

01170  
2  
207

**DIVISION DE ESTUDIOS DE POSGRADO**  
**Facultad de Ingeniería**

**Sistema para el procesamiento en  
tiempo real de Audio Estereo  
usando el DSP TMS320C25**

**TESIS**

**PRESENTADA A LA DIVISION DE ESTUDIOS DE  
POSGRADO**

**FACULTAD DE INGENIERIA**

**DE LA**

**UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO**

**COMO REQUISITO PARA OBTENER EL GRADO DE**

**MAESTRO EN INGENIERIA ELECTRICA  
(ELECTRONICA)**

**CIUDAD UNIVERSITARIA**

**Félix Cuéllar de la Peña**

**1993**

**TESIS CON  
FALLA DE ORIGEN**



## **UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso**

### **DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

# Indice

<b>1</b>	<b>Introducción</b>	<b>1-1</b>
<b>2</b>	<b>Objetivos y Presentación del Problema</b>	<b>2-1</b>
	2.1 Objetivos	2-1
	2.2 Presentación del Problema	2-1
<b>3</b>	<b>La Transformada Discreta de Fourier y sus diferentes implantaciones</b>	<b>3-1</b>
	3.1 Introducción	3-1
	3.2 Transformada de Fourier	3-1
	3.2.1 Obtención del Algoritmo FFT para un DSP TMS320	3-1
	3.2.1.1 Mariposas de la FFT con DIT Radix-2	3-4
	3.2.1.2 Implementación de la mariposa FFT con Escalamiento	3-6
	3.2.1.3 Procesamiento de la FFT en su mismo espacio	3-11
	3.2.1.4 Obtención de Reversión de Bits	3-12
	3.2.1.5 Mariposas Especiales	3-13
	3.2.2 Mariposas de la FFT con DIF Radix-4	3-14
	3.3 La Transformada Rápida de Hartley	3-17
	3.3.1 Obtención de la Transformada de Hartley	3-18
	3.3.2 El Algoritmo Rápido	3-19
<b>4</b>	<b>Principios de Audio Digital</b>	<b>4-1</b>
	4.1 Introducción	4-1
	4.2 Muestreo Discreto en el Tiempo	4-1
	4.2.1 La Naturaleza Sin Perdidas del Muestreo	4-2
	4.2.2 El Teorema de Muestreo	4-4
	4.3 El Fenómeno de Aliasing	4-7
	4.3.1 Frecuencias de Traslape	4-7
	4.3.2 Imágenes y Aliasing	4-9
	4.3.3 Armónicas y Aliasing	4-10
	4.3.4 Solución para el Fenómeno de Aliasing	4-11
	4.4 Cuantización	4-12
	4.4.1 Aproximación Analógica y Digital	4-12

4.4.2 Razón Señal a Error	4-14
4.4.3 Distorsión en la Cuantización	4-17
4.5 Dither	4-19
<b>5 Procesadores Digitales de Señales</b>	<b>5-1</b>
5.1 Antecedentes	5-1
5.2 Historia de Soluciones DSP	5-4
5.3 DSP de la Familia TMS de Texas Instruments	5-4
5.3.1 La Arquitectura TMS320 Básica	5-5
5.3.1.1 Primer Generación de la Familia TMS320	5-6
5.3.1.2 Segunda Generación de la Familia TMS320	5-7
5.3.1.3 Tercera Generación de la Familia TMS320	5-8
5.3.2 Herramientas de Desarrollo y Soporte	5-8
5.3.2.1 Herramientas de Software	5-9
5.3.2.2 Herramientas de Hardware	5-9
5.4 La Arquitectura del TMS320C25	5-9
<b>6 Propuesta de una nueva implementación de la FHT</b>	<b>6-1</b>
6.1 Introducción	6-1
6.2 Elaboración de las Mariposas	6-2
6.2.1 Primera Mariposa	6-3
6.2.2 Segunda Mariposa	6-5
6.2.3 Tercera Mariposa	6-6
6.2.4 Cuarta a Séptima Mariposa	6-10
6.3 Reordenamiento de los Resultados de la Transformada	6-14
6.4 Captura de Datos	6-16
6.4.1 Analógico-Digital	6-16
6.4.2 Digital-Analógico	6-17
6.5 Diferencias entre los Algoritmos A/D y D/A	6-19
6.5.1 Diferencias de Hardware	6-19
6.5.2 Diferencias de Software	6-19
6.6 Descripción General del Algoritmo FHT Implementado	6-20
6.7 Características de la FHT en la Práctica	6-20
<b>7 Diseño y Desarrollo de una Arquitectura para PAD</b>	<b>7-1</b>
7.1 Introducción	7-1
7.2 Características de la Arquitectura	7-1
7.3 Características del ADC DSP56ADC16	7-2
7.4 Operación del Módulo de Transformación Directa	7-3
7.4.1 Consideraciones del Diseño	7-3
7.4.2 Operación del Circuito	7-3
7.4.2.1 Primera Etapa	7-6
7.4.2.2 Segunda Etapa	7-6

7.4.2.3 Tercera Etapa	7-8
7.4.2.4 Cuarta Etapa	7-8
7.5 Operación del Modulo de Transformada Inversa	7-10
7.5.1 Consideraciones del Diseño	7-10
7.5.2 Operación del Circuito	7-10
7.5.2.1 Primera Etapa	7-12
7.5.2.2 Segunda Etapa	7-12
7.5.2.3 Tercera Etapa	7-14
<b>8 Presentación de Resultados y Evaluación</b>	<b>8-1</b>
8.1 Introducción	8-1
8.2 Evaluación del Algoritmo FFT para la solución del proyecto	8-1
8.3 Software	8-3
8.3.1 Software de Simulación en TurboPascal 4	8-4
8.3.2 Software del TMS320 en el Simulador	8-7
8.4 Emulación por Hardware	8-7
8.5 Evaluación de la FHT	8-11
8.5.1 Obtención de la FFT a partir de la FHT	8-12
<b>9 Conclusiones y Perspectivas</b>	<b>9-1</b>
9.1 Conclusiones	9-1
9.1.1 Problemas en la elaboración del Proyecto	9-2
9.2 Perspectivas	9-2
<b>10 Bibliografía</b>	<b>10-1</b>
<b>11 Glosario</b>	<b>11-1</b>

## Apéndices

<b>A FFT Radix-4 para el TMS320C25</b>	<b>A-1</b>
<b>B FHT Radix-2 en TurboPascal 4.0</b>	<b>B-1</b>
<b>C Resultados Simulación FHT en TurboPascal 4.0</b>	<b>C-1</b>
C.1 FHT Directa	C-1
C.2 FHT Inversa	C-6
<b>D FHT Radix-2 para el TMS320C25</b>	<b>D-1</b>

<b>E</b> FHT(Inversa) Radix-2 para el TMS320C25	E-1
<b>F</b> Circuito con el TMS320C25 para la FHT Directa	F-1
<b>G</b> Circuito con el TMS320C25 para la FHT Inversa	G-1
<b>H</b> Resultados Simulación FHT en TurboPascal 4.0	H-1
H.1 FHT Directa	H-1
H.2 FHT Inversa	H-4
<b>I</b> Resultados Simulador TMS320C25	I-1
I.1 FHT Directa	I-1
I.2 FHT Inversa	I-4
<b>J</b> Resultados Emulador TMS320C25	J-1
J.1 FHT Directa	J-1
J.2 FHT Inversa	J-4

## **Anexos**

<b>i</b> FHT en Formato de Punto Flotante	i-1
---	-----

*Les agradezco a mis padres el apoyo  
incondicional para todo y siempre.  
Gracias Rogelio y Miguel por su  
asesoramiento, guía, paciencia y  
motivación constante.*

*Y a tí, Ma. Antonieta gracias por  
ser la luz e inspiración de mi vida.*

*A veces no se encuentran las  
palabras adecuadas para expresar los  
sentimientos, gracias a todos por la fe  
que han tenido en mí.*

# 1

## Introducción

Con el ritmo acelerado de los avances en el campo de la electrónica en las áreas de arquitectura computacional y arquitecturas especializadas, hoy en día es posible aplicar con facilidad conceptos y estructuras que hace algunos años sólo era posible encontrarlos en grandes centros especializados, que además de su alto costo, su acceso estaba restringido a un número reducido de individuos.

Una de tantas áreas que se ha beneficiado de estos avances tecnológicos es la del Procesamiento Digital de Señales (DSP) que demanda una alta capacidad de adquisición de información, dispositivos de almacenamiento, complejidad de cálculo, grandes velocidades, etc. Todos estos requisitos se han ido cubriendo con la ayuda de arquitecturas especializadas, y con el manejo adecuado de los equipos y herramientas computacionales.

El medio ambiente se encuentra saturado de señales, las cuales son adquiridas por una extensa variedad de sensores, procesadas y almacenadas en el mismo instante que su contenido es interpretado para dar una respuesta apropiada. Para realizar un sistema capaz de procesar señales es necesario poder imitar las funciones que realiza el ser humano.

El objetivo de este proyecto es crear un sistema de adquisición, procesamiento y devolución de señales de audio a través de convertidores analógico-digital, que posteriormente, se manipulará para obtener la componente en frecuencia de las señales para un manejo de información más útil; así mismo se elaborará el camino inverso.

Para el desarrollo del sistema se tomó como base la transformada de Fourier la cual convierte información en el dominio del tiempo a el dominio de la frecuencia. Es una herramienta analítica en campos como acústica, óptica, sismología, telecomunicaciones, voz, procesamiento de señales y procesamiento de imágenes.

La utilización de métodos digitales de grabación, reproducción y almacenamiento de señales de audio digital incluyen varios conceptos ajenos a métodos analógicos. Los sistemas digitales tienen poca similitud a sistemas analógicos. El audio es analógico por naturaleza, los sistemas digitales utilizan muestreo y cuantización, que son los pilares de la digitalización

de audio, para transformar la información de audio. Se tienen que tomar precauciones para evitar dos tipos de distorsiones: una condición de frecuencias erróneas llamada aliasing y el error introducido por la cuantización de la señal analógica.

La implementación del proyecto consta de la elaboración de un algoritmo utilizando la transformada de Hartley la cual es más rápida y fácil de implementar que la de Fourier. Con base a la transformada y a la elección de un DSP comercial la arquitectura se desarrolló de tal forma que fuera sencilla y flexible.

El algoritmo que se creó fue comparado con varias implementaciones de varios tipos de transformadas y superó a las mismas en cuanto a su velocidad, por espacio de código y cantidad de hardware necesaria para cubrir las necesidades del sistema. En cuanto a la exactitud que se logró en la implementación, esta no excede un error del 0.05 % a comparación de la implementación realizada en un microcomputadora con operandos de punto flotante.

Las perspectivas de utilización del sistema son amplias debido a el ancho de banda que utiliza y a la flexibilidad de su arquitectura, entre algunas de ellas se puede mencionar: filtrado digital, reconocimiento de voz, implementaciones en multimedia, edición de audio en compañías de música, en equalización de bandas de equipos de sonido comerciales, modulación de sintetizadores, etc.

Como conclusión el sistema que se desarrolló cubre las necesidades de audio digital incrementando la eficiencia del algoritmo de Hartley y economiza al limitar el número de componentes de hardware, esto sin incrementar la complejidad de análisis de la misma y agregando una enorme flexibilidad para expansión del sistema según los requerimientos de aplicaciones particulares.

En el capítulo 2 se plantean los objetivos globales del proyecto y los problemas para su implementación.

En el capítulo 3 se presenta la teoría necesaria para el desarrollo de los algoritmos de transformación elaborados para el Procesador Digital de Señales (DSP) de la familia de Texas Instruments (TI). Así mismo se comparan varios tipos de algoritmos concluyendo cual es el adecuado para la solución del sistema.

En el capítulo 4 se incursiona en los conceptos y fundamentos de audio digital. Los sistemas digitales tienen poca similitud a sistemas analógicos, como el audio es analógico por naturaleza, los sistemas digitales utilizan muestreo y cuantización, que son los pilares de la digitalización de audio, para transformar la información de audio. Se tienen que tomar precauciones para evitar dos tipos de distorsiones: una condición de frecuencias erróneas llamada aliasing y el error introducido por la cuantización de la señal analógica.

En el capítulo 5 se proporciona una breve evolución histórica de los DSP extendiéndose a las características de la familia de TI.

En el capítulo 6 se describe la elaboración de cada una de las partes del algoritmo FHT y su creación en el lenguaje ensamblador para el DSP.

En el capítulo 7 se explica la arquitectura en general así como también cada una de sus partes funcionales.

En el capítulo 8 se presenta una evaluación del sistema y los resultados obtenidos mediante comparaciones bibliográficas, simuladores de hardware, simulaciones en software, emuladores de hardware e implementación del circuito.

En el capítulo 9 se presenta las conclusiones y perspectivas del sistema.

# 2

## Objetivos y Presentación del Problema

### 2.1 Objetivos

El medio ambiente se encuentra saturado de señales, las cuales son adquiridas por una extensa variedad de sensores, procesadas y almacenadas en el mismo instante que su contenido es interpretado para dar una respuesta apropiada e inmediata. Lo anterior plantea: ¿Cuál es el sistema capaz de agrupar todas las características anteriores a tan alta velocidad?

La respuesta no es la de un proyecto multimillonario llevado a cabo por los mejores científicos del mundo, sino uno realizado por la naturaleza desde hace miles de años, los seres vivos.

De lo anterior se concluye que para realizar un sistema capaz de procesar señales es necesario poder imitar las funciones que realiza el ser humano.

El objetivo de este proyecto es crear un sistema de adquisición, procesamiento y devolución de señales de audio en modo estéreo en el tiempo a través de convertidores analógico-digital y digital-analógico, que posteriormente, mediante una arquitectura especializada basada en un DSP (Digital Signal Processor) comercial obtendrá la transformada de frecuencia de las señales para un manejo de información más útil; así mismo se elaborará el camino inverso, de tal manera que se posea un sistema completo.

### 2.2 Presentación del Problema

Una vez establecidos los objetivos del proyecto es necesario evaluar los recursos con que se cuenta para llevar a cabo el sistema planteado. De una evaluación correcta de las

limitaciones y ventajas de cada componente seleccionado dependerá que la estructura del sistema llegue a ser eficiente.

Primero hay que evaluar las características de las señales con las que se propone trabajar y procesar, entre ellas su amplitud, frecuencia, espectros, etc. En segundo término evaluar los elementos electrónicos, circuitos integrados, sensores, herramientas de diseño y simulación disponibles o en su defecto la posibilidad de adquisición, para la realización de la arquitectura especializada con el propósito de dar solución a los problemas de obtención, manejo y procesamiento de las señales.

Una vez que se ha llegado a un entendimiento de las tareas que se deben de llevar a cabo para que el sistema trabaje exitosamente se puede percatar que la necesidad de optimizar las rutinas de software de adquisición, lectura, procesamiento y almacenamiento de la información dentro del sistema especializado es de suma importancia si se quiere conservar un alto rendimiento en la relación de muestras o, en otras palabras, del ancho de banda a manejar. Otra consideración importante es diseñar una arquitectura eficiente y de bajo costo, factible de construir.

# 3

## La Transformada Discreta de Fourier y sus diferentes Implementaciones

### 3.1 Introducción

La transformada de Fourier convierte información en el dominio del tiempo al dominio de la frecuencia. Es una herramienta analítica en campos como acústica, óptica, sismología, telecomunicaciones, voz, procesamiento de señales y procesamiento de imágenes. En sistemas de tiempo discretos, la Transformada Discreta de Fourier (DFT) es la contraparte de la Transformada Continua de Fourier. Como la DFT es computacionalmente intensiva tenía relativamente pocas aplicaciones, inclusive con computadoras actuales. La Transformada Rápida de Fourier (FFT) es el nombre genérico para una clase eficiente de algoritmo computacional que implementa la DFT y es utilizada en el campo de los Procesadores Digitales de Señales (DSP).

### 3.2 Transformada de Fourier

#### 3.2.1 Obtención del Algoritmo FFT para un DSP TMS320

Un método más eficiente de realizar el cálculo computacional de la DFT que reduce significativamente el número requerido de operaciones aritméticas es la decimación en tiempo (DIT). En la FFT,  $N$  es un número factorizable que permite a la DFT de  $N$  puntos el ser reducida en transformadas más pequeñas. El tamaño de la transformada más pequeña que se puede obtener se conoce como el radix de el algoritmo FFT. Así, por ejemplo para un algoritmo FFT radix-2, la más pequeña transformación o "mariposa" (unidad básica

computacional) utilizada es una DFT de 2 puntos. Generalmente, para una FFT de N puntos, existen N muestras resultantes de frecuencia correspondiendo a N muestras de la señal de entrada  $x(n)$ . Para una FFT radix-2, N es una potencia de 2.

El número de operaciones aritméticas puede ser reducido inicialmente mediante la descomposición de la DFT de N puntos en dos DFT de N/2 puntos. Esto significa que la secuencia de tiempo de entrada  $x(n)$  esta reducida en dos subsecuencias de N/2 puntos, que están constituidas por sus muestras de numeración par y de numeración non, con índices expresados matemáticamente como  $2n$  y  $2n+1$ , respectivamente. Sustituyendo estos índices de tiempo en la ecuación original de la DFT

$$\begin{aligned}
 X(k) &= \sum_{n=0}^{N/2-1} x(2n) W_N^{2nk} + \sum_{n=0}^{N/2-1} x(2n+1) W_N^{(2n+1)k} \\
 &= \sum_{n=0}^{N/2-1} x(2n) W_N^{2nk} + W_N^k \sum_{n=0}^{N/2-1} x(2n+1) W_N^{2nk}
 \end{aligned} \tag{1}$$

Como

$$W_N^2 = [e^{-j(2\pi/N)}]^2 = [e^{-j\pi/(N/2)}]^2 = W_{N/2}$$

la ecuación (1) puede ser escrita como

$$\begin{aligned}
 X(k) &= \sum_{n=0}^{N/2-1} x(2n) W_{N/2}^{nk} + W_N^k \sum_{n=0}^{N/2-1} x(2n+1) W_{N/2}^{nk} \\
 &= Y(k) + W_N^k Z(k) \quad k = 0, 1, \dots, N-1
 \end{aligned} \tag{2}$$

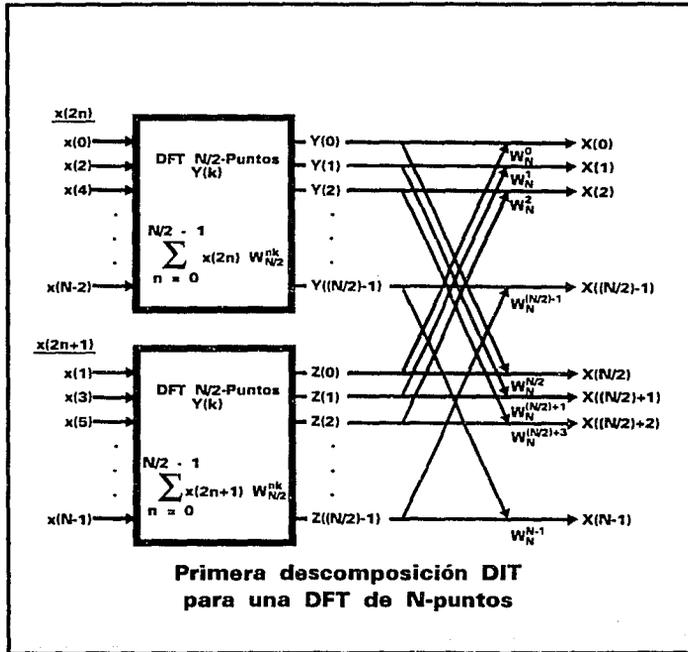
donde  $Y(k)$  es el primer término de la sumatoria y  $Z(k)$  es el segundo término de la sumatoria.

$Y(k)$  y  $Z(k)$  también se pueden observar como la DFT de N/2 puntos de las muestras con numeración par y numeración non, respectivamente. En este caso, el número de multiplicaciones y sumas complejas es aproximadamente  $N + 2(N/2)^2$  porque, de acuerdo a (2), la DFT de N puntos es dividida en dos DFT de N/2 puntos, las cuales son combinadas con N multiplicaciones y sumas complejas. Por lo que, con la división de la DFT de N puntos original en dos DFT de N/2 puntos, el número total de operaciones aritméticas se reduce. Esta reducción es ilustrada en la figura 1.

En la anterior derivada se encuentra implícita la periodicidad de  $X(k)$ ,  $Y(k)$ , y  $Z(k)$ .  $X(k)$  es periódica en k con periodo N, mientras que  $Y(k)$  y  $Z(k)$  son ambas periódicas en k con periodo N/2. Por consiguiente, aunque k varíe sobre N valores desde 0 hasta N-1 para

$X(k)$ , ambas  $Y(k)$  y  $Z(k)$  tienen que ser calculadas para  $k$  entero 0 y  $(N/2)-1$  solamente. La periodicidad de  $Y(k)$  y  $Z(k)$  es también asumida en la figura 1.

Figura 1



La ecuación (2) para  $N/2 \leq k \leq N-1$  puede ser reescrita como

$$X(k+N/2) = \sum_{n=0}^{N/2-1} x(2n) W_{N/2}^{n(k-N/2)} + W_N^{k-N/2} \sum_{n=0}^{N/2-1} x(2n+1) W_{N/2}^{n(k-N/2)} \quad (3)$$

donde  $0 \leq k \leq (N/2)-1$

Como

$$W_{N/2}^{n(k+N/2)} = W_{N/2}^{n(N/2)} W_{N/2}^{nk} = e^{j2\pi n} W_{N/2}^{nk} = W_{N/2}^{nk}$$

y

$$W_N^{k-N/2} = W_N^k e^{-j\pi} = -W_N^k$$

la ecuación (3) puede ser reescrita como

$$\begin{aligned} X(k+N/2) &= \sum_{n=0}^{N/2-1} x(2n) W_{N/2}^{nk} - W_N^k \sum_{n=0}^{N/2-1} x(2n+1) W_{N/2}^{nk} \\ &= Y(k) - W_N^k Z(k) \quad k = 0, 1, \dots, (N/2)-1 \end{aligned} \quad (4)$$

Por lo tanto, la ecuación (2) puede ser usada para procesar la primera mitad del espectro de frecuencia  $X(k)$  para el rango de índices  $0 \leq k \leq (N/2)-1$ , mientras que la ecuación (4) puede ser utilizada para procesar la segunda mitad del espectro de frecuencia  $X(k+N/2)$ .

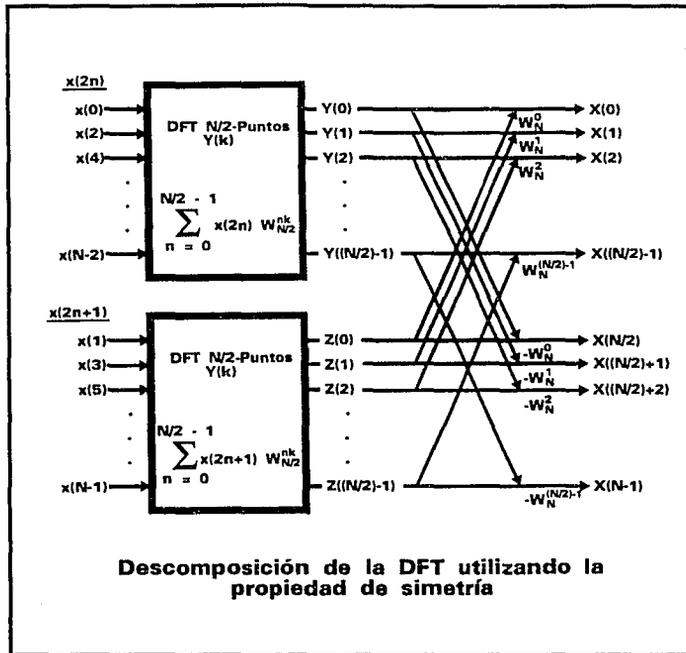
En la figura 2 muestra la situación de cuando la propiedad de simetría del factor de giro (Twiddle Factor) es utilizado para procesar  $X(k)$ . El proceso de decimación mencionado y el aprovechamiento de la simetría reducen el tiempo de procesamiento de la DFT. Mediante la decimación continua de las muestras en el tiempo con numeración par y con numeración non, similarmente se pueden obtener cuatro DFT de  $N/4$ -puntos, resultando en otra mayor reducción en el procesamiento de la DFT. Consecuentemente, para llegar finalmente a el algoritmo de la DIT FFT en radix-2, el proceso de decimación es repetido continuamente hasta que eventualmente la DFT de  $N$  puntos puede ser evaluada como una colección de mariposas de DFT de 2-puntos.

### 3.2.1.1 Mariposas de la FFT con DIT Radix-2

En el algoritmo DIT FFT radix-2, el proceso de la decimación del tiempo pasa por un total de  $M$  etapas, donde  $N = 2^M$  con  $N/2$  FFT de 2-puntos o etapas de mariposa, dando un total de  $(N/2)\log_2 N$  mariposas por cada FFT de  $N$ -puntos.

Para el caso de la implementación de una DFT de 8-puntos con el algoritmo DIT FFT radix-2 discutido anteriormente, las muestras de entrada son procesadas a través de tres etapas. Se requieren cuatro mariposas por etapa, dando un total de doce mariposas en la implementación radix-2. Cada mariposa es una DFT de 2-puntos ilustrada en la figura 3.  $P$  y  $Q$  son las entradas de la mariposa DIT FFT radix-2. En general, las entradas a cada mariposa son complejas así como también el factor de giro.

Figura 2

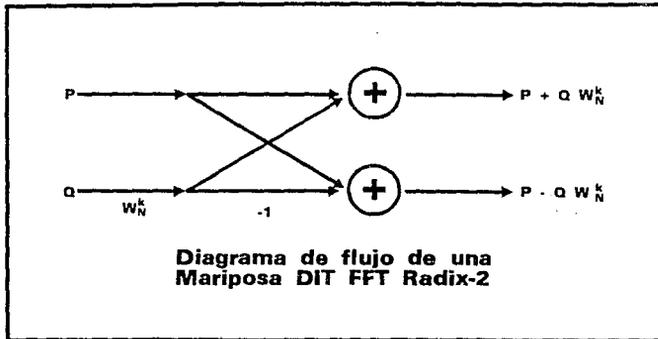


Como se muestra en la figura 3, las salidas P' y Q' de la mariposa radix-2 están dadas por

$$\begin{aligned}
 P' &= P + Q W_N^k \\
 Q' &= P - Q W_N^k
 \end{aligned}
 \tag{5}$$

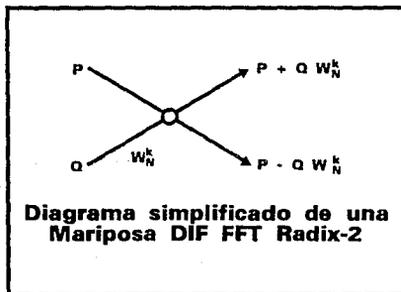
Mientras la figura 3 usa la nomenclatura de señales de gráficas de flujo, otra simbología utilizada para la representación de las mariposas radix-2, es mostrado en la figura 4.

Figura 3



### 3.2.1.2 Implementación de la Mariposa FFT con Escalamiento

Figura 4



En el procesamiento de la FFT, el escalamiento de los resultados intermedios se hace necesario para prevenir el sobreflujo. El TMS320C25 está compuesto de elementos optimizados para el procesamiento digital de señales y un número de registros de corrimiento para escalación dentro de el mismo circuito integrado.

Para poder visualizar por que el escalamiento es necesario, se observa que a partir de la ecuación general de la DFT de N-puntos, aplicando el teorema de Parseval, nos

da

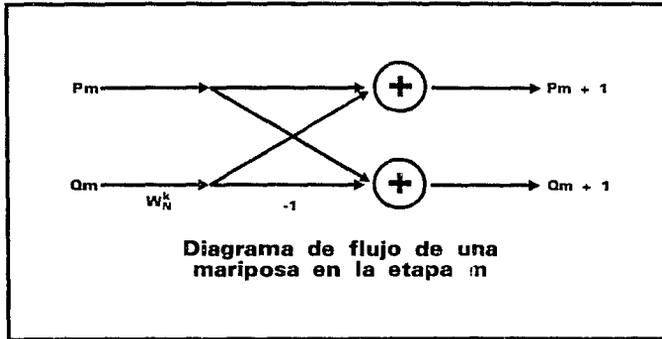
$$\sum_{n=0}^{N-1} x^2(n) = 1/N \sum_{k=0}^{N-1} |X(k)|^2 \quad (6)$$

o

$$N \frac{1}{N} \sum_{n=0}^{N-1} x^2(n) = \frac{1}{N} \sum_{k=0}^{N-1} |X(k)|^2 \quad (7)$$

el valor cuadrático medio de  $X(k)$  es  $N$  veces el valor de la salida de  $x(n)$ . Por consiguiente, el procesamiento de DFT de la secuencia de entrada  $x(n)$ , puede generar sobreflujos cuando se utiliza aritmética de punto fijo si no se realiza el escalamiento. Para poder ver como ocurre el sobreflujo en el procesamiento de FFT, hay que considerar la mariposa general radix-2 en la  $m$ -ésima etapa en una DIT FFT de  $N$ -puntos, tal y como se muestra en la figura 5.

Figura 5



De la figura 5, la forma final de la FFT puede ser escrita como

$$P_{m+1} = P_m + W_N^k Q_m \quad (8)$$

$$Q_{m+1} = P_m - W_N^k Q_m$$

donde  $P_m$  y  $Q_m$  son las entradas, y  $P_{m+1}$  y  $Q_{m+1}$  son las salidas de la  $m$ -ésima etapa de la FFT de  $N$ -puntos. En general,  $P_m$ ,  $Q_m$ ,  $P_{m+1}$ , y  $Q_{m+1}$  son complejas, así como también el factor de giro. El factor de giro puede ser expresado como

$$W_N^k = e^{j(2\pi/N)k} = \cos(X) - j\text{sen}(X) \quad (9)$$

donde  $X = (2\pi/N)k$  y  $j = \sqrt{-1}$ .

Las entradas  $P_m$  y  $Q_m$  pueden ser expresadas en términos de su parte real e imaginaria por

$$P_m = PR + j PI \quad (10)$$

$$Q_m = QR + j QI$$

Mediante la sustitución de los valores de la ecuación (9) y la ecuación (10) la ecuación (8) se convierte en

$$\begin{aligned} P_{m-1} &= PR + j PI + (QR \cos(X) + QI \sin(X) + j (QI \cos(X) - QR \sin(X))) \\ &= (PR + QR \cos(X) + QI \sin(X)) + j (PI + QI \cos(X) - QR \sin(X)) \end{aligned} \quad (11)$$

$$\begin{aligned} Q_{m-1} &= PR + j PI - (QR \cos(X) + QI \sin(X) - j (QI \cos(X) - QR \sin(X))) \\ &= (PR - QR \cos(X) - QI \sin(X)) + j (PI - QI \cos(X) + QR \sin(X)) \end{aligned}$$

Aunque las entradas de cada etapa en las mariposas tienen partes real e imaginaria con magnitud menor a uno, las partes real e imaginaria de las salidas de la ecuación (10) pueden tener una magnitud máxima de

$$1 + 1 \sin(45) + 1 \cos(45) = 2.414213562$$

Para evitar la posibilidad de un sobreflujo, a cada etapa de la FFT se le aplica un factor de escalamiento reductor de dos. De esta manera, si una FFT consiste de M etapas, la salida es reducida por  $2^M = N$ , donde N es la longitud de la FFT. Aún con escalamiento, el sobreflujo es posible debido a el valor máximo de magnitud de la entrada compleja. Esta posibilidad es evitada mediante un escalamiento reductor de la señal de entrada por un factor de 1.207106781, el cual se vuelve a aplicar en la última etapa de la FFT pero como un escalamiento aumentativo. El escalamiento adicional no está implementado en el algoritmo computacional, debido a que se asume que la señal de entrada siempre es real (por lo que la parte imaginaria es cero), y el valor máximo antes mencionado no se puede presentar. El valor máximo para una entrada real es de dos.

Utilizando la ecuación (10), el código para una mariposa implementado en el TMS320C25 está dado en la figura 6. Se asume que todas los valores de las entradas y salidas están en el formato Q15; esto es, que están expresados en aritmética fraccional en su complemento a dos con el punto binario inmediatamente a la derecha del bit de signo (15 bits después de el punto binario). El código incorpora una etapa de escalamiento reductor en dos para la implementación de la mariposa general de la DIT FFT radix-2 con los valores de 16 bits de los senos y cosenos del factor de giro almacenados en el formato Q15. Cabe mencionar que en la realización de la multiplicación fraccional el producto de dos fracciones Q15 de 16 bits da como resultado una fracción de 32 bits de doble precisión en un formato

Q30 con dos bits de signo. Este resultado es presentado en la figura 7, donde S es el bit de signo.

Figura 6

```

:      Codigo de una mariposa general para la DIT FFT Radix-2
:
:      Ecuaciones para la parte real e imaginaria de  $X_m(P)$  y  $X_m(Q)$ .
:      Las localidades PR, PI, QR y QI son utilizadas para los datos de
:      entrada y salida.
:
PR      EQU      0      :  $Re(P_m)$  almacenado en la localidad 0
PI      EQU      1      :  $Im(P_m)$  almacenado en la localidad 1
QR      EQU      2      :  $Re(Q_m)$  almacenado en la localidad 2
QI      EQU      3      :  $Im(Q_m)$  almacenado en la localidad 3
:
:      Igualdad para las partes real e imaginaria del factor giro.
:
COSX    EQU      4      :  $COS(X)$  almacenado en la localidad 4
SENX    EQU      5      :  $SEN(X)$  almacenado en la localidad 5
:
:      Inicialización del Sistema
:
:
INIT:   AORG     >20
        SPM      0
        SXXM
        ROVM
        LDPK     4

```

En la sección de comentarios de la figura 6, ACC, Registro-P, y Registro-T representan a el acumulador de 32 bits, el registro de producto de 32 bits, y el registro temporal de 16 bits de el procesador TMS320C25, respectivamente.

El primer bloque en el código de la mariposa de la figura 6 (empezando con la etiqueta de INIT) es para la inicialización general de el sistema. El segundo bloque de el código (empezando con la etiqueta BTRFLY) aprovecha los bits de doble signo para proporcionar un escalamiento reductor en dos para el termino  $(1/2)(QR \cos(X) + QI \sen(X))$ , el cual es la parte real escalada del producto del factor de giro y  $Q_m$ . Ademas, como el contenido actual de la localidad de memoria QR ya no es requerida para cálculos subsecuentes, QR es utilizado como una unidad de almacenamiento temporal para el término obtenido.

El tercer bloque del código calcula el término  $(1/2)(QI \cos(X) - QR \sen(X))$ , el cual es la parte imaginaria del producto del factor de giro y  $Q_m$ . Mediante la terminación de este cálculo, QI esta también libre para servir como unidad de almacenamiento temporal para el término obtenido.

El cuarto bloque del código calcula la parte real de  $P_{m-1}$  y  $Q_{m-1}$  y proporciona la función de escalamiento de dos por etapa para evitar el sobreflujo. Para implementar esta función, se utiliza el registro de corrimiento de la entrada binaria de el TMS320C25.

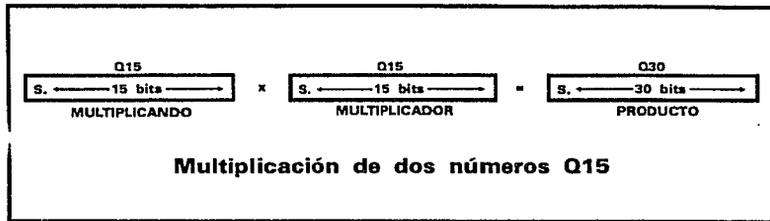
Figura 6 (continuación)

Calcular $QR \cdot \cos(X) + QI \cdot \sin(X)$ y almacenar el resultado en QR.		
BTRFLY:	LT	QR : Registro-T = QR
	MPY	COSX : Registro-P = (1/2)QR * COSX
	LTP	QI : ACC = (1/2)QR * COSX, Registro-T = QI
	MPY	SINX : Registro-P = (1/2)QI * SENX
	APAC	: ACC = (1/2)(QR * COSX + QI * SENX)
	MPY	COSX : Registro-P = (1/2)QI * COSX
	LT	QR : Registro-T = QR
	SACH	QR : QR = (1/2)(QR * COSX + QI * SENX)
Calcular $QI \cdot \cos(X) - QR \cdot \sin(X)$ y almacenar el resultado en QI.		
	PAC	: ACC = (1/2)QI * COSX
	MPY	SENX : Registro-P = (1/2)QR * SENX
	SPAC	: ACC = (1/2)(QI * COSX - QR * SENX)
	SACH	QI : QI = (1/2)(QI * COSX - QR * SENX)
Calcular $\text{Re}(P_{m-1})$ y $\text{Re}(Q_{m-1})$ y almacenar el resultado en PR y QR.		
	LAC	PR,14 : ACC = (1/4)PR
	ADD	QR,15 : ACC = (1/4)(PR + QR * COSX + QI * SENX)
	SACH	PR,1 : PR = (1/2)(PR + QR * COSX + QI * SENX)
	SUBH	QR : ACC = (1/4)(PR - QR * COSX - QI * SENX)
	SACH	QR,1 : QR = (1/2)(PR - QR * COSX - QI * SENX)
Calcular $\text{Im}(P_{m-1})$ y $\text{Im}(Q_{m-1})$ y almacenar el resultado en PI y QI.		
	LAC	PI,14 : ACC = (1/4)PI
	ADD	QI,15 : ACC = (1/4)(PI + QI * COSX + QR * SENX)
	SACH	PI,1 : PR = (1/2)(PI + QI * COSX + QR * SENX)
	SUBH	QI : ACC = (1/4)(PI - QI * COSX - QR * SENX)
	SACH	QI,1 : QR = (1/2)(PI - QI * COSX - QR * SENX)

Inicialmente, el contenido de PR es escalado entre un factor de cuatro (equivalente a hacer un corrimiento a la izquierda de 14 bits). Hay que notar que el corrimiento o función de escalación es realizada mientras que el contenido de PR está siendo almacenado en el acumulador de 32 bits. Dado que el TMS320C25 tiene un acumulador de 32 bits de doble longitud, no se pierde exactitud en el proceso de escalamiento reductor binario. Para generar el resultado final de  $\text{Re}(P_{m-1})$ , el contenido de QR se tiene que sumar al contenido del acumulador con un corrimiento a la derecha en una posición (equivalente a un corrimiento a

la izquierda de 15 bits). Esto implica el sumar  $(1/4)(QR \cos(X) + QI \sin(X))$  a  $(1/4)PR$ , que es el valor actual almacenado en el acumulador. La mitad superior del acumulador es, posteriormente, almacenada en PR con un corrimiento a la izquierda en una posición para dar el término  $(1/2)(PR + QR \cos(X) + QI \sin(X))$ , que es precisamente  $Re(P_{m-1})$  escalado reductivamente por un factor de dos. Este corrimiento o función de escalación es realizada mientras el contenido de la mitad superior del acumulador esta siendo almacenada en PR. En este momento el acumulador sigue teniendo un valor igual a  $(1/4)(PR + QR \cos(X) + QI \sin(X))$ , por lo tanto, para obtener el resultado final de  $Re(Q_{m-1})$ , el contenido sin escalación de QR tiene que ser restado del acumulador. La mitad superior del acumulador es otra vez almacenada en QR con un corrimiento a la izquierda en una posición para dar el término  $(1/2)(PR - QR \cos(X) - QI \sin(X))$ , que es precisamente  $Re(Q_{m-1})$  escalado reductivamente por un factor de dos.

Figura 7



De manera similar, el quinto bloque de el código calcula la parte imaginaria de  $P_{m-1}$  y  $Q_{m-1}$ .

Hay que notar que hasta el momento todas las funciones de escalación realizadas se han elaborado sin retardo adicional de procesamiento debido a la arquitectura del TMS320C25.

En resumen, los valores son escalados por corrimientos a la izquierda de las variables de 16 bits al momento de ser transferidas a el acumulador de 32 bits, manteniéndose precisión en los cálculos.

### 3.2.1.3 Procesamiento de la FFT en su mismo espacio

En la implementación de las mariposas, el conjunto de registros de entrada en memoria de datos (PR, PI, QR y QI) para las dos entradas complejas  $P_m$  y  $Q_m$  son utilizadas para guardar las dos salidas complejas  $P_{m-1}$  y  $Q_{m-1}$ , respectivamente. Cuando el mismo conjunto

de registros de entrada se utiliza como registros de salida para almacenar los resultados de la FFT, el procesamiento de la FFT se dice que esta hecha en el mismo espacio. Por lo tanto, la implementación de la rutina para las mariposas de la FFT en el TMS320C25 está elaborada en el mismo espacio.

Como una regla general, el procesamiento de la FFT en el mismo espacio implica que se necesitan  $2N$  localidades de memoria para una FFT de  $N$ -puntos debido a que las entradas de la FFT pueden ser complejas. En el caso contrario se requieren de  $4N$  localidades de memoria para el procesamiento no elaborado en el mismo espacio.

Otro atributo atractivo de la rutina para las mariposas, es que no se necesitan de registros temporales para cálculos y resultados intermedios.

### 3.2.1.4 Obtención de Reversión de Bits

Como se muestra en la figura 8 las entradas muestreadas en el tiempo  $x(n)$  no están en orden, sino se encuentran revueltas. Tal revoltura de datos o Reversión de Bits es un resultado directo de la derivación de la FFT radix-2. En una inspección más meticulosa se observa que el índice de cada muestra de entrada tiene sus bits en orden invertido como se muestra en la tabla 1.

Tabla 1

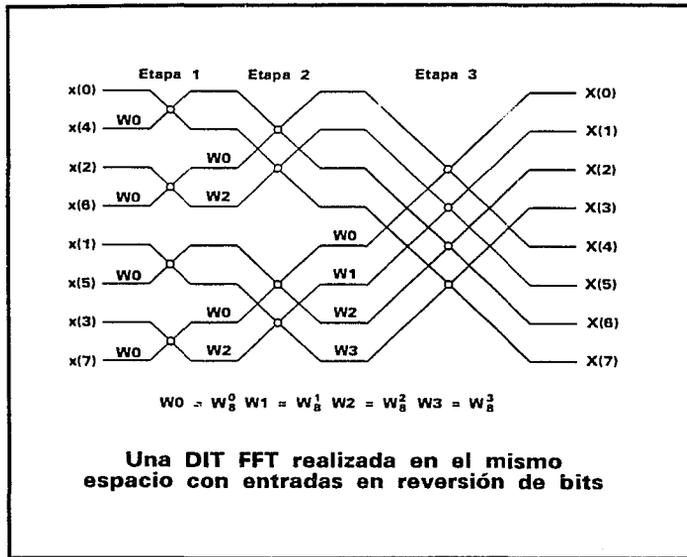
Patrón Índice	Patrón de de bits	Índice con Reversión de bits	Reversión de bits
0	000	000	0
1	001	100	4
2	010	010	2
3	011	110	6
4	100	001	1
5	101	101	5
6	110	011	3
7	111	111	7

Por lo tanto, la secuencia de datos de entrada tiene que ser prerrevuelta antes de ejecutarse la FFT para poder obtener salida de datos en el orden correcto. Para realizar la reversión de bits en la FFT de 8-puntos, mostrada en la figura 8, las parejas de datos de entrada,  $[x(1)$  y  $x(4)]$  y  $[x(3)$  y  $x(6)]$ , deben de ser intercambiadas. De otra manera, la figura 9 tiene las muestras de entrada en orden mediante un reordenamiento

de las mariposas. Sin embargo, ahora las salidas se encuentran con sus bits en orden inverso.

En general, la reversión de bits o reordenación de datos, tiene que ser realizada en la etapa de entrada en las muestras de tiempo (figura 8) o en la etapa de salida en las muestras de frecuencia (figura 9). La reversión de bits puede ser realizada en el mismo espacio. Dicho proceso generalmente requiere el uso de una localidad temporal de memoria.

Figura 8

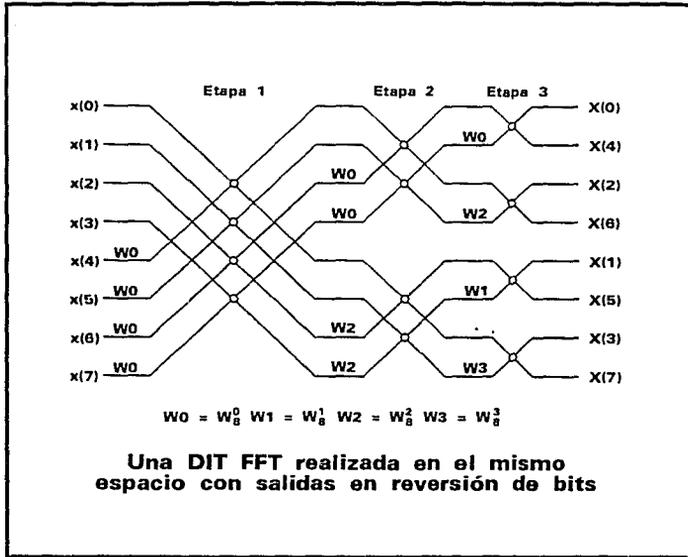


### 3.2.1.5 Mariposas Especiales

Aunque cualquier FFT de N-puntos (donde N es una potencia de 2) puede ser implementada directamente con la mariposa general, normalmente se utilizan mariposas especiales con el propósito de incrementar la velocidad de ejecución de la FFT.

Las mariposas especiales pueden ser codificadas aprovechando ciertos valores del factor de giro de las funciones de seno y coseno. Por ejemplo, cuando el ángulo X toma valores tales como 0, 90, 180 y 270 grados, las mariposas requieren menos código. Otras mariposas especiales también pueden ser codificadas para ángulos con valores de 45, 135, 225 y 315 grados.

Figura 9



### 3.2.2 Mariposas de la FFT con DIF Radix-4

La implementación descrita hasta el momento se refiere a una FFT con DIT radix-2. La DIT de la FFT se calcula mediante el particionamiento de las entradas  $x(n)$  en secuencias más pequeñas y calculando su FFT. Otra alternativa de solución, la secuencia de salida  $X(k)$ , que representa la transformada de Fourier de  $x(n)$ , puede ser desglosada en subsecuencias menores que son calculadas a partir de  $x(n)$ . Este método es llamado Decimación en Frecuencia (DIF). Computacionalmente no existe ninguna diferencia de las dos alternativas de solución.

En una FFT radix-4 cada mariposa contiene cuatro entradas y cuatro salidas en lugar de dos como en el caso de la FFT radix-2. El resultado final es que el nivel de procesamiento de la FFT se reduce y la FFT radix-4 es calculada más rápido que la FFT radix-2.

Para introducir la FFT con DIF radix-4 se toma

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk} \quad k = 0, 1, \dots, N-1 \quad (12)$$

ya se descompone en cuatro sumatorias. Estas cuatro sumatorias corresponden a los cuatro componentes de la mariposa de radix-4. La elección de tener  $N/4$  muestras  $x(n)$  consecutivas en cada suma esta determinada por la elección de la DIF.

$$\begin{aligned} X(k) &= \sum_{n=0}^{N-1} x(n) W_N^{nk} \\ &= \sum_{n=0}^{(N/4)-1} x(n) W_N^{nk} + \sum_{n=N/4}^{(N/2)-1} x(n) W_N^{nk} + \sum_{n=N/2}^{(3N/4)-1} x(n) W_N^{nk} + \sum_{n=3N/4}^{N-1} x(n) W_N^{nk} \\ &= \sum_{n=0}^{(N/4)-1} x(n) W_N^{nk} + W_N^{Nk/4} \sum_{n=0}^{(N/4)-1} x(n+N/4) W_N^{nk} \\ &\quad + W_N^{Nk/2} \sum_{n=0}^{(N/4)-1} x(n+N/2) W_N^{nk} + W_N^{3Nk/4} \sum_{n=0}^{(N/4)-1} x(n+3N/4) W_N^{nk} \end{aligned} \quad (13)$$

$$k = 0, 1, \dots, N-1$$

De la definición del factor de giro se puede mostrar que:

$$W_N^{Nk/4} = (-j)^k, \quad W_N^{Nk/2} = (-j)^k \text{ y } W_N^{3Nk/4} = (j)^k$$

donde  $j$  es la raíz cuadrada de  $-1$ . Con está sustitución la ecuación (13) puede ser reescrita como:

$$X(k) = \sum_{n=0}^{(N/4)-1} [x(n) + (-j)^k x(n+N/4) + (-1)^k x(n+N/2) + (j)^k x(n+3N/4)] W_N^{nk} \quad (14)$$

La ecuación (14), todavía no es la FFT de tamaño  $N/4$ , debido a que el factor de giro depende de  $N$  y no de  $N/4$ . Para convertirla en una FFT de  $N/4$ -puntos la secuencia  $X(k)$  tiene que ser dividida en cuatro secuencias (decimación en frecuencia) para los casos en que  $K = 4r, 4r+1, 4r+2$  y  $4r+3$ .

Introduciendo la segmentación y recordando que

$$W_N^{-nr} = W_{N/4}^{nr}$$

las siguientes cuatro ecuaciones son obtenidas de la ecuación (14):

$$X(4r) = \sum_{n=0}^{(N/4)-1} [x(n) + x(n+N/4) + x(n+N/2) + x(n+3N/4)] W_N^0 W_{N/4}^{nr}$$

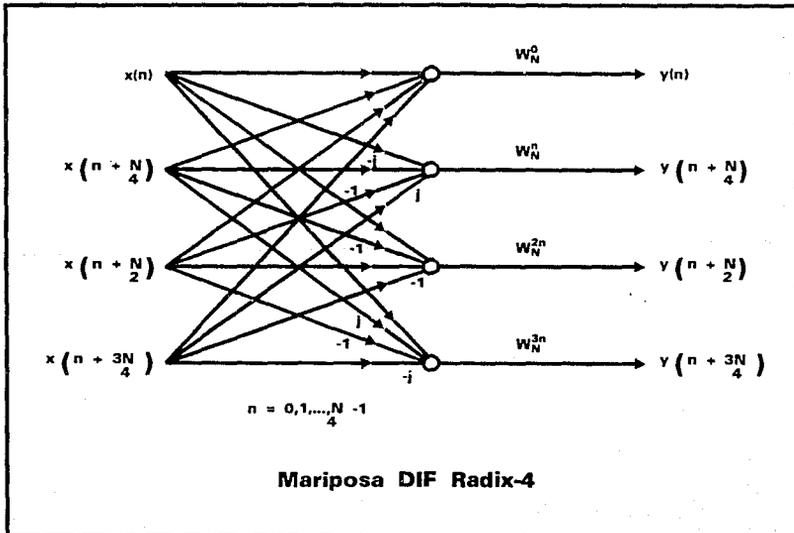
$$X(4r+1) = \sum_{n=0}^{(N/4)-1} [x(n) - j x(n+N/4) - x(n+N/2) + j x(n+3N/4)] W_N^n W_{N/4}^{nr}$$

(15)

$$X(4r+2) = \sum_{n=0}^{(N/4)-1} [x(n) - x(n+N/4) + x(n+N/2) - x(n+3N/4)] W_N^{2n} W_{N/4}^{nr}$$

$$X(4r+3) = \sum_{n=0}^{(N/4)-1} [x(n) + j x(n+N/4) - x(n+N/2) - j x(n+3N/4)] W_N^{3n} W_{N/4}^{nr}$$

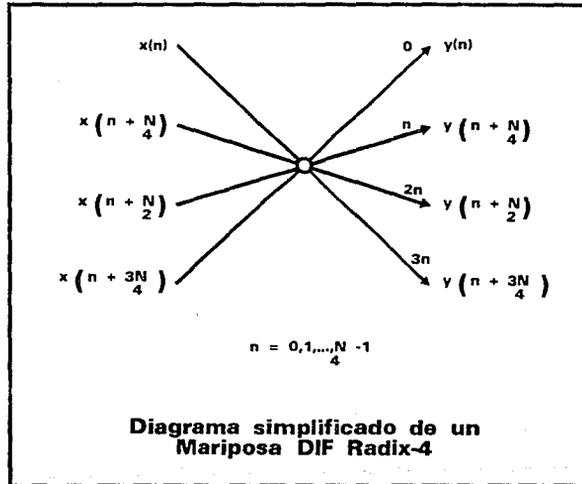
Figura 10



Cada una de estas ecuaciones es ahora, una FFT de N/4-puntos que puede ser procesada mediante la repetición del procedimiento anterior hasta llegar a  $N = 4$ . En general, la FFT de N-puntos (donde N es una potencia de 4) puede ser reducida a el cálculo de la FFT de N/4-puntos mediante la transformación de la señal de entrada  $x(n)$  a una señal intermedia  $y(n)$ , como se sugiere en la ecuación (15). La figura 10 muestra la mariposa radix-4 correspondiente, la cual genera cada termino para las sumatorias de la ecuación (15).

Por simplicidad, normalmente se utiliza la notación de la figura 11 para la mariposa de la FFT con DIF radix-4.

Figura 11



En el apéndice A se encuentra la implementación de la FFT con DIF radix-4 de 256 puntos, para el TMS320C25.

### 3.3 La Transformada Rápida de Hartley

Joseph Fourier le dejó a los matemáticos el legado de la transformada que lleva su nombre, la cual procesa las señales de audio y electromagnéticas de hoy en día.

Sin embargo, la Transformada Rápida de Fourier (FFT) requiere de un gran número de recursos computacionales. La Transformada Rápida de Hartley (FHT) puede obtener los mismos resultados más rápidamente, utilizando un número menor de recursos. Por primera vez descrita por Ronald Brecewell [1] la FHT realiza muchos de los trabajos que son hechos por la FFT.

Ambas transformadas, la FFT y la FHT, permiten el mapeo de una señal continua en el tiempo a una función representada en la frecuencia. La transformada de Fourier convierte una función real en el tiempo  $X(t)$  a una función compleja en frecuencia  $F(f)$ .

La transformada de Hartley convierte una función real en el tiempo  $X(t)$  a una función en frecuencia  $F(f)$ . Como la función de frecuencia de Hartley es real, sólo se necesitan operaciones aritméticas sencillas para realizar su cálculo.

Aún más, los arreglos de datos tipo real solamente requieren de la mitad del espacio en memoria de almacenamiento con relación a un arreglo de números complejos. Esto significa que la transformada de Hartley requiere de menos espacio en memoria de datos que lo que se necesita para la transformada de Fourier. Por lo tanto, la transformada de Hartley será más rápida y ocupará menos espacio que la transformada de Fourier convencional utilizada en aplicaciones de filtrado digital y ampliación de imágenes donde se tiene que procesar una gran cantidad de información.

### 3.3.1 Obtención de la Transformada de Hartley

La ecuación (16) muestra la forma analítica de la transformada de Hartley; y la ecuación (17) la forma de la transformada inversa de Hartley, la cual es utilizada para convertir la función en el dominio de la frecuencia, a su dominio en el tiempo.

$$H(f) = \frac{1}{2\pi} \int x(t) \text{cas}(2\pi ft) dt \quad (16)$$

$$X(t) = \int H(f) \text{cas}(2\pi ft) df \quad (17)$$

donde  $\text{cas}(2\pi ft) = \cos(2\pi ft) + \sin(2\pi ft)$ .

La función  $\text{cas}(2\pi ft)$  fue introducida por R.V.L. Hartley, quien inicialmente propuso la transformada de Hartley en 1942.

Las funciones de transformación de Hartley de las ecuaciones (16) y (17) se enfocan solamente a variables continuas. Sin embargo, como es el caso de los datos manejados por las computadoras, las señales son muestreadas a intervalos de tiempo discreto. Afortunadamente, se puede definir una transformada discreta que puede representar y cuantizar señales continuas, o señales de duración limitada. La forma discreta de la transformada de Hartley es

$$H(f) = \frac{1}{N} \sum_{t=0}^{N-1} X(t) \text{cas}(2\pi t f / N) \quad (18)$$

y

$$X(t) = \sum_{f=0}^{N-1} H(f) \text{cas}(2\pi f t / N) \quad (19)$$

### 3.3.2 El Algoritmo Rápido

Conforme a las ecuaciones (18) y (19), el cálculo computacional de la transformada discreta de Hartley presenta un problema análogo al que se presentó en la transformada discreta de Fourier. Esto es, se tienen que realizar  $N^2$  operaciones aritméticas para calcular la transformada discreta de Hartley para un conjunto de  $N$ -elementos.

Un trabajo publicado por Cooley y Turkey en 1965 llevó al desarrollo del algoritmo rápido para resolver el problema de cálculo de la transformada discreta de Fourier de una serie compleja.

Básicamente, la FFT utiliza un proceso de permutación para bisectar la información hasta que se logran pares. Posteriormente el cálculo de la transformada de Fourier de este par de puntos es muy rápido.

La idea aplicada en el proceso de permutación se basa en que es más fácil reducir la información a pares, calcular la transformada de dicho par y recombinarlos para formar toda la transformada, que calcular la transformada de todo el conjunto de datos.

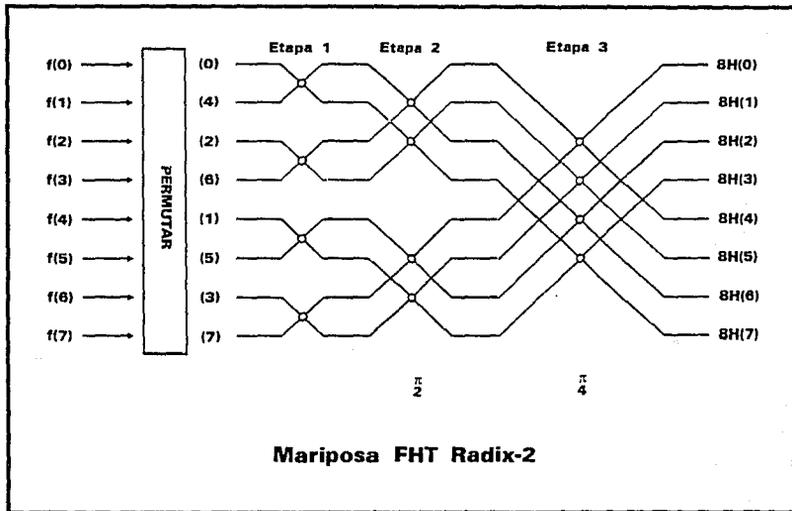
La permutación es particularmente rápida cuanto la cantidad de información es grande. Si todos los pares de dos elementos son agrupados mediante un proceso conocido como de mariposa (debido a la apariencia del diagrama de flujo de información) se puede calcular la transformada de Fourier del conjunto de datos. Se generan aproximadamente  $N \log(N)$  operaciones aritméticas para calcular la transformada de un conjunto de datos de  $N$ -puntos.

Bracewell [20 y 21] demostró que se puede aplicar la misma metodología para la transformada de Fourier en la obtención de un algoritmo para la transformada de Hartley [2]. Similarmente el método utiliza el proceso de permutación para bisectar los datos hasta obtener los pares de información.

La transformada de Hartley de un par de datos (a,b) es  $\frac{1}{2}(a+b, a-b)$ , y el cálculo de tal pa. es muy rápido. También es posible sobreponer la secuencia de dos elementos para calcular la transformada de Hartley del conjunto de datos. Sin embargo, para poder realizar esto, es necesario desarrollar una fórmula que exprese una transformada discreta de Hartley (DHT) completa en términos de su subsecuencias de medio tamaño.

Así mismo, Bracewell también demostró mediante la aplicación del Teorema de Corrimientos y el Teorema de Similitud, que la ecuación (20) expresa la descomposición general de la DHT. Esta descomposición general, genera la DHT deseada mediante la bisección de la información.

Figura 12



Puesto de otra manera, es la regla utilizada para generar los elementos que serán utilizados en el cálculo de las mariposas de la transformada.

$$H(f) = H_1(f) + H_2(f) \cos(2\pi f/N_S) + H_2(N_S-f) \sin(2\pi f/N_S) \quad (20)$$

donde  $N_S$  es el número de elementos en la secuencia de medio tamaño, y por tanto  $N_S = N/2$  para un conjunto de datos de  $N$  elementos. La figura 12 muestra un diagrama completo de mariposa para el cálculo de la FHT de un conjunto de datos de 8-puntos.

La fórmula de descomposición para la FHT difiere de la FFT en un aspecto importante: Los elementos multiplicados por los términos trigonométricos no son simétricos. En la descomposición FFT, los términos multiplicados por el coeficiente trigonométrico involucran solamente términos en  $F(f)$ . En la descomposición FHT ambos términos,  $H(f)$  y  $H(N_S-f)$  tiene coeficientes seno. Esta asimetría se convierte aparente cuando se expresa la transformada discreta en forma de operaciones de matriz: los términos en la matriz FFT son simétricos con respecto a la diagonal de la matriz, mientras que los términos correspondientes en la matriz FHT son asimétricos.

Lo anterior introduce algunos problemas de cálculo computacional, debido a que la implementación de procesos de matrices asimétricas es difícil de desarrollar. Se puede lidiar con esta asimetría mediante la utilización de una variable independiente como el índice para los elementos multiplicados por el coeficiente seno. Este índice decrece mientras que el otro índice crece; este comportamiento es llamado indexación retrograda.

La transformada inversa de Hartley se puede obtener mediante la aplicación del algoritmo FHT nuevamente a las salidas generadas, de esta manera regenerando los datos de entrada. Esto significa que se puede utilizar el mismo programa para calcular la transformada y su inversa. Sin embargo, existe una ligera asimetría entre la FHT y su inversa. En el caso de la transformación tiempo a frecuencia es necesario escalar el resultado del cálculo de la mariposa. Esto es, para un conjunto de datos de entrada de  $N$  elementos, se tiene que dividir la salida de la mariposa por  $N$  para obtener la transformada de Hartley. Esta última operación no tiene que ser aplicada para la transformada de la frecuencia al tiempo.

El apéndice B contiene un algoritmo para la FHT implementado en el lenguaje de programación TurboPascal 4. En realidad el programa consta de tres transformadas: la DHT (transformada de Hartley discreta) en punto flotante, la FHT (transformada rápida de Hartley) en punto flotante y la FHT en notación entera.

La razón de implementar el algoritmo de diferentes maneras se debe a que se desea conocer el comportamiento que tendría la transformada en varios modos de operación y a la vez conocer de que valor sería la diferencia de la transformada en su notación de punto flotante a su notación entera. Este último punto es importante debido a que el procesador utilizado trabaja exclusivamente en notación entera.

# 4

## Fundamentos de Audio Digital

### 4.1 Introducción

La utilización de métodos digitales de grabación, reproducción y almacenamiento de señales de audio digital incluyen varios conceptos ajenos a métodos analógicos. Los sistemas digitales tienen poca similitud con los sistemas analógicos.

Como el audio es analógico por naturaleza, los sistemas digitales utilizan muestreo y cuantización, que son los pilares de la digitalización para transformar la información de audio.

Se tienen que tomar precauciones para evitar dos tipos de distorsiones: una condición de frecuencias erróneas llamada aliasing y el error introducido por la cuantización de la señal analógica [3].

### 4.2 Muestreo Discreto en el Tiempo

Con la grabación analógica continuamente se realiza modulación de la señal, pero en los sistemas digitales se utilizan números. La digitalización emplea el muestreo en el tiempo y cuantización de la amplitud para codificar la señal analógica como valores discretos en el tiempo y la amplitud.

## 4.2.1 La Naturaleza Sin Pérdidas del Muestreo

Para mostrar la diferencia entre un sistema digital y un sistema analógico utilizaremos la siguiente analogía. Las manecillas de reloj se mueven alrededor de la carátula continuamente para mostrar el tiempo, en comparación con un reloj digital donde también se muestra el transcurso del tiempo pero con valores discretos, en otras palabras, enseña muestras de tiempo.

Lo mismo sucede con el audio el cual varía continuamente en el tiempo y puede ser grabado y reproducido ya sea en forma analógica continua o en forma de muestras digitales en el tiempo. Al igual que ambos relojes proporcionan la hora, ambos sistemas de audio proporcionan la misma señal.

El muestreo en el tiempo es el mecanismo principal para definir la señal en un sistema de audio digital. Lo anterior plantea lo siguiente: si un sistema digital muestrea a intervalos de tiempo discreto ¿será posible perder información entre las muestras de tiempo?.

Existiendo las condiciones adecuadas de muestreo entre la entrada y la salida de un sistema digital no se pierde información. Las muestras contienen la misma cantidad de información que la señal continua.

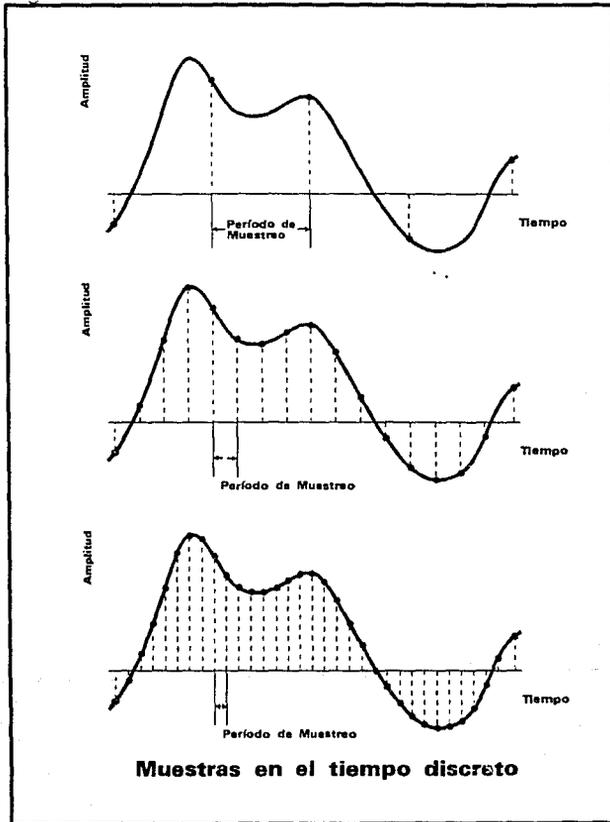
Para establecer claramente este último concepto se plantea lo siguiente: se realiza un experimento en el cual se sujeta una cámara de cine a una motocicleta, la cual recorrerá una ruta compuesta por pendientes, terrenos planos y terrenos accidentados. Al terminar el recorrido se expone la película y se puede percibir que las muestras discretas obtenidas reproducen perfectamente el recorrido en las pendientes y en el terreno plano, sin embargo, al llegar al terreno accidentado la película se ve borrosa. Suponemos que el movimiento vibratorio debido al terreno no plano fue muy rápido para que la cámara captara los cambios. Si ampliáramos el número de recuadros por segundo de tal manera que aumentara la velocidad de captura, sería posible tener una imagen más clara del recorrido. Concluimos de esta manera que de acuerdo a un correcto número de muestras no se pierde información de nuestro recorrido.

Debido a que los sistemas visuales y auditivos del ser humano tienen en concepto poca diferencia es posible contraponer las conclusiones del ejemplo ante un sistema de audio. Si se realiza un muestreo adecuado de la señal de audio en la cual la información que se digitaliza es fiel a la señal analógica, se puede obtener un proceso de muestreo sin pérdida.

En un sistema de audio digital se suaviza la señal de entrada con un filtro paso-bajos para eliminar las frecuencias altas, las cuales no pueden ser correctamente muestreadas. El

diseño del sistema digital tiene su filtrado de frecuencia por arriba del límite auditivo humano.

Figura 13



## 4.2.2 El Teorema de Muestreo

Cuando la señal de entrada es filtrada paso-bajos, teóricamente se puede muestrear la señal de tal manera que no exista pérdida de información entre la salida y la señal de entrada suavizada. Desde el punto de vista del muestreo esto no es una aproximación, sino un caso exacto, de acuerdo a Shannon y a Nyquist. El método de muestreo discreto define exclusivamente valores instantáneos. Cuando la señal es suavizada se pueden reconstruir sin error todos los valores que intervinieron de tal manera que se reconstruye la señal original.

En la figura 13 se tiene una función analógica la cual es muestreada. La amplitud de cada pulso da un número que representa la amplitud de la señal analógica en ese instante. Para cuantificar el evento se define a la **frecuencia de muestreo** como el número de muestras por segundo, su recíproco es la **razón de muestreo**, que se define como el tiempo entre cada muestra. Por ejemplo para una señal con una frecuencia de muestreo de 40,000 muestras/segundo (ciclos/segundo, Hertz) la razón de muestreo es de 1/40,000 segundos (25  $\mu$ s.). Es aparente que para una señal que cambia rápido (con alta frecuencia) se requiere de una frecuencia de muestreo rápida. Por lo tanto, la frecuencia de muestreo determina la respuesta a la frecuencia y el ancho de banda del sistema digital.

La elección de la frecuencia de muestreo es uno de los criterios más importantes de un sistema digital porque determina el ancho de banda del sistema. Esto último presenta lo siguiente: ¿cual debe de ser la razón de muestreo para representar la señal correctamente?.

La teoría de muestreo, descrita por Shannon y Nyquist, propone que se necesita una frecuencia de muestreo de "S" muestras/segundo para representar una señal con un ancho de banda de S/2 Hz. En otras palabras, se debe de muestrear al doble de la frecuencia audible más alta (frecuencia de Nyquist) para tener una representación sin pérdidas. Por ejemplo, para una señal con una respuesta a frecuencia de 0 a 10 KHz según la teoría se necesitaría una frecuencia de muestreo de 20 KHz para una codificación digital adecuada. Un filtro paso-bajos siempre se coloca a la entrada del sistema para eliminar las frecuencias por arriba de la mitad de la frecuencia de muestreo. De igual manera se coloca un filtro a la salida del sistema para eliminar las frecuencias altas creadas dentro del mismo y para suavizar el efecto de escalera que se presenta al reconstruir la señal en su forma analógica original como se muestra en la figura 14.

Otro aspecto del teorema de muestreo que se presenta tiene una relación con el espacio entre muestras, en frecuencias bajas la longitud de onda es grande y existen varias muestras para representar cada periodo. Conforme la frecuencia de la señal va creciendo existen menos muestras para representar cada periodo debido a que la longitud de onda es cada vez más pequeña. En el caso del límite teórico de frecuencia de muestreo (muestreo

crítico), existen solo dos muestras por periodo para una señal de audio de 1/2 la frecuencia de muestreo.

Figura 14

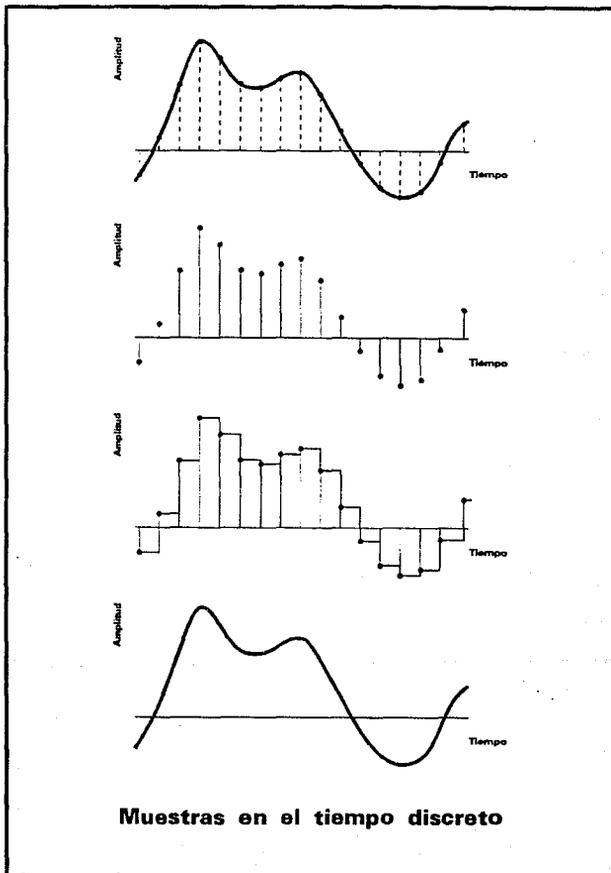
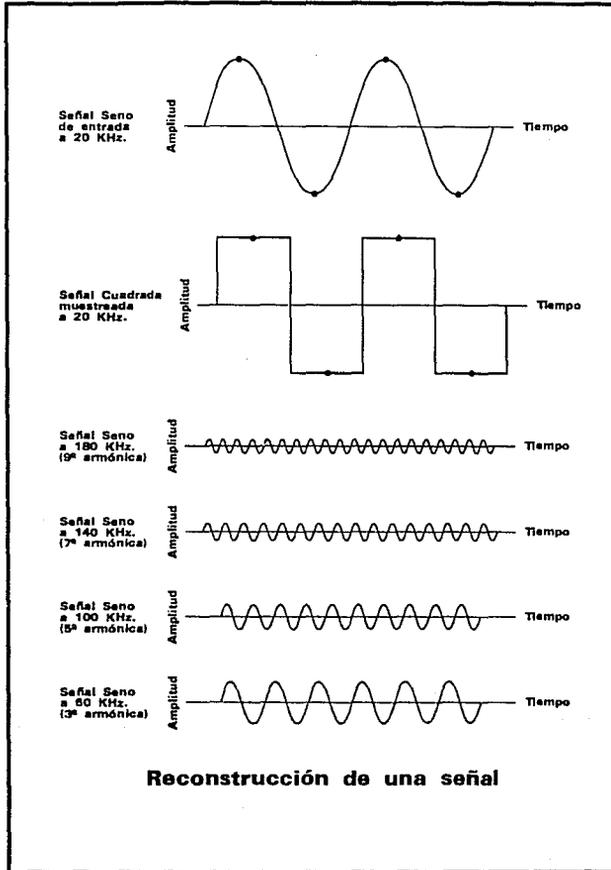


Figura 15



Sin embargo, es posible representar la señal con solo dos muestras. Por ejemplo, se considera el sistema de audio con frecuencia de muestreo de 40 KHz, al cual se le aplica la señal senoidal de 20 KHz como se muestra en la figura 15. El digitalizador produciría dos muestras que se usarían para generar una señal cuadrada a 20 KHz. Esta señal al ser reconstruida no representaría a la señal senoidal. Sin embargo la señal senoidal esta

compuesta por armónicas -señales senoidales a 20, 60, 100, 140, 180, etc. KHz- que serían eliminadas en el momento de pasar por un filtro paso-bajos a la salida del sistema digital de tal manera que la señal resultante sería la senoidal a 20 KHz.

En el ejemplo anterior de muestreo crítico no existe la seguridad de que los tiempos de muestreo coincidan con el máximo y el mínimo de la señal. Las muestras pueden provenir de partes de menor amplitud de la señal senoidal o hasta coincidir con el cruce por el eje cero. En la práctica esto no llega a suceder debido a que no se utiliza el muestreo crítico ya que se crea un margen de seguridad.

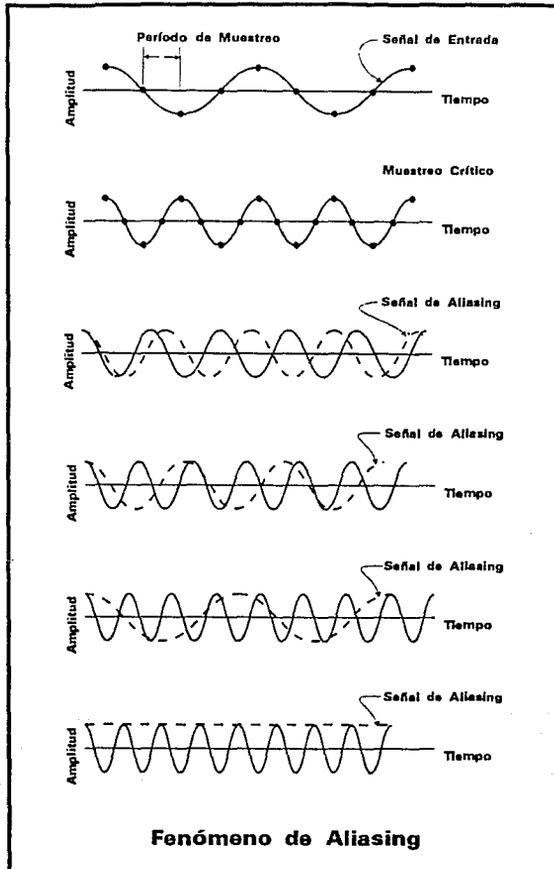
## 4.3 El Fenómeno de Aliasing

Uno de los más importantes conceptos en el diseño de sistemas de digitalización es el de aliasing, el cual se puede explicar como una confusión entre las muestras que se presenta durante el proceso de grabación de la señal. El efecto de aliasing crea componentes falsas de la señal. Las señales erróneas tienen la probabilidad de aparecer dentro del ancho de banda de audio y son imposibles de distinguir de señales legítimas.

### 4.3.1 Frecuencias de Traslape

Aliasing es una consecuencia de una condición no permitida en la teoría de muestreo. Nyquist ha demostrado que la más alta capacidad de frecuencia de un sistema de muestreo solo puede ser igual o menor a la mitad de la frecuencia de muestreo. Si la capacidad de frecuencia es mayor a la mitad de la frecuencia de muestreo el efecto de aliasing es inevitable. La figura 16 muestra la relación que existe entre el crecimiento de la capacidad de frecuencia y la disminución del número de puntos por período. Cuando se llega a la mitad de la frecuencia de muestreo solo existen dos muestras por ciclo el mínimo necesario para representar la naturaleza bipolar de la señal. Si se intenta muestrear a frecuencias mayores el sistema de muestreo seguirá proporcionando muestras a intervalos regulares, pero la variación de la amplitud de las señales fuera del rango crearía información falsa. Conforme las señales fuera del rango crecen, nuevas señales de frecuencias bajas se generan. Si "S" es la frecuencia de muestreo, "F" es la frecuencia fuera del rango de la mitad de la frecuencia de muestreo y "N" es un entero, entonces una nueva frecuencia  $F_a$  es creada en  $F_a = \pm NS \pm F$ . El efecto de aliasing se hace presente en la banda de audio (y sus imágenes) plegadas (frecuencias de plegado) a partir de la frecuencia de muestreo.

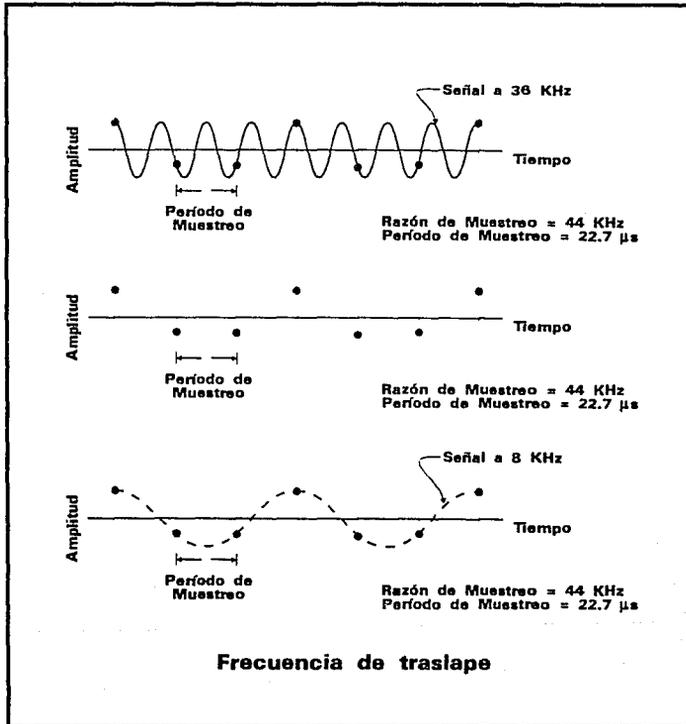
Figura 16



Supongamos que tenemos un sistema digitalizador a 44 KHz y una señal de 36 KHz se ha introducido en el circuito de muestreo tal y como se muestra en la figura 17. La componente  $F_a$  resulta de  $S - F = F_a$  o  $44 - 36 = 8$  KHz. El circuito de muestreo proporcionaría la información equivocada en base a la amplitud de la señal presente. Con referencia en las muestras ningún dispositivo ya sea digital o analógico podría decidir de que

frecuencia se esta hablando (36 KHz o 8 KHz). Después del filtro de salida la señal de 36 KHz ha desaparecido pero la señal de 8 KHz se mantiene lo que induce distorsión en la señal de audio final.

Figura 17

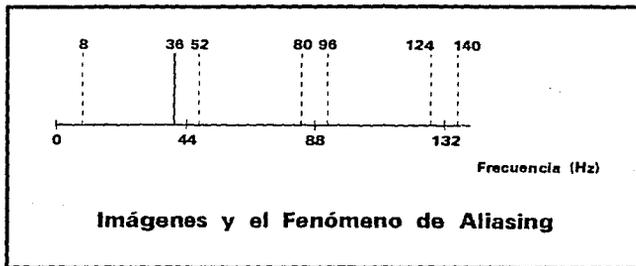


### 4.3.2 Imágenes y Aliasing

Existen otras manifestaciones del efecto de aliasing que se presentan en las imágenes producidas por el muestreo, tal y como se muestra en el espectro de la figura 18. Todas las

componentes son creadas a partir de:  $\pm S \pm F$ ,  $\pm 2S \pm F$ ,  $\pm 3S \pm F$ , etc. La única componente que crea interferencia en frecuencia directamente en el ancho de banda del sistema es la de S-F. La figura 19 muestra como diferentes frecuencias con el fenómeno de aliasing se integran en la banda de audio de 0 a 22 KHz.

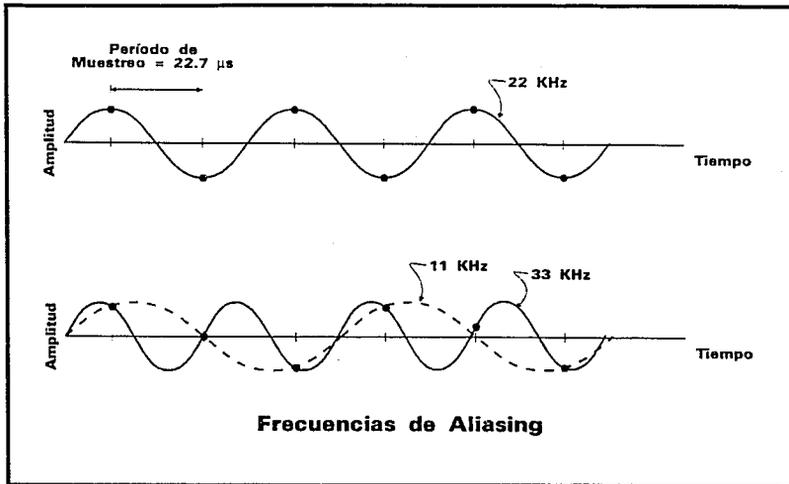
Figura 18



### 4.3.3 Armónicas y Aliasing

En el ejemplo que se ha mencionado solo se trabaja con una función senoidal simple con frecuencias de plegado limitadas a una parte de la señal. Con tonos complejos el fenómeno de aliasing se pueden generar para cada armónica. Por ejemplo, la segunda armónica de una señal compleja con una fundamental de 11 KHz sería de 22 KHz la cual es la muestra crítica de un circuito muestreador de 44 KHz. La tercera armónica es de 33 KHz la cual agrega un factor de aliasing en 11 KHz para sumar a la fundamental. La sexta armónica es de 66 KHz la cual agrega un factor de aliasing en 22 KHz para sumar a la segunda armónica.

Figura 19



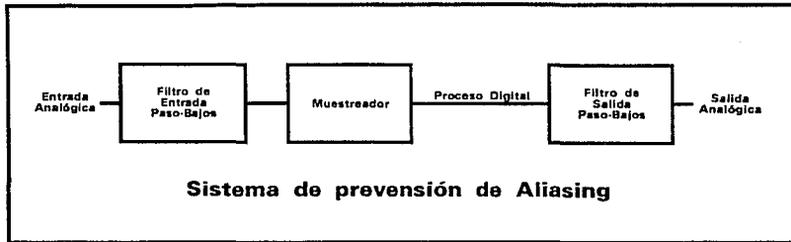
Bajo la mayoría de las condiciones de operación el efecto causado por las armónicas es imposible de distinguir debido a que la amplitud es pequeña y es cubierta por la misma señal de audio.

#### 4.3.4 Solución para el Fenómeno de Aliasing

El fenómeno de aliasing es problemático pero en la práctica no es una condición importante. En un sistema digital de almacenamiento bien diseñado se evita que ocurra el factor de aliasing. La solución mostrada en la figura 20 es sencilla. Se implementa una etapa de entrada que restrinja la señal mediante un filtro paso-bajo (filtro anti-aliasing) diseñado para proporcionar una atenuación considerable a la mitad de la frecuencia de muestreo para asegurarse que la capacidad de frecuencia nunca exceda la mitad de la frecuencia de muestreo.

Es imperativo observar la teoría de muestreo y el filtrado paso-bajo de la señal de entrada en un sistema de digitalización. Si se permite que ocurra aliasing no existe técnica que permita la eliminación de las frecuencias de aliasing del ancho de banda de audio original.

Figura 20



En la figura 21 se muestra un resumen del análisis matemático del espectro de frecuencia con respecto a el fenómeno de aliasing. En la figura 21(A) se aplica el teorema del muestreo y en la figura 21(B) no se aplica teniendo como resultado la introducción del fenómeno de aliasing.

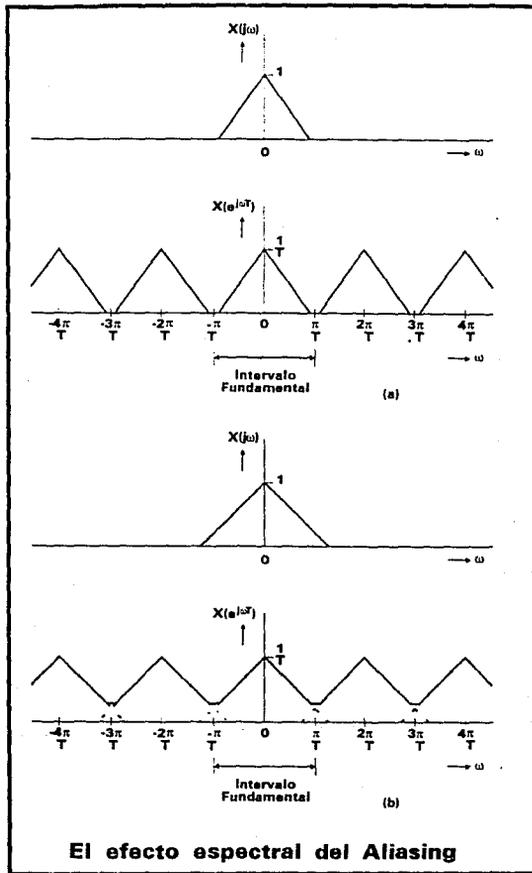
## 4.4 Cuantización

Para grabar una señal de audio se tienen que almacenar información en dos dimensiones. En el muestreo se tiene un almacenamiento implícito de la información de tiempo y la cuantización almacena la información de amplitud. La cuantización se define entonces como el valor medido de la amplitud de la señal analógica muestreada en el tiempo. Con la cuantización, como en la medición de cualquier evento analógico, la exactitud se encuentra limitada por la resolución del sistema. Debido al tamaño finito de las variables, la resolución de un sistema digital es limitado, lo cual introduce un error de medición. Dicho error es similar al ruido en un sistema analógico.

### 4.4.1 Aproximación Analógica y Digital

Para explicar el efecto de cuantización y su error del error inherente en un sistema analógico se utiliza el siguiente ejemplo.

Figura 21



### El efecto espectral del Aliasing

Se conectan dos voltímetros, uno analógico y el otro digital, a una señal de entrada. Si el medidor analógico tiene una carátula con escala adecuada y asumiendo que la lectura visual es exacta se tiene una medición de 1.27 volts.

La misma medición se realiza con un voltímetro digital de dos dígitos y la lectura sería de 1.3 volts. Si la medición se hubiera realizado con un medidor de tres dígitos la lectura sería de 1.27 volts, y con un medidor de cuatro dígitos la lectura podría haber sido de 1.274 volts.

Ambos medidores tiene error. El error en el voltímetro analógico se debe a las características de los elementos mecánicos y a la dificultad en la lectura visual del medidor. Aún con las condiciones ideales en algún momento se pierde la capacidad de medición debido al ruido del medidor.

Con un voltímetro digital la naturaleza del error es diferente. La exactitud está limitada por la resolución del medidor, esto es, por el número de dígitos que se despliegan. Con un mayor número de dígitos se tiene una mayor exactitud, pero el último dígito siempre se encuentra redondeado a el valor desplegado. Bajo las condiciones ideales el último dígito es siempre es exacto. Por ejemplo, un voltaje de 1.3000 volts se mostraría como 1.3. En las peores condiciones el error de redondeo sería de la mitad del intervalo de diferencia, esto es, un voltaje de 1.250 volts sería redondeado a 1.2 o 1.3. Si se utiliza un sistema binario para la medición se dice que el error de resolución del sistema es de la mitad del bit menos significativo (LSb).

Para ambos sistemas el muestrear un fenómeno analógico incluye un error. En el caso de voltímetros una lectura digital es mas robusta. Se obtiene mayor información sobre un sistema analógico cuando es el resultado de datos digitales.

Concluimos que **cuantización** es la técnica de medir un evento analógico para formar un valor numérico. El uso de un sistema digital normalmente se implementa con números binarios. En términos de la cuantización el número posible de valores esta determinado por el tamaño de la palabra de datos, esto es, el número de bits disponibles para la representación.

#### **4.4.2 Razón Señal a Error**

Con un sistema de números binario el tamaño de la palabra de datos determina el número de intervalos de cuantización disponibles. Esto se puede determinar elevando el tamaño de la palabra a una potencia de dos, como se muestra en la tabla 2.

La misma medición se realiza con un voltímetro digital de dos dígitos y la lectura sería de 1.3 volts. Si la medición se hubiera realizado con un medidor de tres dígitos la lectura sería de 1.27 volts, y con un medidor de cuatro dígitos la lectura podría haber sido de 1.274 volts.

Ambos medidores tiene error. El error en el voltímetro analógico se debe a las características de los elementos mecánicos y a la dificultad en la lectura visual del medidor. Aún con las condiciones ideales en algún momento se pierde la capacidad de medición debido al ruido del medidor.

Con un voltímetro digital la naturaleza del error es diferente. La exactitud está limitada por la resolución del medidor, esto es, por el número de dígitos que se despliegan. Con un mayor número de dígitos se tiene una mayor exactitud, pero el último dígito siempre se encuentra redondeado a el valor desplegado. Bajo las condiciones ideales el último dígito es siempre es exacto. Por ejemplo, un voltaje de 1.3000 volts se mostraría como 1.3. En las peores condiciones el error de redondeo sería de la mitad del intervalo de diferencia, esto es, un voltaje de 1.250 volts sería redondeado a 1.2 o 1.3. Si se utiliza un sistema binario para la medición se dice que el error de resolución del sistema es de la mitad del bit menos significativo (LSb).

Para ambos sistemas el muestrear un fenómeno analógico incluye un error. En el caso de voltímetros una lectura digital es mas robusta. Se obtiene mayor información sobre un sistema analógico cuando es el resultado de datos digitales.

Concluimos que **cuantización** es la técnica de medir un evento analógico para formar un valor numérico. El uso de un sistema digital normalmente se implementa con números binarios. En términos de la cuantización el número posible de valores esta determinado por el tamaño de la palabra de datos, esto es, el número de bits disponibles para la representación.

## **4.4.2 Razón Señal a Error**

Con un sistema de números binario el tamaño de la palabra de datos determina el número de intervalos de cuantización disponibles. Esto se puede determinar elevando el tamaño de la palabra a una potencia de dos, como se muestra en la tabla 2.

Tabla 2

Número N de intervalos de cuantización en una palabra binaria.			
$2^1 = 2$	$2^7 = 128$	$2^{13} = 8192$	$2^{19} = 524288$
$2^2 = 4$	$2^8 = 256$	$2^{14} = 16384$	$2^{20} = 1048576$
$2^3 = 8$	$2^9 = 512$	$2^{15} = 32768$	$2^{21} = 2097152$
$2^4 = 16$	$2^{10} = 1024$	$2^{16} = 65536$	$2^{22} = 4194304$
$2^5 = 32$	$2^{11} = 2048$	$2^{17} = 131072$	$2^{23} = 8388608$
$2^6 = 64$	$2^{12} = 4096$	$2^{18} = 262144$	$2^{24} = 16777216$

Si se tiene un número mayor de bits será mejor la aproximación, pero como se ha mencionado siempre existe un error asociado con la cuantización debido al número limitado de amplitudes que pueden ser contenidas en una palabra binaria la cual nunca podrá representar un número infinito de posibilidades analógicas.

Existe un punto en el que el error no puede ser distinguido por el sistema auditivo humano. Al respecto los especialistas en el área han acordado que una adecuada representación se alcanza con 15 a 20 bits.

Para saber el desarrollo que tiene un sistema digitalizador se formula la razón del número total de intervalos ( $N - 1$ ) que el esquema de cuantización cubre con el máximo intervalo de error:

$$\text{Aproximación de Razón señal-a-error} = \frac{\text{número máximo}}{\text{máximo error}}$$

Por ejemplo, con 16 bits :

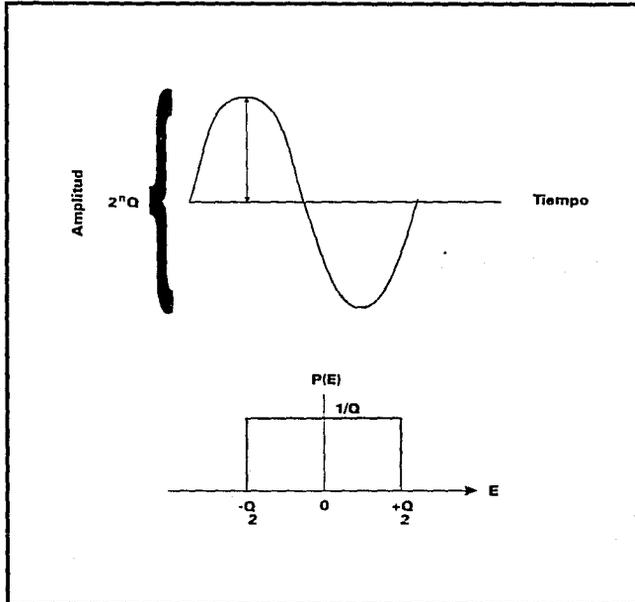
$$\text{Aproximación de Razón señal-a-error} = \frac{65,535}{0.5}$$

= 131,070 que es equivalente a 98 dB. Esta razón de la máxima amplitud representable con el error determina la razón señal-a-ruido (S/E) del sistema. La característica de razón señal-a-error de un sistema digital es similar pero no idéntica en naturaleza a la razón señal-a-ruido de un sistema analógico. La relación S/E puede ser obtenida de manera más rigurosa utilizando niveles de voltaje como se muestra en la figura 22.

La razón señal-a-error se expresa mediante el cálculo del valor máximo de la señal, el valor de error y posteriormente formando su razón de cambio.

Con un sistema de cuantización en donde cada intervalo es amplitud "Q" y "n" es el número de bits el máximo nivel de señal de cero a pico es

Figura 22



$$\frac{1}{2}(2^n Q) = 2^{n-1} Q$$

y el valor máximo de señal RMS es

$$\text{Señal de Voltaje (RMS)} = \frac{2^{n-1} Q}{\sqrt{2}}$$

La señal de error se analiza mediante la examinación de la función de probabilidad del error de cuantización.

Cuando la señal de entrada tiene una amplitud alta y ancho espectro el error tiene una igual probabilidad de ser un valor entre  $+Q/2$  y  $-Q/2$ . Esto se muestra en la figura 22 con una

función de densidad de probabilidad aleatoria. Como el error es aleatorio de muestra a muestra el espectro del error es plano.

La energía contenida en el ruido se calcula realizando la integral de todos los valores de error obtenidos y multiplicando la energía contenida en un error (E) por su probabilidad P(E) dE

$$\text{Energía de Error} = \int_{-Q/2}^{+Q/2} E^2 P(E) dE \quad \text{con } P(E) = \frac{1}{Q} \quad (23)$$

$$\text{Error de Voltaje (RMS)} = \frac{Q}{\sqrt{12}} \quad (24)$$

En terminos de la razón señal-a-error

$$S/E = \frac{2^{n-1} Q}{\frac{Q}{\sqrt{12}}} = 2^n \sqrt{1.5} \quad (25)$$

Por lo tanto la razón señal-a-error es igual a  $2^n \sqrt{1.5}$ , donde "n" es el número de bits. En decibeles se expresa como

$$\text{Razón señal-a-error} \approx 6.02n + 1.76 \text{ dB}$$

Por ejemplo 16 bits dan una S/N de 98 dB; 15 bits 92 dB; 14 bits 86 dB.

### 4.4.3 Distorsión en la Cuantización

El error de cuantización es la diferencia entre el valor analógico muestreado en el tiempo y el valor de intervalo escogido para la cuantización. En el tiempo de muestreo el valor de la amplitud tiene que ser seleccionado del intervalo de cuantización mas cercano. Bajo las mejores condiciones la señal coincide con el intervalo de cuantización. En el otro caso la señal se encuentra en medio de dos intervalos. Concluyendo, el error de cuantización se encuentra limitado por  $\pm 1/2$  del intervalo en el tiempo de muestreo.

Debido a que el error de cuantización esta en función de la señal original no se puede clasificar como ruido, sino mas bien, como una forma de distorsión. Se puede formular un análisis que demuestre la magnitud de la distorsión resultante de la cuantización.

Si "Q" es el intervalo de cuantización también es la amplitud pico a pico del error de cuantización. El valor instantáneo del error de cuantización ( $\pm e$ ) tiene una distribución uniforme de 0 a "Q" y su valor pico se toma como  $e = Q/2$ . La potencia del producto de la distorsión aplicada a una unidad resistiva es

$$D = 2 \int_0^{Q/2} e^2 de \quad (26)$$

$$D = 2 \frac{(Q/2)^3}{3} = \frac{Q^3}{12} \quad (27)$$

Lo anterior demuestra que la potencia de distorsión es independiente de la amplitud de la señal de entrada pero esta relacionada a el tamaño del intervalo de cuantización.

El porcentaje de distorsión se calcula a partir del número de bits "n" en el sistema y del nivel de la señal "S" relativa a la señal máxima posible. Con "n" bits existen un número máximo de  $N = 2^n$  valores y  $2^n - 1$  intervalos (I). El número de intervalos en uso ( $I_s$ ) es

$$I_s = (2^n - 1) \frac{S}{20} \quad (28)$$

El voltaje pico a pico de la señal de nivel "S" se genera de la multiplicación del nivel de cuantización y el número de intervalos utilizados

$$pp = Q I_s$$

El voltaje pico es

$$\frac{Q I_s}{2}$$

Asumiendo una señal senoidal la potencia promedio a una resistencia unitaria  $\frac{1}{4}(Q I_s)^2$  será de

$$P = \frac{Q^2 I_s^2}{8} \quad (29)$$

La potencia de distorsión es

$$\frac{Q^2}{12}$$

Y el porcentaje de distorsión (THD%) es

$$\frac{D}{P} * 100 \%$$

$$\text{THD}\% = \frac{\frac{Q^2}{12}}{\frac{Q^2 I_s^2}{8}} * 100 \% = \frac{2}{3 I_s^2} * 100 \% \quad (30)$$

donde

$$I_s = (2^n - 1) \text{antilog}_{10} S$$

Sustituyendo valores en la ecuación el porcentaje de distorsión es nulo a 0 dB y menor a 0.1 % a -70 dB.

## 4.5 Dither

Es la naturaleza de la cuantización introducir un error. Aún que el error ocurra en un grado muy bajo su presencia debe de ser considerada en un sistema de audio de alta fidelidad. Con una señal compleja de amplitud grande existe poca correlación entre la señal y el error por lo que el error es aleatorio y produce un sonido equivalente al ruido blanco de una señal analógica. Con señales de bajo nivel el comportamiento del ruido se modifica debido a que se correlaciona con la señal dando como resultado distorsión. Para eliminar la correlación se aplica ruido analógico a la señal de audio antes de agregarla a el muestreador. Lo anterior introduce un efecto aleatorio en el error de cuantización. La señal, llamada **dither**, realiza las funciones de enmascarar el efecto de error de cuantización y provee al sistema digital con la habilidad de codificar amplitudes menores que la del bit menos significativo.

Un sistema digital debe de suprimir el error de cuantización. Como la relación va de acuerdo al número de bits en la palabra de cuantización si se incrementa el número de bits se decrementa el error de amplitud a razón de 6 dB por cada bit adicional. Esto último no es económico y no especifica cuantos bits extras son necesarios para reducir el error de

cuantización a niveles no audibles. Un alternativa a este problema es la de agregar una pequeña cantidad de ruido blanco analógico a la señal de entrada. A este ruido se le conoce como **dither**. El dither se a utilizado en diseños desde su aparición en la tecnología de video en 1950.

El dither es una pequeña cantidad de ruido blanco analógico que se agrega a la señal para remover el error de cuantización.

# 5

## Procesadores Digitales de Señales

### 5.1 Antecedentes

En los últimos 50 años ha existido un asombroso cambio en los algoritmos de procesamiento de señales y los medios computacionales utilizados para dicho fin. Antes de la Segunda Guerra Mundial hubo un periodo de procesamiento de señales clásico, caracterizado por la realización de filtros estáticos pasa-bajos, pasa-altos y pasa-bandas, que usaban únicamente el conocimiento en general del espectro de señal y ruido.

Las estadísticas de señal y ruido no eran utilizadas y la mayoría de las implantaciones utilizaban tecnología analógica. Era práctica común diseñar filtros IIR de puros polos como Butterworth, Chevyshev y diseños elípticos, tomando como operaciones primarias las de integración y diferenciación.

Computacionalmente estas técnicas estaban caracterizadas por tomar  $N$  ( $O(N)$ ) tiempo de procesamiento, donde  $N$  era el número de puntos de muestreo de la señal que estaba siendo procesada en cualquier momento.

Después de la Segunda Guerra Mundial, muchas de las aplicaciones, como los codificadores de voz, que habían sido implantados en forma analógica, se volvieron tan complejos que era difícil explorar los efectos en el desarrollo del sistema con la variación de parámetros en el diseño. Por esta razón, se introdujo el procesamiento digital de señales, primero como una técnica de simulación, sin poner atención a su utilidad en aplicaciones a tiempo real, debido a que la tecnología necesaria no se encontraba disponible.

Durante esa época existía una manipulación más refinada del espectro de información usando el limitado conocimiento de las estadísticas de señal y ruido, tales son los casos de los filtros de Wiener. Aún cuando la estadística era utilizada, esta no variaba con el tiempo, además de que no se introdujo un modelo que mostrara como se generaba la información.

Conforme el tiempo transcurría las implantaciones empezaron a ser digitales y un filtro digital IIR de puros ceros fue introducido. Las operaciones primarias, aparte del filtrado mencionado, incluía convolución, correlación y técnicas eficientes para calcular la Transformada Discreta de Fourier (DFT) tal como la FFT (Fast Fourier Transform).

Con relación directa a estas operaciones, la complejidad computacional que representaba se encontraba en el orden de  $N^2$  o  $N \log_2 N$  la cual contrastaba con la dependencia lineal con  $N$ .

Finalmente, durante los últimos 30 años, la manipulación sofisticada del espectro de información incluye el conocimiento detallado de las estadísticas de señal y ruido; se puede citar como ejemplo a los Filtros Adaptables y los Filtros de Kalman. Para estos sistemas las estadísticas pueden variar con el tiempo. Una estructura adicional fue impuesta mediante la aplicación de modelos de como se genera la información, tal es el caso del modelo de codificación lineal predictiva utilizada en el habla.

Conforme la tecnología ha ido mejorando, implantaciones digitales se han materializado para sistemas complejos, tomando las ventajas de precisión, repetibilidad, alta razón de señal-ruido, y la flexibilidad inherente en los sistemas digitales.

Se introdujeron los filtros digitales variables en el tiempo, y aparecieron las operaciones diferenciales de matrices. Las operaciones primarias fueron extendidas más allá de las operaciones de convolución y transformada de Fourier, agregando las de multiplicaciones de matrices y vectores, multiplicaciones de matrices y matrices, solución de sistemas lineales, soluciones de la menor raíz cuadrática media. Los algoritmos para la solución de estos sistemas están caracterizados por tiempos de procesamiento en el orden de  $N^3$ , por lo que llegan o exceden los rangos o límites de efectividad de los sistemas computacionales utilizados.

La documentación citada muestra la naturaleza evolutiva del campo y como la complejidad de las tareas de procesamiento digital de señales ha aumentado no solo teóricamente sino también en la tecnología de circuitos integrados.

La tarea de coordinar el enfoque teórico con el ambiente tecnológico ha recaído en las arquitecturas computacionales. Debido a la naturaleza especial de los algoritmos utilizados en el procesamiento digital de señales, han surgido procesos especializados para la mayoría de las tareas.

El campo de aplicación que se enfoca es el del procesamiento de voz con un ancho de banda limitado, que con el tiempo se ha extendido y ha requerido de toda la capacidad de procesamiento de un sistema computacional contemporáneo, aunado a un bajo costo.

Las técnicas usadas para caracterizar las arquitecturas computacionales para el procesamiento digital de señales, no son diferentes de las utilizadas en las arquitecturas computacionales en general. El énfasis que se coloca en sus atributos puede variar para satisfacer los requerimientos de la aplicación.

El desarrollo de los sistemas de computadoras digitales, tiene dos facetas contrastantes. En primer termino, mayores flujos de información son adquiridos mediante la mejora de la capacidad de procesamiento de los circuitos, esta área es gobernada por la tecnología que se dispone en el momento.

La mayoría de los fabricantes prefieren adquirir el nivel de desarrollo a través de arquitecturas secuenciales convencionales con la más avanzada tecnología. Se obtiene un flujo de información adicional mediante la explotación del paralelismo presente en varias tareas del procesamiento digital de señales. Un gran número de tareas proporcionan una cantidad considerable de paralelismo hasta en algoritmos que han sido empleados tiempo atrás.

Analizando que la combinación de tecnologías avanzadas y la implementación de arquitecturas con el paralelismo a gran escala se han logrado rangos de operación en el orden de 200 millones de operaciones de punto flotante por segundo (MFLOPS) para operandos de 32 bits.

El procesamiento digital de señales agrupa un amplio rango de aplicaciones. Algunos ejemplos son filtros digitales, codificación de voz, procesamiento de imágenes, Transformada Rápida de Fourier (FFT) y audio digital entre otras [4]-[8]. Estas aplicaciones y el procesamiento digital de señales tienen las siguientes características en común :

- \* Algoritmos matemáticamente intensivos,
- \* Operaciones a tiempo real,
- \* Implementación de muestreo de datos,
- \* Flexibilidad en el sistema.

La tabla 3 contiene ejemplos de razón de muestreo para aplicaciones típicas.

Tabla 3

Aplicación	Razón de Muestreo
Control	1 KHz
Telecomunicaciones	8 KHz
Procesamiento de Voz	8 - 10 KHz
Procesamiento de Audio	40 - 48 KHz
Video por Pixel	14 Mhz

En aplicaciones DSP, el procesador puede manejar muestreo de señales en grandes cantidades y desarrollar operaciones aritméticas a tiempo real.

## **5.2 Historia de Soluciones DSP**

En las décadas pasadas las máquinas para el procesamiento digital de señales evolucionaron para poder incluir las características mencionadas previamente. Al principio se utilizó grandes computadoras (Mainframes) para manejar señales en el área digital. Debido a el estado de la tecnología el procesamiento no se realizaba a tiempo real. Conforme a la evolución de la tecnología el procesamiento de arreglos se incorporó a la solución del problema. Debido a su flexibilidad y velocidad los procesadores en arreglo se han convertido en la solución para laboratorios de investigación y se han extendido a diversas aplicaciones.

La evolución de la tecnología de circuitos integrados propicia el diseño de microprocesadores y microcomputadoras más veloces. Como resultado, aplicaciones de procesamiento digital de señales han emigrado de los procesadores de arreglo hacia subsistemas de microprocesadores y a soluciones de un solo circuito integrado.

Esta transferencia da lugar a que el costo baje a un punto en que su aplicación es común. La creciente eficiencia de dichos circuitos de alta integración ha expandido la aplicación de los DSP a procesamiento de gráficas e imágenes y productos de audio de consumo.

## **5.3 DSP de la Familia TMS de Texas Instruments**

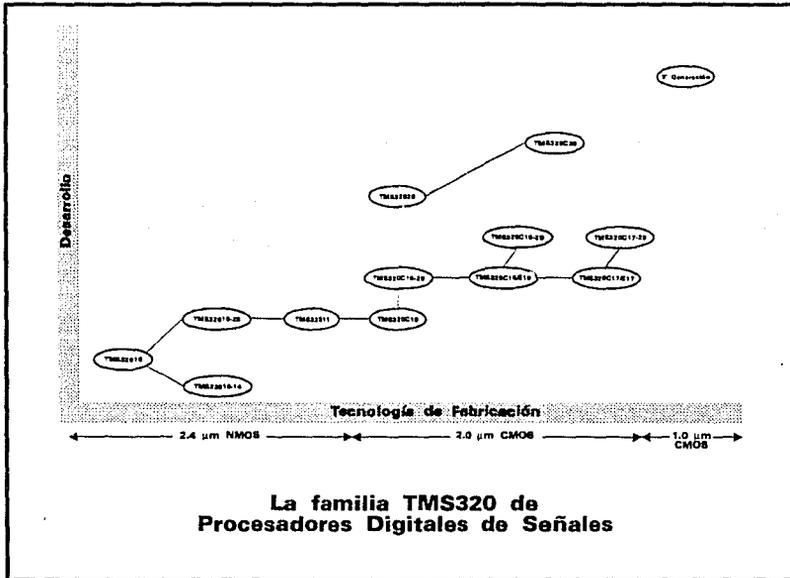
Un reciente avance en la tecnología de DSP ha sido el procesador en un solo circuito integrado tal como la familia de procesadores TMS320. La figura 23 muestra en forma gráfica la familia de procesadores TMS320.

El primer miembro de la familia, el TMS32010, fue lanzado al mercado en 1982. El circuito dio a diseñadores de sistemas la primera microcomputadora capaz de resolver 5 millones de instrucciones por segundo (5 MIPS).

La segunda generación de dispositivos incluyen al TMS32020 y al TMS320C25. El TMS320C25 resuelve 10 MIPS. La expansión de memoria, combinaciones de operaciones de multiplicación/acumulación en un ciclo, la capacidad de multiprocesamiento y las

funciones adicionales de puertos de E/S (Entrada/Salida) han hecho que el TMS320C25 sea de 2 a 4 veces más rápido que su predecesor.

Figura 23



La tercera generación, el TMS320C30, tiene una habilidad computacional de 33 millones de operaciones de punto flotante por segundo (33 MFLOPS). La familia de procesadores TMS320 ha evolucionado en la tecnología de producción.

Los beneficios de la evolución en la producción de semiconductores son las de bajo consumo de potencia, alta capacidad de procesamiento y alta densidad de integración.

### 5.3.1 La Arquitectura TMS320 Básica

Las características de un procesador digital de señales es realizar operaciones aritméticas rápidas y alto flujo de datos para manejar algoritmos matemáticamente intensivos

a tiempo real. En la familia TMS320 esto se logra mediante la aplicación de los siguientes conceptos :

- \* Arquitectura Harvard.
- \* Pipelining Extensivo.
- \* Multiplicadores dedicados en Hardware.
- \* Instrucciones DSP especiales.
- \* Ciclos de instrucción rápidos.

La capacidad de operaciones a tiempo real es aumentada mediante la velocidad absoluta del procesador al ejecutar las instrucciones. Las características anteriores combinadas con la optimización del diseño del circuito integrado le da a los dispositivos DSP ciclos de instrucción con tiempos menores a los 200 ns.

Los ciclos de instrucción de la familia TMS320 se dan en la tabla 4. Estos ciclos hacen que los procesadores TMS320 sean adecuados para aplicaciones DSP a tiempo real.

Si se toman los datos de la tabla 1 se obtiene la cantidad de instrucciones que se pueden realizar en los dispositivos TMS320 para aplicaciones a tiempo real.

Tabla 4

Dispositivo	Tiempo del ciclo
TMS320C10	160-200 ns
TMS32020	160-200 ns
TMS320C25	100-125 ns
TMS320C30	60-75 ns

### 5.3.1.1 Primer Generación de la Familia TMS320

La primera generación de la familia TMS320 incluye al TMS32010 y TMS32011, contruidos con tecnología NMOS de 2.4  $\mu\text{m}$ . El TMS320C10, TMS320C15/E15 y TMS320C17/E17 [9] son contruidos con tecnología CMOS de 1.8  $\mu\text{m}$ . Funciones claves de dichos dispositivos son:

- \* Tiempos de los ciclos de instrucción;
  - 160 ns.
  - 200 ns.
  - 280 ns.

- \* Memoria RAM interna;
  - 144 palabras.
  - 256 palabras(TMS320C15/E15, TMS320C17/E17).
- \* Memoria ROM interna;
  - 1.5 Kpalabras.
  - 4 kpalabras (TMS320C15, TMS 320C17).
- \* Memoria EPROM interna de 4 Kpalabras (TMS320E15, TMS329E17).
- \* Expansión de memoria externa hasta 4 Kpalabras.
- \* Multiplicador paralelo de 16x16 bits con resultado de 32 bits.
- \* Registro de corrimiento de barril, para recorrer datos de memoria a la ALU.
- \* Registro de corrimientos paralelo.
- \* Stack de 4x12 bits.
- \* Dos registros auxiliares para direccionamiento indirecto.
- \* Puerto serial de doble canal (TMS32011, TMS320C17, TMS320E17).
- \* Interface para coprocesador (TMS320C17, TMS320E17).

### 5.3.1.2 Segunda Generación de la Familia TMS320

La segunda generación de TMS320 consta del TMS32020 [10] y el TMS32C25 [11]. La arquitectura de estos procesadores se desarrolla del TMS32010. Funciones claves de estos dispositivos son :

- \* Tiempos de los ciclos de instrucción;
  - 100 ns (TMS320C25).
  - 200 ns (TMS32020).
- \* 4 Kpalabras de ROM interna enmascarable (TMS320C25).
- \* Memoria RAM interna de 544 palabras.
- \* 128 Kpalabras de espacio de memoria para programa.
- \* 8 registros auxiliares con una unidad aritmética dedicada.
- \* Stack de 8 niveles.
- \* Buffer de puerto serial completamente estático.
- \* Estados de espera para comunicación con circuitos más lentos.
- \* Puerto serial para multiprocesamiento o interface con Codecs.
- \* DMA concurrente usando una operación de detención extendida.
- \* Modos de direccionamiento de inversión de bits para Transformada Rápida de Fourier (TMS320C25).
- \* Aritmética de precisión extendida y soporte de filtrado adaptable (TMS320C25).
- \* Operaciones de alta velocidad para instrucciones MAC/MACD para memoria externa (TMS320C25).
- \* Bit de carry acumulador y operaciones relacionadas (TMS320C25).

### **5.3.1.3 Tercera Generación de la Familia TMS320**

El TMS320C30 [12] es la tercera generación de la familia TMS320. Con una velocidad de 33 MFLOPS el TMS320C30 excede la capacidad de procesamiento de cualquier DSP disponible. Funciones claves del dispositivo son :

- \* Tiempos de los ciclos de instrucción de 60 ns.
- \* Dos bloques de memoria RAM de 1K x 32 bits de doble acceso y ciclo sencillo.
- \* Un bloque de memoria ROM de 4K x 32 bits de doble acceso y ciclo sencillo.
- \* Memoria Cache de 64x32 bits.
- \* Instrucciones y datos de 32 bits, y 24 bits de direcciones.
- \* Multiplicador de 32/40 bits entero y de punto flotante.
- \* ALU de 32/40 bits entero y lógico.
- \* Registro de corrimiento de barril de 32 bits.
- \* 8 registros de precisión extendida.
- \* Dos generadores de direcciones con ocho registros auxiliares.
- \* Controlador DMA interno para operaciones de CPU y E/S concurrentes.
- \* Bus periférico y módulos para fácil adaptación.
- \* Soporte de lenguaje de alto nivel.
- \* Soporte de instrucciones para multiprocesamiento.

### **5.3.2 Herramientas de Desarrollo y Soporte**

Los procesadores digitales de señales son microprocesadores de propósito específico. Como cualquier otro microprocesador sin buenas herramientas de desarrollo y soporte técnico es muy difícil integrarlo en un sistema.

La familia TMS320 tiene una variedad de herramientas de desarrollo disponibles. Estas herramientas son módulos de bajo costo que cubren las necesidades de evaluación de aplicaciones, ensambladores/ligadores, simuladores y emuladores completos de hardware. A continuación se describe brevemente estas herramientas.

### **5.3.2.1 Herramientas de Software**

Ensambladores/ligadores y simuladores en software están disponibles en PC (Computadora Personal) y VAX para que los usuarios desarrollen algoritmos para los DSP TMS320. Sus características son :

- \* Ensamblador/ligador: El macroensamblador traduce lenguaje ensamblador a código ejecutable.
- \* Simulador: Simula la operación del dispositivo en software para permitir una verificación del programa.

### **5.3.2.2 Herramientas de Hardware**

Los módulos de evaluación y herramientas de emulación están disponibles para emulación del circuito, desarrollo y pruebas de los algoritmos para el DSP en un ambiente real.

Módulos de Evaluación (EVM): El EVM es un modulo compuesto en una tarjeta que contiene herramientas necesarias para evaluar el dispositivo, así como para proporcionar la emulación de los circuitos.

## **5.4 La Arquitectura del TMS320C25**

El Procesador Digital de Señales TMS320C25 [13] y [14] es una microcomputadora con una arquitectura Harvard interna de 32 bits, y una interface externa de 16 bits. Es una versión en CMOS, la cual es compatible con el microprocesador TMS32020, pero tiene una razón de ejecución de instrucciones del doble, e incluye propiedades adicionales de hardware y software. El conjunto de instrucciones del TMS320C25 es una extensión superior al del TMS32010 y TMS32020, manteniendo la compatibilidad de código con ellos. Además, es completamente compatible con el TMS32020, por lo que programas que estuvieran diseñados para el TMS32020 pueden ejecutarse sin ninguna modificación en el TMS320C25. Algunas de las características más relevantes del TMS320C25 son :

- \* Una ALU y acumulador de 32 bits.

- \* Un tiempo de ciclo de instrucción de 100 ns.
- \* Una multiplicación/acumulación de un solo ciclo.
- \* Uso de tecnología CMOS de baja potencia.
- \* ROM interno de 4 K de 16 bits.
- \* RAM interna de 544 palabras.
- \* Espacio de memoria para datos/programa de 128 K palabras.
- \* Ocho registros auxiliares con unidad aritmética dedicada.
- \* Stack de ocho niveles a un nivel hardware.
- \* Puerto Serial estático de doble buffer.
- \* DMA concurrente que utiliza una operación de espera extendida.
- \* Direccionamiento tipo reversión de bits para FFT.
- \* Fabricación en CMOS de 1.8  $\mu\text{m}$  empaquetado en un PLCC de 68 terminales.

El ciclo de instrucción de 100 ns proporciona una alta capacidad de flujo de datos para muchas aplicaciones. Como la mayoría de las instrucciones del TMS320C25 se ejecutan en un ciclo, puede llegar a realizar 10 millones de operaciones en un segundo.

El TMS320C25 incluye instrucciones para realizar transferencias de datos del espacio de programa al espacio de memoria. Externamente los espacios para memoria de datos y programa se encuentran multiplexada en el mismo bus, con el fin de maximizar el rango de direcciones para ambos espacios, así como también para reducir el número de terminales del circuito integrado. Internamente, la arquitectura del TMS320C25 maximiza el poder de procesamiento mediante la utilización de dos estructuras de bus separadas, programa y datos, para ejecución a alta velocidad.

La ejecución de programas en el dispositivo toma la forma de una cola (pipeline) de instrucción de tres niveles, busca-decodifica-ejecuta. La cola es invisible para el usuario excepto en los casos en la que debe de ser rota, tal como para instrucciones de ramificación. En estos casos, el tiempo de la instrucción toma en cuenta la duración del vaciado y llenado de la cola.

El circuito provee dos grandes bloques de memoria de acceso aleatorio (RAM), con un total de 544 palabras, uno de los cuales puede ser configurado como memoria de programa o memoria de datos. También tiene incluida la facilidad de acceso externo de 64 KPalabras de memoria RAM, para aplicaciones que necesiten una gran capacidad de memoria de datos. Existen 4 KPalabras internos de memoria de lectura (ROM) que están diseñados como memoria de programa, así como también tiene la posibilidad de direccionar externamente 64 KPalabras de memoria de programa. Se pueden ejecutar programas de grandes dimensiones a altas velocidades en el espacio de memoria disponible. Los programas también pueden ser cargados de memoria externa de baja velocidad a la memoria RAM interna para ser ejecutados a mayor velocidad.

El TMS320C25 también tiene incorporado un temporizador por Hardware y la capacidad para transferencia de bloques de datos.

El diagrama de la figura 24 muestra los bloques principales y las vías de flujos de datos del procesador TMS320C25. También muestra las terminales de conexión del circuito integrado.

La arquitectura del TMS320C25 esta construida alrededor de los buses de programa y de datos. El bus de programa transmite los códigos de instrucción y los operandos inmediatos de la memoria de programación. El bus de datos interconecta elementos tales como la Unidad Aritmética y Lógica Central (CALU) y los registros auxiliares a la memoria de datos. Los buses de programa y de datos pueden transmitir datos de la memoria RAM interna y de la memoria de programa externa o interna al multiplicador, todo en un solo ciclo por medio de las operaciones de multiplicación/acumulación.

Existe un alto grado de paralelismo en el dispositivo, por ejemplo, mientras datos se están procesando en la CALU, operaciones aritméticas se pueden implementar en la Unidad Aritmética de Registros Auxiliares (ARAU). Tal paralelismo da como resultado un conjunto de operaciones aritméticas, lógicas y de manipulación de bits muy poderosa, que pueden ser ejecutadas en un solo ciclo.

- 1) Transformación de tiempo a frecuencia de dos canales de 128 puntos cada uno en un solo procesador TMS320C25.
- 2) Transformación de frecuencia a tiempo de dos canales de 128 puntos cada uno en un solo procesador TMS320C25.
- 3) Tiempo aproximado de transformación para ambos canales de 2.27 ms.
- 4) Espacio aproximado en memoria interna de programa de 3400 palabras.
- 5) Utilización exclusivamente de memoria interna de datos.
- 6) Un solo puerto de acceso para la entrega de resultados de la transformada de tiempo a frecuencia.
- 7) Utilización de la interface serial para la lectura del Convertidor Analógico/Digital (ADC) y un puerto adicional para la entrega de datos al Convertidor Digital/Analógico (DAC).

# 7

## Diseño y Desarrollo de una Arquitectura para PAD

### 7.1 Introducción

El diseño de todo el circuito se basó en realizar una arquitectura sencilla, eficiente y flexible. Se buscó que el número de componentes fuera el mínimo necesario para el adecuado funcionamiento de tal manera que el costo, complejidad y probabilidad de falla a partir del diseño del circuito fueran muy bajas.

Entre las ventajas que tiene el circuito debido a su sencillez destaca el de diagnóstico de fallas cuando ocurra un problema el cual es fácil de analizar por el reducido número de componentes que se tiene que revisar, además de que la mayoría son componentes de bajo grado de complejidad facilitando así su funcionamiento.

Hay que mencionar también que el circuito está planeado para que se utilice como una plataforma básica en el diseño de aplicaciones específicas de Procesamiento Digital de Señales de Audio (ADSP). Se puede implementar como una estructura de un circuito microcontrolado complejo hasta como la parte central de hardware de un sistema basado en computadora en conjunto con un software de alta complejidad de procesamiento. Al circuito también se le puede tomar como un módulo fundamental para agregarle diferentes tipos de dispositivos (memorias, codificadores, transmisores, etc.) para utilizarlos en las implementaciones anteriormente mencionadas.

### 7.2 Características de la Arquitectura

Número de Canales	2
Ancho de Banda por canal	44.1 KHz
Número de bits utilizados por canal	15

Procesador Digital de Señales  
Convertidor Analógico/Digital  
Convertidor Digital/Analógico  
Resultados Obtenidos

Analógico/Digital  
Digital/Analógico

TMS320C25  
DSP56ADC16  
AD7528

Señal en Frecuencia  
Señal en el Tiempo

## 7.3 Características del ADC DSP56ADC16

Convertidor Analógico/Digital Sigma-Delta de 16 bits.

El DSP56ADC16 es un convertidor analógico/digital (A/D) de 16 bits, lineal, implementado en un solo circuito integrado, con sobre muestreo, proporcionando muestras a razón de 100 KHz. Utiliza tecnología Sigma-Delta [15] con manejo de ruido de tercer orden y sobre muestreo de 64 bits, el cual tiene un rango dinámico de 96 dB y una razón de señal a ruido de 90 dB para señales en un ancho de banda de 0 a 45.5 KHz con un rizado interno de 0.0001 dB. No requiere de filtros antialiasing y circuitos de muestreo-almacenamiento debido a que son una parte inherente de la tecnología Sigma-Delta. El convertidor también proporciona una salida antes del filtro final de decimación de respuesta impulso finita (FIR Finite Impulse Response) para aplicaciones que requieran de alta velocidad, menor retardo de grupo y 12 bits de exactitud para niveles de CA.

Funciones claves del DSP56ADC16 son [19]:

- \* Resolución de salida de 16 bits del filtro FIR a 100 KHz.
- \* Resolución de salida de 12 bits del filtro Comb a 400 KHz.
- \* Rango Dinámico de 96 dB.
- \* Razón Señal a THD (Total Harmonic Distortion) de 90 dB.
- \* Razón Señal a Ruido de 90 dB.
- \* Rizado interno menor a 0.001 dB.
- \* Máxima relación de salida (FIR: 100 KHz; Comb: 400 KHz).
- \* Máxima relación de entrada: 6.4 MHz.
- \* Máxima frecuencia de reloj interna: 12.8 MHz.
- \* Estabilidad en CD: 10 bits.
- \* Alimentación de Voltaje: +5 Volts  $\pm$  10 %.
- \* Alimentación de Corriente: menor a 100 mA.
- \* Encapsulado de 20 terminales DIP de plástico, cerámica y montaje de superficie.
- \* Encapsulado en un solo circuito integrado.
- \* Filtros Digitales internos.
- \* Interface serial de salida.
- \* Interface sencilla a procesadores DSP56001, 7720 y 320XX.

\* Entradas completamente diferenciales.

## **7.4 Operación del Modulo de Transformación Directa**

### **7.4.1 Consideraciones del Diseño**

Podemos decir que el circuito de transformación es la parte de captación de información del sistema, debido a que toma las señales del mundo analógico y las convierte a información para los sistemas digitales.

El circuito se diseño de tal manera que cada componente depende de un número reducido adicional de elementos de esta forma simplificando su operación.

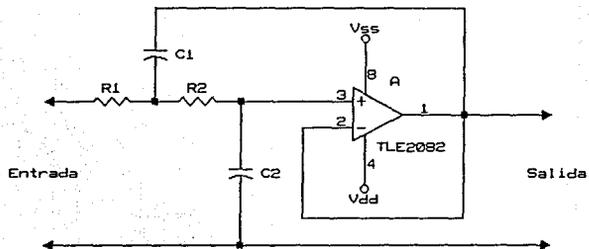
El caso que es más representativo de la condición anterior es la del procesador. Entre las características de operación se tenía que hacer la elección de trabajar con o sin memoria de programa y de datos externa. La decisión fue la de trabajar exclusivamente con memoria interna porque evitaba el tener que implementar un sistema de decodificación de direcciones para la memoria externa y el agregar memoria RAM (memoria de datos) y ROM (memoria de programa) externa. Además hay que tomar en cuenta que la memoria externa tiene que ser de alta velocidad de lo contrario introduciría tiempos de espera al procesador causando que el tiempo disponible para la FHT se redujera a la mitad.

Al tomar la decisión de trabajar exclusivamente con memoria interna, automáticamente se hizo el compromiso de adaptarse a la cantidad de memoria disponible y a el sistema de dispositivos para intercambio de información. De lo anterior se planteo el dividir la memoria interna de datos en cinco partes, que el tamaño de cada transformada fuera de 128 muestras y que el programa fuera una combinación de ejecución en línea y ejecución en ciclo.

Por último, la elección del algoritmo tuvo como consecuencia que el sistema procesador constara solamente de un DSP para realizar la transformación de los dos canales. De haber escogido otro algoritmo (como Fourier) se hubiera requerido de un DSP por canal ya fuera por el espacio requerido de memoria o por la velocidad de procesamiento.

### **7.4.2 Operación del Circuito**

Para poder explicar la operación, características y protocolo del circuito se seguirá el recorrido de la señal (apéndice F).



R1 = 10 KOhm  
 R2 = 10 KOhm  
 C1 = 680 picoF  
 C2 = 1 nanoF

Circuito:

Filtro Paso-Bajos a 22 KHz

Title

Proyecto de Tesis

Size Document Number

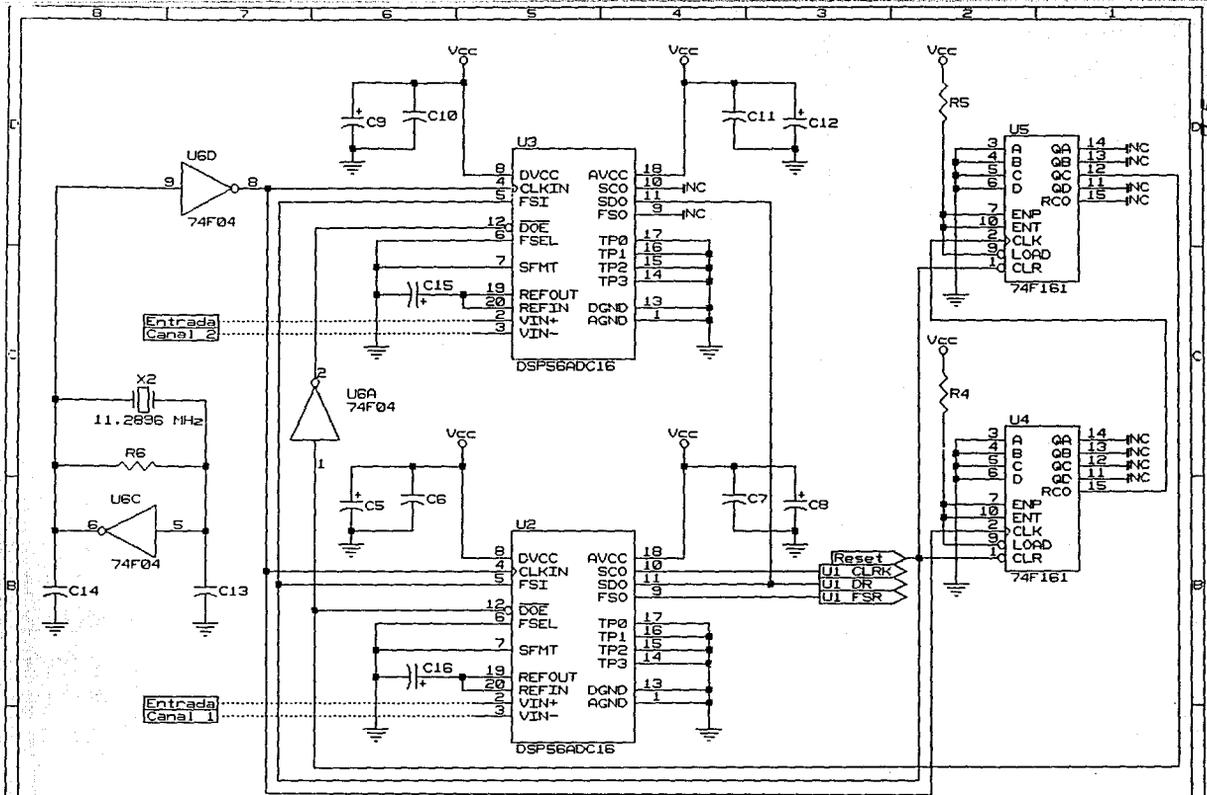
A

Diagrama 1

REV

01

Date: January 31, 1993 Sheet 1 of 1



9-5

- R4 = 1 KOhm
- R5 = 1 KOhm
- R6 = 1 MOhm
- C5 = 100 microF elec.
- C6 = 0.1 microF
- C7 = 100 microF elec.
- C8 = 100 microF elec.
- C10 = 0.1 microF
- C11 = 0.1 microF
- C12 = 100 microF elec.
- C13 = 33 pF
- C14 = 33 pF
- C15 = 100 microF elec.
- C16 = 100 microF elec.

Circuito :	
Conversion de tiempo a frecuencia	
Title	
Proyecto de Tesis	
Size Document Number	REV
A	01
Date: January 31, 1983	Sheet 1 of 1

### **7.4.2.1 Primera Etapa**

La primera etapa del circuito toma la señal analógica que debe de ser entre 0 y 2 volts. Esta señal se filtra paso-bajo a 22 KHz para evitar el efecto de aliasing a través de un filtro activo con un amplificador operacional (diagrama 1). Existe un filtro independiente por cada canal.

### **7.4.2.2 Segunda Etapa**

La segunda etapa consiste de la conversión analógica a digital mediante un ADC de 16 bits (diagrama 2).

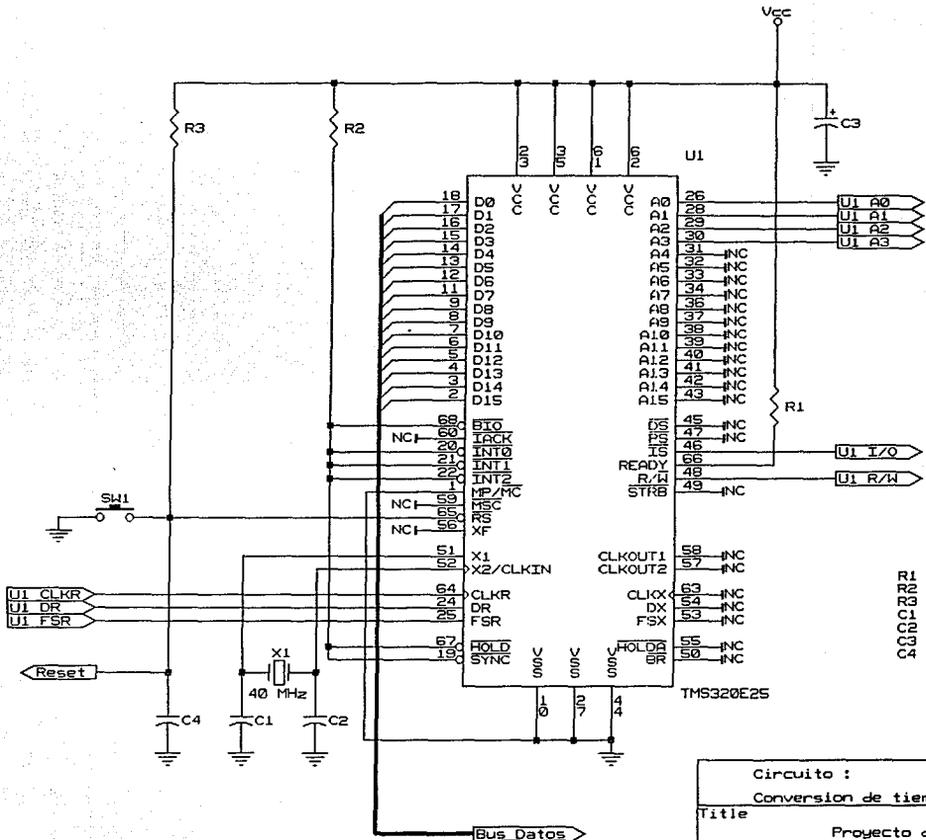
El convertidor, del cual ya se mencionaron sus características, acepta un rango de voltaje de -2 a 2 volts y proporciona su equivalente en formato digital de 16 bits. De los 16 bits del convertidor se usan exclusivamente 15 bits y esto se debe a que al aplicar la FHT los resultados pueden ser negativos y requieren de un bit más para su representación.

A partir de la explicación anterior surge la restricción para que la señal de entrada tenga solo dos volts de variación pico a pico.

Como se mencionó con anterioridad la interface entre el ADC y el DSP se realiza en forma serial y debido a que solo existe uno de dichos puertos es necesario multiplexar la salida de los dos canales.

Para implementar la multiplexión se tuvo que duplicar la frecuencia de muestreo de los ADC e incluir dos contadores binarios de 4 bits cada uno y una compuerta inversora de tal manera que la salida de un canal estuviera activa durante 128 ciclos de reloj (tiempo de transmisión de 16 bits en forma serial) y posteriormente se pasará al otro canal durante 128 ciclos de reloj.

La comunicación serial entre los ADC's y el DSP requiere de tres señales las cuales son: una señal de transmisión de los ADC's que se convierte en recepción para el DSP; una señal de sincronía proporcionada por los ADC's; y una señal de reloj de transmisión que es proporcionada por los ADC's.



- R1 = 1 K $\Omega$ -m
- R2 = 1 K $\Omega$ -m
- R3 = 10 pF
- C1 = 10 pF
- C2 = 10 pF
- C3 = 0.47 micro of elec.
- C4 = 0.47 micro of elec.

Circuito :	
Conversion de tiempo a frecuencia	
Title	
Projecto de Tesis	
Size	Document Number
A	Diagrama 3
Date: January 31, 1993	Sheet 1 of 1
3	2
	REV
	01

Debido a que las señales se tienen que multiplexar se designó a el ADC del canal uno como la generadora de las señales de sincronía y de reloj por lo que las señales del canal dos no se toman en cuenta.

Hay que hacer notar una situación relacionada con la multiplexión. En un momento determinado puede surgir la duda de ¿porqué no se utilizó solamente un ADC y se multiplexaron las señales de ambos canales?. La razón se debe a que la tecnología utilizada en los ADC (Sigma-Delta) no permite la multiplexión de las señales mediante el mismo convertidor.

### **7.4.2.3 Tercera Etapa**

El componente central de la tercera etapa es el DSP (diagrama 3) y opera en base a las interrupciones generadas por el puerto serial. Cada vez que el registro interno de recepción del puerto serial se llena se genera una interrupción. La rutina de software encargada de darle solución a la interrupción inicia enviando el resultado de la FHT previa que se encuentra en la localidad de memoria de datos que se va a utilizar hacia un puerto paralelo. La operación del puerto paralelo se describe en la cuarta etapa.

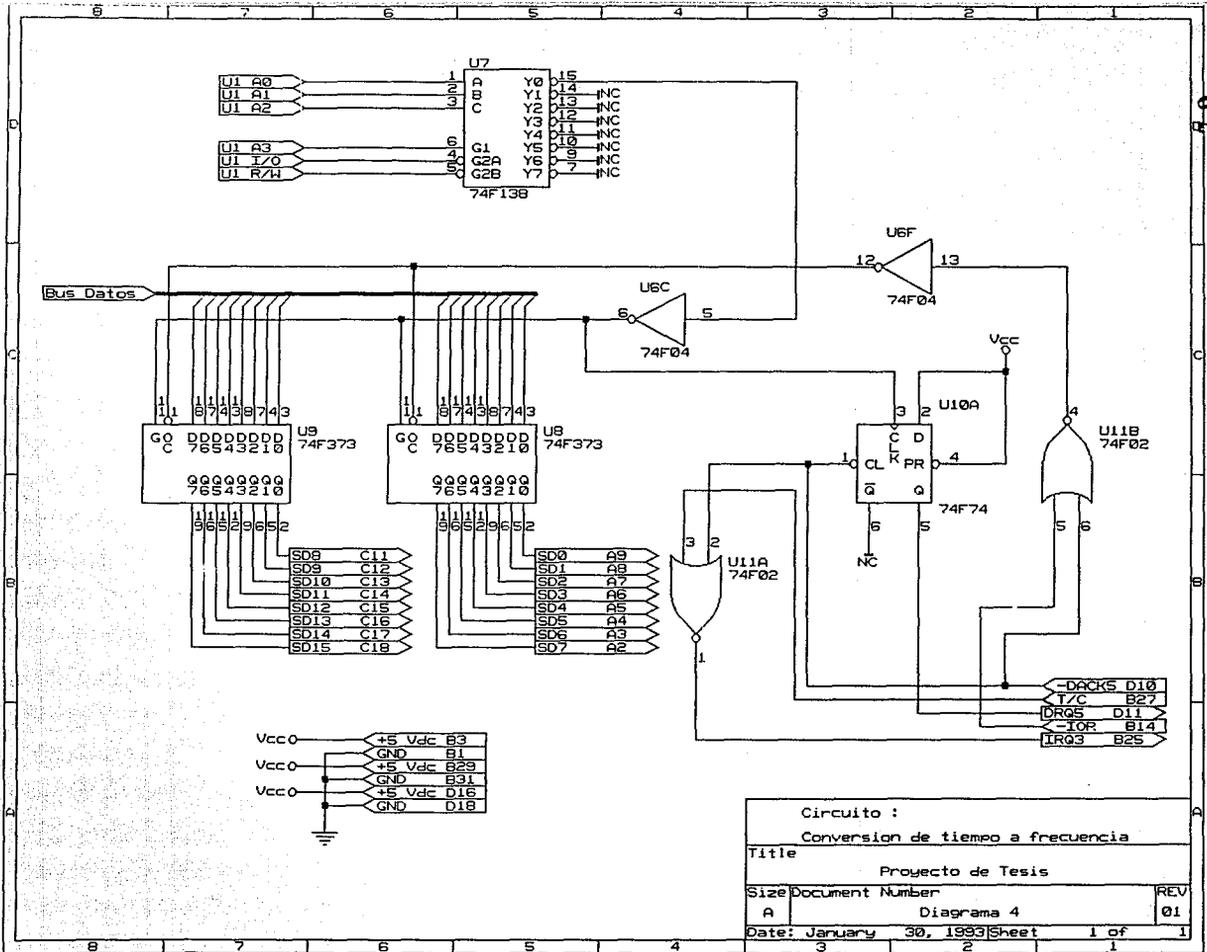
Después de desocupar la localidad de memoria de datos la rutina lee el dato del registro serial y lo almacena en la localidad de memoria correspondiente.

Cuando el buffer de datos del DSP se llena el procesamiento del hardware no sufre ninguna modificación, su operación sigue siendo la misma.

### **7.4.2.4 Cuarta Etapa**

El bus de direcciones, bus de datos y señales de control del DSP conforman parte de la cuarta etapa. Los demás componentes consisten en un decodificador para activar los circuitos de almacenamiento para la muestra enviada por el DSP, dos circuitos de almacenamiento de ocho bits cada uno (dos latches de 8 bits que en total forman los 16 bits) los cuales almacenan el último dato enviado por el DSP y un Flip-Flop para avisar a el dispositivo externo del estado de la transferencia de datos (diagrama 4).

La operación de la etapa inicia cuando el DSP genera las direcciones y señales necesarias para que el decodificador active las unidades de almacenamiento temporal y habilite el Flip-Flop. La habilitación del Flip-Flop sirve para avisar a el dispositivo externo que hay un dato para ser leído.



Circuito :		
Conversion de tiempo a frecuencia		
Title		
Projecto de Tesis		
Size	Document Number	REV
A	Diagrama 4	01
Date:	January 30, 1993	Sheet 1 of 1

La activación de las unidades de almacenamiento temporal se realiza a través de la terminal de escritura. En dicho momento se toma la información del bus de datos del DSP.

El protocolo que se siguió para implementar la lectura de información por un dispositivo externo es aquella utilizada por microcomputadoras en su sistema de DMA (Direct Memory Access). Dicho protocolo funciona de la siguiente manera :

1) Se debe de generar una petición de transferencia DMA. La petición la genera el primer Flip-Flop.

2) Cuando el controlador de DMA esta listo envía una señal de enterado (Acknowledge). La señal de enterado es utilizada como la señal de lectura de las unidades de almacenamiento temporal y para deshabilitar el primer Flip-Flop.

3) Existe otra señal en el controlador de DMA la cual se activa cuando el espacio de memoria de la microcomputadora asignada a el procesa de DMA se termina. Dicha señal en conjunto con la de enterado se combinan para generar una interrupción a la microcomputadora.

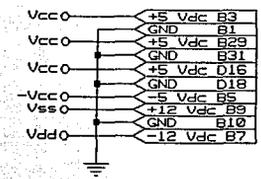
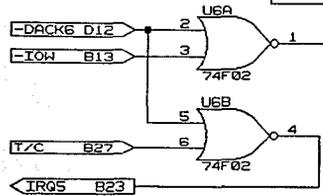
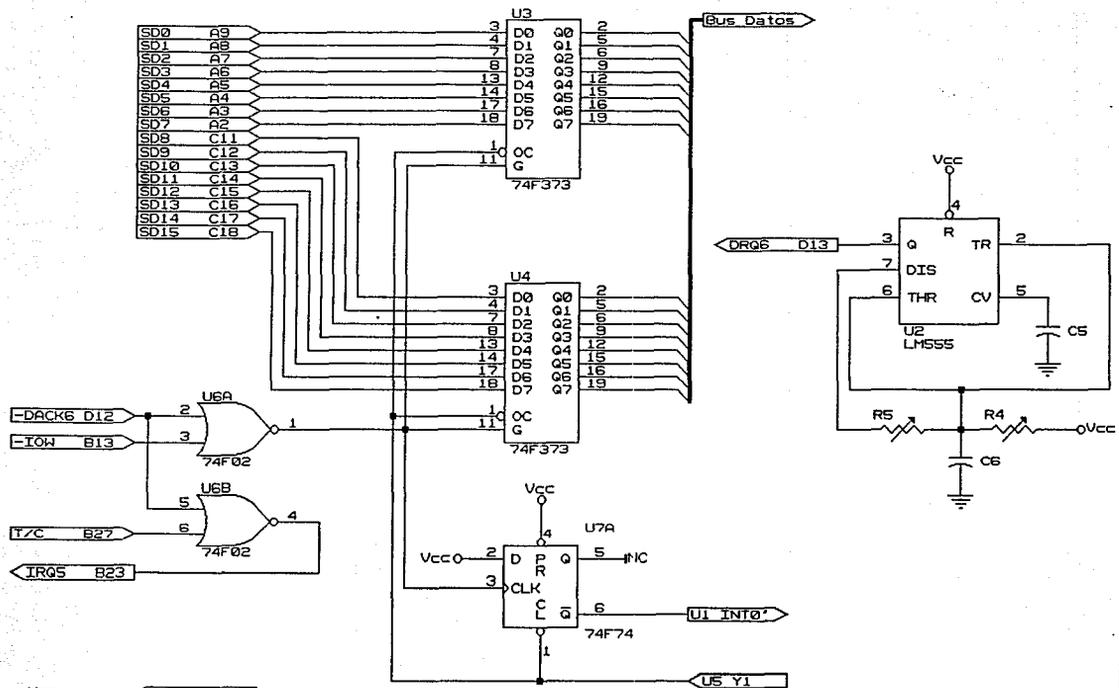
## **7.5 Operación del Modelo de Transformación Inversa**

### **7.5.1 Consideraciones del Diseño**

Las consideraciones que se tomaron para el diseño del circuito de transformación de frecuencia a tiempo son las mismas que para el circuito de transformación de tiempo a frecuencia. La única diferencia que existe es que los resultados que se generan consisten de la información analógica original a partir de los resultados del sistema de conversión analógico/digital. Por lo que podemos concluir que este circuito realiza la operación de conversión digital/analógica.

### **7.5.2 Operación del Circuito**

Para poder explicar la operación, características y protocolo del circuito se seguirá el recorrido de la señal (apéndice G).



R4 = 10 KOhm  
 R5 = 10 KOhm  
 C5 = 0.01 microf  
 C6 = 0.0033 microf

Circuito :		
Conversion de frecuencia a tiempo		
Title		
Proyecto de Tesis		
Size	Document Number	REV
A	Diagrama 5	01
Date:	January 31, 1993	Sheet 1 of 1

7-11

### **7.5.2.1 Primera Etapa**

La primera etapa del circuito esta compuesta por un oscilador creado a partir de un Timer 555. Este circuito genera una señal cuadrada con una frecuencia de 88.2 KHz la cual se utiliza para iniciar el proceso de DMA pero con el flujo inverso de los datos. Esto es, cuando se genera una petición la microcomputadora toma un dato de memoria y lo coloca en dos registros de almacenamiento temporal (diagrama 5). Al mismo tiempo se activa un Flip-Flop el cual tiene la función de generar una interrupción en el DSP.

Al igual que en el circuito de tiempo a frecuencia, cuando la memoria de la microcomputadora se a agotado el sistema genera una interrupción para la microcomputadora avisándole de lo ocurrido.

Cuando el DSP entra en la rutina para darle solución a la interrupción este inicia enviando el resultado de la FHT previa que se encuentra en la localidad de memoria de datos que se va a utilizar hacia un puerto paralelo.

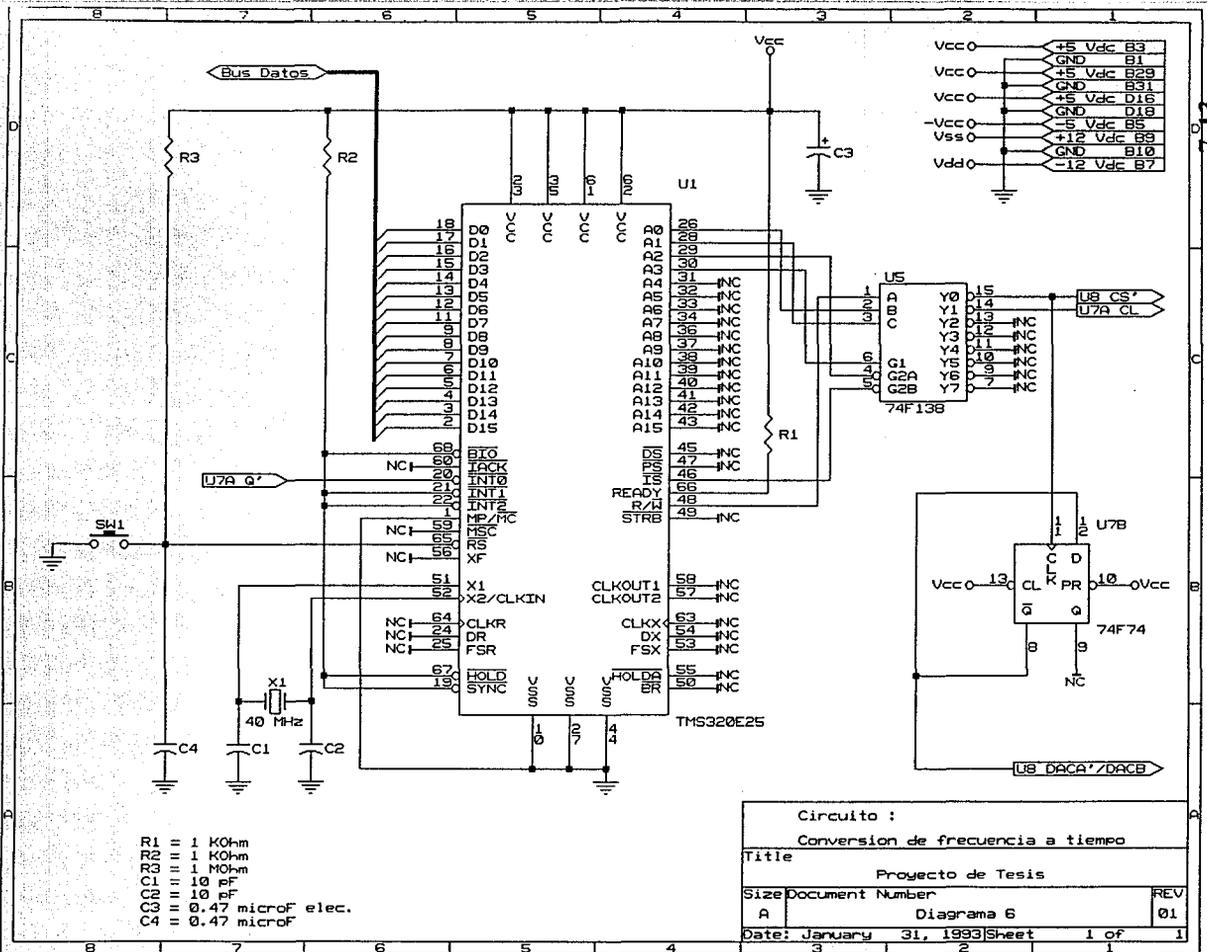
Después lee a través de otro puerto paralelo el dato que se encuentra en los registros temporales y lo almacena en la localidad de memoria correspondiente. Al momento de generar la señal de lectura también se deshabilita el Flip-Flop.

Cuando el buffer de datos del DSP se llena el procesamiento del hardware no sufre ninguna modificación, su operación sigue siendo la misma.

Para este circuito la parte de descodificación de direcciones y señales de control es más extensa debido a que se utilizan dos puertos paralelos cuya función ya ha sido mencionada y un Flip-Flop el cual se utiliza para multiplexar la transferencia de información entre los dos canales (diagrama 6).

### **7.5.2.2 Segunda Etapa**

La segunda etapa esta compuesta por dos DAC's dobles de 8 bits cada uno (diagrama 7). Para poder formar los 16 bits del sistema y realizar una sola escritura se utilizan los DAC's primarios para el canal uno y los DAC's secundarios para el canal dos.



- R1 = 1 KOhm
- R2 = 1 KOhm
- R3 = 1 MOhm
- C1 = 10 pF
- C2 = 10 pF
- C3 = 0.47 microF elec.
- C4 = 0.47 microF

Circuito :		
Conversion de frecuencia a tiempo		
Title		
Projecto de Tesis		
Size Document Number		REV
A	Diagrama 6	01
Date: January 31, 1993	Sheet 1 of 1	

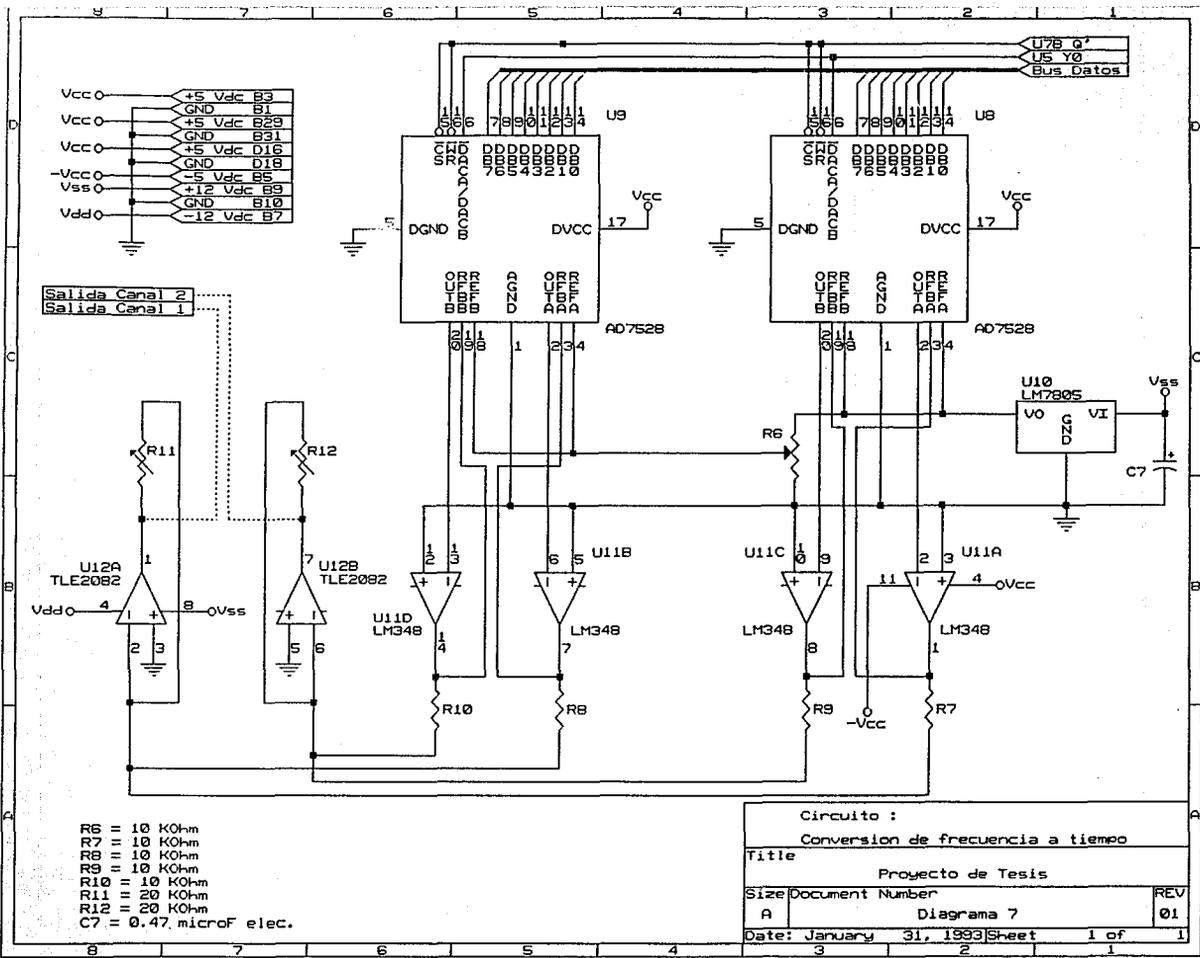
7-13

La fuente de referencia para los voltajes de salida se logra mediante un regulador de voltaje LM7805 para el Byte más significativo y con el voltaje del regulador y una resistencia variable para lograr el bajo voltaje del Byte menos significativo.

La salida de cada par de DAC's es alimentada a un amplificador operacional en configuración de sumador de tal manera que se sumen el voltaje del Byte más significativo con el voltaje del Byte menos significativo.

### **7.5.2.3 Tercera Etapa**

La tercera y última etapa del circuito toma la señal analógica que proporciona el sumador de voltaje y la alimenta a un filtro paso-bajo a 22 KHz(diagrama 1) para eliminar las armónicas superiores a la frecuencia de corte. Existe un filtro independiente por cada canal.



- R6 = 10 KOhm
- R7 = 10 KOhm
- R8 = 10 KOhm
- R9 = 10 KOhm
- R10 = 10 KOhm
- R11 = 20 KOhm
- R12 = 20 KOhm
- C7 = 0.47 microF elec.

Circuito :		
Conversion de frecuencia a tiempo		
Title		
Proyecto de Tesis		
Size	Document Number	REV
A	Diagrama 7	01
Date:	January 31, 1993	Sheet 1 of 1

# 8

## Resultados y Evaluación

### 8.1 Introducción

El principal bloque de interés en la elaboración del proyecto consistió en la creación de un algoritmo para la FHT y el diseño e implementación de una nueva arquitectura basada en el TMS320C25.

Debido a la falta de los componentes principales de hardware y a la necesidad de terminar el proyecto de la mejor manera posible, la elaboración se realizó mediante simuladores de software y emuladores de hardware, los cuales han demostrado que son capaces de brindar resultados con un alto grado de exactitud sin la necesidad de construir todas las partes de hardware requeridas.

### 8.2 Evaluación del Algoritmo FFT para la Solución del Proyecto

La FFT es la herramienta matemática mas utilizada en el mundo del procesamiento digital de señales, y en el capítulo 3 se desarrollaron las bases para la creación de un algoritmo capaz de evaluarla con un procesador de la familia TMS320. Sin embargo, es necesario evaluar el comportamiento de la FFT para la solución del problema.

La característica principal de tiempo del proyecto es la frecuencia de muestreo de 44.1 KHz, lo que implica un lapso de tiempo entre muestras de 22.67573696  $\mu$ s. Y las limitaciones que el componente principal de hardware (TMS320C25) impone son: ciclos de instrucción de 100 ns., tamaño de la memoria interna de datos de 544 palabras y tamaño de la memoria interna de programa de 4096 palabras.

La tabla 5 brinda una evaluación de la operación de varios algoritmos FFT.

Tabla 5

Datos tomados de: Digital Signal Processing Applications with the TMS320 Family Texas Instruments [14].					
Algoritmo	FFT		Ciclos	Ejecución	
	Tamaño	Tipo		Reloj	Tiempo
Radix-2	128 Ptos.	Cíclico	21879	10 MHz	2.1879 ms
Radix-2	256 Ptos.	Cíclico	42416	10 MHz	4.2416 ms
Radix-2	256 Ptos.	En línea	22595	10 MHz	2.2595 ms
Radix-2	1024 Ptos.	En línea	159099	10 MHz	15.9099 ms
Radix-4	256 Ptos.	En línea	15551	10 MHz	1.5551 ms

Combinando la tabla 5 y las condiciones mencionadas se puede tomar una decisión en cuanto a que algoritmo FFT utilizar, o si se tiene que buscar otra alternativa. La tabla 6 muestra la combinación de todos los parámetros descritos. El valor de la columna A se obtuvo del producto del número de muestras y el intervalo de tiempo entre muestras.

Tabla 6

Algoritmo (Ptos/tipo)	[ A (ms) ]	[ B (Palabras) ]	[ C (Palabras) ]
Radix-2( 128/Cíclica)	2.902494331	256	400
Radix-2( 256/Cíclica)	5.804988662	512	450
Radix-2( 256/En Línea)	5.804988662	512	54000*
Radix-2( 1024/En Línea)	23.219954654	2048*	61000*
Radix-4( 256/En Línea)	5.804988662	512	32000*

[A] Tiempo disponible para realizar la transformada.  
 [B] Espacio requerido de memoria interna de datos.  
 [C] Espacio requerido de memoria interna de programa (Aproximado).  
 \* Excede la capacidad interna.

De la tabla 6 se concluye que :

- 1) Todos los algoritmos tienen la capacidad de velocidad si se trabajan con memoria interna de datos y programa.
- 2) Solo los algoritmos de Radix-2(256/En Línea) y Radix-4(256/En Línea) tienen la capacidad de velocidad si se trabajan con memoria externa de datos y programa.

- 3) Se necesitarían dos procesadores para poder realizar la implementación con dos canales.
- 4) Para poder realizar la operación a tiempo real se requiere de memoria externa para el almacenamiento de las muestras, lo cual implica el agregar circuitería adicional para el control de dicha memoria.

Con las conclusiones realizadas y debido a las especificaciones del diseño se opto por implementar otro tipo de transformada. La solución que se escogió permite el cálculo de dos transformadas en el lapso de tiempo en que se recopilan todas las muestras de dos canales a tiempo real. Además el espacio de memoria utilizado es la mitad que el requerido por la FFT, de tal manera que se evita la memoria externa de datos. Otro punto es que el algoritmo cabe en la memoria interna de programa, por lo que la velocidad de procesamiento no se reduce.

Otro tipo de implementación de la DFT que es de alta velocidad es la PFT (Prime-Factor FFT, FFT de factor primo) el cual también se comparo con el algoritmo final de la Transformada de Hartley. La comparación se baso en una tabla con el número de operaciones aritméticas de adición y multiplicación que se necesitan para realizar una transformada de N puntos. La tabla se obtuvo de la bibliografía [16] se tomo de ella para la comparación el número de sumas y multiplicaciones que se necesitan para una transformada de PFT de cuatro factores complejos de 126 puntos, que es el número mas cercano que contenía la tabla, con el número equivalente de operaciones necesarias para la transformada de Hartley de 128 puntos.

	Suma	Multiplicación	Operaciones
PFT 126 puntos	2724	568	3292
FHT 128 puntos	1154	516	1670

Al realizar la comparación de los valores de la tabla anterior para cada una de las transformadas se comprueba que el número de sumas y restas necesarias para evaluar la FHT es inferior a las necesarias para la PFT. Aun mas, si se toma en cuenta que para la implementación se utiliza un DSP para el cual el tiempo de ejecución de una suma o una multiplicación es el mismo el valor que hay que tomar en cuenta es el número total de operaciones. Al tomar el número total de operaciones se comprueba que la FHT realiza aproximadamente la mitad de operaciones que la PFT.

### 8.3 Software

El apéndice C contiene un listado de resultados de una FHT de 128 puntos ejecutado en el programa en TurboPascal 4. Se puede observar que la variación existente entre la FHT

de punto flotante y la FHT en entero es aproximadamente del 0.05 % en el peor de los casos. Hay que tomar en cuenta que la mantisa utilizada en la representación en punto flotante es equivalente a 11 dígitos y la representación en entero es de 5 dígitos. Si las muestras se encuentran normalizadas el error se introduce en la cuarta posición a la derecha del punto decimal.

### **8.3.1 Software de Simulación en TurboPascal 4**

Como se mencionó con anterioridad una de las partes centrales del proyecto fue la creación del algoritmo para la implementación de la FHT, y una de las principales características que se tiene que tomar en cuenta es la exactitud de cálculos en la realización de la transformada. Para poder tener una referencia en cuanto a la exactitud se creó un programa en TurboPascal 4, el cual contiene tres diferentes implementaciones de la transformada de Hartley. Y se le alimentó una serie de 128 puntos con un patrón repetitivo cada 16 muestras.

La primera sección realiza la transformada de Hartley en la forma tradicional, esto es, mediante sumatorias en todo el rango de muestras con variables de 48 bits de tamaño y del tipo punto flotante.

La segunda sección consistió en la elaboración de la transformada rápida de Hartley utilizando el mismo tipo de variables que en la transformada anterior.

Y la tercera sección tomo como base la transformada rápida de Hartley elaborada en punto flotante para crear un procedimiento para la FHT pero con variables 16 bits tipo entero, aun cuando sólo se utilizan 15 bits para la representación del resultado.

Todas las muestra se normalizaron de tal manera que existe un error introducido en el momento de normalizar y desnormalizar los valores de las muestras para poder resolver la FHT de 15 bits en entero.

Debido a que los resultados de la transformada y transformada inversa de Hartley con el tipo de variable entera proporciona una herramienta más útil para cotejar que los resultados que genera el TMS320C25 sean los correcta se obtuvieron los valores para cada mariposa, los cuáles se encuentran en el apéndice H. De igual manera se muestra la evolución de la transformada directa rápida de Hartley en la figura 25 y de la transformada inversa rápida de Hartley en la figura 26.

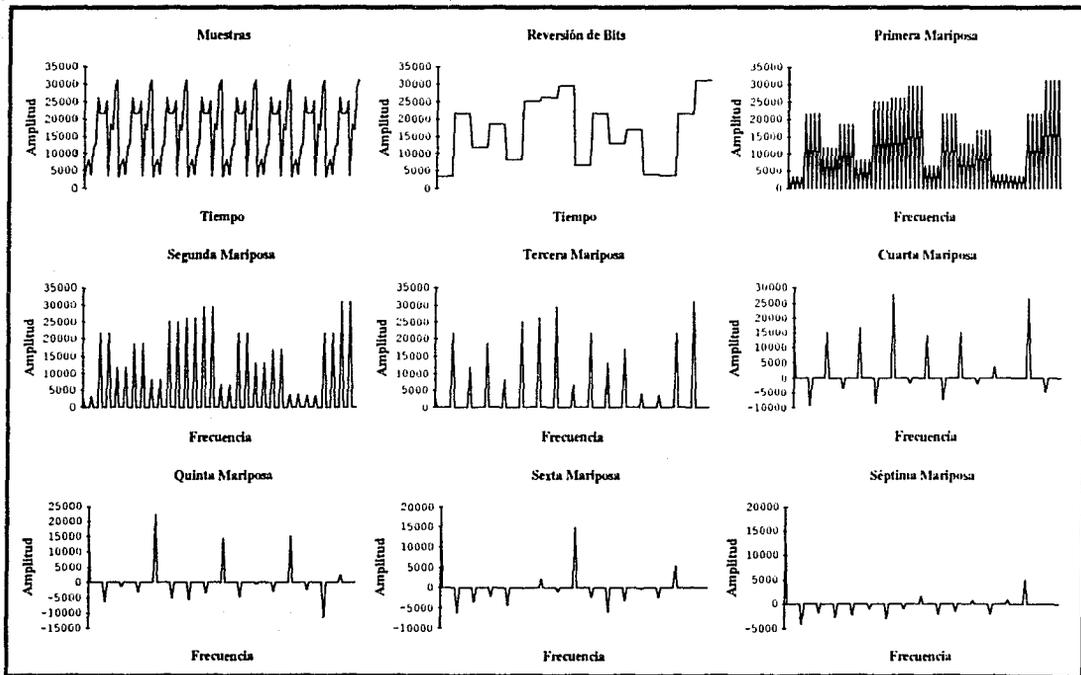


Figura 25

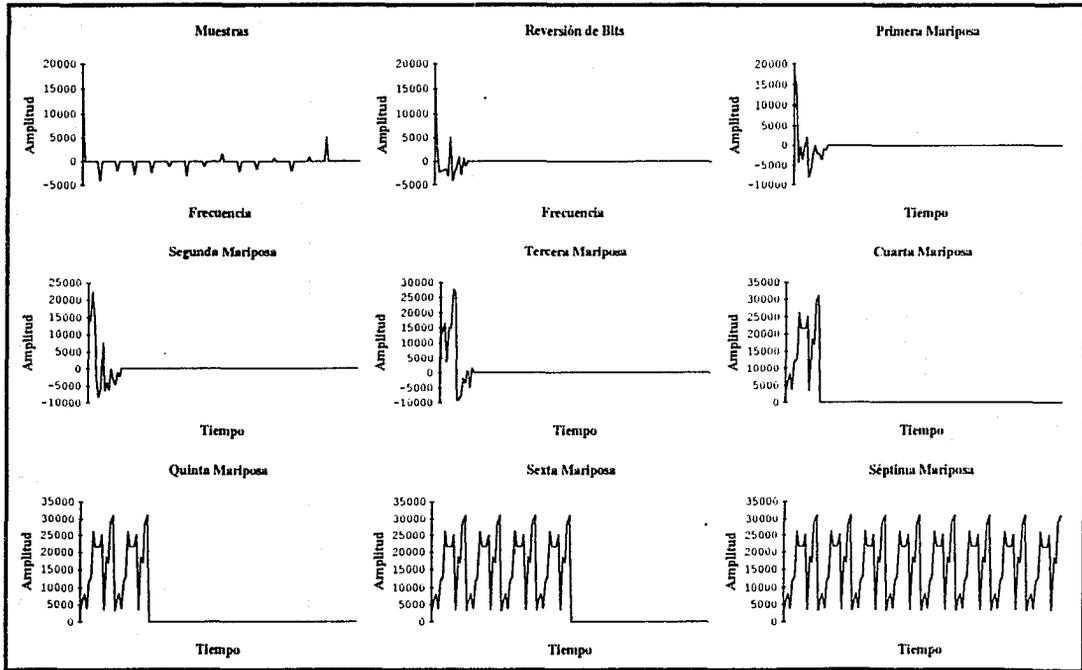


Figura 26

## **8.3.2 Software del TMS320 en el Simulador**

Después de haber elaborado un algoritmo funcional en un lenguaje de alto nivel se inicio la tarea de transportar los procedimientos de la FHT de tipo entero a el ensamblador del TMS320C25.

Asi mismo se generó un archivo de prueba con los mismos valores de muestras que fuera utilizado en el programa de TurboPascal 4, brindando la oportunidad de comprobar los resultados obtenidos con base de datos ya comprobada.

Para la ejecución del algoritmo elaborado en ensamblador TMS320 se utilizo el simulador TMS32020/TMS320C25 Simulator Release 2.0 87.260 de Texas Instruments Incorporated 1987 para transformar las muestras de prueba. Al término de cada mariposa se almacenó el contenido de la memoria de datos en un archivo de resultados, lo cual permite una forma sencilla de cotejar los datos. Los valores obtenidos de la simulación se encuentran en el apéndice I representando cada etapa del software.

## **8.4 Emulación por Hardware**

Una vez que el algoritmo de la FHT funcionó correctamente en el simulador se experimentó trabajarlo en el emulador Texas Instruments TMS320C25 Software Development System Version 1.00 el cual permite la ejecución de un programa en un DSP TMS320C25 sin la necesidad de tener que construir el sistema. La ejecución del programa se realiza a tiempo real y bajo condiciones de operación normales.

Para la evaluación del algoritmo se volvió a utilizar el archivo con las 128 muestras de prueba y al igual que en el simulador se generaron archivos después de cada mariposa, de tal manera que los resultados se pueden cotejar con los previamente obtenidos.

En la figura 27 y en la figura 28 se muestran en forma gráfica la evolución de la transformada directa rápida de Hartley y la transformada inversa rápida de Hartley respectivamente de acuerdo a los resultados proporcionados por el módulo de emulación. Los valores para la graficación se encuentran en el apéndice J, dichos valores representan cada etapa del algoritmo de tal manera que se manifiesta la cada paso de la transformación.

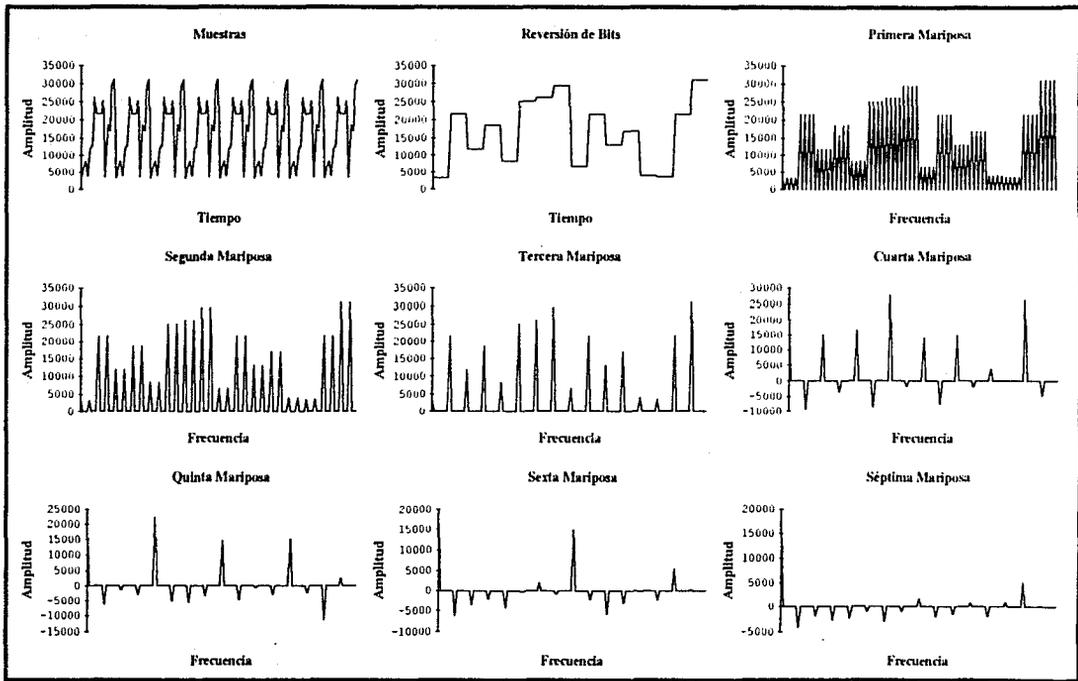


Figura 27

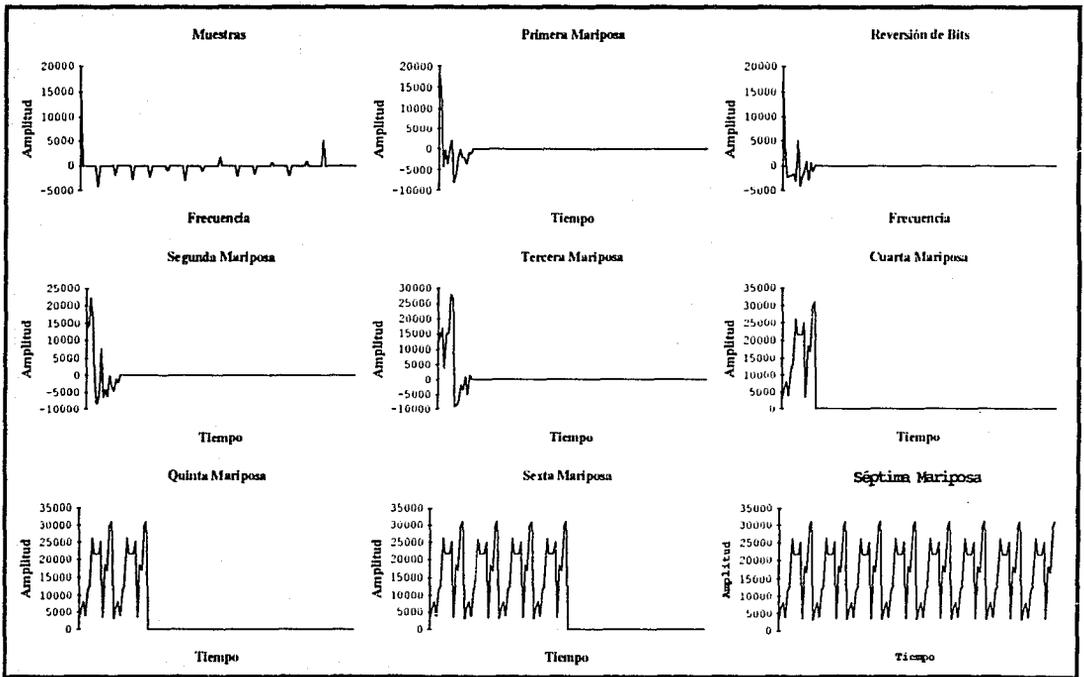
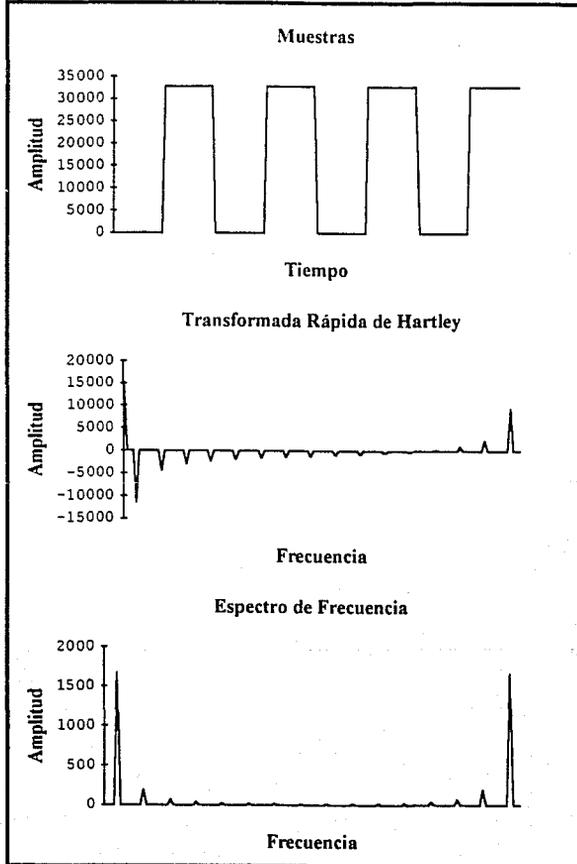


Figura 28

Por último se tiene la figura 29 la cual muestra una señal cuadrada a la cual se le aplica la FHT y posteriormente se obtiene su espectro de frecuencia. Se a eliminado de la gráfica la componente de directa que se encuentra en la muestra número cero, de tal manera que se pueda apreciar el espectro.

Figura 29



El Espectro de Frecuencia se obtiene aplicando el siguiente procedimiento en TurboPascal 4.0 :

```

PROCEDURE FHT_FFT;
VAR
IND:BYTE;
BEGIN
FOR IND:=1 TO 127 DO
SFHT[IND]:=(SQR(FHT[IND]) + SQR(FHT[128 - IND])) / 4;
SFHT[0]:=SQR(FHT[0]) / 2;
END;

```

## 8.5 Evaluación de la FHT.

De las principales ventajas que presenta la implementaron de la FHT con respecto a la Transformada de Fourier en cualquiera de sus formas esta la de una reducción del tiempo de ejecución en un 50 % cuando menos, el cual esta acompañado por una reducción equivalente en la cantidad de memoria necesaria.

En la tabla 7 se presentan el número de operaciones necesarias para la elaboración de la FHT mediante varios tipos de implementaron. La información presentada proviene de la bibliografía [2] y confirma los resultados obtenidos en cuanto al número de operaciones del algoritmo.

Tabla 7

Tamaño	Mult.	Sumas	Operaciones
Radix-2			
4	0	8	8
8	4	26	30
16	20	74	94
32	68	194	262
64	196	482	678
128	516	1154	1670
256	1284	2690	3974
512	3076	6146	9222
1024	7172	13826	20998
2048	16388	30722	47110
4096	36868	67586	104454

Tabla 7 (Continuación)

Tamaño	Mult.	Sumas	Operaciones
<b>Radix-4</b>			
4	0	8	8
16	14	70	84
64	142	450	594
256	942	2498	3440
1024	5294	12802	18096
4096	27310	62466	89776
<b>Split-Radix</b>			
4	0	8	8
8	2	22	24
16	12	64	76
32	42	166	208
64	124	416	540
128	330	998	1328
256	828	2336	3164
512	1994	5350	7344
1024	4668	12064	16732
2048	10698	26854	37552
4096	24124	59168	83292

El tiempo aproximado de transformación, tanto directo como inverso, para ambos canales es de 2.27 ms. El espacio en memoria interna de programa que se requiere es de 3400 palabras utilizando exclusivamente la memoria interna de datos.

### 8.5.1 Obtención de la FFT a partir de la FHT

La transformada discreta de Hartley está definida por la ecuación (32) para una secuencia  $x(n)$  de tamaño  $N$ , donde  $0 \leq n \leq N - 1$

$$\begin{aligned}
 H(k) &= \sum_{n=0}^{N-1} x(n) \operatorname{cas}(2\pi kn/N) & 0 \leq n \leq N - 1 \\
 x(n) &= \sum_{k=0}^{N-1} H(k) \operatorname{cas}(2\pi kn/N) & 0 \leq n \leq N - 1
 \end{aligned}
 \tag{32}$$

donde  $\text{cas}(x) = \cos(x) + \text{sen}(x)$ . La simetría de la transformada es un atributo de la DHT. Hay que hacer notar que la única diferencia que existe con la DFT es la ausencia de "-j" multiplicando al término seno, lo cual implica que la DHT es equivalente a restar la parte imaginaria de la parte real de la DFT de una secuencia de valores reales.

$$\text{DHT}[x(n)] = \text{Re} \{ \text{DFT}[x(n)] \} - \text{Im} \{ \text{DFT}[x(n)] \} \quad (33)$$

Como la parte real de la DFT de una secuencia de valores reales es par y la parte imaginaria es impar, la DFT puede ser calculada fácilmente de la DHT de la siguiente manera:

$$\begin{aligned} \text{Re} \{ \text{DFT}[x(n)] \} &= 1/2 \{ \text{DHT}[x(N-n)] + \text{DHT}[x(n)] \} \\ \text{Im} \{ \text{DFT}[x(n)] \} &= 1/2 \{ \text{DHT}[x(N-n)] - \text{DHT}[x(n)] \} \end{aligned} \quad (34)$$

donde  $x(N) = x(0)$ . Las propiedades de la DHT se pueden comprobar fácilmente mediante la utilización de las propiedades de la DFT.

La característica que tiene la FHT de transportarse y obtener la FFT utilizando el siguiente procedimiento en TurboPascal 4.0 :

```
PROCEDURE FHT_FFT;
VAR
  IND:BYTE;
BEGIN
  FOR IND:=1 TO 127 DO
  BEGIN
    FFTR[IND]:=(FHT[128 - IND] + FHT[IND]) / 2;
    FFTI[IND]:=(FHT[128 - IND] - FHT[IND]) / 2;
  END;
  FFTR[0]:=FHT[0];
  FFTI[0]:=0;
END;
```

# 9

## Conclusiones y Perspectivas

### 9.1 Conclusiones

De los resultados y la evaluación que se lograron del proyecto se aprecia que el manejo de audio digital de alta calidad es hoy en día una herramienta accesible para su aplicación en campos específicos en los que anteriormente se hubiera necesitado de un sistema grande y complejo. La utilización de circuitos especializados permite ver como un segundo término la antes compleja tarea de manipular las señales de audio a tiempo real para lograr información compacta y útil.

El proyecto fue diseñado tomando como punto de partida que el sistema fuera a ser tan solo un bloque básico para la adquisición de audio digital de alta calidad al cual, de acuerdo a las necesidades, se le fueran agregando módulos específicos o en su defecto como una herramienta de análisis del espectro de frecuencia.

Las posibles aplicaciones que puede tener el sistema cabe mencionar:

- 1) Ayudas para minusválidos, en la forma de tomar las señales de audio y convertirlas a señales que otros sistemas sensoriales sean capaces de reconocer y aprovechar.
- 2) Control activo de ruido mediante filtros digitales.
- 3) Productos de audio comerciales y profesionales.
- 4) Evaluación, pruebas y percepción de audio.
- 5) Procesamiento de audio para telecomunicaciones.
- 6) Música por computadora.
- 7) Sistemas Multimedia.

8) Sistemas de audio 3D (Surround Systems).

9) Transmisiones comerciales de audio digital, como para estaciones de radio o el sistema de audio para HDTV.

10) Control de acústica y eco.

11) Análisis y síntesis de sonido.

entre otras muchas aplicaciones las cuales parten de la señal de audio como fuente principal de información.

### **9.1.1 Problemas en la elaboración del Proyecto**

Como se mencionó con anterioridad la falta de los componentes necesarios según especificaciones del diseño ofrece obstáculos que se traducen en tiempo de elaboración.

La falta de componentes proviene de la particularidad de los mismos, de tal manera que para conseguirlos es necesario realizar pedidos especiales para su adquisición en el extranjero.

Otro problema muy particular que surgió debido a la elección del tipo de convertidor analógico-digital fue la de necesitar un cristal piezoeléctrico de una frecuencia especial para el circuito integrado. La frecuencia que se requiere es de 11289600 Hz, la cual no es comercial y genera la necesidad de fabricar dicho cristal. Por fortuna fue posible localizar una empresa en la Ciudad de México que se dedica a fabricar cristales bajo especificaciones particulares y en cantidades bajas (bibliografía [18]).

## **9.2 Perspectivas**

Las perspectivas del proyecto contemplan la terminación de un prototipo para realizar pruebas de calidad de la transformación de la señal bajo condiciones no ideales, funcionalidad del circuito como módulo base y características de señal a ruido entre otras. Posteriormente se pretende incursionar en la elaboración de módulos de expansión para las diferentes aplicaciones específicas que se mencionaron con anterioridad.

# 10

## Bibliografía

- [1] Mark A. O'Neill, "Faster Than Fast Fourier". Byte April 1988, pp. 293-300.
- [2] Henrik V. Sorensen, Douglas L. Jones, C. Sidney Burrus and Michael T. Heideman, "On Computing the Discrete Hartley Transform" IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-33, #4, pp. 1231-1238, October 1985.
- [3] Ken C. Pohlmann, "Principles of Digital Audio". 11711 North College, Carmel, Indiana 46032 USA. SAMS, 1989.
- [4] L.R. Rabiner and B. Gold, "Theory and Application of Digital Signal Processing". Englewood Cliffs, NJ: Prentice Hall, 1975.
- [5] A. V. Oppenheim, Ed., "Applications of Digital Signal Processing" Englewood Cliffs, NJ: Prentice Hall, 1978.
- [6] R. Morris, "Digital Signal Processing Software". Ottawa, Ont., Canada: DSPS Inc., 1983.
- [7] K. McDonough, E. Caudel, S. Magar, and A. Leigh, "Microcomputer with 32-bit arithmetic does high-precision number crunching", Electronics, pp. 105-110, Feb. 24, 1982.
- [8] S. Mager, E. Caudel, and A. Leigh, "A Microcomputer with digital signal processing capability", en 1982 Int. Solid State Conf. Dig. Tech. Pap., pp. 32-33, 284, 285.
- [9] "TMS320 First-Generation Digital Signal Processors Data Sheet". Houston, TX: Texas Instruments Inc., 1987.
- [10] "TMS32020 User's Guide". Houston, TX: Texas Instruments Inc., 1985.

- [11] "TMS320C25 User's Guide". Houston, TX: Texas Instruments Inc., 1986.
- [12] "TMS320C30 User's Guide". Houston, TX: Texas Instruments Inc., 1987.
- [13] "TMS320C2X User's Guide". Houston, TX: Texas Instruments Inc., 1990.
- [14] "Digital Signal Processing Applications with the TMS320 Family"  
Vol. 1 y 2. Houston, TX: Texas Instruments Inc., 1989.
- [15] "1991 Advanced Linear Circuits Seminar". Texas Instruments Inc., 1991.
- [16] C.S. Burrus and T.W. Parks, "DFT/FFT and Convolution Algorithms  
Theory and Implementation". Wiley Interscience, New York 1985.
- [17] Abraham Peled and Bede Liu, "Digital Signal Processing, Theory,  
Design, and Implementation". Robert E. Krieger Publishing  
Company, Malabar, Florida 1976.
- [18] CRICUSA  
Araucarias 11, Ampliación San Marcos Xochimilco  
Tel.: 653-2386 y 653-3231
- [19] Motorola Semiconductor Technical Data. DSP56ADC16/D.
- [20] Bracewall, Ronald N. "The Fast Hartley Transform". Proceedings of the IEEE  
vol. 72, no. 8, p. 1010, 1984.
- [21] Bracewall, Ronald N. "The Discrete Hartley Transform". Journal of the Optical  
Society of America, vol. 73, p. 1832, 1983.

# 11

## Glosario

- ADC** : Del lenguaje ingles Analog to Digital Converter que significa Convertidor Analógico-Digital.
- Aliasing** : Una confusión entre las muestras que se presenta durante el proceso de grabación de la señal.
- DAC** : Del lenguaje ingles Digital to Analog Converter que significa Convertidor Digital-Analógico.
- DFT** : Del lenguaje ingles Discrete Fourier Transform que significa Transformada Discreta de Fourier.
- DHT** : Del lenguaje ingles Discrete Hartley Transform que significa Transformada Discreta de Hartley.
- DIF** : Del lenguaje ingles Decimation In Frequency que significa Decimación en Frecuencia.
- DIT** : Del lenguaje ingles Decimation In Time que significa Decimación en el Tiempo.
- Dither** : Se denomina a la señal que realiza la función de enmascarar el efecto de error de cuantización.
- DSP** : Del lenguaje ingles Digital Signal Processor que significa Procesador Digital de Señales.
- FFT** : Del lenguaje ingles Fast Fourier Transform que significa Transformada Rápida de Fourier.
- FHT** : Del lenguaje ingles Fast Hartley Transform que significa Transformada Rápida de Hartley.

IIR	: Del lenguaje ingles Infinite Impulse Response que significa Respuesta Impulso Infinito e implica un tipo de de filtro activo.
MIP	: Del lenguaje ingles Million Instructions per Second que significa Millón de Instrucciones por Segundo.
MFLOPS	: Del lenguaje ingles Million Floting-Point Operations per Second que significa Millón de Operaciones de Punto Flotante por Segundo.
PFT	: Del lenguaje ingles Prime-Factor FFT que significa FFT de factor primo.
Pipeline	: Una cola formada por registros de almacenamiento que sirve para agilizar la búsqueda de información por parte del procesador.
Radix	: La unidad mínima en que se encuentra dividida una transformada.
RAM	: Del lenguaje ingles Random Access Memory que significa Memoria de Acceso Aleatorio.
ROM	: Del lenguaje ingles Read Only Memory que significa Memoria de Lectura Unicamente.
Stand-Alone	: Implica una unidad o sistema que no requiere de ayuda o dispositivos adicionales básicos para trabajar.
Twiddle	: Del lenguaje ingles, que significa factor de giro y seFactor refiere a un factor numérico para poder resolver una transformada.

# Anexo **i**

## **i FHT en Formato de Punto Flotante**

Debido a que el TMS320C25 es un DSP que sólo soporta operaciones del tipo entero cuando sea necesario realizar el algoritmo con operaciones de punto flotante existen dos alternativas: la primera consiste en cambiar el DSP TMS320C25 por el TMS320C30, dicho procesador puede realizar operaciones de punto flotante inherentes en su conjunto de instrucciones, lo cual sólo requiere de una transformación de las instrucciones utilizadas, esto no es un problema debido a que el TMS320C30 tiene instrucciones equivalentes al TMS320C25 pero para punto flotante; la segunda opción consiste en la de crear rutinas en el TMS320C25 que realicen las operaciones de punto flotante necesarias.

La segunda opción es económica debido a la diferencia de precios que existe entre el TMS320C25 y el TMS320C30 (de siete a diez veces más caro). Sin embargo, esta solución restringe el ancho de banda de la señal a transformar, así como también el número de canales que se pueden manejar.

La principal causa por la que se reduce el ancho de banda y el número de canales se debe a que una instrucción en punto flotante requiere de varias instrucciones en formato entero para poder realizar la misma operación. Mediante la presentación de la siguiente rutina se puede apreciar la diferencia existente.

La siguiente rutina realiza la multiplicación de dos números en formato IEEE del tipo punto flotante de precisión sencilla (1 bit de signo, 8 bits de exponente y 23 bits de mantisa).

LT	LSB	; Carga la parte baja del número
MPY	VCOSL	; Multiplicación parte baja, primera pasada
SPH	Temp1	
LAC	MSB	
ANDK	007FH	
SALC	Dato	
LT	Dato	
MPY	VCOSL	; Multiplicación parte baja, segunda pasada

	LTP	LSB	
	MPY	VCOSM	; Multiplicación parte alta, primera pasada
	LTA	Dato	
	MPY	VCOSM	; Multiplicación parte alta, segunda pasada
	SPL	Temp2	
	ADDS	Temp1	; Suma de resultados
	ADDH	Temp2	; Suma de resultados
	BIT	00H	; Normalizar
	BBNZ	S1	
	SFL		
	BIT	00H	; Normalizar
	BBNZ	S2	
	RPTK	07H	
	SRL		
	SACL	LSB	
	SACH	Dato	
	LAC	MSB,9	
	ADDH	VCOSE	; Suma del exponente
	ADLK	2,16	
	ANDK	01FFH,16	
	ADD	Dato,9	
	XORK	8000H,9	; Signo
	RPTK	08H	
	SRL		
	SACL	MSB	
	B	Terminar	
S2:	RPTK	08H	
	SRL		
	SACL	LSB	
	SACH	Dato	
	LAC	MSB,9	
	ADDH	VCOSE	
	ADLK	1,16	
	ANDK	01FFH,16	
	ADD	Dato,9	
	XORK	8000H,9	
	RPTK	08H	
	SRL		
	SACL	MSB	
	B	Terminar	
S1:	RPTK	09H	
	SRL		
	SACL	LSB	

SACH	Dato
LAC	MSB,9
ADDH	VCOSE
ANDK	01FFH,16
ADD	Dato,9
XORK	8000H,9
RPTK	08H
SRL	
SACL	MSB

Terminar:

La rutina anterior además de ocupar un espacio grande en memoria de programa, necesita de mucho tiempo de ejecución para realizar la multiplicación.

En el mejor de los casos la rutina se tomaría de 47 a 55 ciclos (promedio de 51 ciclos) para ejecutarse. Para una suma en punto flotante mediante operaciones tipo entero sería necesario entre 10 y 30 ciclos (promedio de 20 ciclos) para que se pueda completar la operación.

Si se toma en cuenta el número de ciclos que necesita el algoritmo por cada operación y se sustituye por el equivalente necesario en la simulación de operaciones de punto flotante, se puede estimar el ancho de banda y el número de canales disponibles.

El algoritmo se realiza aproximadamente en 22300 ciclos de los cuales 1670 son de operaciones aritméticas. A cada operación aritmética se le tiene que asociar con instrucciones de carga del dato y almacenamiento del resultado, por lo que es válido concluir que cada operación se tomara de 3 a 5 ciclos en completarse. Lo anterior representa un total de 5010 ciclos en el mejor de los casos y de 8350 en el peor de los casos, tomando el promedio del número de ciclos requeridos (6680 ciclos) y sustituyéndolo por el número equivalente de ciclos para las operaciones de punto flotante

$$\begin{aligned} \text{Número de ciclos} &= \text{Número de sumas} * \text{Ciclos de suma} \\ &+ \text{Número de multiplicaciones} * \text{Ciclos de multiplicación} \end{aligned}$$

$$49396 = 20 * 1154 + 51 * 516$$

También es necesario incluir el tiempo de conversión que requiere el traspasar el valor desde un formato entero a un formato de punto flotante y viceversa, lo cual implica de 4 a 34 ciclos y de 30 a 40 ciclos respectivamente. Si tomamos el promedio de ciclos requeridos y lo sustituimos por el número de veces que se tiene que realizar el intercambio de datos:

$$\begin{aligned} \text{Número de ciclos} &= (\text{Número de muestras} * (\text{Ciclos de entrada} \\ &+ \text{Ciclos de salida})) * \text{Número de canales} \end{aligned}$$

$$13824 = (128 * (19 + 35)) * 2$$

De todas las consideraciones que se mencionaron el número total de ciclos que se requiere para realizar la FHT de dos canales de 128 muestras con operaciones de punto flotante es de :

$$\begin{aligned} \text{Número de ciclos} &= \text{Número de ciclos aritméticos} \\ &+ \text{Número de ciclos de transferencia y conversión} \\ &+ \text{Número de ciclos de instrucciones misceláneas} \end{aligned}$$

$$78240 = 49396 + 13824 + 14820$$

Debido al número de ciclos que se requieren no sería posible realizar la FHT de dos canales con un ancho de banda de 44.1 KHz. En el mejor de los casos se podría aplicar la FHT a un canal con un ancho de banda de 16 KHz.



X6 .set 524  
X7 .set 526  
X8 .set 528  
X9 .set 530  
X10 .set 532  
X11 .set 534  
X12 .set 536  
X13 .set 538  
X14 .set 540  
X15 .set 542  
X16 .set 544  
X17 .set 546  
X18 .set 548  
X19 .set 550  
X20 .set 552  
X21 .set 554  
X22 .set 556  
X23 .set 558  
X24 .set 560  
X25 .set 562  
X26 .set 564  
X27 .set 566  
X28 .set 568  
X29 .set 570  
X30 .set 572  
X31 .set 574  
X32 .set 576  
X33 .set 578  
X34 .set 580  
X35 .set 582  
X36 .set 584  
X37 .set 586  
X38 .set 588  
X39 .set 590  
X40 .set 592  
X41 .set 594  
X42 .set 596  
X43 .set 598  
X44 .set 600  
X45 .set 602  
X46 .set 604  
X47 .set 606  
X48 .set 608  
X49 .set 610  
X50 .set 612  
X51 .set 614  
X52 .set 616  
X53 .set 618

X54 .set 620  
X55 .set 622  
X56 .set 624  
X57 .set 626  
X58 .set 628  
X59 .set 630  
X60 .set 632  
X61 .set 634  
X62 .set 636  
X63 .set 638  
X64 .set 640  
X65 .set 642  
X66 .set 644  
X67 .set 646  
X68 .set 648  
X69 .set 650  
X70 .set 652  
X71 .set 654  
X72 .set 656  
X73 .set 658  
X74 .set 660  
X75 .set 662  
X76 .set 664  
X77 .set 666  
X78 .set 668  
X79 .set 670  
X80 .set 672  
X81 .set 674  
X82 .set 676  
X83 .set 678  
X84 .set 680  
X85 .set 682  
X86 .set 684  
X87 .set 686  
X88 .set 688  
X89 .set 690  
X90 .set 692  
X91 .set 694  
X92 .set 696  
X93 .set 698  
X94 .set 700  
X95 .set 702  
X96 .set 704  
X97 .set 706  
X98 .set 708  
X99 .set 710  
X100 .set 712  
X101 .set 714

X102	.set	716
X103	.set	718
X104	.set	720
X105	.set	722
X106	.set	724
X107	.set	726
X108	.set	728
X109	.set	730
X110	.set	732
X111	.set	734
X112	.set	736
X113	.set	738
X114	.set	740
X115	.set	742
X116	.set	744
X117	.set	746
X118	.set	748
X119	.set	750
X120	.set	752
X121	.set	754
X122	.set	756
X123	.set	758
X124	.set	760
X125	.set	762
X126	.set	764
X127	.set	766
X128	.set	768
X129	.set	770
X130	.set	772
X131	.set	774
X132	.set	776
X133	.set	778
X134	.set	780
X135	.set	782
X136	.set	784
X137	.set	786
X138	.set	788
X139	.set	790
X140	.set	792
X141	.set	794
X142	.set	796
X143	.set	798
X144	.set	800
X145	.set	802
X146	.set	804
X147	.set	806
X148	.set	808
X149	.set	810

X150	.set	812
X151	.set	814
X152	.set	816
X153	.set	818
X154	.set	820
X155	.set	822
X156	.set	824
X157	.set	826
X158	.set	828
X159	.set	830
X160	.set	832
X161	.set	834
X162	.set	836
X163	.set	838
X164	.set	840
X165	.set	842
X166	.set	844
X167	.set	846
X168	.set	848
X169	.set	850
X170	.set	852
X171	.set	854
X172	.set	856
X173	.set	858
X174	.set	860
X175	.set	862
X176	.set	864
X177	.set	866
X178	.set	868
X179	.set	870
X180	.set	872
X181	.set	874
X182	.set	876
X183	.set	878
X184	.set	880
X185	.set	882
X186	.set	884
X187	.set	886
X188	.set	888
X189	.set	890
X190	.set	892
X191	.set	894
X192	.set	896
X193	.set	898
X194	.set	900
X195	.set	902
X196	.set	904
X197	.set	906

X198	.set	908
X199	.set	910
X200	.set	912
X201	.set	914
X202	.set	916
X203	.set	918
X204	.set	920
X205	.set	922
X206	.set	924
X207	.set	926
X208	.set	928
X209	.set	930
X210	.set	932
X211	.set	934
X212	.set	936
X213	.set	938
X214	.set	940
X215	.set	942
X216	.set	944
X217	.set	946
X218	.set	948
X219	.set	950
X220	.set	952
X221	.set	954
X222	.set	956
X223	.set	958
X224	.set	960
X225	.set	962
X226	.set	964
X227	.set	966
X228	.set	968
X229	.set	970
X230	.set	972
X231	.set	974
X232	.set	976
X233	.set	978
X234	.set	980
X235	.set	982
X236	.set	984
X237	.set	986
X238	.set	988
X239	.set	990
X240	.set	992
X241	.set	994
X242	.set	996
X243	.set	998
X244	.set	1000
X245	.set	1002

X246	.set	1004
X247	.set	1006
X248	.set	1008
X249	.set	1010
X250	.set	1012
X251	.set	1014
X252	.set	1016
X253	.set	1018
X254	.set	1020
X255	.set	1022

:	Tabla de Cosenos	
:		
C0	.set	4095
C1	.set	4094
C2	.set	4090
C3	.set	4084
C4	.set	4075
C5	.set	4064
C6	.set	4051
C7	.set	4035
C8	.set	4016
C9	.set	3996
C10	.set	3972
C11	.set	3947
C12	.set	3919
C13	.set	3888
C14	.set	3856
C15	.set	3821
C16	.set	3783
C17	.set	3744
C18	.set	3702
C19	.set	3658
C20	.set	3611
C21	.set	3563
C22	.set	3512
C23	.set	3460
C24	.set	3405
C25	.set	3348
C26	.set	3289
C27	.set	3228
C28	.set	3165
C29	.set	3101
C30	.set	3034
C31	.set	2966
C32	.set	2896
C33	.set	2824
C34	.set	2750

C35 .set 2675  
C36 .set 2598  
C37 .set 2519  
C38 .set 2439  
C39 .set 2358  
C40 .set 2275  
C41 .set 2191  
C42 .set 2105  
C43 .set 2018  
C44 .set 1930  
C45 .set 1841  
C46 .set 1751  
C47 .set 1659  
C48 .set 1567  
C49 .set 1474  
C50 .set 1380  
C51 .set 1285  
C52 .set 1189  
C53 .set 1092  
C54 .set 995  
C55 .set 897  
C56 .set 799  
C57 .set 700  
C58 .set 601  
C59 .set 501  
C60 .set 401  
C61 .set 301  
C62 .set 201  
C63 .set 100  
C64 .set 0  
C65 .set -99  
C66 .set -200  
C67 .set -309  
C68 .set -400  
C69 .set -500  
C70 .set -600  
C71 .set -699  
C72 .set -798  
C73 .set -896  
C74 .set -994  
C75 .set -1091  
C76 .set -1188  
C77 .set -1284  
C78 .set -1379  
C79 .set -1473  
C80 .set -1566  
C81 .set -1658  
C82 .set -1750

C83 .set -1840  
C84 .set -1929  
C85 .set -2017  
C86 .set -2104  
C87 .set -2190  
C88 .set -2274  
C89 .set -2357  
C90 .set -2438  
C91 .set -2518  
C92 .set -2597  
C93 .set -2674  
C94 .set -2740  
C95 .set -2823  
C96 .set -2895  
C97 .set -2965  
C98 .set -3033  
C99 .set -3100  
C100 .set -3164  
C101 .set -3227  
C102 .set -3288  
C103 .set -3347  
C104 .set -3404  
C105 .set -3459  
C106 .set -3511  
C107 .set -3562  
C108 .set -3610  
C109 .set -3657  
C110 .set -3701  
C111 .set -3743  
C112 .set -3782  
C113 .set -3820  
C114 .set -3855  
C115 .set -3887  
C116 .set -3918  
C117 .set -3946  
C118 .set -3971  
C119 .set -3995  
C120 .set -4015  
C121 .set -4034  
C122 .set -4050  
C123 .set -4063  
C124 .set -4074  
C125 .set -4083  
C126 .set -4089  
C127 .set -4093  
C128 .set -4094  
C129 .set -4093  
C130 .set -4089

C131	.set	-4083
C132	.set	-4074
C133	.set	-4063
C134	.set	-4050
C135	.set	-4034
C136	.set	-4015
C137	.set	-3995
C138	.set	-3971
C139	.set	-3946
C140	.set	-3918
C141	.set	-3887
C142	.set	-3855
C143	.set	-3820
C144	.set	-3782
C145	.set	-3743
C146	.set	-3701
C147	.set	-3657
C148	.set	-3610
C149	.set	-3562
C150	.set	-3511
C151	.set	-3459
C152	.set	-3404
C153	.set	-3347
C154	.set	-3288
C155	.set	-3227
C156	.set	-3164
C157	.set	-3100
C158	.set	-3033
C159	.set	-2965
C160	.set	-2895
C161	.set	-2833
C162	.set	-2749
C163	.set	-2674
C164	.set	-2597
C165	.set	-2518
C166	.set	-2438
C167	.set	-2357
C168	.set	-2274
C169	.set	-2190
C170	.set	-2104
C171	.set	-2017
C172	.set	-1929
C173	.set	-1840
C174	.set	-1750
C175	.set	-1658
C176	.set	-1566
C177	.set	-1473
C178	.set	-1379

C179	.set	-1284
C180	.set	-1188
C181	.set	-1091
C182	.set	-994
C183	.set	-896
C184	.set	-798
C185	.set	-699
C186	.set	-600
C187	.set	-500
C188	.set	-400
C189	.set	-300
C190	.set	-200
C191	.set	-99
C192	.set	0
C193	.set	100
C194	.set	201
C195	.set	301
C196	.set	401
C197	.set	501
C198	.set	601
C199	.set	700
C200	.set	799
C201	.set	897
C202	.set	995
C203	.set	1092
C204	.set	1189
C205	.set	1285
C206	.set	1380
C207	.set	1474
C208	.set	1567
C209	.set	1659
C210	.set	1751
C211	.set	1841
C212	.set	1930
C213	.set	2018
C214	.set	2105
C215	.set	2191
C216	.set	2275
C217	.set	2358
C218	.set	2439
C219	.set	2519
C220	.set	2598
C221	.set	2675
C222	.set	2750
C223	.set	2824
C224	.set	2896
C225	.set	2966
C226	.set	3034

C227	.set	3101
C228	.set	3165
C229	.set	3228
C230	.set	3289
C231	.set	3348
C232	.set	3405
C233	.set	3460
C234	.set	3512
C235	.set	3563
C236	.set	3611
C237	.set	3658
C238	.set	3702
C239	.set	3744
C240	.set	3783
C241	.set	3821
C242	.set	3856
C243	.set	3888
C244	.set	3919
C245	.set	3947
C246	.set	3972
C247	.set	3996
C248	.set	4016
C249	.set	4035
C250	.set	4051
C251	.set	4064
C252	.set	4075
C253	.set	4084
C254	.set	4090
C255	.set	4094
:		
:	Tabla de Senos	
:		
S0	.set	512
S1	.set	514
S2	.set	516
S3	.set	518
S4	.set	520
S5	.set	522
S6	.set	524
S7	.set	526
S8	.set	528
S9	.set	530
S10	.set	532
S11	.set	534
S12	.set	536
S13	.set	538
S14	.set	540
S15	.set	542

S16	.set	544
S17	.set	546
S18	.set	548
S19	.set	550
S20	.set	552
S21	.set	554
S22	.set	556
S23	.set	558
S24	.set	560
S25	.set	562
S26	.set	564
S27	.set	566
S28	.set	568
S29	.set	570
S30	.set	572
S31	.set	574
S32	.set	576
S33	.set	578
S34	.set	580
S35	.set	582
S36	.set	584
S37	.set	586
S38	.set	588
S39	.set	590
S40	.set	592
S41	.set	594
S42	.set	596
S43	.set	598
S44	.set	600
S45	.set	602
S46	.set	604
S47	.set	606
S48	.set	608
S49	.set	610
S50	.set	612
S51	.set	614
S52	.set	616
S53	.set	618
S54	.set	620
S55	.set	622
S56	.set	624
S57	.set	626
S58	.set	628
S59	.set	630
S60	.set	632
S61	.set	634
S62	.set	636
S63	.set	638

S64 .set 640  
S65 .set 642  
S66 .set 644  
S67 .set 646  
S68 .set 648  
S69 .set 650  
S70 .set 652  
S71 .set 654  
S72 .set 656  
S73 .set 658  
S74 .set 660  
S75 .set 662  
S76 .set 664  
S77 .set 666  
S78 .set 668  
S79 .set 670  
S80 .set 672  
S81 .set 674  
S82 .set 676  
S83 .set 678  
S84 .set 680  
S85 .set 682  
S86 .set 684  
S87 .set 686  
S88 .set 688  
S89 .set 690  
S90 .set 692  
S91 .set 694  
S92 .set 696  
S93 .set 698  
S94 .set 700  
S95 .set 702  
S96 .set 704  
S97 .set 706  
S98 .set 708  
S99 .set 710  
S100 .set 712  
S101 .set 714  
S102 .set 716  
S103 .set 718  
S104 .set 720  
S105 .set 722  
S106 .set 724  
S107 .set 726  
S108 .set 728  
S109 .set 730  
S110 .set 732  
S111 .set 734

S112 .set 736  
S113 .set 738  
S114 .set 740  
S115 .set 742  
S116 .set 744  
S117 .set 746  
S118 .set 748  
S119 .set 750  
S120 .set 752  
S121 .set 754  
S122 .set 756  
S123 .set 758  
S124 .set 760  
S125 .set 762  
S126 .set 764  
S127 .set 766  
S128 .set 768  
S129 .set 770  
S130 .set 772  
S131 .set 774  
S132 .set 776  
S133 .set 778  
S134 .set 780  
S135 .set 782  
S136 .set 784  
S137 .set 786  
S138 .set 788  
S139 .set 790  
S140 .set 792  
S141 .set 794  
S142 .set 796  
S143 .set 798  
S144 .set 800  
S145 .set 802  
S146 .set 804  
S147 .set 806  
S148 .set 808  
S149 .set 810  
S150 .set 812  
S151 .set 814  
S152 .set 816  
S153 .set 818  
S154 .set 820  
S155 .set 822  
S156 .set 824  
S157 .set 826  
S158 .set 828  
S159 .set 830

S160 .set 832  
S161 .set 834  
S162 .set 836  
S163 .set 838  
S164 .set 840  
S165 .set 842  
S166 .set 844  
S167 .set 846  
S168 .set 848  
S169 .set 850  
S170 .set 852  
S171 .set 854  
S172 .set 856  
S173 .set 858  
S174 .set 860  
S175 .set 862  
S176 .set 864  
S177 .set 866  
S178 .set 868  
S179 .set 870  
S180 .set 872  
S181 .set 874  
S182 .set 876  
S183 .set 878  
S184 .set 880  
S185 .set 882  
S186 .set 884  
S187 .set 886  
S188 .set 888  
S189 .set 890  
S190 .set 892  
S191 .set 894  
S192 .set 896  
S193 .set 898  
S194 .set 900  
S195 .set 902  
S196 .set 904  
S197 .set 906  
S198 .set 908  
S199 .set 910  
S200 .set 912  
S201 .set 914  
S202 .set 916  
S203 .set 918  
S204 .set 920  
S205 .set 922  
S206 .set 924  
S207 .set 926

S208 .set 928  
S209 .set 930  
S210 .set 932  
S211 .set 934  
S212 .set 936  
S213 .set 938  
S214 .set 940  
S215 .set 942  
S216 .set 944  
S217 .set 946  
S218 .set 948  
S219 .set 950  
S220 .set 952  
S221 .set 954  
S222 .set 956  
S223 .set 958  
S224 .set 960  
S225 .set 962  
S226 .set 964  
S227 .set 966  
S228 .set 968  
S229 .set 970  
S230 .set 972  
S231 .set 974  
S232 .set 976  
S233 .set 978  
S234 .set 980  
S235 .set 982  
S236 .set 984  
S237 .set 986  
S238 .set 988  
S239 .set 990  
S240 .set 992  
S241 .set 994  
S242 .set 996  
S243 .set 998  
S244 .set 1000  
S245 .set 1002  
S246 .set 1004  
S247 .set 1006  
S248 .set 1008  
S249 .set 1010  
S250 .set 1012  
S251 .set 1014  
S252 .set 1016  
S253 .set 1018  
S254 .set 1020  
S255 .set 1022

Macros para producir una FFT compleja de 256-puntos en línea

ZERO para el caso en que  $W = 1$  (Theta=0)

Las X's y las Y's son localidades de entrada y salida para la mariposa.  
 Entrada con: ARP = 1, AR1 = X11, AR2 = X1, AR0 = [X1] - [X11]  
 Salida con: ARP = 1, AR1 = SIGUIENTE X11, AR2 = SIGUIENTE X1

```
ZERO      SMACRO
LAC      *0+,15      (0/2)X11
ADD      *0-,15      (0/2)X13
SACH     *0+        X11 - (0/2)(X11 + X13)
SUBHH    *          X13
SACH     *0-,0,AR2  X13 - (0/2)(X11 - X13)
LAC      *0+,14      (0/4)X1
SUB      *0+,14      (0/4)X12
SACH     T1         T1 = (0/4)(X1 - X12)
ADD      *15,AR1    R1(ACC) = (0/4)(X1 + X12)
SUB      *15,AR2    (0/2)X11
SACH     *0-,0,AR1  X12 = R1 - (0/2)X11
ADDDH    *+,AR2    X11
SACH     *+,0,AR1   X1 = R1 + (0/2)X11

LAC      *0+,15      (0/2)Y11
ADD      *0-,15      (0/2)Y13
SACH     *0+        Y11 - (0/2)(Y11 + Y13)
SUBHH    *          Y13
SACH     *0-,0,AR. Y13 - (0/2)(Y11 - Y13)
LAC      *0+,14      (0/4)Y1
SUB      *0+,14      (0/4)Y12
SACH     T2         T2 = (0/4)(Y1 - Y12)
ADD      *15,AR1    S1(ACC) = (0/4)(Y1 + Y12)
SUB      *15,AR2    (0/2)Y11
SACH     *0-,0,AR1 Y12 = S1 - (0/2)Y11
ADDDH    *+,AR2    Y11
SACH     *+,0,AR1   Y1 = S1 + (0/2)Y11

ZALH     T1         T1
ADD      *0-,15      (0/2)Y13
MAR      *+        X11 = T1 + (0/2)Y13
SACH     *0+        X11 = T1 + (0/2)Y13
MAR      *0+        Y13
SUBHH    *+        Y13
SACH     T1         T1 = T1 - (0/2)Y13
```

```
ZALH     T2         T2
ADD      *+,15      (0/2)Y13
SACH     *+        Y13 = T2 + (0/2)X13
SUBHH    *          X13
LAR      AR3,T1     AR3 = T1
SAR      AR3,+     X13 = T1
MAR      *0-
SACH     *+        Y11 = T2 - (0/2)X13

SEND
```

NORMAL Mariposa estándar Radix-4

Las X's y las Y's son localidades de entrada y salida para la mariposa. IA's son las localidades del factor de giro.  
 Entrada con: ARP = 1, AR1 = X11, AR2 = X1, AR0 = [X1] - [X11]  
 Salida con: ARP = 1, AR1 = SIGUIENTE X11, AR2 = SIGUIENTE X1

```
NORMAL    SMACRO      IA1,IA2,IA3
LAC      *0+,15      (0/2)X11
ADD      *0-,15      (0/2)X13
SACH     *0+        X11 - (0/2)(X11 + X13)
SUBHH    *          X13
SACH     *0-,0,AR2  X13 - (0/2)(X11 - X13)
LAC      *0+,14      (0/4)X1
SUB      *0+,14      (0/4)X12
SACH     T1         T1 = (0/4)(X1 - X12)
ADD      *15,AR1    R1(ACC) = (0/4)(X1 + X12)
SUB      *15,AR2    (0/2)X11
SACH     *0-,0,AR1  X12 = R1 - (0/2)X11
ADDDH    *+,AR2    X11
SACH     *+,0,AR1   X1 = R1 + (0/2)X11

LAC      *0+,15      (0/2)Y11
ADD      *0-,15      (0/2)Y13
SACH     *0+        Y11 - (0/2)(Y11 + Y13)
SUBHH    *          Y13
SACH     *0-,0,AR1 Y13 - (0/2)(Y11 - Y13)
ADDDH    *+,AR2    Y11
SACH     *+,0,AR1   Y1 = R1 + (0/2)Y11

LAC      *0+,15      (0/2)Y11
ADD      *0-,15      (0/2)Y13
SACH     *0+        Y11 - (0/2)(Y11 + Y13)
SUBHH    *          Y13
SACH     *0-,0,AR2 Y13 - (0/2)(Y11 - Y13)
LAC      *0+,14      (0/4)Y1
SUB      *0+,14      (0/4)Y12
SACH     T2         T2 = (0/4)(Y1 - Y12)
ADD      *15,AR1    S1(ACC) = (0/4)(Y1 + Y12)
SUB      *15,AR2    (0/2)Y11
SACH     *0-,0,AR1 Y12 = S1 - (0/2)Y11
ADDDH    *+,AR2    Y11
SACH     *+,0,AR1   Y1 = S1 + (0/2)Y11
```

ZALH	T1	
ADD	*0,-15	(I/2) YB
MAR	*	
SACH	*+	XH - T1 + (I/2) YB
MAR	*0+	
SUBH	*	YB
SACH	T1	X1 - T1 - (I/2) YB
ZALH	T2	
ADD	*+,15	(I/2) X3
SACH	T2	T2 - T2 + (I/2) X3
SUBH	*+	X3
MAR	*0-	
SACH	*	YH - T2 - (I/2) X3
LT	*	YH
MPYK	C:IA1:	,CO1 * YH
LTP	*+	XH
MPYK	S:IA1:	S11 * XH
LTS	*	YH
SACH	*-,4	YH + YH * CO1 - XH * S11
MPYK	S:IA1:	S11 * YH
LTP	*AR2	XH
MPYK	C:IA1:	,CO1 * XH
LTA	*-,AR1	YH
SACH	*0+,4,AR2	XH - YH * S11 + XH * CO1
MPYK	C:IA2:	,CO2 * YH
LTP	*+	XH
MPYK	S:IA2:	S12 * XH
LTS	*	YH
SACH	*-,4	YH + YH * CO2 - XH * S12
MPYK	S:IA2:	S12 * YH
LTP	*X12	
MPYK	C:IA2:	,CO2 * XH
LTA	T1,T1	
SACH	*0,-,4	XH - YH * S12 + XH * CO2
MAR	*+,AR1	
MPYK	C:IA3:	,CO3 * T1
LTP	T2	T2
MPYK	S:IA3:	S13 * T2
APAC		
SACH	*-,4,AR2	XH - T1 * CO3 + T2 * S13
MPYK	C:IA3:	,CO3 * T2
LTP	T1	
MPYK	S:IA3:	S13 * T1
LTS	*+,AR1	
SACH	*+,4	YH - T2 * CO3 - T1 * S13

MAR	*0-		
SEND			
SPCIAL	para el caso en que Theta = pi/4		
	Las X's y las Y's son localidades de entrada y salida para la mariposa.		
	Entrada con: ARP = 1, AR1 = XH, AR2 = XH, AR0 = [X3] - [X11]		
	Salida con: ARP = 1, AR1 = SIGUIENTE XH, AR2 = SIGUIENTE XH		
SPCIAL	SMACRO		
LAC	*0+,14	(I/4) XH	
ADD	*0,-14	(I/4) X3	
SACH	*0+	XH - (I/4)(XH + XH)	
SUB	*+,15	(I/2) X3	
SACH	*+	X3 - (I/4)(XH - XH)	
LAC	*0,-14	(I/4) YB	
ADD	*+,14	(I/4) YH	
SACH	*0+	YH - (I/4)(YH + YH)	
SUB	*+,15	(I/2) YB	
SACH	*0,-0,AR2	YB - (I/4)(YH - YH)	
LAC	*0+,14	(I/4) X1	
ADD	*0,-14	(I/4) X2	
SACH	*0+	X1 - (I/4)(X1 + X2)	
SUB	*+,15	(I/2) X2	
SACH	*+	X2 - (I/4)(X1 - X2)	
LAC	*0,-14	(I/4) Y1	
ADD	*+,14	(I/4) Y2	
SACH	*0+	Y1 - (I/4)(Y1 + Y2)	
SUB	*+,15	(I/2) Y2	
SACH	*0-	Y2 - (I/4)(Y1 - Y2)	
LAC	*0,AR1	Y1	
ADD	*0,AR2	YH	
SACL	*-,0,AR1	Y1 - Y1 + YH	
SUB	*-,1	2YH	
T2	T2	T2 - Y1 - YH	
LAC	*0+,0,AR2	XH	
SUB	*	X1	
SACL	T1	X1 - XH - X1	
ADD	*+,1	2X1	
SACL	*0+	X1 - XH + X1	
LT	*+,AR1	XH	

```

MPYK      C32
MAR       *+
LTP       *
          ,Y13
MPYK      C32
SPAC      *,-4
SACH      ,Y13 - (X12 - Y13) * C32
APAC      *0-
LTA       ,X13
SACH      *0+ ,4,AR2
          ,X11 - (X12 + Y13) * C32

MPYK      C32
LTP       *,AR1
MPYK      C32
APAC      *0,-4
SACH      ,X13 - (Y12 + X13) * C32
SPAC
SPAC
NEG
MAR       *+
SACH      *,4,AR2
          ,Y11 + (Y12 - X13) * C32

LAC       T1
SACL      *-
          ,Y12 - T1
LAC       T2
SACL      *0-
          ,X12 - T2
MAR       *+
MAR       *+ ,AR1

LAC       *-
SUB       *+
SACL      *+
          ,Y11 + Y11 - X11
ADD       *+ ,1
          ,2X11
SACL      *0+
          ,X11 + Y11 + X11
LAC       *+
SUB       *-
SACL      *+
          ,X13 - X13 - Y13
ADD       *+ ,1
          ,2Y13
NEG
SACL      *0-
          ,Y13 - (Y13 + X13)
MAR       *+

```

SEND

DIGREV macro para realizar el intercambio de localidades por la  
reversión de dígitos.

DIGREV    SMACRO    LJ

```

LRLK      AR1,XE;
LRLK      AR2,XJ;
          ,AR1 epunta XI
          ,AR1 epunta XJ

ZALH      *,AR2
ADDS      *,AR1
SACL      *+ ,0,AR2
SACH      *+ ,0,AR1
ZALH      *,AR2
ADDS      *,AR1
SACL      *+ ,0,AR2
SACH      *+ ,0,AR1

```

SEND

Rutina principal, la cual llama a los macros definidos con los parámetros necesarios.

.text

```

FF256     1,DPK      0
          SOVM
          SSXM
          SPM        0
          LARP       1

```

Lectura de 256 datos (reales e imaginarios)

```

LRLK      AR1,X0
RPTK      255
IN        *+ ,PA0
RPTK      255
IN        *+ ,PA0

```

Primera pasada

```

LRLK      AR0,256
LRLK      AR1,N64
LRLK      AR2,X0
          ,Val+ del paso entre manposas
          ,X0,X6+X128,N1*2,Y0,Y6+Y128,Y1*2

```

```

NORMAL    1,2,3
NORMAL    2,4,6
NORMAL    3,6,9
NORMAL    4,8,12
NORMAL    5,10,15
NORMAL    6,12,18
NORMAL    7,14,21
NORMAL    8,16,24
NORMAL    9,18,27

```

NORMAL 10,20,30  
 NORMAL 11,22,33  
 NORMAL 12,24,36  
 NORMAL 13,26,39  
 NORMAL 14,28,42  
 NORMAL 15,30,45  
 NORMAL 16,32,48  
 NORMAL 17,34,51  
 NORMAL 18,36,54  
 NORMAL 19,38,57  
 NORMAL 20,40,60  
 NORMAL 21,42,63  
 NORMAL 22,44,66  
 NORMAL 23,46,69  
 NORMAL 24,48,72  
 NORMAL 25,50,75  
 NORMAL 26,52,78  
 NORMAL 27,54,81  
 NORMAL 28,56,84  
 NORMAL 29,58,87  
 NORMAL 30,60,90  
 NORMAL 31,62,93  
 SPCIAL  
 NORMAL 33,66,99  
 NORMAL 34,68,102  
 NORMAL 35,70,105  
 NORMAL 36,72,108  
 NORMAL 37,74,111  
 NORMAL 38,76,114  
 NORMAL 39,78,117  
 NORMAL 40,80,120  
 NORMAL 41,82,123  
 NORMAL 42,84,126  
 NORMAL 43,86,129  
 NORMAL 44,88,132  
 NORMAL 45,90,135  
 NORMAL 46,92,138  
 NORMAL 47,94,141  
 NORMAL 48,96,144  
 NORMAL 49,98,147  
 NORMAL 50,100,150  
 NORMAL 51,102,153  
 NORMAL 52,104,156  
 NORMAL 53,106,159  
 NORMAL 54,108,162  
 NORMAL 55,110,165  
 NORMAL 56,112,168  
 NORMAL 57,114,171

NORMAL 58,116,174  
 NORMAL 59,118,177  
 NORMAL 60,120,180  
 NORMAL 61,122,183  
 NORMAL 62,124,186  
 NORMAL 63,126,189

;  
 ;  
 ;  
 ;

Segunda Pasada

Primer conjunto de mariposas

IRLK AR0,64  
 IRLK AR1,X16  
 IRLK AR2,X0  
 ZERO  
 NORMAL 4,8,12  
 NORMAL 8,16,24  
 NORMAL 12,24,36  
 NORMAL 16,32,48  
 NORMAL 20,40,60  
 NORMAL 24,48,72  
 NORMAL 28,56,84  
 SPCIAL  
 NORMAL 36,72,108  
 NORMAL 40,80,120  
 NORMAL 44,88,132  
 NORMAL 48,96,144  
 NORMAL 52,104,156  
 NORMAL 56,112,168  
 NORMAL 60,120,180

Valor del paso entre mariposas

.X0,X16,X32,X48,X64,X80,X96,X112,X128,X144,X160,X176,X192,X208

.X1,X17,X33,X49,X65,X81,X97,X113,X129,X145,X161,X177,X193,X209

;  
 ;  
 ;

Segundo conjunto de mariposas

IRLK AR1,X80  
 IRLK AR2,X64  
 ZERO  
 NORMAL 4,8,12  
 NORMAL 8,16,24  
 NORMAL 12,24,36  
 NORMAL 16,32,48  
 NORMAL 20,40,60  
 NORMAL 24,48,72  
 NORMAL 28,56,84  
 SPCIAL  
 NORMAL 36,72,108  
 NORMAL 40,80,120

.X0,X80,X96,X112,X64,X160,X176,X192,X112

.X0,X81,X97,X113,X65,X161,X177,X193,X113

NORMAL 44,88,132  
 NORMAL 48,96,144  
 NORMAL 52,104,156  
 NORMAL 56,112,168  
 NORMAL 60,120,180

NORMAL 60,120,180  
 :  
 :  
 Tercera pasada  
 :

Tercer conjunto de mariposas

LRLK ARI,X144  
 LRLK AR2,X128  
 ZERO ;X129,X145,X161,X177,Y129,Y145,Y161,Y177

NORMAL 4,8,12 ;X129,X145,X161,X177,Y129,Y145,Y161,Y177

NORMAL 8,16,24  
 NORMAL 12,24,36  
 NORMAL 16,32,48  
 NORMAL 20,40,60  
 NORMAL 24,48,72  
 NORMAL 28,56,84  
 SPCLAL  
 NORMAL 36,72,108  
 NORMAL 40,80,120  
 NORMAL 44,88,132  
 NORMAL 48,96,144  
 NORMAL 52,104,156  
 NORMAL 56,112,168  
 NORMAL 60,120,180

LARK AR0,16 ;Valor del paso entre mariposas  
 LRLK ARI,X4  
 LRLK AR2,X0

ZERO ;X0,X4,X8,X12,Y0,Y4,Y8,Y12  
 NORMAL 16,32,48  
 SPCLAL  
 NORMAL 48,96,144

LRLK ARI,X20  
 LRLK AR2,X16

ZERO ;X16,X20,X24,X28,Y16,Y20,Y24,Y28  
 NORMAL 16,32,48  
 SPCLAL  
 NORMAL 48,96,144

LRLK ARI,X36  
 LRLK AR2,X32

ZERO ;X32,X36,X40,X44,Y32,Y36,Y40,Y44  
 NORMAL 16,32,48  
 SPCLAL  
 NORMAL 48,96,144

Cuarto conjunto de mariposas

LRLK ARI,X208  
 LRLK AR2,X192  
 ZERO ;X192,X208,X224,X240,Y192,Y208,Y224,Y240

NORMAL 4,8,12 ;X192,X208,X224,X240,Y192,Y208,Y224,Y240

NORMAL 8,16,24  
 NORMAL 12,24,36  
 NORMAL 16,32,48  
 NORMAL 20,40,60  
 NORMAL 24,48,72  
 NORMAL 28,56,84  
 SPCLAL  
 NORMAL 36,72,108  
 NORMAL 40,80,120  
 NORMAL 44,88,132  
 NORMAL 48,96,144  
 NORMAL 52,104,156  
 NORMAL 56,112,168

LRLK ARI,X52  
 LRLK AR2,X48

ZERO ;X48,X52,X56,Y48,Y52,Y56,Y60  
 NORMAL 16,32,48  
 SPCLAL  
 NORMAL 48,96,144

LRLK ARI,X68  
 LRLK AR2,X64

ZERO ;X64,X68,X72,Y64,Y68,Y72,Y76  
 NORMAL 16,32,48  
 SPCLAL  
 NORMAL 48,96,144

LRLK ARI,X84  
 LRLK AR2,X80

ZERO		,X80,X84,X88,X92,Y80,Y84,Y88,Y92	ZERO		,X176,X180,X184,X188,Y176,Y180,Y184,Y188
NORMAL	16,32,48		NORMAL	16,32,48	
SPECIAL			SPECIAL		
NORMAL	48,96,144		NORMAL	48,96,144	
LRLK	AR1,X100		LRLK	AR1,X196	
LRLK	AR2,X96		LRLK	AR2,X192	
ZERO		,X96,X100,X104,X108,Y96,Y100,Y104,Y108	ZERO		,X192,X196,X200,X204,Y192,Y196,Y200,Y204
NORMAL	16,32,48		NORMAL	16,32,48	
SPECIAL			SPECIAL		
NORMAL	48,96,144		NORMAL	48,96,144	
LRLK	AR1,X116		LRLK	AR1,X212	
LRLK	AR2,X112		LRLK	AR2,X208	
ZERO		,X112,X116,X120,X124,Y112,Y116,Y120,Y124	ZERO		,X208,X212,X216,X220,Y208,Y212,Y216,Y220
NORMAL	16,32,48		NORMAL	16,32,48	
SPECIAL			SPECIAL		
NORMAL	48,96,144		NORMAL	48,96,144	
LRLK	AR1,X132		LRLK	AR1,X228	
LRLK	AR2,X128		LRLK	AR2,X224	
ZERO		,X128,X132,X136,X140,Y128,Y132,Y136,Y140	ZERO		,X224,X228,X232,X236,Y224,Y228,Y232,Y236
NORMAL	16,32,48		NORMAL	16,32,48	
SPECIAL			SPECIAL		
NORMAL	48,96,144		NORMAL	48,96,144	
LRLK	AR1,X148		LRLK	AR1,X244	
LRLK	AR2,X144		LRLK	AR2,X240	
ZERO		,X144,X148,X152,X156,Y144,Y148,Y152,Y156	ZERO		,X240,X244,X248,X252,Y240,Y244,Y248,Y252
NORMAL	16,32,48		NORMAL	16,32,48	
SPECIAL			SPECIAL		
NORMAL	48,96,144		NORMAL	48,96,144	
LRLK	AR1,X164				
LRLK	AR2,X160				
ZERO		,X160,X164,X168,X172,Y160,Y164,Y168,Y172			
NORMAL	16,32,48				
SPECIAL					
NORMAL	48,96,144				
LRLK	AR1,X180				
LRLK	AR2,X176				
			Cuarta pasada		
			LARK	AR0,4	Valor del paso entre manpasas
			LRLK	AR1,X1	
			LRLK	AR2,X0	
			ZERO		,X0,X1,X2,X3,Y0,Y1,Y2,Y3
			LRLK	AR1,X5	
			LRLK	AR2,X4	
			ZERO		,X4,X5,X6,X7,Y4,Y5,Y6,Y7

LRLK AR1,X9  
 LRLK AR2,X8  
 ZERO ,X08,X9,X10,X11,Y8,Y9,Y10,Y11  
 LRLK AR1,X13  
 LRLK AR2,X12  
 ZERO ,X12,X13,X14,X15,Y12,Y13,Y14,Y15  
 LRLK AR1,X17  
 LRLK AR2,X16  
 ZERO ,X16,X17,X18,X19,Y16,Y17,Y18,Y19  
 LRLK AR1,X21  
 LRLK AR2,X20  
 ZERO ,X20,X21,X22,X23,Y20,Y21,Y22,Y23  
 LRLK AR1,X25  
 LRLK AR2,X24  
 ZERO ,X24,X25,X26,X27,Y24,Y25,Y26,Y27  
 LRLK AR1,X29  
 LRLK AR2,X28  
 ZERO ,X28,X29,X30,X31,Y28,Y29,Y30,Y31  
 LRLK AR1,X33  
 LRLK AR2,X32  
 ZERO ,X32,X33,X34,X35,Y32,Y33,Y34,Y35  
 LRLK AR1,X37  
 LRLK AR2,X36  
 ZERO ,X36,X37,X38,X39,Y36,Y37,Y38,Y39  
 LRLK AR1,X41  
 LRLK AR2,X40  
 ZERO ,X40,X41,X42,X43,Y40,Y41,Y42,Y43  
 LRLK AR1,X45  
 LRLK AR2,X44  
 ZERO ,X44,X45,X46,X47,Y44,Y45,Y46,Y47  
 LRLK AR1,X49  
 LRLK AR2,X48  
 ZERO ,X48,X49,X50,X51,Y48,Y49,Y50,Y51  
 LRLK AR1,X53  
 LRLK AR2,X52  
 ZERO ,X52,X53,X54,X55,Y52,Y53,Y54,Y55  
 LRLK AR1,X57  
 LRLK AR2,X56  
 ZERO ,X56,X57,X58,X59,Y56,Y57,Y58,Y59  
 LRLK AR1,X61  
 LRLK AR2,X60  
 ZERO ,X60,X61,X62,X63,Y60,Y61,Y62,Y63  
  
 LRLK AR1,X65  
 LRLK AR2,X64  
 ZERO ,X64,X65,X66,X67,Y64,Y65,Y66,Y67  
 LRLK AR1,X69  
 LRLK AR2,X68

ZERO ,X68,X69,X70,X71,Y68,Y69,Y70,Y71  
 LRLK AR1,X73  
 LRLK AR2,X72  
 ZERO ,X72,X73,X74,X75,Y72,Y73,Y74,Y75  
 LRLK AR1,X77  
 LRLK AR2,X76  
 ZERO ,X76,X77,X78,X79,Y76,Y77,Y78,Y79  
 LRLK AR1,X81  
 LRLK AR2,X80  
 ZERO ,X80,X81,X82,X83,Y80,Y81,Y82,Y83  
 LRLK AR1,X85  
 LRLK AR2,X84  
 ZERO ,X84,X85,X86,X87,Y84,Y85,Y86,Y87  
 LRLK AR1,X89  
 LRLK AR2,X88  
 ZERO ,X88,X89,X90,X91,Y88,Y89,Y90,Y91  
 LRLK AR1,X93  
 LRLK AR2,X92  
 ZERO ,X92,X93,X94,X95,Y92,Y93,Y94,Y95  
 LRLK AR1,X97  
 LRLK AR2,X96  
 ZERO ,X96,X97,X98,X99,Y96,Y97,Y98,Y99  
 LRLK AR1,X101  
 LRLK AR2,X100  
 ZERO ,X100,X101,X102,X103,Y100,Y101,Y102,Y103  
 LRLK AR1,X105  
 LRLK AR2,X104  
 ZERO ,X104,X105,X106,X107,Y104,Y105,Y106,Y107  
 LRLK AR1,X109  
 LRLK AR2,X108  
 ZERO ,X108,X109,X110,X111,Y108,Y109,Y110,Y111  
 LRLK AR1,X113  
 LRLK AR2,X112  
 ZERO ,X112,X113,X114,X115,Y112,Y113,Y114,Y115  
 LRLK AR1,X117  
 LRLK AR2,X116  
 ZERO ,X116,X117,X118,X119,Y116,Y117,Y118,Y119  
 LRLK AR1,X121  
 LRLK AR2,X120  
 ZERO ,X120,X121,X122,X123,Y120,Y121,Y122,Y123  
 LRLK AR1,X125  
 LRLK AR2,X124  
 ZERO ,X124,X125,X126,X127,Y124,Y125,Y126,Y127  
  
 LRLK AR1,X129  
 LRLK AR2,X128  
 ZERO ,X128,X129,X130,X131,Y128,Y129,Y130,Y131  
 LRLK AR1,X133

LRLK	AR2,X132	
ZERO		X132,X133,X134,X135,Y132,Y133,Y134,Y135
LRLK	AR1,X137	
LRLK	AR2,X136	
ZERO		X136,X137,X138,X139,Y136,Y137,Y138,Y139
LRLK	AR1,X141	
LRLK	AR2,X140	
ZERO		X140,X141,X142,X143,Y140,Y141,Y142,Y143
LRLK	AR1,X145	
LRLK	AR2,X144	
ZERO		X144,X145,X146,X147,Y144,Y145,Y146,Y147
LRLK	AR1,X149	
LRLK	AR2,X148	
ZERO		X148,X149,X150,X151,Y148,Y149,Y150,Y151
LRLK	AR1,X153	
LRLK	AR2,X152	
ZERO		X152,X153,X154,X155,Y152,Y153,Y154,Y155
LRLK	AR1,X157	
LRLK	AR2,X156	
ZERO		X156,X157,X158,X159,Y156,Y157,Y158,Y159
LRLK	AR1,X161	
LRLK	AR2,X160	
ZERO		X160,X161,X162,X163,Y160,Y161,Y162,Y163
LRLK	AR1,X165	
LRLK	AR2,X164	
ZERO		X164,X165,X166,X167,Y164,Y165,Y166,Y167
LRLK	AR1,X169	
LRLK	AR2,X168	
ZERO		X168,X169,X170,X171,Y168,Y169,Y170,Y171
LRLK	AR1,X173	
LRLK	AR2,X172	
ZERO		X172,X173,X174,X175,Y172,Y173,Y174,Y175
LRLK	AR1,X177	
LRLK	AR2,X176	
ZERO		X176,X177,X178,X179,Y176,Y177,Y178,Y179
LRLK	AR1,X181	
LRLK	AR2,X180	
ZERO		X180,X181,X182,X183,Y180,Y181,Y182,Y183
LRLK	AR1,X185	
LRLK	AR2,X184	
ZERO		X184,X185,X186,X187,Y184,Y185,Y186,Y187
LRLK	AR1,X189	
LRLK	AR2,X188	
ZERO		X188,X189,X190,X191,Y188,Y189,Y190,Y191
LRLK	AR1,X193	
LRLK	AR2,X192	
ZERO		X192,X193,X194,X195,Y192,Y193,Y194,Y195

LRLK	AR1,X197	
LRLK	AR2,X196	
ZERO		X196,X197,X198,X199,Y196,Y197,Y198,Y199
LRLK	AR1,X201	
LRLK	AR2,X200	
ZERO		X200,X201,X202,X203,Y200,Y201,Y202,Y203
LRLK	AR1,X205	
LRLK	AR2,X204	
ZERO		X204,X205,X206,X207,Y204,Y205,Y206,Y207
LRLK	AR1,X209	
LRLK	AR2,X208	
ZERO		X208,X209,X210,X211,Y208,Y209,Y210,Y211
LRLK	AR1,X213	
LRLK	AR2,X212	
ZERO		X212,X213,X214,X215,Y212,Y213,Y214,Y215
LRLK	AR1,X217	
LRLK	AR2,X216	
ZERO		X216,X217,X218,X219,Y216,Y217,Y218,Y219
LRLK	AR1,X221	
LRLK	AR2,X220	
ZERO		X220,X221,X222,X223,Y220,Y221,Y222,Y223
LRLK	AR1,X225	
LRLK	AR2,X224	
ZERO		X224,X225,X226,X227,Y224,Y225,Y226,Y227
LRLK	AR1,X229	
LRLK	AR2,X228	
ZERO		X228,X229,X230,X231,Y228,Y229,Y230,Y231
LRLK	AR1,X233	
LRLK	AR2,X232	
ZERO		X232,X233,X234,X235,Y232,Y233,Y234,Y235
LRLK	AR1,X237	
LRLK	AR2,X236	
ZERO		X236,X237,X238,X239,Y236,Y237,Y238,Y239
LRLK	AR1,X241	
LRLK	AR2,X240	
ZERO		X240,X241,X242,X243,Y240,Y241,Y242,Y243
LRLK	AR1,X245	
LRLK	AR2,X244	
ZERO		X244,X245,X246,X247,Y244,Y245,Y246,Y247
LRLK	AR1,X249	
LRLK	AR2,X248	
ZERO		X248,X249,X250,X251,Y248,Y249,Y250,Y251
LRLK	AR1,X253	
LRLK	AR2,X252	
ZERO		X252,X253,X254,X255,Y252,Y253,Y254,Y255

Contador de reversión de dígito para la FFT de Radix-4

LARP	1
DIGREV	1,64
DIGREV	2,128
DIGREV	3,192
DIGREV	4,16
DIGREV	5,80
DIGREV	6,144
DIGREV	7,208
DIGREV	8,32
DIGREV	9,96
DIGREV	10,160
DIGREV	11,224
DIGREV	13,112
DIGREV	14,176
DIGREV	15,240
DIGREV	17,68
DIGREV	18,132
DIGREV	19,196
DIGREV	21,84
DIGREV	22,148
DIGREV	23,212
DIGREV	24,36
DIGREV	25,100
DIGREV	26,164
DIGREV	27,228
DIGREV	28,52
DIGREV	39,116
DIGREV	30,180
DIGREV	31,244
DIGREV	33,72
DIGREV	34,136
DIGREV	35,200
DIGREV	37,88
DIGREV	38,152
DIGREV	39,216
DIGREV	41,104
DIGREV	42,168
DIGREV	43,232
DIGREV	44,56
DIGREV	45,120
DIGREV	46,184
DIGREV	47,248
DIGREV	49,76
DIGREV	50,140
DIGREV	51,204
DIGREV	53,92
DIGREV	54,156
DIGREV	55,220

DIGREV	57,108
DIGREV	58,172
DIGREV	59,236
DIGREV	61,124
DIGREV	62,188
DIGREV	63,252
DIGREV	66,129
DIGREV	67,193
DIGREV	69,81
DIGREV	70,145
DIGREV	71,209
DIGREV	73,97
DIGREV	74,161
DIGREV	75,225
DIGREV	77,113
DIGREV	78,177
DIGREV	79,241
DIGREV	82,133
DIGREV	83,197
DIGREV	86,149
DIGREV	87,213
DIGREV	89,101
DIGREV	90,165
DIGREV	91,229
DIGREV	93,117
DIGREV	94,181
DIGREV	95,245
DIGREV	98,137
DIGREV	99,201
DIGREV	102,153
DIGREV	103,217
DIGREV	106,169
DIGREV	107,233
DIGREV	109,121
DIGREV	110,185
DIGREV	111,249
DIGREV	114,141
DIGREV	115,205
DIGREV	118,157
DIGREV	119,221
DIGREV	122,173
DIGREV	123,237
DIGREV	126,189
DIGREV	127,253
DIGREV	131,194
DIGREV	134,146
DIGREV	135,210
DIGREV	138,162

DIGREV	139,226
DIGREV	142,178
DIGREV	143,242
DIGREV	147,198
DIGREV	151,214
DIGREV	154,166
DIGREV	155,230
DIGREV	158,182
DIGREV	159,246
DIGREV	163,202
DIGREV	167,218
DIGREV	171,234
DIGREV	174,186
DIGREV	175,250
DIGREV	179,206
DIGREV	183,222
DIGREV	187,238
DIGREV	191,254
DIGREV	199,211
DIGREV	203,227
DIGREV	207,243
DIGREV	219,231
DIGREV	223,247
DIGREV	239,251

Saldos de datos de la FFT

LARP	1
LRLK	ARI,XO
RPTK	255
OUT	*+,PA1
RPTK	255
OUT	*+,PA1
WHOA	B
	WHOA
	.end

# Apéndice B

```
PROGRAM FHT;                                { Félix Cuéllar [Hawk] 496602 ISE80 }

USES
  CRT.PRINTER;

VAR
  DATOS,DATOST,TABLACOS,TABLASEN             :ARRAY [0..127] OF REAL;
  DATOSI,TABLACOSI,TABLASENI,TABLATANI      :ARRAY [0..127] OF LONGINT;
  CONT,INDICE,FACTOR,OPC                     :BYTE;
  ESCALAR                                     :WORD;
  HARTLEY,VN15_TF,VN15_FT                    :ARRAY [0..127] OF REAL;
  FHTD_F,FHTD_T,FHT_F,FHT_T                :ARRAY [0..127] OF REAL;
  FHTI5_F,FHTI5_T                            :ARRAY [0..127] OF LONGINT;
  ARH                                         :TEXT;

PROCEDURE OBT_HAR;                           { Transformada de Hartley }
VAR
  IND1,IND2                                   :BYTE;
  THETA                                       :REAL;
BEGIN
  FOR IND1:=0 TO 127 DO
  BEGIN
    HARTLEY[IND1]:=0;
    FOR IND2:=0 TO 127 DO
    BEGIN
      THETA:=(2 * Pi * IND1 * IND2) / 128;
      HARTLEY[IND1]:=HARTLEY[IND1] + (DATOS[IND2] * (COS(THETA) + SIN(THETA)));
    END;
    HARTLEY[IND1]:=HARTLEY[IND1] / ESCALAR;
    WRITE(' ');
  END;
  WRITELN;
END;

PROCEDURE TABLAS;                             { Creación de tablas de senos y cosenos (real) }
VAR
  IND                                         :BYTE;
  THETA                                       :REAL;
BEGIN
  FOR IND:=0 TO 127 DO
  BEGIN
    THETA:=(2 * Pi * IND) / 128;
    TABLACOS[IND]:=-COS(THETA);
    TABLASEN[IND]:=-SIN(THETA);
  END;
END;

PROCEDURE TABLASI;                            { Creación de tablas de senos y cosenos (entero) }
VAR
  IND                                         :BYTE;
  THETA                                       :REAL;
BEGIN
  FOR IND:=0 TO 127 DO
  BEGIN
    THETA:=(2 * Pi * IND) / 128;
    TABLACOSI[IND]:=-TRUNC(COS(THETA) * ESCALAR);
    TABLASENI[IND]:=-TRUNC(SIN(THETA) * ESCALAR);
  END;
END;
```

```

IF TABLACOS[[IND] = 0 THEN
TABLATANI[[IND]:=TABLASEN[[IND]
ELSE
TABLATANI[[IND]:=TRUNC((SIN(THETA) / COS(THETA)) * ESCALAR);
END;
END;

```

```

PROCEDURE HARTLEY_PER;      { Reversion de bits (real) }
VAR
IND1,IND2,FAC1,FAC2,BIT      :WORD;
TEMP                          :REAL;
BEGIN
FOR IND1:=1 TO 126 DO
BEGIN
IND2:=0;
FAC1:=1;
FAC2:=64;
FOR BIT:=1 TO 7 DO
BEGIN
IF (IND1 AND FAC1) <> 0 THEN
IND2:=IND2 OR FAC2;
FAC1:=FAC1 SHL 1;
FAC2:=FAC2 SHR 1;
END;
IF IND1 < IND2 THEN
BEGIN
TEMP:=DATOS[IND1];
DATOS[IND1]:=DATOS[IND2];
DATOS[IND2]:=TEMP;
END;
END;
END;

```

```

PROCEDURE HARTLEY1;        { Primera mariposa (real) }
VAR
IND1,IND2                      :BYTE;
TCOSSEN                         :REAL;
BEGIN
IND1:=0;
IND2:=1;
REPEAT
TCOSSEN:=DATOS[IND2];
DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
DATOS[IND1]:=(DATOS[IND1] + TCOSSEN) / FACTOR;
IND1:=IND1 + 2;
IND2:=IND2 - 2;
UNTIL IND1 = 128;
END;

```

```

PROCEDURE HARTLEY2;        { Segunda mariposa (real) }
VAR
IND1,IND2                      :BYTE;
TCOSSEN                         :REAL;
BEGIN
IND1:=0;
IND2:=2;
REPEAT
TCOSSEN:=DATOS[IND2];
DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
DATOS[IND1]:=(DATOS[IND1] + TCOSSEN) / FACTOR;
INC(IND1);
INC(IND2);
TCOSSEN:=(DATOS[IND2] * TABLACOS[32] + (DATOS[IND2] * TABLASEN[32]);
DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
DATOS[IND1]:=(DATOS[IND1] + TCOSSEN) / FACTOR;
IND1:=IND1 + 3;
IND2:=IND2 + 3;
UNTIL IND1 = 128;

```

END;

```
PROCEDURE HARTLEY3;          { Tercera mariposa (real) }
VAR
  IND1,IND2
  TCOSSEN,TDATOS
BEGIN
  IND1:=0;
  IND2:=4;
  REPEAT
    TCOSSEN:=(DATOS[IND2]);
    DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    IND1:=IND1 - 2;
    IND2:=IND2 - 2;
    TCOSSEN:=(DATOS[IND2] * TABLACOS[32]) - (DATOS[IND2] * TABLASEN[32]);
    DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    DEC(IND1);
    DEC(IND2);
    TCOSSEN:=(DATOS[IND2] * TABLACOS[16]) - (DATOS[IND2 - 2] * TABLASEN[16]);
    TDATOS:=DATOS[IND2];
    DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    IND1:=IND1 + 2;
    IND2:=IND2 - 2;
    TCOSSEN:=(DATOS[IND2] * TABLACOS[48]) - (TDATOS * TABLASEN[48]);
    DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    IND1:=IND1 + 5;
    IND2:=IND2 + 5;
  UNTIL IND1 = 128;
END;
```

```
PROCEDURE HARTLEY4;          { Cuarta mariposa (real) }
VAR
  IND1,IND2,IND3,IND4,IND5,REP
  TCOSSEN,TDATOS
BEGIN
  IND1:=0;
  IND2:=-8;
  REPEAT
    TCOSSEN:=DATOS[IND2];
    DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    IND1:=IND1 + 4;
    IND2:=IND2 - 4;
    TCOSSEN:=(DATOS[IND2] * TABLACOS[32]) - (DATOS[IND2] * TABLASEN[32]);
    DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    IND1:=IND1 - 3;
    IND2:=IND2 - 3;
    IND3:=8;
    IND4:=6;
    IND5:=48;
  FOR REP:=1 TO 3 DO
  BEGIN
    TCOSSEN:=(DATOS[IND2] * TABLACOS[IND3]) + (DATOS[IND2 - IND4] * TABLASEN[IND3]);
    TDATOS:=DATOS[IND2];
    DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    DATOS[IND1]:=(DATOS[IND1] + TCOSSEN) / FACTOR;
    IND1:=IND1 + IND4;
    IND2:=IND2 - IND4;
    TCOSSEN:=(DATOS[IND2] * TABLACOS[IND3 - IND5]) - (TDATOS * TABLASEN[IND3 - IND5]);
    DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
    DATOS[IND1]:=(DATOS[IND1] + TCOSSEN) / FACTOR;
    IND5:=IND5 - 16;
    IND4:=IND4 - 2;
  END;
```

```

IND3:=IND3 + 8;
IND2:={IND2 - IND4;
DEC(IND2);
IND1:={IND1 - IND4;
DEC(IND1);
END;
IND1:={IND1 - 12;
IND2:={IND2 - 12;
UNTIL IND1 = 128;
END;

```

PROCEDURE HARTLEY5;

{ Quinta mariposa (real) }

```

VAR
IND1,IND2,IND3,IND4,IND5,REP
TCOSSEN,TDATOS
:BYTE;
:REAL;
BEGIN
IND1:=0;
IND2:=16;
REPEAT
TCOSSEN:={DATOS[IND2];
DATOS[IND2]:={DATOS[IND1] - TCOSSEN} / FACTOR;
DATOS[IND1]:={DATOS[IND1] - TCOSSEN} / FACTOR;
IND1:={IND1 - 8;
IND2:={IND2 - 8;
TCOSSEN:={DATOS[IND2] * TABLACOS[32] + (DATOS[IND2] * TABLASEN[32]);
DATOS[IND2]:={DATOS[IND1] - TCOSSEN} / FACTOR;
DATOS[IND1]:={DATOS[IND1] + TCOSSEN} / FACTOR;
IND1:={IND1 - 7;
IND2:={IND2 - 7;
IND3:=4;
IND4:=14;
IND5:=56;
FOR REP:=1 TO 7 DO
BEGIN
TCOSSEN:={DATOS[IND2] * TABLACOS[IND3] + (DATOS[IND2 + IND4] * TABLASEN[IND3]);
TDATOS:=DATOS[IND2];
DATOS[IND2]:={DATOS[IND1] - TCOSSEN} / FACTOR;
DATOS[IND1]:={DATOS[IND1] - TCOSSEN} / FACTOR;
IND1:=IND1 - IND4;
IND2:=IND2 + IND4;
TCOSSEN:={DATOS[IND2] * TABLACOS[IND3 + IND5] + (TDATOS * TABLASEN[IND3 + IND5]);
DATOS[IND2]:={DATOS[IND1] - TCOSSEN} / FACTOR;
DATOS[IND1]:={DATOS[IND1] + TCOSSEN} / FACTOR;
IND5:=IND5 - 8;
IND4:=IND4 - 2;
IND3:=IND3 + 4;
IND2:=IND2 - IND4;
DEC(IND2);
IND1:=IND1 - IND4;
DEC(IND1);
END;
IND1:=IND1 + 24;
IND2:=IND2 + 24;
UNTIL IND1 = 128;
END;

```

PROCEDURE HARTLEY6;

{ Sexta mariposa (real) }

```

VAR
IND1,IND2,IND3,IND4,IND5,REP
TCOSSEN,TDATOS
:BYTE;
:REAL;
BEGIN
IND1:=0;
IND2:=32;
REPEAT
TCOSSEN:={DATOS[IND2];
DATOS[IND2]:={DATOS[IND1] - TCOSSEN} / FACTOR;
DATOS[IND1]:={DATOS[IND1] - TCOSSEN} / FACTOR;
IND1:=IND1 + 16;

```

```

IND2:=IND2 + 16;
TCOSSEN:=(DATOS[IND2] * TABLACOS[32]) - (DATOS[IND2] * TABLASEN[32]);
DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
IND1:=IND1 - 15;
IND2:=IND2 - 15;
IND3:=2;
IND4:=30;
IND5:=60;
FOR REP:=1 TO 15 DO
BEGIN
TCOSSEN:=(DATOS[IND2] * TABLACOS[IND3]) - (DATOS[IND2 - IND4] * TABLASEN[IND3]);
TDATOS:=DATOS[IND2];
DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
IND1:=IND1 - IND4;
IND2:=IND2 - IND4;
TCOSSEN:=(DATOS[IND2] * TABLACOS[IND3 - IND5]) - (TDATOS * TABLASEN[IND3 - IND5]);
DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
IND5:=IND5 - 4;
IND4:=IND4 - 2;
IND3:=IND3 - 2;
IND2:=IND2 - IND4;
DEC(IND2);
IND1:=IND1 - IND4;
DEC(IND1);
END;
IND1:=IND1 + 48;
IND2:=IND2 + 48;
UNTIL IND1 = 128;
END;

```

```

PROCEDURE HARTLEY7;           { Séptima mariposa (real) }
VAR
IND1,IND2,IND3,IND4,IND5,REP           :BYTE;
TCOSSEN,TDATOS                       :REAL;
BEGIN
IND1:=0;
IND2:=64;
REPEAT
TCOSSEN:=(DATOS[IND2];
DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
IND1:=IND1 - 32;
IND2:=IND2 - 32;
TCOSSEN:=(DATOS[IND2] * TABLACOS[32]) - (DATOS[IND2] * TABLASEN[32]);
DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
IND1:=IND1 - 31;
IND2:=IND2 - 31;
IND3:=1;
IND4:=62;
IND5:=62;
FOR REP:=1 TO 31 DO
BEGIN
TCOSSEN:=(DATOS[IND2] * TABLACOS[IND3]) + (DATOS[IND2 + IND4] * TABLASEN[IND3]);
TDATOS:=DATOS[IND2];
DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
IND1:=IND1 - IND4;
IND2:=IND2 - IND4;
TCOSSEN:=(DATOS[IND2] * TABLACOS[IND3 + IND5]) + (TDATOS * TABLASEN[IND3 + IND5]);
DATOS[IND2]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
DATOS[IND1]:=(DATOS[IND1] - TCOSSEN) / FACTOR;
IND5:=IND5 - 2;
IND4:=IND4 - 2;
INC(IND3);

```

```

IND2:=IND2 - IND4;
DEC(IND2);
IND1:=IND1 - IND4;
DEC(IND1);
END;
IND1:=IND1 - 96;
IND2:=IND2 - 96;
UNTIL IND1 = 128;
END;

```

```

PROCEDURE HARTLEY_PERI; { Reversi n de bits (entero) }
VAR
IND1,IND2,FAC1,FAC2,BIT
TEMP
:WORD;
:LONGINT;
BEGIN
FOR IND1:=1 TO 126 DO
BEGIN
IND2:=0;
FAC1:=1;
FAC2:=64;
FOR BIT:=1 TO 7 DO
BEGIN
IF (IND1 AND FAC1) <> 0 THEN
IND2:=IND2 OR FAC2;
FAC1:=FAC1 SHL 1;
FAC2:=FAC2 SHR 1;
END;
IF IND1 < IND2 THEN
BEGIN
TEMP:=DATOS[[IND1];
DATOS[[IND1]]:=DATOS[[IND2];
DATOS[[IND2]]:=TEMP;
END;
END;
END;
END;

```

```

PROCEDURE HARTLEY1I; { Primera mariposa (entero) }
VAR
IND1,IND2
TCOSSEN
:BYTE;
:LONGINT;
BEGIN
IND1:=0;
IND2:=1;
REPEAT
TCOSSEN:=DATOS[[IND2];
DATOS[[IND2]]:=(DATOS[[IND1] - TCOSSEN) DIV FACTOR;
DATOS[[IND1]]:=(DATOS[[IND1] + TCOSSEN) DIV FACTOR;
IND1:=IND1 + 2;
IND2:=IND2 + 2;
UNTIL IND1 = 128;
END;

```

```

PROCEDURE HARTLEY2I; { Segunda mariposa (entero) }
VAR
IND1,IND2
TCOSSEN
:BYTE;
:LONGINT;
BEGIN
IND1:=0;
IND2:=2;
REPEAT
TCOSSEN:=DATOS[[IND2];
DATOS[[IND2]]:=(DATOS[[IND1] - TCOSSEN) DIV FACTOR;
DATOS[[IND1]]:=(DATOS[[IND1] - TCOSSEN) DIV FACTOR;
INC(IND1);
INC(IND2);
TCOSSEN:=DATOS[[IND2];
DATOS[[IND2]]:=(DATOS[[IND1] - TCOSSEN) DIV FACTOR;
DATOS[[IND1]]:=(DATOS[[IND1] + TCOSSEN) DIV FACTOR;

```

```

IND1:=IND1 - 3;
IND2:=IND2 + 3;
UNTIL IND1 = 128;
END;

```

```

PROCEDURE HARTLEY3;          { Tercera mariposa (entero) }
VAR
IND1,IND2                    :BYTE;
TCOSSEN,TDATOS              :LONGINT;
BEGIN
IND1:=0;
IND2:=4;
REPEAT
TCOSSEN:=DATOSI[IND2];
DATOSI[IND2]:=(DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:=(DATOSI[IND1] + TCOSSEN) DIV FACTOR;
INC(IND1);
INC(IND2);
TCOSSEN:=((DATOSI[IND2] + DATOSI[IND2 - 2]) * TABLACOSI[16]) DIV ESCALAR;
TDATOS:=DATOSI[IND2];
DATOSI[IND2]:=(DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:=(DATOSI[IND1] - TCOSSEN) DIV FACTOR;
INC(IND1);
INC(IND2);
TCOSSEN:=DATOSI[IND2];
DATOSI[IND2]:=(DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:=(DATOSI[IND1] + TCOSSEN) DIV FACTOR;
INC(IND1);
INC(IND2);
TCOSSEN:=((DATOSI[IND2] * TDATOS) * TABLACOSI[48]) DIV ESCALAR;
DATOSI[IND2]:=(DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:=(DATOSI[IND1] - TCOSSEN) DIV FACTOR;
IND1:=IND1 + 5;
IND2:=IND2 + 5;
UNTIL IND1 = 128;
END;

```

```

PROCEDURE HARTLEY4;        { Cuarta mariposa (entero) }
VAR
IND1,IND2,IND3,IND4,IND5,REP :BYTE;
TCOSSEN,TDATOS              :LONGINT;
BEGIN
IND1:=0;
IND2:=8;
REPEAT
TCOSSEN:=DATOSI[IND2];
DATOSI[IND2]:=(DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:=(DATOSI[IND1] + TCOSSEN) DIV FACTOR;
IND1:=IND1 - 4;
IND2:=IND2 + 4;
TCOSSEN:=DATOSI[IND2];
DATOSI[IND2]:=(DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:=(DATOSI[IND1] + TCOSSEN) DIV FACTOR;
IND1:=IND1 - 3;
IND2:=IND2 - 3;
IND3:=8;
IND4:=6;
IND5:=48;
FOR REP:=1 TO 3 DO
BEGIN
TCOSSEN:=((DATOSI[IND2] * TABLACOSI[IND3]) + (DATOSI[IND2 + IND4] * TABLASINI[IND3])) DIV ESCALAR;
TDATOS:=DATOSI[IND2];
DATOSI[IND2]:=(DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:=(DATOSI[IND1] + TCOSSEN) DIV FACTOR;
IND1:=IND1 + IND4;
IND2:=IND2 + IND4;
TCOSSEN:=((DATOSI[IND2] * TABLACOSI[IND3 + IND5]) + (TDATOS * TABLASINI[IND3 + IND5])) DIV ESCALAR;
DATOSI[IND2]:=(DATOSI[IND1] - TCOSSEN) DIV FACTOR;

```

```

DATOS[[IND1]:= (DATOS[[IND1] - TCOSSEN) DIV FACTOR;
IND5:=IND5 - 16;
IND4:=IND4 - 2;
IND3:=IND3 - 8;
IND2:=IND2 - IND4;
DEC(IND2);
IND1:=IND1 - IND4;
DEC(IND1);
END;
IND1:=IND1 - 12;
IND2:=IND2 - 12;
UNTIL IND1 = 128;
END;

```

PROCEDURE HARTLEY5; { Quinta mariposa (entero) }

```

VAR
IND1,IND2,IND3,IND4,IND5,REP :BYTE;
TCOSSEN,TDATOS :LONGINT;
BEGIN
IND1:=0;
IND2:=16;
REPEAT
TCOSSEN:=DATOS[[IND2];
DATOS[[IND2]:= (DATOS[[IND1] - TCOSSEN) DIV FACTOR;
DATOS[[IND1]:= (DATOS[[IND1] - TCOSSEN) DIV FACTOR;
IND1:=IND1 + 8;
IND2:=IND2 + 8;
TCOSSEN:=DATOS[[IND2];
DATOS[[IND2]:= (DATOS[[IND1] - TCOSSEN) DIV FACTOR;
DATOS[[IND1]:= (DATOS[[IND1] - TCOSSEN) DIV FACTOR;
IND1:=IND1 - 7;
IND2:=IND2 - 7;
IND3:=4;
IND4:=14;
IND5:=56;
FOR REP:=1 TO 7 DO
BEGIN
TCOSSEN:=((DATOS[[IND2] * TABLACOSI[[IND3] - (DATOS[[IND2 - IND4] * TABLASANI[[IND3]]) DIV ESCALAR;
TDATOS:=DATOS[[IND2];
DATOS[[IND2]:= (DATOS[[IND1] - TCOSSEN) DIV FACTOR;
DATOS[[IND1]:= (DATOS[[IND1] - TCOSSEN) DIV FACTOR;
IND1:=IND1 - IND4;
IND2:=IND2 + IND4;
TCOSSEN:=((DATOS[[IND2] * TABLACOSI[[IND3 + IND5]] - (TDATOS * TABLASANI[[IND3 + IND5]]) DIV ESCALAR;
DATOS[[IND2]:= (DATOS[[IND1] - TCOSSEN) DIV FACTOR;
DATOS[[IND1]:= (DATOS[[IND1] - TCOSSEN) DIV FACTOR;
IND5:=IND5 - 8;
IND4:=IND4 - 2;
IND3:=IND3 + 4;
IND2:=IND2 - IND4;
DEC(IND2);
IND1:=IND1 - IND4;
DEC(IND1);
END;
IND1:=IND1 + 24;
IND2:=IND2 + 24;
UNTIL IND1 = 128;
END;

```

PROCEDURE HARTLEY6; { Sexta mariposa (entero) }

```

VAR
IND1,IND2,IND3,IND4,IND5,REP :BYTE;
TCOSSEN,TDATOS :LONGINT;
BEGIN
IND1:=0;
IND2:=32;
REPEAT
TCOSSEN:=DATOS[[IND2];

```

```

DATOSI[IND2]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
IND1:=IND1 - 16;
IND2:=IND2 - 16;
TCOSSEN:=DATOSI[IND2];
DATOSI[IND2]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
IND1:=IND1 - 15;
IND2:=IND2 - 15;
IND3:=2;
IND4:=30;
IND5:=60;
FOR REP:=1 TO 15 DO
BEGIN
TCOSSEN:= ((DATOSI[IND2] * TABLACOSI[IND3]) + (DATOSI[IND2 - IND4] * TABLASANI[IND3])) DIV ESCALAR;
TDATOS:=DATOSI[IND2];
DATOSI[IND2]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
IND1:=IND1 - IND4;
IND2:=IND2 - IND4;
TCOSSEN:= ((DATOSI[IND2] * TABLACOSI[IND3 + IND5]) + (TDATOS * TABLASANI[IND3 - IND5])) DIV ESCALAR;
DATOSI[IND2]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
IND5:=IND5 - 4;
IND4:=IND4 - 2;
IND3:=IND3 - 2;
IND2:=IND2 - IND4;
DEC(IND2);
IND1:=IND1 - IND4;
DEC(IND1);
END;
IND1:=IND1 - 48;
IND2:=IND2 - 48;
UNTIL IND1 = 128;
END;

```

```

PROCEDURE HARTLEY7;           { Septima mariposa (entero) }
VAR
IND1,IND2,IND3,IND4,IND5,REP      :BYTE;
TCOSSEN,TDATOS                    :LONGINT;
BEGIN
IND1:=0;
IND2:=64;
REPEAT
TCOSSEN:=DATOSI[IND2];
DATOSI[IND2]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:= (DATOSI[IND1] + TCOSSEN) DIV FACTOR;
IND1:=IND1 + 32;
IND2:=IND2 + 32;
TCOSSEN:=DATOSI[IND2];
DATOSI[IND2]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:= (DATOSI[IND1] + TCOSSEN) DIV FACTOR;
IND1:=IND1 + 31;
IND2:=IND2 + 31;
IND3:=1;
IND4:=62;
IND5:=62;
FOR REP:=1 TO 31 DO
BEGIN
TCOSSEN:= ((DATOSI[IND2] * TABLACOSI[IND3]) + (DATOSI[IND2 + IND4] * TABLASANI[IND3])) DIV ESCALAR;
TDATOS:=DATOSI[IND2];
DATOSI[IND2]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:= (DATOSI[IND1] + TCOSSEN) DIV FACTOR;
IND1:=IND1 + IND4;
IND2:=IND2 + IND4;
TCOSSEN:= ((DATOSI[IND2] * TABLACOSI[IND3 + IND5]) + (TDATOS * TABLASANI[IND3 - IND5])) DIV ESCALAR;
DATOSI[IND2]:= (DATOSI[IND1] - TCOSSEN) DIV FACTOR;
DATOSI[IND1]:= (DATOSI[IND1] + TCOSSEN) DIV FACTOR;

```

```

IND5:=IND5 - 2;
IND4:=IND4 - 2;
INC(IND3);
IND2:=IND2 - IND4;
DEC(IND2);
IND1:=IND1 - IND4;
DEC(IND1);
END;
IND1:=IND1 - 96;
IND2:=IND2 - 96;
UNTIL IND1 = 128;
END;

```

```

PROCEDURE FLO_ENT;           { Desnormalizar datos }

```

```

VAR
IND
BEGIN
:BYTE;
FOR IND:=0 TO 127 DO
DATOS[IND]:=TRUNC(DATOS[IND] * ESCALAR);
END;

```

```

PROCEDURE ENT_FLO;         { Normalizar datos }

```

```

VAR
IND
BEGIN
:BYTE;
FOR IND:=0 TO 127 DO
DATOS[IND]:=DATOS[IND] / ESCALAR;
END;

```

```

PROCEDURE C_HARTLEY;      { FHT (real) }

```

```

BEGIN
HARTLEY_PER;
HARTLEY1;
HARTLEY2;
HARTLEY3;
HARTLEY4;
HARTLEY5;
HARTLEY6;
HARTLEY7;
END;

```

```

PROCEDURE C_HARTLEY1;     { FHT (entero) }

```

```

BEGIN
HARTLEY_PER1;
HARTLEY11;
HARTLEY21;
HARTLEY31;
HARTLEY41;
HARTLEY51;
HARTLEY61;
HARTLEY71;
END;

```

```

BEGIN

```

```

CLRSCR;
FOR CONT:=0 TO 7 DO
BEGIN

```

```

INDICE:=CONT * 16;
DATOST[INDICE]:=0.1;
DATOST[INDICE + 1]:=0.2;
DATOST[INDICE + 2]:=0.25;
DATOST[INDICE + 3]:=0.12;
DATOST[INDICE + 4]:=0.36;
DATOST[INDICE + 5]:=0.4;
DATOST[INDICE + 6]:=0.8;
DATOST[INDICE + 7]:=0.66;
DATOST[INDICE + 8]:=0.66;
DATOST[INDICE + 9]:=0.66;

```

```

DATOST[INDICE - 10]:=0.77;
DATOST[INDICE - 11]:=0.11;
DATOST[INDICE - 12]:=0.567;
DATOST[INDICE - 13]:=0.517;
DATOST[INDICE - 14]:=0.9;
DATOST[INDICE - 15]:=0.95;
END;

```

```

FOR CONT:=0 TO 127 DO
  DATOS[CONT]:=DATOST[CONT];
  ESCALAR:=128;
  OBT_HAR;
  FOR CONT:=0 TO 127 DO
    FHTD_F[CONT]:=HARTLEY[CONT];
  FOR CONT:=0 TO 127 DO
    DATOS[CONT]:=HARTLEY[CONT];
  ESCALAR:=1;
  OBT_HAR;
  FOR CONT:=0 TO 127 DO
    FHTD_T[CONT]:=HARTLEY[CONT];

```

```

FOR CONT:=0 TO 127 DO
  DATOS[CONT]:=DATOST[CONT];
  TABLAS;
  FACTOR:=2;
  C_HARTLEY;
  FOR CONT:=0 TO 127 DO
    FHT_F[CONT]:=DATOS[CONT];
  FACTOR:=1;
  C_HARTLEY;
  FOR CONT:=0 TO 127 DO
    FHT_T[CONT]:=DATOS[CONT];

```

```

FOR CONT:=0 TO 127 DO
  DATOS[CONT]:=DATOST[CONT];
  ESCALAR:=32767;
  TABLAS;
  FLO_ENT;
  FACTOR:=2;
  C_HARTLEY;
  FOR CONT:=0 TO 127 DO
    FHT15_F[CONT]:=DATOS[CONT];
  ENT_FLO;
  FOR CONT:=0 TO 127 DO
    VN15_TF[CONT]:=DATOS[CONT];
  FACTOR:=1;
  C_HARTLEY;
  FOR CONT:=0 TO 127 DO
    FHT15_T[CONT]:=DATOS[CONT];
  ENT_FLO;
  FOR CONT:=0 TO 127 DO
    VN15_FT[CONT]:=DATOS[CONT];

```

```

ASSIGN(ARH,'RFHT.TXT');
REWRITE(ARH);
WRITELN(ARH,' Transformada de Hartley ');
WRITELN(ARH);
WRITE(ARH,' Dato DHT Real 48 bits FHT Real 48 bits ');
WRITELN(ARH,'FHT Entera 15 bits Re(FHT)-En(FHT) ');
WRITELN(ARH);
FOR CONT:=0 TO 127 DO
  BEGIN
    WRITE(ARH,DATOST[CONT]:1:3,'FHTD_F[CONT]','FHT_F[CONT],');
    WRITELN(ARH,VN15_TF[CONT]','FHT_F[CONT] - VN15_TF[CONT]);
  END;
WRITELN(ARH);
WRITELN(ARH);
WRITELN(ARH,' Transformada Inversa de Hartley');

```

```
WRITELN(ARH);
WRITE(ARH,' Data DHT Real 48 bits FHT Real 48 bits');
WRITELN(ARH,'FHT Entera 15 bus Re(FHT)-En(FHT) ');
WRITELN(ARH);
FOR CONT:=0 TO 127 DO
BEGIN
WRITE(ARH,DATEOST[CONT]:1:3,' FHTD_T[CONT]:'FHT_T[CONT]');
WRITELN(ARH,VN15_FT[CONT]:'FHT_T[CONT]-VN15_FT[CONT]');
END;
CLOSE(ARH);
END.
```

# Apéndice **C**

## C.1 Transformada Directa de Hartley

Dato	DHT Real 48 bits	FHT Real 48 bits	FHT Entera 15 bits	Re(FHT) - En(FHT)
0.100	5.0149999998E-01	5.0150000000E-01	5.0144962920E-01	5.0370798817E-05
0.200	2.1316282073E-14	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.250	2.3447910280E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.120	-1.1510792319E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.360	-7.1764816312E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.400	-1.0302869669E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.800	-1.9682033781E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	-3.0340174817E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	-1.2479484133E-01	-1.2479484133E-01	-1.2475966674E-01	-3.5174593449E-05
0.660	8.3133500084E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.770	-4.2632564146E-14	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.110	1.2114753645E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.567	-1.8403056856E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.517	-4.0554226644E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.900	-1.1670664435E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.950	-1.2922996007E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.100	-5.7975635824E-02	-5.7975635816E-02	-5.7954649495E-02	-2.0986320806E-05
0.200	1.8678392166E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.250	-1.2452261444E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00

0.120	8.8284934918E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.360	-2.1280754936E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.400	-2.0570212200E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.800	-4.3058889787E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	-4.5510262225E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	-8.3048297665E-02	-8.3048297655E-02	-8.3010345775E-02	-3.7951880813E-05
0.660	3.2898128666E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.770	1.1368683772E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.110	3.6379788071E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.567	-6.6080474426E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.517	-2.0605739337E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.900	-1.7834622668E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.950	3.2684965845E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.100	-6.8500000001E-02	-6.8500000000E-02	-6.8483535264E-02	-1.6464735722E-05
0.200	1.4708234630E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.250	1.1795009414E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.120	1.7053025658E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.360	-3.0411229091E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.400	-5.1159076975E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.800	9.8054897535E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	-2.6432189770E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	-3.1323911603E-02	-3.1323911597E-02	-3.1311990722E-02	-1.1920874840E-05
0.660	1.0089706848E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.770	3.8653524825E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.110	-3.1263880373E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.567	-3.6806113712E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.517	-8.9528384706E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.900	2.8705926525E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.950	-6.8922645369E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.100	-8.9913419743E-02	-8.9913419754E-02	-8.9877010407E-02	-3.6409347103E-05
0.200	1.2619238987E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.250	1.4793499759E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.120	-1.8900436771E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00

0.360	-9.8907548818E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.400	7.8443918028E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.800	1.3926637621E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	-4.8459014579E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	-3.0561725193E-02	-3.0561725212E-02	-3.0549027985E-02	-1.2697226879E-05
0.660	4.0216718844E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.770	7.4038553066E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.110	5.1301185522E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.567	4.4053649617E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.517	4.6895820560E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.900	1.4097167877E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.950	5.2651216720E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.100	4.9374999979E-02	4.9375000000E-02	4.9348429823E-02	2.6570177397E-05
0.200	2.8776980798E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.250	2.9416469260E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.120	7.8159700934E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.360	5.6630256040E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.400	-3.1690206015E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.800	-8.8746787696E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	-8.2707174442E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	-6.3004040209E-02	-6.3004040222E-02	-6.2990203558E-02	-1.3836664039E-05
0.660	1.5251799823E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.770	1.7941204078E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.110	1.3073986338E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.567	2.8421709430E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.517	1.7479351300E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.900	-4.7357673338E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.950	-1.2723155862E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.100	-4.7899364144E-02	-4.7899364184E-02	-4.7883541368E-02	-1.5822815897E-05
0.200	5.0262016771E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.250	1.1516121390E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.120	1.5157652911E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.360	1.7195134205E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00

0.400	1.5241141682E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.800	-2.1884716261E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	3.7623237858E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	2.1245628131E-02	2.1245628125E-02	2.1210364086E-02	3.5264039610E-05
0.660	1.8829382498E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.770	2.1884716261E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.110	1.0061285138E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.567	2.0392576516E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.517	-1.6342482922E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.900	8.1499251792E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.950	-9.7415409073E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.100	-6.0624999931E-02	-6.0625000000E-02	-6.0609759819E-02	-1.5240180744E-05
0.200	1.6527224034E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.250	-7.1764816312E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.120	-2.2325252758E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.360	5.1016968428E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.400	9.8197006082E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.800	-5.2864379541E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	7.1054273576E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	2.7372793202E-02	2.7372793152E-02	2.7344584490E-02	2.8208661149E-05
0.660	-1.3926637621E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.770	1.2207124200E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.110	1.8843593352E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.567	8.8533624876E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.517	3.2585489862E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.900	2.1202595235E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.950	3.6408209780E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.100	1.5403841979E-01	1.5403841975E-01	1.5399639882E-01	4.2020937599E-05
0.200	-1.7507773009E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.250	-1.8630430532E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.120	2.7995383789E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.360	1.2363443602E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.400	3.4106051316E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00

0.800	-2.2183144210E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	-7.3896444519E-13	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.660	4.1143948786E-03	4.1143947419E-03	4.0894802698E-03	2.4914472078E-05
0.660	1.2533973859E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.770	-3.5385028241E-12	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.110	2.6176394385E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.567	2.0520474209E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.517	2.4272139854E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.900	7.2645889304E-11	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00
0.950	1.3233858454E-10	0.0000000000E+00	0.0000000000E+00	0.0000000000E+00

## C.2 Transformada Inversa de Hartley

Dato	DHT Real 48 bits	FIIT Real 48 bits	FHT Entera 15 bits	Re(FIIT) - En(FIIT)
0.100	1.0000000079E-01	9.9999999999E-02	1.0000915555E-01	-9.1555538120E-06
0.200	1.9999999992E-01	2.0000000000E-01	2.0017090365E-01	-1.7090365304E-04
0.250	2.4999999992E-01	2.5000000000E-01	2.5003814814E-01	-3.8148135900E-05
0.120	1.2000000009E-01	1.2000000000E-01	1.2005981628E-01	-5.9816275325E-05
0.360	3.5999999996E-01	3.6000000000E-01	3.5996581927E-01	3.4180731291E-05
0.400	4.0000000018E-01	4.0000000000E-01	4.0003662221E-01	-3.6622211155E-05
0.800	7.999999995E-01	7.999999999E-01	7.9989013337E-01	1.0986662892E-04
0.660	6.599999991E-01	6.599999999E-01	6.5999328593E-01	6.7140654210E-06
0.660	6.599999977E-01	6.600000000E-01	6.5996276742E-01	3.7232580326E-05
0.660	6.599999967E-01	6.600000000E-01	6.5984069338E-01	1.5930661903E-04
0.770	7.699999972E-01	7.700000000E-01	7.6982940153E-01	1.7059846778E-04
0.110	1.1000000012E-01	1.100000000E-01	1.1017181921E-01	-1.7181920373E-04
0.567	5.669999996E-01	5.670000000E-01	5.6688131352E-01	1.1868648016E-04
0.517	5.1700000011E-01	5.170000000E-01	5.1686147649E-01	1.3852351367E-04
0.900	9.0000000021E-01	8.999999999E-01	8.9980773339E-01	1.9226660515E-04
0.950	9.5000000009E-01	9.499999999E-01	9.4967497787E-01	3.2502211798E-04
0.100	1.0000000019E-01	9.999999999E-02	1.0000915555E-01	-9.1555538120E-06
0.200	1.9999999963E-01	2.000000000E-01	2.0017090365E-01	-1.7090365304E-04
0.250	2.4999999964E-01	2.500000000E-01	2.5003814814E-01	-3.8148135900E-05
0.120	1.1999999994E-01	1.200000000E-01	1.2005981628E-01	-5.9816275325E-05
0.360	3.599999982E-01	3.600000000E-01	3.5996581927E-01	3.4180731291E-05
0.400	4.000000001E-01	4.000000000E-01	4.0003662221E-01	-3.6622211155E-05
0.800	7.999999944E-01	7.999999999E-01	7.9989013337E-01	1.0986662892E-04
0.660	6.599999968E-01	6.599999999E-01	6.5999328593E-01	6.7140654210E-06
0.660	6.599999968E-01	6.600000000E-01	6.5996276742E-01	3.7232580326E-05
0.660	6.599999958E-01	6.600000000E-01	6.5984069338E-01	1.5930661903E-04
0.770	7.699999948E-01	7.700000000E-01	7.6982940153E-01	1.7059846778E-04

0.110	1.1000000002E-01	1.1000000000E-01	1.1017181921E-01	-1.7181920373E-04
0.567	5.6700000003E-01	5.6700000000E-01	5.6688131352E-01	1.1868648016E-04
0.517	5.1700000017E-01	5.1700000000E-01	5.1686147649E-01	1.3852351367E-04
0.900	8.999999998E-01	8.999999999E-01	8.9980773339E-01	1.9226660515E-04
0.950	9.5000000015E-01	9.499999999E-01	9.4967497787E-01	3.2502211798E-04
0.100	1.0000000019E-01	9.999999999E-02	1.0000915555E-01	-9.1555538120E-06
0.200	1.999999956E-01	2.000000000E-01	2.0017090365E-01	-1.7090365304E-04
0.250	2.499999957E-01	2.500000000E-01	2.5003814814E-01	-3.8148135900E-05
0.120	1.199999971E-01	1.200000000E-01	1.2005981628E-01	-5.9816275325E-05
0.360	3.599999969E-01	3.600000000E-01	3.5996581927E-01	3.4180731291E-05
0.400	3.999999992E-01	4.000000000E-01	4.0003662221E-01	-3.6622211155E-05
0.800	7.999999940E-01	7.999999999E-01	7.9989013337E-01	1.0986662892E-04
0.660	6.599999963E-01	6.599999999E-01	6.5999328593E-01	6.7140654210E-06
0.660	6.599999963E-01	6.600000000E-01	6.5996276742E-01	3.7232580326E-05
0.660	6.599999975E-01	6.600000000E-01	6.5984069338E-01	1.5930661903E-04
0.770	7.699999962E-01	7.700000000E-01	7.6982940153E-01	1.7059846778E-04
0.110	1.099999987E-01	1.100000000E-01	1.1017181921E-01	-1.7181920373E-04
0.567	5.669999998E-01	5.670000000E-01	5.6688131352E-01	1.1868648016E-04
0.517	5.1700000015E-01	5.170000000E-01	5.1686147649E-01	1.3852351367E-04
0.900	9.0000000013E-01	8.999999999E-01	8.9980773339E-01	1.9226660515E-04
0.950	9.5000000015E-01	9.499999999E-01	9.4967497787E-01	3.2502211798E-04
0.100	1.0000000023E-01	9.999999999E-02	1.0000915555E-01	-9.1555538120E-06
0.200	1.999999953E-01	2.000000000E-01	2.0017090365E-01	-1.7090365304E-04
0.250	2.499999954E-01	2.500000000E-01	2.5003814814E-01	-3.8148135900E-05
0.120	1.199999989E-01	1.200000000E-01	1.2005981628E-01	-5.9816275325E-05
0.360	3.599999981E-01	3.600000000E-01	3.5996581927E-01	3.4180731291E-05
0.400	4.0000000014E-01	4.000000000E-01	4.0003662221E-01	-3.6622211155E-05
0.800	7.999999957E-01	7.999999999E-01	7.9989013337E-01	1.0986662892E-04
0.660	6.599999969E-01	6.599999999E-01	6.5999328593E-01	6.7140654210E-06
0.660	6.599999981E-01	6.600000000E-01	6.5996276742E-01	3.7232580326E-05
0.660	6.599999971E-01	6.600000000E-01	6.5984069338E-01	1.5930661903E-04
0.770	7.699999962E-01	7.700000000E-01	7.6982940153E-01	1.7059846778E-04
0.110	1.1000000009E-01	1.100000000E-01	1.1017181921E-01	-1.7181920373E-04

0.567	5.6700000007E-01	5.6700000000E-01	5.6688131352E-01	1.1868648016E-04
0.517	5.1700000002E-01	5.1700000000E-01	5.1686147649E-01	1.3852351367E-04
0.900	9.0000000011E-01	8.9999999999E-01	8.9980773339E-01	1.9226660515E-04
0.950	9.5000000007E-01	9.4999999999E-01	9.4967497787E-01	3.2502211798E-04
0.100	1.0000000008E-01	9.9999999999E-02	1.0000915555E-01	-9.1555538120E-06
0.200	1.999999956E-01	2.0000000000E-01	2.0017090365E-01	-1.7090365304E-04
0.250	2.499999970E-01	2.5000000000E-01	2.5003814814E-01	-3.8148135900E-05
0.120	1.2000000002E-01	1.2000000000E-01	1.2005981628E-01	-5.9816275325E-05
0.360	3.599999990E-01	3.6000000000E-01	3.5996581927E-01	3.4180731291E-05
0.400	4.0000000003E-01	4.0000000000E-01	4.0003662221E-01	-3.6622211155E-05
0.800	7.999999958E-01	7.9999999999E-01	7.9989013337E-01	1.0986662892E-04
0.660	6.599999987E-01	6.5999999999E-01	6.5999328593E-01	6.7140654210E-06
0.660	6.599999986E-01	6.6000000000E-01	6.5996276742E-01	3.7232580326E-05
0.660	6.599999969E-01	6.6000000000E-01	6.5984069338E-01	1.5930661903E-04
0.770	7.699999977E-01	7.7000000000E-01	7.6982940153E-01	1.7059846778E-04
0.110	1.100000011E-01	1.1000000000E-01	1.1017181921E-01	-1.7181920373E-04
0.567	5.6700000006E-01	5.6700000000E-01	5.6688131352E-01	1.1868648016E-04
0.517	5.1700000021E-01	5.1700000000E-01	5.1686147649E-01	1.3852351367E-04
0.900	9.0000000015E-01	8.9999999999E-01	8.9980773339E-01	1.9226660515E-04
0.950	9.5000000022E-01	9.4999999999E-01	9.4967497787E-01	3.2502211798E-04
0.100	1.0000000027E-01	9.9999999999E-02	1.0000915555E-01	-9.1555538120E-06
0.200	1.999999962E-01	2.0000000000E-01	2.0017090365E-01	-1.7090365304E-04
0.250	2.499999966E-01	2.5000000000E-01	2.5003814814E-01	-3.8148135900E-05
0.120	1.199999989E-01	1.2000000000E-01	1.2005981628E-01	-5.9816275325E-05
0.360	3.599999981E-01	3.6000000000E-01	3.5996581927E-01	3.4180731291E-05
0.400	4.0000000044E-01	4.0000000000E-01	4.0003662221E-01	-3.6622211155E-05
0.800	7.999999965E-01	7.9999999999E-01	7.9989013337E-01	1.0986662892E-04
0.660	6.599999978E-01	6.5999999999E-01	6.5999328593E-01	6.7140654210E-06
0.660	6.599999960E-01	6.6000000000E-01	6.5996276742E-01	3.7232580326E-05
0.660	6.599999968E-01	6.6000000000E-01	6.5984069338E-01	1.5930661903E-04
0.770	7.699999976E-01	7.7000000000E-01	7.6982940153E-01	1.7059846778E-04
0.110	1.100000017E-01	1.1000000000E-01	1.1017181921E-01	-1.7181920373E-04
0.567	5.6700000016E-01	5.6700000000E-01	5.6688131352E-01	1.1868648016E-04

0.517	5.1700000056E-01	5.1700000000E-01	5.1686147649E-01	1.3852351367E-04
0.900	9.0000000020E-01	8.9999999999E-01	8.9980773339E-01	1.9226660515E-04
0.950	9.5000000053E-01	9.4999999999E-01	9.4967497787E-01	3.2502211798E-04
0.100	1.0000000036E-01	9.9999999999E-02	1.0000915555E-01	-9.1555538120E-06
0.200	1.9999999980E-01	2.0000000000E-01	2.0017090365E-01	-1.7090365304E-04
0.250	2.4999999973E-01	2.5000000000E-01	2.5003814814E-01	-3.8148135900E-05
0.120	1.2000000012E-01	1.2000000000E-01	1.2005981628E-01	-5.9816275325E-05
0.360	3.6000000016E-01	3.6000000000E-01	3.5996581927E-01	3.4180731291E-05
0.400	3.999999995E-01	4.0000000000E-01	4.0003662221E-01	-3.6622211155E-05
0.800	7.999999983E-01	7.9999999999E-01	7.9989013337E-01	1.0986662892E-04
0.660	6.599999999E-01	6.5999999999E-01	6.5999328593E-01	6.7140654210E-06
0.660	6.599999993E-01	6.6000000000E-01	6.5996276742E-01	3.7232580326E-05
0.660	6.600000003E-01	6.6000000000E-01	6.5984069338E-01	1.5930661903E-04
0.770	7.699999982E-01	7.7000000000E-01	7.6982940153E-01	1.7059846778E-04
0.110	1.1000000035E-01	1.1000000000E-01	1.1017181921E-01	-1.7181920373E-04
0.567	5.6700000040E-01	5.6700000000E-01	5.6688131352E-01	1.1868648016E-04
0.517	5.1700000040E-01	5.1700000000E-01	5.1686147649E-01	1.3852351367E-04
0.900	9.0000000034E-01	8.9999999999E-01	8.9980773339E-01	1.9226660515E-04
0.950	9.5000000045E-01	9.4999999999E-01	9.4967497787E-01	3.2502211798E-04
0.100	1.0000000033E-01	9.9999999999E-02	1.0000915555E-01	-9.1555538120E-06
0.200	1.999999998E-01	2.0000000000E-01	2.0017090365E-01	-1.7090365304E-04
0.250	2.499999995E-01	2.5000000000E-01	2.5003814814E-01	-3.8148135900E-05
0.120	1.2000000013E-01	1.2000000000E-01	1.2005981628E-01	-5.9816275325E-05
0.360	3.6000000031E-01	3.6000000000E-01	3.5996581927E-01	3.4180731291E-05
0.400	4.0000000050E-01	4.0000000000E-01	4.0003662221E-01	-3.6622211155E-05
0.800	7.999999979E-01	7.9999999999E-01	7.9989013337E-01	1.0986662892E-04
0.660	6.600000005E-01	6.5999999999E-01	6.5999328593E-01	6.7140654210E-06
0.660	6.6000000031E-01	6.6000000000E-01	6.5996276742E-01	3.7232580326E-05
0.660	6.6000000030E-01	6.6000000000E-01	6.5984069338E-01	1.5930661903E-04
0.770	7.7000000014E-01	7.7000000000E-01	7.6982940153E-01	1.7059846778E-04
0.110	1.1000000034E-01	1.1000000000E-01	1.1017181921E-01	-1.7181920373E-04
0.567	5.6700000058E-01	5.6700000000E-01	5.6688131352E-01	1.1868648016E-04
0.517	5.1700000053E-01	5.1700000000E-01	5.1686147649E-01	1.3852351367E-04

0.900	9.0000000062E-01	8.9999999999E-01	8.9980773339E-01	1.9226660515E-04
0.950	9.5000000064E-01	9.4999999999E-01	9.4967497787E-01	3.2502211798E-04

# Apéndice **D**

; Félix Cuéllar [Hawk] 496602 ISE80

.TITLE 'FHT radix-2 de1 28 puntos'

.ASECT "VECTORI",0000H

RESET	B	INIT
INT0	B	ISR0
INT1	B	ISR1
INT2	B	ISR2

.SPACE 0100H

TINT	B	TIME
RINT	B	RCV
XINT	B	XMT
USER	B	PROC

; Tablas de Cosenos y Senos (15 bits)

.ASECT "ROMDATA",0020H

COS	.WORD	+7FFFH
	.WORD	+7FD7H
	.WORD	+7F61H
	.WORD	+7E9CH
	.WORD	+7D89H

.WORD	+7C29H
.WORD	+7A7CH
.WORD	+7883H
.WORD	+7640H
.WORD	+73B5H
.WORD	+70E1H
.WORD	+6DC9H
.WORD	+6A6CH
.WORD	+66CEH
.WORD	+62F1H
.WORD	+5ED6H
.WORD	+5A81H
.WORD	+55F4H
.WORD	+5133H
.WORD	+4C3FH
.WORD	+471CH
.WORD	+41CDH
.WORD	+3C56H
.WORD	+36B9H
.WORD	+30FBH
.WORD	+2B1EH
.WORD	+2527H
.WORD	+1F19H
.WORD	+18F8H
.WORD	+12C7H
.WORD	+0C8BH
.WORD	+0647H
.WORD	0000H
.WORD	-0647H
.WORD	-0C8BH
.WORD	-12C7H
.WORD	-18F8H
.WORD	-1F19H
.WORD	-2527H
.WORD	-2B1EH
.WORD	-30FBH

.WORD -36B9H  
.WORD -3C56H  
.WORD -41CDH  
.WORD -471CH  
.WORD -4C3FH  
.WORD -5133H  
.WORD -55F4H  
.WORD -5A81H  
.WORD -5ED6H  
.WORD -62F1H  
.WORD -66CEH  
.WORD -6A6CH  
.WORD -6DC9H  
.WORD -70E1H  
.WORD -73B5H  
.WORD -7640H  
.WORD -7883H  
.WORD -7A7CH  
.WORD -7C29H  
.WORD -7D89H  
.WORD -7E9CH  
.WORD -7F61H  
.WORD -7FD7H

SEN

.WORD 0000H  
.WORD +0647H  
.WORD +0C8BH  
.WORD +12C7H  
.WORD +18F8H  
.WORD +1F19H  
.WORD +2527H  
.WORD +2B1EH  
.WORD +30FBH  
.WORD +36B9H  
.WORD +3C56H  
.WORD +41CDH

.WORD +471CH  
.WORD +4C3FH  
.WORD +5133H  
.WORD +55F4H  
.WORD +5A81H  
.WORD +5ED6H  
.WORD +62F1H  
.WORD +66CEH  
.WORD +6A6CH  
.WORD +6DC9H  
.WORD +70E1H  
.WORD +73B5H  
.WORD +7640H  
.WORD +7883H  
.WORD +7A7CH  
.WORD +7C29H  
.WORD +7D89H  
.WORD +7E9CH  
.WORD +7F61H  
.WORD +7FD7H  
.WORD +7FFFH  
.WORD +7FD7H  
.WORD +7F61H  
.WORD +7E9CH  
.WORD +7D89H  
.WORD +7C29H  
.WORD +7A7CH  
.WORD +7883H  
.WORD +7640H  
.WORD +73B5H  
.WORD +70E1H  
.WORD +6DC9H  
.WORD +6A6CH  
.WORD +66CEH  
.WORD +62F1H  
.WORD +5ED6H

```

.WORD +5A81H
.WORD +55F4H
.WORD +5133H
.WORD +4C3FH
.WORD +471CH
.WORD +41CDH
.WORD +3C56H
.WORD +36B9H
.WORD +30FBH
.WORD +2B1EH
.WORD +2527H
.WORD +1F19H
.WORD +18F8H
.WORD +12C7H
.WORD +0C8BH
.WORD +0647H

; Vectores de Interrupción
.ASECT "ROMCODE",00A0H

ISR0: RET
ISR1: RET
ISR2: RET
TIME: RET

; Interrupción del puerto serial
RCV: SST TST0 ;Almacenar registro de estado
LARP AR6 ;Cambio de apuntador activo
OUT *,PA8 ;Sacar dato transformado
LAR AR7,DDR ;Cargar dato puerto serial
SAR AR7,*BR0+ ;Almacenar dato en memoria

```

```

LST TST0 ;Restablecer registro de estado
RET ;Regreso de la interrupción

XMT: RET

PROC: RET

; Inicialización
INIT: DINT ;Deshabilitar Interrupciones
ROVM ;Modo de Sobreflujo
CNFD ;Configurar B0 como datos
SSXM ;Configurar signo extendido
LDPK 00H ;Página de datos cero
LALK 00000H
SACL GREG ;Configurar registro (Global)
LALK 0FFFFH
SACL TIM ;Configurar registro (Timer)
SACL PRD ;Configurar registro (Periodo)
LALK 0FFD0H
SACL IMR ;Configurar registro (INT)
FORT 0
STXM ;Configurar transmisión
SFSM ;Configurar sincrona
SPM 00H ;Configurar corrimiento reg P
SXF ;Habilitar bandera externa
LARK AR0,0080H
LARK AR6,0200H
LALK 0200H
SACL BASE
EINT ;Habilitar interrupciones

; Ciclo de espera para el llenado del buffer
CICLO: SAR AR6,INDT
LAC INDT

```

```

ANDK 00FFH
BNZ  CICLO
LAC  INDT
SACL  BASE
ADLK 0100H
ORK  0200H
ANDK 0300H
SACL  INDT
LAR  AR6,INDT

```

; Preparación Mariposas

```

ZALS  BASE
SBLK  0080H
SACL  BASE
HI0:  ZALS  BASE
      ADLK  0080H
      SACL  BASE

```

; Primera Mariposa

```

LARA  R1,BASE
LARA  R2,BASE
LARP  AR2
MAR   *+,AR1
LARK  AR5,003FH ;Número de ciclos
HI1:  LAC  *,15,AR2
      ADD *,15,AR1
      SACH *+,AR2 ; $P_{m+1} = (P_m + W Q_m) / 2$ 
      SUBH *
      SACH *+
      MAR *+,AR1 ; $Q_{m+1} = (P_m - W Q_m) / 2$ 
      MAR *+,AR5
      BANZ HI1,*-,AR1 ;Comparación ciclos

```

; Segunda Mariposa

```

LAR  AR1,BASE
LAR  AR2,BASE
LARP AR2
ADRK 0002H
LARP AR1
LARK AR5,001FH ;Número de ciclos
HI2: LAC  *,15,AR2
      ADD *,15,AR1
      SACH *+,AR2 ; $P_{m+1} = (P_m + W Q_m) / 2$ 
      SUBH *
      SACH *+,AR1 ; $Q_{m+1} = (P_m - W Q_m) / 2$ 
      LAC  *,15,AR2
      ADD *,15,AR1
      SACH *+,AR2 ; $P_{m+1} = (P_m + W Q_m) / 2$ 
      SUBH *
      SACH * ; $Q_{m+1} = (P_m - W Q_m) / 2$ 
      ADRK 0003H
      LARP AR1
      ADRK 0003H
      LARP AR5
      BANZ HI2,*-,AR1 ;Comparación ciclos

```

; Tercera Mariposa

```

LAR  AR1,BASE
LAR  AR2,BASE
LAR  AR3,BASE
LARP AR2
ADRK 0004H
LARP AR3
ADRK 0007H
LARP AR1
LARK AR5,000FH ;Número de ciclos
LALK 0030H
TBLR VCYS

```

HI3: LALK 0050H  
 TBLR VCOS  
 LALK 0090H  
 TBLR VSEN  
 LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*\*+,AR2 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 SUBH \*  
 SACH \*\*+, ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 LAC \*  
 SACL TEMP  
 LT VCYS  
 MPY \*,AR3 ; $W1 = \text{Cos}(2\pi m/128) * Q_m$   
 LTP VCYS  
 MPY \*,AR1 ; $W2 = \text{Sen}(2\pi m/128) * Q_m$   
 APAC ; $W = W1 + W2$   
 SACH TEMPH  
 SACL TEMPL  
 NEG  
 ADD \*,15,AR2  
 SACH \*\*+,AR1 ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ZALS TEMPL  
 ADDH TEMPH  
 ADD \*,15  
 SACH \*\*+, ; $P_{m+1} = (P_m + W Q_m) / 2$   
 LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*\*+,AR2 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 SUBH \*  
 SACH \*\*+, ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 LT VCOS  
 MPY \* ; $W1 = \text{Cos}(2\pi m/128) * Q_m$   
 ZALH TEMP  
 SACH \*  
 LTP VSEN  
 MPY \*,AR1 ; $W2 = \text{Sen}(2\pi m/128) * Q_m$

APAC ; $W = W1 + W2$   
 SACH TEMPH  
 SACL TEMPL  
 NEG  
 ADD \*,15,AR2  
 SACH \*,AR1 ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ZALS TEMPL  
 ADDH TEMPH  
 ADD \*,15  
 SACH \*,AR3 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 ADRK 0008H  
 LARP AR2  
 ADRK 0005H  
 LARP AR1  
 ADRK 0005H  
 LARP AR5  
 BANZ HI3,\*,AR1 ;Comparación ciclos

Cuarta Mariposa

HI4: LAR AR1.BASE  
 LAR AR2.BASE  
 LARP AR2  
 ADRK 0008H  
 LARP AR1  
 LARK AR5,0007H ;Número de ciclos  
 LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*,AR2 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 SUBH \*  
 SACH \* ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ADRK 0004H  
 LARP AR1  
 ADRK 0004H  
 LAC \*,15,AR2  
 ADD \*,15,AR1

HI4F:

SACH \* ,AR2 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 \*  
 SACH \* ,AR1 ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ADRK 03H  
 SAR AR1,TEMP  
 LAR AR3,TEMP  
 SBRK 06H  
 LARP AR2  
 ADRK 03H  
 SAR AR2,TEMP  
 LAR AR4,TEMP  
 SBRK 06H  
 LALK 0028H  
 SACL INDCS1  
 LALK 0058H  
 SACL INDCS2  
 SAR AR5,TEMPS  
 PSHD TEMPS  
 LARK AR5,0002H ;Número de ciclos  
 ZALS INDCS1  
 TBLR VCOS  
 ADLK 0040H  
 TBLR VSEN  
 SBLK 0038H  
 SACL INDCS1  
 LT VCOS  
 MPY \* ,AR4 ; $W1 = \text{Cos}(2\pi n / 128) * Q_m$   
 LTP VSEN  
 MPY \* ,AR2 ; $W2 = \text{Sen}(2\pi n / 128) * Q_m$   
 APAC ; $W = W1 + W2$   
 SACH TEMPH  
 SACL TEMPL  
 NEG  
 LT \* ,AR1  
 ADD \* ,15,AR2  
 SACH \* + ,AR1 ; $Q_{m+1} = (P_m - W Q_m) / 2$

ZALS TEMPL  
 ADDH TEMPH  
 ADD \* ,15  
 SACH \* + ,AR4 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 ZALS INDCS2  
 TBLR VCOS  
 ADLK 0040H  
 TBLR VSEN  
 SBLK 0048H  
 SACL INDCS2  
 MPY VSEN ; $W1 = \text{Sen}(2\pi n / 128) * Q_m$   
 LTP VCOS  
 MPY \* ,AR3 ; $W2 = \text{Cos}(2\pi n / 128) * Q_m$   
 APAC ; $W = W1 + W2$   
 SACH TEMPH  
 SACL TEMPL  
 NEG  
 ADD \* ,15,AR4  
 SACH \* - ,AR3 ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ZALS TEMPL  
 ADDH TEMPH  
 ADD \* ,15  
 SACH \* - ,AR5 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 BANZ HI4F, \* - ,AR2 ;Comparación ciclos  
 POPD TEMPS  
 LAR AR5,TEMPS  
 ADRK 000CH  
 LARP AR1  
 ADRK 000CH  
 LARP AR5  
 BANZ HI4, \* - ,AR1 ;Comparación ciclos

; Quinta Mariposa

LAR AR1,BASE  
 LAR AR2,BASE

HI5

LARP AR2  
 ADRK 0010H  
 LARP AR1  
 LARK AR5,0003H ;Número de ciclos  
 :LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*,AR2 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 SUBH \*  
 SACH \* ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ADRK 0008H  
 LARP AR1  
 ADRK 0008H  
 LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*,AR2 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 SUBH \*  
 SACH \*,AR1 ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ADRK 07H  
 SAR AR1,TEMP  
 LAR AR3,TEMP  
 SBRK 0EH  
 LARP AR2  
 ADRK 07H  
 SAR AR2,TEMP  
 LAR AR4,TEMP  
 SBRK 0EH  
 LALK 0024H  
 SACL INDCS1  
 LALK 005CH  
 SACL INDCS2  
 SAR AR5,TEMPS  
 PSHD TEMPS  
 LARK AR5,0006H ;Número de ciclos  
 ZALS INDCS1  
 TBLR VCOS  
 ADLK 0040H

HI5F:

TBLR VSEN  
 SBLK 003CH  
 SACL INDCS1  
 LT VCOS  
 MPY \*,AR4 ; $W1 = \text{Cos}(2\pi m/128) * Q_m$   
 LTP VSEN  
 MPY \*,AR2 ; $W2 = \text{Sen}(2\pi m/128) * Q_m$   
 APAC ; $W = W1 + W2$   
 SACH TEMPH  
 SACL TEMPL  
 NEG  
 LT \*,AR1  
 ADD \*,15,AR2  
 SACH \*,AR1 ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ZALS TEMPL  
 ADDH TEMPH  
 ADD \*,15  
 SACH \*,AR4 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 ZALS INDCS2  
 TBLR VCOS  
 ADLK 0040H  
 TBLR VSEN  
 SBLK 0044H  
 SACL INDCS2  
 MPY VSEN ; $W1 = \text{Sen}(2\pi m/128) * Q_m$   
 LTP VCOS  
 MPY \*,AR3 ; $W2 = \text{Cos}(2\pi m/128) * Q_m$   
 APAC ; $W = W1 + W2$   
 SACH TEMPH  
 SACL TEMPL  
 NEG  
 ADD \*,15,AR4  
 SACH \*,AR3 ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ZALS TEMPL  
 ADDH TEMPH  
 ADD \*,15

	SACH	*-,AR5	; $P_{m+1} = (P_m + W Q_m) / 2$
	BANZ	HI5F,*-,AR2	; Comparación ciclos
	POPD	TEMPS	
	LAR	AR5,TEMPS	
	ADRK	0018H	
	LARP	AR1	
	ADRK	0018H	
	LARP	AR5	
	BANZ	HI5,*-,AR1	; Comparación ciclos
; Sexta Mariposa			
	LAR	AR1,BASE	
	LAR	AR2,BASE	
	LARP	AR2	
	ADRK	0020H	
	LARP	AR1	
	LARK	AR5,0001H	; Número de ciclos
HI6:	LAC	*,15,AR2	
	ADD	*,15,AR1	
	SACH	*-,AR2	; $P_{m+1} = (P_m + W Q_m) / 2$
	SUBH	*	
	SACH	*	; $Q_{m+1} = (P_m - W Q_m) / 2$
	ADRK	0010H	
	LARP	AR1	
	ADRK	0010H	
	LAC	*,15,AR2	
	ADD	*,15,AR1	
	SACH	*-,AR2	; $P_{m+1} = (P_m + W Q_m) / 2$
	SUBH	*	
	SACH	*-,AR1	; $Q_{m+1} = (P_m - W Q_m) / 2$
	ADRK	0FH	
	SAR	AR1,TEMP	
	LAR	AR3,TEMP	
	SBRK	1EH	
	LARP	AR2	

	ADRK	0FH	
	SAR	AR2,TEMP	
	LAR	AR4,TEMP	
	SBRK	1EH	
	LALK	0022H	
	SACL	INDCS1	
	LALK	005EH	
	SACL	INDCS2	
	SAR	AR5,TEMPS	
	PSHD	TEMPS	
	LARK	AR5,000EH	; Número de ciclos
HI6F:	ZALS	INDCS1	
	TBLR	VCOS	
	ADLK	0040H	
	TBLR	VSEN	
	SBLK	003EH	
	SACL	INDCS1	
	LT	VCOS	
	MPY	*-,AR4	; $W1 = \cos(2\pi n/128) * Q_m$
	LTP	VSEN	
	MPY	*-,AR2	; $W2 = \sin(2\pi n/128) * Q_m$
	APAC		; $W = W1 + W2$
	SACH	TEMPH	
	SACL	TEMPL	
	NEG		
	LT	*-,AR1	
	ADD	*,15,AR2	
	SACH	*+,AR1	; $Q_{m+1} = (P_m - W Q_m) / 2$
	ZALS	TEMPL	
	ADDH	TEMPH	
	ADD	*,15	
	SACH	*+,AR4	; $P_{m+1} = (P_m + W Q_m) / 2$
	ZALS	INDCS2	
	TBLR	VCOS	
	ADLK	0040H	
	TBLR	VSEN	

SBLK 0042H  
 SACL INDCS2  
 MPY VSEN ;W1 = Sen(2πm/128) \* Q<sub>m</sub>  
 LTP VCOS  
 MPY \*,AR3 ;W2 = Cos(2πm/128) \* Q<sub>m</sub>  
 APAC ;W = W1 + W2  
 SACH TEMPH  
 SACL TEMPL  
 NEG  
 ADD \*,15,AR4  
 SACH \*-,AR3 ;Q<sub>m+1</sub> = (P<sub>m</sub> - W Q<sub>m</sub>) / 2  
 ZALS TEMPL  
 ADDH TEMPH  
 ADD \*,15  
 SACH \*-,AR5 ;P<sub>m+1</sub> = (P<sub>m</sub> + W Q<sub>m</sub>) / 2  
 BANZ HI6F,\*-,AR2 ;Comparación ciclos  
 POPD TEMPS  
 LAR AR5,TEMPS  
 ADRK 0030H  
 LARP AR1  
 ADRK 0030H  
 LARP AR5  
 BANZ HI6,\*-,AR1 ;Comparación ciclos

; Séptima Mariposa

HI7:

LAR AR1,BASE  
 LAR AR2,BASE  
 LARP AR2  
 ADRK 0040H  
 LARP AR1  
 LARK AR5,0000H ;Número de ciclos  
 LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*,AR2 ;P<sub>m+1</sub> = (P<sub>m</sub> + W Q<sub>m</sub>) / 2  
 SUBH \*

HI7F:

SACH \* ;Q<sub>m+1</sub> = (P<sub>m</sub> - W Q<sub>m</sub>) / 2  
 ADRK 0020H  
 LARP AR1  
 ADRK 0020H  
 LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*,AR2 ;P<sub>m+1</sub> = (P<sub>m</sub> + W Q<sub>m</sub>) / 2  
 SUBH \*  
 SACH \*-,AR1 ;Q<sub>m+1</sub> = (P<sub>m</sub> - W Q<sub>m</sub>) / 2  
 ADRK 1FH  
 SAR AR1,TEMP  
 LAR AR3,TEMP  
 SBRK 3EH  
 LARP AR2  
 ADRK 1FH  
 SAR AR2,TEMP  
 LAR AR4,TEMP  
 SBRK 3EH  
 LALK 0021H  
 SACL INDCS1  
 LALK 005:11  
 SACL INDCS2  
 SAR AR5,TEMPS  
 PSHD TEMPS  
 LARK AR5,001EH ;Número de ciclos  
 ZALS INDCS1  
 TBLR VCOS  
 ADLK 0040H  
 TBLR VSEN  
 SBLK 003FH  
 SACL INDCS1  
 LT VCOS  
 MPY \*,AR4 ;W1 = Cos(2πm/128) \* Q<sub>m</sub>  
 LTP VSEN  
 MPY \*,AR2 ;W2 = Sen(2πm/128) \* Q<sub>m</sub>  
 APAC ;W = W1 + W2



SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar 50H con 05H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar 30H con 06H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar 70H con 07H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+  
MAR \*+,AR2  
SBRK 0028H  
Cambiar 08H con 08H  
Cambiar 48H con 09H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar 28H con 0AH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar 68H con 0BH

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0050H  
Cambiar 18H con 0CH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar 58H con 0DH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar 38H con 0EH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar 78H con 0FH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+  
MAR \*+,AR2  
SBRK 0034H  
Cambiar 04H con 10H  
Cambiar 44H con 11H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2

SBRK 0020H  
 Cambiar 24H con 12H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*,AR2  
 ADRK 0040H  
 Cambiar 64H con 13H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*,AR2  
 SBRK 0010H  
 Cambiar 14H con 14H  
 Cambiar 54H con 15H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*,AR2  
 SBRK 0020H  
 Cambiar 34H con 16H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*,AR2  
 ADRK 0040H  
 Cambiar 74H con 17H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*,AR2  
 SBRK 0028H  
 Cambiar 0CH con 18H  
 Cambiar 4CH con 19H

ZALS \*.AR1  
 ADDH \*.AR2  
 SACH \*,AR1  
 SACL \*,AR2  
 SBRK 0020H  
 Cambiar 2CH con 1AH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*,AR2  
 ADRK 0040H  
 Cambiar 6CH con 1BH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*,+2  
 SBRK 0010H  
 Cambiar 1CH con 1CH  
 Cambiar 5CH con 1DH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*,AR2  
 SBRK 0020H  
 Cambiar 3CH con 1EH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*,AR2  
 ADRK 0040H  
 Cambiar 7CH con 1FH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+

MAR \*\*+,2  
SBRK 003AH  
; Cambiar 02H con 20H ;  
; Cambiar 42H con 21H ;  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+  
MAR \*\*+,2  
ADRK 0020H  
; Cambiar 22H con 22H ;  
; Cambiar 62H con 23H ;  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+  
MAR \*\*+,2  
SBRK 0010H  
; Cambiar 12H con 24H ;  
; Cambiar 52H con 25H ;  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0020H  
; Cambiar 32H con 26H ;  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H  
; Cambiar 72H con 27H ;  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+

MAR \*\*+,2  
SBRK 0028H  
; Cambiar 0AH con 28H ;  
; Cambiar 4AH con 29H ;  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+  
MAR \*\*+,2  
ADRK 0020H  
; Cambiar 2AH con 2AH ;  
; Cambiar 6AH con 2BH ;  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+  
MAR \*\*+,2  
SBRK 0010H  
; Cambiar 1AH con 2CH ;  
; Cambiar 5AH con 2DH ;  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0020H  
; Cambiar 3AH con 2EH ;  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H  
; Cambiar 7AH con 2FH ;  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+

MAR       \*,+2  
 SBRK       0034H  
 Cambiar 06H con 30H  
 Cambiar 46H con 31H  
 ZALS       \*,AR1  
 ADDH       \*,AR2  
 SACH       \*,AR1  
 SACL       \*+  
 MAR       \*,+2  
 ADRK       0020H  
 Cambiar 26H con 32H  
 Cambiar 66H con 33H  
 ZALS       \*,AR1  
 ADDH       \*,AR2  
 SACH       \*,AR1  
 SACL       \*+  
 MAR       \*,+2  
 SBRK       0010H  
 Cambiar 16H con 34H  
 Cambiar 56H con 35H  
 ZALS       \*,AR1  
 ADDH       \*,AR2  
 SACH       \*,AR1  
 SACL       \*+  
 MAR       \*,+2  
 ADRK       0020H  
 Cambiar 36H con 36H  
 Cambiar 76H con 37H  
 ZALS       \*,AR1  
 ADDH       \*,AR2  
 SACH       \*,AR1  
 SACL       \*+  
 MAR       \*,+2  
 SBRK       0028H  
 Cambiar 0EH con 38H  
 Cambiar 4EH con 39H

ZALS       \*,AR1  
 ADDH       \*,AR2  
 SACH       \*,AR1  
 SACL       \*+  
 MAR       \*,+2  
 ADRK       0020H  
 Cambiar 2EH con 3AH  
 Cambiar 6EH con 3BH  
 ZALS       \*,AR1  
 ADDH       \*,AR2  
 SACH       \*,AR1  
 SACL       \*+  
 MAR       \*,+2  
 SBRK       0010H  
 Cambiar 1EH con 3CH  
 Cambiar 5EH con 3DH  
 ZALS       \*,AR1  
 ADDH       \*,AR2  
 SACH       \*,AR1  
 SACL       \*+  
 MAR       \*,+2  
 ADRK       0020H  
 Cambiar 3EH con 3EH  
 Cambiar 7EH con 3FH  
 ZALS       \*,AR1  
 ADDH       \*,AR2  
 SACH       \*,AR1  
 SACL       \*  
 ADRK       0004H  
 LARP       AR2  
 SBRK       001DH  
 Cambiar 01H con 40H  
 Cambiar 41H con 41H  
 Cambiar 21H con 42H  
 Cambiar 61H con 43H  
 ZALS       \*,AR1

ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,2  
 SBRK 0010H  
 Cambiar 11H con 44H  
 Cambiar 51H con 45H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,2  
 ADRK 0020H  
 Cambiar 31H con 46H  
 Cambiar 71H con 47H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 SBRK 0008H  
 Cambiar 09H con 48H  
 Cambiar 49H con 49H  
 Cambiar 29H con 4AH  
 Cambiar 69H con 4BH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,2  
 SBRK 0010H  
 Cambiar 19H con 4CH  
 Cambiar 59H con 4DH  
 ZALS \*,AR1  
 ADDH \*,AR2

SACH \*,AR1  
 SACL \*+  
 MAR \*+,2  
 ADRK 0020H  
 Cambiar 79H con 4EH  
 Cambiar 79H con 4FH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 SBRK 0014H  
 Cambiar 05H con 50H  
 Cambiar 45H con 51H  
 Cambiar 25H con 52H  
 Cambiar 65H con 53H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 ADRK 0010H  
 Cambiar 15H con 54H  
 Cambiar 55H con 55H  
 Cambiar 35H con 56H  
 Cambiar 75H con 57H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 SBRK 0008H  
 Cambiar 0DH con 58H

;  
;  
;  
; Cambiar 4DH con 59H  
;  
; Cambiar 2DH con 5AH  
;  
; Cambiar 6DH con 5BH  
; ZALS \*.AR1  
; ADDH \*.AR2  
; SACH \*.AR1  
; SACL \*  
; ADRK 0004H  
; LARP AR2  
; ADRK 0010H  
;  
; Cambiar 1DH con 5CH  
;  
; Cambiar 5DH con 5DH  
;  
; Cambiar 3DH con 5EH  
;  
; Cambiar 7DH con 5FH  
; ZALS \*.AR1  
; ADDH \*.AR2  
; SACH \*.AR1  
; SACL \*  
; ADRK 0008H  
; LARP AR2  
; SBRK 000AH  
;  
; Cambiar 03H con 60H  
;  
; Cambiar 43H con 61H  
;  
; Cambiar 23H con 62H  
;  
; Cambiar 63H con 63H  
; Cambiar 13H con 64H  
; Cambiar 53H con 65H  
; Cambiar 33H con 66H  
; Cambiar 73H con 67H  
; ZALS \*.AR1  
; ADDH \*.AR2  
; SACH \*.AR1  
; SACL \*  
; ADRK 0008H  
; LARP AR2  
; ADRK 0008H

;  
; Cambiar 0BH con 68H  
; Cambiar 4BH con 69H  
; Cambiar 2BH con 6AH  
; Cambiar 6BH con 6BH  
; Cambiar 1BH con 6CH  
; Cambiar 5BH con 6DH  
; Cambiar 3BH con 6EH  
; Cambiar 7BH con 6FH  
; ZALS \*.AR1  
; ADDH \*.AR2  
; SACH \*.AR1  
; SACL \*  
; ADRK 0012H  
; LARP AR2  
; ADRK 0045H  
; Cambiar 07H con 70H  
; Cambiar 47H con 71H  
; Cambiar 27H con 72H  
; Cambiar 67H con 73H  
; Cambiar 17H con 74H  
; Cambiar 57H con 75H  
; Cambiar 37H con 76H  
; Cambiar 77H con 77H  
; Cambiar 0FH con 78H  
; Cambiar 4FH con 79H  
; Cambiar 2FH con 7AH  
; Cambiar 6FH con 7BH  
; Cambiar 1FH con 7CH  
; Cambiar 5FH con 7DH  
; Cambiar 3FH con 7EH  
; Cambiar 7FH con 7FH  
;  
; Cambiar 80H con 80H  
; Cambiar C0H con 81H  
; ZALS \*.AR1  
; ADDH \*.AR2

SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0020H  
Cambiar A0H con 82H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H  
Cambiar E0H con 83H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0050H  
Cambiar 90H con 84H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H  
Cambiar D0H con 85H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0020H  
Cambiar B0H con 86H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H  
Cambiar F0H con 87H  
ZALS \*,AR1  
ADDH \*,AR2

SACH \*,AR1  
SACL \*\*+  
MAR \*\*+,2  
SBRK 0028H  
Cambiar 88H con 88H  
Cambiar C8H con 89H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0020H  
Cambiar A8H con 8AH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H  
Cambiar E8H con 8BH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0050H  
Cambiar 98H con 8CH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H  
Cambiar D8H con 8DH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0020H  
Cambiar B8H con 8EH

ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+,AR2  
 ADRK 0040H  
 Cambiar F8H con 8FH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,AR2  
 SBRK 0034H  
 Cambiar 84H con 90H  
 Cambiar C4H con 91H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+,AR2  
 SBRK 0020H  
 Cambiar A4H con 92H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+,AR2  
 ADRK 0040H  
 Cambiar E4H con 93H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,AR2  
 SBRK 0010H  
 Cambiar 94H con 94H  
 Cambiar D4H con 95H  
 ZALS \*,AR1  
 ADDH \*,AR2

SACH \*,AR1  
 SACL \*+,AR2  
 SBRK 0020H  
 Cambiar B4H con 96H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+,AR2  
 ADRK 0040H  
 Cambiar F4H con 97H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,AR2  
 SBRK 0028H  
 Cambiar 8CH con 98H  
 Cambiar CCH con 99H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+,AR2  
 SBRK 0020H  
 Cambiar ACH con 9AH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+,AR2  
 ADRK 0040H  
 Cambiar ECH con 9BH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,2  
 SBRK 0010H

; Cambiar 9CH con 9CH ;

; Cambiar DCH con 9DH ;

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0020H

; Cambiar BCH con 9EH ;

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H

; Cambiar FCH con 9FH ;

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+  
MAR \*\*+,2  
SBRK 003AH

; Cambiar 82H con A0H ;

; Cambiar C2H con A1H ;

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+  
MAR \*\*+,2  
ADRK 0020H

; Cambiar A2H con A2H ;

; Cambiar E2H con A3H ;

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+  
MAR \*\*+,2  
SBRK 0010H

Cambiar 92H con A4H

Cambiar D2H con A5H

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0020H

; Cambiar B2H con A6H ;

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H

; Cambiar F2H con A7H ;

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+  
MAR \*\*+,2  
SBRK 0028H

; Cambiar 8AH con A8H ;

; Cambiar CAH con A9H ;

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+  
MAR \*\*+,2  
ADRK 0020H

; Cambiar AAH con AAH ;

; Cambiar EAH con ABH ;

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+  
MAR \*\*+,2  
SBRK 0010H

Cambiar 9AH con ACH

Cambiar DAH con ADH

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*\*+,AR2

SBRK 0020H

Cambiar BAH con AEH

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*\*+,AR2

ADRK 0040H

Cambiar FAH con AFH

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*\*+

MAR \*\*+,2

SBRK 0034H

Cambiar 86H con B0H

Cambiar C6H con B1H

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*\*+

MAR \*\*+,2

ADRK 0020H

Cambiar A6H con B2H

Cambiar E6H con B3H

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*\*+

MAR \*\*+,2

SBRK 0010H

Cambiar 96H con B4H

Cambiar D6H con B5H

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*\*+

MAR \*\*+,2

ADRK 0020H

Cambiar B6H con B6H

Cambiar F6H con B7H

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*\*+

MAR \*\*+,2

SBRK 0028H

Cambiar 8EH con B8H

Cambiar CEH con B9H

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*\*+

MAR \*\*+,2

ADRK 0020H

Cambiar AEH con BAH

Cambiar EEH con BBH

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*\*+

MAR \*\*+,2

SBRK 0010H

Cambiar 9EH con BCH

Cambiar DEH con BDH

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1  
 SACL \*\*  
 MAR \*\*+,2  
 ADRK 0020H  
 ;  
 Cambiar BEH con BEH ;  
 ;  
 Cambiar FEH con BFH ;  
 ZALS \*,AR1 ;  
 ADDH \*,AR2 ;  
 SACH \*,AR1 ;  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 SBRK 001DH  
 ;  
 Cambiar 81H con C0H ;  
 ;  
 Cambiar C1H con C1H ;  
 ;  
 Cambiar A1H con C2H ;  
 ;  
 Cambiar E1H con C3H ;  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*\*  
 MAR \*\*+,2  
 SBRK 0010H ;  
 ;  
 Cambiar 91H con C4H ;  
 ;  
 Cambiar D1H con C5H ;  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*\*  
 MAR \*\*+,2  
 ADRK 0020H  
 ;  
 Cambiar B1H con C6H ;  
 ;  
 Cambiar F1H con C7H ;  
 ZALS \*,AR1 ;  
 ADDH \*,AR2 ;  
 SACH \*,AR1

SACL \*  
 ADRK 0004H  
 LARP AR2  
 SBRK 0008H  
 Cambiar 89H con C8H  
 Cambiar C9H con C9H  
 Cambiar A9H con CAH  
 Cambiar E9H con CBH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*\*  
 MAR \*\*+,2  
 SBRK 0010H  
 Cambiar 99H con CCH  
 Cambiar D9H con CDH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*\*  
 MAR \*\*+,2  
 ADRK 0020H  
 Cambiar B9H con CEH  
 Cambiar F9H con CFH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 SBRK 0014H  
 Cambiar 85H con D0H  
 Cambiar C5H con D1H  
 Cambiar A5H con D2H  
 Cambiar E5H con D3H  
 ZALS \*,AR1

ADDH	*,AR2	
SACH	*,AR1	
SACL	*	
ADRK	0004H	
LARP	AR2	
ADRK	0010H	
	Cambiar 95H con D4H	:
	Cambiar D5H con D5H	:
	Cambiar B5H con D6H	:
	Cambiar F5H con D7H	:
ZALS	*,AR1	:
ADDH	*,AR2	:
SACH	*,AR1	:
SACL	*	:
ADRK	0004H	:
LARP	AR2	:
SBRK	0008H	:
	Cambiar 8DH con D8H	:
	Cambiar CDH con D9H	:
	Cambiar ADH con DAH	:
	Cambiar EDH con DBH	:
ZALS	*,AR1	:
ADDH	*,AR2	:
SACH	*,AR1	:
SACL	*	:
ADRK	0004H	:
LARP	AR2	:
ADRK	0010H	:
	Cambiar 9DH con DCH	:
	Cambiar DDH con DDH	:
	Cambiar BDH con DEH	:
	Cambiar FDH con DFH	:
ZALS	*,AR1	:
ADDH	*,AR2	:
SACH	*,AR1	:
SACL	*	:

ADRK	0008H	
LARP	AR2	
SBRK	000AH	
	Cambiar 83H con E0H	
	Cambiar C3H con E1H	
	Cambiar A3H con E2H	
	Cambiar E3H con E3H	
	Cambiar 93H con E4H	
	Cambiar D3H con E5H	
	Cambiar B3H con E6H	
	Cambiar F3H con E7H	
ZALS	*,AR1	
ADDH	*,AR2	
SACH	*,AR1	
SACL	*	
ADRK	0008H	
LARP	AR2	
ADRK	0008H	
	Cambiar 8BH con E8H	
	Cambiar CBH con E9H	
	Cambiar ABH con EAH	
	Cambiar EBH con EBH	
	Cambiar 9BH con ECH	
	Cambiar DBH con EDH	
	Cambiar BBH con EEH	
	Cambiar FBH con EFH	
ZALS	*,AR1	
ADDH	*,AR2	
SACH	*,AR1	
SACL	*	
	Cambiar 87H con F0H	
	Cambiar C7H con F1H	
	Cambiar A7H con F2H	
	Cambiar E7H con F3H	
	Cambiar 97H con F4H	
	Cambiar D7H con F5H	

```

;   Cambiar B7H con F6H
;   Cambiar F7H con F7H
;   Cambiar 8FH con F8H
;   Cambiar CFH con F9H
;   Cambiar AFH con FAH
;   Cambiar EFH con FBH
;   Cambiar 9FH con FCH
;   Cambiar DFH con FDH
;   Cambiar BFH con FEH
;   Cambiar FFH con FFH

```

```
TST1 EQU 006CH
```

```
; Buffers
```

```
IOB1 .USECT "BUFFER1",0100H
IOB2 .USECT "BUFFER2",0100H
```

```
B CICLO
```

```
; Registros
```

```
DDR EQU 0000H
DXR EQU 0001H
TIM EQU 0002H
PRD EQU 0003H
IMR EQU 0004H
GREG EQU 0005H
```

```
; Variables
```

```
BASE EQU 0060H
INDCS1 EQU 0061H
INDCS2 EQU 0062H
VCYS EQU 0063H
VCOS EQU 0064H
VSEN EQU 0065H
TEMP EQU 0066H
TEMPH EQU 0067H
TEMPL EQU 0068H
TEMPS EQU 0069H
INDT EQU 006AH
TST0 EQU 006BH
```

# Apéndice **E**

Félix Cuéllar [Hawk] 496602 ISE80

.TITLE 'FHT Inversa radix-2 de 128 puntos'

.ASECT "VECTORI",0000H

RESET B INIT  
 INT0 B ISR0  
 INT1 B ISR1  
 INT2 B ISR2

.SPACE 0100H

TINT B TIME  
 RINT B RCV  
 XINT B XMT  
 USER B PROC

Tablas de Cosenos y Senos (15bits)

.ASECT "ROMDATA",0020H

COS .WORD +7FFFH  
 .WORD +7FD7H  
 .WORD +7F61H  
 .WORD +7E9CH  
 .WORD +7D89H

.WORD +7C29H  
 .WORD +7A7CH  
 .WORD +7883H  
 .WORD +7640H  
 .WORD +73B5H  
 .WORD +70E1H  
 .WORD +6DC9H  
 .WORD +6A6CH  
 .WORD +66CEH  
 .WORD +62F1H  
 .WORD +5ED6H  
 .WORD +5A81H  
 .WORD +55F4H  
 .WORD +5133H  
 .WORD +4C3FH  
 .WORD +471CH  
 .WORD +41CDH  
 .WORD +3C56H  
 .WORD +36B9H  
 .WORD +30FBH  
 .WORD +2B1EH  
 .WORD +2527H  
 .WORD +1F19H  
 .WORD +18F8H  
 .WORD +12C7H  
 .WORD +0C8BH  
 .WORD +0647H  
 .WORD 0000H  
 .WORD -0647H  
 .WORD -0C8BH  
 .WORD -12C7H  
 .WORD -18F8H  
 .WORD -1F19H  
 .WORD -2527H  
 .WORD -2B1EH  
 .WORD -30FBH

.WORD -36B9H  
.WORD -3C56H  
.WORD -41CDH  
.WORD -471CH  
.WORD -4C3FH  
.WORD -5133H  
.WORD -55F4H  
.WORD -5A81H  
.WORD -5ED6H  
.WORD -62F1H  
.WORD -66CEH  
.WORD -6A6CH  
.WORD -6DC9H  
.WORD -70E1H  
.WORD -73B5H  
.WORD -7640H  
.WORD -7883H  
.WORD -7A7CH  
.WORD -7C29H  
.WORD -7D89H  
.WORD -7E9CH  
.WORD -7F61H  
.WORD -7FD7H  
  
SEN .WORD 0000H  
.WORD +0647H  
.WORD +0C8BH  
.WORD +12C7H  
.WORD +18F8H  
.WORD +1F19H  
.WORD +2527H  
.WORD +2B1EH  
.WORD +30FBH  
.WORD +36B9H  
.WORD +3C56H  
.WORD +41CDH

.WORD +471CH  
.WORD +4C3FH  
.WORD +5133H  
.WORD +55F4H  
.WORD +5A81H  
.WORD +5ED6H  
.WORD +62F1H  
.WORD +66CEH  
.WORD +6A6CH  
.WORD +6DC9H  
.WORD +70E1H  
.WORD +73B5H  
.WORD +7640H  
.WORD +7883H  
.WORD +7A7CH  
.WORD +7C29H  
.WORD +7D89H  
.WORD +7E9CH  
.WORD +7F61H  
.WORD +7FD7H  
.WORD +7FFFH  
.WORD +7FD7H  
.WORD +7F61H  
.WORD +7E9CH  
.WORD +7D89H  
.WORD +7C29H  
.WORD +7A7CH  
.WORD +7883H  
.WORD +7640H  
.WORD +73B5H  
.WORD +70E1H  
.WORD +6DC9H  
.WORD +6A6CH  
.WORD +66CEH  
.WORD +62F1H  
.WORD +5ED6H

```

.WORD +5A81H
.WORD +55F4H
.WORD +5133H
.WORD +4C3FH
.WORD +471CH
.WORD +41CDH
.WORD +3C56H
.WORD +36B9H
.WORD +30FBH
.WORD +2B1EH
.WORD +2527H
.WORD +1F19H
.WORD +18F8H
.WORD +12C7H
.WORD +0C8BH
.WORD +0647H

; Vectores de Interrupción

.ASECT "ROMCODE",00A0H

; Interrupción externa cero

ISR0: SST TST0 ;Almacenar registro de estado
      LAR6 AR6 ;Cambio de apuntador activo
      OUT *,PA8 ;Sacar dato transformado
      IN *BR0+,PA9 ;Almacenar dato en memoria
      LST TST0 ;Restablecer registro de estado
      RET ;Regreso de la interrupción

ISR1: RET

ISR2: RET

TIME: RET

```

```

RCV: RET

XMT: RET

PROC: RET

; Inicialización

INIT: DINT ;Deshabilitar Interrupciones
      ROVM ;Modo de Sobrelujo
      CNFD ;Configurar B0 como datos
      SSXM ;Configurar signo extendido
      LDPK 00H ;Página de datos cero
      LALK 00000H
      SACL GREG ;Configurar registro (Global)
      LALK 0FFFFH
      SACL TIM ;Configurar registro (Timer)
      SACL PRD ;Configurar registro (Período)
      LALK 0FFC1H
      SACL IMR ;Configurar registro (INT)
      SPM 00H ;Configurar corrimiento reg P
      SXF ;Habilitar bandera externa
      LARK AR0,0080H
      LARK AR6,0200H
      LALK 0200H
      SACL BASE
      EINT ;Habilitar interrupciones

; Ciclo de espera para el llenado del buffer

CICLO: SAR AR6,INDT
       LAC INDT
       ANDK 00FFH
       BNZ CICLO
       LAC INDT
       SACL BASE

```

ADLK 0100H  
 ORK 0200H  
 ANDK 0300H  
 SACL INDT  
 LAR AR6,INDT

; Preparación Mariposas

ZALS BASE  
 SBLK 0080H  
 SACL BASE  
 H10: ZALS BASE  
 ADLK 0080H  
 SACL BASE

; Primera Mariposa

LAR AR1,BASE  
 LAR AR2,BASE  
 LARP AR2  
 MAR \*+,AR1  
 H11: LARK \* AR5,003FH ;Número de ciclos  
 LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*+,1,AR2 ; $Pm+1 = (Pm + W Qm) / 2$   
 SUBH \*  
 SACH \*+,1 ; $Qm+1 = (Pm - W Qm) / 2$   
 MAR \*+,AR1  
 MAR \*+,AR5  
 BANZ H11,\*-,AR1 ;Comparación ciclos

; Segunda Mariposa

LAR AR1,BASE  
 LAR AR2,BASE  
 LARP AR2

ADRK 0002H  
 LARP AR1  
 LARK AR5,001FH ;Número de ciclos  
 H12: LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*+,1,AR2 ; $Pm+1 = (Pm + W Qm) / 2$   
 SUBH \*  
 SACH \*+,1,AR1 ; $Qm+1 = (Pm - W Qm) / 2$   
 LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*+,1,AR2 ; $Pm+1 = (Pm + W Qm) / 2$   
 SUBH \*  
 SACH \*,1 ; $Qm+1 = (Pm - W Qm) / 2$   
 ADRK 0003H  
 LARP AR1  
 ADRK 0003H  
 LARP AR5  
 BANZ H12,\*-,AR1 ;Comparación ciclos

; Tercera Mariposa

LAR AR1,BASE  
 LAR AR2,BASE  
 LAR AR3,BASE  
 LARP AR2  
 ADRK 0004H  
 LARP AR3  
 ADRK 0007H  
 LARP AR1  
 LARK AR5,000FH ;Número de ciclos  
 LALK 0030H  
 TBLR VCYS  
 LALK 0050H  
 TBLR VCOS  
 LALK 0090H  
 TBLR VSEN

HI3:

LAC	* ,15,AR2	
ADD	* ,15,AR1	
SACH	*+,1,AR2	;Pm+1 = (Pm + W Qm) / 2
SUBH	*	
SACH	*+,1	;Qm+1 = (Pm - W Qm) / 2
LAC	*	
SACL	TEMP	
LT	VCYS	
MPY	* ,AR3	;W1 = Cos(2πm/128) * Qm
LTP	VCYS	
MPY	* ,AR1	;W2 = Sen(2πm/128) * Qm
APAC		;W = W1 + W2
SACH	TEMPH	
SACL	TEMPL	
NEG		
ADD	* ,15,AR2	
SACH	*+,1,AR1	;Qm+1 = (Pm - W Qm) / 2
ZALS	TEMPL	
ADDH	TEMPH	
ADD	* ,15	
SACH	*+,1	;Pm+1 = (Pm + W Qm) / 2
LAC	* ,15,AR2	
ADD	* ,15,AR1	
SACH	*+,1,AR2	;Pm+1 = (Pm + W Qm) / 2
SUBH	*	
SACH	*+,1	;Qm+1 = (Pm - W Qm) / 2
LT	VCOS	
MPY	*	;W1 = Cos(2πm/128) * Qm
ZALH	TEMP	
SACH	*	
LTP	VSEN	
MPY	* ,AR1	;W2 = Sen(2πm/128) * Qm
APAC		;W = W1 + W2
SACH	TEMPH	
SACL	TEMPL	
NEG		

ADD	* ,15,AR2	
SACH	* ,1,AR1	;Qm+1 = (Pm - W Qm) / 2
ZALS	TEMPL	
ADDH	TEMPH	
ADD	* ,15	
SACH	* ,1,AR3	;Pm+1 = (Pm + W Qm) / 2
ADRK	0008H	
LARP	AR2	
ADRK	0005H	
LARP	AR1	
ADRK	0005H	
LARP	AR5	
BANZ	HI3,*-,AR1	;Comparación ciclos

Cuarta Mariposa

LAR	AR1,BASE	
LAR	AR2,BASE	
LARP	AR2	
ADRK	0008H	
LARP	AR1	
LARK	AR5,0007H	;Número de ciclos
LAC	* ,15,AR2	
ADD	* ,15,AR1	
SACH	* ,1,AR2	;Pm+1 = (Pm + W Qm) / 2
SUBH	*	
SACH	* ,1	;Qm+1 = (Pm - W Qm) / 2
ADRK	0004H	
LARP	AR1	
ADRK	0004H	
LAC	* ,15,AR2	
ADD	* ,15,AR1	
SACH	* ,1,AR2	;Pm+1 = (Pm + W Qm) / 2
SUBH	*	
SACH	* ,1,AR1	;Qm+1 = (Pm - W Qm) / 2
ADRK	03H	

HI4:

SAR AR1,TEMP  
 LAR AR3,TEMP  
 SBRK 06H  
 LARP AR2  
 ADRK 03H  
 SAR AR2,TEMP  
 LAR AR4,TEMP  
 SBRK 06H  
 LALK 0028H  
 SACL INDCS1  
 LALK 0058H  
 SACL INDCS2  
 SAR AR5,TEMPS  
 PSHD TEMPS  
 LARK AR5,0002H ;Número de ciclos  
 ZALS INDCS1  
 TBLR VCOS  
 ADLK 0040H  
 TBLR VSEN  
 SBLK 0038H  
 SACL INDCS1  
 LT VCOS  
 MPY \*,AR4 ;W1 = Cos(2πm/128) \* Qm  
 LTP VSEN  
 MPY \*,AR2 ;W2 = Sen(2πm/128) \* Qm  
 APAC ;W = W1 + W2  
 SACH TEMPH  
 SACL TEMPL  
 NEG  
 LT  
 ADD \*,AR1  
 SACH \*,15,AR2  
 ZALS \*,+,1,AR1 ;Qm+1 = (Pm - W Qm) / 2  
 ADDH TEMPL  
 ADD \*,15  
 SACH \*,+,1,AR4 ;Pm+1 = (Pm + W Qm) / 2

HI4F:

ZALS INDCS2  
 TBLR VCOS  
 ADLK 0040H  
 TBLR VSEN  
 SBLK 0048H  
 SACL INDCS2  
 MPY VSEN ;W1 = Sen(2πm/128) \* Qm  
 LTP VCOS  
 MPY \*,AR3 ;W2 = Cos(2πm/128) \* Qm  
 APAC ;W = W1 + W2  
 SACH TEMPH  
 SACL TEMPI.  
 NEG  
 ADD \*,15,AR4  
 SACH \*,-,1,AR3 ;Qm+1 = (Pm - W Qm) / 2  
 ZALS TEMPL  
 ADDH TEMPH  
 ADD \*,15  
 SACH \*,-,1,AR5 ;Pm+1 = (Pm + W Qm) / 2  
 BANZ HI4F,\*,-,AR2 ;Comparación ciclos  
 POPD TEMPS  
 LAR AR5,TEMPS  
 ADRK 000CH  
 LARP AR1  
 ADRK 000CH  
 LARP AR5  
 BANZ HI4,\*,-,AR1 ;Comparación ciclos

; Quinta Mariposa

LAR AR1,BASE  
 LAR AR2,BASE  
 LARP AR2  
 ADRK 0010H  
 LARP AR1  
 LARK AR5,0003H ;Número de ciclos

```

HI5:  LAC      * ,15,AR2
      ADD      * ,15,AR1
      SACH     * ,1,AR2 ;Pm+1 = (Pm + W Qm) / 2
      SUBH     *
      SACH     * ,1 ;Qm+1 = (Pm - W Qm) / 2
      ADRK     0008H
      LARP     AR1
      ADRK     0008H
      LAC      * ,15,AR2
      ADD      * ,15,AR1
      SACH     * ,1,AR2 ;Pm+1 = (Pm + W Qm) / 2
      SUBH     *
      SACH     * ,1,AR1 ;Qm+1 = (Pm - W Qm) / 2
      ADRK     07H
      SAR      AR1,TEMP
      LAR      AR3,TEMP
      SBRK     0EH
      LARP     AR2
      ADRK     07H
      SAR      AR2,TEMP
      LAR      AR4,TEMP
      SBRK     0EH
      LALK     0024H
      SACL     INDCS1
      LALK     005CH
      SACL     INDCS2
      SAR      AR5,TEMPS
      PSHD     TEMPS
      LARK     AR5,0006H ;Número de ciclos
HI5F: ZALS     INDCS1
      TBLR     VCOS
      ADLK     0040H
      TBLR     VSEN
      SBLK     003CH
      SACL     INDCS1
      LT       VCOS

```

```

      MPY      * ,AR4 ;W1 = Cos(2πm/128) * Qm
      LTP      VSEN
      MPY      * ,AR2 ;W2 = Sen(2πm/128) * Qm
      APAC     ;W = W1 + W2
      SACH     TEMPH
      SACL     TEMPL
      NEG
      LT       * ,AR1
      ADD      * ,15,AR2
      SACH     * +,1,AR1 ;Qm+1 = (Pm - W Qm) / 2
      ZALS     TEMPL
      ADDH     TEMPH
      ADD      * ,15
      SACH     * +,1,AR4 ;Pm+1 = (Pm + W Qm) / 2
      ZALS     INDCS2
      TBLR     VCOS
      ADLK     0040H
      TBLR     VSEN
      SBLK     0044H
      SACL     INDCS2
      MPY      VSEN ;W1 = Sen(2πm/128) * Qm
      LTP      VCOS
      MPY      * ,AR3 ;W2 = Cos(2πm/128) * Qm
      APAC     ;W = W1 + W2
      SACH     TEMPH
      SACL     TEMPL
      NEG
      ADD      * ,15,AR4
      SACH     * -,1,AR3 ;Qm+1 = (Pm - W Qm) / 2
      ZALS     TEMPL
      ADDH     TEMPH
      ADD      * ,15
      SACH     * -,1,AR5 ;Pm+1 = (Pm + W Qm) / 2
      BANZ     HI5F,*-,AR2 ;Comparación ciclos
      POPD     TEMPS
      LAR      AR5,TEMPS

```

ADRK 0018H  
 LARP AR1  
 ADRK 0018H  
 LARP AR5  
 BANZ HI5,\*-,AR1 ;Comparación ciclos

; Sexta Mariposa

HI6:  
 LAR AR1,BASE  
 LAR AR2,BASE  
 LARP AR2  
 ADRK 0020H  
 LARP AR1  
 LARK AR5,0001H ;Número de ciclos  
 LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*,1,AR2 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 SUBH \*  
 SACH \*,1 ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ADRK 0010H  
 LARP AR1  
 ADRK 0010H  
 LAC \*,15,AR2  
 ADD \*,15,AR1  
 SACH \*,1,AR2 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 SUBH \*  
 SACH \*,1,AR1 ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ADRK 0FH  
 SAR AR1,TEMP  
 LAR AR3,TEMP  
 SBRK 1EH  
 LARP AR2  
 ADRK 0FH  
 SAR AR2,TEMP  
 LAR AR4,TEMP  
 SBRK 1EH

HI6F:

LALK 0022H  
 SACL INDCS1  
 LALK 005EH  
 SACL INDCS2  
 SAR AR5,TEMPS  
 PSHD TEMPS  
 LARK AR5,000EH ;Número de ciclos  
 ZALS INDCS1  
 TBLR VCOS  
 ADLK 0040H  
 TBLR VSEN  
 SBLK 003EH  
 SACL INDCS1  
 LT VCOS  
 MPY \*,AR4 ; $W1 = \text{Cos}(2\pi m/128) * Q_m$   
 LTP VSEN  
 MPY \*,AR2 ; $W2 = \text{Sen}(2\pi m/128) * Q_m$   
 APAC ; $W = W1 + W2$   
 SACH TEMPH  
 SACL TEMPL  
 NEG  
 LT \*,AR1  
 ADD \*,15,AR2  
 SACH \*,1,AR1 ; $Q_{m+1} = (P_m - W Q_m) / 2$   
 ZALS TEMPL  
 ADDH TEMPH  
 ADD \*,15  
 SACH \*,1,AR4 ; $P_{m+1} = (P_m + W Q_m) / 2$   
 ZALS INDCS2  
 TBLR VCOS  
 ADLK 0040H  
 TBLR VSEN  
 SBLK 0042H  
 SACL INDCS2  
 MPY VSEN ; $W1 = \text{Sen}(2\pi m/128) * Q_m$   
 LTP VCOS

MPY \* ,AR3 ;W2 = Cos(2πm/128) \* Qm  
 APAC ;W = W1 + W2  
 SACH TEMPH  
 SACL TEMPL  
 NEG  
 ADD \* ,15,AR4  
 SACH \* -,1,AR3 ;Qm+1 = (Pm - W Qm) / 2  
 ZALS TEMPL  
 ADDH TEMPH  
 ADD \* ,15  
 SACH \* -,1,AR5 ;Pm+1 = (Pm + W Qm) / 2  
 BANZ HI6F,\*,AR2 ;Comparación ciclos  
 POPD TEMPS  
 LAR AR5,TEMPS  
 ADRK 0030H  
 LARP AR1  
 ADRK 0030H  
 LARP AR5  
 BANZ HI6,\*,AR1 ;Comparación ciclos

; Séptima Mariposa

HI7:

LAR AR1,BASE  
 LAR AR2,BASE  
 LARP AR2  
 ADRK 0040H  
 LARP AR1  
 LARK AR5,0000H ;Número de ciclos  
 LAC \* ,15,AR2  
 ADD \* ,15,AR1  
 SACH \* ,1,AR2 ;Pm+1 = (Pm + W Qm) / 2  
 SUBH \*  
 SACH \* ,1 ;Qm+1 = (Pm - W Qm) / 2  
 ADRK 0020H  
 LARP AR1  
 ADRK 0020H

HI7F:

LAC \* ,15,AR2  
 ADD \* ,15,AR1  
 SACH \* ,1,AR2 ;Pm+1 = (Pm + W Qm) / 2  
 SUBH \*  
 SACH \* ,1,AR1 ;Qm+1 = (Pm - W Qm) / 2  
 ADRK 1FH  
 SAR AR1,TEMP  
 LAR AR3,TEMP  
 SBRK 3EH  
 LARP AR2  
 ADRK 1FH  
 SAR AR2,TEMP  
 LAR AR4,TEMP  
 SBRK 3EH  
 LALK 0021H  
 SACL INDCS1  
 LALK 005FH  
 SACL INDCS2  
 SAR AR5,TEMPS  
 PSHD TEMPS  
 LARK AR5,001EH ;Número de ciclos  
 ZALS INDCS1  
 TBLR VCOS  
 ADLK 0040H  
 TBLR VSEN  
 SBLK 003FH  
 SACL INDCS1  
 LT VCOS  
 MPY \* ,AR4 ;W1 = Cos(2πm/128) \* Qm  
 LTP VSEN  
 MPY \* ,AR2 ;W2 = Sen(2πm/128) \* Qm  
 APAC ;W = W1 + W2  
 SACH TEMPH  
 SACL TEMPL  
 NEG  
 LT \* ,AR1

ADD	*15,AR2	
SACH	*+,1,AR1	; $Q_{m+1} = (P_m - W Q_m) / 2$
ZALS	TEMPL	
ADDH	TEMPH	
ADD	*15	
SACH	*+,1,AR4	; $P_{m+1} = (P_m + W Q_m) / 2$
ZALS	INDCS2	
TBLR	VCOS	
ADLK	0040H	
TBLR	VSEN	
SBLK	0041H	
SACL	INDCS2	
MPY	VSEN	; $W1 = \text{Sen}(2\pi m/128) * Q_m$
LTP	VCOS	
MPY	*AR3	; $W2 = \text{Cos}(2\pi m/128) * Q_m$
APAC		; $W = W1 + W2$
SACH	TEMPH	
SACL	TEMPL	
NEG		
ADD	*15,AR4	
SACH	*-,1,AR3	; $Q_{m+1} = (P_m - W Q_m) / 2$
ZALS	TEMPL	
ADDH	TEMPH	
ADD	*15	
SACH	*-,1,AR5	; $P_{m+1} = (P_m + W Q_m) / 2$
BANZ	HI7F,*-,AR2	; Comparación ciclos
POPD	TEMPS	
LAR	AR5,TEMPS	
ADRK	0060H	
LARP	AR1	
ADRK	0060H	
LARP	AR5	
BANZ	HI7,*-,AR1	; Comparación ciclos

; Cambio de Canal

BIT	BASE,08H
BBZ	HI0

Reordenamiento de Elementos

LAR	AR1,BASE
SBRK	007FH
LARP	AR2
LAR	AR2,BASE
SBRK	0040H
	; Cambiar 00H con 00H
	; Cambiar 40H con 01H
ZALS	*AR1
ADDH	*AR2
SACH	*AR1
SACL	*+,AR2
SBRK	0020H
	; Cambiar 20H con 02H
ZALS	*AR1
ADDH	*AR2
SACH	*AR1
SACL	*+,AR2
ADRK	0040H
	; Cambiar 60H con 03H
ZALS	*AR1
ADDH	*AR2
SACH	*AR1
SACL	*+,AR2
SBRK	0050H
	; Cambiar 10H con 04H
ZALS	*AR1
ADDH	*AR2
SACH	*AR1
SACL	*+,AR2
ADRK	0040H
	; Cambiar 50H con 05H

ZALS \* ,AR1  
ADDH \* ,AR2  
SACH \* ,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar 30H con 06H  
ZALS \* ,AR1  
ADDH \* ,AR2  
SACH \* ,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar 70H con 07H  
ZALS \* ,AR1  
ADDH \* ,AR2  
SACH \* ,AR1  
SACL \*+  
MAR \*+,AR2  
SBRK 0028H  
Cambiar 08H con 08H  
Cambiar 48H con 09H  
ZALS \* ,AR1  
ADDH \* ,AR2  
SACH \* ,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar 28H con 0AH  
ZALS \* ,AR1  
ADDH \* ,AR2  
SACH \* ,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar 68H con 0BH  
ZALS \* ,AR1  
ADDH \* ,AR2  
SACH \* ,AR1  
SACL \*+,AR2

SBRK 0050H  
Cambiar 18H con 0CH  
ZALS \* ,AR1  
ADDH \* ,AR2  
SACH \* ,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar 58H con 0DH  
ZALS \* ,AR1  
ADDH \* ,AR2  
SACH \* ,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar 38H con 0EH  
ZALS \* ,AR1  
ADDH \* ,AR2  
SACH \* ,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar 78H con 0FH  
ZALS \* ,AR1  
ADDH \* ,AR2  
SACH \* ,AR1  
SACL \*+  
MAR \*+,AR2  
SBRK 0034H  
Cambiar 04H con 10H  
Cambiar 44H con 11H  
ZALS \* ,AR1  
ADDH \* ,AR2  
SACH \* ,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar 24H con 12H  
ZALS \* ,AR1  
ADDH \* ,AR2

SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H  
Cambiar 64H con 13H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+  
MAR \*\*+,AR2  
SBRK 0010H  
Cambiar 14H con 14H  
Cambiar 54H con 15H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0020H  
Cambiar 34H con 16H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H  
Cambiar 74H con 17H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+  
MAR \*\*+,AR2  
SBRK 28H  
Cambiar 0CH con 18H  
Cambiar 4CH con 19H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2

SBRK 0020H  
Cambiar 2CH con 1AH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H  
Cambiar 6CH con 1BH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+  
MAR \*\*+,2  
SBRK 0010H  
Cambiar 1CH con 1CH  
Cambiar 5CH con 1DH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
SBRK 0020H  
Cambiar 3CH con 1EH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+,AR2  
ADRK 0040H  
Cambiar 7CH con 1FH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*\*+  
MAR \*\*+,2  
SBRK 003AH  
Cambiar 02H con 20H  
Cambiar 42H con 21H

ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,2  
 ADRK 0020H  
 Cambiar 22H con 22H :  
 Cambiar 62H con 23H :  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,2  
 SBRK 0010H  
 Cambiar 12H con 24H :  
 Cambiar 52H con 25H :  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+,AR2  
 SBRK 0020H  
 Cambiar 32H con 26H ;  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+,AR2  
 ADRK 0040H  
 Cambiar 72H con 27H :  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,2  
 SBRK 0028H  
 Cambiar 0AH con 28H :  
 Cambiar 4AH con 29H :

ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,2  
 ADRK 0020H  
 Cambiar 2AH con 2AH  
 Cambiar 6AH con 2BH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,2  
 SBRK 0010H  
 Cambiar 1AH con 2CH  
 Cambiar 5AH con 2DH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+,AR2  
 SBRK 0020H  
 Cambiar 3AH con 2EH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+,AR2  
 ADRK 0040H  
 Cambiar 7AH con 2FH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*+,2  
 SBRK 0034H  
 Cambiar 06H con 30H  
 Cambiar 46H con 31H

ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*\*+,2  
 ADRK 0020H  
 Cambiar 26H con 32H  
 Cambiar 66H con 33H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*\*+,2  
 SBRK 0010H  
 Cambiar 16H con 34H  
 Cambiar 56H con 35H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*\*+,2  
 ADRK 0020H  
 Cambiar 36H con 36H  
 Cambiar 76H con 37H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*\*+,2  
 SBRK 0028H  
 Cambiar 0EH con 38H  
 Cambiar 4EH con 39H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+

MAR \*\*+,2  
 ADRK 0020H  
 Cambiar 2EH con 3AH  
 Cambiar 6EH con 3BH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*\*+,2  
 SBRK 0010H  
 Cambiar 1EH con 3CH  
 Cambiar 5EH con 3DH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*\*+,2  
 ADRK 0020H  
 Cambiar 3EH con 3EH  
 Cambiar 7EH con 3FH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 SBRK 001DH  
 Cambiar 01H con 40H  
 Cambiar 41H con 41H  
 Cambiar 21H con 42H  
 Cambiar 61H con 43H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*\*+,2

SBRK 0010H  
 Cambiar 11H con 44H  
 Cambiar 51H con 45H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*\*  
 MAR \*\*,+2  
 ADRK 0020H  
 Cambiar 31H con 46H  
 Cambiar 71H con 47H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 SBRK 0008H  
 Cambiar 09H con 48H  
 Cambiar 49H con 49H  
 Cambiar 29H con 4AH  
 Cambiar 69H con 4BH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*\*  
 MAR \*\*,+2  
 SBRK 0010H  
 Cambiar 19H con 4CH  
 Cambiar 59H con 4DH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*\*  
 MAR \*\*,+2  
 ADRK 0020H

Cambiar 39H con 4EH  
 Cambiar 79H con 4FH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 SBRK 0014H  
 Cambiar 05H con 50H  
 Cambiar 45H con 51H  
 Cambiar 25H con 52H  
 Cambiar 65H con 53H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 ADRK 0010H  
 Cambiar 15H con 54H  
 Cambiar 55H con 55H  
 Cambiar 35H con 56H  
 Cambiar 75H con 57H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 SBRK 0008H  
 Cambiar 0DH con 58H  
 Cambiar 4DH con 59H  
 Cambiar 2DH con 5AH  
 Cambiar 6DH con 5BH  
 ZALS \*,AR1

ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 ADRK 0010H  
 Cambiar 1DH con 5CH  
 Cambiar 5DH con 5DH  
 Cambiar 3DH con 5EH  
 Cambiar 7DH con 5FH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0008H  
 LARP AR2  
 SBRK 000AH  
 Cambiar 03H con 60H  
 Cambiar 43H con 61H  
 Cambiar 23H con 62H  
 Cambiar 63H con 63H  
 Cambiar 13H con 64H  
 Cambiar 53H con 65H  
 Cambiar 33H con 66H  
 Cambiar 73H con 67H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0008H  
 LARP AR2  
 ADRK 0008H  
 Cambiar 0BH con 68H  
 Cambiar 4BH con 69H  
 Cambiar 2BH con 6AH  
 Cambiar 6BH con 6BH

Cambiar 1BH con 6CH  
 Cambiar 5BH con 6DH  
 Cambiar 3BH con 6EH  
 Cambiar 7BH con 6FH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0012H  
 LARP AR2  
 ADRK 0045H  
 Cambiar 07H con 70H  
 Cambiar 47H con 71H  
 Cambiar 27H con 72H  
 Cambiar 67H con 73H  
 Cambiar 17H con 74H  
 Cambiar 57H con 75H  
 Cambiar 37H con 76H  
 Cambiar 77H con 77H  
 Cambiar 0FH con 78H  
 Cambiar 4FH con 79H  
 Cambiar 2FH con 7AH  
 Cambiar 6FH con 7BH  
 Cambiar 1FH con 7CH  
 Cambiar 5FH con 7DH  
 Cambiar 3FH con 7EH  
 Cambiar 7FH con 7FH  
 Cambiar 80H con 80H  
 Cambiar 0H con 81H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*,AR2  
 SBRK 0020H  
 Cambiar A0H con 82H

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar E0H con 83H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0050H  
Cambiar 90H con 84H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar D0H con 85H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar B0H con 86H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar F0H con 87H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+  
MAR \*+,2  
SBRK 0028H

Cambiar 88H con 88H  
Cambiar C8H con 89H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar A8H con 8AH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar E8H con 8BH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0050H  
Cambiar 98H con 8CH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar D8H con 8DH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar B8H con 8EH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2

ADRK 0040H  
Cambiar F8H con 8FH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+  
MAR \*+,AR2  
SBRK 0034H  
Cambiar 84H con 90H  
Cambiar C4H con 91H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar A4H con 92H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar E4H con 93H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+  
MAR \*+,AR2  
SBRK 0010H  
Cambiar 94H con 94H  
Cambiar D4H con 95H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar B4H con 96H

ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar F4H con 97H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+  
MAR \*+,AR2  
SBRK 0028H  
Cambiar 8CH con 98H  
Cambiar CCH con 99H  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
SBRK 0020H  
Cambiar ACH con 9AH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+,AR2  
ADRK 0040H  
Cambiar ECH con 9BH  
ZALS \*,AR1  
ADDH \*,AR2  
SACH \*,AR1  
SACL \*+  
MAR \*+,2  
SBRK 0010H  
Cambiar 9CH con 9CH  
Cambiar DCH con 9DH  
ZALS \*,AR1  
ADDH \*,AR2

SACH \*.AR1  
 SACL \*+,AR2  
 SBRK 0020H  
 Cambiar BCH con 9EH  
 ZALS \*.AR1  
 ADDH \*.AR2  
 SACH \*.AR1  
 SACL \*+,AR2  
 ADRK 0040H  
 Cambiar FCH con 9FH  
 ZALS \*.AR1  
 ADDH \*.AR2  
 SACH \*.AR1  
 SACL \*+  
 MAR \*+,2  
 SBRK 003AH  
 Cambiar 82H con A0H  
 Cambiar C2H con A1H  
 ZALS \*.AR1  
 ADDH \*.AR2  
 SACH \*.AR1  
 SACL \*+  
 MAR \*+,2  
 ADRK 0020H  
 Cambiar A2H con A2H  
 Cambiar E2H con A3H  
 ZALS \*.AR1  
 ADDH \*.AR2  
 SACH \*.AR1  
 SACL \*+  
 MAR \*+,2  
 SBRK 0010H  
 Cambiar 92H con A4H  
 Cambiar D2H con A5H  
 ZALS \*.AR1  
 ADDH \*.AR2

SACH \*.AR1  
 SACL \*+,AR2  
 SBRK 0020H  
 Cambiar B2H con A6H  
 ZALS \*.AR1  
 ADDH \*.AR2  
 SACH \*.AR1  
 SACL \*+,AR2  
 ADRK 0040H  
 Cambiar F2H con A7H  
 ZALS \*.AR1  
 ADDH \*.AR2  
 SACH \*.AR1  
 SACL \*+  
 MAR \*+,2  
 SBRK 0028H  
 Cambiar 8AH con A8H  
 Cambiar CAH con A9H  
 ZALS \*.AR1  
 ADDH \*.AR2  
 SACH \*.AR1  
 SACL \*+  
 MAR \*+,2  
 ADRK 0020H  
 Cambiar AAH con AAH  
 Cambiar EAH con ABH  
 ZALS \*.AR1  
 ADDH \*.AR2  
 SACH \*.AR1  
 SACL \*+  
 MAR \*+,2  
 SBRK 0010H  
 Cambiar 9AH con ACH  
 Cambiar DAH con ADH  
 ZALS \*.AR1  
 ADDH \*.AR2

SACH \*,AR1  
 SACL \*,AR2  
 SBRK 0020H  
 Cambiar BAH con AEH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*,AR2  
 ADRK 0040H  
 Cambiar FAH con AFH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*,+2  
 SBRK 0034H  
 Cambiar 86H con B0H  
 Cambiar C6H con B1H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*,+2  
 ADRK 0020H  
 Cambiar A6H con B2H  
 Cambiar E6H con B3H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*,+2  
 SBRK 0010H  
 Cambiar 96H con B4H  
 Cambiar D6H con B5H  
 ZALS \*,AR1  
 ADDH \*,AR2

SACH \*,AR1  
 SACL \*+  
 MAR \*,+2  
 ADRK 0020H  
 Cambiar B6H con B6H  
 Cambiar F6H con B7H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*,+2  
 SBRK 0028H  
 Cambiar 8EH con B8H  
 Cambiar CEH con B9H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*,+2  
 ADRK 0020H  
 Cambiar AEH con BAH  
 Cambiar EEH con BBH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*,+2  
 SBRK 0010H  
 Cambiar 9EH con BCH  
 Cambiar DEH con BDH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*+  
 MAR \*,+2  
 ADRK 0020H

Cambiar BEH con BEH

Cambiar FEH con BFH

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*

ADRK 0004H

LARP AR2

SBRK 001DH

Cambiar 81H con C0H

Cambiar C1H con C1H

Cambiar A1H con C2H

Cambiar E1H con C3H

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*+

MAR \*+,2

SBRK 0010H

Cambiar 91H con C4H

Cambiar D1H con C5H

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*+

MAR \*+,2

ADRK 0020H

Cambiar B1H con C6H

Cambiar F1H con C7H

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*

ADRK 0004H

LARP AR2

SBRK 0008H

Cambiar 89H con C8H

Cambiar C9H con C9H

Cambiar A9H con CAH

Cambiar E9H con CBH

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*+

MAR \*+,2

SBRK 0010H

Cambiar 99H con CCH

Cambiar D9H con CDH

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*+

MAR \*+,2

ADRK 0020H

Cambiar B9H con CEH

Cambiar F9H con CFH

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*

ADRK 0004H

LARP AR2

SBRK 0014H

Cambiar 85H con D0H

Cambiar C5H con D1H

Cambiar A5H con D2H

Cambiar E5H con D3H

ZALS \*,AR1

ADDH \*,AR2

SACH \*,AR1

SACL \*

ADRK 0004H

LARP AR2  
 ADRK 0010H  
 Cambiar 95H con D4H  
 Cambiar D5H con D5H  
 Cambiar B5H con D6H  
 Cambiar F5H con D7H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 SBRK 0008H  
 Cambiar 8DH con D8H  
 Cambiar CDH con D9H  
 Cambiar ADH con DAH  
 Cambiar EDH con DBH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0004H  
 LARP AR2  
 ADRK 0010H  
 Cambiar 9DH con DCH  
 Cambiar DDH con DDH  
 Cambiar BDH con DEH  
 Cambiar FDH con DFH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0008H  
 LARP AR2  
 SBRK 000AH  
 Cambiar 83H con E0H

Cambiar C3H con E1H  
 Cambiar A3H con E2H  
 Cambiar E3H con E3H  
 Cambiar 93H con E4H  
 Cambiar D3H con E5H  
 Cambiar B3H con E6H  
 Cambiar F3H con E7H  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 ADRK 0008H  
 LARP AR2  
 ADRK 0008H  
 Cambiar 8BH con E8H  
 Cambiar CBH con E9H  
 Cambiar ABH con EAH  
 Cambiar EBH con EBH  
 Cambiar 9BH con ECH  
 Cambiar DBH con EDH  
 Cambiar BBH con EEH  
 Cambiar FBH con EFH  
 ZALS \*,AR1  
 ADDH \*,AR2  
 SACH \*,AR1  
 SACL \*  
 Cambiar 87H con F0H  
 Cambiar C7H con F1H  
 Cambiar A7H con F2H  
 Cambiar E7H con F3H  
 Cambiar 97H con F4H  
 Cambiar D7H con F5H  
 Cambiar B7H con F6H  
 Cambiar F7H con F7H  
 Cambiar 8FH con F8H  
 Cambiar CFH con F9H

;  
; Cambiar AFH con FAH  
;  
; Cambiar EFH con FBH  
;  
; Cambiar 9FH con FCH  
;  
; Cambiar DFH con FDH  
;  
; Cambiar BFH con FEH  
;  
; Cambiar FFH con FFH

JOB1 .USECT "BUFFER1",0100H  
JOB2 .USECT "BUFFER2",0100H

B            CICLO

;        Registros

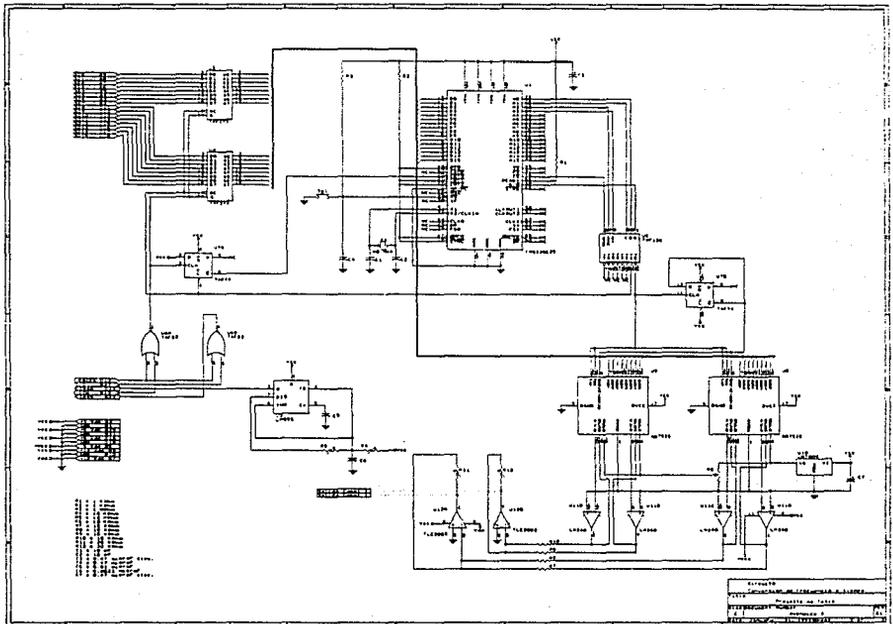
DDR	EQU	0000H
DXR	EQU	0001H
TIM	EQU	0002H
PRD	EQU	0003H
IMR	EQU	0004H
GREG	EQU	0005H

;        Variables

BASE	EQU	0060H
INDCS1	EQU	0061H
INDCS2	EQU	0062H
VCYS	EQU	0063H
VCOS	EQU	0064H
VSEN	EQU	0065H
TEMP	EQU	0066H
TEMPH	EQU	0067H
TEMPL	EQU	0068H
TEMPS	EQU	0069H
INDT	EQU	006AH
TST0	EQU	006BH
TST1	EQU	006CH

;        Buffers





# Apéndice **H**

## H.1 Transformada Directa Rápida de Hartley

- A = Arreglo de prueba representado en forma normalizada.
- B = Arreglo de prueba representado con 15 bits en formato entero.
- C = Resultados de la reversión de bits.
- D = Resultados de la primera mariposa.
- E = Resultados de la segunda mariposa.
- F = Resultados de la tercera mariposa.
- G = Resultados de la cuarta mariposa.
- H = Resultados de la quinta mariposa.
- I = Resultados de la sexta mariposa.
- J = Resultados de la séptima mariposa. FHT terminada.

	A	P	C	D	E	F	G	H	I	J
0.100	3276	3276	3276	3276	3276	3276	12451	13819	18049	16431
0.200	6553	3276	0	0	0	0	0	0	0	0
0.250	8191	3276	3276	0	0	0	0	0	0	0
0.120	3932	3276	0	0	0	0	0	0	0	0
0.360	11796	3276	3276	3276	0	0	0	0	0	0
0.400	13106	3276	0	0	0	0	0	0	0	0
0.800	26213	3276	3276	0	0	0	0	0	0	0
0.660	21626	3276	0	0	0	0	0	0	0	0
0.660	21626	21626	21626	21626	21626	-9175	-6283	-6152	-4088	
0.660	21626	21626	0	0	0	0	0	0	0	0
0.770	25230	21626	21626	0	0	0	0	0	0	0
0.110	3604	21626	0	0	0	0	0	0	0	0
0.567	18578	21626	21626	21626	0	0	0	0	0	0
0.517	16940	21626	0	0	0	0	0	0	0	0
0.900	29490	21626	21626	0	0	0	0	0	0	0
0.950	31128	21626	0	0	0	0	0	0	0	0
0.100	3276	11796	11796	11796	11796	15187	-1368	-3469	-1899	
0.200	6553	11796	0	0	0	0	0	0	0	0
0.250	8191	11796	11796	0	0	0	0	0	0	0
0.120	3932	11796	0	0	0	0	0	0	0	0
0.360	11796	11796	11796	11796	0	0	0	0	0	0
0.400	13106	11796	0	0	0	0	0	0	0	0
0.800	26213	11796	11796	0	0	0	0	0	0	0
0.660	21626	11796	0	0	0	0	0	0	0	0
0.660	21626	18578	18578	18578	18578	-3391	-2892	-2025	-2720	
0.660	21626	18578	0	0	0	0	0	0	0	0

0.770	25230	18578	18578	0	0	0	0	0	0	0
0.110	3604	18578	0	0	0	0	0	0	0	0
0.567	18578	18578	18578	18578	0	0	0	0	0	0
0.517	16940	18578	0	0	0	0	0	0	0	0
0.900	29490	18578	18578	0	0	0	0	0	0	0
0.950	31128	18578	0	0	0	0	0	0	0	0
0.100	3276	8191	8191	8191	8191	16710	22280	-4230	-2244	
0.200	6553	8191	0	0	0	0	0	0	0	0
0.250	8191	8191	8191	0	0	0	0	0	0	0
0.120	3932	8191	0	0	0	0	0	0	0	0
0.360	11796	8191	8191	8191	0	0	0	0	0	0
0.400	13106	8191	0	0	0	0	0	0	0	0
0.800	26213	8191	8191	0	0	0	0	0	0	0
0.660	21626	8191	0	0	0	0	0	0	0	0
0.660	21626	25230	25230	25230	25230	-8519	-5078	-130	-1026	
0.660	21626	25230	0	0	0	0	0	0	0	0
0.770	25230	25230	25230	0	0	0	0	0	0	0
0.110	36042	5230	0	0	0	0	0	0	0	0
0.567	18578	25230	25230	25230	0	0	0	0	0	0
0.517	16940	25230	0	0	0	0	0	0	0	0
0.900	29490	25230	25230	0	0	0	0	0	0	0
0.950	31128	25230	0	0	0	0	0	0	0	0
0.100	3276	26213	26213	26213	26213	27851	-5570	2101	-2945	
0.200	6553	26213	0	0	0	0	0	0	0	0
0.250	8191	26213	26213	0	0	0	0	0	0	0
0.120	3932	26213	0	0	0	0	0	0	0	0
0.360	11796	26213	26213	26213	0	0	0	0	0	0
0.400	13106	26213	0	0	0	0	0	0	0	0
0.800	26213	26213	26213	0	0	0	0	0	0	0
0.660	21626	26213	0	0	0	0	0	0	0	0
0.660	21626	29490	29490	29490	29490	-1638	-3440	-867	-1001	
0.660	21626	29490	0	0	0	0	0	0	0	0
0.770	25230	29490	29490	0	0	0	0	0	0	0
0.110	3604	29490	0	0	0	0	0	0	0	0
0.567	18578	29490	29490	29490	0	0	0	0	0	0
0.517	16940	29490	0	0	0	0	0	0	0	0
0.900	29490	29490	29490	0	0	0	0	0	0	0
0.950	31128	29490	0	0	0	0	0	0	0	0
0.100	3276	6553	6553	6553	6553	14089	14556	14814	1617	
0.200	6553	6553	0	0	0	0	0	0	0	0
0.250	8191	6553	6553	0	0	0	0	0	0	0
0.120	3932	6553	0	0	0	0	0	0	0	0
0.360	11796	6553	6553	6553	0	0	0	0	0	0
0.400	13106	6553	0	0	0	0	0	0	0	0
0.800	26213	6553	6553	0	0	0	0	0	0	0
0.660	21626	6553	0	0	0	0	0	0	0	0
0.660	21626	21626	21626	21626	21626	-7536	-4726	-2305	-2064	
0.660	21626	216260	0	0	0	0	0	0	0	0
0.770	25230	21626	21626	0	0	0	0	0	0	0
0.110	3604	21626	0	0	0	0	0	0	0	0
0.567	18578	21626	21626	21626	0	0	0	0	0	0

0.517	16940	21626	0	0	0	0	0	0	0	0
0.900	29490	21626	21626	0	0	0	0	0	0	0
0.950	31128	21626	0	0	0	0	0	0	0	0
0.100	3276	13106	13106	13106	13106	15023	-467	-5885	-1569	0
0.200	6553	13106	0	0	0	0	0	0	0	0
0.250	8191	13106	13106	0	0	0	0	0	0	0
0.120	3932	13106	0	0	0	0	0	0	0	0
0.360	11796	13106	13106	13106	0	0	0	0	0	0
0.400	13106	13106	0	0	0	0	0	0	0	0
0.800	26213	13106	13106	0	0	0	0	0	0	0
0.660	21626	13106	0	0	0	0	0	0	0	0
0.660	21626	16940	16940	16940	16940	-1917	-2809	-3083	695	0
0.660	21626	16940	0	0	0	0	0	0	0	0
0.770	25230	16940	16940	0	0	0	0	0	0	0
0.110	3604	16940	0	0	0	0	0	0	0	0
0.567	18578	16940	16940	16940	0	0	0	0	0	0
0.517	16940	16940	0	0	0	0	0	0	0	0
0.900	29490	16940	16940	0	0	0	0	0	0	0
0.950	31128	16940	0	0	0	0	0	0	0	0
0.100	3276	3932	3932	3932	3932	3768	15072	-258	-1986	0
0.200	6553	3932	0	0	0	0	0	0	0	0
0.250	8191	3932	3932	0	0	0	0	0	0	0
0.120	3932	3932	0	0	0	0	0	0	0	0
0.360	11796	3932	3932	3932	0	0	0	0	0	0
0.400	13106	3932	0	0	0	0	0	0	0	0
0.800	26213	3932	3932	0	0	0	0	0	0	0
0.660	21626	3932	0	0	0	0	0	0	0	0
0.660	21626	3604	3604	3604	3604	164	-2293	-2420	896	0
0.660	21626	3604	0	0	0	0	0	0	0	0
0.770	25230	3604	3604	0	0	0	0	0	0	0
0.110	3604	3604	0	0	0	0	0	0	0	0
0.567	18578	3604	3604	3604	0	0	0	0	0	0
0.517	16940	3604	0	0	0	0	0	0	0	0
0.900	29490	3604	3604	0	0	0	0	0	0	0
0.950	31128	3604	0	0	0	0	0	0	0	0
0.100	3276	21626	21626	21626	21626	26377	-11304	5418	5046	0
0.200	6553	21626	0	0	0	0	0	0	0	0
0.250	8191	21626	21626	0	0	0	0	0	0	0
0.120	3932	21626	0	0	0	0	0	0	0	0
0.360	11796	21626	21626	21626	0	0	0	0	0	0
0.400	13106	21626	0	0	0	0	0	0	0	0
0.800	26213	21626	21626	0	0	0	0	0	0	0
0.660	21626	21626	0	0	0	0	0	0	0	0
0.660	21626	31128	31128	31128	31128	-4751	2457	274	134	0
0.660	21626	31128	0	0	0	0	0	0	0	0
0.770	25230	31128	31128	0	0	0	0	0	0	0
0.110	3604	31128	0	0	0	0	0	0	0	0
0.567	18578	31128	31128	31128	0	0	0	0	0	0
0.517	16940	31128	0	0	0	0	0	0	0	0
0.900	29490	31128	31128	0	0	0	0	0	0	0
0.950	31128	31128	0	0	0	0	0	0	0	0

## H.2 Transformada Inversa Rápida de Hartley

- A = Arreglo de prueba representado con 15 bits en formato entero.  
 B = Resultados de la reversión de bits.  
 C = Resultados de la primera mariposa.  
 D = Resultados de la segunda mariposa.  
 E = Resultados de la tercera mariposa.  
 F = Resultados de la cuarta mariposa.  
 G = Resultados de la quinta mariposa.  
 H = Resultados de la sexta mariposa.  
 I = Resultados de la séptima mariposa. FHT terminada.  
 J = Resultados de la FHT terminada representados en forma normalizada (con seis dígitos fraccionarios).  
 K = Valores originales.

A	B	C	D	E	F	G	H	I	J	K
16431	16431	18048	13818	12451	3277	3277	3277	3277	0.100009	0.100
0	1617	14814	14556	14090	6559	6559	6559	6559	0.200171	0.200
0	-2244	-4230	22278	16709	8193	8193	8193	8193	0.250038	0.250
0	-1986	-258	15072	3772	3934	3934	3934	934	0.120060	0.120
0	-1899	-3468	-1367	15185	11795	11795	11795	11795	0.359966	0.360
0	-1569	-330	-8321	15022	13108	13108	13108	13108	0.400037	0.400
0	-29452101		-5569	27847	26210	26210	26210	26210	0.799890	0.800
0	5046	-7991	7661	26372	21626	21626	21626	21626	0.659993	0.660
-4088	-4088	-6152	-6282	-9174	21625	21625	21625	21625	0.659963	0.660
0	-2064	-2024	-3946	-8775	21621	21621	21621	21621	0.659841	0.660
0	-1026	-130	-6022	-7180	25225	25225	25225	25225	0.769829	0.770
0	896	-1922	-102	-1707	3610	3610	3610	3610	0.110172	0.110
0	-2720	-2025	-2892	-3390	18575	18575	18575	18575	0.566881	0.567
0	695	-3415	-4550	883	16936	16936	16936	16936	0.516861	0.517
0	-1001	-867	-1158	-4864	29484	29484	29484	29484	0.899808	0.900
0	134	-1135	-2280	1503	31118	31118	31118	31118	0.949675	0.950
-1899	0	0	0	0	0	3277	3277	3277	0.100009	0.100
0	0	0	0	0	0	6559	6559	6559	0.200171	0.200
0	0	0	0	0	0	8193	8193	8193	0.250038	0.250
0	0	0	0	0	0	3934	3934	3934	0.120060	0.120
0	0	0	0	0	0	11795	11795	11795	0.359966	0.360
0	0	0	0	0	0	13108	13108	13108	0.400037	0.400
0	0	0	0	0	0	26210	26210	26210	0.799890	0.800
0	0	0	0	0	0	21626	21626	21626	0.659993	0.660
-2720	0	0	0	0	0	21625	21625	21625	0.659963	0.660
0	0	0	0	0	0	21621	21621	21621	0.659841	0.660
0	0	0	0	0	0	25225	25225	25225	0.769829	0.770
0	0	0	0	0	0	3610	3610	3610	0.110172	0.110
0	0	0	0	0	0	18575	18575	18575	0.566881	0.567
0	0	0	0	0	0	16936	16936	16936	0.516861	0.517
0	0	0	0	0	0	29484	29484	29484	0.899808	0.900
0	0	0	0	0	0	31118	31118	31118	0.949675	0.950
-2244	0	0	0	0	0	0	3277	3277	0.100009	0.100
0	0	0	0	0	0	0	6559	6559	0.200171	0.200

0	0	0	0	0	0	0	8193	8193	0.250038	0.250
0	0	0	0	0	0	0	3934	3934	0.120060	0.120
0	0	0	0	0	0	0	11795	11795	0.359966	0.360
0	0	0	0	0	0	0	13108	13108	0.400037	0.400
0	0	0	0	0	0	0	26210	26210	0.799890	0.800
0	0	0	0	0	0	0	21626	21626	0.659993	0.660
-1026	0	0	0	0	0	0	21625	21625	0.659963	0.660
0	0	0	0	0	0	0	21621	21621	0.659841	0.660
0	0	0	0	0	0	0	25225	25225	0.769829	0.770
0	0	0	0	0	0	0	3610	3610	0.110172	0.110
0	0	0	0	0	0	0	18575	18575	0.566881	0.567
0	0	0	0	0	0	0	16936	16936	0.516861	0.517
0	0	0	0	0	0	0	29484	29484	0.899808	0.900
0	0	0	0	0	0	0	31118	31118	0.949675	0.950
-2945	0	0	0	0	0	0	3277	3277	0.100009	0.100
0	0	0	0	0	0	0	6559	6559	0.200171	0.200
0	0	0	0	0	0	0	8193	8193	0.250038	0.250
0	0	0	0	0	0	0	3934	3934	0.120060	0.120
0	0	0	0	0	0	0	11795	11795	0.359966	0.360
0	0	0	0	0	0	0	13108	13108	0.400037	0.400
0	0	0	0	0	0	0	26210	26210	0.799890	0.800
0	0	0	0	0	0	0	21626	21626	0.659993	0.660
-1001	0	0	0	0	0	0	21625	21625	0.659963	0.660
0	0	0	0	0	0	0	21621	21621	0.659841	0.660
0	0	0	0	0	0	0	25225	25225	0.769829	0.770
0	0	0	0	0	0	0	3610	3610	0.110172	0.110
0	0	0	0	0	0	0	18575	18575	0.566881	0.567
0	0	0	0	0	0	0	16936	16936	0.516861	0.517
0	0	0	0	0	0	0	29484	29484	0.899808	0.900
0	0	0	0	0	0	0	31118	31118	0.949675	0.950
1617	0	0	0	0	0	0	0	3277	0.100009	0.100
0	0	0	0	0	0	0	0	6559	0.200171	0.200
0	0	0	0	0	0	0	0	8193	0.250038	0.250
0	0	0	0	0	0	0	0	3934	0.120060	0.120
0	0	0	0	0	0	0	0	11795	0.359966	0.360
0	0	0	0	0	0	0	0	13108	0.400037	0.400
0	0	0	0	0	0	0	0	26210	0.799890	0.800
0	0	0	0	0	0	0	0	21626	0.659993	0.660
-2064	0	0	0	0	0	0	0	21625	0.659963	0.660
0	0	0	0	0	0	0	0	21621	0.659841	0.660
0	0	0	0	0	0	0	0	25225	0.769829	0.770
0	0	0	0	0	0	0	0	3610	0.110172	0.110
0	0	0	0	0	0	0	0	18575	0.566881	0.567
0	0	0	0	0	0	0	0	16936	0.516861	0.517
0	0	0	0	0	0	0	0	29484	0.899808	0.900
0	0	0	0	0	0	0	0	31118	0.949675	0.950
-1569	0	0	0	0	0	0	0	3277	0.100009	0.100
0	0	0	0	0	0	0	0	6559	0.200171	0.200
0	0	0	0	0	0	0	0	8193	0.250038	0.250
0	0	0	0	0	0	0	0	3934	0.120060	0.120
0	0	0	0	0	0	0	0	11795	0.359966	0.360

0	0	0	0	0	0	0	0	13108	0.400037	0.400
0	0	0	0	0	0	0	0	26210	0.799890	0.800
0	0	0	0	0	0	0	0	21626	0.659993	0.660
695	0	0	0	0	0	0	0	21625	0.659963	0.660
0	0	0	0	0	0	0	0	21621	0.659841	0.660
0	0	0	0	0	0	0	0	25225	0.769829	0.770
0	0	0	0	0	0	0	0	3610	0.110172	0.110
0	0	0	0	0	0	0	0	18575	0.566881	0.567
0	0	0	0	0	0	0	0	16936	0.516861	0.517
0	0	0	0	0	0	0	0	29484	0.899808	0.900
0	0	0	0	0	0	0	0	31118	0.949675	0.950
-1986	0	0	0	0	0	0	0	3277	0.100009	0.100
0	0	0	0	0	0	0	0	6559	0.200171	0.200
0	0	0	0	0	0	0	0	8193	0.250038	0.250
0	0	0	0	0	0	0	0	3934	0.120060	0.120
0	0	0	0	0	0	0	0	11795	0.359966	0.360
0	0	0	0	0	0	0	0	13108	0.400037	0.400
0	0	0	0	0	0	0	0	26210	0.799890	0.800
0	0	0	0	0	0	0	0	21626	0.659993	0.660
896	0	0	0	0	0	0	0	21625	0.659963	0.660
0	0	0	0	0	0	0	0	21621	0.659841	0.660
0	0	0	0	0	0	0	0	25225	0.769829	0.770
0	0	0	0	0	0	0	0	3610	0.110172	0.110
0	0	0	0	0	0	0	0	18575	0.566881	0.567
0	0	0	0	0	0	0	0	16936	0.516861	0.517
0	0	0	0	0	0	0	0	29484	0.899808	0.900
0	0	0	0	0	0	0	0	31118	0.949675	0.950
5046	0	0	0	0	0	0	0	3277	0.100009	0.100
0	0	0	0	0	0	0	0	6559	0.200171	0.200
0	0	0	0	0	0	0	0	8193	0.250038	0.250
0	0	0	0	0	0	0	0	3934	0.120060	0.120
0	0	0	0	0	0	0	0	11795	0.359966	0.360
0	0	0	0	0	0	0	0	13108	0.400037	0.400
0	0	0	0	0	0	0	0	26210	0.799890	0.800
0	0	0	0	0	0	0	0	21626	0.659993	0.660
134	0	0	0	0	0	0	0	21625	0.659963	0.660
0	0	0	0	0	0	0	0	21621	0.659841	0.660
0	0	0	0	0	0	0	0	25225	0.769829	0.770
0	0	0	0	0	0	0	0	3610	0.110172	0.110
0	0	0	0	0	0	0	0	18575	0.566881	0.567
0	0	0	0	0	0	0	0	16936	0.516861	0.517
0	0	0	0	0	0	0	0	29484	0.899808	0.900
0	0	0	0	0	0	0	0	31118	0.949675	0.950

# Apéndice I

## I.1 Transformada Directa Rápida de Hartley

- A = Arreglo de prueba representado en forma normalizada.  
 B = Arreglo de prueba representado con 15 bits en formato entero.  
 C = Resultados de la reversión de bits.  
 D = Resultados de la primera mariposa.  
 E = Resultados de la segunda mariposa.  
 F = Resultados de la tercera mariposa.  
 G = Resultados de la cuarta mariposa.  
 H = Resultados de la quinta mariposa.  
 I = Resultados de la sexta mariposa.  
 J = Resultados de la séptima mariposa, FHT terminada.

A	B	C	D	E	F	G	H	I	J
0.100	3276	3276	3276	3276	3276	12451	13819	18049	16431
0.200	6553	3276	0	0	0	0	0	0	0
0.250	8191	3276	3276	0	0	0	0	0	0
0.120	3932	3276	0	0	0	0	0	0	0
0.360	11796	3276	3276	3276	0	0	0	0	0
0.400	13106	3276	0	0	0	0	0	0	0
0.800	26213	3276	3276	0	0	0	0	0	0
0.660	21626	3276	0	0	0	0	0	0	0
0.660	21626	21626	21626	21626	21626	-9175	-6283	-6154	-4090
0.660	21626	21626	0	0	0	0	0	0	0
0.770	25230	21626	21626	0	0	0	0	0	0
0.110	3604	21626	0	0	0	0	0	0	0
0.567	18578	21626	21626	21626	0	0	0	0	0
0.517	16940	21626	0	0	0	0	0	0	0
0.900	29490	21626	21626	0	0	0	0	0	0
0.950	31128	21626	0	0	0	0	0	0	0
0.100	3276	11796	11796	11796	11796	15187	-1368	-3470	-1901
0.200	6553	11796	0	0	0	0	0	0	0
0.250	8191	11796	11796	0	0	0	0	0	0
0.120	3932	11796	0	0	0	0	0	0	0
0.360	11796	11796	11796	11796	0	0	0	0	0
0.400	13106	11796	0	0	0	0	0	0	0
0.800	26213	11796	11796	0	0	0	0	0	0
0.660	21626	11796	0	0	0	0	0	0	0
0.660	21626	18578	18578	18578	18578	-3391	-2892	-2026	-2723
0.660	21626	18578	0	0	0	0	0	0	0

0.770	25230	18578	18578	0	0	0	0	0	0
0.110	3604	18578	0	0	0	0	0	0	0
0.567	18578	18578	18578	18578	0	0	0	0	0
0.517	16940	18578	0	0	0	0	0	0	0
0.900	29490	18578	18578	0	0	0	0	0	0
0.950	31128	18578	0	0	0	0	0	0	0
0.100	3276	8191	8191	8191	8191	16710	22280	-4231	-2245
0.200	6553	8191	0	0	0	0	0	0	0
0.250	8191	8191	8191	0	0	0	0	0	0
0.120	3932	8191	0	0	0	0	0	0	0
0.360	11796	8191	8191	8191	0	0	0	0	0
0.400	13106	8191	0	0	0	0	0	0	0
0.800	26213	8191	8191	0	0	0	0	0	0
0.660	21626	8191	0	0	0	0	0	0	0
0.660	21626	25230	25230	25230	25230	-8520	-5080	-130	-1027
0.660	21626	25230	0	0	0	0	0	0	0
0.770	25230	25230	25230	0	0	0	0	0	0
0.110	3604	25230	0	0	0	0	0	0	0
0.567	18578	25230	25230	25230	0	0	0	0	0
0.517	16940	25230	0	0	0	0	0	0	0
0.900	29490	25230	25230	0	0	0	0	0	0
0.950	31128	25230	0	0	0	0	0	0	0
0.100	3276	26213	26213	26213	26213	27851	-5571	2101	-2947
0.200	6553	26213	0	0	0	0	0	0	0
0.250	81912	6213	26213	0	0	0	0	0	0
0.120	39322	6213	0	0	0	0	0	0	0
0.360	11796	6213	26213	26213	0	0	0	0	0
0.400	13106	6213	0	0	0	0	0	0	0
0.800	26213	6213	26213	0	0	0	0	0	0
0.660	21626	6213	0	0	0	0	0	0	0
0.660	21626	29490	29490	29490	29490	-1639	-3441	-867	-1002
0.660	21626	29490	0	0	0	0	0	0	0
0.770	25230	29490	29490	0	0	0	0	0	0
0.110	3604	29490	0	0	0	0	0	0	0
0.567	18578	29490	29490	29490	0	0	0	0	0
0.517	16940	29490	0	0	0	0	0	0	0
0.900	29490	29490	29490	0	0	0	0	0	0
0.950	31128	29490	0	0	0	0	0	0	0
0.100	3276	6553	6553	6553	6553	14089	14556	14814	1617
0.200	6553	6553	0	0	0	0	0	0	0
0.250	8191	6553	6553	0	0	0	0	0	0
0.120	3932	6553	0	0	0	0	0	0	0
0.360	11796	6553	6553	6553	0	0	0	0	0
0.400	13106	6553	0	0	0	0	0	0	0
0.800	26213	6553	6553	0	0	0	0	0	0
0.660	21626	6553	0	0	0	0	0	0	0
0.660	21626	21626	21626	21626	21626	-7537	-4727	-2306	-2065
0.660	21626	21626	0	0	0	0	0	0	0
0.770	25230	21626	21626	0	0	0	0	0	0
0.110	3604	21626	0	0	0	0	0	0	0
0.567	18578	21626	21626	21626	0	0	0	0	0

0.517	16940	21626	0	0	0	0	0	0	0
0.900	29490	21626	21626	0	0	0	0	0	0
0.950	31128	21626	0	0	0	0	0	0	0
0.100	3276	13106	13106	13106	13106	15023	-467	-5886	-1570
0.200	6553	13106	0	0	0	0	0	0	0
0.250	8191	13106	13106	0	0	0	0	0	0
0.120	3932	13106	0	0	0	0	0	0	0
0.360	11796	13106	13106	13106	0	0	0	0	0
0.400	13106	13106	0	0	0	0	0	0	0
0.800	26213	13106	13106	0	0	0	0	0	0
0.660	21626	13106	0	0	0	0	0	0	0
0.660	21626	16940	16940	16940	16940	-1917	-2810	-3085	696
0.660	21626	16940	0	0	0	0	0	0	0
0.770	25230	16940	16940	0	0	0	0	0	0
0.110	3604	16940	0	0	0	0	0	0	0
0.567	18578	16940	16940	16940	0	0	0	0	0
0.517	16940	16940	0	0	0	0	0	0	0
0.900	29490	16940	16940	0	0	0	0	0	0
0.950	31128	16940	0	0	0	0	0	0	0
0.100	3276	3932	3932	3932	3932	3768	15072	-258	-1987
0.200	6553	3932	0	0	0	0	0	0	0
0.250	8191	3932	3932	0	0	0	0	0	0
0.120	3932	3932	0	0	0	0	0	0	0
0.360	11796	3932	3932	3932	0	0	0	0	0
0.400	13106	3932	0	0	0	0	0	0	0
0.800	26213	3932	3932	0	0	0	0	0	0
0.660	21626	3932	0	0	0	0	0	0	0
0.660	21626	3604	3604	3604	3604	164	-2294	-2422	896
0.660	21626	3604	0	0	0	0	0	0	0
0.770	25230	3604	3604	0	0	0	0	0	0
0.110	3604	3604	0	0	0	0	0	0	0
0.567	18578	3604	3604	3604	0	0	0	0	0
0.517	16940	3604	0	0	0	0	0	0	0
0.900	29490	3604	3604	0	0	0	0	0	0
0.950	31128	3604	0	0	0	0	0	0	0
0.100	3276	21626	21626	21626	21626	26377	-11305	5419	5047
0.200	6553	21626	0	0	0	0	0	0	0
0.250	8191	21626	21626	0	0	0	0	0	0
0.120	3932	21626	0	0	0	0	0	0	0
0.360	11796	21626	21626	21626	0	0	0	0	0
0.400	13106	21626	0	0	0	0	0	0	0
0.800	26213	21626	21626	0	0	0	0	0	0
0.660	21626	21626	0	0	0	0	0	0	0
0.660	21626	31128	31128	31128	31128	-4751	2457	274	134
0.660	21626	31128	0	0	0	0	0	0	0
0.770	25230	31128	31128	0	0	0	0	0	0
0.110	3604	31128	0	0	0	0	0	0	0
0.567	18578	31128	31128	31128	0	0	0	0	0
0.517	16940	31128	0	0	0	0	0	0	0
0.900	29490	31128	31128	0	0	0	0	0	0
0.950	31128	31128	0	0	0	0	0	0	0

ESTA TESIS NO DEBE  
SALIR DE LA BIBLIOTECA I-3

## I.2 Transformada Inversa Rápida de Hartley

- A = Arreglo de prueba representado con 15 bits en formato entero.  
 B = Resultados de la reversión de bits.  
 C = Resultados de la primera mariposa.  
 D = Resultados de la segunda mariposa.  
 E = Resultados de la tercera mariposa.  
 F = Resultados de la cuarta mariposa.  
 G = Resultados de la quinta mariposa.  
 H = Resultados de la sexta mariposa.  
 I = Resultados de la séptima mariposa. FHT terminada.  
 J = Resultados de la FHT terminada representados en forma normalizada (con seis dígitos fraccionarios).  
 K = Valores originales.

A	B	C	D	E	F	G	H	I	J	K
16431	16431	18048	13816	12445	3264	3264	3264	3264	0.099612	0.100
0	1617	14814	14556	14087	6548	6548	6548	6548	0.199835	0.200
0	-2245	-4232	22280	16709	8190	8190	8190	8190	0.249946	0.250
0	-1987	-258	15072	3767	3931	3931	3931	3931	0.119968	0.120
0	-1901	-3471	-1371	15187	11796	11796	11796	11796	0.359996	0.360
0	-1570	-331	-8325	15024	13106	13106	13106	13106	0.399975	0.400
0	-29472100		-5571	27851	26212	26212	26212	26212	0.799951	0.800
0	5047	-7994	7663	26376	21625	21625	21625	21625	0.659962	0.660
-4090	-4090	-6155	-6286	-9181	21626	21626	21626	21626	0.659993	0.660
0	-2065	-2025	-3948	-8783	21625	21625	21625	21625	0.659962	0.660
0	-1027	-131	-6024	-7183	25227	25227	25227	25227	0.769890	0.770
0	896	-1923	-102	-1709	3602	3602	3602	3602	0.109927	0.110
0	-2723	-2027	-2895	-3391	18578	18578	18578	18578	0.566972	0.567
0	696	-3419	-4555	886	16941	16941	16941	16941	0.517014	0.517
0	-1002	-868	-1159	-4865	29489	29489	29489	29489	0.899960	0.900
0	134	-1136	-2283	1504	31126	31126	31126	31126	0.949919	0.950
-1901	0	0	0	0	0	3264	3264	3264	0.099612	0.100
0	0	0	0	0	0	6548	6548	6548	0.199835	0.200
0	0	0	0	0	0	8190	8190	8190	0.249946	0.250
0	0	0	0	0	0	3931	3931	3931	0.119968	0.120
0	0	0	0	0	0	11796	11796	11796	0.359996	0.360
0	0	0	0	0	0	13106	13106	13106	0.399975	0.400
0	0	0	0	0	0	26212	26212	26212	0.799951	0.800
0	0	0	0	0	0	21625	21625	21625	0.659962	0.660
-2723	0	0	0	0	0	21626	21626	21626	0.659993	0.660
0	0	0	0	0	0	21625	21625	21625	0.659962	0.660
0	0	0	0	0	0	25227	25227	25227	0.769890	0.770
0	0	0	0	0	0	3602	3602	3602	0.109927	0.110
0	0	0	0	0	0	18578	18578	18578	0.566972	0.567
0	0	0	0	0	0	16941	16941	16941	0.517014	0.517
0	0	0	0	0	0	29489	29489	29489	0.899960	0.900
0	0	0	0	0	0	31126	31126	31126	0.949919	0.950
-2245	0	0	0	0	0	0	3264	3264	0.099612	0.100

0	0	0	0	0	0	0	6548	6548	0.199835	0.200
0	0	0	0	0	0	0	8190	8190	0.249946	0.250
0	0	0	0	0	0	0	3931	3931	0.119968	0.120
0	0	0	0	0	0	0	11796	11796	0.359996	0.360
0	0	0	0	0	0	0	13106	13106	0.399975	0.400
0	0	0	0	0	0	0	26212	26212	0.799951	0.800
0	0	0	0	0	0	0	21625	21625	0.659962	0.660
-1027	0	0	0	0	0	0	21626	21626	0.659993	0.660
0	0	0	0	0	0	0	21625	21625	0.659962	0.660
0	0	0	0	0	0	0	25227	25227	0.769890	0.770
0	0	0	0	0	0	0	3602	3602	0.109927	0.110
0	0	0	0	0	0	0	18578	18578	0.566972	0.567
0	0	0	0	0	0	0	16941	16941	0.517014	0.517
0	0	0	0	0	0	0	29489	29489	0.899960	0.900
0	0	0	0	0	0	0	31126	31126	0.949919	0.950
-2947	0	0	0	0	0	0	3264	3264	0.099612	0.100
0	0	0	0	0	0	0	6548	6548	0.199835	0.200
0	0	0	0	0	0	0	8190	8190	0.249946	0.250
0	0	0	0	0	0	0	39313	931	0.119968	0.120
0	0	0	0	0	0	0	11796	11796	0.359996	0.360
0	0	0	0	0	0	0	13106	13106	0.399975	0.400
0	0	0	0	0	0	0	26212	26212	0.799951	0.800
0	0	0	0	0	0	0	21625	21625	0.659962	0.660
-1002	0	0	0	0	0	0	21626	21626	0.659993	0.660
0	0	0	0	0	0	0	21625	21625	0.659962	0.660
0	0	0	0	0	0	0	25227	25227	0.769890	0.770
0	0	0	0	0	0	0	3602	3602	0.109927	0.110
0	0	0	0	0	0	0	18578	18578	0.566972	0.567
0	0	0	0	0	0	0	16941	16941	0.517014	0.517
0	0	0	0	0	0	0	29489	29489	0.899960	0.900
0	0	0	0	0	0	0	31126	31126	0.949919	0.950
1617	0	0	0	0	0	0	3264	3264	0.099612	0.100
0	0	0	0	0	0	0	0	6548	0.199835	0.200
0	0	0	0	0	0	0	0	8190	0.249946	0.250
0	0	0	0	0	0	0	0	3931	0.119968	0.120
0	0	0	0	0	0	0	0	11796	0.359996	0.360
0	0	0	0	0	0	0	0	13106	0.399975	0.400
0	0	0	0	0	0	0	0	26212	0.799951	0.800
0	0	0	0	0	0	0	0	21625	0.659962	0.660
-2065	0	0	0	0	0	0	0	21626	0.659993	0.660
0	0	0	0	0	0	0	0	21625	0.659962	0.660
0	0	0	0	0	0	0	0	25227	0.769890	0.770
0	0	0	0	0	0	0	0	3602	0.109927	0.110
0	0	0	0	0	0	0	0	18578	0.566972	0.567
0	0	0	0	0	0	0	0	16941	0.517014	0.517
0	0	0	0	0	0	0	0	29489	0.899960	0.900
0	0	0	0	0	0	0	0	31126	0.949919	0.950
-1570	0	0	0	0	0	0	0	3264	0.099612	0.100
0	0	0	0	0	0	0	0	6548	0.199835	0.200
0	0	0	0	0	0	0	0	8190	0.249946	0.250
0	0	0	0	0	0	0	0	3931	0.119968	0.120

0	0	0	0	0	0	0	0	0	11796	0.359996	0.360
0	0	0	0	0	0	0	0	0	13106	0.399975	0.400
0	0	0	0	0	0	0	0	0	26212	0.799951	0.800
0	0	0	0	0	0	0	0	0	21625	0.659962	0.660
696	0	0	0	0	0	0	0	0	21626	0.659993	0.660
0	0	0	0	0	0	0	0	0	21625	0.659962	0.660
0	0	0	0	0	0	0	0	0	25227	0.769890	0.770
0	0	0	0	0	0	0	0	0	3602	0.109927	0.110
0	0	0	0	0	0	0	0	0	18578	0.566972	0.567
0	0	0	0	0	0	0	0	0	16941	0.517014	0.517
0	0	0	0	0	0	0	0	0	29489	0.899960	0.900
0	0	0	0	0	0	0	0	0	31126	0.949919	0.950
-1987	0	0	0	0	0	0	0	0	3264	0.099612	0.100
0	0	0	0	0	0	0	0	0	6548	0.199835	0.200
0	0	0	0	0	0	0	0	0	8190	0.249946	0.250
0	0	0	0	0	0	0	0	0	3931	0.119968	0.120
0	0	0	0	0	0	0	0	0	11796	0.359996	0.360
0	0	0	0	0	0	0	0	0	13106	0.399975	0.400
0	0	0	0	0	0	0	0	0	26212	0.799951	0.800
0	0	0	0	0	0	0	0	0	21625	0.659962	0.660
896	0	0	0	0	0	0	0	0	21626	0.659993	0.660
0	0	0	0	0	0	0	0	0	21625	0.659962	0.660
0	0	0	0	0	0	0	0	0	25227	0.769890	0.770
0	0	0	0	0	0	0	0	0	3602	0.109927	0.110
0	0	0	0	0	0	0	0	0	18578	0.566972	0.567
0	0	0	0	0	0	0	0	0	16941	0.517014	0.517
0	0	0	0	0	0	0	0	0	29489	0.899960	0.900
0	0	0	0	0	0	0	0	0	31126	0.949919	0.950
5047	0	0	0	0	0	0	0	0	3264	0.099612	0.100
0	0	0	0	0	0	0	0	0	6548	0.199835	0.200
0	0	0	0	0	0	0	0	0	8190	0.249946	0.250
0	0	0	0	0	0	0	0	0	3931	0.119968	0.120
0	0	0	0	0	0	0	0	0	11796	0.359996	0.360
0	0	0	0	0	0	0	0	0	13106	0.399975	0.400
0	0	0	0	0	0	0	0	0	26212	0.799951	0.800
0	0	0	0	0	0	0	0	0	21625	0.659962	0.660
134	0	0	0	0	0	0	0	0	21626	0.659993	0.660
0	0	0	0	0	0	0	0	0	21625	0.659962	0.660
0	0	0	0	0	0	0	0	0	25227	0.769890	0.770
0	0	0	0	0	0	0	0	0	3602	0.109927	0.110
0	0	0	0	0	0	0	0	0	18578	0.566972	0.567
0	0	0	0	0	0	0	0	0	16941	0.517014	0.517
0	0	0	0	0	0	0	0	0	29489	0.899960	0.900
0	0	0	0	0	0	0	0	0	31126	0.949919	0.950

# Apéndice J

## J.1 Transformada Directa Rápida de Hartley

- A = Arreglo de prueba representado en forma normalizada.
- B = Arreglo de prueba representada con 15 bits en formato entero.
- C = Resultados de la reversión de bits.
- D = Resultados de la primera mariposa.
- E = Resultados de la segunda mariposa.
- F = Resultados de la tercera mariposa.
- G = Resultados de la cuarta mariposa.
- H = Resultados de la quinta mariposa.
- I = Resultados de la sexta mariposa.
- J = Resultados de la séptima mariposa, FHT terminada.

A	B	C	D	E	F	G	H	I	J
0.100	3276	3276	3276	3276	3276	12451	13819	18049	16431
0.200	6553	3276	0	0	0	0	0	0	0
0.250	8191	3276	3276	0	0	0	0	0	0
0.120	3932	3276	0	0	0	0	0	0	0
0.360	11796	3276	3276	3276	0	0	0	0	0
0.400	13106	3276	0	0	0	0	0	0	0
0.800	26213	3276	3276	0	0	0	0	0	0
0.660	21626	3276	0	0	0	0	0	0	0
0.660	21626	21626	21626	21626	21626	-9175	-6283	-6154	-4090
0.660	21626	21626	0	0	0	0	0	0	0
0.770	25230	21626	21626	0	0	0	0	0	0
0.110	3604	21626	0	0	0	0	0	0	0
0.567	18578	21626	21626	21626	0	0	0	0	0
0.517	16940	21626	0	0	0	0	0	0	0
0.900	29490	21626	21626	0	0	0	0	0	0
0.950	31128	21626	0	0	0	0	0	0	0
0.100	3276	11796	11796	11796	11796	15187	-1368	-3470	-1901
0.200	6553	11796	0	0	0	0	0	0	0
0.250	8191	11796	11796	0	0	0	0	0	0
0.120	3932	11796	0	0	0	0	0	0	0
0.360	11796	11796	11796	11796	0	0	0	0	0
0.400	13106	11796	0	0	0	0	0	0	0
0.800	26213	11796	11796	0	0	0	0	0	0
0.660	21626	11796	0	0	0	0	0	0	0
0.660	21626	18578	18578	18578	18578	-3391	-2892	-2026	-2723
0.660	21626	18578	0	0	0	0	0	0	0

0.770	25230	18578	15788	0	0	0	0	0	0	0
0.110	3604	18578	0	0	0	0	0	0	0	0
0.567	18578	18578	18578	18578	0	0	0	0	0	0
0.517	16940	18578	0	0	0	0	0	0	0	0
0.900	29490	18578	18578	0	0	0	0	0	0	0
0.950	31128	18578	0	0	0	0	0	0	0	0
0.100	3276	8191	8191	8191	8191	16710	22280	-4231	-2245	
0.200	6553	8191	0	0	0	0	0	0	0	
0.250	8191	8191	8191	0	0	0	0	0	0	
0.120	3932	8191	0	0	0	0	0	0	0	
0.360	11796	8191	8191	8191	0	0	0	0	0	
0.400	13106	8191	0	0	0	0	0	0	0	
0.800	26213	8191	8191	0	0	0	0	0	0	
0.660	21626	8191	0	0	0	0	0	0	0	
0.660	21626	25230	25230	25230	25230	-8520	-5080	-130	-1027	
0.660	21626	25230	0	0	0	0	0	0	0	
0.770	25230	25230	25230	0	0	0	0	0	0	
0.110	3604	25230	0	0	0	0	0	0	0	
0.567	18578	25230	25230	25230	0	0	0	0	0	
0.517	16940	25230	0	0	0	0	0	0	0	
0.900	29490	25230	25230	0	0	0	0	0	0	
0.950	31128	25230	0	0	0	0	0	0	0	
0.100	3276	26213	26213	26213	26213	27851	-5571	2101	-2947	
0.200	6553	26213	0	0	0	0	0	0	0	
0.250	8191	26213	26213	0	0	0	0	0	0	
0.120	3932	26213	0	0	0	0	0	0	0	
0.360	11796	26213	26213	26213	0	0	0	0	0	
0.400	13106	26213	0	0	0	0	0	0	0	
0.800	26213	26213	26213	0	0	0	0	0	0	
0.660	21626	26213	0	0	0	0	0	0	0	
0.660	21626	29490	29490	29490	29490	-1639	-3441	-867	-1002	
0.660	21626	29490	0	0	0	0	0	0	0	
0.770	25230	29490	29490	0	0	0	0	0	0	
0.110	3604	29490	0	0	0	0	0	0	0	
0.567	18578	29490	29490	29490	0	0	0	0	0	
0.517	16940	29490	0	0	0	0	0	0	0	
0.900	29490	29490	29490	0	0	0	0	0	0	
0.950	31128	29490	0	0	0	0	0	0	0	
0.100	3276	6553	6553	6553	6553	14089	14556	14814	1617	
0.200	6553	6553	0	0	0	0	0	0	0	
0.250	8191	6553	6553	0	0	0	0	0	0	
0.120	3932	6553	0	0	0	0	0	0	0	
0.360	11796	6553	6553	6553	0	0	0	0	0	
0.400	13106	6553	0	0	0	0	0	0	0	
0.800	26213	6553	6553	0	0	0	0	0	0	
0.660	21626	6553	0	0	0	0	0	0	0	
0.660	21626	21626	21626	21626	21626	-7537	-4727	-2306	-2065	
0.660	21626	21626	0	0	0	0	0	0	0	
0.770	25230	21626	21626	0	0	0	0	0	0	
0.110	3604	21626	0	0	0	0	0	0	0	
0.567	18578	21626	21626	21626	0	0	0	0	0	

0.517	16940	21626	0	0	0	0	0	0	0
0.900	29490	21626	21626	0	0	0	0	0	0
0.950	31128	21626	0	0	0	0	0	0	0
0.100	3276	13106	13106	13106	13106	15023	-467	-5886	-1570
0.200	6553	13106	0	0	0	0	0	0	0
0.250	8191	13106	13106	0	0	0	0	0	0
0.120	3932	13106	0	0	0	0	0	0	0
0.360	11796	13106	13106	13106	0	0	0	0	0
0.400	13106	13106	0	0	0	0	0	0	0
0.800	26213	13106	13106	0	0	0	0	0	0
0.660	21626	13106	0	0	0	0	0	0	0
0.660	21626	16940	16940	16940	-1917	-2810	-3085696		
0.660	21626	16940	0	0	0	0	0	0	0
0.770	25230	16940	16940	0	0	0	0	0	0
0.110	3604	16940	0	0	0	0	0	0	0
0.567	18578	16940	16940	16940	0	0	0	0	0
0.517	16940	16940	0	0	0	0	0	0	0
0.900	29490	16940	16940	0	0	0	0	0	0
0.950	31128	16940	0	0	0	0	0	0	0
0.100	3276	3932	3932	3932	3932	3768	15072	-258	-1987
0.200	6553	3932	0	0	0	0	0	0	0
0.250	8191	3932	3932	0	0	0	0	0	0
0.120	3932	3932	0	0	0	0	0	0	0
0.360	11796	3932	3932	3932	0	0	0	0	0
0.400	13106	3932	0	0	0	0	0	0	0
0.800	26213	3932	3932	0	0	0	0	0	0
0.660	21626	3932	0	0	0	0	0	0	0
0.660	21626	3604	3604	3604	3604	164	-2294	-2422	896
0.660	21626	3604	0	0	0	0	0	0	0
0.770	25230	3604	3604	0	0	0	0	0	0
0.110	3604	3604	0	0	0	0	0	0	0
0.567	18578	3604	3604	3604	0	0	0	0	0
0.517	16940	3604	0	0	0	0	0	0	0
0.900	29490	3604	3604	0	0	0	0	0	0
0.950	31128	3604	0	0	0	0	0	0	0
0.100	3276	21626	21626	21626	21626	26377	-11305	5419	5047
0.200	6553	21626	0	0	0	0	0	0	0
0.250	8191	21626	21626	0	0	0	0	0	0
0.120	3932	21626	0	0	0	0	0	0	0
0.360	11796	21626	21626	21626	0	0	0	0	0
0.400	13106	21626	0	0	0	0	0	0	0
0.800	26213	21626	21626	0	0	0	0	0	0
0.660	21626	21626	0	0	0	0	0	0	0
0.660	21626	31128	31128	31128	31128	-4751	2457	274	134
0.660	21626	31128	0	0	0	0	0	0	0
0.770	25230	31128	31128	0	0	0	0	0	0
0.110	3604	31128	0	0	0	0	0	0	0
0.567	18578	31128	31128	31128	0	0	0	0	0
0.517	16940	31128	0	0	0	0	0	0	0
0.900	29490	31128	31128	0	0	0	0	0	0
0.950	31128	31128	0	0	0	0	0	0	0

## J.2 Transformada Inversa Rápida de Hartley

A = Arreglo de prueba representado con 15 bits en formato entero.

B = Resultados de la reversion de bits.

C = Resultados de la primera mariposa.

D = Resultados de la segunda mariposa.

E = Resultados de la tercera mariposa.

F = Resultados de la cuarta mariposa.

G = Resultados de la quinta mariposa.

H = Resultados de la sexta mariposa.

I = Resultados de la séptima mariposa.FHTterminada.

J = Resultados de la FHT terminada representados en forma normalizada (con seis dígitos fraccionarios).

K = Valores originales.

A	B	C	D	E	F	G	H	I	J	K
16431	16431	18048	13816	12445	3264	3264	3264	3264	0.099612	0.100
0	1617	14814	14556	14087	6548	6548	6548	6548	0.199835	0.200
0	-2245	-4232	22280	16709	8190	8190	8190	8190	0.249946	0.250
0	-1987	-258	15072	3767	3931	3931	3931	3931	0.119968	0.120
0	-1901	-3471	-1371	15187	11796	11796	11796	11796	0.359996	0.360
0	-1570	-331	-8325	15024	13106	13106	13106	13106	0.399975	0.400
0	-2947	2100	-5571	27851	26212	26212	26212	26212	0.799951	0.800
0	5047	-7994	7663	26376	21625	21625	21625	21625	0.659962	0.660
-4090	-4090	-6155	-6286	-9181	21626	21626	21626	21626	0.659993	0.660
0	-2065	-2025	-3948	-8783	21625	21625	21625	21625	0.659962	0.660
0	-1027	-131	-6024	-7183	25227	25227	25227	25227	0.769890	0.770
0	896	-1923	-102	-1709	3602	3602	3602	3602	0.109927	0.110
0	-2723	-2027	-2895	-3391	18578	18578	18578	18578	0.566972	0.567
0	696	-3419	-4555	886	16941	16941	16941	16941	0.517014	0.517
0	-1002	-868	-1159	-4865	29489	29489	29489	29489	0.899960	0.900
0	134	-1136	-2283	1504	31126	31126	31126	31126	0.949919	0.950
-1901	0	0	0	0	0	3264	3264	3264	0.099612	0.100
0	0	0	0	0	0	6548	6548	6548	0.199835	0.200
0	0	0	0	0	0	8190	8190	8190	0.249946	0.250
0	0	0	0	0	0	3931	3931	3931	0.119968	0.120
0	0	0	0	0	0	11796	11796	11796	0.359996	0.360
0	0	0	0	0	0	13106	13106	13106	0.399975	0.400
0	0	0	0	0	0	26212	26212	26212	0.799951	0.800
0	0	0	0	0	0	21625	21625	21625	0.659962	0.660
-2723	0	0	0	0	0	21626	21626	21626	0.659993	0.660
0	0	0	0	0	0	21625	21625	21625	0.659962	0.660
0	0	0	0	0	0	25227	25227	25227	0.769890	0.770
0	0	0	0	0	0	3602	3602	3602	0.109927	0.110
0	0	0	0	0	0	18578	18578	18578	0.566972	0.567
0	0	0	0	0	0	16941	16941	16941	0.517014	0.517
0	0	0	0	0	0	29489	29489	29489	0.899960	0.900
0	0	0	0	0	0	31126	31126	31126	0.949919	0.950

-2245	0	0	0	0	0	0	3264	3264	0.099612	0.100
0	0	0	0	0	0	0	6548	6548	0.199835	0.200
0	0	0	0	0	0	0	8190	8190	0.249946	0.250
0	0	0	0	0	0	0	3931	3931	0.119968	0.120
0	0	0	0	0	0	0	11796	11796	0.359996	0.360
0	0	0	0	0	0	0	13106	13106	0.399975	0.400
0	0	0	0	0	0	0	26212	26212	0.799951	0.800
0	0	0	0	0	0	0	21625	21625	0.659962	0.660
-1027	0	0	0	0	0	0	21626	21626	0.659993	0.660
0	0	0	0	0	0	0	21625	21625	0.659962	0.660
0	0	0	0	0	0	0	25227	25227	0.769890	0.770
0	0	0	0	0	0	0	3602	3602	0.109927	0.110
0	0	0	0	0	0	0	18578	18578	0.566972	0.567
0	0	0	0	0	0	0	16941	16941	0.517014	0.517
0	0	0	0	0	0	0	29489	29489	0.899960	0.900
0	0	0	0	0	0	0	31126	31126	0.949919	0.950
-2947	0	0	0	0	0	0	3264	3264	0.099612	0.100
0	0	0	0	0	0	0	6548	6548	0.199835	0.200
0	0	0	0	0	0	0	8190	8190	0.249946	0.250
0	0	0	0	0	0	0	3931	3931	0.119968	0.120
0	0	0	0	0	0	0	11796	11796	0.359996	0.360
0	0	0	0	0	0	0	13106	13106	0.399975	0.400
0	0	0	0	0	0	0	26212	26212	0.799951	0.800
0	0	0	0	0	0	0	21625	21625	0.659962	0.660
-1002	0	0	0	0	0	0	21626	21626	0.659993	0.660
0	0	0	0	0	0	0	21625	21625	0.659962	0.660
0	0	0	0	0	0	0	25227	25227	0.769890	0.770
0	0	0	0	0	0	0	3602	3602	0.109927	0.110
0	0	0	0	0	0	0	18578	18578	0.566972	0.567
0	0	0	0	0	0	0	16941	16941	0.517014	0.517
0	0	0	0	0	0	0	29489	29489	0.899960	0.900
0	0	0	0	0	0	0	31126	31126	0.949919	0.950
1617	0	0	0	0	0	0	0	3264	0.099612	0.100
0	0	0	0	0	0	0	0	6548	0.199835	0.200
0	0	0	0	0	0	0	0	8190	0.249946	0.250
0	0	0	0	0	0	0	0	3931	0.119968	0.120
0	0	0	0	0	0	0	0	11796	0.359996	0.360
0	0	0	0	0	0	0	0	13106	0.399975	0.400
0	0	0	0	0	0	0	0	26212	0.799951	0.800
0	0	0	0	0	0	0	0	21625	0.659962	0.660
-2065	0	0	0	0	0	0	21626	21626	0.659993	0.660
0	0	0	0	0	0	0	21625	21625	0.659962	0.660
0	0	0	0	0	0	0	25227	25227	0.769890	0.770
0	0	0	0	0	0	0	3602	3602	0.109927	0.110
0	0	0	0	0	0	0	18578	18578	0.566972	0.567
0	0	0	0	0	0	0	16941	16941	0.517014	0.517
0	0	0	0	0	0	0	29489	29489	0.899960	0.900
0	0	0	0	0	0	0	31126	31126	0.949919	0.950
-1570	0	0	0	0	0	0	0	3264	0.099612	0.100
0	0	0	0	0	0	0	0	6548	0.199835	0.200
0	0	0	0	0	0	0	0	8190	0.249946	0.250

0	0	0	0	0	0	0	0	3931	0.119968	0.120
0	0	0	0	0	0	0	0	11796	0.359996	0.360
0	0	0	0	0	0	0	0	13106	0.399975	0.400
0	0	0	0	0	0	0	0	26212	0.799951	0.800
0	0	0	0	0	0	0	0	21625	0.659962	0.660
696	0	0	0	0	0	0	0	21626	0.659993	0.660
0	0	0	0	0	0	0	0	21625	0.659962	0.660
0	0	0	0	0	0	0	0	25227	0.769890	0.770
0	0	0	0	0	0	0	0	3602	0.109927	0.110
0	0	0	0	0	0	0	0	18578	0.566972	0.567
0	0	0	0	0	0	0	0	16941	0.517014	0.517
0	0	0	0	0	0	0	0	29489	0.899960	0.900
0	0	0	0	0	0	0	0	31126	0.949919	0.950
-1987	0	0	0	0	0	0	0	3264	0.099612	0.100
0	0	0	0	0	0	0	0	6548	0.199835	0.200
0	0	0	0	0	0	0	0	8190	0.249946	0.250
0	0	0	0	0	0	0	0	3931	0.119968	0.120
0	0	0	0	0	0	0	0	11796	0.359996	0.360
0	0	0	0	0	0	0	0	13106	0.399975	0.400
0	0	0	0	0	0	0	0	26212	0.799951	0.800
0	0	0	0	0	0	0	0	21625	0.659962	0.660
896	0	0	0	0	0	0	0	21626	0.659993	0.660
0	0	0	0	0	0	0	0	21625	0.659962	0.660
0	0	0	0	0	0	0	0	25227	0.769890	0.770
0	0	0	0	0	0	0	0	3602	0.109927	0.110
0	0	0	0	0	0	0	0	18578	0.566972	0.567
0	0	0	0	0	0	0	0	16941	0.517014	0.517
0	0	0	0	0	0	0	0	29489	0.899960	0.900
0	0	0	0	0	0	0	0	31126	0.949919	0.950
5047	0	0	0	0	0	0	0	3264	0.099612	0.100
0	0	0	0	0	0	0	0	6548	0.199835	0.200
0	0	0	0	0	0	0	0	8190	0.249946	0.250
0	0	0	0	0	0	0	0	3931	0.119968	0.120
0	0	0	0	0	0	0	0	11796	0.359996	0.360
0	0	0	0	0	0	0	0	13106	0.399975	0.400
0	0	0	0	0	0	0	0	26212	0.799951	0.800
0	0	0	0	0	0	0	0	21625	0.659962	0.660
134	0	0	0	0	0	0	0	21626	0.659993	0.660
0	0	0	0	0	0	0	0	21625	0.659962	0.660
0	0	0	0	0	0	0	0	25227	0.769890	0.770
0	0	0	0	0	0	0	0	3602	0.109927	0.110
0	0	0	0	0	0	0	0	18578	0.566972	0.567
0	0	0	0	0	0	0	0	16941	0.517014	0.517
0	0	0	0	0	0	0	0	29489	0.899960	0.900
0	0	0	0	0	0	0	0	31126	0.949919	0.950