

(12) SOLICITUD INTERNACIONAL PUBLICADA EN VIRTUD DEL TRATADO DE COOPERACIÓN
EN MATERIA DE PATENTES (PCT)

(19) Organización Mundial de la Propiedad

Intelectual
Oficina internacional



(43) Fecha de publicación internacional
23 de Octubre de 2008 (23.10.2008)

PCT

(10) Número de Publicación Internacional
WO 2008/125708 A1

(51) Clasificación Internacional de Patentes:
G06F 17/14 (2006.01)

(21) Número de la solicitud internacional:

PCT/ES2008/000220

(22) Fecha de presentación internacional:

10 de Abril de 2008 (10.04.2008)

(25) Idioma de presentación:

español

(26) Idioma de publicación:

español

(30) Datos relativos a la prioridad:

P200700983 12 de Abril de 2007 (12.04.2007) ES

(71) Solicitante (para todos los Estados designados salvo US): **UNIVERSIDAD POLITÉCNICA DE MADRID** [ES/ES]; Calle Ramiro de Maezts, 7, 28040 Madrid (ES).

(72) Inventores; e

(75) Inventores/Solicitantes (para US solamente): **GARRIDO GÁLVEZ, Mario** [ES/ES]; Calle Ramiro de

Maezts, 7, 28040 Madrid (ES). **GRAJAL DE LA FUENTE, Jesús** [ES/ES]; Calle Ramiro de Maezts, 7, 28040 Madrid (ES).

(74) Mandatarios: **CARVAJAL Y URQUIJO, Isabel** etc.; c/o Clarke Modet & Co., C/Goya 11, E-28001 Madrid (ES).

(81) Estados designados (a menos que se indique otra cosa, para toda clase de protección nacional admisible): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Estados designados (a menos que se indique otra cosa, para toda clase de protección regional admisible): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ,

[Continúa en la página siguiente]

(54) Title: STORAGE-FREE METHOD AND ARCHITECTURE FOR COMPUTING FFT ROTATIONS

(54) Título: PROCEDIMIENTO Y ARQUITECTURA SIN MEMORIA PARA EL CÁLCULO DE LAS ROTACIONES DE LA FFT

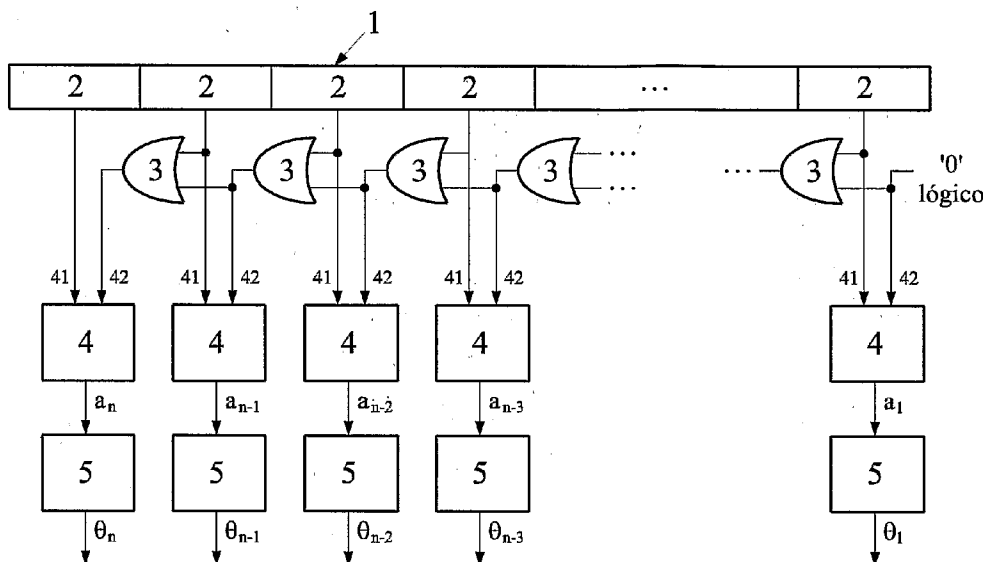


FIG. 1

(57) Abstract: A storage-free method and architecture for computing FFT rotations makes it possible to compute the FFT decomposed according to the Cooley-Tukey algorithm without using stored data. The rotation angles of the FFT steps are generated by a single counter (1) and a circuit comprising adders and logic gates, eliminating the need to store data related to rotation angles. Rotations are computed using a modified CORDIC algorithm which makes it possible to simplify the micro-rotation computing blocks. Moreover, a system is disclosed which uses only two subtractors to compensate for the typical scaling of the CORDIC algorithm.

[Continúa en la página siguiente]

WO 2008/125708 A1



UG, ZM, ZW), euroasiática (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), europea (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Publicada:

- *con informe de búsqueda internacional*
- *antes de la expiración del plazo para modificar las reivindicaciones y para ser republicada si se reciben modificaciones*

(57) Resumen: Procedimiento y arquitectura sin memoria para el cálculo de las rotaciones de la FFT. Permite el cálculo de la FFT descompuesta según el algoritmo Cooley-Tukey, sin recurrir a datos de memoria. Los ángulos de rotación en las etapas de la FFT, se generan a partir de un único contador (1) y un circuito compuesto por sumadores y puertas lógicas, eliminándose la necesidad de almacenar datos relacionados con ángulos de rotación. Las rotaciones se calculan empleando una modificación del algoritmo CORDIC que permite simplificar los bloques de cálculo de las microrrotaciones. Además, se presenta un sistema que utiliza únicamente dos restadores para compensar el escalado propio del algoritmo CORDIC.

Procedimiento y arquitectura sin memoria para el cálculo de las rotaciones de la FFT

Sector técnico.

La invención que se describe se enmarca dentro del campo del procesamiento de señal, más en concreto en lo referente a la elaboración de algoritmos y el desarrollo de arquitecturas de circuito eficientes para llevar a cabo dichos algoritmos.

Estado de la técnica.

La transformada de Fourier es una de las operaciones fundamentales en el campo del procesamiento de señal, especialmente en análisis espectral. Para su aplicación en sistemas digitales se hace uso de la DFT (*Discrete Fourier Transform*), que permite realizar la transformada de Fourier sobre datos almacenados en un ordenador o muestreados mediante un conversor analógico-digital, y cuya fórmula es:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j \frac{2\pi}{N} kn}, \quad k = 0, 1, \dots, N-1$$

En la ecuación, el valor N indica el número de puntos sobre los que se calcula la DFT, $x[n]$ son las muestras en el dominio del tiempo, k es una variable discreta que representa la frecuencia, y $X[k]$ es la señal en el dominio de la frecuencia, que se define para los valores de k desde 0 a $N-1$.

Con el objetivo de calcular la DFT de forma eficiente, el nombre FFT (*Fast Fourier Transform*) engloba un conjunto de algoritmos que reducen el número de operaciones requerido por la DFT. De todos ellos, el algoritmo de Cooley-Tukey [CT65] es el más utilizado. Éste se basa en descomponer la DFT en $n = \log_r N$ etapas de cálculo en cascada, donde r es el *radix* empleado en la descomposición. Por otra parte, la descomposición se puede realizar siguiendo distintos métodos. Los más comunes son el diezmado en tiempo (DIT) y el diezmado en frecuencia (DIF). De esta forma se consigue una reducción en el número de operaciones, que pasa de un orden $O(N^2)$ en la DFT, a un orden $O(N \log_r N)$ en la FFT.

Cada una de las n etapas de la FFT se caracteriza por tener que calcular un conjunto de sumas, restas y rotaciones en el plano complejo. Las sumas y restas se realizan

mediante elementos denominados mariposas, y las rotaciones complejas mediante rotadores.

Cada mariposa recibe r datos de entrada y ofrece r salidas que representan la FFT de r puntos de las entradas. Como la FFT de 2 puntos y la de 4 puntos se pueden realizar mediante rotaciones triviales (0° , 90° , 180° y 270°), lo más habitual es utilizar FFTs de *radix* 2 ó 4, ya que en otros casos es necesario incluir rotadores dentro de las mariposas para llevar a cabo las rotaciones no triviales que aparecen.

En cada etapa s de la FFT, $s \in \{1 \dots n\}$, se calculan N/r mariposas y N rotaciones. Sin embargo, en las arquitecturas de los circuitos de cálculo de la FFT, todas las mariposas se suelen calcular con un único circuito, aprovechando que los datos llegan de formas secuencial. Lo mismo ocurre con las rotaciones: un único rotador es suficiente para llevar a cabo todas las rotaciones de una etapa. De esta forma, se consigue un ahorro importante de componentes. Sin embargo, en el caso de las rotaciones es necesario conocer el ángulo de rotación (o algún dato relacionado con él) que hay que rotar cada una de las muestras que van llegando al rotador.

Existen dos métodos principales para la realización de las rotaciones de la FFT. El primero consiste en rotar el dato de entrada aplicando directamente la fórmula de una rotación, mediante el empleo de multiplicadores reales y sumadores. La otra opción, más utilizada por requerir en general menos recursos, es emplear el algoritmo CORDIC, propuesto por J.E.Volder [Vol59], y que se describe brevemente a continuación.

El algoritmo CORDIC se utiliza para calcular de forma eficiente operaciones matemáticas complicadas en sistemas digitales. Este algoritmo se basa en transformar estas operaciones en un conjunto de sumas y desplazamientos, que son fáciles de llevar a cabo en los circuitos digitales.

Si se utiliza para realizar rotaciones en el plano complejo, el algoritmo CORDIC descompone el ángulo de rotación, θ , en una suma de ángulos, α_i :

$$\theta = \sum_{i=m}^M \alpha_i + \epsilon$$

donde m y M son respectivamente los índices del primer y último ángulo considerado, ϵ

es el error de la aproximación y

$$\alpha_i = \pm tg^{-1}(2^{-i})$$

Habitualmente $m = 0$ y, por lo tanto, el primer ángulo es $\alpha_0 = tg^{-1}(1) = 45^\circ$, lo que hace que el algoritmo sea válido para cualquier valor de θ en el intervalo $[-90^\circ, 90^\circ]$.

- 5 Cuando θ se encuentra fuera de este intervalo se utilizan las rotaciones triviales de 180° y 90° para situarlo en él. De hecho, mediante estas rotaciones es posible dejar el ángulo restante de rotación, z , en el intervalo $[-45^\circ, 45^\circ]$. De acuerdo con esta idea, es posible prescindir de la rotación de 45° y considerar que la primera rotación del algoritmo CORDIC es para el caso $m = 1$, siendo $\alpha_1 = tg^{-1}(2^{-1}) \approx 26,5^\circ$.

- 10 La rotación del dato de entrada al rotador se lleva a cabo mediante una serie de etapas llamadas microrrotaciones, en cada una de las cuales se rota un ángulo α_i de acuerdo con las ecuaciones:

$$x_{i+1} = x_i - y_i \delta_i 2^{-i}$$

$$y_{i+1} = y_i + x_i \delta_i 2^{-i}$$

- 15 donde δ_i indica el sentido de la microrrotación y se calcula de acuerdo con:

$$\delta_i = \text{sign}(z_i)$$

$$z_{i+1} = z_i - \delta_i \cdot \alpha_i$$

siendo $\text{sign}(\eta) = 1$ si $\eta \geq 0$, y $\text{sign}(\eta) = -1$ si $\eta < 0$. Como $\delta_i \in \{-1, 1\}$, se realiza una rotación en cada una de las etapas, bien en sentido positivo o bien en sentido negativo.

- 20 Ello da lugar a una ganancia constante en el rotador, que puede ser compensada, en caso de que sea necesario, simplemente multiplicando las muestras de salida por:

$$K = \prod_{i=m}^M \cos(\alpha_i) = \prod_{i=m}^M \cos(tg^{-1}(2^{-i}))$$

- Para poder rotar adecuadamente cada uno de los datos de entrada al rotador se almacenan en una memoria los valores de los vectores de rotaciones, δ , correspondientes a cada una de las rotaciones que debe realizar el rotador. Estos datos son leídos de la memoria secuencialmente de tal forma que cada dato de entrada al rotador sea rotado el ángulo de rotación correspondiente.

Cuando el número de rotaciones que realiza el rotador de una cierta etapa de la FFT es elevado, la memoria de rotaciones deberá almacenar una gran cantidad de datos, lo que repercute negativamente en el área y en la velocidad de la FFT. Este problema es análogo en el caso de emplear multiplicadores para calcular las rotaciones, puesto que
5 deben ser almacenados los valores del seno y el coseno de los ángulos de rotación.

Ciertas investigaciones han conseguido reducir la memoria de rotaciones a $\log_2 N$ [YCC06], ó $0,5N$ [CP03]. Además, se han utilizado otros métodos de descomposición de la FFT (distintos al algoritmo de Cooley-Tukey) para reducir el número de rotaciones que hay que calcular, como ocurre en [ZH05], donde se emplea el método de descomposición
10 en factores primos, el cual sólo es aplicable cuando el número de puntos de la FFT se puede representar como producto de factores primos entre sí.

Por el contrario, con el procedimiento y la arquitectura propuestos se consigue reemplazar las memorias de rotaciones de todas las etapas de la FFT por un sencillo circuito compuesto por un contador (único para toda la FFT), y unos pocos sumadores y puertas
15 lógicas. Así, a partir del valor del contador se calculan las rotaciones de todas las etapas de la FFT aprovechando características comunes entre ellas. De esta forma, se consigue mejorar significativamente las prestaciones de la FFT, principalmente en cuanto a área. Además, la mejora es más significativa cuanto mayor sea el número de puntos de la FFT que se desea calcular, lo que hace que este procedimiento sea muy adecuado para el
20 cálculo de FFTs de un elevado número de puntos.

Por otra parte, las rotaciones se calculan empleando una modificación del algoritmo CORDIC que permite simplificar los bloques de cálculo de las microrrotaciones respecto a otras opciones existentes [MGBS02, Hu92].

Descripción detallada de la invención

25 Esta invención presenta un procedimiento para el cálculo de las rotaciones de cualquier FFT descompuesta según el algoritmo Cooley-Tukey, y cuyo número de puntos, N , y $radix$, r , son ambos potencia de 2, con los siguientes pasos:

- obtener las secuencias de ángulos de rotación de todas las etapas de la FFT,
- calcular las rotaciones correspondientes a cada una de las etapas de la FFT a partir

de los ángulos de rotación generados,

donde a partir de un único contador se obtienen las secuencias de ángulos de rotación de todas las etapas de la FFT. Este procedimiento tiene la ventaja de que no es necesario recurrir a ningún dato previamente almacenado.

5 Para implementar el procedimiento se utiliza una arquitectura de circuito que comprende:

- un módulo de generación de ángulos que incluye un único contador (1) para toda la FFT, a partir del cual se obtienen los ángulos de rotación de todas las etapas de la FFT sin necesidad de recurrir a ningún dato almacenado previamente,
- 10 ■ un módulo de cálculo de las rotaciones para cada una de las etapas de la FFT. Como se verá posteriormente, las rotaciones de alguna de las etapas de la FFT pueden resultar triviales, en cuyo caso se puede prescindir del módulo de cálculo de las rotaciones de dicha etapa.

Como se explicó anteriormente, una FFT de N puntos puede ser dividida en $n = \log_r N$ etapas, donde r es el *radix* de la FFT. Asumiendo que tanto N como r son potencia de 2, la secuencia de ángulos (de longitud N) de cualquier etapa s de la FFT, $s \in \{1 \dots n\}$, está formada por N/r^s secuencias iguales de longitud $L = r^s$.

La secuencia de ángulos depende del *radix* elegido en la descomposición de la FFT. Así, para *radix* 2, la secuencia de longitud L será:

$$20 \quad a_s = \underbrace{0, 0, 0, 0, \dots}_{2^{s-1}}, \underbrace{0, 1, 2, 3, \dots}_{2^{s-1}}$$

y para *radix* 4,

$$a_s = \underbrace{0, 0, 0, 0, \dots}_{4^{s-1}}, \underbrace{0, 1, 2, 3, \dots}_{4^{s-1}}, \underbrace{0, 2, 4, 6, \dots}_{4^{s-1}}, \underbrace{0, 3, 6, 9, \dots}_{4^{s-1}}$$

Generalizando estas expresiones para cualquier valor de *radix*, la secuencia de ángulos se puede generar concatenando r subsecuencias:

$$25 \quad 0, p, 2p, \dots, (r^{s-1} - 1)p$$

donde $p = 0, 1, 2, \dots, r - 1$.

Para normalizar la secuencia de todas las etapas de la FFT al mínimo ángulo de rotación, $\theta_{min}(rad) = 2\pi/N$, los valores de las secuencias deben ser multiplicados por $q = N/r^s$ o, lo que es lo mismo, desplazados $\log_2(q)$ bits:

$$\theta_s = q \cdot a_s = \underbrace{0, 0, 0, 0, \dots}_{2^{s-1}}, \underbrace{0, q, 2q, 3q, \dots}_{2^{s-1}}$$

La secuencia θ_s contiene los ángulos de rotación, θ , de la etapa s de la FFT, y cada ángulo de rotación $\theta \in [0, N-1]$, siendo su equivalencia en radianes:

$$\theta(rad) = -\frac{2\pi}{N}\theta$$

Desde el punto de vista digital, esta representación de θ como un valor entre 0 y $N-1$ resulta ventajosa por dos motivos. En primer lugar, el ángulo representado es exacto y el número de bits utilizado es mínimo; si el ángulo de rotación se representara en radianes siempre se cometería un error en la aproximación debido a que el número de bits es finito. Por otra parte, el valor N equivale a una vuelta de circunferencia completa, por lo que, como N es potencia de 2, multiplicar por 2π radianes se transforma en un desplazamiento de bits.

Considerando ahora que se dispone de un contador, para una etapa cualquiera, s , de la FFT, la generación de la secuencia de ángulos de rotación de dicha etapa, θ_s , se realiza siguiendo los siguientes pasos:

- tomar los $s \cdot \log_2 r$ bits menos significativos proporcionados por el contador,
- de los bits tomados, multiplicar los $(s-1) \cdot \log_2 r$ bits menos significativos por los $\log_2 r$ más significativos, obteniendo así los valores de la secuencia a_s . Los bits más significativos se corresponden con el valor de p mientras que los menos significativos contarán de 0 a $(r^{s-1} - 1)$.
- multiplicar el resultado por N/r^s , obteniendo así los valores de la secuencia θ_s .

El método explicado sirve para generar tanto los ángulos de la descomposición DIT (Diezmado en tiempo) como los de la descomposición DIF (Diezmado en frecuencia), considerando que $s=1$ es la etapa de entrada de la FFT en el caso de la descomposición DIT, y la etapa de salida en el caso DIF.

Para implementar la generación de ángulos se emplea el módulo de generación de ángulos representado en la figura 1. El módulo de generación de ángulos comprende, además del contador (1), los siguientes elementos:

- lógica combinacional (3), que opera sobre el valor del contador,
- un bloque acumulador (4) por cada etapa de la FFT, para el cálculo de las secuencias a_s .

Por otra parte, como se puede observar en la figura 2, cada bloque acumulador (4) comprende los siguientes elementos:

- un sumador (43), que permite ir actualizando los valores de la secuencia a_s ,
- un registro (45) que proporciona los valores de la secuencia a_s ,
- una puerta lógica (44) para el control del bloque acumulador.

El contador (1) es un contador binario que cuenta de 0 a $N - 1$ de forma periódica. Éste se considera dividido en s partes (2) de $\log_2 r$ bits. Cada una de ellas contiene el valor p de una de las etapas, y se corresponde con la entrada (41) del bloque acumulador de dicha etapa. En concreto, el valor de p para una etapa s incluye los bits que van desde la posición $(s - 1) \cdot \log_2(r) + 1$ a la $s \cdot \log_2 r$ del contador, considerando que el bit menos significativo es el bit 1.

Según la estructura del bloque acumulador (4), el registro (45) se irá incrementando de acuerdo con el valor de p siempre que no se active la señal de control (42).

Por otra parte, los $(s - 1) \cdot \log_2 r$ bits menos significativos del contador (1) realizan una cuenta cíclica desde 0 hasta $(r^{s-1} - 1)$. Cuando la cuenta de estos bits vuelve a cero, se activa la señal de control (42) que hace que se ponga a cero el valor del registro (45), que va proporcionando los valores de la secuencia a_s .

Finalmente, los bloques de escalado (5) no incluyen ningún componente lógico. Ello se debe a que, al ser N y r potencias de 2, el factor de escalado $q = N/r^s$ representa únicamente un desplazamiento fijo de bits, para lo cual sólo hay que ajustar el interconexionado de los bits con la etapa siguiente.

Así, la generación de las secuencias de ángulos de rotación θ_s de todas las etapas de la FFT se lleva a cabo mediante un sencillo circuito que no requiere ningún tipo de

memoria ni la realización de multiplicaciones. Además, sólo es necesario el empleo de un único contador para toda la FFT, a partir de cual se generan todas las secuencias de ángulos de rotación.

El siguiente paso del procedimiento descrito es el cálculo de las rotaciones correspondientes a cada una de las etapas de la FFT a partir de los ángulos de rotación generados. La forma de proceder es la misma para cada una de las etapas de la FFT, y se compone de los siguientes pasos:

- determinar si se deben realizar las rotaciones de 180° y 90° ,
- calcular el vector de rotaciones, δ ,
- obtener el vector de rotaciones adaptado δ' ,
- rotar el dato de entrada al rotador.

En primer lugar, para cada ángulo de rotación θ perteneciente a la secuencia θ_s se determina si se deben realizar las rotaciones de 180° y 90° con el fin de situar el ángulo restante de rotación, z , en el intervalo de definición del algoritmo CORDIC. Nótese que como $\theta(rad) = -2\pi\theta/N$, las rotaciones deberán realizarse en sentido negativo.

El hecho de representar θ como un valor entero, $\theta \in [0, N - 1]$, siendo N potencia de 2, presenta la ventaja de que en la arquitectura del circuito resulta muy sencillo determinar las rotaciones de 180° y 90° que permiten situar el ángulo restante de rotación z en el intervalo $[-45^\circ, 45^\circ]$. Para ello únicamente hay que sumar al conjunto de los dos bits más significativos de θ el siguiente bit más significativo. De esta forma, los dos bits resultantes de la operación indicarán respectivamente si se deben realizar las rotaciones de 180° y 90° . Además, z se obtiene directamente tomando todos los bits de θ excepto los dos más significativos y considerando que el z está expresado en complemento a 2, y cambiado de signo respecto al ángulo en radianes.

A partir de z se determina el vector de rotaciones, δ , de acuerdo con:

$$\delta_i = -\text{sign}(z_i)$$

$$z_{i+1} = z_i + \delta_i \cdot \alpha_i$$

siendo $\text{sign}(\eta) = 1$ si $\eta \geq 0$, y $\text{sign}(\eta) = -1$ si $\eta < 0$. Este cálculo se realiza a partir de $z_0 = z$ en el caso de emplear la microrrotación de $\alpha_0 = 45^\circ$ ($m = 0$); y a partir de $z_1 = z$ si

se considera que la primera microrrotación es la de $\alpha_1 = \tan^{-1}(2^{-1}) \approx 26,5^\circ$. El cambio de signo respecto al procedimiento habitual del CORDIC se debe únicamente a que, como se ha dicho anteriormente, la representación del ángulo en el intervalo $[0, N - 1]$ tiene signo contrario a la representación del ángulo en radianes.

- 5 Las rotaciones de 180° se calculan fácilmente cambiando de signo las componentes real e imaginaria de los datos de entrada, y la de 90° con un cambio de signo e intercambiando dichas componentes.

Habitualmente, a partir de los valores de δ , y una vez aplicadas las rotaciones de 180° y 90° sobre los datos de entrada al rotador, se procede a realizar las microrrotaciones de
10 acuerdo con las ecuaciones:

$$\begin{aligned}x_{i+1} &= x_i - y_i \delta_i 2^{-i} \\y_{i+1} &= y_i + x_i \delta_i 2^{-i}\end{aligned}$$

Sin embargo, es posible modificar este procedimiento para reducir la complejidad del módulo de cálculo de las rotaciones. Con esta idea, se propone un paso en el que se
15 obtiene el vector de rotaciones adaptado, δ' , a partir del vector de rotaciones, δ , de la siguiente forma:

$$\begin{aligned}\delta'_m &= \delta_m \\ \delta'_i &= -\delta_i \cdot \delta_{i-1}, \quad i = m + 1, \dots, M \\ \delta'_{M+1} &= \delta_M\end{aligned}$$

20 donde m es el índice correspondiente a la primera microrrotación y M el correspondiente a la última microrrotación.

Una vez obtenido el vector δ' , el cálculo de las microrrotaciones se realiza de la siguiente forma:

$$\begin{aligned}a_{i+1} &= \begin{cases} a_i + b_i 2^{-i} & \text{si } \delta'_i = -1 \\ b_i + a_i 2^{-i} & \text{si } \delta'_i = 1 \end{cases} \\ b_{i+1} &= \begin{cases} b_i - a_i 2^{-i} & \text{si } \delta'_i = -1 \\ a_i - b_i 2^{-i} & \text{si } \delta'_i = 1 \end{cases}\end{aligned}$$

25

siendo a_i y b_i las entradas de la i -ésima etapa de microrrotación.

Las entradas a la primera etapa de microrrotación, a_m y b_m , valdrán:

$$a_m = x_{IN}$$

$$b_m = y_{IN}$$

siendo x_{IN} e y_{IN} respectivamente la parte real e imaginaria del dato sobre el que hay
5 que realizar las microrrotaciones. Por otra parte, x_{OUT} e y_{OUT} , respectivamente la parte real e imaginaria del dato resultante de aplicar las microrrotaciones, valdrán:

$$x_{OUT} = \begin{cases} a_{M+1} & \text{si } \delta'_{M+1} = -1 \\ b_{M+1} & \text{si } \delta'_{M+1} = 1 \end{cases}$$

$$y_{OUT} = \begin{cases} a_{M+1} & \text{si } \delta'_{M+1} = -1 \\ b_{M+1} & \text{si } \delta'_{M+1} = 1 \end{cases}$$

El módulo de cálculo de las rotaciones de la FFT se representa en la figura 3 y se
10 compone de los siguientes bloques:

- un generador del vector de rotaciones (6),
- un adaptador del vector de rotaciones (7),
- bloques de cálculo de las rotaciones de 180° y 90° (8),
- varios bloques de cálculo de las microrrotaciones (9).

15 El generador del vector de rotaciones (6) determina si hay que rotar 180° y/o 90° el dato de entrada al rotador y obtiene el vector de rotaciones, δ , todo ello a partir del ángulo de rotación, θ , que le llega del módulo de generación de ángulos.

A continuación, el adaptador del vector de rotaciones (7) calcula el vector de rotaciones adaptado, δ' , a partir del vector de rotaciones, δ , de acuerdo con el procedimiento
20 descrito. Desde el punto de vista digital, los valores de δ_i y δ'_i , se representan con un bit, que puede tomar el valor '0' ó '1', en lugar de los valores '-1' y '1' descritos en el procedimiento. Teniendo esto en cuenta, en la implementación de la arquitectura, el procedimiento se transforma en:

$$\delta'_m = \delta_m$$

$$\delta'_i = \delta_i \text{ XOR } \delta_{i-1}, \quad i = m+1, \dots, M$$

$$\delta'_{M+1} = \delta_M$$

Los bloques de cálculo de las rotaciones de 180° y/o 90° (8) rotan el dato de entrada esos ángulos (en sentido negativo) en caso de que sea necesario, de acuerdo con el valor de θ . La rotación de 180° se puede realizar cambiando de signo las componentes real e imaginaria del dato y la de 90° con un cambio de signo e intercambiando dichas
5 componentes.

En la figura 4 se presenta el esquema del bloque de cálculo de las microrrotaciones (9). Las entradas (91) y (92) se corresponden respectivamente con los valores de a_i y b_i descritos en el procedimiento, mientras que el conmutador (93) está controlado por la señal δ'_i (94). Después del conmutador, los datos son desplazados i posiciones (95).
10 Como el valor es fijo para cada etapa de microrrotación, las señales están cableadas, lo que no constituye ningún elemento físico. Finalmente, el circuito se compone de un sumador (96) y de un restador (97).

Así, se consigue simplificar los bloques de cálculo de las microrrotaciones (9) siendo únicamente necesario un conmutador (93), un sumador (96), y un restador (97) en cada
15 uno de ellos.

Como se dijo anteriormente, el dato rotado sale del circuito escalado por un factor constante y, por lo tanto, puede ser compensado. En el caso de $m = 1$ el factor de compensación será:

$$K = \prod_{i=1}^M \cos(\operatorname{tg}^{-1}(2^{-i})) \approx 0,8588$$

20 considerando que $\cos(\operatorname{tg}^{-1}(2^{-i})) \approx 1, \forall i > M$, que es una buena aproximación para $M \geq 7$. Así, teniendo en cuenta que:

$$K = 0,8588 \approx 0,8594 = 1 - 2^{-3} - 2^{-6}$$

la compensación del escalado se puede realizar exclusivamente empleando dos restadores.

25 Por lo tanto, en el caso de que el primer bloque de cálculo de las microrrotaciones sea el correspondiente a $\alpha_1 = \operatorname{tg}^{-1}(2^{-1})$, se puede añadir un módulo adicional de compensación del escalado que consiste en dos restadores.

Breve descripción de los dibujos

Figura 1: Esquema del módulo de generación de ángulos, que incluye un contador (1), lógica combinacional (3), bloques acumuladores (4) y bloques de escalado (5).

Figura 2: Esquema del bloque acumulador (4), que se compone de un sumador (43), un registro (45) y una puerta lógica (44).

Figura 3: Esquema del módulo de cálculo de las rotaciones, que incluye un generador del vector de rotaciones (6), un adaptador del vector de rotaciones (7), bloques de cálculo de las rotaciones de 180° y 90° (8) y bloques de cálculo de las microrrotaciones (9).

Figura 4: Esquema del bloque de cálculo de las microrrotaciones (9), que se compone de un conmutador (93), desplazadores (95), un sumador (96) y un restador (97).

Exposición de un modo de realización de la invención

A continuación se explica la invención para el caso en el que se calculen las rotaciones de una FFT de 8 puntos y *radix* 2. Como $N = 8$, el contador contará cíclicamente de 0 a 7, y se incrementará en cada ciclo de reloj. El número de etapas de la FFT considerada será $n = \log_r N = 3$ y, por lo tanto, $s \in \{1, 2, 3\}$.

La siguiente tabla muestra la forma de obtener las secuencias de rotación θ_s a partir de los valores que va tomando el contador.

Contador		s = 1			s = 2			s = 3		
Dec.	Bin.	p_1	a_1	θ_1	p_2	a_2	θ_2	p_3	a_3	θ_3
0	000	0	0	0	0	0	0	0	0	0
1	001	1	0	0	0	0	0	0	0	0
2	010	0	0	0	1	0	0	0	0	0
3	011	1	0	0	1	1	2	0	0	0
4	100	0	0	0	0	0	0	1	0	0
5	101	1	0	0	0	0	0	1	1	1
6	110	0	0	0	1	0	0	1	2	2
7	111	1	0	0	1	1	2	1	3	3

Las dos primeras columnas de la tabla indican el valor del contador en decimal y en binario respectivamente. Como estamos trabajando con *radix* 2, el valor de p_1 se corresponde con el del bit menos significativo del contador, el de p_2 con el siguiente, y el de p_3 con el más significativo.

- 5 Para obtener la secuencia a_3 se toma el valor de los dos bits menos significativos del contador y se multiplica por p_3 ; para obtener a_2 se multiplica el bit menos significativo del contador por p_2 ; y a_1 vale siempre cero, como debe ocurrir en todas las FFTs para $s = 1$.

Por otra parte, $\theta_s = q_s \cdot a_s$, siendo $q_s = N/r^s$, por lo que $q_1 = 4$, $q_2 = 2$ y $q_3 = 1$.

- 10 A partir de la tabla también se puede entender el funcionamiento de la arquitectura de circuito que lleva a cabo el procedimiento. Así, por ejemplo, la secuencia a_3 empieza valiendo '0' y se incrementa con el valor de p_3 . Cuando el contador llega a 4, como los dos bits menos significativos valen cero, la secuencia se resetea y a continuación se vuelve a incrementar con el valor de p_3 hasta que el contador vuelve a 0, donde se resetea de nuevo la secuencia.

- 15 Para entender el sistema de rotaciones tomaremos el caso en el que $\theta = 3 = B' 011$, que se corresponde con un ángulo en radianes:

$$\theta(rad) = -\frac{2\pi}{N}\theta = -\frac{3\pi}{4}$$

- Sumando a los dos bits más significativos de θ , '01', el siguiente bit, '1', obtenemos el valor '10', por lo que habrá que realizar una rotación de 180° , que situará el ángulo
20 restante en $z = \pi/4$.

A partir de z se obtiene el vector de rotaciones que, en el caso de que $m = 1$ y $M = 8$ valdrá:

$$\delta = 1, 1, 1, -1, 1, -1, -1, 1$$

- 25 A continuación se calcula el vector de rotaciones adaptado de acuerdo con el procedimiento descrito:

$$\begin{aligned}\delta'_m &= \delta_m \\ \delta'_i &= -\delta_i \cdot \delta_{i-1}, \quad i = m+1, \dots, M \\ \delta'_{M+1} &= \delta_M\end{aligned}$$

dando como resultado:

$$\delta' = 1, -1, -1, 1, 1, 1, -1, 1, 1$$

Considerando ahora que el dato de entrada al rotador es el número complejo $10 + 3j$, se aplicará primero la rotación de 180° , con lo que las entradas a la primera etapa de microrrotación (para $m = 1$) serán:

$$a_1 = -10$$

$$b_1 = -3$$

Aplicando el procedimiento de cálculo de las microrrotaciones:

$$a_{i+1} = \begin{cases} a_i + b_i 2^{-i} & \text{si } \delta'_i = -1 \\ b_i + a_i 2^{-i} & \text{si } \delta'_i = 1 \end{cases}$$

$$b_{i+1} = \begin{cases} b_i - a_i 2^{-i} & \text{si } \delta'_i = -1 \\ a_i - b_i 2^{-i} & \text{si } \delta'_i = 1 \end{cases}$$

$$x_{OUT} = \begin{cases} a_{M+1} & \text{si } \delta'_{M+1} = -1 \\ b_{M+1} & \text{si } \delta'_{M+1} = 1 \end{cases}$$

$$y_{OUT} = \begin{cases} a_{M+1} & \text{si } \delta'_{M+1} = -1 \\ b_{M+1} & \text{si } \delta'_{M+1} = 1 \end{cases}$$

siendo δ' el vector de rotaciones adaptado que se ha calculado, y $M = 8$, se obtienen los valores de salida:

$$x_{OUT} = -5,80$$

$$y_{OUT} = -10,69$$

Se puede comprobar fácilmente que el ángulo que se ha rotado el dato de entrada es:

$$\theta(rad) = -2,3593 \approx -\frac{3\pi}{4}$$

Por otra parte, el módulo del dato de salida está escalado respecto al módulo del dato de entrada, y es posible compensarlo multiplicando la salida por $K = 1 - 2^{-3} - 2^{-6}$.

También es posible realizar la compensación del escalado a la entrada de la FFT o en cualquier punto intermedio de la FFT.

Para realizar la compensación se tomará el dato que hay que multiplicar por K , el mismo dato desplazado 3 bits y finalmente el dato desplazado 6 bits. A continuación se restarán del dato sin desplazar los valores desplazados. Así, como los desplazamientos de bits son fijos, únicamente hacen falta dos restadores para llevar a cabo la compensación del escalado.

Aplicación industrial

Debido a que la FFT es un algoritmo ampliamente utilizado en las tecnologías de la información y las comunicaciones, la invención presentada puede tener aplicación en numerosos sistemas de comunicaciones y de procesamiento de señal. En especial, la arquitectura propuesta resulta útil cuando se pretende construir FFTs en plataformas *hardware*, como pueden ser las FPGAs (*Field Programmable Gate Array*). Por otra parte, la invención aporta una gran ventaja respecto a otros diseños en aplicaciones donde es necesario calcular FFTs de un elevado número de puntos.

Referencias

- [CP03] Yun-Nan Chang and Keshab K. Parhi. An efficient pipelined FFT architecture.. *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, 50:322–325, Jun 2003.
- 5 [CT65] J. W. Cooley and J.W Tukey. An algorithm for machine computation of complex fourier series. *Math. Computation*, 19:297–301, Apr. 1965.
- [Hu92] Y. H. Hu. CORDIC-based VLSI Architectures for Digital Signal Processing. *IEEE Signal Processing Magazine*, 9, Jul 1992.
- [MGBS02] K. Maharatna, E. Grass, S. Banerjee, and A. Sundar. CORDIC UNIT. *Patente*
10 *US 20060059215A1*, 20.12. 2002.
- [Vol59] J.E. Volder. The CORDIC trigonometric computing technique. *IRE Trans. Electronic Computers*, EC-8:330–334, Sep. 1959.
- [YCC06] Cheng-Ying Yu, Sau-Gee Chen, and Jen-Chuan Chih. Efficient CORDIC Designs for Multi-Mode OFDM FFT. *Proc. IEEE International Conference on*
15 *Acoustics, Speech and Signal Processing*, 3:1036–1039, May 2006.
- [ZH05] Li Zou and Xiao Huang. 3780-point Discrete Fourier Transformation processor. *Patente EP 1750206A1*, 04.08. 2005.

Reivindicaciones

1. Procedimiento para el cálculo de las rotaciones de cualquier FFT descompuesta según el algoritmo Cooley-Tukey, y cuyo número de puntos, N , y *radix*, r , son ambos potencia de 2, con los siguientes pasos:

- 5 ■ obtener las secuencias de ángulos de rotación de cada una de las etapas de la FFT,
- calcular las rotaciones de cada una de las etapas de la FFT a partir de los ángulos de rotación generados,

caracterizado por que a partir de un único contador se obtienen las secuencias de ángulos de rotación de todas las etapas de la FFT, y para una etapa cualquiera, s , de la FFT, la generación de la secuencia de ángulos de rotación de dicha etapa, θ_s , se realiza siguiendo los siguientes pasos:

- tomar los $s \cdot \log_2 r$ bits menos significativos proporcionados por el contador,
- de los bits tomados, multiplicar los $(s - 1) \cdot \log_2 r$ bits menos significativos por los $\log_2 r$ más significativos, obteniendo así los valores de la secuencia a_s ,
- 15 ■ multiplicar el resultado por N/r^s , obteniendo así los valores de la secuencia θ_s .

2. Procedimiento según la reivindicación 1, caracterizado por que para el cálculo de las rotaciones de cada una de las etapas de la FFT, se obtiene el vector de rotaciones adaptado, δ' , a partir del vector de rotaciones, δ , de la siguiente forma:

$$\begin{aligned} \delta'_m &= \delta_m \\ \delta'_i &= -\delta_i \cdot \delta_{i-1}, \quad i = m + 1, \dots, M \\ \delta'_{M+1} &= \delta_M \end{aligned}$$

donde m es el índice correspondiente a la primera microrrotación y M el correspondiente a la última microrrotación.

25 3. Procedimiento según las reivindicaciones 1 a 2, caracterizado por que para el cálculo de las rotaciones de cada una de las etapas de la FFT, el cálculo de las microrrotaciones se realiza de la siguiente forma:

$$a_{i+1} = \begin{cases} a_i + b_i 2^{-i} & \text{si } \delta'_i = -1 \\ b_i + a_i 2^{-i} & \text{si } \delta'_i = 1 \end{cases}$$

$$b_{i+1} = \begin{cases} b_i - a_i 2^{-i} & \text{si } \delta'_i = -1 \\ a_i - b_i 2^{-i} & \text{si } \delta'_i = 1 \end{cases}$$

siendo a_i y b_i las entradas de la i -ésima etapa de microrrotación.

4. Arquitectura de circuito para implementar el procedimiento descrito en las reivindicaciones 1 a 3, caracterizada por que comprende:

- un módulo de generación de ángulos que incluye un único contador (1) para toda la FFT, a partir del cual se obtienen los ángulos de rotación de todas las etapas de la FFT sin necesidad de recurrir a ningún dato almacenado previamente,
- un módulo de cálculo de las rotaciones para cada una de las etapas de la FFT.

5. Arquitectura de circuito según la reivindicación 4, caracterizada por que el módulo de generación de ángulos comprende, además del contador (1), los siguientes elementos:

- lógica combinacional (3), que opera sobre el valor del contador,
- un bloque acumulador (4) por cada etapa de la FFT, para el cálculo de las secuencias a_s .

6. Arquitectura de circuito según las reivindicaciones 4 y 5, caracterizada por que, dentro del módulo de generación de ángulos, cada bloque acumulador (4) comprende los siguientes elementos:

- un sumador (43), que permite ir actualizando los valores de la secuencia a_s ,
- un registro (45) que proporciona los valores de la secuencia a_s ,
- una puerta lógica (44) para el control del bloque acumulador.

7. Arquitectura de circuito según las reivindicaciones 4 a 6, caracterizada por que el módulo de cálculo de las rotaciones de la FFT, comprende bloques de cálculo de las microrrotaciones (9) que consisten en un conmutador (93), un sumador (96), y un restador (97).

8. Arquitectura de circuito para el cálculo de las rotaciones de la FFT según las reivindicaciones 4 a 7, caracterizada por que en el caso de que el primer bloque de cálculo de las microrrotaciones sea el correspondiente a $\alpha_1 = \text{tg}^{-1}(2^{-1})$, se puede añadir un módulo adicional de compensación del escalado que consiste en dos restadores.

1 / 2

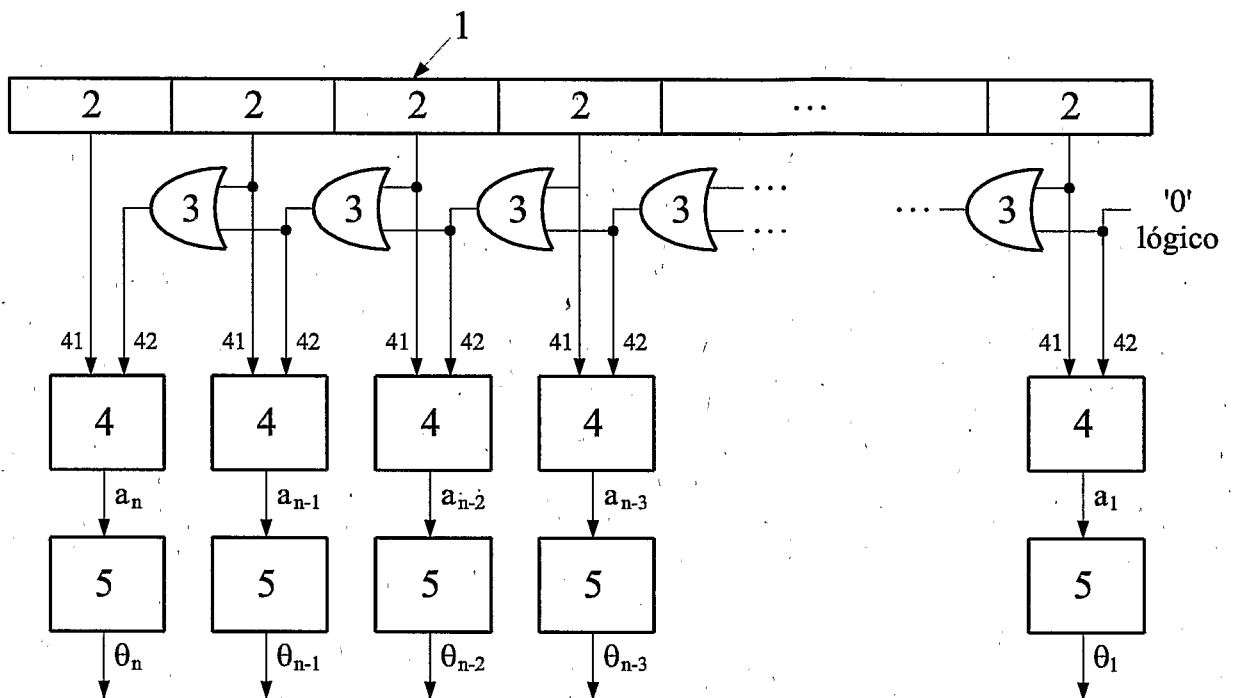
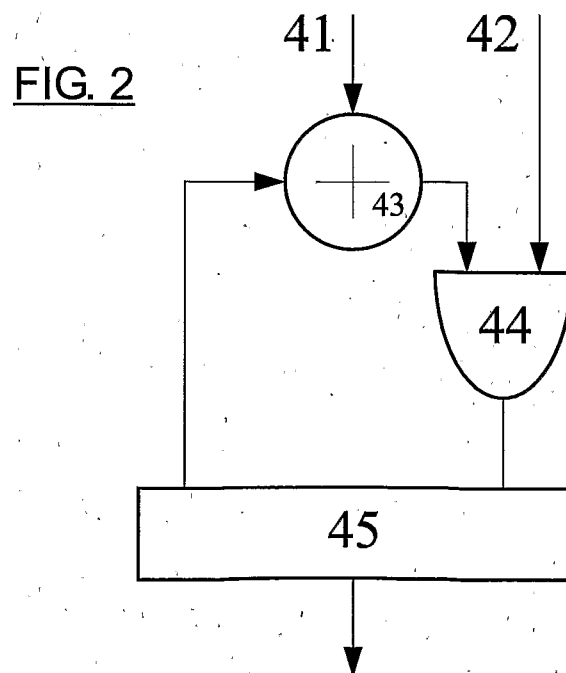


FIG. 1



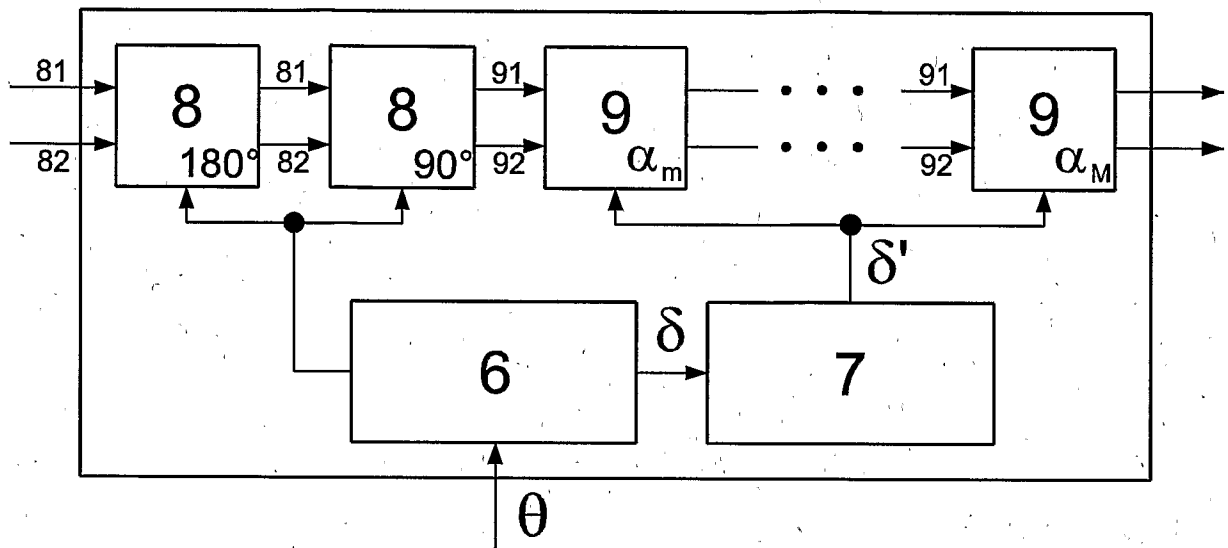


FIG. 3

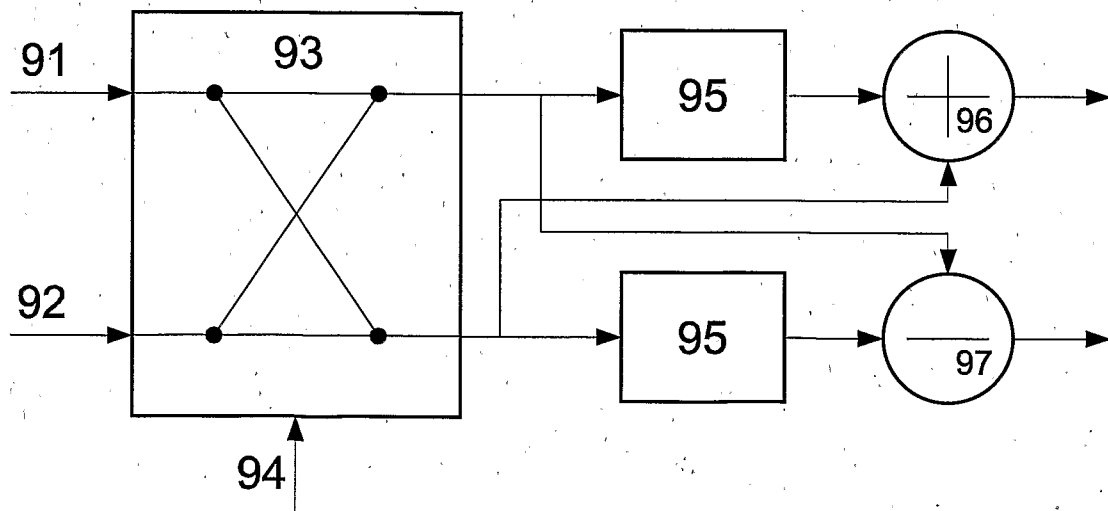


FIG. 4

INTERNATIONAL SEARCH REPORT

International application No.
PCT/ ES 2008/000220

A. CLASSIFICATION OF SUBJECT MATTER

G06F 17/14 (2006.01)

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F17/14

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

INVENES, EPODOC, WPI, XPESP, XPAIP, NPL, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2005216540 A1 (WEN et al.) 29.09.2005, the whole document.	1-8
A	Cheng-Ying Yu; San-Gee Chen; Jen-Chuan Chih."Efficient CORDIC Designs for Multi-Mode OFDM FFT", Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on Volume 3, 14-19 May 2006 Page(s):III - III [online][retrieved on 22.07.2008] Retrieved from the internet: <URL: http://ieeexplore.ieee.org/iel5/11024/34759/01660834.pdf?tp=&arnumber=1660834&isnumber=34759 >	1-8

☒ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A" document defining the general state of the art which is not considered to be of particular relevance.	
"E" earlier document but published on or after the international filing date	
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"O" document referring to an oral disclosure use, exhibition, or other means	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other documents, such combination being obvious to a person skilled in the art
"P" document published prior to the international filing date but later than the priority date claimed	
	"&" document member of the same patent family

Date of the actual completion of the international search

10 September 2008 (10.09.2008)

Date of mailing of the international search report

(16/09/2008)

Name and mailing address of the ISA/
O.E.P.M.

Paseo de la Castellana, 75 28071 Madrid, España.

Facsimile No. 34 91 3495304

Authorized officer

J. Botella Maldonado

Telephone No. +34 91 349 53 82

INTERNATIONAL SEARCH REPORT

International application No.

PCT/ES 2008/000220

C (continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of documents, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Dick, C.."Computing the Discrete Fourier Transform on FPGA Based Systolic Arrays", Field-Programmable Gate Arrays, 1996. FPGA '96. Proceedings of the 1996 ACM Fourth International Symposium on 1996 Page(s):129 - 135 [online][retrieved on 22.07.2008] Retrieved from the internet: <URL:http://ieeexplore.ieee.org/iel5/9475/30061/01377297.pdf?tp=&arnumber=1377297&isnumber=30061>	1-8
A	Groginsky, H.L.; Works, G.A.. "A Pipeline Fast Fourier Transform" Computers, IEEE Transactions on Volume C-19, Issue 11, Nov. 1970 Page(s):1015 - 1019 [on line][retrieved on 22.07.2008] Retrieved from the internet: <URL:http://ieeexplore.ieee.org/iel5/12/35041/01671419.pdf?tp=&arnumber=1671419&isnumber=35041>	1-8
A	Sung, T.-Y.. "Memory-efficient and high-speed split-radix FFT/ IFFT processor based on pipelined CORDIC rotations", Vision, Image and Signal Processing, IEE Proceedings -Volume 153, Issue 4, August 2006 Page(s):405 -410 [online][retrieved on 22.07.2008] Retrieved from the internet: <URL:http://ieeexplore.ieee.org/iel5/2200/35271/01677109.pdf?tp=&arnumber=1677109&isnumber=35271>	1-8

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/ ES 2008/000220

Patent document cited in the search report	Publication date	Patent family member(s)	Publication date
US 2005216540 A	29.09.2005	NONE	-----
-----	-----	-----	-----

INFORME DE BÚSQUEDA INTERNACIONAL

Solicitud internacional N°
PCT/ ES 2008/000220

A. CLASIFICACIÓN DEL OBJETO DE LA SOLICITUD

G06F 17/14 (2006.01)

De acuerdo con la Clasificación Internacional de Patentes (CIP) o según la clasificación nacional y CIP.

B. SECTORES COMPRENDIDOS POR LA BÚSQUEDA

Documentación mínima buscada (sistema de clasificación seguido de los símbolos de clasificación)

G06F17/14

Otra documentación consultada, además de la documentación mínima, en la medida en que tales documentos formen parte de los sectores comprendidos por la búsqueda

Bases de datos electrónicas consultadas durante la búsqueda internacional (nombre de la base de datos y, si es posible, términos de búsqueda utilizados)

INVENES, EPODOC, WPI, XPESP, XPAIP, NPL, INSPEC

C. DOCUMENTOS CONSIDERADOS RELEVANTES

Categoría*	Documentos citados, con indicación, si procede, de las partes relevantes	Relevante para las reivindicaciones N°
A	US 2005216540 A1 (WEN et al.) 29.09.2005, todo el documento.	1-8
A	Cheng-Ying Yu; San-Gee Chen; Jen-Chuan Chih."Efficient CORDIC Designs for Multi-Mode OFDM FFT", Acoustics, Speech and Signal Processing, 2006. ICASSP 2006 Proceedings. 2006 IEEE International Conference on Volume 3, 14-19 May 2006 Page(s):III - III [en línea][recuperado el 22.07.2008] Recuperado de internet: <URL:http://ieeexplore.ieee.org/iel5/11024/34759/01660834.pdf?tp=&arnumber=1660834&isnumber=34759>	1-8

☒ En la continuación del Recuadro C se relacionan otros documentos ☒ Los documentos de familias de patentes se indican en el Anexo

* Categorías especiales de documentos citados:	"T"	documento ulterior publicado con posterioridad a la fecha de presentación internacional o de prioridad que no pertenece al estado de la técnica pertinente pero que se cita por permitir la comprensión del principio o teoría que constituye la base de la invención.
"A" documento que define el estado general de la técnica no considerado como particularmente relevante.	"X"	documento particularmente relevante; la invención reivindicada no puede considerarse nueva o que implique una actividad inventiva por referencia al documento aisladamente considerado.
"E" solicitud de patente o patente anterior pero publicada en la fecha de presentación internacional o en fecha posterior.	"Y"	documento particularmente relevante; la invención reivindicada no puede considerarse que implique una actividad inventiva cuando el documento se asocia a otro u otros documentos de la misma naturaleza, cuya combinación resulta evidente para un experto en la materia.
"L" documento que puede plantear dudas sobre una reivindicación de prioridad o que se cita para determinar la fecha de publicación de otra cita o por una razón especial (como la indicada).	"&"	documento que forma parte de la misma familia de patentes.
"O" documento que se refiere a una divulgación oral, a una utilización, a una exposición o a cualquier otro medio.		
"P" documento publicado antes de la fecha de presentación internacional pero con posterioridad a la fecha de prioridad reivindicada.		

Fecha en que se ha concluido efectivamente la búsqueda internacional.

10 Septiembre 2008 (10.09.2008)

Fecha de expedición del informe de búsqueda internacional

16 de Septiembre de 2008 (16/09/2008)

Nombre y dirección postal de la Administración encargada de la búsqueda internacional
O.E.P.M.

Paseo de la Castellana, 75 28071 Madrid, España.
N° de fax 34 91 3495304

Funcionario autorizado

J. Botella Maldonado

N° de teléfono +34 91 349 53 82

INFORME DE BÚSQUEDA INTERNACIONAL

Solicitud internacional N°
PCT/ES 2008/000220

C (continuación). DOCUMENTOS CONSIDERADOS RELEVANTES		
Categoría*	Documentos citados, con indicación, si procede, de las partes relevantes	Relevante para las reivindicaciones N°
A	Dick, C.."Computing the Discrete Fourier Transform on FPGA Based Systolic Arrays", Field- Programmable Gate Arrays, 1996. FPGA '96. Proceedings of the 1996 ACM Fourth International Symposium on 1996 Page(s):129 - 135 [en línea][recuperado el 22.07.2008] Recuperado de internet: <URL:http://ieeexplore.ieee.org/iel5/9475/30061/01377297.pdf?tp=&arnumber=1377297&isnumber=30061>	1-8
A	Groginsky, H.L.; Works, G.A.. "A Pipeline Fast Fourier Transform" Computers, IEEE Transactions on Volume C-19, Issue 11, Nov. 1970 Page(s):1015 - 1019 [en línea][recuperado el 22.07.2008] Recuperado de internet: <URL:http://ieeexplore.ieee.org/iel5/12/35041/01671419.pdf?tp=&arnumber=1671419&isnumber=35041>	1-8
A	Sung, T.-Y.. "Memory-efficient and high-speed split-radix FFT/ IFFT processor based on pipelined CORDIC rotations", Vision, Image and Signal Processing, IEE Proceedings - Volume 153, Issue 4, August 2006 Page(s):405-410 [en línea][recuperado el 22.07.2008] Recuperado de internet: <URL:http://ieeexplore.ieee.org/iel5/2200/35271/01677109.pdf?tp=&arnumber=1677109&isnumber=35271>	1-8

INFORME DE BÚSQUEDA INTERNACIONAL

Información relativa a miembros de familias de patentes

Solicitud internacional N°

PCT/ES 2008/000220

Documento de patente citado en el informe de búsqueda	Fecha de Publicación	Miembro(s) de la familia de patentes	Fecha de Publicación
US 2005216540 A	29.09.2005	NINGUNO	-----
