

# **Desarrollo de un oído electrónico para la detección precoz de plagas en palmáceas mediante el tratamiento digital de audio.**

---

Máster en Ingeniería de Sistemas Electrónicos

Autor: Roberto Montes Robles

Director: Rafael Masot Peris

31/07/2013



UNIVERSITAT  
POLITÈCNICA  
DE VALÈNCIA



## **Agradecimientos**

En todo trabajo siempre hay un responsable pero mucha gente detrás que contribuye al desarrollo y mejora del mismo. Por esta razón sería muy egoísta por mi parte no nombrarlos tan siquiera.

En primer lugar me gustaría agradecer al Dr. Rafael Masot, por depositar toda su confianza en mí así como por el esfuerzo para suministrarme todos los materiales para su desarrollo.

Agradecer el aporte obtenido por aquellos debates técnicos y cómo no filosóficos transcurridos en las horas de laboratorio. Gracias a Javier Carrascosa, Edwin Loeff, Cristian Olguín y Hector Pardo.

Al grupo GED+PS en general por el buen ambiente de trabajo y por acogerme como uno más desde el primer momento.

A Dr. Rafael Gadea, por sus buenos consejos en la fase inicial de la tesis.

Por último agradecer a mis familiares el gran apoyo moral y logístico depositado en mí desde el primer momento, en especial a mis padres que siempre han estado ahí.

## Contenido

1. Objetivo.....	3
2. Estado del arte.....	3
2.1 Introducción .....	3
2.2 Métodos y materiales empleados .....	5
2.3 Biología del RPW .....	6
2.4 Identificación de los sonidos producidos por el RPW .....	8
3. Procesado de la señal de audio por computador .....	9
3.1 Introducción .....	9
3.2 Análisis de los ficheros de audio y obtención de patrones.....	10
3.2.1 Análisis en el dominio del tiempo.....	11
3.2.2 Análisis en el dominio de la frecuencia .....	23
3.2.3 Análisis tiempo frecuencia (STFT).....	29
4 Implementación del sistema.....	38
4.1 Introducción .....	38
4.2 Implementación Hardware .....	40
4.2.1 Recursos hardware empleados de la placa de desarrollo .....	40
4.2.2 Arquitectura hardware diseñada.....	44
4.3 Implementación Software .....	47
5. Resultados .....	53
6. Conclusiones .....	57
7. Líneas futuras.....	58
8. Bibliografía .....	59

## 1. Objetivo

El objetivo de la presente tesis es el de realizar el diseño e implementación de un oído electrónico, que mediante la captación del sonido producido por la actividad del Picudo Rojo (RPW) *Rhynchophorus ferrugineus* (Olivier) en el interior de la palmera, permita la identificación y detección precoz de la plaga mediante tratamiento digital de audio.

## 2. Estado del arte.

### 2.1 Introducción

El RPW ha sido el causante de grandes pérdidas económicas relacionadas con la infección y destrucción de las palmeras datileras, (*Phoenix dactylifera L.*) en las zonas afectadas por esta plaga. Los costes económicos derivados de la plaga pueden ser separados en directos, originados por la pérdida de producción en la crianza de las palmeras y los causados por la contención de la plaga. Los daños económicos cifrados en EE.UU datan en 138 billones de dólares anuales [Manachini et al. 2012]. Estos daños producen un serio impacto ambiental, afectando de forma indirecta en una reducción del turismo. Además, si estas pérdidas se asociaran a la degradación de la biodiversidad relacionada con los ecosistemas el coste económico aumentaría drásticamente. Por todos estos motivos se puede afirmar que la influencia de esta plaga afecta al patrimonio natural y cultural.

Los primeros casos de infección de la larva del RPW [Al-Manie et al.2007] fueron detectados en el sudeste de Asia (1750-1755 Sri-Lanka). En la pasada década la plaga evolucionó hacia las zonas del este de China y Japón extendiéndose hacia el oeste mediterráneo (España, Egipto, Grecia, Chipre, Israel, Italia, Francia y Turquía) causando importantes daños en las palmeras datileras (*Phoenix dactylifera L.*) y cultivos de palmeras ornamentales (*mainly Phoenix dactylifera L.*). Además esta plaga supone una gran amenaza para la especie de palmáceas *Phoenix theoprasti Greuter* autóctona en bosques como los localizados en la isla de Creta.

Las estrategias de control han de tener en cuenta que la presencia del RPW es una plaga en sí misma. Los diferentes métodos de control y exterminación quedan reflejados en los puntos que se muestran a continuación:

1. Mediante el uso de depredadores naturales. Artículos recientes ponen de manifiesto la existencia de un hongo *Beauveria bassiana* [El-Sufty et al. 2012] catalogado como enemigo natural. Pese a lo esperanzador del descubrimiento las publicaciones muestran una baja efectividad. Puede ser que en un futuro próximo sea una alternativa a considerar, no obstante actualmente se encuentra en fase de experimentación no siendo una opción extrapolable a los métodos de campo.

2. Uso de trampas. Hasta la fecha se trata del elemento más usado para el control y captura de adultos [E.P.P.O. 2005]. Está demostrado que los adultos machos bien alimentados generan un tipo de feromona que es capaz de atraer tanto a otros machos como a las hembras. Dicha feromona solo atrae a estos coleópteros desde una cierta distancia. Por otro lado, las bacterias fruto de la descomposición del tejido de la palmera generan un potente olor por el cual también se sienten atraídos, extendiéndose más todavía el área de atracción. Las trampas utilizadas para la captura del RPW contienen materiales vegetales en descomposición, tales como el tejido de palmeras, piña y caña de azúcar. También se ha notado la influencia del color de la trampa, siendo las de color amarillo más eficaces que las de otros colores. Las versiones más modernas de trampas utilizan feromonas de origen natural o sintético. La captura de adultos reduce considerablemente las probabilidades de infección. Este mecanismo actúa de forma preventiva, no teniendo efecto sobre las palmeras ya infectadas, por lo que se considera un mecanismo de control complementario a otros métodos.
3. Quema de palmeras infectadas. Se trata de una técnica destructiva muy efectiva contra la plaga ya que se destruyen todos los rastros del RPW. Esta técnica está indicada para los casos en los que no existe la posibilidad de salvar la palmera al encontrarse en un estado de infección muy avanzado o terminal.
4. Tratamiento no invasivo mediante técnicas microondas. El uso de microondas está basado en el hecho de que la radiación aplicada en la corona de la palmera, o zona afectada, puede inducir un incremento térmico suficiente para la exterminación de la plaga sin afectar a los tejidos de la planta [Massa et al. 2011]. Este método no invasivo resulta ser muy efectivo en un estado inicial e incluso en estados más avanzados de la infección. El problema que presenta es el elevado precio que supone el tratamiento. Además al igual que en otros tratamientos cuando la palmera se encuentra en un estado muy avanzado de infección, no resulta efectivo.
5. Tratamiento fitosanitario de las palmeras. Actualmente es el método más utilizado para el control de la plaga. Por un lado previene la proliferación de las larvas neonatas eclosionadas en el cogollo de la palmera o en su periferia. Por otro lado ayuda a reducir la población de larvas existentes en el estipe una vez que la palmera ya ha sido infectada.

Para tener garantías de éxito en los diversos tratamientos se hace necesaria la detección temprana de esta plaga. Estudios realizados por el investigador Mankin [Mankin. 2010] también avalan dicha hipótesis. Sin embargo los síntomas externos no son lo suficientemente claros para su identificación precoz, mas bien todo lo contrario. Con los primeros síntomas la palmera suele ya estar en un estado de infección muy avanzado. Entre

los síntomas visuales que se aprecian en las palmeras infectadas [E.P.P.O. 2005] se puede destacar: una fuerte coloración amarillenta del área foliar, la destrucción de las hojas emergentes y una necrosis en las flores. Las hojas comienzan a secarse de forma ascendente desde la corona curvándose hasta que finalmente acaban cayéndose. Internamente las galerías y los daños producidos por las larvas se detectan fácilmente en palmeras con clara infección. Pupas y larvas adultas se encuentran fácilmente en la corona de las palmeras fuertemente infectadas. El tejido de la palmera afectada genera un fuerte olor característico.

Los métodos de identificación basados en tecnología acústica han sido objeto de multitud de artículos científicos. Existen varias publicaciones que constatan sistemas para la detección de termitas [Sheffrahn et al. 1993; Mankin et al. 1997]. También se han realizados estudios para la caracterización acústica de la actividad de gusanos blancos en suelos [Mankin et al. 2000; Zhang et al. 2003] y de insectos en tallos de plantas [Mankin et al. 2004].

Centrando los estudios de investigación para la detección del RPW el investigador Al-Manie [Al-Manie et al. 2007] propone un método teórico de difícil extrapolación a pruebas de campo. Otro autor, Gutiérrez, [Gutiérrez et al. 2010] documenta el diseño de un sensor bioacústico portátil capaz de detectar en un entorno controlado de laboratorio infestación mediante la comparación de la amplitud de la señal auditiva con un determinado umbral (*nivel de señal*) a tan solo una frecuencia. El problema que se presenta es el de distinguir los sonidos producidos por la plaga diferenciándolos del resto de sonidos captados del entorno. Estos sonidos pueden ser producidos por otros insectos, animales pequeños, viento etc... La identificación de la actividad acústica de la larva se basa principalmente en el estudio de patrones específicos generados, como puede ser el masticado, su desplazamiento, mecanismos de comunicación, movimientos peculiares, etc...

## 2.2 Métodos y materiales empleados

A continuación se describe el material empleado para la captura de los audios en las diferentes pruebas de campo. Es conveniente remarcar que tanto el diseño del hardware que se describe en este apartado, como los audios capturados han sido proporcionados por terceros para la realización del trabajo que se muestra, no considerándose objeto de la presente tesis.

El sistema de captura de audios se compone por un micrófono piezoeléctrico modelo Oyster marca Schaller, un amplificador diseñado para esta aplicación y un ordenador portátil (véase Figura 1). Para su correcto funcionamiento el micrófono piezoeléctrico ha sido polarizado con una tensión de 9V. La señal procedente del sensor ha sido acondicionada y amplificada para ser capturada por un ordenador portátil (Figura 2). El circuito ha consistido en un filtro paso alto de entrada para eliminar la componente de

continua del sensor, un amplificador no inversor y un filtro paso bajo de tipo Butterworth de segundo orden para evitar el efecto aliasing.

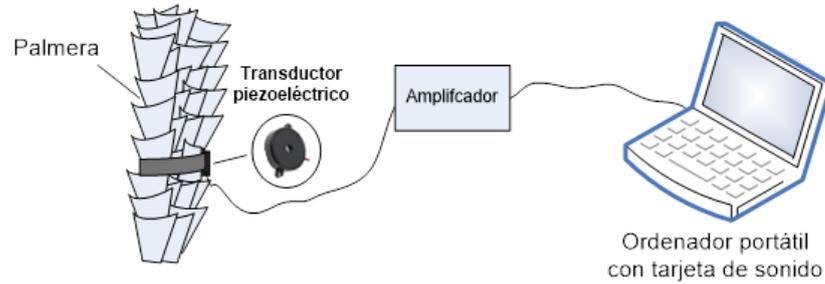


Figura 1. Sistema de captura de audio compuesto por un sensor piezoeléctrico, equipo de acondicionamiento de la señal y ordenador portátil

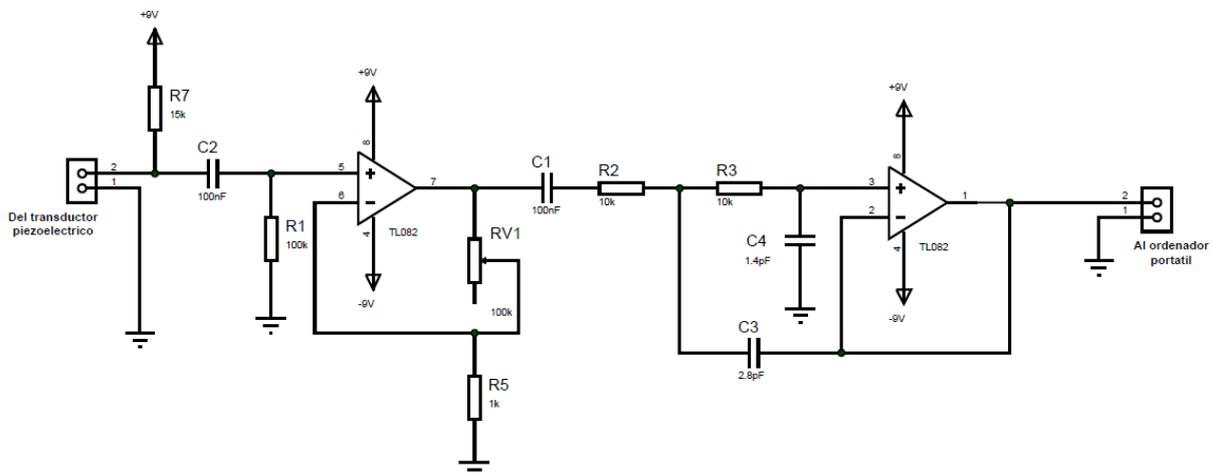


Figura 2. Esquema electrónico del equipo de acondicionamiento de la señal

Los audios capturados en las distintas pruebas de campo han sido almacenados con una frecuencia de muestro de 16384Hz. La captura de los distintos ficheros informáticos tiene el objetivo de procesarlos para encontrar un algoritmo que permita detectar la presencia del coleóptero en la planta.

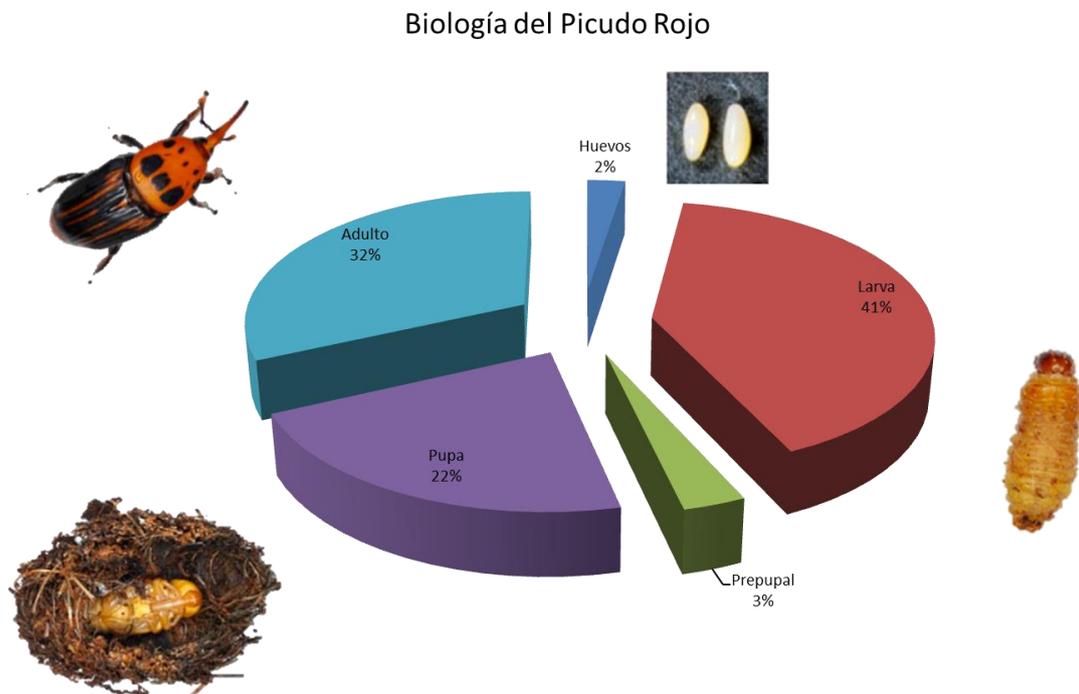
### 2.3 Biología del RPW

El ciclo biológico del RPW [E.M.P.O. 2005] comienza tras la deposición del orden de unos 250 huevos por parte de los adultos hembra depositándolos en cavidades de la superficie de la palmera. El periodo de incubación es aproximadamente de 3 días. Una vez eclosionado el huevo nacen las larvas y éstas se alimentan de tejido vegetal vivo preferentemente fibroso. Las larvas penetran por las zonas blandas de la palmera produciendo cavidades y reduciendo así la transferencia de agua y de nutrientes entre las raíces y el cogollo. La duración media de la larva es de aproximadamente 52 días. Una vez

pasado esta etapa la larva se encuentra en una etapa prepupal que dura entre 4 y 17 días, durante el cual, fabrica su capullo con fibras vegetales. Las larvas se convierten en pupas realizando la metamorfosis durante un periodo de unos 19 días. Los adultos permanecen en el capullo una semana más antes de salir al exterior.

El adulto sale del capullo dando continuidad a una siguiente generación. Tras la etapa de fecundación y puesta de huevos contribuye aumentando así la población de las larvas y consecuentemente el nivel de infestación. Este proceso continúa hasta que haya consumido los recursos vegetales existentes [Cabello 2005]. Un alto grado de deshidratación en la palmera provoca la limitación de los recursos alimenticios, dando lugar a la emigración del adulto en busca de otra palmera. Los adultos pueden vivir del orden de unos 45 días. Su desplazamiento presenta una actividad bimodal combinando estados de reposo con estados de mayor actividad. Los estudios realizados apuntan a una mayor actividad puntualmente entre las 7 y 11 h así como entre las 17 a 19 h aprovechando la luz del sol, tomando descanso durante la noche y las horas de más calor. Una vez alcanzada otra palmera deposita los huevos, comenzando de nuevo el ciclo y extendiéndose así la plaga.

En el gráfico de la Figura 3 se puede apreciar de forma porcentual el tiempo medio de los distintos estados que adquiere el RPW en su ciclo de vida.



**Figura 3.** Gráfico porcentual del tiempo promedio de los distintos estados de vida del RPW [E.M.P.O. 2005]

## 2.4 Identificación de los sonidos producidos por el RPW

Estudios realizados por el investigador Mankin [*Mankin. 2010*] afirman que en los distintos audios capturados en las pruebas de campo existen multitud de sonidos que pueden poseer componentes frecuenciales similares a los producidos por el RPW. Para la distinción de las señales patrón de otras señales acústicas ha de tenerse en cuenta que muchos de los sonidos originados en un ambiente ornamental se producen con periodos de 10 s o más, con frecuencias de hasta 1 kHz. En cambio la actividad del RPW se encuentra en frecuencias entre 1 a 3.8 kHz con periodos que oscilan entre 3 a 15 ms. Por otro lado, las señales de alta frecuencia captadas del RPW en el interior de la palmera se atenúan más fuertemente con la distancia que las de baja frecuencia. De la anterior hipótesis se deriva que señales a distancias de 0.5, 1 y 4 m son notablemente diferentes. Además las larvas pequeñas son susceptibles de generar señales de frecuencia y amplitudes mucho más bajas que las larvas grandes.

Teniendo en cuenta los posibles inconvenientes detallados anteriormente se ha realizado un estudio de los distintos audios recogidos en las pruebas de campo en el que se puede apreciar cuatro sonidos característicos del RPW:

- El mordisco de la larva con sus potentes mandíbulas
- Una especie de gemido
- El movimiento de arrastre de la larva a través de los túneles excavados
- El movimiento repetitivo del abdomen de la pupa dentro del capullo o cápsula ninfal

Como conclusión del análisis de los audios capturados en las pruebas de campo se ha extraído que tanto el mordisco, el gemido, como el movimiento de la larva en el estipe producen sonidos muy sutiles difícilmente identificables con las técnicas de captura de audio utilizadas. En cambio los audios producidos por el movimiento abdominal de la pupa se distinguen fácilmente, esto es debido a que no se ven sometidos a dicha atenuación ya que están dispuestos en la corteza de la palmera. Por esta razón el estudio se ha focalizado en la actividad sonora fruto del movimiento ondulatorio abdominal de la pupa dentro del capullo generado cuando las pupas se sienten amenazadas [*E.P.P.O. 2005*]. Esta amenaza se manifiesta cuando éstas son sometidas a pequeñas vibraciones, olores, frío, calor...

### 3. Procesado de la señal de audio por computador

#### 3.1 Introducción

El objetivo de este apartado es realizar un procesado de las señales de los audios offline tratando de encontrar un algoritmo que muestre como resultado final la presencia o no del sonido relacionado con el movimiento abdominal de la pupa.

El análisis de los audios se ha realizado mediante el software específico de audio Audacity y el software matemático de uso general Matlab de la empresa MathWorks. Por un lado destaca la potencia del software libre Audacity permitiendo un análisis previo de los audios tanto en el dominio del tiempo como en el dominio de la frecuencia. Por otro lado destaca la capacidad de procesado de Matlab, la potencia en cuanto a la representación gráfica así como la innumerable documentación aportada tanto por fuentes oficiales como extraoficiales. Pese a no ser una herramienta de libre distribución, la Universidad Politécnica de Valencia posee licencia de multitud de versiones siendo de fácil acceso para investigadores y docentes.

Los distintos métodos implementados en este apartado se han visto sometidos a un testeo previo con el fin de poder comprobar y evaluar su efectividad. Los archivos de audio de referencia para este análisis comparativo han sido seleccionados atendiendo en primer lugar a los signos de actividad del movimiento abdominal de la pupa diferenciando si existe actividad (*positivos*) de cuando no (*negativos*). En la tabla que se muestra a continuación (*véase Tabla 1*) se ha realizado una clasificación de estos audios, explicando brevemente su procedencia y si contiene signos claros de la presencia del coleóptero o no.

La filosofía de trabajo que se presenta en los siguientes apartados busca como base fundamental la sencillez de implementación en su posterior desarrollo hardware, no por ello sin descuidar uno de los principales objetivos, que es el procesado de la señal en tiempo real. Por esta razón el método descrito parte del análisis más simplificado, empleando señales en el dominio del tiempo. Posteriormente se analizan las señales en el dominio de la frecuencia. Por último se hace un estudio de las señales combinando las dos técnicas anteriores.

	Audio	Descripción
Positivo	Probando0	Movimiento abdominal, intensidad débil
	Probando1	Movimiento abdominal, intensidad media
	Probando2	Movimiento abdominal, intensidad media
	Probando3	Movimiento abdominal, intensidad media
	Probando4	Movimiento abdominal, intensidad media
	Probando5	Movimiento abdominal, intensidad media
	Probando6	Movimiento abdominal, intensidad media
Negativo	Probando7	Movimiento abdominal, intensidad media
	Probando8	Voz
	Probando9	Voces de fondo
	Probando10	Ruido de fondo
	Probando11	Ruido saturado del micrófono
	Probando12	Ruido de fondo
	Probando13	Voz

Tabla 1. Clasificación de varias pistas de audio

### 3.2 Análisis de los ficheros de audio y obtención de patrones.

Resulta inmediato poder localizar mediante la escucha de los audios tomados en las pruebas de campo la procedencia de ciertos patrones de audio relacionados con el movimiento abdominal de la pupa dentro del capullo. Por esta razón el primer objetivo ha sido aislar el patrón de este movimiento y obtener su evolución temporal.

En la parte superior de la Figura 4 se puede apreciar un fragmento de un audio de cinco segundos que contiene el sonido del movimiento abdominal de la pupa. Este fragmento contiene aproximadamente 82.000 muestras con una frecuencia de muestreo de 16384Hz. Realizando un zoom de la misma (*parte inferior de la figura 4*) se puede aislar el patrón de audio de un solo movimiento abdominal. Puede observarse que este audio, considerado como patrón, es de tan solo 4 ms, es decir de unas 64 muestras.

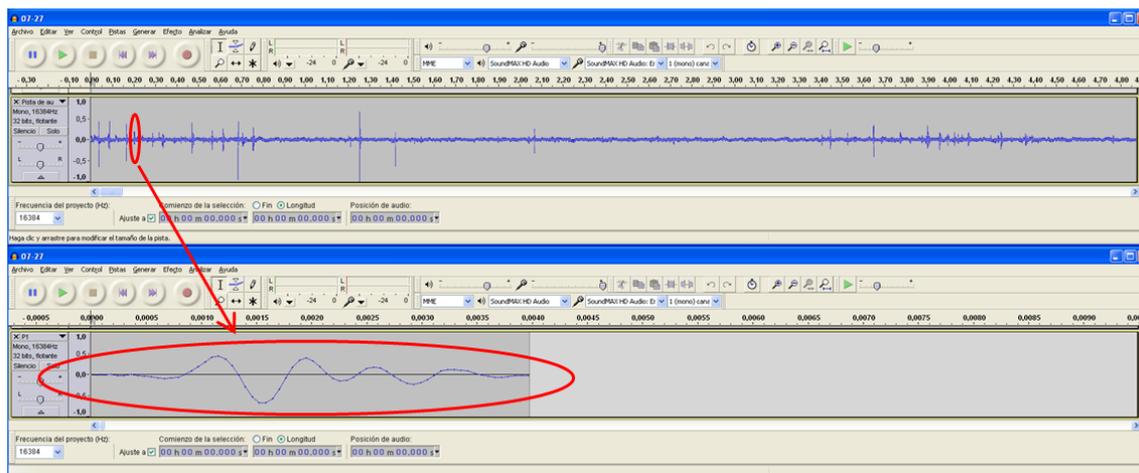


Figura 4. Obtención del patrón del movimiento del abdomen de una larva

Con una de las herramientas del programa Audacity, se han extraído diversos patrones de audio de las grabaciones existentes. Para garantizar la compatibilidad con el programa Matlab, las señales patrón fueron exportadas a ficheros de tipo “wav”. El programa Matlab transforma los ficheros “wav” en matrices representando las amplitudes de forma numérica.

### 3.2.1 Análisis en el dominio del tiempo

#### Análisis por correlación

La técnica de correlación en audio determina la relación lineal o dependencia estadística que existe entre dos señales de audio. El coeficiente de correlación lineal o coeficiente de Pearson se representa con el símbolo “ $\rho$ ”. La operación matemática de este algoritmo queda descrita en la ecuación (1). El objetivo de utilizar un análisis de correlación es el de comparar dos archivos de audio, siendo éstos, el audio a procesar con el audio patrón.

$$\rho(x, y) = \frac{\sigma_{XY}}{\sigma_X \cdot \sigma_Y} \quad (1)$$

Para una clara exposición de las operaciones realizadas se han nombrado las matrices obtenidas en Matlab de la siguiente forma: en primer lugar, la matriz “origen” que hace referencia a un archivo de audio de aproximadamente cinco segundos sobre el cuál se le ha aplicado la matriz del fragmento del sonido del movimiento abdominal llamada “patrón”. En el siguiente paso se ha extraído de la matriz de audio “origen” una sub-matriz a modo de ventana del mismo número de muestras que la matriz “patrón”. Esta nueva sub-matriz se ha correlacionado con la matriz “patrón” determinando así la relación lineal entre las dos matrices. El coeficiente de Pearson “ $\rho$ ” indica el grado de correlación, de forma que si “ $\rho$ ” tiende a uno, las dos matrices tienen una gran dependencia lineal, es decir, tienen

formas muy parecidas. En cualquier otro caso las formas de onda no son similares. Acto seguido la sub-matriz se desplaza una muestra a través de la matriz “origen” (*ventana deslizante*) y de nuevo se vuelven a correlacionar esta matriz con la matriz “patrón” obteniendo otro coeficiente de Pearson. Por último todos los coeficientes “ $\rho$ ” se almacenan en un vector para su posterior análisis.

En la parte superior de la Figura 5 se muestra la representación gráfica en el dominio del tiempo del audio “origen”, marcando con una franja en rojo los periodos de actividad del movimiento abdominal del RPW. La parte inferior de la Figura 5 muestra los valores del coeficiente de Pearson que se han obtenido. Con el objetivo de diferenciar los valores mayores de 0.85 se han rodeado éstos con un círculo rojo. Analizando los resultados representados en la Figura 5, se ha comprobado que cuando el coeficiente de Pearson supera el valor de 0.85 se encuentra presencia del fragmento patrón en el audio “origen” o lo que es lo mismo, el sonido del movimiento abdominal es detectado en el audio puesto a examen. Comparando la parte superior con la inferior de la Figura 5 se aprecia cómo coinciden los picos de audio producidos por el movimiento abdominal de las pupas con valores de “ $\rho$ ” por encima de 0.85. Por tanto se puede afirmar que conociendo el valor del coeficiente de Pearson “ $\rho$ ” puede determinarse si el audio “origen” posee o no dicho sonido.

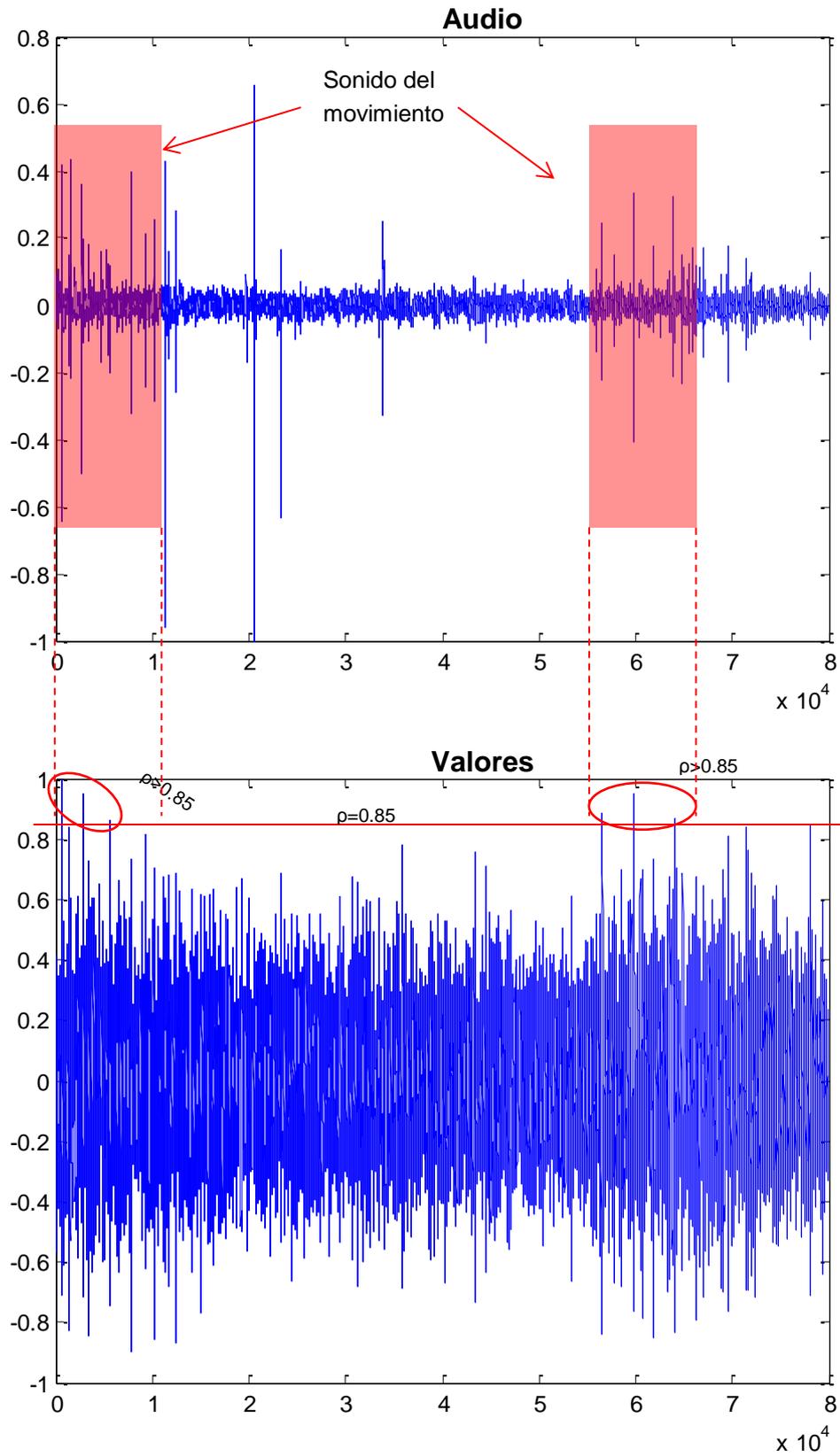


Figura 5. Comparativa entre audio "origen" y valores de " $\rho$ "

Cabe destacar que estos buenos resultados han sido obtenidos de la extracción de un fragmento de audio patrón del propio audio a analizar. Por este motivo se han realizado diferentes pruebas con fragmentos de audio distintos para poder comprobar si el algoritmo mantiene estas buenas prestaciones.

A continuación en la Figura 6, Figura 7, Figura 8 y Figura 9 se representa gráficamente los resultados fruto de la correlación del mismo archivo patrón con otros fragmentos de audio “positivos” capturados en las pruebas de campo.

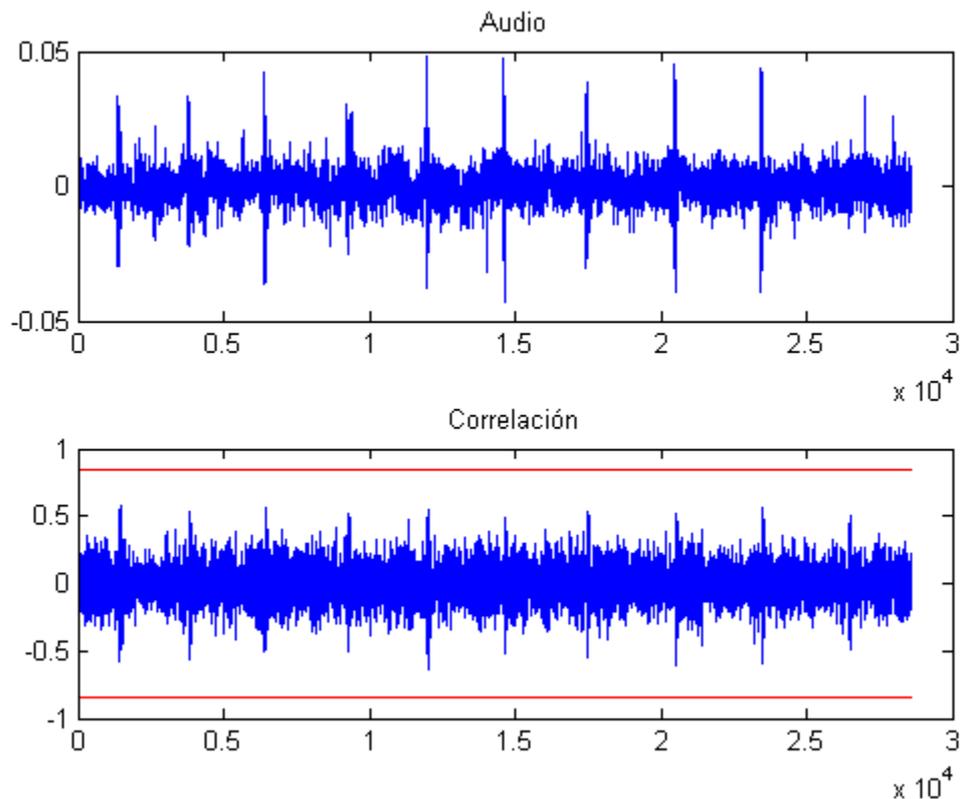


Figura 6. Probando0. Pista de audio (superior). Correlación con el archivo patrón (inferior)

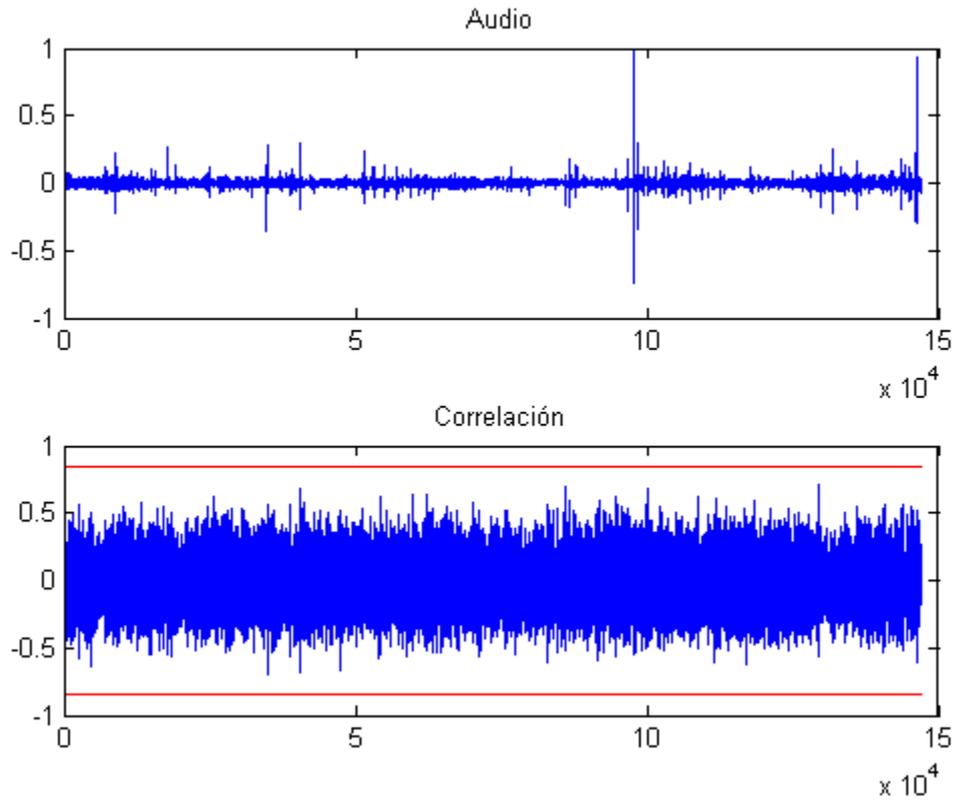


Figura 7. Probando1. Pista de audio (superior). Correlación con el archivo patrón (inferior)

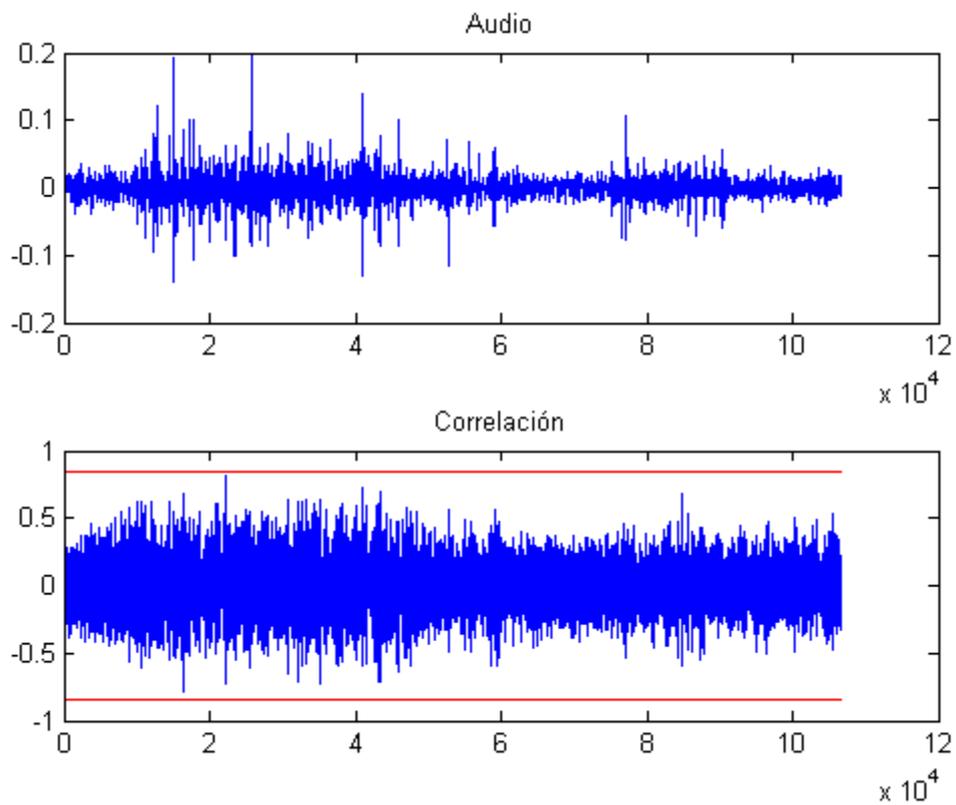


Figura 8. Probando2. Pista de audio (superior). Correlación con el archivo patrón (inferior)

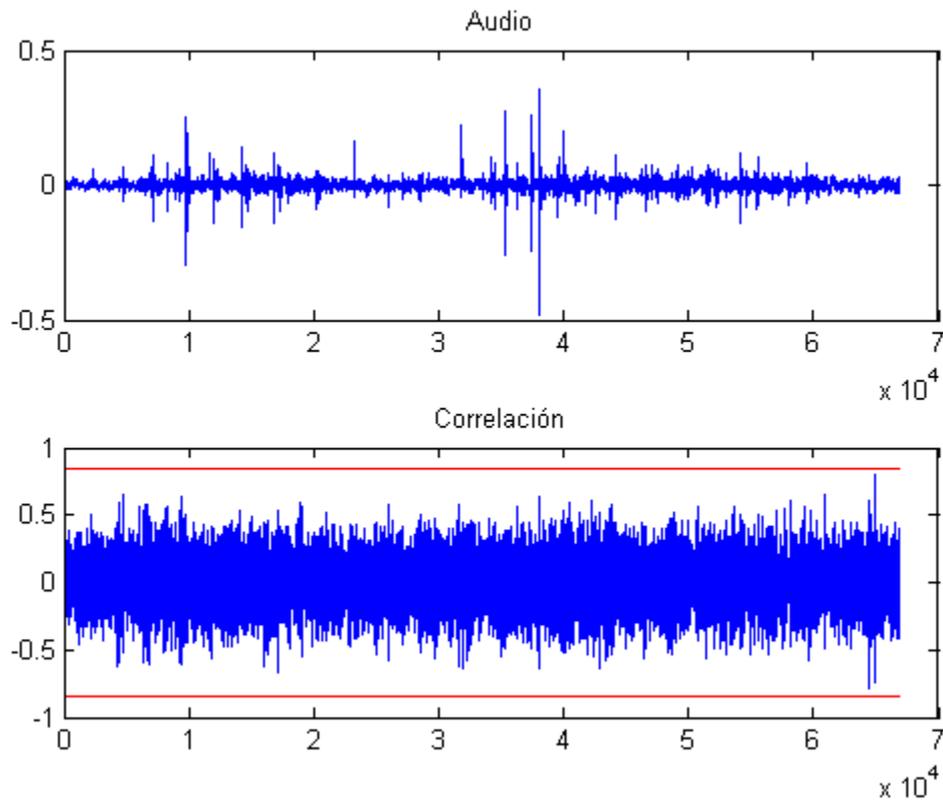


Figura 9. Probando3. Pista de audio (superior). Correlación con el archivo patrón (inferior)

Como se puede apreciar en la Figura 6, Figura 7, Figura 8 y Figura 9 no se alcanza el umbral de decisión para audios catalogados como positivos, exceptuando en el archivo denominado “Probando 6”, fragmento del cual se ha extraído el fichero “patrón”. Por otro lado en la Tabla 2 queda claramente reflejada la ineffectividad de este método con el material utilizado. Queda así comprobado que este algoritmo no permite su generalización en estas condiciones.

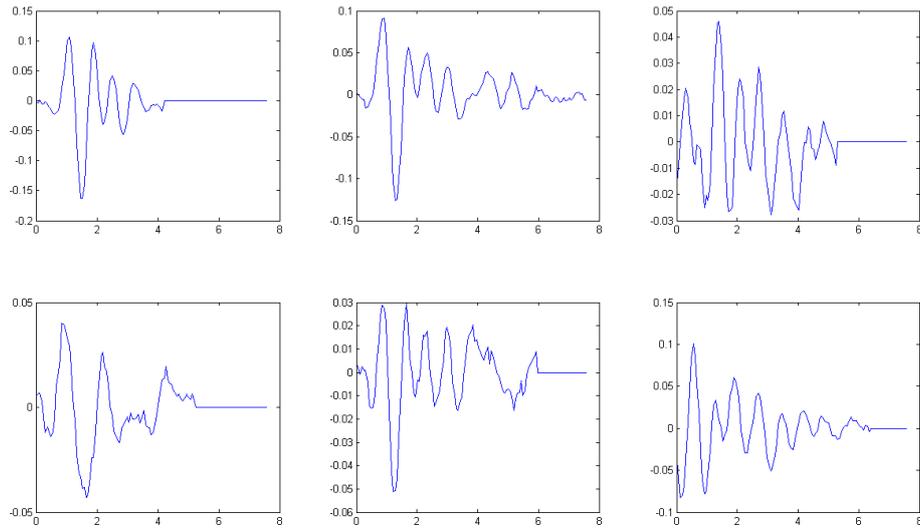
De la conclusión anterior se plantea varias soluciones a fin de corregir las deficiencias del algoritmo. Por un lado existe la necesidad de encontrar un archivo patrón lo suficientemente representativo para que retorne el mayor número de aciertos cuando procesa audios catalogados como “positivos” y viceversa. Para ello se han recopilado varios fragmentos patrón con el objeto de realizar un análisis visual comparativo buscando similitudes ante los mismos. Analizando los distintos patrones de audio al detalle se puede apreciar en la Figura 10 y Figura 11 que la forma del audio es muy dispar, no pudiéndose determinar de forma sencilla unos rasgos característicos que puedan ser comunes a todos. Para comprobar esta disparidad mediante técnicas matemáticas se ha comparado unos patrones con otros realizando análisis estadísticos. Los resultados de estos análisis ofrecen un grado de correlación entre los mismos muy bajo. Este resultado se puede enfocar desde otro punto de vista, afirmando que todas las formas temporales analizadas del tipo patrón

pueden ser igualmente válidas. Por esta razón el hecho de solo disponer de un solo patrón para la detección acústica hace que el algoritmo sea muy selectivo. Para conseguir una detección eficaz, una posible solución podría consistir en disponer de una amplia batería de patrones. No obstante esta solución no garantiza la total cobertura del problema, además supondría un algoritmo muy pesado si se quiere realizar un procesado de señal en tiempo real.

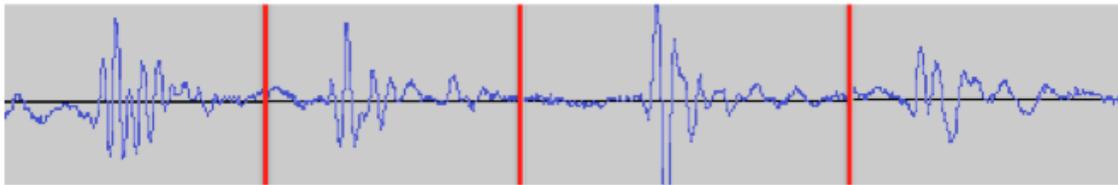
En la Tabla 2 se resume el número de aciertos por cada audio para el algoritmo estudiado

	Audio	Descripción	Aciertos
Positivo	Probando0	Movimiento abdominal, intensidad débil	0
	Probando1	Movimiento abdominal, intensidad media	0
	Probando2	Movimiento abdominal, intensidad media	0
	Probando3	Movimiento abdominal, intensidad media	0
	Probando4	Movimiento abdominal, intensidad media	0
	Probando5	Movimiento abdominal, intensidad media	0
	Probando6	Movimiento abdominal, intensidad media	4
	Probando7	Movimiento abdominal, intensidad media	0
Negativo	Probando8	Voz	0
	Probando9	Voces de fondo	0
	Probando10	Ruido de fondo	0
	Probando11	Ruido saturado del micrófono	0
	Probando12	Ruido de fondo	0
	Probando13	Voz	0

Tabla 2. Tabla resumen mostrando los resultados obtenidos



**Figura 10. Distintos fragmentos identificados**



**Figura 11. Comparativa de fragmentos de forma detallada**

Por otro lado se plantea una “relajación” del algoritmo utilizando un umbral menos exigente para obtener así un mayor número de aciertos. El inconveniente que presenta esta “relajación” es que resulta mucho más probable que se produzca un falso acierto considerándose por tanto una estrategia muy arriesgada. Para corroborar la hipótesis anterior se ha realizado una prueba con un archivo de audio del tipo “negativo”. En la Figura 12 se muestra la representación temporal de los resultados de esta correlación. Como se puede apreciar el coeficiente de Pearson “ $\rho$ ” representado en la parte inferior de la figura se aproxima peligrosamente a 0.8 aun siendo un audio claramente “negativo”. Esta aproximación deja patente que no se puede considerar buena una solución menos restrictiva.

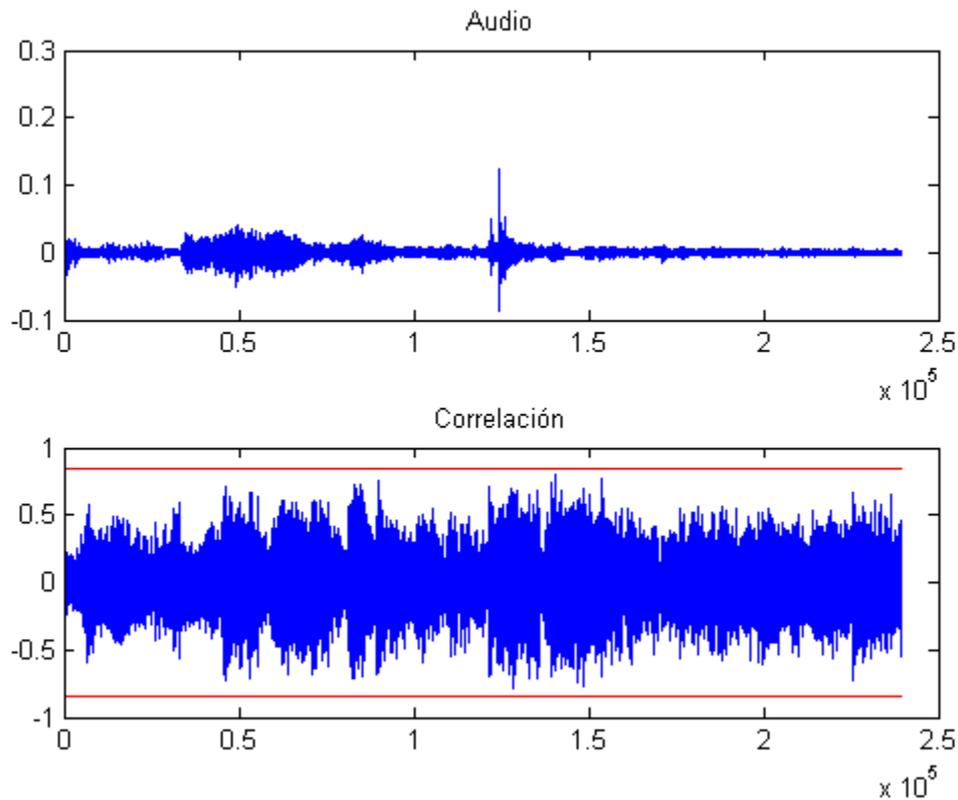


Figura 12. Probando9. Pista de audio (superior). Correlación con el archivo patrón (inferior)

Tras los malos resultados obtenidos en las distintas pruebas realizadas se ha descartado este tipo de análisis.

### Análisis estadístico PCA

Siguiendo con el estudio estadístico de las matrices de audio en el dominio temporal, en este apartado se aborda la técnica de análisis de componentes principales o PCA. El objetivo de esta técnica es el de reducir la dimensionalidad del conjunto de datos.

Se destaca también la gran experiencia del grupo de investigación en el reconocimiento de patrones mediante el análisis estadístico PCA aplicado al campo de la espectroscopia e impedancias [Martinez-Mañez et al. 2004 ; Escriche et al. 2012]. Motivados tras los buenos resultados, se decide aplicarla también al reconocimiento de patrones de audio.

Este tipo de análisis requiere un procesamiento de los datos previo, que consiste en dividir la matriz de datos del audio en fragmentos de 62.5 ms (1024 muestras). Una vez troceado se realiza una identificación manual determinando si existe actividad del movimiento abdominal o no, en cada fragmento. De este análisis se ha podido localizar dicha actividad en las muestras que se presentan a continuación.

$Acierto = [31\ 32\ 33\ 35\ 37\ 39\ 40\ 42\ 45\ 46\ 49\ 50\ 52\ 55];$

Estos datos se han introducido para su tratamiento estadístico en el programa “Solo de la empresa Eigenvector Research”, agregando por un lado los fragmentos de 1024 muestras y por otro lado los resultados objetivo representados de forma binaria. El código binario utilizado toma el valor de “1” cuando se ha detectado actividad y “0” en caso contrario.

A continuación en el la Figura 13 se puede apreciar los resultados obtenidos mediante al análisis estadístico PCA.

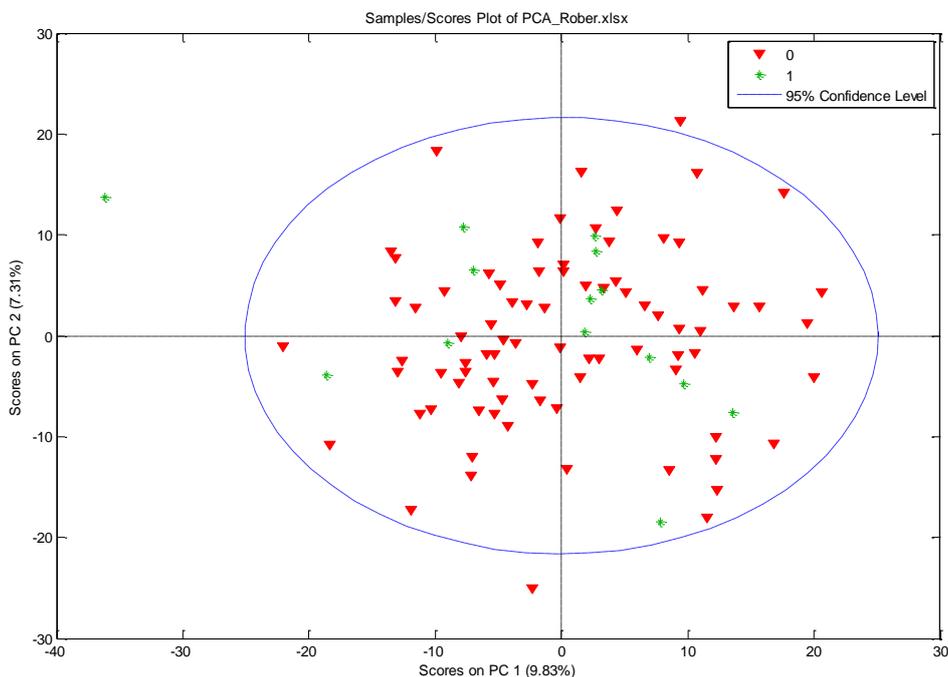


Figura 13. PCA realizada directamente sobre fragmentos de 1024 muestras

Los resultados obtenidos no muestran una diferenciación entre elementos, por lo tanto se descarta también este tipo de análisis.

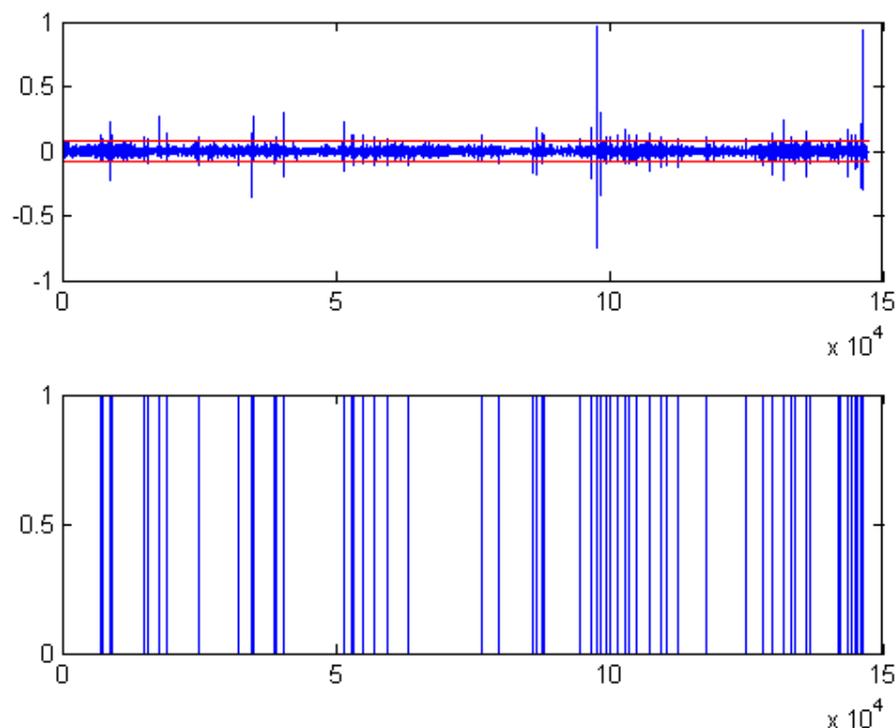
**Análisis mediante umbral de intensidad sonora**

Tras el estudio de diferentes fragmentos de audio con el software Audacity se comprobó que existe una mayor intensidad sonora cuando el movimiento abdominal de la pupa alcanza su máxima elongación, repitiéndose a intervalos durante un cierto periodo de tiempo.

Este tipo de procesado resulta muy interesante debido a la sencillez de implementación y por lo tanto un reducido coste hardware. Éste se encargaría tan solo de detectar las muestras que sobrepasen un cierto umbral catalogándolas como “rebase”. Los resultados son procesados mediante un algoritmo que se centra en encontrar un cierto patrón repetitivo en los “rebases” almacenados.

Por otro lado, se exige que el algoritmo posea la versatilidad de diferenciar un pico de intensidad del resto de audio. Para conseguir este propósito se han estudiado distintos audios, llegando a la conclusión de que existen variaciones que pueden deberse a un aumento del ruido ambiente, a la lejanía de las señales captadas o incluso a una diferente configuración del propio sistema hardware. Para conseguir una referencia umbral en los experimentos se ha empleado un algoritmo matemático basado en la desviación típica de un fragmento de audio. Experimentalmente se ha podido comprobar que tomando dos veces la desviación típica se consigue dicho objetivo.

A continuación, en la Figura 14 parte superior se muestra una representación temporal de un audio con presencia del movimiento abdominal de la pupa (*trazo azul*), mostrando también el umbral obtenido marcado en trazo rojo. En la parte inferior de la Figura 14 se ha representado las veces que se sobrepasa el umbral con la esperanza de ver si este produce o no un patrón repetitivo.



**Figura 14. Detección mediante umbral de un fragmento de audio que presenta sonidos del movimiento abdominal de la pupa**

Por otro lado, se ha realizado la misma prueba empleando dos audios de una grabación de voz. Los resultados se pueden apreciar en la Figura 15 y Figura 16.

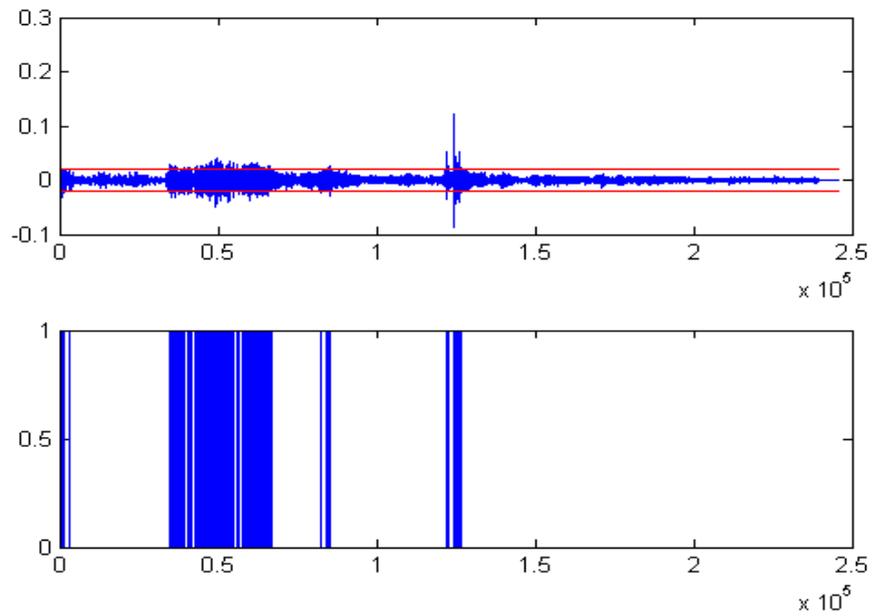


Figura 15. Detección mediante umbral de un fragmento de audio con una voz de fondo

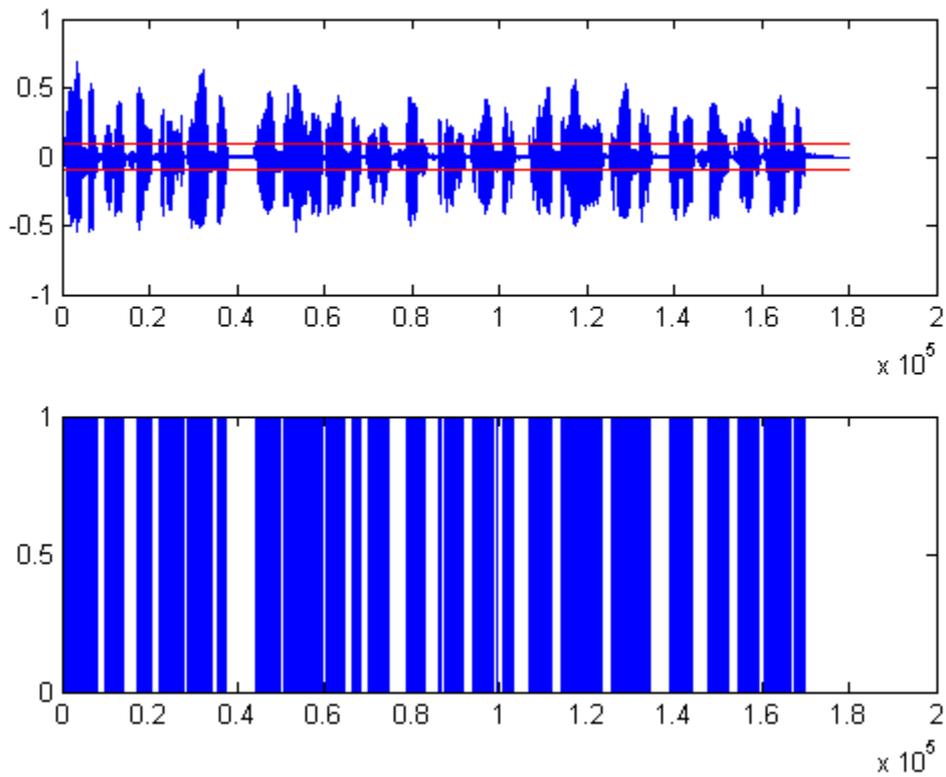


Figura 16. Detección mediante umbral de un fragmento de audio con una voz clara

Tras la experiencia realizada se concluye que el algoritmo detecta correctamente los picos de audio. El problema que se presenta es que en los audios "negativos" también aparecen aciertos y resulta compleja su distinción. Además este tipo de procesado no es capaz de discriminar un sonido de otro por su forma o frecuencia, siendo altamente probable encontrar un gran número de falsas alarmas. Este algoritmo por tanto queda descartado como núcleo de procesado. No obstante, puede ser empleado para una detección previa, de forma que si en el fragmento de audio a analizar no existe ninguna actividad de ningún tipo el sistema permanece en espera.

Los resultados obtenidos mediante técnicas de tratamiento se señalan en el dominio del tiempo ponen de manifiesto la necesidad de emplear otras estrategias de procesado.

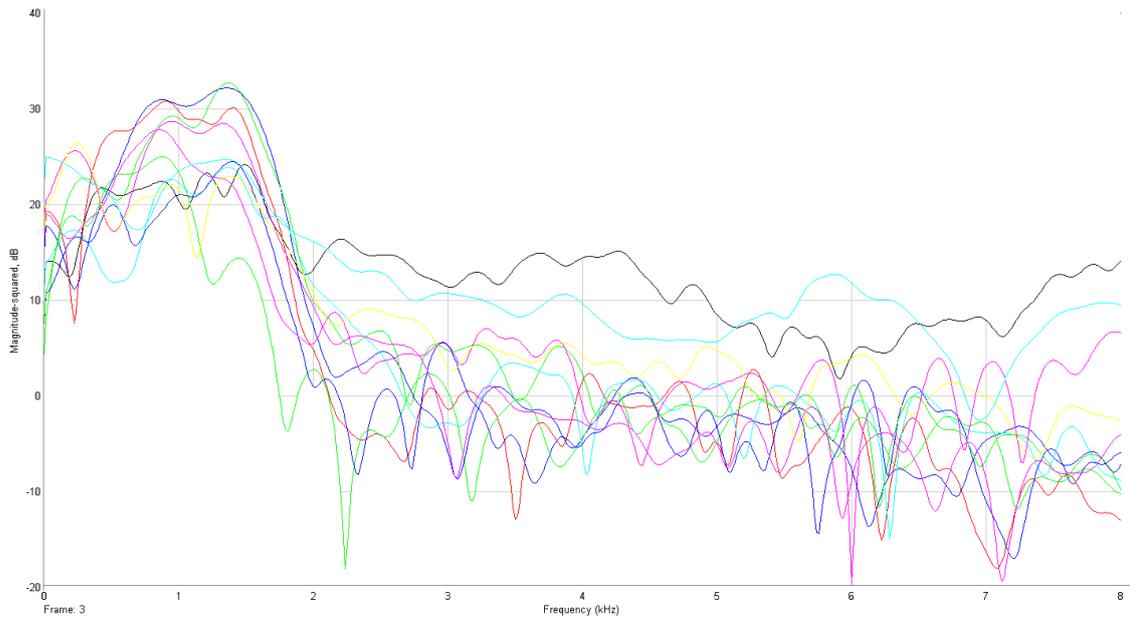
### **3.2.2 Análisis en el dominio de la frecuencia**

El estudio en frecuencias es una técnica muy empleada en el tratamiento y procesado de audio. Para la conversión de la señal temporal al dominio de la frecuencia se ha utilizado la Transformada Rápida de Fourier (FFT). Este algoritmo se encarga de realizar un análisis en frecuencia del fragmento de audio a analizar, descomponiendo la señal como una suma de sinusoidales, asociando a cada frecuencia una energía o potencia.

El estudio en frecuencia de señales ha sido empleado en multitud de ocasiones para el reconocimiento de patrones de audio. Este análisis permite distinguir diferentes tipos de insectos como apunta el investigador Makin [*Makin et al. 2008*] en donde se pueden discriminar por su perfil en frecuencia varios tipos de larvas.

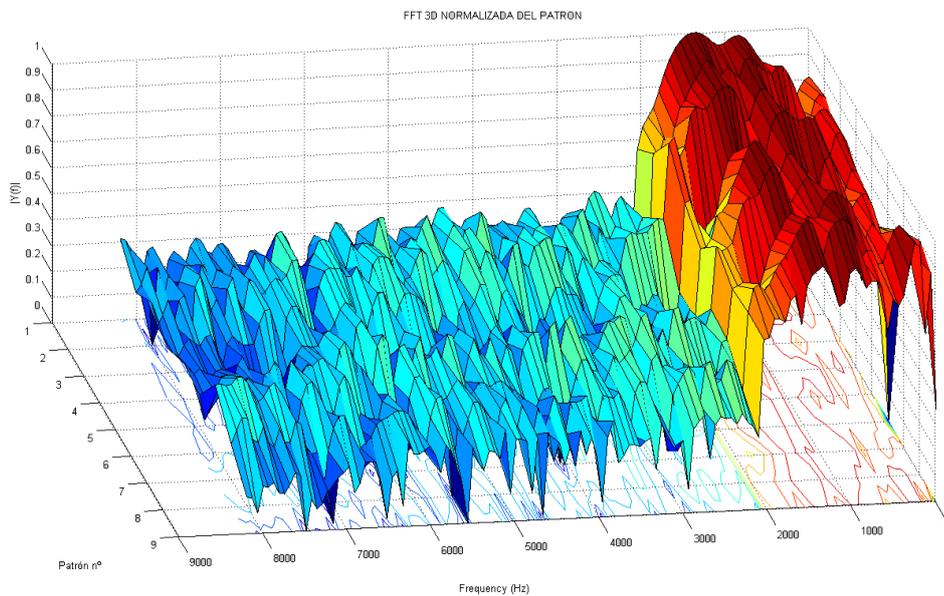
#### **Espectro asociado del sonido patrón**

El primer objetivo consiste en la caracterización del audio "patrón" en el dominio de la frecuencia. En la Figura 17 se ha representado de forma superpuesta diferentes audios patrón. Realizando un análisis comparativo de unos pocos patrones se puede apreciar una gran disparidad en el perfil de frecuencias adoptado. Para conseguir una unificación de los patrones, tratando de que éstos sean los más similares unos a otros, se ha realizado una normalización.



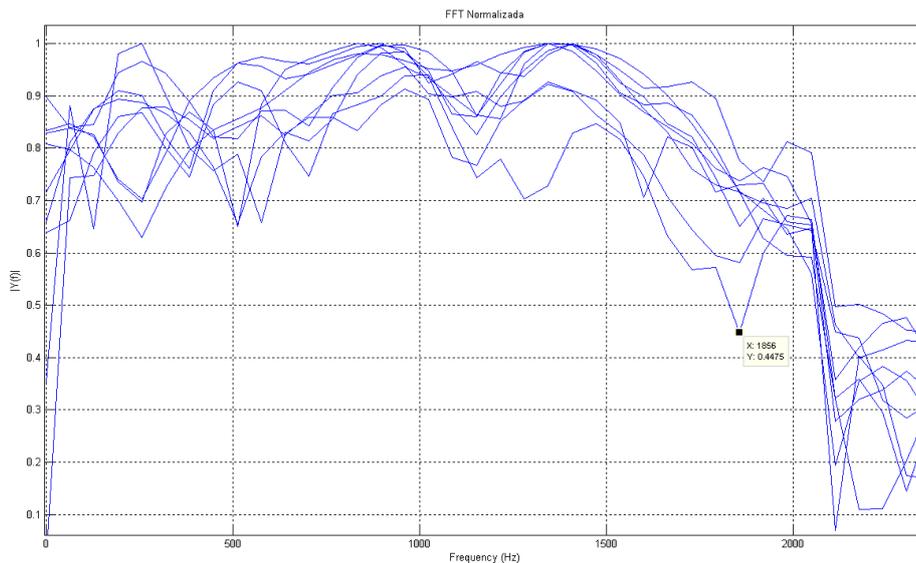
**Figura 17. Análisis espectral de las señales patrón del movimiento abdominal de la pupa**

Seguidamente en la Figura 18 se ha representado los resultados obtenidos fruto de la normalización en un gráfico tridimensional. Dicha representación muestra unas pautas de comportamiento comunes, indicando una clara actividad hasta 2200Hz y la ausencia de la misma a partir de esta frecuencia.



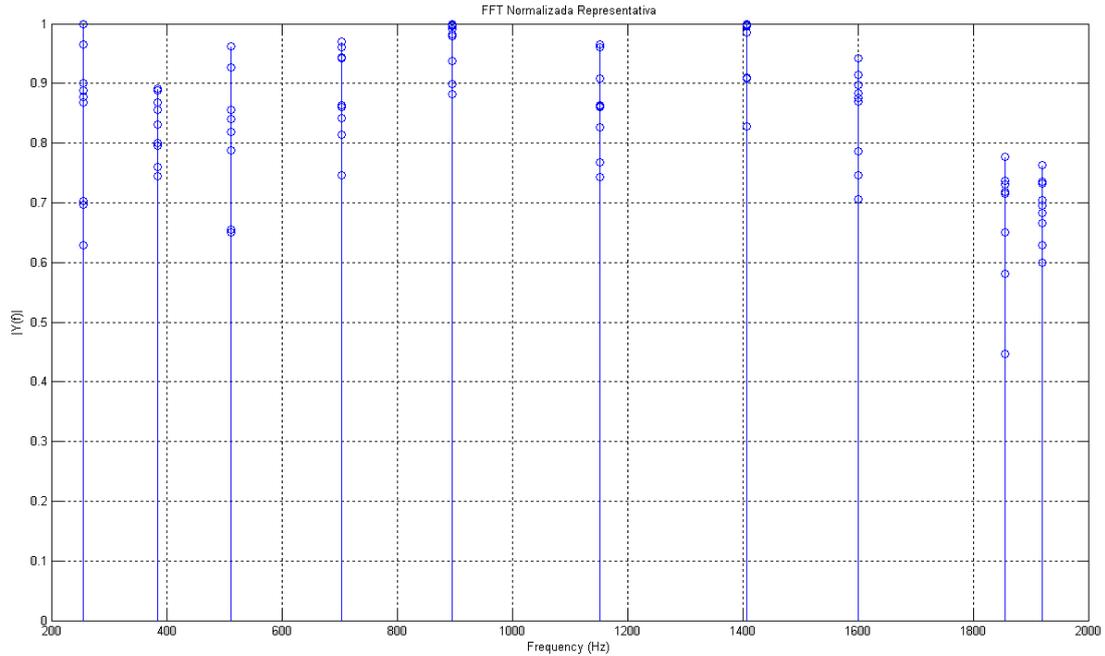
**Figura 18. Representación tridimensional de la FFT normalizada.**

Realizando una representación en 2D (véase *Figura 19*) y superponiendo unos pocos audios patrón, se puede apreciar la existencia de varias frecuencias características. A simple vista se pueden destacar dos máximos a 800Hz y 1400Hz y mínimos locales a 400Hz y 1100Hz. Además de estas frecuencias se pueden tomar otras a partir de 2200Hz ya que resulta tan importante el registro de actividad a frecuencias bajas como la ausencia de la misma a altas frecuencias.



**Figura 19. Representación 2D de la FFT normalizada de los patrones.**

A continuación se ha realizado una representación de los puntos de interés para poder apreciar la dispersión. Como se puede observar en la *Figura 20*, se podría trazar una franja útil para su detección utilizándola a modo de umbral. De esta forma, señales que estén dentro del umbral deberían tener una alta probabilidad de poseer sonidos relacionados con el movimiento abdominal de la pupa.



**Figura 20. Representación gráfica de los puntos de interés de la fft normalizada**

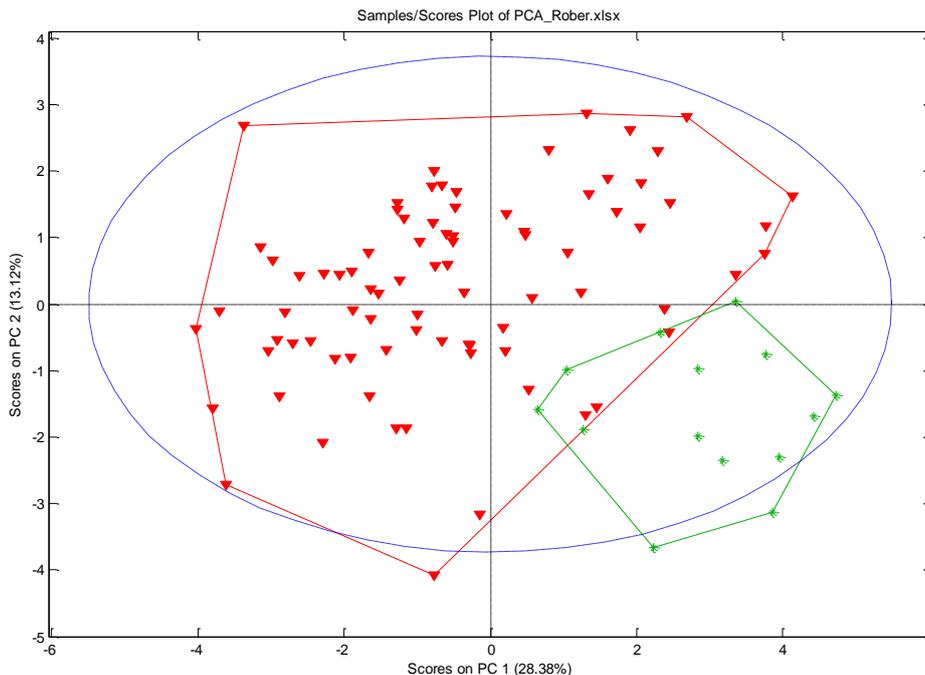
Una vez caracterizado y extraídas las frecuencias de interés de los audios de tipo patrón, el estudio se centra en su tratamiento para su posterior reconocimiento.

### **Análisis estadístico del espectro de frecuencias PCA**

En primer lugar se decide realizar un análisis estadístico del tipo PCA. Al igual que en el análisis PCA realizado en el dominio temporal se ha tomado un audio troceándolo en varios fragmentos y clasificando cada uno de ellos según si posee actividad del movimiento abdominal “1” o no “0”. Esta vez, para limitar la complejidad del modelo a utilizar, se han tomado varias frecuencias de interés.

$f_{rep}=[256, 384, 512, 704, 896, 1152, 1408, 1600, 1856, 1920, 2240, 2912, 3456, 4128, 4496, 5136, 6208];$

Analizando el gráfico PCA de la Figura 21 se puede apreciar que existe una diferenciación en las componentes en frecuencia con presencia del movimiento abdominal formando un área rodeada en verde, de las que no poseen esta característica rodeadas en rojo.

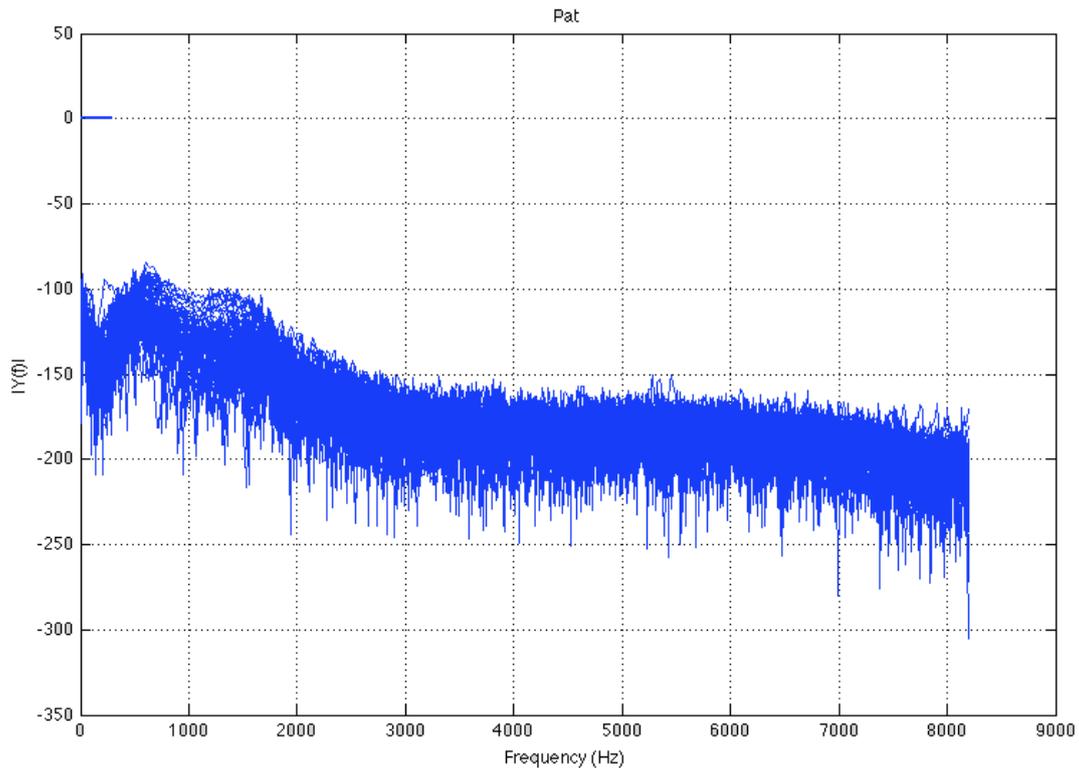


**Figura 21. PCA realizada sobre las FFTs de fragmentos de 1024 muestras**

Pese a los buenos resultados obtenidos, este análisis limita el modelo tan solo a los casos conocidos. Aunque el aporte de más patrones con actividad abdominal resulta una tarea laboriosa pero posible, no es viable acotar el resto de audios ya que forman un conjunto muy extenso. Por este motivo resulta muy complejo realizar un modelo completo con esta técnica.

### **Estudio detallado de multitud de patrones**

Tras los buenos resultados estadísticos obtenidos con el análisis PCA se decide buscar un modelo de audio patrón lo suficientemente representativo. Para ello se ha realizado un análisis comparativo como se aprecia en la Figura 22. Representación en frecuencia de 50 patrones de audio. Revisando el espectro normalizado de 50 patrones de audio se puede concluir que el sonido del movimiento de la pupa produce un espectro muy variado. Esta variación puede deberse a la energía del impulso del movimiento, a la distancia que se encuentre el coleóptero del micrófono, incluso a la posible configuración del equipo.



**Figura 22. Representación en frecuencia de 50 patrones de audio**

Sin embargo pese a las discrepancias mostradas en la Figura 22, la representación muestra cierto comportamiento en común. La superposición de audios afianza la identificación de una actividad clara del sonido del movimiento de la pupa con una frecuencia máxima de 2200Hz y una clara inactividad a partir de esta frecuencia.

Los límites de los distintos patrones superpuestos se han representado con un trazo rojo como se observa en la Figura 23. Para la identificación de estos patrones de audio existe la posibilidad de realizar una discriminación mediante un umbral asociado a una selección de frecuencias. Analizando la banda umbral obtenida en la Figura 23 se puede apreciar que la diferencia en una determinada frecuencia entre la mayor y menor amplitud es de 90dB. Este umbral se considera muy amplio, permitiendo que audios con espectros de formas totalmente diferentes no sean discriminados. Por otro lado, los máximos y mínimos descritos tras el estudio con pocos patrones quedan enmascarados no siendo representativos.

Como conclusión del análisis en frecuencia se puede destacar que existen ciertos rasgos característicos en las señales patrones analizadas, no obstante el sonido producido por el movimiento abdominal de la pupa no deja de ser un movimiento mecánico producido por el golpeteo de un cuerpo contra otro que presenta cierta aleatoriedad. Por esta razón el análisis no queda concluido. En el siguiente punto se realiza un estudio tiempo-frecuencia aprovechando el conocimiento adquirido en ambos dominios.

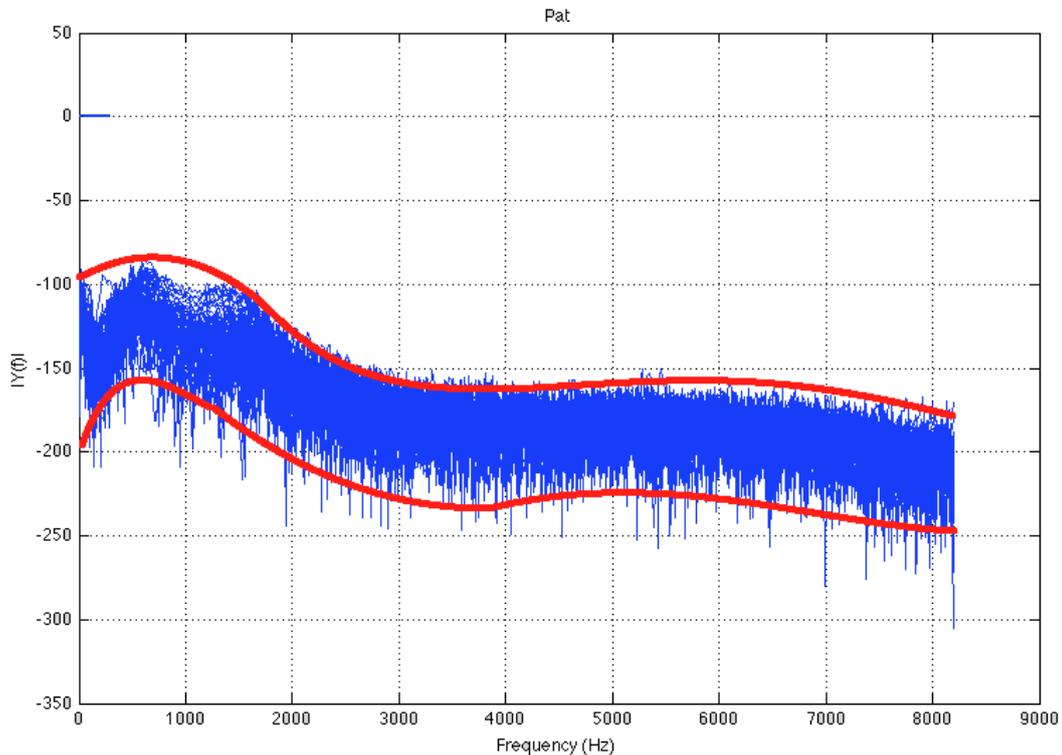


Figura 23. Representación en frecuencia de 50 patrones de audio con franja de acotación

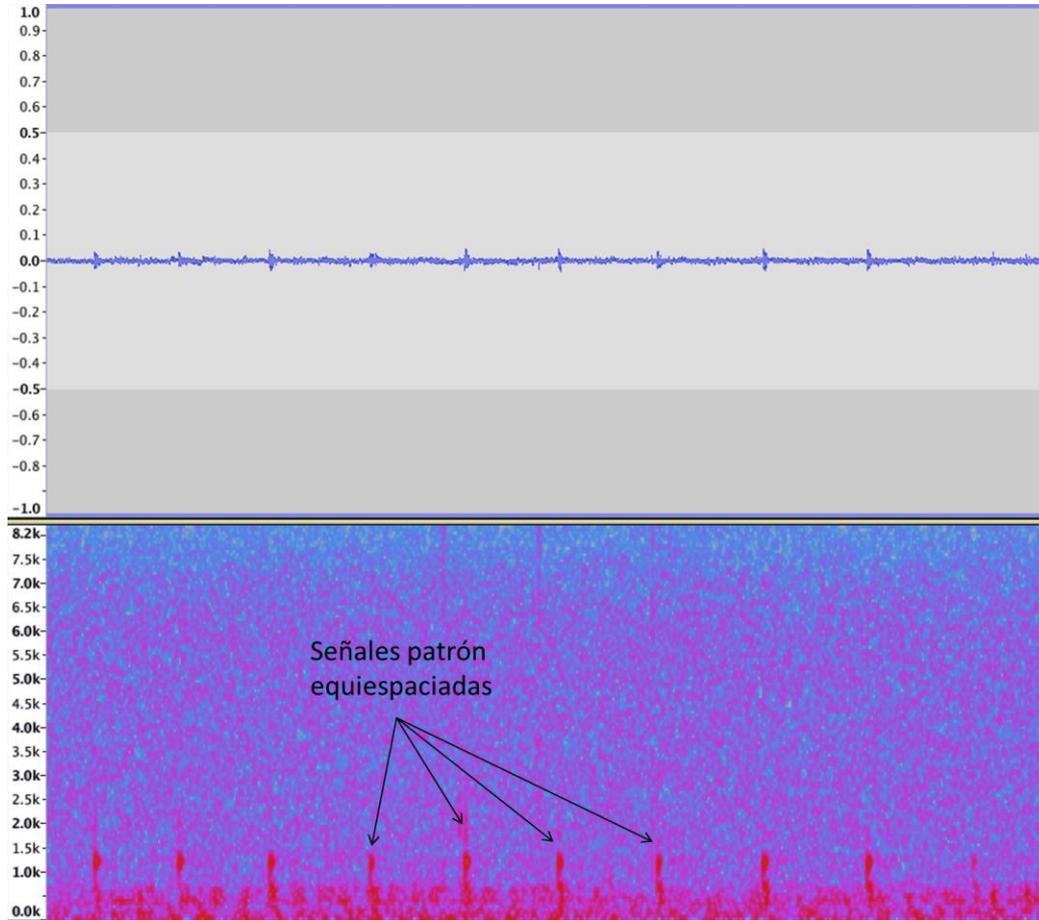
### 3.2.3 Análisis tiempo frecuencia (STFT)

Una de las técnicas más empleada para el análisis de audios continuos en el tiempo es el espectrograma. Básicamente esta técnica consiste en una representación gráfica del espectro de frecuencias, donde normalmente el eje horizontal representa el tiempo, el eje vertical la frecuencia y mediante una escala de grises o colores la amplitud de las señales.

Existen numerosos artículos de investigación que emplean esta técnica para la identificación de insectos en plantas [Mankin 2010 ; Al-Manie et al. 2007 ; Potamitis et al. 2009]. Esta técnica ha tenido una gran acogida en el sector de la investigación para la identificación de patrones ya que combina el análisis en el dominio temporal con el dominio en frecuencia, pudiendo realizar un estudio global del audio.

Para verificar dicha premisa se ha realizado un estudio previo de un audio captado en las pruebas de campo con la herramienta de software Audacity. En la parte superior de la Figura 24 se muestra la representación temporal y en la inferior se aprecia la representación del tipo espectrograma. Analizando el espectrograma se puede apreciar de forma clara el sonido fruto de la actividad del movimiento de la pupa, afianzando la hipótesis de que esta actividad presenta una elevada potencia espectral hasta 2200Hz. Además de esto, también se pudo apreciar cómo el patrón se repite con una cierta periodicidad. Basándose en estos dos comportamientos descritos el algoritmo a desarrollar

debe reconocer un impulso patrón aislado y además detectar su periodicidad en el tiempo. El algoritmo propuesto presenta un mayor grado de robustez que los anteriormente descritos, aumentado por tanto la fiabilidad de detección.



**Figura 24. Movimiento abdominal de la pupa en representación temporal (superior) y representación tipo espectrograma (inferior)**

A continuación, el objetivo se centra en conseguir una representación tiempo-frecuencia como la mostrada en la Figura 24. Para la realización del espectrograma el software Audacity realiza un procesado en frecuencia del tipo FFT dividiendo el audio en fragmentos de igual medida con cierto solapado. Esta técnica se denomina (STFT) o transformada de Fourier de corto tiempo. La STFT cumple la siguiente ecuación (2).

$$X[n, \lambda] = \sum_{m=-\infty}^{\infty} \omega[m]x[n + m]e^{-j\lambda m} \quad (2)$$

Donde  $\omega[m]$  es la ventana en la que se realiza la FFT y  $n$  es el desplazamiento hacia la izquierda de la señal que estamos analizando. El resultado de la STFT es complejo y se suele representar tan sólo el módulo (*o la fase*). En el caso de que se quiera tener buena

resolución en frecuencia se elegirán ventanas largas quedando las señales estacionarias completamente representadas, con el inconveniente de que implican una pérdida de resolución temporal. Caso contrario ocurre para la detección de señales de corta duración o no estacionarias, siendo necesario emplear ventanas con pocas muestras. El objetivo es llegar a un compromiso entre resolución temporal y resolución en frecuencia que permita discernir tanto las señales estacionarias como las no estacionarias.

### Procesado de los datos tiempo frecuencia

A continuación se describe paso a paso el método de procesado realizado mediante la herramienta Matlab a partir de los audios recopilados.

En primer lugar se han dividido los audios en fracciones temporales de medio segundo. Estos fragmentos se agrupan en una matriz para su posterior tratamiento. En la Figura 25 se ha representado esquemáticamente la operación descrita.

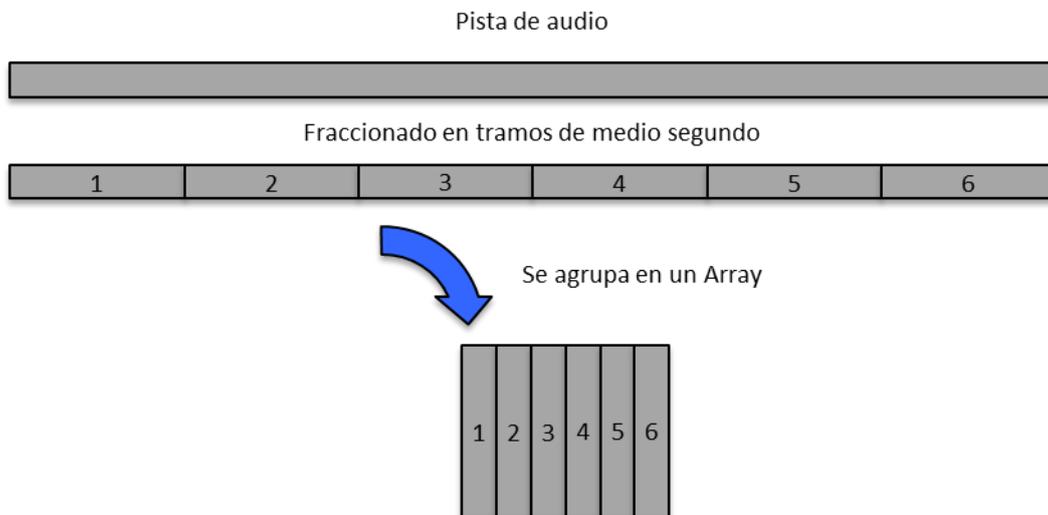


Figura 25. Almacenado de las señales

Cada fracción temporal es tratada de forma independiente, aplicándole una STFT. Para la realización de la STFT se ha utilizado una relación de solapado del 75%.

El escalón de solapado (*véase ecuaciones 3, 4 y 5*) hace referencia al número de muestras necesario que marcan el avance temporal de la señal, siendo en este caso el escalón de 6.22 ms. Con el avance temporal queda fijada la denominada resolución temporal de la STFT. En cuanto a la resolución espectral, se ha determinado de forma experimental un total de 512 muestras. La distribución descrita se ha organizado en formato matricial, disponiendo de un total de 512 filas por 78 columnas. En este caso cada fila hace referencia a un dato de las distintas FFTs por lo tanto una columna agrupa todos

los datos necesarios para realizar una FFT. En la Figura 26, mostrada abajo, queda representada la distribución de datos descrita.

$$N_{fft} = \frac{Length}{Time_{Long}/Frec\_Temp} \quad (3)$$

Obtención del solapado:

$$Overlap = \frac{N_{fft}}{Overlap\_Ratio} \quad (4)$$

Los pasos resultantes se obtienen:

$$Step = N_{fft} - Overlap \quad (5)$$

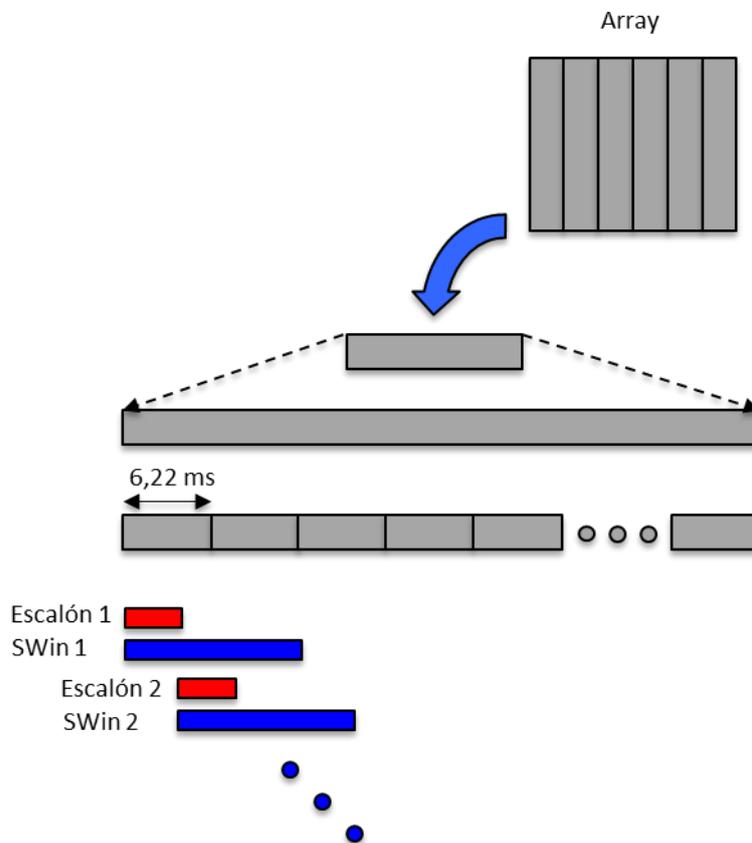


Figura 26. Datalle del solapado

A continuación a cada ventana temporal se le ha aplicado un enventanado de tipo Hamming. Como indica el investigador Potamitis [Potamitis et al. 2009] la ventana Hamming se aplica para reducir la distorsión espectral causada por los abruptos cambios de amplitud de la señal. Seguidamente se ha realizado un procesamiento de señal del tipo FFT múltiple almacenando el resultado en una matriz de 78 columnas por 256 filas. Al igual

que en la matriz anterior las columnas hacen referencia al paso de avance temporal, en cambio las filas hacen referencia a las distintas frecuencias.

A continuación se detallan los cálculos que determinan el tiempo total de la ventana STFT:

$$F_s = 16384 \text{ Hz} \rightarrow 16384 \text{ muestras por segundo}$$

$$\text{Solapado} = 102 \text{ muestras}$$

$$\text{Total muestras} = 102 \cdot 77 + 512 = 8366 \text{ muestras}$$

$$\text{Tiempo utilizado} = \frac{8366}{16384} = 0.51 \text{ s}$$

En la Figura 27 se muestra el espectrograma de un fragmento de audio procesado con la herramienta Matlab. Como se puede apreciar el patrón de audio que produce el movimiento abdominal de la pupa queda perfectamente representado. Esta técnica permite un cambio dimensional del problema ya que existe la posibilidad de tratar el audio con técnicas de procesamiento de imagen. No obstante, para que no se pierda la vinculación con el audio a analizar se van a seguir haciendo referencias a las características auditivas.

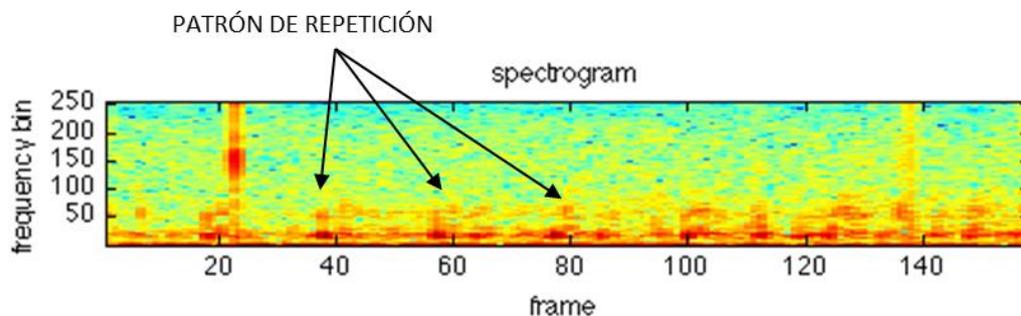


Figura 27. Patrón de repetición en espectrograma con solapado al 75%, resolución temporal de 6,22 ms y resolución en frecuencia de 256 valores

El siguiente paso en el tratamiento del audio ha consistido en la eliminación de las frecuencias bajas, ya que en estas se localiza generalmente el ruido ambiente [Mankin. 2010]. La matriz resultante queda reducida a 241 filas manteniendo las 78 columnas iniciales. Seguidamente se ha realizado una segmentación de la matriz de audio eliminando las señales del espectro con potencias inferiores a un umbral y manteniendo el espectro de mayor potencia o de interés. Dicha eliminación consiste en sustituir el valor espectral obtenido por el valor cero. El resultado se puede apreciar en la Figura 28.

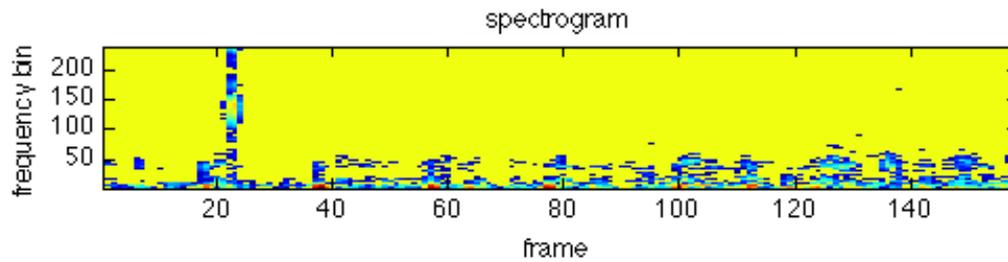


Figura 28. Segmentado del espectrograma.

Una vez alcanzada esta etapa se ha realizado un procesado a fin de obtener una simplificación matricial. Esta técnica ha sido objeto de multitud de artículos como así afirma Schutte [Schutte *et al.* 2007]. Para conseguir esta simplificación se ha utilizado un procesado que consiste en analizar las componentes en frecuencia distintas a cero por cada fracción temporal, aplicando un algoritmo de decisión. El algoritmo de decisión devuelve como resultado una matriz binaria con idénticas dimensiones a la original. Los valores posibles de esta última matriz son “0” representado en color azul y “1” representado por el color rojo. El resultado queda expuesto en la Figura 29.

Basándose en las características extraídas del estudio de componentes en frecuencia descrito anteriormente, los patrones de audio poseen componentes de alta potencia espectral hasta frecuencias de 2200 Hz. Utilizando esta característica se ha considerado un algoritmo de decisión que contempla tres supuestos para completar la matriz binaria:

- Se ha producido alta actividad en la potencia espectral para frecuencias inferiores de 2200Hz, y baja actividad para frecuencias superiores a esta. Como resultado se obtiene “1” para la zona de baja frecuencias y “0” en la zona de altas.
- Se ha producido alta actividad en la potencia espectral para frecuencias superiores a 2200Hz, y baja actividad para frecuencias inferiores a esta. Como resultado se obtiene “0” en la zona de bajas frecuencias y “1” en la zona de altas.
- En los demás casos, tanto la zona de altas frecuencias como las de bajas se completa con “0”.

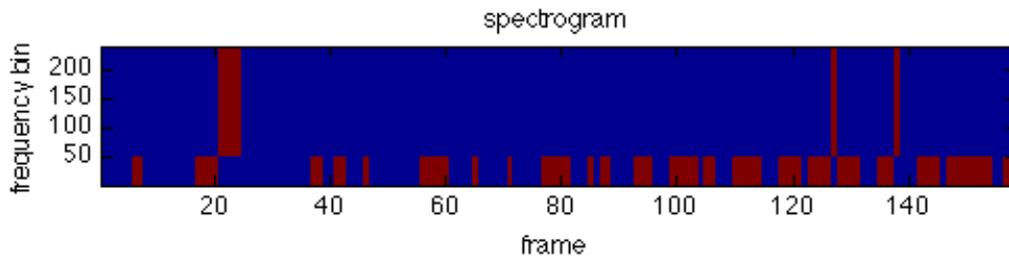


Figura 29. Simplificación binaria del espectrograma

En la Figura 30 se puede apreciar de forma comparativa los distintos pasos descritos. Como se puede observar la señal se ha simplificado a un formato binario, quedando lista para continuar con el procesado.

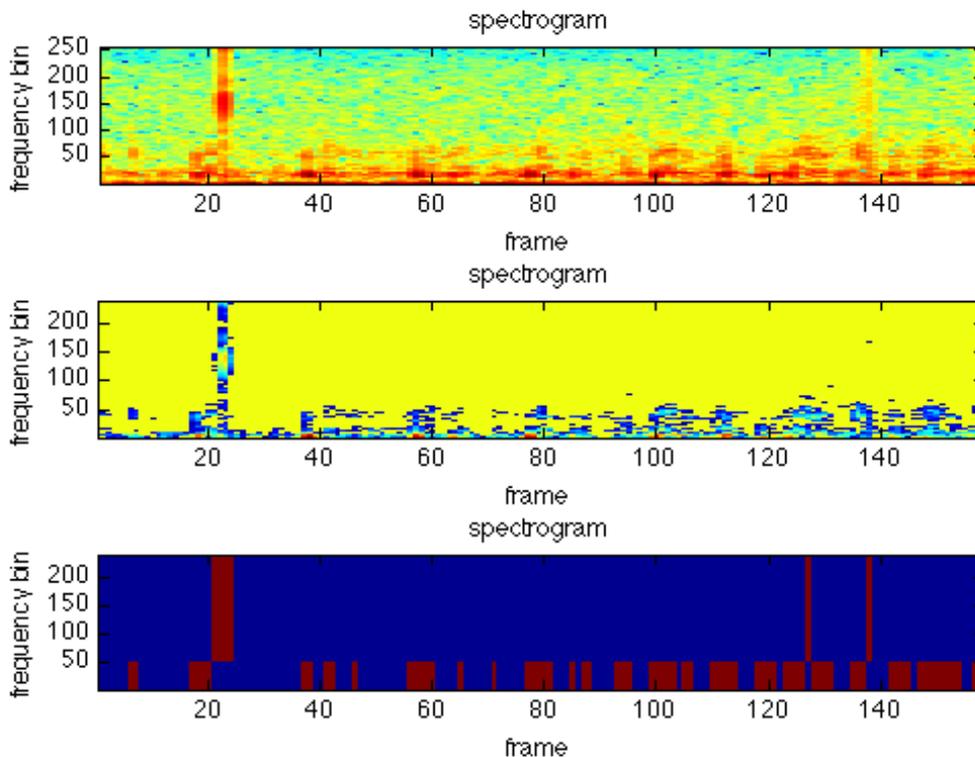


Figura 30. Representación de un espectrograma de una señal (superior). Segmentado del espectrograma (central). Simplificación binaria del espectrograma (inferior)

La última etapa de procesado de la señal está basada en un algoritmo que en sí puede considerarse un sistema experto. Esta técnica emula el razonamiento de un experto en un dominio concreto. Por lo tanto el sistema experto propuesto esta basado en una serie de reglas, fruto en la experiencia tras la detección manual del patón empleando técnicas espectrales.

El algoritmo de decisión que se describe a continuación principalmente trabaja con dos vectores (Alta y Baja), como se muestra en la Figura 31. La extracción de estos vectores es directa, pudiéndose tomar por ejemplo la primera y última fila de esta matriz.

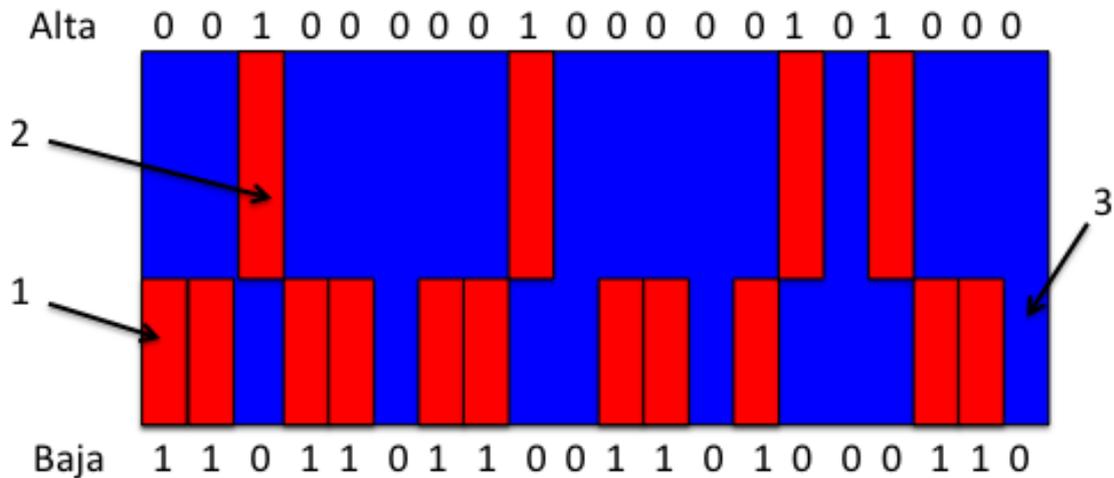


Figura 31. Representación gráfica de la extracción de los vectores “Alta” y “Baja”

Las decisiones tomadas están relacionadas con la frecuencia en la que se producen los impulsos y teniendo en cuenta de si se ha producido actividad o no en las frecuencias de interés o en frecuencias superiores. Una vez recopilada esta información el algoritmo se encarga de contar los pulsos altos y bajos. Para la obtención de un acierto es necesario un número alto de impulsos en bajas frecuencias y un número bajo de impulsos en altas frecuencias, de este modo se realiza un primer descarte. Los audios que hayan superado esta primera fase son sometidos a una segunda fase de descarte, en donde se analizan los tamaños de impulsos de la parte de alta frecuencia. De este modo los impulsos demasiado largos anulan el acierto. Por último se realiza una tercera fase en donde se analizan los audios de la parte baja de frecuencias, analizando la duración de los impulsos así como el intervalo entre ellos.

En la

Tabla 3. Tabla resumen mostrando los resultados obtenidos se puede comprobar que este algoritmo funciona conforme lo esperado devolviendo un alto número de aciertos en señales positivas exceptuando los audios “Probando0” y “Probando5”, en los cuales la potencia o repetición no se adaptan a los mínimos exigidos. Este mismo comportamiento podría apreciarse si un sujeto realizara una identificación de forma manual, ya que no posee una correcta información para su identificación. Por otro lado se puede apreciar un número de aciertos muy bajo o nulo, en el procesado de señales negativas, en el caso de la voz humana o el ruido de fondo.

## MASTER EN SISTEMAS ELECTRÓNICOS

Los estudios preliminares realizados muestran la efectividad del algoritmo empleado, quedando concluido el estudio de procesamiento offline por computador. El estudio de la tesis a partir de este punto se centra en el desarrollo hardware y software para su implementación.

	Audio	Descripción	Aciertos
Positivo	Probando0	Movimiento abdominal, intensidad débil	0
	Probando1	Movimiento abdominal, intensidad media	31
	Probando2	Movimiento abdominal, intensidad media	5
	Probando3	Movimiento abdominal, intensidad media	15
	Probando4	Movimiento abdominal, intensidad media	7
	Probando5	Movimiento abdominal, corto	0
	Probando6	Movimiento abdominal, intensidad media	3
Negativo	Probando7	Movimiento abdominal, intensidad media	54
	Probando8	Voz	0
	Probando9	Voces de fondo	0
	Probando10	Ruido de fondo	0
	Probando11	Ruido saturado del micrófono	0
	Probando12	Ruido de fondo	0
	Probando13	Voz	0

**Tabla 3. Tabla resumen mostrando los resultados obtenidos**

## 4 Implementación del sistema

### 4.1 Introducción

La implementación del sistema se ha realizado en la placa de desarrollo DE2-115 de Terasic. Esta placa incorpora una FPGA Cyclone IV modelo EP4CE115 de Altera. Se ha seleccionado esta plataforma por tener resuelta en gran parte del diseño hardware así como por sus prestaciones. Entre el hardware utilizado destaca el convertor de audio analógico integrado en un códec, la pantalla de visualización LCD 2x16 caracteres, los módulos de memoria RAM de hasta 126Mb así como los interruptores y pulsadores utilizados para la realimentación y configuración de la placa. En la Figura 32 se muestra detalladamente el hardware que incorpora dicha placa así como su aspecto general.

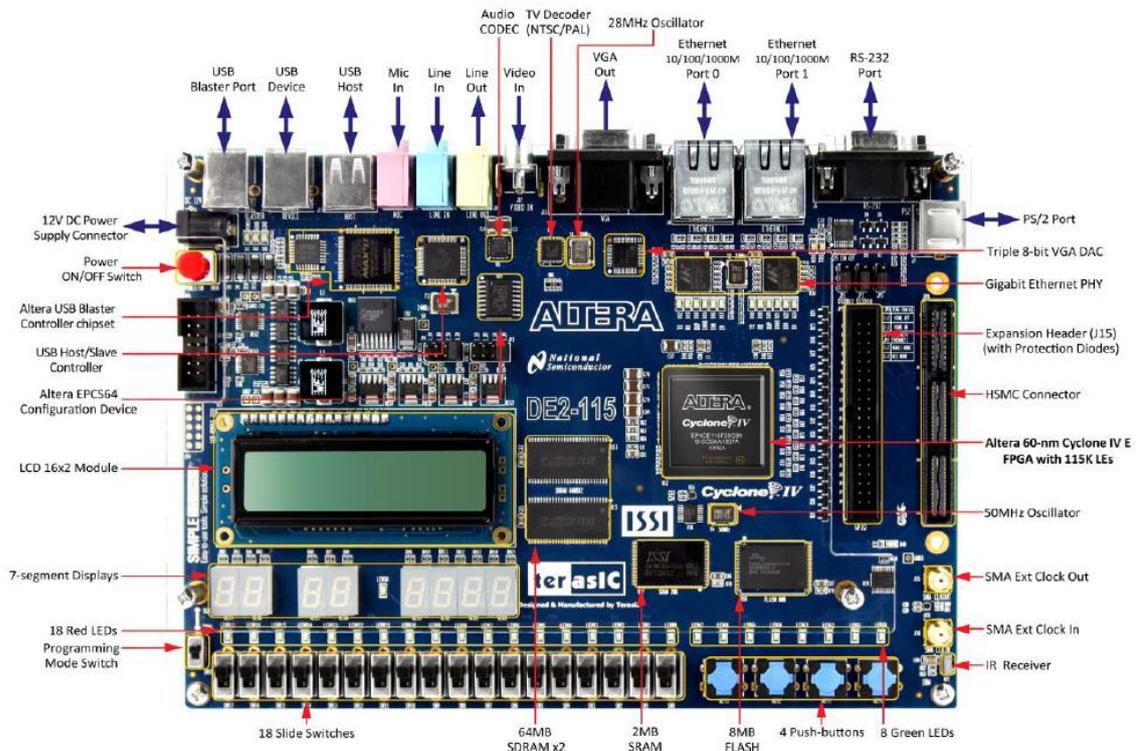


Figura 32. Placa de desarrollo DE2-115

Además del hardware externo ha sido necesario el desarrollo de una arquitectura hardware “Sistem On a Chip” (SOC) en la FPGA. En la Figura 33 se muestra el diagrama de bloques del SOC implementado. Como se aprecia, el sistema posee tres entradas. La primera entrada está compuesta por tres interruptores que establecen la configuración y comunicación con el equipo. El interruptor “SW0” permite al usuario seleccionar la fuente de datos del sistema accediendo a los datos serie del audio códec o a la entrada test. Por otro lado el “SW1” aplica el enventanado de tipo Hamming a la señal de entrada o test. Por último el “SW2” es el encargado de resetear el contador de aciertos que se describirá en el

apartado de software. La segunda entrada comunica con el dispositivo externo audio códec (*para más información véase el apartado 4.2.1*). Por último la entrada de test, que queda integrada en el SOC. Esta entrada está compuesta principalmente por una memoria ROM de 512 valores, en los cuales se ha almacenado una señal con dos tonos a distinta frecuencia. La señal resultante ha sido útil para el desarrollo y comprobación de los distintos bloques hardware así como el buen funcionamiento del procesado software.

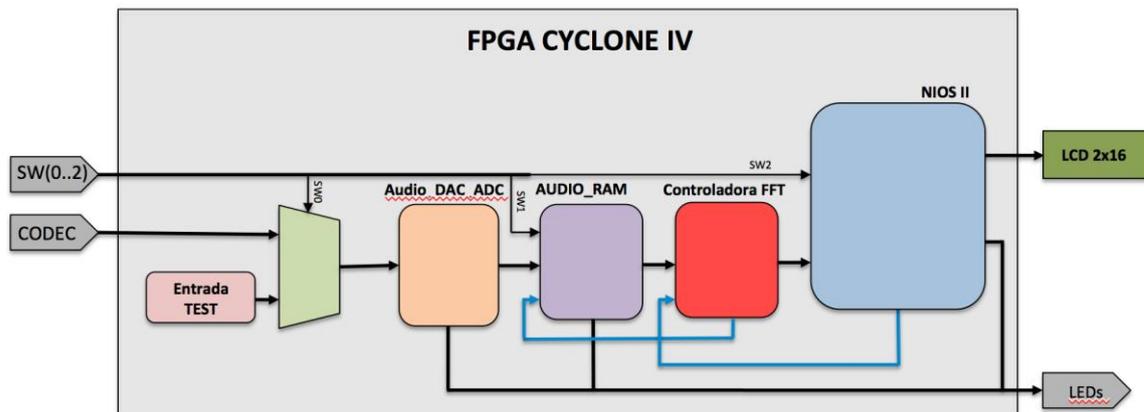


Figura 33. Diagrama de bloques general del sistema Hardware utilizado

Realizando una breve descripción de los bloques hardware implementados, en primer lugar se encuentra el módulo “Audio\_DAC\_ADC”. Este módulo es el encargado de generar las señales necesarias para la comunicación con el audio códec y de convertir los datos serie recibidos en datos paralelos con un ancho de palabra de 16 bits. Seguidamente los datos son almacenados en el bloque “AUDIO\_RAM”. Este bloque se encarga de preparar los datos para ser correctamente transmitidos al siguiente bloque. El flujo de datos avanza hacia el bloque “Controladora FFT”. Este bloque cumple la función de solicitar los datos de la memoria del bloque anterior y de realizar la operación de la FFT con los datos recibidos, llevando a cabo las operaciones necesarias para poder entregar los datos de forma adecuada al sistema NIOS II. Por último el sistema NIOS II procesa los datos y muestra los resultados en el display LDC y a través de los leds.

A continuación se comentan los módulos descritos de forma más detallada.

## 4.2 Implementación Hardware

### 4.2.1 Recursos hardware empleados de la placa de desarrollo

#### Configuración del integrado WM8731

El Códec de audio que integra la placa de desarrollo está encapsulado en el integrado WM8731. Este integrado presenta especificaciones de bajo consumo. Dispone de un conversor ADC sigma delta que permite una resolución de 24 bits con un muestreo que oscila entre 8kHz a 96kHz (véase Figura 34).

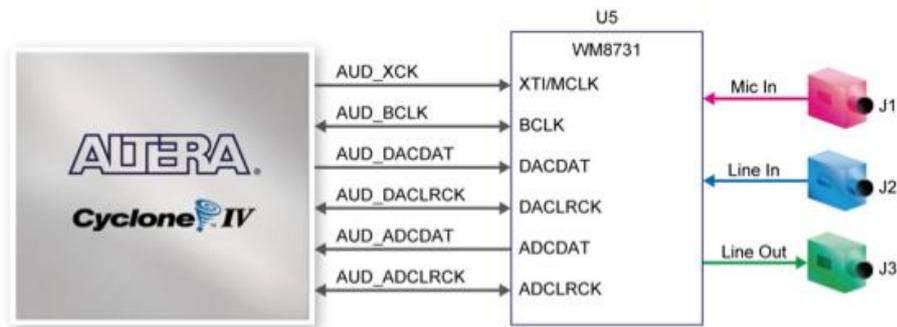


Figura 34. Esquema general del subsistema de audio.

El integrado WM8731 consta de múltiples módulos (véase Figura 35), siendo éstos configurables mediante dos o tres cables (I2C, SPI) en función del estado del pin “MODE” (‘0’ I2C, ‘1’ SIP)

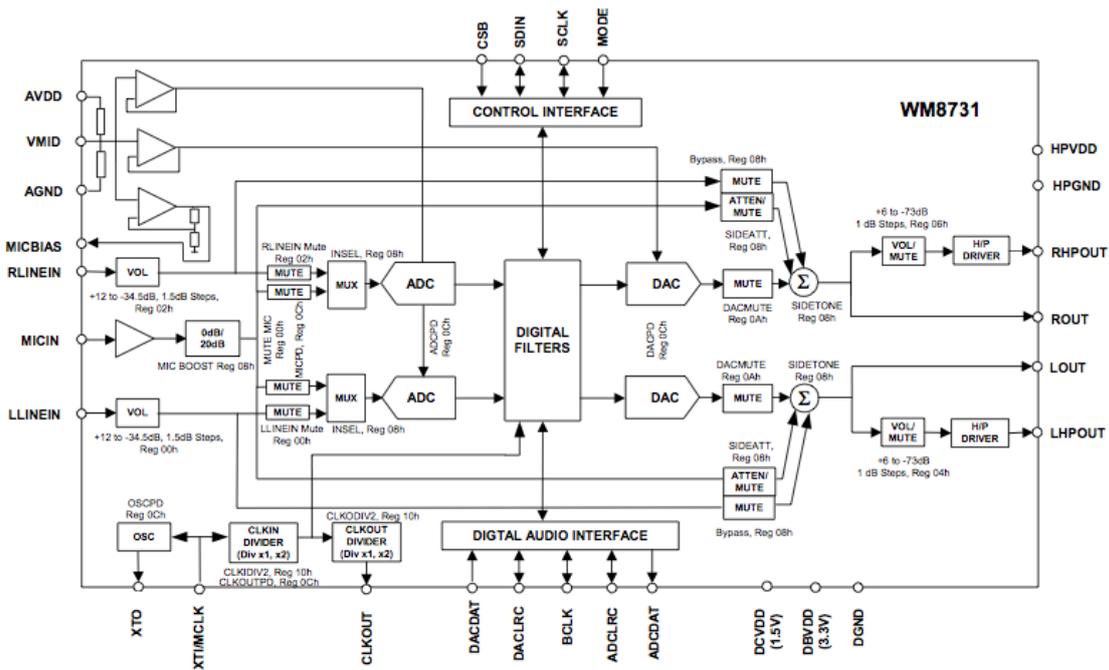


Figura 35. Esquema general del integrado WM8731

Además el fabricante facilita una tabla con los distintos bits de configuración. El objetivo es el de emplear sólo aquellos módulos que intervengan exclusivamente en diseño para así por un lado poder limitar el consumo de energía y por otro conseguir un rendimiento óptimo adaptándose a los requerimientos exigidos.

La configuración elegida para la comunicación ha sido la de esclavo, siendo la FPGA la encargada de gobernar todo el tránsito de datos. En la Figura 36 se aprecia las distintas conexiones entre ambos dispositivos.

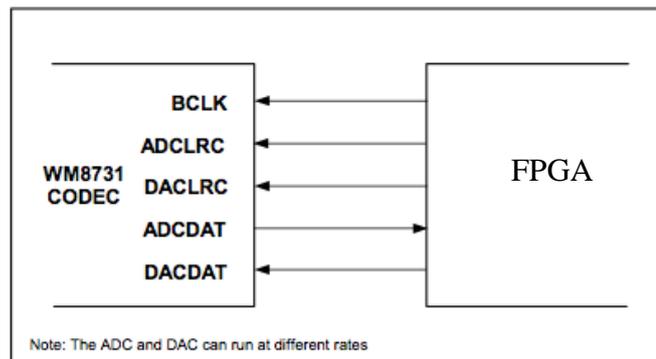


Figura 36. Diagrama del cableado ente el integrado WM8731 y la FPGA

Por otro lado el dispositivo ha de trabajar empleando la línea de entrada “LINE INPUT” configurable a través del bit “INSEL” (véase Figura 37).

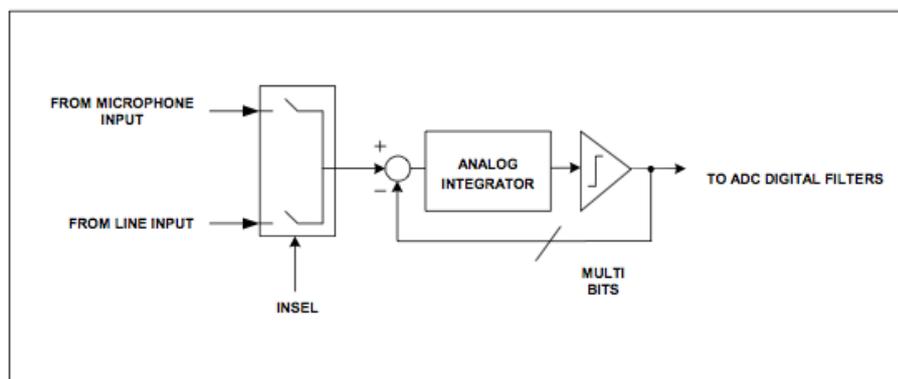


Figura 37. Diagrama de configuración para la captura de audio LINE IN del integrado WM8731

Como ya se mencionaba anteriormente, el módulo conversor digital analógico de audio “DACPD”, entrada de auriculares “ADCHPD” y el módulo de salida externa de audio “OUTPD” se han decidido desconectar por motivos de consumo. Además el oscilador “OSCPD” y el reloj de salida “CLKOUTPD” tampoco son necesarios ya que estos son suministrados directamente por la FPGA.

## MASTER EN SISTEMAS ELECTRÓNICOS

Por otro lado con la herramienta Matlab se ha comprobado que empleando una resolución de 16 bits es suficiente para realizar la identificación. Por esta razón se ha configurado el conversor ADC a una resolución de 16 bits.

En la Tabla 4 suministrada por el fabricante se especifica la frecuencia objetivo de muestreo y la configuración necesaria de los bits implicados para adaptarlo a los requerimientos exigidos.

SAMPLING RATE		MCLK FREQUENCY	SAMPLE RATE REGISTER SETTINGS					DIGITAL FILTER TYPE
ADC	DAC		BOSR	SR3	SR2	SR1	SR0	
kHz	kHz	MHz						
48	48	12.288	0 (256fs)	0	0	0	0	1
		18.432	1 (384fs)	0	0	0	0	
48	8	12.288	0 (256fs)	0	0	0	1	1
		18.432	1 (384fs)	0	0	0	1	
8	48	12.288	0 (256fs)	0	0	1	0	1
		18.432	1 (384fs)	0	0	1	0	
8	8	12.288	0 (256fs)	0	0	1	1	1
		18.432	1 (384fs)	0	0	1	1	
32	32	12.288	0 (256fs)	0	1	1	0	1
		18.432	1 (384fs)	0	1	1	0	
96	96	12.288	0 (128fs)	0	1	1	1	2
		18.432	1 (192fs)	0	1	1	1	
44.1	44.1	11.2896	0 (256fs)	1	0	0	0	1
		16.9344	1 (384fs)	1	0	0	0	
44.1	8 (Note 1)	11.2896	0 (256fs)	1	0	0	1	1
		16.9344	1 (384fs)	1	0	0	1	
8 (Note 1)	44.1	11.2896	0 (256fs)	1	0	1	0	1
		16.9344	1 (384fs)	1	0	1	0	
8 (Note 1)	8 (Note 1)	11.2896	0 (256fs)	1	0	1	1	1
		16.9344	1 (384fs)	1	0	1	1	
88.2	88.2	11.2896	0 (128fs)	1	1	1	1	2
		16.9344	1 (192fs)	1	1	1	1	

**Tabla 4. Tabla de configuración para el muestreo de audio del integrado WM8731**

La ecuación (6) indica los requerimientos hardware necesarios para el correcto funcionamiento del códec. En primer lugar se necesita un reloj MCLK de 18.432Mhz, el cual se ha generado con un PLL interno de la FPGA. Por otro lado se ha generado un reloj “BCLK” con un periodo de 18 ciclos de reloj mayor.

$$F_s = \frac{MCLK}{BOSR} \cdot Sample\ Rate$$

$$F_s = \frac{18.432Mhz}{384} \cdot \frac{2}{3} = \frac{18.432Mhz}{576} = \frac{18.432Mhz}{18 \cdot (2 \cdot 16bits)} = 32Khz \quad (6)$$

En cuanto a la elección del envío de datos, se ha elegido una justificación de los bits a izquierda transmitiendo en primer lugar el MSB. En la Figura 38 se aprecia el cronograma al que se deben ajustar las señales descritas anteriormente.

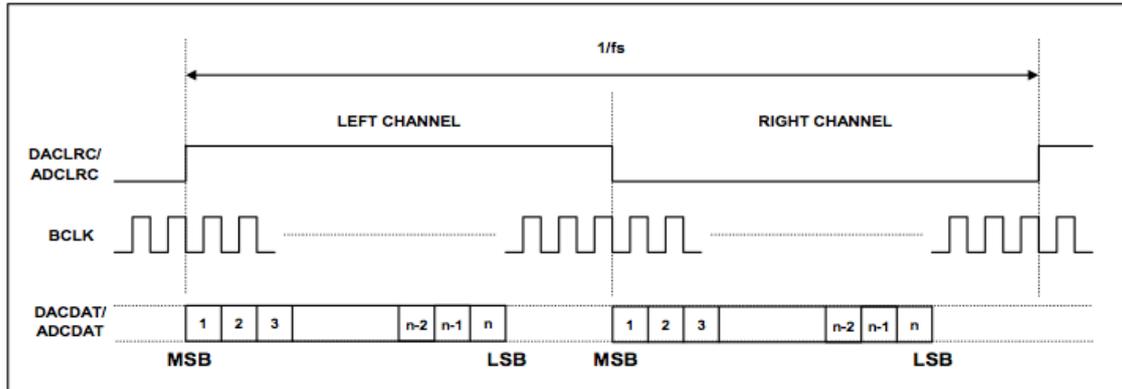


Figura 38. Cronograma del funcionamiento de la comunicación del integrado WM8731.

### Display LCD alfanumérico 2x16

La representación visual de los datos se realiza mediante el display LCD de 2x16 caracteres que integra la placa. En la Figura 39 se muestra una vista esquemática que representa las conexiones que se han realizado para su correcto funcionamiento.

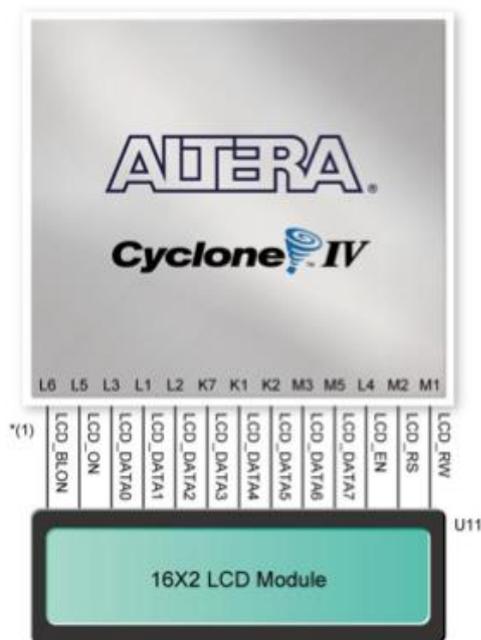


Figura 39. Diagrama de conexiones del display LCD en la placa DE2 115

#### 4.2.2 Arquitectura hardware diseñada

##### Bloque Audio DAC\_ADC

El bloque que se presenta cumple un doble objetivo. Por un lado genera todas las señales necesarias para la correcta comunicación con el códec anteriormente descrito y por otro lado se encarga de paralelizar los datos serie que provienen del códec a una frecuencia de salida de 32KHz.

##### Bloque Audio Ram

Tomando como referencia la tasa de muestreo de los audios almacenados ( $16.384\text{Hz}$ ) y para mantener un alto grado de similitud con el estudio Matlab realizado se ha decidido diezmar la frecuencia de muestro procedente del bloque Audio\_DAC\_ADC de 32kHz a 16kHz. Seguidamente los datos de entrada han sido almacenados para su posterior tratamiento mediante la técnica FFT. Para ello se dispone de un buffer circular compuesto por una memoria RAM de 1024 palabras con un ancho de 16 bits con doble puerto de lectura y escritura. El tamaño de esta memoria se ha elegido teniendo en cuenta que ha de soportar en la entrada un flujo de datos continuo de 16kHz mientras que en la salida ha de trabajar a 90MHz (véase Figura 40).

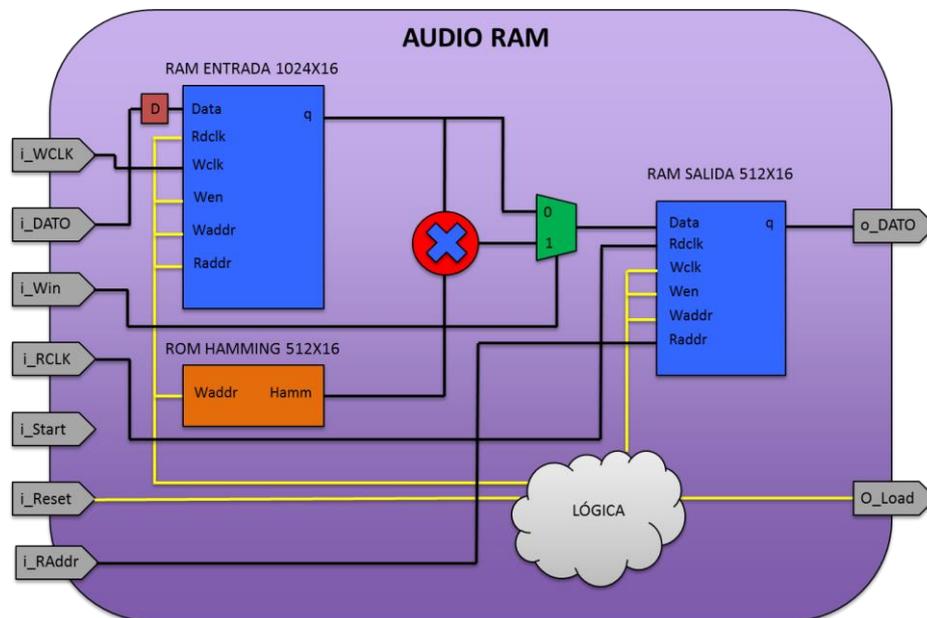


Figura 40. Bloque de almacenamiento de audio.

Inicialmente el sistema arranca con la memoria vacía. Ésta se va completando a la velocidad en que los datos se producen, que en este caso es de 16kHz. La lectura de la memoria RAM de entrada se lleva a cabo en siete pasos, dependiendo del estado que se encuentre el sistema de almacenamiento de la memoria (véase Figura 41). Cuando se ha

alcanzado el nivel séptimo la lógica interna del propio módulo se encarga de leer los primeros 512 valores de la memoria, pero esta vez a la velocidad máxima del sistema (90MHz). Cada estado de la memoria está relacionado con el solapado deducido en el apartado anterior. Una vez leído los datos el sistema permanece a la espera hasta que se haya alcanzado el siguiente estado, para así poder ser leído el siguiente fragmento de audio. En cuanto a la lectura, ésta también se realiza de forma circular, pero en este caso el flujo de datos no es constante. Los datos leídos pueden ser directamente almacenados en la memoria RAM de salida o ser multiplicados por un enventanado del tipo Hamming según la posición del interruptor “SW1”. Como se puede apreciar en la Figura 40 la memoria RAM de salida se ve reducida a la mitad siendo ésta de 512x16. Por último, la señal “o\_Load” indica al módulo FFT que los datos almacenados ya están preparados para su lectura.

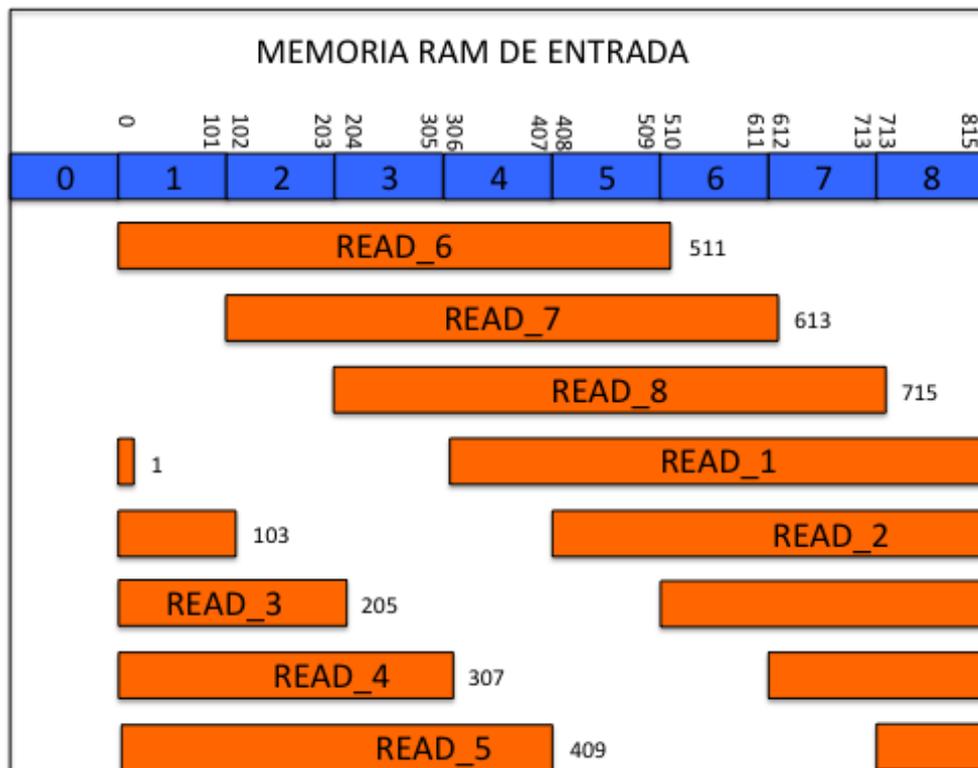


Figura 41. Buffer circular de almacenamiento y lectura de la memoria RAM de entrada.

### Bloque Controladora FFT

El bloque Controlador de FFT se encarga de leer la señal del módulo anterior cuando se detecta un estado alto de la señal “o\_Load”. Acto seguido los datos son introducidos al módulo FFT (*bloque suministrado por Altera*). A los resultados obtenidos (*parte real e imaginaria*) se le ha aplicado la operación matemática módulo (*algoritmo simplificado*) almacenando los resultados en una memoria RAM de 256 posiciones de

profundidad y 16 bits de palabra. Una vez que la memoria está completa se activa la señal “o\_Done” indicando que ya puede ser leída la memoria por el sistema microprocesador NIOS II.

En la Figura 42 se muestra el diagrama de bloques implementado en hardware para este bloque.

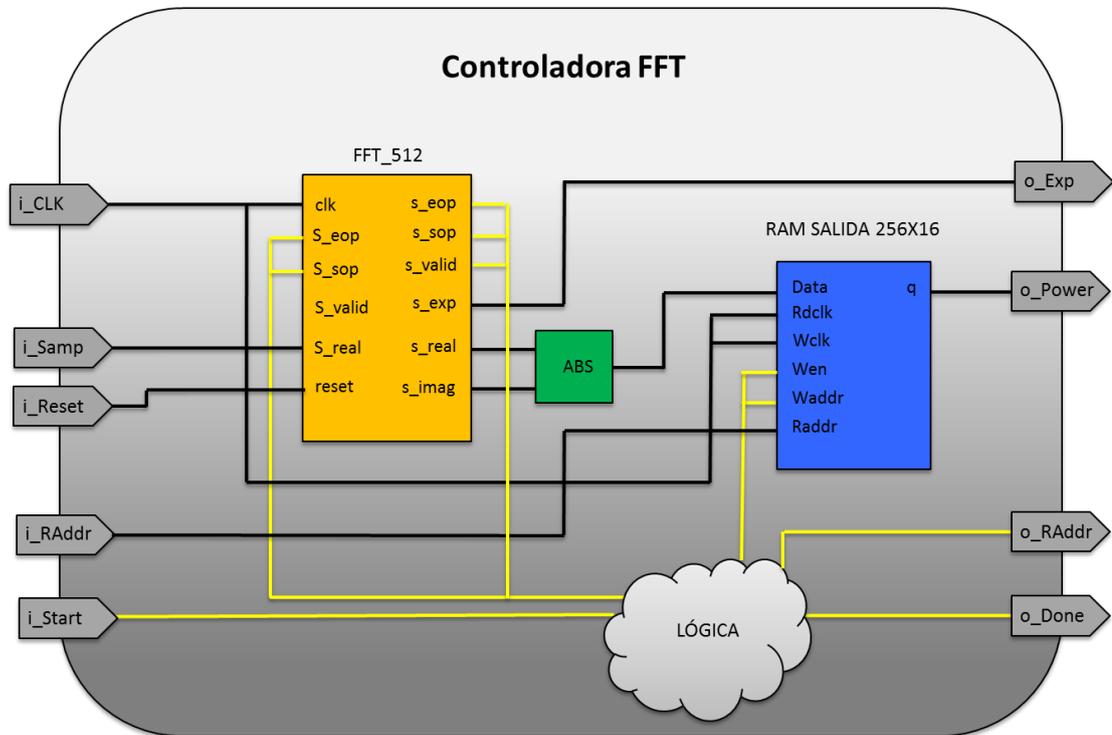


Figura 42. Bloque Controladora FFT.

### Bloque NIOS II

Además del procesado de señales hardware el sistema incluye un microprocesador de 32 bits NIOS II embebido en la propia FPGA. La función principal de este sistema está destinada al almacenamiento de los resultados hardware, y su posterior procesado software. Por otro lado también se encarga del interface hombre máquina mostrando los resultados mediante señales lumínicas en los LEDs así como en la pantalla LCD. Este microprocesador trabaja a la frecuencia de reloj del sistema, 90MHz, siendo completamente síncrono con la parte de procesado de señal hardware de alta velocidad. En la Figura 43 se pueden apreciar las distintas entradas y salidas de las que está compuesto este bloque.

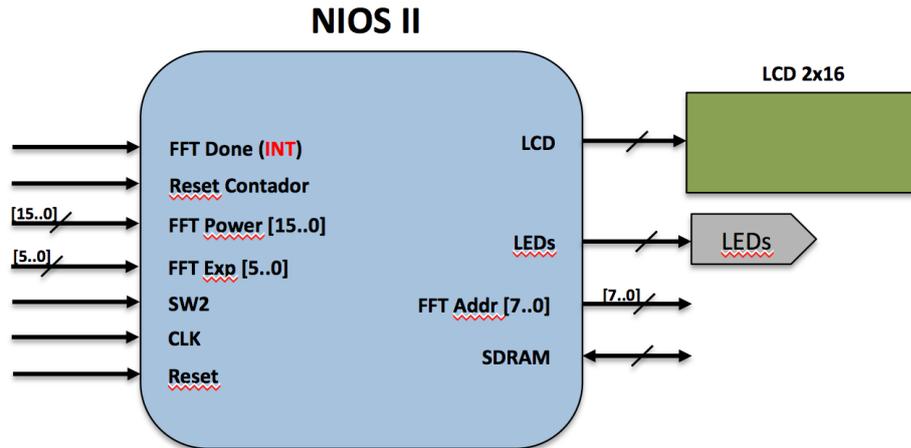


Figura 43. Esquema de la distribución de entradas y salidas de NIOS II

### 4.3 Implementación Software

El software se encarga de almacenar un total de 78 FFTs. Para ello el sistema permanece a la espera hasta detectar un estímulo de la señal FFT Done, que produce una interrupción. Esta interrupción activa un flag permitiendo la lectura de los 256 datos almacenados en RAM del anterior bloque accediendo a ésta a través del bus de direcciones "FFT Addr". De esta forma este resultado es almacenado en una columna de una matriz en la memoria SDRAM. Tras 78 interrupciones, con sus respectivas lecturas, la matriz queda completada. Para su comprobación se ha comparado esta matriz con la obtenida con Matlab, pudiendo destacar una gran similitud si no se tiene en cuenta el offset. Esta discrepancia se debe a que el resultado obtenido en Matlab se ha expresado con variables de tipo real, en cambio en la FPGA los valores son enteros y positivos. A nivel práctico tan solo ha sido necesario adaptar el nivel del umbral a la nueva escala obtenida. Una vez completada la matriz, se ha realizado un procesado de la señal, empleando el mismo algoritmo que se ha empleado en el apartado de procesado de señal con Matlab.

En la Figura 44 se muestra un ejemplo de procesado de la señal mediante Matlab con los datos extraídos de la FPGA.

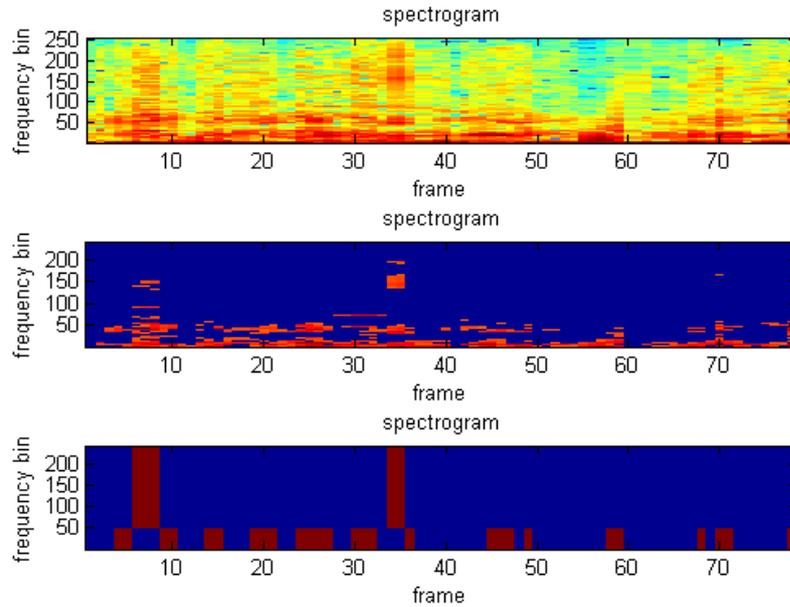


Figura 44. Ejemplo de procesado con los resultados obtenidos mediante la FPGA

Los resultados del procesado software son mostrados por consola devolviendo un “1” cuando se presenta actividad del sonido del movimiento abdominal de la pupa y “0” en caso contrario. También se pueden ver reflejado en la pantalla LCD, quedando los resultados positivos almacenados en una variable incremental. Este contador se puede resetear mediante el interruptor “SW2”.

En la verificación del sistema además de comprobar el buen funcionamiento del procesado de la señal, se ha comprobado que cumple uno de los requisitos principales, el funcionamiento de procesado en tiempo real. Para este estudio se ha utilizado la herramienta “Performance Counter Report” que incorpora QSYS. Los resultados se muestran en la Tabla 5.

```
--Performance Counter Report--
Total Time: 5.29016 seconds (476114356 clock-cycles)
+-----+-----+-----+-----+-----+
| Section      | %   | Time (sec)| Time (clocks)|Occurrences|
+-----+-----+-----+-----+-----+
|STFT          | 98  | 5.18248  | 466423563   | 1         |
+-----+-----+-----+-----+-----+
|Procesado     | 2.01| 0.10644  | 9579798     | 1         |
+-----+-----+-----+-----+-----+
|Resultados    |0.023| 0.00123  | 110958      | 1         |
+-----+-----+-----+-----+-----+
```

Tabla 5. Resultados de la temporización del procesado de un espectrograma con la herramienta “Performance Counter Report”

Del análisis de estos resultados se puede extraer que el proceso que más peso tiene es la STFT. A priori es normal que este proceso sea el que más peso tiene, ya que el algoritmo espera a que se almacenen las 78 FFTs procedentes de los resultados hardware

anteriormente descritos. Para comprobar si el tiempo empleado es correcto se ha realizado un estudio temporal.

$$F_s = 16 \text{ KHz} \rightarrow 16 \text{ kilo muestras por segundo}$$

$$\text{Solapado} = 102 \text{ muestras}$$

$$\text{Total muestras} = 102 \cdot 77 + 512 = 8366 \text{ muestras}$$

$$\text{Tiempo utilizado} = \frac{8366}{16000} = 0.5228 \text{ s}$$

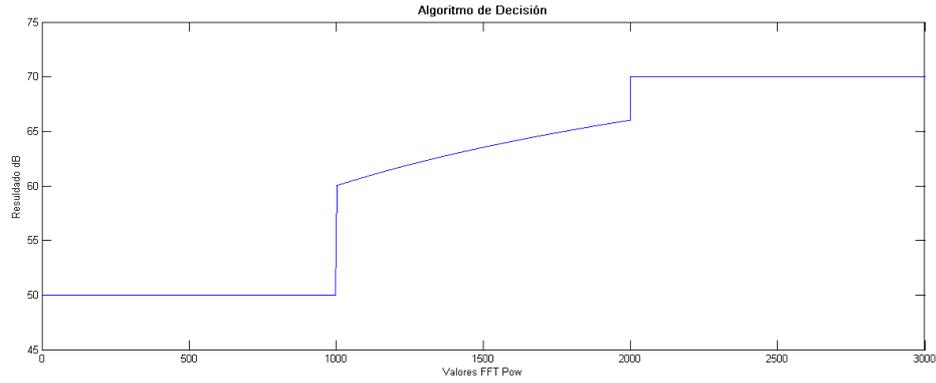
De los cálculos anteriores se puede determinar que la ventana temporal necesita aproximadamente 0.5s, quedando muy distante los 5.18s obtenidos con la herramienta de cronometraje.

Dentro del proceso de la STFT se ha realizado un estudio detallado del subproceso de almacenado, comprobado que la operación que se muestra en la ecuación (7) tiene un alto coste computacional. Además se ha de añadir que ésta se repite 19.968 veces.

$$Res = 20 \cdot \log_{10}(FFTPow) \quad (7)$$

Para solucionar este problema se han realizado diferentes pruebas. En primer lugar se implementó un bloque hardware dedicado. Sin duda una buena alternativa, pero también debe tenerse en cuenta de que esta solución introduce retardos, lo que implica un rediseño completo del hardware. Esta solución se ha descartado a favor de la sencillez y velocidad del uso de tablas. Para completar esta tabla tan solo se han de tomar los valores resultado de las FFTs, que en este caso son enteros y positivos. Además como se realiza una segmentación de los datos a partir de un determinado valor, tan solo ha sido necesario realizar la operación (7) en las proximidades del umbral, para que se permita un margen de desplazamiento para ajustar el algoritmo.

Como se muestra en la Figura 45 el algoritmo final evalúa tres zonas. La primera zona en donde se toman el valor de 50dB. La segunda, en las proximidades del umbral, en donde se extraen los resultados pre-calculados con la ecuación (7) que quedan almacenados en la memoria de la FPGA. Y la última, la zona alta donde se fijan los valores a 70dB.



**Figura 45. Representación gráfica del comportamiento del algoritmo de evaluación de los valores de entrada**

A continuación, en la Tabla 6, generada por la herramienta “Performance Counter Report” se aprecia una sustancial mejora aportada al emplear esta técnica. Los tiempos obtenidos en la sección STFT se aproxima en gran medida a la estimación de cálculos realizada. Pese a dicha mejora, todavía se puede apreciar que la parte de procesado ocupa un 17.3% teniendo unas pérdidas de 0.11s.

```
--Performance Counter Report--
Total Time: 0.6527 seconds (58741876 clock-cycles)
+-----+-----+-----+-----+-----+
| Section      | %   | Time (sec)| Time (clocks)|Occurrences|
+-----+-----+-----+-----+-----+
| STFT         | 82.5| 0.53828  | 48445200    | 1         |
+-----+-----+-----+-----+-----+
| Procesado    | 17.3| 0.11318  | 10185986    | 1         |
+-----+-----+-----+-----+-----+
| Resultados   | 0.19| 0.00123  | 110690      | 1         |
+-----+-----+-----+-----+-----+
```

**Tabla 6. Resultados de la temporización del procesado de un espectrograma con la herramienta “Performance Counter Report” mejorado**

Para lograr aislar el tiempo de procesado del programa principal del almacenamiento de la STFT, se ha incorporado dicho almacenamiento dentro de la interrupción, dejando la operación de procesado en segundo plano. Realizando esta simple operación el sistema solo procesa en los tiempos de espera entre FFTs.

A continuación se ha realizado un estudio que demuestra que esta operación se puede realizar garantizando un procesado de señal en tiempo real.

Tomando como referencia la frecuencia del reloj de sistema de 90MHz con un periodo de 11.11ns se puede realizar una estimación del tiempo de almacenado empleando la ecuación (8).

$$t_{Almacenado} = T_{Sys} \cdot N^{\circ}Ciclos/instrucción \cdot Datos \cdot N^{\circ}FFTs \quad (8)$$

$$t_{Almacenado} = 11.11 \text{ ns} \cdot 4 \cdot 256 \cdot 78 \approx 887.38 \mu\text{s}$$

Si a este tiempo se añade el resto del tiempo empleado en el procesado y visualización de resultados se obtiene el tiempo parcial:

$$t_{parcial} = 0.887ms + 113.18ms + 1.23ms = 0.115 ms$$

Como ya se puede comprobar en la Tabla 6 y también con los cálculos realizados anteriormente, el tiempo en que se dan lugar las 78 FFTs es aproximadamente de 522.8 ms por lo que el tiempo parcial empleado respecto al total supone tan solo un 22.05%, siendo más que suficiente para realizar dichas tareas en tiempo real.

En la Figura 46 y Figura 47 se ha reflejado el diagrama de flujo del programa principal y la rutina de interrupción respectivamente.

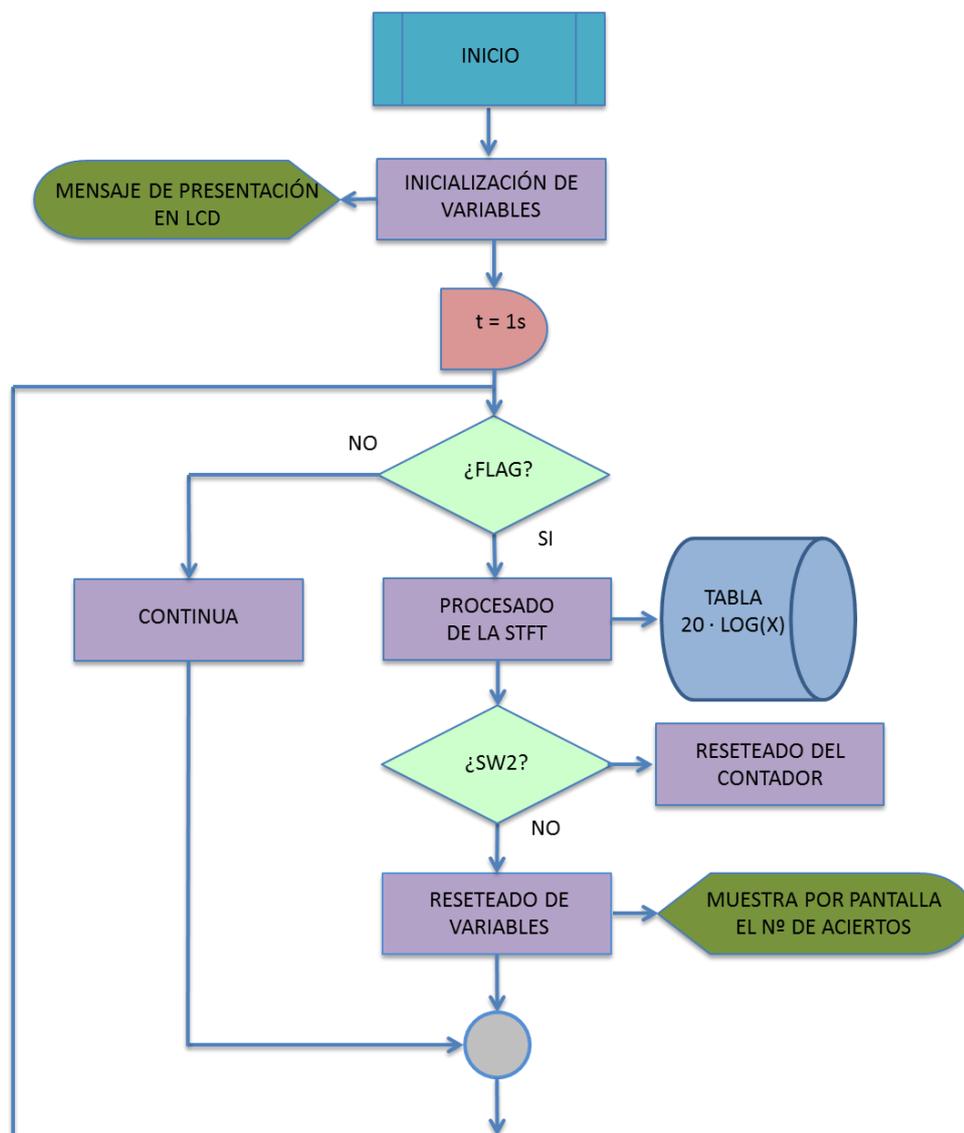


Figura 46. Diagrama de flujo del programa principal de procesado software

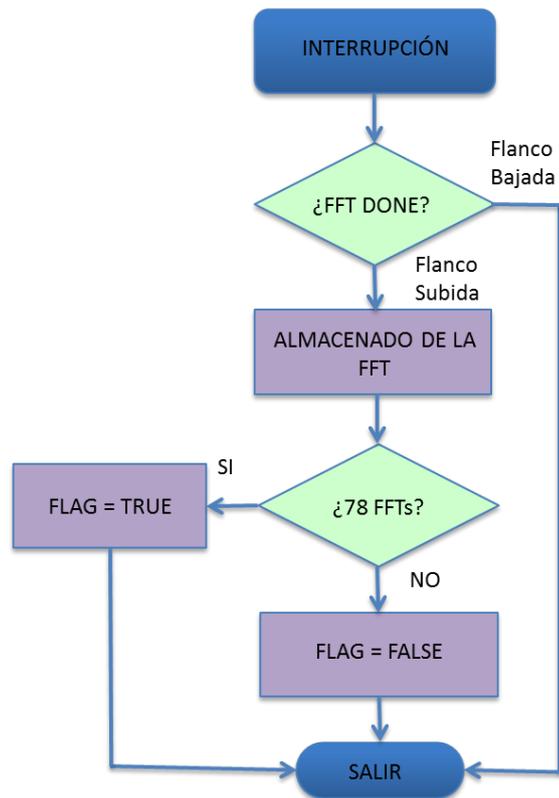


Figura 47. Diagrama de flujo de la rutina de interrupción



De la lectura de la matriz de confusión de procesamiento mediante Matlab (véase Figura 48), analizando las filas se puede extraer que el algoritmo es capaz de distinguir si existe o no presencia del movimiento abdominal. Para el caso de la presencia del movimiento se obtiene un 87.5% mientras que para su ausencia de un 82.2%. Analizando las columnas se puede apreciar la capacidad de predicción, que en este caso ha sido de un 72.9% para la presencia del movimiento abdominal y de un 92.3% para su ausencia.

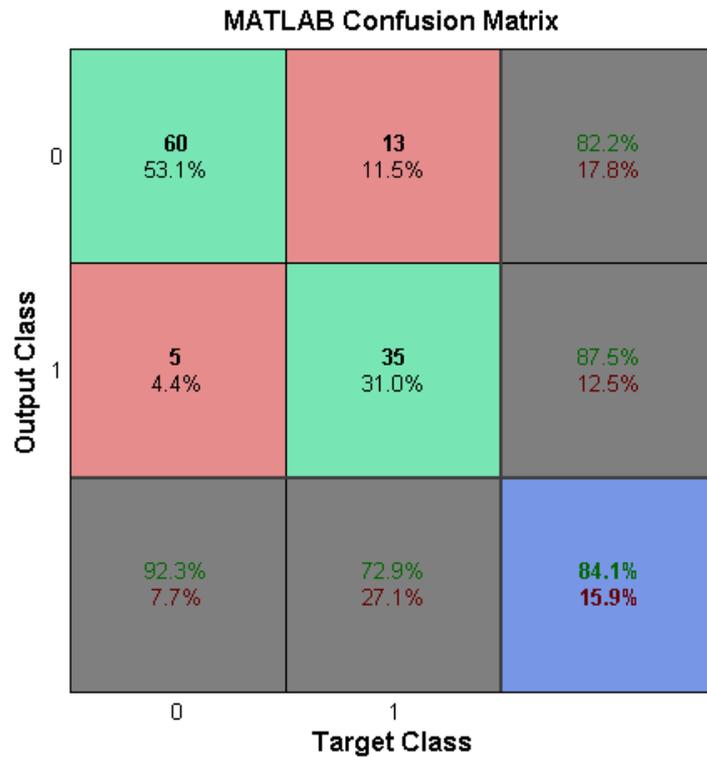


Figura 48. Matriz de confusión con los datos del procesamiento en Matlab

Por otro lado se han analizado los resultados obtenidos tras el procesamiento mediante la FPGA (véase Figura 49). De su análisis por filas se ha extraído una clara diferenciación de clases, siendo de un 88.9% para la presencia de el movimiento abdominal y de un 79.2% para su ausencia. En cuanto a la predicción se ha obtenido un 66.7% para la presencia del movimiento y un 93.8% para su ausencia.

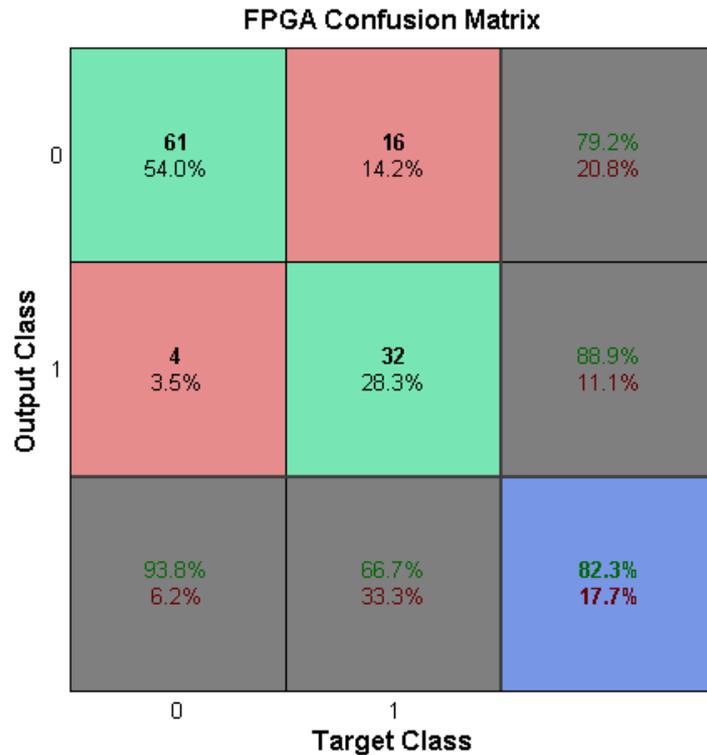


Figura 49. Matriz de confusión con los datos del procesado con la FPGA

Los resultados reflejan un comportamiento similar para los dos procesos, obteniendo un 84.1% de aciertos en Matlab frente a un 82.3% en la PFGA. Por otro lado también se puede apreciar en ambas matrices que los resultados más desfavorables se han obtenido en la predicción de acierto, es decir, aquellos que devuelven un “0” cuando en realidad existe actividad. Este comportamiento a priori puede parecer negativo pero se ha de destacar que en los audios analizados con presencia de sonidos propios del movimiento abdominal acaban teniendo un elevado número de aciertos. Esto es debido a que el movimiento ondulatorio de la pupa se repite continuamente durante un largo periodo de tiempo.

A continuación en la

Tabla 9 se muestran los resultados obtenidos empleando la técnica de procesado descrita en los diagramas de flujo de las Figuras 46 y 47. Como se puede apreciar en el informe del “*Performance Counter Report*” se ha eliminado el proceso de almacenado de la STFT pasándose a efectuar en la rutina de interrupción. De este modo el programa principal tan sólo se ocupa de realizar la tarea de procesado de señal, obteniéndose así un tiempo de procesado muy próximo al estimado (0.5228s).

```
Dentro 0 --Performance Counter Report--
Total Time: 0.534996 seconds (48149623 clock-cycles)
```

Section	%	Time (sec)	Time (clocks)	Occurrences
Procesado	100	0.53500	48149590	1
Resultados	100	0.53500	48149590	1

**Tabla 8. Resultados de la temporización del procesado de un espectrograma con la herramienta “Performance Counter Report”**

Otro de los aspectos a tener en cuenta es la implementación en una placa de bajo coste. Para cubrir esta coste. Para cubrir esta necesidad se ha pensado en la placa de desarrollo DE0-Nano. Como se muestra en la se muestra en la

Tabla 99 este diseño puede ser implementado sin ningún problema. Tan solo debería realizarse el diseño hardware del convertor AD para la captura de audio e incorporar un display LCD.

	Utilizados	DE0-Nano	
<b>Elementos lógicos</b>	17.950	22.320	✓
<b>Memoria embebida</b>	225.081	594.000	✓
<b>Multiplicadores embebidos</b>	37	66	✓
<b>PLLs</b>	3	4	✓

**Tabla 9. Tabla comparativa de los recursos disponibles utilizados frente a los disponibles en la placa de desarrollo DE0-Nano**

## 6. Conclusiones

Esta tesis recoge el estudio de investigación acerca del desarrollo de un equipo para la detección e identificación del RPW en las palmeras.

En primer lugar se ha realizado un estudio del estado del arte recabando información de otros artículos relacionados con la investigación e identificación de este coleóptero. También se han introducido los métodos y herramientas empleadas en las distintas pruebas de campo. Para tener un mayor conocimiento del RPW se ha documentado su ciclo biológico dando a conocer la evolución de la plaga. Una vez expuesta toda la información se ha realizado una justificación acerca del tipo de sonido producido por el RPW que se quiere detectar.

Una vez determinado el patrón de audio el estudio se ha centrado en los análisis de audios empleando algoritmos de procesado de señales. Estos algoritmos de procesado se han realizado de forma offline buscando la mayor sencillez de implementación posible. Por esta razón el estudio se ha centrado en primer lugar en un análisis de la señal de audio en el dominio temporal. Tras los resultados obtenidos se ha descartado el dominio temporal realizando un estudio en el dominio de la frecuencia, mostrando características interesantes en los patrones analizados. Por último se han combinado ambas técnicas (*dominio tiempo-frecuencia*), obteniendo así los mejores resultados.

Una vez propuesto un algoritmo de procesado, se ha empleado un soporte hardware para su implementación, que en esta caso ha sido en la placa DE2-115 de Altera. Por un lado se han aprovechado las características hardware de las que dispone la placa y por otro se ha utilizado el potencial de la FPGA Cyclone IV para realizar un diseño de arquitectura hardware que permita el procesado de la señal en tiempo real. Además se han explotado las posibilidades que ofrece el fabricante Altera incorporando un microprocesador NIOS II embebido de 32 bits para el procesamiento software.

Por último los resultados obtenidos mediante el procesado por computador y por la placa de desarrollo han sido comparados verificando éstos con los obtenidos de forma manual. De este análisis comparativo se ha realizado un análisis estadístico obteniendo un 84.1% de efectividad con el procesado con Matlab y un 82.3% con la FPGA.

## 7. Líneas futuras

El equipo diseñado en esta tesis se encuentra en fase de investigación, y requiere de medidas en campo para comprobar por un lado su correcto funcionamiento y por otro verificar las tasas de éxito obtenidas mediante los dos procesados.

Una vez superada la fase de pruebas en campo, el objetivo es el de minimizar sus componentes externos para poder realizar un equipo portátil y de uso sencillo. El desarrollo de este equipo debe cumplir los siguientes requisitos:

- Ser un equipo portátil con alimentación a baterías ya sea recargables o sustituibles.
- Disponer de una autonomía que permita su uso ininterrumpido durante días o incluso semanas.
- Tener un grado de protección IP que permita su correcto funcionamiento en la intemperie.
- Ser de unas dimensiones con un volumen preferiblemente inferior a 5 cm<sup>3</sup>

Por otro lado se plantea la posibilidad de implementar un algoritmo de procesado más versátil con la capacidad de aprender. De esta forma el mismo equipo podría diferenciar entre distintos audios permitiendo un equipo polivalente para la identificación de varias plagas. En este campo las redes neuronales están cobrando un papel muy importante, con muy buenos resultados en el reconocimiento de voz.

## 8. Bibliografía

[Albiol et al. 1999]

*Tratamiento digital de la señal. Teoría y Aplicaciones*

A. Albiol, V. Naranjo, J. Prades

I.S.B.N.:84-7721-842-0, Universidad Politécnica de Valencia (1999)

[Al-Manie et al. 2007]

*Acoustic Detection of the Red Date Palm Weevil*

M. A. Al-Manie, M. I. Alkanhal

*World Academy of Science, Engineering and Technology 2 (2007)*

[Cabello 2006]

*Population biology and dynamic of the red palm weevil, Rhynchophorus ferrugineus (Olivier,1790)*

T. Cabello

*Jornada internacional sobre el Picudo Rojo de las Palmeras, Moncada, Spain. (2006)*

[E.P.P.O. 2005]

*European and Mediterranean Plant Protection Organization, Rhynchophorus palmarum,*

*OEPP/EPPO Bulletin 35, 468–471, (2005)*

[El-Sufty et al.2012]

*Biological Control of Red Palm Weevi, Rhynchophorus Ferrugineus(Col.:Curculionidae) by the Entomopatogenic Fungus Beauveria Bassiana in United Arab Emirates*

R. El-Sufty, SA Al-Awash, A. M. Al Amiri, AS Shahdad, A. H. Al Bathra, SA Musa

*ISHS Acta Horticulturae 736: III Conferencia Internacional de la palma datilera (2012)*

[Escríche et al. 2012]

*A potentiometric electronic tongue for the discrimination of honey according to the botanical origin. Comparison with traditional methodologies: Physicochemical parameters and volatile profile*

I. Escríche, M. Kadar, E. Domenech and L. Gil-Sanchez

*Journal of Food Engineering 109 (2012) 449 -456*

[Gutiérrez et al. 2010]

*Development of a bioacoustic sensor for the early detection of Red Palm Weevil (Rhynchophorus ferrugineus Olivier)*

A. Gutiérrez , V. Ruiz , E. Moltó , G. Tapia , M. M. Téllez

*Crop Protection29(2010)671e676*

[Manachini et al. 2012]

*Exotic insect pests: The impact of the Red Palm Weevil on natural and cultural heritage in Palermo (Italy)*

B. Manachini\*, N. Billeci , F. Palla

*Dep. Environmental Biology and Biodiversity, University of Palermo, Via Archirafi 38, 90123 Palermo, Italy (2012)*

## MASTER EN SISTEMAS ELECTRÓNICOS

[Mankin et al. 1997]

*Acoustic counting of adult insects with different rates and intensities of sound production in stored wheat*

R. W. Mankin, D. Shuman, J.A. Coffelt  
*J. Econ. Entomol.* 90,1032-1038 (1997)

[Mankin 2000]

*Eavesdropping on Insects Hidden in Soil and Interior Structures of Plants*

R. W. Mankin, J. Brandhorst-Hubbard, K. L. Flanders, M. Zhang, R. L. Crocker, S. L. Lapointe, C. W. McCoy, J. R. Fisher, and D. K. Weaver

[Mankin et al. 2000; Zhang et al. 2003], y de insectos en tallos de plantas [Mankin et al. 2004]  
*J. Econ. Entomol.* 93(4):1173-1182 (2000)

[Mankin 2004]

*Acoustic System for Insect Detection in Plant Stems: Comparisons of *Cephus cinctus* in Wheat and *Metamasius callizona* in *Brornelia**

R. W. Mankin, D. K. Weaver, M. Grieshop, B. Larson, and W. L. Morrill  
*J. Agric. Urban Entomol.* 21(4): 239-248 (October 2004)

[Mankin. 2010]

*Recent Developments in the use of Acoustic Sensors and Signal Processing Tools to Target Early Infestations of Red Palm Weevil in Agricultural Environments.*

R. W. Mankin,  
*Florida Entomologist*, 94(4):761-765. (2011)

[Martínez-Máñez et al. 2004]

*An “electronic tongue” design for the qualitative analysis of natural waters*

R. Martínez-Máñez, J. Soto, E. Garcia-Breijo, L. Gil, J. Ibáñez and E. Llobet  
*Sensors and Actuators B* 104 (2005) 302–307

[Massa et al. 2011]

*Microwave treatment for pest control: the case of *Rhynchophorus ferrugineus* in *Phoenix canariensis**

R. Massa, E. Caprio, M. De Santis, R. Griffò, M. D. Migliore, G. Panariello, D. Pinchera and P. Spigno  
*Bulletin OEPP/EPPO Bulletin* 41, 128–135 (2011)

[Potamitis et al. 2009]

*On Automatic Bioacoustic Detection of Pests: The Cases of *Rhynchophorus ferrugineus* and *Sitophilus oryzae**

I. Potamitis, T. Ganchev, D. Kontodimas,  
*J. Econ. Entomol.* 102(4): 1681D1690 (2009)

[Sheffrahn et al. 1993]

*Evaluation of a novel, hand-held, acoustic emissions detector to monitor termites*

R.H. Sheffrahn, W.P. Robbins, P. Busey, N.Y Su, R.K. Muller  
*J. Econ. Entomol.* 86,1720-1729 (1993)

## MASTER EN SISTEMAS ELECTRÓNICOS

[Schutte et al. 2007]

*Speech Recognition With Localized Time-Frequency Pattern Detectors*

*K. Schutte, J. Glass*

*MIT Computer Science and Artificial Intelligence Laboratory*

*978-1-4244-1746-9/07/\$25.00 ©2007 IEEE*

[Zhang et al. 2003]

*Acoustic Identification and Measurement of Activity Patterns of White Grubs in Soil*

*M. Zhang, L. R. Coker, R. W. Mankin, L. K. Flanders, and L. J. Brandhorst-Hubbard*

*J. Econ. Entomol. 96(6): 1704D1710 (2003)*