

GRADO EN INGENIERÍA ELECTRÓNICA INDUSTRIAL Y
AUTOMÁTICA

TRABAJO DE FIN DE GRADO

**DISEÑO DE NODO INALÁMBRICO PARA
PROCESAMIENTO DE VOZ EN
PREVENCIÓN DE VIOLENCIA SEXUAL**

Autor: Elena Hernando Fernández

Tutora: Celia López Ongil

Fecha: 18-06-2018

AGRADECIMIENTOS

En primer lugar, quisiera agradecer toda la ayuda y apoyo a mi tutora, Celia, que siempre ha estado disponible cuando he necesitado resolver dudas o cuando no sabía por donde encaminar ciertas cuestiones del trabajo, pero sobre todo por depositar su confianza en mí y proponerme formar parte de un proyecto tan grande e interesante. También me gustaría agradecer a Manuel y Jose el tiempo que me han dedicado, los ánimos y todo lo que he aprendido en estos meses gracias a ellos.

Por último, pero no por ello menos importante, agradecer a mis padres todo su esfuerzo en brindarme la oportunidad de estar aquí, terminando la carrera que deseaba estudiar, por obligarme a no rendirme en los momentos más difíciles y por celebrar todos mis logros.

RESUMEN

Este Trabajo Fin de Grado surge de la necesidad de reducir, pero sobre todo prevenir, el alto índice de violencia y agresiones sexuales contra las mujeres que existe en cualquier región del mundo. Para ello, se diseña un sistema de grabación de audio en modo continuo (*streaming*) y de envío mediante un protocolo inalámbrico, *Bluetooth® Low Energy* (BLE), a un teléfono móvil; lo que permite incorporar la detección de pánico en la voz y poder alertar, a través de un dispositivo portable, de un ataque o agresión que se esté produciendo contra la usuaria.

Por otra parte, se ha aprovechado el sistema de adquisición de audio realizado, para diseñar y desarrollar un sistema que optimice y depure las grabaciones realizadas, en una cámara anecoica, para el entrenamiento de los algoritmos que detectan estrés en la voz. Se han definido dos protocolos para la comunicación en dicho sistema, uno inalámbrico mediante *Bluetooth® Low Energy* (BLE), y otro alámbrico mediante un cable USB de 10 metros de alcance.

La realización de este Trabajo Fin de Grado ha tenido una duración de 10 meses, y de forma resumida y muy general, se puede sostener que se ha conseguido unificar el trabajo en los niveles altos de abstracción, generación de algoritmos para el procesamiento de audio, con los más bajos, y cercanos a las capas físicas y de enlace, para la adquisición, preprocesado y transmisión de dicho audio, a través de los componentes electrónicos y los protocolos de comunicación más actuales y demandados en las aplicaciones tecnológicas de redes de sensores (*MEMS, Bluetooth® Low Energy..*).

ABSTRACT

This Final Degree Project arises from the need to reduce, but above all prevent, the high rate of violence and sexual aggression against women that exists in any region of the world. For this reason, a streaming audio recording system is designed and sent via Bluetooth® Low Energy (BLE) to a mobile phone, which allows the incorporation of panic detection in the voice to alert through a portable device, of an attack or aggression that is occurring against the user.

On the other hand, the audio acquisition system has been used to design and develop a system that optimizes and debugs the recordings made in an anechoic chamber for the training of the algorithms that detect voice stress. Two protocols have been defined for communication in this system, one wireless via Bluetooth® Low Energy (BLE), and another wired via a 10-meter USB cable.

The achievement of this Final Degree Project has lasted 10 months, and in a summarized and very general way, we can say that it has been possible to unify the work in the high levels of abstraction, generation of algorithms for the audio processing, with the lowest and closest ones to the physical and connection layers, for the acquisition, preprocessing and transmission of this one, through the most current electronic components and communication protocols demanded in the technological applications of sensor networks (MEMS, Bluetooth® Low Energy ..).

ÍNDICE DE FIGURAS

Figura 1. Mapa que muestra la incidencia de la violencia sobre las mujeres por parte de una pareja íntima por regiones de la OMS.	1
Figura 2. Incidencia de violencia contra la mujer. [4]	2
Figura 3. Red de sensores inalámbricos. [8].....	7
Figura 4. Estructura hardware de un nodo sensor de una WSN. [8].....	7
Figura 5. Diagrama de sectores del uso de dispositivos wearable según la parte del cuerpo. [17]	11
Figura 6. Diagrama de sectores de los dispositivos wearable según su aplicación. [17]	12
Figura 7. Mapa de uso de los wearable en el trabajo y en la vida personal.	13
Figura 8. Esquema del sistema desarrollado para la adquisición y procesamiento del audio....	17
Figura 9. Modulación por código de pulsos (PCM).	19
Figura 10. Explosionado de un micrófono electret.	19
Figura 11. Sección de un micrófono electret.	20
Figura 12. Circuito equivalente de un micrófono electret.	20
Figura 13. Variación de la capacidad de un micrófono MEMS con la amplitud de la onda sonora. [21]	21
Figura 14. Sensibilidad frente temperatura MEMS vs. Electret. [21]	23
Figura 15. Respuesta en frecuencia para diferentes tipos de ECM'S. [21]	23
Figura 16. Respuesta en frecuencia de un conjunto de diferentes MEMS. [21].....	24
Figura 17. Diagrama de bloques de la arquitectura del colgante.	25
Figura 18. Diagrama de bloques típico de un micrófono MEMS con interfaz I ² S. [22]	25
Figura 19. Diagrama de bloque típico de un micrófono MEMS con interfaz PDM. [22].....	26
Figura 20. Modulación por densidad de pulsos (PDM).	26
Figura 21. Etapa de adquisición del audio.	28
Figura 22. Placa de evaluación nRF52 DK.....	29
Figura 23. Asignación de pines de la placa de evaluación nRF52 DK.	30
Figura 24. Módulo PDM. [25]	31
Figura 25. Cronograma de una señal sinusoidal analógica tras convertir a formato PCM y a formato PDM.....	31
Figura 26. Configuración en modo stereo. [25]	32
Figura 27. Configuración en modo mono y micrófono left. [25].....	32
Figura 28. Configuración en modo mono y micrófono right. [25]	32
Figura 29. Flujograma del proceso de adquisición de audio.....	35
Figura 30. Etapa de compresión del audio.....	37
Figura 31. Calidad frente a bitrate de diferentes códecs.	38
Figura 32. Flujograma del proceso de adquisición, compresión y envío del audio.	41
Figura 33. Etapa de envío del audio.	42
Figura 34. Etapa de descompresión del audio.	44
Figura 35. Interfaz de la app utilizada para el envío del audio por Bluetooth®.....	45
Figura 36. Proceso de descompresión mediante su ejecutable a través de la línea de comandos de Windows.....	47
Figura 37. Reproducción de los audios a través de VLC.....	47
Figura 38. Seno a 440 Hz grabado con el micrófono.....	48
Figura 39. Audio de una canción tras descomprimirlo.	49
Figura 40. Audio 'Microphone_02_04_2018_04_21_41' del set 2 de la base de datos.....	51

Figura 41. Cámara anecoica del Departamento de Teoría de la Señal y Comunicaciones.	52
Figura 42. Espectro de frecuencias del ruido de fondo del laboratorio vs. Espectro de frecuencias del ruido de fondo de la cámara.....	52
Figura 43. Sistema cableado para realizar grabaciones en la cámara anecoica.	53
Figura 44. Flujograma de la adquisición del audio mediante protocolo alámbrico.....	54

ÍNDICE DE TABLAS

Tabla 1. Breve descripción de las aplicaciones de algunas redes inalámbricas actuales. [6]	6
Tabla 2. Funciones de los drivers del micrófono.....	34
Tabla 3. Cálculos para estimar el tamaño de los buffers en función de la memoria del microcontrolador.	36
Tabla 4. Parámetros de las funciones opus_encode y opus_decode.	40
Tabla 5. Parámetros de la función ble_nus_string_send. [25].....	43
Tabla 6. Cálculos para el número de paquetes del envío.	44
Tabla 7. Funciones de MATLAB para crear y reproducir un archivo de audio en formato WAV.49	
Tabla 8. Parámetros de la función SEGGER_RTT_printf.....	55

LISTA DE ACRÓNIMOS

PDM	<i>Pulse Density Modulation</i>
TSC	Teoría de la Señal y Comunicaciones
IoT	<i>Internet of Things</i>
AES	<i>Advanced Encryption Standard</i>
WSN	<i>Wireless Sensor Networks</i>
MEMS	<i>Microelectromechanical Systems</i>
WASN	<i>Wearable Acoustic Sensor Networks</i>
SoC	<i>System-on-Chip</i>
CI	Circuito integrado
GPS	<i>Global Positioning System</i>
IDC	<i>International Data Corporation</i>
LOPD	Ley Orgánica de Protección de Datos
PCM	<i>Pulse Code Modulation</i>
SNR	<i>Signal-to-Noise Ratio</i>
ECM	<i>Electret Condenser microphone</i>
JFET	<i>Junction Field-Effect Transistor</i>
ADC	<i>Analog-to-Digital Converter</i>
PCB	<i>Printed Circuit Board</i>
I2S	<i>Integrated Interchip Sound</i>
PWM	<i>Pulse Width Modulation</i>
ARM	<i>Advanced RISC Machine</i>
BLE	<i>Bluetooth Low Energy</i>
SDK	<i>Software Development Kit</i>
CPU	<i>Central Processing Unit</i>
ROM	<i>Read Only Memory</i>
RAM	<i>Random Acces memory</i>
DAC	<i>Digital-to-Analog Converter</i>
UART	<i>Universal Asynchronous Receiver-Transmitter</i>
SPI	<i>Serial Peripheral Interface</i>
RS232	<i>Recommended Standard 232</i>
FFT	<i>Fast Transform Fourier</i>
MP3	<i>MPEG-1 Audio Layer III</i>
WMA	<i>Windows Media Audio</i>
VoIP	<i>Voice over IP</i>
IETF	<i>Internet Engineering Task Force</i>
iOS	<i>iPhone OS</i>
PC	<i>Personal Computer</i>
API	<i>Application Programming Interface</i>
GATT	<i>Generic Attribute</i>
ASCII	<i>American Standard Code for Information Interchange</i>
USB	<i>Universal Serial Bus</i>
RTT	<i>Real Time Transfer</i>
SWD	<i>Serial Wire Debug</i>

ÍNDICE GENERAL

AGRADECIMIENTOS.....	iii
RESUMEN.....	v
ABSTRACT.....	vi
ÍNDICE DE FIGURAS.....	vii
ÍNDICE DE TABLAS.....	ix
LISTA DE ACRÓNIMOS.....	xi
1. INTRODUCCIÓN.....	1
1.1. MOTIVACIÓN DEL TRABAJO.....	1
1.2. OBJETIVOS.....	3
1.3. ESTRUCTURA DEL DOCUMENTO.....	4
2. ESTADO DE LA TÉCNICA.....	5
2.1. ENTORNO SOCIOECONÓMICO.....	12
2.2. NORMATIVA.....	14
3. DESCRIPCIÓN DEL SISTEMA.....	17
3.1. CONCEPTOS PREVIOS.....	18
3.1.1. <i>AUDIO</i>	18
3.1.2. <i>MICRÓFONO</i>	19
3.1.3. <i>FORMATO</i>	25
3.2. DRIVERS DEL MICRÓFONO.....	27
3.2.1. <i>MICROCONTROLADOR</i>	28
3.2.2. <i>FUNCIONES</i>	33
3.2.3. <i>FLUJOGRAMA</i>	34
3.3. COMPRESIÓN/DESCOMPRESIÓN.....	37
4. PUESTA EN MARCHA.....	41
4.1. BLUETOOTH.....	42
4.2. RECEPCIÓN DE AUDIO (MÓVIL Y PC).....	44
4.3. MATLAB.....	48
5. GRABACIONES.....	51
5.1. CÁMARA ANECOICA.....	51
5.2. PROTOCOLO INALÁMBRICO VS. ALÁMBRICO.....	53
6. CONCLUSIONES.....	57
7. PRESUPUESTO.....	59
7.1. SISTEMA PARA EL PROCESAMIENTO DEL AUDIO.....	59
7.2. BINDI.....	60
8. BIBLIOGRAFÍA.....	63

1. INTRODUCCIÓN

1.1. MOTIVACIÓN DEL TRABAJO

Actualmente los casos de agresiones sexuales o violencia de género están a la orden del día y desafortunadamente las cifras de mujeres agredidas o maltratadas cada año no disminuyen. Según la Organización Mundial de la Salud, una de cada tres mujeres de todo el mundo, experimentará violencia física y/o sexual por parte de su pareja o de alguien que no lo sea [1]. Esta incidencia está prácticamente repartida de igual manera en cualquier región del mundo. Figura 1.

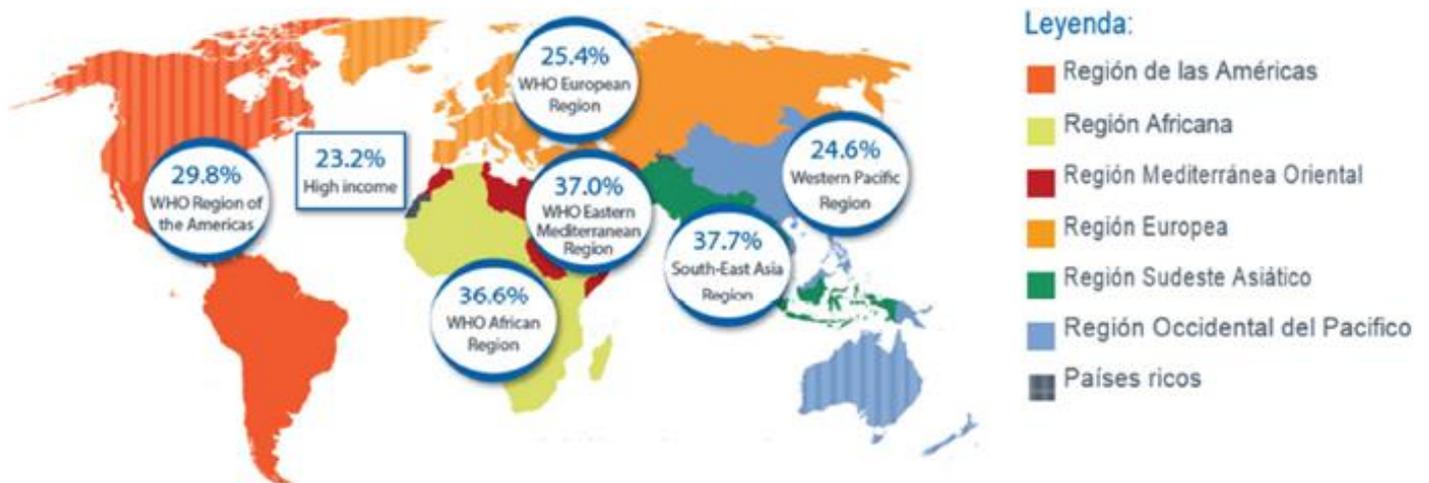


Figura 1. Mapa que muestra la incidencia de la violencia sobre las mujeres por parte de una pareja íntima por regiones de la OMS.¹

Además, según los datos recogidos por el Instituto de la Mujer, en España el 9'6 % de las mujeres han sufrido alguna vez a lo largo de su vida maltrato² [2] y en EE.UU una de cada cinco mujeres en campus universitarios ha sido asaltada [3]. En la Figura 2, se muestra la incidencia de violencia sexual contra mujeres de 15 a 69 años por parte de parejas conyugales y/o de alguien que no lo sea, durante el año 2010. Como ingenieros nos planteamos la forma en que se puede ayudar a reducir esta terrible situación. En este proyecto se ha intentado avanzar en este sentido, defendiendo que si se emplean los avances tecnológicos en el campo de la electrónica en proporcionar un derecho humano fundamental como es la seguridad, podemos contribuir a mejorar la situación de temor que sufren muchas mujeres hoy en día.

Este proyecto se crea con el fin de facilitar la integración de técnicas de reconocimiento de voz en dispositivos portátiles (*wearables*³), para que sean de utilidad a la hora de alertar lo antes posible en casos de violencia mientras se estén produciendo y para alegar pruebas que pudiesen utilizarse ante la justicia. Para ello, se plantea la realización de un sistema que permita la adquisición y el procesamiento de audio durante las grabaciones, en una cámara anecoica, necesarias para entrenar los algoritmos encargados de detectar el pánico en la voz, así como los métodos y técnicas necesarias para su integración final en estos dispositivos. Todo ello surge a raíz de la competición "Anu & Naveen Jain Women's Safety XPRIZE" en la

¹ <http://www.who.int/reproductivehealth/publications/violence/9789241564625/en/>

² Estimación realizada a partir de Instituto de la Mujer (2006): "Porcentaje de mujeres maltratadas según comunidad autónoma".

³ Se entiende como *wearable* un dispositivo que puede llevar puesto el usuario.

cual participa la universidad a través de la propuesta de UC3M4Safety, grupo liderado por Celia López Ongil y que incluye la participación de los departamentos de Tecnología Electrónica, de Ingeniería Telemática, de Teoría de la Señal y Comunicaciones, de Ingeniería Mecánica y el Instituto de Estudio de Género de la Universidad Carlos III de Madrid, así como el Centro de Electrónica Industrial de la Universidad Politécnica de Madrid. El objetivo de la competición es presentar una solución tecnológica no visible que alerte en caso de detección de violencia o agresión sexual para garantizar la seguridad de mujeres de una amplia variedad de clases sociales, por lo que sus requisitos principales son: bajo coste, elevada precisión y rapidez de respuesta en cualquier zona del mundo.

UC3M4Safety propone mediante *Bindi*, dos dispositivos *wearables*, un colgante y una pulsera, ambos provistos de diferentes vías para la activación de la alarma que indica que se está produciendo una situación de violencia o un ataque. Las diferentes vías que producen la activación de dicha alarma son, en el colgante: un botón de pánico y un micrófono que graba el audio para la detección de estrés en la voz; y en la pulsera: un botón de pánico y un conjunto de sensores biométricos (pulso cardíaco, temperatura y conductividad), que también permiten la detección de emociones de miedo o pánico. Una vez que se produce la activación de la alarma por una de estas vías o por varias, se envía al teléfono móvil de la usuaria, que lo recibe en la *App* que gestiona la red creada por el móvil, la pulsera y el colgante y que a su vez manda la señal de alarma a un círculo de ayuda, ya sea a la policía o a un conjunto de contactos de confianza seleccionados por la propia usuaria.

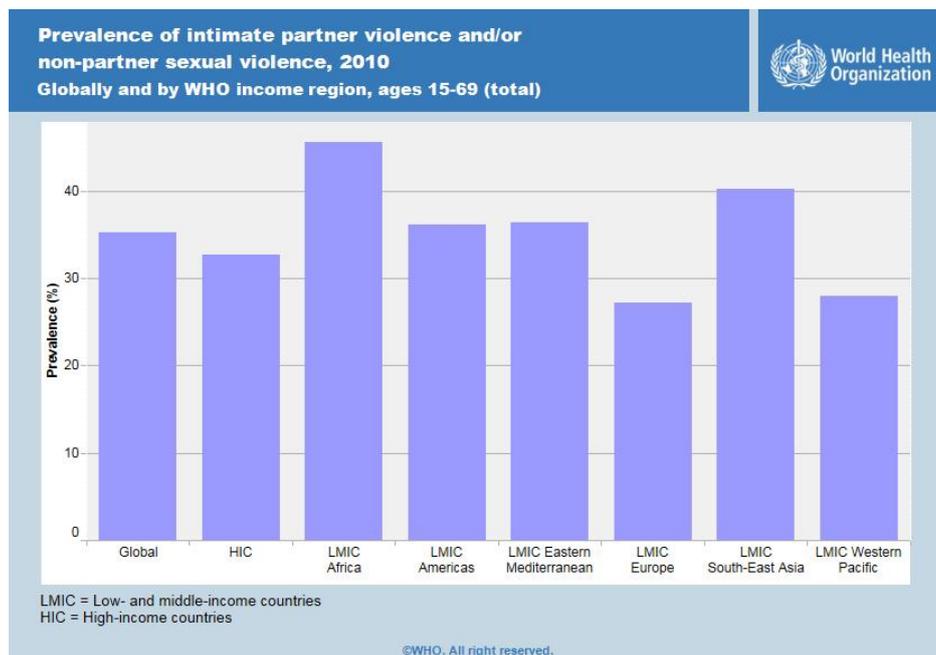


Figura 2. Incidencia de violencia contra la mujer. [4]

1.2. OBJETIVOS

El objetivo de este proyecto es doble:

1. Proporcionar los métodos/técnicas para que se pueda adquirir audio en modo continuo (*streaming*) en un dispositivo portable, *wearable*, denominado *Bindi* (esto valdría para el entrenamiento, pero también para la operación final). Por consiguiente, se consigue tener un dispositivo *wearable* mucho más completo e innovador, capaz de alertar de una situación de violencia a través de dos vías, el botón de pánico y el audio grabado por el micrófono. Esto resulta muy útil en el caso de que la víctima esté inmovilizada y no le sea posible activar la alarma a través del botón de pánico.
2. Proporcionar un sistema para el entrenamiento eficiente de los algoritmos, mediante grabaciones automáticas en la cámara anecoica, y para la optimización del procesamiento del audio.

Si bien el equipo UC3M4Safety está compuesto por 34 personas de 6 grupos de investigación diferentes, cada una de las cuales ha trabajado en el diseño y desarrollo de las distintas partes que componen la solución *Bindi*, en este Trabajo Fin de Grado se presenta el diseño y desarrollo de toda la funcionalidad relacionada con la captura y envío de señales de audio desde uno de los nodos de la solución, bien de forma inalámbrica o bien de forma cableada.

En resumen, este proyecto beneficia a *Bindi* en tanto que permite optimizar la adquisición y el procesamiento del audio durante las numerosas grabaciones, pero necesarias, para entrenar adecuadamente los algoritmos que se deseen incluir en la detección del pánico. Gracias a ello, UC3M4Safety podría plantear la introducción de la detección no solo de estrés en la voz, o de silencios, sino también de palabras clave que determinen que están teniendo lugar agresiones verbales o que permitan identificar y distinguir la voz de la usuaria. Además, siempre se puede tener en mente el almacenamiento de forma temporal del audio recogido, pudiendo presentarse como prueba judicial.

Para poder lograr esto, se han ido siguiendo una serie de pasos a lo largo de la realización de este proyecto, que coinciden con la planificación y ejecución del proyecto:

- Entender cómo funciona la adquisición de audio mediante la interfaz PDM (2 semanas).
- Seleccionar el micrófono y diseñar la interfaz con un periférico de microprocesador que extraiga el audio en un formato eficiente para su posterior procesamiento (1 semana).
- Familiarizarse con el entorno de desarrollo y el microprocesador con el que se va a gestionar el periférico que se conecta al micrófono (4 semanas).
- Programación, depuración y pruebas con el micrófono para asegurarse de que está grabando adecuadamente (6 semanas).
- Realización de algunas grabaciones de prueba en la cámara anecoica para contrastarlas con las llevadas a cabo en el laboratorio con mayor nivel de ruido (3 semanas).
- Realización de los programas *software* que permiten extraer audio del micrófono (*drivers*) (8 semanas).

- Grabaciones, en la cámara anecoica, de la base de datos, para que posteriormente el personal de Teoría de la Señal y Comunicaciones entrene los algoritmos de detección de estrés en la voz (4 semanas).
- Procesamiento del audio grabado: compresión y descompresión (5 semanas).
- Optimización del proceso de grabación en la cámara anecoica para automatizar y agilizar pruebas (4 semanas).
- Redacción de la memoria del Trabajo Fin de Grado (4 semanas).

1.3. ESTRUCTURA DEL DOCUMENTO

La estructura que sigue el presente documento se especifica a continuación:

- **Capítulo 1.** En este capítulo se presenta el tema del proyecto y se expone lo que ha motivado a llevarlo a cabo.
- **Capítulo 2.** Planteamiento del estado de la técnica.
- **Capítulo 3.** Descripción de la arquitectura del sistema.
- **Capítulo 4.** Detalle de la implementación del sistema y las pruebas realizadas.
- **Capítulo 5.** Finalmente, se exponen las conclusiones del proyecto y las líneas futuras a seguir.

2. ESTADO DE LA TÉCNICA

Actualmente, cada vez más personas y objetos están conectados entre sí a través de medios principalmente inalámbricos, formando una red. La idea de tener todo tipo de objetos y personas conectadas en red de forma inalámbrica promete facilitarnos una mejor calidad de vida. La sociedad se dirige hacia el patrón de estar comunicada en todo momento. Es aquí donde surge el concepto de '*Internet of Things*' (*IoT*), el cual hace referencia a la identificación e interconexión de objetos cotidianos a través de *Internet*⁴. En un futuro muy cercano se prevé la implementación de redes inalámbricas inteligentes, autónomas y conscientes de ello. Cuando las personas están conectadas en red, se estimula la economía del conocimiento, se adquieren técnicas e información que ayudan en la búsqueda de trabajo, emprender negocios, cuidar la salud; así como proporcionar accesibilidad a la educación, servicios financieros y, en general, se favorece el desempeño de un papel más participativo en la sociedad. Además, una comunicación inalámbrica, proporciona mayor flexibilidad y se caracteriza por ser multidireccional, sin los problemas de ubicuidad que puede suponer cablear físicamente para conectar puntos. [5]

El espectro electromagnético adquiere gran importancia en las redes inalámbricas de comunicaciones. Esto se debe a que la información, bien sea voz, video o datos, cuando se envía desde un emisor hasta un receptor, se propaga en el espacio libre a través de ondas electromagnéticas (señales), las cuales se clasifican en diferentes bandas según su frecuencia y constituyen el espectro electromagnético. A pesar de que el espectro electromagnético es muy extenso, en telecomunicaciones únicamente se utiliza una pequeña parte de él, puesto que las limitaciones que imponen los sistemas de comunicación, la dimensión de las antenas y las características de propagación de las señales, hacen que este constituya un recurso finito y muy valioso, por lo que debe regularse su uso. [5]

Actualmente, existe una gran variedad de tecnologías de transmisión inalámbrica de datos, como por ejemplo *Wi-Fi*, *Bluetooth*, *ZigBee*, *2G/3G/4G*, etc (Tabla 1). Además, hay que añadir a estos sistemas, nuevas redes en desarrollo como por ejemplo *Thread*, una alternativa en el área de la domótica, o tecnologías que utilizan la "banda blanca" cedida por la televisión digital terrestre para dar cabida a soluciones *IoT* en multitud de áreas. En función de la aplicación a que se destine una tecnología inalámbrica concreta, los factores como el alcance, velocidad de transferencia, seguridad, potencia y autonomía determinarán la mejor opción a la hora de elegir entre una red inalámbrica u otra. [6]

⁴ <http://www.wearablespain.com/Wearable.html>

ZigBee	Bajo consumo en sistemas complejos, seguridad superior, robustez, alta escalabilidad y capacidad para soportar un gran número de nodos. Utilizado en domótica y redes de sensores en aplicaciones <i>IoT</i> .
Wi-Fi	Adecuado en la transferencia de archivos, pero consume demasiada potencia para desarrollar aplicaciones <i>IoT</i> .
Bluetooth	Utilizado en la transmisión de datos de corto alcance. Es clave para desarrollar dispositivos <i>wearables</i> , ya que permite el establecimiento de conexiones <i>IoT</i> a través de <i>smartphones</i> o dispositivos móviles.
Bluetooth Low Energy (BLE)	<i>Bluetooth</i> [®] de baja energía. Ofrece un alcance similar al de la tecnología <i>Bluetooth</i> [®] normal pero con un consumo de energía significativamente reducido. No está diseñado para transferir archivos, es más adecuado para fragmentos de datos. Posee un alto grado de integración en <i>smartphones</i> y dispositivos móviles.

Tabla 1. Breve descripción de las aplicaciones de algunas redes inalámbricas actuales. [6]

En cuanto a la seguridad de las redes inalámbricas, en general sí son vulnerables a ataques. El atacante es capaz de capturar paquetes de datos y de control. Para evitar estos ataques, se emplean técnicas de cifrado tales como *Advanced Encryption Standard (AES)*, un algoritmo de cifrado simétrico declarado como suficientemente seguro para su uso en información clasificada como alto secreto. Este algoritmo permite claves de cifrado de 128, 192 y 256 bits. En las redes inalámbricas se requiere asegurar al menos la protección de la integridad del sistema, la confidencialidad y la privacidad del usuario. [7]

Por otra parte, cabe destacar respecto a las tecnologías inalámbricas el auge que están alcanzando las denominadas redes de sensores inalámbricos (WSN). Los sensores WSN pequeños, robustos, económicos y de baja potencia, distribuyen el *IoT* a objetos instalados en cualquier clase de entorno [8]. Figura 3.

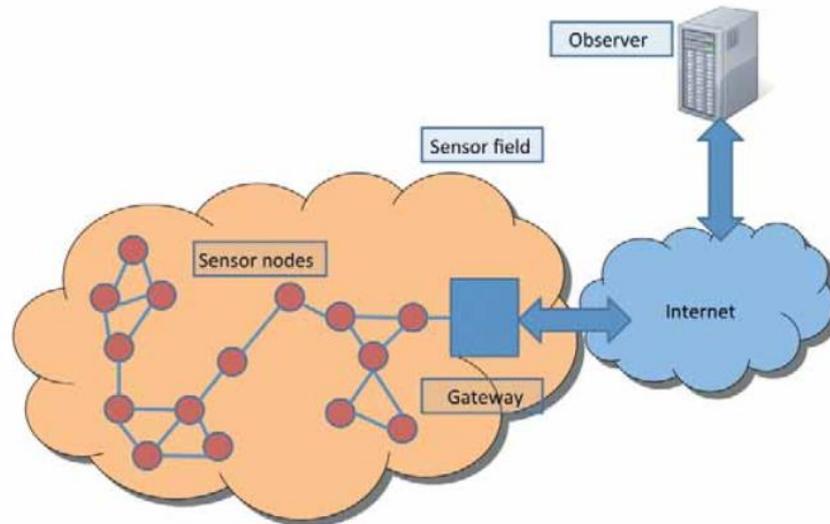


Figura 3. Red de sensores inalámbricos. [8]

Una red de sensores inalámbrica constituye un conjunto de pequeños dispositivos autónomos distribuidos físicamente, denominados nodos de sensores e instalados en torno al área de actuación de un fenómeno (condiciones físicas o ambientales) para ser monitorizado, con la capacidad de almacenar y comunicar datos de forma inalámbrica en una red. Actualmente se crean WSN para numerosas aplicaciones, algunas de ellas: cuidado de la salud, servicios básicos y monitorización remota de fenómenos climatológicos, ambientales, etc.. En el cuidado de la salud, los dispositivos inalámbricos facilitan la monitorización de pacientes, haciendo que este sea menos invasivo. Para servicios básicos como, por ejemplo, la distribución de electricidad, ofrecen un método coste reducido para sistemas de recogida de datos que contribuyan a reducir el consumo de energía y a mejorar el manejo de recursos. La monitorización remota abarca una gran variedad de aplicaciones, en las que la implementación de un sistema inalámbrico destaca por la reducción de costes y una mayor sencillez frente a soluciones cableadas, permitiendo aplicaciones novedosas de medición (detección de incendios, control de sequía, detección de plagas, etc.). En resumen, se incluyen en numerosas tecnologías (biología, medicina, domótica, etc.), aportan interacción de los seres humanos con el medio ambiente y suponen un menor uso de recursos.

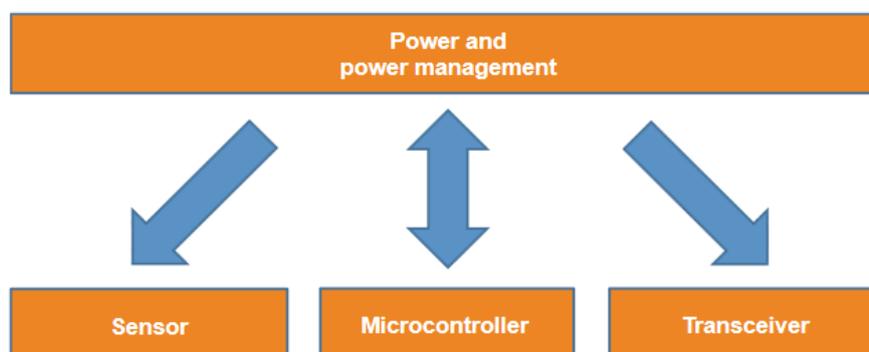


Figura 4. Estructura hardware de un nodo sensor de una WSN. [8]

Un nodo WSN está constituido por diferentes bloques electrónicos (microcontrolador, sistema de adquisición de señales, sistema de actuación y sistema de comunicación inalámbrica) autónomos, alimentados por baterías (Figura 4). Ya que se trata de nodos portátiles, autónomos y probablemente desasistidos, es de suma importancia conocer y optimizar la distribución de su energía disponible. En sistemas alimentados con batería, tener altas tasas de datos y el uso frecuente de radio suponen un mayor consumo de energía, el cual constituye un factor clave en su diseño. Como consecuencia, se debe tener en cuenta el tamaño y el peso de las baterías. El coste reducido y fácil acceso a baterías de carbón zinc o baterías alcalinas o de litio, las convierten en una opción atractiva y rentable [9]. Para prolongar la vida útil de las baterías, se diseña el nodo WSN de forma que: se encienda y transmita datos; y posteriormente, si no tiene operaciones que llevar a cabo, se apague para conservar energía. De esta forma se aumenta la eficiencia de la transmisión, permitiendo al sistema regresar al modo *sleep* con un uso mínimo de energía. Esto implica que el procesador involucrado deba ser capaz de despertar, encenderse y volver al estado *sleep* de manera eficiente. La construcción de estos nodos sensores con los cuales se conforma la red ha sido posible debido a los avances en el área de sistemas micro electromecánicos (MEMS). Los MEMS son dispositivos miniatura fabricados con capacidad de sensado, comunicación y procesamiento. Cada uno de los nodos, está compuesto por tres subsistemas: [10]

- Sensor: mide un parámetro (sensado) del medio ambiente.
- Procesado: realiza el procesamiento de la información obtenida mediante el sensado.
- Comunicación: es el subsistema encargado del intercambio de información con los nodos sensores del resto de la red.

El consumo de energía se caracteriza por lo el gasto referido a los sensores, la comunicación y el procesado. El mayor consumo de energía, tiene lugar en la transmisión de datos, siendo de menor importancia en el procesado y puesta en marcha de los sensores.

Las redes de micrófonos inalámbricos o también conocidas como redes inalámbricas de sensores acústicos (WASN), constituyen la tecnología de próxima generación para la adquisición y el procesamiento de audio. Por el contrario a las matrices de micrófonos tradicionales, las cuales muestrean de forma local un campo de sonido, WASN permite utilizar un mayor número de micrófonos para abarcar un área de mayor interés. Por otra parte, el diseño de dichas WASN resulta una tarea complicada, particularmente para la adquisición de audio en tiempo real y debido al tráfico significativo de datos en la red. Algunos ejemplos de aplicaciones que se benefician de WASN son: [11]

- *Audífonos*. A medida que los micrófonos se convierten en componentes más pequeños y baratos, gracias a la tecnología MEMS, evoluciona su incorporación de forma discreta en la ropa, los muebles, etc.
- *Telefonía manos libres*. El uso de nodos WASN, proporciona una mejora considerable en el grabado del habla, puesto que constituyen una tecnología habilitadora para la comunicación de voz en entornos ruidosos y dinámicos, tales como aeropuertos, fábricas, etc.
- *Monitorización acústica de un entorno*. Un ejemplo de ello puede ser, el rastreo de vehículos o los sistemas de vigilancia.

- “*Inteligencia ambiental*”, término que hace referencia al conjunto de sensores y procesadores que se conectan entre sí incorporándose al entorno de forma discreta, dando lugar a un entorno inteligente capaz de detectar la presencia de un usuario y establecer una comunicación activa con él. Esta comunicación puede consistir en el reconocimiento automático de voz, que empleará técnicas de clasificación del audio, localización e identificación del usuario. Estas redes de sensores acústicos también pueden instalarse en domicilios de personas ancianas, para controlar su patrones de comportamiento y así detectar tempranamente el deterioro cognitivo, o para atender situaciones de emergencia. [12]

Gracias a la nueva generación de micrófonos y su progresiva miniaturización, que ofrecen una alta calidad a un bajo coste, hoy en día tenemos multitud de dispositivos provistos de módulos de recopilación de información mediante audio, su almacenamiento y procesado en tiempo real [13]. Estos dispositivos se utilizan frecuentemente en teléfonos móviles, *laptops*, *notebooks*, para proporcionar control de voz en dispositivos de juego y de realidad virtual e incluso en sistemas industriales, así como para la cancelación de ruido y sistemas antirrobo [14].

En función de a qué estén destinadas las aplicaciones de WSN, se requieren distintos niveles de seguridad. A medida que se aumenta el nivel de seguridad, se precisa más computación, lo cual origina un mayor consumo de energía. Por lo tanto, seleccionar el cifrado adecuado, es crítico tanto para la seguridad de datos como para el consumo de energía. Para lograr dicha seguridad, la criptografía puede ofrecer alguna solución, entre sus tres principales categorías: técnicas simétricas, asimétricas e híbridas. En la técnica simétrica, se usa una única clave entre nodos para cifrar y descifrar los datos. Esta clave permanece oculta dentro de la red. En comparación con otras técnicas de criptografía esta es la más utilizada en WSN, ya que supone menos procesamiento, computación y consumo de energía. La técnica asimétrica utiliza claves privadas, las cuales deben ser confidenciales; y públicas, pueden publicarse en la red de manera libre. Estas últimas, se emplean para el cifrado de datos y consumen más energía y memoria en comparación con la técnica anterior, debido al tamaño más grande de clave. En cuanto a la técnica híbrida, combina técnicas de criptografía simétrica y asimétrica, beneficiándose de las ventajas conjuntas de ambas. Está diseñada para tener menores requisitos de memoria y de procesamiento, lo que conlleva a un menor consumo de energía.

La investigación acerca de las redes de sensores inalámbricos se inició en la década de 1980, y fue a partir de 2001 donde se creó un mayor interés desde las perspectivas industriales y de investigación [15]. Esto fue debido a la reciente disponibilidad de circuitos electrónicos de tamaño cada vez más reducido (nanoelectrónica), baratos y de bajo consumo, tales como los procesadores, radios y sensores inteligentes, que a menudo se integran en un solo *chip* (sistema en un *chip*, *SoC*)

La nanoelectrónica comprende los circuitos electrónicos miniaturizados integrados en *chips* semiconductores, siendo su elemento de base el transistor [16]. Los transistores se consideran la unidad mínima del procesamiento, de modo que su comunicación permite que los procesadores lleven a cabo todas las operaciones que realiza un dispositivo que dispone de una de estas unidades como un *smartphone*, un *tablet* o un ordenador. Hasta hace unos años, el tamaño de los transistores se medía en micrómetros (μm), pero actualmente los circuitos integrados de última generación cuentan con piezas de tan solo 22 nanómetros, y se están introduciendo ya diseños que reducen ese tamaño a la mitad. Gracias a esta miniaturización progresiva de los componentes electrónicos, podemos disfrutar de dispositivos tales como ordenadores o móviles, cada vez más ligeros, con mayor velocidad de procesamiento, mayor capacidad y menor consumo.

Por otra parte, la tecnología *wearable* es capaz de llevar a cabo gran parte de las tareas que realizan los ordenadores y dispositivos móviles, con dispositivos más sofisticados que incorporan sensores para interactuar con el cuerpo y con el entorno [17]. Se halla presente en una amplia variedad de campos que satisfacen nuestras necesidades, con el fin de enriquecer nuestra calidad de vida desempeñando un papel activo en la salud, la seguridad de las personas, el entrenamiento deportivo, etc...⁵

⁵ <http://www.dispositivoswearables.net/>

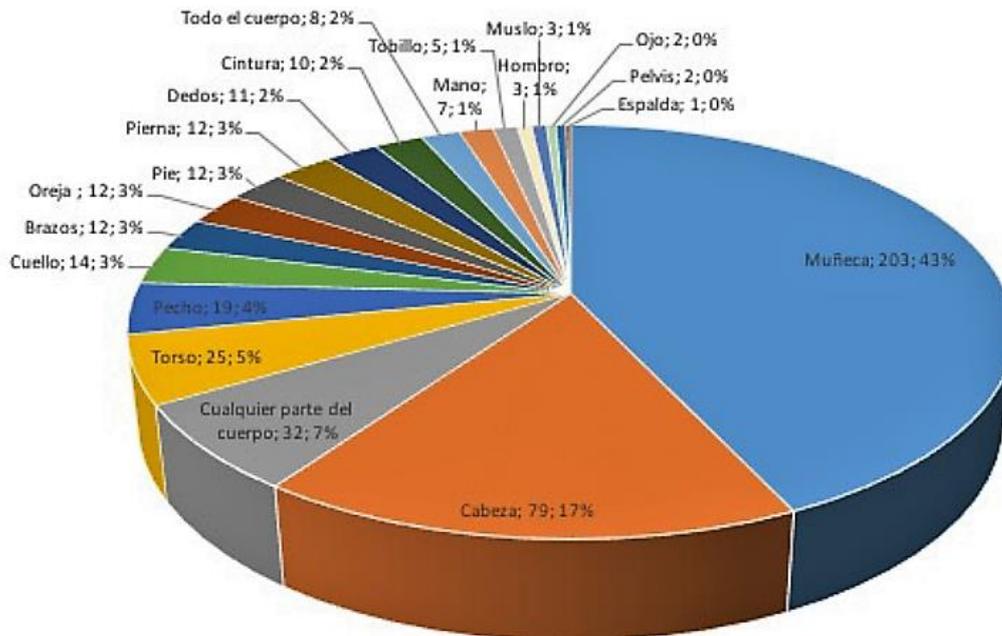


Figura 5. Diagrama de sectores del uso de dispositivos wearable según la parte del cuerpo. [17]

Por tanto, un dispositivo *wearable* es aquel aparato electrónico que se incorpora en una parte de nuestro cuerpo (Figura 5) con la finalidad de realizar alguna función concreta. Una característica fundamental de la tecnología y los dispositivos *wearable* es la capacidad de conectividad inalámbrica (no sólo a *Internet*), que proporciona al usuario acceso a información en tiempo real [17]. La conexión creada entre estos dispositivos da lugar a las ya mencionadas redes de sensores inalámbricos (WSN). El software integrado en los mismos está compuesto de dos partes: sistema operativo o *firmware* y aplicaciones.

En general, los *wearables* recogen información de seguimiento, bien del usuario o del entorno, bien de ambos; y añaden métodos de sincronismo a través de diversas tecnologías inalámbricas. Existen dispositivos desde muy concretos, para monitorizar de forma sencilla un único parámetro, hasta muy elaborados, multisensores y con potente procesamiento interno. Generalmente, cuanto mayor sea el nivel de sencillez, menos energía consumirá y más duración tendrá la batería interna. Las baterías más usadas en los *wearable* son las de ión-litio, las cuales han reemplazado a las de níquel-cadmio. Son pequeñas y frágiles, pero requieren bajo mantenimiento. Actualmente se están estudiando las baterías de grafeno, pero aún resultan muy caras y se encuentran por el momento en fase experimental. Del diseño de unas buenas baterías, en términos de consumo y eficiencia, dependerá el éxito de los dispositivos *wearable* en el futuro. [17]

Hace unos años, cuando la tecnología *wearable* se estaba abriendo camino en el mercado, el desafío era dar con un producto innovador que despertase la curiosidad de los consumidores en el potencial de las nuevas tecnologías, en la actualidad consiste en proponer dispositivos con funciones optimizadas para la recopilación, el análisis y el uso de datos complejos, junto con un diseño apetecible.

En cuanto a la propuesta *Bindi*, la cual cuenta con dos dispositivos *wearables*, colgante y pulsera, integra la adquisición de audio como método adicional e innovador para el reconocimiento de pánico en la voz y activación de la alarma que envía la aplicación móvil al círculo de ayuda seleccionado por la víctima o a las autoridades competentes. Esta cualidad, le diferencia de otros dispositivos presentes en el ámbito de la seguridad y protección de las mujeres ante ataques, como pueden ser *Safelet* (pulsera) o *Artemis* (colgante), los cuales también incluyen la grabación de audio en *streaming* pero tras la activación de la alarma y el envío de la localización GPS, y no como método de activación de la misma, que en el peor de los casos si la víctima se haya inmovilizada por parte del agresor, se pueda suplir la incapacidad de pulsar el botón de pánico.

2.1. ENTORNO SOCIOECONÓMICO

Las aplicaciones de los dispositivos *wearables* se pueden llevar tanto al ámbito personal como al comunitario. Algunos ejemplos concretos de dichas aplicaciones son: en atención a personas mayores, videojuegos, música, biomedicina, educación, apoyo a discapacidad, investigación científica, seguridad, etc [17]. En la Figura 6 se muestran las aplicaciones de los *wearables* en 2016.

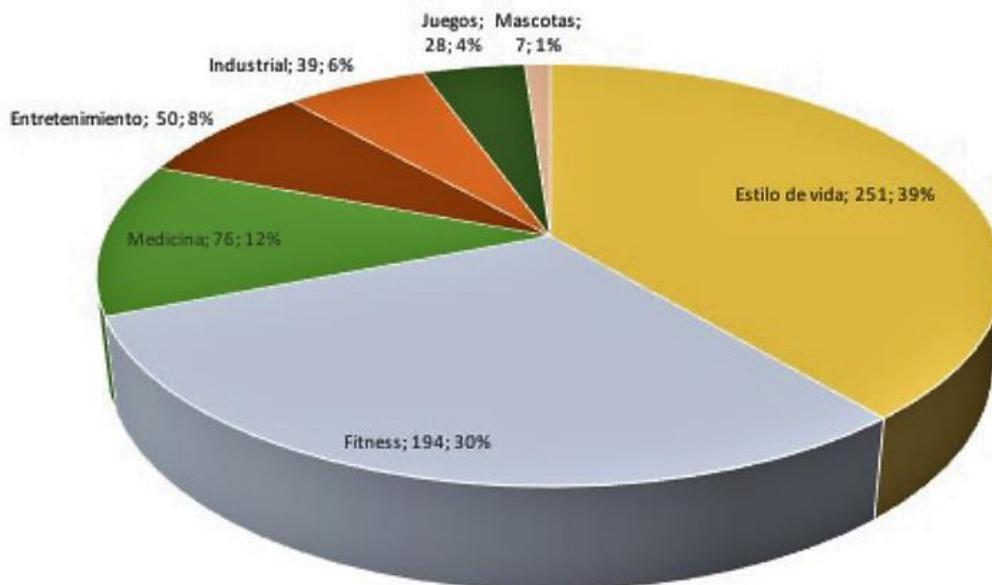


Figura 6. Diagrama de sectores de los dispositivos wearable según su aplicación. [17]

La nueva generación de tecnología *wearable* conforma el camino hacia la completa instauración de *IoT* y sus ventajas. Además, *IoT* puede implementarse en infinidad de áreas: *smart houses*, *fitness* y salud, gestión de datos sobre el medio ambiente, *smart cities*, *smart factories*, coches conectados, *smart hospitals*, objetos de nuestro día a día con sensores y su conexión con otros dispositivos inteligentes, gestión de emergencias con drones, identificación y seguridad, etc.⁶

Según análisis realizados por expertos de IDC (*International Data Corporation*), el sector de la tecnología *wearable* está alcanzando un nuevo nivel de progreso. En 2017 se vendieron acerca de unos 125 millones de dispositivos a nivel mundial, un 20,4% más que en 2016. Se prevé que las cifras prácticamente se dupliquen hasta un total de 240 millones de unidades vendidas en 2021. Además, IDC estima que los próximos *wearable* que encabezen las listas de ventas, serán el *smart clothing* (ropa) y los *hearables sets*, dispositivos digitales que se colocan en el oído y

⁶ <http://www.wearablespain.com/Wearable.html>

ofrecen un *feedback* auditivo de mensajes, notificaciones o actualizaciones climáticas y de noticias. [18]

Estos dispositivos se encuentran cada vez más presentes en diferentes áreas de nuestra vida cotidiana, como en el trabajo, el deporte o la salud. En la Figura 7, se muestra el uso que se da a los *wearable* en el ámbito personal, mediante la leyenda en azul claro, y el uso en el trabajo o el ámbito empresarial, representado con la leyenda azul oscuro.

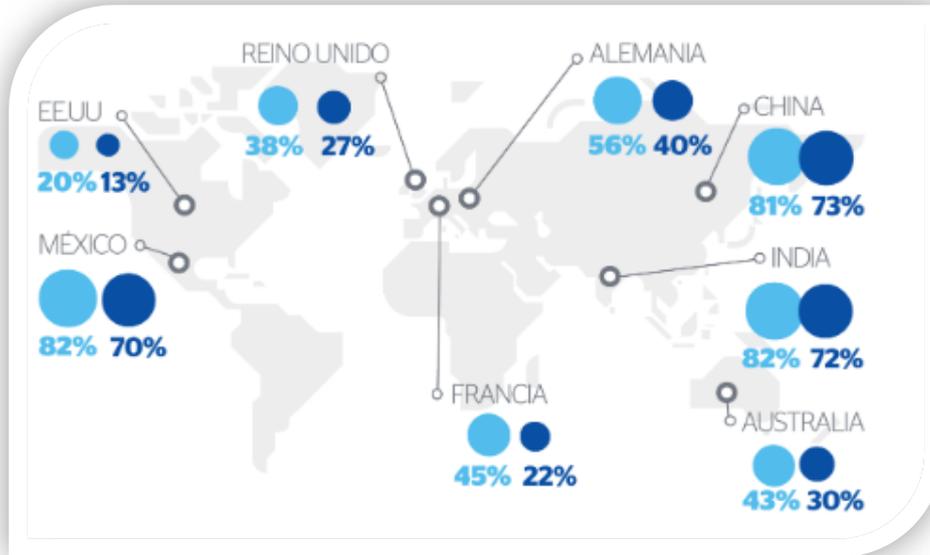


Figura 7. Mapa de uso de los *wearable* en el trabajo y en la vida personal.⁷

Por otra parte, a pesar de que esta tecnología constituye un sector en auge que se está poniendo al frente del mercado de la salud y los deportes, también hay que destacar algunos inconvenientes que probablemente se minimizarán con las futuras mejoras en este mercado en desarrollo, pero con un gran potencial: [17]

- *Connotaciones sociales y psicológicas inherentes a los dispositivos wearable.* Están basadas en el pensamiento “Siempre encendido, siempre conectado” que suscita la tecnología hoy en día a través del *IoT*. También se debe a la gran cantidad de información personal que se recoge mediante estos dispositivos.
- *Privacidad.* Como consecuencia de lo anterior, nace el escepticismo ante el verdadero uso y destino de todos los datos recopilados.
- *Subordinación tecnológica que supone.* Negación de los usuarios a un uso adictivo del aparato, además de la deshumanización que tra consigo el hecho de estar continuamente monitorizado.
- *Coste elevado.* Algunos de los dispositivos relativos a esta tecnología *wearable* aun presentan un coste demasiado alto para ahondar de forma significativa en un público de consumo.

Por ello, uno de los requisitos principales de *Bindi*, para cumplir con las especificaciones establecidas para el concurso “Anu & Naveen Jain Women’s Safety XPRIZE”, era crear una solución económica que no costase más de 40 dólares para que mujeres de toda clase social y

⁷ https://www.bbva.com/wp-content/uploads/2017/10/ebook-cibbva-trends-wearable_0.pdf

condiciones económicas, pudiera permitírsele. Se trata de que un derecho humano tan primordial como la seguridad, pueda estar al alcance de todos a través de un *wearable*.

2.2. NORMATIVA

La legalidad de las grabaciones de voz es uno de los temas más controvertidos en relación con la protección de datos. En la Unión Europea el tratamiento de datos personales y la libre circulación de estos, debe ceñirse a lo establecido en la Directiva 95/46/CE y en la Directiva 2002/58/CE. [19]

En lo referido a grabaciones de voz, el contexto en el que se desarrollan, la finalidad con la que se realizan y si hay o no consentimiento para ello, constituyen factores muy importantes a tener en cuenta. El Tribunal Constitucional dictamina que, si una conversación es grabada por uno de los interlocutores, no se está violando ningún derecho; pero, si la conversación es grabada por un tercero ajeno a la conversación y se divulga más allá de los participantes en ella, entonces sí se está transgrediendo un derecho establecido en la propia Constitución Española. A pesar de ello, la Constitución establece el derecho a la intimidad de las conversaciones, cuando esta conversación alude a tu vida íntima y personal, se encuentra ya en el ámbito privado y no puede ser legal.

De esta forma, a la hora de establecer si una grabación ha violado la Ley de Protección de Datos, se deben considerar dos aspectos: la voz debe ser considerada un dato personal, por lo tanto, debe resguardarse en la Ley de Protección de Datos; y por otra parte debe establecerse si hay o no consentimiento para que esa voz grabada sea presentada, por ejemplo, como prueba ante un tercero, como una Administración de Justicia o un tribunal (Artículo 11, LOPD) [20]. En consecuencia, si la grabación constituye un dato personal, será necesario disponer del consentimiento para su uso (Artículo 6.1, LOPD) [20]. Sin embargo, en los casos que la grabación esté realizada por uno mismo, siendo uno de los participantes de la conversación, no se precisará un consentimiento explícito y la grabación podrá ser utilizada ante un tercero, como puede ser presentarla ante un tribunal a modo de prueba o testimonio.

Según la actual normativa, no será necesario aportar este consentimiento en los siguientes casos: [20]

- **Cuando el uso de los datos se realice con la finalidad de proteger al interesado (Artículo 7.6, LOPD).** *En lo que concierne a este proyecto, este inciso está estrechamente relacionado con el propósito para el cual se realizan las grabaciones de voz en el colgante, ya que no es con más fin que con el de proteger la seguridad de las mujeres y detectar cuando se está produciendo una agresión sexual hacia ellas.*
- Cuando sean recogidos en el marco de una relación contractual o laboral y necesarios para el buen desarrollo de esa relación (Artículo 6.2).
- Cuando los datos estén disponibles en fuentes accesibles al público y su tratamiento sea fundamental para preservar la intimidad de la persona implicada (Artículo 6.2).
- Cuando se recojan para el ejercicio de las funciones básicas de las Administraciones Públicas (Artículo 6.2).

En resumen, dado que lo expuesto en la LOPD sólo permite el tratamiento de dichos datos mediante aprobación del titular o habilitación legal por razones de interés general, y que en el caso de los *wearable* dicho consentimiento se realiza por parte del propio usuario al aceptar las condiciones que son propias del uso del dispositivo, se puede concluir que: *Al utilizar el dispositivo wearable con sus correspondientes sensores, estás aceptando sus términos de uso en cuanto a privacidad y utilización de tus datos.*

Lo expuesto en el párrafo anterior, puede aplicarse tanto a datos fisiológicos o identificativos, como de voz, que es lo que concierne a este proyecto. Dicho así, el colgante como dispositivo *wearable* capaz de detectar estrés en la voz mediante la adquisición del audio, recoge datos del audio de la usuaria, y por tanto en el momento que se produzca la agresión estará adquiriendo esos datos tanto de la víctima como del agresor, sin que ninguno de ellos sea quien está grabando la supuesta conversación de forma directa, pero se entiende que la propia víctima al estar llevando puesto el dispositivo, está dando su aprobación y consentimiento para realizar dicha grabación indirectamente. Por consiguiente, al ser la víctima quien indirectamente inicia la grabación y además forma parte de ella, no es un tercero, junto con el hecho de que dicha grabación es en favor de protegerse, se concluye que su tratamiento y aportación en procesos judiciales es lícita.

3. DESCRIPCIÓN DEL SISTEMA

En este capítulo se presenta el sistema que se ha diseñado y desarrollado en este trabajo. Este sistema debe permitir la adquisición de audio y su transmisión inalámbrica, en formato comprimido, de cara a su posterior uso en aplicaciones que requieran el procesamiento de la voz para la seguridad de las personas. Este uso puede ser a modo de testimonio ante una agresión o a modo de detección de situaciones de *stress* negativo (*distress*) en la usuaria o el usuario. A partir de la especificación recibida, se han estudiado las alternativas disponibles en el mercado, en cuanto a componentes y formatos, así como las opciones de captura, protocolos de envío y procesamiento del audio, para generar un sistema económico, fiable y eficiente en consumo.

En lo que se refiere a *Bindi*, la propuesta de *UC3M4Safety*, este sistema supone un beneficio añadido, ya que optimiza el desempeño de las pruebas y grabaciones necesarias para entrenar adecuadamente los algoritmos creados por el Departamento de Teoría de la Señal y Comunicaciones para detectar: estrés en voz, gritos, silencios, palabras clave a la hora de determinar si se trata de agresión verbal o si la voz se identifica con la del usuario/a, etc. Y siempre sin perder de vista el hecho de poder presentar el propio audio como prueba judicial.

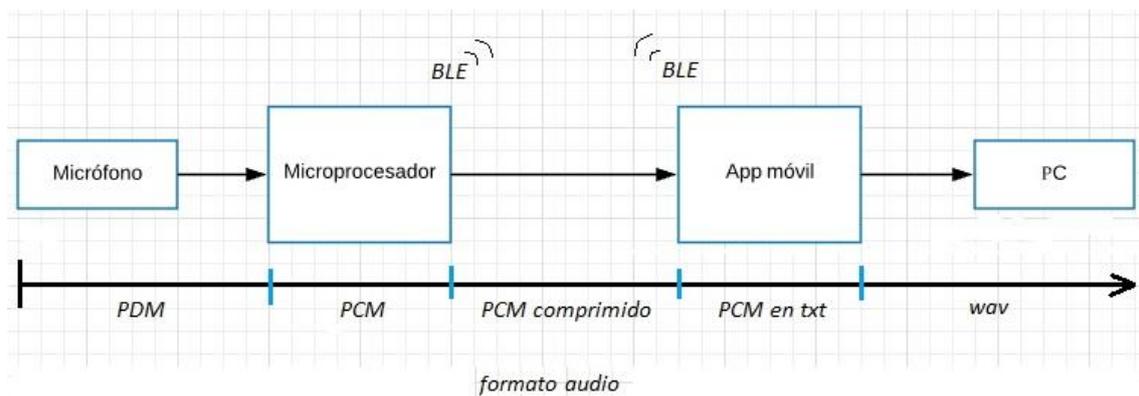


Figura 8. Esquema del sistema desarrollado para la adquisición y procesamiento del audio.

En la Figura 8, se muestra un esquema del sistema junto con una representación de la evolución del formato del audio a lo largo del mismo, durante su operación normal cuando se ha activado la grabación de audio porque se ha detectado una situación de peligro.

Las funciones que se desempeñan en cada etapa son:

- Micrófono: grabación del audio, amplificación de la señal analógica, conversión analógica digital y conversión a formato PDM.
- Microprocesador: adquisición del audio, formato PDM, conversión a formato PCM, compresión de este y envío inalámbrico a través del protocolo Bluetooth® a un teléfono móvil.
- Aplicación móvil (*App*): recepción del audio comprimido enviado por Bluetooth®, creación del fichero *txt* donde se almacena, descompresión de este y procesamiento con herramienta de detección de estrés en audio⁸.

⁸ Este procesamiento ha sido realizado por el Grupo de Procesado Multimedia del Departamento de Teoría de la Señal y Comunicaciones

Para el diseño de *Bindi*, tanto de la red inalámbrica compuesta por el colgante, la pulsera y el teléfono móvil, como del sistema de entrenamiento de la detección de *stress* en voz, se ha añadido la capacidad de comunicación con un ordenador, donde se puedan depurar los algoritmos de compresión, así como los programas de captura y envío de audio. Para esta depuración, y para su integración en la cámara anecoica donde se graban los audios de entrenamiento, se ha incluido la capacidad de comunicación alámbrica y sin compresión de los datos. Por tanto, se ha añadido un último bloque al esquema:

- PC: descompresión del audio y creación del archivo *wav* para su reproducción.

Como anteriormente se menciona, el punto fuerte o la ventaja competitiva que UC3M4Safety propone en su dispositivo *wearable*, consiste en la adquisición de audio que es de utilidad tanto para detectar pánico en la voz, complementando de esta forma al botón del pánico, como para su posterior uso en procesos judiciales sirviendo de prueba o testimonio e incluso para un análisis forense.

3.1. CONCEPTOS PREVIOS

Antes de adentrarse en la descripción de la arquitectura del sistema parte por parte, se exponen una serie de conceptos teóricos que se han analizado y estudiado para contar con un buen criterio a la hora de hacer algunas elecciones previas tales como el formato de audio con el que se va a trabajar o el modelo y tipo de micrófono más adecuado para el fin de este trabajo.

3.1.1. AUDIO

Una señal de audio es una señal eléctrica analógica correspondiente a una onda sonora. Su ancho de banda se encuentra dentro del rango de frecuencias comprendido entre los 20 Hz y los 20 kHz. Esta señal analógica puede someterse a una codificación dando lugar al denominado *audio digital*. Esta codificación digital se divide en dos procesos: el muestreo y la cuantificación digital de la señal.

Mediante el muestreo se fija la amplitud de la señal en instantes concretos de tiempo. Al número de muestras que se toman por unidad de tiempo se le denomina tasa de muestreo o frecuencia de muestreo. Para reproducir el intervalo de frecuencias que conforma el espectro audible (20 Hz-20 kHz), según el Teorema de Nyquist que establece una tasa de muestreo de poco más del doble, se necesita una frecuencia de muestreo de unos 44 kHz.

La cuantificación convierte el nivel, normalmente de tensión, de las muestras fijadas durante el muestreo en un valor entero finito. De modo que, siguiendo una cuantificación lineal de 8 bits, medirá 256 niveles (2^n , siendo n el número de bits) de señal equidistantes. También pueden realizarse cuantificaciones no lineales, por ejemplo logarítmicas, las cuales son adecuadas para señales de baja amplitud en promedio, tales como la voz humana.

El formato de audio digital PCM lineal más usado es el del *Compact-Disc* (CD), con 44 kHz de tasa de muestreo y cuantificación lineal de 16 bits, permitiendo registrar señales con componentes hasta los 20 kHz y con una relación señal a ruido (SNR) de más de 90 dB. Figura 9.

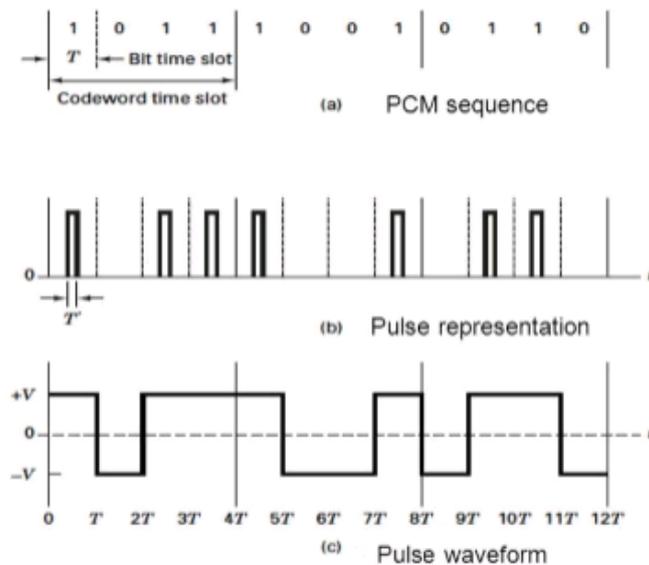


Figura 9. Modulación por código de pulsos (PCM).⁹

3.1.2. MICRÓFONO

Un micrófono es un transductor que esencialmente consta de un condensador variable con una alta impedancia de salida, siendo esta del rango de *Giga-ohmios*. En el interior del encapsulado que conforma el micrófono, la señal de salida del transductor se envía a un preamplificador para acomodar la impedancia de salida a un valor adecuado para su uso posterior.

A la hora de elegir el micrófono más adecuado para este proyecto, se han estudiado y analizado las características de los tipos más comúnmente utilizados. Dichos tipos constituyen:

- Micrófonos de condensador Electret (ECM's): suelen usarse en teléfonos y portátiles. Tienen una respuesta en frecuencia bastante amplia (de 10 Hz a 30 kHz), además son pequeños y bastante sensibles.

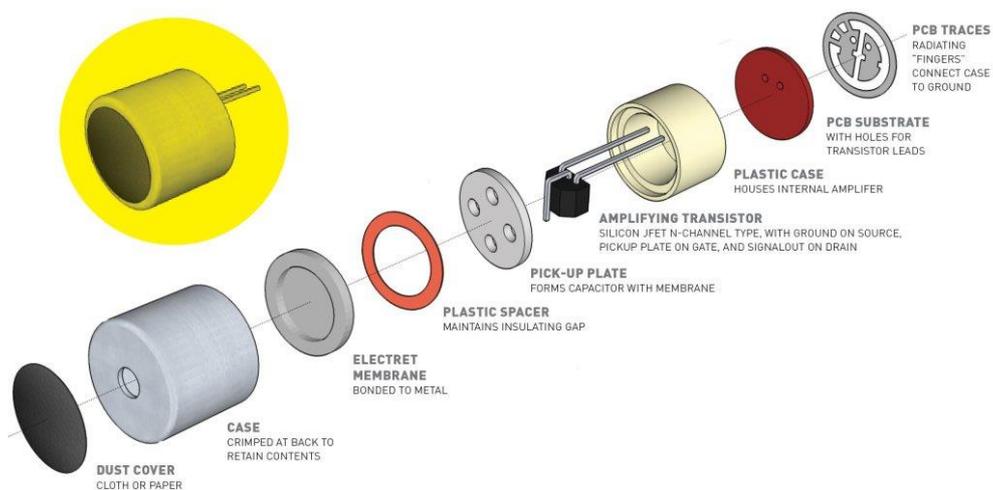


Figura 10. Explosionado de un micrófono electret.¹⁰

⁹ <http://www.ques10.com/p/11513/pulse-code-modulation/>

¹⁰ Charles Platt. Make Newsletter.

Como puede verse en la Figura 10, está compuesto de dos patillas que permiten su soldadura a una PCB y una cápsula que en su parte superior está cubierta por un material poroso fijado con pegamento. La función de dicho material poroso consiste en preservar el material *electret* sensible, del ruido del viento y el polvo. Debajo del material poroso, se encuentra un orificio en la cápsula, por el cual se produce la adquisición del audio.

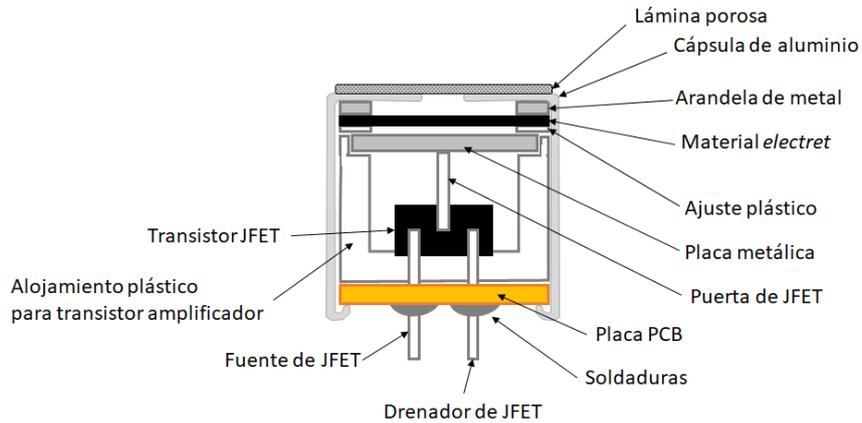


Figura 11. Sección de un micrófono electret.

En el interior de la cápsula de aluminio se encuentra contenido el material *electret* o diafragma y un amplificador. El amplificador consta únicamente de un JFET configurado en fuente común conectada a masa. En el espacio que queda entre la placa metálica y el diafragma se encuentra el dieléctrico del condensador. La característica principal de los micrófonos *electret*, es que vienen con una carga eléctrica en su material dieléctrico desde su fabricación. Figura 11.

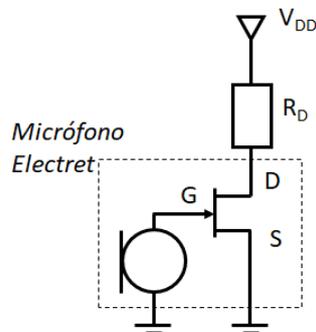


Figura 12. Circuito equivalente de un micrófono electret.

En la Figura 12, puede verse que la puerta del JFET es el pin de control que se conecta a una de las placas del condensador, la fuente se conecta a la masa y el drenador a una resistencia y alimentación de un circuito adicional externo.

En cuanto a su funcionamiento, el *electret* provoca una carga fija y una tensión en el condensador. Sabemos que la tensión en un condensador sigue esta ecuación:

$$V = \frac{Q}{C} \quad \{1\}$$

Donde Q es la carga y C la capacidad del condensador.

Además, la ecuación de la capacidad para un condensador con placas paralelas es la siguiente:

$$C = e \cdot \frac{A}{t} \quad \{2\}$$

Donde e es una constante que depende de las propiedades del material, A es la superficie de las placas y t es la distancia entre las mismas, igual al espesor del material dieléctrico.

Debido a que el diafragma se mueve hacia arriba y hacia abajo por las vibraciones que crean las variaciones de presión sonora, la distancia entre las placas conductoras del condensador se altera y por tanto su capacidad. El volumen del dieléctrico varía puesto que el espesor t se hace más grande y más pequeño con las vibraciones. Y a su vez, la variación de la capacidad del condensador provoca la consecuente variación de tensión entre sus terminales.

A medida que la tensión de la fuente se modifica, la tensión V_{gs} también lo hace puesto que la fuente está conectada a masa, dando lugar a que el JFET cambie su corriente de salida (I_{gs}), variando de forma proporcional a las ondas sonoras y generando una forma de onda de tensión equivalente y amplificada como salida, a través de R_s .

El uso del JFET como amplificador tiene su justificación en que presenta una resistencia de entrada muy grande (del orden de *Mega-ohmios*), lo que implica que la corriente que se extrae del condensador es mínima.

- Micrófonos MEMS (*Microelectromechanical Systems*): de la misma forma que los ECM's, los MEMS funcionan como micrófonos de condensador. En resumen, constan de un diafragma flexible que puede moverse libremente sobre una placa fija posterior, todo fabricado en oblea de silicio. La principal diferencia con el micrófono descrito anteriormente es el tamaño nanométrico. El conjunto constituye un condensador variable con una carga eléctrica fija situada entre el diafragma y la placa posterior. La presión producida por las ondas sonoras, pasa a través de unos orificios en la placa posterior haciendo que el diafragma se mueva en función de la amplitud de dichas ondas. Y el movimiento del diafragma cambia la distancia entre este y la placa, variando a su vez la capacidad transformándola en una señal eléctrica. Figura 13.

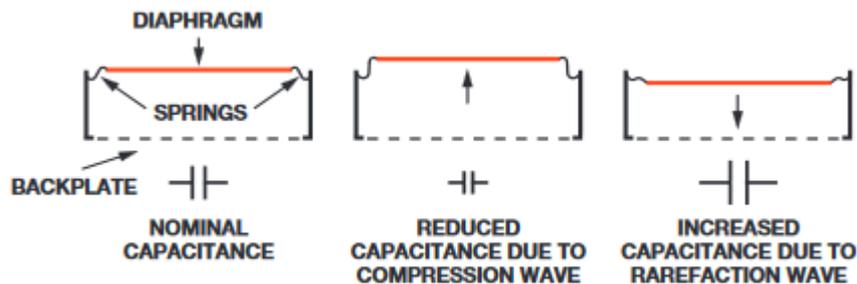


Figura 13. Variación de la capacidad de un micrófono MEMS con la amplitud de la onda sonora. [21]

La tecnología MEMS está impulsando el avance y la renovación de los micrófonos de condensador, ya que está aprovechando los grandes progresos que se han llevado a cabo en la tecnología del silicio durante las últimas décadas. Este avance tecnológico ahorra espacio y aumenta la durabilidad en aplicaciones de entrada de voz, desde teléfonos y *tablets* hasta medidores de nivel de ruido y auriculares con cancelación de ruido.

Los micrófonos MEMS permiten numerosos avances en la calidad del sonido, esto añadido a su reducido tamaño y su bajo consumo, los hace especialmente adecuados para su uso en dispositivos móviles que podrán estar sometidos a entornos ruidosos. Además, los micrófonos digitales eliminan la necesidad de implementar un convertidor ADC entre su salida y la entrada al microprocesador, a diferencia de los micrófonos analógicos que requieren la conversión analógico-digital para poder procesar los datos que se recogen con él [22].

Se ha realizado una comparación entre las características de ambos tipos de micrófonos, obteniendo que en cuanto a: [23]

- **Área:** existen micrófonos MEMS digitales de tan solo $0,70 \text{ mm}^2$, lo cual es una gran ventaja a la hora de utilizarlos en circuitos impresos donde la distribución y aprovechamiento del espacio en la PCB es tan importante.
- **Robustez:** los micrófonos MEMS pueden ensamblarse en el mismo proceso de soldadura por reflujo que el resto de los circuitos integrados que componen una PCB. Sin embargo, los ECM no pueden soportar las altas temperaturas de tal proceso de soldadura, de modo que deben soldarse a mano en numerosas ocasiones.
- **Rendimiento:** los MEMS tienen una mayor "densidad de rendimiento", lo cual hace que su nivel de ruido sea mucho más bajo que el de los ECM. De modo que, aunque haya algunos ECM que presenten una SNR similar a la de los MEMS, generalmente tendrán un volumen mucho mayor.
- **Sensibilidad:** ante vibraciones los MEMS son menos sensibles que los ECM. Esto se debe a que el reducido tamaño del diafragma de los micrófonos MEMS, da lugar a una masa muy baja haciendo que sea mucho menos susceptible a la vibración. Además, los MEMS también muestran una gran estabilidad a lo largo de un amplio rango de temperaturas.

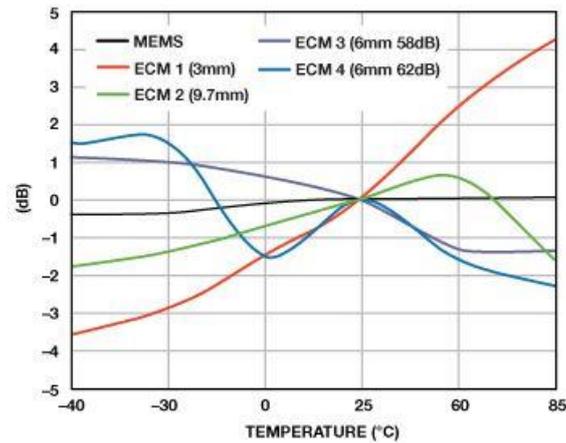


Figura 14. Sensibilidad frente temperatura MEMS vs. Electret. [21]

En la Figura 14, se muestra el cambio en la sensibilidad debido a una variación de la temperatura ambiente entre -40° y 85° . La línea negra corresponde a una variación inferior a 0'5 dB en el rango de temperatura mencionado para los MEMS, mientras que los ECM's introducen una variación de hasta 8 dB, para tamaños menores.

- **Respuesta en frecuencia:** los ECM pueden llegar a tener una variación notable en sus respuestas, sobre todo en bajas y altas frecuencias, en cambio los MEMS tienen una respuesta en frecuencia más uniforme. Todo ello gracias a los controles estrictos en los procesos de fabricación de los MEMS, para asegurar una estabilidad y variación en el rendimiento dispositivo a dispositivo mucho menos significativa.

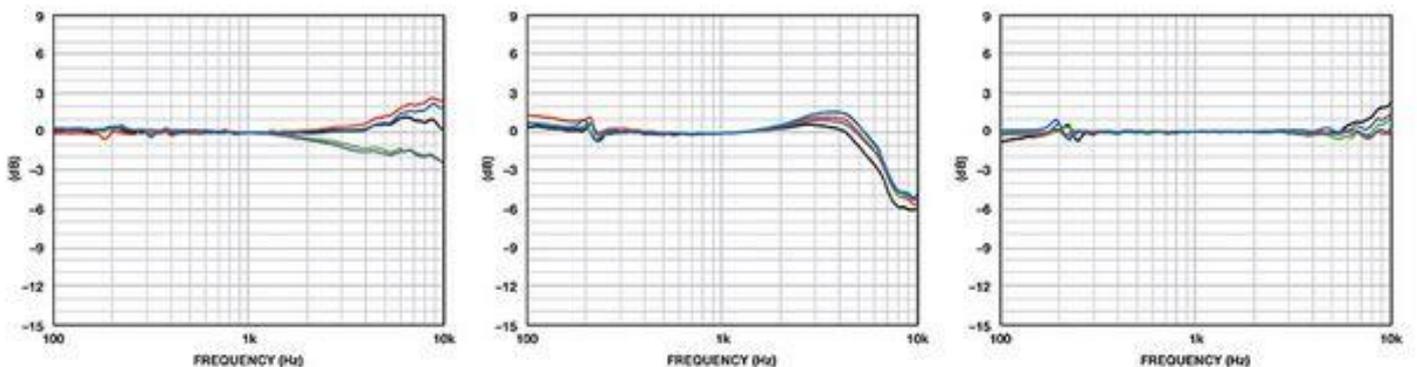


Figura 15. Respuesta en frecuencia para diferentes tipos de ECM'S. [21]

En la Figura 15, se aprecia que la respuesta en frecuencia normalizada de un conjunto de ECM's presenta una diferencia significativa a altas y bajas frecuencias.

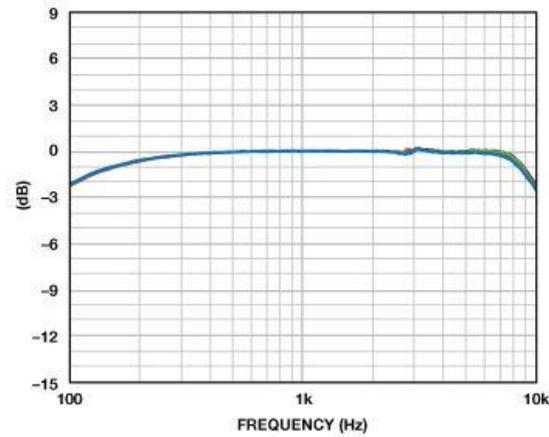


Figura 16. Respuesta en frecuencia de un conjunto de diferentes MEMS. [21]

Sin embargo, en la Figura 16 la respuesta en frecuencia para los distintos tipos de MEMS seleccionados es prácticamente idéntica.

- **Consumo:** los micrófonos MEMS presentan un consumo de energía mucho más bajo. Los niveles de corriente y de tensión necesarios para su funcionamiento son menores al tratarse de dispositivos de dimensiones nanométricas.
- **Coste:** para los MEMS el coste es mayor, pero teniendo en cuenta el ahorro que supone no tener que implementar un ADC y el montaje asociado al mismo, así como la mejor respuesta del micrófono MEMS, es asumible para sistemas embebidos donde el procesamiento digital del audio es una parte muy importante.

En resumen, después de este análisis se ha seleccionado un micrófono MEMS digital puesto que cumple las especificaciones de área reducido, bajo consumo energético y robustez que se requieren para su implementación en la PCB del colgante. Esta placa incluye numerosos componentes tales como un microprocesador con antena inalámbrica de comunicación, un vibrador, un botón de pánico, una batería con su cargador y monitor, etc. lo que requiere un buen aprovechamiento del espacio. Además, la selección de un micrófono MEMS digital es beneficiosa en tanto que tiene integrado en él un convertor analógico-digital (ADC), permitiendo una ruta de adquisición de audio completamente digital hasta el microprocesador. Por último, los micrófonos digitales se emplean a menudo en aplicaciones donde las señales de audio analógicas pueden ser sensibles a interferencias. Una colocación del micrófono próxima al microprocesador permite una manipulación sin inconvenientes cercana a elementos que utilizan un protocolo de comunicación inalámbrico (*Wi-Fi*, *Bluetooth*®...) [22]. En la Figura 17 se muestran los componentes *hardware* incluidos en la arquitectura del colgante.

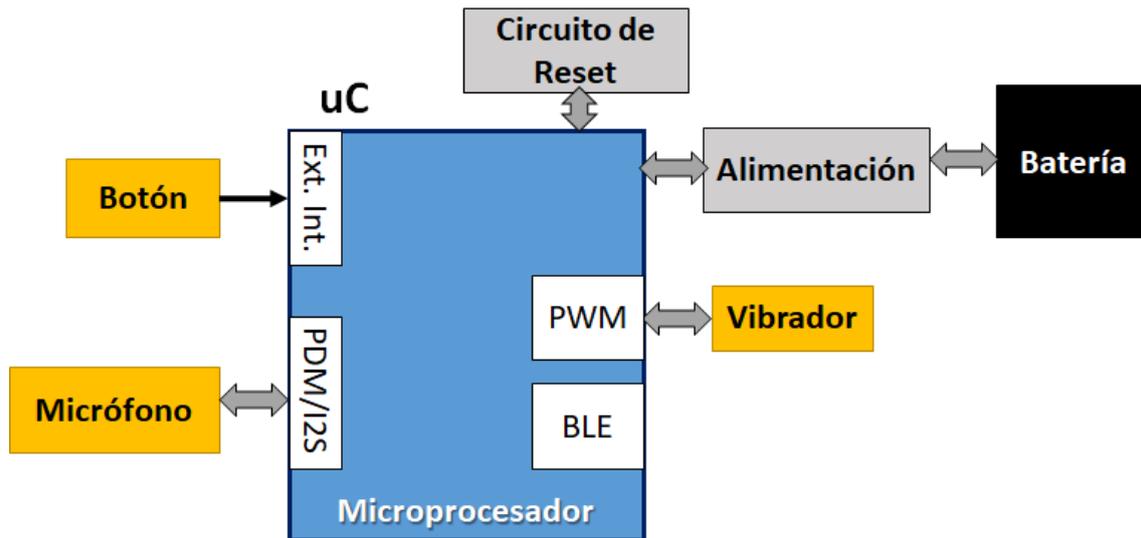


Figura 17. Diagrama de bloques de la arquitectura del colgante.

El micrófono MEMS elegido es un MP34DT02, cuyas características básicas son:

- Bajo consumo de energía
- Omnidireccional
- Salida PDM
- 60 dB SNR
- Sensibilidad -26 dBFS

Entre sus aplicaciones mencionadas en la hoja de catálogo, cabe destacar que es un modelo adecuado para su uso en el reconocimiento de voz [24].

3.1.3. FORMATO

Los micrófonos MEMS digitales, traducen los movimientos de la membrana a niveles altos ('1') y niveles bajos ('0') de lógica, dando lugar a una salida digital que se puede proporcionar tanto en modulación de la señal por densidad de pulsos (PDM), como en formato I2S (*Inter-IC Sound Bus*).

Un micrófono I2S, posee la misma circuitería digital que uno con salida PDM y además incluye un *Decimation Filter*, también conocido como *Filtro Anti-Aliasing*, y un puerto serie para la salida de audio. Figura 18.

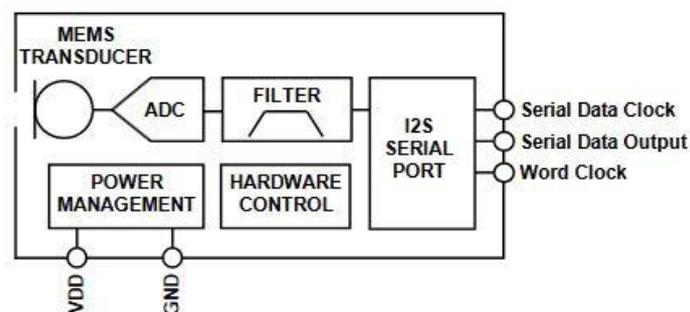


Figura 18. Diagrama de bloques típico de un micrófono MEMS con interfaz I2S. [22]

La interfaz digital I2S se ha venido usando durante años en convertidores de audio y procesadores, pero actualmente también se está integrando en micrófonos. Sin embargo, el formato I2S utiliza dos señales de reloj: un reloj de bit y un reloj de palabra.

La interfaz PDM (*Pulse Density Modulation*) es la más utilizada para micrófonos digitales y da lugar a un flujo de datos de un solo bit altamente sobre muestreados. Figura 19.

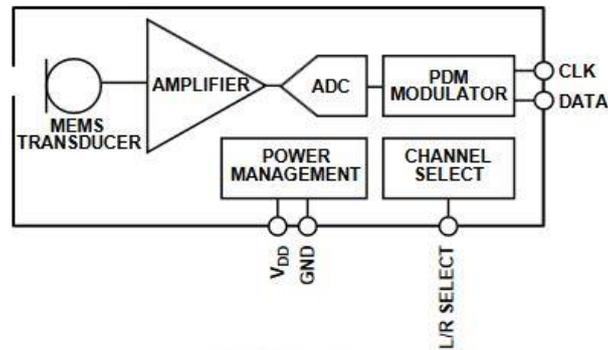


Figura 19. Diagrama de bloque típico de un micrófono MEMS con interfaz PDM. [22]

La densidad de los pulsos en la salida es proporcional a la intensidad de la presión sonora, siendo parecida a la PWM (*Pulse Width Modulation*) empleada en los amplificadores. La diferencia reside en que la PWM utiliza un tiempo entre pulsos constante y la señal se codifica en el ancho de pulso, mientras que la PDM utiliza un ancho de pulso constante y la señal se codifica en el tiempo entre pulsos. Figura 20.

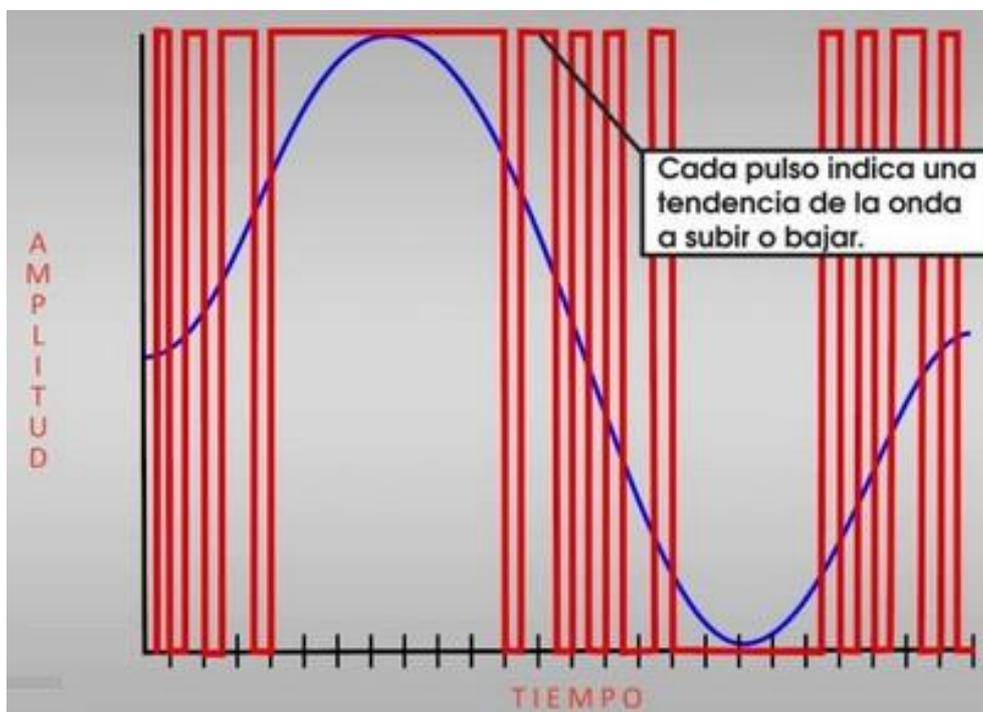


Figura 20. Modulación por densidad de pulsos (PDM).¹¹

¹¹ <http://www.defectodigital.com.mx/audiodigitalsd/>

Además, la mayoría de los micrófonos digitales, incluido el elegido para el proyecto, además de contar con los pines de alimentación, masa y salida de datos, cuentan con una señal de reloj y un pin de control L/R. La modulación por densidad de pulso permite que dos micrófonos se conecten de forma conjunta a una línea de reloj y a otra de datos. Para ello, se configuran los micrófonos de manera que generen su señal de salida en diferentes flancos de la señal del reloj asegurándose que los datos de ambos canales se capturen de forma simultánea. La frecuencia del reloj típica que genera la PDM se encuentra en el rango de 1 MHz a 3'5 MHz. Los datos de salida del micrófono se adquieren en el flanco del reloj seleccionado y durante la otra mitad del ciclo se mantienen en alta impedancia, lo que hace posible que dos salidas digitales compartan una única línea de datos. El pin de control L/R del micrófono fija en qué flanco de reloj se validan los datos. El formato PDM es el más adecuado para aplicaciones de audio portátiles (móviles, *tablets*), en las que el dispositivo puede encontrarse en escenarios ruidosos.

3.2. DRIVERS DEL MICRÓFONO

Se entiende como *driver* a aquel programa *software* que sirve de controlador o gestor de un periférico (componente incluido en un microprocesador y que suele interactuar con entradas y/o salidas de este), en este caso del micrófono. Permite, mediante órdenes implícitas en el *software*, interactuar con el periférico y de esta forma hacer una abstracción del hardware proporcionando una interfaz directa para utilizar el dispositivo. Mediante los *drivers* del micrófono se realiza la adquisición del audio en el microcontrolador.

En este proyecto se ha utilizado un microprocesador de *Nordic Semiconductors*[®], nRF52832, con arquitectura ARM (*Advanced RISC Machine*) Cortex-M4, 32 bits de ancho de palabra y juego de instrucciones Thumb-2. Este microprocesador está integrado en un dispositivo OEM (*Original Equipment Manufacturer*), que incluye la antena de comunicación inalámbrica y que permite utilizar el protocolo Bluetooth[®] gracias a que el *software*¹² está incluido en su memoria de programa. El soporte de *software* para el Soc (*System-on-a-chip*) nRF52832 empleado en este proyecto, se divide en dos partes: *SoftDevices* los cuales son pilas completas de protocolos inalámbricos y el *Software Development Kit* nRF52 SDK que constituye un entorno gráfico que permite diseñar proyectos complejos con bibliotecas básicas de código compilable, para configurar y utilizar los distintos componentes y periféricos del microprocesador.

El kit de desarrollo de *software* nRF52 SDK proporciona código fuente en C y bibliotecas pre compiladas que cuentan con BLE (*Bluetooth[®] Low Energy*), ANT y perfiles de dispositivos, comunicación inalámbrica y ejemplos de aplicaciones. En todo ello se incluyen ejemplos y bibliotecas de controladores para los periféricos del SoC nRF52832 y es compatible con las siguientes herramientas de desarrollo:

- *Segger Embedded Studio*
- *Keil MDK-ARM*

De cara a crear los *drivers* del micrófono, se han empleado ambas herramientas de desarrollo junto con las bibliotecas y *drivers* que proporciona el SDK referidos a la interfaz PDM del microcontrolador.

¹² Este software se denomina *stack* o pila

En la Figura 21 se muestra en qué etapa de diseño y desarrollo del sistema nos encontramos, el colgante donde se integran el micrófono y el microprocesador. A lo largo del presente documento en los correspondientes apartados, se muestra una figura a modo de croquis para situar el punto del que se está tratando.

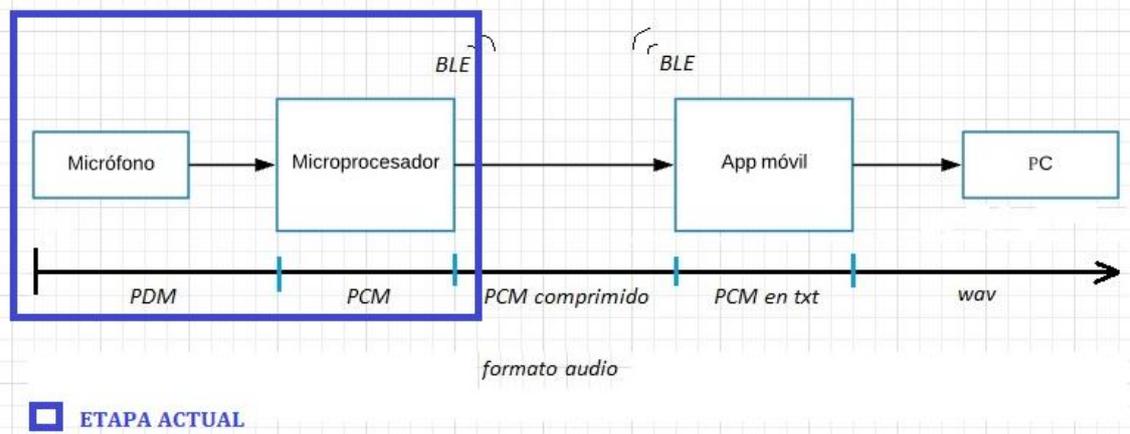


Figura 21. Etapa de adquisición del audio.

3.2.1. MICROCONTROLADOR

Un microcontrolador consiste en un circuito integrado digital y que es programable, lo cual le hace capaz de ejecutar diferentes órdenes que almacena en su memoria¹³. Está compuesto por los mismos bloques básicos de funcionamiento de un ordenador: unidad central de proceso (CPU), memorias de datos y de programa (ROM, RAM, etc.) y periféricos de entrada y salida (I/O). Puede usarse para una gran variedad de aplicaciones como manejo de sensores, controladores, control de motores, robots, entre otros. Se debe establecer su funcionamiento mediante software a través de programas que contengan las instrucciones que debe realizar. Los programas se guardan en una memoria y la CPU se encarga de procesar paso a paso las instrucciones. Los lenguajes de programación característicos utilizados son ensamblador y C.

Un microcontrolador suele estar constituido por una fuente de reloj incorporada y otra basada en un cristal de cuarzo externo, diferentes tipos de memoria (RAM, ROM, FLASH, etc.), dispositivos de entrada y salida tales como DAC's y ADC's (*Digital-to-Analog Converter* y *Analog-to-Digital Converter*), *Timers*, puerto serie UART (*Universal Asynchronous Receiver-Transmitter*) y otros buses de comunicación (I2C o SPI).

Hoy en día, existen numerosos microcontroladores comerciales que están pensados para su utilización con baterías, debido a su bajo consumo de energía. Poseen la capacidad de permanecer en espera hasta que se produzca algún tipo de evento (ej.: pulsar un botón) mediante una interrupción, logrando tener un estado muy bajo de consumo cuando está en reposo.

¹³ <https://es.wikipedia.org/wiki/Microcontrolador>

En este sistema se emplea el microcontrolador nRF52832 con procesador *ARM Cortex-M4F* de 32 bits, consumo de energía ultra bajo y multiprotocolo, ideal para BLE (*Bluetooth® Low Energy*), ANT y aplicaciones inalámbricas de baja potencia de 2'4 GHz. Está incorporado en la placa de evaluación nRF52 DK (*Development Kit*) de *Nordic Semiconductor*. Figura 22.



Figura 22. Placa de evaluación nRF52 DK.¹⁴

Sus características principales son:

- Ejemplos de software disponibles desde nRF52 SDK (*Software Development Kit*)
- Antena PCB integrada
- Conector para antena NFC externa
- Pines para medidas de consumo de potencia
- Interfaz de depuración *Segger J-Link*
- *Serial Wire Debug* (SWD)
- Alimentación 3'6V – 1'7V, batería y USB
- Memoria *Flash* de 512 KB y memoria RAM de 64 KB
- Tasa de transferencia de datos (*Data rates*) 1 Mbps/2Mbps
- Mapeo I/O configurable para E/S analógicas y digitales
- 3 x SPI maestro/esclavo
- 2 x I²C
- UART (RTS/CTS)
- ADC de 12 bits
- Interfaz de micrófono digital (PDM)

En la Figura 23, se muestra la asignación de pines de la placa de evaluación. Como puede observarse consta de: tres pines de alimentación V_{DD} (1'7V-3'6V) y un pin de alimentación 5 V, tres pines de masa (GND), un pin de RESET, pines para los cuatro leds y cuatro botones que incluye, pines para la comunicación UART (RX, TX, RTS y CTS), además de 32 pines I/O de uso general.

¹⁴ <https://www.nordicsemi.com/eng/Products/Bluetooth-low-energy/nRF52-DK>

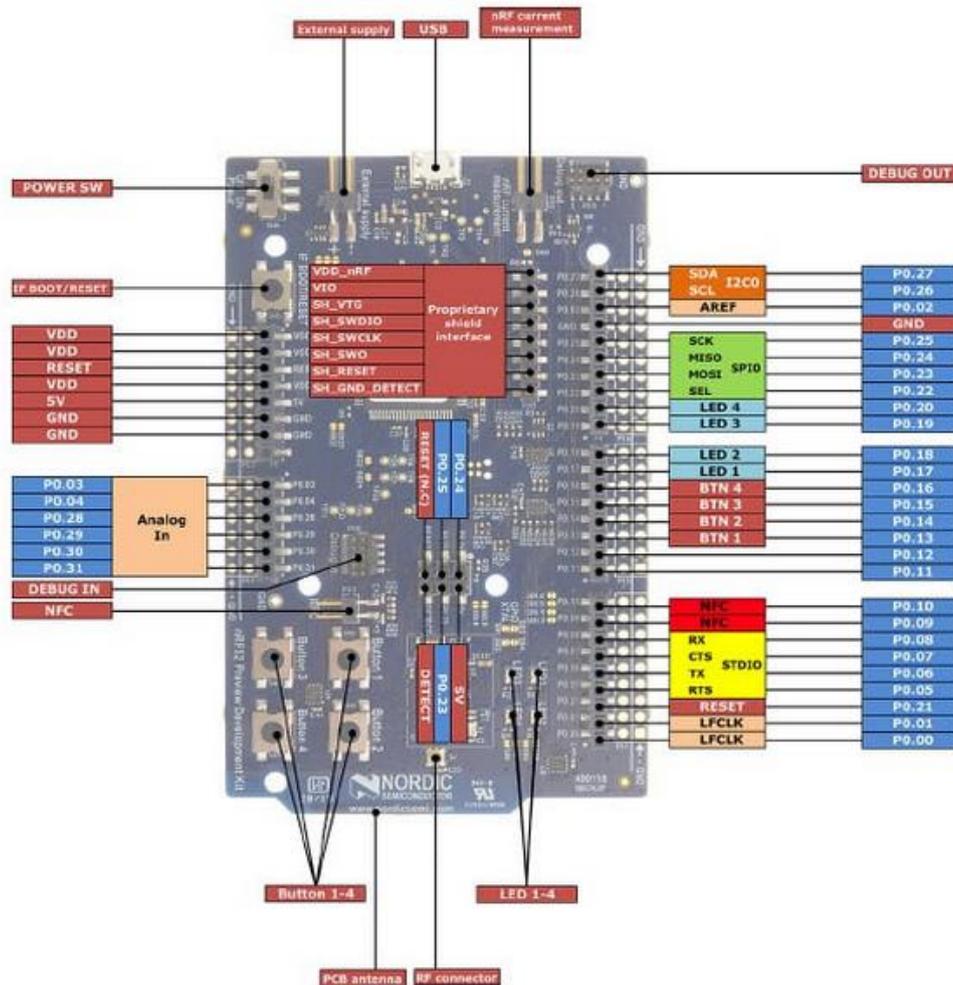


Figura 23. Asignación de pines de la placa de evaluación nRF52 DK.¹⁵

Para este proyecto se ha trabajado con la interfaz PDM para la adquisición de audio con el micrófono. El módulo PDM del microcontrolador es el periférico que se encarga de adquirir una señal de audio procedente de un micrófono MEMS con salida digital. Este bloque genera una señal de reloj y admite datos tanto de un único canal como de doble canal (*Left* y *Right*). Los datos se transfieren directamente a los *buffers*¹⁶ de la RAM usando *EasyDMA*, un módulo que implementan algunos periféricos para obtener acceso directo a la RAM.

Las características principales del módulo PDM son:

- Posibilidad de utilizar dos micrófonos configurados como par *left/right* compartiendo la misma línea de datos.
- Frecuencia de muestreo de salida de 16 kHz, 16 bit por muestra.
- Compatibilidad con *EasyDMA* para el almacenamiento en buffers de las muestras.
- Filtro de decimado (*Decimation filters*)

Como se muestra en la Figura 24, el módulo PDM es capaz de interconectar hasta dos micrófonos con la interfaz PDM. Incluye todos los elementos filtrantes necesarios para dar lugar a muestras PCM. Además, permite la transmisión continua de audio. Adquiere la señal del micrófono en formato PDM, la convierte a PCM y realiza un *Band-Pass Decimation Filter*. Al final obtenemos una señal PCM de 16 bits/muestra a 16 kHz.

¹⁵ <https://os.mbed.com/platforms/Nordic-nRF52-DK/>

¹⁶ Espacio de memoria en el cual se almacenan datos de forma temporal y generalmente para un único uso.

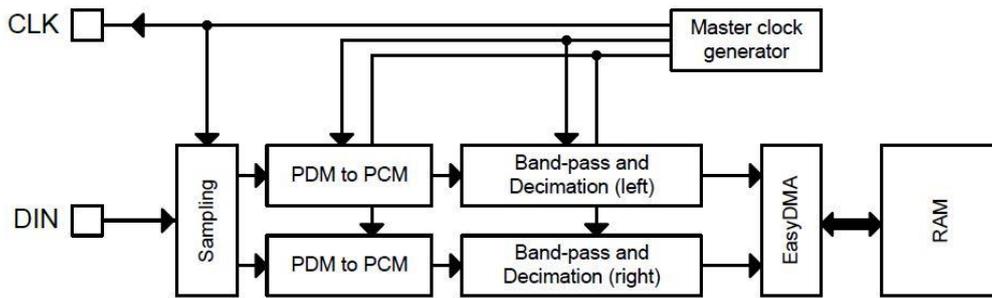


Figura 24. Módulo PDM. [25]

La interfaz PDM, presenta una serie de registros de los cuales toma los datos para realizar sus operaciones. A la hora de configurar los parámetros iniciales de la PDM, se emplean los siguientes registros: PDMCLKCTRL con el campo FREQ para fijar la frecuencia del reloj, MODE con el campo OPERATION para determinar si es estéreo (dos micrófonos) o mono (un micrófono) y el campo EDGE que indica si el muestreo se hace en los flancos de subida (*LeftRising*) o en los flancos de bajada (*LeftFalling*) del reloj.

Para este proyecto se ha usado una frecuencia de reloj de 1'032 MHz, que es la fijada por defecto y se encuentra dentro del rango de frecuencias de entrada que acepta el modelo de micrófono seleccionado (1MHz – 3'25MHz). Al pasar por el filtro de decimado, esta frecuencia de reloj de la PDM, se divide entre 64, dando lugar a la señal PCM de 16'125 kHz (≈ 16 kHz).

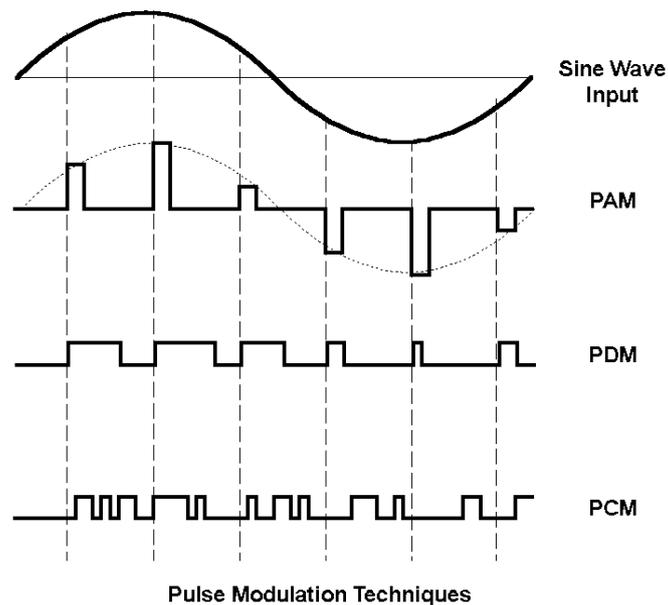


Figura 25. Cronograma de una señal sinusoidal analógica tras convertir a formato PCM y a formato PDM.¹⁷

Como puede verse en la Figura 25, la diferencia entre PCM y PDM reside, en que la codificación PCM asigna valores precisos de amplitud a cada muestra de la señal de audio, mientras que en la codificación PDM la amplitud de la señal está determinada por la densidad de pulsos positivos y negativos registrados en el tiempo.

¹⁷ <http://www.tecnodelinglesalcastellano.com/2011/01/que-diferencia-hay-entre-todos-esos.html>

La operación es mono, ya que se emplea un único micrófono digital y el muestreo se hace en los flancos de bajada (*LeftFalling*). Si se utilizasen dos micrófonos, uno de ellos se configuraría como *left* y el otro como *right* (L/R activo a nivel alto y a GND respectivamente).

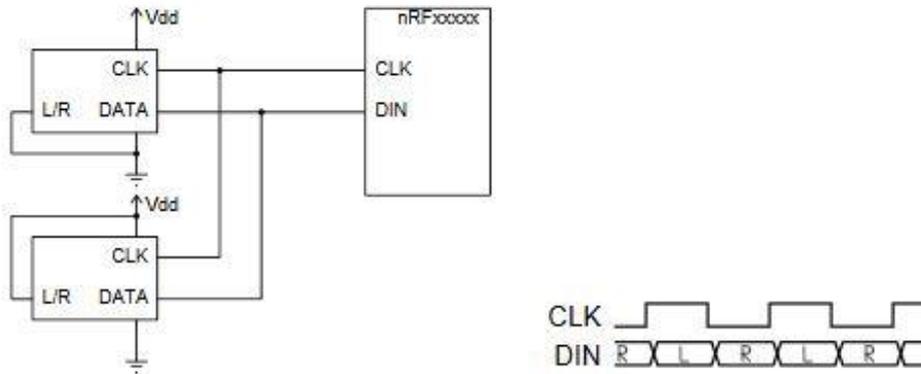


Figura 26. Configuración en modo stereo. [25]

En la Figura 26, se muestra la configuración en estéreo, en el caso de utilizarse dos micrófonos. Ambos micrófonos comparten la línea de datos y se recogen de forma alterna muestras *left* y *right*, en los flancos de subida y en los flancos de bajada respectivamente.

En la Figura 27 y en la Figura 28, se muestran las dos posibles configuraciones de hardware en modo de operación mono, es decir, con un único micrófono.

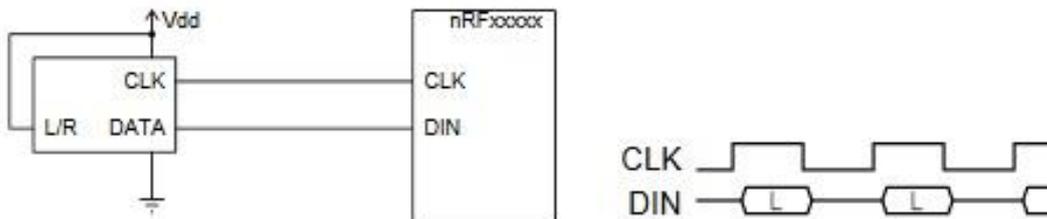


Figura 27. Configuración en modo mono y micrófono left. [25]

De esta forma, se configuraría el micrófono como *left*, conectando el pin de control L/R a la alimentación (V_{DD}) y el muestreo de la línea de datos se haría en los flancos de subida (*LeftRising*).

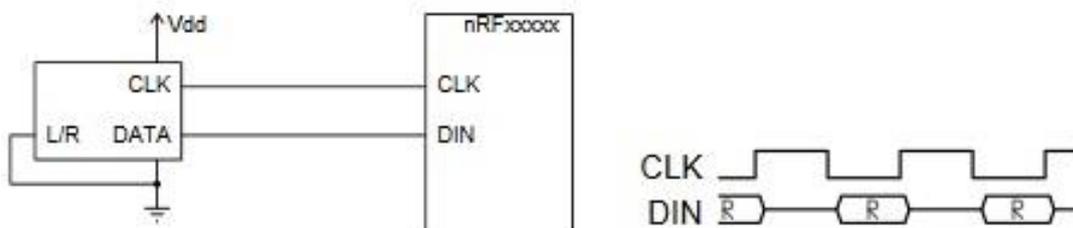


Figura 28. Configuración en modo mono y micrófono right. [25]

Y de forma inversa, si se configura el micrófono como *right*, conectando L/R a GND el muestreo tendría lugar en los flancos de bajada (*LeftFalling*). Así se ha configurado el micrófono utilizado en el proyecto, de modo que durante la adquisición de audio la memoria almacena muestras sucesivas de 16 bits recogidas en los flancos de bajada del reloj.

La operación del módulo se basa en que cada flujo de bits se pasa por un filtro digital que convierte dicho flujo de PDM a PCM de 16 bits. Además, el módulo utiliza *EasyDMA* para almacenar las muestras que salen de los filtros en un buffer en la RAM. Para asegurar un muestreo continuo, en la aplicación se debe actualizar el puntero a la dirección de destino de la RAM una vez se haya llenado el buffer anterior. La continuidad de la transferencia puede iniciarse o pararse mandando las tareas START y STOP. La tarea STOP será efectiva y generará el evento STOPPED una vez que se haya terminado la transferencia del actual flujo de bits. Una vez detenida la transferencia, todos los datos muestreados hasta el momento están disponibles en la RAM.

3.2.2. FUNCIONES

Para facilitar la integración del código correspondiente al funcionamiento del micrófono en el *firmware* del colgante, se han implementado una serie de funciones que conforman los *drivers* del micrófono mediante un archivo *C Header* (*uphone.h*) y un archivo *C Source* (*uphone.c*). De esta forma, se pueden incluir fácilmente en el programa del colgante y usar sus funciones, añadiendo los correspondientes *include*, reduciendo al máximo el volumen de código en el *main*. Además, estos archivos deben incluir los *drivers* que facilita el SDK referidos a la interfaz

En la Tabla 2 se explican dichas funciones y a su vez a las que llaman.

Función	Descripción	Función a la que llama de los drivers PDM
<i>uphone_init</i>	Configura los pines clk del reloj y data de la línea de datos, los cuales se pasan como parámetros en la función e inicializa toda la configuración del módulo PDM a los valores fijados (mode, edge, clock_freq, etc.)	<i>nrf_drv_pdm_init</i>
<i>uphone_uninit</i>	Deshabilita el módulo PDM, parando el muestreo aun si está en proceso	<i>nrf_drv_pdm_uninit</i>
<i>uphone_start</i>	Habilita el módulo PDM e inicia el muestreo	<i>nrf_drv_pdm_start</i>
<i>uphone_stop</i>	Detiene el muestreo de la interfaz PDM después de que termine la transmisión del actual bloque de datos	<i>nrf_drv_pdm_stop</i>
<i>uphone_buffers_set</i>	Se le pasa como parámetros los punteros a los buffers que se van a utilizar durante el muestreo y su tamaño (número de muestras). Se han utilizado tres, que es el mínimo número de buffers necesario para poder mantener la continuidad en la adquisición del audio, ya que cada vez que se esté llenando el buffer actual debe mandarse el siguiente inmediato.	

Tabla 2. Funciones de los drivers del micrófono.

Además, se crean dos estructuras: una para la configuración de los pines *clk* y *data*, y otra para la configuración de los buffers que se usan durante el muestreo, incluyendo su tamaño y sus respectivas variables *flag* que indican cuando están llenos y listos para su procesamiento.

3.2.3. FLUJOGRAMA

En este apartado se muestra un flujograma del *firmware* empleado como versión inicial y a partir del cual se han hecho modificaciones para adaptarlo a las pruebas realizadas en la cámara anecoica y a la propuesta de *UC3M4Safety*. Este flujograma constituye una representación gráfica de la línea de pasos de las acciones que conlleva el proceso de adquisición de audio en el microcontrolador.

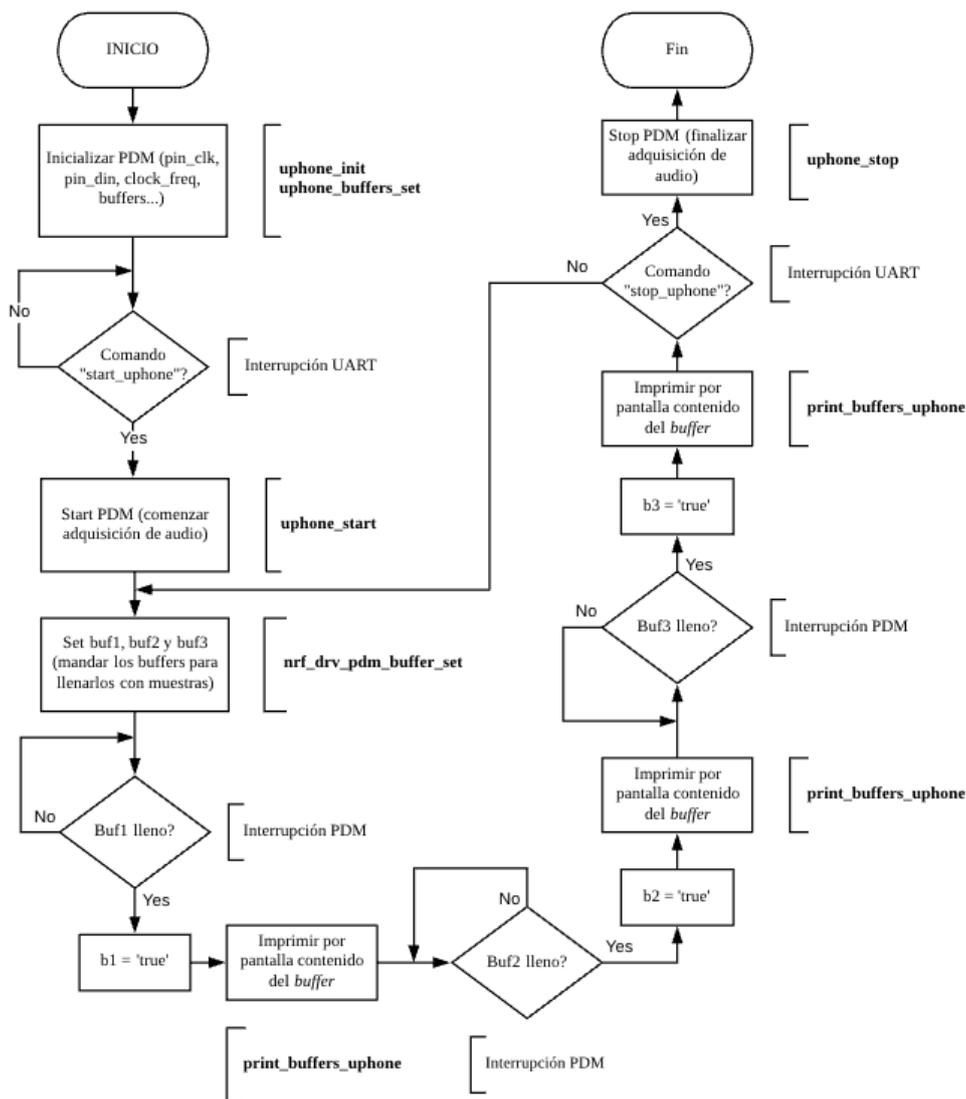


Figura 29. Flujograma del proceso de adquisición de audio.

Siguiendo el hilo que muestra la Figura 29, después de la inicialización del módulo PDM, el comienzo del muestreo se produce cuando se introduce por un terminal serie el comando “start_uphone” y el micro lo recibe a través de la *UART (Universal Asynchronous Receiver-Transmitter)*, desencadenando la llamada a la función *uphone_start*, que a su vez accede a *nrf_drv_pdm_start* y genera el evento *NRF_PDM_EVENT_STARTED*. A continuación, el *driver* PDM solicita un *buffer* mediante el evento *buffer_requested*, de modo que se le pasa mediante la función *nrf_drv_pdm_buffer_set*. Inmediatamente a esto, ya iniciado el proceso de muestreo, el *driver* solicita el siguiente *buffer*, que se termina de llenar después del actual. A partir de aquí mientras se ejecuta el muestreo, el controlador libera cada *buffer* lleno y solicita el próximo.

Una vez que un *buffer* se llena completamente con muestras, se crea a través de una interrupción el evento *NRF_PDM_EVENT_END*, que pone a ‘1’ los *flags (b1, b2 y b3)* correspondientes a cada *buffer* lleno. Cada vez que se llenan los tres *buffers* utilizados en el proceso de muestreo, se imprime su contenido por pantalla con la función *print_buffers_uphone* y se mantiene la continuidad de la adquisición del audio volviendo a mandar los *buffers* tras haberse liberado, o se detiene la grabación al introducir el comando “stop_uphone” por el terminal y recibirlo por la *UART*, parando el muestreo y deshabilitando el módulo PDM mediante la función *uphone_stop* que accede a *nrf_drv_pdm_stop*.

En esta versión inicial, para introducir los comandos 'start_uphone' y 'stop_uphone' se utiliza *Termite*. Y para imprimir por pantalla el contenido de los *buffers* y guardarlo en un archivo *txt*, *Serial Oscilloscope*.

- *Termite* es un terminal RS232 con una interfaz sencilla similar a la de los programas de *Messenger*, con una ventana donde se muestran todos los datos recibidos y una línea para escribir y destaca por su fácil instalación y utilización.
- *Serial Oscilloscope* funciona como un terminal serie básico que además de imprimir los bytes recibidos, es compatible con cualquier secuencia de valores terminados por un carácter de nueva línea ('\r') representándolos como variables dentro de canales de hasta tres osciloscopios. Además, tiene la opción *Log to file*, que guarda los datos que imprime por el terminal en un archivo *csv* (formato separado por comas). Este archivo que se obtiene, es fácil convertirlo en *txt* para llevarlo a *matlab* como una variable y reproducirlo o hacer su *FFT (Fast Furier Transform)*.

El tamaño de los *buffers* usados durante el muestreo, o lo que es lo mismo el número de muestras que contiene cada *buffer*, se puede variar según el valor que se le pasa al parámetro *size* de la función *uphone_buffers_set*. En este proyecto se han empleado *buffers* de tamaño 6000 muestras cada uno conforme a los cálculos expuestos en la Tabla 3.

Frecuencia de muestreo F_s	$F_s = 1'032 \text{ MHz}$	Reloj del módulo PDM
Frecuencia de muestreo F_s'	$F_s' = \frac{F_s}{64} =$ $= \frac{1'032 \text{ MHz}}{64} = 16125 \text{ KHz}$	Tras pasar por el Decimator Filter
Tiempo en el que se recoge una muestra o dato t_s'	$t_s' = \frac{1}{F_s'} =$ $= \frac{1}{16125 \text{ KHz}} = 62'01 \mu s$	Cada muestra son 2 byte, puesto que el contenido de los buffers se ha declarado como una variable de tipo int16
Número de muestras recogidas en un segundo	$\frac{1000000 \mu s}{62'01 \mu s} = 16125 \frac{\text{muestras}}{s}$	
Memoria ocupada en un segundo de audio	$16125 \frac{\text{muestras}}{s} \cdot 2 \frac{\text{bytes}}{\text{muestra}} =$ $= 32250 \frac{\text{bytes}}{\text{segundo}} \approx 31'5 \text{ KB}$	

Tabla 3. Cálculos para estimar el tamaño de los buffers en función de la memoria del microcontrolador.

Como la memoria RAM del microcontrolador es de 64 KB, no se puede almacenar mucho más de dos segundos de audio en total mediante los *buffers*, ya que con un segundo de audio se ocupa prácticamente la mitad de la memoria disponible. Por tanto, el tamaño de los tres *buffers* es de 6000 muestras, lo que da un total de 18000 muestras, que es poco más de un segundo de audio. Ya sea de forma alámbrica o inalámbrica, si queremos grabar más de un segundo de audio, debemos liberar lo grabado lo antes posible. De esta forma, es necesario comprimir para no saturar el canal.

3.3. COMPRESIÓN/DESCOMPRESIÓN

Los *códecs* tienen como función comprimir y descomprimir audio mediante la codificación y decodificación respectivamente, del archivo. Su objetivo principal es reducir el tamaño de los archivos multimedia para que ocupen el menor espacio posible y facilitar su transferencia a través de Internet. Para los diferentes formatos de audio, tales como MP3, WMA, WAVE, etc. se requieren distintos tipos de *códecs* para que la compresión funcione adecuadamente. Por ejemplo, MP3 o WMA se conocen como *códecs* con pérdida puesto que no conservan la total calidad de lo que convierten. Por otro lado, existen *códecs* sin pérdida como WAVE o FLAC, que permiten transmitir en alta calidad el audio completo. Sin embargo, los *códecs* con pérdida son los más utilizados debido a que requieren menos tiempo, recursos y memoria para transferir el audio, lo cual compensa la mínima pérdida de calidad.

En este Trabajo Fin de Grado se ha considerado necesaria la compresión de los datos obtenidos mediante el micrófono digital para su envío de forma adecuada mediante comunicación inalámbrica. El protocolo de comunicación elegido, *Bluetooth® Low Energy-BLE*, no está diseñado para el envío de forma continua (audio en *streaming*), por lo que ha sido necesario reducir el tamaño de los paquetes a enviar.

El proceso de compresión se realiza en el microprocesador dentro del colgante (Figura 30) y tiene como propósito almacenar el audio temporalmente, de forma que ocupe menos espacio, y especialmente favorecer el envío continuado del mismo mediante un protocolo inalámbrico. El objetivo final es el procesamiento de este audio en una aplicación del teléfono móvil para identificar situaciones de peligro mediante técnicas de reconocimiento del habla con la voz de la usuaria o para usarlo posteriormente como testimonio judicial de una posible agresión sexual. Figura 30.

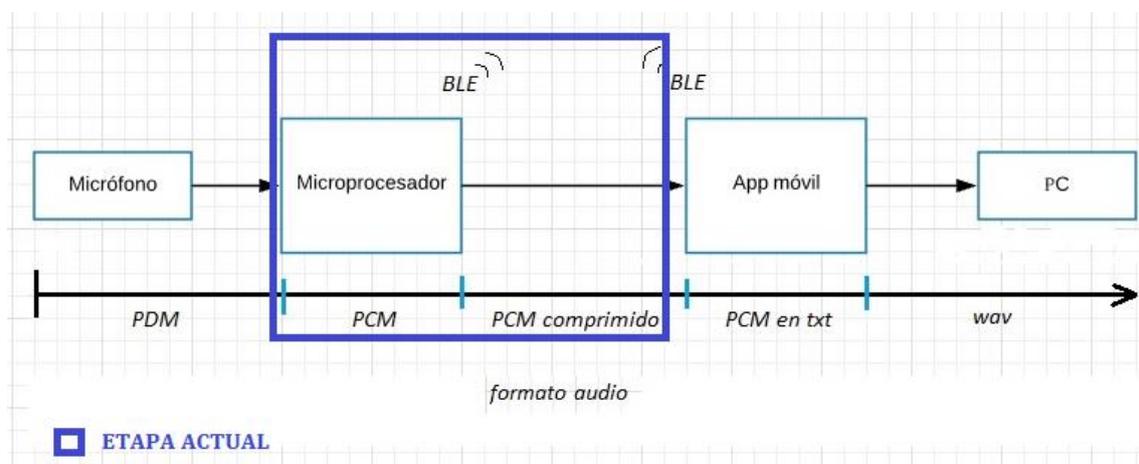


Figura 30. Etapa de compresión del audio.

Para implementar la compresión del audio se ha utilizado el códec de audio Opus, un recurso totalmente abierto, gratuito y versátil. Opus es capaz de manejar una extensa gama de aplicaciones de audio, tales como *VoIP (Voice over IP)*, videoconferencias, chat en juego o actuaciones de música en vivo, así como para su utilización en aplicaciones de transmisión y almacenamiento. Está pensado principalmente para comunicaciones por Internet en tiempo real. Opus destaca por su baja latencia, tiempo que tarda en codificar o decodificar, y por su *bitrate* variable, cantidad necesaria de bits para codificar un segundo de audio y que afecta de manera directa a la calidad de este. Hace tiempo que, en el ámbito de los códecs de audio, MP3 (*MPEG-1/2 Audio Layer III*) predomina, sin embargo presenta la desventaja de que no es un *software* de libre uso. Por ello *Mozilla*, *Google* y *Skype* decidieron crear como alternativa Opus. Este está estandarizado por IETF (*Internet Engineering Task Force*).

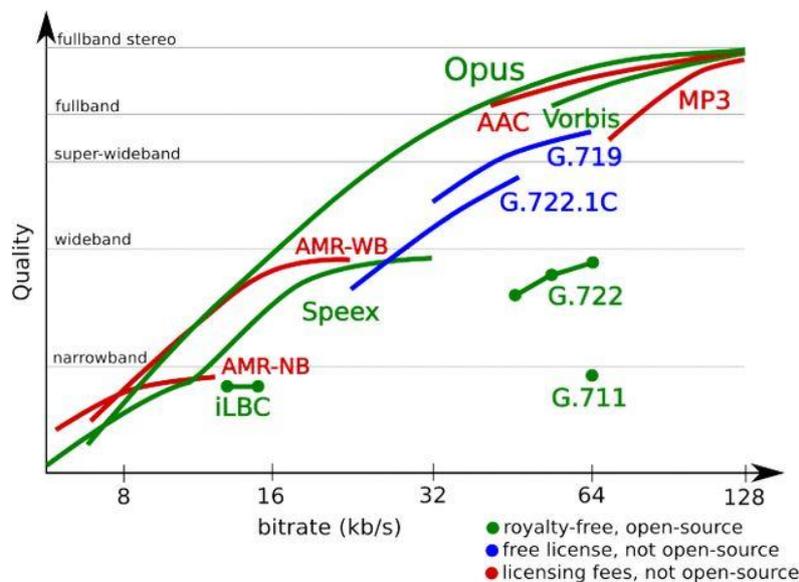


Figura 31. Calidad frente a bit-rate de diferentes códecs. ¹⁸

En la Figura 31 se muestra la calidad que se obtiene, en cuanto al ancho de banda frente al rango de *bitrate*, cuando se utilizan algunos códecs actuales. Además se representa mediante una leyenda según su accesibilidad.

Las funciones que admite Opus son las siguientes:

- *Bit-rate* de 6 kb/s a 510 kb/s
- Frecuencia de muestreo de 8 kHz a 48 kHz
- Habla y música
- Mono y estéreo
- Tamaño de la trama de 2'5 ms a 60 ms
- Ajuste dinámico del *bitrate*, ancho de banda y tamaño de la trama

La configuración usada para la frecuencia de muestreo es de 16'125 kHz, que es lo que recibe de la salida de audio en PCM del microprocesador. La frecuencia del ancho de banda, la cual está relacionada con lo que filtra el compresor para reducir la información, se ha configurado a 6 kHz. Con esta configuración se están teniendo calidades de voz equivalentes a una llamada telefónica.

¹⁸ www.genbeta.com/multimedia/opus-el-codec-de-audio-libre-creado-por-mozilla-google-y-skype

Para este proyecto se han empleado dos funciones de Opus, *opus_encode* y *opus_decode*, para comprimir y descomprimir el audio respectivamente. Para ejecutar y probar el código proporcionado por Opus se ha usado *Visual Studio*, un entorno de desarrollo integrado para sistemas operativos Windows que soporta numerosos lenguajes de programación como C++, *Visual Basic*, *Java*, etc. que permite ver, editar y depurar cualquier tipo de código y así posteriormente generar aplicaciones para *Android*, *iOS* o *Windows*. Las pruebas que se han realizado han sido las siguientes:

- Compresión y descompresión en PC (*opus_encode* y *opus_decode*): antes de implementar el código correspondiente en el microprocesador junto con las funciones del micrófono, se ha llevado a cabo una prueba básica de compresión y descompresión de varios audios, para así verificar que tanto la codificación como la decodificación se realizan adecuadamente por separado.
- Compresión en microprocesador: una vez conseguida la correcta grabación del audio con el micrófono mediante las funciones de los *drivers*, se ha añadido la función de Opus *opus_encode*, para comprimir y así reducir el tamaño del audio, que posteriormente se envía por *Bluetooth*[®] al móvil.
- Compresión en microprocesador + Descompresión en PC: es el procedimiento que se ha llevado a cabo en las grabaciones realizadas en la cámara anecoica, mediante transmisión de audio por interfaz inalámbrica. El proceso de adquisición del audio, junto con su compresión, se llevan a cabo en el microprocesador, el audio ya codificado se envía por *Bluetooth*[®] *Low Energy-BLE* al teléfono móvil guardándose en un fichero de texto y finalmente se descomprime el audio en el PC generando su correspondiente archivo WAV. De esta manera se ha ido grabando la base de datos, para el micrófono empleado en este proyecto, proporcionada por TSC para que posteriormente se entrenen los algoritmos encargados de detectar el estrés en la voz.

En la Tabla 4 se nombran y se explican brevemente los parámetros de las funciones proporcionadas por Opus para comprimir y descomprimir el audio [26].

	Tipo de parámetro	Parámetro	Descripción
<i>opus_encode</i>	entrada	<i>st</i>	Estado del encoder
	entrada	<i>pcm</i>	Señal de entrada, audio a comprimir. Variable de tipo int16.
	entrada	<i>frame_size</i>	Número de muestras por canal en la señal de entrada.
	salida	<i>data</i>	Señal de salida, audio comprimido. Variable de tipo unsigned int8.
	entrada	<i>max_data_bytes</i>	Tamaño de memoria asignado para el audio de salida.
<i>opus_decode</i>	entrada	<i>st</i>	Estado del decoder.
	entrada	<i>data</i>	Señal de entrada, audio a descomprimir. Variable de tipo unsigned int8.
	entrada	<i>len</i>	Número de bytes del audio de salida.
	salida	<i>pcm</i>	Audio de salida, descomprimido. Variable de tipo int16.
	entrada	<i>frame_size</i>	Número de muestras por canal en la señal de entrada.

Tabla 4. Parámetros de las funciones *opus_encode* y *opus_decode*.¹⁹

¹⁹ www.opus-codec.org/docs/

4. PUESTA EN MARCHA

En este capítulo se introducen los recursos que se han añadido a la adquisición de audio para realizar las pruebas de grabación en la cámara anecoica y para desarrollar la propuesta de UC3M4Safety, *Bindi*.

En este punto, se le añade al *firmware* inicial mostrado en el flujograma de la Figura 29, el código correspondiente al compresor (*opus_encode*) y al protocolo de *Bluetooth*[®]. De modo que el flujograma final del *firmware* encargado de la adquisición de audio y su envío comprimido mediante un protocolo inalámbrico tal como *Bluetooth*[®], se muestra a continuación en la Figura 32.

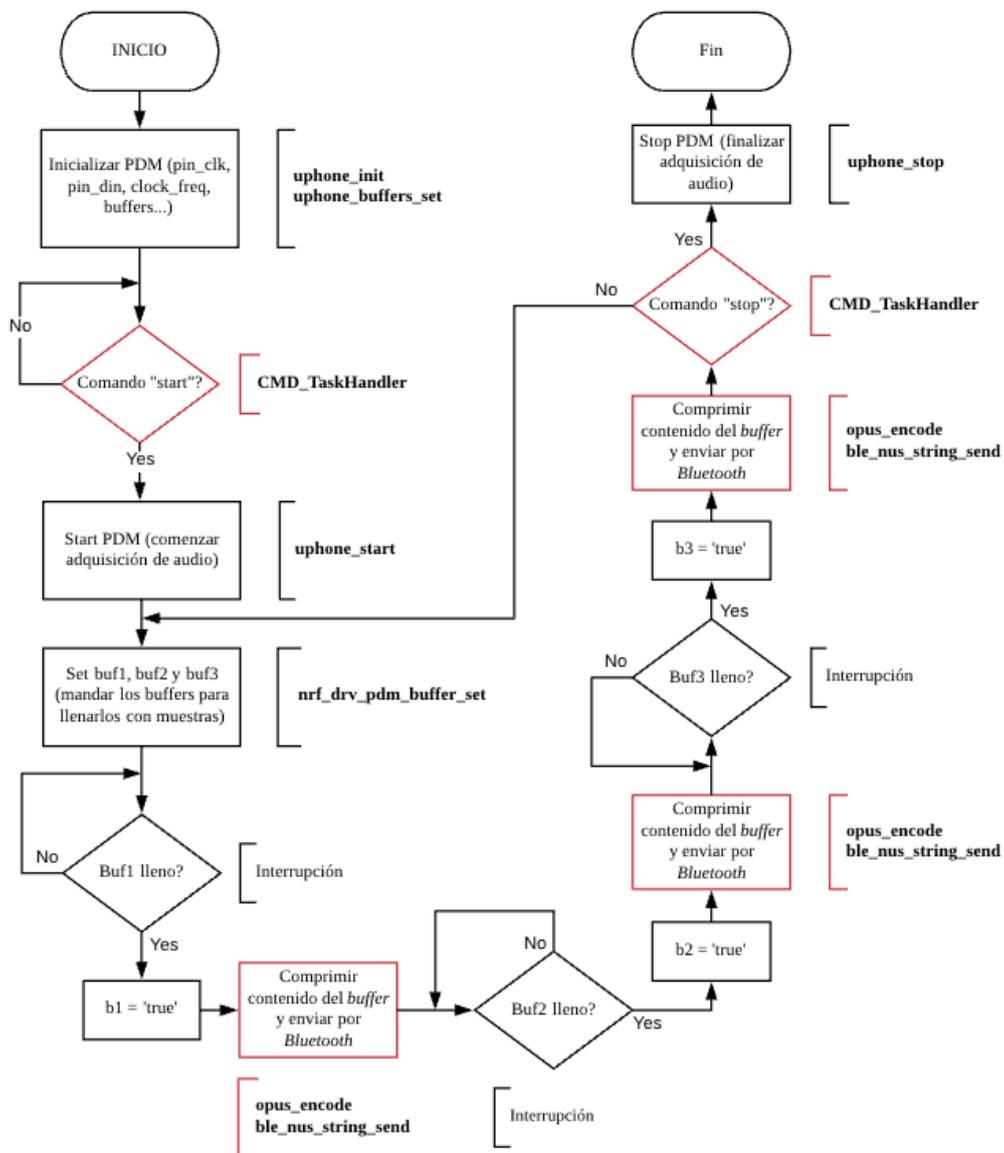


Figura 32. Flujograma del proceso de adquisición, compresión y envío del audio.

Además, como puede observarse en este flujograma, el envío de los comandos que inician y detienen la adquisición de audio se realiza con una función diferente, *CMD_TaskHandler*, que permite recibir los comandos tanto a través de la UART introduciéndolos por un terminal serie (así se hacía inicialmente cuando solo se tenía la adquisición del audio), como a través de *Bluetooth*[®]. Puesto que nunca se van a recibir datos de ambas vías a la vez, en las grabaciones de la cámara se ha utilizado esta función para enviar los comandos a través de la *app* del móvil por *Bluetooth*[®]. Los comandos han pasado a ser: “*start*” para iniciar la grabación, y “*stop*” para detenerla.

4.1. BLUETOOTH

Bluetooth[®] *Low Energy (BLE)* también conocido en ocasiones como *Bluetooth Smart* se presentó como parte de la especificación de *Bluetooth 4.0*. Existen una amplia gama de protocolos *wireless* para su uso en *IoT*, sin embargo, lo que hace que BLE sea especialmente interesante consiste en que es el más sencillo a la hora de implementar la comunicación entre dispositivos pequeños y una aplicación en cualquier plataforma móvil existente (*iOS*, *Android*, etc.). Por ello, se ha utilizado este protocolo en *Bindi* para la transmisión del audio grabado por el colgante hasta la aplicación en *Android* del móvil. La *app* del móvil recibe el audio comprimido que le envía el microcontrolador y crea un fichero *txt* donde lo almacena. El fichero se cierra cuando el micrófono para de grabar.

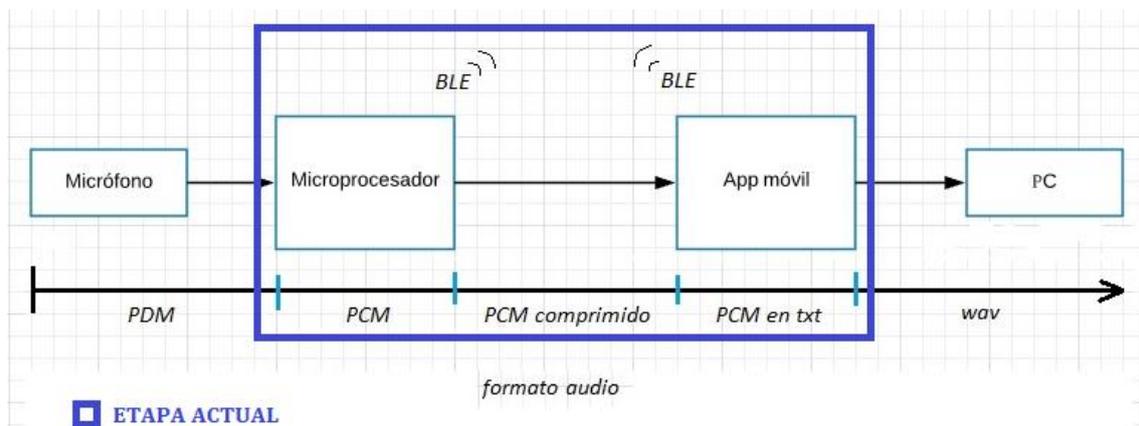


Figura 33. Etapa de envío del audio.

Para introducir el envío del audio por *Bluetooth*[®] *Low Energy* (Figura 33), se añade al *firmware* del colgante el *SoftDevice S132*, que constituye una pila de protocolo *BLE* central y periférica la cual proporciona una interfaz de programación de aplicaciones (API) para construir soluciones de bajo consumo de energía para el *SoC nRF52832*. Cabe destacar, que la implementación del *stack* de *Bluetooth*[®] ha sido realizada en el proyecto *Bindi* por Manuel Canabal, investigador del Departamento de Tecnología Electrónica.

Para el envío de los *buffers* por *Bluetooth*[®] se utiliza la función *ble_nus_string_send*, la cual se muestra en el flujograma del apartado anterior, que incluye el *Software Development Kit* en sus bibliotecas. En la Tabla 5 se exponen los parámetros de la función.

Tipos de parámetros	Nombre	Descripción
entrada	<i>p_nus</i>	Puntero a la estructura NORDIC Uart Service.
entrada	<i>p_string</i>	Cadena de datos que se envían.
entrada/salida	<i>p_length</i>	Tamaño de la cadena, número de bytes.

Tabla 5. Parámetros de la función *ble_nus_string_send*. [25]

El *NORDIC Uart Service*, está basado en un servicio GATT (*Generic Attribute*) con características RX y TX. El perfil GATT, define la forma en la que dos dispositivos *Bluetooth® Low Energy (BLE)* se envían y reciben mensajes, es decir, constituye una estructura jerárquica de datos presente en los dispositivos conectados. GATT establece que las conexiones son exclusivas, un periférico BLE solo puede conectarse a un único dispositivo central a la vez. En *Bindi*, el colgante se conecta mediante este protocolo (BLE) únicamente al teléfono móvil.

Mediante el módulo que incluye la función *ble_nus_string_send*, se puede crear un servicio para enviar y recibir cadenas de texto ASCII de un dispositivo a otro [25], en este caso, del colgante al teléfono móvil, o si se quisiera viceversa. Por ello, con *CMD_TaskHandler* (Figura 32. Flujograma del proceso de adquisición, compresión y envío del audio. Figura 32), se puede gestionar la recepción de los comandos enviados por *Bluetooth®* con la *app* del móvil además de por la UART.

En el envío del audio por *Bluetooth®*, para determinar el número de paquetes a enviar, se ha tenido que tener en cuenta algunos de los parámetros establecidos para la compresión y se ha modificado el tamaño de los *buffers* del muestreo PDM, a través de uno de los parámetros de *uphone_buffers_set*, explicado anteriormente (Tabla 2), reduciéndolos a 320 muestras. Esta reducción en el número de muestras de cada *buffer* se ha estimado beneficiosa en tanto que:

- Al disminuir las muestras, se tienen *buffers* que requieren menos espacio en la memoria del micro.
- Su proceso de compresión también supone un ahorro de espacio, al ser más pequeños.
- En el peor de los casos, si se pierden paquetes durante el envío del audio por *Bluetooth®*, teniendo menor número de muestras por *buffer*, se pierde menos información.

Por otra parte, cabe destacar que, al tener *buffers* más pequeños se llenan antes, con su consiguiente solicitud del próximo *buffer* a llenar y todo ello implica que haya interrupciones con mayor frecuencia en el microcontrolador.

Los cálculos para el número de paquetes en el envío del audio se explican en la Tabla 6.

Nº de muestras	Tiempo	Tamaño	Nº de paquetes
320	$320 \cdot \frac{1}{16125 \text{ kHz}} \approx 20 \text{ ms}$	$15 \frac{\text{kbit}}{\text{s}} \cdot 20 \text{ ms} \approx 37 \text{ bytes}$	20 bytes 17 bytes

Tabla 6. Cálculos para el número de paquetes del envío.

Como ahora el tamaño de los *buffers* es de 320 muestras, el *frame_size* que se le pasa a *opus_encode* es 320 (explicado en Tabla 4.). Entonces, para cada *buffer* se realiza una compresión de 320 muestras, lo cual equivale a 20 ms de audio teniendo en cuenta la frecuencia de muestreo (Tabla 6). Como anteriormente se ha explicado, el compresor se configura con un *bit-rate* de 15 kbits/s. De modo que, si se comprimen 20 ms a una velocidad de 15 kbits/s, se tienen 37 bytes de audio comprimido. Ahora, el envío de los 37 bytes de audio por *Bluetooth*[®], se realiza en dos paquetes: primero uno de 20 bytes, y luego otro de 17 bytes.

4.2. RECEPCIÓN DE AUDIO (MÓVIL Y PC)

En este apartado se explica la última etapa del sistema, que consta de la recepción, en la *App* del teléfono móvil, del audio comprimido y su posterior descompresión en el PC. Figura 34.

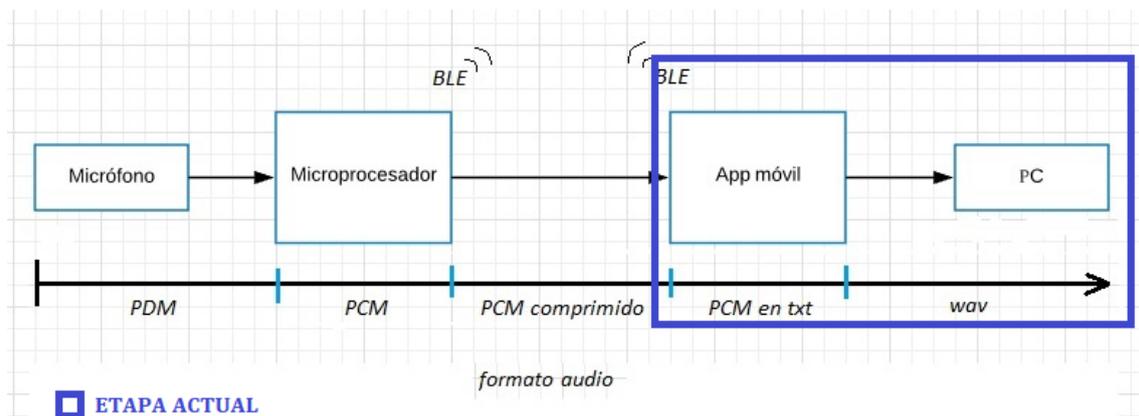


Figura 34. Etapa de descompresión del audio.

En esta etapa, no se ha desarrollado un *software* propio, sino un protocolo de operación que permitirá activar la grabación del micrófono, así como la recepción del audio grabado, su descompresión, si es necesaria, así como su conversión a otros formatos para su posterior procesado. Este protocolo consta de cinco pasos, los tres primeros se ejecutan en el teléfono móvil, donde principalmente se utiliza sólo una aplicación (*App*), y los otros dos en el PC; en este último se utilizan varias herramientas *software*, que sirven para MM.

Teléfono móvil (Aplicación → camara sin h) Figura 35

Esta aplicación está basada en otra llamada *nrf_uart*, la cual fue modificada por Manuel Canabal, investigador contratado en el Departamento de Tecnología Electrónica y miembro del equipo *UC3M4Safety*, para gestionar el almacenamiento de los datos en un fichero de texto.

1. Iniciar la conexión entre el colgante y el teléfono móvil.
 - I. Al entrar en la *app* lo primero que se solicita es activar el *Bluetooth*®. Una vez activado, la interfaz de la *app* es muy similar a la de un terminal serie, donde hay una casilla para datos y en el resto de la pantalla se visualiza la comunicación entre los dispositivos: lo enviado (TX) y lo recibido (RX). En la parte superior de la ventana, se encuentra la opción *Conectar*, que al seleccionarla provoca la aparición inmediata de una ventana emergente, mostrando los dispositivos *Bluetooth*® disponibles alrededor.
 - II. Una vez que encuentra el colgante, se selecciona y se establece la conexión. A partir de aquí, aparece el primer mensaje en pantalla informando que se ha establecido una conexión.
2. Realizar la grabación de audio.
 - I. Esperar tres segundos desde que se establece la conexión con el colgante antes de enviarle ningún comando, para que ambos dispositivos puedan establecer los parámetros iniciales de la comunicación y no haya problemas en el envío.
 - II. Enviar la orden de arranque de la grabación, comando *start*, que aparece en la pantalla como TX. Enseguida, se recibe el mensaje de que se ha creado un archivo *txt* donde se empiezan a guardar los datos, como RX.
 - III. Esperar el tiempo necesario para que se reproduzca todo el audio a grabar.
 - IV. Detener la adquisición de audio, comando *stop*, que al igual que con *start* se muestra por la pantalla de la comunicación, y se recibe un mensaje de que se ha cerrado el archivo *txt*, finalizando la grabación. Figura 35.

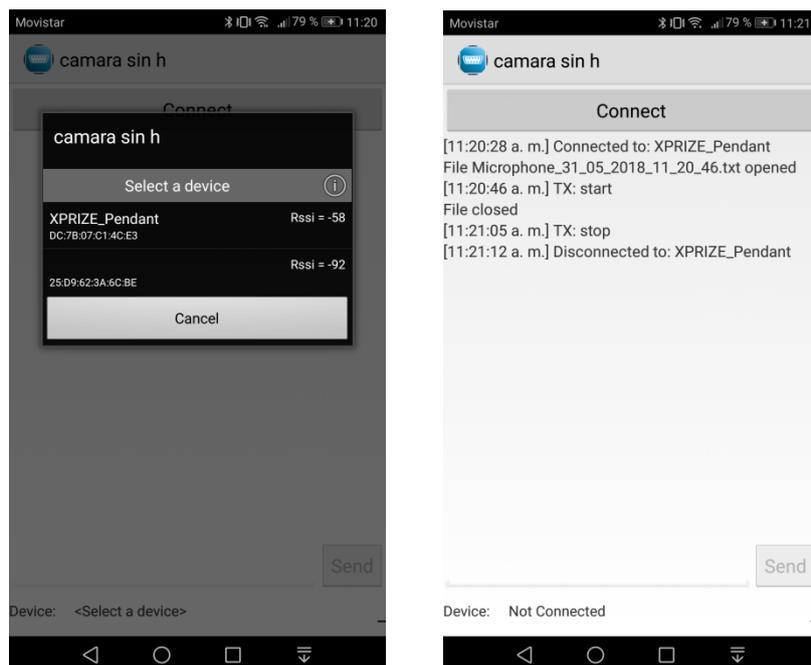


Figura 35. Interfaz de la *app* utilizada para el envío del audio por *Bluetooth*®.

Teléfono móvil (Gestor de archivos)

3. Descarga de ficheros de audio adquiridos al PC.
 - I. Pasar al PC los ficheros *txt* generados por la *app*, los cuales se crean en la carpeta *Descargas* del teléfono móvil y se nombran con el formato '*Microphone_dd_mm_aa_hh_mm_ss.txt*'.

Ordenador personal PC (Archivo ejecutable → *opus_decoder.exe*)

4. Descompresión de los ficheros de audio adquiridos y descargados en el PC.
 - I. En el PC, proceder a descomprimir las muestras almacenadas en el *txt* y convertirlas a un formato *wav* para su posterior reproducción. Se ha generado en *Visual Studio* a partir del código desarrollado para utilizar la función *opus_decode* (Tabla 4), un archivo ejecutable (*opus_decoder.exe*) que se podrá ejecutar en un ordenador cualquiera con sistema operativo *Windows* sin necesidad de tener instalado *Visual Studio*²⁰. Ejecutarlo a través de *Windows PowerShell*, una interfaz de consola o de línea de comandos mediante instrucciones (Figura 36).
 - II. Al programa de este ejecutable, se le ha añadido código para que cree un fichero *wav* donde almacene lo que descomprima del archivo *txt* de la *app* y además, cuando se le llame por el terminal junto con su nombre, se especifiquen tres parámetros más: el nombre del archivo *wav* donde se quiere almacenar el audio descomprimido, si existe dicho archivo se escribe en él y si no lo crea; el nombre del archivo *txt* del que se quiere descomprimir el audio; y un valor por el cual se multiplican todas las muestras del audio descomprimido para que al reproducirlo se escuche con mayor volumen. Este último parámetro simplemente se ha añadido por el hecho de que, como anteriormente se ha explicado, al descuantificar (dividir entre 2^{16}) la señal a la hora de reproducir el audio mediante la función *sound* de MATLAB para escucharlo nítido y al no multiplicar la señal por ningún valor del fondo de escala, el audio se escucha con un volumen bajito ya de por sí. Mediante la declaración *int main(int argc, char *argv[])*, en el código del programa del descompresor (*opus_decoder*), se pasan los parámetros del mismo a través de la línea de comandos de *Windows PowerShell*.
 - III. El primer argumento, el entero *argc*, comprende el número de argumentos que recibe el programa, en este caso cuatro: el nombre del ejecutable del programa (el primer argumento siempre se reserva para incluir el nombre del programa), el nombre del *wav* donde se almacena el audio de salida, el nombre del *txt* con el audio de entrada y un valor numérico para aumentar el volumen del audio al reproducirlo. Y el segundo argumento, **argv[]*, se trata de un puntero al *array* de *char* que contiene los parámetros que han sido pasados manteniendo su orden de escritura.

²⁰ Sin embargo, solo con el ejecutable sin disponer del proyecto que contiene el código del programa (en este caso, el descompresor), no se puede acceder a dicho código, solo se puede ejecutar

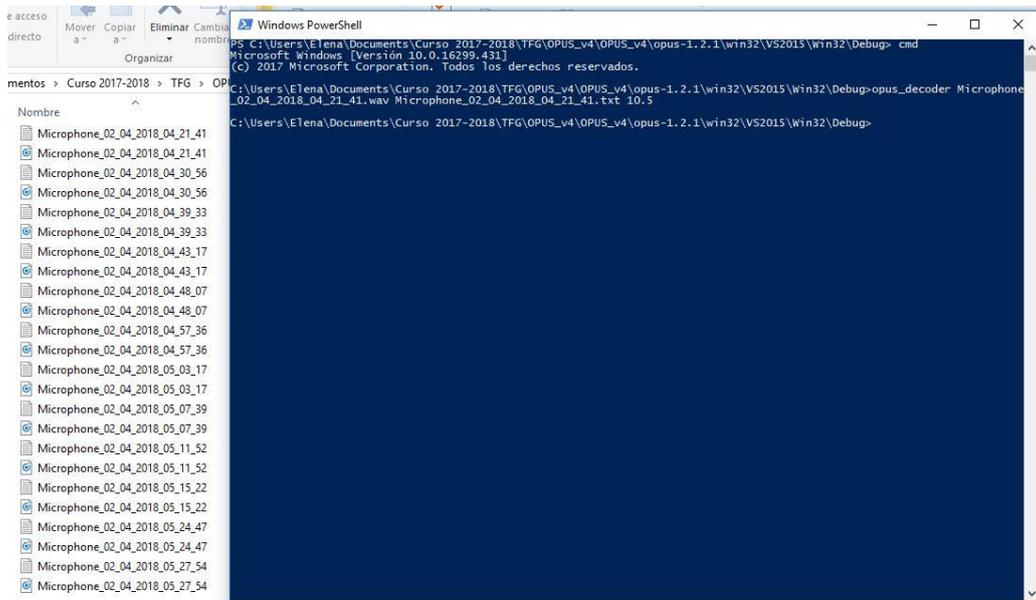


Figura 36. Proceso de descompresión mediante su ejecutable a través de la línea de comandos de Windows.

Para que no haya problemas a la hora de pasarle como parámetros del ejecutable el nombre del wav y del txt, deben encontrarse en la misma ubicación del propio ejecutable. En la Figura 36 puede verse que mediante la línea de comandos se abre el exe del descompresor, con nombre *opus_decoder*, que se le pasa con el resto de los parámetros. Si el wav aún no se ha creado, se creará igualmente en la misma carpeta que el ejecutable. También se aprecia como los sucesivos wav que se van creando tras descomprimir el audio de los respectivos txt obtenidos de las grabaciones de la base de datos en la cámara, se van guardando todos en la misma carpeta.

5. Reproducción de los audios ya descomprimidos y en formato wav.
 - I. Posteriormente, reproducir los audios para comprobar que en general estén correctos. Para ello, se ha utiliza el reproductor multimedia VLC, capaz de reproducir numerosos formatos tanto de audio como de vídeo y disponible en multitud de sistemas operativos. Figura 37.

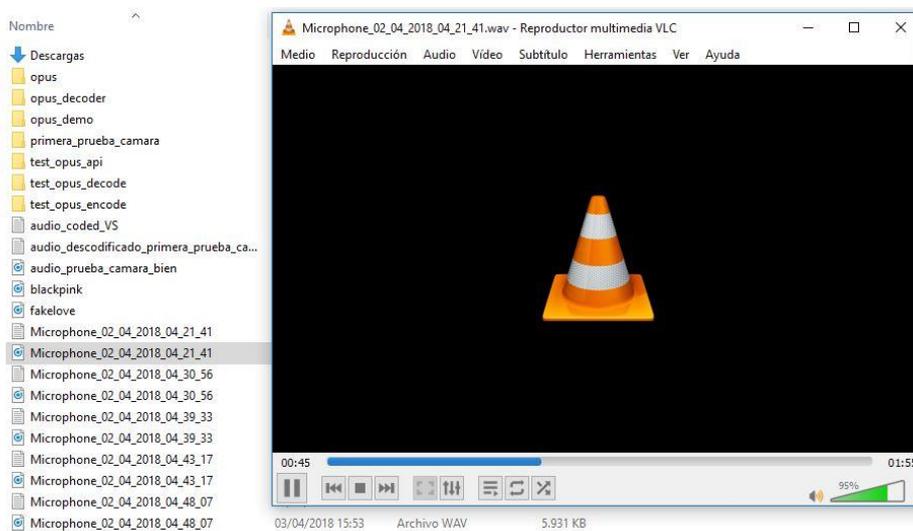


Figura 37. Reproducción de los audios a través de VLC.

4.3. MATLAB

En este proyecto se ha utilizado MATLAB®, la herramienta de *software* matemático que proporciona un entorno de desarrollo integrado con un lenguaje de programación propio y que está disponible para numerosas plataformas entre las cuales se encuentra *Windows*®. Ha sido de gran utilidad a la hora de verificar el funcionamiento adecuado de la adquisición del audio en el micrófono mediante el *firmware* inicial mostrado en el flujograma de la Figura 29. Para ello, se ha realizado una prueba de concepto simple: se ha grabado un tono a 440 Hz, que equivale a la nota musical 'La' y el audio resultante se ha llevado a MATLAB y se ha trazado, junto con su transformada de *Fourier* (FFT, *Fast Fourier Transform*) para ver sus componentes en frecuencia. Figura 38.

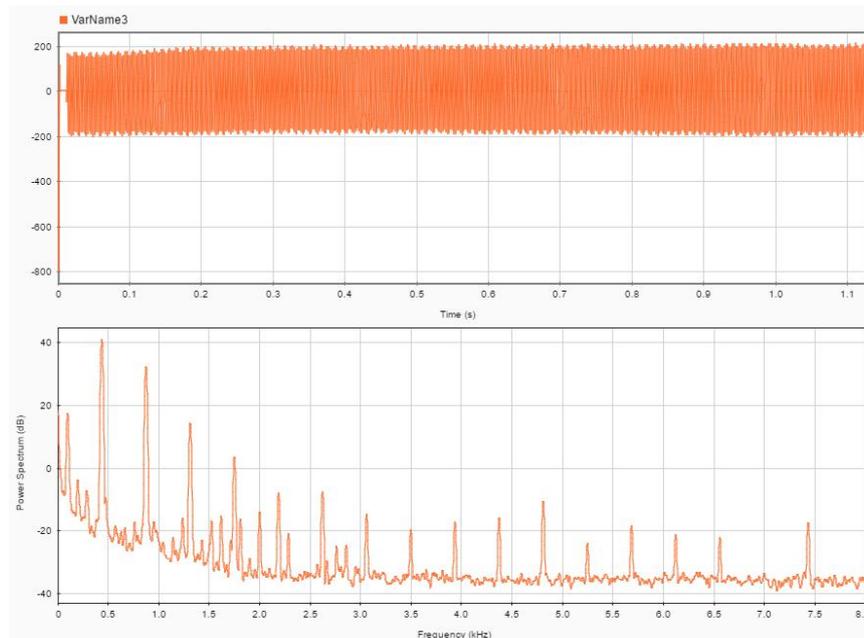


Figura 38. Seno a 440 Hz grabado con el micrófono.

Una vez que se visualizó que efectivamente la señal correspondía a un seno, que en función del volumen al que se emitiera la nota 'La', salía con mayor o menor amplitud, y sus componentes en frecuencia eran coherentes (armónico fundamental, segundo armónico...), se grabaron algunos tonos más a otras frecuencias para asegurar definitivamente que la adquisición del audio básica, sin comprimir ni enviar por *Bluetooth*®, se estaba haciendo correctamente.

También se utilizó MATLAB en la verificación de la compresión y descompresión. Tras añadir la compresión del audio después de su adquisición, como muestra el flujograma de la Figura 32, se hicieron algunas pruebas grabando tramos de música que una vez comprimidas en el microcontrolador, se descomprimían en el PC para reproducirlas y comprobar su validez.

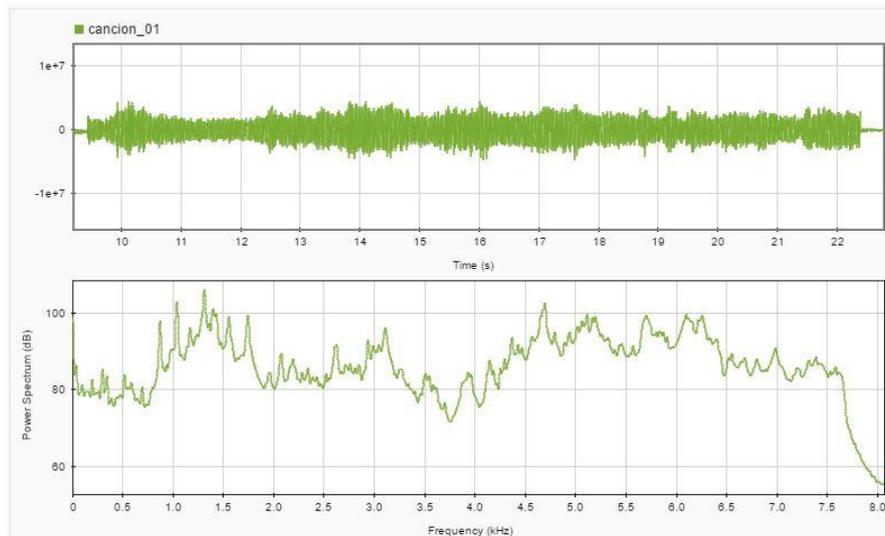


Figura 39. Audio de una canción tras descomprimirlo.

Para la reproducción de los audios, mediante las funciones que ofrece MATLAB se crearon archivos con formato WAV, lo cual sirvió para comprobar la calidad de lo grabado (Figura 39). Primero se crea el fichero WAV donde se quiere almacenar el audio, dándole un nombre con la orden $filename = 'FileWavName.wav'$. Una vez creado el fichero, se usan las dos funciones de la Tabla 7 para almacenar el audio en él y posteriormente reproducirlo.

<i>audiowrite(filename,y, F_s)</i>	Escribe una matriz de datos de audio, y , en el fichero $filename$ con una frecuencia de muestreo F_s . De modo que si a $filename$ se le asigna un fichero con un nombre concreto y formato WAV, se escriben los valores de y en dicho fichero WAV. ²¹
<i>sound(y, F_s)</i>	Reproduce por el altavoz la señal de audio y a una frecuencia de muestreo F_s . ²²

Tabla 7. Funciones de MATLAB para crear y reproducir un archivo de audio en formato WAV.

Si reproducimos los audios tal cual los obtenemos de las grabaciones, la señal se escucha distorsionada, ya que al ser datos en formato digital (0xAAAA) la amplitud es muy grande y los altavoces del PC los reproducen con distorsión. Por tanto, para escuchar el audio de forma nítida, por así decirlo, hay que descuantificar la señal: dividir entre 2^{16} sus valores (2^n , siendo n el número de bits), basándonos en los 16 bits/muestra del audio de salida en formato PCM.

²¹ <https://www.mathworks.com/help/matlab/ref/audiowrite.html>

²² <https://www.mathworks.com/help/matlab/ref/sound.html>

5. GRABACIONES

Como se ha mencionado anteriormente, una de las motivaciones de llevar este proyecto a cabo, es la optimización del sistema a la hora de realizar las pruebas y grabaciones precisas para entrenar los algoritmos que se incorporan en el *software* de *Bindi* como método adicional de detección del pánico. Puesto que en versiones posteriores de *Bindi*, se estudiará la posibilidad de detectar por medio del audio, la presencia de gritos o palabras clave, es completamente necesario llevar a cabo una gran variedad de grabaciones de la forma más rápida y óptima posible.

En concreto, para este proyecto se ha llevado a cabo la grabación completa de la base de datos que incluye numerosos audios en los que el hablante está sometido a distintos niveles de estrés, presente en la voz (Figura 40). De esta forma, se pueden entrenar los algoritmos para el audio recogido por el modelo de micrófono MEMS del colgante. Las grabaciones se han llevado a cabo en dos días diferentes: el primer día se grabó el *set 1*, un conjunto de dieciséis audios de duración total 1 hora y 20 minutos; y el segundo día se grabó el *set 2*, un total de veinte audios de duración 1 hora y media.

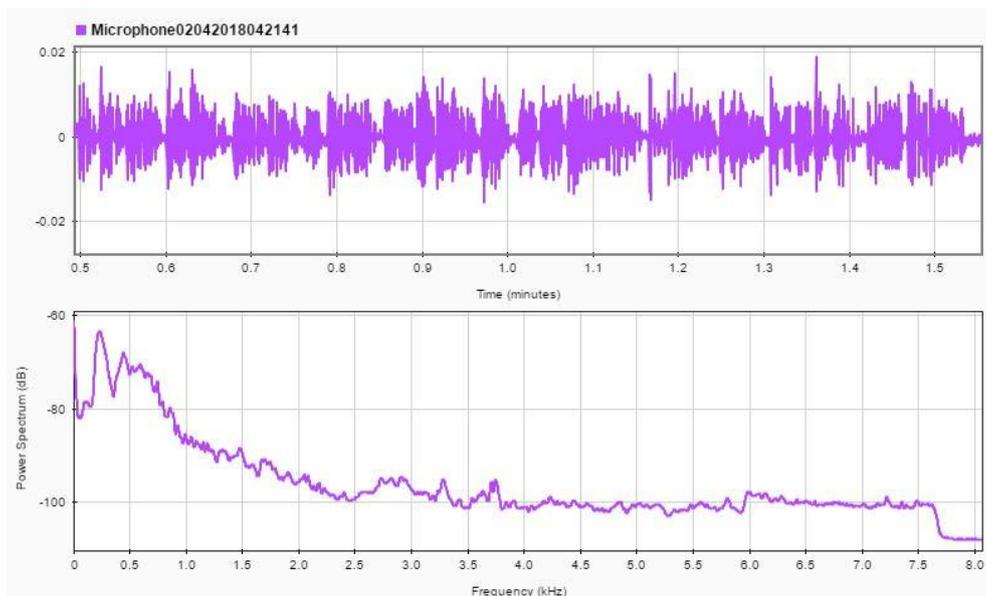


Figura 40. Audio 'Microphone_02_04_2018_04_21_41' del set 2 de la base de datos

5.1. CÁMARA ANECOICA

El concepto de cámara anecoica corresponde a un lugar libre de ruido, ecos y reverberaciones, ya sean internos, como externos. Las clásicas cámaras anecoicas siempre han sido amplias, provistas de materiales absorbentes acústicos sobre sus paredes, techo e incluso suelo. Esto requiere la necesidad de recubrir toda la cámara internamente con absorbente, para así amortiguar resonancias y eliminar esas reverberaciones y ecos no deseados [27]. Figura 41.



Figura 41. Cámara anecoica del Departamento de Teoría de la Señal y Comunicaciones.²³

Las grabaciones de la base de datos tuvieron lugar en la cámara anecoica del Departamento de Teoría de la Señal y Comunicaciones, ya que es un lugar ideal para realizar medidas de audio en las que se requiera la presencia de ruido, eco, o cualquier otra impureza que perturbe la calidad del audio lo más mínima posible. De hecho, para comparar la calidad del sonido en diferentes escenarios, se realizó una grabación del ruido de fondo en la cámara para compararlo con el propio del laboratorio.

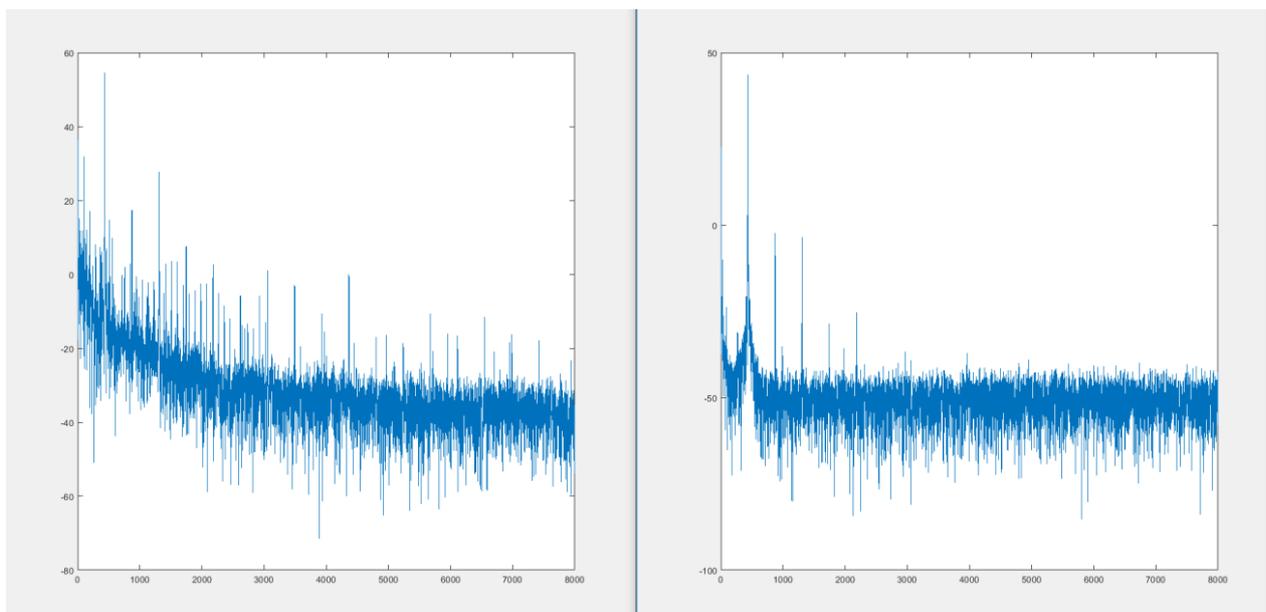


Figura 42. Espectro de frecuencias del ruido de fondo del laboratorio vs. Espectro de frecuencias del ruido de fondo de la cámara.

Como se puede observar en la Figura 42, la señal correspondiente al ruido de fondo en el laboratorio tiene mayor número de componentes armónicos especialmente a bajas frecuencias, que la señal correspondiente al ruido de fondo en la cámara anecoica. Esto se debe a que en el laboratorio el ruido puede provenir de diferentes fuentes durante las grabaciones: puertas, equipos, personas trabajando, rozamientos, fluorescentes, aire acondicionado, etc.

23

http://hosting01.uc3m.es/InvestigaUc3m/descargas/escuela/teoria_de_la_senal/Departamento%20de%20Teoria%20de%20la%20Senal%20y%20Comunicaciones.pdf

5.2. PROTOCOLO INALÁMBRICO VS. ALÁMBRICO

En este apartado, se describe la posibilidad de realizar la adquisición de audio de forma alámbrica, en vez de utilizar *Bluetooth*[®] como protocolo inalámbrico para el envío de audio para su posterior procesamiento en el PC.

1. Primero, se realizó el sistema inalámbrico para *Bindi*, que además también sirvió posteriormente para las grabaciones de la base de datos en la cámara anecoica.
2. Se incluyó el PC para depurar los algoritmos de compresión, de captura de audio, de detección de pánico, etc.
3. Más tarde, se observó que, para entrenar el sistema, el *Bluetooth*[®] impedía arrancar y parar las grabaciones de forma remota, puesto que la cámara anecoica absorbe totalmente las posibles reflexiones causadas por ondas acústicas y electromagnéticas, eliminando todas las frecuencias, de manera que no hay posibilidad de que el *Bluetooth*[®] pueda llegar a dos dispositivos separados por las paredes y puerta de dicha cámara. Tras esta observación, se tomó la decisión de optar por una solución cableada para el sistema destinado al entrenamiento.
4. Se implementó entonces, un sistema que sustituyera el protocolo inalámbrico por un cable USB en ambos de sus terminales, de alcance 10 metros. Figura 43.
5. Por otra parte, si bien la compresión y descompresión del audio se realizaba con motivo del envío de este por protocolo inalámbrico, el sistema cableado incluye la posibilidad de comprimir o no, puesto que sea de utilidad para analizar el efecto de la compresión en el microprocesador y optimizar el códec empleado. Como puede verse en el flujograma de la Figura 44, se ha mantenido la compresión, puesto que para entrenar se debería tener la base de datos comprimida (a través del microprocesador) tal como la va a tratar los algoritmos de detección de estrés.

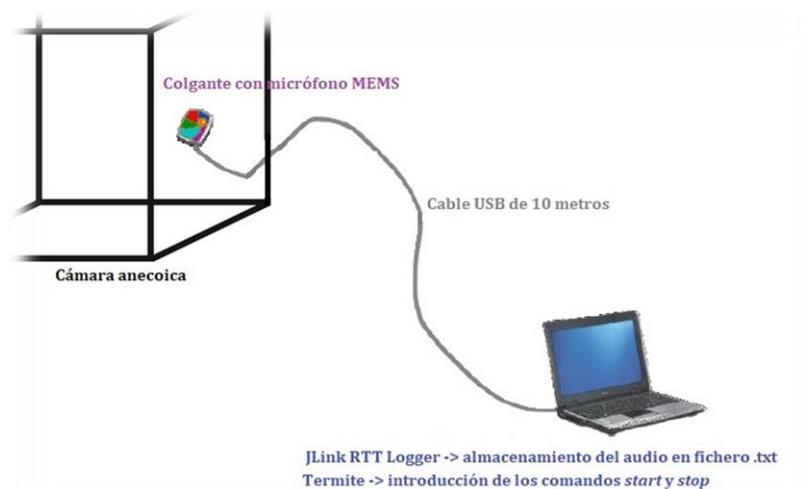


Figura 43. Sistema cableado para realizar grabaciones en la cámara anecoica.

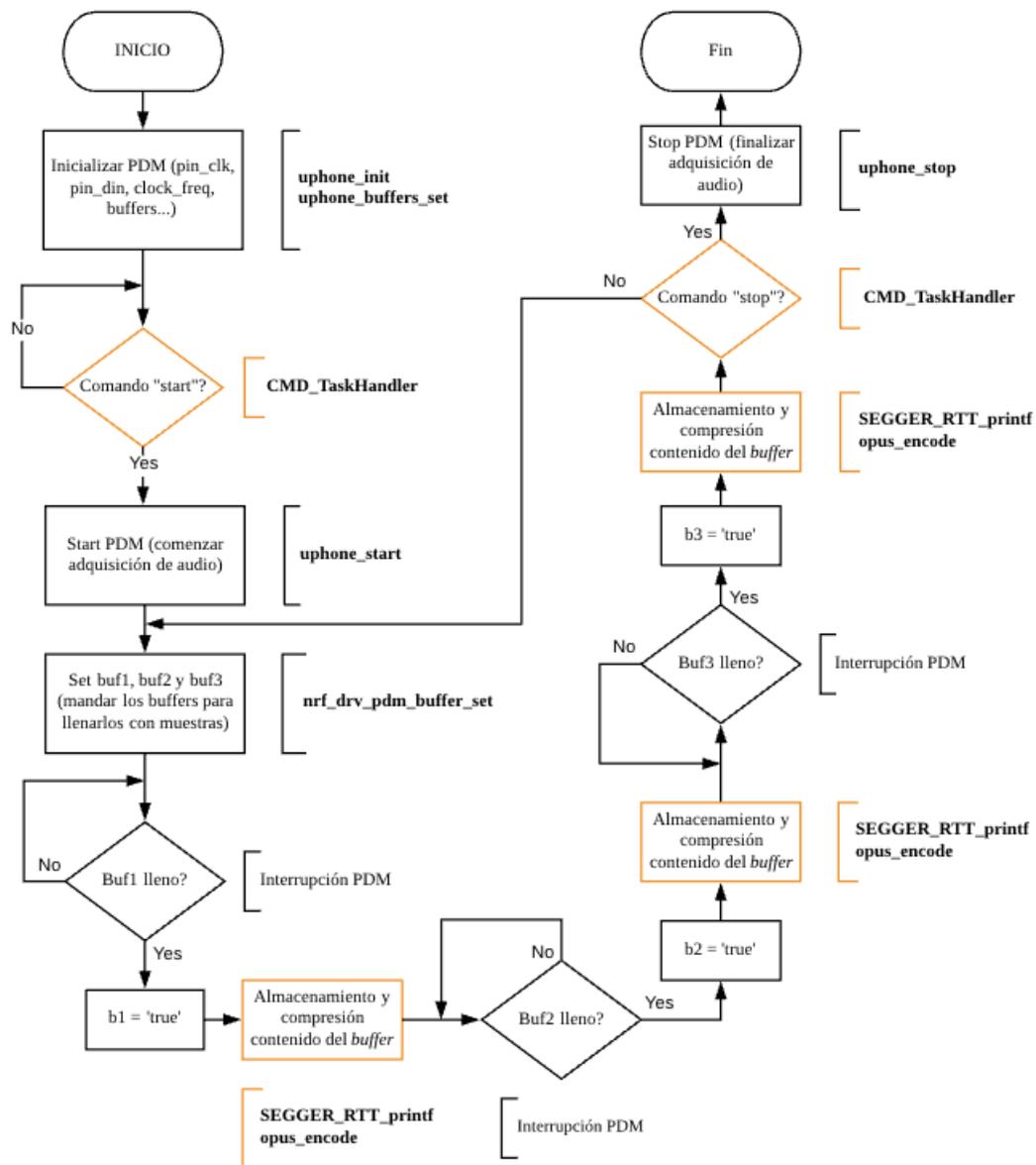


Figura 44. Flujograma de la adquisición del audio mediante protocolo alámbrico.

Adquisición del audio de forma alámbrica (Cable USB de 10 metros)

1. Iniciar grabación.
 1. Introducir el comando *start* a través del terminal serie Termite, utilizado anteriormente para pruebas iniciales. Se sigue utilizando *CMD_TaskHandler* para recibir los comandos, que como ya se ha dicho, recibe la cadena de caracteres que se envíe tanto por la UART como por *Bluetooth*[®]. En este caso, se reciben por la UART.
2. Reproducir el audio a grabar.
 1. Se utiliza la función *SEGGGER_RTT_printf*, para imprimir datos formateados por la interfaz de depuración *SEGGGER JLink RTT (Real-Time Transfer)*. Esta función está proporcionada por la biblioteca *RTT Logging* del SDK.

De esta forma, se almacena el audio grabado en un fichero *txt*, de igual forma que tenía lugar en el móvil a través de la *App* o inicialmente, mediante *Serial Oscilloscope* durante las pruebas de verificación de la adquisición de audio (Figura 29), el cual permitía mediante la opción *Log to file* guardar los datos imprimidos por el terminal.

- II. Mediante *Jlink RTT Logger*, seleccionar una serie de parámetros para configurar la comunicación y el destino del fichero de texto con el audio grabado. Esta herramienta integra RTT: transmisión en tiempo real que permite la comunicación bidireccional a través del canal de depuración con una alta velocidad de transferencia. Además, permite leer los datos enviados y registrarlos en un archivo, todo ello a través de la función mencionada en el paso anterior, *SEGGER_RTT_printf*. Sus parámetros pueden verse en la Tabla 8.

Tipo de parámetro	Nombre	Descripción
entrada	<i>BufferIndex</i>	Índice del buffer de subida/envío. Tiene dos <i>buffers</i> , para usar el primero se le pasa un 0 y para el segundo un 1.
entrada	<i>*sFormat</i>	Puntero a la cadena de datos formateada a imprimir en el archivo <i>txt</i> .

Tabla 8. Parámetros de la función *SEGGER_RTT_printf*.

El terminal de *Jlink RTT Logger*, tiene en su inicio, la configuración de unos parámetros iniciales:

- Nombre del dispositivo con el cual se va a comunicar. Por defecto, será el nombre del dispositivo conectado mediante USB al *Jlink*, que en este caso es el del microcontrolador usado, *nRF52832_XXAA*.
- *Target interface* o interfaz de destino. Puede ser 0 para JTAG o 1 para SWD. Puesto que la interfaz de depuración que posee el microcontrolador *nRF52832* es *Single Wire Debug* (SWD), se configura como 1.
- Velocidad de la interfaz en kHz. Por defecto es 4000 kHz.
- Nombre o índice del canal RTT que se va a utilizar. Por defecto, está configurado en *channel 1*, pero se modifica al 0, que es el parámetro que se le pasa a la función *SEGGER_RTT_printf* para seleccionar el *buffer* de subida que se va a utilizar.
- Nombre del archivo donde se almacenan los datos enviados. Se debe escribir la ruta completa, junto con el nombre, en la que se guardará el archivo creado.

6. CONCLUSIONES

A continuación, en este apartado del proyecto, se exponen las conclusiones sobre el sistema desarrollado. Además, se comentan las posibles mejoras en un trabajo futuro.

En este trabajo se ha realizado:

1. Los *drivers* para gestionar el micrófono MEMS desde el microprocesador ARM *Cortex-M4F* de 32 bits, consumo de energía ultra bajo y multiprotocolo, especialmente indicado para aplicaciones con *Bluetooth® Low Energy* (BLE).

2. Sistema de grabación de audio para *wearable*, que puede enviar audio en *streaming* mediante BLE a un teléfono móvil, para la competición “Anu & Naveen Jain Women’s Safety XPRIZE” en la que UC3M4Safety participó con la propuesta *Bindi*.

⇒ *Mejora*: Añadir al sistema de reconocimiento de estrés en la voz, la detección de este en gritos o palabras clave que indiquen que se está produciendo una agresión verbal y que sirvan para identificar la voz de la usuaria. Todas estas integraciones en el reconocimiento de estrés en la voz, están pensadas de cara a su alegación como prueba judicial o testimonio de la agresión. Para ello, se requieren más grabaciones y pruebas en la cámara, las cuales se facilitarían con una mayor optimización del sistema de entrenamiento, que se describe en las mejoras de los siguientes puntos.

3. Sistema de depuración de grabación de audio mediante la conexión alámbrica e inalámbrica a un PC, con interfaz de comandos en terminal serie o en teléfono móvil respectivamente, y descarga automática de ficheros. Descompresión y conversión a formato WAV.

⇒ *Mejora*: Analizar el efecto de la compresión en el microprocesador y optimizar el códec empleado, mediante la comparación de la calidad del audio recogido mediante la incorporación del códec y sin él, en el sistema alámbrico.

4. Sistema de grabación remota de audios en cámara anecoica mediante cable, para entrenamiento de sistema de reconocimiento de estrés en la voz, mediante interfaz de comandos en terminal serie, con posibilidad de compresión/descompresión o datos en crudo y conversión a formato WAV.

⇒ *Mejora*: Optimizar el sistema de entrenamiento ejecutando todos los pasos desde MATLAB mediante un *script* en el que se manden los parámetros ya fijados de la comunicación junto con los comandos de inicio y detención de la grabación. De esta forma, evitamos los cambios de programa en los diferentes pasos del proceso junto con el archivo de audio, el cual ya tendríamos disponible para su procesamiento y reproducción directa en MATLAB.

Este trabajo se ha llevado a cabo a lo largo de 10 meses, se han escrito alrededor de 600 líneas de código en C y se han definido dos protocolos para la grabación de audio en aplicaciones tecnológicas que buscan la lucha contra la violencia de género, pero en general para unir las capas más altas de la generación de algoritmos de procesamiento de audio con las capas más bajas de adquisición, preprocesado y transmisión del mismo utilizando los componentes electrónicos y los protocolos de comunicación más recientes (*MEMS, Bluetooth® Low Energy..*).

7. PRESUPUESTO

En este capítulo se presenta el presupuesto estimado del proyecto.

7.1. SISTEMA PARA EL PROCESAMIENTO DEL AUDIO

En este apartado se muestra el presupuesto referido a los materiales y equipos requeridos en el desarrollo del sistema y empleados durante la realización de las grabaciones necesarias para el entrenamiento de los algoritmos que detectan el estrés en la voz, las cuales se realizan en la cámara anecoica.

MATERIALES	
Item	Coste
NRF52-DK, Dev Kit para nRF52832 SoC	33'15 €
STEVAL-MKI155V2 (Array de 4 micrófonos MEMS)	38'68 €
Cable USB alcance 10 metros	19'50 €
SUBTOTAL	91'33€

EQUIPOS				
Item	Tiempo de uso	Periodo de amortización	Coste total	Coste
Portátil	8 meses	24 meses	1500 €	500'00€
Osciloscopio	3 meses	24 meses	1500 €	187'50 €
Fuente de alimentación	3 meses	24 meses	1000 €	125'00 €
Polímetro	4 meses	24 meses	200 €	33'33 €
Cámara anecoica	15 días	-	40 €/día	600 €
Licencias MATLAB	-	-	-	2.000'00 €
SUBTOTAL				3.445'83 €

TOTAL	
Subtotal	Coste total
Materiales	91'33 €
Equipos	3.445'83 €
	3.537'16 €

7.2. BINDI

En este apartado se expone el presupuesto referido al colgante, uno de los dos dispositivos *wearable* de la propuesta de UC3M4Safety, *Bindi*, presentada al concurso "Anu & Naveen Jain Women's Safety XPRIZE".

COLGANTE			
Item	Cantidad	Coste unitario	Coste total
Cargador de batería	1	5'080 €	5'080 €
Batería 500mAh	1	6'460 €	6'460 €
Monitor de batería	1	4'680 €	4'680 €
Módulo microprocesador con antena	1	6'110 €	6'110 €
Micrófono	1	1'530 €	1'530 €
Memoria SRAM 1024KB	1	1'830 €	1'830 €
Placa vibrador	1	2'380 €	2'380 €
Oscilador 32'768kHz	1	1'510 €	1'51 €
Reguladores de voltaje LDO 3'3V	1	0'090 €	0'090 €
Diodo Zener 0'5W	2	0'287€	0'574 €
Resistencia shunt 20mΩ	1	0'566 €	0'566 €
Resistencia 10Ω	1	0'435 €	0'435€
Resistencia SMD 1W	4	0'150 €	0'600 €

100Ω 1%			
Resistencia SMD 1W 1kΩ 1%	2	0'082 €	0'164 €
Resistencia SMD 1/10W 100kΩ 1%	2	0'082 €	0'164 €
Condensador SMD 100000pF	7	0'040 €	0'280 €
Condensador SMD 1μF	2	0'080 €	0'160 €
Condensador SMD 10nF	1	0'080 €	0'080 €
		TOTAL	32'693 €

8. BIBLIOGRAFÍA

- [1] W. H. Organization, «Global and regional estimates of violence against women: prevalence and health effects of intimate partner violence and non-partner sexual violence: executive summary,» 2013.
- [2] I. d. I. Mujer, «Porcentaje de mujeres maltratadas según comunidad autónoma,» 2006.
- [3] E. Saíz, «Una de cada cinco universitarias de EE UU sufre agresiones sexuales,» *El País*, 22 Enero 2014.
- [4] W. H. Organization, «Violencia contra la mujer,» 2010.
- [5] M. A. C. Juárez, «Utilización eficiente del espectro electromagnético para comunicaciones inalámbricas inteligentes,» 2015.
- [6] R. Components, «11 redes inalámbricas fundamentales para Internet de las Cosas,» 2016.
- [7] B. Software, «BitZipper,» 2018. [En línea]. Available: <http://www.bitzipper.com/es/aes-encryption.html>.
- [8] I. E. Commission, «Internet of Things: Wireless Sensor Networks,» 2014.
- [9] N. Instruments, «¿Qué es una red de sensores inalámbricos?,» 22 Abril 2009.
- [10] S. R. M. Cantillo, *Desarrollo de aplicaciones basadas en WSN*, Universidad Politécnica de Valencia, 2010.
- [11] A. Bertrand, «Applications and trends in wireless acoustic sensor networks,» de *IEEE Symposium on Communications and Vehicular Technology*, Bélgica, 2011.
- [12] G. Ottoy, B. Thoen y L. De Strycker, «A Low-Power MEMS Microphone Array for Wireless Acoustic Sensors,» de *2016 IEEE Sensors Applications Symposium (SAS)*, Italia, 2016.
- [13] I. F. Akyildiz, T. Melodia y K. R. Chowdhury, «Wireless Multimedia Sensor,» *Proceedings of the IEEE*, vol. 96, nº 10, pp. 1588 - 1605, 2008.
- [14] D.-K. Electronics y E. E. , «MEMS Technology for Microphones in Audio Applications,» 2012.
- [15] J. S. Rueda Rueda y J. M. Talavera Portocarrero, «De las redes de sensores inalámbricas al Internet de las cosas: ¿Tecnologías complementarias o antagonistas?,» de *Congreso Internacional en Innovación y Apropiación de las Tecnologías de la Información y la Comunicación - CIINATIC 2016, At Bucaramanga, Colombia*, Colombia, 2016.
- [16] «Ecured,» 13 Junio 2018. [En línea]. Available: <https://www.ecured.cu/Nanoelectr%C3%B3nica>.

- [17] J. L. Ordóñez, «Dispositivos y tecnologías wearables,» 2016.
- [18] I. D. Corporation, «Worldwide Wearables Market to Nearly Double by 2021, According to IDC,» 2017.
- [19] L. G. Montoro, «Wearables: Qué son, cómo funcionan y que peligros entrañan para nuestra privacidad,» Centro de Estudios de Consumo, Universidad de Castilla La Mancha, 2015.
- [20] *Ley Orgánica 15/1999, de 13 de Diciembre, de Protección de Datos de Carácter Personal*, 1999.
- [21] J. Lewis y D. B. Moss, «MEMS Microphones, the Future for Hearing aids,» 2013.
- [22] J. Lewis, «Analog and Digital MEMS Microphone Design Considerations,» Analog Devices, 2013.
- [23] J. W. A. Morcelli y J. Widder, «Basic principles of MEMS microphones,» *EDN Network*, 2014.
- [24] STMicroelectronics, *MP34DT02 – Data Sheet MEMS audio sensor omnidirectional digital microphone*, 2014.
- [25] N. Semiconductor, *nRF52832 Product Specification v1.4.*, 2018.
- [26] Doxygen, *Opus 1.2*, 2011.
- [27] J. S. garcía, *Diseño y construcción de una cámara anecoica para medidas de audio*, Universidad Carlos III de Madrid, 2013.