

**CÁLCULO DEL ESPECTRO DE IMPEDANCIA ELÉCTRICA DE TEJIDO DE
CUELLO UTERINO POR TRANSFORMADA RÁPIDA DE FOURIER
IMPLEMENTANDO LA TEORÍA DE MUESTREO POR DESFASE EN UN DSP
MOTOROLA DE LA FAMILIA 56800.**



***LUDWING DARIO GARCÍA MATEUS
FRANCISCO CLEMENTE VALLE DIAZ
Dir. JAIME GUILLERMO BARRERO PÉREZ
Codir DAVID ALEJANDRO MIRANDA MERCADO***



***UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES
BUCARAMANGA
2006***

**CÁLCULO DEL ESPECTRO DE IMPEDANCIA ELÉCTRICA DE TEJIDO DE
CUELLO UTERINO POR TRANSFORMADA RÁPIDA DE FOURIER
IMPLEMENTANDO LA TEORÍA DE MUESTREO POR DESFASE EN UN DSP
MOTOROLA DE LA FAMILIA 56800.**

**LUDWING DARIO GARCÍA MATEUS
FRANCISCO CLEMENTE VALLE DIAZ**

Proyecto de grado presentado para optar por el título de Ingeniero Electrónico

**Dir. JAIME GUILLERMO BARRERO PÉREZ
Codir DAVID ALEJANDRO MIRANDA MERCADO**



**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y
TELECOMUNICACIONES
BUCARAMANGA**

2005

A Dios por abrirme una puerta después de que otra se cerrará,
A mi madre por ser mi pilar y mi fuerza... gracias a ti llegue hasta acá.
De mi familia especialmente a Nata, Camilo, Apu, nona, Mary, Nancy² y Claudio y
en general a la familia Mateus por su apoyo incondicional.
A Liliana, porque su amor y paciencia transformó mi manera de enfrentar la vida.
A mis amigos Pacho, Armando, Oscar, Arnold, Tatiana, Angélica, Jerson, Alba,
Fidel, Astrid, Nicolay, Rudy, Pedro y Karen.
Gracias David por ser un maestro y hacer posible lo que parecía imposible.

Ludwing Garcia Mateus

A Dios por que gracias a el todo se puede
A mi madre por que todo lo que soy es gracias a ella
A mis tíos y tías por que me han enseñado que el amor de la familia es la fortaleza
necesaria para enfrentar el día a día
A mis hermanos, primos y sobrinos que son las razones para seguir adelante
A mis amigos que han estado con migo en las buenas y en las malas Ludwing,
Arnol, Robiel, Armando, Oscar, Chicho, Fidel, Astrid, Yudith, Jerson, Angelica,
Omar.
A David por ser amigo y maestro

Francisco C. Valle Díaz

*Dios no nos hubiera dado la posibilidad de soñar, si no nos hubiera dado la
oportunidad de hacer realidad nuestros sueños*

Autores

TABLA DE CONTENIDO

LISTA DE FIGURAS	8
LISTA DE ANEXOS	12
INTRODUCCIÓN	16
1. FUNDAMENTACIÓN TEÓRICA	18
1.1. ESPETRO DE IMPEDANCIA ELÉCTRICA	18
1.1.1. <i>Medición del espectro de impedancia eléctrica</i>	19
1.1.2. <i>Resultados obtenidos</i>	20
1.2. TEORÍA DE MUESTREO POR DESFASE (TMD)	22
1.2.1. <i>Planteamiento de algoritmos para el muestreo de señales basados en la TMD</i>	26
1.2.2. <i>Consideraciones en la aplicación de la TMD</i>	27
1.3. ANÁLISIS DE ERROR PARA SEÑALES MUESTREADAS	29
1.3.1. <i>Variación de error para reconstrucción de señales en la TMD</i>	29
1.3.2. <i>Ponderación de error</i>	33
1.3.2.1. <i>Promediado de n veces la señal adquirida</i>	33
1.4. TRANSFORMADA RÁPIDA DE FOURIER	35
1.5. PROCESADOR DIGITAL DE SEÑALES	39
2. IMPLEMENTACIÓN DE LA FFT Y LA TMD EN MATLAB	41
2.1. IMPLEMENTACIÓN DE LA TMD EN MATLAB	41
2.2. IMPLEMENTACIÓN DE UNA FFT EN MATLAB	43
2.2.1. <i>Selección del tipo de código a implementar</i>	43
2.2.2. <i>Implementación del código de la FFT con base en el algoritmo de diezrado en el tiempo</i>	44
2.2.2.1. <i>Organización de la secuencia de entrada en orden de bit inverso</i>	44
2.2.2.2. <i>Implementación de la transformada discreta de Fourier</i>	48
3. IMPLEMENTACIÓN DE LA TMD Y FFT EN EL PROCESADOR DIGITAL DE SEÑALES	51
3.1. IMPLEMENTACIÓN DE LA TEORÍA DE MUESTREO POR DESFASE (TMD) EN EL PROCESADOR DIGITAL DE SEÑALES	51
3.1.1. <i>Planteamiento del problema de TMD para medición de espectro de impedancia eléctrica</i>	51
3.1.2. <i>Diagrama de flujo del algoritmo implementado para la TMD</i>	52
3.1.3. <i>Implementación de la TMD en el procesador digital de señales</i>	54
3.2. IMPLEMENTACIÓN DEL ALGORITMO PARA LA OBTENCIÓN DE LA TRANSFORMADA RÁPIDA DE FOURIER (FFT) DE UNA SECUENCIA DE DATOS EN EL PROCESADOR DIGITAL DE SEÑALES	57
3.2.1. <i>Planteamiento del problema de la Transformada Rápida de Fourier (FFT) para la medición del espectro de impedancia eléctrica</i>	57

3.2.2. Diagrama de flujo del algoritmo implementado para la Transformada Rápida de Fourier (FFT)	60
3.2.3. Implementación de la FFT en el procesador digital de señales	62
4. RESULTADOS Y ANÁLISIS DE RESULTADOS	67
4.1. TRANSFORMADA RÁPIDA DE FOURIER	67
4.2. TEORÍA DE MUESTREO POR DESFASE	74
4.2.1 Muestreo de la señal triangular	76
4.2.2. Muestreo de la señal senoidal	79
4.2.3. Corrección del error generado por el muestreo	81
4.3. ESPECTRO DE IMPEDANCIA ELÉCTRICA	87
4.3.1. Método usado para la medición del EIE	88
4.3.2. Impedancia de transferencia aplicando Transformada Rápida De Fourier	90
4.3.3. Pruebas de EIE practicadas usando el método propuesto	90
5. CONCLUSIONES	93
5.1. TRANSFORMADA RÁPIDA DE FOURIER	93
5.2. TEORÍA DE MUESTREO POR DESFASE	94
5.3. ANÁLISIS DE ERROR PARA SEÑALES MUESTREADAS	94
5.4. ESPECTRO DE IMPEDANCIA ELÉCTRICA	95
5.5. FORMACIÓN PERSONAL	95
6. RECOMENDACIONES	96
6.1. ACERCA DE CODEWARRIOR 6.1	97
6.2. TEORÍA DE MUESTREO POR DESFASE TMD	97
6.3. CÁLCULO DEL ESPECTRO DE IMPEDANCIA ELÉCTRICO EN TEJIDO CERVICAL IMPLEMENTANDO LA TMD	98

LISTA DE FIGURAS

FIGURA 1. MODELACIÓN CIRCUITAL DE TEJIDO DE UN SOLO LAZO.....	20
FIGURA 2. MODELO CIRCUITAL DE RABBAT PARA UN SISTEMA FISIOLÓGICO UNICELULAR.....	20
FIGURA 3.A. MAGNITUD DE LA IMPEDANCIA OBTENIDA A UN MODELO CIRCUITAL DE TEJIDO DE RESISTENCIA Y CAPACITOR EN PARALELO CON EL MÉTODO DEL PULSO TRANSITORIO.....	21
FIGURA 3.B. IMPEDANCIA DE TEJIDO QUE CONTIENEN UNA MEZCLA DE FIBRAS Y GRASA.....	22
FIGURA 4. SEÑAL ANALÓGICA Y PERIÓDICA.....	23
FIGURA 5. SEÑAL MUESTREADA A UNA MUESTRA POR PERÍODO DE SEÑAL MEDIANTE LA TMD.....	23
FIGURA 6. MUESTRAS DE LA SEÑAL SEPARADAS POR FOTOGRAMAS....	24
FIGURA 7. RECONSTRUCCIÓN DE UN PERIODO DE LA SEÑAL A PARTIR DE LOS FOTOGRAMAS DE LA.....	28
FIGURA 8. SEÑAL ANALÓGICA MUESTREADA CON UN PERIODO DE MUESTREO DE LA DE LA TMD MÚLTIPLO ENTERO DEL PERIODO DE LA SEÑAL ANALÓGICA (PERIODO DE MUESTREO = 6 Y PERIODO DE LA SEÑAL = 2).....	30
FIGURA 9a. OSCILACIÓN DE UNA ONDA PROVOCADA POR INESTABILIDAD EN FRECUENCIA DE LA FUENTE GENERADORA.....	31
FIGURA 9b. AMPLIACIÓN OSCILACIÓN DE UNA ONDA PROVOCADA POR INESTABILIDAD EN FRECUENCIA DE LA FUENTE GENERADORA.....	36
FIGURA 10. FUNCIÓN ESCALÓN.....	36
FIGURA 11. TRANSFORMADA DE FOURIER DE LA FUNCIÓN ESCALÓN.....	37

FIGURA 12. MUESTRAS DE LA TRANSFORMADA DE FOURIER DE LA SEÑAL ESCALÓN.....	37
FIGURA 13. MUESTRAS EQUIESPACIADAS DE LA SEÑAL ESCALÓN.....	38
FIGURA 14. TRANSFORMADA DISCRETA DE FOURIER.....	45
FIGURA 15. ORGANIZACIÓN DE LA SECUENCIA DE ENTRADA EN ORDEN DE BIT INVERSO ANTES DE INGRESAR AL BLOQUE DE FFT.....	45
FIGURA 16. SECUENCIA DE ENTRADA EN ORDEN NORMAL ANTES DE INGRESAR AL BLOQUE DE FFT.....	46
FIGURA 17. ORDENAMIENTO DE UNA SECUENCIA EN ORDEN DE BIT INVERSO.....	49
FIGURA 18. ESTRUCTURA GENERAL DEL CÓDIGO ESCRITO PARA OBTENER LA FFT EN EL SIMULADOR MATLAB.....	53
FIGURA 19. DIAGRAMA DE FLUJO DEL CÓDIGO IMPLEMENTADO PARA LA TMD.....	55
FIGURA 20. BEANS ADC Y DE PROPÓSITO GENERAL PARA LA ADQUISICIÓN DE MUESTRAS.....	55
FIGURA 21. AJUSTE DEL TIEMPO DE MUESTREO.....	56
FIGURA 22. RUTINA PARA EL MUESTREO DE DATOS.....	59
FIGURA 23. DIAGRAMA DE BLOQUES PARA LA OBTENCIÓN DEL ESPECTRO DE IMPEDANCIA ELÉCTRICA.....	59
FIGURA 25. DIAGRAMA DE FLUJO DEL CÓDIGO IMPLEMENTADO PARA LA FFT.....	61
FIGURA 26. CODEWARRIOR EN MODO SIMULACIÓN PARA LA FAMILIA 56800.....	62
FIGURA 27. BEANS USADOS PARA EJECUTAR LAS FUNCIONES SENO Y COSENO.....	63
FIGURA 28. COMPILACIÓN DEL PROYECTO.....	63
FIGURA 29. SIMULACIÓN EN MODO SIMULADOR.....	64

FIGURA 30. VECTOR DE RESULTADOS PARA UNA FFT DE OCHO PUNTOS CUYA SECUENCIA DE ENTRADA ES UNA RAMPA DE OCHO MUESTRAS.....	65
FIGURA 31. SIMULACIÓN EN LÍNEA CON EL PROCESADOR DIGITAL DE SEÑALES.....	66
FIGURA 32. TRASFORMADA DE FOURIER DE UNA RAMPA DE 64 DATOS....	67
FIGURA 33. TRANSFORMADA DE FOURIER DE UNA RAMPA DE 64 DATOS..	68
FIGURA 34. TRANSFORMADA DE FOURIER DE UNA RAMPA DE 128 DATOS.....	69
FIGURA 35. SEÑAL DE ENTRADA.....	69
FIGURA 36. PORCENTAJE DE ERROR ENTRE LA FFT OBTENIDA MEDIANTE EL COMANDO BASE Y EL CÓDIGO DE FFT ESCRITO EN MATLAB.	70
FIGURA 37. ESTA GRÁFICA REPRESENTA EL ERROR DEL ALGORITMO FFT_128 COMPARADO CON EL COMANDO BASE DE MATLAB, CUANDO LA SEÑAL DE ENTRADA ES UNA RAMPA.....	71
FIGURA 38. GRÁFICA DE ERROR PORCENTUAL LA USAR EL BEAN DFR1:DSP_FUNC_DFR COMPARADO CON EL COMANDO BASE DE MATLAB, CUANDO LA SEÑAL DE ENTRADA ES UNA RAMPA.....	72
FIGURA 39. GRÁFICA DE ERROR PORCENTUAL DEL ALGORITMO FFT_64..	73
FIGURA 40. DATOS MUESTREADOS.....	76
FIGURA 41. DATOS DE TENSIÓN ORGANIZADOS.....	77
FIGURA 42. GRAFICA DE SEÑAL RECONSTRUIDA EXPORTANDO LAS MUESTRAS A LA HERRAMIENTA DE SIMULACIÓN MATLAB 7.0.....	78
FIGURA 43. DATOS MUESTREADOS.....	79

FIGURA 44. DATOS DE TENSIÓN ORGANIZADOS.....	80
FIGURA 45. GRAFICA DE SEÑAL RECONSTRUIDA EXPORTANDO LAS MUESTRAS A LA HERRAMIENTA DE SIMULACIÓN MATLAB 7.0.....	81
FIGURA 46. MUESTREOS DE LA SEÑAL SENOIDAL DE PERIODO 5,58MS Y DE AMPLITUD 2VPP.....	82
FIGURA 47. ORDENAMIENTO DE LA ONDA1 DE LA FIGURA 46 POR ALGORITMO DE DETECCIÓN DE FLANCO.....	83
FIGURA 48. SEÑAL OBTENIDA DESPUÉS DE LA SUMA DE PONDERACIÓN..	84
FIGURA 49. COMPARACIÓN GRÁFICA DE UNA SEÑAL PONDERADA CON UNA MUESTREADA.....	85
FIGURA 50. SIMULACIÓN DE MUESTREO.....	86
FIGURA 51. COMPARACIÓN DE LA SEÑAL TEÓRICA CON LA SEÑAL PONDERADA.....	87
FIGURA 52. MODELO CIRCUITAL DE TEJIDO DE CERVIX.....	89
FIGURA 53. ESPECTRO DE IMPEDANCIA ELÉCTRICA OBTENIDA A UN MODELO DE TEJIDO CERVICAL.....	91

LISTA DE ANEXOS

- Anexo A1. Teoría de muestreo por desfase (*TMD*)
- Anexo A2. Análisis en frecuencia de la Teoría de muestreo por desfase(*TMD*)
- Anexo B. Códigos implementados en Matlab
- Anexo C. Códigos implementados en *Codewarrior*

RESUMEN

TÍTULO:

*CÁLCULO DEL ESPECTRO DE IMPEDANCIA ELÉCTRICA DE TEJIDO DE CUELLO UTERINO POR TRANSFORMADA RÁPIDA DE FOURIER IMPLEMENTANDO LA TEORÍA DE MUESTREO POR DESFASE EN UN DSP MOTOROLA DE LA FAMILIA 56F800**

AUTORES:

*GARCIA MATEUS, Ludwing Dario, VALLE DÍAZ, Francisco Clemente***

PALABRAS CLAVE:

Impedancia eléctrica, Transformada Rápida de Fourier, teoría de Muestreo por Desfase, cervix, tejido de cuello uterino.

DESCRIPCIÓN:

Se presentan los resultados del cálculo del espectro de impedancia eléctrica por Transformada Rápida de Fourier a un modelo circuital de tejido de cuello uterino, así como la aplicación de la teoría de Muestreo por Desfase para el muestreo de señales, y la formulación y aplicación de dos métodos de corrección de error para señales muestreadas.

Para realizar el cálculo del espectro de impedancia eléctrica en tejido de cuello uterino, se diseñó un algoritmo (software) que permite adquirir y procesar datos de voltaje y corriente obtenidos del tejido de cervix por medio del método de los cuatro electrodos. Este algoritmo se puede implementar en cualquier DSP de la familia 56F80X siempre que su capacidad de memoria de datos y memoria de programas lo permita.

El algoritmo diseñado consta de tres etapas, la primera realiza la adquisición de datos de tensión y corriente implementado la teoría de muestreo por desfase, con lo cual se hace posible el muestreo de señales a frecuencias inclusive por debajo de la frecuencia máxima de ésta, la segunda realiza la Transformada de Fourier de los datos adquiridos, y la tercera etapa la obtención de la impedancia de transferencia del tejido de cuello uterino.

La validación del software diseñado se realizó implementando el algoritmo creado en un DSP56F801 de Motorola, y las pruebas se practicaron sobre un modelo circuital de tejido.

* Proyecto de grado

** Facultad de Físico Mecánicas, Programa de Ingeniería Electrónica, Director: Jaime Guillermo Barreo Pérez, Ing. Mpe, Codirector: David Alejandro Miranda Mercado, Ing. & Físico Msc.

Como aporte final se propuso dos métodos de corrección de error para señales muestreadas, el primero de ellos aplicando aproximación por derivación y el segundo basado en la ponderación de muestreos de la señal.

ABSTRACT

TITLE:

ELECTRICAL IMPEDANCE SPECTRUM CALCULATE OF CERVICAL TISSUE BY FAST FOURIER TRANSFORM IMPLEMENTING OFF-SET SAMPLING THEORY ON 56F800 MOTOROLA FAMILY DSP

AUTHORS:

GARCIA MATEUS, Ludwing Dario, VALLE DÍAZ, Francisco Clemente **

KEY WORDS:

Electrical Impedance, Fast Fourier Transform, Off-set Sampling Theory, cervix, cervical tissue.

DESCRIPTION:

The results of electrical impedance spectrum calculate by Fast Fourier Transform to a cervix tissue electrical model, as well as the application of the Off-set Sampling theory for sampling signals, and formulation and application of tow off-set correction methods for sampled signals are presented in this work.

To make calculation of electrical impedance spectrum in cervical tissue, an algorithm (software) for acquisition and processing samples on current and tension obtained with the four electrodes method was designed. The algorithm can be implemented in any 56F800 family DSP whether its data memory and code memory allows that.

The designed algorithm has three parts, the first one makes the acquisition of current and tension samples through the Off-Set Sampling theory, because of that is possible sampling signals inclusive whit sampling frequencies down of the highest signal frequency, the second one makes the Fast Fourier Transform of the acquired dataset, and the last one allows the performance of cervix tissue transference impedance.

For designed software validation the created algorithm was implemented in a 56F801DSP, and the probes were made in a tissue circuital model.

Finally two off-set correction methods for sampled signals were proposed, one of this using approximation by derivation, and the other based in samples signal average.

** Facultad de Físico Mecánicas, Programa de Ingeniería Electrónica, Director: Jaime Guillermo Barreo Pérez, Ing. Mpe, Codirector: David Alejandro Miranda Mercado, Ing. & Físico Msc.

INTRODUCCIÓN

El cáncer de tejido de cuello uterino es la primera causa de muerte de la población femenina en Santander, y una de las principales causas de muerte de la población femenina mundial. Entre los motivos por los cuales esta enfermedad sigue cobrando vidas se encuentran los largos periodos de tiempo (alrededor de una semana) que toma en arrojar resultados un examen de detección como la citología y el costo que implica este tipo de pruebas. En respuesta a este problema social, la Universidad Industrial de Santander a través de un grupo interdisciplinario que involucra a la E3T*, la facultad de Salud y la escuela de Física, esta llevando a cabo estudios que permitan desarrollar un método de detección precoz de cáncer de cerviz, basado en el cálculo del espectro de impedancia eléctrica que permita obtener resultados en un tiempo no mayor a veinte minutos y cuyo costo sea muy pequeño comparado con exámenes como la citología.

Uno de los puntos clave para hacer realidad este método de detección precoz, radica en lograr la implementación de un método para el cálculo del espectro de impedancia eléctrica, que a través de una sola medición, permita conocer el espectro en un rango de frecuencias de interés para diagnosticar el estado del tejido cervical.

El objetivo de este trabajo fue desarrollar un método para el cálculo del espectro de impedancia eléctrica en tejido de cuello uterino con los requerimientos mencionados, haciendo uso de herramientas como la Transformada Rápida de Fourier y la teoría de muestreo por desfase.

* E3T se entiende como escuela de Ingeniería Eléctrica Electrónica y de Telecomunicaciones

El desarrollo del método para cálculo del espectro de impedancia eléctrica (EIE) se trata a lo largo de cinco capítulos. El primer capítulo abarca la fundamentación teórica donde se trata el EIE, teoría de muestreo por desfase, métodos de corrección de error, la Transformada Rápida de Fourier (*FFT*) y el procesador digital de señales.

Las implementaciones de prueba, previas a las implementaciones finales, se hacen en la herramienta de simulación Matlab versión 7.0. Estas se tratan en el capítulo dos.

En el capítulo tres se abordan las implementaciones finales que se realizan en la herramienta de programación de DSPs Codewarrior versión 6.1. En este numeral se desarrollan los códigos para obtener la FFT y para la aplicación de la TMD al muestreo de señales.

Las pruebas realizadas y el análisis de resultados se exponen en el capítulo cuarto, donde adicionalmente se aplica el método de ponderación de error para la corrección del error generado al muestrear señales.

Las conclusiones se describen en el capítulo cinco y en el capítulo seis se hacen las recomendaciones para futuros trabajos basados en los logros obtenidos en éste.

1. FUNDAMENTACIÓN TEÓRICA

En este capítulo se presentan conceptos claves para una adecuada comprensión del trabajo documentado en el presente libro, además sirve al lector para ubicarse en un contexto que permite un entendimiento de los resultados y las conclusiones planteadas por los autores.

1.1. ESPETRO DE IMPEDANCIA ELÉCTRICA

En este numeral se da una breve definición sobre espectro de impedancia eléctrica y a que se debe. Además se describe la forma de medirlo y algunos resultados obtenidos en esta área, que son de importancia para el trabajo que se desarrolla en el presente documento.

El espectro de impedancia es la resistividad eléctrica hallada a través de una excitación a diferentes frecuencias. Este concepto puede ser usado para el análisis de bioimpedancia¹, el cual permite analizar el estado del tejido biológico frente a ciertas patologías. Esto debido a que las características físicas, biológicas y eléctricas de las células cambian cuando se ven afectadas por estas patologías. Por ejemplo cuando el tejido de cervix enfrenta los diferentes estadios de cáncer, cambia la forma y distribución de las células que componen dicho tejido, haciendo que cambie a la vez su espectro de impedancia eléctrica^{2,3}.

¹ LOZANO –NIETO A., REZYWOWSKI D., “*Electrical Models For Bioimpedance Measurements*”, Penn State University, Commonwealth College, Wilkes - Barre Campus

² MIRANDA D., “*Detección Precoz De Cáncer De Cuello Uterino basada En Espectro De Impedancia Eléctrica*”, Trabajo de Maestría en ingeniería Grado, Universidad Industrial de Santander, 2005

³ ROBLES A., CHAVEZ F., “*MEDICIÓN DE IMPEDANCIA ELÉCTRICA EN TEJIDO CERVICO UTERINO IN-VITRO*”, Trabajo de Grado, Universidad Industrial de Santander, 2005

1.1.1. Medición del espectro de impedancia eléctrica

Dentro de los métodos más usados para la medición del espectro de impedancia eléctrica en tejidos, se encuentra el método de los cuatro electrodos⁴, en el cual se emplea (por lo general) una punta de medición compuesta por cuatro electrodos donde dos de estos sirven para inyectar una corriente de características conocidas, y los dos restantes permiten medir la tensión diferencial que esta corriente produce al atravesar el tejido que es objeto de estudio. Los datos tomados a través de esta técnica son promediados y después ajustados al modelo de Cole-Cole para hallar los parámetros de impedancia a bajas y altas frecuencias y el valor de capacitancia del tejido⁵.

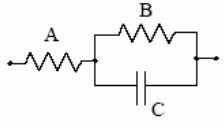
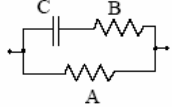
Aunque el objetivo de esta técnica es la realización de mediciones in-vivo o ex-vivo, para probar el desempeño de equipos y métodos de medición, se pueden emplear modelos de circuitos equivalentes de tejido normal como los que se observan en las figuras 1 y 2, en donde R_{∞} es la resistencia a altas frecuencias, R_0 es la resistencia a bajas frecuencias, α es una constante de relajación del tejido. Cabe anotar que estos modelos no contienen consideraciones como el efecto de los electrodos y la calibración que se les debe realizar antes de las mediciones, lo cual puede modificar algunos parámetros de escala o forma en la impedancia medida⁶.

⁴ PALKO T., BIALOKOZ F., WEGLARZ J., “*MULTIFREQUENCY DEVICE FOR MEASUREMENT OF THF COMPLEX*”, Institute of Precision and Biomedical Engineering, Warsaw University of Technology. Chodkiewina 81152.02-525 Wafsa

⁵ BROWN Brian H, TIDY John A, BOSTON Karen, BLACKETT Anthony D, SMALLWOOD Rod H, SHARP Frank, “*Relation between tissue structure and imposed electrical current flow in cervical neoplasia*”, *INSTITUTE OF PHYSICS PUBLISHING. Physiol. Meas.* **23** (2002) 159–168

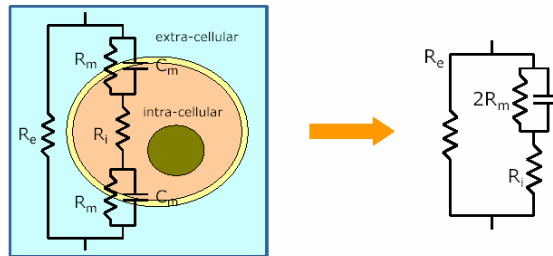
⁶ ALEMÁN B., PÁEZ B., “*MODELADO Y SIMULACIÓN DE TEJIDO EPITELIAL ESTRATIFICADO*”, Trabajo de grado de pregrado, Universidad Industrial de Santander, 2006

Figura 1. Modelo del circuito de un tejido de un solo lazo

Modelo circuital	Impedancias	Admitancias
 <p>a)</p>	<p>A: R_{∞}</p> <p>B: $\Delta R = R_0 - R_{\infty}$</p> <p>C: $-j\Delta R \left(\frac{f_{cz}}{f} \right)^{(1-\alpha_z)}$</p>	<p>A: G_{∞}</p> <p>B: $\frac{G_0 G_{\infty}}{G_{\infty} - G_0}$</p> <p>C: $j \frac{G_{\infty}^2}{G_{\infty} - G_0} \left(\frac{f}{f_{cy}} \right)^{(1-\alpha_y)}$</p>
 <p>b)</p>	<p>A: R_0</p> <p>B: $\frac{R_0 R_{\infty}}{R_0 - R_{\infty}}$</p> <p>C: $-j \frac{R_0^2}{R_0 - R_{\infty}} \left(\frac{f_{cz}}{f} \right)^{(1-\alpha_z)}$</p>	<p>A: G_0</p> <p>B: $G_{\infty} - G_0$</p> <p>C: $j(G_{\infty} - G_0) \left(\frac{f}{f_{cy}} \right)^{(1-\alpha_y)}$</p>

Fuente: Contributions to the measurement of electrical impedance for living tissue ischemia injury monitoring. Ivorra Cano Antoni.

Figura 2. Modelo del circuito de Rabbat para un sistema fisiológico unicelular



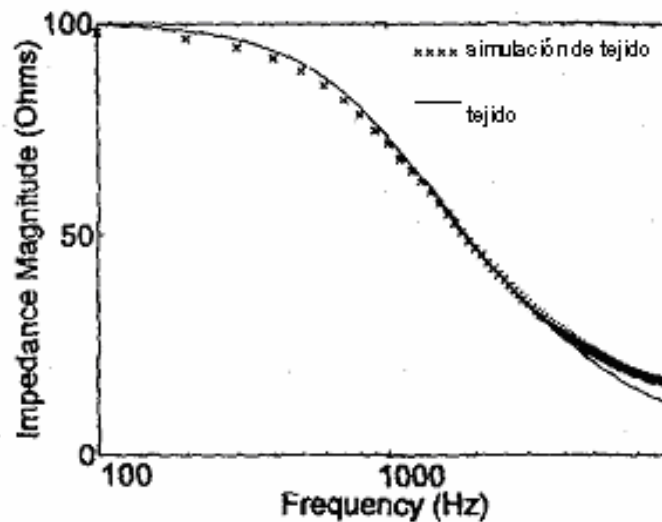
Fuente: Contributions to the measurement of electrical impedance for living tissue ischemia injury monitoring. Ivorra Cano Antoni .

1.1.2. Resultados obtenidos

Aunque son muchos y muy importantes los resultados obtenidos en investigaciones de EIE, es importante para el presente trabajo resaltar, el establecimiento de la región del espectro que contiene la información más

importante para la detección precoz de cáncer, la cual es llamada la zona beta que comprende frecuencias desde 5 – 10 kHz hasta 1MHz, además de la distribución que presenta el EIE^{7,8,9} la cual se puede observar en las figuras 3.a y 3.b, las cuales se diferencian en un factor de escala.

Figura 3.a. Magnitud de la impedancia obtenida a un modelo de circuito de tejido de resistencia y capacitor en paralelo con el método del pulso transitorio



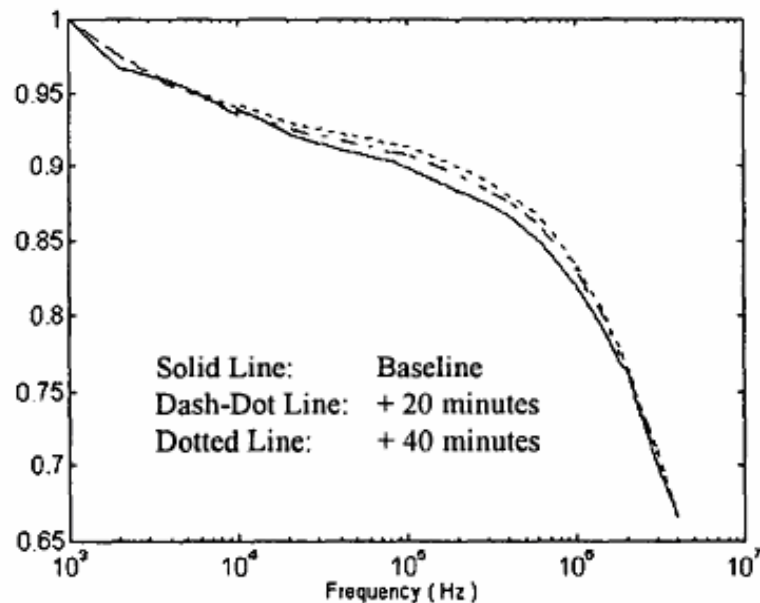
Fuente: Szlavik B., “Pulsed Transient Tissue Conductivity Measurement, Department of Electrical and Computer Engineering and Medicine”, 18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Amsterdam 1996

⁷ TUSTANLL B., WANG W., CHENG Z., “ An Analysis Of Systematic Errors In The Electrical Impedance Measurement Of Tissue Characteristics During An In-Vitro Study In The Frequency Range LkHz To 4mhz“, Dept of Electronic and Electrical Engineering, De Montfort University, The Gateway, Leicester, UK. Proceedings - 19th International Conference - IEEEEMBS Oct. 30 - Nov. 2, 1997 Chicago, IL. USA

⁸ Opcit Brown pp 19

⁹ Opcit Miranda pp 18

Figura 3.b. Impedancia de tejido que contiene una mezcla de fibras y grasa tomada diferentes tiempos



Fuente: TUSTANLL B., WANG W., CHENG Z., "An Analysis Of Systematic Errors In The Electrical Impedance Measurement Of Tissue Characteristics During An In-Vitro Study In The Frequency Range LkHz To 4mhz", - 19th International Conference - IEEE IEMBS Oct. 30 - Nov. 2, 1997 Chicago, IL. USA

1.2. TEORÍA DE MUESTREO POR DESFASE (TMD)

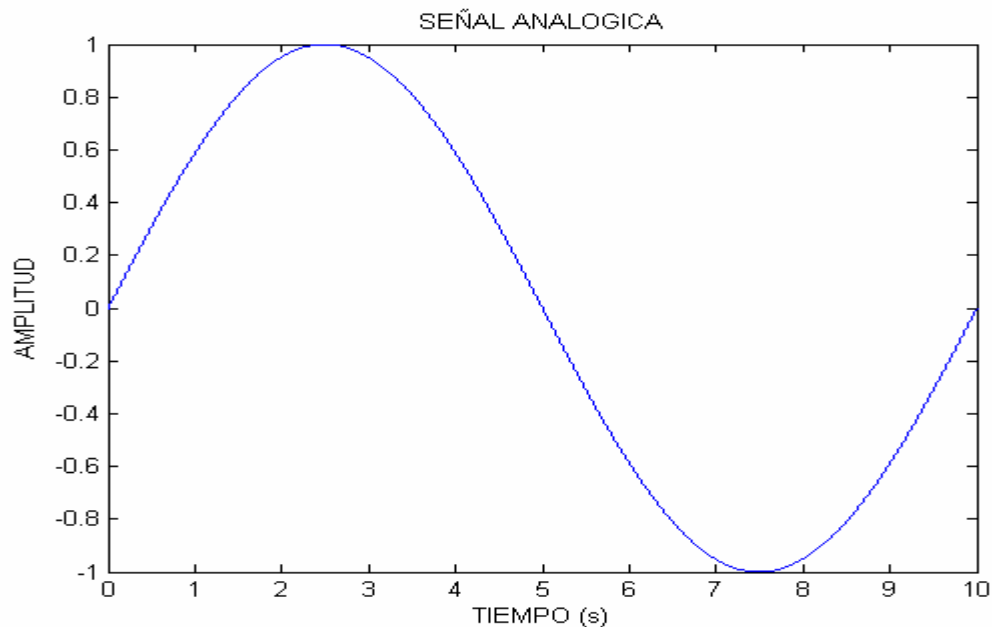
La teoría de muestreo por desfase que se aplica en este trabajo, fué formulada por el ing. David A. Miranda¹⁰, y nació como una técnica alternativa para lograr el muestreo de señales de alta frecuencia, con dispositivos que disponen de una frecuencia de muestreo inferior a dos veces la frecuencia de la señal. Se debe tener en cuenta que para su aplicación hay condiciones iniciales que se deben establecer como el período exacto (el grado de exactitud es importante si se quieren obtener buenos resultados) de la señal que se desea adquirir, el número

¹⁰ Miranda D., Barrero J., "Teoría de Muestreo Por Desfase" IX Simposio de Tratamiento de Señales, Imágenes y Visión Artificial, Manizales - Colombia 2004.

de periodos de señal a muestrear en los que se desea completar un periodo de está, y el número de muestras por periodo de muestreo que se desea tener. Para una mejor comprensión de esta técnica el sustento matemático se puede examinar en el ANEXO A.

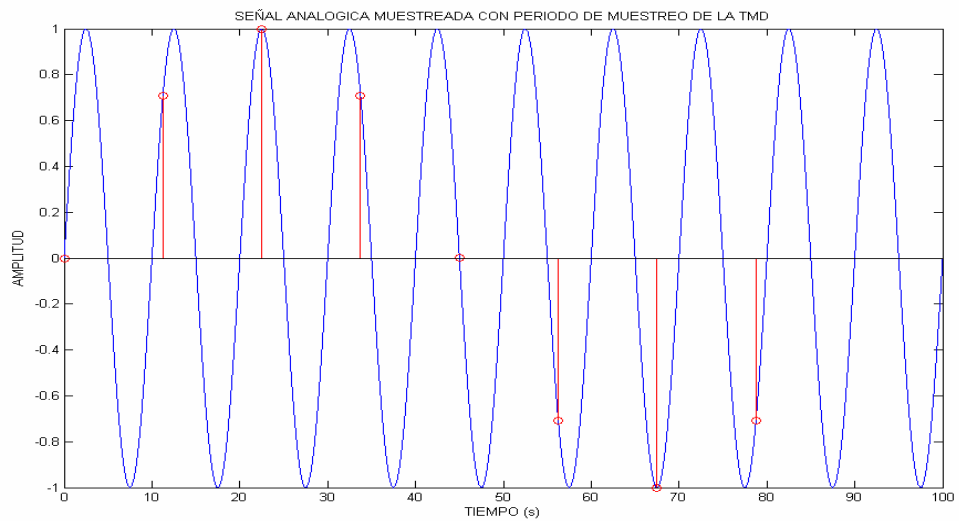
A través de esta técnica de muestreo, es posible muestrear de una manera eficiente señales analógicas periódicas y no periódicas de duración finita (figura 4). Una forma didáctica de entender la forma en que la teoría funciona, es llevarla a su caso extremo, en el cual se toma una muestra por periodo de la señal (figura 5). Partiendo de este supuesto, se muestrea la señal, y cada una de estas muestras se ubica en un fotograma (figura 6).

Figura 4. Señal analógica y periódica



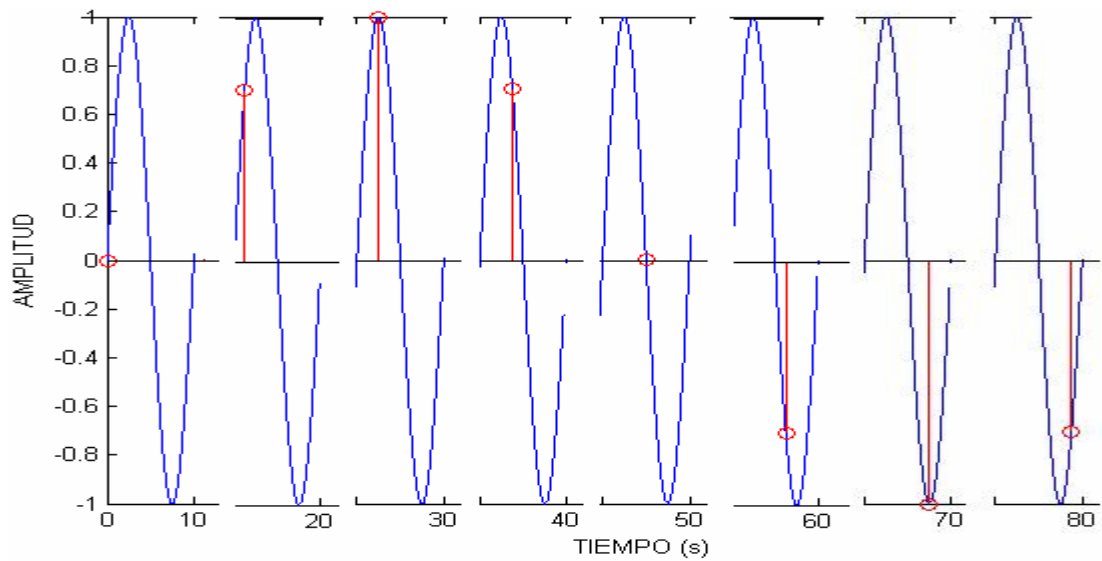
Fuente: Autores del proyecto

Figura 5. Señal muestreada a una muestra por periodo de señal mediante la TMD



Fuente: Autores del proyecto

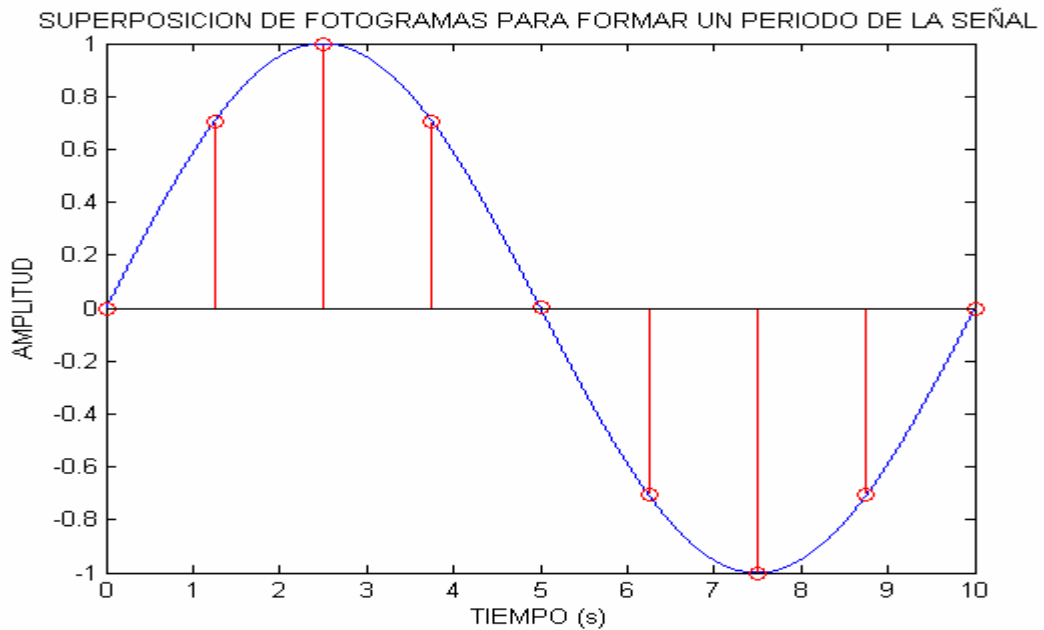
Figura 6. Muestras de la señal separadas por fotogramas



Fuente: Autores del proyecto

Una vez que las muestras de la señal se han separado en fotogramas, estos se superponen de manera que solo se observe un periodo de la señal continua (figura 7).

Figura 7. Reconstrucción de un periodo de la señal a partir de los fotogramas de la



Fuente: Autores del proyecto

De esta manera se pueden reconstruir digitalmente señales tomando hasta una muestra por período de la señal usando la TMD*, siempre y cuando se tengan suficientes periodos de la señal analógica para hacerlo, o tomar todas las muestras en un solo período de la señal con un período de muestreo lo suficientemente pequeño con respecto al periodo de la señal.

El sustento matemático de la TMD se encuentra en el ANEXO A1 (teoría de muestreo por desfase), además su análisis en frecuencia se encuentra en el ANEXO A2, el cual fue proporcionado por el codirector del proyecto.

* Cuando se lea TMD significa Teoría de Muestreo por Desfase

1.2.1. Planteamiento de algoritmos para el muestreo de señales basados en la TMD

Una vez visto el sustento matemático de la TMD en el ANEXO A, se busca formular un algoritmo que permita de manera óptima muestrear una señal que cumpla con las condiciones a las que limita la teoría (señal de duración finita).

Se comienza por plantear un periodo de muestreo (1).

$$\tau = \frac{T}{N} \times \left(1 \pm \frac{1}{M} \right) \quad (1)$$

T = Periodo conocido de la señal a muestrear en el dominio del tiempo

N = Números de muestras tomadas por periodo de la señal

M = Número de periodos donde la señal muestreada vuelve a ser periódica

Con esto la señal resultante tiene el mismo periodo de la señal original, pero con un tamaño de $\mu = M \times N$ muestras.

Cuando se cumplen todas las consideraciones planteadas anteriormente, incluyendo las restricciones expuestas en la TMD (ver ANEXO A), se pueden muestrear señales a una frecuencia cercana a su máxima frecuencia o incluso a una menor a la frecuencia de la señal. Esto se logra gracias a que se cambia el requerimiento de velocidad de muestreo por el de precisión al momento de muestrear.

Se debe considerar que los algoritmos planteados en este documento necesitan que la frecuencia de muestreo de la herramienta en la que se implemente el algoritmo, cuente en el caso más crítico, con una frecuencia muy cercana a la de

la señal a muestrear, ya que si se desea muestrear con herramientas que trabajen a una velocidad menor a la velocidad de la señal, se deben hacer modificaciones a los algoritmos para que sean aplicables, esto debido a que la teoría de muestreo por desfase no presenta estas limitaciones, dicha modificación a los algoritmos no se realiza en este trabajo de grado ya que no hace parte de los objetivos.

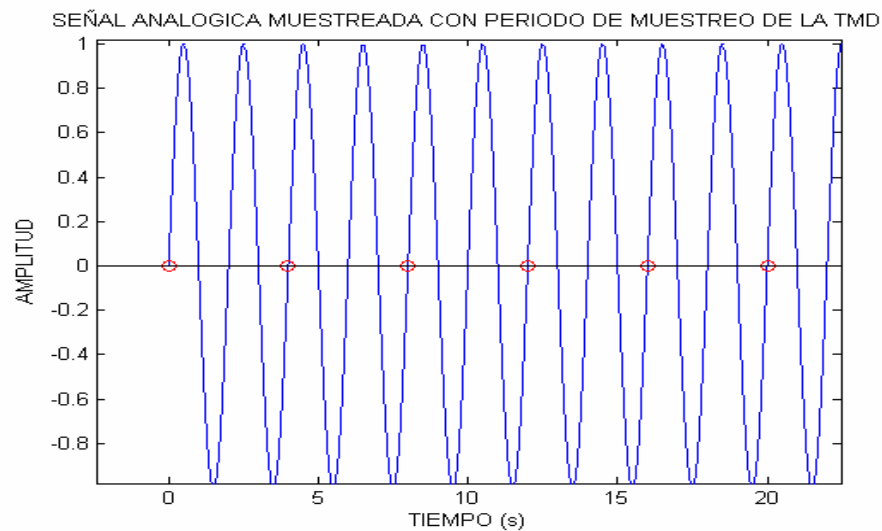
1.2.2. Consideraciones en la aplicación de la TMD

Aunque la teoría de muestreo por desfase (TMD) es una técnica muy eficiente, presenta consideraciones para su correcta aplicación como un número de periodos de señal suficiente, periodo de muestreo no múltiplo del periodo de la señal, capacidad de sincronización con el periodo de muestreo del dispositivo y estabilidad de la frecuencia de la señal a muestrear.

La necesidad de contar con un número de periodos suficiente, para realizar el muestreo de la señal, se debe a que la TMD reemplaza el requerimiento de velocidad de muestreo por un número determinado de periodos de señal para tomar las muestras.

En cuanto al periodo de muestreo que propone la TMD, este no debe ser múltiplo entero del periodo de la señal, ya que si esto ocurre, al reorganizar las muestras tomadas, la señal obtenida no representa un periodo de la señal original ya que éstas corresponderán a momentos de tiempo donde la amplitud es la misma. Lo anterior se ilustra en la gráfica 8, donde las muestras (pequeños círculos sobre el eje de tiempo) se reorganizan para intentar crear un periodo de la señal original, pero se obtiene una señal de amplitud cero.

Figura 8. Señal analógica muestreada con un periodo de muestreo de la de la TMD múltiplo entero del periodo de la señal analógica (periodo de muestreo = 6 y periodo de la señal = 2)



Fuente: Autores del proyecto

La sincronización entre el tiempo de muestreo del dispositivo con el que se realiza la adquisición de datos, y el periodo de muestreo propuesto por la TMD es crucial, pues si esto no es posible de hacer manteniendo un error bajo*, la señal reconstruida con las muestras adquiridas puede distar mucho de la señal original, o en un caso extremo puede llegar a parecerse a la señal de la figura 8.

Otro aspecto a tener en cuenta para realizar un correcto muestreo apoyándose en la TMD es la estabilidad de la frecuencia de la señal que se desea muestrear, ya que si ésta no es estable, las muestras tomadas no permitirán la reconstrucción de un periodo de señal ya que para obtener el periodo de muestreo τ es necesario contar con un valor de frecuencia de señal F bastante preciso.

* Entiéndase error en esta línea como la diferencia entre el tiempo de muestreo del dispositivo con el que se muestrea y el periodo de muestreo propuesto por la TMD

1.3. Análisis de error para señales muestreadas

En ocasiones cuando se adquieren señales usando la teoría de muestreo por desfase, se desea eliminar el error inmerso en el muestreo debido a inestabilidades de la fuente que genera la señal a adquirir, errores de cuantificación o simplemente ruido introducido por los conductores. A continuación se presenta un análisis del error elaborado por los autores del trabajo y el codirector, además de un análisis de ponderación del mismo para su minimización.

Como solución a esto se proponen dos soluciones, una por variación de error para reconstrucción de señales y otra por ponderación de error.

1.3.1. Variación de error para reconstrucción de señales en la TMD

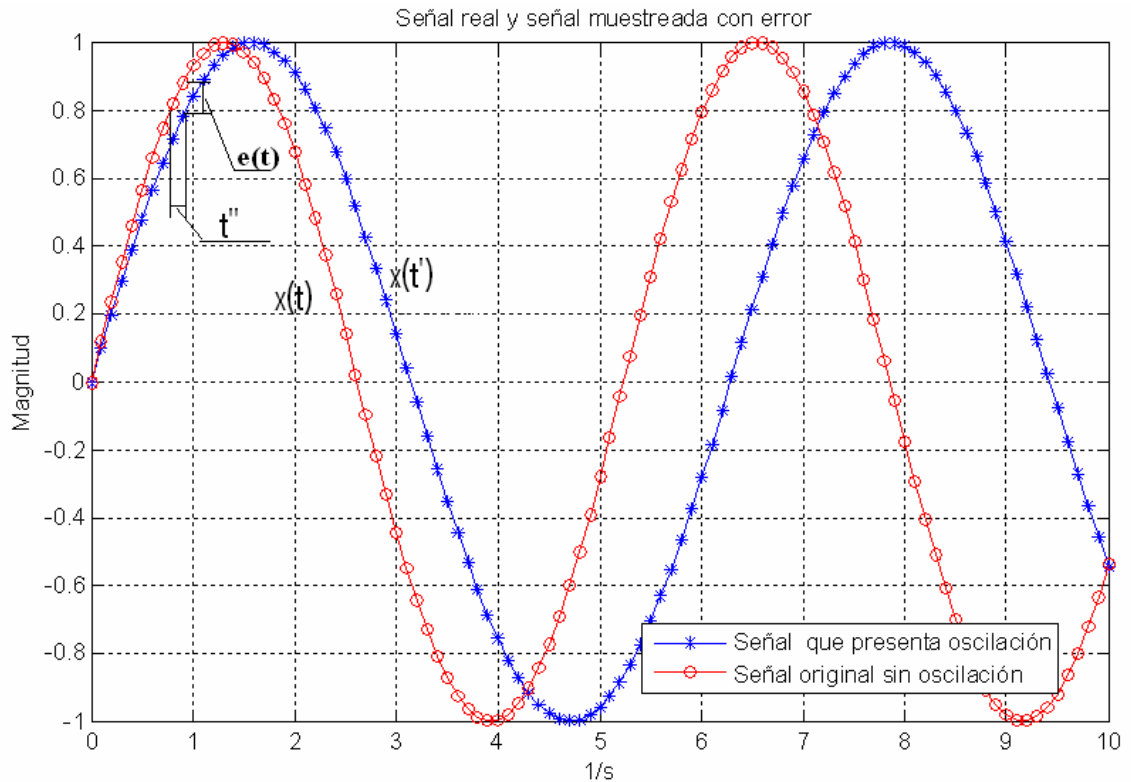
Al muestrear una señal, ésta puede presentar inestabilidades en frecuencia como la observada en la figura 9. Una de las causas de esta inestabilidad, es que la fuente de donde proviene la señal no es estable ya que su frecuencia oscila en un pequeño rango alrededor de una frecuencia central.

En la figura 9 se ve como la señal original (o señal real sin error) $x(t)$ oscila cambiando su periodo por pequeños instantes de tiempo convirtiéndose en $x(t')$, donde $t' = t - t''^*$, lo cual altera las mediciones al momento de adquirir una muestra de la señal. Sin embargo, a través de un razonamiento matemático se puede disminuir el error que se produce hasta llegar a eliminarlo totalmente (al menos de manera teórica).

* t'' es un Δt que debe ser entendido solo como una diferencia temporal y no como un desplazamiento de la señal en el tiempo

Para iniciar el análisis de variación de error entiéndase $x(t)$ como la señal original, $x(t')$ la señal alterada por la oscilación de la fuente es decir la señal medida y $e(t)$ el error entre la señal original y la señal alterada por la oscilación de la fuente.

Figura 9a. Oscilación de una onda provocada por inestabilidad en frecuencia de la fuente generadora



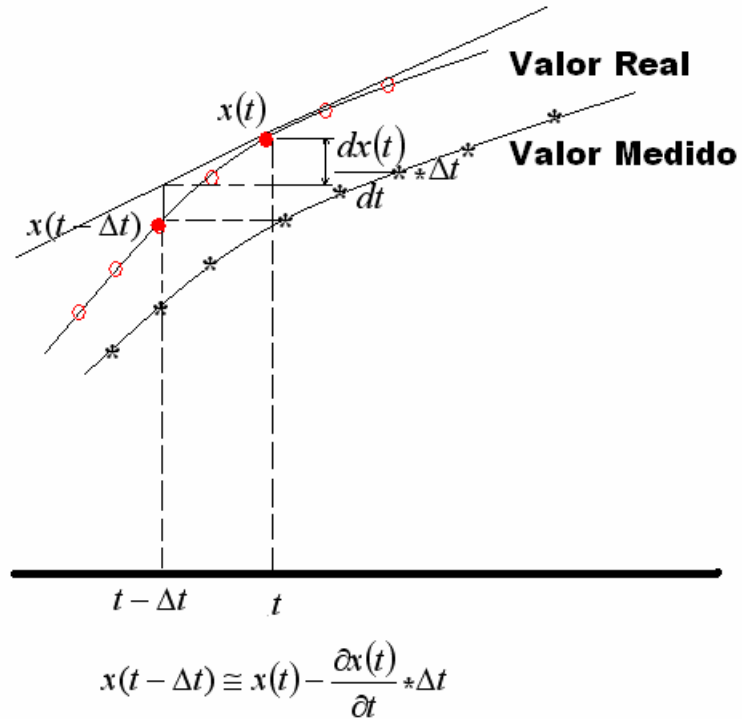
Fuente: Autores del proyecto.

Con base en la figura 9 se puede describir el error entre la señal $x(t)$ y la señal $x(t')$ como se muestra en la ecuación (2).

$$e_A(t) = x(t) - x(t') = x(t) - x(t - \Delta t)^* \quad (2)$$

* Se expresa t'' como Δt ya que es una variación alrededor de t con un valor muy pequeño comparado con t y t'

Figura 9b. Ampliación de oscilación de una onda provocada por inestabilidad en frecuencia de la fuente generadora



Fuente: Autores del proyecto.

Tomando la aproximación de Taylor de primer grado, alrededor de cero para Δt en un punto p (punto que pertenece al rango de $x(t)$ donde se está tomando la muestra), se tiene:

$$x(t - \Delta t) \cong x(t) - \left. \frac{\partial x(t)}{\partial t} \right| \Delta t \quad (3)$$

Si el límite de $e(t)$ cuando Δt tiende a cero, debido a que Δt es pequeño, como lo es al compararlo con el periodo de la señal, que es la señal a la cual se le eliminará el error, entonces¹¹:

$$e(t) = \left. \frac{dx(t)}{dt} \right| \Delta t \quad (4)$$

Nótese que Δt representa la diferencia temporal debida a la variación de la frecuencia de la señal, al cual puede ser representada por una variable aleatoria:

$$\Delta t = \tau(t) \quad (5)$$

De la figura 9 asumiendo que $x(t)$ es la señal real y $x(t - \Delta t)$ es la señal medida con el osciloscopio, se relacionan con $e(t)$ de la siguiente manera:

$$x_{real} = x_{medido} + e(t) \quad (6)$$

$$x_{real} = x_{medido} + \left. \frac{dx(t)}{dt} \right|_p \tau(t) \quad (7)$$

Por ejemplo, para una señal sinusoidal $x(t) = x_0 \text{sen}(wt)$ y $\tau(t)$, una distribución de la variación temporal de $x(t)$ se tiene:

$$x_{real} = x_{medido} - \tau(t)x_0 w \cos(wt) \quad (8)$$

¹¹ LEITHOLD L., "Calculo con geometría analítica" Editorial Harla sexta edición 1992 pag 186

La ecuación (8) muestra que es posible eliminar el error producido* (segundo factor del lado derecho) para obtener $x_{real} = x_{medido}$.

1.3.2. Ponderación de error

Nótese que la ecuación (8) indica que $x_{real} = x_{medido}$ si se logra atenuar el error $e(t)$, lo cual es posible promediando n veces x_{real} dada la naturaleza aleatoria de $\tau(t)$. Con base en lo anterior, se puede interpretar lo expuesto en el numeral 1.3.1 para eliminar el error producido al muestrear señales por medio de una ponderación del mismo.

1.3.2.1. Promediado de n veces la señal adquirida

Supóngase que se muestrea n veces una misma señal por medio de un conversor análogo-digital aplicando la TMD. Aunque en principio todas las muestras medidas deberían ser exactamente iguales a la señal real (que es la señal que en teoría se espera obtener), en la práctica se tiene que no lo son, en parte por lo expuesto en el numeral 1.3.1 acerca de la inestabilidad en frecuencia de la fuente generadora. Sin embargo esta diferencia se puede minimizar para obtener la señal real promediando los n muestreos de ésta como se muestra a continuación.

Sea $x_r(t)$ la señal real a muestrear mediante un conversor (señal que en teoría se espera obtener), $e(t)$ el error que introduce (por ejemplo la inestabilidad del generador de señal en frecuencia o los errores de cuantificación introducidos por

* Si su media es cero

el conversor), $x_{r_i}(t)$ un muestreo de la señal $x_r(t)$ y $\hat{x}_r(t)$ la señal producto del promedio de los i muestreos de $x_r(t)$.

Partiendo de:

$$\hat{x}_r(t) = \frac{(x_{r_1} + e_1) + (x_{r_2} + e_2) + \dots + (x_{r_n} + e_n)}{n} \quad (9)$$

En donde cada factor $(x_{r_i} + e_i)$ representa el i ésimo muestreo de $x_r(t)$ junto al error que se presenta en dicho muestreo.

Ahora si $x_{r_1} \approx x_{r_2} \dots \approx x_{r_n}^*$ debido a que son muestreos de la misma señal, entonces se tiene que:

$$\hat{x}_r(t) = \frac{nx_{r_n} + e_1 + e_2 + \dots + e_n}{n} \quad (10)$$

Como los errores son pequeños en comparación con la señal y a medida que n aumenta, nx_{r_n} crece rápidamente con respecto al error, por lo tanto:

Si $nx_{r_n} \gg e_1 + e_2 + \dots + e_n$

Entonces se tiene que:

* Se puede considerar que las señales muestreadas son casi iguales debido a que el muestreo corresponde a la misma señal y la adquisición de muestras se realiza con un único conversor

$$\hat{x}_r(t) \approx \frac{nx_{rn}(t)}{n} \approx x_{rn}(t) \quad (11)$$

De la ecuación (12) se aprecia que después de realizar el promedio de n muestreos $(x_{r_i} + e_i)$, la señal $\hat{x}_r(t)$ que es la señal producto de los muestreos, será muy similar al muestreo i ésimo sin error, y por lo tanto a la señal real $x_r(t)$ (ecuación (13)).

$$\hat{x}_r(t) \approx x_r(t) \quad (12)$$

Aunque la ecuación (12) no muestra nada novedoso por si sola, si es que la igualdad entre la señal real y la señal obtenida mediante el promedio de sus n muestreos, será mayor a mayor número de muestreos a promediar.

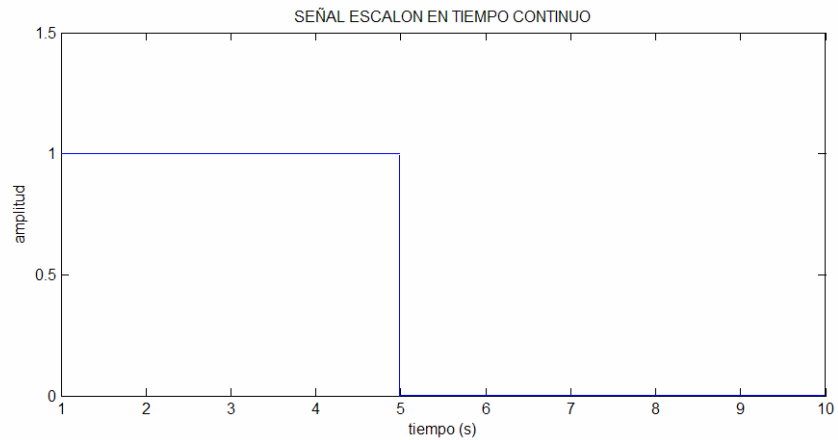
1.4. TRANSFORMADA RÁPIDA DE FOURIER

La transformada rápida de fourier (FFT) se puede obtener por medio de un conjunto de algoritmos computacionales, que permiten realizar la Transformada Discretas de Fourier (*DFT*) de una manera eficiente, ya que el tiempo de procesamiento se reduce, debido a que se reduce el número de multiplicaciones y sumas implícitas en su obtención.

Una forma de entender la Transformada Discreta de Fourier es imaginarse una señal continua, y a esta señal se le calcula su transformada de Fourier* (figura 10).

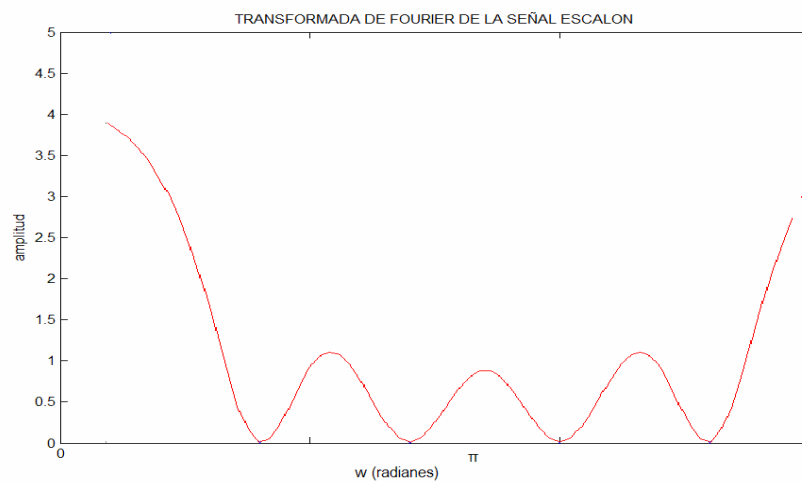
* Las figuras 11 y 12 representan la magnitud de la Transformada de Fourier de la señal escalón

Figura 10. Función escalón



Fuente: Autores del proyecto

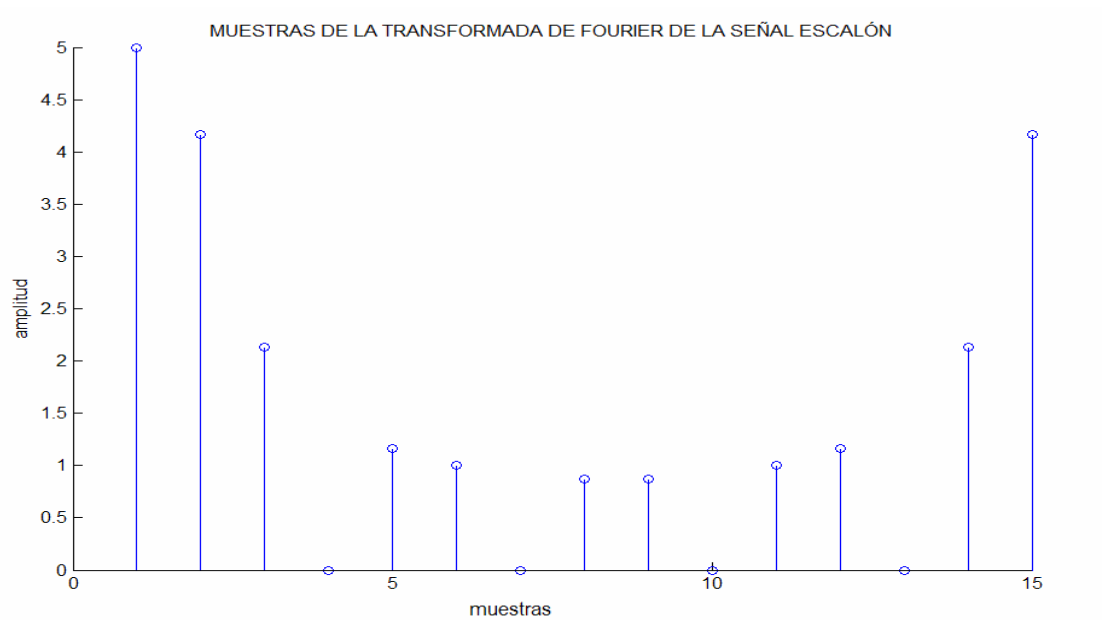
Figura 11. Transformada de Fourier de la función escalón



Fuente: Autores del proyecto

A esta transformada de Fourier se le toman **N** muestras equiespaciadas en frecuencia, y la secuencia de datos que queda es la Transformada Discreta de Fourier (*DFT*) (figura 12).

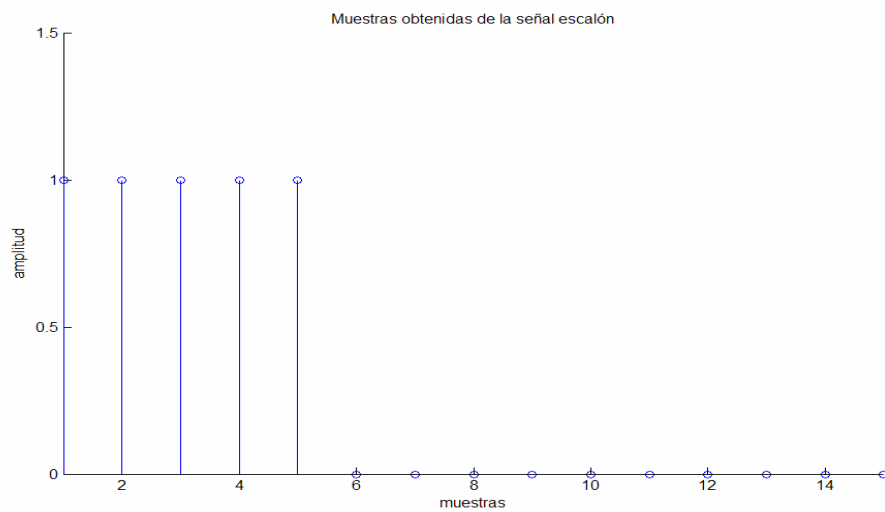
Figura 12. Muestras de la Transformada de Fourier de la señal escalón



Fuente: Autores del proyecto

Otra manera de entender la transformada discreta de Fourier, es visualizar una señal continua (figura 10), a la cual se le aplica un muestreo (figura 13).

Figura 13. Muestras equiespaciadas de la señal escalón

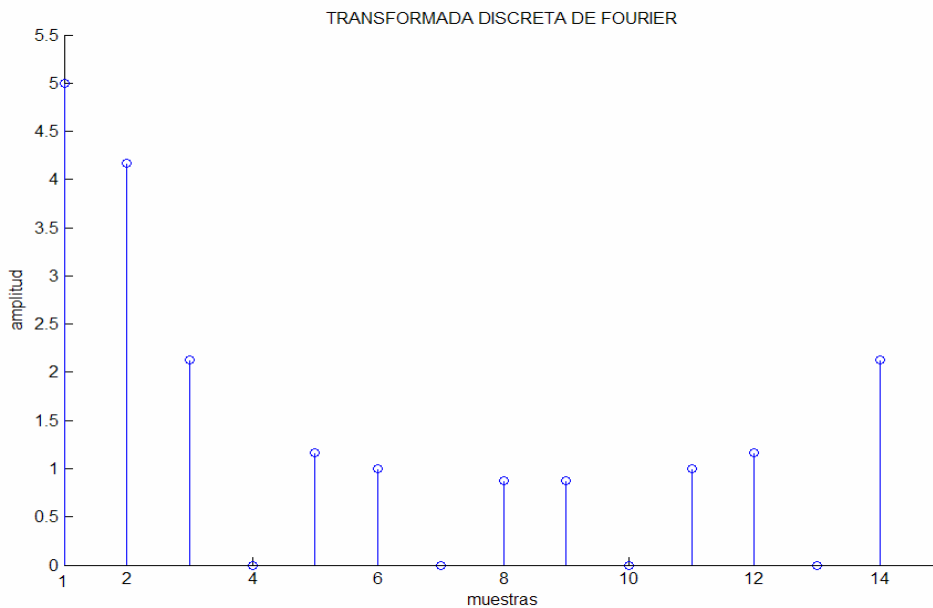


Fuente: Autores del proyecto

De esta señal discreta resultante, se toma un intervalo finito de muestras, y a ese intervalo se le aplica la ecuación de análisis (13), donde el vector $X[k]$ representa la Transformada Discreta de Fourier de las muestras tomadas al escalón de entrada que se muestran en la figura 13 (figura 14), y k va de 0 hasta $N-1$, con N igual al número de muestras.

$$X[K] = \sum_{n=0}^{N-1} x[n] \cdot W^{nK} \quad K=0,1,\dots,N-1 \quad (13)$$

Figura 14. Transformada Discreta de Fourier



Fuente: Autores del proyecto

De manera análoga, se pueden conocer las muestras de una señal a partir de los coeficientes de la Transformada de Fourier aplicando la ecuación de síntesis (14).

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] W_N^{-kn} \quad n=0,1,\dots,N-1 \quad (14)$$

1.5. PROCESADOR DIGITAL DE SEÑALES

Como ya se mencionó en el presente texto, los objetivos fundamentales de este trabajo son el desarrollo de dos algoritmos, uno de muestreo basado en la teoría de muestreo por desfase (TMD), y otro de cómputo de la FFT, de manera implícita se encuentra entonces planteada la necesidad de trabajar con un dispositivo que permita realizar la conversión analógica digital y el procesamiento digital de señales, para tal tarea es entonces usado un procesador digital de señales (DSP) de la familia motorola 56800. Los algoritmos desarrollados se implementan en un módulo de evaluación 56F801 y en una tarjeta de desarrollo 56F805, estos dos DSPs difieren en el tamaño de la memoria y en la elección de las unidades periféricas. El fabricante de estos dispositivos proporciona una herramienta software para la programación de los mismos mediante un PC, esta es el Metrowerks Codewarrior. La programación se puede realizar tanto en el lenguaje C, como en Assembler.

En el marco del proyecto de maestría en Ingeniería Electrónica presentado por el ingeniero David Alejandro Miranda para la detección precoz de cáncer de cuello uterino, se elige trabajar con un procesador digital de señales (DSP) de motorola de la familia 56800, por tal razón en el presente texto no se justificará de manera rigurosa la selección de este dispositivo en comparación con otros productos similares llámense microprocesadores etc. Esto debido a que el presente proyecto

como ya se mencionó en la introducción, está encaminado a fortalecer la línea de investigación en detección precoz de cáncer de cervix¹².

¹² Opcit Miranda pp 18

2. IMPLEMENTACIÓN DE LA FFT Y LA TMD EN MATLAB

Antes de la implementación de los códigos de la Transformada Rápida de Fourier (*FFT*), y la teoría de muestreo por desfase (*TMD*) en el DSP, se realizó su simulación mediante la herramienta software Matlab, con el fin de verificar que los códigos arrojaban los resultados esperados y de esta manera descartar posibles errores de programación, al momento de la implementación en el Procesador Digital de Señales (*DSP*).

2.1. IMPLEMENTACIÓN DE LA TMD EN MATLAB

Antes de seleccionar el código a implementar en el Procesador Digital de Señales (*DSP*), se trabajó en la creación de algunos algoritmos fallidos, de los cuales se presenta a manera de ilustración el *Código Inicial** (ver anexo B). Estos códigos fallidos permitieron mediante una depuración más rigurosa, llegar a los códigos finales (*TMD generalizada con desfase negativo*, *TMD generalizada con desfase positivo* y *TMD para una muestra*, ver anexo B). Estos códigos (*TMD generalizada con desfase negativo* y *TMD generalizada con desfase positivo*) difieren en el signo del desfase en la ecuación usada para calcular el periodo de muestreo (ver ecuación 1), el primero presenta un signo negativo, mientras el segundo presenta un signo positivo.

Lo

* Este al igual que todos los códigos desarrollados tanto para Matlab versión 7.0 como para Codewarrior versión 6.1, se encuentran en la carpeta ANEXOS del CD “*CÁLCULO DEL ESPECTRO DE IMPEDANCIA ELÉCTRICA DE TEJIDO DE CUELLO UTERINO POR TRANSFORMADA RÁPIDA DE FOURIER IMPLEMENTANDO LA TEORÍA DE MUESTREO POR DESFASE EN UN DSP MOTOROLA DE LA FAMILIA 56800.*” donde se compilan todos los trabajos relacionados con el presente trabajo de grado.

que buscan todos los códigos es organizar las muestras usando un vector de tiempos, el cual mediante el uso de matemática sencilla ubica las muestras en el orden que deben quedar y no en el orden en que originalmente fueron tomadas. En el caso del *código inicial*, se ve claramente que es el más extenso en cuanto a líneas de programación, también el que realiza más operaciones, por tanto es fácil prever que será el que más tiempo de procesamiento emplea. Este utiliza dos períodos de muestreo distintos, uno calculado con la ecuación para realizar el muestreo por desfase, y el otro, una unidad de tiempo más rápido, debido a esto muestrea de manera alternante el comienzo y el final de la señal. Además la parte tomada con el menor periodo realiza un barrido mas rápido de la señal que la parte tomada con el periodo de muestreo original. Se ve también que en este código el espacio entre muestra es irregular.

Los códigos *TMD generalizada con desfase negativo* y *TMD generalizada con desfase positivo*, son más eficientes ya que presentan un número menor de líneas de código y de operaciones. Estos comparten con el *código inicial* la forma esencial del vector de tiempos, encargado de ubicar cada muestra en su lugar, en estos códigos se ve claramente un periodo de muestreo aparente, que permite tener las muestras igualmente espaciadas unas de otras.

Finalmente se tiene un cuarto código que presenta menos operaciones que los tres anteriores, lo que reduce el tiempo de operación, este es el *TMD para una muestra*. Si se ven los resultados de este y se comparan con los códigos 2 o 3 no se notaría ninguna diferencia ya que a pesar de que el código en si es distinto su programación se basó por completo en los resultados de los dos anteriores tomando solo una muestra por periodo de la señal, para este caso en particular el código cumple con todo lo planteado por la teoría de muestreo por desfase. Cuando el número de muestras es distinto de uno este código (código cuatro) pierde confiabilidad, en esos casos se recomienda usar el código *TMD generalizada con*

desfase negativo o *TMD generalizada con desfase positivo*, ya que estos fueron diseñados para cumplir todo lo planteado en la teoría de muestreo por desfase.

2.2. IMPLEMENTACIÓN DE UNA FFT EN MATLAB

Una de las dos razones por las que se implementa la Transformada Rápida de Fourier (*FFT*) en la herramienta de simulación Matlab, es verificar que los códigos arrojan los resultados esperados, y de esta manera, eliminar una posible fuente de error al momento de implementarlos en el procesador digital de señales (*DSP*). La segunda razón, radica en que al crear un código en lugar de implementar uno ya existente, se puede llegar a una mejor comprensión de qué es lo que se está haciendo, cuales son las ventajas y desventajas frente a otros códigos que realizan igual función, y cuáles son sus límites y posibles causas de error.

Finalmente es de tener presente, que el código escrito busca, antes de un cómputo rápido, tener un manejo eficiente en el uso de la memoria.

2.2.1. Selección del tipo de código a implementar

Aunque existe gran variedad de algoritmos que se pueden implementar para obtener la *FFT*, se descartan algoritmos como el de Goertzel y la transformada Chirp porque el objetivo de estos es conocer información de la *FFT* en una porción específica del espectro, y lo que se busca es obtener información sobre todo el intervalo de frecuencias (si el dispositivo usado para el muestreo lo permite), ya que aunque el rango de frecuencias de interés para detección de cáncer no supera 1MHz, a futuro los resultados se desean extrapolar a otras aplicaciones en estudio de tejidos que requieren un rango de frecuencias mas amplio.

Tomando en cuenta el anterior planteamiento, se decide implementar el algoritmo de Diezmado en el Tiempo base 2, el cual permite obtener la DFT en todo el intervalo de frecuencias.

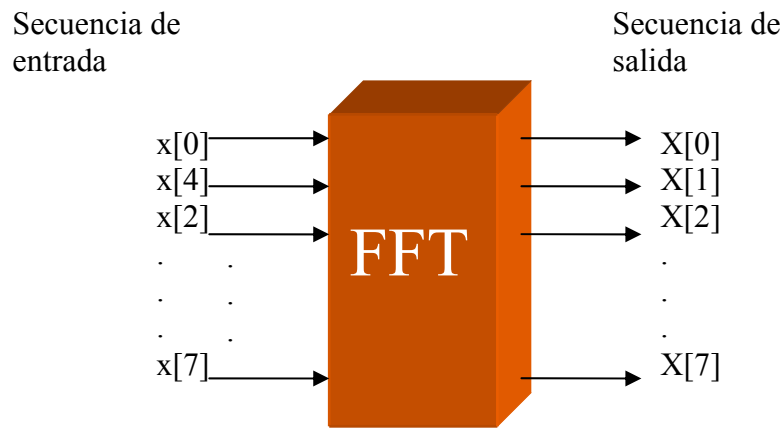
2.2.2. Implementación del código de la FFT con base en el algoritmo de diezmado en el tiempo

La forma como se implementa el código para el algoritmo, comienza separando la Transformada Rápida de Fourier en dos partes. La primera es la organización de la secuencia de entrada en orden de bit inverso, y la segunda la implementación de la Transformada Discreta de Fourier.

2.2.2.1. Organización de la secuencia de entrada en orden de bit inverso

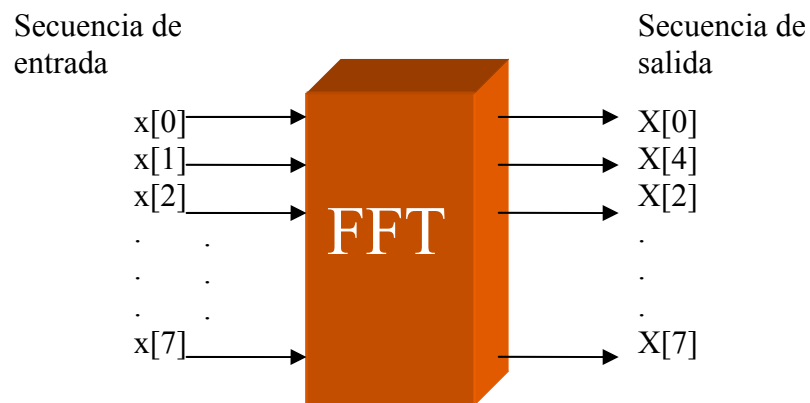
Cuando se organiza la secuencia de entrada en orden de bit inverso, se garantiza que la secuencia de salida, la cual contiene los coeficientes de la *DFT*, está en orden normal. Si este ordenamiento no se hace, y la secuencia de entrada se encuentra en orden normal, la secuencia de salida se encuentra en orden de bit inverso (figura 15). Cabe anotar que esto ocurre no sólo para el algoritmo de diezmado en el tiempo, también para el algoritmo de diezmado en frecuencia, el cual realiza una función equivalente al algoritmo de diezmado en el tiempo.

Figura 15. Organización de la secuencia de entrada en orden de bit inverso antes de ingresar al bloque de $FFT^{*,**}$



Fuente: Autores del proyecto

Figura 16. Secuencia de entrada en orden normal antes de ingresar al bloque de FFT



Fuente: Autores del proyecto

* $x[...]$ representa una muestra de la secuencia de entrada, mientras que $X[...]$ representa un coeficiente de Fourier de la secuencia de salida

** Para hacer fácil la comprensión del ejemplo se usa una FFT de ocho punto

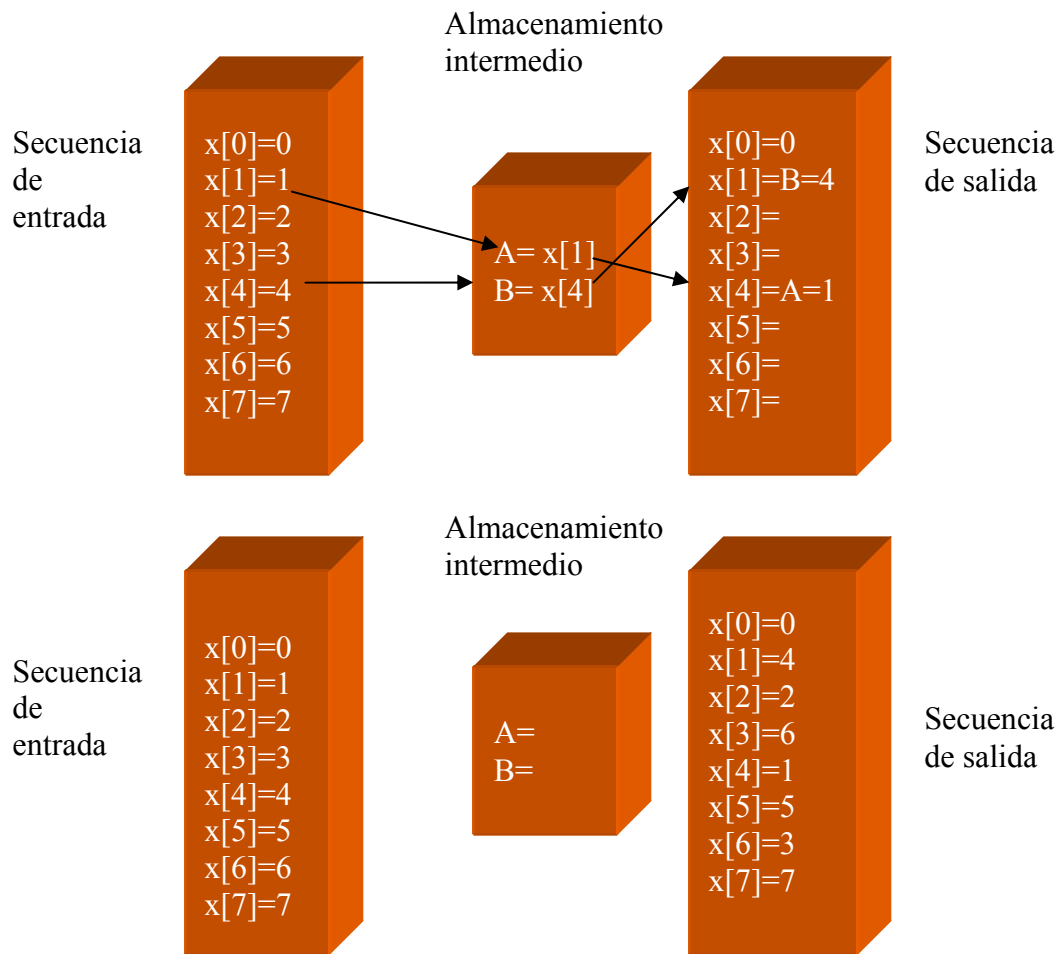
Para ordenar la secuencia de entrada se emplea el cómputo en el mismo lugar¹³, ya que aunque presenta la desventaja de requerir de memoria de acceso aleatorio, se hace un uso más eficiente de esta.

Cuando se habla de cómputo en el mismo lugar para hacer el ordenamiento en orden de bit inverso, significa que el proceso para ordenar una pareja de elementos de la secuencia de entrada se repite para el resto y este consiste simplemente en tomar los dos números que se van a intercambiar, almacenarlos en variables intermedias y después cambiarlos de posición (figura 17^{*}).

¹³ OPPENHEIM A., SCHAFER R., BUCK J., *Tratamiento De Señales En Tiempo Discreto*, Prentice Hall, 2ª edición, pp 642

* En este y en todos los ejemplos que siguen, se emplea una *FFT* de ocho puntos a menos que se especifique lo contrario

Figura 17. Ordenamiento de una secuencia e orden de bit inverso*



Fuente: Autores del proyecto

El código implementado para el ordenamiento de una secuencia en orden de bit inverso, se puede observar en el Anexo B, *Bit Inverso*.

Aunque en el simulador Matlab existe el comando *bitrevorder* que realiza la misma función que el código implementado para la organización del bit inverso en

* Los valores dados al vector x son aleatorios

lenguaje C, se debe tener en cuenta que uno de los objetivos del presente proyecto de grado es implementar el código de la Transformada Rápida de Fourier en el *DSP*, lo que hace necesario crear un código que realice esta organización (si el código implementado se realiza en lenguaje C), o usar una de las opciones de organización con las que cuenta el bean que realiza la *FFT* en CodeWarrior.

2.2.2.2. Implementación de la transformada discreta de Fourier

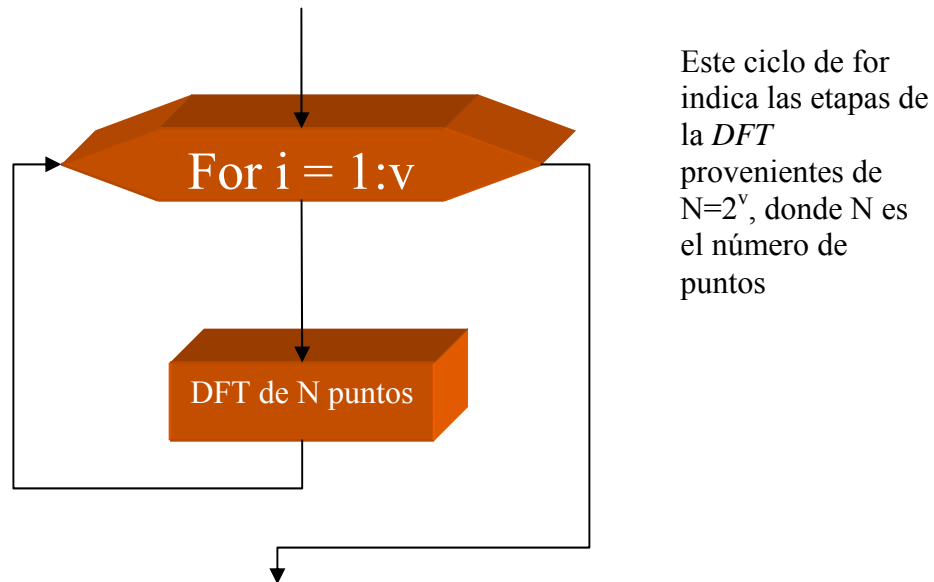
El propósito de este numeral, mas allá de explicar paso a paso el código creado, tiene como finalidad explicar el porqué se escribió de la manera en que se encuentra.

Una vez se ha reorganizado el vector de entrada* se le aplica a este la transformada discreta de Fourier por medio del código creado (ver Anexo B, *Algoritmo de FFT creado*). Este código se basa en el algoritmo de diezmado en el tiempo¹⁴ y sigue la estructura general que se muestra en la figura 18.

* Entiéndase vector y secuencia como lo mismo

¹⁴ OPPENHEIM A., SCHAFFER R., BUCK J., *Tratamiento De Señales En Tiempo Discreto*, Prentice Hall, 2ª edición, pp 637-648

Figura 18. Estructura general del código escrito para obtener la *FFT* en el simulador Matlab



Fuente: Autores del proyecto

Al momento de escribir el código, se tienen en cuenta dos aspectos. El primero es almacenar la menor cantidad de datos posibles, con el fin de optimizar la memoria, y el segundo es disminuir el error causado por la acumulación de errores provenientes de las operaciones realizadas en cada etapa de la *DFT*.

Una forma de reducir la cantidad de datos almacenados es seleccionar de manera correcta el manejo que se le da a los factores *twiddle*. Al momento de definir como se van a obtener hay dos opciones:

1. Crear todos los factores y almacenarlos en memoria
2. Calcular los valores cuando se van necesitando

Se toma la segunda opción porque el espacio disponible para el almacenamiento de los datos es limitado, debido a que la cantidad de muestras de tensión, provenientes del tejido de cuello uterino que se desea tomar, debe ser maximizada. Además, aunque el tiempo de procesamiento aumenta, este es del orden de los microsegundos, aspecto que no es relevante si se tiene en cuenta que el tiempo que emplea la toma de una medición es de 10 segundos en promedio, y uno de los propósitos del macroproyecto (creación de un bioimpedanciometro) es que el resultado de la medición, después de terminada la toma de datos, este disponible en un lapso de tiempo de 10 minutos.

Para disminuir el error causado por la acumulación de errores provenientes de las operaciones como sumas y multiplicaciones en las etapas ejecutadas en la Transformada Discreta de Fourier (figura 18), hay contadores que vuelven a cero en cada una de las v etapas y el recálculo de algunas variables como M , que es la magnitud del vector *twiddle* (ver Anexo B, *Algoritmo de FFT creado*), se recalculan en cada etapa.

3. IMPLEMENTACIÓN DE LA TMD Y FFT EN EL PROCESADOR DIGITAL DE SEÑALES

En este capítulo se aborda primero el planteamiento del problema, el diagrama de flujo del algoritmo desarrollado y su implementación en el *DSP*, para la Teoría de Muestreo por Desfase (*TMD*). En segunda instancia se realizan las mismas etapas en la implementación del algoritmo para obtener la Transformada Discreta de Fourier de una secuencia de datos muestreados (*FFT*).

3.1. IMPLEMENTACIÓN DE LA TEORÍA DE MUESTREO POR DESFASE (*TMD*) EN EL PROCESADOR DIGITAL DE SEÑALES

El proceso de llevar a acabo la implementación de la teoría de muestreo por desfase en el DSP se explica a continuación, partiendo del planteamiento del problema y finalizando con el algoritmo elaborado.

3.1.1. Planteamiento del problema de TMD para medición de espectro de impedancia eléctrica.

La razón de usar la teoría de muestreo por desfase (*TMD*) es probar mediante una aplicación práctica el desempeño de este método de muestreo, ya que con él se pueden obtener beneficios como una frecuencia de muestreo menor a la frecuencia de Nyquist, hecho que se puede aprovechar en diversas aplicaciones como por ejemplo en detección precoz de cáncer, donde se requiere adquirir

señales de que llegan hasta 1MHz (región beta¹⁵), lo cual si se realiza por el método tradicional (muestreo a la frecuencia de Nyquist), requiere conversores análogo – digital de al menos 2MHz de frecuencia de conversión, mientras que al realizarlo con la TMD, esta frecuencia puede reducirse a la mitad.

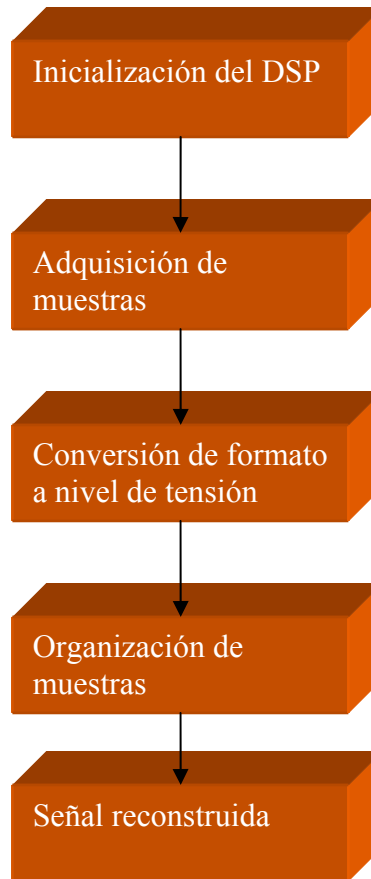
Esta característica de reducción en la frecuencia de adquisición se usa en el presente trabajo para aprovechar al máximo el equipo disponible, ya que para la toma de datos de tensión y corriente (con las que se halla la impedancia de transferencia) se cuenta con un DSP 56F801 de Motorola que tiene una frecuencia máxima adquisición de 588kHz (teóricos, ya que no se está considerando el tiempo que toma en almacenar la muestra tomada).

3.1.2. Diagrama de flujo del algoritmo implementado para la *TMD*

A continuación se muestra el diagrama de flujo de los pasos que se llevan a cabo para la obtención de las muestras

Figura 19. Diagrama de flujo del código implementado para la *TMD*

¹⁵ Opcit LOZANO pp18



Fuente: Autores del proyecto

El primer paso es inicializar el DSP, dentro del cual se activa el conversor analógico-digital por medio del método *AD1_Start()* del bean *AD1_ADC* del DSP¹⁶. Una vez se ha activado el ADC se toman las muestras leyendo el registro *ADCA_ADRSLT0* del conversor.

Aunque el ajuste de la frecuencia de muestreo en el DSP es muy exacto, después de adquirir cada muestra, ésta debe ser almacenada, acción que incrementa el

¹⁶ MOTOROLA, "Processor Expert help-2.55 for codewarrior plug-in for Motorola", Codewarrior versión 6.1.

tiempo de muestreo que se considera. Para resolver este inconveniente, se midió el tiempo que demora el tomar un conjunto determinado de muestras, leerlas del registro (*ADCA_ADRSLT0*) y almacenarlas en un vector, por medio de un bean de propósito general que mantiene una salida de 3,3V mientras se realiza la operación y cambia a 0V cuando finaliza este proceso. Esto con el fin de visualizar una onda cuadrada por medio de un osciloscopio, lo cual permitió medir el tiempo real de muestreo (tiempo que demora en adquirir una muestra, almacenarla en una posición de memoria y estar listo para adquirir la siguiente muestra).

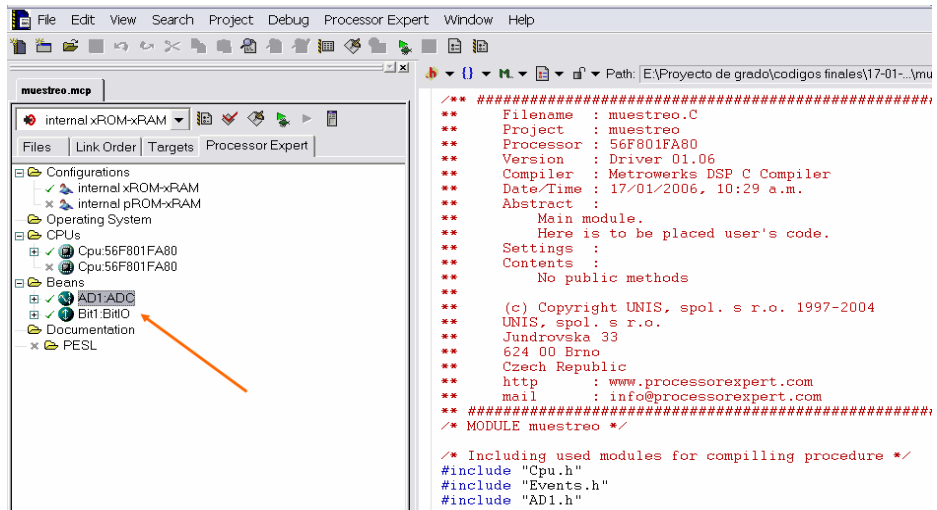
3.1.3. Implementación de la *TMD* en el procesador digital de señales

El primer paso a seguir para implementar la *TMD* en el procesador digital de señales es medir de manera correcta el tiempo de muestreo (según se explica en el numeral anterior), operación que requiere la inclusión de un bean^{*,**} de propósito general y uno de ADC como se muestra en la figura 20.

Figura 20. Beans ADC y de propósito general para la adquisición de muestras

* Entiéndase bean como un macro de la herramienta *Codewarrior*, que facilita la programación en el modo procesador experto, entre los macros mas usados en el desarrollo del proyecto estan el de conversión analógica digital, modulación por ancho de pulso y *FFT*

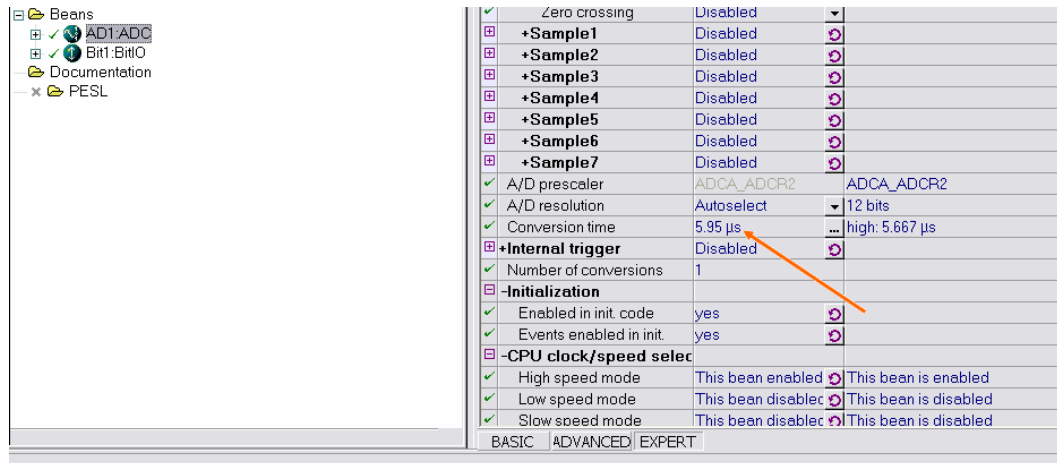
** *Processor expert* o procesador experto es un modo de programación facilitado por la herramienta *Codewarrior* que se apoya en el uso de macros para realizar la programación en lenguaje C.



Fuente: Autores del proyecto

Una vez establecido el tiempo de muestreo, éste se fija en el tiempo de conversión del bean ADC como se muestra en la figura 21.

Figura 21. Ajuste del tiempo de muestreo *



Fuente: Autores del proyecto

* *Conversion time* programa cada cuanto se toma una muestra por medio del conversor

3.2. IMPLEMENTACIÓN DEL ALGORITMO PARA LA OBTENCIÓN DE LA TRANSFORMADA RÁPIDA DE FOURIER (*FFT*) DE UNA SECUENCIA DE DATOS EN EL PROCESADOR DIGITAL DE SEÑALES

Finalmente se presentan dos algoritmos para la obtención de la *FFT*, uno para el *DSP* 56F801y otro para el *DSP* 56F805, los dos dispositivos son tecnología de Motorola.

La diferencia entre los dos códigos implementados es el número de muestras a las que se le obtiene la *FFT*, ya que la capacidad de almacenamiento de datos es distinta en los dos *DSPs* citadas antes.

Cabe anotar que una de las ventajas de los códigos implementados (y además razón por la que se escribieron de esa manera) es su posibilidad de funcionar con cualquier cantidad de muestras (cantidad en base dos) simplemente cambiando unos pocos valores que se indican en los comentarios del código *FFT_N puntos* del Anexo C y considerando la capacidad del *DSP* en el que se va a implementar.

3.2.1. Planteamiento del problema de la Transformada Rápida de Fourier (*FFT*) para la medición del espectro de impedancia eléctrica.

En el trabajo desarrollado por el Doctor Brian H. Brown y su equipo se encuentra que cuando la célula pasa de estadíos normales a estadíos de precáncer y cáncer, ésta presenta cambios en su forma (alteraciones en el núcleo y la membrana) y en la manera en que se dispone en el tejido cervical (crecimiento anormal de la célula reduciendo el espacio intercelular), hace que se presenten variaciones en su

comportamiento eléctrico¹⁸, permitiendo así la posibilidad de diagnosticar la etapa cancerígena en que se encuentra el tejido mediante el estudio de su impedancia eléctrica.

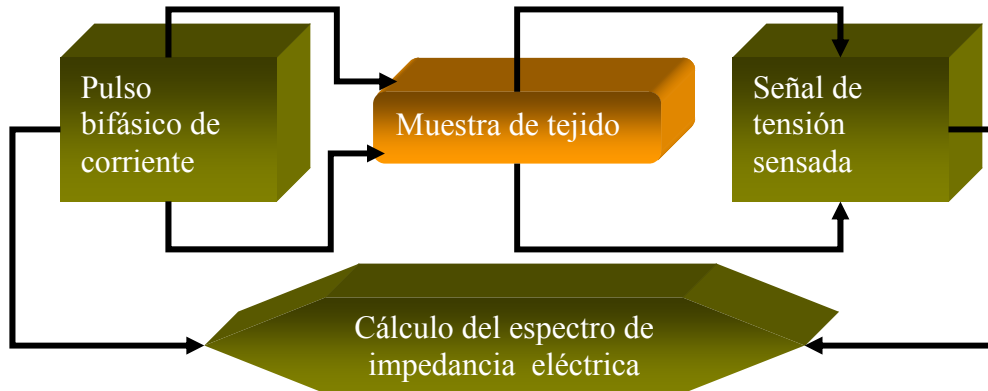
En el cálculo de la impedancia eléctrica hecho por el Doctor Brown y su equipo, se realiza la medición de la tensión que se produce en el tejido, al inyectar corriente siguiendo el método de los cuatro electrodos (figura 23). Estas mediciones se efectúan a diferentes frecuencias con el fin de conocer los parámetros R (resistencia a bajas frecuencias), S (resistencia predominante a altas frecuencias) y C (efecto capacitivo del tejido), los cuales representan respectivamente la resistividad de los fluidos entre células, la resistividad de los fluidos internos de la célula y los efectos capacitivos de la membrana celular. Una vez obtenidos estos parámetros y aplicando algún modelo de detección de parámetros como el de Cole-Cole¹⁹, se puede obtener la impedancia del tejido en estudio y determinar el estado, en el que se encuentre (normal, neoplasia (precancer), cáncer invasivo) (figura 24).

Aunque el método usado por el Dr. Brown arroja resultados de gran importancia, es impráctico a la hora de realizar el tamizaje de un gran número de mujeres, sobre el que se desee obtener información en un amplio espectro de frecuencias, ya que para cada frecuencia en cada mujer se debe hacer una medición. Lo que se propone es obtener con una sola medición toda la información del espectro de frecuencias, aplicando Transformada Rápida de Fourier a los datos de tensión y de corriente.

¹⁸ D.C. WALKER, BROWN Brian H, HOSE D.R., SMALLWOOD Rod H, *"Modelling the Electrical Impedivity of Normal and Premalignant Cervical Tissue"*, Department of Medical Physics and Clinical Engineering University of Shiffeld. Electronics Letters Online No. 20001118, 7 July 2000

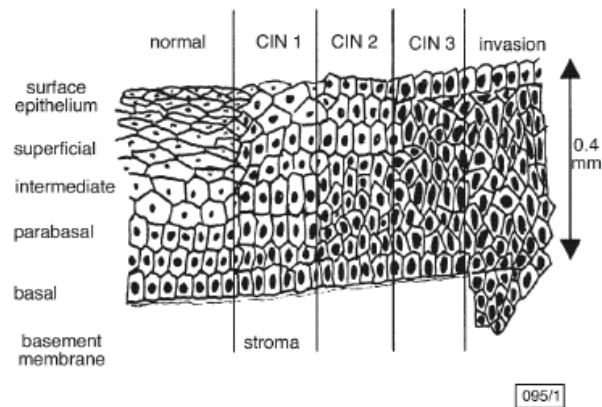
¹⁹ Opcit. BROWN, pp 19

Figura 23. Diagrama de bloques para la obtención del espectro de impedancia eléctrica



Fuente: Autores del proyecto

Figura 24. Cambios asociados a la estructura del tejido con la progresión de Neoplasia Intraepitelial Cervical (CIN, Cervical Intraepitelial Neoplasia) en epitelio escamoso cervical²⁰



Fuente: D.C. Walker, B.H. Brown, D.R. Hose and R.H. Smallwood “Modelling the electrical impedivity of normal and premalignant cervical tissue”

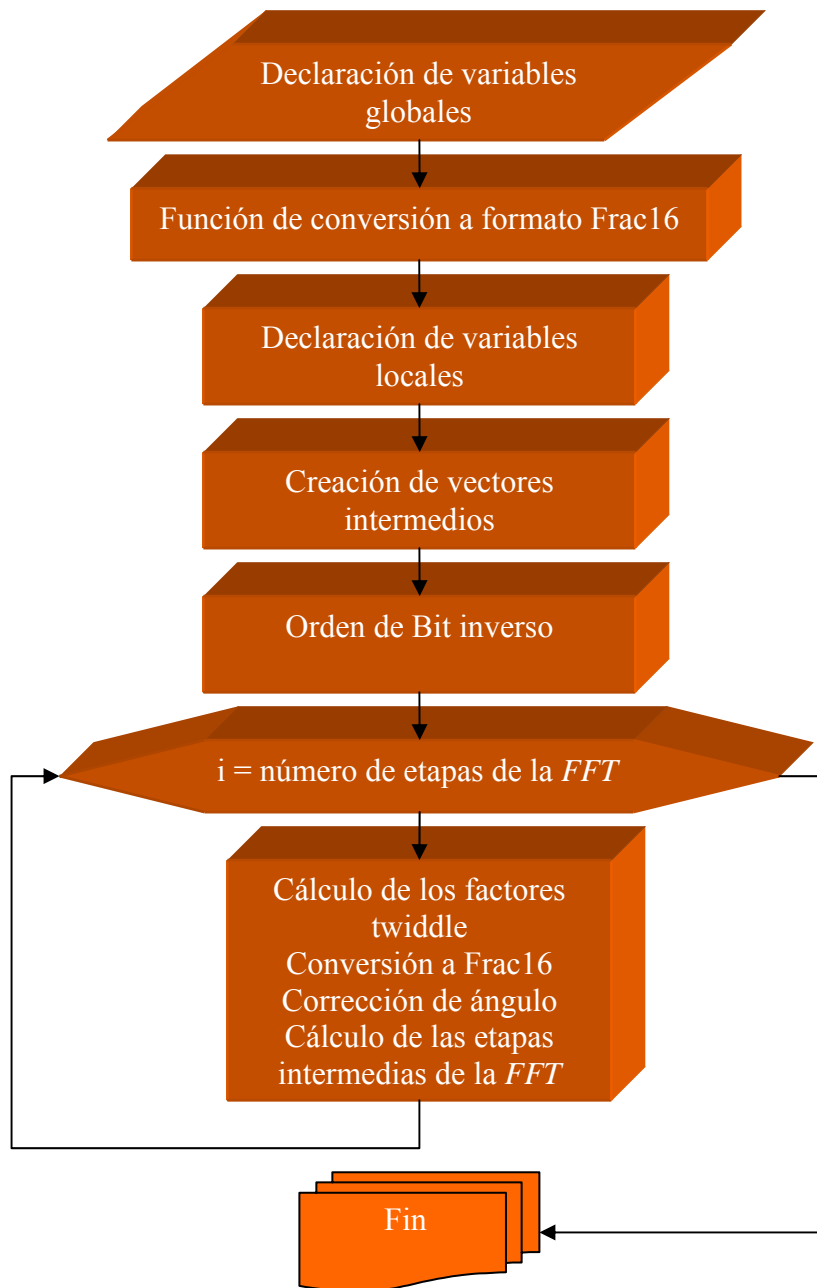
²⁰ibit. BROWN.

3.2.2. Diagrama de flujo del algoritmo implementado para la Transformada Rápida de Fourier (*FFT*)

Las características con las que cuenta este código son las mencionadas en el numeral 2.2. *IMPLEMENTACIÓN DE UNA FFT EN MATLAB*. Entre los aspectos a destacar en el código implementado está la posibilidad de obtener la parte real y la parte imaginaria de la Transformada Rápida de Fourier u obtener su magnitud.

El diagrama de flujo del código implementado en el *DSP56F801* de Motorola para una *FFT* de ocho puntos se muestra en la figura 25 (Este diagrama de flujo es aplicable al código implementado en el *DSP56F805* de Motorola para una *FFT* de 1024 puntos).

Figura 25. Diagrama de flujo del código implementado para la *FFT*



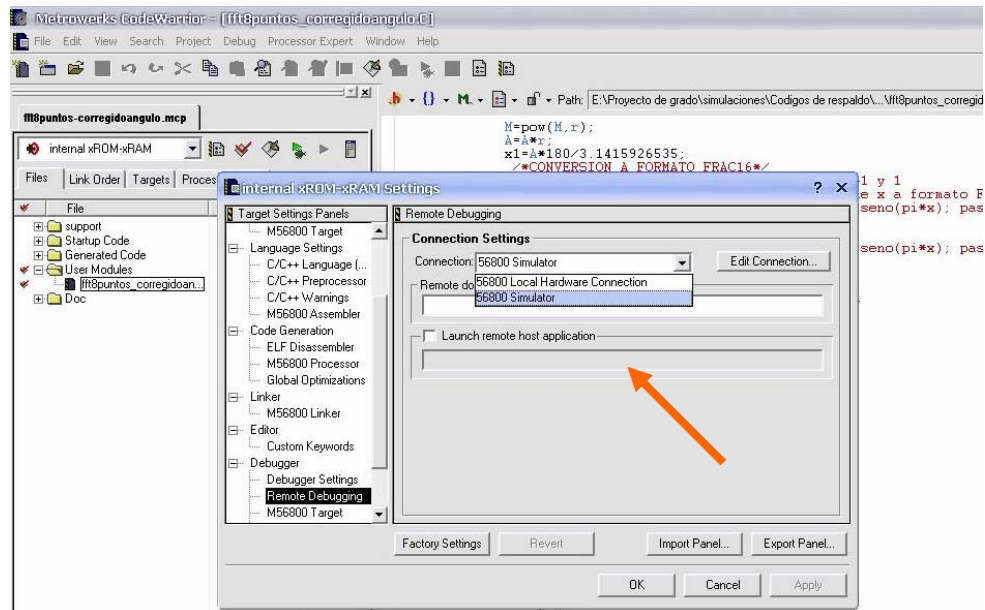
Fuente: Autores del proyecto.

3.2.3. Implementación de la FFT en el procesador digital de señales

Para llegar a la implementación de los códigos en el DSP, se realizó una conversión del código *Algoritmo de FFT creado* en el simulador *Matlab 7.0* (*Algoritmo de FFT creado, Anexo B*) hasta obtener los códigos para el DSP 56F801 y el DSP 56F805 (*FFT_N puntos* y *FFT_N puntos para 56F805, Anexo C* respectivamente).

Los códigos se implementaron en Metrowerks CodeWarrior 6.1 en el modo **Processor Expert** en lenguaje C. En primera instancia se trabajó *codewarrior* en modo simulador y posteriormente en línea con el DSP. Las implementaciones iniciales de los códigos se hacen en el DSP 56F801 por medio del programa *56800 simulador de CodeWarrior* (Figura 26).

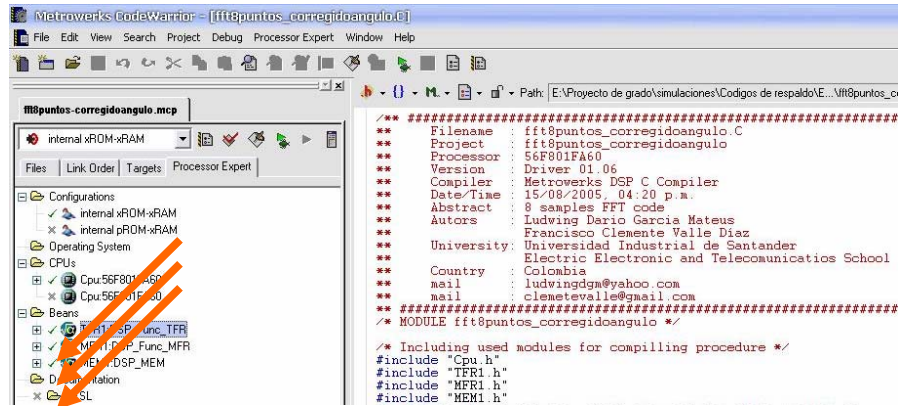
Figura 26. CodeWarrior en modo simulación para la familia 56800



Fuente: Autores del proyecto.

Debido a que se requiere ejecutar funciones seno y coseno es necesario incluir los **beans DSP_Func_TFR*** y **DSP_Func_MFR, DSP_MEM** (Figura 27).

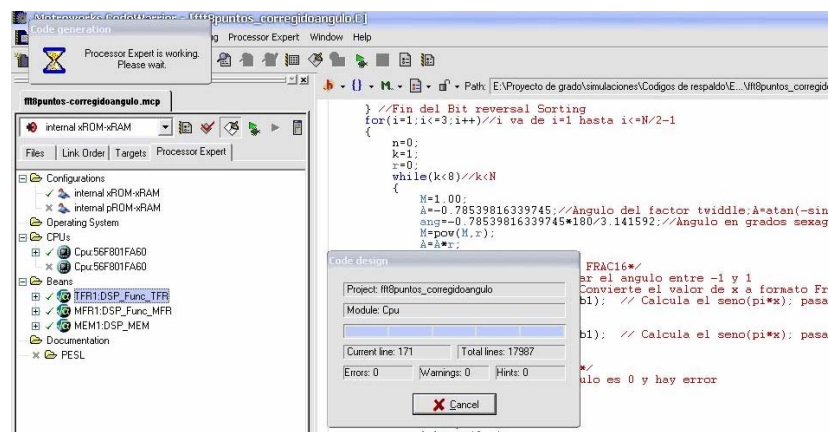
Figura 27. Beans usados para ejecutar las funciones seno y coseno



Fuente: Autores del proyecto.

Después de incluidos los **beans** necesarios y escrito el código se procede a compilar y ejecutar (depurar) el programa (figura 28)

Figura 28. Compilación del proyecto

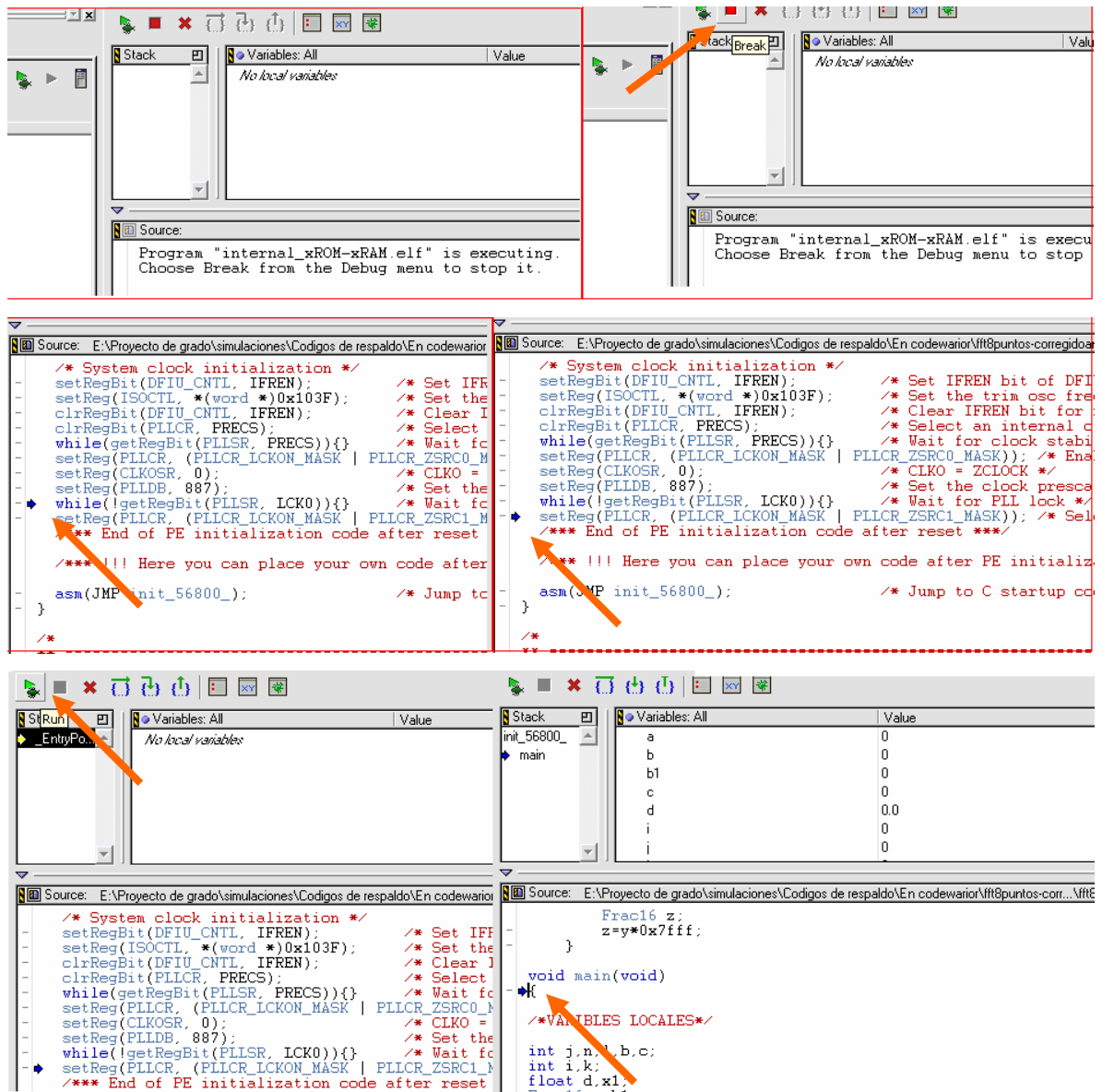


Fuente: Autores del proyecto.

* El bean necesario para obtener las funciones seno y coseno es el DSP_Func_TFR, los dos restantes DSP_Func_MFR, DSP_MEM, se incluyen porque el primero lo requiere para su funcionamiento

Debido a que se está trabajando el *codewarrior* en modo simulador, es necesario detener la depuración y realizar manualmente un paso (avanzar una operación en lenguaje ensamblador) para continuar con la simulación (figura 29).

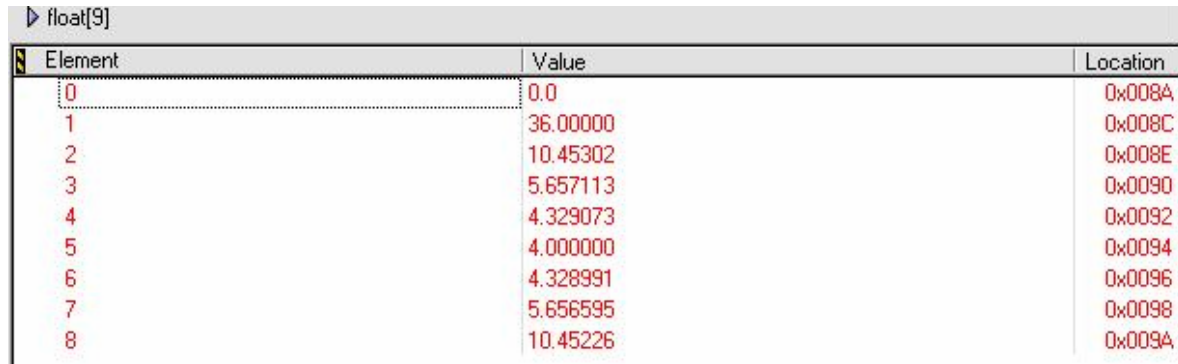
Figura 29. Simulación en modo simulador



Fuente: Autores del proyecto.

Después de ejecutar el programa se visualiza el vector *xreal*. Los resultados de la simulación se observan en la figura 30.

Figura 30. Vector de resultados *xreal* para una *FFT* de ocho puntos cuya secuencia de entrada es una rampa de ocho muestras ** .



Element	Value	Location
0	0.0	0x008A
1	36.00000	0x008C
2	10.45302	0x008E
3	5.657113	0x0090
4	4.329073	0x0092
5	4.000000	0x0094
6	4.328991	0x0096
7	5.656595	0x0098
8	10.45226	0x009A

Fuente: Autores del proyecto.

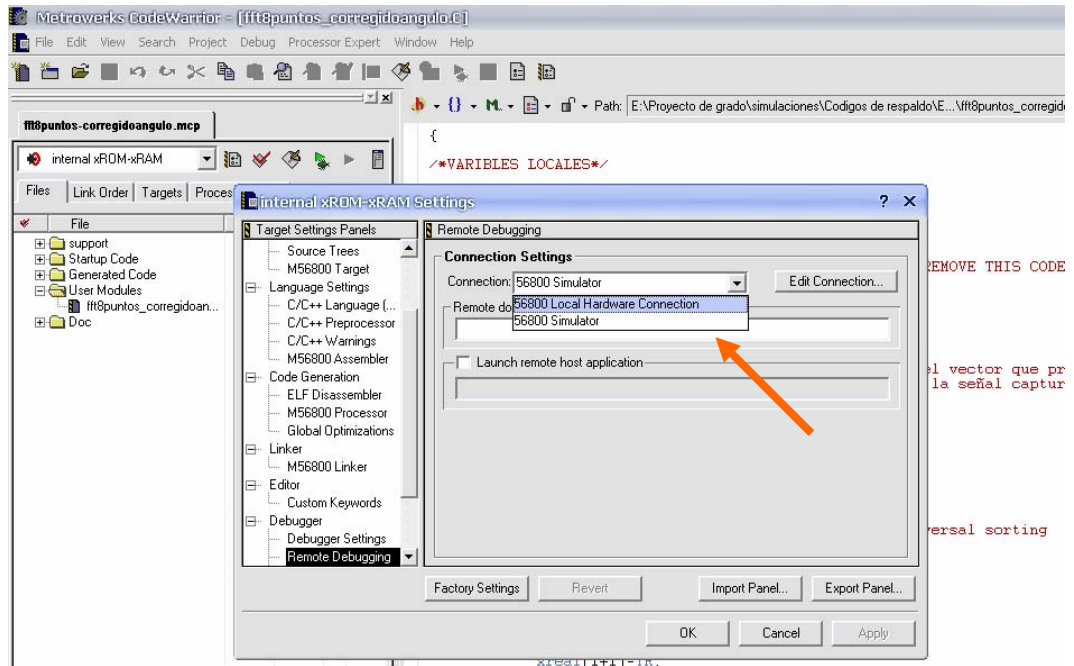
El vector *xreal* contiene las magnitudes de la *FFT* cuyas muestras de entrada (secuencia a la que se aplica la *FFT*) son una señal rampa* .

Una vez se usa el simulador y se verifican los resultados, se ejecuta el programa nuevamente, pero esta vez usando la conexión en línea con el *DSP* para continuar con la validación de los resultados (Figura 31).

** En la figura el elemento cero no hace parte de los resultados, los elementos del vector de resultados comienzan con el elemento uno

* Muestras de la señal de entrada 1, 2, 3, 4, 5, 6, 7, 8

Figura 31. Simulación en línea con el procesador digital de señales



Fuente: Autores del proyecto.

Los resultados que se obtuvieron en línea con el DSP, son los mismos obtenidos con el *codewarrior* en modo simulador.

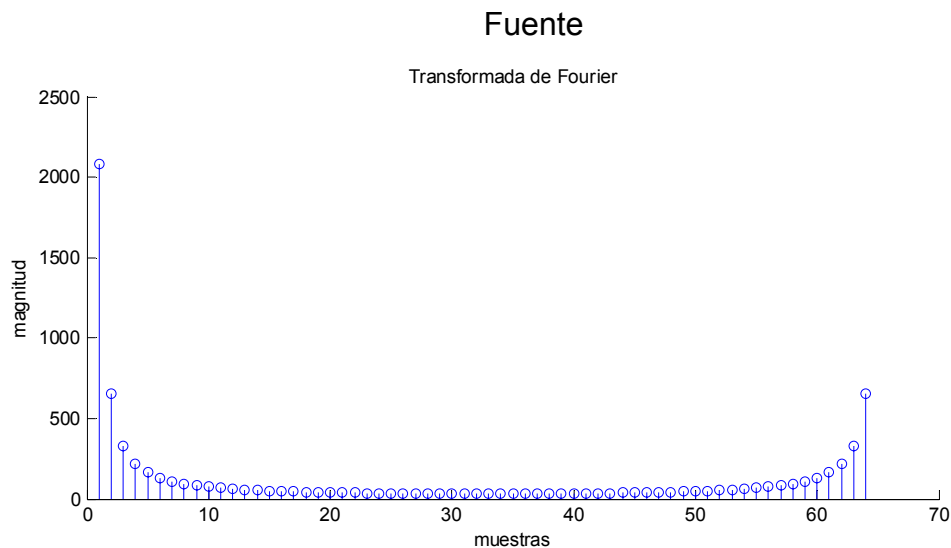
4. RESULTADOS Y ANÁLISIS DE RESULTADOS

En este capítulo se muestran los resultados y el análisis de los mismos cuando se aplicaron los algoritmos de la TMD y el de la FFT en el DSP. Además de la combinación de estos para realizar el cálculo del espectro de impedancia eléctrica.

4.1. TRANSFORMADA RÁPIDA DE FOURIER

Las pruebas realizadas con el algoritmo de la transformada rápida de Fourier (*FFT_N puntos*) consistieron en tomar una señal de prueba (rampa de 64 datos) y calcular la transformada de Fourier de la misma usando el DSP 56F801 de Motorola. El resultado obtenido es el mostrado en la figura 32.

Figura 32. Transformada de Fourier de una rampa de 64 datos.

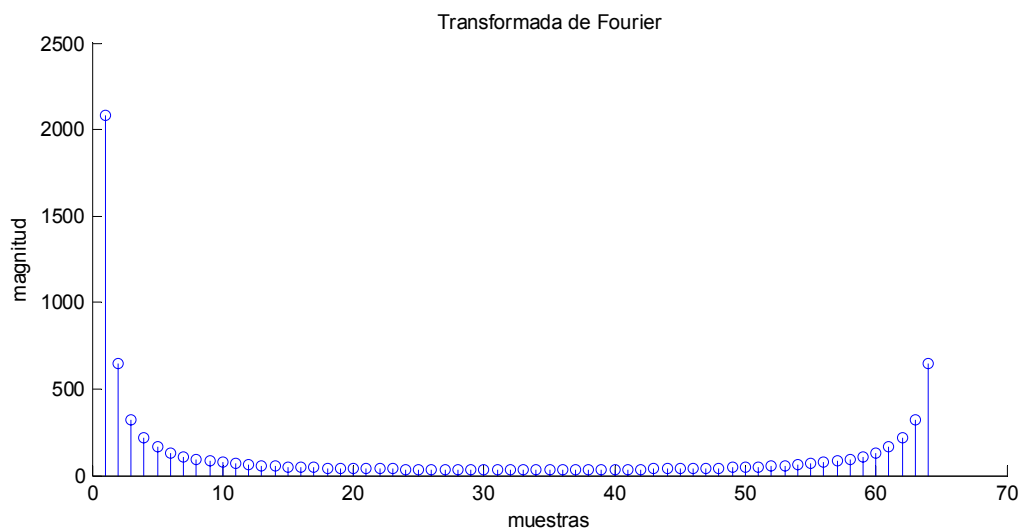


Fuente: Autores del Proyecto.

También se trabajó un algoritmo diseñado usando el **bean** *DFR1:DSP_Func_DFR* (que permite efectuar la *FFT* de un conjunto de muestras)

que se encuentra incorporado al *processor expert* de *Codewarrior 6.1*. Los resultados obtenidos con este segundo algoritmo para la misma señal de prueba se muestran en la figura 33.

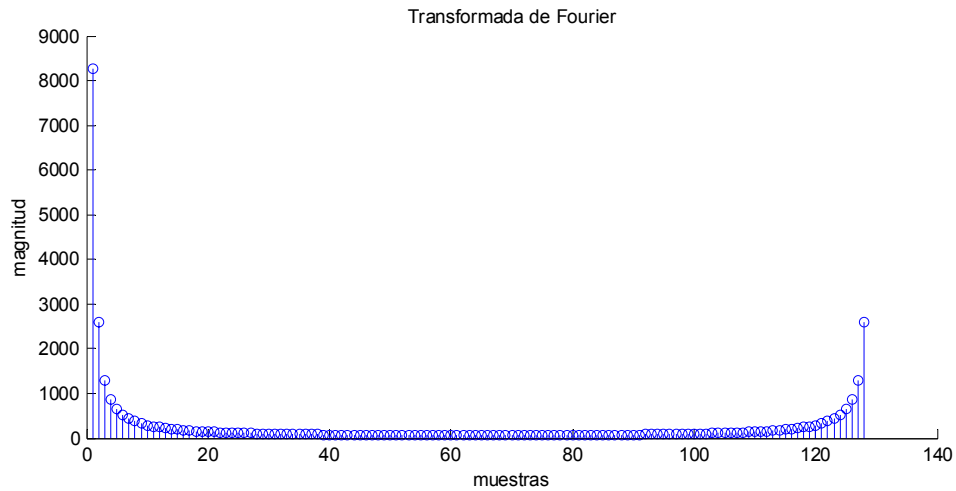
Figura 33. Transformada de Fourier de una rampa de 64 datos obtenida con el DSP 56F801 de Motorola.



Fuente: Autores del proyecto.

Trabajando el DSP 56f801 en modo simulador y usando el algoritmo1 se obtuvo la Transformada de Fourier de una rampa de 128 datos, el resultado se muestra en al figura 34.

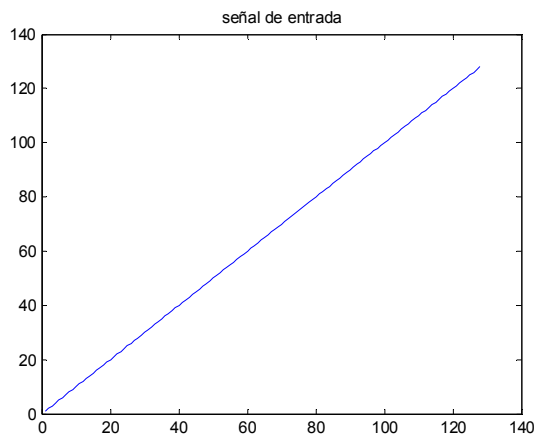
Figura 34. Transformada de Fourier de una rampa de 128 datos.



Fuente: Autores del proyecto.

En la figura 35 se muestra la señal de entrada o señal de prueba utilizada para realizar la validación del código escrito para obtener una Transformada de Fourier (*FFT*) base 2 (Esto se realiza con una modificación del código 7 ANEXO 2 para trabajar a 128 muestras).

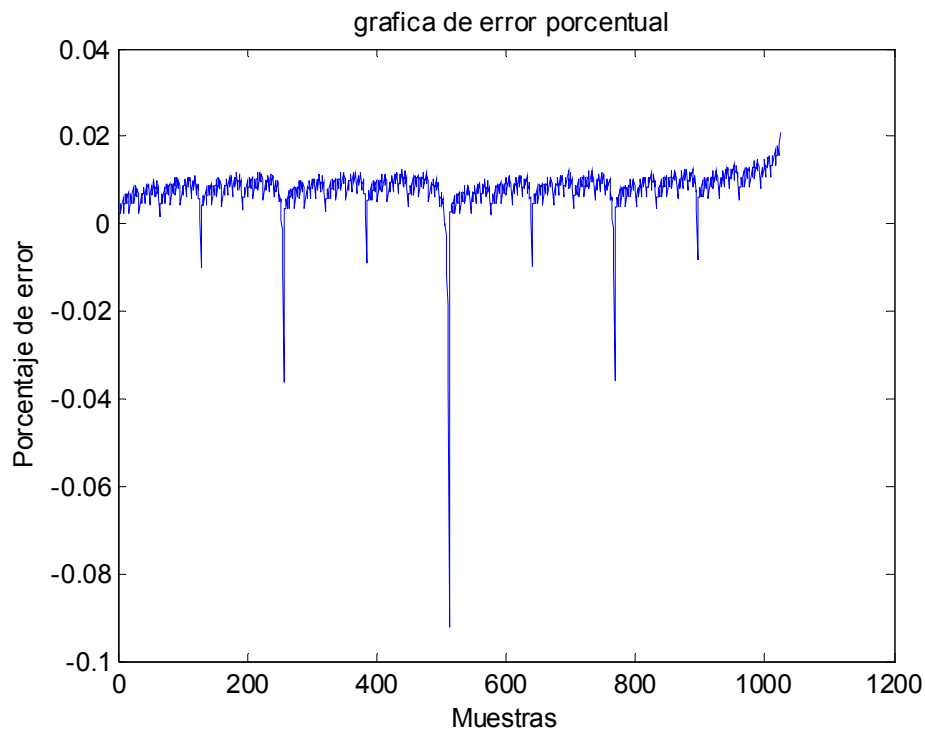
Figura 35. Señal de entrada



Fuente: Autores del proyecto.

Esta validación consiste en realizar una caracterización de errores cometidos por el algoritmo. Se comparará el algoritmo creado contra el comando *fft*^{*} de matlab, además también se obtiene el error de comparar el resultado de obtener una FFT usando el bean *DFR1:DSP_Func_DFR* (bean que permite efectuar la *FFT* de un conjunto de muestras) que se encuentra incorporado al *processor expert* de *codewarrior*.

Figura 36. Porcentaje de error entre la magnitud de la FFT obtenida mediante el comando base y la magnitud de la FFT obtenida con el código escrito en Matlab.



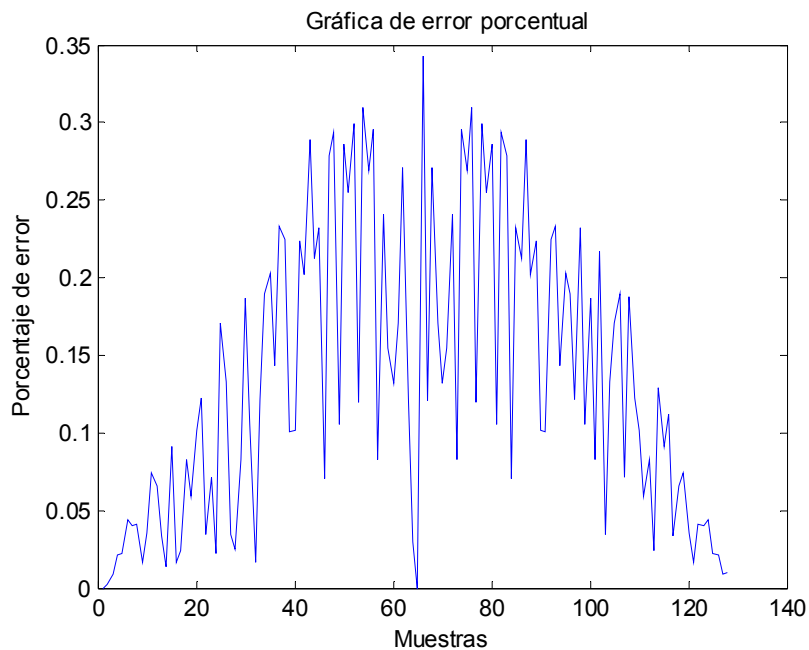
Fuente: Autores del proyecto.

* A este comando en Matlab se hará referencia como comando base

En la figura 36 se visualiza el porcentaje de error de obtener la *FFT* con el algoritmo creado en la herramienta Matlab y la *FFT* obtenida mediante el comando base.

Con base en el código creado en matlab para obtener una *FFT* de *N* muestras, se implementa un código que combina lenguaje C y métodos de programación de beans en el software CodeWarrior 6.1 para obtener una *FFT* de 128 puntos (Figura 37). Los resultados obtenidos se exportan a Matlab con el fin de realizar la comparación y se grafica el error porcentual. Este procedimiento se repite para una *FFT* de 64 puntos, realizada tanto con el bean *FFT (DFR1:DSP_Func_DFR)* como con el algoritmo diseñado en el presente proyecto.

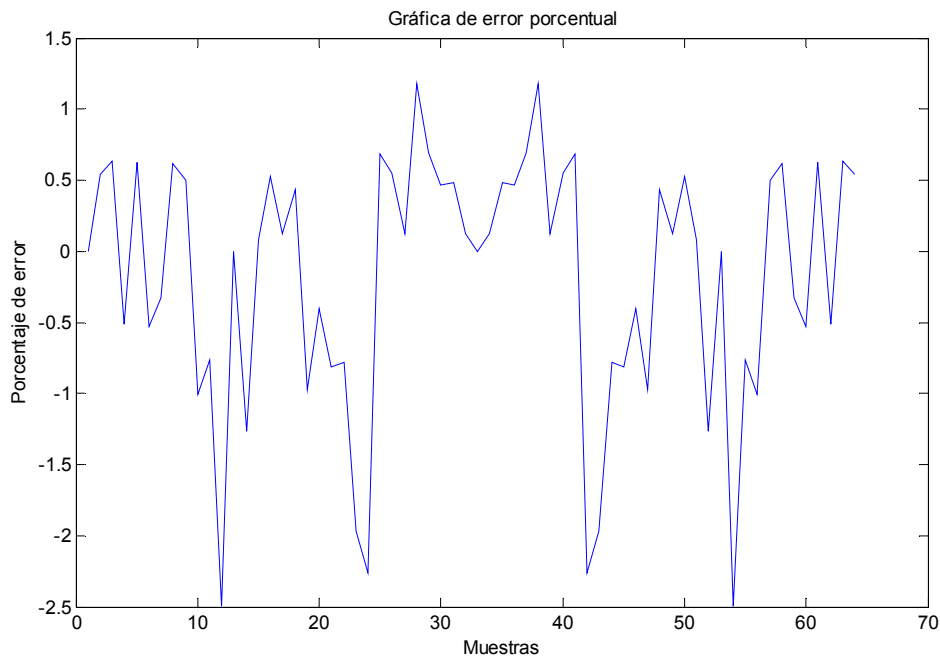
Figura 37. Gráfica del error obtenido al comparar la magnitud de la *FFT* obtenida con el algoritmo *FFT_128* y la magnitud de la *FFT* obtenida con el comando base de Matlab, cuando la señal de entrada es una rampa



Fuente: Autores del proyecto.

El error de la figura 37 (algoritmo de FFT_128 compilado en modo simulador) se ve aumentado al momento de exportar los datos del codewarrior a Matlab ya que el vector original del cálculo de la FFT se encuentra en formato “float” y debe ser convertido a formato “int”, para poder ser exportado a matlab, en este paso se pierden alrededor de tres cifras decimales que deben ser aproximadas a solo una (este aumento en el error se presenta siempre que se va a exportar datos del DSP hacia MATLAB, como es de suponer también se encuentra presente en el resto de las gráficas de error porcentual). Se realizó una FFT de 128 puntos ya que éste es el máximo tamaño que soporta la tarjeta de desarrollo dsp56F801 demo.

Figura 38. Gráfica de error porcentual obtenida al comparar la FFT obtenida con el bean *DFR1:DSP_Func_DFR* y la FFT obtenida con el comando base de Matlab, cuando la señal de entrada es una rampa

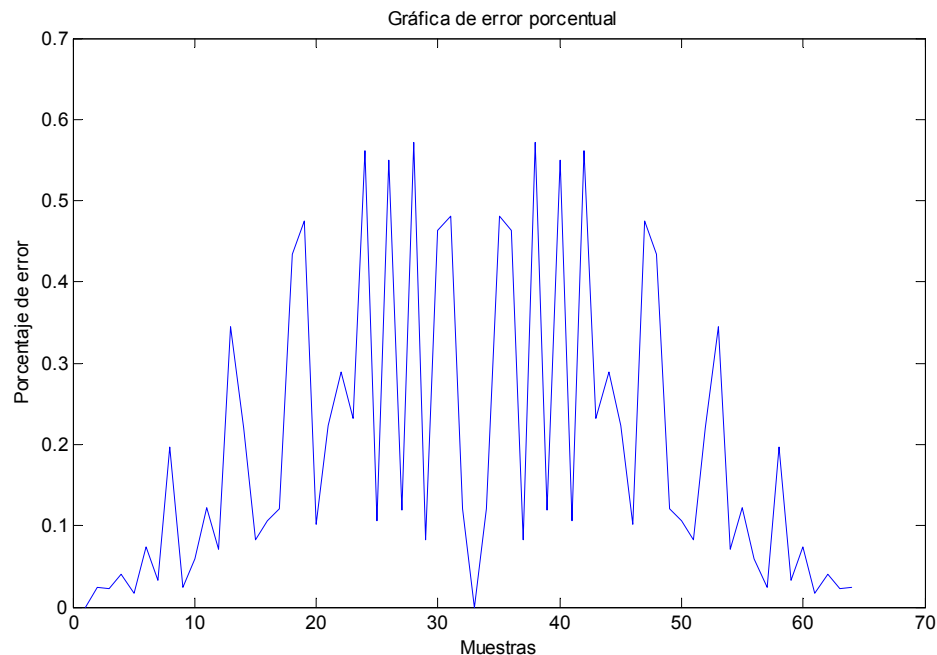


Fuente: Autores del proyecto.

La figura 38 muestra el error generado al comparar la FFT de 64 muestras obtenida por medio del bean *DFR1:DSP_Func_DFR* de CodeWarrior y el comando base. Para esto se usa una señal de entrada tipo rampa de 64 muestras. Aunque el error que presenta es mayor que el obtenido con el código creado por los autores en Codewarrior6.1, la FFT obtenida con mediante el bean *DFR1:DSP_Func_DFR* se implementó en los códigos finales para la obtención del espectro de impedancia eléctrica, ya que presenta menos inconvenientes con el manejo de memoria. El porcentaje de error medio cuadrático obtenido con base en los valores de la figura 38 se muestra en la ecuación (15).

$$\varepsilon_{med_cuadratico} = \sqrt{\sum \frac{x^2}{n}} = 1,6026 \quad (15)$$

Figura 39. Gráfica de error porcentual del algoritmo FFT_64



Fuente: Autores del proyecto.

La figura 39 representa el error obtenido al comparar el algoritmo FFT diseñado en el presente proyecto (FFT_64 muestras compiladas en línea con el DSP) y la FFT obtenida mediante el comando base. La señal de entrada usada es una rampa de 64 muestras. El porcentaje de error medio cuadrático obtenido con base en los valores de la figura 39 se muestra en la ecuación (16).

$$\varepsilon_{med_cuadratico} = \sqrt{\sum \frac{x^2}{n}} = 0,4045 \quad (16)$$

Como puede observarse, al comparar el error medio cuadrático para la FFT obtenida mediante el bean *DFR1:DSP_Func_DFR* con el error medio cuadrático de la FFT obtenida con el código algoritmo FFT_64, en el primer caso el porcentaje de error es mayor que en el segundo, lo cual hace que el algoritmo FFT_64 sea más exacto. Sin embargo, el algoritmo diseñado con el bean realiza un manejo más eficiente de memoria, aspecto que solo puede apreciarse al momento de la implementación en el DSP.

4.2. TEORÍA DE MUESTREO POR DESFASE

Las pruebas de muestreo por desfase realizadas se llevaron a cabo con el DSP56F801 de Motorola tomando, como señales de prueba ondas de tipo cuadrado, triangular y senoidal. Estas ondas se tomaron de un generador de señales Hung Chang 9205C de la E3T* a una frecuencia de 143,26 kHz; la frecuencia de muestreo es 145,56 kHz y el número de muestras tomadas es una por periodo de la señal durante 64 periodos.

Tomando la ecuación (1) los datos a reemplazar son:

* E3T Escuela de ingeniería Eléctrica Electrónica y de Telecomunicaciones

$T=6,98\mu s$

$N = 1$ muestra

$M = 64$ periodos

El periodo de muestreo que se obtiene es de $6,87\mu s$.

Debido a la inestabilidad en frecuencia que presenta el generador de señales empleado **, ya que al intentar fijar un periodo de onda de $6,98\mu s$, éste oscila desde $6,96\mu s$ a $7\mu s$ aproximadamente, para una forma de onda cuadrada y al muestrear los dos tipos de onda restantes, esta oscilación aumenta variando entre $6,88\mu s$ y $7,6\mu s$, esto genera un error aleatorio en los datos tomados.

Adicional al error que imprime la pequeña oscilación de la frecuencia del generador, la base de tiempo del *DSP* no es completamente estable aunque tenga como referencia el cristal de $8,0$ MHz, por lo que los tiempos en que se toman las muestras no siempre son iguales.

A continuación se presentan algunas de las pruebas hechas *** con ondas triangulares y senoidales. El orden en que se muestran los gráficos es primero la adquisición de muestras, las cuales están en el formato que usa el *DSP*, donde los valores de tensión tomados entre $0V$ y $3,3V$ se traducen a valores entre 0 y 32768 . La siguiente gráfica presenta la conversión de formato a nivel de tensión, de acuerdo con la ecuación (17). La última gráfica muestra los valores de tensión organizados con base en el vector de tiempo creado.

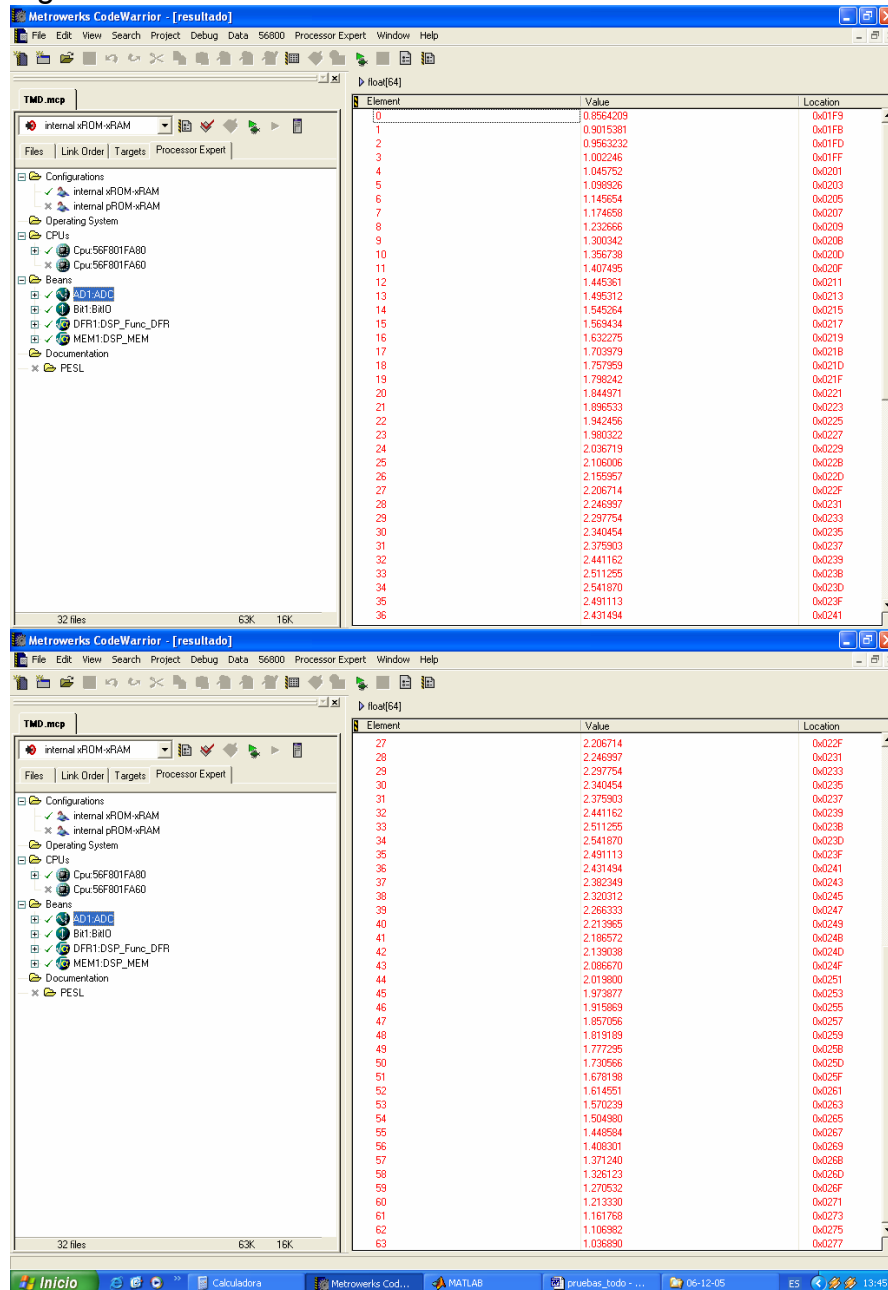
$$V_{tomado} = (V_{adc} / 32768) \times 3,3 \quad (17)$$

** Entiéndase inestabilidad del generador como la oscilación que éste presenta al fijar un periodo de onda específico

*** No todas las pruebas hechas se documentan en este trabajo, ya que su número es grande y el resultado obtenido esta reflejado en lo que acá se presenta

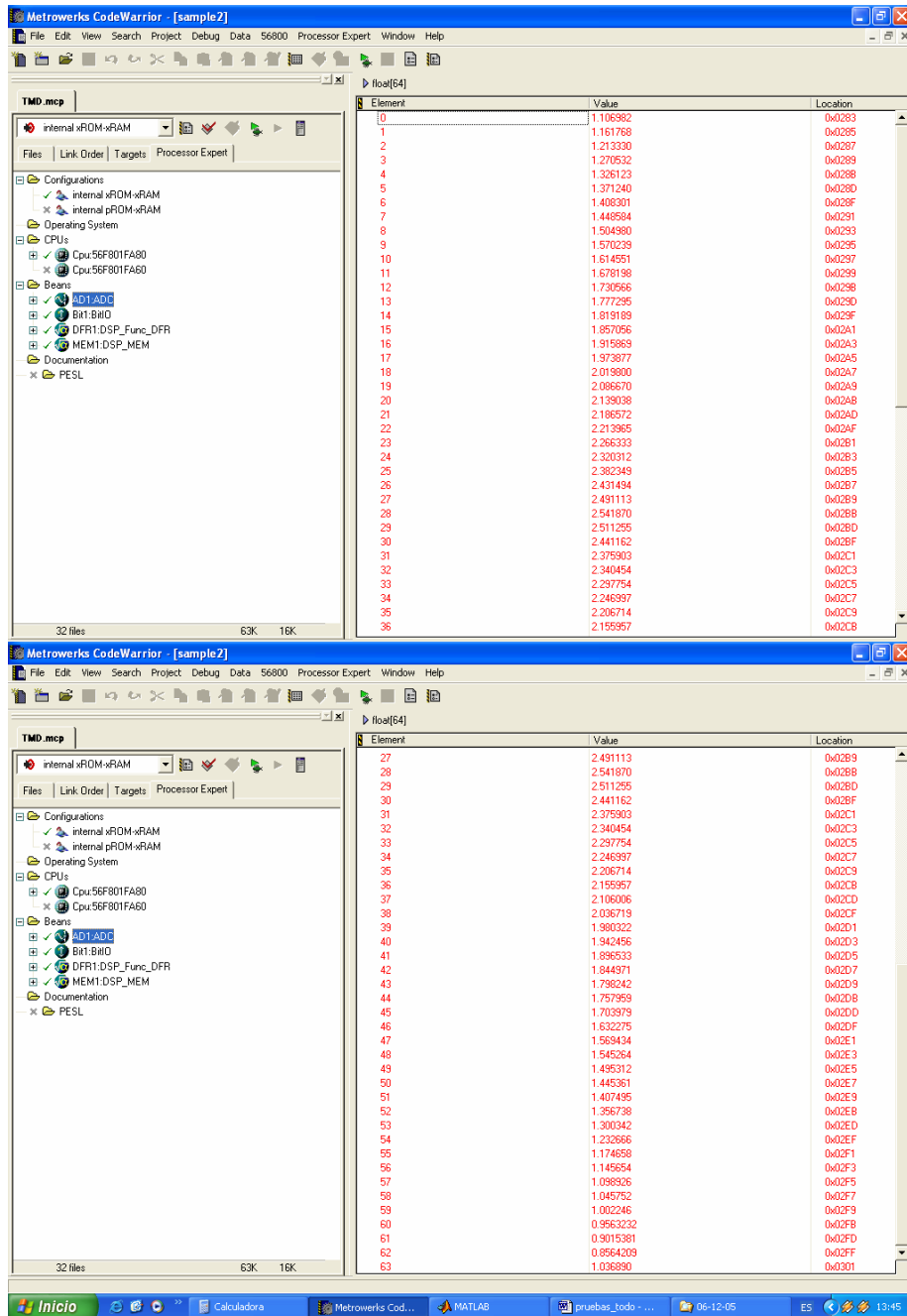
4.2.1 Muestreo de la señal triangular

Figura 40. Datos muestreados



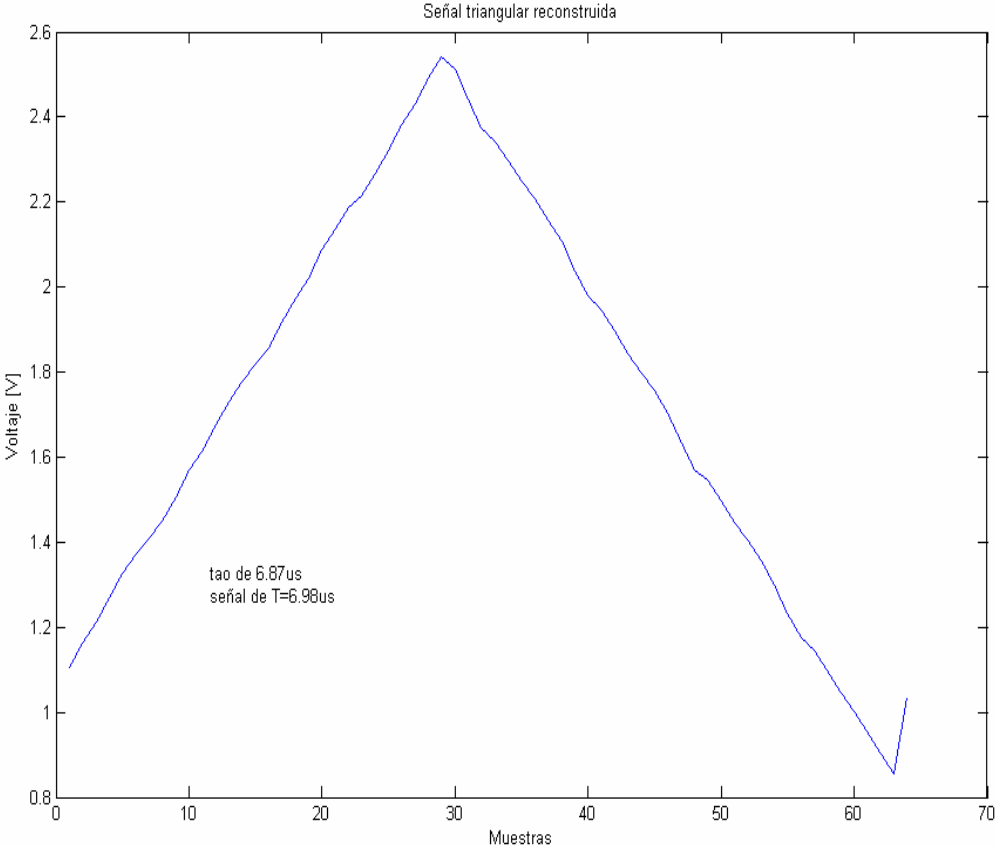
Fuente: Autores del proyecto.

Figura 41. Datos de tensión organizados.



Fuente: Autores del proyecto

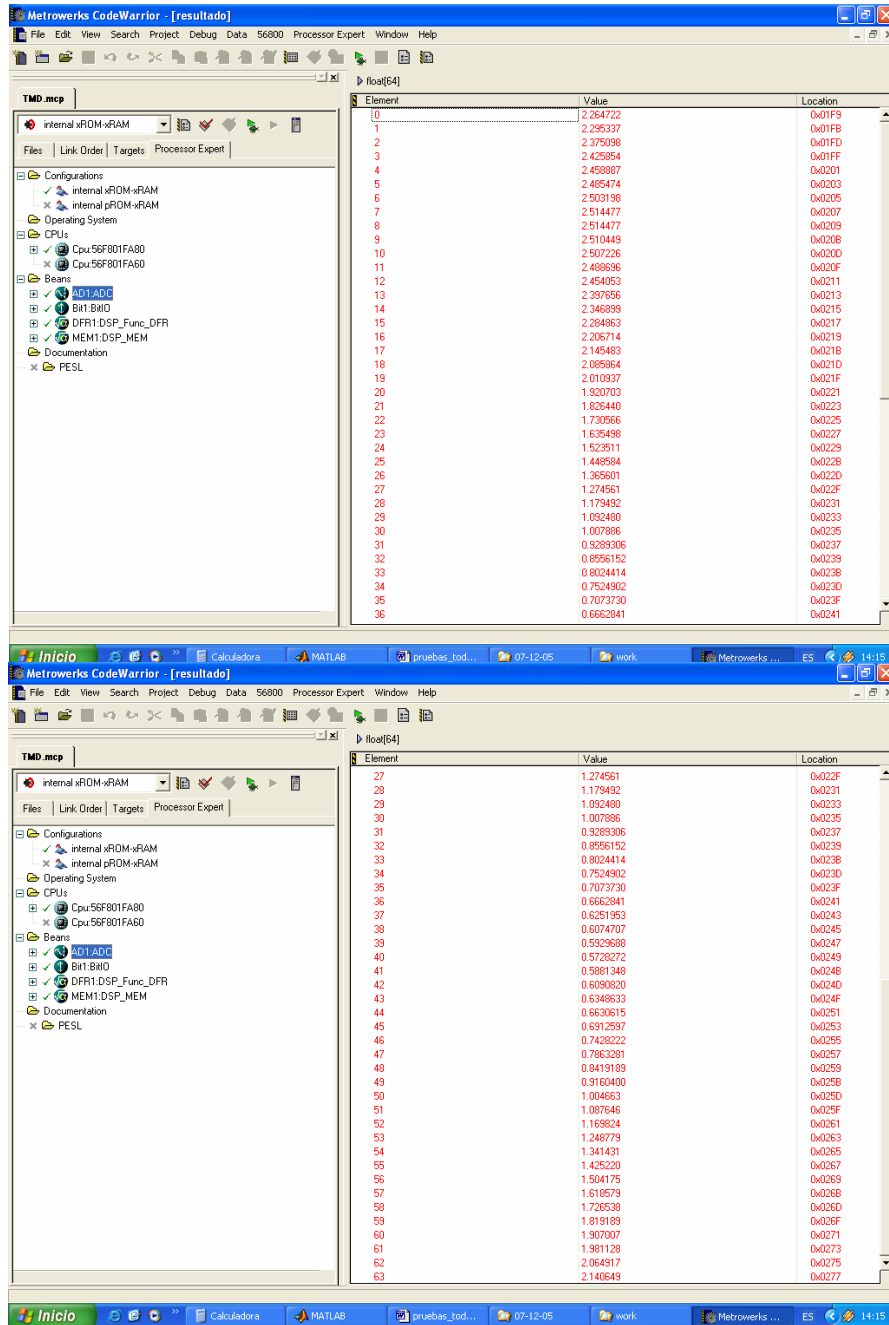
Figura 42. Grafica de señal reconstruida exportando las muestras a la herramienta de simulación Matlab 7.0



Fuente: Autores del proyecto

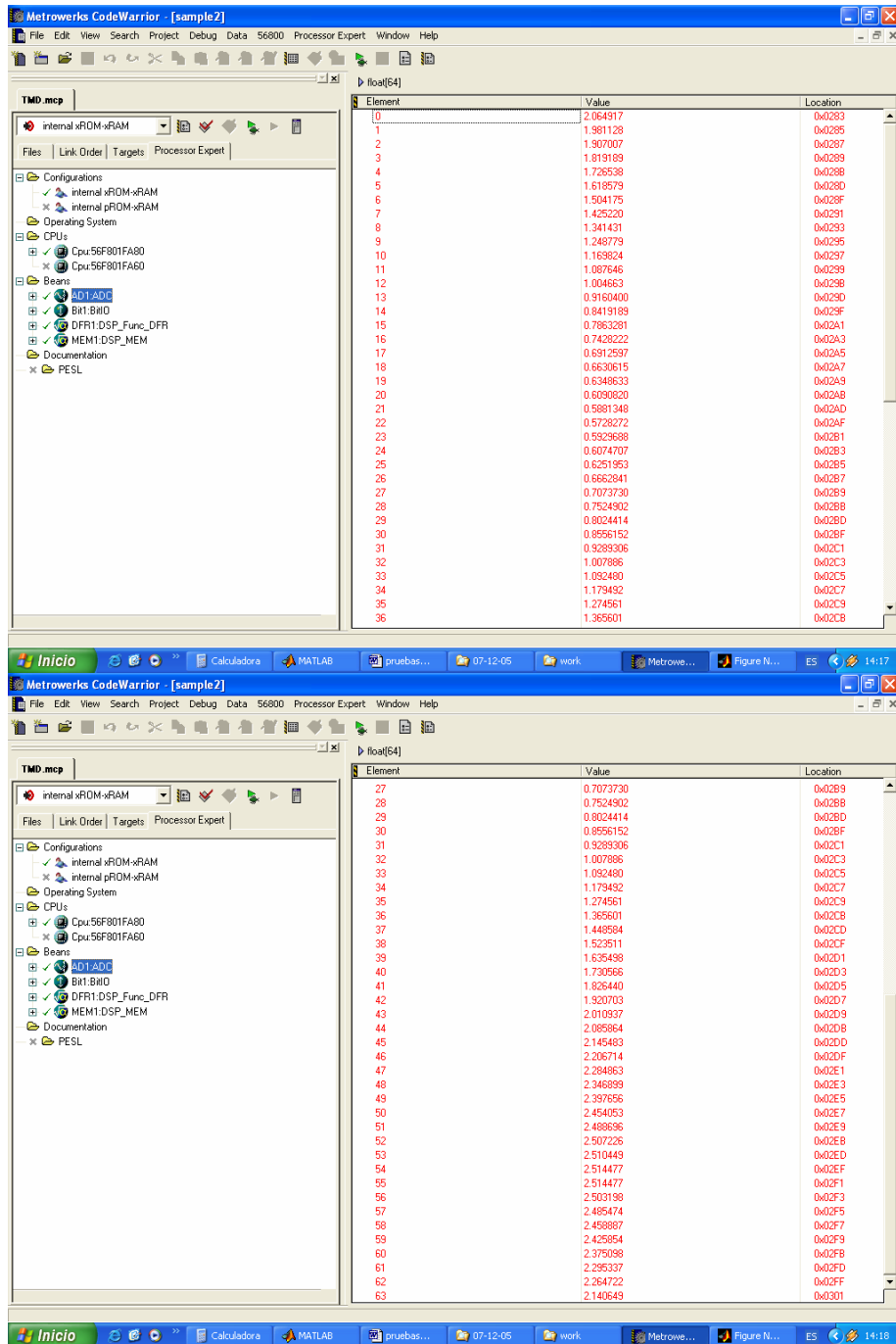
4.2.2. Muestreo de la señal senoidal

Figura 43. Datos muestreados



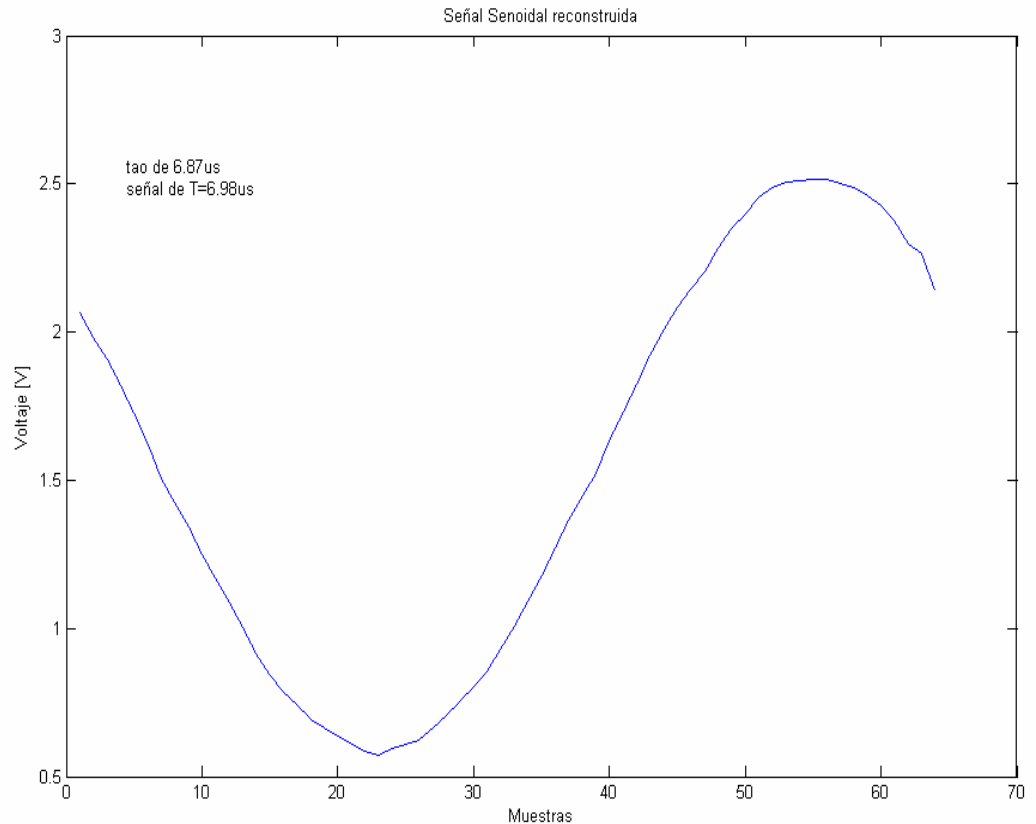
Fuente: Autores del proyecto

Figura 44. Datos de tensión organizados.



Fuente: Autores del proyecto

Figura 45. Gráfica de señal reconstruida exportando las muestras a la herramienta de simulación Matlab 7.0



Fuente: Autores del proyecto

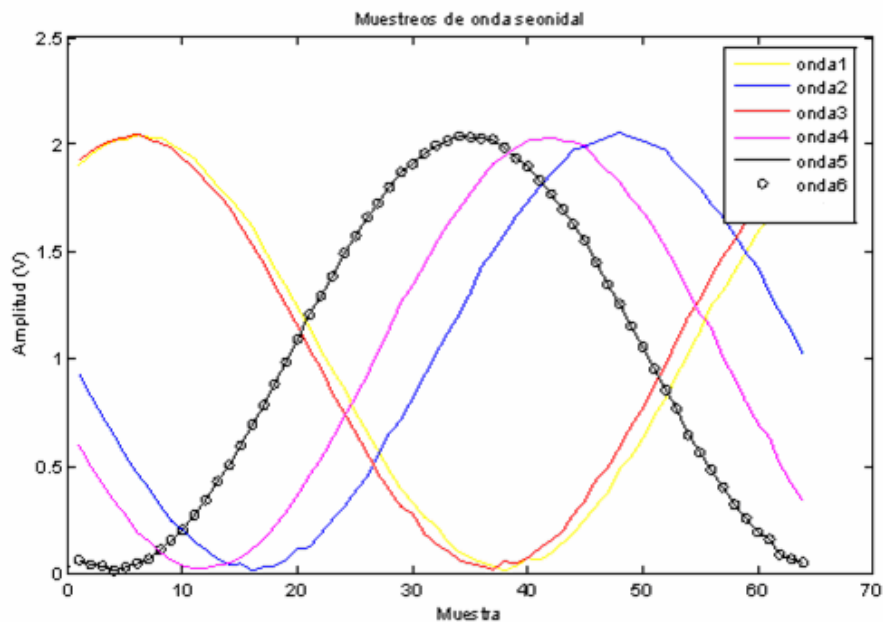
4.2.3. Corrección del error generado por el muestreo

Más importante que hallar el error entre la señal real y la señal que se ha obtenido del muestreo, ya sea aplicando la TMD u otro tipo de técnica o teorema, es lograr eliminar ese error de manera que la onda obtenida después del muestreo esté lo más cercana a la señal real. Para ilustrar esto se hizo un muestreo de una señal senoidal cuyo periodo es $5,58\mu\text{s}$ y de amplitud $2V_{pp}$, por medio de la TMD, la cual

se implementó en un DSP 56F801 de Motorola. La onda es muestreada a 64 muestras por periodo de señal.

El teorema aplicado para esta corrección de error es el expuesto en el numeral 1.3.2 sobre ponderación de error; y el procedimiento que se llevó a cabo fue realizar el muestreo de la señal en seis ocasiones, lo cual se muestra en la figura 46.

Figura 46. Muestras de la señal senoidal de periodo es $5,58\mu\text{s}$ y de amplitud $2V_{pp}$

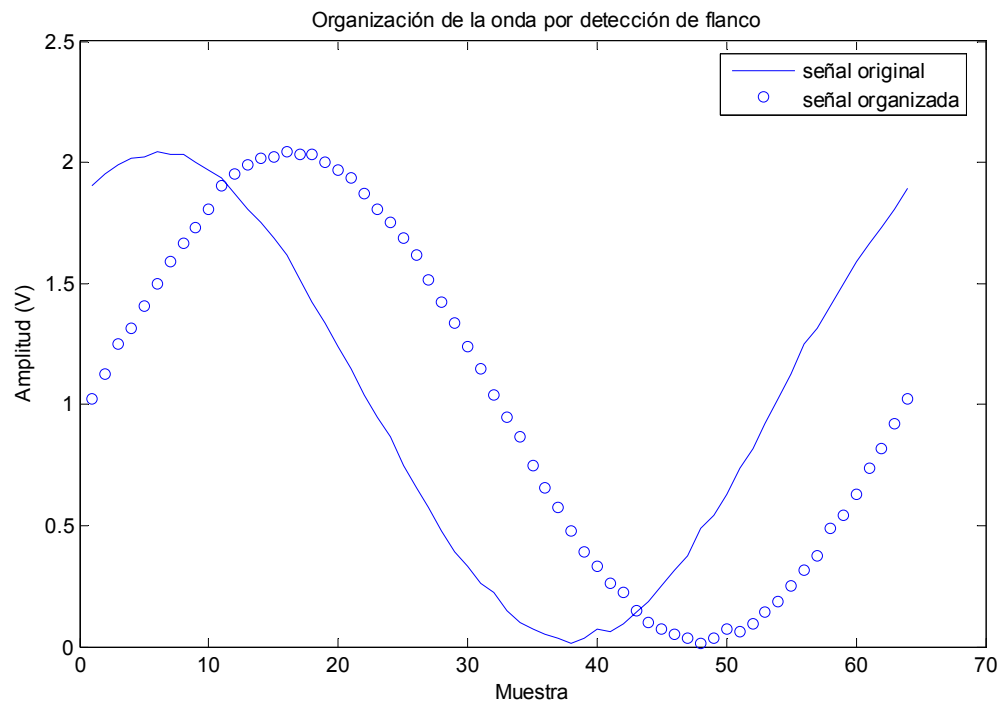


Fuente: Autores del proyecto

Una vez se obtuvieron los seis muestreos por medio del DSP, las muestras se exportaron a Matlab, para posteriormente reorganizarlas por medio de un algoritmo de detección de flanco como se muestra en la figura 47 para que la

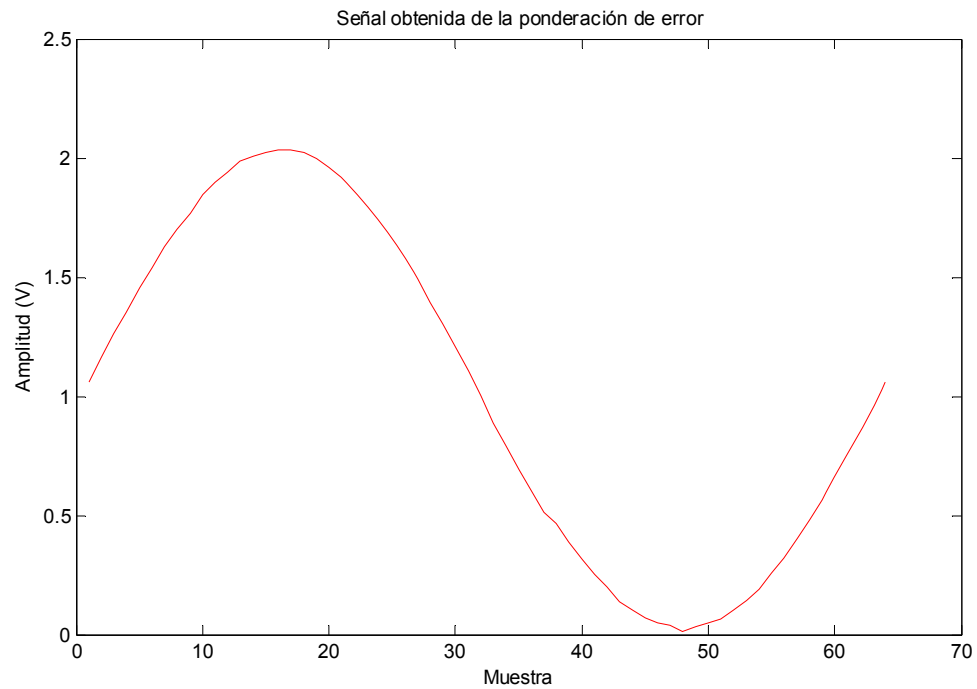
suma de ponderación sea correcta. La forma de onda ponderada se muestra en la figura 48.

Figura 47. Ordenamiento de la onda1 de la figura 46 por algoritmo de detección de flanco



Fuente Autores del proyecto

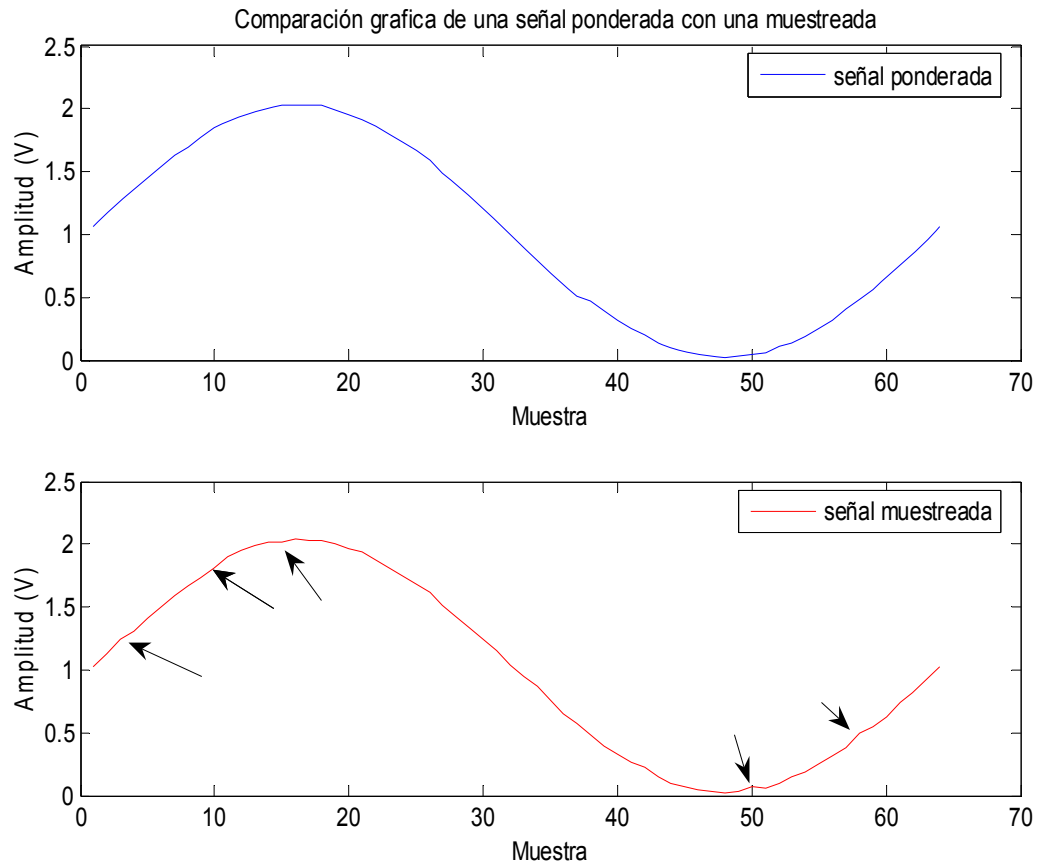
Figura 48. Señal obtenida después de la suma de ponderación



Fuente Autores del proyecto

En la figura 49 se puede ver una comparación de una señal resultante de la ponderación de seis muestreos con una señal producto de un muestreo. En la gráfica de la señal muestreada se indican con flechas algunas partes de la onda que contienen error (señal muestreada), y que son eliminadas al aplicársele la ponderación de error (señal ponderada).

Figura 49. Comparación gráfica de una señal ponderada con una muestreada



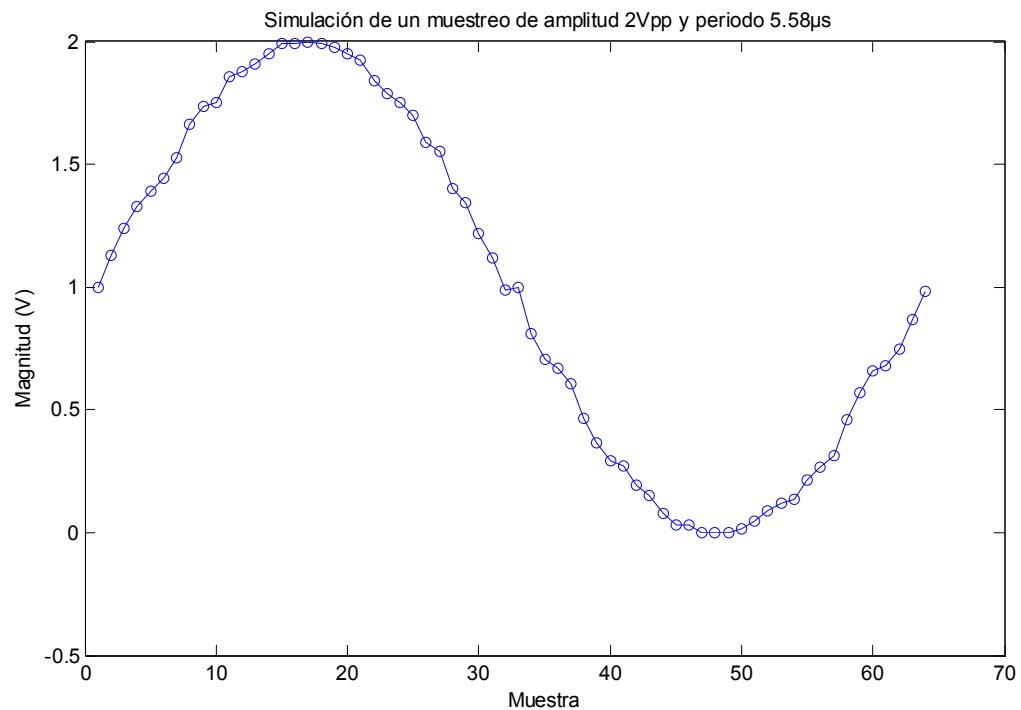
Fuente: Autores del proyecto

Para finalizar el análisis de error y probar que la señal ponderada se acerca más a la señal original que la señal producto de un solo muestreo, estas señales se simulan en la herramienta Matlab y se comparan con una señal teórica (señal sin error) de similares características. La razón por la que no se hace la comparación entre la señal ponderada (obtenida ponderando los muestreos realizados con el

DSP) y una señal muestreada de manera tradicional ^{*}, es que el procesador digital de señales no cuenta con una base de tiempos que permita una frecuencia de muestreo de al menos el doble de frecuencia de la señal muestreada.

En la figura 50 se muestra la simulación uno de los muestreos de la figura 46.

Figura 50. Simulación de muestreo ^{**}



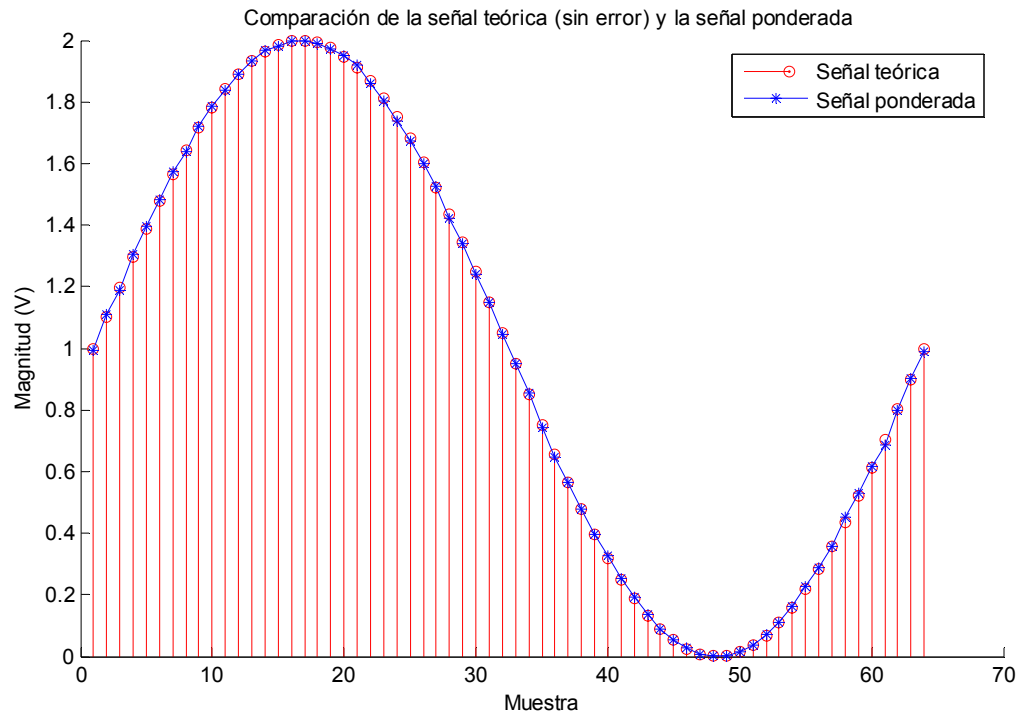
Fuente: Autores del proyecto

De igual manera que con los muestreos obtenidas con el DSP, los muestreos simulados se ponderan. La comparación de la señal teórica con la señal ponderada se muestra en la figura 51.

^{*} Muestreo realizado con una frecuencia de muestreo igual a la frecuencia de muestreo aparente que ofrece la TMD

^{**} Para reproducir esta señal, a las muestras de una señal sinusoidal generadas en matlab, se les suma un pequeño valor aleatorio generado con el comando *rand*, donde el valor es diferente para cada muestra

Figura 51. Comparación de la señal teórica con la señal ponderada



Fuente: Autores del proyecto

Además de la notable mejoría que se aprecia gráficamente entre la figura 50 y la figura 51, el error porcentual entre la señal teórica y la simulación de un muestreo es de 1,75%, mientras que el mismo error entre la señal teórica y la señal obtenida mediante la ponderación de 10 muestreos es de 0,60%.

4.3. ESPECTRO DE IMPEDANCIA ELÉCTRICA

Para la medición del espectro de impedancia eléctrica (como se menciona en el numeral 1.1.2) se parte del comportamiento eléctrico del tejido biológico, el cual se puede representar por una resistencia y una capacitancia en serie, las cuales

representan los fluidos intracelulares y efectos capacitivos de la membrana celular respectivamente, y este arreglo en paralelo con una resistencia, la cual simula los efectos resistivos de los fluidos extracelulares. Partiendo de este modelado, y apoyado en pruebas hechas por otros investigadores en el campo de espectro de impedancia eléctrica²¹, se estableció que el rango de frecuencias adecuado para realizar una medición del espectro de impedancia eléctrica en tejido, que contenga una información completa, debía barrer un rango de frecuencias entre 10kHz y 1MHz, ya que para frecuencias mas bajas a 10kHz solo se obtiene información del contenido extracelular del tejido, y a frecuencias por encima de 1MHz la información obtenida corresponde al contenido intracelular. Debido a que el rango de frecuencias de interés, es relativamente amplio, las investigaciones consultadas de manera previa a la realización del presente trabajo, sobre detección precoz de cáncer, muestran que para obtener por ejemplo, una curva de 128 puntos que describa el comportamiento de un tejido, se requiere la realización de 128 mediciones de espectro de impedancia eléctrica a 128 diferentes frecuencias²² (sin contar las mediciones erradas).

Para hacer más eficiente la toma de datos al momento de tamizar* un número grande de mujeres se propone aplicar la impedancia de transferencia apoyándose en la Transformada de Fourier, ya que al apoyarse en esta técnica, con una sola medición se puede obtener la información completa del estado del tejido para un rango de frecuencias amplio.

4.3.1. Método usado para la medición del EIE

Para la medición del EIE* se uso el modelo eléctrico del tejido expuesto en el numeral 1.1.2 del presente trabajo y se le adicionaron dos resistencias de 800Ω en

²¹Opcit Lozano pp18

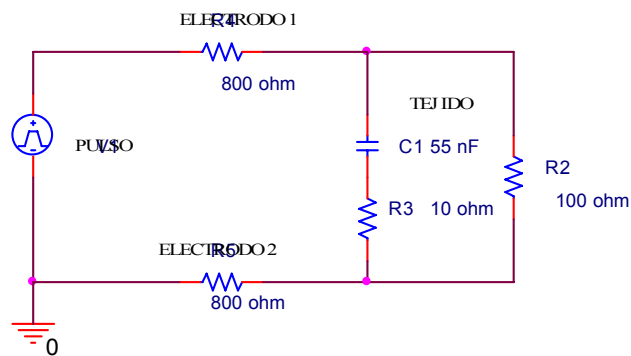
²² Opcit Robles pp18

* Entiéndase tamizar como realizar la toma de datos para obtener el espectro de impedancia eléctrica a un grupo de mujeres

* EIE Espectro de Impedancia Eléctrica

serie con el arreglo para simular la resistencia que introducen los electrodos de prueba del bioimpedanciómetro^{23,24} como se muestra en la figura 52.

Figura 52. Modelo eléctrico del tejido de Cervix



Fuente: Autores del proyecto

Para excitar el tejido se hizo una variación del método de cuatro electrodos, ya que en lugar de excitar con corriente y medir tensión, se excitó con tensión y se midió corriente (en el momento de realizar la medición de corriente es necesario usar un filtro antisolapamiento, este filtro no es suministrado por los autores del proyecto). De esta manera, sobre la resistencia del electrodo 2 se midió la caída de tensión para obtener la corriente ($V_{\text{electrodo2}} / 800\Omega$) que está circulando por el circuito que simula el tejido. Una vez se obtuvo la corriente y la tensión que está atravesando el tejido, se pudo obtener la impedancia de transferencia.

²³ Opcit ROBLES pp18

²⁴ Opcit MIRANDA pp18

4.3.2. Impedancia de transferencia aplicando Transformada Rápida De Fourier

Para obtener el EIE por medio de la impedancia de transferencia aplicando FFT, se midió la tensión y la corriente que atraviesan el tejido, y a cada una de éstas dos formas de onda se les aplicó la FFT y se operaron para hallar la impedancia ($FFT_{\text{tensión}} / FFT_{\text{corriente}}$). La ventaja de este método, como se menciona de manera previa en el numeral 3.2.1, es que basta una sola medición a una frecuencia para obtener la información del EIE en un amplio rango de frecuencias.

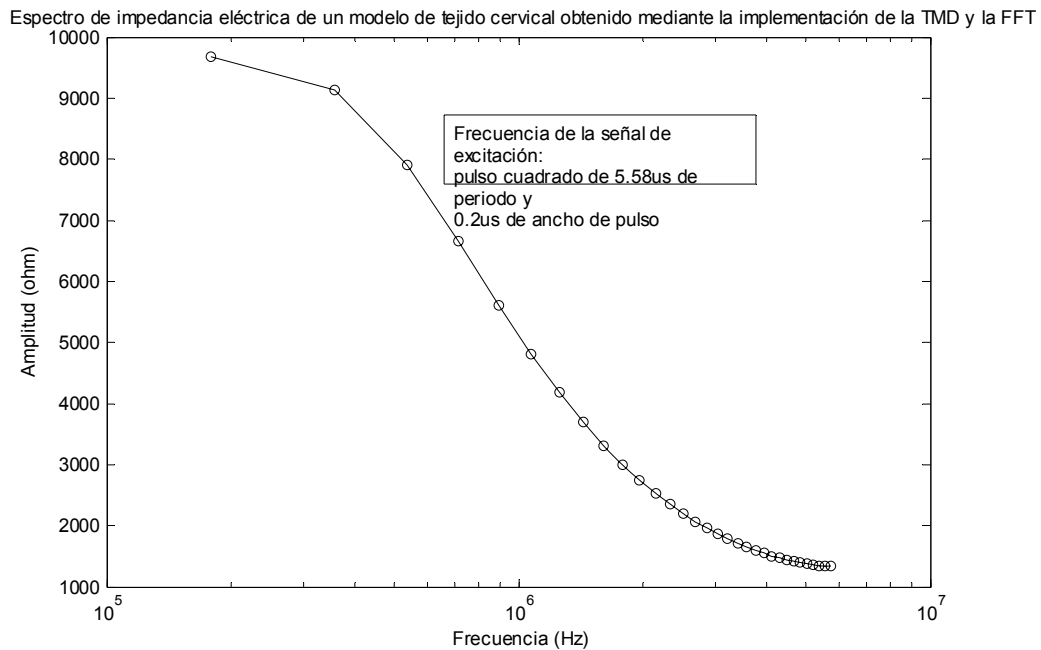
4.3.3. Pruebas de EIE practicadas usando el método propuesto

Con base en los planteamientos hechos de manera previa se realizó la medición del espectro de impedancia eléctrica muestreando las señales de tensión y corriente a dos frecuencias distintas, de las cuales se presenta una. Cabe anotar que la tensión con la que se realizan las pruebas no cumple con los estándares médicos, ya que la fuente de excitación usada es un DSP de iguales características al empleado para realizar las mediciones (56F801 de Motorola), esto debido a que en el momento de la realización de las pruebas no se contaba con una fuente de corriente que cumpliera con los requerimientos y el diseño de una esta fuera de los objetivos del presente trabajo.

Debido a los limitados recursos con que cuenta el DSP 56F801 de Motorola (solo 1Kbyte de memoria **Ram** de datos e igual capacidad de memoria **Ram** de programas), el código a implementar se dividió en tres subcódigos para realizar las mediciones y hallar el EIE (los códigos se muestran en el Anexo C, *Códigos para hallar el Espectro de Impedancia Eléctrica*).

La medición se realizó excitando con un pulso de periodo 5,58 μ s, ancho de 0,1 μ s y una amplitud de 3,3V. Con lo cual se obtiene el espectro mostrado en la figura 53.

Figura 53. Espectro de impedancia eléctrica obtenida a un modelo de tejido cervical



Fuente: Autores del proyecto.

Para la validación de los resultados obtenidos, se toma como base la documentación sobre medición de espectro de impedancia eléctrica hecha por otros grupos de investigación^{25,26}. Al comparar los resultados obtenidos, se ve una clara similitud en las formas de onda y en los valores de impedancia, a excepción de un factor de escala que puede deberse a las condiciones en que se realizaron las mediciones, las cuales son un poco distintas a las usadas por los

²⁵ Opcit Robles pp18

²⁶ Opcit TUSTANLL pp21

investigadores de la literatura consultada. Entre estas condiciones se encuentran un nivel de tensión de 3,3V (nivel mucho mayor al usado en este tipo de mediciones, el cual no supera los milivoltios), un pulso de periodo 5.58 μ s y ancho de 0.1 μ s (para excitación de tejido es recomendado un pulso de 100 μ s de periodo y 1 μ s de ancho de pulso^{27,28}), y la no calibración de los electrodos, la cual se debe hacer en solución salina de manera previa al experimento.

Cabe anotar que al obtener el EIE con una frecuencia de excitación distinta, características como el factor de escala y la amplitud del rango del espectro varían.

²⁷ Opcit PALKO pp19

²⁸ Opcit Brown pp19

5. CONCLUSIONES

En este capítulo se abordan las conclusiones sobre el trabajo realizado, las cuales se exponen en cuatro ítems, en el primero sobre la *FFT*, en el segundo la teoría de muestreo por desfase, y en los ítems tres y cuatro, las conclusiones sobre análisis de error sobre señales muestreadas y espectro de impedancia eléctrica respectivamente.

5.1. TRANSFORMADA RÁPIDA DE FOURIER

Los resultados obtenidos en el numeral 4.1 al comparar el comando *FFT* de la herramienta de software *Matlab 7.0*, con el algoritmo escrito por los autores del proyecto (AnexoB, Código *FFT_N puntos*) y la *FFT* obtenida mediante el *bean FFT* de Codewarrior, muestran que el primero presenta un menor porcentaje de error que el segundo. Esto se evidencia en la figura 39, en la cual una *FFT_64* muestras obtenida con el código escrito, tiene un porcentaje de error medio cuadrático de 0,4045, mientras que el mismo error arrojado por la *FFT_64* obtenida por medio del bean *DFR1:DSP_Func_DFR* que provee Codewarrior es de 1,6026. No obstante, en las implementaciones finales para obtener el espectro de impedancia eléctrica se usa la *FFT* del bean y no el código creado, ya que el primero hace un mejor manejo de memoria que el segundo, porque emplea menos vectores intermedios, aunque el costo del primero con respecto al segundo es menos exactitud en los datos obtenidos.

Una de las anomalías observadas en la herramienta de software Codewarrior versión 6.1, es la discrepancia entre los datos de capacidad obtenidos en el modo simulador y el modo en línea, ya que al ejecutar el código creado para la obtención de la *FFT* en línea con el DSP, la máxima capacidad de muestras posibles es de 64, mientras que en modo simulador se pueden obtener con el mismo código transformadas de Fourier de hasta 128 muestras.

5.2. TEORÍA DE MUESTREO POR DESFASE

El uso de la Teoría de Muestreo por Desfase (*TMD*) permite muestrear señales con una frecuencia de muestreo (casi igual a la frecuencia de la señal) inferior a la frecuencia de Nyquist. Es de resaltar que esta forma de adquisición de señales requiere un adecuado ajuste de la frecuencia de muestreo en el conversor que se use, además de que se debe conocer la frecuencia de la señal, y asegurar que durante el proceso de muestreo, su frecuencia no cambie.

En cuanto a los resultados obtenidos, los cuales se aprecian en las figuras 45 y 42, se ve que las señales reconstruidas tomando una muestra por periodo de señal, representan adecuadamente un periodo de la señal. Acerca del error que está inmerso en la señal muestreada, en lugar de un análisis de error, se propone corregirlo haciendo uso del método expuesto en el numeral 1.3.2.1 por medio del cual se puede eliminar y obtener una señal con un nivel de error menor.

5.3. ANÁLISIS DE ERROR PARA SEÑALES MUESTREADAS

Aunque el abarcar un método para la corrección del error de señales producto de un muestreo no estaba dentro de los objetivos del presente trabajo, surgió la necesidad de tratarse debido a que al intentar sincronizar el CAD* del DSP con la señal a muestrear, se apreciaban pequeñas oscilaciones en la frecuencia de ésta, lo cual podía llevar a errores en la señal adquirida. Para solucionar esto se hizo el análisis de error presentado en el numeral 1.3.1 y se propuso un método en el numeral 1.3.2.

Una aplicación práctica del método de ponderación de error se presentó en el numeral 4.2.3, con una señal senoidal de $5,58\mu\text{s}$ y de $2V_{pp}$, para lo cual se obtuvo una reducción del 1,15% del porcentaje de error de la señal producto de la

* CAD entiéndase como conversor analógico digital

ponderación de diez muestreos con respecto a la señal obtenida en un solo muestreo. Sin embargo si se pondera un mayor número de muestreos, por ejemplo 20, el mismo error se reduce en un 5,51%.

5.4. ESPECTRO DE IMPEDANCIA ELÉCTRICA

Los resultados obtenidos al aplicar la FFT para obtener el EIE permiten conocer el comportamiento de un tejido en un amplio espectro de frecuencias con solo una medición (numeral 4.3.3). A diferencia de los trabajos de este tipo consultados de manera previa, como el desarrollado por el Dr. Brown de la Universidad de Shifield, que se describen a lo largo de los numerales 1.1.2 a 1.1.5, en el cual, para obtener un espectro con 7 muestras realizó 7 mediciones en cada paciente, mientras que en las pruebas hechas para el presente trabajo, con una medición se obtienen para el mismo rango 64 muestras (numeral 4.3.3).

Cabe aclarar que los resultados obtenidos, muestran una clara similitud en las formas de onda y en los valores de impedancia, a excepción de un factor de escala que puede deberse a las condiciones en que se realizaron las mediciones, las cuales se mencionan en el numeral 4.3.3.

Exceptuando la discrepancia en factores de escala, la cual es solucionable, se puede apreciar el aporte que se hace con el presente trabajo a la ciencia médica al proveer una forma de reducir los tiempos de tamizaje y de esta manera hacer menos traumático el examen de detección precoz de cáncer de cervix.

5.5. FORMACIÓN PERSONAL

A través de la realización del trabajo de grado, se adquirió la formación necesaria para enfrentar un proyecto teniendo en cuenta sus etapas de preparación,

planteamiento del problema y resolución del mismo. Además que una de las mejores formas de sortear un gran obstáculo, es separarlo en mas pequeños, lo cual permite abarcarlo de manera mas eficiente e identificar posibles errores en los que se este incurriendo.

6. RECOMENDACIONES

6.1. ACERCA DE CODEWARRIOR 6.1

Al usar el bean *DFR1:DSP_Func_DFR* que provee Codewarrior versión 6.1 para la obtención de la FFT, se debe prever que el valor del coseno de cero grados (grados sexagesimales) que genera es erróneo y se debe corregir.

Para corroborar que un código implementado y compilado en modo simulador no supera la capacidad de memoria (tanto de datos como de programas), es necesario repetir la compilación en modo en línea, ya que se comprobó que estos dos datos no siempre son iguales.

El tiempo de muestreo que presenta el conversor analógico – digital del DSP 56F801 de Motorola, ya sea en el tiempo de conversión del **bean** o manejando el tiempo de conversión directamente por registros, corresponde solo al tiempo que se demora en tomar una muestra y que esta esté lista para ser leída, pero para aplicaciones donde el número de muestras a tomar es mayor a uno, y además estas muestras deseen almacenarse, este tiempo se aumenta en casi un 30%. Un ejemplo de esto se presenta cuando se desea tomar un *set* de muestras y almacenarlas a razón de 5,58µs por muestra, en el bean el tiempo de conversión a fijar es de 3,96µs.

6.2. TEORÍA DE MUESTREO POR DESFASE TMD

Para que la aplicación de la teoría de muestreo por desfase se aplique de manera satisfactoria es necesario tener una señal a muestrear estable y un dispositivo para muestreo confiable, de lo contrario, para minimizar el error se sugiere aplicar a la señal muestreada una técnica de corrección de error como las expuestas en el numeral 1.3.

6.3. CÁLCULO DEL ESPECTRO DE IMPEDANCIA ELÉCTRICO EN TEJIDO CERVICAL IMPLEMENTANDO LA TMD

Para lograr obtener un amplio espectro de frecuencias es importante tener un periodo de muestreo aparente pequeño, ya que entre más pequeño sea este, mayor será el ancho de banda obtenido.