



---

---

## Instituto Politécnico Nacional

Escuela Superior de Ingeniería Mecánica y Eléctrica

Unidad Culhuacan

Sección de Estudios de Posgrado e Investigación

### **“Implementación en un FPGA de un Método para la Inserción de una Marca de Agua en Archivos Musicales mediante de Ocultamiento por Eco”**

## **TESIS**

Para obtener el grado de:

**Maestro en Ciencias de Ingeniería en Microelectrónica**

Presenta:

**Juan Antonio Ríos Chávez**

Directores:

**Dr. Enrique Escamilla Hernández**

**Dr. Daniel Cruz Pérez**





*INSTITUTO POLITÉCNICO NACIONAL*  
*SECRETARÍA DE INVESTIGACIÓN Y POSGRADO*

*CARTA CESIÓN DE DERECHOS*

En la Ciudad de México el día 8 del mes Diciembre del año 2011, el que suscribe Ing. Juan Antonio Ríos Chávez alumno (a) del Programa de Maestría en Ciencias de Ingeniería en Microelectrónica con número de registro A100539, adscrito a SEPI-ESIME-CULHUACAN, manifiesta que es autor (a) intelectual del presente trabajo de Tesis bajo la dirección de Dr. Enrique Escamilla Hernández y Dr. Daniel Cruz Pérez y cede los derechos del trabajo intitulado Implementación en un FPGA de un Método para la Inserción de una Marca de Agua en Archivos Musicales mediante de Ocultamiento por Eco, al Instituto Politécnico Nacional para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben reproducir el contenido textual, gráficas o datos del trabajo sin el permiso expreso del autor y/o director del trabajo. Este puede ser obtenido escribiendo a la siguiente dirección [jriosc0500@ipn.mx](mailto:jriosc0500@ipn.mx). Si el permiso se otorga, el usuario deberá dar el agradecimiento correspondiente y citar la fuente del mismo.

Ing. Juan Antonio Ríos Chávez  
Nombre y firma



# INSTITUTO POLITÉCNICO NACIONAL SECRETARÍA DE INVESTIGACIÓN Y POSGRADO

## ACTA DE REVISIÓN DE TESIS

En la Ciudad de México D. F. siendo las 17:00 horas del día 5 del mes de diciembre del 2011 se reunieron los miembros de la Comisión Revisora de la Tesis, designada por el Colegio de Profesores de Estudios de Posgrado e Investigación de SEPI-ESIME-CULH. para examinar la tesis titulada:

"Implementación en un FPGA de un Método para la inserción de una Marca de Agua en Archivos Musicales mediante Ocultamiento por Eco"

Presentada por el alumno:

Ríos	Chávez	Juan Antonio						
Apellido paterno	Apellido materno	Nombre(s)						
	Con registro:	A	1	0	0	5	3	9

aspirante de:

MAESTRÍA EN CIENCIAS DE INGENIERÍA EN MICROELECTRÓNICA

Después de intercambiar opiniones los miembros de la Comisión manifestaron **APROBAR LA TESIS**, en virtud de que satisface los requisitos señalados por las disposiciones reglamentarias vigentes.

### LA COMISIÓN REVISORA

Directores de tesis

Dr. Enrique Escamilla Hernández	Dr. Daniel Cruz Pérez

Dra. Mariko Nakano Miyatake	Dr. Juan Carlos Sánchez García

Dr. Mario González Lee



PRESIDENTE DEL COLEGIO DE PROFESORES

Dr. Gonzalo Isaac Duchén Sánchez

## **Resumen**

El mercado de agua digital de audio, tiene una importancia significativa debido a la protección de derechos de autor, objetivo para el cual, esencialmente, fueron diseñadas. En esta investigación se desarrolla e implementa un algoritmo de inserción de marcas de agua en archivos musicales, que cumpla requerimientos de robustez y fragilidad para que sea utilizado como autenticación de archivos multimedia y servir como prueba ante tribunales y determinar cuando la marca de agua haya sido modificada o alterada.

## **Abstract**

The digital audio watermarking has a significant importance due to copyright protection, purpose for which, essentially, they were designed. This research develops and implements an algorithm to embed watermarks into music files which meets requirements of robustness and fragility to be used as authentication of media and serve as evidence in court and determine when the watermark has been modified or altered.

## **Agradecimientos**

Quiero expresar mi agradecimiento más sincero a mis directores de tesis, al Dr. Enrique Escamilla Hernández y al Dr. Daniel Cruz Pérez por su paciencia, apoyo incondicional y ayuda en mi realización como maestrante profesional, por orientarme y aconsejarme acertadamente para la culminación de esta investigación.

También agradezco la atención proporcionada por los profesores de la SEPI de ESIME Culhuacan por la ayuda que en algún momento de mi estancia solicité y fue atendida inmediatamente.

Al CONACyT por su apoyo económico para realizar mis estudios de manera íntegra.

Al programa PIFI por las experiencias y apoyos recibidos.

A mi familia por su incondicional apoyo y por apoyar las decisiones que yo he tomado.

Y por último pero no menos importante, agradezco a Dios por permitirme vivir esta maravillosa experiencia.

Juan Antonio Ríos Chávez

## Planteamiento del Problema

En la actualidad, existe una investigación amplia en el campo del marcado de agua digital de archivos multimedia, pero relativamente es menor la investigación que se ha realizado exclusivamente para audio debido a la complejidad que implica el estudiar el sistema auditivo humano con la finalidad de buscar la inaudibilidad de los datos ocultos.

Las marcas de agua digitales en audio han sido propuestas idealmente para resolver el problema del copyright y/o la protección de derechos de autor, pero en esta investigación se propone diseñar un algoritmo de inserción que tenga un determinado grado de fragilidad para poder utilizar la marca de agua como un verificador de integridad y así poder utilizarla en testimoniales como prueba ante un tribunal pretendiendo que la grabación de un testimonio sea utilizada como prueba a la que se le incrusta la marca de agua para asegurarse que dicha grabación no sea editada, modificada o alterada por alguna de las partes involucradas según les convenga.

Por este motivo se realizó una investigación del marcado de agua digital en audio y se eligió trabajar con el algoritmo de inserción de marcas de agua por ocultamiento de eco. La elección de este algoritmo radica en su fácil implementación en software y en hardware ya que consiste en añadir un retraso de la señal original e incrustar la marca de agua en ello, en el retraso se inserta un "1" y cuando no existe tal retraso se inserta un "0". Para el aumento de la robustez se manejan diferentes modificaciones, algunas son, modificar la amplitud de la señal retrasada con la finalidad de que se "combine" con la señal original para aumentar la imperceptibilidad y no notar la diferencia de la original y la marcada, además se pretende insertar múltiples ecos para así generar una difícil distorsión y/o extracción intencional o no intencional de la marca de agua. La implementación en el FPGA permitirá que el algoritmo funcione en tiempo real, es decir, que en el momento que el usuario introduce el audio que desea marcar, éste se someta al proceso de inserción en el cual se le incrusta la marca de agua y posteriormente a la salida se obtenga el mismo audio pero marcado. Un FPGA, debido a su arquitectura, hace posible la implementación del algoritmo de inserción de marcas de agua propuesto ya que la capacidad y recursos disponibles en este hardware pueden utilizarse para más de una aplicación, esto hace un uso óptimo de memoria en el dispositivo, ya que solo se ocuparía lo necesario para el sistema y lo restante podría utilizarse en otra aplicación, por último, evitamos usar más recursos y depender de la computadora y de algún software de desarrollo.



agua. Para una marca de agua digital de audio, se numeran las siguientes características deseables [1]:

- ✓ **Inaudible**
- ✓ **Seguridad**
- ✓ **Fragilidad**
- ✓ **Robustez**
- ✓ **Constante de tasa de bit**
- ✓ **Transparencia Perceptual**
- ✓ **Fidelidad**

Ahora que conocemos los conceptos y requerimientos utilizados en el mercado de agua digital en audio y los parámetros necesarios para incrementar la eficiencia del mismo.

Algunos autores que específicamente trabajan con el algoritmo de ocultamiento por eco, a continuación se citan algunos de los artículos leídos y la descripción teórica de cada uno de ellos:

**Foo Say Wei, Dong Qi** proponen un sistema de marcado de agua digital mediante ocultamiento por eco en señales estéreo no ciego, las señales se dividen en segmentos y la marca de agua se incorpora al ocultar los ecos en estos segmentos. Los ecos integrados representan los bits de marca de agua.

Para marcas de agua de un solo canal, se utilizan ecos con dos diferentes retrasos. Uno de ellos es utilizado para codificar bits de marca de agua "1" y el otro se utiliza para codificar bits de marca de agua "0". Proponen un esquema de marcado, el cual consiste en que el eco está integrado sólo cuando el nivel de energía del tramo previsto sea superior a un determinado umbral. Se hace eco con un valor de retardo se utiliza para representar los bits. De aquí tomo la manera de generar un eco de la señal original, agregando un retraso de la misma y el sistema de detección que utilizan para extraer la marca de agua incrustada.

**Iynkaran Natgunanathan, Yong Xiang** proponen un sistema de marcado de agua digital en señales de audio estéreo basado en el método de extensión de tiempo en ocultamiento por eco con la finalidad de superar la baja robustez de los sistemas tradicionales de este tipo. En la inserción, las señales de eco con polaridades opuestas son adheridas a ambos canales de la señal original. Esto mejora la imperceptibilidad del esquema de marcado de agua, desde que se agrega la marca de agua que tienen similares efectos en ambos canales. Luego se desarrolla la parte del decodificador, con el objetivo de aumentar la robustez de la marca de agua frente a ataques comunes. El proceso de inserción y el proceso

de detección utilizan las ventajas de ambos canales de las señales estéreo, esto significativamente reduce la interferencia de la señal original en la extracción de la marca de agua que es la principal razón para errores de detección en los métodos tradicionales de extensión de tiempo. De esta propuesta obtengo la forma de insertar los bits de marca de agua de manera que sean imperceptibles al oído humano, pero al mismo optimizando el proceso de inserción para que sean robustos a los ataques comunes aplicados a sistemas de marcado de agua digital en audio.

**Shine K. P., Krishna Kumar S.** mencionan que en el algoritmo de ocultamiento por eco tradicional se insertan retrasos de la señal con la finalidad de incrustar unos o ceros. La elección del núcleo de eco determina la fidelidad y la robustez de la marca de agua, por este motivo, proponen la extensión bipolar del núcleo de eco que presenta mejoras en la fidelidad del audio marcado sin afectar la detección de la marca de agua incrustada.

**Gui-jun Nian, Shu-xun Wang, Yun-lu Ge** tratan el tema de inserción de marcas de agua en audio desde el fenómeno de reverberación del mismo que es uno de los efectos más usados en audio. El método que proponen consiste en insertar reverberación a las grabaciones de audio con la finalidad de que el audio marcado suene natural como si fuera reproducido en una sala de conciertos real. LA marca de agua se obtiene convolucionando la señal de audio con la respuesta al impulso de la sala. Los parámetros para el cálculo de la respuesta al impulso de la sala se utilizan como la clave secreta que se debe utilizar en el proceso de detección. El proceso de detección utiliza una decodificación homomórfico que separa la función del núcleo del cepstrum de la señal de audio con marca de agua y luego tomar la correlación cruzada de la secuencia de retardo de impulso de la sala, elemento necesario para conseguir la marca de agua mediante la búsqueda del pico.

**S.A. Chou, S.F. Hsieh** proponen una extensión bilateral de tiempo en el núcleo de eco para mejorar la robustez frente a ataques maliciosos, simultáneamente, incrementando el valor del pico del núcleo cepstrum para optimizar la detección. Se basa en las propiedades de la secuencia PN y expansiones logarítmicas y binomiales

**J. A. Ríos Chávez, E. Escamilla Hernández, M. González Lee** proponen una técnica esteganográfica para detectar la existencia de mensajes secretos incrustados por el método de ocultamiento por eco. Se basaron en la regla de cambio de la ventana deslizante cepstrum, un coeficiente en el dominio cepstrum para realizar el sistema de detección por eco. Este artículo facilita el entendimiento para calcular el cepstrum y la manera de utilizarlo para el proceso de detección, en

el algoritmo propuesto una de sus fortalezas es la detección a ciegas que se utiliza mediante el cálculo del cepstrum y de la posterior detección de picos de eco que existen en el audio marcado y hacer una comparación para identificar cuando se insertó un cero y cuando se insertó un uno, todo esto sin la necesidad de contar con el archivo de audio original ni de conocer la marca de agua incrustada.

**Nedeljko Cvejic, Tapio Seppanen** Muestra un novedoso método para insertar una marca de agua basado en el método del espectro extendido en descompresión, se presentan secuencias de ruido de audio. El sistema toma eficientemente ventajas del fenómeno de enmascaramiento en HAS con el fin de integrar los datos de marca de agua por debajo del umbral de enmascaramiento de la señal de audio. La detección de la marca de agua es hecha mediante la detección a ciegas, es decir, que no se necesita la señal de audio original. Ninguna de las transformaciones y del dominio de la frecuencia son realizadas ya sea en la parte de inserción o detección del esquema propuesto. En pruebas experimentales, el sistema resultó ser robusto contra los ataques comunes y contra otros algoritmos de marcado de agua. La evaluación de la calidad subjetiva del algoritmo mostró que la marca de agua incrustada introduce una distorsión baja que es inaudible en la señal de audio resultante.

**Paraskevi Bassia, Ioannis Pitas** proponen un método de marcado de agua que ofrece la protección de derechos de autor para una señal de audio, procesado en el dominio del tiempo. Las modificaciones de la intensidad de la señal de audio son limitadas por la necesidad de producir una señal de salida que es perceptualmente idéntica a la señal original. El método es clasificado como detector a ciegas, ya que no requiere la señal original en el proceso de detección de la marca de agua. La señal de marca de agua es generada usando una llave, un número conocido solamente por el propietario de los derechos de autor. La inserción de la marca de agua depende de la amplitud de la señal de audio y de la frecuencia, en una forma de minimizar la audibilidad de la señal de marca de agua, además, es robusta para manipulaciones comunes de la señal de audio como la codificación MPEG, filtrado, remuestreo, recuantización, cortado de la señal. Sin embargo, el método no es robusto para ataques más sofisticados, por ejemplo, si la señal marcada ha sido sujeta a cambios en la escala de tiempo, no se podría detectar la marca de agua. Para la inserción de la marca de agua se asume una señal de audio de tamaño  $M$  con  $x(i)$  muestras, el cual se divide en  $N_s$  segmentos de  $N$  muestras cada uno, se denota por  $x_k(i)=x(kN+i)$  para  $i=0, N-1$  y  $k=0, \dots, N_s-1$  cada uno de esos segmentos es marcado con la misma secuencia bipolar  $w(i)$  que pertenece al conjunto  $\{-1,1\}$  y que es generado por un mapa caótico de umbral (para este paper se utilizó el Renyi Map). La señal de marca de agua es insertada en cada segmento de audio siguiendo tres pasos, en el primer

paso la señal de marca de agua  $w(i)$  es modulada por la señal de audio para producir la señal dependiente de la marca de agua  $w'_k(i) = \alpha x_k(i) \oplus w(i)$  donde  $\alpha$  es una constante que controla la amplitud de la señal de marca de agua. En el siguiente paso,  $w'_k(i)$  es filtrada usando un filtro pasabajas tipo Hamming de orden  $L$ .

$$w''_k(i) = \sum_{l=0}^{L-1} b_l w'_k(i-l)$$

Donde  $b_l$  son los coeficientes del filtro. Finalmente la señal dependiente de la señal de marca de agua filtrada, es incrustada en el segmento  $x_k$  de la señal original para producir el segmento de la señal marcada  $y_k = x_k(i) + w''_k(i)$

**Ingemar J. Cox, Matt L. Loom, Jeffrey A. Bloom** es un artículo que explica las aplicaciones a las que se someten las marcas de agua y las características que deben tener para ser óptimas. Explica cinco aplicaciones que consideran como las más importantes para el audio, tales como, seguimiento de difusión, que menciona que podemos utilizar marcas de agua para el seguimiento de difusión, poniendo una marca de agua única en cada clip de sonido anterior a la emisión. Estaciones de monitoreo automático puede recibir emisiones y buscar estas marcas de agua, la identificación de cuándo y dónde cada clip aparece, propietario de identificación, se puede usar la marca de agua para proveer marcas complementarias a los derechos de autor convirtiéndose en una parte integral del contenido, prueba de la propiedad, la marca de agua no solo puede ser usada para identificar la propiedad de derechos de autor sino también para probar realmente la propiedad, marcas de agua transaccionales, permiten al distribuidor identificar el origen de una copia ilegal y control de copias, . Además muestra las propiedades necesarias para tratar de hacer una marca de agua ideal, las cuales son, robustez, es una característica en la que la marca de agua sobrevive a operaciones de procesamiento digital, manipulación de la resistencia, se refiere a que un sistema de marcado de agua es resistente a ataques hostiles, Fidelidad, Característica que permite que la degradación que provoca la inserción de la marca de agua sea difícil de percibir por el oyente, costo computacional, falso positivo, es una detección de una marca de agua en el audio en el que actualmente no está insertada la marca de agua y falso negativo, es cuando en el proceso de detección se dice que no hay insertada una marca de agua cuando en realidad si lo hay.

**Hyoung Joong Kim** Este artículo examina los sistemas de marcado de agua en señales de audio. Describe principalmente cuatro esquemas de inserción de marcas de agua, el primero es el método de cuantización que consiste en cuantificar una muestra valuada en "X" asignándole un nuevo valor a esa muestra

basado en el valor de la muestra cuantificada, el segundo es el método del espectro extendido o difuso, que incrusta una secuencia pseudo-aleatoria y detecta la marca de agua calculando la correlación entre la secuencia de ruido pseudo-aleatorio y la marca de agua de la señal de audio, el tercer método es el de dos conjuntos que básicamente la idea consiste en hacer dos ajustes en la señal de audio original, para el proceso de detección si los dos ajustes son diferentes se puede decir que la marca de agua está presente, tales decisiones se han hecho a base de pruebas hipotéticas, el cuarto es el método de réplica en el cual la señal original de audio puede ser usada como marca de agua, un ejemplo es la generación de eco de la misma señal, se introduce a la señal original en el dominio del tiempo.

**C. A. Genaro Olivas** realiza una exhaustiva revisión y análisis de algoritmos existentes para marcado de agua digital en audio y la implementación de uno de ellos en un FPGA. La elección fue el algoritmo de Patchwork Modificado, esto dependió de factores como la imperceptibilidad y la robustez de la marca de agua de la factibilidad del algoritmo para implementarlo en el dispositivo.

### **Preguntas de problematización**

1. ¿Sería posible la implementación del algoritmo en FPGA con la finalidad de hacer el proceso en tiempo real?
2. Si la marca de agua aumenta su robustez, ¿Qué tipo de problemas se pueden presentar en el proceso de detección?
3. ¿Cómo podría hacerse más robusto el algoritmo de inserción de marcas de agua a través de ocultamiento por eco?

## **Objetivo General**

- ✓ Implementar en un FPGA un sistema de inserción de marcas de agua en archivos musicales basado en el ocultamiento por eco y así evitar depender de la PC y de software de diseño.

## **Objetivos Específicos**

- ✓ Analizar comparativamente las diferentes técnicas de marcado de agua digital en audio
- ✓ Diseñar un esquema mediante bloques de Xilinx para ejemplificar el algoritmo de inserción mediante ocultamiento por eco
- ✓ Implementar el algoritmo de inserción mediante ocultamiento por eco en un FPGA

## **Justificación**

En la actualidad existen diversos algoritmos de marcado de agua digital para archivos multimedia (Audio, imágenes, datos, video), pero relativamente son pocos los que dedican su estudio a archivos de audio debido a la complejidad que esto implica ya que previamente se debe hacer un estudio general del oído humano y sus características auditivas.

Para el audio, se han trabajado diferentes algoritmos de marcado, algunos de ellos varían en cuanto a complejidad, el grado de robustez de la marca de agua, la inaudibilidad de la misma, y la resistencia que presenta el algoritmo a los ataques que se llegara a someter el archivo marcado.

En este trabajo se propone la implementación en un FPGA (Field Programmable Gate Array) virtex II pro de xilinx, de un algoritmo robusto para la inserción de marcas de agua en señales de audio basado en la técnica de ocultamiento por eco, método que resulta relativamente sencillo de implementar debido a la baja complejidad matemática que implica, ya que consiste en agregar una señal idéntica a la original sólo que con un determinado retraso todo esto en el dominio del tiempo, la adición de múltiples ecos, etc., estas modificaciones pueden aumentar la inaudibilidad y la robustez de la marca de agua, características básicas de un sistema de marcado de agua digital en audio. Con la implementación antes mencionada, se pretende llevar a nivel de hardware el algoritmo de inserción, evitando el uso de la PC y de algún software de programación, para así obtener un sistema en el que a la entrada se introduzca el audio a marcar, se procesa en línea y a la salida se obtiene el mismo audio con la marca de agua incrustada.

# Índice

	<b>Página</b>
Resumen	IV
Agradecimientos	V
Planteamiento del Problema	VI
Estado de la Cuestión	VII
Preguntas de problematización	XII
Objetivos	XIII
Justificación	XIV
<b>Capítulo 1. Aproximaciones conceptuales del Mercado de Agua Digital en Audio</b>	
1.1 Señales de Audio	2
1.1.1 Características	2
1.2 Sistema Auditivo Humano	4
1.2.1 Estructura del Oído Humano	5
1.2.1.1 Estructura Física del Oído Humano	6
1.3 Efecto del Enmascaramiento	7
1.4 Marcas de Agua Digitales	9
1.4.1 Requerimientos	10
1.4.1.1 Clasificación de los Sistemas de Detección	11
1.4.2 Aplicaciones	11
1.5 Algoritmos de inserción de marcas de agua en audio	13
1.5.1 Proceso de Marcado	14
1.5.2 Proceso Intermedio (Ataques)	14
1.5.3 Proceso de Detección	15
1.5.4 Comparativa entre los algoritmos	16
1.5.4.1 Bit menos Significativo (LSB)	16
1.5.4.2 Patchwork Modificado	17
	XV

1.5.4.3 Espectro Extendido	18
1.5.4.4 Variación de Amplitud	19
<b>Capítulo 2. Características del Algoritmo de Ocultamiento por Eco</b>	
2.1 Inserción de la Marca de Agua	23
2.1.1 Parámetros Iniciales	24
2.1.2 Cálculos Efectuados	24
2.1.3 Esquema del Proceso de Marcado	26
2.2 Detección de la Marca de agua	28
2.2.1 Parámetros Iniciales	29
2.2.2 Esquema del Proceso de Detección	29
<b>Capítulo 3. Implementación y Pruebas</b>	
3.1 Entorno Matlab	33
3.2 Implementación del algoritmo de AM (Modulación de Amplitud)	33
3.2.1 Metodología de Inserción de AM	34
3.2.2 Metodología de Detección de AM	34
3.3 Implementación del algoritmo de Dos Conjuntos	35
3.3.1 Metodología de Inserción de Dos Conjuntos	35
3.3.2 Metodología de Detección de Dos Conjuntos	36
3.4 Implementación del algoritmo de Ocultamiento por Eco	37
3.4.1 Metodología de Inserción para Ocultamiento por Eco	37
3.4.2 Metodología de Detección para Ocultamiento por Eco	38
3.5 Pruebas de Robustez	40
3.6 Inaudibilidad de la marca de agua	42
3.7 Arquitectura del FPGA	43
3.8 Implementación del algoritmo en bloques de Xilinx	45
	XVI

3.8.1 Metodología de Inserción en Xilinx	53
3.8.2 Metodología de Detección en Xilinx	54
3.8.3 Implementación en el FPGA Virtex II pro de Xilinx	55
<b>Capítulo 4. Resultados Obtenidos</b>	
4.1 Resultados Obtenidos de la Implementación en MATLAB	58
4.1.1 Resultados para el algoritmo AM	58
4.1.2 Resultados para el algoritmo de Dos Conjuntos	60
4.1.3 Resultados para el algoritmo de Ocultamiento por Eco	62
4.2 Resultados de la Implementación en el FPGA del algoritmo de Ocultamiento por Eco	69
<b>Conclusiones</b>	71
<b>Referencias</b>	72
<b>Ciberografía</b>	73

# Índice de figuras

	<b>Página</b>
<b>Capítulo 1. Aproximaciones Conceptuales del Mercado de Agua Digital en Audio</b>	
Fig. 1.1 <b>A.</b> Línea cero <b>B.</b> Área de baja presión <b>C.</b> Área de alta presión	2
Fig. 1.2 Forma de onda completa <b>A.</b> Longitud de onda <b>B.</b> Amplitud <b>C.</b> Periodo	3
Fig. 1.3 Dos velocidades de muestreo <b>A.</b> Velocidad de muestreo baja, que distorsiona la forma de sonido original <b>B.</b> Velocidad de muestreo alta, que reproduce perfectamente la forma de sonido original	4
Fig. 1.4 Área de audición	5
Fig. 1.5 Estructura interna del oído humano	7
Fig. 1.6 Características de un Enmascaramiento Temporal	8
Fig. 1.7 Características de un enmascaramiento en Frecuencia	9
Fig. 1.8 Esquema general de Mercado de Agua Digital (Inserción y Detección)	13
Fig. 1.9 Proceso de Inserción de una Marca de Agua	14
Fig. 1.10 Procesamiento Intermedio (Ataques a la señal marcada)	15
Fig. 1.11 Proceso de Detección de la Marca de Agua	16
Fig. 1.12 Esquema del Algoritmo de inserción por el método LSB	16
Fig. 1.13 Esquema de inserción por el método de espectro extendido	18
Fig. 1.14 Esquema de inserción de marcas de agua mediante Variación de Amplitud	20
<b>Capítulo 2. Diseño de la Propuesta (Algoritmo basado en Ocultamiento por Eco)</b>	
Fig. 2.1 Cepstrum de la señal original con un retraso (eco) $d_1$	23
Fig. 2.2 Cepstrum de la señal original con un retraso (eco) $d_2$	23
Fig. 2.3 Esquema general del umbral de inserción en audio	26
Fig. 2.4 Segmentación en bloques del umbral de inserción	26
Fig. 2.5 Esquema de incrustación de un "0"	27

Fig. 2.6 Esquema de incrustación de un “1”	27
Fig. 2.7 Generación del archivo de audio marcado	27
Fig. 2.8 Umbral de detección genérico	29
Fig. 2.9 Umbral de detección genérico seccionado en bloques	30
Fig. 2.10 Gráfica del cepstrum (original y eco)	30

### **Capítulo 3. Implementación y Pruebas**

Fig. 3.1 Logo de MatLab	33
Fig. 3.2 Interfaz gráfica del algoritmo elegido	39
Fig. 3.3 Arquitectura del FPGA Virtex II Pro de Xilinx	43
Fig. 3.4 Tecnología activa de interconexión	45
Fig. 3.5 Bloque “counter”	45
Fig. 3.6 Parámetros establecidos para la implementación del Bloque “counter”	46
Fig. 3.7 Bloque “ROM”	46
Fig. 3.8 Parámetros del bloque “ROM”	47
Fig. 3.9 Bloque “Delay”	47
Fig. 3.10 Parámetros a modificar para el bloque “Delay”	47
Fig. 3.11 Bloque “Constant”	48
Fig. 3.12 Tabla de parámetros del bloque “Constant”	48
Fig. 3.13 Bloque “Relational”	48
Fig. 3.14 Parámetros del bloque “Relational”	48
Fig. 3.15 Bloque “CMult”	49
Fig. 3.16 Ventana de parámetros del bloque “CMult”	49
Fig. 3.17 Bloque “AddSub”	49
Fig. 3.18 Ventana de parámetros del bloque “AddSub”	50
Fig. 3.19 Bloque “Mux”	50
Fig. 3.20 Ventana de parámetros del bloque “Mux”	50
Fig. 3.21 Bloque “White Gaussian Noise Generator”	51

Fig. 3.22 Ventana de parámetros del bloque “White Gaussian Noise Generator”	51
Fig. 3.23 Bloque “Reinterpret”	51
Fig. 3.24 Ventana de parámetros del bloque “Reinterpret”	51
Fig. 3.25 Diagrama de Incrustación de un “1”	52
Fig. 3.26 Diagrama de Incrustación de un “0”	52
Fig. 3.27 Señal original y su correspondiente retraso	52
Fig. 3.28 Generador de secuencia binaria aleatoria (Marca de Agua)	53
Fig. 3.29 Diagrama en bloques de System Generator para incrustar 1 bit (“1” o “0”)	53
Fig. 3.30 Diagrama en bloques de System Generator para detectar la marca de agua	54
Fig. 3.31 Project Navigator	55

#### **Capítulo 4. Resultados Obtenidos**

Fig. 4.1 Señal de entrada y señal marcada para AM	57
Fig. 4.2 Correlación entre la señal marcada y la señal original	57
Fig. 4.3 Espectrograma de la señal de entrada y la marcada para AM	58
Fig. 4.4 Señal de entrada y señal marcada para Dos Conjuntos	59
Fig. 4.5 Correlación entre la señal marcada y la señal original	59
Fig. 4.6 Espectrograma de la señal de entrada y la marcada para Dos Conjuntos	60
Fig. 4.7 Señal de entrada y señal marcada para Ocultamiento por Eco	61
Fig. 4.8 Correlación entre la señal marcada y la señal original	61
Fig. 4.9 Espectrograma de ambas señales para Ocultamiento por Eco	62
Fig. 4.10 Prueba de inaudibilidad MOS	67
Fig. 4.11 Cepstrum para un bloque con un “1” incrustado	67
Fig. 4.12 Cepstrum para un “0” incrustado	68
Fig. 4.13 Marca de agua incrustada y marca de agua detectada	68
Fig. 4.14 Salida del sistema de inserción	69

# Índice de Tablas

## **Capítulo 2. Diseño de la Propuesta (Algoritmo basado en Ocultamiento por Eco)**

Tabla 2.1 Requerimientos para la detección ciega y no ciega	28
---	----

## **Capítulo 3. Implementación y Pruebas**

Tabla 3.1 Listado y descripción de los ataques	41
--	----

Tabla 3.2 Puntuación Media de Opinión	42
---------------------------------------	----

Tabla 3.3 Grado de Diferencia Objetiva	42
--	----

## **Capítulo 4. Resultados Obtenidos**

Tabla 4.1 Algunos ataques soportados y no soportados para el algoritmo AM	58
---	----

Tabla 4.2 Algunos ataques soportados y no soportados para el algoritmo de Dos Conjuntos	60
---	----

Tabla 4.3 Robustez de la marca de agua frente a ataques de “Remuestreo”	63
---	----

Tabla 4.4 Robustez de la marca de agua frente a ataques de “AWGN”	64
---	----

Tabla 4.5 Algunos ataques soportados y no soportados	65
--	----

Tabla 4.6 ODG de las categorías musicales	66
---	----

# Capítulo 1

## Aproximaciones Conceptuales del Mercado de Agua Digital en Audio

### Introducción

En esta sección se muestran de manera general los conceptos utilizados y que son necesarios conocer para el análisis de algoritmos de marcado de agua digital en audio. Además se verifican algunos algoritmos existentes para este tipo de procesos y la comparación entre ellos.

## 1.1 Señales de Audio

El sonido empieza con vibraciones en el aire [21] [22], como las que producen las cuerdas de una guitarra, las cuerdas vocales o un altavoz. Estas vibraciones fuerzan la unión de las moléculas cercanas de aire, lo que eleva ligeramente la presión de aire. Las moléculas de aire sometidas a presión empujan a las otras moléculas de aire que las rodean, que empujan a las moléculas colindantes, y así sucesivamente. Cuando las zonas de alta presión se desplazan por el aire, dejan detrás áreas de baja presión. Cuando estas oleadas de cambios de presión llegan hasta nosotros, vibran en los receptores de nuestros oídos y escuchamos las vibraciones en forma de sonido.

Cuando se observa una forma de onda visual que representa audio, refleja estas ondas de presión de aire. La línea cero de la forma de onda es la presión del aire en reposo. Cuando la línea sube a un pico, representa una presión más elevada; si baja a un valle, representa una presión más baja.

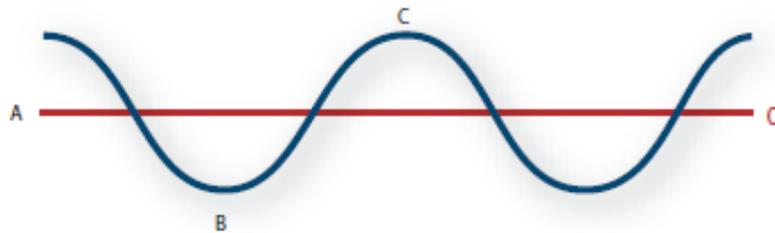


Fig. 1.1 A. Línea cero B. Área de baja presión C. Área de alta presión

### 1.1.1 Características

Algunas medidas describen las formas de onda:

- ✓ **Amplitud:** Refleja el cambio de presión desde el pico de la forma de onda hasta el mínimo. Las formas de onda de alta amplitud son altas; las de baja amplitud son más silenciosas.
- ✓ **Ciclo:** Describe una única secuencia repetida de cambios de presión, desde presión cero a alta presión, a baja presión y de nuevo a cero.
- ✓ **Frecuencia:** Se mide en hertz (Hz) y describe el número de ciclos por segundo. (Por ejemplo, una forma de onda de 1.000 Hz tiene 1.000 ciclos)

por segundo.) Cuando mayor sea la frecuencia, más alto será el tono musical.

- ✓ **Fase:** Se mide en 360 grados e indica la posición de una forma de onda en un ciclo. Cero grados es el punto de inicio, seguido de 90° a alta presión, 180° en el punto central, 270° a baja presión y 360° en el punto final.
- ✓ **Longitud de onda:** Se mide en unidades, como pulgadas o centímetros, y es la distancia entre dos puntos con el mismo grado de fase. A medida que aumenta la frecuencia, disminuye la longitud de onda.

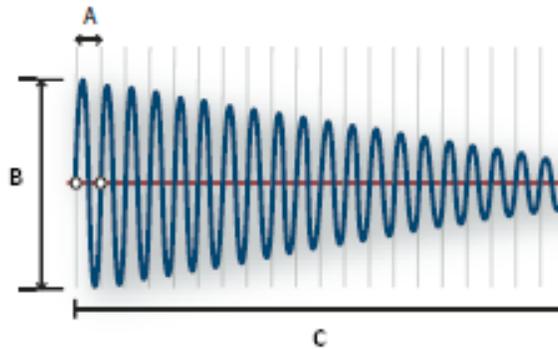


Fig. 1.2 Forma de onda completa **A.** Longitud de onda **B.** Amplitud **C.** Periodo

Un micrófono convierte las ondas de sonido bajo presión en cambios de tensión en un cable: la alta presión se convierte en tensión positiva, mientras que la baja presión lo hace en negativa. Cuando estos cambios de tensión viajan a través de un cable de micrófono, puede grabarse en cinta como cambios en intensidad magnética o en discos de vinilo. Un altavoz funciona como un micrófono, pero a la inversa: toma las señales de tensión de un audio que graba y vibra para volver a crear la onda de presión.

A diferencia de los medios de almacenamiento analógicos, como las cintas magnéticas o los discos de vinilo, los equipos informáticos almacenan información de audio de forma digital como una serie de ceros y unos. En el almacenamiento digital, la forma de onda original se desglosa en instantáneas individuales denominadas muestras. Este proceso se conoce normalmente como digitalización o muestreo del audio, pero en ocasiones recibe el nombre de conversión de analógico a digital.

Cuando graba en un equipo desde un micrófono, por ejemplo, los conversores de analógico a digital transforman la señal analógica en muestras digitales que los equipos pueden almacenar y procesar.

Las velocidades de muestreo indican el número de instantáneas digitales que se toman en una señal de audio cada segundo.

Esta velocidad determina el intervalo de frecuencias de un archivo de audio. Cuanto más alta sea la velocidad de muestreo, más se asemejará la forma de la onda digital a la forma de la onda analógica original. Las velocidades de muestreo bajas limitan el intervalo de frecuencias que pueden grabarse, lo que puede dar como resultado una grabación que no representa correctamente el sonido original.

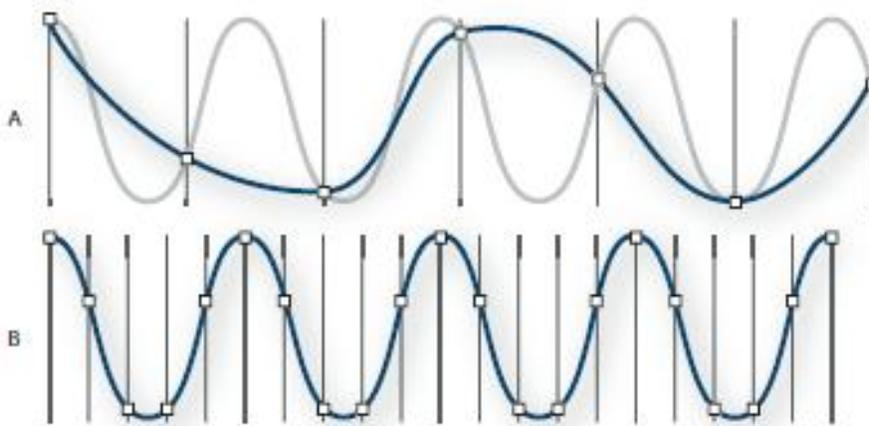


Fig. 1.3 Dos velocidades de muestreo **A.** Velocidad de muestreo baja, que distorsiona la forma de sonido original **B.** Velocidad de muestreo alta, que reproduce perfectamente la forma de sonido original

## 1.2 Sistema Auditivo Humano

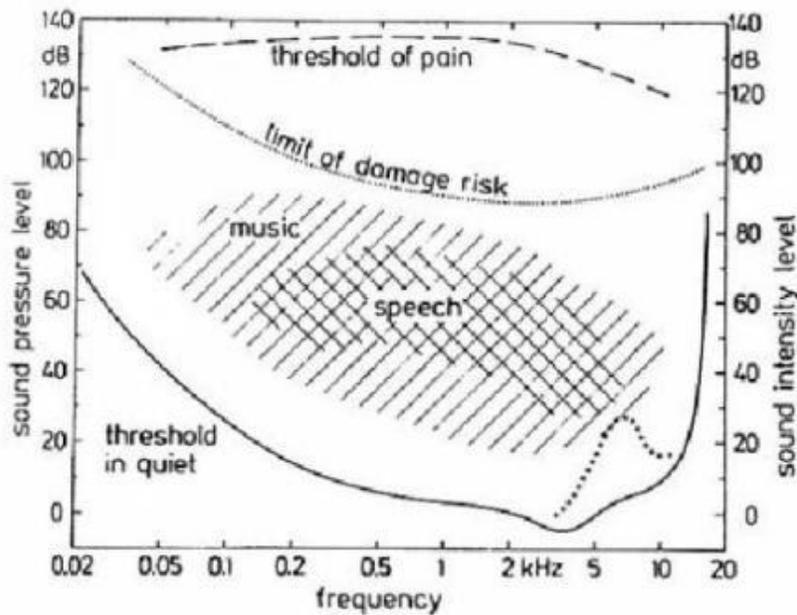
Las características de nuestro oído son cruciales a la hora de diseñar un algoritmo de marcado. El marcado digital de señales de audio explota las propiedades del Sistema Auditivo Humano (HAS) para introducir la marca en aquellas zonas en las que el oído es menos sensible.

El ser humano es capaz de detectar únicamente aquellos sonidos que se encuentren dentro de un determinado rango de amplitudes y frecuencias. En este sentido, se puede establecer una analogía entre el aparato auditivo y un sistema electrónico de audio: en base al concepto convencional del rango dinámico.

Se define el rango dinámico del oído como la relación entre la máxima potencia sonora que éste puede manejar y la mínima potencia necesaria para detectar un sonido. Asimismo, el rango de frecuencias asignado convencionalmente al sistema auditivo va desde los 20 Hz hasta los 20 kHz, aún cuando este rango puede variar de un sujeto a otro o disminuir en función de la edad del sujeto, de trastornos

auditivos o de una pérdida de sensibilidad debida a la exposición a sonidos de elevada intensidad. Ahora bien, la sensibilidad del sistema auditivo no es independiente de la frecuencia; por el contrario, dos sonidos de igual presión sonora pueden provocar distintas sensaciones de intensidad o sonoridad, dependiendo de su contenido espectral.

En la figura 1.4 podemos apreciar el gráfico que relaciona el nivel de presión sonora (dado en dB) y la frecuencia (dado en Hz) de una señal audible y el efecto que produce en el oído humano, se clasifican los espacios en los que el oído humano puede escuchar sin problema alguno (música y voz) y los espacios en los que es susceptible a riesgo de daños y dolor que puede producir lo que se escuche en esos rangos.



**Umbrals de audibilitat.**

Fig. 1.4 Àrea de audició

### 1.2.1 Estructura del oído humano

La audición es el resultado de una serie de procesos acústicos, mecánicos, nerviosos y mentales dentro de la combinación oído/cerebro que dan a una persona la impresión del sonido [26]. La impresión que un humano recibe no es idéntica a la forma de onda acústica verdadera presente en el canal auditivo porque parte de la entropía de la onda se pierde.

La agudeza del oído humano es asombrosa, ya que puede detectar cantidades minúsculas de distorsión y aceptar un enorme rango dinámico. El único criterio de calidad de que se dispone consiste en el hecho de que si el oído es incapaz de detectar distorsión alguna, se dice que el sonido es perfecto. Por tanto, el criterio de calidad es completamente subjetivo y sólo se puede comprobar mediante pruebas de audición.

### 1.2.1.1 Estructura física del oído humano

El oído se divide en tres zonas [26]; oído externo, oído medio y oído interno, de acuerdo a su ubicación en el cráneo. El oído externo es la parte del aparato auditivo que se encuentra en posición lateral al tímpano. Comprende el pabellón auditivo externo y el conducto auditivo externo, que mide tres centímetros de longitud.

En la siguiente Figura 1.5 se muestra un esquema de la anatomía del oído humano, pudiendo distinguirse:

- **Oído medio:** Se encuentra situado en la cavidad timpánica, cuya cara externa está formada por el tímpano, que lo separa del oído externo. Incluye el mecanismo responsable de la conducción de las ondas sonoras hacia el oído medio y actúa como un transformador adaptador de impedancias que mejora la transferencia de potencia.
- **Oído interno:** Se encuentra en el interior del hueso temporal que contiene los órganos encargados del equilibrio. Está separado del oído medio por la ventana oval. El oído interno consiste en una serie de canales membranosos alojados en una parte densa del hueso temporal y está dividido en: cóclea, vestíbulo y tres canales semicirculares (Figura 1.5). Estos tres canales se comunican entre sí y contienen un fluido gelatinoso denominado endolinfa.

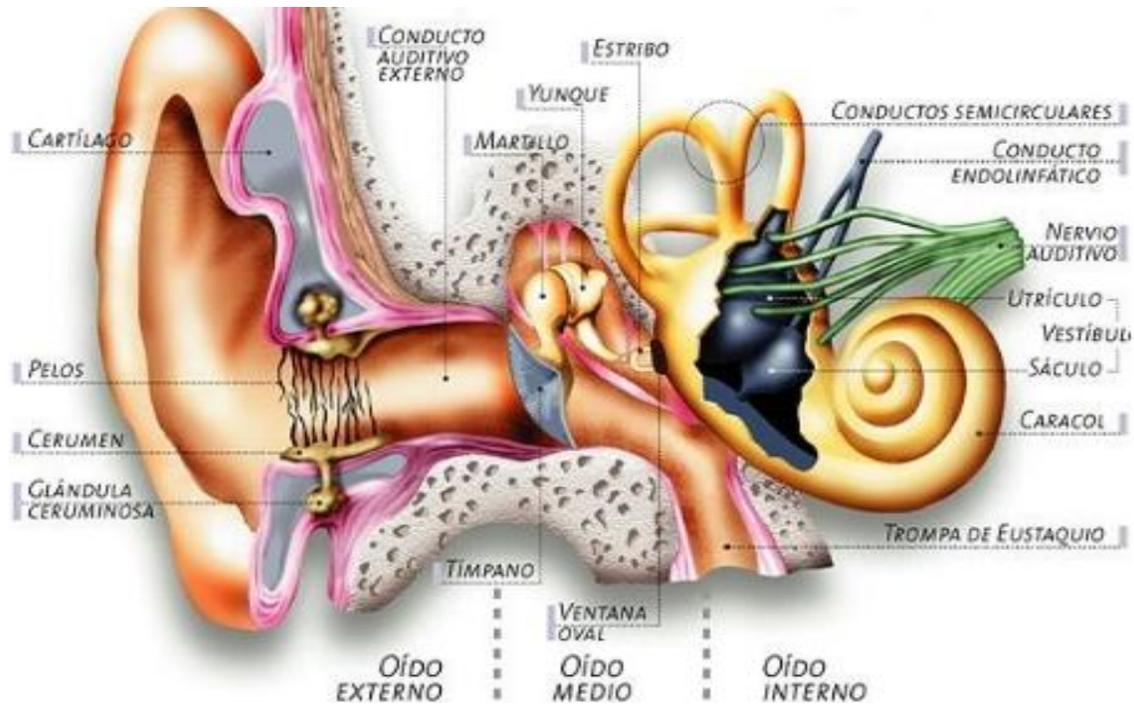


Fig. 1.5 Estructura interna del oído humano

### 1.3 Efecto del Enmascaramiento

El umbral de audibilidad [1] es en el que un tono puro de una frecuencia dada, a la mínima presión sonora eficaz que puede ser escuchada, en ausencia de todo ruido de fondo. El umbral auditivo representa la presión sonora mínima que produce la sensación de audición.

En el campo auditivo el rango de frecuencia audible va desde los 20Hz a los 20kHz (teóricamente), pero el oído no es igualmente sensible a todas estas frecuencias. Las más audibles son las ubicadas en el medio del espectro, aproximadamente entre 1kHz y 5 kHz. El oído es menos sensible para frecuencias bajas y altas. Esta característica de menor agudeza para los tonos graves favorece el enmascaramiento de los sonidos que produce el cuerpo humano.

Se denomina enmascaramiento a la reducción total o parcial de la sensibilidad de un oyente para percibir un determinado sonido, provocado por la presencia simultánea de otro. Cuando un sonido hace que otro sea menos audible, porque ambos se producen al mismo tiempo, se dice que se produjo un fenómeno de enmascaramiento.

El sonido cuyo umbral de audibilidad se ha modificado se denomina sonido enmascarado y al otro, sonido enmascarador.

La interacción entre dos estímulos presentados al mismo tiempo depende en gran medida de las características de los sonidos. Existen algunos lineamientos que rigen el enmascaramiento:

- ✓ Un sonido posee mayor poder para enmascarar, si se intenta enmascarar con otro que tenga una frecuencia parecida. En cambio, resulta muy difícil de enmascarar con otro de frecuencia diferente, alejada en el espectro. El enmascaramiento podrá realizarse, pero el nivel de precisión sonora necesario tendrá que ser más importante que el primer caso.
- ✓ Un sonido de determinada frecuencia tiene más poder para enmascarar sobre otro de frecuencia más aguda, que sobre otro de frecuencia más grave. Por lo tanto, si se está buscando el efecto de enmascaramiento, es más fácil lograrlo si el sonido que enmascara es de frecuencia más grave que el enmascarado.

Podemos dividir el enmascaramiento sonoro entre enmascaramiento temporal y enmascaramiento en frecuencia. A continuación se muestran dos esquemas que plasman las características y diferencias entre los dos tipos de enmascaramiento que existen.



Fig. 1.6 Características de un Enmascaramiento Temporal



Fig. 1.7 Características de un enmascaramiento en Frecuencia

## 1.4 Marcas de Agua Digitales

En la actualidad, el intercambio de imágenes, audio, video, texto y demás archivos multimedia se ha vuelto una actividad cotidiana a lo largo del mundo, pero este tipo de acciones llevan a que los dichos archivos sean manipulados y/o distribuidos de manera ilegal, generar copias sin el permiso del autor y violando las leyes que protegen a los propietarios de obras intelectuales. Para solucionar este problema se ha creado una técnica que incrusta datos en la propiedad intelectual a manera de identificador, esta técnica se conoce como marcado de agua digital. Existen diversos algoritmos para incrustar marcas de agua en imágenes y video, sin embargo, relativamente pocos son los usados para incrustar datos en señales de audio debido a la complejidad del sistema auditivo humano [2].

El marcado de agua digital (watermarking) es una técnica esteganográfica, que consiste en transmitir un mensaje oculto entre un transmisor y un receptor.

Las marcas de agua digitales han sido propuestas como una solución eficiente para la protección de los derechos de copia y propiedad de los archivos de datos multimedia, posibilitando la identificación de la fuente, autor, propietario, distribuidor o consumidor autorizado, de imágenes digitales, grabaciones de audio o video. La principal ventaja de estos sistemas consiste en que la marca es inseparable del contenido del archivo. Sin embargo, existen algunas cuestiones que necesitan ser resueltas, antes de que estas técnicas puedan ser eficazmente aplicadas en los escenarios de la vida real.

La marca de agua digital es un código de identificación que se inserta directamente en el contenido de un archivo multimedia (imagen, audio, video), de manera que sea difícil de apreciar por el sistema perceptual humano, pero fácil de detectar usando un algoritmo dado y una clave.

Un sistema de marcas de agua involucra un proceso de marcado y otro de detección que, generalmente, requieren una clave de propósito similar a la clave utilizada en los sistemas esteganográficos. El nivel de disponibilidad de la clave, determinará quién o quiénes podrán leer o detectar la marca de agua. En la práctica, la mayoría de las técnicas de marcas de agua pueden considerarse como sistemas esteganográficos simétricos, en los que se emplea una sola clave, variando en ellos el nivel de acceso a esa clave [2].

Las marcas de agua en audio se basan en la imperfección del sistema auditivo del ser humano. Sin embargo, cabe recalcar que el oído humano es mucho más sensible que cualquier otro sentido sensorial en el cuerpo humano.

Aunque en la actualidad las técnicas de marcas de agua en audio están lejos de la perfección, durante la última década los sistemas de marcas de agua han sido aplicados ampliamente. Estos sistemas son muy sofisticados tanto en términos de robustez como en términos de imperceptibilidad. La robustez y la imperceptibilidad son requerimientos importantes en las marcas de agua.

#### 1.4.1 Requerimientos

Para una marca de agua digital de audio, se numeran las siguientes características deseables [1]:

- ✓ **Inaudible:** La marca de agua digital integrada en el audio debe ser inaudible para el oído humano.
- ✓ **Seguridad:** Una eliminación desautorizada y/o detección de la marca de agua deben ser imposibles incluso si el sistema básico usado para marcado de agua es conocido.
- ✓ **Fragilidad:** Una marca de agua frágil tiene como objetivo probar autenticidad de la señal, por lo tanto, no tiene que ser robusta contra técnicas de procesamiento o ataques intencionales de la señal, puesto que una falla en el proceso de detección muestra que la señal ha sido alterada y no es auténtica más.

- ✓ **Robustez:** La marca de agua debería ser imposible de manipular por operaciones intencionales o no intencionales sin degradar la calidad percibida a un punto significativo reduciendo su valor comercial. Tales operaciones son, por ejemplo, filtrado, muestreo, compresión, ruido, cultivo, conversión A/D-D/A, etc.
- ✓ **Constante de tasa de bit:** El dominio de flujo de bit no debe incrementar la tasa de bit.
- ✓ **Transparencia Perceptual:** Los algoritmos de inserción de marcas de agua tienen que insertar datos adicionales sin afectar la calidad perceptual de la señal original de audio.
- ✓ **Fidelidad:** La fidelidad de una marca de agua es usualmente definida por una similaridad perceptual entre la secuencia de audio original y la secuencia de audio marcada.

#### 1.4.1.1 Clasificación de los Sistemas de Detección

Los sistemas de detección se clasifican en dos rubros, a continuación se define cada uno de ellos [2]:

- ✓ **Sistemas de Detección Ciegos.** Es aquel que en el proceso de detección no necesita conocer los datos originales para detectar la marca, sólo con la clave y la señal marcada el sistema será capaz de detectar la marca de agua incrustada en el proceso de inserción.
- ✓ **Sistemas de Detección No Ciegos.** Es aquel que en el proceso de detección necesita conocer los datos originales para estimar la marca incrustada. Un sistema Ciego sería ideal pero su desventaja radica en su complejidad, ya que se necesitan mecanismos de autodetección y sincronización.

#### 1.4.5 Aplicaciones

Las aplicaciones para las que pueden servir los sistemas de marcado de agua digital en audio [2-4], dependen en gran medida de los requerimientos que cumplen y de las características o parámetros con los que fue diseñado el sistema. A continuación se enlistan las aplicaciones más comunes:

## ✓ **Copyright**

Las aplicaciones relacionadas con los Derechos de Autor fueron las que en primer lugar motivaron el desarrollo del watermarking para audio, y es también una aplicación de gran importancia en sistemas de huellas digitales. Abajo se muestran algunas derivaciones del copyright:

- Prueba de propiedad: Utilizar la marca de agua como un identificador para poder ser capaz de comprobar que es el legítimo dueño de la canción.
- Identificación de emisión de audio: Las estaciones de radio deben pagar por las canciones que emiten. Los propietarios monitorizan las transmisiones de radio y así verificar que se estén pagando debidamente los derechos de autor.
- Rastreo de copias ilícitas: Consiste en marcar las grabaciones para que sean protegidas antes de su distribución y así poder utilizar sistemas de rastreo de archivos de audio.
- Determinación del origen de copias ilícitas: Sucede cuando se encuentra una copia del archivo en un sitio web sin la autorización del autor pero el propietario de dicho sitio es el último eslabón de una cadena de piratería, entonces se intenta buscar el origen de dicha cadena cuando se infringió la política de uso de dicho material.

## ✓ **Servicios de valor añadido**

En este tipo de aplicaciones se utilizan sistemas de marcado de agua digital sólo para añadir valor a la canción, es decir, sirve como transporte de información de la obra y sólo nos preocuparíamos que fuera robusta frente a ataques no intencionados como lo son, filtrado, compresión, conversión A/D, etc.) También se utilizan para marcar el audio con texto, como en las estaciones de radio que marcan el audio con el nombre de la canción, artista y el nombre de la estación de radio.

## ✓ **Verificación de la Integridad**

Aplicación utilizada en testimoniales, por ejemplo, cuando un testimonio previamente grabado es usado como evidencia en un tribunal, se pretende asegurar que la marca de agua no haya sido editada, modificada o alterada por alguna de las partes involucradas según les convenga para que esto suceda las marcas de agua deben ser frágiles (concepto contrario a la robustez). Otra aplicación muy común en estos tiempos es el control de los tonos polifónicos para

celular que se comercializan, con la marca de agua podría haber un control mayor sobre copias ilícitas.

## 1.5 Algoritmos de Inserción de Marcas de agua en Audio

Un algoritmo de inserción de marcas de agua digitales consta básicamente de tres procesos [2-5]:

- ✓ Proceso de inserción (Transmisor)
- ✓ Ataques intermedios (Medio)
- ✓ Proceso de detección (Receptor)

A continuación se muestra el esquema general del marcado de agua digital (Watermarking):

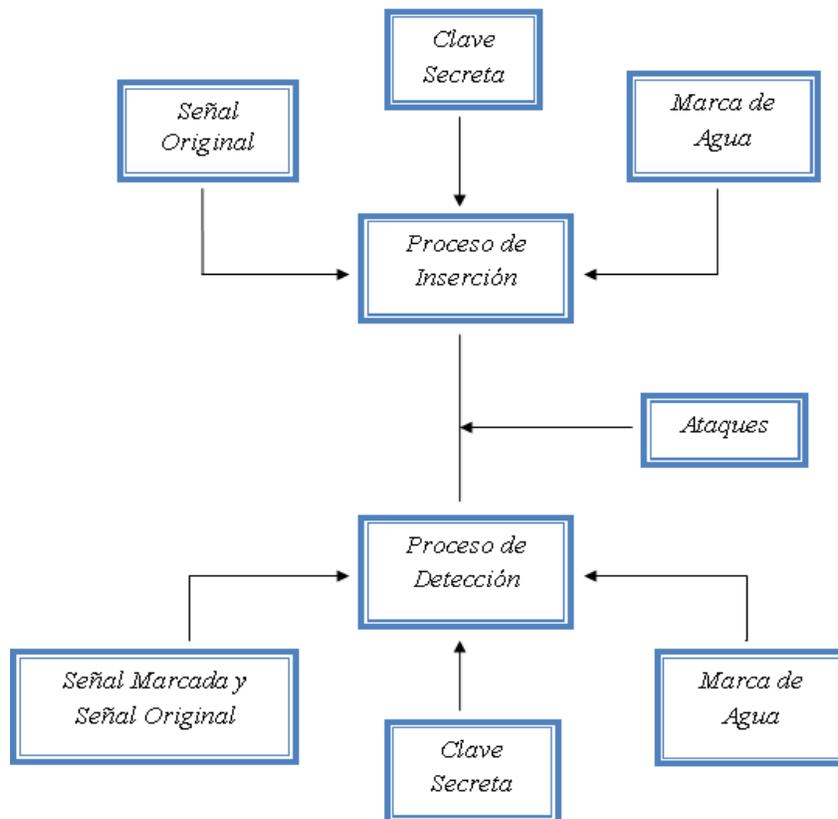


Fig. 1.8 Esquema general de Marcado de Agua Digital (Inserción y Detección)

La marca digital contiene datos que pueden ser utilizados en distintas aplicaciones, consiste en una secuencia de dígitos binarios. A primera vista podría

parecer que la información que puede ir en la marca está limitada, por el número de bits máximo que ésta pueda contener, pero en ocasiones, el contenido de la marca de agua es la entrada a una base de datos que contenga la información deseada. De esta forma se consigue que la información que se puede introducir sea muy elevada y no se encuentre limitada por la técnica usada o la longitud de la señal a marcar.

### 1.5.1 Proceso de Marcado

Como se puede ver en la Figura 1.9 para el proceso de marcado se necesita la señal original (señal huésped en donde se incrustará la marca de agua), asociar una clave secreta para poder extraer la marca en el proceso de detección y la propia marca de agua, todo esto será procesado mediante el algoritmo de marcado elegido.

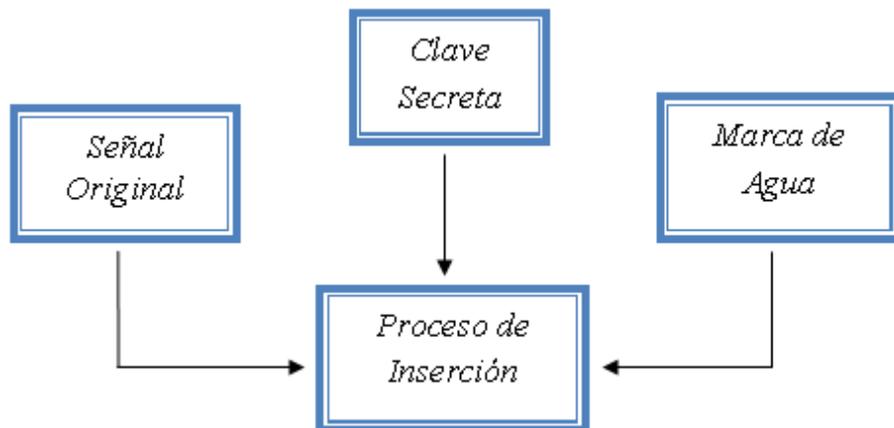


Fig. 1.9 Proceso de Inserción de una Marca de Agua

### 1.5.2 Proceso Intermedio (Ataques)

La señal marcada sufrirá un procesamiento intermedio, conocido como los ataques, desde que es marcada hasta que un receptor se propone extraer dicha marca. Este procesamiento que sufre la señal se considerará a efectos del sistema como un ataque a la misma. Ya que tras él, la señal se ha modificado de alguna manera y esto puede afectar negativamente a la marca que lleva el archivo. El ataque puede ser:

- ✓ **Intencionado:** Aquéllos que intentan modificar o borrar la marca a propósito.
- ✓ **No intencionado:** Son modificaciones que sufre el audio. Por ejemplo, que la canción pase por un canal altamente ruidoso (ruido de transmisión), que pase por una etapa de remuestreo, que se comprima a formato MP3, que sufra conversión A/D o D/A etc.

Los algoritmos de marcado deben diseñarse para que la marca digital insertada sea robusta frente a estos ataques, ya sean intencionados o no. En contraste, existen ciertas aplicaciones en las que conviene que la marca sea frágil ante estos ataques.

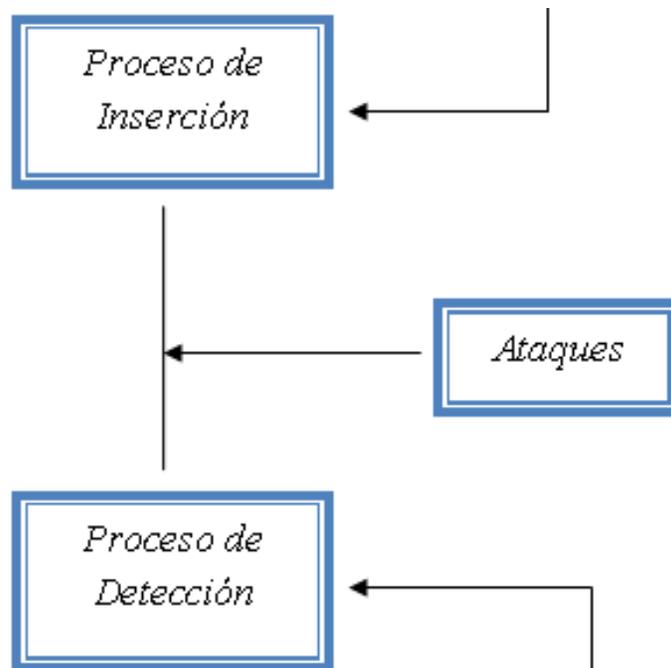


Fig. 1.10 Procesamiento Intermedio (Ataques a la señal marcada)

### 1.5.3 Proceso de Detección

Este proceso consiste básicamente en descifrar y extraer la marca de agua que originalmente fue insertada, en la detección se necesita la señal marcada (la señal a la que se le incrustó la marca de agua), la señal original, la clave secreta y en algunos casos dependiendo del sistema de detección usado.

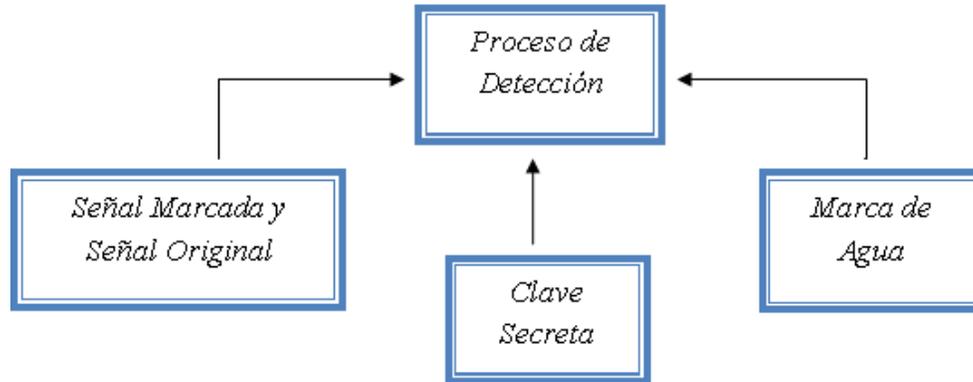


Fig. 1.11 Proceso de Detección de la Marca de Agua

### 1.4.5 Comparativa entre algoritmos

A continuación se describen algunos de los algoritmos más comunes utilizados en el marcado de agua digital de audio [3-6]:

#### 1.4.5.1 Bit menos significativo (LSB)

El método del Bit menos significativo se basa en la sustitución del bit mencionado de la señal portadora por el patrón de bits de la señal de marca de agua. Es un método que opera en el dominio temporal y es uno de los más sencillos utilizados para el marcado de agua digital. Este método, por lo general, no se desarrolla en bajo las características del modelo psicoacústico con las que podría minimizar algún factor a la distorsión ocasionada por el cambio de la codificación del bit menos significativo. En literatura consultada como [] contribuyen al mejoramiento del algoritmo con la introducción de un factor que se relaciona directamente con el sistema auditivo humano y con éste mejorar la robustez y la imperceptibilidad de la marca de agua incrustada. La metodología de inserción se muestra a continuación:

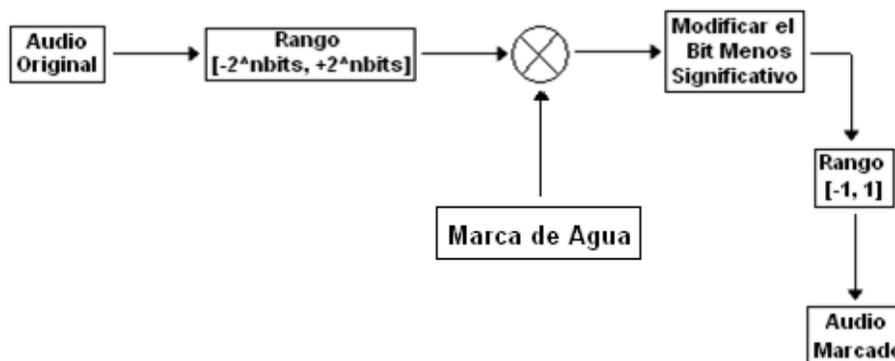


Fig. 1.12 Esquema del Algoritmo de inserción por el método LSB

Para el proceso de detección, sólo se necesita extraer la secuencia de bit menos significativo y esa será la secuencia de marca incrustada en el proceso de inserción. Algunas ventajas de este método son la alta imperceptibilidad que proporciona el proceso de inserción y la cantidad alta de bits que es posible incrustar, pero, la desventaja principal es que el método no muestra un grado de robustez suficiente para sobrevivir a los procesos a los que se necesite someter el audio marcado tales como; compresión, adición de ruido, filtraje, etc., ya que la marca de agua se destruye completamente cuando estos ataques suceden.

### 1.4.5.2 Patchwork Modificado

El algoritmo de Patchwork Modificado trabaja en el dominio de la frecuencia, debido a que directamente interviene la Transformada Discreta del Coseno (DCT). El algoritmo puede utilizar otros dominios de frecuencia pero se seleccionó la DCT debido a que ésta no contiene componentes imaginarias debido a que sólo se obtienen los coeficientes cosenoidales. La metodología del algoritmo consiste en:

- a) Dividir el audio en bloques de tamaño N
- b) Generar una secuencia binaria usando una llave secreta
- c) Se generan dos conjuntos de índices  $I^0$  e  $I^1$
- d) A cada bloque se le aplica la DCT (Discrete Cosine Transform)
- e) Se generan dos subconjuntos del conjunto total, uno llamado  $A=[a_1, a_2, a_3, \dots, a_n]$  y uno llamado  $B=[b_1, b_2, b_3, \dots, b_n]$
- f) Se calculan las medias muestrales y el error estadístico estándar de cada bloque con las siguientes ecuaciones:

$$\bar{a} = \frac{1}{n} \sum_{i=1}^n a_i \quad \bar{b} = \frac{1}{n} \sum_{i=1}^n b_i \quad (1)$$

$$S = \sqrt{\frac{\sum_{i=1}^n (a_i - \bar{a})^2 + \sum_{i=1}^n (b_i - \bar{b})^2}{n(n-1)}} \quad (2)$$

- g) Y por último se aplican las ecuaciones de inserción siguientes y se les aplica la IDCT (Inverse Discrete Cosine Transform) a cada conjunto de muestras:

$$a_i^* = a_i + \text{sign}(\bar{a} - \bar{b}) \sqrt{C} \frac{S}{2} \quad b_i^* = b_i - \text{sign}(\bar{a} - \bar{b}) \sqrt{C} \frac{S}{2} \quad (3)$$

Para el proceso de detección, se asocian la marca de agua y la llave secreta a una semilla de un generador de números aleatorios. Se obtienen dos subconjuntos;  $A_0=[a_{01}, \dots, a_{0n}]$  y  $B_0=[b_{01}, \dots, b_{0n}]$  para detectar un bit "0" y

$A_1=[a_{11},\dots,a_{1n}]$  y  $B_1=[b_{11},\dots,b_{1n}]$  para detectar un bit “1”, se calculan los parámetros estadísticos para cada grupo de muestras con la siguiente ecuación:

$$T^2 = \frac{(\bar{a}-\bar{b})^2}{s^2} \quad (4)$$

Se define un solo parámetro de T cómo el máximo de cada uno de los bloques y ahora se compara este valor con un valor de umbral para decidir si está insertado un “1” o un “0” y se genera la marca de agua asociando a la semilla obteniendo así los bits incrustados.

### 1.4.5.3 Espectro Extendido

El espectro extendido es una técnica general para incrustar marcas de agua que puede ser implementada en algún dominio de transformación o en el dominio del tiempo. La marca de agua es agregada a la cobertura total del dominio en el que se esté trabajando, y como tal, con una marca de compuesta por números pseudo-aleatorios y la escala apropiada, resulta ser discreta e imperceptible.

Para incrustar una marca de agua con la técnica de espectro extendido primero la señal portadora debe ser transformada en el dominio deseado, por ejemplo, se puede dividir el audio en bloques de tamaño N con una duración M y después transformar cada uno en el dominio de la frecuencia a través de la transformada rápida de Fourier (FFT). Para efectos de inaudibilidad, el tamaño de los bloques podría ser de 3969 muestras con una duración de 90ms. La figura 2.12 muestra el diagrama general de la técnica de espectro extendido.

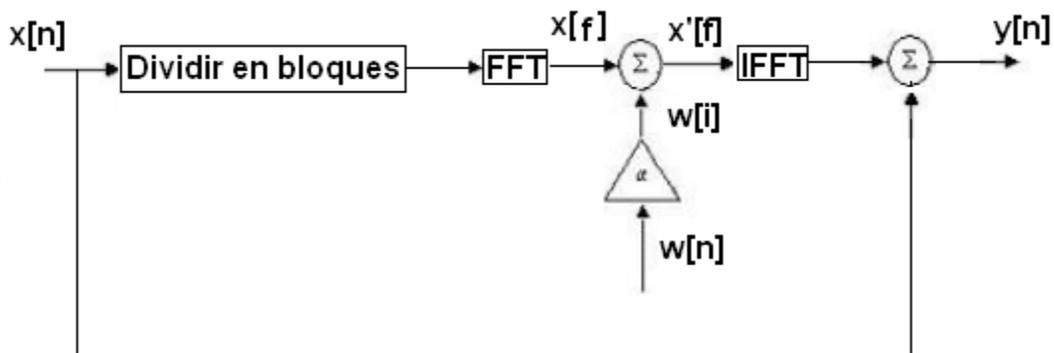


Fig. 1.13 Esquema de inserción por el método de espectro extendido

Una vez que se haya elegido el dominio a trabajar, una versión escalada de la secuencia de marca de agua pseudo-aleatoria que pertenece a (-1,1) es agregada a la señal

$$x'[f] = x[f] + \alpha * w[i] \quad (5)$$

Donde  $x[f]$  es la transformada de Fourier de cada uno de los bloques y  $w[i]$  es el correspondiente valor de marca de agua de la versión ensanchada de la señal de marca de agua original. El factor de escalamiento  $\alpha$  es elegido para que sea tan alto como sea posible mientras sigue siendo inaudible, esto es para garantizar la máxima resistencia a los ataques basados en el ruido. Cada bloque se transforma de nuevo en el dominio del tiempo y se añade a la parte correspondiente de la señal de entrada.

Para detectar la marca de agua incrustada por espectro extendido existen diferentes maneras de hacerlo. La idea básica es simplemente aplicar la inversa del proceso de inserción, para esto se requiere conocer el archivo original antes de incrustarle la marca de agua. La siguiente ecuación nos muestra la diferencia de los bloques procesados reducidos en un factor de escala  $\alpha$ :

$$w[i] = (y[n] - x[f])/\alpha \quad (6)$$

El resultado es un promedio sobre el tamaño de bloque. Si el promedio es mayor que cero el bit de marca de agua correspondiente es un 1 y si es inferior a cero el bit de marca de agua correspondiente a -1.

#### 1.4.5.4 Variación de Amplitud

La variación de amplitud es quizá el esquema más simple de inserción de marcas de agua en audio. El sistema consiste en una ligera variación de la de la amplitud de la señal huésped original de acuerdo con la marca de agua que será incorporada como se muestra en la figura 2.13. Variando la amplitud de la señal original por un pequeño porcentaje se produce una variación inaudible resultando una señal marcada que sonoriza igual que la señal original.

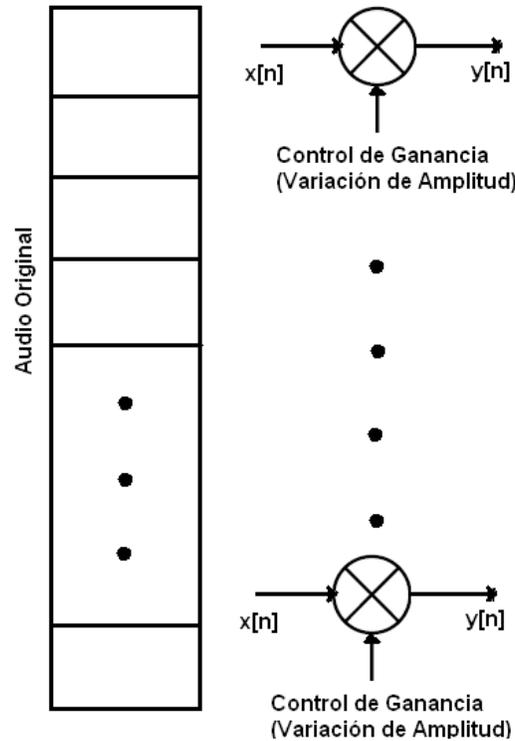


Fig. 1.14 Esquema de inserción de marcas de agua mediante Variación de Amplitud

El proceso de inserción la marca de agua se extiende a través de bloques de la señal original. Para contener el archivo de audio con marca de agua, se genera un vector de salida. Por cada bit de marca de agua que sea incrustado, la amplitud de todas las muestras en un bloque dado se modifican (o se escalan) en función de si es un bit '1' o un bit '0' lo que se vaya a incrustar. Si se incrusta un '1' todas las muestras en el bloque corresponderán a una magnitud  $\alpha$  de la señal original y si lo que se desea incrustar es un '0' todas las muestras en el bloque corresponderán a una magnitud  $\beta$  de la misma señal original. Estas variaciones son inaudibles para el oído humano pero, en consecuencia, altamente afectado por el ruido aditivo. Las siguientes ecuaciones muestran las condiciones de inserción para cada bit a incrustar:

$$y(n) = \begin{cases} \alpha * x(n); & W_i = 1 \\ \beta * x(n); & W_i = 0 \end{cases} \quad (7)$$

Para detectar la marca incrustada por el método de variación de la amplitud, se necesita tener conocimiento de los coeficientes de variación de  $\alpha$  y  $\beta$ , y del audio original antes de ser marcado. Suponiendo que el archivo de audio original está disponible, la marca de agua incrustada  $W_i$  puede ser recuperada mediante la comparación de la amplitud relativa de la señal de marca de agua a la señal principal, como lo muestra la siguiente ecuación:

$$y(n) = \begin{cases} 1 & \text{si } y(n) = \alpha * x(n) \\ 0 & \text{si } y(n) = \beta * x(n) \end{cases} \quad (8)$$

A continuación en el capítulo 2 se presenta el desarrollo del algoritmo seleccionado para su implementación, el proceso de inserción, los cálculos efectuados para el enmascaramiento (inaudibilidad) y el proceso de detección.

# Capítulo 2

## Diseño de la Propuesta (Algoritmo basado en Ocultamiento por Eco)

### Introducción

Este apartado muestra y detalla las características del algoritmo de inserción de marcas de agua seleccionado: El Ocultamiento por Eco

## 2.1 Inserción de la Marca de Agua

El algoritmo elegido fue el de Ocultamiento de Marcas de Agua a través de insertar ecos de la señal a marcar [7-11]. Este algoritmo se desarrolla en el dominio del tiempo, de ahí su facilidad de manejo y su baja complejidad computacional.

El algoritmo consiste en agregar pequeños retrasos a la señal original, un retraso para cada bit a insertar [1,0] y cada retraso con una amplitud propia, esto con la finalidad de enmascararlo audiblemente hablando. Visualmente si graficamos el espectro de las señales con un 1 insertado o con un 0, veremos las siguientes figuras:

### Cepstrum Núcleo "0"

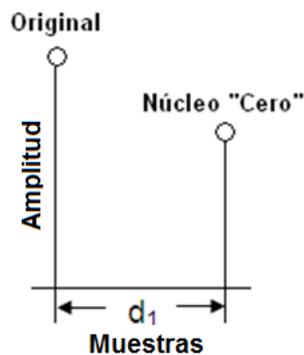


Fig. 2.1 Cepstrum de la señal original con un retraso (eco)  $d_1$

### Cepstrum Núcleo "1"

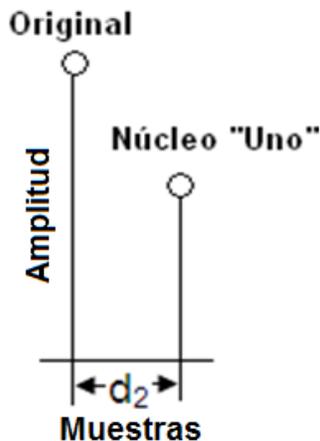


Fig. 2.2 Cepstrum de la señal original con un retraso (eco)  $d_2$

### 2.1.1 Parámetros Iniciales

A continuación se describen los parámetros iniciales de los elementos utilizados para realizar la metodología de inserción:

Los archivos musicales procesados contienen las siguientes características:

- ✓ Formato: WAV
- ✓ Frecuencia de muestreo: 44100kHz
- ✓ Tamaño promedio: 20MB
- ✓ Longitud promedio del audio: 10'000,000 de muestras
- ✓ Categoría Musical: Pop, Rock, Balada, Banda, Electrónica
- ✓ Longitud de marca de agua: 120 bits
- ✓ Tamaño de los bloques de inserción: 1024 muestras
- ✓ Condiciones de Inserción:

$$y(n) = \begin{cases} \alpha * x(n); & w_i = 0 \\ \alpha * x(n) + \beta * x(n - d); & w_i = 1 \end{cases} \quad (9)$$

Cabe mencionar que las características antes mencionadas se aplican para archivos de audio de duración aproximada de 4 minutos, tiempo que generalmente es lo que dura una canción.

### 2.1.2 Cálculos Efectuados

Para realizar el proceso de inserción, se necesita calcular el número de muestras que se debe retrasar la señal original con respecto a las condiciones de inserción antes mencionadas. Dicho cálculo se debe realizar de acuerdo a la teoría del enmascaramiento temporal, en la figura 1 podemos ver que existe una audibilidad en la parte plana de la gráfica ya que se está reproduciendo en el rango audible, pero en el momento que éste va decayendo es donde se produce un post-enmascaramiento (<50ms) y es en esa zona donde se insertará el eco ya que en esta parte un traslape del eco producido será imperceptible al oído humano.

Se sabe que la frecuencia de muestreo de los archivos de audio procesados es de 44100kHz, es decir, que se reproducen 44100 muestras en un segundo por lo tanto se debe calcular cuántas muestras hay en 50ms, que es el límite del enmascaramiento, y así realizar el retraso (eco) para insertar un “1” dado que para insertar un “0” la señal pasará sin modificación alguna. Abajo se muestran los cálculos efectuados y el número de muestras de retraso resultante:

Si 44100 muestras  $\longrightarrow$  1 segundo

“X” muestras  $\longrightarrow$  50 milisegundos

$$X = \frac{(50 \text{ ms})(44100 \text{ muestras})}{1 \text{ s}} = 2205 \text{ muestras}$$

El resultado obtenido indica que el retraso de la señal para incrustar un “1” debe ser máximo de 2250 muestras para que los datos incrustados (marca de agua) sean imperceptibles (inaudibles) al oído humano.

Las amplitudes de la señal original y de la señal de retraso son diferentes entre sí y lógicamente la amplitud de la señal de retraso debe ser menor que la amplitud de la señal original debido a que si se retrasa la original y no se modifica la amplitud, ésta sería audible y distorsionaría la sonoridad del audio original. En base a las anteriores hipótesis, la ecuación (9) queda definida a continuación y se convierte en la ecuación (10):

$$y(n) = \begin{cases} 1 * x(n); & w_i = 0 \\ 1 * x(n) + \frac{1}{3} * x(n - 2205); & w_i = 1 \end{cases} \quad (10)$$

Donde;

$\alpha = 1$ ; *amplitud de la señal original*

$\beta = \frac{1}{3}$ ; *amplitud de la señal de retraso*

### 2.1.3 Esquema del proceso de marcado

El esquema de marcado mediante ocultamiento por eco propuesto en este trabajo consiste en los siguientes pasos:

1. Establecer un umbral de inserción de la marca de agua

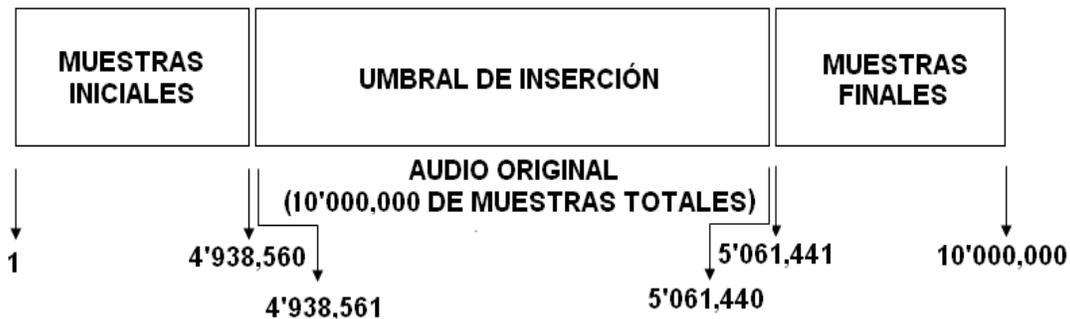


Fig. 2.3 Esquema general del umbral de inserción en audio

En la figura 2.3 puede verse que el umbral de inserción se encuentra en la parte media del audio original, esto es porque en los extremos de algunas canciones (principio o final) existen silencios que podrían evidenciar la marca de agua incrustada y para evitarlos se decidió incrustarla en las muestras centrales.

2. Dividir el umbral de inserción en bloques de tamaño de 1024 muestras cada uno, para posteriormente insertar un bit de marca de agua en cada bloque.

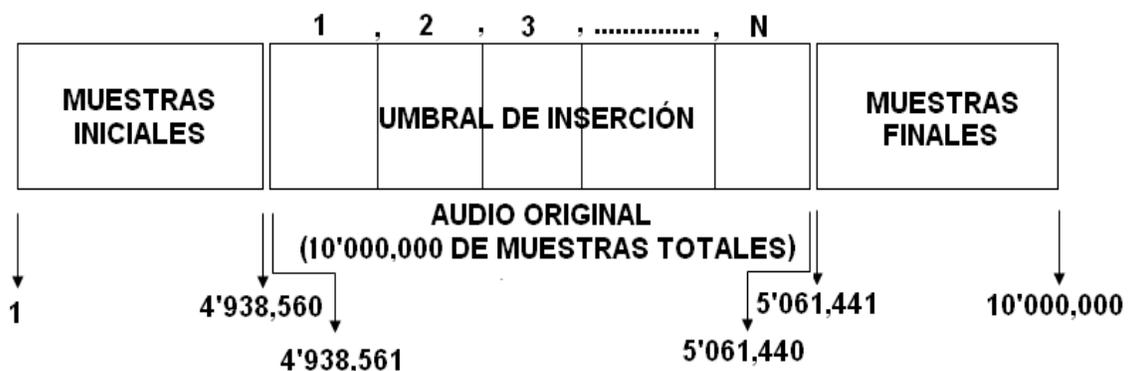


Fig. 2.4 Segmentación en bloques del umbral de inserción

El valor establecido para N es de 120 bits debido a que esa es la longitud de la marca de agua, por lo tanto, el umbral de inserción estará dividido en 120 bloques de longitud de 1024 muestras cada uno, en donde se insertará la marca de agua.

3. Se define la marca de agua y se sigue la siguiente regla de inserción

Para incrustar un “0”

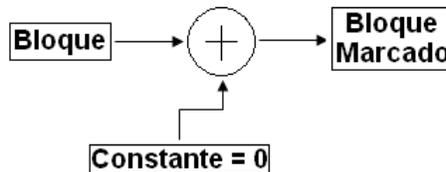


Fig. 2.5 Esquema de incrustación de un “0”

Para incrustar un “1”

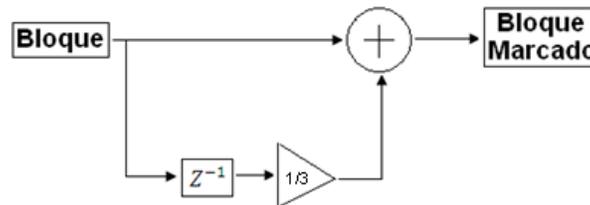


Fig. 2.6 Esquema de incrustación de un “1”

4. Por último se concatena los bloques procesados con la marca de agua, las muestras iniciales y las muestras finales para así generar el archivo de audio marcado.

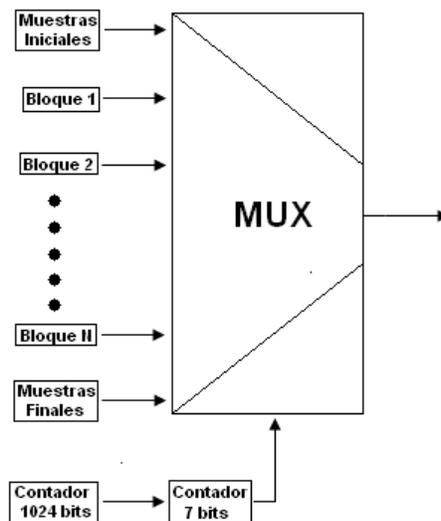


Fig. 2.7 Generación del archivo de audio marcado

Como se puede visualizar en la figura 2.7, los bloques marcados, las muestras iniciales y las muestras finales son procesadas mediante un multiplexor para que vaya permitiendo el paso de bloque por bloque en determinados tiempos y acomodando el contador que lo controla y a la salida obtener el archivo total pero con la marca incrustada a diferencia del archivo original.

## 2.2 Detección de la Marca de agua

Cuando se inserta una marca de agua en un archivo multimedia se necesita crear un sistema de detección de dicha marca de agua debido a que, dependiendo de la aplicación, se necesite detectar el mensaje oculto y verificar su integridad. El proceso de detección diseñado es ciego comparado con el sistema no ciego del algoritmo de ocultamiento por eco tradicional. A continuación se hace una comparación entre dichos sistemas de detección para el sistema de marcado propuesto:

Tabla 2.1 Requerimientos para la detección ciega y no ciega

REQUERIMIENTOS	
SISTEMA NO CIEGO	SISTEMA A CIEGAS
Audio Original	Audio Marcado
Audio Marcado	
Marca de Agua Incrustada	Umbral de Inserción
Umbral de Inserción	
Tamaño de Bloque	Tamaño de Bloque

Como puede verse en la Tabla 2.1 un sistema de detección no ciego necesita tener conocimiento previo del audio original (audio si la marca incrustada), del audio marcado (audio con la marca incrustada), la propia marca de agua, el umbral de inserción de la misma y el tamaño de los bloques de incrustación. A

continuación se muestra de manera detallada el proceso de detección y los parámetros iniciales necesarios para su funcionamiento:

### 2.2.1 Parámetros Iniciales

Para realizar la detección por el método propuesto se necesitan algunos parámetros iniciales, estos se definen a continuación:

- ✓ Audio Marcado
- ✓ Umbral de Inserción
- ✓ Tamaño de los bloques de inserción

### 2.2.2 Esquema del Proceso de Detección

La diferencia existente de este algoritmo de detección con los algoritmos de detección tradicionales radica en que no se necesita tener conocimiento previo del audio original ni de la marca de agua a incrustar

1. El umbral de inserción fue propuesto de manera genérica ya que la mayoría de las canciones tienen una duración entre tres y cuatro minutos de reproducción y como ya se había explicado en el proceso de inserción, para evitar los silencios se toma en la muestra inicial del umbral

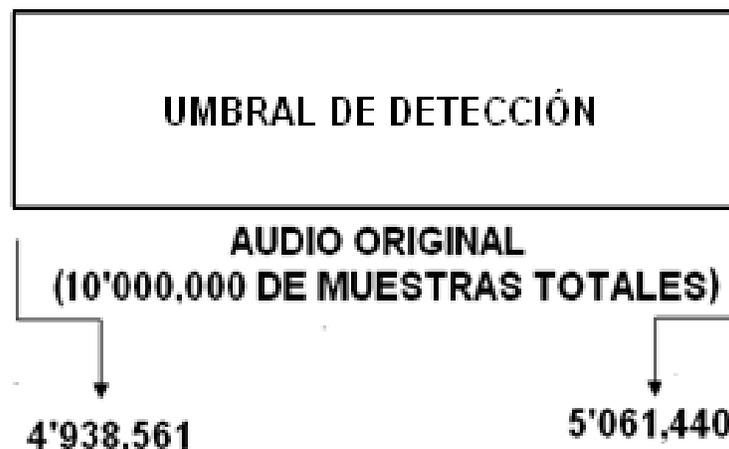


Fig. 2.8 Umbral de detección genérico

2. Se divide el umbral de inserción en bloques de 1024 muestras



Fig. 2.9 Umbral de detección genérico seccionado en bloques

3. Se calcula el Cepstrum de cada bloque

$$C(n) = \mathcal{F}^{-1}(\log(|\mathcal{F}(x(n))|)^2) \quad (11)$$

$$C(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log|X(e^{jw})| e^{jwn} dw \quad (12)$$

El análisis cepstral se utiliza en el procesamiento de señales no lineales e impredecibles para aplicaciones diversas, sobre todo en el área de procesamiento de imágenes, de voz o de audio. Puede decirse que el cepstrum es la transformada de inversa de Fourier del logaritmo de la potencia espectral de la señal. Se corresponde en el dominio temporal y su unidad es la cuefrenca. En este trabajo el análisis cepstral se usa para determinar las ondas fundamentales y sus respectivos ecos (si los tuviera) con sus amplitudes relativas.

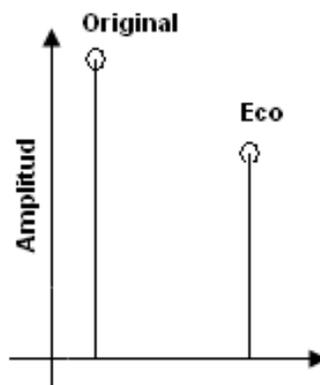


Fig. 2.10 Gráfica del cepstrum (original y eco)

Por consiguiente, cuando se realice el cálculo del cepstrum de cada bloque, si existe un eco, será porque un "1" fue incrustado, pero si la amplitud del eco es nula, será porque un "0" fue incrustado. Por último se genera la secuencia de bits obtenidos.

# Capítulo 3

## Implementación y Pruebas

### Introducción

En este capítulo se aborda la implementación de algunos algoritmos dedicados al marcado de agua digital en audio programados en Matlab y la implementación del algoritmo elegido (Ocultamiento por Eco) en dos plataformas: en Matlab y a manera de hardware, en un FPGA (Field Programmable Gate Array) con la finalidad de no depender de los recursos de una PC

### 3.1 Entorno Matlab

Antes de implementar el algoritmo de Ocultamiento por Eco en el FPGA, se programaron en la plataforma MatLab algunos algoritmos para verificar que funciona conforme a la teoría y los parámetros previamente establecidos.

MatLab es conocido como el “laboratorio de matrices”, es un software matemático que ofrece un entorno de desarrollo integrado (IDE). Es un lenguaje que tiene como prestaciones la manipulación de matrices, la representación de datos y funciones, la implementación de algoritmos, creación de interfaces gráficas de usuario (GUI) y quizá la herramienta más atractiva, la comunicación con programas en otros lenguajes y con dispositivos de hardware tales como, DSP (Digital Signal Processors), FPGA (Field Programmable Gate Array), Microcontroladores, etc. Además es un lenguaje que permite ampliar sus capacidades con las llamadas “cajas de herramientas (toolbox)” o con simulink con los “paquetes de bloques (blocksets)”.

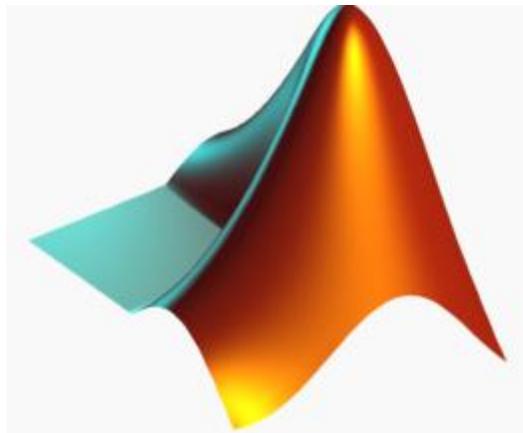


Fig. 3.1 Logo de MatLab

### 3.2 Implementación del algoritmo AM (Modulación de Amplitud)

En este apartado se desarrolla la implementación en la plataforma Matlab del algoritmo de Modulación de Amplitud, el cual ya fue explicado en el capítulo 2 y a continuación se muestra el algoritmo programado dividido en metodología de inserción y metodología de detección.

### 3.2.1 Metodología de Inserción de AM

En las siguientes líneas se muestra la metodología a seguir para construir el algoritmo implementado en Matlab:

1. Se hace la lectura del archivo de audio a marcar (audio original)

```
[y,fs1] = wavread('Sortilegio.wav');
```

2. Se establece el umbral de Inserción en bloques de 1024 muestras.

```
Inicio = 4938560;  
Final = 5061440;  
for Inicio:1024:Final
```

3. Se genera la marca de agua anotando el texto a ocultar compuesto por una secuencia de 0's y 1's [14] y con una semilla de inicialización del generador de números aleatorios.

```
Mrk=('JUAN ANTONIO RIOS CHAVEZ');  
semilla=3;  
marca=secuencia_binaria;
```

4. Se aplican las reglas de inserción para incrustar un "1" y un "0"

```
for K=1:block:(mIn*block)  
    if (marca(temp) == 1)  
        %Se aplica la siguiente regla de inserción  
        %.99 si es un '1' , .98 si es un '0'  
        m4(K:(K-1)+block) = .99*y(K:(K-1)+block);  
    else  
        m4(K:(K-1)+block) = .98*y(K:(K-1)+block);  
    end  
    temp = temp+1;  
end
```

6. Se genera el audio marcado

```
wavwrite(m4,fs1,'am_out.wav');
```

### 3.2.2 Metodología de Detección de AM

A continuación se muestran los pasos a seguir para detectar la marca de agua incrustada en el proceso de inserción:

1. Se lee el archivo de audio marcado

```
[m4,fs2] = wavread('am_out.wav');
```

2. Se establece el umbral de inserción en bloques de 1024 muestras

```
Inicio = 4938560;  
Final = 5061440;  
for Inicio:1024:Final
```

3. Se verifica la condición, si la amplitud es 0.98 o mayor se insertó un “1”, y en caso contrario, si la amplitud es menor a 0.98 se insertó un “0”

```
for K=1:blk:(mIn*blk)  
x(K:(K-1)+blk) = .98*x(K:(K-1)+blk);  
%for whatever reason, the statement below must be  
%greater than or equal in order to correctly identify  
%the watermark.  
if (abs(w(K:(K-1)+blk)) >= abs(x(K:(K-1)+blk)))  
wrec(L)=1  
else  
wrec(L)=0  
end  
end  
end
```

### 3.3 Implementación del algoritmo de Dos Conjuntos

En este apartado se desarrolla la implementación en la plataforma Matlab del algoritmo de Dos Conjuntos basado en la técnica de Patchwork Modificado, el cual ya fue explicado en el capítulo 2 y a continuación se muestra el algoritmo programado dividido en metodología de inserción y metodología de detección.

#### 3.3.1 Metodología de Inserción de Dos Conjuntos

Para incrustar la marca de agua por el método de Dos Conjuntos se siguieron los siguientes pasos:

1. Se da lectura del audio a marcar

```
[y,fs1] = wavread('Sortilegio.wav');
```

2. Se genera la marca de agua a partir de una semilla de inicialización

```
semilla=3;  
marca=randint(1,bits,[0,1],semilla)
```

3. Se divide el audio en bloques de 1024 muestras y se aplica el siguiente proceso a cada bloque:

```
x=dct(block);
indice=randint(1, (2*n), [k1,k2], semilla);
A=x(1, indice(1:n));
B=x(1, indice(n+1:(2*n)));
```

4. Se calculan los parámetros estadísticos a partir de los conjuntos A y B (Calcular la media y la desviación estándar)

```
d=sign(Amedia-Bmedia)*sqrt(C)*S/2);
```

5. Se crean los dos nuevos subconjuntos

```
A2=A+d;
B2=B-d;
```

6. Se reemplazan los anteriores conjuntos por los nuevos que ya se crearon

```
X2=x;
X2(1, indice(1:n))=A2;
X2(1, indice(n+1:2*n))=B2;
X2(1, n+1:N-n)=x(n+1:N-n);
Markedblock=idct(X2);
```

### 3.3.2 Metodología de Detección de Dos Conjuntos

La metodología de detección es enlistada en las siguientes líneas

1. Se da lectura del audio marcado

```
[m4, fs2] = wavread('am_out.wav');
```

2. Se divide el audio en bloques de 1024 muestras y se aplica el siguiente proceso a cada bloque y considerando cada semilla asociada:

```
for i=1:n_semillas
    x=dct(block);
    indice(i, :)=randint(1, (2*n), [k1,k2], semilla);
    A(i, :)=x(1, indice(1:n));
    B(i, :)=x(1, indice(n+1:(2*n)));
```

3. Se calculan los parámetros estadísticos a partir de los conjuntos A y B (Calcular la media y la desviación estándar para todas las semillas)

```
d=sign(Amedia-Bmedia)*sqrt(C)*S/2);
```

4. La semilla detectada será aquella en la que para el valor de umbral  $T^2$  sea máximo.
5. Por último se genera la marca asociada a la semilla detectada

### 3.4 Implementación del algoritmo de Ocultamiento por Eco

El algoritmo elegido para implementación en la interfaz gráfica y en el procesador FPGA fue el algoritmo de inserción mediante Ocultamiento por Eco, esto debido a su baja complejidad computacional, al fácil entendimiento de la metodología y a la detección a ciegas que puede ofrecer el algoritmo (la detección a ciegas es considerado como un proceso ideal debido a que no se necesita tener conocimiento previo de los parámetros de inserción ni del mensaje oculto).

#### 3.4.1 Metodología de Inserción para Ocultamiento por Eco

En las siguientes líneas se detallan los pasos a seguir para la implementación del algoritmo en Matlab:

7. Leemos el archivo musical con la instrucción “wavread” de MatLab

```
[y,fs1] = wavread('Sortilegio.wav');
```

8. Se establece el umbral de inserción en bloques de 1024 muestras

```
Inicio = 4938560;  
Final = 5061440;  
for Inicio:1024:Final
```

9. Se genera la marca de agua anotando el texto a ocultar compuesto por una secuencia de 0's y 1's [14] y con una semilla de inicialización del generador de números aleatorios

```
Mrk=('JUAN ANTONIO RIOS CHAVEZ');  
semilla=3;  
marca=secuencia_binaria;
```

10. Se genera una matriz para alojar el retraso que se producirá de la señal original

```
m5(1:441) = 0;
m5(442:m2) = y(1:(m2-441));
```

#### 11. Se aplican las reglas de inserción para un “0” y para un “1”

```
if (marca(temp) == 1)
    %Eco agregado para incrustar un '1'
    m4(K:(K-1)+block) = 0.333333*m5(K:(K-1)+block) + y(K:(K+block));
else
    %no se agrega eco para incrustar un '0'
    m4(K:(K-1)+block) = y(K:(K-1)+block);
end
temp = temp + 1;
end
```

#### 12. Se genera el audio marcado

```
wavwrite(m4, fs1, 'echo_out.wav');
```

### 3.4.2 Metodología de Detección para Ocultamiento por Eco

A continuación se detallan los pasos a seguir para la detección de la marca de agua:

#### 4. Se lee el archivo de audio marcado

```
[m4, fs2] = wavread('echo_out.wav');
```

#### 5. Se establece el umbral de inserción en bloques de 1024 muestras

```
Inicio = 4938560;
Final = 5061440;
for Inicio:1024:Final
```

#### 6. Se realiza el cálculo del cepstrum para cada bloque

```
L=1;
for block=Inicio:1024:Final

    a=fft(m4);
    b=abs(a).^2;
    c=log(b);
    d=ifft(c);
    e=abs(d).^2;
```

## 7. Se genera la marca de agua detectada

```
if e(block) >= 0.33
    w_rec(L)=1;
else
    w_rec(L)=0;
end
end
L=L+1;
```

El proceso de detección se aplicó a los audios marcados pero después de que fueron “atacados” por diferentes procesos digitales y así verificar la robustez de la marca de agua ante éstos.

Algunos centros de investigación como el instituto Fraunhofer [23] se han dedicado al estudio de marcado de agua digital y su implementación en software de aplicación. La franquicia más sobresaliente de este instituto es Music Trace [24] en la que se utiliza la marca de agua para servicios de valor añadido. En este trabajo se implementaron los procesos elegidos en una GUI (Guide User Interface), con plataforma en MatLab, para la facilidad de manejo del algoritmo elegido (Ocultamiento por Eco). La figura 3.2 muestra la interfaz programada:



Fig. 3.2 Interfaz gráfica del algoritmo elegido

Los pasos para el uso de la interfaz gráfica de usuario del esquema propuesto se enuncian a continuación:

La barra de menús contiene tres menús:

1. Algoritmo: Contiene cuatro artículos en formato PDF que explican el algoritmo de ocultamiento por eco, la robustez e inaudibilidad del mismo y algunos otros diseños propuestos por otros autores.
2. Instrucciones: Muestra un instructivo que explica la manera de usar la interfaz, que básicamente se constituye de los siguientes apartados:
  - El botón [1] y [5] sirve para seleccionar el archivo de audio, original y marcado respectivamente.
  - En el apartado [2] teclear la frase que se desea ocultar en el audio.
  - El botón [3] sirve para ocultar la frase tecleada en el archivo de audio original elegido.
  - La imagen marcada con [4] es el logo creado para el software generado.
  - Cuando se elige el archivo con la marca de agua incrustada, se necesita conocer cuál es el mensaje incrustado, para esto sirve el botón [6].
  - El apartado [7] muestra el mensaje extraído del audio marcado.
3. El menú power point contiene una presentación en formato .pptx que muestra una explicación visual del método de ocultamiento por eco.

### **3.5 Pruebas de Robustez**

Se realizaron pruebas de ataques hechos al audio marcado con la finalidad de medir la robustez [12] de la marca de agua. Estos ataques se hicieron mediante la herramienta MatLab y otra que es llamada Audio Stirmark.

Audio Stirmark [13] es una herramienta útil para evaluar la robustez de sistemas de marcado de agua digital en audio a través de varios “ataques” o procesos digitales realizados a los archivos marcados. Para verificar la robustez se mide el BER [15]. La base de datos de Audio Stirmark cuenta con diversos ataques que se aplican al audio marcado, a continuación se enlistan dichos procesos o ataques contenidos en la herramienta de evaluación y la descripción de cada uno de ellos en la Tabla 3.1:

**Tabla 3.1 Listado y descripción de los ataques**

<b>Ataque</b>	<b>Descripción</b>
AddBrumm	Agrega baja frecuencia para simular el impacto de una fuente de alimentación
AddDynNoise	Agrega un ruido blando dinámico que depende de la fuerza de la original
AddFFTNoise	Agrega ruido blanco en el dominio de la frecuencia
AddNoise	Agrega ruido blanco a la señal de audio
AddSinus	Agrega una señal senoidal al audio
Amplify	Cambia la fuerza de la señal de audio
BassBoost	Incrementa el bajo de la señal
BitChanger	Cambia aleatoriamente algunos bits de la señal
Compressor	Aumenta o decreta el volumen de pasajes tranquilos
CoySample	Copia partes de los valores muestreados
CutSamples	Remueve periódicos valores de muestra del audio
Exchange	Intercambia cada par de muestras consecutivas
ExtraStereo	Incrementa la parte stereo de la señal de audio
FFT_Invert	Invierte la parte real y la imaginaria de las muestras
FFT_RealReverse	Invierte sólo la parte real de la señal de audio
FlippSample	Intercambia muestras periódicamente dentro de la señal
Invert	Invierte todas las muestras de la señal, 180°
LSBZero	Cambia todos los bits menos significativos a "0"
Noise_Max	Agrega una máxima longitud de secuencia
Normalizer1	Normaliza la amplitud de un valor dado
Normalizer2	Igual que Normalizer1 pero independiente de altas o bajas frecuencias
Nothing	Simplemente es una derivación sencilla
PitchScale	Escala el tono sin cambiar el tiempo
RC_HighPass	Simula un filtro pasa altas RC
RC_LowPass	Simula un filtro pasa bajas RC
ReplaceSamples	Cambia muestras por valores aleatorios
Resampling	Cambia la tasa de muestreo de la señal
Smooth	Suaviza las muestras
Smooth2	Las muestras vecinas son modificadas en anchura
Stat1	Promedios de las muestras con sus vecinos próximos
Stat2	Promedios de las muestras con sus vecinos próximos y anteriores
TimeStretch	Extiende el tiempo sin afectar la señal
VoiceRemove	Opuesto de ExtraStereo
ZeroCross	Utilizado como un limitador.
Zerolength1	Si el valor de las muestras es "0", este inserta mas muestras "0" para cada canal
Zerolength2	Si la muestra de algún canal es "0", en esos valores se inserta un número determinado de muestras

Además de las pruebas de robustez, se debe verificar que la marca de agua sea inaudible al oído humano y que la calidad sonora sea lo menos deteriorada posible. El siguiente apartado muestra los estándares en los que fueron basados dichas pruebas.

### 3.6 Inaudibilidad de la marca de agua

La marca de agua incrustada debe ser imperceptible (inaudible) al oído humano y por consiguiente, no afectar la calidad del audio que la está alojando, para esto se hicieron pruebas bajo estándares y se realizó una Evaluación Perceptual de la Calidad del Audio (PEAQ por sus siglas en inglés) [16-17] mediante pruebas MOS (Mean Opinion Score) y pruebas de ODG (Objective Difference Grade), el primero es una prueba subjetiva que refleja la calificación de las personas a la calidad del audio marcado según su percepción y el segundo es el cálculo de un coeficiente que indica el grado de deterioro que sufre la señal al incrustarle la marca de agua. Los estándares anteriores se rigen bajo las siguientes tablas:

Tabla 3.2 Puntuación Media de Opinión

<b>Grado MOS (Mean Opinion Score)</b>	<b>Descripción del deterioro</b>
5	Imperceptible
4	Perceptible, pero no molesto
3	Ligeramente molesto
2	Molesto
1	Muy molesto

Tabla 3.3 Grado de Diferencia Objetiva

<b>ODG (Objective Difference Grade)</b>	<b>Descripción Del deterioro</b>
0.0	Imperceptible
-1.0	Perceptible, pero no molesto
-2.0	Ligeramente molesto
-3.0	Molesto
-4.0	Muy molesto

### 3.7 Arquitectura del FPGA

Las FPGA [18] son dispositivos semiconductores que contienen bloques de lógica cuya conexión y funcionalidad pueden ser programadas tantas veces como el usuario/desarrollador quiera. Su ámbito de aplicación es el mismo que el de los ASIC (Application Specific Integrated Circuit). Sus ventajas son una gran versatilidad y su bajo precio por unidad y al desarrollar sistemas, por lo que hoy en día son ampliamente utilizadas.

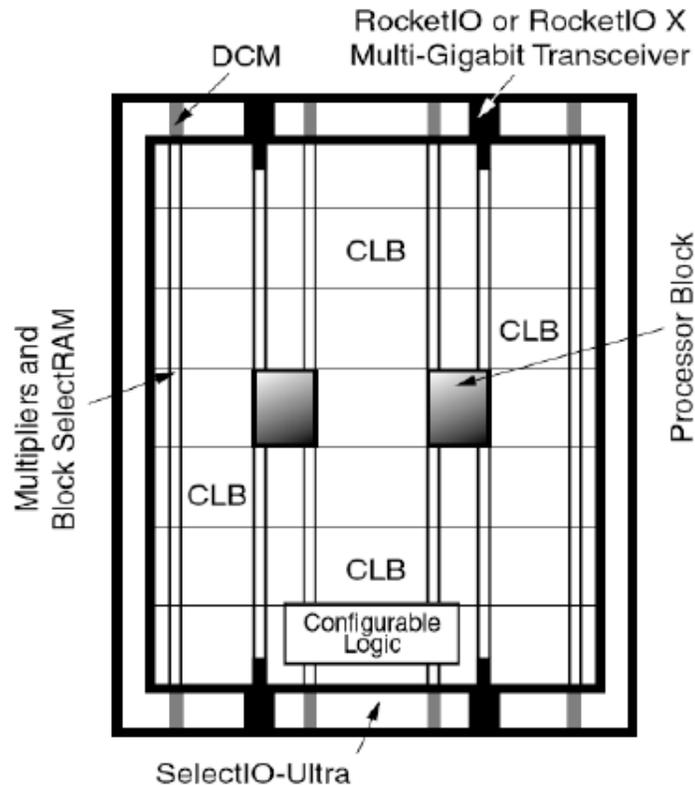


Fig. 3.3 Arquitectura del FPGA Virtex II Pro de Xilinx

En la figura 3.3 podemos ver un esquema de la arquitectura de la FPGA con la que contamos en nuestro trabajo de investigación, la Virtex-II Pro de Xilinx, que hoy en día es el principal fabricante de FPGA a nivel mundial. Abajo se describen los componentes del procesador:

- ✓ CLB: Configurable Logic Blocks. Son los bloques que llevan a cabo la carga funcional del sistema. Constan en nuestro caso de 4 slices cada uno, y son, por supuesto, programables en casi todos sus parámetros. Cada uno de los slices está compuesto por dos módulos funcionales básicos implementables en forma de LUT (LookUpTable) de 4 entradas, en forma de registro de desplazamiento de 16 bits o de memoria SelectRAM distribuída también de 16 bits. Los dos elementos de memoria son implementados según la

decisión del usuario, pudiendo ser utilizados como latches activados por nivel o flip-flops por flanco, ambos de diversa índole. Además, cada slice cuenta con una gran cantidad de lógica de control y acarreo para llevar a cabo funciones más complejas o con mayor número de entradas.

- ✓ IOB: In/Out Blocas. Esta FPGA cuenta con varios tipos de módulos de E/S. Aunque cambian sus especificaciones, su función es la de proporcionar una interfaz entre el exterior del dispositivo y la red de interconexión interna.
- ✓ DCM: Digital Clock Managers. Módulo que gestiona las señales de reloj, de modo que lleguen a la vez a todos los puntos del dispositivo y que a su vez puede generar otras a partir de las recibidas.
- ✓ POWERPC 405E: Power PC Hard Core Processor 405E de IBM. Cuenta con dos procesadores empotrados dentro del dispositivo, por lo que resulta apto para el desarrollo de sistemas empotrados basados en co-diseños HW-SW. Son procesadores RISC (Reduced Instruction Set Architecture) de 32 bits que alcanzan una frecuencia de trabajo máxima de 400MHz. Su diseño se basa en la arquitectura Power de IBM, con una segmentación en 5 etapas y con una memoria caché de primer nivel de 16KB separada para instrucciones y datos.
- ✓ BRAM: Block Random Access Memory. Incorpora a su vez bloques de 18K de RAM distribuida por columnas entre el resto de elementos. El FPGA con código XCV2VP30, dispositivo con el que contamos, implementa 136 bloques de memoria RAM dedicada de 18Kb, es decir, 2448KB distribuidos por todo el dispositivo. En caso de necesitar más memoria que la ofertada por uno de esos componentes, es fácilmente ampliable si se comunican varios módulos en cascada.
- ✓ Multiplicadores de 18 bits: Se trata de bloques dedicados capaces de multiplicar dos números de 18 bits con signo de forma rápida y con bajo consumo si se compara con un multiplicador similar sintetizado en slices. Pueden usarse independientemente o junto a bloques de SelectRAM con los que se sincronizan perfectamente.
- ✓ Red de Interconexión: Los recursos locales y globales de enrutado de la Virtex-II están optimizados para conseguir la máxima velocidad de transferencia posible sin comprometer para ello la flexibilidad. Es por ello que la arquitectura de la red de interconexión local se distribuye junto al resto de los elementos de la placa, a través de switches idéntico. Globalmente, estos switches quedan unidos mediante los canales globales de enrutado desplegados en la placa. Dependiendo de las necesidades de conexión entre dos elementos dados, se elegirá el mejor camino entre la líneas existentes, ya sean horizontales o verticales, largas, hex, directas o dobles y con mejores o peores prestaciones.

El FPGA basa su funcionamiento en la cadena de configuración o bitstream, cargada por el usuario. Ésta es almacenada en la memoria SRAM que incorpora para tal fin. Esta cadena binaria de configuración es la que denota las funciones, enrutamiento y demás información necesaria.

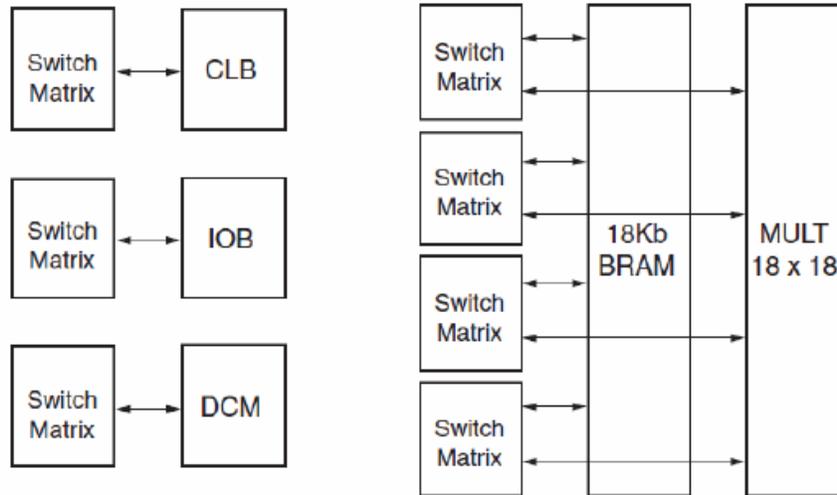


Fig. 3.4 Tecnología activa de interconexión

### 3.8 Implementación del algoritmo en bloques de Xilinx

Para la implementación del algoritmo se realizó un diseño mediante bloques de simulink y utilizando las librerías exclusivas de Xilinx System Generator (XSG)[19]. Ésta es una herramienta de MatLab que permite utilizar núcleos de códigos e implementaciones de funciones orientadas al procesamiento digital de señales en FPGA que pueden ser manejadas en Simulink. Los bloques que se necesitan para realizar el sistema de implementación se describen enseguida:

El bloque “counter” implementa un libre corrimiento o un contador limitado ascendente o descendente. La salida del contador puede ser expresada con un número de punto fijo con signo o sin él. La figura 3.5 muestra el bloque que se encuentra en las librerías: Basic Elements, Control Logic, Math e Index. En este bloque podemos establecer el inicio del contador, el final, la frecuencia de muestreo a la que se está trabajando, el periodo, la precisión de la señal de salida, etc.



Fig. 3.5 Bloque “counter”

A continuación se muestran los parámetros utilizados en la implementación para el bloque “counter”

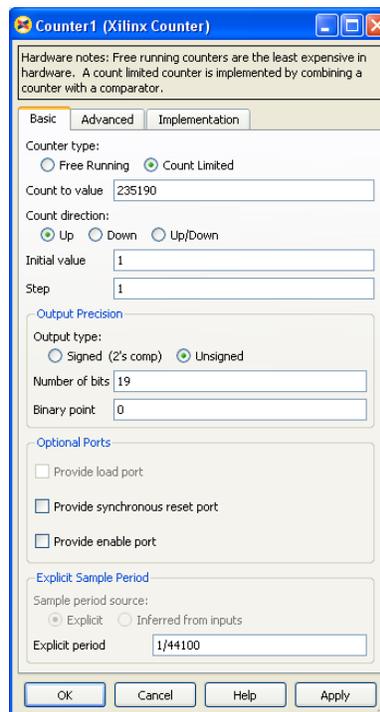


Fig. 3.6 Parámetros establecidos para la implementación del Bloque “counter”

Como puede verse en la figura 3.6 se define un contador limitado hacia un valor de número de muestras, estableciendo un valor inicial y un número de saltos, además se define el número de bits para la precisión de la salida y el periodo al que se está trabajando el sistema. Estos parámetros cambian dependiendo del bloque en el que estemos trabajando, ya que como se mencionó antes, la inserción de la marca de agua se realiza mediante bloques de 1024 muestras cada uno, a excepción del bloque de muestras iniciales y del bloque de muestras finales.

El bloque ROM se utiliza para leer los datos de alguna variable y los guarda para su posterior manipulación. El bloque se encuentra en las siguientes librerías: Control Logic, Memory e Index.

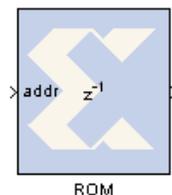


Fig. 3.7 Bloque “ROM”

Los parámetros de adaptación del bloque ROM se muestran en la siguiente figura:

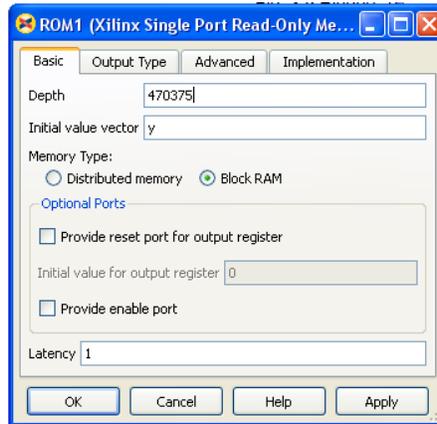


Fig. 3.8 Parámetros del bloque “ROM”

En el bloque ROM tecleamos la profundidad (Longitud total del audio que estamos trabajando) y se le asigna el nombre de la variable que hace la lectura del audio en el archivo \*.m

El bloque “Delay” implementa un retraso fijo de L ciclos. Las librerías donde se encuentra el bloque son: Basic Elements, Memory e Index.

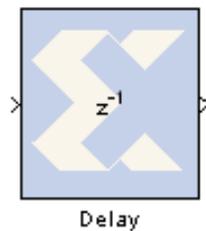


Fig. 3.9 Bloque “Delay”

El bloque “Delay” contiene la siguiente lista de parámetros:

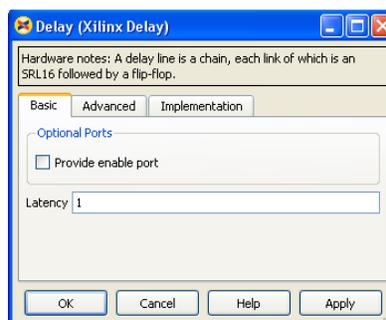


Fig. 3.10 Parámetros a modificar para el bloque “Delay”

El bloque “Constant” genera una constante que puede ser un valor de punto fijo, un valor booleano, o una instrucción DSP48.



Fig. 3.11 Bloque “Constant”

El bloque “Constant” contiene la siguiente lista de parámetros:

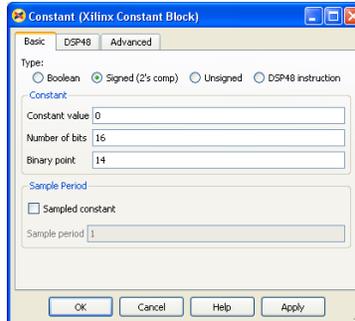


Fig. 3.12 Tabla de parámetros del bloque “Constant”

En esta ventana mostrada en la fig. 3.9 se elije el valor de la constante a tratar y el tipo de variable que deseamos a la salida.

El bloque “Relational” implementa un comparador de las entradas “a” y “b”

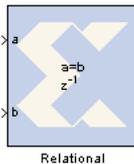


Fig. 3.13 Bloque “Relational”

Los parámetros del bloque “Relational” consisten en modificar la operación de comparación que se desea implementar, tal y como se muestra en la figura siguiente:

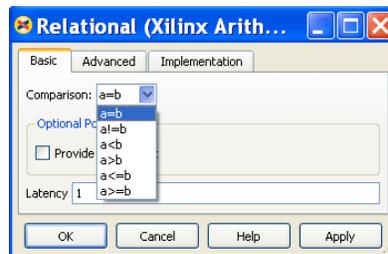


Fig. 3.14 Parámetros del bloque “Relational”

El bloque “CMult” implementa un operador de ganancia, en la que la salida es igual al producto de la entrada por un valor de constante.

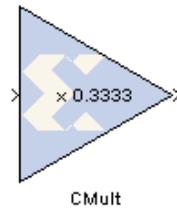


Fig. 3.15 Bloque “CMult”

La implementación del bloque consiste en multiplicar la señal de entrada por una ganancia para amplificar o reducir la amplitud de esa señal.

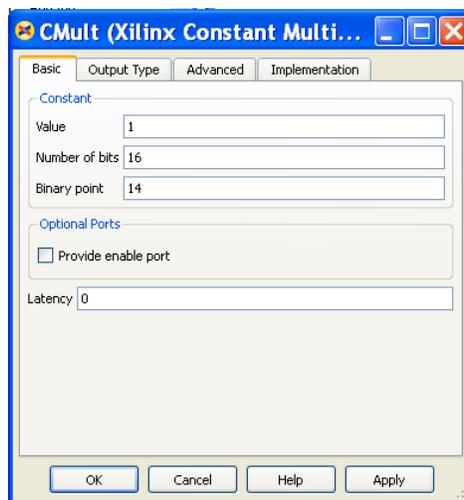


Fig. 3.16 Ventana de parámetros del bloque “CMult”

El bloque “AddSub” implementa un sumador o un substractor dependiendo la aplicación. Puede ser cambiada la operación de manera manual o dinámicamente.

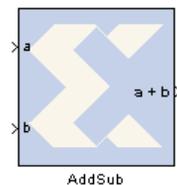


Fig. 3.17 Bloque “AddSub”

En la ventana de parámetros del bloque “AddSub” podemos elegir la opción a implementar, una suma, una resta o ambas.

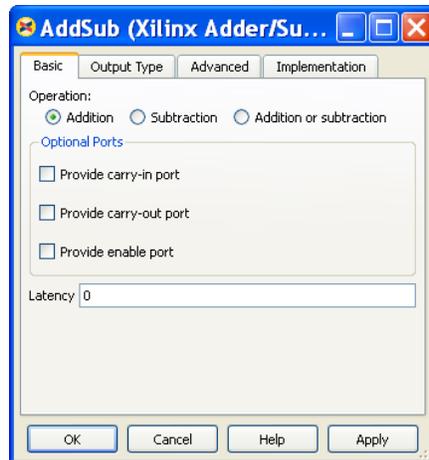


Fig. 3.18 Ventana de parámetros del bloque “AddSub”

El bloque “Mux” implementa un multiplexor. El bloque tiene una entrada selectora y un número de entradas configurables por el usuario con rango desde 2 hasta 1024.

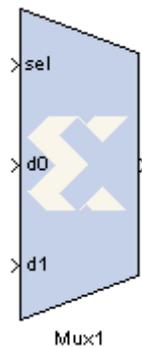


Fig. 3.19 Bloque “Mux”

El multiplexor se define constatando el número de entradas que queremos “switchear”



Fig. 3.20 Ventana de parámetros del bloque “Mux”

El bloque “White Gaussian Noise Generator” genera números aleatorios en base a la teoría de ruido blanco gaussiano.

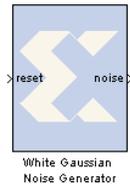


Fig. 3.21 Bloque “White Gaussian Noise Generator”

El bloque se implementa anotando una semilla de inicialización del generador de número aleatorios.

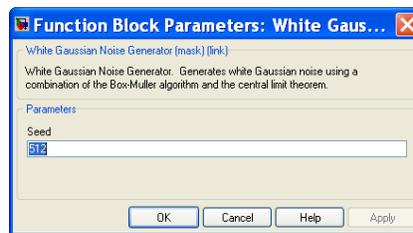


Fig. 3.22 Ventana de parámetros del bloque “White Gaussian Noise Generator”

El bloque “reinterpret” somete a la salida a algún otro tipo de dato. El número de bits de entrada siempre será el mismo que a la salida.

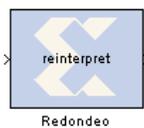


Fig. 3.23 Bloque “Reinterpret”

Para este bloque se necesita elegir el tipo de número al cual se quiere transformar la entrada.

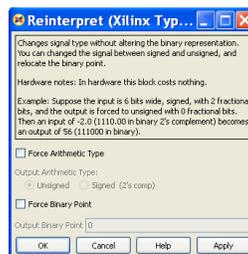


Fig. 3.24 Ventana de parámetros del bloque “Reinterpret”

Para la implementación del algoritmo en bloques de simulink, es necesario diseñar las condiciones de inserción. Las siguientes figuras muestran la implementación de las ecuaciones mencionadas en el capítulo 2

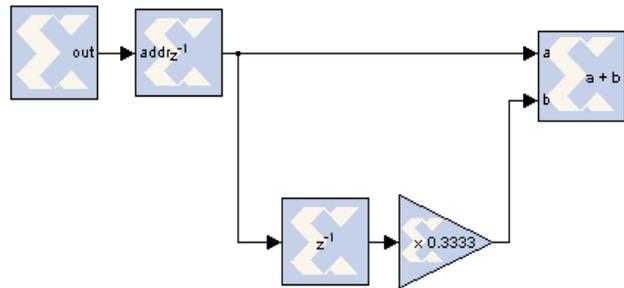


Fig. 3.25 Diagrama de Incrustación de un "1"

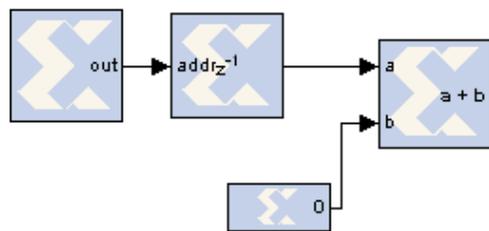


Fig. 3.26 Diagrama de Incrustación de un "0"

Para verificar que el arreglo propuesto para la incrustación de un "1" produce un eco (retraso) se visualizó en un osciloscopio las señales de salida y se muestran en la figura 3.27

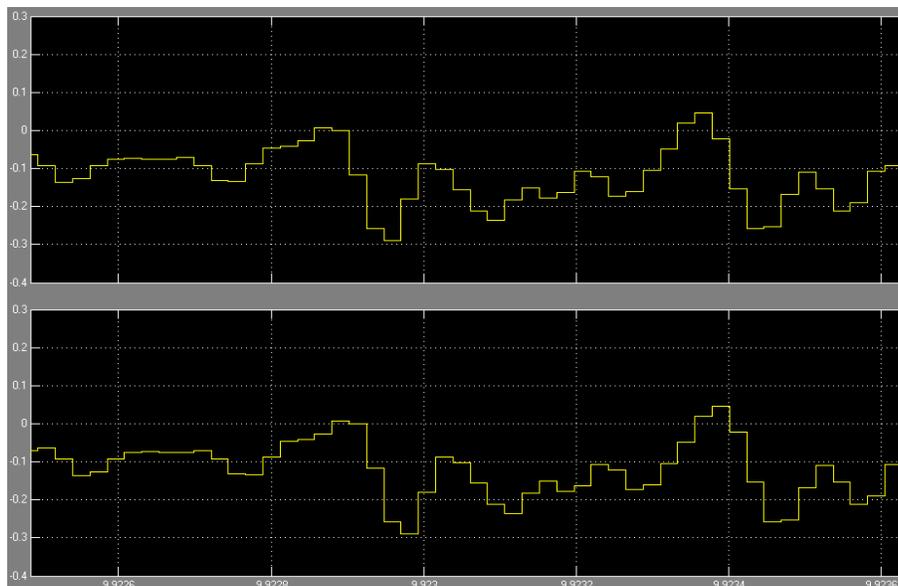


Fig. 3.27 Señal original y su correspondiente retraso

### 3.8.1 Metodología de Inserción en Xilinx

Después necesitamos generar la marca de agua a incrustar, para esto se diseñó el esquema de la figura 3.28 que consiste en generar una secuencia aleatoria de 120 bits de marca de agua. El diseño basa su funcionamiento en un generador de ruido blanco en el que se toman ciertas decisiones para generar ceros y generar unos dependiendo de lo que se obtenga del generador.

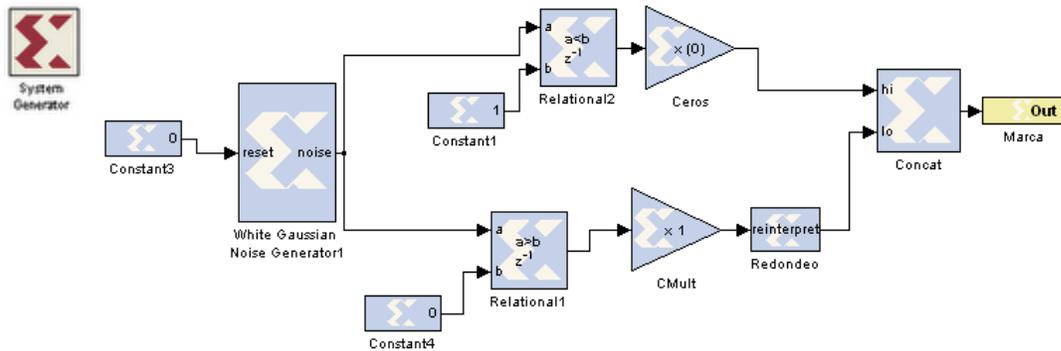


Fig. 3.28 Generador de secuencia binaria aleatoria (Marca de Agua)

La figura 3.29 muestra el esquema de incrustación de cero o uno para un bloque de muestras que depende de la entrada que tenga ya sea un "1" o un "0" implementado en bloques de System generator. Cabe mencionar que el esquema general consta de la incrustación de 120 bits (que no es mostrado debido a que el esquema es demasiado grande en cuanto a tamaño). Su funcionamiento consiste en seccionar en bloques de 1024 muestras y someterlos a las reglas de inserción, si es un "1" existe una adición de un retraso del mismo al bloque original, caso contrario, si es un "0" el bloque queda totalmente intacto. A la salida del sistema general, se van "switcheando" los bloques mediante un multiplexor controlado por un contador direccionador para que se vayan colocando en el tiempo que le corresponda a cada uno y así generar el audio marcado.

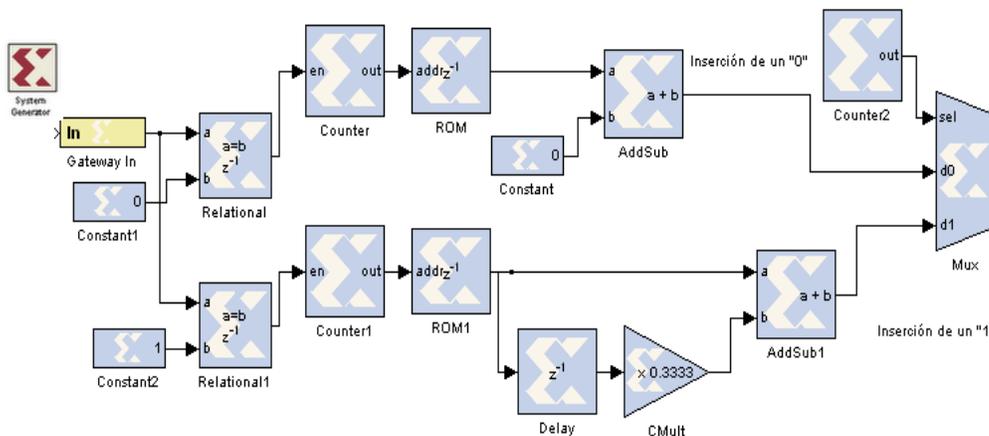


Fig. 3.29 Diagrama en bloques de System Generator para incrustar 1 bit ("1" o "0")

### 3.8.2 Metodología de Detección en Xilinx

Para el proceso de detección implementado en el FPGA se hizo un proceso no ciego ya que para el cálculo del cepstrum hay una complejidad mayor y no es óptimo implementarlo sofisticadamente debido a que si se incrusta la marca de agua por algún otro método, el proceso de detección se inutiliza. La figura 3.30 muestra el esquema de implementación:

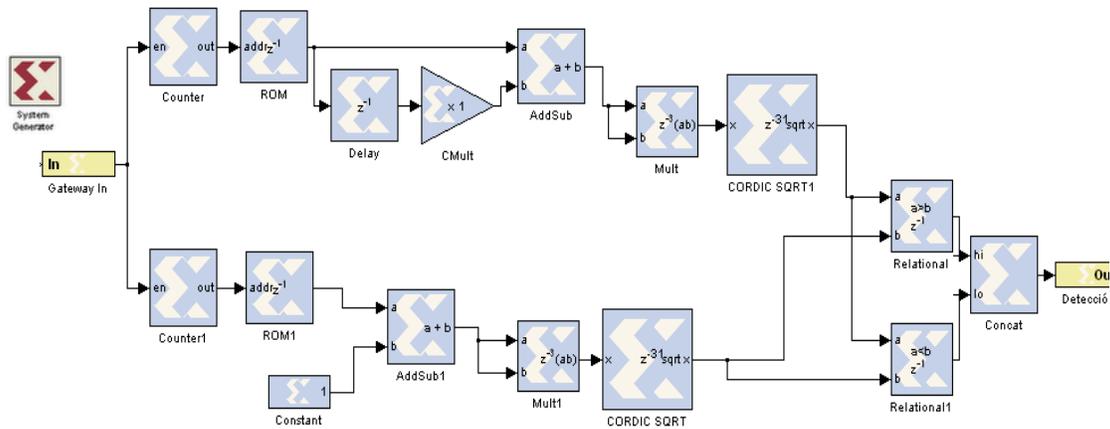


Fig. 3.30 Diagrama en bloques de System Generator para detectar la marca de agua

### 3.8.3 Implementación en el FPGA Virtex II pro de Xilinx

Para programar el sistema en el FPGA se necesitan establecer los parámetros necesarios mediante el software ISE, éste nos ayudará a generar el archivo con extensión \*.bit el cuál será cargado al FPGA. ISE posee una interfaz gráfica de usuario GUI (Graphic User Interface) que se denomina Project Navigator. Para el inicio de un nuevo proyecto, se tiene la etapa de diseño. En esta etapa lo que se hace es seleccionar el tipo de descripción de diseño. Los diseños se pueden introducir en diferentes formatos. Los más utilizados son: los esquemáticos, los diagramas de estados y las descripciones hardware en VHDL o Verilog.

Posteriormente en la síntesis se compila el diseño y se crea la netlist, una vez compilado, se puede simular a nivel funcional. Después de esta simulación, mediante las herramientas de implementación como la traducción, el mapeo y el ruteo, se permite la especificación de restricciones o indicaciones, a través del archivo \*.cfg, para realizar una implementación óptima en el FPGA. Así se puede realizar una simulación temporal, de esta forma se puede estimar cómo se comportará el sistema en el FPGA. La simulación a nivel funcional no tiene en cuenta los retardos provocados por el hardware y a nivel temporal simula el diseño

teniendo en cuenta la configuración en hardware. Finalmente se crea el fichero de programación \*.bit y se descarga el diseño sobre el dispositivo.

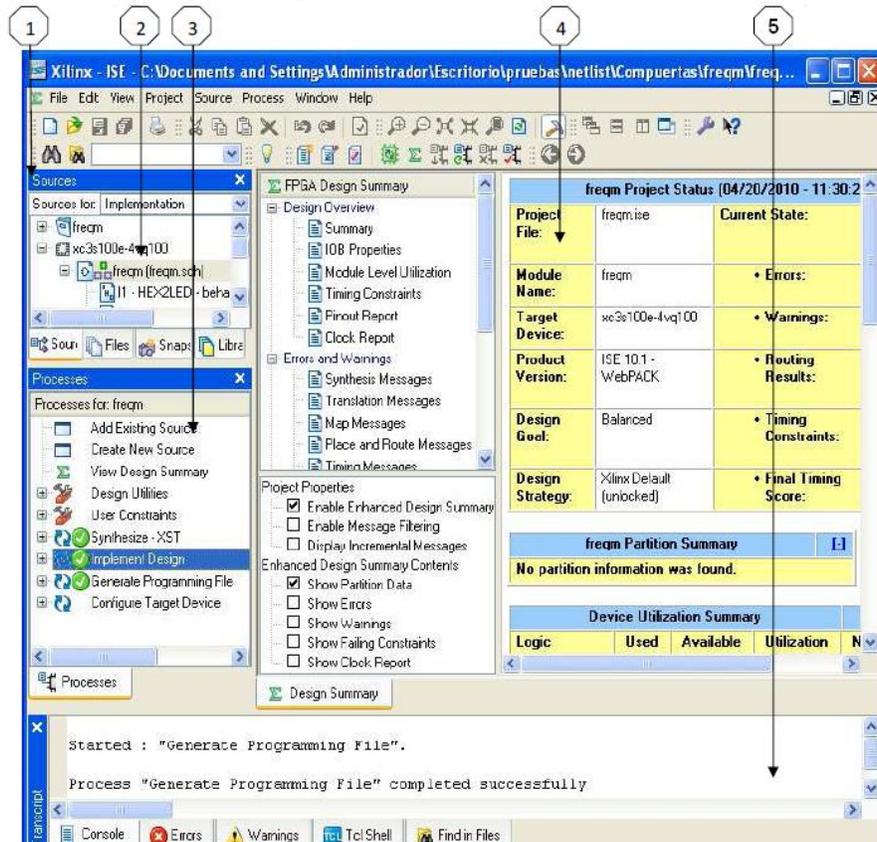


Fig. 3.31 Project Navigator

En la Figura 3.31 se muestra la interfaz de Project Navigator. En el punto número 1 se puede observar la barra de herramientas. En el punto número 2 se puede apreciar la ventana de fuentes, en la que se ve los archivos fuente que constituyen el proyecto. El punto 3, se indica la ventana de procesos. Estos procesos pueden ser, síntesis, simulación, implementación entre otros. En el punto 4, se representa el área de trabajo, esta área puede ser de tipo esquemático y también VHDL, dependiendo del tipo de descripción del proyecto seleccionado. Finalmente el punto 5, es la ventana de transcripción, en donde se puede ver el estado de las operaciones ya procesadas y las que se están procesando.

El siguiente capítulo muestra los resultados de las implementaciones realizadas, tanto en MatLab como en el FPGA.

# Capítulo 4

## Resultados Obtenidos

### Introducción

En este capítulo se muestran los resultados obtenidos de las implementaciones realizadas, tanto de las pruebas de inserción, detección robustez e inaudibilidad

## 4.1 Resultados Obtenidos de la Implementación en MATLAB

En las siguientes figuras se mostrarán los resultados y gráficos obtenidos de la implementación en MATLAB para cada uno de los algoritmos programados y en concreto para el algoritmo seleccionado.

### 4.1.1 Resultados para el algoritmo AM

A continuación se muestran las gráficas a manera de comparación visual del audio marcado por modulación de amplitud respecto del audio original:

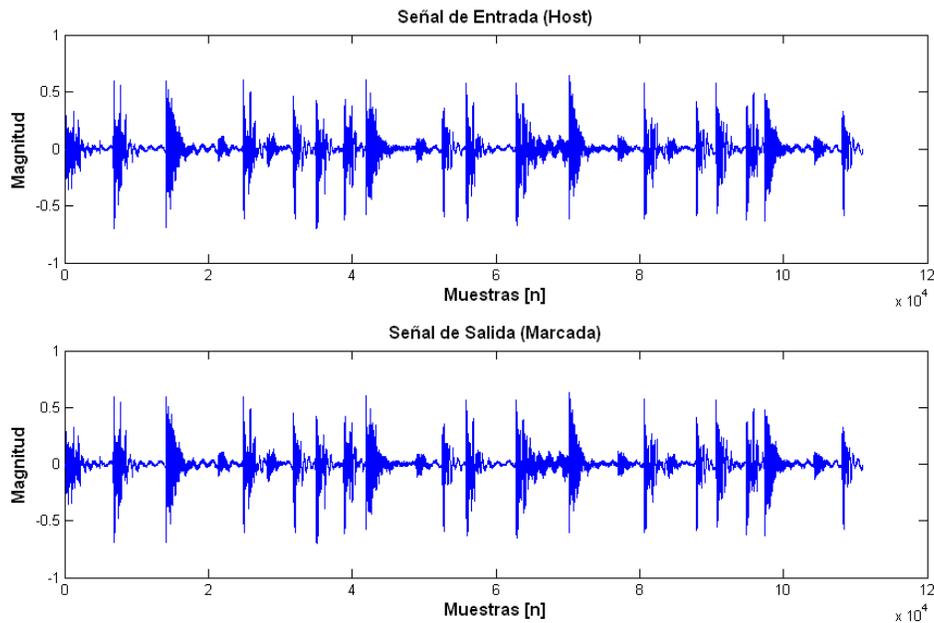


Fig. 4.1 Señal de entrada y señal marcada para AM

Visualizando la gráfica anterior, puede decirse que no hay una distorsión considerable del archivo de audio marcado, pero para corroborar esta aseveración se calcula la correlación de las dos señales obtenidas (original y marcada):

$$r = \begin{bmatrix} 1.0000000000000000 & 0.999959945236314 \\ 0.999959945236314 & 1.0000000000000000 \end{bmatrix}$$

Fig. 4.2 Correlación entre la señal original y la señal marcada

Se obtuvo una matriz en donde podemos visualizar que es próxima a la matriz identidad y que el coeficiente de relación (o cifra de mérito) es de 0.9999 cifra que

sin duda comprueba lo obtenido en las gráficas anteriores y que indica la similitud entre ambas señales. Cabe mencionar que este coeficiente es una cifra que varía entre 0 y 1 siendo que entre más próximo se encuentre a 1 existe una correlación mayor entre ambas señales.

Para cotejar lo anterior se obtuvo el espectrograma de ambas señales como se muestra en la figura 4.3

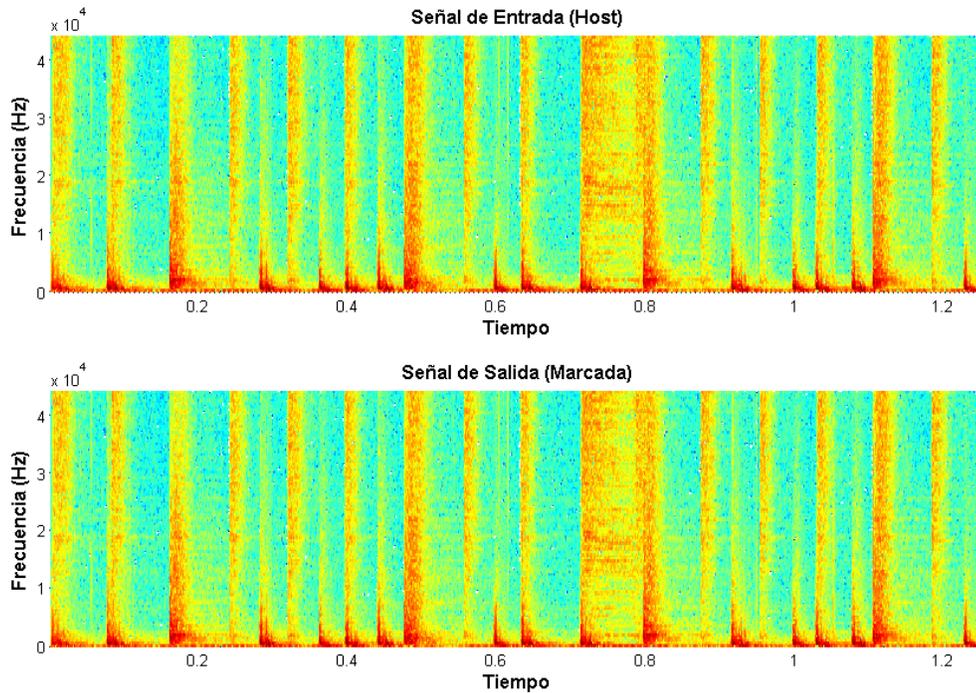


Fig 4.3 Espectro de la señal de entrada y la marcada para AM

La siguiente tabla muestra algunos ataques a los que se sometió el archivo marcado, cuáles fueron soportados y cuáles no lo fueron.

Tabla 4.1 Algunos ataques soportados y no soportados para el algoritmo AM

<b>Modulación de Amplitud</b>	
<b>Robusta</b>	<b>Frágil</b>
Remuestreo	AWGN
Nothing	AddDynNoise
Amplify	BitChanger
Invert	Compressor
ExtraStereo	CopySample
TimeStrech	CutSamples
	LSBZero
	RC_HighPass
	RC_LowPass

### 4.1.2 Resultados para el algoritmo de Dos Conjuntos

A continuación se muestran las gráficas a manera de comparación visual del audio marcado por Dos Conjuntos respecto del audio original:

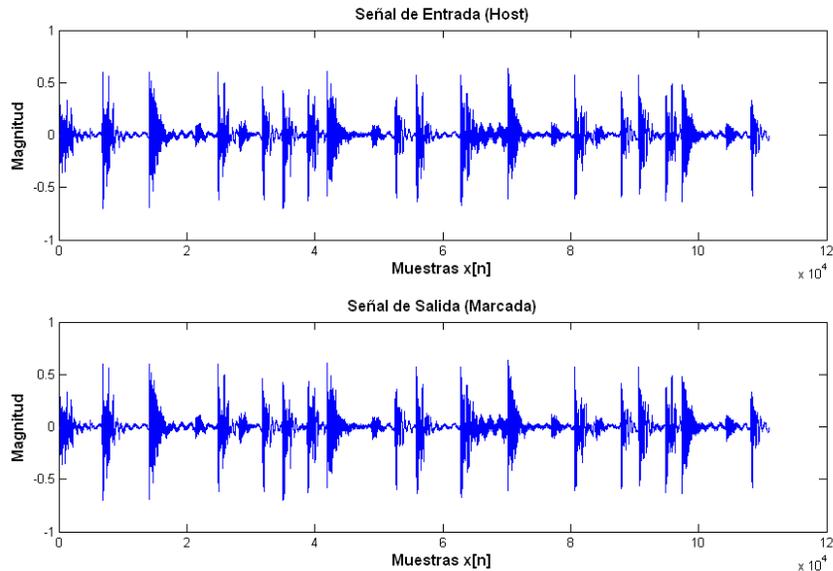


Fig. 4.4 Señal de entrada y señal marcada para Dos Conjuntos

Con la figura 4.4 puede verse que la diferencia entre ambas señales (original y marcada) es relativamente nula, para verificar esto se realizó el cálculo del coeficiente de correlación obteniendo lo siguiente:

$$r = \begin{bmatrix} 1.0000000000000000 & 0.999999865740178 \\ 0.999999865740178 & 1.0000000000000000 \end{bmatrix}$$

Fig. 4.5 Correlación entre la señal original y la señal marcada

Se obtuvo una matriz en donde podemos visualizar que es próxima a la matriz identidad y que el coeficiente de relación (o cifra de mérito) es de 0.9999 cifra que sin duda comprueba lo obtenido en las gráficas anteriores y que indica la similitud entre ambas señales. Cabe mencionar que este coeficiente es una cifra que varía entre 0 y 1 siendo que entre más próximo se encuentre a 1 existe una correlación mayor entre ambas señales.

Para cotejar lo anterior se obtuvo el espectrograma de ambas señales como se muestra en la figura 4.6

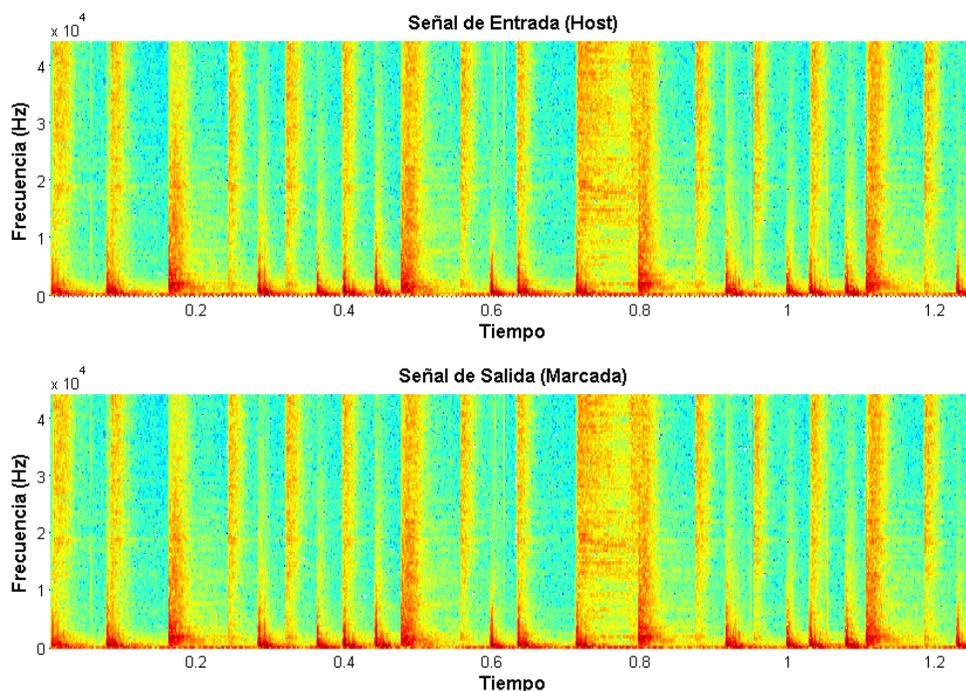


Fig 4.6 Espectro de la señal de entrada y la marcada para Dos Conjuntos

La siguiente tabla muestra algunos de los ataques a los que es robusta la marca de agua incrustada por el método de Dos Conjuntos y a cuáles ataques es frágil:

Tabla 4.2 Algunos ataques soportados y no soportados para el algoritmo de Dos Conjuntos[19]

Algoritmo de Dos Conjuntos	
Robusta	Frágil
RC_Highpass	Cutsamples
RC_Lowpass	ZeroCross
Invert	ZeroLenght
Compressor	ZeroRemove
Amplify	Flipsample
Addsinus	Copysample
Nothing	FFT_Stat1
Extrastereo	FFT_test

### 4.1.3 Resultados para el algoritmo de Ocultamiento por Eco

En las siguientes figuras se mostrarán los resultados y gráficos obtenidos de la implementación del algoritmo de Ocultamiento por Eco en MATLAB.

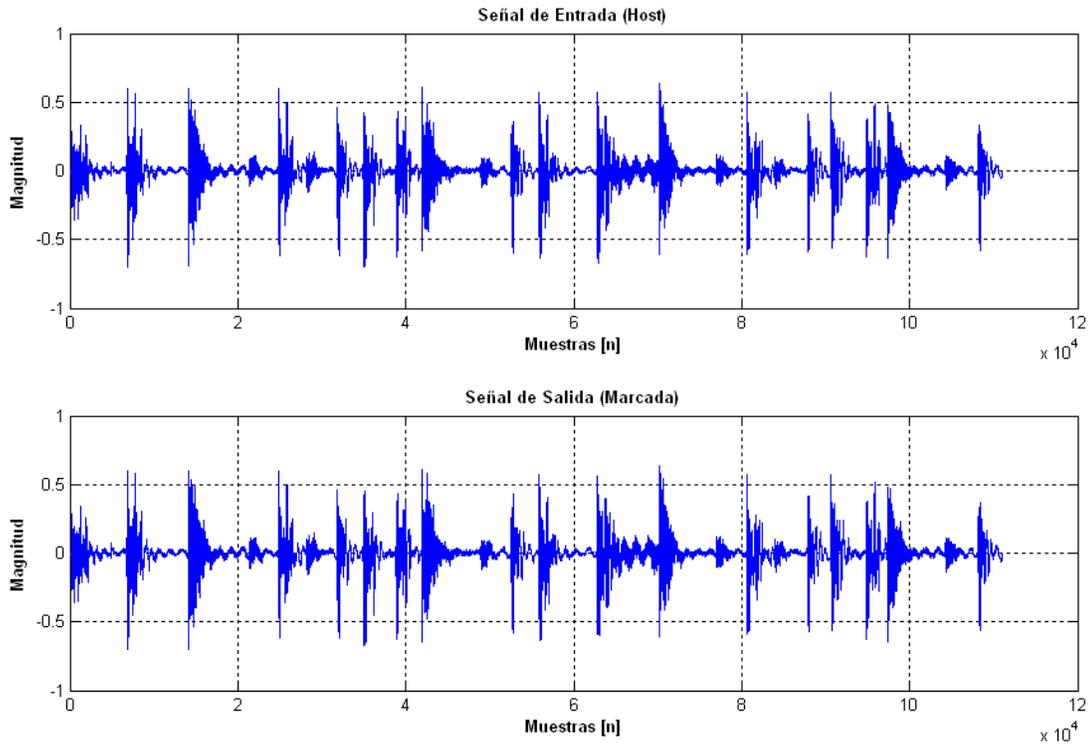


Fig. 4.7 Señal de entrada y señal marcada para Ocultamiento por Eco

Se puede visualizar que los retrasos agregados a la señal marcada para incrustar un “1” no afectan considerablemente la señal original debido al control de ganancia de amplitud adaptado al eco. Para comprobar esta situación se hizo el cálculo de la correlación de ambas señales y abajo se muestra lo obtenido:

$r =$

```
1.0000000000000000    0.984171360839425
0.984171360839425    1.0000000000000000
```

Fig. 4.8 Correlación entre la señal marcada y la señal original

Se obtuvo una matriz en donde podemos visualizar que es próxima a la matriz identidad y que el coeficiente de relación (o cifra de mérito) es de 0.9841 cifra que sin duda comprueba lo obtenido en las gráficas anteriores y que indica la similitud

entre ambas señales. Cabe mencionar que este coeficiente es una cifra que varía entre 0 y 1 siendo que entre más próximo se encuentre a 1 existe una correlación mayor entre ambas señales.

Para cotejar lo anterior se obtuvo el espectrograma de ambas señales como se muestra en la figura 4.9

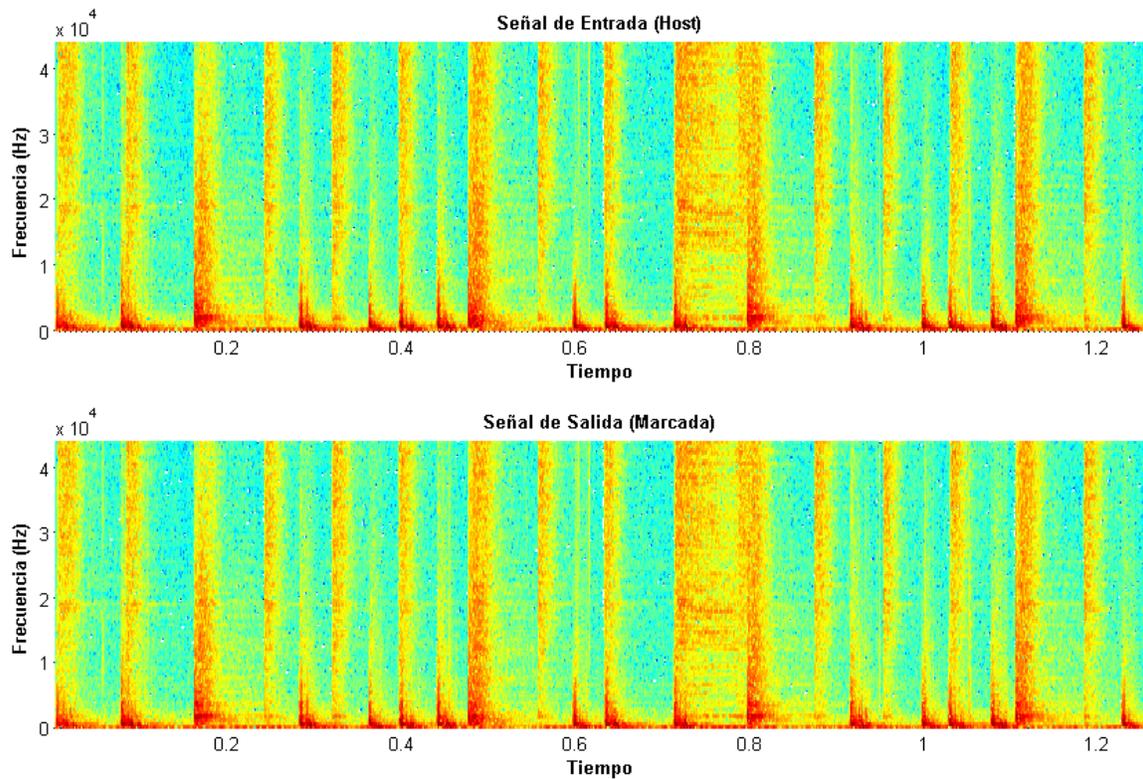


Fig. 4.9 Espectrograma de ambas señales

Se puede ver que la intensidad de la señal marcada se atenúa un poco, este efecto se debe a que se está incrustando información más no adhiriéndola, por lo que existe una sustitución de bits del audio original que provoca dicha atenuación.

Las pruebas de robustez e inaudibilidad se hicieron a 100 archivos musicales que previamente fueron marcados con 120 bits mediante el algoritmo propuesto, con características similares pero divididos en cinco categorías musicales (Pop, Rock, Balada, Banda, Electrónica) de los cuales solo se reportan 20, se midieron los bits erróneos de marca de agua que hay en comparación con los que fueron incrustados y las pruebas MOS y el cálculo del ODG.

La Tabla 4.3 muestra la robustez frente a ataques de remuestreo a la señal marcada. Se utilizan diferentes tasas de muestreo para verificar que la marca de agua que fue incrustada a la señal de audio sobrevive a este tipo de procesamiento.

### 4.3 Robustez de la marca de agua frente a ataques de “Remuestreo”

<b>Categoría</b>	<b>Frecuencia de Muestreo (kHz)</b>	<b>Bits Erróneos / Total de bits incrustados</b>	<b>Bit Error Rate (% BER)</b>
Pop	Fs/4	1/120	0.833
	Fs/2	0/120	0
	2*Fs	0/120	0
	4*Fs	2/120	1.666
Electronic	Fs/4	2/120	1.666
	Fs/2	1/120	0.833
	2*Fs	2/120	1.666
	4*Fs	3/120	2.5
Banda	Fs/4	0/120	0
	Fs/2	0/120	0
	2*Fs	0/120	0
	4*Fs	1/120	0.833
Ballad	Fs/4	1/120	0.833
	Fs/2	0/120	0
	2*Fs	1/120	0.833
	4*Fs	2/120	1.666
Rock	Fs/4	1/120	0.833
	Fs/2	0/120	0
	2*Fs	0/120	0
	4*Fs	2/120	1.666

Como se puede visualizar en la Tabla 4.3, la marca de agua insertada por el método propuesto es prácticamente robusta a cualquier tasa de muestreo debido a que sólo hay un mínimo de BER.

La siguiente tabla muestra la robustez de la marca de agua frente a la adición de un ruido blanco gaussiano de diferentes intensidades de relación señal a ruido.

Tabla 4.4 Robustez de la marca de agua frente a ataques de “AWGN”

<b>Categoría</b>	<b>Intensidades de SNR (dB)</b>	<b>Bits Erróneos / Bits incrustados Totales</b>	<b>Bit Error Rate (% BER)</b>
Pop	1	76/120	63.333
	5	58/120	48.333
	10	41/120	34.166
	15	20/120	16.666
Electronic	1	94/120	78.333
	5	79/120	65.833
	10	62/120	51.667
	15	53/120	41.267
Banda	1	38/120	31.778
	5	25/120	20.823
	10	17/120	14.176
	15	11/120	9.198
Ballad	1	51/120	42.5
	5	44/120	36.667
	10	30/120	25
	15	29/120	24.177
Rock	1	83/120	69.156
	5	77/120	64.167
	10	62/120	51.667
	15	49/120	40.833

En la Tabla 4.4 podemos ver que la marca de agua no es lo suficientemente robusta para ataques de adición de ruido blanco gaussiano debido a que en la mayoría de las categorías musicales y con sus respectivas intensidades de ruido, la marca de agua se destruye en promedio a la mitad de sus bits de composición, esto genera que no pueda detectarse el mensaje oculto dentro del audio.

En las tablas anteriores se desarrollaron dos ataques comunes en el marcado de agua digital, esto con la intención de mostrar el estudio realizado a cada uno de los ataques proporcionados por la herramienta Audio StirMark (ver Tabla 3.1) y se verificó a cuáles es robusta la marca de agua insertada por el método de ocultamiento propuesto y se obtuvieron los resultados de la Tabla 4.5:

Tabla 4.5 Algunos ataques soportados y no soportados por el algoritmo de Ocultamiento por Eco

<b>Ocultamiento por Eco</b>	
<b>Robusta</b>	<b>Frágil</b>
Remuestreo	AWGN
Nothing	AddDynNoise
Amplify	BitChanger
ExtraStereo	CopySample
TimeStrech	CutSamples
Compressor	LSBZero
AddBrumm	RC_HighPass
	RC_LowPass
	ReplaceSamples
	Invert
	Zeros (ZeroCross, Zerolength1 y 2, ZeroRemove)

Para las pruebas de inaudibilidad se calculó el Grado de diferencia Objetiva (ODG) y se hizo una comparación con la Tabla 3.3 para verificar el probable deterioro producido por el procesamiento del audio. En la Tabla 4.6 podemos visualizar los resultados obtenidos:

Tabla 4.6 ODG de las categorías musicales

<b>Categoría</b>	<b>Nombre de la Canción</b>	<b>ODG</b>
Pop	Dopamina (by Belinda)	-0.753
	Contigo (by Yuridia)	-0.895
	Nada puede cambiarme (by Paulina Rubio)	-0.101
	Mentiras Piadosas (by Alejandra Guzmán)	-0.832
Electronic	Las pequeñas cosas (by Jot Dog)	-2.761
	Lentamente (Fey)	-1.676
	Nada de más (by Belanova)	-2.183
	Provócame (by Fey)	-1.752
Banda	Y tú (by Julián Álvarez)	-0.166
	Mujeres divinas (by Vicente Fernández)	-0.108
	No me queda más (by Selena)	-0.126
	Ya no (by Limon)	-0.178
Ballad	Chiquitita (by Abba)	-0.419
	Canta corazón (by Alejandro Fernández)	-0.610
	Sortilegio (by Il divo)	-0.291
	Hasta mi final (by Il divo)	-0.523
Rock	I feel good (by James Brown)	-1.361
	Absolutamente (by Fangoria)	-0.594
	Lejos estamos mejor (by Motel)	-0.431
	Frágil (by Allison)	-0.849

Los resultados teóricos obtenidos muestran que la mayoría de los archivos musicales marcados se encuentran dentro de la descripción de perceptible pero no molesto y la descripción imperceptible, lo cual hace ver que el algoritmo es inaudible para el oído humano.

De manera subjetiva se midió el MOS (Mean Opinion Score) frente a un auditorio de 61 personas, en el cual las personas calificaban la percepción de la calidad del audio que percibían de acuerdo a la Tabla 3.2 y a continuación se muestran los resultados:

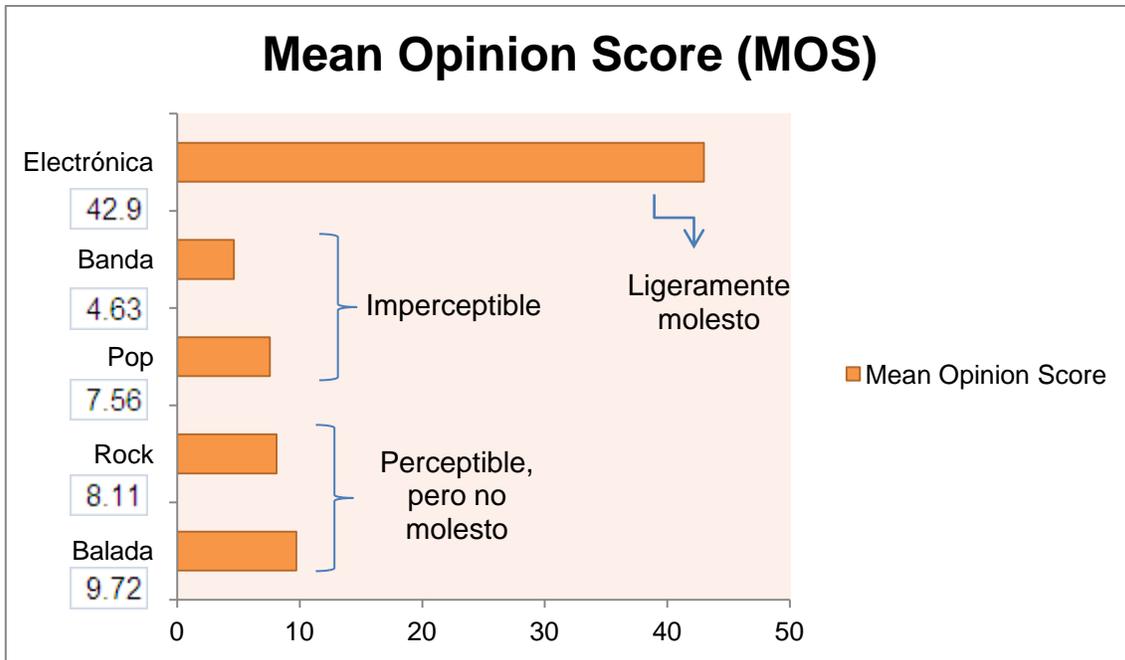


Fig. 4.10 Prueba de inaudibilidad MOS

Para el proceso de detección se tuvo que calcular el cepstrum a cada bloque de inserción con la finalidad de verificar si existía un eco en ellos o no y así decidir si es un “1” o un “0” incrustados. Las gráficas obtenidas para cada caso se muestran a continuación:

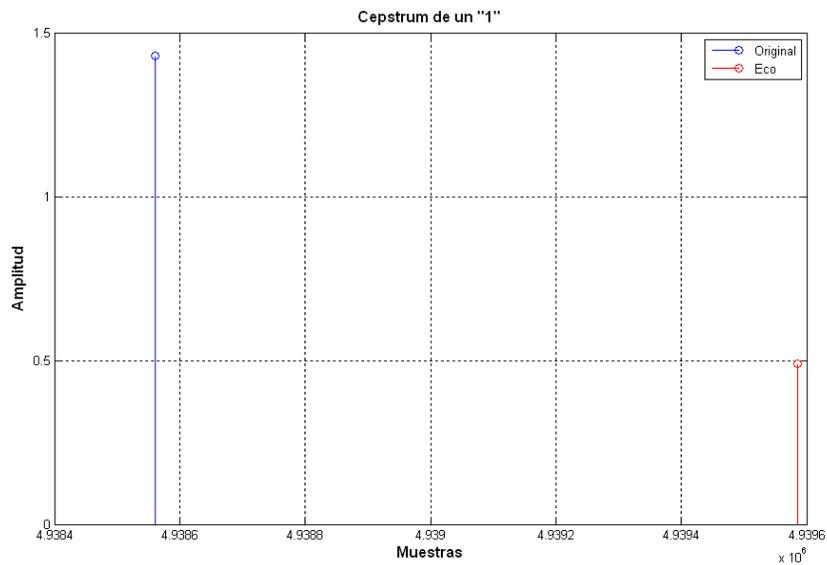


Fig. 4.11 Cepstrum para un bloque con un “1” incrustado

Para la inserción de un “0” se hizo el mismo cálculo del cepstrum pero la amplitud del eco resulta ser prácticamente nula debido a que no se le suministra un retraso para insertar el bit:

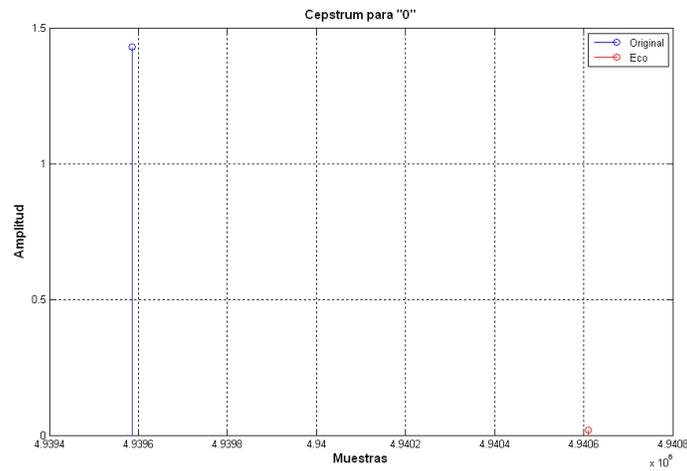


Fig. 4.12 Cepstrum para un “0” incrustado

De acuerdo a las anteriores imágenes vemos que si el cepstrum es mayor a cuatro indica que hay un retroceso (eco) y por lo tanto es un “1” incrustado y si ese umbral es mucho menor a 4 será el que el bloque no tuvo relaciones, pero si el eco es de amplitud considerable y estamos enojados.

## 4.2 Resultados Obtenidos de la Implementación en el FPGA

A la salida de cada subsistema (Inserción, Detección y generación de marca de agua) se obtuvieron algunos gráficos que se muestran a continuación:

La siguiente figura muestra la salida de la generación de la marca de agua incrustada, y la salida del esquema de detección de la marca de agua.

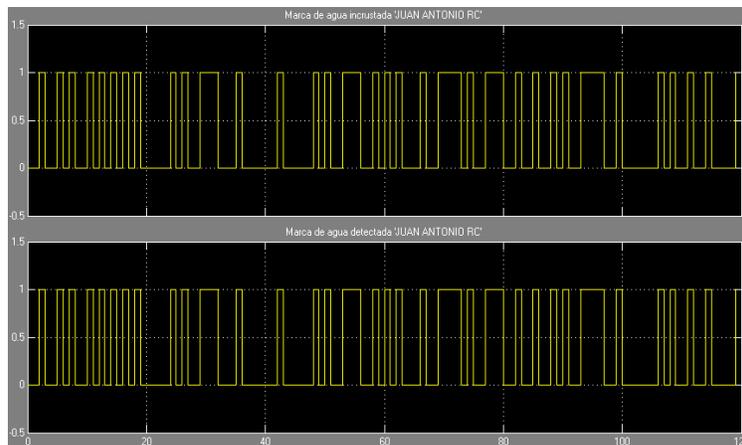


Fig. 4.13 Marca de agua incrustada y marca de agua detectada

Como puede verse la marca de agua generada (120 bits incrustados) es la misma que la señal de marca de agua obtenida del generador. En este caso resulta la detección óptima al 100% debido a que no existe ningún ataque de por medio. Idealmente la detección de la marca de agua no debe sufrir deterioros o modificaciones de la señal de marca de agua incrustada.

A la salida del sistema de inserción se debe obtener el audio marcado reconstruido de los bloques que anteriormente fueron procesados

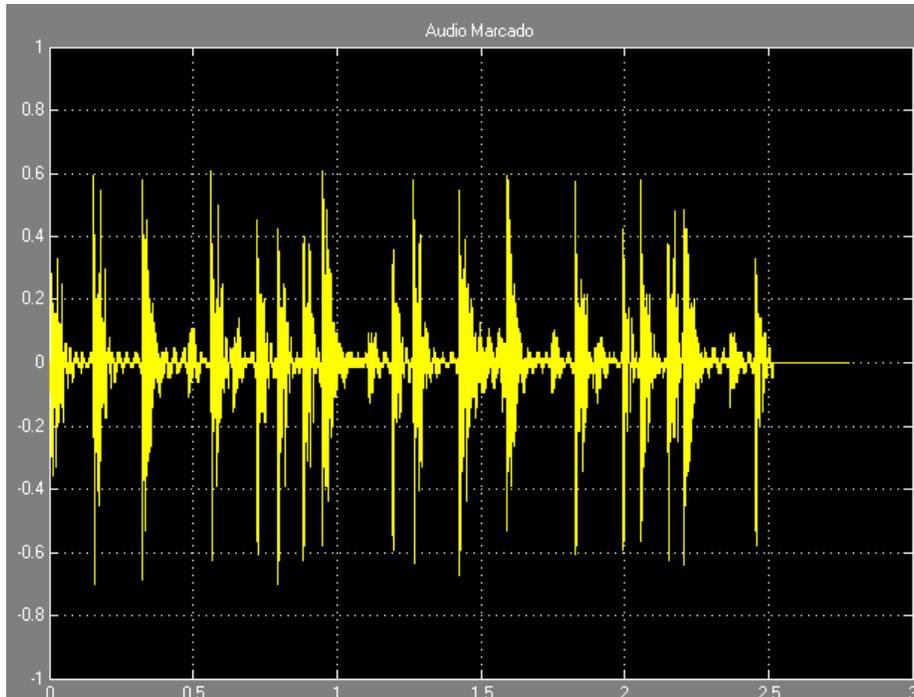


Fig. 4.14 Salida del sistema de inserción

## Conclusiones

Las técnicas de marcado de agua digital pueden analizarse en el dominio temporal o frecuencial, de ahí depende su complejidad matemática, tal y como puede verse en el capítulo 1. El algoritmo elegido trabaja en el dominio temporal y su análisis resulta mucho más sencillo.

Los resultados obtenidos de la evaluación de la robustez de la marca de agua incrustada muestran que no es robusta para algunos ataques o procesamientos digitales a los que se someta el archivo marcado. Estos resultados nos hacen ver que el algoritmo resulta frágil de manera general por lo que las aplicaciones que podría tener son:

- ✓ Como identificador: a manera de mostrar el nombre del propietario de alguna obra musical.
- ✓ Radiodifusión: etiqueta de canciones o información adicional de la emisión de radiodifusoras.
- ✓ Como prueba ante un tribunal ya que si alguna grabación forma parte de alguna prueba ilícita y ésta es modificada a favor de alguna parte, la fragilidad de la marca de agua será de apoyo para determinar cuando la grabación fue procesada.

Para aumentar la robustez de la marca de agua puede optarse por agregar múltiples ecos en el algoritmo de inserción, pero a sabiendas que esto aumentaría la complejidad del sistema de detección y costo computacional.

La implementación del algoritmo de inserción de marcas de agua mediante ocultamiento por eco se realizó mediante la implementación de bloques funcionales de la herramienta Xilinx System Generator (XSG). XSG es una herramienta multifuncional en la que por medio de la unión de bloques se pueden realizar las aplicaciones que se necesiten sin necesidad de programar directamente en VHDL ya que los bloques están predefinidos con el lenguaje y programar directamente mediante texto probablemente se hubiesen requerido miles de líneas que habría que verificar para obtener los resultados correctos y además un alto grado de complejidad computacional.

## Referencias

- [1] H. F. Eberhard, *"Psychoacoustics, facts and models"*, Munchen, German, Ed. Springer, August 2006.
- [2] M. Hrncar, J. Krajcovic, *"Principles of Audio Watermarking"*, Bytcka Zilina, Slovak Republic, 2005, pp. 1-3.
- [3] Michael A. *"Audio Watermarking: Features, applications and algorithms"*, Department for Security Technology for Graphics and Communication Systems, Darmstadt, Germany, IEEE, 2000.
- [4] Nedeljko, C., T. Seppanen, *Digital Audio Watermarking Techniques and Technologies: Applications and Benchmarks*, Information Science Reference, Hershey New York, 2007.
- [5] Matt. L M., Ingemar J. C., *"A review of watermarking principles and practices"*, Digital Signal Processing in Multimedia Systems, 461-465, USA, 1999
- [6] Elham K. M., et. al., *"A Comparison of some Audio Watermarking Methods"*, University of Notre Dame, USA, IEEE.
- [7] Foo. S. W., Dong Q., *"Audio Watermarking of Stereo Signals Based on Echo-Hiding Method"*, Nanyang Technological University, Singapore, IEEE, 2009.
- [8] Chou S. A., S. F. Hsieh, *"An Echo-Hiding Watermarking Technique Based on Bilateral Symmetric Time Spread Kernel"*, Department of Communication Engineering, National Chiao Tung University Hsinchu, Taiwan, Republic of China, IEEE, 2005.
- [9] J. A Ríos Chávez, E. Escamilla Hernández, M. González Lee, *"Audio Watermarking of WAV Files by Echo Modulation"*, SEPI ESIME UC, México D.F. 2011
- [10] Shine K. P., Krishna K. S., *"Extended Bipolar Echo Kernel for Audio Watermarking"*, Centre for Development of Advanced Computing, Trivandrum, India, IEEE, 2009.
- [11] Gui-jui N., et. al., *"Research of Improved Echo Data Hiding: Audio Watermarking Based on Reverberation"*, Jilin University, China, ICASSP, 2007.
- [12] H.J. Kim, S. Xiang, and S. Maitra, *"Robustness Analysis of Patchwork Watermarking Schemes"*, IEEE, Transactions on Speech and Audio Processing, Seoul Korea, 2001, pp. 1-2
- [13] Fabien A. P. P., Ross J. A., *"Attacks on Copyright Marking Systems"*, University of Cambridge, Computer Laboratory, USA, IEEE, 1998.
- [14] Kaliappan G., *"Audio Steganography Using Bit Modification"*, Department of Engineering, Purdue University Calumet, Hammond, 2000.
- [15] J.E. Gilley, *"Bit Error Rate Simulation using Matlab"*, Transcrypt International, Inc., August 2003, pp. 1-7.
- [16] Recommendation ITU-R BS.1387-1 Method for objective measurements of perceived audio quality, 2001.

- [17] T.C. Thanuja, Dr. R. Nagaraj, "Schemes for Evaluating Signal Processing Properties of Audio Watermarking", India, IJCSNS, July 2008.
- [18] Tal Mizrahi, et. al., "Real Time Implementation for Digital watermarking in Audio Signals Using Perceptual Masking", Dept. of EE, Technion, Haifa Israel, 2001.
- [19] C. A. Genaro Olivas, "Marcado de Agua para Audio Digital en Lógica Programable", Tesis de Maestría, SEPI ESIME CU, Diciembre 2009.

## **Ciberografía**

- [20] <http://www.sacm.org.mx/inicio.asp>
- [21] <http://www.adobe.com/audition>
- [22] <http://www.adobe.com/soundbooth>
- [23] <http://www.iis.fraunhofer.de>
- [24] <http://www.musictrace.de>
- [25] <http://www.watermarkingworld.org>
- [26] <http://estudiarsonido.wordpress.com/>

# **Anexos**

## **Introducción**

A continuación se presenta la productividad obtenida con la investigación desarrollada.

**INNOVA-IPN 2011 (author)**

[Help](#) [Sign out](#)

[Overview](#) [New Submission](#) [Paper 50](#) [INNOVA-IPN 2011](#) [EasyChair](#)

**INNOVA-IPN 2011 Submission 50**

**The submission has been saved!**

\*\*\*\*\*

If you want to **change any information** about your paper or withdraw it, use links in the upper right corner.

\*\*\*\*\*

- [Update information](#)
- [Update authors](#)
- [Submit a new version](#)
- [Withdraw](#)

**Paper 50**

**Title:** Algoritmo de marcado digital mediante ocultamiento

**Paper:** [Word open XML document](#)

**Attachment:** [file](#)

Algoritmo

**Keywords:** Mercado digital

**Topics:** Aplicaciones de software (apps)

**Abstract:** Es un sistema capaz de marcar archivos digitales de fácil manejo para el usuario final y con la finalidad de proteger los derechos de autor.

**Time:** Oct 27, 20:14 GMT

**Fax:**

**Address:** AV SANTA ANA No 1000 COL. SAN FRANCISCO CULHUACAN MEXICO DF 04430 Mexico

**Authors**

	Name	Email	Country	Affiliation	
<b>Authors:</b>	Juan Antonio Ríos Chávez	jcbaez@prodigy.net.mx	Mexico	ESIME CULHUACAN	✓

*Note: the rightmost column marks corresponding authors*

México, D. F. a 20 de Octubre de 2011.

Eng. Juan Antonio Rios Chavez[1],PhD. Enrique Escamilla Hernandez[2]  
NATIONAL POLYTECHNIC INSTITUTE, ESIME U Cuahuacan  
jriasc0500@ipn.mx, eescamilla@ipn.mx

Estimado *Eng. Rios*:

Nos es muy grato informar a usted, que su ponencia ha sido aceptada, para ser presentada en la **VIGESIMASEGUNDA . REUNIÓN INTERNACIONAL DE OTONO DE COMUNICACIONES, COMPUTACIÓN, ELECTRÓNICA, AUTOMATIZACIÓN, ROBÓTICA Y EXPOSICIÓN INDUSTRIAL ROC&C'2011**, a celebrarse del 27 de Noviembre al 3 de Diciembre del 2011, en el *Centro Internacional Acapulco, Gro, México*, para lo cual agradeceremos se sirva enviar por fax y/o correo el siguiente documento debidamente firmado.

Ing. Vicente Núñez González  
Director Ejecutivo  
IEEE Sección México  
Presente..

### **CARTA COMPROMISO**

Mediante este documento y de acuerdo con los ESTATUTOS DEL IEEE, me comprometo a presentar personalmente la ponencia cuyo titulo es,

#### **37.- GENERATING A GRAPHICAL USER INTERFACE FOR EMBEDDING A WATERMARK INTO AUDIO SIGNALS BY ECHO MODULATION**

o bien, enviar a un representante con conocimientos suficientes del tema, notificando al IEEE, de este cambio, *por lo menos 10 días antes del evento*.

**Atentamente,**

Eng. Juan Antonio Rios Chavez[1],PhD. Enrique Escamilla Hernandez[2]

**IMPORTANTE:** Para realizar los TRAMITES DE REGISTRO EN EL HOTEL con mucho gusto, le anexamos la documentación correspondiente.

El IEEE Sección México le extiende una muy cordial invitación a participar como *Moderador* en esta **ROC&C'2011**, moderando algunas de las *Secciones Técnicas de su interés y especialidad*. En caso de aceptar ayudarnos, le suplicamos nos lo comunique, para considerarlo en el **Programa Técnico**.

Sin otro particular de momento, aprovechamos la ocasión para enviarle un cordial saludo y expresarle nuestro reconocimiento y gratitud por participar en la **ROC&C'2011**.

**Atentamente,**

  
Ing. Vicente Núñez González



México D.F. a 07 de octubre de 2011

JUAN ANTONIO RÍOS CHÁVEZ

SEPI-ESIME Unidad Culhuacan-IPN

Por medio de la presente el Comité de **Ciencias Físico-Matemáticas** del VI Congreso Nacional Estudiantil de Investigación (CNEI), VI Congreso de Investigación Politécnica (CIP) y Ira Jornada de Prototipos (JP), le notifica que su trabajo titulado: “**MÉTODO ROBUSTO DE INSERCIÓN DE MARCAS DE AGUA EN SEÑALES DE AUDIO**” ha sido aceptado.

El programa definitivo del Congreso estará disponible en la página web <http://www.cip.ipn.mx>, donde podrá consultar el día y la hora de su exposición.

Sin más por el momento y agradeciendo su participación, se le envía un cordial saludo.

**A T E N T A M E N T E**

  
**M. en C. Andrés Hernández Pineda**  
Presidente del Comité Organizador del  
VI CNEI y del VI CIP y la Ira JP

VI CONGRESO NACIONAL ESTUDIANTIL DE INVESTIGACIÓN VI CONGRESO DE INVESTIGACIÓN POLITÉCNICA 1ra. JORNADA DE PROTOTIPOS

otorga la presente **CONSTANCIA A**

Juan Antonio Ríos Chávez, Enrique Escamilla Hernández, Mario González Lee

Por su valiosa participación en la modalidad **ORAL** con el trabajo:

*"metodo robusto de insercion de marca de agua en señales de audio"*

25, 26 y 27 de octubre del 2011, CFIE-UPDCE-IPN, Ciudad de México

*"La técnica al servicio de la patria"*

M. en C. Andrés Hernández Pineda  
Presidente del VI Congreso Nacional Estudiantil de Investigación  
VI Congreso de Investigación Politécnica y 1ra Jornada de Prototipos

Dra. Yoloxóchitl Bustamante Díez  
Directora General  
Instituto Politécnico Nacional



**INSTITUTO POLITÉCNICO NACIONAL  
ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA  
UNIDAD CULHUACAN**



Otorga el presente

**RECONOCIMIENTO**

AL

**Ing. Juan Antonio Ríos Chávez**  
SEPI ESIME UNIDAD CULHUACAN

Por su destacada participación como ponente, con la Conferencia **"Mercado de Agua Digital en Sonido Mediante Ocultamiento por Eco"**, en el Décimo Séptimo Ciclo de Exposiciones y Conferencias sobre Acústica (CECA 2011)

**"PORQUE LA ACÚSTICA NO ES DE OTRO MUNDO, PERO SI DE OTRO NIVEL"**

México, D. F. a 10 de noviembre de 2011

Atentamente

*"La Técnica al Servicio de la Patria"*

M. en C. Alberto Paz Gutiérrez  
Director



**ceca**



**INSTITUTO POLITÉCNICO NACIONAL**  
**ESCUELA SUPERIOR DE INGENIERÍA MECÁNICA Y ELÉCTRICA**  
**UNIDAD CULHUACAN**

**75**  
*Años*  
INSTITUTO POLITÉCNICO NACIONAL  
1936-2011

**3er. Foro Nacional de Matemáticas en el Marco de la Ingeniería**  
**“CIENCIA E INGENIO REUNIDOS EN UN SÓLO ESPACIO”**  
**5, 6 y 7 DE OCTUBRE DE 2011**

Otorga el presente

**RECONOCIMIENTO**  
**A: JUAN ANTONIO RÍOS CHÁVEZ**

*Por su presentación del PROTOTIPO:*

*“Algoritmo de Mercado Digital Mediante Ocultamiento por Eco”*

ESTADOS UNIDOS MEXICANOS  
ATENTAMENTE

“LA TÉCNICA AL SERVICIO DE LA PATRIA”

S.E.P. I.P.N.  
ESCUELA SUPERIOR DE INGENIERÍA  
MECÁNICA Y ELÉCTRICA  
UNIDAD CULHUACAN

DIRECCION



  
M. en C. Alberto Paz Gutiérrez  
Director

  
Dra. Rosario del Pilar Gibert Delgado  
Coordinadora General