



# UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

## Medición de desplazamiento y ángulo con alta resolución utilizando el DSP TMS320C50

T E S I S

QUE PARA OBTENER EL GRADO DE:  
INGENIERO ELÉCTRICO  
ELECTRÓNICO  
PRESENTA

283131

**JOSÉ RAMÓN GUZMÁN  
MOSQUEDA**

DIRECTOR DE TESIS: M.I. BENJAMÍN VALERA OROZCO

SINODALES

M.I. LUIS ARTURO HARO RUÍZ  
M.I. LARRY HIPÓLITO ESCOBAR SALGUERO  
ING. EDUARDO RAMÍREZ SÁNCHEZ  
ING. ALEJANDRO SOSA FUENTES



MEXICO, D.F. SEPTIEMBRE del 2000.



Universidad Nacional  
Autónoma de México



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

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

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

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

**AGRADECIMIENTOS**

*A mis padres y hermanos por todo el apoyo recibido.*

*A mis familiares y amigos.*

# Índice temático

<b>Introducción</b>	<b>1</b>
<b>1 Metrología dimensional</b>	<b>5</b>
1.1 Generalidades	6
1.1.1 Principios de transducción	8
1.2 Medición de desplazamiento	10
1.2.1 El transformador diferencial linealmente variable (LVDT)	14
1.3 Medición de ángulo	19
1.3.1 El transformador diferencial variable rotatorio (RVDT)	21
<b>2 Procesamiento digital de señales</b>	<b>23</b>
2.1 Señales y sistemas discretos	23
2.1.1 Señales	24
2.1.2 Sistemas	27
2.2 Muestreo y reconstrucción	30
2.2.1 Muestreo de señales	31
2.2.2 Teorema de muestreo	34
2.2.3 Cuantificación de señales	37
2.2.4 Codificación de señales	38
2.2.5 Reconstrucción de señales	38
2.3 Herramientas para el procesamiento digital de señales	38
2.3.1 Análisis de Fourier	39
2.3.2 Promedios en el tiempo	45

<b>3 Implementación para la medición de ángulo y desplazamiento</b>	<b>47</b>
3.1 Esquema utilizado	48
3.1.1 Cálculo de la relación salida-entrada	49
3.1.2 Cálculo del desplazamiento	52
3.1.3 Estimación de la fase	53
3.2 Programación del algoritmo de medición en lenguaje ensamblador	55
3.2.1 Programa principal	55
3.2.2 Interrupciones	59
3.3 Programación de la interface con el usuario (LVDT.EXE)	60
3.4 LVDT.EXE. Manual de operación	62
3.4.1 Menú Archivo	64
3.4.2 Menú Editar	64
3.4.3 Menú Opciones	65
3.4.4 Menú Ayuda	67
3.4.5 Barra de herramientas	68
3.4.6 Procedimiento típico para operar el instrumento	68
<b>4 Resultados y conclusiones</b>	<b>71</b>
4.1 Resultados	71
4.2 Calibración	73
4.2.1 Calibración del instrumento para un LVDT	75
4.2.2 Calibración del instrumento para un RVDT	81
4.3 Conclusiones y trabajo a futuro	86
<b>Anexo A Diagramas electrónicos</b>	<b>89</b>
<b>Anexo B Diagramas de clases</b>	<b>93</b>
<b>Anexo C Procesador digital de señales</b>	<b>101</b>
C.1 Arquitectura	103
C.2 Organización	107
C.2.1 Modos de direccionamiento	109

---

C.3 Tarjeta DSK	111
C.4 Algoritmos básicos de procesamiento digital de señales	115
C.4.1 Operaciones aritméticas en punto flotante	116
C.4.2 Transformada de Fourier (FFT) en punto flotante	123
<b>Bibliografía</b>	<b>127</b>



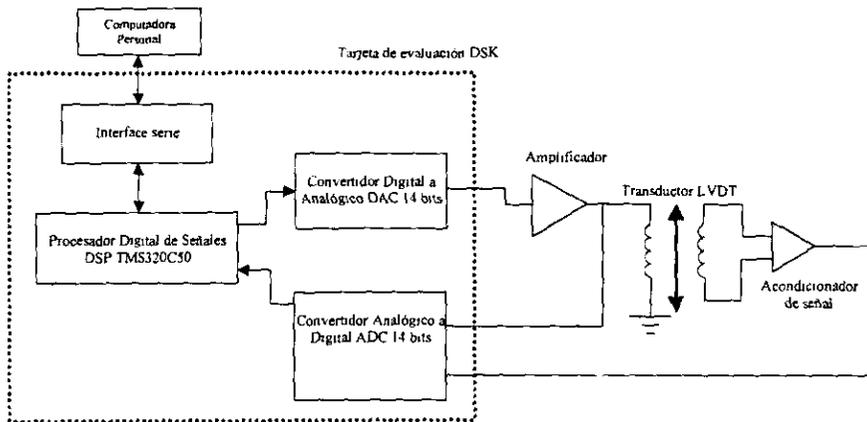
# INTRODUCCIÓN

En la actualidad en el Centro de Instrumentos UNAM, particularmente en el Laboratorio de Metrología, se desempeñan actividades de medición dimensional certificadas que requieren un alto grado de exactitud y resolución. Nuestro interés reciente, y debido al alcance actual de la electrónica digital, ha sido enfocado al desarrollo de instrumentos propios para la medición, que introduzcan ideas originales planteadas con base en la experiencia adquirida por los académicos y alumnos del laboratorio.

Uno de los principales problemas de los equipos de medición es su compleja electrónica que necesitan para adecuar las señales de los transductores y proporcionar mediciones confiables. Tal es el caso de la medición de desplazamiento y ángulo con alta resolución utilizando transductores del tipo L/RVDT (Transformadores Diferenciales Linealmente/Angularmente Variables) de efecto inductivo, en donde los sistemas electrónicos analógicos requieren de varios dispositivos electrónicos de ajuste, estabilidad, y calibración. Aunque existen distintos transductores para la medición de desplazamiento y ángulo como los de alta estabilidad basados en la detección de transiciones luminosas (codificadores ópticos) con cada vez más altas resoluciones, y que tienden a desplazar a los tradicionales transductores inductivos, los transductores inductivos siguen teniendo una amplia presencia en el campo de la metrología dimensional debido principalmente a su confiabilidad de operación bajo ambientes industriales de desgaste y suciedad.

En nuestro concepto, los altos grados de exactitud y resolución de los transductores L/RVDT, sugieren el empleo de técnicas recientes basadas en procesamiento digital de señales, para que la medición de ángulo y desplazamiento resulte una alternativa comparable en cuanto estabilidad a los dispositivos ópticos. El sistema propuesto en este trabajo de tesis se muestra en la figura 1.

El sistema fue diseñado para reducir al mínimo el número de componentes electrónicos externos a una tarjeta DSK (DSP Started Kit, DSP Herramienta de Desarrollo) y así hacer todo el procesamiento de las señales y la calibración por medio de software. Esto tiene la ventaja de calibrar el equipo y hacer mejoras o modificaciones fácilmente. Dependiendo de las modificaciones, éstas se pueden realizar sin tener que cambiar los componentes electrónicos o la arquitectura de la tarjeta DSK.



**Figura 1.** Sistema para la medición de desplazamiento y ángulo.

Como puede observarse en la figura 1, el sistema está basado en la tarjeta DSK; los únicos componentes externos a la DSK son un amplificador a la salida del convertidor digital-analógico, una red de acoplamiento para la entrada del LVDT, y un amplificador diferencial a la salida del LVDT para conectarlo a la entrada del convertidor analógico-digital. Estos amplificadores son construidos con amplificadores operacionales simples y de bajo costo. También se cuenta con una interfaz serie para conectar la tarjeta a la PC y mostrar la lectura. Esta interfaz es complementada con un programa hecho en Visual C++ para proporcionar un entorno amigable y fácil de utilizar por cualquier usuario incluso con poco conocimiento en metrología dimensional. Esta interfaz también fue creada para poder interactuar con otros programas de Windows como Excel y así poder transferir directamente las lecturas obtenidas por la tarjeta. Esta interfaz también cuenta con todas las opciones más comunes que se tienen en los equipos de medición similares para facilitar los procesos de medición.

El esquema de la figura ofrece las siguientes ventajas bien conocidas de los sistemas de instrumentación digitales sobre los tradicionales analógicos:

- Intercomunicación con equipos de cómputo y programas de terceros.
- Independencia a la variación en los valores nominales de los componentes electrónicos y voltajes de alimentación.
- Ajustes y calibración por programación.
- Posibilidades de auto-calibración y auto-diagnóstico.

El principio de operación mostrado en la figura 1 es como sigue:

Para registrar el desplazamiento o ángulo, el transductor se alimenta con una señal senoidal de amplitud y frecuencia fijas y conocidas, esta señal es sintetizada por el DSP. La señal en el secundario es digitalizada y devuelta al

DSP, esta señal es proporcional al desplazamiento del palpador. La señal del primario también es digitalizada y devuelta al DSP para considerar cambios de amplitud en ésta, debidos a las variaciones de impedancia. El DSP procesa las señales y proporciona la relación entre la amplitud del secundario y la del primario. Por último, esta lectura es ajustada por el software de la computadora dependiendo de la curva de calibración del transductor.

El propósito del proyecto planteado no es competir con instrumentos similares. En lugar de ello, buscamos desarrollar y experimentar con técnicas de procesamiento digital de señales que superen el desempeño de los equipos tradicionalmente conocidos. La justificación que planteamos no radica básicamente en la electrónica empleada, sino en la programación del algoritmo de medición, que es el área en donde buscamos innovar. En este sentido, usamos una plataforma hardware (DSP TMS320C50) como herramienta de desarrollo para la implementación de algoritmos de medición originales. Finalmente, el instrumento desarrollado contará con la certificación y trazabilidad a patrones internacionales, ya que nuestro Laboratorio de Metrología cuenta con la infraestructura y acreditamientos oficiales ante SECOFI. Este instrumento tendrá un alcance de  $\pm 1000s$  ó  $\pm 250\mu m$ , con una resolución de 0.1s ó 0.1 $\mu m$ .

El trabajo está dividido en 4 capítulos y 3 anexos, para su mejor desarrollo y entendimiento, el contenido es el siguiente:

El capítulo 1 es una introducción a la metrología dimensional y a las técnicas utilizadas para la medición de desplazamiento y ángulo. Se describen los distintos tipos de transductores utilizados para medir magnitudes físicas. También se describirán las características principales de los transductores L/RVDT así como su modelado. Este capítulo es importante porque describe las características de las señales utilizadas por el transductor, ya que esta es la base para la implementación del algoritmo de medición en la DSK.

El capítulo 2 es un breve resumen acerca de los sistemas y señales, así como de las herramientas del procesamiento digital de señales y sus características. Esto servirá para poder entender el procesamiento que efectúa el DSP sobre las señales para obtener la medición deseada. En este capítulo se describe la transformada rápida de Fourier y algunas de sus propiedades.

En el capítulo 3 se describe la implementación del algoritmo de medición tanto en el DSP como en la computadora, mostrando el diagrama de flujo de cada programa y las opciones de la interfaz del usuario así como su uso. También se describe la forma utilizada por la tarjeta DSK para sintetizar en todo tiempo la señal senoidal, así como el manejo de las interrupciones de transmisión y recepción seriales hacia el convertidor. En este capítulo se muestra la forma en la que se implementa la comunicación de la PC con el DSP.

Por último, en el capítulo 4 se muestran los resultados obtenidos y las curvas de calibración del equipo. También se muestra el desempeño del sistema comparado con los equipos electrónicos utilizados actualmente, se presentan las conclusiones así como sugerencias para mejorar el desempeño del equipo o para modificaciones futuras.

El anexo A muestra los diagramas electrónicos de los circuitos empleados en el proyecto. Estos circuitos son utilizados para acoplar y acondicionar las señales de entrada y salida que maneja el transductor L/RVDT.

El anexo B muestra los diagramas de clases de la aplicación para Windows *LVDT.EXE*. Estos diagramas sirven para comprender, a nivel de programador, cómo se programó la aplicación y en qué parte del programa se encuentran los aspectos importantes del código fuente.

El anexo C es una descripción del DSP, cómo es su arquitectura, programación, y cuáles son sus características más importantes que lo hacen un dispositivo ideal para el desarrollo del proyecto. También se describe la tarjeta que se está utilizando (DSK), analizando las características de sus periféricos especialmente el convertidor analógico a digital y digital a analógico. En este capítulo también se describe la forma en la que se programaron los algoritmos básicos utilizados para el procesamiento de las señales, principalmente la transformada rápida de Fourier y las operaciones aritméticas básicas a 64 bits en punto flotante.

El código fuente de los programas, tanto el del DSP como el de la PC, no se incluye en este texto. Esto se debe a que los capítulos desarrollados están enfocados a que el lector comprenda la manera en la que se diseñó e implementó el instrumento y no a la forma de programar el DSP o la PC.

Cabe hacer notar que el proyecto fue diseñado para ser utilizado como producto terminado. Entonces, el propósito es que sea utilizado en el Centro de Instrumentos de la UNAM, principalmente en el Laboratorio de Metrología, para servicios y proyectos que requieran mediciones de desplazamiento y ángulo certificadas. Así mismo, se espera que este proyecto sea la base para el futuro desarrollo de otros equipos basados en arquitecturas con DSP's.

# CAPÍTULO 1

## Metrología dimensional

En este capítulo se da una breve descripción de la metrología dimensional, prestando mayor interés por los métodos para medir desplazamiento y ángulo. El capítulo se enfoca principalmente a dar una descripción de los transductores del tipo LRVDF así como de sus características reales e ideales, también se presta interés por las señales que manejan estos transductores.

Antes de comenzar con el campo de estudio de la metrología dimensional, se definirán algunos términos muy utilizados en el área. Estos términos están relacionados con los parámetros que debe incluir cualquier equipo de medición.

*Sensibilidad:* Es la razón del movimiento del indicador de un instrumento al cambio en la variable medida que causa este movimiento, en otras palabras, es la relación de la señal de salida o respuesta del instrumento con respecto al cambio de entrada o variable medida.

*Exactitud:* Es la desviación de la lectura respecto a una entrada conocida, a esta desviación se le conoce como *error*. Generalmente la exactitud se expresa como un porcentaje de la lectura de la escala completa.

*Precisión:* Es la capacidad de un instrumento para reproducir cierta lectura con una exactitud dada, es decir, es la medida de la repetibilidad de un instrumento.

*Resolución:* Es el cambio mínimo del valor medido al cual el instrumento responde.

Como ya se mencionó, la desviación de la lectura al valor conocido se llama error, pero en muchas ocasiones experimentales no se tiene un valor conocido para compararlo con las lecturas del instrumento; sin embargo, se puede confiar en que el instrumento esté dentro de un margen mayor o menor al del valor real. En este caso se dice que el margen expresa la *incertidumbre* de las lecturas del instrumento [1].

*Calibración:* Es la comparación de un instrumento contra un patrón o estándar conocido, con esto se logran reducir los errores de exactitud. Los procedimientos de calibración implican una comparación del instrumento con: 1) un patrón primario; 2) un patrón secundario con una mayor exactitud; o 3) con una fuente de entrada conocida [1].

## 1.1 Generalidades

Las mediciones juegan un papel importante en la validación de las leyes de la ciencia. También son esenciales para estudiar, desarrollar y vigilar muchos dispositivos y procesos. Sin embargo, el proceso mismo de medición implica muchos pasos antes de producir un conjunto útil de información. Para estudiar los métodos que produzcan mediciones efectivas, se considerará el proceso de medición como una secuencia de cinco operaciones. Estas operaciones se pueden citar como siguen:

El diseño de un dispositivo eficiente de medición. Esto comprende una selección adecuada del equipo disponible y una interconexión correcta de los diferentes componentes e instrumentos.

El manejo inteligente del aparato de medición.

El registro de los datos de un modo claro y completo. La información registrada debe ser una referencia inequívoca para interpretaciones futuras.

El cálculo de la exactitud de la medición y las magnitudes de posibles errores implícitos.

La preparación de un informe que describa la medición y sus resultados para aquellos que puedan interesarse en su empleo.

Como puede observarse, el punto 1 requiere de un dispositivo de medición, pero existen distintos instrumentos de medición para una misma variable física. Esto es porque generalmente el diseño de un instrumento de medición se basa en algún principio de transducción y existen distintos transductores para una misma variable.

Un transductor es un dispositivo que proporciona una salida utilizable en respuesta a una magnitud física, propiedad o condición específica que se desea medir. en el caso de la metrología dimensional las variables físicas son el desplazamiento y ángulo. Cada transductor tiene ventajas y desventajas sobre los demás. en la tabla 1.1 se muestran distintos tipos de transductores. así como sus aplicaciones.

La descripción de un transductor se basa generalmente en las consideraciones siguientes:

¿Qué se desea medir? (magnitud).

¿Cuál es el principio operativo de la porción eléctrica del transductor en la que se origina la salida? (elemento de transducción, principio de transducción).

¿Qué elemento del transductor responde directamente a la magnitud que se mide?

¿Qué ventajas tiene el transductor?

¿Cuál es el intervalo de operación del transductor?

Sensores	Magnitudes			
	Posición Distancia Desplazamiento	Velocidad	Aceleración Vibración	Temperatura
Resistivos	Potenciómetros Galgas Magnetoresistencias		Galgas + masa-resorte	RTD Termistores
Capacitivos	Condensador diferencial			
Inductivos y Electromagnéticos	LVDT Corrientes Foucault Efecto Hall	Ley Faraday LVDT Efecto Hall	LVDT + masa-resorte	
Generadores			Piezoelectrónicos + masa-resorte	Termopares Piezoelectrónicos
Digitales	Codificadores incrementales y absolutos	Codificadores incrementales		Osciladores de cuarzo
Uniones p-n	Fotoelectrónicos			Diodo Transistor Convertidores P-I
Ultrasonidos	Reflexión	Efecto Doppler		

Tabla 1.1. Transductores.

Sensores	Magnitudes			
	Presión	Caudal Flujo	Nivel	Fuerza
Resistivos	Potenciómetros + tubo Bourdón	Anemómetros de hilo caliente Galgas + voladizo Termistores	Potenciómetro + flotador Termistores LDR	Galgas
Capacitivos	Condensador variable + diafragma		Condensador variable	Galgas capacitivas
Inductivos y Electromagné- ticos	LVDT + diafragma Reluctancia variable + diafragma	LVDT + rotámetro Ley Faraday	LVDT + flotador Corrientes Foucault	Magneto elástico LVDT + célula carga
Generadores	Piezoeléctricos			Piezoeléctricos
Digitales	Codificador + tubo Bourdón	Vórtices		
Uniones p-n			Fotoeléctricos	
Ultrasonidos		Efecto Doppler Tiempo tránsito Vórtices	Reflexión Absorción	

**Tabla 1.1. Transductores (Continuación).**

Como puede observarse, el diseño de un buen instrumento de medición se basa en la correcta elección del transductor, generalmente esto se hace teniendo en cuenta las condiciones del medio donde se va a operar y el uso que se le va a dar. Para esto es necesario conocer el principio de operación de los transductores. La siguiente sección está enfocada a dar una breve descripción de los principios de transducción de algunas magnitudes físicas.

### 1.1.1 Principios de transducción

El objeto de esta sección es dar a conocer de manera general los principios de transducción de los transductores típicos más comúnmente utilizados.

Los transductores capacitivos convierten un cambio de la magnitud a medir en un cambio de capacitancia. Este cambio varía dependiendo de la disposición del arreglo. Dado que un condensador consiste básicamente de dos placas conductoras paralelas separadas por un dieléctrico, el cambio de

capacitancia puede ocasionarse por el movimiento de una de estas placas o por el movimiento del material dieléctrico.

Los transductores inductivos convierten un cambio de la magnitud a medir en un cambio de la autoinductancia de un devanado. Los cambios de inductancia pueden efectuarse mediante el movimiento de un núcleo ferromagnético interior al devanado.

Los transductores reluctivos convierten un cambio de la magnitud en un cambio de tensión c.a. debido al cambio de la reluctancia del núcleo magnético entre dos o más devanados con una excitación de c.a. aplicada al sistema, en esta categoría se encuentra el I.VDT. El cambio de reluctancia se realiza mediante el movimiento de un núcleo magnético interior a los devanados.

Los transductores electromagnéticos convierten un cambio de la magnitud a medir en una fuerza electromotriz inducida en un conductor debido al cambio de un flujo magnético.

Los transductores piezoeléctricos convierten un cambio de magnitud a medir en un cambio de tensión, esto se debe a las propiedades de ciertos materiales que generan tensión o carga electrostática cuando se encuentran sometidos a un esfuerzo mecánico.

Los transductores resistivos convierten un cambio de magnitud a medir en un cambio de resistencia eléctrica. Los materiales pueden ser tanto conductores como semiconductores y el cambio se puede deber a distintos factores como el calentamiento o enfriamiento, aplicación de esfuerzos mecánicos, por la humedad o por el movimiento de las escobillas de un reostato.

Los transductores potenciométricos convierten un cambio de la magnitud a medir en un cambio de la relación de tensiones entre sus terminales debido al movimiento de una escobilla sobre un elemento resistivo.

Las galgas extensiométricas convierten un cambio en la magnitud a medir en un cambio de resistencia que se debe a la deformación mecánica. Es un transductor resistivo especial, se compone de dos o cuatro transductores de esfuerzo resistivo conectados a un circuito puente al que se le aplica una tensión de excitación.

Los fotoconductores convierten un cambio en la magnitud a medir en un cambio de conductancia de un material semiconductor, esto puede ser ocasionado por un incremento en la cantidad de luz que incide sobre el material.

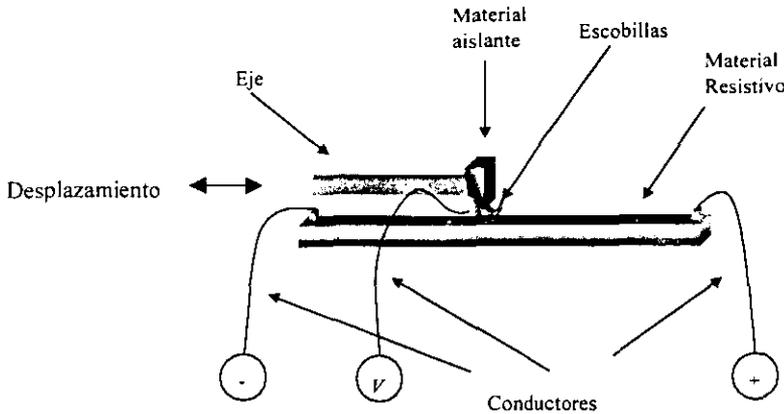
Los transductores fotovoltaicos convierten un cambio de la magnitud a medir en un cambio de tensión generada cuando la iluminación que incide sobre el material cambia.

Los transductores termoeléctricos convierten un cambio de la magnitud en un cambio de fuerza electromotriz, esto puede ser generado por la diferencia de temperaturas en la unión de dos materiales conductores distintos [2].

## 1.2 Medición de desplazamiento

Las dimensiones dimensionales son mediciones del tamaño de un objeto, en tanto que una medida de desplazamiento implica la medición del movimiento de un punto de una posición a otra. Estos tipos de mediciones son el campo de estudio de la metrología dimensional.

Existen distintos transductores para la medición del desplazamiento. Por ejemplo, los transductores de desplazamiento potenciométrico, en los cuales un contacto deslizante (escobilla) se mueve sobre un material resistivo y está aislado eléctricamente del eje sensor. El transductor se muestra en la figura 1.1.

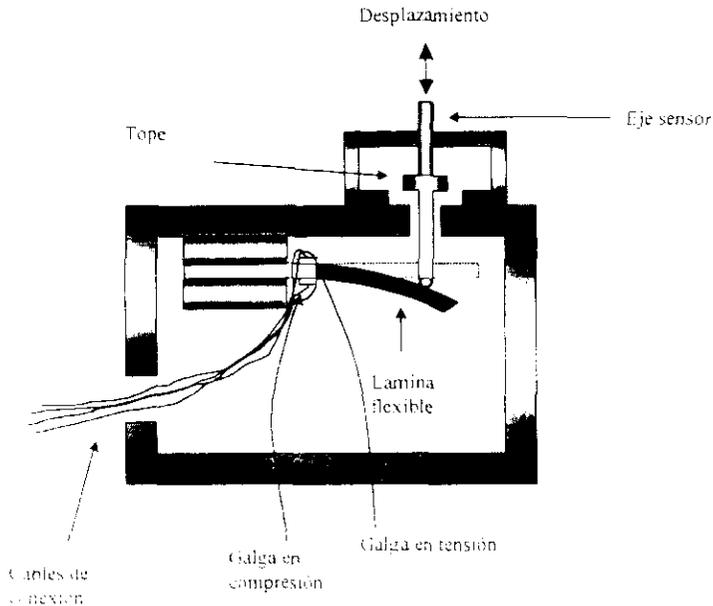


**Figura 1.1.** Transductor de desplazamiento potenciométrico.

Este tipo de transductores se alimentan con una diferencia de potencial entre sus bornes (indicados por - y + en la figura 1.1) y la salida  $V$  es proporcional al desplazamiento. Los elementos potenciométricos de resolución continua están hechos de plástico conductor, película de carbón, película metálica o mezclas metal-cerámica. Existen varias formas de las

escobillas que son generalmente de cobre. Una desventaja de estos transductores es que el contacto entre las escobillas y el material resistivo es afectado por factores externos como la contaminación, por lo tanto estos transductores deben quedar bien sellados en la salida del eje y los contactos eléctricos [2].

El desplazamiento también puede ser medido con galgas extensiométricas, las cuales consisten en flexores en los que están unidas las galgas extensiométricas, una en tensión y otra en compresión como se muestra en la figura 1.2.



**Figura 1.2.** Transductor con galgas extensiométricas.

Las galgas configuran semipuentes de dos elementos activos. Sin embargo, las galgas extensiométricas son raramente utilizadas en este tipo de sensores de desplazamiento, son mucho más aplicadas como elementos transductores accesorios de otros transductores como las células de carga, acelerómetros, sondas de peso, control de tensión, perfiles, etc. [2].

Existen otros métodos para medir dimensiones o desplazamiento en los cuales no se necesita un contacto físico con el objeto a medir, estos métodos ocupan transductores ópticos. Cuando la superficie de un objeto es suficientemente reflectora puede usarse una fuente y un sensor de luz para medir desplazamientos pequeños como se muestra en la figura 1.3.

La salida del sensor decrece exponencialmente con la distancia al objeto. Algunos diseños avanzados utilizan fibras ópticas para emitir y recibir la luz reflejada por la superficie, con esto se logra medir pequeños desplazamientos con elevadas resoluciones.

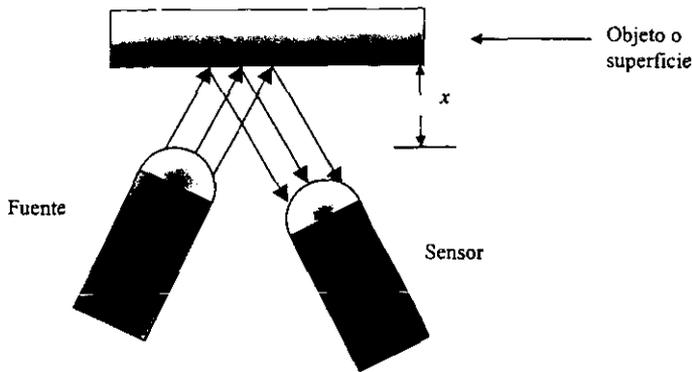


Figura 1.3. Principio para medir desplazamientos por métodos ópticos.

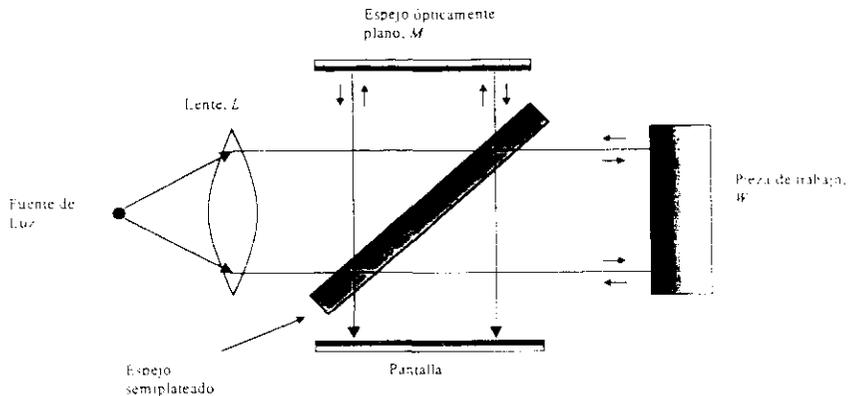
Un método óptico para medir dimensiones con mucha exactitud se basa en el principio de la interferencia de la luz. Al instrumento basado en este principio se le llama interferómetro. La interferencia de la luz se da cuando dos haces inciden en un mismo punto, la amplitud de la onda en el punto depende de la fase con la que los dos haces lleguen. Si los dos haces llegan con la misma fase la amplitud aumenta, en cambio, si los haces llegan con una diferencia de fases de media longitud de onda la amplitud se hace cero.

El principio de interferencia ofrece la posibilidad práctica de medir pequeños defectos de superficie y calibrar los bloques calibradores. Para estos propósitos se emplea un interferómetro como se muestra en la figura 1.4. La luz se colima mediante un lente  $L$ , el espejo semiplataado refleja la mitad de la luz hacia el espejo  $M$  ópticamente plano y permite la transmisión de la otra mitad de la luz hacia la superficie de trabajo  $W$ . Ambos haces se reflejan y se recombinan en el espejo semiplataado y se transmiten a la pantalla.

Si el instrumento está bien construido, se presentan franjas de interferencia en la pantalla debidas a las variaciones dimensionales de la pieza de trabajo. El interferómetro se usa principalmente para la calibración de bloques calibradores y otras aplicaciones donde se requieren mediciones dimensionales absolutas con extrema precisión [1].

Otro método óptico para hacer mediciones dimensionales o de desplazamiento se basa en los codificadores lineales. Estos codificadores entregan una salida digital sin la necesidad de utilizar convertidores

analógico a digital. Se utilizan tres métodos de transducción: 1) El codificador tipo escobilla que consta de una barra móvil con segmentos conductores o aislantes en su superficie; los segmentos conductores están unidos a un mismo punto eléctrico. 2) El codificador óptico o fotoeléctrico que consta de una barra transparente y que tiene segmentos opacos en su superficie y 3) El codificador magnético que consta de una barra móvil y que su superficie cuenta con segmentos magnetizados.

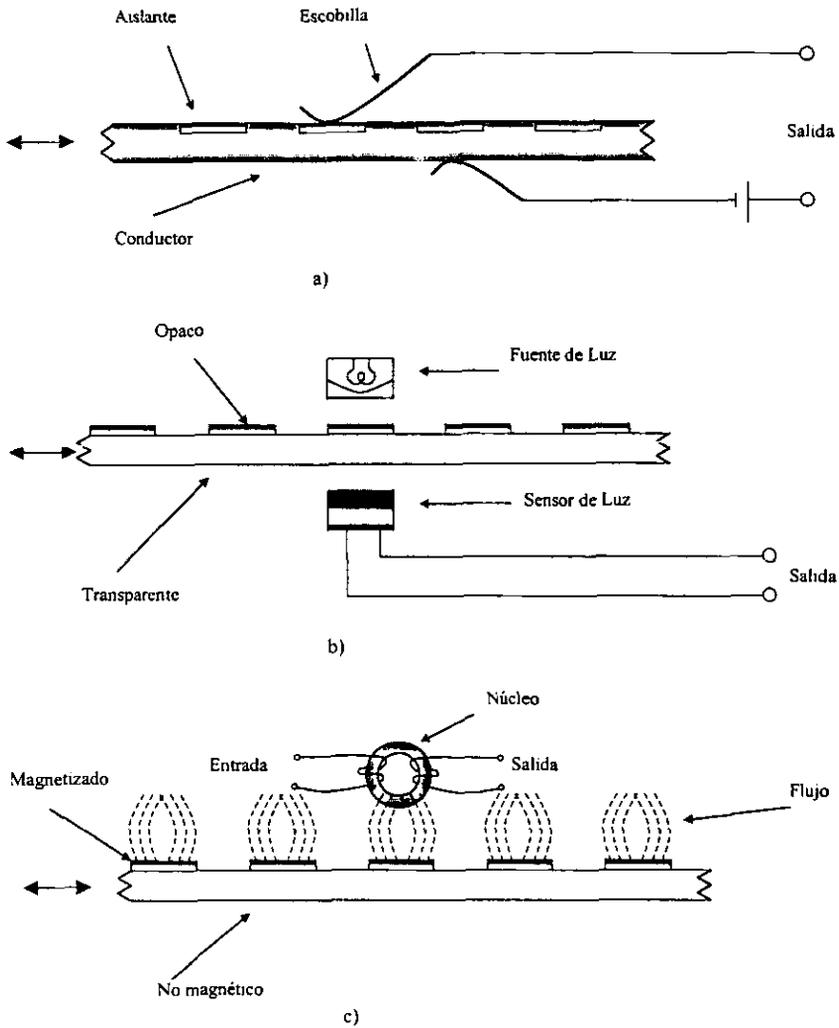


**Figura 1.4.** Método de interferómetro para medir desplazamientos

La función de estos tres tipos de transductores es la misma: proporcionar una salida de 1 o 0 dependiendo la posición de la barra como se muestra en la figura 1.5. En el codificador de escobillas cuando el contacto toca un segmento conductor se genera una salida de 1, de lo contrario se genera un 0. En el codificador óptico el 1 lógico se genera cuando la luz pasa de un extremo al otro y en el codificador magnético el 1 se genera cuando pasa un segmento no magnetizado por el sensor ya que el núcleo no se satura y la señal pasa de un extremo a otro.

Existen transductores de desplazamiento (codificadores) de dos tipos: 1) Los codificadores incrementales que producen pulsos igualmente espaciados en cada sector, estos pulsos son contados e indican el desplazamiento. 2) Los codificadores absolutos que producen un número codificado digitalmente, este número indica la posición pero utiliza varias cabezas lectoras y los segmentos están en una disposición multicanal. En este tipo de codificadores se pueden utilizar varias técnicas de codificación como BCD, Gray, etc. [2].

Por último, para la medición dimensional y de desplazamiento, se menciona el transductor de desplazamiento reluctivo, ya que dentro de estos se encuentra el transductor de nuestro mayor interés, el LRVDT.

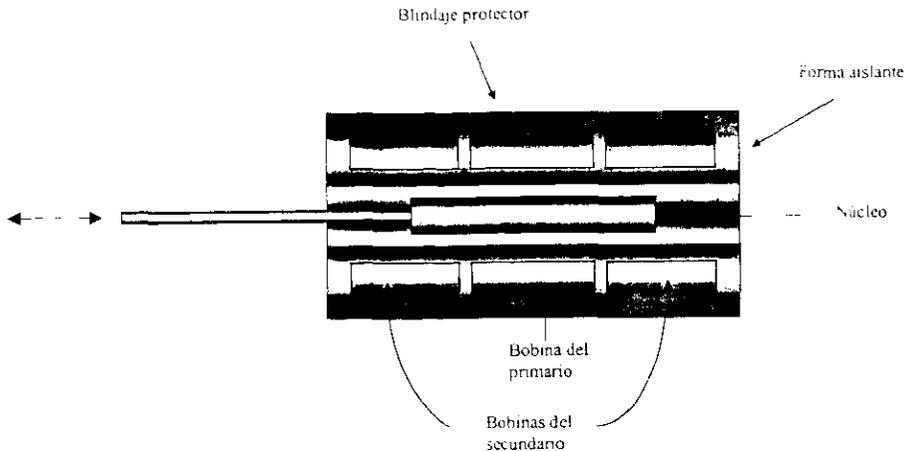


**Figura 1.5.** Tipos de transducción codificada: (a) tipo escobilla, (b) tipo fotoeléctrico y (c) tipo magnético.

### 1.2.1 El transformador diferencial linealmente variable (LVDT)

El LVDT es un transductor que básicamente consiste de un arreglo lineal de tres bobinas, una para el primario y dos para el secundario. Tiene

un núcleo magnético que se mueve con libertad dentro del arreglo como se indica en la figura 1.6.



**Figura 1.6.** Construcción de un LVDT.

El principio de operación se basa en la variación de la inductancia mutua entre un primario y cada uno de los dos secundarios al desplazarse a lo largo de su interior un núcleo de material ferromagnético, arrastrado por un vástago no ferromagnético, unido a la pieza cuyo movimiento se desea medir. El embobinado central (primario) se alimenta con una tensión alterna, en la posición central las tensiones inducidas en cada secundario son iguales y al apartarse de dicha posición el núcleo, una de las dos tensiones crece y la otra se reduce en la misma magnitud. Normalmente los dos devanados se conectan como se indica en la figura 1.7.

La salida de las dos bobinas de los extremos (bobinas del secundario) depende del acoplamiento magnético entre el núcleo y las bobinas. Este acoplamiento depende a su vez, de la posición del núcleo. En tanto el núcleo permanezca cerca del centro la salida es bastante lineal. Hay un corrimiento de fase de  $180^\circ$  de un lado de la posición nula al otro.

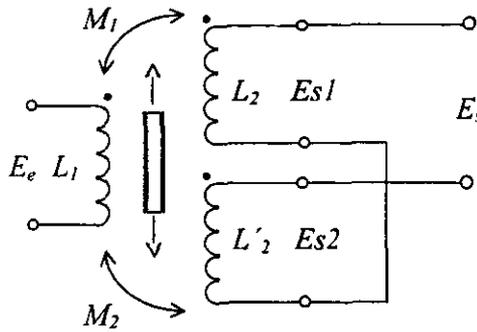


Figura 1.7. Conexión de los embobinados de un LVDT

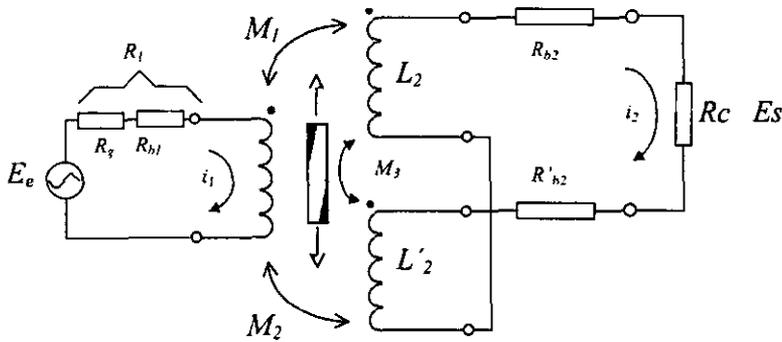


Figura 1.8. Modelo de un LVDT

El modelo matemático correspondiente se deduce del análisis de la figura 1.8. Si la resistencia total en el secundario se designa como  $R_2$ , se tiene entonces:

$$R_2 = R_{b2} + R'_{b2} + R_c \quad (1.1)$$

y en el primario

$$E_e = i_1(R_1 + sL_1) + i_2(-M_1s + M_2s) \quad (1.2)$$

En el secundario se tiene:

$$0 = i_1(-sM_1 + sM_2) + i_2(R_2 + sL_2 + sL'_2 - sM_3) \quad (1.3)$$

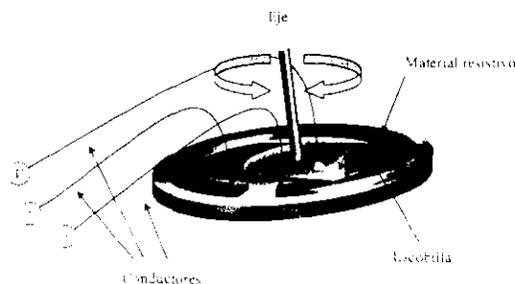
Con estas dos ecuaciones se tiene:

$$i_2 = \frac{-s(M_2 - M_1)E_e}{s^2[L_1(L_2 + L'_2 - 2M_3) - (M_2 - M_1)^2] + s[R_2L_1 + R_1(L_2 + L'_2 - 2M_3)] + R_1R_2} \quad (1.4)$$

aumenta proporcionalmente cuando el palpador se aleja de esta posición. Cabe hacer notar que la salida del secundario se defasa  $180^\circ$  cuando pasa de un lado de la posición nula al otro, con esto se puede detectar en qué lado se encuentra el núcleo.

### 1.3 Medición de ángulo

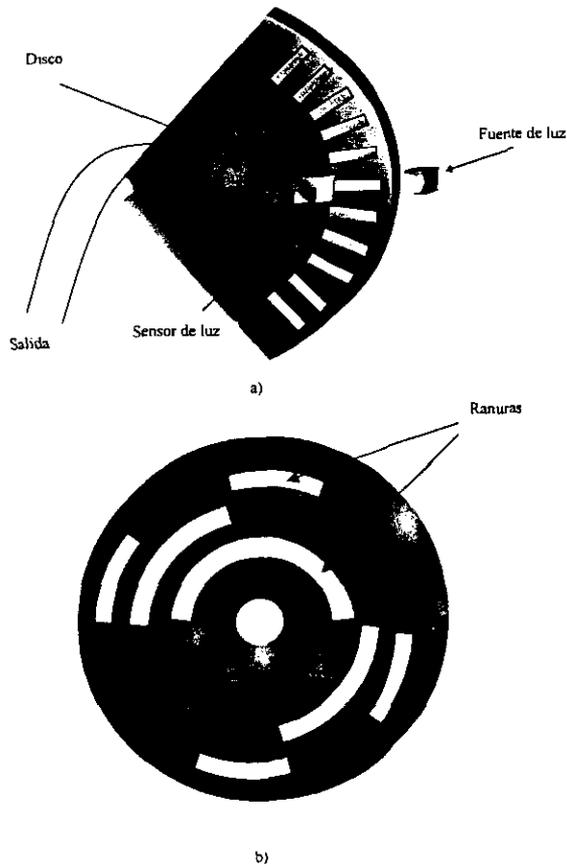
Los métodos de medición de ángulo están basados en los métodos de medición de desplazamiento haciendo cambios en la configuración de los transductores, por ejemplo, se puede medir ángulo con los transductores potenciométricos como se muestra en la figura 1.10



**Figura 1.10.** Transductor potenciométrico para la medición de ángulo.

El funcionamiento es el mismo que el del transductor potenciométrico de desplazamiento, la escobilla se mueve sobre el elemento resistivo. Se tiene un voltaje a la salida que es proporcional a la posición angular del eje. Los rangos de desplazamiento angular están comprendidos entre  $10^\circ$  y  $350^\circ$  pero en algunos transductores de intervalo pequeño se utilizan amplificadores mecánicos internos. También se pueden utilizar resistencias helicoidales en potenciómetros multivuelta cuyo rango excede los  $360^\circ$ .

Dentro de los métodos más utilizados para medir ángulo se encuentran los codificadores ópticos. Se pueden utilizar los mismos tres métodos de transducción para los codificadores angulares que los utilizados en los codificadores lineales, la diferencia es que los segmentos se encuentran sujetos en un disco en lugar de estar situados en una barra como se muestra en la figura 1.11.



**Figura 1.11.** Codificador óptico: a) Incremental y b) Absoluto.

Existen codificadores angulares incrementales y absolutos, su funcionamiento es idéntico al de los codificadores lineales, en los incrementales, al girar el disco, un pulso es acumulado en un contador, en número de pulsos es una lectura del desplazamiento angular. En los codificadores absolutos se tienen varios lectores y las ranuras están situadas en configuración multicanal, como se muestra en la figura 1.11b, para cada posición del disco existe un código que indica su posición absoluta. El codificador de la figura 1.11b es de tres bits, entre más canales tenga con ranuras, el codificador tendrá más bits y será por lo tanto de mayor resolución.

Otro método para medir ángulo, se basa en el principio del transformador diferencial lineal variable, al transductor se le conoce como transformador diferencial variable rotatorio (RVDT).

### 1.3.1 El transformador diferencial variable rotatorio (RVDT)

El RVDT es un transductor que tiene el mismo principio de funcionamiento del LVDT, produce una señal eléctrica que es linealmente proporcional al desplazamiento angular. Los desplazamientos que puede detectar este transformador son bastante pequeños en comparación con los que puede detectar un nivel de burbuja o un nivel mecánico.

El RVDT detecta ángulos mediante el desplazamiento del núcleo ferromagnético dentro de un transformador especial como lo muestra la figura 1.12. El transformador tiene un embobinado primario y dos secundarios. Los tres embobinados se devanan en el mismo tubo aislante. El embobinado primario se devana en el centro del tubo, y los dos devanados secundarios (que tienen el mismo número de vueltas) se conectan en serie y en oposición. Esto significa que el acoplamiento mutuo entre cada devanado secundario y el primario es igual, el voltaje  $E_v$  entre los conductores del devanado secundario será cero, aún cuando el primario esté excitado por una señal de c.a.

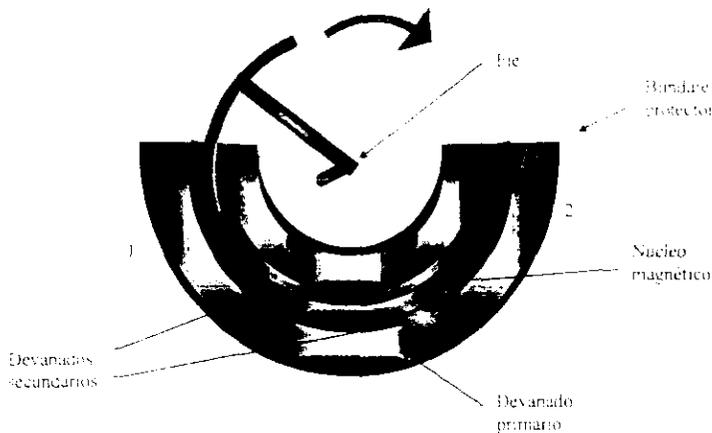


Figura 1.12. Configuración de los devanados de un RVDT.

Si se centra el núcleo ferromagnético (con respecto a la longitud del transformador), el acoplamiento mutuo entre cada devanado secundario y el primario será igual. Siempre que exista ésta condición,  $E_v = 0$ . Sin embargo si el núcleo se mueve con respecto a su posición central, el acoplamiento

mutuo entre cada devanado secundario y el primario ya no será igual. Por ejemplo, si se mueve el núcleo como se indica en la figura 1.12, el acoplamiento mutuo entre el devanado secundario 1 y el primario aumentará, mientras que el acoplamiento mutuo entre el devanado secundario 2 y el primario disminuirá. Un corrimiento de la posición del núcleo hacia el otro sentido tendrá el efecto contrario. Como resultado de los cambios en el acoplamiento mutuo, el voltaje  $E$ , entre los conductores de salida conectados a los devanados secundarios ya no será cero. En lugar de ello (para desplazamientos angulares pequeños del núcleo), este voltaje de salida será linealmente proporcional a la magnitud del desplazamiento angular.

Como puede observarse, el funcionamiento es el mismo que el del LVDT, lo único que cambia es la disposición del arreglo físico para permitir la medición de desplazamientos angulares. El modelado del RVDT es el mismo que el modelado del LVDT, por lo tanto, las señales que se manejan tienen las mismas características así como los transductores en sí tienen las mismas ventajas y desventajas.

## CAPÍTULO 2

# Procesamiento digital de señales

El procesamiento digital de señales se ha desarrollado rápidamente en los últimos años. Este desarrollo ha sido resultado de los grandes avances en la tecnología digital de las computadoras y de las técnicas de fabricación de circuitos integrados que permiten grandes escalas de integración. Con el desarrollo de circuitos integrados digitales cada vez más rápidos, es posible implementar sistemas que desempeñen procesamiento de señales tal como filtrado, amplificación, modulación, demodulación, codificación, decodificación, etc., sin la necesidad de utilizar componentes electrónicos externos o que son difíciles de implementar con circuitos electrónicos analógicos. El presente capítulo analiza los sistemas discretos así como algunas técnicas del procesamiento digital de señales y algunas de sus propiedades que son de interés para implementar el algoritmo de medición en el DSP.

### 2.1 Señales y sistemas discretos

El estudio de los sistemas y señales juega un papel importante en la ingeniería. A continuación se dará una definición de las señales y los sistemas así como su clasificación y se analizará a qué tipo pertenecen las señales utilizadas en el transductor y en el procesador digital de señales.

### 2.1.1 Señales

Las señales describen una gran variedad de fenómenos físicos y aunque pueden representarse de muchas formas, la información está contenida en un patrón de variaciones que representa alguna forma determinada. Estas variaciones de cualquier señal pueden ser con respecto al tiempo, al espacio o a cualquier otra variable independiente. Matemáticamente se puede describir una señal como una función de una o más variables independientes [4]. La tabla 2.1 muestra la clasificación de las señales [5].

Señales	{	a) <i>Determinísticas y aleatorias</i>
		b) <i>Periódicas y no periódicas</i>
		c) <i>Par e impar</i>
		d) <i>De energía y de potencia</i>
		e) <i>Continuas en el tiempo y discretas en el tiempo</i>
		f) <i>Analógicas y digitales</i>

Tabla 2.1. Clasificación de las señales.

- a) Una señal es determinística cuando puede modelarse con una función explícita del tiempo de tal manera que se puede conocer su valor en cualquier instante de tiempo. Una señal es aleatoria cuando no se puede modelar mediante una función explícita del tiempo, para describirla se requiere agregar un modelo de tipo probabilístico. Lo único que se puede determinar es cuál es la probabilidad de que un valor aparezca en un instante dado.
- b) Una señal periódica es aquella que satisface la siguiente relación:

$$x(t) = x(t + T_0)$$

Donde a  $T_0$  se le denomina periodo fundamental.

Una señal no periódica es aquella que no satisface la relación indicada anteriormente. Las señales periódicas y no periódicas o aperiódicas tienen a su vez las siguientes subclasificaciones.

Señales periódicas	{	<i>Sinusoidales (seno, coseno, etc.)</i>
		<i>No sinusoidales (cuadrada, triangular, etc.)</i>

$$\text{Señales no periódicas} \begin{cases} \text{Estrictamente limitadas en el tiempo (Pulso rectangular, etc.)} \\ \text{Asintóticamente limitadas en el tiempo (Exponencial unilateral, etc.)} \\ \text{No limitadas en el tiempo (Escalón, etc.)} \\ \text{Cuasiperiódicas} \end{cases}$$

Una señal es par si es idéntica a su contraparte invertida en el tiempo, es decir, con su reflejo respecto al origen. Una señal par  $x(t)$  cumple con la siguiente relación:

$$x(-t) = x(t)$$

y una señal impar cumple con la relación:

$$x(-t) = -x(t)$$

Una señal impar, como puede observarse en la relación anterior, debe ser necesariamente 0 en  $t=0$ .

- c) La *Energía Total* de una señal  $x(t)$  se define como [5]:

$$E = \lim_{T \rightarrow \infty} \int_{-T}^T x(t)^2 dt$$

y la *Potencia Media Total* de una señal  $x(t)$  se define como [5]:

$$P = \lim_{T \rightarrow \infty} \frac{1}{T} \int_{-T}^T x(t)^2 dt$$

Una señal es de energía si y solo si su energía tiene un valor finito diferente de cero. Una señal es de potencia si y solo si su potencia media total tiene un valor finito diferente de cero.

Una señal de energía no puede ser una señal de potencia y viceversa (son mutuamente exclusivas). Las señales aperiódicas estricta y asintóticamente limitadas en el tiempo son señales de energía. Las señales periódicas y las aperiódicas no limitadas en el tiempo y cuasiperiódicas, así como las señales aleatorias son señales de potencia.

- d) Las señales continuas en el tiempo son aquellas que tienen un valor definido para cualquier instante de tiempo. Las señales discretas en el tiempo son señales que toman valores para ciertos instantes de tiempo y para los valores intermedios la señal no está definida.
- e) Una señal analógica es aquella que es continua en el tiempo y en amplitud, a diferencia de una señal digital que es discreta en tiempo y en amplitud.

Las señales de nuestro interés son las señales digitales, la clasificación anterior aunque fue ejemplificada para señales analógicas

también es aplicada a las señales digitales, por ejemplo, una señal digital  $x(n)$  periódica cumple con la siguiente relación:

$$x(n) = x(n + T_0)$$

Donde  $T_0$  es el periodo de la función.

Una señal analógica puede convertirse en una señal digital y viceversa utilizando un convertidor analógico a digital o digital a analógico, este tema se explicará en la siguiente sección. Las señales de nuestro interés son señales determinísticas, periódicas (sinusoidales), de potencia, discretas y digitales.

Nótese que una señal discreta o digital  $x(n)$  no está definida para instantes entre dos muestras sucesivas, también es incorrecto pensar que la señal es cero si  $n$  no es un entero. Existen varias representaciones para las señales discretas [6]:

1. Representación funcional, como:

$$x(n) = \begin{cases} 2, & \text{para } n = 1, 2, 3, 5 \\ 1, & \text{para } n = 4, 6, 7 \\ 0, & \text{en cualquier otro caso} \end{cases}$$

2. Representación tabular, como:

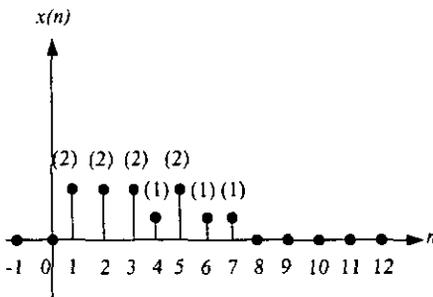
$$\begin{array}{cccccccccccc} n & \dots & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & \dots \\ x(n) & \dots & 0 & 2 & 2 & 2 & 1 & 2 & 1 & 1 & 0 & \dots \end{array}$$

3. Representación en secuencia, como:

$$x(n) = \{ \dots 0, 0, 2, 2, 2, 1, 2, 1, 1, 0, \dots \}$$

↑

4. Representación gráfica, como:



Dentro de las señales discretas, determinísticas, periódicas y de potencia se encuentran unas muy importantes: las señales sinusoidales. Una señal sinusoidal en tiempo discreto puede representarse como lo indica la siguiente expresión:

$$x(n) = A \cos(\omega n + \theta), \quad -\infty < n < \infty \quad (2.1)$$

Donde  $n$  es una variable entera denominada número de muestra,  $A$  es la amplitud de la señal,  $\omega$  es la frecuencia angular y  $\theta$  es la fase. La relación (2.1) también puede expresarse en términos de la frecuencia discreta:

$$x(n) = A \cos(2\pi f_d n + \theta), \quad -\infty < n < \infty \quad (2.2)$$

La frecuencia  $f_d$  (discreta) tiene dimensiones de ciclos por muestra. Las señales sinusoidales en tiempo discreto cumplen con las siguientes propiedades.

1. Una señal sinusoidal en tiempo discreto es periódica sólo si su frecuencia  $f_d$  es un número racional.
2. Las señales sinusoidales en tiempo discreto cuyas frecuencias están separadas por un múltiplo entero de  $2\pi$  son idénticas.

$$\cos[(\omega + 2\pi)n + \theta] = \cos(\omega n + 2\pi n + \theta) = \cos(\omega n + \theta)$$

Analizando esta situación se puede ver que las secuencias de dos señales sinusoidales cualesquiera con frecuencias dentro del intervalo  $-\pi < \omega < \pi$  son distintas, cualquier señal sinusoidal con una frecuencia  $\omega > \pi$  es idéntica a una secuencia obtenida a partir de una señal sinusoidal de frecuencia  $\omega - \pi$ . Debido a esta similitud se denomina a la señal sinusoidal de mayor frecuencia como un *alias* de la señal sinusoidal de frecuencia  $\omega - \pi$ .

También se puede expresar el intervalo de frecuencias dentro de las cuales las señales sinusoidales discretas son distintas por la siguiente relación:

$$-\pi < \omega < \pi \quad (2.3)$$

3. La mayor tasa de oscilación de una señal sinusoidal discreta se tiene cuando  $\omega = \pi$  o  $\omega = -\pi$ .

## 2.1.2 Sistemas

Un sistema puede ser descrito como una configuración de componentes o elementos que desempeñan una función. Un sistema tiene al

menos una señal de entrada y al menos una señal de salida [7]. La tabla 2.2 muestra la clasificación de los sistemas.

Sistemas	{	a) Causales y no causales
		b) Estáticos y dinámicos
		c) Estocásticos y determinísticos
		d) Parámetros concentrados y parámetros distribuidos
		e) Lineales y no lineales
		f) Continuos y discretos
		g) Invariantes y variantes en el tiempo

**Tabla 2.2.** Clasificación de los sistemas.

- a) Un sistema es causal si su salida en cualquier instante de tiempo sólo depende de los valores de la entrada presente y de los valores pasados. Dicho sistema también se le conoce como *no anticipativo* ya que la salida del sistema no se anticipa a valores futuros de la entrada.
- b) Se dice que un sistema es *sin memoria*, *amnésico*, *instantáneo* o de *memoria cero* si su salida para cada valor del tiempo depende solamente de la entrada en ese instante. A estos sistemas también se les conoce como sistemas estáticos. El concepto de memoria de un sistema corresponde a la presencia de un mecanismo o elemento que almacena información sobre los valores de entrada en instantes distintos de tiempo. A los sistemas con la capacidad de almacenar datos en memoria se les llaman sistemas dinámicos.
- c) Un sistema es determinístico cuando a mismas condiciones de excitación se obtiene una misma respuesta.
- d) Un sistema de parámetros concentrados es aquel en el cual las dimensiones físicas de sus elementos son despreciables en comparación a la longitud de onda de las señales que maneja el sistema o de la frecuencia normal de operación.
- e) Se dice que un sistema es lineal, ya sea en el tiempo continuo o en tiempo discreto, si cumple con la importante propiedad de *superposición*: si una entrada consiste en la suma ponderada de varias señales, entonces la salida es simplemente la superposición, es decir, la suma ponderada de las respuestas del sistema a cada señal de entrada. Si un sistema tiene una respuesta  $y_1$  a una entrada  $x_1$  y una respuesta  $y_2$  a una entrada  $x_2$  entonces es lineal si:

$$\alpha x_1 + \beta x_2 \rightarrow \alpha y_1 + \beta y_2$$

A la propiedad de sumar las dos señales de salida cuando la entrada es la suma de las dos señales  $x_1$  y  $x_2$  se le conoce como *aditividad*, mientras que a la propiedad de multiplicar la misma constante de la señal de entrada a la señal de salida se le conoce como *homogeneidad*.

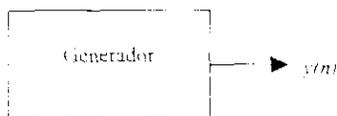
- f) Un sistema continuo es aquel en el cual la señal de entrada y salida son continuas en el tiempo, a diferencia de un sistema discreto en el cual sus señales son digitales. En las aplicaciones de tiempo discreto hay que diseñar dispositivos que realicen operaciones sobre las señales discretas, a estos dispositivos se les llama *sistemas en tiempo discreto* o *sistemas discretos* [6]. En otras palabras, un sistema discreto es un conjunto de elementos que opera sobre una señal de entrada en tiempo discreto para generar otra señal denominada salida o respuesta del sistema. Los sistemas continuos y discretos pueden ser conectados con convertidores analógico a digital (A/D) o digital a analógico (D/A).
- g) Un sistema es invariante en el tiempo si el comportamiento y características del mismo están fijos en el tiempo, en otras palabras, un sistema es invariante en el tiempo si un corrimiento de tiempo en la señal de entrada produce un corrimiento de tiempo en la señal de salida.

Los sistemas de nuestro interés son sistemas discretos, invariantes en el tiempo, de parámetros concentrados, dinámicos y causales. En nuestro caso se tiene un sistema de procesamiento que se puede definir de la siguiente manera:

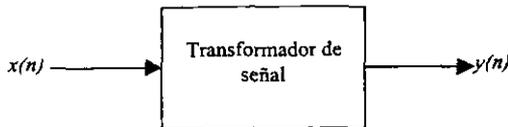
Los sistemas de procesamiento de señales son elementos que realizan algunas operaciones básicas tales como amplificación, filtrado, transformaciones no lineales, modulación, detección, etc., sobre señales generadas externa o internamente. El resultado es ya sea otra señal o una medición desplegada en forma analítica o digital.

Los procesadores de señales se pueden dividir en tres grupos:

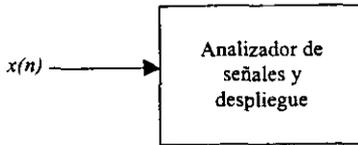
- Generadores de señales: sólo hay señal de salida.



- Transformadores de señales: hay señales de entrada y de salida.



- Analizadores de señales: sólo hay señal(es) de entrada, la salida es un despliegue de los resultados del análisis.



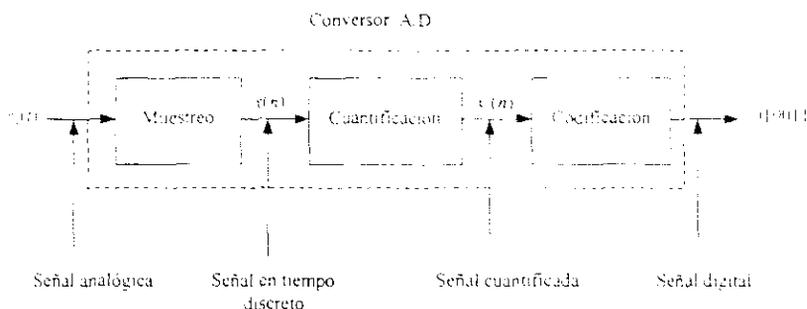
En nuestro caso se tiene que el sistema es de tipo analizador de señal ya que se encarga de aplicar algoritmos a las dos señales de entrada y la salida es un dato que es el resultado del análisis que se realizó.

Como ya se mencionó anteriormente, los sistemas analógicos y digitales se pueden conectar unos con otros utilizando convertidores analógico a digital o digital a analógico. Lo que hacen es transformar una señal de un tipo al otro. El siguiente tema es una explicación del proceso que se tiene que seguir para convertir una señal analógica a digital y viceversa.

## 2.2 Muestreo y reconstrucción

La mayoría de las señales de interés práctico (voz, sísmicas, sonoras, audio, video, etc.) son analógicas. Para procesar este tipo de señales utilizando sistemas digitales es necesario convertirlas en señales digitales. Este proceso es llamado *conversión analógica a digital* y se realiza en dispositivos llamados convertidores analógico a digital (ADC). En los sistemas digitales del tipo transformadores de señales también se requiere reconstruir la señal digitalizada ya procesada, esto se hace con dispositivos llamados convertidores digital a analógico (DAC).

La conversión analógica a digital se puede ver como una secuencia de tres pasos [6] como se muestra en la figura 2.1.



**Figura 2.1.** Partes de un convertidor analógico a digital.

1. **Muestreo.** Este paso es la conversión de una señal en tiempo continuo a una señal en tiempo discreto. Para hacer esto se obtienen "muestras" de la señal en tiempo continuo. Si  $x_a(t)$  es la entrada al muestreador, la salida será  $x_a(nT) = x(n)$ , en donde  $T$  es el intervalo de muestreo.
2. **Cuantificación.** En este paso se convierte la señal en tiempo discreto en una señal de tiempo discreto con valores discretos de amplitud. El valor de cada muestra se representa mediante un valor seleccionado de un conjunto de valores posibles. A la diferencia entre la muestra de la señal en tiempo discreto  $x(n)$  y la salida del cuantificador  $\hat{x}(n)$  se denomina *error de cuantificación*.
3. **Codificación.** Aquí cada valor cuantificado se representa por una secuencia binaria perteneciente a un código.

Uno de los aspectos más importantes en la conversión de una señal analógica a digital es, sin duda, el paso de muestrear la señal. Si la señal no se muestrea correctamente, la señal que se quiere digitalizar puede mostrar un perfil distinto a la señal original. A este error se le conoce como *aliasing*.

### 2.2.1 Muestreo de señales

Existen varias formas de muestrear una señal, sin embargo, el tipo de muestreo más utilizado es el *muestreo periódico o uniforme* que se describe por la relación:

$$x(n) = x_a(nT), \quad -\infty < n < \infty \quad (2.4)$$

Donde  $x(n)$  es la señal en tiempo discreto obtenida de las muestras de la señal analógica  $x_a(t)$  cada  $T$  segundos como se muestra en la figura 2.2.

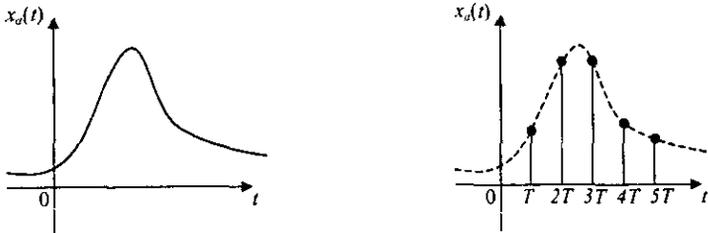
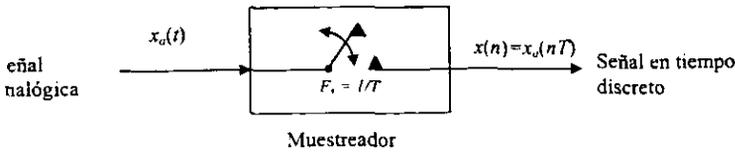


Figura 2.2. Muestreo de una señal analógica.

Al intervalo de tiempo  $T$  se le denomina *periodo de muestreo* y su recíproco  $1/T=f_s$ , se le denomina *frecuencia de muestreo*. Como puede observarse en la relación (2.4) el muestreo uniforme establece una relación entre las variables  $t$  y  $n$  del tiempo continuo y del tiempo discreto respectivamente.

$$t = nT = \frac{n}{f_s} \quad (2.5)$$

Por lo tanto existe una relación entre la frecuencia de las señales analógicas  $f$  y la frecuencia de las señales de tiempo discreto  $f_d$ , por ejemplo, considere la siguiente señal analógica:

$$x_a(t) = A \cos(2\pi f t + \theta) \quad (2.6)$$

Si se muestrea periódicamente a una frecuencia  $f_s$ , entonces de (2.5) y (2.6) se tiene:

$$x(n) = A \cos\left(\frac{2\pi n f}{f_s} + \theta\right) \quad (2.7)$$

comparando con (2.2) se tiene que:

$$f_d = \frac{f}{f_s} \quad (2.8)$$

La frecuencia discreta también se le conoce como frecuencia normalizada o relativa, esto puede observarse en la relación (2.8). Como puede observarse en esta relación, se puede obtener la frecuencia en Hertz si se conoce la frecuencia de muestreo de una señal digital.

Recordando el intervalo de frecuencias dentro del cual las señales sinusoidales discretas son distintas y sustituyendo la ecuación (2.8) en (2.3), tenemos que:

$$\frac{f_s}{2} < f < \frac{f_s}{2} \quad (2.9)$$

Como sabemos la frecuencia máxima de una señal en tiempo discreto es  $f_d = 1/2$ , el valor máximo de  $f$  para una velocidad de muestreo  $f_s$  es:

$$f_{max} = \frac{f_s}{2} = \frac{1}{2T} \quad (2.10)$$

El muestreo periódico produce una ambigüedad ya que la máxima frecuencia de una señal en tiempo continuo que puede determinarse unívocamente cuando dicha señal se muestrea a una frecuencia  $f_s$   $1/T$  es  $f_{max} = f_s/2$ .

Esto es muy importante por si se requiere saber el periodo de muestreo  $T$  para convertir una señal de tiempo continuo a tiempo discreto. Supongamos que una señal puede representarse como una suma de señales sinusoidales de distintas amplitudes, frecuencias y fases como lo indica la siguiente relación:

$$x_s(t) = \sum_{i=1}^N A_i \cos(2\pi f_i t + \theta_i) \quad (2.11)$$

Donde  $N$  indica el número de componentes de frecuencia.

Si suponemos la representación (2.11) para una señal continua, podemos obtener una  $f_{max}$  o podemos considerar una frecuencia máxima significativa de la señal. En la práctica a veces se realiza un filtrado de la señal para asegurar que no existan componentes de frecuencia por encima de  $f_{max}$ . El conocimiento de esta  $f_{max}$  permite seleccionar la frecuencia de muestreo apropiada ya que sabemos que la frecuencia más alta de una señal analógica que puede reconstruirse sin ambigüedad cuando se muestrea a una frecuencia  $f_s$  es  $f_s/2$ .

Para evitar ambigüedades que resultan del aliasing, se debe seleccionar una velocidad de muestreo lo suficientemente alta, es decir, se debe escoger  $f_s/2$  mayor que  $f_{max}$ , por lo tanto para evitar el problema del aliasing, se selecciona  $f_s$  según la siguiente relación:

$$f_s > 2f_{max} \quad (2.12)$$

Donde  $f_{max}$  es la frecuencia más alta de la señal analógica.

Esta relación también es el resultado del *teorema de muestreo*, el cual se enuncia más adelante.

El proceso de muestreo puede ser implementado de muchas maneras, la más popular es la operación de *muestreo y retención*. La salida de este proceso es llamada *modulación por amplitud de pulsos* (PAM) porque los intervalos sucesivos de la salida pueden ser descritos como una secuencia de pulsos con amplitudes entregadas de las muestras de la señal de entrada. La señal puede ser recuperada a partir de la señal PAM filtrándola con un filtro paso bajas. Una pregunta importante es ¿Cuánto puede un filtro aproximar una señal PAM a su señal original? La respuesta puede encontrarse revisando el teorema de muestreo [8].

## 2.2.2 Teorema de muestreo

Una señal limitada en banda que no tiene componentes espectrales arriba de una frecuencia  $f_{max}$  Hertz puede ser determinada únicamente por valores adquiridos a intervalos uniformes de  $T_s$  segundos, donde:

$$T_s \leq \frac{1}{2f_{max}} \quad (2.13)$$

A ésta declaración particular se le conoce también como el *teorema de muestreo uniforme* [8]. La tasa de muestreo  $f_s = 2f_{max}$  es llamada la *tasa de Nyquist*. [8] Si sustituimos en la relación (2.13)  $f_s = 1/T_s$ , se llega a una relación conocida como criterio de Nyquist:

$$f_s \geq 2f_{max} \quad (2.14)$$

El criterio de Nyquist es una condición teóricamente suficiente para permitir a una señal analógica ser reconstruida completamente de un conjunto de muestras en tiempo discreto uniformemente espaciadas. Primero examinemos el caso del muestreo ideal con una secuencia de funciones impulso. Asumiendo una señal  $x(t)$  como se muestra en la figura 2.3. con su transformada de Fourier  $X(f)$  que es cero fuera del intervalo  $(-f_{max} < f < f_{max})$ . El muestreo de  $x(t)$  puede ser visto como el producto de  $x(t)$  con un tren periódico de funciones impulso unitario  $x_\delta(t)$  definido como:

$$x_\delta(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \quad (2.15)$$

Donde  $T_s$  es el periodo de muestreo y  $\delta(t)$  es la función delta de Dirac. Escogiendo  $T_s = 1/2f_{max}$  para satisfacer el criterio de Nyquist, y recordando las propiedades de la función impulso se tiene:

$$\begin{aligned} x(t)\delta(t-t_0) &= x(t_0)\delta(t-t_0) \\ x_s(t) &= x(t)x_\delta(t) = \sum_{n=-\infty}^{\infty} x(nT_s)\delta(t-nT_s) \end{aligned} \quad (2.16)$$

Utilizando la propiedad de convolución en frecuencia de la transformada de Fourier en la que el producto de dos funciones en el dominio del tiempo es la convolución de las mismas en el dominio de la frecuencia y sabiendo que:

$$X_s(f) = \sum_{n=-\infty}^{\infty} \delta(f - nf_s) \tag{2.17}$$

Si consideramos otra propiedad de la función impulso en la que la convolución de una función con un impulso es la misma función corrida, es decir,  $X(f) * \delta(f - nf_s) = X(f - nf_s)$ , se tiene entonces:

$$X_s(f) = X(f) * X_s(f) = \frac{1}{T_s} \sum_{n=-\infty}^{\infty} X(f - nf_s) \tag{2.18}$$

Podemos ver que el espectro  $X_s(f)$  de la señal muestreada dentro del intervalo del ancho de banda original de la señal  $x(t)$  es exactamente el mismo que el de la señal original, esto se ilustra gráficamente en la figura 2.3.

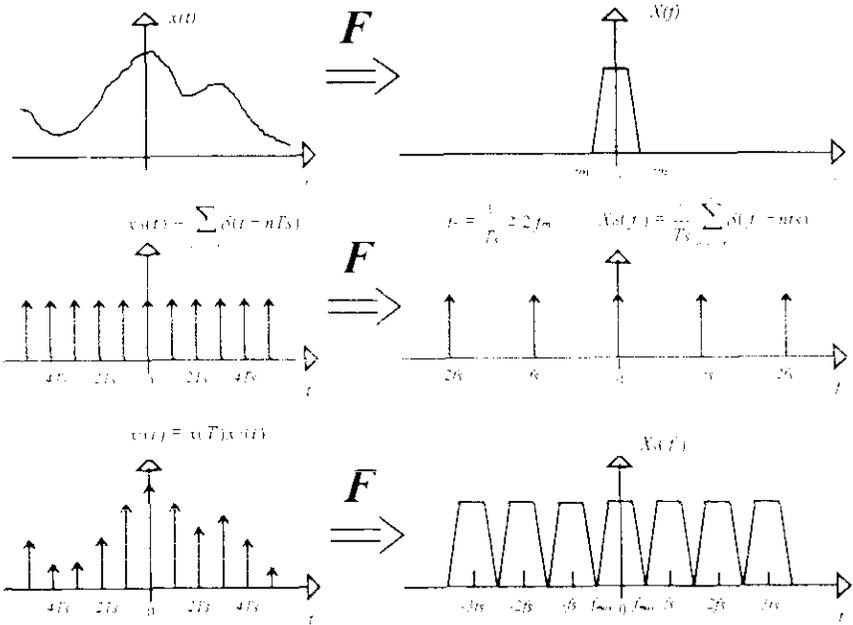


Figura 2.3. Muestreo ideal.

Aunque el muestreo instantáneo es un modelo conveniente existe una forma más práctica de realizar el muestreo de una señal analógica limitada en banda que es multiplicar  $x(t)$  por un tren de pulsos como se muestra en la

figura 2.4. Cada pulso es de ancho  $T$  y amplitud  $1/T$ . El muestreo es llamado natural ya que cada pulso en la secuencia  $x_s(t)$  retiene la forma de su correspondiente segmento analógico durante el intervalo del pulso.

Cuando no se cumple el criterio de Nyquist se tiene un error de muestreo denominado *aliasing*. La señal no puede reconstruirse debido a que existe una superposición de las componentes en frecuencia de la señal. Las figuras 2.5 y 2.6 muestran el aliasing de una señal en el dominio del tiempo y la frecuencia respectivamente.

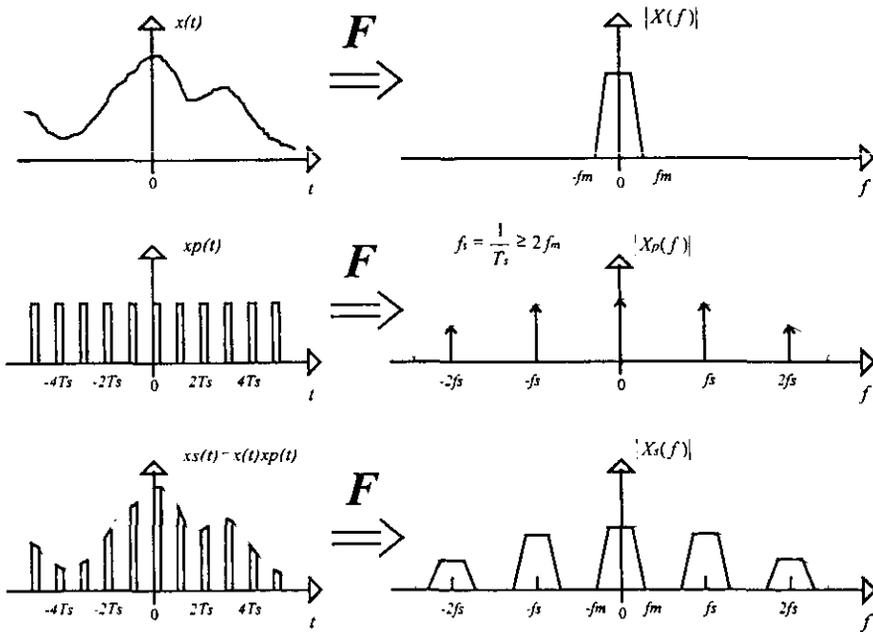


Figura 2.4. Muestreo natural.

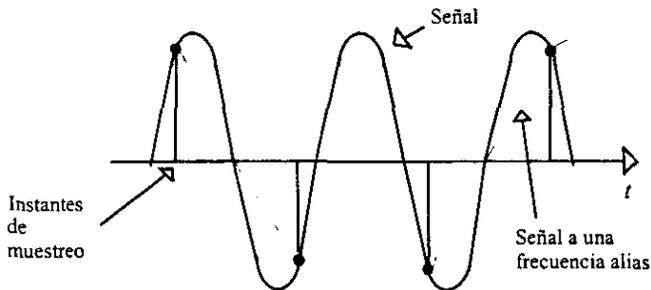


Figura 2.5. Aliasing en el dominio del tiempo.

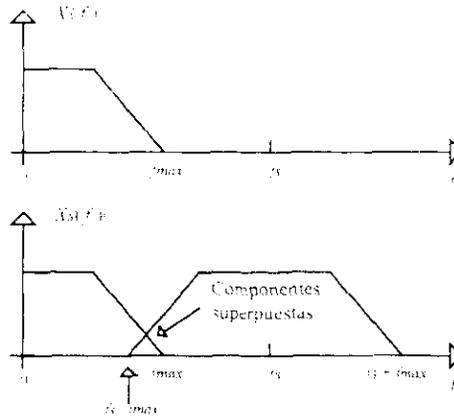


Figura 2.6. Aliasing en el dominio de la frecuencia.

### 2.2.3 Cuantificación de señales

La figura 2.7 muestra una señal sinusoidal analógica muestreada y cuantificada.

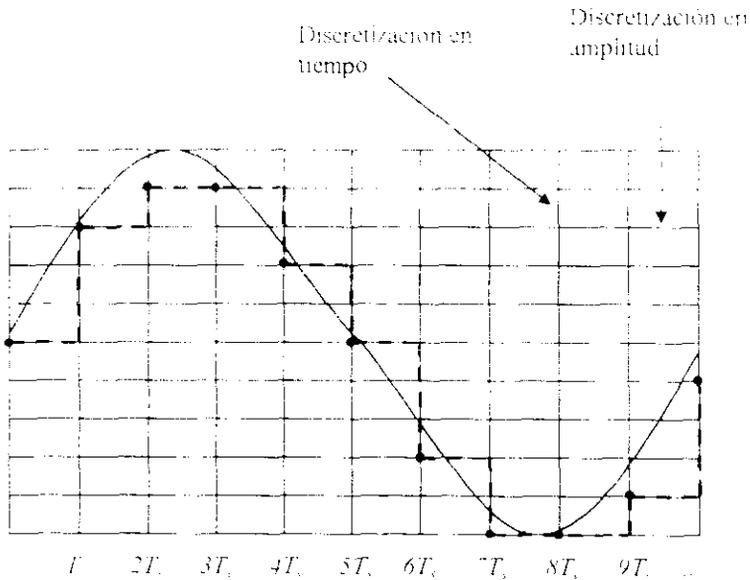


Figura 2.7. Cuantificación de una señal.

Las líneas horizontales indican los niveles de cuantificación, después de muestrear una señal y pasarla al tiempo discreto, se debe discretizar en amplitud, este paso se realiza en el cuantificador y se realiza utilizando un mantenedor de orden cero. Si la tasa de muestreo satisface el criterio de Nyquist, el único error que se tiene en la conversión A/D es en el cuantificador y a su vez éste depende del número de niveles de cuantificación que se tengan [6].

## 2.2.4 Codificación de señales

Después del proceso de cuantificación se le asigna un número binario a cada muestra dependiendo de su nivel de cuantificación. Existe un número binario único para cada nivel de cuantificación, así que si se disponen de  $L$  niveles se deben tener al menos  $b$  bits que cumplan con  $2^b \geq L$  o lo que es lo mismo  $b \geq \log_2 L$ . Por ejemplo para la figura 2.7 se necesitan 4 bits ya que se cuenta con 11 niveles de cuantificación. Los conversores comerciales generalmente son de 8 a 16 bits y su costo depende mucho de su velocidad.

## 2.2.5 Reconstrucción de señales

La reconstrucción de señales es el proceso en el que se convierte una señal digitalizada en una señal analógica. Para realizar esta tarea se utilizan dispositivos denominados convertidores digital a analógico (DAC). El convertidor digital a analógico más simple es el mantenedor de orden cero y que simplemente mantiene constante el valor de una muestra hasta recibir la siguiente. En la práctica se utiliza un filtro a la salida del convertidor A/D para suavizar la señal de salida, este filtro no deja pasar las componentes de alta frecuencia indeseables en la señal y que se deben al mantenedor de orden cero.

## 2.3 Herramientas para el procesamiento digital de señales

Existen distintas operaciones para el procesamiento digital de señales tales como la correlación, convolución, filtrado, etc., que nos permiten manejar la señal de entrada y obtener la señal de salida deseada. Sin embargo una de las herramientas más poderosas es la *Transformada de Fourier*.

### 2.3.1 Análisis de Fourier

La transformada de Fourier es una de las diferentes herramientas útiles en el análisis y diseño de sistemas, otra son las series de Fourier. Estas herramientas implican la descomposición de las señales en términos de componentes sinusoidales. Con esta descomposición se dice que una señal está representada en el *dominio de la frecuencia* [9].

La representación matemática básica de las señales periódicas es la serie de Fourier la cual es una suma de sinusoidales relacionadas armónicamente como indica la siguiente relación [9]:

$$x(t) = \frac{1}{2} a_0 + \sum_{n=1}^{\infty} [a_n \cos(n\omega_0 t) + b_n \sin(n\omega_0 t)] \quad (2.19)$$

Una serie como la representada por (2.19) se llama *serie trigonométrica de Fourier*. La representación en series de Fourier de una función periódica representa la función periódica como la suma de componentes sinusoidales que tienen diferentes frecuencias. La componente sinusoidal de frecuencia  $\omega_n = n\omega_0$  se denomina *enésima armónica* de la función periódica. La primera armónica generalmente se conoce como componente fundamental porque tiene el mismo periodo de la función.

Los coeficientes de la serie trigonométrica de Fourier se obtienen de las siguientes relaciones [9]:

$$a_0 = \frac{2}{T} \int_0^T x(t) dt \quad (2.20)$$

$$a_n = \frac{2}{T} \int_0^T x(t) \cos(n\omega_0 t) dt, \quad n = 0, 1, 2, \dots \quad (2.21)$$

$$b_n = \frac{2}{T} \int_0^T x(t) \sin(n\omega_0 t) dt, \quad n = 0, 1, 2, \dots \quad (2.22)$$

En muchas aplicaciones de las series de Fourier es conveniente expresar estas series en términos de los exponenciales complejos  $e^{j\omega t}$ . Si consideramos una serie como la descrita en (2.19) y sustituimos las relaciones entre el seno y coseno con las funciones exponenciales se puede llegar a una serie denominada *forma compleja de la serie de Fourier* o *serie compleja de Fourier* de  $x(t)$ .

$$x(t) = \sum_{n=-\infty}^{\infty} c_n e^{jn\omega_0 t} \quad (2.23)$$

donde

$$c_n = \frac{1}{2} (a_n - jb_n) \quad (2.24)$$

De esta misma serie se puede obtener información a partir de los coeficientes  $c_n$ , tomando en cuenta que son números complejos se tiene que:

$$c_n = c_n e^{j\phi_n} \quad (2.25)$$

donde

$$c_n = \frac{1}{2} \cdot a_n^2 + b_n^2 \quad (2.26)$$

y

$$\phi_n = \tan^{-1} \left( -\frac{b_n}{a_n} \right) \quad (2.27)$$

La gráfica de la magnitud de los coeficientes complejos  $c_n$  contra la frecuencia se denomina *espectro de amplitud* de la función y la gráfica del ángulo de fase  $\phi_n$  de  $c_n$  contra la frecuencia se denomina *espectro de fase* de la función.

La serie de Fourier nos da información acerca de la amplitud y fase de los armónicos que componen una función periódica, pero si se tienen funciones que no son periódicas se puede considerar que el periodo  $T$  tiende a infinito, así se convierte en una función no periódica.

El objetivo de este texto no es hacer demostraciones o desarrollar paso a paso la serie o transformada de Fourier sino más bien es mostrar al lector la información que proporciona y las fórmulas adecuadas que se tienen que utilizar para las señales que se manejan en el DSP. Para ver información más detallada acerca del análisis de Fourier se pueden consultar un sin número de referencias bibliográficas, entre ellas se puede consultar las referencias [6] y [9].

Cuando se hace que  $T \rightarrow \infty$  y considerando que  $\omega_0 = 2\pi/T$ , la serie de Fourier descrita por (2.19) se convierte en:

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) e^{j\omega t} d\omega \quad (2.28)$$

$$X(\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt \quad (2.29)$$

La expresión (2.28) se conoce como la *transformada de Fourier* y a la expresión (2.29) se le conoce como *transformada inversa de Fourier*. La función  $X(\omega)$  es en general compleja y se puede expresar como [6]:

$$X(\omega) = |X(\omega)| e^{j\phi(\omega)} \quad (2.30)$$

Donde  $|X(\omega)|$  se denomina espectro de magnitud de  $x(t)$  y  $\phi(\omega)$  se denomina espectro de fase de  $x(t)$ . Como vimos anteriormente la representación en series de Fourier de una señal periódica en tiempo continuo puede tener infinitas componentes en frecuencia debido a que el

intervalo de frecuencias en tiempo continuo es desde  $-\infty$  hasta  $\infty$ , pero el intervalo de frecuencias en el tiempo discreto se limita en el intervalo  $(-\pi, \pi)$  o  $(0, 2\pi)$ .

Una señal en tiempo discreto de período fundamental  $N$  puede tener componentes de frecuencia separadas  $2\pi/N$  radianes o  $f=1/N$  ciclos. De aquí que la representación en series de Fourier de señales periódicas en tiempo discreto contendrá como máximo  $N$  componentes de frecuencia siendo ésta la diferencia fundamental entre la representación en series de Fourier de una señal en tiempo continuo y una en tiempo discreto.

Si tenemos una secuencia periódica  $x(n)$  de período  $N$ , es decir,  $x(n) = x(n+N)$  para todo  $n$ , entonces la representación en series de Fourier de esta secuencia periódica consta de  $N$  funciones exponenciales armónicamente relacionadas:

$$x(n) = \sum_{k=0}^{N-1} c_k e^{j\omega_k n} \quad (2.31)$$

donde  $c_k$  son los coeficientes de la serie y se obtienen a partir de la siguiente relación [6]:

$$c_k = \frac{1}{N} \sum_{n=0}^{N-1} x(n) e^{-j\omega_k n} \quad (2.32)$$

La expresión (2.31) se denomina serie de Fourier en tiempo discreto. Los coeficientes de Fourier proporcionan la descripción de  $x(n)$  en el dominio de la frecuencia, en el sentido de que  $c_k$  representa la amplitud y la fase asociada a la componente de frecuencia.

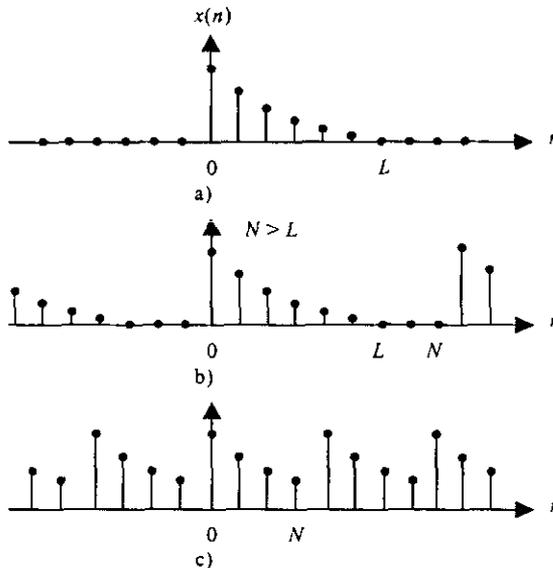
Igual que en el caso continuo, el análisis en frecuencia de las señales aperiódicas en tiempo discreto implica una transformada de Fourier de la señal en el dominio del tiempo. Esta transformada de Fourier de una señal de energía finita en tiempo discreto  $x(n)$  se define como:

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n) e^{-j\omega n} \quad (2.33)$$

$X(\omega)$  es una descomposición de  $x(n)$  en sus componentes en frecuencia. Note que la transformada de Fourier de una señal no periódica en tiempo discreto es una función continua de la frecuencia, esto nos lleva al desarrollo de otra herramienta para el análisis en frecuencia de señales en tiempo discreto, a esta otra representación se le conoce como *Transformada de Fourier Discreta (DFT)* y se muestra en la siguiente expresión [6]:

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-j\omega_k n} \quad k=0,1,2,\dots,N-1 \quad (2.34)$$

La transformada discreta de Fourier es una versión muestreada de la transformada de Fourier para señales en tiempo discreto, esto tiene la desventaja que la señal reconstruida a partir de las muestras de la DFT no es la señal original sino una señal que es periódica de periodo  $N$  y es una versión con aliasing de  $x(n)$ , pero si el número de muestras es mayor o igual al número de muestras de la señal  $x(n)$  lo que se tiene es la repetición de la señal  $x(n)$ . Esto se ilustra en la figura 2.8.



**Figura 2.8.** a) Señal original  $x(n)$  de  $L$  muestras. b) Reconstrucción de la señal a partir de su espectro discretizado con  $N$  muestras (sin aliasing). c) Reconstrucción de  $x(n)$  a partir de su espectro con  $N < L$  (con aliasing).

La transformada discreta de Fourier inversa se obtiene a partir de la siguiente expresión:

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j2\pi kn/N} \quad n = 0, 1, 2, \dots, N-1 \quad (2.35)$$

Existen distintos métodos para obtener la transformada discreta de Fourier a partir de una secuencia finita  $x(n)$ , uno es realizando la suma con la que se define la transformada, otro es de manera matricial, pero existe un método que permite obtener la transformada discreta de Fourier de una manera más rápida sin hacer demasiadas operaciones, este método es llamado *transformada rápida de Fourier (FFT)* [6].

Para desarrollar algoritmos computacionales eficientes para el cálculo de la DFT se utiliza una estrategia de dividir una DFT de  $N$  puntos en DFT's más pequeñas. Este algoritmo se conoce como FFT [6].

Otra forma de expresar la DFT (2.34) es:

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{nk} \quad k = 0, 1, 2, \dots, N-1 \quad (2.36)$$

donde

$$W = e^{-j2\pi/N} \quad (2.37)$$

Si calculamos una DFT cuando  $N=2$  tenemos:

$$X(k) = \sum_{n=0}^{N-1} x(n)W_2^{nk} = x(0)W_2^{0k} + x(1)W_2^{1k} \quad k = 0, 1. \quad (2.37)$$

En donde  $W_2 = e^{-j2\pi/2} = e^{-j\pi} = -1$ . Entonces para  $n=0$  y  $n=1$  se tiene que  $W_2^{0k} = (-1)^0 = 1$  y  $W_2^{1k} = (-1)^k$ , de aquí que:

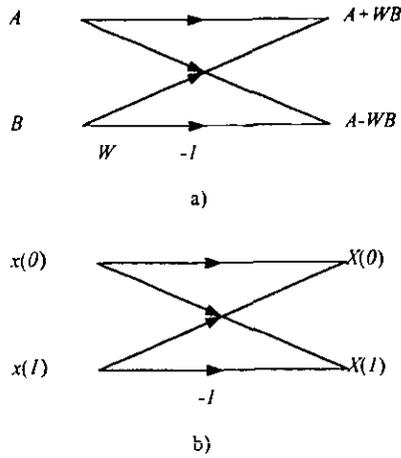
$$\begin{aligned} X(0) &= x(0) + x(1) \\ X(1) &= x(0) - x(1) \end{aligned} \quad (2.38)$$

Las operaciones de (2.38) pueden expresarse en la figura 2.9

A esta operación se le conoce como *mariposa*. Otra característica de la FFT es que la secuencia de entrada debe tener un orden especial a este ordenamiento se le llama *del mayor al menor tiempo* (DIT) por eso algunas veces se le conoce como FFT DIT. Si la secuencia de entrada no se decima, hay que ordenar la secuencia de salida.

Con esta idea se plantea el siguiente procedimiento para el cálculo de la FFT para  $N$  puntos (donde  $N$  debe ser potencia de 2).

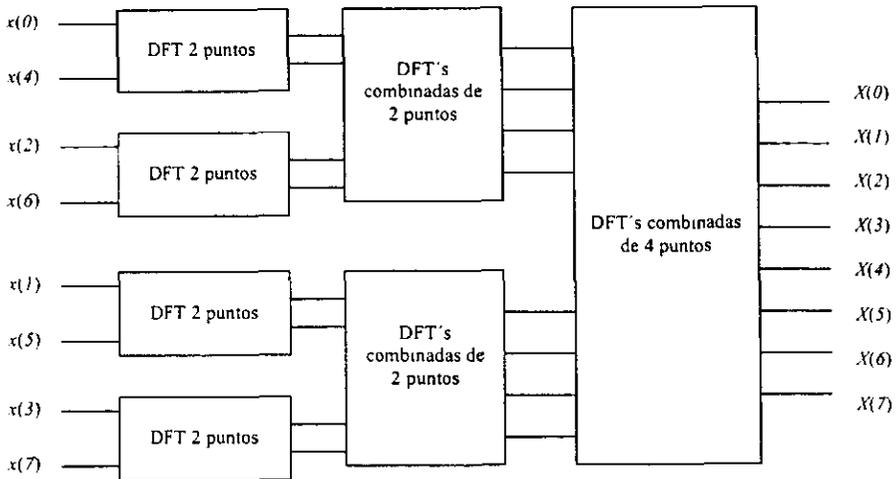
- Decimar la DFT de  $N$  puntos en dos DFT's de  $N/2$  puntos y determinar las ecuaciones.
- Decimar cada DFT de  $N/2$  puntos en dos DFT's de  $N/4$  puntos y determinar las ecuaciones correspondientes.



**Figura 2.9.** Operación mariposa. a) General y b) Caso para las ecuaciones (2.38).

- Continuar el proceso hasta que las DFT's resulten de 2 puntos. Aquí termina el proceso.

Este método se puede ilustrar en la figura 2.10 [6].



Decimación

**Figura 2.10.** Cálculo de la FFT para una secuencia  $x(n)$  de 8 puntos.

Cada uno de los bloques de la figura 2.16 representan la operación mariposa.

### 2.3.2 Promedios en el tiempo

Los promedios en el tiempo se determinan solo para señales de potencia, los promedios en el tiempo de las señales de energía valen cero.

El operador promedio en el tiempo se define como:

$$\langle x(t) \rangle = \begin{cases} \lim_{T \rightarrow \infty} \frac{1}{T} \int_{t_1}^{t_2} x(t) dt & \text{Para señales aperiódicas} \\ \frac{1}{T_0} \int_{t_0}^{t_0+T_0} x(t) dt & \text{Para señales periódicas} \end{cases} \quad (2.39)$$

La mejor aproximación que puede hacerse de una cantidad a partir de un conjunto de lecturas es la media aritmética de los datos y el valor más probable se podría obtener a partir de un conjunto finito de los mismos, es decir:

$$\langle x \rangle = \frac{1}{N} \sum_{i=1}^N x_i \quad (2.40)$$

Otro tipo de promedio en el tiempo es el valor cuadrático medio, se define como:

$$\langle x^2 \rangle = \begin{cases} \lim_{T \rightarrow \infty} \frac{1}{T} \int_{t_1}^{t_2} x^2(t) dt & \text{Señales aperiódicas} \\ \frac{1}{T_0} \int_{t_0}^{t_0+T_0} x^2(t) dt & \text{Señales periódicas} \end{cases} \quad (2.41)$$

El valor cuadrático medio es igual a la potencia media total de la señal.

El promedio en el tiempo utilizado en el algoritmo es la media aritmética ya que como se mencionó es el valor más probable que puede tomar la lectura a partir de un conjunto de lecturas calculado.

## CAPÍTULO 3

# Implementación para la medición de ángulo y desplazamiento

Tomando en cuenta el estudio anterior, el problema ahora es encontrar la forma de medir desplazamiento y ángulo utilizando el transductor L RVDT y la tarjeta DSK. Como ya se mencionó, el esquema está basado en la tarjeta DSK. Entonces, se necesita desarrollar una interfaz electrónica entre esta tarjeta y el transductor, así como una interfaz entre la tarjeta y la PC. La idea es que la tarjeta procese las señales que maneja el transductor y envíe una lectura, la cual será enviada a la computadora para ser mostrada al usuario. Así las señales del transductor serán manejadas por la tarjeta, y las opciones de medición en el instrumento serán manejadas por la computadora mediante una interfaz de usuario. El presente capítulo describe la manera de obtener las lecturas con la tarjeta y la forma en la que se implementaron los programas tanto en el DSP como en la computadora.

El código fuente, tanto en el DSP como en la PC, no se incluye en este texto, pero está disponible para su revisión en archivos de computadora. La idea es que el lector entienda la forma en que se implementó el algoritmo de medición, en lugar de decifrar las extensas páginas de código ensamblador del DSP y Visual C++ de la PC.

### 3.1 Esquema utilizado

Recordando que el L/RVDT maneja señales sinusoidales y que el desplazamiento del vástago es proporcional a la relación de transformación, se propone el siguiente sistema mostrado en la figura 3.1 para la medición de desplazamiento y ángulo.

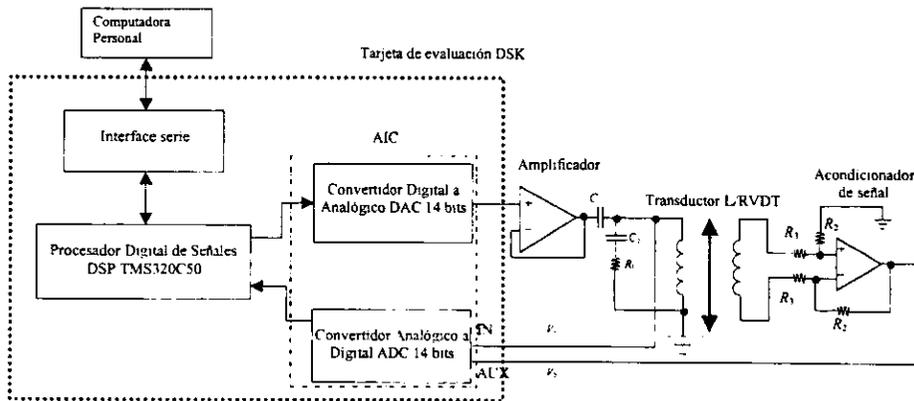


Figura 3.1. Sistema para la medición de desplazamiento y ángulo.

El amplificador a la salida del DAC está en configuración de seguidor y la red formada por  $C_1$ ,  $C_2$  y  $R_1$  sirve para acoplar la salida del seguidor a la entrada de L/RVDT (para obtener más información acerca del transductor L/RVDT, ver los temas 1.2.1 y 1.3.1). Como disponemos de 2 entradas analógicas en el AIC (ver anexo C), (IN y AUX) la entrada del L/RVDT también es una de las entradas del AIC, en este caso se conecta a la entrada IN ( $V_1$ ). El acondicionador de señal es un amplificador operacional en configuración de amplificador diferencial, esto sirve para eliminar el ruido de modo común a la salida del transductor y también le da una ganancia a la señal del secundario. La salida de este amplificador alimenta a la entrada AUX del AIC ( $V_2$ ).

Los elementos externos a la tarjeta, como son la red de acoplamiento y el acondicionador de señal a la entrada y salida del transductor respectivamente, están en una placa de circuito impreso y son conectados a la DSK a través de los conectores de expansión.

El valor del resistor  $R_2$  es de  $100\text{k}\Omega$  y el valor del resistor  $R_3$  es de  $1.5\text{k}\Omega$ . Con esto se obtiene una ganancia en el amplificador diferencial de

66.66. Esta ganancia fue calculada previamente midiendo experimentalmente la amplitud máxima de la salida del secundario.

La red de acoplamiento fue diseñada basada en una red de acoplamiento típica de un amplificador de audio; los valores tomados son los mismos que se utilizan para acoplar una carga de  $4\Omega$ , excepto el valor del capacitor 1:  $C_1=10\mu\text{F}$ ,  $C_2=10\text{nF}$  y  $R=1\Omega$ . La red es en esencia un filtro paso banda. El valor de  $C_1$  para acoplar una carga de  $4\Omega$  es de  $1000\mu\text{F}$  y si es menor, la frecuencia de corte inferior se incrementa. En nuestro caso la señal es de 5 kHz y el valor de  $C_1$  pudo ser reducido de  $1000\mu\text{F}$  a  $10\mu\text{F}$ .

### 3.1.1 Cálculo de la relación salida-entrada

Para analizar el procesamiento de datos requerido para obtener el desplazamiento del LVDT, hay que tener en cuenta que su primario se alimenta con una señal sinusoidal de amplitud y frecuencia fijas [ $V_1 = A_1 \cos(\omega t)$ ] y la amplitud de la sinusoidal de salida de su secundario varía desde cero hasta un valor máximo  $A_{max}$ , [ $V_2 = \pm A_2 \cos(\omega t - \phi) + r(t)$ ; donde  $0 \leq A_2 \leq A_{max}$  y  $r(t)$  es el ruido]. Esta variación de amplitud es la que nos indicará la posición para un sentido u otro del con respecto a la posición nula del LVDT. Las dos sinusoidales anteriores se digitalizan con convertidores analógicos a digital de 14 bits y se toman 4 muestras por periodo de cada una de ellas. Esto se debe a las características del ADC, ya que su velocidad máxima de muestreo es de 19.2kHz. Debido a estas características del ADC, se genera una señal sinusoidal de 4.8kHz ( $19.2\text{kHz}/4$ ) para que se puedan obtener 4 muestras por periodo de las señales de entrada.

La resolución teórica en escala completa de  $\pm 1000\text{s}$  o  $\mu\text{m}$  con 14 bits es de  $0.1221\text{s}$  o  $\mu\text{m}$  y no de  $0.1\text{s}$  o  $\mu\text{m}$  como se requiere. Pero como se verá más adelante en el capítulo 4, el algoritmo de procesamiento interpola en un nivel submuestra y obtiene lecturas estables con resolución de  $0.1\text{s}$  o  $\mu\text{m}$ .

Como los voltajes de referencia del ADC para las dos entradas son iguales, la ganancia del amplificador diferencial debe ser tal que la salida máxima del secundario ocupe todo el intervalo dinámico del convertidor analógico a digital. Por lo anterior, cuando las dos señales se encuentren en su valor mínimo o máximo, las secuencias de salida de los dos convertidores serán las mismas.

El ruido a la salida del secundario es de modo común, así que es eliminado con el amplificador diferencial. También existe un rizo en los voltajes de polarización de la tarjeta que se deben al oscilador de alta frecuencia. Como este ruido es de alta frecuencia, no afecta al desempeño del sistema, ya que es absorbido por el algoritmo de medición (son las

componentes de alta frecuencia de la transformada de Fourier y éstas son filtradas por el algoritmo).

Para obtener la lectura se debe obtener primero la relación de transformación; se toma en cuenta el voltaje del primario para evitar errores debidos a la variación del mismo.

$$a = \frac{V_2}{V_1} \quad (3.1)$$

Nótese que el valor de amplitud de una señal sinusoidal es independiente de su fase. No obstante, existen métodos numéricos para el cálculo de la amplitud en donde influye la fase. Por lo tanto, el retardo implícito en los circuitos electrónicos de acondicionamiento de señal y respuesta del L/RVDT no deben influir en el cálculo.

El valor de la amplitud de una señal sinusoidal digitalizada se puede obtener de varias formas, la elección de una de ellas depende del grado de exactitud que se requiera. Obsérvese la figura 3.2, en donde se muestra una señal sinusoidal digitalizada.

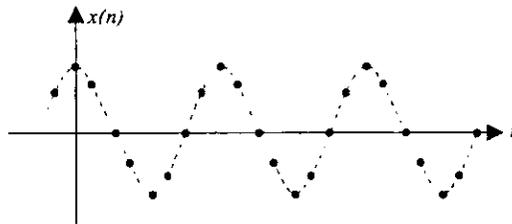


Figura 3.2. Muestras de una señal sinusoidal con diferente fase.

Puede observarse en la figura que la forma obtenida con las muestras depende de la fase, las muestras oscuras son distintas de las muestras claras aunque pertenezcan a una misma señal y se tenga la misma tasa de muestreo. Entre menos muestras se tengan de la señal, la diferencia es mayor. El método elegido debe considerar esta situación para evitar errores en el cálculo de la amplitud.

La tabla 3.1 muestra la comparación de varios métodos para obtener el valor de amplitud de una señal sinusoidal (coseno) digital con distintas fases y distinto número de muestras. El valor teórico de la amplitud es 1.

Los valores de la tabla 3.1a y 3.1b fueron obtenidos utilizando Matlab. Como puede observarse, los mejores métodos son los de procesamiento digital de señales que corresponden a la transformada rápida de Fourier y la autocorrelación. Además el valor calculado de amplitud es independiente de la fase; pero otra ventaja de la transformada de Fourier es,

como ya se estudió, que se puede obtener la fase de la señal a partir del espectro. Sin duda alguna es más fácil programar una autocorrelación que una transformada rápida de Fourier, pero la desventaja de la autocorrelación es que el resultado se ve muy afectado por el ruido de la señal. Precisamente esto es mayor en la espiga que contiene la potencia de la señal, que es con la que se calcula el valor de la amplitud. Esta espiga contiene la suma de la potencia de la señal y el ruido. La correlación tendría un mayor uso en la recuperación de una señal inmersa en ruido.

Método\Defasamiento	0°	30°	50°	60°	90°
Valor máximo	1	0.997859	0.996195	0.997859	1
Integración trapecio	0.947554	0.960740	0.973335	0.980274	0.991353
Autocorrelación	1.0	1.0	1.0	1.0	1.0
FFT	1.0	1.0	1.0	1.0	1.0

**Tabla 3.1a.** Cálculo de la amplitud tomando 32 muestras.

Método\Defasamiento	0°	30°	50°	60°	90°
Valor máximo	1	0.866025	0.766044	0.866025	1
Integración trapecio	0.588666	0.804133	0.829332	0.804133	0.588666
Autocorrelación	1.0	1.0	1.0	1.0	1.0
FFT	1.0	1.0	1.0	1.0	1.0

**Tabla 3.1b.** (Continuación) Cálculo de la amplitud tomando 4 muestras

Para calcular el valor de la amplitud de una señal sinusoidal, se obtiene la magnitud de la componente de la transformada de Fourier. De aquí que:

$$a = \frac{\text{abs}(FFT|W(f_0))}{\text{abs}(FFT|W(f_0))} \quad (3.2)$$

Donde  $f_0$  denota la espiga correspondiente a la componente de frecuencia de la señal. En nuestro caso se toman 4 muestras por periodo, no obstante, el cálculo de la transformada de Fourier se implementa con 64 muestras de la señal, es decir, 16 periodos. Por lo tanto, la espiga que contiene la información sobre la amplitud y fase de la señal es la espiga número 16, tomando en cuenta que la espiga cero es la que contiene la componente de corriente directa de la señal; es decir, en (3.2)  $f_0=16$ .

Para obtener la amplitud de la señal sinusoidal basta calcular el módulo del la componente en frecuencia correspondiente. La forma más sencilla, si la representación de los números complejos es binómica (nuestro caso), es obtener la raíz cuadrada de la suma de las dos partes al cuadrado.

### 3.1.2 Cálculo del desplazamiento

Para obtener el valor de desplazamiento, hay que tener en cuenta que el sistema hace una adquisición de datos relativa, es decir, los valores de amplitud de las señales de entrada al AIC ( $V_1$  y  $V_2$ ), cuando alcanzan su amplitud máxima, cubren todo el intervalo dinámico del convertidor y son iguales en amplitud (cuando el desplazamiento en el L/RVDT es  $\pm 100\%$ ). La ganancia del amplificador diferencial está calculada para que la señal ocupe todo el intervalo dinámico del convertidor y se obtenga la mayor resolución. De igual forma, el convertidor de la señal de entrada al transductor está programado para que esta señal también ocupe todo el intervalo dinámico del convertidor. Por lo tanto, de (3.2), el máximo valor del cociente será 1, es decir,  $a$  variará entre 0 y 1.

Para obtener los valores de amplitud reales de las señales de entrada, ( $V_1$ ), y salida ( $V_2$ ), se debe multiplicar la amplitud calculada de cada señal ( $A_1$ ,  $A_2$ ) por su respectiva ganancia (ya que los valores numéricos están normalizados). Si denotamos al voltaje de amplitud de la señal de entrada digitalizada como  $A_1$  y el voltaje de amplitud de la señal de salida digitalizada como  $A_2$ , entonces:

$$V_1 = A_1 G_1 \quad (3.3)$$

$$V_2 = A_2 G_2 \quad (3.4)$$

el desplazamiento del L/RVDT se obtiene con la fórmula:

$$D = \frac{V_2 x_{max}}{V_1 n_{max}} \quad (3.5)$$

donde  $x_{max}$  es el desplazamiento máximo del transductor en sus unidades correspondientes y  $n_{max}$  es la relación de transformación máxima  $V_2/V_1$  que corresponde a la máxima relación de  $a$ . Cuando las señales alcanzan su máxima amplitud se tiene que:

$$n_{max} = \frac{G_2}{G_1} \quad (3.6)$$

sustituyendo (3.6) en (3.5) se tiene que:

$$D = \frac{A_2}{A_1} x_{max} \quad (3.7)$$

por último se tiene que:

$$D = a \cdot x_{max} \quad (3.8)$$

Se puede ver que el valor del desplazamiento se obtiene directamente calculando el valor de la relación  $a$  y multiplicándolo por el desplazamiento máximo del L RVDI.

Otro método posible que se utilizó en este proyecto es obtener pares ordenados del valor de  $a$  contra el desplazamiento  $D$ , en otras palabras, obtener la curva de calibración del instrumento y encontrar el valor que ajusta los valores de  $a$  al desplazamiento.

La otra parte importante es obtener el valor de fase de la señal del secundario con respecto al primario, esto también se puede hacer a partir del resultado de la FFT, como se describe en la siguiente sección. El valor de fase que se toma como referencia es el del primario, si la fase del secundario es cercana a esta fase, se enviará un resultado positivo a la computadora, de lo contrario se enviará un valor negativo.

### 3.1.3 Estimación de la fase

Para obtener la fase de una componente espectral, se obtiene la tangente inversa de la relación del negativo de su parte imaginaria entre la real (ver capítulo 2), es decir:

$$\theta = \arctan \frac{b}{a} \quad (3.9)$$

Donde  $a$  y  $b$  son la parte real e imaginaria de la componente en frecuencia respectivamente.

Para evitar el cálculo de la tangente inversa, se pueden analizar algunas propiedades de la función tangente para saber cuándo existe un cambio de fase. Obsérvese la figura 3.3.

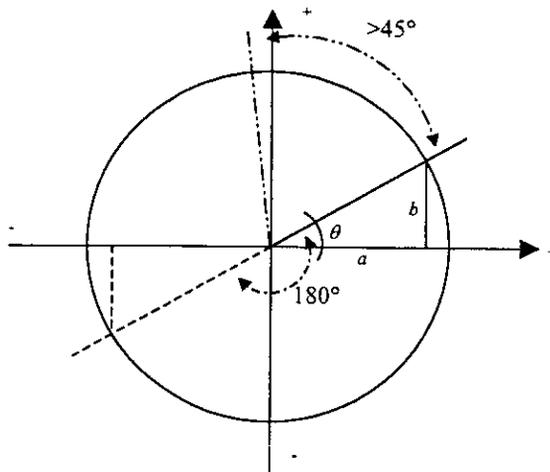


Figura 3.3. Función tangente.

Según la figura se tiene que

$$\tan(\theta) = \frac{b}{a} \quad (3.10)$$

En la figura 3.3 se puede observar que cuando existe un cambio de fase de  $180^\circ$  las dos componentes cambian de signo; pero se puede observar también en la figura que si la componente mayor cambia de signo, existe al menos un defasamiento de  $45^\circ$ . Por lo tanto, para verificar si hubo un cambio de fase, se pueden verificar los signos de la parte mayor (real o imaginaria) de la componente en frecuencia; si son iguales es que no hubo defasamiento indicando un desplazamiento positivo en el L/RVDT y la lectura es positiva, de lo contrario hubo defasamiento indicando un desplazamiento negativo y la lectura es negativa.

El método propuesto es, entonces, obtener la amplitud de la señal de entrada y de la señal de salida, hacer la división entre las señales y obtener el signo de la lectura a partir de la fase. Para estabilizar la lectura contra variaciones del tiempo se obtiene el promedio de 10 lecturas y se envía a la computadora para ser mostrado.

Teniendo en mente el algoritmo descrito en 3.1, se puede comenzar a describir la implementación del mismo en la DSK. A continuación se explica la forma en la que se llevó a cabo esta tarea.

## 3.2 Programación del algoritmo de medición en lenguaje ensamblador

La programación en lenguaje ensamblador no es tarea fácil, pero permite al diseñador aprovechar los recursos del dispositivo al máximo. Esto tiene la ventaja de optimizar el tiempo de procesamiento y el uso de memoria y así realizar el procesamiento de las señales de una manera rápida y eficiente.

### 3.2.1 Programa principal

En resumen, el DSP tiene que realizar 4 tareas principales casi todas al mismo tiempo: 1) Generar la señal sinusoidal que alimentará la entrada del transductor, esta señal siempre debe estar presente y debe tener poca distorsión armónica. 2) Hacer la adquisición de datos de las señales de entrada al AIC, esta tarea no se tiene que realizar en todo momento, se realiza antes de calcular cada lectura. 3) Realizar el procesamiento para obtener la lectura, esta tarea se realiza después de la adquisición de las señales. 4) Transmitir los resultados a la PC, esta tarea se realiza al final del procesamiento. Para facilitar la programación del AIC, se comprobó experimentalmente que es conveniente hacer que el DSP atienda en todo momento las peticiones de interrupción generadas por el mismo, no obstante que la asignación a memoria de datos correspondientes a muestras cuantizadas, se lleva a cabo solamente en períodos específicos asignados a tal tarea. De esta forma, en la tarjeta DSK se están realizando 3 tareas principales al mismo tiempo. Como se muestra en la figura 3.4.

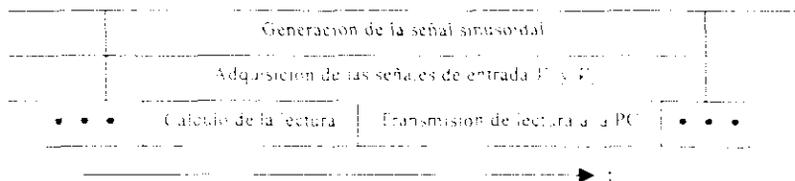


Figura 3.4. Tareas que desempeña el DSP.

La figura 3.5 muestra el diagrama de flujo, basado en el algoritmo de medición desarrollado en 3.1, del programa en la tarjeta DSK.

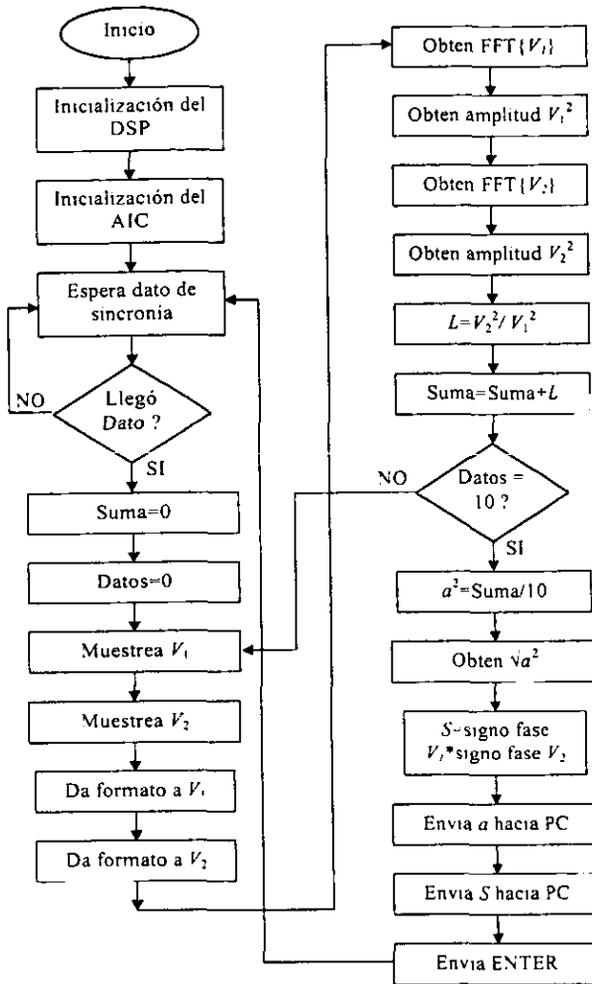


Figura 3.5. Diagrama de flujo del algoritmo de medición en el DSP.

Además del algoritmo mostrado en el diagrama de flujo de la figura 3.5, el DSP realiza las tareas de generación de la señal sinusoidal y adquisición. Estas tareas se programan con las interrupciones de transmisión y recepción del puerto serie.

La inicialización del DSP tiene que establecer el modo de memoria del DSP, iniciar el AIC y habilitar las rutinas de interrupción del puerto serial síncrono. También se inicializan las variables globales que el mismo programa utiliza.

En el ciclo principal del DSP, se espera un dato por puerto serie RS232 para sincronizarse con la PC. En cuanto recibe el dato, comienza la adquisición de las señales  $V_1$  y  $V_2$ . Como el AIC no tiene dos convertidores analógico-digital pero tiene dos canales de entrada para el mismo convertidor, primero se muestrea  $V_1$  y luego se cambia de canal para muestrear  $V_2$ . Cuando termina de adquirir las 64 muestras de cada señal, se cambia el formato de los datos adquiridos de punto fijo a punto flotante a 64 bits. Tal tarea se realiza fácilmente normalizando cada dato.

Cuando los dos vectores  $V_1$  y  $V_2$  tienen el formato numérico con el que el algoritmo opera, se calcula la transformada de Fourier del vector de datos de  $V_1$  y se lee la espiga 16 que tiene la información sobre la amplitud y fase de la señal. Se obtiene el valor de amplitud al cuadrado y se almacena temporalmente en memoria. Después se calcula la transformada de Fourier del vector de datos  $V_2$  y se obtiene también la amplitud al cuadrado de la señal. Posteriormente se realiza la división y se almacena el resultado en una variable temporal, con el fin de ir acumulando las lecturas obtenidas hasta sumar 10 consecutivas y obtener el promedio  $a^2$ .

Cuando se tiene la lectura verdadera  $a^2$ , se obtiene la raíz cuadrada; se analizan los signos de la parte mayor (real o imaginaria) de las componentes de las dos señales; si los signos son iguales se envía un "cero" indicando que el dato es positivo, si son distintos se envía un "uno" que indica que la lectura es negativa. Primero se envía la lectura  $a$ , comenzando por la parte baja de la mantisa, seguida por la parte media y luego la parte alta, por último se envía el exponente. En cada caso, como la transmisión es de 8 bits y los datos son de 10, se envía la parte menos significativa y luego la parte más significativa del dato.

Después de enviar la lectura,  $a$ , se envía el signo,  $S$ , y por último se envía un carácter de control para indicar que la transmisión fue correcta, el carácter enviado es *ENTER*.

El ciclo principal llama a la subrutina que calcula la FFT y ésta a su vez llama a las subrutinas que realizan las operaciones aritméticas básicas, como se muestra en la figura 3.6. El programa principal también llama a las subrutinas de suma, multiplicación, división y raíz cuadrada directamente.

La manera en la que se implementó el programa fue considerando que toda operación se hace en las subrutinas, así la subrutina principal de la FFT utiliza casi todos los recursos en cuanto a registros se refiere (acumulador, acumulador buffer, registros auxiliares, repetición de bloques). La subrutina de la FFT no realiza respaldo de estos registros, por lo tanto, el programa tiene que controlar los datos almacenados en los registros utilizados.

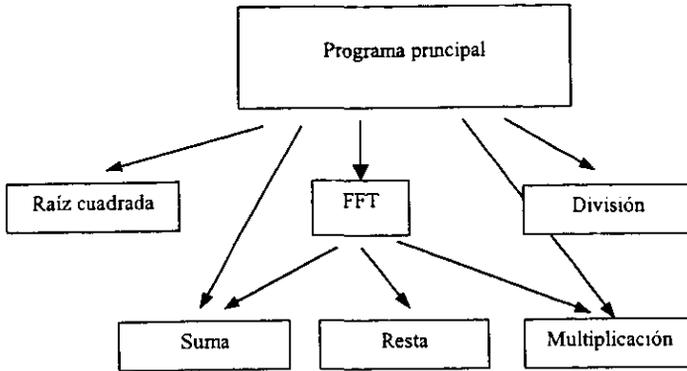


Figura 3.6. Llamadas a subrutinas.

Las subrutinas de las operaciones aritméticas también modifican los acumuladores y en algunos casos se modifican un par de registros auxiliares. No obstante, si se adopta el enfoque sobre la base de subrutinas, la programación no es complicada ni más tardada.

Para llamar a la subrutina de la FFT se debe pasar como parámetro la dirección donde comienza el vector real de datos. También se deben iniciar dos variables previamente a la llamada: una es la longitud del vector menos 1,  $N-1$ , y otra es el número de bits con los que se puede representar  $N$ , es decir,  $L$  (recuérdese que  $N=2^L$ , ver capítulo 3). También debe existir en memoria un vector llamado  $WN$ , que contiene los valores  $\omega_N$ 's para calcular las mariposas de la FFT. Este vector depende de  $N$ , y está en orden, es decir, desde  $\omega_N^0$ , hasta  $\omega_N^{N/2}$ . Si se cambia el valor de  $N$ , no es necesario modificar la subrutina de la FFT, solo es necesario cambiar  $N$ ,  $L$  y  $WN$ . Hay que tener en cuenta que la subrutina de la FFT utiliza bastantes variables temporales. De esta forma, el programa principal no debe utilizar de manera directa estas variables, ya que son modificadas por la subrutina. El resultado es almacenado en un vector llamado  $FOUR$ . Este vector es complejo; la parte real e imaginaria de cada componente es guardada consecutivamente en memoria, es decir, cada componente del espectro utiliza dos espacios numéricos de memoria (recuerde que el espacio necesario para cada número representado en el DSP es de 4 palabras).

Las subrutinas de suma, resta, multiplicación y división utilizan dos variables temporales llamadas  $W$  y  $WT$ , cada una de 4 palabras. Las subrutinas trabajan sobre estas variables. Por ejemplo, la suma es  $WT=W+WT$ , los dos operandos deben guardarse en las variables y el resultado se almacena en una de ellas. La idea es que cuando se requieran realizar varias sumas, solamente se tenga que modificar la variable  $W$  y el resultado final estará en  $WT$ . La resta trabaja igual que la suma, hay dos subrutinas de resta  $WT=W-WT$  y  $WT=WT-W$ . La división también se realiza

con estas dos variables  $WT-WWT$ . La operación de la raíz cuadrada también utiliza una variable temporal, es decir,  $WT-WWT$ . La única operación en la que los operandos no son las variables temporales es la multiplicación, los operandos son apuntados por los registros auxiliares y el resultado es almacenado en una variable temporal,  $W-1R0*AR1$ . Se diseñó así para facilitar los ciclos de multiplicación-acumulación.

### 3.2.2 Interrupciones

Como ya se mencionó, con las interrupciones se maneja la síntesis y adquisición de datos; la interrupción de transmisión del puerto serie genera la señal sinusoidal y la interrupción de recepción del puerto serie realiza la adquisición de datos.

La figura 3.7 muestra el diagrama de flujo de la subrutina de interrupción de recepción del puerto serie sincrónico.

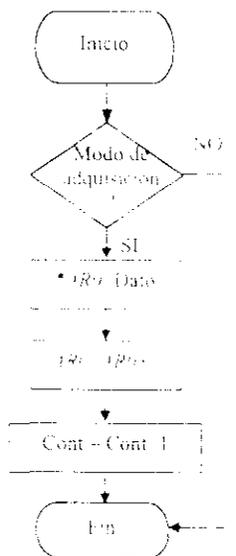


Figura 3.7. Diagrama de flujo de la interrupción de recepción.

La interrupción de recepción es bastante sencilla. Solo verifica si el programa se encuentra en modo de adquisición; si lo está, guarda el dato en la localidad de memoria apuntada por  $AR0$  y lo incrementa. También incrementa un contador que indica el número de muestras que se han tomado de la señal. Si no se encuentra en modo de adquisición sale de la interrupción.

La figura 3.8 muestra el diagrama de flujo de la rutina de interrupción de transmisión del puerto serie síncrono.

Esta interrupción verifica si el programa se encuentra en modo de adquisición, para saber si se tiene que reprogramar el AIC. El AIC se reprograma cada vez que se adquiere el último dato de cada secuencia para cambiar de canal (IN o AUX). Por defecto siempre transmite el dato siguiente de la señal sinusoidal.

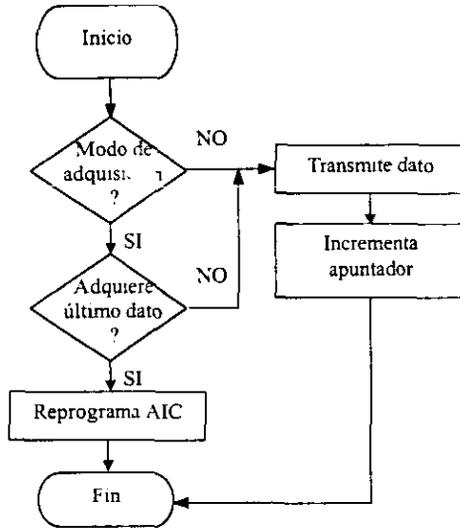


Figura 3.8. Diagrama de flujo de la interrupción de transmisión.

### 3.3 Programación de la interface con el usuario (LVDT.EXE)

El programa en la PC es muy importante ya que le da un aspecto de operabilidad al instrumento. Debe ser una interfaz fácil de utilizar y de aspecto amigable con el usuario. El programa realiza tareas muy sencillas ya que el DSP obtiene casi el resultado final. En general, la computadora solo tiene que leer el dato, ajustarlo con un algoritmo sencillo a la curva de calibración y mostrarlo en pantalla.

También existen otras tareas que debe realizar la computadora como: cargar el programa en el DSP, ejecutarlo, sincronizarse en la comunicación, visualizar la lectura y atender las opciones del usuario.

El diagrama de flujo del programa se muestra en la figura 3.9.

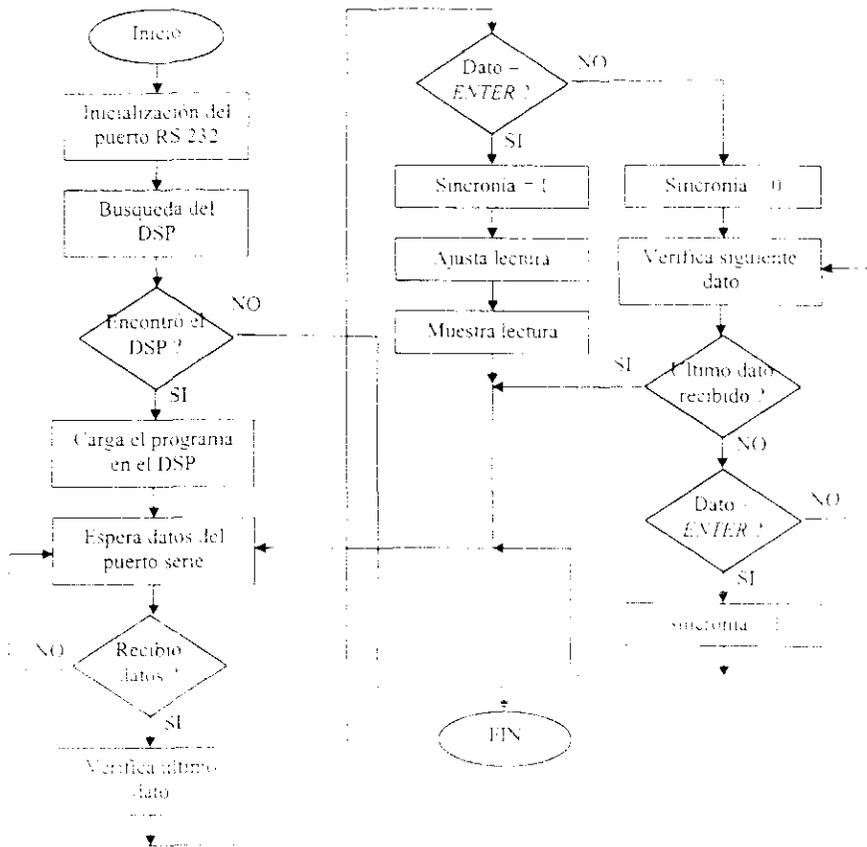


Figura 3.9. Diagrama de flujo del programa de la PC.

Como ya se había mencionado, el DSP espera un dato de la PC antes de enviar cada lectura. Este dato es enviado por la PC cada medio segundo, la rutina está programada en un temporizador; cuando se genera el evento solo transmite el dato.

El programa en la computadora está hecho en Visual C++ 6 utilizando programación orientada a objetos (OOP) y las clases de Microsoft (MFC) para tener una interfaz amigable y fácil de utilizar. El programa primero inicia el puerto serie RS232 para establecer la comunicación con el DSP. Cuando el puerto está abierto, verifica si el DSP está conectado; si no lo detecta termina con el programa, si el DSP está presente baja el programa en ensamblador a la tarjeta.

Una vez que la tarjeta ya está ejecutando el programa que acaba de ser cargado por la computadora, la PC espera un evento por el puerto serie. Si llegan datos por el puerto serie, verifica que el último dato sea el dato de control (*ENTER*). Si es así, entonces el programa ya se sincronizó con la tarjeta y los datos son válidos. El dato es leído, ajustado y mostrado en pantalla.

Si el carácter de control no es *ENTER*, entonces el programa busca en todos los datos recibidos y toma el primer dato que sea igual a *ENTER*; espera la siguiente secuencia de datos y verifica el carácter de control, si es *ENTER* entonces ya está en sincronía, si no, vuelve a buscarlo.

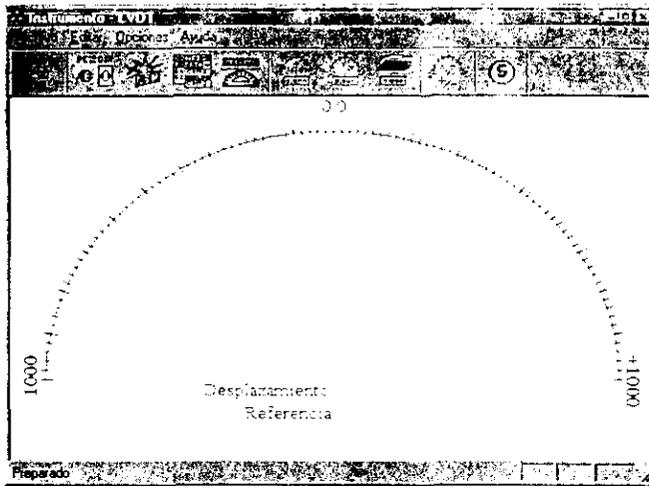
En general, el programa en la PC es sencillo. Para ver cómo está organizado en Visual C++, el anexo B muestra el diagrama de clases de este programa junto con sus métodos y variables.

### 3.4 LVDT.EXE. Manual de operación

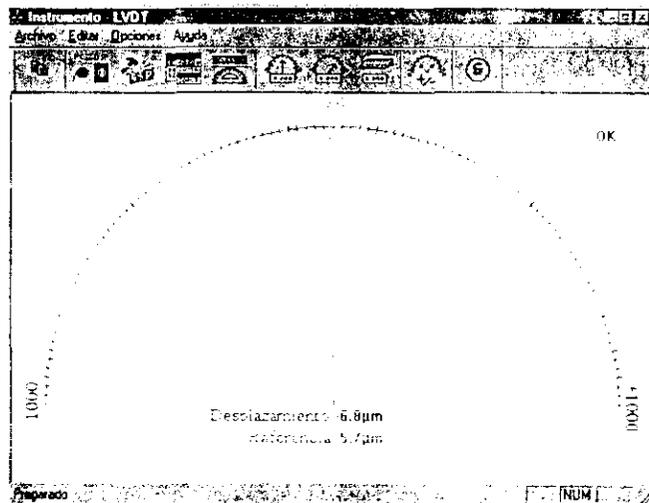
El programa de la computadora, como se ha estado mencionando, fue hecho en Visual C++ 6. Esto quiere decir que es una aplicación para Windows y los requerimientos del sistema son:

- Windows 98.
- Las bibliotecas MFC42D.DLL, MFC042D.DLL y MSVCRTD.DLL.
- 2.5 Mb en disco duro.
- 16 Mb en RAM.
- Puerto serie RS232.

A continuación se describen las opciones con las que cuenta el programa, y la forma de operarlo. La aplicación funciona dentro de la filosofía de Windows para facilitar su uso y para que cualquier usuario con nociones del sistema operativo de Windows 98 pueda encontrar fácilmente las opciones dentro de los menús. La figura 3.10 muestra la pantalla al comenzar el programa en la computadora.



a)



b)

Figura 3.10. Programa LVDY. a) Al iniciar. b) Al conectarse y sincronizarse.

La aplicación consta de 4 menús principales: **Archivo**, **Editar**, **Opciones** y **Ayuda**. También cuenta con 10 botones que conforman la barra de herramientas. El área principal consta de las siguientes partes:

- Un área con la etiqueta “Desplazamiento” donde se muestra la lectura numéricamente.
- Un área con la etiqueta “Referencia” donde se muestra el valor de la referencia numéricamente. Esta referencia es establecida por el usuario al seleccionar las opciones *Precargar Valor* y *Establecer 0*. Su valor por omisión es cero.
- Un área que simula una escala analógica donde se muestra la lectura con una aguja.

Tanto la aguja como las lecturas sólo se habilitan cuando se establece la conexión entre la PC y la DSK, de lo contrario no se muestran en pantalla.

### 3.4.1 Menú Archivo

El menú de **Archivo** solamente contiene la opción *Salir* como se muestra en la figura 3.11.



Figura 3.11. Opciones del menú Archivo.

Esta opción siempre está habilitada. Se puede salir del programa en cualquier momento no importando el estado en el que éste se encuentre. Automáticamente la aplicación cierra el puerto si es que éste se encuentra abierto y envía un reset al DSP, si es que se encuentra trabajando.

### 3.4.2 Menú Editar

El menú **Editar** solamente contiene la opción *Copiar* como se muestra en la figura 3.12.

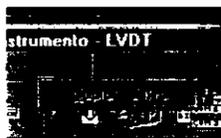


Figura 3.12. Opciones del menú **Editar**.

Esta opción sólo se habilita cuando la aplicación obtiene lecturas de la DSK, y copia el valor mostrado en "Desplazamiento" al portapapeles para que pueda ser transferido a cualquier otra aplicación de Windows que requiera el dato.

### 3.4.3 Menú Opciones

En el menú de **Opciones** se encuentran casi todas las opciones de la aplicación que son *Conectar*, *Reset*, *Configuración*, *Modo*, *Establecer 0*, *Restablecer 0*, *Precargar Valor* y *Alternar Signo* como se muestra en la figura 3.13.

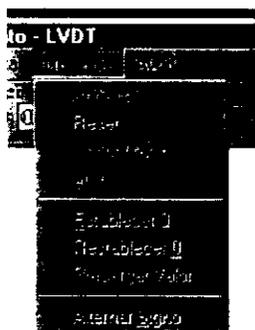


Figura 3.13. Opciones del menú **Opciones**.

*Conectar* se habilita cuando comienza la aplicación o después de dar un reset a la DSK. Se deshabilita cuando se ha establecido la comunicación entre la PC y la DSK. *Conectar* verifica que la tarjeta DSK esté conectada; si esta conectada carga el programa que ejecutará el DSP, cuando termina de cargarlo se deshabilita la opción.

*Reset* se habilita después de establecer la comunicación entre la PC y la DSK, cuando no hay comunicación se deshabilita esta opción. *Reset* detiene la ejecución del programa en el DSP y detiene la comunicación entre la computadora y la tarjeta.

*Configuración* se habilita cuando comienza la aplicación o después de dar un reset a la DSK. Se deshabilita cuando se ha establecido la comunicación entre la PC y la DSK. *Configuración* establece los parámetros del puerto serie RS232 mostrando una caja de diálogo como la que se muestra en la figura 3.14.

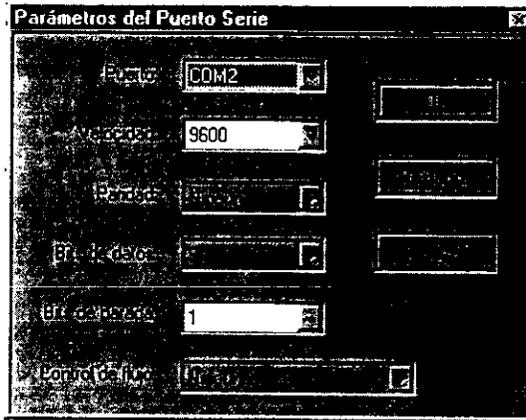


Figura 3.14. Caja de diálogo para establecer parámetros de comunicación.

Como puede observarse en la figura anterior, sólo se pueden establecer 3 parámetros: el puerto, la velocidad de transmisión y los bits de parada. Los demás parámetros tienen valores que no pueden ser cambiados pero se muestran en la caja para que el usuario los pueda ver por si necesita saber su valor en algún momento determinado.

*Modo*. Esta opción siempre se mantiene habilitada y sirve para que el usuario ingrese el tipo de medición que se llevará a cabo, desplegando una caja de diálogo como la que se muestra en la figura 3.15.

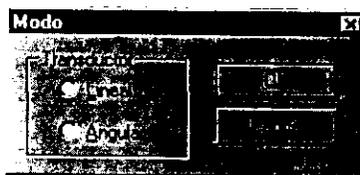


Figura 3.15. Caja de diálogo para establecer modo de medición.

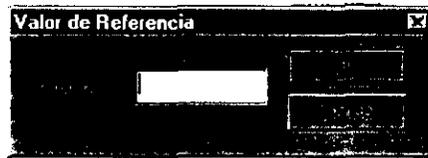
Quando se cambia el tipo de medición cambian las unidades de las lecturas mostradas en pantalla, para que el usuario sepa qué tipo de medición se está llevando a cabo. Es importante establecer este modo ya

que las lecturas son diferentes y su valor de calibración depende del tipo de transductor que se utilice.

*Establecer 0.* Esta opción se habilita cuando existe comunicación entre la PC y el DSP. Cuando se muestran lecturas en pantalla y se elige esta opción, la lectura actual es tomada como referencia o como *cero*, no importa si ya existe una referencia distinta de cero, esta opción hace que la lectura de desplazamiento presente sea tomada como referencia.

*Restablecer 0.* Esta opción se habilita cuando existe comunicación entre la PC y el DSP y simplemente pone a cero la referencia, así la lectura será absoluta.

*Precargar Valor.* Esta opción se habilita cuando existe comunicación entre la PC y el DSP y sirve para establecer el valor de referencia desde el teclado. La figura 3.16 muestra la caja de diálogo que aparece al ejecutar esta opción.

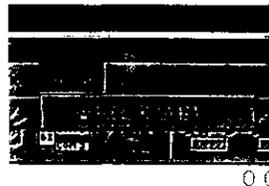


**Figura 3.16.** Caja de diálogo para establecer la referencia desde el teclado.

*Alternar Signo.* Esta opción se habilita cuando existe comunicación entre la PC y el DSP y sirve para cambiar el signo de las lecturas. Así se puede alternar el signo que se quiera para cualquier sentido del desplazamiento.

### 3.4.4 Menú Ayuda

El menú de **Ayuda** solamente contiene la opción *Acerca de LVDT* como se muestra en la figura 3.17.



00

**Figura 3.17.** Opciones del menú **Ayuda**.

Esta opción siempre está habilitada y muestra información acerca de la versión del programa.

Todos los menús cuentan con teclas aceleradoras para facilitar el acceso desde el teclado. Estas teclas están subrayadas y se activan presionando la tecla *ALT* junto con la tecla subrayada del menú que se quiera. Estando dentro de cada menú, se puede presionar directamente la tecla subrayada de cada opción para ejecutarla.

### 3.4.5 Barra de herramientas

Para facilitar el ingreso a las opciones dentro de los menús de la aplicación, se cuenta con una barra de herramientas que permite el acceso directo a las funciones más utilizadas. La barra de herramientas se muestra en la figura 3.18.

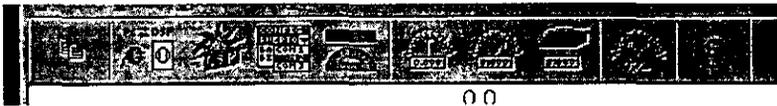


Figura 3.18. Barra de herramientas de la aplicación LVDT.

Cuando se deja el cursor del ratón sobre un icono de la barra de herramientas por un par de segundos, aparece la función que realiza el icono y además en la barra de estado aparece una breve descripción de la misma.

Las opciones que se encuentran directamente en la barra de herramientas son: *Copiar*, *Conectar*, *Reset*, *Configurar*, *Modo*, *Restablecer 0*, *Establecer 0*, *Precargar Valor*, *Alternar Signo* y *Salir*. La barra de herramientas también puede ser cambiada de lugar y colocarse fuera del área de la aplicación, incluso puede ser removida si no se utiliza.

### 3.4.6 Procedimiento típico para operar el instrumento

La operación del instrumento es bastante sencilla. A continuación se muestran los pasos a seguir para conectar el instrumento y obtener lecturas.

1. Conectar la tarjeta al puerto serie RS232 de la computadora.
2. Alimentar con energía c.a. la tarjeta DSK expandida.

3. Ejecutar la aplicación LVDT
4. Establecer los parámetros del puerto serial (Menú *Opciones* ⇒ *Configurar*).
5. Establecer el tipo de medición (Menú *Opciones* ⇒ *Modo*).
6. Establecer la comunicación entre la PC y la DSK (Menú *Opciones* ⇒ *Conectar*).
7. Esperar a que aparezca el indicador **OK** en la parte superior izquierda de la pantalla.

Después de seguir estos 7 pasos, se obtendrán lecturas en la aplicación. El paso 6 es tardado (aproximadamente 3 minutos), ya que carga el programa en ensamblador que ejecutará el DSP. Una vez terminada la conexión, se pueden utilizar las opciones antes descritas para obtener las lecturas deseadas. El indicador **OK** sirve para indicar al usuario que el instrumento esta trabajando correctamente.

## CAPÍTULO 4

# Resultados y conclusiones

En el presente capítulo se describe el desempeño que tiene el instrumento y la manera en que éste fue calibrado. Es importante hacer notar que el instrumento fue debidamente calibrado en el Laboratorio de Metrología del Centro de Instrumentos de la UNAM. El laboratorio cuenta con la infraestructura y acreditamientos oficiales ante SECOFI, por lo tanto, el instrumento cuenta con la certificación y trazabilidad a patrones internacionales. También se menciona la manera en la que se obtuvieron los valores de ajuste de las curvas de calibración.

### 4.1 Resultados

Sin duda alguna el primer resultado es que la implementación del instrumento, con calidad de producto terminado, permite su uso en los servicios que actualmente presta el laboratorio de Metrología del CI UNAM. De esta manera, dotamos al laboratorio con infraestructura de desarrollo propio, gran calidad y con la confiabilidad y respaldo que ofrecen las características de nuestro instrumento.

Otro resultado importante es la interpolación lograda mediante el procesamiento en el DSP. No obstante que el número de bits en el AIC (convertidor analógico a digital de la tarjeta DSK) no alcanza teóricamente la resolución establecida ( $0.1 \mu\text{m/s}$ ), el algoritmo de procesamiento en el DSP resuelve en un nivel submuestra, permitiendo lecturas estables con tal resolución. Adicionalmente, una mejora importante con respecto a

instrumentos comerciales similares, es la ampliación del alcance de medición. Comparando el número de niveles de cuantización de los convertidores de 14 bits  $\pm 2^{13} = \pm 8192$ , contra el correspondiente número alcanzado por el algoritmo en el DSP,  $\pm 1000/0.1 = \pm 10000$ , observamos claramente el trabajo de interpolación submuestra logrado.

El algoritmo fue programado en ensamblador debido, principalmente, a que la tarjeta DSK es una herramienta básica de desarrollo. Dentro de las limitaciones de la DSK se encuentra que la memoria disponible es la interna al DSP, por lo tanto, si el algoritmo es programado en un compilador (por ejemplo C), la memoria que se necesita para almacenar el programa rebasa las capacidades de la DSK. Otro aspecto, es que la programación en ensamblador permite utilizar el dispositivo al máximo, se pueden aprovechar todas las características del DSP optimizando los tiempos de ejecución y el tamaño del código.

Al verificar el desempeño del instrumento se encontró un error máximo de  $0.2\mu\text{m}$  en la región no lineal nula del LVDT y de  $0.5\mu\text{m}$  hacia las regiones no lineales cercanas al  $\pm 100\%$  del desplazamiento del LVDT. La estabilidad en las lecturas fue de  $\pm 1$  dígito. El desempeño se verificó enfáticamente para el transductor de desplazamiento, ya que es más confiable la generación de desplazamientos en el laboratorio que la generación de ángulos. La calibración de desplazamiento se realizó por comparación, generando desplazamientos conocidos. La calibración de ángulo se hizo por comparación, tomando las lecturas de otro equipo como patrón.

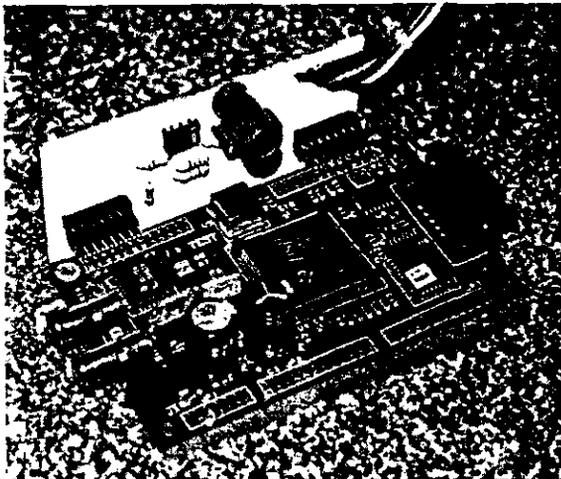


Figura 4.1. Instrumento para la medición de desplazamiento y ángulo.

El instrumento no tiene elementos eléctricos o electrónicos de ajuste o calibración como potenciómetros. Los ajustes y la calibración se hacen con software, esto tiene la ventaja de que se puede ajustar la respuesta del transductor en las regiones no lineales con modelos matemáticos no lineales. La arquitectura es muy pequeña y la placa de expansión adicional es más pequeña que la tarjeta DSK. La DSK no fue modificada en ningún aspecto, sólo se le añadieron conectores de expansión (headers). El instrumento final se muestra en la figura 4.1.

El instrumento también cuenta con una interfaz completa con todas las opciones comúnmente utilizadas para la medición de desplazamiento y ángulo. Entre estas opciones se encuentran:

- Cambio de valor de referencia.
- Precargar valor de referencia.
- Cambio de signo de la lectura.
- Cambio de transductor (lineal/angular).

En resumen el instrumento cuenta con las siguientes características:

- Resolución de 0.1  $\mu\text{m}$  o 0.1 s
- Alcance de  $\pm 1000 \mu\text{m}$  o s
- Porcentaje de error a escala máxima de 0.01%.
- Estabilidad de  $\pm 1$  dígito
- Interfaz amigable y fácil de utilizar

## 4.2 Calibración

Para calibrar el instrumento se realizó un esquema de generación de desplazamiento/ángulo. Las lecturas obtenidas directamente del procesamiento de la DSK fueron analizadas contra un patrón de desplazamiento/ángulo.

Las variables que contribuyen a la formación de la incertidumbre del proceso de calibración son muchas y muy variadas. El proceso de calibración se basa en los procedimientos internos del Laboratorio de Metrología. De esta forma, con procesos controlados y estudiados, se disminuye la influencia de tales variables. En este caso, las conclusiones

que se pueden derivar de la estimación de incertidumbre son las siguientes [10]:

- **Errores humanos.** Se trabaja cuidadosamente para eliminar totalmente, en la medida de lo posible, su efecto. La razón es que es imposible estimar su valor, por lo tanto la técnica es reducir su influencia de manera que el resultado de la calibración pueda considerarse en buena medida libre de errores humanos.
- **Generación confiable de desplazamientos y ángulos patrón.** El diseño del experimento contempla como principal variable a controlar la generación confiable de desplazamiento y ángulo patrón. La verificación en la generación de desplazamiento y ángulo con un instrumento nos proporciona una primera idea sobre la confiabilidad del experimento. Un punto aparte es el trabajo cuidadoso que se realizó de diseño, permitiendo la estabilidad y confiabilidad del proceso de calibración. De ésta manera podemos considerar que los desplazamientos lineales y angulares generados durante el experimento cuentan con un alto grado de confiabilidad y son fácilmente reproducibles.
- **La incertidumbre del instrumento.** Contempla errores de nuestro equipo. De antemano, la incertidumbre del instrumento se puede expresar como un error debido al proceso electrónico de conversión A/D en el sistema, variaciones en los componentes electrónicos, variación en los voltajes de referencia, etc. Como ya se ha mencionado, la conversión implica un proceso de cuantización, a su vez la cuantización implica un error, que se expresa en forma de dígitos menos significativos erróneos. En la incertidumbre del instrumento no se contemplan errores debidos a la no linealidad del transductor o a la inestabilidad del proceso de medición. Para la no linealidad se realiza un análisis especial de los resultados del experimento.
- **Características del transductor L/RVDT.** En este caso, el proceso completo de calibración contempla las imperfecciones del transductor. Aunque de antemano se puede evaluar el desempeño del L/RVDT (con los datos del fabricante), en el proceso de calibración no se puede realizar un análisis separado. En realidad el proceso engloba tanto los errores del instrumento como los errores del transductor.

La calibración tiene como principal objetivo "fabricar números" de forma confiable. Por tal motivo el reporte de los números se debe acompañar por la descripción del protocolo a seguir. Los números no indican gran cosa si no son acompañados de evaluaciones previas,

protocolos y procedimientos de calibración, así como de un buen análisis de datos y resultados.

Con la idea anterior en mente, el presente capítulo tiene como objetivo presentar los números obtenidos, detallando los protocolos y procedimientos observados.

### 4.2.1 Calibración del instrumento para un LVDT

- **Protocolo de medición.**

**Tarea:** Calibración de un instrumento para la medición de desplazamiento mediante intercomparación.

**Instrumento a calibrar:** Medidor de desplazamiento y ángulo con el TMS320C50. Alcance de medición  $\pm 1000\mu\text{m}$ . Resolución  $0.1\mu\text{m}$ .

**Instrumento patrón:** Calibrador de comparadores marca Mitutoyo. Modelo 521-105. No de serie 601015. Inv. UNAM 1152119. Alcance 5mm. Div. mínima  $0.0002\text{mm}$ . Exactitud  $\pm 0.2\mu\text{m}$ .

**Transductor:** LVDT modelo F HF-1056 Serie M1

- **Procedimiento.**

El montaje del experimento se puede observar en la figura 4.2.

El transductor se sujeta en una base rígida y el palpador hace contacto directamente con el calibrador de comparadores. Se conectan las terminales del transductor a las de la placa de expansión de la tarjeta DSK. Como puede observarse el montaje es robusto y sencillo, lo que permite aislar vibraciones y deformaciones. Posteriormente se inicia la aplicación del instrumento en la computadora y se conecta a la tarjeta. Una vez funcionando, se pueden obtener las lecturas del instrumento que varían al girar el tornillo del calibrador.

El procedimiento es como sigue:

1. Se gira el tornillo de manera que se genere un desplazamiento adecuado para el alcance de medición del instrumento a calibrar.
2. Se anotan las mediciones en  $\mu\text{m}$  registradas tanto por el instrumento patrón como por el instrumento a calibrar.

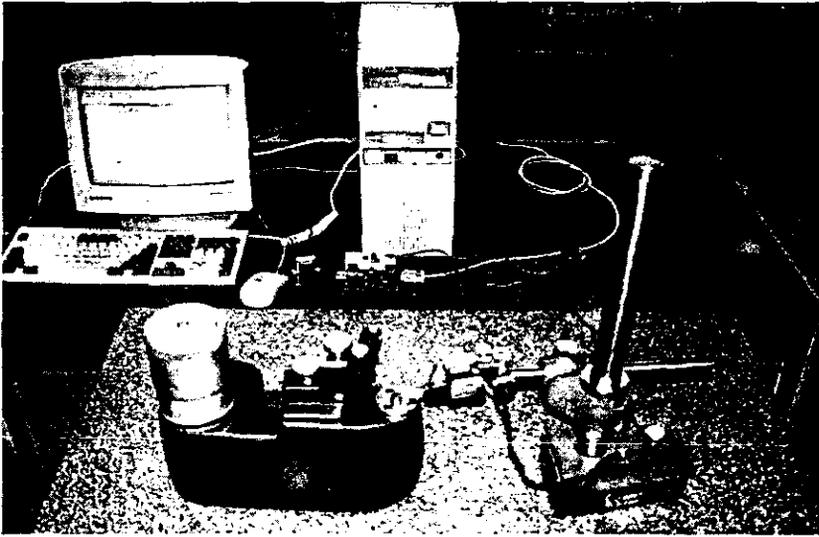


Figura 4.2. Esquema para la calibración utilizando un LVDT.

3. Se repite el paso 1 de manera que se cubra el alcance de medición del instrumento a calibrar.
4. Se realiza el mismo proceso en 1, 2 y 3 pero considerando un menor alcance de medición del instrumento. De esta forma se consigue evaluar el desempeño del instrumento en la región no lineal cercana a cero.

Las lecturas tomadas se muestran en la tabla 4.1.

Desplazamiento ( $\mu\text{m}$ )	Lectura
0	2.9
1	5
2	8.2
3	11.7
4	15.3
5	19
6	22.7
7	26.3
8	30
9	33.7
10	37.4
20	73.9
30	110.2

Desplazamiento ( $\mu\text{m}$ )	Lectura
0	-2.9
1	-4.5
2	-7.6
3	-11
4	-14.4
5	-17.9
6	-21.4
7	-25
8	-28.6
9	-32.1
10	-35.8
20	-72.2
30	-108.5

Tabla 4.1. Calibración lineal, sin ajustar constante de calibración.

Desplazamiento ( $\mu\text{m}$ )	Lectura
40	146.8
50	183.6
60	219.9
70	256.5
80	292.8
90	329.2
100	365.9
110	402.5
120	438.9
130	475.4
140	512
150	548.5
160	585
170	621.6
180	658
190	694.7
200	731.1

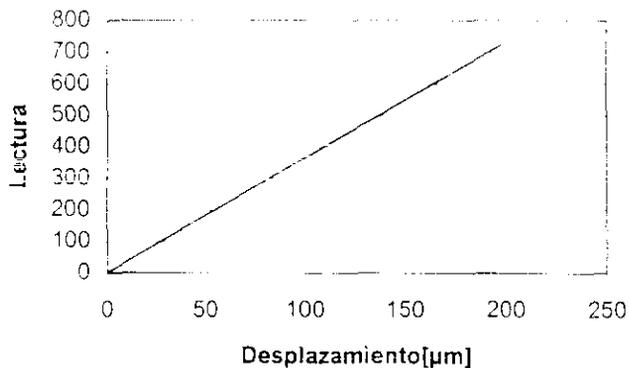
a)

Desplazamiento ( $\mu\text{m}$ )	Lectura
40	-144.9
50	-181.4
60	-217.9
70	-254.3
80	-290.4
90	-326.9
100	-363.3
110	-399.7
120	-436.3
130	-472.4
140	-508.9
150	-545.2
160	-581.5
170	-618.1
180	-654.2
190	-690.6
200	-726.9

b)

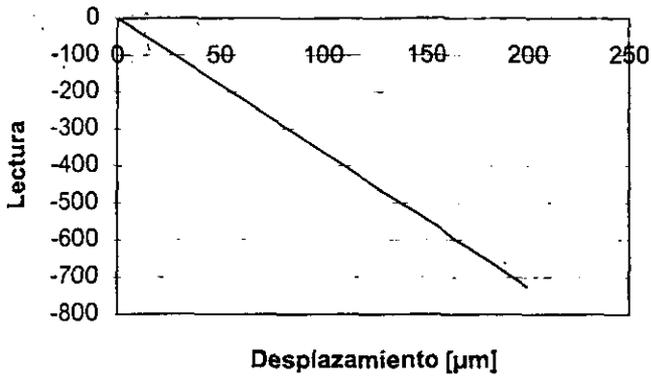
**Tabla 4.1.** Calibración lineal, sin ajustar constante de calibración (Continuación).

La tabla 4.1a muestra las lecturas en la región positiva y la tabla 4.1b muestra las lecturas en la región negativa. Las curvas de calibración se muestran en la figura 4.3.



a)

**Figura 4.3.** Curvas de calibración lineal del instrumento.



b)

Figura 4.3. Curvas de calibración lineal del instrumento (Continuación).

Para hacer un análisis más detallado de la región no lineal cercana al 0% del desplazamiento, se realizó una evaluación detallada de tal región. Los datos se muestran en la tabla 4.2 y la curva de calibración en la figura 4.4.

Desplazamiento ( $\mu\text{m}$ )	Lectura
-2	-8.05
-1.9	-7.75
-1.8	-7.38
-1.7	-7.1
-1.6	-6.78
-1.5	-6.45
-1.4	-6.12
-1.3	-5.8
-1.2	-5.45
-1.1	-5.24
-1	-4.88
-0.9	-4.5
-0.8	-4.32
-0.7	-4.07
-0.6	-3.78
-0.5	-3.57
-0.4	-3.37
-0.3	-3.18
-0.2	-3.01
-0.1	-2.89
0	-2.84

Desplazamiento ( $\mu\text{m}$ )	Lectura
0.1	-2.8
0.2	2.83
0.3	2.92
0.4	3.04
0.5	3.18
0.6	3.4
0.7	3.63
0.8	3.82
0.9	4.15
1	4.44
1.1	4.66
1.2	5.04
1.3	5.35
1.4	5.68
1.5	5.98
1.6	6.29
1.7	6.65
1.8	6.95
1.9	7.28
2	7.68

Tabla 4.2. Lecturas del instrumento en la región no lineal de LVDT.

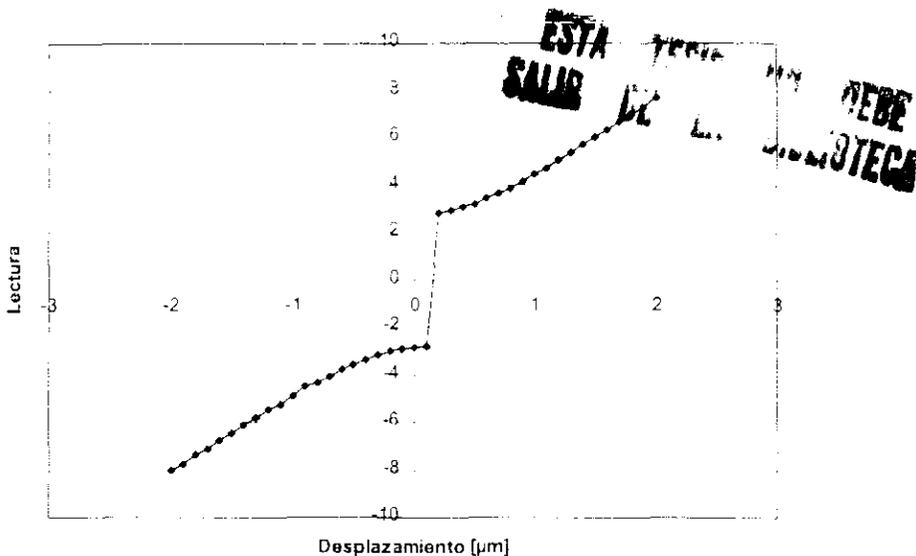


Figura 4.4. Región no lineal del transductor LVDT.

Para ajustar las lecturas obtenidas por el sistema a los datos del calibrador, se usan métodos estadísticos de regresión lineal. Cuando se tiene una variable dependiente que no se controla en el experimento pero que depende de una variable independiente o de **regresión**, se puede obtener una ecuación de predicción que recibe el nombre de **ecuación de regresión** [11].

El término **regresión lineal** implica que  $y$  está linealmente relacionado con  $x$  por la ecuación de regresión lineal. Si  $a$  y  $b$  representan las estimaciones a partir de los datos muestrales, entonces se tiene que:

$$y = a + bx \quad (4.1)$$

Donde las estimaciones  $a$  y  $b$  representan la intercepción y pendiente de  $y$  respectivamente. El símbolo  $\hat{y}$  se utiliza para distinguir entre el valor estimado que da la línea de regresión muestral y un valor experimental real observado para algún valor de  $x$ .

Existen varios métodos para obtener las estimaciones de  $a$  y  $b$ , uno de ellos es el método de "mínimos cuadrados". Con este método se encuentran  $a$  y  $b$  de tal forma que la suma de los cuadrados de los residuos es mínima. Esta suma de residuos recibe el nombre de "suma de cuadrados de los errores alrededor de la línea de regresión" y se representa por  $SSE$ . Los valores de las estimaciones se pueden obtener a partir de  $SSE$ .

$$SSE = \sum_{i=1}^n e_i^2 = \sum_{i=1}^n (y_i - \hat{y}_i)^2 = \sum_{i=1}^n (y_i - a - bx_i)^2 \quad (4.2)$$

A partir de la ecuación (4.2) se pueden obtener los valores de  $a$  y  $b$ . Se resuelve como un problema de cálculo de valores máximos y mínimos. La ecuación se deriva con respecto a las dos variables  $a$  y  $b$ , las dos ecuaciones resultantes se igualan a cero para obtener el mínimo.

Las siguientes expresiones son el resultado del desarrollo para la obtención de  $a$  y  $b$  [11].

$$b = \frac{n \sum_{i=1}^n x_i y_i - \left( \sum_{i=1}^n x_i \right) \left( \sum_{i=1}^n y_i \right)}{n \sum_{i=1}^n x_i^2 - \left( \sum_{i=1}^n x_i \right)^2} \quad (4.3)$$

$$a = \frac{\sum_{i=1}^n y_i - b \sum_{i=1}^n x_i}{n} \quad (4.4)$$

Hay que hacer notar que en las curvas de calibración de las figuras 4.3 y 4.4 se gráfica desplazamiento contra lectura porque la variable de control fue el desplazamiento generado por el calibrador; en realidad lo que se requiere es obtener el desplazamiento a partir de una lectura, es decir, el desplazamiento es la variable  $\hat{y}$  y la lectura la variable  $x$ .

Para calcular los valores de  $a$  y  $b$  se toman los datos de las tablas 4.1a y 4.1b. No se considera la región no lineal, es decir, el primer dato de cada tabla no es tomado en cuenta.

Para la parte positiva se tiene que:

$$a = -0.227374015$$

$$b = 0.273886026$$

Y para la parte negativa se tienen los siguientes valores de ajuste:

$$a = 7.00159072 \cdot 10^{-2}$$

$$b = 0.275025288$$

Como ya se mencionó el modelo matemático de ajuste es el de la ecuación (4.1). Para la región cercana a cero se utiliza un modelo no lineal logarítmico:

$$\hat{y} = a + b \ln(x) \quad (4.5)$$

Los datos utilizados son tomados de la tabla 4.2. En la curva de calibración se puede observar que la región no lineal comienza aproximadamente cuando el desplazamiento entra en el intervalo de  $-1.5$  a  $1.5 \mu\text{m}$ . También puede observarse que las lecturas están desplazadas  $0.1$

$\mu\text{m}$ , es decir, el valor mínimo que alcanza la lectura (cero absoluto) es de -2.8 (ver tabla 4.2).

El ajuste en esta región no lineal también se hace por partes: positiva y negativa. Los valores calculados de  $a$  y  $b$  para el modelo (4.5) para la parte positiva son:

$$a = -1.53464487$$

$$b = 1.639710426$$

Y para la parte negativa se tienen los siguientes valores de ajuste:

$$a = -1.69264716$$

$$b = 1.76937853$$

Una vez obtenida la lectura de la tarjeta, se ajusta con el modelo correspondiente, (4.1) o (4.5) dependiendo de la región donde se encuentre y el signo de la lectura. Este valor será el dato equivalente al desplazamiento lineal del transductor y es el valor mostrado en pantalla de la interface con el usuario.

El procedimiento empleado para calibrar el instrumento con transductor de ángulo es similar al que se empleó para calibrar el instrumento con transductor de desplazamiento.

## 4.2.2 Calibración del instrumento para un RVDT

- **Protocolo de medición.**

**Tarea:** Calibración de un instrumento para la medición de ángulo mediante intercomparación.

**Instrumento a calibrar:** Medidor de desplazamiento y ángulo con el TMS320C50. Alcance de medición  $\pm 1000\text{s}$ . Resolución 0.1s.

**Instrumento patrón:** Nivel electrónico con amplificador digital. Marca Federal. Modelo 832P-204 205. Alcance  $\pm 1000$  seg. Div. Mínima 0.1 seg. Exactitud  $\pm 1$  dígito.

**Transductor:** RVDT modelo: EGH-13. Serie: S N.

- **Procedimiento.**

El montaje del experimento se puede observar en la figura 4.5.



Figura 4.5. Esquema para la calibración utilizando un RVDT.

El transductor está sobre una placa estable generadora de ángulo que se levanta de un extremo mediante un tornillo micrométrico. En la placa se cuenta con otro transductor de ángulo junto al correspondiente del instrumento a calibrar. Este transductor está conectado al nivel electrónico Federal, instrumento patrón. El transductor del instrumento a calibrar se conecta a la placa de expansión de la tarjeta DSK. Como puede observarse, el montaje es robusto y sencillo. Posteriormente se inicia la aplicación del instrumento en la computadora y se realiza la conexión a la tarjeta. Una vez funcionando, se pueden obtener las lecturas del instrumento que varían al girar el tornillo micrométrico del calibrador.

El procedimiento es como sigue:

1. Se gira el tornillo micrométrico de manera que se genere un ángulo adecuado para el alcance de medición del instrumento a calibrar.
2. Se anotan las mediciones en segundos de arco registradas tanto por el instrumento patrón como por el instrumento a calibrar.
3. Se repite el paso 1 de manera que se cubra el alcance de medición del instrumento a calibrar.

Las lecturas tomadas se muestran en la tabla 4.3.

Federal(seg.)	Lectura
0	1.7
11.1	11.8
20.7	21.3
31.6	32.5
40.8	41.9
49.7	51
62.6	64.3
70.4	72.1
81.8	83.6
91.2	93.3
100	102.2
112.3	114.9
120.2	122.9
132.2	135.6
140.1	143.5
151.4	154.9
160.9	164.7
171	174.8
185.4	189.7
191.1	195.5
202	206.3
300	306.6
403	412.4
499	510.3
599	612.6
699	715.6
800	818.4
899	919.4
1000	1022.2

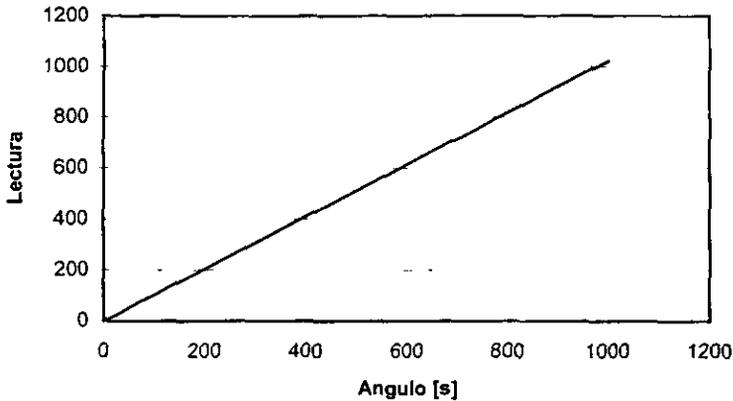
a)

Federal(seg.)	Lectura
0	-1.7
-10.4	-10.5
-21.4	-21.8
-34.1	-34.8
-48.1	-49.3
-53.8	-55.1
-61.4	-63.2
-71.2	-73.3
-80	-82.4
-90.2	-92.9
-100	-103.2
-110	-113.3
-119.7	-123.3
-130	-133.9
-140.3	-144.6
-150.9	-155.5
-159.8	-164.6
-172.3	-177.5
-180.8	-186.3
-191.7	-197.7
-202	-206.6
-301	-307.8
-401	-410.5
-500	-513.2
-601	-617
-700	-719.3
-800	-822.5
-907	-932.8
-1000	-1029

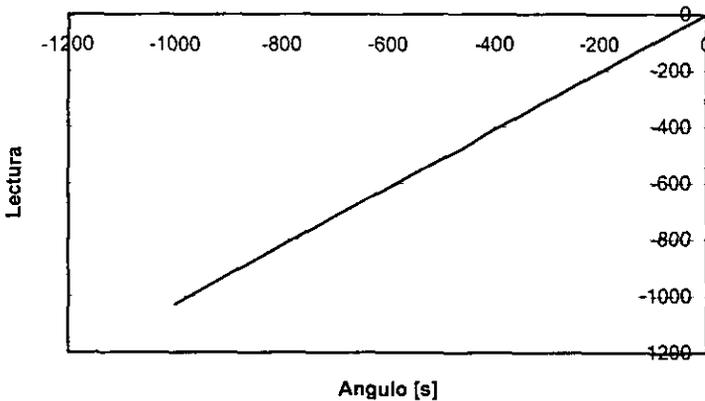
b)

**Tabla 4.3.** Lecturas del instrumento sin ajustar contra desplazamiento angular.

La tabla 4.3a muestra las lecturas en la región positiva y la tabla 4.3b muestra las lecturas en la región negativa. Las curvas de calibración se muestran en la figura 4.6.



a)



b)

**Figura 4.6.** Curvas de calibración angular del instrumento.

Para ajustar las lecturas obtenidas por el sistema a los datos del calibrador, se realiza el mismo procedimiento que se hizo para ajustar las lecturas de desplazamiento lineal.

Hay que hacer notar que en las curvas de calibración de la figura 4.6 se gráfica ángulo contra lectura porque la variable de control fue el desplazamiento angular generado por el calibrador; en realidad lo que se requiere es obtener el ángulo a partir de una lectura, es decir, el desplazamiento angular es la variable  $y$  y la lectura la variable  $x$ .

una opción de autocalibración, en la que el instrumento encuentre los valores de ajuste teniendo desplazamientos lineales o angulares conocidos. Se puede desarrollar una opción de autodiagnóstico para saber si el instrumento está funcionando adecuadamente y para programarlo de tal forma que reconozca las regiones no lineales del transductor.

Un trabajo más a fondo sería desarrollar un instrumento autónomo, de tal manera que no necesite de una PC para funcionar. Para realizar esta tarea es necesario añadir un circuito electrónico de despliegue, un teclado, software específico para manejar estos dispositivos y una arquitectura propia en donde resida el algoritmo de medición en ROM.

Finalmente, para tener un mejor desempeño del instrumento se puede agregar un convertidor analógico a digital con mayor resolución (más bits) y mayor velocidad de conversión por canal. Se podrían obtener lecturas más estables y con mayor resolución, además de que el número de lecturas por segundo sería mayor. La abundancia de datos, permitiría mayor estabilidad para mejores resoluciones.

## Anexo A

# Diagramas electrónicos

A continuación se muestran los diagramas electrónicos de los circuitos utilizados en el proyecto para acondicionar las señales del transductor. Ambos circuitos, el de salida y el de entrada, son bastante simples y utilizan componentes comerciales. No se utilizaron dispositivos de precisión aunque no se descarta completamente su uso, ya que podrían mejorar la respuesta o estabilidad del instrumento para mayor alcance o para mayor resolución.

La figura A.1 muestra el diagrama del circuito que se emplea a la salida del convertidor D/A de la tarjeta DSK.

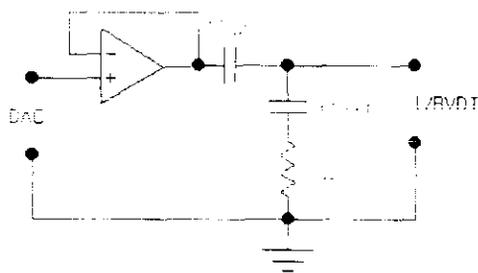


Figura A.1. Circuito utilizado para acoplar la salida del DAC al transductor.

La función de este circuito es acoplar la salida del convertidor a la entrada del transductor. La figura A.2 muestra el diagrama del circuito utilizado a la salida del transductor.

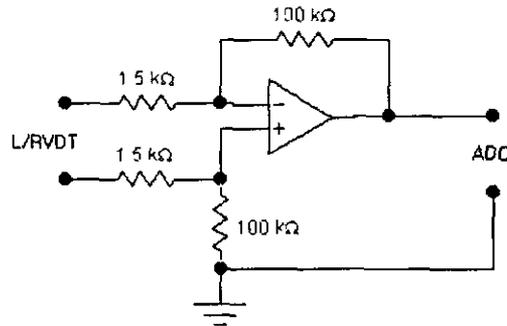


Figura A.2. Circuito utilizado para acondicionar la señal de salida del transductor.

Este circuito es un amplificador diferencial y su función es acondicionar la señal del secundario del transformador. Con esto se logra utilizar todo el rango dinámico del convertidor A/D. También filtra el ruido de modo común.

Estos dos circuitos se montan sobre una placa de circuito impreso. La placa se puede diseñar de un tamaño bastante compacto. En el proyecto, la placa se diseñó de tal manera que pudiera conectarse a la DSK sin utilizar cables. La placa se conecta con los conectores soldados a la DSK (headers). El diseño de esta placa de circuito impreso se muestra en la figura A.3.

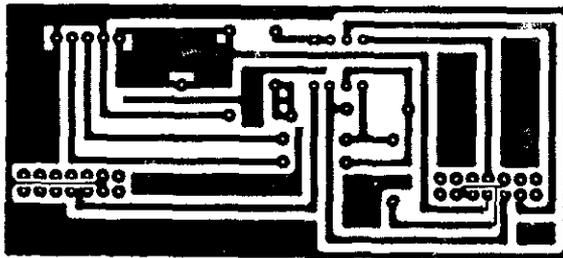


Figura A.3. Placa de circuito impreso.

La placa con los componentes montados se muestra en la figura A.4.

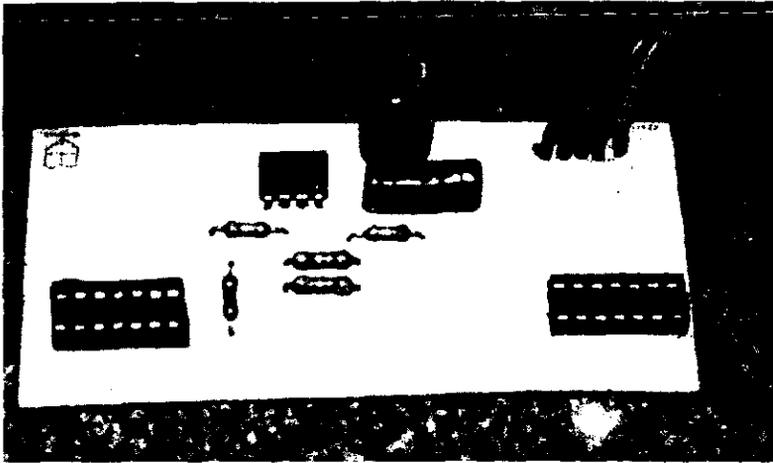


Figura A.4. Placa de circuito impreso con componentes.

## ANEXO B

# Diagramas de clases

El programa en la computadora fue hecho en Visual C++ 6. Generalmente los programas hechos para Windows o para cualquier sistema operativo con interfaz gráfica, ocupan mucho código relacionado con los gráficos. Para facilitar la comprensión de la aplicación, a continuación se muestran las clases con las que cuenta el programa así como las funciones y variables de cada clase. Con esto se tendrá una idea de cómo funciona la aplicación y en que partes se realiza cada tarea que se describe en la implementación.

La figura B.1 muestra las clases que conforman la aplicación.

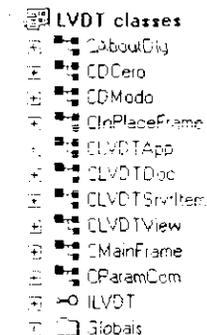


Figura B.1. Clases de la aplicación LVDT.

El programa principal cuenta con 10 clases, cada una desempeña una tarea específica. En conjunto, las clases desempeñan las tareas que van

desde construir y mostrar la ventana en la pantalla de la computadora, mostrar los menús, barra de estado y de herramientas hasta la tarea de la aplicación principal del instrumento.

La figura B.2 muestra la clase CAboutDlg.

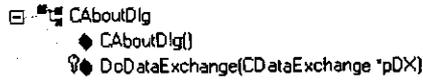


Figura B.2. Clase CAboutDlg.

Esta clase muestra la caja de diálogo que contiene la información acerca de la aplicación. Esta clase es creada por el asistente de Visual C++ siempre que se crea una aplicación.

La figura B.3 muestra la clase CDCero.

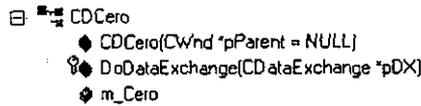


Figura B.3. Clase CDCero.

Esta clase muestra una caja de diálogo que se utiliza para precargar el valor de referencia. La caja de diálogo contiene un área de edición en la cual se introduce el valor numérico deseado.

La figura B.4 muestra la clase CDModo.

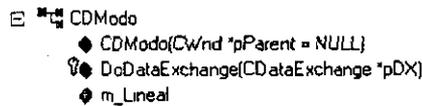


Figura B.4. Clase CDModo.

La clase CDModo se utiliza para mostrar una caja de diálogo que contiene las opciones del tipo de transductor que se está utilizando. Estas opciones son *Lineal* y *Angular*. Sólo se puede escoger una de estas dos opciones.

La figura B.5 muestra la clase CInPlaceFrame.

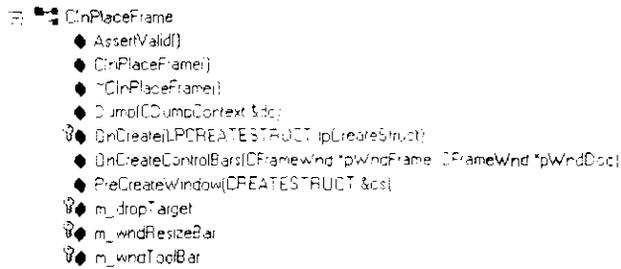


Figura B.5. Clase CInPlaceFrame.

Esta clase es creada por el asistente del Visual C++ y maneja el marco alrededor del servidor, barra de herramientas, barra de estado, y cajas de diálogos.

La figura B.6 muestra la clase CLVDTApp.



Figura B.6. Clase CLVDTApp.

La clase CLVDTApp es creada por el asistente de Visual C++ y contiene información de la aplicación. La función *InitInstance* solo se ejecuta una vez al comenzar la aplicación.

La figura B.7 muestra la clase CLVDTDoc.

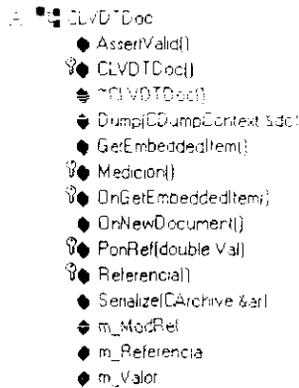


Figura B.7. Clase CLVDTDoc.

La clase de documento es una de las clases más importantes de la aplicación entera. Esta clase también es creada por el asistente de Visual C++. Contiene información de las variables importantes de la aplicación y una de las funciones que tiene que desempeñar es la de guardar y/o leer los datos a archivos cuando es necesario. A esta tarea se le denomina *serialización*.

La figura B.8 muestra la clase CLVDTSrvrItem.

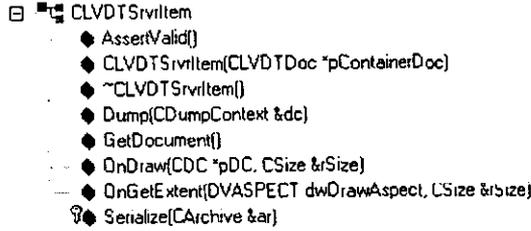


Figura B.8. Clase CLVDTSrvrItem.

Esta clase provee una interface con la aplicación del contenedor que abre el documento LVDT. Describe un documento completo que es embebido dentro de otro documento o una porción de un documento LVDT que es ligado a una parte del documento del contenedor. Esta clase también es creada por el asistente del Visual C++.

La figura B.9 muestra la clase CLVDTView.



La clase vista también es creada por el asistente del Visual C++, y al igual que la clase documento, es de las más importantes de la aplicación. En esta clase se programa la forma de mostrar los datos de la clase de documento. Aquí también se programa la edición de los datos. En nuestro caso, esta clase contiene casi todo el código de la aplicación LVDT.

Esta clase controla el puerto serie RS232 y la vista, es por esto que contiene la mayor parte del código del programa. Recuerde que la aplicación solamente recibe datos a través del puerto serie y muestra las lecturas en pantalla.

La figura B.10 muestra la clase CMainFrame.



Figura B.10. Clase CMainFrame.

La clase CMainFrame se encarga de construir y dar la vista a la ventana principal. Esta clase es creada por el asistente de Visual C++.

La figura B.11 muestra la clase CParamCom.

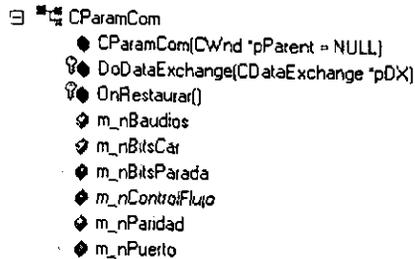
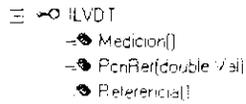


Figura B.11. Clase CParamCom.

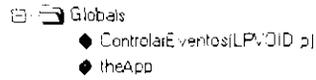
Esta clase se encarga de mostrar la caja de diálogo que se utiliza en la aplicación para modificar los parámetros del puerto serie.

La figura B.12 muestra las propiedades y métodos disponibles cuando la aplicación se utiliza como servidor.



**Figura B.12.** Propiedades y métodos de la aplicación LVDT.

La figura B.13 muestra las variables globales de la aplicación. Estas variables son conocidas por todas las clases del programa.



**Figura B.13.** Variables globales.

## ANEXO C

# Procesador digital de señales\*

En la actualidad se han desarrollado dispositivos digitales bastante rápidos, con mucha memoria y relativamente fáciles de usar. Dentro de estos dispositivos se encuentran los llamados procesadores digitales de señales (DSP's), que son procesadores especialmente diseñados para realizar operaciones rápidas y procesamiento en tiempo real. Por sus características, son cada vez más utilizados y han sustituido a algunos sistemas analógicos. Aunque en algunas áreas es imposible sustituir un sistema analógico, los DSP's han encontrado una buena aceptación dentro de un gran número de aplicaciones. La tabla C-1 muestra la mayoría de aplicaciones de los DSP's. El caso de la instrumentación para la metrología dimensional no es la excepción: este proyecto está buscando la forma de sustituir los tradicionales sistemas analógicos, utilizados en los sistemas de medición dimensional, por sistemas digitales con mejores características. Particularmente en aspectos como resolución, alcance, calibración, estabilidad contra el tiempo, estabilidad contra variaciones de temperatura y estabilidad contra la variación en los valores nominales de los componentes electrónicos. El presente capítulo describe las características de la familia de procesadores digitales TMS320C5x de Texas Instruments con la finalidad de proporcionar las ventajas que se tienen al usar uno de estos dispositivos. También se describe la tarjeta "DSP Started Kit" (DSK) utilizada en el proyecto y la forma de programar las operaciones aritméticas básicas en punto flotante, ya que los DSP's de esta familia son de punto fijo.

---

\* El presente anexo es un resumen de las referencias bibliográficas [12], [13] y [14]. Para obtener mayor información o una descripción más detallada de algún tema específico, se pueden consultar estas referencias.

Automotor	Consumo	Control
Teléfono celular Radio digital Control del motor Posicionamiento global Navegación Análisis de vibración Comando de voz	Radio/TV digital Juguetes educativos Sintetizador de música Radar detector Máquinas contestadoras	Control de unidades de disco Control de maquinaria Control de impresoras láser Control de motores Control de robots
Propósito general	Imagen/Gráficos	Industria
Filtrado adaptivo Convolución Correlación Filtrado digital Transformadas rápidas de Fourier Transformadas de Hilbert Generador de funciones	Rotación 3-D Animación digital Reconocimiento de patrones Compresión de imágenes Visión robótica	Control numérico Monitoreo de líneas de potencia Robótica Acceso de seguridad
Instrumentación	Medicina	Militar
Filtrado digital Generador de funciones PLL Procesamiento sísmico Análisis espectral Análisis transitorio	Monitoreo fetal Monitoreo de pacientes Equipo ultrasónico Equipo de monitoreo	Procesamiento de imágenes Guía de misiles Navegación Radar Radio frecuencia Seguridad en comunicaciones
Telecomunicaciones		Voz
Modems de 1200 a 19200 bps Ecuación Teléfono celular Interpolación digital de voz FAX Video conferencia		Reconocimiento de voz Síntesis de voz Correo de voz

**Tabla C.1.** Aplicaciones de los DSP's.

## C.1 Arquitectura

La unidad central de proceso de la familia C5x está basada en la unidad central del C25 con mejoras adicionales en la arquitectura. Los dispositivos de la generación C5x son capaces de ejecutar el doble de la velocidad de los dispositivos de la generación C2x. La generación C5x consiste de los siguientes dispositivos:

- C50. Es un procesador digital de señales CMOS con 10k palabras de RAM interna y 2k palabras de ROM interna.
- C51. Es un procesador digital de señales CMOS con 2k palabras de RAM interna y 8k palabras de ROM interna.
- C53. Es un procesador digital de señales CMOS con 4k palabras de RAM interna y 16k palabras de ROM interna.

La combinación de una arquitectura Harvard (buses separados para la memoria de datos y de programa), periféricos adicionales internos, memoria interna y un conjunto de instrucciones altamente especializados es la base de la flexibilidad de operación y velocidad de los dispositivos DSP.

Características del DSP TMS320C50:

- Ciclo de 35-50 ns. ejecución de instrucciones en punto fijo (28.6-20 MIPS).
- Operación basada en memoria RAM.
- RAM 9k x 16 bits de simple acceso interna para programa datos.
- ROM 2k x 16 bits interna.
- RAM 1056 x 16 bits de doble acceso interna.
- Máximo direccionamiento de memoria externa de 224k x 16 bits.
- Unidad aritmética y lógica (ALU) de 32 bits, acumulador (ACC) de 32 bits y acumulador buffer (ACCB) de 32 bits.
- Unidad lógica paralela (PLU) de 16 bits.
- Multiplicador de 16 x 16 bits con producto de 32 bits.

- Multiplicación/acumulación de un ciclo de instrucción.
- 8 registros auxiliares para direccionamiento indirecto.
- Pila de 8 niveles.
- Corrimiento de datos de 0 a 16 bits hacia la izquierda y derecha.
- Dos buffer circulares.
- Instrucciones de movimiento de bloques de memoria.
- Puerto serial síncrono full-duplex.
- Puerto serial de múltiple acceso por división de tiempo (TDM).
- 64k de puertos paralelos de entrada/salida, 16 son mapeados.
- Operación pipeline de cuatro niveles.
- Modo de direccionamiento de bit-reverso para FFT's radix 2.

La arquitectura de un DSP TMS320 consiste de tres segmentos básicos:

- Unidad central de proceso (CPU).
- Memoria.
- Circuitos periféricos-interface.

El C5x desempeña aritmética de complemento a 2 utilizando la unidad aritmética y lógica (ALU) y el acumulador (ACC). La ALU es una unidad aritmética de propósito general que utiliza palabras de 16 bits tomadas de la memoria de datos, entregadas de las instrucciones o del resultado de 32 bits del multiplicador. La ALU también desempeña operaciones booleanas. El resultado de las operaciones es almacenado en el ACC y éste es la segunda entrada de la ALU.

El C5x cuenta también con una unidad lógica paralela (PLU) que efectúa operaciones lógicas sobre los datos sin afectar el contenido del acumulador. La PLU provee la habilidad de manipular bits requerida por un controlador de alta velocidad y simplifica la habilitación y borrado de bits.

El multiplicador desempeña multiplicaciones de 16 x 16 bits en complemento a 2 con resultado de 32 bits en un ciclo de instrucción. Este multiplicador consiste de tres elementos: Arreglo del multiplicador, registro del producto (PREG) y registro temporal (TREG0) que almacena el

multiplicando. El valor del multiplicador proviene de la memoria de datos, de la memoria de programa o directamente de la instrucción. Este multiplicador interno permite al dispositivo desempeñar eficientemente operaciones fundamentales para el procesamiento digital de señales como la convolución, correlación y filtrado.

Ocho niveles de pila guardan el contenido del contador de programa durante las llamadas a subrutinas o interrupciones. En las interrupciones los registros estratégicos (ACC, ACCB, INDX, PMST, PREG, ST0, ST1, TREGs) son almacenados en una pila de un nivel y restaurados cuando se regresa de la interrupción.

La arquitectura del C5x se muestra en la figura C.1. Esta arquitectura está construida alrededor de dos buses principales: el bus de datos y el bus de programa. El bus de datos interconecta varios elementos tal como la unidad central aritmética y lógica (CALU) y los registros auxiliares a la memoria de datos. El bus de programa acarrea el código y los operandos inmediatos desde la memoria de programa. Juntos, los buses de datos y programa, pueden acarrear datos desde la memoria de datos interna y desde la memoria de programa interna o externa al multiplicador en un simple ciclo para operaciones de multiplicación/acumulación.

Los dispositivos C5x poseen un alto grado de paralelismo, esto es, mientras los datos están siendo operados en la CALU, operaciones aritméticas pueden ser ejecutadas en la unidad aritmética de registros auxiliares (ARAU). La unidad central aritmética y lógica se compone de un bloque de corrimiento de 16 bits, del multiplicador, de la unidad aritmética y lógica, un acumulador (ACC), un acumulador buffer (ACCB) y bloques adicionales de corrimiento a la salida del multiplicador y del acumulador.

El hardware interno del C5x ejecuta funciones que otros procesadores típicamente implementan en software o microcódigo, por ejemplo, la multiplicación de 16 x 16 bits en un ciclo de instrucción, corrimiento de datos y manipulación de direcciones.

Algunos de los bloques principales que conforman la arquitectura del C5x que se muestran en la figura C.1 son: **ACC**, un acumulador de 32 bits accesible en dos partes, la parte alta ACCH y la parte baja ACCL; **ACCB**, un registro utilizado para almacenar temporalmente el contenido del acumulador; **ALU**, una unidad aritmética y lógica con entradas y salida de 32 bits; **ARAU**, una unidad aritmética de 16 bits que realiza operaciones sin signo para calcular direcciones en modo indirecto utilizando los registros auxiliares; **AUXREGS**, un bloque que contiene ocho registros auxiliares de 16 bits utilizados como apuntadores para el direccionamiento indirecto; **ARP**, un registro de 3 bits que apunta al registro auxiliar seleccionado; **CALU**, un grupo formado por la ALU, el multiplicador y registros de corrimiento; **C**, un bit que almacena el acarreo a la salida de la ALU; **CBCR**

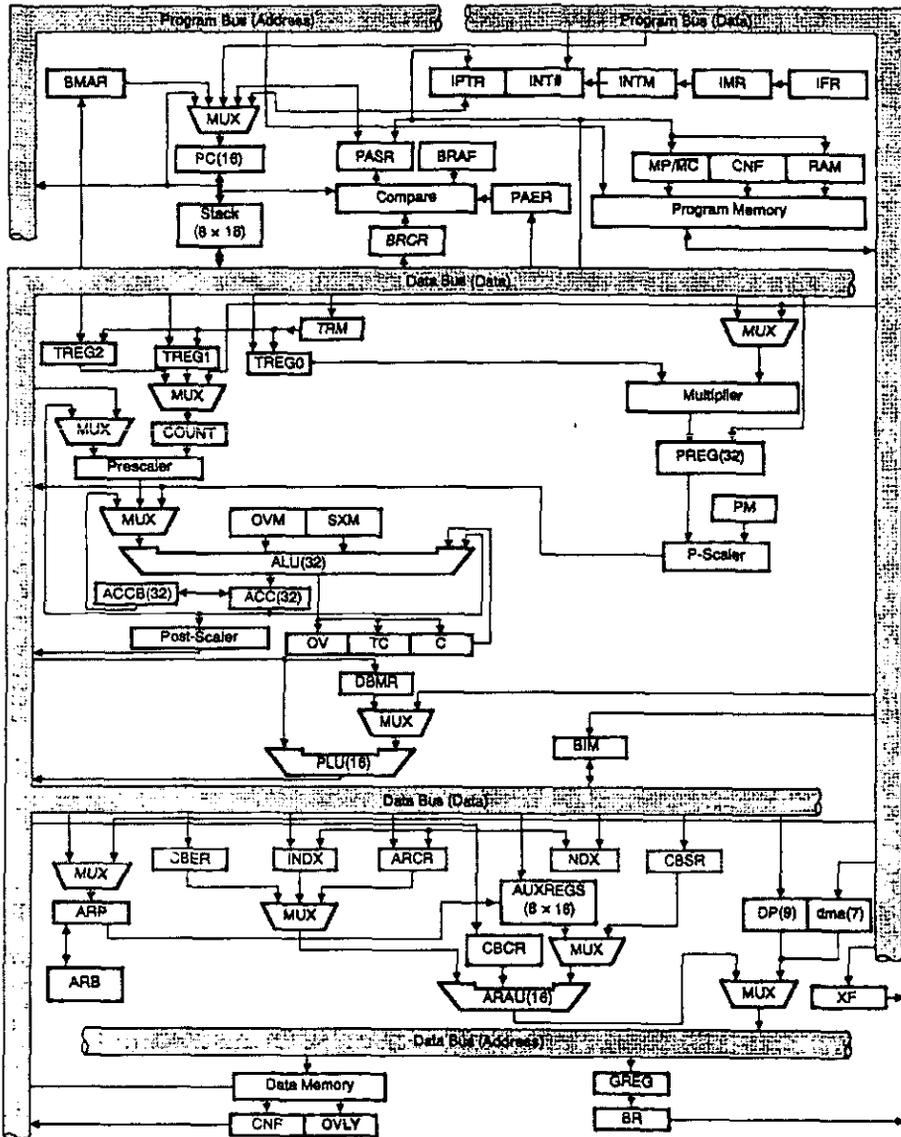


Figura C.1. Arquitectura interna del C5x.

un registro de 8 bits utilizado para habilitar/deshabilitar los buffer circulares. **CBBER1/2**, dos registros de 16 bits que indican la dirección final de los buffer circulares. **CBSR1/2**, dos registros de 16 bits que indican la dirección inicial de los buffers circulares. **DP**, un registro de 9 bits que

contiene la dirección de la página actual. Cada página es de 128 palabras y hay 512 páginas de memoria direccionable. **INDX**, un registro de 16 bits que especifica el tamaño del incremento mayor a 1 para el modo de direccionamiento indirecto. **OV**, un bit que indica el desbordamiento de una operación aritmética en la ALU. **PLU**, una unidad lógica paralela que ejecuta operaciones lógicas sobre la memoria de datos. **PM**, dos bits que definen el modo de corrimiento a la salida del multiplicador. **SXM**, un bit que controla el modo en que las operaciones aritméticas son realizadas, con signo o sin signo. **ST0**, **ST1**, **PMST**, tres registros de estado de 16 bits que contienen bits de estado y control. **TREG0**, Un registro de 16 bits que almacena temporalmente un operando del multiplicador. **TC**, un bit que almacena resultados de la ALU, PLU y de operaciones de bits.

## C.2 Organización

Veintiocho registros del procesador son mapeados dentro de la memoria de datos. Adicionalmente, existe memoria que es reservada para 64 registros mapeados en la página cero. Estas localidades de memoria de datos son reservadas para el control de periféricos y registros de los puertos de entrada salida.

Estos registros se muestran en la tabla C.2.

El rango de direccionamiento de memoria total es de 214k x 16 bits. El espacio de memoria está dividido en cuatro segmentos específicos: 64k de programa, 64k de datos locales, 32k de datos globales y 64k de puertos de entrada y salida. El paralelismo natural de la arquitectura de los dispositivos C5x permite al dispositivo desempeñar tres operaciones de memoria en cualquier ciclo de máquina dado: búsqueda de la instrucción, lectura del operando y escritura del operando.

El diseño de estos DSP's está basado en la arquitectura Harvard. Como ya se mencionó, esta arquitectura tiene espacios de memoria múltiple que pueden ser accedidos en tres buses paralelos, esto hace posible el acceso al programa y a los datos simultáneamente. Los tres buses paralelos son el bus de lectura escritura de programa (PAB), el bus de lectura de datos (DAB1) y el bus de escritura de datos (DAB2). Cada bus accesa a diferentes espacios de memoria para los diferentes aspectos de la operación del dispositivo. La figura C.2 muestra el mapa de memoria del C50.

Los dispositivos C5x incluyen una cantidad considerable de memoria interna. Como ya se mencionó, el C50 incluye 2k palabras de ROM, 9k palabras para programa datos de simple acceso (SARAM) y 1056 palabras

Nombre	Dirección	Descripción
-	0-3	Reservado
IMR	4	Registro de máscara de interrupción
GREG	5	Registro de localización global de memoria
IFR	6	Registro de banderas de interrupción
PMST	7	Registro de estado del modo del procesador
RPTC	8	Registro contador de repeticiones
BRCR	9	Contador de repetición de bloque
PASR	A	Dirección inicial de programa para bloque de
PAER	B	repetición
TREG0	C	Dirección final de programa para bloque de repetición
TREG1	D	Registro temporal para multiplicando
TREG2	E	Registro temporal para contador de corrimiento dinámico
DBMR	F	Registro temporal usado como apuntador de bit en
AR0-AR7	10-17	prueba dinámica de bit
INDX	18	Manipulación dinámica de bit
ARCR	19	Registros auxiliares
CBSR1	1 <sup>a</sup>	Registro índice
CBER1	1B	Registro auxiliar de comparación
CBSR2	1C	Dirección inicial del buffer circular 1
CBER2	1D	Dirección final del buffer circular 1
CBCR	1E	Dirección inicial del buffer circular 2
MAR	1F	Dirección final del buffer circular 2
		Registro de control del buffer circular
		Registro de dirección de movimiento de bloque
-	20-4F	Registros de periféricos mapeados en memoria
	50-5F	Puertos de entrada/salida mapeados en memoria

**Tabla C.2.** Memoria reservada del C5x. en la página cero.

de doble acceso (DARAM). El bloque de 9k de RAM de simple acceso puede mapearse en programa y/o espacio de datos, esta memoria requiere un ciclo completo de máquina para desempeñar una operación de lectura o de escritura. El bloque de memoria de RAM, de doble acceso, puede ser leído y escrito en el mismo ciclo. Las 1056 palabras de RAM de doble acceso están configuradas en tres bloques: El bloque 0 (B0) que es de 512 palabras en la dirección 0100h-02FFh en el espacio de memoria de datos o en la dirección 0FE00h-0FFFFh en espacio de programa, el bloque 1 (B1) es de 512 palabras en la dirección 0300h-04FFh en memoria de datos y el bloque 2 (B2) es de 32 palabras en la dirección 060h en la memoria de datos.

Hex	Programa	Hex	Datos	
0000	Interrupciones y Reservado (Interna)	0000	Registros mapeados	
002F				
0030	ROM Interna	0060	DARAM B2 interna	
07FF				
0800	SARAM Interna (RAM=1) Externa (RAM=0)	00FF	Reservada	
		0100	DARAM B0 interna (CNF=0)	
			Reservada (CNF=1)	
			02FF	
			0300	DARAM B1 interna
2BFF	Externa	04FF	Reservada	
2C00		0500		
		07FF	SARAM interna (OVLY=1)	
FDFE		0800		
FE00		2BBF	Externa	
		2C00		
FFFF		FFFF		

Figura C.2. Mapa de memoria del TMS320C50.

La organización de la memoria es muy importante para explotar al máximo las características del DSP. De ahí que un aspecto que se debe estudiar bien sean los modos de direccionamiento, ya que a veces la flexibilidad del código se encuentra relacionada con el acceso a memoria.

### C.2.1 Modos de direccionamiento

El C5x puede direccionar un total de 64k palabras de memoria de programa y 96k palabras de memoria de datos. Los datos utilizados como operandos de las instrucciones son obtenidos de una de las siguientes formas:

- Directamente de la dirección del bus utilizando el modo de direccionamiento directo relativo a la página de memoria apuntada por DP. En el modo de direccionamiento directo, DP apunta a una de las 512 páginas (cada página es de 128 palabras). La dirección de la memoria de datos apunta a la palabra deseada dentro de la página. Por lo tanto la dirección es formada concatenando los 9 bits del registro DP y los 7 bits de la instrucción.

- Utilizando el modo de direccionamiento de memoria mapeada dentro de la página cero. El modo de direccionamiento de memoria mapeada opera como el modo de direccionamiento directo excepto que los bits más significativos de la dirección son forzados a cero, no importando el contenido del registro DP. Esto permite al usuario acceder a los registros mapeados directamente sin tener que cambiar la página o el registro auxiliar.
- Utilizando el modo de direccionamiento indirecto. En este modo el registro auxiliar de 16 bits seleccionado apunta a la memoria de datos. Mientras el registro auxiliar seleccionado provee la dirección de memoria de datos y el dato es manipulado por la CALU, el contenido del registro auxiliar puede ser manipulado a través de la ARAU.
- Utilizando el registro de instrucción en el modo inmediato corto. El operador puede residir como parte de la instrucción en el código de máquina. En el caso del modo de direccionamiento inmediato corto, el operando está contenido en la instrucción de una palabra.
- Utilizando el contador de programa en el modo inmediato largo. En este caso el operador está inmediatamente después de la instrucción en la secuencia del programa. El operador en modo inmediato largo es de 16 bits.
- Utilizando el CPU accediendo a un registro en el modo de acceso a registro. El operador puede venir de un registro del CPU, este tipo de operador es utilizado en casos especiales. La CALU utiliza este modo al multiplicar un dato por TREG0.
- Utilizando la segunda palabra en el modo de direccionamiento inmediato largo. Un operando es direccionado por la segunda palabra de una instrucción de dos palabras. El PC es almacenado en un registro temporal y el valor inmediato largo es cargado en el PC que es restaurado del registro temporal una vez completada la instrucción. Esta técnica es usada cuando dos direcciones de memoria son requeridas por la ejecución de la instrucción.
- Utilizando el registro de dirección de bloque de memoria (BMAR) en el modo de direccionamiento del registro de bloque de memoria. Este modo trabaja como el modo de direccionamiento inmediato largo con la excepción de que la dirección proviene de BMAR. La ventaja de esta técnica es que la dirección del bloque de memoria puede ser cambiada durante la ejecución del programa.

El conjunto de instrucciones de lenguaje ensamblador del C5x soporta aplicaciones de DSP y de propósito general. La mayoría de instrucciones se ejecutan en un ciclo de instrucción, las instrucciones de salto condicional son las instrucciones más tardadas y utilizan 4 ciclos de instrucción para ejecutarse. El conjunto de instrucciones del C5x cuenta con instrucciones de suma, manipulación de bits, resta, multiplicación, multiplicación acumulación, operaciones lógicas, operaciones que facilitan la programación de aritmética en punto flotante, instrucciones para el movimiento de bloques de memoria, instrucciones para repetición sin retardos, instrucciones de prueba, etc. La arquitectura junto con el conjunto de instrucciones del C5x y el acceso a memoria hacen del DSP un dispositivo ideal para la programación de algoritmos complejos que necesitan de rapidez y flexibilidad o para algoritmos que necesitan hacer operaciones o procesamiento en tiempo real.

El estudio de un dispositivo DSP o incluso de cualquier microprocesador o microcontrolador es bastante extenso, es por eso que este texto solo revisa a grandes rasgos los aspectos que se consideran más importantes del DSP como son su arquitectura y su organización de la memoria para dar énfasis a la potencialidad del dispositivo. La operación del dispositivo en otros aspectos como interrupciones, puertos, subrutinas, etc., es en esencia muy parecida a cualquier dispositivo digital aunque se tienen diferencias porque el DSP fue diseñado para incrementar la eficiencia en todos los aspectos posibles. Por lo tanto el presente capítulo está muy lejos de explicar a fondo el manejo y operación del DSP. Si se quiere aprender a utilizarlo, es necesario consultar el manual proporcionado por el fabricante, en las referencias [12] y [15]. Un aspecto importante que debe estudiarse es la arquitectura externa del dispositivo así como los periféricos externos utilizados. El siguiente tema explica la configuración de la tarjeta utilizada y describe las características del convertidor analógico a digital utilizado, con esto se tendrá una idea clara de los recursos con los que se contaron para desarrollar el proyecto.

### **C.3 Tarjeta DSK**

Cualquier dispositivo digital programable debe tener una configuración externa mínima para poder funcionar. La Texas Instruments proporciona una tarjeta de aplicación para el DSP TMS320C50 llamada *DSP started kit* (DSK) [13]. La tarjeta cuenta con los recursos mínimos pero suficientes para operar el DSP. La figura C.3 muestra los componentes que conforman la arquitectura de la tarjeta DSK.

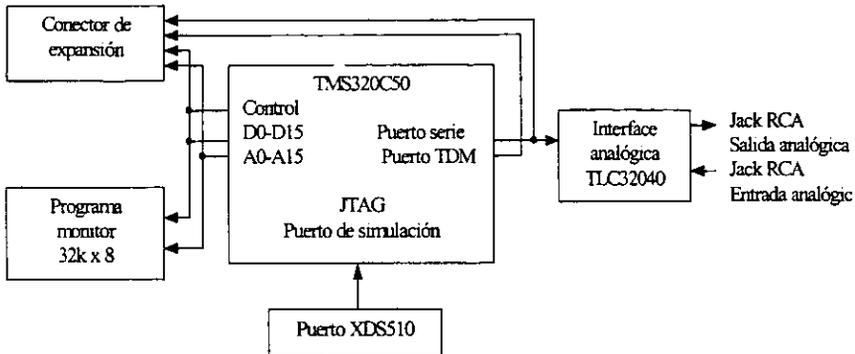


Figura C.3. Diagrama a bloques de la tarjeta DSK.

Características:

- DSP TMS320C50.
- Ciclo de instrucción de 50 ns.
- PROM de 32k bytes (Memoria de solo lectura programable).
- Adquisición analógica de datos con calidad de voz a través del TLC32040.
- Conectores estándar RCA para entrada y salida analógicas que proveen conexión directa con un micrófono y bocina.
- Conector de simulación XDS510.
- Bus de expansión de E/S para diseño externo.
- Conector RS232.

La figura C.3 muestra las conexiones de la tarjeta, la comunicación con la PC es a través del puerto RS232 en la tarjeta DSK. Los 32k bytes de PROM contienen el programa monitor. Como puede observarse todos los pines del C50 son conectados a interfaces externas de entrada/salida. Estas interfaces externas incluyen 4 conectores de 24 pines, un conector de 4 pines y un conector de 14 pines.

El circuito de interface analógica, TLC32040, interactúa con el TMS320C50 a través del puerto serie. Dos conectores RCA proveen entrada y salida analógicas a la tarjeta.

Como puede observarse en la figura C.3, la tarjeta DSK es una de las tarjetas de aplicación más simples: inclusive no tiene memoria externa disponible, los 19K de RAM interna del C50 proveen suficiente memoria para la mayoría de aplicaciones. La PROM sólo es para inicializar y no puede ser accesada después. Una porción de la memoria interna es reservada por el programa monitor. La figura C.4 muestra el mapa de memoria del DSK C50.

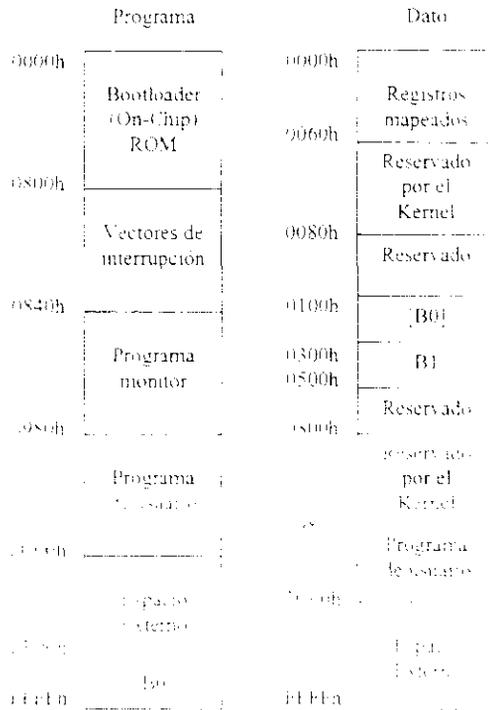


Figura C.4. Mapa de memoria del DSK C50

La memoria RAM de doble acceso interna B2 es reservada como buffer de los registros de estado. La memoria interna RAM de acceso simple es configurada como memoria de programa y de datos. El kernel (programa monitor) es almacenado en el área de 0x840h-0x980h. Los vectores de interrupción son localizados desde 0x800h.

El C50 DSK se comunica con la PC utilizando los pines XF y BIO a través del puerto serial RS232. La tarjeta también cuenta con su ensamblador y depurador que facilitan el desarrollo de software.

El circuito de interface analógica tiene las siguientes características:

multiplicación y división para facilitar la programación de operaciones más complejas como la FFT.

La diferencia básica entre un número en punto fijo y un número en punto flotante es que el número en punto fijo sólo cuenta con mantisa y el de punto flotante cuenta con mantisa y exponente. Las ventajas de representar a un número en punto fijo es que todos los bits se utilizan para la mantisa, así el número quedará representado con mayor precisión. Las ventajas de representar un número en punto flotante son que se tiene un intervalo más amplio de valores que puede tomar el número y que los números grandes y pequeños tienen la misma mantisa, por lo tanto, la misma precisión en su representación. El formato propuesto es el siguiente:

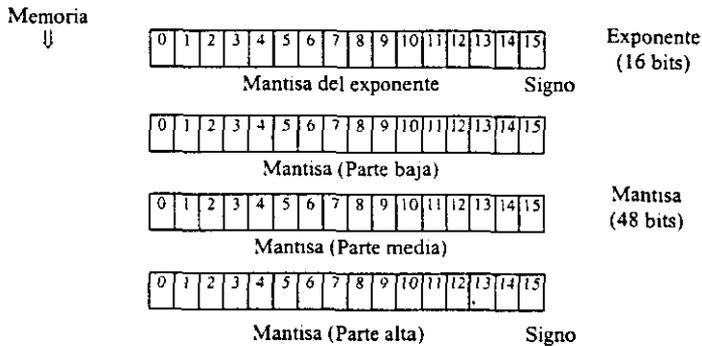


Figura C.6. Representación de un número en punto flotante a 64 bits.

La base del exponente es dos para facilitar las operaciones. También el formato elegido tiene exponente de 16 bits con signo, ya que el DSP trabaja con números de 16 bits signados y facilita la programación de las operaciones. Con este formato se pueden representar números positivos desde  $+0.99999999999999993 \cdot 2^{+32767}$  hasta  $+0.99999999999999993 \cdot 2^{-32768}$  o números negativos desde  $-1 \cdot 2^{32767}$  hasta  $-1 \cdot 2^{32768}$ .

Como puede observarse, la mantisa normalizada facilita la comprensión y por lo tanto el manejo de estos números. A continuación se explica la forma en que se programaron las operaciones aritméticas básicas con este formato.

### C.4.1 Operaciones aritméticas en punto flotante

Para realizar una suma en punto flotante primero hay que verificar cual número es menor en magnitud y desplazar la mantisa en dirección al bit menos significativo de ese número hasta igualar los exponentes. La

suma se realiza considerando que es un número de 3 palabras. El exponente final es el exponente mayor. El diagrama de flujo de la figura C-7 muestra el procedimiento para realizar una suma a 64 bits en punto flotante.

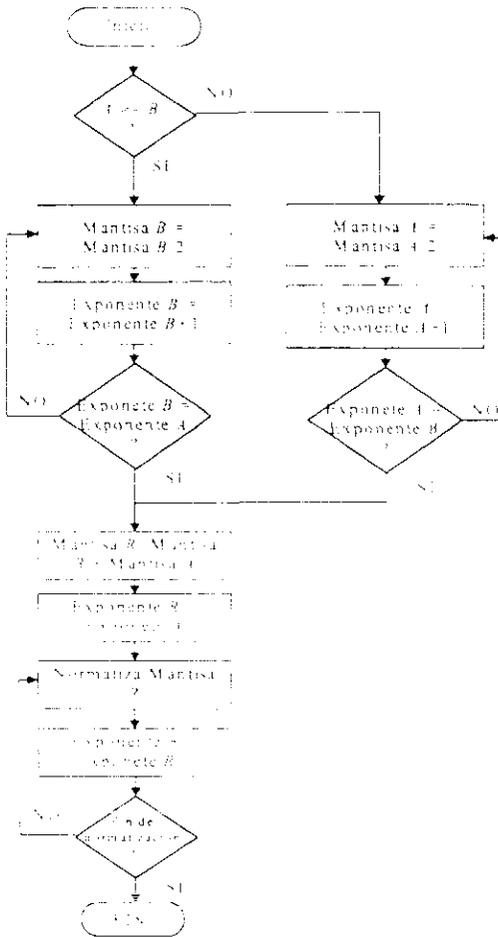


Figura C.7. Diagrama de flujo de la subrutina de suma.

Se suman dos números  $A$  y  $B$  y el resultado es almacenado en  $R$ . Nótese que al final de la suma el resultado tiene que normalizarse, esto es, se debe correr la mantisa hasta hacerla lo más cercano posible a 1. Si el resultado es distinto de cero, la mantisa siempre debe ser mayor o igual a 0.5. Con la normalización se evita la pérdida de precisión, resultando en otra ventaja de un número en punto flotante contra un número en punto fijo.

Para realizar una resta se realiza un procedimiento similar, lo único que cambia es la operación; las mantisas se restan en el orden deseado y el resultado también es normalizado. La ventaja de manejar este formato es que las mantisas están en complemento a 2 y esto facilita las operaciones aritméticas, no se tiene que verificar los signos para sumar o restar.

Una operación más complicada es la multiplicación de dos números de 64 bits en punto flotante, ya que se tienen que hacer operaciones con la mantisa y el exponente de cada número. Las mantisas se multiplican, el resultado se normaliza y los exponentes se suman. Como cada mantisa está en complemento a 2 y es de 3 palabras, la multiplicación debe realizarse de una manera especial: haciendo multiplicaciones parciales con signo y sin signo, ya que la parte media y baja de los números no tienen signo, para posteriormente sumarlas.

Para multiplicar un número signado con un número no signado se puede utilizar el algoritmo mostrado en la figura C.8.

Como puede observarse hay que multiplicar  $A$  y  $B$  como si fueren números con signo, y si el bit más significativo de  $B$  (el correspondiente al signo en  $A$ ) es 1, entonces añadir  $A$  al resultado para obtener el producto final.

Considere la siguiente representación para los números  $A$  y  $B$

$$A = -2^{15} A_{15} + 2^{14} A_{14} + \dots + 2^1 A_1 + 2^0 A_0$$

$$B = 2^{15} B_{15} + 2^{14} B_{14} + \dots + 2^1 B_1 + 2^0 B_0$$

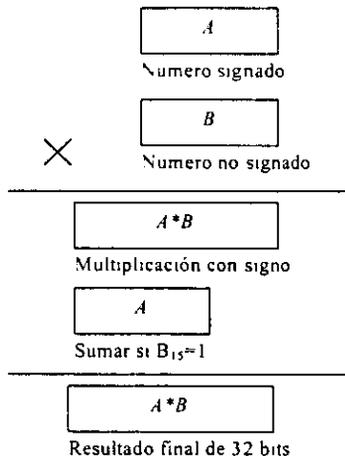


Figura C.8. Multiplicación de un número sin signo y uno con signo.

Si se multiplica  $A*B$  se tiene:

$$A * B = A * (2^{2j} B_{j+1} + 2^{2j} B_{j+2} + \dots + 2^j B_1 + 2^0 B_0) - 2^{2j} B_{j+1} A + 2^{2j} B_{j+2} A + \dots + 2^j B_1 A + 2^0 B_0 A \quad (C.1)$$

Pero si se realiza la multiplicación considerando que los dos números son signados se tiene:

$$A * B = -2^{2j} B_{j+1} A + 2^{2j} B_{j+2} A + \dots + 2^j B_1 A + 2^0 B_0 A \quad (C.2)$$

Si observamos las ecuaciones (C.1) y (C.2) vemos que son diferentes cuando  $B_{j+1} = 1$ . Para igualar las dos expresiones, se debe sumar dos veces  $2^{2j} B_{j+1}$ , es decir,  $2^{2j+1} B_{j+1}$  y el resultado será idéntico al de la ecuación (C.1).

Con este método se puede implementar la multiplicación de dos números de 3 palabras con signo, siguiendo el algoritmo que se muestra en la figura C.9.

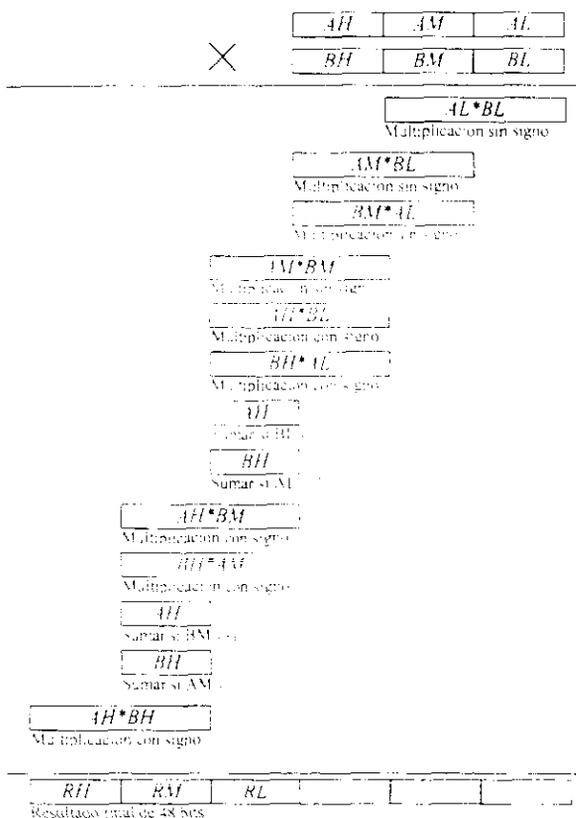


Figura C.9. Multiplicación de dos números de 3 palabras signados.

Nótese que el resultado final es sólo una parte del resultado total de la multiplicación, por lo tanto las primeras tres multiplicaciones pueden omitirse para agilizar el cálculo del producto, teniendo un error pequeño en los bits menos significativos. Sabiendo cómo se multiplican las mantisas de los dos números, se puede seguir el algoritmo mostrado en el diagrama de flujo de la figura C.10, para realizar una multiplicación de 64 bits en punto flotante.

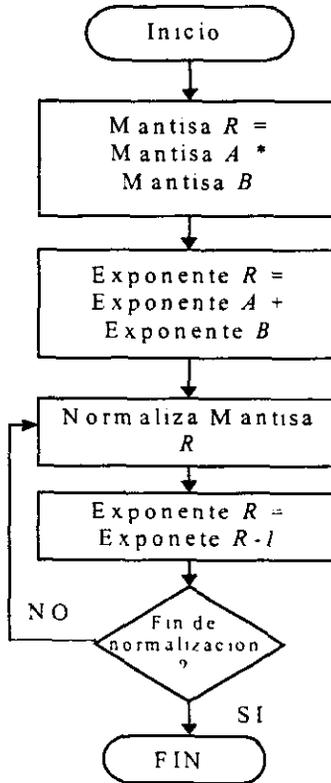


Figura C.10. Multiplicación de dos números de 64 bits en punto flotante.

Otra operación aritmética es la división, la cual se realiza de la forma más básica: primero se divide la mantisa y luego los exponentes se restan. Para dividir las mantisas se hace una iteración donde los dos números se restan hasta alcanzar el resultado. La rutina es similar a la de la multiplicación, la figura C.11 muestra el diagrama de flujo para la división.

Para dividir la mantisa se realiza un ciclo. Las veces que se realiza el ciclo depende de los bits que se quieran obtener de precisión, en este caso el ciclo se realiza 47 veces ya que la mantisa es de 47 bits más signo. En

cad. ciclo se verifica si el dividendo es mayor que el divisor; si es mayor se resta, se suma 1 al resultado final y el resultado de la resta es tomado en la siguiente iteración; si es menor no se hace nada. Al final de cada ciclo se duplican el dividendo y el resultado final. Cuando termina el ciclo se tiene el resultado final en el mismo formato. Cabe señalar que todas las operaciones (duplicar, suma y resta) son a 48 bits de precisión. El diagrama de flujo de la figura C.12 muestra la forma de programar el algoritmo de la división de dos números de 3 palabras ( $R=A/B$ ). Se considera que los números están en formato Q47 y el resultado queda en el mismo formato.

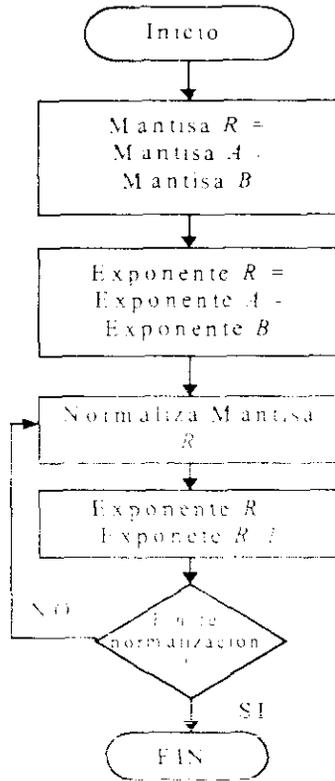


Figura C.11. División de dos números de 64 bits en punto flotante

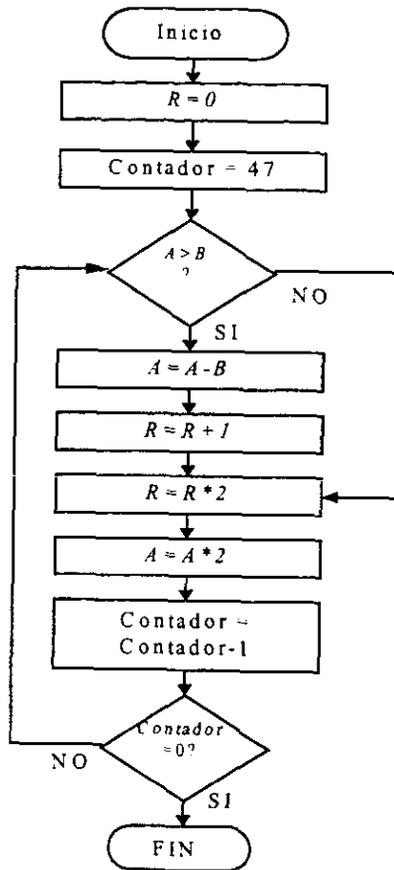


Figura C.12. División de dos números en Q47.

La última operación aritmética es la raíz cuadrada, ésta se realiza de una manera sencilla, elevando al cuadrado un número inicial y comparando para ver si no excedió al número del cual se quiere obtener la raíz. El diagrama de flujo de esta operación se muestra en la figura C.13.

El número del cual se quiere extraer la raíz se denota por  $A$ , y  $R$  es el resultado. El ciclo se realiza 47 veces porque es el número de bits de precisión de la mantisa. El número  $N$  comienza con un valor inicial que se obtiene al habilitar el bit más significativo, en este caso, el bit más significativo es el bit 46. Se utiliza una variable temporal llamada  $Temp$  para almacenar el resultado de la suma de  $N$  y la mantisa de  $R$ . Si el cuadrado de esta suma es menor que  $A$ , entonces el resultado es almacenado en la mantisa de  $R$ . Se hace un corrimiento a  $N$  de 1 bit a la derecha para el siguiente ciclo. Al final del ciclo se tiene la mantisa de la

raíz cuadrada de  $A$ . El exponente del resultado,  $R$ , se obtiene dividiendo entre dos el exponente de  $A$ . Sin embargo, si este exponente es impar, la mantisa se tiene que dividir entre  $\sqrt{2}$ , de igual forma, si el exponente es impar y es a su vez positivo, se tiene que sumar 1 al exponente final de  $R$ .

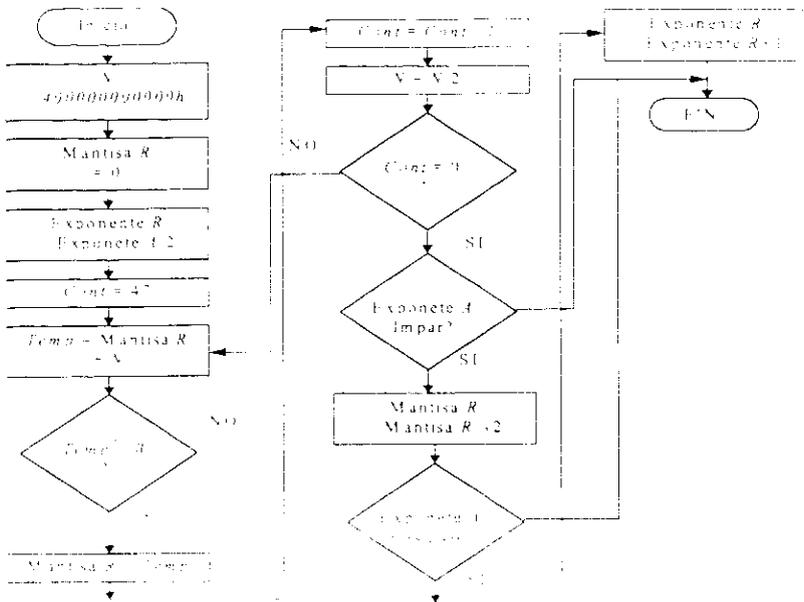


Figura C.13. Raíz cuadrada de un número a 64 bits en punto flotante.

Teniendo las subrutinas que realizan las operaciones aritméticas de suma, resta, multiplicación y división en el formato de 64 bits en punto flotante, ya se puede comenzar a describir la forma en la que se puede programar el algoritmo de la FFT DIT Radix 2.

### C.4.2 Transformada de Fourier (FFT) en punto flotante

Lo primero que hay que considerar es que el vector de valores de entrada es real y el resultado de la transformada de Fourier es complejo. La forma más fácil de asignar memoria para datos complejos, es copiar un arreglo continuo de partes reales en memoria a un arreglo discontinuo que considere las partes imaginarias de los datos y las inicie a cero. Cuando se pasan los valores a este otro arreglo, se puede hacer la decimación y así el arreglo con el que se trabajará ya estará decimado, con esto el resultado final de la subrutina de la FFT ya estará en el orden apropiado.

La operación más importante de la FFT es la mariposa. En realidad la subrutina de la FFT, después de la decimación, sólo son ciclos anidados que realizan la operación mariposa entre dos valores complejos. El resultado es almacenado en las mismas localidades de estos valores. En cada ciclo se calculan los índices de los números complejos que se van a utilizar en la mariposa y se calcula el índice del valor correspondiente del  $W_n$  que va a ser utilizado en la mariposa. Con esto debe suponerse que se necesita un vector de valores de  $W_n$  para poder calcular la transformada rápida de Fourier de un vector. Este vector  $W_n$  depende de la dimensión del vector de entrada, aunque la subrutina es general para cualquier  $N$  potencia de dos. La figura C.14 muestra el diagrama de flujo para programar el algoritmo de la FFT.

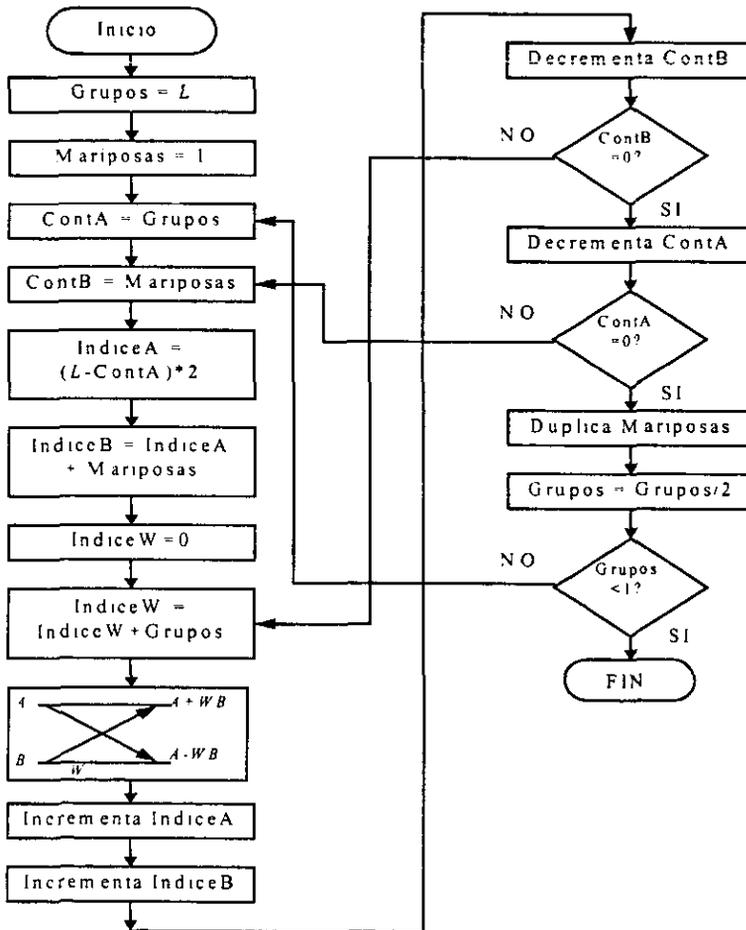


Figura C.14. Algoritmo para calcular la DIT FFT.

En el diagrama de flujo,  $L$  es el número de bits que son necesarios para representar la dimensión del vector, es decir,  $N=2^L$ , donde  $N$  es la longitud del vector. Nótese que antes de realizar el ciclo representado en el diagrama de flujo de la figura C.14, hay que hacer la decimación. Esto se puede implementar fácilmente ya que el DSP tiene instrucciones optimizadas para realizar esta operación particular. Utilizando direccionamiento indirecto, se pueden obtener los índices directos para la decimación especificando la longitud del vector. La salida de esta subrutina es el espectro en frecuencia del vector de entrada.

Programando estas pequeñas subrutinas en el DSP, es posible realizar un programa que procese las señales de entrada con mucha precisión para obtener los resultados esperados.

# BIBLIOGRAFÍA

- [1] Jack P. Holman. Métodos experimentales para ingenieros.  
4ª Edición. McGraw-Hill.
- [2] Harry N. Norton. Sensores y analizadores.  
Gustavo Gili, S.A. Colección Electrónica-Informática.
- [3] Ramón Pallás Areny. Sensores y acondicionadores de señal.  
2ª Edición. Marcombo.
- [4] Alan V. Oppenheim, Alan S. Willsky. Señales y sistemas.  
2ª edición. Prentice Hall.
- [5] Apuntes de la materia de Comunicaciones Analógicas.  
Profesor Jesús Reyes García
- [6] John G. Proakis, Dimitris G. Manolakis. Digital Signal Processing.  
Principles, algorithms and applications  
3<sup>rd</sup> Edition. Prentice Hall.
- [7] Herbert P. Neff. Continuous and Discrete Linear Systems.  
Krieger Publishing Company 1991
- [8] Bernard Sklar. Digital Communications. Fundamentals and  
applications.  
Prentice Hall.
- [9] Hwei P. Shu. Análisis de Fourier.  
Editorial Addison-Wesley Iberoamericana.

- [10] Valera Orozco Benjamín, Padilla Olvera Sergio y Ruiz Botello Gerardo. Informe de Calibración. Centro de Instrumentos UNAM. Sección de Metrología. Mayo de 1995.
- [11] Ronald E. Walpole, Raymond H. Myers. Probabilidad y Estadística. 4ª Edición. MacGraw Hill.
- [12] Texas Instruments. TMS320C5x User's Guide. 1993.
- [13] Texas Instruments. TMS320C5x DSP Started Kit. User's Guide. 1994.
- [14] Larry Escobar Salguero. Procesadores digitales de señales de la familia TMS320 de Texas Instruments y el TMS320C50. Facultad de Ingeniería, 1997..