

Compresión de Señales de Voz en Tiempo Real usando Paquetes Wavelet

Jorge Ivan Marín, Oscar Hernán Ocampo, Alex García Quinchía.

Grupo de Procesamiento Digital de Señales y
Procesadores - GDSPROC Centro de Estudios e
Investigaciones de la Facultad de Ingeniería - CEIFI
Universidad del Quindío, A.A.460, Armenia- Colombia
gdsproc@uniquindio.edu.co

Abstract— En este trabajo se presentan dos estructuras de un sistema de compresión de voz basados en Paquetes Wavelet. Una de forma dinámica y otra estática. El sistema dinámico implementado sobre un DSP TMS320C6701, realiza un filtrado adaptativo y una codificación lineal predictiva (LPC : *Linear Prediction Coding*). Mientras que el sistema estático utiliza un árbol wavelet fijo y una codificación basada en el método del diccionario; implementado en la tarjeta de desarrollo FPGA SPARTAN III.

Keywords— Compresión, Transformada Wavelet, Paquetes Wavelet, Codificación.

I. INTRODUCCIÓN

Los sistemas de compresión de voz tienen como fin eliminar de la información original las partes innecesarias o redundantes, representando la información de forma compacta y acondicionada a las limitaciones en la transmisión o el almacenamiento[1]. Entre los métodos de compresión tradicionales y más utilizados están los esquemas basados en la Transformada Discreta del Coseno (DCT : *Discrete Cosine Transform*). Su importancia radica en que el espectro de la DCT permite determinar el contenido energético de la señal. Sin embargo, ciertos autores[2], [3] han mostrado que la Transformada Wavelet (TW : *Transform Wavelet*) presenta mejores resultados, y sobre todo cuando se trata de analizar señales no estacionarias como la voz. La TW se caracteriza por brindar información frecuencia-tiempo de la señal, debido a que emplea una ventana de longitud variable, la cual permite una mayor resolución y mejor localización en el tiempo de las diferentes componentes de frecuencia de la señal. Además, permite emplear gran variedad de wavelets u onditas que se asemejan a la señal a analizar, lo que facilita la descomposición en varias subbandas.

Existen varios métodos para la compresión de señales de voz empleando la TW, pero específicamente el proceso que se describe está basado en la Transformación por Paquetes Wavelet[3]. Nuestro interés se centra en este método debido a que ofrece una mayor flexibilidad en comparación a la Transformada Wavelet Discreta (DWT: *Discrete Wavelet Transform*), dado a que los paquetes permiten una descomposición subbanda completa, donde se puede escoger una forma de árbol que contengan los coeficientes necesarios para la codificación de la señal. Otra opción,

propuesta por Teolis [4] es La Transformada Wavelet Sobrecompleta (OCWT: *Overcomplete Wavelet Transform*), pero, el principal inconveniente es que las funciones base no son ortogonales por lo cual se presenta una redundancia innecesaria.

Por otro lado, en los sistemas dinámicos, se realiza una Codificación Lineal Predictiva (LPC: *Linear Prediction Coding*) basada en la forma de onda que presentan ciertos coeficientes en el dominio Wavelet. Esta codificación es ampliamente usada en sistemas de telefonía celular como CELP, VSELP y GSM[5].

II. MARCO TEÓRICO

A. Codificación de Voz

Los codificadores de voz se caracterizan por seguir la forma de la onda de la señal, o por extraer los parámetros del modelo fisiológico y de excitación glótica de la voz. Entre los codificadores más utilizados están[6]: la codificación subbanda, que consiste en dividir el espectro en subbandas para ser codificadas individualmente; la codificación predictiva, que almacena la diferencia o el valor precedente (diferencia de la predicción o término del error); y la codificación por transformaciones, en la cual se realizan transformaciones matemáticas de la señal en bloque de segmentos de la señal de voz, donde cada segmento de voz se caracteriza mediante coeficientes de transformación. Ejemplo de esta última técnica están la DCT y las Wavelets.

En este trabajo se utilizaron dos tipos de codificaciones: la codificación basada en *Run-Length Encoded (RLE)* para la FPGA, y la codificación lineal predictiva (LPC: *Linear Prediction Coding*) para el sistema desarrollado en el DSP. La RLE es un tipo de codificación que consiste en encontrar secuencias repetidas de datos, y sustituirlas por un código de control que indica el número sucesivo de repeticiones del mismo dato. Si no se encuentra ninguna correlación entre dos datos sucesivos, a cada uno se le incluye un código de control que indica que dicho dato se transmite sin compresión. Por su parte, la codificación lineal predictiva, en el sistema propuesto es usada para predecir la forma de onda de los coeficientes de los nodos terminales que tiene menor relevancia de la Transformada por Paquetes Wavelet, mientras que los coeficientes de las aproximaciones se transmiten sin modificaciones.

El principio de un LPC es que el valor actual de una muestra $s(n)$, puede predecirse a partir de un número finito de muestras anteriores: $s(n-1)$, ..., $s(n-p)$, con un error asociado $e(n)$ utilizando un filtro lineal sólo polos[5]:

$$s(n) = e(n) + \sum_{k=1} \alpha_k s(n-k) \quad (1)$$

El error de predicción (también conocido como señal residual), $e(n)$, es simplemente la diferencia entre el valor actual de la señal, $s(n)$, y el valor que se predijo, $s(n)$:

$$e(n) = s(n) - \hat{s}(n)\alpha_k \quad (2)$$

Los factores que otorgan el peso, α_k , son encontrados al minimizar el error cuadrático medio (E). Para N muestras:

$$E = \frac{(\sum_{i=0}^{N-1} e^2(i))}{2} \quad (3)$$

B. Transformada Wavelet

El término wavelet se define como una “pequeña onda” o función localizable en el tiempo. Esta debe cumplir algunas condiciones [4] que son, el ser una función de energía concentrada en el tiempo con media cero y exhibir alguna oscilación en el tiempo. Una función wavelet genera una familia de wavelets cuando esta se dilata o se contrae, para obtener distintos niveles de resolución, de esta forma, por medio de las versiones trasladadas en el tiempo de la wavelet de una cierta escala se genera un subespacio de señales. El propósito de la TW consiste en descomponer una señal $f(\tau)$ en una combinación lineal de una familia wavelet (versiones dilatadas y desplazadas de una función madre $g(\tau)$), lo cual se puede denotar a través de operador producto interno como[4]:

$$(W_g f)(t, s) = \langle f, \tau_t D_s g \rangle \quad (4)$$

donde W_g representa el operador transformada wavelet, τ_t el operador desplazamiento y D_s el operador de dilatación. la wavelet madre $g(\tau)$ se puede interpretar como la respuesta al impulso de un filtro pasa-banda, y la TW como la convolución con un filtro pasa-banda dilatado con la escala s . Cuando la escala es grande, $W_g f$ detectará las componentes de baja frecuencia de la señal $f(\tau)$. Sin embargo, si la escala disminuye, el filtro $D_s g$ es más estrecho, por lo que la TW corresponde a detalles más finos de la señal. Se puede notar que la TW de una señal unidimensional es una función bidimensional, en la cual los dos ejes son t (tiempo) y s (escala), donde la escala está asociada al inverso de la frecuencia, de allí que la TW sea por excelencia una transformación frecuencia-tiempo de la señal.

La representación dada por la ecuación 4 es conocida como la Transformada Wavelet en Tiempo Continuo y es altamente redundante y costosa a nivel computacional, por lo cual, para aplicaciones en tiempo real se realiza una discretización tanto en la escala como en el tiempo en potencias de dos o diádicas por medio de las relaciones $s = 2^k$ y $t = 2^k n$. Esto brinda un esquema que es conocido como la Transformada Wavelet Discreta (DWT), la cual se calcula mediante la utilización de un banco de filtros, en los que la

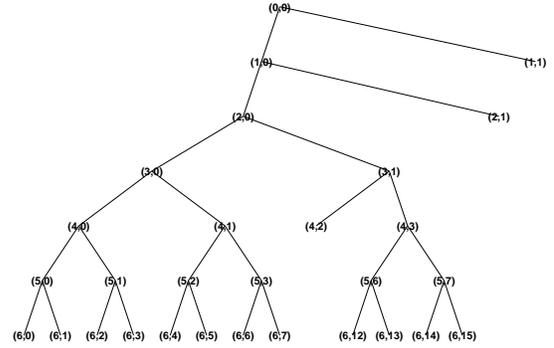


Fig. 1.

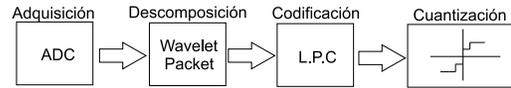


Fig. 2. Esquema de compresión con wavelet

respuesta al impulso de los filtros FIR $h(-n)$ (Pasa-Bajo) y $g(-n)$ (Pasa-alto) están asociados con la función Wavelet Madre. Además, los coeficientes a_k , d_k son conocidos como aproximaciones y detalles respectivamente[7], [8].

La descomposición en octavas que ofrece la DWT puede ser útil en un sistema de compresión, pero en este caso sería deseable disponer de una transformación que proporcione una buena resolución en regiones arbitrarias del plano tiempo-frecuencia. Este tipo de flexibilidad lo proporcionan los Paquetes Wavelet (PW) presentados por Coifman y Wickerhauser, y constituyen una generalización de la DWT[3]. En el análisis de PW, los detalles al igual que las aproximaciones se subdividen de manera sucesiva para producir nuevos espacios, esto se puede ver en la Figura 1.

III. SISTEMA DE COMPRESIÓN

El sistema de compresión propuesto se muestra en la Figura 2. Como puede notarse, el esquema consta de una transformación, codificación y cuantización. Cada una de estos bloques se realiza tanto para el compresor dinámico como para el estático, implementados en el DSP TMS320C6701 y la tarjeta de desarrollo FPGA SPARTAN III, respectivamente.

El cálculo de la transformada por Paquetes Wavelet para el compresor estático se determina mediante un árbol fijo que se obtuvo realizando pruebas de diferentes voces (hombres y mujeres), muestreadas a $8KHz$ en bloques de 1024 datos; pronunciando palabras de alta periodicidad como “murciélago” y “farmacéutico”.

Los bloques de 1024 datos se cargaron en el GUI Wave-menu del *toolbox Wavelet* de Matlab, eligiendo la opción de *Wavelet Packet 1-D*. Utilizando las Daubechies como wavelets madre por su similitud con las muestras, pero en particular la *db8* y la *db4*.

Se usaron las funciones de *shannon* y *norm* (potencia 1) para el cálculo de la entropía, y como la longitud de los

datos es 1024, el nivel de profundización máximo usado fue de 10.

Se demostró que el nivel de profundización que más se repiten es el 6 con el árbol de la Figura 1, con *db8* y la función *norm (potencia 1)*.

Por otra parte, para la codificación de la señal con árbol dinámico se propone emplear el algoritmo de Coifman y Wickerhauser[3] para la selección del mejor árbol. Si L es el nivel máximo, existen 2^L modos de descomponer la señal (Figura ??). Puesto que este número puede ser muy grande, es importante encontrar la mejor base que representa de manera eficiente y apropiada la señal a analizar. El algoritmo consiste en un filtrado adaptativo en el que el árbol de descomposición va cambiando en función del comportamiento temporal de la señal, esto conduce a mejores resultados debido a que se busca una representación óptima de la señal escogiendo una base ortonormal. Esto se logra minimizando una función de coste, típicamente la entropía, definida para una secuencia X_i como[3], [7]:

$$\mu(\{X_I\}) = - \sum P_n \ln P_n \quad (5)$$

donde

$$P_n = \frac{\|X_n\|^2}{\sum \|X_n\|^2} \quad (6)$$

La mejor base es para cuando la transformación de la señal tiene la mínima función de coste. El procedimiento para calcular la mejor base del Paquete Wavelet es el siguiente:

1. Escoger L como número máximo de niveles de descomposición.
2. Mientras el nivel de descomposición sea menor que L , se realiza los siguientes pasos:
 - Se calcula la función de coste para los coeficientes de la transformada en el nodo¹ i, n .
 - Se descompone el nodo i, n y se aplica la función de coste a los nodos $i + 1, 2n + 1$ y $i + 1, 2n$.
 - Si $\mu(i, n) > \mu(i + 1, 2n) + \mu(i + 1, 2n + 1)$ entonces permanecen los nodos $i + 1, 2n$ y $i + 1, 2n + 1$, de lo contrario solo se mantiene el nodo i, n , o también conocido como nodo padre.

IV. IMPLEMENTACION

Para la implementación de la estructura del árbol dinámico en tiempo real sobre el DSP, se empleó un procesador TMS320C6701 disponible en la tarjeta de desarrollo EVM6x, la cual incorpora un CODEC de audio empleado para la adquisición de la señal. En la captura de datos se usa una frecuencia de muestreo de 8KHz, una resolución de

¹ i Corresponde al nivel de descomposicion.
 n Es un nodo del nivel i

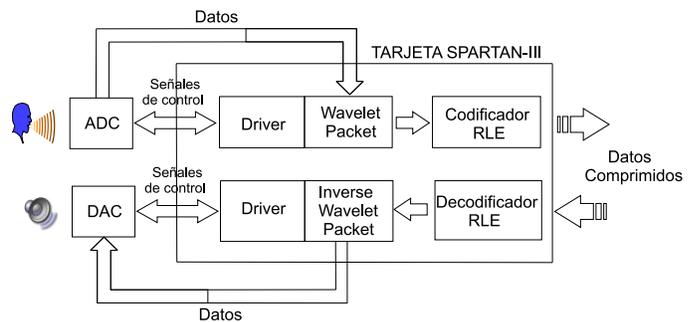


Fig. 3. Diseño del compresor en la FPGA

16bits, y la técnica de entrada/salida por Acceso Directo a Memoria (DMA) empleando buffer doble de 1024 muestras, para cada bloque se calcula un mejor árbol que se implementa eficientemente con ayuda de la estructura polifase. Esta estructura, por la anidación de ciclos for, permite que el compilador pueda generar un código optimizado para la implementación.

Con el fin de garantizar la continuidad en el procesamiento de cada bloque, en el cálculo del PW, se tuvo en cuenta conservar las condiciones iniciales de cada uno de los filtros FIR que conforman el árbol Wavelet. Además, se incluyó dentro del sistema de compresión un esquema de reducción del ruido que emplea la técnica *Wavelet Denoising* con umbral fuerte [7].

Por otro lado, se aplican filtros LPC de orden 10 a los coeficientes de los nodos terminales del PW que presentan menor relevancia, transmitiendo únicamente para dichos nodos los coeficientes calculados por el LPC. En la etapa de síntesis se recibe la estructura del árbol, los coeficientes LPC y los de los nodos que no se codificaron en la descomposición, estos coeficientes se introducen en el filtro LPC de síntesis, el cual es excitado mediante ruido blanco para recuperar los coeficientes Wavelet y reconstruir la señal con el Paquete Wavelet Inverso de dicho bloque.

Por otra parte, para la implementación de la estructura estática, ésta se realizó sobre la tarjeta de desarrollo FPGA SPARTAN-III de XILINX, utilizando el *ADS7825* para la adquisición de datos, con una frecuencia de muestreo de 40KHz y una resolución de 16 bits, el estilo de diseño utilizado fue el *estructural* debido a la facilidad en el manejo de jerarquías, en el descompresor, la recuperación de la señal se hace mediante el *DAC725* que tiene las mismas características del ADC. El esquema de cada uno de los componentes que forman el diseño se muestra en la Figura 3.

Los módulos *Driver* proporcionan las señales de control a los convertidores, El componente codificador elimina la redundancia y hace la cuantización. Las conexiones simbolizan buses de 16 Bits. el componente *Wavelet Packet* calcula el árbol mostrado en la Figura 4, el cual es el segundo árbol más probable en las pruebas de análisis de voz.

La elección de este árbol se debió a las limitaciones hard-

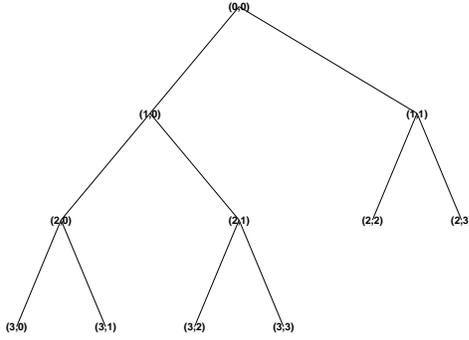


Fig. 4. Árbol Implementado

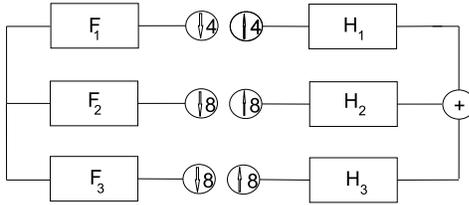


Fig. 5. Banco de Filtros Combinados

ware de la FPGA. Con el fin de facilitar la codificación del árbol wavelet estático se empleó una estructura que consiste en emplear, para la descomposición, un único filtro y un diezmador por 2^m , por cada uno de los nodos terminales del árbol, siendo m el número de niveles de partición que fueron necesarios para llegar desde la base del árbol terminal. En el caso de la síntesis se empleó un interpolador por 2^m a la entrada de un filtro.

La respuesta al impulso de los filtros empleados en la descomposición dependen del nodo en particular, y se calculan por medio de la iteración recursiva de las siguientes ecuaciones en diferencia:

$$f_l[n] = \sum_{k=0}^{N_l} f_{l-1}[n - 2^l k] h_l[k] \quad (7)$$

con $0 \leq n \leq (N_{l-1} - 1) + 2^l(N_l - 1)$, $l \geq 1$, $f_0[n] = h_0[n]$, N_l , la longitud del vector $h_l[n]$, N_{l-1} , la longitud de la secuencia f_{l-1} , y $h_{0,1}, 2, \dots$ las respuestas al impulso de los filtros QMF pasa-alto y pasa-alto necesarios para llegar desde la base del árbol ($l = 0$) hasta el nodo terminal. Para la síntesis, la respuesta al impulso es:

$$f_l[n] = \sum_{k=0}^{N_l} f_{l-1}[\frac{n-k}{2}] h_l[k] \quad (8)$$

Con las mismas condiciones para N , h y f de la descomposición. Finalmente, cabe mencionar, que en la implementación final se utilizan 6 de los 12 multiplicadores disponibles en la FPGA XC3S200FT256.

V. CONCLUSIONES

En este trabajo se mostró la factibilidad de implementación en un procesador digital de señales y en una FPGA de un sistema de compresión basado en paquetes wavelets. El sistema de compresión implementado en la FPGA tiene la limitante de contar con un árbol estático, a diferencia con el sistema basado en DSP, en el cual dicho árbol es dinámico y se evalúa por cada trama de datos, con el fin de ofrecer una mejor calidad en la compresión.

VI. RECONOCIMIENTOS

Este trabajo se ha podido realizar gracias al apoyo financiero de la Universidad del Quindío a través del proyecto 222.

REFERENCIAS

- [1] D. Salomon, *Data Compression Methods*. New York: Springer Verlag, 2001.
- [2] K. Rao and P. YIP, *The Transform and Data Compression Handbook*. Boca Raton: CRC Press LLC, 2001.
- [3] M. V. Wickerhauser, *Adapted Wavelet Analysis From Theory to Software*. Massachusetts: A K Peters, 1994.
- [4] A. Teolis, *Computational signal Processing with Wavelets*. Boston: Birkhauser, 1998.
- [5] J. A. B. Jiménez, "Estudio de sistemas de compresión de voz digital orientado a telefonía celular," 2002.
- [6] B. C. y Christian Schlageter, "Codificadores de voz," 1999.
- [7] R. M. Rao and Bopardikar, *Wavelet Transform*. Wesley: Addison, 1998.
- [8] Fliege, *Multirate Digital Signal*. New York: Jhon Wiley and Sons Ltd., 1994.