimación de la posición de la fuente sonora y bloque de visualización 3D.

2.2 Hardware

2.2.1 Micrófonos

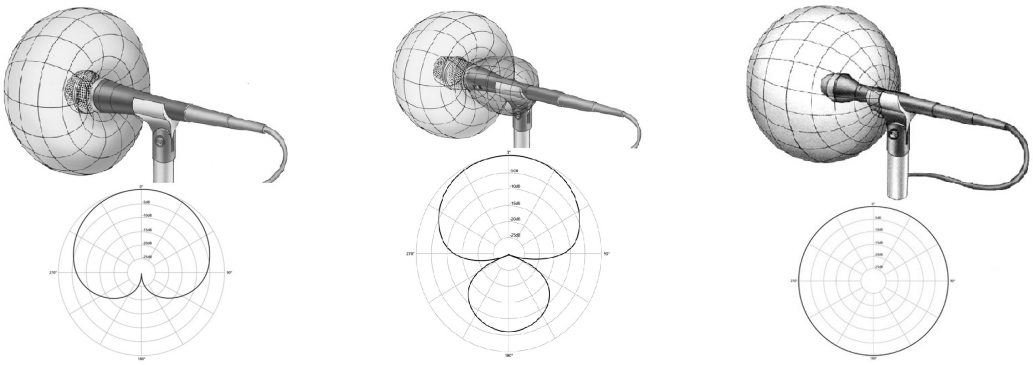
Un micrófono funciona como un transductor o sensor electroacústico, el cual es el mecanismo encargado de convertir las señales sonoras en energía eléctrica. Existen varios tipos de micrófonos disponibles, como pueden ser los electrostáticos (o de condensador), dinámicos, de cinta, de carbono, piezoeléctricos, de fibra óptica, láser, líquido, micro-electromagnético, etc.

Además, cada micrófono también se puede clasificar según sus patrones de directividad. Estos definen la forma en la que el micrófono va a responder a los sonidos que vienen desde distintas direcciones, la cual se va a definir tanto con el ángulo de captación del sonido que tiene el micrófono como con el ángulo de máximo rechazo, por lo que se dará una idea de cómo se debe situar el micrófono para mejorar la captación de una fuente sonora deseada y evitar las que no interesen. Los patrones de directividad más comunes son:

- Cardioide: su ángulo de captación será de aproximadamente 130º. Este patrón es más sensible a sonidos que llegan de frente, y menos a los que llegan desde la parte trasera. Se puede observar en la figura 3.5a.

- Supercardioide: su ángulo de captación será de aproximadamente 115º, que es un ángulo más ajustado que el cardioide, por lo que permite un mayor rechazo, aunque es sensible a sonidos que procedan de la parte trasera. Ofrece mejor aislamiento al ruido ambiente y es más resistente a acoples. Se puede ver en la figura 3.5b.

- Omnidireccional: su ángulo de captación será de 360º, independientemente de su orientación. Su patrón polar es una esfera perfecta, tal y como se muestra en la figura 3.5c. No es recomendable para rechazar una fuente sonora frente a otra.



(a) Cardioide. [8] (b) Supercardioide. [8] (c) Omnidireccional. [9]

Figura 2.2: Patrones de directividad más comunes en micrófonos.

Se puede encontrar más información de estos y otros tipos de patrones de directividad en [10]. En el presente proyecto, se han elegido micrófonos con patrón polar omnidireccional, ya que al encontrarse estos en un array situado en una pared se requieren al menos 180º de ángulo de captación, por lo que es el más adecuado. Los micrófonos que se van a utilizar son:

- Modelo Shure series MX391/O: se trata de micrófonos de superficie omnidireccionales de condensador electrect, y su apariencia se puede observar en la figura 2.3a. Al tener un tamaño pequeño, permiten construir el array con mayor facilidad, y además proporcionan una alta calidad de grabación gracias a su alta sensibilidad (siendo de -21.5 dB (81.4 mV/Pa), en el caso de los omnidireccionales) y su amplio rango de frecuencia (comprendido entre 50 y 17000 Hz). Disponen también de un preamplificador de ganancia, el cual conseguirá aumentar el nivel de la señal de entrada actuando directamente sobre la tensión de entrada, mejorando su calidad. Para más información sobre este modelo se puede visitar [11].
- Modelo Sennheiser series MKE 2-P-C y MKE 2-5-C: se trata de micrófonos enchufables de condensador omnidireccionales para proporcionar alta calidad acústica y solidez. Estos tienen un tamaño todavía más reducido que los Shure series MX391/O, y proporcionan una grabación de audio de alta calidad. Su rango de frecuencias está comprendido entre los 20 y 20000 Hz, por lo que es más amplio que en los micrófonos anteriores, aunque tienen una sensibilidad mucho menor, siendo esta de -2.5 dB (5 mV/Pa). Este factor es importante tenerlo en cuenta, ya que la sensibilidad es el nivel de salida que se va a tener en el micrófono ante una señal acústica, por lo que va a permitir conocer cómo de sensible es el micrófono a la hora de captar sonidos débiles, por lo que este modelo tendrá más dificultades para captar la señal que el anteriormente descrito. Su apariencia se puede ver en la figura 2.3b. Para obtener más características de este micrófono visitar [12].

(a) Micrófono Shure MX391/O. [11] (b) Micrófono Sennheiser MKE 2-P-C. [12]

Figura 2.3: Micrófonos utilizados en el proyecto.

Debido a que el mayor rango de frecuencias es el perteneciente a los del modelo Sennheiser, siendo este de 20000 Hz, por el Teorema de Nyquist se sabe que, como mínimo, se debe muestrear la señal de entrada con una frecuencia igual o superior a 40000 Hz. Las frecuencias de muestreo más comunes y que cumplen esta condición son las siguientes, y por lo tanto son las que se deben utilizar a la hora de adquirir las señales de audio.

- 44100 Hz
- 48000 Hz
- 88200 Hz
- 96000 Hz

Se debe tener en cuenta que, al estar utilizando micrófonos de tipo condensador, es necesario emplear alimentación *phantom* para su correcto funcionamiento, la cual es una tensión continua (DC) que se encarga de alimentar este tipo de micrófonos y, en su caso, los circuitos adicionales que pudieran albergar. Estos pueden ser, por ejemplo, un circuito preamplificador de la señal generada en la cápsula, convertidores de impedancia, etc.

El valor más frecuente de esta alimentación es de +48 V. Cabe añadir que este tipo de alimentación es denominada ”fantasma” debido a que esta se envía directamente a través de los mismos cables que se utilizan para transmitir el audio.

2.2.2 Agrupaciones de micrófonos

En el presente apartado se va a proceder a explicar la arquitectura perteneciente a los arrays de micrófonos. Todo el montaje generado y utilizado de este Trabajo de Fin de Grado se lleva a cabo en la sala ***ispace*** de la Escuela Politécnica Superior de la Universidad de Alcalá de Henares. Un factor importante que se debe tener en cuenta es que esta sala dispone de algunas paredes que poseen cierta curvatura, lo cual puede afectar a la hora de realizar las medidas para conocer la posición exacta, tanto de los arrays como de los micrófonos que se encuentran dentro de este entorno. Para poder apreciar de forma más clara este problema, en la figura 2.4 se puede ver el plano de la sala.

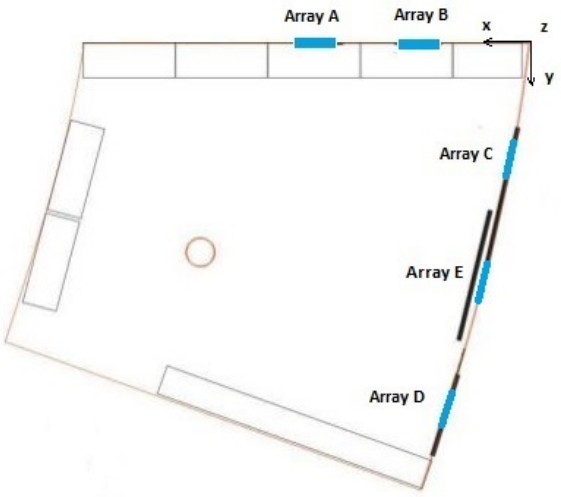


Figura 2.4: Planos de la habitación ***ispace***.

Tal y como se puede ver en la figura 2.4, en total se dispone de cinco arrays o agrupaciones de micrófonos, distribuidos de tal manera que se intenta cubrir en la medida de lo posible todas las posiciones en las que se pueda encontrar un locutor en la zona útil del espacio. Para seguir un criterio, estos arrays se nombran en orden alfabético de izquierda a derecha como: array A, array B, array C, array D y array E, respectivamente.

Hay que tener en cuenta que de todos estos arrays no todos disponen de micrófonos que estén correctamente conectados y funcionando. Los únicos que cumplen esto son los arrays A y C, y en el presente proyecto se han implementado las conexiones necesarias para que también estén operativos los micrófonos situados en los arrays B, D y E. Todo esto se explicará más adelante en el capítulo 3.

Los micrófonos que se encuentran funcionando en los arrays A y C son el modelo Shure series MX391/O, de los cuales se han descrito sus características en el apartado 2.2.1 del presente capítulo. Para que el sistema de localización sea adecuado, los micrófonos dentro de los arrays deben tener una cierta estructura física (geometría) que se diseña con el objetivo de tener buenas prestaciones en tareas de localización, la cual se muestra a continuación en la figura 2.5. Como se puede observar, se tienen tres micrófonos situados en línea en la parte inferior del array, con los cuales el sistema será capaz de discriminar mejor todas las posiciones horizontales de la sala. Además, se añade un micrófono adicional en la parte superior del array, el cual será el encargado de ayudar en la discriminación de la posición vertical, de manera que con el conjunto total de micrófonos que conforman un array el sistema será capaz de determinar cualquier posición que ocupe el hablante.

En el caso del array A, la separación entre los micrófonos 0, 1 y 2 es de 20 cm, y entre los micrófonos 1 y 3 se tienen 30 cm. De igual forma, en el array C la separación entre los micrófonos 4, 5 y 6 es de 20 cm, y entre el micrófono 5 y 7 existen 30 cm. El convenio que se ha seguido para nombrar el orden de los micrófonos es el mostrado en dicha figura.

Figura 2.5: Estructura de los arrays A y C.

Dentro de la sala del ***ispace*** se han realizado diversas medidas para determinar las posiciones que tienen las agrupaciones de micrófonos. Estos resultados se muestran en la siguiente tabla 2.1, donde las medidas están en metros, y las coordenadas se refieren al sistema de referencia mostrado en la figura 2.4.

Por otro lado, los micrófonos que componen los arrays B y D son el modelo Sennheiser series MKE 2-P-C y MKE 2-5-C, de los cuales, al igual que en el caso anterior, se han descrito sus características en el subapartado 2.2.1 del presente capítulo. La estructura de los micrófonos que siguen estos arrays para la adecuada localización del hablante es igual que la de los arrays anteriores, y se puede ver en la figura 2.6.

Al igual que en los arrays descritos anteriormente, en el array B, la separación entre los micrófonos 8, 9 y 10 es de 20 cm, y entre los micrófonos 9 y 11 se tienen 30 cm. De igual forma, en el array D la

Array A			
Micrófono	x (m)	y (m)	z (m)
Mic0	3.482	0.020	2.261
Mic1	3.282	0.020	2.261
Mic2	3.082	0.020	2.261
Mic3	3.282	0.020	2.561
Array C			
Micrófono	x (m)	y (m)	z (m)
Mic4	0.296	1.970	2.270
Mic5	0.330	2.168	2.270
Mic6	0.364	2.364	2.270
Mic7	0.330	2.168	2.570

Tabla 2.1: Posiciones micrófonos arrays A y C.

separación entre los micrófonos 12, 13 y 14 es de 20 cm, y entre los micrófonos 13 y 15 hay 30 cm. El convenio que se va a seguir para nombrar el orden de los micrófonos es el mostrado en dicha figura.

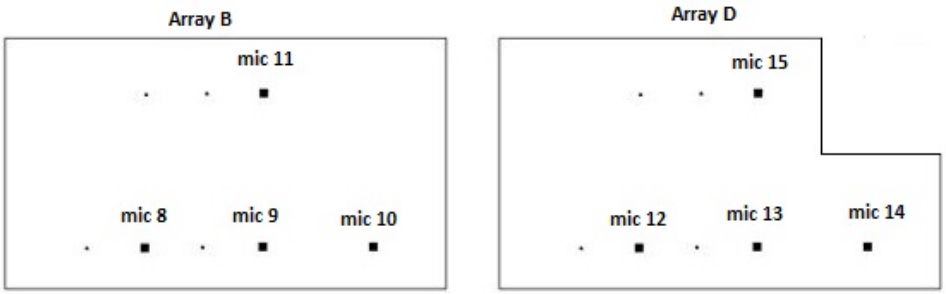


Figura 2.6: Estructura de los arrays B y D.

Las posiciones que tienen los micrófonos de los arrays B y D se muestran en la siguiente tabla 2.2, donde las medidas están en metros, y las coordenadas están referidas también al sistema de referencia mencionado anteriormente.

Array B			
Micrófono	x (m)	y (m)	z (m)
Mic8	2.025	0.020	2.261
Mic9	1.825	0.020	2.261
Mic10	1.625	0.020	2.261
Mic11	1.825	0.020	2.561
Array D			
Micrófono	x (m)	y (m)	z (m)
Mic12	1.111	5.702	2.270
Mic13	1.164	5.895	2.270
Mic14	1.216	6.087	2.270
Mic15	1.164	5.895	2.570

Tabla 2.2: Posiciones micrófonos arrays B y D.

Por último, el array E está compuesto por micrófonos del modelo Shure series MX391/O. Para este array se ha decidido cambiar la topología, utilizando un total de ocho micrófonos y distribuyéndolos en

forma de heptágono, dejando un octavo situado en el centro, tal y como se puede ver en la figura 2.7. Con esta configuración se pretende conseguir determinar una posición más precisa del hablante, determinando no solo la horizontal o la vertical, sino también el ángulo, por lo que se hará una detección más exacta. En este caso, el heptágono se ha trazado inscrito en una circunferencia de diámetro 30 cm.

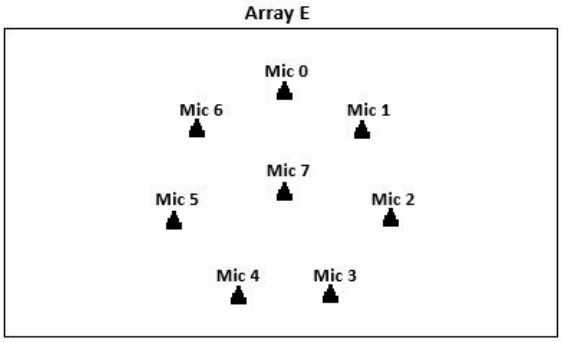


Figura 2.7: Estructura del array E.

Al igual que los arrays anteriores, es necesario determinar su posición en la sala. Estos resultados se muestran en la siguiente tabla 2.3, donde las medidas están en metros, y las coordenadas que se siguen son las mostradas en la figura 2.4.

Array A			
Micrófono	x (m)	y (m)	z (m)
Mic0	0.786	3.962	2.587
Mic1	0.837	4.092	2.520
Mic2	0.845	4.112	2.400
Mic3	0.806	4.012	2.285
Mic4	0.755	3.882	2.285
Mic5	0.727	3.812	2.400
Mic6	0.735	3.842	2.520
Mic7	0.786	3.962	2.429

Tabla 2.3: Posiciones micrófonos array E.

2.2.3 Previo y convertidor A/D

Las señales que generan los diferentes micrófonos al captar la onda sonora son muy débiles, por lo que se requiere de un circuito que amplifique estas señales de bajo nivel antes de utilizarlas posteriormente para su procesamiento. El previo será el dispositivo encargado de realizar esta tarea, y en definitiva, se encarga de proporcionar más ganancia, filtrar y modificar la señal, devolviendo una señal apta para poder trabajar sobre ella.

En este proyecto el previo utilizado es el modelo RME OctaMic II, requiriéndose tres de ellos para poder trabajar con los veinticuatro canales. La apariencia de estos se puede ver en la figura 2.8. Además, posee un ADC interno encargado de digitalizar la señal de audio proveniente de los micrófonos, y se encarga de aplicar la alimentación *phantom* de +48 V requerida sobre los micrófonos de condensador.

Figura 2.8: Previo RME OctaMic II. [6]

Asimismo, tiene incorporados diferentes controles e indicadores LEDs que permiten, por ejemplo:

- Control de ganancia que se aplica sobre cada uno de los canales con el fin de ajustar el nivel de la amplificación.
- Control de la activación/desactivación de la alimentación *phantom*.
- Indicador de saturación de la señal.
- Ajuste de la frecuencia de sincronismo de todos los canales, pudiéndose elegir su valor, si proviene de una fuente externa o se fija de forma interna, etc.
- Inversión de fase de la señal.

En el capítulo 3, apartado 3.3 se describirán en profundidad todos estos aspectos, además de mostrarse las conexiones y configuraciones elegidas para el correcto funcionamiento de este proyecto.

2.2.4 Tarjeta de adquisición multicanal

Tras ser amplificada y digitalizada, la señal de audio debe ser muestreada para poder procesarla posteriormente. De esta tarea se encarga la tarjeta de adquisición de audio multicanal, que en este proyecto es el modelo PCI RME Hammerfall DSP 9652. Su apariencia se puede ver en la figura 2.9, y al igual que el dispositivo anterior, sus conexiones y configuraciones necesarias se describirán detalladamente en el capítulo 3, apartado 3.3.

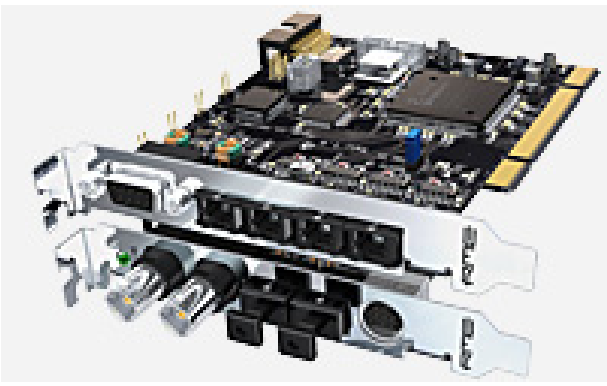


Figura 2.9: Tarjeta de adquisición multicanal PCI RME HDSP 9652.[7]

2.3 Software

2.3.1 Bloque de adquisición de audio multicanal

Este primer bloque va a ser la base para todos los que posteriormente se describen, ya que se encarga de la adquisición de los datos sobre los que se va a trabajar. Asimismo, se debe tener en cuenta que la calidad de las señales de audio adquiridas va a depender de los elementos utilizados en la parte hardware, y debe ser lo suficientemente buena para conseguir que el demostrador sea eficiente y fiable, proporcionando unos resultados que se ajusten a la realidad. Este bloque, por lo tanto, se encargará únicamente de transmitir correctamente las muestras que se obtienen de la tarjeta de adquisición de audio multicanal.

Para implementar y dar soporte a la adquisición de audio multicanal, se utilizan en primer lugar, los micrófonos descritos en el apartado 2.2.1 del presente capítulo, en segundo lugar los previos (con sus ADC) mostrados en la sección 2.2.3, y la tarjeta de adquisición de audio multicanal mostrada en la sección 2.2.4. Estos dos últimos se encargan de preamplificar, filtrar y digitalizar la señal de audio obtenida en los micrófonos. Es importante tener presente que, dependiendo del tipo de micrófono elegido, la frecuencia de muestreo utilizada y la cuantificación realizada a la hora de adquirir el audio debe adaptarse al ancho de banda de la voz y sus características, lo que está garantizado porque el hardware usado es para audio profesional.

Dado que al tratarse de una aplicación en la que los micrófonos se sitúan alejados a una distancia superior a un metro del locutor, estos van a ser muy sensibles a problemas de reverberación, ruido aditivo y baja relación señal a ruido, por lo que es importante situar los micrófonos dentro de agrupaciones de micrófonos con el fin de mejorar la calidad de la señal de audio capturada, reduciendo y eliminando los posibles errores y ruidos que pueda introducir el entorno.

Para la elaboración de la librería encargada de la captura de audio, se definen un conjunto de funciones que, haciendo uso de las librerías *RtAudio* y $SNDFile$, conforman una librería para la adquisición, reproducción y generación de audio. Esta librería da soporte a un sistema de adquisición y procesamiento de muestras de audio multiformato, multicanal y con diferentes frecuencias de muestreo en tiempo real, dotando de flexibilidad al sistema para adaptarse a cualquier requisito.

2.3.2 Bloque de cálculos de correlación

En este bloque se realizarán los cálculos de correlación CSP (*Cross-Power Spectrum Phase*), los cuales estarán basados en medir las diferencias temporales entre las señales adquiridas entre cada uno de los pares de micrófonos de los arrays disponibles. Es importante que todos los cálculos realizados en este bloque se almacenen, ya que, posteriormente, los bloques de detección de actividad de voz y estimación de la posición de la fuente sonora harán uso de ellos.

La técnica de la correlación cruzada es efectiva para la estimación del retardo temporal cuando las señales solo se ven afectadas por una fuente de ruido incorrelado, aunque no es tan eficiente cuando se encuentra en presencia de una fuerte reverberación, ya que la señal mostrará una alta correlación con sus réplicas. Por ello, se ha partido de la *Generalized Cross Correlation (GCC)* [13], que consiste en aplicar un prefiltrado a las señales antes de calcular su correlación con el objetivo de mejorar los resultados que ofrece la correlación cruzada común. Para realizar este filtrado se utiliza el sistema de ponderación *Phase Amplitude Transform (PHAT)*, el cual se usa con el fin de obtener una amplitud unitaria para todas las componentes de frecuencia, preservando al mismo tiempo las fases que contienen la información sobre el retardo temporal. De esta forma resulta el algoritmo GCC-PHAT.

Los cálculos mencionados consisten en determinar, para cada una de las combinaciones posibles entre pares de micrófonos, los siguientes elementos:

- Obtención del espectro de las señales de un par de micrófonos.
- Segmentación de las señales a lo largo del tiempo en ventanas.
- Cálculo de la Transformada de Fourier para cada una de las ventanas obtenidas. Este espectro es el utilizado para obtener la diferencia de fase entre las dos señales a través de la siguiente ecuación:

$$\phi(t,f) = \frac{S_1(t,f) \cdot S_2^*(t,f)}{|S_1(t,f)| \cdot |S_2(t,f)|} \tag{2.1}$$

- Cálculo de la Transformada inversa, obteniéndose así el valor de correlación, tal y como se puede ver en la siguiente ecuación:

$$C(t,\tau) = \int_{-\infty}^{+\infty} \phi(t,f) e^{j2\pi f \tau} df \tag{2.2}$$

- Obtención del valor de CSP máximo correspondiente a cada ventana.

Con los valores de CSP máximos obtenidos, en el bloque de detección de actividad de voz se logrará distinguir entre si la ventana contiene voz o solo silencio, ya que valores elevados de CSP significarán que en esa ventana se contiene voz, mientras que valores muy pequeños indicarán que no la hay.

2.3.3 Bloque de detección de actividad de voz

En este apartado se va a describir el funcionamiento del módulo de detección de actividad de voz (o VAD (*Voice Activity Detector*)), el cual se va a encargar de determinar los periodos de la señal donde se contiene o no voz. Esta tarea puede resultar complicada debido a que se tendrá la presencia de ruido además de la señal de voz.

La estructura básica de un detector de actividad de voz se muestra en la figura 2.10, y las fases que este sigue son:

Figura 2.10: Estructura del bloque VAD.

- **Segmentación**

  Esta etapa consiste en procesar la señal de entrada mediante segmentos o ventanas (frames o tramas), las cuales tienen normalmente una duración de entre 20-40 ms. Se elige este tamaño de ventana en base a las propiedades de la señal de voz, para conseguir un adecuado compromiso entra la resolución temporal y espectral del análisis. Las suposiciones iniciales más comunes son:

  – El ruido ambiente es aditivo a la señal de voz, es decir, la energía en los periodos de voz será la suma de la energía de la señal de ruido más la energía de la señal de voz limpia.

- El segmento de señal de voz tiene un valor de energía mayor que el segmento de ruido ambiente.
- La voz es estacionaria en periodos de tiempo cortos, por ejemplo, T <40 ms.
- El ruido es también estacionario para periodos de tiempo mucho más largos, por ejemplo, T >2 seg.
- La voz tiene más componentes periódicas que el ruido.
- El espectro de voz es más ”organizado” que el correspondiente al ruido.

- **Extracción de características**

  Esta etapa se encarga de obtener información de la señal segmentada, la cual se puede obtener tanto del dominio del tiempo como del dominio de la frecuencia. En este proyecto se va a utilizar la relacionada con el dominio temporal, ya que mediante las agrupaciones de micrófonos se pueden extraer los valores de la diferencia en el tiempo de llegada de la onda sonora a cada par de micrófonos, obteniéndose una señal retrasada en el tiempo y atenuada respecto a otra a partir de las dos señales obtenidas. En esta idea se centran los algoritmos basados en TDOA para localización. Posteriormente se integran los cálculos obtenidos para cada par de micrófonos en un solo valor global. Se van a tener dos módulos:

  - Módulo de estimación de la métrica de cada par de micrófonos: es el método más común para determinar la diferencia en el tiempo de llegada, haciendo uso de las señales procedentes de un par de micrófonos. Requiere el cómputo de la función de correlación cruzada *Cross Correlation (CC)*, y se va a implementar la técnica basada en el análisis del *Cross-Power Spectrum Phase*, también llamado *Generalized Cross-Correlation*, explicada en el apartado 2.3.2.

    La correlación se trata de una medida de similitud entre dos señales, siendo máxima cuando dos señales sean similares en forma y estén en fase. De esta forma, cuando se estén analizando señales que corresponden a voz, tendrán una morfología que presentará una mayor similitud que en el caso de ruido ambiente y se obtendrán valores máximos de correlación para los desplazamientos temporales que produzcan mayores coincidencias entre ambas señales.
  - Módulo de integración de métrica para pares de micrófonos: se encarga de procesar para cada trama el valor de correlación máximo de todos los pares de micrófonos que se han utilizado para el cálculo de las correlaciones individuales. La estrategia implementada se basa en el cálculo del máximo valor de entre todos los datos de CSP disponibles de todas las combinaciones de pares de micrófonos.

    El número total de pares de micrófonos sobre los que se va a realizar el cálculo viene dado por la suma de las posibles combinaciones de dos micrófonos que se puedan hacer dentro de un mismo array, teniendo en cuenta el conjunto de todos los arrays disponibles. Se ha llevado a cabo esta implementación para crear un sistema flexible ante posibles incorporaciones de nuevos arrays de micrófonos.

- **Decisión**

  Esta etapa se encarga de determinar si la porción de señal analizada es voz o no. Para implementar esto, se utiliza una máquina de estados o autómata que se encarga de determinar en qué estado se encuentra el sistema en cada momento. Se tendrán 3 estados distintos: espera, voz o no voz. Dependiendo del estado en el que se encontrara en momentos anteriores y mediante la comparación del resultado de los cálculos de CSP con un umbral de decisión, se puede determinar si la ventana de audio actual se trata de voz o ruido. Los parámetros que determinarán en qué estado se encuentra el sistema en cada instante son los que corresponden a la duración de la voz y la duración del silencio,

estableciendo un cierto margen para posibles pequeños silencios que no supongan una ausencia de voz.

El correcto funcionamiento de este bloque de detección de actividad de voz o VAD va a ser crucial para el correcto funcionamiento del sistema. Esto se debe a que, si se detectan periodos de silencio, se podrán descartar estas tramas. Gracias a esto, no entrarán en escena ni el bloque de cálculo de potencia acústica dirigida ni el módulo de visualización 3D, por lo que no se malgastarán recursos del procesador en cálculos innecesarios, y de esta forma se tendrá un mejor funcionamiento del sistema, siendo más eficiente. De esta forma solo se realizarán cálculos sobre los segmentos de señal que sean voz, optimizando el funcionamiento.

2.3.4 Bloque de estimación de la posición de la fuente sonora

El objetivo del presente bloque es localizar al hablante dentro del espacio inteligente, y va a depender directamente del bloque de detección de actividad de voz, ya que, si éste determina que una ventana no contiene voz, la trama se descartará, por lo que no se aplicará sobre ella el algoritmo de localización de la fuente sonora.

Este algoritmo se denomina SSL (*Sound Source Localization*), y está basado en cálculos de respuesta en potencia dirigida con transformación de fase, SRP-PHAT (*Steered Response Power with Phase Amplitude Transform*), el cual se encarga de evaluar la actividad acústica en localizaciones específicas, orientando el patrón de directividad del array utilizando la técnica de *beamforming*.

Técnica de *beamforming*

Es muy importante elegir micrófonos que permitan el diseño e implementación de un sistema de localización basado en patrones de directividad dirigidos para poder llevar a cabo el método de *beamforming*, el cual permite dirigir el patrón de directividad de un array hacia las distintas direcciones espaciales. Esta técnica es necesaria, ya que, por lo general, el locutor no permanecerá quieto en un sitio, sino que se moverá libremente por el espacio, siendo necesario realizar un seguimiento de su localización con el fin de explotar las características de los arrays de micrófonos: centrando el patrón de recepción en los alrededores de la localización de la fuente con el objetivo de evitar ruidos indeseados, otras posibles fuentes de audio y reverberaciones.

El patrón de directividad de un array lineal de sensores idénticos y equi-espaciados depende del número de micrófonos que componen dicho array, la separación entre estos y la frecuencia. Además, la máxima ganancia se tiene para las señales que se reciben de una dirección perpendicular al plano que forma el array ($\phi = 90^{o}$).

Con el fin de mejorar la señal capturada, se va a utilizar la técnica de *Filter-and-sum Beamformer*, ya que suma en fase las señales deseadas (las que provienen del plano perpendicular con un ángulo de 90^{o}) aumentando su potencia, y disminuye la potencia de las señales procedentes de posiciones no deseadas y ruidos al sumarlas.

Técnicas de estimación de la dirección de llegada

Para obtener la diferencia del tiempo de llegada de la voz a los distintos pares de micrófonos que hay en un array TDOA (*Time Difference of Arrival*) se utiliza la técnica de análisis de CSP, también conocida como GCC, descrita en el bloque de cálculos de correlación, en el apartado 2.3.2 de este mismo capítulo.

Una vez se conoce la diferencia temporal entre las señales de dos micrófonos, se puede hacer uso de ella junto con la información relativa a la posición espacial de los micrófonos con el fin de generar

hiperboloides de revolución en 3D que representan los lugares geométricos donde puede encontrarse el locutor de acuerdo con el TDOA obtenido. Éstas curvas hiperbólicas generadas por un único par de micrófonos, se intersectan con las curvas obtenidas del resto de micrófonos dando lugar a una estimación de la localización del hablante.

Estas técnicas de estimación de la dirección de llegada hacen uso de *beamforming*, pudiendo ser aplicado tanto a la captura de la señal de audio como a la localización de la fuente sonora. En este caso, la localización de la fuente no es conocida, por lo que el *beamformer* se usará para dirigir el array sobre un conjunto de localizaciones espaciales en un espacio de búsqueda predefinido, conociéndose esta técnica como *Steered Response Power*. Posterior a esto, se utiliza un estimador *Maximum Likelihood (ML)* con el fin de buscar un máximo en la potencia de salida, el cual debe coincidir con la localización del hablante.

Con el fin de conseguir mejores resultados, se hará uso de filtros PHAT, que han demostrado ser útiles en términos de la estimación de TDOA, resultando así una técnica robusta que crea una equivalencia entre SRP y la suma de todas las transformadas de fase de las posibles combinaciones de pares de micrófonos. Esta técnica se denomina SRP-PHAT y su robustez radica en el hecho de que explota la redundancia espacial de los micrófonos promediando todos los posibles pares de cruces GCC-PHAT.

Problemas del algoritmo SRP-PHAT

Esta técnica dependiente de la potencia puede presentar en la práctica picos en un determinado número de localizaciones incorrectas, debido a las condiciones de reflexión del entorno o al efecto de la geometría del array, induciendo a error los resultados de la localización. La elección de los filtros adecuados puede ayudar a minimizar estos efectos. Como se ha visto, la estrategia seguida por *Phase Transform* PHAT de ponderación de cada componente frecuencial por igual ha demostrado obtener respuestas correctas en situaciones prácticas.

2.3.5 Bloque de visualización 3D

En este último bloque se hará una representación virtual del espacio inteligente donde se ha realizado las pruebas en 3 Dimensiones (3D). Esto permitirá la visualización de la localización final obtenida, de forma que se dará una idea más clara de la localización del hablante.

En dicho entorno se dispone tanto de la representación del entorno donde se lleva a cabo este trabajo, la sala *ispace*, como la representación de ciertos objetos que permiten una interacción hombre-máquina a modo de actuadores del espacio inteligente. Todos los elementos que forman este espacio se pueden ver en la figura 2.11. Asimismo, este bloque posibilita hacer cambios de vistas, proyecciones, observar cambios en los actuadores del entorno virtual o representar la posición del locutor dentro de dicho entorno.

Para desarrollar este sistema de visualización virtual se ha hecho uso de la librería de gráficos `OpenGL`, que define una *Application Programming Interface (API)* multilenguaje y multiplataforma que permite escribir aplicaciones que produzcan gráficos 2D y 3D. La interfaz consiste en más de 250 funciones diferentes que pueden usarse para dibujar escenas tridimensionales complejas a partir de primitivas geométricas simples, tales como puntos, líneas y triángulos. Para más información sobre `OpenGL` se puede consultar en [14].

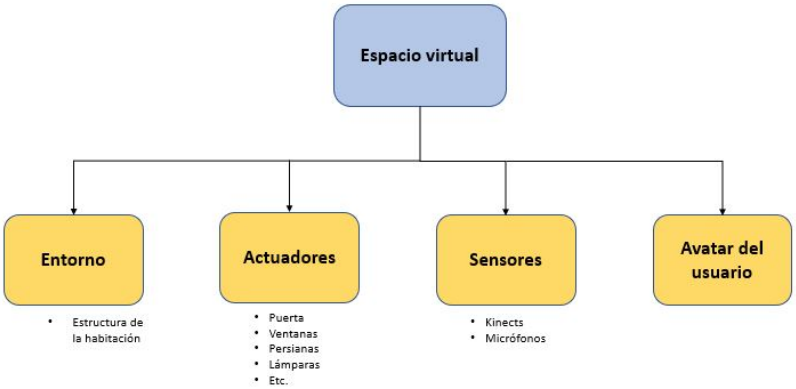
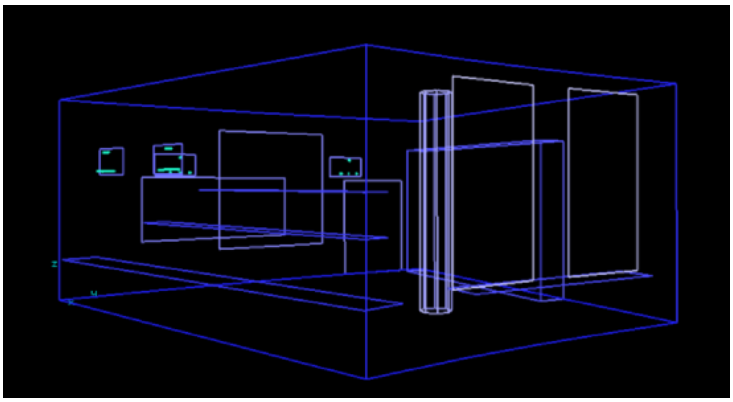
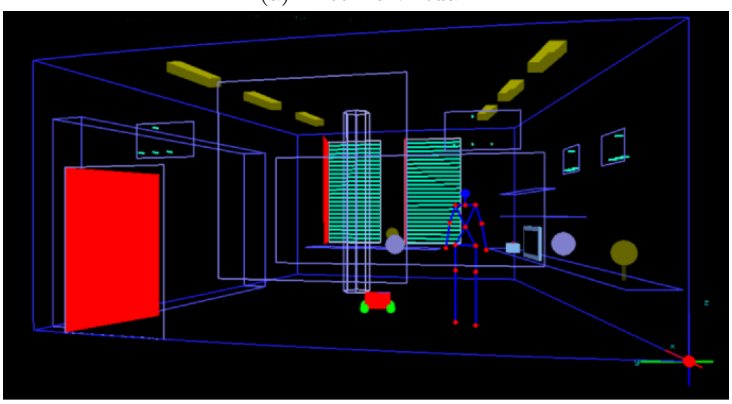


Figura 2.11: Elementos que forman el espacio virtual.

La representación visual que nos ofrece este sistema consta de una base o entorno sobre el que aparecen una serie de objetos, como son los actuadores y sensores. El entorno del espacio virtual constituye la estructura del mismo y se puede observar en la figura 2.12a. Únicamente incluye las estructuras básicas que permiten modelar el propio espacio. Para ello el programa desarrollado carga un fichero base en el que se encuentra definida la estructura del entorno y que incluye: suelo, techo, las cuatro paredes, marco de la puerta, dos marcos de ventana, una columna central, dos mesas laterales, una balda larga en la pared derecha, una balda corta en la pared derecha, un armario en la pared izquierda, una pizarra, una superficie de proyección y cuatro marcos donde se sitúan cuatro arrays de micrófonos que se encuentran en la sala.



(a) Entorno virtual.



(b) Entorno virtual con objetos.

Figura 2.12: Visualización 3D.

Esta estructura no es modificable por el usuario y compone el esqueleto sobre el que se han construido el resto de los actuadores. Además, el uso de la librería `OpenGL` nos permite implementar una serie de transformaciones que permiten adaptar la vista, hacer zoom, realizar rotaciones en los tres ejes y alternar entre vista 2D y 3D.

Sobre el entorno inteligente virtual se incluyen diferentes objetos, como son los actuadores y sensores que podrían encontrarse en el *ispace*. Concretamente, algunos de los actuadores que se representan en este espacio virtual no corresponden con los actuadores reales que existen en la sala, sino que se ha preferido utilizar actuadores virtuales de modo que se pueda tener una idea más amplia en cuanto a los posibles objetos de ejemplo, de los cuales podría disponerse en el espacio inteligente en un futuro.

Asimismo, existe la posibilidad de añadir una serie de actuadores, que son objetos controlados por la lógica del sistema del espacio virtual y pueden realizar acciones en función de dicha lógica, como puede ser una puerta que se abre o se cierra, encender o apagar la luz del techo, persianas que se suben o se bajan... En el presente proyecto no se hará uso de esto.

En el espacio virtual se han incluido también los sensores disponibles, con el objetivo de proporcionar una referencia visual rápida al usuario que le permita saber dónde se encuentran situados en el espacio real. Los sensores que se han representado son dos sensores Kinect y los diferentes arrays de micrófonos con los que cuenta la sala.

Capítulo 3

Desarrollo

3.1 Introducción

En este nuevo capítulo se expone el procedimiento seguido para el desarrollo y mejora del funcionamiento del presente proyecto. En primer lugar, se explica el proceso de desarrollo e implementación del montaje de los micrófonos incluidos en los nuevos arrays. Para ello, se van a describir todos los pasos necesarios para realizar su conexión de forma correcta con el fin de que tengan un funcionamiento adecuado.

Además, también se incluye un apartado en el que se explicarán todas las configuraciones y conexiones necesarias para el correcto funcionamiento del sistema de los elementos hardware, que serán los micrófonos, el previo y la tarjeta de adquisición de audio multicanal, así como el software necesario para su configuración.

Por último, se va a describir como se ha realizado el programa de detección de pasos por cero, el cual será necesario para la etapa de evaluación y resultados. Se van a describir cada uno de los pasos seguidos para ambos procesos y por qué estos son necesarios.

3.2 Implementación de arrays de micrófonos

La finalidad del presente apartado es describir las conexiones y configuraciones necesarias para la correcta integración de los micrófonos de los nuevos arrays en el demostrador en tiempo real. Se debe tener en mente por qué es importante integrar estos nuevos arrays, por lo que a continuación se describen los factores a tener en cuenta con respecto a los micrófonos:

- Cantidad y calidad de los micrófonos: se requieren micrófonos de alta calidad de grabación, y cuanto mayor sea el número de estos micrófonos, mayor será la calidad de la localización del hablante dentro del espacio inteligente, siendo esta más precisa. Además, se debe tener en cuenta que su calidad conseguirá hacer que el sistema sea más robusto en condiciones acústicas adversas, como por ejemplo mucho ruido ambiente, interferencias o reverberación. Las características de los micrófonos utilizados se pueden ver en el capítulo 2, apartado 2.2.1.
- Posición relativa entre pares de micrófonos y su posición frente a la fuente sonora: se busca que los micrófonos estén situados correctamente y con una estructura adecuada, ya que estos factores pueden ser capaces de reducir drásticamente el número de micrófonos para conseguir la misma calidad en la localización. La estructura de estos arrays se describe en el capítulo 2, apartado 2.2.2.

3.2.1 Implementación micrófonos arrays B y D

Para estos arrays B y D se van a utilizar los micrófonos del modelo Sennheiser, y para realizar las conexiones estos micrófonos desde el array en el que estén situados hasta la tarjeta de adquisición multicanal se ha empleado un cable de audio paralelo apantallado. Este tipo de cable dispone de una malla incorporada, la cual servirá como mecanismo con el cual se lograrán reducir las posibles interferencias electromagnéticas existentes en el ambiente de trabajo, causadas por los dispositivos de radiofrecuencia y las múltiples fuentes de interferencia que consisten los dispositivos electrónicos.

También se han empleado conectores de tres polos de cable XLR del modelo NC3MXX Neutrik macho para lograr conectar los micrófonos a la tarjeta de adquisición, y estos se pueden ver en la figura 3.1.

Figura 3.1: Conector NC3MXX Neutrik macho.

Con el fin de eliminar el ruido que pueda introducir la alimentación *phantom*, la cual es necesaria en micrófonos de tipo condensador, tal y como se explica en el apartado 2.2.1 del capítulo 2, y mejorar así la señal adquirida, se ha diseñado un filtro RC, el cual estará formado por una resistencia de 100 kΩ y un condensador de 47 μF. Este se trata de un filtro paso alto, cuya frecuencia de corte se muestra a continuación, pudiéndose observar que se trata de una frecuencia muy baja. Esto se debe a que se quiere evitar que afecte a alguna frecuencia dentro del rango del espectro de voz, el cual está comprendido entre los 20 y 20000 Hz. Asimismo, el esquema de este filtro se puede ver en la figura 3.2, donde los nodos 1, 2 y 3 se refieren a los tres pines de los que dispone el conector mencionado anteriormente.

$$f_c = \frac{1}{2\pi RC} = 0{,}0338Hz \tag{3.1}$$

Se ha decidido introducir este filtro en el interior de la cápsula del conector NC3MXX, soldando los componentes en él siguiendo el esquema mencionado anteriormente, por lo que se debe tener muy clara la forma de colocarlos con el fin de que encajen correctamente dentro del reducido espacio del conector, y que además no haya contactos indeseados para evitar que se produzca un cortocircuito. El resultado final de este montaje se puede ver en la figura 3.3, además del resultado final, pudiéndose apreciar cómo queda perfectamente encajado dentro de la cápsula, sin tener ningún problema.

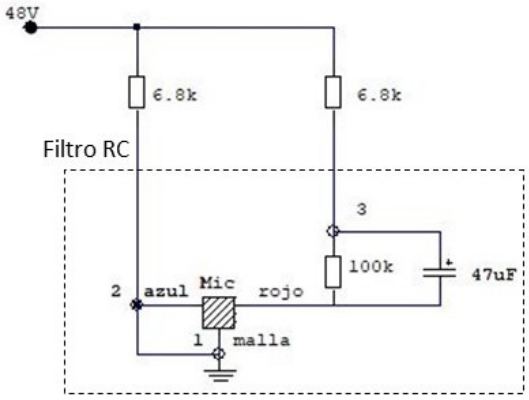


Figura 3.2: Estructura del filtro RC diseñado.

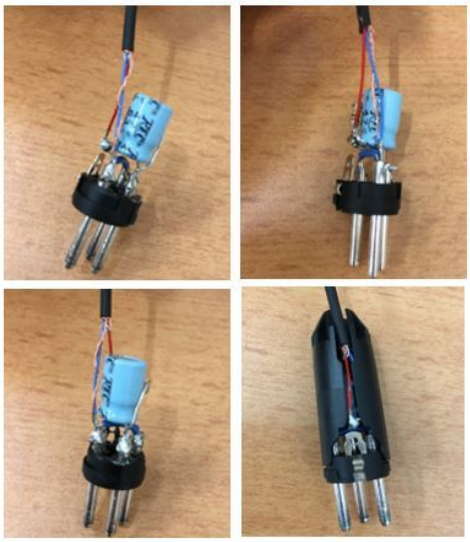
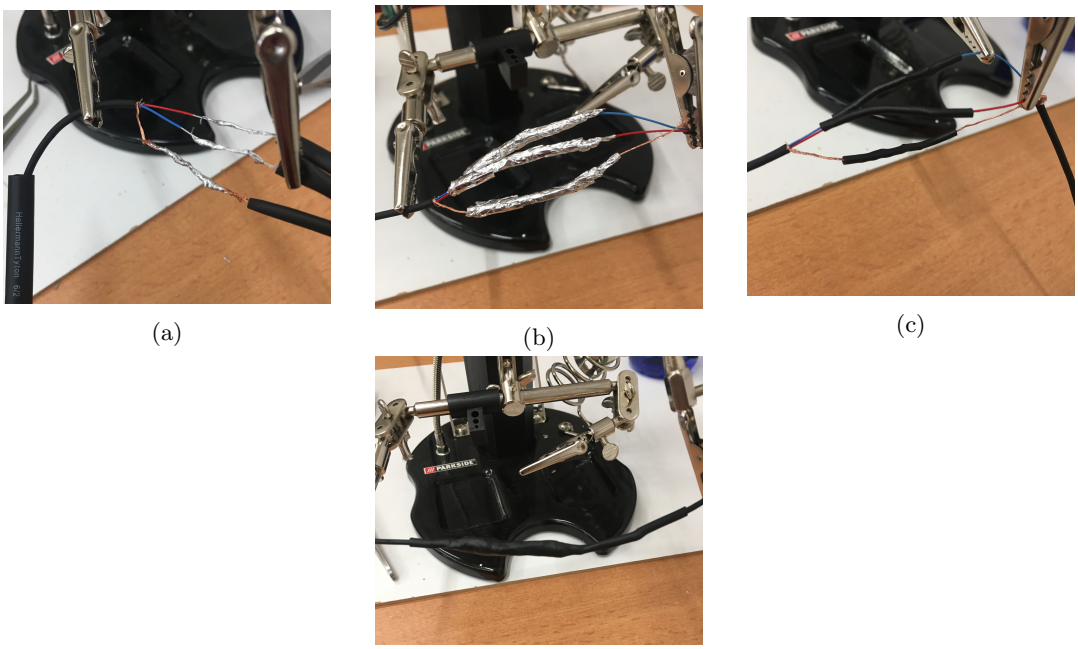


Figura 3.3: Soldadura filtro RC.

Además de este filtro, se debe solucionar un segundo problema, el cual es que el cable disponible en los micrófonos es demasiado corto, por lo que no es suficiente para llegar hasta la tarjeta de adquisición de audio multicanal. Este problema se solventará realizando un empalme de este cable a otro de mayor longitud. Los pasos a seguir para realizarlo de forma correcta son: estañar previamente todos los cables entrelazarlos entre ellos, soldándolos después, de forma que se tenga un buen contacto óhmnico, y que la unión sea fuerte. Posteriormente, cada una de las tres soldaduras realizadas (una por cada cable) se debe asegurar recubriéndolas con fundas termorretráctiles. Estas fundas consisten en un aislante que encoje al entrar en contacto con aire caliente, adaptándose a la soldadura y dejándola perfectamente sellada. Acto seguido, se ha decidido recubrir todas estas uniones con papel de aluminio, con el fin de construir una malla adicional y conseguir evitar las posibles interferencias y ruidos que puedan interferir, actuando como una especie de malla. Y, por último, se recubre todo esto con otra funda termorretráctil, sellando toda la unión. Se pueden ver todos estos pasos seguidos uno por uno en la figura 3.4.



(a) (b) (c)

(d)

Figura 3.4: Empalme cable de audio.

3.2.2 Implementación micrófonos array E

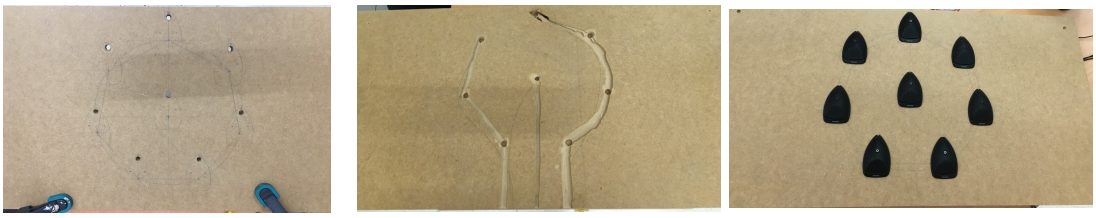
Para implementar el array E se han utilizado ocho micrófonos del modelo Shure. Estos micrófonos, al igual que los anteriores, tienen un cable demasiado corto para poder cubrir la distancia desde este array hasta los previos, por lo que se deben ampliar utilizando el mismo cable de audio apantallado que en el utilizado anteriormente. Con este fin, deben realizarse dos empalmes por cada uno de los micrófonos, realizándose estos de la misma forma descrita en el apartado anterior.

Una vez realizadas todas las soldaduras necesarias, se debe comprobar que estas son correctas. Para ello, se ha utilizado un multímetro para medir que efectivamente haya continuidad en el cable. Además, también se ha verificado si su funcionamiento es correcto a la hora de conectarlos a la tarjeta de adquisición de audio multicanal, lo cual se explica en el apartado 4.4 del capítulo 4.

Con la verificación de que las conexiones de estos ocho micrófonos están realizadas correctamente, ya se pueden situar en el tablero del array E. Se ha utilizado una tabla de madera de dimensiones 42 x 80 cm, y en ella se ha trazado un heptágono inscrito en una circunferencia de un diámetro de 30 cm. En cada uno de los vértices de este heptágono se debe situar cada uno de los micrófonos. Asimismo, el octavo se situará en el centro de dicha circunferencia.

Una vez marcadas las posiciones de los micrófonos, es necesario realizar taladros con el fin de poder fijarlos en la tabla y pasar el cable de señal. Por un lado, se debe realizar uno por cada micrófono, atravesando toda la madera, con el objetivo de poder pasar el cable a la parte trasera, y, por otro lado, dos por cada micrófono para sujetarlos en el tablero. También harán falta cuatro taladros más, uno en cada esquina del tablero, para poder fijar el array en la pared.

Además, también será necesario realizar unos surcos en la parte posterior de la madera (usando una fresadora), con la finalidad de poder conducir los cables para conseguir llevarlos hasta la tarjeta de adquisición a través de una serie de canaletas situadas en la pared. Los resultados de todos estos pasos se pueden ver en la figura 3.5, y también el resultado final.



(a) (b) (c)

Figura 3.5: Implementación del array E.

3.3 Configuración y conexionado de los elementos hardware

En este apartado se tiene como objetivo explicar todas las conexiones existentes en el sistema correspondientes a la parte hardware. La estructura global de esta parte se puede ver en la figura B.1, donde se observa que los micrófonos que se encuentran en los arrays, se deben conectar a cada uno de los previos. Puesto que estos previos disponen de ocho canales cada uno, se necesitan un total de tres para la implementación del sistema, ya que se tienen veinticuatro micrófonos.

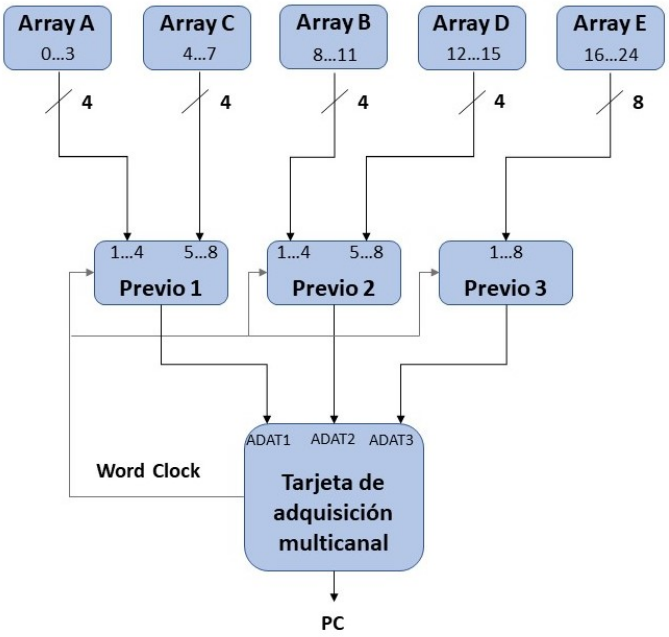


Figura 3.6: Esquema global de la parte hardware.

De esta forma, los micrófonos de los arrays A y C se conectan en el primer previo, y de igual forma los micrófonos de los arrays B y D se conectan al segundo previo. Por último, en el último previo se encuentran los micrófonos que conforman el array E.

Por último, las salidas de cada uno de los previos se deben conectar a la tarjeta de adquisición de audio multicanal, la cual a su vez se debe conectar al ordenador donde se ejecuta la aplicación del demostrador en tiempo real. Además será la encargada de proporcionan la señal de *Word Clock*, que servirá como sincronismo entre todos los elementos.

En primer lugar, es necesario configurar los previos adecuadamente, para tener una correcta sincronización y buen funcionamiento del sistema. Se han elegido los del modelo RME OctaMic II, y su fun-

cionamiento y estructura se pueden ver en el bloque 2 del apartado 2.2.3. A continuación, se describirán tanto las conexiones hardware como la configuración del software necesarias para ello. Esta información se ha consultado en el manual de usuario [15].

Lo primero que se debe hacer es configurar los *DIP Switches*. Estos son unos interruptores que se encuentran en la parte trasera de cada uno de los previos, y son los encargados de configurar aspectos como: tipo de sincronización, tipo de reloj a utilizar (interno o externo), frecuencia del reloj, tipos de velocidades existentes, etc. Su estructura se puede ver en la figura 3.7, y sus funcionalidades son:

- 1: sincronización externa con fuente *AES* o con *Word Clock*. En este caso, se decide que la sincronización sea mediante *Word Clock*, ya que se va a fijar en la aplicación la frecuencia de reloj a la hora de ejecutarla, por lo que este switch debe estar en la posición inferior.
- 2: reloj interno (*master*) o externo (*slave*). Al utilizar *Word Clock*, se va a utilizar un reloj externo, por lo que al igual que el switch anterior, este también debe estar en la posición inferior.
- 3: reloj interno a 44.1 kHz o 48 kHz. La frecuencia de muestreo utilizada para adquirir audio va a ser la de 48 kHz, por lo que se debe colocar en la posición inferior.
- 4: activa el modo de doble velocidad. En este proyecto es irrelevante, por lo que este switch debe estar en su posición inferior.
- 5: activa el modo de cuádruple velocidad. Al igual que el caso anterior, no interesa este modo, por lo que el switch se sitúa en la posición inferior.
- 6: estado de la salida *AES*, que puede ser profesional o consumidor. En este caso se elige profesional, por lo tanto, el switch debe estar en la posición superior.

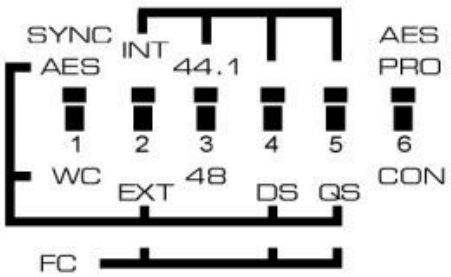


Figura 3.7: Estructura *DIP Switches*.

El *Word Clock*, que va a ser la señal de reloj que se va a utilizar para sincronizar todos los previos disponibles, se debe conectar a través de un cable coaxial. Este debe estar conectado por un lado a la salida de la tarjeta de adquisición multicanal WC output y por otro lado a la entrada WC disponible en cada uno de los previos. El mismo cable estará conectado a los tres previos a la vez, mediante terminaciones en T de 75 Ω, ya que esta entrada no está terminada por sí sola. La tarjeta de adquisición elegida para este proyecto es la PCI Hammerfall DSP 9652, y se describe su funcionalidad en el apartado 2.2.4 del capítulo 2. Sus características y configuraciones se han consultado en [16].

Además de los *DIP Switches*, también se debe configurar el interruptor que está en la parte trasera, para que se tenga una impedancia de 75 Ω, y la salida de datos de cada previo, $MAIN$, se conecta a la tarjeta de adquisición multicanal, en las entradas $ADAT$.

Una vez realizadas las conexiones necesarias y haber realizado todas las configuraciones internas, antes de adquirir el audio se debe configurar también el panel frontal de cada uno de los previos. Estos disponen

de diferentes interruptores para poder configurar elementos como: activar o desactivar la alimentación *phantom*, activar o desactivar un filtro paso alto, etc. También dispone de LEDs para indicar los diferentes estados disponibles. Todo el panel frontal se muestra en la figura 3.8, y a continuación se explica el funcionamiento de cada una de las opciones disponibles.

Figura 3.8: Parte frontal Octa Mic II.

- +48 V LED: cuando está encendido, indica que está activa la alimentación *phantom*. En este caso, debe aparecer siempre encendido en todos los micrófonos, ya que es necesario para su funcionamiento, tal y como se ha descrito en el subapartado 2.2.1.
- CLIP LED: se encenderá en color rojo cada vez que el nivel de señal supere un cierto umbral, indicando así que la señal se está recortando a causa de saturación. Se enciende 2 dB antes de que se supere dicho umbral. En este caso, está configurado como Hi Gain (ganancias altas), por lo que se encenderá al superar un nivel de salida de +17 dBu.
- SIG LED: indica la presencia de una señal de entrada. Debe estar encendido, ya que, si no lo está, no se tiene una señal de entrada correcta.
- GAIN: con este control se puede ajustar la ganancia entre +6 y +60 dB. Para realizar grabaciones es necesario que esté situada en el máximo valor, dado que los frentes de audio estarán alejados y nos interesa máxima ganancia.
- +48 switch: activa la alimentación *phantom*. En este caso debe estar activo, ya que se tienen micrófonos de condensador que requieren este tipo de alimentación para su funcionamiento, tal y como se describe en el apartado 2.2.1.
- LO CUT switch: activa un filtro paso alto con una frecuencia de corte de 80 Hz, de manera que se van a eliminar ruidos que pueda introducir la red eléctrica, ruidos de bajas frecuencias, etc. En este caso se va a tener activado.
- PHASE switch: cambia la polaridad 180º. Se utiliza en casos en los que se tienen múltiples micrófonos situados en posiciones distintas o pueda haber riesgo de tener cables mal soldados. En estos casos pueden ocurrir cancelaciones de fase o cambios de sonido. Con esto activo se pueden eliminar dichos errores, añadiendo una inversión de fase adicional, aunque no estará activo para el desarrollo del proyecto, ya que se quieren tener las señales tal y como se capturan.
- Clip Hold: se activa o desactiva cuando se presiona durante 2 segundos. Si está activo, cada vez que hay un recorte en la señal de entrada por saturación, el LED se quedará parpadeando una vez cada segundo. De esta manera se podrá saber si se han producido saturaciones o no en algún momento.
- Hi Gain / +4 dBu / -10 dBV: define el nivel de referencia de los niveles de salida. En este caso, se configura como Hi Gain (ganancias altas).

Además de todo esto, es necesario utilizar el programa Hammerfall DSP Settings para configurar las opciones correspondientes de la tarjeta de adquisición de audio multicanal. Su interfaz es la mostrada en la figura 3.9, y como se puede ver permite configurar:

- Sample Clock Source: en esta sección se puede seleccionar la frecuencia de la fuente de reloj. Se puede elegir tanto una específica, como en este caso de 48 kHz, o elegir la opción *AutoSync*, con la que detectará la frecuencia del reloj automáticamente.

- SynCheck: en este apartado se pueden ver los dispositivos disponibles y su estado. Si aparecen como *No Lock* significa que no hay señal disponible, si aparecen con *Lock* quiere decir que están conectados, pero no se encuentran sincronizados, y por último, si aparecen con el estado *Sync* quiere decir que están conectados y que además están sincronizados. Las entradas que interesan son ADAT1 In, ADAT2 In y ADAT3 In, las cuales son las tres tarjetas de adquisición, y estas deben aparecer como *Sync* para asegurar que están todas ellas sincronizadas a la frecuencia de 48 kHz.

- Preference Sync Ref: permite configurar qué fuente de reloj se quiere preferentemente. Esto puede ser el *Word clock*, alguna de las tarjetas de adquisición, etc. Como la frecuencia de reloj en este caso va a estar marcada por el *Word clock*, se selecciona este.

- AutoSync Ref: se indica la frecuencia y la entrada que se está utilizando.

- System Clock: permite ver el modo en el que está seleccionado el reloj, el cual debe ser *master*, y la frecuencia a la que está trabajando.

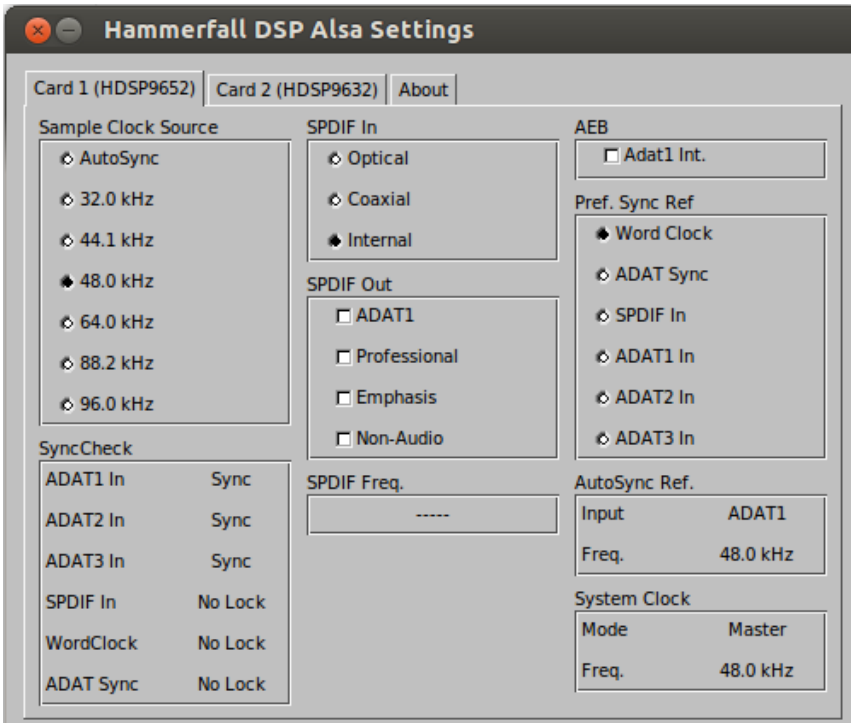


Figura 3.9: Interfaz Hammerfall DSP Settings.

Por otro lado, también se va a utilizar el programa Hammerfall DSP Mixer, el cual permite ver los niveles de señal que tienen cada uno de los micrófonos que se encuentran conectados a la tarjeta de adquisición a través de los previos. Su interfaz se puede ver en la figura 3.10.

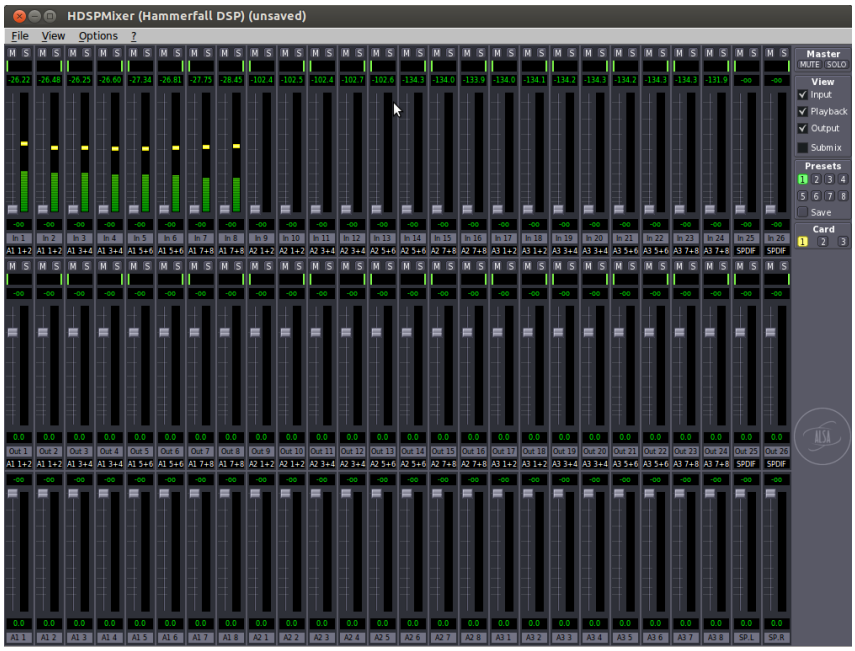


Figura 3.10: Interfaz Hammerfall DSP Mixer.

3.4 Aplicación de detección de pasos por cero

Un requisito crucial que debe cumplir este sistema de localización de hablantes dentro de un espacio inteligente es la sincronización, y por ello en el capítulo 4 se analizará con diversas pruebas. Para realizarlas, se va a requerir medir retardos entre señales para su comparación con la precisión teórica determinada a partir de la geometría de los elementos implicados. Con el fin de medir estos retardos, se ha decidido tomar como referencia los pasos por cero de las respectivas señales adquiridas. Con el objetivo de obtener resultados consistentes, se ha decidido analizar varios periodos de señal para poder comprobar que el comportamiento de las señales es siempre el mismo y para facilitar esta tarea (haciéndola más automática), se ha creado una aplicación que se encarga de detectar los pasos por cero de las señales de entrada.

Las señales de entrada se encontrarán en ficheros de audio de formato *.wav* de 32 bits flotante, por lo que el programa debe ser capaz de abrir estos ficheros para poder analizarlos posteriormente. Asimismo, es necesario que además de detectar los pasos por cero de las señales, el programa debe ser capaz de distinguir entre flancos de subida y flancos de bajada, para que a la hora de comparar los diferentes tonos se haga de forma correcta y coherente, comparando así flancos de subida con flancos de subida y flancos de bajada con flancos de bajada. Para hacer esto, el programa sigue los siguientes pasos:

- Si la muestra anterior es negativa o igual a cero y la muestra actual es positiva, entonces se tiene un paso por cero de flanco de subida.

- Si la muestra anterior es positiva o igual a cero y la muestra actual es negativa, entonces se tiene un paso por cero de flanco de bajada.

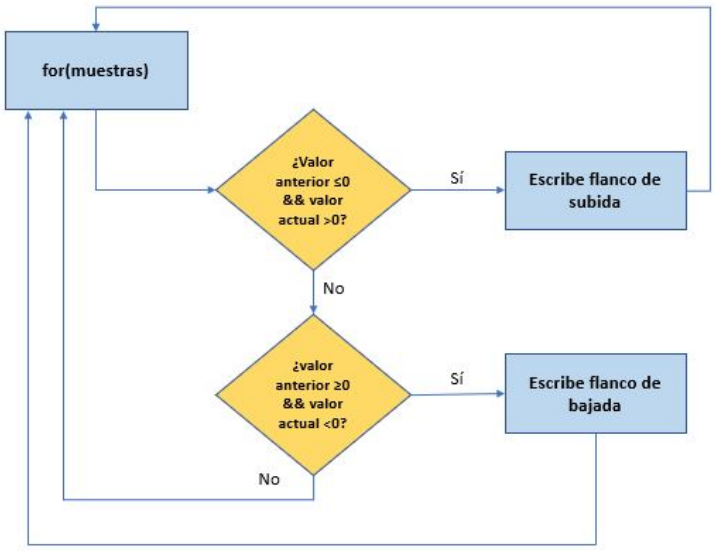


Figura 3.11: Pseudocódigo de programa detector de pasos por cero.

Cada vez que el programa encuentra un paso por cero de flanco de subida, este escribe el número de muestra donde se ha detectado en un fichero llamado *PasosPorCeroup.txt*. Se hará lo mismo con los pasos por cero de flanco de bajada, llamándose el fichero en este caso *PasosPorCerodown.txt*.

Para ejecutar este programa, únicamente es necesario pasarle como argumento de entrada la ruta del fichero del que se quieran obtener los pasos por cero.

3.5 Conclusiones

A lo largo de este capítulo se ha intentado exponer de forma clara y precisa todo el trabajo desarrollado en el montaje físico, cableado y construcción de los elementos físicos del sistema completo de arrays de micrófonos y su conexionado a los dispositivos de captura. Se han expuesto las soluciones tomadas para la implementación de los nuevos arrays, así como los problemas encontrados a la hora de llevarlas a cabo.

Además, se ha explicado de forma detallada la estructura de la parte hardware, de manera que quede más clara. Igualmente, se han descrito todas las conexiones y configuraciones para que el sistema funcione correctamente, estando todos los elementos bien sincronizados.

Se ha explicado también paso a paso el funcionamiento del programa encargado de detectar pasos por cero, tanto de flancos de subida como de bajada, para poder analizar y evaluar posteriormente el comportamiento del sistema, de forma que se verificará el sincronismo entre todos los componentes que integran el proyecto, como micrófonos, tarjeta de adquisición de audio multicanal, etc.

Capítulo 4

Evaluación y resultados

4.1 Introducción

En el presente apartado se explicarán las diferentes pruebas prácticas que se han realizado sobre el sistema de localización de fuentes de audio en tiempo real. Estos experimentos tienen el objetivo de demostrar el correcto funcionamiento del sistema completo, además de depurar y detectar los errores que tenía el software de partida, si los hubiere, con el fin de poder solucionarlos y mejorar el funcionamiento del sistema total.

Para ello, se van a realizar pruebas para verificar el sincronismo entre todos los elementos del sistema, y también se verificará el funcionamiento de todos los micrófonos implementados, para asegurar así que aportan información correcta.

Por otro lado, se comprobará el funcionamiento del demostrador en tiempo real con un número de micrófonos determinando, empezando con dieciséis y terminando con veinticuatro canales. Y, por último, se realiza una evaluación perceptual del demostrador, con el fin de comprobar si se mejora o no el funcionamiento del sistema con los micrófonos añadidos, probando primero cada array por separado, y después todos en conjunto. Finalmente, se expondrán los resultados obtenidos y se determinará la efectividad del diseño.

4.2 Verificación de los micrófonos

4.2.1 Micrófonos Sennheiser

En este apartado se pretende verificar que, una vez soldados todos los componentes necesarios en los nuevos micrófonos y realizado los empalmes oportunos, su funcionamiento es correcto. Para ello, en primer lugar, se conectan a la tarjeta de adquisición multicanal. Se debe verificar que esté activa la alimentación *phantom*, y el LED SIG debe aparecer encendido. Una vez hecho esto, con la ayuda del programa Hammerfall DSP Mixer se puede observar si funcionan adecuadamente, viendo sus niveles de señal. Estos niveles deben aumentar al hacer ruido o hablar. El funcionamiento tanto de la tarjeta de adquisición como del programa se explican detalladamente en el apartado 2.2.4 del capítulo 2.

Tal y como se vió en el apartado 2.2.1 del capítulo 2, estos micrófonos del modelo Sennheiser tienen un nivel de señal inferior a los del modelo Shure, debido a que su sensibilidad es mucho menor. Esta diferencia se puede apreciar en la figura 4.1.

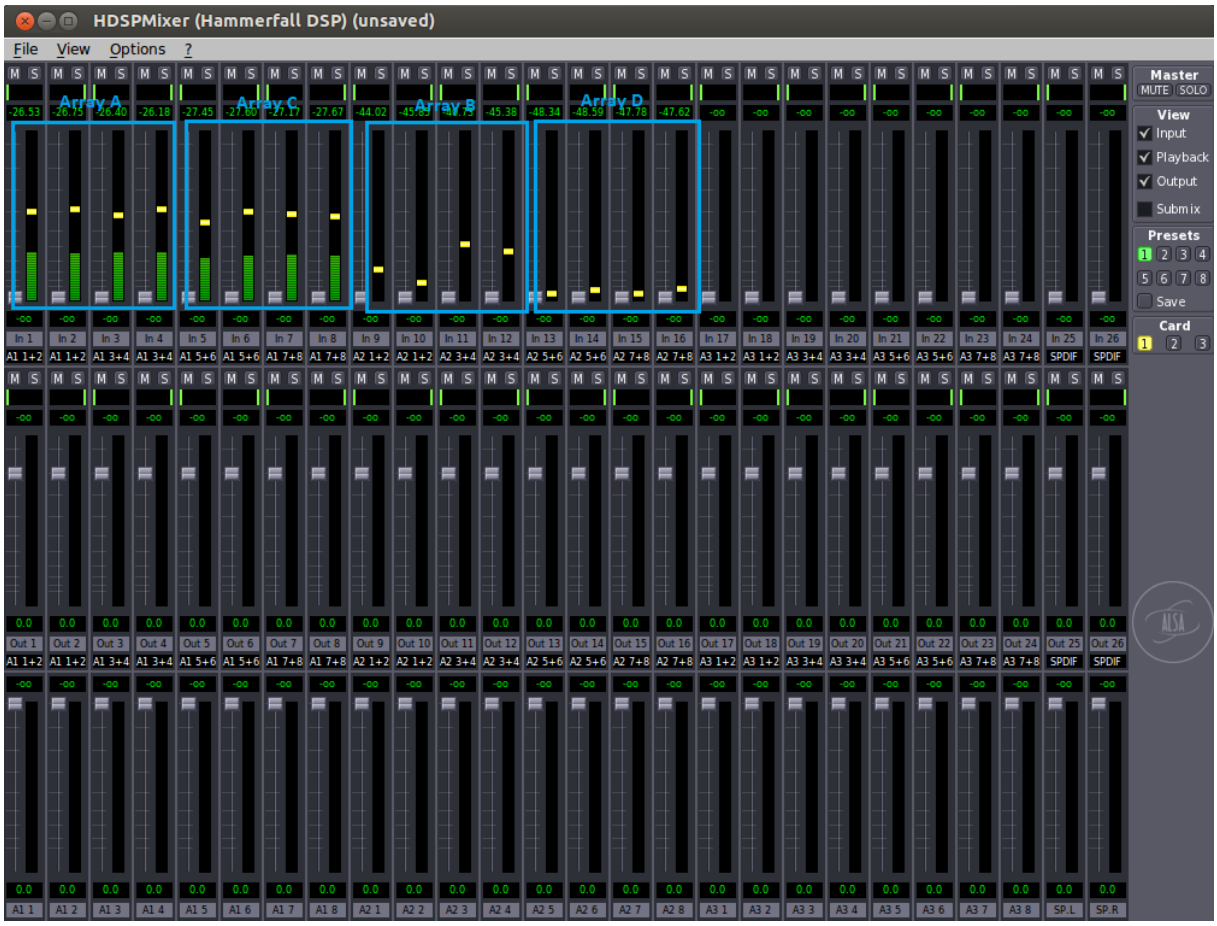


Figura 4.1: Nivel de señal de micrófonos Sennheiser en Hammerfall DSP Mixer.

También se puede comprobar este efecto en las señales grabadas, las cuales van a tener una amplitud mucho menor que en el caso de los micrófonos del modelo Shure. Se puede ver en la figura 4.2.

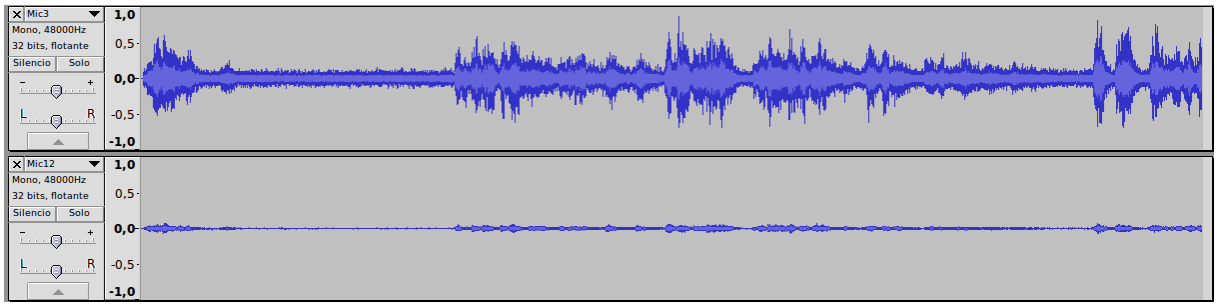


Figura 4.2: Comparativa entre señales de modelo Shure frente a modelo Sennheiser.

A la hora de ajustar la ganancia de estos micrófonos al máximo, se ha encontrado un problema, ya que cuatro de ellos tienen un nivel de señal demasiado elevado, y tienden a saturar constantemente. Se muestra en la figura 4.3 un ejemplo de este comportamiento.

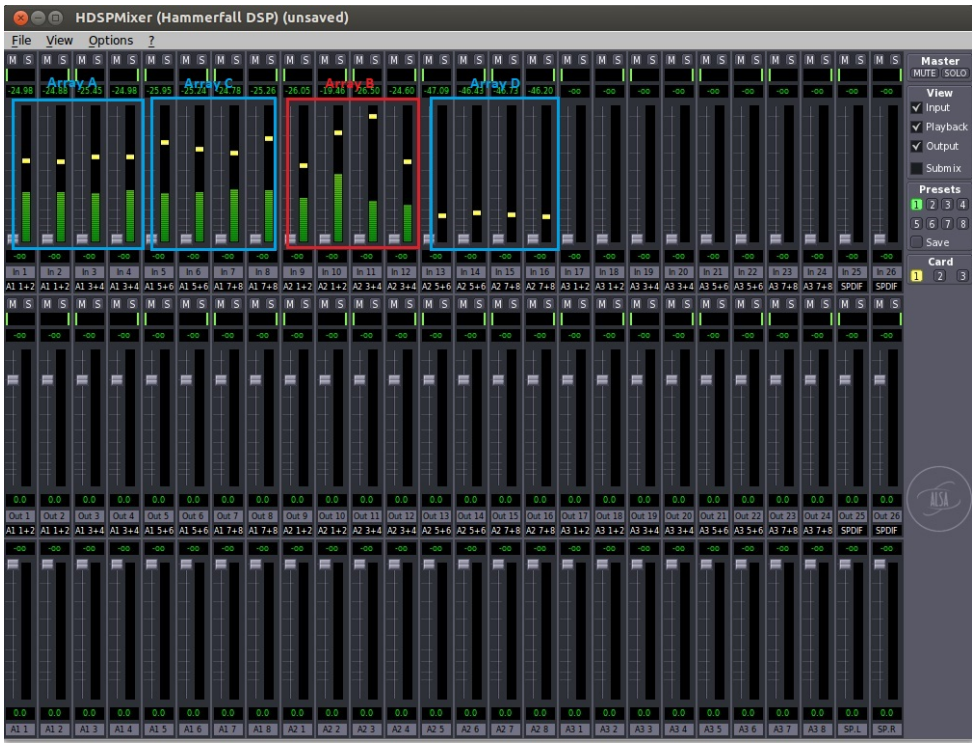


Figura 4.3: Micrófonos Sennheiser con nivel de señal inadecuado.

Con el fin de verificar si aún con estos niveles de señal estuvieran trabajando adecuadamente, se han realizado grabaciones con ellos. Antes de utilizar el código fuente para esta tarea, se ha decidido utilizar el programa Audacity para ello. Con él se comprueba que efectivamente estos cuatro primeros micrófonos tienen un comportamiento inadecuado, pudiéndose escuchar solamente ruido en sus grabaciones, mientras que en los cuatro siguientes se pueden escuchar perfectamente las grabaciones realizadas. Esto se puede ver en la figura 4.4, donde se muestra la comparativa de las señales entre los micrófonos estropeados y los que funcionan correctamente.

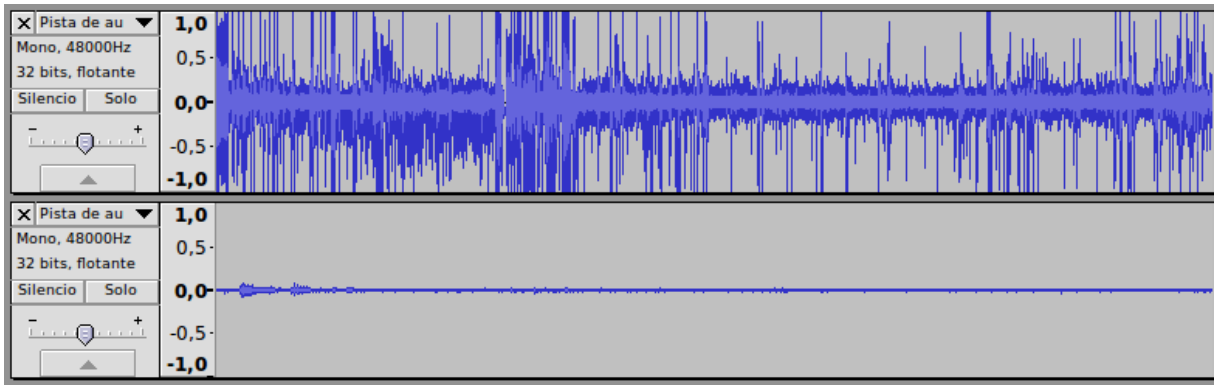


Figura 4.4: Comparación de funcionamiento erróneo con correcto.

Con el fin de localizar el origen de que estos micrófonos no funcionen correctamente, se ha situado uno de los micrófonos que funciona bien a la posición de uno de los que están estropeados. Al hacer esto, el micrófono que funcionaba correctamente ha comenzado a saturar, por lo que se deduce que es el previo de la tarjeta de adquisición el que se encuentra en mal estado. Esto puede deberse a que se detectaron cortos en dos de los micrófonos y hubo que rehacer las soldaduras. Por ello, para descartar este problema,

se ha reemplazado este previo por uno nuevo, y solo se hace uso de los micrófonos Sennheiser situados en el array D, eliminando el array B.

4.2.2 Micrófonos Shure

Al igual que los micrófonos anteriores, se procede a conectar los micrófonos a la tarjeta de adquisición multicanal, y se comprueba con el programa Hammerfall DSP Mixer si estos tienen un nivel de señal adecuado. Como se puede ver en la figura 4.5, dichos micrófonos funcionan correctamente.

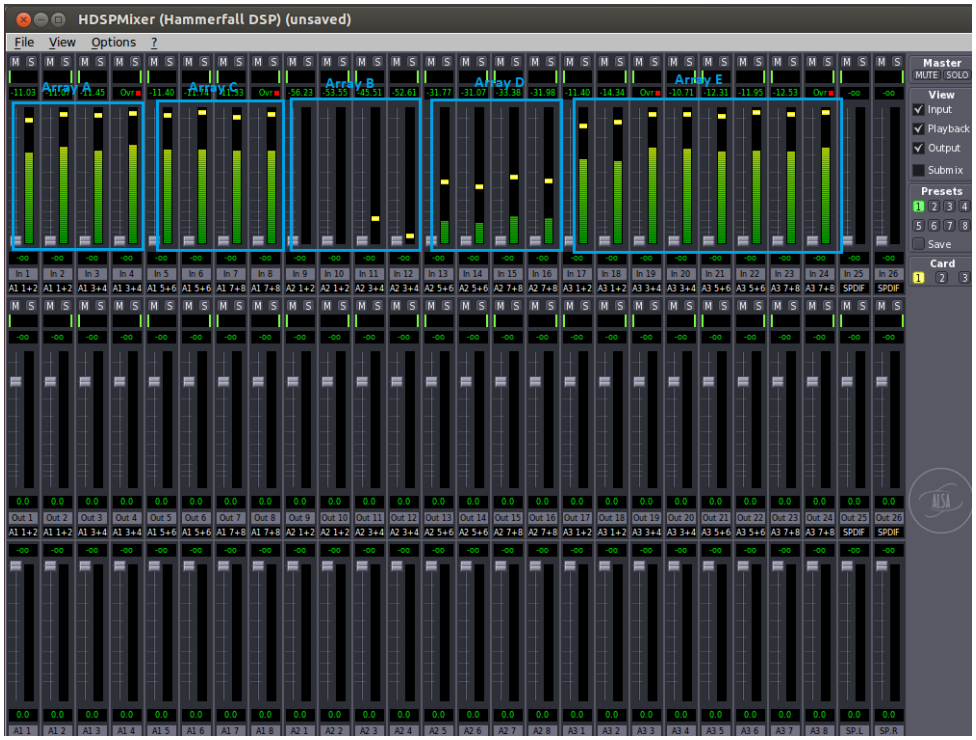


Figura 4.5: Funcionamiento de los nuevos micrófonos en Hammerfall DSP Mixer.

Además de esto, se pasa a realizar una grabación de voz para verificar que los micrófonos graban correctamente con el programa Audacity. Las formas de onda obtenidas se pueden ver en la figura 4.6.

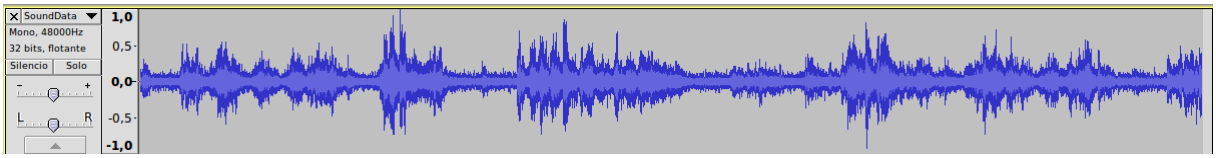


Figura 4.6: Formas de onda nuevos micrófonos Shure.

4.3 Sincronismo del sistema

En primer lugar, para verificar el sincronismo entre los diferentes elementos que forman el sistema, se ha decidido realizar una serie de grabaciones de audio de tonos de diferentes frecuencias con la aplicación del demostrador en tiempo real, ya que con esta forma de onda se podrán determinar de forma muy clara factores como el periodo de la señal, los posibles retardos, etc.

Con las grabaciones obtenidas se pretende comprobar que la adquisición de estas señales sea correcta, comparando la precisión teórica de los retardos con las medidas sobre las señales capturadas. Es importante comprobar esto, ya que, si la adquisición no fuera correcta o si los micrófonos no estuvieran sincronizados correctamente entre ellos, se estarían realizando tanto los cálculos de correlación como los cálculos de la estimación de la localización del hablante, descritos en el capítulo 2, sobre información errónea, lo cual causaría una ubicación del locutor inadecuada.

Para realizar estas pruebas, se han fijado cuatro posiciones a lo largo de la sala ***ispace***, tal y como se muestra en la figura 4.7, de manera que se intentan cubrir todas las posiciones más significativas para poder analizar el funcionamiento de los arrays. Es importante que desde estas posiciones los retardos entre cada par de micrófonos sean lo suficientemente grandes como para poder distinguirlos, pero no tan grandes como para superar el valor de un periodo de señal, ya que sino no se podrían observar dichos retardos adecuadamente.

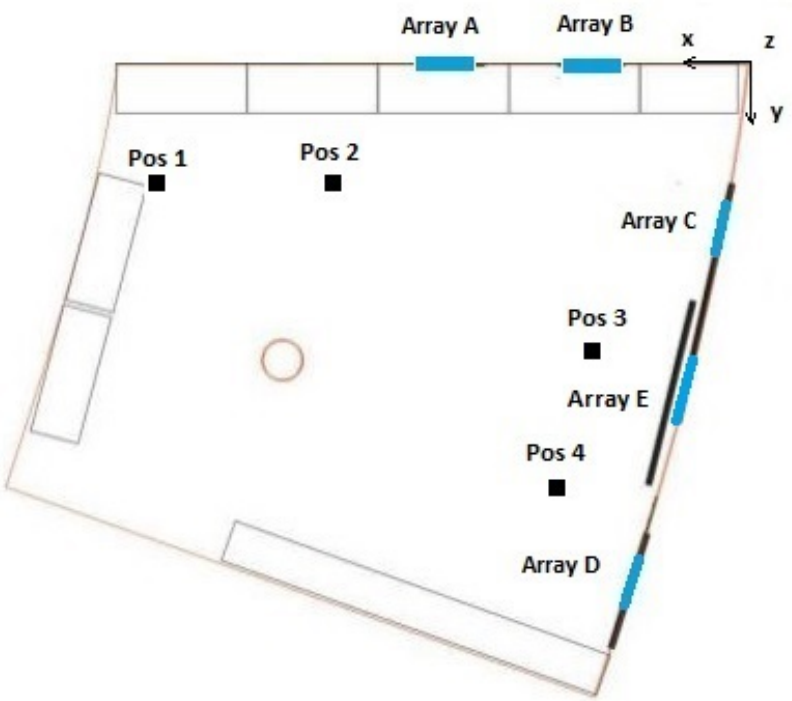


Figura 4.7: Posiciones marcadas en el ***ispace***.

Para realizar las medidas fácilmente se ha hecho uso de un metro láser, y en un primer lugar se hicieron a mano alzada. Esto generó problemas de rigurosidad en estas medidas, ya que, al ser hechas de esta forma, no siempre se tenía situado el metro exactamente en la misma posición, de forma que se ha hecho uso de un trípode. De esta forma, siempre se tendrá el altavoz posicionado exactamente en el mismo punto para todas las grabaciones que se quieran realizar, y también se podrán medir de manera más fiable y exacta las distancias desde estas posiciones a los micrófonos. El trípode y el altavoz utilizados se pueden ver en la figura 4.8.

Figura 4.8: Trípode más altavoz utilizados.

Para asegurar que las medidas sean correctas y lo más fiables posible, se ha hecho uso de dos referencias distintas para realizarlas:

- Medidas directas a los diferentes micrófonos desde cada una de las posiciones marcadas.
- Medidas de la posición relativa en la sala de la posición, y posterior cálculo de las distancias a los micrófonos. Para calcular estas distancias, se ha utilizado lo siguiente: teniendo dos puntos en el espacio tridimensional, tales que $A(x_1, y_1, z_1)$, $B(x_2, y_2, z_2)$, se sabe que la distancia entre ellos seguirá la siguiente ecuación:

$$d_{AB} = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \tag{4.1}$$

Las distancias relativas de cada posición dentro de la sala se pueden ver en la tabla 4.1, donde se utilizan los ejes de coordenadas mostrados en la figura 4.7 de este capítulo. También se puede apreciar en dicha figura la curvatura existente en las paredes, la cual se ha tenido en cuenta a la hora de realizar las medidas.

Tanto las distancias directas a los micrófonos de cada array desde cada una de las posiciones como las medidas relativas en la sala obtenidas se pueden ver en las siguientes tablas, 4.1, 4.2 y 4.3. Para las posiciones relativas dentro de la sala se utilizan los ejes de coordenadas mostrados en la figura 4.7 de este capítulo. También se puede apreciar en dicha figura la curvatura existente en las paredes, la cual se ha tenido en cuenta a la hora de realizar las medidas.

Posiciones	x (m)	y (m)	z (m)
1	6.828	1.344	1.512
2	4.638	1.352	1.515
3	1.889	3.883	1.515
4	2.460	5.437	1.516

Tabla 4.1: Coordenadas de las posiciones de evaluación en la sala.

Distancias al array A				
Posiciones	Mic0 (m)	Mic1 (m)	Mic2 (m)	Mic3 (m)
1	3.821	4.002	4.236	4.128
2	1.851	1.974	2.099	2.147
3	4.154	3.962	3.829	4.053
4	2.780	2.633	2.491	2.729
Distancias al array C				
Posiciones	Mic4 (m)	Mic5 (m)	Mic6 (m)	Mic7 (m)
1	3.821	4.002	4.236	4.128
2	1.851	1.974	2.099	2.147
3	4.154	3.962	3.829	4.053
4	2.780	2.633	2.491	2.729

Tabla 4.2: Distancias directas de las posiciones a los micrófonos arrays A y C.

Distancias al array E								
Posiciones	Mic0	Mic1	Mic2	Mic3	Mic4	Mic5	Mic6	Mic7
3	1.835	1.777	1.706	1.666	1.692	1.756	1.820	1.746
4	2.605	2.473	2.407	2.425	2.528	2.618	2.641	2.523

Tabla 4.3: Distancias directas de las posiciones a los micrófonos array E.

Con el fin de realizar las pruebas, ha sido necesario utilizar unos altavoces amplificadores para reproducir los distintos tonos mencionados, ya que, en las pruebas iniciales con un teléfono móvil, no se llega a conseguir un nivel de amplitud adecuado de la señal grabada, por lo que no se puede analizar correctamente. Aunque también se debe tener en cuenta que la señal no puede estar saturada, porque de lo contrario se distorsionaría introduciéndose armónicos en ella, por lo que no se puede tener un volumen demasiado elevado.

Se requiere además que la emisión sea tipo mono, no estéreo, por lo que se ha configurado en el ordenador en el que se han reproducido los tonos que se emita el audio únicamente por uno de los dos altavoces. Estos tonos se han generado con el programa Audacity, siendo estos de tipo mono, de 32 bits flotante y con frecuencias de: 220, 330, 440, 880, 1000, 2000, 4000, 6000, 8000, 10000, 12000, 14000, 16000, 18000, 20000 Hz, con la intención de abarcar todo el espectro de frecuencias sonoras audibles.

Desde cada una de las cuatro posiciones se ha emitido cada uno de los tonos, y para estas pruebas se ha utilizado una frecuencia de muestreo de 48000 Hz.

Los cálculos que se quieren realizar sobre estas grabaciones son determinar los retardos entre cada par de micrófonos, de manera que se va a poder verificar la correcta sincronización entre los mismos. En primer lugar, se han calculado estos retardos de forma teórica, obteniéndose los resultados mostrados en las siguientes tablas, 4.4, 4.5, 4.6 y 4.7.

Como se puede observar, las mayores diferencias entre las tablas de los retardos obtenidos utilizando las medidas directas y los obtenidos con las medidas relativas, son de 7 muestras de retardo, lo cual equivale a 10 cm. Esto es un error razonable, ya que se están realizando las medidas a mano y algunas paredes tienen curvatura, de forma que se pueden estar cometiendo estos errores.

Retardos al array A						
Posición	Mic0-1	Mic0-2	Mic0-3	Mic1-2	Mic1-3	Mic2-3
1	26	58	43	33	18	15
2	18	35	42	18	25	7
3	8	12	2	4	10	14
4	9	16	2	7	11	18
Retardos al array C						
Posición	Mic4-5	Mic4-6	Mic4-7	Mic5-6	Mic5-7	Mic6-7
1	2	2	4	6	0	6
2	1	1	8	10	2	8
3	20	40	7	24	9	33
4	27	45	14	19	13	31

Tabla 4.4: Retardos entre pares de micrófonos con medidas directas arrays A y C.

Retardos al array E				
Posición	Mic0-3	Mic0-4	Mic1-6	Mic3-4
3	24	20	5	3
Posición	Mic0-2	Mic0-5	Mic0-6	Mic3-5
4	28	1	5	26

Tabla 4.5: Retardos entre pares de micrófonos con medidas directas array E.

Retardos al array A						
Posición	Mic0-1	Mic0-2	Mic0-3	Mic1-2	Mic1-3	Mic2-3
1	26	52	36	26	10	16
2	18	37	35	19	18	1
3	9	18	1	8	9	18
4	5	8	2	4	7	11
Retardos al array C						
Posición	Mic4-5	Mic4-6	Mic4-7	Mic5-6	Mic5-7	Mic6-7
1	2	2	4	6	1	7
2	0	1	8	8	1	7
3	22	44	7	16	21	37
4	25	50	16	10	24	34

Tabla 4.6: Retardos entre pares de micrófonos con medidas relativas arrays A y C.

Retardos al array E				
Posición	Mic0-3	Mic0-4	Mic1-6	Mic3-4
3	29	24	9	5
Posición	Mic0-2	Mic0-5	Mic0-6	Mic3-5
4	29	8	11	30

Tabla 4.7: Retardos entre pares de micrófonos con medidas relativas array E.

Para obtenerlos, se hace uso de la siguiente ecuación. Para ella, se ha utilizado una velocidad del sonido de c = 344.82 m/s (para una temperatura de 22 ºC), y una frecuencia de muestreo de 48000 Hz. Además, las distancias d1 y d2 deben aparecer en metros, y se definen según la figura 4.9. Es conveniente obtener estos retardos en número de muestras, ya que posteriormente se podrán analizar más fácilmente sobre las señales de audio, puesto que es algo más exacto que utilizar otras unidades, como por ejemplo milisegundos.

$$\frac{(d_2 - d_1)}{c} f_s = \text{"}retardo\text{"}[muestras] \tag{4.2}$$

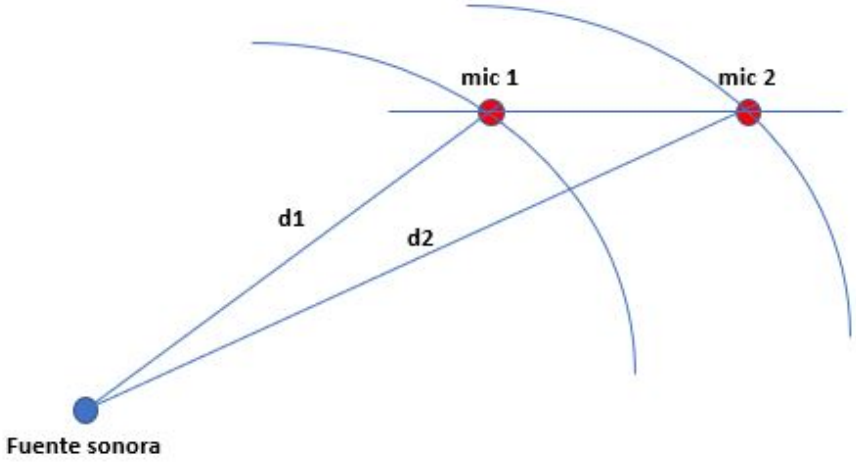


Figura 4.9: Cálculo de retardos.

Con el objetivo de comprobar que realmente estos retardos tienen sentido, se describe a continuación un estudio cuantitativo aproximado de como deberían ser. Para que sea más fácil de entender, se tiene en la figura 4.10 una representación de las distancias directas hacia los arrays A y C.

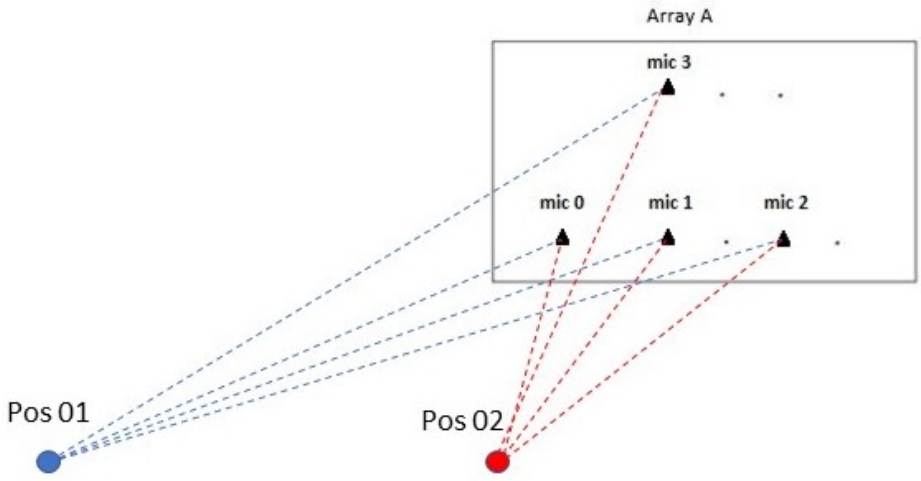


Figura 4.10: Representación retardos array A.

En primer lugar, se observa que desde la posición 1, los retardos en el array A deben entre los micrófonos 0-1 y 1-2 serán parecidos, y entre los micrófonos 0-2 se tendrá aproximadamente el doble que estos. También, que entre los micrófonos 1-3 y 2-3 se tendrán los menores retardos, y que entre los

micrófonos 0-3 se tendrá un mayor retardo. También, desde esta posición, al tener los micrófonos del array C de frente, los retardos entre estos micrófonos serán muy pequeños pequeños.

Asimismo, se observa que desde la posición 2 entre los micrófonos 0-1 y 1-2 se tendrán retardos similares y menores que en el caso de la posición anterior, por estar más cerca al array A. Por su parte, el retardo entre 0-2 será aproximadamente el doble que estos, al igual que antes. Además, se intuye que el retardo entre los micrófonos 2-3 será mínimo, y que entre los micrófonos 0-3 se tendrá el mayor retardo. También se sabe que, al igual que en el caso anterior, los retardos a los micrófonos del array C serán pequeños.

Continuando con las posiciones 3 y 4, los retardos entre los micrófonos del array C seguirán la misma pauta que los retardos entre los micrófonos del array A para las posiciones 2 y 1, respectivamente. Además, los retardos a los micrófonos del array A serán, para los micrófonos 0-1 y 1-2 similares, y que el retardo entre los micrófonos 0-2 debe ser el doble que esto. Además, el retardo entre los micrófonos 0-3 debe ser mínimo.

Por último, para el caso del array E solo se van a considerar un par de retardos máximos y mínimos desde las posiciones 3 y 4, ya que al disponer de ocho micrófonos hay demasiadas combinaciones como para analizarlas todas. De esta forma, como se puede ver en la figura 4.11, los máximos retardos para la posición 3 se tienen entre los micrófonos0-3 y 0-4, y los mínimos para 1-6 y 3-4. Igualmente, para la posición 4, los máximos retardos son para los micrófonos 0-2 y 3-5, y los mínimos para 0-5 y 0-6.

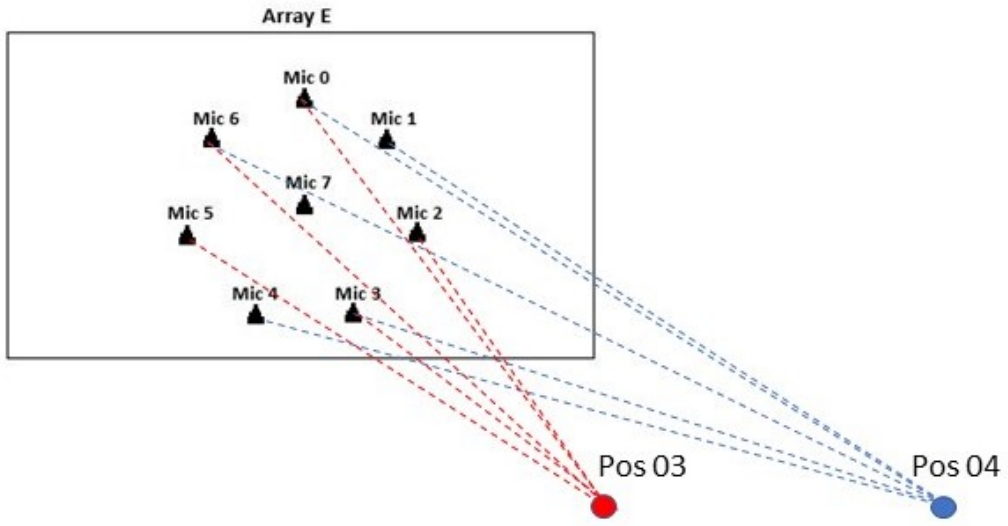


Figura 4.11: Representación retardos array E.

Como se puede comprobar en las tablas 4.4, 4.5, 4.6 y 4.7, en la mayoría de los casos estos retardos concuerdan con los esperados. Hay que tener en cuenta que las mayores diferencias entre los retardos son de unas 7 muestras aproximadamente, lo cual equivale a unos 10 cm, por lo que se consideran unos valores razonables con la realidad, ya que se han podido producir errores en las medidas.

Una vez calculados y comprobados los retardos teóricos, se puede pasar a analizar las formas de onda obtenidas. Para ello se utiliza la herramienta Audacity, ya que permite:

- Ver de forma clara las señales grabadas, permitiendo ampliar y elegir las unidades (como muestras, segundos, etc).

- Mostrar el espectrograma de las señales, de forma que se puede observar qué frecuencias tienen los posibles ruidos que puedan afectar a las señales.
- Recortar varias señales a la vez, de manera que se podrán cortar todas exactamente en el mismo punto, sin afectar a sus retardos.
- Realizar grabaciones de audio multicanal con la tarjeta de adquisición.

Como se puede ver en las tablas que muestran los retardos obtenidos teóricamente en muestras, los mayores valores de retardo entre cada par de micrófonos son de unas 50 muestras, por lo que se deben elegir unas señales con una frecuencia tal que su periodo sea mayor de 50 muestras, ya que, de lo contrario, no se podrán apreciar los retardos correctamente. Las frecuencias que cumplen este requisito son:

- 220 Hz: 218 muestras.
- 330 Hz: 145 muestras.
- 440 Hz: 109 muestras.
- 880 Hz: 55 muestras.
- 1000 Hz: 48 muestras.

En el caso de las grabaciones de los tonos de frecuencias de 220 y 330 Hz, se tiene el problema de que estas presentan mucho ruido, por lo que no se pueden utilizar para analizar los resultados. Esto se debe a que a causa de este ruido se producen varios pasos por cero que no son los deseados. Siguiendo la misma línea, las señales de 440 Hz también son señales ruidosas, aunque en menor medida, pudiéndose observar un fragmento de estas señales en la figura 4.12.

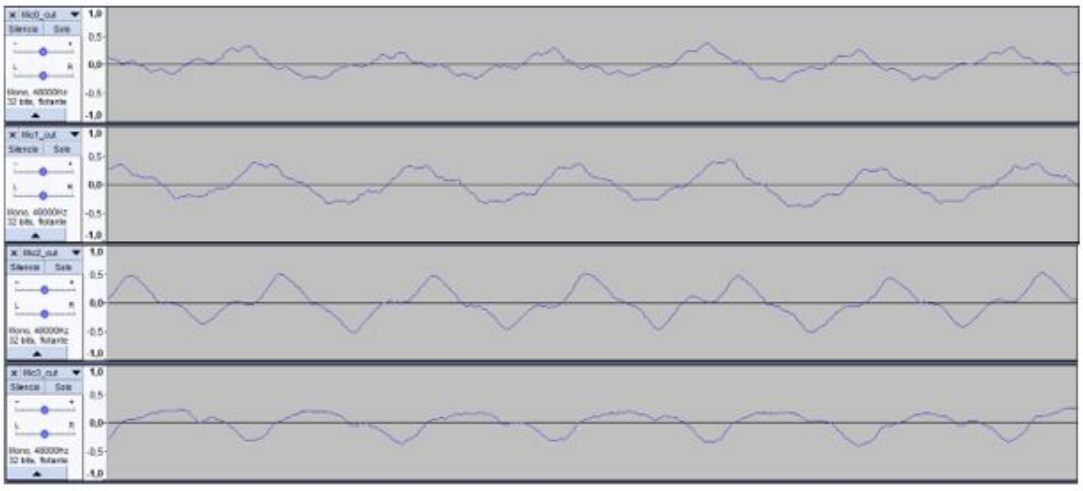


Figura 4.12: Señales con ruido.

Se ha analizado el espectrograma de estas señales para averiguar a qué frecuencias se introducen estos ruidos. Para ello, se debe cambiar en Audacity la vista de forma de onda a espectrograma. Asimismo, para poder analizarlo, es necesario configurar el ancho de banda de frecuencias, siendo en este caso entre 0 y 2000 Hz. También es necesario utilizar un tamaño de ventana grande, ya que de esta manera se tendrá una mayor precisión. Todas las configuraciones realizadas se pueden ver en la figura 4.13.

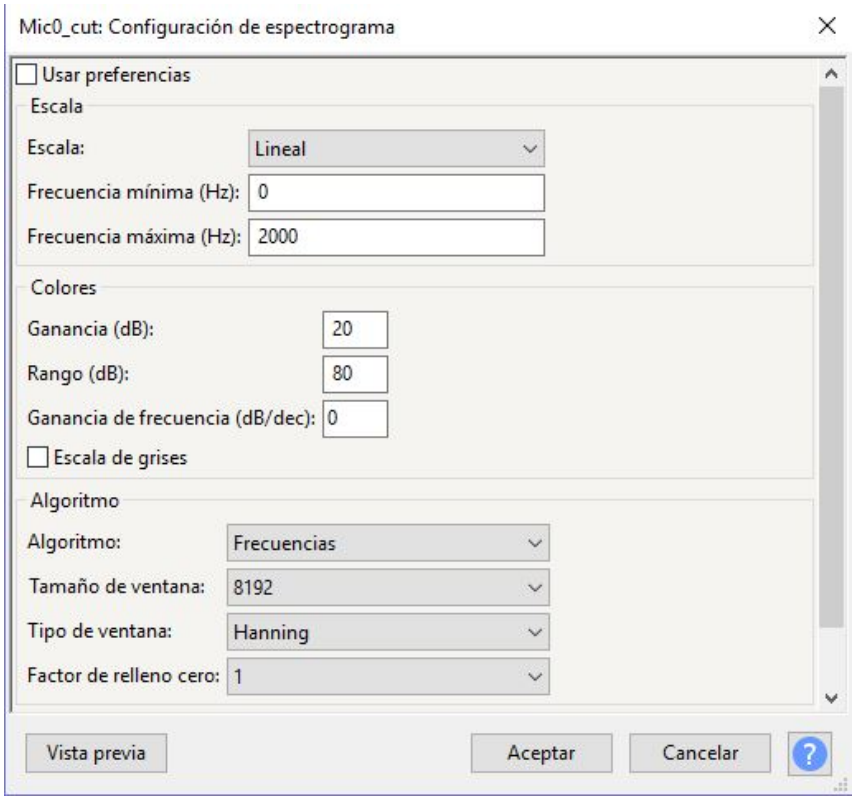


Figura 4.13: Configuración del espectrograma en Audacity.

En la figura 4.14 se puede observar el espectrograma obtenido para esta señal, y como se puede apreciar, a una frecuencia de 440 Hz aparece la señal deseada. De igual modo, a una frecuencia de 1.3 KHz aparece el ruido causante de la deformación de la señal. Puesto que esta última frecuencia es tres veces la de la señal, se justifica como el tercer armónico de la señal.

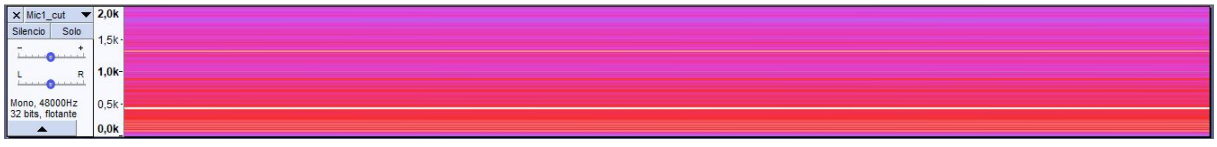


Figura 4.14: Espectrograma de señal de 440 Hz.

Se han obtenido los retardos de las grabaciones de 440 Hz, pero se ha de tener en cuenta que no son muy fiables debido al ruido que presentan. Con el caso de las señales de 880 y 1000 Hz sí se puede trabajar, y son las que se van a utilizar para ver los retardos de dichas señales, puesto que van a ser las que mejor van a mostrar los mismos. Sus formas de onda se pueden ver en la figura 4.15.

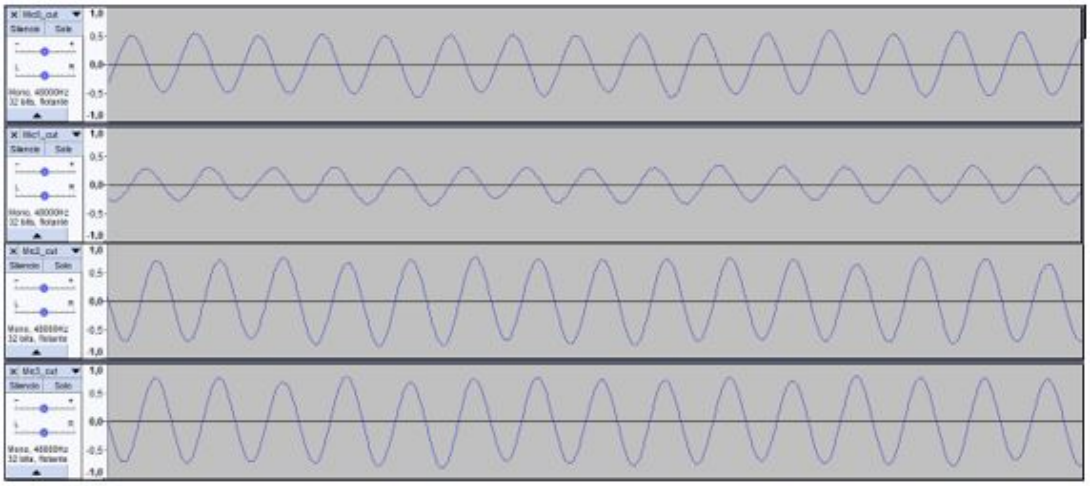


Figura 4.15: Señales sin ruido, frecuencia de 880 Hz.

Puesto que se deben obtener resultados consistentes, es necesario analizar varios periodos de señal, con el objetivo de comprobar que se comportan de la misma forma durante periodos de tiempo largos. Para facilitar esta tarea, se ha creado un programa que detecta los pasos por cero, distinguiendo entre flancos de subida y de bajada, y cuyo funcionamiento y estructura se describe en el apartado 3.4 del capítulo 3.

Antes de analizar los ficheros de audio grabados con dicho programa, será necesario recortar las señales de audio obtenidas hasta donde se ha producido el tono, eliminando la parte de la grabación donde no se encuentra la señal de interés, con el fin de que sea más sencillo analizar los resultados. Todos los audios grabados desde una posición y a una frecuencia concretas deben ser recortados en el mismo punto, de lo contrario los retardos obtenidos no serán válidos.

Los datos de estos ficheros se pasarán a un Excel, donde se va a proceder a realizar los cálculos necesarios. Lo primero será comprobar que el periodo es correcto para cada una de las grabaciones. Para ello se deben restar las muestras consecutivas donde se han obtenido los pasos por cero. Para dar por válidos estos valores de periodo, se calcula la media de todos los valores obtenidos utilizando la función $PROMEDIO()$, y se comprueba que estos no se hayan desviado de forma excesiva con la función $DESVEST()$. Estos resultados se obtienen de forma correcta, tal y como se puede ver a continuación en la tabla 4.8.

Posición	440 Hz	880 Hz	1000 Hz
1	T=109.1447, desv=2.3249	T=54.5454, desv=0.8053	T=48.0022, desv=0.7017
2	T=109.2083, desv=4.5636	T=54.5451, desv=0.7740	T=48.0033, desv=0.9400
3	T=109.0941, desv=1.7481	T=54.5465, desv=1.2480	T=48.0029, desv=1.6018
4	T=109.0846, desv=2.8242	T=54.5439, desv=0.6648	T=47.9990, desv=1.0118
Teórico	109	55	48

Tabla 4.8: Periodos obtenidos.

El siguiente paso es calcular los retardos entre cada par de micrófonos. Para ello, se ha procedido de igual forma que para comprobar el periodo, calculando el promedio y la desviación de los resultados para observar que sean consistentes. Los resultados obtenidos para cada una de las frecuencias en cada posición para los arrays A y C se pueden ver en las siguientes tablas 4.10, 4.11 y 4.12. Para facilitar la comparación, se ha repetido la tabla 4.9, con los retardos teóricos calculados anteriormente.

Retardos al array A						
Posición	Mic0-1	Mic0-2	Mic0-3	Mic1-2	Mic1-3	Mic2-3
1	26	58	43	33	18	15
2	18	35	42	18	25	7
3	8	12	2	4	10	14
4	9	16	2	7	11	18
Retardos al array C						
Posición	Mic4-5	Mic4-6	Mic4-7	Mic5-6	Mic5-7	Mic6-7
1	2	2	4	6	0	6
2	1	1	8	10	2	8
3	20	40	7	24	9	33
4	27	45	14	19	13	31

Tabla 4.9: Retardos esperados para arrays A y C.

Retardos al array A						
Posición	Mic0-1	Mic0-2	Mic0-3	Mic1-2	Mic1-3	Mic2-3
1	28	45	17	17	12	28
2	16	24	62	8	40	37
3	7	20	5	9	12	13
4	8	10	6	7	8	11
Retardos al array C						
Posición	Mic4-5	Mic4-6	Mic4-7	Mic5-6	Mic5-7	Mic6-7
1	2	3	6	8	1	5
2	1	2	5	12	4	7
3	22	45	9	18	21	35
4	25	49	15	17	10	34

Tabla 4.10: Retardos en muestras entre cada par de micrófonos para 440 Hz array A.

Retardos al array A						
Posición	Mic0-1	Mic0-2	Mic0-3	Mic1-2	Mic1-3	Mic2-3
1	27	50	45	23	18	7
2	14	33	33	11	12	1
3	8	19	4	10	11	14
4	9	15	5	5	7	12
Retardos al array C						
Posición	Mic4-5	Mic4-6	Mic4-7	Mic5-6	Mic5-7	Mic6-7
1	4	2	9	5	3	7
2	3	1	9	10	2	9
3	25	44	6	16	20	38
4	22	50	16	10	23	33

Tabla 4.11: Retardos en muestras entre cada par de micrófonos para 880 Hz array A.

Retardos al array A						
Posición	Mic0-1	Mic0-2	Mic0-3	Mic1-2	Mic1-3	Mic2-3
1	23	47	9	24	15	10
2	13	38	33	3	15	18
3	8	17	5	7	9	15
4	7	16	4	7	10	16
Retardos al array C						
Posición	Mic4-5	Mic4-6	Mic4-7	Mic5-6	Mic5-7	Mic6-7
1	2	1	6	11	2	7
2	1	3	8	9	1	6
3	24	39	10	21	11	35
4	26	47	13	15	12	37

Tabla 4.12: Retardos en muestras entre cada par de micrófonos para 1000 Hz array A.

De igual forma, se han obtenido los resultados para el array E, pudiendo ver los resultados esperados en la tabla 4.13, y los obtenidos en las tablas 4.14 y 4.15.

Retardos al array E				
Posición	Mic0-3	Mic0-4	Mic1-6	Mic3-4
3	24	20	5	3
Posición	Mic0-2	Mic0-5	Mic0-6	Mic3-5
4	28	1	5	26

Tabla 4.13: Retardos esperados para array E.

Retardos al array E				
Posición	Mic0-3	Mic0-4	Mic1-6	Mic3-4
3	18	15	10	3
Posición	Mic0-2	Mic0-5	Mic0-6	Mic3-5
4	23	6	4	29

Tabla 4.14: Retardos obtenidos para 880 Hz array E.

Retardos al array E				
Posición	Mic0-3	Mic0-4	Mic1-6	Mic3-4
3	20	23	4	3
Posición	Mic0-2	Mic0-5	Mic0-6	Mic3-5
4	24	4	10	28

Tabla 4.15: Retardos obtenidos para 1000 Hz array E.

En definitiva, los resultados obtenidos para las grabaciones de frecuencia de 440 Hz no siempre son coherentes con los obtenidos teóricamente. Esto es debido a los ruidos introducidos en estas señales, tal y como se ha explicado previamente, por lo que no son resultados concluyentes.

Por otro lado, los resultados obtenidos para las frecuencias de 880 y 1000 Hz sí que son coherentes con los obtenidos de forma teórica en su mayoría, ya que sus valores no se desvían más de 7 muestras, y esto se justifica con los posibles errores cometidos a la hora de realizar las medidas. Con todo esto, se puede concluir que la configuración hardware y el software, además del funcionamiento de la librería de adquisición de audio, tanto para los arrays A y C como para el array E, es correcta.

4.4 Funcionamiento del demostrador en tiempo real con los nuevos micrófonos

Una vez se comprueba que los micrófonos funcionan con Audacity, se pasa a utilizar el demostrador en tiempo real para realizar las grabaciones. Para ello, es necesario indicar en los argumentos de entrada el fichero correcto *.sim* que contenga las superficies de la habitación que se quieran mostrar en el espacio 3D, y especificarle cuántos canales se quieren utilizar, siendo en este caso como máximo veinticuatro canales.

4.4.1 Análisis de la ejecución con un número determinado de micrófonos

En el sistema de partida, a la hora de intentar ejecutar la aplicación con los dieciséis micrófonos se han encontrado problemas, ya que el programa no es capaz de ejecutarse correctamente, apareciendo el mensaje de error *Segmentation fault*. Este tipo de fallos se producen cuando se realizan accesos a zonas de la memoria en las cuales no se autorización. Con el fin de detectar el origen de este fallo, se han realizado diferentes pruebas. En primer lugar, se han ido añadiendo los micrófonos uno por uno. Con esta prueba se ha comprobado que con un número de micrófonos de ocho, nueve, diez u once el sistema es capaz de funcionar adecuadamente, pero a la hora de incorporar el doceavo micrófono se comprueba que ciertas veces el sistema es capaz de ejecutarse, pero otras no. Además, con un número de canales superior a doce, nunca es capaz de ejecutarse.

Por otro lado se ha intentado depurar el programa para detectar el fallo que pudiera encontrarse en el código, pero cada vez se produce en una parte distinta del mismo. En consecuencia, al no ser un fallo consistente, ya que cada vez que se ejecuta falla en un punto del código distinto, o incluso ciertas veces funciona correctamente, nos lleva a pensar que el fallo puede tener su origen en el manejo de la memoria.

En primer lugar, se ha pasado a verificar qué memoria tiene disponible el ordenador del *ispace* utilizado para llevar a cabo estas ejecuciones. Esta se puede visualizar con el comando *gtop*, y el resultado se puede ver en la figura 4.16, observándose que se tienen disponibles en total 4 GBs de memoria.

	total	used	free	shared	buffers	cached
Mem:	3953	2209	1743	0	339	870
-/+ buffers/cache:		1000	2953			
Swap:	7632	0	7632			
Total:	11586	2209	9376			

Figura 4.16: Memoria disponible en el PC del *ispace*.

En segundo lugar, con el comando htop se ha comprobado el uso que hace el programa de la memoria cada vez que este se lanza. Nada más arrancarlo, el programa ocupa la memoria mostrada en la figura 4.17. El problema viene cuando el programa lleva un tiempo ejecutándose, ya que, a medida que pasa el tiempo, este va ocupando cada vez una cantidad mayor de memoria, tal y como se puede ver en el figura 4.18. Esto supone un problema, puesto que cuando lleguen a consumirse los 4 GBs disponibles, el programa empezará a hacer uso de la memoria *Swap*. Llegados a este punto, el ordenador se ralentiza, ya que al ser esta una memoria virtual, todos los accesos que se hagan a ella van a ser mucho más lentos.

Todo ello indica un problema de *memory leak* que hay que solucionar, y aunque el *Segmentation fault* no sea por esto, si que puede contribuir a que suceda.

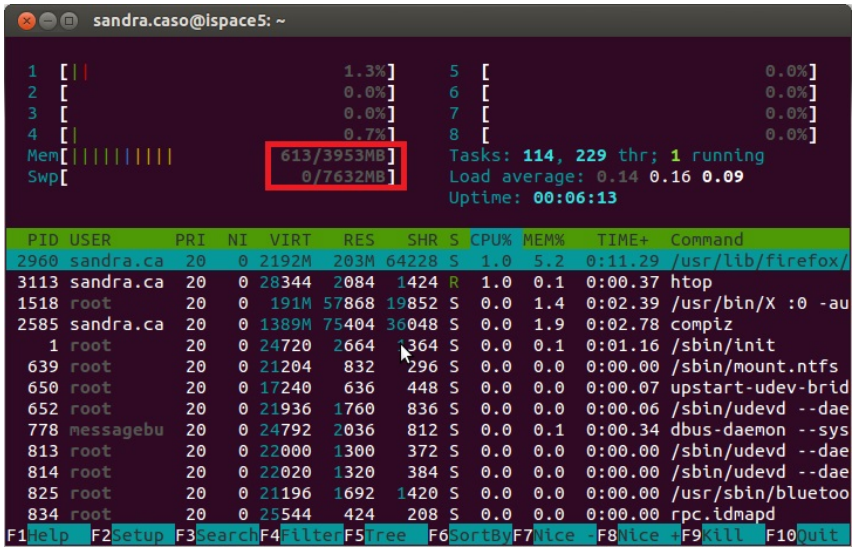


Figura 4.17: Memoria utilizada al iniciar el programa.

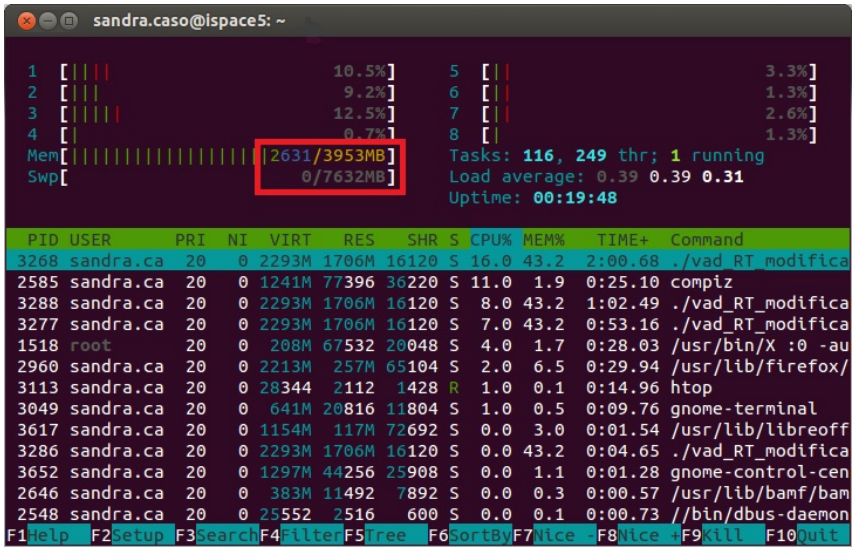


Figura 4.18: Memoria utilizada al cabo de un tiempo de ejecución.

Este comportamiento nos lleva a utilizar el programa Valgrind, el cual permite la depuración de problemas de memoria, así como realizar pruebas de rendimiento de programas. Un ejemplo de los tipos de problemas que permite detectar son:

- Lectura o escritura de posiciones de memoria fuera de los límites de la memoria reservada.
- Lectura o escritura de posiciones de memoria que habían sido previamente liberadas.
- Uso de memoria no inicializada.

Para ejecutarlo, simplemente se debe añadir el comando *valgrind* delante de la ejecución normal del programa. Los resultados de este programa se han guardado en un fichero de extensión *.log*, con el fin de poder analizarlos más cómodamente. Los fallos encontrados son:

- *Conditional jump or move depends on uninitialised value(s)*: este fallo define que se están utilizando variables sin inicializar previamente. Se comprueba en todos los casos que no es crítico, ya que sí que se están inicializando de forma previa a su utilización.
- *Invalid read of size 1*: se define este fallo ya que se intenta leer una posición de memoria que ya se había liberado anteriormente. Este fallo no es crítico, ya que solo se intenta hacer una lectura, no una escritura, lo cual sería un problema.
- *Invalid write of size 8*: este fallo sí es crítico, ya que el programa trata de escribir en una posición de memoria donde no está reservada. A la hora de analizar este fallo, se encuentra que en la librería de adquisición de audio se tienen varios *mallocs* que reservan una memoria insuficiente a la necesaria.

Una vez arreglado dicho problema, se pasa a ejecutar de nuevo el programa Valgrind. De esta forma, ya no se tienen los errores, por lo que se pasa a lanzar el programa con los dieciséis micrófonos, funcionando este correctamente, de forma que se consiguen grabar todos los canales.

4.4.2 Uso de 20 canales

Una vez comprobado el funcionamiento con dieciséis micrófonos, se pasa a añadir ocho micrófonos más, los correspondientes al array E. Se comprueba como el programa se ejecuta correctamente con todos ellos.

Para mejorar el funcionamiento del sistema, se ha considerado cambiar el valor fijado del umbral de detección de voz, ya que se observa que se representa la posición calculada del hablante en momentos de silencio, por lo que se sospecha que se tiene un umbral demasiado bajo. Para saber con qué valor se debe ajustar dicho umbral, se han estudiado los valores de CSP máximos obtenidos durante tramos de voz, para lo cual, se han guardado en un fichero *.txt* a la hora de ejecutarlo. Estos valores de silencio están en torno a valores de CSP entre 0.026 y 0.033.

El valor del umbral se encontraba fijado con un valor de 0.028, por lo que, como se puede ver en los resultados, es un valor poco ajustado a la realidad, y se ha modificado a un valor de 0.034. Con este nuevo valor, se corrigen las representaciones erróneas del hablante durante periodos de tiempo de silencio.

4.5 Evaluación perceptual del demostrador

Por último, la prueba que queda por realizar es observar el funcionamiento del sistema completo con todos los canales añadidos, y comprobar de qué forma influyen en el comportamiento de este. Para ello, se ha decidido, en un primer lugar, comprobar el funcionamiento del sistema con cada uno de los arrays por separado, y finalmente, el funcionamiento global con todos.

En primer lugar, se tratará solamente con el array A. Para ello, se debe indicar a la aplicación que solo se hará uso de cuatro canales. Una vez hecho esto, se coloca el altavoz en la posición 1 de la sala, ya que es la posición más restrictiva, se emite un tono de 880 Hz y se observa el comportamiento. La mayor parte del tiempo se obtiene una ubicación correcta, pero en varias ocasiones el resultado es erróneo.

A continuación, se pasa a hacer uso solo del array C, para lo cual se deben indicar ocho canales a la hora de ejecutar la aplicación, y bajar al mínimo la ganancia de los cuatro micrófonos correspondientes al array A en la tarjeta de adquisición. De esta manera, solo se considerarán los cuatro micrófonos del array C. Al igual que antes, se emite un tono de 880 Hz, pero esta vez desde la posición 3 de la sala, ya que es la más restrictiva para este array, y se obtienen unos resultados muy similares que los obtenidos con el array A.

Siguiendo con el mismo procedimiento, se selecciona solo el array D, indicando doce canales y situando las ganancias de los ocho micrófonos que no proceden al mínimo, de forma que solo se consideren para realizar los cálculos los cuatro micrófonos del array D. Para este caso es necesario modificar el umbral fijado, ya que los valores de CSP en los micrófonos Sennheiser son mucho menores que el umbral de 0.034 seleccionado, estando en torno a valores de 0.008. Una vez modificado, al igual que en los casos anteriores se emite un tono de 880 Hz desde la posición 3 de la sala y se observa el comportamiento, obteniéndose peores resultados que para los arrays anteriores.

Por último, se pasa a comprobar el array E. Para ello, se seleccionan veinte canales, y al igual que en los casos anteriores se posicionan al mínimo las ganancias de todos los micrófonos que no sean los ocho que forman este array, y se emite un tono de 880 Hz, esta vez desde la posición 4 de la sala. Con este array se obtienen los mejores resultados, teniendo la mayoría del tiempo resultados correctos, aunque en ciertas ocasiones falla.

Una vez probados todos los arrays por separado, se ejecuta la aplicación del demostrador en tiempo real con todos ellos, y, situando el altavoz en el medio de la sala *ispace*, se reproduce un tono de 880 Hz. Los resultados que se obtienen son muy parecidos a los que se consiguen con los arrays A o C, siendo un poco mejores. Por lo tanto, será necesario realizar en un futuro pruebas sobre el sistema ejecutándolo de forma offline, con el fin de comparar los resultados que se obtengan con los obtenidos en la ejecución en tiempo real.

4.6 Conclusiones

En este capítulo se han expuesto los diferentes experimentos realizados para evaluar el correcto funcionamiento del demostrador de localización de fuentes de audio en tiempo real que se ha implementado.

Para ello, en un primer lugar, se ha verificado el funcionamiento de los micrófonos implementados en este proyecto, con el fin de demostrar que trabajan correctamente. En el caso de los micrófonos del modelo Shure, los ocho micrófonos funcionan bien, pero en el caso de los del modelo Sennheiser, cuatro de ellos no funcionan, escuchándose solo ruido en las grabaciones realizadas con ellos. Queda como líneas futuras comprobar si el problema se encuentra en que estos micrófonos están estropeados o si viene dado porque el previo se encuentre en malas condiciones.

Posterior a esto, se han realizado diferentes grabaciones de audio de diferentes tonos en distintas posiciones de la sala para poder verificar así el correcto sincronismo de todos los procesos del sistema. Los resultados obtenidos de estas pruebas son correctos, ya que los que no lo son están justificados a causa del ruido que se introduce en las señales. Por lo tanto, se llega a la conclusión de que el sincronismo es correcto, y por tanto, las configuraciones del hardware y el funcionamiento de la librería de adquisición de audio también. De esta forma, se descarta que los fallos de localización sean debido a esto.

Además de esto, se han descubierto problemas en la memoria gracias a la ejecución del programa Valgrind, consiguiendo resolverlos todos. Estos errores impedían la ejecución del programa con más de doce micrófonos, haciendo imposible trabajar con todos los canales disponibles.

Por último, se ha realizado una evaluación perceptual del sistema, donde se concluye que los mejores resultados se obtienen para el array E. Asimismo, se debe realizar en un futuro pruebas exhaustivas del comportamiento del sistema trabajando en modo offline, y comparar estos resultados con los obtenidos en tiempo real.

Capítulo 5

Conclusiones y líneas futuras

5.1 Introducción

A continuación, se expondrán las conclusiones más relevantes que se han podido extraer de la realización del presente proyecto. Asimismo, también se expondrán los resultados obtenidos en la evaluación del sistema.

Además se mostrarán líneas de trabajo futuras que han surgido en la consecución de las fases del proyecto y dan valor a aquellas tareas de interés que se consideran importantes utilizando el presente trabajo.

5.2 Conclusiones

Este trabajo se ha centrado en la evaluación, diseño e implementación de un sistema demostrador en tiempo real de captura y procesamiento de audio multicanal para localización de hablantes mediante agrupaciones de micrófonos dentro de espacios inteligentes. Para ello se ha estudiado el funcionamiento de cada uno de los módulos que han formado parte de él, y se han realizado diversas pruebas con el objetivo de encontrar posibles fallos en el código que empeorasen el funcionamiento del sistema para poder así corregirlos.

En la implementación del sistema se ha completado el despliegue hardware disponible, añadiendo elementos y finalizando el conexionado completo de los que ya se encontraban en el Grupo de Investigación, con el fin de disponer de un entorno amplio de experimentación futura y desarrollar aplicaciones de demostración en tiempo real. De esta forma, se ha completado la instalación de dos nuevos arrays de cuatro micrófonos cada uno, y se ha construido desde cero uno adicional compuesto por ocho micrófonos. En el proceso sólo se han podido poner en funcionamiento cuatro de los cinco arrays disponibles, lo que se traduce en que sean utilizables 20 de los 24 micrófonos disponibles, quedando como trabajos futuros la comprobación del funcionamiento de los cuatro que se consideran estropeados, ya que se tienen dudas de si el elemento de fallo es el previo microfónico o los micrófonos en sí.

Además, en la parte de evaluación del sistema se ha podido comprobar a través de una serie de grabaciones de tonos que el sincronismo en el sistema es correcto, y, por lo tanto que todas las configuraciones y conexiones de la parte hardware son correctas. Esto es importante, ya que de no haber una correcta sincronización en el sistema, se estaría realizando los cálculos de correlación y de estimación de la posición de la fuente sonora sobre datos erróneos.

Asimismo, con la evaluación del funcionamiento de la aplicación con más de doce micrófonos, se ha descubierto que había fallos importantes en el código disponible, debido a problemas de reserva insuficiente de memoria, de forma que se realizaban accesos a memoria restringida, y el consiguiente fallo de segmentación. Al arreglar este error, ha sido posible utilizar la totalidad de los canales disponibles y mejorar el funcionamiento perceptual de la aplicación de demostración.

Por último, con las pruebas de evaluación perceptual del demostrador, se verifica que se mejora su funcionamiento con todos estos cambios, siendo necesario ajustar el umbral para evitar errores en la detección de la posición del hablante.

Tras la realización de este proyecto se han obtenido las siguientes conclusiones:

- Se ha diseñado y realizado la implementación de tres nuevos arrays, añadiendo un total de dieciséis micrófonos al sistema y completando todo el conexionado del sistema.
- Se ha diseñado e implementado una nueva topología de array heptagonal, y se ha analizado su estructura, teniendo como objetivo posterior su comparación con la topología en T de la que ya se disponía.
- Se ha verificado el correcto sincronismo entre los elementos hardware que componen el sistema, de forma que se asegura que las configuraciones y conexiones realizadas sobre el hardware de captura de audio multicanal son correctas.
- Se ha verificado el correcto funcionamiento de las librerías de soporte a la captura de audio multicanal
- Se han solucionado problemas encontrados con el programa Valgrind sobre el manejo de la memoria, lo cual impedía utilizar más de doce canales, pudiéndose añadir ahora los veinticuatro disponibles.
- Se ha generado una documentación completa del despliegue hardware del sistema, fundamental para futuros trabajos en demostradores de la algorítmica de procesamiento de audio multicanal en el grupo.

5.3 Líneas futuras

En este apartado se proponen algunas de las líneas futuras de investigación para mejorar el trabajo expuesto en este proyecto. Las futuras mejoras están enfocadas tanto a la implementación como a la evaluación del sistema y generación de nuevos recursos y propuestas algorítmicas:

- Diseñar y realizar el despliegue de hardware de captura multicanal en espacios temporales, que requieren que el hardware sea trasladable con facilidad, para lo que ya se dispone de los elementos necesarios.
- Evaluar las prestaciones y capacidades de procesamiento en tiempo real, y las limitaciones en función de la configuración del sistema.
- Verificar los problemas del array de micrófonos que no se ha logrado hacer funcionar, determinando si es debido a un problema en el previo en los micrófonos en sí.
- Realizar pruebas exhaustivas sobre el comportamiento de la aplicación con una ejecución offline, y comparar los resultados obtenidos con los resultantes de la ejecución en tiempo real.

- Realizar la implementación de un módulo de actualización del umbral de decisión en el sistema de detección de actividad de voz y lo haga flexible y robusto ante posibles cambios en el entorno acústico, pero evitando que sea vulnerable y propenso a cambios bruscos en el nivel de umbral que se aplicará en el autómata.
- Proponer y evaluar algoritmos de detección de voz alternativos a la máquina de estados implementada por el autómata del módulo VAD y realizar una comparativa para obtener el rendimiento de cada uno de ellos.
- Proponer y evaluar algoritmos de localiación alternativos a los disponibles en la actualidad, llevando a su implementación práctica todos los desarrollos del grupo en esta temática.
- Realizar una adquisición de una base de datos que dé soporte a los trabajos de investigación en procesamiento de audio multicanal dentro del grupo.

Bibliografía

[1] M. P. González, "Diseño, implementación y evaluación de un demostrador de localización de fuentes de audio en tiempo real," Master's thesis, Escuela Politécnica Superior. Universidad de Alcalá. Spain, 2015.

[2] J. P. M. González, "Diseño, implementación de un sistema de adquisición, procesamiento y generación de audio multicanal en aplicaciones para espacios inteligentes," Master's thesis, Escuela Politécnica Superior. Universidad de Alcalá. Spain, 2014.

[3] R. P. Nuño, "Estudio, implementación y evaluación de un sistema de detección de actividad de voz en espacios inteligentes," Master's thesis, Escuela Politécnica Superior. Universidad de Alcalá. Spain, 2014.

[4] D. C. Pérez, "Diseño, implementación y evaluación de una interfaz de control multimodal en un espacio inteligente: control gestual," Master's thesis, Escuela Politécnica Superior. Universidad de Alcalá. Spain, 2013.

[5] C. Castro, "Speaker localization techniques in reverberant acoustic environments," Master's thesis, School of Electrical Engineering. Royal Institute of Technology (KTH). Sweden, 2007.

[6] "Características rme octamic ii," https://www.thomann.de/pics/bdb/119911/5862132_800.jpg.

[7] "Características rme hammerfall dsp 9652," https://www.thomann.de/es/rme_digi_9652_hdsp.

[8] "Diagrama patrones cardioide y supercardioide," https://esblogshure.wordpress.com/2016/11/25/cual-es-la-diferencia-entre-el-beta-87a-y-beta-87c/.

[9] "Diagrama patrón omnidireccional," http://www.rockcamp.es/blog/los-cacharros-del-sonido-ii-microfonos-2/.

[10] "Información sobre patrones polares en micrófonos," http://www.earpro.es/noticias/microfonos-con-multiples-patrones-polares-que-donde-y-como/.

[11] "Micrófono shure mx391," http://www.shure.es/productos/microflex/mx391_mx392_mx393 [Último acceso 6/septiembre2017].

[12] "Características micrófonos sennheiser," https://assets.sennheiser.com/global-downloads/file/5712/MKE2_Gold_Manual_07_2015_EN.pdf.

[13] C. H. Knapp and G. C. Carter, "The generalized correlation me- thod for estimation of time delay." *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-24, no. 4, pp. 320–327, August 1976.

[14] "Página sobre opengl," https://es.wikipedia.org/wiki/OpenGL [Último acceso 5/Septiembre/2017].

[15] "Manual de usuario de rme octamic ii," https://www.rme-audio.de/download/octamic2_e.pdf.

[16] "Manual de usuario de rme pci hdsp 9652," https://www.rme-audio.de/download/hdsp9652_e.pdf.

[17] "Información sobre gnu/linux en wikipedia," http://es.wikipedia.org/wiki/GNU/Linux [Último acceso 6/diciembre/2017].

[18] L. Lamport, *LaTeX: A Document Preparation System, 2nd edition.* Addison Wesley Professional, 1994.

[19] "Página de la aplicación cvs," http://savannah.nongnu.org/projects/cvs/ [Último acceso 6/diciembre/2017].

[20] "Página de la aplicación gcc," http://savannah.gnu.org/projects/gcc/ [Último acceso 6/diciembre/2017].

[21] "Página de la aplicación make," http://savannah.gnu.org/projects/make/ [Último acceso 6/diciembre/2017].

Apéndice A

Herramientas y recursos

Las herramientas que han sido necesarias para desarrollar este proyecto son las siguientes:

- PC compatible
- Sistemas de captura de audio multicanal de alta calidad disponibles en el Grupo:
  - Micrófonos de cabeza de alta calidad del modelo Shure (con sistema inalámbrico).
  - Micrófonos electret de alta calidad del modelo Sennheiser series MKE 2-P-C y MKE 2-5-C, siendo necesarios un total de ocho.
  - Micrófonos electret de alta calidad del modelo Shure series MX391/O, siendo necesarios un total de dieciséis.
  - Preamplificadores de 8 canales de RME con conversor A/D incorporado (RME Octamic).
  - Tarjeta de adquisición multicanal PCI RME Hammerfall DSP 9652, siendo necesarias un total de tres.
- Herramientas para la implementación de los nuevos arrays:
  - Soldador.
  - Cable de audio apantallado.
  - Recubrimiento termorretráctil.
  - Resistencias y condensadores para la implementación de los filtros.
  - Tablero para la implementación del array E.
- Sistema operativo GNU/Linux [17].
- Entorno de desarrollo Eclipse Luna/ Emacs.
- Procesador de textos LaTeX[18].
- Control de versiones CVS [19].
- Compilador C/C++ gcc [20].
- Gestor de compilaciones make [21].

Apéndice B

Especificaciones

A continuación se van a explicar las especificaciones necesarias para poder replicar el hardware del sistema. El esquema del mismo se puede ver a continuación en la figura B.1.

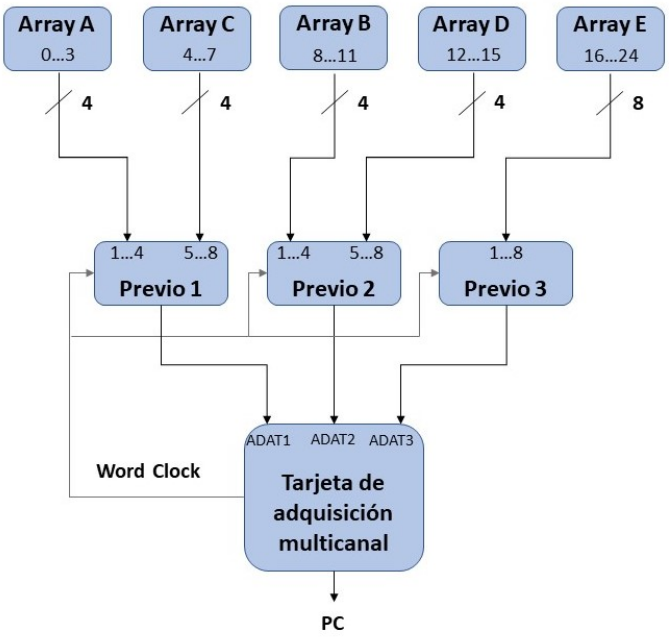


Figura B.1: Esquema global de la parte hardware.

Los elementos que conforman el sistema son:

- Micrófonos: se utilizan dos modelos distintos, dieciséis del modelo Shure MX391/O y ocho del modelo Sennheiser MKE 2-P-C y MKE 2-5-C. Estos se sitúan en las agrupaciones de micrófonos, y se conectan a los diferentes previos. Se sigue el orden mostrado en la figura B.1.
- Previos: se utilizan tres previos del modelo RME OctaMic II, con ocho canales cada uno, y se encargan de preamplificar y digitalizar la señal de los micrófonos. Estos deben tener activada la alimentación *phantom*, tener el control de ganancia al máximo y se debe configurar para que la señal de reloj venga de una fuente externa, en este caso de la tarjeta de adquisición. La salida de cada previo debe conectarse a la tarjeta de adquisicición multicanal.

- Tarjeta de adquisición multicanal: se usa el modelo PCI RME HDSP 9652, y se encarga de muestrear la señal procedente de los previos y de generar la señal de sincronismo. Esta debe conectarse al PC en el que se ejecute la aplicación.

Una vez se tiene implementada la parte hardware, se puede ejecutar la aplicción del demostrador de captura de audio en tiempo real, y para ello se deben indicar los siguientes parámetros, entre otros, pudiéndose ver un ejemplo de ejecución en la figura B.2.

- Número de canales.
- Frecuencia de muestreo.
- Número de elementos del buffer circular.

```
./vad_RT_modificado -g 0.12 -J 0.2 -U 3 -B 1 -f 48000 -n 0.05 -s 0.05 -p 8192 -w m -x 1 -r 3
-j 0 -k 0 -z 2 -u 0 -q 0 -a 0 -L 0 -H 8000 -b 150 --freq-srp 0 -R 1 -N 0 -X 0 -Z 0.05 -F 0 -E
0 -G 24 -W 0 -v ../../far-field/environments/IspaceRoom/ispace-onlyArrayA+C.sim --channels all
--numchannels 24 --audiodevid 0 --offset 0 --nitems 20000
```

Figura B.2: Ejemplo ejecución de la aplicación.

Universidad de Alcalá
Escuela Politécnica Superior

ESCUELA POLITECNICA
SUPERIOR

Universidad
de Alcalá